CAR TRANSMISSION

Overview

This report presents a machine learning classification model that predicts the probability of a car having a manual transmission. The model was trained on a dataset of 22 cars, with each car's transmission type labeled as either ***(1) "manual"*** *or* ***(0) "automatic"****. The model was then evaluated on a test set of 10 cars, and it achieved an accuracy of 100%.*

Data

The data was extracted from the 1974 *Motor Trend* US magazine and comprises fuel consumption and 10 aspects of automobile design and performance for 32 automobiles (1973–74 models).

Results

The model was evaluated on a test set of 10 cars, and it labelled cars correctly in every case.

Conclusion

The results of this study suggest that machine learning can be used to effectively predict the transmission type of a car. The model developed in this study can be used by vintage car dealerships and vintage car enthusiasts that are unsure of the transmission type of a car they are interested in.

Exploratory Data Analysis

Raw data consists of 32 samples and 12 features with no missing values.

Features

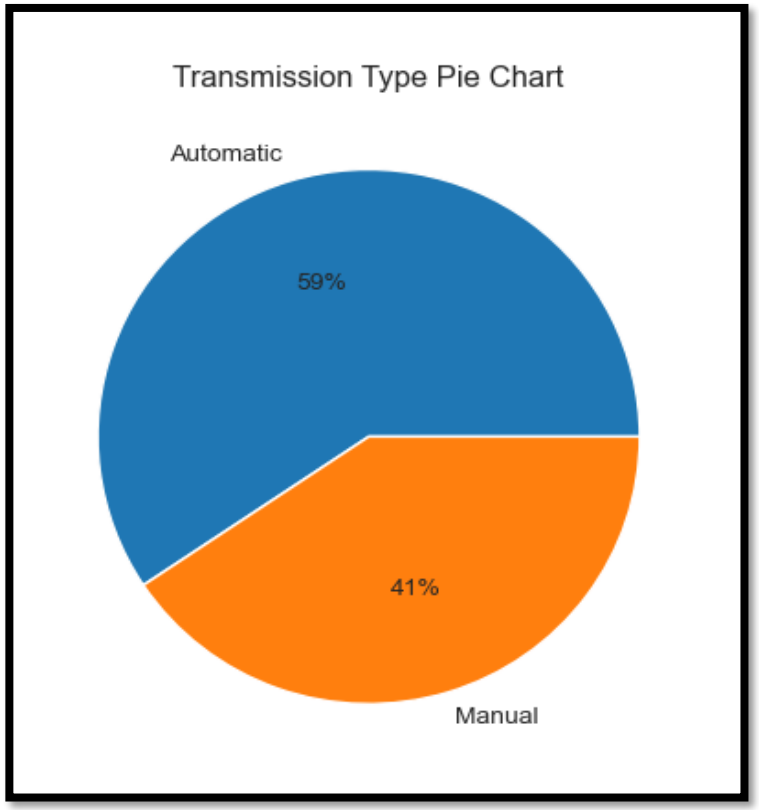
Feature name	Description	Data type	Feature name	Description	Data type
Model Name	Model Name	String	**wt**	Weight (1000 lbs.)	Decimal
mpg	Miles Per Gallon	Decimal	**qsec**	Quarter Mile Time	Decimal
cyl	Number of Cylinders	Integer	**vs**	Engine (0 = V-shaped, 1 = straight)	Categorical (0,1)
disp	Displacement	Decimal	**am**	Transmission (0 = automatic, 1 = manual)	Categorical (0,1)
hp	Horsepower	Integer	**gear**	Number of forward gears	Integer
drat	Rear Axle Ratio	Decimal	**carb**	Number of carburetors	Integer

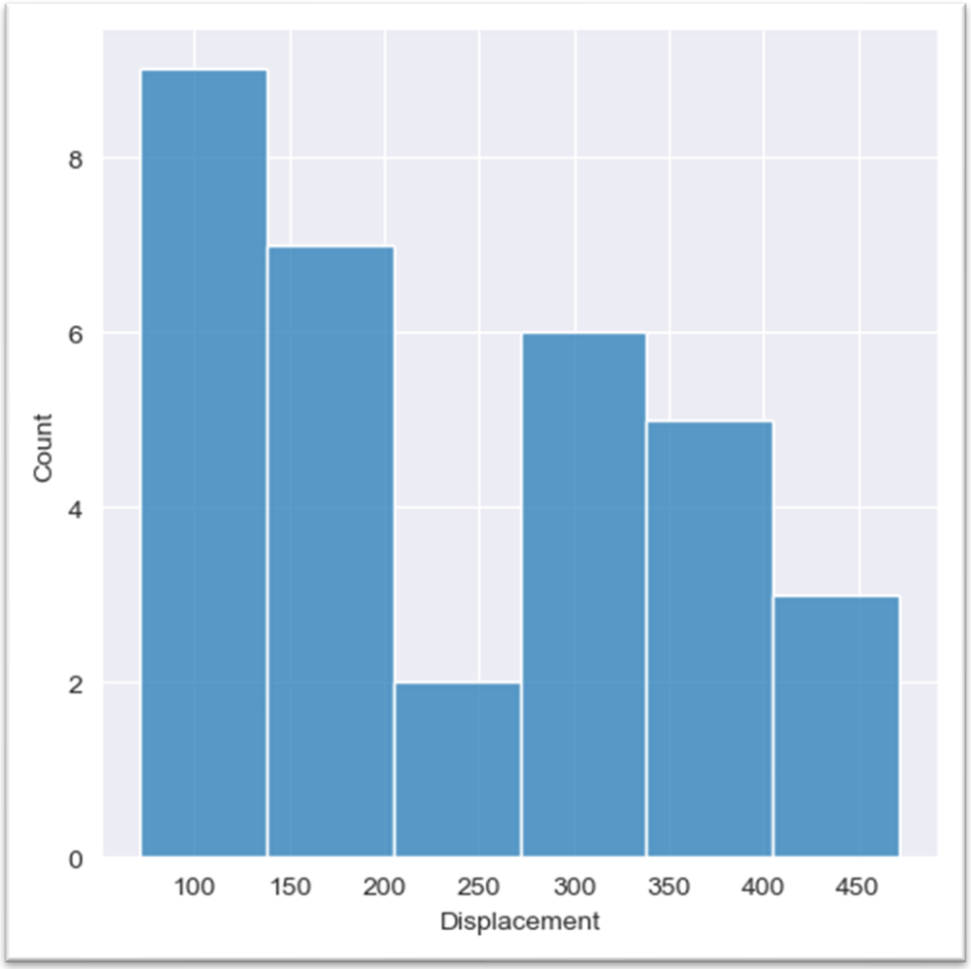
Univariate analysis

TABLE 1 DESCRIPTIVE STATISTICS

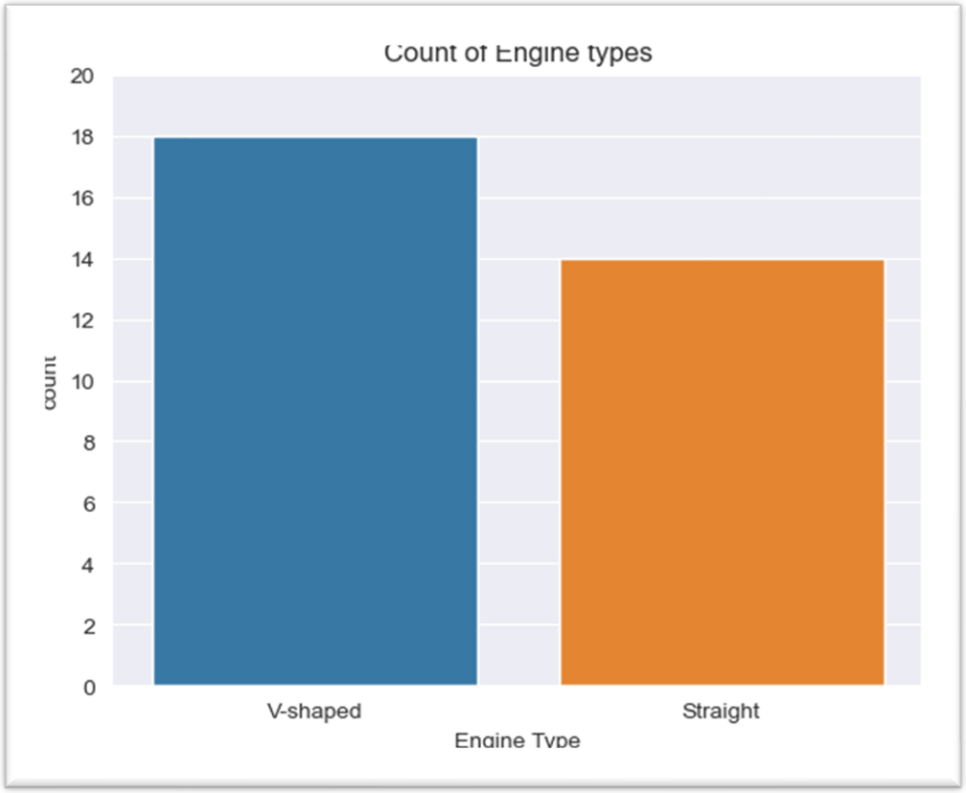
	mpg	cyl	disp	hp	drat	wt
count	32,0	32,0	32,0	32,0	32,0	32,0
mean	20,1	6,2	230,7	146,7	3,6	3,2
std	6,0	1,8	123,9	68,6	0,5	1,0
min	10,4	4,0	71,1	52,0	2,8	1,5
25%	15,4	4,0	120,8	96,5	3,1	2,6
50%	19,2	6,0	196,3	123,0	3,7	3,3
75%	22,8	8,0	326,0	180,0	3,9	3,6
max	33,9	8,0	472,0	335,0	4,9	5,4

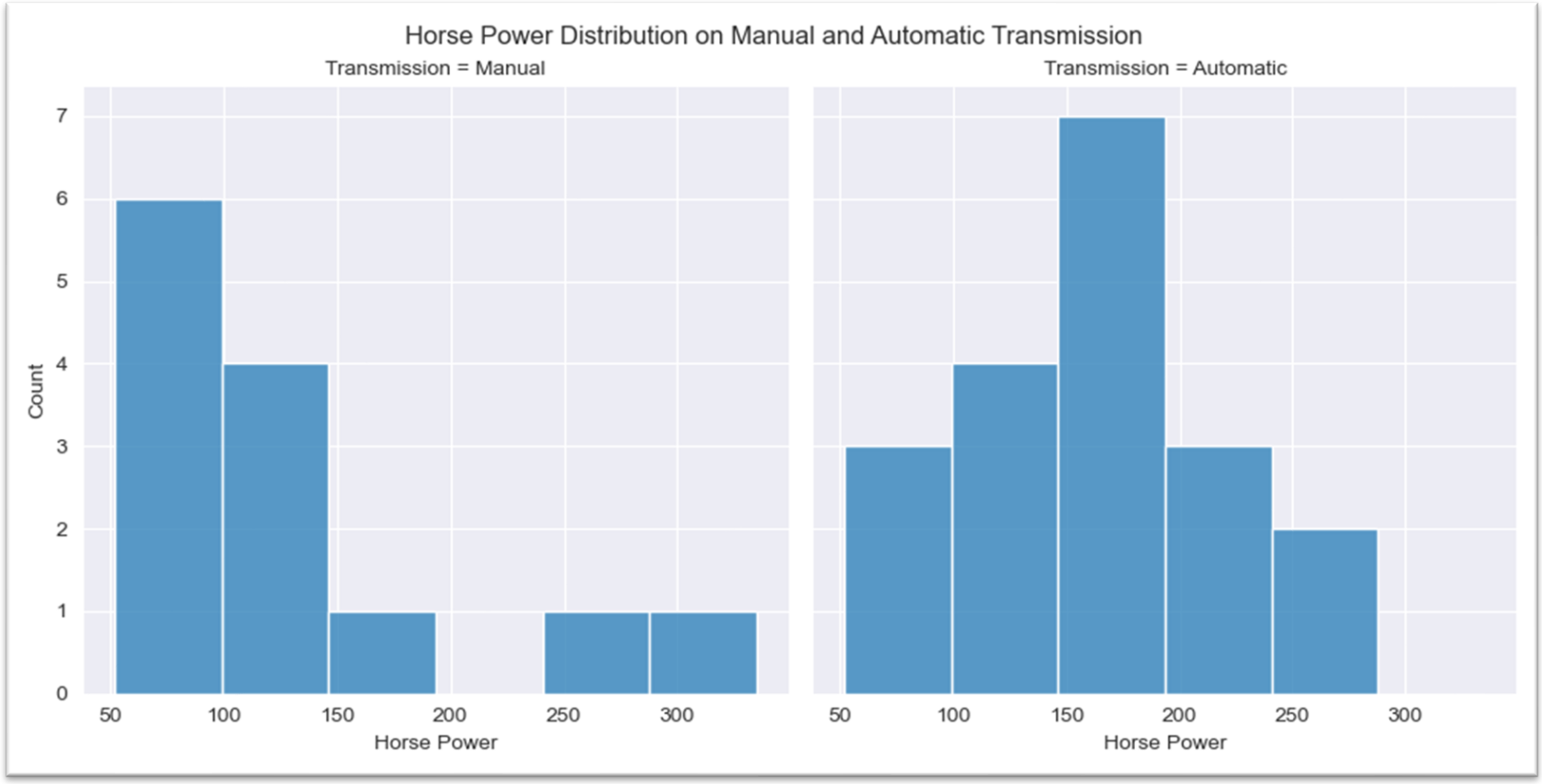
The positive and negative class ratio is important in machine learning classification because it can affect the performance of the model. If the ratio is too skewed, the model may not be able to learn the minority class well and may make more errors in classifying those samples. Here it is quite balanced.



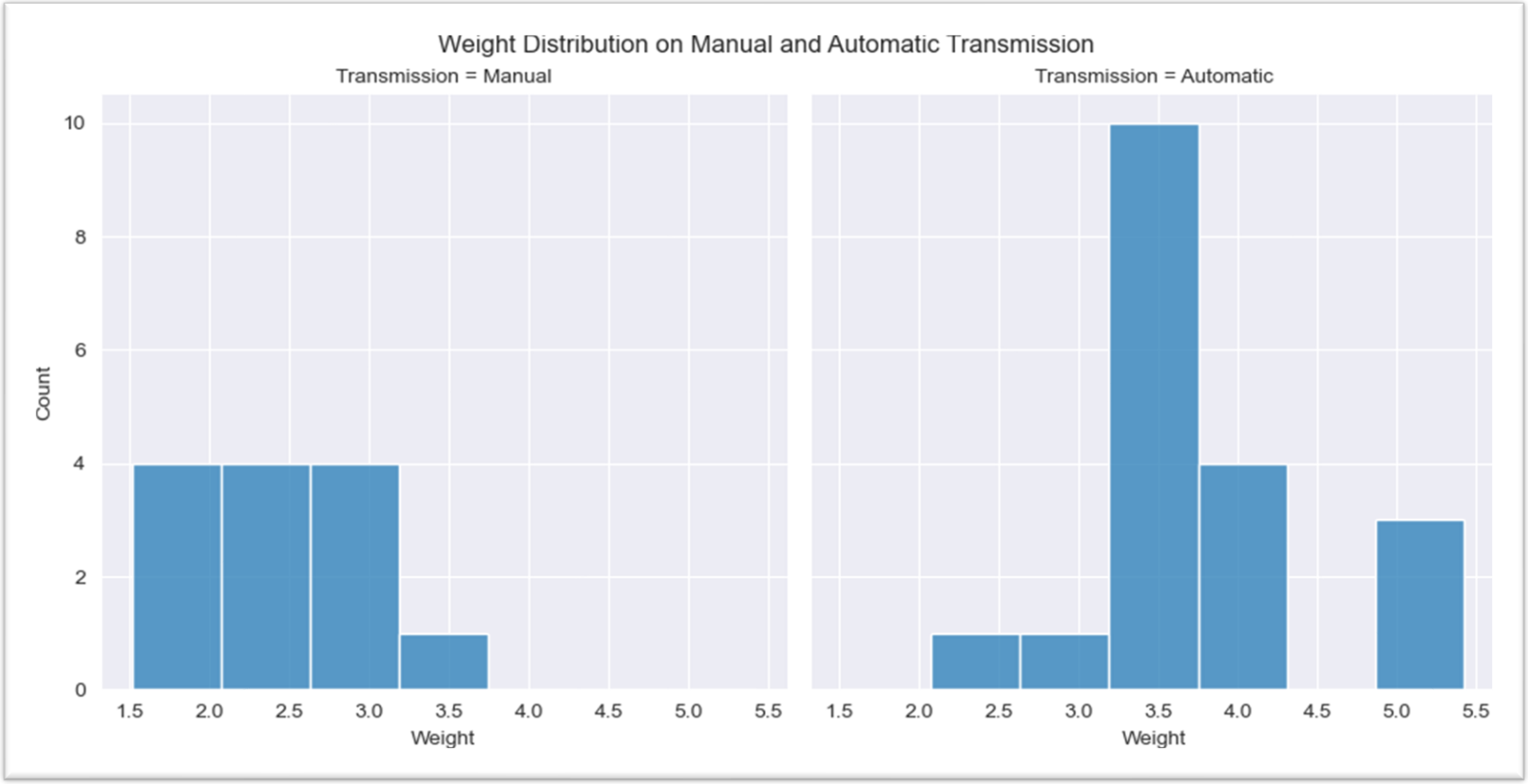


Interesting Displacement Distribution. It could mean there are classes that divide this feature.

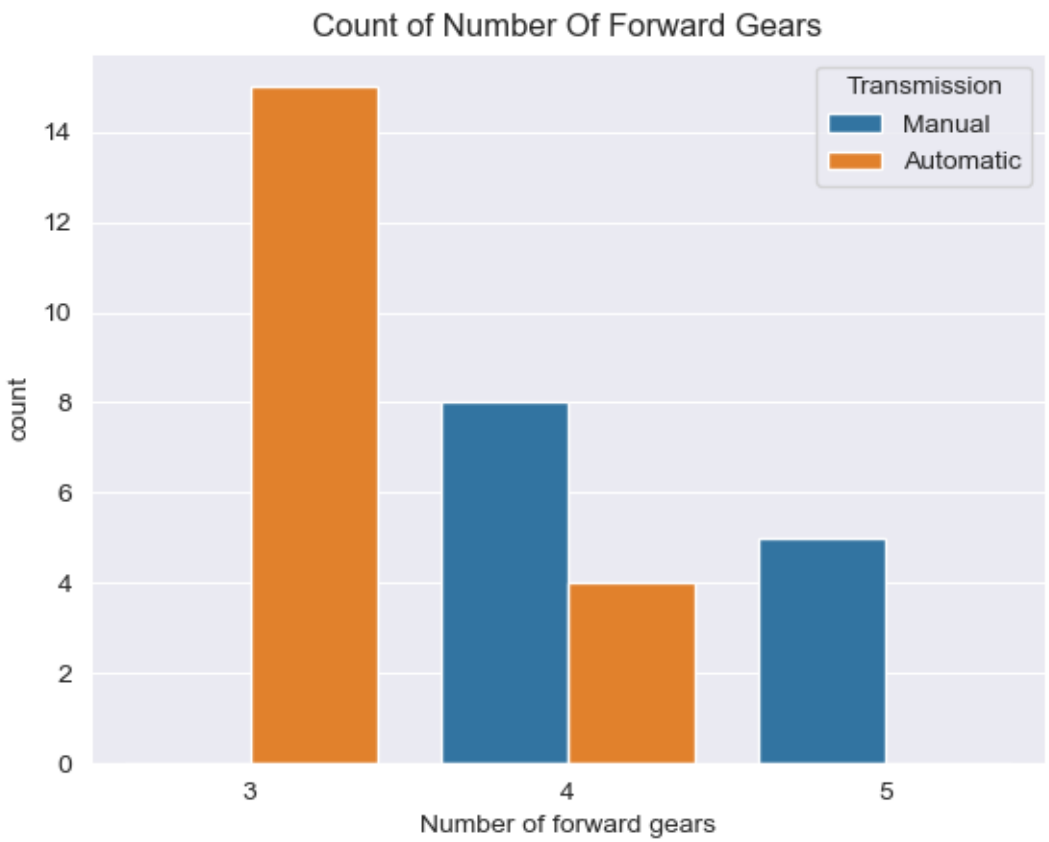




How transmission type discriminates Horsepower feature. Manual transmission cars have lower Horsepower than automatic vehicles. Manual cars have mostly from 50 to 150 HP while automatic cars have mostly from 100 to 250.



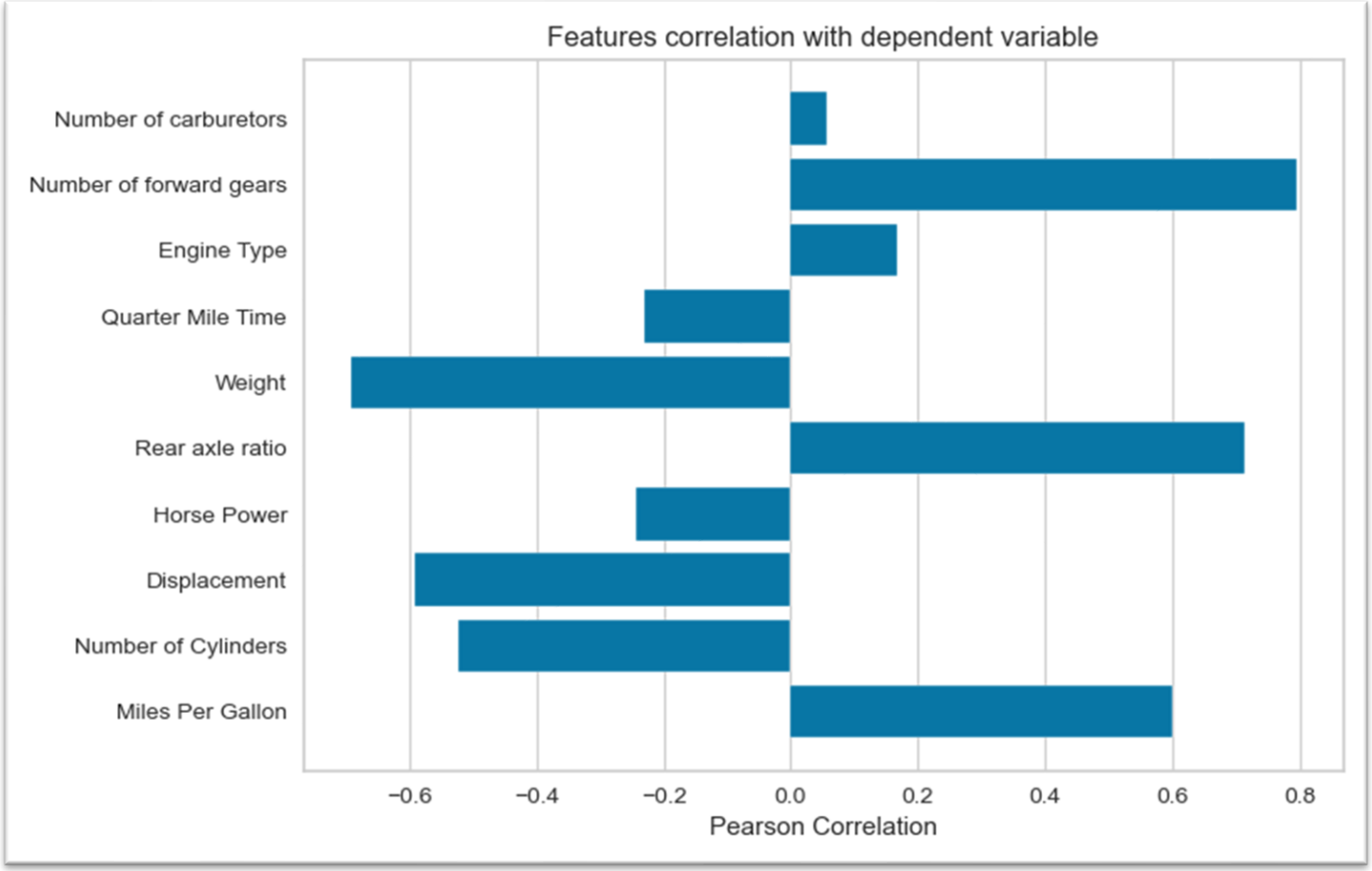
Coincidentally, Manual cars are generally much lighter in weight than automatic cars. Automatic cars are predominately 3500 to 4000 lbs., indicating Weight could be a strong predictor



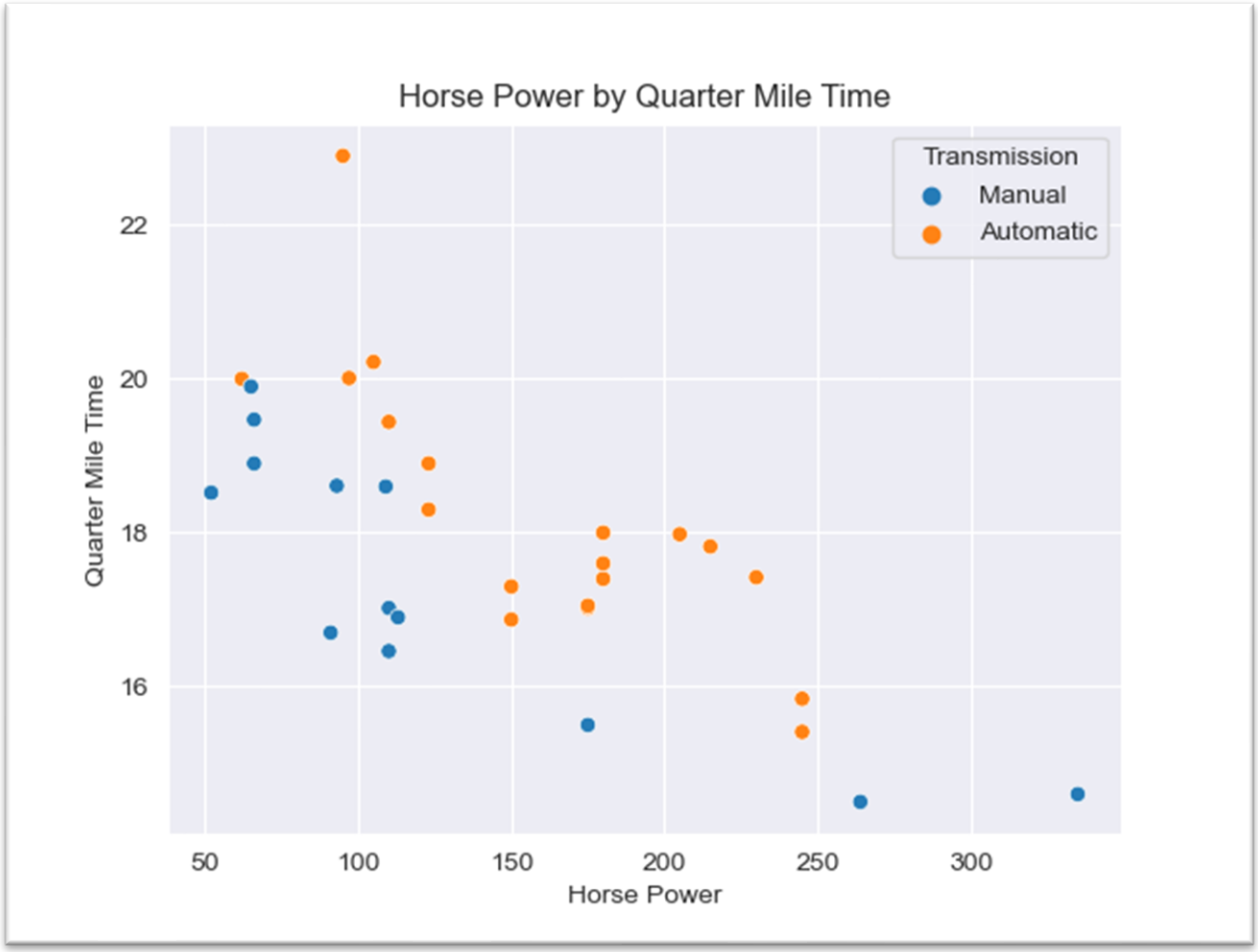
Interesting visual. It shows how no cars with 3 gears were observed to be manual, while the majority of 4 gears and all of 5 gears cars are manual. It shows that the number of forward gears is a strong predictor feature.

Bivariate Analysis

Pearson correlation scores for each feature. Correlation scores closer to 1 indicate that with increase of that feature, transmission type is more likely to be manual. As expected, greater number of forward gears means more likeliness the car is manual. Accordingly, the more weight the car has the less likely it is to be manual.



Graph also suggests manual cars use more gas and have larger Rear Axle Ratio and Lower Displacement.



Scatter plot depicting how Horsepower influences Quarter Mile Time, by transmission type. Expectedly, more Horsepower means faster car.

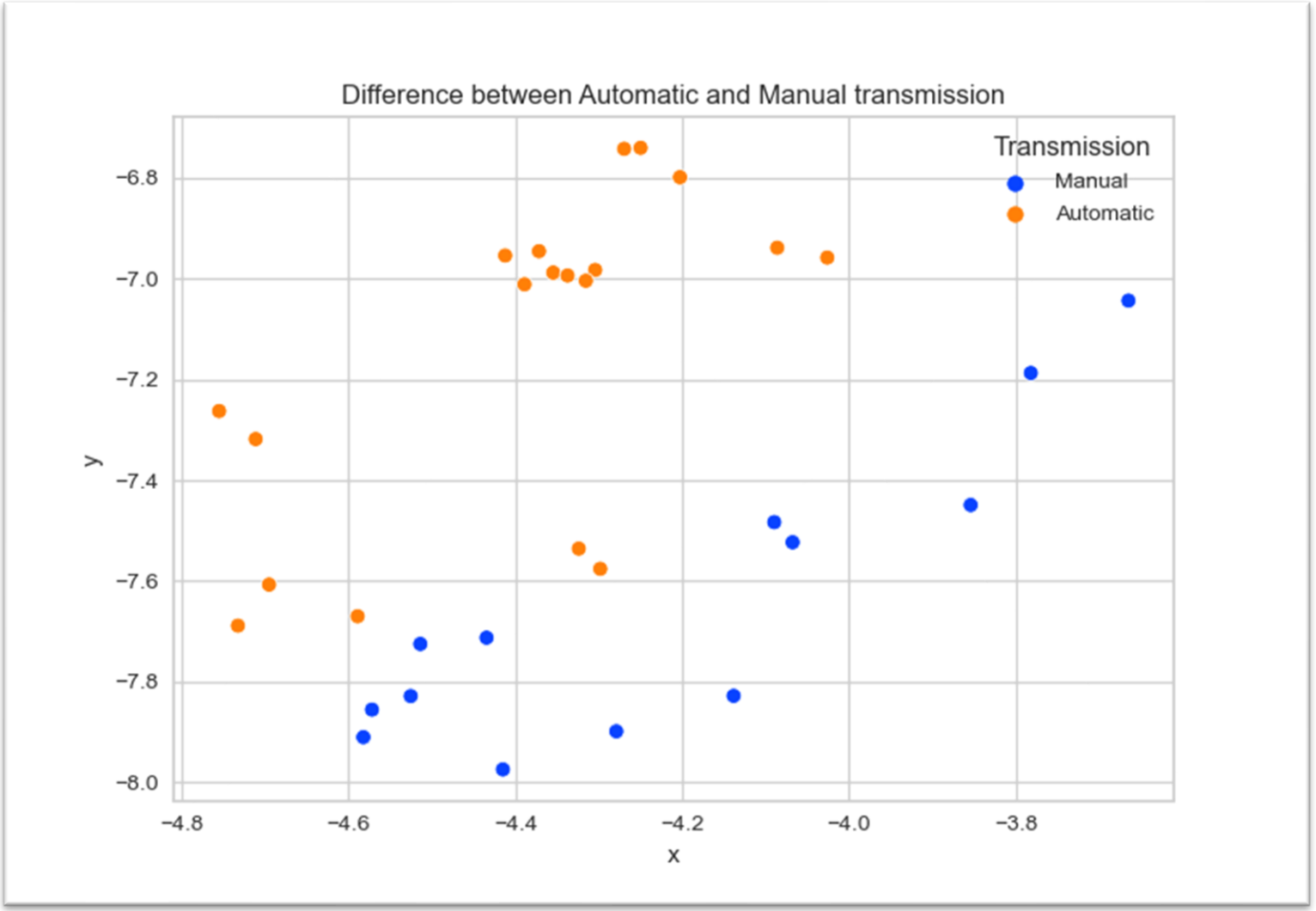
Points do not mix, which means there is a clear difference in classes.

Similarly, heavier cars use more gas. It is clear how data points for glasses group without significant overlap.

Visualizing data using dimensionality reduction (t-SNE)

Using dimensionality reduction technique, we can make our data easier to visualize. Reducing data to two dimensions and then adding a target label makes the results easier to interpret.

Here, we see clear cut difference between classes**. It is almost possible to draw the line that separates the classes**, indicating a linear classification model would be best to use, e.g., Logistic Regression.



Methodology

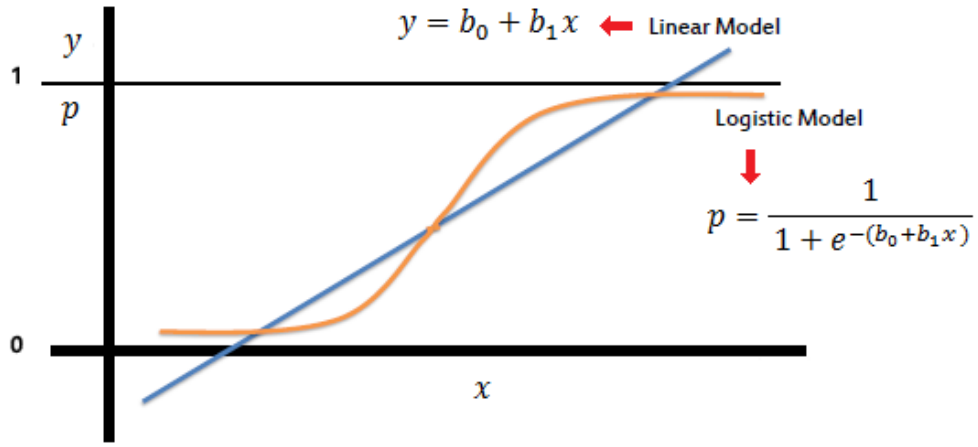
Preprocessing steps

1. Split dependent variable (am) from dependent variables,
2. Drop Name column,
3. Standardize variables with many values, where we want to preserve the distribution (numeric features),
4. Normalize ordinal features.

After preprocessing, we are left with a matrix of data ready for prediction model.

Classifier

For this task, Logistic Regression classifier is used. Since data has clear differences between classes (as seen on scatter plot above). Linear classifier is the best choice. Logistic regression applies Logistic (sigmoid function), and gives probabilities for classes, which is exactly what is needed in this problem.



Model tuning

L1 and L2 Regularization

Regularization are techniques to prevent overfitting. Overfitting occurs when a model learns the training data too well and is unable to generalize to new data. This is an unwanted phenomenon, and we can introduce a penalty to the cost function to prevent it. Cost function is an evaluation metric for the fitted model. It should be as close to 0 as possible. For linear models that is usually the Mean Squared Error.

L1 and L2 regularization are two techniques used to prevent overfitting in logistic regression.

L1 regularization adds a penalty to the sum of the absolute values of the coefficients in the model. This encourages the coefficients to be small or zero, which can help to reduce the complexity of the model. It works as feature selection, because when coefficient drops to zero, that feature is ignored.

L2 regularization adds a penalty to the sum of the squared values of the coefficients in the model. This also encourages the coefficients to be small, but it is less aggressive than L1 regularization. L2 regularization spreads coefficients more evenly.

We can select regularization strength with a hyperparameter. Using grid search technique, we can create several possible hyperparameter combinations, evaluate each one and select the best model.

Using cross-validated grid search, it was found the model performed best with L2 regularization with strength of C = 0.75

Results

After training the model on 22 samples and testing on 10 samples, out of which 6 were of positive class (Manual), model predicted each class correctly.

Accuracy, precision, recall and Area under Roc curve were all 1.

Confusion matrix:

TRUE\PREDICTED	AUTOMATIC	MANUAL
AUTOMATIC	4	0
MANUAL	0	6

Implementation

The model was built in Python programming language.

Libraries used:

- Pandas – data manipulation and analysis
- Numpy – linear transformation and matrices
- Seaborn, matplotlib – data visualization
- Scikit-learn – preprocessing, machine learning and pipelining