SUPPLEMENTARY MATERIALS FOR MACHINE LEARNING MODELS COMPARISON

PREPROCESSING OF THE DATASET

The dataset does not have any missing values. The values of most continuous variables have been normalized; depending on their distribution, standardisation may be applied if necessary.

The distributions of the variables used in modelling, being normalized, do not seem to require further standardisation. Normalization eliminates the effect of magnitude differences across features. Further standardisation is not necessary for Multiple Linear Regression. Windspeed, a feature in the machine learning models, is positively skewed. The count of bike rentals is similarly positively skewed. Log transformation or square root transformation can be used for better predictive analysis.

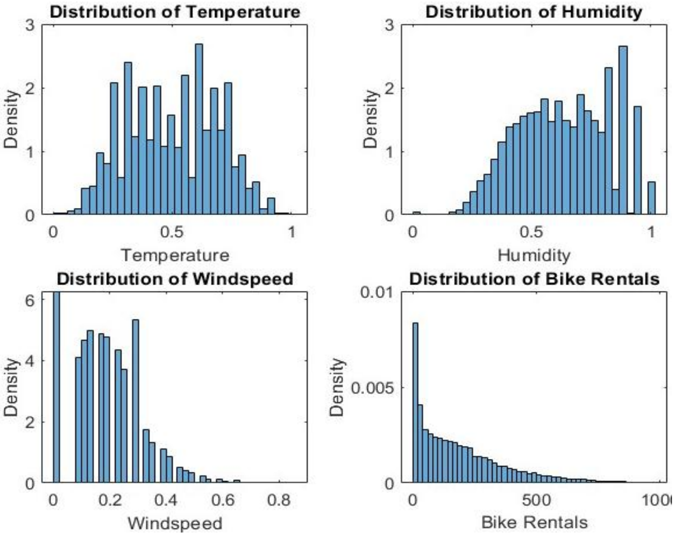


Figure 1: Distributions of variables (Normalised values for Temperature, Humidity and Windspeed)

The distribution of bike rentals by hour experiences peak at 8 a.m. and 5 p.m. (The graph can be seen on the poster). A new binary variable is created to be implemented in the model: 'peak_hour' takes a value of 1 from 8 a.m. to 9 a.m. and from 5 p.m. to 6 p.m. 'season' which has values 1 for winter, 2 for spring, 3 for summer and 4 for autumn, is one-hot encoded before modelling. 'weathersit' is also one-hot encoded for the values 1(Clear, Few clouds), 2(Mist, Cloudy), 3(Light Snow, Light Rain) and 4(Heavy rain, heavy snow, fog).

INTERMEDIATE RESULTS

During the initial analysis of the dataset, the variables below were considered for the machine learning models to predict bike rentals. The model below acts as an experimental one before actually working on developing the most suitable model for the dataset. The independent variables like windspeed, humidity, seasons and weather were selected based on domain knowledge while temperature and hour (peak hours) were selected based on their relationship with the target variable. During training, the model performed well according to performance metrics on training data (R^2 was 0.50 and RMSE was 127.98), however there were signs on overfitting and deficiency in the model. The model would not perform well consistently since sparsity present due to the dummy variables; p-values of various features were insignificant. In the next section, we will highlight how the features were selected for our baseline models.

Table 1: Experimental model

Dependent variable	**Independent variables**
Count	Temperature, Humidity, Windspeed, Season (Dummy), Weather (Dummy), Working Day, Holiday, Peak Hours (Binary)

Since including the dummy variables for 'weather' and 'season' led to rank deficient regression matrix which adversely affect machine precision, it was decided that we will run the regression using only temperature, humidity and windspeed initially and add other variables one at a time to the model until no further improvement in R^2 and RMSE is observed. The model yielding the highest R^2 will be the baseline model. The MLR model with only the three features has a value of 0.246 which suggests low predicting power. A slight improvement to 0.499 is observed when the variable for months and peak hours are included. The dummy variables created for weather and season together in the model do not improve the model by much (R squared is 0.5). The dummy variables were removed from the model; the variable month was included since seasonal trend can still be observed and the weather can be predicted for that month; Table 2 shows the MLR model used as a baseline, R squared of 0.51 and adjusted R squared of 0.509 are observed.

Table 2: MLR Baseline Model (Model 1)	
Dependent variable	**Independent variables**
Count	Temperature, Humidity, Windspeed, Month (Categorical), Peak Hours (Binary), Working Day

IMPLEMENTATION DETAILS AND RESULTS

For all models, the dataset is split into a 70% for training and 30% for testing, since we have a large number of records (17000 rows), we will have enough data for both training and testing without over-reliance on either. A cross-validation of 10-fold is applied on the training set to evaluate the effectiveness of model; each fold contains 10% of the data for validation, leaving 90% for training. The model would be tested on each datapoint which should make it more robust.

After the baseline MLR model was established, different variations were tested to identify the most suitable MLR model for comparison. The performance metrics of the model ran have been summarised in the table

Figure 1: Feature Coefficients (Interaction Terms)

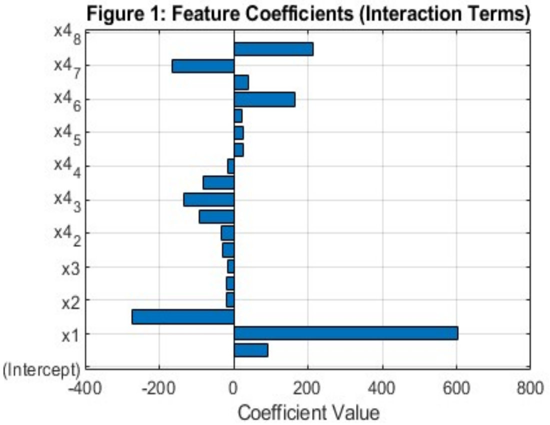
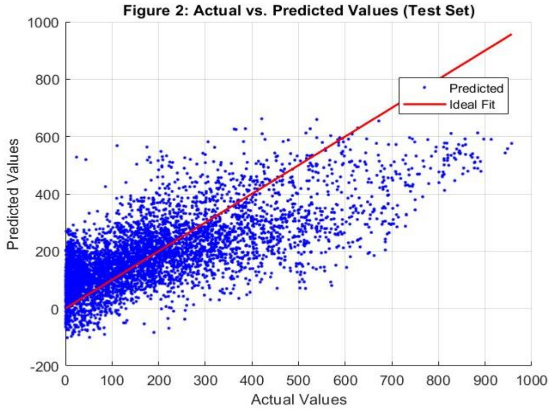


Figure 2: Actual vs. Predicted Values (Test Set)



In the second MLR model, the dependent variable is log-transformed to see if there is any improvement since a positive skewness was observed in its distribution. In the third model of MLR, interaction terms are introduced to identify underlying relationship between the features. Stepwise regression is run to find the most significant interaction terms: ‘temperature * workday and ‘temperature * peak_hour’ are found to have the lowest p-value. These two are implemented in model 3. The coefficients are visualized in Figure 1.

The fourth model being tested is a Ridge Regression one which aims at reducing multicollinearity in the model, shrinking coefficients towards zero, reducing the weight of each independent variable output (J.M & Kavlakoglu, 2023). Lasso, Least Absolute Shrinkage and Selection Operator, is another regularization method that addresses overfitting and poor performance from the model (Ranstam & Cook, 2018). The implementation of

Lasso regularization yielded very strange results; I struggled to implement this regularization technique on MATLAB. The last MLR model that is tested is a polynomial one where the variables: humidity and windspeed are squared which led to overfitting. Figure 2 shows the predicted values of polynomial regression. This version on MLR struggled at higher values (wider spread) and introduced significant complexity with poor generalization.

Models	RMSE (Training)	R-squared (Training)	RMSE (Testing)	R-squared (Testing)
MLR 1 (Baseline MLR model)	126.85	0.507	127.23	0.513
MLR 2 (Log transformed)	1.10	0.400	1.09	0.402
MLR 3 (Interaction terms)	**125.70**	**0.516**	**125.49**	**0.526**
MLR 4 (Ridge Regularization)	180.81	-	183.98	-0.017
MLR 5 (Polynomial Regression)	127.98	0.499	128.08	0.506
RFR 1 (Baseline RFR model)	132.36	0.465	132.85	0.469
RFR 2 (Hyperparameters tuned)	**126.44**	**0.512**	**127.15**	**0.514**

For the baseline random forest regression model, 100 trees and 10 splits have been used. K-fold cross validation is used on the random forest regressions since the results will be compared with multiple linear regression which is not a 'bagging' model. Out-of-bar has been considered as a metric for comparison between random forest models, but this would lead to higher computational cost, ultimately not needed for the comparisons between the two machine learning models. The next step in the experimental analysis for random forest regression is to use grid search for optimisation. The hyperparameters being tuned are number of trees ('NumLearningCycle'), tree depth('MaxNumSplits') and the minimum sample required to create each leaf node ('MinLeafSize'). Other hyperparameters such as feature reduction have not been included to minimise computational power. Grid search is done by an algorithm looping through combinations of hyperparameters, the results are recorded and the combination with the highest R^2 is used in the next random forest regression model. In figure 3, the top five hyperparameter combinations can be seen where R^2 is around 0.5. The best combination based on R^2 is 100 trees, 30 splits and 10 for sample size for each leaf node. Random Forest regression is run again with those hyperparameters, and the results are compared with the multiple linear regression model with interaction terms.

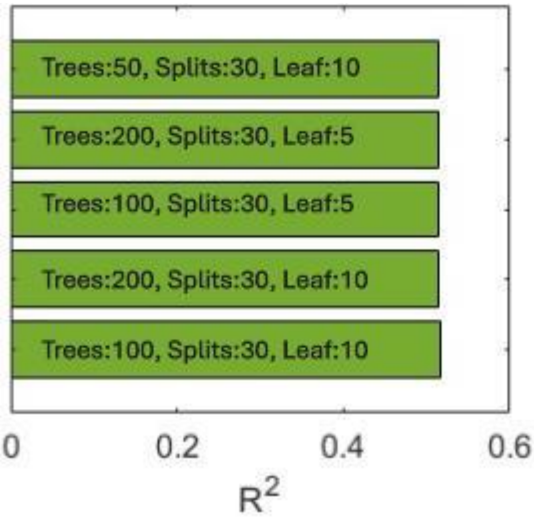


Figure 3: Top 5 Hyperparameter combinations

Time-based cross validation may be a more suitable validation process than k-fold for this dataset. Most research papers where this dataset has been utilised involved some time series analysis. Time-based cross validation has not been implemented due to insufficient time and since most of the work had already been done using k-fold cross validation. Time-based cross validation should be considered for any future work with this dataset to avoid data leakage especially when using past data to predict future.

GLOSSARY

Multiple Linear Regression: A regression technique where the relationship between one dependent variable and multiple independent variables are modelled in a linear equation.

Random Forest Regression: An ensemble machine learning method where predictions are made using the average of the output of multiple decision trees.

Support Vector Machine Regression: A regression technique where a hyperplane is identified in a high-dimensional space to minimize prediction errors.

Anchor Regression: A regression technique where anchor variables are introduced to stabilize predictions.

K-fold cross validation: A validation method where the dataset is split into a number of subsets K and model is trained on K-1 subsets and then tested on the remaining one iteratively.

Time-based cross validation: A validation method where temporal structure is respected by training on past data and testing on future data to simulate real-world situations.

Grid search: A hyperparameter optimization technique that continuously search through specified parameter grid to identify the best combination for a model.

Maximum Splits: Parameter that refers the depth of a decision tree to avoid overfitting.

Minimum Leaf Size: Minimum number of datapoints required in a leaf node of a tree.

R-squared: Statistical measure that explains the proportion of the dependent variable explained by the independent variables in the model.

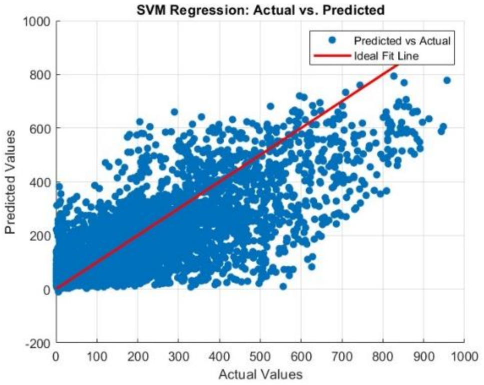
Root Mean Squared Error: A metric that measures the average magnitude of errors in predictions (square root of mean squared error (MSE)).

REC Curve: Also known as Regression Error Characteristic Curve is a version of the ROC curve but for regression models. It shows the cumulative distribution of prediction errors.

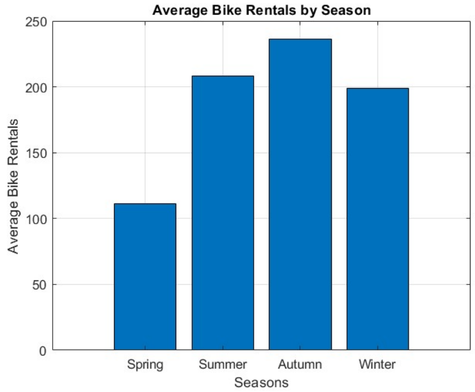
REFERENCES

Ph.D., J.M. and Kavlakoglu, E. (2023) What is ridge regression? IBM. Available at: https://www.ibm.com/topics/ridge-regression (Accessed: 25 November 2024).

Ranstam, J. and Cook, J.A. (2018) 'Lasso regression', British Journal of Surgery, 105(10), pp. 1348–1348. doi:10.1002/bjs.10895.



Appendix 1: SVM Regression Results



Appendix 2: Bike Rentals by season