This contains answers for computation questions 1,6, 7 and 8

Question 1

Design Matrix

We generated 20 samples

```
z <- 11:30
b0 <- 17
b1 <- 0.5
sigma <- 1.4
eps <- rnorm(z,0,sigma)

y <- b0 + b1*z + eps
#Design Matrix
Z = cbind(1, z)
```

Designed matrix contains the first columns of ones and second column of explanatory variable.

$$\begin{bmatrix} 1 & 11 \\ 1 & 12 \\ \vdots & \vdots \\ 1 & 29 \\ 1 & 30 \end{bmatrix}$$

Model 1

$$y_i = \beta_0 + \beta_1 z_i + \epsilon_i$$

```
Call:
lm(formula = y ~ z)

Residuals:
    Min      1Q  Median      3Q     Max 
-2.9959 -1.0198  0.3540  0.8162  1.8540 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept) 18.66446    1.05547  17.684 7.97e-13 ***
z            0.43624    0.04956   8.802 6.13e-08 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 1.278 on 18 degrees of freedom
Multiple R-squared:  0.8115,    Adjusted R-squared:  0.801 
F-statistic: 77.47 on 1 and 18 DF,  p-value: 6.129e-08
```

Model 2

$$y_i - \bar{y} = \beta_0 + \beta_1 z_i + \epsilon_i$$

Compared to model 1, the intercept β_0 got shifted by by $-\bar{y}$.
β_1 stays the same. The R Square stays the same

```
Call:
lm(formula = (y - ybar) ~ z)

Residuals:
    Min      1Q  Median      3Q     Max 
-2.9959 -1.0198  0.3540  0.8162  1.8540 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept) -8.94288    1.05547  -8.473 1.07e-07 ***
z            0.43624    0.04956   8.802 6.13e-08 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 1.278 on 18 degrees of freedom
Multiple R-squared:  0.8115,    Adjusted R-squared:  0.801 
F-statistic: 77.47 on 1 and 18 DF,  p-value: 6.129e-08
```

Model 3

$$y_i - \bar{y} = \beta_1 z_i + \epsilon_i$$

Compared to model 1, R Squared dropped significantly

```
Call:
lm(formula = (y - ybar) ~ z - 1)

Residuals:
    Min      1Q  Median      3Q     Max 
-4.2686 -3.0960 -0.6825  1.1293  4.3148 

Coefficients:
  Estimate Std. Error t value Pr(>|t|)
z  0.03198    0.02917   1.096    0.287

Residual standard error: 2.778 on 19 degrees of freedom
Multiple R-squared:  0.0595,	Adjusted R-squared:  0.009995 
F-statistic: 1.202 on 1 and 19 DF,  p-value: 0.2866
```

Model 4

$$cy_i = \beta_0 + \beta_1 z_i + \epsilon_i$$

Let $c = 5$

```
Call:
lm(formula = c * y ~ z)

Residuals:
    Min      1Q  Median      3Q     Max 
-14.979  -5.099   1.770   4.081   9.270 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept)  93.3223     5.2773  17.684 7.97e-13 ***
z             2.1812     0.2478   8.802 6.13e-08 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 6.391 on 18 degrees of freedom
Multiple R-squared:  0.8115,	Adjusted R-squared:  0.801 
F-statistic: 77.47 on 1 and 18 DF,  p-value: 6.129e-08
```

The R-square does not change compared to model 1 but the estimated parameters got multiplied by c

Question 6

Fitting Generalized Linear Models
Description
glm is used to fit generalized linear models, specified by giving a symbolic description of the linear predictor and a description of the error distribution.
Usage glm(formula, family = gaussian, data, weights, subset, na.action, start = NULL, etastart, mustart, offset, control = list(...), model = TRUE, method = "glm.fit", x = FALSE, y = TRUE, singular.ok = TRUE, contrasts = NULL, ...)
glm.fit(x, y, weights = rep.int(1, nobs), start = NULL, etastart = NULL, mustart = NULL, offset = rep.int(0, nobs), family = gaussian(), control = list(), intercept = TRUE, singular.ok = TRUE)
S3 method for class 'glm' weights(object, type = c("prior", "working"), ...)
Arguments
formula:
an object of class "formula" (or one that can be coerced to that class): a symbolic description of the model to be fitted. The details of model specification are given under 'Details'.
family:
a description of the error distribution and link function to be used in the model. For glm this can be a character string naming a family function, a family function or the result of a call to a family function. For glm.fit only the third option is supported. (See family for details of family functions.)
data:
an optional data frame, list or environment (or object coercible by as.data.frame to a data frame) containing the variables in the model. If not found in data, the variables are taken from environment(formula), typically the environment from which glm is called.
Value
coefficients:
a named vector of coefficients
residuals:
the working residuals, that is the residuals in the final iteration of the IWLS fit. Since cases with zero weights are omitted, their working residuals are NA.
deviance:
up to a constant, minus twice the maximized log-likelihood. Where sensible, the constant is chosen so that a saturated model has deviance zero.
aic:
A version of Akaike's An Information Criterion, minus twice the maximized log-likelihood plus twice the number of parameters, computed via the aic component of the family. For binomial and Poison families the dispersion is fixed at one and the number of parameters is the number of coefficients. For gaussian, Gamma and inverse gaussian families the dispersion is estimated from the residual deviance, and the number of parameters is the number of coefficients plus one. For a gaussian family the MLE of the dispersion is used so this is a

valid value of AIC, but for Gamma and inverse gaussian families it is not. For families fitted by quasi-likelihood the value is NA.

Question 7

7a

$$\hat{\beta} \sim N(\beta, \sigma^2(Z'Z)^{-1})$$

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

$$E(\hat{\beta}) = \beta = \begin{bmatrix} \beta_0 \\ \beta_1 \end{bmatrix} = \begin{bmatrix} 17.0 \\ 0.5 \end{bmatrix}$$

This is the β we generated

```
var_beta_hat <- (sigma^2)*solve(t(Z)%*%Z)
```

$$Cov(\hat{\beta}) = \sigma^2(Z'Z)^{-1} = \begin{bmatrix} 1.33663158 & -0.060421053 \\ -0.06042105 & 0.002947368 \end{bmatrix}$$

$E(\hat{\epsilon}) = 0$

```
I = diag(20)
H = Z%*%(solve(t(Z)%*%Z))%*%t(Z)
var_eps_hat <- (sigma^2)*(I-H)
```

$Cov(\hat{\epsilon}) = \sigma^2(I - H)$ with $H = Z(Z'Z)^{-1}Z'$

$$Cov(\hat{\epsilon}) = \begin{bmatrix} 1.596e+00 & \dots & 1.68e-01 \\ -3.36e-01 & \dots & 1.40e-01 \\ \vdots & \vdots & \\ 1.68e-01 & \dots & 1.596e+00 \end{bmatrix}$$

7b

Based on question 3b, the MLE estimate for the parameter $\hat{\beta}$ is the same as the Least Square

$$\hat{\beta} = (Z'Z)^{-1}Z'Y$$

```
Y = as.matrix(y)
Z = as.matrix(cbind(1,z)) #design matrix
beta_hat <- solve(t(Z)%*%Z)%*%(t(Z)%*%Y)
```

I got $\hat{\beta} = [15.8035082, 0.5547679]'$

Now, we are estimating $\hat{\sigma}^2$. The numerator is SSE, same between ML and LS, but the denominator is different. In LS, we divide SSE by $n - r - 1$. In ML, we divide SSE by n, as proven in question 3b

For LS:

$$\hat{\sigma}^2 = \frac{(Y - Z\hat{\beta})'(Y - Z\hat{\beta})}{n - (r + 1)}$$

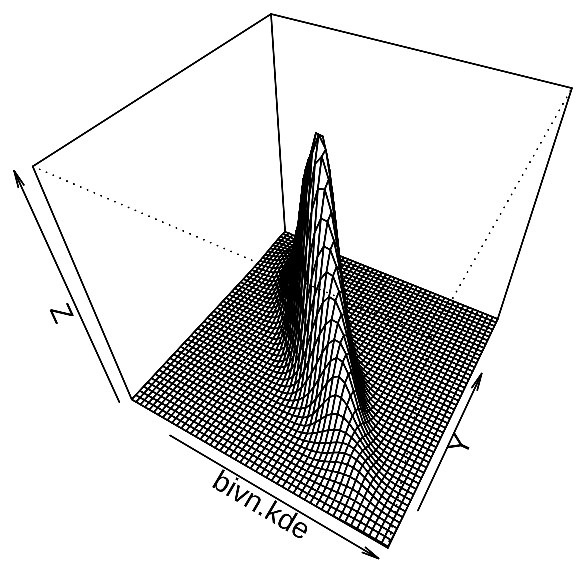
For ML:

$$\hat{\sigma}^2 = \frac{(Y - Z\hat{\beta})'(Y - Z\hat{\beta})}{n}$$

I got $\hat{\sigma}^2 = 2.75$ for LS and 2.48 for ML

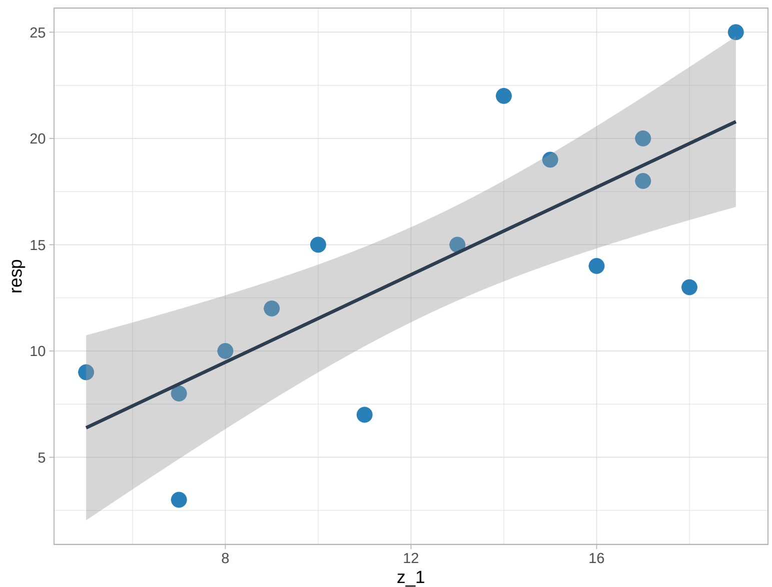
```
n = 20
r = 1
denominator_LS = n-r-1
denominator_ML = n

SSE = (t(Y-Z%*%beta_hat)%*%(Y-Z%*%beta_hat))
sigma_hat_sqrt_LS = SSE/denominator_LS
sigma_hat_sqrt_ML = SSE/denominator_ML
```

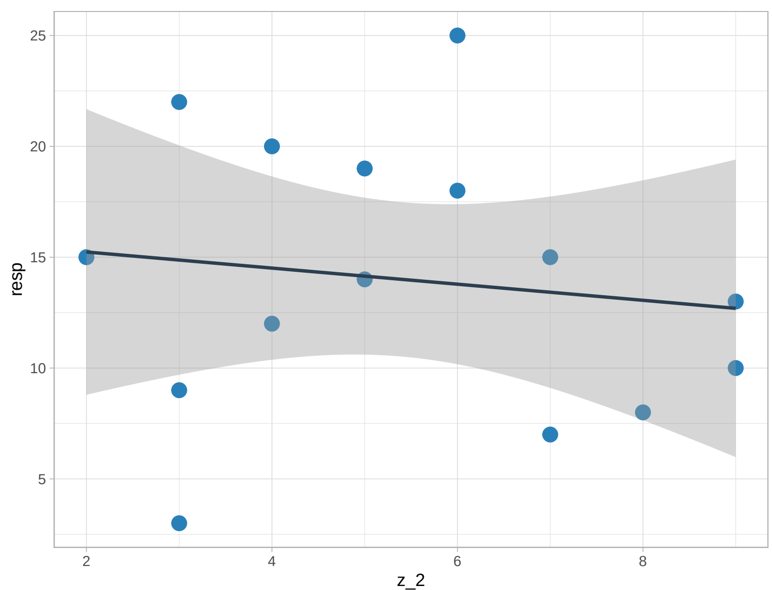
7c



Question 8

8.a



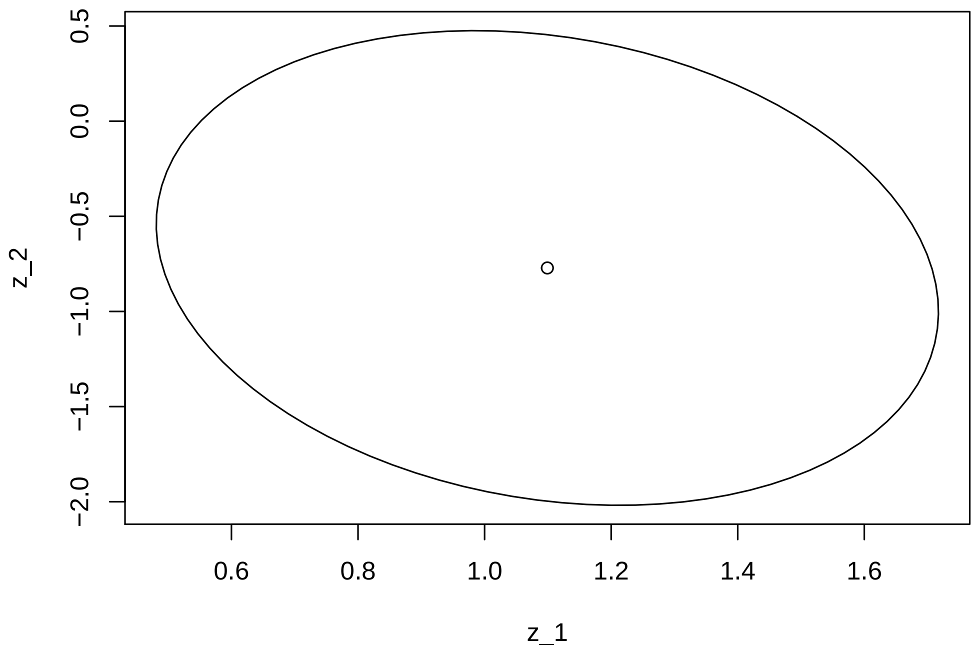


8.b

```
Call:
lm(formula = resp ~ z_1 + z_2, data = data)

Residuals:
   Min     1Q Median     3Q    Max
-6.916 -2.410  1.015  1.887  4.390

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)   4.5362     3.4459   1.316 0.212630
z_1           1.0992     0.2217   4.958 0.000332 ***
z_2          -0.7715     0.4474  -1.724 0.110310
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 3.721 on 12 degrees of freedom
Multiple R-squared:  0.6779,	Adjusted R-squared:  0.6243
F-statistic: 12.63 on 2 and 12 DF,  p-value: 0.001116
```

8.c

```
> confint(fit, level=0.95)
                 2.5 %     97.5 %
(Intercept) -2.9717845 12.0440962
z_1          0.6161005  1.5822635
z_2         -1.7463784  0.2034107
```



8.d

```
Model 1: resp ~ z_1
Model 2: resp ~ z_1 + z_2
  Res.Df    RSS Df Sum of Sq Pr(>Chi)
1     13 207.36
2     12 166.19  1    41.171  0.08467 .
```

We reject the null of $\beta_2 = 0$ at $\alpha = 10\%$ meaning that removing z_2 would decrease the predicted power of the model

8.e

$$z_0'\hat{\beta} \pm t_{n-r-1}\frac{\alpha}{2}\sqrt{\hat{\sigma}^2 z_0'(Z'Z)^{-1}z_0}$$

Our confidence interval is [1.58, 10.5]

```
est_resid_var = (summary(fit)$sigma)**2
#(t(Y-Z%*%beta_hat)%*%(Y-Z%*%beta_hat))/12
z_0 = matrix(c(1,7,8))
```

```
z0prime = t(z_0)
y_0_hat = z0prime%*%beta_hat
t = qt(1-0.025,12) # 95% CI with df = n-r-1 =15-2-1= 12
ZprimeZ_inv = solve(t(Z)%*%Z)
sqrt_component = sqrt(est_resid_var*z0prime%*%ZprimeZ_inv%*%z_0)
right_CI = y_0_hat + (t* sqrt_component)
left_CI = y_0_hat - (t* sqrt_component)
```

8.f

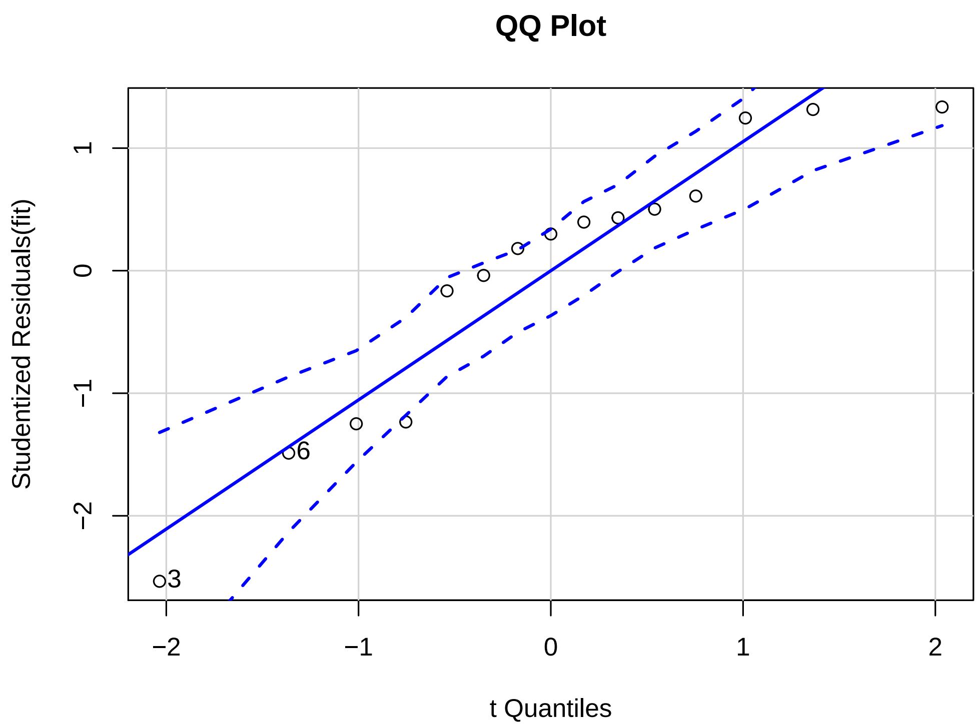
$$z_0'\hat{\beta} \pm t_{n-r-1}\frac{\alpha}{2}\sqrt{\hat{\sigma}^2(1 + z_0'(Z'Z)^{-1}z_0)}$$

Our confidence interval is [-3.2, 15.3]

```
sqrt_component_unobs = sqrt(est_resid_var*(1+z0prime%*%ZprimeZ_inv%*%z_0))
right_CI_unobs = y_0_hat + (t* sqrt_component_unobs)
left_CI_unobs = y_0_hat - (t* sqrt_component_unobs)
```

8.g

Outliers



Based on the plot above, observation number 3 and 6 seems to be an outliers.

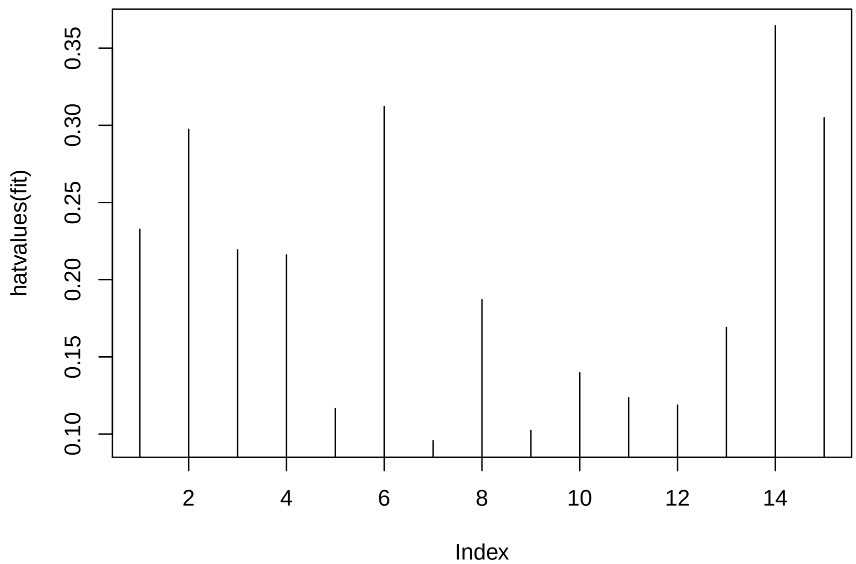
Left: with outlier. Right: without outlier

```
Call:
lm(formula = resp ~ z_1 + z_2, data = data)

Residuals:
   Min     1Q Median     3Q    Max 
-6.916 -2.410  1.015  1.887  4.390 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept)   4.5362     3.4459   1.316 0.212630    
z_1           1.0992     0.2217   4.958 0.000332 ***
z_2          -0.7715     0.4474  -1.724 0.110310    
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 3.721 on 12 degrees of freedom
Multiple R-squared:  0.6779,	Adjusted R-squared:  0.6243 
F-statistic: 12.63 on 2 and 12 DF,  p-value: 0.001116
```

```
Call:
lm(formula = resp ~ z_1 + z_2, data = dat_no_outlier)

Residuals:
    Min      1Q  Median      3Q     Max 
-5.1666 -0.7557  0.0058  1.1073  3.7044 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept)   6.0996     3.4594   1.763 0.108343    
z_1           1.0443     0.1962   5.322 0.000337 ***
z_2          -0.7744     0.4107  -1.886 0.088691 .  
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 2.978 on 10 degrees of freedom
Multiple R-squared:  0.7684,	Adjusted R-squared:  0.7221 
F-statistic: 16.59 on 2 and 10 DF,  p-value: 0.0006663
```

R^2 does improve after taking out the outliers.

Leverage




```
Call:
lm(formula = resp ~ z_1 + z_2, data = dat_no_lev)

Residuals:
      5       7       8       9      10      11      12      1
-2.4152  1.8723 -0.9839  3.5297  0.6184 -0.7637 -4.1552  2.297

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  10.7206     7.3447   1.460   0.2042
z_1           1.0275     0.4016   2.559   0.0507 .
z_2          -1.8012     0.7929  -2.272   0.0723 .
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 3.042 on 5 degrees of freedom
Multiple R-squared:  0.7227,    Adjusted R-squared:  0.6118
F-statistic: 6.515 on 2 and 5 DF,  p-value: 0.0405
```

Influential points are 1,2,3,4,6,14,15

R^2 decreases when taking out these points from the original model.These high leverage points contribute to the explanatory power of the model, so some of these might be influential in terms of R^2

Influential Points

Chart of Cook's distance to detect observations that strongly influence fitted values of the model. Oservation 3, 6, 14 are influential points