DATA423 Assignment 3

FIND THE BEST PERFORMING REGRESSION MODEL

Nicole Dunn | 75958138 | 30/09/2022

Data description

The data consists of 920 observations of 21 variables. The outcome variable is called "Response". Evaluating the numeric data shows that there are some outliers but when the IQR multiplier is set to 2.6 these outliers disappear so are not of great significance.

The overall missingness of the data is fairly minimal, there are no excessively missing variables or observations. There are 6 predictor variables with no missingness and these are "ObservationDate", "BloodType", "Alcohol", "Coffee" and "Exercise". The missing values visually appear random, with no apparent meaning.

There are 3 groups observed in a correlation chart. Two of those are larger groups, one consisting of 6 predictor variables and the response variable with very high positive correlation and the other having more of a mix between high and low positive correlation between the 6 predictor variables. The variables within each of these groups are similarly named having "Reagent" in the name. The last group is slightly smaller with not as strong of a correlation to any other variables, but the two high high correlation also have similar names starting with "Reagent". The "Response" variable is the only variable to show strong negative correlation with another variable, this is with "Exercise", and there is some medium-negative correlation with the second group and the outcome variable.

There is one nominal variable, "BloodType" which has a low cardinality as there are only 4 main groups of blood types for humans "A", "B", "AB" and "O", and RhD has not been measured in the data to slightly inflate the cardinality.

There is a variety of scales within the numeric data. "Alcohol" is 0-4, "Coffee" is 0-7, "Exercise" is 0-12, "ChemoTreatments" is 0-5 and the reagents sit between 0-1000.

The "ObservationDate" is correctly formatted and the format is in the universal format of YYYY-mm-dd.

Strategies

MISSING DATA:

As mentioned earlier, there are no excessively missing variables or observations to discard.

Due to the Rpart tree having multiple nodes, we can rule out "Missing Completely at Random" and so we cannot use partial deletion as this would generate bias.

For the methods tried, we employed kNN (neighbours = 5) imputation, this was an acceptable approach for most methods as it produced very reasonable results. The best model ended up using bag imputation as this reduced the RMSE from 402.14 to 397.71.

There was no missingness within the response variable and this means we do not need to discard any observations related to that type of missingness.

OUTLIERS:

Since the outliers were of little concern, all observations are to be retained. Some robust methods were attempted but they provided little improvement to the results and other methods were chosen for further exploration.

FEATURE ENGINEERING:

Centering and scaling was chosen to better compare the models as the variety of results from the numeric data needed an easier way to be modelled. Imputing the data before centering and scaling improved the RMSE minimally as well.

Missing values were dealt with using the imputation method as we cannot use partial deletion. The imputation methods were adjusted according to the best model.

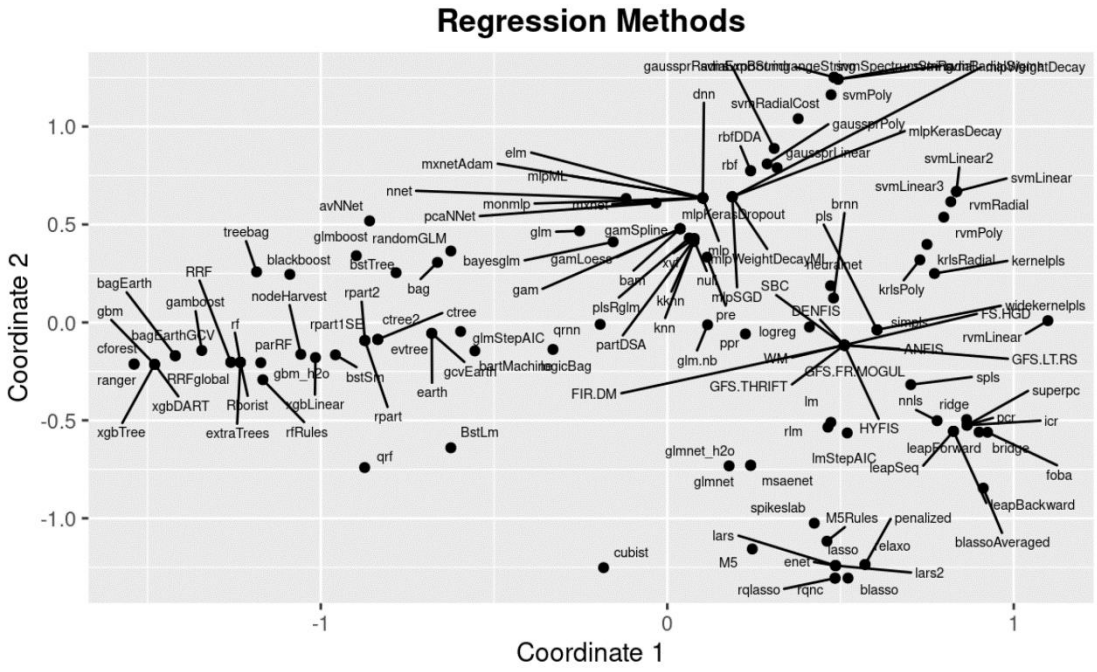
The variable "BloodType" was treated with dummy encoding as it had low cardinality. Other than removing the date variable from the data set in certain models, no other data was changed for the methods.

TUNING & ASSESSING

The test set was 20% of the entire data set, this was held aside for the models to use so that the best model can be chosen. The best model was selected using stratified sampling based on the "Response" variable. The hyper-parameters were tuned using 25 bootstrap re-samples of the training data to produce the best estimation of them.

METHODS:

When choosing methods to try, the strategy was based on the graph introduced in the tutorials. Visually I tried to choose methods evenly throughout the graph to get the greatest chance of getting the best model. Some of the models chosen were kNN, Spikeslab, Cubist, bagEarth, QRF and NNet, as seen in the below image these are visually spread out. The other strategy was by using prior experience and comfort with certain models used in previous projects, this allowed another degree of exploration as well.



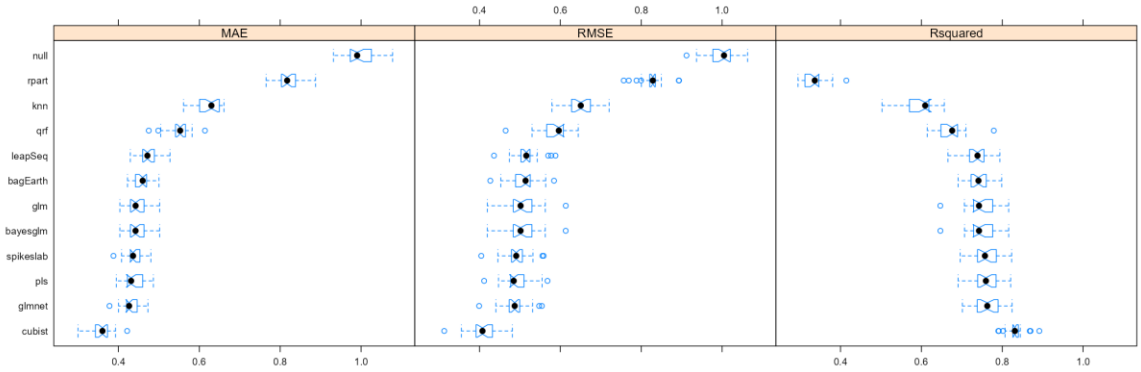
Trials:

Method	Description	Notes	Reason chosen
avNNet - Model Averaged Neural Network	Neural Network, Ensemble Model, Bagging, L2 Regularization, Accepts Case Weights	Very slow to train, But very simple to implement	Wanted to learn more about Neural Networks
NNet - Neural Network	Neural Network, L2 Regularization, Accepts Case Weights	Very slow to train	Wanted to learn more about Neural Networks
QRF	Random Forest, Ensemble Model, Bagging, Implicit Feature Selection, Quantile Regression, Robust Model	Quick to train, easy to understand with clear graphing	Completely Random, was at the bottom of the above graph
GLM	Generalized Linear Model, Linear Classifier, Two	Very quick to train	Completely Random, was in the middle of the above graph

	Class Only, Accepts Case Weights		
BayesGLM	Generalized Linear Model, Logistic Regression, Linear Classifier, Bayesian Model, Accepts Case Weights	Very quick to train	Completely Random
LeapSeq	Linear Regression, Feature Selection Wrapper	Medium length to train, has interesting output	Completely Random, was on the right edge of the above graph
bagEarth	Multivariate Adaptive Regression Splines, Ensemble Model, Implicit Feature Selection, Bagging, Accepts Case Weight	Long to train, had some difficulty with the code implementation	Completely Random, was on the left edge of the above graph
Cubist	Rule-Based Model, Boosting, Ensemble Model, Prototype Models, Model Tree, Linear Regression, Implicit Feature Selection	Quick to train after crashing a couple of times	Like QRF was separate from the other models, so was curious about the method
kNN	Prototype Models	Very simple and clean	Wanted a kNN method for comparison
Spikeslab	Linear Regression, Bayesian Model, Implicit Feature Selection	Quick to train	Completely Random, was on the bottom right edge of the above graph

Models:

The following models were successfully trained, a visual summary is shown below. Only two models performed worse than the null model and thus have been omitted from the below visualization.



Model	**Processing Steps**	**Hyper-parameters**	**Resampled Performance**
rpart	"center", "scale", "dummy"	Cp: 0.03	RMSE: 798.05 R^2: 0.34 MAE: 627.84
Knn	"impute_knn", "center", "scale", "dummy"	K = 13	RMSE: 630.45 R^2: 0.60 MAE: 476.99
QRF	"impute_knn", "center", "scale", "dummy"	Mtry: 12.00	RMSE: 565.77 R^2: 0.67 MAE: 420.40
Leapseq	"impute_knn", "center", "scale", "dummy"	Nvmax: 6	RMSE: 499.22 R^2: 0.74 MAE: 364.28
bagEarth	"impute_knn", "center", "scale", "dummy"	Degree: 1.00 Nprune: 12.00	RMSE: 494.10 R^2: 0.74 MAE: 351.21
GLM	"impute_knn", "center", "scale", "dummy"	None	RMSE: 488.91 R^2: 0.75 MAE: 342.62
BayesGLM	"impute_knn", "center", "scale", "dummy"	None	RMSE: 488.81 R^2: 0.75 MAE: 342.56
Spikeslab	"impute_knn", "center", "scale", "dummy"	Vars: 16.00	RMSE: 475.68 R^2: 0.76 MAE: 337.67
PLS	"impute_knn", "center", "scale", "dummy"	Ncomp: 9.00	RMSE: 476.21 R^2: 0.76 MAE: 335.26
GLMnet	"impute_bag", "center", "scale", "dummy"	Alpha: 0.32 Lambda: 10.81	RMSE: 471.67 R^2: 0.76 MAE: 331.51

Cubist	"impute_bag", "center", "scale", "dummy"	Committees: 20.00 Neighbours: 9.00	RMSE: 397.71 R^2: 0.83 MAE: 274.08

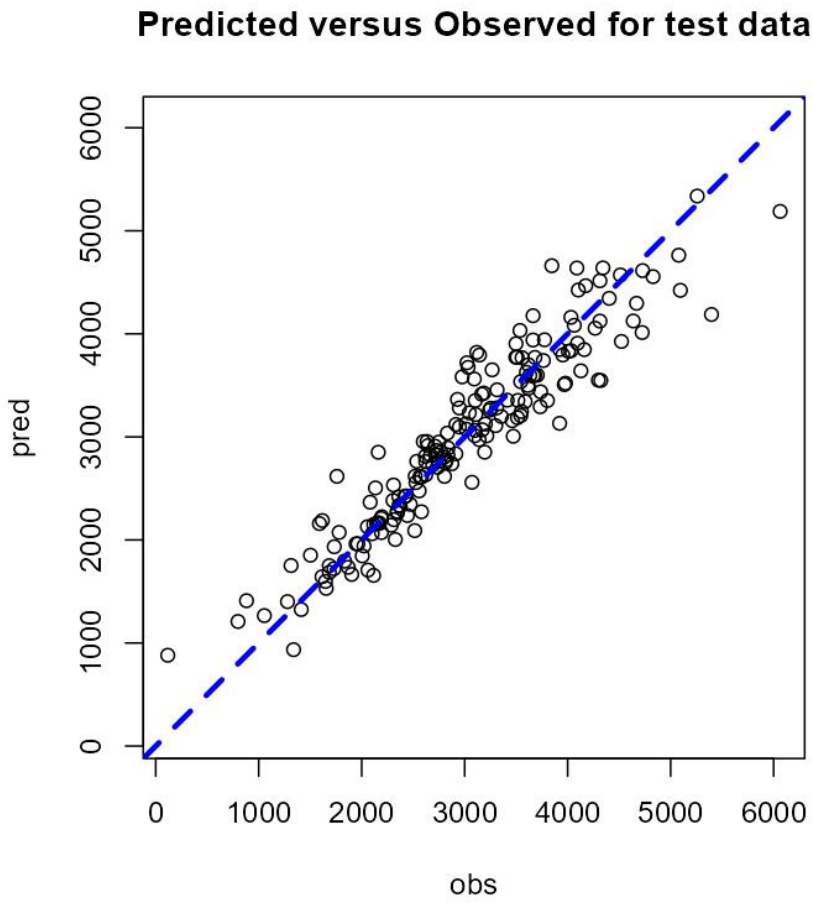
There was a very clear jump in RMSE between the first- and second-best models, which made determining the best model very easy.

Best Model:

Based on RMSE, the best model is **Cubist**, and it is significantly better than the **GLMnet** model by nearly a full 100 difference.

PERFORMANCE ON UNSEEN DATA:

When the test data was predicted using the best model, it generated the following graph.

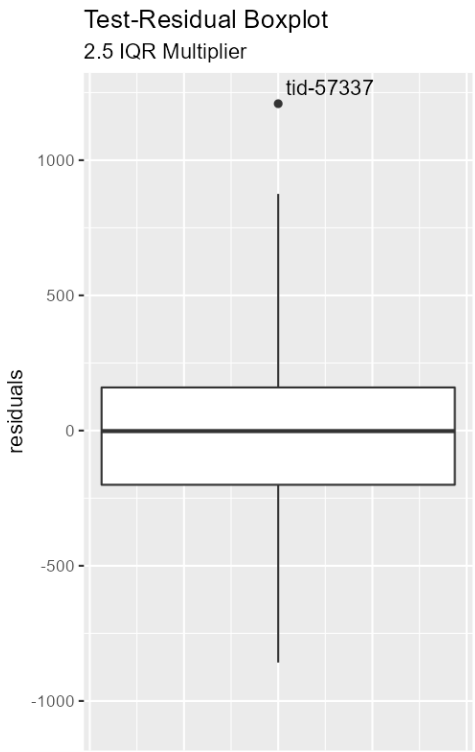


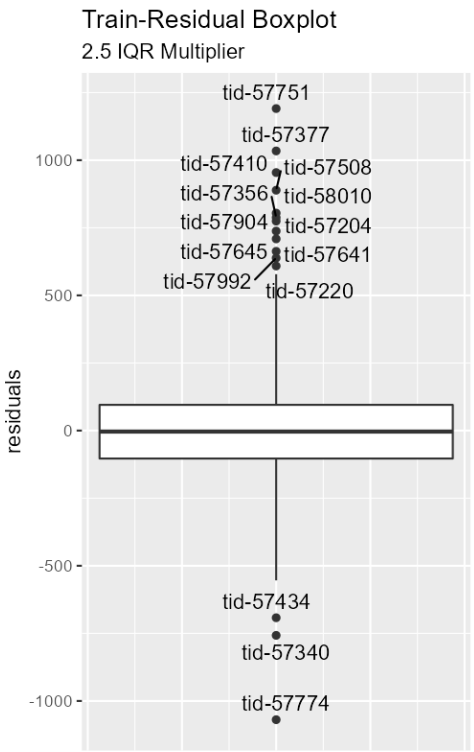
The performance for the prediction using the Cubist method is as below.

Test Metric	Value
RMSE	326.926
MAE	0.886
R^2	239.890

OBSERVATIONS THAT DO NOT FIT THE MODEL:

The model based outliers with an IQR multiplier of 2.5 are observed below:

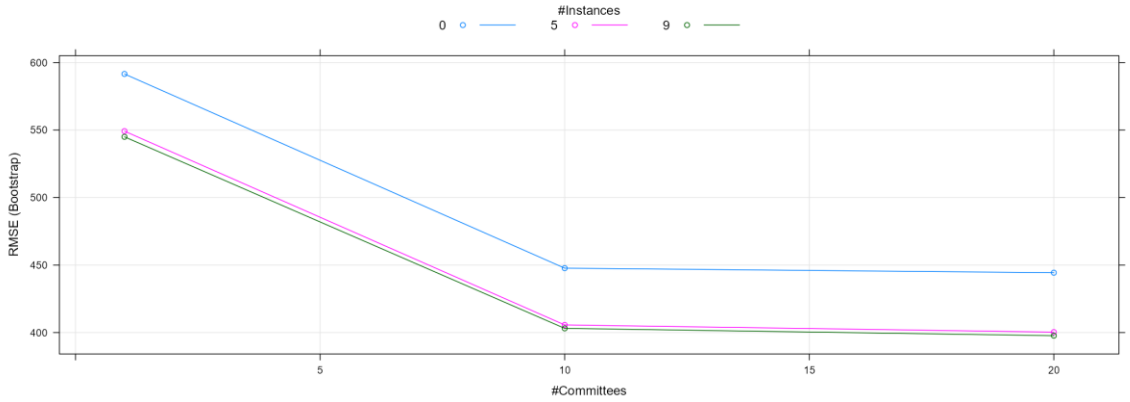




METHOD DESCRIPTION:

The best model uses the Cubist method.

The Cubist method is a rule-based model which is an extension of Quinlan's M5 model tree (Cubist Regresion Models, n.d.). It works by using a linear regression model at the terminal node of the tree and generating a prediction. The prediction is then smoothed by gathering the nearest neighbours and generating an average of the training set, this occurs throughout all the nodes of the tree. Rules are then developed, and the tree is then either pruned or combined for simplication. The establishment of rules is independent of the choice using instances.



This image displays the relationship between the model performance and the two tuning parameters, as estimated using bootstrapping.

Transparency:

As the Cubist model is part of the CART labelling methods, it does provide some global transparency. There is some difference between Cubist models and other CART models such as an optional boosting-like procedure called committees and the predictions generated by the model rules can be adjusted using nearby points from the training data set.

References

Cubist Regresion Models. (n.d.). Retrieved from r-project: https://cran.r-project.org/web/packages/Cubist/vignettes/cubist.html