Model Validation

The goal in Model Validation is to evaluate the quality of the predictions the model generates (the model's performance). This is performed through cross validation. The model's adequacy has already been established through the residual analysis.

```
library(caret)
```

```
## Warning: package 'caret' was built under R version 4.0.3
```

```
## Loading required package: lattice
```

```
## Loading required package: ggplot2
```

```
data <- read.csv("cleaned_data_5.csv", fileEncoding="UTF-8-BOM")


model <- lm(Weight ~
            Width +
            Length1 +
            Height +
            Species_Pike +
            Species_Smelt +
            Species_Whitefish ,
            data = data)

summary(model)
```

```
##
## Call:
## lm(formula = Weight ~ Width + Length1 + Height + Species_Pike +
##     Species_Smelt + Species_Whitefish, data = data)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -4.6278 -0.5128 -0.0424  0.4406  3.7161
##
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)       -8.16172    0.31660 -25.779  < 2e-16 ***
## Width              1.11711    0.20841   5.360 3.03e-07 ***
## Length1            0.64943    0.04172  15.565  < 2e-16 ***
## Height             0.46026    0.03642  12.636  < 2e-16 ***
## Species_Pike      -3.20450    0.70686  -4.533 1.17e-05 ***
## Species_Smelt      1.63367    0.37477   4.359 2.39e-05 ***
## Species_Whitefish  0.97924    0.44952   2.178   0.0309 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.058 on 152 degrees of freedom
## Multiple R-squared:  0.9871, Adjusted R-squared:  0.9866
## F-statistic:  1933 on 6 and 152 DF,  p-value: < 2.2e-16
```

This is the model determined to be the best through variable selection.

```
set.seed(101)
n_train <- ceiling(0.8 * length(data$Weight))
train_sample <- sample(c(1:length(data$Weight)), n_train)
train_data <- data[train_sample, ]
test_data <- data[-train_sample, ]
```

This randomly splits the data set into a traing data set and a testing data set. To measure the performance of the model and the quality of its predictions, cross validation is used. This involves fitting fitting the model using the training data, and using this model to predict the responses on the test set.

```
model <- lm(Weight ~
            Width +
            Length1 +
            Height +
            Species_Pike +
            Species_Smelt +
            Species_Whitefish ,
            data = train_data)

summary(model)
```

```
##
## Call:
## lm(formula = Weight ~ Width + Length1 + Height + Species_Pike +
##     Species_Smelt + Species_Whitefish, data = train_data)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -4.6995 -0.5485 -0.0137  0.4101  3.2264
##
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)       -8.16271    0.35105 -23.252  < 2e-16 ***
## Width              0.98164    0.22311   4.400 2.35e-05 ***
## Length1            0.67223    0.04552  14.766  < 2e-16 ***
## Height             0.46653    0.04023  11.596  < 2e-16 ***
## Species_Pike      -3.21867    0.80021  -4.022 0.000101 ***
## Species_Smelt      1.50787    0.43260   3.486 0.000685 ***
## Species_Whitefish  1.00205    0.45615   2.197 0.029944 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.064 on 121 degrees of freedom
## Multiple R-squared:  0.9869, Adjusted R-squared:  0.9862
## F-statistic:  1516 on 6 and 121 DF,  p-value: < 2.2e-16
```

```
predictions <- predict(model, test_data)

R_sq <- R2(predictions, test_data$Weight)
RMSE <- RMSE(predictions, test_data$Weight)
MAE <- MAE(predictions, test_data$Weight)
```

```r
print(c(R_sq, RMSE, MAE))
```

```
## [1] 0.9878644 1.0691612 0.8428491
```

The quality of the predictions is about the same using the test data. The root mean square error (RMSE) and mean absolute error (MAE) should be small on a well performing model. To give context to these values, we can divide RMSE by the mean of the response variable to give a prediction error rate.

```r
pred_error <- RMSE / mean(test_data$Weight)
pred_error
```

```
## [1] 0.05897247
```

This means that there is only a 0.0589725 error on average (in the unit of measure weight is). Since the data set is not huge, there could be a risk that observations with a large absolute value could have all been partitioned exclusively to the test or training set (creating bias). Even though the prediction error is low, we have no basis of comparison.

```r
R_sq <- 0
RMSE <- 0
MAE <- 0

for(i in 1:20){

  n_train <- ceiling(0.8 * length(data$Weight))
  train_sample <- sample(c(1:length(data$Weight)), n_train)
  train_data <- data[train_sample, ]
  test_data <- data[-train_sample, ]


  model <- lm(Weight ~
              Width +
              Length1 +
              Height +
              Species_Pike +
              Species_Smelt +
              Species_Whitefish ,
              data = train_data)

  summary(model)

  predictions <- predict(model, test_data)

  R_sq <- R_sq + R2(predictions, test_data$Weight)
  RMSE <- RMSE + RMSE(predictions, test_data$Weight)
  MAE <- MAE + MAE(predictions, test_data$Weight)

}

R_sq = R_sq / 20
RMSE = RMSE / 20
```

```r
MAE = MAE / 20

print(c(R_sq, RMSE, MAE))
```

```
## [1] 0.9866953 1.0384956 0.7575819
```

The average of R^2, RMSE, and MAE is almost the same as what was initially computed. This shows that the inital cross validation was not a one off.