**Capstone Project – Data Wrangling**

**Data set**

The data set for this project is from the UCI Machine Learning Repository.

<https://archive.ics.uci.edu/ml/datasets/p53+Mutants>

**Description of the dataset in the website**

The data consists of 16,772 instances and 5409 attributes per instance. Attributes 1-4826 represent 2D electrostatic and surface based features. 4827-5408 represents 3D distance based features. The 5409th attribute is the class attribute which is either active or inactive. The class labels are to be interpreted as follows: ‘active’ represents transcriptionally competent, active p53 whereas the ‘inactive’ label represents cancerous, inactive p53.

**Download and Processing**

The data set was downloaded as a .data file and read in as a csv file. It was more than 1 GB in size and it took a long time to import and store as a data frame. Processing the file for any information took a long time and I decided to read the file in chunks and do data cleaning/wrangling to the chunks.

The first 1000 rows of the data set was then downloaded and all the processing was done in this chunk - df. After checking df.info and dtypes, df.head() gave a preview to the first 5 rows. The first thing noticed was, missing values in certain rows/columns, an entire column of NaN and that the columns did not have names/descriptions. The number of ‘active’ entries were 11 vs 989 ‘inactive’ labels.

To minimize the confusion between numbered columns and rows, I attached the prefix ‘2D’ to the first 4826 column numbers and the prefix ‘3D’ to 4827th – 5408th column using string methods and labeled the 5409th column as ‘Type’ as it indicated the class label. The instance tags (description for each row or mutation) for the dataset was in a separate file and was also imported and added as a column ‘mutations’ to the data frame. The column with NaNs was dropped. There were also some rows with ‘?’. Since, there were no descriptions of the features available and there were 5408 of them for each mutation, it seemed best to drop all missing values after converting them to NaNs. This brought the ‘inactive’ entries to 976.

The ‘mutations’ column contains the amino information on each mutation (row). For eg:

‘%a119e\_l125p’ denotes 2 point mutations separated by ‘\_’, the first one is ‘a119e’ and the second one is ‘l125p’. The ‘a119e’ is read as – ‘a’ at position 119 of the protein chain is replaced by ‘e’.

The dataset has 2, 3, 4, 5 and 6 point mutations in this chunk. So, I added a column ‘count’ showing the number of mutations. For each row entry (mutation), I am adding distance features to the data set i.e. the distance between the first and second mutations (difference in position numbers), second and third, and so on. So, the single mutations will have a distance of 0, the 2 point will have 1 entry for distance, 3 point will have 2 entries, 4 point will have 3 entries and so on. So 7 columns named ‘distance1’, ‘distance2’,…,’distance7’ were added to the data set and the mutations were analysed for ‘count’ and then the ‘distance’ features were appended based on count.

**Next Steps**

- correlation of features

- Pandas profiling

- PCA

- Data visualization

- Model