1 Data Preprocessing

1. There were no errors or missing values in the dataset.

2. Gradient descent algorithms tend to work better when the input features have similar scales. If the features have different scales, the optimization may take longer or get stuck in local minima.

   Rescaling the data can help prevent numerical issues. When features have different scales, the matrix of input data can become poorly fit, which can cause numerical instability during training.

   It also improves the interpretability of the model. If one feature is in meters and another feature is in kilometers, the coefficients of the model will be difficult to interpret.

3. Models that perform regularization according to the norm of the parameters are sensitive to the scale of the input features or variables. If the input features have very different scales, the regularization penalty may be dominated by the features with large scales and may not effectively control the model's complexity. Essentially, rescaling ensures that all parameters have equal importance.

4. Rescaling ensures that each feature has equal importance. If the features have different scales, then some features may dominate the distance calculation, leading to biased predictions.

   Distance-based models are sensitive to the scale of the input features. Features with large scales may overshadow features with smaller scales.

   Rescaling the data can make the model interpretable. If one feature is measured in meters and another feature is measured in kilometers, the distance between two data points will be dominated by the kilometers feature.

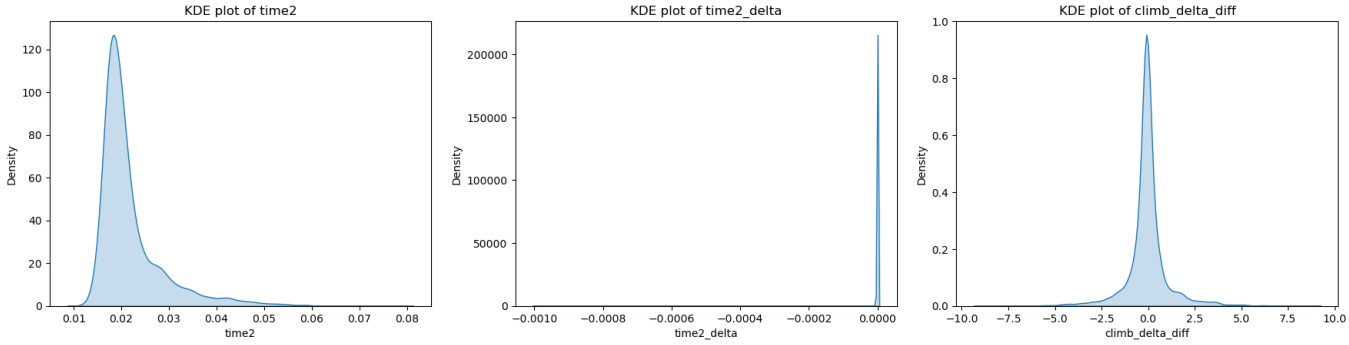
5.
```
in sample error: 0.005300060606060606
out sample error: 0.029683672727272726
in sample error: 0.005304063299663301
out sample error: 0.029392521212121214
[0.029180484848484485, 0.02884484848484848, 0.03139187878787879, 0.030262545454545455, 0.027228242424242426, 0.02850339393939394, 0.030491030303030305, 0.031527878787878785, 0.02883721212121212, 0.030569212121212122]
[0.028286424242424245, 0.028020000000000003, 0.029015030303030306, 0.028037575757575758, 0.02804169696969697, 0.027090181818181822, 0.03254569696969697, 0.03304278787878788, 0.02885260606060606, 0.030993212121212123]
in sample t-Test
TtestResult(statistic=-0.07971425097307931, pvalue=0.9382088806133239, df=9)
out sample t-Test
TtestResult(statistic=0.6087505128419045, pvalue=0.5577400530280223, df=9)
```

2 Feature Selection

1. The advantage of having many features is that there is more information at our disposal that can be used to learn a complex function. More features allow for more complex models that can capture more complex relationships between the input features and the target variable.

   The disadvantage of too many features can lead to a phenomenon known as the curse of dimensionality, where the complexity of the data and the number of possible feature combinations grow exponentially with the number of features. This can make it difficult to find the best model and lead to overfitting, where the model is too complex and fits the training data too well, but does not generalize well to new data.

2. Three features to discard are the climb_delta_diff, time2_delta, and time2. The climb_delta_diff graph has a low variance. Features with low variance (features that do not vary much across the dataset) are less informative and may not help the model to learn patterns in the data. The time2_delta als has a low variance is overlapped by other features KDE graphs. Features that are highly correlated with other features may also be candidates for discarding. This is because highly correlated features can provide redundant information to the model, and keeping both features may increase the risk of overfitting. The time2 graph KDE is also highly correlated with other features.



3. A feature that may be discarded is feature 33. It had a correlation score below a threshold of 0.01 with a value of 0.00287. A low correlation score indicates that the feature may not be strongly related to the target variable and not provide much information to the model.

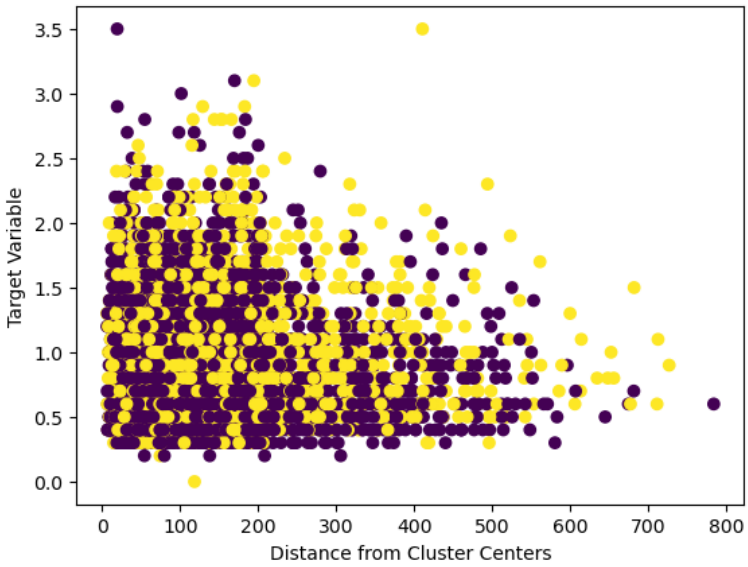
   A feature that may be kept is feature 6. It had a correlation score of 0.705. A high correlation score between a feature and the target variable indicates that the feature is strongly related to the target variable. Changes in the feature value are highly correlated with changes in the target variable. This suggests that the feature may be informative and could potentially be useful for predicting the target variable.

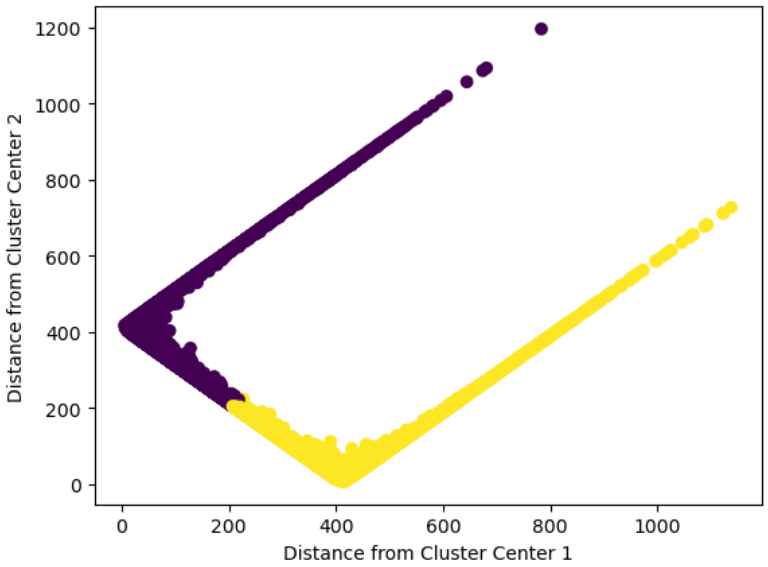
4. A pair of features would be time1 and set. These features are highly correlated, so they can be redundant in training and cause overfitting. Since time1 is similar to many of the other time features, it may be good to discard time1 and keep set since it is a feature based on a different aspect of the target.

3 Clustering

1. The clustering was not informative. The separation between the two clusters on the x-axis indicates how distinct they are in terms of their features. If the separation is large, it suggests that the two clusters have very different feature profiles, which may be informative for predicting the target variable. If the separation is small, it suggests that the two clusters are more similar in terms of their features, and may be less informative for predicting the target variable.

   Points that are farther from the cluster centers may have features that are less similar to other points in their cluster, and may therefore have more variable target values. However, if there is a strong relationship between the target variable and certain features that are not well-represented in the cluster center, some of the points farther from the center may still have similar target values.



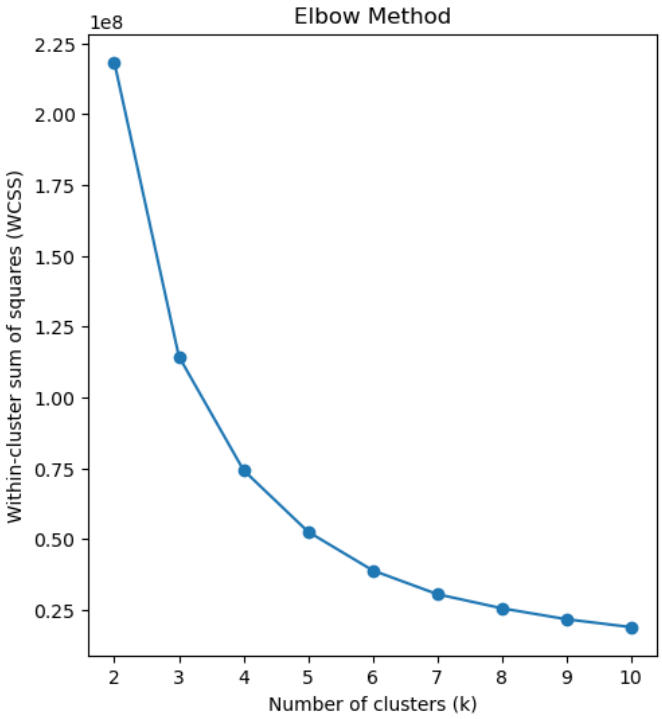


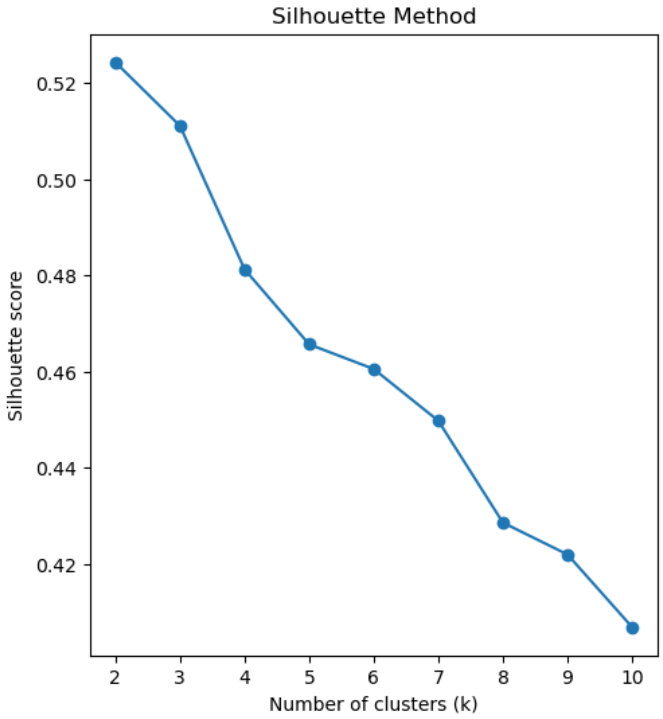
2. There different ways to determine an optimal value for k. Two ways are displayed here.

   The elbow method uses the Within-Cluster-Sum-of-Squares (WCSS). The WCSS measures the sum of squared distances between each data point and its assigned centroid, so a smaller WCSS value indicates better clustering. Even though 10 clusters had the lowest WCSS, it may be bette to use 9 clusters at times since the score is similar but may be computationally less costing.

The silhouette method provides a quantitative measure of how well each data point fits into its assigned cluster, and the optimal k value corresponds to the highest average silhouette score across all data points. The silhouette score for a single data point is calculated as the difference between the average distance to other points in its own cluster and the average distance to the nearest cluster. A lower silhouette score is not optimal.

Looking at both, the optimal k would be about 4 or 5 through the tradeoffs from each method.





4 PCA

1. The first and second principal components are plotted on the x and y axes, respectively, and represent the two dimensions of the data that explain the most variance.

   The target variable is typically represented using color or size of the points in the plot. This allows you to see how the target variable is related to the principal components. You can also look at the distribution of points along each principal component to get a sense of how much variance is explained by each component. If one principal component explains significantly more variance than the other, you may be able to simplify your analysis by focusing on just that component.

   In order to interpret a colorized 3D PCA plot, we utilize a general step-based approach in which consideration was given to the legend/axes of the plot itself, the presence of clusters (if any are obviously present), the distribution, and any possible outliers:

   1. Check the legend: The legend will tell you the meaning of each color on the plot. Each color represents a different category of data points based on a categorical variable. Understanding the categories is crucial to interpreting the plot correctly.

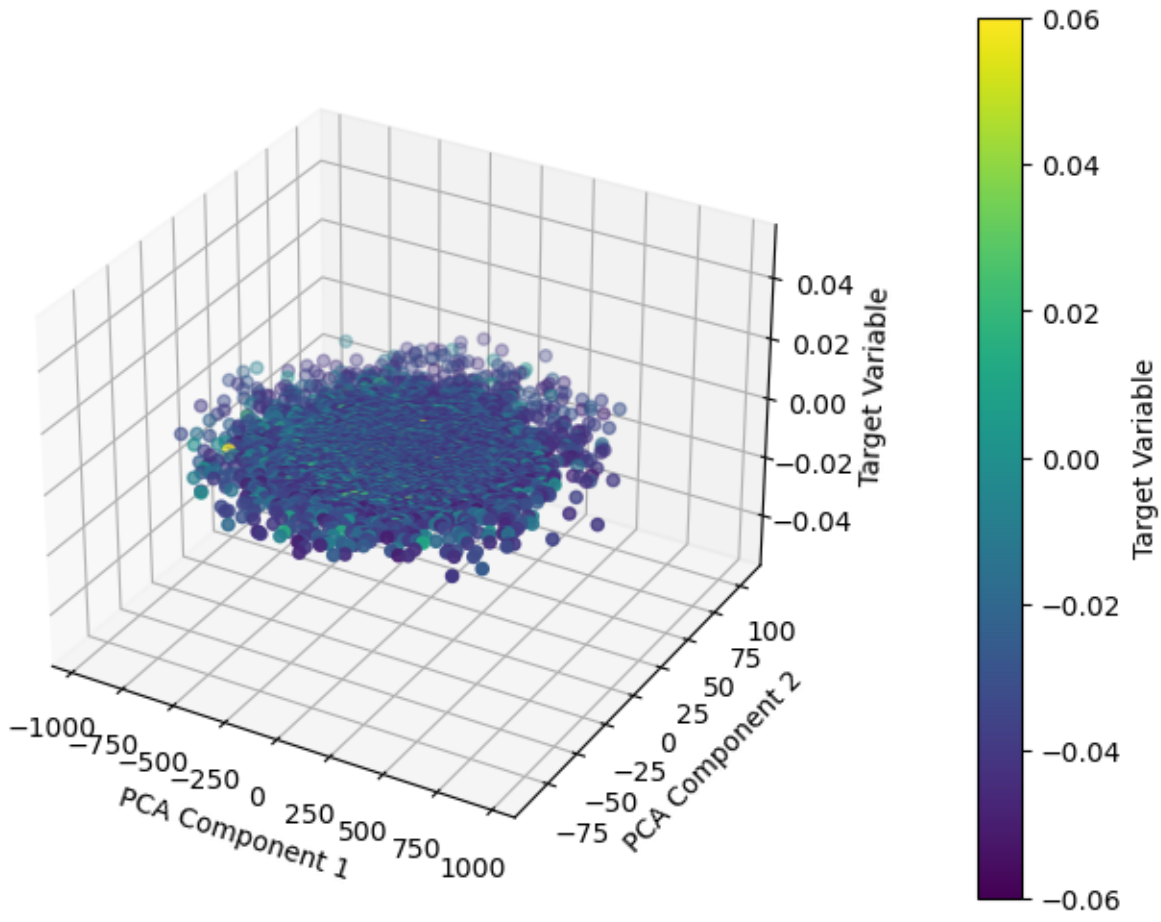
   2. Examine the axes: The axes of the plot will give you an idea of which variables are being represented in each dimension. For example, the x-axis might represent the first principal component, the y-axis might represent the second principal component, and the z-axis might represent the third principal component.

   3. Look for clusters: Clusters of similarly colored data points in the plot indicate that those data points share some underlying similarity. For example, if you see a cluster of red data points in one part of the plot, it might indicate that those data points share some common characteristic that is captured by the categorical variable represented by the red color.

   4. Analyze the distribution: The distribution of the data points in the plot can also give you insights into the relationships between the variables. For example, if the data points are evenly distributed across the plot, it might indicate that there is no strong relationship between the variables.

   5. Consider outliers: Outliers in the plot might represent data points that are unusual or different from the rest of the data. Understanding why certain data points are outliers can give you insights into the underlying relationships between the variables.

   In such consideration, no clusters immediately stood out as obvious and no distinct outliers were immediately brought to our attention. This, along with a lack of obvious variability in distribution, suggests a lack of substantial relationship between variables in the present scenario being visualized.



2.

```
---------- 10  Components----------
[[ 1.94824839e+01  3.25051621e+01 -1.39508845e+01 ...  1.31925215e+00
   9.44441025e-02  2.52298939e-02]
 [-1.96601662e+02 -2.99416414e+01 -1.04082500e+01 ... -7.90024973e-02
  -2.28625061e-02  8.11248561e-03]
 [-1.81559248e+02 -5.94195053e+00 -8.29915898e+00 ... -2.35391236e-01
   1.14768099e-01 -9.19919406e-02]
 ...
 [-3.70424928e+02  1.58053783e+01  2.65423525e+01 ... -2.01924942e-01
   6.07184895e-02 -2.81194462e-02]
 [-4.08531275e+02  1.27530039e+01 -7.46151777e+00 ... -1.57646597e-01
   3.90759558e-02 -3.11126029e-03]
 [-5.56477160e+02  1.01967177e+01  1.40331834e+01 ... -6.70465930e-02
  -4.60929764e-02  4.78717998e-02]]
---------- 20  Components----------
[[ 1.94824839e+01  3.25051621e+01 -1.39508845e+01 ...  1.32047179e-03
  -3.05280819e-04  1.84237326e-04]
 [-1.96601662e+02 -2.99416414e+01 -1.04082500e+01 ... -1.77315213e-05
  -7.26967693e-05  6.57005440e-05]
 [-1.81559248e+02 -5.94195053e+00 -8.29915898e+00 ... -2.54893912e-05
  -6.28115565e-05 -3.27698319e-05]
 ...
 [-3.70424928e+02  1.58053783e+01  2.65423525e+01 ... -4.92776508e-06
  -9.96937971e-05  1.12010863e-04]
 [-4.08531275e+02  1.27530039e+01 -7.46151777e+00 ...  6.34311450e-04
   9.54002262e-04 -9.63171333e-04]
 [-5.56477160e+02  1.01967177e+01  1.40331834e+01 ... -2.18724736e-05
  -6.10437756e-05  6.09924509e-05]]
```

```
---------- 30  Components----------
[[ 1.94824839e+01  3.25051621e+01 -1.39508845e+01 ...  6.69508131e-05
  -1.36750803e-04 -1.32062168e-05]
 [-1.96601662e+02 -2.99416414e+01 -1.04082500e+01 ... -1.72876838e-06
  -4.03390537e-06  1.56203891e-06]
 [-1.81559248e+02 -5.94195053e+00 -8.29915898e+00 ... -2.82513310e-06
  -1.15892599e-05 -2.98605484e-06]
 ...
 [-3.70424928e+02  1.58053783e+01  2.65423525e+01 ... -3.22359201e-06
  -1.37612699e-05 -1.97590431e-06]
 [-4.08531275e+02  1.27530039e+01 -7.46151777e+00 ... -1.08596542e-06
   1.89638362e-05 -7.75189812e-06]
 [-5.56477160e+02  1.01967177e+01  1.40331834e+01 ... -2.67847785e-06
  -6.27121773e-06  9.48095407e-07]]
---------- 40  Components----------
[[ 1.94824839e+01  3.25051621e+01 -1.39508845e+01 ...  8.14003553e-15
  -2.13867442e-14 -1.97698590e-14]
 [-1.96601662e+02 -2.99416414e+01 -1.04082500e+01 ... -2.32422214e-14
  -4.67519973e-14 -1.03207231e-13]
 [-1.81559248e+02 -5.94195053e+00 -8.29915898e+00 ... -2.48227531e-14
   1.61905005e-14 -5.94313733e-14]
 ...
 [-3.70424928e+02  1.58053783e+01  2.65423525e+01 ... -1.03644183e-16
  -1.29984918e-15 -1.08814619e-15]
 [-4.08531275e+02  1.27530039e+01 -7.46151777e+00 ... -4.39343933e-15
   8.12286244e-16  2.86835385e-15]
 [-5.56477160e+02  1.01967177e+01  1.40331834e+01 ...  2.92204022e-16
  -1.51088802e-15 -1.86473162e-15]]
```

3. The negative R^2 scores suggest that the linear regression models are not fitting the data well, and in fact are performing worse than a horizontal line. This can be due to many factors, such as non-linear relationships between the features and the target variable, high variance in the data, or the need for

more complex models.

One possible explanation for the decreasing performance with increasing number of PCA components is that too many components are being used, which may result in overfitting to the training data and poor generalization to new data. Alternatively, it could be due to a poor choice of hyperparameters in the PCA or linear regression models.

To further investigate the issue, you may want to try different combinations of hyperparameters and/or explore other models and performance metrics. Additionally, you could analyze the relationships between the features and the target variable more closely, for example by plotting the data and fitting non-linear models.

```
PCA with 10 components - Average R^2 score: -1.1513
PCA with 20 components - Average R^2 score: -18.7283
PCA with 30 components - Average R^2 score: -5694493951.6642
PCA with 40 components - Average R^2 score: -3552288062163501275152384.0000
Original standardized dataset - Average R^2 score: -11978076907850751832752128.0000
```