1. **Assumptions-**

Feature are mixture of both categorical and continuous so I need to treat them accordingly.

After doing multiple iteration bivariate and multivariate analysis, we have some of basic insights:

- Features x067, x094, x095, and x096 are all zeros so they will not having importance in prediction so removed them.
- Some feature have more than 75% null values.
- All continuous features are having null values so replaced them with mean of the feature.
- 41 out of 303 are having missing values.
- 45 features are having only 3 values across the data set, so considering them as categorical.
- On the complete dataset output, variable "Y" is ranging from 300 to 840.
- After calculating the feature importance these are the most important features:
    - X06
    - X025
    - X026
    - X027
    - X060
    - X067

2. **Description of your methodology and solution path:**

Step 1: Import the libraries: load basic package such as numpy, pandas, sklearn, matplotlib
Step 2: Load the data-set: spilt that into input variable and output variable
Step 3: **Feature Engineering -** Missing value Treatment:

a) - If it has more than 75% of missing values then it is advised to delete data point. One has to make sure that after we have deleted the data, there is no addition of bias.

b) - Replace all the NaN values with either mean, median or most frequent value. This is an approximation, which can add variance to the data set. However, the loss of the data can be negated by this method, which yields better results compared to removal of rows and columns.

Step 4: Continuous/Categorical Identification: Divide the data into categorical and continuous variable after removing indecisive features which have more null values and which column have constant values.
Step 5: **Train/Test** Split and Encode the Categorical data: Split the data into train and test set. Do the one hot encoding in categorical feature to convert them into numerical features.
Step 6: Feature Normalization: Normalize the continuous features with z values(x-mean/sigma). Apply that normalizer on train, test continuous features, and Combine the Cont and cat features
Step 7: Model Training: Perform regression algorithm on the training set.
Step 8: Model Testing/Tuning: Test the model on the test dataset. In addition, save the model after successful validation. Parameters are tuning basis on performance of model.
Step 9: Model Validation and Deployment: Validate the model in 4-fold validation to see the model is not overfitting and it is stable and write all the steps in sequence to execute unseen data. I would prefer pipeline(sklearn) method to execute all the commands.

3. **List of algorithms and techniques you used**

I tried multiple regression algorithm such as

- Linear Regression
- MLP(multi layered network of perceptron)
- Gradient Boosting
- Random Forest
- XGBoost

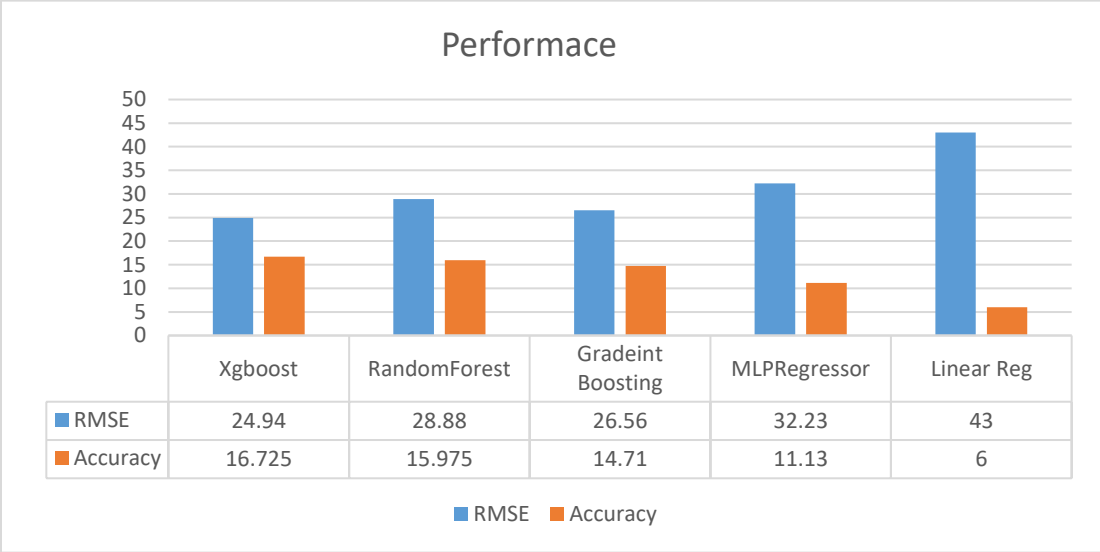
4. List of tools and frameworks you used:

Majorly python and its libraries are used. Packages such as

- Numpy
- Matplotlib
- sklearn/Xgboost
- pandas

5. Results and evaluation of your models

Xgboost is performing better than the other models with 25 rmse and 17% accuracy



Instructions to run:

1. Please use python3.5 and above version.
2. Please provide all the pickle file and save them at the same location you are running the file.
3. It takes 40 sec to execute the code and 1-2 gb of ram.
4. Command to run the file

python whub.py input_file

Requirement	Version
Pandas	0.21.1
sklearn	0.19.1
numpy	1.13.1
python	3.5.4