DATA624_Homework7

John Ferrara

2025-04-05

Instructions

In Kuhn and Johnson do problems 6.2 and 6.3. There are only two but they consist of many parts. Please submit a link to your Rpubs and submit the .rmd file as well.

Question 6.2

Developing a model to predict permeability (see Sect. 1.4) could save significant resources for a pharmaceutical company, while at the same time more rapidly identifying molecules that have a sufficient permeability to become a drug:

(a) Start R and use these commands to load the data:

```
library(AppliedPredictiveModeling)
```

```
## Warning: package 'AppliedPredictiveModeling' was built under R version 4.4.2
```

```
data(permeability)
```

The matrix fingerprints contains the 1,107 binary molecular predictors for the 165 compounds, while permeability contains permeability response.

(b) The fingerprint predictors indicate the presence or absence of substructures of a molecule and are often sparse meaning that relatively few of the molecules contain each substructure. Filter out the predictors that have low frequencies using the nearZeroVar function from the caret package. How many predictors are left for modeling?

```
fin_df <- data.frame(fingerprints)
print(nrow(fin_df))#165 rows
```

```
## [1] 165
```

```
print(nrow(t(fin_df)))#1107 predictors
```

```
## [1] 1107
```

```
## Limiting to those predictors that have variance.
#Getting those with little variance.
no_var <- nearZeroVar(fin_df)
filtered_fingerprints<- fin_df[,-no_var]

#print(head(filtered_fingerprints))
print(nrow(filtered_fingerprints))#165
```

```
## [1] 165
```

```
print(nrow(t(filtered_fingerprints))) #388
```

```
## [1] 388
```

```
### There are a total of 388 predictors / columns left for the analysis.
```

(c) Split the data into a training and a test set, pre-process the data, and tune a PLS model. How many latent variables are optimal and what is the corresponding resampled estimate of R2?

```
## Joining the permeability vector to the main df before splitting the training data.

#Confirming it is 165 rows long before adding as columns
print(length(permeability)) #165
```

```
## [1] 165
```

```
fin_df <- cbind(filtered_fingerprints, permeability)

## Splitting into training and test. 70 / 30 split
training_split <- createDataPartition(fin_df$permeability, p = 0.7, list = FALSE)
training_data <- fin_df[training_split,]
print(nrow(training_data)) #117
```

```
## [1] 117
```

```
print(nrow(t(training_data))) #389
```

```
## [1] 389
```

```
test_data <- fin_df[-training_split,]
print(nrow(test_data)) #48
```

```
## [1] 48
```

```
print(nrow(t(test_data))) #389
```

```
## [1] 389
```

```
#Data is joined and split into training and test groups.
```

```
set.seed(55)

## kfold cross validation
cros_val <- trainControl(method = "cv", number = 10)

## Now building the pls model with this. Using the pre processing in chapter to center and sc
ale.
pls_model <- train(permeability ~ ., data = training_data, method = "pls", preProc = c("cente
r", "scale"),   tuneLength = 20, trControl = cros_val)

print(pls_model)
```

```
## Partial Least Squares
##
## 117 samples
## 388 predictors
##
## Pre-processing: centered (388), scaled (388)
## Resampling: Cross-Validated (10 fold)
## Summary of sample sizes: 105, 105, 105, 106, 105, 105, ...
## Resampling results across tuning parameters:
##
##   ncomp  RMSE      Rsquared   MAE
##    1     12.58853  0.3511641   9.356453
##    2     11.60144  0.4544098   8.366885
##    3     12.05765  0.4393652   9.130589
##    4     12.27880  0.4394690   9.369868
##    5     12.27715  0.4449749   9.423895
##    6     11.79750  0.4773522   9.154883
##    7     12.15269  0.4673002   9.696704
##    8     12.34894  0.4608957   9.803761
##    9     12.92097  0.4286110  10.327925
##   10     13.23454  0.4124511  10.606670
##   11     13.32668  0.4060148  10.494620
##   12     13.71110  0.3976863  10.548275
##   13     13.56240  0.4142326  10.442903
##   14     13.74217  0.4038169  10.603450
##   15     13.62405  0.4053626  10.456459
##   16     13.43965  0.4238671  10.203302
##   17     13.83938  0.4147860  10.490994
##   18     13.77051  0.4205518  10.426430
##   19     13.97746  0.4094807  10.592096
##   20     14.19270  0.4000833  10.718478
##
## RMSE was used to select the optimal model using the smallest value.
## The final value used for the model was ncomp = 2.
```

Taking a look at the model results above, the latent variables that are optimal for this model are 2. After running the model on the training data, the lowest error values (RMSE 11.7 | MAE 8.2) was 2 variables and this also had the highest r^2 value at ~0.512. In short a pls model with 2 components is ideal.

(d) Predict the response for the test set. What is the test set estimate of R2?

```
## Running it on the test data
test_preds <- predict(pls_model, newdata = test_data)

print(postResample(pred = test_preds, obs = test_data$permeability))
```

```
##       RMSE   Rsquared        MAE
## 12.1647314  0.5202086  8.3202388
```

```
      #RMSE   Rsquared        MAE
#13.6063581  0.2713528  8.5561909

## The r^2 for the test set is 0.27
```

(e) Try building other models discussed in this chapter. Do any have better predictive performance?

```
## Other models discussed in this chapter were:

# OLS regression
ols_model <- train(permeability ~ ., data = training_data, method = "lm", preProc = c("cente
r", "scale"), trControl = cros_val)
```

```
## Warning in predict.lm(modelFit, newdata): prediction from rank-deficient fit;
## attr(*, "non-estim") has doubtful cases
## Warning in predict.lm(modelFit, newdata): prediction from rank-deficient fit;
## attr(*, "non-estim") has doubtful cases
## Warning in predict.lm(modelFit, newdata): prediction from rank-deficient fit;
## attr(*, "non-estim") has doubtful cases
## Warning in predict.lm(modelFit, newdata): prediction from rank-deficient fit;
## attr(*, "non-estim") has doubtful cases
## Warning in predict.lm(modelFit, newdata): prediction from rank-deficient fit;
## attr(*, "non-estim") has doubtful cases
## Warning in predict.lm(modelFit, newdata): prediction from rank-deficient fit;
## attr(*, "non-estim") has doubtful cases
## Warning in predict.lm(modelFit, newdata): prediction from rank-deficient fit;
## attr(*, "non-estim") has doubtful cases
## Warning in predict.lm(modelFit, newdata): prediction from rank-deficient fit;
## attr(*, "non-estim") has doubtful cases
## Warning in predict.lm(modelFit, newdata): prediction from rank-deficient fit;
## attr(*, "non-estim") has doubtful cases
## Warning in predict.lm(modelFit, newdata): prediction from rank-deficient fit;
## attr(*, "non-estim") has doubtful cases
```

```
print(ols_model)
```

```
## Linear Regression
##
## 117 samples
## 388 predictors
##
## Pre-processing: centered (388), scaled (388)
## Resampling: Cross-Validated (10 fold)
## Summary of sample sizes: 105, 105, 106, 105, 105, 105, ...
## Resampling results:
##
##   RMSE     Rsquared   MAE
##   29.8877  0.2580011  21.10244
##
## Tuning parameter 'intercept' was held constant at a value of TRUE
```

```
# RMSE      Rsquared   MAE
#  26.08582  0.2149182  17.51395

test_preds_ols <- predict(ols_model, newdata = test_data)
```

```
## Warning in predict.lm(modelFit, newdata): prediction from rank-deficient fit;
## attr(*, "non-estim") has doubtful cases
```

```
print(postResample(pred = test_preds_ols, obs = test_data$permeability))
```

```
##       RMSE   Rsquared        MAE
## 28.9716851  0.1406138 19.5017028
```

```
#RMSE    Rsquared         MAE
#33.36353485  0.09437608 17.42874107


# Ridge Regression model
ridge_model <- train(permeability ~ ., data = training_data, method = "ridge", preProc = c("c
enter", "scale"),   tuneLength = 20, trControl = cros_val)
```

```
## Warning: model fit failed for Fold05: lambda=0.0000000 Error in if (zmin < gamhat) { : mis
sing value where TRUE/FALSE needed
```

```
## Warning in nominalTrainWorkflow(x = x, y = y, wts = weights, info = trainInfo,
## : There were missing values in resampled performance measures.
```

```
print(ridge_model)
```

```
## Ridge Regression
##
## 117 samples
## 388 predictors
##
## Pre-processing: centered (388), scaled (388)
## Resampling: Cross-Validated (10 fold)
## Summary of sample sizes: 105, 106, 105, 105, 105, 107, ...
## Resampling results across tuning parameters:
##
##   lambda        RMSE         Rsquared   MAE
##   0.0000000000     16.78449  0.4010052    11.886117
##   0.0001000000   7432.70364  0.1925270  4035.002624
##   0.0001467799   8649.20983  0.2113661  5392.668059
##   0.0002154435   1646.90596  0.2091461   710.633984
##   0.0003162278   2839.24160  0.1022610  1276.747770
##   0.0004641589  10681.62792  0.2667660  7243.315800
##   0.0006812921    547.95967  0.2030475   301.325185
##   0.0010000000    335.42721  0.3221792   172.942270
##   0.0014677993   2837.00493  0.2929745  1502.741819
##   0.0021544347     27.27480  0.3492623    21.111022
##   0.0031622777     56.01914  0.3208350    39.397471
##   0.0046415888     14.31350  0.3893171    10.767258
##   0.0068129207     13.81930  0.4059457    10.452938
##   0.0100000000   1276.76490  0.4230961   726.528869
##   0.0146779927     13.12967  0.4276944    10.058409
##   0.0215443469     12.90388  0.4335900     9.984634
##   0.0316227766     12.62272  0.4450382     9.748127
##   0.0464158883     12.46719  0.4525066     9.599264
##   0.0681292069     12.42141  0.4575246     9.548504
##   0.1000000000     12.34554  0.4632535     9.574879
##
## RMSE was used to select the optimal model using the smallest value.
## The final value used for the model was lambda = 0.1.
```

```
#lambda        RMSE          Rsquared   MAE
#0.1000000000      12.20011  0.4841976 9.502962e+00

test_preds_ridge <- predict(ridge_model, newdata = test_data)
print(postResample(pred = test_preds_ridge, obs = test_data$permeability))
```

```
##       RMSE   Rsquared        MAE
## 13.0256479  0.4791969  9.0590933
```

```
#      RMSE   Rsquared        MAE
#13.4624090  0.3555569  9.4449031

# Lasso Regression model
lasso_model <- train(permeability ~ ., data = training_data, method = "lasso", preProc = c("c
enter", "scale"),   tuneLength = 20, trControl = cros_val)
print(lasso_model)
```

```
## The lasso
##
## 117 samples
## 388 predictors
##
## Pre-processing: centered (388), scaled (388)
## Resampling: Cross-Validated (10 fold)
## Summary of sample sizes: 105, 105, 106, 105, 105, 105, ...
## Resampling results across tuning parameters:
##
##   fraction   RMSE      Rsquared   MAE
##   0.1000000  11.03291  0.5004920   8.017154
##   0.1421053  11.01264  0.4815066   7.900521
##   0.1842105  11.23266  0.4447080   8.099254
##   0.2263158  11.31899  0.4328353   8.167599
##   0.2684211  11.28798  0.4317873   8.178458
##   0.3105263  11.34393  0.4257644   8.220985
##   0.3526316  11.49308  0.4142110   8.352751
##   0.3947368  11.66008  0.4003222   8.553292
##   0.4368421  11.82742  0.3881127   8.746691
##   0.4789474  11.96278  0.3804394   8.909622
##   0.5210526  12.08143  0.3753035   9.039031
##   0.5631579  12.18870  0.3708012   9.146026
##   0.6052632  12.27187  0.3672951   9.235108
##   0.6473684  12.38499  0.3628674   9.329889
##   0.6894737  12.48411  0.3586532   9.419492
##   0.7315789  12.68825  0.3503381   9.597973
##   0.7736842  12.93145  0.3394492   9.808901
##   0.8157895  13.13502  0.3307069   9.979756
##   0.8578947  13.33402  0.3230556  10.129946
##   0.9000000  13.54018  0.3145848  10.285269
##
## RMSE was used to select the optimal model using the smallest value.
## The final value used for the model was fraction = 0.1421053.
```

```
#fraction   RMSE      Rsquared   MAE
#  0.1842105  10.53842  0.5652147  7.571420

test_preds_lasso <- predict(lasso_model, newdata = test_data)
print(postResample(pred = test_preds_lasso, obs = test_data$permeability))
```

```
##       RMSE   Rsquared        MAE
## 12.8780797  0.4632927  9.5707901
```

```
#     RMSE   Rsquared        MAE
#14.2522321  0.1930075  9.2703352
```

(f) Would you recommend any of your models to replace the permeability laboratory experiment?

Of all the models I ran, when performed on the test set the ridge regression model had the highest r^2 for the test data at 0.355, while the PLS had a r^2 of 0.27. I would choose the ridge model here as a result.

Question 6.3

A chemical manufacturing process for a pharmaceutical product was discussed in Sect. 1.4. In this problem, the objective is to understand the relationship between biological measurements of the raw materials (predictors), measurements of the manufacturing process (predictors), and the response of product yield. Biological predictors cannot be changed but can be used to assess the quality of the raw material before processing. On the other hand, manufacturing process predictors can be changed in the manufacturing process. Improving product yield by 1 % will boost revenue by approximately one hundred thousand dollars per batch:

(a) Start R and use these commands to load the data:

```
library(AppliedPredictiveModeling)

data(ChemicalManufacturingProcess)
```

The matrix processPredictors contains the 57 predictors (12 describing the input biological material and 45 describing the process predictors) for the 176 manufacturing runs. ‘yield’ contains the percent yield for each run.

(b) A small percentage of cells in the predictor set contain missing values. Use an imputation function to fill in these missing values (e.g., see Sect. 3.8).

```
print(summary(ChemicalManufacturingProcess))
```

```
##      Yield       BiologicalMaterial01 BiologicalMaterial02 BiologicalMaterial03
##  Min.   :35.25   Min.   :4.580        Min.   :46.87        Min.   :56.97
##  1st Qu.:38.75   1st Qu.:5.978        1st Qu.:52.68        1st Qu.:64.98
##  Median :39.97   Median :6.305        Median :55.09        Median :67.22
##  Mean   :40.18   Mean   :6.411        Mean   :55.69        Mean   :67.70
##  3rd Qu.:41.48   3rd Qu.:6.870        3rd Qu.:58.74        3rd Qu.:70.43
##  Max.   :46.34   Max.   :8.810        Max.   :64.75        Max.   :78.25
##
##  BiologicalMaterial04 BiologicalMaterial05 BiologicalMaterial06
##  Min.   : 9.38        Min.   :13.24        Min.   :40.60
##  1st Qu.:11.24        1st Qu.:17.23        1st Qu.:46.05
##  Median :12.10        Median :18.49        Median :48.46
##  Mean   :12.35        Mean   :18.60        Mean   :48.91
##  3rd Qu.:13.22        3rd Qu.:19.90        3rd Qu.:51.34
##  Max.   :23.09        Max.   :24.85        Max.   :59.38
##
##  BiologicalMaterial07 BiologicalMaterial08 BiologicalMaterial09
##  Min.   :100.0        Min.   :15.88        Min.   :11.44
##  1st Qu.:100.0        1st Qu.:17.06        1st Qu.:12.60
##  Median :100.0        Median :17.51        Median :12.84
##  Mean   :100.0        Mean   :17.49        Mean   :12.85
##  3rd Qu.:100.0        3rd Qu.:17.88        3rd Qu.:13.13
##  Max.   :100.8        Max.   :19.14        Max.   :14.08
##
##  BiologicalMaterial10 BiologicalMaterial11 BiologicalMaterial12
##  Min.   :1.770        Min.   :135.8        Min.   :18.35
##  1st Qu.:2.460        1st Qu.:143.8        1st Qu.:19.73
##  Median :2.710        Median :146.1        Median :20.12
##  Mean   :2.801        Mean   :147.0        Mean   :20.20
##  3rd Qu.:2.990        3rd Qu.:149.6        3rd Qu.:20.75
##  Max.   :6.870        Max.   :158.7        Max.   :22.21
##
##  ManufacturingProcess01 ManufacturingProcess02 ManufacturingProcess03
##  Min.   : 0.00          Min.   : 0.00          Min.   :1.47
##  1st Qu.:10.80          1st Qu.:19.30          1st Qu.:1.53
##  Median :11.40          Median :21.00          Median :1.54
##  Mean   :11.21          Mean   :16.68          Mean   :1.54
##  3rd Qu.:12.15          3rd Qu.:21.50          3rd Qu.:1.55
##  Max.   :14.10          Max.   :22.50          Max.   :1.60
##  NA's   :1              NA's   :3              NA's   :15
##  ManufacturingProcess04 ManufacturingProcess05 ManufacturingProcess06
##  Min.   :911.0          Min.   : 923.0         Min.   :203.0
##  1st Qu.:928.0          1st Qu.: 986.8         1st Qu.:205.7
##  Median :934.0          Median : 999.2         Median :206.8
##  Mean   :931.9          Mean   :1001.7         Mean   :207.4
##  3rd Qu.:936.0          3rd Qu.:1008.9         3rd Qu.:208.7
##  Max.   :946.0          Max.   :1175.3         Max.   :227.4
##  NA's   :1              NA's   :1              NA's   :2
##  ManufacturingProcess07 ManufacturingProcess08 ManufacturingProcess09
##  Min.   :177.0          Min.   :177.0          Min.   :38.89
##  1st Qu.:177.0          1st Qu.:177.0          1st Qu.:44.89
```

```
##  Median :177.0          Median :178.0          Median :45.73
##  Mean   :177.5          Mean   :177.6          Mean   :45.66
##  3rd Qu.:178.0          3rd Qu.:178.0          3rd Qu.:46.52
##  Max.   :178.0          Max.   :178.0          Max.   :49.36
##  NA's   :1              NA's   :1
##  ManufacturingProcess10 ManufacturingProcess11 ManufacturingProcess12
##  Min.   : 7.500         Min.   : 7.500         Min.   :   0.0
##  1st Qu.: 8.700         1st Qu.: 9.000         1st Qu.:   0.0
##  Median : 9.100         Median : 9.400         Median :   0.0
##  Mean   : 9.179         Mean   : 9.386         Mean   : 857.8
##  3rd Qu.: 9.550         3rd Qu.: 9.900         3rd Qu.:   0.0
##  Max.   :11.600         Max.   :11.500         Max.   :4549.0
##  NA's   :9              NA's   :10             NA's   :1
##  ManufacturingProcess13 ManufacturingProcess14 ManufacturingProcess15
##  Min.   :32.10          Min.   :4701           Min.   :5904
##  1st Qu.:33.90          1st Qu.:4828           1st Qu.:6010
##  Median :34.60          Median :4856           Median :6032
##  Mean   :34.51          Mean   :4854           Mean   :6039
##  3rd Qu.:35.20          3rd Qu.:4882           3rd Qu.:6061
##  Max.   :38.60          Max.   :5055           Max.   :6233
##                         NA's   :1
##  ManufacturingProcess16 ManufacturingProcess17 ManufacturingProcess18
##  Min.   :   0           Min.   :31.30          Min.   :   0
##  1st Qu.:4561           1st Qu.:33.50          1st Qu.:4813
##  Median :4588           Median :34.40          Median :4835
##  Mean   :4566           Mean   :34.34          Mean   :4810
##  3rd Qu.:4619           3rd Qu.:35.10          3rd Qu.:4862
##  Max.   :4852           Max.   :40.00          Max.   :4971
##
##  ManufacturingProcess19 ManufacturingProcess20 ManufacturingProcess21
##  Min.   :5890           Min.   :   0           Min.   :-1.8000
##  1st Qu.:6001           1st Qu.:4553           1st Qu.:-0.6000
##  Median :6022           Median :4582           Median :-0.3000
##  Mean   :6028           Mean   :4556           Mean   :-0.1642
##  3rd Qu.:6050           3rd Qu.:4610           3rd Qu.: 0.0000
##  Max.   :6146           Max.   :4759           Max.   : 3.6000
##
##  ManufacturingProcess22 ManufacturingProcess23 ManufacturingProcess24
##  Min.   : 0.000         Min.   :0.000          Min.   : 0.000
##  1st Qu.: 3.000         1st Qu.:2.000          1st Qu.: 4.000
##  Median : 5.000         Median :3.000          Median : 8.000
##  Mean   : 5.406         Mean   :3.017          Mean   : 8.834
##  3rd Qu.: 8.000         3rd Qu.:4.000          3rd Qu.:14.000
##  Max.   :12.000         Max.   :6.000          Max.   :23.000
##  NA's   :1              NA's   :1              NA's   :1
##  ManufacturingProcess25 ManufacturingProcess26 ManufacturingProcess27
##  Min.   :   0           Min.   :   0           Min.   :   0
##  1st Qu.:4832           1st Qu.:6020           1st Qu.:4560
##  Median :4855           Median :6047           Median :4587
##  Mean   :4828           Mean   :6016           Mean   :4563
##  3rd Qu.:4877           3rd Qu.:6070           3rd Qu.:4609
```

```
##  Max.   :4990           Max.   :6161           Max.   :4710
##  NA's   :5              NA's   :5              NA's   :5
##  ManufacturingProcess28 ManufacturingProcess29 ManufacturingProcess30
##  Min.   : 0.000         Min.   : 0.00          Min.   : 0.000
##  1st Qu.: 0.000         1st Qu.:19.70          1st Qu.: 8.800
##  Median :10.400         Median :19.90          Median : 9.100
##  Mean   : 6.592         Mean   :20.01          Mean   : 9.161
##  3rd Qu.:10.750         3rd Qu.:20.40          3rd Qu.: 9.700
##  Max.   :11.500         Max.   :22.00          Max.   :11.200
##  NA's   :5              NA's   :5              NA's   :5
##  ManufacturingProcess31 ManufacturingProcess32 ManufacturingProcess33
##  Min.   : 0.00          Min.   :143.0          Min.   :56.00
##  1st Qu.:70.10          1st Qu.:155.0          1st Qu.:62.00
##  Median :70.80          Median :158.0          Median :64.00
##  Mean   :70.18          Mean   :158.5          Mean   :63.54
##  3rd Qu.:71.40          3rd Qu.:162.0          3rd Qu.:65.00
##  Max.   :72.50          Max.   :173.0          Max.   :70.00
##  NA's   :5                                     NA's   :5
##  ManufacturingProcess34 ManufacturingProcess35 ManufacturingProcess36
##  Min.   :2.300          Min.   :463.0          Min.   :0.01700
##  1st Qu.:2.500          1st Qu.:490.0          1st Qu.:0.01900
##  Median :2.500          Median :495.0          Median :0.02000
##  Mean   :2.494          Mean   :495.6          Mean   :0.01957
##  3rd Qu.:2.500          3rd Qu.:501.5          3rd Qu.:0.02000
##  Max.   :2.600          Max.   :522.0          Max.   :0.02200
##  NA's   :5              NA's   :5              NA's   :5
##  ManufacturingProcess37 ManufacturingProcess38 ManufacturingProcess39
##  Min.   :0.000          Min.   :0.000          Min.   :0.000
##  1st Qu.:0.700          1st Qu.:2.000          1st Qu.:7.100
##  Median :1.000          Median :3.000          Median :7.200
##  Mean   :1.014          Mean   :2.534          Mean   :6.851
##  3rd Qu.:1.300          3rd Qu.:3.000          3rd Qu.:7.300
##  Max.   :2.300          Max.   :3.000          Max.   :7.500
##
##  ManufacturingProcess40 ManufacturingProcess41 ManufacturingProcess42
##  Min.   :0.00000        Min.   :0.00000        Min.   : 0.00
##  1st Qu.:0.00000        1st Qu.:0.00000        1st Qu.:11.40
##  Median :0.00000        Median :0.00000        Median :11.60
##  Mean   :0.01771        Mean   :0.02371        Mean   :11.21
##  3rd Qu.:0.00000        3rd Qu.:0.00000        3rd Qu.:11.70
##  Max.   :0.10000        Max.   :0.20000        Max.   :12.10
##  NA's   :1              NA's   :1
##  ManufacturingProcess43 ManufacturingProcess44 ManufacturingProcess45
##  Min.   : 0.0000        Min.   :0.000          Min.   :0.000
##  1st Qu.: 0.6000        1st Qu.:1.800          1st Qu.:2.100
##  Median : 0.8000        Median :1.900          Median :2.200
##  Mean   : 0.9119        Mean   :1.805          Mean   :2.138
##  3rd Qu.: 1.0250        3rd Qu.:1.900          3rd Qu.:2.300
##  Max.   :11.0000        Max.   :2.100          Max.   :2.600
##
```

```
## Columns that contain null/ NA values.
#ManufacturingProcess01 ManufacturingProcess02 ManufacturingProcess03 ManufacturingProcess04
ManufacturingProcess05 ManufacturingProcess06 ManufacturingProcess07
#ManufacturingProcess08 ManufacturingProcess10 ManufacturingProcess11 ManufacturingProcess12
ManufacturingProcess14 ManufacturingProcess22 ManufacturingProcess23
#ManufacturingProcess24 ManufacturingProcess25 ManufacturingProcess26 ManufacturingProcess27
ManufacturingProcess28 ManufacturingProcess29 ManufacturingProcess30
#ManufacturingProcess31 ManufacturingProcess33 ManufacturingProcess34 ManufacturingProcess35
ManufacturingProcess36 ManufacturingProcess40 ManufacturingProcess41

## Imputing the values.
chem_df <- data.frame(ChemicalManufacturingProcess)
## Keeping it simple with taking the median values of each of the columns, as most columns th
at have NA only have one NA
imputed <- preProcess(chem_df, method = "medianImpute")
chem_df_imputed <- predict(imputed, chem_df)

print(summary(chem_df_imputed)) # No more nulls
```

```
##      Yield       BiologicalMaterial01 BiologicalMaterial02 BiologicalMaterial03
##  Min.   :35.25   Min.   :4.580        Min.   :46.87        Min.   :56.97
##  1st Qu.:38.75   1st Qu.:5.978        1st Qu.:52.68        1st Qu.:64.98
##  Median :39.97   Median :6.305        Median :55.09        Median :67.22
##  Mean   :40.18   Mean   :6.411        Mean   :55.69        Mean   :67.70
##  3rd Qu.:41.48   3rd Qu.:6.870        3rd Qu.:58.74        3rd Qu.:70.43
##  Max.   :46.34   Max.   :8.810        Max.   :64.75        Max.   :78.25
##  BiologicalMaterial04 BiologicalMaterial05 BiologicalMaterial06
##  Min.   : 9.38        Min.   :13.24        Min.   :40.60
##  1st Qu.:11.24        1st Qu.:17.23        1st Qu.:46.05
##  Median :12.10        Median :18.49        Median :48.46
##  Mean   :12.35        Mean   :18.60        Mean   :48.91
##  3rd Qu.:13.22        3rd Qu.:19.90        3rd Qu.:51.34
##  Max.   :23.09        Max.   :24.85        Max.   :59.38
##  BiologicalMaterial07 BiologicalMaterial08 BiologicalMaterial09
##  Min.   :100.0        Min.   :15.88        Min.   :11.44
##  1st Qu.:100.0        1st Qu.:17.06        1st Qu.:12.60
##  Median :100.0        Median :17.51        Median :12.84
##  Mean   :100.0        Mean   :17.49        Mean   :12.85
##  3rd Qu.:100.0        3rd Qu.:17.88        3rd Qu.:13.13
##  Max.   :100.8        Max.   :19.14        Max.   :14.08
##  BiologicalMaterial10 BiologicalMaterial11 BiologicalMaterial12
##  Min.   :1.770        Min.   :135.8        Min.   :18.35
##  1st Qu.:2.460        1st Qu.:143.8        1st Qu.:19.73
##  Median :2.710        Median :146.1        Median :20.12
##  Mean   :2.801        Mean   :147.0        Mean   :20.20
##  3rd Qu.:2.990        3rd Qu.:149.6        3rd Qu.:20.75
##  Max.   :6.870        Max.   :158.7        Max.   :22.21
##  ManufacturingProcess01 ManufacturingProcess02 ManufacturingProcess03
##  Min.   : 0.00          Min.   : 0.00          Min.   :1.47
##  1st Qu.:10.80          1st Qu.:19.30          1st Qu.:1.53
##  Median :11.40          Median :21.00          Median :1.54
##  Mean   :11.21          Mean   :16.76          Mean   :1.54
##  3rd Qu.:12.12          3rd Qu.:21.50          3rd Qu.:1.55
##  Max.   :14.10          Max.   :22.50          Max.   :1.60
##  ManufacturingProcess04 ManufacturingProcess05 ManufacturingProcess06
##  Min.   :911.0          Min.   : 923.0         Min.   :203.0
##  1st Qu.:928.0          1st Qu.: 986.8         1st Qu.:205.7
##  Median :934.0          Median : 999.2         Median :206.8
##  Mean   :931.9          Mean   :1001.7         Mean   :207.4
##  3rd Qu.:936.0          3rd Qu.:1008.7         3rd Qu.:208.7
##  Max.   :946.0          Max.   :1175.3         Max.   :227.4
##  ManufacturingProcess07 ManufacturingProcess08 ManufacturingProcess09
##  Min.   :177.0          Min.   :177.0          Min.   :38.89
##  1st Qu.:177.0          1st Qu.:177.0          1st Qu.:44.89
##  Median :177.0          Median :178.0          Median :45.73
##  Mean   :177.5          Mean   :177.6          Mean   :45.66
##  3rd Qu.:178.0          3rd Qu.:178.0          3rd Qu.:46.52
##  Max.   :178.0          Max.   :178.0          Max.   :49.36
##  ManufacturingProcess10 ManufacturingProcess11 ManufacturingProcess12
##  Min.   : 7.500         Min.   : 7.500         Min.   :   0.0
```

```
##  1st Qu.: 8.700         1st Qu.: 9.000         1st Qu.:   0.0
##  Median : 9.100         Median : 9.400         Median :   0.0
##  Mean   : 9.175         Mean   : 9.386         Mean   : 852.9
##  3rd Qu.: 9.500         3rd Qu.: 9.825         3rd Qu.:   0.0
##  Max.   :11.600         Max.   :11.500         Max.   :4549.0
##  ManufacturingProcess13 ManufacturingProcess14 ManufacturingProcess15
##  Min.   :32.10          Min.   :4701           Min.   :5904
##  1st Qu.:33.90          1st Qu.:4828           1st Qu.:6010
##  Median :34.60          Median :4856           Median :6032
##  Mean   :34.51          Mean   :4854           Mean   :6039
##  3rd Qu.:35.20          3rd Qu.:4882           3rd Qu.:6061
##  Max.   :38.60          Max.   :5055           Max.   :6233
##  ManufacturingProcess16 ManufacturingProcess17 ManufacturingProcess18
##  Min.   :   0           Min.   :31.30          Min.   :   0
##  1st Qu.:4561           1st Qu.:33.50          1st Qu.:4813
##  Median :4588           Median :34.40          Median :4835
##  Mean   :4566           Mean   :34.34          Mean   :4810
##  3rd Qu.:4619           3rd Qu.:35.10          3rd Qu.:4862
##  Max.   :4852           Max.   :40.00          Max.   :4971
##  ManufacturingProcess19 ManufacturingProcess20 ManufacturingProcess21
##  Min.   :5890           Min.   :   0           Min.   :-1.8000
##  1st Qu.:6001           1st Qu.:4553           1st Qu.:-0.6000
##  Median :6022           Median :4582           Median :-0.3000
##  Mean   :6028           Mean   :4556           Mean   :-0.1642
##  3rd Qu.:6050           3rd Qu.:4610           3rd Qu.: 0.0000
##  Max.   :6146           Max.   :4759           Max.   : 3.6000
##  ManufacturingProcess22 ManufacturingProcess23 ManufacturingProcess24
##  Min.   : 0.000         Min.   :0.000          Min.   : 0.00
##  1st Qu.: 3.000         1st Qu.:2.000          1st Qu.: 4.00
##  Median : 5.000         Median :3.000          Median : 8.00
##  Mean   : 5.403         Mean   :3.017          Mean   : 8.83
##  3rd Qu.: 8.000         3rd Qu.:4.000          3rd Qu.:14.00
##  Max.   :12.000         Max.   :6.000          Max.   :23.00
##  ManufacturingProcess25 ManufacturingProcess26 ManufacturingProcess27
##  Min.   :   0           Min.   :   0           Min.   :   0
##  1st Qu.:4834           1st Qu.:6021           1st Qu.:4563
##  Median :4855           Median :6047           Median :4587
##  Mean   :4829           Mean   :6016           Mean   :4563
##  3rd Qu.:4876           3rd Qu.:6069           3rd Qu.:4609
##  Max.   :4990           Max.   :6161           Max.   :4710
##  ManufacturingProcess28 ManufacturingProcess29 ManufacturingProcess30
##  Min.   : 0.0           Min.   : 0.00          Min.   : 0.00
##  1st Qu.: 0.0           1st Qu.:19.70          1st Qu.: 8.80
##  Median :10.4           Median :19.90          Median : 9.10
##  Mean   : 6.7           Mean   :20.01          Mean   : 9.16
##  3rd Qu.:10.7           3rd Qu.:20.40          3rd Qu.: 9.70
##  Max.   :11.5           Max.   :22.00          Max.   :11.20
##  ManufacturingProcess31 ManufacturingProcess32 ManufacturingProcess33
##  Min.   : 0.0           Min.   :143.0          Min.   :56.00
##  1st Qu.:70.1           1st Qu.:155.0          1st Qu.:62.00
##  Median :70.8           Median :158.0          Median :64.00
```

```
##  Mean   :70.2           Mean   :158.5          Mean   :63.56
##  3rd Qu.:71.4           3rd Qu.:162.0          3rd Qu.:65.00
##  Max.   :72.5           Max.   :173.0          Max.   :70.00
##  ManufacturingProcess34 ManufacturingProcess35 ManufacturingProcess36
##  Min.   :2.300          Min.   :463.0          Min.   :0.01700
##  1st Qu.:2.500          1st Qu.:490.0          1st Qu.:0.01900
##  Median :2.500          Median :495.0          Median :0.02000
##  Mean   :2.494          Mean   :495.6          Mean   :0.01959
##  3rd Qu.:2.500          3rd Qu.:501.0          3rd Qu.:0.02000
##  Max.   :2.600          Max.   :522.0          Max.   :0.02200
##  ManufacturingProcess37 ManufacturingProcess38 ManufacturingProcess39
##  Min.   :0.000          Min.   :0.000          Min.   :0.000
##  1st Qu.:0.700          1st Qu.:2.000          1st Qu.:7.100
##  Median :1.000          Median :3.000          Median :7.200
##  Mean   :1.014          Mean   :2.534          Mean   :6.851
##  3rd Qu.:1.300          3rd Qu.:3.000          3rd Qu.:7.300
##  Max.   :2.300          Max.   :3.000          Max.   :7.500
##  ManufacturingProcess40 ManufacturingProcess41 ManufacturingProcess42
##  Min.   :0.00000        Min.   :0.00000        Min.   : 0.00
##  1st Qu.:0.00000        1st Qu.:0.00000        1st Qu.:11.40
##  Median :0.00000        Median :0.00000        Median :11.60
##  Mean   :0.01761        Mean   :0.02358        Mean   :11.21
##  3rd Qu.:0.00000        3rd Qu.:0.00000        3rd Qu.:11.70
##  Max.   :0.10000        Max.   :0.20000        Max.   :12.10
##  ManufacturingProcess43 ManufacturingProcess44 ManufacturingProcess45
##  Min.   : 0.0000        Min.   :0.000          Min.   :0.000
##  1st Qu.: 0.6000        1st Qu.:1.800          1st Qu.:2.100
##  Median : 0.8000        Median :1.900          Median :2.200
##  Mean   : 0.9119        Mean   :1.805          Mean   :2.138
##  3rd Qu.: 1.0250        3rd Qu.:1.900          3rd Qu.:2.300
##  Max.   :11.0000        Max.   :2.100          Max.   :2.600
```

(c) Split the data into a training and a test set, pre-process the data, and tune a model of your choice from this chapter. What is the optimal value of the performance metric?

```
## Splitting the same way in previous question with 70 / 30 train/test
training_split <- createDataPartition(chem_df_imputed$Yield, p = 0.7, list = FALSE)
training_data <- chem_df_imputed[training_split,]

print(nrow(training_data)) #124
```

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

```
print(nrow(t(training_data))) #58
```

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

```
test_data <- chem_df_imputed[-training_split,]
print(nrow(test_data)) #52
```

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

```
print(nrow(t(test_data))) #58
```

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

```
#Data is joined and split into training and test groups.

### Model Start
## crossvalidation
cross_val <- trainControl(method = "cv", number = 10)
## Choosing PLS as before because a lot of predictors
pls_model <- train(Yield ~ ., data = training_data, method = "pls", preProc = c("center", "sc
ale"),   tuneLength = 20, trControl = cros_val)
```

```
## Warning in preProcess.default(thresh = 0.95, k = 5, freqCut = 19, uniqueCut =
## 10, : These variables have zero variances: BiologicalMaterial07
```

```
print(pls_model)
```

```
## Partial Least Squares
##
## 124 samples
##  57 predictor
##
## Pre-processing: centered (57), scaled (57)
## Resampling: Cross-Validated (10 fold)
## Summary of sample sizes: 111, 112, 111, 110, 112, 112, ...
## Resampling results across tuning parameters:
##
##   ncomp  RMSE      Rsquared   MAE
##    1     1.743454  0.3848032  1.265995
##    2     2.090435  0.4483916  1.299609
##    3     1.684349  0.4884593  1.137722
##    4     1.917514  0.4703228  1.200486
##    5     2.164873  0.4822886  1.290156
##    6     2.194309  0.4662026  1.302664
##    7     2.398166  0.4509602  1.393619
##    8     2.503817  0.4398075  1.462878
##    9     2.603515  0.4302625  1.511631
##   10     2.640057  0.4299287  1.531501
##   11     2.523398  0.4362536  1.486035
##   12     2.407645  0.4283151  1.469341
##   13     2.394987  0.4193345  1.484479
##   14     2.280194  0.4175881  1.472641
##   15     2.345575  0.4154113  1.508872
##   16     2.430022  0.4096232  1.556598
##   17     2.600111  0.4043572  1.641440
##   18     2.882322  0.4036897  1.749258
##   19     3.098463  0.4029102  1.826827
##   20     3.361116  0.4023438  1.925809
##
## RMSE was used to select the optimal model using the smallest value.
## The final value used for the model was ncomp = 3.
```

```
 #ncomp  RMSE      Rsquared   MAE
#   3     1.319743  0.5803008  1.050852

## The model chose three components from the PLS predictors as the optimal value.
```

(d) Predict the response for the test set. What is the value of the performance metric and how does this compare with the resampled performance metric on the training set?

```
## Running it on the test data
test_preds <- predict(pls_model, newdata = test_data)

print(postResample(pred = test_preds, obs = test_data$Yield))
```

```
##      RMSE  Rsquared       MAE
## 1.1703978 0.6537831 0.9595592
```

```
# RMSE  Rsquared       MAE
#1.9545288 0.2765253 1.1098108


## The r^2 opn the test data from the Pls model trained on the trainingdata is lower than the training data. The R^2 on the test data is 0,27 with a RMSE of 1.95, while on the training data it the r^2 was 0.58 with a root mean sqrd error of ~1.3. The model performed more poorly on the test data.
```

(e) Which predictors are most important in the model you have trained? Do either the biological or process predictors dominate the list?

```
# Checking the varibles for what is important
print(varImp(pls_model))
```

```
## Warning: package 'pls' was built under R version 4.4.3
```

```
##
## Attaching package: 'pls'
```

```
## The following object is masked from 'package:caret':
##
##     R2
```

```
## The following object is masked from 'package:stats':
##
##     loadings
```

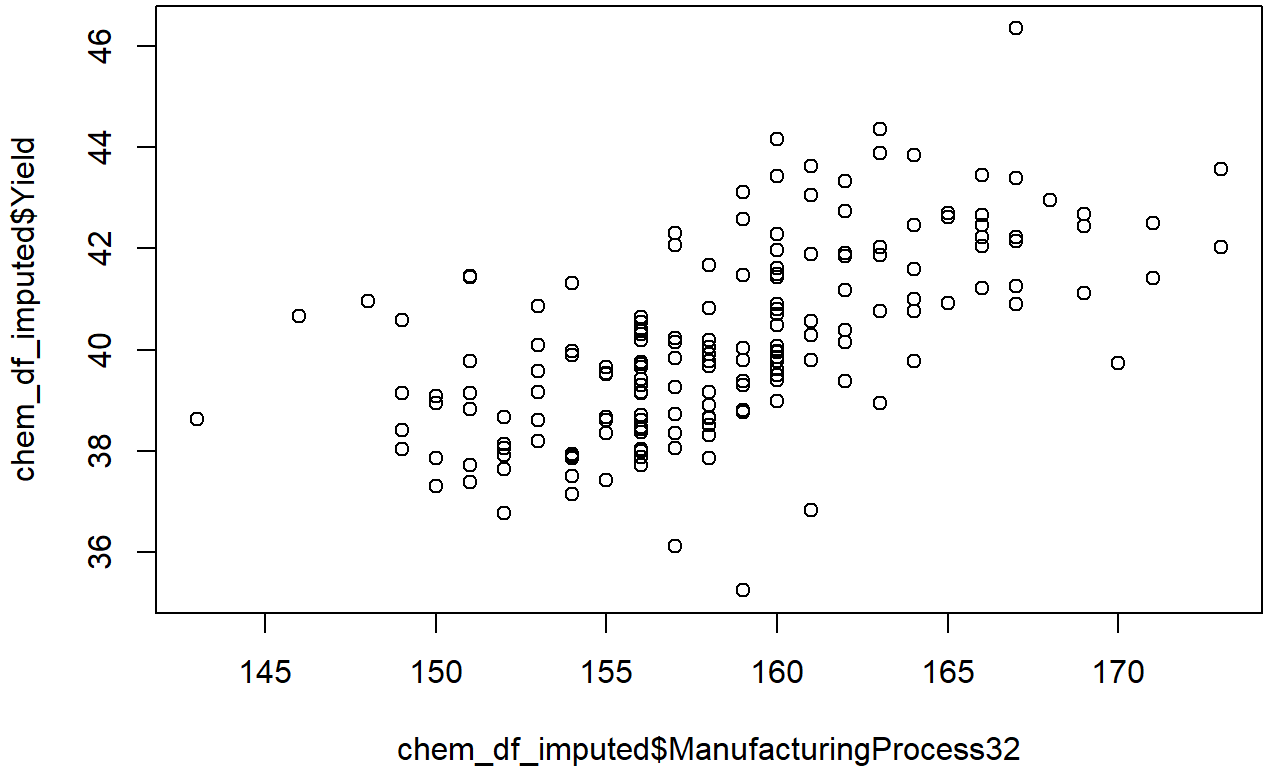
```
## pls variable importance
##
##   only 20 most important variables shown (out of 57)
##
##                         Overall
## ManufacturingProcess32  100.00
## ManufacturingProcess09   87.65
## ManufacturingProcess13   83.75
## ManufacturingProcess17   77.10
## ManufacturingProcess36   73.43
## ManufacturingProcess06   70.85
## ManufacturingProcess11   58.80
## ManufacturingProcess12   55.36
## BiologicalMaterial02     52.80
## BiologicalMaterial08     51.68
## ManufacturingProcess33   50.41
## BiologicalMaterial06     50.32
## BiologicalMaterial03     46.88
## ManufacturingProcess34   45.70
## ManufacturingProcess37   45.26
## BiologicalMaterial12     44.87
## BiologicalMaterial11     43.38
## BiologicalMaterial01     43.36
## BiologicalMaterial04     42.93
## ManufacturingProcess28   39.26
```

```
## The top 6 predictors in this model are the Manufacturing / Process predictors. So the answ
er would be the process predictors as those that are dominating the list. While there are 3 b
iological predictors in the top 10 variables, 70% are manufacturing / process predic
```

(f) Explore the relationships between each of the top predictors and the response. How could this information be helpful in improving yield in future runs of the manufacturing process?

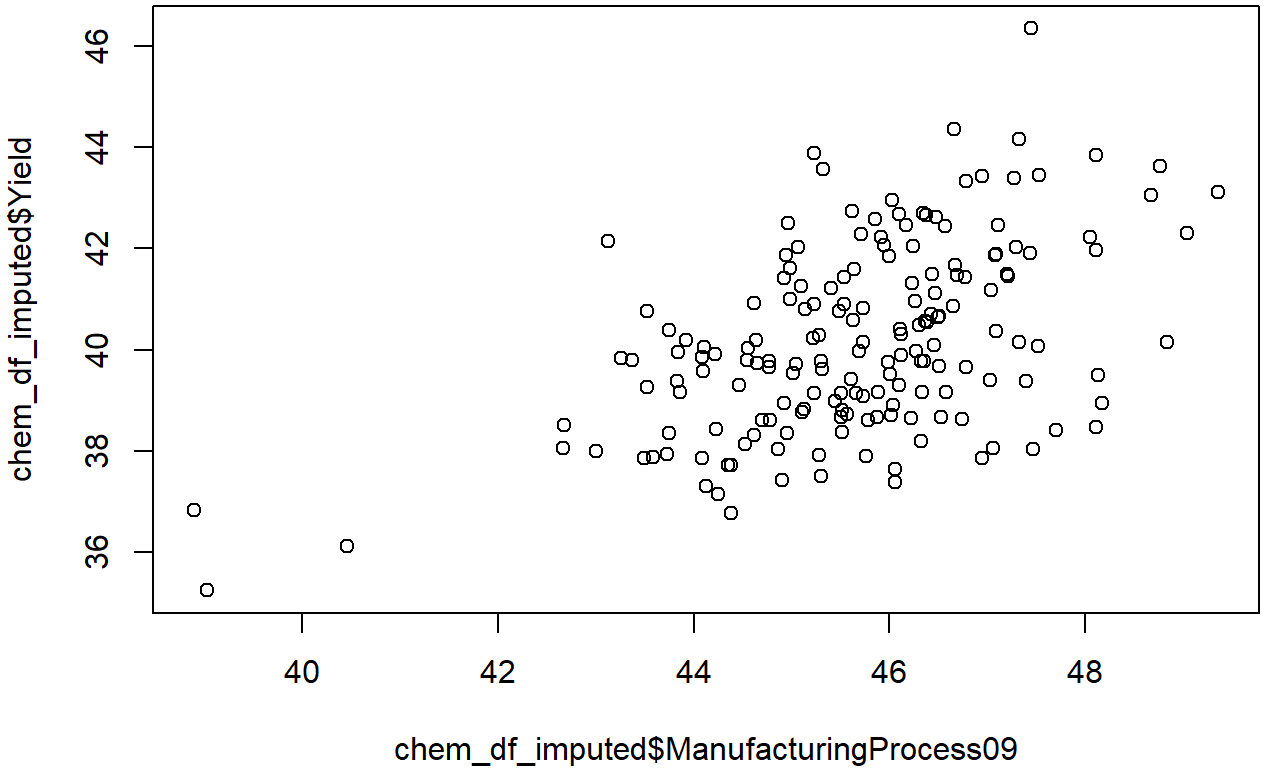
```
## Looking at the top 5 predictors.

print(plot(chem_df_imputed$ManufacturingProcess32, chem_df_imputed$Yield))
```



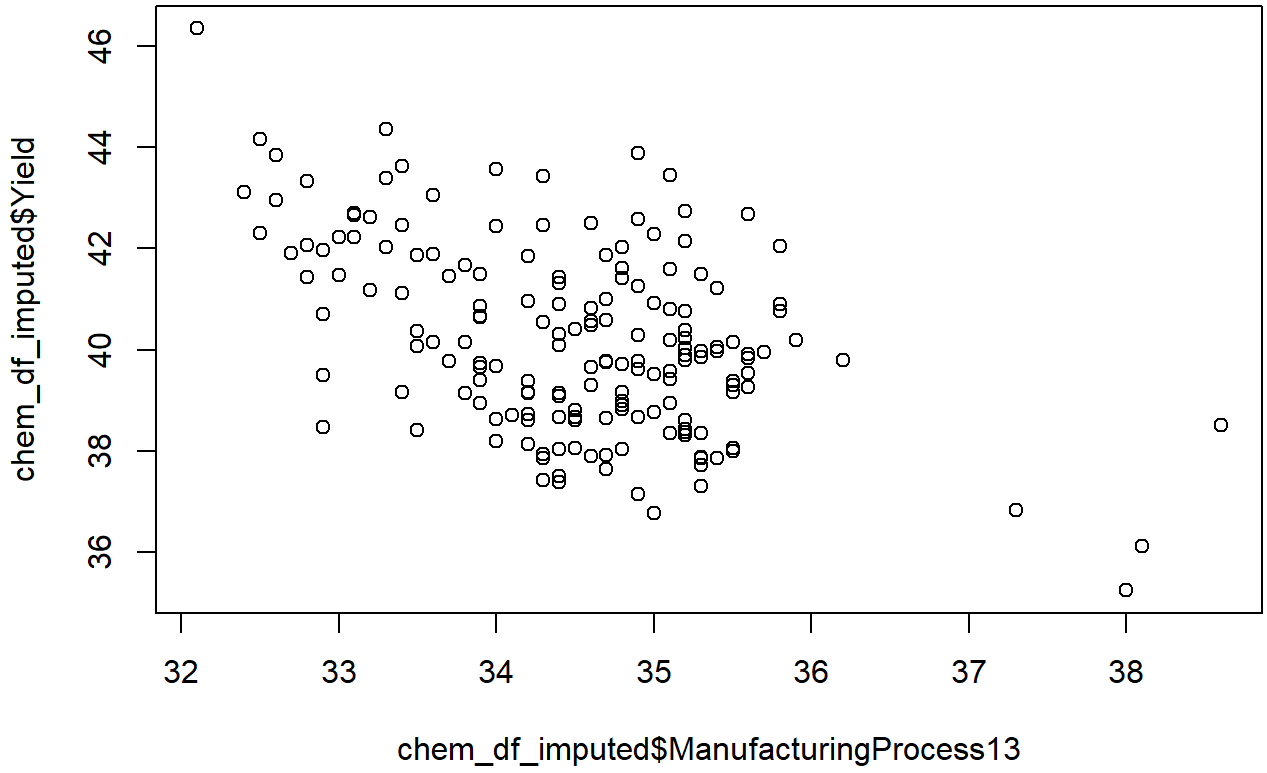

```
## NULL
```

```
print(plot(chem_df_imputed$ManufacturingProcess09, chem_df_imputed$Yield))
```



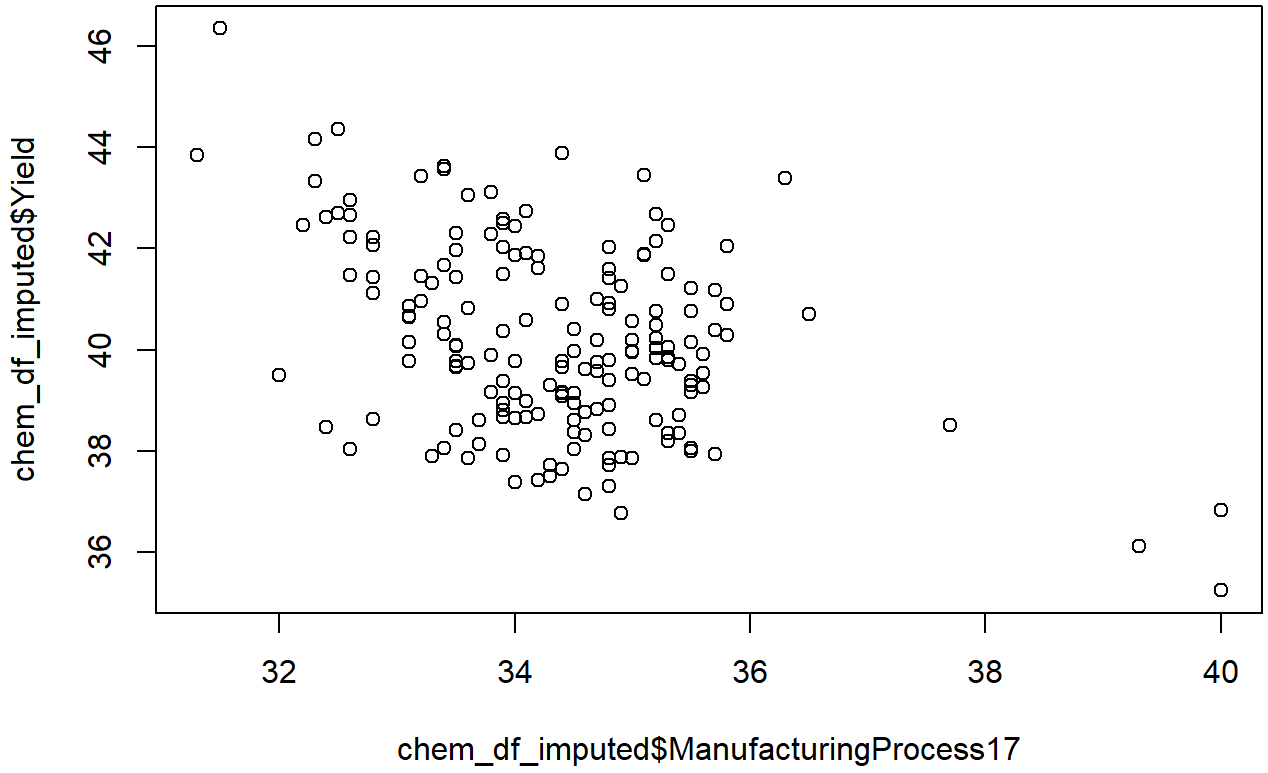

```
## NULL
```

```
print(plot(chem_df_imputed$ManufacturingProcess13, chem_df_imputed$Yield))
```



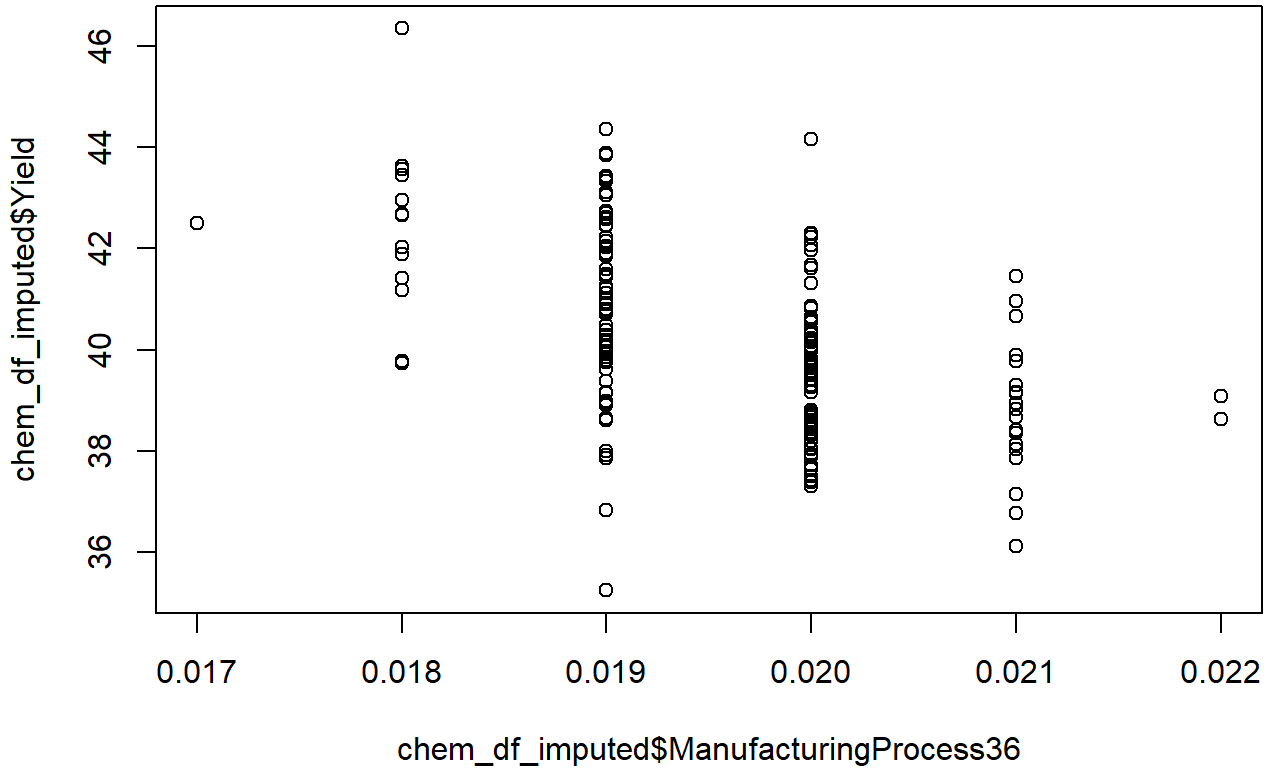

```
## NULL
```

```
print(plot(chem_df_imputed$ManufacturingProcess17, chem_df_imputed$Yield))
```



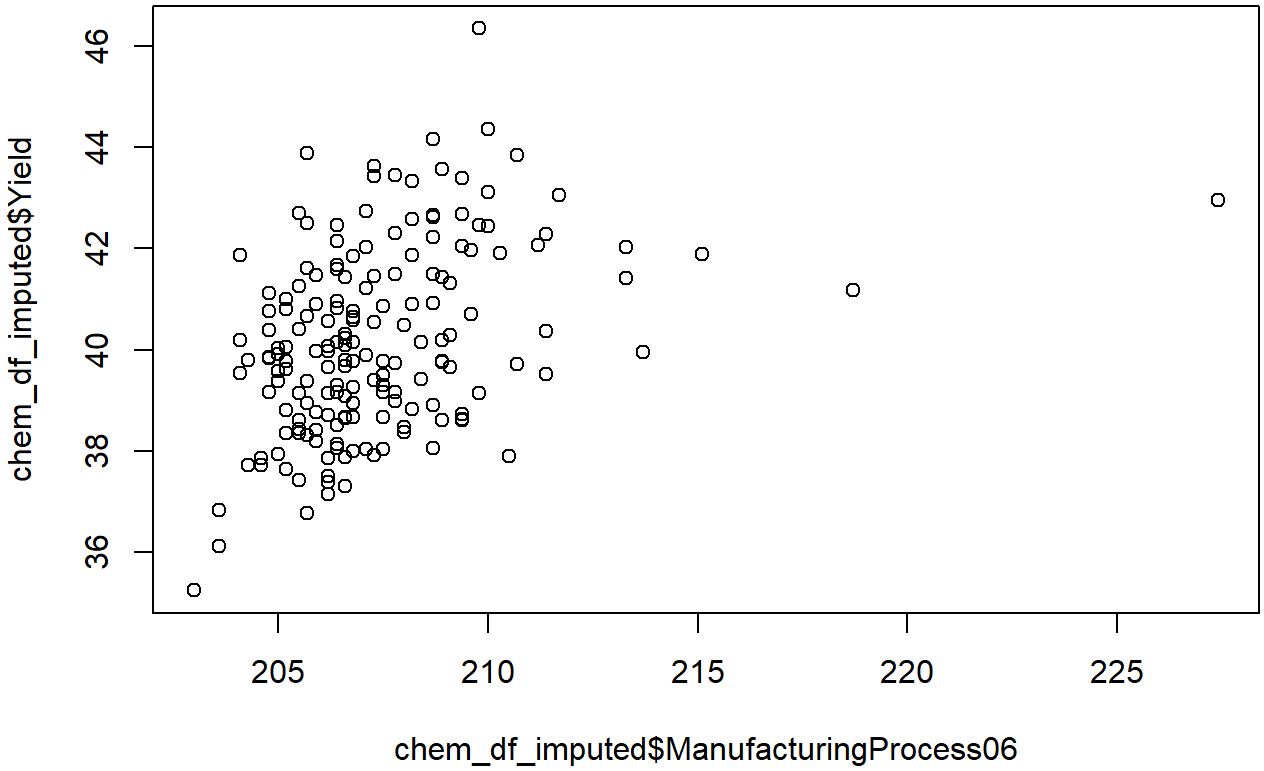

```
## NULL
```

```
print(plot(chem_df_imputed$ManufacturingProcess36, chem_df_imputed$Yield))
```




```
## NULL
```

```
print(plot(chem_df_imputed$ManufacturingProcess06, chem_df_imputed$Yield))
```




```
## NULL
```

The predictors ManufacturingProcess32, ManufacturingProcess09, and ManufacturingProcess06 all have positive correlations with yield. These would be the processes that if improved woul d have the strongest return on yield. ManufacturingProcess36 seems to be impacted by another third variable, but also seems to have a bit of a negative relationship with yield.Lastly, Ma nufacturingProcess13 has little to negative relationship with yield based on the plot. Overal l the first three predictors listed would be the most impactful if improved when looking at a ttempting to increase yield.