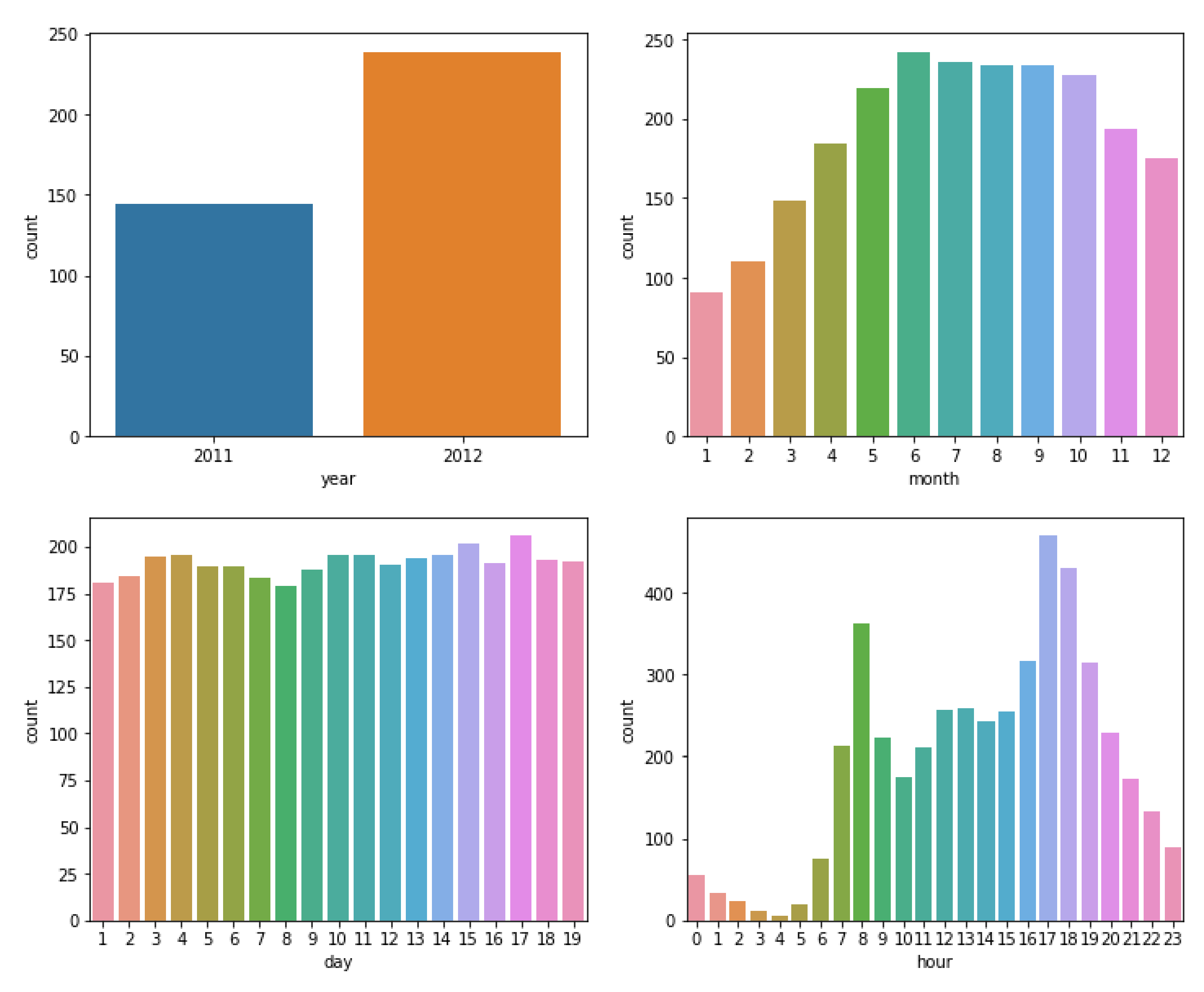
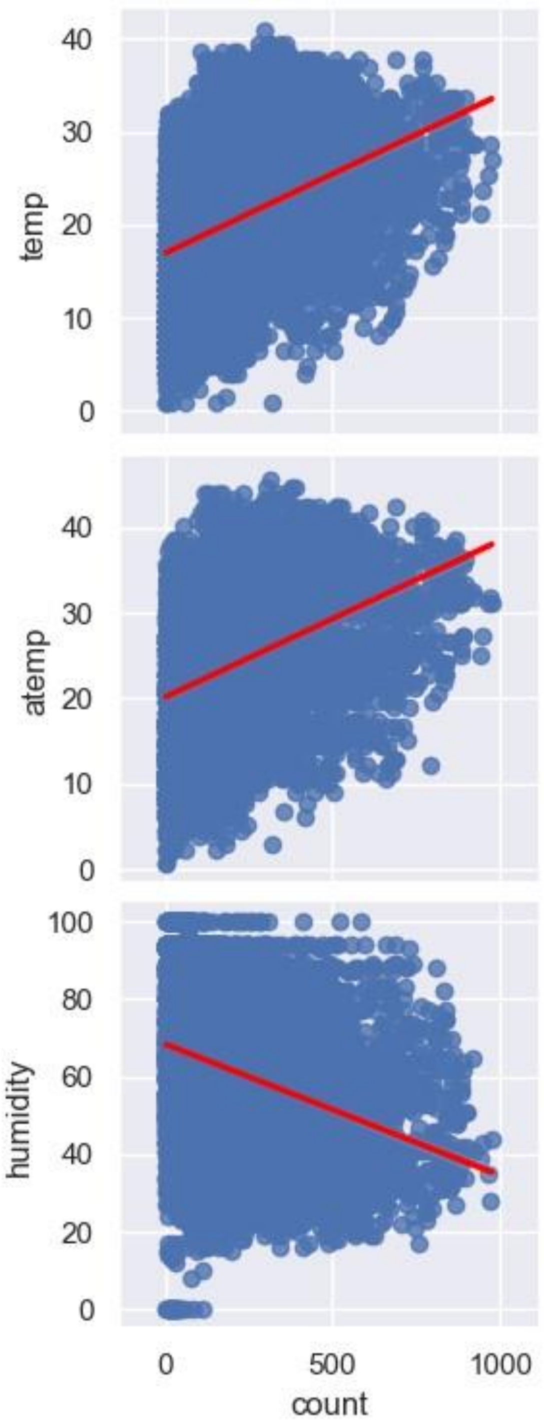
Bike Sharing Demand Prediction

Onur Dilsiz
Oktay Özel

Data Visualization





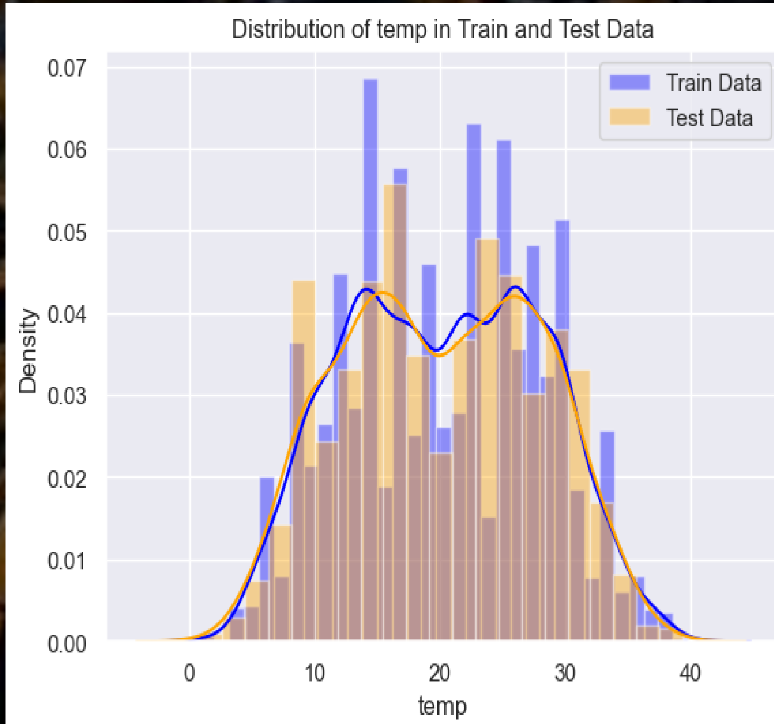
Feature Engineering

- Drop features (casual, registered)
- Check for missing values: No missing values in the data
- Create new datafields from “datetime” and drop “datetime”

2011-01-20 01:00:00 -> Year(0-1) - Month(1-12) - Day(1-31) - Hour(0-23)

- Removed Outliers from the Data

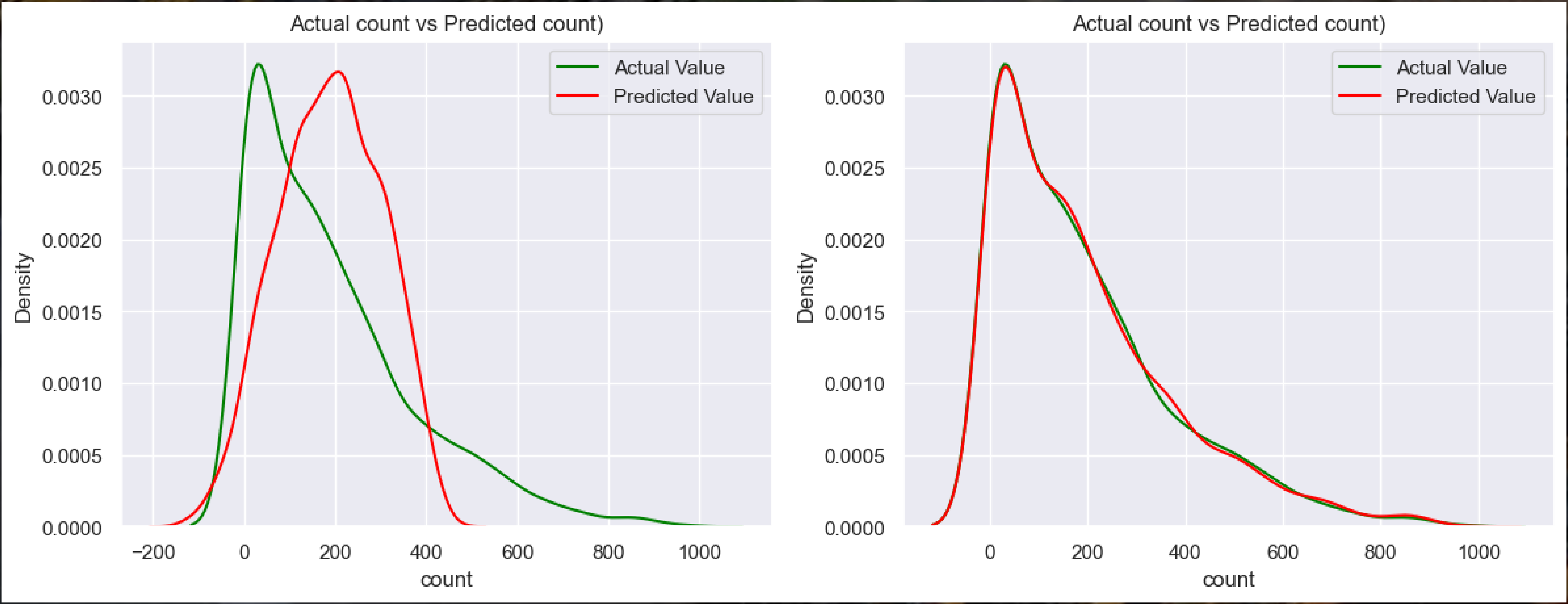
(10886, 15) -> (10739, 15) Dropped 147 samples



Assessments of Different Models on Our Problem

	R-squared	Mean Squared Error	Mean Absolute Error	Median Absolute Error
Linear Regression	0.388	19952.578	105.311	82.316
Decision Tree	0.999	3679.449	34.098	18.0
Random Forest	0.992	1513.828	24.369	14.87

Linear Regression vs Decision Tree



Bike Sharing Demand Prediction

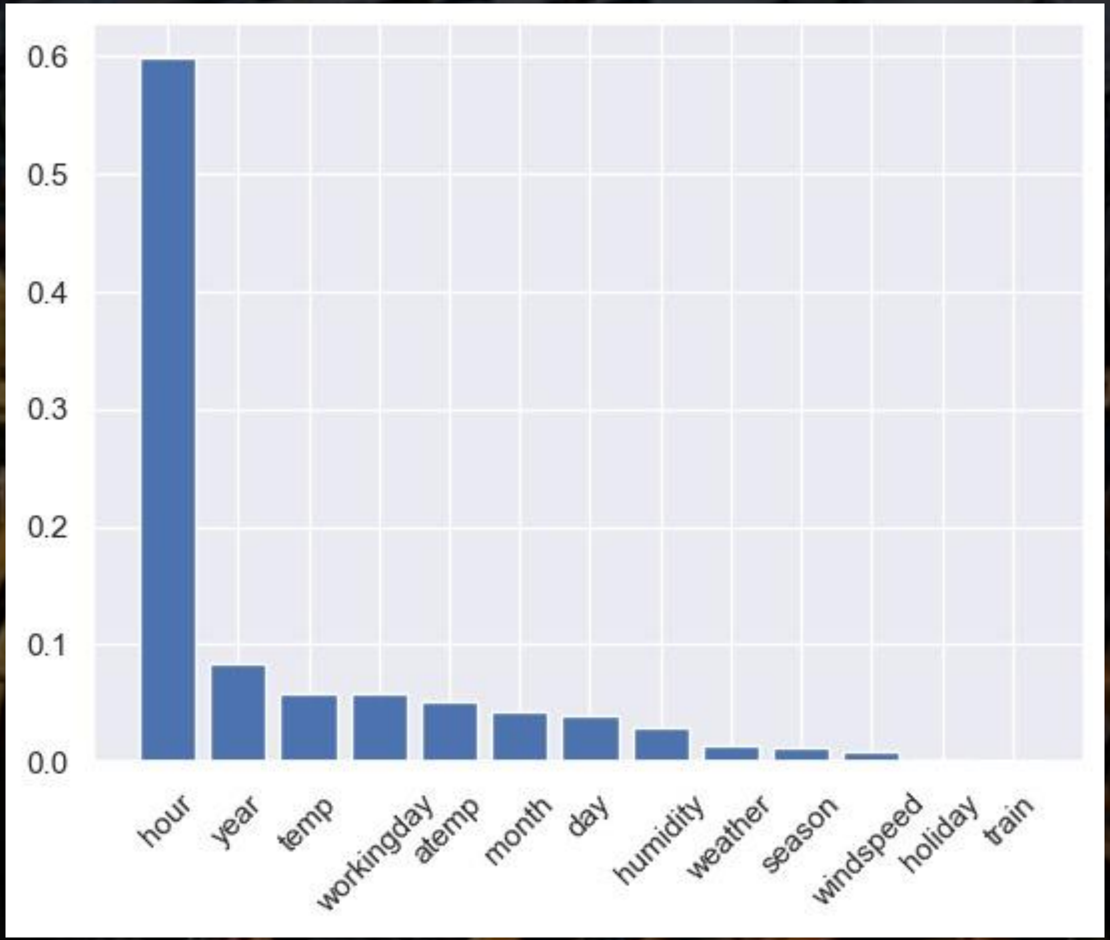
Final Presentation

Onur Dilsiz - 2019400036
Oktay Özel - 2019400108

Non – Linear Feature Importance Analysis

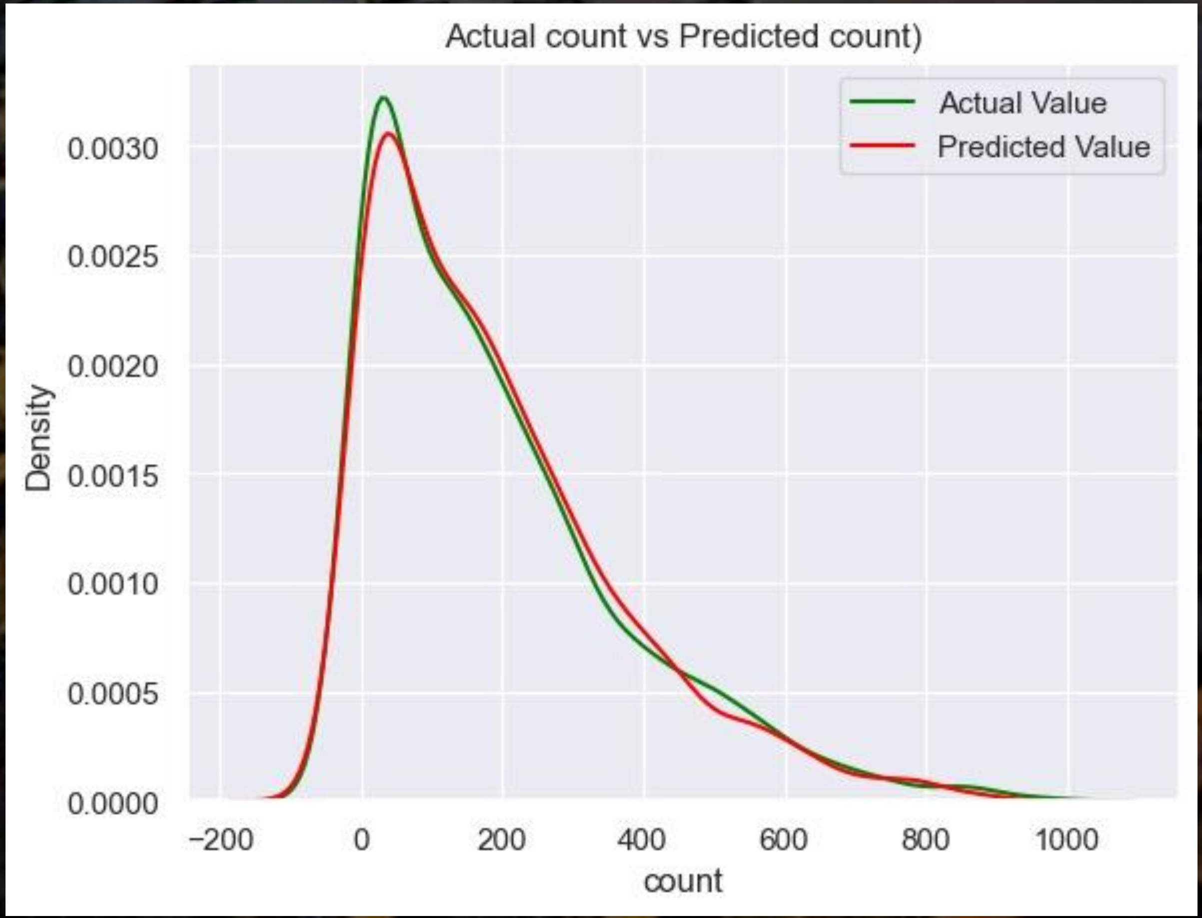
In the progress report we have used linear regression for the feature importance and correlation matrix, however it is not reasonable to use linear regression for a non-linear data.

So here there is the method that we came up with: Using a tree. We can fit the data to decision tree and calculate the feature importance depending on how much each feature reduces impurity of the tree.



XGBoost(Extreme Gradient Boosting)

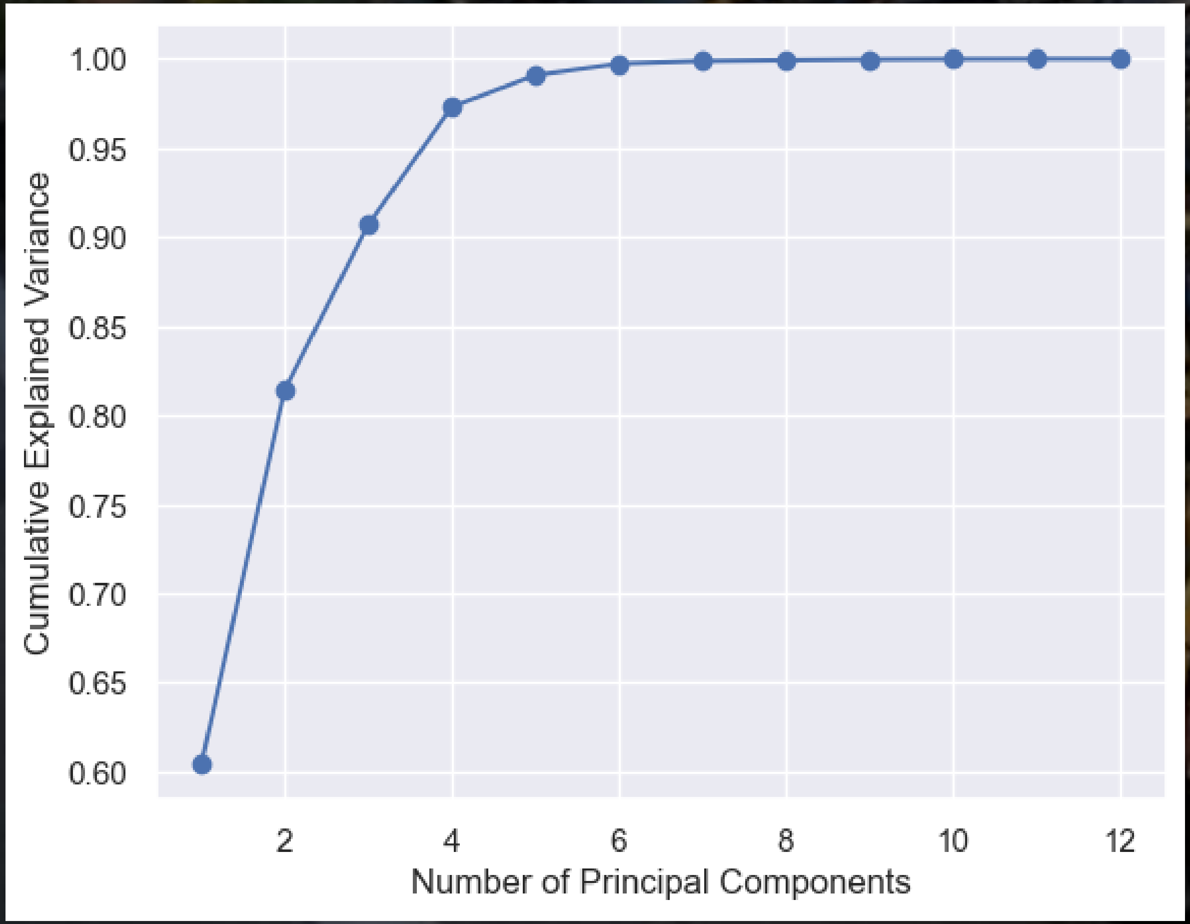
XGBoost is a gradient boosting algorithm known for its high performance in predictive modeling tasks. It employs an ensemble of shallow decision trees, combining them sequentially to correct errors and create a robust model. With features such as regularization, parallelization, and automatic handling of missing values, used both In classification and regression problems

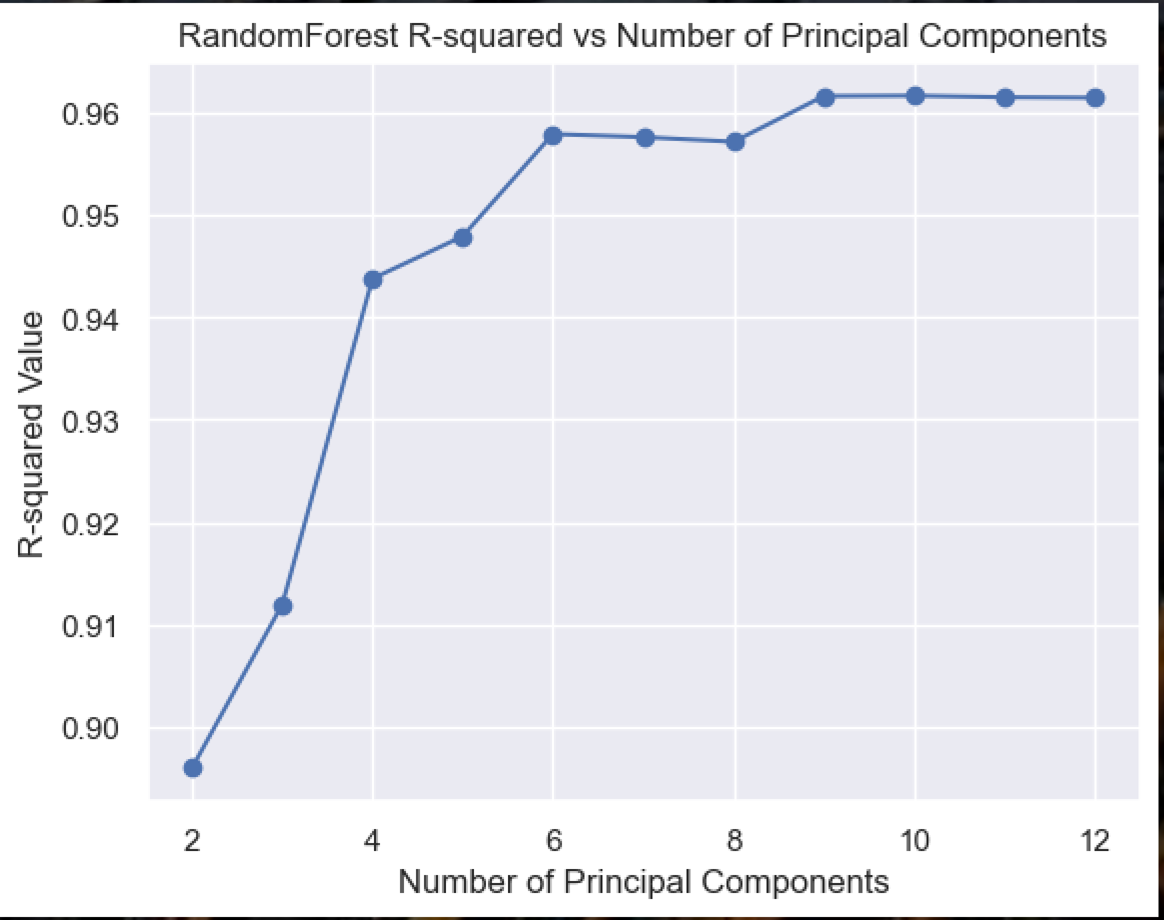


Performance Comparison of Different Models

	R-squared	Mean Squared Error	Mean Absolute Error	Median Absolute Error
Linear Regression	0.388	19952.578	105.311	82.316
ElasticNet	0.386	19999.925	104.789	82.317
Decision Tree	0.999	3679.449	34.098	18.0
Random Forest	0.992	1513.828	24.369	14.87
XGBoost	0.988	1852.12	29.39	19.74

Principle Component Analysis(PCA)





Method	RMSLE
Linear Regression	1.339
XGBoost	0.640
Decision Tree	0.54971
Random Forest	0.46938

RMSLE Scores

Root Mean Squared Logarithmic Error (RMSLE). The RMSLE is calculated as

$$\sqrt{\frac{1}{n}\sum_{i=1}^{n}(\log(p_i+1)-\log(a_i+1))^2}$$