INFO 529

Final Project

Authors:
Chia-Hsuan Chou
Venkata Prudhvi Raj Indana
Jing Wang

May 06, 2016

Contents

I. Goal

Acute myeloid leukemia (AML) is a cancer of bone marrow and the blood. The ultimate goal of this challenge is to interpret the dataset of AML patients and try to figure out what drives AML. A predictive model will be proposed to help physicians to better assess patient's potential response to treatment.

II. Background

Acute myeloid leukemia (AML) is a cancer of the bone marrow and the blood. Mutations in the myeloid line of blood stem cells lead to the formation of aberrant myeloid blasts and white blood cells. If untreated, these highly proliferative cancerous cells impede the development of normal blood cells and eventually cause death.

In 2014, it is predicted that there will be at least 18,860 new cases of AML, and 10,460 deaths from AML. There is urgency in finding better treatments for this type of leukemia, as only about a quarter of the patients diagnosed with AML survive beyond 5 years.

AML is a collection of diseases that share a common clinical presentation despite arising from diverse mutations and genetic events. Array technologies at the gene, mRNA, microRNA and protein level have helped define the prognosis of AML patients. Interestingly, most AMLs seem to have only a couple of gene mutations, but AML patient prognosis is quite diverse. One reason for this is differences in protein signaling.

The AML Outcome Prediction Challenge provides a unique opportunity to access and interpret a rich dataset for AML patients that includes clinical covariates, select gene mutation status and proteomic data. Capitalizing on a unique AML reverse phase protein array (RPPA) dataset obtained at M.D. Anderson Cancer Center that captures 271 measurements for each patient, participants of the DREAM 9 Challenge will help uncover what drives AML. Outcomes of this Challenge have the potential to be used immediately to tailor therapies for newly diagnosed leukemia patients and to accelerate the development of new drugs for leukemia.

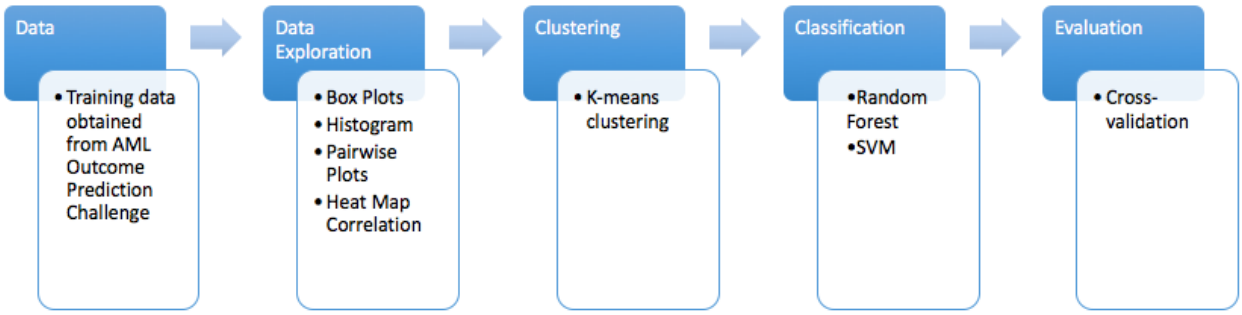
The Challenge asks participants to develop predictive models based on the AML dataset that can be used by physicians to better assess a patient's potential response to treatment, to more accurately predict the corresponding remission time, and to predict the ultimate survival time of patients following treatment.

The training data provided consists of measurements from 191 patients diagnosed with AML, who were treated at M.D. Anderson Cancer Center and received ara-C based chemotherapy. These measurements include 40 covariates describing patient demographics, cytogenics and mutation

status, and the results of several standard blood tests. See table below for a detailed description. In addition, the data includes proteomic measurements collected using 231 antibodies indicating the levels of either total or phosphorylated proteins for each patient.

III. Method

The pipeline of this project is shown below.



1. Data Preprocessing

Real world clinical data is often plagued by missing values. The data given in the AML Outcome Prediction Challenge is no exception. These data points are mostly represented by the letters "NA", though the letters "ND" are also used to describe missing values for a few clinical covariates. There are many ways to deal with missing data points, we used KNN imputation to replace missing values from the nearest neighboring column

2. Data Exploration

Before conducting any data analysis, we don't have any clue and assumption about how the data looks like and the relationship within the data. Therefore, the first step is to explore the data roughly and that will give us some pictures and hints for the further analysis. Here we choose boxplot, histogram, scatterplot and heat map for the data visualization from our training data.

Firstly, boxplot is the easiest and the most convenient way to know the quartiles of one or more sets of data intuitively. Moreover, boxplot clearly show us the outliers which influences the statistical values of the data and the variance which measures how the data points spread out.

Secondly, histogram helps us to estimate the probability distribution of a numerical continuous variables by binning the range of value roughly. Here we also draw the density estimation on our histogram figure; therefore, it enable us comparing the distributions of different features clearly.

Thirdly, scatterplot, which use a collection of points and point's cluster in a band, tells us the correlation between any two features from the data set. Here notice that the correlation between two features does not imply the causality which means we only can know the relationship between different features.

Finally, heap map gives us a correlation matrix which let us examine the data intuitively. Moreover, it also tells us the degree of correlation by colors and presents the cluster of features in the data roughly. Based on the information above, we are able to depict the outline of our data.

3. Unsupervised Machine Learning Analysis—Clustering

Although we have already done the data exploration, we only have little picture about what our results should look like. Therefore, in order to address the problem, we derive a structure by clustering the data based on relationships among features in the data. Here we implement K-means clustering which is the most common and easiest way to think of clustering and since it's computational time is not expensive, it is a best way to be implemented in our case which we have many features.

The goal of K-means clustering is to partition the observations into k different number of clusters; therefore, it aims to find the centroid of each cluster. The algorithm minimizes the L2 norm distance from the data points to the cluster centroid. The minimization objective function is written as,

$$\min \sum_{i=1}^{k} \sum_{x \in C_i} \|x - \mu_i\|^2,$$

where C_i is the set of points which belong to cluster i. The algorithm continues finding the centroid of each cluster until it converges and meets the stop criterion which means the assignments of the means do not chance from one iteration to the next iteration by some threshold.

4. Feature Selection—Random Forest

Since we have more features than observation in our training data, we have to consider curse of high dimensionality may influence our analysis and cause the bad prediction. Curse of high dimensionality happens when the dimensionality increases. The volume of the space increases so fast that the available data become sparse which the scenario meets our case completely. Here we utilize Random Forest to rank our features to obtain most important features and apply them for further classification.

Random forest can be thought as an arbitrary number of decision trees and by inputting

every features from our data, those trees can vote for every feature and ultimately give us the most popular features. Random Forest overcome the overfitting drawback that decision trees may cause by applying bootstrap aggregation to multiple decision trees by a training set. Below shows the formula of Random Forest Prediction.

$$s = \frac{1}{K}\sum_{K=1}^{K} K^{th}$$

where K runs over every decision trees in the forest. Moreover, Random Forest can rank the most i-th importance from the features by assigning those features a score and it can be calculated with the out-of-bag error for each data point and averaged over the forest. The rank is very helpful for selecting features and avoiding curse of high dimensionality. After feature selection, we are able to implement further supervised learning models on the data efficiently.

5. Supervised Machine Learning Analysis—Classification

SVM is one kind of supervised learning models for data classification and regression analysis. It is to design a hyper plane that classifies all training vectors into two classes. For a given training dataset, the SVM algorithm is able to generate a model that assigns new data points into one of the two classes, which makes it a non-probabilistic linear classifier. The best separation would be the hyper plane that leaves the maximum margin from both classes. The points lying on the boundaries of the hyperplane are called support vectors.
Other than linear classification, SVM is also able to perform non-linear classification using the kernel techniques, which maps the input into high-dimensional spaces.

6. Evaluation

Specifically, cross-validation is used to evaluate model performance. Cross-validation is a model validation technique used for assessing model accuracy. One round of k fold cross-validation involves three steps: partitioning the data into k equal subsets; performing analysis on k-1 subsets; validating using the left 1 subset. In order to reduce variability, k rounds of validation are performed and the results are averaged over all rounds. After cross validation, the mean accuracy is calculated which indicates how good the model performance is.

IV. Result

1. Data Exploration

There are 191 patients and 272 features in our training data. Firstly, since there are too many features, here we randomly choose 15 continuous features which are feature 42 to feature 56 in our training data to conduct the data exploration and visualize the boxplot and histogram in Figure 1. For the boxplot, we can see that the 15 features distribute similar to each other; however, there are some features which have lots of outliers. For the histogram and the density estimation, we use feature 42 to 44 to display the distribution and we can tell that the distribution of these three features have lots of overlaps and do not have significant difference.

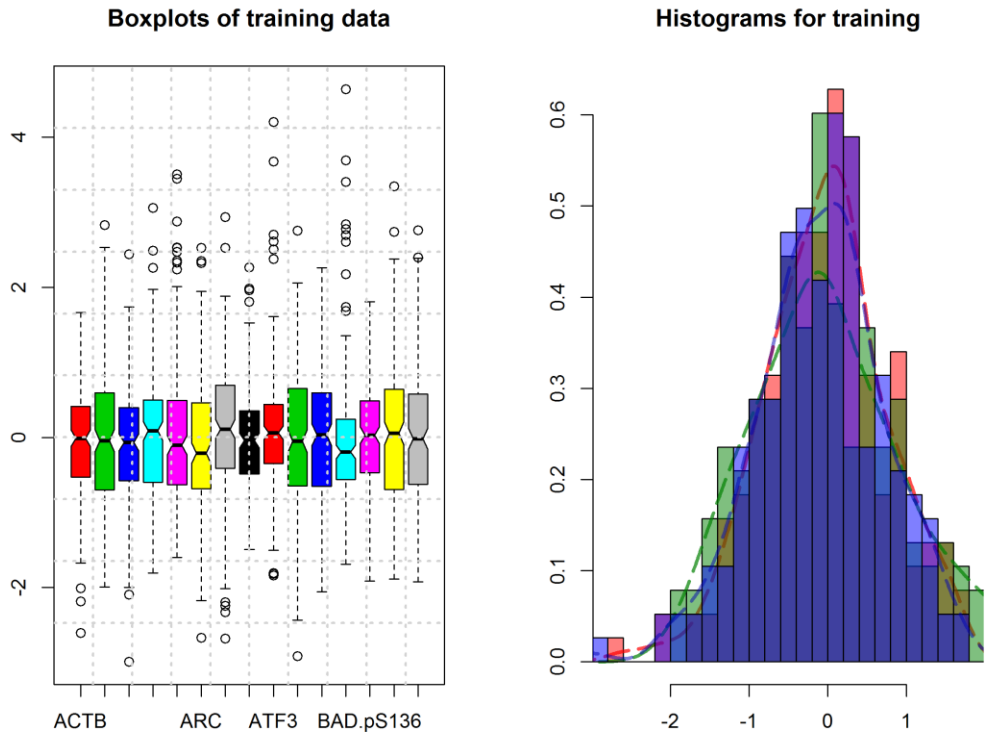


Figure 1. Boxplot of feature 42 to 56 and histogram of feature 42 to 44

Secondly, after boxplot and histogram, we draw the scatter plot of feature 42 to 45 in order to see the correlation between features which is shown in Figure 2. We find out that these four features are less correlated to each other, which means that the features are more independent from each other.

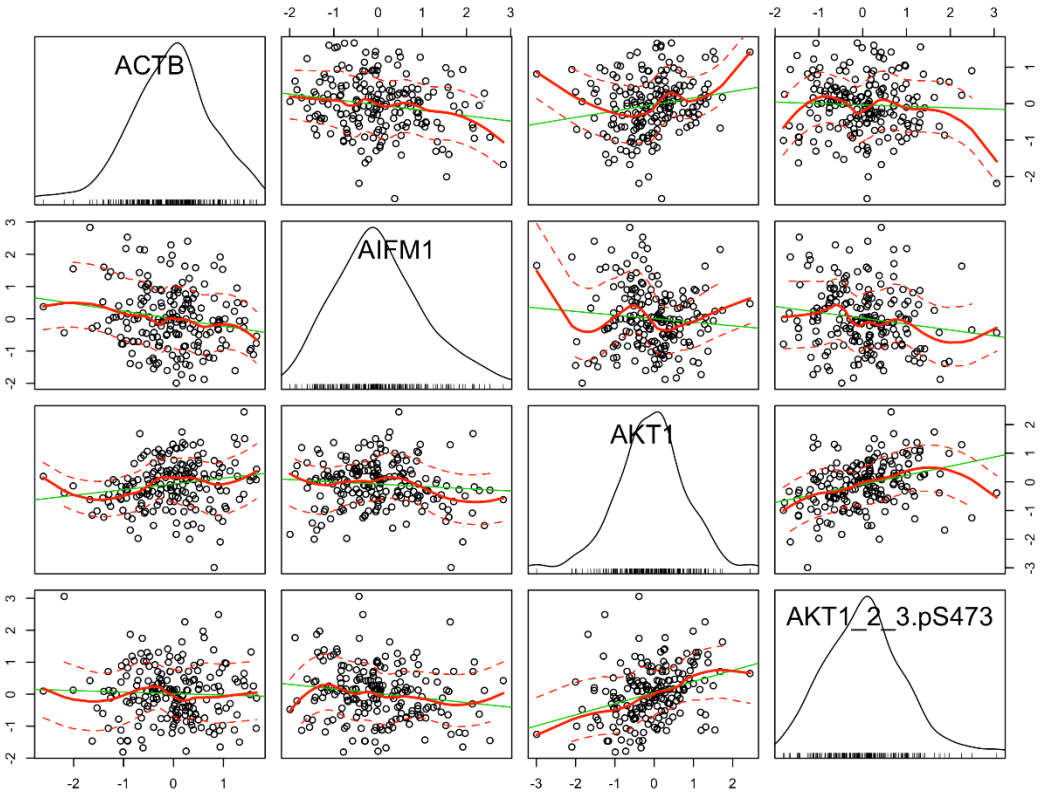


Figure 2. Scatter plot of feature 42 to 45

Thirdly, we use heat map to visualize the degree of the correlation between feature 42 to 56 which is displayed in Figure 3. We can tell that there is not lots of strong correlation between each other. Also, we can see that there is no significant cluster within these features.

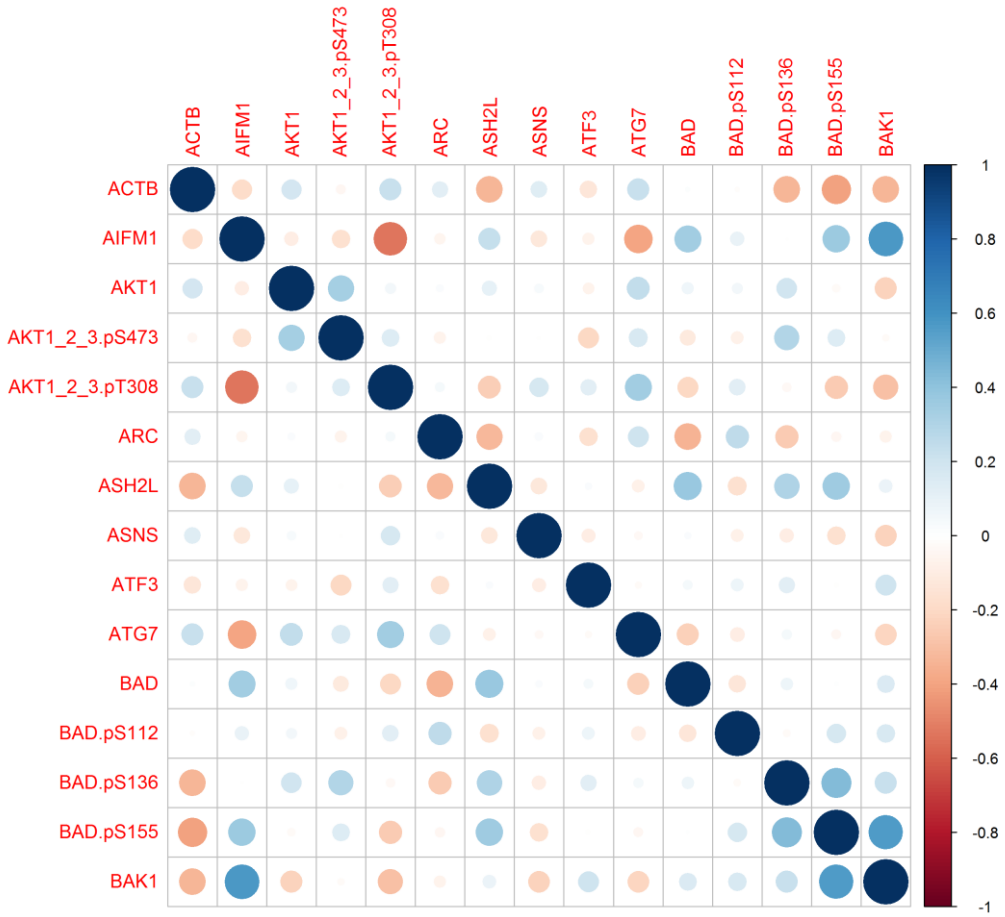


Figure 3. Heat map of feature 42 to 56

2. Unsupervised Machine Learning Analysis—Clustering

This section we implement k-means clustering for unsupervised machine learning. Firstly, we calculate the distance between 272 features within these 91 patients by Euclidean matrix and then compute 20 iterations of the within groups sum of squares and plot the curve as Figure 4. However, it is still hard to decide the number of clusters based on the plot. Then, we try to make 6 clusters to see if the clustering can give use some clues. The 6 clusters are as follow.

Cluster 1:

[1] "CASP9"	"CCND1"	"CDKN1A"	"CTNNA1"
[5] "ERBB3"	"FOXO3"	"IGF1R"	"IGFBP2"
[9] "NF2.pS518"	"NPM1"	"NRP1"	"ODC1"
[13] "PIK3CA"	"PIM2"	"PLAC1"	"PPARA"

[17] "PRKCB.I" "PTGS2" "SMAD5.pS463" "SMAD6"
[21] "SPP1" "STAT5A_B.pY694" "TCF4" "VHL"
[25] "YAP1"

Cluster 2:
[1] "ACTB" "AKT1_2_3.pT308" "ARC" "BAD.pS112"
[5] "CBL" "CTSG" "FN1" "ITGA2"
[9] "ITGAL" "ITGB3" "KIT" "LCK"
[13] "LEF1" "LYN" "MAPK1_3.pT202Y204" "MAPK14.pT180Y182"
[17] "NOTCH1.cl1744" "PRKCA" "PRKCA.pS657" "PTK2"
[21] "RAC1_2_3" "RPS6.pS235_236" "RPS6.pS240_244" "SRC"
[25] "SRC.pY416" "SRC.pY527" "STAT1" "TGM2"
[29] "VASP" "YAP1p" "YWHAZ"

Cluster 3:
[1] "AIFM1" "ASH2L" "BAD" "BAD.pS155"
[5] "BAK1" "BAX" "BCL2" "CASP9.cl330"
[9] "CCNB1" "CCNE1" "CD44" "COPS5"
[13] "CREB1" "DIABLO" "DLX1" "ELK1.pS383"
[17] "ERG" "FOXO3.S318_321" "GAB2.pY452" "H3K27Me3"
[21] "LSD1" "MTOR" "MTOR.pS2448" "MYC"
[25] "NPM1.3542" "NR4A1" "PARP1" "RB1"
[29] "SMAD1" "SMAD2.pS245" "SSBP2" "TRIM24"
[33] "WTAP" "ZNF296"

Cluster 4:
[1] "ASNS" "BAD.pS136" "BIRC2" "BIRC5"
[5] "BMI1" "CCNE2" "CDK1" "CDK2"
[9] "CDK4" "CLPP" "CTNNB1.pS33_37_41" "EGFR"
[13] "EGFR.pY992" "EIF2AK2.pT451" "ERBB2.pY1248"

"GRP78"
[17] "H3histon" "H3K4Me2" "H3K4Me3"
"HDAC1"
[21] "HDAC2" "HDAC3" "IRS1.pS1101"
"MCL1"
[25] "MDM4" "MSI2" "NF2"
"PIM1"
[29] "PPARG" "PPP2R2A_B_C_D" "PRKCD.pS664"
"SFN"
[33] "SMAD2.pS465" "SMAD4" "SOCS2"
"SQSTM0"
[37] "TAZ" "TAZ.pS89" "TNK1"
"TRIM62"

Cluster 5:
[1] "ATF3" "BCL2L1" "BCL2L11"
[4] "CASP3.cl175" "CASP7.cl198" "CASP9.cl315"
[7] "CAV1" "CCND3" "CD74"
[10] "CDKN2A" "CTNNB1" "EIF2S1.pS51."
[13] "ERBB2" "FOXO1.pT24_FOXO3.pT32" "GATA1"
[16] "GATA3" "HIF1A" "HSPA1A_L"
[19] "HSPB1" "JMJD6" "JUNB"
[22] "JUN.pS73" "KDR" "MAPT"
[25] "MET.pY1230_1234_1235" "NOTCH3" "PARP1.cl214"
[28] "PRKAA1_2" "PRKCB.II" "PRKCD.pS645"
[31] "CDKN1B" "RPS6KB1" "RPS6KB1.pT389"
[34] "STAT1.pY701" "STAT3" "STAT3.pS727"
[37] "STAT3.pY705" "STAT5A_B" "STAT6.pY641"
[40] "TP53" "TP53.pS15" "YWHAE"
[43] "ZNF346"

Cluster 6:
[1] "AKT1" "AKT1_2_3.pS473" "ATG7"
"BECN1"
[5] "BID" "BRAF" "CASP3"
"CASP8"
[9] "CREB1.pS133" "DUSP6" "EGLN1"
"EIF2AK2"

[13] "EIF2S1" "EIF4E" "Fli1"
"GAB2"
[17] "GAPDH" "GSKA_B" "GSKA_B.pS21_9"
"HNRNPK"
[21] "HSP90AA1_B1" "INPP5D" "INPPL1"
"LGALS3"
[25] "MAP2K1" "MAP2K1_2.pS217_221" "MAPK1"
"MAPK14"
[29] "MAPK9" "MDM2" "NCL"
"PA2G4"
[33] "PA2G4.pS65" "PA2G4.pT37_46" "PA2G4.pT70"
"PARK7"
[37] "PDK1" "PDK1.pS241" "PIK3R1_2"
"PRKAA1_2.pT172"
[41] "PRKCD.pT507" "CDKN1B.pS10" "PTEN"
"PTEN.pS380T382T383"
[45] "PTPN11" "RB1.pS807_811" "RELA"
"RPS6"
[49] "SIRT1" "SMAD2" "SMAD3"
"SMAD5"
[53] "SPI1" "STK11" "STMN1"
"TSC2"
[57] "XIAP" "XPO1"

After visualizing the clustering as Figure 5, we can find out that there is no distinct cluster in our features. It is really hard to tell the difference among these 6 clusters since they all gather in the same part in the graph. In other words, unsupervised machine learning does not give us any clue of predicting the result since curse of dimensionality displays completely here so that we should seek other method to conduct the further prediction.

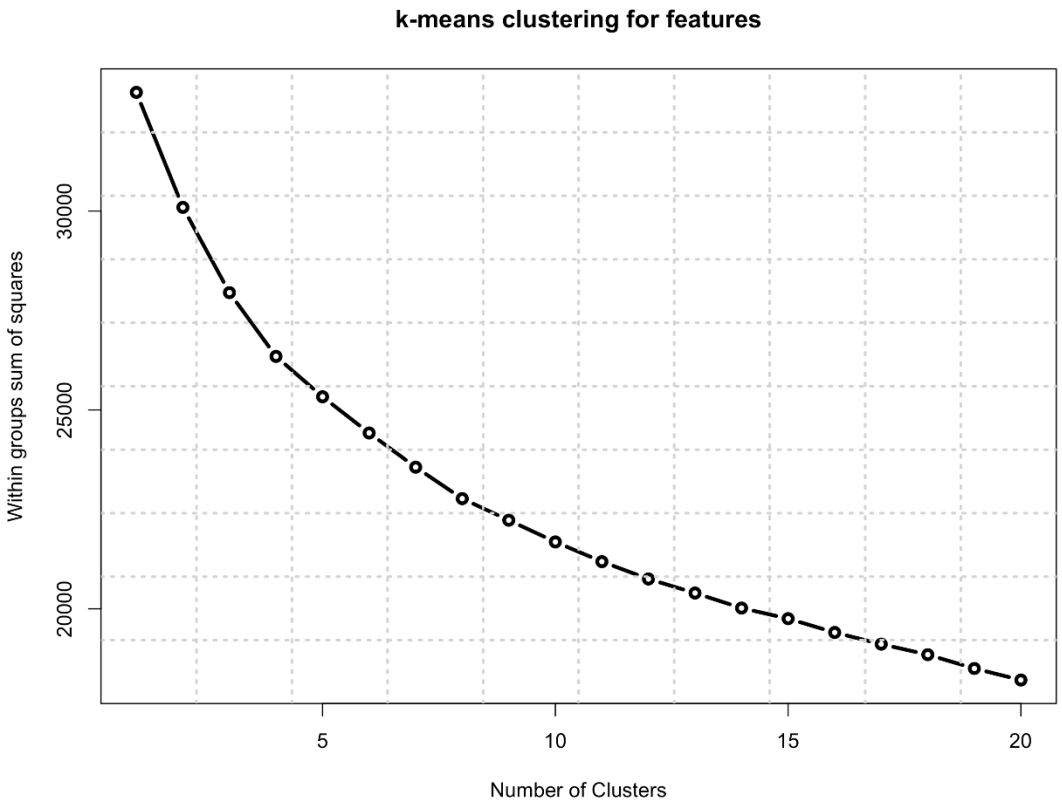


Figure 4. WSS plot of k-means clustering

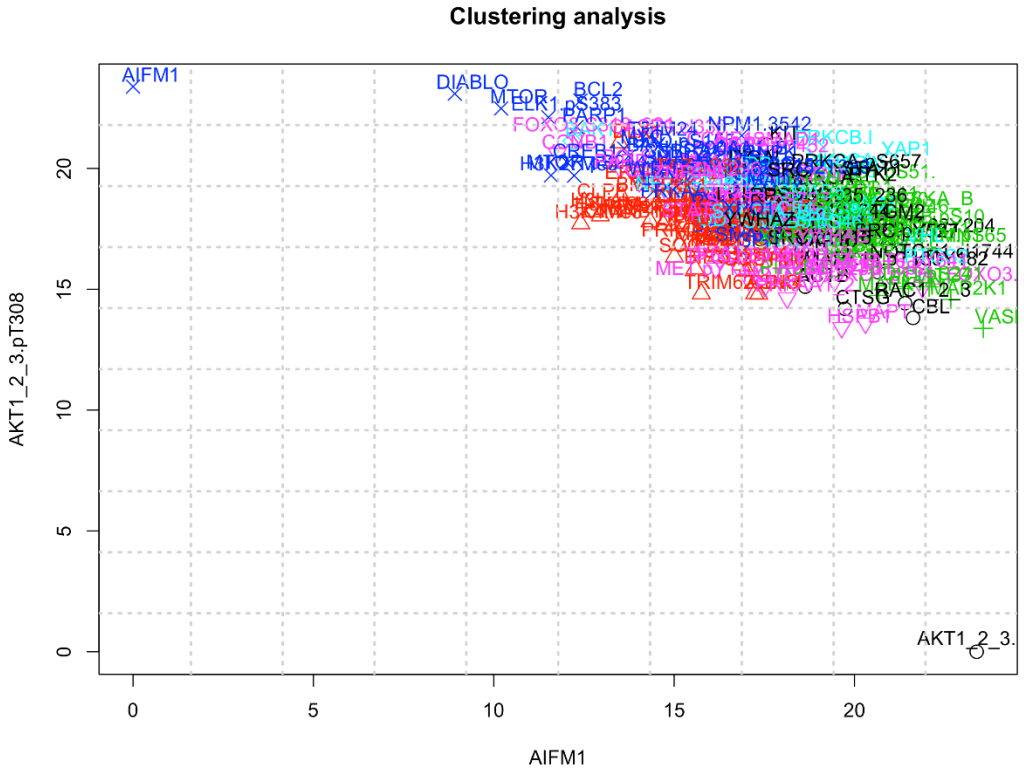


Figure 5. k-means clustering based on feature 42 to 272

3. Feature Selection—Random Forest

In order to perform feature selection, random forest is applied. R package "randomForest" is loaded, which allows us to build random forest on the entire training set. One round of tree growth includes generating 2500 trees and the trees are grown for 50 times. The importance of each feature is then ranked by the mean decrease gini. The outcome indicates that the range between the most important feature and the least important one is about (1.755, 0.164). When we plot all the features with their importance as y value, we observed a significant importance drop after the 8th top feature (Figure 6). Thus, we decided to choose the top 8 features for further classification tasks. Specifically, the 8 selected features are "PTEN.pS380T382T383" "PIK3CA" "ERG" "CTSG" "TGM2" "CASP7.cl198" "BAD" "CDKN2A".

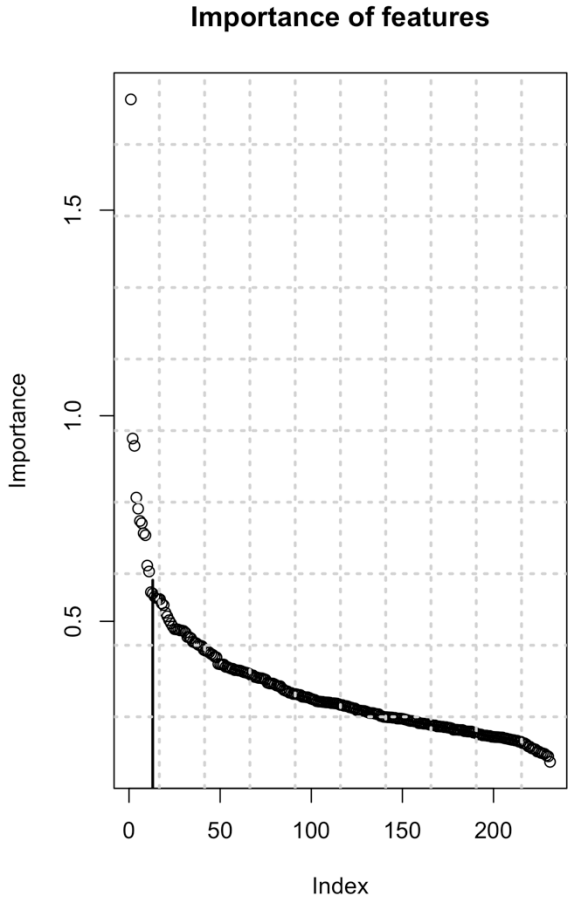


Figure 6. Plot of importance of features in descending order

4. Supervised Classification and evaluation

SVM is a popular classification technique for binary class prediction. Specifically, we split the entire training data in to 5 subsets and perform leave-one-out cross validation to test SVM prediction accuracy. The 8 features selected using random forest above is applied here as the only attributes to train to model. We use the training set to generate a SVM hyper plane and assign the test data to one of the two classes. After cross validation, the mean accuracy of SVM is about 75% (shown in table 1).

	y.test	
svm.pred	CR	RESISTANT
CR	30	7
RESISTANT	5	6

Table 1. SVM prediction accuracy table

V. Conclusion

The prediction of SVM model is about 75%, which is close to most ranked team who finished this task before. When we look at the raw data, we realize that the sample size is too small regarding the large amount of attributes involved. Curse of dimensionality is easy to happen. Thus, after data preprocessing and exploration, we decided to perform feature selection. Using random forest to generate top important rules is only one way of doing it. The feature selection step can be further refined after carefully reviewing the attributes and try to find more meaningful connections among them. Besides, we decided to choose the top 8 features which may lead to some accuracy lose. It is hard to determine the balance between more features and high accuracy.

After feature selection, SVM model is generated to give us a prediction accuracy of 75%. SVM is only one of the ways to do classification. Other models such as neuron network and logistic regression can also be tested for this task. We believe that after comparing different models, we can optimize the feature selection step and get higher prediction accuracy.

VI. Author Contributions

- Chia-Hsuan Chou: data exploration, clustering
- Venkata Prudhvi Raj Indana: background research, data preprocessing
- Jing Wang: random forest, SVM and evaluation
- Our group members are contributed evenly to the project.