

STAT****/STAT****: Linear Models

Assignment 1

__****

InstantGrad



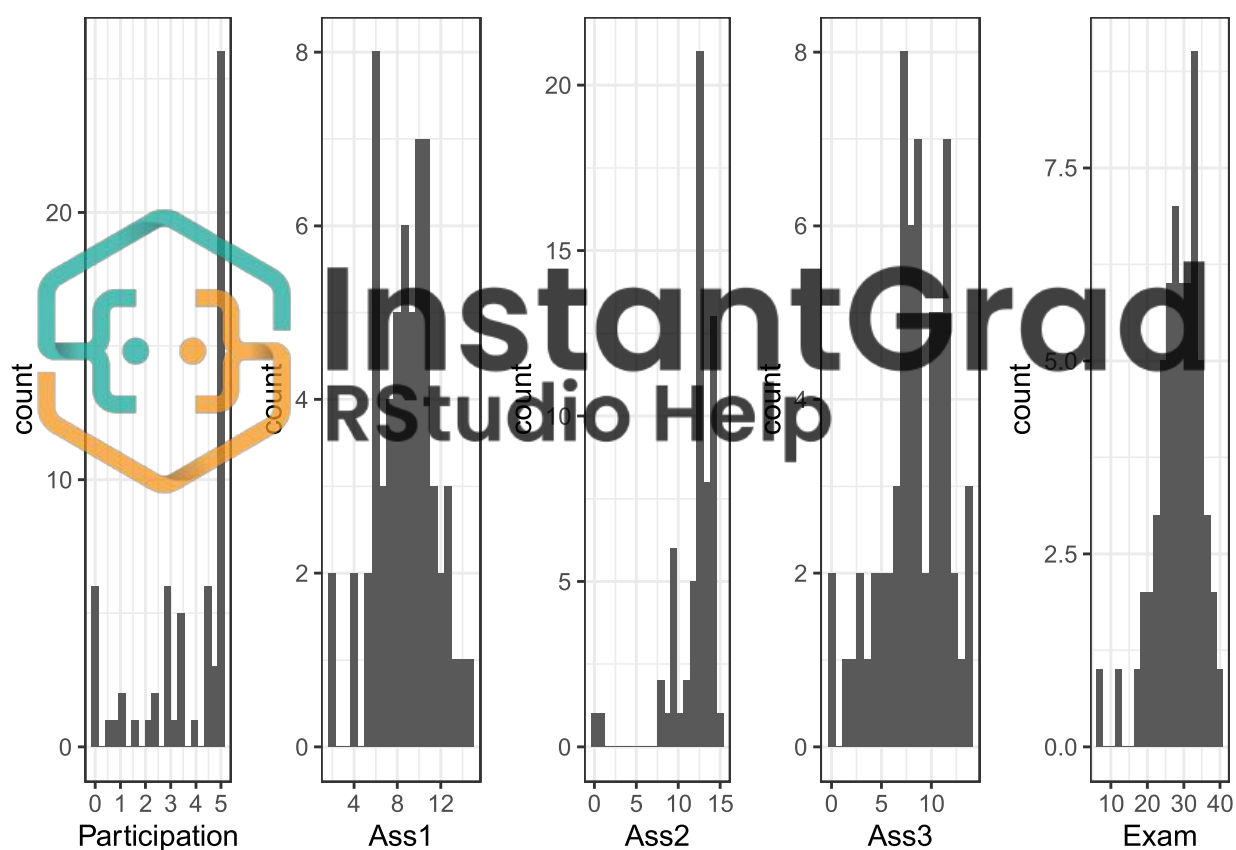
InstantGrad
RStudio Help

Appendix 1

```
# Set working directory
setwd(dirname(rstudioapi::getSourceEditorContext()$path))

# Load .csv file
marks <- read.csv("Marks.csv")

p1 <- ggplot(marks, aes(x=Participation)) + geom_histogram(bins = 20)+theme_bw()
p2 <- ggplot(marks, aes(x=Ass1)) + geom_histogram(bins = 20)+theme_bw()
p3 <- ggplot(marks, aes(x=Ass2)) + geom_histogram(bins = 20)+theme_bw()
p4 <- ggplot(marks, aes(x=Ass3)) + geom_histogram(bins = 20)+theme_bw()
p5 <- ggplot(marks, aes(x=Exam)) + geom_histogram(bins = 20)+theme_bw()
grid.arrange(p1,p2,p3,p4,p5,nrow=1)
```



```

p1 <- ggplot(marks, aes(x = Participation, y =Exam )) + geom_point(size=0.1) +
geom_smooth(formula = y ~ x, method = "loess", se = FALSE) +
  ylab("Exam mark (Out of 50)") + xlab("Participation (Out of 5)") + theme_bw()

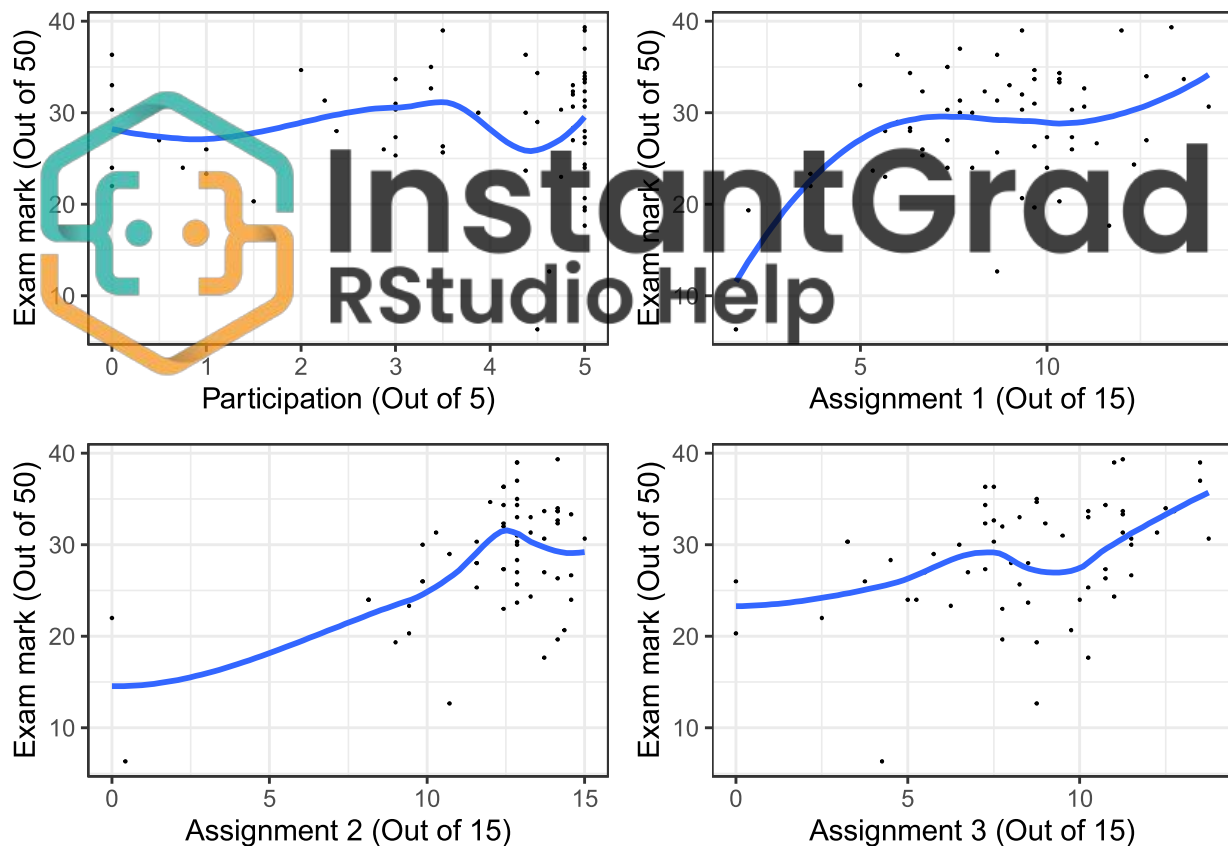
p2 <- ggplot(marks, aes(x = Ass1, y =Exam )) + geom_point(size=0.1) +
geom_smooth(formula = y ~ x, method = "loess", se = FALSE) +
  ylab("Exam mark (Out of 50)") + xlab("Assignment 1 (Out of 15)") + theme_bw()

p3 <- ggplot(marks, aes(x = Ass2, y =Exam )) + geom_point(size=0.1) +
geom_smooth(formula = y ~ x, method = "loess", se = FALSE) +
  ylab("Exam mark (Out of 50)") + xlab("Assignment 2 (Out of 15)") + theme_bw()

p4 <- ggplot(marks, aes(x = Ass3, y =Exam )) + geom_point(size=0.1) +
geom_smooth(formula = y ~ x, method = "loess", se = FALSE) +
  ylab("Exam mark (Out of 50)") + xlab("Assignment 3 (Out of 15)") + theme_bw()

grid.arrange(p1,p2,p3,p4,nrow=2)

```



Appendix 2

```
# Linear model
m1 <- lm(Exam ~ Ass1 + Ass2 + Ass3, data=marks)

# Summary of model m1
summary(m1)
```

```
##
## Call:
## lm(formula = Exam ~ Ass1 + Ass2 + Ass3, data = marks)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -14.8569  -2.7596   0.9453   3.3997   8.7781
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  13.9026     3.2471   4.282 7.34e-05 ***
## Ass1          0.1669     0.3247   0.514  0.6092
## Ass2          0.8876     0.3613   2.456  0.0172 *
## Ass3          0.3045     0.2761   1.103  0.2748
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 5.48 on 56 degrees of freedom
## (2 observations deleted due to missingness)
## Multiple R-squared:  0.2862, Adjusted R-squared:  0.248
## F-statistic: 7.485 on 3 and 56 Df, p-value: 0.0002678
##
# Confidence intervals model m1
confint(m1)
```

```
##              2.5 %      97.5 %
## (Intercept)  7.3977937 20.4073365
## Ass1         -0.4835773  0.8174537
## Ass2          0.1637531  1.6113703
## Ass3         -0.2485935  0.8576475
```

Appendix 4

```
# Full model
mfull <- lm(Exam ~ Participation + Ass1 + Ass2 + Ass3, data = marks)

# Reduced model
mred <- lm(Exam ~ Ass1 + Ass2, data = marks)

# Anova Reduced vs Full model
anova(mred,mfull)

## Analysis of Variance Table
##
## Model 1: Exam ~ Ass1 + Ass2
## Model 2: Exam ~ Participation + Ass1 + Ass2 + Ass3
##   Res.Df    RSS Df Sum of Sq    F Pr(>F)
## 1      57 1718.5
## 2      55 1619.4  2    99.114 1.6831 0.1952
```



Appendix 5

Model with Ass1

```
ma <- lm(Exam ~ Ass1, data=marks)
summary(ma)
```

```
##
## Call:
## lm(formula = Exam ~ Ass1, data = marks)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -16.537  -3.147   1.207   3.724   9.887
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  21.4949     2.5867    8.31 1.86e-11 ***
## Ass1         0.8253     0.2855    2.89  0.0054 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 5.959 on 58 degrees of freedom
## (2 observations deleted due to missingness)
## Multiple R-squared:  0.1259, Adjusted R-squared:  0.1108
## F-statistic: 8.355 on 1 and 58 DF, p-value: 0.005404
```

Model with Ass2

```
mb <- lm(Exam ~ Ass2, data=marks)
summary(mb)
```

```
##
## Call:
## lm(formula = Exam ~ Ass2, data = marks)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -14.3517  -2.9129   0.3654   3.1525   9.4517
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  14.3687     3.1853    4.511 3.20e-05 ***
## Ass2         1.1806     0.2571    4.592 2.41e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 5.458 on 58 degrees of freedom
## (2 observations deleted due to missingness)
## Multiple R-squared:  0.2666, Adjusted R-squared:  0.254
## F-statistic: 21.09 on 1 and 58 DF, p-value: 2.408e-05
```

Appendix 6

```
# Model with Ass1 and Ass2
```

```
m2 <- lm(Exam ~ Ass1 + Ass2, data=marks)
```

```
summary(m2)
```

```
##
```

```
## Call:
```

```
## lm(formula = Exam ~ Ass1 + Ass2, data = marks)
```

```
##
```

```
## Residuals:
```

```
##      Min       1Q   Median       3Q      Max  
## -14.4994  -3.1604   0.5575   3.5249   9.4085
```

```
##
```

```
## Coefficients:
```

```
##              Estimate Std. Error t value Pr(>|t|)  
## (Intercept)  14.0619      3.2501   4.327 6.17e-05 ***  
## Ass1         0.1832      0.3250   0.564 0.57517  
## Ass2         1.0749      0.3195   3.364 0.00138 **
```

```
## ---
```

```
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

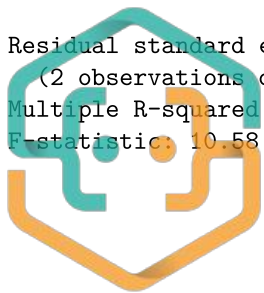
```
##
```

```
## Residual standard error: 5.491 on 57 degrees of freedom
```

```
## (2 observations deleted due to missingness)
```

```
## Multiple R-squared:  0.2707, Adjusted R-squared:  0.2451
```

```
## F-statistic: 10.58 on 2 and 57 DF, p-value: 0.0001238
```



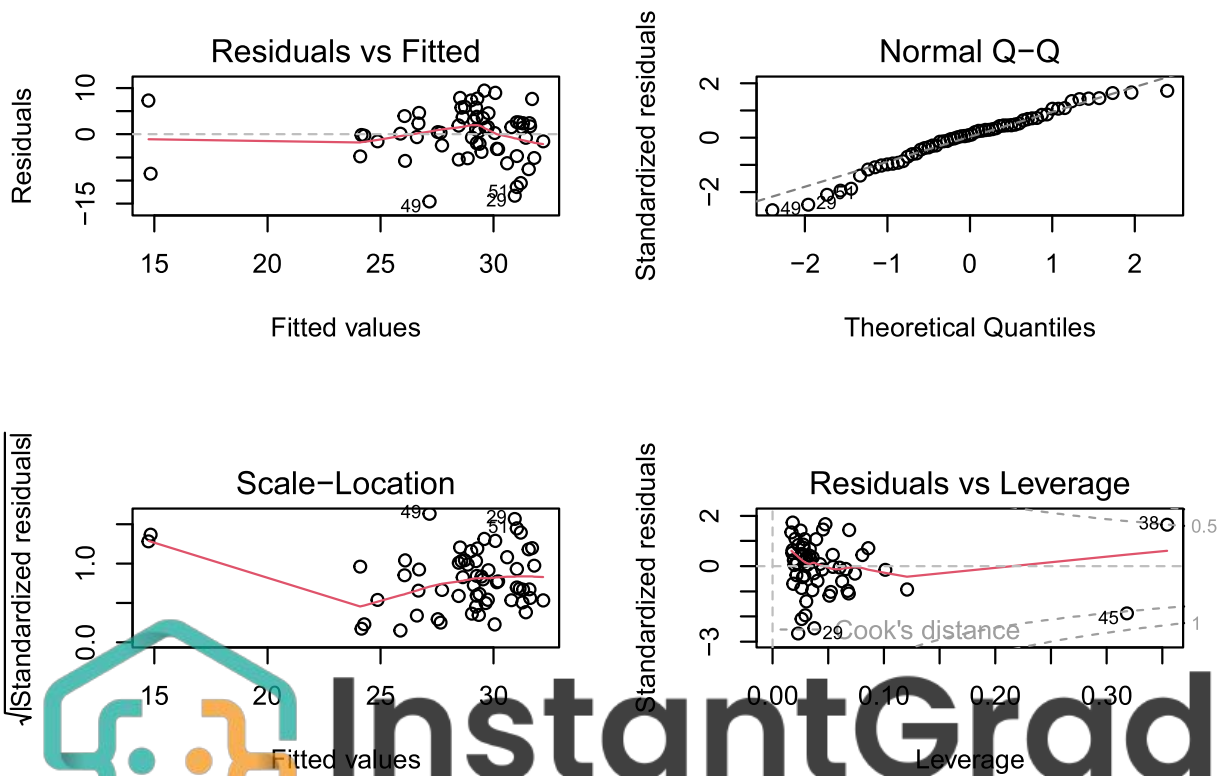
InstantGrad
RStudio Help

Appendix 7

```
# Model with Ass1 and Ass2
m2 <- lm(Exam ~ Ass1 + Ass2, data=marks)
summary(m2)

##
## Call:
## lm(formula = Exam ~ Ass1 + Ass2, data = marks)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -14.4994  -3.1604   0.5575   3.5249   9.4085
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  14.0619      3.2501   4.327 6.17e-05 ***
## Ass1         0.1832      0.3250   0.564 0.57517
## Ass2         1.0749      0.3195   3.364 0.00138 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 5.491 on 57 degrees of freedom
## (2 observations deleted due to missingness)
## Multiple R-squared:  0.2707, Adjusted R-squared:  0.2451
## F-statistic: 10.58 on 2 and 57 DF, p-value: 0.0001238

# Plots of Model m2
par(mfrow=c(2,2))
plot(lm(Exam ~ Ass1 + Ass2, data=marks))
```

```
# Breusch-Pagan test
bptest(m2)
```

```
##
## studentized Breusch-Pagan test
##
## data: m2
## BP = 0.7597, df = 2, p-value = 0.684
```

Appendix 8

```
# Count rows
n <- nrow(marks)
index<-as_tibble(c(1:60))

# Calculate leverage for each observation under model fitted
colnames(index)<-c('Observation')
lev <- as_tibble(hatvalues(m2))

# Calculate DFBetas for each observation under model fitted
colnames(lev) <- c('Leverage')
dfbetas <- as_tibble(dfbetas(m2))

# Calculate Cook's distance for each observation under model fitted
colnames(dfbetas) <- c('DFBeta0', 'DFBeta1', 'DFBeta2')
cooksD <- as_tibble(cooks.distance(m2))
colnames(cooksD)<-c('CooksD')
tibble(lev,dfbetas, cooksD)

## # A tibble: 60 x 5
##   Leverage DFBeta0 DFBeta1 DFBeta2   CooksD
##   <dbl>    <dbl>    <dbl>    <dbl>    <dbl>
## 1  0.0206  0.0236  0.0696 -0.0444  0.00839
## 2  0.0626  0.0262 -0.131  0.0585  0.00808
## 3  0.0308 -0.0198  0.0178  0.00989  0.000698
## 4  0.0603 -0.0102 -0.00517 0.0111  0.0000593
## 5  0.0689 -0.150  0.311  -0.0646  0.0511
## 6  0.0171  0.0195 -0.0148  0.0268  0.0105
## 7  0.0685 -0.131  -0.215  0.240  0.0289
## 8  0.0360  0.0100  0.0482 -0.0352  0.00151
## 9  0.0216 -0.000379 -0.0241  0.00949  0.000918
## 10 0.0329  0.0471  -0.0478  0.00543  0.00210
## # ... with 50 more rows

# Save diagnostics
diagnostics<-tibble(index,lev,dfbetas,cooksD)

# Threshold
thresh <- 2/sqrt(60)
p_plus_1<- 3
thresh_leverage<-2*p_plus_1/60

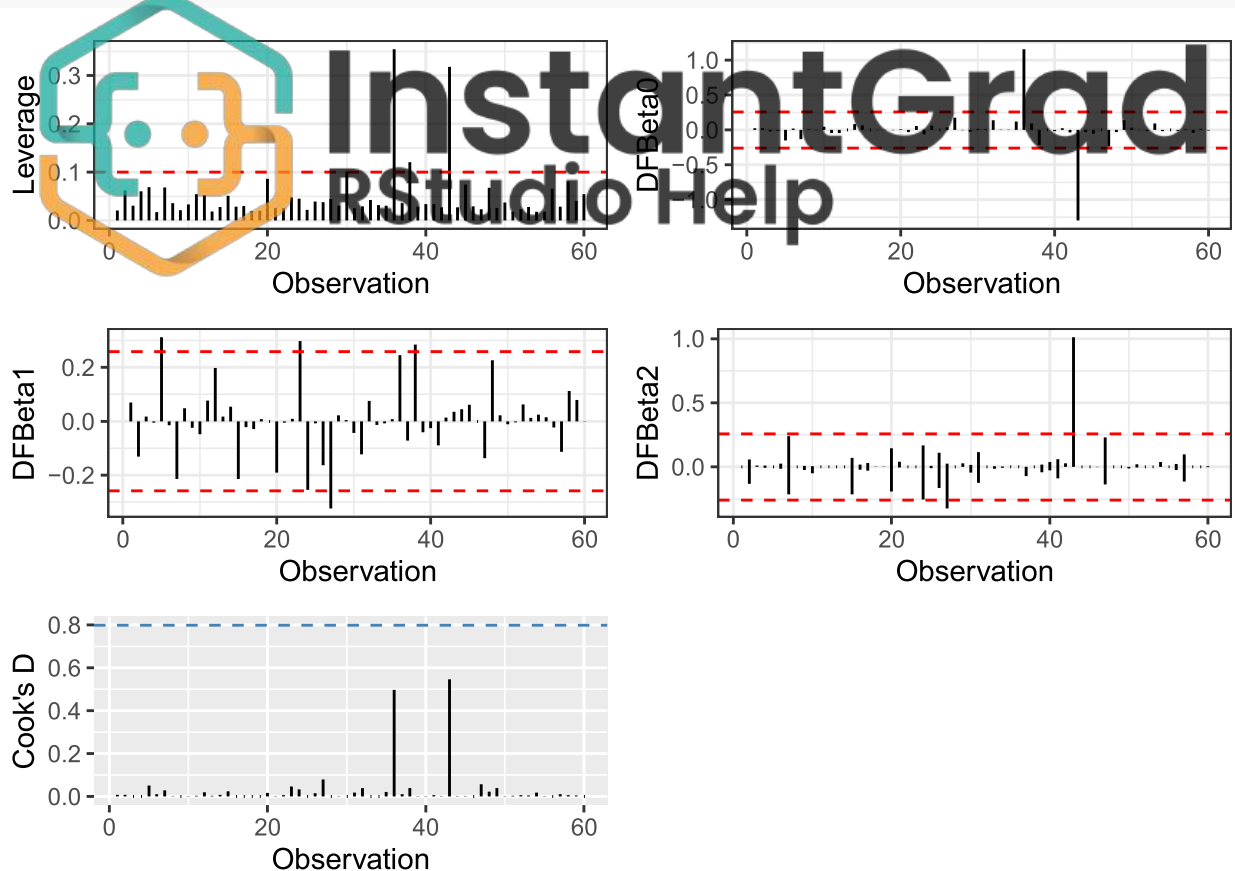
# Graphs are in order of the questions
# Construct Plot
p1<-ggplot(diagnostics, aes(x = Observation,
                           ymax=pmax(0, Leverage), ymin = pmin(0, Leverage))) +
  geom_linerange()+theme_bw() +
  geom_hline(yintercept=thresh_leverage, col="red", linetype="dashed") +
  labs(y="Leverage")
p2<-ggplot(diagnostics, aes(x = Observation, y=Leverage,
                           ymax=pmax(0, DFBeta0), ymin = pmin(0, DFBeta0))) +
  geom_linerange()+theme_bw() +
  geom_hline(yintercept=thresh, col="red", linetype="dashed") +
  geom_hline(yintercept=-thresh, col="red", linetype="dashed") +
```

```

labs(y="DFBeta0")
p3<- ggplot(diagnostics, aes(x = Observation, ymax=pmax(0, DFBeta1),
                           ymin = pmin(0, DFBeta1))) +
  geom_linerange()+theme_bw() +
  geom_hline(yintercept=thresh, col="red", linetype="dashed") +
  geom_hline(yintercept=-thresh, col="red", linetype="dashed") +
  labs(y="DFBeta1")
p4<- ggplot(diagnostics, aes(x = Observation, ymax=pmax(0, DFBeta2),
                           ymin = pmin(0, DFBeta1))) +
  geom_linerange()+theme_bw() +
  geom_hline(yintercept=thresh, col="red", linetype="dashed") +
  geom_hline(yintercept=-thresh, col="red", linetype="dashed") +
  labs(y="DFBeta2")
p5<-ggplot(diagnostics, aes(x = Observation, ymax=pmax(0, CooksD),
                           ymin = pmin(0, CooksD))) +
  geom_linerange()+
  geom_hline(yintercept=qf(0.5,p_plus_1,n-p_plus_1),
            col="steelblue", linetype="dashed") +
  labs(y="Cook's D")

grid.arrange(p1,p2,p3,p4,p5,nrow=3)

```



8a - Flagged leverage

```
diagnostics %>% filter(Leverage > thresh_leverage) %>% select(Observation,Leverage) %>%  
  mutate(Rule = "Exceeds 0.10")
```

```
## # A tibble: 4 x 3  
##   Observation Leverage Rule  
##       <int>     <dbl> <chr>  
## 1         30     0.101 Exceeds 0.10  
## 2         36     0.354 Exceeds 0.10  
## 3         38     0.121 Exceeds 0.10  
## 4         43     0.318 Exceeds 0.10
```

8b - Flagged DfBeta0 (Exam mark)

```
diagnostics %>% filter(DFBeta0 > thresh | DFBeta0 < -thresh) %>% select(Observation,DFBeta0) %>%  
  mutate(Rule = "Exceeds absolute(0.2582)")
```

```
## # A tibble: 2 x 3  
##   Observation DFBeta0 Rule  
##       <int>     <dbl> <chr>  
## 1         36     1.16 Exceeds absolute(0.2582)  
## 2         43    -1.30 Exceeds absolute(0.2582)
```

8b - Flagged DfBeta1 (Ass1)

```
diagnostics %>% filter(DFBeta1 > thresh | DFBeta1 < -thresh) %>% select(Observation,DFBeta1) %>%  
  mutate(Rule = "Exceeds absolute(0.2582)")
```

```
## # A tibble: 4 x 3  
##   Observation DFBeta1 Rule  
##       <int>     <dbl> <chr>  
## 1          5     0.311 Exceeds absolute(0.2582)  
## 2         23     0.298 Exceeds absolute(0.2582)  
## 3         27    -0.323 Exceeds absolute(0.2582)  
## 4         38     0.284 Exceeds absolute(0.2582)
```

8b - Flagged DfBeta2 (Ass2)

```
diagnostics %>% filter(DFBeta2 > thresh | DFBeta2 < -thresh) %>% select(Observation,DFBeta2) %>%  
  mutate(Rule = "Exceeds absolute(0.2582)")
```

```
## # A tibble: 2 x 3  
##   Observation DFBeta2 Rule  
##       <int>     <dbl> <chr>  
## 1         36    -1.10 Exceeds absolute(0.2582)  
## 2         43     1.01 Exceeds absolute(0.2582)
```

8c - Flagged CooksD

```
diagnostics %>% filter(CooksD > 0.45 | CooksD > 1) %>%  
  select(Observation,CooksD) %>% mutate(Rule = "Exceeds Rule of tumb 1: Ci > 0.5") %>%  
  mutate(Rule = case_when(CooksD < 0.5 ~ "Almost Exceeds Rule of tumb 1: Ci > 0.5", TRUE ~ Rule))
```

```
## # A tibble: 2 x 3  
##   Observation CooksD Rule  
##       <int>     <dbl> <chr>  
## 1         36     0.496 Almost Exceeds Rule of tumb 1: Ci > 0.5  
## 2         43     0.546 Exceeds Rule of tumb 1: Ci > 0.5
```