Statistical Learning
Final Project

RIDING THE SURGE

Leveraging Machine Learning to Predict Dynamic Pricing in Ride-Sharing

Prepared By

Group 11

Ruwinda Rowel - s15654
Darshi Yashodha - s15584
Sithmi Pehara - s15494
Anjana Jayasinghe - s15627

ABSTRACT

This report introduces a data-driven approach to develop a dynamic pricing model for a ride-sharing company. Utilizing historical ride data, we employ machine learning techniques to predict optimal fares based on real-time market conditions. For this task, we used a comprehensive dynamic pricing dataset sourced from Kaggle. This study contributes to a better understanding of fare optimization in the ride-sharing industry, benefiting the ride-sharing company, customers, drivers, investors, and researchers alike.

CONTENTS

LIST OF FIGURES

LIST OF TABLES

INTRODUCTION

Dynamic pricing is an application of data science that involves adjusting product or service prices based on various factors in real time. It is employed by businesses to optimize their revenue and profitability by setting flexible prices that respond to market demand, customer behavior, and competitor pricing. Dynamic pricing has become a widely adopted strategy across industries, including travel and hospitality, transportation, eCommerce, power companies, and entertainment, leveraging vast amounts of data to adjust prices in response to changing market conditions.

"Dynamic pricing uses data to understand and act upon any number of changing market conditions, maximizing the opportunity for revenue,"

- Alex Shartsis, founder and CEO of Perfect Price

This report embarks on a journey to unravel the intricate dynamics of dynamic pricing strategies within the ride-sharing industry, with a focus on constructing a predictive model, designed to offer a comprehensive insight into the factors shaping fare optimization within this sector.

DESCRIPTION OF THE QUESTION

Ride-hailing platforms such as Uber, Lyft, and DiDi have achieved explosive growth and reshaped urban transportation. Ride-share platforms are contemporary businesses that match passengers with drivers, unlike taxis that can be hailed from the street. In the literature, the problem of optimizing the operations of such companies is mostly considered in static settings. Ride-sharing platforms are considered successful when ride-sharing fleet companies are making a profit through assigning optimal pricing and are able to manage their resources efficiently. In a dynamic pricing strategy, the aim is to maximize revenue and profitability by pricing services at the right level that balances supply and demand dynamics. It allows businesses to adjust prices dynamically. This study addresses the challenge of determining optimal ride prices over time, considering real-

world dynamics where both ride requests and available drivers fluctuate. By incorporating factors like market growth rate and balancing supply and demand dynamics, our model aims to maximize total profit for ride-share platforms while ensuring customer satisfaction, competitive pricing, and platform attractiveness.
Therefore, the objectives of this study are,

- Identify the influential predictors variables that serve as key drivers of predicting optimal fares for rides in real-time using dynamic pricing strategy.
- Build a dynamic pricing model that incorporates the provided features to predict optimal fares for rides in real-time.
- By improving operational efficiency, further modelling the change in profit percentage when the company shifts to dynamic pricing strategy from static pricing strategy.

DESCRIPTION OF THE DATASET

The dynamic pricing dataset sourced from Kaggle comprises 1000 observations and includes ten variables, of which four are categorical. It is based on a hub and spoke system where each location is a part of a wider metropolitan area the ride sharing app is frequently used. The primary response variable, 'Historical_Cost_of_Ride', represents the fare calculated based solely on ride duration. In addition, a new variable named 'Adjusted_Cost' (to enhance fare calculations by incorporating multiple factors beyond ride duration) and 'Profit_Percentage' has been introduced.

No	Variable	Data type	Description
1	Number_of_Riders	Numerical-Discrete	Historical number of riders who have used the app in specific area
2	Number_of_Drivers	Numerical-Discrete	Historical number of drivers in that specific area
3	Location_Category	Categorical-Nominal	The category that specific location falls under
4	Customer_Loyalty_Status	Categorical-Ordinal	Loyalty rating given to customers in that area
5	Number_of_Past_Rides	Numerical-Discrete	The historical count of rides a customer has taken
6	Average_Ratings	Numerical-Continuous	Average ratings given to driver in that specific location by riders from that location
7	Time_of_Booking	Categorical-Nominal	Time period of the day when most of the rides are booked
8	Vehicle_Type	Categorical-Ordinal	Main type of vehicle requested in that area
9	Expected_Ride_Duration	Numerical-Discrete	Expected duration of the ride from pick up to drop off
10	Historical_Cost_of_Ride	Numerical-Continuous	Cost of the ride borne by the rider, including any penalties incurred.
11	Adjusted_Cost	Numerical-Continuous	Historical cost adjusted for number of riders and no of drivers. (Newly defined variable)
12	Profit_percentage	Numerical-Continuous	Change in profit percentage when the company shifts to dynamic pricing strategy from static pricing strategy. (Newly defined variable)

Table 1:- : Description of the variables

FEATURE ENGINEERING

As we mentioned earlier, we defined two variables for our data set.

1. ***Adjusted_Cost :-*** The originally present variable in the dataset " Historical_Cost_of_Ride" only depends on the predictor variable "Expected_Ride_Duration". So, by further referencing we identified that variable as Static cost for each ride in the dataset. For implementation of dynamic pricing strategy, we created a new variable "Adjusted_Cost" by combining " Historical_Cost_of_Ride" with demand and supply levels. It will capture high-demand periods and low-supply scenarios to increase prices,

while low-demand periods and high-supply situations will lead to price reductions. This newly defined variable acts as response variable for modelling the dynamic price. (formulas used for creating new variable will be on appendix).

2. ***Profit_percentage:-*** Change in profit percentage when the company shifts to dynamic pricing strategy from static pricing strategy. This variable is used as the response variable for modelling change in profit percentage.

$$\text{Profit percentage} = \frac{\text{Adjusted Cost} - \text{Historical Cost of Ride}}{\text{Historical Cost of Ride}} * 100$$

DATA PRE-PROCESSING

- The data set was checked for duplicates and missing values. There were no duplicates and missing values.
- Checked for outliers. Not many significant outliers have been observed in the dataset. So, we decided to keep the outliers as they are in the dataset.
- Then, the dataset was split into training and test sets. The training set consisted of 800 observations. Descriptive analysis was conducted using the training set.
- From the predictor space we remove the historical ride cost variable as we use this variable directly to create our new response variable.

IMPORTANT RESULTS OF DESCRIPTIVE ANALYSIS

Distribution of Main Response Variable: Adjusted Cost

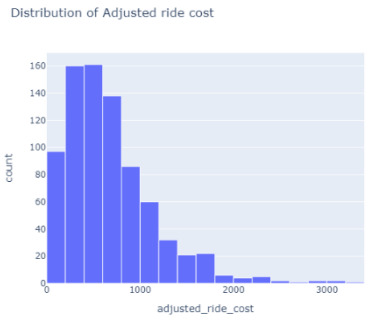


Figure 1:Histogram of Adjusted Cost

The distribution of the main response variable is left skewed with more observations with low adjusted cost values. This is mainly because most of the customers in this ride sharing company schedule the rides at low costs. The mean and median of adjusted cost variable is 680.79 and 582.77 respectively.

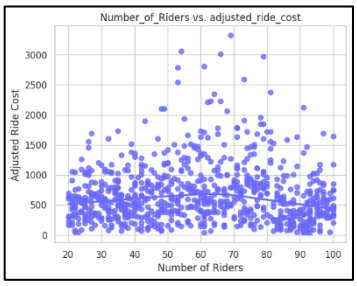


Figure 2:- Scatter Plot No of Riders vs Adjusted Cost

Relationship of Predictor Variables with Adjusted Cost

The analysis of the data set reveals several key findings. While no supply curve is evident between adjusted cost and the number of riders, a clear demand curve emerges between adjusted cost and the number of drivers, indicating a decrease in price with decreasing demand and available drivers.

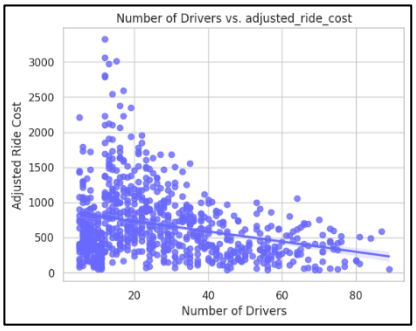


Figure 3:- Scatter Plot No of Drivers vs Adjusted Cost

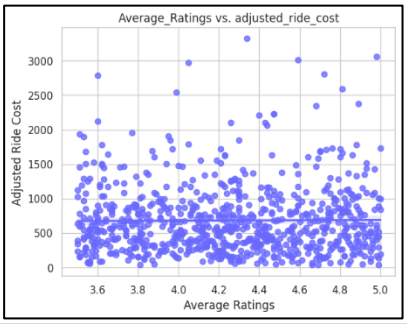
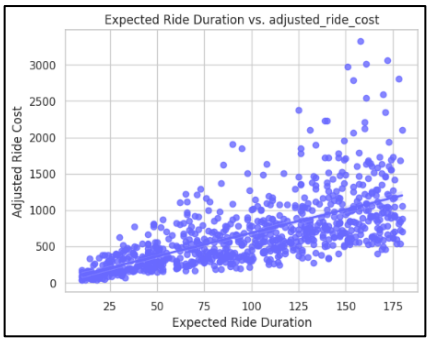


Figure 4:- Scatter Plot Average Rating vs Adjusted Cost



However, as expected, longer rides tend to incur higher fares. Surprisingly, average ratings do not appear to impact pricing directly.

Figure 5:- Scatter Plot Expected Ride Duration vs Adjusted Cost

The bar plots of average adjusted ride cost versus categorical predictors are shown in the below. By considering these bar plots some significant patterns can be identified. Premium vehicle types and afternoon bookings exhibit notably higher mean adjusted ride costs, while location categories and customer loyalty show no clear distinction.

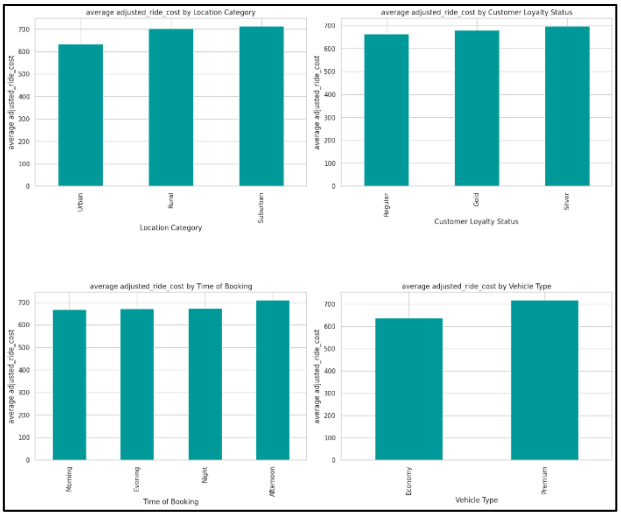
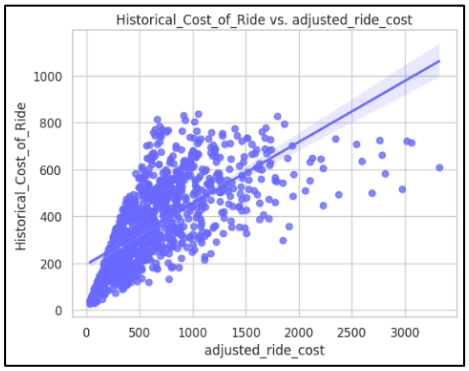


Figure 6:- Bar Plot for Mean of Categorical variables with Adjusted Cost



Comparing historical cost to adjusted ride cost demonstrates a rapid linear increase between the two variables. This is justifiable because we use historical ride cost to create new response adjusted ride cost.

Figure 7:- Scatter Plot Adjusted Cost vs Historical Ride Cost

The scatter plot analyzing adjusted ride cost versus expected ride duration across vehicle types indicates a linear relationship, though there's limited variation between the two types of vehicles.

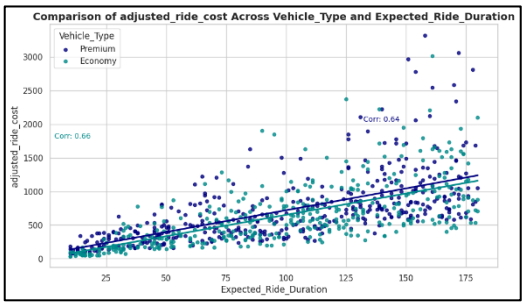


Figure 8:- Scatter Plot Expected Ride Duration vs Adjusted Cost grouped by Vehicle Type

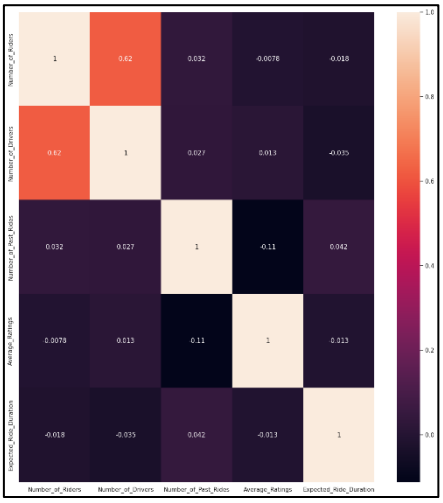


Figure 10:- Correlation between Numerical Predictors

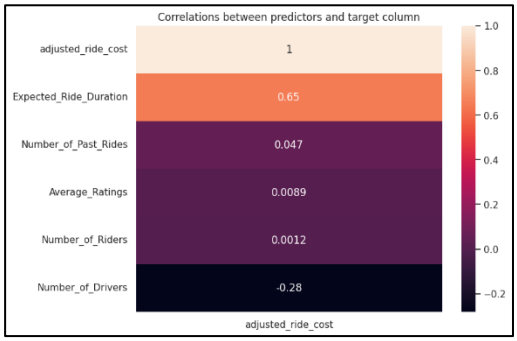


Figure 9:-Correlation of Numerical Predictors with Response

After exploring the training dataset graphically, we utilize the relationship with each predictor with adjusted cost variable numerically. The analysis reveals that adjusted ride cost correlates positively with expected ride duration, indicating that longer rides tend to incur higher costs and somewhat negatively with the number of drivers, suggesting that as the number of available drivers increases, pricing may decrease due to increased competition.

By considering correlation heat map between predictors, there is a moderately strong correlation between the number of riders and drivers indicating a close relationship between demand and supply dynamics which indicated multicollinearity.

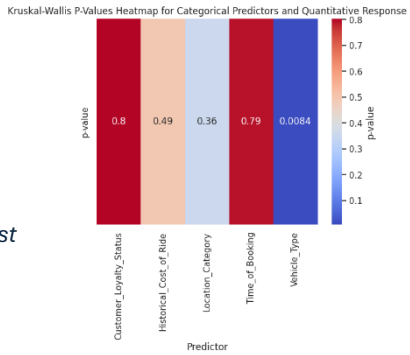


Figure 11:- KW values for Categorical Predictors with Adjusted Cost

Relationship of Predictor Variables with Profit Percentage

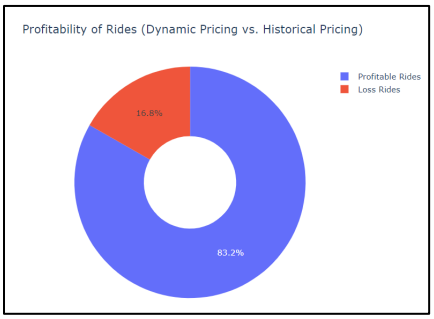


Figure 12:- Distribution of profit and loss transitioning to Dynamic Pricing

Transitioning from static to dynamic pricing results in 83.2% profitable rides and 16.8% loss rides for the company. This ensures the enhancement of profitability of the company through a dynamic pricing strategy.

By initial graphical and numerical analysis, we identified that only no of riders and number of drivers are influential factors for change in profit percentage.

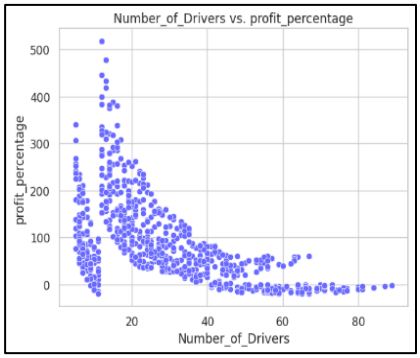


Figure 14:- Scatter Plot of Number of Drivers vs Profit Percentage

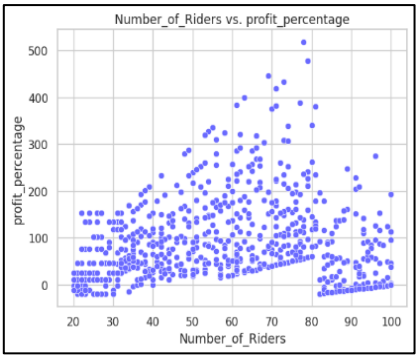


Figure 13:- Scatter Plot of Number of Riders vs Profit Percentage

The decreasing curve observed in the profit percentage versus the number of drivers suggests that as the number of drivers increases, individual driver earnings may decrease due to heightened competition, leading to lower overall profit margins for the company. Conversely, the increasing, somewhat curved relationship between profit percentage and the number of riders implies that as the number of riders grows, the company's profitability tends to improve.

Most predictors show no linear relationship with profit percentage, except for a negative association with the number of drivers. Customer loyalty status emerges as somewhat related to profit percentage based on the Kruskal-Walli's test.

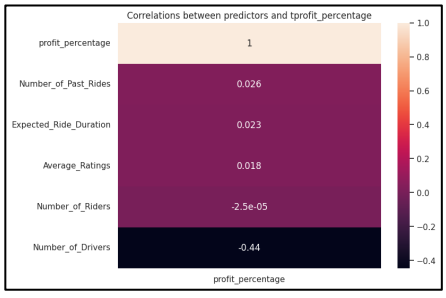


Figure 16:- Correlation of Numerical Predictors with Profit Percentage

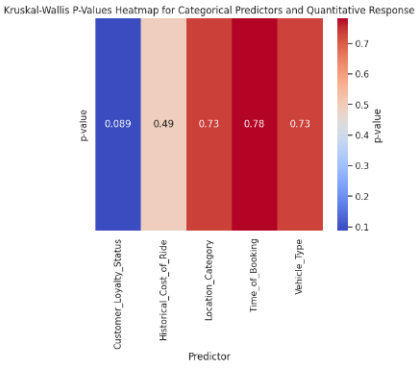


Figure 15:- KW values of Categorical Predictors with Profit Percentage

Cluster Analysis

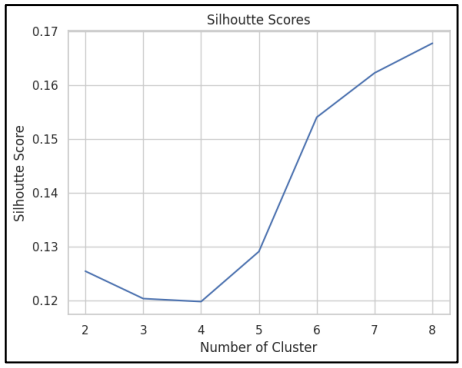


Figure 18:- Plot of Average Inertia vs No of clusters.

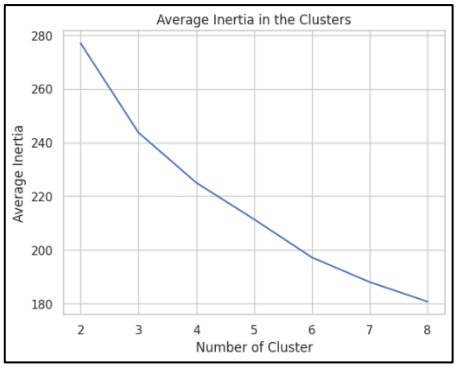


Figure 17:- Silhouette Score Plot

Finding out whether clusters exist in our dataset is important given that it may help with improving results in the advanced Analysis. For our dataset given the mixed and non-standard data, we used the k-medioids method with the distance criterion as the gower distance method. Applying this to our dataset and observing the silhouette scores and average inertieas as above. Observing the inertia plot we can see as the number of hypothesized clusters increases, the Average Inertia does not flatten out, indicating there are not many distinct clusters. The silhouette score also increases, which is common as the number of data points also increase. Thus, by looking towards these two graphs we mainly concluded that there are no clusters in our dataset.

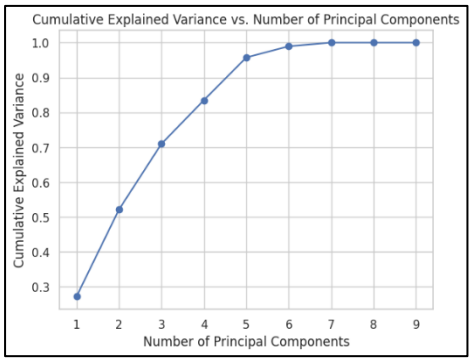


Figure 20:- Scree Plot of PCA

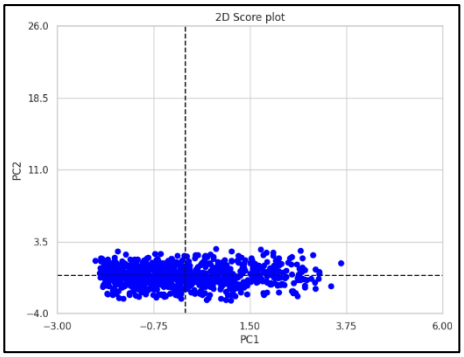


Figure 19:- Score Plot of PCA

To support our hypothesis, we also conducted a Principal Component Analysis on the quantitative variables and partial least squares scoring of all the variables. In the PCA. The first two principal components explained roughly 53% of the variation in the quantitative variables. Plotting the first two principal components also indicated signs of no clusters.

IMPORTANT RESULTS OF ADVANCED ANALYSIS

In advance analysis our focus is to build two machine learning models to predict the adjusted ride cost and profit percentage. We didn't identify any significant outliers in the data preprocessing stage. So, we carried out advanced analysis using the whole dataset without removing any data points. In the stage of cluster analysis, we identified only one cluster using all datapoints. So, we decided to fit models to considering whole dataset. Numerical variables were standardized, while nominal and ordinal qualitative variables were encoded with One-Hot Encoder and Ordinal Encoder in Python, ensuring appropriate data transformation for subsequent analysis.

<u>Building a model to predict Adjusted Cost</u>

Multiple Linear Regression (MLR) – Forward Selection

	RMSE	R2
Training set	345.6619	0.4957
Test set	324.2187	0.5126

Table 2:- : Evaluation metrics for MLR (Adjusted Cost)

We opted for MLR as our initial model as it is the simplest and fundamental model that aligns well with the characteristics of our dataset. While exploring feature selection methods, forward selection showed better results than backward selection and the model with all predictors. So, we use MLR with forward selection as the base model for predicting adjusted cost.

Furthermore, to check the validity of the given model we conducted a residual analysis.

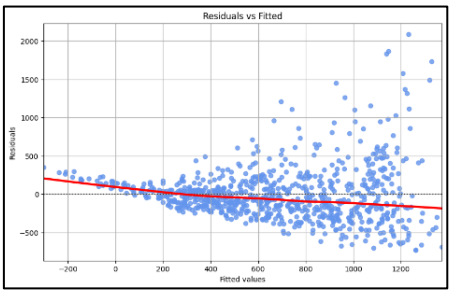


Figure 21:- : Residual Vs Fitted Values for MLR (Adjusted Cost)

1. Linearity

The correlation plot underscores small associations between predictors except "expected ride duration" and the response variable, indicating a limited strength of linearity in the model.

2. Independence & Homoscedasticity

The residual vs. predicted plot reveals a lack of random scattering around the zero-center line, accompanied by a corn shape. This implies a violation of both homoscedasticity and independence of residuals.

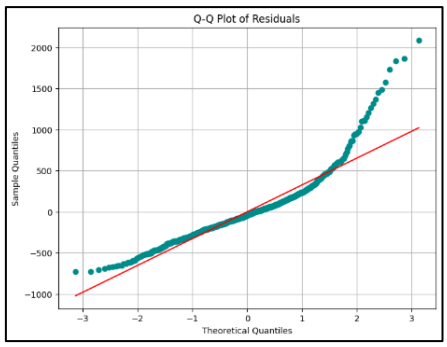


Figure 22:- : Q Q plot of Residuals for MLR (Adjusted Cost)

3. Multicollinearity

The variables 'No. of riders' and 'Average ratings' exhibit Variance Inflation Factor (VIF) values exceeding 10, indicating the presence of multicollinearity.

4. Multivariate Normality

Q-Q plot indicates the departure from normal distribution assumptions in the model's residuals.

Regularization Methods

Following the MLR analysis, the incorporation of regularization methods is recommended. These techniques address potential pitfalls observed in the MLR model, providing a valuable means to improve overall robustness and reliability of the model.

Model	Best parameter	Train RMSE	Test RMSE	Train R2	Test R2
Ridge	10.0	328.4889	312.9150	0.5446	0.5460
Lasso	10.0	331.1707	312.4690	0.5371	0.5473
Elastic Net	0.1	328.6806	312.5744	0.5440	0.5470

Table 3:- : Evaluation metrics for Ridge, Lasso, & Elastic Net (Adjusted Cost)

By considering the test RMSE values of three shrinkage methods, all three values are approximately equal. However, test and train RMSE for shrinkage methods are lower than that of our base model (MLR) indicate that shrinkage methods outperform MLR for this dataset.

Dimension Reduction Methods

	RMSE	R2
Training set	328.4239	0.5447
Test set	313.4026	0.5446

Table 4: - : Evaluation metrics for PLSR (Adjusted Cost)

Principal component regression and Partial least square regression have been suggested as dimension reduction techniques. Since PLSR considers both response and predictors when dimension reduction process, we consider only PLSR for modelling purposes. By using dimension reduction technique for dataset our main aim to remove multicollinearity not to reduce dimensions as our dataset only contains 10 predictors. Using cross validation, we found the optimal number of

components to be 6 indicating no dimensionality reduction. Performance of PLSR is shown in the above table.

Considering the results we obtained for regularization methods it is evident that there is room for improvement for our model. So, we decided to further improve our model by employing tree-based methods.

Tree -Based Methods

Tree-based algorithms are robust to outliers and multicollinearity issues, and they can capture non-linear relationships between predictors and the response variable.

When examining the Variance Inflation Factor (VIF) values among predictors, we observed multicollinearity issues. Furthermore, when assessing the correlation between the response variable (adjusted_cost) and predictor variables, most predictors showed weak correlations with the response variable. This suggests a potential non-linear and joint relationship between predictors and the response variable.

The specialty of tree-based modelling lies in the fact that there are not many assumptions to satisfy, and there is no need to scale the data before modelling.

Regression Trees

	RMSE	R2
Training set	200.0465	0.8311
Test set	253.0858	0.7030

Table 5:- : Evaluation metrics for Regression Trees (Adjusted Cost)

Regression trees partition a data set into smaller groups and then fit a simple model (constant) for each subgroup. Upon training the model and tuning parameters the following results were obtained. We can observe a somewhat higher training R-squared value than testing R-squared which is indication of slight over-fitting.

Random Forest

Random Forest is a machine learning algorithm that uses regression trees as its base learning model. The underlying assumption of Random Forest is that each tree will make different mistakes, so combining the results of multiple trees should be more accurate than any single tree. This way, the model fits several decision tree regressors on various sub-samples of the dataset and uses averaging to improve the predictive accuracy and control over-fitting.

	RMSE	R2
Training set	132.6605	0.9257
Test set	163.4659	0.8761

Table 6:- : Evaluation metrics for Random Forest (Adjusted Cost)

Upon training the model and tuning parameters the following results were obtained. We can observe a considerable higher training R^2 value than testing R^2 which is an indication of over-fitting. But comparing RMSE values of random forest with regression tree model it has been lowered indicating the higher performance of Random Forest model.

XGBoost

XGBoost is a powerful open-source tool designed to help build better models and works by combining decision trees and gradient boosting. It is a boosting algorithm that uses bagging, which trains multiple decision trees and then combines the results. It allows XGBoost to learn more quickly than other algorithms but also gives it an advantage in situations with many features to consider. Due to the popularity of XGT boost outperforming Random Forest Classifier in various aspects, it was also decided to run XGBoost on the Dynamic Pricing Dataset.

	RMSE	R2
Training set	116.0465	0.9432
Test set	160.0257	0.8813

Table 7:- : Evaluation metrics for XGB (Adjusted Cost)

Upon training the model and tuning parameters the following results were obtained. In here testing R^2 is increases and training R^2 decreases when comparing to the Random Forest. This implies XGB controls the over-fitting issue. Considering XGB as the best model to predict adjusted cost we plotted a variable importance plot for XGB model.

According to variable importance plot 'Expected ride duration', 'No of riders, and 'Number of drivers' are more important to the model. All other variables are almost not important to the model. Then we refitted the model by removing not important variables and tuning the parameters of the model. But in that case the model performance was reduced by decreasing test R^2. So, we decided to retain all predictors for the model and XGB with all predictors was considered as the best model.

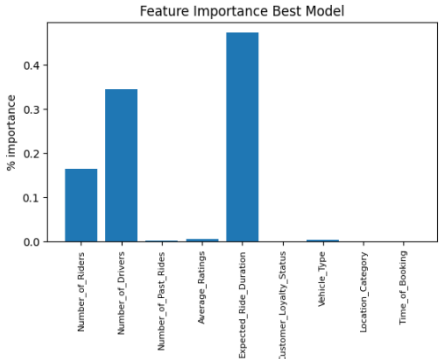


Table 8:- : Feature Importance plot of XGB (Adjusted Cost)

Building a model to predict Change in Profit Percentage

As the second objective of the project, we fitted a model to predict change in profit percentage of company when transisioning the static pricing strategy to dynamic pricing strategy.

Multiple Linear Regression (MLR) – Forward Selection

	RMSE	R2
Training set	79.6219	0.2027
Test set	76.0965	0.1624

Table 9:- : Evaluation metrics for MLR (Profit Percentage)

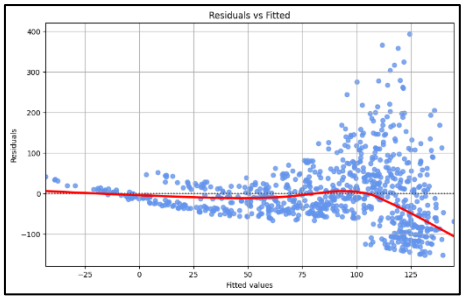


Figure 23:- : Residual Vs Fitted Values for MLR (Profit Percentage)

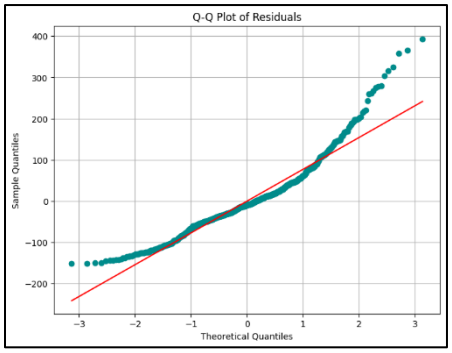


Figure 24:- Q Q plot of Residuals for MLR (Profit Percentage)

In here also we fitted MLR model as our base model. Again, while exploring feature selection methods, forward selection showed better results than backward selection.

Furthermore, to check the validity of the given model we conducted a residual analysis.

1. Linearity

The correlation plot underscores small associations between predictors except "number of drivers" and the response variable, indicating a limited strength of linearity in the model.

2. Independence & Homoscedasticity

The residual vs. predicted plot reveals a lack of random scattering around the zero-center line, accompanied by a corn shape. This implies a violation of both homoscedasticity and independence of residuals.

3. Multicollinearity

The variables 'No. of riders' and 'Average ratings' exhibit Variance Inflation Factor (VIF) values exceeding 10, indicating the presence of multicollinearity.

4. Multivariate Normality

Q-Q plot indicates the departure from normal distribution assumptions in the model's residuals.

Regularization Methods

Following the MLR analysis, the incorporation of regularization methods is recommended. These techniques address potential pitfalls observed in the MLR model, providing a valuable means to improve overall robustness and reliability of the model.

Model	Best parameter	Train RMSE	Test RMSE	Train R2	Test R2
Ridge	10.0	73.2117	72.7212	0.3259	0.2351
Lasso	1.0	73.3381	72.9607	0.3235	0.2300
Elastic Net	0.1	73.2622	72.5661	0.3249	0.2383

Table 10:- : Evaluation metrics for Ridge, Lasso, & Elastic Net (Profit Percentage)

By considering the test RMSE values of three shrinkage methods, all three values are approximately equal. However, test and train RMSE for shrinkage methods are lower than that of our base model (MLR) indicate that shrinkage methods outperform MLR for this dataset.

Dimension Reduction Methods

	RMSE	R2
Training set	73.1967	0.3261
Test set	72.9156	0.2310

Table 11:-Evaluation metrics for PLSR (Profit Percentage)

As mentioned earlier, due to some advantages we perform only PLSR as dimension reduction techniques. Similar to objective 1, our main aim is to mitigate multicollinearity by employing the PLSR. Using cross validation, we found the optimal number of components to be 6 indicating no dimensionality reduction. Performance of PLSR is shown in the above table. The PLSR method indicates somewhat overfitting and lower test R^2 when comparing to shrinkage methods.

Tree -Based Methods

When examining the Variance Inflation Factor (VIF) values among predictors, we observed multicollinearity issues. Furthermore, when assessing the correlation between the response variable (profit_percentage) and predictor variables, most predictors showed weak correlations with the response variable. This suggests a potential non-linear and joint relationship between predictors and the response variable. Already mentioned that tree-based methods are robust to outliers and multicollinearity issues, and they can capture non-linear relationships between predictors and the response variable. Due to these advantages, we tried tree-based methods with our data set to model profit percentage as response variable.

	RMSE	R2
Training set	12.4045	0.9806
Test set	14.7560	0.9685

Table 12:- Evaluation metrics for Regression Tree (Profit Percentage)

First step we fitted regression tree as baseline tree-based method. Upon training the model and tuning parameters the following results were obtained. We can observe a somewhat higher training RMSE value than testing RMSE which is an indication of slight over-fitting. Even though regression trees are the basic tree-based algorithm it performs vastly higher than regression-based algorithm.

For further improvement of the model's performance our next try is the random forest algorithm. Upon training the model and tuning parameters the following results were obtained. There is not much improvement of performance in random forest model but gap between training RMSE and test RMSE is further increased indicating further overfitting.

	RMSE	R2
Training set	11.9325	0.9821
Test set	12.4626	0.9775

Table 13:- Evaluation metrics for Random Forest (Profit Percentage)

	RMSE	R2
Training set	8.8098	0.9902
Test set	9.4328	0.9871

Table 14:- Evaluation metrics for XGB (Profit Percentage)

Our next try is XGBoost method to model profit percentage. Upon training the model and tuning parameters the following results were obtained. When comparing to the performance of Random Forest model, XGBoost model performs well by lowering test RMSE. But still there is somewhat considerable gap between training RMSE and test RMSE which indicated overfitting.

Considering XGB as the best model to predict adjusted cost we plotted a variable importance plot for XGB model. According to variable importance plot 'No of riders', and 'Number of drivers' are more important to the model. All other variables are almost not important to the model. Then we refitted the model by removing not important variables and tuned the parameters of the model. The model's performance is as follows.

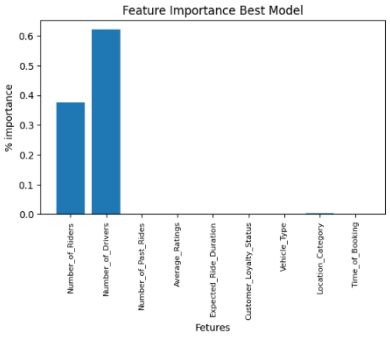


Figure 25:- : Feature Importance plot of XGB (Profit Percentage)

	RMSE	R2
Training set	8.6142	0.9898
Test set	8.9990	0.9893

Table 15:- Evaluation metrics for XGB 2 (Profit Percentage)

Surprisingly, the model performance was increased by lowering the test RMSE and the gap between training RMSE and testing RMSE also decreased by indicating effective control of over fitting. So, we considered the XGB model with removing unwanted predictors as the best model to predict profit percentage.

ISSUES ENCOUNTED AND PROPOSED SOLUTIONS

During our advanced analysis, we encountered several challenges. Fortunately, we did not face any issues during the pre-processing stage. Moving into the analysis, we identified violations of key assumptions in the MLR model. To mitigate this, we implemented diagnostic checks. In regularization techniques, we used cross-validation to optimize regularization parameters. We faced difficulties in generating a scree plot prevented us from identifying components in PLSR. We explored innovative approaches to address these issues. Finally, in tree-based methods, some variables exhibited zero importance. The solution involved the removal of these variables, followed by model refitting, optimizing the predictive power of the model. This troubleshooting approach, addressing challenges at each stage of the analysis, demonstrates our commitment to producing reliable and robust insights in the face of analytical hurdles.

DISCUSSION AND CONCLUSION

The R2 and RMSE values of all types of optimal models for modelling adjusted cost and change in profit percentage (on Test data and Training data) are summarized as follows.

Predict Adjusted Cost

Model	**Training**		**Testing**	
	RMSE	R2	RMSE	R2
MLR (forward selection)	345.6619	0.4957	324.2187	0.5126
Ridge	328.4889	0.5446	312.9150	0.5460
Lasso	331.1707	0.5371	312.4690	0.5473
Elastic Net	328.6806	0.5440	312.5744	0.5470
PLSR	328.4239	0.5447	313.4026	0.5446
Decision Trees	200.0465	0.8311	253.0858	0.7030
Random Forest	132.6605	0.9257	163.4659	0.8761
XGBoost	116.0465	0.9432	160.0257	0.8813

Table 16:- Summary of all R^2 & RMSE values (Adjusted Cost)

Predict Profit Percentage

Model	Training		Testing	
	RMSE	R2	RMSE	R2
MLR (forward selection)	79.6219	0.2027	76.0965	0.1624
Ridge	73.2117	0.3259	72.7212	0.2351
Lasso	73.3381	0.3235	72.9607	0.2300
Elastic Net	73.2622	0.3249	72.5661	0.2383
PLSR	73.1967	0.3261	72.9156	0.2310
Decision Trees	12.4045	0.9806	14.7560	0.9685
Random Forest	11.9325	0.9821	12.4626	0.9775
XGBoost	8.6142	0.9898	8.9990	0.9893

Table 17:- Summary of all R^2 & RMSE values (Profit Percentage)

As uncovered during the stepwise Advanced Analysis, XGBoost Regressor is the best model for predicting both adjusted cost for dynamic pricing strategy and profit percentage with a relatively high-Test R2 and lowest testing RMSE among all algorithms, along-side a minimal difference between train and test R 2 implying that control of over-fitting. Note that for predicting adjusted cost XGB model with all predictors is the best model and predicting profit percentage XGB model removed all unwanted predictors is the best model.

Hence, the data-products to predict adjusted cost and profit percentage will be developed containing a back-end feature where the input variables of the factors affecting to adjusted cost and profit percentage will be analyzed using a relevant Hyper-parameter tuned XGBoost Regressor to give an output of the adjusted cost and change in profit percentage.

REFERENCES

1. https://thecleverprogrammer.com/2023/06/26/dynamic-pricing-strategy-using-python/#google_vignette
2. https://medium.com/@baabak/dynamic-pricing-using-machine-learning-5e882282effe
3. https://www.altexsoft.com/blog/dynamic-pricing-explained-use-in-revenue-management-and-pricing-optimization/
4. https://www.tandfonline.com/doi/full/10.1080/23311916.2023.2230710
5. Yan, Chiwei & Zhu, Helin & Korolko, Nikita & Woodard, Dawn. (2019). Dynamic pricing and matching in ride-hailing platforms. Naval Research Logistics (NRL). 67. 10.1002/nav.21872.
6. Song, J.; Cho, Y.J.; Kang, M.H.; Hwang, K.Y. An Application of Reinforced Learning-Based Dynamic Pricing for Improvement of Ridesharing Platform Service in Seoul. *Electronics* **2020**, *9*, 1818. https://doi.org/10.3390/electronics9111818

APPENDIX

1. Link for the datasets: Dynamic Pricing Datasets
2. The python code used in our project is conveniently accessible through our GitHub repository. You can find the Colab notebooks containing all relevant code by following this GitHub link: https://github.com/ruwindarowel/Dynamic-Price-Prediction-for-ride-sharing-apps.git