# Machine Learning Analysis Project

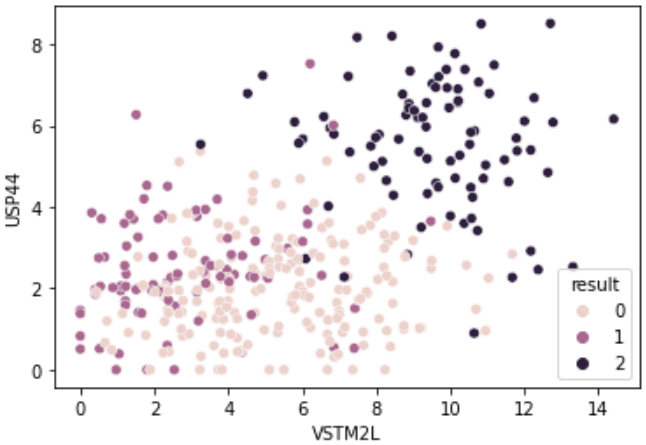
By: Atoosa Ayazbkahsh

## Data Analysis

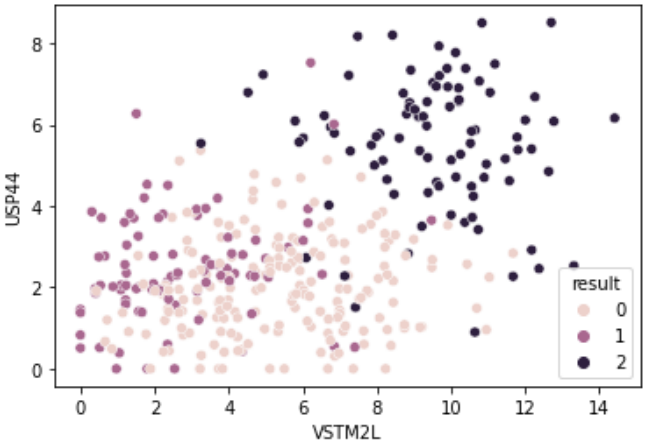
The training dataset was in 16340 rows × 1846 columns involving 16340 features and 1840 samples. The feature/sample ratio was significantly high, and overfitting was probable. So, feature selection was preferable to achieve a reliable ML model.

## Machine Learning Algorithm

Because of the low number of samples in comparison with features deep-learning approach could not be implemented. The train Data were separated into training and cross-validation sections *with train_test_split.* So, 20% of the total training data was sectioned into the cross-validation. The logistic Regression algorithm was chosen via the 'sag' solver due to its speed. The cross-validation accuracy was 99.73%. Visualized data of two random features for Actual and ML-predicted results can be seen below.



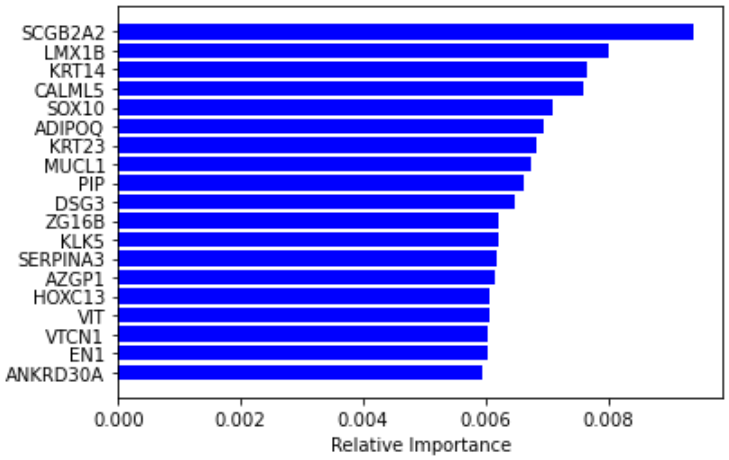

Actual Data



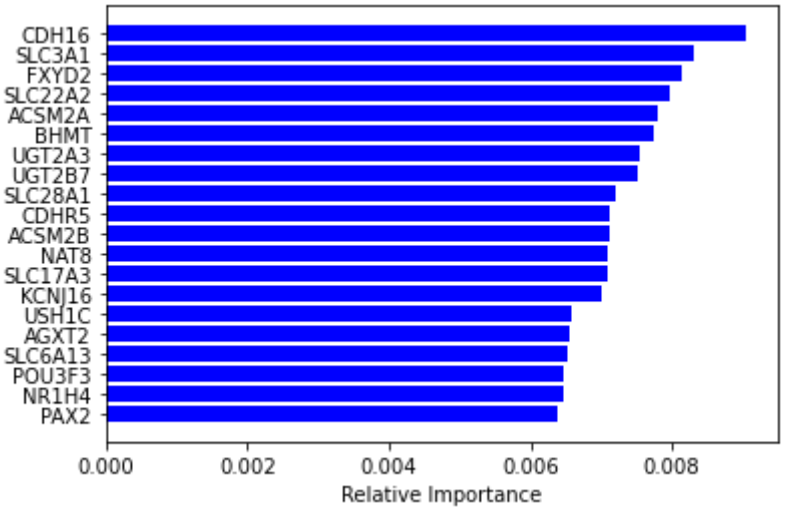

Prediction Result

## Feature Selection

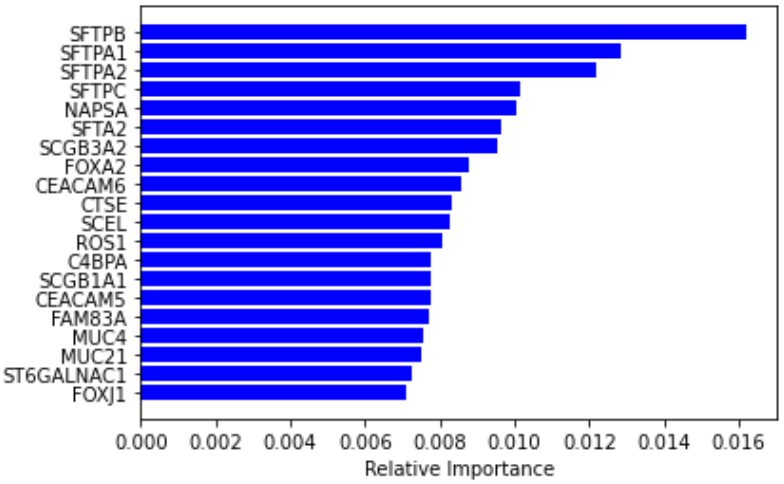
The ".coef_" attribute was used to reach a correlation between features and labels. The top 20 features related to "breast invasive carcinoma" is demonstrated below.



The top 20 features related to “kidney clear cell carcinoma” is demonstrated below.




The top 20 features related to “lung adenocarcinoma” is demonstrated below.




The top 100 features of each 3 labels were added up, and duplicates were deleted. Then via Logistic Regression, the model was trained with the shrunk data and predicted the cross-validation data. This time accuracy reached 100%. The result was saved in the **ML_predict_Ayazbakhsh.csv** file.

```
In [40]:
1 LR1.predict(X_test)
2 round(LR1.score(X_test,y_test),4)

Out[40]: 1.0
```

Furthermore, unsupervised **Agglomerative Clustering** was implemented too, and its accuracy on training data was 98.64%.

```
In [61]:
1 clustering = AgglomerativeClustering(n_clusters = 3).fit_predict(X_train)

In [62]:
1 clustering

Out[62]: array([2, 0, 1, ..., 0, 0, 1], dtype=int64)

In [57]:
1 accuracy_score(clustering, y_train)

Out[57]: 0.986449864498645
```