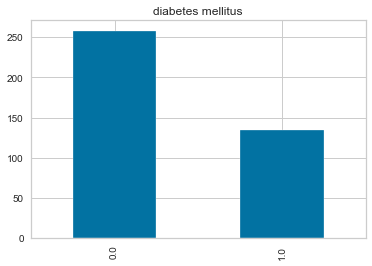
**IMPLEMENTATION REPORT**

**Introduction**

This document is a brief description of what kinds of visualization tools were used during our diabetes prevalence analytics pipeline,what insights each tool gives us into our data. Some of these insights might be already known (but perhaps not yet proven) while others are completely new and surprising and also key recommendations to the key stakeholders on what these insights mean and what they should be used for.

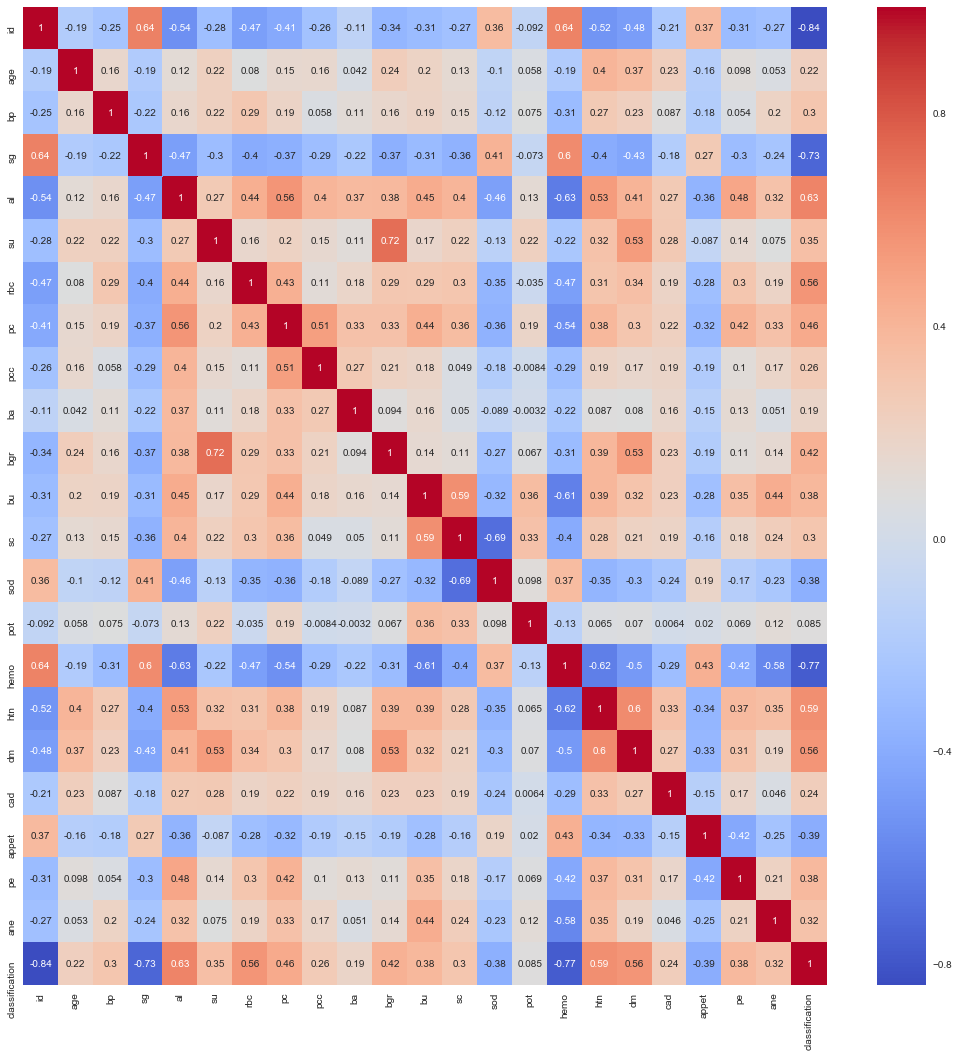
To begin with is a bar plot which is very good and well known for comparing categorical data



From the context of our diabetes prevalence prediction, the bar plot above shows that of 400 patients whose records were taken at Apollo Hospitals, 134 of them have diabetes mellitus while 258 patients did not have diabetes .

Since our dataset has around 20 features and 400 rows. A good way to quickly check correlations among these 20 features is by using heat map.

**A correlation heatmap showing correlation among all the features with diabetes mellitus (dm)**

****

Colour palletes are used to represent what kindof correlation exist between which features. In this case; Red means positive while blue means negative. The stronger the color, the larger the correlation magnitude.

Checking out correlations of other features with diabetes mellitus (dm) from the heat map above, it is found out that features age, bp, al, su, rbc, pc, pcc, ba, bgr, bu, sc, htn, cad, pe, ane and classification have a positive correlation unlike the rest of the features.

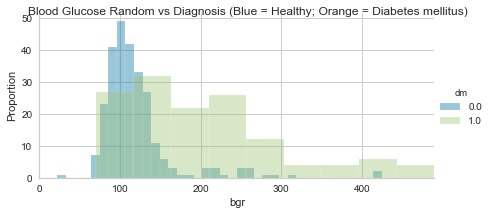
Among the features with the correlation su, bgr ,htn, class have a stronger correlation with diabetes mellitus(dm) which is not the case with other features.

Now, to decide which features to drop, the heatmap above was analysed and those features with the lowest correlation with diabetes mellitus(dm) in our diabetes prevalence prediction were dropped.

Therefore, the features id, sg, sod, pot, appet and hemo fall under this category hence were dropped before proceeding with more visualization of the kidney\_disease.csv dataset.

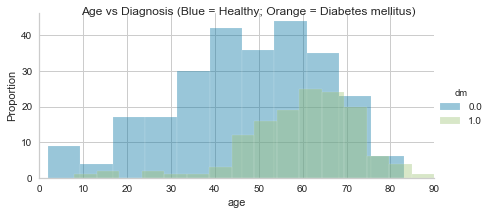
The reason for dropping of the above features will further enable our future model to easily learn with the most relevant features (features that have a positive correlation with diabetes mellitus (dm)) and thereafter evaluate its performance using the three performance metrics of precision, recall and first score.

**A histogram showing of blood glucose random ( bgr ) against diagnosis**



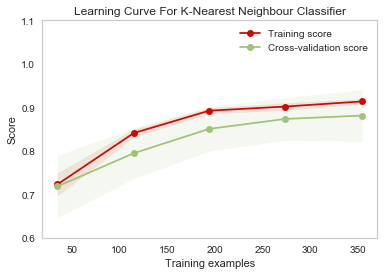
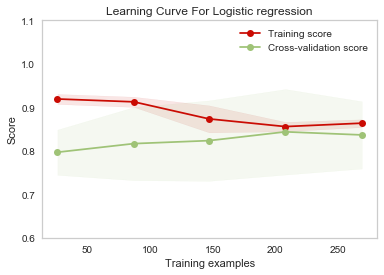
From this visualization, it is clear that the higher the blood glucose levels, the higher the chances of the patient bearing diabetes mellitus.

**A histogram showing of age against diagnosis**



From the visualization it can be said that the more a patient ages, the higher the chances of acquiring diabetes mellitus. Typically a person from around the age of 45 years are in greater risk.

**Learning Curve**



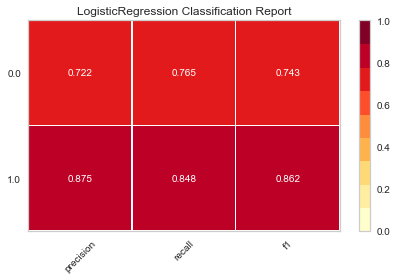
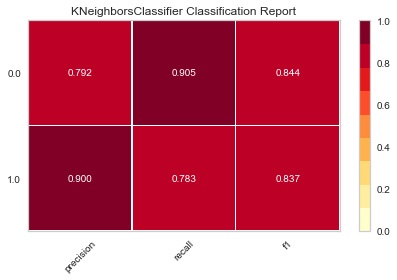
The Training and validation score for **Logistic Regression** Model converges to a low score (value ~ 0.8). The gap between the curves is bigger, when all the training data is used. Therefore, we will not benefit from adding more data to train the model since it has a high bias characteristic. This curve depicts an under-fit model; model is not able to learn efficiently on training data.

We will probably have to use an estimator or a parametrization of the current estimator that can learn more complex concepts (i.e. has a lower bias).

For **K Nearest Neighbor Classifier**, the training score is greater than the validation score for the maximum number of training samples and converges towards a higher accuracy score; this implies that the model would benefit from adding more training data to increase generalization. This curve is also a good-fit, and has a low bias and high variance characteristic.

Therefore, this means that the K-NN model is more efficient than the Logistic Regression model.

**YellowBrick Classification report**



The classification reports shows a representation of the main classification metrics (precision, recall and f1 score) on per-class basis (1.0, 0.0). Thus giving a more detailed and deeper intuitive summary of the classifier behavior over global accuracy.

**Logistic Regression**has unbalanced classification metrics for its classes. Axis for Class 1.0 is redder than 0.0 thus it is has higher precision, recall and f1 score. This imbalance weakens the overall performance of this model.

**K Nearest Neighbor** classifierhas relatively balanced classification metrics for its classes. Axis for Class 1.0 and 0.0 have strong precision, recall and f1 score. This strengthens the overall performance of this model.

Overall, K Nearest Classifier is redder than Logistic Regression hence it is a. more efficient model selection

**Deployed API with Flask on localhost Server.**

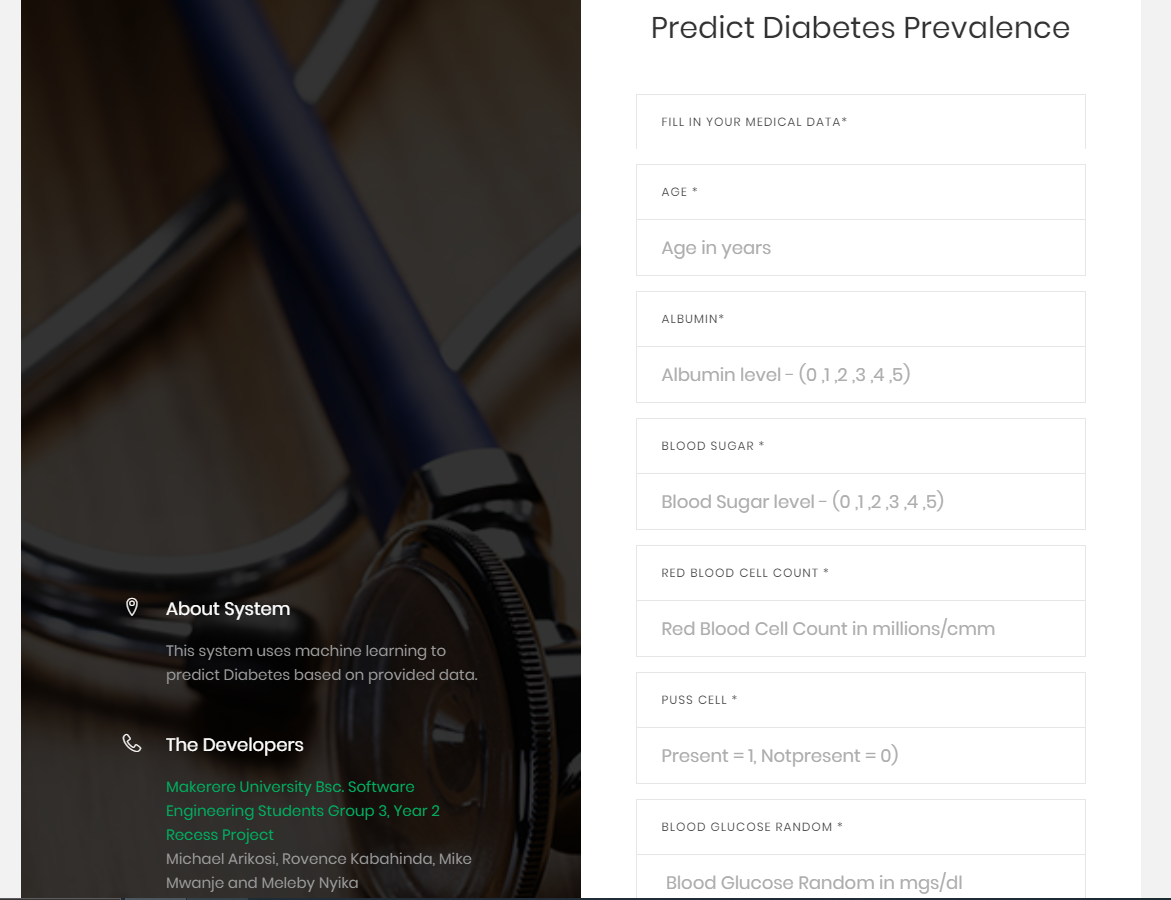
****

Figure 1 User Requested to enter required data to run the Diabetes Test Prediction

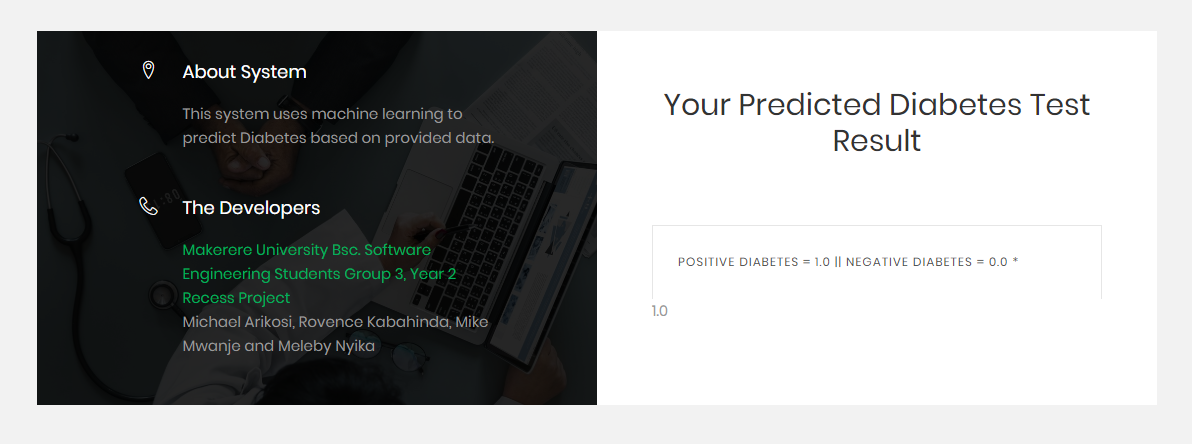
****

Figure 2 Results returned to the user after prediction API has finished processing

**Recommendations**

1. Add more observations (rows)to the dataset to improve overall accuracy and the predictive strength of the model as it has more data from which to learn.
2. Add more complete observations to the dataset having nonull cells. This reduces the negative implications of backward filling for null values in the data. Backward filling causes uncertain distribution of data, which may skew the accuracy of the model. More accurate and real data helps to build a more realistic and accurate model.
3. We recommend dropping the Null rows when the dataset grows in size and has numerous observations with completely filled features. This will increase accuracy and predictive strength by working with only real and accurate data.
4. Increase the balance between the classification features (dm) i.e. Ratio of (0.0:1.0) in the. This reduces the negative implications of oversampling data in K- Neighbor Classifier model as it tries to balance the classification Classes.
5. Drop features that are irrelevant to the training of the model. These features will skew the accuracy as they cause a high bias trade-off in the learning of the model. These features have a very low correlation coefficient relative to (dm) ie less than 0.3.