Question 3 & 4 using Houses Data

2024-03-14

Question 3, part (a)

```
library(tidyverse)
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.2 --
## v ggplot2 3.4.0     v purrr   1.0.1
## v tibble  3.1.8     v dplyr   1.1.0
## v tidyr   1.3.0     v stringr 1.5.0
## v readr   2.1.3     v forcats 1.0.0
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
library(dplyr)
Houses<-read_table("http://www.stat.ufl.edu/~aa/glm/data/Houses.dat")
```

```
## Warning: Missing column names filled in: 'X8' [8]
```

```
##
## -- Column specification --------------------------------------------------------
## cols(
##   case = col_double(),
##   taxes = col_double(),
##   beds = col_double(),
##   baths = col_double(),
##   new = col_double(),
##   price = col_double(),
##   size = col_double(),
##   X8 = col_logical()
## )
```

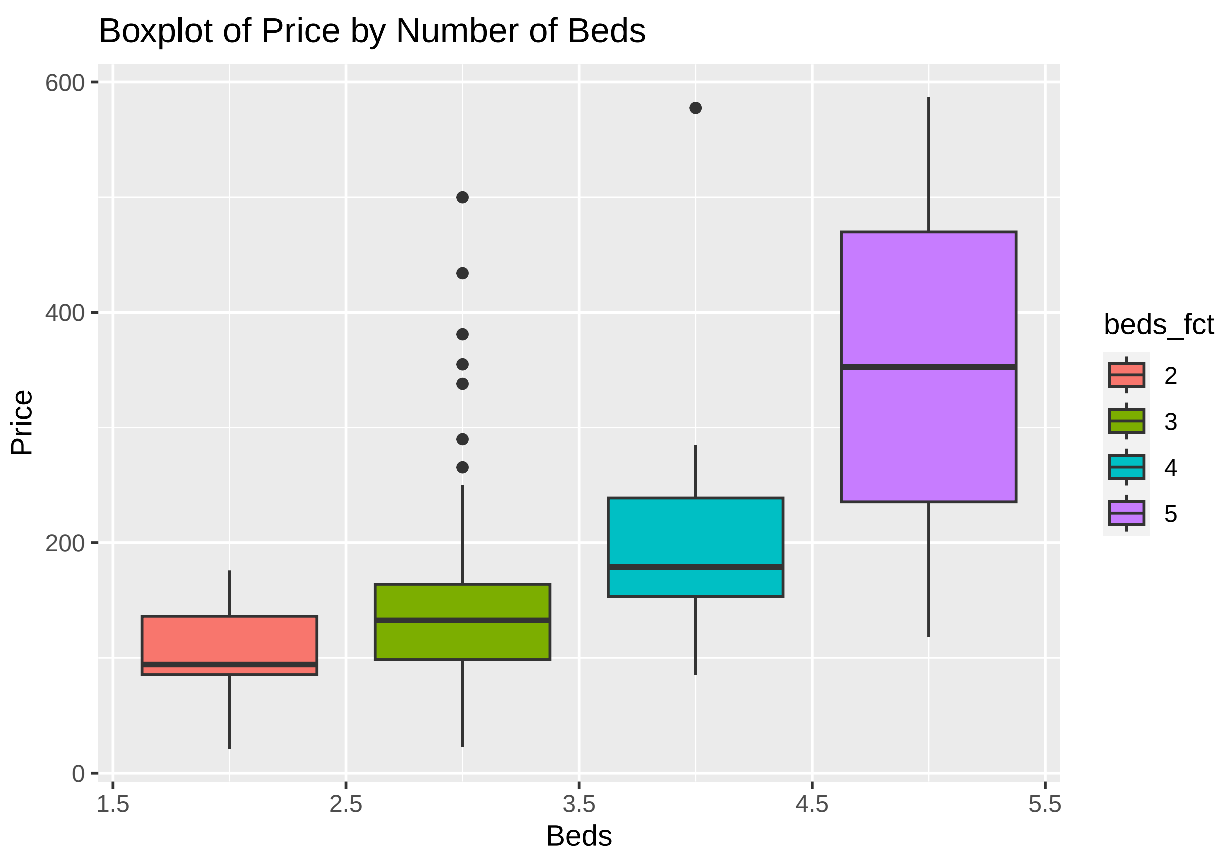
```
Houses <- Houses %>% select(-X8)
```

```
summary_stats <- summary(Houses)
Houses <- Houses %>%
  mutate(beds_fct = factor(beds)) %>%
  mutate(new_fct = factor(new)) %>%
  mutate(baths_fct = factor(baths))
summary_stats
```

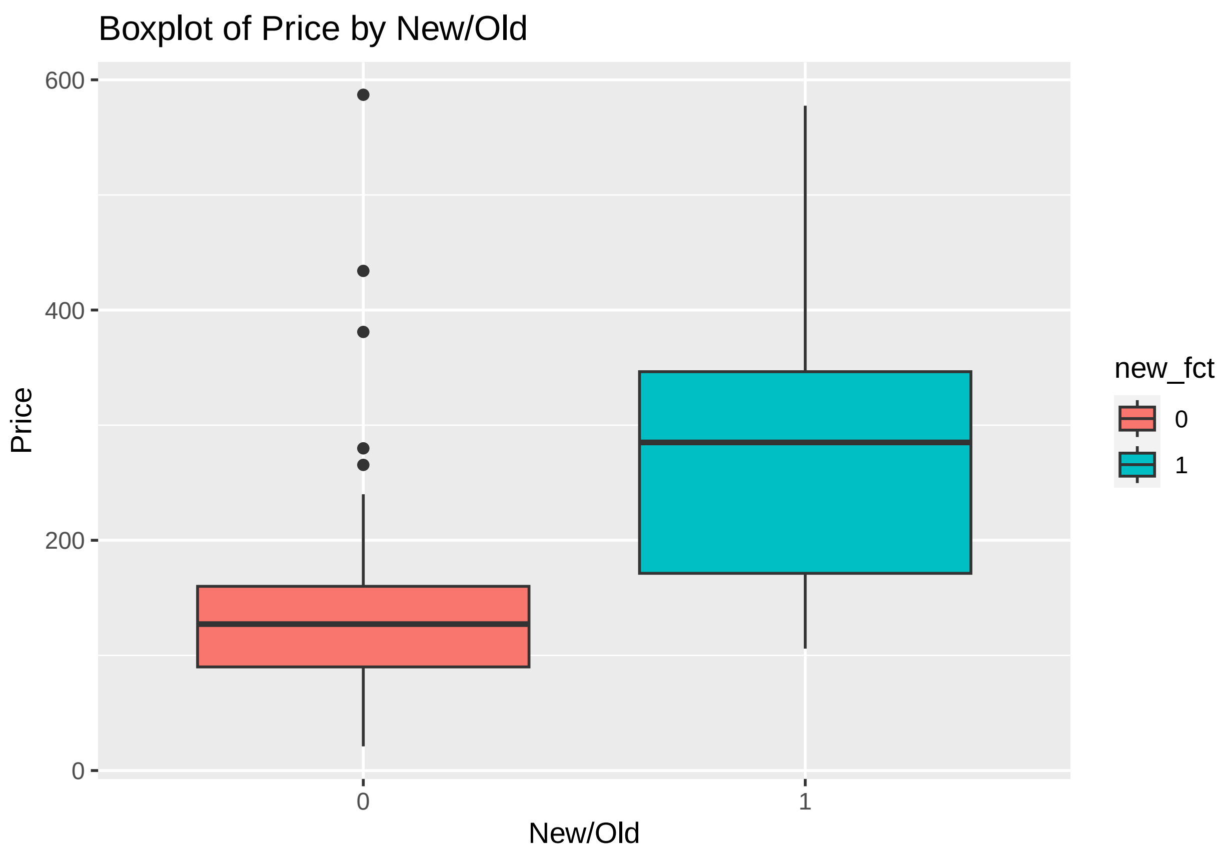
```
##        case              taxes             beds          baths              new
```

```
##  Min.   :  1.00   Min.   :  20   Min.   :2   Min.   :1.00   Min.   :0.00
##  1st Qu.: 25.75   1st Qu.:1178   1st Qu.:3   1st Qu.:2.00   1st Qu.:0.00
##  Median : 50.50   Median :1614   Median :3   Median :2.00   Median :0.00
##  Mean   : 50.50   Mean   :1908   Mean   :3   Mean   :1.96   Mean   :0.11
##  3rd Qu.: 75.25   3rd Qu.:2238   3rd Qu.:3   3rd Qu.:2.00   3rd Qu.:0.00
##  Max.   :100.00   Max.   :6627   Max.   :5   Max.   :4.00   Max.   :1.00
##      price             size
##  Min.   : 21.00   Min.   : 580
##  1st Qu.: 93.22   1st Qu.:1215
##  Median :132.60   Median :1474
##  Mean   :155.33   Mean   :1629
##  3rd Qu.:169.62   3rd Qu.:1865
##  Max.   :587.00   Max.   :4050
```

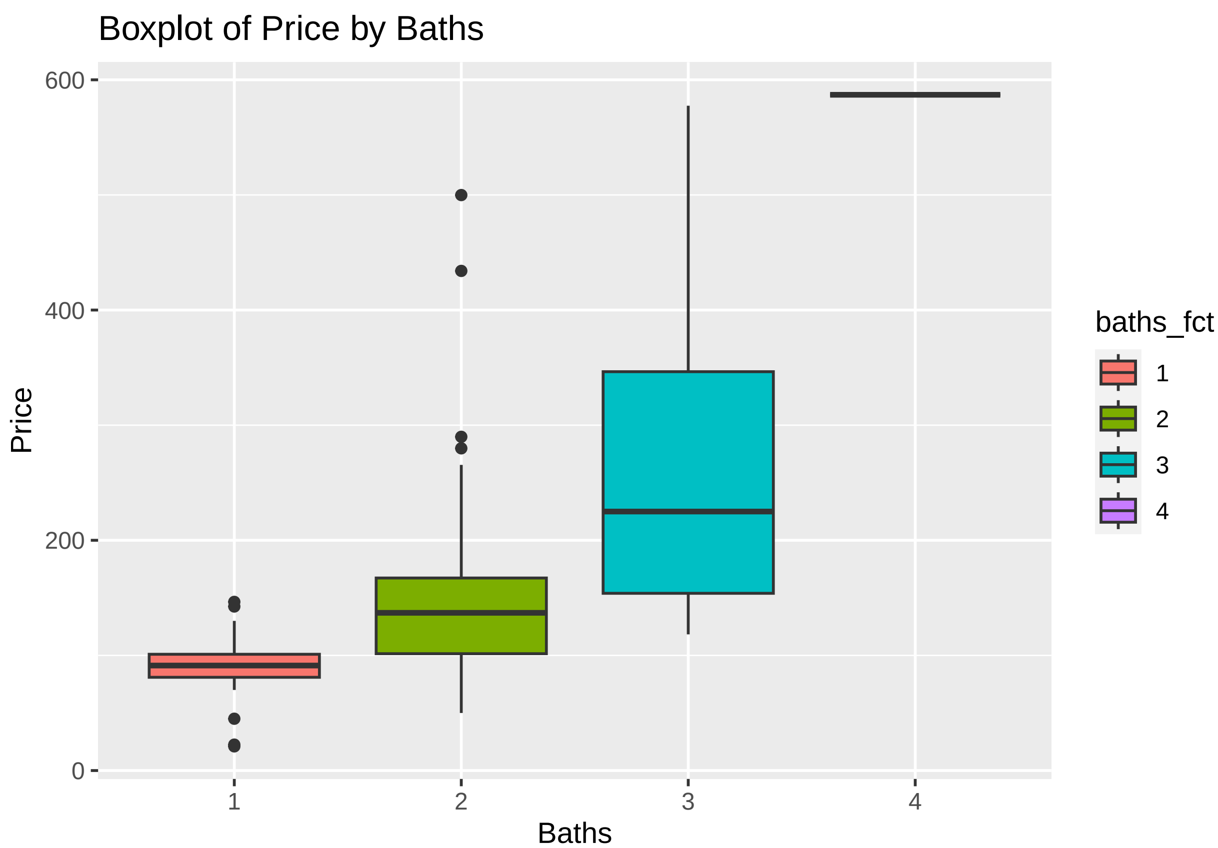
```r
# Create a boxplot
boxplot_beds <- ggplot(Houses, aes(x = beds, y = price, fill = beds_fct)) +
  geom_boxplot() + labs(title = "Boxplot of Price by Number of Beds", x = "Beds", y = "Price")
boxplot_beds
```




```r
boxplot_new <- ggplot(Houses, aes(x = new_fct, y = price, fill = new_fct)) +
  geom_boxplot() + labs(title = "Boxplot of Price by New/Old", x = "New/Old", y = "Price")
boxplot_new
```

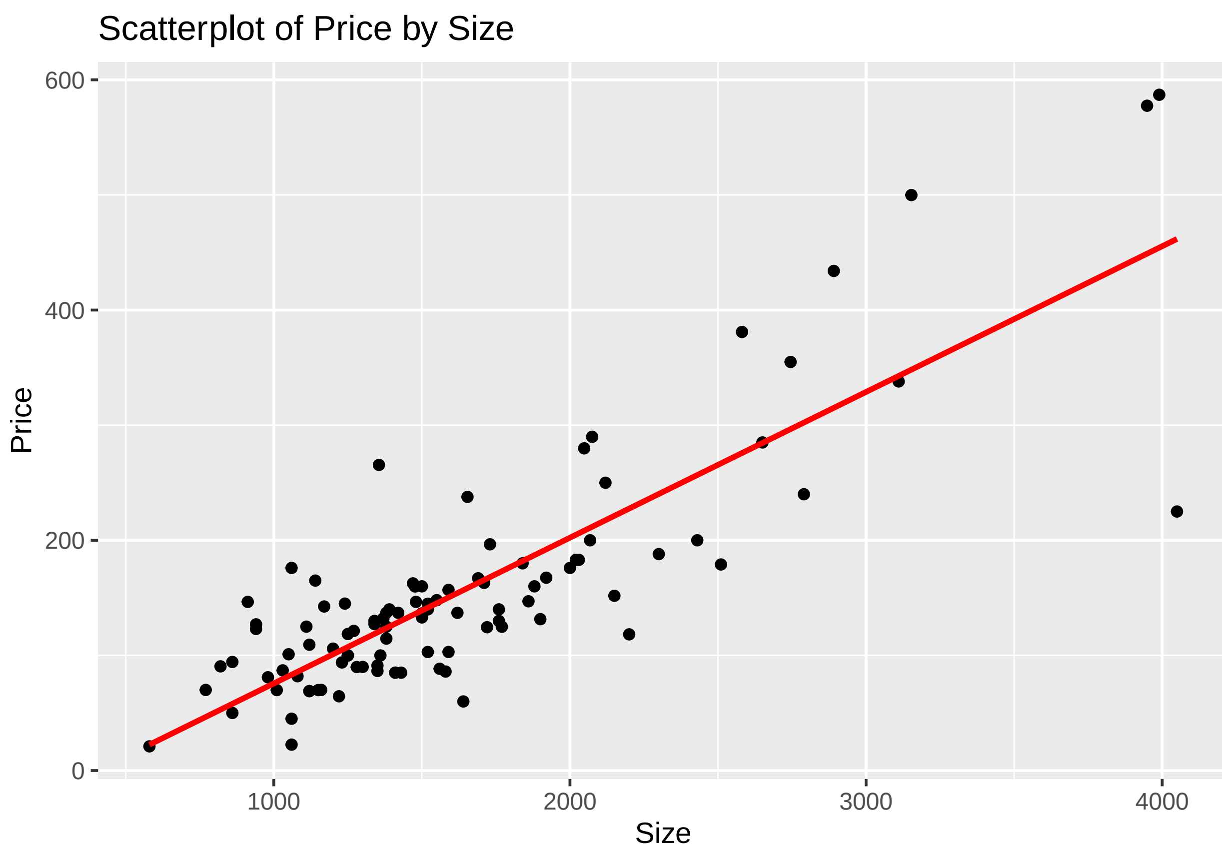



```
boxplot_bath <- ggplot(Houses, aes(x = baths_fct, y = price, fill = baths_fct)) +
  geom_boxplot() + labs(title = "Boxplot of Price by Baths", x = "Baths", y = "Price")
boxplot_bath
```



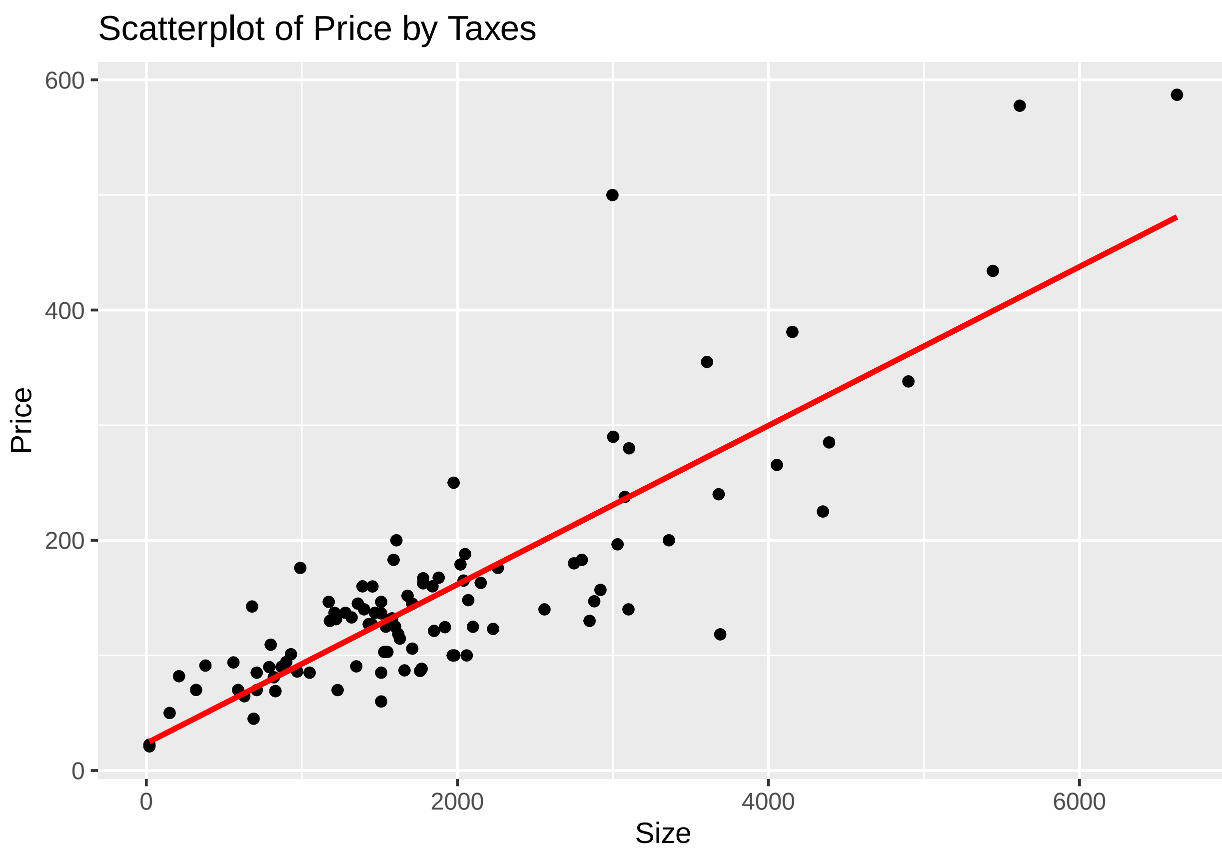

```
# Create scatterplots
scatterplot_size <- ggplot(Houses, aes(x = size, y = price)) +
  geom_point(color = "black") +
  labs(title = "Scatterplot of Price by Size", x = "Size", y = "Price") +
  geom_smooth(method = "lm", se = FALSE, color = "red")  # Add linear regression line
scatterplot_size
```

```
## `geom_smooth()` using formula = 'y ~ x'
```




```r
scatterplot_taxes <- ggplot(Houses, aes(x = taxes, y = price)) +
  geom_point(color = "black") +
  labs(title = "Scatterplot of Price by Taxes", x = "Size", y = "Price") +
  geom_smooth(method = "lm", se = FALSE, color = "red")  # Add linear regression line
scatterplot_taxes
```

```
## `geom_smooth()` using formula = 'y ~ x'
```



Question 3, part (b)

```
library(MASS)
```

```
##
## Attaching package: 'MASS'

## The following object is masked from 'package:dplyr':
##
##     select
```

```
library(tidyverse)

# Load the data (assuming the dataset is correctly formatted and accessible at the given URL)
Houses <- read_table("http://www.stat.ufl.edu/~aa/glm/data/Houses.dat")
```

```
## Warning: Missing column names filled in: 'X8' [8]
```

```
##
## -- Column specification --------------------------------------------------------
## cols(
##   case = col_double(),
##   taxes = col_double(),
##   beds = col_double(),
```

```
##   baths = col_double(),
##   new = col_double(),
##   price = col_double(),
##   size = col_double(),
##   X8 = col_logical()
## )
```

```r
# Start with a model that only includes the intercept
initial_model <- glm(price ~ 1, data = Houses, family = gaussian)

# Use stepAIC to perform forward selection
final_model <- stepAIC(initial_model, scope = list(lower = initial_model, upper = ~ size + new + baths 
                       direction = "forward", trace = FALSE)

# Print the summary of the final model
summary(final_model)
```

```
##
## Call:
## glm(formula = price ~ taxes + size + new, family = gaussian,
##     data = Houses)
##
## Deviance Residuals:
##      Min        1Q    Median        3Q       Max
## -165.501   -25.426     1.449    20.536   168.747
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -21.353776  13.311487  -1.604  0.11196
## taxes         0.037231   0.006735   5.528 2.78e-07 ***
## size          0.061704   0.012499   4.937 3.35e-06 ***
## new          46.373703  16.459019   2.818  0.00588 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 2225.115)
##
##     Null deviance: 1015150  on 99  degrees of freedom
## Residual deviance:  213611  on 96  degrees of freedom
## AIC: 1060.5
##
## Number of Fisher Scoring iterations: 2
```

Interpretation for the intercept: cannot be interpreted because at the intercept, the house would have 0ft^2 which does not exist (there is no house that has 0 square footage)

Interpretation for "taxes" interpretation: For every 1 monetary unit increase in tax, the price of the house increases by 37.231 units, keeping all other factors constant.

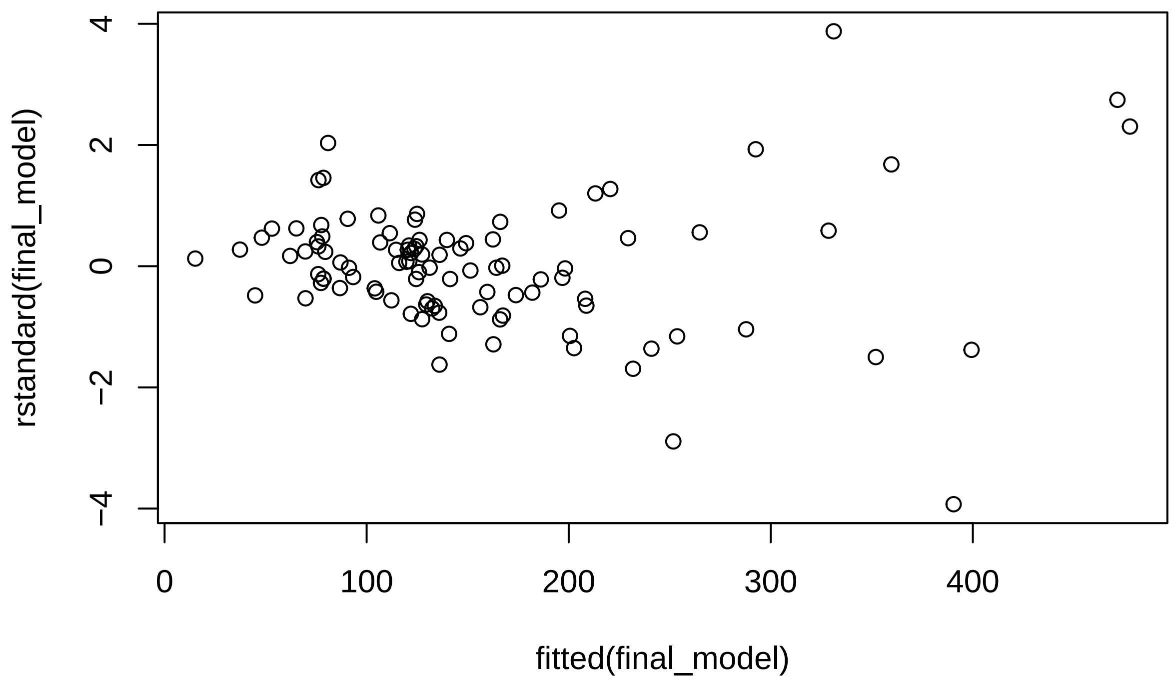
Interpretation for "size" coefficient: For every 1ft^2 increase in house size, the price of the house increases by 61.704 units, keeping all other factors constant.

Interpretation for "new" coefficient: If the house is new, then the price increases by 46,373.703 units, keeping all other factors constant.

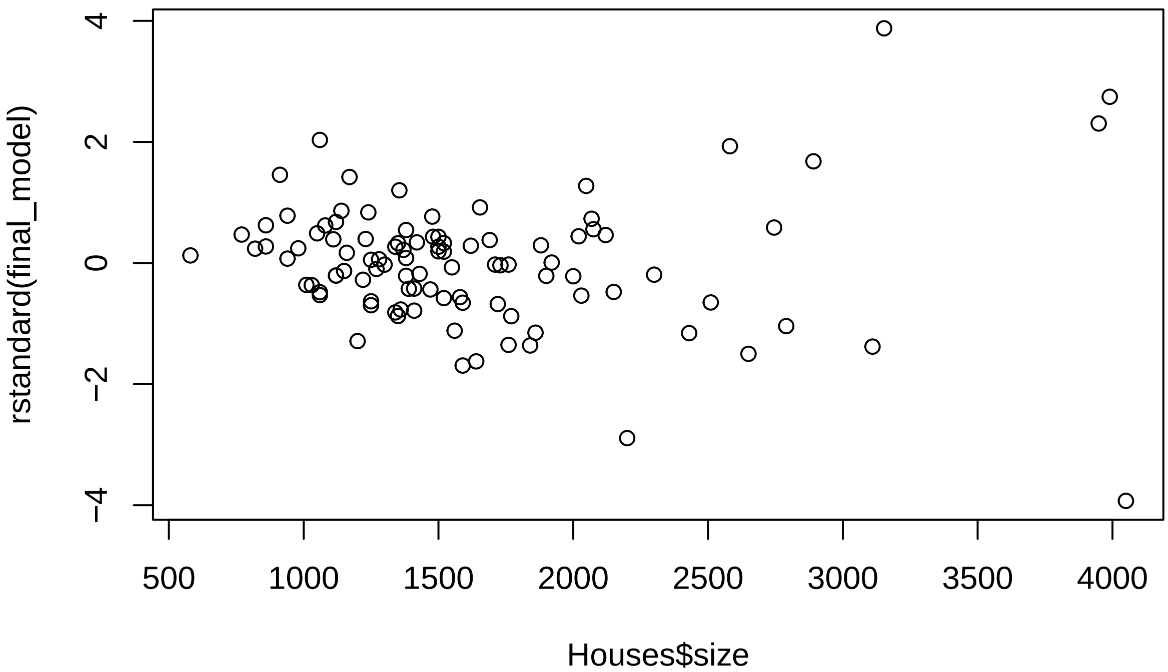
The coefficients for “taxes” and “size” are significant at the 0.001 significance level. The coefficient “new” is significant at the significance level of 0.01.

Question 3, part (c)

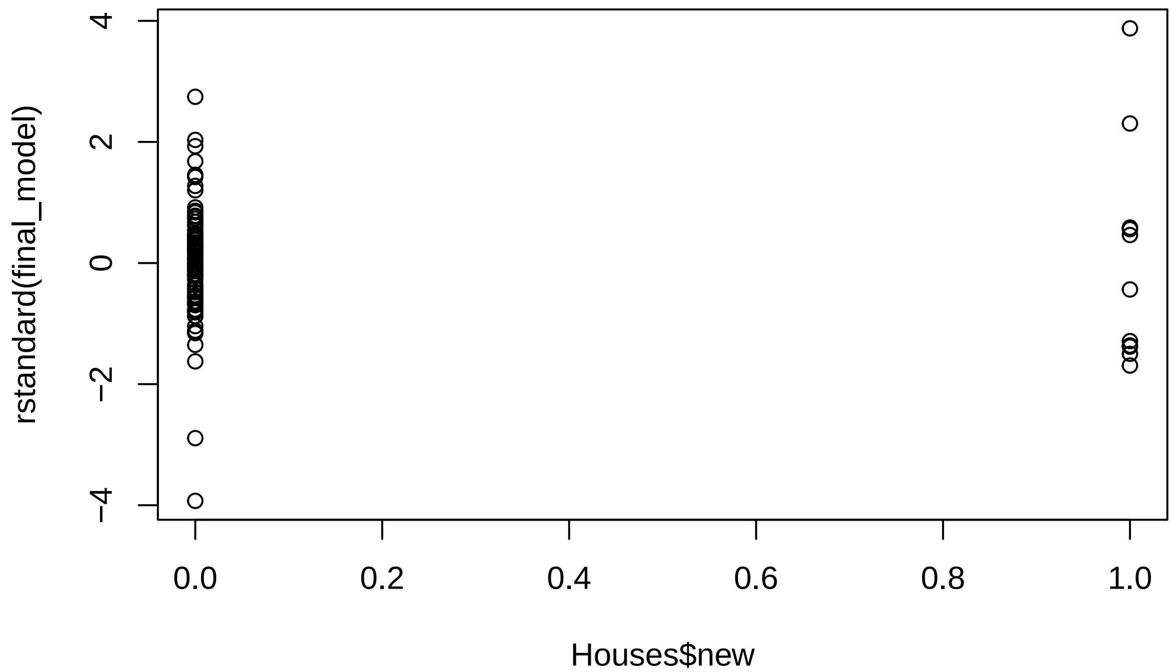
```
plot(fitted(final_model), rstandard(final_model))
```



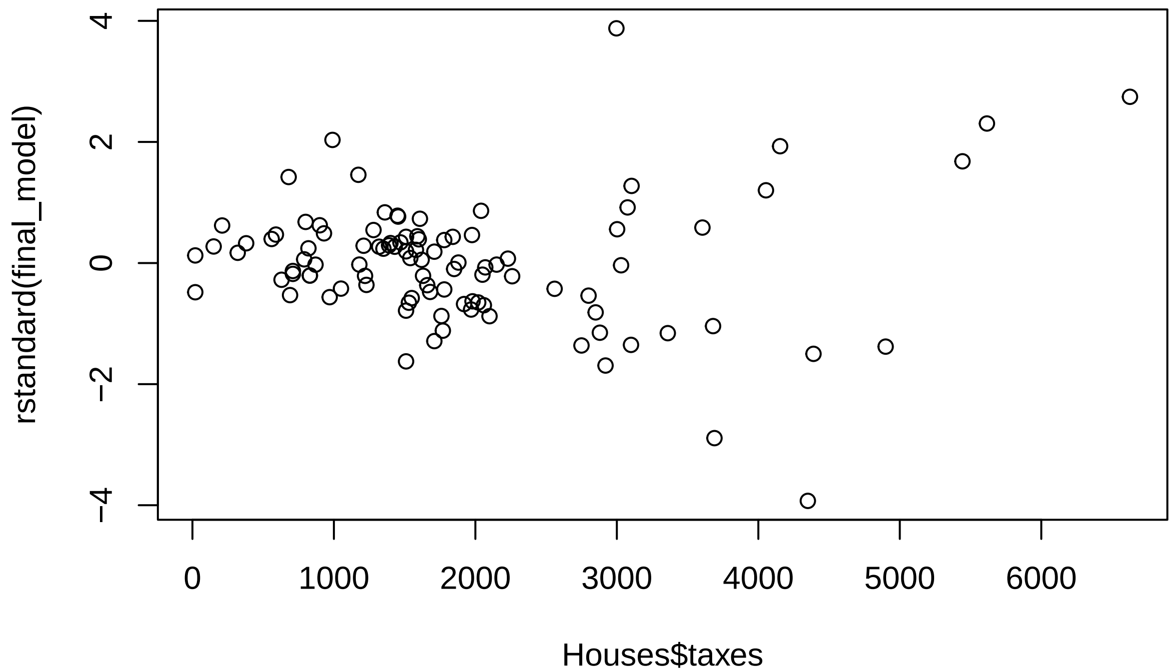

```
plot(Houses$size, rstandard(final_model))
```




```r
plot(Houses$new, rstandard(final_model))
```




```r
plot(Houses$taxes, rstandard(final_model))
```

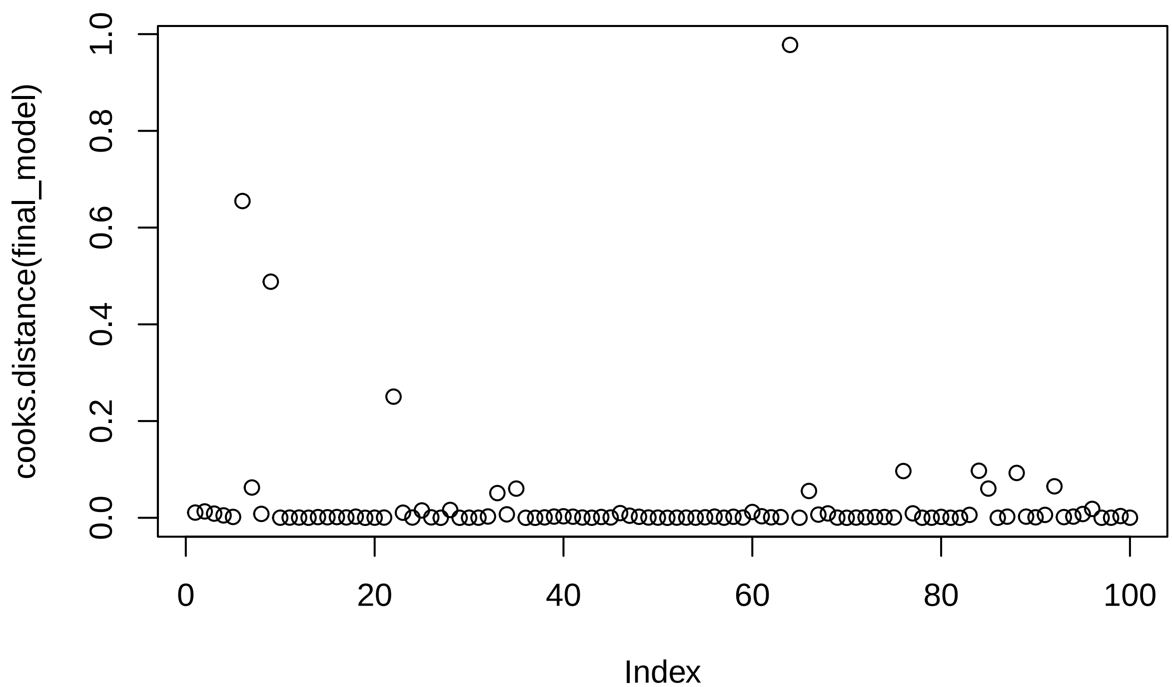



```r
cooks.distance(final_model)
```

```
##            1            2            3            4            5            6
## 1.078456e-02 1.315824e-02 8.619918e-03 4.920737e-03 1.809970e-03 6.549551e-01
##            7            8            9           10           11           12
## 6.260280e-02 8.048676e-03 4.881408e-01 2.251352e-04 4.384764e-04 4.574508e-04
##           13           14           15           16           17           18
## 1.373033e-04 1.413387e-03 1.092134e-03 1.535427e-03 7.640824e-04 2.338046e-03
##           19           20           21           22           23           24
## 3.353992e-06 2.591576e-04 4.928218e-04 2.504672e-01 1.054520e-02 5.572734e-04
##           25           26           27           28           29           30
## 1.521903e-02 9.290217e-04 1.117122e-05 1.623957e-02 1.715714e-05 1.929057e-04
##           31           32           33           34           35           36
## 2.102760e-04 2.706049e-03 5.111319e-02 7.116128e-03 6.032755e-02 4.707863e-06
##           37           38           39           40           41           42
## 2.096743e-06 8.211651e-04 2.543559e-03 3.066160e-03 2.284339e-03 5.586928e-04
##           43           44           45           46           47           48
## 1.000773e-04 1.632202e-03 6.683837e-04 9.673343e-03 4.360830e-03 2.072089e-03
##           49           50           51           52           53           54
## 5.963095e-04 3.482698e-04 4.483238e-05 2.986438e-04 3.838077e-04 2.225785e-04
##           55           56           57           58           59           60
## 9.783018e-04 2.430106e-03 1.929057e-04 2.134000e-03 4.334312e-04 1.193688e-02
##           61           62           63           64           65           66
## 3.182059e-03 1.000906e-03 1.423989e-03 9.777287e-01 1.129688e-04 5.544516e-02
##           67           68           69           70           71           72
```

```
## 6.597399e-03 9.100364e-03 3.650868e-04 1.889924e-05 4.178125e-04 1.000418e-03
##           73           74           75           76           77           78
## 1.479375e-03 1.576228e-03 6.414552e-04 9.663748e-02 9.139247e-03 8.238551e-05
##           79           80           81           82           83           84
## 1.557409e-04 1.833019e-03 8.669092e-05 2.053612e-05 6.025534e-03 9.721679e-02
##           85           86           87           88           89           90
## 6.042962e-02 3.290783e-07 2.359411e-03 9.271435e-02 2.393902e-03 9.988925e-04
##           91           92           93           94           95           96
## 6.013145e-03 6.508909e-02 1.424746e-03 2.469907e-03 7.812240e-03 1.833148e-02
##           97           98           99          100
## 1.146778e-05 1.453589e-04 3.549497e-03 2.261316e-04
```

```
plot(cooks.distance(final_model))
```




```
cooks_distance <- cbind(Houses$case,Houses$size,Houses$new,Houses$taxes,Houses$price,fitted(final_model)

final_model <- lm(price ~ size+new+taxes, data=Houses)
print(summary(final_model))
```

```
##
## Call:
## lm(formula = price ~ size + new + taxes, data = Houses)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
```

```
## -165.501  -25.426    1.449   20.536  168.747
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -21.353776  13.311487  -1.604  0.11196
## size          0.061704   0.012499   4.937 3.35e-06 ***
## new          46.373703  16.459019   2.818  0.00588 **
## taxes         0.037231   0.006735   5.528 2.78e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 47.17 on 96 degrees of freedom
## Multiple R-squared:  0.7896, Adjusted R-squared:  0.783
## F-statistic: 120.1 on 3 and 96 DF,  p-value: < 2.2e-16
```

```
cooks_distance <- data.frame(cooks_distance)
cooks_distance %>% filter(X8>0.9)
```

```
##    X1   X2 X3   X4  X5       X6        X7        X8
## 64 64 4050  0 4350 225 390.5007 -3.928075 0.9777287
```

```
# remove observation 64 because it has a cook distance greater than 0.9

final_model2 <- lm(price ~ size+new+taxes, subset(Houses, case != 64))
print(summary(final_model2))
```

```
##
## Call:
## lm(formula = price ~ size + new + taxes, data = subset(Houses,
##     case != 64))
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -140.558  -26.425    1.549   20.040  151.326
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -42.666802  13.238161  -3.223  0.00174 **
## size          0.082156   0.012470   6.589 2.46e-09 ***
## new          34.105507  15.428046   2.211  0.02946 *
## taxes         0.032732   0.006291   5.203 1.13e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 43.44 on 95 degrees of freedom
## Multiple R-squared:  0.8225, Adjusted R-squared:  0.8169
## F-statistic: 146.8 on 3 and 95 DF,  p-value: < 2.2e-16
```

```
glm(price ~ size+new+taxes, subset(Houses, case != 64), family=gaussian)
```

```
##
## Call:  glm(formula = price ~ size + new + taxes, family = gaussian,
```

```
##     data = subset(Houses, case != 64))
##
## Coefficients:
## (Intercept)         size          new        taxes
##   -42.66680      0.08216     34.10551      0.03273
##
## Degrees of Freedom: 98 Total (i.e. Null);  95 Residual
## Null Deviance:       1010000
## Residual Deviance: 179300    AIC: 1034
```

Having a new house (instead of old) changes the price of the house from 46 000 units to 34 000 units.

The R^2 increased from 78% to 82%.

Question 4, part (a)

```r
library(MASS)
library(tidyverse)

# Initial null model with no predictors
null_model <- glm(price ~ 1, data = Houses, family = Gamma(link = "log"))

# Full model with all predictors
full_model <- glm(price ~ size + new + baths + beds + taxes, data = Houses, family = Gamma(link = "log")

# Step-wise model selection using forward selection based on AIC
step_model <- stepAIC(null_model, scope = list(lower = null_model, upper = full_model),
                      direction = "forward", trace = FALSE)

summary(step_model)
```

```
##
## Call:
## glm(formula = price ~ taxes + size + new, family = Gamma(link = "log"),
##     data = Houses)
##
## Deviance Residuals:
##      Min        1Q    Median        3Q       Max
## -1.01549  -0.20745   0.02085   0.15776   0.69305
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 4.043e+00  7.797e-02  51.857  < 2e-16 ***
## taxes       2.216e-04  3.945e-05   5.619 1.88e-07 ***
## size        2.703e-04  7.321e-05   3.693 0.000368 ***
## new         1.920e-01  9.640e-02   1.992 0.049237 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for Gamma family taken to be 0.07633115)
##
##     Null deviance: 31.9401  on 99  degrees of freedom
## Residual deviance:  8.3536  on 96  degrees of freedom
## AIC: 1024.1
```

```
##
## Number of Fisher Scoring iterations: 5
```

Intercept: not plausible because no house has 0ftˆ2 and pays 0 taxes

taxes: For each one-monetary unit increase in taxes, the expected house price increases by a factor of exp(0.0002216), which is statistically significant at 0.001.

size (Coefficient: 2.703e-04): For each one-unit increase in size, the expected house price increases by a factor of exp(0.0002703), which is also statistically significant at 0.001.

new (Coefficient: 1.920e-01): New houses have their expected price about 21% higher than older houses, which is statistically significant (though to a mcuh lesser degree than size and tax) at 0.05.

The model's AIC is 1024.1, indicating its relative quality of fit.

Question 4, part (b)

```r
library(MASS)
library(tidyverse)

# Initial model
null_model_identity <- glm(price ~ 1, data = Houses, family = Gamma(link = "identity"))

# Full model with all predictors and identity link
full_model_identity <- glm(price ~ size + new + baths + beds + taxes, data = Houses, family = Gamma(lin

# Stepwise model selection using forward selection based on AIC with identity link
step_model_identity <- stepAIC(null_model_identity, scope = list(lower = null_model_identity, upper = fu
                        direction = "forward", trace = FALSE)

summary(step_model_identity)
```

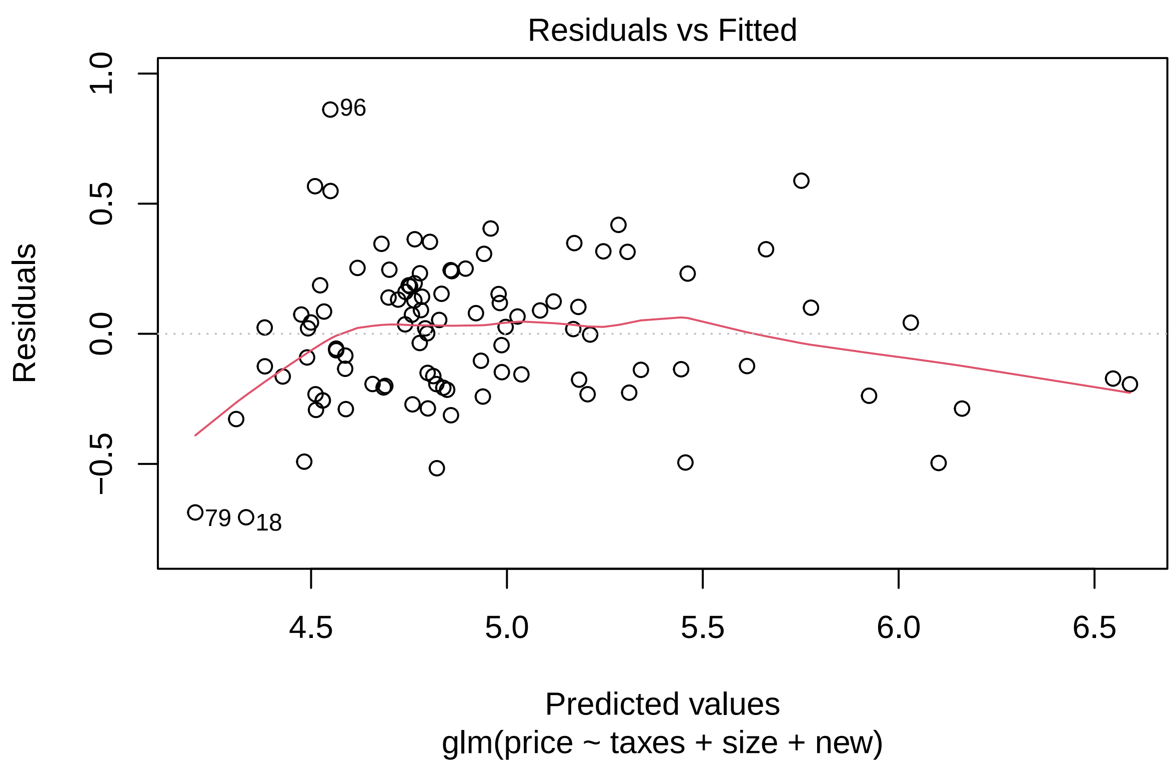
```
##
## Call:
## glm(formula = price ~ taxes + size + beds + baths, family = Gamma(link = "identity"),
##     data = Houses)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -0.8509  -0.2003  -0.0276   0.1515   0.5789
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  18.860815  13.711268   1.376  0.17219
## taxes         0.038452   0.005017   7.664 1.53e-11 ***
## size          0.065148   0.012452   5.232 9.98e-07 ***
## beds        -20.678022   6.286555  -3.289  0.00141 **
## baths         9.585793   6.491052   1.477  0.14304
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for Gamma family taken to be 0.06861316)
##
##     Null deviance: 31.9401  on 99  degrees of freedom
## Residual deviance:  6.6025  on 95  degrees of freedom
```

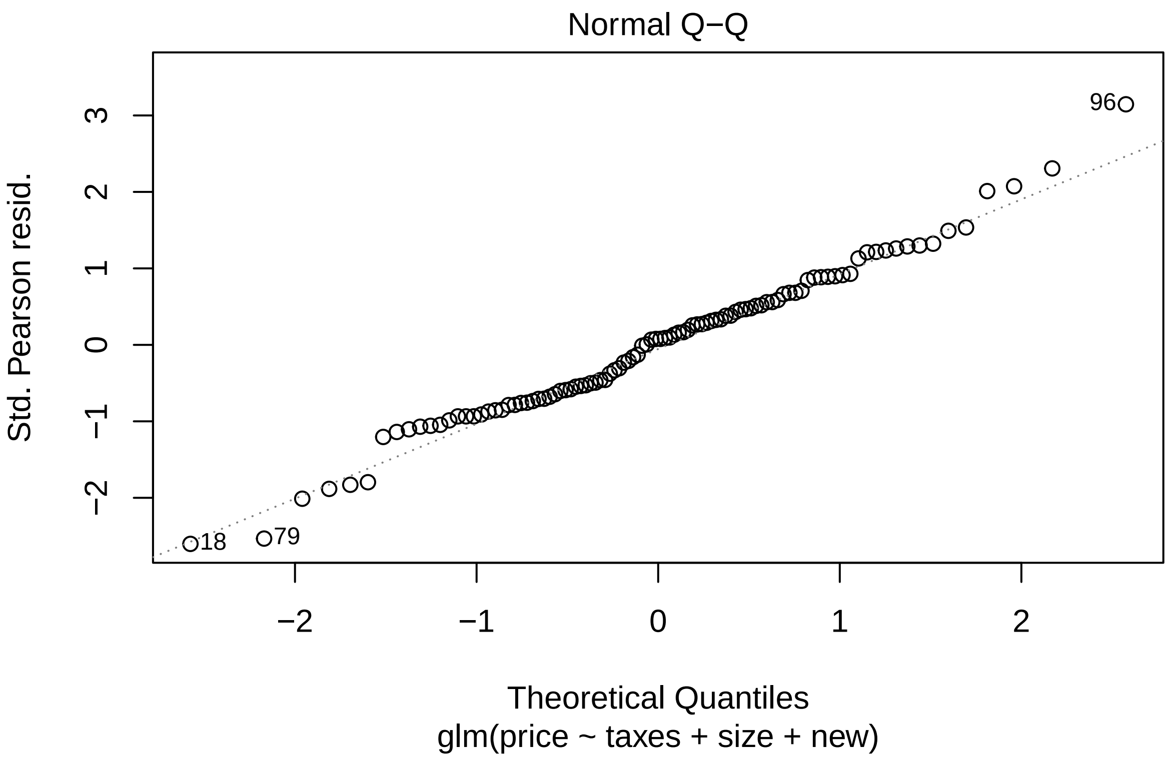
```
## AIC: 1002.3
##
## Number of Fisher Scoring iterations: 7
```

The Gamma model with an identity link function directly relates changes in predictors to absolute changes in house price, unlike the log link function, which relates to percentage changes. Both Gamma models in parts (a) and (b) show that taxes and size are significant, with the identity link also finding beds significant. Compared to the normal linear model from Question 3 part (b), the Gamma model with identity link has a lower AIC, suggesting a potentially better fit, and similar to the normal model, it interprets coefficients as absolute changes in price.

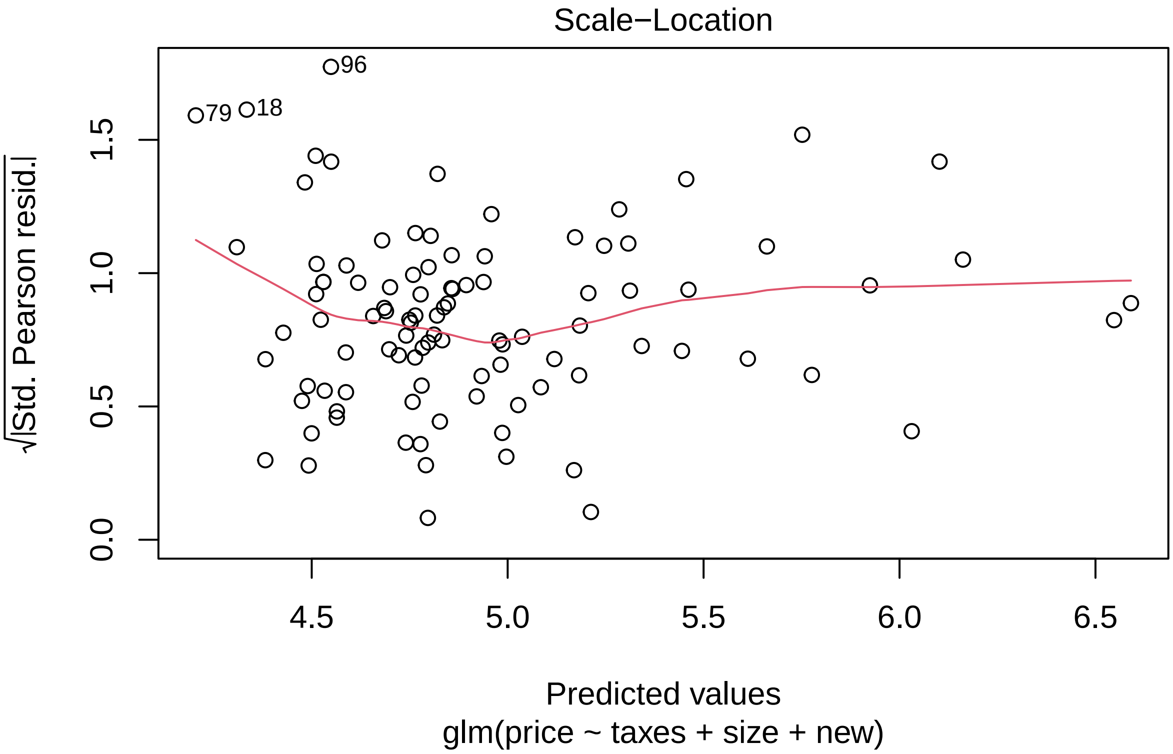
Question 4, part (c)

```r
# Diagnostics for the model with log link
plot(step_model, which = 1:6)
```

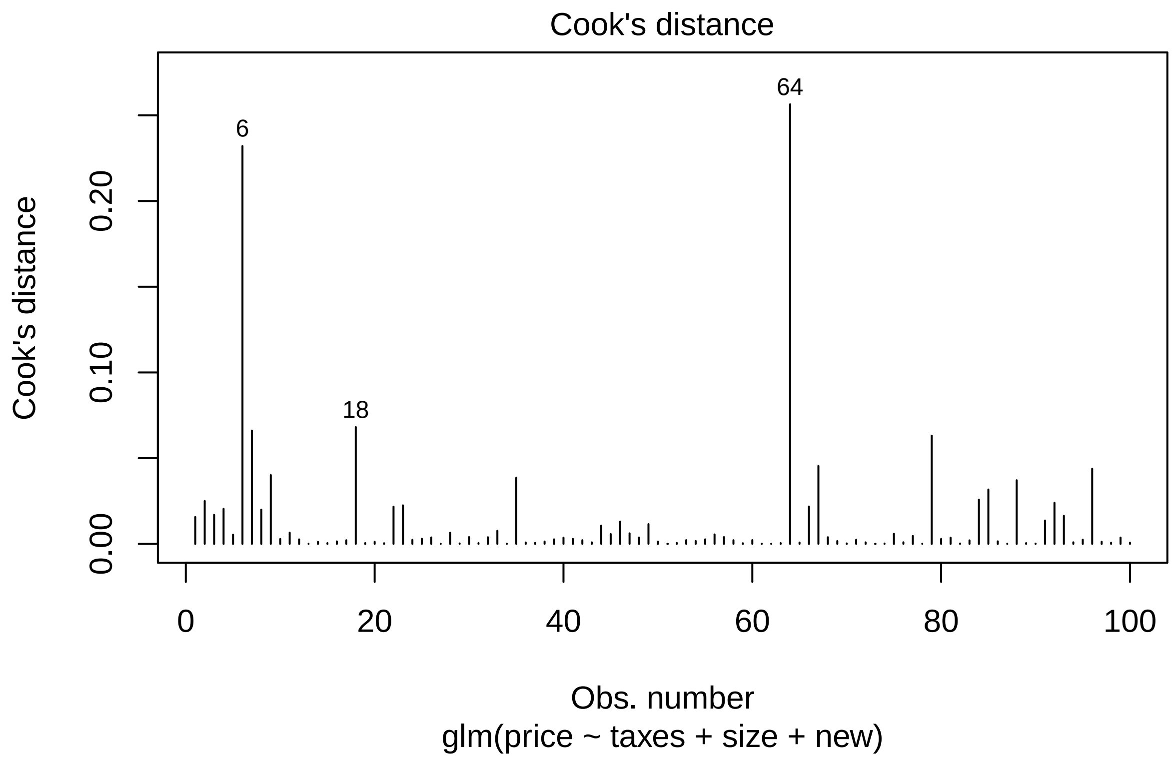


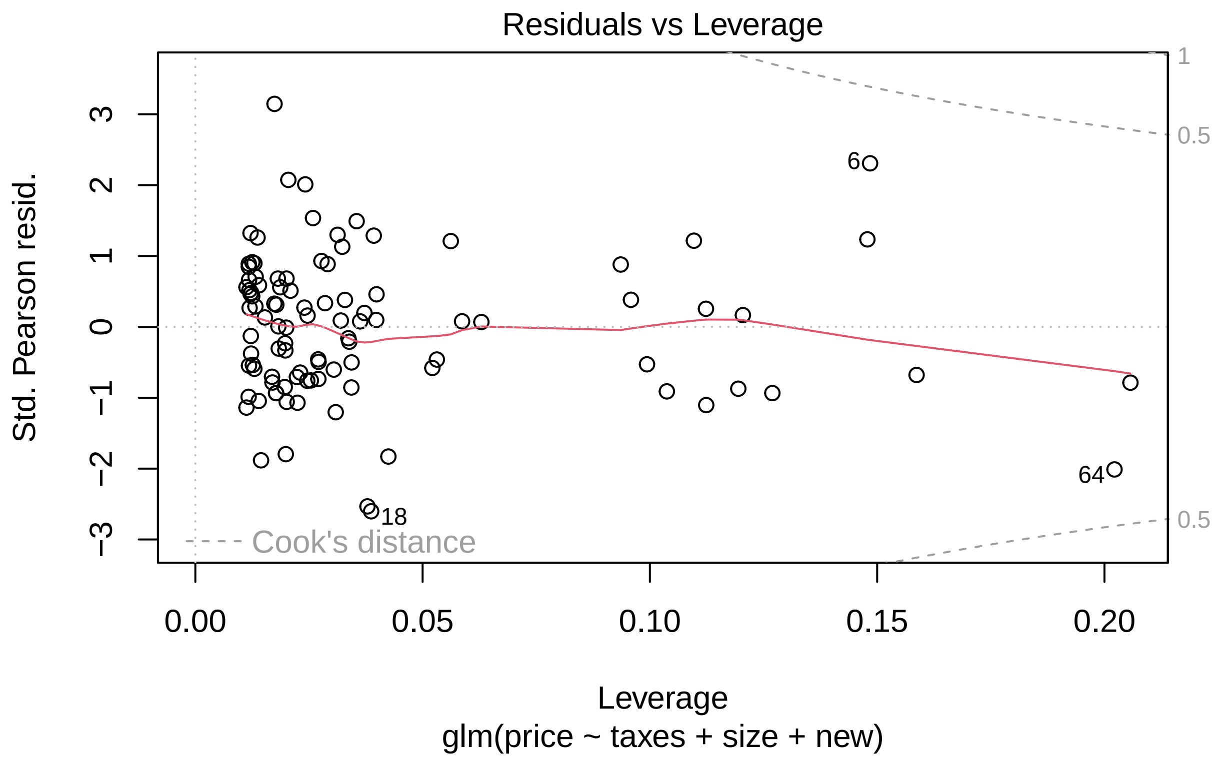


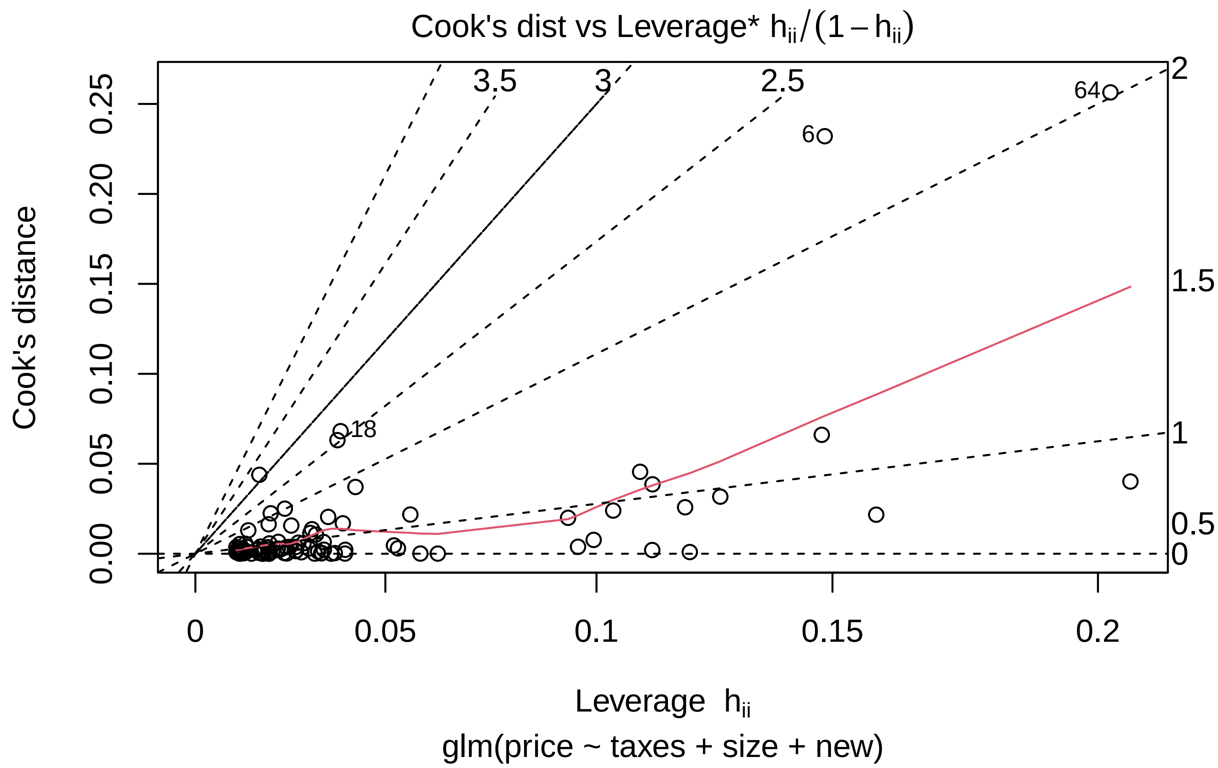
Normal Q−Q
Std. Pearson resid.
Theoretical Quantiles
glm(price ~ taxes + size + new)
96
18
79
−2
−1
0
1
2
3



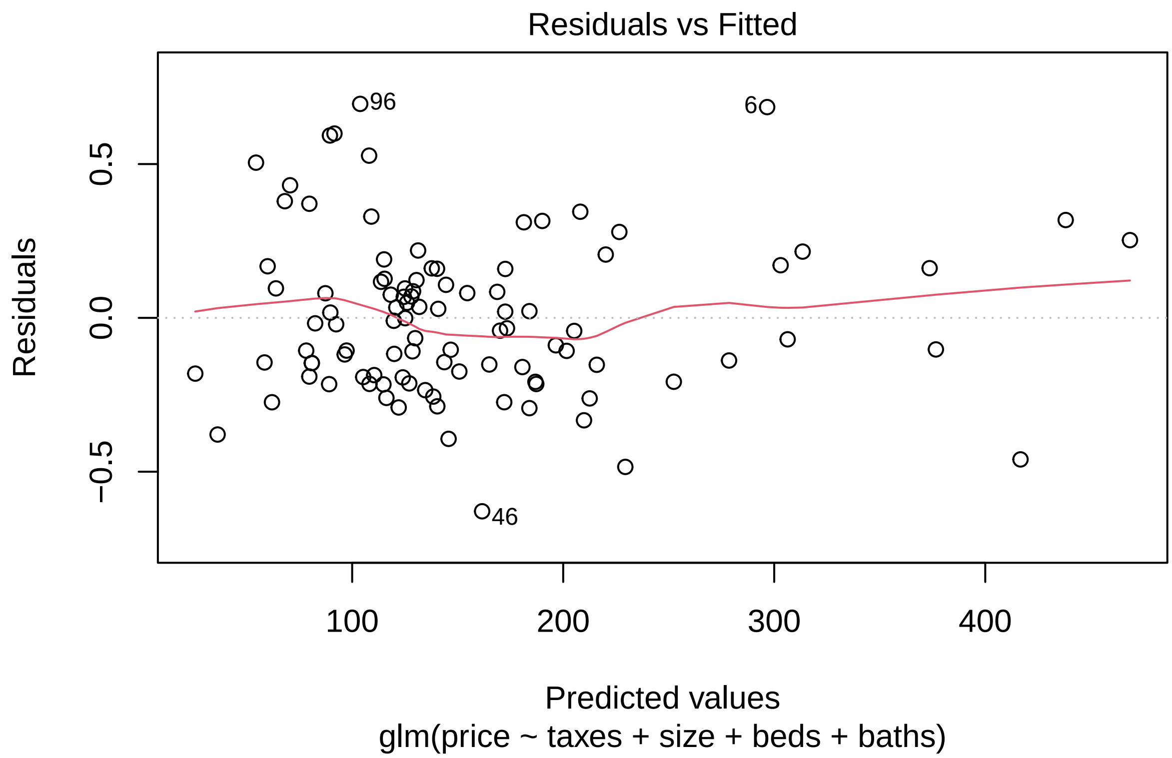
Scale−Location
96
79
18
√|Std. Pearson resid.|
0.0
0.5
1.0
1.5
4.5
5.0
5.5
6.0
6.5
Predicted values
glm(price ~ taxes + size + new)

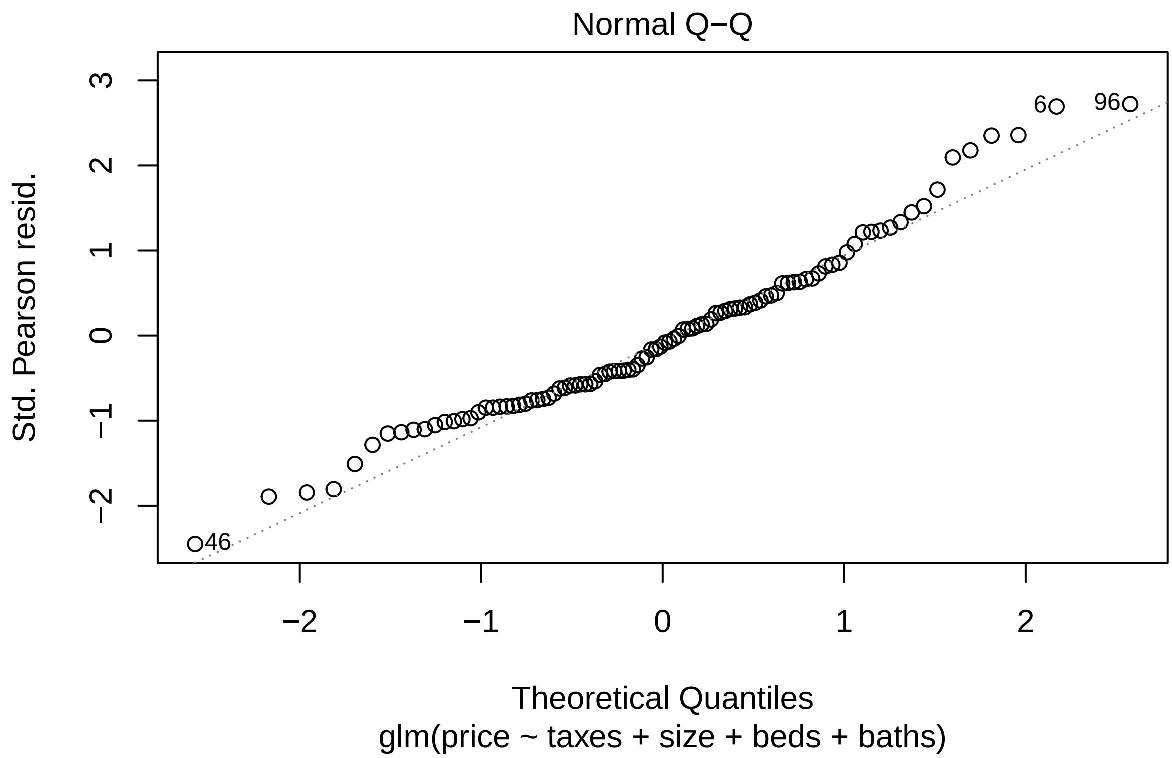

Cook's distance
Cook's distance
0.00
0.10
0.20
6
18
64
0
20
40
60
80
100
Obs. number
glm(price ~ taxes + size + new)

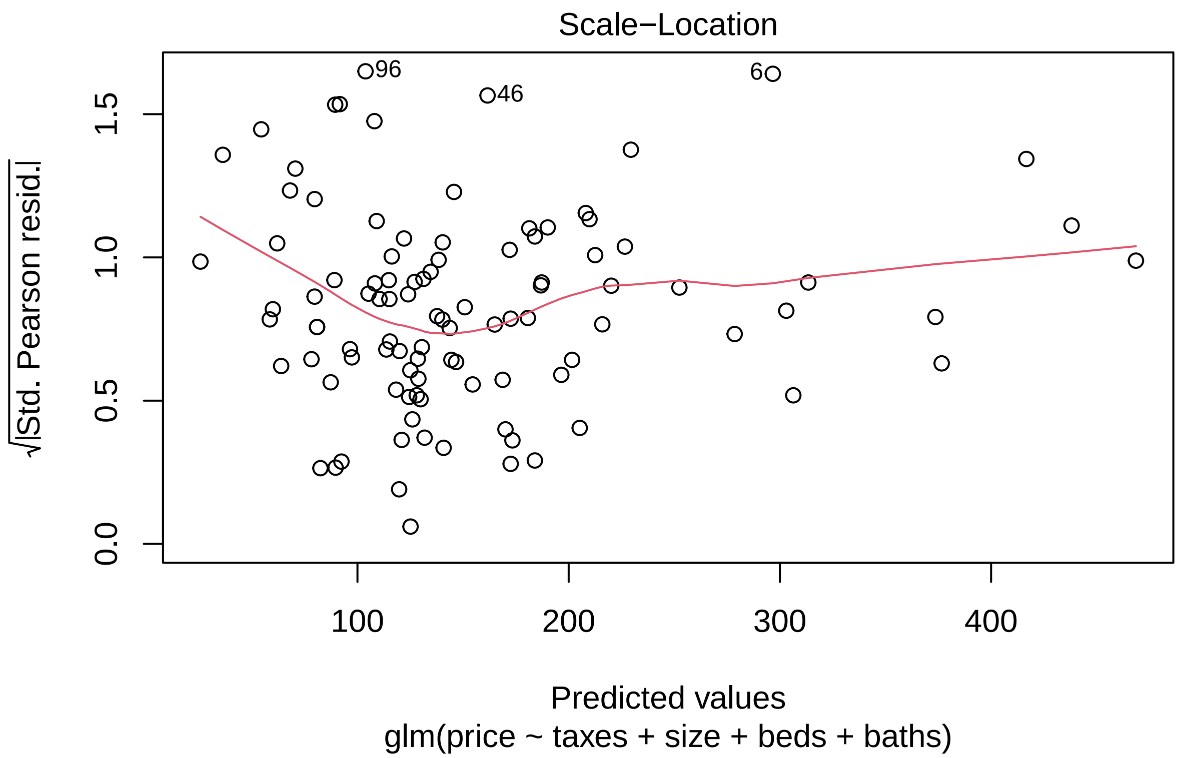

Residuals vs Leverage
Std. Pearson resid.
3
2
1
0
−1
−2
−3
1
0.5
0.5
6
64
18
Cook's distance
0.00
0.05
0.10
0.15
0.20
Leverage
glm(price ~ taxes + size + new)



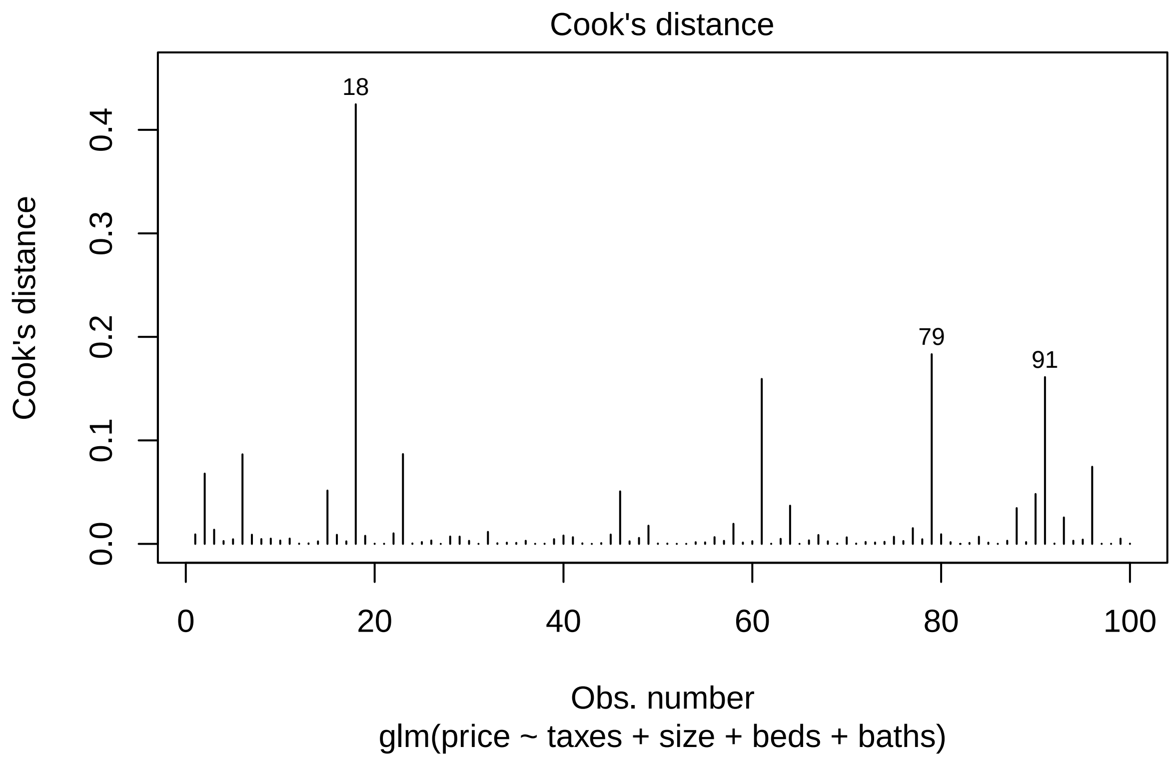

```r
# Diagnostics for the model with identity link
plot(step_model_identity, which = 1:6)
```

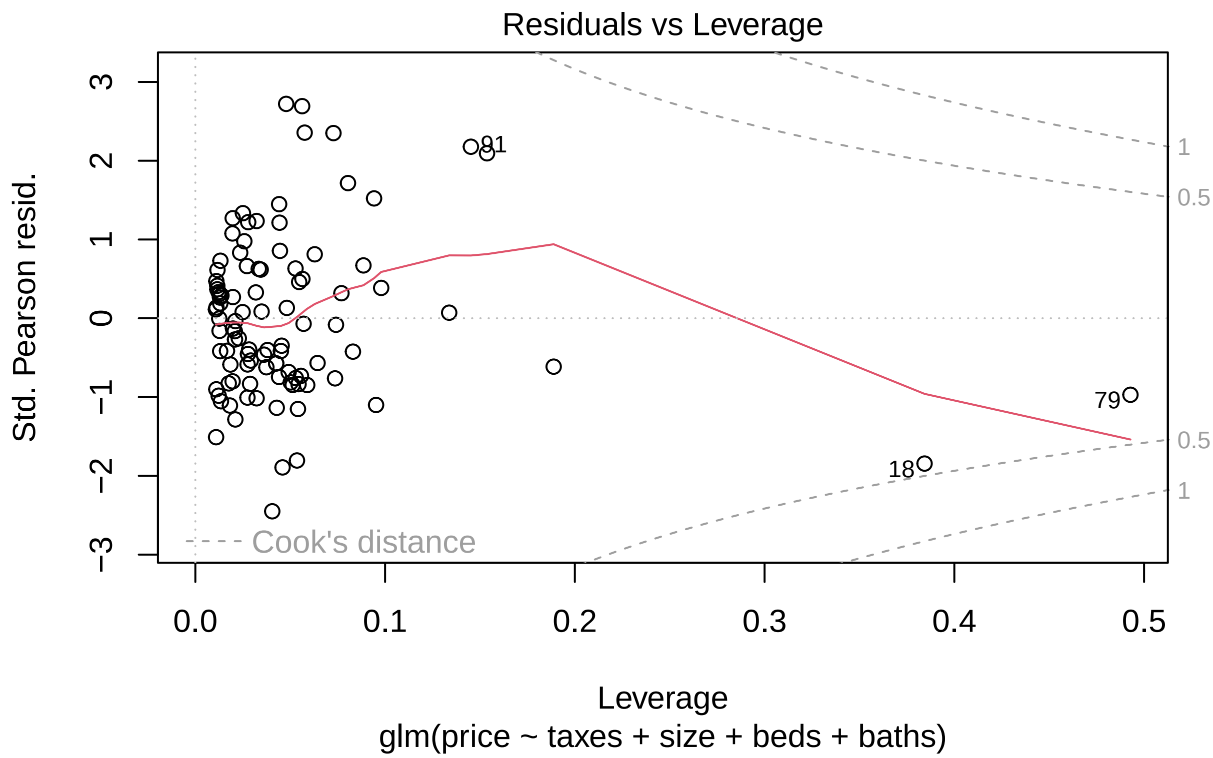

Residuals vs Fitted
Residuals
0.5
0.0
−0.5
96
6
46
100
200
300
400
Predicted values
glm(price ~ taxes + size + beds + baths)


Normal Q−Q
Std. Pearson resid.
Theoretical Quantiles
glm(price ~ taxes + size + beds + baths)
46
6
96
−2
−1
0
1
2
3

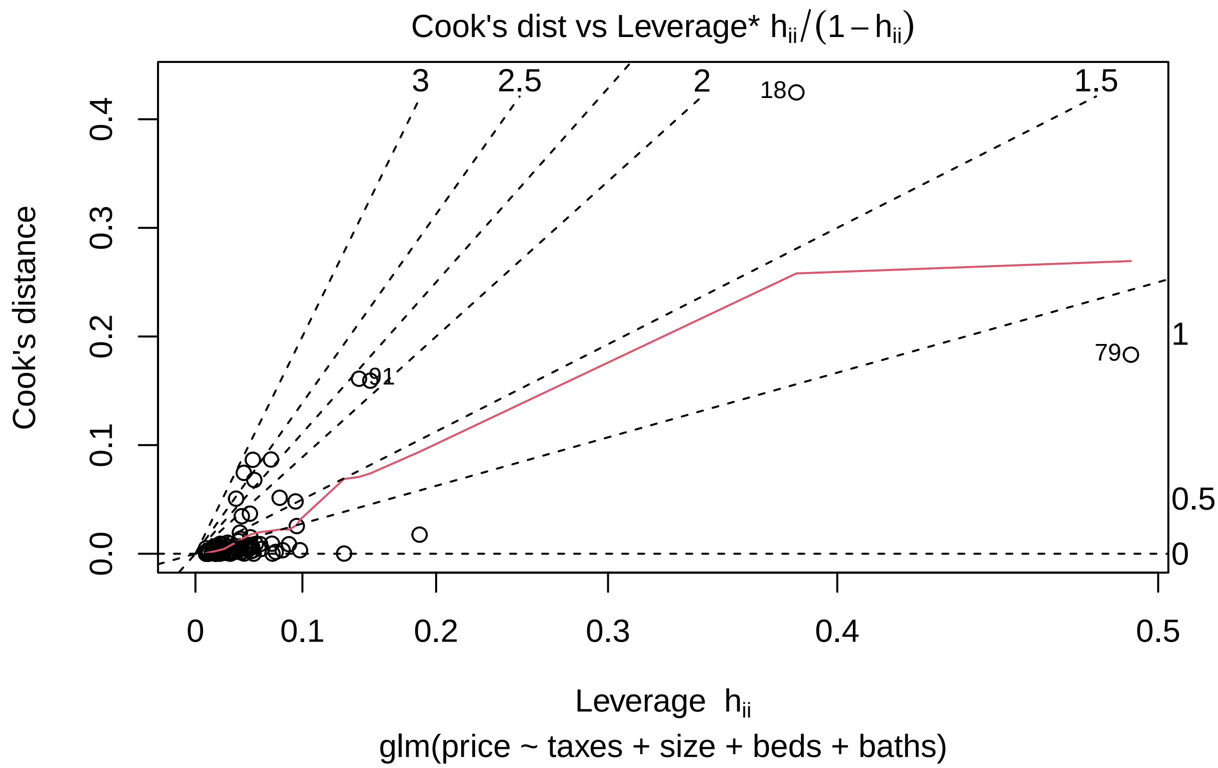


Scale−Location
96
46
6
1.5
1.0
0.5
0.0
√|Std. Pearson resid.|
100
200
300
400
Predicted values
glm(price ~ taxes + size + beds + baths)


Cook's distance
Cook's distance
0.0
0.1
0.2
0.3
0.4
18
79
91
0
20
40
60
80
100
Obs. number
glm(price ~ taxes + size + beds + baths)



Residuals vs Leverage
Std. Pearson resid.
3
2
1
0
−1
−2
−3
0.0
0.1
0.2
0.3
0.4
0.5
Leverage
glm(price ~ taxes + size + beds + baths)
Cook's distance
1
0.5
0.5
1
91
79
18




```r
# Goodness of fit comparison
cat("Goodness of Fit for Model with Log Link:\n")
```

```
## Goodness of Fit for Model with Log Link:
```

```r
cat("AIC:", AIC(step_model), "\n")
```

```
## AIC: 1024.103
```

```r
cat("Goodness of Fit for Model with Identity Link:\n")
```

```
## Goodness of Fit for Model with Identity Link:
```

```r
cat("AIC:", AIC(step_model_identity), "\n")
```

```
## AIC: 1002.288
```