Data 624 Homework 9 Chapter 8 Regression Trees and Rule-Based Models

Enid Roman

2024-11-12

8.1. Recreate the simulated data from Exercise 7.2:

(a) Fit a random forest model to all of the predictors, then estimate the variable importance scores:

	Overall
V1	8.6053659
V2	6.8312592
V3	0.7415349
V4	7.8833841
V5	2.2447503
V6	0.1360542
V7	0.0559509
V8	-0.0681958
V9	0.0031962
V10	-0.0547059

Did the random forest model significantly use the uninformative predictors (V6 – V10)?

To determine if the random forest model significantly used the uninformative predictors (V6–V10), we need to analyze the importance scores of these variables. Random forests assign an importance score to each feature, reflecting how much each variable contributes to improving the model's predictive accuracy. Higher importance scores indicate more influential features, while lower scores show minimal or no contribution.

In this case, the importance scores for V6 to V10 are markedly lower compared to the other predictors (V1 to V5):

V6: 0.090 V7: 0.0012 V8: -0.2215 V9: -0.1229 V10: -0.0387

These low or negative values (particularly for V8, V9, and V10) imply that these predictors added little to no value in improving the model's predictive capability. In the context of random forests, a negative or near-zero importance score typically suggests that the variable may have been detrimental or irrelevant to model performance.

Negative scores, such as those seen in V8, V9, and V10, suggest that these predictors might have introduced noise rather than useful information, potentially making the model slightly more accurate without them.

Given these observations, it appears that the random forest model did not rely significantly on V6–V10. Their minimal importance scores suggest they were either largely ignored in the decision splits or did not contribute positively to model accuracy, indicating they are uninformative in this setup.

In summary, based on the variable importance scores, the random forest model did not significantly use the uninformative predictors (V6–V10). This aligns with the expectation that these variables lack meaningful information for effective prediction in this context.

(b) Now add an additional predictor that is highly correlated with one of the informative predictors. For example:

```
## [1] 0.9476651
```

Fit another random forest model to these data.

	Overall
V1	5.2642601
V2	6.0717543
V3	0.5104075
V4	6.8056776
V5	2.1812743
V6	0.2943445
V7	-0.0149050
V8	-0.0946168
V9	0.0176924
V10	-0.0763153
duplicate1	4.2867865

```
## [1] 0.9414424
```

	Overall
V1	4.2848843
V2	6.4189248
V3	0.3564442
V4	6.7188576
V5	1.8657590
V6	0.2968678
V7	0.0030368
V8	-0.0469257
V9	-0.0572195
V10	-0.0299011
duplicate1	2.7121937
duplicate2	3.2843725

Did the importance score for V1 change?

The importance score for V1 did not change significantly when adding a predictor (duplicate1) that is highly correlated with V1. The importance score for V1 remained high, indicating that it is still a crucial predictor in the model.

What happens when you add another predictor that is also highly correlated with V1?

When adding another predictor (duplicate2) that is also highly correlated with V1, the importance score for V1 remained high. This suggests that the model still considers V1 to be a critical predictor, even when there are multiple highly correlated predictors.

(c) Use the cforest function in the party package to fit a random forest model using conditional inference trees. The party package function varimp can calculate predictor importance. The conditional argument of that function toggles between the traditional importance measure and the modified version described in Strobl et al. (2007). Do these importances show the same pattern as the traditional random forest model?

Table 4: Variable Importance (Unconditional)

Variable	Importance
V1	3.3599062
V2	5.2317332
V3	0.0021304
V4	6.9257293
V5	1.4703217
V6	0.0626435
V7	0.0305772
V8	-0.0029718
V9	-0.0242242
V10	-0.0261708
duplicate1	2.3413373
duplicate2	3.8952317

Table 5: Variable Importance (Conditional)

Variable	Importance
V1	1.1701642
V2	4.1737290
V3	-0.0019259
V4	5.2999822
V5	0.8793066
V6	0.0455207
V7	-0.0079579
V8	0.0075147
V9	0.0019064
V10	-0.0101954
duplicate1	0.6876771
duplicate2	1.1431488

Both traditional random forest and cforest conditional importance measures assign high values to variables V1, V2, and V4, indicating these predictors are likely influential in both models.

Traditional importance for these variables is generally higher, suggesting some potential inflation in importance due to correlations or interactions, as cforest is designed to reduce this kind of bias.

In both methods, V6 to V10 exhibit near-zero or negative importance, reinforcing that these predictors are likely uninformative or redundant.

This low importance aligns well between both models, suggesting that the cforest model agrees with the traditional model in identifying these variables as less influential.

In both methods, V6 to V10 exhibit near-zero or negative importance, reinforcing that these predictors are likely uninformative or redundant.

This low importance aligns well between both models, suggesting that the cforest model agrees with the traditional model in identifying these variables as less influential.

The importance pattern between the cforest model and the traditional random forest is largely similar, with both identifying V1, V2, and V4 as important and V6 to V10 as uninformative. However, the conditional cforest model tends to reduce the inflated importance of highly correlated variables like the duplicates. This difference underscores cforest's effectiveness in mitigating biases due to variable correlations, aligning with Strobl et al.'s findings (2007).

(d) Repeat this process with different tree models, such as boosted trees and Cubist. Does the same pattern occur?

	var	rel.inf
V4	V4	26.8022751
V2	V2	20.7215937
V1	V1	14.8071934
V5	V5	11.0190400
duplicate2	duplicate2	8.2794921
duplicate1	duplicate1	7.5194197
V3	V3	7.2656566
V7	V7	1.1558453
V6	V6	1.0577820
V8	V8	0.5373390
V9	V9	0.4504699
V10	V10	0.3838930

The boosted tree model assigns high importance to V1, V2, and V4, consistent with the random forest models. V6 to V10 have low importance, indicating they are likely uninformative predictors.

	Overall
V1	72.0
V2	54.5
V4	49.0
V3	42.0
V5	40.0
V6	11.0
V7	0.0
V8	0.0
V9	0.0
V10	0.0

To determine whether the variable importance patterns were consistent across different models (Random Forest models RF1, RF2, RF3, cforest with unconditional importance cfImp1, cforest with conditional importance cfImp2, Boosted Trees, and Cubist), we need to compare the importance scores of each predictor across all models.

V1 and V2 consistently have high importance scores across nearly all models. These variables are likely significant predictors, as they maintain high importance in both traditional and alternative tree-based methods.

V4 also appears with high importance in multiple models, though there may be slight variation in its rank relative to V1 and V2.

Variables V6 to V10 generally show low or even negative importance scores, indicating that these predictors are likely uninformative. Both the Random Forest and cforest models (conditional and unconditional) assign these variables low scores, aligning with the expectation that they contribute little to model performance.

Cubist and Boosted Trees models reinforce this pattern, with negligible importance for V6 to V10.

The duplicate variables (e.g., duplicate1 and duplicate2) display some importance in certain models, but their scores are generally lower than those of V1, V2, and V4. This suggests that while the duplicates provide some information, they do not add unique value beyond the original variables.

The conditional importance scores (cfImp2) tend to be more conservative than the unconditional scores (cfImp1). This is expected since conditional importance controls for correlations between predictors, often resulting in lower scores for variables that are redundant or collinear.

Conditional importance confirms the low utility of V6 to V10, further supporting the pattern observed in other models.

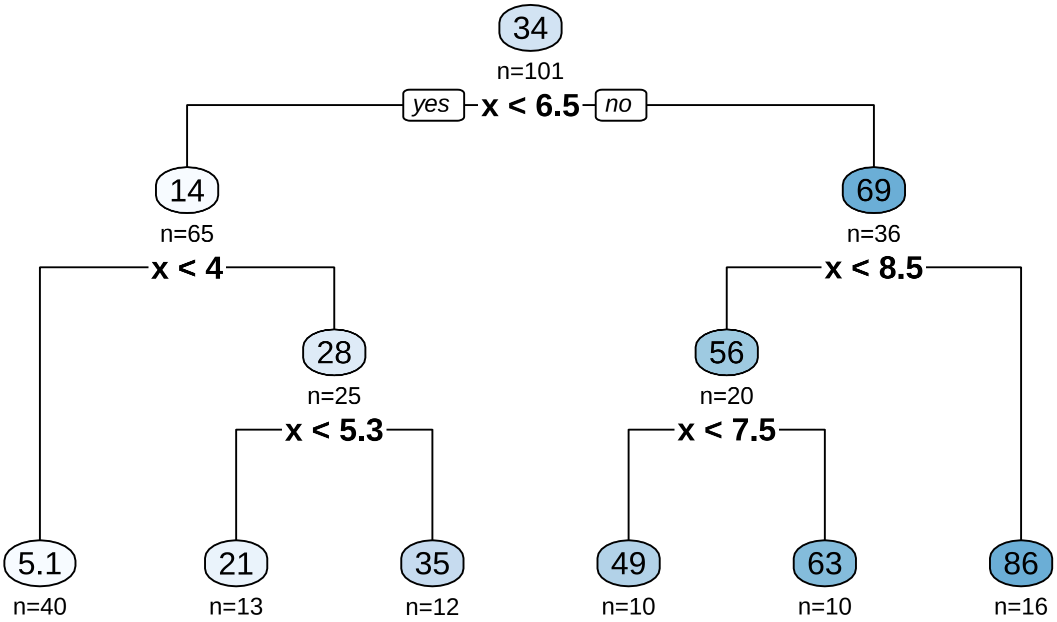
The general pattern of high importance for V1, V2, and V4, with low importance for V6 to V10, is consistent across models. This suggests robustness in identifying key predictors and uninformative variables, regardless of the model variation.

While there are slight differences in exact importance rankings, the top predictors remain largely the same, and the uninformative predictors are consistently minimized in importance.

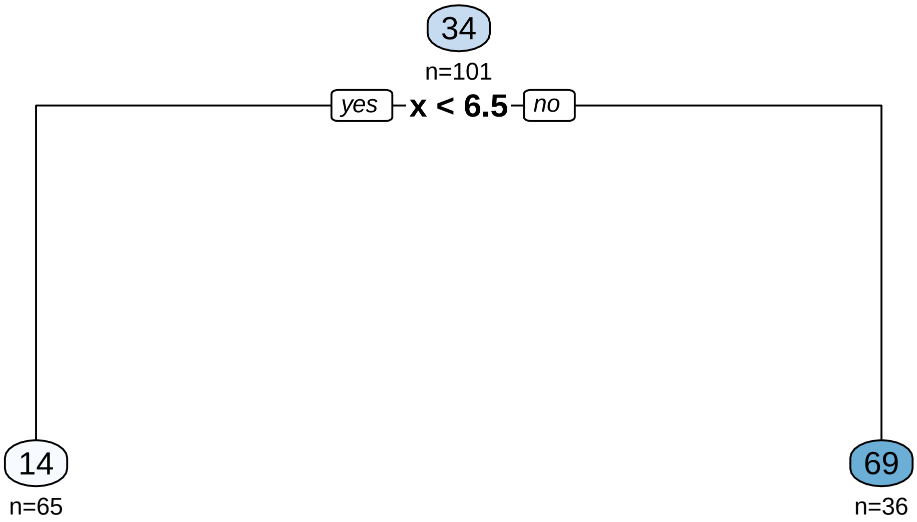
In conclusion, the pattern of variable importance observed in the initial Random Forest model holds across boosted trees, cforest with conditional importance, and Cubist. This consistency indicates that V1, V2, and V4 are reliably important predictors, while V6 to V10 are largely uninformative across different modeling techniques.

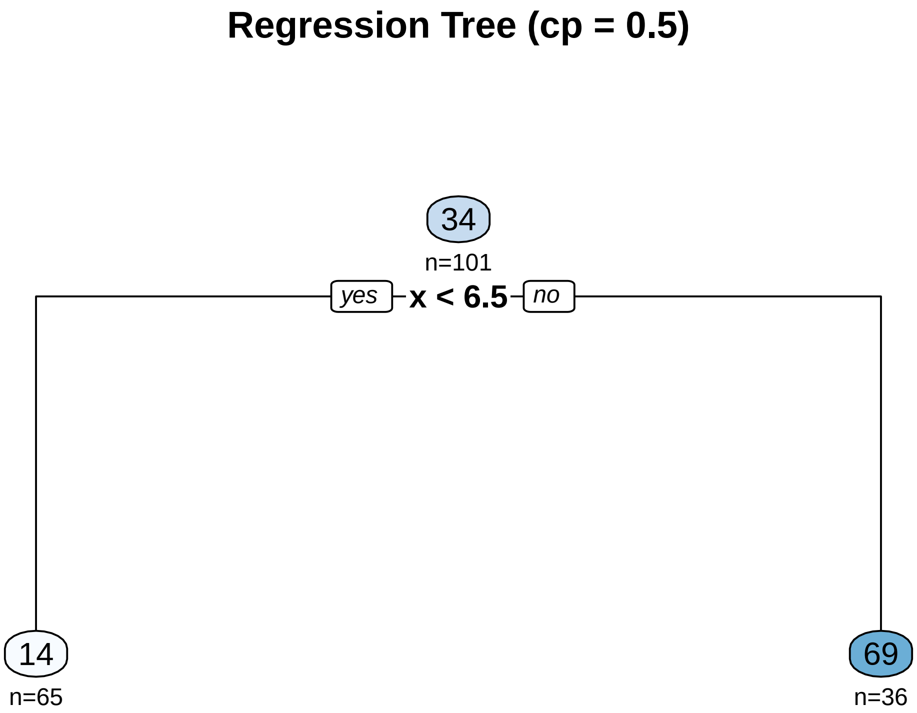
8.2. Use a simulation to show tree bias with different granularities.

Regression Tree (cp = 0.01)



Regression Tree (cp = 0.1)





The regression trees with different complexities (cp = 0.01, cp = 0.1, cp = 0.5) demonstrate the impact of tree bias on model performance. As the complexity parameter (cp) increases, the tree becomes more pruned, leading to a simpler model with fewer splits.

In the first tree (cp = 0.01), the model is highly complex, capturing the quadratic relationship between x and y with multiple splits. This tree is likely overfitting the data, capturing noise rather than the underlying pattern.

In the second tree (cp = 0.1), the model is less complex, with fewer splits and a smoother decision boundary. While this tree still captures the quadratic trend, it is less detailed than the first tree, potentially reducing overfitting.

In the third tree (cp = 0.5), the model is highly pruned, resulting in a simple linear fit. This tree underfits the data, failing to capture the quadratic relationship between x and y.

The bias-variance trade-off is evident in these trees, with the first tree having high variance (overfitting), the third tree having high bias (underfitting), and the second tree striking a balance between the two.

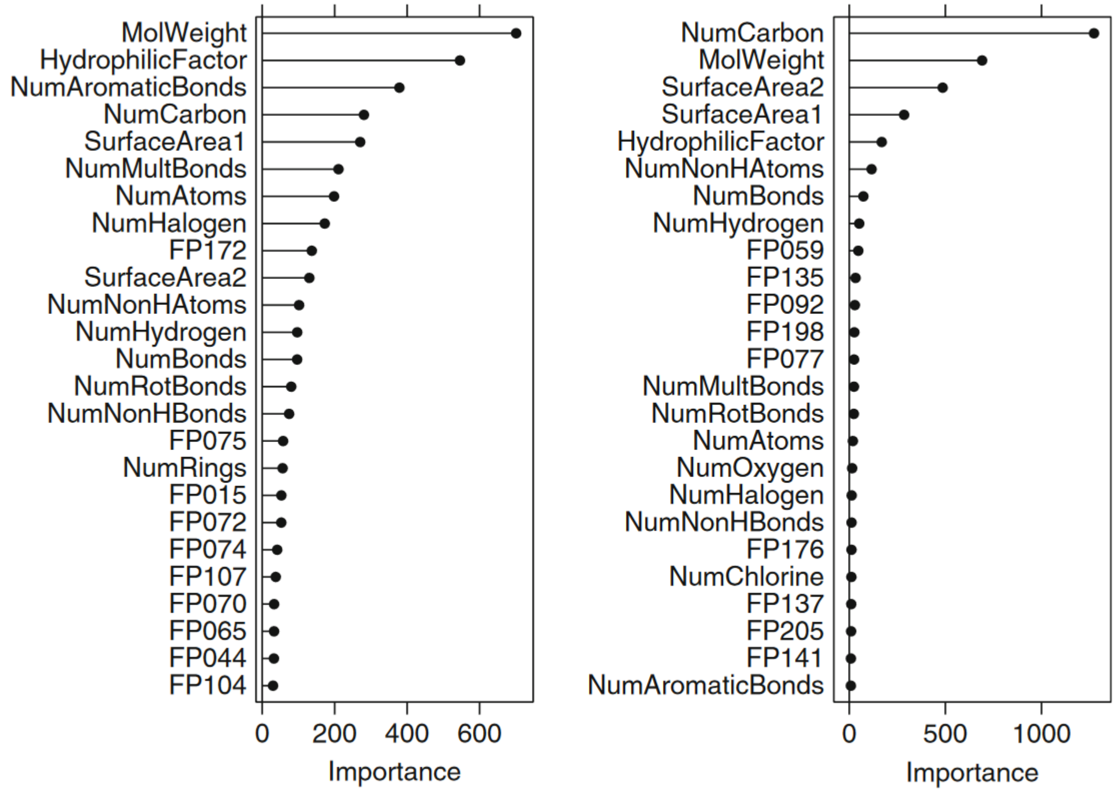
The choice of complexity parameter (cp) influences the tree's granularity, affecting the model's ability to capture the underlying relationship in the data. By adjusting the cp value, we can control the trade-off between bias and variance, optimizing the model's predictive performance.

In summary, the regression trees with different complexities illustrate the impact of tree bias on model interpretability and generalization. By tuning the complexity parameter, we can adjust the granularity of the tree to balance bias and variance, leading to more robust and accurate predictions.

8.3. In stochastic gradient boosting the bagging fraction and learning rate will govern the construction of the trees as they are guided by the gradient. Although the optimal values of these parameters should be obtained through the tuning process, it is helpful to understand how the magnitudes of these parameters affect magnitudes of variable importance. Figure 8.24 provides the variable importance plots for boosting using two extreme values for the bagging fraction (0.1 and 0.9) and the learning rate (0.1 and 0.9) for the solubility data. The left-hand plot has both parameters set to 0.1, and the right-hand plot has both set to 0.9:

```r
# Load the necessary libraries
#library(magick)

knitr::include_graphics("C:/users/enidr/OneDrive/Documents/CUNY SPS DATA 624/Data 624 Week 12 HW 9/Scre
```



(a) Why does the model on the right focus its importance on just the first few of predictors, whereas the model on the left spreads importance across more predictors?

Here's a detailed analysis of why the model on the right (with a high bagging fraction and learning rate of 0.9) focuses its importance on a few key predictors, while the model on the left (with a low bagging fraction and learning rate of 0.1) spreads importance across a broader set of predictors:

The learning rate controls how much each new tree contributes to the final model. A lower learning rate (as on the left) means that each tree has a smaller effect, allowing the model to gradually learn the relationships in the data. This slower learning process tends to engage a wider variety of predictors across the sequence of trees, as the model builds its understanding more evenly.

With a high learning rate (as on the right), each tree has a more significant impact on the model's predictions. As a result, the model quickly converges on a few strong predictors that immediately improve predictive accuracy. This can lead to the concentration of importance on these primary predictors.

The bagging fraction (or subsample rate) controls the proportion of the training data used to build each tree. A low bagging fraction (left plot) introduces more randomness, causing the model to rely on a diverse set of predictors as each tree is fitted on a unique subset of data. This enhances the exploration of different predictors, leading to a more distributed importance profile.

A high bagging fraction (right plot) means that each tree uses a large portion of the data, allowing the model to focus on the most predictive features consistently across trees. Consequently, the model converges on a few dominant predictors, leading to a more concentrated importance on those variables.

When both the learning rate and bagging fraction are high (as in the right plot), the model learns quickly from a consistent subset of data, reinforcing the importance of the strongest predictors in each iteration. This rapid convergence prioritizes the few predictors that immediately improve model performance, leading to a skewed importance distribution.

In contrast, when both parameters are low (left plot), the model adopts a more conservative approach, exploring a broader range of predictors due to the randomness in sampling and the slower learning process. This results in a more spread-out variable importance, as more predictors are evaluated over time.

The model on the right emphasizes only a few predictors due to the high bagging fraction and learning rate, leading to a focus on the most predictive variables early in the boosting process. The model on the left, with lower values for both parameters, takes a more gradual and exploratory approach, assigning importance to a wider range of predictors across the boosting iterations. This highlights how tuning parameters in stochastic gradient boosting can impact the diversity and concentration of variable importance.

(b) Which model do you think would be more predictive of other samples?

The model on the left, with a lower learning rate and bagging fraction, is likely to be more predictive of other samples for several reasons:

Lower learning rates and a lower bagging fraction introduce more randomness and slower learning, allowing the model to consider a broader range of features. This helps the model generalize better to unseen data, as it doesn't become overly reliant on a few dominant predictors specific to the training data.

A model with high learning rates and high bagging fractions (right model) often risks overfitting to the specific patterns in the training set, especially if it quickly converges on a small set of strong predictors. This means it might perform well on the training data but less effectively on new samples that might have different feature distributions or noise.

The left model, with its spread-out importance, is more likely to capture a range of signals across various predictors, making it more robust to variations in other samples. This robustness can be especially valuable if future samples are expected to be somewhat different from the current training data.

While the model on the right may show higher accuracy on the current dataset, the model on the left is more likely to perform consistently across different samples, making it a more reliable choice for broader prediction tasks.

(c) How would increasing interaction depth affect the slope of predictor importance for either model in Fig. 8.24?

Increasing the interaction depth (the maximum depth of each tree) in stochastic gradient boosting would likely cause the model to rely more heavily on higher-order interactions between predictors. This would impact the slope of predictor importance in both models shown in Fig. 8.24, but in slightly different ways depending on the other tuning parameters (bagging fraction and learning rate).

Here’s how increased interaction depth might affect each model:

Model on the Left (Low Learning Rate and Bagging Fraction):

With a low learning rate and low bagging fraction, increasing the interaction depth would allow the model to explore more complex relationships among the predictors, possibly spreading the importance even more across different predictors.

While it would still spread importance across many predictors, the increased interaction depth could accentuate the importance of some key predictors that are particularly relevant in multi-variable interactions. However, due to the low learning rate, this effect would be gradual, and the importance would likely remain more evenly distributed compared to the model on the right.

Model on the Right (High Learning Rate and Bagging Fraction):

With a high learning rate and high bagging fraction, increasing the interaction depth would likely amplify the focus on the top few predictors, resulting in an even steeper slope of importance. The model may quickly converge on strong predictor interactions that drive the response variable, further concentrating importance on a small subset of predictors.

A high interaction depth, combined with high learning rate and bagging fraction, can lead to overfitting, where the model heavily relies on complex interactions in the training data that may not generalize well to new samples. This would make the predictor importance even more skewed toward a few predictors, reflecting a narrow and potentially less generalizable focus.

Increased interaction depth would cause both models to place more emphasis on predictor interactions.

For the left model, this may lead to a modest increase in the importance of key predictors but would still maintain a broader distribution due to the lower learning rate.

For the right model, the increased interaction depth would likely sharpen the focus on a few top predictors, making the importance slope even steeper and potentially increasing the risk of overfitting.

The model on the right might become more predictive on training data with higher interaction depth but would likely be less generalizable to new data compared to the left model.

8.7. Refer to Exercises 6.3 and 7.5 which describe a chemical manufacturing process. Use the same data imputation, data splitting, and pre-processing steps as before and train several tree-based models:

Data Imputation

```
## ’data.frame':    176 obs. of  58 variables:
##  $ Yield                 : num  38 42.4 42 41.4 42.5 ...
##  $ BiologicalMaterial01  : num  6.25 8.01 8.01 8.01 7.47 6.12 7.48 6.94 6.94 6.94 ...
##  $ BiologicalMaterial02  : num  49.6 61 61 61 63.3 ...
##  $ BiologicalMaterial03  : num  57 67.5 67.5 67.5 72.2 ...
##  $ BiologicalMaterial04  : num  12.7 14.6 14.6 14.6 14 ...
##  $ BiologicalMaterial05  : num  19.5 19.4 19.4 19.4 17.9 ...
##  $ BiologicalMaterial06  : num  43.7 53.1 53.1 53.1 54.7 ...
##  $ BiologicalMaterial07  : num  100 100 100 100 100 100 100 100 100 100 ...
##  $ BiologicalMaterial08  : num  16.7 19 19 19 18.2 ...
##  $ BiologicalMaterial09  : num  11.4 12.6 12.6 12.6 12.8 ...
##  $ BiologicalMaterial10  : num  3.46 3.46 3.46 3.46 3.05 3.78 3.04 3.85 3.85 3.85 ...
##  $ BiologicalMaterial11  : num  138 154 154 154 148 ...
##  $ BiologicalMaterial12  : num  18.8 21.1 21.1 21.1 21.1 ...
##  $ ManufacturingProcess01: num  NA 0 0 0 10.7 12 11.5 12 12 12 ...
##  $ ManufacturingProcess02: num  NA 0 0 0 0 0 0 0 0 0 ...
##  $ ManufacturingProcess03: num  NA NA NA NA NA NA 1.56 1.55 1.56 1.55 ...
##  $ ManufacturingProcess04: num  NA 917 912 911 918 924 933 929 928 938 ...
##  $ ManufacturingProcess05: num  NA 1032 1004 1015 1028 ...
##  $ ManufacturingProcess06: num  NA 210 207 213 206 ...
##  $ ManufacturingProcess07: num  NA 177 178 177 178 178 177 178 177 177 ...
##  $ ManufacturingProcess08: num  NA 178 178 177 178 178 178 178 177 177 ...
##  $ ManufacturingProcess09: num  43 46.6 45.1 44.9 45 ...
##  $ ManufacturingProcess10: num  NA NA NA NA NA NA 11.6 10.2 9.7 10.1 ...
##  $ ManufacturingProcess11: num  NA NA NA NA NA NA 11.5 11.3 11.1 10.2 ...
##  $ ManufacturingProcess12: num  NA 0 0 0 0 0 0 0 0 0 ...
##  $ ManufacturingProcess13: num  35.5 34 34.8 34.8 34.6 34 32.4 33.6 33.9 34.3 ...
##  $ ManufacturingProcess14: num  4898 4869 4878 4897 4992 ...
##  $ ManufacturingProcess15: num  6108 6095 6087 6102 6233 ...
##  $ ManufacturingProcess16: num  4682 4617 4617 4635 4733 ...
##  $ ManufacturingProcess17: num  35.5 34 34.8 34.8 33.9 33.4 33.8 33.6 33.9 35.3 ...
##  $ ManufacturingProcess18: num  4865 4867 4877 4872 4886 ...
##  $ ManufacturingProcess19: num  6049 6097 6078 6073 6102 ...
##  $ ManufacturingProcess20: num  4665 4621 4621 4611 4659 ...
##  $ ManufacturingProcess21: num  0 0 0 0 -0.7 -0.6 1.4 0 0 1 ...
##  $ ManufacturingProcess22: num  NA 3 4 5 8 9 1 2 3 4 ...
##  $ ManufacturingProcess23: num  NA 0 1 2 4 1 1 2 3 1 ...
##  $ ManufacturingProcess24: num  NA 3 4 5 18 1 1 2 3 4 ...
##  $ ManufacturingProcess25: num  4873 4869 4897 4892 4930 ...
##  $ ManufacturingProcess26: num  6074 6107 6116 6111 6151 ...
##  $ ManufacturingProcess27: num  4685 4630 4637 4630 4684 ...
##  $ ManufacturingProcess28: num  10.7 11.2 11.1 11.1 11.3 11.4 11.2 11.1 11.3 11.4 ...
##  $ ManufacturingProcess29: num  21 21.4 21.3 21.3 21.6 21.7 21.2 21.2 21.5 21.7 ...
##  $ ManufacturingProcess30: num  9.9 9.9 9.4 9.4 9 10.1 11.2 10.9 10.5 9.8 ...
##  $ ManufacturingProcess31: num  69.1 68.7 69.3 69.3 69.4 68.2 67.6 67.9 68 68.5 ...
##  $ ManufacturingProcess32: num  156 169 173 171 171 173 159 161 160 164 ...
##  $ ManufacturingProcess33: num  66 66 66 68 70 70 65 65 65 66 ...
##  $ ManufacturingProcess34: num  2.4 2.6 2.6 2.5 2.5 2.5 2.5 2.5 2.5 2.5 ...
##  $ ManufacturingProcess35: num  486 508 509 496 468 490 475 478 491 488 ...
##  $ ManufacturingProcess36: num  0.019 0.019 0.018 0.018 0.017 0.018 0.019 0.019 0.019 0.019 ...
##  $ ManufacturingProcess37: num  0.5 2 0.7 1.2 0.2 0.4 0.8 1 1.2 1.8 ...
##  $ ManufacturingProcess38: num  3 2 2 2 2 2 2 2 3 3 ...
##  $ ManufacturingProcess39: num  7.2 7.2 7.2 7.2 7.3 7.2 7.3 7.3 7.4 7.1 ...
##  $ ManufacturingProcess40: num  NA 0.1 0 0 0 0 0 0 0 0 ...
##  $ ManufacturingProcess41: num  NA 0.15 0 0 0 0 0 0 0 0 ...
##  $ ManufacturingProcess42: num  11.6 11.1 12 10.6 11 11.5 11.7 11.4 11.4 11.3 ...
##  $ ManufacturingProcess43: num  3 0.9 1 1.1 1.1 2.2 0.7 0.8 0.9 0.8 ...
##  $ ManufacturingProcess44: num  1.8 1.9 1.8 1.8 1.7 1.8 2 2 1.9 1.9 ...
##  $ ManufacturingProcess45: num  2.4 2.2 2.3 2.1 2.1 2 2.2 2.2 2.1 2.4 ...
```

```
## [1] 106
```

```
## [1] 0
```

Split the data into training and test sets

```
## [1] 124  58
```

```
## [1] 52 58
```

Pre-process the data

```
## ’data.frame’:    124 obs. of  58 variables:
##  $ Yield                 : num  -1.217 1.041 0.699 1.299 1.904 ...
##  $ BiologicalMaterial01  : num  -0.224 2.325 2.325 1.543 -0.412 ...
##  $ BiologicalMaterial02  : num  -1.567 1.281 1.281 1.872 0.629 ...
##  $ BiologicalMaterial03  : num  -2.6543 -0.0861 -0.0861 1.0795 -0.6164 ...
##  $ BiologicalMaterial04  : num  0.234 1.333 1.333 0.971 1.633 ...
##  $ BiologicalMaterial05  : num  0.496 0.413 0.413 -0.385 1.75 ...
##  $ BiologicalMaterial06  : num  -1.407 1.065 1.065 1.464 0.563 ...
##  $ BiologicalMaterial07  : num  -0.0898 -0.0898 -0.0898 -0.0898 -0.0898 ...
##  $ BiologicalMaterial08  : num  -1.24 2.36 2.36 1.12 1.24 ...
##  $ BiologicalMaterial09  : num  -3.308 -0.7 -0.7 -0.113 -1.687 ...
##  $ BiologicalMaterial10  : num  1.102 1.102 1.102 0.428 1.627 ...
##  $ BiologicalMaterial11  : num  -1.817 1.366 1.366 0.128 1 ...
##  $ BiologicalMaterial12  : num  -1.73 1.04 1.04 1.04 0.68 ...
##  $ ManufacturingProcess01: num  0.18 -6.353 -6.353 -0.327 0.405 ...
##  $ ManufacturingProcess02: num  0.567 -1.925 -1.925 -1.925 -1.925 ...
##  $ ManufacturingProcess03: num  0.2563 -0.0513 0.3588 -0.0513 0.4613 ...
##  $ ManufacturingProcess04: num  0.525 -3.224 -3.384 -2.263 -1.302 ...
##  $ ManufacturingProcess05: num  -0.484 0.129 0.564 1.074 0.651 ...
##  $ ManufacturingProcess06: num  -0.661 -0.147 2.602 -0.767 0.652 ...
##  $ ManufacturingProcess07: num  -0.19 1.01 -0.99 1.01 1.01 ...
##  $ ManufacturingProcess08: num  -0.272 0.931 -1.074 0.931 0.931 ...
##  $ ManufacturingProcess09: num  -1.837 -0.416 -0.519 -0.492 -0.245 ...
##  $ ManufacturingProcess10: num  -5.34e-02 3.47e-01 1.17e-16 -3.74e-01 3.20e-01 ...
##  $ ManufacturingProcess11: num  -0.1365 0.5246 0.7833 0.0647 1.4157 ...
##  $ ManufacturingProcess12: num  -0.463 -0.463 -0.463 -0.463 -0.463 ...
##  $ ManufacturingProcess13: num  0.9568 0.2876 0.2876 0.0964 -0.4773 ...
##  $ ManufacturingProcess14: num  0.728 0.367 0.71 2.426 2.3 ...
##  $ ManufacturingProcess15: num  1.082 0.728 0.981 3.19 3.005 ...
##  $ ManufacturingProcess16: num  0.294 0.138 0.181 0.416 0.543 ...
##  $ ManufacturingProcess17: num  0.979 0.418 0.418 -0.304 -0.705 ...
##  $ ManufacturingProcess18: num  0.154 0.182 0.17 0.202 0.147 ...
##  $ ManufacturingProcess19: num  0.444 1.086 0.975 1.617 1.905 ...
##  $ ManufacturingProcess20: num  0.285 0.179 0.155 0.27 0.36 ...
##  $ ManufacturingProcess21: num  0.289 0.289 0.289 -0.63 -0.498 ...
##  $ ManufacturingProcess22: num  0.0213 -0.4597 -0.1591 0.7429 1.0436 ...
##  $ ManufacturingProcess23: num  0.927 -1.207 -0.579 0.676 -1.207 ...
##  $ ManufacturingProcess24: num  0.921 -0.761 -0.593 1.593 -1.265 ...
##  $ ManufacturingProcess25: num  0.128 0.183 0.172 0.259 0.124 ...
##  $ ManufacturingProcess26: num  0.133 0.21 0.201 0.274 0.232 ...
##  $ ManufacturingProcess27: num  0.318 0.203 0.186 0.316 0.323 ...
##  $ ManufacturingProcess28: num  0.876 0.951 0.951 0.989 1.007 ...
##  $ ManufacturingProcess29: num  0.541 0.698 0.698 0.854 0.906 ...
##  $ ManufacturingProcess30: num  0.713 0.235 0.235 -0.147 0.905 ...
##  $ ManufacturingProcess31: num  -0.1348 -0.1039 -0.1039 -0.0885 -0.2735 ...
##  $ ManufacturingProcess32: num  -0.495 2.738 2.358 2.358 2.738 ...
##  $ ManufacturingProcess33: num  0.952 0.952 1.747 2.542 2.542 ...
##  $ ManufacturingProcess34: num  -1.764 2.047 0.141 0.141 0.141 ...
##  $ ManufacturingProcess35: num  -0.88 1.2743 0.0567 -2.566 -0.5053 ...
##  $ ManufacturingProcess36: num  -0.652 -1.849 -1.849 -3.046 -1.849 ...
##  $ ManufacturingProcess37: num  -1.11 -0.679 0.398 -1.757 -1.326 ...
##  $ ManufacturingProcess38: num  0.676 -0.745 -0.745 -0.745 -0.745 ...
##  $ ManufacturingProcess39: num  0.273 0.273 0.273 0.33 0.273 ...
##  $ ManufacturingProcess40: num  0.0547 -0.4672 -0.4672 -0.4672 -0.4672 ...
##  $ ManufacturingProcess41: num  -0.0654 -0.4343 -0.4343 -0.4343 -0.4343 ...
##  $ ManufacturingProcess42: num  0.237 0.4114 -0.199 -0.0246 0.1934 ...
##  $ ManufacturingProcess43: num  2.066 0.087 0.186 0.186 1.275 ...
##  $ ManufacturingProcess44: num  0.0277 0.0277 0.0277 -0.2369 0.0277 ...
##  $ ManufacturingProcess45: num  0.61 0.396 -0.031 -0.031 -0.245 ...
```

```
## ’data.frame’:    52 obs. of  58 variables:
##  $ Yield                 : num  1.271 1.652 1.618 0.402 -0.488 ...
##  $ BiologicalMaterial01  : num  2.325 1.558 0.775 2.094 1.601 ...
##  $ BiologicalMaterial02  : num  1.281 2.157 1.939 0.594 0.974 ...
##  $ BiologicalMaterial03  : num  -0.0861 1.1186 1.0331 -0.2156 -0.135 ...
##  $ BiologicalMaterial04  : num  1.333 0.856 1.938 1.754 1.949 ...
##  $ BiologicalMaterial05  : num  0.413 -0.495 0.446 1.239 1.756 ...
##  $ BiologicalMaterial06  : num  1.065 1.409 1.48 0.461 0.844 ...
##  $ BiologicalMaterial07  : num  -0.0898 -0.0898 -0.0898 -0.0898 -0.0898 ...
##  $ BiologicalMaterial08  : num  2.36 1.88 2.07 1.56 2.01 ...
##  $ BiologicalMaterial09  : num  -0.7 0.239 0.662 -0.348 0.31 ...
##  $ BiologicalMaterial10  : num  1.102 0.412 1.742 2.005 2.383 ...
##  $ BiologicalMaterial11  : num  1.3657 0.0257 1.474 1.1921 2.3237 ...
##  $ BiologicalMaterial12  : num  1.041 0.667 1.54 0.605 2.002 ...
##  $ ManufacturingProcess01: num  -6.353 0.124 0.405 -0.102 0.799 ...
##  $ ManufacturingProcess02: num  -1.92 -1.92 -1.92 -1.92 -1.92 ...
##  $ ManufacturingProcess03: num  0.0513 0.8714 0.3588 0.8714 0.7689 ...
##  $ ManufacturingProcess04: num  -2.423 0.14 -0.501 -0.661 -0.501 ...
##  $ ManufacturingProcess05: num  1.259 -0.451 0.418 1.074 3.63 ...
##  $ ManufacturingProcess06: num  1.1393 1.1393 1.893 -0.6786 0.0308 ...
##  $ ManufacturingProcess07: num  -0.99 -0.99 1.01 -0.99 1.01 ...
##  $ ManufacturingProcess08: num  0.931 0.931 0.931 -1.074 0.931 ...
##  $ ManufacturingProcess09: num  0.6129 2.5274 2.0608 -0.0939 -0.8349 ...
##  $ ManufacturingProcess10: num  0.561 3.257 1.388 -0.214 -0.347 ...
##  $ ManufacturingProcess11: num  1.071 2.997 2.709 0.841 0.553 ...
##  $ ManufacturingProcess12: num  -0.463 -0.463 -0.463 -0.463 -0.463 ...
##  $ ManufacturingProcess13: num  -0.477 -2.007 -0.86 1.244 0.957 ...
##  $ ManufacturingProcess14: num  0.2041 -2.0359 -0.0668 1.5229 1.7939 ...
##  $ ManufacturingProcess15: num  0.863 -0.757 1.031 2.246 2.313 ...
##  $ ManufacturingProcess16: num  0.138 -0.176 0.159 0.397 0.459 ...
##  $ ManufacturingProcess17: num  -0.224 -0.384 -0.545 1.22 0.979 ...
##  $ ManufacturingProcess18: num  0.1587 -0.0908 -0.0725 0.1381 0.2045 ...
##  $ ManufacturingProcess19: num  1.507 -0.352 -0.153 1.573 2.104 ...
##  $ ManufacturingProcess20: num  0.1788 -0.06 0.0124 0.2632 0.3934 ...
##  $ ManufacturingProcess21: num  0.289 2.126 0.289 0.289 0.289 ...
##  $ ManufacturingProcess22: num  -0.76 -1.36 -1.06 -1.36 -0.76 ...
##  $ ManufacturingProcess23: num  -1.834 -1.207 -0.579 -1.207 -0.579 ...
##  $ ManufacturingProcess24: num  -0.929 -1.265 -1.097 -1.265 -0.929 ...
##  $ ManufacturingProcess25: num  0.1193 -0.0498 -0.0246 0.0987 0.1353 ...
##  $ ManufacturingProcess26: num  0.193 0.102 0.105 0.18 0.197 ...
##  $ ManufacturingProcess27: num  0.1858 0.046 0.0797 0.2558 0.345 ...
##  $ ManufacturingProcess28: num  0.97 0.97 0.951 0.951 0.951 ...
##  $ ManufacturingProcess29: num  0.75 0.645 0.645 0.698 0.75 ...
##  $ ManufacturingProcess30: num  0.713 1.957 1.67 0.809 0.618 ...
##  $ ManufacturingProcess31: num  -0.196 -0.366 -0.32 -0.196 -0.181 ...
##  $ ManufacturingProcess32: num  1.9773 0.0752 0.4556 1.5969 -0.4955 ...
##  $ ManufacturingProcess33: num  0.952 0.554 0.554 1.747 0.156 ...
##  $ ManufacturingProcess34: num  2.047 0.141 0.141 0.141 -1.764 ...
##  $ ManufacturingProcess35: num  1.181 -1.91 -1.629 -0.037 1.743 ...
##  $ ManufacturingProcess36: num  -0.652 -0.652 -0.652 -0.652 1.741 ...
##  $ ManufacturingProcess37: num  2.121 -0.464 -0.033 0.613 -0.679 ...
##  $ ManufacturingProcess38: num  -0.745 -0.745 -0.745 0.676 0.676 ...
##  $ ManufacturingProcess39: num  0.273 0.33 0.33 0.386 0.33 ...
##  $ ManufacturingProcess40: num  2.142 -0.467 -0.467 -0.467 -0.467 ...
##  $ ManufacturingProcess41: num  2.332 -0.434 -0.434 -0.434 -0.434 ...
##  $ ManufacturingProcess42: num  0.019 0.281 0.15 0.193 0.237 ...
##  $ ManufacturingProcess43: num  -0.012 -0.21 -0.111 0.186 0.879 ...
##  $ ManufacturingProcess44: num  0.292 0.557 0.557 0.292 -0.237 ...
##  $ ManufacturingProcess45: num  0.183 0.183 0.183 0.396 0.396 ...
```

Train several tree-based models

Table 8: Random Forest Variable Importance

	Overall
BiologicalMaterial01	1.0250689
BiologicalMaterial02	1.4550055
BiologicalMaterial03	5.9695779
BiologicalMaterial04	2.4935383
BiologicalMaterial05	0.7129550
BiologicalMaterial06	3.7359421
BiologicalMaterial07	0.0005086
BiologicalMaterial08	1.4371030
BiologicalMaterial09	1.3084073
BiologicalMaterial10	0.6959830
BiologicalMaterial11	2.9299639
BiologicalMaterial12	6.1489521
ManufacturingProcess01	0.9435431
ManufacturingProcess02	0.6483790
ManufacturingProcess03	0.3637617
ManufacturingProcess04	0.5762328
ManufacturingProcess05	1.1475834
ManufacturingProcess06	2.6527207
ManufacturingProcess07	0.0668778
ManufacturingProcess08	0.1224550
ManufacturingProcess09	3.4785271
ManufacturingProcess10	0.9026901
ManufacturingProcess11	2.0330432
ManufacturingProcess12	0.4326927
ManufacturingProcess13	2.8946511
ManufacturingProcess14	1.1526189
ManufacturingProcess15	1.5735640
ManufacturingProcess16	0.8626007
ManufacturingProcess17	5.8170626
ManufacturingProcess18	1.4992938
ManufacturingProcess19	1.4261193
ManufacturingProcess20	1.4365852
ManufacturingProcess21	1.6587681
ManufacturingProcess22	0.6126686

	Overall
ManufacturingProcess23	0.8103363
ManufacturingProcess24	1.1744794
ManufacturingProcess25	1.0061482
ManufacturingProcess26	0.7727862
ManufacturingProcess27	0.8181480
ManufacturingProcess28	3.9845243
ManufacturingProcess29	1.0267283
ManufacturingProcess30	1.2909734
ManufacturingProcess31	6.6565852
ManufacturingProcess32	33.0039965
ManufacturingProcess33	0.7751148
ManufacturingProcess34	0.3618439
ManufacturingProcess35	0.6434773
ManufacturingProcess36	1.6743117
ManufacturingProcess37	1.0821300
ManufacturingProcess38	0.1257839
ManufacturingProcess39	1.3273693
ManufacturingProcess40	0.0560122
ManufacturingProcess41	0.3654626
ManufacturingProcess42	0.6595481
ManufacturingProcess43	0.3945774
ManufacturingProcess44	0.2757154
ManufacturingProcess45	0.3537814

Table 9: Gradient Boosting Variable Importance

	Overall
BiologicalMaterial01	6.4341814
BiologicalMaterial02	2.8127243
BiologicalMaterial03	14.8459444
BiologicalMaterial04	2.5890775
BiologicalMaterial05	6.6833813
BiologicalMaterial06	0.9957944
BiologicalMaterial07	0.0000000
BiologicalMaterial08	1.9293870
BiologicalMaterial09	7.2278283
BiologicalMaterial10	6.4850761
BiologicalMaterial11	11.8184609
BiologicalMaterial12	23.3544791
ManufacturingProcess01	14.9190522
ManufacturingProcess02	3.0785285
ManufacturingProcess03	0.8368619
ManufacturingProcess04	4.8241147
ManufacturingProcess05	9.5715555
ManufacturingProcess06	46.0552692
ManufacturingProcess07	0.6198725
ManufacturingProcess08	1.0742612
ManufacturingProcess09	19.5011478
ManufacturingProcess10	11.6925874
ManufacturingProcess11	0.7219243

	Overall
ManufacturingProcess12	0.0000000
ManufacturingProcess13	10.2311028
ManufacturingProcess14	9.9916996
ManufacturingProcess15	2.7262477
ManufacturingProcess16	5.4975773
ManufacturingProcess17	29.2255062
ManufacturingProcess18	2.4795584
ManufacturingProcess19	8.2243394
ManufacturingProcess20	7.8010646
ManufacturingProcess21	9.5239938
ManufacturingProcess22	9.8349179
ManufacturingProcess23	3.7286159
ManufacturingProcess24	14.1616677
ManufacturingProcess25	4.4557453
ManufacturingProcess26	6.4343469
ManufacturingProcess27	3.9392445
ManufacturingProcess28	0.1363688
ManufacturingProcess29	7.4740151
ManufacturingProcess30	7.7532269
ManufacturingProcess31	7.8154062
ManufacturingProcess32	149.7787687
ManufacturingProcess33	0.3485179
ManufacturingProcess34	0.5070327
ManufacturingProcess35	2.7553915
ManufacturingProcess36	1.1289224
ManufacturingProcess37	8.7236303
ManufacturingProcess38	0.3696117
ManufacturingProcess39	9.2049393
ManufacturingProcess40	0.0000000
ManufacturingProcess41	0.0000000
ManufacturingProcess42	1.2032836
ManufacturingProcess43	5.8319692
ManufacturingProcess44	1.9204451
ManufacturingProcess45	1.0357162

Table 10: Cubist Variable Importance

	Overall
ManufacturingProcess17	42.5
ManufacturingProcess39	21.0
ManufacturingProcess32	47.5
BiologicalMaterial12	22.0
ManufacturingProcess31	4.0
BiologicalMaterial05	2.5
ManufacturingProcess06	14.0
ManufacturingProcess22	1.0
BiologicalMaterial04	3.0
ManufacturingProcess29	14.5
ManufacturingProcess25	14.5
ManufacturingProcess09	12.5

	Overall
BiologicalMaterial03	11.0
ManufacturingProcess13	10.5
ManufacturingProcess33	10.5
ManufacturingProcess11	10.0
ManufacturingProcess26	8.0
BiologicalMaterial08	8.0
ManufacturingProcess24	8.0
BiologicalMaterial09	7.5
BiologicalMaterial02	7.0
ManufacturingProcess27	7.0
BiologicalMaterial06	5.0
ManufacturingProcess18	4.0
ManufacturingProcess16	3.5
ManufacturingProcess15	3.5
ManufacturingProcess20	3.5
ManufacturingProcess04	3.0
ManufacturingProcess03	2.5
ManufacturingProcess23	2.5
ManufacturingProcess05	2.0
BiologicalMaterial10	2.0
ManufacturingProcess12	2.0
ManufacturingProcess19	2.0
ManufacturingProcess21	2.0
ManufacturingProcess01	1.5
BiologicalMaterial11	1.5
ManufacturingProcess43	1.0
ManufacturingProcess02	1.0
BiologicalMaterial01	0.0
BiologicalMaterial07	0.0
ManufacturingProcess07	0.0
ManufacturingProcess08	0.0
ManufacturingProcess10	0.0
ManufacturingProcess14	0.0
ManufacturingProcess28	0.0
ManufacturingProcess30	0.0
ManufacturingProcess34	0.0
ManufacturingProcess35	0.0
ManufacturingProcess36	0.0
ManufacturingProcess37	0.0
ManufacturingProcess38	0.0
ManufacturingProcess40	0.0
ManufacturingProcess41	0.0
ManufacturingProcess42	0.0
ManufacturingProcess44	0.0
ManufacturingProcess45	0.0

Table 11: Bagged Tree Variable Importance

	Overall
BiologicalMaterial01	0.1435384

	Overall
BiologicalMaterial02	0.1814203
BiologicalMaterial03	0.5798112
BiologicalMaterial04	0.2007435
BiologicalMaterial05	0.2899065
BiologicalMaterial06	0.2959157
BiologicalMaterial07	0.0000000
BiologicalMaterial08	0.2973043
BiologicalMaterial09	0.3102775
BiologicalMaterial10	0.1051152
BiologicalMaterial11	0.5915191
BiologicalMaterial12	0.8861626
ManufacturingProcess01	0.2944574
ManufacturingProcess02	0.1560424
ManufacturingProcess03	0.0211690
ManufacturingProcess04	0.1466957
ManufacturingProcess05	0.1728725
ManufacturingProcess06	0.5584584
ManufacturingProcess07	0.0514549
ManufacturingProcess08	0.0108126
ManufacturingProcess09	0.6598441
ManufacturingProcess10	0.2194264
ManufacturingProcess11	0.5608103
ManufacturingProcess12	0.0699787
ManufacturingProcess13	0.5120909
ManufacturingProcess14	0.3759635
ManufacturingProcess15	0.4535918
ManufacturingProcess16	0.2416880
ManufacturingProcess17	1.0955617
ManufacturingProcess18	0.2502621
ManufacturingProcess19	0.3272336
ManufacturingProcess20	0.2773384
ManufacturingProcess21	0.3755464
ManufacturingProcess22	0.0639143
ManufacturingProcess23	0.1269655
ManufacturingProcess24	0.2260277
ManufacturingProcess25	0.2290149
ManufacturingProcess26	0.1301623
ManufacturingProcess27	0.2417423
ManufacturingProcess28	0.4171542
ManufacturingProcess29	0.1861834
ManufacturingProcess30	0.3179287
ManufacturingProcess31	0.7062981
ManufacturingProcess32	0.7588114
ManufacturingProcess33	0.0548682
ManufacturingProcess34	0.0000000
ManufacturingProcess35	0.1346992
ManufacturingProcess36	0.0832284
ManufacturingProcess37	0.1663320
ManufacturingProcess38	0.0157006
ManufacturingProcess39	0.1964913
ManufacturingProcess40	0.0160962
ManufacturingProcess41	0.0160962

	Overall
ManufacturingProcess42	0.1275682
ManufacturingProcess43	0.1064286
ManufacturingProcess44	0.0563482
ManufacturingProcess45	0.1288447

(a) Which tree-based regression model gives the optimal resampling and test set performance?

Table 12: Performance Metrics for Tree-based Models

Model	RMSE	Rsquared	MAE
Random Forest	0.7014812	0.6604003	0.5401092
Gradient Boosting	0.6974752	0.6490253	0.5267574
Cubist	0.5656923	0.7543585	0.4446262
Bagged Tree	0.7269549	0.5957590	0.5366908

Based on the performance metrics provided, the Cubist model shows the best overall performance across the given metrics, indicating it likely gives the optimal resampling and test set performance among the listed tree-based regression models.

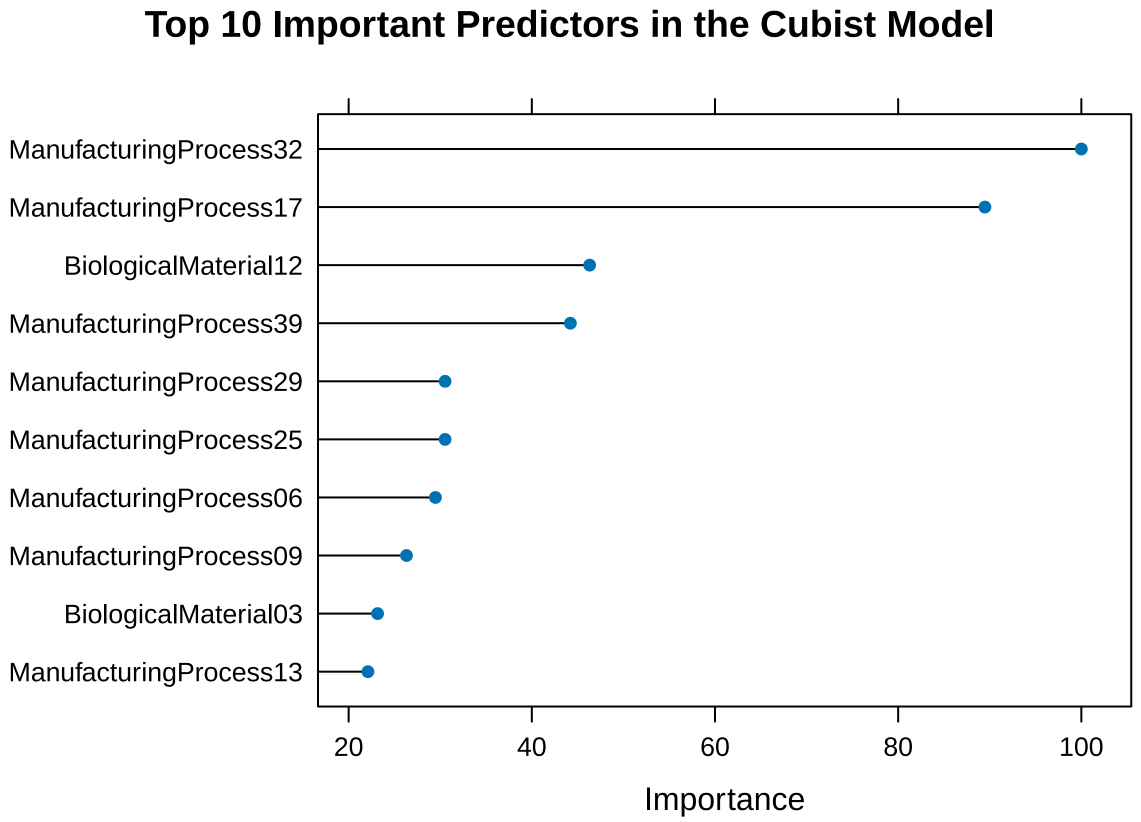
RMSE (Root Mean Squared Error): Lower RMSE values indicate better performance. The Cubist model has the lowest RMSE at 0.5657, indicating better prediction accuracy.

R^2 (R-squared): Higher R-squared values indicate better fit, with values closer to 1 being preferable. The Cubist model has the highest R-squared at 0.7544, indicating it explains the variance in the target variable better than the others.

MAE (Mean Absolute Error): Lower MAE values indicate better performance, reflecting smaller prediction errors. The Cubist model also has the lowest MAE at 0.4446, showing it has lower prediction errors on average.

(b) Which predictors are most important in the optimal tree-based regression model?

	Overall
ManufacturingProcess32	100.00000
ManufacturingProcess17	89.47368
BiologicalMaterial12	46.31579
ManufacturingProcess39	44.21053
ManufacturingProcess29	30.52632
ManufacturingProcess25	30.52632
ManufacturingProcess06	29.47368
ManufacturingProcess09	26.31579
BiologicalMaterial03	23.15789
ManufacturingProcess13	22.10526



These predictors have been ranked based on their importance values, with ManufacturingProcess32 being the most influential predictor, followed by ManufacturingProcess17. The chart I created accurately reflects this ranking by displaying these predictors in descending order of importance.

Do either the biological or process variables dominate the list?

Based on the top 10 important predictors you provided, it appears that the process variables dominate the list. Here's a breakdown:

Process Variables:

ManufacturingProcess32 ManufacturingProcess17 ManufacturingProcess39 ManufacturingProcess29 ManufacturingProcess25 ManufacturingProcess06 ManufacturingProcess09 ManufacturingProcess13 There are 8 process-related variables in the top 10.

Biological Variables:

BiologicalMaterial12 BiologicalMaterial03

There are 2 biological variables in the top 10.

The process variables clearly dominate the list of top predictors, with 8 out of the top 10 most important predictors being related to manufacturing processes. This suggests that in this model, the manufacturing processes have a greater impact on the target variable than the biological materials.

How do the top 10 important predictors compare to the top 10 predictors from the optimal linear and nonlinear models?

```
## ### Linear Regression Results ###
##      RMSE  Rsquared       MAE 
## 0.8285972 0.5095169 0.6407973
```

```
## ### SVM Results ###
```

```
##      RMSE  Rsquared       MAE 
## 0.7642341 0.5855119 0.5910040
```

```
## ### KNN Metrics Results ###
```

```
##      RMSE  Rsquared       MAE 
## 0.8462222 0.4379732 0.7106254
```

```
## # weights:  60
## initial  value 158.409217 
## final  value 138.010105 
## converged
## # weights:  178
## initial  value 162.657532 
## final  value 138.010105 
## converged
## # weights:  296
## initial  value 191.618501 
## final  value 138.010105 
## converged
## # weights:  60
## initial  value 186.915115 
## iter  10 value 137.524487
## iter  20 value 97.266784
## iter  30 value 92.747332
## iter  40 value 91.607796
## iter  50 value 91.486835
## iter  60 value 90.730225
## iter  70 value 90.567804
## iter  80 value 90.519484
## final  value 90.519302 
## converged
## # weights:  178
## initial  value 176.114373 
## iter  10 value 104.640026
## iter  20 value 90.514797
## iter  30 value 88.620311
## iter  40 value 87.744114
## iter  50 value 87.683245
## iter  60 value 87.680905
## iter  70 value 87.680309
## iter  80 value 87.678479
## iter  90 value 87.628359
## iter 100 value 87.444198
## final  value 87.444198 
## stopped after 100 iterations
## # weights:  296
## initial  value 189.483654
## iter  10 value 104.716186
## iter  20 value 91.277749
## iter  30 value 88.841857
## iter  40 value 87.611289
## iter  50 value 87.394278
## iter  60 value 87.100503
## iter  70 value 86.190594
## iter  80 value 85.902408
## iter  90 value 85.713085
## iter 100 value 85.568056
## final  value 85.568056
## stopped after 100 iterations
## # weights:  60
## initial  value 160.269187
## iter  10 value 138.011602
## iter  20 value 138.010624
## iter  30 value 138.009455
## iter  40 value 138.008026
## iter  50 value 138.006236
## iter  60 value 138.003914
## iter  70 value 138.000769
## iter  80 value 137.996240
## iter  90 value 137.989109
## iter 100 value 137.976114
## final  value 137.976114
## stopped after 100 iterations
## # weights:  178
## initial  value 186.709623
## iter  10 value 138.023010
## iter  20 value 138.007882
## iter  30 value 89.379531
## iter  40 value 82.118709
## iter  50 value 81.647256
## iter  60 value 80.532407
## iter  70 value 80.430132
## iter  80 value 80.362522
## iter  90 value 80.316221
## iter 100 value 80.023866
## final  value 80.023866
## stopped after 100 iterations
## # weights:  296
## initial  value 187.207704
## iter  10 value 138.027654
## iter  20 value 134.947281
## iter  30 value 81.222016
## iter  40 value 80.470900
## iter  50 value 80.313861
## iter  60 value 80.274329
## iter  70 value 80.270310
## iter  80 value 80.250275
## iter  90 value 80.213986
## iter 100 value 80.189836
## final  value 80.189836
## stopped after 100 iterations
## # weights:  60
## initial  value 101.307121
## final  value 92.036094
## converged
## # weights:  178
## initial  value 124.865488
## final  value 92.036104
## converged
## # weights:  296
## initial  value 139.515740
## final  value 92.036104
## converged
## # weights:  60
## initial  value 109.119678
## iter  10 value 80.534650
## iter  20 value 67.039634
## iter  30 value 60.298857
## iter  40 value 57.900940
## iter  50 value 57.663056
## iter  60 value 57.621939
## final  value 57.621779
## converged
## # weights:  178
## initial  value 105.893055
## iter  10 value 63.155531
## iter  20 value 57.256234
## iter  30 value 55.599448
## iter  40 value 55.445834
## iter  50 value 55.407712
## iter  60 value 55.387912
## iter  70 value 55.385010
## iter  80 value 55.381640
## iter  90 value 55.381568
## final  value 55.381563
## converged
## # weights:  296
## initial  value 131.507907
## iter  10 value 68.606231
## iter  20 value 59.213952
## iter  30 value 55.703478
## iter  40 value 54.639093
## iter  50 value 54.418161
## iter  60 value 54.403564
## iter  70 value 54.401825
## iter  80 value 54.401700
## iter  90 value 54.401688
## final  value 54.401686
## converged
## # weights:  60
## initial  value 138.474675
## iter  10 value 92.050679
## iter  20 value 92.047655
## iter  30 value 92.047572
## iter  40 value 92.047487
## iter  50 value 92.047400
## iter  60 value 92.047311
## iter  70 value 92.047220
## iter  80 value 92.047127
## iter  90 value 92.047033
## iter 100 value 92.046938
## final  value 92.046938
## stopped after 100 iterations
## # weights:  178
## initial  value 120.206681
## iter  10 value 92.043810
## iter  20 value 92.000145
## iter  30 value 58.143055
## iter  40 value 51.932807
## iter  50 value 51.702084
## iter  60 value 51.630965
## iter  70 value 50.756677
## iter  80 value 49.820559
## iter  90 value 49.381560
## iter 100 value 49.262591
## final  value 49.262591
## stopped after 100 iterations
## # weights:  296
## initial  value 147.156743
## iter  10 value 92.051166
## iter  20 value 91.974336
## iter  30 value 60.266153
## iter  40 value 54.820069
## iter  50 value 54.579474
## iter  60 value 54.549254
## iter  70 value 54.444602
## iter  80 value 53.383752
## iter  90 value 53.358021
## iter 100 value 53.295363
## final  value 53.295363
## stopped after 100 iterations
## # weights:  60
## initial  value 117.190271
## final  value 110.782090
## converged
## # weights:  178
## initial  value 113.079356
## final  value 110.782033
## converged
## # weights:  296
## initial  value 149.893435
## final  value 110.782090
## converged
## # weights:  60
## initial  value 141.003780
## iter  10 value 78.572536
## iter  20 value 60.236589
## iter  30 value 58.910427
## iter  40 value 58.812757
## iter  50 value 58.208768
## iter  60 value 58.136179
## iter  70 value 58.134301
## iter  70 value 58.134301
## iter  70 value 58.134301
## final  value 58.134301
## converged
## # weights:  178
## initial  value 138.333129
## iter  10 value 79.541039
## iter  20 value 61.189426
## iter  30 value 56.688572
## iter  40 value 56.257648
## iter  50 value 55.847008
## iter  60 value 55.838369
## final  value 55.838318
## converged
## # weights:  296
## initial  value 134.838236
## iter  10 value 76.420657
## iter  20 value 59.057489
## iter  30 value 55.568802
## iter  40 value 54.362658
## iter  50 value 54.155544
## iter  60 value 54.135473
## iter  70 value 54.133407
## iter  80 value 54.133272
## final  value 54.133268
## converged
## # weights:  60
## initial  value 135.720714
## iter  10 value 110.789892
## iter  20 value 110.789489
## iter  30 value 110.789048
## iter  40 value 110.788563
## iter  50 value 110.788024
## iter  60 value 110.787422
## iter  70 value 110.786741
## iter  80 value 110.785961
## iter  90 value 110.785058
## iter 100 value 110.783995
## final  value 110.783995
## stopped after 100 iterations
## # weights:  178
## initial  value 123.378233
## iter  10 value 110.765688
## iter  20 value 110.709914
## iter  30 value 110.123752
## iter  40 value 54.972985
## iter  50 value 51.836905
## iter  60 value 51.683795
## iter  70 value 51.515530
## iter  80 value 50.391524
## iter  90 value 50.364623
## iter 100 value 50.345001
## final  value 50.345001
## stopped after 100 iterations
## # weights:  296
## initial  value 141.732263
## iter  10 value 110.793343
## iter  20 value 110.778974
## iter  30 value 110.660371
## iter  40 value 86.334223
## iter  50 value 79.180988
## iter  60 value 74.266659
## iter  70 value 73.752894
## iter  80 value 66.339096
## iter  90 value 66.199043
## iter 100 value 65.804161
## final  value 65.804161
## stopped after 100 iterations
## # weights:  60
## initial  value 148.113088
## final  value 111.535550
## converged
## # weights:  178
## initial  value 145.117085
## final  value 111.535550
## converged
## # weights:  296
## initial  value 143.304121
## final  value 111.535550
## converged
## # weights:  60
## initial  value 156.555144
## iter  10 value 84.514917
## iter  20 value 71.569678
## iter  30 value 67.748206
## iter  40 value 67.227625
## iter  50 value 65.939477
## iter  60 value 65.859060
## iter  70 value 65.855550
## final  value 65.855547
## converged
## # weights:  178
## initial  value 136.224260
## iter  10 value 80.261554
## iter  20 value 66.092943
## iter  30 value 62.951071
## iter  40 value 62.496674
## iter  50 value 62.353414
## iter  60 value 62.327894
## iter  70 value 62.326304
## iter  80 value 62.324749
## iter  90 value 62.324532
## final  value 62.324522
## converged
## # weights:  296
## initial  value 146.680123
## iter  10 value 83.112685
## iter  20 value 65.941609
## iter  30 value 63.912779
## iter  40 value 63.373760
## iter  50 value 63.258548
## iter  60 value 63.227053
## iter  70 value 63.223925
## iter  80 value 63.219998
## iter  90 value 63.219416
## iter 100 value 63.219165
## final  value 63.219165
## stopped after 100 iterations
## # weights:  60
## initial  value 124.833859
## iter  10 value 111.555435
## iter  20 value 111.541104
## iter  30 value 111.540732
## iter  40 value 111.540332
## iter  50 value 111.539903
## iter  60 value 111.539442
## iter  70 value 111.538945
## iter  80 value 111.538409
## iter  90 value 111.537829
## iter 100 value 111.537198
## final  value 111.537198
## stopped after 100 iterations
## # weights:  178
## initial  value 161.920614
## iter  10 value 111.553035
## iter  20 value 95.602727
## iter  30 value 59.241038
## iter  40 value 57.511543
## iter  50 value 57.164501
## iter  60 value 56.286448
## iter  70 value 56.276988
## iter  80 value 56.272791
## iter  90 value 56.210819
## iter 100 value 56.128738
## final  value 56.128738
## stopped after 100 iterations
## # weights:  296
## initial  value 127.151680
## iter  10 value 111.548190
## iter  20 value 111.542780
## iter  30 value 111.542326
## iter  40 value 111.541751
## iter  50 value 111.541013
## iter  60 value 111.540032
## iter  70 value 111.538624
## iter  80 value 111.536341
## iter  90 value 111.531779
## iter 100 value 111.518732
## final  value 111.518732
## stopped after 100 iterations
## # weights:  60
## initial  value 181.776571
## final  value 145.337517
## converged
## # weights:  178
## initial  value 181.869205
## final  value 145.337517
## converged
## # weights:  296
## initial  value 143.951731
## iter  10 value 83.319117
## iter  20 value 82.231265
## iter  30 value 81.884397
## iter  40 value 81.841650
## iter  50 value 81.741808
## iter  60 value 81.589500
## iter  70 value 81.573077
## iter  80 value 81.561944
## iter  90 value 81.559009
## iter 100 value 81.558159
## final  value 81.558159
## stopped after 100 iterations
## # weights:  60
## initial  value 155.853743
## iter  10 value 97.055107
## iter  20 value 92.477952
## iter  30 value 91.461080
## iter  40 value 90.965453
## iter  50 value 90.955164
## iter  60 value 90.935642
## final  value 90.932132
## converged
## # weights:  178
## initial  value 166.778734
## iter  10 value 96.910061
## iter  20 value 88.587100
## iter  30 value 85.988713
## iter  40 value 85.057806
## iter  50 value 84.804151
## iter  60 value 84.728888
## iter  70 value 84.725088
## iter  80 value 84.724202
## iter  90 value 84.723368
## iter 100 value 84.723169
## final  value 84.723169
## stopped after 100 iterations
## # weights:  296
## initial  value 198.690489
## iter  10 value 104.259596
## iter  20 value 89.866836
## iter  30 value 87.064758
## iter  40 value 86.091774
## iter  50 value 85.926309
## iter  60 value 85.865185
## iter  70 value 85.833350
## iter  80 value 85.814931
## iter  90 value 85.813828
## iter 100 value 85.812805
## final  value 85.812805
## stopped after 100 iterations
## # weights:  60
## initial  value 150.614189
## iter  10 value 145.015189
## iter  20 value 89.209391
## iter  30 value 83.175383
## iter  40 value 82.977342
## iter  50 value 82.973476
## iter  60 value 82.970022
## iter  70 value 82.858768
## iter  80 value 82.846047
## iter  90 value 82.841032
## iter 100 value 82.838131
## final  value 82.838131
## stopped after 100 iterations
## # weights:  178
## initial  value 170.768161
## iter  10 value 145.351392
## iter  20 value 145.349933
## iter  30 value 145.348123
## iter  40 value 145.345799
## iter  50 value 145.342715
## iter  60 value 145.338469
## iter  70 value 145.332191
## iter  80 value 145.321688
## iter  90 value 145.299853
## iter 100 value 145.224687
## final  value 145.224687
## stopped after 100 iterations
## # weights:  296
## initial  value 166.363171
## iter  10 value 145.355437
## iter  20 value 145.343050
## iter  30 value 92.044827
## iter  40 value 80.653268
## iter  50 value 80.301636
## iter  60 value 79.703345
## iter  70 value 79.609287
## iter  80 value 79.510267
## iter  90 value 79.484632
## iter 100 value 79.469318
## final  value 79.469318
## stopped after 100 iterations
## # weights:  60
## initial  value 174.773008
## final  value 143.064074
## converged
## # weights:  178
## initial  value 176.681649
## final  value 143.064074
## converged
## # weights:  296
## initial  value 195.881704
## final  value 143.064074
## converged
## # weights:  60
## initial  value 181.294881
## iter  10 value 108.027898
## iter  20 value 98.707361
## iter  30 value 95.310955
## iter  40 value 94.334734
## iter  50 value 94.204835
## iter  60 value 94.201133
## final  value 94.201130
## converged
## # weights:  178
## initial  value 179.174043
## iter  10 value 99.846334
## iter  20 value 92.206549
## iter  30 value 91.774180
## iter  40 value 91.671668
## iter  50 value 91.661835
## iter  60 value 91.661385
## iter  70 value 91.661172
## final  value 91.661152
## converged
## # weights:  296
## initial  value 175.732185
## iter  10 value 110.341082
## iter  20 value 92.481877
## iter  30 value 90.713957
## iter  40 value 90.480468
## iter  50 value 90.425496
## iter  60 value 90.383544
## iter  70 value 90.381714
## iter  80 value 90.381636
## iter  90 value 90.379520
## iter 100 value 90.378988
## final  value 90.378988
## stopped after 100 iterations
## # weights:  60
## initial  value 166.990597
## iter  10 value 143.067314
## iter  20 value 143.066285
## iter  30 value 143.065045
## iter  40 value 143.063518
## iter  50 value 143.061585
## iter  60 value 143.059050
## iter  70 value 143.055571
## iter  80 value 143.050478
## iter  90 value 143.042274
## iter 100 value 143.026757
## final  value 143.026757
## stopped after 100 iterations
## # weights:  178
## initial  value 168.769996
## iter  10 value 143.078181
## iter  20 value 143.077831
## iter  30 value 143.077455
## iter  40 value 143.077049
## iter  50 value 143.076608
## iter  60 value 143.076124
## iter  70 value 143.075590
## iter  80 value 143.074994
## iter  90 value 143.074323
## iter 100 value 143.073556
## final  value 143.073556
## stopped after 100 iterations
## # weights:  296
## initial  value 168.936341
## iter  10 value 143.073110
## iter  20 value 97.142341
## iter  30 value 90.965991
## iter  40 value 89.933203
## iter  50 value 88.910535
## iter  60 value 88.846388
## iter  70 value 88.812875
## iter  80 value 88.803529
## iter  90 value 88.495830
## iter 100 value 88.256801
## final  value 88.256801
## stopped after 100 iterations
## # weights:  60
## initial  value 138.260947
## final  value 131.801757
## converged
## # weights:  178
## initial  value 141.604088
## final  value 131.801815
## converged
## # weights:  296
## initial  value 175.752055
## final  value 131.801815
## converged
## # weights:  60
## initial  value 149.172494
## iter  10 value 94.729193
## iter  20 value 84.430060
## iter  30 value 80.327851
## iter  40 value 77.809102
## iter  50 value 77.529677
## iter  60 value 77.504886
## final  value 77.504847
## converged
## # weights:  178
## initial  value 137.266172
## iter  10 value 83.241424
## iter  20 value 76.538862
## iter  30 value 74.118106
## iter  40 value 73.808411
## iter  50 value 73.464952
## iter  60 value 73.419749
## iter  70 value 73.411253
## iter  80 value 73.389536
## iter  90 value 73.387761
## final  value 73.387692
## converged
## # weights:  296
## initial  value 168.672312
## iter  10 value 94.201811
## iter  20 value 79.946866
## iter  30 value 75.761222
## iter  40 value 74.300040
## iter  50 value 73.167837
## iter  60 value 73.026479
## iter  70 value 72.942206
## iter  80 value 72.930807
## iter  90 value 72.930130
## iter 100 value 72.929665
## final  value 72.929665
## stopped after 100 iterations
## # weights:  60
## initial  value 175.930692
## iter  10 value 131.821657
## iter  20 value 131.814654
## iter  30 value 131.814178
## iter  40 value 131.813679
## iter  50 value 131.813146
## iter  60 value 131.812562
## iter  70 value 131.811903
## iter  80 value 131.811129
## iter  90 value 131.810189
## iter 100 value 131.809042
## final  value 131.809042
## stopped after 100 iterations
## # weights:  178
## initial  value 142.340054
## iter  10 value 131.820464
## iter  20 value 131.818558
## iter  30 value 131.816050
## iter  40 value 131.813384
## iter  50 value 131.810397
## iter  60 value 131.806628
## iter  70 value 131.801299
## iter  80 value 131.792798
## iter  90 value 131.776667
## iter 100 value 131.733322
## final  value 131.733322
## stopped after 100 iterations
## # weights:  296
## initial  value 146.118608
## iter  10 value 131.817873
## iter  20 value 131.809262
## iter  30 value 131.802764
## iter  40 value 131.794284
## iter  50 value 131.778924
## iter  60 value 131.718897
## iter  70 value 80.438595
## iter  80 value 72.227322
## iter  90 value 71.678628
## iter 100 value 69.925565
## final  value 69.925565
## stopped after 100 iterations
## # weights:  60
## initial  value 184.009125
## final  value 133.263981
## converged
## # weights:  178
## initial  value 152.726831
## final  value 133.263981
## converged
## # weights:  296
## initial  value 153.708974
## final  value 133.263981
## converged
## # weights:  60
## initial  value 157.861770
## iter  10 value 107.750495
## iter  20 value 92.953617
## iter  30 value 89.336987
## iter  40 value 87.520725
## iter  50 value 86.881407
## iter  60 value 86.656911
## final  value 86.655489
## converged
## # weights:  178
## initial  value 183.524237
## iter  10 value 108.872204
## iter  20 value 90.376848
## iter  30 value 87.223766
## iter  40 value 85.254958
## iter  50 value 83.214067
## iter  60 value 82.831769
## iter  70 value 82.808170
## iter  80 value 82.800759
## iter  90 value 82.800333
## iter 100 value 82.798724
## final  value 82.798724
## stopped after 100 iterations
## # weights:  296
## initial  value 175.584864
## iter  10 value 106.157978
## iter  20 value 85.210206
## iter  30 value 83.886088
## iter  40 value 83.774191
## iter  50 value 83.761625
## iter  60 value 83.754455
## final  value 83.754349
## converged
## # weights:  60
## initial  value 200.319270
## iter  10 value 133.279642
## iter  20 value 133.273870
## iter  30 value 133.271864
## iter  40 value 133.156751
## iter  50 value 97.921034
## iter  60 value 81.555161
## iter  70 value 77.972400
## iter  80 value 77.896650
## iter  90 value 77.888740
## iter 100 value 77.862130
## final  value 77.862130
## stopped after 100 iterations
## # weights:  178
## initial  value 164.883403
## iter  10 value 133.279119
## iter  20 value 133.278813
## iter  30 value 133.278489
## iter  40 value 133.278146
## iter  50 value 133.277781
## iter  60 value 133.277393
## iter  70 value 133.276978
## iter  80 value 133.276530
## iter  90 value 133.276044
## iter 100 value 133.275514
## final  value 133.275514
## stopped after 100 iterations
## # weights:  296
## initial  value 161.161927
## iter  10 value 133.274350
## iter  20 value 133.269226
## iter  30 value 133.268602
## iter  40 value 133.267597
## iter  50 value 133.265629
## iter  60 value 133.260989
## iter  70 value 133.250450
## iter  80 value 133.219004
## iter  90 value 125.496264
## iter 100 value 80.734713
## final  value 80.734713
## stopped after 100 iterations
## # weights:  60
## initial  value 139.289941
## final  value 113.388734
## converged
## # weights:  178
## initial  value 159.415392
## final  value 113.388734
## converged
## # weights:  296
## initial  value 157.924706
## final  value 113.388734
## converged
## # weights:  60
## initial  value 158.693071
## iter  10 value 92.294823
## iter  20 value 84.618266
## iter  30 value 81.965763
## iter  40 value 80.027005
## iter  50 value 79.939616
## final  value 79.939411
## converged
## # weights:  178
## initial  value 140.418674
## iter  10 value 88.021358
## iter  20 value 81.842120
## iter  30 value 78.681442
## iter  40 value 77.717065
## iter  50 value 77.487255
## iter  60 value 77.426775
## iter  70 value 77.422786
## iter  80 value 77.422477
## final  value 77.422469
## converged
## # weights:  296
## initial  value 165.535946
## iter  10 value 97.494707
## iter  20 value 82.696975
## iter  30 value 79.238579
## iter  40 value 77.218711
## iter  50 value 77.062680
## iter  60 value 77.049897
## iter  70 value 77.036654
## iter  80 value 77.036210
## iter  90 value 77.036161
## final  value 77.036159
## converged
## # weights:  60
## initial  value 166.189958
## iter  10 value 113.404199
## iter  20 value 113.401138
## iter  30 value 113.382158
## iter  40 value 113.378200
## iter  50 value 113.372235
## iter  60 value 113.362123
## iter  70 value 113.341011
## iter  80 value 113.269326
## iter  90 value 99.035456
## iter 100 value 84.807476
## final  value 84.807476
## stopped after 100 iterations
## # weights:  178
## initial  value 169.886711
## iter  10 value 113.401477
## iter  20 value 113.393225
## iter  30 value 113.376935
## iter  40 value 113.255804
## iter  50 value 79.882365
## iter  60 value 73.399362
## iter  70 value 70.859908
## iter  80 value 70.504168
## iter  90 value 70.416897
## iter 100 value 70.390989
## final  value 70.390989
## stopped after 100 iterations
## # weights:  296
## initial  value 134.474839
## iter  10 value 113.393804
## iter  20 value 113.393080
## iter  30 value 113.392140
## iter  40 value 113.390858
## iter  50 value 113.388993
## iter  60 value 113.385857
## iter  70 value 113.380445
## iter  80 value 113.371023
## iter  90 value 113.342510
## iter 100 value 88.885979
## final  value 88.885979
## stopped after 100 iterations
## # weights:  60
## initial  value 145.021439
## final  value 118.724424
## converged
## # weights:  178
## initial  value 188.896637
## final  value 118.724424
## converged
## # weights:  296
## initial  value 157.323671
## final  value 118.724424
## converged
## # weights:  60
## initial  value 182.144353
## iter  10 value 93.069991
## iter  20 value 86.410864
## iter  30 value 83.502132
## iter  40 value 82.819543
## iter  50 value 82.809890
## final  value 82.809885
## converged
## # weights:  178
## initial  value 158.901052
## iter  10 value 94.501950
## iter  20 value 81.845022
## iter  30 value 78.488711
## iter  40 value 78.283946
## iter  50 value 78.267683
## iter  60 value 78.266405
## iter  70 value 78.266051
## final  value 78.266036
## converged
## # weights:  296
## initial  value 174.945795
## iter  10 value 106.603328
## iter  20 value 80.887562
## iter  30 value 78.078284
## iter  40 value 77.791137
## iter  50 value 77.684926
## iter  60 value 77.659723
## iter  70 value 77.609261
## iter  80 value 77.586141
## iter  90 value 77.585167
## iter 100 value 77.585067
## final  value 77.585067
## stopped after 100 iterations
## # weights:  60
## initial  value 166.154310
## iter  10 value 118.743596
## iter  20 value 118.737144
## final  value 118.734833
## converged
## # weights:  178
## initial  value 172.013418
## iter  10 value 118.738755
## iter  20 value 116.993118
## iter  30 value 75.937415
## iter  40 value 73.742646
## iter  50 value 71.227487
## iter  60 value 70.846993
## iter  70 value 70.815772
## iter  80 value 70.810602
## iter  90 value 70.702603
## iter 100 value 70.697218
## final  value 70.697218
## stopped after 100 iterations
## # weights:  296
## initial  value 171.730989
## iter  10 value 118.744873
## iter  20 value 118.656580
## iter  30 value 80.002323
## iter  40 value 79.695294
## iter  50 value 79.251453
## iter  60 value 76.058110
## iter  70 value 75.864023
## iter  80 value 74.517500
## iter  90 value 74.274810
## iter 100 value 74.129143
## final  value 74.129143
## stopped after 100 iterations
## # weights:  60
## initial  value 186.110176
## final  value 109.855667
## converged
## # weights:  178
## initial  value 128.839743
## final  value 109.855667
## converged
## # weights:  296
## initial  value 131.125067
## final  value 109.855667
## converged
## # weights:  60
## initial  value 155.751991
## iter  10 value 100.277713
## iter  20 value 81.715729
## iter  30 value 75.790914
## iter  40 value 75.298564
## iter  50 value 75.246101
## final  value 75.245972
## converged
## # weights:  178
## initial  value 162.567528
## iter  10 value 85.334160
## iter  20 value 73.818580
## iter  30 value 73.333944
## iter  40 value 73.173337
## iter  50 value 73.144720
## final  value 73.144487
## converged
## # weights:  296
## initial  value 157.095965
## iter  10 value 86.463141
## iter  20 value 73.779182
## iter  30 value 73.034642
## iter  40 value 72.943623
## iter  50 value 72.938515
## iter  60 value 72.938295
## final  value 72.938293
## converged
## # weights:  60
## initial  value 156.048261
## iter  10 value 109.870813
## iter  20 value 109.861960
## iter  30 value 102.544661
## iter  40 value 96.627910
## iter  50 value 94.912038
## iter  60 value 94.903210
## iter  70 value 94.812553
## iter  80 value 90.794094
## iter  90 value 87.916746
## iter 100 value 87.847291
## final  value 87.847291
## stopped after 100 iterations
## # weights:  178
## initial  value 134.351854
## iter  10 value 109.869498
## iter  20 value 83.970868
## iter  30 value 66.618439
## iter  40 value 66.242093
## iter  50 value 66.131203
## iter  60 value 66.116424
## iter  70 value 65.987269
## iter  80 value 65.958577
## iter  90 value 65.953170
## iter 100 value 65.946573
## final  value 65.946573
## stopped after 100 iterations
## # weights:  296
## initial  value 189.104124
## iter  10 value 109.871622
## iter  20 value 104.218545
## iter  30 value 81.727719
## iter  40 value 81.599631
## iter  50 value 81.480786
## iter  60 value 77.744483
## iter  70 value 77.241535
## iter  80 value 77.092946
## iter  90 value 77.066576
## iter 100 value 76.903381
## final  value 76.903381
## stopped after 100 iterations
## # weights:  60
## initial  value 164.226056
## final  value 97.748603
## converged
## # weights:  178
## initial  value 121.132444
## final  value 97.748603
## converged
## # weights:  296
## initial  value 144.017104
## final  value 97.748603
## converged
## # weights:  60
## initial  value 167.626915
## iter  10 value 83.223329
## iter  20 value 75.701093
## iter  30 value 74.164476
## iter  40 value 72.768485
## iter  50 value 70.974647
## iter  60 value 70.886570
## final  value 70.885772
## converged
## # weights:  178
## initial  value 171.469773
## iter  10 value 87.357656
## iter  20 value 74.227670
## iter  30 value 70.937289
## iter  40 value 69.435873
## iter  50 value 68.985208
## iter  60 value 68.794145
## iter  70 value 68.772730
## iter  80 value 68.771453
## iter  90 value 68.771255
## final  value 68.771242
## converged
## # weights:  296
## initial  value 132.689817
## iter  10 value 82.576172
## iter  20 value 70.427869
## iter  30 value 68.550304
## iter  40 value 67.534431
## iter  50 value 67.309237
## iter  60 value 67.283753
## iter  70 value 67.282857
## final  value 67.282848
## converged
## # weights:  60
## initial  value 173.877571
## iter  10 value 97.765155
## iter  20 value 97.763688
## iter  30 value 97.760923
## iter  40 value 97.760860
## iter  50 value 97.760793
## iter  60 value 97.760721
## iter  70 value 97.760643
## iter  80 value 97.760558
## iter  90 value 97.760465
## iter 100 value 97.760364
## final  value 97.760364
## stopped after 100 iterations
## # weights:  178
## initial  value 152.874747
## iter  10 value 97.761650
## iter  20 value 97.755598
## iter  30 value 97.755009
## iter  40 value 97.754272
## iter  50 value 97.753088
## iter  60 value 97.737557
## iter  70 value 97.277356
## iter  80 value 65.440542
## iter  90 value 64.966982
## iter 100 value 64.161437
## final  value 64.161437
## stopped after 100 iterations
## # weights:  296
## initial  value 163.418309
## iter  10 value 97.764497
## iter  20 value 97.749144
## iter  30 value 97.738271
## iter  40 value 97.392482
## iter  50 value 64.834613
## iter  60 value 63.326431
## iter  70 value 62.673905
## iter  80 value 62.593769
## iter  90 value 62.354861
## iter 100 value 62.318675
## final  value 62.318675
## stopped after 100 iterations
## # weights:  60
## initial  value 142.872384
## final  value 128.793557
## converged
## # weights:  178
## initial  value 135.197497
## final  value 128.793557
## converged
## # weights:  296
## initial  value 137.754192
## final  value 128.793557
## converged
## # weights:  60
## initial  value 132.442359
## iter  10 value 84.856177
## iter  20 value 80.560448
## iter  30 value 76.401523
## iter  40 value 75.909349
## iter  50 value 75.807991
## final  value 75.807788
## converged
## # weights:  178
## initial  value 152.126524
## iter  10 value 94.769113
## iter  20 value 77.547680
## iter  30 value 74.856109
## iter  40 value 74.144952
## iter  50 value 73.709279
## iter  60 value 73.528741
## iter  70 value 73.513494
## iter  80 value 73.513218
## iter  80 value 73.513217
## iter  80 value 73.513217
## final  value 73.513217
## converged
## # weights:  296
## initial  value 139.064980
## iter  10 value 83.909448
## iter  20 value 75.481878
## iter  30 value 73.908697
## iter  40 value 73.562398
## iter  50 value 73.346848
## iter  60 value 73.274148
## iter  70 value 73.265649
## iter  80 value 73.265085
## final  value 73.265060
## converged
## # weights:  60
## initial  value 156.629530
## iter  10 value 128.813098
## iter  20 value 128.806486
## iter  30 value 128.788864
## iter  40 value 128.784779
## iter  50 value 128.778615
## iter  60 value 128.768174
## iter  70 value 128.746321
## iter  80 value 128.670717
## iter  90 value 117.806334
## iter 100 value 88.788443
## final  value 88.788443
## stopped after 100 iterations
## # weights:  178
## initial  value 130.753256
## iter  10 value 128.815147
## iter  20 value 128.787983
## iter  30 value 128.218721
## iter  40 value 84.188390
## iter  50 value 78.167696
## iter  60 value 74.997237
## iter  70 value 74.735772
## iter  80 value 74.650828
## iter  90 value 74.639245
## iter 100 value 74.589219
## final  value 74.589219
## stopped after 100 iterations
## # weights:  296
## initial  value 133.631278
## iter  10 value 128.798169
## iter  20 value 128.775359
## iter  30 value 108.993604
## iter  40 value 91.505387
## iter  50 value 80.997516
## iter  60 value 75.423304
## iter  70 value 72.930909
## iter  80 value 72.692487
## iter  90 value 72.659602
## iter 100 value 72.629630
## final  value 72.629630
## stopped after 100 iterations
## # weights:  60
## initial  value 145.691426
## final  value 123.780074
## converged
## # weights:  178
## initial  value 149.460380
## final  value 123.780074
## converged
## # weights:  296
## initial  value 176.983085
## final  value 123.780074
## converged
## # weights:  60
## initial  value 140.947702
## iter  10 value 100.253043
## iter  20 value 88.327515
## iter  30 value 83.385732
## iter  40 value 82.972564
## iter  50 value 82.866662
## iter  60 value 82.421734
## iter  70 value 82.421021
## iter  70 value 82.421021
## iter  70 value 82.421021
## final  value 82.421021
## converged
## # weights:  178
## initial  value 147.687932
## iter  10 value 94.886488
## iter  20 value 84.273040
## iter  30 value 81.181986
## iter  40 value 80.119671
## iter  50 value 79.818474
## iter  60 value 79.796113
## iter  70 value 79.795880
## final  value 79.795876
## converged
## # weights:  296
## initial  value 177.186375
## iter  10 value 92.790822
## iter  20 value 81.683812
## iter  30 value 80.144148
## iter  40 value 79.606902
## iter  50 value 79.412225
## iter  60 value 79.316121
## iter  70 value 79.288447
## iter  80 value 79.269402
## iter  90 value 79.268411
## iter 100 value 79.267771
## final  value 79.267771
## stopped after 100 iterations
## # weights:  60
## initial  value 147.693161
## iter  10 value 123.797607
## iter  20 value 123.797465
## iter  30 value 123.797319
## iter  40 value 123.797172
## iter  50 value 123.797021
## iter  60 value 123.796867
## iter  70 value 123.796710
## iter  80 value 123.796549
## iter  90 value 123.796385
## iter 100 value 123.796216
## final  value 123.796216
## stopped after 100 iterations
## # weights:  178
## initial  value 143.013820
## iter  10 value 123.744835
## iter  20 value 123.489969
## iter  30 value 77.711733
## iter  40 value 75.621187
## iter  50 value 75.557064
## iter  60 value 75.549732
## iter  70 value 75.547560
## iter  80 value 75.495021
## iter  90 value 75.413386
## iter 100 value 73.900969
## final  value 73.900969
## stopped after 100 iterations
## # weights:  296
## initial  value 129.319646
## iter  10 value 123.772937
## iter  20 value 123.684515
## iter  30 value 83.781545
## iter  40 value 78.280694
## iter  50 value 73.327623
## iter  60 value 72.690263
## iter  70 value 72.558142
## iter  80 value 72.227772
## iter  90 value 72.142289
## iter 100 value 71.545219
## final  value 71.545219
## stopped after 100 iterations
## # weights:  60
## initial  value 151.908023
## final  value 106.922881
## converged
## # weights:  178
## initial  value 132.969844
## final  value 106.922881
## converged
## # weights:  296
## initial  value 172.155040
## final  value 106.922881
## converged
## # weights:  60
## initial  value 149.585390
## iter  10 value 103.712356
## iter  20 value 90.429418
## iter  30 value 82.055336
## iter  40 value 78.426929
## iter  50 value 78.065661
## iter  60 value 77.923777
## iter  70 value 77.919362
## final  value 77.919345
## converged
## # weights:  178
## initial  value 179.051341
## iter  10 value 99.451844
## iter  20 value 77.870210
## iter  30 value 76.307973
## iter  40 value 76.102199
## iter  50 value 76.077733
## iter  60 value 76.077357
## iter  70 value 76.077197
## final  value 76.077195
## converged
## # weights:  296
## initial  value 162.031124
## iter  10 value 99.958534
## iter  20 value 81.361885
## iter  30 value 78.935328
## iter  40 value 76.113329
## iter  50 value 76.009308
## iter  60 value 75.869239
## iter  70 value 75.782110
## iter  80 value 75.747830
## iter  90 value 75.745660
## final  value 75.745653
## converged
## # weights:  60
## initial  value 161.237381
## iter  10 value 106.838861
## iter  20 value 94.185265
## iter  30 value 71.381289
## iter  40 value 69.529914
## iter  50 value 69.220039
## iter  60 value 69.168526
## iter  70 value 69.074411
## iter  80 value 69.014212
## iter  90 value 69.011152
## iter 100 value 69.009819
## final  value 69.009819
## stopped after 100 iterations
## # weights:  178
## initial  value 156.366314
## iter  10 value 106.934558
## iter  20 value 106.879236
## iter  30 value 89.204401
## iter  40 value 72.756637
## iter  50 value 71.233226
## iter  60 value 70.583484
## iter  70 value 70.541445
## iter  80 value 70.524984
## iter  90 value 70.498042
## iter 100 value 70.467415
## final  value 70.467415
## stopped after 100 iterations
## # weights:  296
## initial  value 131.668012
## iter  10 value 106.907481
## iter  20 value 106.853515
## iter  30 value 106.026327
## iter  40 value 78.011730
## iter  50 value 72.009302
## iter  60 value 69.805221
## iter  70 value 69.508036
## iter  80 value 68.858350
## iter  90 value 68.768420
## iter 100 value 68.735624
## final  value 68.735624
## stopped after 100 iterations
## # weights:  60
## initial  value 142.357258
## final  value 106.903656
## converged
## # weights:  178
## initial  value 125.552936
## final  value 106.903656
## converged
## # weights:  296
## initial  value 136.824222
## final  value 106.903656
## converged
## # weights:  60
## initial  value 140.476288
## iter  10 value 94.726039
## iter  20 value 75.516994
## iter  30 value 68.422885
## iter  40 value 63.361709
## iter  50 value 63.105978
## iter  60 value 62.726765
## iter  70 value 62.445603
## iter  80 value 62.034019
## iter  90 value 61.935613
## iter 100 value 61.724151
## final  value 61.724151
## stopped after 100 iterations
## # weights:  178
## initial  value 159.433741
## iter  10 value 74.447051
## iter  20 value 61.932960
## iter  30 value 60.963926
## iter  40 value 59.577046
## iter  50 value 59.125196
## iter  60 value 59.121496
## iter  70 value 59.120874
## final  value 59.120784
## converged
## # weights:  296
## initial  value 139.768945
## iter  10 value 74.627299
## iter  20 value 61.289732
## iter  30 value 57.820990
## iter  40 value 57.284592
## iter  50 value 57.144836
## iter  60 value 57.114363
## iter  70 value 57.111296
## final  value 57.111268
## converged
## # weights:  60
## initial  value 121.031540
## iter  10 value 106.893008
## iter  20 value 106.882975
## iter  30 value 106.864164
## iter  40 value 106.806441
## iter  50 value 103.669793
## iter  60 value 65.775990
## iter  70 value 63.754937
## iter  80 value 63.274640
## iter  90 value 63.217140
## iter 100 value 63.163567
## final  value 63.163567
## stopped after 100 iterations
## # weights:  178
## initial  value 168.410076
## iter  10 value 106.920184
## iter  20 value 106.919858
## iter  30 value 106.919595
## iter  40 value 106.916979
## iter  50 value 105.877059
## iter  60 value 59.891366
## iter  70 value 58.511502
## iter  80 value 58.261831
## iter  90 value 58.056220
## iter 100 value 57.929362
## final  value 57.929362
## stopped after 100 iterations
## # weights:  296
## initial  value 114.109874
## iter  10 value 70.290917
## iter  20 value 61.666918
## iter  30 value 59.771340
## iter  40 value 59.273518
## iter  50 value 58.783373
## iter  60 value 58.604113
## iter  70 value 57.984375
## iter  80 value 57.935111
## iter  90 value 57.890430
## iter 100 value 57.141566
## final  value 57.141566
## stopped after 100 iterations
## # weights:  60
## initial  value 182.898574
## final  value 139.171930
## converged
## # weights:  178
## initial  value 196.040336
## final  value 139.171930
## converged
## # weights:  296
## initial  value 180.910072
## final  value 139.171930
## converged
## # weights:  60
## initial  value 244.518833
## iter  10 value 139.002924
## iter  20 value 125.463387
## iter  30 value 113.337994
## iter  40 value 107.875994
## iter  50 value 106.804479
## iter  60 value 106.725470
## final  value 106.725259
## converged
## # weights:  178
## initial  value 179.508169
## iter  10 value 112.618068
## iter  20 value 104.852449
## iter  30 value 104.213797
## iter  40 value 104.103233
## iter  50 value 104.099788
## final  value 104.099742
## converged
## # weights:  296
## initial  value 179.134552
## iter  10 value 114.494351
## iter  20 value 106.855663
## iter  30 value 105.016942
## iter  40 value 104.042678
## iter  50 value 103.125369
## iter  60 value 102.806250
## iter  70 value 102.603273
## iter  80 value 102.572866
## iter  90 value 102.556670
## iter 100 value 102.553679
## final  value 102.553679
## stopped after 100 iterations
## # weights:  60
## initial  value 183.037942
## iter  10 value 139.182711
## iter  20 value 122.947223
## iter  30 value 113.534271
## iter  40 value 112.215968
## iter  50 value 104.653253
## iter  60 value 103.959912
## iter  70 value 103.942349
## iter  80 value 103.937665
## iter  90 value 102.528126
## iter 100 value 102.402525
## final  value 102.402525
## stopped after 100 iterations
## # weights:  178
## initial  value 196.295532
## iter  10 value 139.189471
## iter  20 value 138.663877
## iter  30 value 100.130732
## iter  40 value 98.014490
## iter  50 value 97.017215
## iter  60 value 97.000742
## iter  70 value 96.997309
## iter  80 value 96.991967
## iter  90 value 96.981883
## iter 100 value 96.964601
## final  value 96.964601
## stopped after 100 iterations
## # weights:  296
## initial  value 210.331367
## iter  10 value 139.191750
## iter  20 value 139.177512
## iter  30 value 139.176995
## iter  40 value 139.176292
## iter  50 value 139.175283
## iter  60 value 139.173711
## iter  70 value 139.170949
## iter  80 value 139.165246
## iter  90 value 139.150277
## iter 100 value 139.072348
## final  value 139.072348
## stopped after 100 iterations
## # weights:  60
## initial  value 131.677040
## final  value 107.144018
## converged
## # weights:  178
## initial  value 125.521209
## final  value 107.144018
## converged
## # weights:  296
## initial  value 126.664962
## final  value 107.144018
## converged
## # weights:  60
## initial  value 139.495992
## iter  10 value 90.261702
## iter  20 value 74.769455
## iter  30 value 71.800408
## iter  40 value 71.402626
## iter  50 value 71.226121
## iter  60 value 70.531654
## iter  70 value 70.496001
## final  value 70.495591
## converged
## # weights:  178
## initial  value 158.656334
## iter  10 value 81.105634
## iter  20 value 69.630650
## iter  30 value 69.057877
## iter  40 value 68.392502
## iter  50 value 68.281028
## iter  60 value 68.268435
## iter  70 value 68.242263
## iter  80 value 68.241275
## iter  90 value 68.241146
## final  value 68.241085
## converged
## # weights:  296
## initial  value 151.172355
## iter  10 value 88.494252
## iter  20 value 68.632905
## iter  30 value 67.863043
## iter  40 value 67.754685
## iter  50 value 67.715027
## iter  60 value 67.712739
## iter  70 value 67.704912
## iter  80 value 67.702342
## final  value 67.702328
## converged
## # weights:  60
## initial  value 154.540761
## iter  10 value 107.133492
## iter  20 value 107.127327
## iter  30 value 107.116209
## iter  40 value 107.091353
## iter  50 value 106.993404
## iter  60 value 79.391017
## iter  70 value 65.542880
## iter  80 value 65.439071
## iter  90 value 65.413707
## iter 100 value 65.410687
## final  value 65.410687
## stopped after 100 iterations
## # weights:  178
## initial  value 122.091101
## iter  10 value 107.116252
## iter  20 value 106.858608
## iter  30 value 72.576661
## iter  40 value 64.407887
## iter  50 value 63.103732
## iter  60 value 62.384891
## iter  70 value 62.294432
## iter  80 value 62.253726
## iter  90 value 62.226373
## iter 100 value 61.944413
## final  value 61.944413
## stopped after 100 iterations
## # weights:  296
## initial  value 139.264488
## iter  10 value 107.153209
## iter  20 value 89.484628
## iter  30 value 62.850982
## iter  40 value 60.585692
## iter  50 value 60.174626
## iter  60 value 60.094470
## iter  70 value 60.080676
## iter  80 value 60.019401
## iter  90 value 59.948107
## iter 100 value 59.928908
## final  value 59.928908
## stopped after 100 iterations
## # weights:  60
## initial  value 154.863332
## final  value 120.914182
## converged
## # weights:  178
## initial  value 178.408863
## final  value 120.914182
## converged
## # weights:  296
## initial  value 185.353190
## final  value 120.914182
## converged
## # weights:  60
## initial  value 158.418834
## iter  10 value 90.816808
## iter  20 value 79.886117
## iter  30 value 77.113404
## iter  40 value 76.155021
## iter  50 value 76.024363
## iter  60 value 76.001152
## iter  70 value 75.991864
## final  value 75.991859
## converged
## # weights:  178
## initial  value 148.875738
## iter  10 value 80.186538
## iter  20 value 73.822865
## iter  30 value 72.766687
## iter  40 value 72.581639
## iter  50 value 72.575329
## iter  60 value 72.574831
## iter  70 value 72.574635
## final  value 72.574633
## converged
## # weights:  296
## initial  value 152.890091
## iter  10 value 85.046445
## iter  20 value 73.301353
## iter  30 value 72.035850
## iter  40 value 71.853466
## iter  50 value 71.598043
## iter  60 value 70.931796
## iter  70 value 70.890178
## iter  80 value 70.885675
## iter  90 value 70.885015
## iter 100 value 70.884620
## final  value 70.884620
## stopped after 100 iterations
## # weights:  60
## initial  value 136.849519
## iter  10 value 120.911189
## iter  20 value 120.908736
## iter  30 value 120.905421
## iter  40 value 120.900709
## iter  50 value 120.893360
## iter  60 value 120.880008
## iter  70 value 120.847491
## iter  80 value 120.652913
## iter  90 value 72.883209
## iter 100 value 67.001350
## final  value 67.001350
## stopped after 100 iterations
## # weights:  178
## initial  value 158.579307
## iter  10 value 120.925068
## iter  20 value 120.625325
## iter  30 value 69.914740
## iter  40 value 67.811532
## iter  50 value 67.590096
## iter  60 value 67.457328
## iter  70 value 67.446959
## iter  80 value 67.441825
## iter  90 value 67.438373
## iter 100 value 66.462038
## final  value 66.462038
## stopped after 100 iterations
## # weights:  296
## initial  value 159.515987
## iter  10 value 120.923858
## iter  20 value 120.917901
## iter  30 value 120.916486
## iter  40 value 120.913534
## iter  50 value 120.903810
## iter  60 value 120.859408
## iter  70 value 120.114912
## iter  80 value 69.900488
## iter  90 value 67.722681
## iter 100 value 66.781436
## final  value 66.781436
## stopped after 100 iterations
## # weights:  60
## initial  value 144.595725
## final  value 130.833202
## converged
## # weights:  178
## initial  value 162.301131
## final  value 130.833202
## converged
## # weights:  296
## initial  value 160.155003
## final  value 130.833202
## converged
## # weights:  60
## initial  value 190.752124
## iter  10 value 99.243148
## iter  20 value 91.869189
## iter  30 value 84.167639
## iter  40 value 82.668467
## iter  50 value 82.515260
## iter  60 value 82.450067
## final  value 82.450055
## converged
## # weights:  178
## initial  value 144.786101
## iter  10 value 96.272166
## iter  20 value 82.272777
## iter  30 value 80.448718
## iter  40 value 79.722711
## iter  50 value 79.096695
## iter  60 value 78.757609
## iter  70 value 78.626258
## iter  80 value 78.551631
## iter  90 value 78.536827
## iter 100 value 78.536700
## final  value 78.536700
## stopped after 100 iterations
## # weights:  296
## initial  value 198.346033
## iter  10 value 103.202607
## iter  20 value 83.573393
## iter  30 value 80.345678
## iter  40 value 78.954207
## iter  50 value 78.075775
## iter  60 value 77.707136
## iter  70 value 77.526414
## iter  80 value 77.501402
## iter  90 value 77.499527
## iter 100 value 77.499281
## final  value 77.499281
## stopped after 100 iterations
## # weights:  60
## initial  value 147.814249
## iter  10 value 101.717262
## iter  20 value 88.179796
## iter  30 value 88.165237
## iter  40 value 88.145059
## iter  50 value 87.779292
## iter  60 value 87.773395
## iter  70 value 87.769619
## iter  80 value 87.765722
## iter  90 value 87.012448
## iter 100 value 84.778126
## final  value 84.778126
## stopped after 100 iterations
## # weights:  178
## initial  value 171.840826
## iter  10 value 130.851343
## iter  20 value 130.844549
## iter  30 value 130.766848
## iter  40 value 113.349785
## iter  50 value 79.218502
## iter  60 value 78.492163
## iter  70 value 77.981998
## iter  80 value 77.759374
## iter  90 value 77.744317
## iter 100 value 73.906827
## final  value 73.906827
## stopped after 100 iterations
## # weights:  296
## initial  value 173.383393
## iter  10 value 130.846514
## iter  20 value 130.835034
## iter  30 value 130.824113
## iter  40 value 130.691582
## iter  50 value 83.674135
## iter  60 value 76.790041
## iter  70 value 73.118928
## iter  80 value 72.762779
## iter  90 value 72.609906
## iter 100 value 72.591074
## final  value 72.591074
## stopped after 100 iterations
## # weights:  60
## initial  value 200.572732
## final  value 133.801472
## converged
## # weights:  178
## initial  value 160.845361
## final  value 133.801472
## converged
## # weights:  296
## initial  value 182.618272
## final  value 133.801472
## converged
## # weights:  60
## initial  value 173.798736
## iter  10 value 131.301647
## iter  20 value 99.606997
## iter  30 value 98.184770
## iter  40 value 96.590899
## iter  50 value 96.463513
## iter  60 value 96.427417
## final  value 96.427371
## converged
## # weights:  178
## initial  value 185.302189
## iter  10 value 103.269985
## iter  20 value 94.548356
## iter  30 value 93.956401
## iter  40 value 93.904150
## iter  50 value 93.900127
## iter  60 value 93.899786
## final  value 93.899761
## converged
## # weights:  296
## initial  value 214.712712
## iter  10 value 124.384964
## iter  20 value 96.057429
## iter  30 value 93.927522
## iter  40 value 93.669537
## iter  50 value 93.539594
## iter  60 value 93.534598
## iter  70 value 93.534319
## final  value 93.534295
## converged
## # weights:  60
## initial  value 191.016494
## iter  10 value 133.808362
## iter  20 value 133.786882
## iter  30 value 133.779121
## iter  40 value 133.764240
## iter  50 value 133.723947
## iter  60 value 133.350149
## iter  70 value 94.906606
## iter  80 value 93.053623
## iter  90 value 92.928916
## iter 100 value 92.774964
## final  value 92.774964
## stopped after 100 iterations
## # weights:  178
## initial  value 205.925056
## iter  10 value 133.814515
## iter  20 value 133.808845
## iter  30 value 133.808495
## iter  40 value 133.808015
## iter  50 value 133.807268
## iter  60 value 133.806086
## iter  70 value 133.804539
## iter  80 value 133.802653
## iter  90 value 133.799960
## iter 100 value 133.795319
## final  value 133.795319
## stopped after 100 iterations
## # weights:  296
## initial  value 192.505715
## iter  10 value 133.825159
## iter  20 value 133.808937
## iter  30 value 98.518895
## iter  40 value 86.927292
## iter  50 value 86.422395
## iter  60 value 86.372817
## iter  70 value 86.332278
## iter  80 value 86.328890
## iter  90 value 86.326403
## iter 100 value 86.324093
## final  value 86.324093
## stopped after 100 iterations
## # weights:  60
## initial  value 173.391896
## final  value 122.706866
## converged
## # weights:  178
## initial  value 140.125316
## final  value 122.706866
## converged
## # weights:  296
## initial  value 198.393822
## final  value 122.706866
## converged
## # weights:  60
## initial  value 140.604214
## iter  10 value 95.849814
## iter  20 value 83.991439
## iter  30 value 78.822215
## iter  40 value 77.869625
## iter  50 value 77.566676
## iter  60 value 77.553424
## iter  70 value 77.552513
## final  value 77.552500
## converged
## # weights:  178
## initial  value 150.438069
## iter  10 value 83.684745
## iter  20 value 75.738017
## iter  30 value 74.830491
## iter  40 value 74.760890
## iter  50 value 74.746764
## iter  60 value 74.721905
## iter  70 value 74.640475
## iter  80 value 74.630246
## iter  90 value 74.629393
## iter 100 value 74.629295
## final  value 74.629295
## stopped after 100 iterations
## # weights:  296
## initial  value 158.427214
## iter  10 value 94.627055
## iter  20 value 76.011361
## iter  30 value 74.718456
## iter  40 value 74.460139
## iter  50 value 74.369542
## iter  60 value 74.253569
## iter  70 value 74.245876
## iter  80 value 74.245535
## iter  90 value 74.245501
## final  value 74.245495
## converged
## # weights:  60
## initial  value 164.619756
## iter  10 value 122.700476
## iter  20 value 90.935341
## iter  30 value 90.226362
## iter  40 value 89.498320
## iter  50 value 89.465710
## iter  60 value 89.460218
## iter  70 value 89.434848
## iter  80 value 88.801343
## iter  90 value 88.792905
## iter 100 value 88.767022
## final  value 88.767022
## stopped after 100 iterations
## # weights:  178
## initial  value 130.321741
## iter  10 value 122.767972
## iter  20 value 122.767670
## iter  30 value 122.767366
## iter  40 value 122.767060
## iter  50 value 122.766751
## iter  60 value 122.766440
## iter  70 value 122.766127
## iter  80 value 122.765809
## iter  90 value 122.765489
## iter 100 value 122.765164
## final  value 122.765164
## stopped after 100 iterations
## # weights:  296
## initial  value 140.339550
## iter  10 value 122.716021
## iter  20 value 122.710209
## iter  30 value 122.705683
## iter  40 value 122.689849
## iter  50 value 122.625446
## iter  60 value 95.876180
## iter  70 value 79.051710
## iter  80 value 70.836541
## iter  90 value 69.758926
## iter 100 value 69.054177
## final  value 69.054177
## stopped after 100 iterations
## # weights:  60
## initial  value 169.769513
## final  value 117.150773
## converged
## # weights:  178
## initial  value 148.277456
## final  value 117.150773
## converged
## # weights:  296
## initial  value 168.280941
## final  value 117.150773
## converged
## # weights:  60
## initial  value 149.924526
## iter  10 value 100.696774
## iter  20 value 84.179037
## iter  30 value 79.464098
## iter  40 value 75.989119
## iter  50 value 74.797594
## iter  60 value 74.119896
## iter  70 value 73.856566
## final  value 73.856444
## converged
## # weights:  178
## initial  value 164.399163
## iter  10 value 103.900809
## iter  20 value 76.156206
## iter  30 value 72.401839
## iter  40 value 71.433634
## iter  50 value 71.376809
## iter  60 value 71.371818
## final  value 71.371650
## converged
## # weights:  296
## initial  value 190.560717
## iter  10 value 97.739637
## iter  20 value 73.778230
## iter  30 value 70.444982
## iter  40 value 69.695949
## iter  50 value 69.618669
## iter  60 value 69.596662
## iter  70 value 69.591161
## iter  80 value 69.590507
## iter  90 value 69.590117
## iter 100 value 69.590044
## final  value 69.590044
## stopped after 100 iterations
## # weights:  60
## initial  value 145.994356
## iter  10 value 117.163779
## iter  20 value 116.886438
## iter  30 value 98.433313
## iter  40 value 82.015977
## iter  50 value 79.963346
## iter  60 value 78.431618
## iter  70 value 78.334836
## iter  80 value 78.324499
## iter  90 value 78.317468
## iter 100 value 77.706011
## final  value 77.706011
## stopped after 100 iterations
## # weights:  178
## initial  value 142.839119
## iter  10 value 117.163386
## iter  20 value 117.156649
## iter  30 value 117.107194
## iter  40 value 116.074636
## iter  50 value 68.504653
## iter  60 value 64.048069
## iter  70 value 63.733319
## iter  80 value 63.617572
## iter  90 value 63.492186
## iter 100 value 63.156960
## final  value 63.156960
## stopped after 100 iterations
## # weights:  296
## initial  value 134.901354
## iter  10 value 116.958967
## iter  20 value 77.156147
## iter  30 value 70.135461
## iter  40 value 69.613839
## iter  50 value 69.493187
## iter  60 value 67.621297
## iter  70 value 67.138300
## iter  80 value 66.343416
## iter  90 value 65.023966
## iter 100 value 64.902413
## final  value 64.902413
## stopped after 100 iterations
## # weights:  60
## initial  value 143.764213
## final  value 128.990871
## converged
## # weights:  178
## initial  value 163.666274
## final  value 128.990871
## converged
## # weights:  296
## initial  value 162.297407
## final  value 128.990871
## converged
## # weights:  60
## initial  value 155.591660
## iter  10 value 109.727636
## iter  20 value 87.967219
## iter  30 value 83.771744
## iter  40 value 83.093820
## iter  50 value 82.956742
## iter  60 value 82.953461
## final  value 82.953424
## converged
## # weights:  178
## initial  value 155.837848
## iter  10 value 110.559181
## iter  20 value 90.093653
## iter  30 value 85.469037
## iter  40 value 82.906909
## iter  50 value 79.545559
## iter  60 value 79.255041
## iter  70 value 79.172462
## iter  80 value 79.169434
## final  value 79.169370
## converged
## # weights:  296
## initial  value 168.967623
## iter  10 value 105.423450
## iter  20 value 82.978534
## iter  30 value 78.777465
## iter  40 value 77.719776
## iter  50 value 77.494938
## iter  60 value 77.422020
## iter  70 value 77.386677
## iter  80 value 77.376058
## iter  90 value 77.374803
## iter 100 value 77.374727
## final  value 77.374727
## stopped after 100 iterations
## # weights:  60
## initial  value 149.156379
## iter  10 value 128.979992
## iter  20 value 128.973818
## iter  30 value 128.963526
## iter  40 value 128.942466
## iter  50 value 128.872989
## iter  60 value 106.073078
## iter  70 value 78.350941
## iter  80 value 74.458338
## iter  90 value 73.893744
## iter 100 value 73.761682
## final  value 73.761682
## stopped after 100 iterations
## # weights:  178
## initial  value 180.409705
## iter  10 value 129.003449
## iter  20 value 128.972205
## iter  30 value 128.947574
## iter  40 value 128.820573
## iter  50 value 85.112076
## iter  60 value 74.041065
## iter  70 value 72.674093
## iter  80 value 71.862132
## iter  90 value 71.622498
## iter 100 value 71.574585
## final  value 71.574585
## stopped after 100 iterations
## # weights:  296
## initial  value 154.070785
## iter  10 value 128.991310
## iter  20 value 128.834732
## iter  30 value 77.616287
## iter  40 value 73.029292
## iter  50 value 72.737682
## iter  60 value 72.698203
## iter  70 value 72.486284
## iter  80 value 72.425353
## iter  90 value 72.420362
## iter 100 value 72.417661
## final  value 72.417661
## stopped after 100 iterations
## # weights:  60
## initial  value 164.027010
## final  value 121.884691
## converged
## # weights:  178
## initial  value 169.595370
## final  value 121.884691
## converged
## # weights:  296
## initial  value 144.768141
## final  value 121.884691
## converged
## # weights:  60
## initial  value 170.598454
## iter  10 value 89.327653
## iter  20 value 83.579912
## iter  30 value 83.122813
## final  value 83.121137
## converged
## # weights:  178
## initial  value 157.476528
## iter  10 value 88.879104
## iter  20 value 81.153489
## iter  30 value 79.396950
## iter  40 value 79.087217
## iter  50 value 79.003032
## iter  60 value 78.999977
## final  value 78.999964
## converged
## # weights:  296
## initial  value 159.585024
## iter  10 value 94.625004
## iter  20 value 82.653221
## iter  30 value 81.167872
## iter  40 value 80.981516
## iter  50 value 80.965076
## iter  60 value 80.964563
## iter  70 value 80.964494
## final  value 80.964493
## converged
## # weights:  60
## initial  value 141.836237
## iter  10 value 121.898910
## iter  20 value 121.892523
## iter  30 value 121.891865
## iter  40 value 121.891170
## iter  50 value 121.890426
## iter  60 value 121.889605
## iter  70 value 121.888667
## iter  80 value 121.887556
## iter  90 value 121.886195
## iter 100 value 121.884467
## final  value 121.884467
## stopped after 100 iterations
## # weights:  178
## initial  value 156.205562
## iter  10 value 121.896076
## iter  20 value 121.895628
## iter  30 value 121.895124
## iter  40 value 121.894556
## iter  50 value 121.893915
## iter  60 value 121.893194
## iter  70 value 121.892381
## iter  80 value 121.891461
## iter  90 value 121.890409
## iter 100 value 121.889187
## final  value 121.889187
## stopped after 100 iterations
## # weights:  296
## initial  value 156.038965
## iter  10 value 121.894182
## iter  20 value 121.882320
## iter  30 value 121.872610
## iter  40 value 96.472835
## iter  50 value 75.837759
## iter  60 value 74.507008
## iter  70 value 73.025054
## iter  80 value 72.344976
## iter  90 value 72.077480
## iter 100 value 72.042995
## final  value 72.042995
## stopped after 100 iterations
## # weights:  296
## initial  value 179.733449
## iter  10 value 97.031521
## iter  20 value 82.604387
## iter  30 value 79.044723
## iter  40 value 76.992855
## iter  50 value 76.444921
## iter  60 value 76.247254
## iter  70 value 76.056877
## iter  80 value 76.035584
## iter  90 value 76.035126
## final  value 76.035088
## converged
```

```
## ### Neuro Network Metrics Results ###
```

```
##      RMSE  Rsquared       MAE
## 0.8993722 0.5441986 0.7042420
```

Table 14: Linear Model Variable Importance

	Overall
BiologicalMaterial01	0.4144884
BiologicalMaterial02	0.4183276
BiologicalMaterial03	0.3296683
BiologicalMaterial04	0.1004094
BiologicalMaterial05	1.4830950
BiologicalMaterial06	0.6722537
BiologicalMaterial07	1.0122836
BiologicalMaterial08	0.4479805
BiologicalMaterial09	0.1656918
BiologicalMaterial10	0.0016398
BiologicalMaterial11	1.1880094
BiologicalMaterial12	1.0740032
ManufacturingProcess01	0.2780557
ManufacturingProcess02	0.0016257
ManufacturingProcess03	1.5058096
ManufacturingProcess04	0.7357273
ManufacturingProcess05	0.8834755
ManufacturingProcess06	1.6381746
ManufacturingProcess07	0.8857378
ManufacturingProcess08	0.4512921
ManufacturingProcess09	0.6382246
ManufacturingProcess10	0.1007782
ManufacturingProcess11	0.7667228
ManufacturingProcess12	0.5933077
ManufacturingProcess13	0.9720934
ManufacturingProcess14	0.2959231
ManufacturingProcess15	0.2557222
ManufacturingProcess16	0.1957528
ManufacturingProcess17	0.0743751
ManufacturingProcess18	0.6963078
ManufacturingProcess19	0.6411591
ManufacturingProcess20	0.7223646
ManufacturingProcess22	0.4755496
ManufacturingProcess23	0.8996397
ManufacturingProcess24	1.2013709
ManufacturingProcess25	0.8672575
ManufacturingProcess26	0.8041314
ManufacturingProcess27	0.6766606
ManufacturingProcess28	3.2011086
ManufacturingProcess29	2.2099760
ManufacturingProcess30	0.2744709
ManufacturingProcess31	0.7048491
ManufacturingProcess32	4.1098153
ManufacturingProcess33	2.5217929
ManufacturingProcess34	0.4899749
ManufacturingProcess35	0.9323066
ManufacturingProcess36	1.0568528
ManufacturingProcess37	1.0046325
ManufacturingProcess38	0.9991690
ManufacturingProcess39	0.1916321

	Overall
ManufacturingProcess40	0.8355733
ManufacturingProcess41	0.8810996
ManufacturingProcess42	0.5277907
ManufacturingProcess43	1.3659767
ManufacturingProcess44	0.4718168
ManufacturingProcess45	1.6214527

Table 15: SVM Variable Importance

	Overall
BiologicalMaterial01	0.1076615
BiologicalMaterial02	0.1807012
BiologicalMaterial03	0.2631946
BiologicalMaterial04	0.1413654
BiologicalMaterial05	0.0367758
BiologicalMaterial06	0.2909056
BiologicalMaterial07	0.0100028
BiologicalMaterial08	0.1473037
BiologicalMaterial09	0.1555370
BiologicalMaterial10	0.0724908
BiologicalMaterial11	0.1629170
BiologicalMaterial12	0.2412185
ManufacturingProcess01	0.1578287
ManufacturingProcess02	0.1331255
ManufacturingProcess03	0.0129271
ManufacturingProcess04	0.1098940
ManufacturingProcess05	0.0161460
ManufacturingProcess06	0.2402510
ManufacturingProcess07	0.0007811
ManufacturingProcess08	0.0000261
ManufacturingProcess09	0.2587624
ManufacturingProcess10	0.0708371
ManufacturingProcess11	0.2043809
ManufacturingProcess12	0.1571690
ManufacturingProcess13	0.2747904
ManufacturingProcess14	0.0641521
ManufacturingProcess15	0.1395217
ManufacturingProcess16	0.0643132
ManufacturingProcess17	0.3063856
ManufacturingProcess18	0.1083749
ManufacturingProcess19	0.0718435
ManufacturingProcess20	0.0401452
ManufacturingProcess21	0.0590100
ManufacturingProcess22	0.0028819
ManufacturingProcess23	0.0132709
ManufacturingProcess24	0.0670870
ManufacturingProcess25	0.1663295
ManufacturingProcess26	0.1076723
ManufacturingProcess27	0.0729450
ManufacturingProcess28	0.0776908

	Overall
ManufacturingProcess29	0.1889376
ManufacturingProcess30	0.1247785
ManufacturingProcess31	0.2174880
ManufacturingProcess32	0.3734379
ManufacturingProcess33	0.1850785
ManufacturingProcess34	0.0183944
ManufacturingProcess35	0.0719678
ManufacturingProcess36	0.2131597
ManufacturingProcess37	0.0537773
ManufacturingProcess38	0.0032073
ManufacturingProcess39	0.0018747
ManufacturingProcess40	0.0008630
ManufacturingProcess41	0.0003102
ManufacturingProcess42	0.0001593
ManufacturingProcess43	0.0288916
ManufacturingProcess44	0.0056120
ManufacturingProcess45	0.0016508

Table 16: KNN Variable Importance

	Overall
BiologicalMaterial01	0.1076615
BiologicalMaterial02	0.1807012
BiologicalMaterial03	0.2631946
BiologicalMaterial04	0.1413654
BiologicalMaterial05	0.0367758
BiologicalMaterial06	0.2909056
BiologicalMaterial07	0.0100028
BiologicalMaterial08	0.1473037
BiologicalMaterial09	0.1555370
BiologicalMaterial10	0.0724908
BiologicalMaterial11	0.1629170
BiologicalMaterial12	0.2412185
ManufacturingProcess01	0.1578287
ManufacturingProcess02	0.1331255
ManufacturingProcess03	0.0129271
ManufacturingProcess04	0.1098940
ManufacturingProcess05	0.0161460
ManufacturingProcess06	0.2402510
ManufacturingProcess07	0.0007811
ManufacturingProcess08	0.0000261
ManufacturingProcess09	0.2587624
ManufacturingProcess10	0.0708371
ManufacturingProcess11	0.2043809
ManufacturingProcess12	0.1571690
ManufacturingProcess13	0.2747904
ManufacturingProcess14	0.0641521
ManufacturingProcess15	0.1395217
ManufacturingProcess16	0.0643132
ManufacturingProcess17	0.3063856

	Overall
ManufacturingProcess18	0.1083749
ManufacturingProcess19	0.0718435
ManufacturingProcess20	0.0401452
ManufacturingProcess21	0.0590100
ManufacturingProcess22	0.0028819
ManufacturingProcess23	0.0132709
ManufacturingProcess24	0.0670870
ManufacturingProcess25	0.1663295
ManufacturingProcess26	0.1076723
ManufacturingProcess27	0.0729450
ManufacturingProcess28	0.0776908
ManufacturingProcess29	0.1889376
ManufacturingProcess30	0.1247785
ManufacturingProcess31	0.2174880
ManufacturingProcess32	0.3734379
ManufacturingProcess33	0.1850785
ManufacturingProcess34	0.0183944
ManufacturingProcess35	0.0719678
ManufacturingProcess36	0.2131597
ManufacturingProcess37	0.0537773
ManufacturingProcess38	0.0032073
ManufacturingProcess39	0.0018747
ManufacturingProcess40	0.0008630
ManufacturingProcess41	0.0003102
ManufacturingProcess42	0.0001593
ManufacturingProcess43	0.0288916
ManufacturingProcess44	0.0056120
ManufacturingProcess45	0.0016508

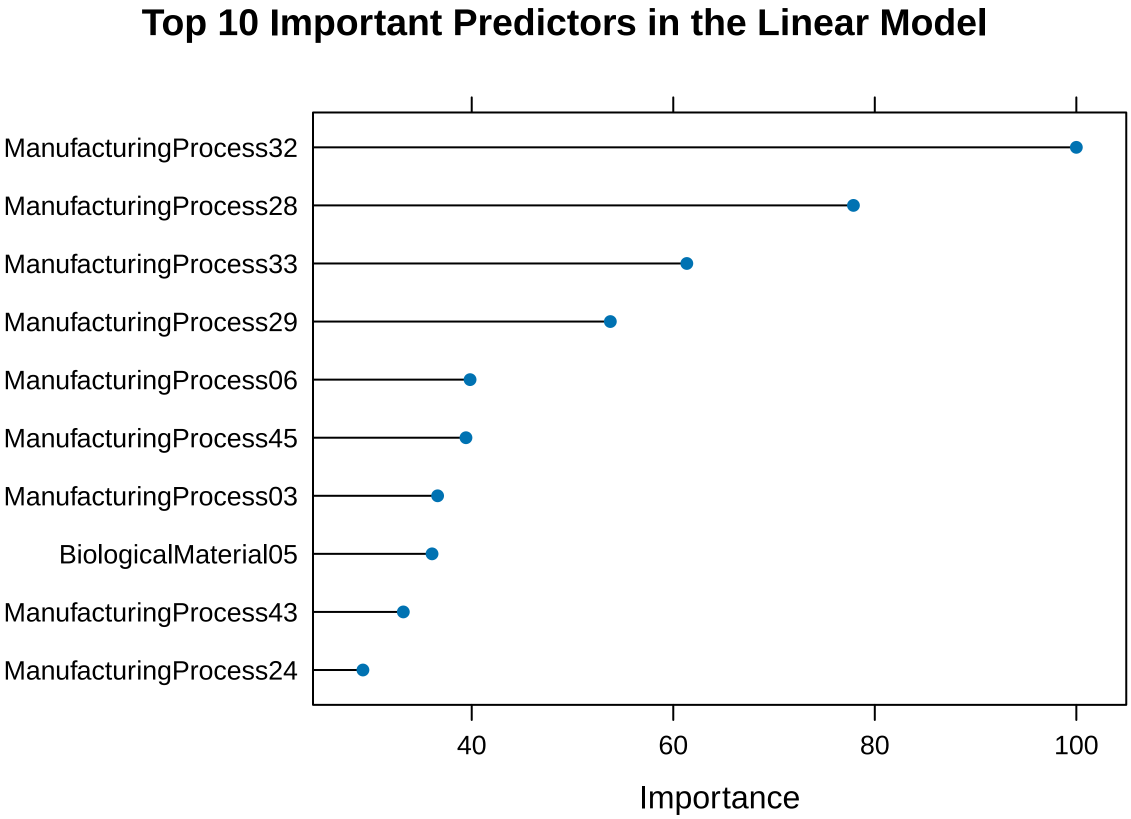
Table 17: Neuro Network Variable Importance

	Overall
BiologicalMaterial01	0.7723617
BiologicalMaterial02	0.2804983
BiologicalMaterial03	0.7901909
BiologicalMaterial04	1.7277147
BiologicalMaterial05	3.2560049
BiologicalMaterial06	0.3108047
BiologicalMaterial07	0.2706206
BiologicalMaterial08	1.0305662
BiologicalMaterial09	1.4867648
BiologicalMaterial10	2.1312740
BiologicalMaterial11	1.9452196
BiologicalMaterial12	1.1860290
ManufacturingProcess01	1.8214109
ManufacturingProcess02	0.8447873
ManufacturingProcess03	2.3812633
ManufacturingProcess04	3.1753533
ManufacturingProcess05	2.1955969
ManufacturingProcess06	2.8088904

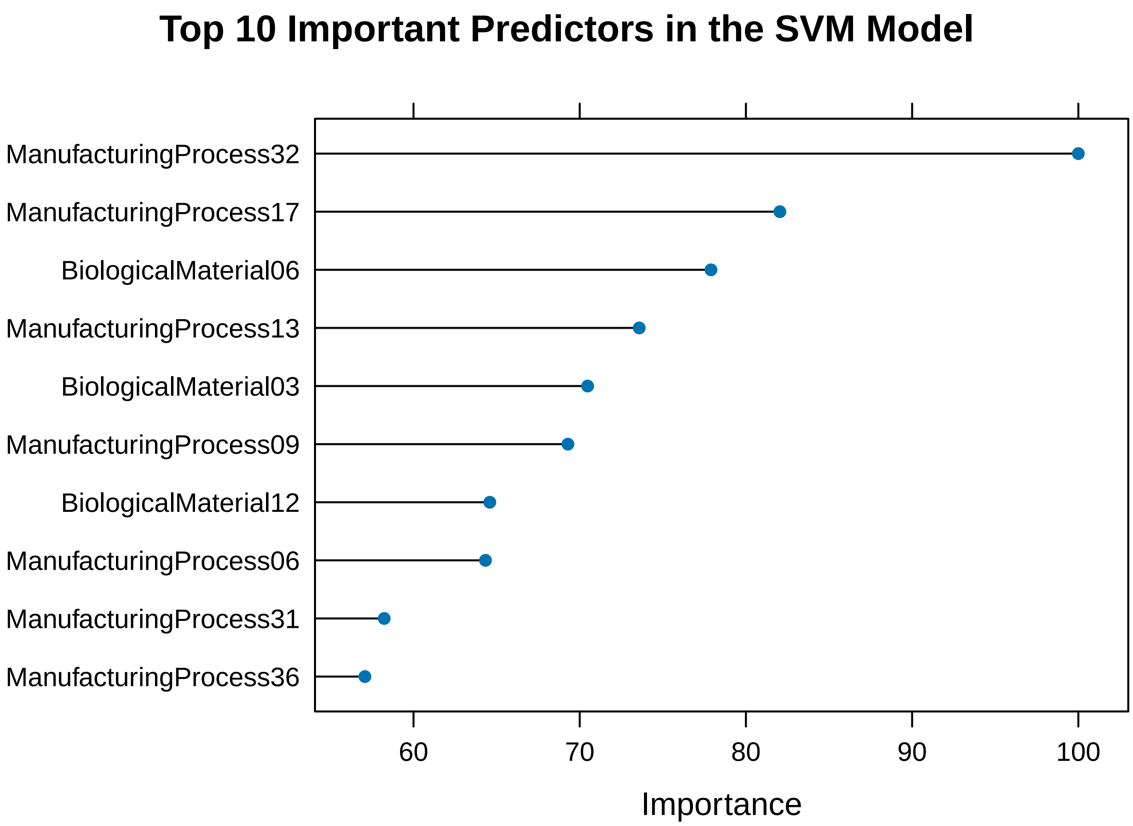
	Overall
ManufacturingProcess07	1.3537290
ManufacturingProcess08	1.0890765
ManufacturingProcess09	3.2152691
ManufacturingProcess10	1.6766743
ManufacturingProcess11	1.5229254
ManufacturingProcess12	2.3792366
ManufacturingProcess13	3.3285803
ManufacturingProcess14	1.0474859
ManufacturingProcess15	0.9919241
ManufacturingProcess16	0.6012087
ManufacturingProcess17	2.4083483
ManufacturingProcess18	0.7388137
ManufacturingProcess19	2.6871357
ManufacturingProcess20	0.7940063
ManufacturingProcess21	0.8359628
ManufacturingProcess22	1.7166856
ManufacturingProcess23	1.9142858
ManufacturingProcess24	3.7776712
ManufacturingProcess25	0.1586315
ManufacturingProcess26	0.4912437
ManufacturingProcess27	0.3799888
ManufacturingProcess28	3.3811218
ManufacturingProcess29	2.5123412
ManufacturingProcess30	0.6572429
ManufacturingProcess31	1.0279280
ManufacturingProcess32	9.4391202
ManufacturingProcess33	1.7215565
ManufacturingProcess34	1.0269644
ManufacturingProcess35	0.6682380
ManufacturingProcess36	2.3534155
ManufacturingProcess37	3.4206330
ManufacturingProcess38	0.8200280
ManufacturingProcess39	3.8372936
ManufacturingProcess40	1.0862780
ManufacturingProcess41	1.2860056
ManufacturingProcess42	0.6907966
ManufacturingProcess43	1.2595291
ManufacturingProcess44	1.8477924
ManufacturingProcess45	1.4394803

	Overall
ManufacturingProcess32	100.00000
ManufacturingProcess28	77.88060
ManufacturingProcess33	61.34496
ManufacturingProcess29	53.75483
ManufacturingProcess06	39.83626
ManufacturingProcess45	39.42922
ManufacturingProcess03	36.61428
BiologicalMaterial05	36.06137
ManufacturingProcess43	33.21052

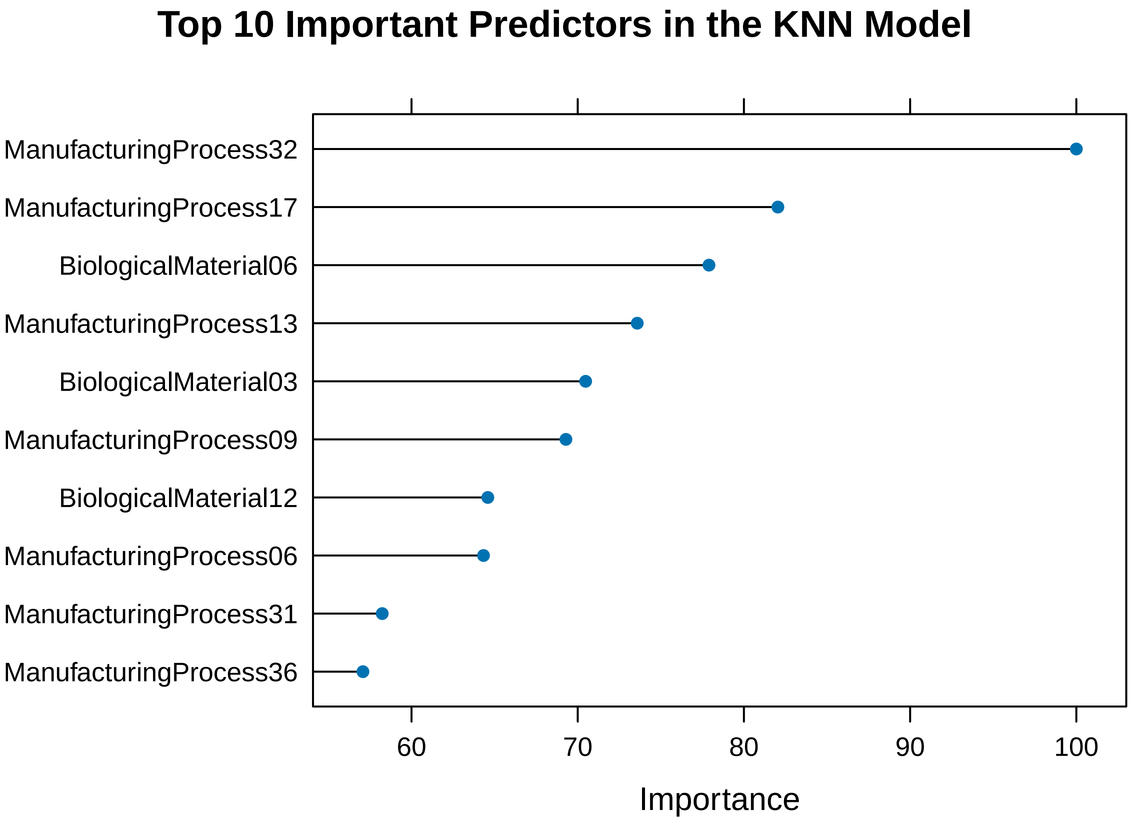
	Overall
ManufacturingProcess24	29.20374



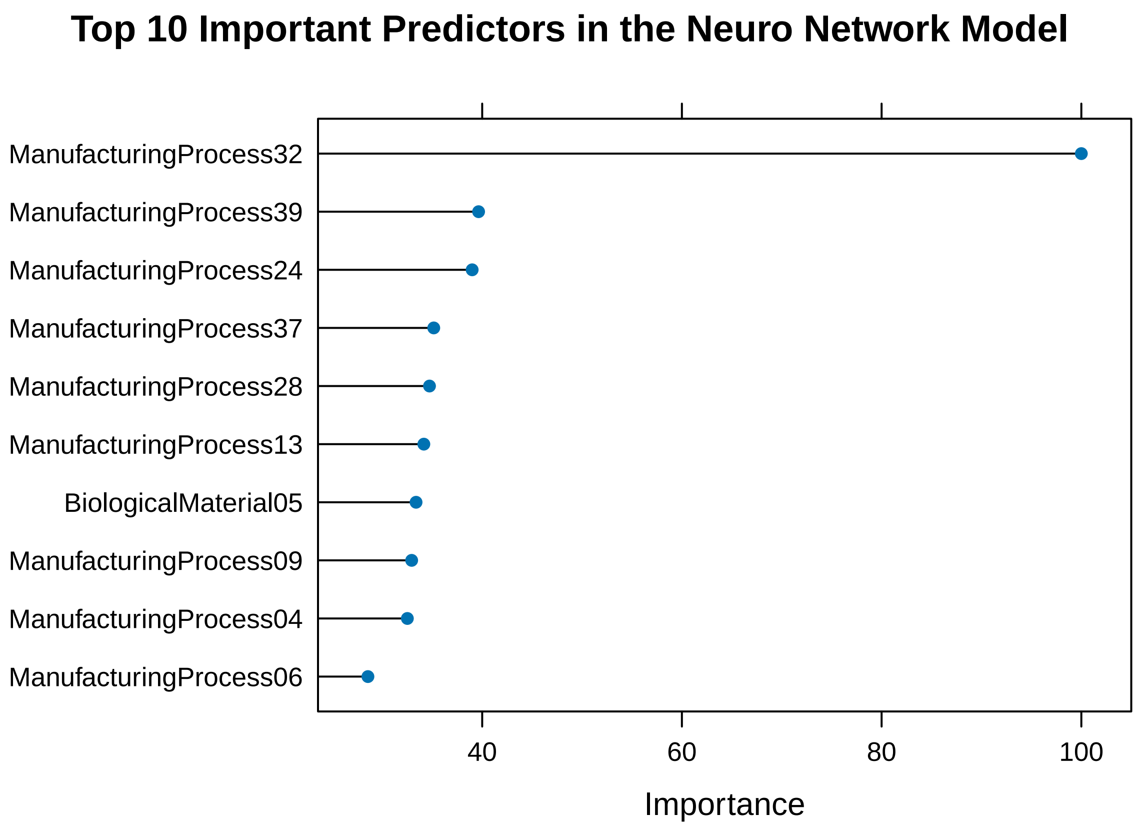
	Overall
ManufacturingProcess32	100.00000
ManufacturingProcess17	82.04335
BiologicalMaterial06	77.89778
ManufacturingProcess13	73.58212
BiologicalMaterial03	70.47676
ManufacturingProcess09	69.28981
BiologicalMaterial12	64.59155
ManufacturingProcess06	64.33243
ManufacturingProcess31	58.23648
ManufacturingProcess36	57.07735


Top 10 Important Predictors in the SVM Model

	Overall
ManufacturingProcess32	100.00000
ManufacturingProcess17	82.04335
BiologicalMaterial06	77.89778
ManufacturingProcess13	73.58212
BiologicalMaterial03	70.47676
ManufacturingProcess09	69.28981
BiologicalMaterial12	64.59155
ManufacturingProcess06	64.33243
ManufacturingProcess31	58.23648
ManufacturingProcess36	57.07735



	Overall
ManufacturingProcess32	100.00000
ManufacturingProcess39	39.63867
ManufacturingProcess24	38.99622
ManufacturingProcess37	35.14903
ManufacturingProcess28	34.72328
ManufacturingProcess13	34.15713
BiologicalMaterial05	33.37511
ManufacturingProcess09	32.93617
ManufacturingProcess04	32.50607
ManufacturingProcess06	28.55732



To compare the top 10 important predictors across the optimal Cubist, Linear, SVM, KNN, and Neural Network models, we can focus on identifying patterns and any overlapping predictors. Here are the key observations:

Comparison and Analysis of Top 10 Important Predictors:

Common Predictors Across Models:

ManufacturingProcess32 appears as the top predictor in almost all models (Cubist, Linear, SVM, KNN, and Neural Network), indicating it has strong predictive power across both linear and nonlinear models.

ManufacturingProcess06 is also a common predictor across all models, indicating its consistent importance.

ManufacturingProcess09 and ManufacturingProcess13 appear frequently, showing up in most models.

Biological Variables Across Models:

In the Cubist model, BiologicalMaterial12 and BiologicalMaterial03 are important.

In the SVM and KNN models, BiologicalMaterial06 and BiologicalMaterial03 are among the top predictors.

The Neural Network model includes BiologicalMaterial05 in its top predictors.

Linear Model has BiologicalMaterial05 in its top 10 as well.

This suggests that while biological variables are important, the specific biological predictor differs slightly depending on the model.

Process Variables Dominance:

Across all models, process-related variables dominate the list of top predictors, indicating that manufacturing processes are generally more influential in predicting the target variable.

Models like the Linear Model and Neural Network tend to favor more unique process-related variables such as ManufacturingProcess28, ManufacturingProcess33, ManufacturingProcess24, and ManufacturingProcess45.

Insights and Interpretation of Variable Importance:

Consistency Across Models:

Some predictors, like ManufacturingProcess32 and ManufacturingProcess06, are consistently important across all models. This consistency suggests these predictors have a strong and likely linear relationship with the target variable, making them valuable regardless of model complexity.

Variability with Model Type:

The Cubist and SVM models tend to include a mix of both manufacturing process variables and biological material variables, indicating that these models might capture more nuanced interactions between biological and process-related factors.

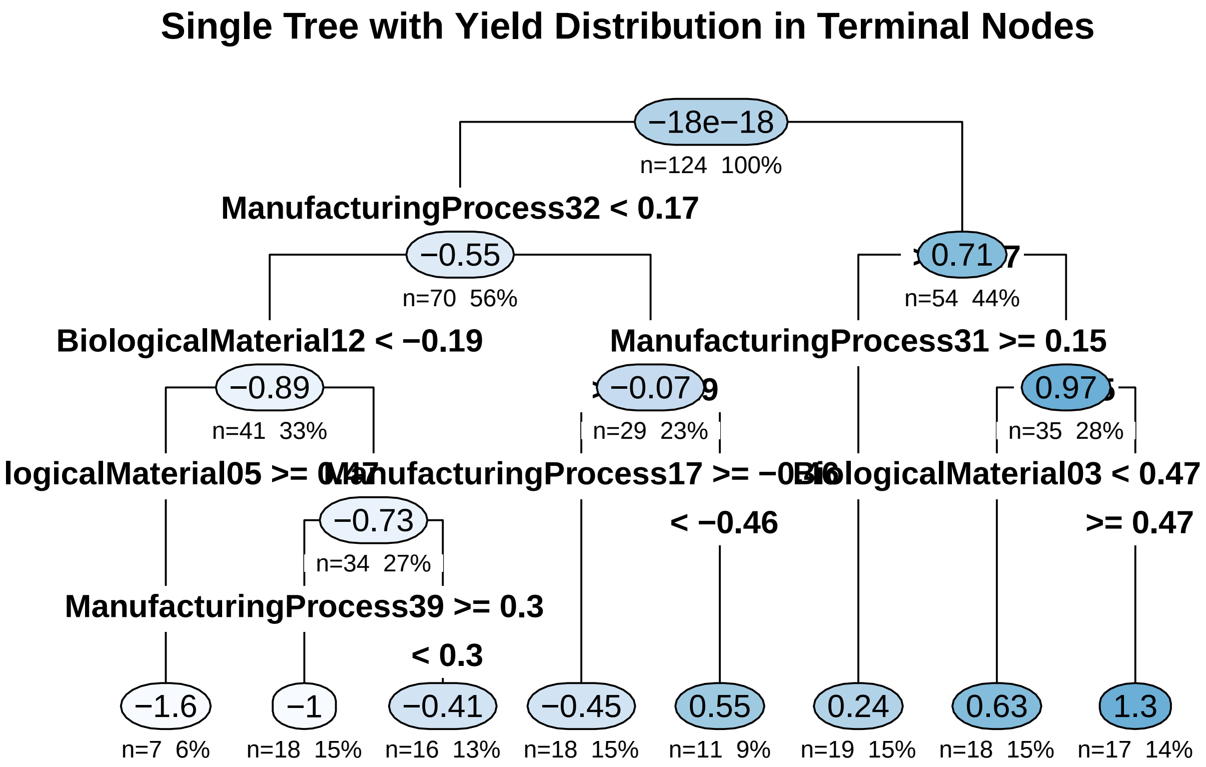
The Neural Network model, known for capturing complex patterns, still ranks process-related variables highly, but includes some biological materials that may interact nonlinearly with the process variables.

Model-Specific Important Variables:

Certain variables are more model-specific, such as ManufacturingProcess45 in the Linear model or ManufacturingProcess04 in the Neural Network model. These variables might have non-linear or interaction effects that are better captured by specific model structures.

ManufacturingProcess32 and ManufacturingProcess06 emerge as the most critical predictors across different model types, indicating their fundamental importance to the target variable. Process variables generally dominate the top predictor list, but the inclusion of biological variables varies by model, with some models (e.g., SVM and KNN) including multiple biological materials. This analysis suggests that while process variables are consistently impactful, certain biological factors play a secondary role, especially in nonlinear models like SVM and KNN.

(c) Plot the optimal single tree with the distribution of yield in the terminal nodes.



Single Tree with Yield Distribution in Terminal Nodes
−18e−18
n=124 100%
ManufacturingProcess32 < 0.17
−0.55
n=70 56%
0.71
n=54 44%
BiologicalMaterial12 < −0.19
ManufacturingProcess31 >= 0.15
−0.89
n=41 33%
−0.07
n=29 23%
0.97
n=35 28%
logicalMaterial05 >= 0.47
ManufacturingProcess17 >= −0.46
BiologicalMaterial03 < 0.47
< −0.46
>= 0.47
−0.73
n=34 27%
ManufacturingProcess39 >= 0.3
< 0.3
−1.6
n=7 6%
−1
n=18 15%
−0.41
n=16 13%
−0.45
n=18 15%
0.55
n=11 9%
0.24
n=19 15%
0.63
n=18 15%
1.3
n=17 14%

Does this view of the data provide additional knowledge about the biological or process predictors and their relationship with yield?

Yes, this decision tree provides additional insights into the relationship between the biological and process predictors with yield. Here's a breakdown of the knowledge we can gain from this tree visualization:

Hierarchical Structure and Importance

ManufacturingProcess32 is the root node, which implies it is the most critical variable in determining yield. The first split based on ManufacturingProcess32 indicates that variations in this predictor have a substantial impact on yield, setting the primary direction for the rest of the splits.

The next most important predictors are BiologicalMaterial12 and ManufacturingProcess31, as they are immediately involved in the subsequent splits. This hierarchical ordering implies that ManufacturingProcess32, ManufacturingProcess31, and BiologicalMaterial12 are likely the most influential in determining yield outcomes.

Relationship Patterns in Predictors For nodes where ManufacturingProcess32 is below 0.17, the model further segments based on BiologicalMaterial12. This suggests that BiologicalMaterial12 plays a significant role in explaining yield variability when ManufacturingProcess32 is low.

For nodes where ManufacturingProcess32 is above 0.17, ManufacturingProcess31 and BiologicalMaterial03 play a crucial role, with specific thresholds of 0.15 and 0.47, respectively. This indicates that BiologicalMaterial03 becomes more important in determining yield under certain conditions of ManufacturingProcess31.

Interaction Effects

The tree reveals potential interaction effects between biological and process variables. For instance:

When BiologicalMaterial12 is less than 0.19 and ManufacturingProcess32 is low, the tree further segments based on BiologicalMaterial05 and ManufacturingProcess39. This suggests an interaction where certain levels of BiologicalMaterial12 in conjunction with ManufacturingProcess32 influence yield.

A similar interaction is seen with ManufacturingProcess31 and BiologicalMaterial03 on the right side of the tree, where the yield outcomes vary based on both the values of ManufacturingProcess31 and BiologicalMaterial03.

Yield Predictions at Terminal Nodes

Each terminal node represents a different yield outcome, with values that range from negative (e.g., -1.6) to positive (e.g., 1.3).

Certain combinations of biological and process values lead to higher yield predictions. For example:

High values in ManufacturingProcess31 and BiologicalMaterial03 (>= 0.47) result in a yield of 1.3. Conversely, low values in ManufacturingProcess32 and BiologicalMaterial12 result in lower yield outcomes, indicating a negative impact on yield when these conditions are met.

Threshold Effects

The decision tree model identifies specific threshold values (e.g., ManufacturingProcess32 $<$ 0.17, BiologicalMaterial12 $<$ 0.19) where the yield distribution changes. This implies that small changes around these thresholds could lead to significant changes in yield, providing actionable insights for optimizing manufacturing and biological parameters.

Interpretation of Biological vs. Process Variables

Process Variables: ManufacturingProcess32 and ManufacturingProcess31 have the largest impact, suggesting that optimizing these process parameters could lead to better yield.

Biological Variables: Although BiologicalMaterial12 and BiologicalMaterial03 are less dominant than the main process variables, they still play a significant role, especially under specific conditions of the process variables. This indicates that the biological variables act as secondary factors, influencing yield in the presence of certain process conditions.

This decision tree offers a more nuanced understanding of how both biological and process variables interact to influence yield. The hierarchical splits and interactions reveal that:

Process variables like ManufacturingProcess32 and ManufacturingProcess31 are primary drivers. Biological variables have an additive effect, refining the yield prediction under certain process conditions.

Thresholds identified in this tree provide actionable points that could be used for process optimization, helping to achieve a higher yield by controlling specific variables.

This visualization is valuable for understanding not just the importance of each variable but also how combinations of biological and process factors jointly impact yield outcomes.