Final Capstone

A Data Science project meant for improving transportation services.

David Booker-Earley

Presentation Outline

- Intro
  - Problem Statement and Goal
  - Data Overview
- Exploration
  - Target & Variables of Interest
- Machine Learning Models
  - Predictions
  - Clusters
- Next Steps
- Appendix

The Problem

- Essential Workers
  - Commute via bus amid pandemic
- Social Distancing (SD)
  - Limits passengers per bus
- Bus Times
  - Need to be accurate
  - Need to account for SD

The Problem

- Essential Workers
  - Commute via bus amid pandemic
- Social Distancing (SD)
  - Limits passengers per bus
- Bus Times
  - Need to be accurate
  - Need to account for SD

"What could possibly go wrong?"

- Delays!
  - Occurred before pandemic & SD
  - Will likely occur during
- May cause more problems
  - Bus may be at full capacity
  - Workers stuck waiting
  - Productivity hindered (again)
- Need to account for delays!

Still waiting for the bus, I’ll be there ASAP

.. These “en route” struggles are real

Manager

Okay, stay safe, please hurry, we’re really understaffed

K, I’ll wait faster lol

Approaching the Problem

1. *Which bus-routes or rail-lines are typically delayed?*
   .. How can these delays be predicted?

2. *Which days of the week typically have the most frequent and longest duration of bus or rail delays?*

3. *How can data from previous years be used now to project the number of in-service-vehicles needed during (and after) the pandemic?*

Goal

- ➢ Predict whether a bus will be delayed based on pre-pandemic conditions.
- ➢ Discover days with the longest delays for the most delayed bus routes.

How?

- ➢ Scientific Wizardry!
  - ○ Thank you, machine learning algorithms!

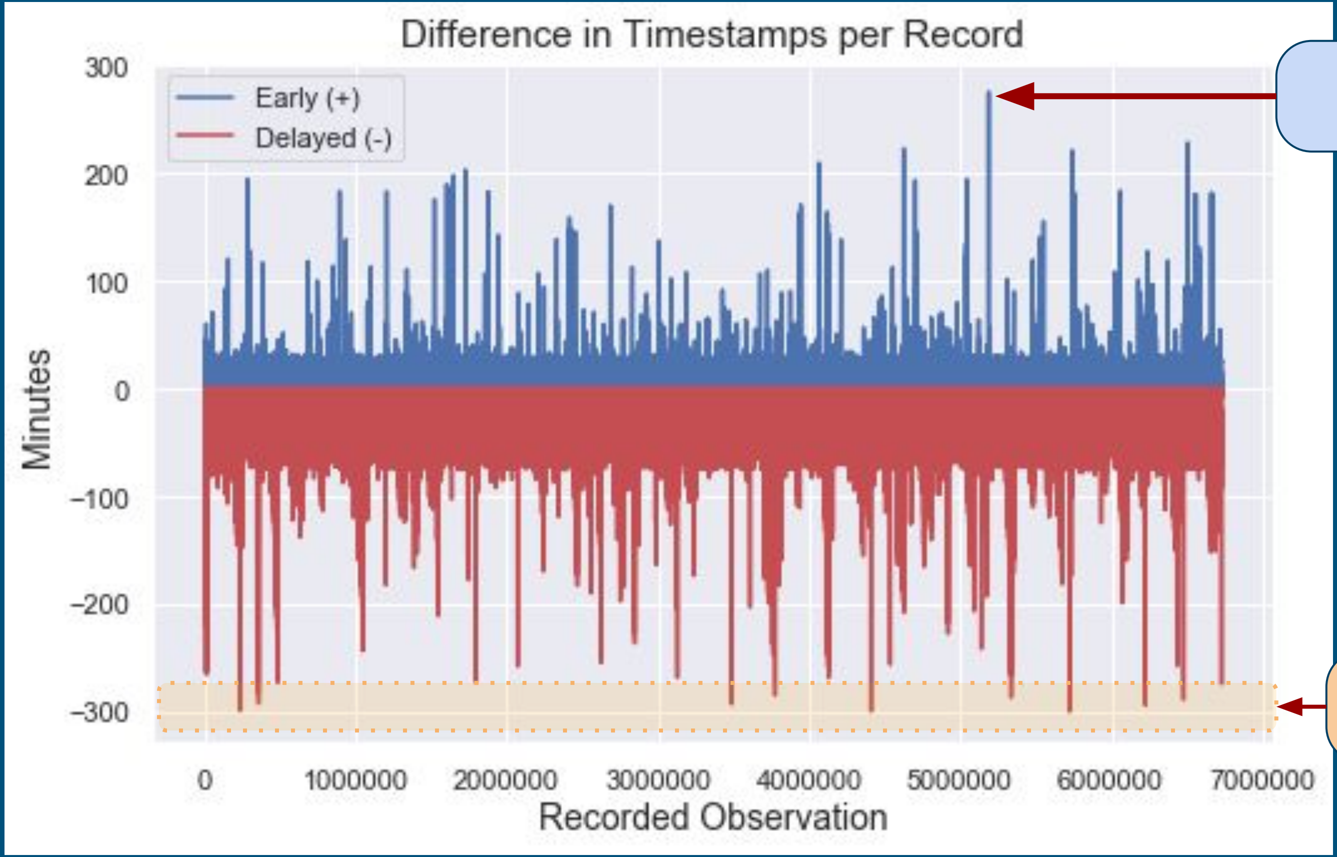
Data Overview

2017 New York City MTA Bus Records

- ➢ Located on Kaggle
- ➢ Provides bus-performance data for various months
  - ○ Location, Route, Arrival Time, Date, etc.
- ➢ Selected Data → June, 2017
  - ○ Over six million observations
  - ○ 17 columns

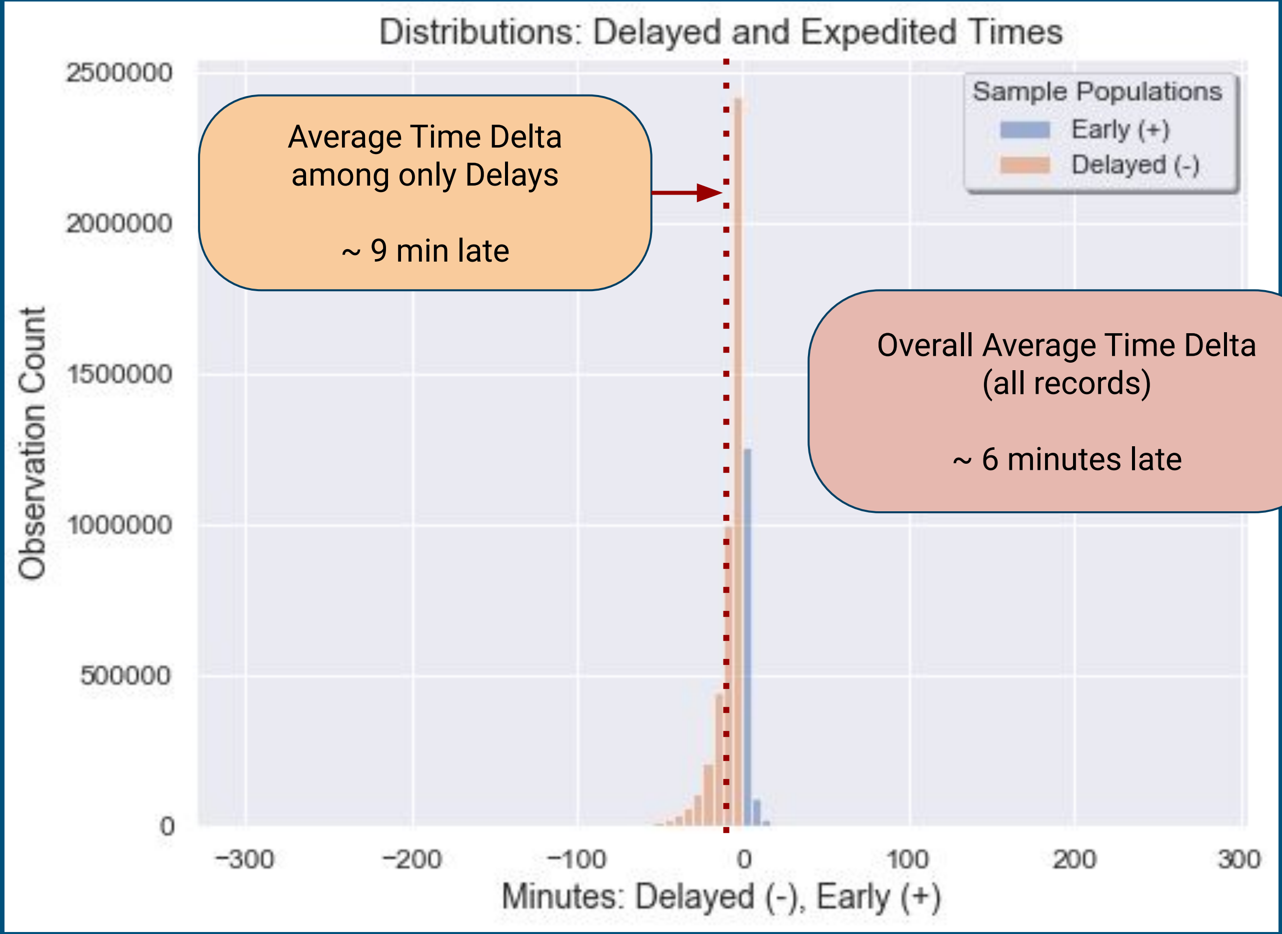
Exploration!

.. How early or late were the buses?



Earliest bus:
~ 275 min

Major delays:
~ 300 min



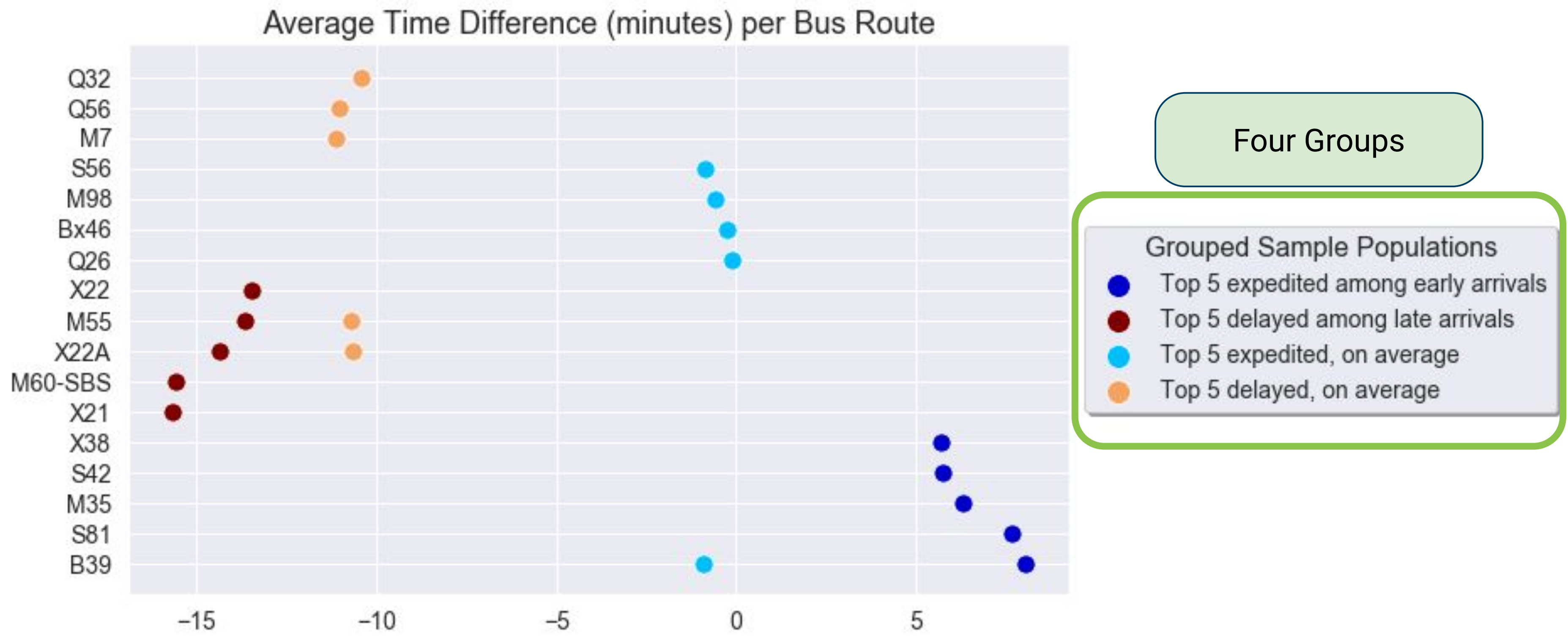
Average Time Delta among only Delays

~ 9 min late

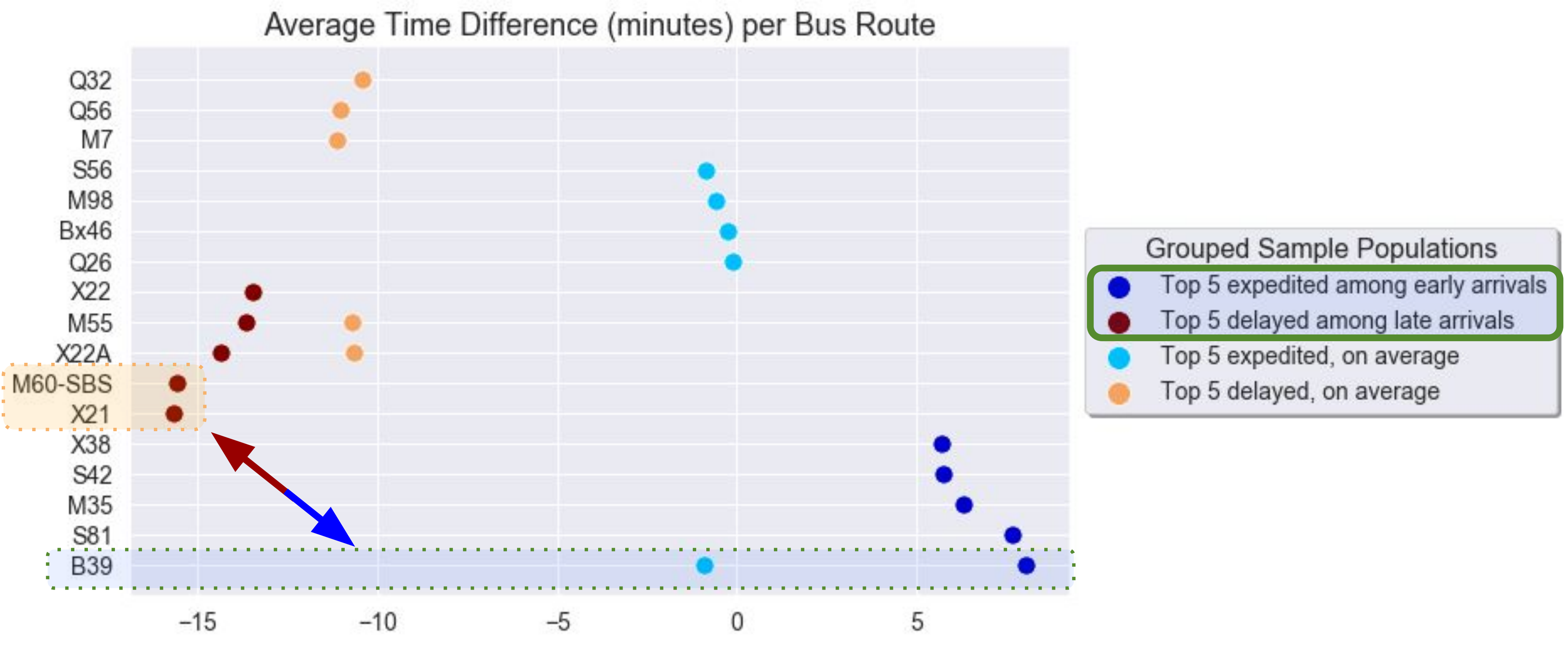
Overall Average Time Delta (all records)

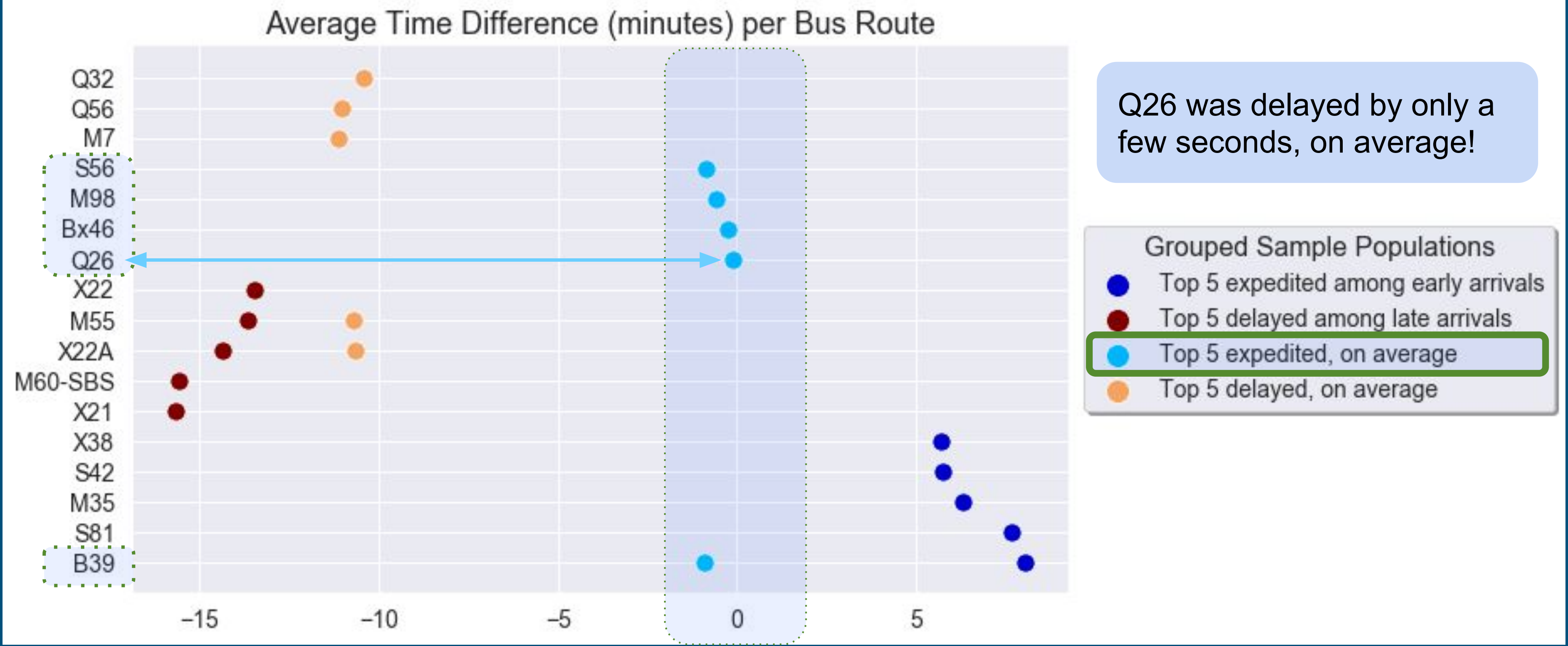
~ 6 minutes late

On average, which buses were the earliest and latest?

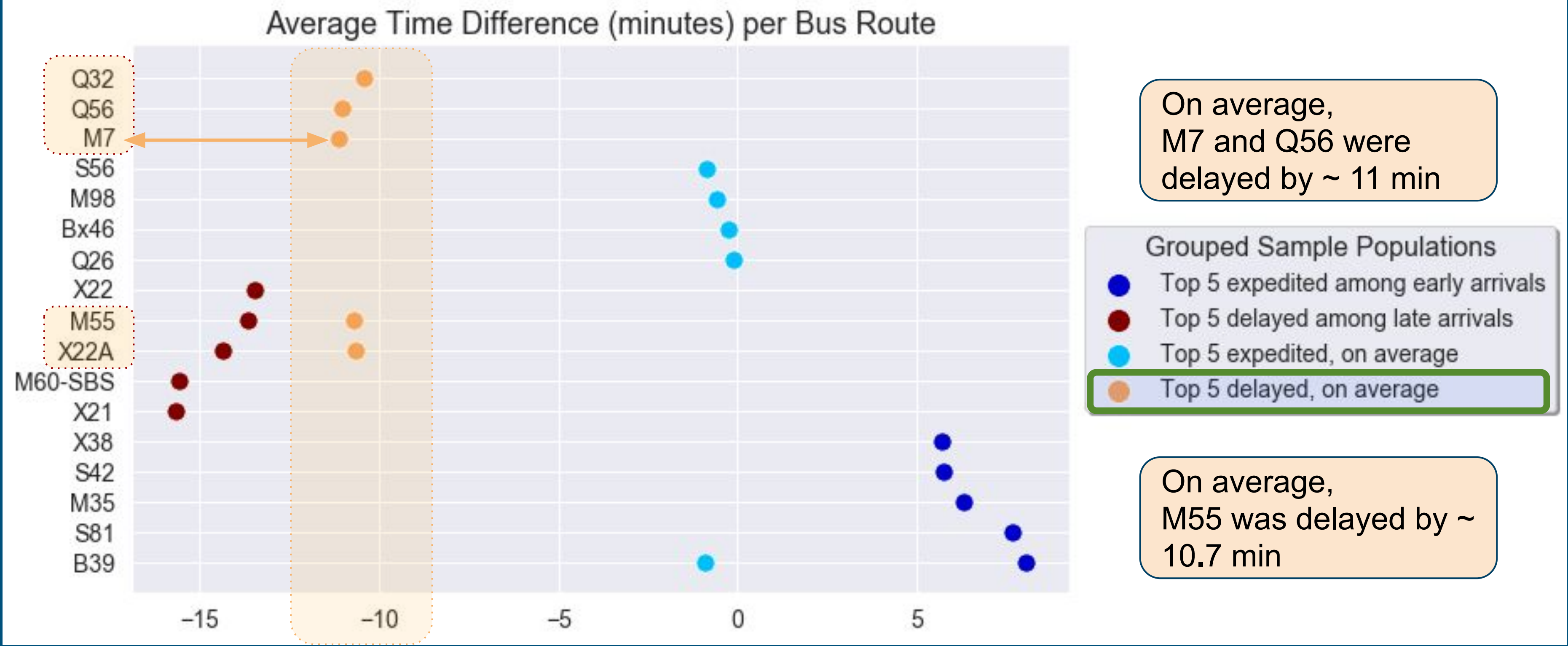

Average Time Difference (minutes) per Bus Route
Q32
Q56
M7
S56
M98
Bx46
Q26
X22
M55
X22A
M60-SBS
X21
X38
S42
M35
S81
B39
−15
−10
−5
0
5
Grouped Sample Populations
● Top 5 expedited among early arrivals
● Top 5 delayed among late arrivals
● Top 5 expedited, on average
● Top 5 delayed, on average

Four Groups


Average Time Difference (minutes) per Bus Route
Q32
Q56
M7
S56
M98
Bx46
Q26
X22
M55
X22A
M60-SBS
X21
X38
S42
M35
S81
B39
−15
−10
−5
0
5
Grouped Sample Populations
● Top 5 expedited among early arrivals
● Top 5 delayed among late arrivals
● Top 5 expedited, on average
● Top 5 delayed, on average



Q26 was delayed by only a few seconds, on average!

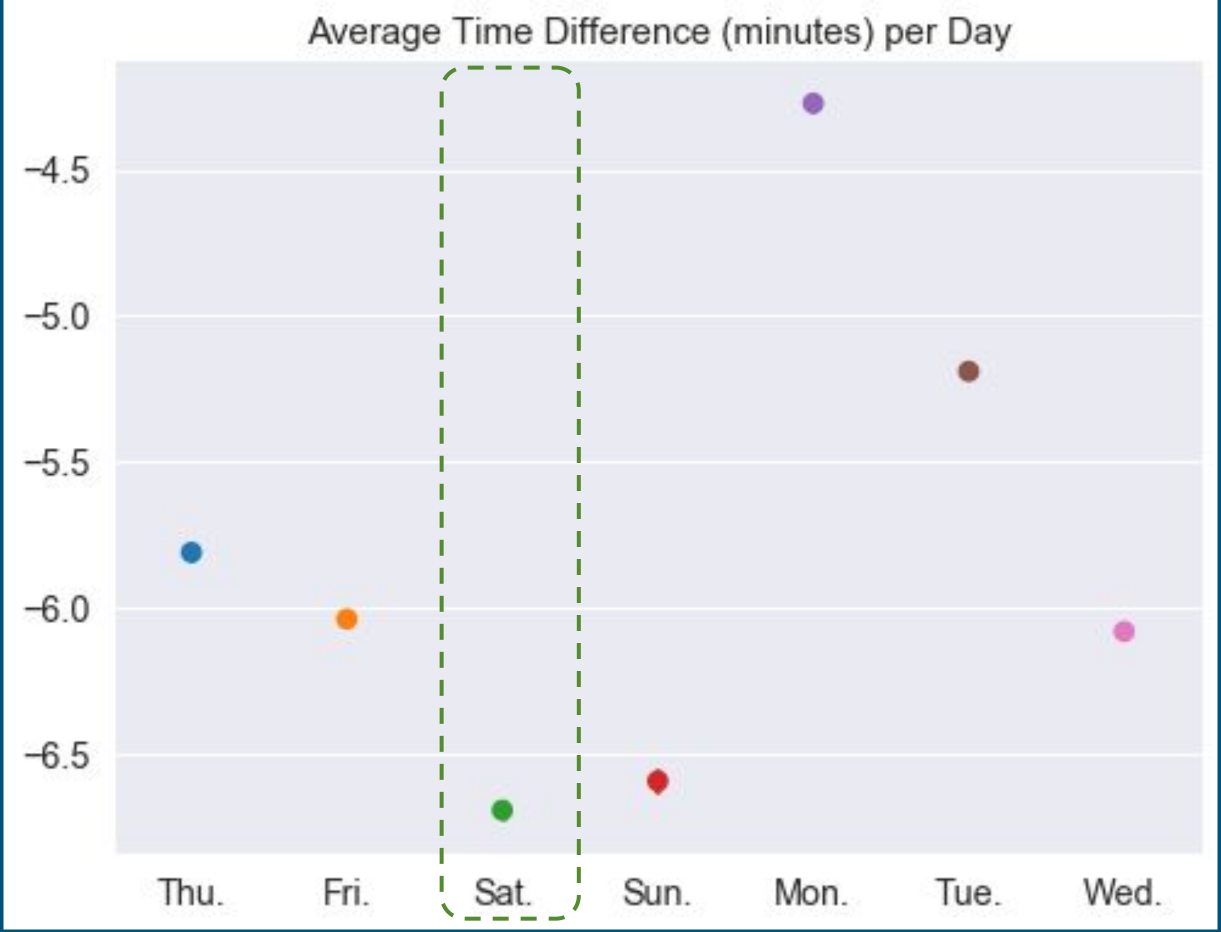


On average,
M7 and Q56 were
delayed by ~ 11 min

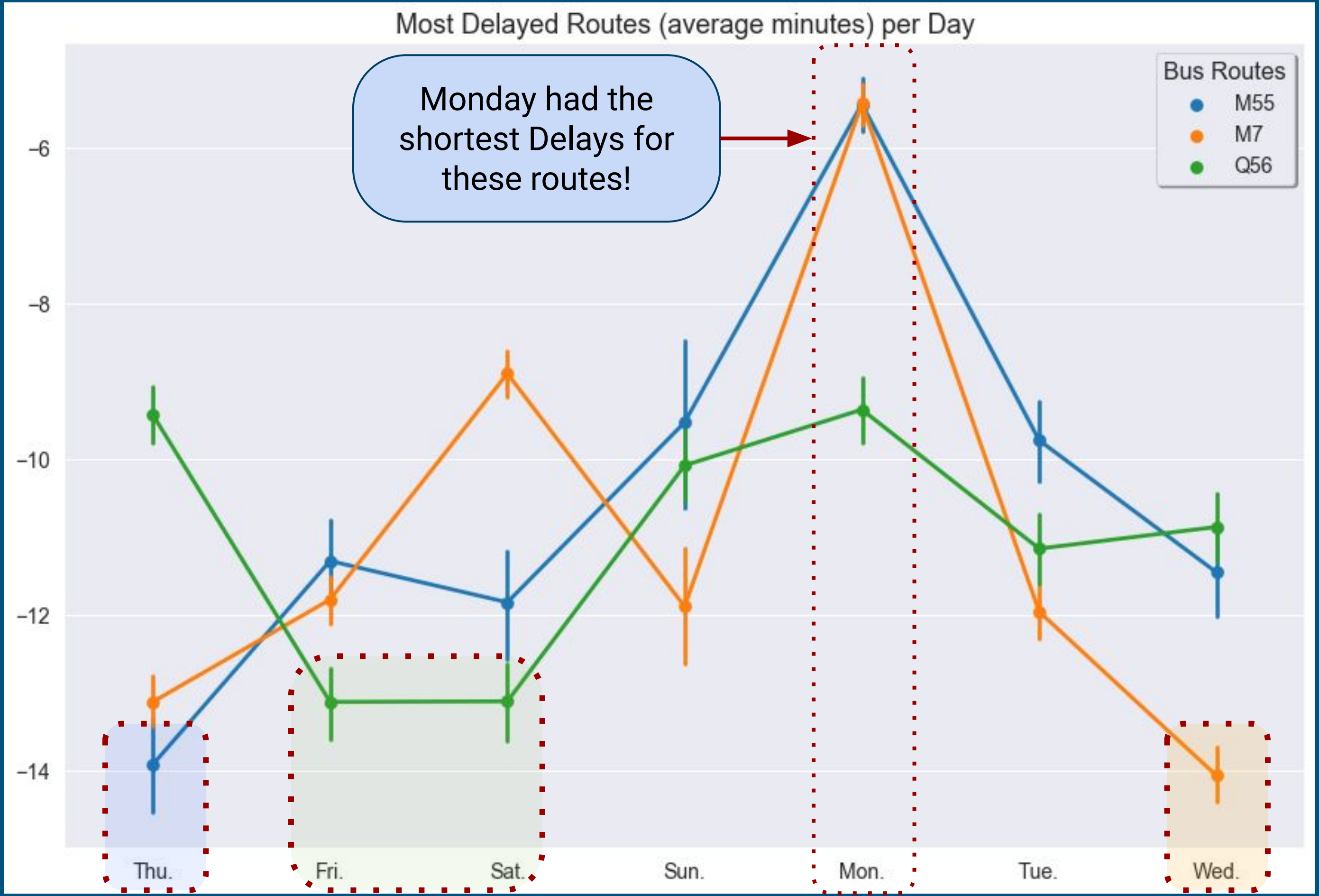
On average,
M55 was delayed by ~
10.7 min

Remember:
Overall Average Delay ~ 6 min

On average, which day of the week had the longest delay?


Average Time Difference (minutes) per Day
-4.5
-5.0
-5.5
-6.0
-6.5
Thu.
Fri.
Sat.
Sun.
Mon.
Tue.
Wed.

On average, which day of the week had the longest delay for the 3 most delayed routes?



Monday had the shortest Delays for these routes!

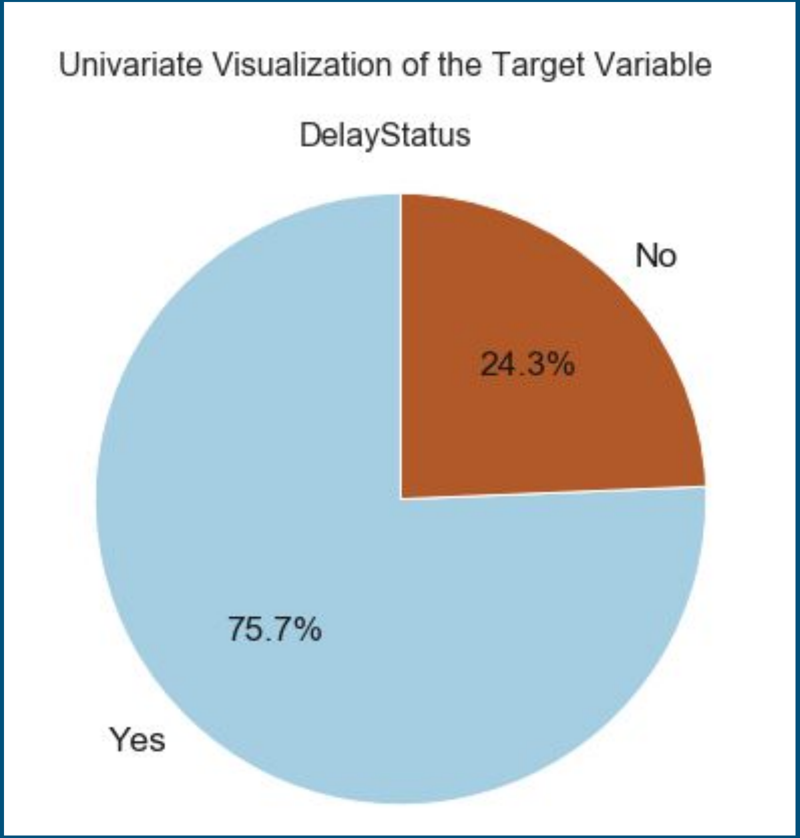
So .. Where is the Target variable?

Target?

DelayStatus

Making predictions?

Binary Classification Problem



No Count: 1,413,009

Yes Count: 4,391,359

Classes are Imbalanced!

Identifying the Target Variable (categorical)

Predictions!

Evaluating Model Performance

Which evaluation metric?

- ~~Accuracy~~
- Precision
- Recall
- ROC-AUC

Trade-off! But both are important!

Models should predict each target class fairly well

Best Two Supervised Models

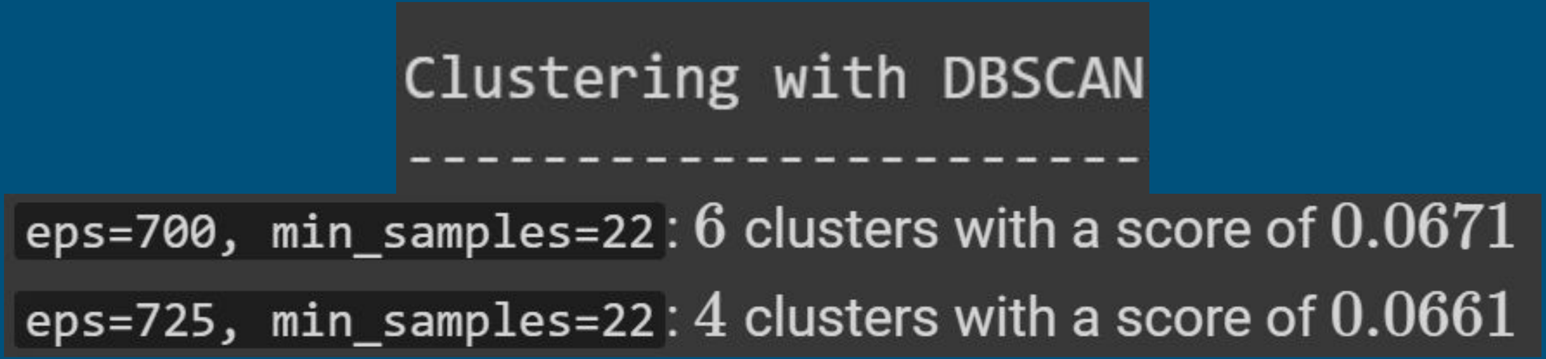
... Although all models underperformed overall ...

- 1st Place → Random Forest (PCA Feature-Set Variation) ran for ~ 2 hours
  - Correctly predicted 86% of delays with 76% precision
  - Predicted only 13% of non-delays with 24% precision
  - ROC-AUC Scores: 50.35% for "No" | 49.65% for "Yes"
  - Difference in Scores: 0.70%

- 2nd Place → Gradient Boosting (PCA Feature-Set Variation) ran for ~ 6 hours
  - Correctly predicted 77% of delays with 76% precision
  - Predicted only 22% of non-delays with 24% precision
  - ROC-AUC Scores: 51.38% for "No" | 48.62% for "Yes"
  - Difference in Scores: 2.76%

Clusters!

Best Variations of each Clustering Algorithm

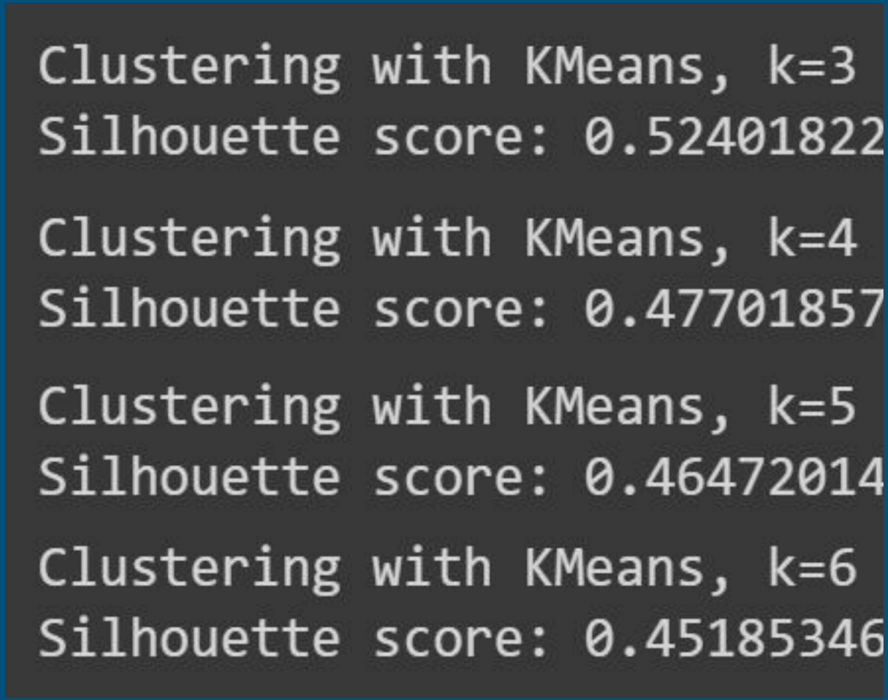


```
Clustering with DBSCAN
--------------------------
```

`eps=700, min_samples=22`: 6 clusters with a score of 0.0671

`eps=725, min_samples=22`: 4 clusters with a score of 0.0661

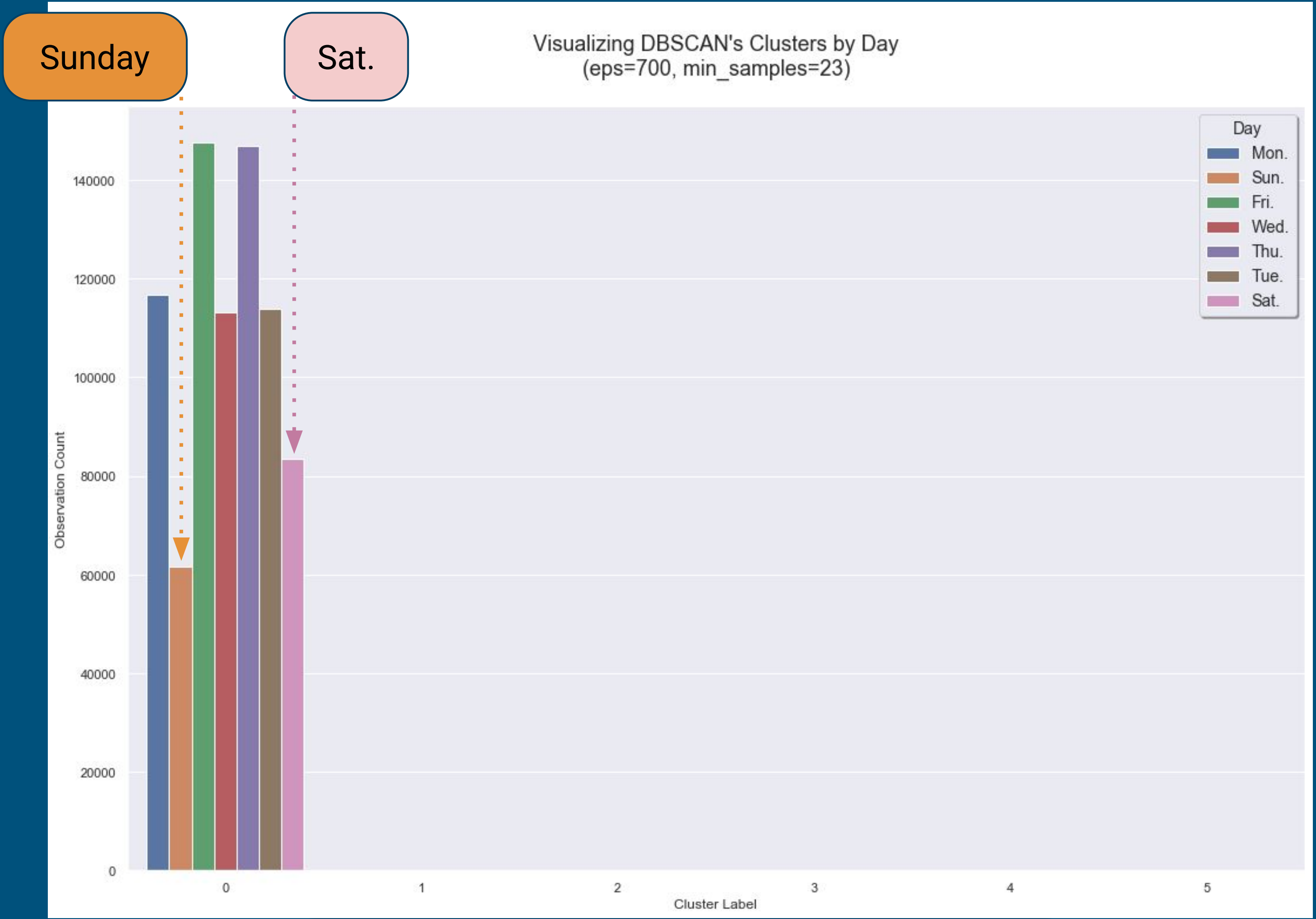
Best Variations of each Clustering Algorithm



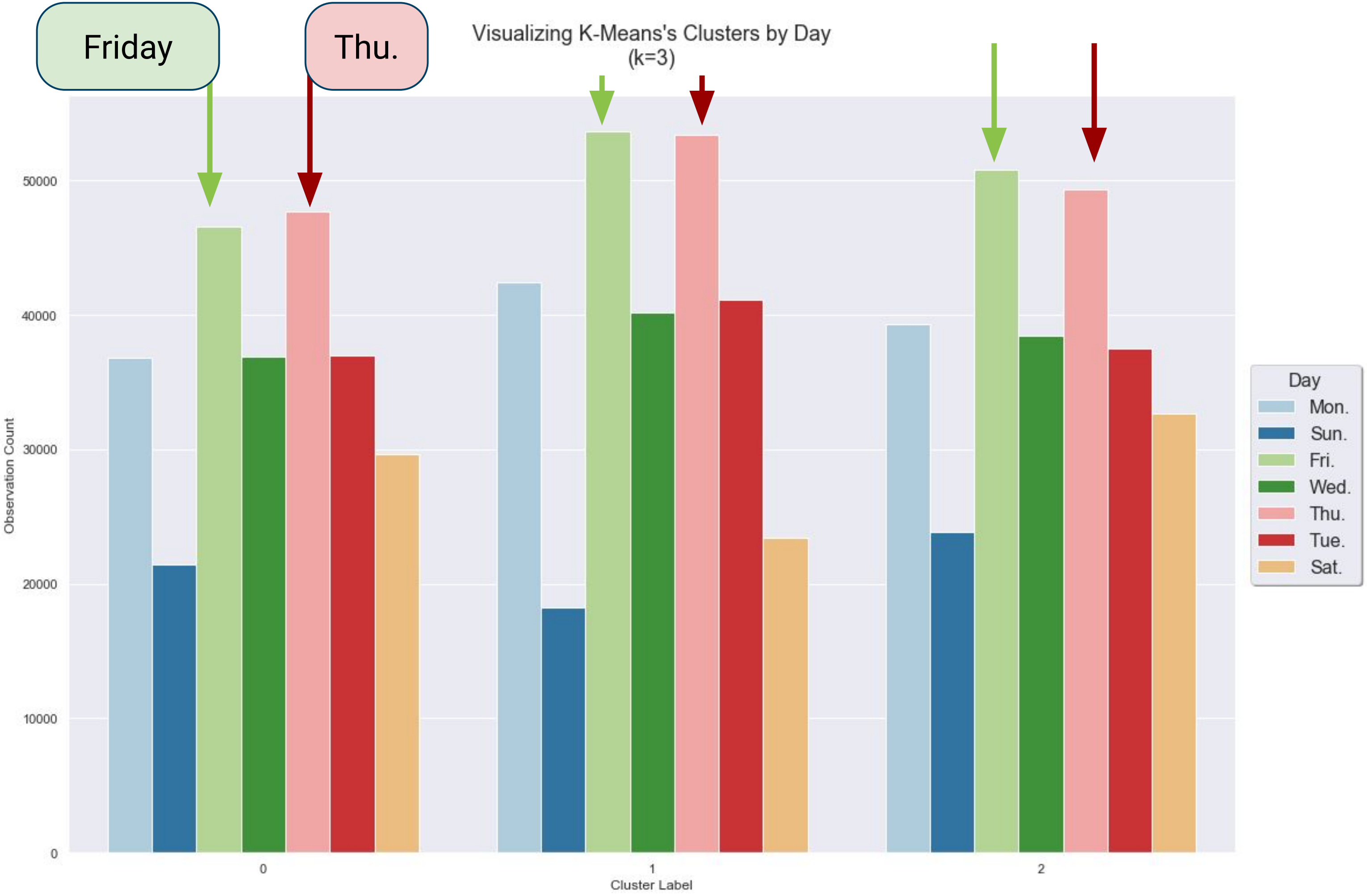
```
Clustering with KMeans, k=3
Silhouette score: 0.52401822

Clustering with KMeans, k=4
Silhouette score: 0.47701857

Clustering with KMeans, k=5
Silhouette score: 0.46472014

Clustering with KMeans, k=6
Silhouette score: 0.45185346
```

Best Score

How many records are in each cluster, with respect to days of the week?

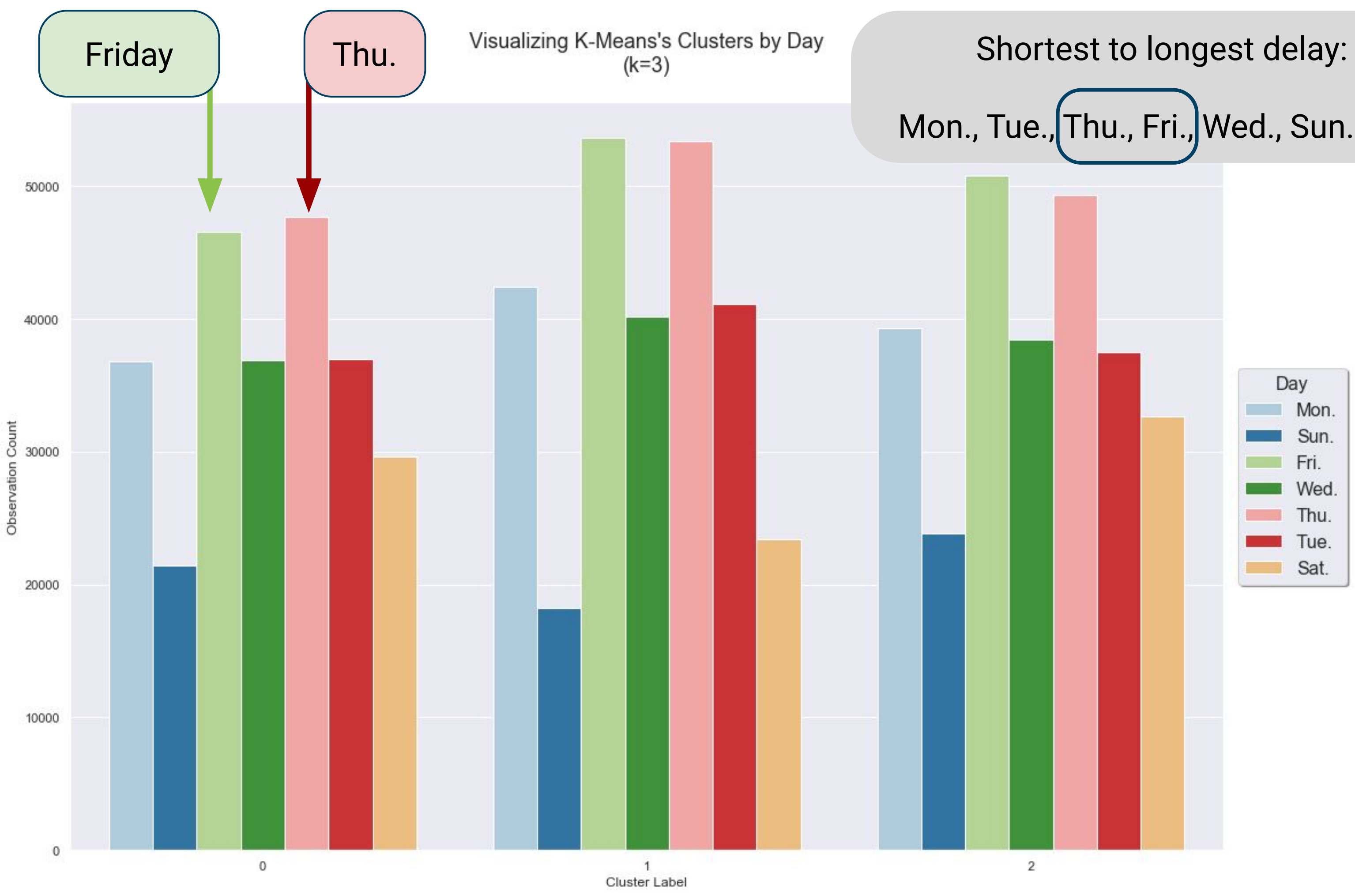


Sunday
Sat.
Visualizing DBSCAN's Clusters by Day
(eps=700, min_samples=23)
Day
Mon.
Sun.
Fri.
Wed.
Thu.
Tue.
Sat.
Observation Count
140000
120000
100000
80000
60000
40000
20000
0
Cluster Label
0
1
2
3
4
5


Visualizing K-Means's Clusters by Day
(k=3)
Observation Count
0
10000
20000
30000
40000
50000
0
1
2
Cluster Label
Day
Mon.
Sun.
Fri.
Wed.
Thu.
Tue.
Sat.

Friday

Thu.



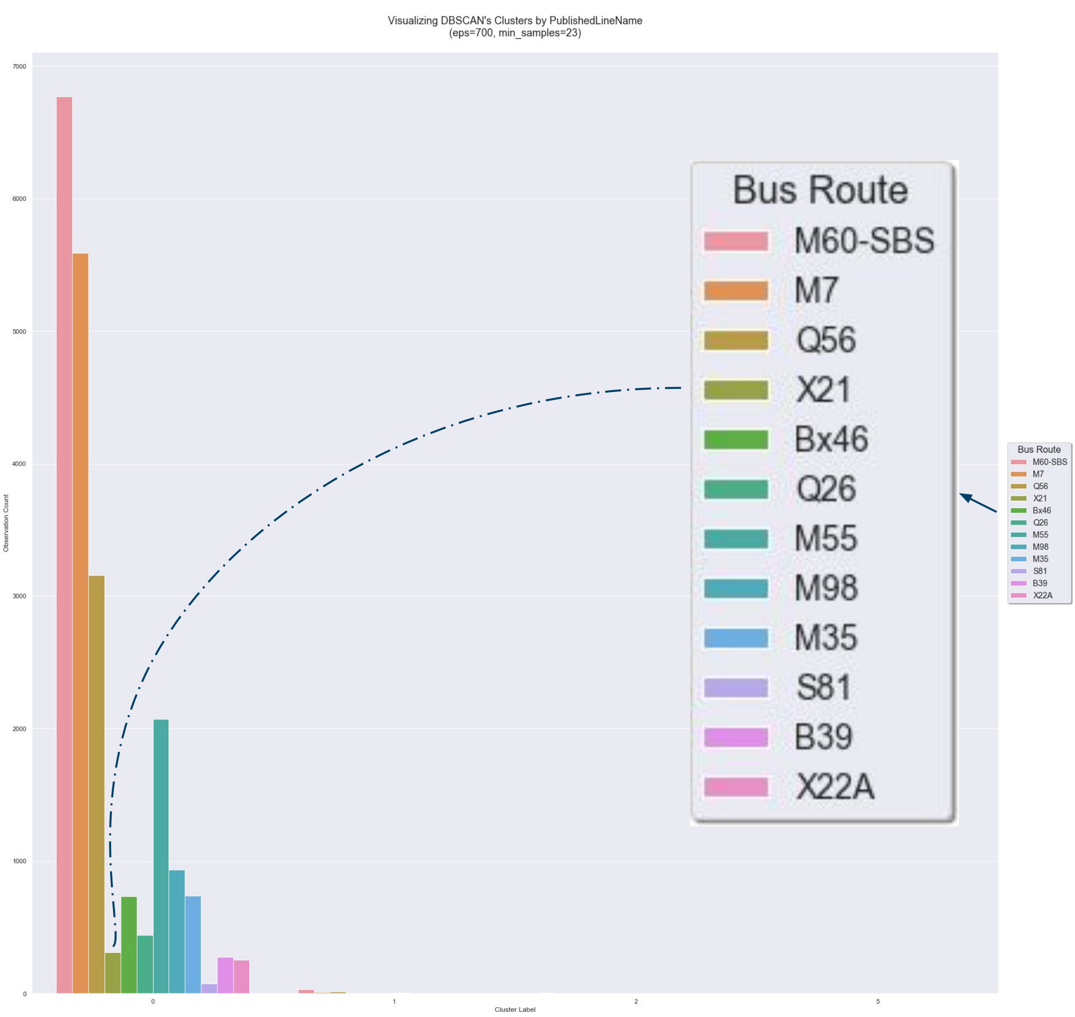
Friday

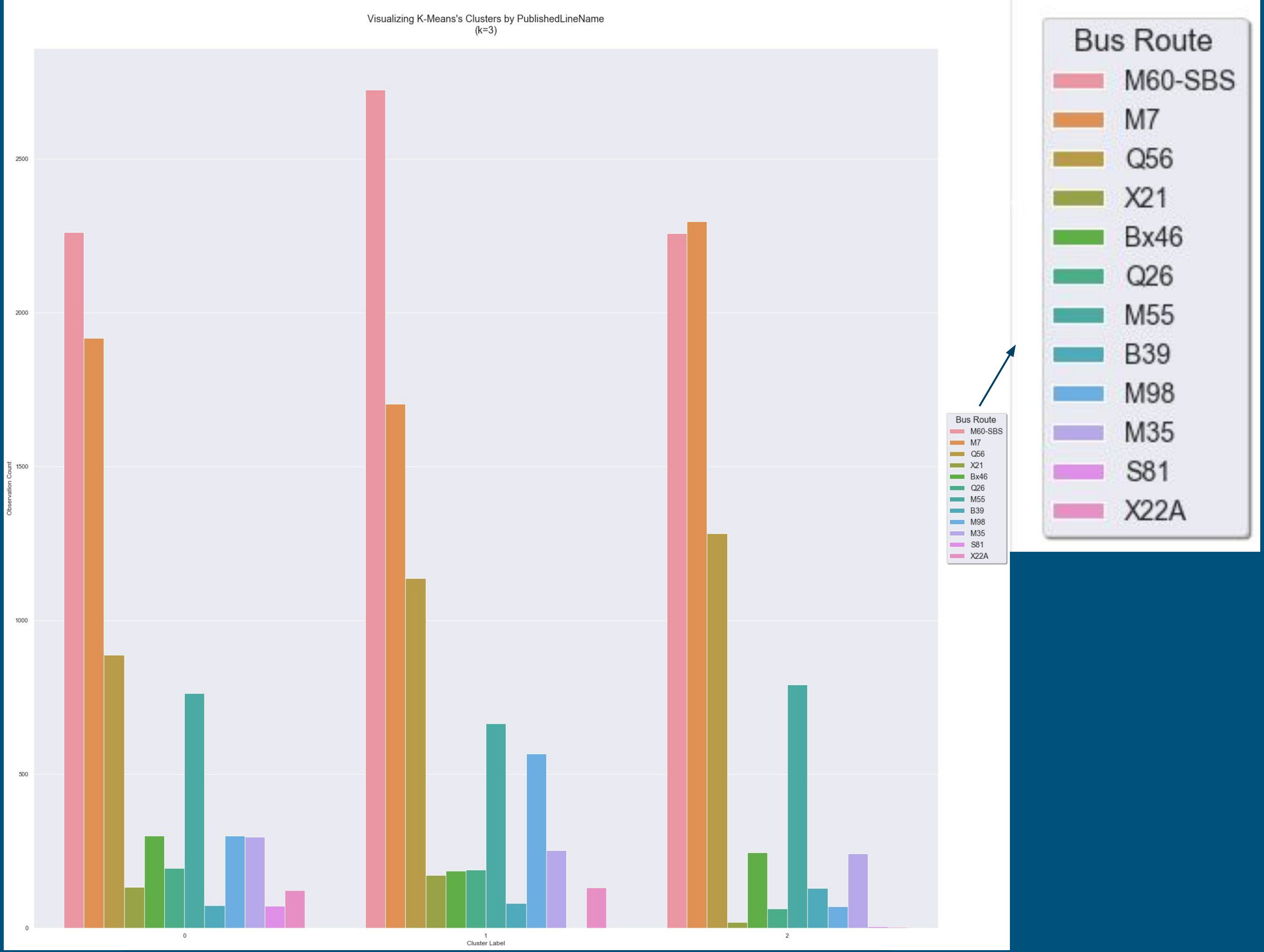
Thu.

Shortest to longest delay:

Mon., Tue., Thu., Fri., Wed., Sun., Sat.

How many records are in each cluster, w.r.t. the bus routes?


Visualizing DBSCAN's Clusters by PublishedLineName
(eps=700, min_samples=23)
Observation Count
7000
6000
5000
4000
3000
2000
1000
0
Cluster Label
0
1
2
5
Bus Route
M60-SBS
M7
Q56
X21
Bx46
Q26
M55
M98
M35
S81
B39
X22A
Bus Route
M60-SBS
M7
Q56
X21
Bx46
Q26
M55
M98
M35
S81
B39
X22A


Visualizing K-Means's Clusters by PublishedLineName
(k=3)
Observation Count
2500
2000
1500
1000
500
0
0
1
2
Cluster Label
Bus Route
M60-SBS
M7
Q56
X21
Bx46
Q26
M55
B39
M98
M35
S81
X22A
Bus Route
M60-SBS
M7
Q56
X21
Bx46
Q26
M55
B39
M98
M35
S81
X22A

Clusters: Important Notes

- Records could be clustered based on a combination of the following:
  - Duration of delay (average duration, max., min., frequency, etc.)
  - The time intervals per day in which delays occurred,
  - Bus routes
  - Days of the week (Each cluster showed a similar number of records for Mon, Tue, & Wed)

- Other Details:
  - Normal business hours may have more consistent patterns.
  - Hours at which delays occurred could be very similar for some days.
  - Service maintenance may occur on weekends.

Clusters: Important Notes

- Thu. and Fri. were represented the most in each cluster.
    - They had similar delay-averages.

- Sat. & Sun. were represented the least in each cluster.
    - They also had similar delay averages.

- Routes M60-SBS, M7, and Q56:
    - Consistently had largest presence across all clusters.
    - If a bus for route M60-SBS is delayed, it will typically arrive 15 minutes behind schedule.

Next Steps

As always → Further research is needed

- Ridership per bus route, & ridership per section (or bus stop) for each route.
- Delays due to overcrowded buses under normal conditions.
- Delays due to service malfunctions (like inoperable buses).
- Delays due to civil road work or traffic accidents.
- Delays due to weather patterns.
- etc.

Next Steps for Further Research

1. Investigate how NYC bus ridership and performances have changed from June of 2017 through summer of 2020 (to find more patterns amid the pandemic).
2. Analyze how delayed (or expedited) arrival times per day have changed over time for the most delayed bus routes.
3. Explore various aspects of emergencies, public events, and weather data for June 2017 (and 2020).
4. Implement ML algorithms using Big Data technologies with additional data (more records, more factors, etc.).
5. Use supervised machine learning models (perhaps ensembles or Neural Networks) to estimate the actual expected arrival times as a regression problem.
6. Expand on evaluation metrics and tools for all algorithms used.
7. Based on those research results, discuss the newly discovered patterns surrounding bus delays.

Thank you for your time!

Any questions?

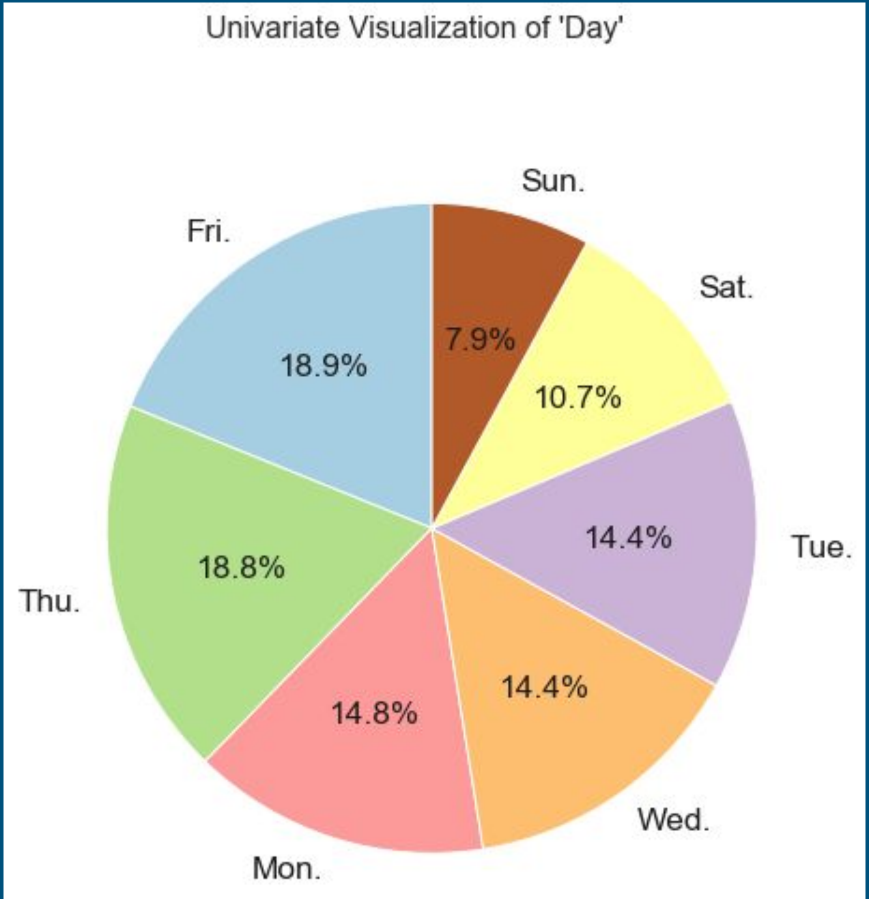
Appendix

Who is this project for?

- Chief Executive Officers
- Data Scientists
- Machine Learning enthusiasts
- Transportation Providers
- .. anyone who is inherently inquisitive :)

Exploration

The dataset consists mostly of weekday trips.



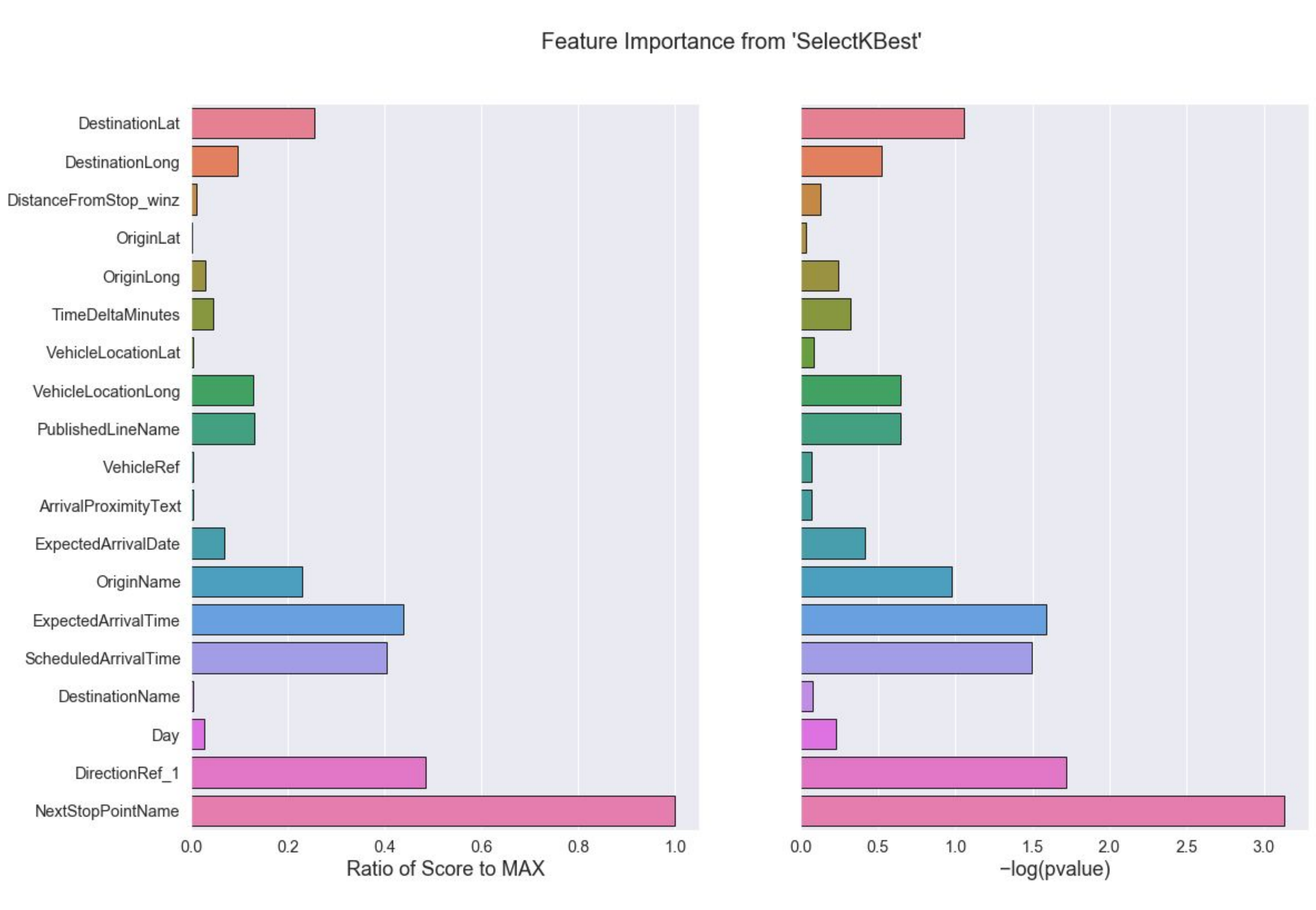
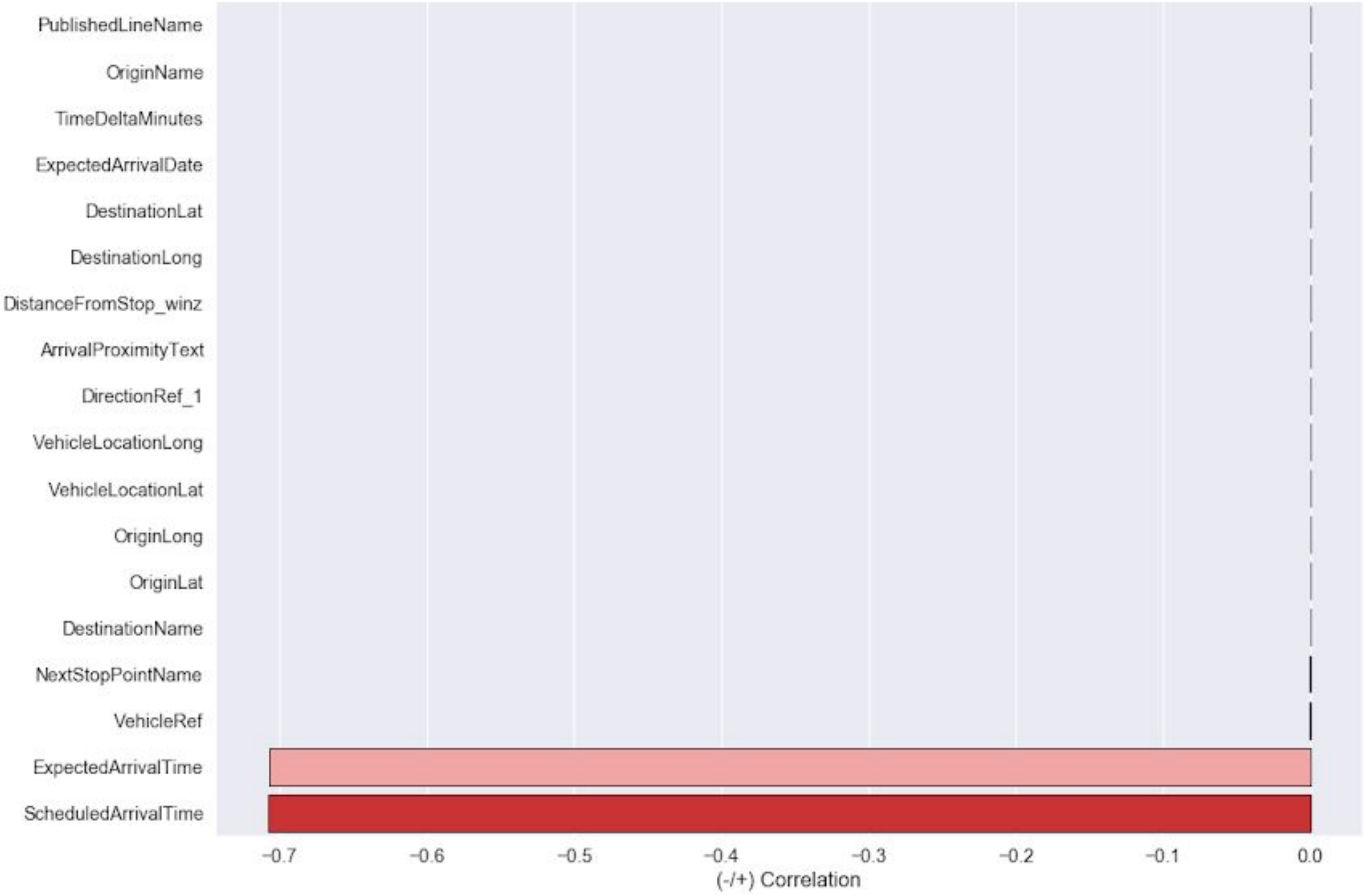

Feature Importance from 'SelectKBest'
DestinationLat
DestinationLong
DistanceFromStop_winz
OriginLat
OriginLong
TimeDeltaMinutes
VehicleLocationLat
VehicleLocationLong
PublishedLineName
VehicleRef
ArrivalProximityText
ExpectedArrivalDate
OriginName
ExpectedArrivalTime
ScheduledArrivalTime
DestinationName
Day
DirectionRef_1
NextStopPointName
0.0
0.2
0.4
0.6
0.8
1.0
Ratio of Score to MAX
0.0
0.5
1.0
1.5
2.0
2.5
3.0
−log(pvalue)

Figure 4.1 - Scores sorted by Correlation between Feature & PC:
Most Descriptive Features for 'PC_1'



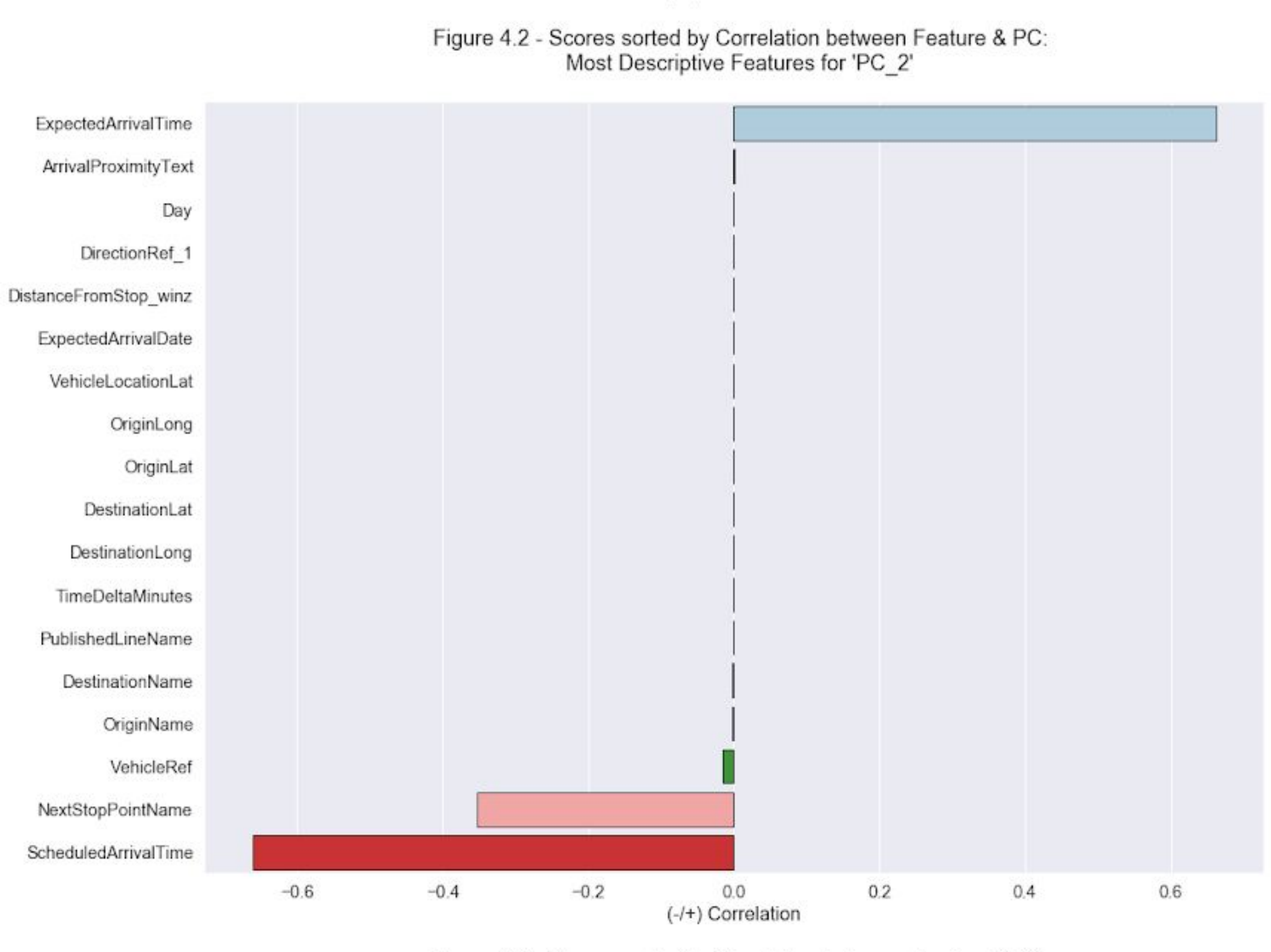


Figure 4.2 - Scores sorted by Correlation between Feature & PC:
Most Descriptive Features for 'PC_2'
ExpectedArrivalTime
ArrivalProximityText
Day
DirectionRef_1
DistanceFromStop_winz
ExpectedArrivalDate
VehicleLocationLat
OriginLong
OriginLat
DestinationLat
DestinationLong
TimeDeltaMinutes
PublishedLineName
DestinationName
OriginName
VehicleRef
NextStopPointName
ScheduledArrivalTime
−0.6
−0.4
−0.2
0.0
0.2
0.4
0.6
(-/+) Correlation

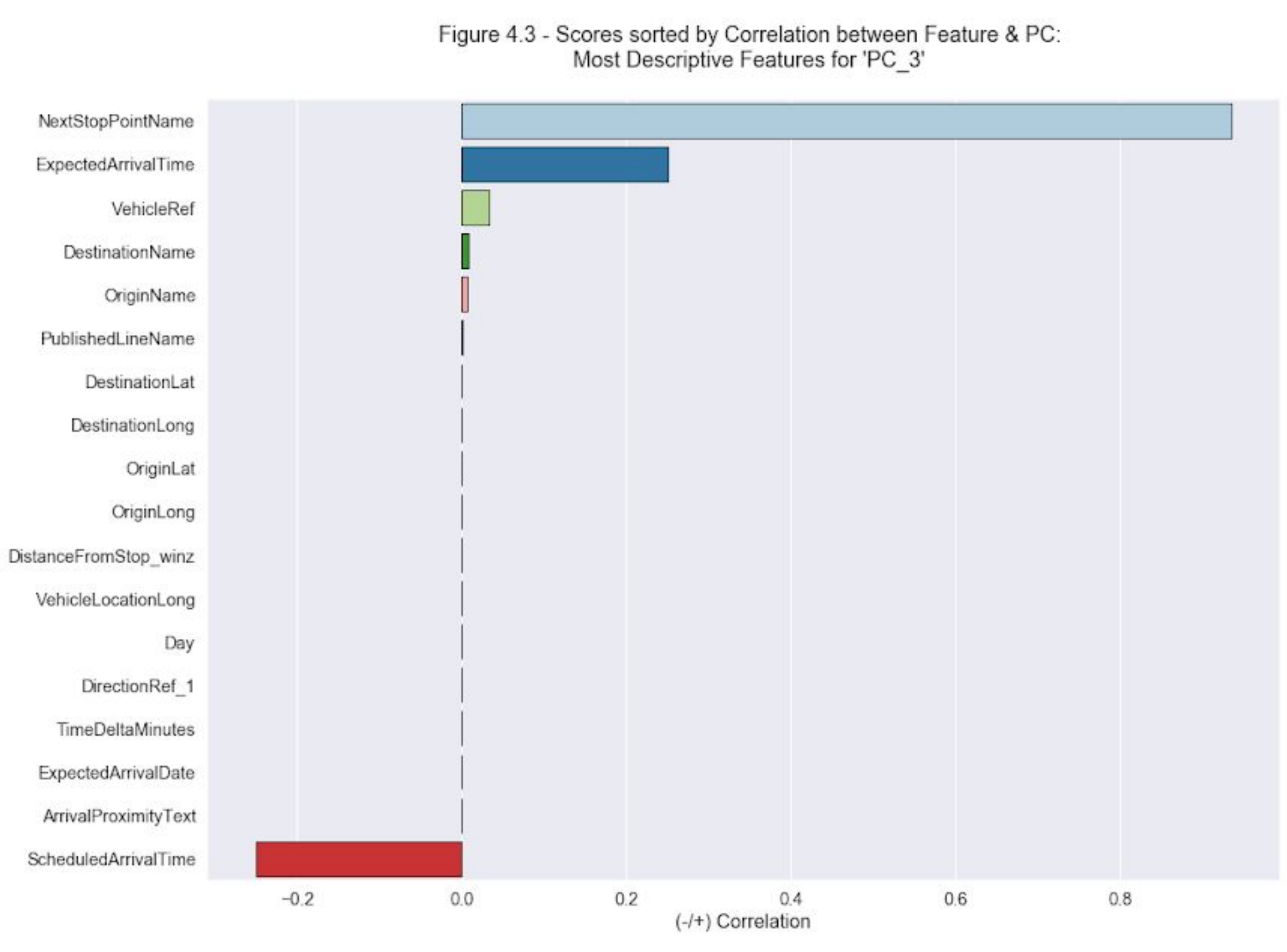


Figure 4.3 - Scores sorted by Correlation between Feature & PC: Most Descriptive Features for 'PC_3'

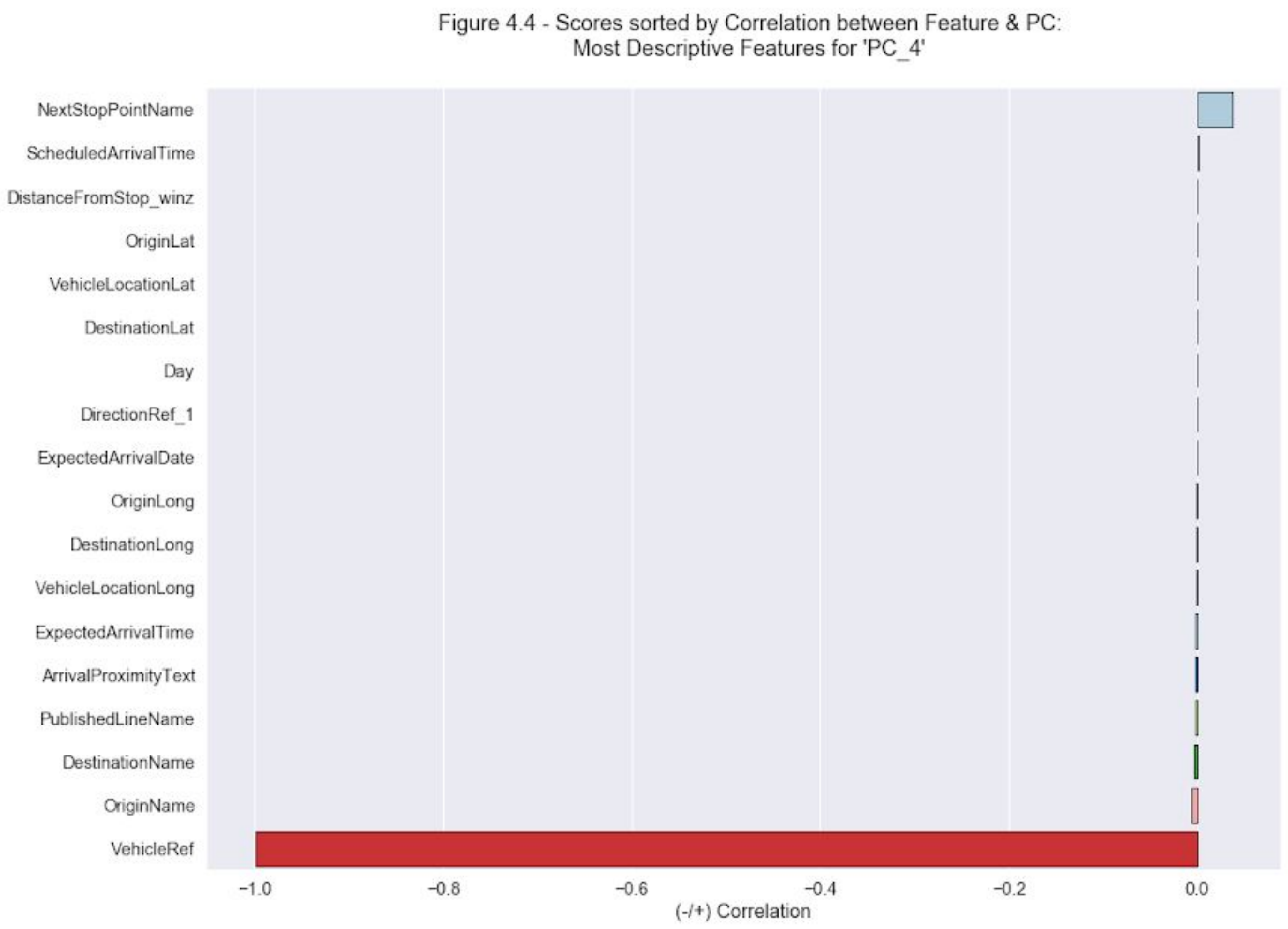
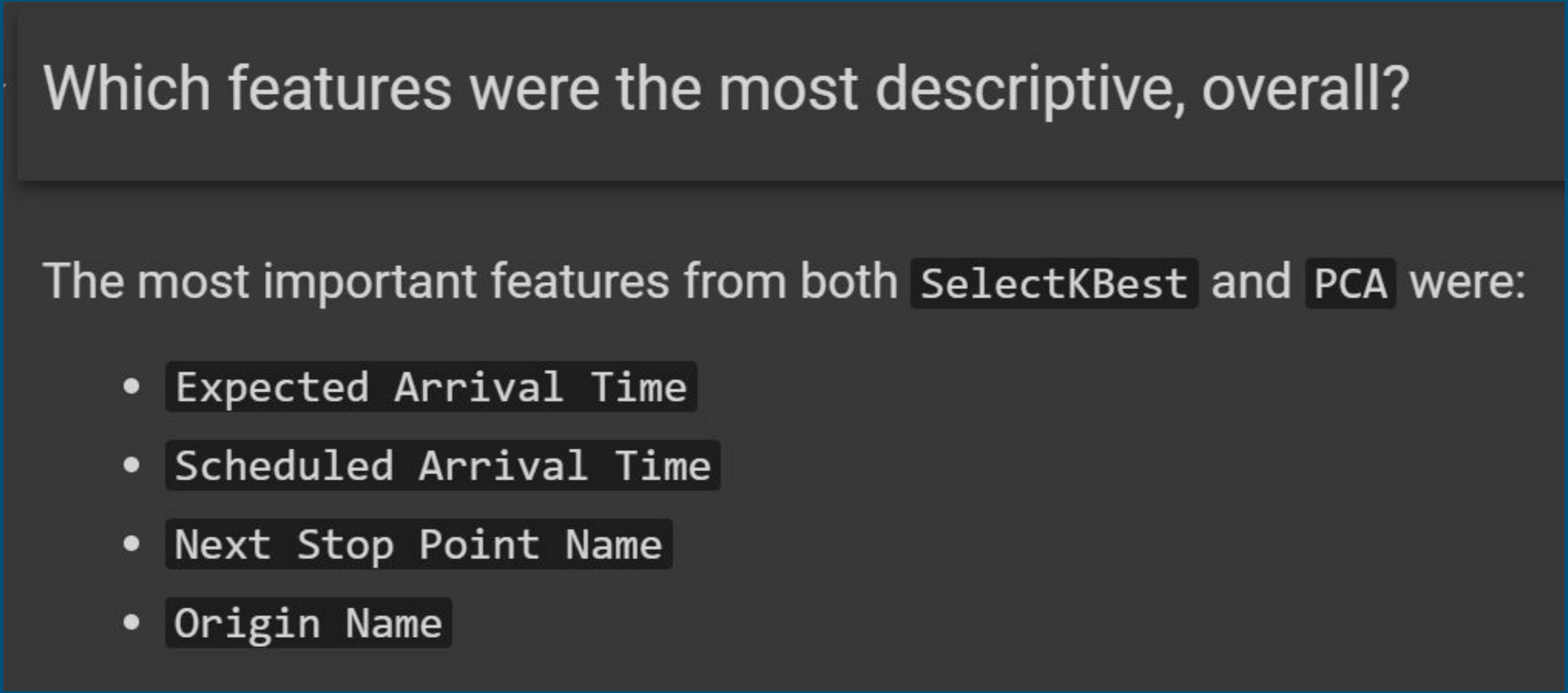


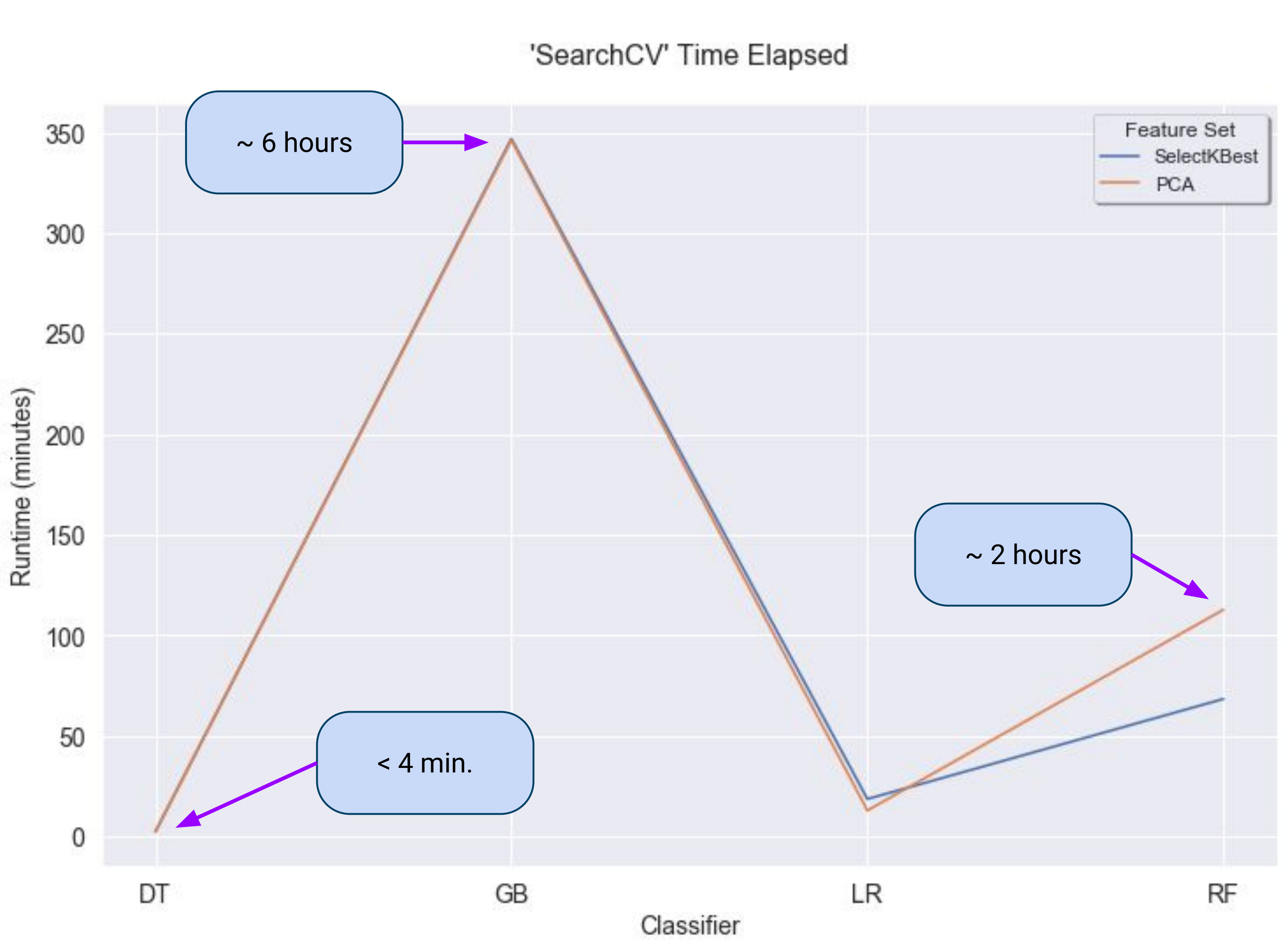
Figure 4.4 - Scores sorted by Correlation between Feature & PC:
Most Descriptive Features for 'PC_4'
NextStopPointName
ScheduledArrivalTime
DistanceFromStop_winz
OriginLat
VehicleLocationLat
DestinationLat
Day
DirectionRef_1
ExpectedArrivalDate
OriginLong
DestinationLong
VehicleLocationLong
ExpectedArrivalTime
ArrivalProximityText
PublishedLineName
DestinationName
OriginName
VehicleRef
−1.0
−0.8
−0.6
−0.4
−0.2
0.0
(-/+) Correlation



Which features were the most descriptive, overall?

The most important features from both `SelectKBest` and `PCA` were:

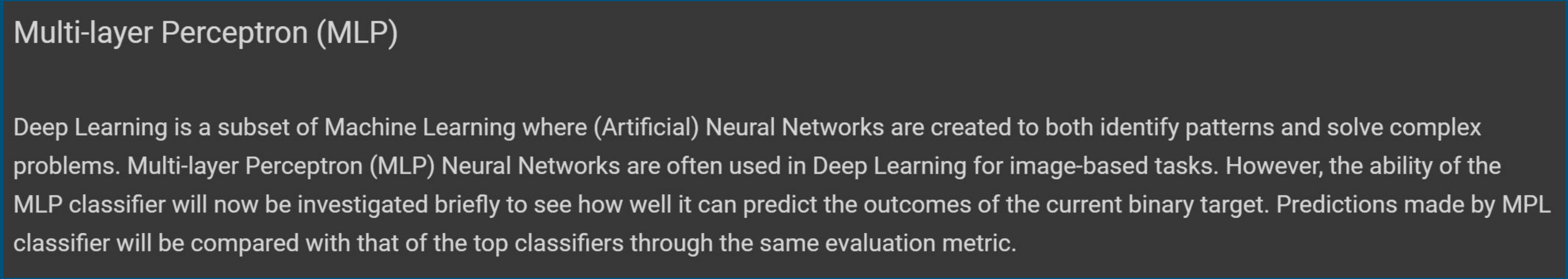
- `Expected Arrival Time`
- `Scheduled Arrival Time`
- `Next Stop Point Name`
- `Origin Name`



~ 6 hours

~ 2 hours

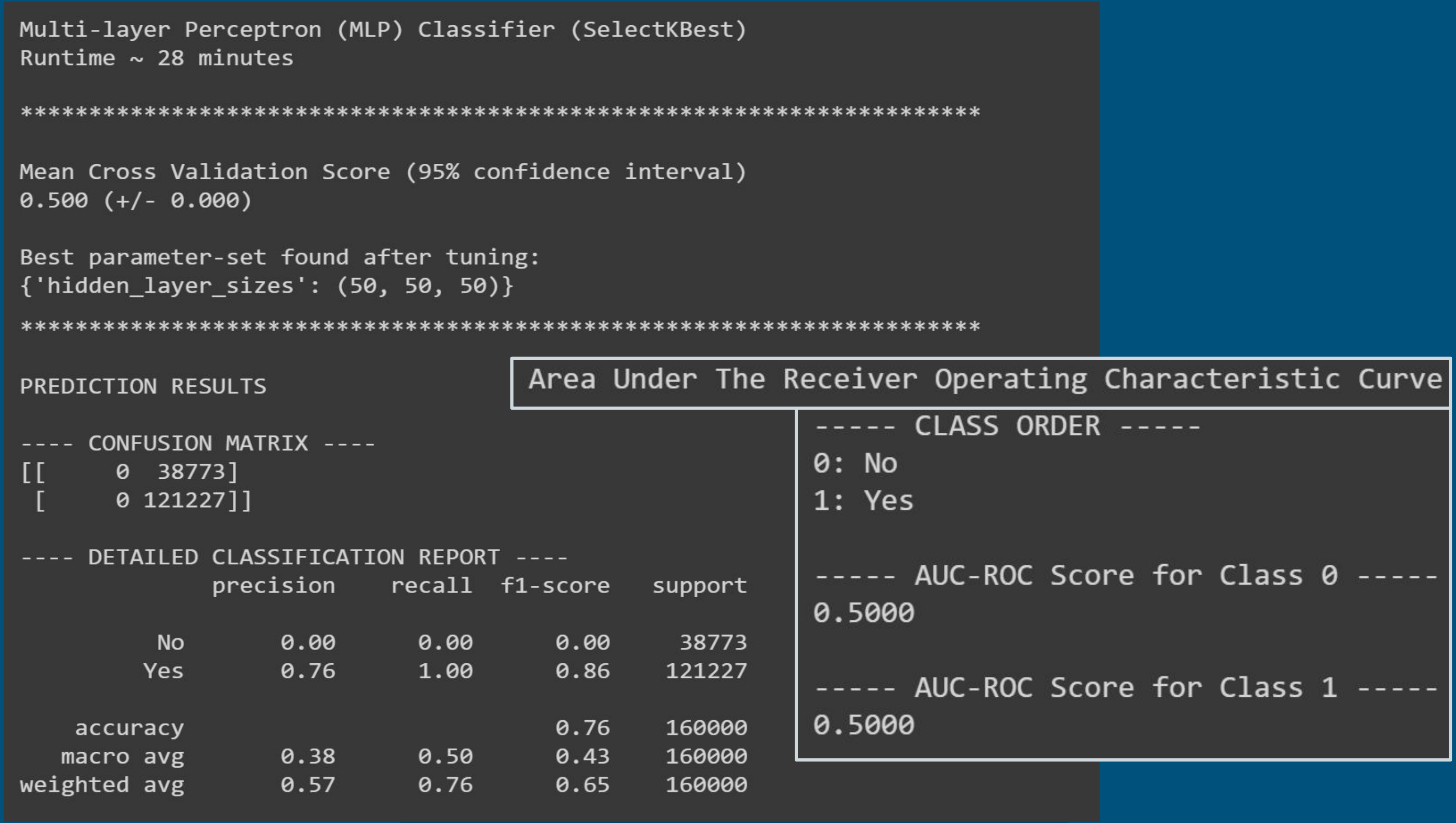
< 4 min.



Multi-layer Perceptron (MLP)

Deep Learning is a subset of Machine Learning where (Artificial) Neural Networks are created to both identify patterns and solve complex problems. Multi-layer Perceptron (MLP) Neural Networks are often used in Deep Learning for image-based tasks. However, the ability of the MLP classifier will now be investigated briefly to see how well it can predict the outcomes of the current binary target. Predictions made by MPL classifier will be compared with that of the top classifiers through the same evaluation metric.

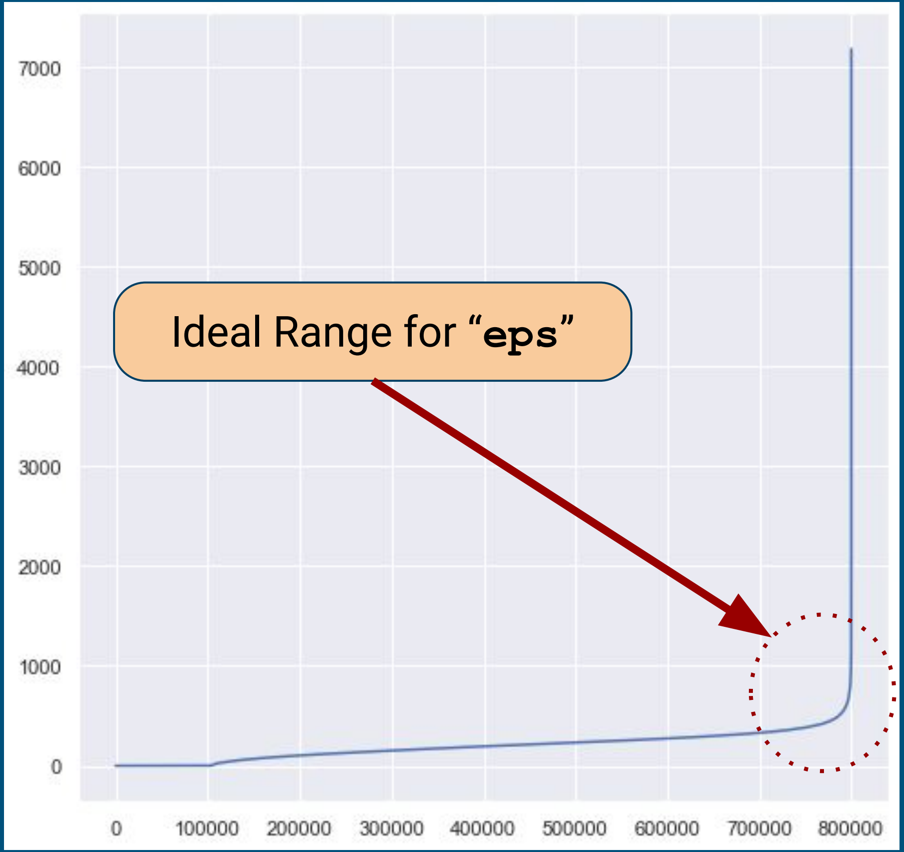
Deep Learning, MLP Neural Network



```
Multi-layer Perceptron (MLP) Classifier (SelectKBest)
Runtime ~ 28 minutes

**********************************************************************

Mean Cross Validation Score (95% confidence interval)
0.500 (+/- 0.000)

Best parameter-set found after tuning:
{'hidden_layer_sizes': (50, 50, 50)}

**********************************************************************

PREDICTION RESULTS

---- CONFUSION MATRIX ----
[[     0  38773]
 [     0 121227]]

---- DETAILED CLASSIFICATION REPORT ----
              precision    recall  f1-score   support

          No       0.00      0.00      0.00     38773
         Yes       0.76      1.00      0.86    121227

    accuracy                           0.76    160000
   macro avg       0.38      0.50      0.43    160000
weighted avg       0.57      0.76      0.65    160000
```

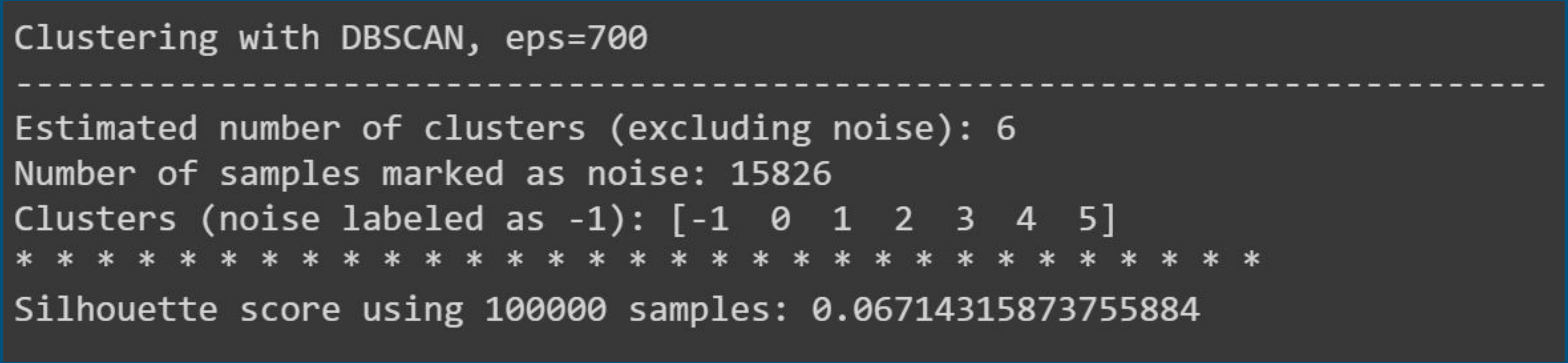
```
Area Under The Receiver Operating Characteristic Curve
```

```
----- CLASS ORDER -----
0: No
1: Yes

----- AUC-ROC Score for Class 0 -----
0.5000

----- AUC-ROC Score for Class 1 -----
0.5000
```

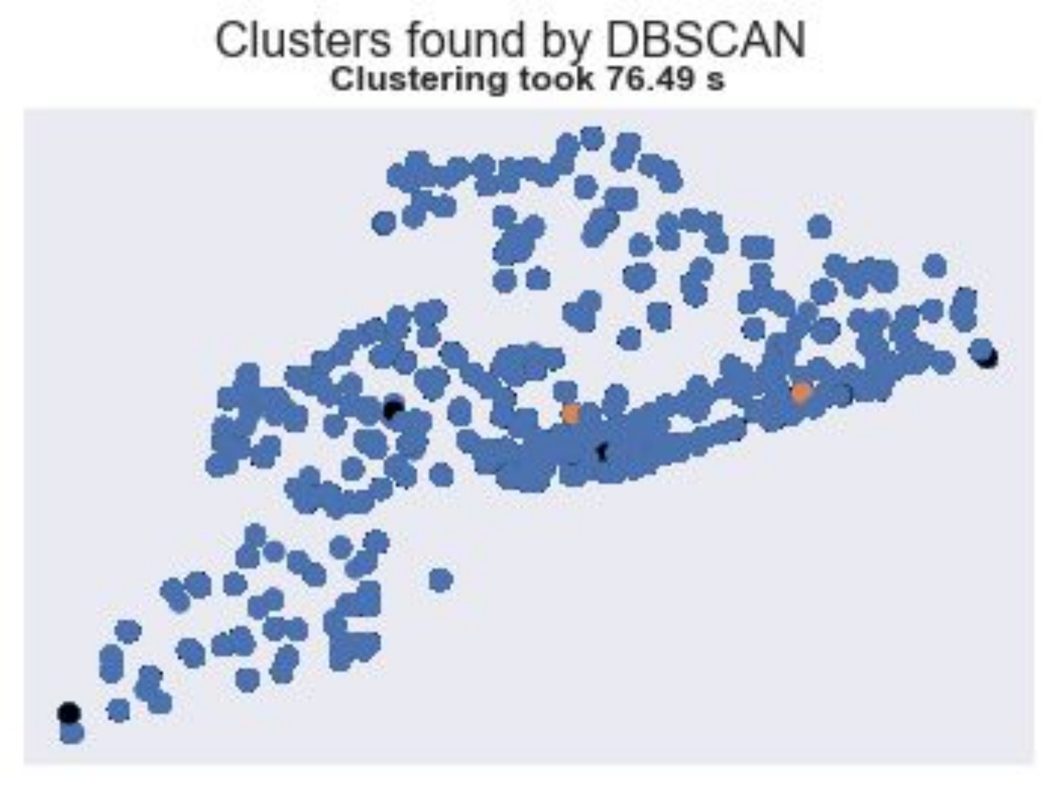


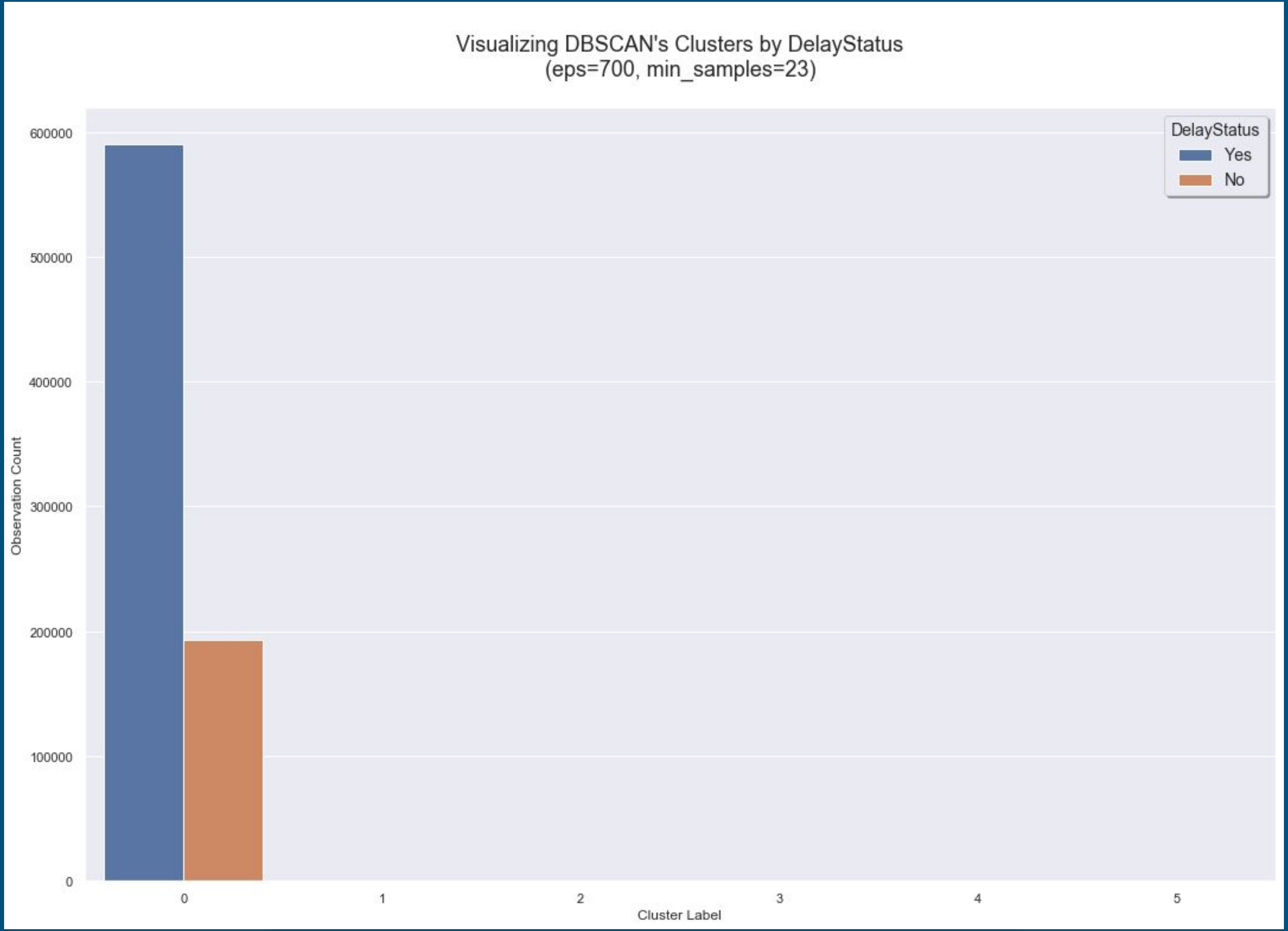
Determining “eps” parameter for DBSCAN from elbow

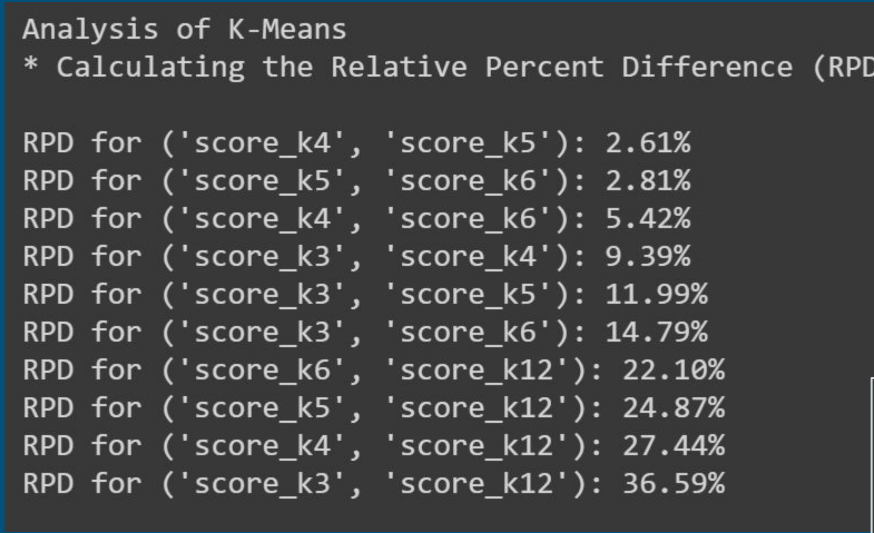
Best DBSCAN variation



```
Clustering with DBSCAN, eps=700
---------------------------------------------------------------------------
Estimated number of clusters (excluding noise): 6
Number of samples marked as noise: 15826
Clusters (noise labeled as -1): [-1  0  1  2  3  4  5]
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
Silhouette score using 100000 samples: 0.06714315873755884
```

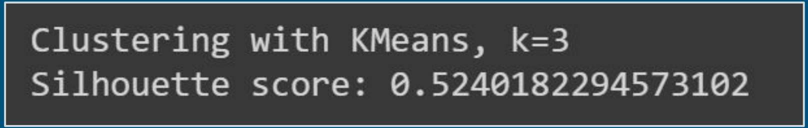

Clusters found by DBSCAN
Clustering took 76.49 s


Visualizing DBSCAN's Clusters by DelayStatus
(eps=700, min_samples=23)
DelayStatus
Yes
No
Observation Count
600000
500000
400000
300000
200000
100000
0
0
1
2
3
4
5
Cluster Label



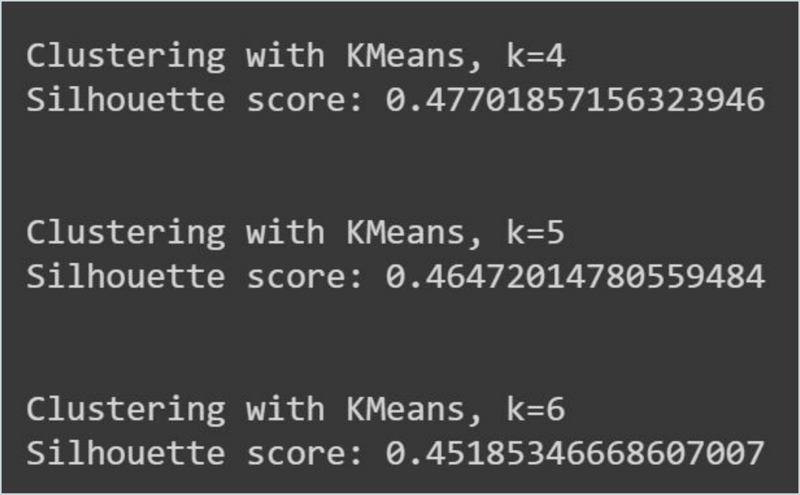
```
Analysis of K-Means
* Calculating the Relative Percent Difference (RPD) of Silhouette Scores

RPD for ('score_k4', 'score_k5'): 2.61%
RPD for ('score_k5', 'score_k6'): 2.81%
RPD for ('score_k4', 'score_k6'): 5.42%
RPD for ('score_k3', 'score_k4'): 9.39%
RPD for ('score_k3', 'score_k5'): 11.99%
RPD for ('score_k3', 'score_k6'): 14.79%
RPD for ('score_k6', 'score_k12'): 22.10%
RPD for ('score_k5', 'score_k12'): 24.87%
RPD for ('score_k4', 'score_k12'): 27.44%
RPD for ('score_k3', 'score_k12'): 36.59%
```

Silhouette Scores: K-Means

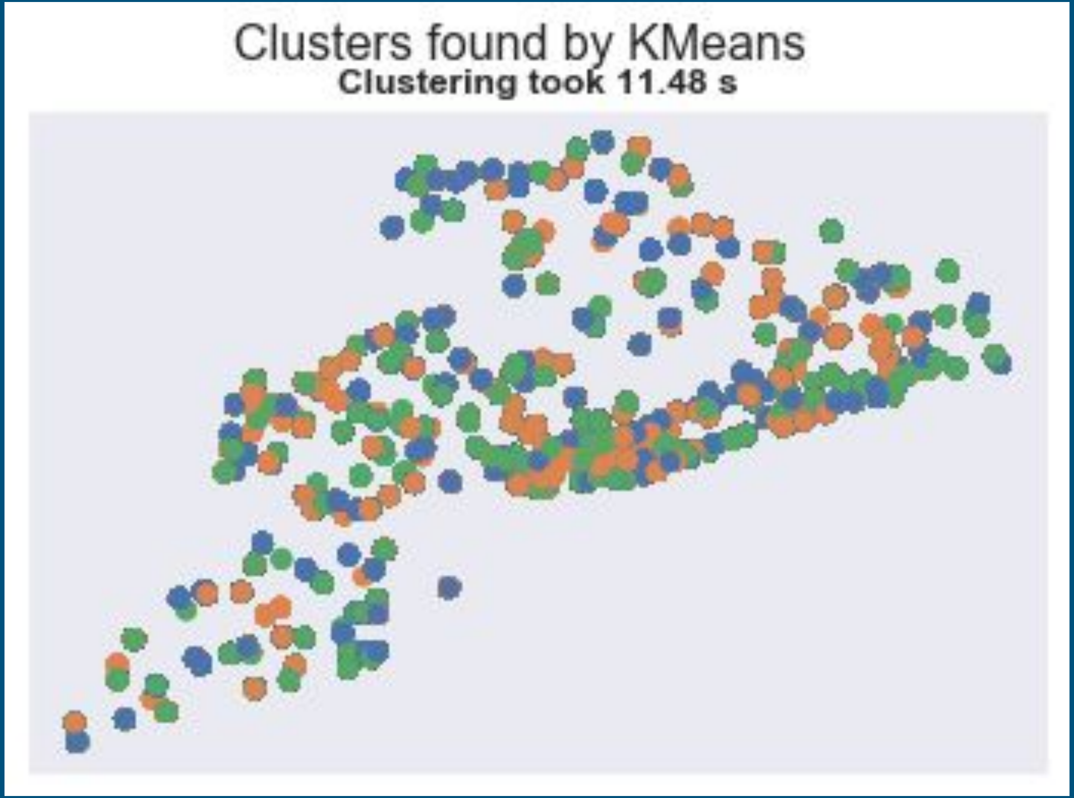


```
Clustering with KMeans, k=3
Silhouette score: 0.5240182294573102
```

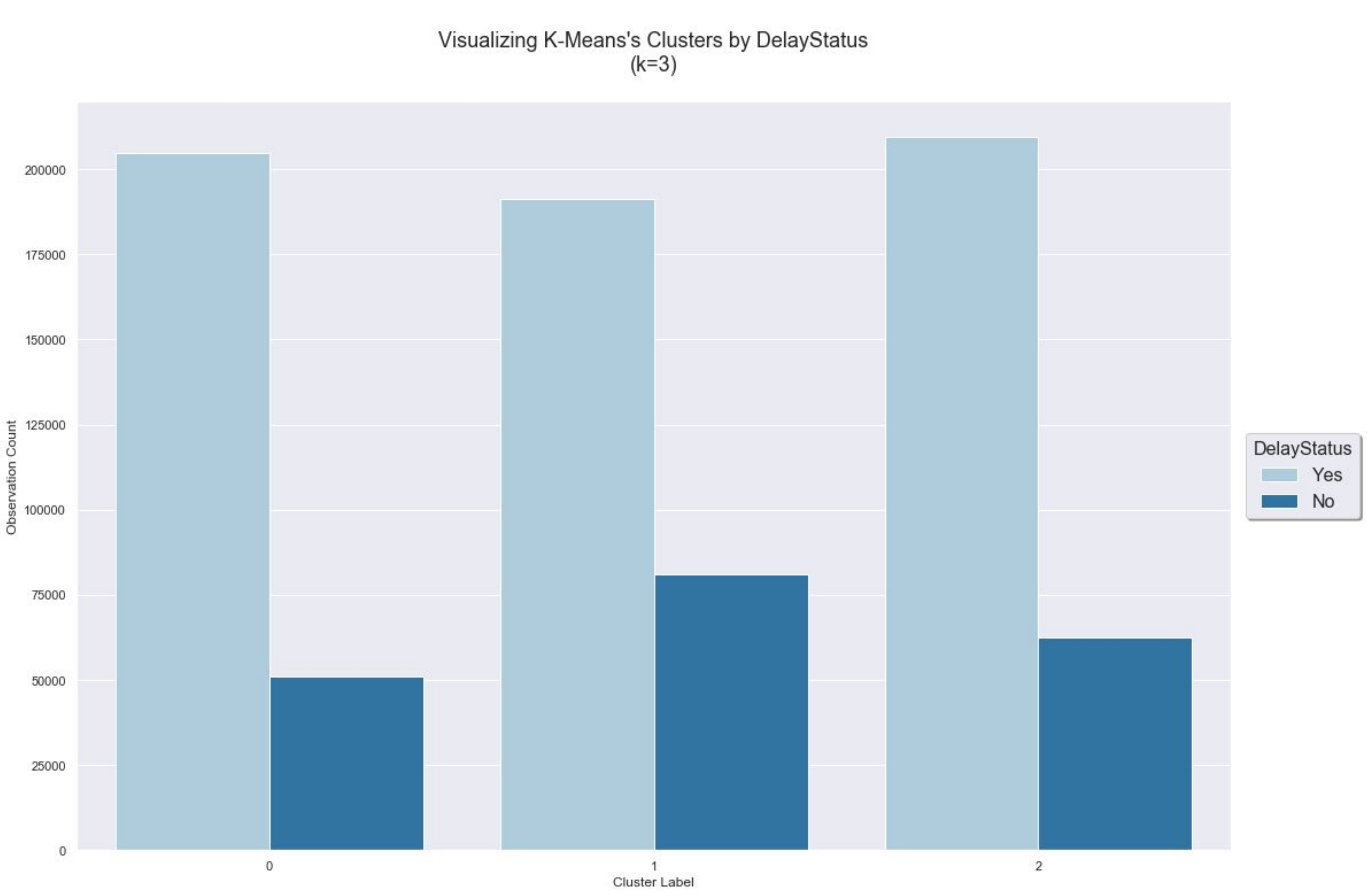
Highest (best) score

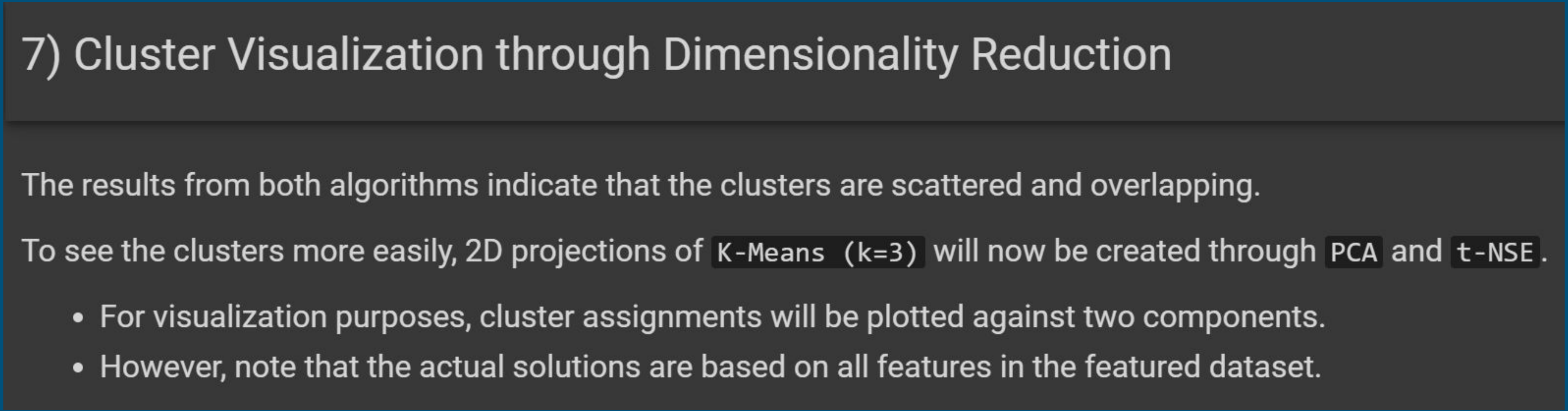


```
Clustering with KMeans, k=4
Silhouette score: 0.47701857156323946


Clustering with KMeans, k=5
Silhouette score: 0.46472014780559484


Clustering with KMeans, k=6
Silhouette score: 0.45185346668607007
```



Clusters by K-Means → `k=3`


Visualizing K-Means's Clusters by DelayStatus
(k=3)
Observation Count
0
25000
50000
75000
100000
125000
150000
175000
200000
Cluster Label
0
1
2
DelayStatus
Yes
No

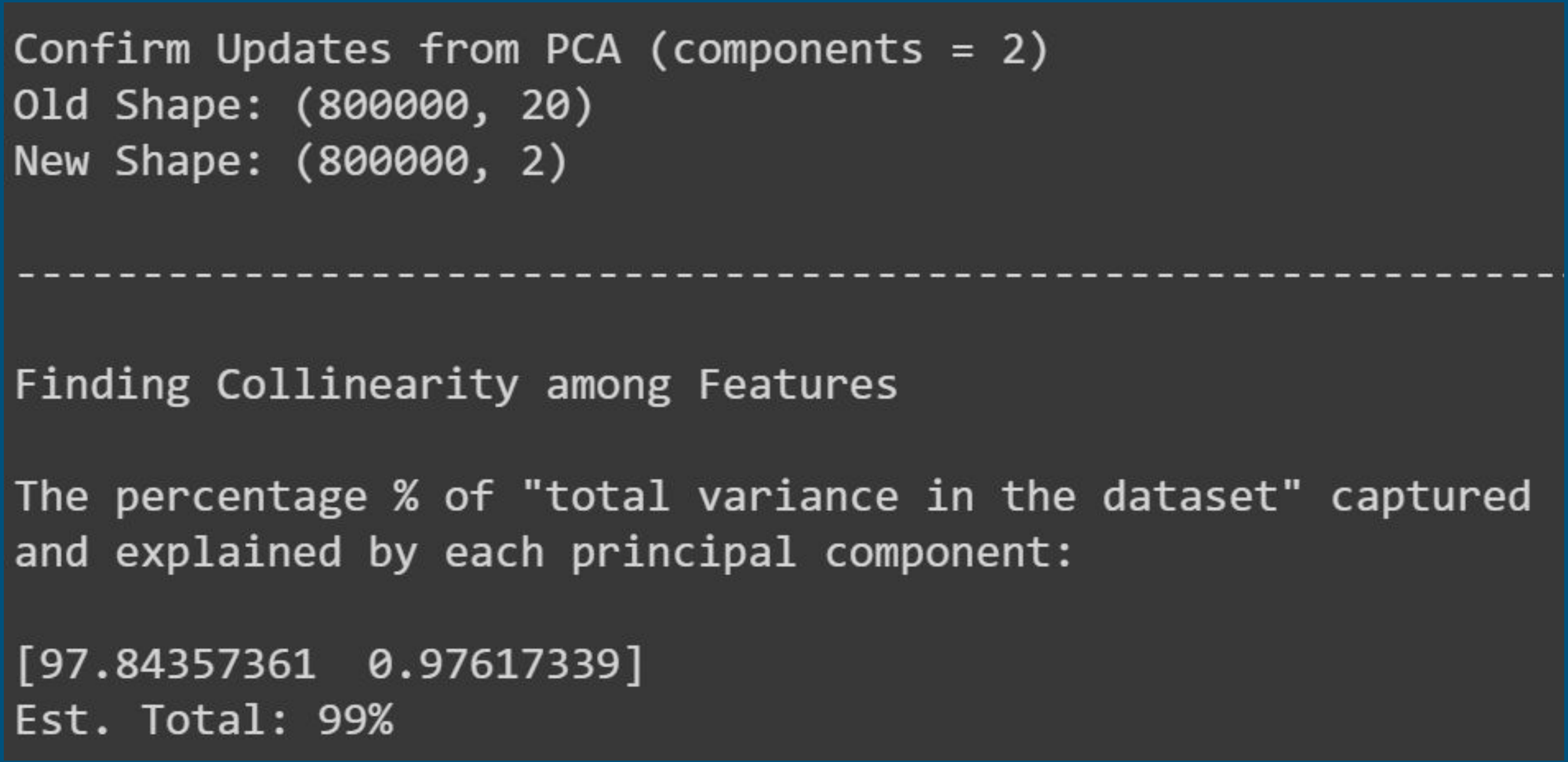


7) Cluster Visualization through Dimensionality Reduction

The results from both algorithms indicate that the clusters are scattered and overlapping.

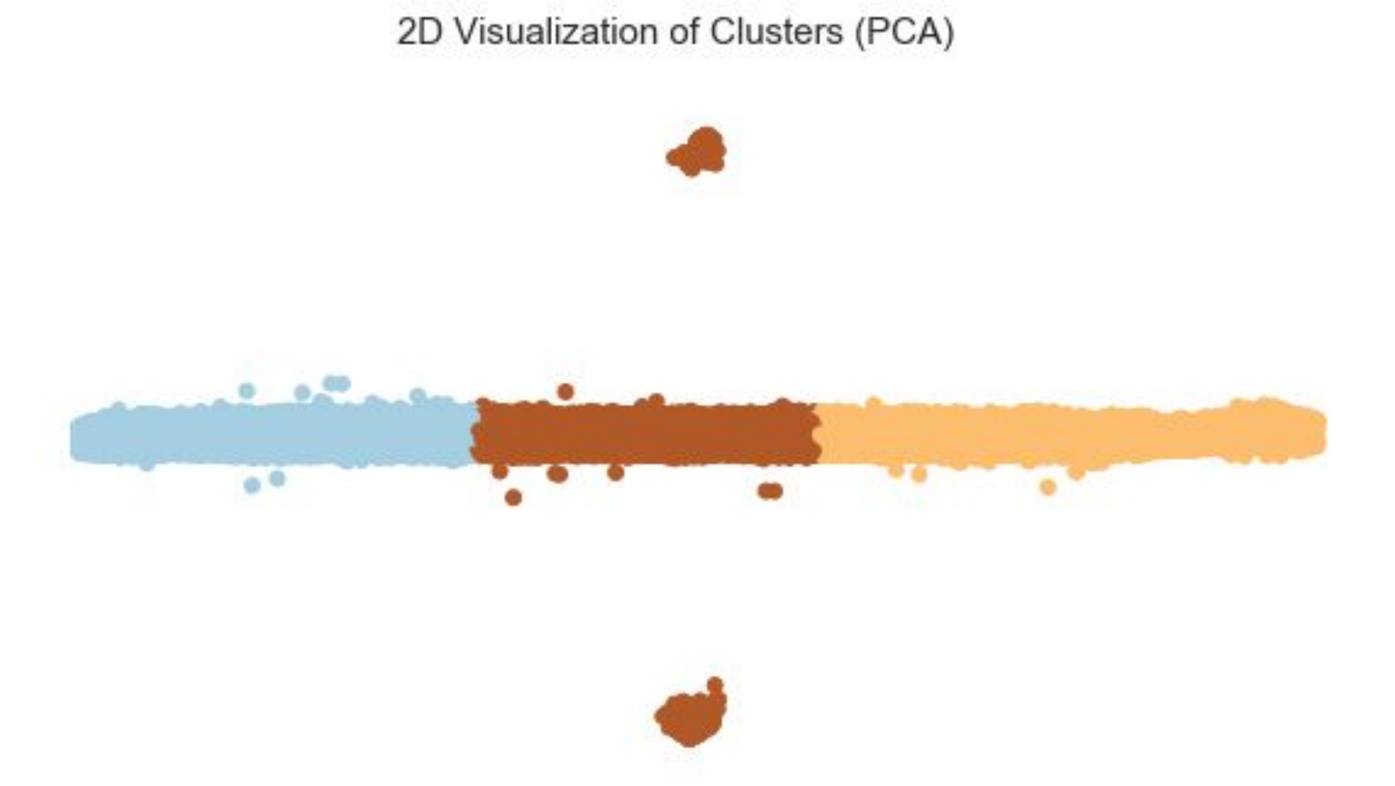
To see the clusters more easily, 2D projections of `K-Means (k=3)` will now be created through `PCA` and `t-NSE`.

- For visualization purposes, cluster assignments will be plotted against two components.
- However, note that the actual solutions are based on all features in the featured dataset.

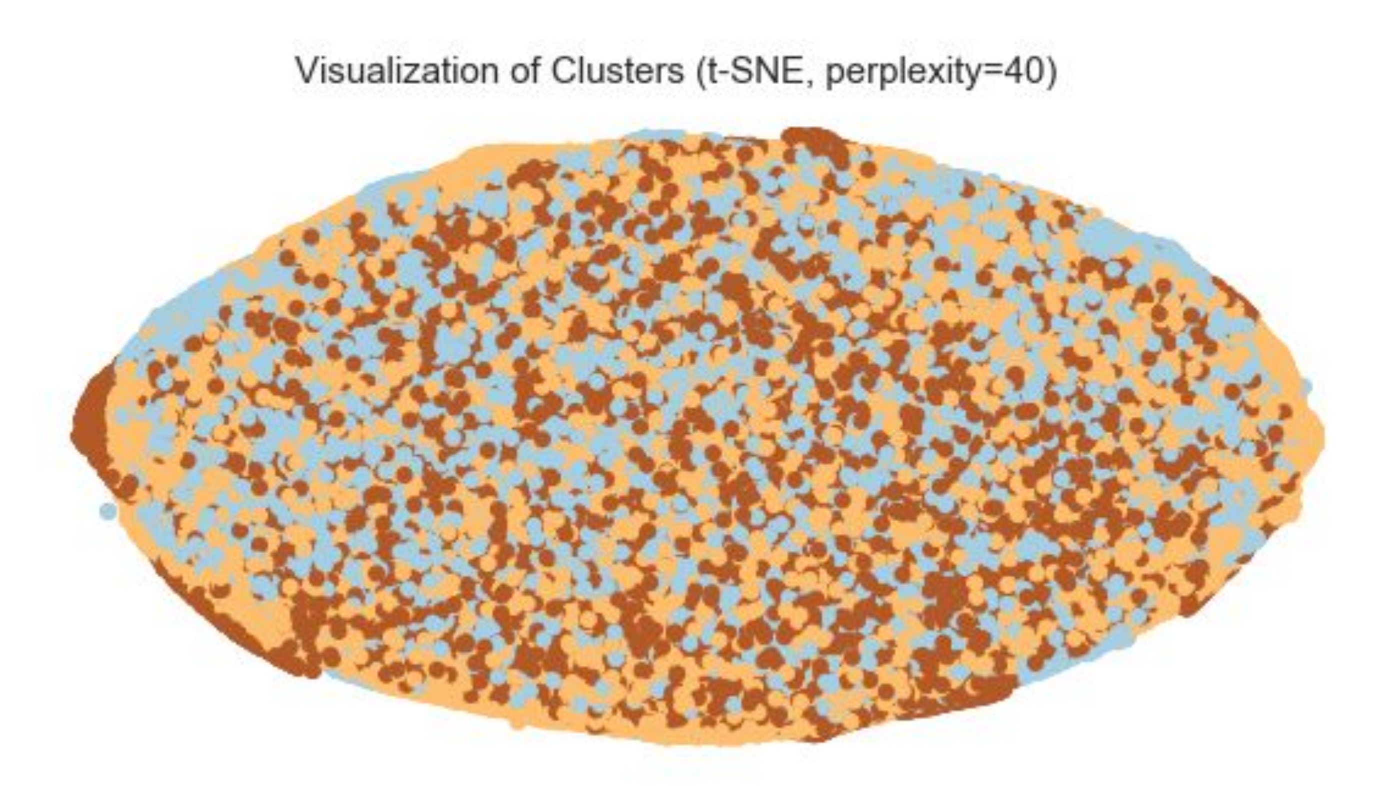


```
Confirm Updates from PCA (components = 2)
Old Shape: (800000, 20)
New Shape: (800000, 2)


------------------------------------------------------------


Finding Collinearity among Features

The percentage % of "total variance in the dataset" captured
and explained by each principal component:

[97.84357361  0.97617339]
Est. Total: 99%
```

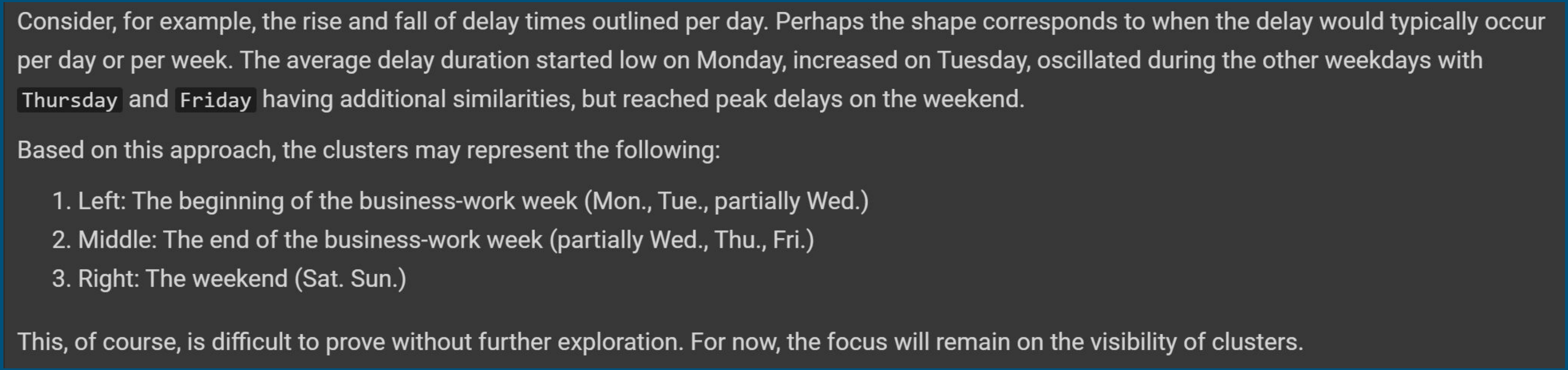
Cluster Visualization through Dimensionality Reduction → PCA



Cluster Visualization through Dimensionality Reduction → PCA



Cluster Visualization through Dimensionality Reduction → t-SNE (perplexity=40)



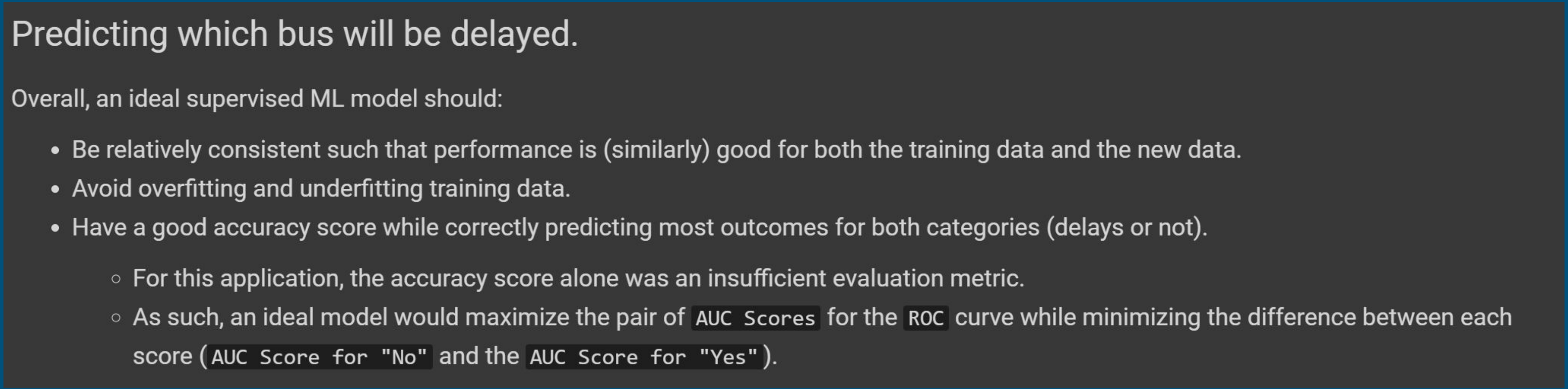
Consider, for example, the rise and fall of delay times outlined per day. Perhaps the shape corresponds to when the delay would typically occur per day or per week. The average delay duration started low on Monday, increased on Tuesday, oscillated during the other weekdays with `Thursday` and `Friday` having additional similarities, but reached peak delays on the weekend.

Based on this approach, the clusters may represent the following:

1. Left: The beginning of the business-work week (Mon., Tue., partially Wed.)
2. Middle: The end of the business-work week (partially Wed., Thu., Fri.)
3. Right: The weekend (Sat. Sun.)

This, of course, is difficult to prove without further exploration. For now, the focus will remain on the visibility of clusters.

Analysis Summary for PCA and t-SNE

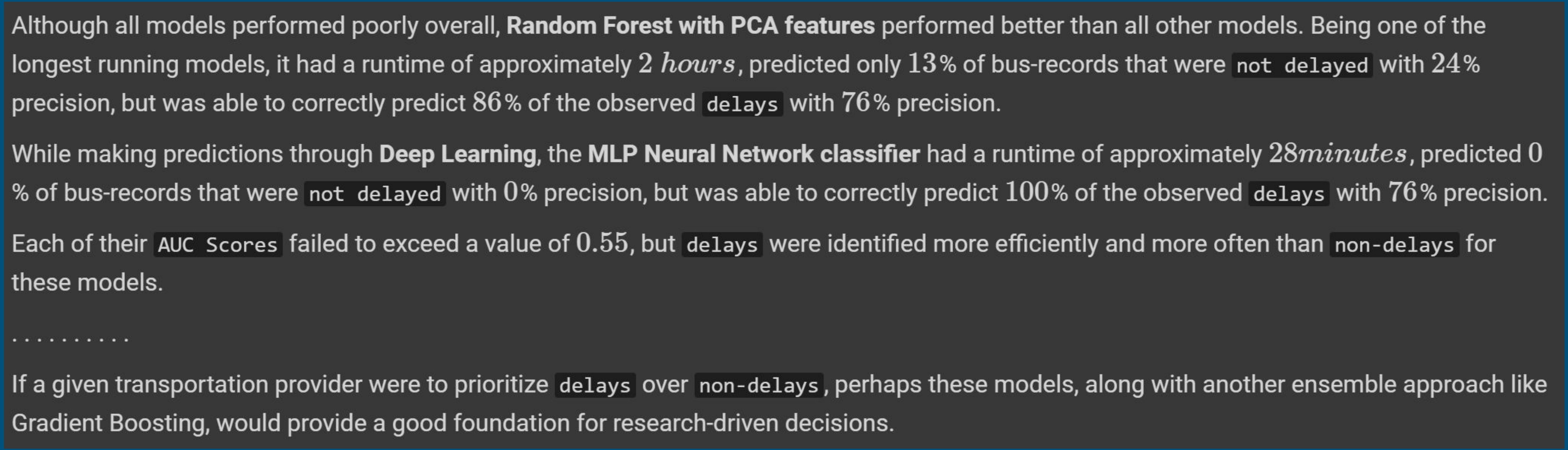


Predicting which bus will be delayed.

Overall, an ideal supervised ML model should:

- Be relatively consistent such that performance is (similarly) good for both the training data and the new data.
- Avoid overfitting and underfitting training data.
- Have a good accuracy score while correctly predicting most outcomes for both categories (delays or not).
    - For this application, the accuracy score alone was an insufficient evaluation metric.
    - As such, an ideal model would maximize the pair of `AUC Scores` for the `ROC` curve while minimizing the difference between each score (`AUC Score for "No"` and the `AUC Score for "Yes"`).

Other Remarks



Although all models performed poorly overall, **Random Forest with PCA features** performed better than all other models. Being one of the longest running models, it had a runtime of approximately $2\ hours$, predicted only 13% of bus-records that were `not delayed` with 24% precision, but was able to correctly predict 86% of the observed `delays` with 76% precision.

While making predictions through **Deep Learning**, the **MLP Neural Network classifier** had a runtime of approximately $28minutes$, predicted 0 % of bus-records that were `not delayed` with 0% precision, but was able to correctly predict 100% of the observed `delays` with 76% precision.

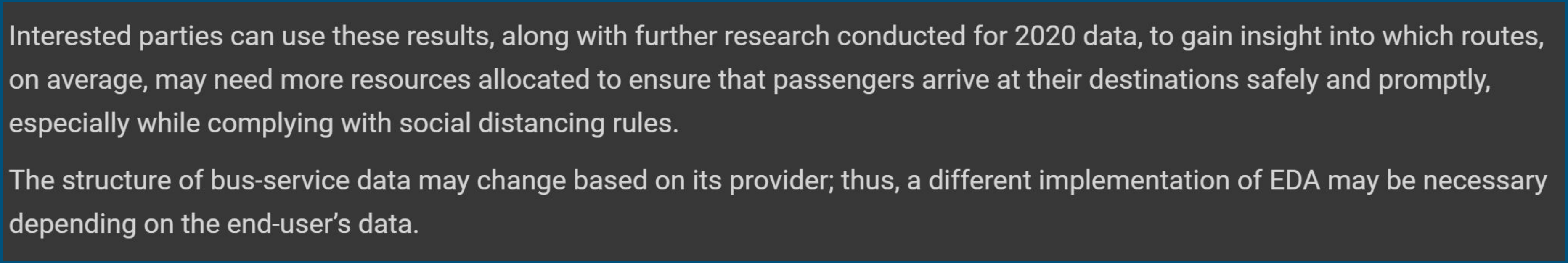
Each of their `AUC Scores` failed to exceed a value of 0.55, but `delays` were identified more efficiently and more often than `non-delays` for these models.

. . . .

If a given transportation provider were to prioritize `delays` over `non-delays`, perhaps these models, along with another ensemble approach like Gradient Boosting, would provide a good foundation for research-driven decisions.

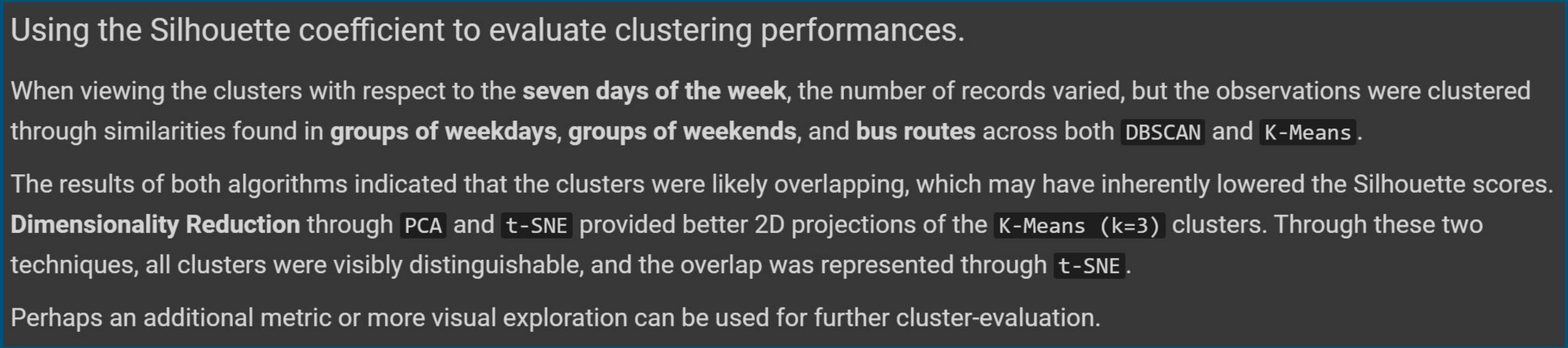
Other Remarks

Other Remarks



Interested parties can use these results, along with further research conducted for 2020 data, to gain insight into which routes, on average, may need more resources allocated to ensure that passengers arrive at their destinations safely and promptly, especially while complying with social distancing rules.

The structure of bus-service data may change based on its provider; thus, a different implementation of EDA may be necessary depending on the end-user’s data.



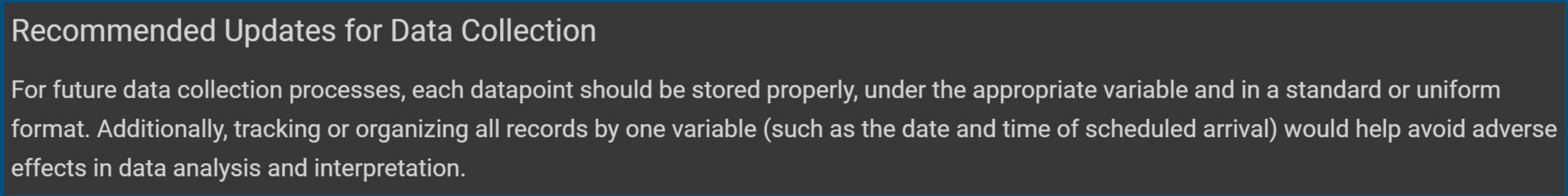
Using the Silhouette coefficient to evaluate clustering performances.

When viewing the clusters with respect to the **seven days of the week**, the number of records varied, but the observations were clustered through similarities found in **groups of weekdays**, **groups of weekends**, and **bus routes** across both `DBSCAN` and `K-Means`.

The results of both algorithms indicated that the clusters were likely overlapping, which may have inherently lowered the Silhouette scores. **Dimensionality Reduction** through `PCA` and `t-SNE` provided better 2D projections of the `K-Means (k=3)` clusters. Through these two techniques, all clusters were visibly distinguishable, and the overlap was represented through `t-SNE`.

Perhaps an additional metric or more visual exploration can be used for further cluster-evaluation.

Other Considerations



Recommended Updates for Data Collection

For future data collection processes, each datapoint should be stored properly, under the appropriate variable and in a standard or uniform format. Additionally, tracking or organizing all records by one variable (such as the date and time of scheduled arrival) would help avoid adverse effects in data analysis and interpretation.

Other Considerations

Sources

1. Kaggle Datasets
   a. https://www.kaggle.com/stoney71/new-york-city-transport-statistics
2. Decorative Pictures on Title Slide
   a. https://pixabay.com/