Homework 02 Solutions

2022-09-08

36-617: Applied Linear Models
Fall 2022
Solutions

```
library(arm)  ## includes lme4, MASS, Matrix
library(ggplot2); theme_set(theme_bw())
library(gridExtra) ## to arrange ggplots...
```

Problem 1

Let $y = X\beta + \epsilon$, where $y = (y_1, \ldots, y_n)^T$ is an $n \times 1$ column vector, X is an $n \times (p+1)$ matrix whose first column is all 1's, $\beta = (\beta_0, \ldots, \beta_p)^T$ is a $(p+1) \times 1$ column vector, and $\epsilon = (\epsilon_1, \ldots, \epsilon_n)^T \sim N(0, \sigma^2 I)$ is an $n \times 1$ random column vector, following a multivariate Normal distribution with mean vector 0 and variance-covariance matrix $\sigma^2 I$, where I is the $n \times n$ identity matrix.

Problem 1(a)

Let

$$U = \begin{bmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{bmatrix}$$

be an $n \times 1$ column of all 1's.

i Using the definition of the hat matrix H, show $HX = X$

ii Using your result in (i), show $HU = U$

Part (i)

$HX = [X(X^TX)^{-1}X^T]X = X(X^TX)^{-1}(X^TX) = X$

Part (ii)

Two ways to think about it:

- ***In words:*** *Since $HX = X$ we know that all of the columns of X are unchanged when we premultiply X by H. In particular, since the first column of X is U, we know U will be unchanged when we premultiply by H: $HU = U$.*
- ***In math:*** *Let*

$$E_1 = \begin{bmatrix} 1 \\ 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix},$$

the $n \times 1$ column vector with a 1 in the first row and 0's in the remaining $n-1$ rows. Since the first column of X is U, we know $XE_1 = U$. Therefore,

$$HU = H(XE_1) = (HX)E_1 = XE_1 = U$$

Problem 1(b)

Use properties of the hat matrix $H = X(X^TX)^{-1}X^T$ to show that the column vector

$$y - \hat{y} = \begin{bmatrix} y_1 - \hat{y}_1 \\ y_2 - \hat{y}_2 \\ \vdots \\ y_n - \hat{y}_n \end{bmatrix}$$

can be written as $y - \hat{y} = (I - H)y$.

Since $Hy = \hat{y}$ and $Iy = y$,

$$y - \hat{y} = Iy - Hy = (I - H)y$$

Problem 1(c)

For any $n \times 1$ column vector

$$a = \begin{bmatrix} a_1 \\ a_2 \\ \vdots \\ a_n \end{bmatrix}$$

we know that $a^TU = \sum_{i=1}^n a_i$. Use this fact, together with your results in (a) and (b), to show that

$$\sum_{i=1}^n (y_i - \hat{y}_i) = 0.$$

$$\sum_{i=1}^n (y_i - \hat{y}_i) = \underbrace{(y-\hat{y})^T}_{1\times n} \underbrace{U}_{n\times 1} = \underbrace{[(I-H)y]^T}_{1\times n} \underbrace{U}_{n\times 1} = \underbrace{y^T}_{1\times n} \underbrace{(I-H)}_{n\times n} \underbrace{U}_{n\times 1} = \underbrace{y^T}_{1\times n} \underbrace{(U-U)}_{n\times 1} = \underbrace{Y^T}_{1\times n} \underbrace{0}_{n\times 1} = \underbrace{0}_{1\times 1}$$

Aside: Hence y and $\hat{y}$ have the same sample mean $\overline{y}$, and the residuals vs fitted plot should always be "balanced", in some sense, around the horizontal line at $\hat{e} = 0$.

Problem 2

Let $y = X\beta + \epsilon$, where $y = (y_1, \ldots, y_n)^T$ is an $n \times 1$ column vector, X is an $n \times (p+1)$ matrix whose first column is all 1's, $\beta = (\beta_0, \ldots, \beta_p)^T$ is a $(p+1) \times 1$ column vector, and $\epsilon = (\epsilon_1, \ldots, \epsilon_n)^T \sim N(0, \sigma^2 I)$ is an $n \times 1$ random column vector, following a multivariate Normal distribution with mean vector 0 and variance-covariance matrix $\sigma^2 I$, where I is the $n \times n$ identity matrix.

Problem 2(a)

Use properties of the hat matrix $H = X(X^TX)^{-1}X^T$ and the multivariate Normal distribution as discussed in class, to show[1]

$$\hat{e} \sim N(0, (I - H)\sigma^2)$$

[1] Very similar to problem 1(b).

where $\hat{e}$ is the column vector $\hat{e} = y - \hat{y}$.

We will use several facts from lecture. First, let $W \sim N(\mu, \Sigma)$ be an arbitrary multivariate normal random vector and A a matrix whose second dimension is the same as the length of W, so that AW is defined. Then we have

$$AW \sim N(A\mu, A\Sigma A^T) \tag{1}$$

Additionally, we have from lecture that

$$\begin{align} HX\beta &= X\beta \tag{2} \\ (I-H)(I-H)^T &= (I-H)(I-H) = (I-H) \tag{3} \\ y &\sim N(X\beta, \sigma^2 I) \tag{4} \end{align}$$

Let $\hat{y}$ be the $n \times 1$ column vector of fitted values.

$$\begin{align*} \hat{e} &= y - \hat{y} \\ &= y - Hy \\ &= (I-H)y \\ &\sim N((I-H)X\beta, (I-H)\sigma^2 I(I-H)^T) && \text{(by facts (1) and (4))} \\ &= N(0, \sigma^2(I-H)(I-H)^T) && \text{(by fact (2))} \\ &= N(0, \sigma^2(I-H)) && \text{(by fact (3))} \end{align*}$$

Problem 2(b)

Let H be the hat matrix for the multivariate regression model $y = X\beta + \epsilon$ as in part (a), and let H_1 be the hat matrix for the intercept-only model $y = \beta_0 + \epsilon$.

i Show that the fitted values $\hat{y}$ for the intercept-only model is an $n \times 1$ column vector, all of whose entries are $\overline{y}$, that is,

$$\hat{y} = \begin{bmatrix} \hat{y}_1 \\ \vdots \\ \hat{y}_n \end{bmatrix} = \begin{bmatrix} \overline{y} \\ \vdots \\ \overline{y} \end{bmatrix} \tag{$*$}$$

(where the first "=" is the definition of $\hat{y}$ and the second "=" is what I want you to show).

ii Find a simple expression, in terms of (some or all of) y, I, H and H_1, for the sample covariance

$$\text{Cov}(y, \hat{y}) = \frac{1}{n}\sum_{i=1}^{n}(y_i - \overline{y})(\hat{y}_i - \overline{y}).$$

*(**Hint:** We can rewrite Cov$(y, \hat{y}) = \frac{1}{n}(y - \overline{y})^T(\hat{y} - \overline{y})$, where $\hat{y}$ is the column vector of fitted values from $y = X\beta + \epsilon$ and, abusing notation slightly, $\overline{y}$ is the column vector in $(*)$ above, i.e. the fitted values from the intercept-only model $y = \beta_0 + \epsilon$.)*

iii Continue along the lines of the calculations in part (ii) to show that the sample correlation between y and $\hat{y}$ can be written as

$$\text{Corr}(y, \hat{y}) = \sqrt{\frac{SS_{reg}}{SST}}$$

and hence R^2 for the regression model $y = X\beta + \epsilon$ really is the squared correlation between y and $\hat{y}$:

$$R^2 = \text{Corr}(y, \hat{y})^2 \ .$$

Part (i)

In the intercept-only model, the design matrix X is an $n \times 1$ matrix in which each entry is 1. This can be used to derive the hat matrix H_1:

$$\begin{aligned}
X^T X &= [n]_{1\times 1} && \text{(the } 1 \times 1 \text{ matrix whose only entry is } n\text{)} \\
\implies (X^T X)^{-1} &= [1/n]_{1\times 1} && \text{(the } 1 \times 1 \text{ matrix whose only entry is } 1/n\text{)} \\
\implies X(X^T X)^{-1} X^T &= \frac{1}{n} X X^T \\
&= \frac{1}{n} \begin{bmatrix} 1 & \dots & 1 \\ \vdots & \ddots & \vdots \\ 1 & \dots & 1 \end{bmatrix}_{n \times n} \\
&= H_1 && \text{(by the definition of the hat matrix)}
\end{aligned}$$

For any i in $1, 2, \dots, n$, we derive $\hat{y}_i$ by multiplying the ith row of H_1 by the vector y. In other words, for the intercept-only model, $\hat{y}_i = \frac{1}{n} \sum_{i=1}^{n} y_i = \overline{y}$ for all i.

Part (ii)

To avoid using $\overline{y}$ to denote both a scalar and a vector, let's immediately use the fact that $H_1 y$ is equal to the $n \times 1$ vector $\begin{bmatrix} \overline{y} & \overline{y} & \dots & \overline{y} \end{bmatrix}^T$, as shown in the previous problem. Rewriting the covariance in vector form, per the problem suggestion, we have:

$$\begin{aligned}
\operatorname{Cov}(y, \hat{y}) &= \frac{1}{n} \sum_{i=1}^{n} (y_i - \overline{y})(\hat{y}_i - \overline{y}) \\
&= \frac{1}{n} (y - H_1 y)^T (\hat{y} - H_1 y) \\
&= \frac{1}{n} (y - H_1 y)^T (Hy - H_1 y) \\
&= \frac{1}{n} [(I - H_1) y]^T (H - H_1) y \\
&= \frac{1}{n} y^T (I - H_1)^T (H - H_1) y && \text{(since } (AB)^T = B^T A^T\text{)} \\
&= \frac{1}{n} y^T (H - H_1 - H_1^T H + H_1^T H_1) y \\
&= \frac{1}{n} y^T (H - H_1 H) y && \text{(since } H_1^T = H_1 \text{ and } H_1 H_1 = H_1\text{)} \\
&= \frac{1}{n} y^T (H - H_1) y && \text{(since } H_1 H = H H_1 = H_1\text{)}
\end{aligned}$$

Part (iii)

Recall that the sample correlation is the standardized sample covariance, defined as

$$
\begin{aligned}
widehat\text{Corr}(y,\hat{y}) &= \frac{\hat{\text{Cov}}(y,\hat{y})}{\sqrt{\hat{\sigma}_y^2\hat{\sigma}_{\hat{y}}^2}} \\
&= \frac{\sum_{i=1}^n (y_i-\overline{y})(\hat{y}_i-\overline{y})}{\sqrt{\sum_{i=1}^n (y_i-\overline{y})^2 \sum_{i=1}^n (\hat{y}_i-\overline{y})^2}} \\
&= \frac{(y-H_1y)^T(\hat{y}-H_1y)}{\sqrt{(y-H_1y)^T(y-H_1y)(\hat{y}-H_1y)^T(\hat{y}-H_1y)}} \\
&= \frac{y^T(H-H_1)y}{\sqrt{y^T(I-H_1)^T(I-H_1)yy^T(H-H_1)^T(H-H_1)y}} && \text{(using the previous problem)} \\
&= \frac{\sqrt{y^T(H-H_1)y}}{\sqrt{y^T(I-H_1)^T(I-H_1)y}} \\
&= \frac{\sqrt{y^T(H-H_1)y}}{\sqrt{y^T(I-H_1)y}} && \text{(using a version of fact (3))} \\
&= \sqrt{\frac{SS_{reg}}{SST}} \\
&= \sqrt{R^2} && \text{(by the definition of } R^2\text{)}
\end{aligned}
$$

And therefore, $R^2 = \widehat{\text{Corr}}(y,\hat{y})^2$.

Problem 2(c)

Show that $\hat{e}$ and $\hat{y}$ have sample correlation 0, and hence a scatter plot of $\hat{e}$ vs $\hat{y}$ should show no increasing or decreasing overall trend.

The sample correlation is 0 iff the sample covariance is 0, so let's just concern ourselves with the sample covariance and not worry about the denominator in the correlation.

Recall that the residuals sum to 0 by construction, which also means that their sample mean is 0. Hence:

$$
\begin{aligned}
\hat{\text{Cov}}(\hat{e},\hat{y}) &= \frac{1}{n}\sum_{i=1}^n (\hat{e}_i - 0)(\hat{y}_i-\overline{y}) \\
&= \frac{1}{n}(\hat{e})^T(\hat{y}-\overline{y}) \\
&= \frac{1}{n}(y-\hat{y})^T(\hat{y}-\overline{y}) \\
&= \frac{1}{n}[(I-H)y]^T(H-H_1)y \\
&= \frac{1}{n}y^T(I-H)(H-H_1)y && \text{(since } (I-H)^T=(I-H)\text{)} \\
&= \frac{1}{n}y^T(H-H_1-HH+HH_1)y \\
&= \frac{1}{n}y^T(H-H_1-H+H_1)y && \text{(since } HH_1=H_1\text{)} \\
&= 0
\end{aligned}
$$

Aside: It is interesting to note that the results of problem #1 (c) and problem #2 (b) and (c) did not depend at all on the normality assumption, or even on whether the model fits the data well or not. In other words, even if the fit is terrible, it will still be true that the sum of $\hat{e}$ is zero, $R^2 = [Corr(y, \hat{y})]^2$, and the sample correlation between $\hat{e}$ and $\hat{y}$ is zero. The only thing that matters is that the model have an intercept, i.e. the X matrix should have a column of 1's.

Problem 3

[Based on Gelman & Hill. Ch 3, #1, p. 49] The file `pyth.dat`, in the same folder as this hw, contains outcome `y` and inputs `x1`, `x2` for 40 data points, with a further 20 points with the inputs but no observed outcome (for this problem we will ignore these last 20 points). Save the file to your working directory and read it into R using the `read.table()` function.

Problem 3(a)

Fit the two models

$$\begin{aligned} \textbf{M1:} \quad \texttt{y} &= \beta_0 + \beta_1\ \texttt{x1} + \varepsilon \\ \textbf{M2:} \quad \texttt{y} &= \beta_0 + \beta_1\ \texttt{x2} + \varepsilon \end{aligned}$$

Which model provides a better fit for `y`? Why?

```
gh.data <- read.table("pyth.dat",header=T)
gh.data <- gh.data[apply(gh.data,1,function(x) {!any(is.na(x))}),]
str(gh.data)
```

```
## 'data.frame':    40 obs. of  3 variables:
##  $ y : num  15.68 6.18 18.1 9.07 17.97 ...
##  $ x1: num  6.87 4.4 0.43 2.73 3.25 5.3 7.08 9.73 4.51 6.4 ...
##  $ x2: num  14.09 4.35 18.09 8.65 17.68 ...
```

```
M1 <- lm(y ~ x1, data=gh.data)
M2 <- lm(y ~ x2, data=gh.data)

summary(M1)
```

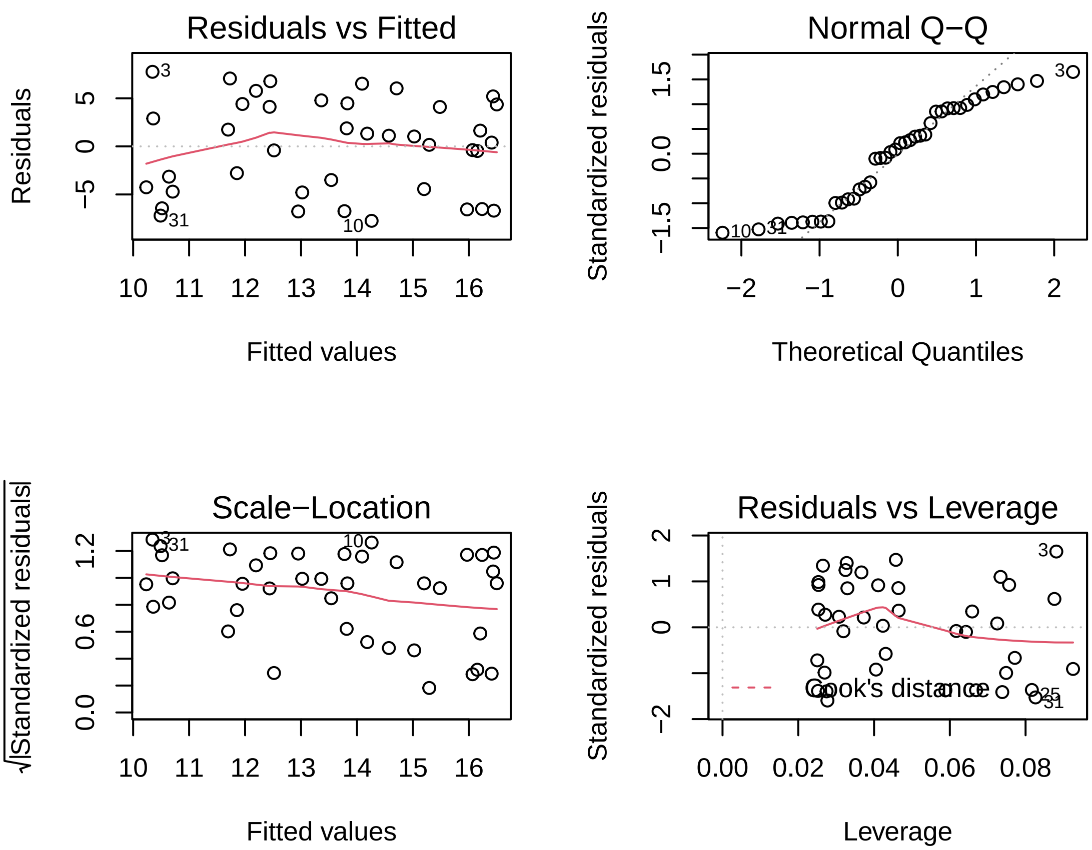
```
##
## Call:
## lm(formula = y ~ x1, data = gh.data)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -7.7409 -4.5056  0.7114  4.3739  7.7547
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  10.0633     1.5526   6.481 1.25e-07 ***
## x1            0.6559     0.2499   2.625   0.0124 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4.921 on 38 degrees of freedom
## Multiple R-squared:  0.1535, Adjusted R-squared:  0.1312
## F-statistic:  6.89 on 1 and 38 DF,  p-value: 0.01242
```

```
summary(M2)
```

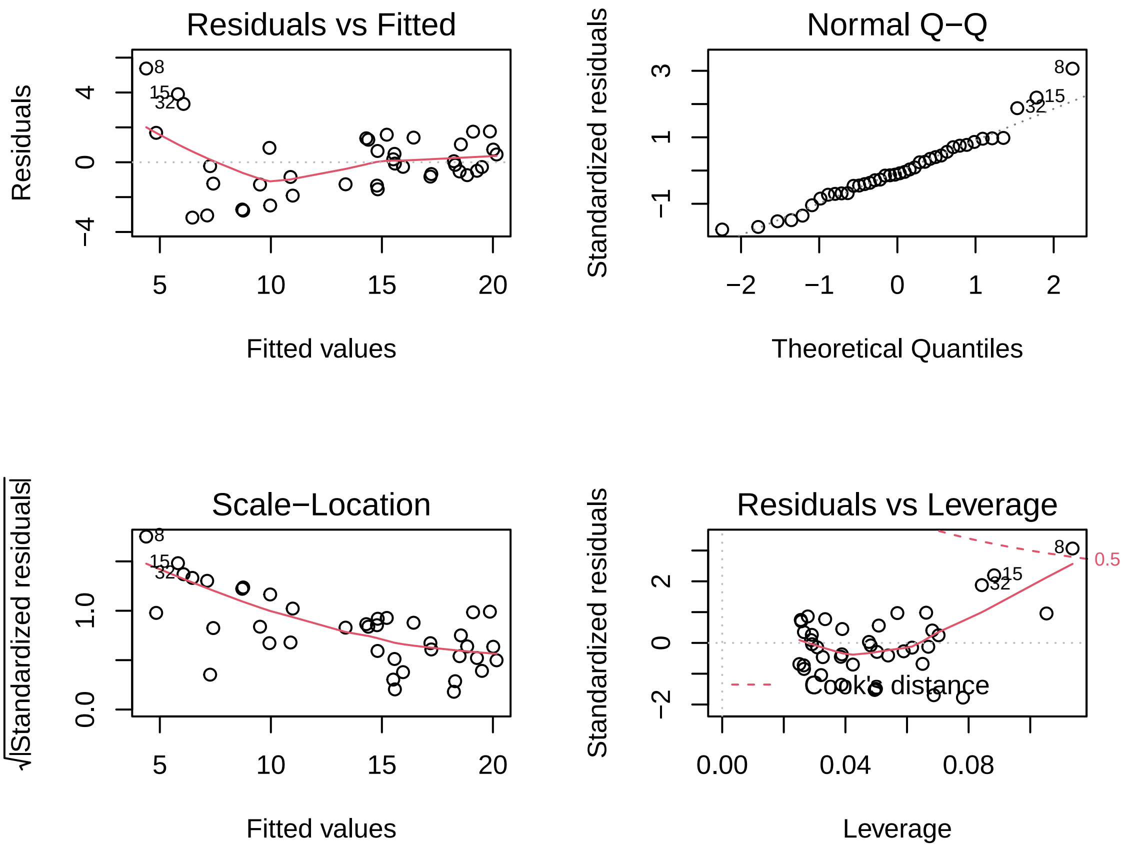
```
##
## Call:
```

```
## lm(formula = y ~ x2, data = gh.data)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -3.1751 -1.2352 -0.1867  1.0899  5.3755
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  3.78532    0.66037   5.732 1.33e-06 ***
## x2           0.83223    0.05017  16.589  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.863 on 38 degrees of freedom
## Multiple R-squared:  0.8787, Adjusted R-squared:  0.8755
## F-statistic: 275.2 on 1 and 38 DF,  p-value: < 2.2e-16
```

```
##
par(mfrow=c(2,2))
plot(M1)
```




```
plot(M2)
```



The R^2 is much lower for M1 (0.1535) than for M2 (0.8787). Neither set of residual diagnostic plots looks great: the residual vs fitted and scale-location plots somewhat favor M1, and the QQ plots somewhat favor M2. The Cook's distances are a bit better for M1 also. For R^2 and normality of residuals, I prefer M2.

Problem 3(b)

Construct new variables `y2 = y`^{2}, `x12 = x1`^{2}, and `x22 = x2`^{2} and fit the models

$$\begin{aligned} \textbf{M3:} \quad \texttt{y2} &= \beta_0 + \beta_1 \ \texttt{x12} + \varepsilon \\ \textbf{M4:} \quad \texttt{y2} &= \beta_0 + \beta_1 \ \texttt{x22} + \varepsilon \end{aligned}$$

Compare the fits of these two models to the models in part (a). Which fits best? Why?

```r
attach(gh.data)
y2 <- y^2
x12 <- x1^2
x22 <- x2^2
detach()
M3 <- lm(y2 ~ x12,data=gh.data)
M4 <- lm(y2 ~ x22,data=gh.data)

summary(M3)
```

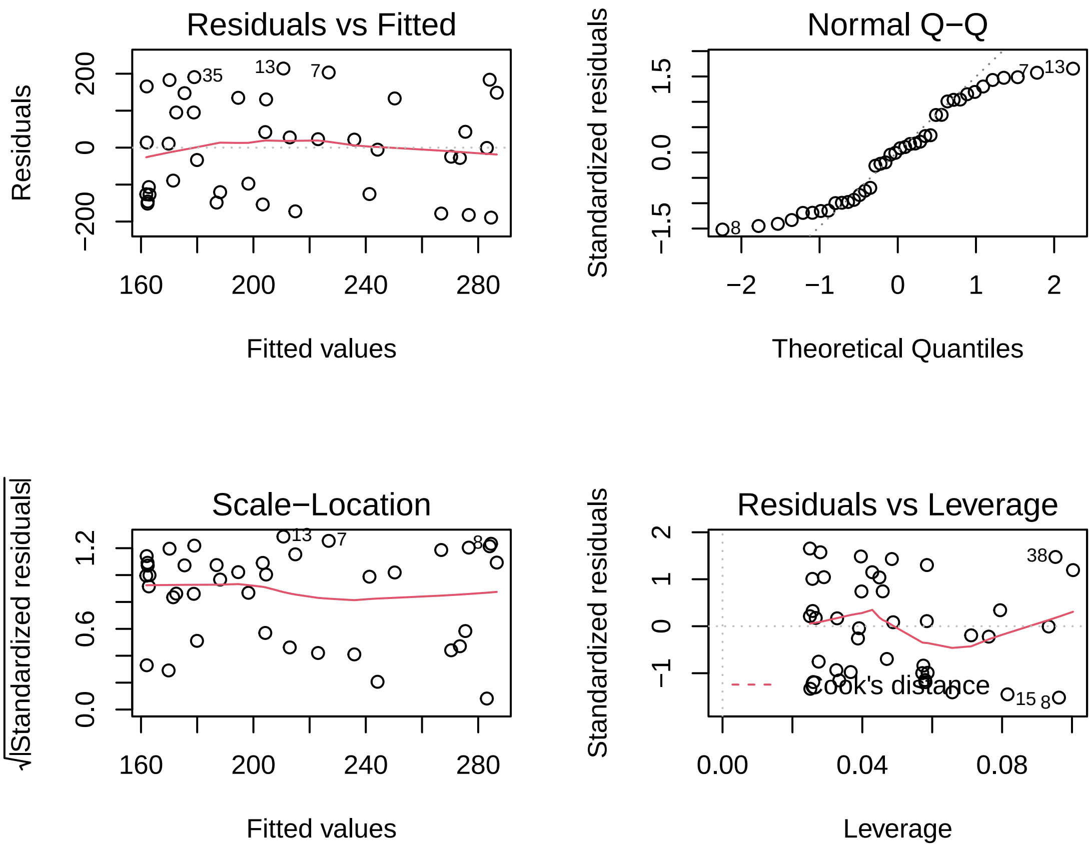
```
##
```

```
## Call:
## lm(formula = y2 ~ x12, data = gh.data)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -189.324 -125.674    4.988  131.052  214.089
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 161.7831    31.7873   5.090    1e-05 ***
## x12           1.2971     0.6242   2.078   0.0445 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 131.1 on 38 degrees of freedom
## Multiple R-squared:  0.102,  Adjusted R-squared:  0.07841
## F-statistic: 4.318 on 1 and 38 DF,  p-value: 0.04452
```

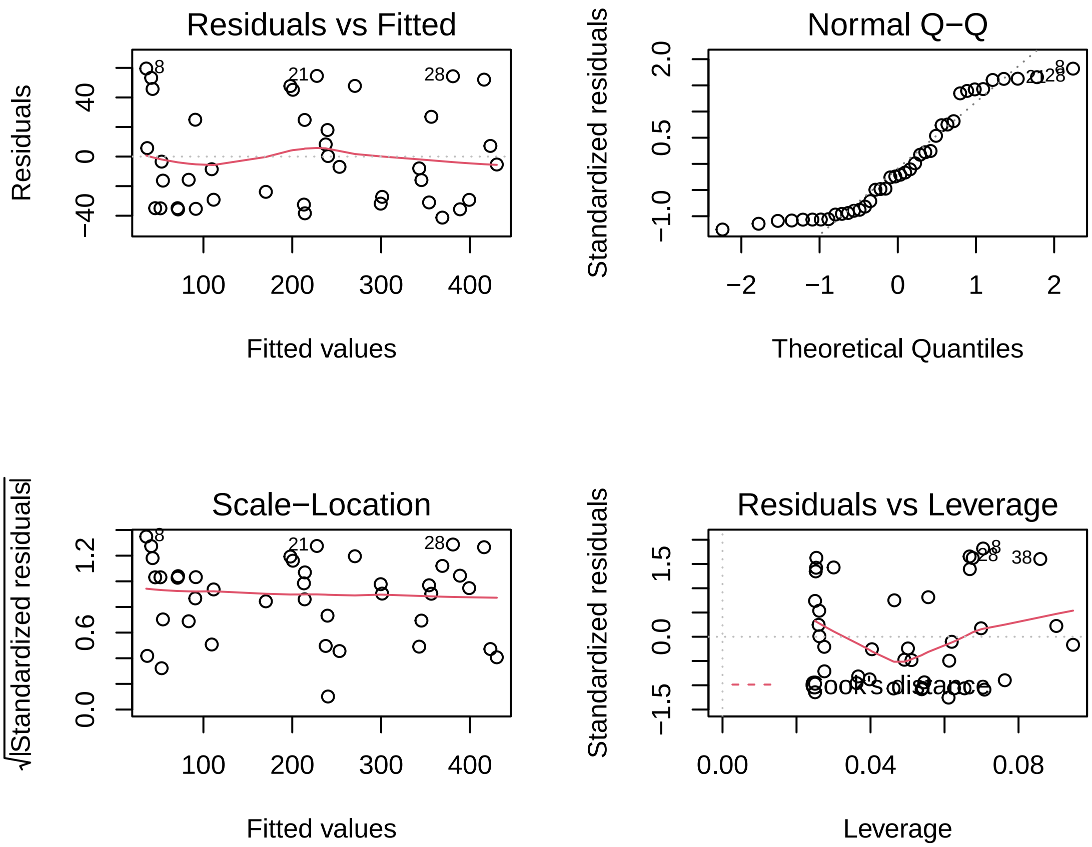
```
summary(M4)
```

```
##
## Call:
## lm(formula = y2 ~ x22, data = gh.data)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -41.280 -31.224  -7.463  25.422  59.571
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  35.1583     9.0306   3.893 0.000387 ***
## x22           1.0198     0.0419  24.338  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 33.96 on 38 degrees of freedom
## Multiple R-squared:  0.9397, Adjusted R-squared:  0.9381
## F-statistic: 592.3 on 1 and 38 DF,  p-value: < 2.2e-16
```

```
par(mfrow=c(2,2))
plot(M3)
```


Residuals vs Fitted
Residuals
Fitted values
Normal Q−Q
Standardized residuals
Theoretical Quantiles
Scale−Location
√|Standardized residuals|
Fitted values
Residuals vs Leverage
Standardized residuals
Cook's distance
Leverage

```
plot(M4)
```



Model M4 has the highest R^2 (0.9397), and has residuals vs fitted and scale-location plots that are at least as good as any of the others; on the other hand, we seem to be losing normality of the residuals. Nevertheless I prefer M4 so far.

problem 3(c)

Fit both of the models

$$\begin{aligned} \textbf{M5:} \quad \mathtt{y} &= \beta_0 + \beta_1\ \mathtt{x1} + \beta_2\ \mathtt{x2} + \varepsilon \\ \textbf{M6:} \quad \mathtt{y2} &= \beta_0 + \beta_1\ \mathtt{x12} + \beta_2\ \mathtt{x22} + \varepsilon \end{aligned}$$

Compare these to the earlier models. Which fits best? Why?

```r
M5 <- lm(y ~ x1 + x2, data=gh.data)
M6 <- lm(y2 ~ x12 + x22, data=gh.data)

summary(M5)
```

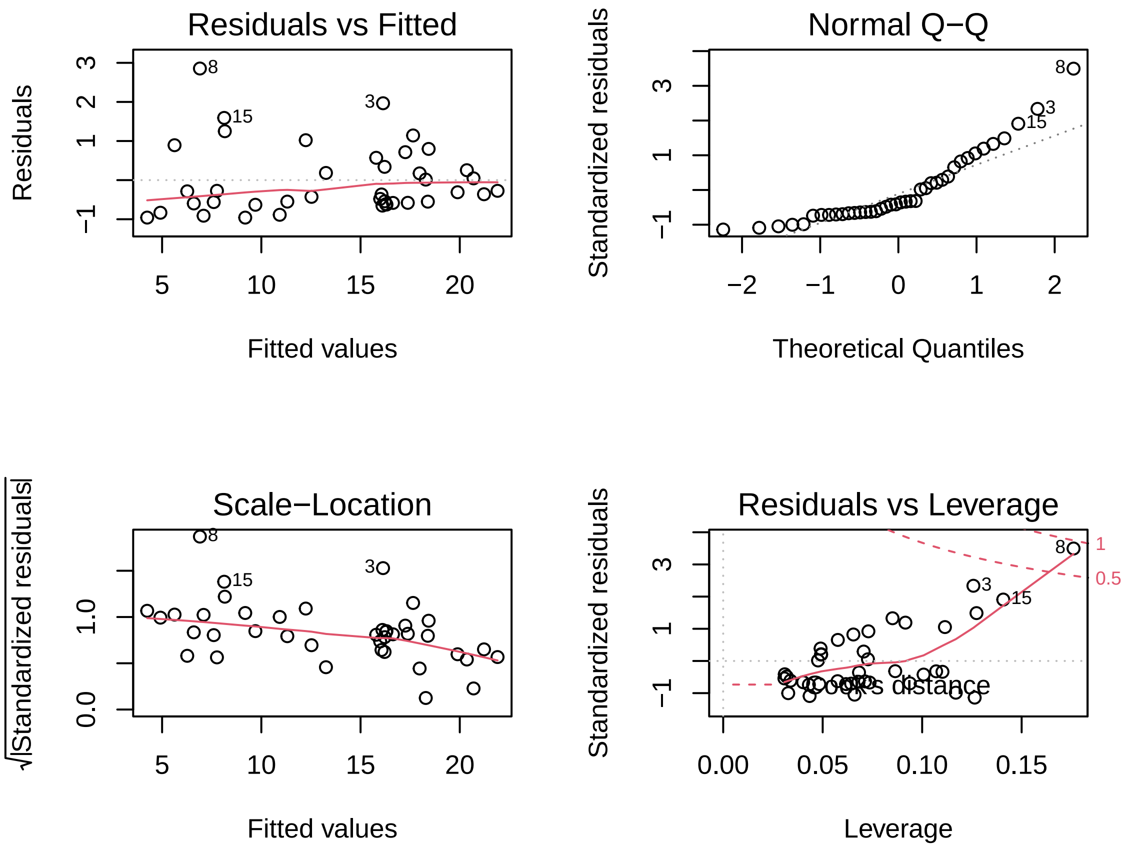
```
##
## Call:
## lm(formula = y ~ x1 + x2, data = gh.data)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
```

```
## -0.9585 -0.5865 -0.3356  0.3973  2.8548
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  1.31513    0.38769   3.392  0.00166 **
## x1           0.51481    0.04590  11.216 1.84e-13 ***
## x2           0.80692    0.02434  33.148  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.9 on 37 degrees of freedom
## Multiple R-squared:  0.9724, Adjusted R-squared:  0.9709
## F-statistic: 652.4 on 2 and 37 DF,  p-value: < 2.2e-16
```

```r
summary(M6)
```

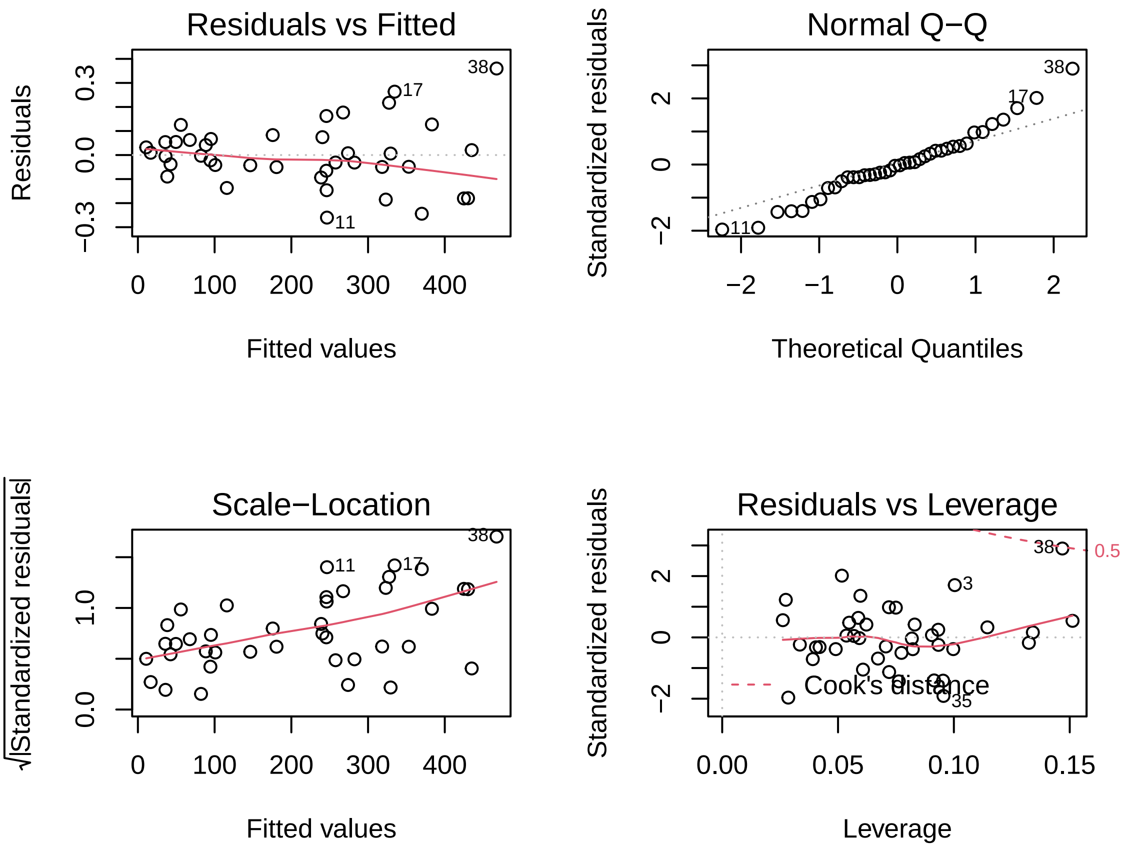
```
##
## Call:
## lm(formula = y2 ~ x12 + x22, data = gh.data)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.26020 -0.05391 -0.00396  0.06367  0.35990
##
## Coefficients:
##              Estimate Std. Error  t value Pr(>|t|)
## (Intercept) 0.0026691  0.0422669    0.063     0.95
## x12         0.9999672  0.0006419 1557.713   <2e-16 ***
## x22         0.9998685  0.0001663 6011.909   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1344 on 37 degrees of freedom
## Multiple R-squared:      1,  Adjusted R-squared:      1
## F-statistic: 2.013e+07 on 2 and 37 DF,  p-value: < 2.2e-16
```

```r
par(mfrow=c(2,2))
plot(M5)
```



Residuals vs Fitted
Residuals
Fitted values
Normal Q−Q
Standardized residuals
Theoretical Quantiles
Scale−Location
√|Standardized residuals|
Fitted values
Residuals vs Leverage
Standardized residuals
Cook's distance
Leverage


```
plot(M6)
```



Putting both x1 and x2 in the model for y really improved the model: M5 has an R^2 of 0.9724, and both predictors are significant (have coefficient estimates significantly different from zero). However the residual diagnostic plots don't look great; in particular it seems like the residuals have a lot of right skew, and leverage seems tio increase with the size of the standardized resduals.

M6 is really winning, though: $R^2 = 1$, and the QQ plot shows good agreement between the residuals and the normal distribution. There seems to be some evidence for non-constant variance though: the residuals vs fitted plot fans out as the fitted valuess increase, and the scale-location plot tells a similar story. On the other hand, only one data point seems to have a concerning Cook's distance.

Based on all of this I like M6 best. Looking at the estimated coefficients for M6, I notice something interesting: $\hat{\beta}_0$ is indistinguishable from 0, and both $\hat{\beta}_1$ and $\hat{\beta}_2$ equal 1, to at least two decimal places (even if we compute the 95% CI's!).

Problem 3(d)

Can you find a simple, recognizable function `x3 = (something involving both x1 and x2)`, so that

$$\textbf{M7:}\quad \texttt{y} \;=\; \beta_0 + \beta_1\ \texttt{x3} + \varepsilon$$

provides a fit comparable to the best fitting models above? What is going on?

In problem 3(c) we saw that the model M6 was very nearly

$$y^2 = x1^2 + x2^2 + \varepsilon$$

If we ignore ε, take square roots, and put ε and some "unknown'' regression coefficients back in, we get a model like

$$y = \beta_0 + \beta_1\sqrt{x1^2 + x2^2} + \varepsilon$$

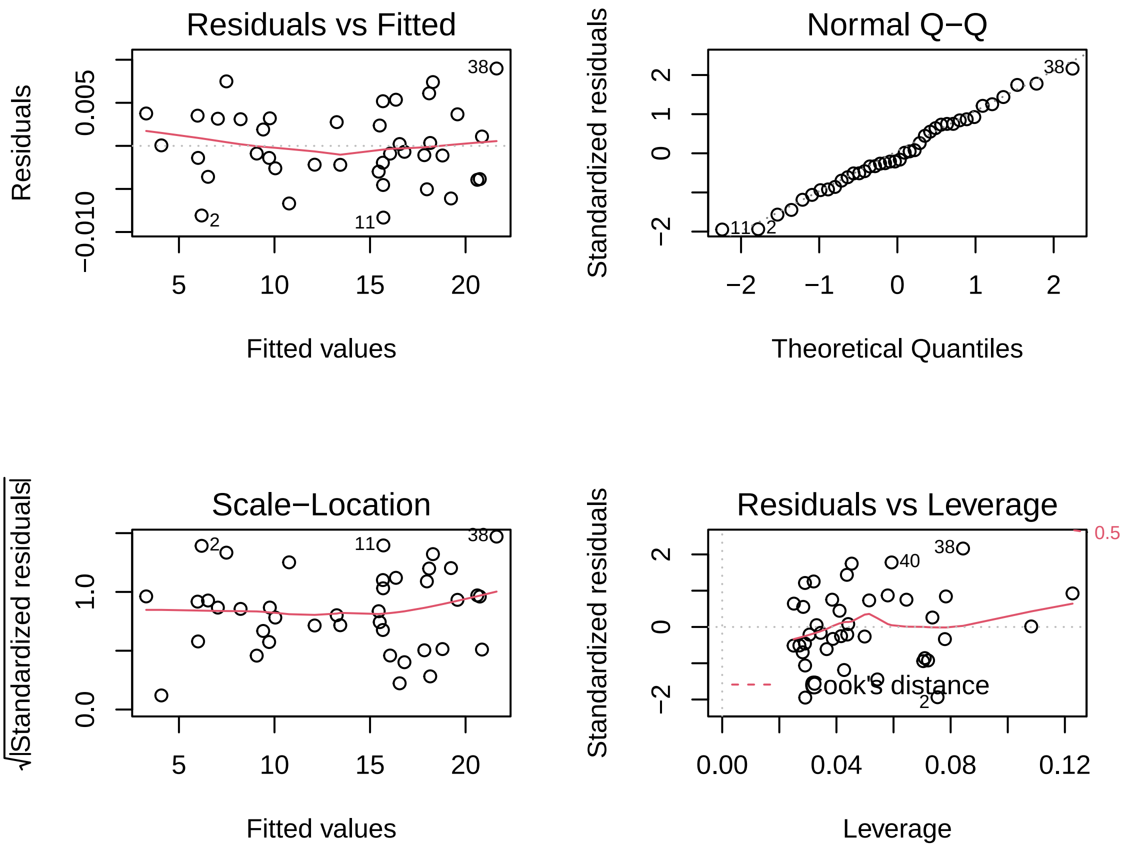
i.e., y is the distance to the origin from some points $(x1, x2)$ in Cartesian space.

Let's try fitting this model:

```
M7 <- lm(y ~ I(sqrt(x1^2 + x2^2)),data=gh.data)
summary(M7)
```

```
##
## Call:
## lm(formula = y ~ I(sqrt(x1^2 + x2^2)), data = gh.data)
##
## Residuals:
##        Min         1Q     Median         3Q        Max
## -0.0083283 -0.0027000 -0.0007907  0.0031643  0.0089809
##
## Coefficients:
##                       Estimate Std. Error  t value Pr(>|t|)
## (Intercept)          0.0018422  0.0019159    0.962    0.342
## I(sqrt(x1^2 + x2^2)) 0.9998313  0.0001316 7596.431   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.00434 on 38 degrees of freedom
## Multiple R-squared:      1,  Adjusted R-squared:      1
## F-statistic: 5.771e+07 on 1 and 38 DF,  p-value: < 2.2e-16
```

```
par(mfrow=c(2,2))
plot(M7)
```



This seems to confirm our suspicions! $R^2 = 1$, the estimated regression coefficients are essentially $\hat{\beta}_0 = 0$ and $\hat{\beta}_1 = 1$, and the diagnostic plots looks great:

- *The residual vs fitted plot shows little vertical structure.*
- *The QQ plot shows good adherence to normality.*
- *The scale-location plot is consistent with constant-variance residuals.*
- *None of the data points has Cook's distance above 0.5.*