
Highly Informative Analytical Platforms for Rapid, Non-Invasive Diagnosis and Stratification of Patients with Cancer

Thesis by
Ophir Vermesh

*In Partial Fulfillment of the Requirements
for the Degree of
Doctor of Philosophy*



CALIFORNIA INSTITUTE OF TECHNOLOGY

Pasadena, California

2011

(Defended August 30, 2010)

© 2011

Ophir Vermesh

All Rights Reserved

“Our truest life is when we are in dreams awake.”

-Henry David Thoreau

Acknowledgments

I wish to express my deepest gratitude to a number of people who have had an indelible impact on my graduate experience and without whom this dissertation would not have been possible. First of all, I would like to thank my advisor, Professor James Heath, for his vision and guidance. He has created a lab environment that encourages thinking outside the box and coming up with novel, big ideas to solve some of the most seemingly intractable scientific problems. It was truly a privilege to have access to the nearly unlimited resources of his first-rate lab, including a top-of-the-line in-house microfabrication facility. The Nanosystems Biology Cancer Center (NSBCC), which encompasses Caltech, UCLA, and Leroy Hood's Institute for Systems Biology (ISB), and which is headed by Jim, fosters the kind of interdisciplinary, collaborative environment that is needed to enable progress in these projects.

I would also like to thank Professor Paul Mischel (UCLA) and Dr. Tim Cloughesy (UCLA) for their prominent collaborative roles in our clinical trial on glioblastoma patients. We received all of our GBM patient blood samples from Dr. Cloughesy's neuro-oncology clinic. Dr. Cloughesy also provided a large amount of clinical information about each patient, allowing us to investigate differences in many clinical parameters within each of the cohorts he designated. Professor Mischel, with his broad and deep expertise in the glioblastoma field, helped us choose a suitable protein panel for our plasma assays, highlighted those aspects of the patient cohorts that were most interesting to investigate further, and helped us to fully appreciate the significance and potential impact of our results. I would also like to thank him for his generous gift in providing us with a full set of capture and detection antibodies when our first set ran out. Further, I would like to extend my gratitude to Dr. Alyssa Ziman for providing us with blood

samples from healthy individuals for our control experiments. I would also like to thank Tom Bourgeois and his colleagues at the Clinical Immunology Research Laboratory (CIRL) at UCLA for centrifuging and processing the hundreds of patient samples used in our study. Thanks also to Bruz Marzolf at ISB for printing our DNA microarray-spotted slides and for troubleshooting issues related to DNA-loading, spot morphology, spot-to-spot and slide-to-slide consistency, and so forth.

I had the distinct pleasure of working closely with my brother, Udi Vermesh, on the glioblastoma clinical trial. We both appreciated just how rare it is for two brothers to end up in the same graduate school, in the same lab, and working on the same project. Because our experiments and assays were long and arduous, often extending well into the night, it was a blessing to work side-by-side with Udi, with whom I could joke, share entertaining experiences, and take breaks to go to the gym. Our shifted work schedules also matched up quite well. I also had the pleasure of collaborating with Dr. Rong Fan (now an Assistant Professor at Yale), with whom I co-authored the *Nature Biotechnology* paper “Integrated barcode chips for rapid, multiplexed analysis of proteins in microliter quantities of blood.” The paper could not have had the impact it did without Rong’s impressive clinical experiments and his method for patterning DNA at high-density. I would also like to thank Dr. Brian Yen, who was instrumental in the early development of the blood separation chip. He wrote a program in MATLAB that optimized the lengths and widths of the various microfluidic device channels to allow for a high degree of plasma separation efficiency. His experience with microfluidics was extremely helpful in guiding various aspects of the design and fabrication process. Thank you to Marino DiFranco, our undergraduate SURF student, for writing the various batch files we used to automate our statistical analysis. Thanks also to Alok Srivastava (ISB) for his help in the early DNA, antibody,

and DEAL conjugate validation experiments. I would also like to thank Chao Ma and Kiwook Hwang for their help with the tissue engineering project (not described herein).

Other colleagues and good friends who I would like to thank are Gabe Kwong, Tiffany Huang, and Shawn Sarkaria. My conversations with Gabe greatly enhanced my knowledge of and intensified my interest in biomedical technology development, and inspired me to become innovative in my own right. He also greatly assisted me in the early stages of the tissue engineering project. From Tiffany, I learned an enormous amount about GBM as well as about the various experimental methods used in cellular and molecular biology. Both Tiffany and Shawn (Mischel Lab) were instrumental in identifying a panel of proteins with high biological relevance to glioblastoma and to cancer generally. Our multiplexed assay platform was designed to target the 35 proteins they chose. In addition, they optimized the procedure for conjugating antibodies to DNA and contributed purified conjugates for our experiments.

I would also like to thank Mike McAlpine (now an Assistant Professor at Princeton) and Dr. Slobodan Mitrovic for many exciting and entertaining conversations about science and other topics throughout the years. I would like to thank Diane Robinson for her administrative help, for being a good friend, and for just being fun. I would also like to thank Kevin Kan for ensuring the smooth operation of the clean room facility and, more recently, the biology lab. Thanks to Jackie Barton, Jack Beauchamp, and Nate Lewis for serving on my thesis committee. Finally, I would like to thank my parents for their love and support throughout the years, and helping to shape me into who I am today.

Abstract

As the tissue that contains the largest representation of the human proteome, blood is the most important fluid for clinical diagnostics. However, although changes of plasma protein profiles reflect physiological or pathological conditions associated with many human diseases, only a handful of plasma proteins are routinely used in clinical tests. Reasons for this include the intrinsic complexity of the plasma proteome, the heterogeneity of human diseases and the rapid degradation of proteins in sampled blood. The first part of this thesis reports an integrated microfluidic system, the integrated blood barcode chip (IBBC) that can sensitively sample a large panel of protein biomarkers over broad concentration ranges and within 10 minutes of sample collection. It enables on-chip blood separation and rapid measurement of a panel of plasma proteins from quantities of whole blood as small as those obtained by a finger prick. The device holds potential for inexpensive, noninvasive and informative clinical diagnoses, particularly in point-of-care settings.

Proteomic approaches, on which the IBBC platform is based, have shown great promise in recent years for correctly classifying and diagnosing cancer patients. However, no large antibody-based microarray studies have yet been conducted to evaluate and validate plasma molecular signatures for detection of glioblastoma and monitoring of its response to therapy. In the second part of this thesis, plasma samples from 46 glioblastoma patients (72 total samples) are compared with those of 47 healthy controls with respect to the plasma levels of 35 different proteins known to be generally associated with tumor growth, survival, invasion, migration, and immune regulation. Average-linkage hierarchical clustering of the patient data stratified the two groups effectively, permitting accurate assignment of test samples into either GBM or healthy

control groups with a sensitivity and specificity as high as 90% and 94%, respectively (when test samples within unbiased clusters were removed). The accuracy of these assignments improved (sensitivity and specificity as high as 94% and 96%, respectively) when the cluster analysis was repeated on increasingly trimmed sets of proteins that exhibited the most statistically significant ($p < 0.05$) differential expression. The diagnostic accuracy was also higher for test samples that fell into more homogeneous clusters. Intriguingly, test samples that fell within perfectly homogeneous clusters (all members belonging to the same group) could be diagnosed with 100% accuracy. Using the same 35-protein panel, we then analyzed plasma samples from GBM patients who were treated with the chemotherapeutic drug Avastin (Bevacizumab) in an effort to stratify patients based on treatment-responsiveness. Specifically, we compared 52 samples from (25) patients who exhibited tumor recurrence with 51 samples from (21) patients who did not exhibit recurrence. Again, several proteins were highly differentially expressed and cluster analysis provided effective stratification of patients between these two groups (sensitivity and specificity of 90% and 96%, respectively).

Table of Contents

Acknowledgments	iv
Abstract	vii
Table of Contents	ix
Table of Figures	xii
List of Tables	xiii
1 Introduction.....	1
1.1 Blood: The Most Information-Rich Biological Fluid	1
1.2 Proteomic Technologies	2
1.3 On-Chip Plasma Separation and Detection	4
1.4 Thesis Overview.....	5
1.5 References.....	7
2 Integrated Barcode Chips for Rapid, Multiplexed Analysis of Proteins in Microliter Quantities of Blood.....	9
2.1 Introduction	9
2.2 Experimental Methods	11
2.2.1 Micropatterning of Barcode Array	11
2.2.2 Fabrication of IBBCs	12
2.2.3 Clinical Specimens of Cancer Patient Sera	12
2.2.4 Collecting a Finger Prick of Blood	13
2.2.5 Execution of Blood Separation and Plasma Protein Measurement using IBBCs	13
2.2.6 Quantitation and Statistics	14
2.3 Results and Discussion	15
2.4 References	26
2.5 Appendix A: Supplementary Methods	28
2.5.1 DNA-Encoded Antibody Libraries (DEAL) Technique	28
2.5.2 Serum Protein Biomarker Panels and Oligonucleotide Labels	29
2.5.3 Cross-Reactivities of Oligonucleotide Labels	32
2.5.4 Patterning of Barcode Arrays	33
2.5.5 Fabrication of IBBCs	41

2.5.6	Execution of Blood Separation and Multi-Parameter Protein Assay using IBBCs	42
2.5.7	Consideration of Microfluidic Environment for Rapid Immunoassay ...	45
2.6	Appendix B: Supplementary Data	47
2.6.1	Blind Test of Serum Samples Containing Unknown hCG Concentrations	47
2.6.2	Protein Cross-Reactivities	48
2.6.3	Dilution Curves for all Proteins used in the DEAL Barcode Assay	48
2.6.4	Standardized Quantification of the Patient Serum DEAL Barcode Data	50
2.6.5	ELISA Validation of DEAL Barcode Assay	54
2.6.6	Cancer Patients: Medically-Relevant Information	56
2.7	Additional References	57
3	Plasma Proteome Profiling of Glioblastoma Multiforme: <i>Characterizing Biomarker Signatures of Disease and Treatment Response</i>	58
3.1	Introduction	58
3.2	Experimental Methods	65
3.2.1	DNA-Encoded Antibody Libraries (DEAL) Technique	65
3.2.2	Antibody Array Platform	65
3.2.3	Multiplexed Assays on Patient Plasma	66
3.2.4	Plasma Collection and Processing	67
3.2.5	Data Processing and Statistics	67
3.2.6	Classification of Patients	68
3.3	Results	71
3.3.1	Evaluation of DNA-Directed Antibody Microarrays	71
3.3.2	Classification of GBM Patients versus Healthy Controls	73
3.3.3	Diagnostic Strength as a Function of Protein Panel Size	76
3.3.4	GBM Patients on Avastin – Classification of Tumor Growth vs. No Growth	79
3.4	Discussion	85
3.5	Appendix: Supplementary Information	92
3.5.1	DNA-Encoded Antibody Libraries (DEAL) Technique	92
3.5.2	Serum Protein Biomarker Panels and Oligonucleotide Labels	92
3.6	References	97

4	Computational and Analytical Tools for Diagnostic Measurements.....	100
4.1	Automation of Data Processing and Analysis	100
4.2	Average-Linkage Hierarchical Clustering	109
4.3	Test Sample Classification: “Guilt by Association”	112
4.4	Appendix: Excel Macros for Data Analysis	115
4.4.1	Processing GenePix-Scanned Array Data to Create a Master Dataset ..	115
4.4.2	Graphing Patient Data from the Master Dataset	140
4.4.3	File Preparation for Cluster Analysis and Diagnostic Testing	152
4.4.4	Assessing the Diagnostic Performance of “Guilt-by-Association” Classification of Test Samples within Hierarchical Clusters	170
4.4.5	Macros for Working with AnalyseIt	174
4.4.6	User Interface Macros	176
4.4.7	String Manipulations	179
4.4.8	Other Useful Macros	182
4.4.9	Batch Files for Running Cluster 3.0 and Java Treeview	186

Table of Figures

Figure 1	Composition of 1 milliliter of whole blood.....	2
Figure 2.1	Design of an integrated blood barcode chip (IBBC).....	10
Figure 2.2	Measurement of human chorionic gonadotropin (hCG) in sera.....	17
Figure 2.3	Multiplexed protein measurements of clinical patient sera.....	21
Figure 2.4	IBBC for the rapid measurement of a panel of serum biomarkers from a finger prick of whole blood.....	25
Figure 2.5	Schematic depiction of multi-parameter detection of proteins in integrated microfluidics using the DNA-Encoded Antibody Library (DEAL) technique....	29
Figure 2.6	Cross-hybridization assay for all 13 DNA oligomer pairs that were used for encoding the registry of antibody barcode arrays.....	33
Figure 2.7	Microchannel-guided flow patterning of DEAL barcode arrays.....	35
Figure 2.8	Effects of polylysine coating on DEAL assay.....	36
Figure 2.9	Increased sensitivity observed in immunoassays run on DEAL barcode arrays.....	38
Figure 2.10	Schematic of human plasma proteome.....	40
Figure 2.11	AutoCAD design of an IBBC.....	40
Figure 2.12	Blind test of hCG-containing unknown samples.....	47
Figure 2.13	A cross-reactivity assay for all the biomarker panel of 12 proteins.....	49
Figure 2.14	Dilution curves for the 12 proteins measured using DEAL-based barcodes entrained within microfluidic channels.....	49
Figure 2.15	Comparison of fluorescence intensities quantitated using GenePix 6.0 and NIH ImageJ	51
Figure 2.16	Quantitation of the fluorescence intensities from measurements of 11 breast cancer patients.....	53
Figure 2.17	Quantitation of the fluorescence intensities from measurements of 11 prostate cancer patients.....	53
Figure 2.18	Validation of PSA detection using ELISA.....	55
Figure 3.1	Assay platform and methodology.....	72
Figure 3.2	Classification of GBM patients vs. healthy controls.....	76
Figure 3.3	Diagnostic strength vs. protein number for “GBM vs. Healthy Control” cohort.....	77
Figure 3.4	Classification of GBM patients on Avastin – tumor growth vs. no growth.....	83
Figure 3.5	Diagnostic Accuracy of the Candidate Biomarkers, TGFβ1 and HGF, separately and together.....	84

Figure 3.6	Test for DNA cross-hybridization.....	96
Figure 4.1	Master patient dataset: organization of clinical information.....	103
Figure 4.2	The “Run Analysis” command button in the Excel add-ins toolbar.....	106
Figure 4.3	The “ClusterPrep” user interface.....	107
Figure 4.4	Illustration of clustering by visually grouping patient samples based on protein profile similarities.....	111
Figure 4.5	Classifying test samples via “Guilt by Association”: illustrative examples.....	114

List of Tables

Table 2.1	List of Proteins and Corresponding DNA Codes.....	30
Table 2.2	List of DNA Sequences used for Spatial Encoding of Antibodies.....	31
Table 2.3	Cancer Patients: Selected Information.....	56
Table 3.1	GBM Patients vs. Healthy Control Population Characteristics.....	63
Table 3.2	Avastin-Treated GBM Patients: Characteristics of Patient Population with and without Tumor Recurrence.....	64
Table 3.3	List of Proteins and Corresponding DNA Codes.....	93
Table 3.4	List of DNA Sequences used for Spatial Encoding of Antibodies.....	93
Table 3.5	Antibody Vendors and Catalogue Numbers.....	95

1 Introduction

1.1 Blood: The Most Information-Rich Biological Fluid

In addition to its essential role in transporting oxygen to the various organs of the body, blood is an extremely accessible and incredibly informative diagnostic fluid. Because all tissues and organs are vascularized, blood is infused with biomolecular clues that can potentially report on the physiological or pathological state of cells throughout the body. However, because blood is a complex fluid consisting of a large and varied cellular component and immense biochemical diversity, it presents key analytical challenges that must be met in order to successfully obtain pertinent information. As summarized in **Figure 1**, cells, including erythrocytes (red blood cells), leukocytes (white blood cells), and thrombocytes (platelets), constitute approximately 45% of the total blood volume, and the liquid component, or plasma, constitutes the remainder. Most diagnostic analyses on blood are typically carried out on the cell-free plasma component because the cells would otherwise introduce sample instability and matrix effects that would interfere with accurate analyte quantitation. Plasma is in itself an extraordinarily diverse medium that contains well over 100,000 different proteins spanning 12 orders of magnitude in concentration (10^{11} - 10^0 pg/mL).¹ As a result, for some analytical methods, high abundance plasma proteins, such as albumin and immunoglobulins, must be removed in order to detect low abundance proteins such as cytokines. In addition to these commonly discussed components, additional diagnostic indicators in blood include circulating tumor cells (CTCs),²⁻⁵ DNA and even RNA,

which was long thought to be too unstable, especially in the ribonuclease-rich environment of the blood, to have any real diagnostic value.

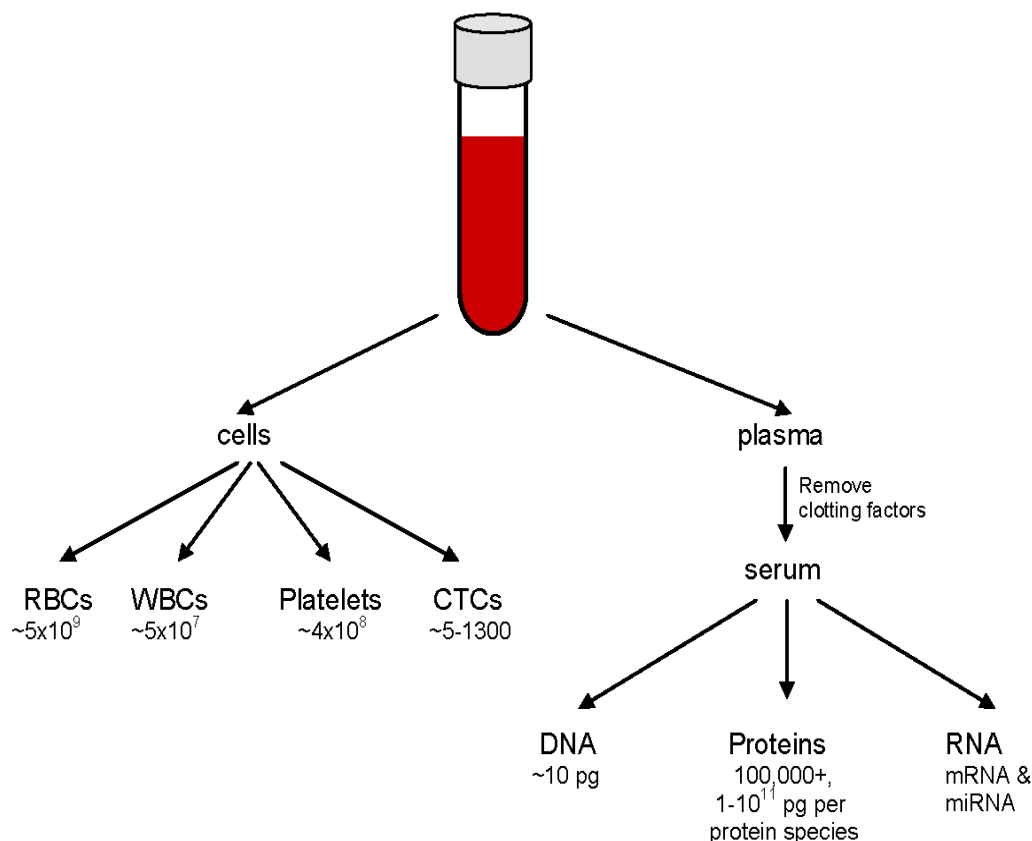


Figure 1 Composition of 1 milliliter of whole blood. (RBC=Red Blood Cell; WBC=White Blood Cell; CTC=Circulating Tumor Cell).

1.2 Proteomic Technologies

Because the plasma proteome is incredibly rich in organ- and disease-specific biomolecular detail, there has been a tremendous effort over the last decade to characterize multi-protein signatures for disease diagnosis rather than relying on single biomarkers.⁶ Many studies have shown that multi-protein signatures can diagnose cancer and other diseases with greater specificity and sensitivity than single biomarkers.¹ However, these efforts have also

revealed a number of inadequacies in current proteomics technologies that have hindered their routine use for clinical diagnostic purposes. For example, 2-D poly(acrylamide) gel electrophoresis is a valuable research tool that was among the first technologies used for separation and detection of plasma proteins. However, the approach is too sample-intensive, low throughput, insensitive, and laborious for proteomic characterization of sample-limited clinical assays.⁷

Several versions of mass spectrometry have demonstrated the utility of multiplexed analyses, though they have not yet achieved broad clinical application. Surface-Enhanced Laser Desorption/Ionization Time of Flight Mass Spectrometry, SELDI-TOF/MS (substrates are commercially marketed by Bio-Rad under the name ProteinChip), are chromatographic surfaces that can be treated to have diverse affinities⁵ to capture a subset of proteins based on their hydrophobicity or charge, amongst other chemical/physical properties.⁶ Retained proteins are then identified by mass spectrometric analysis. Increased specificity can be realized by coating the SELDI surfaces with antibodies to proteins of interest.⁷ Several reports in the literature have validated SELDI-TOF mass-spectrometric signatures for the plasma detection of prostate⁸ and breast^{9,10} cancer. Furthermore, the technique has shown promise for evaluating therapeutic responsiveness to chemotherapies.¹¹ The advantages of SELDI are its small sample volume requirements ($\sim 20 \mu\text{L}$),¹² high sensitivity,¹³ and speed. In addition, because patient classification is based on a differential spectral signature, it is not essential to be able to identify the actual proteins that are differentially expressed. However, the technology is limited in clinical utility by a lack of reproducibility,¹⁴ sample processing time, and instrument expense.

A plasma proteomic technique that has shown great promise in recent years from the perspectives of multiplexing capability, throughput, and sensitivity is the protein microarray.¹⁵

Potentially, thousands of antibodies and/or proteins can be arrayed on a single protein microarray slide. In addition, the technology can be coupled with existing amplification techniques, such as gold or silver amplification or with rolling circle amplification (RCA), to greatly enhance sensitivity. Protein and antibody-based microarrays have been utilized to identify biomarkers for early diagnosis of epithelial ovarian cancers,^{16,17} as well as for classification of patients with autoimmune disease and cancers of the prostate, bladder, pancreas, and stomach. In the vast majority of cases, a panel of differentially expressed plasma markers demonstrated significantly improved diagnostic accuracy as compared to each component protein. This technology is expected to continue to grow as the antibody repertoire becomes more comprehensive and as antibodies to different isoforms and post-translational modifications of proteins become available.

1.3 On-Chip Plasma Separation and Detection

The goals of separating plasma from whole blood on-chip are to be able to scale down sample volumes, increase sample processing speed, decrease the time from sample collection to detection, and avoid the variability associated with typical sample handling procedures. A number of strategies for separating plasma from whole blood on-chip have been reported in the last decade, including filtration, on-chip centrifugation, lattice sorting, and plasma skimming.

For direct filtration, size-exclusion barriers are lithographed within microfluidic channels such that cells, which are larger than openings in the barrier, are blocked from entering, while plasma passes through freely. The major disadvantage of direct filtration methods is clogging of the on-chip filter within a relatively short period of time.¹⁸ By comparison, cross-flow filtration,

in which the fluid is filtered transverse to its direction of flow, is more resistant to clogging and can extend device longevity.¹⁹⁻²¹

Using the principle of deterministic lateral displacement, Davis *et al.* created an array of posts with specifically defined post-to-post and row-to-row distances, in order to “bump” cells that are above a critical hydrodynamic diameter along the angle of the array, while particles below this critical diameter flowed straight through the device without being displaced laterally.²² In this manner, cells of different sizes could be separated from each other, and cell-free plasma could be obtained with almost no dilution of plasma and at a volume flow rate of 1 μ L/minute. In the lab-on-a-disk method for plasma separation, metering chambers, siphon-based hydrophilic extraction channels, plasma collection and detection chambers were positioned at different radial distances on a CD-sized disk and fluid flow between the chambers could be controlled by adjusting the spin speed.^{23,24}

An alternative method for plasma separation involves plasma skimming, wherein differential flow rates at the bifurcation of a channel are utilized to preferentially guide cells into one daughter channel while cell-free plasma enters the other. Cells travel preferentially into the higher flow-rate channel due to the pressure gradient and shear forces created by the flow-rate differential. Yang and co-workers have described several devices that are able to obtain plasma purities approaching 100%.^{25,26}

1.4 Thesis Overview

All of the techniques described to this point are accompanied by the limitations stemming from the inherent instability of proteins. Any degree of sample manipulation or purification therefore introduces error into the analysis.⁶ Direct methods of analysis, while difficult to

implement, have distinct advantages. One such direct analytical platform, the Integrated Blood Barcode Chip (IBBC) described in **Chapter 2**, integrates plasma separation and target detection analysis onto a single chip for rapid and unadulterated plasma proteomic analysis.²⁷ Plasma skimming is utilized to achieve on-chip plasma purification, while multiplexed protein analysis is achieved via the DNA-Encoded Antibody Library (DEAL) technology.²⁸ Eight different biomarkers, corresponding to liver, prostate, and immune function, can be assayed in each lane within only ten minutes, using as little as a fingerprick of blood. In a separate experiment on pre-centrifuged plasma from breast- and prostate-cancer patients, patterns of biomarker up- and down-regulation could be discerned that distinguished the two cancers as well as subgroups of patients having the same cancer.

In **Chapter 3**, the assay panel is expanded to 35 oncologically-relevant proteins in order to define a plasma biomolecular signature that can distinguish patients with glioblastoma multiforme, the most aggressively malignant of brain tumors, from healthy controls. The panel is also used to identify a signature that can stratify GBM patients based on their responsiveness to chemotherapy (i.e. the anti-VEGF monoclonal antibody, Avastin). In both cases, the differential expression of a number of proteins yielded excellent clustering of patients into separate experimental and control groups, allowing for highly accurate classification and diagnosis of test samples. Although these studies were conducted within ELISA-like wells rather than on-chip, detection of the validated biomarker set could easily be accomplished within an IBBC. The ability to quickly run this many protein assays in parallel from a single drop of blood would be expected to significantly reduce assay costs by minimizing reagent requirements and labor. In the future, it could also facilitate dynamic biomarker monitoring by allowing assays to be performed on a minute-by-minute or hourly basis, with minimal blood loss or discomfort to the patient.

In **Chapter 4**, the computational and analytical tools that were developed in-house to quickly process and analyze large data sets are introduced and described in detail. The software takes as its input the intensity values acquired from the fluorescent scans of the assayed slides. The output files consist of statistical analyses and graphs of the entire data set, as well as files that interface in automated fashion with Excel statistics software, *Cluster 3.0*, and Java *TreeView* in order to create cluster maps for patient classification and diagnostic testing. The software also affords relatively quick and straightforward analysis of differential protein expression between experimental and control groups, thereby facilitating data-mining for disease biomarkers.

1.5 References

- 1 Anderson, N. & Anderson, N. The human plasma proteome. *Molecular & Cellular Proteomics* **1**, 845 (2002).
- 2 Ashworth, T. A case of cancer in which cells similar to those in the tumours were seen in the blood after death. *Aust Med J* **14**, 169-174 (1869).
- 3 Wittekind, C. & Neid, M. Cancer invasion and metastasis. *Oncology* **69**, 14-16 (2005).
- 4 Paterlini-Brechot, P. & Benali, N. Circulating tumor cells (CTC) detection: clinical impact and future directions. *Cancer Letters* **253**, 180-204 (2007).
- 5 Nagrath, S. *et al.* Isolation of rare circulating tumour cells in cancer patients by microchip technology. *Nature* **450**, 1235-1239 (2007).
- 6 Rosenblatt, K. P. *et al.* Serum Proteomics in Cancer Diagnosis and Management. *Ann. Rev. Medicine* **55**, 97-112 (2004).
- 7 Wulfskuhle, J. D., Liotta, L. A. & Petricoin, E. F. Proteomic Applications for the Early Detection of Cancer. *Nature Reviews: Cancer* **3**, 267-276 (2003).
- 8 Adam, B. L. *et al.* Serum protein fingerprinting coupled with a pattern-matching algorithm distinguished prostate cancer from benign prostate hyperplasia and healthy men. *Cancer Res.* **62**, 3609-3614 (2002).
- 9 Li, J., Zhang, Z., Rosenzweig, J., Wang, Y. Y. & Chang, D. W. Proteomics and bioinformatics approaches for identification of serum biomarkers to detect breast cancer. *Clin. Chem.* **48**, 1296-1304 (2002).
- 10 Hu, Y., Zhang, S., Yu, J., Liu, J. & Zheng, S. SELDI-TOP-MS: The proteomics and bioinformatics approaches in the diagnosis of breast cancer. *Breast* **14**, 250-255 (2005).
- 11 Pusztai, L. *et al.* Pharmacoproteomic analysis of prechemotherapy and postchemotherapy plasma samples from patients receiving neoadjuvant or adjuvant chemotherapy for breast carcinoma. *Cancer* **100**, 1814-1822 (2004).
- 12 Abramovitz, M. & Leyland-Jones, B. A systems approach to clinical oncology: Focus on breast cancer. *Proteome Sci.* **4**, 1-15 (2006).

- 13 Srinivas, P. R., Verma, M., Zhao, Y. & Srivastava, S. Proteomics for Cancer Biomarker Discovery. *Clin. Chem.* **48**, 1160-1169 (2002).
- 14 Coombes, K. R., Morris, J. S., Hu, J., Edmonson, S. R. & Baggerly, K. A. Serum proteomics profiling- a young technology begins to mature. *Nature Biotech.* **23**, 291-292 (2005).
- 15 Fung, E. T., Thulassiraman, V., Weinberger, S. R. & Dalmasso, E. A. Protein biochips for differential profiling. *Curr. Op. Biotech.* **12**, 65-69 (2001).
- 16 Mor, G. *et al.* Serum protein markers for early detection of ovarian cancer. *Proc. Nat. Acad. Sci.* **102**, 7677-7682 (2005).
- 17 Miller, J. C. *et al.* Antibody microarray profiling of human prostate cancer sera: Antibody screening and identification of potential biomarkers. *Proteomics* **3**, 56-63 (2003).
- 18 Moorthy, J. & Beebe, D. J. *In situ* fabricated porous filters for microsystems. *Lab Chip* **3**, 62-66 (2003).
- 19 Chen, X., Cui, D. F., Liu, C. C. & Li, H. Microfluidic chip for blood cell separation and collection based on crossflow filtration. *Sens. Actuators B* **130**, 216-221 (2008).
- 20 Van Delinder, V. & Groisman, A. Separation of Plasma from Whole Human Blood in a Continuous Cross-Flow in a Molded Microfluidic Device. *Anal. Chem.* **78**, 3765-3771 (2006).
- 21 Crowley, T. A. & Pizziconi, V. Isolation of plasma from whole blood using planar microfilters for lab-on-a-chip applications. *Lab Chip* **5**, 922-929 (2005).
- 22 Davis, J. *et al.* Deterministic hydrodynamics: Taking blood apart. *Proceedings of the National Academy of Sciences* **103**, 14779 (2006).
- 23 Steigert, J. *et al.* Integrated siphon-based metering and sedimentation of whole blood on a hydrophilic lab-on-a-disk. *Biomed. Microdevices* **9**, 675-679 (2007).
- 24 Haberle, S., Brenner, T., Zengerle, R. & Ducree, J. Centrifugal extraction of plasma from whole blood on a rotating disk. *Lab Chip* **6**, 776-781 (2006).
- 25 Yang, S., Undar, A. & Zahn, J. D. A microfluidic device for continuous, real time blood plasma separation. *Lab on a Chip* **6**, 871-880 (2006).
- 26 Yang, S., Ji, B., Undar, A. & Zahn, J. D. Microfluidic Devices for Continuous Blood Plasma Separation and Analysis During Pediatric Cardiopulmonary Bypass Procedures. *ASAIO Journal* **52**, 698-704 (2006).
- 27 Fan, R. *et al.* Integrated Blood Barcode Chips. *Nature Biotech.*
- 28 Bailey, R. C., Kwong, G. A., Radu, C. G., Witte, O. N. & Heath, J. R. DNA-Encoded Antibody Libraries: A Unified Platform for Multiplexed Cell Sorting and Detection of Genes and Proteins. *J. Am. Chem. Soc.* **129**, 1959-1967 (2007).

2 Integrated Barcode Chips for Rapid, Multiplexed Analysis of Proteins in Microliter Quantities of Blood

2.1 Introduction

Microfluidics has permitted the miniaturization of conventional techniques to enable high-throughput and low-cost measurements in basic research and clinical applications.^{1,2} Systems for biomolecular assays^{3,4} and bio-separations,^{5,6} including the separation of circulating tumor cells or plasma from whole blood,⁷⁻⁹ have been reported. We developed the integrated blood barcode chip (IBBC) to address the need for microchips that integrate on-chip plasma separations from microliter quantities of whole blood with rapid in situ measurements of multiple plasma proteins. The immunoassay region of the chip is a microscopic barcode, integrated into a microfluidics channel and customized for the detection of many proteins and/or for the quantification of a single or few proteins over a broad concentration range. We demonstrate versatility of this barcode immunoassay by detecting human chorionic gonadotropin (hCG) from human serum over a 10^5 concentration range and by stratifying 22 cancer patients via multiple measurements of a dozen blood protein biomarkers for each patient. We also use the IBBC to assay a blood protein biomarker panel from whole human blood, performing all key steps in the immunoassay within 10 minutes of blood collection by finger prick.

We first present an overview of the IBBC and then discuss control of assay sensitivity, extension of a single protein assay to an assay for a large panel of biomarkers and, finally,

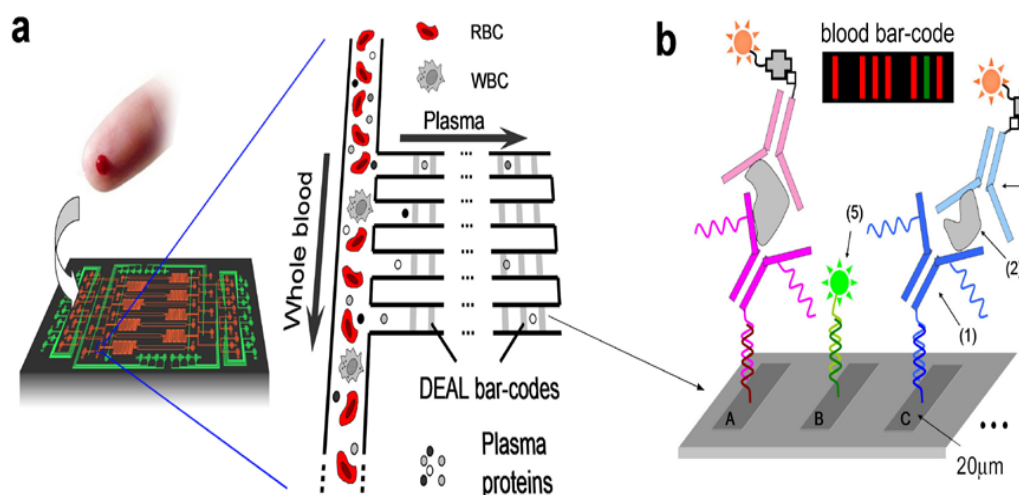


Figure 2.1 Design of an integrated blood barcode chip (IBBC). (a) Scheme depicting plasma separation from a finger prick of blood by harnessing the Zweifach-Fung effect. Multiple DNA-encoded antibody barcode arrays are patterned within the plasma-skimming channels for in situ protein measurements. (b) DEAL barcode arrays patterned in plasma channels for in situ protein measurement. A, B, C indicate different DNA codes. (1)–(5) denote DNA-antibody conjugate, plasma protein, biotin labeled detection antibody, streptavidin-Cy5 fluorescence probe and complementary DNA-Cy3 reference probe, respectively. The inset represents a barcode of protein biomarkers, which is read out using fluorescence detection. The green bar represents an alignment marker.

integration of plasma separation from whole blood, followed by the rapid measurement of a panel of protein biomarkers. **Figure 2.1** shows the design of an IBBC for blood separation and *in situ* protein measurement. We designed a polydimethylsiloxane (PDMS)-on-glass chip to perform 8–12 separate multiprotein assays sequentially or in parallel, starting from whole blood.

The Zweifach-Fung effect describes highly polarized blood cell flow at branch points of small blood vessels.^{9–11} A component of the IBBC, redesigned from a previous report,⁹ exploits this hydrodynamic effect by flowing blood through a low-flow-resistance primary channel with high-resistance, centimeter-long channels that branch off it at right angles (**Figure 2.1a**). As the resistance ratio is increased between the branches and the primary channel, a critical streamline moves closer to the primary channel wall adjoining the branch channels. Blood cells with a

radius larger than the distance between this critical streamline and the primary channel wall are directed away from the high-resistance channels, and ~15% of the plasma is skimmed into the high-resistance channels. The remaining whole blood is directed toward a waste outlet. The glass base of the plasma skimming channels is patterned with a dense barcode-like array of single-stranded DNA (ssDNA) oligomers before assembly of the microfluidics chip. A full barcode is repeated multiple times within a single plasma-skimming channel, and each barcode sequence constitutes a complete assay.

2.2 Experimental Methods

2.2.1 Micropatterning of Barcode Array

A PDMS mold containing 13–20 parallel microfluidic channels, with each channel conveying a different DNA oligomer as DEAL code, was fabricated by soft lithography. The PDMS mold was bonded to a polylysine-coated glass slide via thermal treatment at 80°C for 2 h. The polyamine surfaces permit significantly higher DNA loading than do more traditional aminated surfaces. DNA ‘bars’ of 2 μm in width have been successfully patterned using this technique. In the present study, a 20- μm channel width was chosen because the fluorescence microarray scanner we used has a resolution of 5 μm . Nevertheless, the current design already resulted in a DNA barcode array an order of magnitude denser than conventional microarrays fabricated by pin-spotting. The coding DNA solutions (A-M for the cancer serum test and AA-HH for the finger-prick blood test) prepared in 1X PBS were flowed into individual channels, and then allowed to evaporate completely. Finally, the PDMS was peeled off and the substrate with DNA barcode arrays was baked at 80°C for 2–4 h. The DNA solution concentration was ~100

μM in all experiments except in the hCG test, leading to a high loading of $\sim 6 \times 10^{13}$ molecules/ cm^2 (assuming 50% was collected onto substrate).

2.2.2 Fabrication of IBBCs

The fabrication of PDMS devices for the IBBCs was accomplished through a two-layer soft lithography approach. The control layer was molded from a SU8 2010 negative photoresist ($\sim 20 \mu\text{m}$ in thickness) silicon master using a mixture of GE RTV 615 PDMS prepolymer part A and part B (5:1). The flow layer was fabricated by spin-casting the pre-polymer of GE RTV 615 PDMS part A and part B (20:1) onto a SPR 220 positive photoresist master at $\sim 2,000$ r.p.m. for 1 minute. The SPR 220 mold was $\sim 17 \mu\text{m}$ in height after rounding by thermal treatment. The control layer PDMS chip was then carefully aligned and placed onto the flow layer, which was still situated on its silicon master, and an additional 60 minutes thermal treatment at 80°C was performed to enable bonding. Afterward, this two-layer PDMS chip was cut off the flow layer master and access holes were drilled. Finally, the two-layer PDMS chip was thermally bonded onto the barcode-patterned glass slide, yielding a completed integrated blood barcode chip (IBBC). In this chip, the DEAL barcode stripes are oriented perpendicular to the microfluidic assay channels. Typically, 8–12 identical units were integrated in a single chip with the dimensions of $2.5 \text{ cm} \times 7 \text{ cm}$.

2.2.3 Clinical Specimens of Cancer Patient Sera

The stored serum samples from 11 breast cancer patients (all female) and 11 prostate cancer patients (all male) were acquired from Asterand. Nineteen out of 22 patients were

European-American and the remaining three were Asian, Hispanic and African-American. The medical history is summarized in **Table 2.3**.

2.2.4 Collecting a Finger Prick of Blood

The human whole blood was collected according to the protocol approved by the institutional review board of the California Institute of Technology. Finger pricks were performed using BD microtainer contact-activated lancets. Blood was collected with SAFE-T-FILL capillary blood collection tubes (RAM Scientific), which we prefilled with 80 μL of 25 mM EDTA solution. A 10 μL volume of fresh human blood from a healthy volunteer was collected in an EDTA-coated capillary, dispensed into the tube, and rapidly mixed by inverting a few times. The spiked blood sample was prepared in a similar way except that 40 μL of 25 mM EDTA solution and 40 μL of recombinant solution were mixed and pre-added in the collection tube. Then 2 μL of 0.5 M EDTA was added to bring the total EDTA concentration up to 25 mM.

2.2.5 Execution of Blood Separation and Plasma Protein Measurement using IBBCs

The IBBCs were first blocked with the buffer solution for 30–60 minutes. The buffer solution prepared was 1% wt/vol bovine serum albumin fraction V (Sigma) in 150 mM 1X PBS without calcium/magnesium salts (Irvine Scientific). The fluid loading was conducted using a Tygon plastic tubing that is interfaced to the IBBC inlet with a 23 gauge metal pin. The Fluidigm solenoid unit was exploited to control the pressure (on/off) for both control valves and flow channels. A pressure of 8–10 p.s.i. was applied to actuate the valves, whereas the loading of fluid into assay channels was carried out with a lower pressure (0.5–3 p.s.i.) depending on the channel

flow resistance and the desired flow rate. Then DNA-antibody conjugates (~50–100 nM) were flowed through the plasma assay channels for ~30–45 minutes. This step transformed the DNA arrays into capture-antibody arrays. Unbound conjugates were washed off by flowing buffer solution through the channels. At this step, the IBBC was ready for the blood test. Two blood samples prepared as mentioned above were flowed into the IBBCs within 1 minute of collection. The IBBC quickly separated plasma from whole blood, and the plasma proteins of interest were captured in the assay zone where DEAL barcode arrays were located. This whole process from finger-prick to plasma protein capture took <10 minutes. In the cancer-patient serum experiment, the as-received serum samples were flowed into IBBCs without any pre-treatment (that is, no purification or dilution). Afterwards, a mixture of biotin-labeled detection antibodies (~50–100 nM) for the entire protein panel and the fluorescence Cy5-streptavidin conjugates (~100 nM) were flowed sequentially into IBBCs to complete the DEAL immunoassay. The unbound fluorescence probes were rinsed off by flowing the buffer solution for 10 minutes. At last, the PDMS chip was removed from the glass slide. The slide was immediately rinsed in ½X PBS solution and deionized water and then dried with a nitrogen gun. Finally, the DEAL barcode slide was scanned by a microarray scanner.

2.2.6 Quantitation and Statistics

All the barcode array slides used in quantification were scanned using an Axon GenePix 4000B two-color laser microarray scanner at the same instrument settings. For 635 nm and 532 nm excitation lasers, respectively, the following settings were used: Laser Power - 100% and 33%; Optical Gain - 800 and 700; Brightness/Contrast - 87 and 88. The output JPEG images were carefully skewed and resized to fit the standard barcode array mask design. Then, an image

processing software, NIH *ImageJ*, was used to produce intensity line profiles of barcodes in all assay channels. Finally, all the line profile data files were loaded into a home-developed program embedded as an Excel macro to generate a spreadsheet that lists the average intensities of all 13 bars in each of 20 barcodes. The means and standard deviations were computed using Microcal *Origin*. Non-supervised clustering of patients was performed using literature methods and algorithms.¹² To assess the significance of two patient (sub)groups, Student's t analysis was performed on selected proteins and all *p*-values were calculated at a significance level of 0.05, if not otherwise specified.

2.3 Results and Discussion

We used the DNA-encoded antibody library (DEAL) technique¹³ (**Figure 2.5**) to detect proteins within the plasma-skimming channels. DEAL technology involves using DNA-directed immobilization of antibodies to convert a pre-patterned ssDNA barcode microarray into an antibody microarray, thereby providing a powerful means for spatial encoding.^{14,15} The sequences of all ssDNA oligomer pairs used (labeled A/A'-M/M'), and their corresponding antibodies, are listed in **Tables 2.1** and **2.2**. To minimize cross-reactivity, these ssDNA molecules were designed *in silico* and then validated through a full orthogonality test (**Figure 2.6**). In that experiment, each of the complementary DNA molecules with Cy3 fluorescent label was added to a microwell containing a full primary ssDNA barcode array. The results showed only negligible cross-hybridization signals. In the DEAL assay, each capture antibody is tagged with approximately three copies of an ssDNA oligomer that is complementary to ssDNA oligomers that have been surface-patterned into a microscopic barcode within the immunoassay region of the chip. Flow-through of the DNA-antibody conjugates transforms the DNA

microarray into an antibody microarray for the subsequent surface-bound immunoassay. Because DNA patterns are robust to dehydration and can survive elevated temperatures (80–100°C), the DEAL approach circumvents the denaturation of antibodies often associated with typical microfluidics fabrication.

As only a few microliters of blood is normally sampled from a finger prick, on-chip plasma separation yields only a few hundred nanoliters of plasma. The ssDNA barcodes were patterned at a high density using microchannel-guided flow patterning (**Figure 2.7**) to measure a large panel of protein biomarkers from this small volume. We used a PDMS mold that was thermally bonded onto a polyamine-coated glass slide to pattern the entire ssDNA barcode. Polyaminated surfaces permit substantially higher DNA loading than do more traditional aminated surfaces¹⁶ and provide for an accompanying increase in assay sensitivity (**Figures 2.8** and **2.9**). Different solutions, each containing a specific ssDNA oligomer, were flowed through different channels and evaporated through the gas-permeable PDMS stamp, resulting in individual stripes of DNA molecules. One complete set of stripes represents one barcode. All measurements used 20- μm -wide bars spaced at a 40- μm pitch. This array density represents an approximately tenfold increase over a standard spotted array (typical dimensions are 150- μm diameter spots at a 400- μm pitch), thus expanding the numbers of proteins that can be measured within a small volume. No alignment between the barcode array and the plasma channels (IBBC chip design presented in **Figure 2.11**) was required. All protein assays used one color fluorophore and were spatially identified using a reference marker that fluoresced at a different color.

We first illustrate aspects of the barcode assays via the measurement of a single biomarker, human chorionic gonadotropin (hCG), in undiluted human serum over a broad

concentration range. HCG is widely used for pregnancy testing and is a biomarker for gestational trophoblastic tumors and germ cell cancers of the ovaries and testes. For this assay, the barcode was customized by varying the DNA loading during the flow patterning step. The DNA barcode contained 13 regions (**Figure 2.2a**). There were two bars of oligomer B designed to detect the protein, tumor necrosis factor-alpha (TNF- α), as a negative control, one reference bar (oligomer M), one blank, and nine bars of oligomer A (designed for hCG detection and flow patterned at ssDNA concentrations that were varied from 200 μ M to 2 μ M). To perform the assay, we flowed a mixture of A'-anti-hCG and B'-anti-TNF- α through assay channels. Next, a series of standard hCG serum samples and two hCG samples of unknown concentration were flowed through separate assay channels. Biotinylated detection antibodies for hCG and TNF- α were then applied

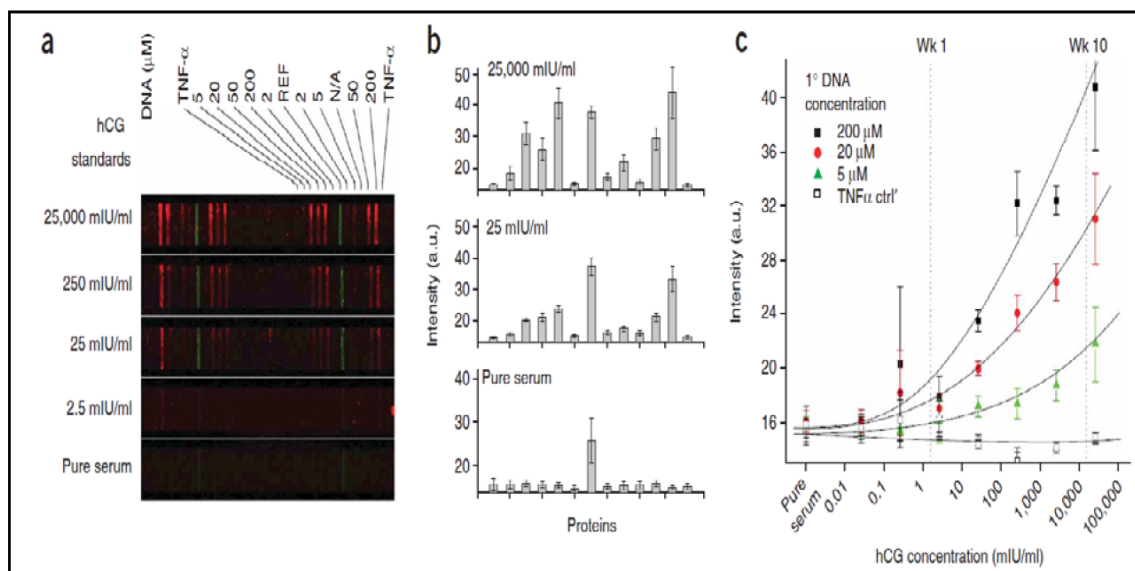


Figure 2.2 Measurement of human chorionic gonadotropin (hCG) in sera.

(a) Fluorescence images of DEAL barcodes showing the measurement of a series of standard serum samples spiked with hCG. The bars used to measure hCG were patterned with DNA strand A at different concentrations. TNF- α encoded by strand B was employed as a negative control. The green bars (strand M) serve as references. (b) Quantification of the full barcodes for three selected samples. (c) Mean values of fluorescence signals corresponding to three sets of bars with different DNA loadings. Broken lines indicate the typical physiological levels of hCG in sera after 1 or 10 weeks of pregnancy. Error bars, 1 s.d.

followed by a final developing step using fluorescent Cy5-labeled streptavidin (red) for all protein channels and Cy3-labeled M' oligomer (green) for the reference channels (**Figure 2.2a**). Quantifying the fluorescence intensity (**Figure 2.2b,c**) revealed a sensitivity (~ 1 mIU/ml) comparable to the enzyme-linked immunosorbent assay (ELISA) and a broad detected concentration range ($\sim 10^5$). Using the microfluidics-entrained DEAL barcode in a blind test, we measured the hCG levels in the two unknown serum samples. Our measured levels, estimated at 6 and 400 mIU/ml for unknowns 1 and 2, are in good agreement with the values of 12 and 357 mIU/ml, respectively, obtained from an independent lab test (**Figure 2.12**). Even without quantification, the analyte concentrations can be estimated by eye through pattern recognition of the full barcode. The bar with the highest DNA-loading rendered the highest sensitivity, whereas the bar with the lowest DNA-loading was used to discriminate samples with high analyte concentrations. For example, the 25,000 mIU/ml and 250 mIU/ml hCG samples can be visually distinguished using stripes patterned with lower DNA concentrations, whereas the stripes loaded from 200 μ M DNA solutions do not readily distinguish these samples. For circumstances in which accurate photon counting is not available, visual barcode inspection permits a rough estimation of the target quantity—a potential point-of-care application. When levels of hCG are tracked during pregnancy, concentrations in the blood increase from ~ 5 mIU/ml in the first week of pregnancy to $\sim 2 \times 10^5$ mIU/ml 10 weeks after conception. The IBBC can cover such a broad physiological hCG range with reasonable accuracy.

To evaluate multiplexed measurements of a panel of 12 protein markers using the microfluidic DEAL barcode regions of the IBBCs, we quantified the cross-reactivity between the stripes within the DNA-encoded immunoassays. This test involved twelve human serum proteins, including: ten cytokines - interferon (IFN)- γ , TNF- α , interleukin (IL)-2, IL-1 α , IL-1 β ,

transforming growth factor (TGF)- β 1, IL-6, IL-10, IL-12, granulocyte-macrophage colony-stimulating factor (GM-CSF); a chemokine - macrophage chemoattractant protein (MCP)-1; and the cancer biomarker, prostate-specific antigen (PSA). The results showed negligible cross-talk, with typical photon counts <2% compared to the correctly paired antigen-antibody complexes (**Figure 2.13**). We also assayed serial dilutions (from 5 nM to 1 pM) for these proteins on the DEAL barcode chip to establish a set of calibration curves for future estimates of protein concentration in sera (**Figure 2.14**). We fixed all the parameters associated with laser scanning and fluorescence quantification (e.g., power, gain, brightness and contrast) and performed quantitative analysis. Depending on the antibodies used, the estimated sensitivity varied from <1 pM for IL-1 β and IL-12 to ~30 pM for TGF- β , and was comparable to the detection limits of ELISA based on the same antibody pairs. For example, according to the specifications of commercial kits (eBioscience), the detection limit for cytokines like TNF- α and IL-1 β is ~8 pg/ml (~0.5 pM), which compares favorably with our observations. However, the statistical variation of the measured signals is relatively large compared to a commercial ELISA assay—a variation that is likely due to the fact that our chips are manufactured manually.

We assessed the utility of the DEAL barcodes for clinical blood samples by measuring the same 12 proteins from small amounts of stored serum collected from 22 cancer patients. These serum samples were thawed, and then assayed using two chips, each containing 12 separate assay units operated in parallel. In every unit, 20 full DEAL barcodes in each assay channel were used for statistical sampling. The proteins in this panel (**Figure 2.3a**) - the prostate cancer marker, PSA, and eleven proteins secreted by white blood cells - have been associated with tumor microenvironment formation, tumor progression, and tumor metastasis.¹⁷⁻¹⁹ Thus, this panel provides information relevant to multiple aspects of cancer.

Figure 2.3b shows fluorescence images, each depicting four sets of randomly picked barcodes obtained from the 22 patient samples. The medical records for all patients are summarized in **Table 2.3**. B01–B11 denote 11 samples from breast cancer patients, whereas P01–P11 are from prostate cancer patients. Many proteins were successfully detected with high signal-to-noise ratios, and the barcode signatures are distinctive from patient to patient, excepting the assays on P05, P04, P10 and B10. These assays are from individuals who are heavy smokers (~11–20 cigarettes daily). Only one serum sample (P06) from a heavy smoker did not exhibit a high background. This high background may result from elevated blood content of the fluorescent protein carboxyhemoglobin, which has been shown to be relevant to the pathogenesis of lung diseases of smokers.²⁰ Although we have also measured high background in a number of stored serum samples, we have never measured a high background in assays from freshly collected blood, as described below. The results imply that, at least for stored samples, some pre-purification of the plasma or serum will be required to assay serum protein levels.

Barcode intensities were then quantified and the statistic mean value for each protein was computed (**Figures 2.15, 2.16, and 2.17**). The cancer marker PSA clearly distinguished between the breast cancer and the prostate cancer patients. The only exception was a false-positive result from B10 that had high nonspecific background. We independently validated our PSA measurements for all patient sera using standard ELISA. For eight of the prostate cancer patients, we compared our results with clinical ELISA measurements provided by the serum supplier. The results (**Figure 2.3c**) validated the applicability of the DEAL barcodes for assaying complex clinical samples. However, the statistical accuracy of the PSA barcode assay was not high, revealing only a modest linear correlation between ELISA and DEAL (**Figure 2.18**). Again, this is likely due to our manual chip manufacturing process. We are currently automating our barcode

fabrication, assay execution, and image quantification in an effort to bring statistical uncertainties to within 10–20%, close to the state of the art.

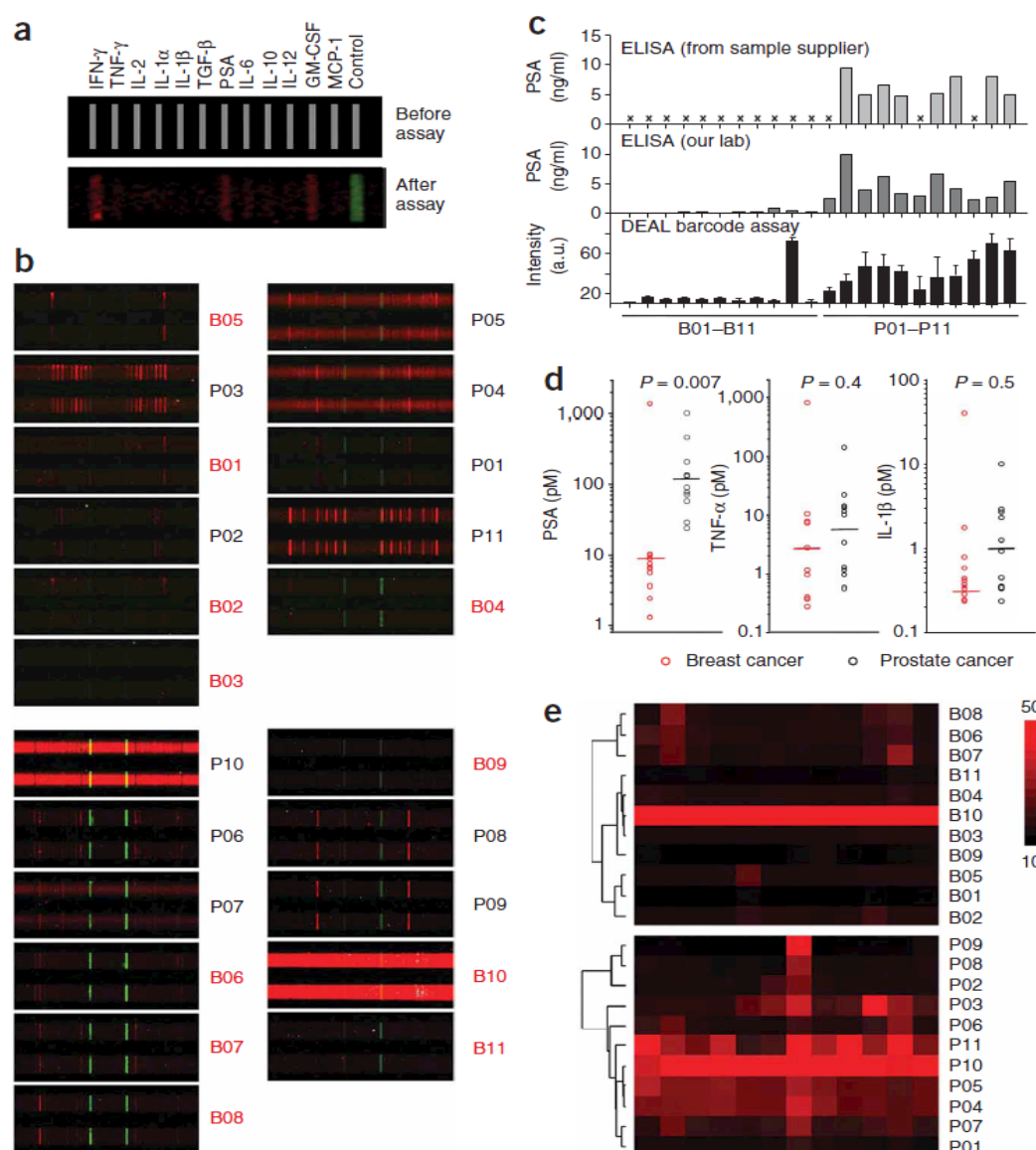


Figure 2.3 Multiplexed protein measurements of clinical patient sera. (a) Layout of the barcode array used in this study. Green denotes the reference (strand M). (b) Representative fluorescence images of barcodes used to measure the cancer marker PSA and 11 cytokines from 22 cancer patient serum samples. B01–B11, samples from breast cancer patients; P01–P11, samples from prostate cancer patients. The left and right columns represent measurements on different chips. (c) Validation of PSA DEAL barcode measurement using ELISA. x denotes PSA measurements were not provided by the serum supplier. Error bars, 1 s.d. (d) Distribution of estimated concentrations of PSA, TNF- α , and IL-1 β in all serum samples. The horizontal bars mark the mean values. (e) Complete non-supervised clustering of breast and prostate cancer patients on the basis of protein patterns.

The cancer patient barcode data could be analyzed for absolute protein levels by comparing those data against the barcode quantification plots (**Figure 2.14**). Results for PSA, TNF- α and IL-1 β are shown in **Figure 2.3d**. PSA concentrations range from 22 pM to 1 nM (or 0.7 to 33 ng/ml) with a log-scale mean of 117 pM (3.8 ng/ml) for prostate cancer patients. The estimated PSA concentrations for breast cancer patient sera have a mean of 9.1 pM. PSA readily differentiates between these two patient groups with good statistical accuracy ($P = 0.0007$). Nevertheless, the absolute PSA levels measured by either the standard ELISA or by the barcode assay are below those determined by the clinical ELISA—a likely result of sample degradation during storage (**Figure 2.3c**). As would be expected, neither TNF- α nor IL-1 β allows prostate and breast cancer patients to be distinguished ($P = 0.4$ and 0.5 , respectively, at a significance level of 0.2). Our estimates of absolute protein levels indicate that the protein concentration ranges assessed by the DEAL barcode assay are clinically relevant for patient diagnostics. For example, the serum level of cytokines such as interleukins and tumor necrosis factors can reach ~ 10 – 100 pg/ml in cancer patients,²¹ ~ 500 pg/ml in rheumatoid arthritis patients, and 41 ng/ml²² in septic shock.²³ These levels can all be captured using the barcode assay format.

We performed a complete non-supervised clustering (that is, using only the levels of assayed proteins without assigning any weight factors) of patients and generated a heat map (**Figure 2.3e**) to assess the potential of this technology for patient stratification. This analysis is only presented as a proof of principle. Nevertheless, the results are encouraging. For example, the measured profiles of breast cancer patients can be classified into three subsets—non-inflammatory, IL-1 β positive and TNF- α /GMCSF positive ($P_{\text{TNF}\alpha} = 0.005$, $P_{\text{GMCSF}} = 0.04$ for the latter two subsets). The prostate cancer patient data were classified into two major subsets based upon the inflammatory protein levels ($P_{\text{TNF}\alpha} = 0.016$, $P_{\text{GMCSF}} = 0.012$). The multiplexed

measurement of cytokines²⁴ is relevant to cancer diagnostics and prognostics.^{25,26} Our results demonstrate that IBBCs can be applied to the multi-parameter analysis of human health-relevant proteins in serum.

The ultimate goal behind developing the IBBC was to measure the levels of a large number of proteins in human blood within a few minutes of sampling that blood, to avoid the protein degradation that can occur when plasma is stored. In a typical 96-well plate immunoassay, the biological sample of interest is added, and the protein diffuses to the surface-bound antibody. Under adequate flow conditions, diffusion is no longer important, and the only parameter that limits the speed of the assay is the protein/antibody binding kinetics (the Langmuir isotherm),²⁷ thus allowing the immunoassay to be completed in just a few minutes.²⁸ Flow through our plasma skimming channels proceeds at velocities $> \sim 0.1 \text{ mm sec}^{-1}$ and can operate continuously and with near 100% efficiency unless the blood flow is clogged.

For whole blood analysis, the microfluidic channels of IBBCs were precoated with bovine serum albumin blocking buffer. The DNA barcodes were transformed into antibody barcodes as described above, and blood samples were flowed into the device within 1 minute of fingerprick collection. The time from that finger prick to completion of blood flow through the device was ~ 9 minutes. We sampled both as-collected whole blood and protein-spiked blood from healthy volunteers. **Figure 2.4a** shows the effective separation of plasma in an IBBC. The few red blood cells that did enter the plasma channels (**Figure 2.4a**, right panel) did not affect the subsequent protein assay.

The plasma proteins detected in this whole-blood analysis experiment included a cancer marker (PSA), four cytokines, and three other functional proteins - complement C3, C-reactive protein (CRP), and plasminogen - involved in the complement system, inflammatory response,

fibrin degradation and liver toxicity (**Tables 2.1** and **2.2**). After exposure of the barcode assay region to the separated, flowing plasma for 8 minutes, the detection antibody solution and the fluorescence probes were added to complete the assay. All proteins in the spiked blood were detected (**Figure 2.4b,c**). Cytokines gave the strongest fluorescence signals due to the higher affinities of their cognate antibodies. The measurement of the unspiked fresh blood established a baseline for a healthy volunteer, in which IL-6, IL-10, C3, and plasminogen were detected. Using IBBCs for the separation and analysis of freshly collected blood consistently resulted in very clean DEAL barcodes, with little or no evidence of biofouling. We are planning a study to assess the importance of rapid measurements for obtaining accurate protein levels.

Our IBBC enables the rapid measurement of a panel of plasma proteins from a finger prick of whole blood. Integration of microfluidics and DNA-encoded antibody arrays enables reliable processing of blood and in situ measurement of plasma proteins within a time scale that is short enough to avoid most protein degradation processes that can occur in sampled blood. Use of the IBBC represents a minimally invasive, low-cost, and robust procedure, and potentially represents a realistic clinical diagnostic platform.

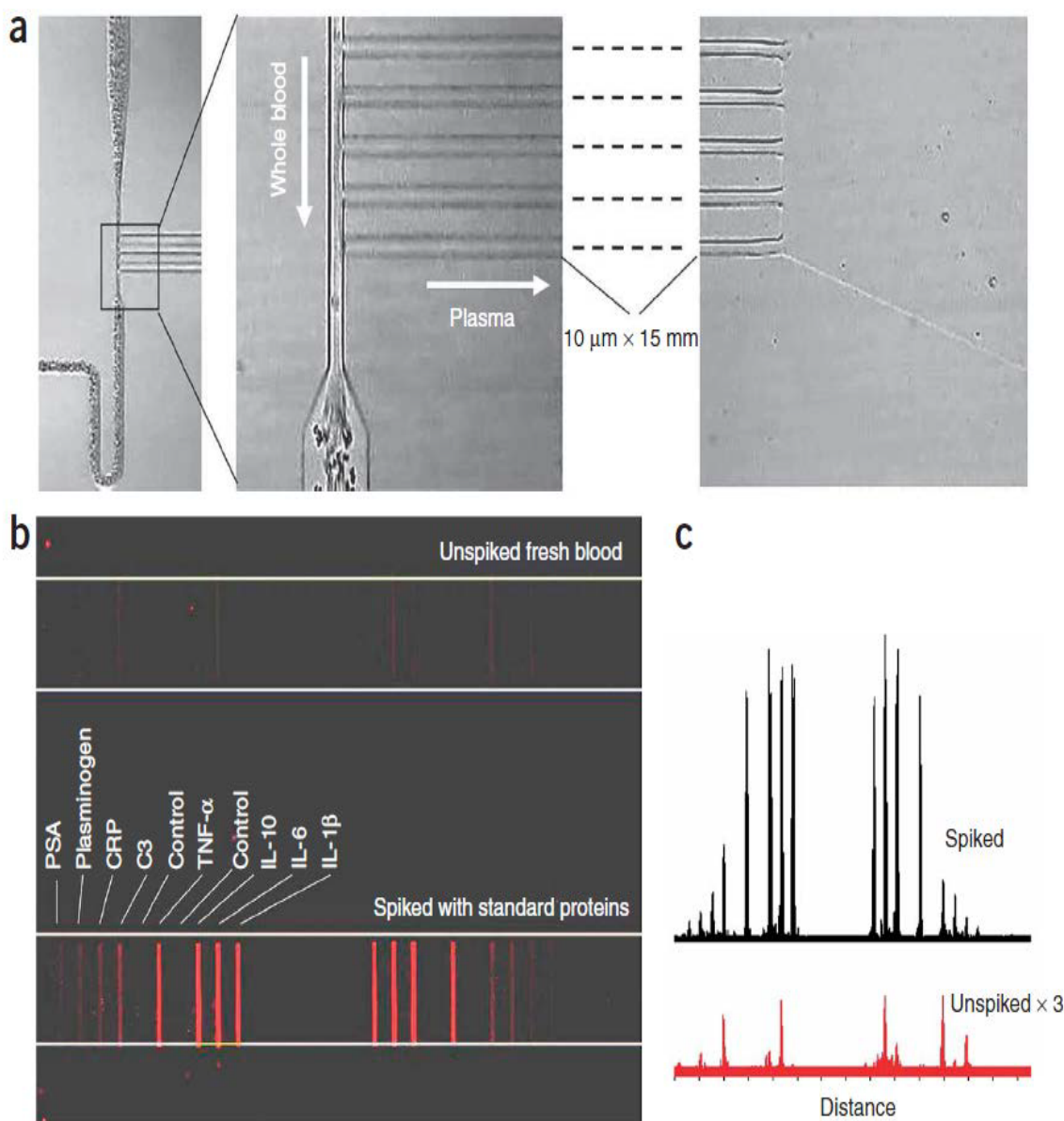


Figure 2.4 IBBC for the rapid measurement of a panel of serum biomarkers from a finger prick of whole blood. (a) Optical micrographs showing the effective separation of plasma from fresh whole blood. A few red blood cells occasionally seen downstream of the plasma channels did not affect the protein assay. **(b)** Fluorescence image of blood barcodes in two adjacent microchannels of an IBBC, on which both the unspiked and spiked fresh whole blood collected from a healthy volunteer were separately assayed. Eight plasma proteins are indicated. All bars, $20\ \mu\text{m}$ wide. **(c)** Fluorescence line profiles of the barcodes for both unspiked and spiked whole blood samples. The distance corresponds to the full length shown in **b**.

2.4 References

- 1 Sia, S. K. & Whitesides, G. M. Microfluidic devices fabricated in poly(dimethylsiloxane) for biological studies. *Electrophoresis* **24**, 3563-3576 (2003).
- 2 Quake, S. R. & Scherer, A. From micro- to nanofabrication with soft materials. *Science* **290**, 1536-1540 (2000).
- 3 Huang, B. *et al.* Counting low-copy number proteins in a single cell. *Science* **315**, 81-84 (2007).
- 4 Ottesen, E. A., Hong, J. W., Quake, S. R. & Leadbetter, J. R. Microfluidic digital PCR enables multigene analysis of individual environmental bacteria. *Science* **314**, 1464-1467 (2006).
- 5 Huang, L., Cox, E., Austin, R. & Sturm, J. Continuous particle separation through deterministic lateral displacement. *Science* **304**, 987 (2004).
- 6 Chou, C. F. *et al.* Sorting biomolecules with microdevices. *Electrophoresis* **21**, 81-90 (2000).
- 7 Toner, M. & Irimia, D. Blood-on-a-chip. *Annual Review of Biomedical Engineering* **7**, 77-103 (2005).
- 8 Nagrath, S. *et al.* Isolation of rare circulating tumour cells in cancer patients by microchip technology. *Nature* **450**, 1235-1239 (2007).
- 9 Yang, S., Ündar, A. & Zahn, J. A microfluidic device for continuous, real time blood plasma separation. *Lab on a Chip* **6**, 871-880 (2006).
- 10 Svanes, K. & Zweifach, B. W. Variations in small blood vessel hematocrits produced in hypothermic rates by micro-occlusion. *Microvascular Research* **1**, 210-220 (1968).
- 11 Fung, Y. C. Stochastic flow in capillary blood vessels. *Microvascular Research* **5**, 34-38 (1973).
- 12 Eisen, M., Spellman, P., Brown, P. & Botstein, D. Cluster analysis and display of genome-wide expression patterns. *Proceedings of the National Academy of Sciences* **95**, 14863 (1998).
- 13 Bailey, R., Kwong, G., Radu, C., Witte, O. & Heath, J. DNA-encoded antibody libraries: a unified platform for multiplexed cell sorting and detection of genes and proteins. *J. Am. Chem. Soc* **129**, 1959-1967 (2007).
- 14 Boozer, C., Ladd, J., Chen, S. F. & Jiang, S. T. DNA-directed protein immobilization for simultaneous detection of multiple analytes by surface plasmon resonance biosensor. *Analytical Chemistry* **78**, 1515-1519 (2006).
- 15 Niemeyer, C. M. Functional devices from DNA and proteins. *Nano Today* **2**, 42-52 (2007).
- 16 Pirrung, M. C. How to make a DNA chip. *Angewandte Chemie-International Edition* **41**, 1277 (2002).
- 17 Coussens, L. & Werb, Z. Inflammation and cancer. *Nature* **420**, 860-867 (2002).
- 18 Lin, W. W. & Karin, M. A cytokine-mediated link between innate immunity, inflammation, and cancer. *Journal of Clinical Investigation* **117**, 1175-1183 (2007).
- 19 De Marzo, A. M. *et al.* Inflammation in prostate carcinogenesis. *Nature Reviews Cancer* **7**, 256-269 (2007).
- 20 Ashton, H. & Telford, R. Smoking and carboxhemoglobin. *Lancet* **2**, 857-858 (1973).
- 21 Chopra, V., Dinh, T. & Hannigan, E. Serum levels of interleukins, growth factors and angiogenin in patients with endometrial cancer. *Journal of Cancer Research and Clinical Oncology* **123**, 167-172 (1997).
- 22 Öncül, O., Top, C. & Çavu lu. Correlation of serum leptin levels with insulin sensitivity in patients with chronic hepatitis-C infection. *Diabetes Care* **25**, 937 (2002).
- 23 Pinsky, M. *et al.* Serum cytokine levels in human septic shock. Relation to multiple-system organ failure and mortality. *Chest* **103**, 565 (1993).
- 24 Schweitzer, B. *et al.* Multiplexed protein profiling on microarrays by rolling-circle amplification. *Nature Biotechnology* **20**, 359-365 (2002).

- 25 Lambeck, A. J. A. *et al.* Serum cytokine profiling as a diagnostic and prognostic tool in ovarian cancer: A potential role for interleukin 7. *Clinical Cancer Research* **13**, 2385-2391 (2007).
- 26 Gorelik, E. *et al.* Multiplexed immunobead-based cytokine profiling for early detection of ovarian cancer. *Cancer Epidemiology Biomarkers & Prevention* **14**, 981-987 (2005).
- 27 Heath, J. R. & Davis, M. E. Nanotechnology and cancer. *Annual Review of Medicine* **59**, 405 (2007).
- 28 Zimmermann, M., Delamarche, E., Wolf, M. & Hunziker, P. Modeling and optimization of high-sensitivity, low-volume microfluidic-based surface immunoassays. *Biomedical Microdevices* **7**, 99-110 (2005).

2.5 Appendix A: Supplementary Methods

2.5.1 DNA-Encoded Antibody Libraries (DEAL) Technique

The critical technique upon which this study is based is the DNA-encoded antibody library (DEAL) method.¹ When DEAL is utilized to measure proteins, it is used as follows (**Figure 2.5**). Capture antibodies (CAs) against the protein of interest are chemically labeled with single-stranded DNA (ssDNA) oligomers, yielding ssDNA-CA conjugates. The coupling reaction is accomplished using succinimidyl 4-formylbenzoate (SFB, Solulink) and succinimidyl 4-hydrazinonicotinate acetone hydrazone in *N,N*-dimethylformamide (DMF) (SANH, Solulink) as conjugation agents to link amine termini on DNA oligomers to the amine side-groups of proteins.¹ A size-exclusion column is used to purify the product by removing excess unreacted DNA molecules. Separately, the complementary ssDNA oligomers are deposited in a barcode pattern on a poly-L-lysine coated glass slide using microchannel-guided patterning (details described in **Figure 2.7**). At the beginning of a DEAL protein assay, incubation of ssDNA-CA conjugates with the complementary spatially-patterned ssDNA array assembles the CAs onto those specific sites through DNA hybridization. This step transforms the DNA microarray into an antibody microarray that is ready for a protein sandwich assay. Biological samples (i.e. plasma isolated from human whole blood) can be applied onto the CA microarray and antigens can be captured. Finally, detection antibodies and/or fluorescent read-out probes are introduced sequentially to complete the immuno-sandwich assay. DNA oligomer sequences are chosen with appropriate melting temperatures to optimize room-temperature hybridization to complementary strands while minimizing cross-hybridization (<5% in fluorescence signal).

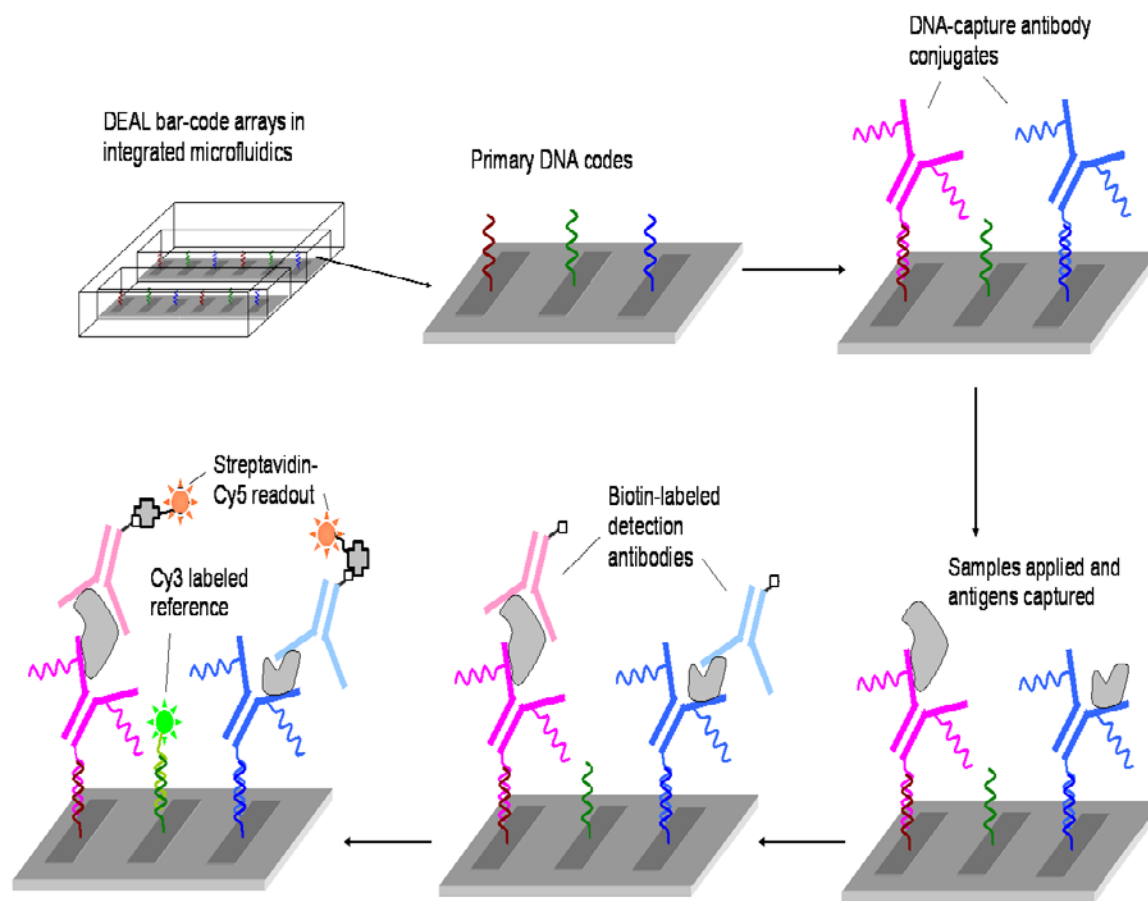


Figure 2.5 Schematic depiction of multi-parameter detection of proteins in integrated microfluidics using the DNA-Encoded Antibody Library (DEAL) technique.

2.5.2 Serum Protein Biomarker Panels and Oligonucleotide Labels

The protein panels used in the cancer-patient serum experiment (panel 1) and finger-prick blood test (panel 2), the corresponding DNA codes, and their sequences are summarized in **Tables 2.1** and **2.2**. These DNA oligomers were synthesized by Integrated DNA Technologies (IDT), and purified by high pressure liquid chromatography (HPLC). The quality was confirmed by mass spectrometry.

Table 2.1 List of Proteins and Corresponding DNA Codes

DNA-code	Human Plasma Protein	Abbreviation
<u>Panel (1)</u>		
A/A'	Interferon-gamma	IFN- γ
B/B'	Tumor necrosis factor-alpha	TNF- α
C/C'	Interleukin-2	IL-2
D/D'	Interleukin-1 alpha	IL-1 α
E/E'	Interleukin-1 beta	IL-1 β
F/F'	Transforming growth factor beta	TGF- β
G/G'	Prostate specific antigen (total)	PSA
H/H'	Interleukin-6	IL-6
I/I'	Interleukin-10	IL-10
J/J'	Interleukin-12	IL-12
K/K'	Granulocyte-macrophage colony stimulating factor	GM-CSF
L/L'	Monocyte chemoattractant protein -1	MCP-1
M/M'	Blank control/reference	
<u>Panel (2)</u>		
AA/AA'	Interleukin-1 beta	IL-1 β
BB/BB'	Interleukin-6	IL-6
CC/CC'	Interleukin-10	IL-10
DD/DD'	Tumor necrosis factor-alpha	TNF- α
EE/EE'	Complement Component 3	C3
FF/FF'	C-reactive protein	CRP
GG/GG'	Plasminogen	Plasminogen
HH/HH'	Prostate specific antigen (total)	PSA

Table 2.2 List of DNA Sequences used for Spatial Encoding of Antibodies

Sequence Name	Sequence	T _m °C (50mM NaCl)
A	5'-AAAAAAAAAAAAATCCTGGAGCTAAGTCCGTA-3'	57.9
A'	5' NH3-AAAAAAAAAAAAATACGGACTTAGCTCCAGGAT-3'	57.2
B	5'-AAAAAAAAAAAAAGCCTCATTGAATCATGCCTA-3'	57.4
B'	5' NH3-AAAAAAAAAAAAATAGGCATGATTCAATGAGGC-3'	55.9
C	5'-AAAAAAAAAAAAAGCACTCGTCTACTATCGCTA-3'	57.6
C'	5' NH3-AAAAAAAAAAAAATAGCGATAGTAGACGAGTGC-3'	56.2
D	5'-AAAAAAAAAAAAATGGTCGAGATGTCAGAGTA-3'	56.5
D'	5' NH3-AAAAAAAAAAAAATACTCTGACATCTCGACCAT-3'	55.7
E	5'-AAAAAAAAAAAAATGTGAAGTGGCAGTATCTA-3'	55.7
E'	5' NH3-AAAAAAAAAAAAATAGATACTGCCACTTCACAT-3'	54.7
F	5'-AAAAAAAAAAAAATCAGGTAAGGTTACGGTA-3'	56.9
F'	5' NH3-AAAAAAAAAAAAATACCGTGAACCTTACCTGAT-3'	56.1
G	5'-AAAAAAAAAAGAGTAGCCTTCCCGAGCATT-3'	59.3
G'	5' NH3-AAAAAAAAAAAAATGCTCGGGAAGGCTACTC-3'	58.6
H	5'-AAAAAAAAAATTGACCAAAGTGGTGCG-3'	59.9
H'	5' NH3-AAAAAAAAAACGCACCGCAGTTTGGTCAAT-3'	60.8
I	5'-AAAAAAAAAATGCCCTATTGTTGCGTCGGA-3'	60.1
I'	5' NH3-AAAAAAAAAATCCGACGCAACAATAGGGCA-3'	60.1
J	5'-AAAAAAAAAATCTTCTAGTTGTCGAGCAGG-3'	56.5
J'	5' NH3-AAAAAAAAAACCTGCTCGACAACTAGAAGA-3'	57.5
K	5'-AAAAAAAAAATAATCTAATTCTGGTCGCGG-3'	55.4
K'	5' NH3-AAAAAAAAAACCGCGACCAGAATTAGATTA-3'	56.3
L	5'-AAAAAAAAAAGTGATTAAGTCTGCTTCGGC-3'	57.2
L'	5' NH3-AAAAAAAAAAGCCGAAGCAGACTTAATCAC-3'	57.2
M	5'-AAAAAAAAAAGTCGAGGATTCTGAACCTGT-3'	57.6
M'	5' NH3-AAAAAAAAAACAGGTTCAGAATCCTCGAC-3'	56.9
AA'	5' NH3-AAAAAAAAAAGTCACAGACTAGCCACGAAG-3'	58
BB	5'-AAAAAAAAAAGCGTGTGTGGACTCTCTA-3'	58.7
BB'	5' NH3-AAAAAAAAAATAGAGAGAGTCCACACACGC-3'	57.9
CC	5'-AAAAAAAAAATCTTCTAGTTGTCGAGCAGG-3'	56.5
CC'	5' NH3-AAAAAAAAAACCTGCTCGACAACTAGAAGA-3'	57.5
DD	5'-AAAAAAAAAAGATCGTATGGTCCGCTCTCA-3'	58.8

DD'	5' NH3-AAAAAAAAAATGAGAGCGGACCATACGATC-3'	58
EE	5'-AAAAAAAAAAGCACTAACTGGTCTGGGTCA-3'	59.2
EE'	5' NH3-AAAAAAAAAATGACCCAGACCAGTTAGTGC-3'	58.4
FF	5'-AAAAAAAAAATGCCCTATTGTTGCGTCGGA-3'	60.1
FF'	5' NH3-AAAAAAAAAATCCGACGCAACAATAGGGCA-3'	60.1
GG	5'-AAAAAAAAAACTCTGTGAACTGTCATCGGT-3'	57.8
GG'	5' NH3-AAAAAAAAAAACCGATGACAGTTCACAGAG-3'	57
HH	5'-AAAAAAAAAAGAGTAGCCTTCCCGAGCATT-3'	59.3
HH'	5' NH3-AAAAAAAAAATGCTCGGGAAGGCTACTC-3'	58.6

* All amine-terminated strands were linked to antibodies to form DNA-antibody conjugates using SFB/SANH coupling chemistry described by R. Bailey *et al.*¹ Codes AA-HH were used in the experiment examining fresh whole blood from a healthy volunteer. Codes A-M were used for the molecular analyses of cancer patient serum samples.

All matched antibody pairs and standard proteins (recombinants) were received from eBioscience except those described below. The antibody pairs for human C3 and CRP were received from Abcam. Their recombinants were from Sigma. The antibody pair and recombinant protein for human plasminogen were received from Molecular Innovations. The antibody pair for PSA was received from Biodesign. The PSA recombinant was from R&D Systems. The capture and detection antibodies for human hCG were received from Abcam and Chromoprobe, respectively. The antibody pair and the recombinant for human GM-CSF were both received from BD biosciences. All oligonucleotides were synthesized by Integrated DNA Technologies.

2.5.3 Cross-Reactivities of Oligonucleotide Labels

A full orthogonality analysis was performed to quantitate the cross-hybridization between the stripes within the DEAL barcode arrays. A 13-well PDMS slab was placed onto a barcode array chip consisting of thirteen distinct strands of coding ssDNA (A-M). In each well, a solution containing only one kind of complementary ssDNA from A'-M' (labeled with Cy3) was added,

and successful hybridization was visualized by fluorescence using a 532 nm laser excitation. The result (Figure 2.6) indicates negligible cross-hybridization across the entire panel of DNA codes used in our DEAL barcode assay.

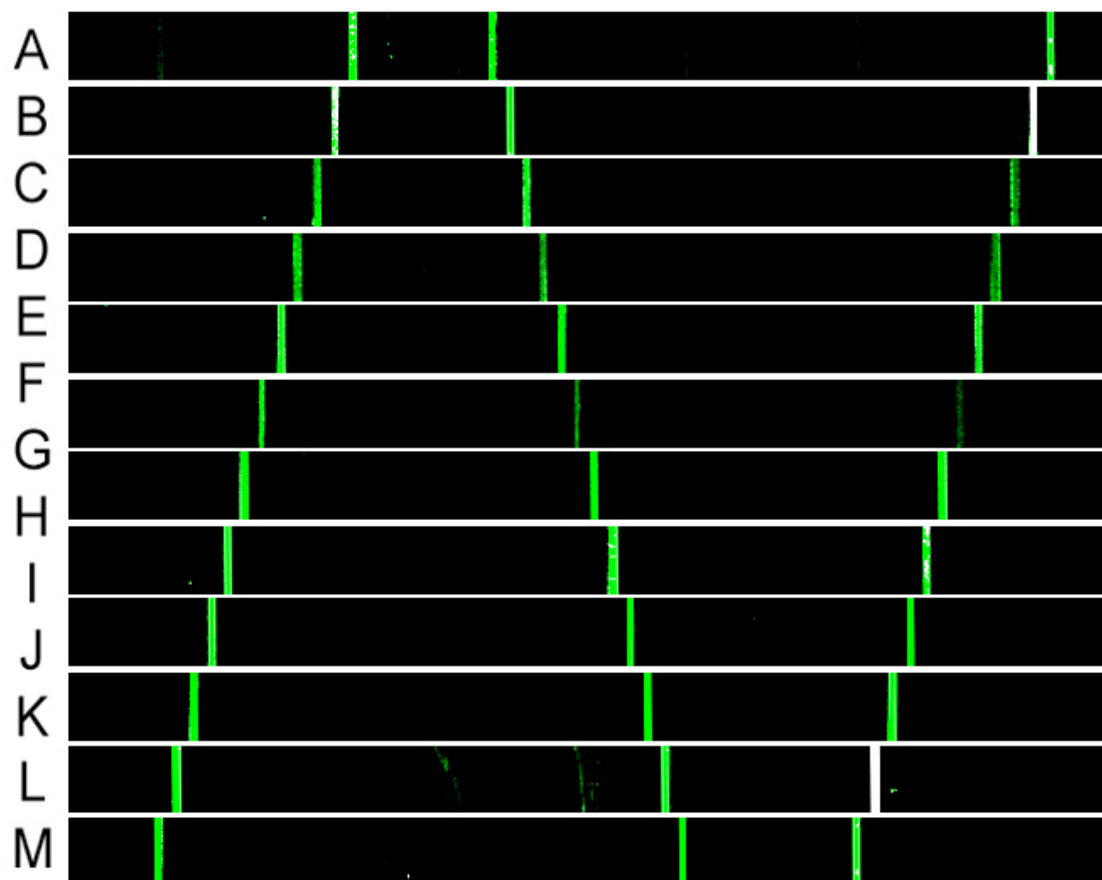


Figure 2.6 Cross-hybridization assay for all 13 DNA oligomer pairs that were used for encoding the registry of antibody barcode arrays.

2.5.4 Patterning of Barcode Arrays

Using the microchannel-guided flow-patterning approach (Figure 2.7), we fabricated DEAL barcode arrays that were ~10-fold denser than conventional microarrays. Microcontact printing can generate high density arrays of biomolecules with spot sizes of a few micrometers

(μms),^{2,3} but extending stamping to large numbers of biomolecules is awkward because of the difficulty in aligning multiple stamps to produce a single microarray. Direct microfluidics-based patterning of proteins has been reported, but DNA flow-patterning with sufficient loading remains less successful compared to conventional spotting methods.^{4,5} In the flow patterning process, a polydimethylsiloxane (PDMS) mold containing 13-20 parallel microfluidic channels, with each channel conveying a different biomolecule capture agent, was used. The number of channels could readily be expanded to include 100 or more different capture agents. Poly-amine coated glass surfaces permitted significantly higher DNA loading than do more traditional aminated surfaces, with a corresponding increase in assay sensitivity (**Figure 2.8**). DNA “bars” of 2 micrometers in width could be successfully patterned. In the present study, a 20-micrometer (μm) channel width was chosen because the fluorescence microarray scanner utilized has a resolution of 5 μm . The fabrication details are as follows:

Mold fabrication. The microfluidic-patterning chips were made by molding a PDMS elastomer from a master template, which was prepared using photolithography to create a photoresist pattern on a Si wafer. An alternative was to make a silicon “hard” master by transferring the photolithographically-defined pattern into the underlying silicon wafer using a deep reactive ion etching (DRIE) process.⁶ The first method offers rapid prototyping, while the second method yields a robust and reusable mold, permitting higher throughput chip fabrication.

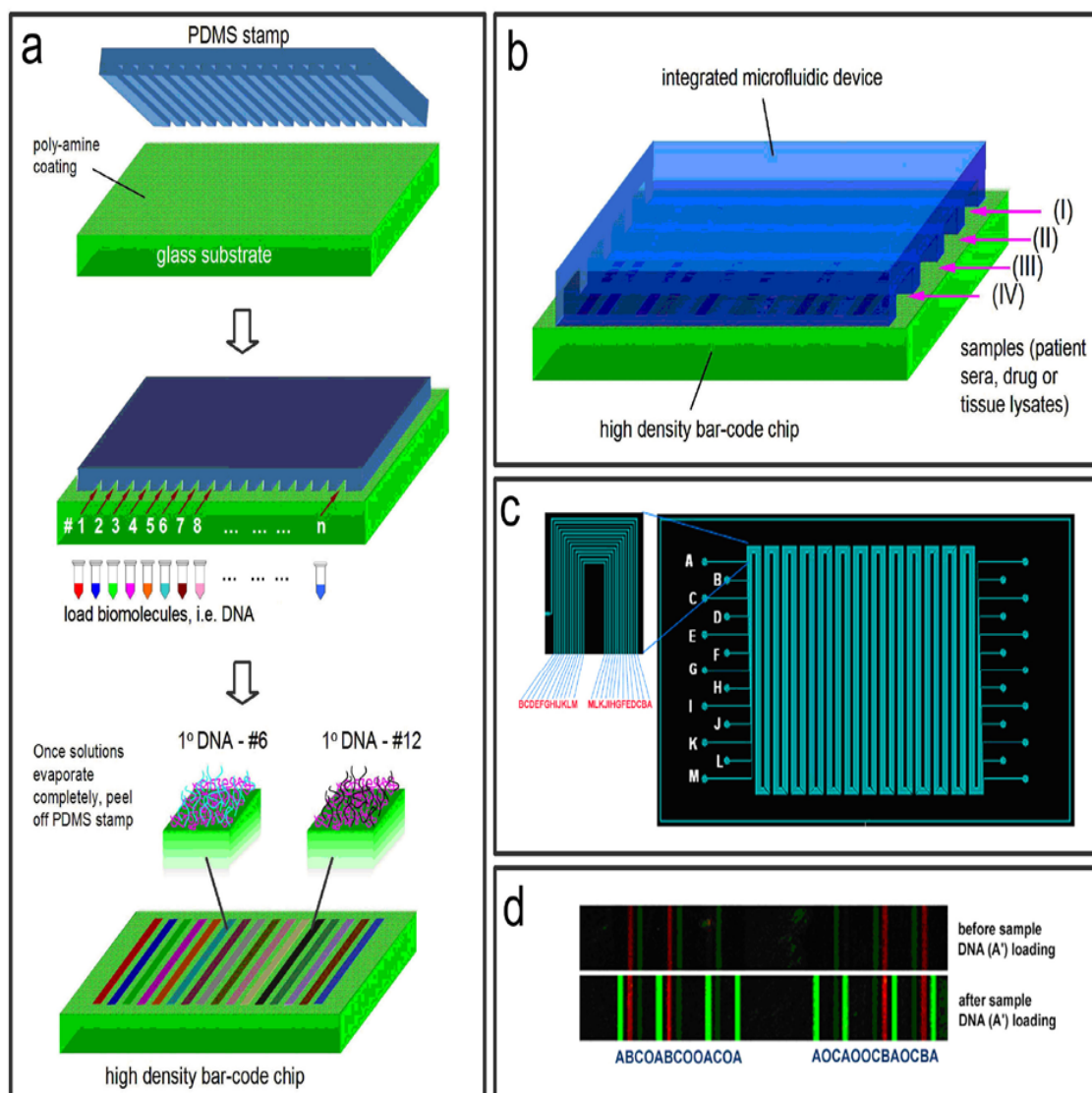


Figure 2.7 Microchannel-guided flow patterning of DEAL barcode arrays. (a) Depiction of the procedure. Each DNA bar is 20µm wide and spans the dimensions of the glass substrate. (b) Integration of a DEAL barcode-patterned glass slide with microfluidics for multiplexed protein assays. (c) Mask design of a 13-channel barcode. A-M denotes the flow channels for the different DNA molecules. (d) Validation of successful patterning of DNA molecules by specific hybridization of oligomer A to its fluorescent complementary strand A'. The primary strands B and C were pre-tagged with red and green dyes as references.

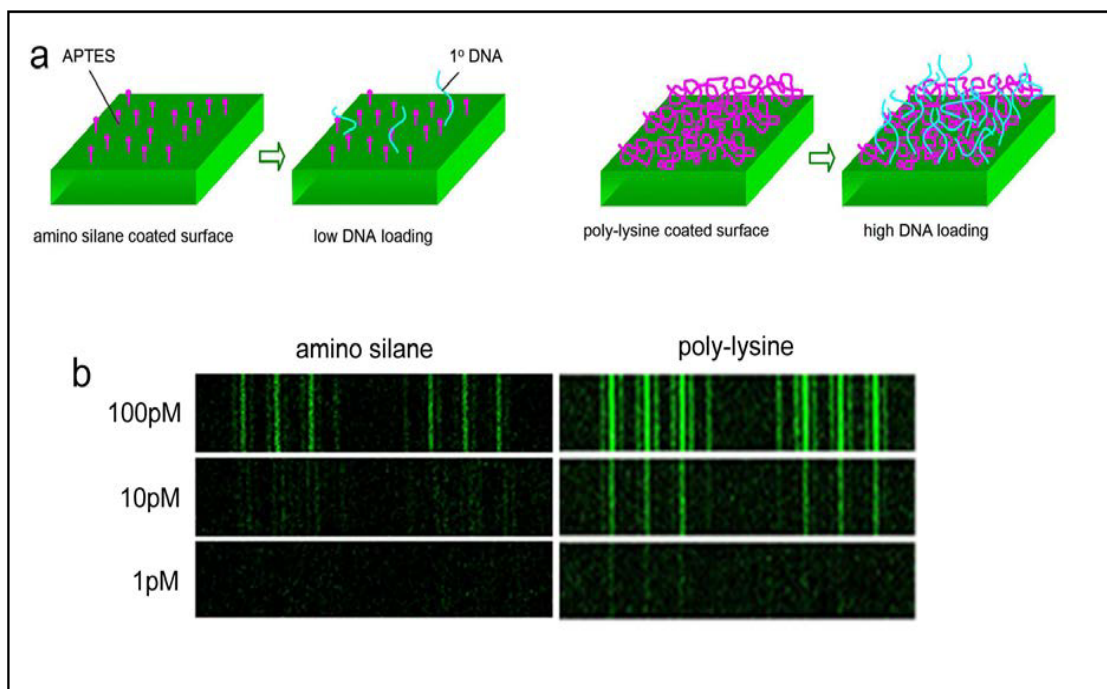


Figure 2.8 Effects of polylysine coating on DEAL assay. (a) Schematic illustration of polylysine coating for increased loading of DNA oligomer codes. (b) Fluorescence images showing a comparative study of the measurement of three human cytokines (IFN- γ , TNF- α , and IL-2) using substrates coated with amino-silane and polylysine, respectively.

PDMS patterning-chip fabrication. A polydimethylsiloxane (PDMS) elastomer slab was fabricated by casting the Sylgard[®] 184 wet PDMS (prepolymer:curing agent = 10:1 (w/w)) onto the molds described above followed by a curing step at 80°C for 50 minutes. This slab was peeled from the mold and was bonded onto a glass surface, which provided the base walls for the flow channels. Prior to bonding, the glass surface was pre-coated with the polyamine polymer, poly-L-lysine (Sigma-Aldrich), to increase DNA loading. The coating process is described elsewhere.¹⁸ The number of microfluidic channels determines the size of the barcode array. In the present work, the PDMS chip, as shown in **Figure 2.7c**, contained 13 to 20 parallel microchannels designed to cover a large area (3cm \times 2cm) of the glass slide with the DNA barcode microarray.

DEAL barcode patterning. Solutions, each containing a different primary DNA oligomer prepared in 1X PBS buffer, were flowed into each of the microfluidic channels. Then, the solution-filled chip was placed in a desiccator for several hours (or overnight) to allow solvent (water) to evaporate completely through the gas-permeable PDMS, leaving the DNA molecules behind. Last, the PDMS elastomer was removed from the glass slide, and the barcode-patterned DNA was fixed to the glass surface by thermal treatment at 80°C for 4 hours, or by UV cross-linking. Potassium phosphate crystals precipitate during solution evaporation, but are readily removed by rapidly dipping the slide in deionized water. The barcode-patterned DNA arrays demonstrated a marked improvement in sensitivity as compared to conventional pin-spotted microarrays (**Figure 2.9**). A side-by-side comparison study was performed by running DEAL assays on three cytokines under identical conditions. Using the microchannel-guided flow patterning method, a glass slide was patterned with DNA oligomers **A**, **B**, **C**, and a blank control **O** (20 μm -wide bars; 50-100 μM DNA solutions). The pin-spotted array, with a typical spot size of 150-200 μm , was printed at the Institute for Systems Biology using 100 μM oligomer concentrations. Six sets of spots were printed, corresponding to oligomers **A**, **B**, **C**, **D**, **E**, and **F**. Poly-L-lysine coated slides were used for both types of arrays.

Before the DEAL assay, the capture antibodies were conjugated to DNA oligomer codes as follows: **A'** to IFN- γ , **B'** to TNF- α , and **C'** to IL-2. Protein standards were diluted in 1% BSA/PBS solution at concentrations ranging from 1fM to 1nM. The incubation time for each step (blocking, conjugate hybridization, sample binding, detection-antibody binding, and fluorescent-molecule binding) was 30 minutes. The results (**Figure 2.9b**) reveal that the DEAL

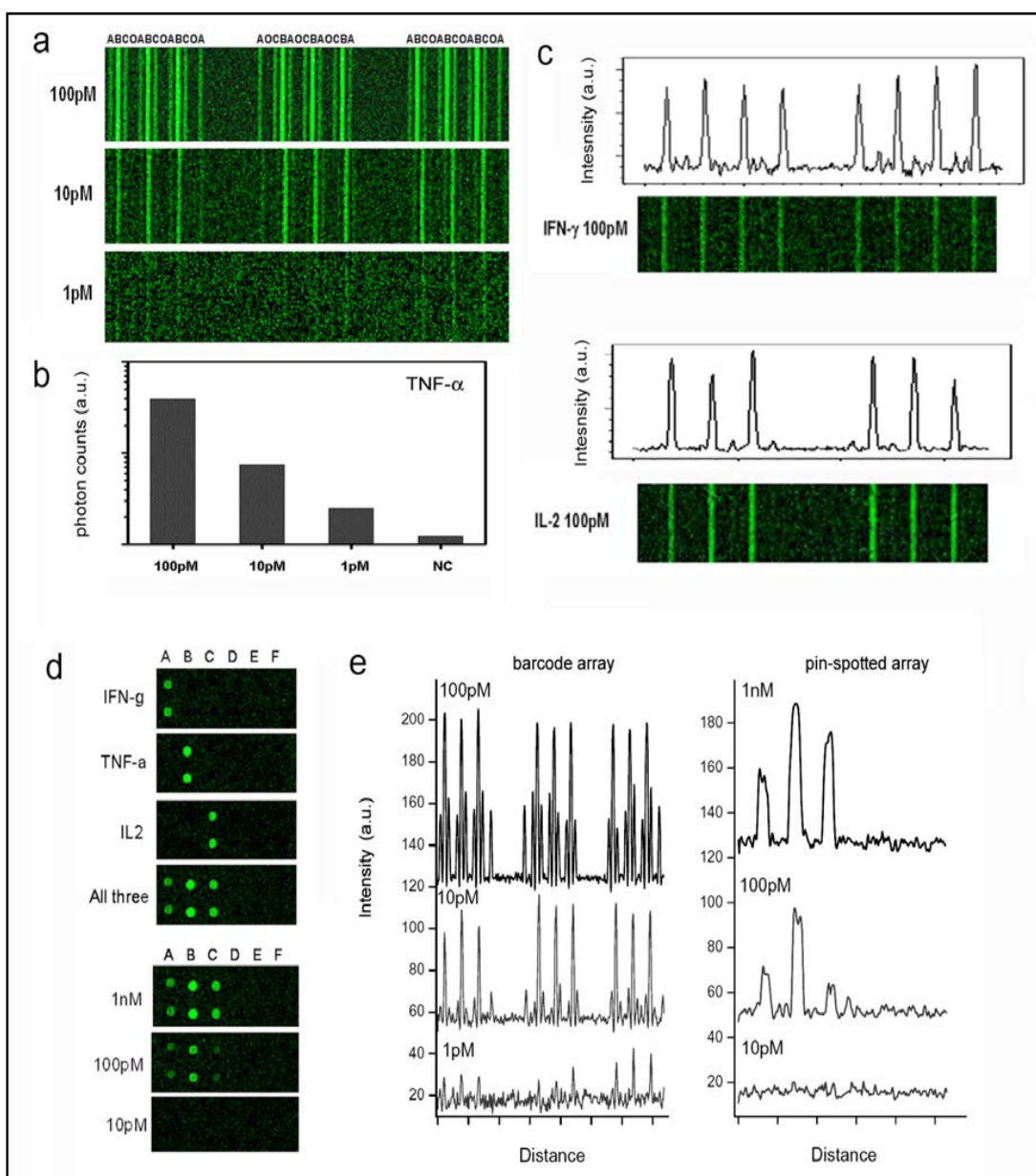


Figure 2.9 Increased sensitivity observed in immunoassays run on DEAL barcode arrays. (a) Concentration-dependent fluorescence signal for the detection of three human cytokines (A: IFN- γ , B: TNF- α , C: IL-2, O: negative control) using a DEAL barcode array. The bar width is 20 μ m. (b) Quantitation of fluorescence intensity vs. TNF- α concentration. (c) Measurements of individual proteins, IFN- γ and IL-2, reveal no distinguishable cross-reactivity. (d) Comparison of the microfluidics flow-patterned DEAL microarrays with DEAL microarrays patterned using a conventional DNA pin-spotting method. The spot size is \sim 150-200 μ m. (e) Fluorescence line profiles for the DEAL barcode array in a and the pin-spotted array in d at different protein concentrations. The curves were amplified in the y-coordinates for better visualization.

barcode array sensitivity is similar to the projected sensitivity limit of the commercial ELISA assay (~10 pg/mL, or 0.8 pM; eBioscience). Taking the example of the TNF- α assay, the detection sensitivity of the DEAL barcode array (better than 1 pM) is substantially improved over the 10-100 pM limit found for the microarrays spotted using conventional methods (**Figure 2.9d**). Therefore, the DEAL barcode array combines ELISA-like sensitivity with a high degree of multiplexing for protein measurements.

The difference in sensitivity between the barcode array and pin-spotted array platforms is likely attributable to the difference in feature size. The barcode array has a line-width of 20 μm , whereas the spot diameter in conventional arrays is more than 150 μm . These results are consistent with a recent report which demonstrated that DNA microarrays with smaller spot sizes could detect DNA with increased sensitivity.⁷

The ELISA-like sensitivity of the DEAL barcode assays is key for realizing the multiplexed measurement of human plasma proteins in blood. The human plasma proteome is comprised of three major classes of proteins – classical plasma proteins, tissue leakage proteins, and cell-cell signaling molecules (cytokines and chemokines). Cell-cell signaling molecules are biologically informative in a variety of physiological and pathological processes, i.e. tumor host immunity and inflammation. The concentration range of plasma proteins within the human plasma proteome spans 12 orders of magnitude, and the lowest end is approximately at the detection limit of mass spectrometry – a high-throughput protein profiling technique. The state-of-the-art for clinical protein measurements is still the typically low-throughput ELISA assay. The high performance of the DEAL barcode chip, including its increased sensitivity, is a key to realizing highly multiplexed measurements of a panel of proteins from small quantities of clinical blood samples.

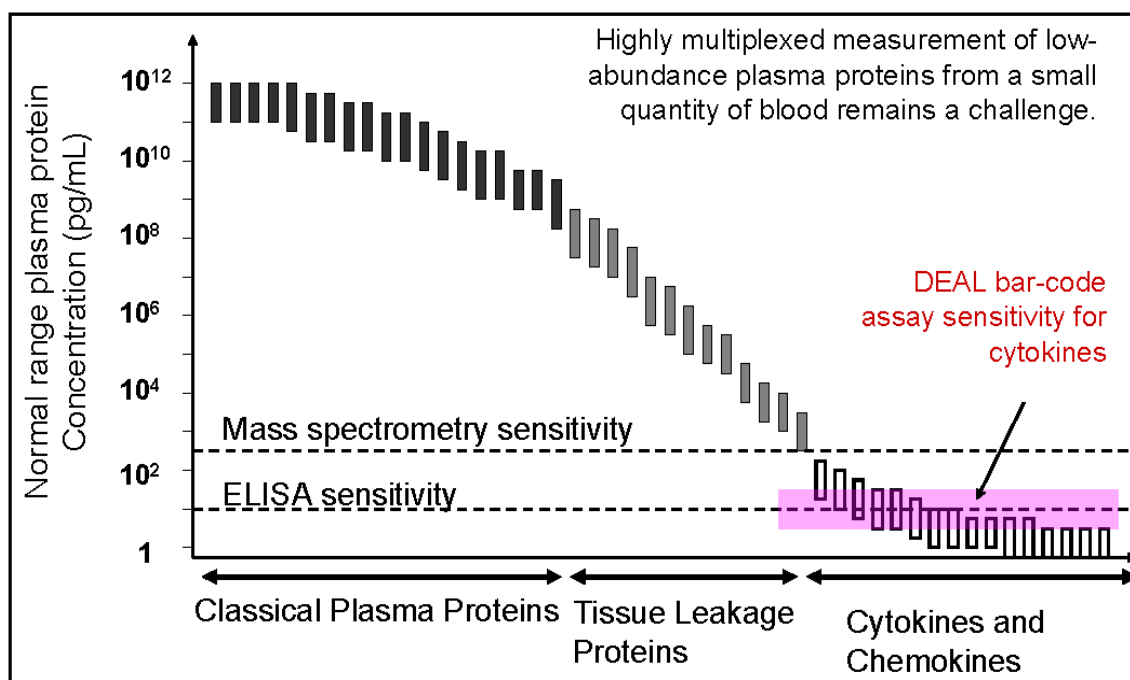


Figure 2.10 Schematic of human plasma proteome (refer to N.L. Anderson and N.G. Anderson, *Molecular & Cellular Proteomics* 11, 845, 2001). Our work demonstrates that the DEAL barcode assay has a markedly increased sensitivity, comparable to ELISA, leading to the feasibility of multiplexed detection of plasma proteins, including low-abundance cell-cell signaling molecules, e.g. cytokines and chemokines, from a small quantity of sample.

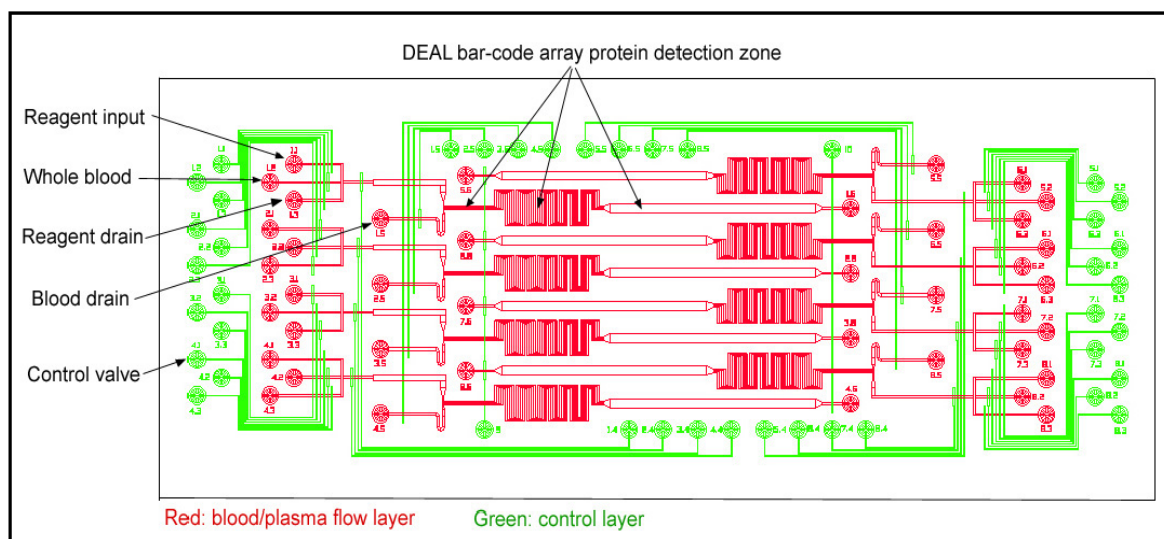


Figure 2.11 AutoCAD design of an IBBC. Underneath the PDMS microfluidic chip is a large-scale DNA barcode array. Flow layer in red; control layer in green.

2.5.5 Fabrication of IBBCs

The fabrication of the IBBCs was accomplished through a two-layer soft lithography approach.^{8,9} A representative chip design is shown in **Figure 2.11**. The silicon master for the control layer (red) was fabricated by exposing a spin-coated SU8 2010 negative photoresist film (~20- μm thickness). Prior to molding, the master was silanized in a trimethylchlorosilane (TMCS) vapor box for 20 minutes. A mixture of GE RTV 615 PDMS prepolymer part A and part B (5:1) was prepared, homogenized, and then applied onto the control layer master. After degassing for 15 minutes, the PDMS was cured at 80°C for 50 minutes. The solidified PDMS chips were then cut and peeled off the master, and access holes were drilled with a 23-gauge stainless-steel hole punch.

The flow-layer master (blue) was fabricated using SPR 220 positive photoresist. After exposure and development, the photoresist pattern was baked at 120°C in a convection oven to round the flow channels. The resultant flow layer was typically 15-20 μm in thickness. Silanization using TMCS was performed right before applying the fluid PDMS prepolymer. Next, a mixture of GE RTV 615 PDMS part A and part B (20:1) was prepared, homogenized, degassed, and then spun onto the flow layer master at 2000-3000 rpm for 1 minute. It was cured at 80°C for 30 minutes, at which point the PDMS control layer was carefully aligned and placed onto the flow layer. Finally, an additional 60-minute thermal treatment at 80°C was performed to bond the two PDMS layers together. The bilayer chip was then peeled off of the flow-layer master and access holes were drilled.

The last assembly step was to bond the PDMS chip to the DEAL barcode slide via thermal treatment at 80°C for 4 hours, yielding a completed integrated blood barcode chip (IBBC). In this chip, the DEAL barcode stripes are orientated perpendicular to the microfluidic

assay channels. The IBBC features a microfluidic biological fluid-handling module, specifically a whole blood separation unit, and a DEAL barcode array for highly multiplexed protein measurements. In a typical design, 8-12 identical blood separation and detection units were integrated within a single 2.5 cm x 7 cm chip.

2.5.6 Execution of Blood Separation and Multi-Parameter Protein Assay using IBBCs

The compatibility of the DEAL technique with integrated microfluidics yielded rapid blood separations and reliable measurements of a panel of proteins. The experimental procedure is detailed below.

- a. *Blocking*: Prior to use of the IBBC, all microfluidic channels were blocked with the assay buffer solution (1% w/v BSA/PBS solution prepared by adding 98% pure Bovine Serum Albumin, Fraction V (Sigma) to 150 mM 1X PBS without calcium/magnesium salts (Irvine Scientific) for 30-60 minutes.
- b. *DEAL formation (introducing conjugates)*: A solution containing all the DNA-antibody conjugates was flowed through the assay channels of the IBBCs for ~30-45 minutes, thus transforming the DNA barcode microarray into an antibody microarray, enabling the subsequent surface-bound immunoassay. The unbound conjugates were removed by flowing the assay buffer solution for 10 minutes. The DEAL-conjugate solution was prepared by mixing all synthesized conjugates in 1% BSA/PBS with a final concentration of 5 µg/mL. The DNA coding oligomers were pre-tested for orthogonality to ensure that cross-hybridization between non-complementary oligomer strands yielded a fluorescence intensity that did not exceed 5% of the complementary pair signal intensity.

- c. *Collecting a finger-prick of blood:* Finger pricks were carried out using BD Microtainer Contact-Activated Lancets (purple lancet – for low volume, single blood drop). Blood was collected with SAFE-T-FILL capillary blood collection tubes (RAM Scientific), which were prefilled with a 25 mM EDTA solution as discussed below. Two samples were prepared from the drop of whole blood:
- i. Unspiked Blood Samples: The blood collection tube was pre-filled with 80 μL of 25 mM EDTA solution, and then 10 μL of fresh human blood was collected in the EDTA-coated capillary, dispensed into the tube and rapidly mixed by inverting a few times.
 - ii. Spiked Blood Samples: The blood collection tube was pre-filled with 40 μL of 25 mM EDTA solution. Forty microliters of recombinant protein solution, containing all the protein standards, was added. Then, 2 μL of 0.5 M EDTA was added to bring the total EDTA concentration up to 25 mM. Finally, 10 μL of fresh human blood was collected in an EDTA-coated capillary, added to the tube and quickly mixed by inverting a few times. The final concentrations for all protein standards were ~ 10 nM. However, the quality of these “standards” and the affinity of capture antibodies vary substantially. The purpose of spiking in protein standards was to contrast the signal at high protein concentrations with that of as-collected fresh whole blood.
- d. *Blood sample assay:* These two blood samples were flowed into the IBBCs within 1 minute of collection. The plasma was quickly separated from blood cells within the chip, and the proteins of interest were captured in the downstream assay zone containing the DEAL barcode arrays. The entire process from finger prick to the completion of plasma

protein capture was very rapid (<10 minutes). Owing to the reduced diffusion barrier in a flowing microfluidic environment, the sample assay was executed within *ten minutes*. With regards to the cancer patient serum tests, the as-received serum samples (Asterand) were flowed into IBBCs without further treatment.

- e. *Applying detection antibodies*: A mixture of biotin-labeled detection antibodies was flowed into the microfluidic devices for ~30 minutes to complete the DEAL assay. The detection-antibody solution contained biotinylated detection antibodies at ~5 μM prepared in 1% BSA/PBS. Afterwards, unbound detection antibodies in the IBBCs were removed by flowing the assay buffer for 10 minutes.
- f. *Fluorescence probes*: For the cancer serum experiments, Cy5 fluorescent dye-labeled streptavidin and the reference, Cy3-labeled complementary ssDNA (DNA code M/M'), were mixed together and were then flowed into the IBBCs for 30 minutes. Finally, the assay buffer was flowed for 10 minutes to remove unbound Streptavidin-Cy5.
- g. *Rinse*: The PDMS blood chip device was removed from the DNA-patterned glass slide. The slide was immediately dipped 6 times each in the following solutions in order: 1% BSA/PBS solution, 1X PBS solution, $\frac{1}{2}\text{X}$ PBS solution, deionized Millipore H_2O . The slide was rinsed for a few seconds under a Millipore H_2O stream, and then dried with a nitrogen gun.
- h. *Optical readout*: The slide was scanned by an Axon Instruments GenePix Scanner. The finest resolution (5 μm) was selected. Two color channels (the green Cy3 channel and the red Cy5 channel) were turned on to collect fluorescence signals.

2.5.7 Consideration of Microfluidic Environment for Rapid Immunoassay

In a microfluidic environment, at sufficiently high flow rates, diffusion is not limiting, and the rate at which a biological assay can be completed is determined by the kinetic parameters that describe the capture of the biomolecule by the surface-bound capture agent.¹⁰ Under chemical equilibrium, the relative amount of biomolecule that is complexed to the surface-bound capture agent is given by:

$$K_{eq} = [\text{biomolecule-CA complex}] / [\text{biomolecule in solution}][\text{Surface Bound CA}]$$

where K_{eq} is the equilibrium constant. For a given concentration of biomolecule, the surface-bound assay sensitivity depends upon several factors, including:

- The equilibrium constant, K_{eq} : a large K_{eq} corresponds to a large amount of the biomolecule-CA complex.
- The concentration of surface-bound CA: we find that the sensitivity limits of the assay directly correlate with the concentration of surface-bound CA.¹ During microchannel-guided flow-patterning of the DEAL barcode arrays, the glass surface was modified by treatment with poly-L-lysine (a poly-amine), yielding a three-dimensional matrix for DNA adsorption and markedly increasing the amount of DNA loading. Our DNA-loading density is estimated to be 6×10^{13} molecules/cm², an order of magnitude higher than typical loading densities on amino-silane coated glass slides.¹¹ As a result, the protein detection sensitivity was improved by an order of magnitude, and the dynamic range was increased to 4 orders of magnitude, as compared with 2-3 orders of magnitude for the small-molecule amine (i.e. amino-propyl-triethoxyl silane, APTES) functionalized glass surface. The comparative study is shown in **Figure 2.8**.

- Feature size: smaller feature sizes can lead to increased sensitivities.⁷ The feature sizes (20 μm -wide stripes) of DEAL barcode chips are substantially smaller than are generated using more traditional spotting methods (150- μm diameter spots).

2.6 Appendix B: Supplementary Data

2.6.1 Blind Test of Serum Samples Containing Unknown hCG Concentrations

Two serum samples containing unknown concentrations of hCG were measured in a blind test using a DEAL barcode assay. These two samples were introduced alongside eight standards of hCG-spiked serum on the same DEAL barcode chip. Fluorescent images were acquired at the same laser settings and quantified from 20 sets of barcodes. Using the resulting standard curve (at 200- μ M DNA loading), we estimated the hCG concentration. The results are shown in **Figure 2.12** (inset table), and compared to the test results from an outside independent laboratory (Labcorp; the test was requested by the Nanotechnology Characterization Laboratory at the National Cancer Institute). The DEAL barcode assay result is in reasonable agreement with the Labcorp results.

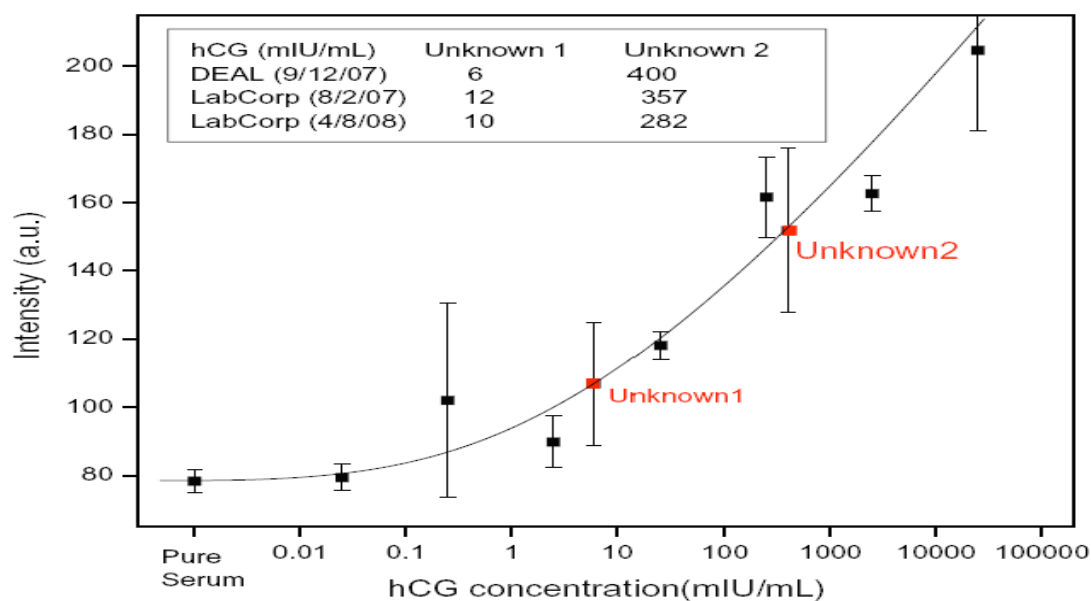


Figure 2.12 Blind test of hCG-containing “unknown” samples. The black squares correspond to the 200 μ M DNA loading shown in **Figure 2.2**, and the red squares show the statistical means of two unknowns. The insets show representative images of barcode assay results for two unknowns, plus a table showing the estimated concentrations measured using the DEAL barcode assay and by an independent laboratory (Labcorp).

2.6.2 Protein Cross-Reactivities

We assessed the level of cross-reactivity of each antigen with DEAL stripes that are not specific to that antigen. DNA-encoded capture antibodies and biotinylated detection antibodies for all 12 antigens were used as usual, but a distinct antigen (10 nM) was added to each assay lane. Cy5-Streptavidin (red-fluorescence tag) was added to visualize the extent of analyte capture. The reference marks (DNA strand M) were visualized in all lanes with fluorescent green Cy3-M' DNA molecules. The 12 proteins showed a negligible extent of cross-reactivity (**Figure 2.13**), with typical photon counts under 2% compared to the correctly paired antigen-antibody complexes. Even at this low level, most of these cross-talk signals were found to be due to degraded recombinants (protein standards) and did not appear reproducibly once a new recombinant was used. This minimal cross-talk was also validated in pin-spotted microarrays using the same set of primary DNA codes. The negligible cross-talk in our DEAL assays is largely attributable to our significant efforts to screen for orthogonal DNA pairs (**Tables 2.1 and 2.2**). A non-fully orthogonal DNA pair leads to cross-hybridization, and resultant cross-reactivity in the DEAL protein assay.¹

2.6.3 Dilution Curves for all Proteins used in the DEAL Barcode Assay

We performed assays on serial dilutions of all 12 proteins on the DEAL barcode chip. Because each device allows a maximum of 12 parallel assays to be executed, we chose 6 lanes for cross-talk validation, leaving 6 lanes for dynamic range studies. As a result, we combined 2 proteins in each assay lane (**Figure 2.14**). On the same chip, we assayed all proteins over the concentration range of 1 nM – 1 pM (except PSA and TGF- β : 5 nM to 5 pM), and quantified the fluorescence signal vs. concentration for all 12 antigens (**Figure 2.14b**). All assay lanes were

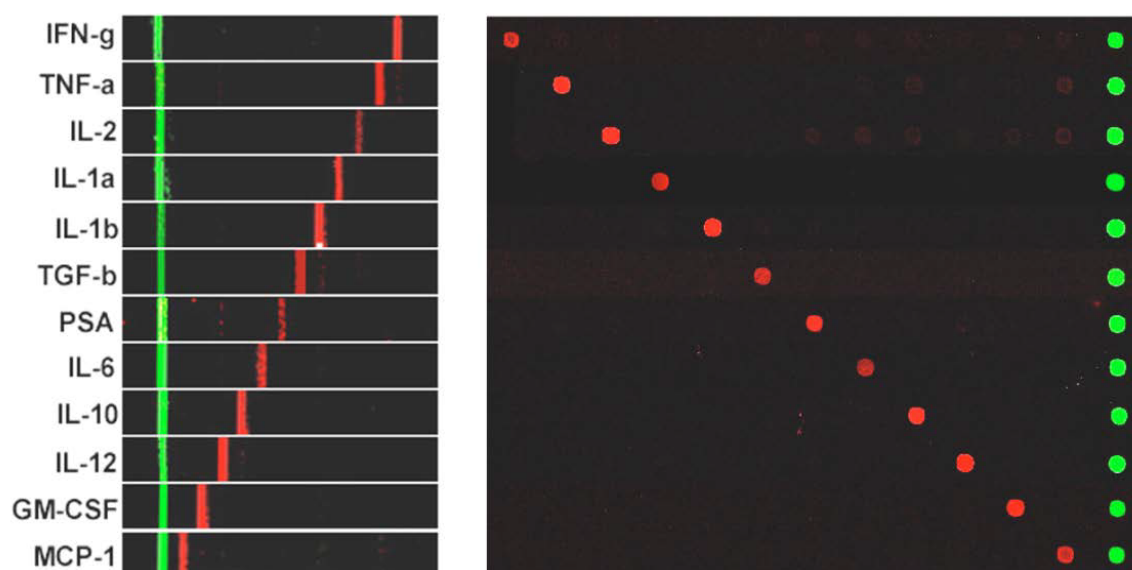


Figure 2.13 A cross-reactivity assay for the entire biomarker panel of 12 proteins. Both barcode (left panel) and pin-spotted (right panel) microarray formats are shown. The green bars represent the reference stripe/spot – M. Each protein can be readily identified by its distance from the reference. The designation of proteins in the barcode is the same as in **Figures 2.3 and 2.4**.

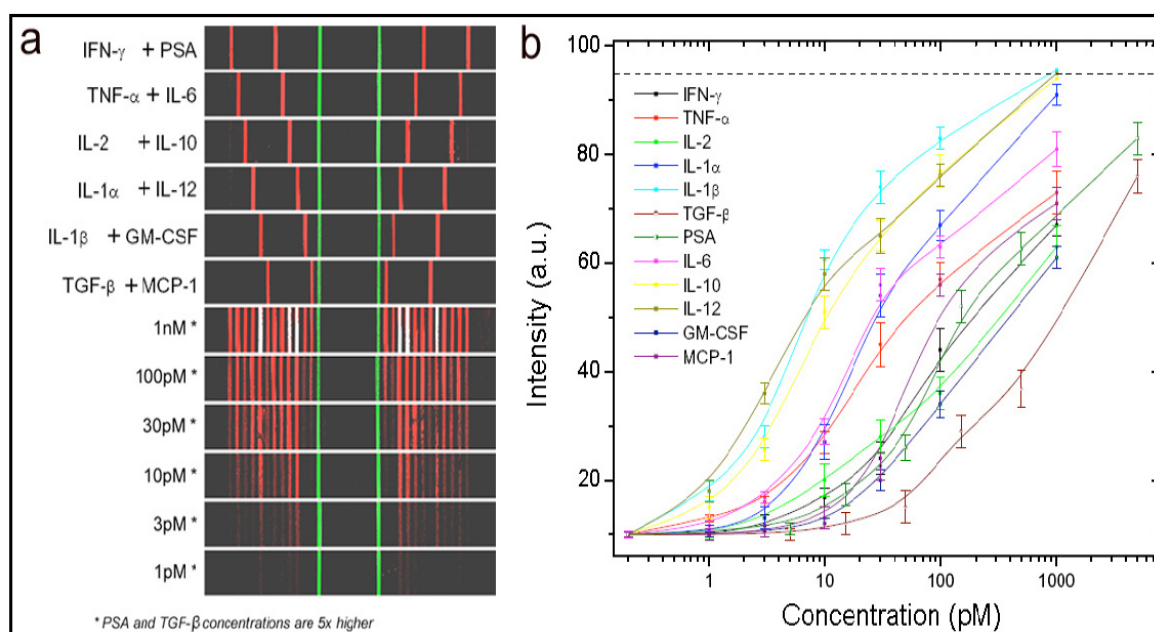


Figure 2.14 Dilution curves for the 12 proteins measured using DEAL-based barcodes entrained within microfluidic channels. (a) Barcode images from one device showing minimal crosstalk, and a series of standard antigens ranging from 1 nM to 1 pM for all 12 proteins (* the concentrations of PSA and TGF- β are 5x higher). (b) Quantitation of fluorescence intensity vs. concentration for all 12 proteins. Error bars: 1 s.d.

imaged using the GenePix scanner (with the same scanning parameters as described in the Experimental Methods section). Apparently, the estimated sensitivity varies substantially depending upon the antibodies being used, from ~0.3 pM (e.g. IL-1 β and IL-12) to 30 pM (TGF- β). The TGF- β antibody pair has a relatively lower binding affinity and a poorer detection limit in ELISA (~70 pg/mL, compared to 5-10 pg/mL for most other cytokines, according to the specifications sheet). Predictably, this gave rise to a poorer performance in the DEAL assay. Although these curves reflect the ability of the microfluidics-patterned DEAL assays to assess specific antigens over broad concentration ranges, the statistical variation is relatively large compared to a commercial ELISA assay.

2.6.4 Standardized Quantification of the Patient Serum DEAL Barcode Data

Barcode signal quantitation was performed through a standardized process designed to minimize arbitrary bias in the image analysis. First, the fluorescence from the barcodes was visualized under fixed conditions, using the Axon GenePix 4000B two-color laser microarray scanner with identical instrument settings. For 635 nm and 532 nm excitation lasers, respectively, the following settings were used: Laser Power - 100% and 33%; Optical Gain - 800 and 700; Brightness/Contrast - 87 and 88. Next, the resulting JPEG image was exported from *GenePix 6.0* and was resized to match the standard barcode-array mask-design image. NIH *ImageJ* was employed to generate an intensity line profile of each assay channel and subchannel.

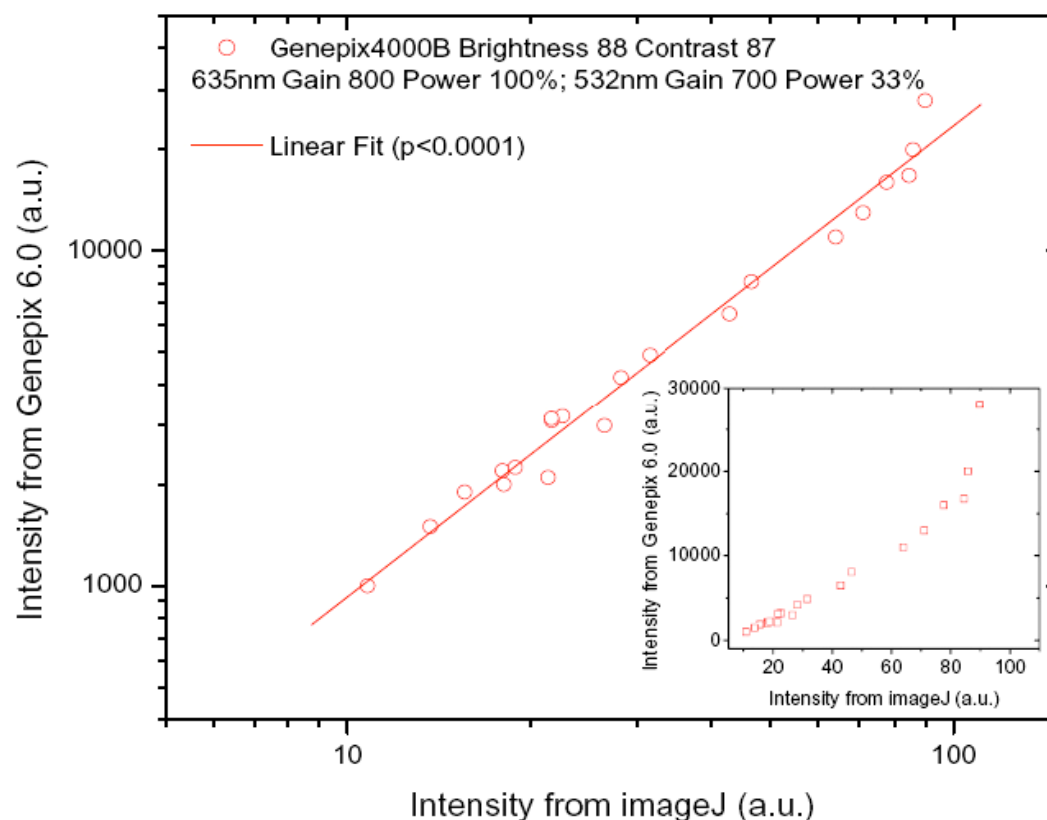


Figure 2.15 Comparison of fluorescence intensities quantitated using the *GenePix 6.0* and NIH *ImageJ* programs. The graph shows good linear correlation up to 85 a.u. (per pixel), where the *ImageJ* quantitation begins to saturate.

Each data point in the line profile was averaged from 40 pixels (200 μm) along the vertical direction. Fourth, all the line profile data files were loaded into a custom-written Excel macro that generated a spreadsheet tabulating the average intensities of all 13 bars (1 bar = 5×40 pixels, or $25 \times 200 \mu\text{m}$ area) in each of 20 barcodes. The statistical analysis generated mean values and standard deviations (in Microcal Origin).

To demonstrate the feasibility of using the NIH *ImageJ* program, we performed a calibration by comparing the photocounts from the *GenePix* scan and the brightness values quantitated from *ImageJ* (Figure 2.15). A linear correlation ($p < 0.0001$) exists if the intensity from *ImageJ* is no higher than 90. The scale of brightness in red-green-blue (RGB) mode is 0-255, so each color has a brightness maximum at 85. Since a common baseline (~ 10) is

superimposed onto the red fluorescence signal, the actual maximum is at ~90-100. The calibration data validates the use of *ImageJ* for quantitating the DEAL barcode data. We are currently pursuing a fully automated process for feature recognition, signal quantitation, and concentration extraction.

The mean values of measured protein levels for every patient were exported into a matrix for non-supervised clustering of patients. This was performed using the software, *Cluster 3.0*, and the heat map was generated using the software, *Java Treeview*. To assess the statistical significance between two patient (sub)groups, the Student's *t* analysis was performed on selected proteins and all *p*-values were calculated at a significance level of 0.05 if not otherwise specified. This standardized quantitation process diminished any possible biases in the manual quantitation process, but was unable to identify and exclude interfering speckles and noises in the images. A dust particle atop a barcode can give rise to an extremely bright scattering signal, thus causing large errors in quantitating such a barcode. In addition, several samples such as P04, P05, P10, and B10 exhibited significant bio-fouling and introduced a large non-specific background into quantitation. These issues remain to be resolved in further development of the DEAL barcode assay.

Quantitation and statistics of the barcodes for all patient sera are shown in **Figures 2.16** and **2.17**. The data for all proteins in these plots were shown in the same order as indicated in **Figure 2.3a**.

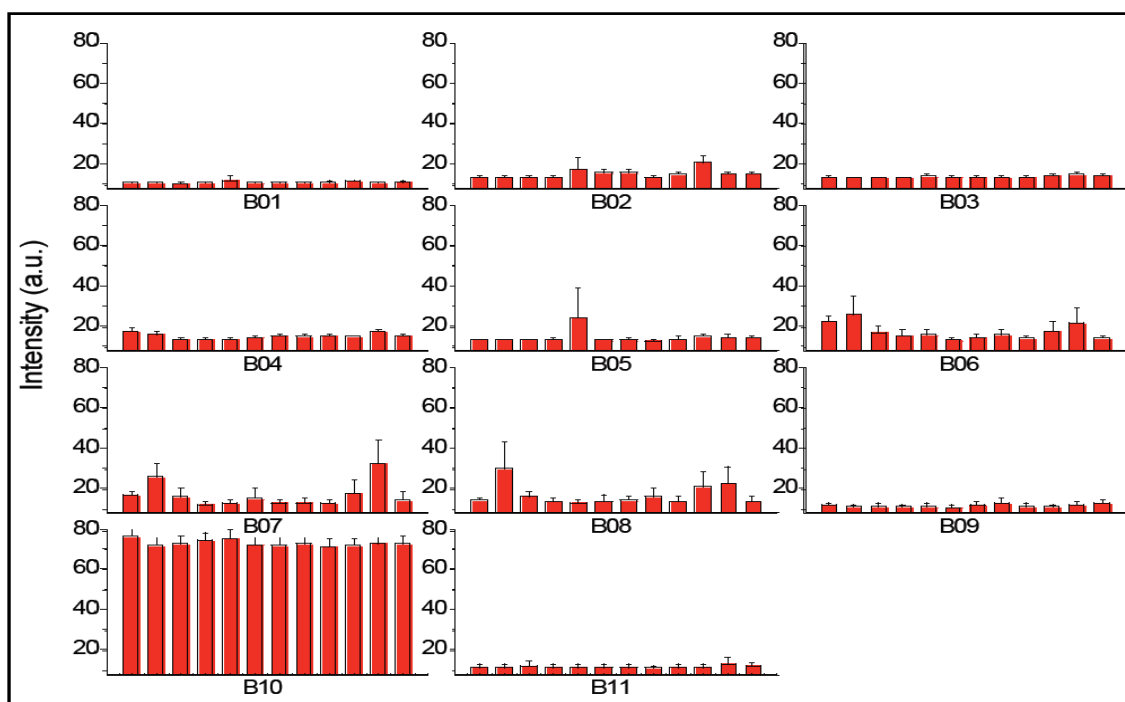


Figure 2.16 Quantitation of the fluorescence intensities from measurements of 11 breast cancer patients (B01-B11). The proteins are shown in the same order as described in Figure 2.3a.

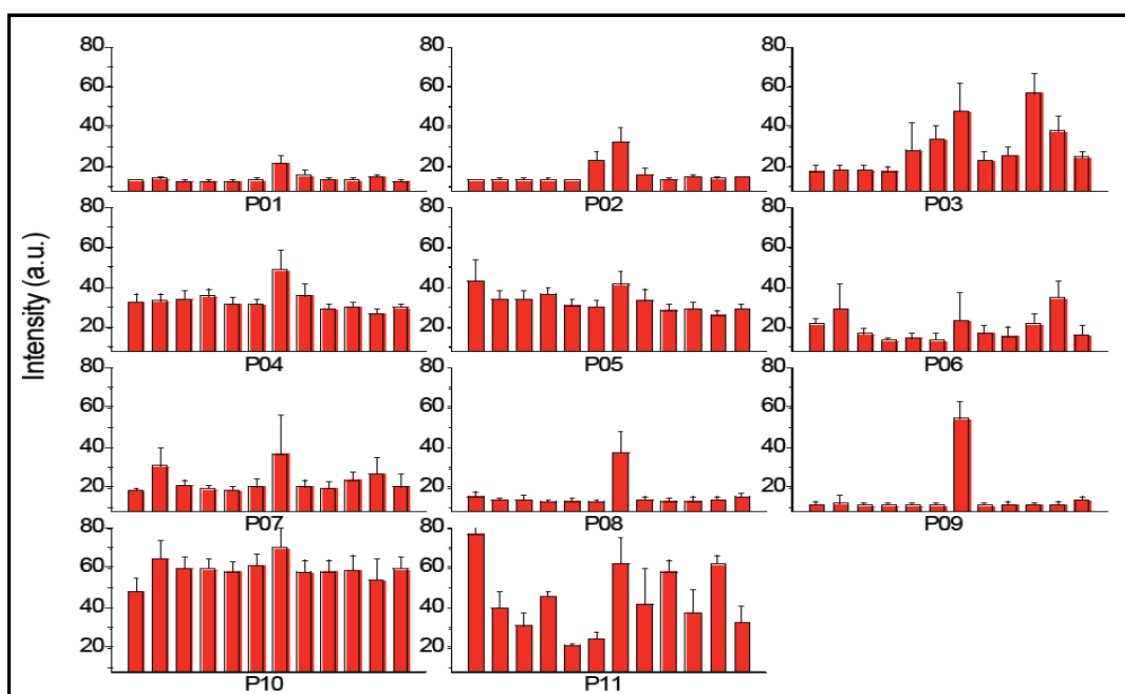


Figure 2.17 Quantitation of the fluorescence intensities from measurements of 11 prostate cancer patients (P01-P11). The proteins are shown in the same order as described in Figure 2.3a.

2.6.5 ELISA Validation of DEAL Barcode Assay

Given our limited serum sample quantities, and the amount of material required for carrying out standard protein measurements, we could only perform strategic cross-validations of the DEAL barcode measurements. PSA was the single protein that readily discriminated between breast and prostate cancer patients, and, in addition, clinical measurements of PSA levels for most of the prostate cancer patients were available through the serum vendor (Asterand). Thus, for all 22 patient serum samples, we performed enzyme-linked immunosorbant sandwich assays (ELISAs) to independently assess the PSA levels.

A comparative study of PSA levels measured by ELISA and by the DEAL barcode assays is shown in **Figure 2.3c**. One set of ELISA data was collected in our lab using the standard 96-well plate format. A second set of ELISA data, for 8 of the prostate cancer patient serum samples, was measured in commercial (clinical) labs. The DEAL barcode data is taken from **Figure 2.3b**. For our own ELISA measurements (**Figure 2.18a**), the first row of wells was loaded with PSA standards at serial dilutions. In all, 22 serum samples and a negative control (buffer) were measured. All three data sets are presented in **Figure 2.3c**, and are in good agreement with one another. The DEAL barcode assay detects the presence of PSA with 100% accuracy, but is less accurate relative to the ELISA tests in terms of quantitating small changes in concentration. This may be due, in part, to slight variations both in our manual chip assembly procedures and in our manual assaying procedures. It may also be that the higher sensitivity and concentration range of the DEAL assays is accompanied by a reduction in accuracy. We are in the process of fully automating both our microfluidics barcode patterning method and our assay

procedure. Those advances will allow us to more fully assess the trade-offs between the sensitivity and accuracy of the DEAL barcode arrays versus traditional ELISA formats.

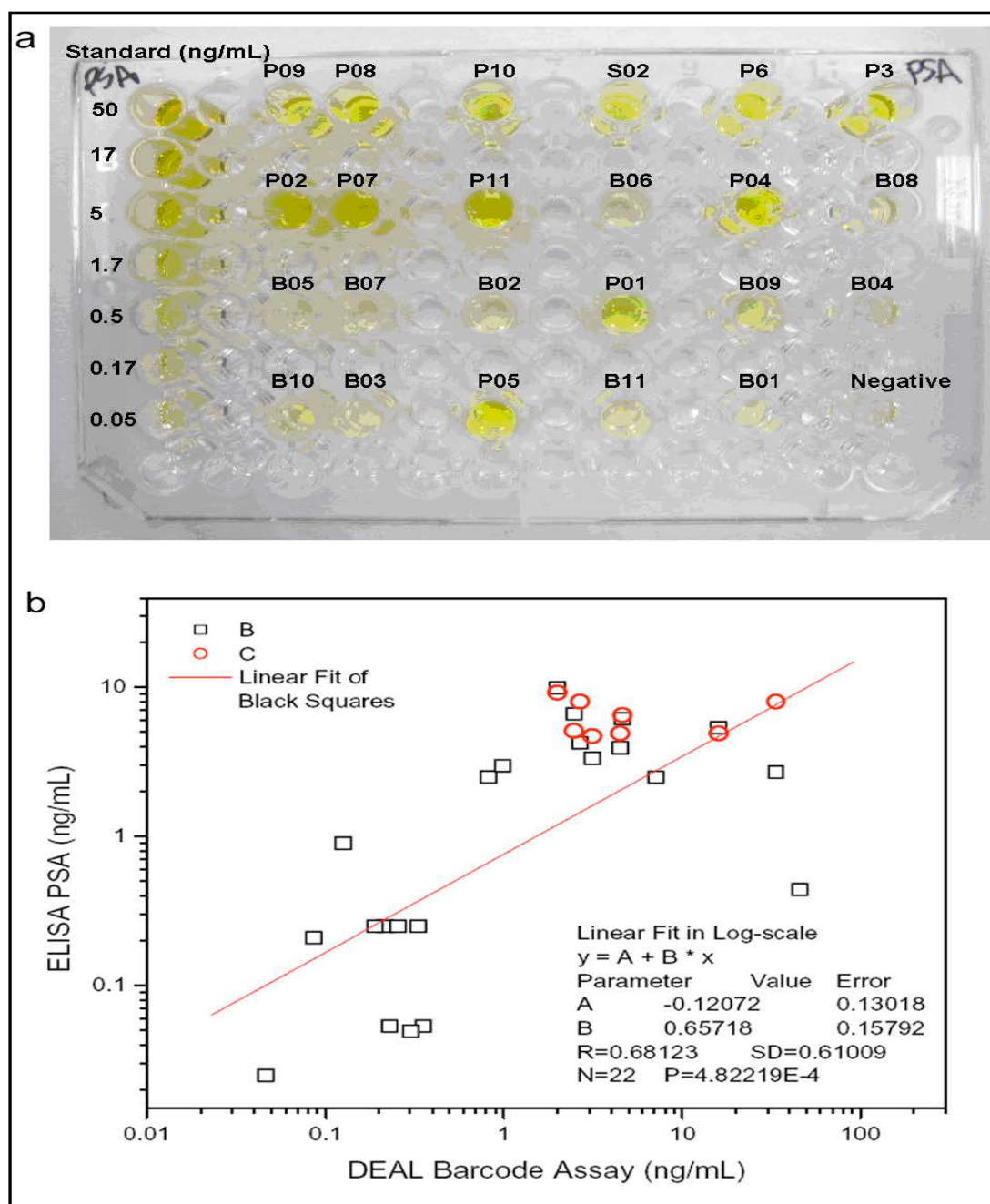


Figure 2.18 Validation of PSA detection using ELISA. (a) ELISA assays performed on PSA standards and 23 as-received serum samples. (b) Correlation between our ELISA and DEAL barcode assays. All DEAL tests were done in multiplex. Levels for most of the prostate cancer patients were available through the serum vendor (Asterand). Thus, for all 22 patient serum samples, we performed enzyme-linked immunosorbent sandwich assays (ELISA) to independently assess the PSA levels.

In comparison with the clinical ELISA results measured on fresh sera, our own ELISA measurements are generally lower in intensity. This likely reflects the influence of long-duration storage (>1year) and freeze/thaw cycles (>1 cycle). The correlation between the ELISA and DEAL barcode assays is analyzed using literature methods.¹² The R-value is comparable to literature measurements that compare multiplexed assays against ELISA standards.¹³ The good agreement between standard ELISA and the DEAL barcode method with respect to PSA measurements provides validation of the IBBC for measuring proteins from human sera.

2.6.6 Cancer Patients: Medically-Relevant Information

Full medical records were provided for all serum samples acquired from Asterand. Selected information is listed in **Table 2.3**. Sample IDs were excluded for privacy protection.

Table 2.3 Cancer Patients: Selected Information

PATIENT	CANCER	GENDER/AGE	RACE	UICC STAGE	GLEASON SCORE	OTHERS
B01	Breast	Female/62	Caucasian	T2N0M0		wine 200mL/day
B02	Breast	Female/79	Caucasian	T4N2M0		
B03	Breast	Female/71	Caucasian	T1cNXM0		1-2 drinks/day
B04	Breast	Female/72	Caucasian	T2NXM0		hypertension
B05	Breast	Female/89	Caucasian	T3N0MX		arthritis
B06	Breast	Female/56	Asian	T1NXM0		
B07	Breast	Female/54	Caucasian	T2N2M0		hypertension, obesity
B08	Breast	Female/55	Caucasian	T2NXM0		1-5 cigs/day, wine 200mL/day
B09	Breast	Female/83	Caucasian	T4N0M0		coronary artery disease, cerebral atherosclerosis
B10	Breast	Female/63	Hispanic	T3N2MX		6-10cigs/day, hyperthyroid, hypertension, osteoarthritis
B11	Breast	Female/63	Caucasian	T1NXM0		arterial hypertension
P01	Prostate	Male / 51	Caucasian	T2cNXM0	4+3=7	
P02	Prostate	Male / 64	Caucasian	T3bN0MX	3+4=7	
P03	Prostate	Male / 47	Caucasian	T2cN0M0	3+3=6	hypertension
P04	Prostate	Male / 55	Caucasian	T2bN0M0	3+3=6	11-20 cigs/day
P05	Prostate	Male / 73	Caucasian	T3aN0MX	4+4=8	hypertension, 11-20 cigs/day
P06	Prostate	Male/64	Caucasian	T3N0M0		chronic bronchitis, 11-20cigs/day
P07	Prostate	Male/60	Caucasian	T3aN0M0	3+4=7	gastroesophageal reflux
P08	Prostate	Male/72	African Am.	T2aN0MX	3+3=6	1-5cigs/day
P09	Prostate	Male/78	Caucasian	T3aN1MX	4+3=7	hypertension, atrial fibrillation
P10	Prostate	Male/66	Caucasian	T2aN0MX	3+3=6	hypertension, 11-20 cigs/day
P11	Prostate	Male / 47	Caucasian	T2cN0M0	3+3=6	hypertension
S01	Unknown					
S02	Unknown					

2.7 Additional References

- 1 Bailey, R., Kwong, G., Radu, C., Witte, O. & Heath, J. DNA-encoded antibody libraries: a unified platform for multiplexed cell sorting and detection of genes and proteins. *J. Am. Chem. Soc* **129**, 1959-1967 (2007).
- 2 Michel, B. *et al.* Printing meets lithography: Soft approaches to high-resolution patterning. *Chimia* **56**, 527-542 (2002).
- 3 Lange, S. A., Benes, V., Kern, D. P., Horber, J. K. H. & Bernard, A. Microcontact printing of DNA molecules. *Analytical Chemistry* **76**, 1641-1647 (2004).
- 4 Delamarche, E., Bernard, A., Schmid, H., Michel, B. & Biebuyck, H. Patterned delivery of immunoglobulins to surfaces using microfluidic networks. *Science* **276**, 779-781 (1997).
- 5 Bernard, A., Michel, B. & Delamarche, E. Micromosaic immunoassays. *Analytical Chemistry* **73**, 8-12 (2001).
- 6 Thuillier, G. & Malek, C. Development of a low cost hybrid Si/PDMS multi-layered pneumatic microvalve. *Microsystem Technologies* **12**, 180-185 (2005).
- 7 Dandy, D., Wu, P. & Grainger, D. Array feature size influences nucleic acid surface capture in DNA microarrays. *Proceedings of the National Academy of Sciences* **104**, 8223 (2007).
- 8 Thorsen, T., Maerkl, S. & Quake, S. Microfluidic large-scale integration. *Science* **298**, 580 (2002).
- 9 Hong, J. & Quake, S. Integrated nanoliter systems. *Nature Biotechnology* **21**, 1179-1183 (2003).
- 10 Heath, J. R. & Davis, M. E. Nanotechnology and cancer. *Annual Review of Medicine* **59**, 405 (2007).
- 11 Pirrung, M. C. How to make a DNA chip. *Angewandte Chemie-International Edition* **41**, 1277 (2002).
- 12 Fredriksson, S. *et al.* Multiplexed protein detection by proximity ligation for cancer biomarker validation. *Nature Methods* **4**, 327-329 (2007).
- 13 Schweitzer, B. *et al.* Multiplexed protein profiling on microarrays by rolling-circle amplification. *Nature Biotechnology* **20**, 359-365 (2002).

3 Plasma Proteome Profiling of Glioblastoma Multiforme: *Characterizing Biomarker Signatures of Disease and Treatment Response*

3.1 Introduction

Glioblastoma multiforme (GBM) [WHO grade IV astrocytoma] is the most common primary brain tumor in adults and the most aggressive form of glioma.¹ Due to its highly proliferative and infiltrative nature, GBM carries the poorest prognosis of any cancer, with a median patient survival of ~12 months, despite major advances in chemotherapy, radiation therapy, and surgery over the last few decades.² Although glioblastoma patients share many disease features in common, the fact that patients can differ tremendously in their response to therapy suggests that the cancer is molecularly heterogeneous. Indeed, it is known that genetically, there are two routes of glioblastoma development. Primary or *de novo* GBM, which is typically characterized by sudden onset of high grade malignancy and an older age of onset, involves EGFR amplification and inactivation of the PTEN gene due to loss of heterogeneity at chromosome 10.^{3,4} Secondary GBM, which is defined by progression from a lower-grade astrocytoma, and often presents at a younger age, initially involves chromosome 17 deletions and inactivation of the p53 gene, followed by a series of other mutations as the tumor undergoes malignant transformation.²⁻⁴ However, even within these two broad categories, patients differ in

the types of subsequent chromosomal alterations and mutations their tumors exhibit, as well as in rates of tumor growth and progression, overall survival, and types of treatments to which they respond.

Gene expression profiling has been instrumental in further elucidating key molecular players involved in GBM growth and progression, as well as the supporting cast of molecules that exhibit less pronounced changes,^{5,6} greatly facilitating the search for candidate GBM biomarkers. However, gene expression profiling provides a window only to RNA expression levels, whereas much of the information processing within the cell occurs at the level of protein network interactions. Often the relationship between RNA and protein expression level is nonlinear⁷ due to additional post-transcriptional controls.^{8,9} Therefore, key drug targets could be differentially expressed at the protein level but not the RNA level.¹⁰ In addition, post-translational processing and modifications can alter the activities of proteins and their locations within the cell.⁸ This information cannot be obtained solely by profiling gene expression.

Proteomic approaches pick up where genomic approaches leave off by allowing one to survey disease-related changes in global protein expression, find correlations between proteins that are similarly differentially expressed, and analyze those changes in the context of known or prospective protein signaling pathways and networks. In particular, antibody-based microarray technology has facilitated the simultaneous high-affinity profiling of numerous proteins from relatively small samples of cell and tissue lysates, culture media,⁷ and bodily fluids, such as blood,¹¹ urine,¹² saliva, tears,^{13,14} and cerebrospinal fluid.¹⁵ The advantages of this form of multiplexed protein detection over other approaches, such as 2D-PAGE and mass spectrometry, are its higher throughput and sensitivity, scalability, ease of use, cost-effectiveness, smaller sample requirements (< 50 μ L), straightforward protein quantitation, and its ability to detect low

abundance plasma proteins without the need for tedious protein fractionation steps.^{7,11,16} As such, this technology represents a promising platform for novel disease-biomarker discovery. In addition, because small quantities of sample are sufficient to obtain enormous amounts of information, new opportunities are afforded for minimally-invasive diagnosis, stratification, and monitoring of cancer patients.¹⁶

Blood is an ideal fluid for minimally-invasive detection of cancer-associated markers.¹⁷ Cancer cells, like any other cell, secrete proteins into the bloodstream that can provide important information about their physiological and pathological state.¹⁷ As well, intracellular proteins and cell-surface receptors are released into the circulation when cancer cells die. Antibody-based microarrays can permit the simultaneous, sensitive detection of many of these circulating factors from very small sample volumes - as little as a fingerprick's volume worth of blood (10-50 μ L).¹⁶ It might nevertheless be expected that plasma detection of brain tumor markers would be challenging because the blood-brain barrier (BBB) greatly limits the free passage of proteins and other molecules between the two compartments.¹⁸ However, the integrity of this barrier becomes greatly compromised at sites of inflammation¹⁹ or neovascularization,²⁰⁻²² which both typically accompany glioblastoma tumors. In addition, glioblastoma tumors secrete soluble factors that disrupt the blood-brain barrier.²³

Unfortunately, for the vast majority of cancers and other diseases, no biomarkers have thus far been discovered with adequate specificity and sensitivity for whole-population screening or disease monitoring. Relatively few serum biomarkers have been FDA-approved for cancer monitoring, and just one – prostate specific antigen (PSA) – is approved for disease screening.²⁴ Likewise for glioblastoma, although gene expression profiling has allowed for the discovery of numerous protein biomarker candidates, none of these proteins on its own has achieved broad

application for routine clinical diagnosis, prognosis, or monitoring of GBM, or for evaluating or predicting therapeutic response.²⁵

However, it has become increasingly recognized that large panels of proteins, in which each component protein has relatively poor disease specificity on its own, can, as a group, provide a highly sensitive and specific molecular signature of disease.^{26,27} A number of studies have demonstrated the ability of antibody-based microarrays to identify protein expression patterns that can discriminate between patients with cancer (of the bladder,²⁸ pancreas,²⁹ prostate,³⁰ or stomach³¹) and normal controls. In theory, a sufficiently informative protein biomarker panel could stratify a given disease into subgroups based on unique molecular phenotypes, much as has been shown in gene expression profile studies.^{5,10} Treatments could then be customized to the tumor's specific set of molecular alterations. This would greatly contrast with the current expensive and time-consuming trial-and-error, watch-and-wait approach of administering a chemotherapeutic, awaiting a response, and then changing the medication if no response is achieved. All the while, the patient's tumor continues to advance in grade and stage.

The routine use of antibody-based microarrays for multiplexed, high-throughput plasma biomarker detection and patient classification requires that these platforms be created using standardized methods that optimize sensitivity, reproducibility, cost, and compatibility with microfluidic chip-based environments. While many approaches for arraying antibodies on slide surfaces have been investigated, DNA-directed antibody immobilization provides a number of unique advantages in this regard. For one thing, as compared to directly spotted antibodies, DNA-tethered antibodies exhibit less denaturation and possess greater orientational freedom, allowing a larger proportion of antibodies to be oriented such that their binding sites are

accessible to cognate antigens.^{32,33} Studies have also shown that this approach offers improved spot homogeneity and reproducibility, and far more economical use of antibody materials.³² Importantly for multiplexed point-of-care diagnostics, DNA-directed immobilization is amenable to microfluidic chip assembly because the antibodies can be arrayed subsequent to bonding of the PDMS stamp with the DNA-spotted slide - a thermal process that would otherwise destroy the antibodies.^{16,34}

The goal of the present study was to determine whether a plasma protein signature could be elucidated that would be able to differentiate patients with glioblastoma (n=46) from healthy controls (n=47) via a simple blood test that uses fingerprick volumes (≤ 50 μ L) of blood (Patient Characteristics shown in **Table 3.1**). We also sought to elucidate a plasma biomarker signature indicative of tumor growth - and, conversely, treatment response - in Avastin-treated GBM patients (Patient Characteristics shown in **Table 3.2**). Our platform consisted of capture-antibody arrays created by DNA-directed assembly within ELISA-like wells. These antibodies were targeted against 35 distinct proteins known to be generally associated with tumor growth, survival, migration, invasion, angiogenesis, and immune-regulation. The platform allowed us to profile even low-abundance analytes (such as cytokines and growth factors) in plasma using microliter-scale sample volumes. We detected a number of proteins that were differentially expressed with high statistical significance ($p < 0.05$), allowing us to use these plasma biomarker signatures to classify patients into the aforementioned experimental and control groups with high sensitivity and specificity.

Table 3.1 GBM Patients vs. Healthy Control Population Characteristics

		GBM		Healthy	
		Patients	Samples	Patients	Samples
<u>Total</u>		46	72	47	47
<u>Age</u>	Median	56	56	39	39
	Mean	56	56	41	41
	Range	30-82	30-82	18-72	18-72
<u>Gender</u>	Male	28	48	22	22
	Female	18	24	18	18
	N/A			7	7
<u>Treatment</u>	No Avastin	11	14	47	47
	Avastin	24	36	0	0
	Avastin 184	2	6	0	0
	Pre-Avastin	7	14	0	0
	N/A	2	2	0	0
<u>Recurrence</u>	New	5	9	N/A	N/A
	1st	14	24	N/A	N/A
	2nd	5	6	N/A	N/A
	3rd	2	7	N/A	N/A
	4th	2	2	N/A	N/A
<u>Blood Draws</u>	1	27		47	
	2	12		0	
	3	4		0	
	4	1		0	
	5	1		0	

Table 3.2 Avastin-Treated GBM Patients: Characteristics of Patient Population with and without Tumor Recurrence

		Tumor Growth		No Tumor Growth	
		Patients	Samples	Patients	Samples
Total		25	52	21	51
Age	Median	56		57	
	Mean	55		54	
	Range	30-82		30-71	
Gender	Male	16	30	10	33
	Female	9	22	11	17
Treatment	Avastin	20	44	19	49
	Avastin 184	5	8	2	2
T2 Levin Score	-2	16	30	0	0
	-1	8	16	0	0
	0	3	4	21	51
	1	1	1	0	0
	2	1	1	0	0
T1C Levin Score	-2	11	22	0	0
	-1	12	16	0	0
	0	6	6	21	51
	1	5	5	0	0
	2	1	3	0	0
Recurrence	New	3	5	4	13
	1st	9	15	7	11
	2nd	3	6	2	2
	3rd	3	5	1	1
	4th	1	1	2	3
	6th	1	1	0	0
	N/A	12	18	5	5
Blood Draws	1	13		9	
	2	3		5	
	3	6		2	
	4	1		3	
	5	1		1	
	6	1		0	
	7	0		0	
	8	0		0	
	9	0		1	

3.2 Experimental Methods

3.2.1 DNA-Encoded Antibody Libraries (DEAL) Technique

The antibody assembly platform used here is based on the DNA-encoded antibody library (DEAL) method.¹ The DEAL assays were performed as previously described (**Section 2.5.1**) except that instead of using microchannel-guided flow patterning, ssDNA oligomers complementary to the ssDNA-CA conjugates (100 μ M in a 50% DMSO/water mixture) were spotted onto a poly-L-lysine coated glass slide (150 μ m spots spaced 300 μ m center-to-center) using an array spotter (VersArray Chip Writer Pro, BioRad). Each spot also contained 10 μ M of oligo M as a spot loading control. DNA oligomer sequences were again chosen with appropriate melting temperatures to optimize 37°C hybridization to complementary strands while minimizing cross-hybridization (<5% in fluorescence signal).

3.2.2 Antibody Array Platform

Our platform consists of ELISA-like wells assembled by bonding a PDMS slab with pre-cut square holes to a poly-lysine-coated glass substrate onto which 6x6 oligonucleotide arrays have been pre-spotted. Thirty-five distinct DNA-addressed antibodies are directed to their complementary spots during the assay. The assay wells accommodate up to 200 μ L of sample, but in fact, only about 20-50 μ L are needed to obtain a reasonable signal-to-noise. We used 50 μ L of plasma for all of our assays.

3.2.3 Multiplexed Assays on Patient Plasma

Each microarray well (12 wells/slide) was first blocked with 200 μ L of blocking buffer - 3% wt/vol bovine serum albumin fraction V (Sigma) in 150 mM 1X PBS without calcium/magnesium salts (Irvine Scientific) – for 1 hour in a 37°C incubator. The wells were then aspirated, and 50 μ L of a cocktail containing 35 different DNA-antibody conjugates (50 nM each) in blocking buffer were pipetted into the wells to transform the DNA arrays into capture-antibody arrays (**Figure 3.1**). After incubation at 37°C for 1 hour, the wells were aspirated, and then rinsed with blocking buffer 4-5 times to remove excess unbound conjugate. At this step, the wells were ready for the blood test. Fifty-microliter undiluted plasma samples were added to each well and allowed to incubate for 1 hour at 37°C. The samples were then aspirated and each well was again rinsed 4-5 times with blocking buffer. Next, a cocktail containing the 35 biotinylated detection antibodies (50 nM each) in blocking buffer was added to each well (50 μ L) and was allowed to incubate for 1 hour at 37°C. The wells were aspirated and rinsed 4-5 times with blocking buffer, followed by incubation of a solution containing 50 nM Streptavidin-Cy5 (eBioscience) and 50 nM M'-Cy3 for 1 hour at 37°C. The wells were aspirated, and rinsed 4-5 times with blocking buffer. The PDMS well template was then peeled off the slide within a blocking buffer bath, and the slide was allowed to incubate in the bath for 1 minute at room temperature. The slide was then immersed in 150 mM 1X PBS, ½X PBS, and twice in deionized water in separate 50-mL falcon tubes for 1 minute, 10 seconds, and 2 seconds, respectively. The slide was then spun dry and scanned by a fluorescence microarray scanner.

3.2.4 Plasma Collection and Processing

Blood samples were collected by standard phlebotomy techniques in 10-mL blood collection tubes containing ACD-A anticoagulant (BD Vacutainer yellow-top glass tubes). The samples were centrifuged at $1500 \times g$ for 15 minutes, and the plasma was collected and subdivided into 200 μ L aliquots. Plasma samples were frozen at -80°C within 2 hours of collection to minimize degradation of plasma proteins by proteases. Each aliquot was thawed just once as needed.

3.2.5 Data Processing and Statistics

Post-assay, all array slides were scanned using a two-color laser fluorescence microarray scanner (GenePix 4200A Professional, Axon Instruments) at the same instrument settings. For the 635 nm and 532 nm excitation wavelengths, the laser powers were 70% and 50%, respectively, and the optical gains were 550 and 500, respectively. Spot intensities were quantified with the software program *GenePix Pro 6.0* using the fixed circle method. For each sample, the local background was subtracted from each spot, and the average and standard deviation were taken for each of the 35 sets of six repeated spots. A semi-global normalization method was used for chip-to-chip normalization. Briefly, the coefficient of variation (CV) was calculated for each analyte over all samples and ranked. The 15% of analytes (5 analytes) with the lowest CV-values were used to calculate the normalization factor $N_i = S_i/\mu$, where S_i is the sum of the signal intensities of the 5 analytes for each sample, and μ is the average of S_i from all samples. The dataset generated from each sample was then divided by the normalization factor N_i . Universally, all datasets contained at least 4 analytes that had comparable intensities to negative controls run in separate experiments. Therefore, the net intra-assay intensities were

calculated by subtracting each background-corrected analyte intensity by the mean intensity of the 4 lowest-intensity analytes. Unsupervised two-way average linkage hierarchical clustering (*Cluster 3.0*) was then performed on an entire patient cohort data set, and the resulting heat map and dendrogram were viewed using Java *TreeView*. The statistical significance (both Mann-Whitney and t-test *p*-values) of differential protein expression between experimental and control groups was analyzed using the *AnalyseIt* add-in for Microsoft Excel. This add-in was also used to generate box plots for each measured analyte across each study group.

3.2.6 Classification of Patients

Two-by-two contingency tables and diagnostic parameters - sensitivity, specificity, negative predictive value (NPV), and positive predictive value (PPV) - were calculated by repeated random sub-sampling cross-validation. An Excel macro developed in-house was used to randomly assign 10 patients to a test set, leaving the remainder of patients as the training set. Unsupervised two-way average linkage hierarchical clustering (*Cluster 3.0*) was then performed on the entire patient cohort dataset (now containing 10 unknowns) and the resulting heat map and dendrogram were viewed using Java *Treeview*. The ten unknown patients were then manually classified as belonging to the experimental (Group A) or control group (Group B) based on the following decision rules (x = the fraction of members within the unknown's cluster that belong to the same group):

1. The minimum number of clusters incorporating the unknown and at least 5 other members is analyzed. If all members of this cluster fall into the same group ($x = 1$), the unknown is classified as a member of that group with high confidence (this is considered

a homogeneous zone). If $x > 0.75$, the unknown is still considered to be part of the majority group (with average confidence) but the cluster is no longer considered a homogeneous zone. If $x < 0.75$, then...

2. The minimum number of clusters incorporating the unknown and at least 8 other members is analyzed. If now $x > 0.75$, the unknown is considered to be part of the majority group. If $0.5 < x < 0.75$, the unknown is still considered to be part of the majority group, but with low confidence. In this case...
3. The minimum number of clusters incorporating the unknown and at least 14 other members is analyzed using the same decision rules as in 2.
4. If $x \sim 0.5$ after step 3, then the unknown remains unclassified and is removed from the analysis. Alternatively, an $x \sim 0.5$ is sufficient to remove the unknown sample from the analysis even if the unknown is grouped within a smaller cluster if the members of that cluster are closely correlated, yet far less correlated with the nearest neighboring cluster.
5. If in step 1, the unknown is part of a cluster containing 4 or fewer members that are all highly correlated with each other relative to the nearest neighboring cluster, the unknown is assigned to the majority group with low confidence if $0.5 < x < 0.66$, average confidence if $0.66 < x < 0.75$ and high confidence if $x = 1$.
6. If two or more unknowns are nearest neighbors, these unknowns remain unclassified and are removed from the analysis.

This random sub-sampling was then repeated 10 times with replacement (100 unknown events), such that some patients may have been randomly assigned to a test sample more than once, while others not at all. An Excel macro developed in-house then compared the predicted

and actual classifications, and output the total number of true-positives, false-positives, true-negatives, and false-negatives in a 2x2 contingency table, as well as the sensitivity, specificity, NPV, and PPV of the diagnostic evaluation. This constituted the full diagnostic evaluation for a dataset. For the two patient cohorts examined in this study, diagnostic evaluation was also performed on trimmed datasets consisting of subsets of n proteins (from the initial 35-protein panel) that exhibited the most statistically significant differential expression between experimental and control groups (where $n = 3, 6, 9, 12, 16, 20, 25$). For each dataset, points were plotted in ROC space (sensitivity vs. 1-specificity) to assess the predictive power of the test.

3.3 Results

3.3.1 Evaluation of DNA-Directed Antibody Microarrays

Preliminary experiments were run in advance to validate a set of 35 orthogonal oligos that exhibited minimal cross-hybridization (<5%). In addition, the full panel of DNA-conjugated antibodies was validated with a set of cognate recombinants to ensure that there was minimal cross-talk between each recombinant and non-cognate spots. Each DNA spot was co-loaded with reference DNA (at 10% of the spot's total DNA loading), which, once hybridized with a dye-conjugated complement, served as a DNA-loading control. For each oligo, the spot loading was highly consistent both across an entire slide as well as between slides.

The fluorescent readouts from all plasma samples assayed on the 35-plex antibody array platform were analyzed for spot homogeneity, reproducibility, and signal-to-noise. A representative image of the fluorescent readout from a single assay well is shown in **Figure 3.1**. Each well contained a total of six repeats of 6x6 spot arrays (35 antibodies + 1 green Cy3-conjugated reference oligonucleotide). The spots were circular, well-defined, and radially homogeneous. There was very little intra-assay variation in the intensities of each set of repeats. In addition, spot intensities tended to be highly consistent even between duplicate assays run on separate slides.

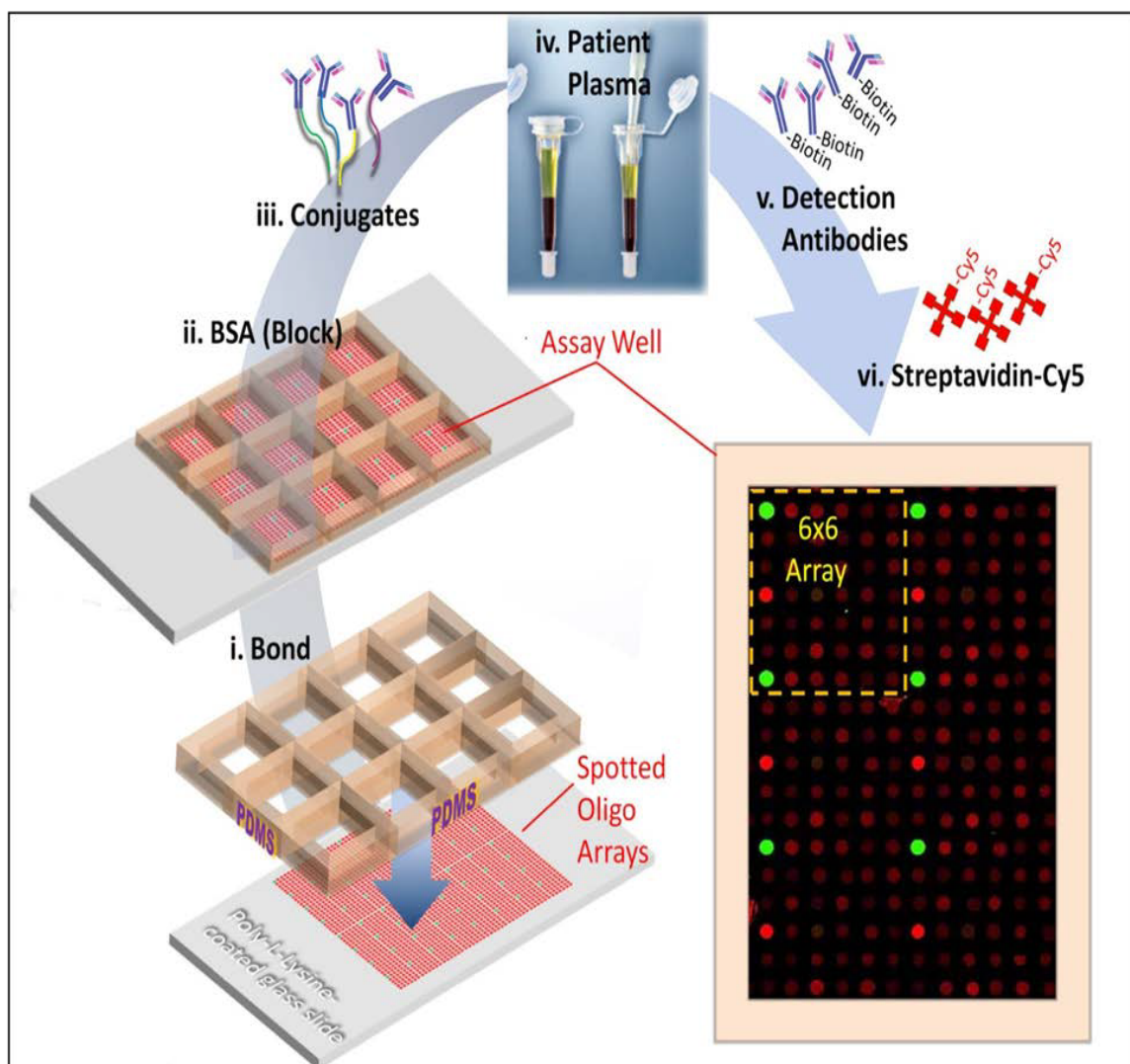


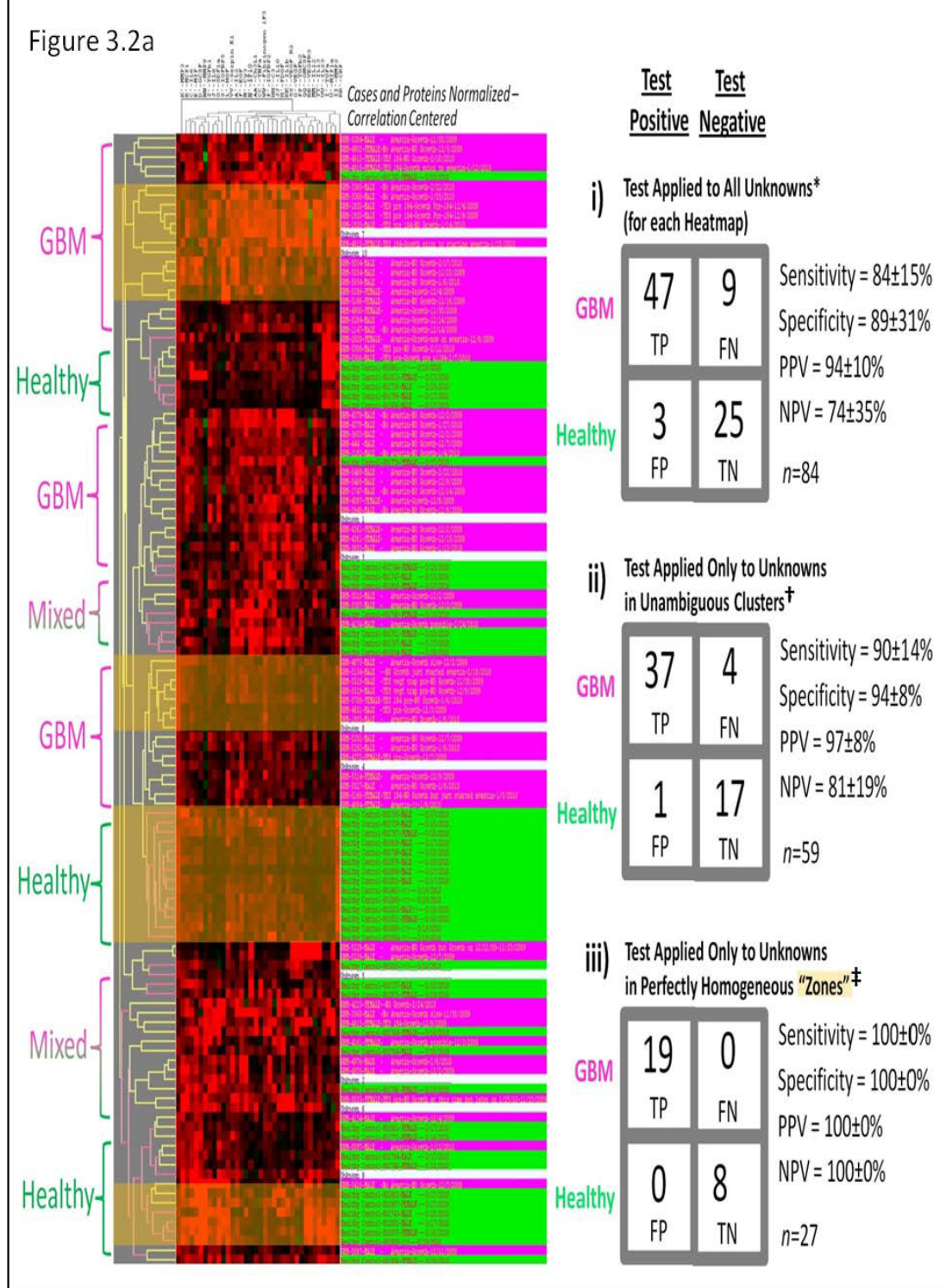
Figure 3.1 Assay platform and methodology. Polylysine-coated slides were spotted with 36 distinct oligos in 6x6 array repeats. A PDMS slab containing square holes was set on top of the spotted substrates, forming ELISA-like assay wells. Assays were performed by: i. blocking the wells with BSA, ii. incubating with conjugates to transform the DNA microarray into an antibody microarray, iii. adding a different patient plasma sample to each well, iv. incubating with biotinylated detection antibodies, and v. adding Streptavidin-Cy5 (Red Spots) and M'-Cy3 (Green Reference Spots). Thorough rinses were performed between each of these steps. Each well contained six full repeats of the 6x6 antibody arrays. The platform was used to detect 35 distinct proteins from 40-50 μ L of plasma. Readout was performed with a fluorescence scanner.

3.3.2 Classification of GBM Patients versus Healthy Controls

We compared plasma samples from 46 GBM patients (72 samples total - for some patients, plasma samples from multiple collection dates were available) with those of 47 healthy controls with respect to the plasma levels of 35 different proteins known to be generally associated with tumor growth, survival, invasion, migration, and immuno-regulation. Two-way average-linkage hierarchical clustering allowed these two groups to be discriminated with a sensitivity and specificity of $84 \pm 15\%$ and $89 \pm 31\%$, respectively (**Figure 3.2a**). The heat map is divided into numerous islands of GBM patient, healthy control, and mixed population clusters without a clean separation between the two groups. We then sought to determine whether the diagnostic accuracy could be improved by removing those test samples from diagnostic evaluation that did not fall into highly biased clusters (i.e. $> 70\%$ of the cluster members belong to the same group). Within the subpopulation of test samples that fell into highly discriminatory clusters, the sensitivity and specificity improved to $90 \pm 14\%$ and $94 \pm 8\%$, respectively, albeit with a diagnosable population size that was 70% of the original. Among test samples that clustered entirely with members of a single group (“homogeneous zones”), the sensitivity and specificity both approached 100%. Thirty percent of samples fell into one of these homogeneous zones, allowing that subpopulation to be diagnosed with near-perfect accuracy. (For a more detailed discussion of hierarchical clustering and examples of highly biased clusters and homogeneous zones, see **Sections 4.2 and 4.3.**)

We then repeated the cluster analysis with a trimmed panel that included only the nine proteins with the most statistically significant (Mann-Whitney and t-test p -values < 0.05) differential expression (**Figure 3.2b**). These included: MMP3, PDGF, IP10, IGFBP2, VEGF, IL13, GM-CSF, MMP9, and CRP. The resultant heat map shows far improved classification of

Figure 3.2a

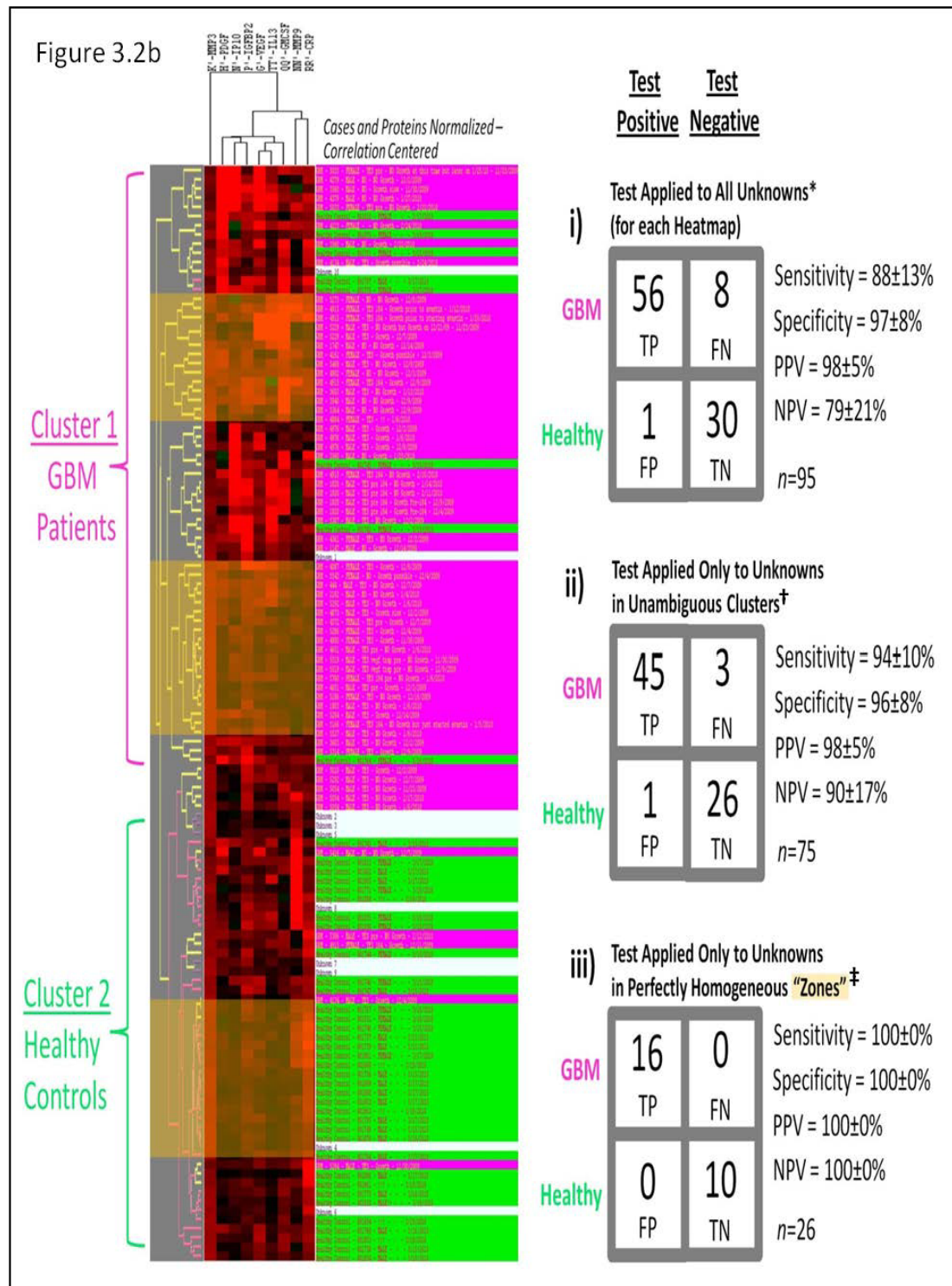


*These 84 (unknown) events constituted 64 (of 93) distinct patients and 72 (of 119) independent samples.

†These 59 (unknown) events constituted 45 (of 93) distinct patients and 50 (of 119) independent samples.

‡These 27 (unknown) events constituted 20 (of 93) distinct patients and 22 (of 119) independent samples.

Note: The number of events in each square of a 2x2 contingency table represents the sum of outcomes of 10 tests, with each test consisting of 10 blindly and randomly generated unknowns from the list of 119 independent samples. No sample is assigned to be an unknown more than once per test. However, the number of events exceeds the number of independent samples because over the course of 10 tests, some samples may be randomly assigned as unknowns multiple times.



*These 95 (unknown) events constituted 58 (of 93) distinct patients and 71 (of 119) independent samples.

†These 75 (unknown) events constituted 46 (of 93) distinct patients and 54 (of 119) independent samples.

‡These 26 (unknown) events constituted 20 (of 93) distinct patients and 22 (of 119) independent samples.

Note: The number of events in each square of a 2x2 contingency table represents the sum of outcomes of 10 tests, with each test consisting of 10 blindly and randomly generated unknowns from the list of 119 independent samples. No sample is assigned to be an unknown more than once per test. However, the number of events exceeds the number of independent samples because over the course of 10 tests, some samples may be randomly assigned as unknowns multiple times.

Figure 3.2 Classification of GBM patients and healthy controls. (a) Average linkage hierarchical clustering (unsupervised) was performed on 35-protein datasets from each of 72 GBM patient plasma samples and 47 healthy control samples. A computer program was used to randomly assign patients to trial and test sets (multiple times), and the disease status (GBM vs. healthy) of each test set member (unknown) was predicted based on the status of nearest neighbors in its cluster. 2x2 contingency tables were generated and relevant statistical parameters were calculated for diagnostic tests that evaluated: i) all unknowns in the heat map; ii) only unknowns in clusters where a sizeable majority of members - including the nearest neighbor - shared the same status; iii) only unknowns in completely homogeneous clusters where *all* members shared the same status - so-called homogeneous “zones”. (b) The heat map in a was “trimmed” by performing average-linkage hierarchical clustering (supervised) on the nine proteins that exhibited the most significant differences (lowest *p*-values) between GBM patients and healthy controls. Note the significant improvement in stratification of patients into just two main relatively homogenous groups – a “GBM Patient” cluster and a “Healthy Control” cluster. Compare with the more numerous small clusters and “mixed” groups in a. TP = True Positive, TN = True Negative, FP = False Positive, FN = False Negative, PPV = Positive Predictive Value, NPV = Negative Predictive Value. Label Colors: **magenta = growth**; **green = no growth**; **blue = growth possible or slow**. **Highlighted areas on heat map are “zones”**.

GBM patients and healthy controls into two separate clusters, with few misclassifications in each cluster. By using this trimmed protein panel, the sensitivity and specificity achieved were $88 \pm 13\%$ and $97 \pm 8\%$, respectively. As before, those samples (20% of the sample population) that did not decisively cluster with a particular group were removed from diagnostic evaluation, and the sensitivity and specificity among the resulting diagnosable population improved to $94 \pm 10\%$ and $96 \pm 8\%$, respectively. Again, both sensitivity and specificity approached 100% among test samples that clustered within perfectly homogeneous zones.

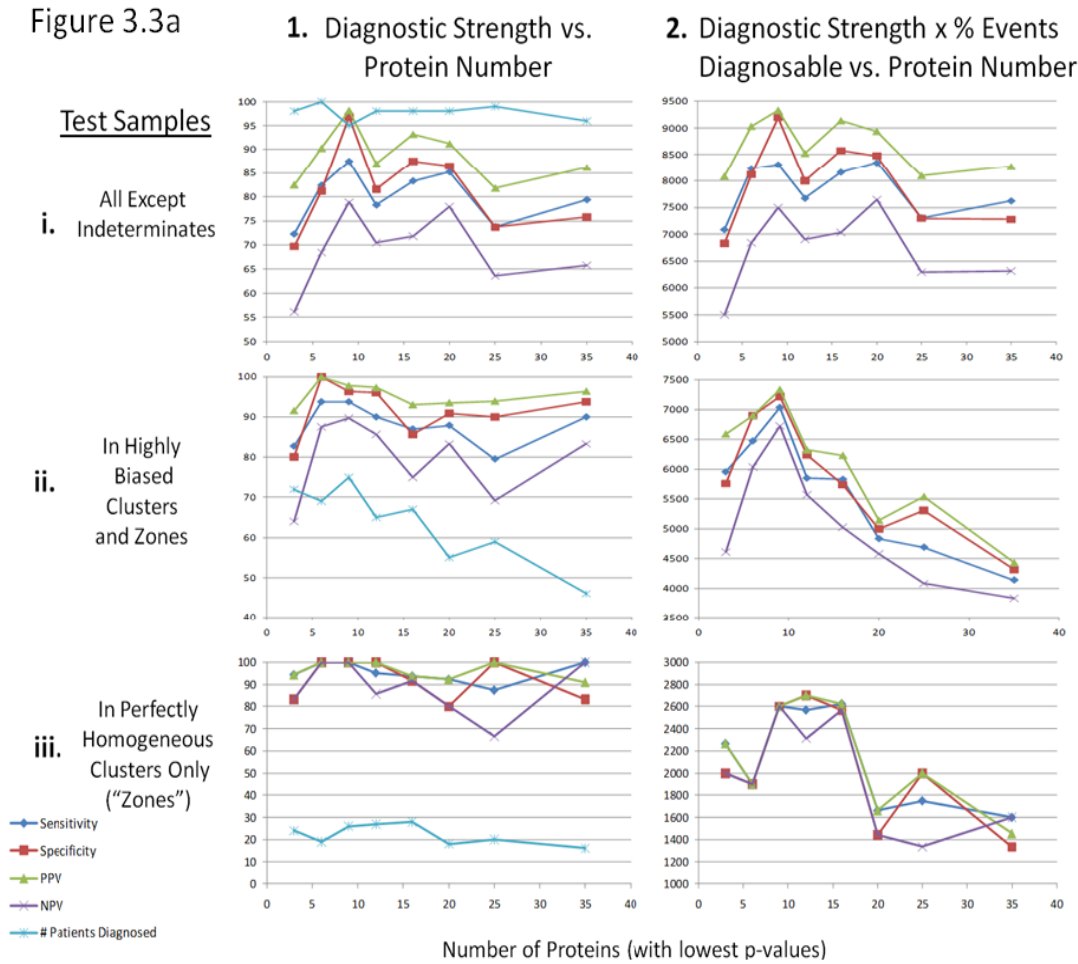
3.3.3 Diagnostic Strength as a Function of Protein Panel Size

The cluster analysis was repeated for *n*-protein subsets of the original 35 protein panel, where $n = 3, 6, 9, 12, 16, 20, 25$, and 35 of the most statistically significant discriminators of GBM and health status. Diagnostic test sensitivity, specificity, and positive and negative predictive values were calculated for each of these subsets. Those test samples that did not decisively cluster with a particular group were removed from the evaluation. As can be seen in

Figure 3.3a - Column 1, the sensitivity and specificity remain about level as one trims the panel from 35 to 20 proteins. Both parameters increase as the panel is trimmed from 16 proteins onwards, with a peak at 6 proteins, followed by a sharp drop as the panel size is reduced further. On the other hand, the percentage of samples evaluable increases steadily as one trims the panel from 35 proteins down to 9 proteins and then tapers off. Since the strength of a diagnostic test lies not only in its diagnostic accuracy but also in the percentage of the population it can evaluate, we designated an artificial parameter S to represent the product of a diagnostic value and the percentage of patients diagnosable for each n -protein subset. As can be seen in **Figure 3.3 – Column 2**, this parameter increases steadily as the protein panel size is reduced, peaking at 9 proteins and then falling off sharply. Therefore, the 9-protein subset appears to optimize test performance by achieving a high diagnostic accuracy while still maintaining the ability to diagnose a large fraction of the sample population.

Figure 3.3 Diagnostic strength vs. protein number for “GBM vs. Healthy Control” cohort. (a) Diagnostic Parameters (sensitivity, specificity, NPV, PPV) were plotted for varying subsets of n proteins that exhibited the most significant plasma concentration differences (lowest p -values) between GBM patients and healthy controls, where $n=3, 6, 9, 12, 16, 20, 25, 35$. A diagnostic test was conducted and a 2x2 contingency table was formulated for each n -set. Column 1) Diagnostic parameters and percentage of events diagnosable are plotted against n . Column 2) The product of each diagnostic parameter and the percentage of events diagnosable is plotted against n . Diagnostic tests evaluated: Row i) all unknowns in the hierarchical clusterings; Row ii) unknowns in clusters where a sizeable majority of members - including the nearest neighbor - shared the same status; Row iii) unknowns in completely homogeneous clusters where *all* members shared the same status - so-called “zones”. Most measures of diagnostic strength tended to peak at around the 9-protein set. (b) Relevant diagnostic parameters for each n -set were plotted in ROC space. Cases i, ii, and iii are the same as before. Note the improvement in predictive power (distance of points from 45° line) from case i \rightarrow ii \rightarrow iii. Also note the high predictive power of the 9-set in i. and of both the 6- and 9-sets in ii. and iii. The coordinate (0,100) represents perfect classification.

Figure 3.3a



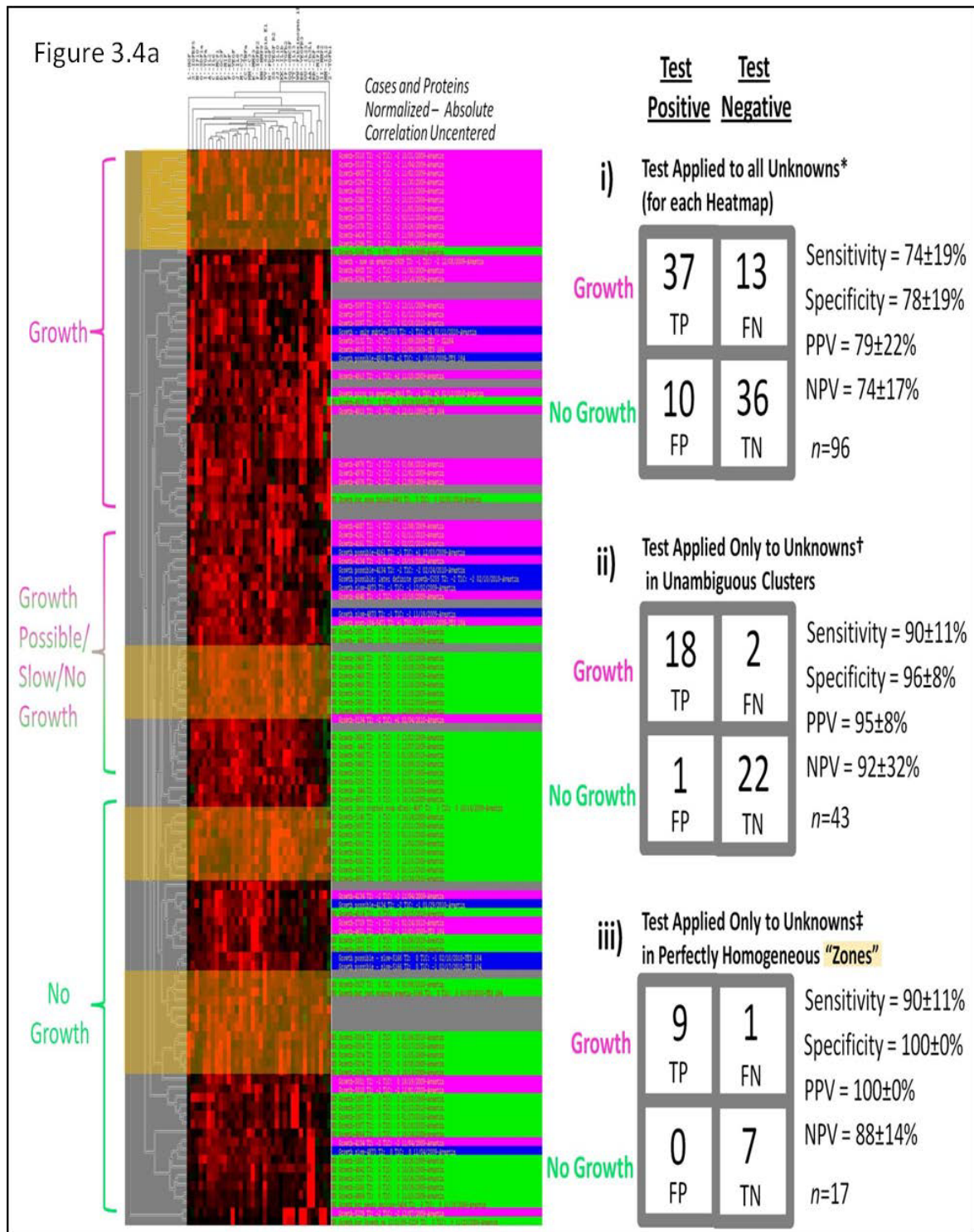
The predictive power of each n -protein subset for classifying GBM patients and healthy controls can also be evaluated by plotting the true positive rate (sensitivity) vs. the false positive rate (1-specificity) for each subset. As observed in **Figure 3.3b**, the 6- and 9-protein subsets yield points in ROC (Receiver-Operating Characteristic) space that are furthest in perpendicular distance from the 45° line, suggesting that this number of differentially expressed proteins maximizes the predictive power. Points for all protein-panel subsets move further from the diagonal line as the sample population is trimmed to include: only those samples in clusters that are highly biased – i.e. the great majority (>70%) of members belong to either the GBM patient or healthy control groups (**Figure 3.3b, Row ii**); or, only those samples in clusters that are completely biased - all members belong to one of the two groups (**Figure 3.3b, Row iii**). Perfect classification was achievable in both 6- and 9-protein subsets when analyzing only the subset of test samples that were located within perfectly homogeneous clusters. As a whole, the data in **Figure 3** shows that by performing cluster analysis on patient plasma samples assayed for the 6 or 9 proteins most significantly differentially expressed, a very high degree of predictive power can be achieved among samples in highly biased clusters. In addition, the 9-protein set optimized the predictive power and the number of patients diagnosable. Furthermore, a subgroup of these patients who fell into perfectly homogeneous clusters could be diagnosed with near certainty.

3.3.4 GBM Patients on Avastin – Classification of Tumor Growth vs. No Growth

We then assayed plasma samples from GBM patients treated with the chemotherapeutic drug Avastin (Bevacizumab) with respect to the same 35-protein panel as before. Specifically, we compared 52 samples from (25) patients who exhibited tumor growth (according to MRI imaging) with 51 samples from (21) patients who exhibited no tumor growth since their last

evaluation. Two-way average-linkage hierarchical clustering allowed these two groups to be discriminated with a sensitivity and specificity of $74 \pm 10\%$ and $78 \pm 19\%$, respectively (**Figure 3.4a**). When only patient samples within highly biased clusters were analyzed (45% of the total sample population), the sensitivity and specificity improved to $90 \pm 11\%$ and $96 \pm 8\%$, respectively. The sensitivity remained the same but the specificity increased to 100% when test samples only in perfectly homogeneous clusters were analyzed (20% of sample population).

The heat map is divided into 3 main sections consisting of samples from: 1. patients whose tumors have grown since their last evaluation (recurrence); 2. patients whose tumors have remained stable since their last evaluation (no recurrence); and 3. a mixed population of patients exhibiting either possible growth, slow growth, or no growth. The patient samples were then clustered with respect to the 4 proteins that were differentially expressed with the highest statistical significance (i.e. both Mann-Whitney and Student's t-test $p < 0.05$). **Figure 3.4b** shows clustering of patient samples into 3 main groups: 1. tumor growth; 2. no tumor growth; and 3. mixed population: consisting of both patients with and without tumor growth. Particularly notable is that serum levels of HGF and TGF β 1 appear to be highly upregulated in the tumor growth group as compared with the no growth group. The cytokines MIP1 α and IL12 (not shown in the heat map) are also highly upregulated in the growth group. In addition, VEGFR2 appears to be highly down-regulated, while IL2 is only somewhat downregulated, in the growth group compared with the no growth group. The alterations in cytokine levels observed in the plasma of patients with growing tumors with respect to non-growing tumors may not necessarily be attributable to changes in tumor production and secretion of these cytokines. Rather, they may actually reflect changes in systemic responses to the growing tumor, such as inflammatory-associated or other immune-mediated responses.



*These 96 (unknown) events constituted 36 (of 46) distinct patients and 75 (of 103) independent samples.

†These 43 (unknown) events constituted 21 (of 46) distinct patients and 34 (of 103) independent samples.

‡These 17 (unknown) events constituted 12 (of 46) distinct patients and 16 (of 103) independent samples.

Note: The number of events in each square of a 2x2 contingency table represents the sum of outcomes of 10 tests, with each test consisting of 10 blindly and randomly generated unknowns from the list of 103 independent samples. No sample is assigned to be an unknown more than once per test. However, the number of events exceeds the number of independent samples because over the course of 10 tests, some samples may be randomly assigned as unknowns multiple times.

Figure 3.4 Classification of GBM patients on Avastin – tumor growth vs. no growth.

(a) Average linkage hierarchical clustering (unsupervised) was performed on 35-protein datasets from each of 122 plasma samples derived from GBM patients treated with Avastin. A computer program was used to randomly assign patients to trial and test sets (multiple times), and the tumor growth status of each test set member (unknown) was predicted based on the status of nearest neighbors in its cluster. 2x2 contingency tables were generated and relevant statistical parameters were calculated for diagnostic tests that evaluated: i) all unknowns in the heat map; ii) only unknowns in clusters where a sizeable majority of members - including the nearest neighbor - shared the same status; iii) only unknowns in completely homogeneous clusters where *all* members shared the same status - so-called “zones”. TP = True Positive, TN = True Negative, FP = False Positive, FN = False Negative, PPV = Positive Predictive Value, NPV = Negative Predictive Value. **Highlighted areas on heat map are “zones”.** (b) The heat map in a was “trimmed” by performing average-linkage hierarchical clustering (supervised) on the four proteins that exhibited the most significant differences (lowest *p*-values) when comparing patients with tumor growth to those with no growth. Patients with high HGF and/or high TGFβ1 levels tended to exhibit tumor growth, while patients with high VEGFR2 levels and/or low TGFβ1 tended to exhibit no tumor growth. Label Colors: **magenta = growth**; **green = no growth**; **blue = growth possible or slow**.

Closer examination of TGFβ1 and HGF revealed that both were differentially regulated with high statistical significance ($p=0.0078$ and $p=0.0055$, respectively) when comparing Avastin-treated GBM patients exhibiting tumor growth with those exhibiting no growth. Therefore, we decided to assess the classification accuracy of each of these markers on its own as well as both together. Intriguingly, plasma TGFβ1 was highly upregulated in the context of tumor growth (2 orders-of-magnitude fold-change in fluorescent intensity), with very little plasma expression in the absence of growth (as shown in **Figure 3.5**). As a result, TGFβ1 alone proved to be a highly sensitive biomarker, correctly classifying 86% of patients who were known to have growing tumors (sensitivity = 86%). However, it was not specific in that it did not accurately classify patients known to have stable tumors (specificity = 53%). In addition, the positive and negative predictive values for TGFβ1 were modest at 70% and 75%, respectively. Although the statistical significance of HGF differential expression was slightly better than that of TGFβ1, its accuracy as a biomarker was offset by the fact that its fold-change between the two groups was not nearly as high. Its specificity (65%) was higher than that of TGFβ1, but its

sensitivity was lower (73%). The positive and negative predictive values for HGF were 73% and 65%, respectively. Encouragingly, when both biomarkers were used together, the resulting diagnostic test exhibited the best predictive accuracies of the two tests, attaining the higher sensitivity level of TGF β 1 (86%) and the higher specificity of HGF (65%). In addition, the PPV and NPV for the diagnostic pair (76% and 79%, respectively) were higher than those of either biomarker alone (**Figure 3.5**).

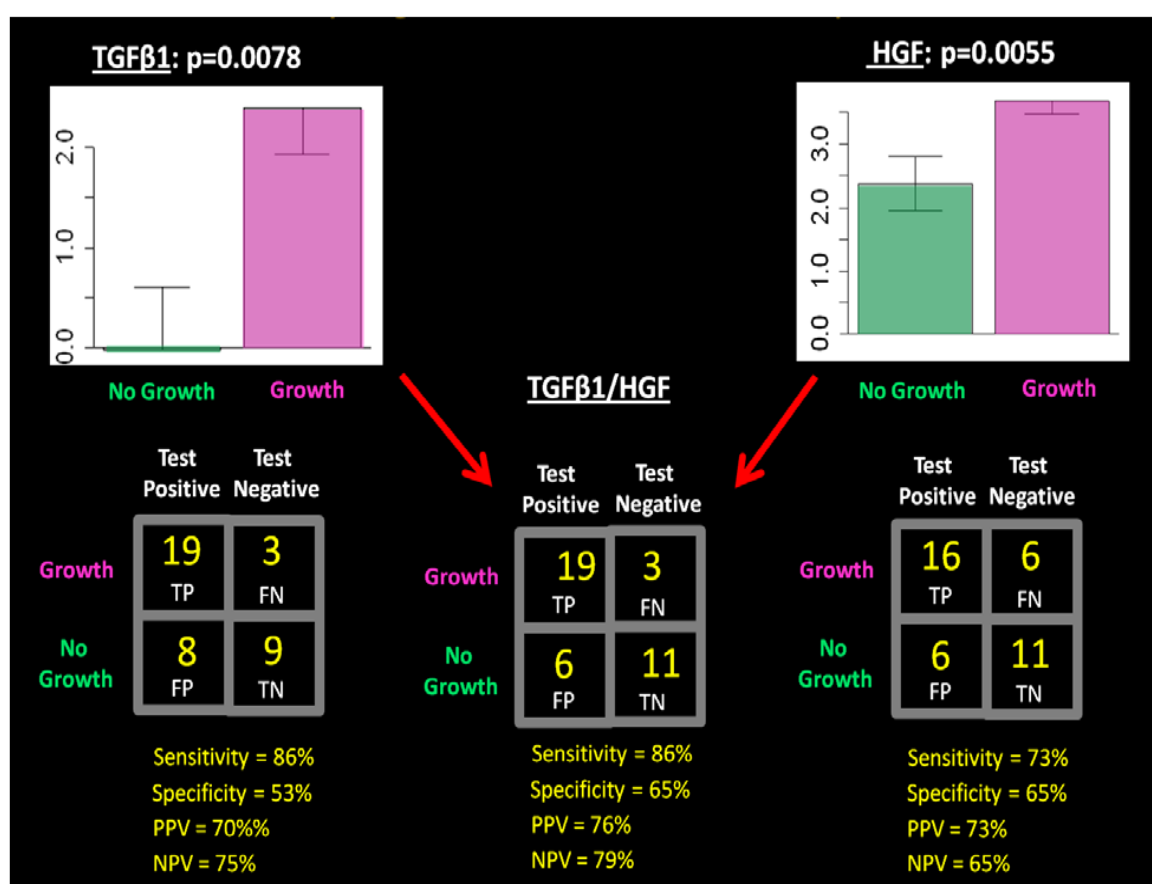


Figure 3.5 Diagnostic Accuracy of the Candidate Biomarkers, TGF β 1 and HGF, separately and together. Note that, as a biomarker pair, the best of the two individual markers' sensitivities and specificities are matched, and the PPV and NPV also improve.

3.4 Discussion

While antibody microarrays have been used in the past to profile cancers of the lung, liver, ovary, prostate, pancreas, colon, and bladder, this is the first study demonstrating their use for plasma profiling of glioblastoma. In this study, we have shown that by interrogating a relatively large panel of 35 plasma proteins, biomarker signatures could be straightforwardly elucidated that could differentiate GBM patients from healthy controls, and that could classify GBM patients treated with Avastin based on whether they were responsive to therapy. Furthermore, none of these proteins on its own has been shown to be an effective biomarker for cancer diagnosis or for treatment response. Therefore, this study also reaffirmed past observations that large panels of proteins can serve as highly sensitive and specific biomarker signatures of disease, even when each component protein is a poor disease-marker on its own.

The study also showed that the predictive power of patient classification by hierarchical clustering depended on the number of differentially expressed proteins analyzed. Tests that included those proteins that were statistically significantly differentially expressed ($p < 0.05$) had greater predictive power compared with tests that additionally contained large numbers of non-discriminatory proteins or compared with tests that contained too few discriminatory proteins. The implication for future biomarker signature discovery from large numbers of proteins is that variously sized subsets of differentially expressed proteins should be evaluated to find the optimally-sized set for maximal predictive power.

In this study, the accuracy of test sample classification was also dependent on the fraction of members within the test sample's cluster that belonged to the same group (experimental vs. control). Therefore, prediction accuracy improved when evaluating only those test samples in highly biased clusters, and approached 100% within completely homogeneous clusters. Of

course, the fraction of diagnosable patients decreased as the tolerance for cluster heterogeneity was reduced. In light of this, an optimal tolerance was chosen that maximized diagnostic accuracy while minimizing the fraction of patients left out of the diagnostic evaluation. In this study, we evaluated only three tolerance settings, corresponding to exclusion of test samples in clusters that were: i. perfectly heterogeneous, ii. <70% homogeneous, and iii. <100% homogeneous. However, the study could potentially be reanalyzed with a larger set of tolerances to find an even better optimum. Alternatively, all patients could have been included in the diagnostic evaluation, but with the appreciation that the diagnosis of patients in certain clusters would be more accurate than in others. In theory, an accuracy score or confidence level could be calculated for grouping within any cluster. Subsequently, only those patients whose diagnoses have a predicted accuracy greater than, say, 90% would be triaged for therapy, whereas all others would have to undergo further tests to ascertain their diagnosis. Based on our results, the diagnostic accuracy would be expected to increase with the homogeneity of a test sample's cluster, with even higher accuracies likely attainable in homogeneous clusters of larger size.

Though it might have been anticipated that plasma protein detection of brain tumors would be difficult due to the blood-brain barrier, in fact, we were able to detect differential expression of a number of factors. Many of these have been associated with systemic cancers or have been previously shown to be differentially expressed in culture media from GBM cell lines and primary cells, in the CSF fluid of GBM patients, or even in patient sera. For example, VEGF, a powerful mediator of endothelial cell proliferation and angiogenesis generally, which was found to be upregulated in GBM patients in this study, has also been shown previously to be highly secreted from GBM cell lines and primary tumors, and to be expressed in the CSF fluid of glioblastoma patients.³⁵ VEGF is typically associated with advanced tumor stage and poor

prognosis in a variety of cancers.³⁵ While no difference in serum expression of VEGF in the context of GBM was found by some,³⁶ our results corroborate reports that have shown upregulated serum expression of VEGF.³⁷ VEGF is known to promote microvascular permeability,³⁵ which likely plays a role in the enhanced BBB leakiness at sites of characteristically highly neovascularized GBM tumors, thereby permitting its detection (as well as detection of a whole host of other tumor-associated proteins) in the plasma.

PDGF, which also has an important role in glioblastoma angiogenesis – particularly, in peri-endothelial cell recruitment³⁸ - was found to be upregulated in Avastin-treated GBM patients with growing tumors in this study. This finding is supported by past studies that have demonstrated that PDGF and its receptor are co-overexpressed in glioblastoma-derived cell lines as well as in primary GBM tumors, promoting neovascularization and tumor progression by an autocrine mechanism.^{39,40}

The fact that HGF levels were highly overexpressed in the plasma of Avastin-treated GBM patients exhibiting tumor growth as compared to those with stable tumors ($p=0.0055$) confirms previous reports demonstrating that higher tumor HGF content and higher CSF levels of HGF are correlated with increased tumor malignancy and poorer prognosis.⁴¹ HGF has been implicated in synergizing with VEGF to promote glioma angiogenesis and increased microvessel density, particularly by inducing endothelial cell proliferation.^{42,43} It is also known that c-Met receptor activation by HGF enhances several oncogenic mechanisms, including cell cycle progression, proliferation, survival, migration, and invasion, and that GBM progression can be mediated by an HGF/c-Met autocrine loop.⁴² Since tumors require extensive neovascularization for sustained growth, and considering the instrumental role HGF plays in GBM progression, its heightened presence in the plasma of patients with tumor growth seems sensible.

Serum IGFBP2, a binding protein that regulates the bioavailability and bioactivity of IGFs, was also found to be upregulated in this study, corroborating past reports of elevated serum and CSF levels of IGFBP2 in patients with GBM and higher grade gliomas generally.^{44,45} IGFBP2 has previously been shown to be involved in tumor growth regulation both *in vitro* and *in vivo*, and to promote glioma cell migration and invasion.^{46,47} Its increased expression has therefore also been associated with increased glioma malignancy and poorer patient prognosis.⁴⁴

The ability to detect these proteins in the blood is perhaps less surprising when considering the leaky nature of newly-forming blood vessels in and around a glioblastoma tumor,²⁰ as well as the inflammation-associated increase in BBB permeability in the tumor's vicinity.^{48,49} Furthermore, not all the differentially expressed proteins detected are products of tumor cells. Many of these proteins, and particularly the cytokines, are likely secreted from inflammatory and immune cells located either in proximity to the tumor or much farther away, representing a systemic immune or inflammatory anti-tumor response. For example, GM-CSF, IP-10/CXCL-10, and IL13 were all found to be highly expressed in GBM patient plasma as compared to healthy controls. In addition, IL12, MIP1 α , and TGF β 1 were all found to be highly differentially expressed in Avastin-treated GBM patients with growing tumors as compared to those with stable tumors.

Serum GM-CSF expression has previously been shown to be increased in GBM patients.³⁷ This is not surprising considering its important, yet conflicting, roles in promoting tumor proliferation, migration, and angiogenesis on the one hand,^{37,50} while on the other hand stimulating myeloproliferation in order to mount an immune/inflammatory attack against growing tumors.^{49,50} Likewise, MIP1 α and its receptors have been shown to be overexpressed in GBM cells *in vitro*, and likely serve to attract appropriate subsets of inflammatory and immune

effector cells - including lymphocytes and macrophages - to the sites of tissue damage for repair.⁵¹ However, this antitumor activity may be outweighed by an autocrine loop that promotes proliferation of the tumor cells.⁵¹

High differential upregulation of TGF β 1 in patients with growing tumors is consistent with studies showing that tumors, such as glioblastoma, can lose their cytostatic responsiveness to this cytokine, and can instead respond to it by producing PDGF, the tumor growth promoter mentioned previously. Alternatively, tumors can overproduce and utilize TGF β 1 to suppress an antitumor host immune response and evade immune surveillance.⁵² Intriguingly, in this study, patients with growing tumors expressed plasma TGF β 1 at fluorescent intensities approximately 2 orders of magnitude greater than in patients with non-growing tumors ($p=0.0078$). As a result, TGF β 1 on its own was shown to be a sensitive candidate biomarker (sensitivity = 86%) for tumor growth in Avastin-treated GBM patients. By using TGF β 1 in conjunction with HGF, the sensitivity remained the same, while the specificity improved to match that of HGF (65%). Both the PPV and the NPV also improved (to 76% and 79%, respectively).

The high expression of IP-10/CXCL-10 seen in GBM patients in this study could also reflect the immune system's attempt to inhibit further tumor growth. This cytokine is secreted by monocytes, endothelial cells, and fibroblasts as a chemoattractant for recruitment of monocyte-lineage cells, T cells, and NK cells that can participate in an anti-tumor response.⁵³ In addition, it has previously been implicated in inhibition of angiogenesis,⁵³ which is vital for tumor growth. Because its upregulation is induced by IFN γ , it is believed to contribute to the IFN γ -dependent anti-tumor effects of IL12.⁵⁴ This is also consistent with the upregulation of IL12 observed in this study in Avastin-treated GBM patients with tumor growth as compared to those with stable tumors. Interestingly, it also has conflicting tumor-promoting and proliferative effects on non-

transformed astrocytes and cultured glioma cells, and its presence has been correlated with increased malignancy grade.⁵⁵ However, its role as a discriminatory marker in this study may be confounded by the fact that our experimental population was older than the control population, and IP-10 levels naturally increase with age, doubling between ages 40 and 70-80.⁵⁶ Of all the analytes studied, IL13, a cytokine known to have both pro- and anti-tumor effects, showed the highest GBM patient plasma overexpression. This may be attributable to IL13 insensitivity in GBM patients as a result of GBM tumor overexpression of the “decoy” inhibitory receptor IL13R 2,⁵⁷ which may be leading to a compensatory increase in IL13 production.

Surprisingly, the levels of both CRP and MMP9 were actually decreased in GBM patient plasma as compared with healthy controls, and VEGFR2 levels were downregulated in Avastin-treated GBM patients with growing tumors as compared to those with stable tumors. Because of MMP9’s documented involvement in promoting tumor invasion, as well as its anti-apoptotic and pro-angiogenic effects,^{58,59} its decreased plasma level in GBM patients in this study was unanticipated. The decrease in VEGFR2, a VEGF receptor, is also unexpected since one-third of primary glioblastomas harbor amplifications in 3 receptor tyrosine kinase genes that are juxtaposed on chromosome 4: KIT, PDGFRA, and VEGFR2.⁶⁰ Furthermore, past studies have shown that VEGFR2 (and VEGFR1) is highly expressed in primary GBM tumors.⁶¹ However, VEGFR2 downregulation could be explained by the fact that these receptors are internalized by the cell when bound by ligand. Since VEGF levels are high, a significant amount of receptor internalization could be taking place.

The fact that the plasma samples used in this study could be interrogated by multiplexed antibody arrays within ELISA-like wells allowed relatively small sample volumes (<50 μ L) to be used. This suggests that these assays can in the future be performed using blood from a

fingerprick rather than the much larger quantities (milliliters) typically harvested by phlebotomy. In addition, the DNA-directed assembly of antibodies makes this platform amenable for use within microfluidics platforms, since DNA arrays can withstand the bonding temperatures required for platform assembly whereas directly-spotted antibody arrays cannot.^{16,34} Therefore, a promising next step would be to integrate these arrays and antibody panels within a microfluidics-based blood separation diagnostic device much like the Integrated Blood Barcode Chip (IBBC) we previously described.¹⁶ Because on-chip blood separation obviates the need for centrifugation and other blood processing steps, and due to the faster kinetics of ligand capture under conditions of fluid flow, all the assay steps within the microfluidic environment can be performed in under an hour. Consequently, a point-of-care diagnostic chip that probes for the most highly discriminatory proteins described herein for classifying patients into GBM or healthy subgroups (or for gauging treatment response) would allow patients to be diagnosed or monitored using a simple fingerprick blood test within a short time after walking into a doctor's office.

Future studies could also enlarge the microarray panel to hundreds of plasma proteins and evaluate even larger patient populations with varying grades of glioma. This could allow for higher resolution stratification of patients into diagnostic and treatment groups based on their molecular phenotypes, which could be more informative than histological grading alone. Additional studies could also assess the ability of these types of assays to classify patients as responders or non-responders shortly after initiation of treatment. Currently, using contrast-enhanced MRI imaging, it can take at least a week or more to discern whether a tumor is still growing or stable. However, it is likely that molecular changes within the tumor are occurring long before these changes manifest as visible tumor growth and progression. Therefore, a blood

test that could evaluate treatment response within hours of administration of a chemotherapeutic would allow doctors to arrive at the most effective treatment in the shortest possible time. The resulting benefits to the patient's health as well as the cost-savings could be significant.

3.5 Appendix: Supplementary Information

3.5.1 DNA-Encoded Antibody Libraries (DEAL) Technique

The advantages of DEAL are multifold. First, the fact that DNA hybridization is utilized as an assembly strategy allows for multiple proteins to be detected within the same microenvironment, since the primary antibodies for the various proteins to be detected can each be labeled with a different ssDNA oligomer. Second, antibodies are not particularly stable, and as a result, surfaces onto which antibodies are attached are unstable towards drying, heating, etc. This means that antibodies must be attached to the surface immediately prior to use. Using DNA hybridization as an assembly strategy means that the surface can be prepared ahead of time, dried out, heated, shipped around, etc. The instability of antibodies also makes protein assays difficult to execute within microfluidics environments, since the antibodies cannot survive the microfluidics fabrication process. This is, again, circumvented with the DEAL approach.

3.5.2 Serum Protein Biomarker Panels and Oligonucleotide Labels

The protein panels used in the cancer-patient serum experiment (panel 1) and finger-prick blood test (panel 2), the corresponding DNA codes, and their sequences are summarized in **Tables 3.3 and 3.4**. These DNA oligomers were synthesized by Integrated DNA Technologies (IDT), and purified by high pressure liquid chromatography (HPLC). The quality was confirmed by mass spectrometry.

Table 3.3 List of Proteins and Corresponding DNA Codes

DNA-code	Human Plasma Protein	Abbreviation
A/A'	Interleukin-2	IL-2
B/B'	Monocyte Chemotactic Protein 1	MCP1
C/C'	Interleukin-6	IL-6
D/D'	Granulocyte-Colony Stimulating Factor	G-CSF
E/E'	Macrophage Migration Inhibitory Factor	MIF
F/F'	Epidermal Growth Factor	EGF
G/G'	Vascular Endothelial Growth Factor	VEGF
H/H'	Platelet Derived Growth Factor	PDGF
I/I'	Transcription Growth Factor alpha	TGF α
J/J'	Interleukin-8	IL-8
K/K'	Matrix Metalloproteinase 3	MMP3
L/L'	Hepatocyte Growth Factor	HGF
M/M'	Reference (Cy3)	M'-Cy3
N/N'	Interferon-Inducible Protein 10	IP10/CXCL10
O/O'	Stromal Cell-Derived Factor 1	SDF1
P/P'	Insulin-like Growth Factor Binding Protein 2	IGFBP2
S/S'	Insulin-like Growth Factor Binding Protein 5	IGFBP5
U/U'	Macrophage Inflammatory Protein 1 alpha	MIP1 α
Z/Z'	Transcription Growth Factor Beta 1	TGF β 1
AA/AA'	Chitinase 3-like 1	Ch3L1
BB/BB'	Vascular Endothelial Growth Factor Receptor 3	VEGFR3
CC/CC'	Tumor Necrosis Factor alpha	TNF α
HH/HH'	Granulocyte-macrophage colony stimulating factor	C3
III/II'	Matrix Metalloproteinase 2	MMP2
JJ/JJ'	Interleukin-10	IL-10
KK/KK'	Interleukin-1 beta	IL-1 β
MM/MM'	Interleukin-12	IL-12
NN/NN'	Matrix Metalloproteinase 9	MMP9
PP/PP'	Transforming Growth Factor Beta 2	TGF β 2
QQ/QQ'	Granulocyte Macrophage Colony-Stimulating Factor	GM-CSF
RR/RR'	C-Reactive Protein	CRP
SS/SS'	Vascular Endothelial Growth Factor Receptor 2	VEGFR2
TT/TT'	Interleukin-13	IL-13
UU/UU'	Interleukin-23	IL-23
VV/VV'	Serpin E1	Serpin E1
WWW/WW'	Fibrinogen	Fibrinogen

Table 3.4 List of DNA Sequences used for Spatial Encoding of Antibodies

Sequence Name	Sequence)	T _m °C (50mMNaCl)
A	5'-AAAAAAAAAAAAATCCTGGAGCTAAGTCCGTA-3'	57.9
A'	5' NH3-AAAAAAAAAATACGGACTTAGCTCCAGGAT-3'	57.2
B	5'-AAAAAAAAAAAAAGCCTCATTGAATCATGCCTA-3'	57.4
B'	5' NH3-AAAAAAAAAATAGGCATGATTCAATGAGGC-3'	55.9
C	5'-AAAAAAAAAAAAAGCACTCGTCTACTATCGCTA-3'	57.6
C'	5' NH3-AAAAAAAAAATAGCGATAGTAGACGAGTGC-3'	56.2
D	5'-AAAAAAAAAAAAATGGTCGAGATGTCAGAGTA-3'	56.5
D'	5' NH3-AAAAAAAAAATACTCTGACATCTCGACCAT-3'	55.7

E	5'-AAAAAAAAAAAAAATGTGAAGTGGCAGTATCTA-3'	55.7
E'	5' NH3-AAAAAAAAAATAGATACTGCCACTTCACAT-3'	54.7
F	5'-AAAAAAAAAAAAAATCAGGTAAGGTTACCGTA-3'	56.9
F'	5' NH3-AAAAAAAAAATACCGTGAACCTTACCTGAT-3'	56.1
G	5'-AAAAAAAAAAGAGTAGCCTTCCCGAGCATT-3'	59.3
G'	5' NH3-AAAAAAAAAATGCTCGGGAAGGCTACTC-3'	58.6
H	5'-AAAAAAAAAATTGACCAAACGCGGTGCG-3'	59.9
H'	5' NH3-AAAAAAAAACGCACCGCAGTTTGGTCAAT-3'	60.8
I	5'-AAAAAAAAAATGCCCTATTGTTGCGTCGGA-3'	60.1
I'	5' NH3-AAAAAAAAAATCCGACGCAACAATAGGGCA-3'	60.1
J	5'-AAAAAAAAAATCTTCTAGTTGTGCGAGCAGG-3'	56.5
J'	5' NH3-AAAAAAAAACCTGCTCGACAACTAGAAGA-3'	57.5
K	5'-AAAAAAAAAATAATCTAATTCTGGTCGCGG-3'	55.4
K'	5' NH3-AAAAAAAAACCGCGACCCAGATTAGATTA-3'	56.3
L	5'-AAAAAAAAAAGTGATTAAGTCTGCTTCGGC-3'	57.2
L'	5' NH3-AAAAAAAAAGCCGAAGCAGACTTAATCAC-3'	57.2
M	5'-AAAAAAAAAAGTCGAGGATTCTGAACCTGT-3'	57.6
M'	5' NH3-AAAAAAAAAACAGGTTCTGAGTCTCGAC-3'	56.9
AA	5'-AAAAAAAAAATAAGCCAGTGTGTGTCT-3'	58
AA'	5' NH3-AAAAAAAAAAGACACGACACTGGCTTA-3'	58.1
BB	5'-AAAAAAAAAAGTCTGATCCCATCGCGTAT-3'	57.8
BB'	5' NH3-AAAAAAAAAATACGCGATGGGATCAGACT-3'	57.8
CC	5'-AAAAAAAAAAGAGGTCAGTTCACGAAGCTC-3'	58.2
CC'	5' NH3-AAAAAAAAAAGAGCTTCGTGAACCTGACCTC-3'	58.2
HH	5'-AAAAAAAAAAGCACTAAGTGGTCTGGGTCA-3'	59.2
HH'	5' NH3-AAAAAAAAAATGACCCAGACAGTTAGTGC-3'	58.4
II	5'-AAAAAAAAAAGTCAGGTGTTGCGGCTCATT-3'	60.1
II'	5' NH3-AAAAAAAAAATGAGCGCGAACACCTGAC-3'	59.4
JJ	5'-AAAAAAAAAAGATCGTATGGTCCGCTCTCA-3'	58.8
JJ'	5' NH3-AAAAAAAAAATGAGAGCGGACCATACGATC-3'	58
KK	5'-AAAAAAAAAAGCAGGTCATCGAACTCTCAG-3'	56.7
KK'	5' NH3-AAAAAAAAAAGTGAAGTTCGATGACCTGT-3'	57.5
MM	5'-AAAAAAAAAAGGCGGCTATTGACGAACCTCT-3'	59.5
MM'	5' NH3-AAAAAAAAAAGAGTTCGTCAATAGCCGCC-3'	58.8
NN	5'-AAAAAAAAAAGCAGGGAATTGCCGACCATA-3'	59.9
NN'	5' NH3-AAAAAAAAAATATGGTCGGCAATTCCTGC-3'	59.1
PP	5'-AAAAAAAAAAGCGGCGTGTCTCAGAATAT-3'	59.8
PP'	5' NH3-AAAAAAAAAATATTCTGAGACACGCCGCG-3'	58.9
QQ	5'-AAAAAAAAAATCCGGTCTCATCGCTGAAT-3'	58.2
QQ'	5' NH3-AAAAAAAAAATTCAGCGATGAGACCGGAT-3'	58.2
RR	5'-AAAAAAAAAATGCTCACATCGCAGGTAC-3'	57.6
RR'	5' NH3-AAAAAAAAAAGTACCTGCGATGTGAGCATT-3'	58.3
SS	5'-AAAAAAAAAAGCGCTAATGACGGCAGTGCA-3'	60.4
SS'	5' NH3-AAAAAAAAAATGCACTGCCGTCTTAGCGT-3'	60.3
TT	5'-AAAAAAAAAATGTGTCCGAACGTCGAGCT-3'	59.8
TT'	5' NH3-AAAAAAAAAAGCTCGACGTTCCGACACAT-3'	59.8
UU	5'-AAAAAAAAAAGCCGTCGGTTCAGGTCATAT-3'	59.4
UU'	5' NH3-AAAAAAAAAATATGACCTGAACCGACGGC-3'	58.7
VV	5'-AAAAAAAAAAGTCGCGGGTCTGCACATAT-3'	59.9
VV'	5' NH3-AAAAAAAAAATATGTGCAGAACCCGCGAC-3'	59.2

* All amine-terminated strands were linked to antibodies to form DNA-antibody conjugates using SFB/SANH coupling chemistry described by R. Bailey *et al.*³³ Codes AA-HH were used in the experiment examining fresh whole blood from a healthy volunteer. Codes A-M were used for the molecular analyses of cancer patient serum samples.

Table 3.5 Antibody Vendors and Catalogue Numbers

	Company Name	Capture Antibody (Catalogue #)	Detection Antibody (Catalogue #)
IL2	BD	555051	555040
MCP1	eBioscience	16-7099-85	13-7096-85
IL6	eBioscience	16-7069-85	13-7068-85
G-CSF	R&D systems	Mab214	BAF214
MIF	R&D systems	mab289	baf289
EGF	R&D systems	MAB636	BAF236
VEGF	R&D systems	mab293	baf293
PDGF	R&D systems	MAB1739	BAF221
TGFα	R&D systems	AF-239-NA	BAF239
IL8	BD	554718	554716
MMP3	R&D systems	AF513	BAF513
HGF	R&D systems	MAB694	BAF294
IP10	R&D systems	MAB266	BAF266
SDF1	R&D systems	MAB350	BAF310
IGFBP2	R&D systems	MAB6741	BAF674
IGFBP5	R&D systems	MAB8751	BAF875
MIP1a	R&D systems	AF-270-NA	BAF270
TGFb1	BD	559119	559119
Ch3L1	R&D systems	DY2599	DY2599
VEGFR3	R&D systems	MAB349	BAM3492
TNFα	eBioscience	16-7348-85	13-7349-85
C3	Abcam	ab17455-100	ab14232-50
MMP2	R&D systems	DY1496	DY1496
IL10	eBioscience	16-7108-85	13-7109-85
IL1β	eBioscience	16-7018-85	13-7016-85
IL12	eBioscience	14-7128-82	13-7129-85
MMP9	R&D systems	MAB9092	BAM909
TGFb2	R&D systems	DY302	DY302
GM-CSF	BD	554502	554505
CRP	R&D systems	MAB17071	BAM17072
VEGF R2	R&D systems	MAB3573	BAF357
IL13	eBioscience	16-7139-81	13-7138-81
IL23	eBioscience	14-7238-85	13-7129-85
Serpin E1	R&D systems	MAB1786	BAF1786
Fibrinogen	Abcam	ab10066-250	ab14790-200

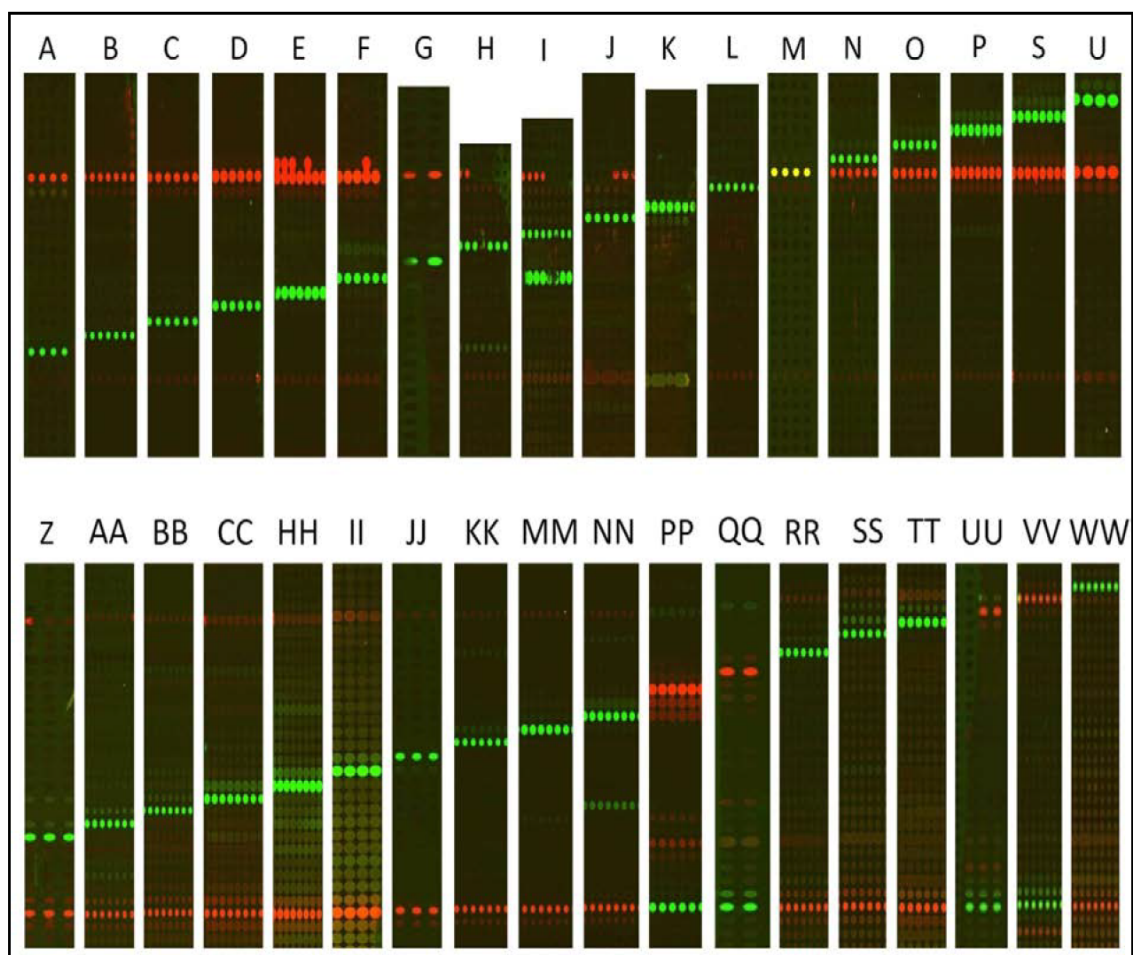


Figure 3.6 Test for DNA cross-hybridization. 36 ssDNA oligonucleotides (dye conjugates) were each tested separately for orthogonality against the full set of 36 surface-bound complementary strands. **Red = M'-Cy5 Reference**; **Green = Cy3-conjugated oligonucleotide**. However, for PP, QQ, UU, and VV: Green = M'-Cy3 Reference; Red = Cy5-conjugated oligonucleotide. Cross-hybridization was far less than the 5% cut-off for all oligonucleotides except for "I", where the cross-hybridization level with "F" was 8% (and can be distinctly seen in the image).

3.6 References

- 1 McLendon, R. *et al.* Comprehensive genomic characterization defines human glioblastoma genes and core pathways. *Nature* **455**, 1061-1068 (2008).
- 2 Fauci, A. *Harrison's Principles of Internal Medicine*. (McGraw-Hill New York, 2005).
- 3 Deimling, A. *et al.* Subsets of glioblastoma multiforme defined by molecular genetic analysis. *Brain Pathology* **3**, 19-26 (1993).
- 4 Jiang, R. *et al.* Pathway alterations during glioma progression revealed by reverse phase protein lysate arrays. *Proteomics* **6**, 2964-2971 (2006).
- 5 Freije, W. *et al.* Gene expression profiling of gliomas strongly predicts survival. *Cancer Research* **64**, 6503 (2004).
- 6 Sreekanthreddy, P. *et al.* Identification of Potential Serum Biomarkers of Glioblastoma: Serum Osteopontin Levels Correlate with Poor Prognosis. *Cancer Epidemiology Biomarkers & Prevention* **19**, 1409 (2010).
- 7 Huang, R., Huang, R., Fan, Y. & Lin, Y. Simultaneous detection of multiple cytokines from conditioned media and patient's sera by an antibody-based protein array system. *Analytical Biochemistry* **294**, 55-62 (2001).
- 8 Iwadata, Y. *et al.* Molecular classification and survival prediction in human gliomas based on proteome analysis. *Cancer Research* **64**, 2496 (2004).
- 9 Schwartz, S., Weil, R., Johnson, M., Toms, S. & Caprioli, R. Protein profiling in brain tumors using mass spectrometry. *Clinical Cancer Research* **10**, 981 (2004).
- 10 Mischel, P., Cloughesy, T. & Nelson, S. DNA-microarray analysis of brain cancer: molecular classification for therapy. *Nature Reviews Neuroscience* **5**, 782-792 (2004).
- 11 Kingsmore, S. Multiplexed protein measurement: technologies and applications of protein and antibody arrays. *Nature Reviews Drug Discovery* **5**, 310-321 (2006).
- 12 Lv, L. & Liu, B. High-throughput antibody microarrays for quantitative proteomic analysis. *Expert Review of Proteomics* **4**, 505-513 (2007).
- 13 Varnum, S., Woodbury, R. & Zangar, R. A protein microarray ELISA for screening biological fluids. *Methods in Molecular Biology* **264**, 161-172 (2004).
- 14 Sack, R. *et al.* Membrane array characterization of 80 chemokines, cytokines, and growth factors in open-and closed-eye tears: angiogenin and other defense system constituents. *Investigative Ophthalmology & Visual Science* **46**, 1228 (2005).
- 15 Kastenbauer, S., Angele, B., Sporer, B., Pfister, H. & Koedel, U. Patterns of protein expression in infectious meningitis: a cerebrospinal fluid protein array analysis. *Journal of Neuroimmunology* **164**, 134-139 (2005).
- 16 Fan, R. *et al.* Integrated barcode chips for rapid, multiplexed analysis of proteins in microliter quantities of blood. *Nature Biotechnology* **26**, 1373-1378 (2008).
- 17 Hanash, S., Pitteri, S. & Faca, V. Mining the plasma proteome for cancer biomarkers. *Nature* **452**, 571-579 (2008).
- 18 Rubin, L. & Staddon, J. The cell biology of the blood-brain barrier. *Annual Review of Neuroscience* **22**, 11-28 (1999).
- 19 Abbott, N. Inflammatory mediators and modulation of blood-brain barrier permeability. *Cellular and Molecular Neurobiology* **20**, 131-147 (2000).
- 20 Kumar, V., Abbas, A. & Fausto, N. *Robbins & Cotran Pathologic Basis of Disease (7th ed.)*. (2005).
- 21 Leon, S., Folkerth, R. & Black, P. Microvessel density is a prognostic indicator for patients with astroglial brain tumors. *Cancer* **77**, 362-372 (1996).

- 22 Kargiotis, O., Rao, J. & Kyritsis, A. Mechanisms of angiogenesis in gliomas. *Journal of Neuro-Oncology* **78**, 281-293 (2006).
- 23 Schneider, S. *et al.* Glioblastoma cells release factors that disrupt blood-brain barrier features. *Acta Neuropathologica* **107**, 272-276 (2004).
- 24 Ludwig, J. & Weinstein, J. Biomarkers in cancer staging, prognosis and treatment selection. *Nature Reviews Cancer* **5**, 845-856 (2005).
- 25 Jiang, R., Li, J., Fuller, G. & Zhang, W. Proteomic Profiling of Human Brain Tumors. *CNS Cancer*, 553-575 (2009).
- 26 Ray, S. *et al.* Classification and prediction of clinical Alzheimer's diagnosis based on plasma signaling proteins. *Nature Medicine* **13**, 1359-1362 (2007).
- 27 Wulfkühle, J., Liotta, L. & Petricoin, E. Proteomic applications for the early detection of cancer. *Nature Reviews Cancer* **3**, 267-275 (2003).
- 28 Sanchez-Carbayo, M., Succi, N., Lozano, J., Haab, B. & Cordon-Cardo, C. Profiling bladder cancer using targeted antibody arrays. *American Journal of Pathology* **168**, 93 (2006).
- 29 Orzechowski, R. *et al.* Antibody microarray profiling reveals individual and combined serum proteins associated with pancreatic cancer. *Cancer Research* **65**, 11193 (2005).
- 30 Miller, J. *et al.* Antibody microarray profiling of human prostate cancer sera: antibody screening and identification of potential biomarkers. *Proteomics* **3**, 56-63 (2003).
- 31 Ellmark, P. *et al.* Identification of protein expression signatures associated with *Helicobacter pylori* infection and gastric adenocarcinoma using recombinant antibody microarrays. *Molecular & Cellular Proteomics* **5**, 1638 (2006).
- 32 Wacker, R., Schröder, H. & Niemeyer, C. Performance of antibody microarrays fabricated by either DNA-directed immobilization, direct spotting, or streptavidin-biotin attachment: a comparative study. *Analytical Biochemistry* **330**, 281-287 (2004).
- 33 Bailey, R. C., Kwong, G. A., Radu, C. G., Witte, O. N. & Heath, J. R. DNA-encoded antibody libraries: A unified platform for multiplexed cell sorting and detection of genes and proteins. *Journal of the American Chemical Society* **129**, 1959-1967 (2007).
- 34 Bailey, R., Kwong, G., Radu, C., Witte, O. & Heath, J. DNA-encoded antibody libraries: a unified platform for multiplexed cell sorting and detection of genes and proteins. *Journal of the American Chemical Society* **129**, 1959-1967 (2007).
- 35 Hicklin, D. & Ellis, L. Role of the vascular endothelial growth factor pathway in tumor growth and angiogenesis. *Journal of Clinical Oncology* **23**, 1011 (2005).
- 36 Takano, S. *et al.* Concentration of vascular endothelial growth factor in the serum and tumor tissue of brain tumor patients. *Cancer Research* **56**, 2185 (1996).
- 37 Rafat, N., Beck, G., Schulte, J., Tuettenberg, J. & Vajkoczy, P. Circulating endothelial progenitor cells in malignant gliomas. *Journal of Neurosurgery: Pediatrics* **112** (2010).
- 38 Guo, P. *et al.* Platelet-derived growth factor-B enhances glioma angiogenesis by stimulating vascular endothelial growth factor expression in tumor endothelia and by promoting pericyte recruitment. *American Journal of Pathology* **162**, 1083 (2003).
- 39 Hermansson, M. *et al.* Endothelial cell hyperplasia in human glioblastoma: coexpression of mRNA for platelet-derived growth factor (PDGF) B chain and PDGF receptor suggests autocrine growth stimulation. *Proceedings of the National Academy of Sciences of the United States of America* **85**, 7748 (1988).
- 40 Guha, A., Dashner, K., McBlack, P., Wagner, J. & Stiles, C. Expression of PDGF and PDGF receptors in human astrocytoma operation specimens supports the existence of an autocrine loop. *International Journal of Cancer* **60**, 168-173 (1995).

- 41 Garcia-Navarrete, R., Garcia, E., Arrieta, O. & Sotelo, J. Hepatocyte growth factor in cerebrospinal fluid is associated with mortality and recurrence of glioblastoma, and could be of prognostic value. *Journal of Neuro-Oncology* **97**, 347-351 (2010).
- 42 Abounader, R. & Lattera, J. HGF/c-Met Signaling and Targeted Therapeutics in Brain Tumors. *CNS Cancer*, 933-952 (2009).
- 43 Schmidt, N. *et al.* Levels of vascular endothelial growth factor, hepatocyte growth factor/scatter factor and basic fibroblast growth factor in human gliomas and their relation to angiogenesis. *International Journal of Cancer* **84**, 10-18 (1999).
- 44 Lin, Y. *et al.* Plasma IGFBP-2 levels predict clinical outcomes of patients with high-grade gliomas. *Neuro-oncology* **11**, 468 (2009).
- 45 Hoefflich, A. *et al.* Insulin-like growth factor-binding protein 2 in tumorigenesis: protector or promoter? *Cancer Research* **61**, 8601 (2001).
- 46 Dunlap, S. *et al.* Insulin-like growth factor binding protein 2 promotes glioma development and progression. *Proceedings of the National Academy of Sciences* **104**, 11736 (2007).
- 47 Wang, H. *et al.* Insulin-like growth factor binding protein 2 enhances glioblastoma invasion by activating invasion-enhancing genes. *Cancer Research* **63**, 4315 (2003).
- 48 Fossati, G. *et al.* Neutrophil infiltration into human gliomas. *Acta Neuropathologica* **98**, 349-354 (1999).
- 49 Frei, K. *et al.* Granulocyte-macrophage colony-stimulating factor (GM-CSF) production by glioblastoma cells. Despite the presence of inducing signals GM-CSF is not expressed in vivo. *The Journal of Immunology* **148**, 3140 (1992).
- 50 Mueller, M. *et al.* Autocrine growth regulation by granulocyte colony-stimulating factor and granulocyte macrophage colony-stimulating factor in human gliomas with tumor progression. *American Journal of Pathology* **155**, 1557 (1999).
- 51 Kouno, J. *et al.* Up-regulation of CC chemokine, CCL3L1, and receptors, CCR3, CCR5 in human glioblastoma that promotes cell growth. *Journal of Neuro-Oncology* **70**, 301-307 (2004).
- 52 Massagué, J. & Gomis, R. The logic of TGF [beta] signaling. *FEBS letters* **580**, 2811-2820 (2006).
- 53 Enderlin, M. *et al.* TNF- α and the IFN- γ -inducible protein 10 (IP-10/CXCL-10) delivered by parvoviral vectors act in synergy to induce antitumor effects in mouse glioblastoma. *Cancer Gene Therapy* **16**, 149-160 (2008).
- 54 Phelps, C. & Korneva, E. *Cytokines and the Brain*. (Elsevier Science Ltd, 2008).
- 55 Maru, S. *et al.* Chemokine production and chemokine receptor expression by human glioma cells: role of CXCL10 in tumour cell proliferation. *Journal of Neuroimmunology* **199**, 35-45 (2008).
- 56 Shurin, G. *et al.* Dynamic alteration of soluble serum biomarkers in healthy aging. *Cytokine* **39**, 123-129 (2007).
- 57 Liu, H. *et al.* Interleukin-13 sensitivity and receptor phenotypes of human glial cell lines: non-neoplastic glia and low-grade astrocytoma differ from malignant glioma. *Cancer Immunology, Immunotherapy* **49**, 319-324 (2000).
- 58 Yong, V., Power, C., Forsyth, P. & Edwards, D. Metalloproteinases in biology and pathology of the nervous system. *Nature Reviews Neuroscience* **2**, 502-511 (2001).
- 59 Egeblad, M. & Werb, Z. New functions for the matrix metalloproteinases in cancer progression. *Nature Reviews Cancer* **2**, 163-176 (2002).
- 60 Puputti, M. *et al.* Amplification of KIT, PDGFRA, VEGFR2, and EGFR in gliomas. *Molecular Cancer Research* **4**, 927 (2006).
- 61 Huang, H., Held-Feindt, J., Buhl, R., Mehdorn, H. & Mentlein, R. Expression of VEGF and its receptors in different brain tumors. *Neurological Research* **27**, 371-377 (2005).

4 Computational and Analytical Tools for Diagnostic Measurements

4.1 Automation of Data Processing and Analysis

Analyzing highly-multiplexed protein assays from large numbers of patients requires an efficient means of processing large datasets. Automating the computational steps from data acquisition to statistical analysis can save a considerable amount of time and effort. In fact, without automation, scaling clinical trials to assays of hundreds or thousands of proteins and patient samples would render analyses of the resulting datasets intractable. A straightforward approach for creating algorithms to manipulate data in Microsoft Excel is to write macro procedures in Visual Basic for Applications (VBA).

In our clinical trial examining patients with glioblastoma, plasma samples were assayed for 35 proteins (and a spiked reference oligo) within ELISA-like wells (12 per slide), each containing six repeating 6x6 spot arrays. These wells were fashioned by bonding a PDMS slab with 12 square holes to a DNA-spotted, polylysine-coated glass substrate. The output file from the GenePix scanner software gives the row, column, and block (or well) number of each spot based on its location in a graphical spot array template whose parameters (number of blocks, rows, and columns, as well as spot sizes and spacings) are defined by the user. Had all 12 square holes in the PDMS slab been cut with uniform dimensions and spacings, and had the PDMS slab been precisely aligned with respect to the spots on the slide, the registry of oligo spots in all wells and among all slides would be identical. In other words, the identity of a spot located

within a particular row and column of a well would be the same for all wells. A list in which the row and column positions of each spot within a well (block) are matched with their corresponding identifiers could then be input into the GenePix analysis software, allowing for instant spot assignment.

However, in our study, the square holes in the PDMS slab were cut by hand, resulting in slight variability in the well dimensions and spacings. Furthermore, we did not attempt to align the PDMS slab with the spotted arrays in any way, as this would have greatly extended fabrication time and effort. As a result, the registry of spots could vary considerably across wells on the same slide and between different slides. Consequently, some means of accurately assigning an identifier to all assay spots in a well was needed. To accomplish this, we designated one of the oligos (oligo M) as a reference. To distinguish this spot from all other spots, we incubated all wells (in the final assay step) with a Cy3 (green) dye-conjugated oligo having sequence complementarity with oligo M. By contrast, all remaining assay spots fluoresced red due to development of the protein assays with Stretavidin-Cy5. Since the oligos were spotted in the same order within all arrays of the slide, the oligo identity (and its associated antibody) for any given spot could be determined by counting its row and column distance from the green reference oligo. Alternatively, an Excel macro (or VBA subroutine) could be written, as was done here, that accomplishes the spot assignment task in exactly the same way, but far more quickly.

Macros were also written to perform all subsequent data handling steps (see Appendix, **Section 4.4**). For example, once the spot positions within a well and their fluorescent intensity values were assigned to a specific oligo/antibody, the average intensity and standard deviation of all repeats were calculated for each protein. Experiments showed that at least 4 proteins in each

sample assay exhibited intensities close to those in negative controls (which were performed by substituting 3% BSA/PBS for plasma samples). Therefore, a baseline intensity (intensity of a spot in the absence of cognate protein) for each patient sample could be approximated by averaging the intensities of the 4 lowest-intensity proteins within each assay. The (mean) intensities for all proteins and the baseline protein intensity level were then displayed graphically for all patients and transferred to Powerpoint automatically. Finally, the mean protein intensity values for each of the 12 patients on a slide were collated (into 12 rows) onto a single Excel worksheet for subsequent processing. This procedure was repeated in automated fashion for all patient samples on all slides. Datasets containing the baseline-subtracted intensity values and standard deviations (for all patients) were created in a similar fashion. A subroutine was written that could transfer the collated data from all open Excel workbooks (each containing its analysis of a different 12-patient slide) to a new Excel file, such that the data for all patients could be found in a single Excel worksheet. Patient ID numbers and clinical information were then manually transferred and aligned with their corresponding row of data. The final result was a master dataset in which each row – corresponding to a distinct patient sample - contained the patient characteristics and clinical information, mean protein intensities, baseline-subtracted mean protein intensities, and standard deviations. More specifically, the format of the master worksheet was as follows: Column A – Tumor Growth Status (Growth vs. No Growth); Column B – Blood Collection Date; Columns C and D – Patient Last Name and First Name, respectively; Column E – IOIS Number; Column F – Patient ID Number; Column G – Date of Birth; Column H – Current Age; Column I – Alive or Deceased; Column J – Overall Survival; Column K – Initial Pathology; Column L – Current Pathology; Column M – Gender; Column N – Chemotherapy Drug (i.e. Avastin vs. No Avastin); Column O – “Was patient on Avastin at the

time of the blood collection date in Column B?"; Column P – Tumor Recurrence Number; Column Q - Chemotherapy Start Date; Column R – Chemotherapy End Date; Columns U through BD – Mean Fluorescent Intensities (Baseline Subtracted) for Proteins 1 through 35 (plus M'-Cy3 reference). Columns BF through CO – Standard Deviations for Proteins 1 through 35 (plus M'-Cy3 reference); Column CQ – Time of Blood Sample Collection; Column CR – Time at which Plasma Sample was Frozen; Column CS – Total Processing Time; Columns CU through ED – Proteins 1 through 35 (plus M'-Cy3 reference) Mean Fluorescent Intensities (Non-Baseline Subtracted). In summary, all the data and relevant clinical information for every single patient in the study was included in the master worksheet (See **Figure 4.1** below).

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y
		Collection Date	LAST	FIRST	Initials	Patient ID	DOB	Current Age	Status	Overall Survival	Initial Pathology	Current Pathology	Gender	AVASTIN	Avastin at Time	Recurrence	Start Date	Stop Date	TTP: [Chen TTS: [Chen	A-HL2	B'-MCP1	C'-IL6	D'-GCSF	E'-MIF	
1	Growth	10/19/2009						45.58	DECEASED	409	UNK	GBM	FEMALE	YES							5832.4	3454.6	2119.2	2861.6	162
2	Growth	10/19/2009						30.75	ALIVE	913	GBM	GBM	MALE	YES							3838.85	2084.6	1981.6	2339.2	233
3	Growth	10/19/2009						59.67	ALIVE	561	GBM	GBM	MALE	YES							4874.15	2764.35	2666.55	3699.95	2593
4	Growth	10/23/2009						49.83	ALIVE	266	GBM	GBM	FEMALE	yes							1302.2	592.2	430.6	508	29
5	Growth	10/21/2009						63.33	ALIVE	413	UNK	GBM	MALE	YES							3595.2	4698.667	2574.6	1584	112
6	Growth	10/26/2009						37.92	ALIVE	210	UNK	GBM	MALE	yes							961.2	900.2	669.8	560.8	72
7	Growth	11/2/2009						65.58	ALIVE	474	UNK	GBM	FEMALE	YES							2455.5	3327.3	3382.1	4176.3	1583.4
8	Growth	11/5/2009						49.83	ALIVE	266	GBM	GBM	FEMALE	yes							1222.4	493.8	912.6	1173.6	55
9	Growth	11/4/2009						30.75	ALIVE	913	GBM	GBM	MALE	YES							3311.467	1552.133	1090.2	1376	115
10	Growth	11/4/2009						63.33	ALIVE	413	UNK	GBM	MALE	YES							5026.9	5961.3	2933.7	4636.833	31
11	Growth	11/9/2009						65.58	ALIVE	453	GBM	GBM	MALE	YES							3162.567	4689.3	6509.5	3934.5	725
12	Growth	11/9/2009						52.83	ALIVE	701	GBM	GBM	MALE	yes							2684.8	1125.2	1221	3965.933	241
13	Growth	11/10/2009						65.58	ALIVE	474	UNK	GBM	FEMALE	YES							722.4	817	1054.8	960.4	-12
14	Growth	11/10/2009						61	ALIVE	476	GBM	GBM	FEMALE	YES							612	619.8	-30.6	552	86
15	Growth	11/30/2009						75.67	ALIVE	332	UNK	GBM	MALE	YES		1st	8/13/2009	9/8/2009	1/59	169	932.5	3177.7	2595.1	3869.833	2324.8
16	Growth	12/4/2009						30.75	ALIVE	934	GBM	GBM	MALE	YES		2nd	9/22/2008	10/19/2009	408	594	590.1	931.1	227.7	1056.1	41
17	Growth	1/6/2010						82.67	ALIVE	455	UNK	GBM	MALE	YES		1st	4/28/2009	5/27/2009	50	576	6167.633	2551.3	1587.3	2451.7	175
18	Growth	1/12/2010						52.5	ALIVE	401	GBM	GBM	MALE	YES		New	1/20/2009	9/2/2009	1/239	574	757.2	1642.867	781.2	2702.867	1383.8
19	Growth	11/30/2009						65.58	ALIVE	495	UNK	GBM	FEMALE	YES		1st	3/24/2009	6/29/2009	1/11	511	965.4	534.4	291.6	333.4	3
20	Growth	12/2/2009						82.67	ALIVE	455	UNK	GBM	MALE	YES		1st	4/28/2009	5/27/2009	50	576	3127.7	2618.9	2077.7	2544.3	138
21	Growth	12/2/2009						63.33	ALIVE	434	UNK	GBM	MALE	YES		1st	3/11/2009	11/4/2009	252	524	1286.6	933.8	795.2	556	29
22	Growth	12/4/2009						49.83	ALIVE	287	GBM	GBM	FEMALE	YES		1st	9/8/2009	11/16/2009	55	143	364.8	500.2	275.8	329	6
23	Growth	12/7/2009						48.5	ALIVE	329	GBM	GBM	MALE	YES		1st	8/16/2009	12/7/2009	1/24	163	836	573.8	459.2	1947	5
24	Growth	12/8/2009						59.83	ALIVE	970	UNK	GBM	FEMALE	YES		New	8/16/2009	9/16/2009	517	597	1043.333	1269.733	1785.733	1646.733	2365.1
25	Growth	12/9/2009						82.67	ALIVE	455	UNK	GBM	MALE	YES		1st	4/28/2009	5/27/2009	50	576	2922.233	2585.9	2019.7	2217.1	154
26	Growth	12/11/2009						52.5	ALIVE	401	GBM	GBM	MALE	YES		New	1/20/2009	9/2/2009	1/239	574	678.4	815.2	853.4	983	50
27	Growth	12/14/2009						75.67	ALIVE	332	UNK	GBM	MALE	YES		1st	8/13/2009	9/8/2009	1/59	169	1055.1	781.3667	256.5	344.7	50
28	Growth	1/11/2010						52.42	ALIVE	886	UNK	GBM	FEMALE	YES		1st	8/25/2008	10/20/2008	58	522	1156	1784.333	1463.667	2089.333	13
29	Growth	2/4/2010						50.5	ALIVE	1359	GBM	GBM	MALE	YES		4th	1/27/2010	3/3/2010			1730.9	893.3	848.3	894.3	86
30	Growth	2/12/2010						49.83	ALIVE	287	GBM	GBM	FEMALE	YES		1st	9/8/2009	11/16/2009	55	143	425.6667	257.8667	540.6	941.6667	314.6
31	Growth	2/18/2010						52.5	ALIVE	401	GBM	GBM	MALE	YES		New	1/20/2009	9/2/2009	1/239	574	1024.267	1224.8	917.6	1920.8	111
32	Growth	2/24/2010						62.83	ALIVE	1835	GBM	GBM	MALE	YES		3rd	8/25/2008	1/27/2010	50	50	318.4	657.6	284.4	837.5333	48
33	Growth	2/22/2010						52.42	ALIVE	886	UNK	GBM	FEMALE	YES		1st	8/25/2008	10/20/2008	58	522	1724.6	1078.6	914.6	1560.6	97
34	Growth	12/9/2009						61	ALIVE	497	GBM	GBM	FEMALE	YES 184		3rd	1/25/2010	1/7/2010	5	5	515.6	238.6	570	623	84
35	Growth	12/11/2009						61	ALIVE	497	GBM	GBM	FEMALE	YES 184		3rd	1/25/2010	1/7/2010	5	5	707.5	231.1	337.9	174.3	34

Figure 4.1 Master patient dataset: organization of clinical information. Only a portion of the full dataset is shown. (Patient identifiers have been removed).

Macros were also written to automate graphing of the patient data within the master worksheet. One of these macros graphs the protein data in each row (corresponding to a unique patient sample) in a separate graph, all of which can then be automatically transferred to a

Powerpoint file. Other macros can display the protein data from all of a patient's blood collections in a single graph (once the file has been sorted first by patient name and then by collection date), such that changes in protein levels within the patient's plasma can be traced over time. These macros then repeat the process for all patients in the worksheet.

From the master worksheet, patient cohorts can straightforwardly be created by reorganizing, sorting, and trimming the data with regard to any one of the parameters in the clinical information columns. For example, one could sort the dataset based on current clinical pathology (Column L) to extract a cohort of GBM patients vs. healthy controls. To create a cohort in which tumor growth status is compared among Avastin-treated GBM patients, the dataset is sorted first by Column L (GBM vs. No GBM), then by Column N (Avastin vs. No Avastin), and finally by Column A (Tumor Growth vs. No Growth). Patients who do not have GBM and are not on Avastin are subsequently removed from the set.

Once these cohorts are created, a series of subroutines are required to facilitate statistical and graphical analysis, hierarchical clustering of the data, and the utilization of these hierarchical clusters for patient classification. The "RunClusterPrep" macro accomplishes these tasks as follows. First, the patient data worksheet is reorganized and formatted appropriately for compatibility with the clustering software, *Cluster 3.0*. Second, experimental and control group mean and median fluorescent intensities are calculated for each protein assayed (as well as the differences and root-mean-square distances between experimental and control group means and medians). These values are then displayed graphically. Next, an additional file is created in which the experimental and control data (for each protein) are formatted for facile transfer to and analysis by "AnalyseIt", a statistical software add-in for Excel (For details and additional related macros, see **Section 4.4.5**). The user can then run a number of different statistical tests on the

transferred data (now residing in tabulated form within an AnalyseIt template file). In our clinical trial, we most commonly utilized the Student's t-test (sensitive to differences in population means) and Mann Whitney test (sensitive to differences in population medians) to assess the statistical significance (p -value) of differential protein expression between experimental and control groups. We also utilized AnalyseIt's box plot function to be able to visually compare (for each protein) the experimental and control population means, standard deviations, and 95% confidence intervals, as well as medians, quartiles, outliers, and general spread of the data.

In addition, the "RunClusterPrep" subroutine facilitates diagnostic testing in the following way. The subroutine randomly assigns a certain number of patients (number specified by the user) within a cohort dataset as "unknown" test samples. The resulting test file, containing both "known" and "unknown" patient samples, is converted to text format, such that it can then be clustered (by Average-Linkage Hierarchical Clustering) using *Cluster 3.0*. The cluster map (or heat map) can subsequently be viewed using Java *TreeView*. In a classification scheme that can most appropriately be described as "guilt-by-association", the unknown patients are classified by the tester as belonging to the experimental or control group based on the majority diagnosis of neighbors within their cluster. The macro "CalculateStatistics" (**Section 4.4.4**) then compares the predicted and actual diagnoses, determines the true positives/negatives and false positives/negatives, and creates a 2x2 contingency table for these values. Measures of diagnostic accuracy, such as the sensitivity, specificity, positive predictive value (PPV), and negative predictive value (NPV), are then calculated by the macro. The RunClusterPrep subroutine creates multiple test files (number specified by the user), each with its own set of randomly assigned unknown samples. Thus, the "guilt-by-association" classification procedure can be repeated

multiple times, allowing the diagnostic accuracy of the procedure to be assessed with greater statistical power.

As mentioned before, the number of test files to be created and the number of unknowns to be assigned within each test file are specified by the user. In addition, the user must specify the number of proteins being examined. To facilitate entry of these parameters by the user, a customized user interface has been created. This interface also allows the user to specify the directory into which the new folder, “NewTrialFolder” – containing the files to be created by the “RunClusterPrep” macro - should be saved. The combination of the “RunClusterPrep” macro (with its associated subroutines) and the user interface form a software package we call “ClusterPrep”. To initiate or “open” the program, we have created a command button for the Excel Add-Ins Toolbar labeled “RunAnalysis” (see below).

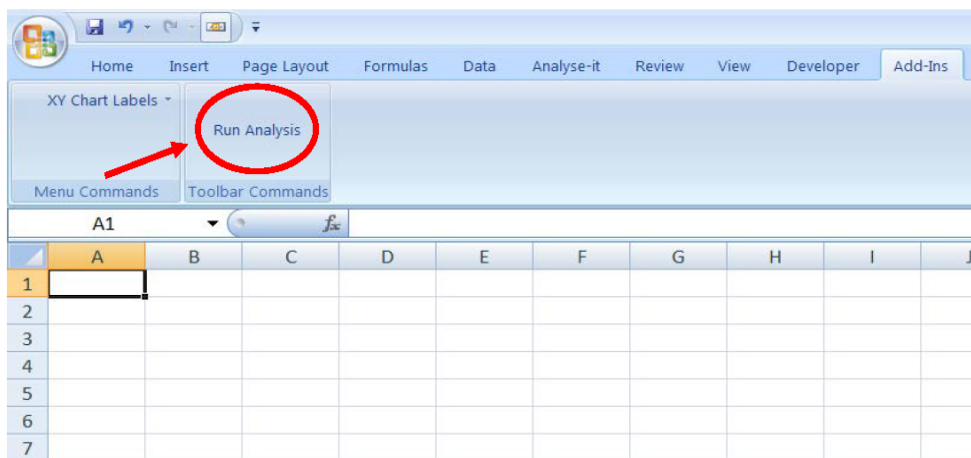
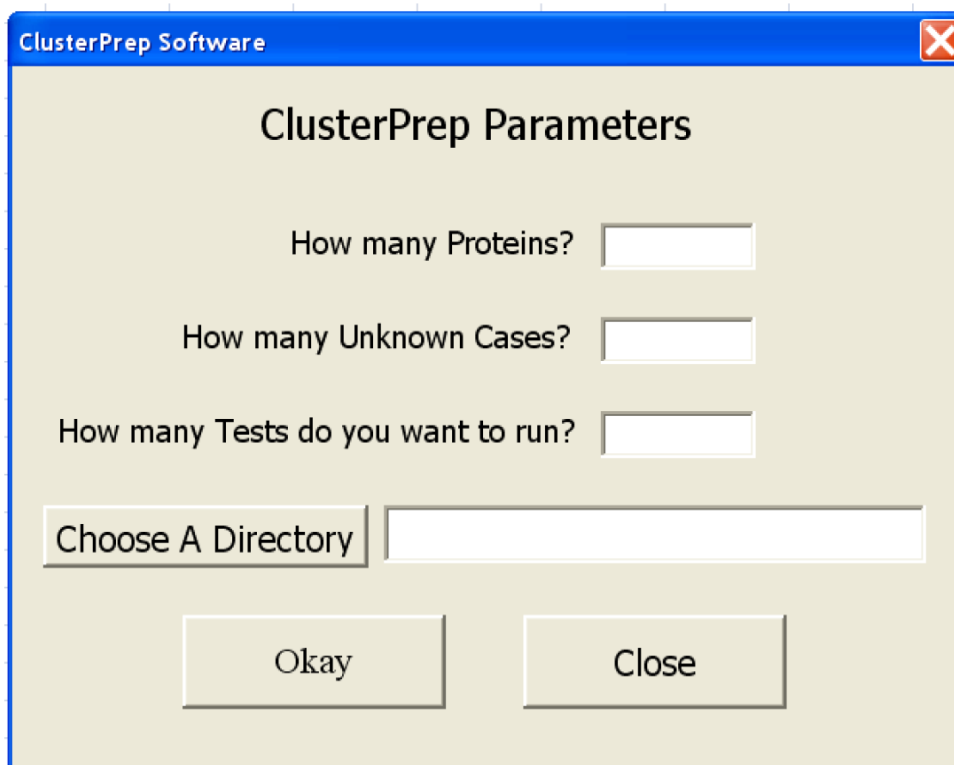


Figure 4.2 The “Run Analysis” command button in the Excel add-ins toolbar. When this command button is clicked, the “ClusterPrep” software program is initiated.

When this button is clicked, the user interface is first displayed (**Figure 4.3**). Once the user inputs the required parameters and clicks “Okay”, the “RunClusterPrep” macro and its associated subroutines are executed. The output files and folders are typically created within about a

The image shows a Windows-style dialog box titled "ClusterPrep Software" with a standard close button (X) in the top right corner. The main title of the dialog is "ClusterPrep Parameters". It contains three input fields, each preceded by a label: "How many Proteins?" followed by a text box, "How many Unknown Cases?" followed by a text box, and "How many Tests do you want to run?" followed by a text box. Below these is a button labeled "Choose A Directory" next to a larger text box for specifying a directory. At the bottom of the dialog are two buttons: "Okay" and "Close".

ClusterPrep Software

ClusterPrep Parameters

How many Proteins?

How many Unknown Cases?

How many Tests do you want to run?

Choose A Directory

Okay Close

Figure 4.3 The “ClusterPrep” user interface. The user inputs the number of proteins being analyzed, the number of samples to randomly set aside as test samples, and the number of tests to run. The user also designates the directory into which the output files will be saved.

minute; however, much longer times are needed if the number of test files and unknowns specified by the user is great. For our data analysis, we typically chose to have “ClusterPrep” create 10 test files with 10 unknowns in each file.

While the “ClusterPrep Software” package greatly increases the efficiency of statistical analysis and of creating files for cluster analysis, transferring these files into *Cluster 3.0* manually is still a time-consuming task. Therefore, we have created a batch file that executes the cluster analysis on each test file in the “NewTrialFolder” directory directly from the command line. The batch file can be edited to produce multiple Cluster output files (.cdt) for each test file, each with a different distance/similarity measure, normalization, and clustering method. For this clinical trial, we used the Average-Linkage Hierarchical Clustering method with the Pearson

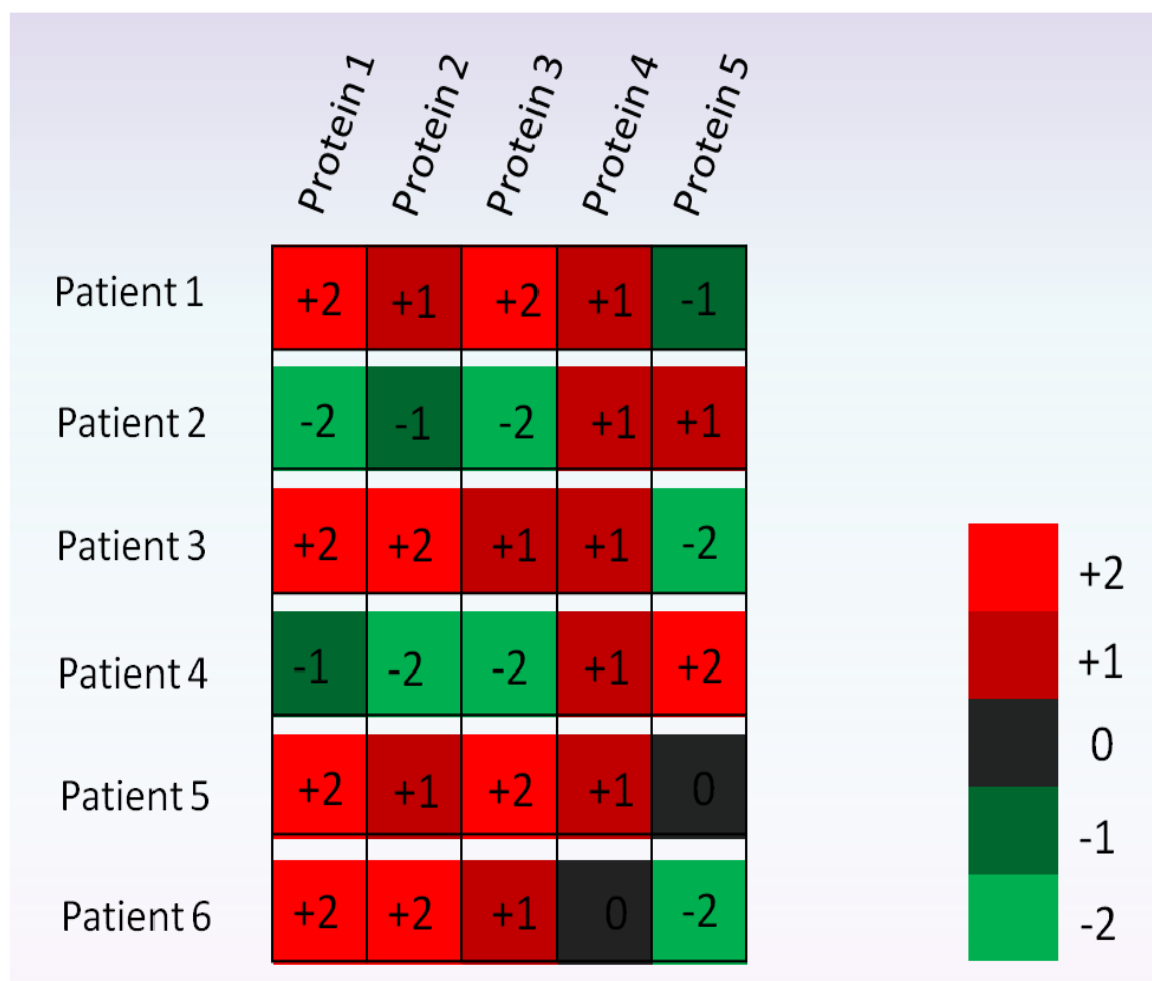
correlation as the distance measure. The parameters that were adjusted included indicating whether normalization would be performed on the proteins only, samples only, or both, and whether the Pearson Correlation would be centered or uncentered (See *Cluster 3.0* Manual for more information). A different .cdt file could be created for each of these permutations. Typically, we chose to normalize across both proteins and samples. This means that for both variables, all values in each row (or column) are multiplied by a scalar such that the sum of the squares of the values in each row (or column) is 1 (a separate scalar is computed for each row). The batch file had to be placed in the folder containing the test files (saved as text) created by the “ClusterPrep Software”, where it could be executed by double-clicking on its icon. An additional batch file was created that could then open each .cdt file in Java *TreeView*, adjust the contrast of the heat map, and save the heat map as a .png file within the same directory. Finally, a macro was created for Microsoft Powerpoint that would transfer and center each .png file in the given directory onto a separate slide in a Powerpoint Presentation.

4.2 Average-Linkage Hierarchical Clustering

The master dataset contains all protein intensities for all patient samples. The clustering algorithm groups the samples based on the similarities between their component protein intensities. To illustrate how this is done, let's say we were studying the plasma levels of 5 different proteins in 6 different patients, and we obtained the following intensity scores (where -2 is the lowest intensity and +2 is the highest):

	<i>Protein 1</i>	<i>Protein 2</i>	<i>Protein 3</i>	<i>Protein 4</i>	<i>Protein 5</i>
Patient 1	+2	+1	+2	+1	-1
Patient 2	-2	-1	-2	+1	+1
Patient 3	+2	+2	+1	+1	-2
Patient 4	-1	-2	-2	+1	+2
Patient 5	+2	+1	+2	+1	0
Patient 6	+2	+2	+1	0	-2

To better visualize this table of values, we can convert these intensity scores to colors. For example, higher intensity scores can be assigned as brighter red, lower intensity scores as brighter green, and middle intensities as black. This would lead to the following heat map:



By casually glancing at the color-coded rows, one can begin to group these patients according to similarities between their protein profiles. For example, Patient 1's protein profile looks most similar to that of Patient 5 (alternating bright and dark red for Proteins 1-4 followed by a lower intensity in Protein 5). Therefore, these two patients can be grouped together by branches intersecting at a node (**Figure 4.4**). Similarly, Patient 3's profile is almost identical to Patient 6's

profile (except for Protein 4), so these two patients can be coupled. Likewise, Patients 2 and 4 can be grouped together. Among these 3 pairs of patients, the first and second pairs most closely resemble each other in that they generally exhibit higher intensities for Proteins 1-4, and lower intensities for Protein 5. Therefore, these two pairs can be linked into a single cluster. Finally, this cluster is linked with the third pair, which is more distantly related as it has low intensities for Proteins 1-3 followed by higher intensities for Proteins 4 and 5. It is noteworthy that the lengths of the branches are set to the distance between the joined items. Therefore, more highly correlated patient samples are joined by shorter branches, whereas more distantly related patient samples are joined by longer branches (see **Figure 4.4** below).

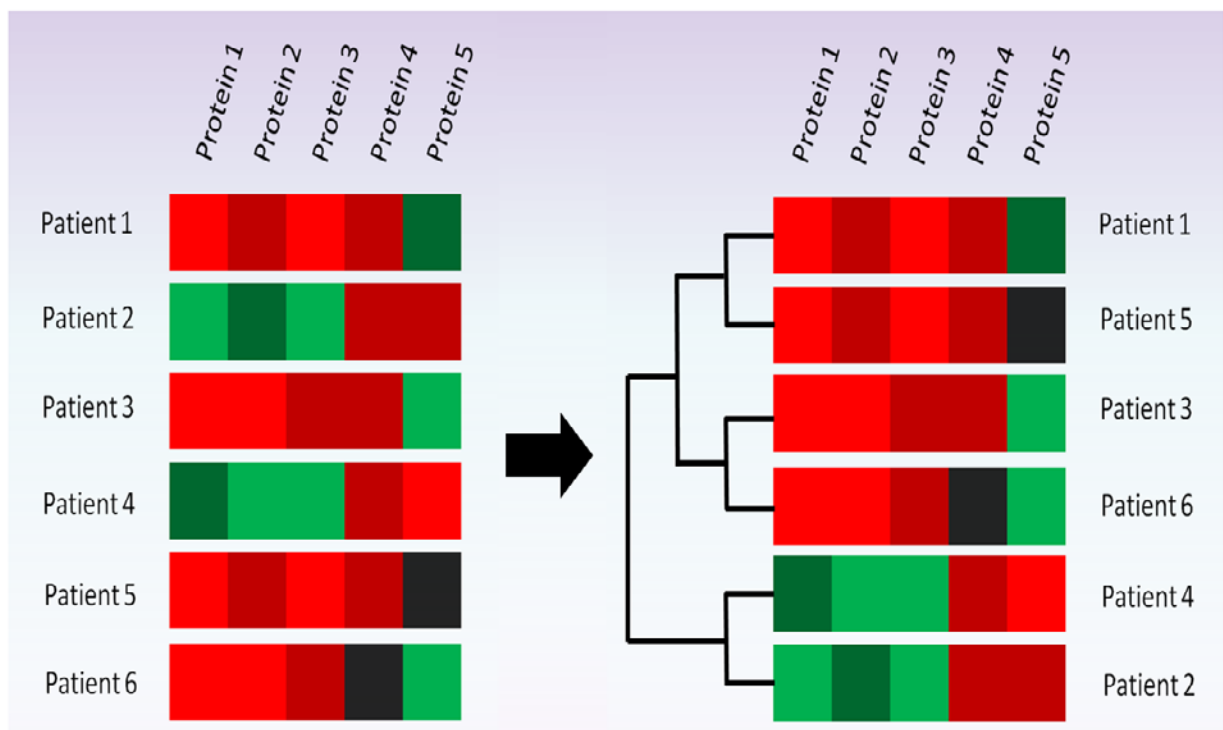


Figure 4.4 Illustration of clustering by visually grouping patient samples based on protein profile similarities.

While clustering of patient samples might be accomplished visually for small sample sizes and few assayed proteins, much larger datasets – like our 120 samples x 35 protein set – requires the clustering analysis to be done computationally. As such, the correlation between patients’ protein profiles must be determined mathematically. This is most commonly done using the Pearson correlation between the protein profiles, though other distance measures (Euclidean, city-block, and non-parametric measures) can also be used. The Pearson correlation r_{xy} between the protein profiles of two patient samples (X and Y) is given by:

$$r_{xy} = \frac{\sum_{i=1}^n (x_i - \bar{x})(y_i - \bar{y})}{(n - 1)s_x s_y}$$

where x_i and y_i are the fluorescent intensities of the i th protein, \bar{x} and \bar{y} are the mean protein intensities, and s_x and s_y are the corresponding standard deviations, of samples X and Y , respectively. These correlation coefficients are calculated and the resulting clustering of the data is accomplished using *Cluster 3.0*. In addition, for this study, the clustering algorithm was set to “Average-Linkage Clustering”, in which the distance between the two patient samples, X and Y , is the mean of all pairwise distances between their component protein intensities.

4.3 Test Sample Classification: “Guilt by Association”

As mentioned previously, test samples were classified based on the majority diagnosis of their nearest neighbors. To illustrate how this “guilt-by-association” technique works, let’s look at a few examples of test samples (“unknowns”) within clusters containing varying ratios of experimental (magenta) and control group (green) samples. In **Figure 4.5a**, we have a cluster containing two unknowns, two GBM patients, and two healthy controls. Since this cluster is

evenly split between experimental and control samples, no determination can be made about the classification of the two unknowns. As can be seen, in unbiased clusters such as these, the test sample classification is indeterminate. In this study, test samples with indeterminate classifications were excluded from further analysis. In **Figure 4.5b**, the unknown resides within a cluster in which there are 2 samples from patients with tumor growth and 3 samples from patients with no tumor growth (since their last MRI scan). Because this cluster has a slight “No Growth” bias, the unknown is classified as having no tumor growth (control group). However, the confidence level in this assignment is not very high since the number of “No Growth” samples barely exceeds the number of “Growth” samples. In **Figure 4.5c**, the unknown is situated within a cluster in which there are 4 “No Tumor Growth” samples and only one “Tumor Growth” sample. This is an example of a highly biased cluster, in which the unknown can be unambiguously assigned to the “No Tumor Growth” group with a relatively high level of confidence. Finally, in **Figure 4.5d**, the unknown is located within a homogeneous cluster (or “zone”) in which all members belong to the “Tumor Growth” group. Therefore, the test sample can be assigned to the “Tumor Growth” group with a very high level of confidence. In this study, the diagnostic accuracy of the “guilt-by-association” classification technique was assessed: i. for all unknowns (excluding indeterminates); ii. for the set of unknowns within highly biased and homogeneous clusters; and iii. for unknowns within homogeneous clusters only. The diagnostic accuracy of classifying test samples using “guilt-by-association” within each of these groups is discussed in **Sections 3.3** and **3.4**.

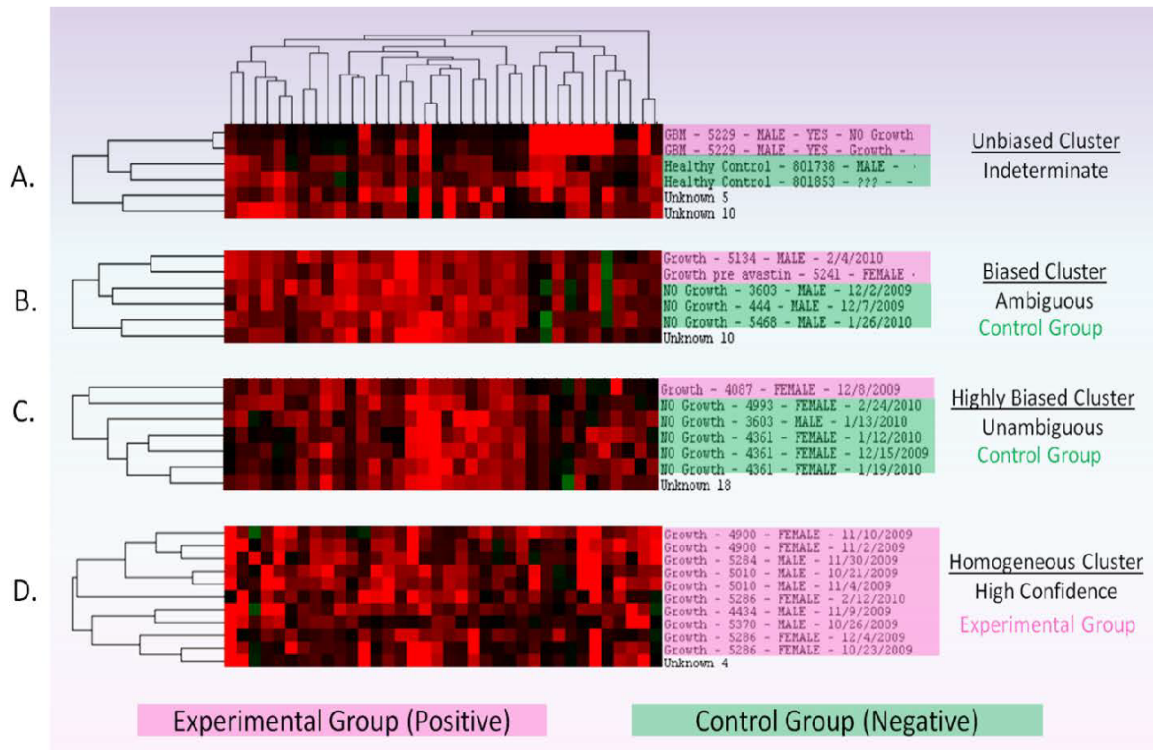


Figure 4.5 Classifying test samples via “Guilt by Association”: illustrative examples. In (a), the cluster is unbiased so the classification of the test samples (“unknowns”) is not possible. In (b), the cluster is biased, but only slightly, so the test sample is assigned with low confidence to the control group based on the majority diagnosis. In (c), the cluster is highly biased, so the test sample can be assigned unambiguously to the control group. In (d), the cluster is homogeneously comprised of patients from the experimental group, so the test sample is assigned to this group with very high confidence.

4.4 Appendix: Excel Macros for Data Analysis

4.4.1 Processing GenePix-Scanned Array Data to Create a Master Dataset

'The following subroutine ("RunProgramFor6x6Arrays") takes GenePix data
'(text format) that has been transferred to an excel file and formats it
'for statistical and graphical analysis. In particular, for each of the 12
'blocks (patient wells) in each file (slide), a new sheet is created. The
'oligo names are then tabulated on each sheet in exactly the order in
'which they appear on the slide. The spots with the highest intensity in
'the green channel (Cy3) are then assigned as oligo M (reference oligo),
'and the tabulated oligo order is then used to assign all other spots. The
'six repeats of each oligo/antibody spot (red channel - Cy5) are then
'organized into a list beneath each oligo name, and these columns are then
'sorted in alphabetical order by oligo name. The mean and standard
'deviation of six repeats are calculated for each oligo/antibody. Outliers
'are removed and the mean and standard deviation are then re-calculated.
'The mean values are then graphed (for each sheet), with error bars
'corresponding to the standard deviations. The mean intensity values for
'all proteins for each of the 12 sheets are then collated into one
'(additional) sheet. Furthermore, a baseline (or background) intensity is
'calculated for each graph (patient well) based on the average intensity
'of the 4 proteins with the lowest intensities. This baseline is added to
'each patient graph, and the baseline-subtracted protein intensity values
'are calculated. Each of the 12 patient graphs is then transferred to a
'separate slide within a Powerpoint file.

```
Sub RunProgramFor6x6Arrays()
```

```
    FormatSheetFor6x6Arrays
```

```
    'Formats the GenePix data in excel such that only the "Block", "Row",  
    '"Column", "Cy5 Mean", "Cy5 SD", "Cy3 Mean", and "Cy3 SD" Columns are  
    'shown (minus the headings). Due to variation in the GenePix output  
    'file, this step must sometimes be performed manually.
```

```
    NewSheetForEachBlock
```

```
    'Creates a new sheet for each block/well of patient data (for a total of  
    '12 sheets)
```

```
    WritesOligoOrderOnSheetFor6x6Array
```

```
    'Tabulates the order of the oligos exactly as they appear on the slide  
    'The following macros are run on all 12  
    'sheets(patient samples) in the excel file.
```

```
    PlaceOligoOrderOnEachSheet
```

```
    'Copies this table to all sheets of the excel file
```

```
    PlaceMOnEachSheetFor6x6Arrays
```

```
    'Finds the highest intensity green (Cy3) spots and assigns them  
    'as oligo M
```

OligoIDFor6x6ArrayForEachSheet

'Fills in the oligo ID for each spot using M as a reference and the tabulated oligo order.

OligoAndIntensityOnlyForEachSheet

'Result displays only the oligo ID and associated mean Cy5 (protein) intensity

CollatesIntensityValues4EachOligo4EachSheet

'Displays intensity values of all 6 spot repeats under each oligo ID.

AlphabeticalOrderForEachSheet

'Lists the columns in alphabetical order by oligo ID: i.e.
"A,B,C...Z,AA,BB,CC...

MeanAndStandardDeviationForEachSheet

'Displays the mean intensity and standard deviation for the 6 spot repeats of each oligo/protein

EliminatesLowValuesForEachSheet

'Eliminates intensity values less than a set threshold, typically ~100 for background.

FindsConsistencyAndThrowsOutSingleOutlierForEachSheet

'Throws out 3 of the 6 repeats for a given oligo/protein if the spots in the first round of array spotting are significantly brighter than those in the second round.
'Otherwise, throws out a single outlier (that minimizes the SD of the remaining repeats).

InsertGraphForEachSheet

'Inserts graph of the mean intensity values of each oligo/protein for each patient sample (sheet).

FormatChartForEachSheet

'Formats each graph to a set max x- and y-scale (typically 37 and 15000)

ErrorBarsForEachSheet

'Inserts up and down error bars with magnitude equal to the standard deviation.

CollateData

'Collates the mean intensities of proteins from all 12 patient samples onto a single sheet.

Baseline

'Uses average of 4 lowest protein intensity values as baseline, then subtracts all values by the baseline value.
'It then collates the background-subtracted data from all sheets on a single sheet.

TransferAllGraphsOnSheetsToPowerpoint

```
'Creates a new Powerpoint file and transfers all graphs on each sheet
to a separate slide
```

```
End Sub
```

Procedures Called by the “RunProgramFor6x6Arrays” Macro

```
Sub FormatSheetFor6x6Arrays()
```

```
'This subroutine trims the GenePix data file in excel so that it
'contains only the "Block", "Row", "Column", "Cy5 (635 nm wavelength)
'Mean", "Cy5 SD", "Cy3 (594 nm wavelength) Mean", and "Cy3 SD" Columns
'are shown. These row containing the headings is subsequently deleted.
'This subroutine runs properly if the "Block" heading appears in the
'first column when the file is transferred from GenePix to Excel.
'Otherwise, the file should be formatted manually.
```

```
    Rows("1:32").Select
    Selection.Delete Shift:=xlUp
    Columns("D:I").Select
    Selection.Delete Shift:=xlToLeft
    Columns("F:M").Select
    Selection.Delete Shift:=xlToLeft
    Columns("H:H").Select
    Columns("F:F").ColumnWidth = 8.89
    Columns("H:AQ").Select
    Selection.Delete Shift:=xlToLeft
    Range("J4").Select
    Columns("D:D").ColumnWidth = 9.33
    Rows("1:1").Select
    Selection.Delete Shift:=xlUp
```

```
End Sub
```

```
Sub NewSheetForEachBlock()
```

```
'This program creates 9 additional worksheets and fills each
'of the resulting 12 worksheets with data from one of the 12
'wells (corresponding to blocks on "Sheet1") on the slide

'The data must reside on "Sheet1" and the Workbook
'must start out with exactly 3 worksheets for this
'program to work properly
```

```
    ActiveWorkbook.Worksheets("Sheet1").Range("A1").Select

    For i = 1 To 9

        Sheets.Add After:=Sheets(Sheets.Count)

    Next i
```

```

For i = 1 To 11

    ActiveWorkbook.Worksheets("Sheet1").Select
    Range(Range("A1").Offset((217 * i) - i, 0), _
    Range("A1").Offset(i + (215 * (i + 1)), 6)).Select
    Selection.Cut
    ActiveWorkbook.Worksheets(i + 1).Select
    Range("A1").Select
    ActiveSheet.Paste

Next i

End Sub

```

```

Sub WritesOligoOrderOnSheetFor6x6Array()

'This program creates a 6x6 table of the 36 oligo names
'(at 'Sheet1, L7') in the row/column order in which they
'appear on the slide.

    ActiveWorkbook.Worksheets(1).Select

    Range("L7") = "U"
    Range("L8") = "II"
    Range("L9") = "QQ"
    Range("L10") = "WW"
    Range("L11") = "F"
    Range("L12") = "L"

    Range("M7") = "S"
    Range("M8") = "HH"
    Range("M9") = "PP"
    Range("M10") = "VV"
    Range("M11") = "E"
    Range("M12") = "K"

    Range("N7") = "P"
    Range("N8") = "CC"
    Range("N9") = "NN"
    Range("N10") = "UU"
    Range("N11") = "D"
    Range("N12") = "J"

    Range("O7") = "O"
    Range("O8") = "BB"
    Range("O9") = "MM"
    Range("O10") = "TT"
    Range("O11") = "C"
    Range("O12") = "I"

    Range("P7") = "N"
    Range("P8") = "AA"
    Range("P9") = "KK"
    Range("P10") = "SS"

```

```

Range("P11") = "B"
Range("P12") = "H"

Range("Q7") = "M"
Range("Q8") = "Z"
Range("Q9") = "JJ"
Range("Q10") = "RR"
Range("Q11") = "A"
Range("Q12") = "G"

```

End Sub

```
Sub PlaceOligoOrderOnEachSheet()
```

```

'This subroutine copies the table of ordered oligo names (created in
'WritesOligoOrderOnSheetFor6x6Array") and pastes it at "L7"
'on each of the 12 sheets

```

```
    For i = 1 To 11
```

```

        ActiveWorkbook.Worksheets("Sheet1").Select
        Range("K7:V12").Select
        Selection.Copy
        ActiveWorkbook.Worksheets(i + 1).Select
        Range("K7").Select
        ActiveSheet.Paste

```

```
    Next i
```

End Sub

```
Sub PlaceMOnEachSheetFor6x6Arrays()
```

```

'This subroutine finds the reference oligos M in all 12 sheets
'(by running "FindMFor6x6Arrays" in each sheet)

```

```
    For i = 1 To 12
```

```

        ActiveWorkbook.Worksheets(i).Select
        FindMFor6x6Arrays

```

```
    Next i
```

End Sub

```
Sub OligoIDFor6x6ArrayForEachSheet()
```

```

'This subroutine runs the "OligoIDFor6x6Array" program on each sheet/
'(block) to fill in the oligo name assignments for all spots in
'all blocks/sheets

```

```
    For i = 1 To 12
```

```

        ActiveWorkbook.Worksheets(i).Select

```



```

        OligoIDFor6x6Array

    Next i

End Sub

```

```

Sub OligoAndIntensityOnlyForEachSheet()

'This subroutine trims the data set to just the column of oligo names
'and their associated red-channel (Cy5) mean intensities for all
'12 sheets (blocks)

    For i = 1 To 12

        ActiveWorkbook.Worksheets(i).Select
        Range("A2").Select
        OligoAndIntensityOnly

    Next i

End Sub

```

```

Sub CollatesIntensityValues4EachOligo4EachSheet()

'This subroutine lists the intensity values for the six spot repeats
'of each oligo/antibody under the name of that oligo (for all 36
'oligos), and repeats this for all 12 sheets/blocks.

Dim i As Integer

    For i = 1 To 12

        ActiveWorkbook.Worksheets(i).Select
        CollatesIntensityValues4EachOligo

    Next i

End Sub

```

```

Sub AlphabeticalOrderForEachSheet()

'This subroutine places the columns of oligo intensity values in
'alphabetical order according to their oligo names:
'Importantly, it ensures that ordering is from A->Z, followed
'by AA->WW, as opposed to AA coming directly after A, and so forth.
'It does this by adding an extra worksheet, placing double-letter oligo
'names in that sheet, alphabetically ordering them, and then appending
'them with the ordered single-letter names in the previous
'sheet. The extra sheet is then deleted. This is repeated for all
'12 worksheets. A command prompt asks the user whether they want to
'delete the extra sheet (12 times). Click "Okay" all 12 times.

Dim i, j, StringLength As Integer
Dim myString As String

```

```

Dim ws As Worksheet

i = 0
j = 1

For j = 1 To ActiveWorkbook.Sheets.Count

    i = 0
    ActiveWorkbook.Worksheets(j).Select
    Sheets.Add After:=ActiveSheet
    ActiveSheet.Name = "TwoLetterOligos"
    ActiveWorkbook.Worksheets(j).Select
    Range("C1").Select

    Do
        myString = Range("C1").Offset(0, i).Text
        StringLength = Len(myString)

        If StringLength > 1 Then

            ActiveCell.EntireColumn.Select
            Selection.Cut
            ActiveWorkbook.Worksheets("TwoLetterOligos").Select
            Range("C1").Offset(0, i).Select
            ActiveSheet.Paste

        End If

        ActiveWorkbook.Worksheets(j).Select
        i = i + 1
        Range("C1").Offset(0, i).Select

    Loop Until i = 36

    ActiveWorkbook.Worksheets(j).Select
    DeleteEmptyColumns
    AlphabeticalOrder
    ActiveWorkbook.Worksheets("TwoLetterOligos").Select
    DeleteEmptyColumns
    AlphabeticalOrder

    Range("A1:Z7").Select
    Selection.Cut
    ActiveWorkbook.Worksheets(j).Select
    Range("A1").Select

    Do
        Range("A1").Offset(0, k).Select
        k = k + 1
    Loop Until IsEmpty(ActiveCell)

    ActiveSheet.Paste
    ActiveWorkbook.Worksheets("TwoLetterOligos").Delete
    k = 0

```

```

    Next j
End Sub

```

```

Sub MeanAndStandardDeviationForEachSheet()

'This subroutine outputs the mean and standard deviation of the intensity
'values for the six spot repeats for each oligo/antibody (beneath
'each list of intensity values). This is repeated for all 12 sheets.

Dim i As Integer

    For i = 1 To 12

        ActiveWorkbook.Worksheets(i).Select
        MeanAndStandardDeviation

    Next i

End Sub

```

```

Sub EliminatesLowValuesForEachSheet()

'This subroutine deletes all data values on a sheet that are less than
'100 Intensity Units. Typically, such low intensity values correspond
'to background, and suggest a defect in the spot loading or assay
'in that region. However, it could also suggest that the area was
'covered by PDMS and therefore unavailable for the assay.

    For i = 1 To 12

        ActiveWorkbook.Worksheets(i).Select
        EliminatesLowValues

    Next i

End Sub

```

```

Sub FindsConsistencyAndThrowsOutSingleOutlierForEachSheet()

'This subroutine carries out the two-mode outlier elimination of
'the "FindsConsistencyOrThrowsOutASingleOutlier" code, and repeats it
'for all 12 sheets/blocks

Dim ws As Worksheet

    For i = 1 To 12

        ActiveWorkbook.Worksheets(i).Select
        FindsConsistencyOrThrowsOutASingleOutlier

    Next i

```

End Sub

Sub InsertGraphForEachSheet()

'This subroutine graphs the mean intensity values for each column of
'oligo/protein intensities (on the same graph). As a result, the
'mean intensities for all proteins in a patient sample can quickly
'be evaluated visually. This is repeated for all 12 patient samples
'(worksheets) assayed on the slide.

For i = 1 To 12

 ActiveWorkbook.Worksheets(i).Select
 InsertGraph

Next i

End Sub

Sub FormatChartForEachSheet()

'This subroutine formats each chart to maximum scales on the x-
'and y- axes of 37 and 15000, respectively. Of course these
'values can be re-set to values of one's choosing. This is
'repeated for all 12 sheets/blocks.

For i = 1 To 12

 ActiveWorkbook.Worksheets(i).Select
 FormatChart

Next

End Sub

Sub ErrorBarsForEachSheet()

'This subroutine adds two-sided error bars (up- and down- magnitudes
'corresponding to standard deviations) to the graph of mean
'protein intensity values. This is repeated for all 12 sheets
'(all 12 patient graphs).

For i = 1 To 12

 ActiveWorkbook.Worksheets(i).Select
 ErrorBars

Next

End Sub

Sub InsertBaselineForEachSheet()

```
'This subroutine sorts the mean protein intensities from smallest to
'largest and places them in row 17. It then takes the average of
'the first 4 and 9 smallest values and places them in rows 18
'19, respectively, as well as adding them as baselines to the
'patient graph. This is based on the observation that the 4
'lowest values in a patient graph typically exhibit intensities
'equivalent to a negative control (non-specific IgG). This is
'repeated for all 12 sheets.
```

```
For i = 1 To 12
    ActiveWorkbook.Worksheets(i).Select
    InsertBaseline
Next
```

```
End Sub
```

```
Sub SubtractBaselineForEachSheet()
```

```
'This subroutine subtracts the baseline value from each of the 35
'mean protein intensities to yield a baseline-subtracted
'net mean protein intensity. It places these values in Row 24.
'This is repeated for all 12 sheets.
```

```
For i = 1 To 12
    ActiveWorkbook.Worksheets(i).Select
    SubtractBaseline
Next
```

```
End Sub
```

```
Sub CollateBackgroundSubtractedData()
```

```
'This subroutine collates all background subtracted mean protein
'intensity values from all 12 worksheets onto a single
'worksheet.
```

```
Sheets.Add After:=Sheets(Sheets.Count)
Range("A1") = "Collated Background Subtracted Data"
```

```
For i = 1 To 12

    ActiveWorkbook.Worksheets(i).Select
    Range("24:24").Select
    Selection.Copy
    ActiveWorkbook.Worksheets(Sheets.Count).Select
    Range("A2").Offset(i, 0).Select
    ActiveSheet.Paste
```

```
Next i
```

```
End Sub
```

```
Sub Baseline()
```

```
'This subroutine runs the "InsertBaselineForEachSheet",
'"SubtractBaselineForEachSheet", and
```

```
'"CollateBackgroundSubtractedData" subroutines
```

```
    InsertBaselineForEachSheet
    SubtractBaselineForEachSheet
    CollateBackgroundSubtractedData
```

```
End Sub
```

```
Sub CollateData()
```

```
'This subroutine collates the mean protein intensity values for all
'12 patients on a single sheet.
```

```
    Sheets.Add After:=Sheets(Sheets.Count)
    Range("A1") = "Collated Data"
    For i = 1 To 12
        ActiveWorkbook.Worksheets(i).Select
        Range("11:11").Select
        Selection.Copy
        ActiveWorkbook.Worksheets(Sheets.Count).Select
        Range("A2").Offset(i, 0).Select
        ActiveSheet.Paste
    Next i
```

```
End Sub
```

```
Sub CollateStandardDeviations()
```

```
'This subroutine collates the standard deviations for all 36
'proteins from all 12 worksheets (into a table of values)
'onto a single sheet. Each patient's values are listed in
'a separate row.
```

```
    Sheets.Add After:=Sheets(Sheets.Count)
    Range("A1") = "Collated Standard Deviations"
```

```
    For i = 1 To 12

        ActiveWorkbook.Worksheets(i).Select
        Range("12:12").Select
        Selection.Copy
        ActiveWorkbook.Worksheets(Sheets.Count).Select
        Range("A2").Offset(i, 0).Select
        ActiveSheet.Paste
```

```
    Next i
```

```
End Sub
```

```
'"TransferAllGraphsOnSheetsToPowerpoint" Procedure - See Appendix 4.8
```

Subroutines Called by the Above Procedures

```
Sub FindMFor6x6Arrays()
```

```
'This subroutine searches for intensity values in the green (Cy3) channel  
'that exceed 20000 AU, and labels them as the reference oligo M
```

```
Range("F1").Select
```

```
Do
```

```
  If ActiveCell.Value > 20000 Then  
    ActiveCell.Offset(0, 2).Value = "M"  
  End If
```

```
  ActiveCell.Offset(1, 0).Select
```

```
Loop Until ActiveCell.Offset(-1, 2).Value = "M"
```

```
  If ActiveCell.Offset(5, 0).Value > 20000 Then  
    ActiveCell.Offset(5, 2).Value = "M"  
  End If
```

```
End Sub
```

```
Sub OligoIDFor6x6Array()
```

```
'This subroutine searches for the three sets of oligo M pairs in a  
'worksheet, placed by the "FindMFor6x6Arrays" or the  
'"PlaceMOnEachSheetFor6x6Arrays" programs, and uses them as a reference to  
'guide the correct assignment of oligo names (using the table of ordered  
'oligo names created in "WritesOligoOrderOnSheetFor6x6Array" and/or  
'"PlaceOligoOrderOnEachSheet" to all other spots listed (by rows and  
'columns) in the block (on the sheet)
```

```
i, j, k, m = 0
```

```
Range("H1").Select
```

```
Do
```

```
  If ActiveCell.Value = "M" Then
```

```
    'Once this condition is satisfied, the index i gets the value of the  
    'column previous to M; this is useful because that's how many cells  
    'we need to count back to get to and select the first column within  
    'the row in which M resides; the index i's value does not change  
    'from this point until the the entire block is sequenced
```

```
    m = ActiveCell.Offset(0, -5).Value  
    'the index M gets the row number at which oligo M resides
```

```
    For j = 0 To 5
```

```
      'The index j allows us to select each cell in M's row, starting i
```

'cells above (or i cells to the left of M in the array sequence)

```
If Not IsEmpty(Range("Q7").Offset(0, -i + j).Cells) Then
```

```
ActiveCell.Offset(-i + j, 0).Value = _
Range("Q7").Offset(0, -i + j).Value
```

```
For k = 0 To 17
```

```
ActiveCell.Offset((-i + j) - 12 * (m - 1) + 6 * (2*k), _
0) = Range("Q7").Offset((6 - (m - k - 1)) Mod 6, _
-i + j).Value
ActiveCell.Offset((-i + j) - 12 * (m - 1) + _
6 * ((2 * k) + 1), 0) =
Range("Q7").Offset((6 - (m - k - 1)) Mod 6, -i+j).Value
```

```
Next k
```

'Since the oligo M resides in the mth row of the block (say 4th 'row), we need to offset by 3 rows to get us to the first row of 'the block. To get to the first row, we therefore need to 'multiply 3 (in this example) by the number of columns (12) in 'the array sequence (3x12=36). In other words, if we subtract 36 'from the i+j offset (from M's location in the oligo assignment 'column), we will hit the oligo in the array sequence at the same 'column offset position but in the first row of the block. The 'oligo at the same position in the next row down is assigned to 'the cell 12 cells below in the oligo assignment column and so 'forth (in multiples of 12) until that oligo position in all 18 'rows are accounted for. Notice that if oligo M is in the 4th row 'of of the block, the value of the oligo in the first row of the 'block (3 rows up) is the same as if you go (6-3) rows down in the 'array sequence table, hence the 6-(M-1). By taking the Mod 6 of 'this value, we ensure that we always stay within the confines of 'the 6-row array sequence table. The index k then allows us to 'scan through values in each row at the same column offset 'position.

```
Else
```

```
ActiveCell.Offset(-i+j, 0).Value = Range("Q7").Offset(0, _
-i + j - 6).Value
```

```
For k = 0 To 17
```

```
ActiveCell.Offset((-i + j) - 12 * (m - 1) + 6 * (2*k), _
0) = Range("Q7").Offset((6 - (m - k - 1)) Mod 6, _
-i + j - 6).Value
ActiveCell.Offset((-i + j) - 12 * (m - 1) + 6 * ((2*k) _
+ 1), 0) = Range("Q7").Offset((6 - (m - k - 1)) Mod 6, _
-i + j - 6).Value
```

```
Next k
```



```

        End If

    Next j

End If

ActiveCell.Offset(1, 0).Select
'This will continue to offset the selected cell until a cell
'containing M is reached

i = (i + 1) Mod 6
'The index i is the same as the block column value of the previous
'cell in the oligo assignment column

Loop Until ActiveCell.Offset(-1, 0).Value = "M"

End Sub

```

```

Sub OligoAndIntensityOnly()

'This subroutine trims the data set to just the column of oligo names
'and their associated red-channel (Cy5) mean intensities. The names
'are moved from column "H" to column "A". The intensities are
'moved from column "D" to column "B". All other data is deleted.

    Columns("H:H").Select
    Selection.Cut
    Columns("A:A").Select
    ActiveSheet.Paste

    Columns("D:D").Select
    Selection.Cut
    Columns("B:B").Select
    ActiveSheet.Paste

    Columns("C:Z").Select
    Selection.Delete

    Rows("1:1").Select
    Selection.Insert Shift:=xlDown, CopyOrigin:=xlFormatFromLeftOrAbove

End Sub

```

```

Sub CollatesIntensityValues4EachOligo()

'This subroutine lists the intensity values for the six spot repeats
'of each oligo/antibody under the name of that oligo (for all 36
'oligos).

Dim i, j, k As Integer
Dim myRange As Object

    ActiveWorkbook.ActiveSheet.Select

```

```

Range("A2").Select
i, k = 1
j = 0

Do
    CurrentCell = ActiveCell.Value
    Range("A2").Select
    Range("A2").Offset(-1, i + 1).Select

    Set myRange = Range("C1")
    ActiveCell.Value = CurrentCell
    Range("A2").Select

    Do
        If ActiveCell.Value = CurrentCell Then

            myRange.Offset(k, j).Value = ActiveCell.Offset(0, 1)
            k = k + 1
            Range(ActiveCell, ActiveCell.Offset(0, 1)).Delete Shift:=xlUp
            ActiveCell.Offset(-1, 0).Select

        End If

        ActiveCell.Offset(1, 0).Select

    Loop Until IsEmpty(ActiveCell)

    Range("A2").Select
    i = i + 1
    j = j + 1
    k = 1

Loop Until IsEmpty(ActiveCell)

End Sub

```

```

Sub AlphabeticalOrder()

'This subroutine sorts a list or table in alphabetical order by
'headings in the first row.

Range("A1:AZ10").Select
ActiveSheet.Sort.SortFields.Clear
ActiveSheet.Sort.SortFields.Add Key:=Range("A1:AZ1"), _
    SortOn:=xlSortOnValues, Order:=xlAscending, DataOption:=xlSortNormal

With ActiveSheet.Sort

    .SetRange Range("A1:AZ10")
    .Header = xlGuess
    .MatchCase = False
    .Orientation = xlLeftToRight
    .SortMethod = xlPinYin
    .Apply

```

```

End With

End Sub

```

```

Sub DeleteEmptyColumns()

'This subroutine deletes any empty columns from a worksheet
'containing the list of protein intensities arranged in
'alphabetical order by heading (protein name).

Dim i, j As Integer

i, j = 0

Range("A1").Select

Do
    If Not IsEmpty(ActiveCell) Then
        Range("A1").Offset(0, i).Select
        i = i + 1
    Else
        ActiveCell.EntireColumn.Delete
        j = j + 1
    End If

Loop Until i + j = 50

End Sub

```

```

Sub MeanAndStandardDeviation()

'This subroutine outputs the mean and standard deviation of the intensity
'values for the six spot repeats for each oligo/antibody (beneath
'each list of intensity values).

Range("A9").Select
ActiveCell.FormulaR1C1 = "=AVERAGE(R[-7]C:R[-2]C) "
Range("A9").Select
Selection.Copy
Range("A9:AJ9").Select
ActiveSheet.Paste

Range("A10").Select
Application.CutCopyMode = False
ActiveCell.FormulaR1C1 = "=STDEV(R[-8]C:R[-3]C) "
Range("A10").Select
Selection.Copy
Range("A10:AJ10").Select
ActiveSheet.Paste

End Sub

```

```

Sub EliminatesLowValues()

'This subroutine deletes all data values on a sheet that are less than
'100 Intensity Units. Typically, such low intensity values correspond
'to background, and suggest a defect in the spot loading or assay
'in that region. However, it could also suggest that the area was
'covered by PDMS and therefore unavailable for the assay.

Dim i, j As Integer

i , j = 0

Range("A2").Select

For j = 0 To 35

    Range("A2").Offset(0, j).Select

    For i = 0 To 5

        If ActiveCell.Value < 100 Then

            ActiveCell.ClearContents

        End If

        ActiveCell.Offset(1, 0).Select

    Next i

Next j

End Sub

```

```

Sub FindsConsistencyOrThrowsOutASingleOutlier()

'This subroutine eliminates outliers (from the list of intensity values
'of the six repeats for each oligo/column) from calculations of mean and
'standard deviation by one of two modes: 1) eliminating 3 outliers
'if the difference between intensity values in odd and even numbered
'rows is greater than 25%, or 2) if this is not the case, throwing
'out a single value that minimizes the standard deviation of the
'remaining values. The purpose of 1) is to circumvent an issue
'arising from array-spotting oligos in two separate runs: with half
'the repeats of each oligo spotted in the first run, and the other
'half spotted in the second run. By the time the second half are
'spotted, the humidity has caused the slides to become too resistant
'to oligo binding. As a result, the first set of (3) repeats yields
'significantly greater intensity values compared with the second set.
'In those cases, the second set of (3) repeats are eliminated from
'calculations of mean and standard deviation (and only the first set
'of (3) repeats is counted.

```

```

Dim i, j, k, x, y, z As Integer

i, j, k, x, y, z = 0

Range("A2").Select

For j = 0 To 35

    Range("A2").Offset(0, j).Select
    z = 0

    For i = 0 To 5

        If Not IsEmpty(ActiveCell) Then
            z = z + 1
        End If

        ActiveCell.Offset(1, 0).Select

    Next i

    If z > 3 Then

        Range("A2").Offset(0, j).Select
        i, x, y = 0

        Do While Not IsEmpty(ActiveCell)

            If (ActiveCell.Value - ActiveCell.Offset(1, _
                0).Value) / ActiveCell.Value > 0.25 Then
                x = x + 1
            End If

            y = y + 1
            ActiveCell.Offset(2, 0).Select

        Loop

        If IsEmpty(ActiveCell) And y < 3 Then

            x = 0
            y = 0
            Range("A2").Offset(1, j).Select

            For y = 0 To 2

                If Not IsEmpty(ActiveCell) Then

                    If (ActiveCell.Value - ActiveCell.Offset(-1, _
                        0).Value) / ActiveCell.Value > 0.25 Then
                        x = x + 1
                    End If
                End If
            Next y
        End If
    End If
End For

```

```

        End If

        ActiveCell.Offset(2, 0).Select

    Next y
End If

If x <> 3 Then

    x = 0
    y = 0
    Range("A2").Offset(0, j).Select

    Do While Not IsEmpty(ActiveCell)

        If (ActiveCell.Value - ActiveCell.Offset(1, _
            0).Value) / ActiveCell.Value < -0.25 Then

            x = x + 1

        End If

        y = y + 1
        ActiveCell.Offset(2, 0).Select

    Loop

    If IsEmpty(ActiveCell) And y < 3 Then

        x = 0
        y = 0

        Range("A2").Offset(1, j).Select

        For y = 0 To 2

            If Not IsEmpty(ActiveCell) Then

                If (ActiveCell.Value - ActiveCell.Offset(-1, _
                    0).Value) / ActiveCell.Value < -0.25 Then
                    x = x + 1
                End If

            End If

            ActiveCell.Offset(2, 0).Select

        Next y

    End If

End If

```

```

Range("A2").Offset(6, j).Select

If x = 3 Then

    ActiveCell.Offset(10, 0).FormulaR1C1 = "=Average(R[-16]C, _
        R[-14]C,R[-12]C) "
    ActiveCell.Offset(10, 0).Select
    Selection.Copy
    Selection.PasteSpecial Paste:=xlPasteValues, _
        Operation:=xlNone, SkipBlanks:=False, Transpose:=False
    ActiveCell.Offset(1, 0).Select
    ActiveCell.FormulaR1C1 = "=Average(R[-16]C,R[-14]C,R[-12]C) "
    ActiveCell.Select
    Selection.Copy
    Selection.PasteSpecial Paste:=xlPasteValues, _
        Operation:=xlNone, SkipBlanks:=False, Transpose:=False
    ActiveCell.Offset(-8, 0).FormulaR1C1 = "=Max(R[7]C, R[8]C) "
    ActiveCell.Offset(-8, 0).Select
    Selection.Copy
    Selection.PasteSpecial Paste:=xlPasteValues, _
        Operation:=xlNone, SkipBlanks:=False, Transpose:=False

    With ActiveCell.Font
        .Color = -16776961
        .TintAndShade = 0
    End With

    If ActiveCell.Value = ActiveCell.Offset(8, 0).Value Then
        ActiveCell.Offset(1, 0).FormulaR1C1 = "=stdev(R[-9]C, _
            R[-7]C,R[-5]C) "
        ActiveCell.Offset(1, 0).Select
        Selection.Copy
        Selection.PasteSpecial Paste:=xlPasteValues, _
            Operation:=xlNone, SkipBlanks:=False, Transpose:=False

        With ActiveCell.Font
            .Color = -16776961
            .TintAndShade = 0
        End With

        With Range("A2,A4,A6").Offset(1, j).Font
            .Color = -16776961
            .TintAndShade = 0
        End With

    End If

    If ActiveCell.Value = ActiveCell.Offset(7, 0).Value Then

        ActiveCell.Offset(1, 0).FormulaR1C1 = _
            "=stdev(R[10]C,R[-8]C,R[-6]C) "
        ActiveCell.Offset(1, 0).Select
        Selection.Copy

```

```

Selection.PasteSpecial Paste:=xlPasteValues, _
Operation:=xlNone, SkipBlanks:=False, Transpose:=False

With ActiveCell.Font
    .Color = -16776961
    .TintAndShade = 0
End With

With Range("A2,A4,A6").Offset(0, j).Font
    .Color = -16776961
    .TintAndShade = 0
End With

End If

Else

Range("A2").Offset(i, j).Select
i = 0

For i = 0 To 5

    Range("A2").Offset(i, j).Select
    Selection.Cut
    Range("A2").Offset(30, j).Select
    ActiveSheet.Paste
    Range("A2").Offset(i, j).Select
    ActiveCell.Offset(12, 0).FormulaR1C1 = "=stdev(R" & _
        2 & "C:R" & 7 & "C)"
    ActiveCell.Offset(12, 0).Select
    Selection.Copy
    Selection.PasteSpecial Paste:=xlPasteValues, _
        Operation:=xlNone, SkipBlanks:=False, Transpose:=False
    ActiveCell.Offset(10, 0).FormulaR1C1 = "=average(R"& _
        2 & "C:R" & 7 & "C)"
    ActiveCell.Offset(10, 0).Select
    Selection.Copy
    Selection.PasteSpecial Paste:=xlPasteValues, _
        Operation:=xlNone, SkipBlanks:=False, Transpose:=False
    Range("A2").Offset(30, j).Select
    Selection.Cut
    Range("A2").Offset(i, j).Select
    ActiveSheet.Paste

Next i

Range("A2").Offset(19, j).FormulaR1C1 = "=min(R" & _
    14 & "C:R" & 19 & "C)"
Range("A2").Offset(19, j).Select
Selection.Copy
Selection.PasteSpecial Paste:=xlPasteValues, _
    Operation:=xlNone, SkipBlanks:=False, Transpose:=False
Range("A14").Offset(0, j).Select

```



```

Do While ActiveCell.Value <> Range("A21").Offset(0, j).Value
    ActiveCell.Offset(1, 0).Select
Loop

Range("A12").Offset(0, j).Value = ActiveCell.Value

With Range("A12").Offset(0, j).Font
    .Color = -16776961
    .TintAndShade = 0
End With

Range("A11").Offset(0, j).Value = ActiveCell.Offset(10, _
0).Value

With Range("A11").Offset(0, j).Font
    .Color = -16776961
    .TintAndShade = 0
End With

ActiveCell.Offset(-12, 0).Select

With Selection.Font
    .Color = -16776961
    .TintAndShade = 0
End With

Range("A14:A30").Offset(0, j).Delete

End If

Range("A13:A20").Offset(0, j).Select
Selection.ClearContents

Else

Range("A2").Offset(9, j).FormulaR1C1 = "=average(R" & _
2 & "C:R" & 7 & "C)"

If z > 1 Then
    Range("A2").Offset(10, j).FormulaR1C1 = "=stdev(R" & 2 _
    & "C:R" & 7 & "C)"
End If

Range("11:12").Select
Selection.Copy
Selection.PasteSpecial Paste:=xlPasteValues, _
Operation:=xlNone, SkipBlanks:=False, Transpose:=False

With Selection.Font
    .Color = -16776961
    .TintAndShade = 0
End With

```

```

        End If

    Next j

End Sub

```

```

Sub InsertGraph()

'This subroutine graphs the mean intensity values for each column of
'oligo/protein intensities (on the same graph). As a result, the
'mean intensities for all proteins in a patient sample can quickly
'be evaluated visually.

    Rows("11:11").Select
    ActiveSheet.Shapes.AddChart.Select
    ActiveChart.SetSourceData Source:=ActiveSheet.Range("$11:$11")
    ActiveChart.ChartType = xlXYScatter
    ActiveChart.Axes(xlValue).Select
    ActiveChart.Axes(xlValue).MaximumScale = 60000

End Sub

```

```

Sub FormatChart()

'This subroutine formats each chart to maximum scales on the x-
'and y- axes of 37 and 15000, respectively. Of course these
'values can be re-set to values of one's choosing.

    ActiveSheet.ChartObjects(1).Activate

    If ActiveChart.HasLegend = True Then
        ActiveChart.Legend.Select
        Selection.Delete
    End If

    ActiveSheet.ChartObjects(1).Activate
    ActiveChart.Axes(xlCategory).Select
    ActiveChart.Axes(xlCategory).MinorUnit = 1
    ActiveChart.Axes(xlCategory).MajorUnit = 37
    Selection.MinorTickMark = xlInside
    ActiveChart.Axes(xlCategory).MaximumScale = 37
    ActiveChart.Axes(xlValue).Select
    ActiveChart.Axes(xlValue).MaximumScale = 15000

End Sub

```

```

Sub ErrorBars()

'This subroutine adds two-sided error bars (up- and down- magnitudes
'corresponding to standard deviations) to the graph of mean
'protein intensity values.

    ActiveWorkbook.ActiveSheet.Select

```

```

ActiveSheet.ChartObjects(1).Activate

With ActiveChart.SeriesCollection(1)
    .ErrorBar Direction:=xlY, Include:=xlBoth, _
    Type:=xlCustom, Amount:=ActiveSheet.Range("I2:I2"), _
    MinusValues:=ActiveSheet.Range("I2:I2")
End With

End Sub

```

```

Sub InsertBaseline()

'This subroutine sorts the mean protein intensities from smallest to
'largest and places them in row 17. It then takes the average of
'the first 4 and 9 smallest values and places them in rows 18
'19, respectively, as well as adding them as baselines to the
'patient graph. This is based on the observation that the 4
'lowest values in a patient graph typically exhibit intensities
'equivalent to a negative control (non-specific IgG).

Range("A11:AJ11").Select
Selection.Copy
Range("A17:AJ17").Select
ActiveSheet.Paste
Application.CutCopyMode = False
ActiveSheet.Sort.SortFields.Clear
ActiveSheet.Sort.SortFields.Add Key:=Range("A17:AJ17"), _
    SortOn:=xlSortOnValues, Order:=xlAscending, DataOption:=xlSortNormal

With ActiveSheet.Sort
    .SetRange Range("A17:AJ17")
    .Header = xlGuess
    .MatchCase = False
    .Orientation = xlLeftToRight
    .SortMethod = xlPinYin
    .Apply
End With

Range("A18").Select
ActiveCell.FormulaR1C1 = "=AVERAGE(R[-1]C,R[-1]C[4])"
Range("A19").Select
ActiveCell.FormulaR1C1 = "=AVERAGE(R[-2]C,R[-2]C[9])"
Range("A18:A19").Select
Selection.Copy
Range("A18:AJ19").Select
Selection.PasteSpecial Paste:=xlPasteValues, Operation:=xlNone, _
    SkipBlanks:=False, Transpose:=False
ActiveSheet.ChartObjects(1).Select
ActiveChart.SeriesCollection.NewSeries
ActiveChart.SeriesCollection(2).Name = """"Series2""""
ActiveChart.SeriesCollection(2).XValues = Range("$A$1:$AJ$1")
ActiveChart.SeriesCollection(2).Values = Range("$A$18:$AJ$18")
ActiveChart.SeriesCollection.NewSeries

```

```

ActiveChart.SeriesCollection(3).Name = ""Series3""
ActiveChart.SeriesCollection(3).XValues = Range("$A$1:$AJ$1")
ActiveChart.SeriesCollection(3).Values = Range("$A$19:$AJ$19")
ActiveChart.SeriesCollection(3).Select

With Selection
    .MarkerStyle = 3
    .MarkerSize = 2
End With

ActiveChart.SeriesCollection(2).Select

With Selection
    .MarkerStyle = 1
    .MarkerSize = 2
End With

End Sub

```

```

Sub SubtractBaseline()

'This subroutine subtracts the baseline value from each of the 35
'mean protein intensities to yield a baseline-subtracted
'net mean protein intensity. It places these values in Row 24.

Range("A24").Select
ActiveCell.FormulaR1C1 = "=R[-13]C-R[-6]C"
Range("A24").Select
Selection.Copy
Range("A24:AJ24").Select
ActiveSheet.Paste
Range("A24:AJ24").Select
Selection.Copy
Selection.PasteSpecial Paste:=xlPasteValues, Operation:=xlNone, _
    SkipBlanks:=False, Transpose:=False

End Sub

```

Collating Patient Data from all Patient Files

```

Sub CollateAllCollatedStandardDeviations()

'This subroutine collates the tables of standard deviations
'from each open workbook into a single sheet within a
'new workbook. For this code to run properly, the tables
'of standard deviations must be located on the last
'worksheet of all workbooks (typically "Sheet26").

Dim i As Integer
Dim NewWorkbook As Excel.Workbook

Set NewWorkbook = Application.Workbooks.Add

```

```

For i = 1 To Workbooks.Count

    Workbooks(i).Activate
    ActiveWorkbook.Worksheets(Sheets.Count).Activate
    ActiveSheet.Range("A3:AJ14").Select
    Selection.Copy
    NewWorkbook.Worksheets(1).Activate
    ActiveSheet.Range("A2").Offset(13 * (i - 1), 0).Select
    ActiveSheet.Paste

Next i

End Sub

```

```

Sub CollateAllCollatedNonBaselineSubMeans()

'This subroutine collates the tables of non-baseline subtracted mean
'protein intensities from each open workbook into a single sheet
'within a new workbook. For this code to run properly, the
'tables of mean values must be located on "Sheet25" within
'each workbook.

Dim i As Integer
Dim NewWorkbook As Excel.Workbook

Set NewWorkbook = Application.Workbooks.Add

For i = 1 To Workbooks.Count

    Workbooks(i).Activate
    ActiveWorkbook.Worksheets("Sheet25").Activate
    ActiveSheet.Range("A3:AJ14").Select
    Selection.Copy
    NewWorkbook.Worksheets(1).Activate
    ActiveSheet.Range("A2").Offset(13 * (i - 1), 0).Select
    ActiveSheet.Paste

Next i

End Sub

```

4.4.2 Graphing Patient Data from the Master Dataset

'The following macro sorts the master dataset by patient name followed by
 'blood collection date (such that all samples corresponding to each patient
 'are listed in chronological order by collection date). Depending on which
 'procedure is then used (see below for options), a variety of different
 'graphical analyses are enabled. For example, if one chooses the
 '"GraphEachSelection" procedure, the protein data for each row/sample will
 'be graphed separately (Fluorescent Intensity vs. Protein Identity).
 'Alternatively, if one chooses "GraphTimeCourseData3", each patient's time

'course data (Protein Intensity vs. Blood Collection Date) for eachprotein
'will be displayed on a single chart. (See below for description of other
'alternatives).

Sub RunGraphAllSelections()

```
SortbyDateForEachName
GraphEachSelection
    'Note: This line can be interchanged with GraphTimeCourseData,
    'GraphTimeCourseData2, or GraphTimeCourseData
FormatAllChartsOnSheet
TransferAllGraphsOnSheetsToPowerpoint
```

End Sub

Sub SortbyDateForEachName()

'This subroutine uses as input an excel file in which the patient
'last names have been sorted alphabetically in Column C (with
'first names in column D) and in which the blood collection
'dates are listed in column B. It then sorts the data set
'by date for each last name.

```
Dim i, j, k As Integer
Dim str As String
```

```
i, j = 0
k = 1
```

```
ExtractFirstWord
ActiveSheet.Cells(2, 3).Select
```

```
Do While Not IsEmpty(ActiveCell)
```

```
    i = 0
```

```
    Do While InStr(1, Trim(ActiveSheet.Cells(2 + i + j, 3).Value),
Trim(ActiveSheet.Cells(2 + j + (i + 1), 3).Value), vbTextCompare) <>
0 And InStr(1, Trim(ActiveSheet.Cells(2 + i + j, 57).Value),
Trim(ActiveSheet.Cells(2 + j + (i + 1), 57).Value), vbTextCompare) <>
0
```

```
        i = i + 1
```

```
    Loop
```

```
    i = i + 1
    ActiveWorkbook.ActiveSheet.Sort.SortFields.Clear
    ActiveWorkbook.ActiveSheet.Sort.SortFields.Add Key:=Range("B:B"),
    SortOn:=xlSortOnValues, Order:=xlAscending, DataOption:=xlSortNormal
```

```
    With ActiveWorkbook.ActiveSheet.Sort
        .SetRange Range(Cells((2 + j), 3), Cells((i + j + 1), _
```

```

        3)).EntireRow
        .Header = xlGuess
        .MatchCase = False
        .Orientation = xlTopToBottom
        .SortMethod = xlPinYin
        .Apply
    End With

    j = j + i
    ActiveSheet.Cells(2, 3).Offset(j, 0).Select

Loop

End Sub

```

```

Sub GraphEachSelection()

'This subroutine creates a graph of the mean intensities for all 36
'proteins in each row of patient data (each row of data corresponds
'to a different patient sample). For this code to run properly, the
'36 proteins must be located in columns U:BD. The graphs are labeled
'with the patient name, diagnosis, growth status, chemotherapy drug,
'and blood collection date. Unlike earlier version of this code, in
'this version, the range of cells to be graphed is selected before the
'chart is created, which speeds up the computing time considerably. In
'addition, the marker size is more compact, and the chart title acquires
'the same color as the excel row from which it's derived. As in past
'versions, the chart background color alternates from blue to
'gray between different patients.

Dim i, j, k, m As Integer

i , j = 0
k , m = 1

    ActiveSheet.Cells(2, 3).Select

    Do While Not IsEmpty(ActiveCell)

        i = 0

        Do

            Union(Range(Cells(1, 21), Cells(1, 56)), Range(Cells(i + j + 2, _
            21), Cells(i + j + 2, 56))).Select
            ActiveSheet.Shapes.AddChart.Select

            'Baseline-Subtracted Data
            ActiveChart.SetSourceData Source:=Union(Range(Cells(1, 21),
            Cells(1,56)), Range(Cells(i + j + 2, 21), Cells(i + j + 2, 56))),
            PlotBy:=xlRows

            'Non-Baseline-Subtracted Data (Option)...
```

```

'ActiveChart.SetSourceData Source:=Union(Range(Cells(1, 99),
Cells(1, 134)), Range(Cells(i + j + 2, 99), Cells(i + j + 2,
134))), PlotBy:=xlRows

With ActiveChart

    .ChartType = xlXYScatter
    .SetElement (msoElementChartTitleAboveChart)

    With .ChartTitle
        .Text = StringConcat(" ", ActiveSheet.Cells(2 + i + j, _
            4).Value, ActiveSheet.Cells(2 + i + j, 3).Value, "-", _
            ActiveSheet.Cells(2 + i + j, 12).Value, "-", _
            ActiveSheet.Cells(2 + i + j, 1).Value, Chr(10), _
            "Avastin", ActiveSheet.Cells(2 + i + j, 14).Value, _
            Chr(10), CStr (ActiveSheet.Cells(2 + i + j, 2).Value))
        .Font.Size = 10
        .Font.Name = "Calibri (Body)"
        .Font.Color = ActiveSheet.Cells(2 + i + j, _
            3).Font.Color
    End With

    With .Axes(xlCategory)
        .MinorUnit = 1
        .MajorUnit = 37
        .MaximumScale = 37
        .MinorTickMark = xlTickMarkInside
        .TickLabels.Delete
    End With

    With .Axes(xlValue)
        .MinorUnit = 1000
        .MajorUnit = 5000
        .MinimumScale = -5000
        .MaximumScale = 30000
    End With

    .HasLegend = False
    .SeriesCollection(1).ErrorBar Direction:=xlY, _
        Include:=xlBoth, Type:=xlCustom, _
        Amount:=ActiveSheet.Range(Cells(i + j + 2, 58), _
            Cells(i + j + 2, 93)), _
        MinusValues:=ActiveSheet.Range(Cells(i + j + 2, 58), _
            Cells(i + j + 2, 93))
    .SeriesCollection(1).ErrorBars.Border.ColorIndex = 5

End With

If k > 0 Then
    With ActiveChart.ChartArea.Fill
        .Visible = True
        .ForeColor.SchemeColor = 15
        .BackColor.SchemeColor = 17
        .TwoColorGradient msoGradientHorizontal, 1
    End With
End If

```



```

        End With

    End If

    If k < 0 Then
        With ActiveChart.ChartArea.Fill
            .Visible = True
            .ForeColor.SchemeColor = 41
            .BackColor.SchemeColor = 17
            .TwoColorGradient msoGradientHorizontal, 1
        End With
    End If

    For m = 1 To ActiveChart.SeriesCollection.Count
        ActiveChart.SeriesCollection(m).MarkerSize = 4
    Next m

    i = i + 1
    ActiveSheet.ChartObjects(i + j).Visible = False

    Loop Until InStr(1, ActiveSheet.Cells(1 + i + j, 3).Value, _
        ActiveSheet.Cells(1 + j + (i + 1), 3).Value, vbTextCompare) = 0 _
        And InStr(1, ActiveSheet.Cells(1 + i + j, 52).Value, _
        ActiveSheet.Cells(1 + j + (i + 1), 52).Value, vbTextCompare) = 0

    k = -1 * k
    j = j + i
    ActiveSheet.Cells(2, 3).Offset(j, 0).Select

Loop

For i = 1 To ActiveSheet.ChartObjects.Count
    ActiveSheet.ChartObjects(i).Visible = True
Next i

End Sub

```

```

Sub GraphTimeCourseData()

    'This subroutine graphs all 36 proteins/spot mean intensity values
    'at every collection time points for each patient. The chart
    'title consist of the patient number and diagnosis. Each
    'patient graph plots intensity as a function of protein ID.
    'The color-coding for the time points is given in the legend.

    Dim i, j, k As Integer
    Dim str As String
    Dim x As Object

    i, j = 0
    k = 1

    Application.ScreenUpdating = False

```

```

ExtractFirstWord
ActiveSheet.Cells(2, 3).Select

Do While Not IsEmpty(ActiveCell)

    i = 0

    Do While InStr(1, ActiveSheet.Cells(2 + i + j, 3).Value, _
        ActiveSheet.Cells(2 + j + (i + 1), 3).Value, vbTextCompare) <> 0 And
        InStr(1, ActiveSheet.Cells(2 + i + j, 57).Value, ActiveSheet.Cells(2
        + j + (i + 1), 57).Value, vbTextCompare) <> 0

        i = i + 1

    Loop

    i = i + 1
    Union(Range(Cells(1, 21), Cells(1, 56)), Range(Cells(j + 2, 21), _
        Cells(j + (i + 1), 56))).Select
    ActiveSheet.Shapes.AddChart.Select
    ActiveChart.SetSourceData Source:=Union(Range(Cells(1, 21), _
        Cells(1, 56)), Range(Cells(j + 2, 21), Cells(j + (i + 1), 56))), _
        PlotBy:=xlRows 'PlotBy:=xlColumns
    ActiveChart.ChartType = xlXYScatter
    ActiveChart.SetElement (msoElementChartTitleAboveChart)
        ActiveChart.ChartTitle.Text = StringConcat(" ", "Patient#", _
        ActiveSheet.Cells(2 + j, 5).Value, "-", ActiveSheet.Cells(2 + j, _
        12).Value)
    ActiveChart.ChartTitle.Font.Size = 10
    ActiveChart.ChartTitle.Font.Name = "Calibri (Body)"

    With ActiveChart.PlotArea
        .Width = 300
        .Height = 175
    End With

    With ActiveChart.Legend
        .Left = 300
        .Width = 50
        .Height = 300
        .Top = 35
        .Font.Size = 6
    End With

    k = 1

    For Each x In ActiveChart.SeriesCollection
        ActiveChart.SeriesCollection(k).Name = StringConcat(" - ", _
            CStr (ActiveSheet.Cells(1 + k + j, 2).Value), _
            ActiveSheet.Cells(1 + k + j, 14).Value, _
            ActiveSheet.Cells(1 + k + j, 1))
            x.MarkerSize = 4

        With ActiveChart.SeriesCollection(k)

```

```

.MarkerForegroundColorIndex = 2 + k
.MarkerBackgroundColorIndex = 2 + k
.ErrorBar Direction:=xlY, Include:=xlBoth, _
Type:=xlCustom, Amount:=ActiveSheet.Range(Cells(1 + k + j, _
58), Cells(1 + k + j, 93)), _
MinusValues:=ActiveSheet.Range(Cells(1 + k + j, 58), _
Cells(1 + k + j, 93))
.ErrorBars.Border.ColorIndex = 2 + k
End With

k = k + 1

Next x

With ActiveChart

With .Axes(xlCategory)
.MinorUnit = 1
.MajorUnit = 37
.MaximumScale = 37
.MinorTickMark = xlTickMarkInside
End With

With .Axes(xlValue)
.MinorUnit = 100
.MajorUnit = 1000
.MinimumScale = -1000
.MaximumScale = 10000
End With

End With

j = j + i
ActiveSheet.Cells(2, 3).Offset(j, 0).Select

Loop

Application.ScreenUpdating = True

End Sub

```

```

Sub GraphTimeCourseData2()

'This subroutine creates 6 graphs showing mean protein intensity vs.
'collection date for each set of 6 (out of the 36) distinct
'proteins/spots for each patient. The chart title consist of
'the patient number and diagnosis. The color-coding for the
'proteins is given in the legend.

Dim i, j, k, ProteinGroup, intIndex As Integer
Dim vntLabels As Variant
Dim str As String

```

```

Dim x As Object

i , j = 0
k = 1

Application.ScreenUpdating = False

ExtractFirstWord
ActiveSheet.Cells(2, 3).Select

Do While Not IsEmpty(ActiveCell)

    i = 0

    Do While InStr(1, ActiveSheet.Cells(2 + i + j, 3).Value, _
        ActiveSheet.Cells(2 + j + (i + 1), 3).Value, vbTextCompare) <> 0
        And InStr(1, ActiveSheet.Cells(2 + i + j, 57).Value, _
            ActiveSheet.Cells(2 + j + (i + 1), 57).Value, vbTextCompare) <> 0

            i = i + 1

    Loop

    i = i + 1

    For ProteinGroup = 0 To 5

        Union(Range(Cells(1, 21 + (6 * ProteinGroup)), Cells(1, _
            26 + (6 * ProteinGroup))), Range("B1"), Range(Cells(j + 2, 2), _
            Cells(j + (i + 1), 2)), Range(Cells(j + 2, _
            21 + (6 * ProteinGroup)), Cells(j + (i + 1), _
            26 + (6 * ProteinGroup)))).Select
        ActiveSheet.Shapes.AddChart.Select
        ActiveChart.SetSourceData Source:=Union(Range(Cells(1, _
            21 + (6 * ProteinGroup)), Cells(1, 26 + (6 * ProteinGroup))), _
            Range("B1"), Range(Cells(j + 2, 2), Cells(j + (i + 1), 2)), _
            Range(Cells(j + 2, 21 + (6 * ProteinGroup)), _
            Cells(j + (i + 1), 26 + (6 * ProteinGroup)))), PlotBy:=xlColumns
        ActiveChart.ChartType = xlLineMarkers
        ActiveChart.SetElement (msoElementChartTitleAboveChart)
        ActiveChart.ChartTitle.Text = StringConcat(" ", "Patient#", _
            ActiveSheet.Cells(2 + j, 5).Value, "-", _
            ActiveSheet.Cells(2 + j, 12).Value)

        ActiveChart.ChartTitle.Font.Size = 10
        ActiveChart.ChartTitle.Font.Name = "Calibri (Body)"

        With ActiveChart.PlotArea
            .Width = 300
            .Height = 175
            .Top = 20
        End With

        With ActiveChart.Legend

```

```

.Left = 320
.Width = 50
.Height = 70
.Top = 50
.Font.Size = 6
End With

k = 1

For Each x In ActiveChart.SeriesCollection

    x.MarkerSize = 4

    With ActiveChart.SeriesCollection(k)
        .MarkerForegroundColorIndex = 2 + k
        .MarkerBackgroundColorIndex = 2 + k
        .ErrorBar.Direction:=xlY, Include:=xlBoth, _
        Type:=xlCustom, Amount:=ActiveSheet.Range(Cells(j + 2, _
        58 + (6 * ProteinGroup) + k - 1), Cells(j + i + 1, _
        58 + (6 * ProteinGroup) + k - 1)), _
        MinusValues:=ActiveSheet.Range(Cells(j + 2, _
        58 + (6 * ProteinGroup) + k - 1), Cells(j + i + 1, _
        58 + (6 * ProteinGroup) + k - 1))
        .ErrorBars.Border.ColorIndex = 2 + k

        With .Border
            .ColorIndex = 2 + k
            .Weight = 2.5
            .LineStyle = xlContinuous
        End With

    End With
    k = k + 1

Next x

With ActiveChart

    With .Axes(xlCategory)
        With .TickLabels
            .Alignment = xlCenter
            .Offset = 100
            .Orientation = -40
        End With
    End With

    With .Axes(xlValue)
        .MinimumScale = -1000
        .MaximumScale = 5000
    End With

    With .Parent
        .Left = 100
    End With
End With

```

```

        .Width = 500
        .Top = 75
        .Height = 440
    End With

    End With

Next ProteinGroup

j = j + i
ActiveSheet.Cells(2, 3).Offset(j, 0).Select

Loop

Application.ScreenUpdating = True

End Sub

```

```

Sub GraphTimeCourseData3()

'This subroutine creates graphs showing mean protein intensity vs.
'collection date for the full set of 36 distinct proteins/
'spots for each patient. The chart title consist of the
'patient number and diagnosis. The color-coding for all
'proteins is given in the legend.

Dim i, j, k, ProteinGroup, intIndex As Integer
Dim vntLabels As Variant
Dim str As String
Dim x As Object

i = 0
j = 0
k = 1

Application.ScreenUpdating = False

ExtractFirstWord
ActiveSheet.Cells(2, 3).Select

Do While Not IsEmpty(ActiveCell)

    i = 0

    Do While InStr(1, ActiveSheet.Cells(2 + i + j, 3).Value, _
        ActiveSheet.Cells(2 + j + (i + 1), 3).Value, vbTextCompare) <> 0
        And InStr(1, ActiveSheet.Cells(2 + i + j, 57).Value, _
        ActiveSheet.Cells(2 + j + (i + 1), 57).Value, vbTextCompare) <> 0

        i = i + 1

    Loop

```

```

i = i + 1

Union(Range(Cells(1, 21), Cells(1, 56)), Range("B1"), _
Range(Cells(j + 2, 2), Cells(j + (i + 1), 2)), _
Range(Cells(j + 2, 21), Cells(j + (i + 1), 56))).Select
ActiveSheet.Shapes.AddChart.Select
ActiveChart.SetSourceData Source:=Union(Range(Cells(1, 21), _
Cells(1, 56)), Range("B1"), Range(Cells(j + 2, 2), _
Cells(j + (i + 1), 2)), _
Range(Cells(j + 2, 21), Cells(j + (i + 1), 56))), PlotBy:=xlColumns
'PlotBy:=xlRows
ActiveChart.ChartType = xlLineMarkers
ActiveChart.SetElement (msoElementChartTitleAboveChart)
ActiveChart.ChartTitle.Text = StringConcat(" ", "Patient#", _
ActiveSheet.Cells(2 + j, 5).Value, "-", ActiveSheet.Cells(2 + j, _
12).Value)
ActiveChart.ChartTitle.Font.Size = 10
ActiveChart.ChartTitle.Font.Name = "Calibri (Body)"

With ActiveChart.PlotArea
    .Width = 270
    .Height = 175
    .Top = 20
End With

With ActiveChart.Legend
    .Left = 350
    .Width = 50
    .Height = 330
    .Top = 10
    .Font.Size = 5
End With

k = 1

For Each x In ActiveChart.SeriesCollection

    x.MarkerSize = 4

    With ActiveChart.SeriesCollection(k)

        .MarkerForegroundColorIndex = 2 + k
        .MarkerBackgroundColorIndex = 2 + k
        .ErrorBar Direction:=xlY, Include:=xlBoth, Type:=xlCustom, _
Amount:=ActiveSheet.Range(Cells(j + 2, _
58 + (6 * ProteinGroup) + k - 1), Cells(j + i + 1, _
58 + (6 * ProteinGroup) + k - 1)), _
MinusValues:=ActiveSheet.Range(Cells(j + 2, _
58 + (6 * ProteinGroup) + k - 1), Cells(j + i + 1, _
58 + (6 * ProteinGroup) + k - 1))
        .ErrorBars.Border.ColorIndex = 2 + k

        With .Border
            .ColorIndex = 2 + k

```

```

        .Weight = 2.5
        .LineStyle = xlContinuous
    End With

    End With

    k = k + 1

Next x

With ActiveChart

    With .Axes(xlCategory)
        With .TickLabels
            .Alignment = xlCenter
            .Offset = 100
            .Orientation = -40
        End With
    End With

    With .Axes(xlValue)
        .MinimumScale = -5000
        .MaximumScale = 20000
    End With

    With .Parent
        .Left = 100
        .Width = 500
        .Top = 75
        .Height = 440
    End With

End With

j = j + i
ActiveSheet.Cells(2, 3).Offset(j, 0).Select

Loop

Application.ScreenUpdating = True

End Sub

```

```

Sub FormatAllChartsOnSheet()

'This subroutine formats each chart to maximum scales on the x-
'and y- axes of 37 and 10000, respectively. Of course these
'values can be re-set to values of one's choosing. Tick
'marks are placed on the inside of the x-axis. The legend is
'deleted. This is repeated for all 12 sheets/blocks.

For i = 1 To ActiveSheet.ChartObjects.Count

```



```

ActiveSheet.ChartObjects(i).Activate

If ActiveChart.HasLegend = True Then
    ActiveChart.Legend.Select
    Selection.Delete
End If

    ActiveSheet.ChartObjects(i).Activate
    ActiveChart.Axes(xlCategory).Select
    ActiveChart.Axes(xlCategory).MinorUnit = 1
    ActiveChart.Axes(xlCategory).MajorUnit = 37
    Selection.MinorTickMark = xlInside
    ActiveChart.Axes(xlCategory).MaximumScale = 37
    ActiveChart.Axes(xlValue).Select
    ActiveChart.Axes(xlValue).MaximumScale = 10000

Next i

End Sub

```

4.4.3 File Preparation for Cluster Analysis and Diagnostic Testing

'This section describes the "RunClusterPrep" macro, which formats and 'prepares cohort datasets for statistical analysis (by Excel and AnalyseIt) 'and for later cluster analysis (by Cluster 3.0). It also creates a 'worksheet for assessing the accuracy of classifying patients within 'hierarchical clusters based on "guilt-by-association".

'This macro begins by creating a new directory, "NewTrialFolder", which 'contains a number of excel files: "Format4Cluster", "Format4AnalyseIt", 'and "Diagnostic Performance" as well as a number of "Case" subfolders 'and an "All Text Files" folder.

'Sheet1 of "Format4Cluster" contains all the patient data (experimental 'and control) in a format that, once saved as a text document, can be used 'by the software Cluster 3.0. In particular, all the pertinent clinical 'information for each patient sample is listed in the first column. The 'first row contains only headers (i.e. protien names). The intersection 'between each row and column contains the intensity value of a single 'protein for a single patient sample.

'Sheet2 of "Format4Cluster" separates the experimental and control 'data and displays the calculated mean and median intensities for each 'protein in each group (both on the sheet and graphically). It also 'displays the differences (and root-mean-square distances) between 'experimental and control means and medians.

' "Format4AnalyseIt" contains the experimental and control group data for 'each protein in a format that can easily be transferred into and analyzed 'by "AnalyseIt", a statistical analysis add-in for Excel. Specifically,

'for each protein, the column of intensity values for experimental (red) and control(green) groups are situated adjacent to each other in table format. When the command button "Activate AnalyseIt-Dataset Defined" is clicked, an AnalyseIt excel file,"AnalyseIt-Dataset Defined" opens up into which the table of experimental and control columns for each protein can be transferred, one at a time, for a whole host of statistical tests available in the AnalyseIt toolbar. The macros "TransferNext2AnalyseIt" and "TransferPrevious2AnalyseIt" were written to allow one to toggle to the next or previous protein's data within the "Format4AnalyseIt" worksheet and instantly transfer that data table to the "AnalyseIt-Dataset Defined" worksheet by clicking on left or right arrows within the latter sheet.

'In addition, this subroutine facilitates diagnostic testing. It randomly assigns a certain number of patients (number specified by the user) to be "unknown" test samples. This can be repeated multiple times (i.e. multiple cases/tests), as specified by the user. Each of these case/test files (containing both data from known samples and randomly assigned unknowns) is saved into its own case subfolder within the "NewTrialFolder" directory. The resulting data sets are then saved as text documents (that are compatible with Cluster 3.0) in the "Text Files" folder within the case subfolder. Separately, all text files from all cases/tests are also saved in the "All Text Files" folder within the "NewTrialFolder" directory.

'The "Diagnostic Performance" file contains the actual diagnoses for all randomly assigned unknowns in all tests. However, these are hidden from view until the user has entered all their diagnostic predictions in the "Prediction" column and clicked on the "Diagnostic Performance!" command button. At that point, the predictions are scored and 2x2 contingency tables are created containing the numbers of true- and false- positives, and true- and false- negatives for each test. In addition, the specificity, sensitivity, and positive and negative predictive values for each test are indicated. Most importantly, also created is a table that contains the overall values (over all tests run) for all of these diagnostic parameters.

```
Public strNewFolderPathAndName As String
Public strFolderPathAndName As String
Public NumUnknowns As Integer    'Number of Unknowns for Test Set
Public NumProteins As Integer    'Number of Proteins to examine
Public TestNumber As Integer    'Number of Tests to Perform
Public CurrentTest As Integer
Public RangeA, RangeB As Range
```

'Note: NumUnknowns, NumProteins, and TestNumber are User-Defined

```
Sub RunClusterPrep()
```

```
Dim CurrentCasePathName As String
```

```
Application.ScreenUpdating = False
```

```

Format4Cluster
CreateNewDirectoryAndSaveAs 'Creates "NewTrialFolder" and Saves Excel
Files as "Format4Cluster" and "Format4AnalyseIt Files"
Workbooks("Format4AnalyseIt").Activate
Format4AnalyseIt 'Formats the excel file for use with the AnalyseIt
add-in in Excel"
ActiveWorkbook.Save
ActiveWorkbook.Close

'Create Case Folders for each File of Unknowns

For CurrentTest = 1 To TestNumber 'Test number is set on the user form

    CurrentCasePathName = strNewFolderPathAndName & "Case" & _
        CurrentTest & "\"
    Mkdir CurrentCasePathName
    Workbooks.Open FileName:=strNewFolderPathAndName & _
        "Format4Cluster.xlsx"
    Workbooks("Format4Cluster").Activate
    SelectRandomCases (CurrentCasePathName)
    'Selects NumUnknowns random cases as unknowns (where NumUnknowns is
    'defined by the user), creates new sheet for each unknown with the
    'set of knowns, and saves as notepad file in Case\Text Files folder

    ActiveWorkbook.SaveAs FileName:=CurrentCasePathName & "Case" & _
        CurrentTest & ".xlsx", FileFormat:=xlOpenXMLWorkbook, _
        CreateBackup:=False
    'Saves Excel File containing 20 unknowns, one in each sheet, in the
    appropriate Case Folder
    ActiveWorkbook.Close

Next CurrentTest

Workbooks.Open FileName:=strNewFolderPathAndName & _
    "Format4Cluster.xlsx"
ActiveWorkbook.Sheets(1).Cells.Copy
Sheets(2).Select
ActiveSheet.Paste
TwoCategoriesGraphMeansMedians2
'Outputs the mean and median intensity values for each protein within
experimental and control groups (and graphs them).
ActiveWorkbook.Save
ActiveWorkbook.Close
PrepareNewSheetForStatistics
'Creates and formats a sheet for diagnostic testing
ActiveWorkbook.Save
ActiveWorkbook.Close

Application.ScreenUpdating = True

End Sub

```

Procedures Called by the “RunClusterPrep” Macro

'The following subroutines are used directly by the "RunClusterPrep"
'subroutine: Format4Cluster, CreateNewDirectoryAndSaveAs, Format4AnalyseIt,
'SelectRandomCases, TwoCategoriesGraphMeansMedians2, and
'PrepareNewSheetForStatistics.

Sub Format4Cluster()

'To be compatible with Cluster 3.0, header/label information can be placed
'only in the first row and column, with all remaining rows and columns
'containing the mean fluorescent intensity values for each protein
'(columns) within each patient sample (rows). (See Cluster 3.0 Manual).

'This subroutine formats a patient data file such that all the relevant
'clinical parameters (namely, tumor growth status, IOIS#, gender, blood
'collection date, current diagnosis, and chemo drug treatment, are
'concatenated in a single cell (in the left-most column). All other
'patient information columns except the protein data values are deleted,
'such that the data set begins in the 2nd column of the worksheet. The
'first row of headers (protein/conjugate names) is maintained.

Dim i As Integer

i = 0

ActiveWorkbook.ActiveSheet.Activate
Union(Range("P:T"), Range("F:K"), Range("C:D"), Range("BE:EF")).Select
Selection.Delete

'Column A = Growth Status
'Column C = IOIS
'Column D = Current Pathology
'Column F = Avastin Status
'Column E = Gender
'Column B = Collection Date

Range("C2").Select

Do While Not IsEmpty(ActiveCell)
 Range("G2").Offset(i, 0).Value = StringConcat(" - ",
 Range("A2").Offset(i, 0).Value, Range("C2").Offset(i, 0).Value, _
 Range("D2").Offset(i, 0).Value, Range("F2").Offset(i, 0).Value, _
 Range("E2").Offset(i, 0).Value, _
 Range("B2").Offset(i, 0).Value)
 Range("C2").Offset(i, 0).Select
 i = i + 1

Loop

Range("G2").Offset(i - 1, 0).ClearContents
Range("A:F").Delete
Range("A1").ClearContents

End Sub

Sub CreateNewDirectoryAndSaveAs()

'This subroutine creates a new directory, "NewTrialFolder1", on the
'Desktop. If a folder named "NewTrialFolder1" already exists, the
'name of the new folder will be "NewTrialFolder2" and so forth.
'It then creates a subdirectory within this folder called
'"All Text Files". Finally, it saves the active excel workbook
'as "Format4Cluster" and "Format4AnalyseIt".

Dim strFolderPath As String

Dim n As Integer

n = 1

strFolderPathAndName = "C:\Documents and Settings\Heath Group\Desktop"
ActiveWorkbook.ActiveSheet.Cells.Copy
Workbooks.Add
ActiveSheet.Paste

strNewFolderPathAndName = strFolderPathAndName & "\NewTrialFolder\
strFolderPathAndName = strFolderPathAndName & "\NewTrialFolder"

Do While Dir(strNewFolderPathAndName, vbDirectory) <> ""
strNewFolderPathAndName = strFolderPathAndName & n & "\"
n = n + 1
Loop

MkDir strNewFolderPathAndName
'This is now the NewTrialFolder\
MkDir strNewFolderPathAndName & "All Text Files"
'This Folder Goes into the NewTrialFolder

ActiveWorkbook.SaveAs FileName:=strNewFolderPathAndName &
"Format4Cluster.xlsx", _
FileFormat:=xlOpenXMLWorkbook, CreateBackup:=False

ActiveWorkbook.SaveAs FileName:=strNewFolderPathAndName &
"Format4AnalyseIt.xlsx", _
FileFormat:=xlOpenXMLWorkbook, CreateBackup:=False

End Sub

Sub Format4AnalyseIt()

'This procedure formats the experimental and control group data within a
cohort dataset so that it can easily be transferred into and analyzed
'by "AnalyseIt", a statistical analysis add-in for Excel. Specifically,
'for each protein, the column of intensity values for experimental (red)
'and control (green) groups are situated adjacent to each other.

InsertColumns

```

ExtractFirstWord4AnalyseIt
ChangeCellFontColorAndPlaceColumnsAdjacently2
PaintColumnFontBlack
Range("A2").Select

Call GenButtons("Activate AnalyseIt-Dataset Defined", _
    "OpenAnalyseItDataSetDefined")

```

End Sub

```

Sub SelectRandomCases(ByVal FilePathName As String)

'This subroutine selects a number of cases randomly to serve as
'unknowns in a test set. The number of random cases (NumUnknowns)
'is assigned by the user in the user form. After a case is assigned
'as an unknown, it is moved to the bottom of the patient sample
'list. The next unknown is randomly assigned from the list of
'remaining samples (excluding the previously assigned unknowns).

'The subroutine then calls two functions: the first creates a
'separate worksheet for the set of patient samples with each
'unknown, as well as with all the unknowns combined. The second
'function saves each of these as a notepad file.

'The subroutine receives the file path name as an argument which it
'relays to the "SaveToNotepad" function.

Dim RandomIndex, i, m, n, UpperBound As Integer

i = 0

ActiveWorkbook.ActiveSheet.Activate

Range("A2").Offset(i, 0).Select
Do While Not IsEmpty(ActiveCell)
    Range("A2").Offset(i, 0).Select
    i = i + 1
Loop

UpperBound = i - 2

For m = 1 To NumUnknowns

    RandomIndex = Int((UpperBound - 1 + 1) * Rnd + 1)
    Range("A2").Offset(RandomIndex, 0).EntireRow.Select
    Selection.Copy
    Range("A2").Offset(i, 0).Select
    ActiveSheet.Paste
    Range("A2").Offset(RandomIndex, 0).EntireRow.Select
    Selection.Delete
    UpperBound = UpperBound - 1

Next m

```

```

Range(Range("A2").Offset(i - m + 1, 0), Range("A2").Offset(i - 1, _
    NumProteins)).Select
Selection.Font.ColorIndex = 3
Range("A2").Offset(i - m).EntireRow.Select
Selection.Delete

Range("B:B").Select
Selection.Insert
Range("A:A").Select
Selection.Copy
Range("B:B").Select
ActiveSheet.Paste

For n = 0 To NumUnknowns - 1
    Range("B2").Offset(i - m + n).Value = StringConcat(" ", "Unknown", _
        n + 1)
Next n

Call NewSheetForEachUnknown(i, m)
Call SaveToNotepad(i, FilePathName)

```

End Sub

Sub TwoCategoriesGraphMeansMedians2()

```

' This subroutine splits category 1 samples (typically
' experimental) and category 2 samples (typically control)
' by 8 empty rows. It then calculates the mean, median
' and standard deviation for all protein intensities in
' each category and lists them in blue under the last row
' of that category. Two graphs are created: one of the
' mean and the other of the median protein intensity values
' for the two categories (category 1 - red; category 2 -
' green). The difference between the category means and
' medians are also calculated for each protein. The
' absolute value is taken for each of these, and sorted
' from smallest to largest. In addition the root-mean-square
' is calculated for the set of means and the set of medians.

```

```

Dim i, j, k, m, n As Integer
Dim str1, str2 As String

```

```

i = 0
j = 0

```

```

ActiveWorkbook.ActiveSheet.Select

```

```

' Insert Column

```

```

ActiveSheet.Range("B2").Select
Selection.EntireColumn.Select
Selection.Insert Shift:=xlRight

```

```

' Extracts Words Before First Dash in label and places it in Column B
ExtractFirstWord4AnalyseIt

```

```

'ExtractWordsBeforeDash
Range("B2").Select
Selection.EntireColumn.Select
Selection.Copy
Selection.PasteSpecial Paste:=xlPasteValues, Operation:=xlNone, _
SkipBlanks:=False, Transpose:=False
Range("B2").Select

'Find First Row after Category 1 (by counting number of rows - i - in
'category 1)
Do While StrComp(ActiveSheet.Range("B2").Offset(i, 0).Value, _
    ActiveSheet.Range("B2").Offset(i + 1, 0).Value, vbTextCompare) = 0
    i = i + 1
Loop

Range("B2").Offset(i + 1, 0).EntireRow.Select

'Place m = 8 empty rows between Category 1 and Category 2
For m = 1 To 8
    Selection.Insert Shift:=xlDown
Next m

'Find First Row after Category 2 (by counting number of rows - j - in
'category 2)
Do While StrComp(ActiveSheet.Range("B2").Offset((i + 1) + m + j, _
    0).Value, ActiveSheet.Range("B2").Offset((i + 1) + m + (j + 1), _
    0).Value, vbTextCompare) = 0
    j = j + 1
Loop

ActiveSheet.Range("B2").Offset((i + 1) + m + j, 0).Select

'Get String Values (such as "Growth" vs. "No Growth")
str1 = Range("B2").Value
str2 = Range("B2").Offset(i + 1 + m).Value

'Delete Column Containing Extracted First Word
ActiveSheet.Range("B2").Select
Selection.EntireColumn.Select
Selection.Delete

'Average, Median, and Standard Deviation of All Values For Each Protein
'in Category 1
Range("B2").Offset(i + 1, 0).Select
ActiveCell.FormulaR1C1 = "=AVERAGE(R[" & (-i - 1) & "]C:R[-1]C)"
Range("A2").Offset(i + 1, 0).Value = StringConcat(" ", str1, _
"- Average")

Range("B2").Offset(i + 2, 0).Select
ActiveCell.FormulaR1C1 = "=MEDIAN(R[" & (-i - 2) & "]C:R[-2]C)"
Range("A2").Offset(i + 2, 0).Value = StringConcat(" ", str1, _
"- Median")

```



```

Range("B2").Offset(i + 3, 0).Select
ActiveCell.FormulaR1C1 = "=STDEV(R[" & (-i - 3) & "]C:R[-3]C)"
Range("A2").Offset(i + 3, 0).Value = StringConcat(" ", str1, _
"- Standard Deviation")

Range(Range("B2").Offset(i + 1, 0), Range("B2").Offset(i + 3, _
0)).Select
Selection.Copy
Range(Range("B2").Offset(i + 1, 0), Range("B2").Offset(i + 3, _
35)).Select
ActiveSheet.Paste
Selection.Font.ColorIndex = 33

'Average, Median, and Standard Deviation of All Values For Each Protein
'in Category 2
Range("B2").Offset((i + 1) + m + (j + 1), 0).Select
ActiveCell.FormulaR1C1 = "=AVERAGE(R[" & (-j - 2) & "]C:R[-1]C)"
Range("A2").Offset((i + 1) + m + (j + 1), 0).Value = _
StringConcat(" ", _ str2, "- Average")

Range("B2").Offset((i + 1) + m + (j + 2), 0).Select
ActiveCell.FormulaR1C1 = "=MEDIAN(R[" & (-j - 3) & "]C:R[-2]C)"
Range("A2").Offset((i + 1) + m + (j + 2), 0).Value = _
StringConcat(" ", str2, "- Median")

Range("B2").Offset((i + 1) + m + (j + 3), 0).Select
ActiveCell.FormulaR1C1 = "=STDEV(R[" & (-j - 4) & "]C:R[-3]C)"
Range("A2").Offset((i + 1) + m + (j + 3), 0).Value = _
StringConcat(" ", str2, "- Standard Deviation")

Range(Range("B2").Offset((i + 1) + m + (j + 1), 0),
Range("B2").Offset((i + 1) + m + (j + 3), 0)).Select
Selection.Copy
Range(Range("B2").Offset((i + 1) + m + (j + 1), 0), _
Range("B2").Offset((i + 1) + m + (j + 3), 35)).Select
ActiveSheet.Paste
Selection.Font.ColorIndex = 33

'Graph Average Protein Values for Both Categories
Union(Range(Cells(1, 1), Cells(1, 37)), Range(Cells(i + 3, 1), _
Cells(i + 3, 37)), Range(Cells((i + 1) + m + (j + 3), 1), _
Cells((i + 1) + m + (j + 3), 37))).Select
ActiveSheet.Shapes.AddChart.Select
ActiveChart.SetSourceData Source:=Union(Range(Cells(1, 1), _
Cells(1, 37)), Range(Cells(i + 3, 1), Cells(i + 3, 37)), _
Range(Cells((i + 1) + m + (j + 3), 1), Cells((i + 1) + m + (j + 3), _
37))), PlotBy:=xlRows

With ActiveChart
    .SeriesCollection(1).ErrorBar Direction:=xlY, Include:=xlBoth, _
    Type:=xlCustom, Amount:=ActiveSheet.Range(Cells(i + 5, 2), _
Cells(i + 5, 37)), MinusValues:=ActiveSheet.Range(Cells(i + 5, _
2), Cells(i + 5, 37))
    .SeriesCollection(2).ErrorBar Direction:=xlY, Include:=xlBoth, _

```

```

        Type:=xlCustom, Amount:=ActiveSheet.Range(Cells((i + 5) + m + _
        (j + 1), 2), Cells((i + 5) + m + (j + 1), 37)), _
        MinusValues:=ActiveSheet.Range(Cells((i + 5) + m + (j + 1), 2), _
        Cells((i + 5) + m + (j + 1), 37))
End With

FormatActiveChart
ActiveChart.ChartTitle.Text = StringConcat(" ", Range("A1").Value, _
str1, "vs.", str2, "- Means")

'Graph Median Protein Values for Both Categories
Union(Range(Cells(1, 1), Cells(1, 37)), Range(Cells(i + 4, 1), _
Cells(i + 4, 37)), Range(Cells((i + 4) + m + (j + 1), 1), _
Cells((i + 4) + m + (j + 1), 37))).Select
ActiveSheet.Shapes.AddChart.Select
ActiveChart.SetSourceData Source:=Union(Range(Cells(1, 1), Cells(1, _
37)), Range(Cells(i + 4, 1), Cells(i + 4, 37)), _
Range(Cells((i + 4) + m + (j + 1), 1), Cells((i + 4) + m + (j + 1), _
37))), PlotBy:=xlRows

With ActiveChart
    .SeriesCollection(1).ErrorBar Direction:=xlY, Include:=xlBoth, _
    Type:=xlCustom, Amount:=ActiveSheet.Range(Cells(i + 5, 2), _
    Cells(i + 5, 37)), MinusValues:=ActiveSheet.Range(Cells(i + 5, _
    2), Cells(i + 5, 37))
    .SeriesCollection(2).ErrorBar Direction:=xlY, Include:=xlBoth, _
    Type:=xlCustom, Amount:=ActiveSheet.Range(Cells((i + 5) + m + _
    (j + 1), 2), Cells((i + 5) + m + (j + 1), 37)), _
    MinusValues:=ActiveSheet.Range(Cells((i + 5) + m + (j + 1), _
    2), Cells((i + 5) + m + (j + 1), 37))
End With

FormatActiveChart
ActiveChart.ChartTitle.Text = StringConcat(" ", Range("A1").Value, _
str1, "vs.", str2, "- Medians")
n = (i + 1) + m + (j + 5) 'The A2 Offset index for "Mean Difference"

'Mean Difference
Range("A2").Offset(n, 0).Value = "Mean Difference"
Range(Range("B2").Offset(n, 0), Range("B2").Offset(n, _
35)).FormulaR1C1 = "=SUM(R[" & -5 - j - m & "]C, -R[-4]C)"
Range("A2").Offset(n + 3, 0).Value = "RMS of Means"
Range("B2").Offset(n + 3, 0).FormulaR1C1 = _
"=SQRT(SUMSQ(R[-3]C:R[-3]C[35])/COUNTA(R[-3]C:R[-3]C[35]))"

'Median Difference
Range("A2").Offset(n + 1, 0).Value = "Median Difference"
Range(Range("B2").Offset(n + 1, 0), Range("B2").Offset(n + 1, _
35)).FormulaR1C1 = "=SUM(R[" & -5 - j - m & "]C, -R[-4]C)"
Range("A2").Offset(n + 4, 0).Value = "RMS of Medians"
Range("B2").Offset(n + 4, 0).FormulaR1C1 = _
"=SQRT(SUMSQ(R[-3]C:R[-3]C[35])/COUNTA(R[-3]C:R[-3]C[35]))"

'Mean Difference Absolute Value

```

```

Range("A2").Offset(n + 7, 0).Value = "Abs(Mean Difference)"
Range(Range("B2").Offset(n + 7, 0), Range("B2").Offset(n + 7, _
35)).FormulaR1C1 = "=ABS(R[-7]C)"

'Median Difference Absolute Value
Range("A2").Offset(n + 8, 0).Value = "Abs(Median Difference)"
Range(Range("B2").Offset(n + 8, 0), Range("B2").Offset(n + 8, _
35)).FormulaR1C1 = "=ABS(R[-7]C)"

'Sort Mean Difference Abs
Range(Range("B2").Offset(n + 7, 0), Range("B2").Offset(n + 7, _
35)).Select
Selection.Copy

Range("B2").Offset(n + 10, 0).Select
Selection.PasteSpecial Paste:=xlPasteValues, Operation:=xlNone, _
SkipBlanks:=False, Transpose:=False

Range(Range("B2").Offset(n + 10, 0), Range("B2").Offset(n + 10, _
35)).Select
ActiveSheet.Sort.SortFields.Clear
ActiveSheet.Sort.SortFields.Add Key:=Range(Range("B2").Offset(n + 10, _
0), Range("B2").Offset(n + 10, 35)), SortOn:=xlSortOnValues, _
Order:=xlAscending, DataOption:=xlSortNormal

With ActiveSheet.Sort
    .SetRange Range(Range("B2").Offset(n + 10, 0), _
Range("B2").Offset(n + 10, 35))
    .Header = xlGuess
    .MatchCase = False
    .Orientation = xlLeftToRight
    .SortMethod = xlPinYin
    .Apply
End With

Range("A2").Offset(n + 10, 0).Value = "Sorted-Abs(Mean Difference)"

'Sort Median Difference Abs
Range(Range("B2").Offset(n + 8, 0), Range("B2").Offset(n + 8, _
35)).Select

Selection.Copy
Range("B2").Offset(n + 11, 0).Select
Selection.PasteSpecial Paste:=xlPasteValues, Operation:=xlNone, _
SkipBlanks:=False, Transpose:=False

Range(Range("B2").Offset(n + 11, 0), Range("B2").Offset(n + 11, _
35)).Select
ActiveSheet.Sort.SortFields.Clear
ActiveSheet.Sort.SortFields.Add Key:=Range(Range("B2").Offset(n + 11, _
0), Range("B2").Offset(n + 11, 35)), SortOn:=xlSortOnValues, _
Order:=xlAscending, DataOption:=xlSortNormal
With ActiveSheet.Sort

```

```

        .SetRange Range(Range("B2").Offset(n + 11, 0), _
        Range("B2").Offset(n + 11, 35))
        .Header = xlGuess
        .MatchCase = False
        .Orientation = xlLeftToRight
        .SortMethod = xlPinYin
        .Apply
    End With

    Range("A2").Offset(n + 11, 0).Value = "Sorted-Abs(Median Difference)"

End Sub

```

```

Sub PrepareNewSheetForStatistics()

'This subroutine creates a new excel workbook, saves it as
'Diagnostic Performance" and formats it for running diagnostic
'tests. Columns of Actual and Predicted diagnoses are located in
'columns A and C respectively. A heading is created for each test
'number (the number of tests/runs was specified earlier by the
'user). For each test, the excel case file in the corresponding
'case folder is automatically opened, and the "unknowns" (patient
'samples randomly assigned to be unknown test samples) are copied
'and pasted beneath the appropriate test heading in the "Diagnostic
'Performance" file. For each unknown, the first word (corresponding
'to the diagnosis) is extracted and placed in the adjacent cell in
'Column B. These cells are then hidden so that the tester cannot
'reference them (cheat) when assigning predicted diagnoses. A
'command button called "Diagnostic Performance!" is created,
'which runs the "CalculateStatistics" subroutine when clicked.

Dim i, m, Test As Integer
Dim RangeA, RangeB As Range
'RangeA and RangeB are First and Last Cell (respectively) of Each Test

    strNewFolderPathAndName = "C:\Documents and Settings\Heath _
    Group\Desktop\"
    'ActiveWorkbook.ActiveSheet.Activate
    Workbooks.Add
    FileName4Paste = strNewFolderPathAndName & _
    "Diagnostic Performance.xlsx"
    ActiveWorkbook.SaveAs FileName:=FileName4Paste, _
    FileFormat:=xlOpenXMLWorkbook, CreateBackup:=False

    Range("A1").Value = Category1Name & " vs."
    Range("A1").Font.Bold = True

    If StrComp(Category2Name, "no", vbTextCompare) = 0 Then
        Range("B1").Value = Category2Name & " " & Category1Name
    Else
        Range("B1").Value = Category2Name
        'Should have global variable for Cat2 Name
    End If

```

```

Range("B1").Font.Bold = True

Test = 1
Set RangeA = Range("A1").Offset((NumUnknowns + 2) * (Test - 1) + 12, 0)

With RangeA.Offset(-3, 0)
    .Value = "Actual"
    .Font.Bold = True
End With

With RangeA.Offset(-3, 2)
    .Value = "Predicted"
    .Font.Bold = True
End With

For Test = 1 To TestNumber

    Set RangeA = Range("A1").Offset((NumUnknowns + 2) * (Test - 1) + _
        12, 0)
    Set RangeB = RangeA.Offset((NumUnknowns - 1), 0)
    CurrentCasePathName = strNewFolderPathAndName & "Case" & Test & "\"
    FileName4Copy = CurrentCasePathName & "Case" & Test & ".xlsx"
    Workbooks.Open FileName:=FileName4Copy
    ActiveWorkbook.Sheets(1).Activate
    Range(Range("A2").Offset((NumRows - 1) - NumUnknowns, 0), _
        Range("A2").Offset(NumRows - 2, 0)).Select
    Selection.Copy
    ActiveWorkbook.Close
    Workbooks("Diagnostic Performance").Activate
    RangeA.Offset(-1, 0).Value = "Test" & Test
    RangeA.Offset(-1, 0).Font.Underline = True
    RangeA.Offset(-1, 2).Value = "Test" & Test
    RangeA.Offset(-1, 2).Font.Underline = True
    RangeA.Select
    ActiveSheet.Paste
    RangeA.Offset(0, 1).Select
    ActiveCell.FormulaR1C1 = "=LEFT(RC[-1],FIND("" - "" ,RC[-1])-1)"
    Selection.Copy
    Range(RangeA.Offset(0, 1), RangeB.Offset(0, 1)).Select
    ActiveSheet.Paste
    Range(RangeA, RangeB.Offset(0, 1)).NumberFormat = ";;;;"

Next Test

Range("B:B").Copy
Range("B:B").PasteSpecial Paste:=xlPasteValues, Operation:=xlNone, _
    SkipBlanks:=False, Transpose:=False
Workbooks("Diagnostic Performance").Activate
Range("A3").Value = TestNumber & " Tests"
Range("A4").Value = NumUnknowns & " Unknowns Each"

Call GenButtons("Diagnostic Performance!", "CalculateStatistics")

End Sub

```

Subroutines Called by the Above Procedures

```
'The following subroutines are used directly by the Format4AnalyseIt
'subroutine: InsertColumns, ExtractFirstWord4AnalyseIt,
'ChangeCellFontColorAndPlaceColumnsAdjacently2, PaintColumnFontBlack
'GenButtons.
```

```
Sub InsertColumns()
```

```
'This subroutine inserts two blank columns between columns
'of protein intensity values.
```

```
    ActiveSheet.Range("B2").Select
```

```
    Do While Not IsEmpty(ActiveCell)
        Selection.EntireColumn.Select
        Selection.Insert Shift:=xlRight
        Selection.Insert Shift:=xlRight
        ActiveCell.Offset(0, 3).Select
    Loop
```

```
    ActiveSheet.Range("C2").Select
```

```
    Do While Not IsEmpty(ActiveCell.Offset(0, 1))
        Selection.EntireColumn.Select
        Selection.Font.ColorIndex = 0
        ActiveCell.Offset(0, 3).Select
    Loop
```

```
End Sub
```

```
Sub ExtractFirstWord4AnalyseIt()
```

```
'This subroutine extracts the first word of each cell
'in column A and places it in the adjacent cell in
'column B.
```

```
NumRows = 1
```

```
    Range("A2").Select
```

```
    Do While Not IsEmpty(ActiveCell)
        ActiveCell.Offset(1, 0).Select
        NumRows = NumRows + 1
    Loop
```

```
    Range("B2").Select
    ActiveCell.FormulaR1C1 = "=LEFT(RC[-1],FIND("" "" ,RC[-1])-1) "
    Range("B2").Select
    Selection.Copy
    Range(Range("B2"), Range("B2").Offset(NumRows - 2, 0)).Select
    ActiveSheet.Paste
```

End Sub

Sub ChangeCellFontColorAndPlaceColumnsAdjacently2()

'This subroutine color-codes all rows corresponding to one
'diagnosis (typically, experimental group) red, and color-codes
'all rows corresponding to the other diagnosis (typically,
'control group) green. It then calls a function that places
'the column of protein values for the control group (green)
'adjacent to the columns of protein values for the
'experimental group (red). This allows the values from both
'groups to be tabulated in the correct format to be copied into
'an AnalyseIt Add-in file in Excel for statistical analysis.

Dim Cat1, Cat2 As Variant

Dim i, j As Integer

i, j = 0

Range("B2").Select
Selection.EntireColumn.Select
Selection.Copy
Selection.PasteSpecial Paste:=xlPasteValues, Operation:=xlNone, _
SkipBlanks:=False, Transpose:=False
Range("B2").Select

Do While StrComp(ActiveSheet.Range("B2").Offset(i, 0).Value, _
ActiveSheet.Range("B2").Offset(i + 1, 0).Value, vbTextCompare) = 0
i = i + 1

Loop

Range(Range("B2"), Range("B2").Offset(i, 0)).EntireRow.Select
Selection.Font.ColorIndex = 3
Cat1 = Split(Range("B2").Offset(i, 0).Value, " ")
Category1Name = Cat1(0)

Do While StrComp(ActiveSheet.Range("B2").Offset(i + 1 + j, 0).Value, _
ActiveSheet.Range("B2").Offset((i + 1) + (j + 1), 0).Value, _
vbTextCompare) = 0
j = j + 1

Loop

Range(Range("B2").Offset(i + 1, 0), Range("B2").Offset(j + i + 1, _
0)).EntireRow.Select
Selection.Font.ColorIndex = 4
Cat2 = Split(Range("B2").Offset(j + i + 1, 0).Value, " ")
Category2Name = Range("B2").Offset(j + i, 0).Value

Call PlaceColumnsAdjacently(i, j)
Range("D2").Select

End Sub

```

Sub PaintColumnFontBlack()

'This subroutine adjusts the color of empty columns
'(between green and red columns) to black.

    ActiveSheet.Range("C2").Select

    Do While Not IsEmpty(ActiveCell.Offset(0, 1))
        Selection.EntireColumn.Select
        Selection.Font.ColorIndex = 0
        ActiveCell.Offset(0, 3).Select
    Loop

```

End Sub

```

Sub GenButtons(ByVal strCaption As String, ByVal strAction As String)

'This function generates a command button by first
'receiving two string arguments. The first is the text
'that will appear on the command button. The second
'string argument is the subroutine the command button
'will run when clicked. The coordinates on the excel
'worksheet at which the command button is to be placed
'are also set.

Dim cBtn As Button

Set cBtn = ActiveSheet.Buttons.Add(0, 0, 175, 25)

cBtn.OnAction = strAction
cBtn.Caption = strCaption

End Sub

```

Subroutines Called by the Above Subroutines

```

'The following subroutine is called by the subroutine
'"ChangeCellFontColorAndPlaceColumnsAdjacently2".

Sub PlaceColumnsAdjacently(ByVal i As Integer, ByVal j As Integer)

'This subroutine places the column of protein intensity values for
'the control group (green) adjacent to the columns of protein values
'for the experimental group (red). This allows the values from both
'groups to be tabulated in the correct format to be copied into
'an AnalyseIt Add-in file in Excel for statistical analysis.

k = 0

    Range("D2").Select

```



```

Do While Not IsEmpty(Range("D2").Offset(0, 3 * k))
    Range(Range("D2").Offset(i + 1, 3 * k), Range("D2").Offset(j+i+1, _
        3 * k)).Select
    Selection.Cut
    Range("D2").Offset(0, 3 * k + 1).Select
    ActiveSheet.Paste
    Range("D1").Offset(0, 3 * k).Select
    Selection.Copy
    ActiveCell.Offset(0, 1).Select
    ActiveSheet.Paste
    k = k + 1
Loop

```

End Sub

'The following functions are used directly by the
 '"SelectRandomCases" subroutine: NewSheetForEachUnknown
 'and SaveToNotepad.

```
Sub NewSheetForEachUnknown(ByVal i As Integer, ByVal m As Integer)
```

'This function creates a separate worksheet containing
 'the set of 'known' patient samples with each unknown,
 'as well as one with all the unknowns combined (and labels
 'each sheet as such). The arguments i and m are integers
 'passed by the "SelectRandomCases" function. The integer
 'i refers to the row number of the last patient sample
 'on the worksheet. The integer m is a number one unit
 'greater than the number of unknowns.

```
Dim j As Integer
```

```

    ActiveWorkbook.Worksheets(1).Select
    Range(Range("B1"), Range("B1").Offset(i - 1, NumProteins)).Select
    Selection.Copy
    Sheets.Add After:=Sheets(Sheets.Count)
    ActiveSheet.Paste
    ActiveSheet.Name = "AllUnknowns"

```

```
For j = 1 To NumUnknowns
```

```

    ActiveWorkbook.Worksheets(1).Select
    Union(Range(Range("B1"), Range("B1").Offset(i - m, NumProteins)), _
        Range(Range("B1").Offset(i - m + j, 0), _
            Range("B1").Offset(i - m + j, NumProteins))).Select

    Selection.Copy
    Sheets.Add After:=Sheets(Sheets.Count)
    ActiveSheet.Paste
    ActiveSheet.Name = Range("A1").Offset(i - m + 1, 0).Value

```

```
Next j
```

```
Application.DisplayAlerts = False
```

```

Sheets("Sheet2").Delete
Sheets("Sheet3").Delete
Application.DisplayAlerts = True

```

End Sub

```
Sub SaveToNotepad(ByVal i As Integer, ByVal FilePathName As String)
```

```

'This subroutine gets the path and file name of an excel workbook
'in which the first worksheet contains a set of 'known' patient
'samples with the full set of randomly assigned unknowns. Each
'subsequent worksheet contains the set of 'known' patient samples
'with each unknown individually. Each of the worksheets is
'saved as a text file (for use directly with Cluster 3.0) in
'both the "All Text Files" subfolder within the "NewTrialFolder"
'directory, and in the "Text Files" folder within a "Case"
'subfolder (also in the "NewTrialFolder" directory).

```

```

Dim strPath, strFileName, strPathAndFilename As String
Dim n As Integer

```

```

'Note: FilePathName = CurrentCasePathName
'CurrentCasePathName = strNewFolderPathAndName & "Case" & Test & "\"

```

```

MkDir FilePathName & "Text Files"
'This folder goes into the Test/Case Folder

```

```

ActiveWorkbook.Worksheets(2).Select
ActiveWorkbook.SaveAs FileName:=FilePathName & "Text Files\Test" & _
    CurrentTest & "_AllUnknowns.txt", FileFormat:=xlText, _
    CreateBackup:=False
ActiveWorkbook.SaveAs FileName:=strNewFolderPathAndName & _
    "All Text Files\Test" & CurrentTest & "_AllUnknowns.txt", _
    FileFormat:=xlText, CreateBackup:=False

```

```
For n = 1 To NumUnknowns
```

```

    ActiveWorkbook.Worksheets(n + 2).Select
    ActiveWorkbook.SaveAs FileName:=FilePathName & "Text Files\Test" & _
        CurrentTest & "_Unknown" & n & ".txt", _
        FileFormat:=xlText, CreateBackup:=False
    ActiveWorkbook.SaveAs FileName:=strNewFolderPathAndName & _
        "All Text Files\Test" & CurrentTest & "_Unknown" & n & ".txt", _
        FileFormat:=xlText, CreateBackup:=False

```

```
Next n
```

End Sub

4.4.4 Assessing the Diagnostic Performance of “Guilty-by-Association” Classification of Test Samples within Hierarchical Clusters

```
Sub CalculateStatistics()
```

```
'This subroutine is activated when the "Diagnostic Performance!"
'command button is clicked in the "Diagnostic Performance" excel
'file. The subroutine compares the actual diagnoses within each
'test (column B) with the predicted diagnoses entered in by the
'tester (column C). If the predicted diagnosis is correct, a
'check mark is shown in the cell adjacent to the prediction
'(column D). Otherwise, a red x is shown, and an indication of
'whether the prediction was a false negative (FN) or false
'positive (FP) is given in column G. The samples within each
'test are numbered in column E.
```

```
'In addition, 2x2 contingency tables are drawn (with appropriate
'labels) for each test, indicating the numbers of true positives,
'true negatives, false positives, and false negatives. The sensitivity
'and specificity are given 2 columns to the right of the table,
'and the positive and negative predictive values are given
'two rows beneath the table. A contingency table indicating
'overall values (for all tests combined) is shown at the top.
```

```
Dim i, m, Test As Integer
Dim TruePositive, TrueNegative, FalsePositive, FalseNegative As Integer
Dim PPV, NPV, Sensitivity, Specificity As Double
Dim OverallTP, OverallTN, OverallFP, OverallFN As Integer
Dim OverallPPV, OverallNPV, OverallSensitivity, _
    OverallSpecificity As Double
Dim RangeAConst, RangeTable, OverallTable As Range
Dim strSplitCat1, strSplitCat2, strSplitActual, strSplitPredicted, tNum, _
    nUnk As Variant
```

```
OverallTP , OverallTN, OverallFP, OverallFN = 0
TruePositive , TrueNegative, FalsePositive, FalseNegative = 0
Sensitivity , Specificity, PPV, NPV = 0
```

```
ActiveWorkbook.ActiveSheet.Activate
```

```
Range("A:A").NumberFormat = "General"
Range("D:D").Font.Name = "Wingdings"
strSplitCat1 = Split(Range("A1").Value, " ")
strSplitCat2 = Split(Range("B1").Value, " ")
nUnk = Split(Range("A4").Value, " ")
NumUnknowns = CInt(nUnk(0))
tNum = Split(Range("A3"), " ")
TestNumber = CInt(tNum(0))
Test = 1
```

```
Set RangeAConst = Range("A1").Offset((NumUnknowns + 2) * (Test - 1) _
    + 12, 0)
```

```

For Test = 1 To TestNumber

    i = 0
    TruePositive, TrueNegative, FalsePositive, FalseNegative = 0
    Sensitivity, Specificity, PPV, NPV = 0

    Do
        Set RangeA = Range("A1").Offset((NumUnknowns + 2) * (Test - 1) _
            + 12, 0)
        Set RangeB = RangeA.Offset((NumUnknowns - 1), 0)

        RangeA.Offset(i, 0).Select
        strSplitActual = Split(RangeA.Offset(i, 1).Value, " ")
        strSplitPredicted = Split(RangeA.Offset(i, 2).Value, " ")

        If RangeA.Offset(i, 1).Value <> "" And RangeA.Offset(i, _
            2).Value <> "" Then

            If StrComp(strSplitActual(0), strSplitPredicted(0), _
                vbTextCompare) = 0 Then
                RangeA.Offset(i, 3).Value = "ü"

                If StrComp(strSplitActual(0), strSplitCat1(0), _
                    vbTextCompare) = 0 Then
                    'RangeA.Offset(i, 6).Value = "TP"
                    TruePositive = TruePositive + 1
                    OverallTP = OverallTP + 1
                End If

                If StrComp(strSplitActual(0), strSplitCat2(0), _
                    vbTextCompare) = 0 Then
                    'RangeA.Offset(i, 6).Value = "TN"
                    TrueNegative = TrueNegative + 1
                    OverallTN = OverallTN + 1
                End If
            Else
                RangeA.Offset(i, 3).Value = "û"
                RangeA.Offset(i, 3).Font.ColorIndex = 3

                If StrComp(strSplitActual(0), strSplitCat1(0), _
                    vbTextCompare) = 0 Then
                    RangeA.Offset(i, 6).Value = "FN"
                    FalseNegative = FalseNegative + 1
                    OverallFN = OverallFN + 1
                End If

                If StrComp(strSplitActual(0), strSplitCat2(0), _
                    vbTextCompare) = 0 Then
                    RangeA.Offset(i, 6).Value = "FP"
                    FalsePositive = FalsePositive + 1
                    OverallFP = OverallFP + 1
                End If
            End If
        End Do
    End For

```

```

        End If

    End If

    RangeA.Offset(i, 4).Value = i + 1
    i = i + 1

Loop Until IsEmpty(ActiveCell.Offset(1, 0))

If TruePositive + FalsePositive <> 0 Then
    PPV = Round((TruePositive / (TruePositive + _
        FalsePositive)) * 100, 1)
End If

If TrueNegative + FalseNegative <> 0 Then
    NPV = Round((TrueNegative / (TrueNegative + _
        FalseNegative)) * 100, 1)
End If

If TruePositive + FalseNegative <> 0 Then
    Sensitivity = Round((TruePositive / (TruePositive + _
        FalseNegative)) * 100, 1)
End If

If TrueNegative + FalsePositive <> 0 Then
    Specificity = Round((TrueNegative / (TrueNegative + _
        FalsePositive)) * 100, 1)
End If

Set RangeTable = RangeA.Offset(0, 7)

With RangeTable

    .Offset(0, 1) = "Positive"
    .Offset(0, 2) = "Negative"
    .Offset(1, 0) = Range("A1").Value
    .Offset(2, 0) = Range("B1").Value
    .Offset(1, 1) = "TP = " & TruePositive
    .Offset(2, 2) = "TN = " & TrueNegative
    .Offset(2, 1) = "FP = " & FalsePositive
    .Offset(1, 2) = "FN = " & FalseNegative
    .Offset(4, 1) = "PPV = " & PPV & "%"
    .Offset(4, 2) = "NPV = " & NPV & "%"
    .Offset(1, 4) = "Sensitivity = " & Sensitivity & "%"
    .Offset(2, 4) = "Specificity = " & Specificity & "%"
    .Offset(0, 7) = Sensitivity
    .Offset(0, 8) = Specificity
    .Offset(0, 9) = PPV
    .Offset(0, 10) = NPV

End With

RangeTable.Offset(1, 1).BorderAround ColorIndex:=0, Weight:=xlThin

```

```

RangeTable.Offset(1, 2).BorderAround ColorIndex:=0, Weight:=xlThin
RangeTable.Offset(2, 1).BorderAround ColorIndex:=0, Weight:=xlThin
RangeTable.Offset(2, 2).BorderAround ColorIndex:=0, Weight:=xlThin

Next Test

If OverallTP + OverallFP <> 0 Then
    OverallPPV = Round((OverallTP / (OverallTP + OverallFP)) * 100, 1)
End If

If OverallTN + OverallFN <> 0 Then
    OverallNPV = Round((OverallTN / (OverallTN + OverallFN)) * 100, 1)
End If

If OverallTP + OverallFN <> 0 Then
    OverallSensitivity = Round((OverallTP / (OverallTP + _
    OverallFN)) * 100, 1)
End If

If OverallTN + OverallFP <> 0 Then
    OverallSpecificity = Round((OverallTN / (OverallTN + _
    OverallFP)) * 100, 1)
End If

Set OverallTable = RangeAConst.Offset(-10, 7)

OverallTable.Select

With OverallTable
    .Offset(-1, 1) = "Overall"
    .Offset(-1, 2) = "Overall"
    .Offset(0, 1) = "Positive"
    .Offset(0, 2) = "Negative"
    .Offset(1, 0) = Range("A1").Value
    .Offset(2, 0) = Range("B1").Value
    .Offset(1, 1) = "TP = " & OverallTP
    .Offset(2, 2) = "TN = " & OverallTN
    .Offset(2, 1) = "FP = " & OverallFP
    .Offset(1, 2) = "FN = " & OverallFN
    .Offset(4, 1) = "PPV = " & OverallPPV & "%"
    .Offset(4, 2) = "NPV = " & OverallNPV & "%"
    .Offset(1, 4) = "Sensitivity = " & OverallSensitivity & "%"
    .Offset(2, 4) = "Specificity = " & OverallSpecificity & "%"
End With

Range(OverallTable.Offset(-1, 0), OverallTable.Offset(4, _
4)).Font.ColorIndex = 3
OverallTable.Offset(1, 1).BorderAround ColorIndex:=3, Weight:=xlThick
OverallTable.Offset(1, 2).BorderAround ColorIndex:=3, Weight:=xlThick
OverallTable.Offset(2, 1).BorderAround ColorIndex:=3, Weight:=xlThick
OverallTable.Offset(2, 2).BorderAround ColorIndex:=3, Weight:=xlThick

```

End Sub

4.4.5 Macros for Working with *AnalyseIt*

'The following set of macros facilitates straightforward transfer of cohort data (experimental and control data for each protein) into a pre-defined table format within an AnalyseIt template. They also facilitate the transfer of AnalyseIt graphs into Powerpoint.

```
Public Category1Name, Category2Name As String
```

'These are provided by the ChangeCellFontColorAndPlaceColumnsAdjacently subroutine

```
Public NumRows As Integer
```

```
Sub OpenAnalyseItDataSetDefined()
```

'This subroutine opens an Excel "AnalyseIt" workbook in which a table has been created containing two column headers: category 1 (experimental group) and category 2 (control group). The number and types of variables (i.e. categorical), and the type of dataset (list, one-way, or two-way table) have all been pre-defined to facilitate ease of use with AnalyseIt.

```
Dim PathName, FileName, FilePathAndName As String
Dim wBook As Workbook
```

```
'Set wBook = Workbooks("AnalyseIt-DatasetDefined")
```

```
    ActiveWorkbook.ActiveSheet.Activate
    PathName = ActiveWorkbook.Path
    FileName = ActiveWorkbook.Name
    FilePathAndName = PathName & "\" & FileName
    'If wBook Is Nothing Then
    Workbooks.Open FileName:="C:\Documents and Settings\Heath
    Group\Desktop\AnalyseIt-DatasetDefined.xlsm"
    'End If
    Workbooks("AnalyseIt-DatasetDefined").Sheets("Dataset").Activate
    Range("E4").Value = FilePathAndName
```

```
End Sub
```

```
Sub TransferNext2AnalyseIt()
```

```
Dim varFileName As Variant
Dim strFileName As String
Dim myString As String
```

```
On Error Resume Next
```

```
    Workbooks("AnalyseIt-DatasetDefined").Activate
    varFileName = Split(Range("E4").Value, "\")
```

```

strFileName = varFileName(UBound(varFileName))
Workbooks(strFileName).Activate

Do While ActiveCell.Font.ColorIndex <> 3
    ActiveCell.Offset(0, 1).Select
Loop

Do While ActiveCell.Font.ColorIndex = 3
    ActiveCell.Offset(-1, 0).Select
Loop

ActiveCell.Offset(1, 0).Select
ActiveCell.Offset(0, 3).Select
ExtractStringAfterDash (strFileName)
myString = ActiveCell.Offset(-1, 2).Value
Range(ActiveCell, ActiveCell.Offset(146, 1)).Select
Selection.Copy
Workbooks("AnalyseIt-DatasetDefined").Activate
ActiveSheet.Range("B6").Select
ActiveSheet.Paste
ActiveSheet.Range("B3").Value = StringConcat(" ", myString, "Levels
for", Range("B5").Value, "vs.", Range("C5").Value)

```

End Sub

```

Sub TransferPrevious2AnalyseIt()

Dim varFileName As Variant
Dim strFileName As String
Dim myString As String

On Error Resume Next

Workbooks("AnalyseIt-DatasetDefined").Activate
varFileName = Split(Range("E4").Value, "\")
strFileName = varFileName(UBound(varFileName))
Workbooks(strFileName).Activate

Do While ActiveCell.Font.ColorIndex <> 3
    ActiveCell.Offset(0, 1).Select
Loop

Do While ActiveCell.Font.ColorIndex = 3
    ActiveCell.Offset(-1, 0).Select
Loop

ActiveCell.Offset(1, 0).Select
ActiveCell.Offset(0, -3).Select
ExtractStringAfterDash (strFileName)
myString = ActiveCell.Offset(-1, 2).Value
Range(ActiveCell, ActiveCell.Offset(146, 1)).Select
Selection.Copy
Workbooks("AnalyseIt-DatasetDefined").Activate

```



```

ActiveSheet.Range("B6").Select
ActiveSheet.Paste
ActiveSheet.Range("B3").Value = StringConcat(" ", myString, "Levels
for", Range("B5").Value, "vs.", Range("C5").Value)

```

End Sub

```

Sub TransferAnalyseItGraphsToPowerpoint()

Dim i As Integer
Dim ppt, pres, NewSlide As Object
Dim s As PowerPoint.Slide
Dim shp As PowerPoint.Shape
Dim ws As Worksheet

Set ppt = CreateObject("powerpoint.application")
Set pres = ppt.Presentations.Add

i = 1

For Each ws In ActiveWorkbook.Worksheets
    ws.Select
    PrintTheScreen
    Set NewSlide = pres.Slides.Add(i, ppLayoutBlank)
    NewSlide.Shapes.Paste
    i = i + 1
Next ws

ppt.Visible = True

```

End Sub

4.4.6 User Interface Macros

```

Private Sub OkayButton_Click()

'This subroutine allows the user to input values into the
'"ClusterPrep" user form for the number of proteins to be
'analyzed, the number of samples to be randomly assigned
'as unknowns (test samples), and the number of runs desired.
'These integer values are then assigned to the global
'variables "NumProteins", "NumUnknowns, and "TestNumber" for
'use in the "RunClusterPrep" subroutine. The user also
'specifies the directory into which the statistical analysis
'files generated will be placed. If any of the fields in the
'user form remain unfilled, a message box prompts the user
'to fill in that field. Upon clicking "Okay", the
'"RunClusterPrep" subroutine gets underway.

Dim iRow As Long
Dim ws As Worksheet
Dim str As String

```

```

If Trim(Me.ProteinTextBox.Value) = "" Then
    Me.ProteinTextBox.SetFocus
    MsgBox "Enter number of proteins"
    Exit Sub
End If

If Trim(Me.UnknownCasesTextBox.Value) = "" Then
    Me.UnknownCasesTextBox.SetFocus
    MsgBox "Enter the number of unknowns"
    Exit Sub
End If

If Trim(Me.TestsTextBox.Value) = "" Then
    Me.TestsTextBox.SetFocus
    MsgBox "Enter number of tests"
    Exit Sub
End If

If Trim(Me.DirectoryTextBox.Value) = "" Then
    Me.DirectoryTextBox.SetFocus
    MsgBox "Please choose a directory"
    Exit Sub
End If

'copy the data to the database
NumProteins = Me.ProteinTextBox.Value
NumUnknowns = Me.UnknownCasesTextBox.Value
TestNumber = Me.TestsTextBox.Value
'strDirectoryPathName = Me.DirectoryTextBox.Value

RunClusterPrep

End Sub

```

```

Sub FolderSelection()

'This subroutine assigns the folder path and name chosen
'by the user via the SelectFolder function to a string.
'It then displays a message box containing that string.
'If no folder was chosen, it displays the message
'"Cancel was pressed".

    strFolderPathAndName = SelectFolder("Select Folder", "")

    If Len(strFolderPathAndName) Then
        MsgBox strFolderPathAndName
    Else
        MsgBox "Cancel was pressed"
    End If

End Sub

```

```
Function SelectFolder(Optional Title As String, Optional TopFolder _
                    As String) As String
```

```
'This function opens up a hierarchical menu of directories
'such that the user can choose a folder (in which to save files,
'for example). The function uses two optional arguments. The first
'is the dialog caption and the second is is to specify the top-most
'visible folder in the hierarchy. The default is "My Computer."
```

```
Dim objShell As New Shell32.Shell
Dim objFolder As Shell32.Folder
```

```
'If you use 16384 instead of 1 on the next line,
'files are also displayed
```

```
Set objFolder = objShell.BrowseForFolder(0, Title, 1, TopFolder)
If Not objFolder Is Nothing Then
    SelectFolder = objFolder.Items.Item.Path
End If
```

```
End Function
```

```
Private Sub ChooseDirectory_Click()
```

```
'Upon clicking the "Choose Directory" button on the user
'form, this subroutine runs the FolderSelection subroutine,
'which allows the user to select the directory into which
'their files are to be saved. A string containing the file
'path and name then fills the directory field in the user
'form.
```

```
FolderSelection
Me.DirectoryTextBox.Value = strFolderPathAndName
```

```
End Sub
```

```
Private Sub CloseButton_Click()
```

```
'Upon clicking the "Close" button, this subroutine deletes
'all values from the user form.
```

```
Unload Me
```

```
End Sub
```

```
Private Sub ClusterPrep_QueryClose(Cancel As Integer, _
    CloseMode As Integer)
```

```
If CloseMode = vbFormControlMenu Then
    Cancel = True
    MsgBox "Please use the button!"
End If
```

```
End Sub
```

4.4.7 String Manipulations

```
Sub ExtractFirstWord()

    Range("BE2").Select
    ActiveCell.FormulaR1C1 = _
        "=IF(LEN(RC[-49])=0, \"\", IF(ISERR(FIND(\" \" ,RC[-53])),RC[53], _
            LEFT(RC[-53],FIND(\" \" ,RC[-53])-1)))"
    Range("BE2").Select
    Selection.Copy
    Range("BE2:BE500").Select
    ActiveSheet.Paste
```

```
End Sub
```

```
Sub ExtractStringAfterDash(ByVal strAfterDash As String)

    Workbooks(strAfterDash).ActiveSheet.Activate
    ActiveCell.Offset(-1, 2).FormulaR1C1 = "=Mid(RC[-2],FIND(\"\"-\"\", _
    RC[-2])+1,20)"
```

```
End Sub
```

```
Sub ExtractWordsBeforeDash()

    Dim m As Integer

    m = 1

    Range("A2").Select

    Do While Not IsEmpty(ActiveCell)

        ActiveCell.Offset(1, 0).Select
        m = m + 1

    Loop

    Range("B2").Select
    ActiveCell.FormulaR1C1 = "=LEFT(RC[-1],FIND(\"\" - \"\",RC[-1])-1)"

    'Or Extract Words Before Space
    'ActiveCell.FormulaR1C1 = "=LEFT(RC[-1],FIND(\" \" ,RC[-1])-1)"
    Range("B2").Select

    Selection.Copy
    Range(Range("B2"), Range("B2").Offset(m - 2, 0)).Select
    ActiveSheet.Paste
```

```
End Sub
```

```

Sub ConvertDateToString()
Dim i As Integer

    ActiveWorkbook.ActiveSheet.Activate

    Do While Not IsEmpty(ActiveCell)
        ActiveCell.Value = "" & CStr (ActiveCell.Value)
        ActiveCell.Offset(1, 0).Select
    Loop

End Sub

```

```

Function StringConcat(Sep As String, ParamArray Args()) As String
' StringConcat
' This function concatenates all the elements in the Args array,
' delimited by the Sep character, into a single string. This function
' can be used in an array formula.
Dim s As String
Dim n,m,NumDims,LB,RN,CN As Long
Dim R As Range
Dim IsArrayAlloc As Boolean

    ' If no parameters were passed in, return
    ' vbNullString.
    If UBound(Args) - LBound(Args) + 1 = 0 Then
        StringConcat = vbNullString
        Exit Function
    End If

    For n = LBound(Args) To UBound(Args)
        ' Loop through the Args
        If IsObject(Args(n)) = True Then
            ' OBJECT
            ' If we have an object, ensure it
            ' it a Range. The Range object
            ' is the only type of object we'll
            ' work with. Anything else causes
            ' a #VALUE error.
            If TypeOf Args(n) Is Excel.Range Then
                ' If it is a Range, loop through the
                ' cells and create append the elements

```

```

' to the string S.
' .....
For Each R In Args(n).Cells
    s = s & R.Text & Sep
Next R

Else
    ' .....
    ' Unsupported object type. Return
    ' a #VALUE error.
    ' .....
    StringConcat = CVErr(xlErrValue)
    Exit Function
End If

Else If IsArray(Args(n)) = True Then

    On Error Resume Next
    ' .....
    ' ARRAY
    ' If Args(N) is an array, ensure it
    ' is an allocated array.
    ' .....
    IsArrayAlloc = (Not IsError(LBound(Args(n))) And _
        (LBound(Args(n)) <= UBound(Args(n))))

    On Error GoTo 0

    If IsArrayAlloc = True Then
        ' .....
        ' The array is allocated. Determine
        ' the number of dimensions of the
        ' array.
        ' .....
        NumDims = 1

        On Error Resume Next

        Err.Clear
        NumDims = 1

        Do Until Err.Number <> 0

            LB = LBound(Args(n), NumDims)

            If Err.Number = 0 Then
                NumDims = NumDims + 1
            Else
                NumDims = NumDims - 1
            End If

        Loop

        ' .....
        ' The array must have either

```

```

' one or two dimensions. Greater
' that two causes a #VALUE error.
' .....
```

```

If NumDims > 2 Then
    StringConcat = CVErr(xlErrValue)
    Exit Function
End If

If NumDims = 1 Then

    For m = LBound(Args(n)) To UBound(Args(n))
        If Args(n)(m) <> vbNullString Then
            s = s & Args(n)(m) & Sep
        End If
    Next m

Else

    For RN = LBound(Args(n), 1) To UBound(Args(n), 1)
        For CN = LBound(Args(n), 2) To UBound(Args(n), 2)
            s = s & Args(n)(RN, CN) & Sep
        Next CN
    Next RN
End If

Else
    s = s & Args(n) & Sep
End If

Else
    s = s & Args(n) & Sep
End If

Next n

' .....
```

```

' Remove the trailing Sep character
' .....
```

```

If Len(Sep) > 0 Then
    s = Left(s, Len(s) - Len(Sep))
End If

StringConcat = s

End Function
```

4.4.8 Other Useful Macros

```
Sub AddRunAnalysisCommandBarButton()
```

```

' This subroutine adds a button called "Run Analysis" to the
' Excel Add-Ins command bar which, when clicked, opens the
```

```
'user form and runs the ClusterPrep analysis on the active
'Excel Worksheet.
```

```
Dim AddBtn As CommandBarButton
```

```
Set AddBtn = CommandBars("Standard").Controls.Add
```

```
With AddBtn
    .Caption = "Run Analysis"
    .OnAction = "ClusterPrep.Show"
    .Style = msoButtonCaption
End With
```

```
End Sub
```

```
Sub TransferAllGraphsOnSheetsToPowerpoint()
```

```
'This subroutine transfers all graphs on a worksheet to a
'powerpoint file. It repeats this for all sheets in the workbook.
```

```
Dim ppt, pres, NewSlide As Object
Dim i As Integer
```

```
i = 1
```

```
Set ppt = CreateObject("powerpoint.application")
```

```
ppt.Visible = True
```

```
Set pres = ppt.Presentations.Add
```

```
For Each ws In Worksheets
```

```
    ws.Activate
```

```
    For j = 1 To ActiveSheet.ChartObjects.Count
```

```
        ActiveSheet.ChartObjects(j).Select
        ActiveSheet.ChartObjects(j).Copy
        Set NewSlide = pres.Slides.Add(i, ppLayoutBlank)
        ActiveChart.CopyPicture Appearance:=xlScreen, Size:=xlScreen, _
        Format:=xlPicture
        NewSlide.Select
        NewSlide.Shapes.Paste.Select
        ppt.ActiveWindow.Selection.ShapeRange.ScaleWidth 1.1, msoFalse, _
        msoScaleFromBottomRight
        ppt.ActiveWindow.Selection.ShapeRange.ScaleHeight 1.1, _
        msoFalse, msoScaleFromBottomRight
        ppt.ActiveWindow.Selection.ShapeRange.Align msoAlignCenters, True
        ppt.ActiveWindow.Selection.ShapeRange.Align msoAlignMiddles, True
        ppt.ActiveWindow.Selection.SlideRange.Shapes(1).Select
        i = i + 1
```

```
    Next j
```

```
Next ws
```


End Sub

```
Sub ImportABunch()

'This subroutine copies and pastes each .png file within the
'directory specified (in this case, the Desktop) into
'a separate slide in powerpoint.

Dim strTemp, strPath, strFileSpec As String
Dim Sld As Slide
Dim Pic, TextShape As Shape
Dim ShpRange As ShapeRange

' Edit these to suit:
strPath = "C:\Documents and Settings\Heath Group\Desktop\"
strFileSpec = "*.png"

strTemp = Dir(strPath & strFileSpec)

Do While strTemp <> ""
    ActiveWorkbook.ActiveSheet.Activate
    'Range("A1").Value = strTemp
    'Range("A2").Value = InStr(1, strTemp, "corr", vbTextCompare)
    strTemp = Dir
Loop
```

End Sub

```
Sub FormatActiveChart()

Dim k As Integer
Dim x As Object

With ActiveChart
    .ChartType = xlXYScatter
    .SetElement (msoElementChartTitleAboveChart)
    .PlotArea.Width = 330
    .HasLegend = True
    .Legend.Position = xlLegendPositionTop
    .Legend.Left = 90
    .Legend.Top = 20

    With .ChartTitle
        .Font.Size = 10
        .Font.Name = "Calibri (Body)"
    End With

    With .Axes(xlCategory)
        .MinorUnit = 1
        .MajorUnit = 37
        .MaximumScale = 37
        .MinorTickMark = xlTickMarkInside
    End With
```

```

        With .Axes(xlValue)
            .MinorUnit = 100
            .MajorUnit = 500
            .MinimumScale = -500
            .MaximumScale = 1500
        End With

    End With

    k = 1

    For Each x In ActiveChart.SeriesCollection

        x.MarkerSize = 4

        With ActiveChart.SeriesCollection(k)
            .MarkerForegroundColorIndex = 2 + k
            .MarkerBackgroundColorIndex = 2 + k
            .ErrorBars.Border.ColorIndex = 2 + k
        End With

        k = k + 1
    Next x

End Sub

```

```

Sub DeleteAllChartsOnEachSheet()

    'This subroutine deletes all charts on all sheets
    'of the active workbook

    For i = 1 To ActiveWorkbook.Worksheets.Count

        ActiveWorkbook.Worksheets(i).Select
        DeleteAllChartsOnSheet

    Next

End Sub

```

```

Sub DeleteAllChartsOnSheet()

    'This subroutine deletes all charts on the active worksheet

    Dim myshape As Shape

    For Each myshape In ActiveSheet.Shapes
        myshape.Delete
    Next myshape

End Sub

```

```
Sub PrintTheScreen()

    Application.SendKeys "{1068}"
    'Application.SendKeys "{1068}"
    DoEvents

End Sub
```

4.4.9 Batch Files for Running Cluster 3.0 and Java Treeview

Recall that among the “ClusterPrep” output files are a set of text files (typically 10) that contain data from all patients in a cohort as well, including a set of randomly assigned test samples. Manually opening and processing each of these files in Cluster 3.0 can be a very time-consuming process. Moreover, one may want to create .cdt files (Cluster files) with a number of different normalization, centering, and clustering permutations. To automate this process, we wrote a batch file “clusterstuff.bat” to allow all the text files created by the “ClusterPrep” software to be automatically processed by Cluster 3.0. This batch file instructs cluster to produce 8 (2 sets of 4) different .cdt files. The first set utilizes a centered Pearson correlation, whereas the second set utilizes an uncentered correlation. Within each set, the following normalization methods are employed: 1. No normalization, 2. Proteins normalized for each patient sample, 3. Patient samples normalized for each protein, and 4. Both proteins and patient samples normalized. Typically, only the .cdt files produced using method 4 were used. Note that Cluster 3.0 gives the option of normalizing by genes and arrays rather by proteins and patient samples. This is because the program is typically used for cluster analysis of gene expression microarrays. However, these designations (i.e. genes vs. proteins) are interchangeable. The batch file, stored in the Cluster 3.0 folder (C:\Program Files\Stanford University\Cluster 3.0), is shown below:

```
@echo off

set filename=%1

set namer=%filename:~0,-4%
set namer=%namer%_CorrUncentered.txt
type %1 > %namer%
cluster -f %namer% -g 1 -e 1 -m a
del %namer%

set namer=%filename:~0,-4%
set namer=%namer%_CorrCentered.txt
type %1 > %namer%
cluster -f %namer% -g 2 -e 2 -m a
del %namer%

set namer=%filename:~0,-4%
set namer=%namer%_NormalizedGenes_CorrUncentered.txt
type %1 > %namer%
```

```
cluster -f %namer% -ng -g 1 -e 1 -m a
del %namer%
```

```
set namer=%filename:~0,-4%
set namer=%namer%_NormalizedGenes_CorrCentered.txt
type %1 > %namer%
cluster -f %namer% -ng -g 2 -e 2 -m a
del %namer%
```

```
set namer=%filename:~0,-4%
set namer=%namer%_NormalizedArray_CorrUncentered.txt
type %1 > %namer%
cluster -f %namer% -na -g 1 -e 1 -m a
del %namer%
```

```
set namer=%filename:~0,-4%
set namer=%namer%_NormalizedArrays_CorrCentered.txt
type %1 > %namer%
cluster -f %namer% -na -g 2 -e 2 -m a
del %namer%
```

```
set namer=%filename:~0,-4%
set namer=%namer%_NormalizedGeneArrayCorrUncentered.txt
type %1 > %namer%
cluster -f %namer% -ng -na -g 1 -e 1 -m a
del %namer%
```

```
set namer=%filename:~0,-4%
set namer=%namer%_NormalizedGeneArray_CorrCentered.txt
type %1 > %namer%
cluster -f %namer% -na -ng -g 2 -e 2 -m a
del %namer%
```

This batch file is executed when the file “analysis.bat” is clicked. The latter file is placed in the directory containing the text files that are to be analyzed by Cluster 3.0. The “analysis.bat” file contains the following set of instructions:

```
@echo off
```

```
set path=%path%;C:\Program Files\Stanford University\Cluster 3.0;
for /f %%a in ('dir /b *.txt') do clusterstuff.bat %%a
```