Description of Data

There were two datasets, one labeled for train and the other for test. They contained ten features including work type, gender, age, hypertension, smoking status, marriage status, residence type, heart disease, glucose level, and bmo.

In the training dataset we had 783 patients with stroke and 42617 without. It is worth noting that 'stroke' patients only comprised 1.8% of the dataset. The majority of patients in the study (57.2%) had been employed privately. About 1% of those with private employment had strokes while those self employed were about half as prevalent (0.57%) and government jobs even less (0.20%) and those with work type 'children' at .004%. We saw that a slightly higher percentage of male patients endured strokes than female (1.98% to 1.68%). Those who were of age 45 or older comprised 95.53% of the population of those with strokes.

Total Count

```
+------+-----+
|stroke|count|
+------+-----+
|     1|  783|
|     0|42617|
+------+-----+
```

Work Type Distribution

```
+-------------+---------+-----------------+
|    work_type|work_type|work_type_percent|
+-------------+---------+-----------------+
| Never_worked|      177| 0.40783410138249|
|Self-employed|     6793|15.65207373271889|
|      Private|    24834|57.22119815668203|
|     children|     6156|14.18433179723502|
|     Govt_job|     5440|12.53456221198157|
+-------------+---------+-----------------+
```

Work Type with Strokes

```
+-------------+---------+-----------------+
|    work_type|work_type|work_type_percent|
+-------------+---------+-----------------+
|Self-employed|      251| 0.57834101382488|
|      Private|      441| 1.01612903225806|
|     children|        2| 0.00460829493088|
|     Govt_job|       89| 0.20506912442396|
+-------------+---------+-----------------+
```

Males with strokes

```
+------+------------+------------------+
|gender|gender_count|        percentage|
+------+------------+------------------+
|  Male|         352|1.9860076732114647|
+------+------------+------------------+
```

Females with strokes

```
+------+------------+------------------+
|gender|gender_count|        percentage|
+------+------------+------------------+
|Female|         431|1.6793298266121177|
+------+------------+------------------+
```

Percent 45 and over of those with strokes

```
+------------------+
|percentage_over_45|
+------------------+
| 95.53001277139208|
+------------------+
```

Preprocessing Steps

We first removed the ID column as it was not relevant. We did SQL queries for each feature to count nulls. We added the mean BMI for those patients without BMI information and ‘No Status’ to the smokers with null fields.

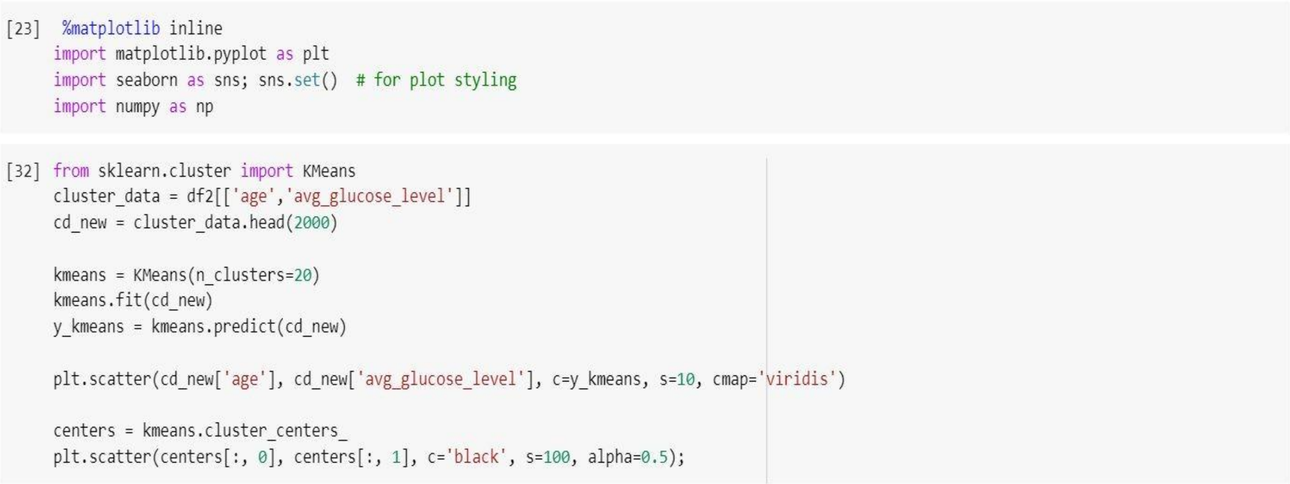
```
+------------+
|smoking_null|
+------------+
|       13292|
+------------+

+--------+
|bmi_null|
+--------+
|    1462|
+--------+

+-----------+
|gender_null|
+-----------+
|          0|
+-----------+

+--------+
|age_null|
+--------+
|       0|
+--------+

+-----------------+
|hypertension_null|
+-----------------+
|                0|
+-----------------+

+------------------+
|heart_disease_null|
+------------------+
|                 0|
+------------------+

+-----------------+
|ever_married_null|
+-----------------+
|                0|
+-----------------+

+--------------+
|work_type_null|
+--------------+
|             0|
+--------------+

+-------------------+
|Residence_type_null|
+-------------------+
|                  0|
+-------------------+

+----------------+
|avg_glucose_null| +----------------+
|               0|
+----------------+
```

Results

Description of approaches

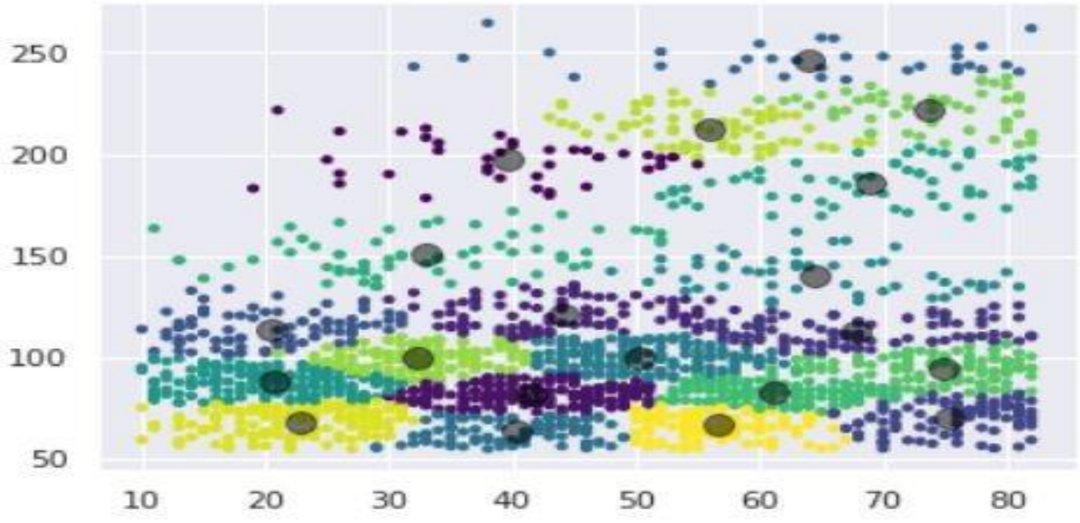
Clustering Approach

The clustering approach is used to analyze the structure of the data. In which data points divides into a number of groups such that data points in the same groups are more similar to other data points in the same group. In this data we formed K-means clustering between age of patients and average glucose level of patients of first 2000 data points and found that the average glucose level is high amongst patient whose age is above 40 as clusters formed at high glucose level values after the age of 40 in the clustering data.



```
[23]  %matplotlib inline
     import matplotlib.pyplot as plt
     import seaborn as sns; sns.set()  # for plot styling
     import numpy as np
```

```
[32] from sklearn.cluster import KMeans
     cluster_data = df2[['age','avg_glucose_level']]
     cd_new = cluster_data.head(2000)

     kmeans = KMeans(n_clusters=20)
     kmeans.fit(cd_new)
     y_kmeans = kmeans.predict(cd_new)

     plt.scatter(cd_new['age'], cd_new['avg_glucose_level'], c=y_kmeans, s=10, cmap='viridis')

     centers = kmeans.cluster_centers_
     plt.scatter(centers[:, 0], centers[:, 1], c='black', s=100, alpha=0.5);
```



Classifier for “Danger” or “No Danger” of Stroke

We created a predictor using DecisionTreeClassifier in pyspark with the imputed values and also without. We split the data into 70% training and 30% testing. We used the standard inputs for pyspark which were a max depth of 5, maximum bins of 32, and minimum instances per node as 1. If the prediction was 1, this meant the patient was in danger of a stroke and if it was 0 it meant the patient was not in danger. We had very high accuracy for these, 98.% with imputation and 97.8% without. However, the area under the ROC curve was below .7 for both classifiers indicating that our models were not as impressive as the accuracy implies. This is due to the imbalanced dataset where there are far more patients without strokes than with strokes.

Predictor to Predict the Probability

We used LinearRegression in pyspark machine learning package to predict the probability of a stroke. As with the decision tree, we used 70% training and 30% testing from the dataset labeled ‘train.’ We set the maximum number of iterations to 100 and the convergence tolerance to 10e-6. With and without using imputation both of our R^2 values were very close to 0 indicating that our model performed very poorly.

Decision Tree With imputation

```
+----------+--------------------+------+--------------------+
|prediction|         probability|stroke|            features|
+----------+--------------------+------+--------------------+
|       0.0|[0.99248940036341...|     0|(16,[0,2,5,6,10,1...|
|       0.0|[0.99248940036341...|     0|(16,[1,2,8,10,11,...|
|       0.0|[0.99248940036341...|     0|(16,[1,2,3,5,6,10...|
|       0.0|[0.99248940036341...|     0|(16,[1,2,8,10,11,...|
|       0.0|[0.99248940036341...|     0|(16,[0,2,3,5,7,10...|
|       0.0|[0.99248940036341...|     0|(16,[1,2,3,6,10,1...|
|       0.0|[0.99248940036341...|     0|(16,[0,2,4,5,6,10...|
|       0.0|[0.99248940036341...|     0|(16,[0,2,6,11,12,...|
|       0.0|[0.99248940036341...|     0|(16,[0,2,5,6,10,1...|
|       0.0|[0.95117493472584...|     0|(16,[0,2,5,6,10,1...| +----------+-----
---------------+------+--------------------+
only showing top 10 rows

A Decision Tree algorithm had an accuracy of: 98.31%
Test Area Under ROC: 0.5343429341583658
```

Decision Tree Without imputation

```
+----------+--------------------+------+--------------------+
|prediction|         probability|stroke|            features|
+----------+--------------------+------+--------------------+
|       0.0|[0.88469601677148...|     0|[1.0,0.0,80.0,0.0...|
|       0.0|[0.99069028156221...|     0|(15,[0,2,5,6,11,1...|
|       0.0|[0.99069028156221...|     0|(15,[1,2,5,6,11,1...|
|       0.0|[0.99069028156221...|     0|(15,[0,2,5,6,10,1...|
|       0.0|[0.96218020022246...|     0|(15,[1,2,5,8,11,1...|
|       0.0|[0.88469601677148...|     0|[1.0,0.0,77.0,1.0...|
|       0.0|[0.99069028156221...|     0|(15,[0,2,5,6,10,1...|
|       0.0|[0.99069028156221...|     0|(15,[0,2,5,8,11,1...|
|       0.0|[0.99069028156221...|     0|(15,[0,2,5,6,11,1...|
|       0.0|[0.99069028156221...|     0|(15,[1,2,6,11,12,...| +----------+-----
---------------+------+--------------------+
only showing top 10 rows

A Decision Tree algorithm had an accuracy of: 97.87%
Test Area Under ROC: 0.642065828907934
```

Linear Regression With Imputation

```
+--------------------+------+--------------------+
|          prediction|stroke|            features|
+--------------------+------+--------------------+
|0.018183016008959746|     0|(16,[0,2,8,11,12,...|
|0.018183016008959746|     0|(16,[0,2,8,10,11,...|
|0.018183016008959746|     0|(16,[0,2,8,10,11,...|
|0.018183016008959746|     0|(16,[0,2,8,10,11,...|
|0.018183016008959746|     0|(16,[0,2,8,11,12,...| +---
-----------------+------+--------------------+ only
showing top 5 rows

R Squared (R2) on test data = -1.27512e-05 numIterations:
1
objectiveHistory: [0.5000000000000001]
+--------------------+
|           residuals|
+--------------------+
|-0.01818301600895...|
|-0.01818301600895...|
|-0.01818301600895...|
|-0.01818301600895...|
|-0.01818301600895...|
|-0.01818301600895...|
|-0.01818301600895...|
|-0.01818301600895...|
|-0.01818301600895...|
|-0.01818301600895...|
|-0.01818301600895...|
|-0.01818301600895...|
|-0.01818301600895...|
|-0.01818301600895...|
|-0.01818301600895...|
|-0.01818301600895...|
|-0.01818301600895...|
|-0.01818301600895...|
|-0.01818301600895...|
|-0.01818301600895...| +---
-----------------+ only
showing top 20 rows
 RMSE:
0.133613
r2: -0.000000
```

Linear Regression With Imputation

```
+-------------------+------+--------------------+
|         prediction|stroke|            features|
+-------------------+------+--------------------+
|0.01801624415456559|     0|(15,[0,2,9,11,12,...|
|0.01801624415456559|     0|(15,[0,2,9,11,12,...|
|0.01801624415456559|     0|(15,[0,2,9,11,12,...|
|0.01801624415456559|     0|(15,[0,2,9,11,12,...|
|0.01801624415456559|     0|(15,[0,2,9,11,12,...| +---
----------------+------+--------------------+ only
showing top 5 rows

R Squared (R2) on test data = -0.000376239 numIterations:
1
objectiveHistory: [0.5000000000000001]
+--------------------+
|           residuals|
+--------------------+
|-0.01801624415456559|
|-0.01801624415456559|
|-0.01801624415456559|
|-0.01801624415456559|
|-0.01801624415456559|
|-0.01801624415456559|
|-0.01801624415456559|
|-0.01801624415456559|
|-0.01801624415456559|
|-0.01801624415456559|
|-0.01801624415456559|
|-0.01801624415456559|
|-0.01801624415456559|
|-0.01801624415456559|
|-0.01801624415456559|
|-0.01801624415456559|
|-0.01801624415456559|
|-0.01801624415456559|
|-0.01801624415456559|
|-0.01801624415456559| +---
-----------------+ only
showing top 20 rows

RMSE: 0.133010
r2: -0.000000
```

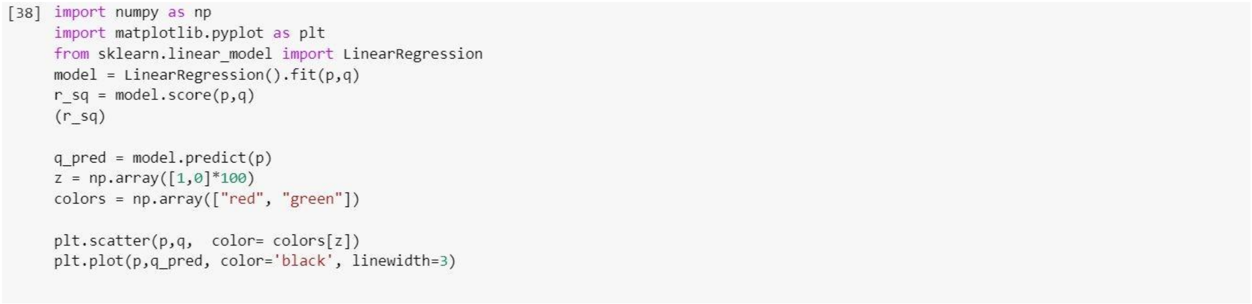
Comparison of three approaches

Clustering is used to find general information of a dataset and visualize the data without actually creating a classifier. Regression is used for predicting something that is not discrete and classification is used to predict which category something belongs to. In this case, we used linear regression to output a number from 0 to 1 and used that as the probability of a patient having a stroke. We used decision trees to predict which class (at risk or not at risk) each patient belonged to.

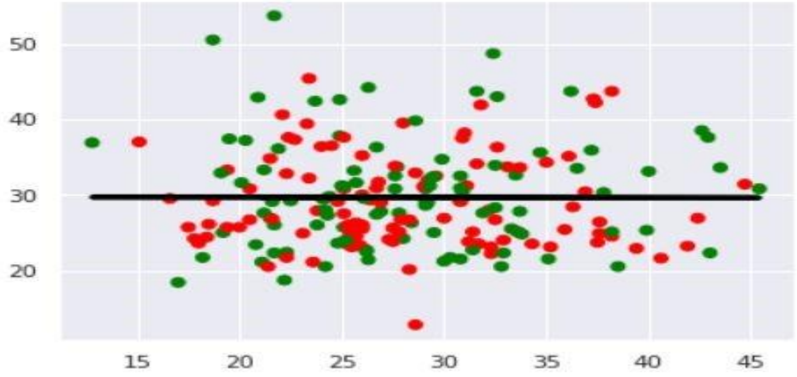
Effect of Imputation

Imputation had a minor effect on the decision tree as well as the regression model. The predictions were both slightly worse without imputation. However, our models did not perform very well in either case. A more pronounced improvement may have been found if we used techniques to balance the dataset.

We took data values of 'bmi' with imputation and without imputation and formed a linear regression model between them and found that the values of data (red) without imputation are closer to being linear than the values (green) with imputation which show that there will be no such effect of imputation on final result as compared to the result from without imputation.



```
[38] import numpy as np
     import matplotlib.pyplot as plt
     from sklearn.linear_model import LinearRegression
     model = LinearRegression().fit(p,q)
     r_sq = model.score(p,q)
     (r_sq)

     q_pred = model.predict(p)
     z = np.array([1,0]*100)
     colors = np.array(["red", "green"])

     plt.scatter(p,q,  color= colors[z])
     plt.plot(p,q_pred, color='black', linewidth=3)
```



Discussion

This project is similar to that of a data scientist as they would use multiple techniques to understand and evaluate a dataset. They would also used methods such as imputation to prevent losing valuable data. Additionally, data scientists would have to deal with issues such as having an imbalanced dataset in terms of the target values.