# Utilizing machine learning techniques to rapidly identify MUC2 expression in colon cancer tissues

Thesis by

Preethi Periyakoil

In partial fulfillment of the requirements for the degree of

Bachelor of Science in Computer Science


CALIFORNIA INSTITUTE OF TECHNOLOGY

Pasadena, California

2018

ACKNOWLEDGEMENTS

I would like to thank my mentors Dr. Yisong Yue (California Institute of Technology) and Dr. Michael Clarke (Stanford Institute for Stem Cell Biology and Regenerative Medicine), as well as my co-mentor Dr. Debashis Sahoo (University of California, San Diego) for their unwavering encouragement and support. I am also grateful to Dr. Erin Burkett, Ms. Donna Wrubiewskie, and Ms. Melissa Ray for their aid in finding resources for this project.

# ABSTRACT

Colorectal cancer is the third-most common form of cancer among American men and women. Like most tumors, colon cancer is sustained by a subpopulation of "stem cells" that possess the ability to self-renew and differentiate into more specialized cell types. It would be useful to detect stem cells in images of colon cancer tissue, but the first step in being able to do so is to know what genes are expressed in the stem cells and how to detect their expression pattern from the tissue images. Machine learning (ML) is a powerful tool that is widely used in biological research as a novel and innovative technique to facilitate rapid diagnosis of cancer. The current study demonstrates the feasibility and effectiveness of using ML techniques to rapidly detect the expression of the gene MUC2 (mucin 2) in colon cancer tissue images. We analyzed histological images of colon cancer and segmented the nuclei to look for features (area, perimeter, eccentricity, compactness, etc.) that correlate with high or low levels of MUC2. Grid search was then run on this data set to tune the hyper-parameters, and the following models were tested as potential classifiers: random forest, gradient boosting, decision trees with AdaBoost, and support vector machines. Of all of the tested models, it was found that the random forest classifier (f1 score of 0.71) and the gradient boosting classifier (f1 score of 0.72) were able to predict the output label most accurately. Under certain conditions, we have identified four features that have predictive capabilities. Predicting individual gene expression with machine learning is the first step in detecting genes that are specific to cancer stem cells in the early stages of cancer, while there is still hope for a cure.

# TABLE OF CONTENTS

# INTRODUCTION

With the exception of some skin cancers, colorectal cancer is the third most prevalent form of cancer in the United States, among both women and men. In 2018 alone, 150000 cases of colorectal cancer are expected to occur, of which 50,000 are predicted to be fatal[1]. The prevalence of colon cancer calls for early diagnosis of the disease, as well as a way to quickly predict the prognosis of colon tumors. Thus, strategies to locate cancer stem cells in colon cancer may result in identifying patients in whom timely treatment of the disease will result in cure.

Small subsets of cancer cells constitute a reservoir of self-sustaining cells with the exclusive ability to self-renew and maintain tumors. These cells are known as cancer stem cells. Cancer stem cells have the ability to differentiate into all specialized cell types found in a particular cancer sample[2]. To identify cancer stem cells, we first need to be able to accurately predict the gene expression of cancer tissue. Effective detection of cancer stem cells will result in diagnosis of very early-stage cancers while they are amenable to cure.

Large volumes of gene expression data and images of cancer tissue are now publicly available to the medical research community. Biological data sets, such as gene expression data, tend to be vast and complex and can be effectively analyzed through sophisticated computational techniques such as machine learning (ML). Various ML algorithms have been used to analyze sequencing data and predict gene expression[3]. Convolutional neural networks have also been used to analyze breast cancer histopathological images[4]. While some studies have been conducted to analyze gene expression[5] and images of colon cancer[6], the correlation between the gene expression

data and the cancer image data has not been sufficiently elucidated. ML provides an effective approach for analyzing this correlation. If ML can be used to predict the expression of certain genes in cancer tissue images, it can possibly be used to detect the presence of cancer stem cells.

MUC2 is a gene that encodes for a protein in the mucin family (also called mucin 2), which is secreted onto mucous membranes. MUC2 is primarily expressed in goblet cells, which are present in the epithelial lining of the lumen of the colon; thus, MUC2 is indicative of a more differentiated phenotype of cancer. Studies show that loss of expression of MUC2 is correlated with lower survival, and that silencing MUC2 promotes colon cancer metastasis[7].

The goal of the current study was to determine if ML techniques could be utilized to successfully and rapidly classify specific regions of colon cancer tissue as either high or low in MUC2. MUC2 was chosen as the gene of interest in this study because the dataset of colon cancer images is evenly distributed (i.e., the image data set has relatively equal amounts of MUC2-high and MUC2-low images) and is ideally suitable for ML analysis. This is a supervised learning problem for which the solution involves creating a classifier that predicts level of MUC2 expression. As little work has been done in this area, we seek to fill this knowledge gap by creating and validating a classification model. If successful, this approach will pave the pathway to automating rapid and early identification of colon cancer and possibly other malignant tumors.

# METHODS

*Nuclei Segmentation with HistomicsTK*

UCSD's Hegemon Tools[8], which derive Boolean implications from large-scale genome microarray datasets, was used to identify patient identification numbers (IDs) that were associated with both low and high expressions of MUC2. Images of cross-sections of colon tissue samples were obtained from the Cancer Digital Slide Archive using these patient IDs[9]. Based on the data obtained from the Hegemon Tools, each image was labeled as +1 if its corresponding patient ID was found to be MUC2-high, and as -1 if its patient ID was found to be MUC2-low. 437 patient IDs were found in total, and 897 images were downloaded.

Each colon cancer tissue image contained several regions of cancer cell clusters, as well as of stromal cells, fibroblasts, muscle tissue, and necrotic tissue. The cancer regions, which were the only regions of interest, were cropped and isolated with the help of Dr. Michael Clarke and Dr. Piero Dalerba, who are experts in cancer stem cell biology. Each of these isolated regions, which will be referred to as annotations, was saved as a separate image file and labeled as MUC2-high (+1) or MUC2-low (-1), depending on the image from which it was cropped (which was based on the corresponding patient ID). From a total set of 897 images, 61 were annotated and used for training and cross-validation of the ML models. These images were manually analyzed, and 2336 annotations were made.

For each cancer region, OpenCV (an open-source computer vision software package)[10] was used to segment the nuclei. In order to do so, a stain color map was obtained for hematoxylin and eosin (dyes that stain nuclei) from the HistomicsTK

package (http://github.com/DigitalSlideArchive/HistomicsTK). This stain color map was used to filter out all but the range of RGB values corresponding to the violet color of the nuclei stains, and OpenCV was then used to find the contours in this filtered image. Each nucleus was modeled as an ellipse, and a set of numerical features was obtained for each one. These features included: area, moments (of which there were 24), aspect ratio, perimeter, major axis, minor axis, radius, equivalent diameter, extent, defect, compactness, and eccentricity. Each of these features was averaged across all the nuclei for that annotation. These mean values, along with the output label based on the patient ID, corresponded to one data point; there were a total of 2336 data points.

Relevant features were used to create a feature vector, which was then used to create a classification model using various algorithms. Using the Python package scikit-learn[11], grid search was run to tune the hyper-parameters of each of the following classifiers on the entire data set: a random forest classifier, an Adaboost classifier with a decision tree as its base estimator, a gradient boost classifier, and a support vector machine classifier. The cross-validation accuracy of the grid search was compared across models to determine the most accurate one for this data set. A correlation heat map was created for this data set to visualize the correlation between the MUC2 output label and each individual feature, and the precision, recall, f1 scores, and support were calculated and recorded for each model.

*Nuclear Segmentation with the Blue Ratio Transform*

The nuclei segmentation and ML was conducted in parallel with the following alternate procedure. The Blue Ratio transform was used to enhance the nuclei staining[12]. OTSU thresholding was performed on the transformed image[13]. The holes in the

thresholded image were filled, and binary opening and closing were performed. The connected components and centroids for each contour were computed, and the Watershed algorithm was used to draw outlines around individual nuclei[14]. Finally, OpenCV was used to find the contours within each outline, which isolated all the nuclei individually. As before, each nucleus was modeled as an ellipse, and the same set of numerical features was obtained for each one. However, instead of averaging the nuclear features across each annotation, the features of each nucleus were recorded as individual data points, along with the output label based on the patient ID. This new procedure led to approximately 1.5 million data points, one for each nucleus.

Grid search was also performed on this data set to tune the hyper-parameters of the same ML models as used in the first iteration of this procedure: support vector machines, a random forest classifier, an AdaBoost classifier with a decision tree as its base estimator, and a gradient boost classifier. The models were scored using cross-validation and the precision, recall, f1 score, and support were all calculated and recorded. A correlation heat map could not be drawn for this large data set due to potentially large volumes of noise, so histograms were drawn to compare the frequency of values for every individual feature in MUC2-high versus MUC2-low tissue samples.

# RESULTS

A correlation heat map was drawn for this data set to visually identify features that corresponded with high or low levels of MUC2. This heat map is shown in Figure 1:

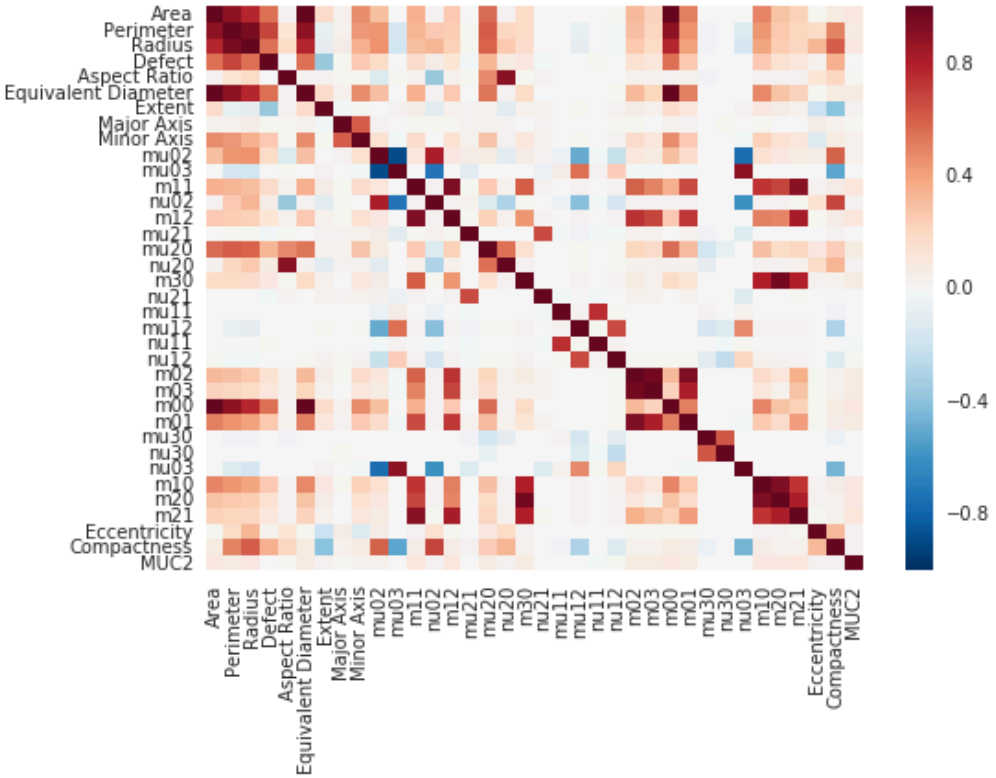
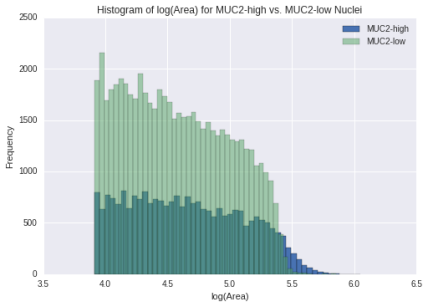



**Figure 1: Heat map showing the correlation between each pair of features. The last column and last row show the correlation between each feature and the MUC2 output label. Blue indicates a negative correlation, while red denotes a positive correlation.**

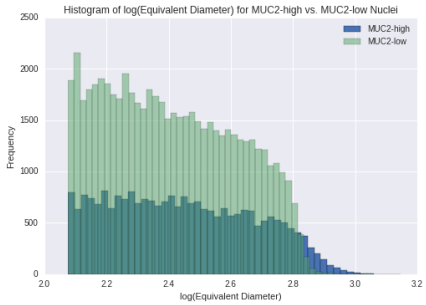
The last column and the last row show the correlation between each feature and the MUC2 label; there is very little correlation between any of the features and the output label (almost all the cells in the heat map are white in this row and column). These results may have been attributed to one of these two reasons: (1) the stain color map was too specific for the images used (because certain images had different coloring), or (2) the data set was too small. Thus, this procedure was repeated with a different algorithm for nuclei segmentation, as well as with equal weight given to each individual nucleus.

Using the second procedure (with the Blue Ratio transform and recording individual nuclei), a preliminary data set was created from three MUC2-high slide images (23993 nuclei) and two MUC2-low slide images (58195 nuclei), which led to a total of 82188 data points. For each feature, a histogram was plotted to show any differences between MUC2-high and MUC2-low nuclei. No significant differences between MUC2-high and MUC2-low were found in the distributions for any individual feature, so they were again plotted on a logarithmic scale. The following four features were found to have differences in the distributions between MUC2-high and MUC2-low: area, equivalent diameter, m00, and m12 (m00 and m12 are two of the 24 image moments that were collected as individual features). These histograms are shown in Figure 2:

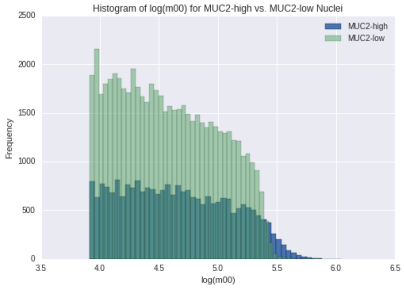
A




B




C




D

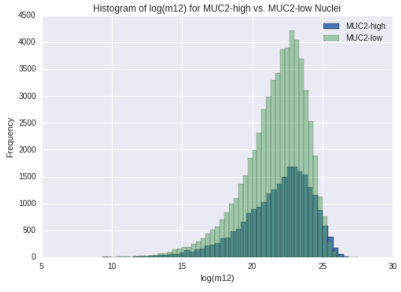



**Figure 2: Histograms showing distribution of values for the following features in MUC2-high and MUC2-low nuclei: (A) Area, (B) Equivalent Diameter, (C) m00, (D) m12. Feature values are displayed on a logarithmic scale.**

The right tails of the histograms show that MUC2-high tissue has more nuclei that have higher values for all four features. Because the preliminary data set had more MUC2-low than MUC2-high nuclei, it was surprising to find that more MUC2-high nuclei had higher values for these four features. Figure 3 shows a closer view of the right tails of each of the histograms shown in Figure 2:

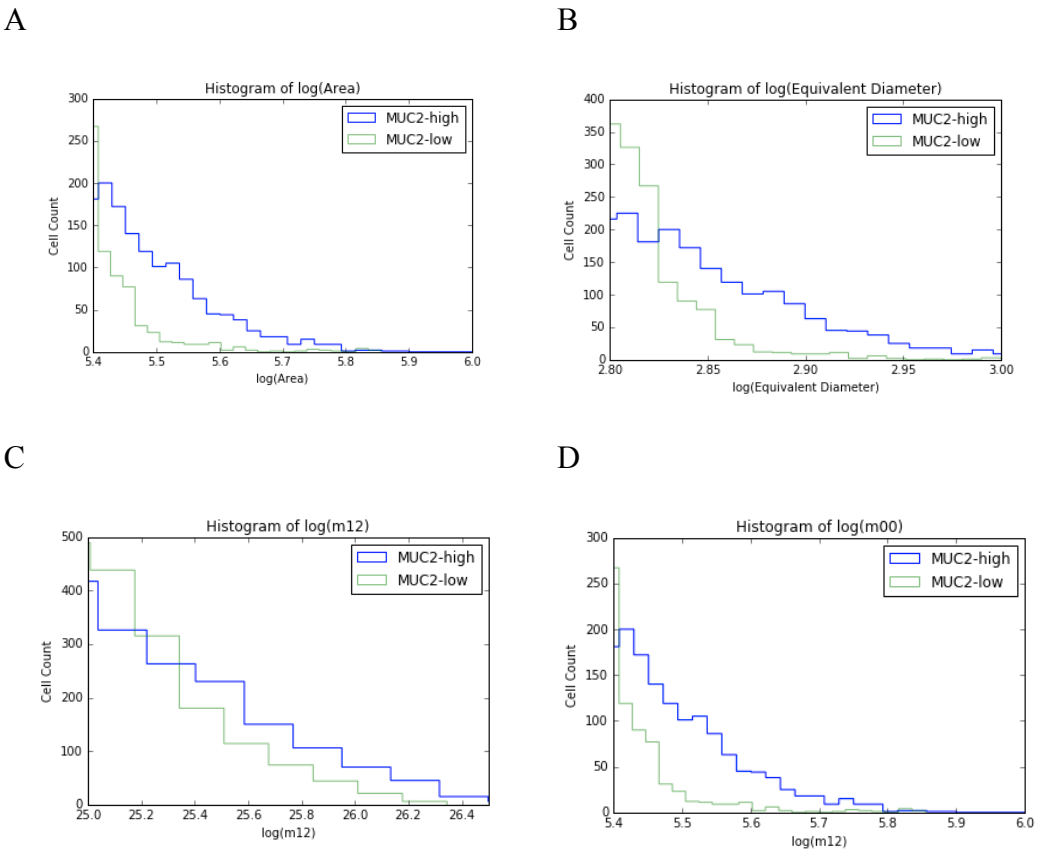



**Figure 3: Close-up view of the right tails of the histograms shown in Figure 2.**

The histograms above show that MUC2-high tissue will have more cells with larger area, equivalent diameter, and moments m00 and m12, than MUC2-low tissue do. Because MUC2 is usually identified through tissue staining, the morphology of nuclei that are positive for MUC2 is not well known. These results suggest that tissue containing high volumes of large nuclei (or nuclei with large equivalent diameters) may be associated with high levels of MUC2. Moments are a quantitative measure of the spatial distribution

of a set of points, and for these images, these moments are based on pixel intensity. It is unknown how this information correlates with existing information about colon cancer nuclei, and thus further experimentation should be conducted to analyze this relationship.

Grid search was conducted on the data set generated by each procedure, and the precision, recall, F1 scores, and support were calculated and recorded for each ML model. The calculated scores are reported in the following tables:

| Model | Output Label | Precision | Recall | F1 Score | Support |
|---|---|---|---|---|---|
| SVM | -1.0 | No convergence | No convergence | No convergence | No convergence |
| | +1.0 | No convergence | No convergence | No convergence | No convergence |
| | Average over total data set | No convergence | No convergence | No convergence | No convergence |
| Adaboost | -1.0 | 0.62 | 0.58 | 0.60 | 452 |
| | +1.0 | 0.67 | 0.70 | 0.68 | 536 |
| | Average over total data set | 0.65 | 0.65 | 0.65 | 988 |
| Random Forest | -1.0 | 0.72 | 0.62 | 0.67 | 452 |
| | +1.0 | 0.71 | 0.79 | 0.75 | 536 |
| | Average over total data set | 0.72 | 0.72 | 0.71 | 988 |
| Gradient Boosting | -1.0 | 0.70 | 0.68 | 0.69 | 452 |
| | +1.0 | 0.74 | 0.76 | 0.75 | 536 |
| | Average over total data set | 0.72 | 0.72 | 0.72 | 988 |

**Table 1: The precision, recall, f1 score, and support for each ML model with which grid search was performed on the first data set (with HistomicsTK and averaging the nuclei features, 2336 data points).**

The results in Table 1 show that for the first data set, gradient boosting and random forest classification more accurately predict the output labels in this data set than Adaboost, which converged but had a low f1 score, or support vector machines (SVM), which did not even converge. Because the correlation heat map showed little to no correlation

between any individual feature and the output label, the results shown in Table 1 were unexpected. The lack of correlation led us to believe that the results of the cross-validation would not be much better than random guessing (i.e., with an f1 score close to 0.5), so the f1 scores of 0.71 and 0.72 for random forest and gradient boosting, respectively, is much better than expected.

The same scores were computed for the ML models used to fit the data set generated with the second procedure (with the Blue Ratio transform and with the separate recording of the features of individual nuclei). These results are shown in Table 2:

| Model | Output Label | Precision | Recall | F1 Score | Support |
|---|---|---|---|---|---|
| SVM | -1.0 | No convergence | No convergence | No convergence | No convergence |
| | 1.0 | No convergence | No convergence | No convergence | No convergence |
| | Average over total data set | No convergence | No convergence | No convergence | No convergence |
| Adaboost | -1.0 | 0.52 | 0.54 | 0.53 | 520 |
| | 1.0 | 0.58 | 0.56 | 0.57 | 587 |
| | Average over total data set | 0.55 | 0.55 | 0.55 | 1107 |
| Random Forest | -1.0 | 0.60 | 0.57 | 0.59 | 520 |
| | 1.0 | 0.64 | 0.66 | 0.65 | 587 |
| | Average over total data set | 0.62 | 0.62 | 0.62 | 1107 |
| Gradient Boosting | -1.0 | 0.56 | 0.50 | 0.53 | 520 |
| | 1.0 | 0.60 | 0.65 | 0.62 | 587 |
| | Average over total data set | 0.58 | 0.58 | 0.58 | 1107 |

**Table 2: The precision, recall, f1 score, and support for each ML model with which grid search was performed on the second data set (with the Blue Ratio transform and recording individual nucleus features, approximately 1.5 million data points).**

As before, the grid search for SVM did not converge, and random forest and gradient boosting were found to predict the output labels more accurately than the Adaboost

classifier. Because the Blue Ratio transform was found to be more effective at segmenting nuclei than filtering the nuclei with the HistomicsTK package, it was expected that the f1 scores and precision and recall would be much higher for the second data set than for the first. Thus, these results are unexpected as well. However, the f1 scores for the random forest and gradient boosting classifiers are still greater than 0.5, and further fine-tuning of these models will hopefully improve the accuracy.

# DISCUSSION

The heat map for the first data set (Figure 1) showed that in the first data set, which was obtained by using HistomicsTK and averaging the nuclear features of each image annotation, there was little to no correlation between any of the collected features and the output label of MUC2-high or MUC2-low. In other words, this initial assessment suggested that predicting MUC2 expression through analyzing colon cancer tissue images would be next to impossible, and that any ML model used to fit this data set would not perform much better than random guessing. However, the precision, recall, and f1 scores for the random forest and gradient boost classifiers were close to 0.7, which indicates that these models perform much better than was expected. It is unclear why these models performed so well, and thus further research must be conducted to find out why. It is also not known which specific features are highly correlated with MUC2-high or MUC2-low tissue, and thus different methods of correlation or principal component analysis must be performed on the data set to find these features.

On the other hand, the Blue Ratio transform was found to be much more accurate and effective at segmenting nuclei than the HistomicsTK package, and recording the features of individual nuclei was expected to prevent loss of data (which was the anticipated disadvantage of averaging nuclear features across one annotation). Therefore, it was predicted that the precision, recall, and f1 scores would be much higher for the ML models used to fit to the second data set. However, although the scores were greater than 0.5 (i.e., they were better than random guessing), they were worse than those computed for the first data set. A possible reason for this result is that there may have been large amounts of noise in the second data set. It is possible that some additional tuning of the

parameters will improve the grid search result, but further research needs to be conducted in order to do so.

Our study is limited by the consideration that only 61 slide images were analyzed, even though large volumes of data were produced from said images. Because the cells in cancer tissue are extremely heterogeneous in shape, size, and other morphological features, it is necessary to have a high volume of data that is all encompassing of these morphological features. Thus, more images must be annotated and added to the data set, and any computed results will thus be more accurate. The other limitation of this study is that MUC2 is difficult to identify by eye, and thus the ML algorithms have not been trained on cells that are specifically known to express MUC2 (it is known that an entire tissue is MUC2 high, but little is known about how to visually identify the gene in a tissue sample). Thus, further training of the ML models must be conducted on a training set that consists of tissues specifically stained for MUC2. Doing so will enable the models to better identify features that are indicative of MUC2-high cells, and will thus make them more effective at predicting a tissue sample as MUC2-high or MUC2-low.

# CONCLUSION

We conducted a study to determine whether it was possible to perform ML on colon cancer tissue images and predict the expression of the gene MUC2. Predicting gene expression from an image of cancer tissue is an extremely useful tool, as doing so could potentially help identify cancer stem cells in a tissue region. However, in order to do so, we must first find a way to predict the expression of individual genes in cancer tissue images. This study shows that performing ML on images to predict gene expressions is possible, and further research should be conducted to improve these results and create models that can predict expression more accurately.

# REFERENCES

[1] American Cancer Society. Key Statistics for Colorectal Cancer. [homepage on the Internet]. 2016 [cited 2018 Feb 1]. Available from: American Cancer Society, Web site: https://www.cancer.org/cancer/colon-rectal-cancer/about/key-statistics.html

[2] Dalerba P, Cho RW, Clarke MF. Stem Cells: Models and Concepts. Annu Rev Med. 2007; 58:267-84.

[3] Li X. Using ML to predict gene expression and discover sequence motifs. 2012 [cited 2018 Feb 8]. Available from: Columbia University, Web site: http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.819.9134&rep=rep1&type=pdf

[4] Araújo T, Aresta G, Castro E, Rouco J, Aguiar P, Eloy C et al. Classification of breast cancer histology images using Convolutional Neural Networks. PLoS One 2017; 12(6):.

[5] Chen H, Shen J, Wang L, Song J. Towards data analytics of pathogen-host protein-protein interaction: a survey. Faculty of Engineering and Information Sciences - Papers: Part A. 2016 Jan 1;377–88.

[6] Komura D, Ishikawa S. Machine learning methods for histopathological image analysis. Computational and Structural Biotechnology Journal. 2018;16:34–42.

[7] Betge J, Schneider NI, Harbaum L, Pollheimer MJ, Lindtner RA, Kornprat P, et al. MUC1, MUC2, MUC5AC, and MUC6 in colorectal cancer: expression profiles and clinical significance. Virchows Arch. 2016;469(3):255–65.

[8] Sahoo D, Dill DL, Gentles AJ, Tibshirani R, Plevritis SK (2008) Boolean implication networks derived from large scale, whole genome microarray datasets. Genome Biol **9**:R157

[9] Gutman DA, Khalilia MA, Lee S, Nalisnik M, Mullen Z,beezley J et al. The Digital Slide Archive: A Software Platform for Management, Integration, and Analysis of Histology for Cancer Research. Cancer Research November 2017; 77(21):.

[10] Bradski GR, Kaehler A. Learning OpenCV: computer vision with the OpenCV library. 1. ed., [Nachdr.]. Beijing: O'Reilly; 2011. 555 p. (Software that sees).

[11] Pedregosa F, Varoquaux G, Gramfort A, Michel V, Thirion B, Grisel O, et al. Scikit-learn: Machine Learning in Python. Journal of Machine Learning Research. 2011;12(Oct):2825–30.

[12] Chang H, Loss LA, Spellman PT, Borowsky A, Parvin B. Batch-invariant nuclear segmentation in whole mount histology sections. In: 2012 9th IEEE International Symposium on Biomedical Imaging (ISBI). 2012. p. 856–9.

[13] OpenCV: Image Thresholding [Internet]. [cited 2018 February 1]. Available from: https://docs.opencv.org/3.4.0/d7/d4d/tutorial_py_thresholding.html

[14] Han B. Watershed Segmentation Algorithm Based on Morphological Gradient Reconstruction. In: 2015 2nd International Conference on Information Science and Control Engineering. 2015. p. 533–6.