

## Lab-05

Rezaur Rashid

2022-03-31

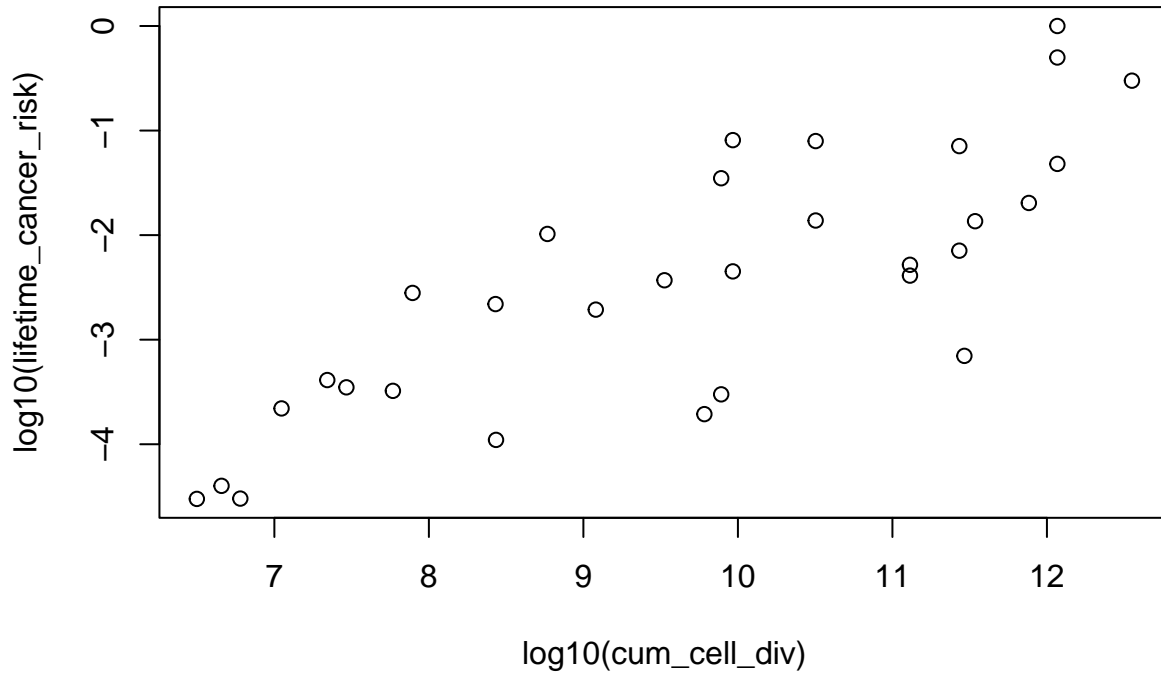
### Problem-1A:

On a log10-log10 scale graph Lifetime\_cancer\_risk (on the y-axis) vs. CumulativeCellDivisions (on the x-axis)

```
myT = read.table('data/cancerRisk.txt', header = TRUE, sep='\t')

lifetime_cancer_risk = myT[, c('Lifetime_cancer_risk')]
cum_cell_div = myT[, c('CumulativeCellDivisions')]

plot(log10(cum_cell_div), log10(lifetime_cancer_risk))
```

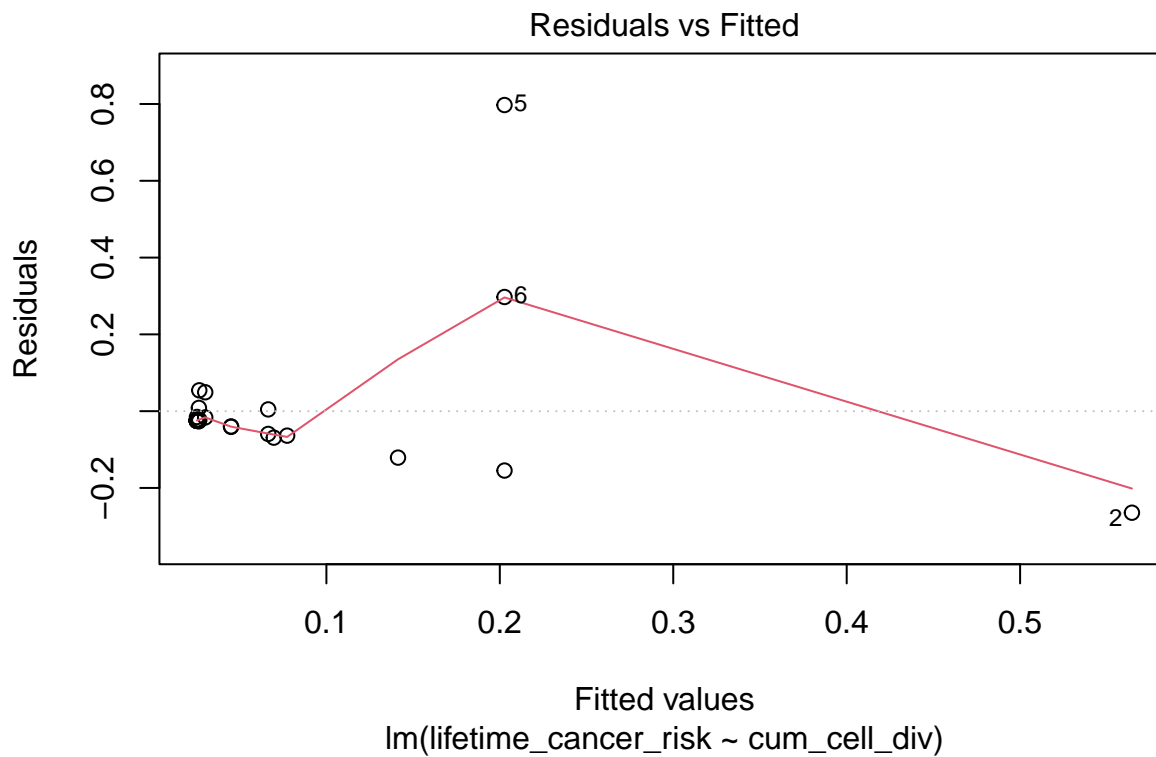


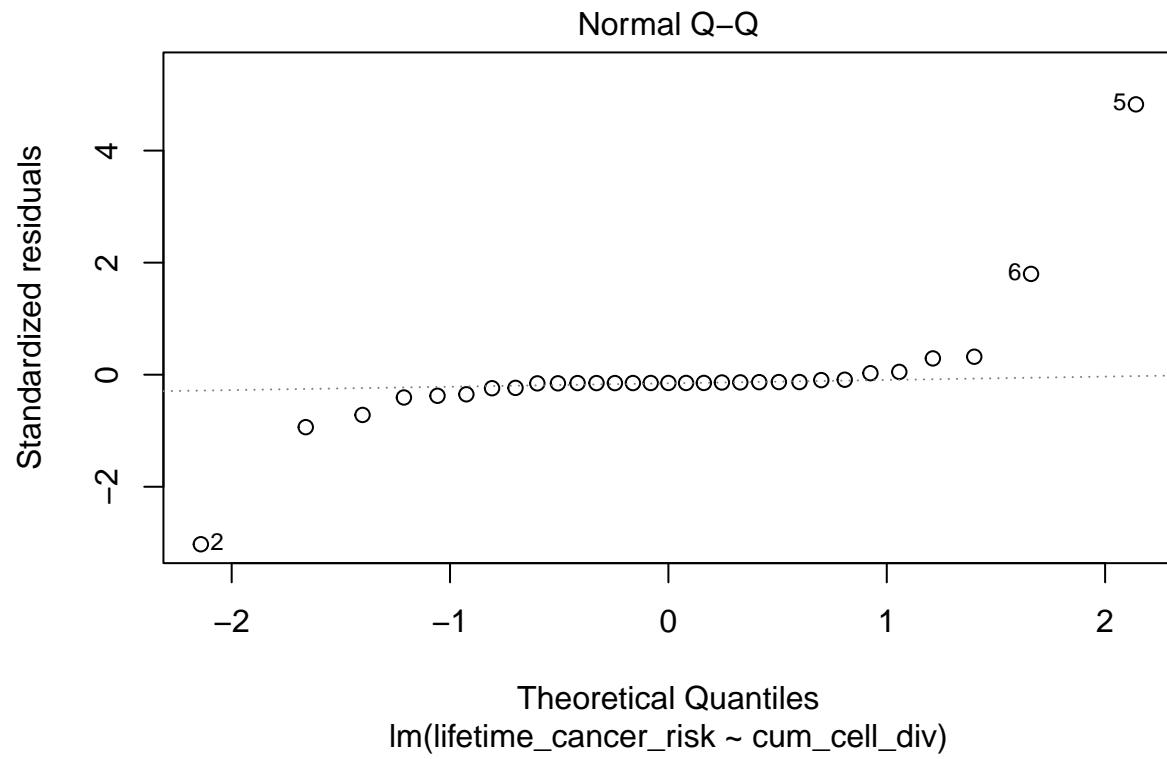
### Problem-1B:

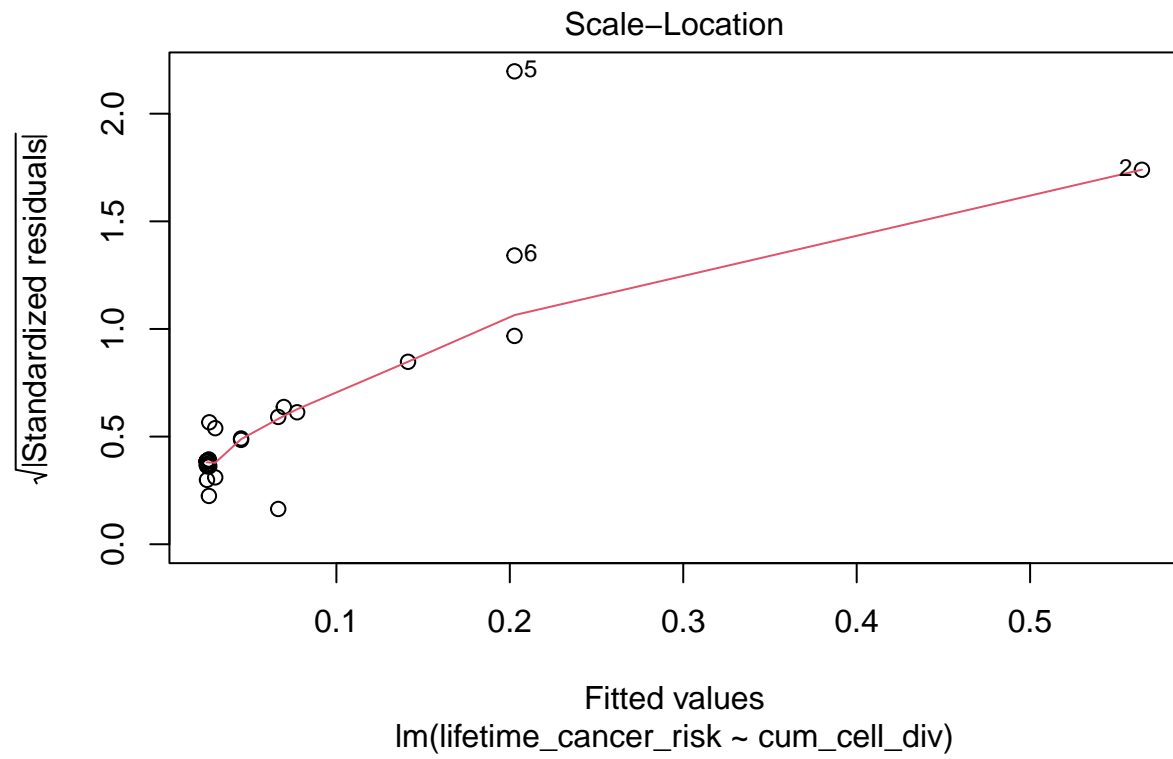
Using the `lm` function, fit a linear model with `Lifetime_cancer_risk` as the Y variable and `CumulativeCellDivisions` as the x-data. Add the regression line to the plot using the function `abline(myLm)`

```
myLm = lm(lifetime_cancer_risk~cum_cell_div)
```

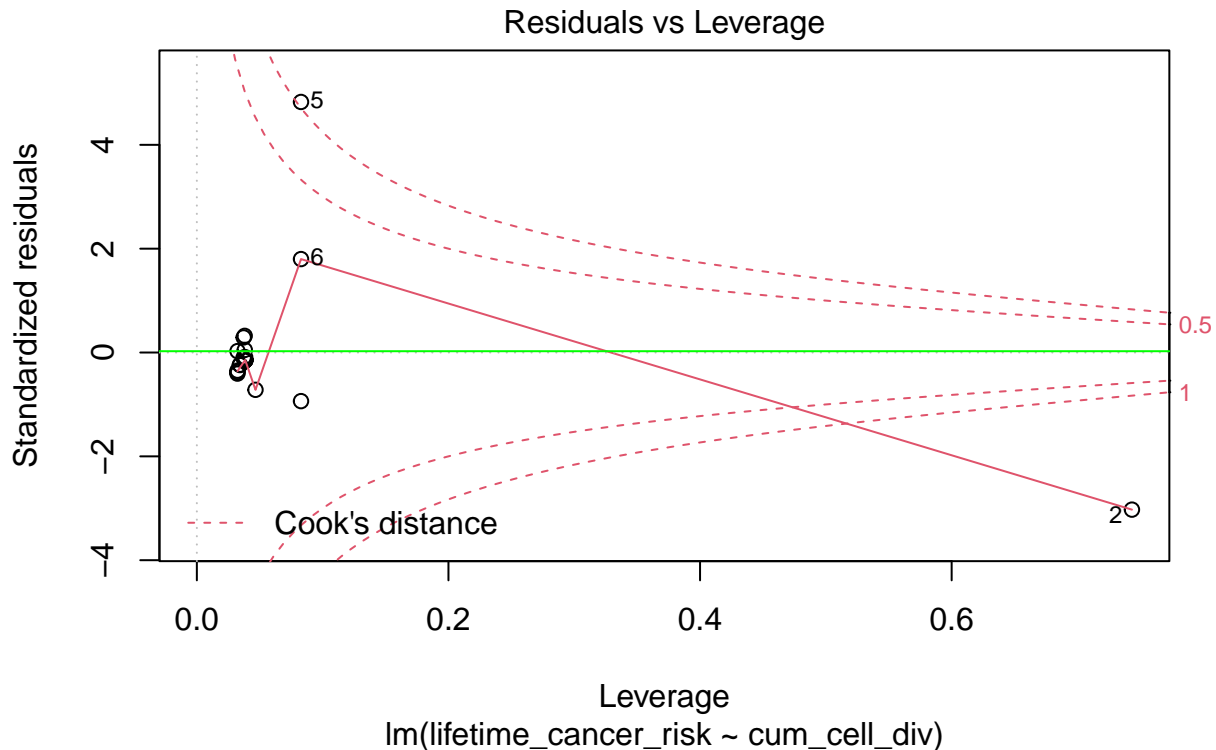
```
plot(myLm)
```







```
abline(myLm, col='green')
```



### Problem-1C:

What is the p-value for the null hypothesis that the slope of the regression between these two variables is zero? What is the r-squared value of the model?

```
sumT = summary(myLm)

anova( myLm)$"Pr(>F)"[1]

## [1] 0.002027674

cor(lifetime_cancer_risk, cum_cell_div) * cor(lifetime_cancer_risk, cum_cell_div)

## [1] 0.2839264

paste('p-value: ', sumT$coefficients[2,4])

## [1] "p-value: 0.00202767415572347"

paste('r-squared value: ',sumT$r.squared)

## [1] "r-squared value: 0.283926428052786"
```

### Problem-1D:

From the independence and normality assumptions of linear regression, we assume that the data has constant variance and they are normally distributed. For **Problem-1** Data. if we look at the Q-Q plot we see that almost all the points reside in the regression line which tells us that the data has very small differences in their variance (this is of of the cleanest biological data that has constant variance).

## Problem-2:

```
myTcaseCont = read.table('data/caseControlData.txt', header = TRUE, sep='\t')

myTbmi = read.table('data/BMI_Data.txt', header = TRUE, sep='\t')

casecont_sample = myTcaseCont[, c("sample")]
casecont_col = colnames(myTcaseCont)
casecont_col = casecont_col[! casecont_col %in% c('sample')]

bmi_val = myTbmi[, c("bmi")]
bmi_id = myTbmi[, c("studyid")]

case_id = vector()
for (i in 1:length(casecont_sample)) {
  subStr = substring(casecont_sample[i],1,10)
  case_id[i] = subStr
}

new_bmi_id = vector()
new_bmi_val = vector()
index = 1
for (i in 1:length(bmi_id)) {
  id = bmi_id[i]
  val = bmi_val[i]
  if (id %in% case_id == TRUE){
    new_bmi_id[index] = id
    new_bmi_val[index] = val
    index = index + 1
  }
}

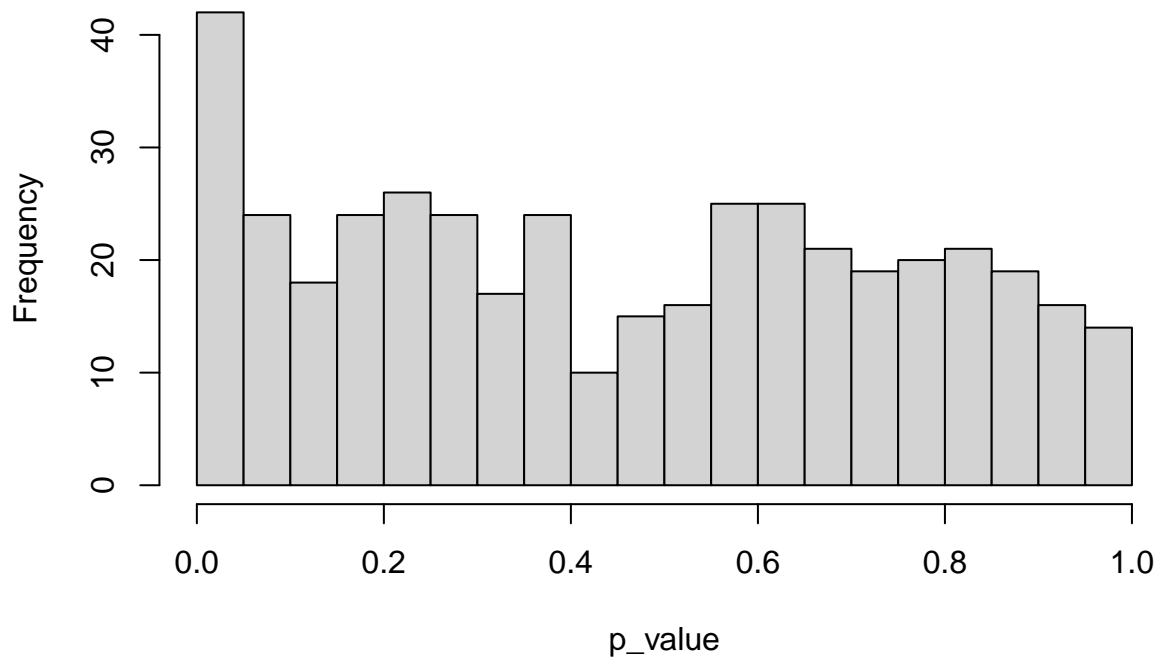
new_bmi_val[29] = mean(new_bmi_val, na.rm = TRUE)

p_value = vector()

for (i in 1:length(casecont_col)) {
  col_name = casecont_col[i]
  otu = myTcaseCont[, c(col_name)]
  myLmBMI = lm(new_bmi_val~otu)
  pVal = anova( myLmBMI)$"Pr(>F)"[1]
  p_value[i]=pVal
}

hist(p_value, breaks = 25)
```

### Histogram of p\_value



```
pValue_adj = p.adjust(p_value, method = 'BH')
```

```
sum(pValue_adj <= 0.1)
```

```
## [1] 0
```

The p\_values before applying p.adjust() do not seem to be entirely uniformly distributed since we can see a peak at pvalue = 0.0.