Data Analysis Exam 2

Keltin Grimes

11/6/2021

Introduction

This goal of this report is to predict the age of abalones. The dataset we use for this investigation consists of the following variables:

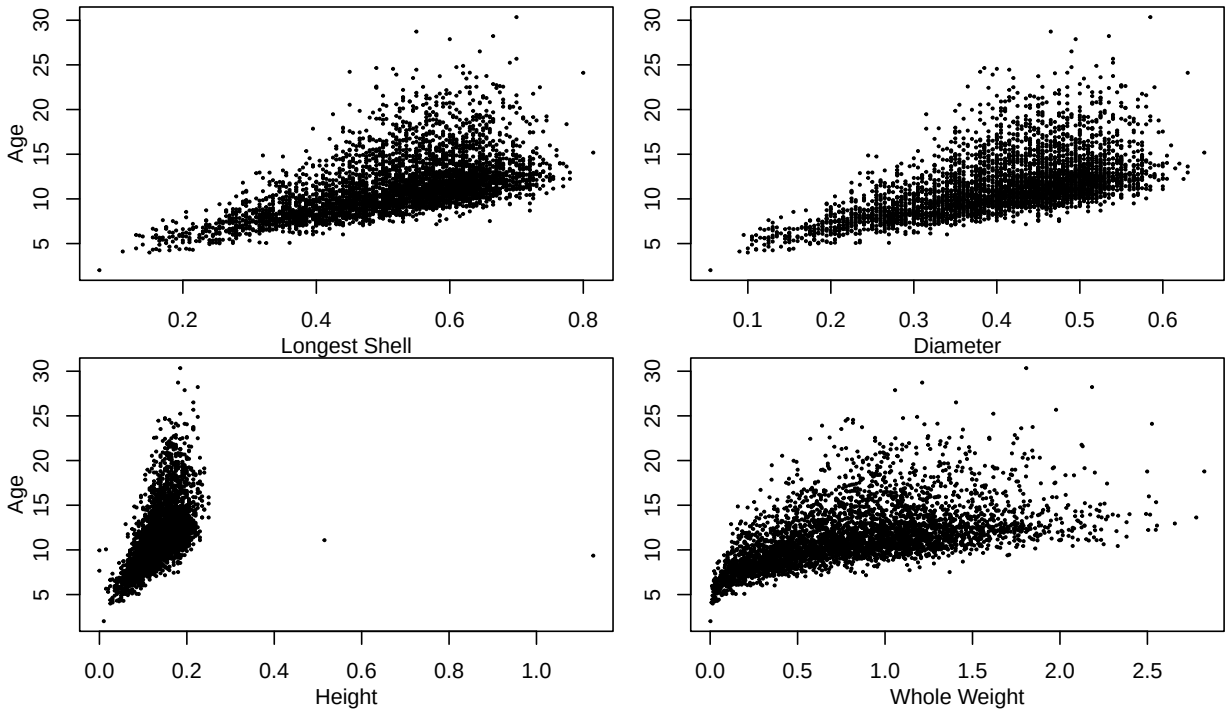
Variable Name	Description
Type	Sex: Male (M), Female (F), and Infant (I)
LongestShell	Longest Shell Measurement
Diameter	Diameter
Height	Height
WholeWeight	Total Weight
ShuckedWeight	Shucked Weight
VisceraWeight	Viscera Weight
ShellWeight	Shell Weight
Age	Abalone Age

Exploratory Data Analysis

The data exhibit three main features which we will need to pay attention to: non-constant variance, non-linear relationships, and correlated covariates. We will describe each in more detail.

Non-Constant Variance

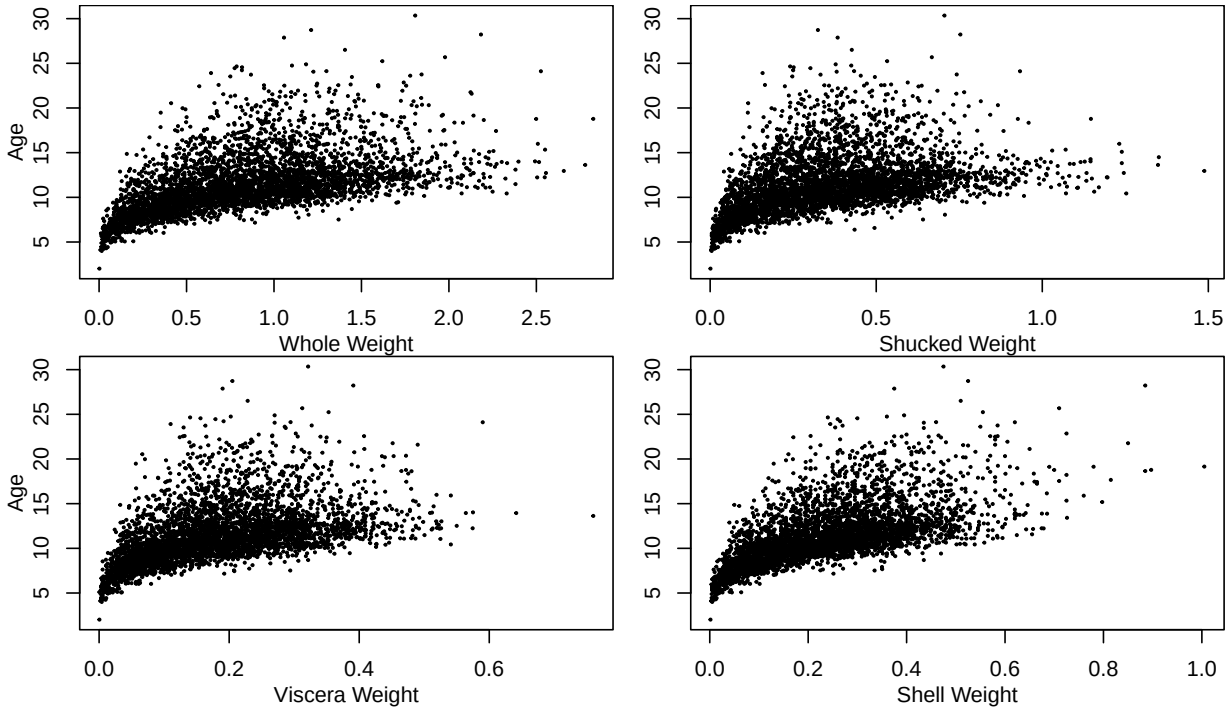
All of the continuous covariates demonstrate non-constant variance, specifically the variance of Age tends to increase as the covariates increase. This is exemplified below by `LongestShell`, `Diameter`, `Height`, and `WholeWeight`.



Examining the variance of the residuals will be crucial, especially for linear regression. Note the two extreme outliers for the `Height` variable; we will have to examine these points closely as well.

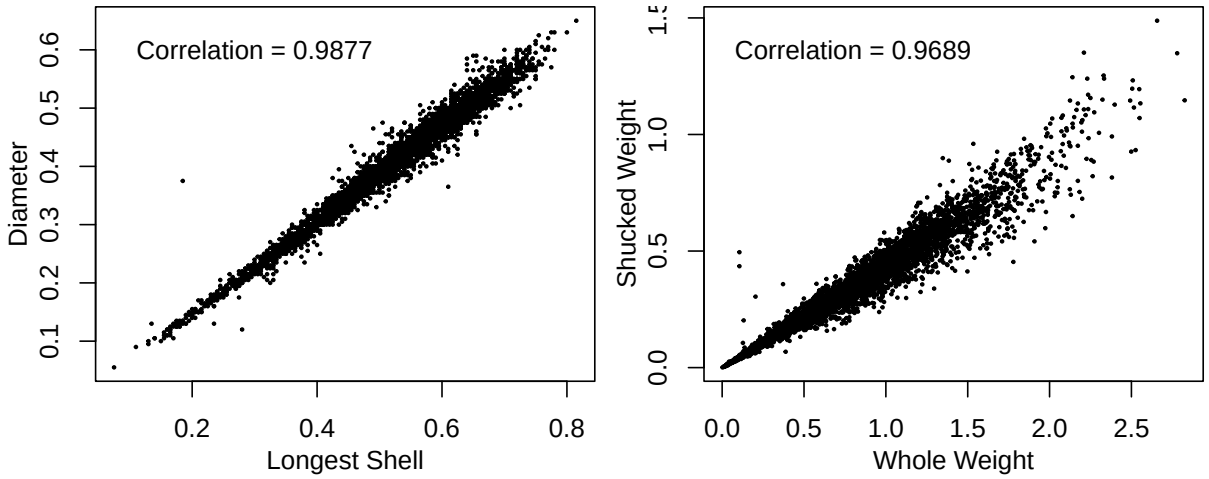
Non-linear Relationships

The relationship between Age and the four weight-based variables is clearly non-linear. We plot them below. We may have to transform these variables when fitting linear models to the data.



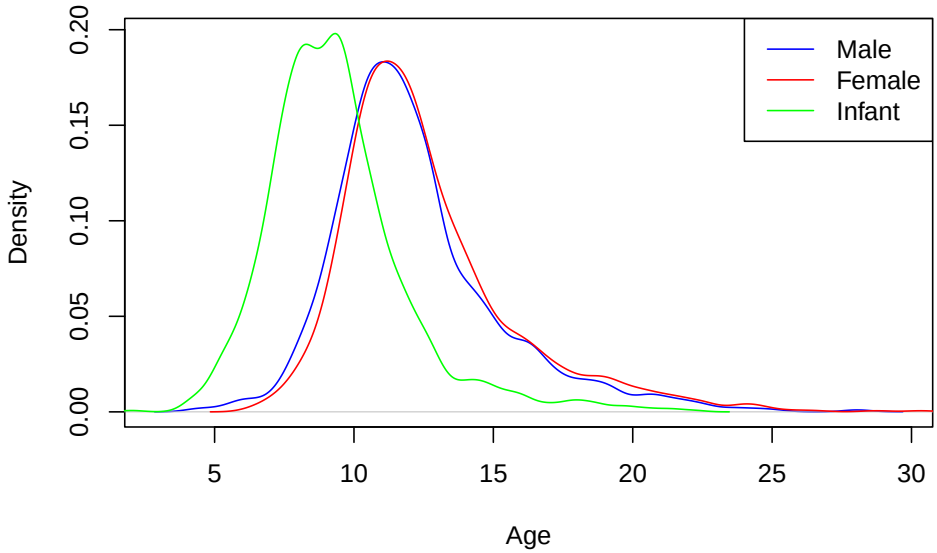
Correlated Covariates

All of the continuous covariates exhibit strong correlation with one another. The smallest Pearson Correlation Coefficient of two covariates is still larger than 0.75, with some pairs even having correlation values over 0.95. This suggests that we may have to model interaction between some of the covariates. Two of the most extreme cases are shown below.



Type

For completeness, we show the estimated density of `Age` for each of the three `Type` categories. We can see that the distribution of Age is almost identical for Males versus Females, but the distribution for Infants is noticably differant than the other two.



Train-Test Splits

For some of the models we plan to investigate it is expensive to perform cross-validation. For this reason we split the data into a training and testing dataset. We randomly assign 3341 observations to the training

set and the remaining 835 to the testing set, constituting approximately 80% and 20% of the total data, respectively. We withold the test set until the very end, regardless of whether it is feasible to perform cross validation, so that we can fairly compare all models.

Modeling

We will build three models: a linear regression model, an additive model, and a random forest.

Linear Regression

We start out by regressing `Age` against all the other variables. This will give us a baseline model with which to compare our other models. We construct the model

$$\text{Age} = \beta_0 + \beta_1 \text{Type}_I + \beta_2 \text{Type}_M + \beta_3 \text{LongestShell} + \beta_4 \text{Diameter} + \beta_5 \text{Height} + \beta_6 \text{WholeWeight} +$$

$$\beta_7 \text{ShuckedWeight} + \beta_8 \text{VisceraWeight} + \beta_9 \text{ShellWeight} + \epsilon$$

where Type_I is the indicator variable for `Type = 'I'` and Type_M is the indicator variable for `Type = 'M'`.

This model has a F-test p-value of approximately zero and a R^2 of 0.5328. All variables are significant at $\alpha = 0.05$ except for Type_M and LongestShell.

Additive Model

We use the `mgcv` package to fit an additive model. We model each covariate with splines, except for `Type` which we keep linear because it is categorical. This model explains 57.7% of the deviance. The indicator variable for `Type = 'I'` is significant, and all of the continuous covariates are approximately signficant.

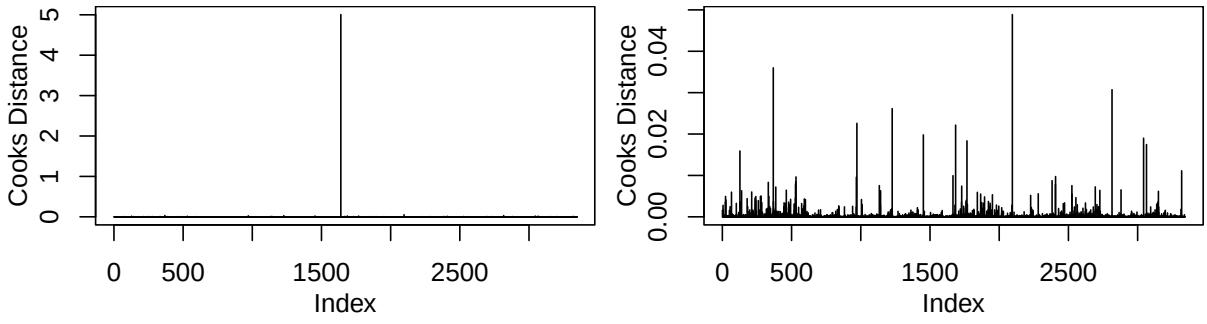
Random Forest

Finally, we use the `randomForest` package to fit a random forest model. This model explains 56.49% of the variance in the data.

Diagnostics and Model Selection

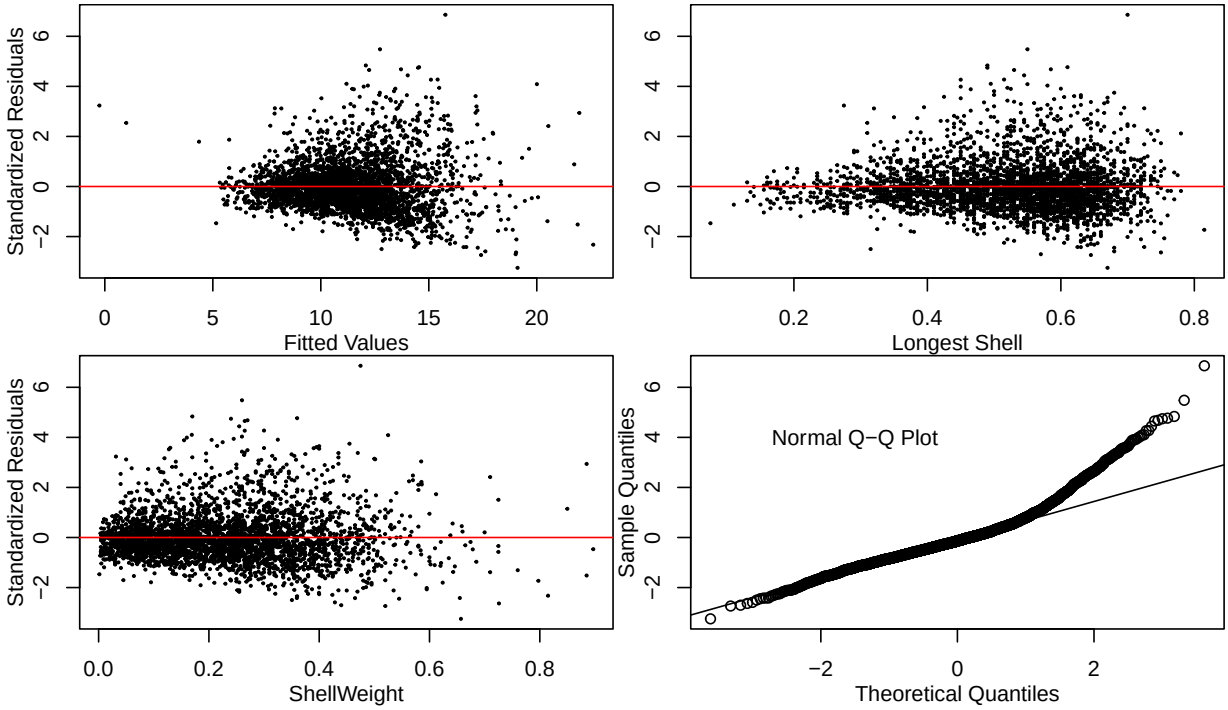
Linear Model Diagnostics

Outliers Looking at the Cooks Distances of the residuals for the linear model, we can see that there is an extreme outlier. We remove this observation from the training data, refit the model, and find another large outlier. These two observations are the ones pointed out previously which have extremely large values for `Height`. We decided to remove both of these points from the training set entirely, and refit all of the models. Below we show the Cooks Distances from the linear model before (left) and after (right) removing the two outliers. We can see that after removing these points there are no obvious outliers.



Interaction Terms Despite the high correlation among the covariates, adding interaction terms did not substantially improve the residuals, and only increased the R^2 by about 0.01, which did not seem worth it for the increase in model complexity. We tried a variety of different combinations of interaction terms, and always got roughly the same result.

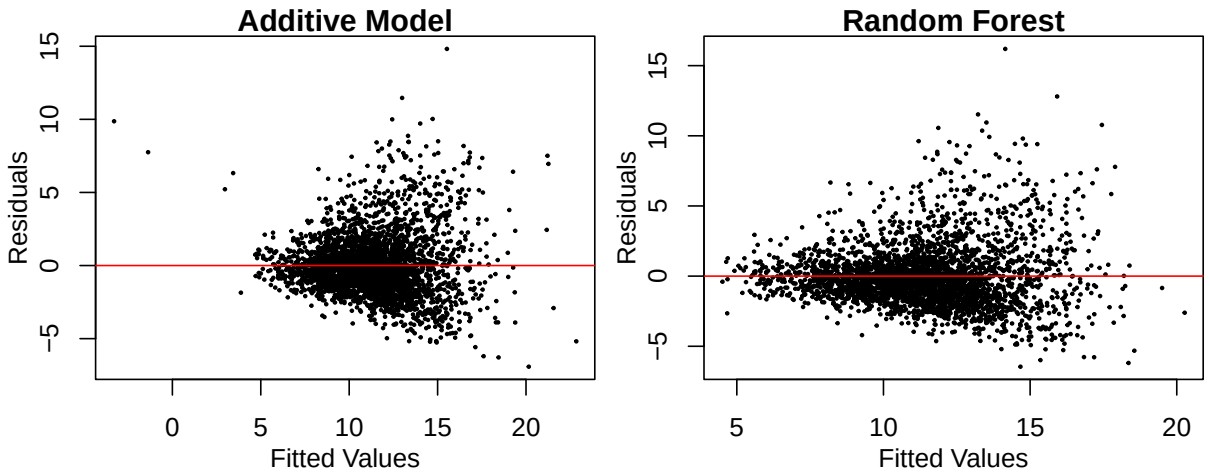
Transformations Despite the non-linear relationships between Age and the four weight-based covariates, the relationships between the residuals and the covariates are fairly linear. The only transformation that improved the residuals was applying the square root function to `ShellWeight`. Below are some of the residual diagnostic plots.



The main issues with the model are the non-constant variance of the residuals and the fact that the distribution of the residuals is not very normal, as shown in the Q-Q Plot. These violate the assumptions of Multiple Linear Regression, and must be taken into account when evaluating this model.

Additive Model and Random Forest

Beyond removing the two outliers, we did not discover any reason to modify the covariates for the additive model and random forest. Their residuals are shown below.



Model Evaluation

Now that we have our final models, we evaluate each of them on the held-out test set. The test set error and 95% confidence intervals for the true error are given below. We can see that the additive model and random forest have about the same error on the test set, while the linear model is slightly worse, although all three confidence intervals are overlapping.

Test Set Error	95% Confidence Interval
5.1874	(4.4616, 5.9132)
4.9934	(4.2439, 5.743)
4.9428	(4.1766, 5.709)

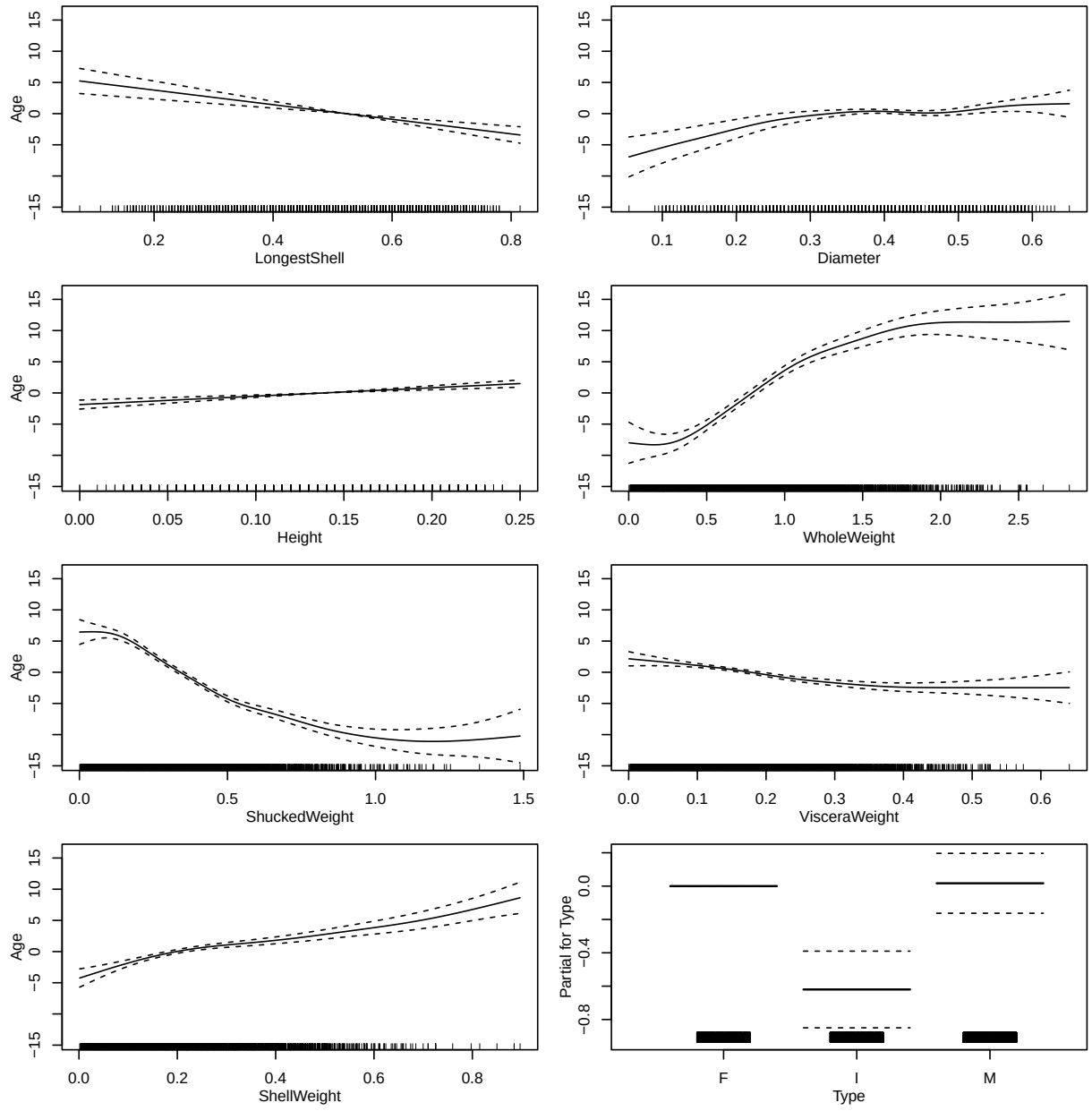
Final Model

We recommend using the additive model. The assumptions for the linear regression model were clearly violated, and had slightly worse error on the held-out test set. The additive model and the random forest had similar performance, but the additive model is more interpretable.

This additive model is of the form

$$\text{Age} = \alpha + m_1(\text{Type}_I) + m_2(\text{Type}_M) + m_3(\text{LongestShell}) + m_4(\text{Diameter}) + m_5(\text{Height}) + m_6(\text{WholeWeight}) +$$
$$m_7(\text{ShuckedWeight}) + m_8(\text{VisceraWeight}) + m_9(\text{ShellWeight}) + \epsilon$$

where m_1 is a linear model and m_2 through m_9 are spline-based models. Below we show each of the component models that make up the additive model.



Based on the plots it looks like Whole Weight, Shucked Weight, and Shell Weight are the most important features, although all variables have a non-zero relationship with Age. We can also see from the plot of `Type` that the age of Infants are generally lower than Males and Females, as we pointed out earlier. The coefficient for $m_1(\text{Type}_I)$ is significant, and the approximate p-values for the continuous covariates shows they are all significant.

Limitations

We were unable to construct a linear model that satisfied the assumptions of Multiple Linear Regression, so it may be wrong to consider the linear model at all. We were able to construct a better model however, so ultimately it was not an issue.

Code Appendix

```
## ---- echo=FALSE, include=FALSE-----------------------------------------------
## Import Packages
library(knitr)
library(mgcv)
library(randomForest)

## Load Data
df = read.csv("abalone.csv", head=TRUE)
df = df[,2:10]
df$Type = as.factor(df$Type)


## ----echo=FALSE---------------------------------------------------------------
# Introduction

## Create Table of Variable Descriptions
variables = names(df)
descriptions = c(
  "Sex: Male (M), Female (F), and Infant (I)",
  "Longest Shell Measurement",
  "Diameter",
  "Height",
  "Total Weight",
  "Shucked Weight",
  "Viscera Weight",
  "Shell Weight",
  "Abalone Age"
)
variable_info = data.frame("Variable Name"=variables, "Description"=descriptions,
                           check.names=FALSE)
kable(variable_info)


## ---- echo=FALSE, fig.dim=c(8,4.5)-------------------------------------------
# Exploratory Data Analysis

## Plot Age vs LongestShell, Diameter, Height, and WholeWeight
par(mfrow=c(2,2),oma = c(0, 1, 0, 0), mar = c(3, 2, 0, 1), mgp = c(2, 1, 0), xpd = NA)
plot(df$LongestShell, df$Age, pch=19, cex=0.25, xlab="Longest Shell", ylab="Age")
plot(df$Diameter, df$Age, ylab=NA, pch=19, cex=0.25, xlab="Diameter")
plot(df$Height, df$Age, pch=19, cex=0.25, xlab="Height", ylab="Age")
plot(df$WholeWeight, df$Age, ylab=NA, pch=19, cex=0.25, xlab="Whole Weight")


## ---- echo=FALSE, fig.dim=c(8,4.5)-------------------------------------------
## Plot Age vs. WholeWeight, ShuckedWeight, VisceraWeight, and ShellWeight
par(mfrow=c(2,2),oma = c(0, 1, 0, 0), mar = c(3, 2, 0, 1), mgp = c(2, 1, 0), xpd = NA)
plot(df$WholeWeight, df$Age, pch=19, cex=0.25, xlab="Whole Weight", ylab="Age")
plot(df$ShuckedWeight, df$Age, ylab=NA, pch=19, cex=0.25, xlab="Shucked Weight")
plot(df$VisceraWeight, df$Age, pch=19, cex=0.25, xlab="Viscera Weight", ylab="Age")
plot(df$ShellWeight, df$Age, ylab=NA, pch=19, cex=0.25, xlab="Shell Weight")
```

```r
## ---- echo=FALSE, fig.dim=c(8,3)-----------------------------------------------
## Covariate Correlations
corrs = cor(df[,2:9])

## Plot LongestShell vs. Diameter, and WholeWeight vs. ShuckedWeight
par(mfrow=c(1,2),oma = c(0, 1, 0, 0), mar = c(3, 2, 0, 1.5), mgp = c(2, 1, 0), xpd = NA)
plot(df$LongestShell, df$Diameter, pch=19, cex=0.25,
     xlab="Longest Shell", ylab="Diameter")
text(0.3, 0.6, "Correlation = 0.9877")
plot(df$WholeWeight, df$ShuckedWeight, pch=19, cex=0.25,
     xlab="Whole Weight", ylab="Shucked Weight")
text(0.8, 1.365, "Correlation = 0.9689")


## ----echo=FALSE, out.width="80%", fig.align='center'---------------------------
## Conditional distribution of Age given Type
male_density = density(df$Age[df$Type == 'M'])
female_density = density(df$Age[df$Type == 'F'])
infant_density = density(df$Age[df$Type == 'I'])

plot(male_density, col = 'blue',
     ylim=c(0,max(male_density$y, female_density$y, infant_density$y)),
     xlab="Age", ylab="Density", main=NA)
lines(female_density, col='red')
lines(infant_density, col='green')
legend('topright', c('Male', 'Female', 'Infant'), lty = c(1,1),
       col = c('blue', 'red', 'green'))


## ----echo=FALSE----------------------------------------------------------------
## Train-Test Splits
set.seed(36401)
split_indices = sample.int(nrow(df), size=835)
train = df[-split_indices, ]
test = df[split_indices, ]


## ----echo=FALSE----------------------------------------------------------------
# Modeling

## Baseline Linear Regression Model
out = lm(Age ~ ., data=train)


## ----echo=FALSE----------------------------------------------------------------
## Additive Model
out_add = gam(Age ~ Type + s(LongestShell) + s(Diameter) + s(Height) +
                s(WholeWeight) +  s(ShuckedWeight) + s(VisceraWeight) +
                s(ShellWeight), data=train)


## ----echo=FALSE----------------------------------------------------------------
## Random Forest
```

```
forest_out = randomForest(Age ~ ., importance = TRUE, data = train, ntree = 500)


## ---- echo=FALSE, fig.dim=c(8,2)-----------------------------------------------
# Diagnostics and Model Selection

## Plot Cooks Distance Before Removing Outlier
par(mfrow=c(1,2),oma = c(0, 1, 0, 0), mar = c(3, 2, 0, 1.5), mgp = c(2, 1, 0), xpd = NA)
plot(cooks.distance(out), type='h', ylab="Cooks Distance")

## Remove Outlier and Refit Twice
train = train[-which.max(cooks.distance(out)),]
out = lm(Age ~ ., data=train)
train = train[-which.max(cooks.distance(out)),]
out = lm(Age ~ ., data=train)

## Also Refit Additive and Random Forest
out_add = gam(Age ~ Type + s(LongestShell) + s(Diameter) + s(Height) +
                s(WholeWeight) +  s(ShuckedWeight) + s(VisceraWeight) +
                s(ShellWeight), data=train)
out_forest = randomForest(Age ~ ., importance = TRUE, data = train, ntree = 500)

## Plot Cooks Distance After Removing Outliers
plot(cooks.distance(out), type='h', ylab="Cooks Distance")


## ---- echo=FALSE, fig.dim=c(8,4.5)---------------------------------------------
## Transform ShellWeight
out2 = lm(Age ~ Type + LongestShell + Diameter + Height + WholeWeight +
           ShuckedWeight + VisceraWeight + I(sqrt(ShellWeight)) , data=train)
## Plot Diagnostics
par(mfrow=c(2,2),oma = c(0, 1, 0, 0), mar = c(3, 2, 0, 1), mgp = c(2, 1, 0), xpd = NA)
plot(fitted(out2), rstudent(out2), pch=19, cex=0.25,
     xlab="Fitted Values", ylab="Standardized Residuals")
abline(h=0, col='red', xpd=FALSE)
plot(train$LongestShell, rstudent(out2), ylab=NA, pch=19, cex=0.25,
     xlab="Longest Shell")
abline(h=0, col='red', xpd=FALSE)
plot(train$ShellWeight, rstudent(out2), pch=19, cex=0.25,
     xlab="ShellWeight", ylab="Standardized Residuals")
abline(h=0, col='red', xpd=FALSE)
qqnorm(rstudent(out2), main=NA)
qqline(rstudent(out2), xpd=FALSE)
text(-1.5, 4, "Normal Q-Q Plot")


## ---- echo=FALSE, fig.dim=c(8,3)-----------------------------------------------
## Plot Additive Model Residuals
par(mfrow=c(1,2),oma = c(0, 1, 0, 0), mar = c(3, 2, 1, 1.5), mgp = c(2, 1, 0), xpd = NA)
plot(fitted(out_add), residuals(out_add), pch=19, cex=0.25, main="Additive Model",
     xlab="Fitted Values", ylab="Residuals")
abline(h=0, col='red', xpd=FALSE)
## Plot Random Forest Residuals
```

```r
plot(out_forest$predicted, train$Age - out_forest$predicted, pch=19, cex=0.25,
     main="Random Forest", xlab="Fitted Values", ylab="Residuals")
abline(h=0, col='red', xpd=FALSE)


## ----echo=FALSE-----------------------------------------------------------
## Get Test Set Predictions
test_X = test[,1:8]
test_y = test$Age
preds_linear = predict(out2, test_X)
preds_add = predict(out_add, test_X)
preds_forest = predict(out_forest, test_X)

Error_Conf = function(y_hat, y) {
  b_ijs = (y - y_hat)^2
  r_hat = mean(b_ijs)
  se = sqrt(sum((b_ijs - r_hat)^2) / (length(y)-1)) / sqrt(length(y))
  z = -qnorm(0.05/2)

  lower = round(r_hat - z*se, digits=4)
  upper = round(r_hat + z*se, digits=4)
  return(c(round(r_hat, digits=4), paste0('(',lower,', ',upper,')')))
}

## Construct Error Confidence Intervals
error_linear = Error_Conf(preds_linear, test_y)
error_add = Error_Conf(preds_add, test_y)
error_forest = Error_Conf(preds_forest, test_y)

## Display in Table
error_info = data.frame(
  "Test Set Error"=c(error_linear[1], error_add[1], error_forest[1]),
  "95% Confidence Interval"=c(error_linear[2], error_add[2], error_forest[2]),
  check.names=FALSE
)
kable(error_info)


## ---- echo=FALSE, , fig.dim=c(8,8)----------------------------------------
# Final Model

## Plot Sub-models of Additive Model
par(mfrow=c(4,2),oma = c(0, 1, 0, 0), mar = c(4, 3, 0, 1), mgp = c(2, 1, 0), xpd = NA)
plot(out_add, all.terms=TRUE, select=1, xlab="LongestShell", ylab="Age")
plot(out_add, all.terms=TRUE, select=2, xlab="Diameter", ylab=NA)
plot(out_add, all.terms=TRUE, select=3, xlab="Height", ylab="Age")
plot(out_add, all.terms=TRUE, select=4, xlab="WholeWeight", ylab=NA)
plot(out_add, all.terms=TRUE, select=5, xlab="ShuckedWeight", ylab="Age")
plot(out_add, all.terms=TRUE, select=6, xlab="VisceraWeight", ylab=NA)
plot(out_add, all.terms=TRUE, select=7, xlab="ShellWeight", ylab="Age")
plot(out_add, all.terms=TRUE, select=8, xlab="Type", ylab=NA)
```

```
## ----code = readLines(knitr::purl(knitr::current_input(), documentation = 1)), echo = T, eval = F----
## NA
```