Aalto-yliopisto
Aalto-universitetet
Aalto University

22.3.2023

BorgWarner Friction Plate Validation Testing Case

TU-E5030 – Creating Value with Analytics D

Tuomo Salo

Executive summary

Six machine learning prediction models were evaluated based on their accuracy, and Random forest -model was determined to be the most efficient in determining which parts would fail the pre-test process in the new factory. This model predicted about 95 % of the negatives correctly. The models feature importance list and its decision tree graphs help visualize the required parameters to non-technical employees. The model should be used instead of the engineer-led heuristic parameter setting process. Finally, the most impactful variables were determined by the model to be material variable mat.2 and process variable var.7. Most straight-forward action to pass almost all parts is to ask suppliers for materials with specification var.2 to be over 8.08.

1. Applying different machine learning models to the problem

The process of manufacturing friction plates at BorgWarner can be analysed with multiple variables related to its manufacturing. These are categorized as process settings, environmental conditions in production, and data on input materials. Table 1 depicts key values obtained from the models' predictions, where accuracy means true predictions divided by all the data points in the data, and AUC measures the true positive rate against false positive rate.

Table 1. Models' accuracy and area under ROC curve.

Model	*Accuracy*	*AUC*
Decision trees	0.891	0.972
Gradient boosting	0.860	0.947
Random forest	0.844	0.979
SVM	0.844	0.908
Neural network	0.828	0.836
Logistic regression	0.750	0.817

The results are fairly even but notably decision tree model achieves highest accuracy of correct predictions and best identifies true positives compared to false positives (i.e., AUC rate). Decision trees also has the highest recall (0.852) value and very high precision (0.885) value ("Sensitivity" and "Pos Pred Value", Appendix 1.12.) Recall portrays which percentage of actual positives the model identified correctly, and precision tells how many of the predicted positives were actually positive. Decision tree also has the highest 95 % confidence interval lower bound at 0.788 which makes it the most robust model here.

Random forest trails close behind decision trees and has different value in its key statistics (Appendix 1.1). Its recall was 0.667 but precision was very high at 0.947 which means that positives identified by this model are at a 95 % rate actually positive. Therefore, high confidence in its positive predictions comes at the cost of missing one third of the actual positives as the model is more careful at picking the positives.

Gradient boosting offers a middle ground between the two previously discussed models. It has a recall value of 0.740 and precision of 0.909 which means it is intended more towards picking the safe positives while missing more of the true positives (Appendix 1.7). It is slightly better at identifying both negatives and positives compared to random forest.

In regard to identifying most negatives, random forest model is the best performer with a specificity value of 0.973. Next is gradient boost with 0.943 specificity and third is decision tree with the value of 0.912. Specificity tells the percentage of all negatives found.

Other models have proven to be unviable with this dataset. They do not possess any meaningful positive sides compared to the best performers: Decision trees, Gradient boosting, or Random forest -models. Notably, logistic regression performs clearly the worst (Appendix 1.10) even though hyperparameters were iterated to improve its performance.

2. Model suitability selection

Based on the previous chapter, the models can be categorised by the type of prediction accuracy needed:

1. Decision trees model provides the highest identification rate of the true positives and true negatives, which makes it the choice when as many as possible positive cases need to be caught even though some of them are actually negatives.
2. Random forest model identified positive rates which were correct in 95 % of the time and is best suitable when all the identified cases need to be actually positive. This can be case when corrective actions are costly and should not be wasted on false positives. This model also catches the most negative cases.
3. Gradient boosting is in the middle of the aforementioned models. It should be used when false positives can be tolerated more than with random forests, but more positive cases need to be correctly identified.

In this case, the most important goal is to identify the parts that would not pass the pre-test on the first try. Therefore, we aim to maximize the amount of correctly predicted negatives. Additionally, since all parts need to be tested by customer requirements anyway, the number of false positives is not important to the model performance.

Random forest model identifies the 97 % of the negative cases and is the first model that we look closer. The interpretability of the model needs to be considered as it is valuable to see the weights of variables assigned by the model, so that improvement targets on the manufacturing floor can be easily identified. Random forest can output feature importances and its classifying process can be visualized neatly with tree graphs.

Feature importances of random forest are plotted in Appendix 1.14, where two variables var.7 and mat.2 emerge as the most important. Curiously, these two are the same variables that Gradientboost had also identified to be the most impactful. Additionally, tree based models are easy to visualize when depth is kept reasonable (Figure 1). To summarize, random forest is a great model to help determine the bottleneck in part test fails.

Random forest is highly flexible and its variance-bias trade-off can be iterated by changing its parameters, such as maximum number of nodes. In my tests, maximum nodes of 10 produced low variance and accurate results with different random seeds of the training data. This low number of nodes also produces reasonably readable tree visualizations, which helps anyone understand and interpret the model's results, which was required of the tool.

3. Recommendation on using prediction models

The random forest prediction model is robust and reliable in predicting which variables cause increasing fail-rates of the part testing process. The model should be utilized in determining threshold values of different material and process variables to identify targets for improvement. It should smooth the learning curve difficulties in moving to the new production facility. The model also has uses in the main factory in Germany and could be utilized in other processes as well.

The model should be used to identify key factors from the variables to improve process quality and testing pass-rate. It is useful in determining key causations of the form “when material quality x is 10, set y to 20”. This streamlines the headwork and replaces heuristic model of determining complex causations and dependencies between variables, which has been ineffective.

In particular, the model identified process var.7 and material characteristics mat.2 as the most important features affecting the pass-rate of testing. The new factory should focus on determining the relationships of these variables in the process and find the optimum values that maximize testing success-rate. Predicted exact values are investigated in chapter 4 on the next page.

However, tree-based algorithms fail to consider the linear relationships between variables. The model implements discrete threshold values which recursively partition the data into smaller subsets further down the tree. Luckily in this case the limitation did not have impact on the usefulness of the model, but it could play a role later when the most prominent corrective actions have been completed.

Tree visualization graphs help to understand relationships between different variables in an intuitive way (see Appendix 1.15 and Figure 1) and allows anyone to grasp the reasoning of new parameters. Additionally, model should be constantly fed with updated data from the testing rounds and used to print updated tree visualizations to gain new insight and cumulate the improvement process.

4. Identified corrective actions

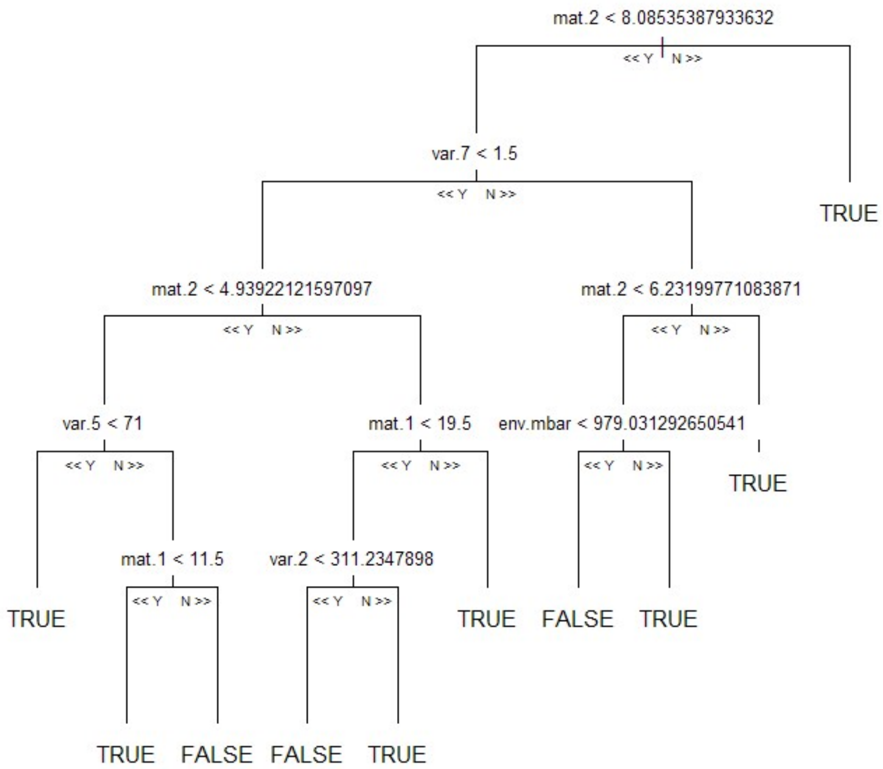


Figure 1. Threshold variable values of Random forest visualizing tree when maxnodes = 10.

Figure 1 illustrates the models interpretation of the causal effects of the testing process. Investigating it provides insight on what variables to tweak in the process and what material characteristics have most influence on the pass-rate.

Looking at the model's decision making process, it is clear that material variable mat.2 needs to achieve value over 8.08 to pass every time. If this is not possible because of supplier limitations, the supplier should make the parameter at least over 6.23.

Whenever mat.2 quality is under 8.08 and over 6.23, var.7 should be set to action requires the manufacturing process to achieve var.7 quality of at least 1.5 which should be the main target at first.

If supplier material quality mat.2 cannot be kept at the level of 6.23 and above, additional processes var.5 and var.2 need to be developed as a secondary focus. The var.5 is more significant process according to Appendix 1.15 importances.

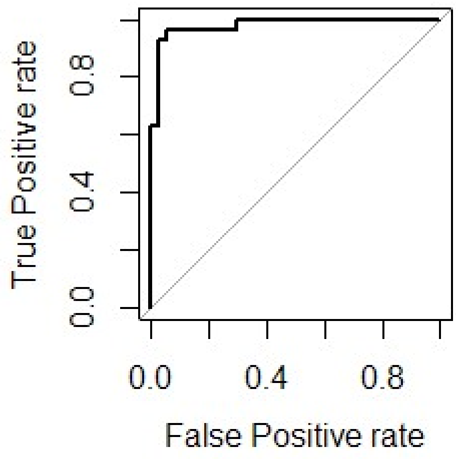
Table 2.List of corrective actions to improve pre-test pass-rate

Priority	Action	Target value
1.	Demand specific quality from suppliers	mat.2 > 8.08 (secondary > 6.23)
-	Manage process 7 based on material 2	var.7 >= 1.5, when mat.2 > 6.23
-	Manage process 5 based on material 2	var.5 < 71, when mat.2 < 4.94
-	Manage process 2 based on material 1	var.2 > 311.2, when mat.1 < 19.5

Appendix 1

```
                 Reference
Prediction FALSE TRUE
     FALSE    36    9
     TRUE      1   18

               Accuracy : 0.8438
                 95% CI : (0.7314, 0.9224)
    No Information Rate : 0.5781
    P-Value [Acc > NIR] : 5.021e-06
                  Kappa : 0.6663
 Mcnemar's Test P-Value : 0.02686
            Sensitivity : 0.6667
            Specificity : 0.9730
         Pos Pred Value : 0.9474
         Neg Pred Value : 0.8000
             Prevalence : 0.4219
         Detection Rate : 0.2812
   Detection Prevalence : 0.2969
      Balanced Accuracy : 0.8198
```

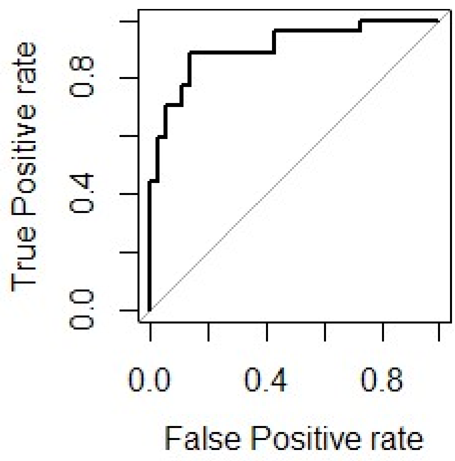
Appendix 1.1. Random forest Confusion matrix and statistics.



Appendix 1.2. Random forest ROC curve. Area under curve is 0.979.

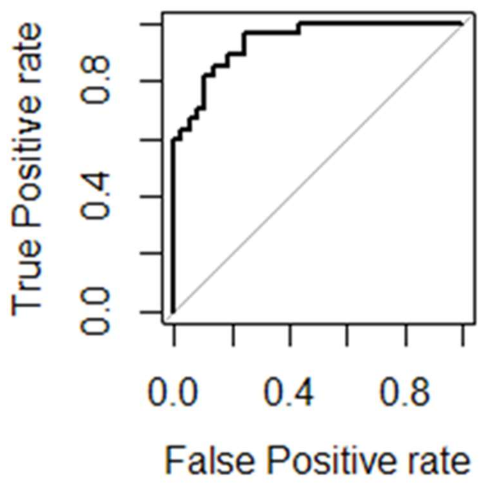
```
                 Reference
Prediction FALSE TRUE
     FALSE    33    6
     TRUE      4   21

               Accuracy : 0.8438
                 95% CI : (0.7314, 0.9224)
    No Information Rate : 0.5781
    P-Value [Acc > NIR] : 5.021e-06
                  Kappa : 0.6764
 Mcnemar's Test P-Value : 0.7518
            Sensitivity : 0.7778
            Specificity : 0.8919
         Pos Pred Value : 0.8400
         Neg Pred Value : 0.8462
             Prevalence : 0.4219
         Detection Rate : 0.3281
   Detection Prevalence : 0.3906
      Balanced Accuracy : 0.8348
```

Appendix 1.3. Support Vector Machine Confusion matrix.



Appendix 1.4. SVM ROC curve. Area under curve is 0.908.

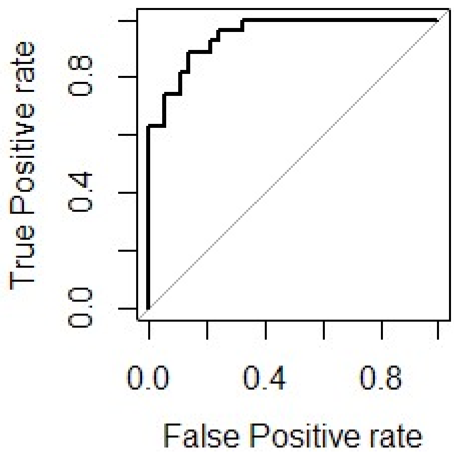
```
                 Reference
Prediction FALSE TRUE
     FALSE     33    7
     TRUE     4     20

               Accuracy : 0.8281
                 95% CI : (0.7132, 0.911)
    No Information Rate : 0.5781
    P-Value [Acc > NIR] : 1.868e-05
                  Kappa : 0.6423
Mcnemar's Test P-Value  : 0.5465
            Sensitivity : 0.7407
            Specificity : 0.8919
         Pos Pred Value : 0.8333
         Neg Pred Value : 0.8250
             Prevalence : 0.4219
         Detection Rate : 0.3125
   Detection Prevalence : 0.3750
      Balanced Accuracy : 0.8163
```



Appendix 1.5. Neural network *Machine Confusion matrix.*

```
          Reference
Prediction FALSE TRUE
     FALSE    35    7
     TRUE      2   20

               Accuracy : 0.8594
                 95% CI : (0.7498, 0.9336)
    No Information Rate : 0.5781
    P-Value [Acc > NIR] : 1.207e-06
                  Kappa : 0.7043
 Mcnemar's Test P-Value : 0.1824
            Sensitivity : 0.7407
            Specificity : 0.9459

         Pos Pred Value : 0.9091
         Neg Pred Value : 0.8333
             Prevalence : 0.4219
         Detection Rate : 0.3125
   Detection Prevalence : 0.3438
      Balanced Accuracy : 0.8433
```

Appendix 1.6. Neural network ROC curve. Area under curve is 0.936.



Appendix 1.7. Gradient boosting *Confusion matrix.*

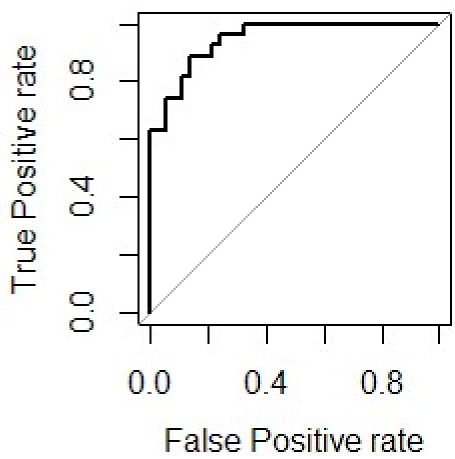
Appendix 1.8. Gradient boosting ROC curve. Area under curve is 0.947.

```
var.7        var.7 33.0143068
mat.2        mat.2 23.7011764
env.mbar env.mbar 10.6047375
var.6        var.6  5.8255896
var.1        var.1  5.2262039
var.2        var.2  4.5100962
mat.3        mat.3  4.0878032
var.9        var.9  4.0471233
var.3        var.3  2.9678389
env.temp env.temp  1.5432630
env.hum   env.hum  1.5228273
mat.1        mat.1  1.0244900
var.5        var.5  0.8505562
var.8        var.8  0.7843917
var.4        var.4  0.2895962
```

Appendix 1.9. Gradient boosting variable importances.

```
          Reference
Prediction FALSE TRUE
     FALSE    31   10
     TRUE      6   17

               Accuracy : 0.75
                 95% CI : (0.626, 0.8498)
    No Information Rate : 0.5781
    P-Value [Acc > NIR] : 0.003221
                  Kappa : 0.477
 Mcnemar's Test P-Value : 0.453255
            Sensitivity : 0.6296
            Specificity : 0.8378
         Pos Pred Value : 0.7391
         Neg Pred Value : 0.7561
             Prevalence : 0.4219
         Detection Rate : 0.2656
   Detection Prevalence : 0.3594
      Balanced Accuracy : 0.7337
```

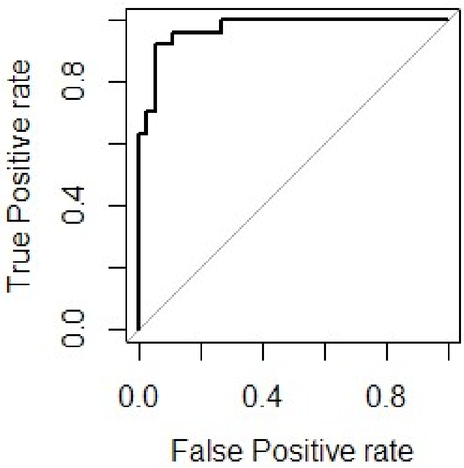
Appendix 1.10. Logistic regression *Confusion matrix.*



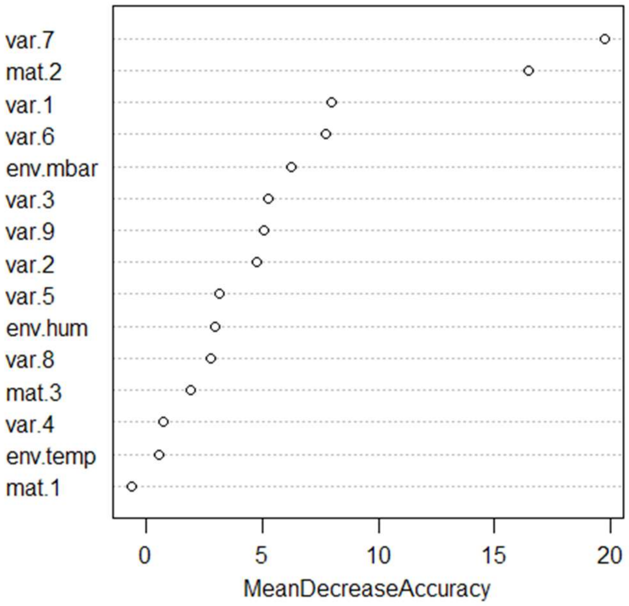
Appendix 1.11. Logistic regression ROC curve. Area under curve is 0.817.

```
               Reference
Prediction FALSE TRUE
     FALSE    34    4
     TRUE      3   23

               Accuracy : 0.8906
                 95% CI : (0.7875, 0.9549)
    No Information Rate : 0.5781
    P-Value [Acc > NIR] : 4.795e-08

                  Kappa : 0.7746
 Mcnemar's Test P-Value : 1

            Sensitivity : 0.8519
            Specificity : 0.9189
         Pos Pred Value : 0.8846
         Neg Pred Value : 0.8947
             Prevalence : 0.4219
         Detection Rate : 0.3594
   Detection Prevalence : 0.4062
      Balanced Accuracy : 0.8854
```

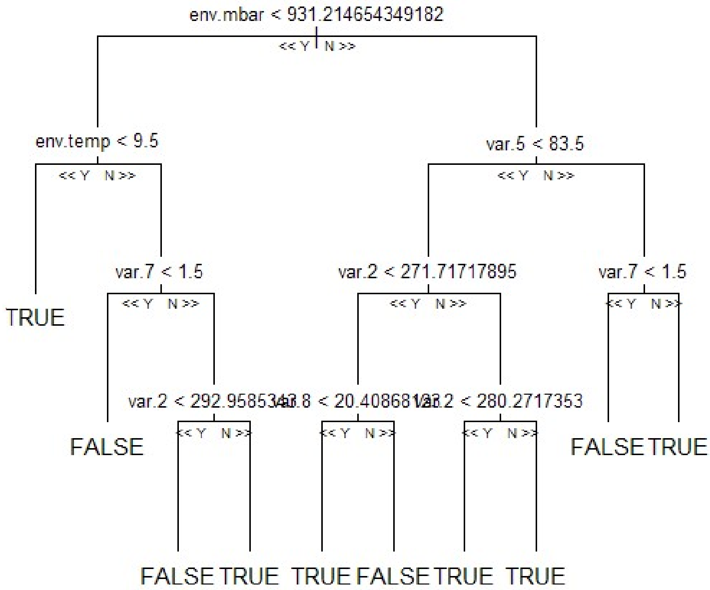
Appendix 1.12. Decision tree *Confusion matrix.*



Appendix 1.13. Decision tree ROC curve. Area under curve is 0.972.



Appendix 1.14. Random forest variable importances.



Appendix 1.15. Random forest variable importances when maxnodes = 5.