AI-AUG-Minor Project

by Siddhartha Sinha

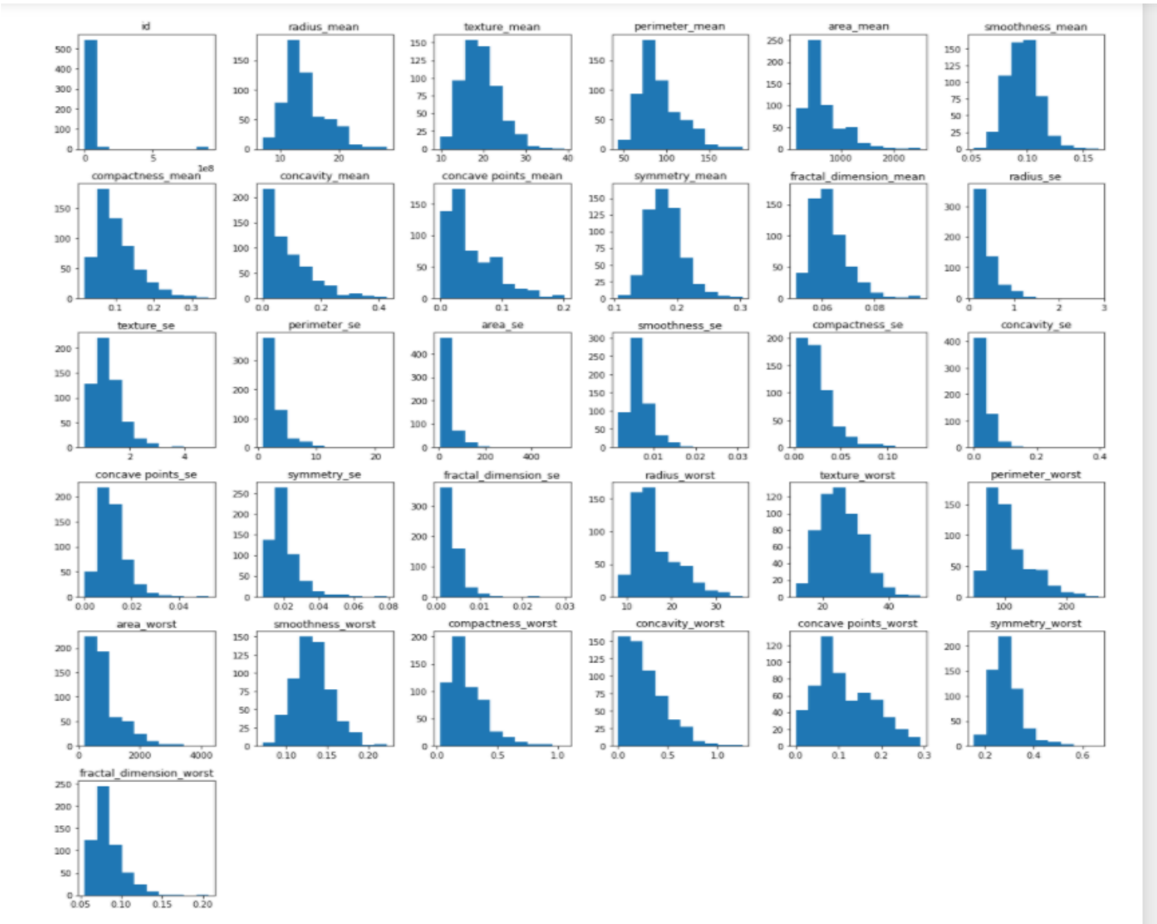
Contact: - 9939733269

Email: - siddharthsinha8@gmail.com

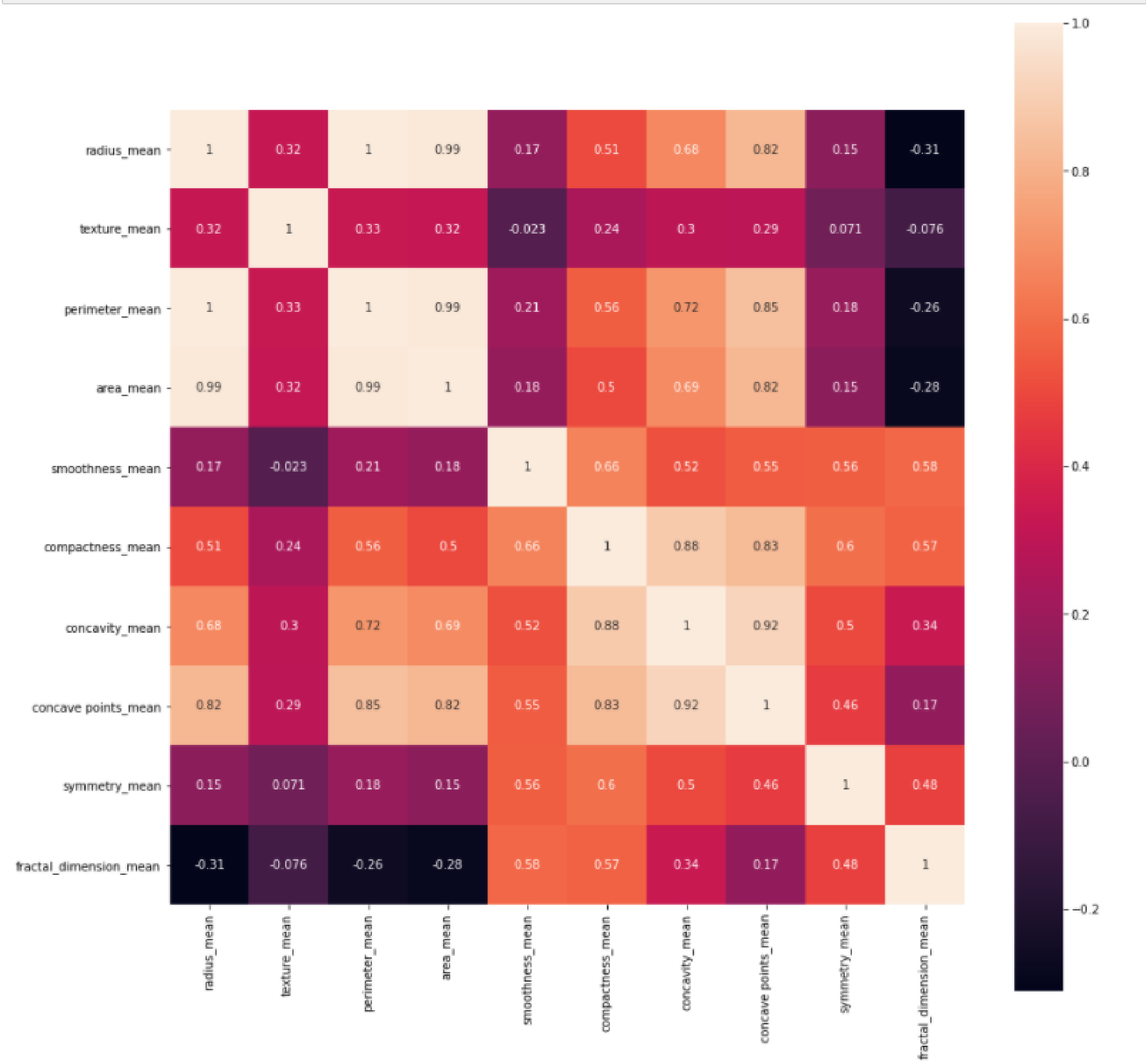
1) How cleaning/EDA was performed?

: - Using **"df.info ()"** statement and overview of dataset used is shown. It had 33 Columns and 569 Rows. The 33rd Column had all null values rest of the columns didn't have any null values. The Last Column was removed using **"del df['Unnamed: 32']"** function. The first column contained patient id which was of no utility and would spoil the results of the dataset. Dataset was then split into x features and y target variable using **"x=df.iloc[:,2:].values"** and **"y=df.iloc[:,1].values"** . For Centring the data I used **"y=le_x1.fit_transform(y)"** then split transform target variable in test and train set using **"x_train, x_test, y_train, y_test = train_test_split(x,y,test_size=0.2, random_state=0)"** then I centred the train and test data for the features using **"x_train = sc.fit_transform(x_train)"** and **"x_train = sc.fit_transform(x_train)".**

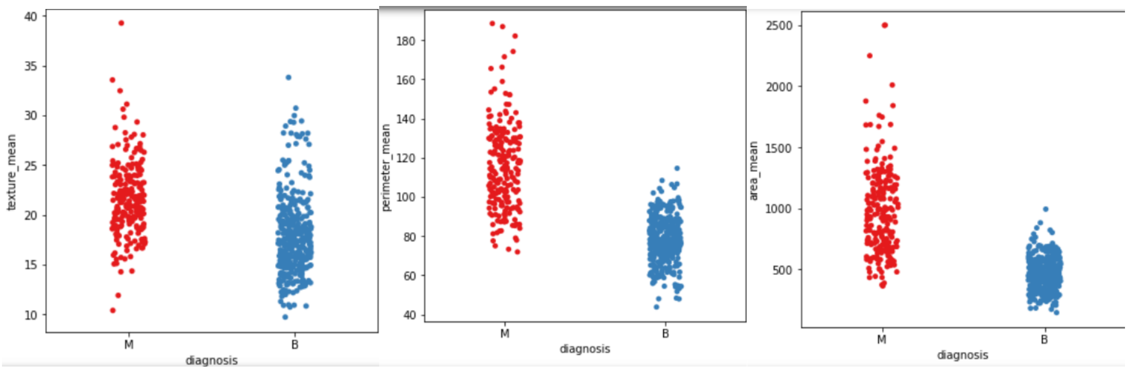
Data Visualization was performed using **"df.hist(bins=10,figsize=(20,20),grid=False)"** which showed

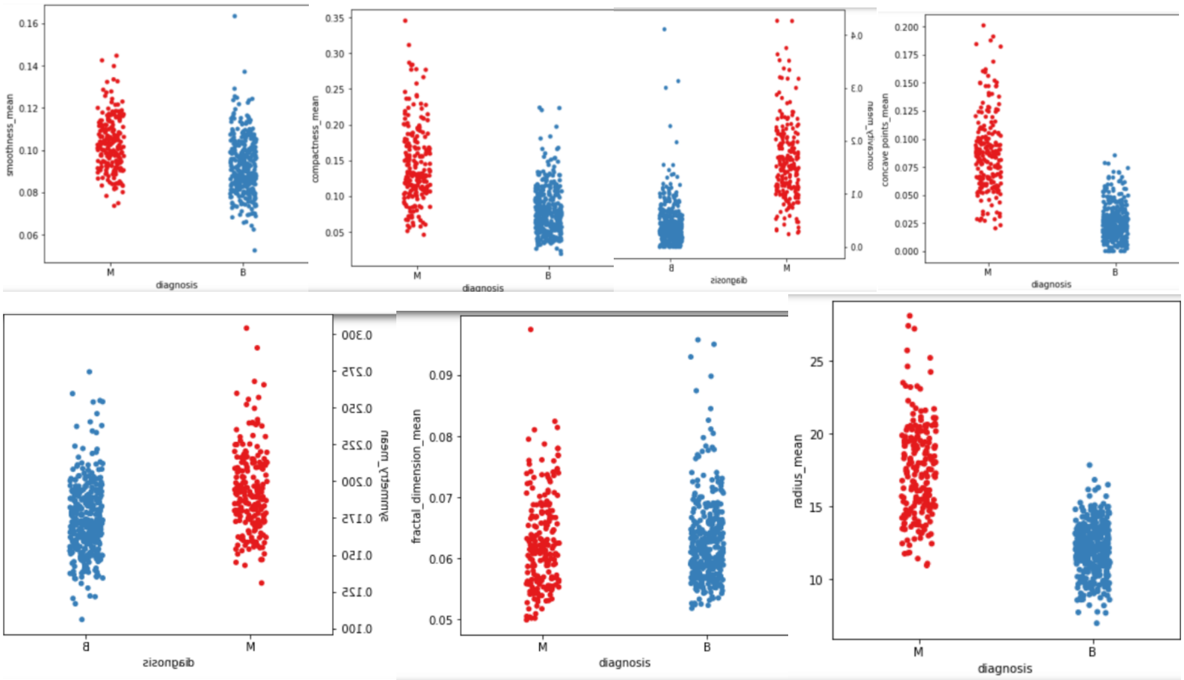


And **“heat = sb.heatmap(df[f_mean].corr(), vmax=1, square=True, annot=True)”**

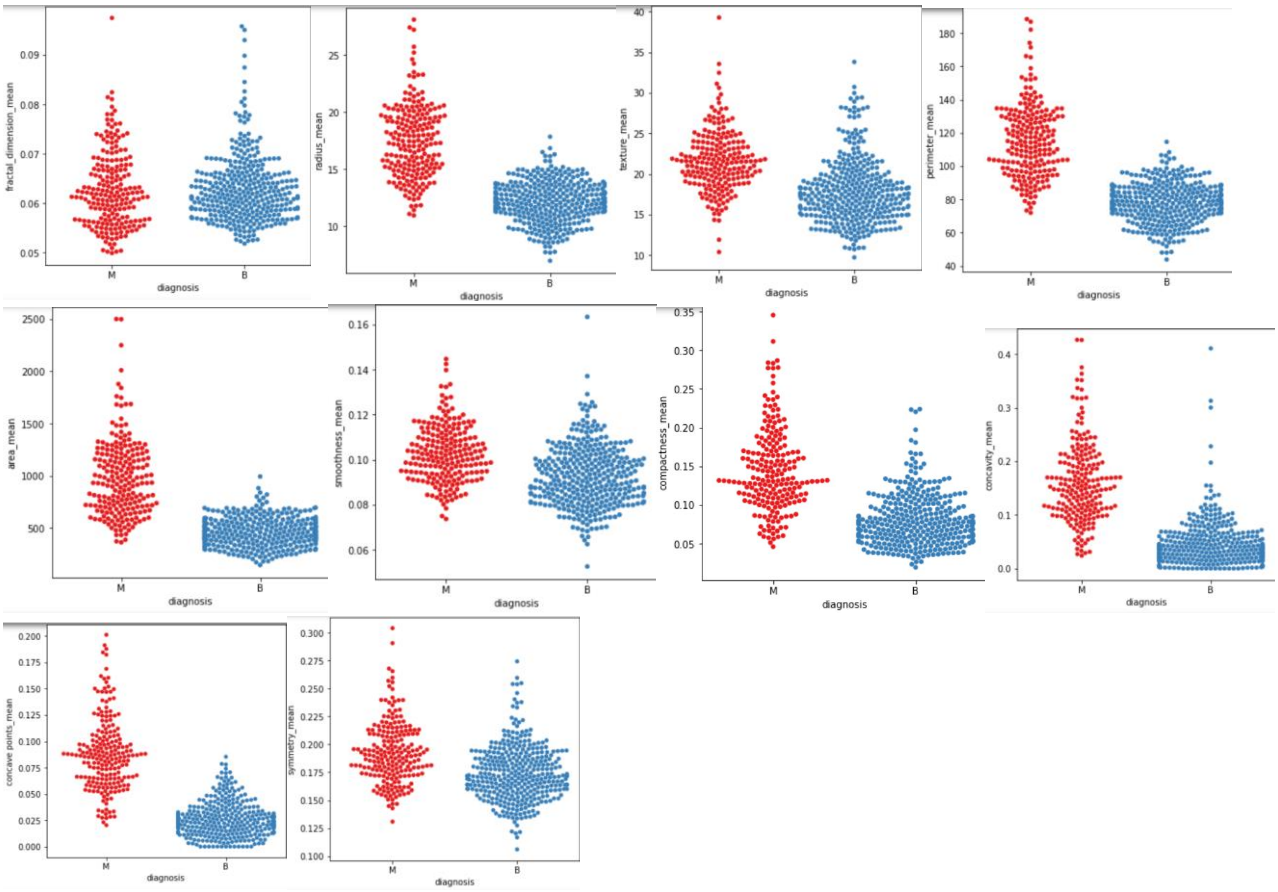


sb.stripplot(x='diagnosis', y= columns, data= df, jitter=True, palette = 'Set1');





sb.swarmplot(x='diagnosis', y= columns, data= df, palette = 'Set1')



2)Your independent and dependent feature?

: - Independent Feature: diagnosis

Dependent Feature:

radius_mean
texture_mean
perimeter_mean
area_mean
smoothness_mean
compactness_mean
concavity_mean
concave points_mean
symmetry_mean
fractal_dimension_mean
radius_se
texture_se
perimeter_se
area_se
smoothness_se
compactness_se
concavity_se
concave points_se
symmetry_se
fractal_dimension_se
radius_worst
texture_worst
perimeter_worst
area_worst
smoothness_worst
compactness_worst
concavity_worst
concave points_worst
symmetry_worst
fractal_dimension_worst

3) Why and how selection/engineering/scaling were performed?

:- Feature scaling had to be done as the featured ranged from magnitude of 10^1 to 10^3 so using

```
from sklearn.preprocessing import StandardScaler
sc=StandardScaler()
x_train = sc.fit_transform(x_train)
x_test = sc.transform(x_test)
```

Feature scaling was done.

4) Which Activation function was chosen and why?

: - ReLU Activation function was chosen for the hidden layers while Sigmoid Function was chosen for the Output layer. The rectified linear activation function overcomes the vanishing gradient problem (output the input directly if it is positive, otherwise, it will output zero), allowing models to learn faster and perform better.

Sigmoid is function is a logistic function and by setting the parameter we can classify Values into Discrete Values

5) Which optimizer was chosen and why?

Adam optimization is used as it is a stochastic gradient descent method that is based on adaptive estimation of first-order and second-order moments. This algorithms leverages the power of adaptive learning rates methods to find individual learning rates for each parameter. It also has advantages of Adagrad, which works really well in settings with sparse gradients, but struggles in non-convex optimization of neural networks.

6) Which Neural Network and why? Describe Your Neural structuring.

Dense Artificial Neural Network is built because the layers are fully connected and they provides learning features from all the combinations of the features of the previous layer, whereas a convolutional layer relies on consistent features with a small repetitive field.

Structure: Input layer = 30 inputs + 1 bias and 16 output
1st layer (Hidden layer) = 16 inputs (31/2 =15.5 ==16) and 16 outputs
2nd layer (Hidden layer) = 16 inputs and 16 outputs
3rd layer (Output layer) = 16 inputs and 1 output