Capital Bikeshare

Washington, DC

Trip Prediction Modeling
&
Cost Optimization

Report Date: March 10th, 2024

Report By:
Zaheer Soleh

(DNSC 6314: Machine Learning)

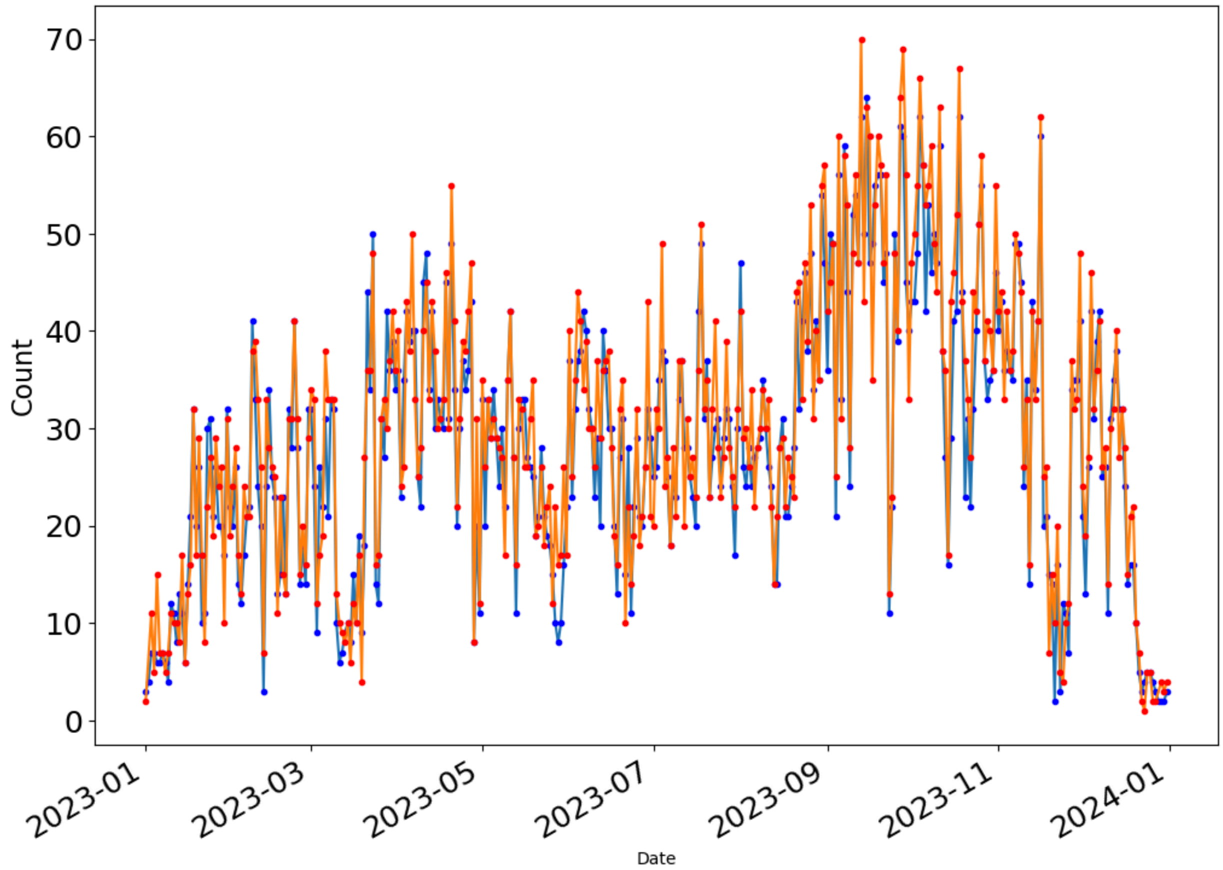
1. Exploratory Analysis:

The purpose of this report is to predict the number of pickups and dropoffs for the GWU School of Business Capital Bikeshare station for each day throughout the year based on the weather conditions and formulate a plan to optimize the bikes and docks available at that station. In doing so, we can optimize the business objectives more efficiently using multiple predictive machine learning models.

Before we start modeling, we need a preliminary analysis of the available datasets to gain insights and possible strategies to achieve the necessary objectives.

1.1 BikeShare Trips Data:

The first dataset we look at is the historical account of bikeshare data in the area to understand and visualize the trends that exist on a daily basis. The data consists of information on each trip taken, including its start and end station, the date and time, and other geographical information. Since we are interested in the distribution and trends of pickups and dropoffs by the day we aggregate and evaluate the number of pickups and dropoffs at the GWU station for each day. Below is a plot of the trends in pickups and drops throughout the year of 2023.



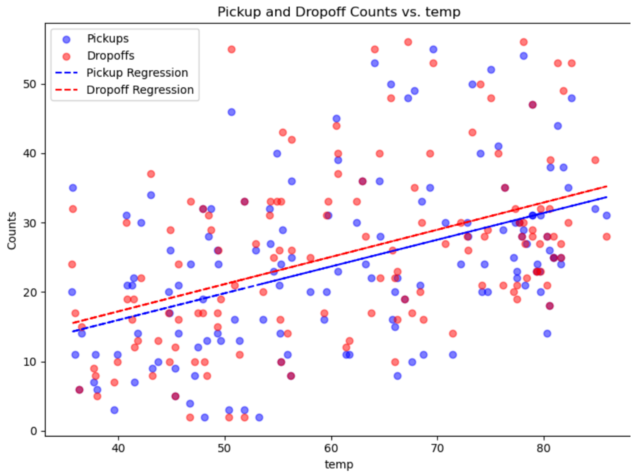
The data indicates that the busiest months are in the fall of the year with the highest pickups and dropoffs, while the winter months appear to have the lowest ridership and spring and summer stay around the same level.

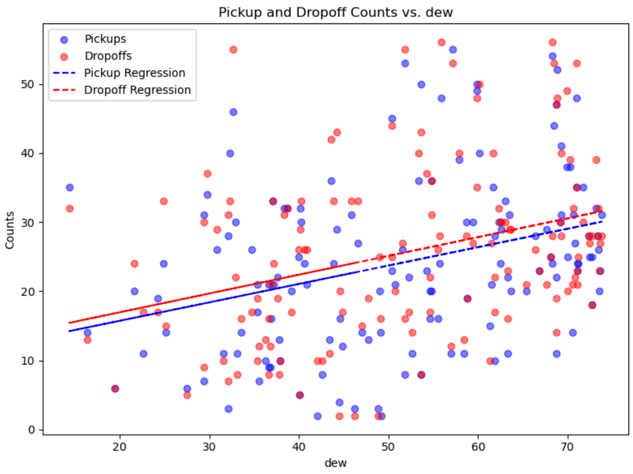
1.2 Weather Data:

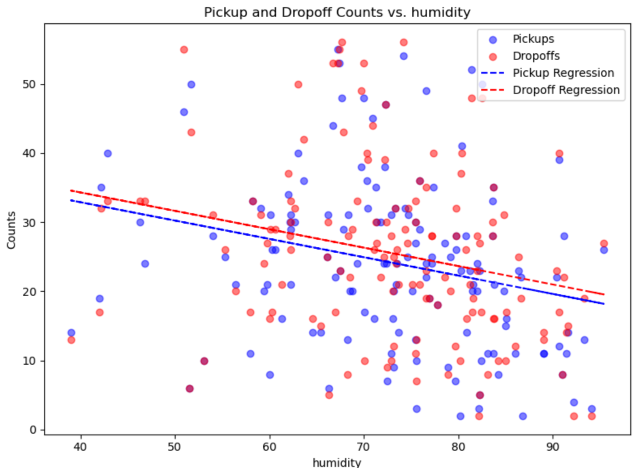
Since we need to predict the ridership numbers with the help of weather features the open-source weather data is imported from the weather data services at the visual crossing. This data provided us with up to 17 indicators to choose from while training our models for the predictions. These features include parameters like temperature, precipitation, humidity and more.

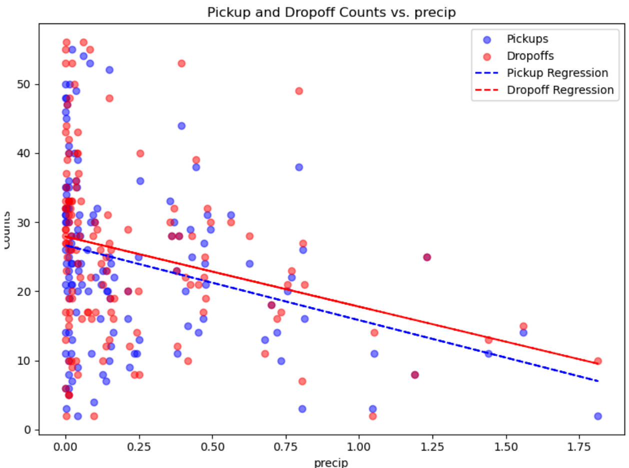
1.3 Merging Trip and Weather Data for Preliminary Analysis

Once we matched the weather data for each day with the aggregated data for the number of pickups and dropoffs for each day, we ran basic statistical tests to identify which weather features could explain the maximum amount of variance in our predictions and evaluate the correlation between them. As shown below, temperature, dew, humidity, and precipitation exhibit the highest correlation between pickups and dropoffs, which helps us get started with the initial steps of the predictive models.









2. Predictive Modeling:

We employ five different predictive models with varying levels of complexity and accuracy using the weather data for each day. Starting with the simplest and the most explainable model and moving toward the most complex model, we explored the different predictions each model offers.

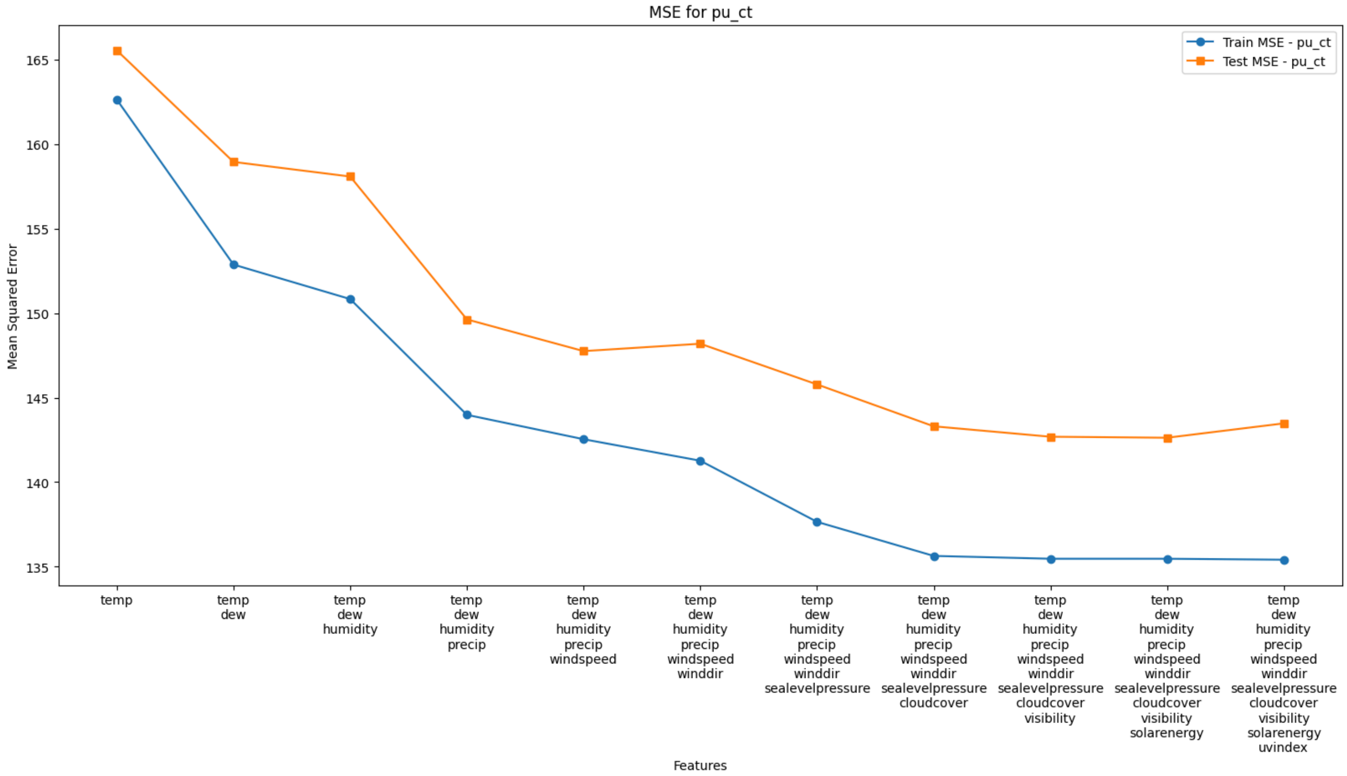
2.1 Linear Regression

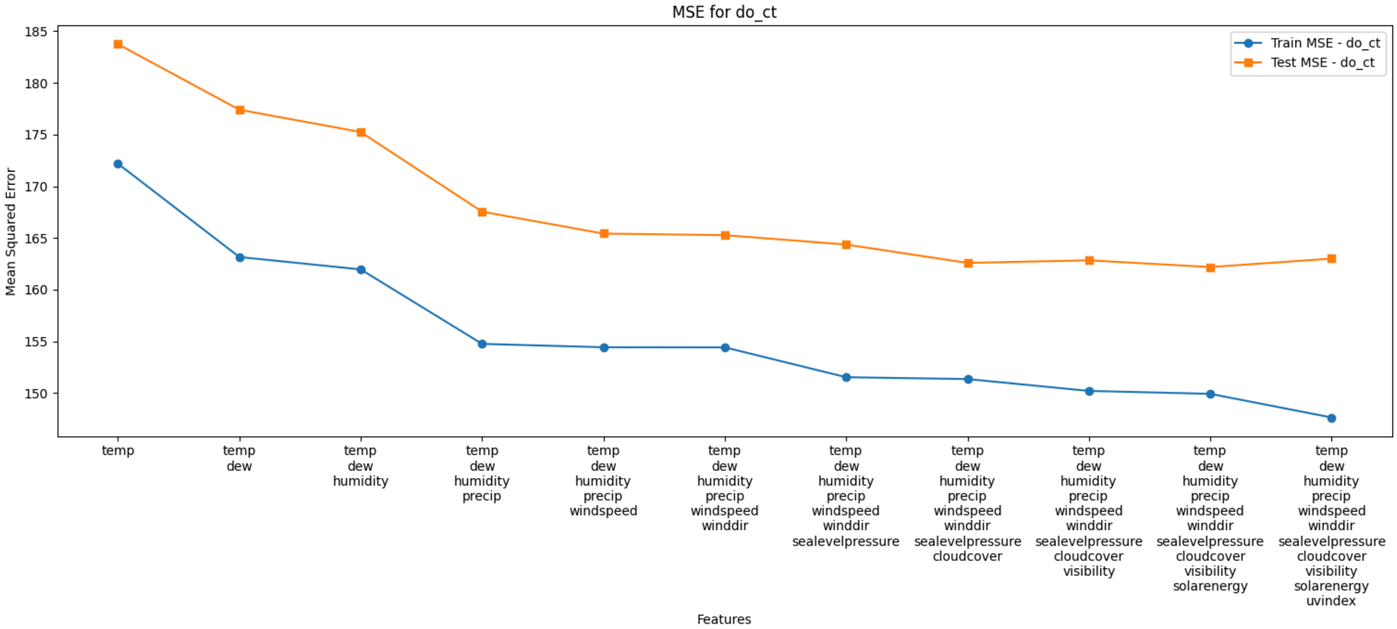
Using the findings from the exploratory data analysis, we identified the first four important features that could predict our target variable. To test these hypotheses, we started with one weather feature, i-e, temperature, applied a linear regression model, and added more and more features until the out-of-sample accuracy of our prediction could not be improved any further while also trying to balance the model interpretability and complexity.

In the linear regression models, we chose the best model based on our Mean Squared Error (MSE), the interaction between different weather parameters, as well as the R-squared value which tells us what proportion of the predicted values could be explained by the combination of our selected features.

2.1.1 Mean Square Error:

The plots below show the variation in MSE as a function of the number of features being used in each subsequent version of the linear regression.





2.1.2 Cross Validation

With the initial insight on the combination of weather features that give us the most accurate prediction of pickups and dropoffs, the next step in our modeling involved the process of cross validation to reduce the variance in our model. We computed the best number of k-folds to split our data into training and testing folds that produced the lowest root mean square error (RMSE).

The best CV score was achieved by setting the number of folds to 10 for both pickup and dropoff predictions.

```
Best CV for pick up: 10, with lowest RMSE: 12.275488654585725
Best CV for drop off: 10, with lowest RMSE: 13.58918061158711
```

2.1.3 Hyperparameter tuning with RCF and GridSearch

Using the cross-validation results from above along with recursive feature elimination we got the ranking of the features available as predictors.

```
Ranking of features for 'pu_ct':
             Feature  Rank
3             precip     1
```

```
Ranking of features for 'do_ct':
             Feature  Rank
3             precip     1
```

10 uvindex 2	11 moonphase 2
6 sealevelpressure 3	10 uvindex 3
8 visibility 4	8 visibility 4
0 temp 5	4 windspeed 5
11 moonphase 6	1 dew 6

Using another hyperparameter tuning technique of GridSearchCV we got the best features for the most accurate linear regression model:

```
Fitting 10 folds for each of 14 candidates, totalling 140 fits

Best number of features for pick up: {'n_features_to_select': 11}
Fitting 10 folds for each of 14 candidates, totalling 140 fits
Best number of features for drop off: {'n_features_to_select': 11}

Combined best features for 'pu_ct' and 'do_ct': ['temp', 'dew', 'humidity', 'precip', 'windspeed', 'sealevelpressure', 'cloudcover', 'visibility', 'solarenergy', 'uvindex', 'moonphase', 'temp_moonphase']
```

2.1.4 Final Linear Regression Model

Using the cross validation and hyperparameter tuning above we achieved the following final scores for our linear model:

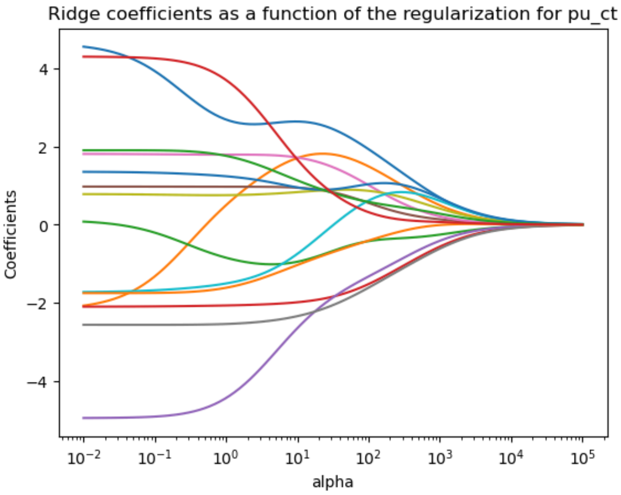
```
For 'pick up': Training MSE: 140.98241802536037, R2: 0.27379803830546146
For 'pick up': Test MSE: 147.4763265885259, R2: 0.20762834678248454
For 'drop off': Training MSE: 169.01204456106186, R2:
0.24662484688567354
For 'drop off': Test MSE: 147.86338238340528, R2: 0.20573293038110507
```

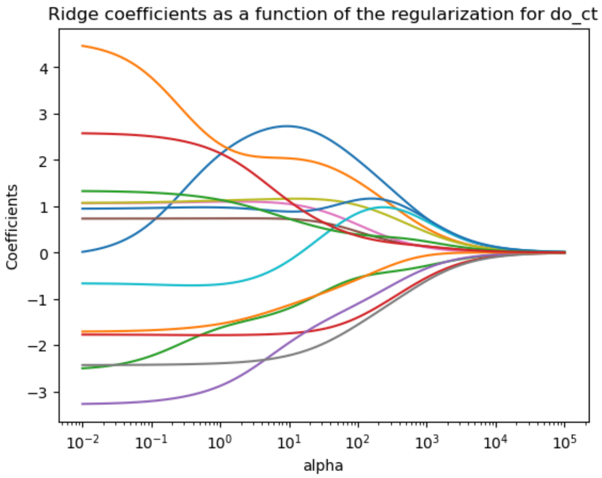
After cross-validation, the model demonstrates a noteworthy enhancement in its predictive capabilities on unseen data, as evidenced by the improvements in both 'pick up' and 'drop off' scenarios. Specifically, for the 'pick up' scenario, the Test MSE decreased from 149.38 to 146.48, and the R2 score increased from 0.1974 to 0.213, indicating a stronger model performance on the test data. Similarly, in the 'drop off' scenario, the Test MSE improved from 146.08 to 143.97, with the R2 score rising from 0.2153 to 0.2266. These improvements underscore the model's enhanced ability to generalize despite the observed slight reduction in training performance metrics such as the MSE and R2 values. The application of cross-validation has evidently contributed to mitigating overfitting, leading to a model that is more robust and reliable for predicting unseen data. This validates the decision to select the model post-cross-validation as a more accurate and generalizable approach.

2.2 Ridge Regression

After testing linear regression, we moved to a more complex form of regression known as ridge regression that trains the model while minimizing the higher impacts of some prediction features by scaling our predictors. This step helps us reduce the chances of overfitting our model due to the existence of multicollinearity in our data as well.

Following plots show the best coefficients recommended by the model for the regularization of our model:





```
The coefficients for pick up are:
temp                   0.696133
dew                    0.497156
humidity              -0.255161
precip                -0.562901
windspeed             -0.413192
winddir                0.137205
sealevelpressure       0.212161
cloudcover            -0.614774
visibility             0.412513
solarenergy            0.610232
uvindex                0.677839
moonphase             -0.009379
temp_moonphase         0.280822
temp_windspeed         0.068147
dtype: float64
```

```
The coefficients for drop off are:
temp                   0.737813
dew                    0.524131
humidity              -0.284938
precip                -0.540332
windspeed             -0.339906
winddir                0.133240
sealevelpressure       0.113540
...
moonphase             -0.064938
temp_moonphase         0.238159
temp_windspeed         0.133708
dtype: float64
```

2.2.1 Cross Validation

Similar to Linear Regression we evaluate the best k-fold and alpha coefficients for cross validation of the Ridge Regression model.

```
Best k for pick up: 9
Best alpha for pick up: 205.65123083486515

Best k for drop off: 10
Best alpha for drop off: 3274.549162877725
```

2.2.2 Final Model for Ridge Regression

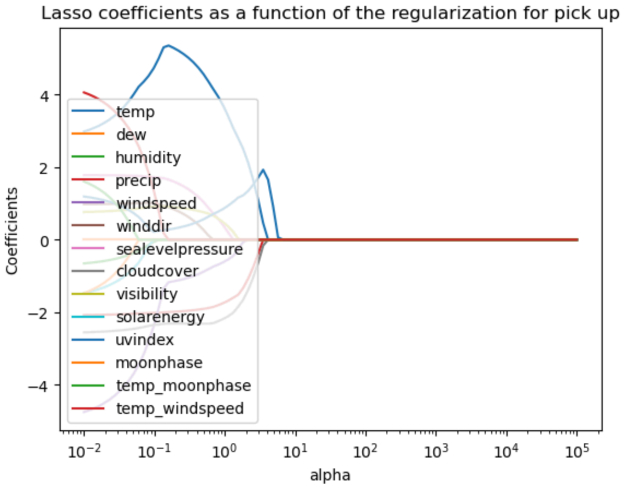
Based on the results above we retrained the model and evaluate new coefficients for the ridge regression model:

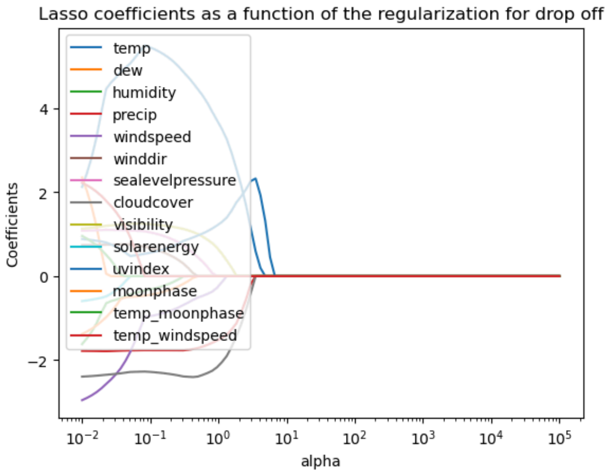
```
The coefficients for pick up are:
temp                    1.352887
dew                     1.026515
humidity               -0.346985
precip                 -1.086741
windspeed              -0.887301
winddir                 0.301693
sealevelpressure        0.565328
cloudcover             -1.146801
visibility              0.710149
solarenergy             0.834780
uvindex                 1.026307
moonphase              -0.171335
temp_moonphase          0.446383
temp_windspeed          0.130724
dtype: float64
```

```
The coefficients for drop off are:
temp                    0.303629
dew                     0.207271
humidity               -0.137214
precip                 -0.224615
windspeed              -0.125101
winddir                 0.060875
sealevelpressure        0.033842
...
moonphase              -0.003738
temp_moonphase          0.116260
temp_windspeed          0.064143
dtype: float64
```

2.3 LASSO

Least Absolute Shrinkage and Selection Operator (LASSO) is used to add another regularization parameter to the linear regression model in order to make the parameters more regularized and perform an automatic feature selection methodology that reduces non reduces less relevant features' coefficients to zero. We run this model to get new set of parameters which can be visualized in the following plots:





Lasso regression coefficients for different weather-related predictors (like temperature, dew point, humidity, etc.) as a function of the regularization parameter alpha. As alpha increases, Lasso regularization increasingly penalizes non-zero coefficients, driving them towards zero and thus performing feature selection. The analysis helps to identify which predictors are most relevant to the model by observing which coefficients shrink toward zero as regularization intensifies.

2.3.1 Cross Validation

Same as before we performed cross validation for LASSO to reduce the overall variance in the model predictions and measured the following coefficients for its regularization and cross validation parameters:

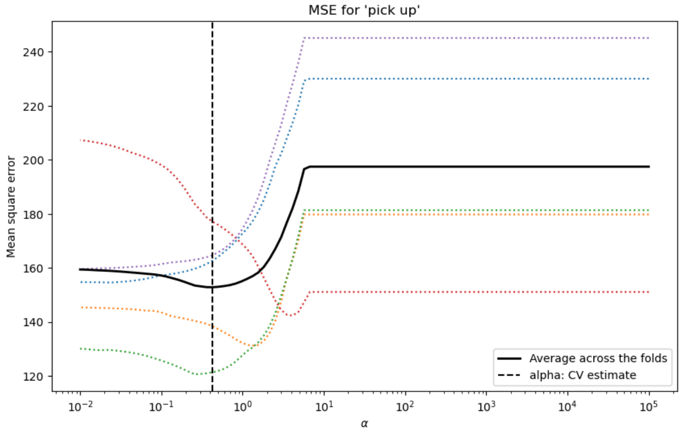
```
The best alpha from LassoCV for pick up: 0.4229242874389499

The best alpha from LassoCV for drop off: 0.8111308307896868
```

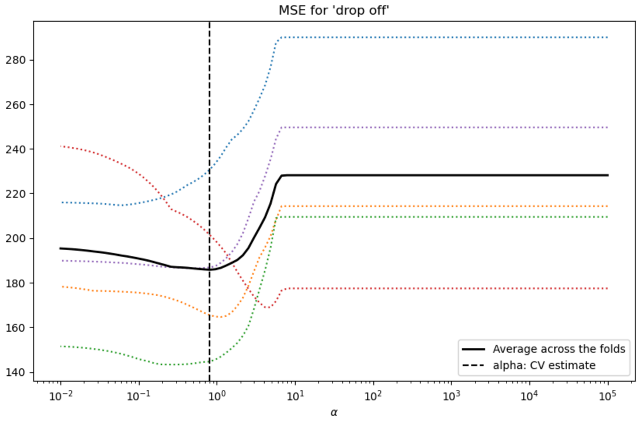
2.3.2 Final Model for LASSO

With the regularization parameters above the final LASSO model produces the following results:

```
The coefficients for pick up are:
temp                   4.742825
dew                    0.000000
humidity              -0.000000
precip                -1.926417
windspeed             -0.948257
winddir                0.412305
sealevelpressure       1.177175
cloudcover            -2.313938
visibility             0.793521
solarenergy            0.000000
uvindex                0.500555
moonphase             -0.000000
temp_moonphase         0.000000
temp_windspeed         0.000000
dtype: float64
```

```
The coefficients for drop off are:
temp                   3.919300
dew                    0.000000
humidity              -0.000000
precip                -1.587743
windspeed             -0.367965
winddir                0.000000
sealevelpressure       0.074053
...
moonphase             -0.000000
temp_moonphase        -0.000000
temp_windspeed        -0.000000
dtype: float64
```





2.4 Elastic Net

Elastic net is a model that combines both the ridge regression and lasso models into one and requires the regularization with respect to both their parameters/coefficients.

2.4.1 Cross Validation

We used our data to first train using this model and got the following results:

```
The best alpha from ElasticNetCV for pick up: 0.23253197138037357
The best alpha from ElasticNetCV for drop off: 0.46500367310245894
```

2.4.2 Final Model

Using this model produced the following results for coefficients:

```
The coefficients for pick up are:
temp                 2.807177
dew                  1.649141
humidity            -0.494947
precip              -1.843250
windspeed           -1.506734
winddir              0.719682
sealevelpressure     1.360508
cloudcover          -2.093979
visibility           0.863851
solarenergy          0.000000
uvindex              0.955166
moonphase           -0.163926
temp_moonphase       0.247568
temp_windspeed       0.409272
dtype: float64
```

```
The coefficients for drop off are:
temp                 2.472432
dew                  1.492053
humidity            -0.421243
precip              -1.501686
windspeed           -0.815114
winddir              0.312771
sealevelpressure     0.492665
...
moonphase           -0.216414
temp_moonphase       0.000000
temp_windspeed       0.000000
dtype: float64
```

2.5 KNN Regressor

KNN regressor is another prediction model that is a non-parametric method that predicts a numerical target based on a similarity measure. We trained our model to achieve the following results for MSE values:

```
MSE for 'pick up' on training data:
133.8467579908676
MSE for 'pick up' on testing data:
191.43945205479451
```

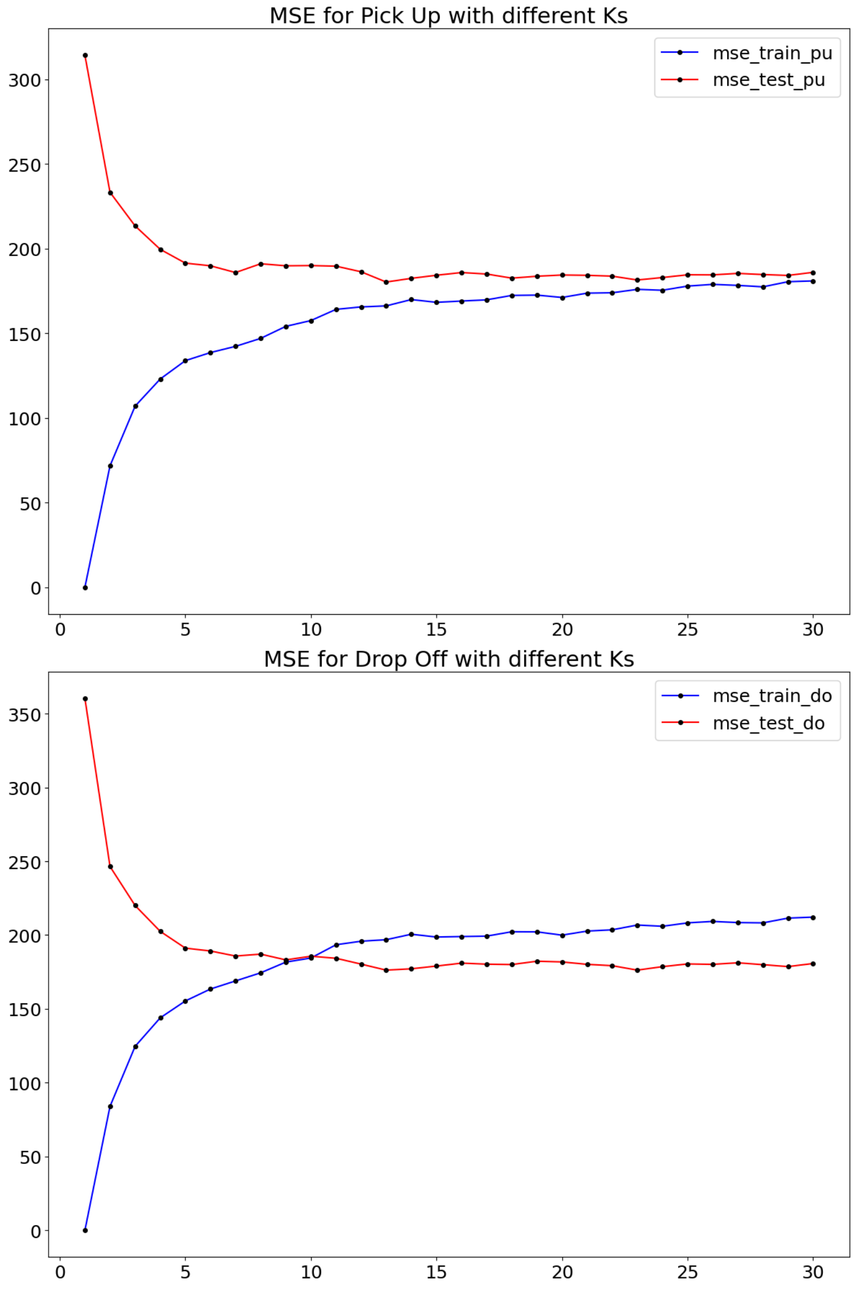
```
MSE for 'drop off' on training
data: 155.36858447488586
MSE for 'drop off' on testing data:
191.16027397260274
```

2.5.1 Cross Validation - K-fold Validation

The following optimal cross validation parameters were found:

```
Optimal K for 'pick up': 13
Optimal K for 'drop off': 13
```

We can also visualize how the k values change the testing and training values below:



2.5.2 Final Model for KNN Regressor

Using the results from cross validation the final model produces the following coefficients:

```
MSE for 'pick up' on training data: 166.18443165545378
MSE for 'pick up' on testing data: 180.25301937261895

MSE for 'drop off' on training data: 196.86366215449462
MSE for 'drop off' on testing data: 176.24856123855068
```

3. Performance Evaluation:

In order to choose which model performs best, first we need to define each kind of performance and how they impact each other and the final recommendations on choosing the best model.

3.1 Prediction Performance:

The prediction performance refers to the accuracy of the model with regards to the actual value of pickups and dropoffs for each day. As we saw for each model earlier, there is a slightly different metric for measuring the accuracy or performance given its parameters and constraints. Below we outline the performance metrics of each of the final models that explain their error rates and the variance in the predicted number that can be directly explained by the predictors.

Table I: Comparison of Model Performance in Prediction

	Linear Regression	Ridge Regression	LASSO	Elastic Net	KNN Regressor
Test MSE Pickup	147.5	151.41	152.88	151.95	180.25
Test MSE Drop Off	147.9	151.29	149.44	149.74	176.25
Test R2 Pickup	0.2076	0.1940	0.2126	0.2093	0.0315
Test R2 Drop Off	0.2057	0.1995	0.2207	0.2178	0.0535

3.2 Decision Performance:

For using the predictions from each of the models we first define the objective function of using that predicted pickups and dropoffs. If we define the decision in terms of lost opportunity or more specifically the cost of no pickup available and no dropoff available, we can define the objective as follows:

Cost = a*max(0,actual_pickups - predicted_pickups) +
b*max(0,actual_dropoffs - predicted_dropoffs)

Where a = penalty for each no bike penalty
b = penalty for each no dock penalty

Based on the predicted pickups and dropoffs from each model and using the model above, keeping in mind that the GWu station has a maximum of 17 docks installed, we can measure the loss from using each of the models for each day and average it over the whole time period.

The table below represents the comparison between predicted and actual number for the first day in the data.

Table II: Comparison of Decision Performance using Prediction (For First Day)

	Linear Regression	Ridge Regression	LASSO	Elastic Net	KNN Regressor
Predicted Pickups	26	28	27	27	30
Actual Pickups	25	25	25	25	25
Predicted Drop Offs	27	30	28	28	29
Actual Drop Offs	26	26	26	26	26

If we use all of the models and average the loss over the year, the following numbers reflect predicted loss from each model:

Table III: Comparison of Decision Performance using Prediction (Average)

	Linear Regression	Ridge Regression	LASSO	Elastic Net	KNN Regressor
Average Loss	95.26	96.46	95.06	95.17	96.12

4. Conclusion:

4.1 Key Findings:

Looking at table I we can see that all models exhibit similar levels of performance in terms of test MSE and R2 scores with slight variations.

- Linear Regression shows the lowest MSE for the Pickup data, suggesting it predicts with less error for Pickup. Its R2 scores are moderately low, indicating it explains some but not a lot of variance in both Pickup and Drop Off predictions.

- Ridge Regression has slightly higher MSEs than Linear Regression, indicating slightly poorer predictions in terms of error for both Pickup and Drop Off. The R2 scores are lower than Linear Regression, suggesting it explains less variance in the outcomes.

- LASSO has higher MSEs for Pickup and slightly better for Drop Off compared to Linear Regression. It has the highest R2 scores among the models, meaning it accounts for a slightly higher proportion of variance in both Pickup and Drop Off predictions.

- Elastic Net's MSEs are quite close to those of Ridge Regression and LASSO, and its R2 scores are similar to LASSO's, suggesting comparable explanatory power for the variance in the dependent variables.

- KNN Regressor has the highest MSEs, indicating the poorest prediction accuracy in terms of error. It also has the lowest R2 scores, suggesting it explains the least variance for both Pickup and Drop Off.

Overall Linear Regression has the best prediction error rate for the Pickup task, while LASSO and Elastic Net have better R2 scores. LASSO seems to be a good balance between MSE and R2, providing the best R2 scores for both tasks. This indicates it might be the best at generalizing despite not always having the lowest prediction error. Ridge Regression and Elastic Net are comparable and are outperformed by Linear Regression in terms of MSE for Pickup but not by much. KNN Regressor performs the worst across both tasks and metrics, indicating it may not be suitable for this particular problem or requires parameter tuning. The R2 values across all models are low, none exceeding 0.23, which could indicate that the models are not capturing much of the variance in the dataset or that the dataset itself has a lot of inherent variability that is difficult to model with these techniques.

4.2 Recommendations:

Looking at Table III we can see that LASSO is the model with the lowest average loss, suggesting it has the best overall prediction performance over the period considered. This aligns with its performance in the single-day predictions in Table II, where it closely matched the actual numbers.

Based on the data provided, LASSO would be recommended as it not only performed well on the first day but also showed the smallest average loss over an extended period, indicating consistent performance. However, it is essential to consider other factors such as the models' performance on unseen data (generalizability), the computational cost of training and using the model, and the importance of understanding the model's decisions (interpretability) before making a final decision. If the performance differences are not significant, simpler models like Linear Regression might be preferred due to their interpretability and ease of use.

4.3 Limitations:

The usage of these models comes with some challenges and limitations as they only account for about 20% of the variance in our predictions. The rest of the predictions can be from a source of biased parameters and would not accurately reflect the situation as evident in the performance and decisions evaluation sections.

We can also see that some months have more data or trips than others so the predictions could be skewed in favor of those months and not accurately represent the whole year on a day-by-day basis.

Nevertheless, it provides insights into the bike trip trends and gives some indications and suggestions for useful analysis and decision-making.

6. Appendices:

Appendices are attached as a separate document and the section under heading Part II should be referred for references above.