Data Scientist Test Report

1. List of any assumptions that you made

- Nearly 30+ features have missing values between 30% to 90%. For this test, it is assumed this cannot be addressed and dropped features with 75% or more missing values.
- Related to the earlier point, data imputation is assumed as the method to fix the missing values for features with less than 75% missing values.
- Nearly 98% of the features have outliers (considering a IQR * 1.5) between 20 to 17000 outliers. It is assumed this is the nature of data. Still some experiments will be done on outlier methods for statistical reasons and presented in the evaluation results.
- Nearly 98% of the features are skewed with non-normal distribution. It is assumed this is the nature of data. Still some experiments will be done by applying data transformation methods to reduce the skew.
- Based on the above points and data visualization it is assumed there is a non-linear relationship between features and the target variable and modelling will be done using methods like tree-based, ensemble and neural networks.

2. Description of your methodology and solution path

- Problem Statement understanding
  - Understood the functional requirements, evaluation metrics, and testing requirements from the document provided
  - Understood the deliverables required
- Setting up experimentation framework
  - Created an experimentation framework in excel to gather details and metrics from different experiments to be conducted and select the model with the best performance metrics
  - Selected RMSE, MAE, R2, Adjust R2, Accuracy (prediction error < 3) to be tracked
- Exploratory data analysis
  - Analyzed and visualized missing values, outliers, feature-feature correlations, feature-target correlations, skewness, descriptive statistics
  - Used tools like Dtale, pandas, matploib to understand and visualize the data
- Splitting data into Train and Test datasets
  - Data was randomly split into train and test datasets in the ratio 75:25
- Data cleaning
  - Performed basic data cleaning by removing features with very low variance (<0.25) and features with > 75% missing values

    - Created multiple options of imputation and outlier treatment for experimentation of performance during model buidling
        - Imputation - evaluated between mean, median, most-frequent strategies using cross validation with a regressor algorithm and comparing RMSE to select the best method. Also created an option of using KNN based imputation
        - Outlier treatment – created options for using outlier capping using percentile, IQR cutoff and Isolation Forest outlier removal methods and no outlier removal.
- Data visualization and Model building selection
    - Used Principal component analysis to reduce dimensionality to visualize the relationship between the features and target variable.
    - Used scatter plot and correlation matrix to understand strength of relationship between feature-target and feature-feature
    - Understood the strong non-linear relationship between the features and target
    - The non-linear relationship demanded usage of tree based, ensemble based, neural network-based models to be experimented
- Feature Selection
    - Created multiple options for experimentation of performance during model building. Below are the options
    - Performed mutual information analysis between features and target to select the top k features. Selected k based on cross validation using a regressor and RMSE score.
    - Performed PCA and reduce the dimensionality with 95% explained variance
    - All feature selection was also selected as an option
- Feature Transformation
    - As algorithms like Lasso and Neural networks work with normally distributed data, explored PowerTransform(box cox) and QuantileTransform methods to transform data during model building.
- Model Training, Validation and Tuning
    - Created baseline model using Lasso Regression. Conducted experiments with different combinations of outlier treatment, feature selection methods and established baselines
    - Conducted experiments with RandomForest regression algorithm with different combinations of imputation, outlier treatment, feature selection along with cross validation and hyperparameter tuning
    - Conducted experiments with XGboost algoithm with different combinations of imputation, outlier treatment, feature selection along with cross validation and hyperparameter tuning
    - Conducted experiments with Neural network algoithm with different combinations of imputation, outlier treatment, feature selection along with cross validation and hyperparameter tuning
    - Model validation and tuning was done using RMSE. However other metrics mean absolute error, r2, adjusted r2, accuracy were also tracked.
    - Model performance tuning and bias-variance tradeoff was done using cross validation and Hyperparameter tuning

- Model Evaluation & Selection
    - For all the experiments conducted the models were evaluated on the test dataset and metrics captured.
    - The cross-validation test scores from the earlier step were verified to be closer to the test scores indicating the validation process was stable.
    - The best model was selected based on the best RMSE score. The accuracy, mean absolute error, r2 scores were also verified to be consistent with RMSE
- Model Deployment/Submission for running
    - Tested the deployment of the model using docker on the test dataset

- Provided a docker library for testing the model with a different hold-out data set

3. List of algorithms and techniques you used

- Exploratory data analysis
  - Pandas understand descriptive statistics and missing values
  - Matploib visualization for outliers, skewness, correlations

- Data Cleaning
  - Low variance treatment
    - Variance Thresholding
  - Missing value treatment:
    - Pandas to remove features with high % of null values
  - Imputation methods experimented. Selected Most-frequent using cross-validation
    - Mean
    - Median
    - Most-frequent
    - KNN imputation methods
  - Outlier removal methods. Selected No outlier removal method based on RMSE
    - Percentile based(capping feature between 10^{th} and 95^{th} percentile)
    - IQR based outlier removal
    - Isolation Forest based outlier removal
    - No outlier removal

- Data Visualization
  - PCA and Matplotlib to understand relationship between features and target variable
  - Matplotlib to understand correlations and mutual information

- Feature Selection
  - Experimented below methods, selected all features(after cleaning) based on RMSE score
    - Mutual Information technique between features and target
      - Select best number of features using cross validation using Linear Regression
    - Pearson correlation matrix to identify and remove correlation between features
    - Principal component Analysis -to reduce the dimensionality
    - Select All features (after data cleaning)

- Feature Transformation
    - Experimented the below methods for Lasso and Multi Layer perceptron algorithms. Ensemble methods like Random Forest, XGB work well with non-normal distributions.
        - PowerTransform (converting non-normal distribution to normal distribution)
        - QuantileTransfrom(converting non-normal distribution to normal distribution)

- Model Building

    - Lasso Regression
        - Cross Validation, Hyperparameter tuning using GridSearchCV
    - Random Forest
        - Cross Validation, Hyperparameter tuning using RandomSearchCV
    - XGboost
        - Cross Validation, Hyperparameter tuning using RandomSearchCV, Early stopping methods
    - Neural network(ANN/MLP)
        - Hyperparameter tuning, Callbacks(Early stopping, Model Checkpoint)

4. List of tools and frameworks you used

- Python
- Jupyter notebook
- Pandas
- Scikit-learn
- XGBoost
- Keras
- Tensorflow
- Matplotlib
- Docker

5. Results and evaluation of your models

The below table summarizes the details of the all the experiments conducted. The columns have details of the specific strategy/option experimented. Based on the experiments a XGBoost model trained on 20000 low depth(depth 3) trees with missing value removal strategy of 75% missing values, imputation strategy of imputation with most frequent values in the features, not removing any outliers, variance thresholding of features at 0.25, no feature selection (beyond variance thresholding) and no feature transformation is the best model. The Test RMSE is 26.333 and Test Accuracy (predictions vs actual < 3) is 14.4. The MAE is 18.879, R2 score 0.95.

Due to a strong non-linear relationship between the features and target, this is the best performance that could be achieved with tried experimentation and tuning. The R2 score of 0.95 indicates a strong model however the accuracy of predictions (predictions vs actual < 3) is low at 14.4.

In the below table, the columns can be interpreted as

- Missing value removal - % of null values considered in the features for removal. Values – 75%, None (No columns removed)
- Missing value Impute strategy – Strategy used for imputing the missing values in the features. Values – Most Frequent, KNN
- Outlier handling – Strategy used for handling outlier values - Values - ISO(0.1): Isolated Forest, (10,90): Feature Value capping at 10^{th} and 90^{th} percentile, IQR : Value capped at IQR cut-off limits, None: No outliers removed
- Variance Threshold - The criteria used to remove features based on variance . Values - 0.25 (Features with less than 0.25 variance)
- Transform - the transformation strategy used to convert non-normal distribution to normal distribution. Values - Quantile, Power, None(No transform)
- Model Type - ML algorithm used for training. Values - Lasso, RF(Random Forest), XGBoost, MLP (Neural network)
- Model details - high level detail of hyperparameter tuning done . Values – RF(Hyp-30 iter) : RF with 30 iterations of Hyperparameter tuning, similarly for XGboost, XGboost (20000 trees low depth) : XGBoost with # trees used for training, MLP : 3 layer Neural network, MLP(7-layer) : 7 layer neural network
- CV Test RMSE - RMSE from hyperparameter tuning and cross validation
- RMSE(Train and Test) - Root mean squared score
- MAE (Train and Test) – Mean absolute score
- R2 (Train and Test) - R-squared score
- Adjusted R2 (Train and Test) – Adjusted R-squared score
- Accuracy (Train and Test) -- Absolute Predicted vs Actual < 3 / Total Predictions *100

Experiment	Missing value % Removal	Missing value Impute strategy	Outlier handling	Variance Treshold	Feature Selection	Transform	Model Type	Model Details	Train RMSE	Train MAE	Train R2	Train Adjusted R2	Train Accuracy	CV Test RMSE	Test RMSE	Test MAE	Test R2	Test Adjusted R2	Test Accuracy
1	75%	Most Frequent	ISO(0.1)	0.25	None	Quantile	Lasso	Lasso(Hyp)	39.556	30.663	0.886	0.885	6.696	39.636	40.269	31.2	0.885	0.885	6.708
2	75%	Most Frequent	None	0.25	None	Quantile	Lasso	Lasso(Hyp)	41.975	32.742	0.874	0.874	6.34	42.043	41.825	32.609	0.876	0.876	6.292
3	75%	Most Frequent	10,90	0.25	None	Quantile	Lasso	Lasso(Hyp)	42.817	33.61	0.869	0.869	5.684	42.891	42.647	33.395	0.871	0.87	5.72
5	75%	Most Frequent	10,90	0.25	None	None	RF	RF(Hyp-10 iter)	27.381	20.095	0.946	0.946	12.555	32.585	31.761	23.104	0.928	0.928	11.276
7	75%	Most Frequent	ISO(0.1)	0.25	PCA(0.95)	Quantile	Lasso	Lasso(Hyp)	38.092	29.413	0.894	0.893	7.181	38.247	38.648	29.674	0.894	0.893	7.08
8	75%	Most Frequent	ISO(0.1)	0.25	PCA(0.95)	None	RF	RF(Hyp-30 iter)	60.259	46.804	0.734	0.734	4.545	69.389	73.734	56.668	0.614	0.614	3.86
9	75%	Most Frequent	10,90	0.25	None	None	XGBoost	XGBoost(Hyp-10	26.546	19.48	0.95	0.950	12.843	29.998	29.352	21.432	0.939	0.94	11.756
10	75%	Most Frequent	10,90	0.25	Kbest(100)	None	XGBoost	XGBoost(Hyp-10	26.613	19.51	0.949	0.949	12.78	30.179	29.653	21.679	0.938	0.94	11.98
11	75%	Most Frequent	10,90	0.25	Kbest(100)	None	RF	RF(Hyp-10 iter)	28.514	21.31	0.942	0.942	10.933	33.897	33.201	24.661	0.922	0.922	9.6
12	75%	Most Frequent	None	0.25	Kbest(100)	Quantile	Lasso	Lasso(Hyp)	38.723	29.99	0.893	0.893	7.027	38.871	38.646	30.078	0.894	0.893	6.852
13	75%	Most Frequent	10,90	0.25	Kbest(100)	Quantile	Lasso	Lasso(Hyp)	40.357	31.561	0.884	0.884	5.972	40.492	40.058	31.215	0.886	0.885	6.016
15	75%	Most Frequent	ISO(0.1)	0.25	None	None	RF	RF(Hyp-10 iter)	24.982	18.268	0.954	0.954	13.576	31.174	31.793	22.917	0.928	0.928	11.492
16	75%	Most Frequent	ISO(0.1)	0.25	Kbest(100)	None	XGBoost	XGBoost(Hyp-10	25.861	18.868	0.951	0.951	13.566	30.54	30.54	22.237	0.934	0.93	12.02
17	75%	Most Frequent	ISO(0.1)	0.25	Kbest(100)	Power	MLP	MLP	28.803	19.961	0.939	0.939	11.671	30.35	30.35	22.175	0.935	0.934	10.88
18	75%	Most Frequent	ISO(0.1)	0.25	None	None	XGBoost	XGBoost(Hyp-10	25.321	18.382	0.953	0.953	13.56	28.672	28.978	21.07	0.94	0.94	11.94
19	75%	Most Frequent	ISO(0.1)	0.25	Kbest(100)	None	RF	RF(Hyp-10 iter)	26.968	19.674	0.947	0.947	13.132	31.983	32.569	23.565	0.925	0.925	11.368
20	75%	KNN	ISO(0.1)	0.25	Kbest(100)	None	XGBoost	XGBoost(Hyp-10	29.777	21.779	0.935	0.935	11.301	30.29	61.235	48.256	0.734	0.73	4.488
21	75%	KNN	ISO(0.1)	0.25	Kbest(100)	Power	MLP	MLP	28.383	20.107	0.941	0.941	10.945	30.321	30.321	22.379	0.935	0.935	10.256
22	75%	Most Frequent	None	0.25	Kbest(100)	Power	MLP	MLP	42.423	20.657	0.872	0.871	12.456	30.183	30.183	21.887	0.935	0.935	11.536
23	75%	Most Frequent	None	0.25	Kbest(100)	None	XGBoost	XGBoost(Hyp-10	28.575	20.908	0.942	0.942	12.021	30.427	29.876	21.929	0.937	0.94	11.448
24	75%	Most Frequent	None	0.25	Kbest(100)	None	RF	RF(Hyp-10 iter)	29.435	21.761	0.938	0.938	11.581	33.334	32.681	23.965	0.924	0.924	10.58
25	75%	Most Frequent	None	0.25	None	None	XGBoost	XGBoost(Hyp-10	25.925	18.924	0.952	0.952	13.057	29.162	28.465	20.772	0.943	0.94	12.42
26	75%	Most Frequent	None	0.25	None	None	RF	RF(Hyp-10 iter)	28.813	21.285	0.941	0.941	11.536	32.688	32.022	23.391	0.927	0.927	10.772
27	75%	KNN	ISO(0.1)	0.25	Kbest(100)	None	RF	RF(Hyp-10 iter)	24.299	17.956	0.957	0.957	13.053	32.262	56.187	45.702	0.776	0.776	3.772
28	75%	Most Frequent	10,90	0.25	Kbest(100)	Quantile	MLP	MLP	30.547	22.01	0.933	0.933	12.285	31.93	31.93	23.102	0.928	0.927	11.436
29	75%	Most Frequent	None	0.25	None	Power	MLP	MLP	26.073	18.668	0.951	0.951	13.52	28.888	28.888	20.786	0.941	0.940	12.8
30	75%	Most Frequent	ISO(0.1)	0.25	None	Power	MLP	MLP	26.403	19.283	0.949	0.949	11.812	29.654	29.654	21.711	0.938	0.937	11.092
31	75%	Most Frequent	10,90	0.25	None	Power	MLP	MLP	30.131	21.138	0.935	0.935	13.752	30.596	30.596	21.902	0.934	0.933	12.956
32	75%	Most Frequent	IQR	0.25	None	None	XGBoost	XGBoost(Hyp-30	22.159	16.412	0.965	0.965	14.396	32.511	30.796	22.461	0.933	0.93	12.016
33	75%	Most Frequent	None	0.25	None	None	XGBoost	XGBoost(20000 t	20.222	13.165	0.975	0.975	0	26.658	26.333	18.879	0.951	0.95	14.408
34	None	Most Frequent	None	None	None	None	XGBoost	XGBoost(20000 t	24.151	17.431	0.958	0.958	0	40.293	40.293	32.057	0.885	0.88	6.184
35	75%	Most Frequent	None	0.25	None	None	XGBoost	XGBoost(Hyp-30	26.389	19.252	0.95	0.950	12.891	29.662	28.83	20.979	0.941	0.94	12.28
36	75%	Most Frequent	None	0.25	None	Power	MLP	MLP(7-layer)	23.499	16.836	0.961	0.960	13.36	28.35	28.35	20.42	0.943	0.942	11.888

Some key observations on the results of the experiments:

XBboost and Neural networks (MLP) have comparable performance in terms of RMSE. As expected, Lasso does not have good performance due to the non-linear relationship of data.

Model Type	Average of Test RMSE	Min of Test RMSE	Max of Test RMSE
Lasso	40.69	38.65	42.65
MLP	29.99	28.35	31.93
RF	32.34	31.76	33.20
XGBoost	30.31	26.33	40.29
Grand Total	**32.52**	**26.33**	**42.65**

Statistical imputation methods lead to better performance compared to algorithm based methods like KNN

Imputation Method	Average of Test RMSE	Min of Test RMSE	Max of Test RMSE
KNN	49.25	30.32	61.24
Most Frequent	30.74	26.33	40.29
Grand Total	**32.88**	**26.33**	**61.24**

Removal of features with more than 75% missing values has a significant difference in performance vs not removing those features

Missing value removal	Average of Test RMSE	Min of Test RMSE	Max of Test RMSE
0.75	32.23	26.33	42.65
None	40.29	40.29	40.29
Grand Total	**32.52**	**26.33**	**42.65**

All outlier treatment strategies including No outlier treatment lead to comparable performance

Outlier treatment	Average of Test RMSE	Min of Test RMSE	Max of Test RMSE
10,90	31.08	29.35	33.20
IQR	30.80	30.80	30.80
ISO(0.1)	30.65	28.98	32.57
None	30.59	26.33	40.29
Grand Total	**30.74**	**26.33**	**40.29**

There is no significant difference between the performance by using 100 top features vs all features(after data cleaning)

Outlier treatment	Average of Test RMSE	Min of Test RMSE	Max of Test RMSE
Kbest(100)	31.22	29.65	33.20
None	30.44	26.33	40.29
Grand Total	**30.74**	**26.33**	**40.29**