Physical Measures Diagnostics

Katie, Rita, and Chang

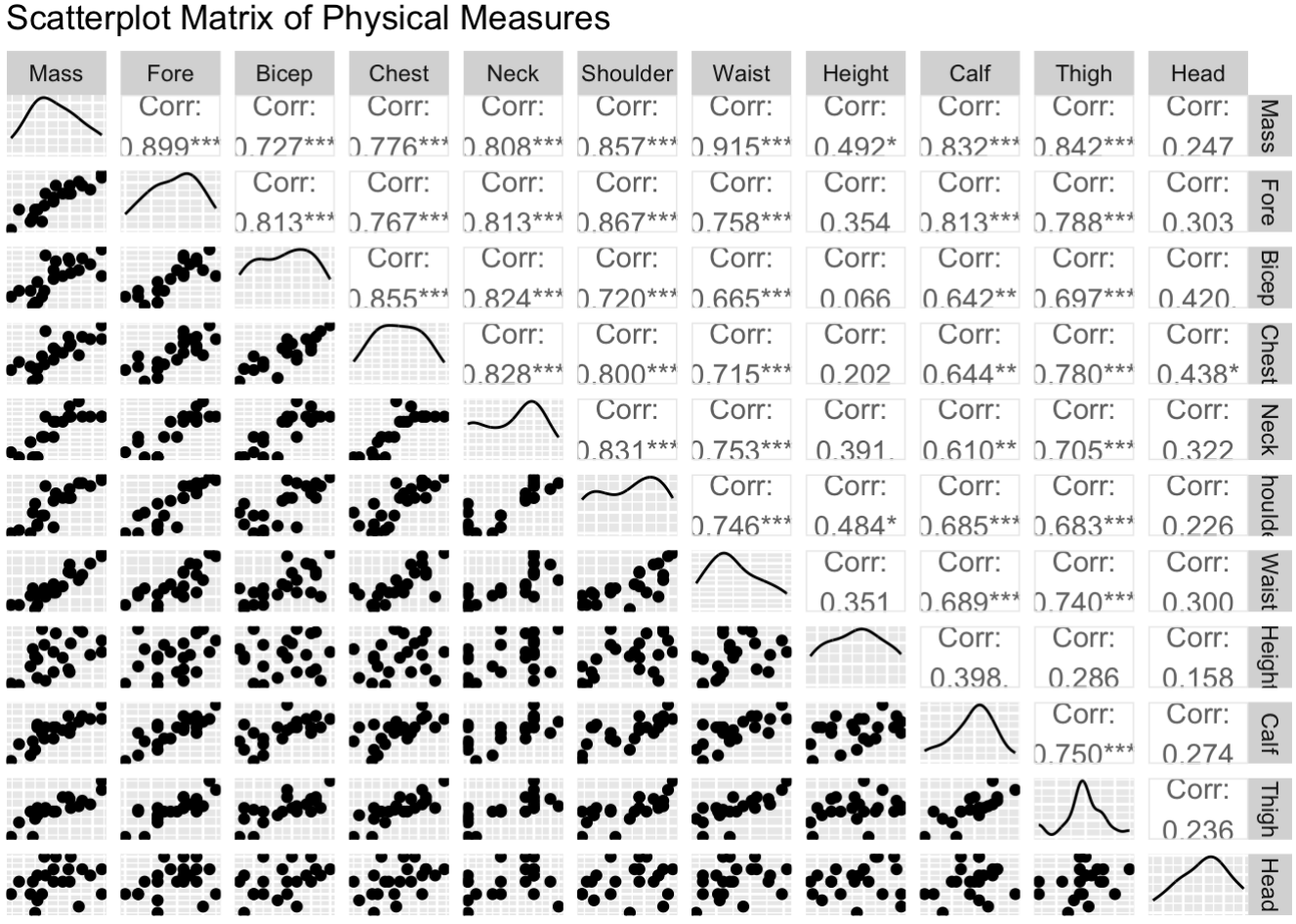
2023-10-16

Scatterplot Matrix

```
library(GGally)
ggpairs(PhysicalMeasures, axisLabels = "none",
        title = "Scatterplot Matrix of Physical Measures")
```

```
# corr codes
```

Scatterplot Matrix



Stepwise Regression

```
library("MASS")
full.phys <- lm(Mass ~ ., data = PhysicalMeasures)
step.phys <- stepAIC(full.phys, direction = "both",
                     trace = FALSE)
summary(step.phys)
```

```
##
## Call:
## lm(formula = Mass ~ Fore + Waist + Height + Calf + Thigh + Head,
##     data = PhysicalMeasures)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -3.2362 -1.3426 -0.0132  0.9784  4.5197
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -79.72624   23.88925  -3.337  0.00450 **
## Fore          1.79485    0.48536   3.698  0.00215 **
## Waist         0.65671    0.09719   6.757 6.45e-06 ***
## Height        0.25388    0.08059   3.150  0.00661 **
## Calf          0.50718    0.34671   1.463  0.16415
## Thigh         0.43298    0.22801   1.899  0.07698 .
## Head         -0.65722    0.38200  -1.720  0.10590
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.077 on 15 degrees of freedom
## Multiple R-squared:  0.9744, Adjusted R-squared:  0.9641
## F-statistic:    95 on 6 and 15 DF,  p-value: 4.501e-11
```

Multiple regression with selected model (output of stepwise regression)

```
stepmodel <- lm(formula = Mass ~ Fore + Waist + Height + Calf + Thigh + Head, data = Phy
sicalMeasures)
summary(stepmodel)
```

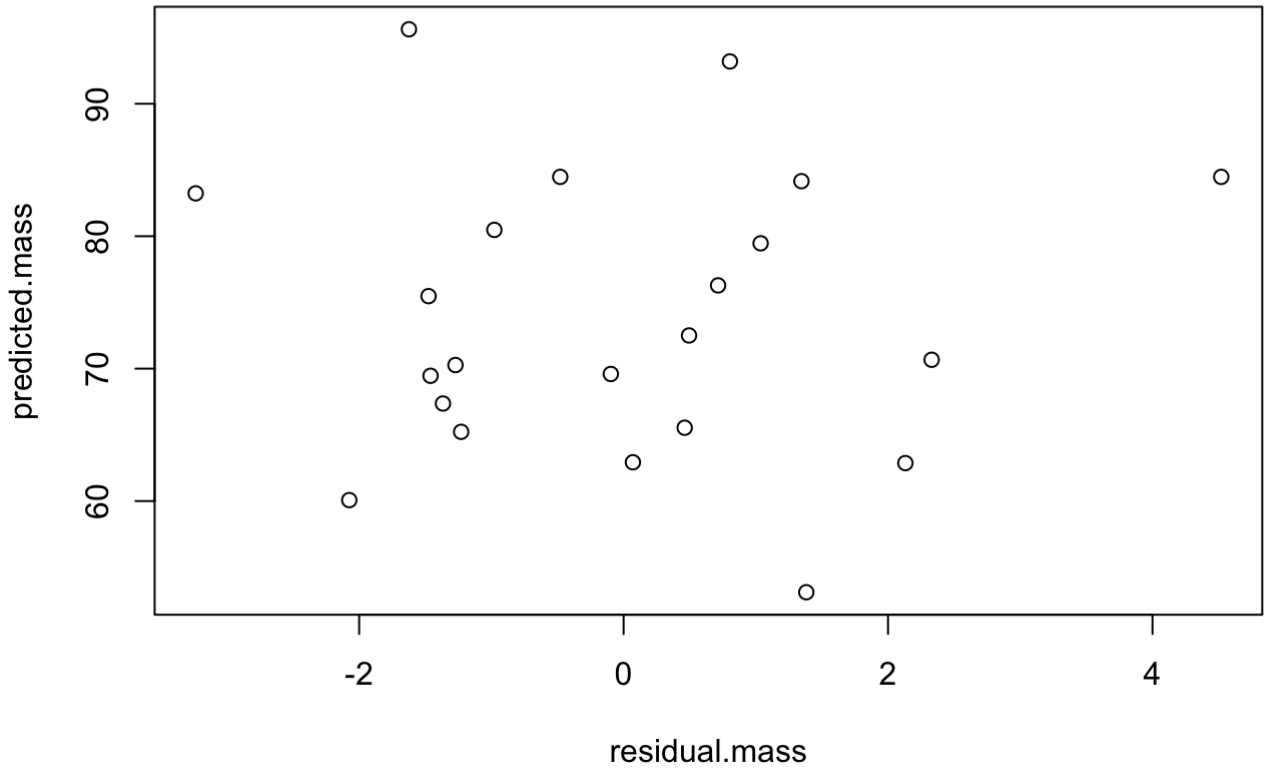
```
##
## Call:
## lm(formula = Mass ~ Fore + Waist + Height + Calf + Thigh + Head,
##     data = PhysicalMeasures)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -3.2362 -1.3426 -0.0132  0.9784  4.5197
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -79.72624   23.88925  -3.337  0.00450 **
## Fore          1.79485    0.48536   3.698  0.00215 **
## Waist         0.65671    0.09719   6.757 6.45e-06 ***
## Height        0.25388    0.08059   3.150  0.00661 **
## Calf          0.50718    0.34671   1.463  0.16415
## Thigh         0.43298    0.22801   1.899  0.07698 .
## Head         -0.65722    0.38200  -1.720  0.10590
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.077 on 15 degrees of freedom
## Multiple R-squared:  0.9744, Adjusted R-squared:  0.9641
## F-statistic:    95 on 6 and 15 DF,  p-value: 4.501e-11
```

Diagnostics

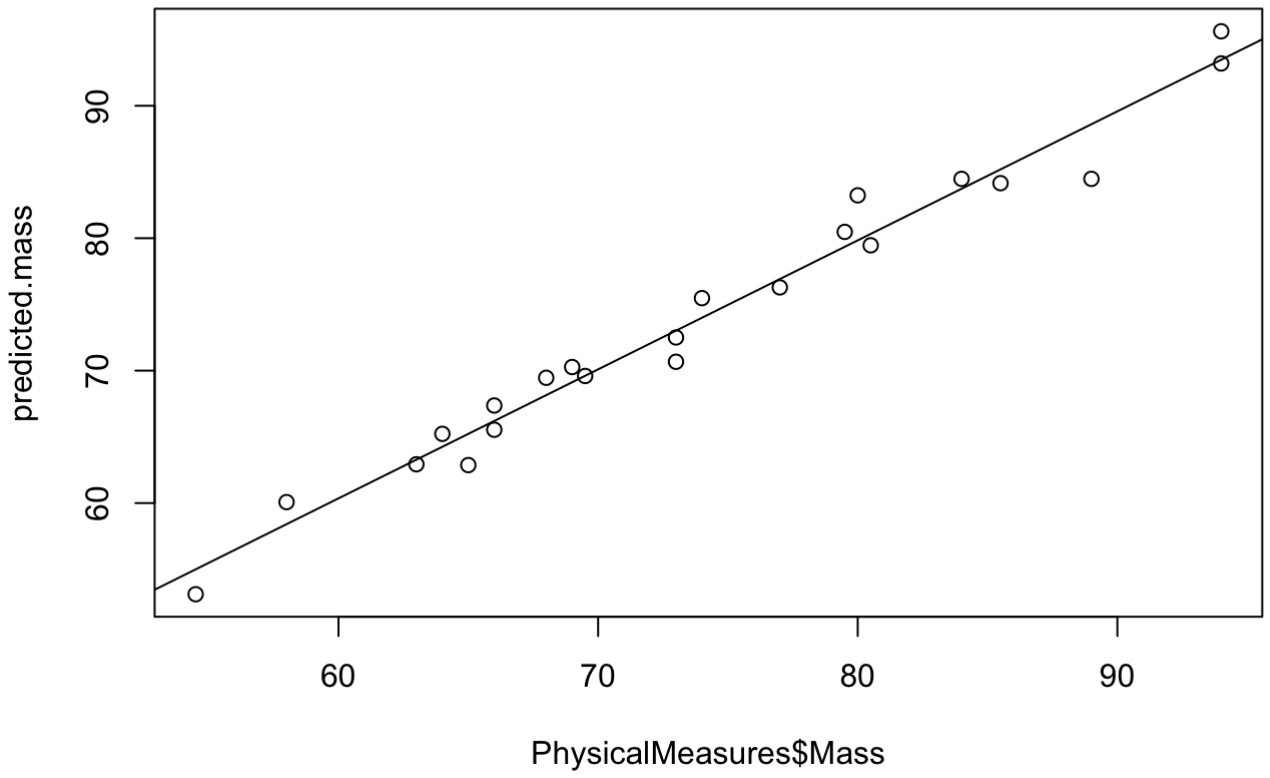
```
library("broom")
diagnostics <- augment(stepmodel)
```

Plots

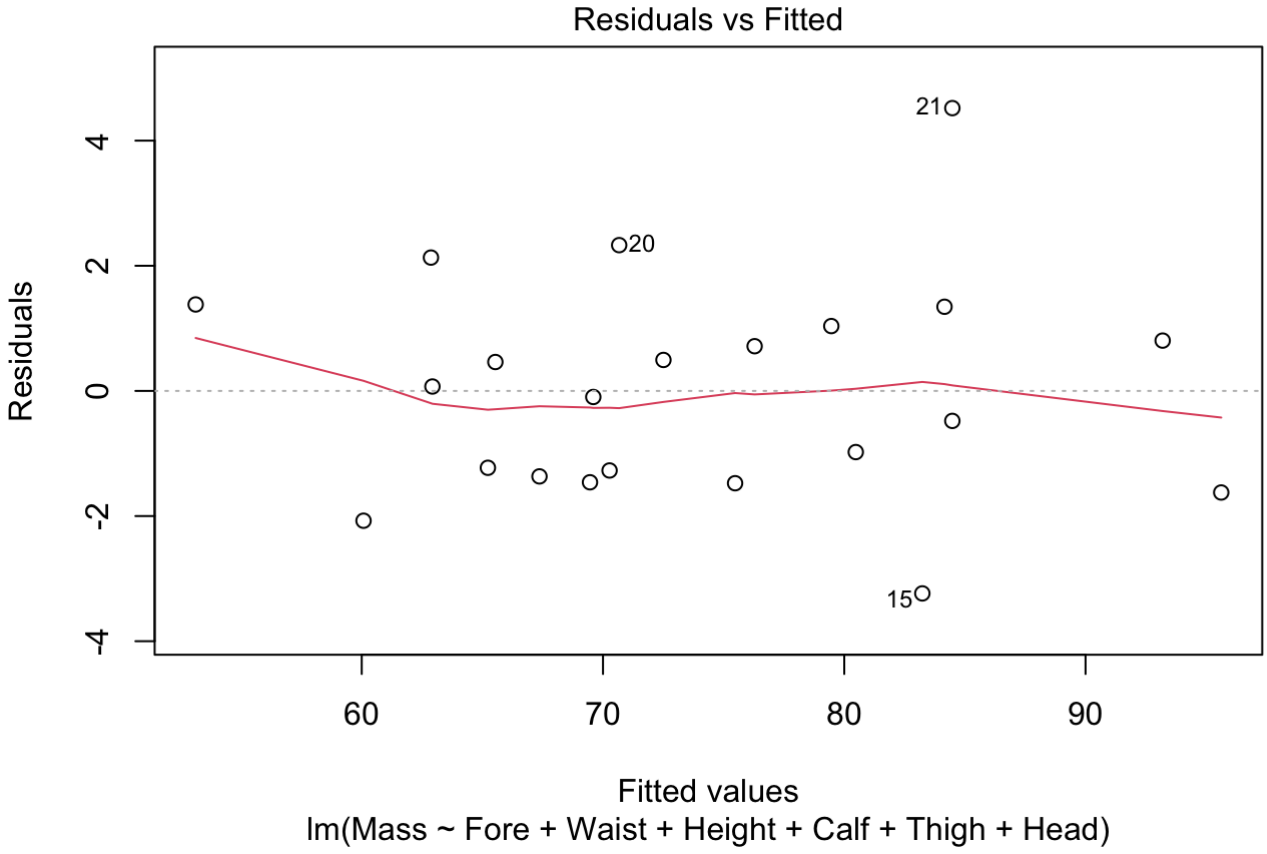
```
residual.mass = residuals(stepmodel)
predicted.mass = predict(stepmodel)
plot(residual.mass, predicted.mass)
```

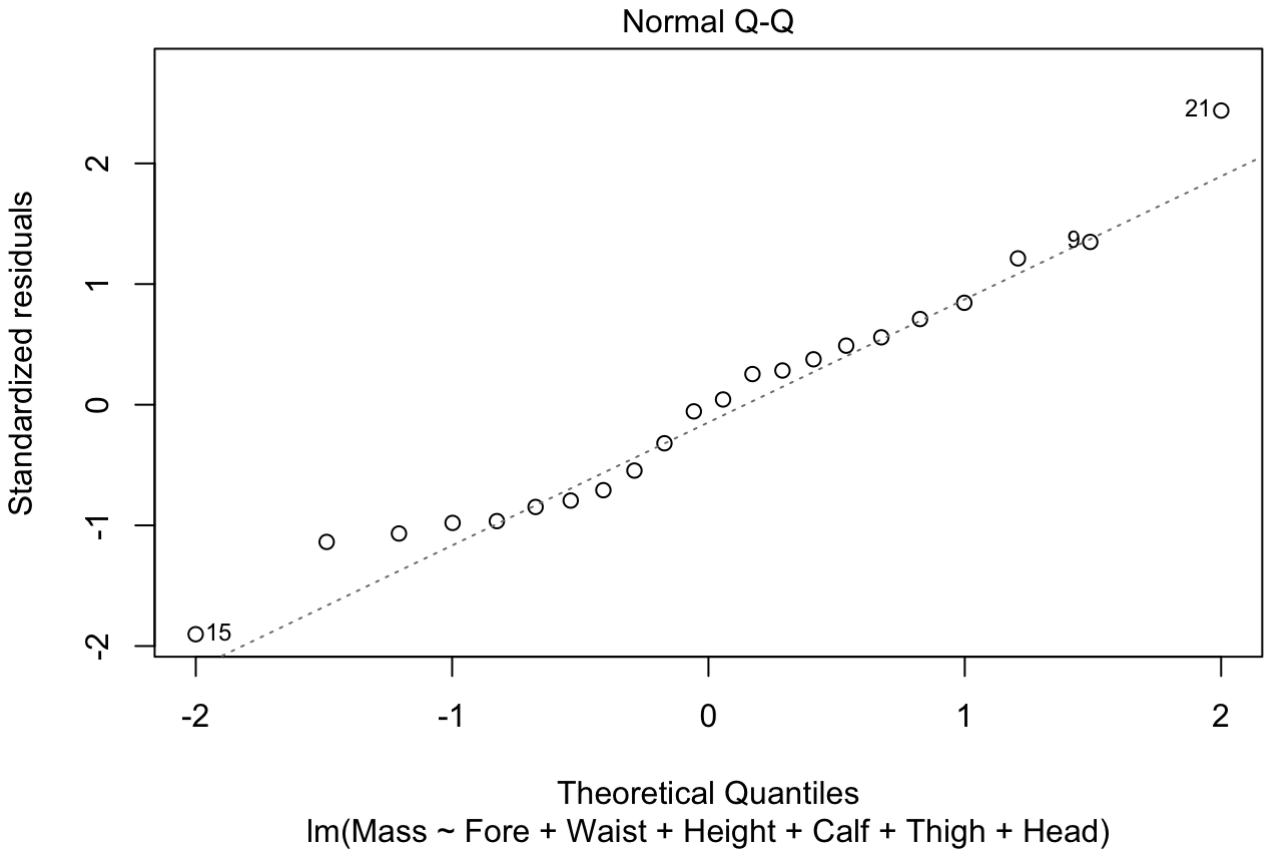


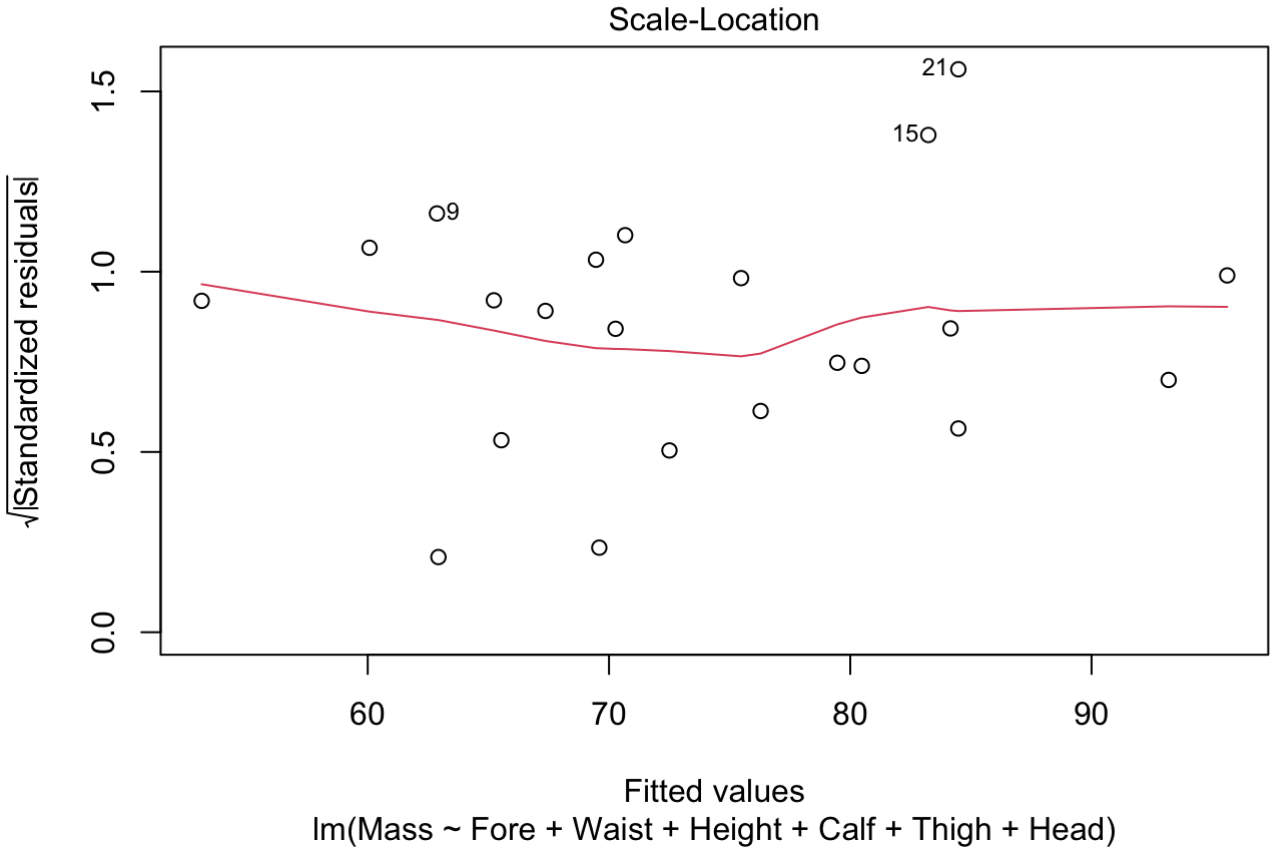

```
plot(PhysicalMeasures$Mass, predicted.mass)
reg = lm(predicted.mass ~ PhysicalMeasures$Mass)
abline(reg)
```

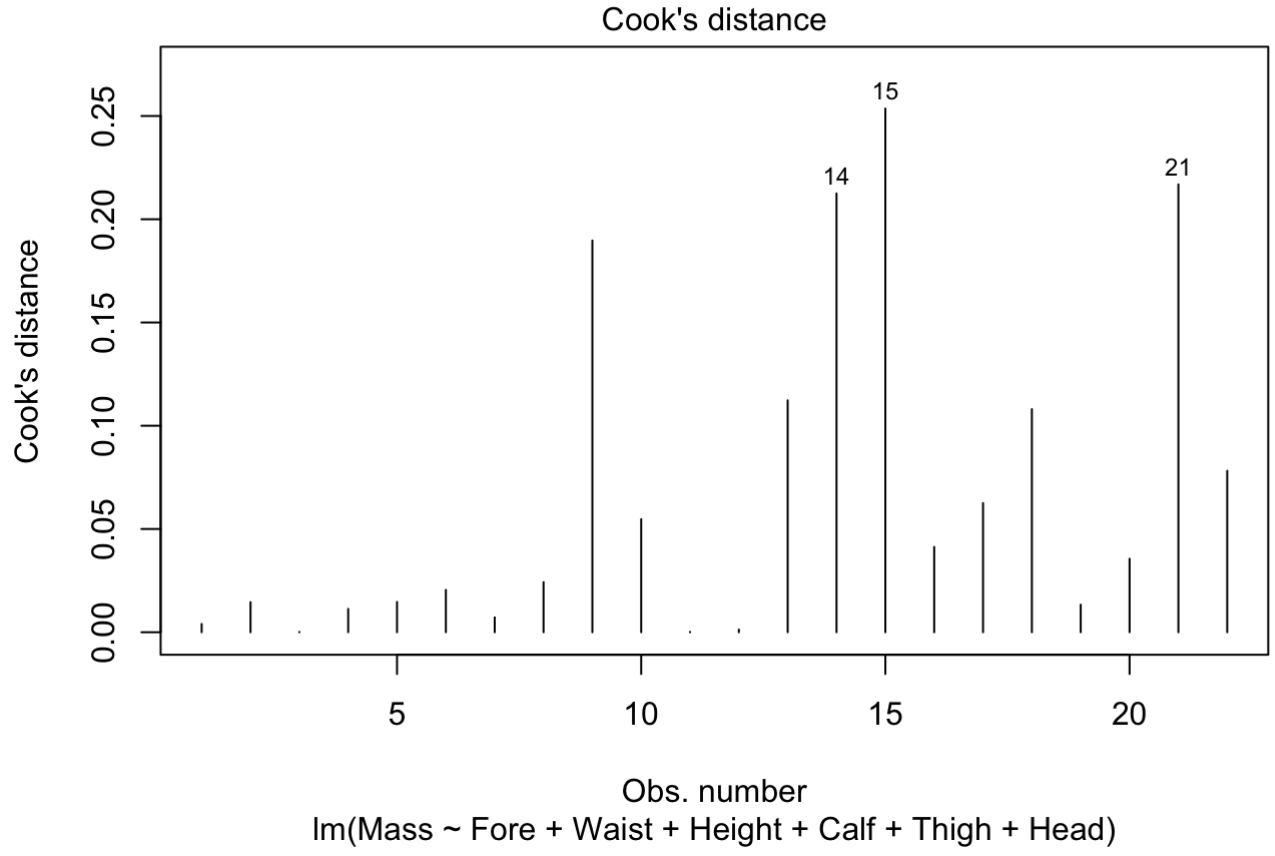


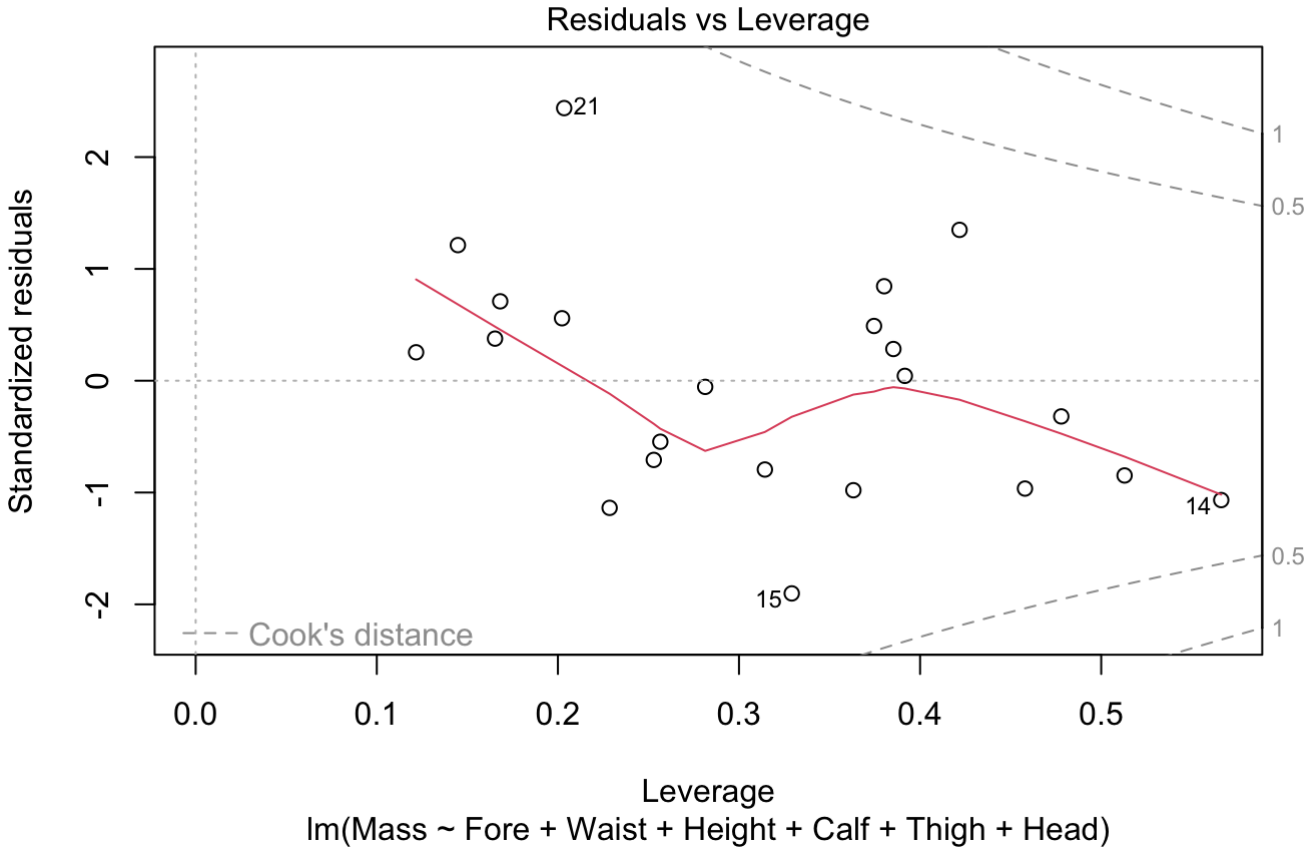

```
plot(stepmodel, which = c(1,2,3,4,5,6))
```

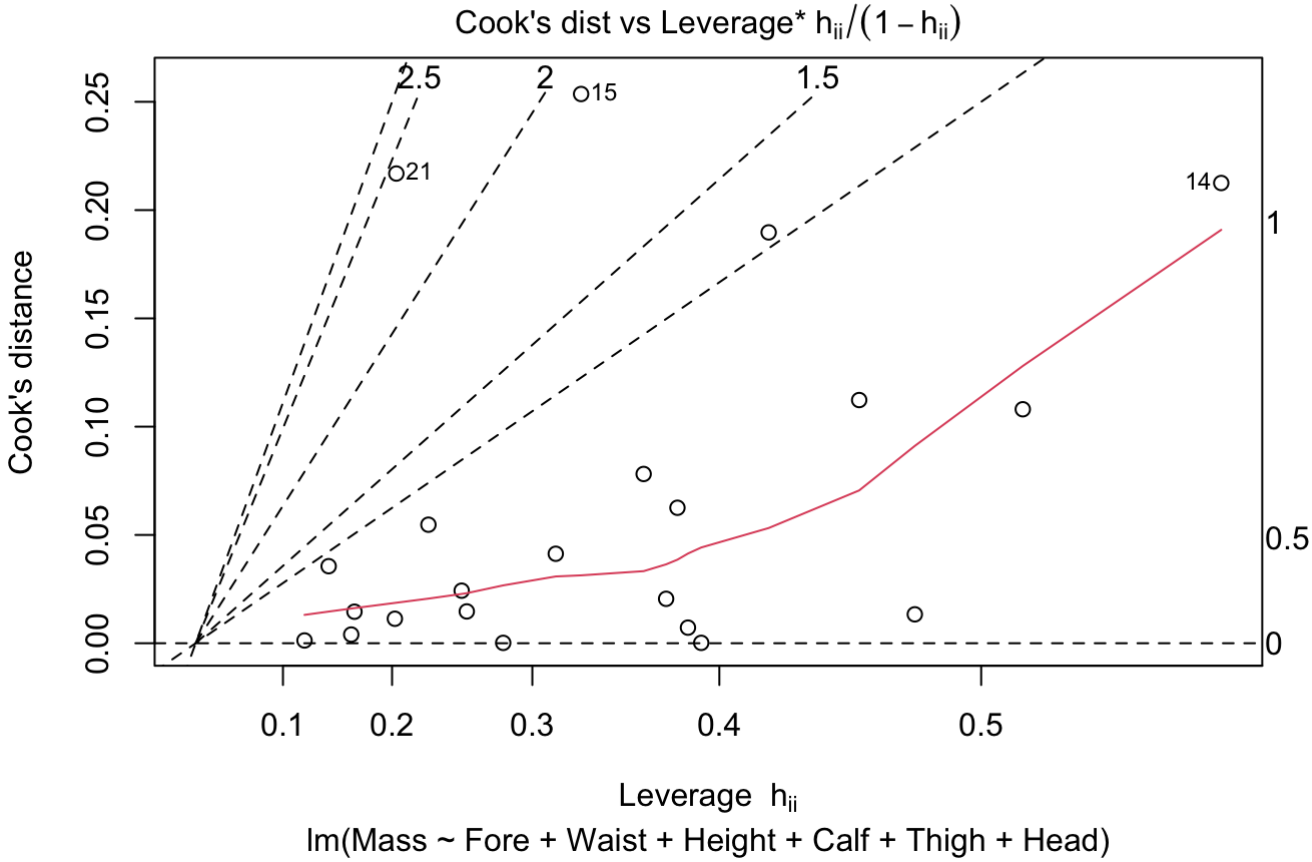

Residuals vs Fitted
Residuals
Fitted values
lm(Mass ~ Fore + Waist + Height + Calf + Thigh + Head)
21
20
15
4
2
0
-2
-4
60
70
80
90


Normal Q-Q
Standardized residuals
Theoretical Quantiles
lm(Mass ~ Fore + Waist + Height + Calf + Thigh + Head)
21
9
15
2
1
0
-1
-2


Scale-Location
√|Standardized residuals|
0.0
0.5
1.0
1.5
21
15
9
60
70
80
90
Fitted values
lm(Mass ~ Fore + Waist + Height + Calf + Thigh + Head)

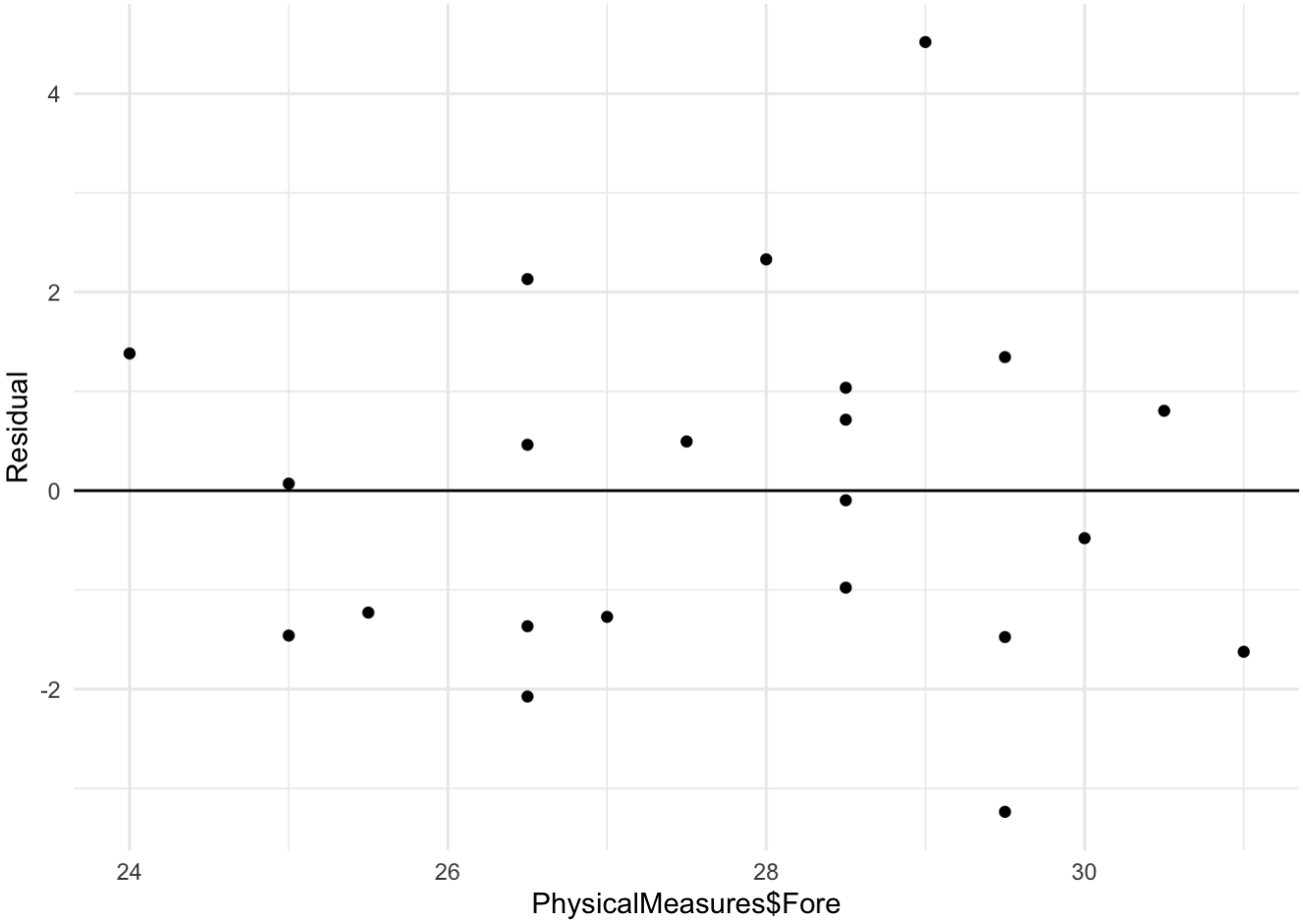

Cook's distance
Cook's distance
0.00
0.05
0.10
0.15
0.20
0.25
15
14
21
5
10
15
20
Obs. number
lm(Mass ~ Fore + Waist + Height + Calf + Thigh + Head)


Residuals vs Leverage
Standardized residuals
Leverage
lm(Mass ~ Fore + Waist + Height + Calf + Thigh + Head)
Cook's distance
21
14
15
1
0.5
0.5
1


Cook's dist vs Leverage* $h_{ii}/(1-h_{ii})$
Cook's distance
Leverage h_{ii}
lm(Mass ~ Fore + Waist + Height + Calf + Thigh + Head)
2.5
2
1.5
15
21
14
1
0.5
0

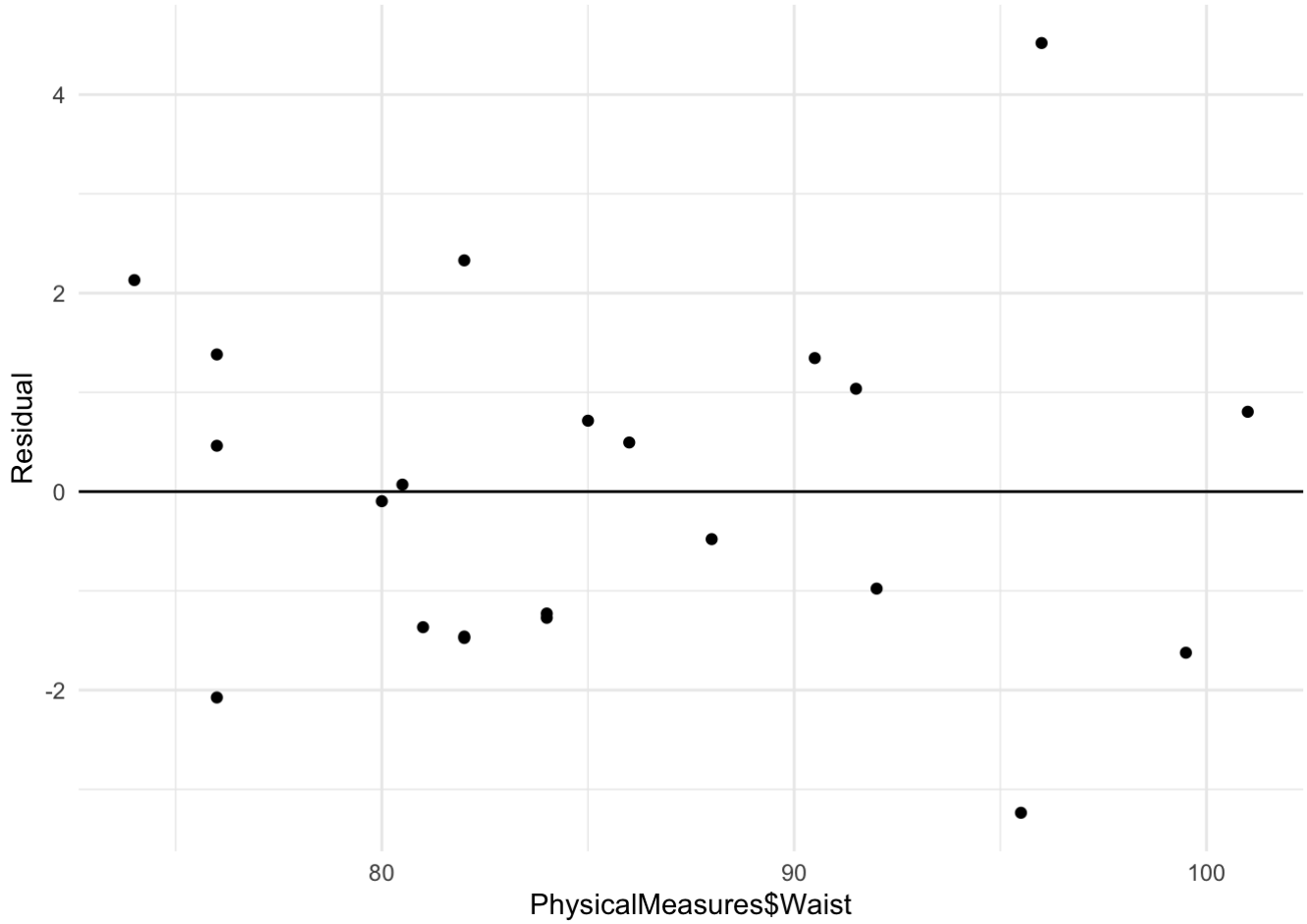
Residuals by Fore

```
library("GGally")
ggplot(diagnostics) +
  geom_point(aes(x = PhysicalMeasures$Fore, y = .resid)) +
  geom_hline(yintercept = 0) +
  ylab("Residual") +
  theme_minimal()
```



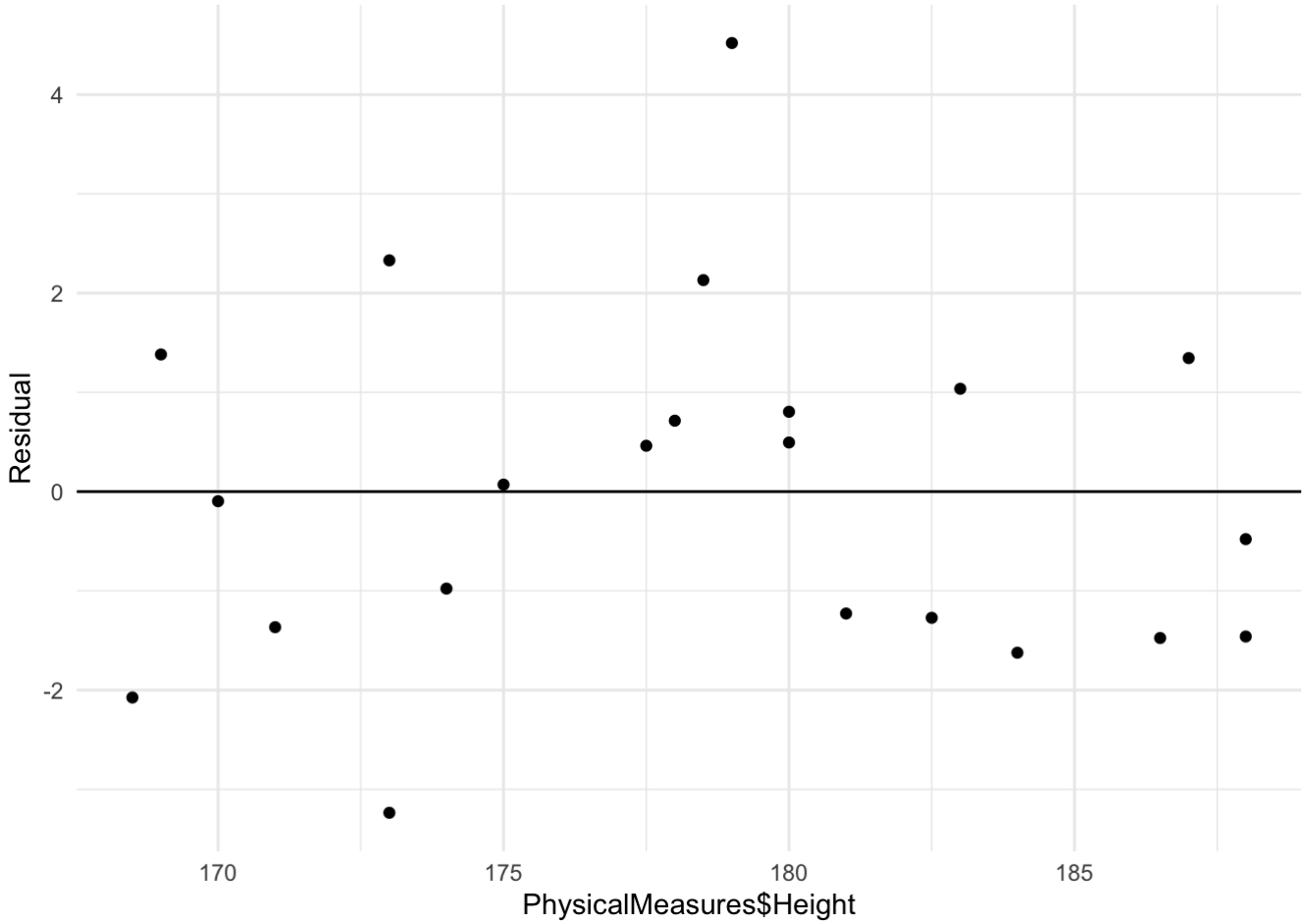
Residuals by Waist

```
library("GGally")
ggplot(diagnostics) +
  geom_point(aes(x = PhysicalMeasures$Waist, y = .resid)) +
  geom_hline(yintercept = 0) +
  ylab("Residual") +
  theme_minimal()
```



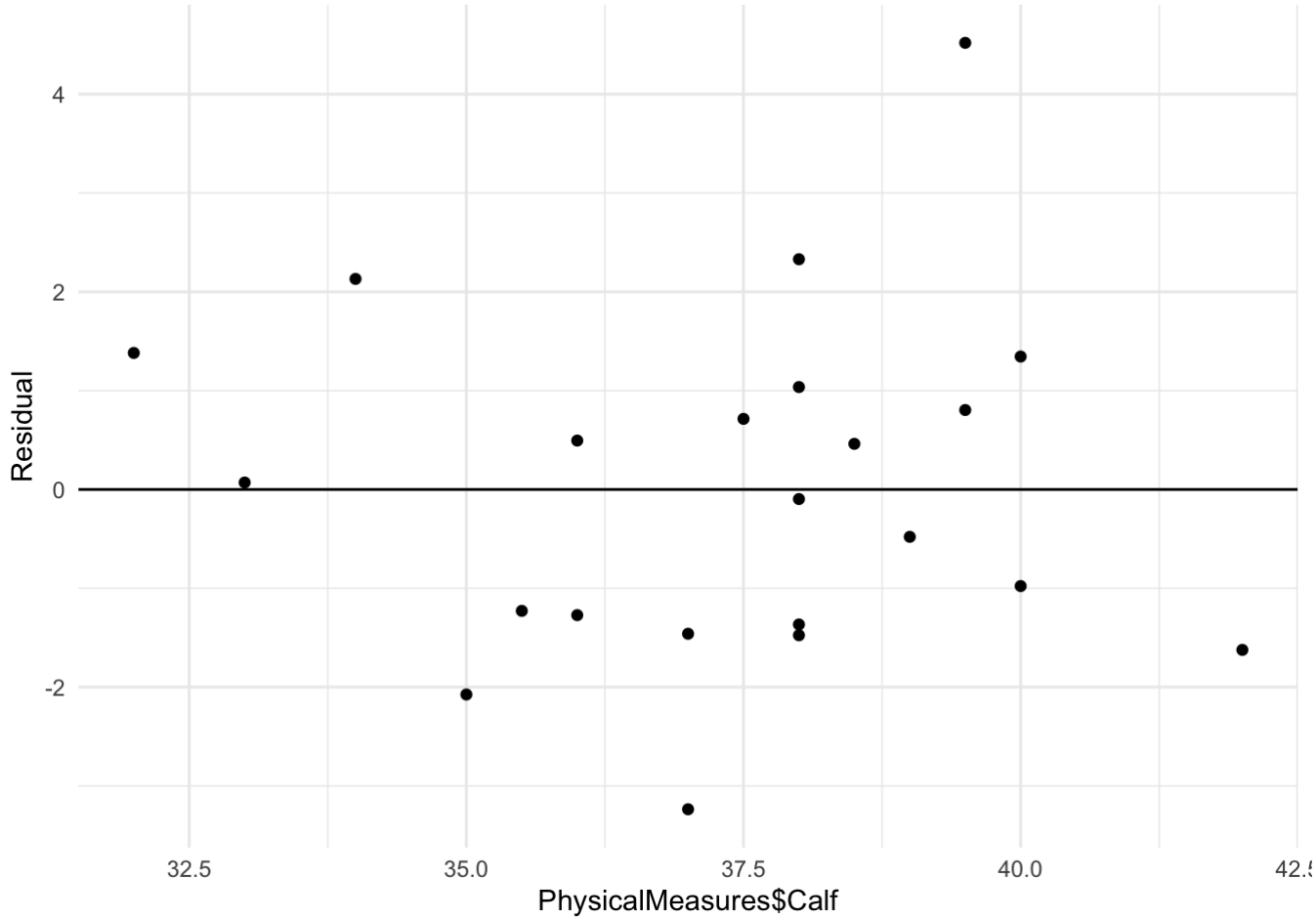
Residuals by Height

```
library("GGally")
ggplot(diagnostics) +
  geom_point(aes(x = PhysicalMeasures$Height, y = .resid)) +
  geom_hline(yintercept = 0) +
  ylab("Residual") +
  theme_minimal()
```



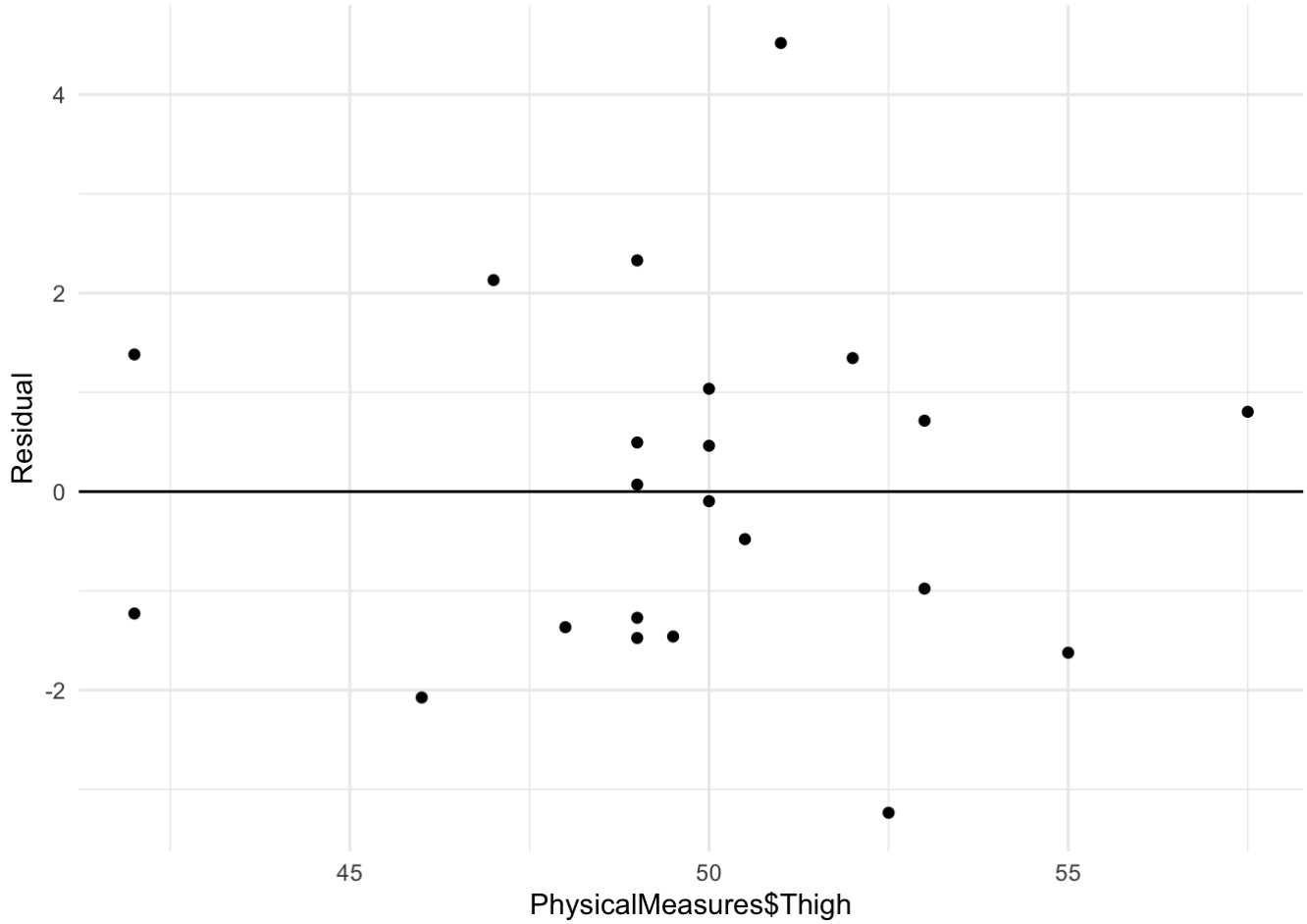
Residuals by Calf

```
library("GGally")
ggplot(diagnostics) +
  geom_point(aes(x = PhysicalMeasures$Calf, y = .resid)) +
  geom_hline(yintercept = 0) +
  ylab("Residual") +
  theme_minimal()
```



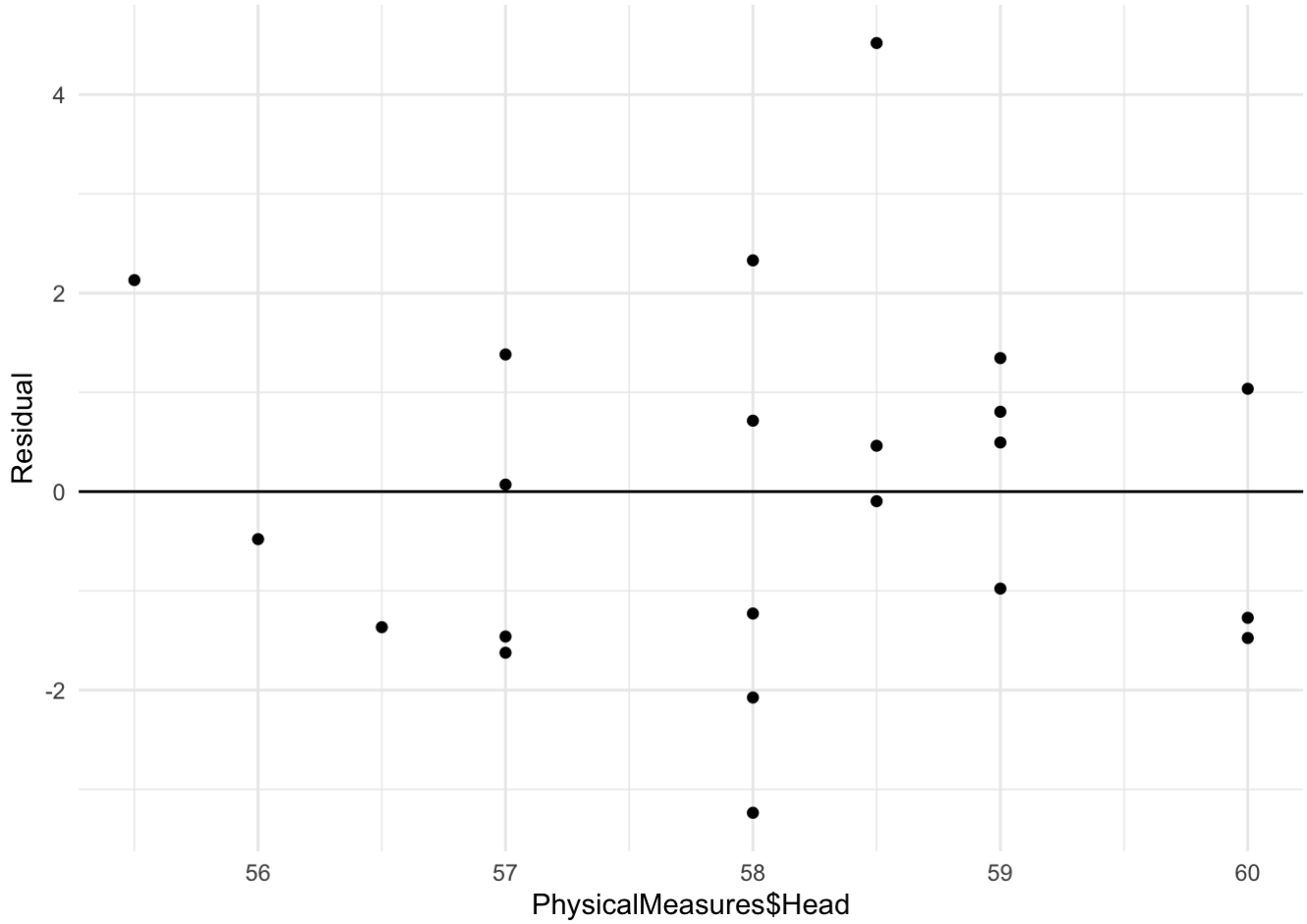
Residuals by Thigh

```
library("GGally")
ggplot(diagnostics) +
  geom_point(aes(x = PhysicalMeasures$Thigh, y = .resid)) +
  geom_hline(yintercept = 0) +
  ylab("Residual") +
  theme_minimal()
```



Residuals by Head

```
library("GGally")
ggplot(diagnostics) +
  geom_point(aes(x = PhysicalMeasures$Head, y = .resid)) +
  geom_hline(yintercept = 0) +
  ylab("Residual") +
  theme_minimal()
```



Ridge Regression

```
library("glmnet")
```

```
## Loading required package: Matrix
```

```
## Loaded glmnet 4.1-4
```

```
x = data.matrix(PhysicalMeasures [,c('Fore', 'Bicep', 'Chest', 'Neck', 'Shoulder', 'Wais
t', 'Height', 'Calf', 'Thigh', 'Head')])
y = PhysicalMeasures$Mass
model <- glmnet(x, y, alpha = 0)
summary(model)
```

```
##           Length Class     Mode
## a0         100   -none-    numeric
## beta      1000   dgCMatrix S4
## df         100   -none-    numeric
## dim          2   -none-    numeric
## lambda     100   -none-    numeric
## dev.ratio  100   -none-    numeric
## nulldev      1   -none-    numeric
## npasses      1   -none-    numeric
## jerr         1   -none-    numeric
## offset       1   -none-    logical
## call         4   -none-    call
## nobs         1   -none-    numeric
```

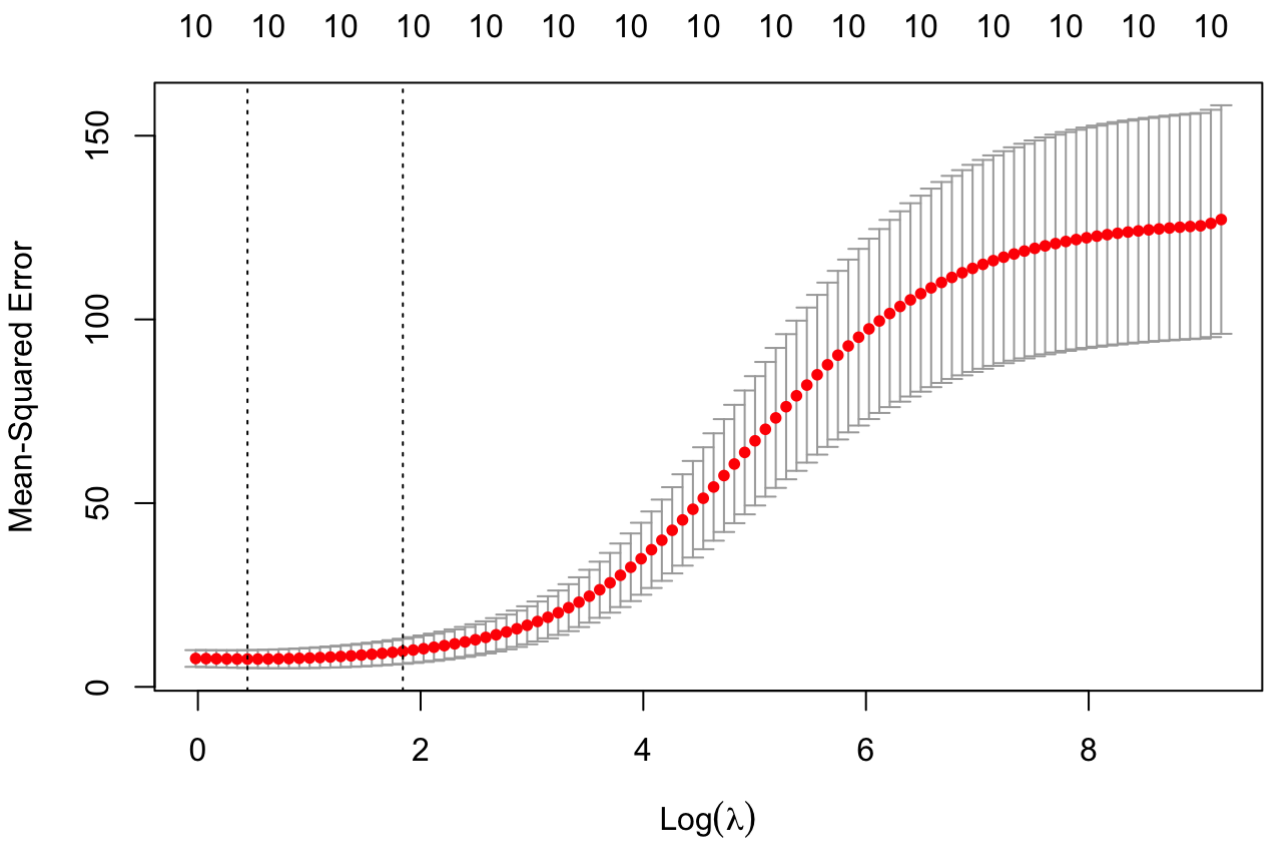
```
cv_model <- cv.glmnet(x, y, alpha = 0)
```

```
## Warning: Option grouped=FALSE enforced in cv.glmnet, since < 3 observations per
## fold
```

```
best_lambda <- cv_model$lambda.min
best_lambda
```

```
## [1] 1.560719
```

```
plot(cv_model)
```




```
best_model <- glmnet(x, y, alpha = 0, lambda = best_lambda)
coef(best_model)
```

```
## 11 x 1 sparse Matrix of class "dgCMatrix"
##                       s0
## (Intercept) -88.43738949
## Fore          1.03103730
## Bicep         0.13981215
## Chest         0.08261911
## Neck          0.16426946
## Shoulder      0.15908660
## Waist         0.48072832
## Height        0.23071563
## Calf          0.67548313
## Thigh         0.43966745
## Head         -0.55973427
```

Lasso Regression

```
library("glmnet")
x = data.matrix(PhysicalMeasures [,c('Fore', 'Bicep', 'Chest', 'Neck', 'Shoulder', 'Wais
t', 'Height', 'Calf', 'Thigh', 'Head')])
y = PhysicalMeasures$Mass
model <- glmnet(x, y, alpha = 1)
summary(model)
```

```
##           Length Class     Mode
## a0         75    -none-    numeric
## beta      750    dgCMatrix S4
## df         75    -none-    numeric
## dim         2    -none-    numeric
## lambda     75    -none-    numeric
## dev.ratio  75    -none-    numeric
## nulldev     1    -none-    numeric
## npasses     1    -none-    numeric
## jerr        1    -none-    numeric
## offset      1    -none-    logical
## call        4    -none-    call
## nobs        1    -none-    numeric
```

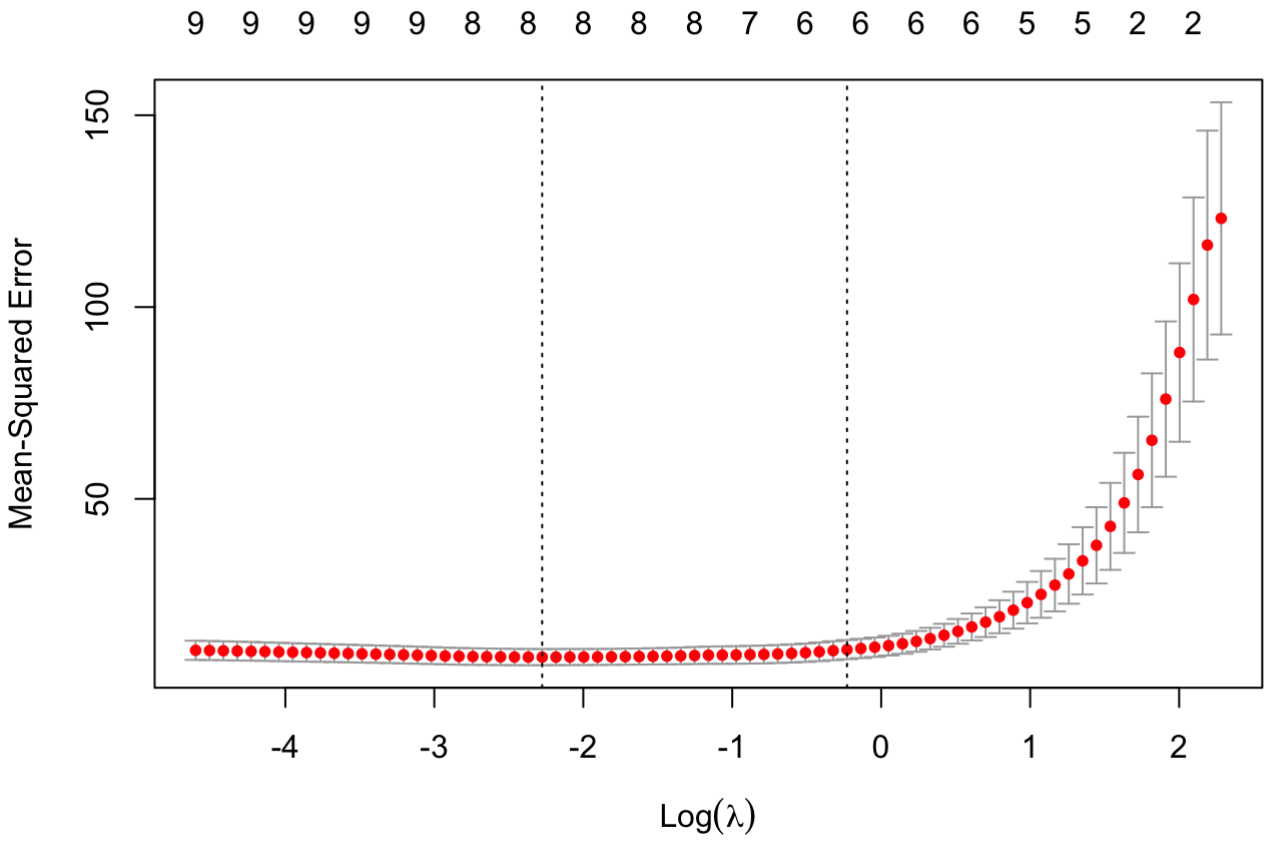
```
cv_model <- cv.glmnet(x, y, alpha = 1)
```

```
## Warning: Option grouped=FALSE enforced in cv.glmnet, since < 3 observations per
## fold
```

```
best_lambda <- cv_model$lambda.min
best_lambda
```

```
## [1] 0.1026849
```

```
plot(cv_model)
```




```
best_model <- glmnet(x, y, alpha = 1, lambda = best_lambda)
coef(best_model)
```

```
## 11 x 1 sparse Matrix of class "dgCMatrix"
##                       s0
## (Intercept) -80.91318784
## Fore          1.58861183
## Bicep         .
## Chest         0.09552940
## Neck          .
## Shoulder      0.02419127
## Waist         0.63077994
## Height        0.24647595
## Calf          0.53440295
## Thigh         0.35124799
## Head         -0.63520240
```

Multiple regression with selected model (output of stepwise regression)

```
lassomodel <- lm(formula = Mass ~ . - Bicep - Neck, data = PhysicalMeasures)
summary(lassomodel)
```

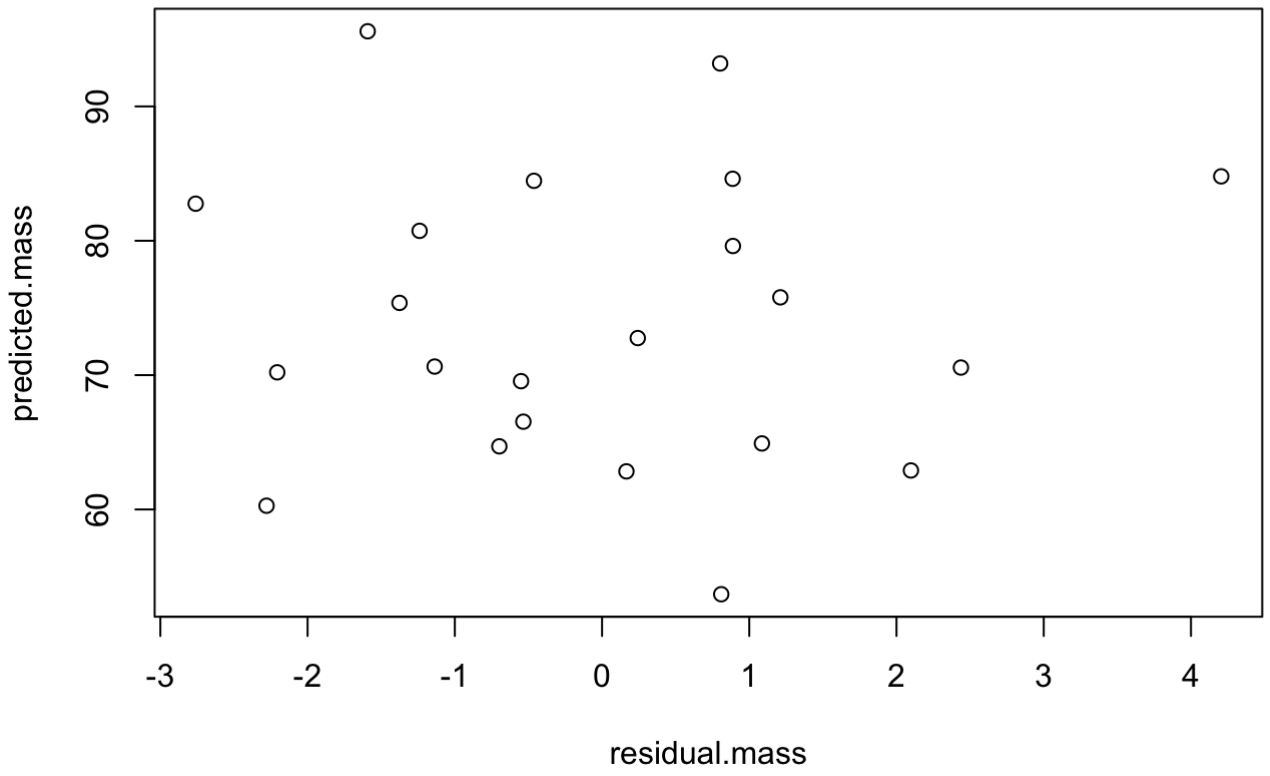
```
##
## Call:
## lm(formula = Mass ~ . - Bicep - Neck, data = PhysicalMeasures)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -2.7601 -1.2137 -0.1485  0.8885  4.2055
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -74.79622   26.21680  -2.853   0.0136 *
## Fore          1.69688    0.69860   2.429   0.0304 *
## Chest         0.15598    0.17867   0.873   0.3985
## Shoulder     -0.03469    0.22329  -0.155   0.8789
## Waist         0.64014    0.10503   6.095 3.81e-05 ***
## Height        0.27801    0.10345   2.687   0.0186 *
## Calf          0.53790    0.36486   1.474   0.1642
## Thigh         0.30383    0.28304   1.073   0.3026
## Head         -0.85502    0.47644  -1.795   0.0960 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.146 on 13 degrees of freedom
## Multiple R-squared:  0.9763, Adjusted R-squared:  0.9617
## F-statistic: 66.89 on 8 and 13 DF,  p-value: 2.599e-09
```

Diagnostics

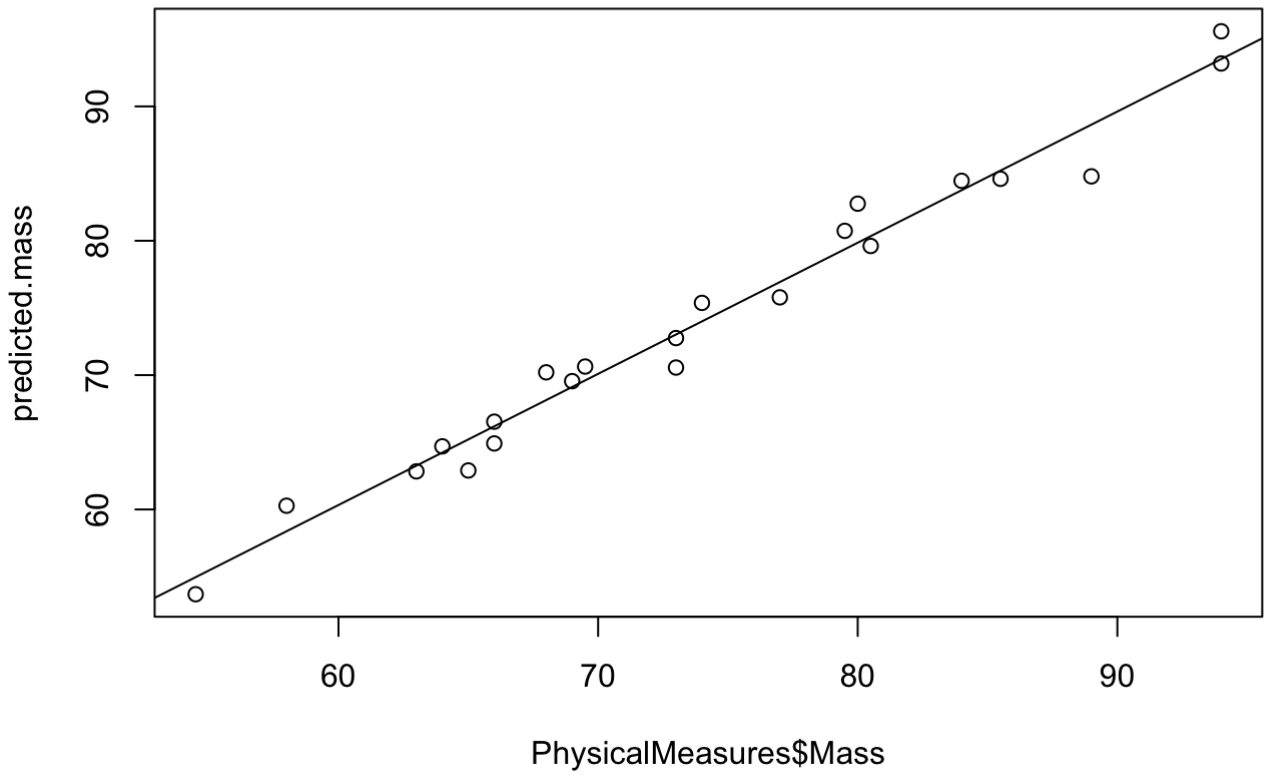
```
library("broom")
diagnostics <- augment(lassomodel)
```

Plots

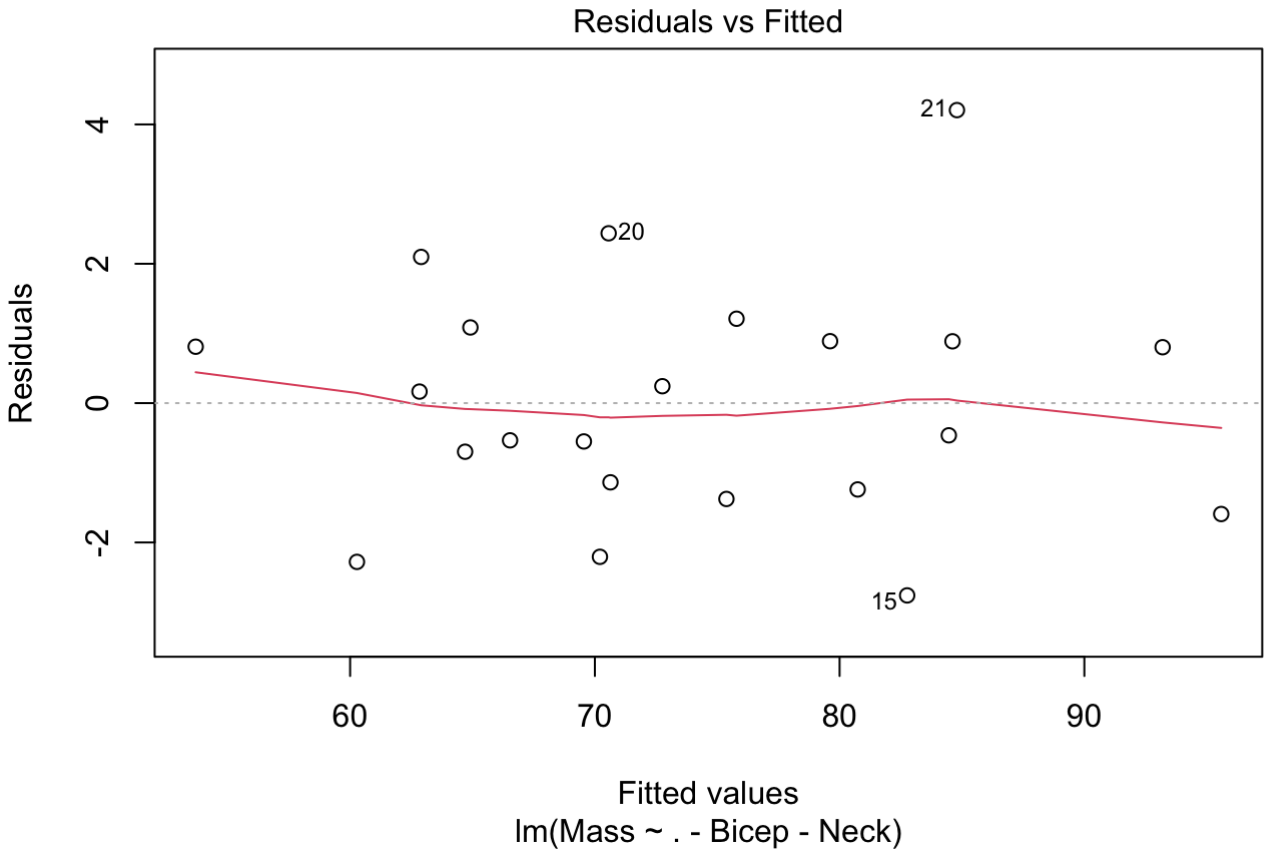
```
residual.mass = residuals(lassomodel)
predicted.mass = predict(lassomodel)
plot(residual.mass, predicted.mass)
```

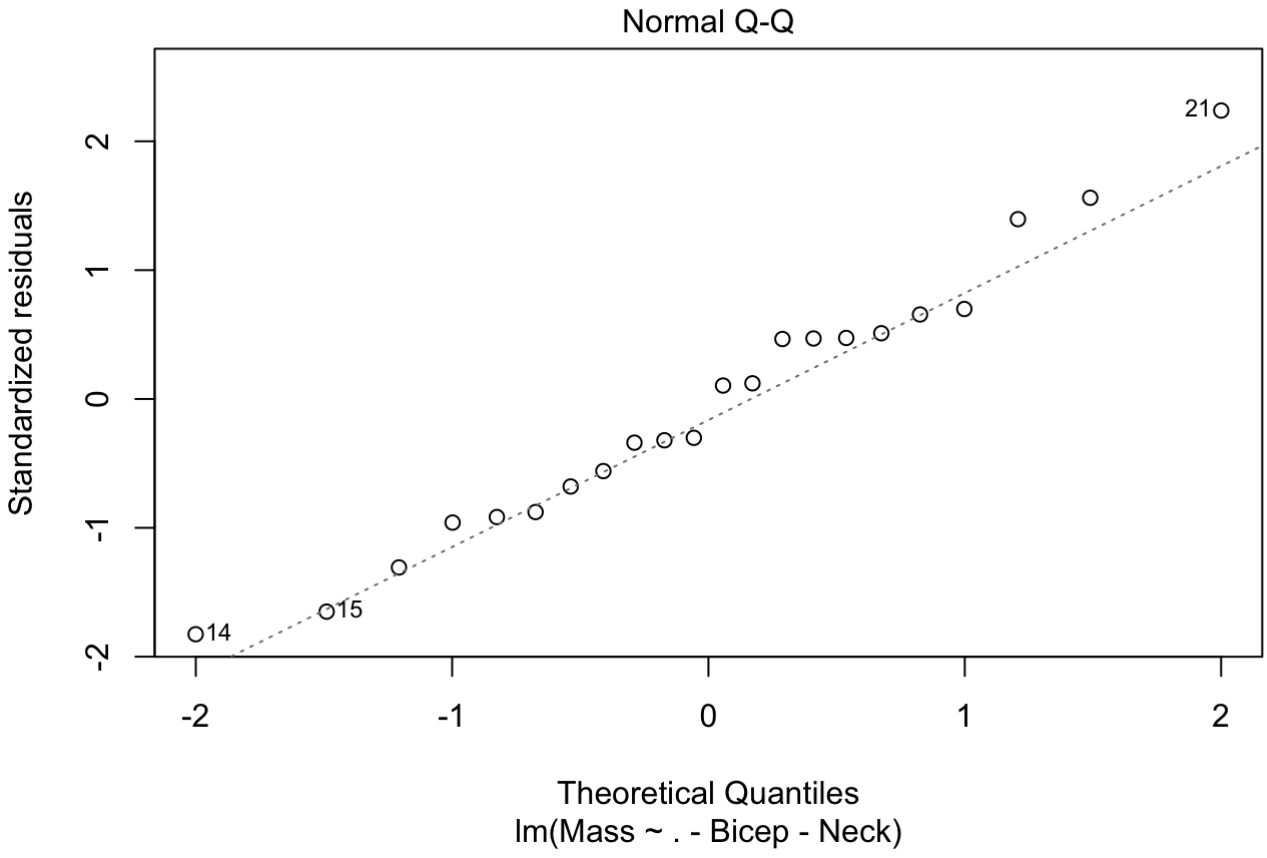


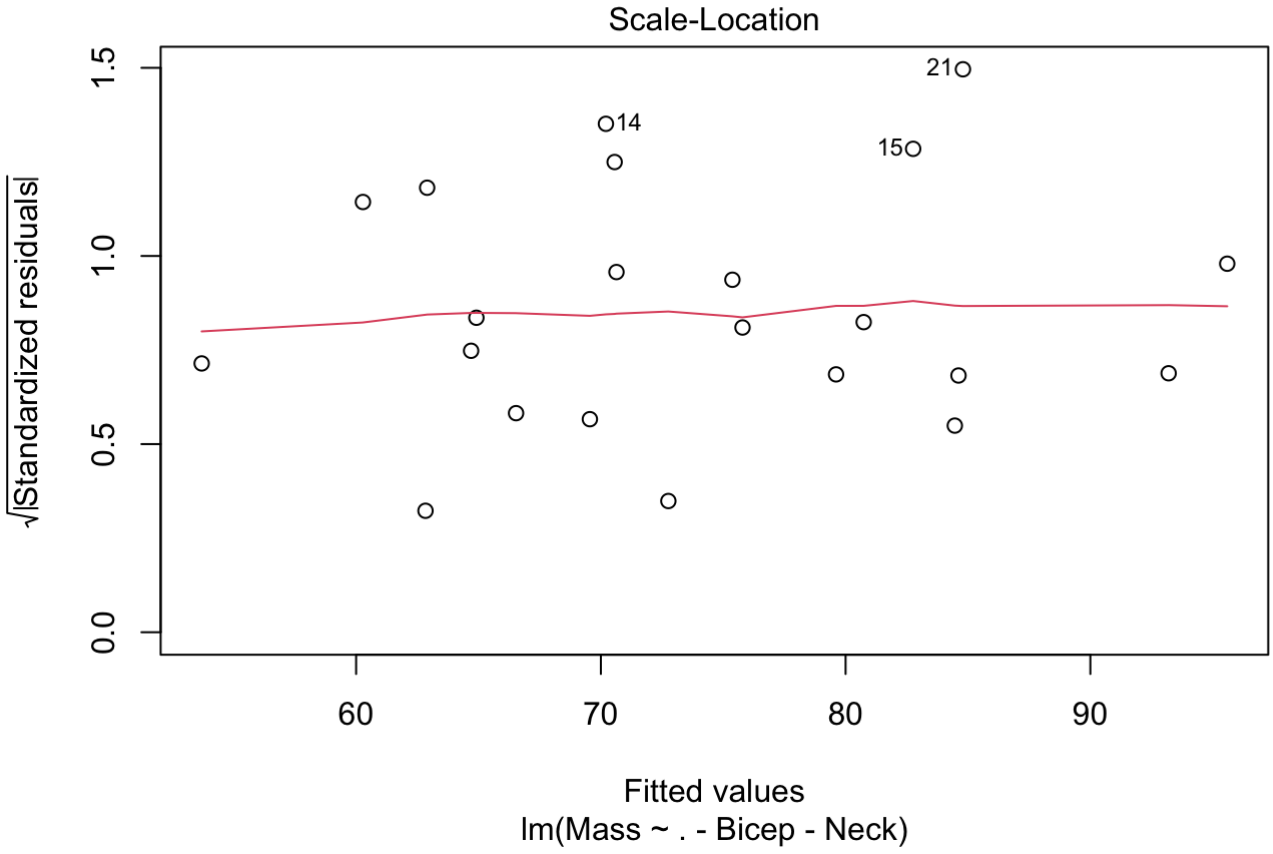

```
plot(PhysicalMeasures$Mass, predicted.mass)
reg = lm(predicted.mass ~ PhysicalMeasures$Mass)
abline(reg)
```

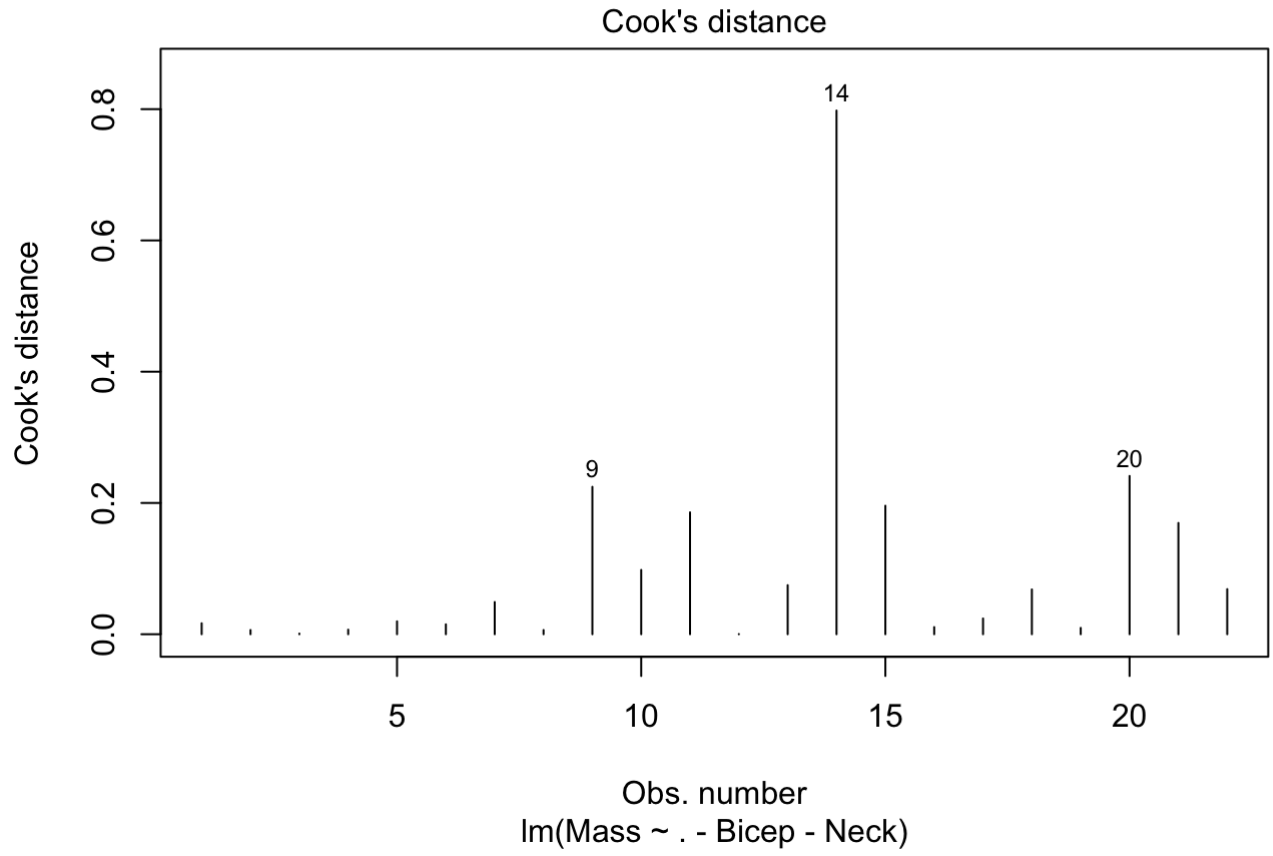


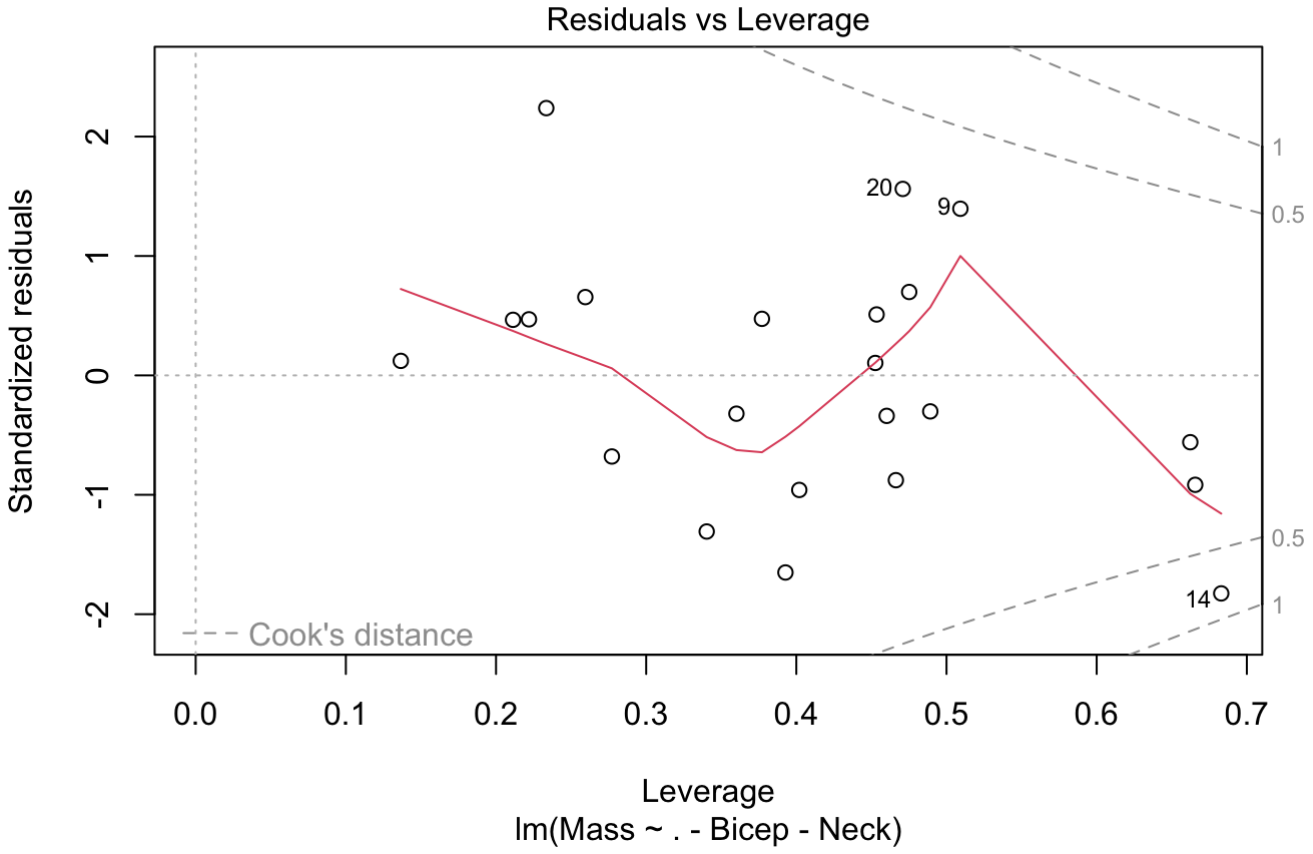

```
plot(lassomodel, which = c(1,2,3,4,5,6))
```

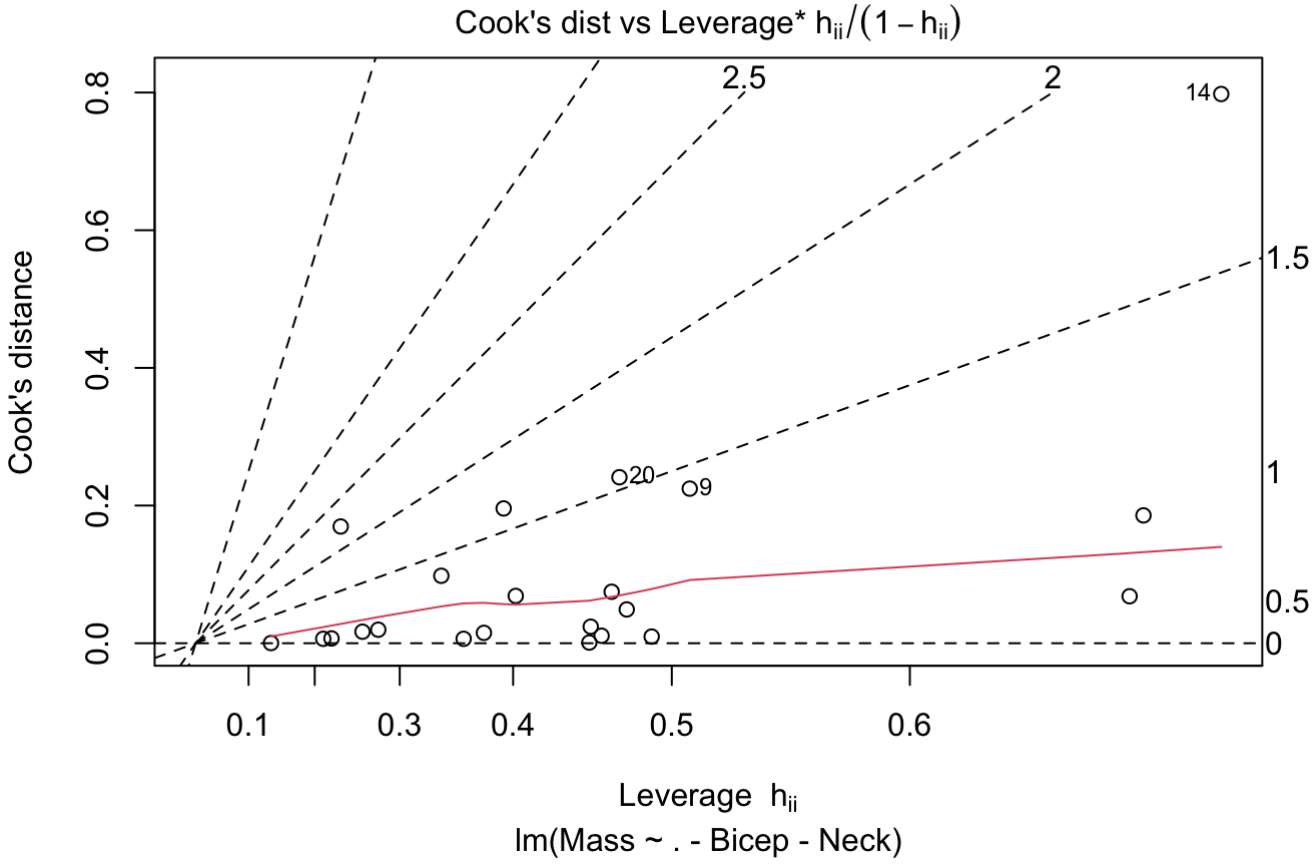

Residuals vs Fitted
Residuals
Fitted values
lm(Mass ~ . - Bicep - Neck)
21
20
15
4
2
0
-2
60
70
80
90


Normal Q-Q
Standardized residuals
Theoretical Quantiles
lm(Mass ~ . - Bicep - Neck)
21
15
14
2
1
0
-1
-2


Scale-Location
√|Standardized residuals|
1.5
1.0
0.5
0.0
21
14
15
60
70
80
90
Fitted values
lm(Mass ~ . - Bicep - Neck)

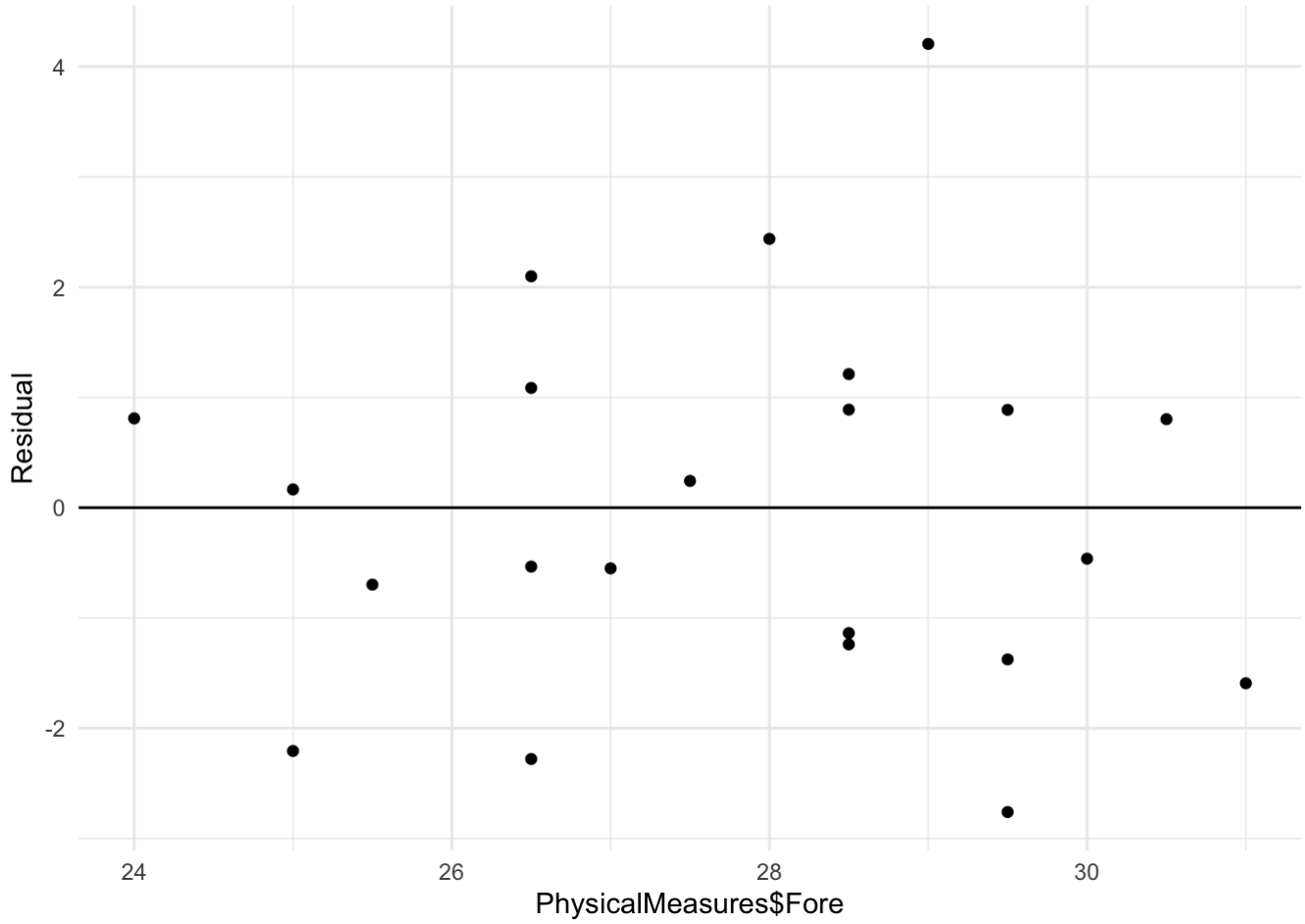

Cook's distance
Cook's distance
0.8
0.6
0.4
0.2
0.0
14
9
20
5
10
15
20
Obs. number
lm(Mass ~ . - Bicep - Neck)


Residuals vs Leverage
Standardized residuals
Leverage
lm(Mass ~ . - Bicep - Neck)
Cook's distance
20
9
14
1
0.5


Cook's dist vs Leverage* $h_{ii}/(1-h_{ii})$
Cook's distance
Leverage h_{ii}
lm(Mass ~ . - Bicep - Neck)
2.5
2
1.5
1
0.5
20
9
14

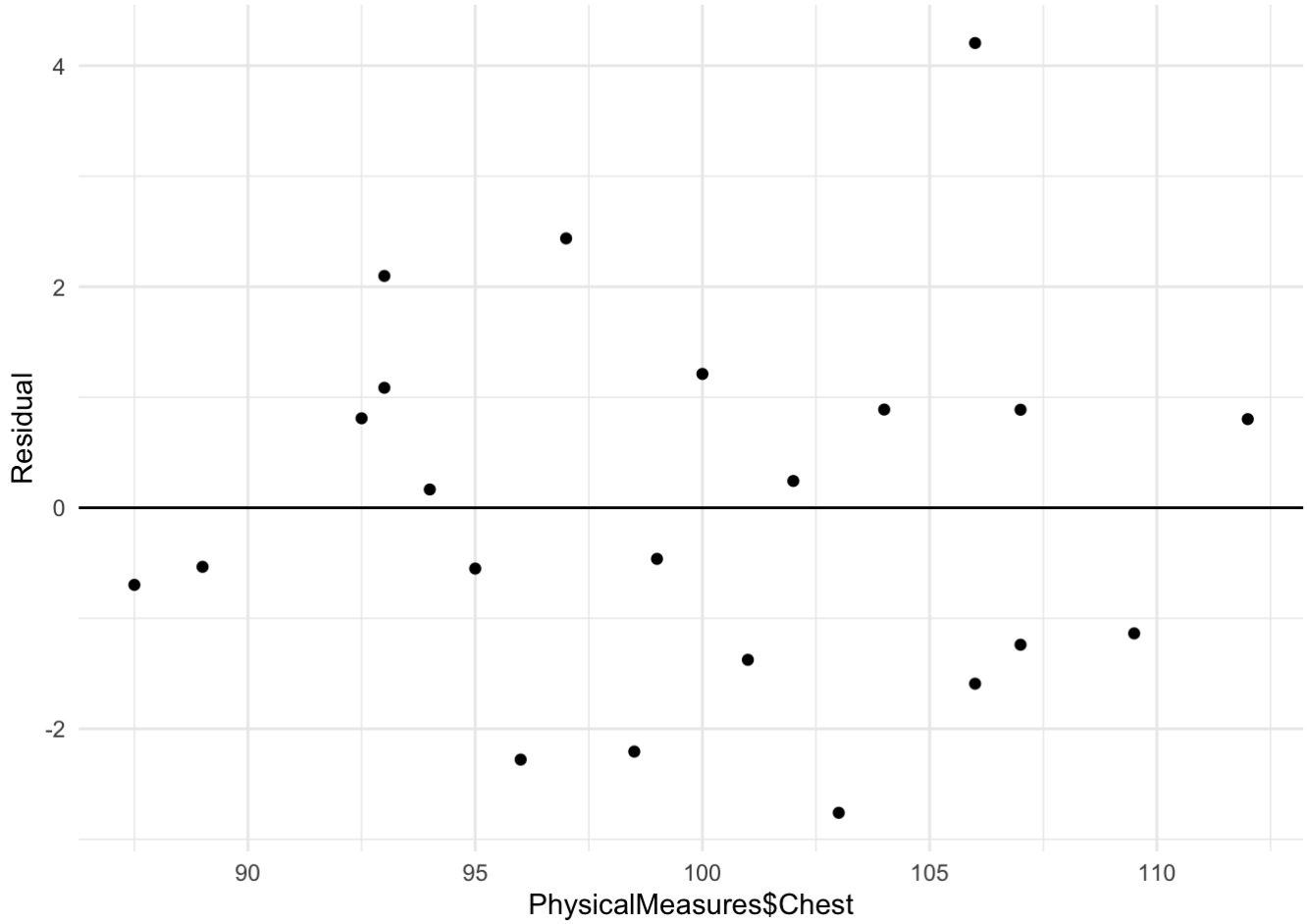
Residuals by Fore

```
library("GGally")
ggplot(diagnostics) +
  geom_point(aes(x = PhysicalMeasures$Fore, y = .resid)) +
  geom_hline(yintercept = 0) +
  ylab("Residual") +
  theme_minimal()
```



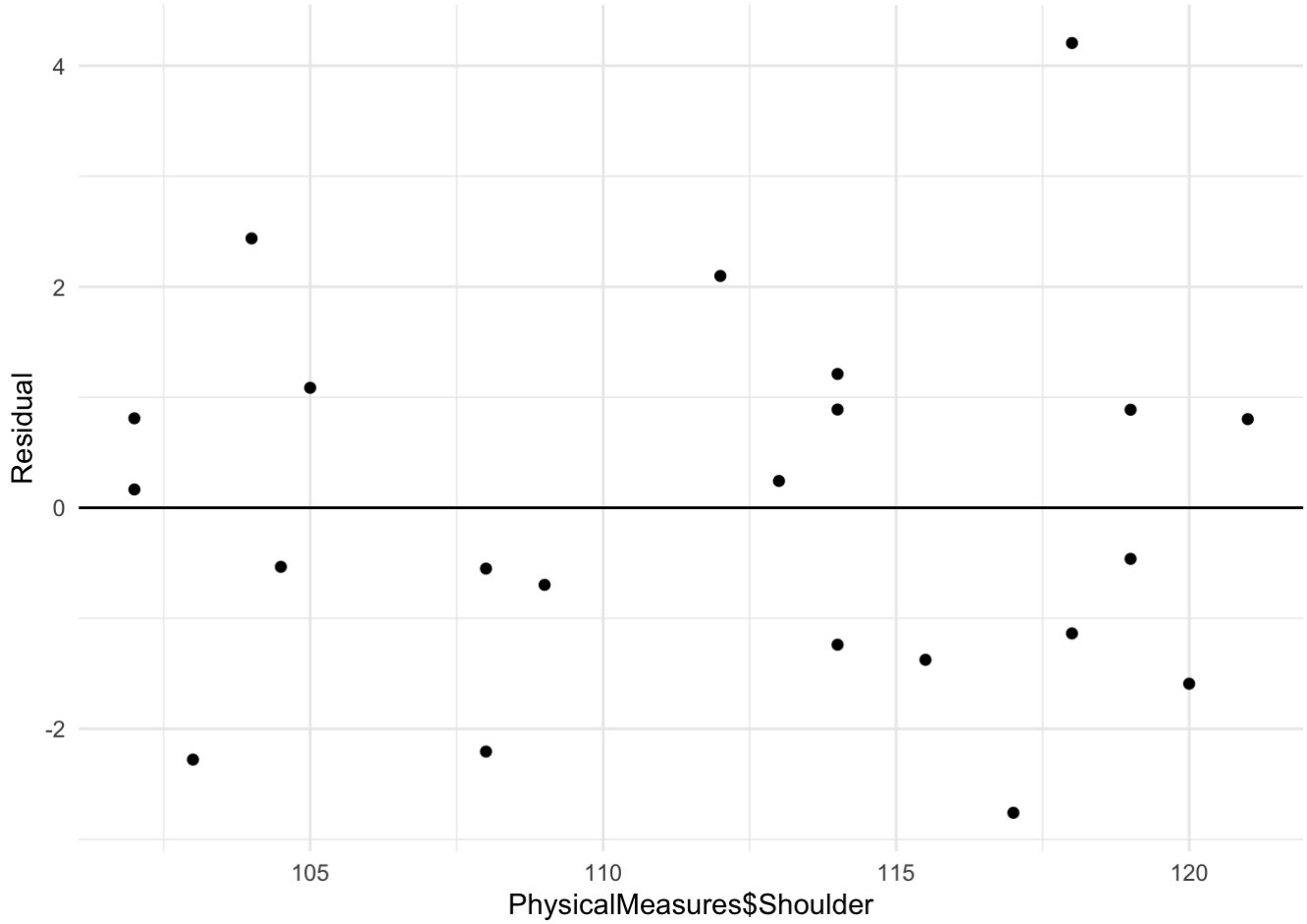
Residuals by Chest

```
library("GGally")
ggplot(diagnostics) +
  geom_point(aes(x = PhysicalMeasures$Chest, y = .resid)) +
  geom_hline(yintercept = 0) +
  ylab("Residual") +
  theme_minimal()
```



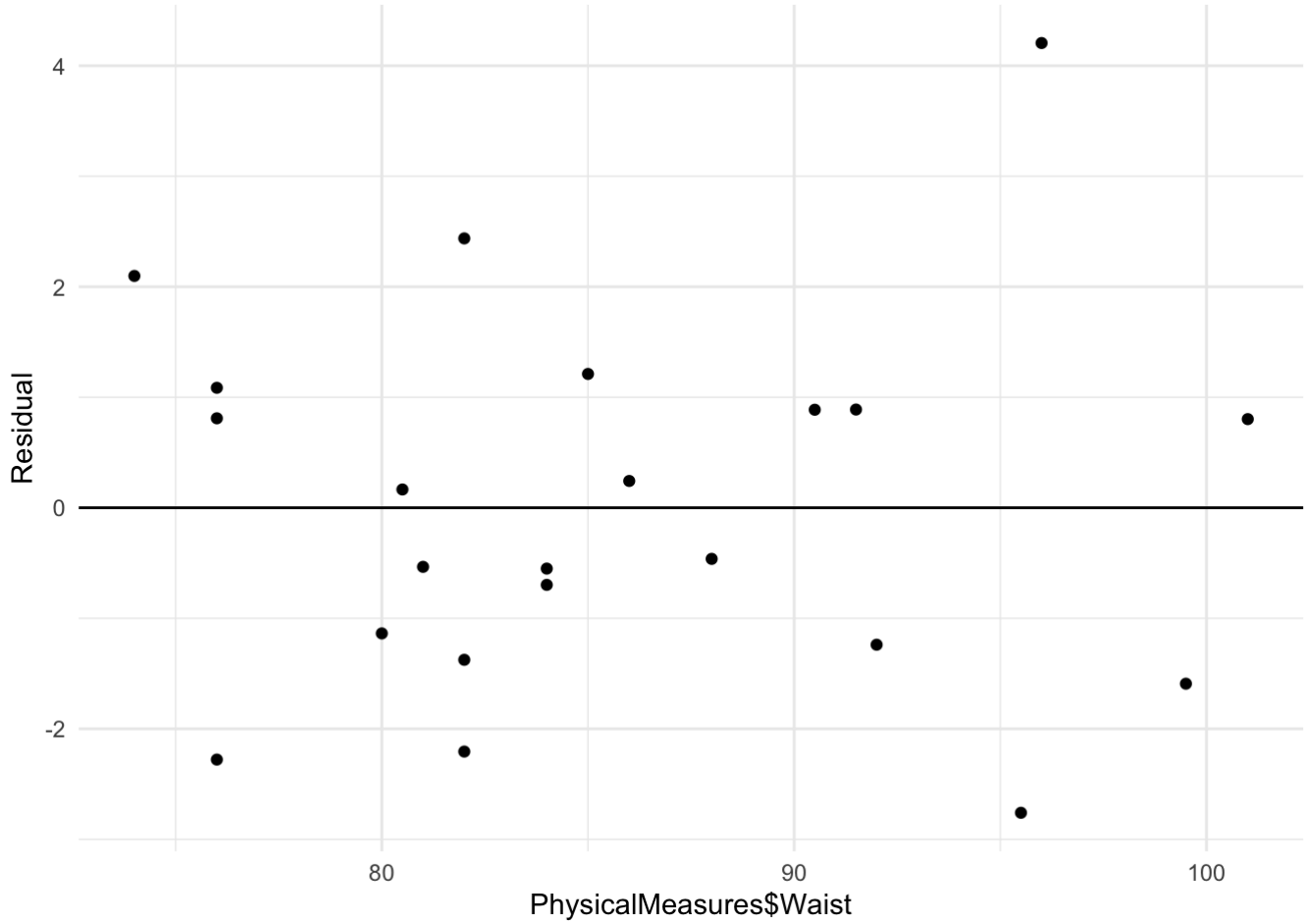
Residuals by Shoulder

```
library("GGally")
ggplot(diagnostics) +
  geom_point(aes(x = PhysicalMeasures$Shoulder, y = .resid)) +
  geom_hline(yintercept = 0) +
  ylab("Residual") +
  theme_minimal()
```



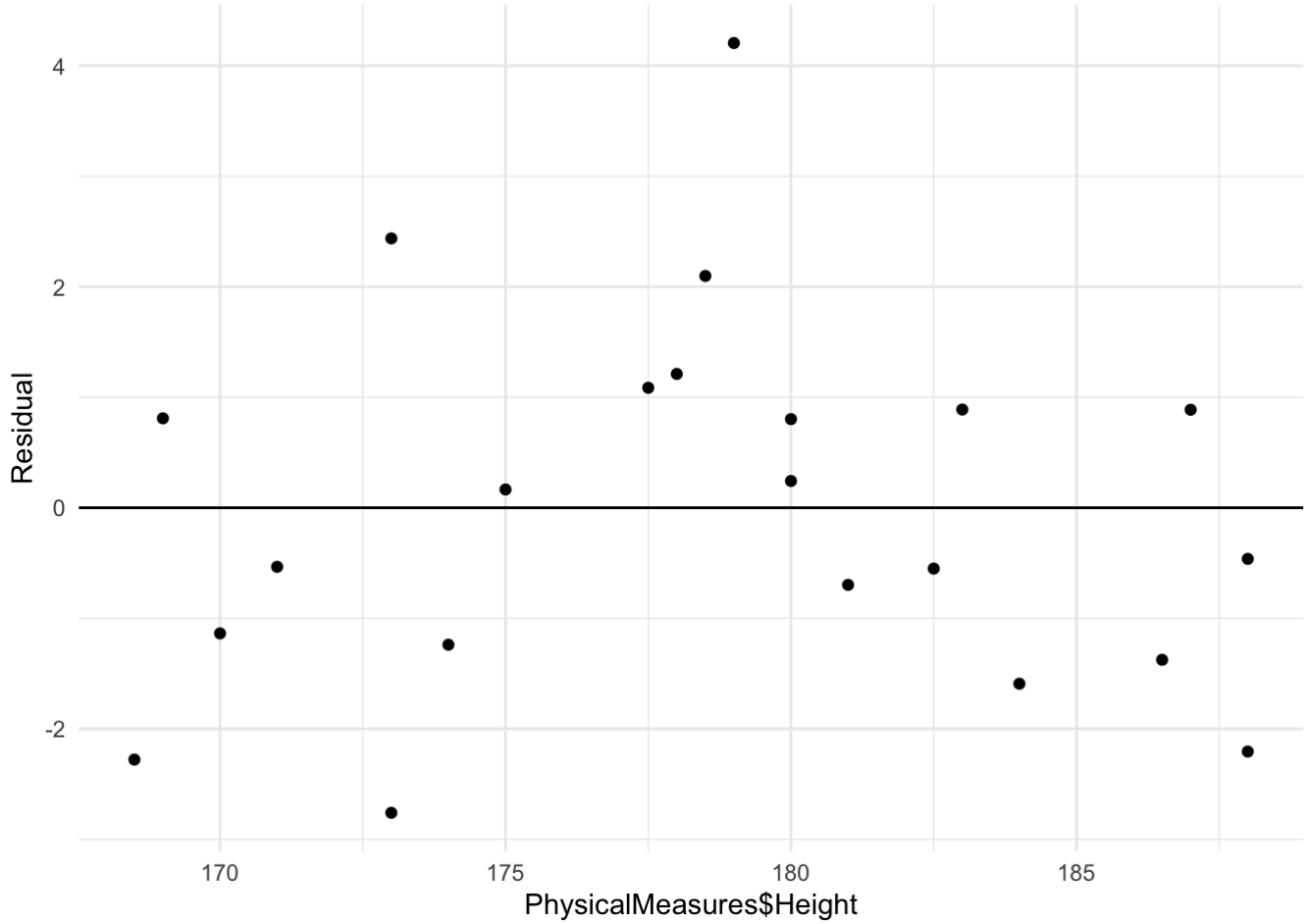
Residuals by Waist

```
library("GGally")
ggplot(diagnostics) +
  geom_point(aes(x = PhysicalMeasures$Waist, y = .resid)) +
  geom_hline(yintercept = 0) +
  ylab("Residual") +
  theme_minimal()
```



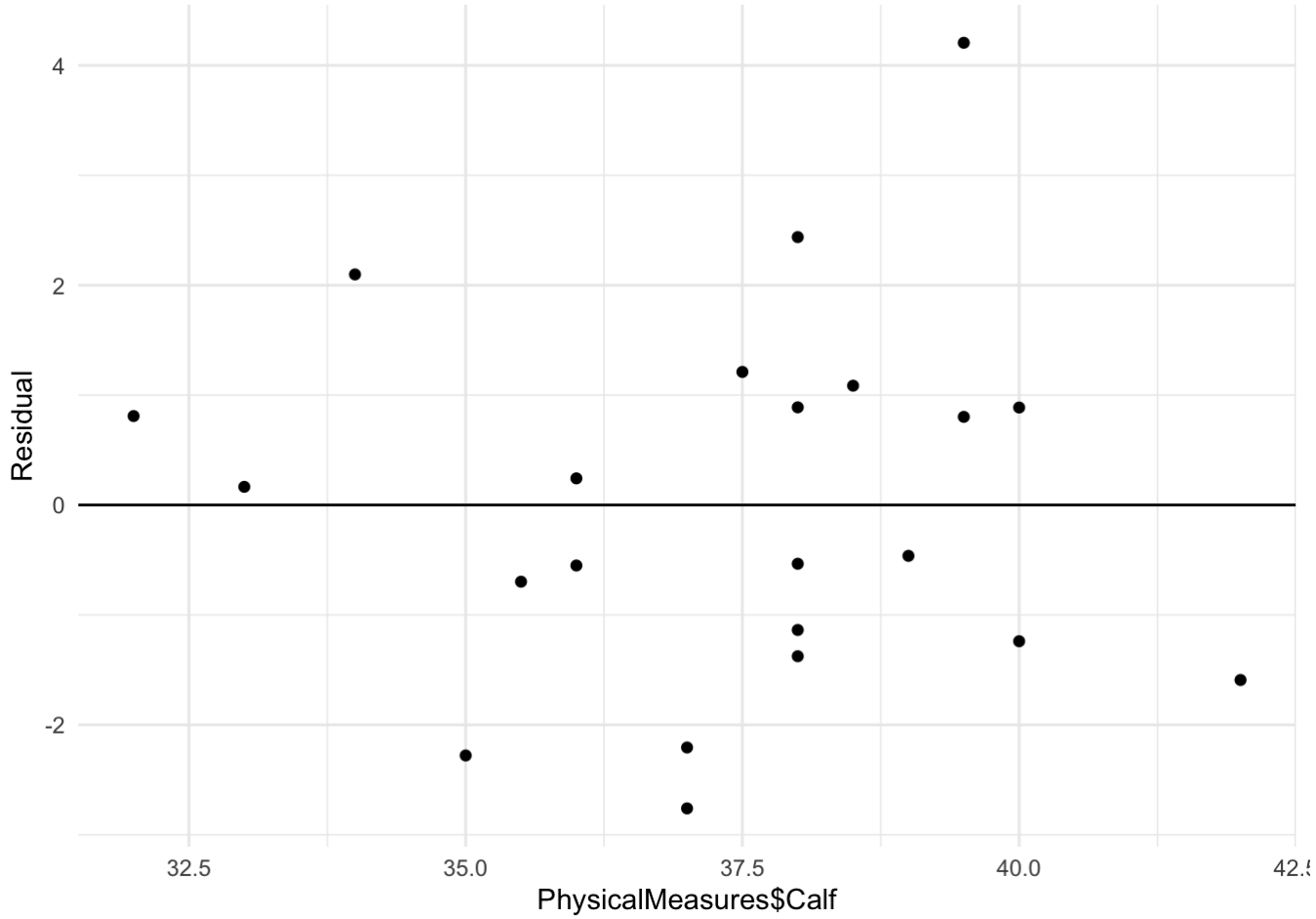
Residuals by Height

```
library("GGally")
ggplot(diagnostics) +
  geom_point(aes(x = PhysicalMeasures$Height, y = .resid)) +
  geom_hline(yintercept = 0) +
  ylab("Residual") +
  theme_minimal()
```



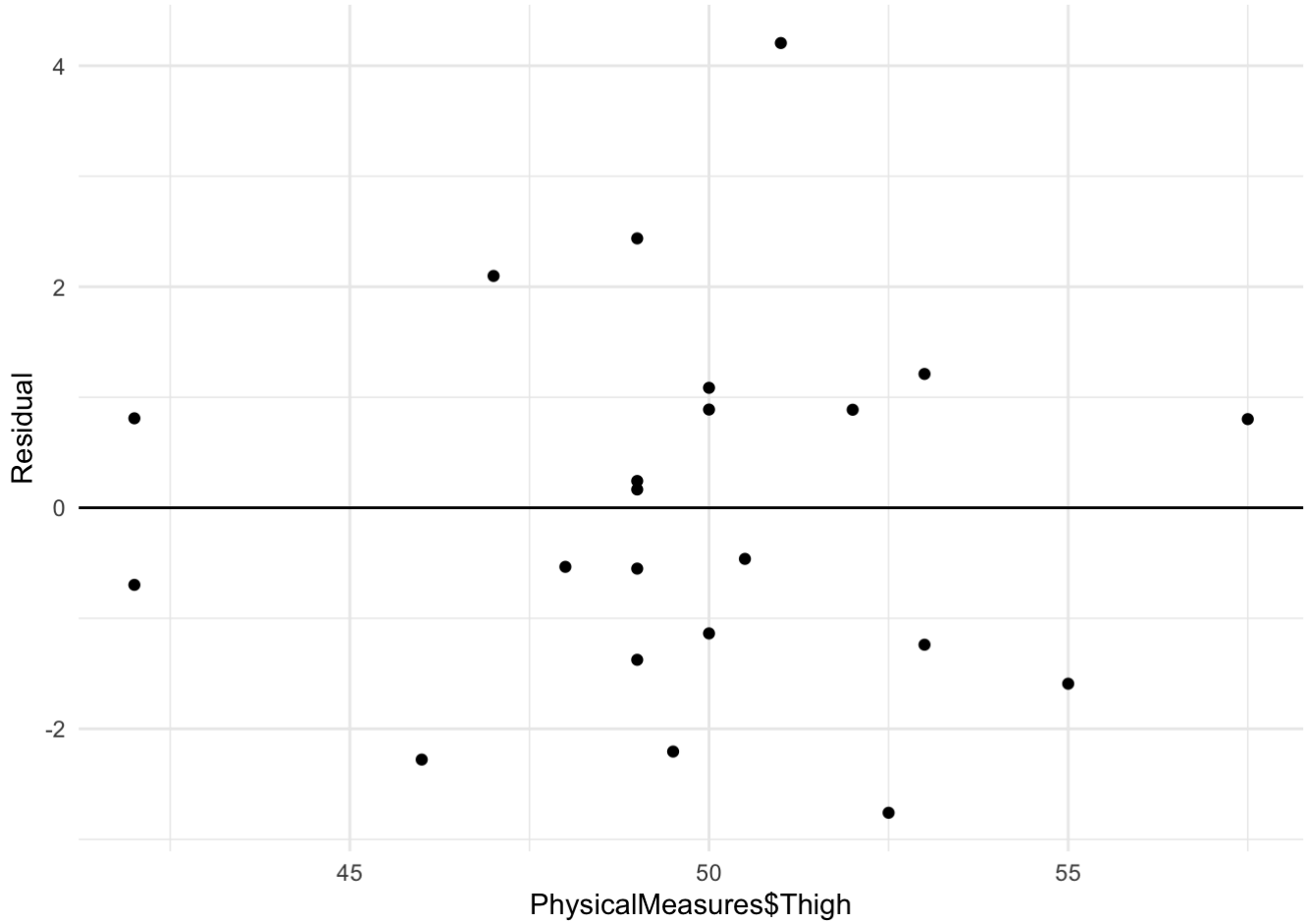
Residuals by Calf

```
library("GGally")
ggplot(diagnostics) +
  geom_point(aes(x = PhysicalMeasures$Calf, y = .resid)) +
  geom_hline(yintercept = 0) +
  ylab("Residual") +
  theme_minimal()
```



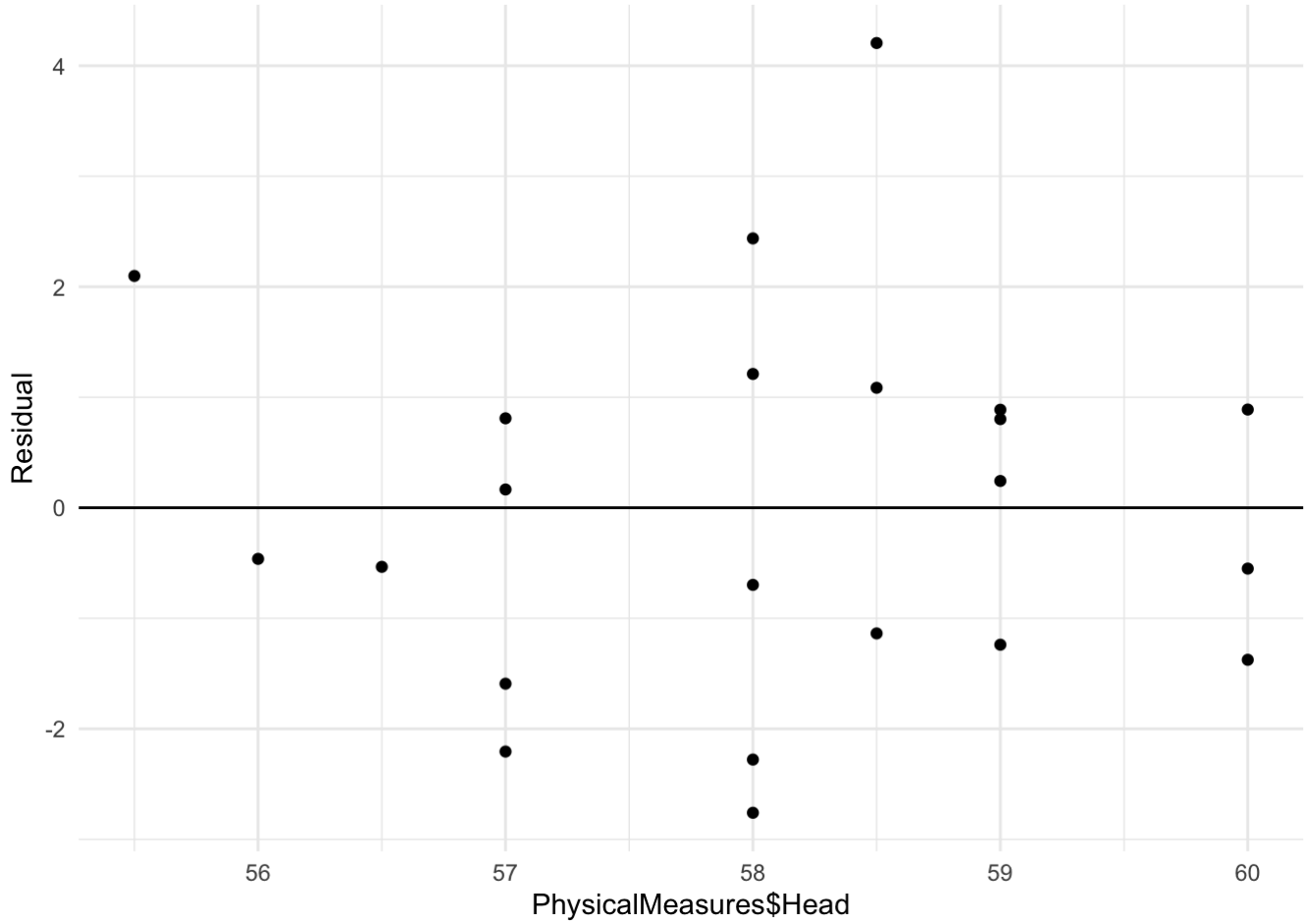
Residuals by Thigh

```
library("GGally")
ggplot(diagnostics) +
  geom_point(aes(x = PhysicalMeasures$Thigh, y = .resid)) +
  geom_hline(yintercept = 0) +
  ylab("Residual") +
  theme_minimal()
```



Residuals by Head

```
library("GGally")
ggplot(diagnostics) +
  geom_point(aes(x = PhysicalMeasures$Head, y = .resid)) +
  geom_hline(yintercept = 0) +
  ylab("Residual") +
  theme_minimal()
```



The stepwise model is a better fit

The stepwise model has a higher adjusted r-squared value