Workflow: Predicting Algae Blooms

PROBLEM DESCRIPTION

The ability to monitor and perform early forecasts of various river algae blooms is crucial to control the ecological harm they can cause. The dataset which is used to train the learning model consists of:

- chemical properties of various water samples of European rivers
- the quantity of seven algae in each of the samples, and
- the characteristics of the collection process for each sample.

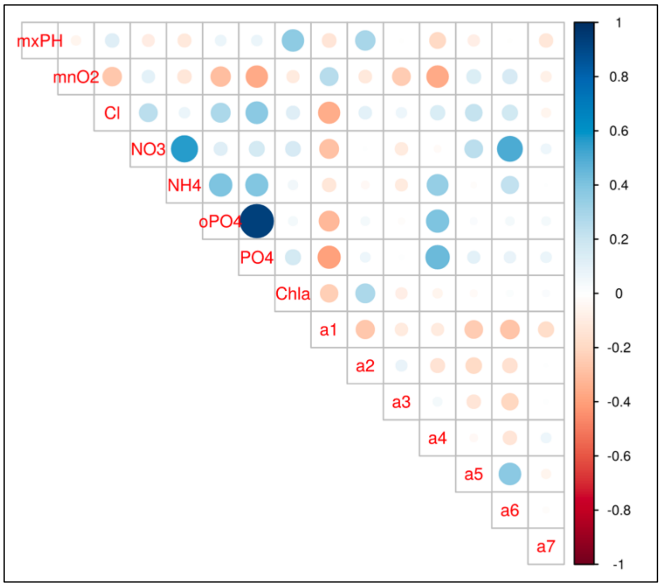
What is the data science motivation for such a model? After all, we can analyze water samples to determine if various harmful algae are present or absent. Chemical monitoring is cheap and easy to automate, whereas biological analysis of samples is expensive and slow. Another answer is that analyzing the samples for harmful content does not provide a better understanding of what drives the production of algae: it just tells us which samples contain algae.

The algae blooms dataset has 338 observations of 18 variables each: *season*, *size*, *speed*, *mxPH*, *mnO2*, *Cl*, *NO3*, *NH4*, *oPO4*, *PO4*, *Chla*, *a1*, *a2*, *a3*, *a4*, *a5*, *a6*, *a7*.

- 3 of the fields are categorical (*season*, *size*, *speed*, which refer to the data collection process)
- of the numerical fields, 8 have names that sound vaguely "chemical"
- the remaining fields refer to various algae blooms

We can get a better feel for the data frame by observing it as an array (first 6 rows):

season	size	speed	mxPH	mnO2	Cl	NO3	NH4	oPO4	PO4	Chla	a1	a2	a3	a4	a5	a6	a7
winter	small	medium	8.00	9.8	60.800	6.238	578.000	105.000	170.000	50.0	0.0	0.0	0.0	0.0	34.2	8.3	0.0
spring	small	medium	8.35	8.0	57.750	1.288	370.000	428.750	558.750	1.3	1.4	7.6	4.8	1.9	6.7	0.0	2.1
autumn	small	medium	8.10	11.4	40.020	5.330	346.667	125.667	187.057	15.6	3.3	53.6	1.9	0.0	0.0	0.0	9.7
spring	small	medium	8.07	4.8	77.364	2.302	98.182	61.182	138.700	1.4	3.1	41.0	18.9	0.0	1.4	0.0	1.4
autumn	small	medium	8.06	9.0	55.350	10.416	233.700	58.222	97.580	10.5	9.2	2.9	7.5	0.0	7.5	4.1	1.0
winter	small	high	8.25	13.1	65.750	9.248	430.000	18.250	56.667	28.4	15.1	14.6	1.4	0.0	22.5	12.6	2.9



A portrait of the relationships between the variables is provided by the correlogram on the left (for the numerical variables).

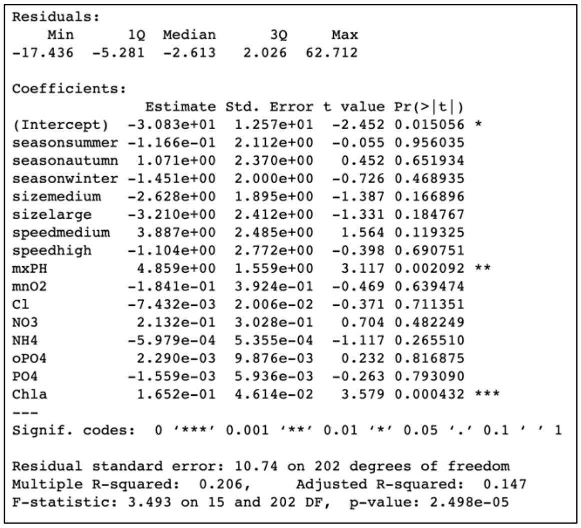
For now, we assume that the dataset has been properly explored and understood, and that any problems related to invalid data (outliers, etc.) have been solved.

PREDICTION MODELS

Our goal is to build a predictive model for the various algae blooms *a1* – *a7*. It is a supervised learning tasks; in order to mitigate overfitting (a consequence of the bias-variance trade-off), we set aside a test set on which the models (which will be learned on the training set) are evaluated. We use a 65%-35% split (218 – 120 randomly selected training/test observations).

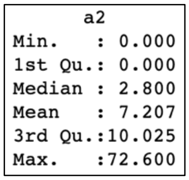
GENERALIZED LINEAR MODEL

As a baseline model, we run a linear model to predict *a2*, for example, against all the predictor variables, but using only the training set as data. The results are summarized below.



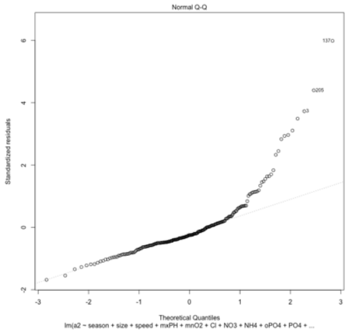
```
Residuals:
    Min      1Q  Median      3Q     Max
-17.436  -5.281  -2.613   2.026  62.712

Coefficients:
                Estimate Std. Error t value Pr(>|t|)
(Intercept)   -3.083e+01  1.257e+01  -2.452 0.015056 *
seasonsummer  -1.166e-01  2.112e+00  -0.055 0.956035
seasonautumn   1.071e+00  2.370e+00   0.452 0.651934
seasonwinter  -1.451e+00  2.000e+00  -0.726 0.468935
sizemedium    -2.628e+00  1.895e+00  -1.387 0.166896
sizelarge     -3.210e+00  2.412e+00  -1.331 0.184767
speedmedium    3.887e+00  2.485e+00   1.564 0.119325
speedhigh     -1.104e+00  2.772e+00  -0.398 0.690751
mxPH           4.859e+00  1.559e+00   3.117 0.002092 **
mnO2          -1.841e-01  3.924e-01  -0.469 0.639474
Cl            -7.432e-03  2.006e-02  -0.371 0.711351
NO3            2.132e-01  3.028e-01   0.704 0.482249
NH4           -5.979e-04  5.355e-04  -1.117 0.265510
oPO4           2.290e-03  9.876e-03   0.232 0.816875
PO4           -1.559e-03  5.936e-03  -0.263 0.793090
Chla           1.652e-01  4.614e-02   3.579 0.000432 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 10.74 on 202 degrees of freedom
Multiple R-squared:  0.206,     Adjusted R-squared:  0.147
F-statistic: 3.493 on 15 and 202 DF,  p-value: 2.498e-05
```

We see that the adjusted R² coefficient is fairly small. Furthermore, if the linear model is a good fit, the residuals should have a mean of zero and be "small", which doesn’t seem to be the case (at least, relative to the range of *a2*, see 6-pt summary to the right).



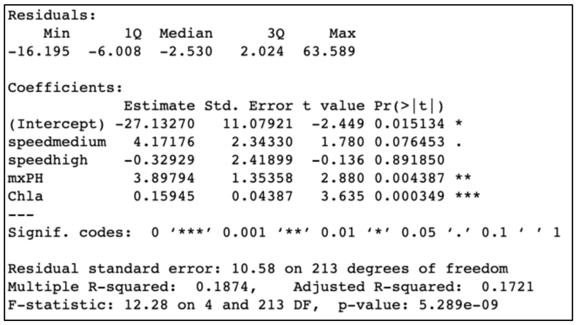
```
       a2
Min.   : 0.000
1st Qu.: 0.000
Median : 2.800
Mean   : 7.207
3rd Qu.:10.025
Max.   :72.600
```

The normal QQ-plot for the residuals (see figure on the right), in particular, seem to indicate that linearity of the data is probably not met, as an assumption.



On the other hand, the F−statistic seems to indicate some (linear) dependence on the predictor variables.

Backward elimination stepwise selection suggests that the best linear model for *a2* involves *speed*, *mxPH*, and *Chla*.

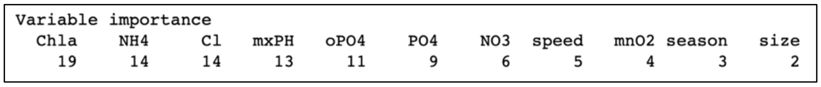


```
Residuals:
    Min      1Q  Median      3Q     Max
-16.195  -6.008  -2.530   2.024  63.589

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) -27.13270   11.07921  -2.449 0.015134 *
speedmedium   4.17176    2.34330   1.780 0.076453 .
speedhigh    -0.32929    2.41899  -0.136 0.891850
mxPH          3.89794    1.35358   2.880 0.004387 **
Chla          0.15945    0.04387   3.635 0.000349 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 10.58 on 213 degrees of freedom
Multiple R-squared:  0.1874,    Adjusted R-squared:  0.1721
F-statistic: 12.28 on 4 and 213 DF,  p-value: 5.289e-09
```

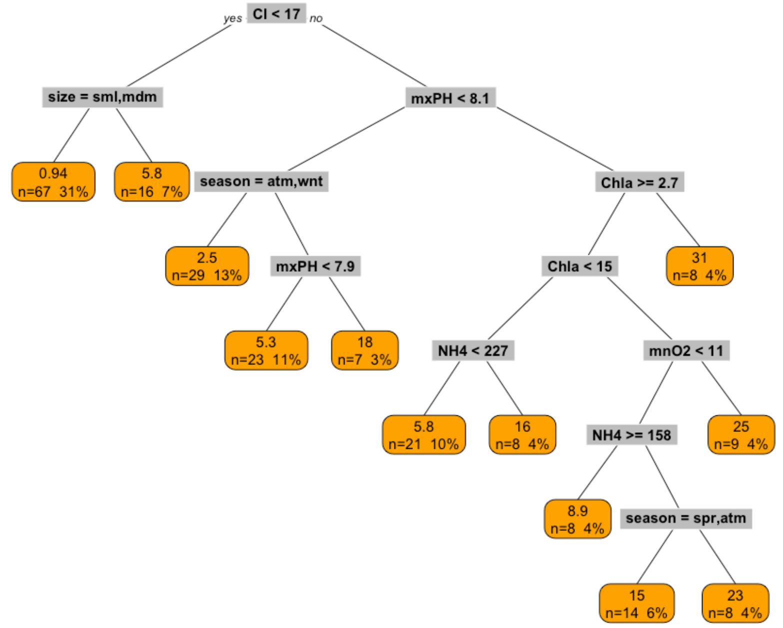
The fit is still not ideal (the value of the adjusted R² is quite small).

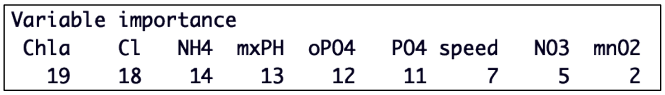
REGRESSION TREE MODEL

An alternative to regression is the use of regression trees. A recursive partition tree for *a2* is shown below, as is a pruned tree, with the relative importance of the variables for both models:

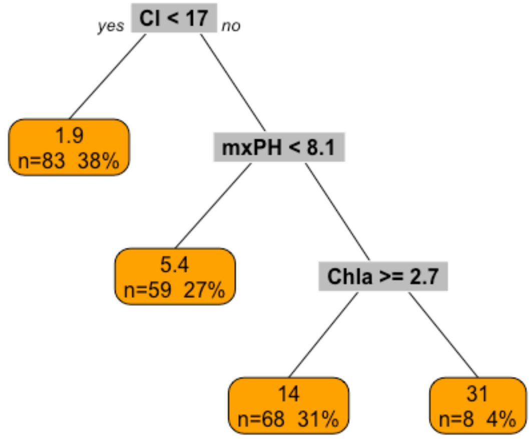


```
Variable importance
  Chla     NH4      Cl    mxPH    oPO4      PO4     NO3   speed    mnO2  season     size
    19      14      14      13      11        9       6       5       4       3        2
```





```
Variable importance
 Chla    Cl   NH4  mxPH  oPO4   PO4 speed   NO3  mnO2
   19    18    14    13    12    11     7     5     2
```

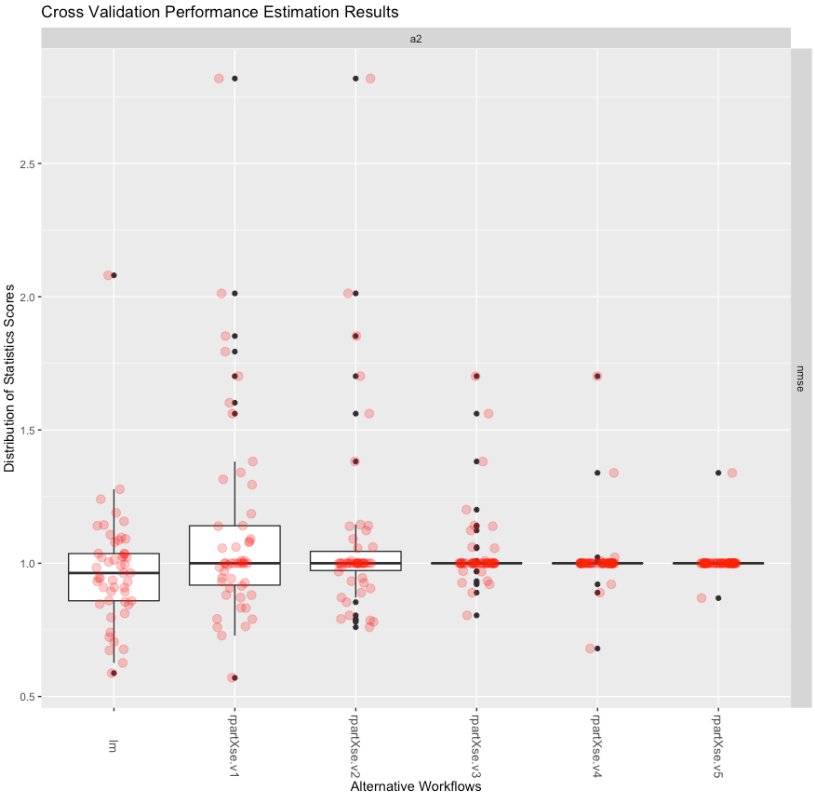


MODEL EVALUATION

At this stage, we know that the linear model is not great for *a2*, and we have grown regression trees for *a2* but we have not yet discussed whether these models are good fits for *a2*, to say nothing of the remaining 6 algae concentrations.

Various metrics can be used to determine how the predicted values on the test set compare to the actual values: we will use the normalized mean squared error (NMSE). NMSE is unitless: values between 0 and 1 indicate that the model performs better than the baseline; values greater than 1 indicate that the model's performance is sub-par.

The test NMSE for the linear model and for a family of regression tree models (one for 5 different values of a growth/pruning parameter) is estimated using 5 repetitions of 10-fold cross-validation. For each model, the results for the 50 cross-validated models are shown in the image to the right. Summaries for the 50 models for each approach are found below.

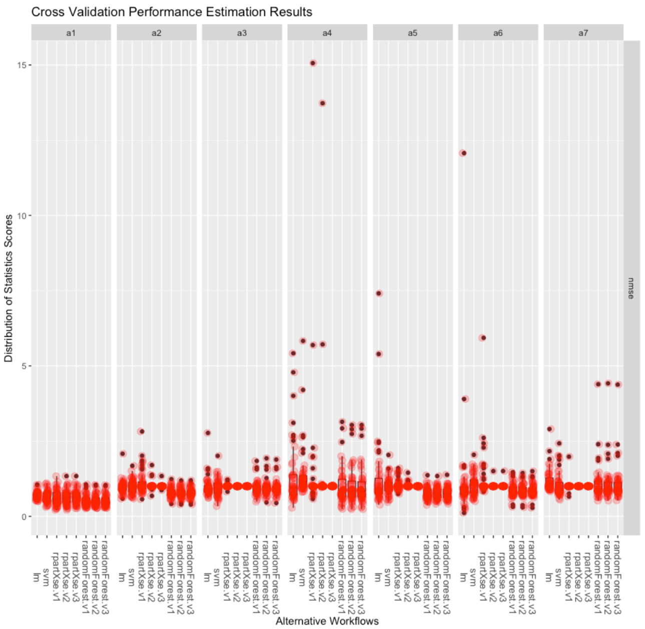


```
lm             nmse | rpartXse.v1   nmse | rpartXse.v2   nmse
avg       0.9880781 | avg      1.0333720 | avg      1.0596868
std       0.3682616 | std      0.3406970 | std      0.3147441
med       0.9470239 | med      1.0000000 | med      1.0000000
iqr       0.2817843 | iqr      0.1842643 | iqr      0.0435684
min       0.4869917 | min      0.6171205 | min      0.5049684
max       2.5236216 | max      2.4535376 | max      2.4535376
rpartXse.v3    nmse | rpartXse.v4   nmse | rpartXse.v5   nmse
avg        1.028517 | avg       1.012748 | avg       1.001631
std        0.230181 | std       0.078035 | std       0.011533
med        1.000000 | med       1.000000 | med       1.000000
iqr        0.000000 | iqr       0.000000 | iqr       0.000000
min        0.528342 | min       0.819828 | min       1.000000
max        2.365684 | max       1.413850 | max       1.081548
```

It's not necessarily clear which of the models has smaller values of NMSE overall, although it does seem that the latter versions of the regression tree models are not substantially better than the baseline model. The first regression tree model sometimes produces very small NMSE values, but that's offset by some of the larger values it also produces (similarly for the linear model). At any rate, visual evidence seems to suggest that the linear model is the best predictive model for a2 given the training data.

This might seem disheartening at first given how poorly the linear model performed, but it is helpful to remember that there is no guarantee that a decent predictive model even exists. Furthermore, regression trees and linear models are only two of a whole collection of possible models. How do support vector regression or random forests models perform, for instance?

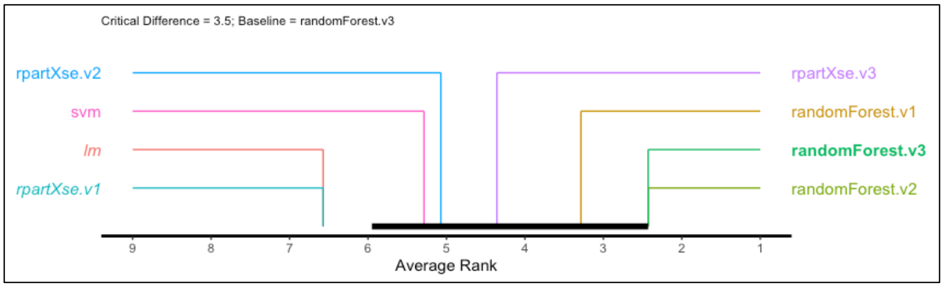
We repeat the task of estimating test NMSE *via* 5 replicates of 10-fold cross-validation for 8 models (linear regression, support vector regression, 3 regression trees, 3 random forests) for all target variables (*a1* – *a7*) simultaneously. We are not looking for a single model which will optimize all learning tasks at once, but rather that we can prepare and evaluate the models for each target variable with the same bit of code. The results are shown in the figure to the right. The top performers (average value of NMSE) for each response are shown on the next page.



```
Rank.a1      model   est.nmse | Rank.a2      model   est.nmse | Rank.a3      model   est.nmse | Rank.a4      model est.nmse
1 randomForest.v2 0.5217204   | 1 randomForest.v3 0.7798749   | 1 randomForest.v3 0.9377108   | 1      rpartXse.v3 1.001453
2 randomForest.v3 0.5228744   | 2 randomForest.v2 0.7806831   | 2 randomForest.v2 0.9400108   | 2 randomForest.v3 1.006496
3 randomForest.v1 0.5264328   | 3 randomForest.v1 0.7849360   | 3 randomForest.v1 0.9431801   | 3 randomForest.v1 1.006806
Rank.a5      model   est.nmse | Rank.a6      model   est.nmse | Rank.a7      model   est.nmse
1 randomForest.v1 0.7626241   | 1 randomForest.v2 0.8590227   | 1       rpartXse.v2   1.00000
2 randomForest.v2 0.7675794   | 2 randomForest.v3 0.8621478   | 2       rpartXse.v3   1.00000
3 randomForest.v3 0.7681834   | 3 randomForest.v1 0.8663869   | 3       rpartXse.v1   1.00797
```

At first glance, the 3rd random forest model (the one that build predictions on 700 trees, as opposed to 200 and 500 for the other random forests models) seems to perform best, but these rankings do not report on the standard error, and so we cannot tell whether the differences between the estimated test NMSEs are statistically significant on the basis of the estimates alone.

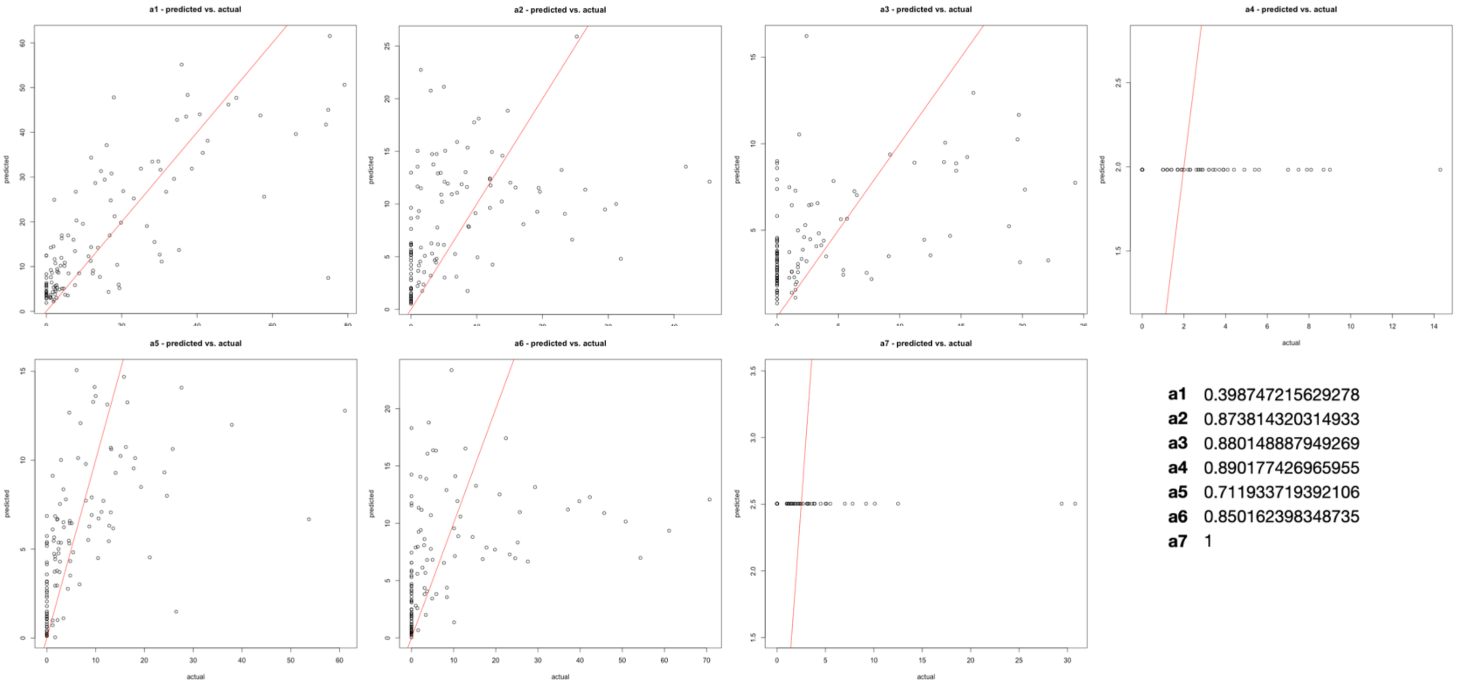
Using the 3rd random forest model as a baseline, we compute the rank differences to the other 7 models for all target variables. The critical rank difference is 3.52. On average, the rank difference to the other models is shown in the list on the right. We can reject with 95% certainty that the performance of the baseline method is the same as that of the linear model and the first regression tree model (`rpartXse.v2`), but not that it is better than the other 5 models. The information is also displayed in the Bonferroni-Dunn CD diagram below.



lm	4.57
svm	2.14
rpartXse.v1	4.28
rpartXse.v2	3.28
rpartXse.v3	2.85
randomForest.v1	1.14
randomForest.v2	0.57

MODEL PREDICTIONS

The best performer for each target response was identified from the cross-validation procedure above: for each target variable *a1* – *a7*, we run the best performer on the original training data to learn a model that is used to predict the appropriate target response for observations in the original test set. Scatterplots of predicted (y-axis) vs. actual levels (x-axis) for test observations are shown below (top: *a1* – *a4*, bottom: *a5* – *a7*), as are the true test NMSEs.



a1	0.398747215629278
a2	0.873814320314933
a3	0.880148887949269
a4	0.890177426965955
a5	0.711933719392106
a6	0.850162398348735
a7	1