STAT542 Group Project: Skin Cancer Diagnostics

Xinlei Zhang (xinleiz2) Xinyi Song (xinyis8) Lin Zhu (linzhu5)

I. Description and Summary

In this Project, we have 150 skin images of malignant moles as well as 150 images of benign moles. We aim to differentiate the malignant moles from those benign ones according to various features of their images. Label '1' is given to the malignant observations and label '0' is given to the benign ones.

The project can be divided into two parts. In the first part, we focus on the original RGB characteristics of the images. The intensities of Red, Green and Blue colors are bound together into one independent variable, which is also the only predictor we use to train and test our models in this part. Three different models are selected to do classifications. We get accuracy as from the Generalized Linear Regression model, accuracy as 0.6447 from the Extreme Gradient Boosting model and accuracy as 0.7320 from the Random Forest model. Random Forest performs the best when there is only one variable of RGB intensities.

In order to get more interpretable results, we transform the RGB intensities into six new independent variables through feature engineering: Brightness Asymmetry, Color Variation (Count of Color Types and RGB Standard Errors) and Mole Diameter. With these six predictors, we apply Support Vector Machine model and Random Forest model again. And Random Forest beats Support Vector Machine model with accuracies as 0.8537 vs. 0.6330. And accuracy is also elevated around 0.13 after feature engineering.

II. Classification Based on RGB Intensities

■ Data Processing

First, we cut all the images into the same 200*250-pixels-size, next we extract the RGB intensities as three columns of numeric values, with each column represents one of the Red, Green, Blue color. Then, we combine them by row, so that we can get an independent

variable of exactly 200*250*3 = 150000 rows for each image. So, our final data is a matrix of 150*150000 dimension.

For diagnostic, we label the malignant image as '1' while the benign image as '0' to run our classification model. Also, we define the accuracy as the proportion of observations that are correctly predicted.

Additionally, to evaluate the model performance, we randomly select 75% of total observations (benign and malignant equally weighted) as train data to train the classification models, and the rest of data as test data.

■ Classification Models

- **Generalized Linear Regression – Logistic Regression Model**

Logistic regression is used to describe data and to explain the relationship between one dependent binary variable and one or more nominal, ordinal, interval or ratio-level independent variables, and the elastic net is a regularized regression method that linearly combines the L1 and L2 penalties of the lasso and the ridge methods. We used the Ten-Fold Cross-Validation to figure out the lambda.min value by calling the function *cv.glmnet*, and then fitted the logistic regression model with sentiment versus key features by calling the *glmnet* function in package "glmnet" and setting "family=binomial" for canonical link function of logistic regression.

We tried both lasso regression and ridge regression by tuning the parameter alpha, here alpha=0 stands for ridge regression while alpha=1 represents lasso regression. For our images, ridge performs slightly better than lasso does in terms of accuracy. Ridge Regression shrinks the coefficients of the variable depends on its importance to the model accuracy rather than providing 0.

The accuracy of ridge regression is 0.64474, which indicates that approximate 64.5% of the total skin image was correctly diagnosed. For generalized linear classification model – logistic model with L2 norm regularization, it is easy to perform, but when the dimension of data is extremely large, the efficiency problem is concerned. Since here the predictors are all pixels, the model does not have good interpretability as usual linear model does.

- **Tree Model - Extreme Gradient Boosting**

As for the tree model, we chose to use boosting tree to analyze our data and do prediction. For boosting tree, trees are grown sequentially: each tree is grown using information from previously grown trees and each tree is fitted on a modified version of the original data set. For better performance, we select the XGBoost algorithm. This algorithm uses a more regularized model formalization to control over-fitting.

For this model, we mainly focused on tuning two parameters: nrounds and eta. For nrounds, it controls the maximum number of iterations. For classification, it is similar to the number of trees to grow. Eta controls the learning rate, it is the rate at which our model learns patterns in data. After every round, it shrinks the feature weights to reach the best optimum. Also, lower eta slower computation. For outputting value, if it is smaller than zero we would label it as zero while label it as one if it is larger than zero.

After tuning parameters, we found that when eta=0.2 and nounds = 240, XGBoost model could reach accuracy to 0.62. We found that XGBoost utilizes the power of parallel processing which makes it faster than GBM. Also, it is useful for handling the missing values. And in terms of efficiency, XGBoost runs much faster than linear model does especially with high dimensional data.

- **Random Forest**

Random Forest is an ensemble learning method for both classification and regression. It operates by building multiple decision trees at training time and outputting the class that is

the mode of the classes (for classification problem) or mean prediction value (for regression problem).

Previously we have tried the Extreme Gradient Boosting Tree model, it performed pretty well but still had some drawbacks for this dataset. firstly, decision tree may cause the problem of overfitting because it just builds one tree on the whole training set. Also, we should notice that a slightly change in the dataset may cause a totally different result of the decision tree, it is a kind of high variance algorithm.

Fortunately, random forest algorithm can overcome all those problems of decision tree. first of all, random forest can overcome the problem of overfitting because it builds multiple decision tree at training time, also because of these, it has less variance than one single decision tree. Also, random forest can achieve higher accuracy on test data and it maintains high accuracy even when there is large proportion of data are missing.

For random forest, we mainly tune the parameter node size. nodesize refers to how many observations we want in the terminal nodes. This parameter is directly related to tree depth. Higher the number, lower the tree depth. With lower tree depth, the tree might even fail to recognize useful signals from the data. After tuning parameters, we found that when nodezise = 5, the model performs well, its accuracy could reach 0.7230.

III. Classification Based on Feature Engineering

■ Clinical Literature Review

After reviewing several literatures, we select three relevant characteristics for skin cancer image diagnosis that are possible to extract.

- **Asymmetry**

  Asymmetrical skin growths, in which one part of the mole is different from the other, may indicate skin cancer. For example, when the left side of a mole is dark while the right side is lighter, it has higher possibility of being malignant.

- **Color Variation**

  If a mole has multiple colors or the distribution of the colors is very uneven, it has higher possibility of being malignant.

- **Diameter**

  Larger diameter of a mole may indicate skin cancer.

References:

[1]. Skin Cancer: A Practical Approach, Alfonso Baldi; Paola Pasquali; Enrico P. Spugnini Editors, Springer Science+Business Media New York 2014

[2]. Research on Skin Cancer Cell Detection Using Image Processing, Enakshi Jana; Ravi Subban ; S. Saraswathi, Publisher: IEEE, 2017

[3]. https://www.mayoclinic.org/diseases-conditions/skin-cancer/multimedia/melanoma/sls-20076095?s=1

■ Feature Engineering

- **Asymmetry**

  We mainly focus on the brightness asymmetry of the images. The images are already cut into the same 200*250 size to make sure they have same number of pixels. We calculate the brightness of the pixels as this formula:

$$Brightness = \frac{(Intensity_R \times 299)+((Intensity_G \times 587)+((Intensity_B \times 114)}{1000}$$

  According to our observation of the images, most of them are not askew with horizontal and vertical axes. So to simplify the problem, we first carve up the images into two parts horizontally and vertically. Next, we grab the maximum brightness (the darkest pixel), and the first quantile of brightness (the comparatively lightest pixel except for the normal skin pixels) from both parts. Then, we calculate the brightness difference for the two parts as this formula:

$$Brightness\ Difference = \sqrt{\left(Brightness_{P1,darkest} - Brightness_{P2,lightest}\right)^2 + \left(Brightness_{P1,lightest} - Brightness_{P2,darkest}\right)^2}.$$

After that, we have two brightness difference values, one for top and bottom parts, and the other for left and right parts. We select the larger one to represent the brightness difference for this certain image.

References:

- https://www.w3.org/TR/AERT/#color-contrast

- **Color Variation**

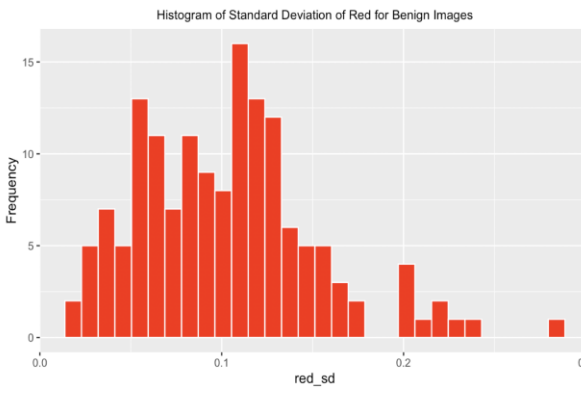
  Here we use the cut images to focus on the part of moles.

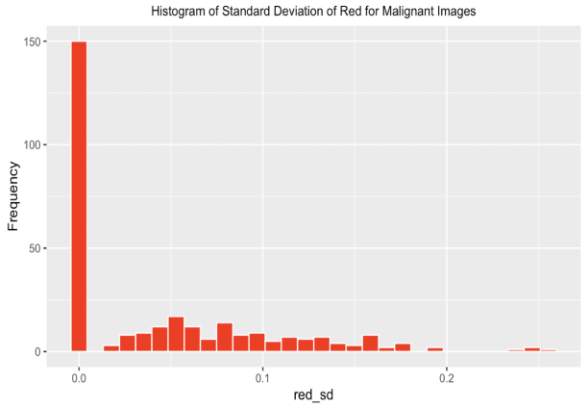
  - *Number of Color Types*

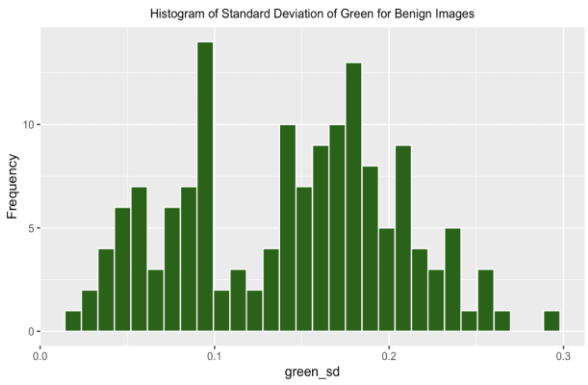
    We extract the number of different Hex colors through magick package in R to roughly see color variation of the images.

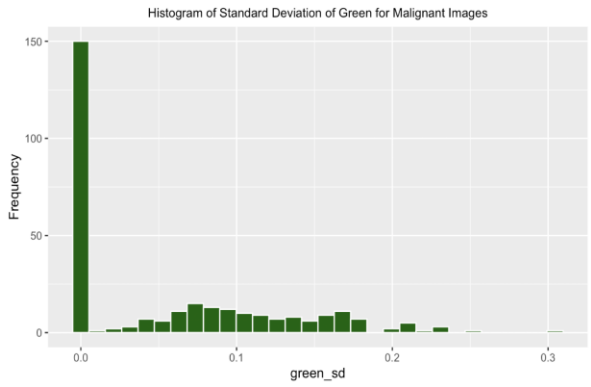
  - *Color Standard Deviation*

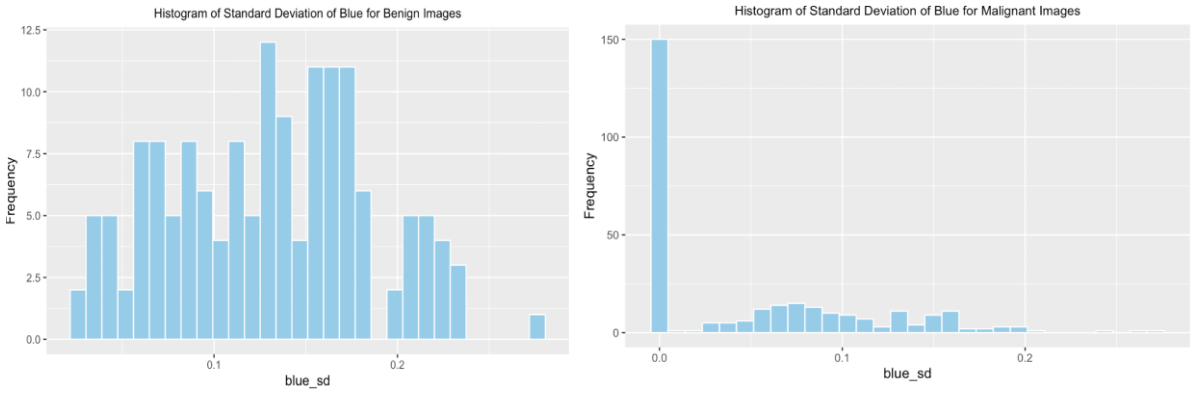
    To explore the information of color variation on benign and malignant images, we plot the histograms of standard deviation of corresponding pixels of three colors (red, green and blue) for both 150 benign and 150 malignant images.











Based on the plots above, we can find that there exist great differences regarding standard deviation of each color (red, green and blue) between benign images and malignant images. The standard deviation of each color in benign images looks normal while for that of malignant images, most of the observations have few variations. Therefore, we think the color variation could be an important feature for classifying benign and malignant images. We calculate the standard deviation of intensities for color Red, Green and Blue respectively as a detailed indication of color variation of the images.

- **Diameter**

  Here we use the original images without cutting. Again, to simplify the problem, we will calculate the diameter approximately by horizontal and vertical axes. First, we choose the middle rows of pixels horizontally and vertically. Then, we choose the first pixel of the horizontal middle row as our 'control group'. This is because according to our observation of the images, some of them seem like being taken from a microscope as the corners are dark, while the leftmost middle part is always normal skin. Then we calculate the relative error for each pixel as this formula:

$$Relative\ Error_i = \sum_{C=R,G,B} \frac{|\ Intensity_{C,i} - Intensity_{C,normal}\ |}{Intensity_{C,normal}}.$$

  After that, we observe the overall distributions of the differences, and we set the tolerance as 25% choose the top 75% to be the pixels that are obviously darker than normal skin so that the count of them can be the approximate diameter of the mole. Notice that we have chosen the larger count of the middle row and column.

After data engineering, we get 6 predictors in all: Asymmetry, Number of Color Types, Red Color Standard Deviation, Green Color Standard Deviation, Blue Color Standard Deviation and Diameter.

■ Classification Models

- **Support Vector Machine**

Support Vector Machine is one kind of supervised machine learning method in the statistics field. Given labeled training data, the algorithm of SVM outputs an optimal hyperplane which categorizes testing data. There are several different types of SVM such as Linear SVM in separable case and Non-linear SVM. Here, we choose to use linear SVM in separable case since after feature engineering, there only exists six variables in our model. And non-linear SVM is more suitable to high dimensional data.

For this classification model, here we set type = 'C-classification' since our response variable y is a factor. Also, set kernel='linear' which represents linear SVM in separable case. And we chose not to scale data. We mainly tune parameter cost: 'Cost' quantifies the penalty associated with having an observation on the wrong side of the classification boundary. Here, we set the cost as 1000. Here, the accuracy of SVM reached 0.6330 with our new features.

- **Random Forest**

Here, regarding our new selected features, we tried random forest again. random forest algorithm can overcome all those problems of decision tree. first of all, random forest can overcome the problem of overfitting because it builds multiple decision tree at training time, also because of these, it has less variance than one single decision tree. Also, random forest can achieve higher accuracy on test data and it maintains high accuracy even when there is large proportion of data are missing.

We mainly tuned parameters ‘mtry’ and ‘nodesize’. Here, ‘nodesize’ refers to how many observations we want in the terminal nodes. This parameter is directly related to tree depth. Higher the number, lower the tree depth. With lower tree depth, the tree might even fail to recognize useful signals from the data. And mtry refers to how many variables we should select at a node split. It is number of variables available for splitting at each tree node. After tuning parameters, we find that when mtry=6, nodesize =2, the model could reach accuracy of 0.8537, which indicated that our feature engineer performed well. And the new selected features made the classification model perform much better regarding both accuracy and interpretability.

IV. Conclusion

Based on the analysis above, we can find that when using pixels themselves to train classification model and work as reference for diagnostics, although the trained model performed well, it was not interpretable. After feature engineering, we chose six features including brightness asymmetry, number of color types, color variation (red, green and blue), diameter of moles in the skin cancer images and used these variables to train the classification model, not only did the accuracy of classification improved a lot, but also let the model be more interpretable, which verified our feature engineer procedure. We could conclude that based on the analysis above, the brightness asymmetry, color variation of moles, mole diameter and number of color types of image could be regarded as important features for skin cancer diagnostics based on skin images.