Synopsis

Exploratory Data Analysis (EDA): I begin by conducting Exploratory Data Analysis to gain insights into the dataset's distributions, relationships and potential features for prediction. Notable things to be visualized are correlations, distributions, and interactions among variables. This process allowed me to identify key variables that could impact vehicle performance.

Model Comparison: To predict the transmission type, three potential models were considered: Random Forest, Decision Tree Classifier, and Logistic Regression. These models were trained and compared using their performance metrics. Initial results showed promising outcomes, with the Random Forest and Decision Tree Classifier models outperforming the Logistic Regression model in terms of predictive accuracy.

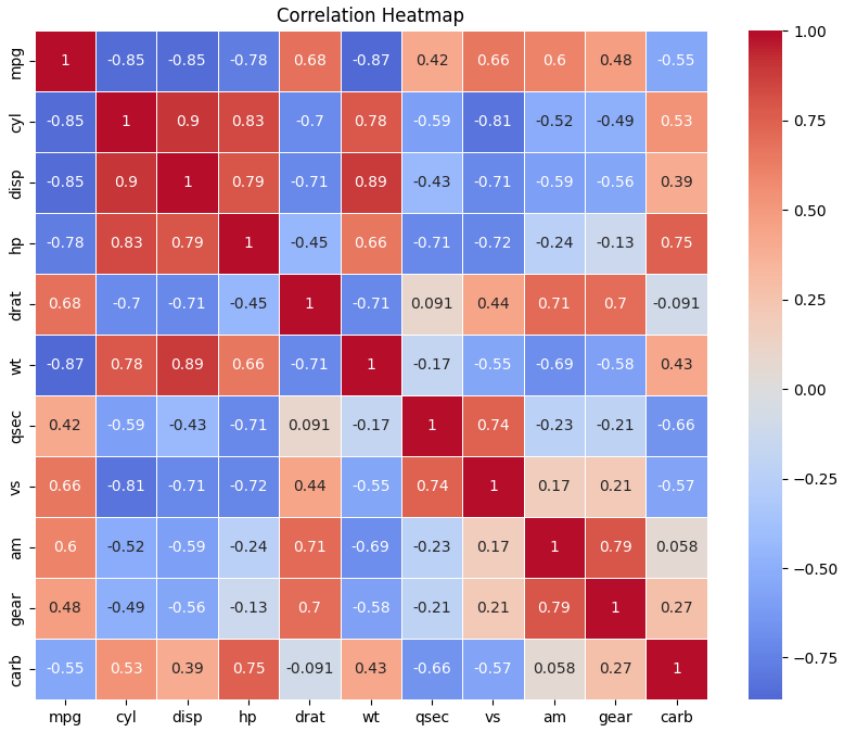
Logistic Regression Improvement: Despite the Logistic Regression model's initial performance lagging behind, room for improvement was identified. By narrowing down the dataset to only include the “mpg” and “qsec” features, I was able to enhance the model's predictive capabilities. The revised Logistic Regression model's performance improved significantly, aligning it with the accuracy of the other two models.

Feature Importance Insights: Further investigating the importance of features in the tree-based models, revealed intriguing patterns. Feature importance plots showed that “mpg” and “qsec” played a dominant role in predicting transmission type in the Logistic Regression model. The Decision Tree Classifier and Random Forest models highlighted “gear” and “wt” as significant predictors as well.

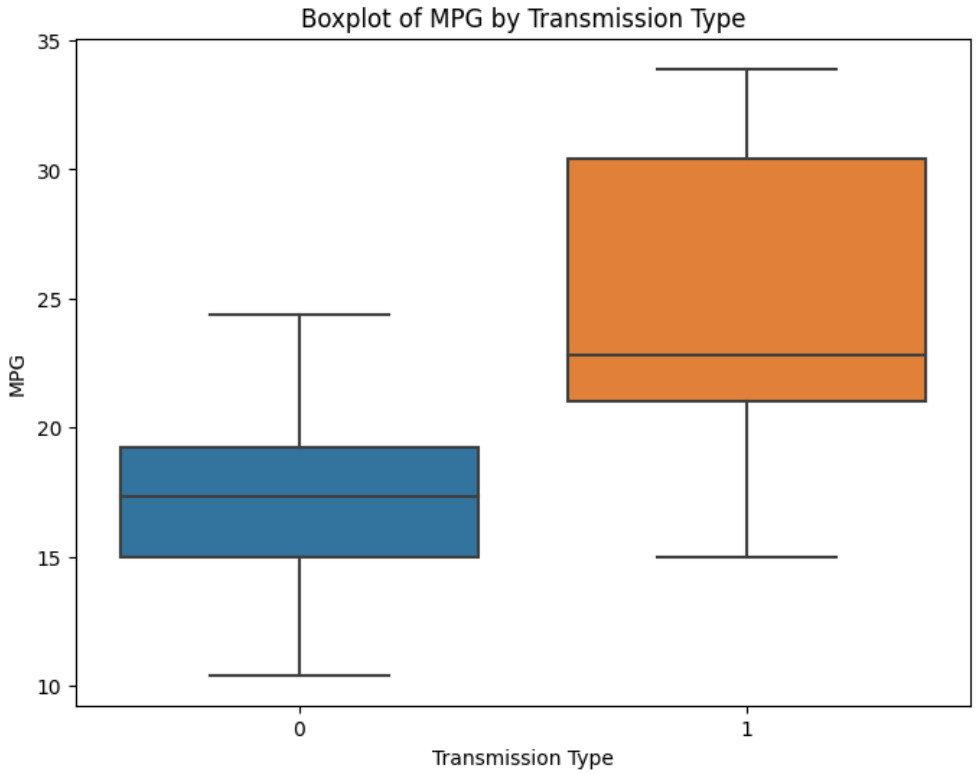
Exploring the data

The dataset originates from the 1974 Motor Trend US magazine and provides comprehensive information about 32 automobiles, specifically the 1973–74 models. It encompasses various attributes that shed light on both the performance and design aspects of these vehicles. The dataset is structured as a dataframe that is comprised of 11 numeric variables. These variables include measurements like miles per gallon (mpg), number of cylinders (cyl), engine displacement (disp), horsepower (hp), weight (wt) in thousands of pounds, 1/4 mile time (qsec), engine type (vs: 0 = V-shaped, 1 = straight), transmission type (am: 0 = automatic, 1 = manual), and the number of carburetors (carb) etc.

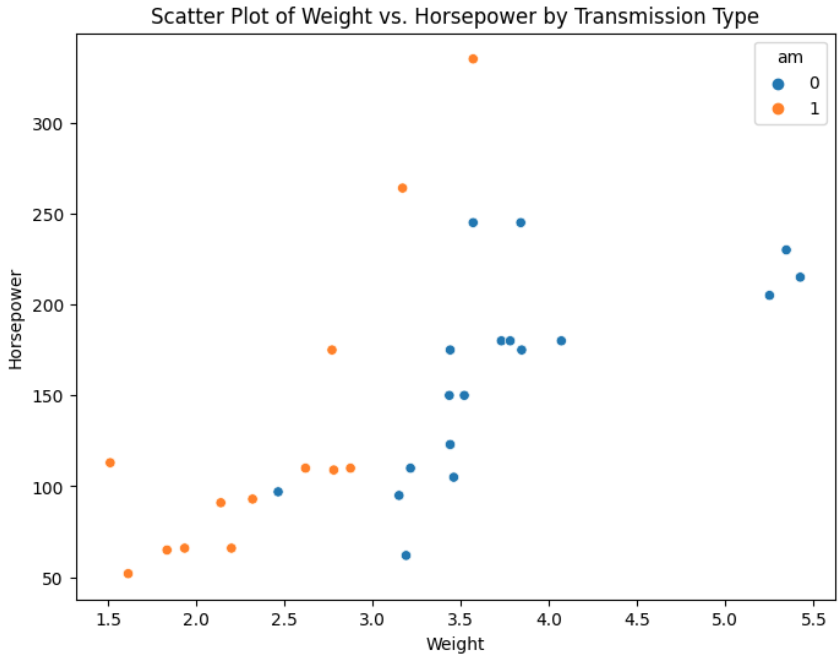
The correlation matrix tells its story.



- Miles per Gallon (mpg) vs. Other Variables: The miles per gallon (mpg) variable exhibits interesting correlations with other attributes. It negatively correlates with variables like cylinders (cyl), displacement (disp), and weight (wt), suggesting that higher mpg values are associated with fewer cylinders, lower displacement, and lighter weight vehicles. This pattern indicates a potential trend towards more fuel-efficient and smaller vehicles.
- Engine Type (vs) and Transmission Type (am): The engine type and transmission type variables are binary, representing V-shaped engines (vs = 0) and automatic transmissions (am = 0). These variables exhibit correlations with other attributes. For example, vehicles with V-shaped engines (vs = 0) tend to have more cylinders (cyl) and a higher number of carburetors (carb), while vehicles with automatic transmissions (am = 0) have higher weight (wt) and a lower number of gears (gear).
- Weight (wt) and Performance: Weight (wt) has noteworthy correlations with various performance-related attributes. It negatively correlates with variables like 1/4 mile time (qsec) and engine type (vs), indicating that lighter vehicles tend to have quicker acceleration and V-shaped engines.
- Performance Attributes: The performance-related variables, such as horsepower (hp), 1/4 mile time (qsec), and rear axle ratio (drat), exhibit correlations that align with intuitive expectations. For instance, horsepower (hp) positively correlates with 1/4 mile time (qsec), implying that higher horsepower vehicles tend to achieve quicker acceleration times.



This boxplot tells us there's a significant difference in miles per galon within the distribution of different transmission types. It might be something to look out for later.



In this scatterplot graph we can see an obvious trend in the different transmission types when compared by horsepower vs weight. Automatic cars seem to be of larger weight on average, while being packed with more horsepower than the manual cars.

Testing the models

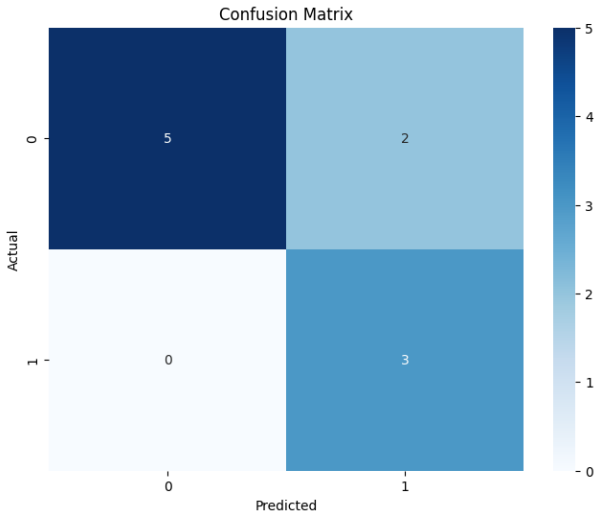
In the pursuit of classifying whether a car has an automatic or manual transmission, the choice of algorithms is crucial to ensure accurate predictions and insightful analysis. I selected three distinct algorithms—Logistic Regression, Decision Trees, and Random Forest—for this classification task, each offering unique advantages that align with the nature of the problem and the dataset characteristics.

In short, I chose Logistic Regression for its simplicity and interpretability, Decision Trees for their ability to handle non-linear relationships and provide clear insights, and Random Forest for its ensemble nature, which improves predictive accuracy and reduces overfitting. By employing this trio of algorithms, I aimed to achieve accurate classification results while gaining deeper insights into the factors influencing the type of transmission in automobiles.

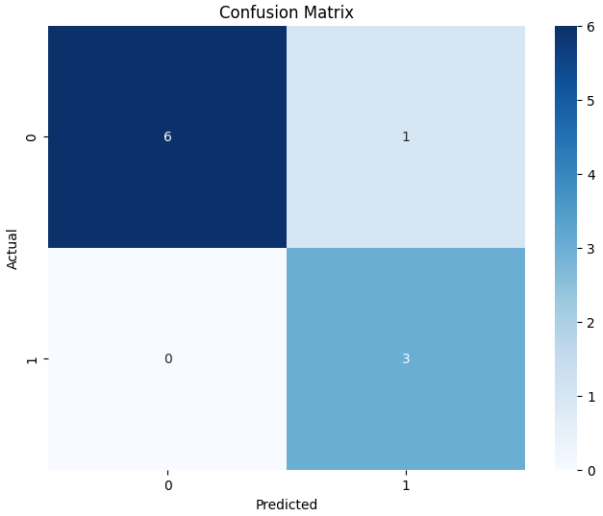
Because the dataset is relatively small I opted for a larger test set than I usually go for (0.3 ratio is still reasonable but I wanted to point that out).

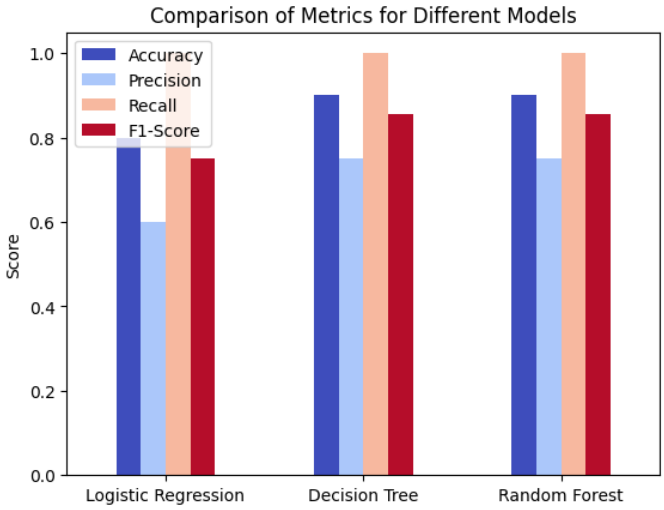
The different models perform as follows.

Logistic Regression:



Random Forest and Decision Tree classifiers:





Logistic Regression:

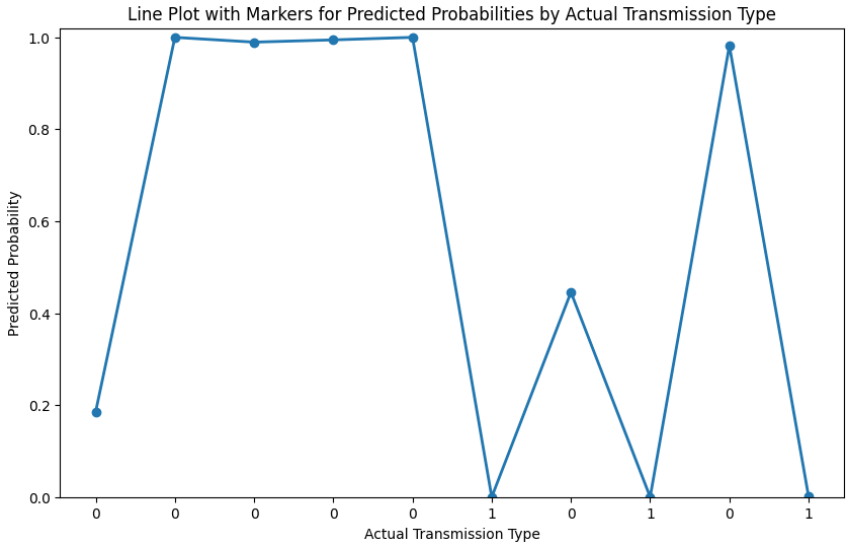
- Precision (accuracy of positive predictions): 60%
- Recall (sensitivity or true positive rate): 100%
- F1-Score (harmonic mean of precision and recall): 75%
- Accuracy: 80%

Decision Tree Classifier and Random Forest Classifier:

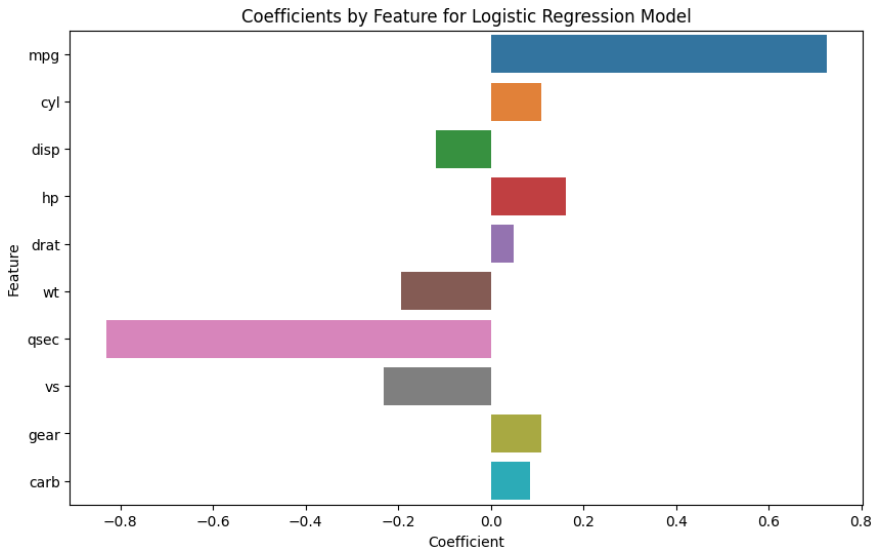
- Precision: 75%
- Recall: 100%
- F1-Score: 86%
- Accuracy: 90%

All three models demonstrate strong recall values, indicating their effectiveness in correctly identifying manual transmissions (class 1). The tree-based models' ability to achieve higher precision and F1-Score could be attributed to their capability to capture non-linear relationships and complex interactions between features. The high recall across all models implies a strong ability to identify manual transmissions, which could be crucial for practical applications where such instances hold significance.

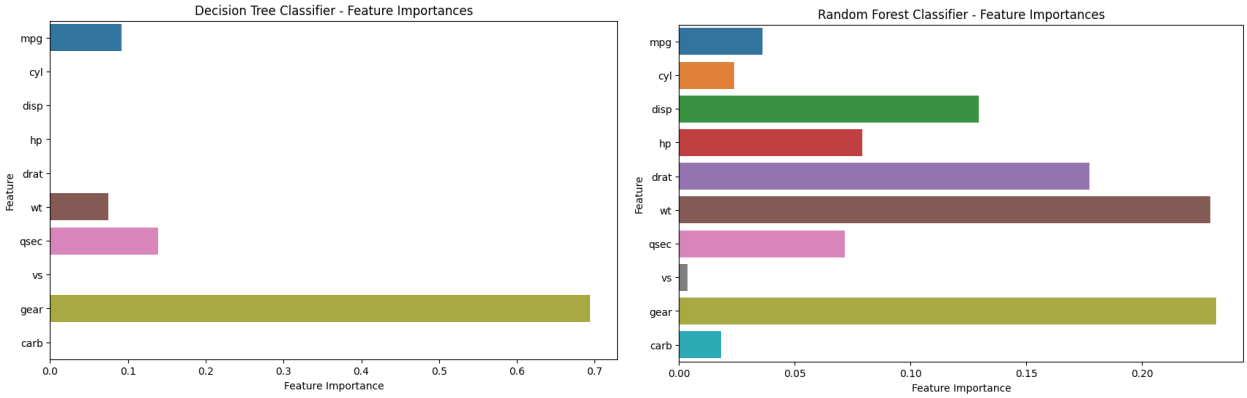
The Logistic Regression model has slightly lower precision and F1-Score compared to the tree-based models, suggesting it may struggle a bit more with distinguishing between the two classes. So, let's try and fix that!



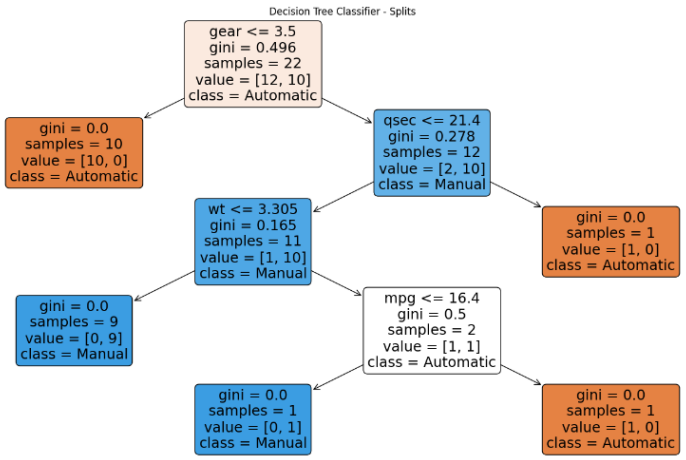
The line plot describes probabilities for the automatic class while using the Logistic Regression model. On the X-axis there are the actual transmission types by class (0,1) and the Y-axis represents the probabilities of each decision.



Here we it is clear that the most impactful features for the logistic regression model are “qsec” and “mpg”. This might be useful for having in mind when trying to improve the model’s performance.



When it comes to the features the tree algorithms used, they are quite different but overlap in that they both use the “gear” feature heavily in their decision-making, which is quite interesting as it has a high positive correlation with the target variable. On the next picture, I’ll visualize the splits of the decision tree classifier algorithm.



This provides insight into how the decisions of each class were made.

Improvements

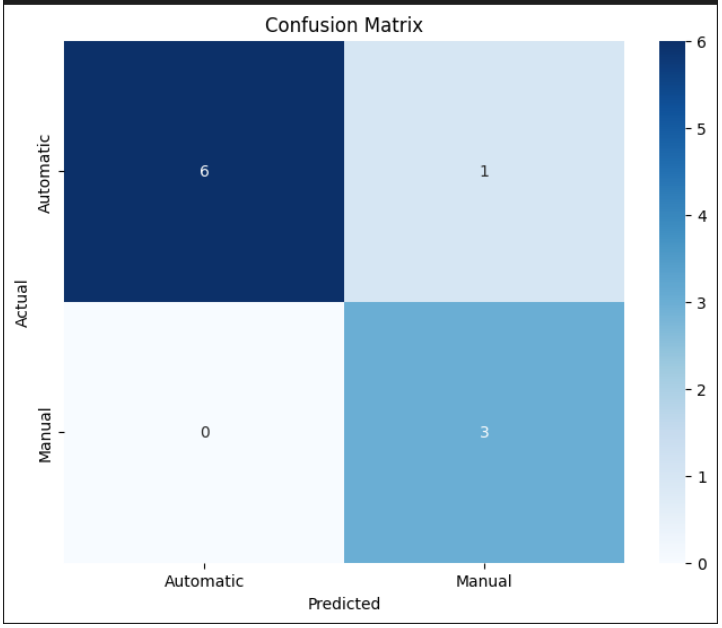
Let's try to fix the performance of Logistic regression so it doesn't lag behind the competition. I did this by cutting down on the features that the model didn't use. This could be detrimental in the case of tree focused algorithms, as they are very reliant on a larger amount of features, but in the case of the logistic regression algorithm, it yielded improvements. By removing 'cyl', 'disp', 'hp', 'drat', 'wt', 'gear' and 'carb' variables I've done one or more of the following:

- Reduced Noise: Unimportant variables may introduce noise and irrelevant information to the model. With their elimination, the model focuses on the features that have more of an impact on the target variable, leading to a cleaner signal, so to speak.
- Prevented Multicollinearity: Unimportant variables can sometimes correlate with other important variables, leading to multicollinearity. This can destabilize coefficient estimates and be impactful in the model's reliability.
- Enhanced Feature Importance: With fewer variables, the model's feature importance becomes more meaningful and accurate, allowing us to focus on the truly impactful features

And the results speak for themselves, they are in line with the tree based models now!

Classification Report:

	precision	recall	f1-score	support
0	1.00	0.86	0.92	7
1	0.75	1.00	0.86	3
accuracy			0.90	10
macro avg	0.88	0.93	0.89	10
weighted avg	0.93	0.90	0.90	10



Model Limitations and Weaknesses:

- Assumptions of Logistic Regression: Logistic regression assumes linearity between features and the log odds of the target variable, which might not hold in all cases. Violation of assumptions can lead to inaccurate predictions.
- Feature Dependency: Logistic regression assumes that features are independent of each other. If there is high multicollinearity, it can affect the reliability of coefficient estimates
- Sensitive to Outliers: Logistic regression can be sensitive to outliers, which can skew the coefficient estimates and affect model performance.

Suggestions for Improvement:

- Feature Engineering: Explore more feature transformations, interactions, or polynomial terms to capture non-linear relationships or interactions.
- Regularization: Implement regularization techniques like L1 (Lasso) or L2 (Ridge) regularization to prevent overfitting and improve generalization.
- Ensemble Methods: Experiment with ensemble methods like Gradient Boosting, or XGBoost to capture non-linear relationships and improve predictive accuracy.
- Domain Knowledge: Incorporate domain-specific knowledge to identify relevant features and potentially engineer new features that align with the problem.
- Hyperparameter Tuning: Optimize hyperparameters using techniques like grid search or Bayesian optimization to find the best configuration for the model.
- Advanced Techniques: Explore more advanced techniques like Neural Networks or Support Vector Machines, which can handle complex relationships and non-linear patterns.
- External Data: Incorporate external data sources that might provide additional insights and improve model generalization