**Report on first mini-project Math6380J.**

**Team: Ismailov Akhan. (One person).**

**Data: Drug Sensitivity Data by Cleave.**

**Problem.**

The data given mRNA, CNV, mutation, set collection 1 and 2. 129 training cell lines with IC50\_24, IC50\_48, IC50\_72. The problem is to predict IC50\_72 on 20 training cell lines.

**Motivation.**

Predicting cancer cell's drug sensitivity will help to personolize treatment for every person based on their genes. Attempts to solve this problem will create new methods in high-dimensional statistics.

**Part one.**

I report a score of 0.20063 (MSE) on Kaggle, using linear regression for top 10 PCA components.

My pipeline for this algorithm was:

1) Divide data into training and cross validation sets.

2) Apply PCA.

3) Learn weights using Linear Regression using Gradient Descent.

My data over all training will be X dimension 129 \* 36646 (18875+17771) and Y dimension 129. X combines mRNA and CNV.

1) Carefull division of 129 training samples into training and cross validation data sets. I divided it into 109 training and 20 cross validation. Because of small sample size I noticed that random shuffling of it gives very different statistics in each set, their variance can become 1.0~1.1 in training and 0.3~0.5 in cross validation sets. Because in small 20 size cross validation, variation of variance is high and can be very different from variance of whole 129 samples. So these two sets may become very different, which gives strange loss values during training for training and cross validation sets. Intuitively, second should be bigger during almost all steps but, it even becomes smaller by >= x1.5 times. I explain it by, suppose you predict mean value for all cross validation and mean for all training samples, MSE will be ~0.4 in cross valid and ~1.0 in training, due to low variance (around 0.4) in cross validation set and high (around 1.0) variance in training set. And it is very problematic to understand training process if cross validation performs considerably better. Second problem is that trained weights for training data may not generalize to cross validation data set if both sets represent different parts of the whole 129 training samples, which makes decision of whether the model is good difficult. To choose right shuffling of sample data I looped through seed values for random generator that gave me nearly same variance and mean for both sets.

2) Reducing dimensionality using PCA. PCA reduces dimension by finding the most correlated directions in high-dimensional space with the IC50\_72. I applied PCA on 149 by 36646 data. Where 18875 columns from mRNA and 17771 from CNV. I chose number of components by looking at results from cross validation set. Finally, stopped at 10 components, which is small enough to not overfit training data set and consequently generalize to cross validation and test sets of size 20.

3) And finally applying linear regression on 10 feautures extracted from PCA to get weights on parameters. In this case we do not need penalties such as Lasso, Ridge or Elastic Net. Because they do not give improvement, based on emperical results from cross validation, for 10 features over 129 data samples. 10 is small enough number to generalize without penalties on weights.

**Part two.**

After performing linear classification using 10 PCA components, I understood that PCA sacrifices interpretation over performance. In next attempt, which is more interpretable, I used Forward Stepwise Variable Selection using Ridge regression with different lambda (shrinkage) parameters and applying linear prediction on selected variables. So, this technique will reduce dimension from current 36646 as well. Its performance was noticeably worse than previous approach on my cross validation set. Firstly, I chose penalty because it has potential to decrease variance and improve generalization. Secondly, I chose ridge in this stepwise variable selection because I did not need to find sparse solution (over currently small amount of selected variables at each step) that gives lasso L1 penalty.

I tried to analyse the data further to understand why variable selection did not work quite well. The main idea behind variable selection is that at every step it chooses variable that is the most correlated to LC50\_72 given current chosen variables. And the trap is that top 100 correlated variables in the training set only share 1 common variable with the top 100 correlated variables in the cross validation set, in my division of the sample data. So, not suprisingly learned coefficients and variables in training set performed poorly in cross validation set.

In total, I expected that PCA will perform better than linear prediction using selected variables. Because in some sense variable selection chooses directions that best correlate with the data. Where directions limited to be (0, 0, 0, ..., 1, ..., 0, 0). Here, 1 is chosen variable (dimension, direction, paramater, feature) over 36646 given. While PCA also chooses directions that best correlate with the data, and these directions not limited to be sparse. However, in linear prediction using selected variables we could be using different transformations on these selected variables, which cannot be done in this small training set (variance increases, generalization fails). I think that further understanding the domain of the problem can give better features and possible transormations of the data from current raw gene sequences.

**Conclusion.**

In addition to apply well-known techniques, firstly, I found that smart division of small training set into training and cross validation sets is needed, because if we choose cross validation set which has different statistics from training set generalization to cross validation set and assesment of the model will fail. Secondly, I understood that selection of stand-alone variables by their correlation to IC50\_72 is problematic given small Ns (109 and 20) and very large P (36646), because top 100 correlated with IC50\_72 variables in training set and top 100 in cross validation set share only 1 common variable, which rises problems during generalization of the model into cross validation set. I hope that these findings and understandings can help development of the better models in the future.