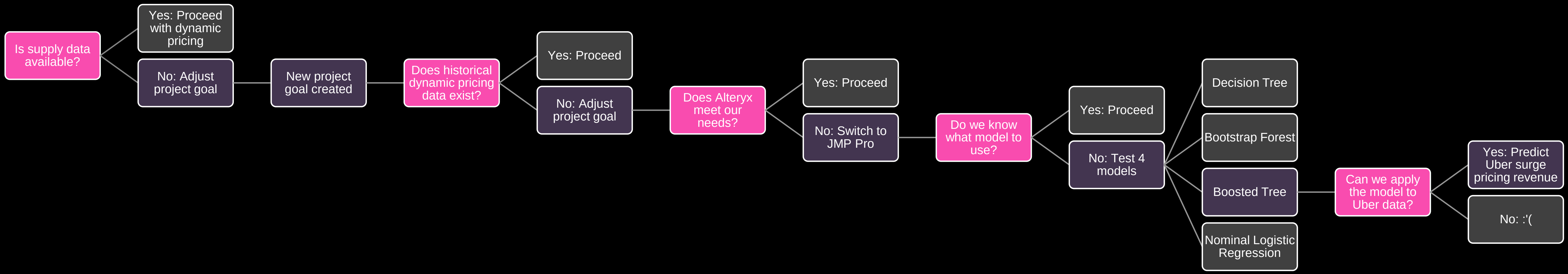
Rideshare: Dynamic Pricing Project: Final Results

MKT 6V99.501: AI IN MARKETING
PRESENTED BY: FALGUNI DAS

Problem Definition

We want to help rideshare companies utilize historical data to predict when surge pricing can be used to maximize revenue and ensure current demand is met with quality, efficient service.

Our Journey: A Tree of Failures & Learnings



Data Preparation-Alteryx

1

Data Cleaning

- Renamed column headers for clarity
- Removed taxi data to keep the focus on rideshare
- Created time of day columns
- Designated base price vs. surge price

2

Researching Events & Neighborhoods

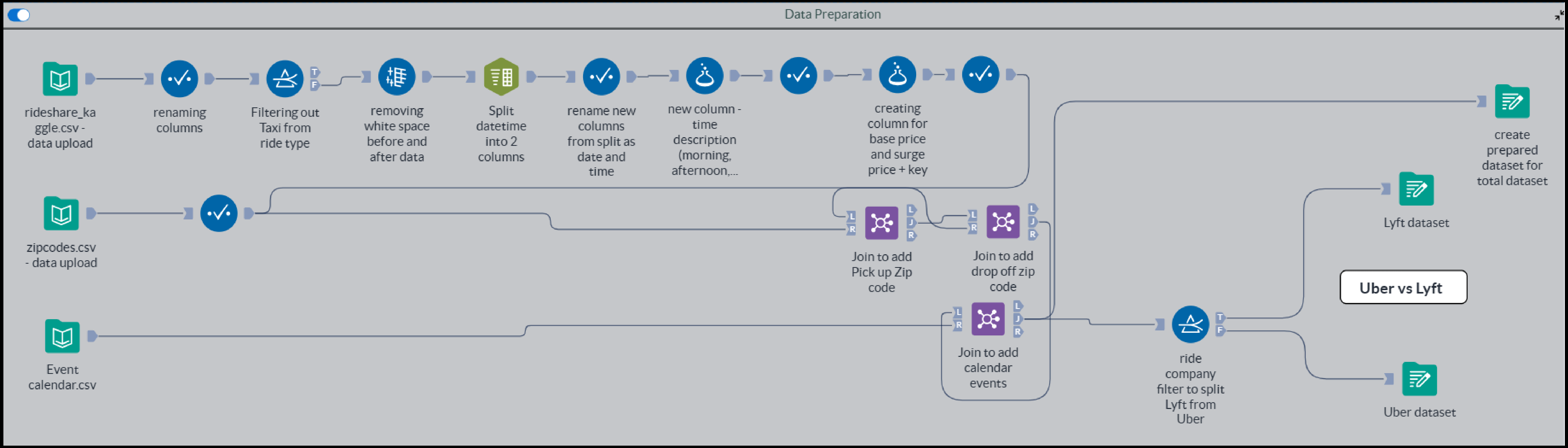
- Established criteria for events
- Researched events and zip codes via government websites & XYZ
- Cleaned data and created keys to join into datasheet

3

Joining 3 Datasheets

- CSV with original datasheet from Kaggle
- CSV of zip codes and neighborhoods
- CSV of largescale events with keys

Alteryx Workflow for Data Preparation



Lyft Model Preparation –JMP Pro

1 Data Type Identification

- Identified 53 Continuous Variables
- Identified 1 Ordinal Variable
- Identified 14 Nominal Variables

2 Variable Validation

- Checked for Multicollinearity
  - removed 43 continuous variables
- Removed 7 nominal variables due to data duplication
- Added 1 Boolean Variable
  - Surge (Y=1, N=0)

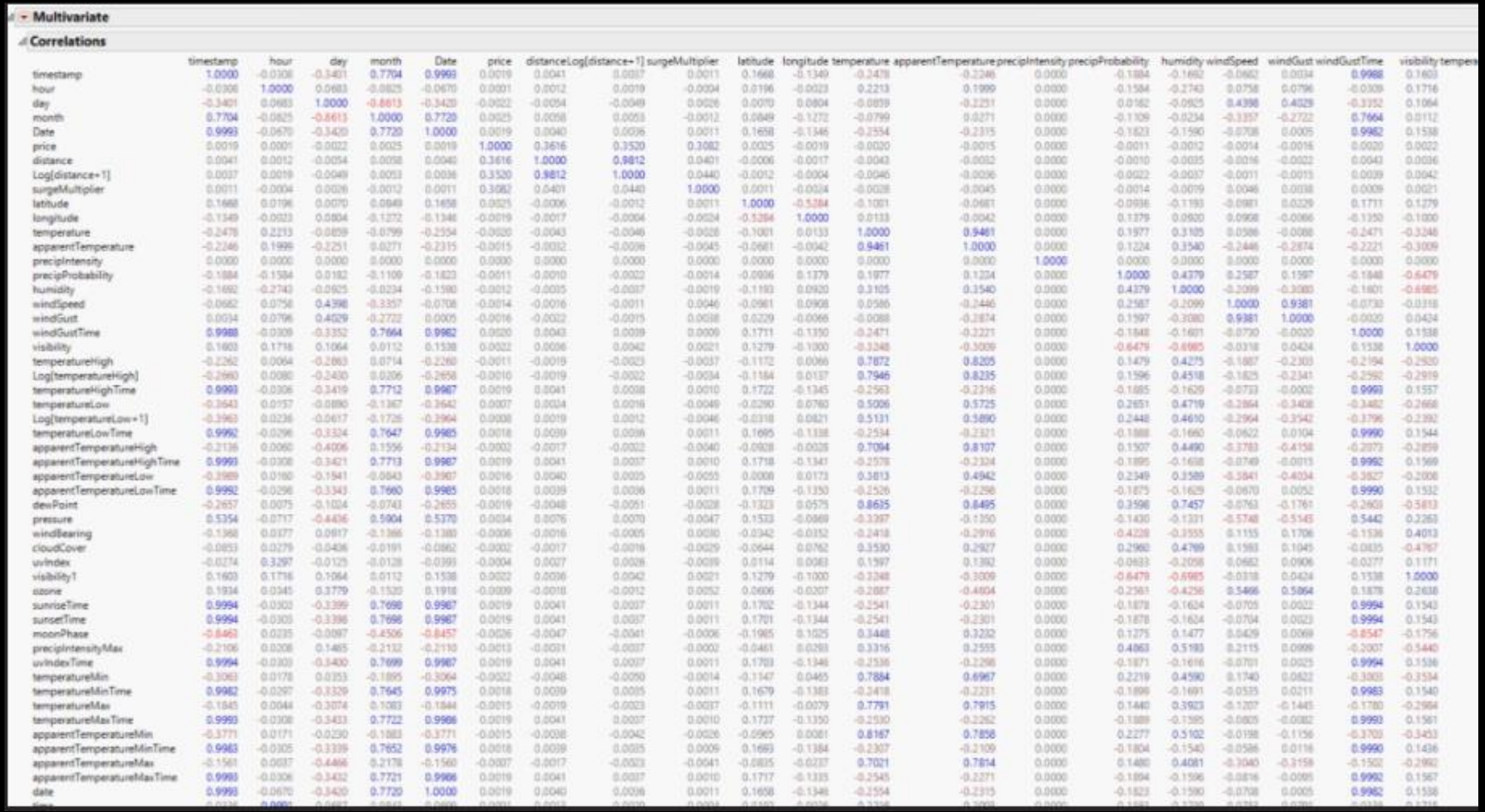
3 Data Transformation

- Checked for outliers
- Confirmed normal distribution
  Transformed skewed variables
  - Log
  - Log + 1

4 Training Dataset

- Split the data into 3 sets
  - 50% Training Set
  - 30% Validation Set
  - 20% Test Set

JMP Pro Model Preparation - Multicollinearity

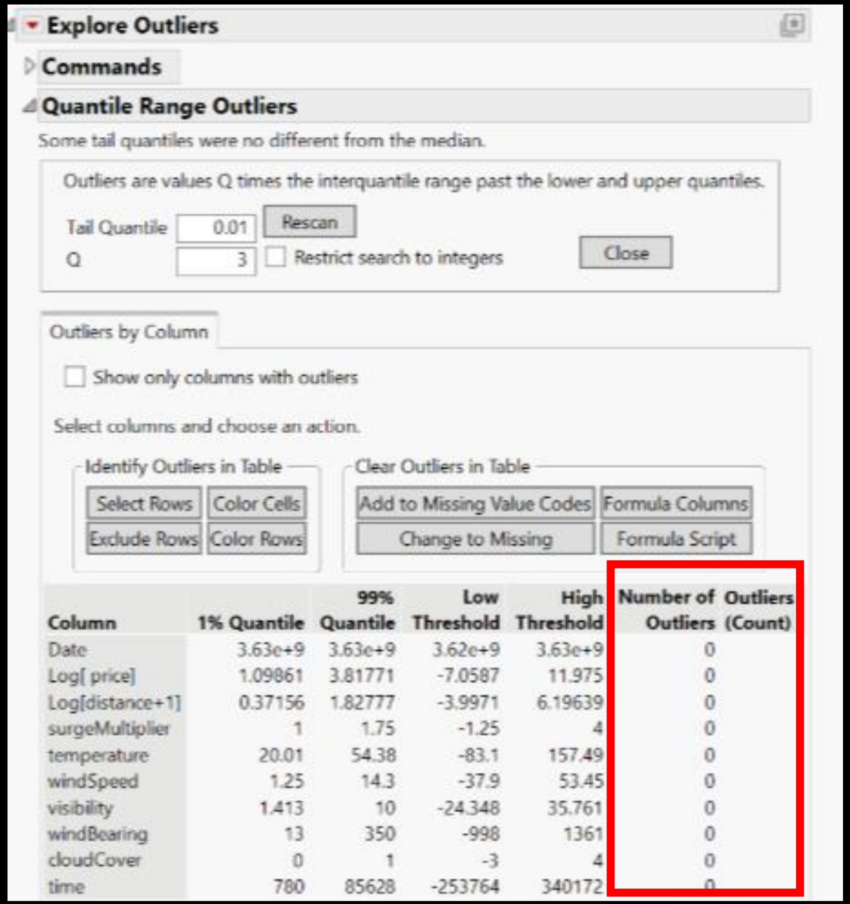


Correlations

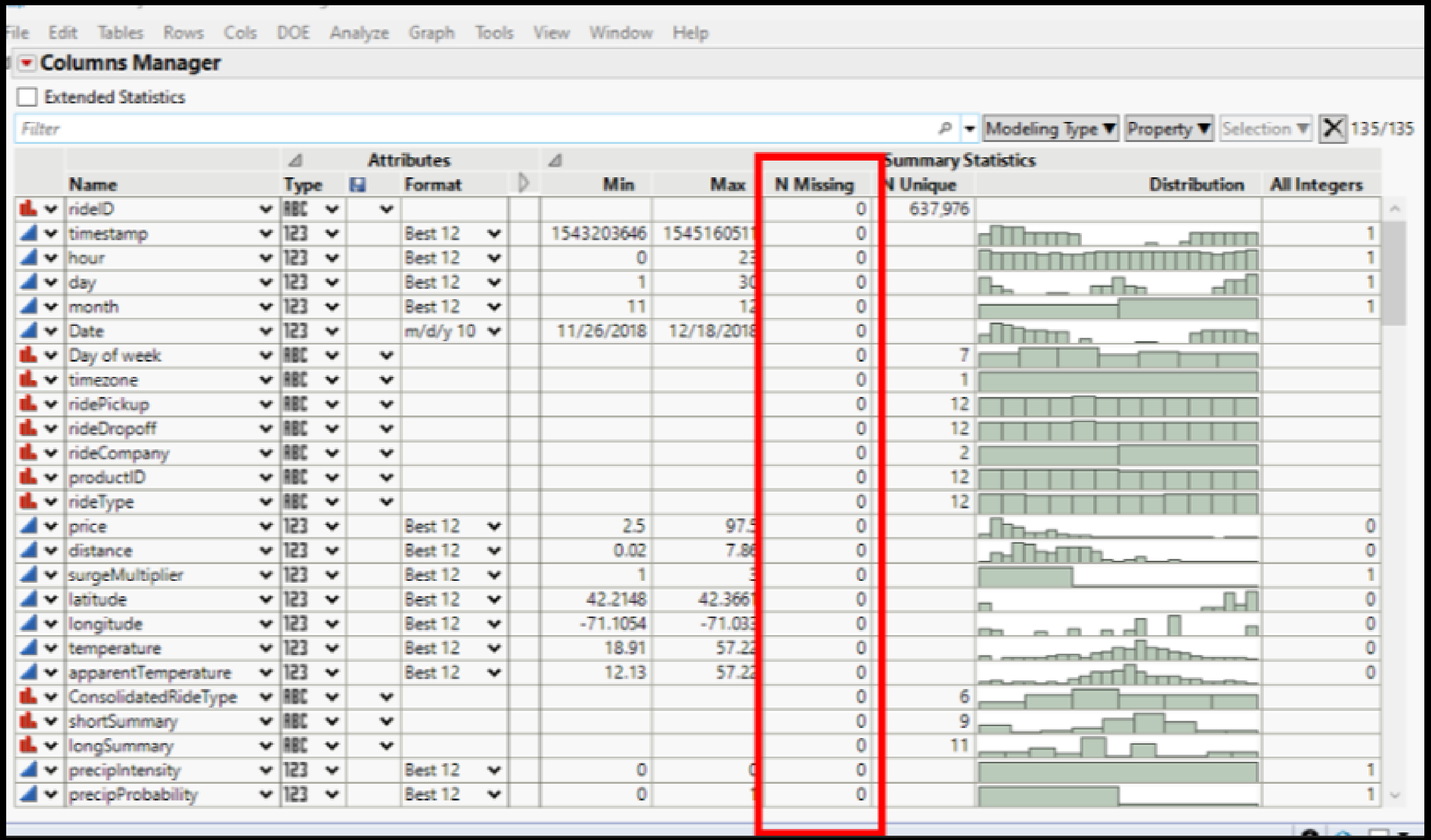
	Date	price	Log[distance+1]	surgeMultiplier	temperature	windSpeed	visibility	windBearing	cloudCover	uvIndex	time
Date	1.0000	0.0019	0.0036	0.0011	-0.2554	-0.0708	0.1538	-0.1380	-0.0862	-0.0393	-0.0699
price	0.0019	1.0000	0.3520	0.3082	-0.0020	-0.0014	0.0022	-0.0006	-0.0002	-0.0004	0.0001
Log[distance+1]	0.0036	0.3520	1.0000	0.0440	-0.0046	-0.0011	0.0042	-0.0005	-0.0016	0.0026	0.0020
surgeMultiplier	0.0011	0.3082	0.0440	1.0000	-0.0028	0.0046	0.0021	0.0030	-0.0029	-0.0039	-0.0004
temperature	-0.2554	-0.0020	-0.0046	-0.0028	1.0000	0.0586	-0.3248	-0.2418	0.3530	0.1597	0.2216
windSpeed	-0.0708	-0.0014	-0.0011	0.0046	0.0586	1.0000	-0.0318	0.1155	0.1593	0.0682	0.0753
visibility	0.1538	0.0022	0.0042	0.0021	-0.3248	-0.0318	1.0000	0.4013	-0.4767	0.1171	0.1715
windBearing	-0.1380	-0.0006	-0.0005	0.0030	-0.2418	0.1155	0.4013	1.0000	-0.2178	0.1150	0.0385
cloudCover	-0.0862	-0.0002	-0.0016	-0.0029	0.3530	0.1593	-0.4767	-0.2178	1.0000	-0.0656	0.0277
uvIndex	-0.0393	-0.0004	0.0026	-0.0039	0.1597	0.0682	0.1171	0.1150	-0.0656	1.0000	0.3292
time	-0.0699	0.0001	0.0020	-0.0004	0.2216	0.0753	0.1715	0.0385	0.0277	0.3292	1.0000

- Checking for correlated and inversely correlated variables and removed variables with values above 0.5000
- Removed 43 continuous variables

JMP Pro Model Preparation – Outliers & Missing Da



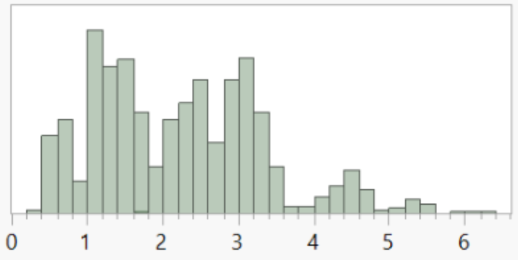
- Of the remaining variables there are no outliers



- Additionally, there are no missing data points in any of the variables

JMP Pro Model Preparation - Transformations

distance



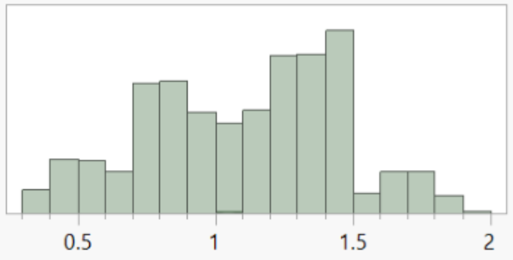
Quantiles

100.0%	maximum	6.33
99.5%		5.37
97.5%		4.6
90.0%		3.41
75.0%	quartile	2.97
50.0%	median	2.14
25.0%	quartile	1.27
10.0%		0.91
2.5%		0.55
0.5%		0.43
0.0%	minimum	0.39

Summary Statistics

Mean	2.1869756
Std Dev	1.0866217
Std Err Mean	0.0019598
Upper 95% Mean	2.1908168
Lower 95% Mean	2.1831344
N	307408
N Missing	0

Log[distance+1]



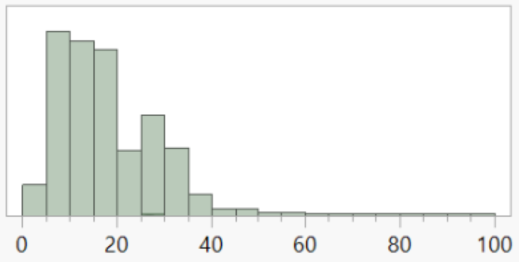
Quantiles

100.0%	maximum	1.9919755
99.5%		1.8515995
97.5%		1.7227666
90.0%		1.4838747
75.0%	quartile	1.3787661
50.0%	median	1.1442228
25.0%	quartile	0.8197798
10.0%		0.6471032
2.5%		0.4382549
0.5%		0.3576744
0.0%	minimum	0.3293037

Summary Statistics

Mean	1.100034
Std Dev	0.3476817
Std Err Mean	0.0006271
Upper 95% Mean	1.1012631
Lower 95% Mean	1.0988049
N	307408
N Missing	0

price



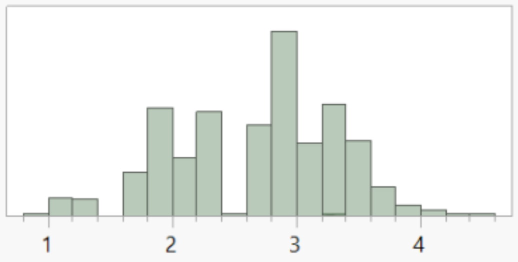
Quantiles

100.0%	maximum	97.5
99.5%		52.5
97.5%		38.5
90.0%		32.5
75.0%	quartile	22.5
50.0%	median	16.5
25.0%	quartile	9
10.0%		7
2.5%		3.5
0.5%		3
0.0%	minimum	2.5

Summary Statistics

Mean	17.351396
Std Dev	10.019171
Std Err Mean	0.0180707
Upper 95% Mean	17.386814
Lower 95% Mean	17.315978
N	307408
N Missing	0

Log[price]



Quantiles

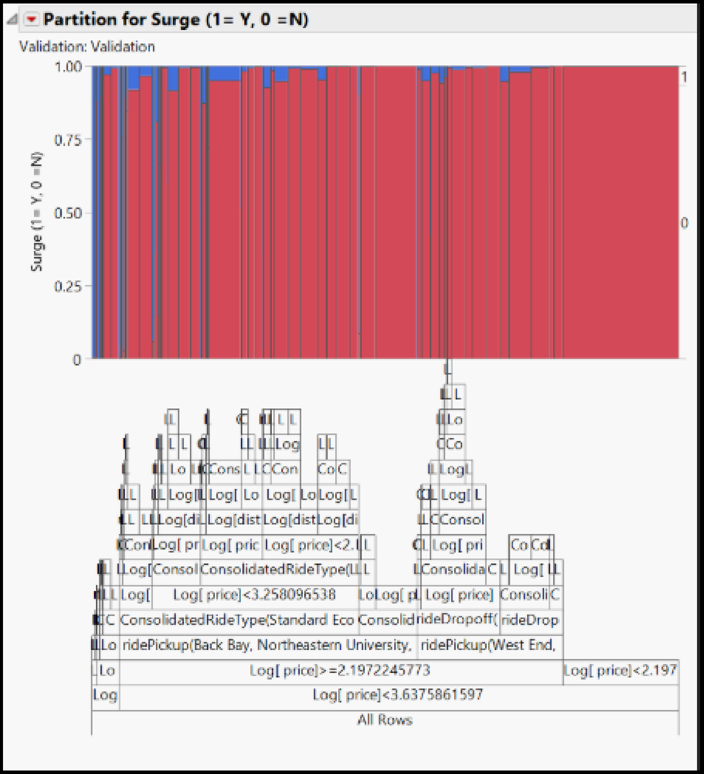
100.0%	maximum	4.5798524
99.5%		3.9608132
97.5%		3.6506582
90.0%		3.4812401
75.0%	quartile	3.1135153
50.0%	median	2.8033604
25.0%	quartile	2.1972246
10.0%		1.9459101
2.5%		1.252763
0.5%		1.0986123
0.0%	minimum	0.9162907

Summary Statistics

Mean	2.675201
Std Dev	0.6259745
Std Err Mean	0.001129
Upper 95% Mean	2.6774139
Lower 95% Mean	2.6729882
N	307408
N Missing	0

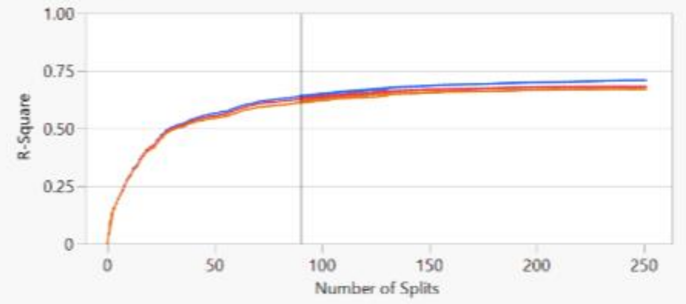
- Transformed Distance and Price because they were showing right skewness
  - Distance – Log +1 transformation
  - Price – Log transformation

Model 1 - Decision Tree



	RSquare	N	Number of Splits
Training	0.642	153704	90
Validation	0.632	92222	
Test	0.618	61482	

Split History



Validation Data in Red
Test Data in Orange

Fit Details

Measure	Training	Validation	Test	Definition
Entropy RSquare	0.6422	0.6319	0.6180	1-Loglike(model)/Loglike(0)
Generalized RSquare	0.6975	0.6886	0.6757	(1-(L(0)/L(model))^(2/n))/(1-L(0)^(2/n))
Mean -Log p	0.0886	0.0922	0.0958	∑ -Log(ρ[j])/n
RASE	0.1492	0.1521	0.1550	$\sqrt{\sum(y[j]-\rho[j])^2/n}$
Mean Abs Dev	0.0446	0.0457	0.0466	∑ |y[j]-ρ[j]|/n
Misclassification Rate	0.0259	0.0266	0.0278	∑ (ρ[j]≠ρMax)/n
N	153704	92222	61482	n

Confusion Matrix

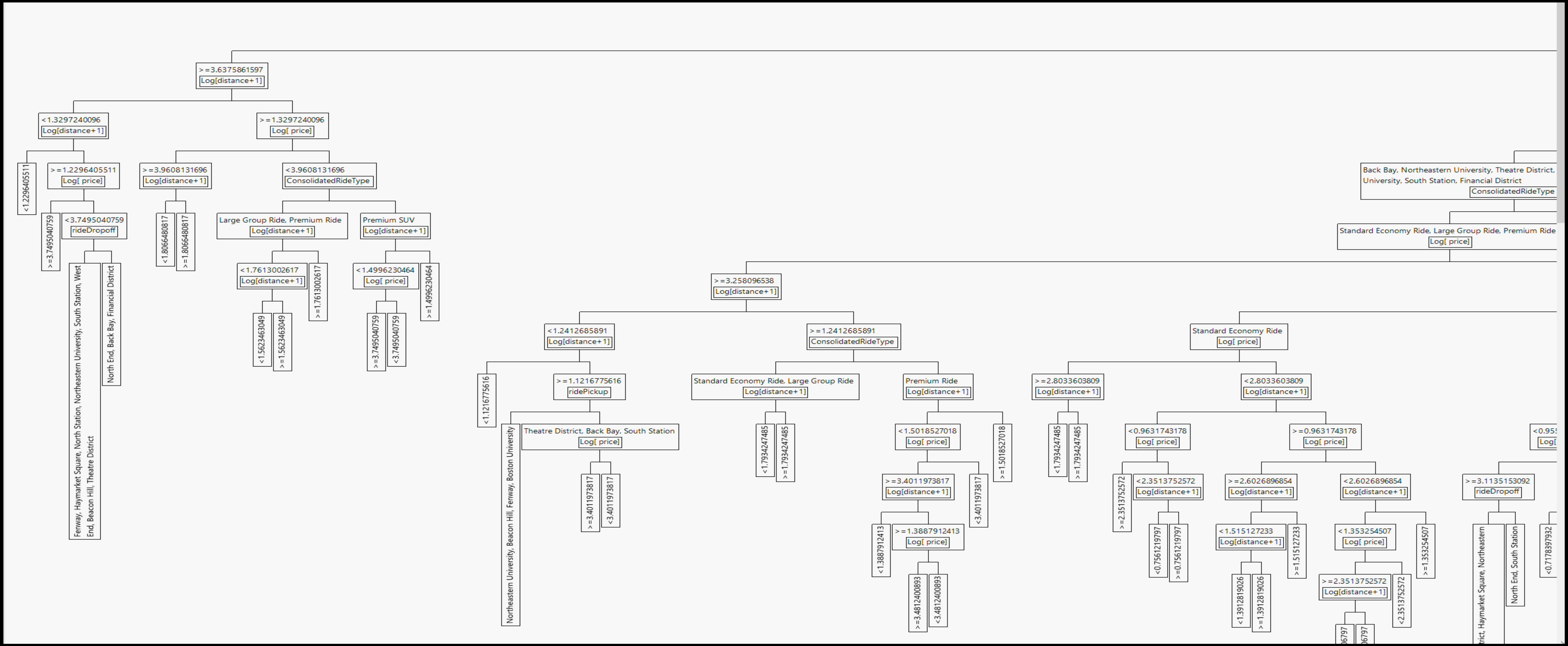
Actual Surge (1= Y, 0 =N)	Training Predicted Count 0	Training Predicted Count 1	Actual Surge (1= Y, 0 =N)	Validation Predicted Count 0	Validation Predicted Count 1	Actual Surge (1= Y, 0 =N)	Test Predicted Count 0	Test Predicted Count 1
0	142546	757	0	85424	458	0	56921	327
1	3231	7170	1	1996	4344	1	1382	2852

Actual Surge (1= Y, 0 =N)	Training Predicted Rate 0	Training Predicted Rate 1	Actual Surge (1= Y, 0 =N)	Validation Predicted Rate 0	Validation Predicted Rate 1	Actual Surge (1= Y, 0 =N)	Test Predicted Rate 0	Test Predicted Rate 1
0	0.995	0.005	0	0.995	0.005	0	0.994	0.006
1	0.311	0.689	1	0.315	0.685	1	0.326	0.674

Model Results (Test data):

- Model Type = Decision Tree – 90 splits or nodes
- Entropy R-Squared = 0.6180
- Misclassification rate = 0.0278
- Confusion Matrix = correctly predicted 56921 “No” values + 2852 “Yes” values = 59773 correct predictions = 0.9722

Model 1 - Decision Tree Video



Model 2 - Bootstrap Forest

Bootstrap Forest for Surge (1= Y, 0 =N)

Specifications

Target	Surge (1= Y, 0 =N)	Training Rows:	153704
Validation Column:	Validation	Validation Rows:	92222
		Test Rows:	61482
Number of Trees in the Forest:	500	Number of Terms:	18
Number of Terms Sampled per Split:	11	Bootstrap Samples:	153704
		Minimum Splits per Tree:	10
		Minimum Size Split:	307

Overall Statistics

Measure	Training	Validation	Test	Definition
Entropy RSquare	0.5827	0.5701	0.5634	1-Loglike(model)/Loglike(0)
Generalized RSquare	0.6418	0.6305	0.6241	(1-(L(0)/L(model))^(2/n))/(1-L(0)^(2/n))
Mean -Log p	0.1033	0.1076	0.1094	∑ -Log(ρ[j])/n
RASE	0.1665	0.1697	0.1718	$\sqrt{\sum(y[j]-\rho[j])^2/n}$
Mean Abs Dev	0.0678	0.0692	0.0696	∑ |y[j]-ρ[j]|/n
Misclassification Rate	0.0345	0.0357	0.0365	∑ (ρ[j]≠ρMax)/n
N	153704	92222	61482	n

Number of Trees
93

Confusion Matrix

Training

Actual Surge (1= Y, 0 =N)	Predicted Count 0	1
0	142946	357
1	4941	5460

Validation

Actual Surge (1= Y, 0 =N)	Predicted Count 0	1
0	85665	217
1	3076	3264

Test

Actual Surge (1= Y, 0 =N)	Predicted Count 0	1
0	57110	138
1	2109	2125

Training

Actual Surge (1= Y, 0 =N)	Predicted Rate 0	1
0	0.998	0.002
1	0.475	0.525

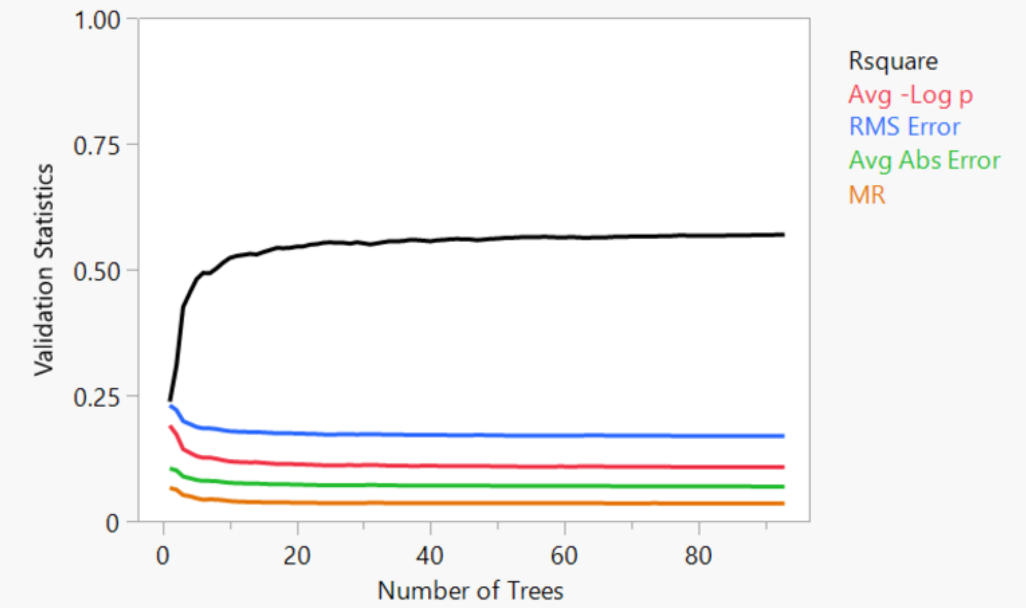
Validation

Actual Surge (1= Y, 0 =N)	Predicted Rate 0	1
0	0.997	0.003
1	0.485	0.515

Test

Actual Surge (1= Y, 0 =N)	Predicted Rate 0	1
0	0.998	0.002
1	0.498	0.502

Cumulative Validation



Model Results (Test data):

- Model Type = Bootstrap Forest – 93 trees
- Entropy R-Squared = 0.5634
- Misclassification rate = 0.0365
- Confusion Matrix = correctly predicted 57110 “No” values + 2125 “Yes” values = 59235 correct predictions = 0.9635

Model 3 – Boosted Tree

Boosted Tree for Surge (1= Y, 0 =N)

Specifications

Target	Surge (1= Y, 0 =N)	Number of training rows:	153704
Validation Column:	Validation	Number of validation rows:	92222
Number of Layers:	159	Number of test rows:	61482
Splits per Tree:	20		
Learning Rate:	0.199		
Overfit Penalty:	0.0001		

Overall Statistics

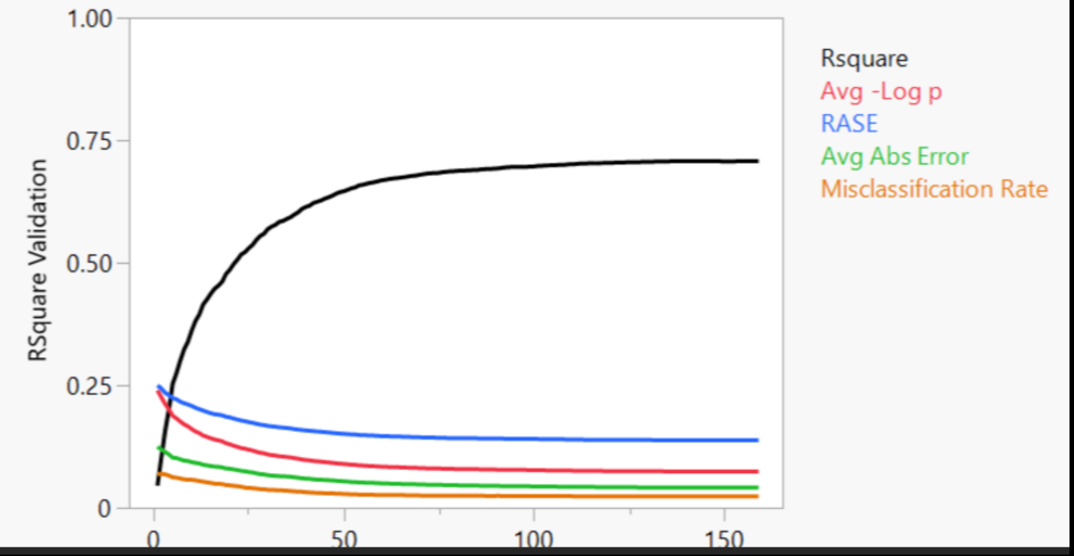
Measure	Training	Validation	Test	Definition
Entropy RSquare	0.7412	0.7063	0.7008	1-Loglike(model)/Loglike(0)
Generalized RSquare	0.7866	0.7562	0.7514	$(1-(L(0)/L(model))^{(2/n)})/(1-L(0)^{(2/n)})$
Mean -Log p	0.0641	0.0735	0.0750	$\sum -\mathrm{Log}(\rho[j])/n$
RASE	0.1272	0.1374	0.1389	$\sqrt{\sum(y[j]-\rho[j])^2/n}$
Mean Abs Dev	0.0373	0.0405	0.0409	$\sum \lvert y[j]-\rho[j]\rvert/n$
Misclassification Rate	0.0191	0.0226	0.0228	$\sum (\rho[j]\neq\rho Max)/n$
N	153704	92222	61482	n

Confusion Matrix

Actual Surge (1= Y, 0 =N)	Training Predicted Count 0	Training Predicted Count 1	Validation Predicted Count 0	Validation Predicted Count 1	Test Predicted Count 0	Test Predicted Count 1
0	143073	230	85621	261	57076	172
1	2701	7700	1827	4513	1227	3007

Actual Surge (1= Y, 0 =N)	Training Predicted Rate 0	Training Predicted Rate 1	Validation Predicted Rate 0	Validation Predicted Rate 1	Test Predicted Rate 0	Test Predicted Rate 1
0	0.998	0.002	0.997	0.003	0.997	0.003
1	0.260	0.740	0.288	0.712	0.290	0.710

Cumulative Validation



Model Results (Test data):

- Model Type = Boosted Tree – 159 layers
- Entropy R-Squared = 0.7008
- Misclassification rate = 0.0228
- Confusion Matrix = correctly predicted 57076 “No” values + 3007 “Yes” values = 60083 correct predictions = 0.9772

Model 4 - Regression - Nominal Logistic

Nominal Logistic Fit for Surge (1= Y, 0 =N)

Effect Summary

Source	Logworth	PValue
Log[price]	2590.960	0.00000
ConsolidatedRideType	2457.180	0.00000
Log[distance+1]	1888.654	0.00000
ridePickup	525.597	0.00000
rideDropoff	261.969	0.00000
cloudCover	3.331	0.00047
shortSummary	2.833	0.00147
windBearing	1.927	0.01182
Day of week	1.382	0.04147
uvIndex	1.018	0.09592
Date	0.804	0.15714
timeDescription	0.716	0.19246
windSpeed	0.429	0.37258
visibility	0.226	0.59461
EventLocation_Arrival	0.194	0.63945
time	0.186	0.65228
temperature	0.131	0.73883
EventDescription_Arrival	0.000	0.99956

Fit Details

Measure	Training	Validation	Test	Definition
Entropy RSquare	0.5898	0.5853	0.5720	1-Loglike(model)/Loglike(0)
Generalized RSquare	0.6486	0.6449	0.6323	(1-(L(0)/L(model))^(2/n))/(1-L(0)^(2/n))
Mean -Log p	0.1015	0.1038	0.1073	∑ -Log(ρ[j])/n
RASE	0.1564	0.1579	0.1610	$\sqrt{\sum(y[j]-\rho[j])^2/n}$
Mean Abs Dev	0.0524	0.0529	0.0539	∑ |y[j]-ρ[j]|/n
Misclassification Rate	0.0283	0.0287	0.0299	∑ (ρ[j]≠ρMax)/n
N	153704	92222	61482	n

Nominal Logistic Fit for Surge (1= Y, 0 =N)

Effect Summary

Source	Logworth	PValue
Log[price]	2593.833	0.00000
ConsolidatedRideType	2460.084	0.00000
Log[distance+1]	1890.921	0.00000
ridePickup	526.357	0.00000
rideDropoff	268.013	0.00000
cloudCover	3.855	0.00014
shortSummary	2.936	0.00116
Day of week	1.738	0.01826
windBearing	1.730	0.01861

Fit Details

Measure	Training	Validation	Test	Definition
Entropy RSquare	0.5896	0.5852	0.5725	1-Loglike(model)/Loglike(0)
Generalized RSquare	0.6483	0.6448	0.6328	(1-(L(0)/L(model))^(2/n))/(1-L(0)^(2/n))
Mean -Log p	0.1016	0.1039	0.1072	∑ -Log(ρ[j])/n
RASE	0.1564	0.1579	0.1608	$\sqrt{\sum(y[j]-\rho[j])^2/n}$
Mean Abs Dev	0.0524	0.0529	0.0539	∑ |y[j]-ρ[j]|/n
Misclassification Rate	0.0284	0.0286	0.0299	∑ (ρ[j]≠ρMax)/n
N	153704	92222	61482	n

Model Results (Test data):

- Model Type = Nominal Logistic – Regression
- Entropy R-Squared = 0.5725
- Misclassification rate = 0.0299
- Classification rate = 0.9701

Battle of the Models – Model Selection

Model Summary & Selection				
Model = Surge (1=Y, 0=N)	**Entropy R-Squared**	**Missclassification Rate**	**Classification Rate**	**# of Nodes/ Trees/Layers**
Nominal Logistic	0.5725	3.0%	97.0%	N/A
Decision Tree	0.6180	2.8%	97.2%	90
Bootstrap Forest	0.5634	3.7%	96.3%	93
Boosted Tree	0.7008	2.3%	97.7%	159

Selection Results (Test data):

- Model = Boosted Tree
  - Highest Entropy R-Squared
  - Lowest Misclassification rate
  - Highest Classification rate

Model Application - From Lyft to Uber

Surge Rate Distribution - Lyft		
Surge Rate	**Record Count by Rate**	**% Distribution**
1.00	663	4%
1.25	6150	39%
1.50	4303	27%
1.75	2362	15%
2.00	2239	14%
2.50	154	1%
3.00	12	0%
Total	**15883**	**100%**

Surge Rate Distribution - Uber		
Surge Rate	**Record Count by Rate**	**% Distribution**
1.00	478	4%
1.25	4435	39%
1.50	3103	27%
1.75	1703	15%
2.00	1615	14%
2.50	111	1%
3.00	9	0%
Total	**11454**	**100%**

Assumptions:

- Leveraging the Lyft model predicted records with surge pricing
  - Calculated the distribution of records by surge rate
  - Applied the distribution to the model predicted records for Uber

Model Application - Uber Surge Prediction

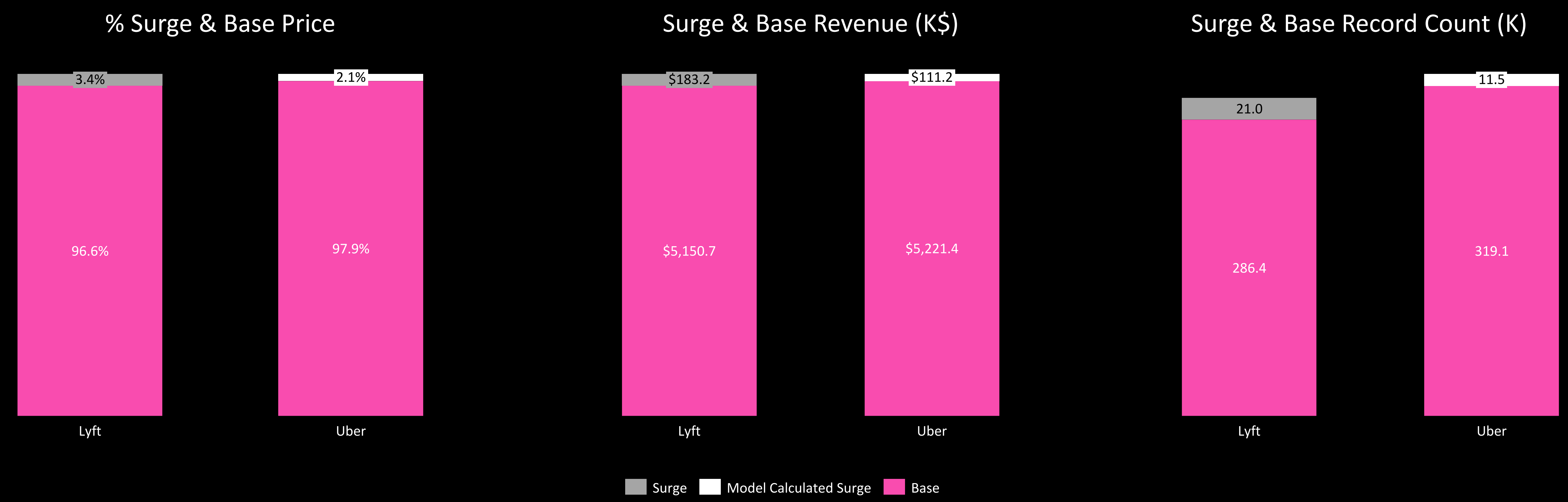
Adding surge pricing can increase Uber's revenue by $111k or ~2.1%

Surge Rate Distribution & Revenue Estimation - Uber					
Surge Rate	**Record Count by Rate**	**% Distribution**	**Without Surge**	**With Surge**	**Additional Revenue**
1.00	478	4%	$9,269	$9,269	$0
1.25	4435	39%	$85,977	$107,471	$21,494
1.50	3103	27%	$60,156	$90,234	$30,078
1.75	1703	15%	$33,021	$57,786	$24,766
2.00	1615	14%	$31,301	$62,602	$31,301
2.50	111	1%	$2,153	$5,382	$3,229
3.00	9	0%	$168	$503	$336
Total	**11454**	**100%**	**$222,045**	**$333,248**	**$111,204**

Assumptions:

- Surge rate distribution applied to the 11454 Uber records that the model predicted would have surge
- Average price for Uber records that are predicted to have surge pricing is $19.39 or $222k in revenue
- With the surge rate applied to the average price, the revenue has the potential to increase to $333k
- Possible additional revenue Uber can recognize is $111k, a 2.1% increase vs their revenue with no surge

Model Results - %, Revenue, & Record Count of Surge & Base Price by Ride Share Company

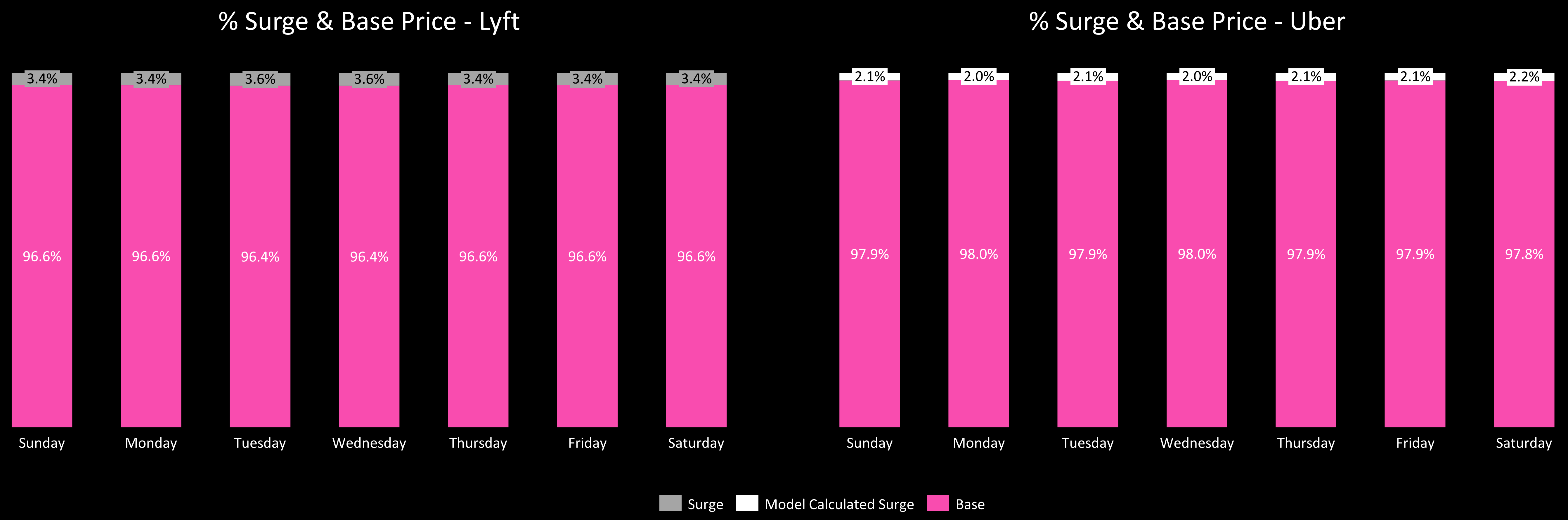


Comparisons:

- % Surge and Base Price
- Surge & Base Revenue (K$)
- Surge & Base Record count (K)

The model surge results for Uber are directionally similar to the actual surge results for Lyft

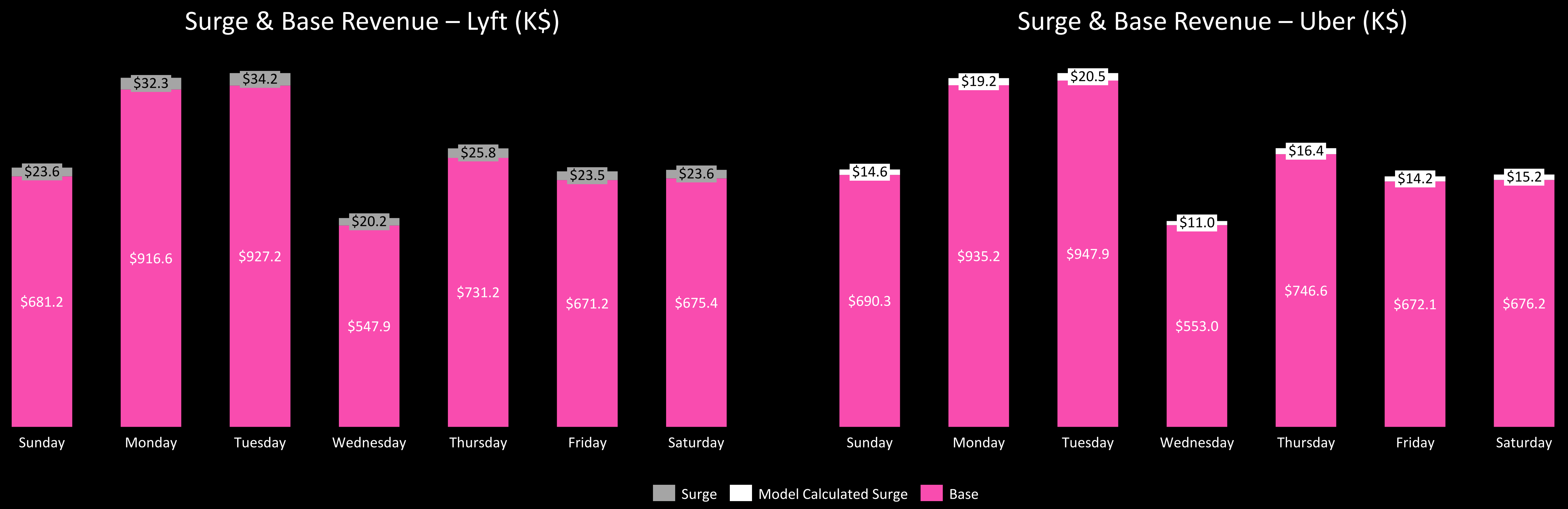
Model Results - % Surge & Base Price by Day and Ride Share Company



% Surge & Base Price by Day of the Week

- Looking at the % of surge vs the base price there are no visual signs that applying the model to Uber data creates unreasonable results

Model Results - Total Revenue - Surge & Base by Day and Ride Share Company

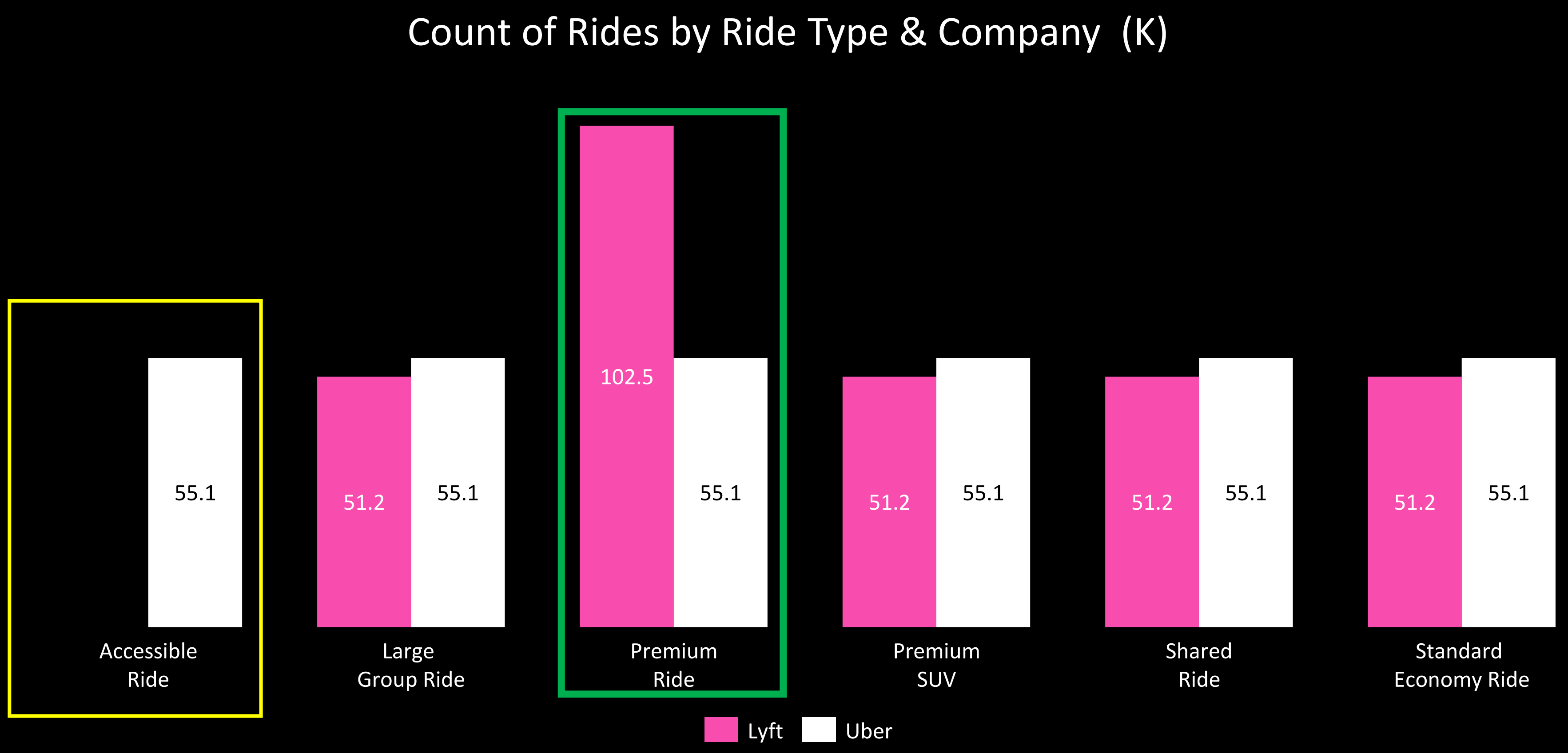


Surge & Base Revenue by Day of the Week

- Looking at the revenue by surge and base prices there are no visual signs that applying the model to Uber data creates unreasonable results
- However, even if the results do not look unreasonable, they do look like they could be low

What is the driver for the lower model results?

Model Result Validation - Total Count - Ride by Ride Type & Company



Lyft & Uber Ride count by Ride Type (K)

- The primary driver for Lyft's higher level of surge pricing is driven by the higher number of premium rides
- Uber needs t review accessible rides as there is no data in the model because the Lyft data did not have any

Overall, the model's predictions for Surge pricing for Uber produces results that are in line with expectations based on the data.

Uber ☆ vs. Lyft ♡

Total Distance from Pick-up Point

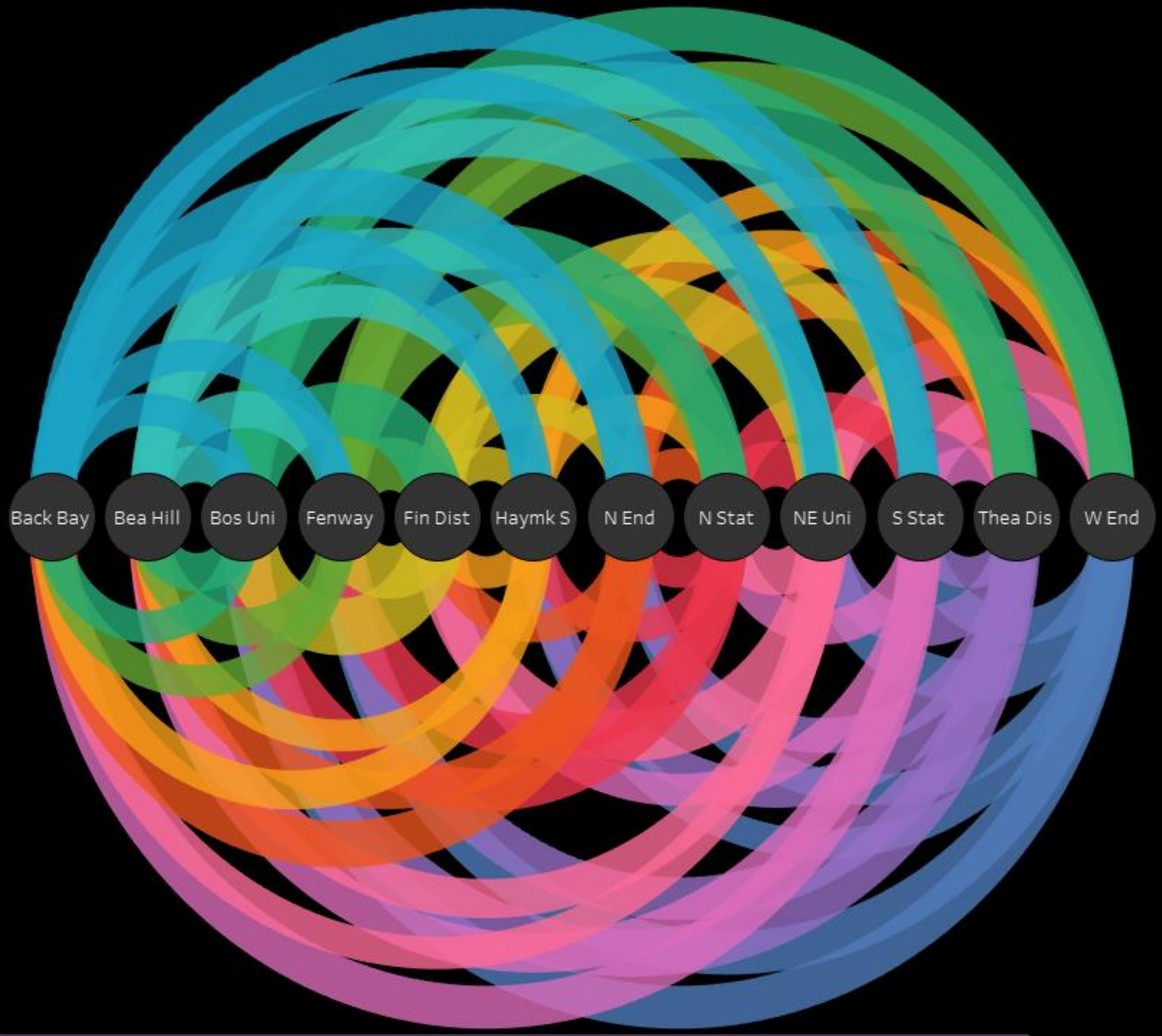
- Assumptions:
  1. Dots on the map represent the total distance the driver traveled from the pick-up point
  2. The dot location is based on the neighborhood where the pick-up location was.

  Though there are minor differences in the two maps, they are visually similar.

Finding 5 - Flow Between Pick-up & Drop-off Neighborhoods

Ride Pick-up Neighborhood	Ride Drop-off Neighborhood					
Back Bay	Boston Uni	Fenway	Haymarket Sq	North End	NE Uni	South Stat
Beacon Hill	Boston Uni	Fenway	Haymarket Sq	North End	NE Uni	South Stat
Boston Uni	Back Bay	Beacon Hill	Financial Dist	North Stat	Theatre Dist	West End
Fenway	Back Bay	Beacon Hill	Financial Dist	North Stat	Theatre Dist	West End
Financial Dist	Boston Uni	Fenway	Haymarket Sq	North End	NE Uni	South Stat
Haymarket Sq	Back Bay	Beacon Hill	Financial Dist	North Stat	Theatre Dist	West End
North End	Back Bay	Beacon Hill	Financial Dist	North Stat	Theatre Dist	West End
North Stat	Boston Uni	Fenway	Haymarket Sq	North End	NE Uni	South Stat
NE Uni	Back Bay	Beacon Hill	Financial Dist	North Stat	Theatre Dist	West End
South Stat	Back Bay	Beacon Hill	Financial Dist	North Stat	Theatre Dist	West End
Theatre Dist	Boston Uni	Fenway	Haymarket Sq	North End	NE Uni	South Stat
West End	Boston Uni	Fenway	Haymarket Sq	North End	NE Uni	South Stat

ARC SANKEY



The data exploration on the connections between ride pick-up neighborhoods and drop-off neighborhoods indicate that every pick-up neighborhood had 6 drop-off neighborhoods. The relationships between pick-up and drop-off neighborhoods can be seen in both the summary table and the Arc Sankey Diagram

Overall, there are small differences in the thicknesses of the arcs between the different ride paths

Ethical Considerations & Mitigation

1

Transparency and Customer Trust

Transparency is vital to maintaining customer trust when implementing surge pricing. Customers should understand the reasons for price increases, such as demand spikes due to weather or local events.

Provide customers separate pricing for base and surge, allows for clear identification of pricing factors.

2

Preventing Exploitative Pricing Practices

Unregulated surge pricing during emergencies or high-demand events can lead to customer dissatisfaction and harm the brand's reputation.

Implementing a surge rate constraint to ensure surge pricing remains within reasonable thresholds.

3

AI Accountability and Bias Mitigation

Bias in AI-driven pricing can unfairly target specific neighborhoods or times, leading to inequitable outcomes.

We reduced multicollinearity by removing highly correlated variables from the dataset during the model preparation.

4

Ethical Use of Public Data

Using weather and event data responsibly ensures the model accurately reflects real-world demand without creating artificial surges.

We cross-referenced event data from multiple sources and filtered it using strict criteria to include only high-impact events likely to influence demand.

Recommendations for Rideshare Companies

Implement Surge Pricing

Uber should use our surge pricing model to implement surge pricing, increase revenues and compete with Lyft. Update model for Accessible Ride considerations

Predict Surge Rates

A second model should be created to use surge predicted records to then predict the surge rate by leveraging a regression model and the dummy variables that were discarded

Explore Dynamic Pricing

Both rideshare companies have an opportunity to take surge pricing a step further by exploring dynamic pricing models.

References

1. Mengjing Chen, Weiran Shen, Pingzhong Tang, Song Zuo. "Dispatching Through Pricing: Modeling Ride-Sharing and Designing Dyn Mengjing Chen, Weiran Shen, Pingzhong Tang, Song Zuo. "Dispatching Through Pricing: Modeling Ride-Sharing and Designing Dynamic Prices." *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence (IJCAI-19)*. International Joint Conferences on Artificial Intelligence, 2019. 165-171.

2. Mobius. *Dynamic Pricing Dataset: Maximize Revenue and Profitability by Dynamic Pricing*. 28 January 2024. Kaggle. September 2024. < www.kaggle.com/datasets/arashnic/dynamic-pricing-dataset/data>.

3. Mohammad Asghari, Cyrus Shahabi. "ADAPT-Pricing: A Dynamic And Predictive Technique for Pricing to Maximize Revenue in Ridesharing Platforms." *Association for Computing Machinery*. Proceedings of the 26th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems, , 2018. 189-198.

4. Munde, Ravi. *Uber Lyft Cab Prices*. 14 November 2019. 14 September 2024. <https://www.kaggle.com/datasets/ravi72munde/uber-lyft-cab-prices>.

5. Sahota, Neil. "Harnessing AI For Dynamic Pricing For Your Business." 24 June 2024. *Forbes*.