Assignment 4

Tommy Maaiveld, Krishnakanth Sasi, Halil Kaan Kara, Group 6

Question 1

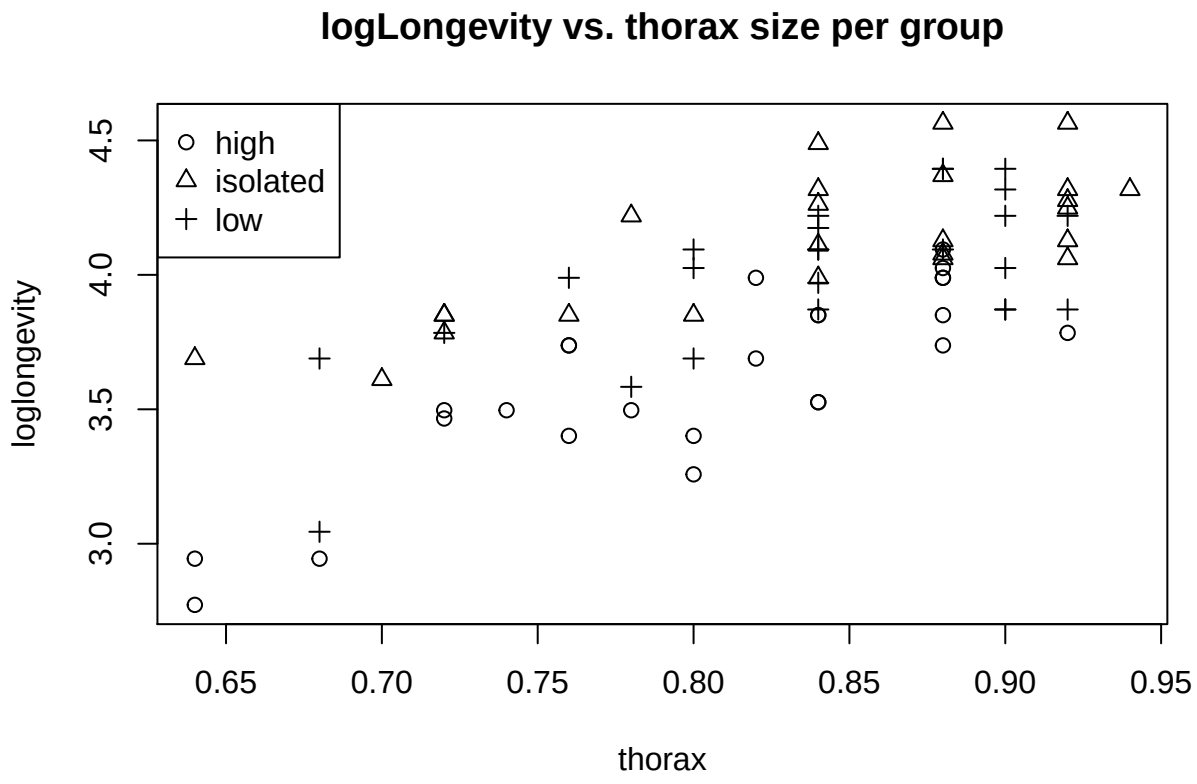
Section 1

```
fruitflies = read.table(file="data/fruitflies.txt", header=TRUE)
fruitflies = cbind(fruitflies, log(fruitflies[,2])); names(fruitflies)[4]="loglongevity"
head(fruitflies,3) # some output deleted
```

```
##   thorax longevity activity loglongevity
## 1   0.64        40 isolated     3.688879
## 2   0.70        37 isolated     3.610918
## 3   0.72        44 isolated     3.784190
```

Section 2

```
plot(loglongevity~thorax,pch=unclass(activity),
     main = "logLongevity vs. thorax size per group")
legend('topleft',legend=levels(fruitflies$activity),pch=1:3)
```



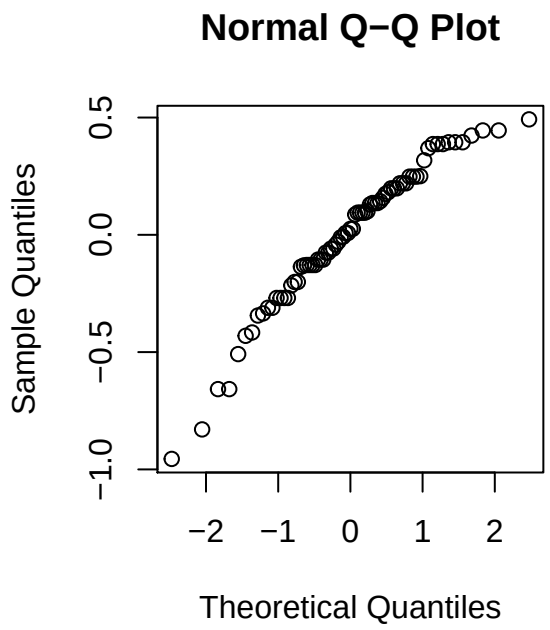
The plot shows a linear correlation between thoractic length (`thorax`) and log longevity. It seems to indicate that flies with the `activity` factor set to `high` live less long than those with `low`, which in turn score lower than those with `isolated`, assuming equal thoractic length between specimens.

Section 3

```
fruitfliesaov = lm(loglongevity~activity, data=fruitflies)
attach(fruitfliesaov)
anova(fruitfliesaov)
```

```
## Analysis of Variance Table
##
## Response: loglongevity
##           Df Sum Sq Mean Sq F value    Pr(>F)
## activity   2 3.6665  1.8333  19.421 1.798e-07 ***
## Residuals 72 6.7966  0.0944
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
qqnorm(residuals(fruitfliesaov));
```



According to this analysis, `activity` seems likely to have an effect on `loglongevity`, since the p-value < 0.05 (p-value $\approx 1.8 \times 10^{-7}$). Thus, sexual activity seems to influence longevity. The QQ-plot of the residuals looks relatively normal.

Section 4

the analysis shows that sexual activity decreases longevity in fruitflies, since the `activity` factor affects `loglongevity` negatively for levels `low` and `high` compared to level `isolated`.

```
summary(fruitfliesaov)
```

```
##
## Call:
## lm(formula = loglongevity ~ activity, data = fruitflies)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.95531 -0.13338  0.02552  0.20891  0.49222
##
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)       3.60212    0.06145  58.621  < 2e-16 ***
## activityisolated  0.51722    0.08690   5.952 8.82e-08 ***
## activitylow       0.39771    0.08690   4.577 1.93e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.3072 on 72 degrees of freedom
## Multiple R-squared:  0.3504, Adjusted R-squared:  0.3324
## F-statistic: 19.42 on 2 and 72 DF,  p-value: 1.798e-07
```

the rounded longevity estimates for each level of the factor `activity` are 3.6 for fruitflies of level `high`, 4 for fruitflies of level `low`, and 4.12 for fruitflies of level `isolated`. Lifespan seems to decrease for factor levels representing higher levels of sexual activity.

Section 5

```
fruitfliesfullaov = lm(loglongevity~thorax+activity, data=fruitflies)
attach(fruitfliesfullaov)
anova(fruitfliesfullaov)
```

```
## Analysis of Variance Table
##
## Response: loglongevity
##           Df Sum Sq Mean Sq F value Pr(>F)
## thorax     1 5.4322  5.4322 132.175 <2e-16 ***
## activity   2 2.1129  1.0565  25.705  4e-09 ***
## Residuals 71 2.9180  0.0411
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
summary(fruitfliesfullaov)$coef
```

```
##                   Estimate Std. Error  t value     Pr(>|t|)
```

```
## (Intercept)      1.2189341 0.24864843 4.902239 5.787723e-06
## thorax           2.9789877 0.30665052 9.714602 1.138552e-14
## activityisolated 0.4099810 0.05839296 7.021070 1.074333e-09
## activitylow      0.2857017 0.05848770 4.884817 6.183669e-06
```

The output gives the following rounded estimates for the model coefficients μ, $\beta\,\alpha_1$ and α_2:
$\mu = 1.219$
$\beta = 2.979$
$\alpha_{low} = 0.286$
$\alpha_{iso} = 0.41$
Where μ is the estimate for an average fly with high sexual activity, α_k are the influences of each factor level k, and of β is the parameter applied to the explanatory variable values $X_{k,n}$. The p-values are all virtually zero, meaning there is almost no risk of a type I error. Sexual activity is very likely to influence longevity, regardless of whether thorax length is taken into account.

Section 6

Sexual activity decreases longevity, since the estimate is lowest for flies with high sexual activity, and highest for isolated flies.

```
mean(fruitflies[,1]); min(fruitflies[,1])
```

```
## [1] 0.8245333
```

```
## [1] 0.64
```

For an average fly with a thorax length ≈ 0.82, the value for parameter β given above can be used in a sum to compute all three estimates for loglongevity based on variable X_{avg}:

$\mu + \beta * X_{avg} = 1.219 + 2.979 * 0.82 = 3.675$ (high sexual activity)

$\mu + \beta * X_{avg} + \alpha_1 = 1.219 + 2.979 * 0.82 + 0.286 = 3.961$ (low sexual activity)

$\mu + \beta * X_{avg} + \alpha_2 = 1.219 + 2.979 * 0.82 + 0.41 = 4.085$ (isolated specimen)

In order to compute the estimates for a fly as small as the smallest fly in the dataset, the term X is substituted with X_{min}, the thorax size of the smallest fly in the dataset (0.64).

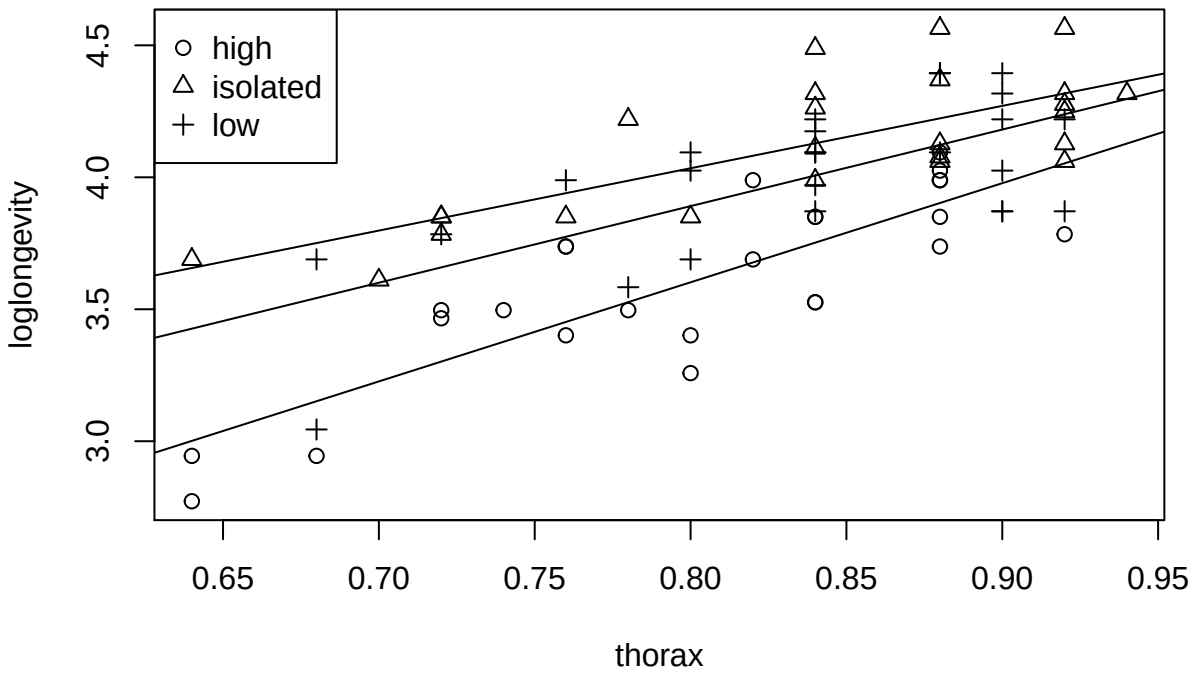
$\mu + \beta * X_{min} = 1.219 + 2.979 * 0.64 = 3.125$ (high sexual activity)

$\mu + \beta * X_{min} + \alpha_1 = 1.219 + 2.979 * 0.64 + 0.286 = 3.411$ (low sexual activity)

$\mu + \beta * X_{min} + \alpha_2 = 1.219 + 2.979 * 0.64 + 0.41 = 3.535$ (isolated specimen)

Section 7

```
plot(loglongevity~thorax,pch=unclass(activity))
for (i in c("high", "low", "isolated"))
  abline(lm(loglongevity~thorax,data=fruitflies[fruitflies$activity==i,]))
legend('topleft',legend=levels(fruitflies$activity),pch=1:3)
```

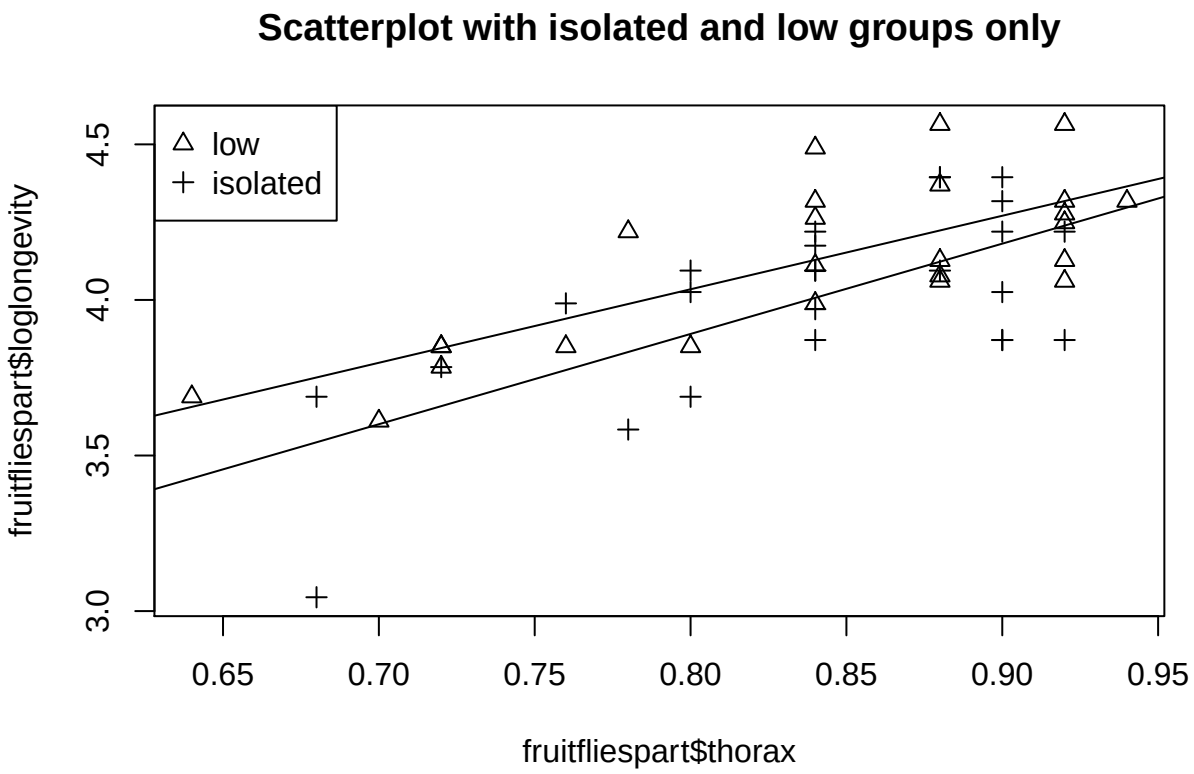


The given plot shows fit lines for each level of the `activity` factor. Thorax length correlates positively with longevity (bigger flies live longer), meaning β is expected to be nonzero. The fit lines in the plot converge slightly, although the true lines could still be parallel. In other words, the parameter β is similar for each factor level, meaning the dependence on thorax length is similar on each level. This means that the lifespan of a given fruitfly is affected by its sexual activity, regardless of its size, and bigger flies live longer within each factor level. The slight convergence indicates that larger flies seem less affected by differing levels of sexual activity.

Section 8

By removing experimental units with factor level `high`, the differences between two similar groups (the group with no sexual activity and with low sexual activity) can be analysed. The difference between these two groups is particularly difficult to detect without considering the explanatory variable `thorax`, as can be seen in this scatterplot.

```
fruitfliespart = fruitflies[1:50,]

plot(fruitfliespart$loglongevity~fruitfliespart$thorax,
     pch=unclass(fruitfliespart$activity),
     main="Scatterplot with isolated and low groups only")
for (i in c("low", "isolated"))
  abline(lm(loglongevity~thorax,
            data=fruitfliespart[fruitfliespart$activity==i,]))
legend('topleft',legend=c("low","isolated"),pch=2:3)
```



To show these, ANOVA tests are conducted for the first 50 fruit flies (groups `isolated` and `low`). The first analysis does not take `thorax` into account, while the second analysis considers its effects.

```
ffnothoraxpartaov = lm(loglongevity~activity, data=fruitfliespart)
anova(ffnothoraxpartaov)
```

```
## Analysis of Variance Table
##
## Response: loglongevity
##           Df Sum Sq  Mean Sq F value Pr(>F)
## activity   1 0.1785 0.178542  2.2376 0.1412
## Residuals 48 3.8300 0.079791
```

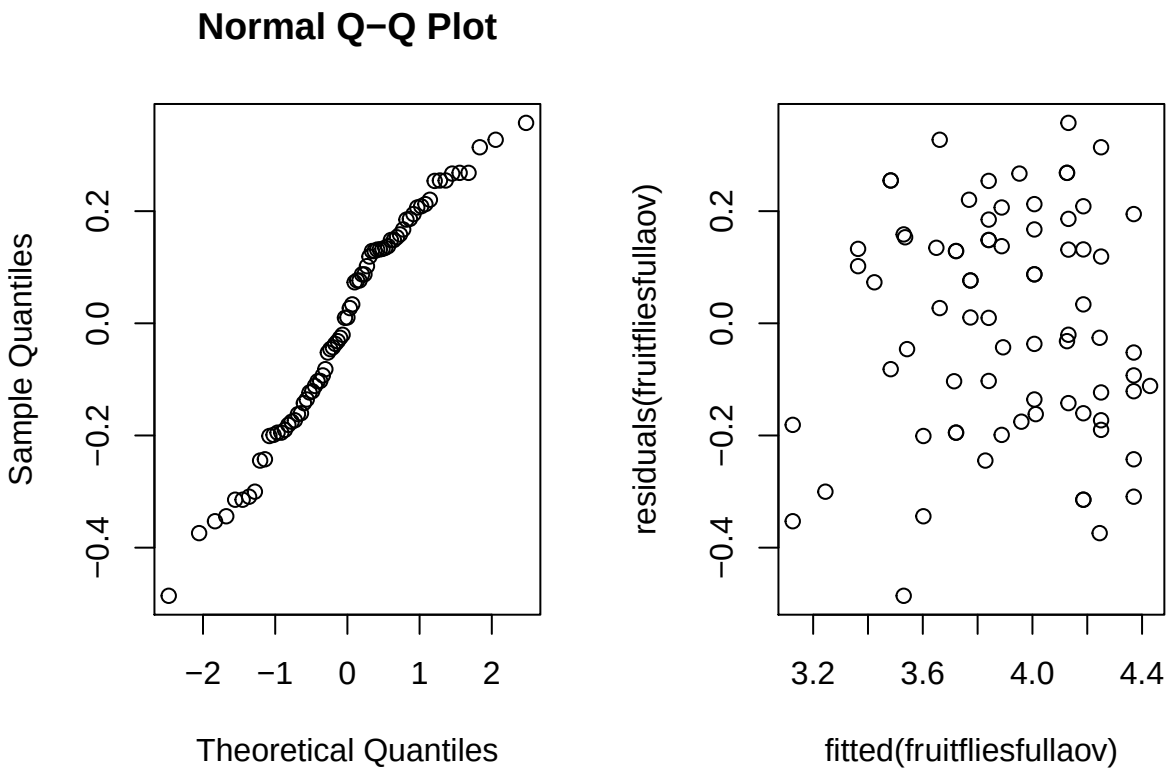
```
fruitfliespartaov = lm(loglongevity~thorax+activity, data=fruitfliespart)
anova(fruitfliespartaov)
```

```
## Analysis of Variance Table
##
## Response: loglongevity
##           Df  Sum Sq Mean Sq F value    Pr(>F)
## thorax     1 1.92385 1.92385  47.751 1.118e-08 ***
## activity   1 0.19109 0.19109   4.743   0.03447 *
## Residuals 47 1.89359 0.04029
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

The p-value for the non-thorax test indicates that the effect of `activity` is not significant (p-value $\approx$ 0.14). Including the `activity` factor produces a p-value $\approx$ 0.034, meaning that once noise introduced by the explanatory variable `thorax` is considered, the difference between the experimental groups becomes significant.

Section 9

```
par(mfrow=c(1,2))
qqnorm(residuals(fruitfliesfullaov))
plot(fitted(fruitfliesfullaov),residuals(fruitfliesfullaov))
```



From these plots, the assumption of normality does not seem to be violated, and the data appears reasonably homoskedastic. The variance of the residuals does not increase greatly for higher fitted values.

Section 10

```
ffnonlogaov = lm(longevity~thorax+activity, data=fruitflies)
attach(ffnonlogaov)
anova(ffnonlogaov)
```

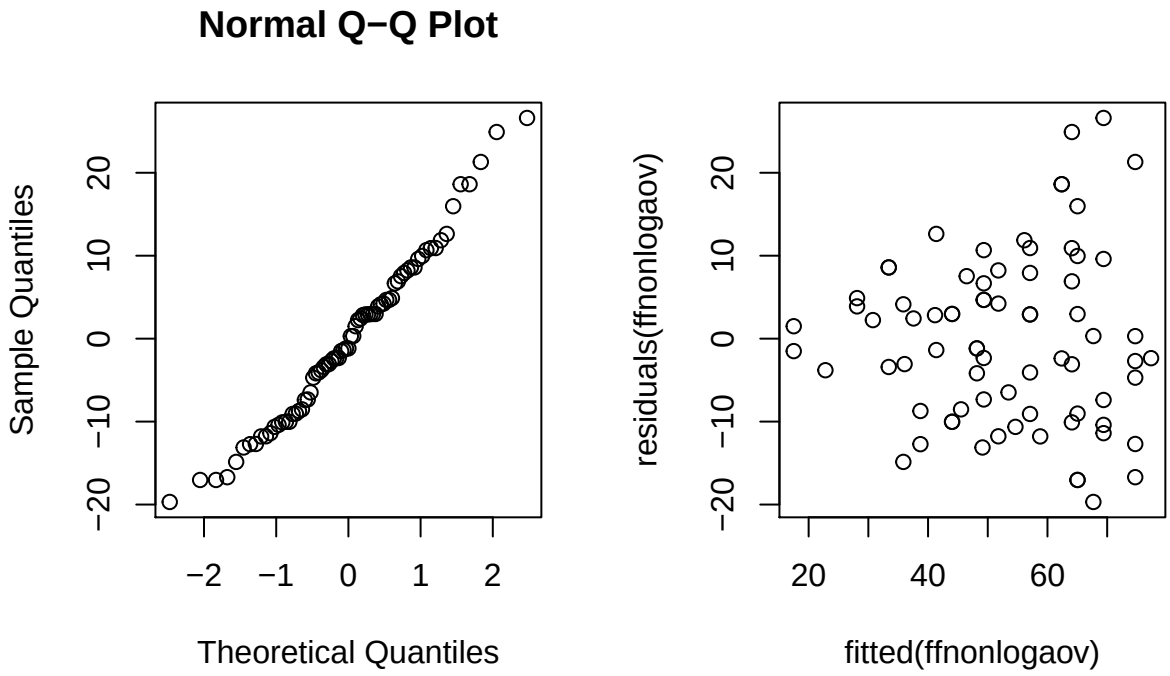
```
## Analysis of Variance Table
```

```
##
## Response: longevity
##           Df  Sum Sq Mean Sq F value    Pr(>F)
## thorax     1 10959.3 10959.3 101.409 2.557e-15 ***
## activity   2  4966.7  2483.4  22.979 2.016e-08 ***
## Residuals 71  7673.0   108.1
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
summary(ffnonlogaov)$coef
```

```
##                    Estimate Std. Error   t value     Pr(>|t|)
## (Intercept)      -67.37460  12.750486 -5.284081 1.325213e-06
## thorax           132.61825  15.724785  8.433708 2.623943e-12
## activityisolated  20.06574   2.994343  6.701218 4.126605e-09
## activitylow       13.05355   2.999201  4.352344 4.434649e-05
```

```
par(mfrow=c(1,2))
qqnorm(residuals(ffnonlogaov))
plot(fitted(ffnonlogaov),residuals(ffnonlogaov))
```



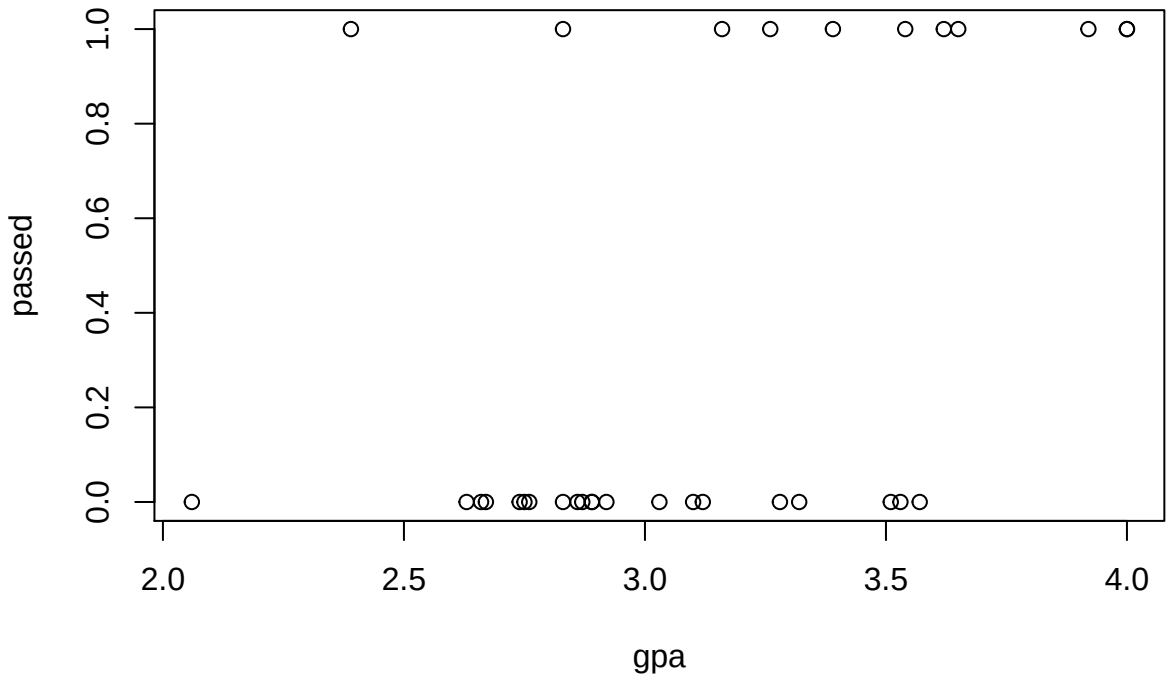
In the plot on the right, the residuals appear more heteroskedastic than in the non-log transformed data, since the variance of the residuals seems to increase as the fitted parameter on the x-axis increases. Log transforming the `longevity` variable improves the homoskedasticity of the data and decreases the variance in residuals for increasingly large fitted values. Normality does not seem significantly affected by the log transformation.

Question 2

Section 1

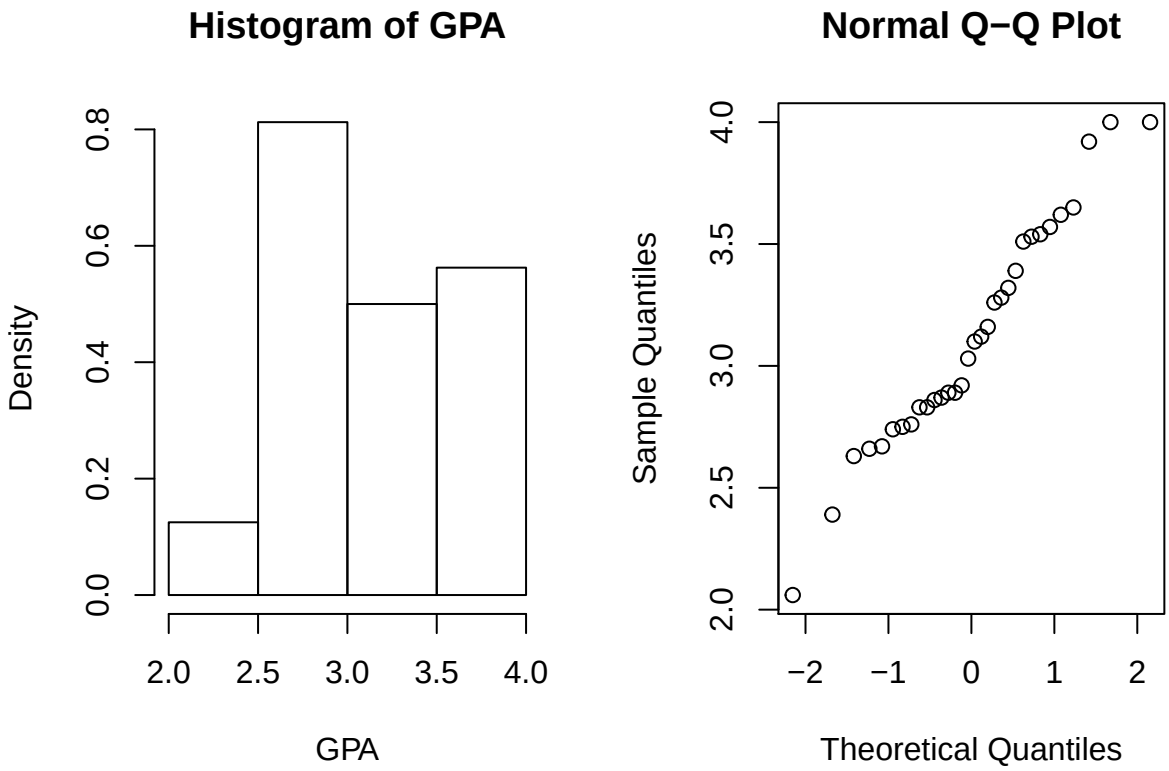
The given data set `psi.txt` consists of one binary response and two explanatory variables which one of them is also a binary variable. A scatterplot of response variable `passed` as explained by variable `gpa` can be seen below. A linear formula can not be fitted to this data, since the response variable is binomial.

```
psiData = read.table("./data/psi.txt", header = TRUE)
psiDataNonFactor = data.frame(psiData)

psiData$passed = ifelse(test=psiData$passed == 1, yes="Pass", no="Fail")
psiData$passed = as.factor(psiData$passed)

psiData$psi = ifelse(test=psiData$psi == 1, yes="Yes", no="No")
psiData$psi = as.factor(psiData$psi)

par(mfrow=c(1,1))
plot(passed ~ gpa, data = psiDataNonFactor)
```



The numeric variable `gpa` seems to be from a standard normal distribution and its histogram and QQ-Plot can be seen below. As a first step, binary variables are converted into factors.

```
par(mfrow=c(1,2))
hist(psiData$gpa, freq = FALSE, xlab = "GPA", main = "Histogram of GPA")
qqnorm(psiData$gpa)
```



A contingency table of the two binary variables can be seen in the table below. From this table it can be said that psi seems effective, since more students have passed upon receiving psi.

```
xtabs(~passed + psi, data = psiData)
```

```
##       psi
## passed No Yes
##   Fail 15   6
##   Pass  3   8
```

Section 2

The output of the basic logistic regression model fitted using the `glm` command using both numeric and binary variables can be seen below. The model is trained on training data set and validated on test data set as can be seen below. The test data set uses 10% of the whole data set without replacement.

```
# Fit the model
logRegModel = glm(passed ~ psi + gpa, data = psiData, family = "binomial")
logSummary = summary(logRegModel)
logSummary
```

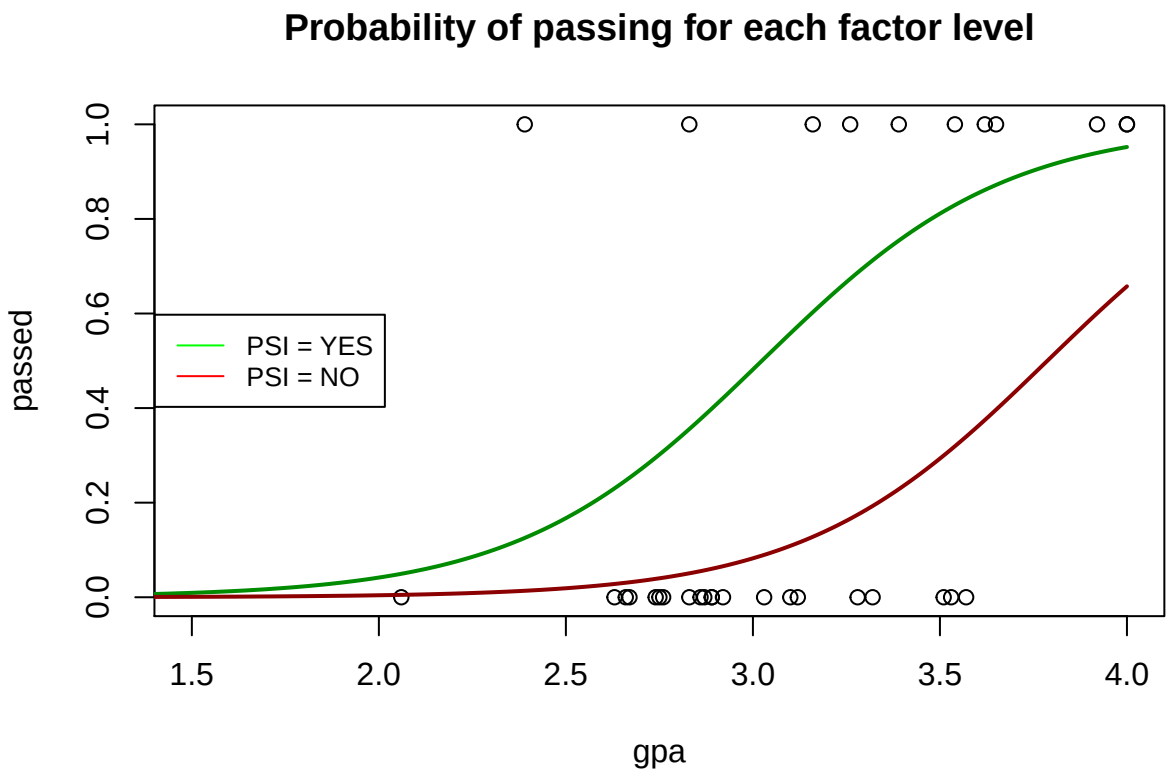
```
##
## Call:
## glm(formula = passed ~ psi + gpa, family = "binomial", data = psiData)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.8396  -0.6282  -0.3045   0.5629   2.0378
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)  -11.602      4.213  -2.754  0.00589 **
## psiYes         2.338      1.041   2.246  0.02470 *
## gpa            3.063      1.223   2.505  0.01224 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 41.183  on 31  degrees of freedom
## Residual deviance: 26.253  on 29  degrees of freedom
## AIC: 32.253
##
## Number of Fisher Scoring iterations: 5
```

The output shows that the model corresponds to the equation given below.

$P(Y) = \Psi(-11.6015646 + (2.3377756) * psi + (3.0633672) * gpa)$

The predictions of the model can be seen in the graph below, where the red line represents the probability of passing without psy dependent on GPA, while the green line shows the probability of passing with.

```
newdat1 = data.frame(gpa=seq(0, 4, len=300))
newdat2 = data.frame(gpa=seq(0, 4, len=300))

newdat1$psi = 1
newdat1$psi = ifelse(test=newdat1$psi == 1, yes="Yes", no="No")
newdat1$psi = as.factor(newdat1$psi)
newdat2$psi = 0
newdat2$psi = ifelse(test=newdat2$psi == 1, yes="Yes", no="No")
newdat2$psi = as.factor(newdat2$psi)

newdat1$passed = predict(logRegModel, newdata=newdat1, type="response")
newdat2$passed = predict(logRegModel, newdata=newdat2, type="response")

par(mfrow=c(1,1))
plot(passed ~ gpa, data = psiDataNonFactor, xlim = c(1.5 ,4),
     main="Probability of passing for each factor level")
lines(passed ~ gpa, data = newdat1, col="green4", lwd=2)
lines(passed ~ gpa, data = newdat2, col="red4", lwd=2)
legend(x = "left", legend=c("PSI = YES", "PSI = NO"),
       col=c("green", "red"), lty=1:1, cex=0.8)
```



Section 3

From the table given in Section 1, we can calculate the probability of a student passing the assignment given he or she received psi is $P(Passed = TRUE|PSI = TRUE) = 0.5714286$. From the predictions made with the model given in Section 2, we see higher probabilities for students which received psi. Also, the graph for logistic curve given in Section 2 clearly demonstrates the positive effect of psi on passing ratio. Finally, we can check the coefficient of the equation for psi which is 2.3377756. All this information points toward that psi works.

Section 4

```
testSec4 = read.table("./data/psi-section4.txt", header = TRUE)
testSec4$passed = ifelse(test=testSec4$passed == 1, yes="Pass", no="Fail")
testSec4$passed = as.factor(testSec4$passed)
testSec4$psi = ifelse(test=testSec4$psi == 1, yes="Yes", no="No")
testSec4$psi = as.factor(testSec4$psi)

testSec4 # Passed column is irrelevant in this case
```

```
##   passed psi gpa
## 1   Fail Yes   3
## 2   Fail  No   3
```

```
predicted = predict(logRegModel, testSec4, type = "response")
predicted
```

```
##          1          2
## 0.48158645 0.08230274
```

The probabilities for one student passing with a gpa of 3 and receiving psi and one student having gpa of 3 and not receiving psi are 0.482 and 0.082 respectively.

Section 5

Estimation of relative change in odds can be seen below with the command. This command yields two numbers for each explanatory variable.

```
odds = round(exp(logRegModel$coefficients), 3)
odds
```

```
## (Intercept)      psiYes         gpa
##       0.000      10.358      21.399
```

The output shows that if the student has received psi, the odds of that student passing increase by a factor of 10.358. Therefore, it can be said that psi works better than the standard teaching method since odds of a student increase upon receiving psi, regardless of the student's gpa. Also, we can say for a one unit increase in gpa of a student, the odds of that student passing increase by a factor of 21.399 of the teaching method.

Section 6

The table of the alternate analysis can be seen below. This table looks familiar compared to the table shown in Section 1. With the table from Section 1 in mind, the numbers 15 and 6 can be assumed to be the students that failed regardless of the teaching method. This table shows the combinations of the binary response variable and the binary explanatory variable.

```
x = matrix(c(3, 15, 8, 6), 2, 2)
x
```

```
##      [,1] [,2]
## [1,]    3    8
## [2,]   15    6
```

```
fisher.test(x)
```

```
##
##  Fisher's Exact Test for Count Data
##
## data:  x
## p-value = 0.0265
## alternative hypothesis: true odds ratio is not equal to 1
## 95 percent confidence interval:
##  0.02016297 0.95505763
## sample estimates:
## odds ratio
##  0.1605805
```

The outcome of Fisher's Exact Test with the p-value of 0.0265, describes that the difference of the probabilities of the matrix are statistically significant. This means that the proportions for one variable are not the same for different values of the second variable, meaning the variables are not independent of each other.

Section 7

Fisher's Exact Test yields whether the proportions of two nominal variables are different depending on the value of the other variable. Thus, it is a test of independence. This test is not appropriate for this case since we are looking for the probability of a student passing or failing based on the teaching method and gpa.

Section 8

Fisher's Exact Test

```
- Advantage: Identifies the significance of a relationship.
- Disadvantage: It fits for small data sets but is computationally expensive for large
data sets.
```

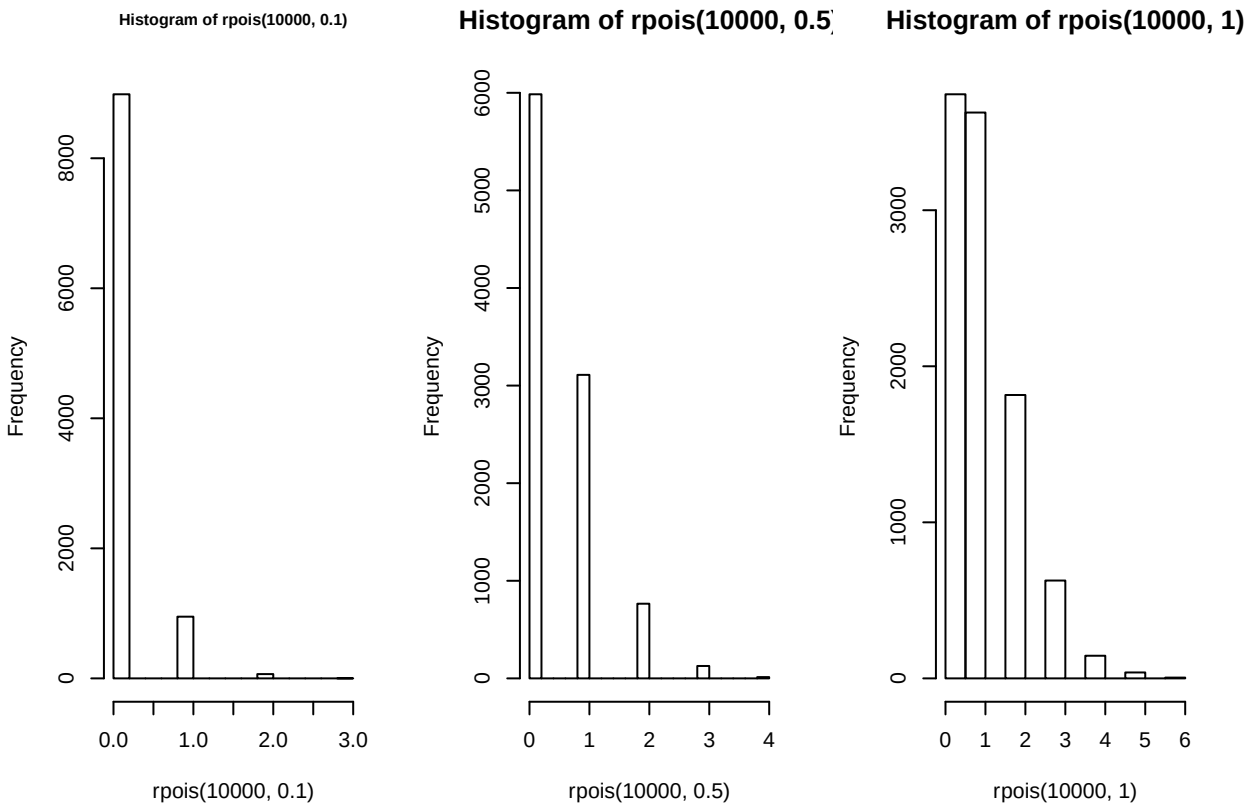
Logistic Regression

```
- Advantage: Constructs a model that measure the relationship between the dependent
variable and the independent variables by estimating the probabilities using a
logistic function.
- Disadvantage: Logistic Regression will not work if there is a feature that
completely separates the two classes.
```

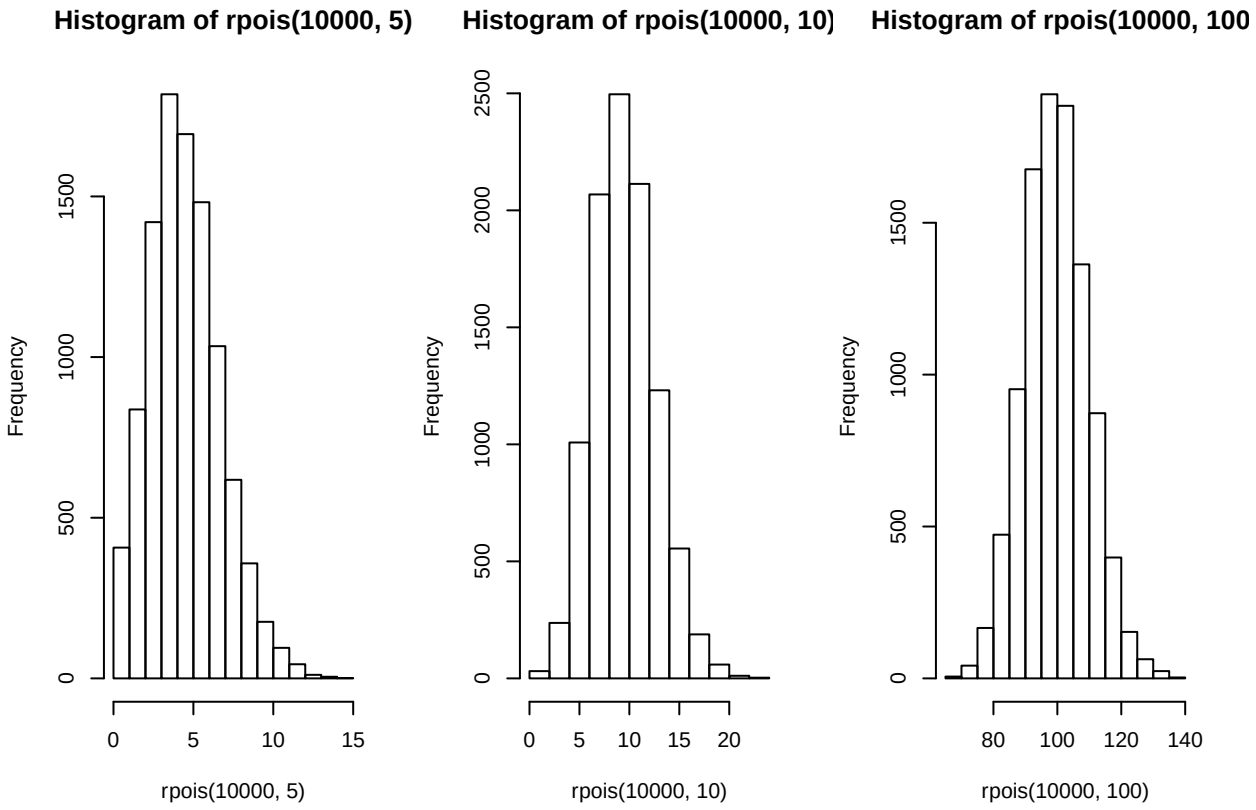
Question 3

Section 1

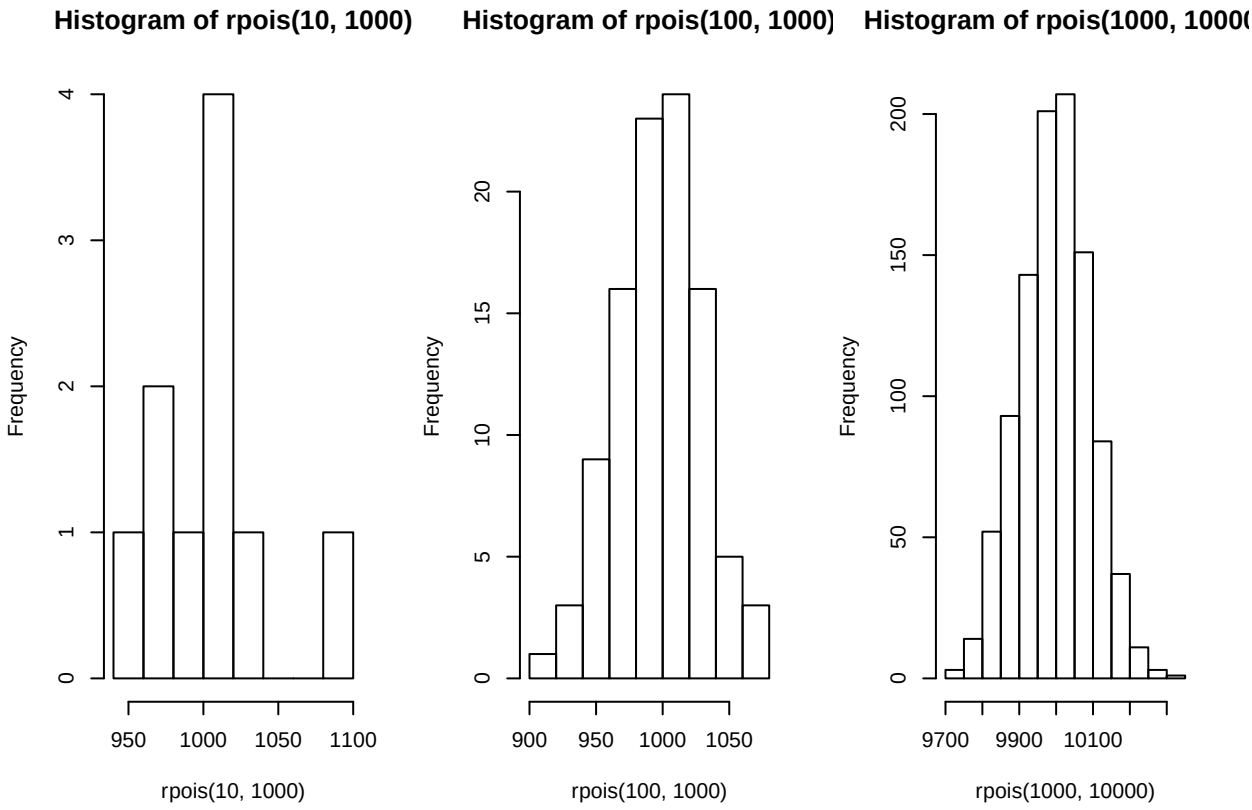
```
par(mfrow=c(1,3))
hist(rpois(10000,.1), cex.main=.8); hist(rpois(10000,.5)); hist(rpois(10000,1))
```




```
hist(rpois(10000,5)); hist(rpois(10000,10)); hist(rpois(10000,100))
```




```
hist(rpois(10,1000)); hist(rpois(100,1000)); hist(rpois(1000,10000))
```



For larger values of λ, the distribution is similar to a normal distribution with the mean and variance both equal to λ. Parameter n is of limited influence - it merely determines the amount of values to be sampled from the Poisson distribution. So long as a reasonable amount of points are sampled, the same distribution should emerge for equal λ.

Section 2

In order for the distribution of a randomly distributed variable Y to be in a location-scale family as a given random variable X, Y must have the same distribution as $a + bX$ for some parameters a and b (in other words, $Y \overset{d}{=} a + bX$, where $Y \overset{d}{=}$ means 'equal in distribution'.

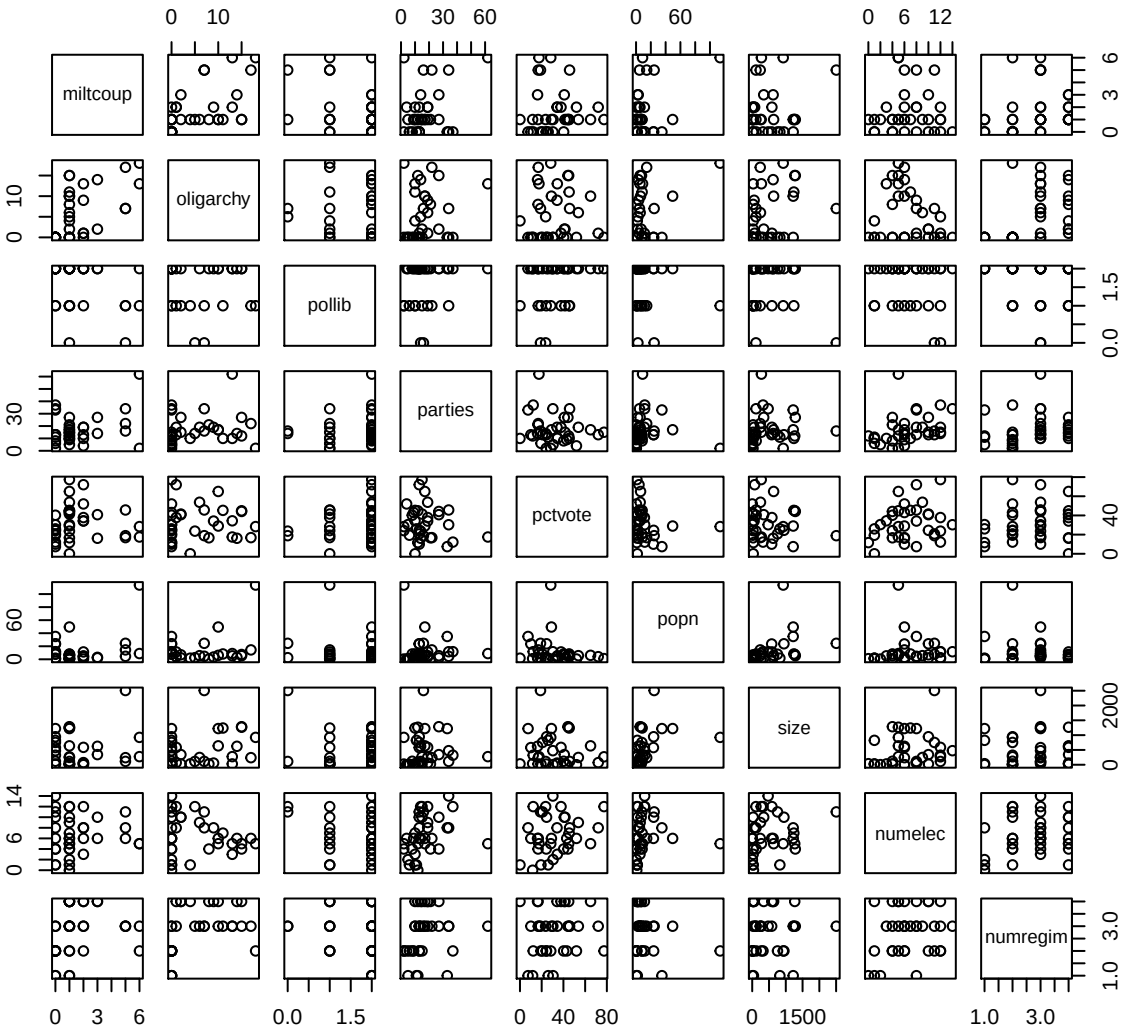
In the case of the Poisson distribution, the distribution is both scaled by parameter λ, since the mean and variance are both equal to λ. Thus, it can be said that, given a variable Y and a variable X that follow a Poisson distribution, $Y \overset{d}{=} \lambda X$, which satisfies the above condition for location-scale families.

However, for very small values of lambda ($\lambda < 1$),where the distribution looks less similar to a normal distribution, it may prove difficult to produce Poisson distributions with larger λ values via a linear transformation, as a scaling transformation may not be able to fit a normal distribution.

Section 3

```
africa = read.table("data/africa.txt",header=TRUE)
africaglm=glm(miltcoup~oligarchy+pollib+parties+pctvote+popn+size+numelec+numregim,
```

```
                family=poisson,data=africa)

plot(africa)
```




```
summary(africaglm)
```

```
##
## Call:
## glm(formula = miltcoup ~ oligarchy + pollib + parties + pctvote +
##     popn + size + numelec + numregim, family = poisson, data = africa)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.3443  -0.9542  -0.2587   0.3905   1.6953
##
```

```
## Coefficients:
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept) -0.5102693  0.9053301  -0.564  0.57301
## oligarchy    0.0730814  0.0345958   2.112  0.03465 *
## pollib      -0.7129779  0.2725635  -2.616  0.00890 **
## parties      0.0307739  0.0111873   2.751  0.00595 **
## pctvote      0.0138722  0.0097526   1.422  0.15491
## popn         0.0093429  0.0065950   1.417  0.15658
## size        -0.0001900  0.0002485  -0.765  0.44447
## numelec     -0.0160783  0.0654842  -0.246  0.80605
## numregim     0.1917349  0.2292890   0.836  0.40303
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 1)
##
##     Null deviance: 65.945  on 35  degrees of freedom
## Residual deviance: 28.668  on 27  degrees of freedom
## AIC: 111.48
##
## Number of Fisher Scoring iterations: 6
```

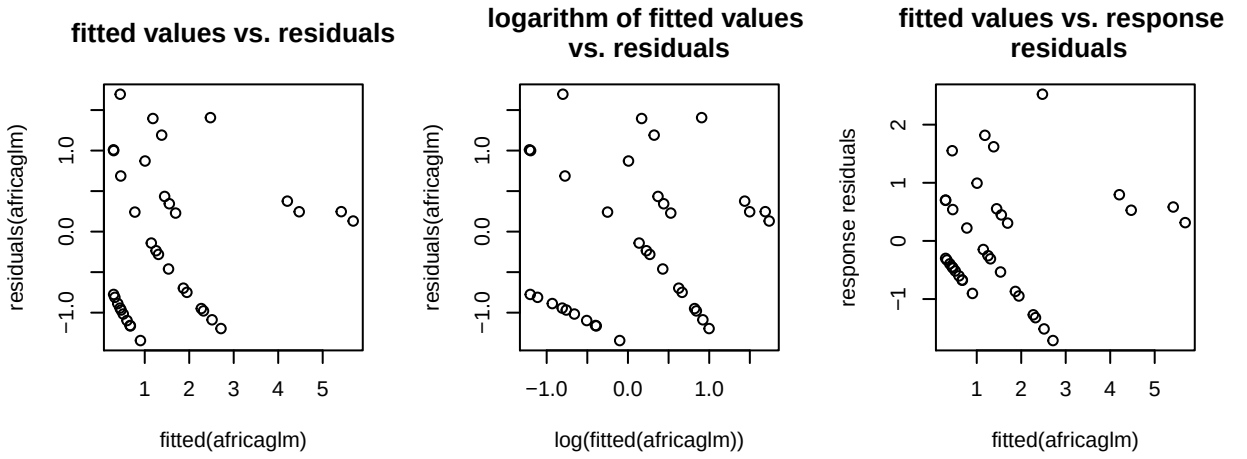
```
confint(africaglm)
```

```
##                     2.5 %       97.5 %
## (Intercept) -2.4335049109  1.148089620
## oligarchy    0.0045915288  0.141483576
## pollib      -1.2570629668 -0.182012570
## parties      0.0080568606  0.052321186
## pctvote     -0.0054171503  0.032940743
## popn        -0.0038404317  0.022244262
## size        -0.0007146351  0.000272539
## numelec     -0.1438197483  0.114689702
## numregim    -0.2632334399  0.643070807
```

```
coef(africaglm)
```

```
##   (Intercept)     oligarchy        pollib       parties       pctvote
## -0.5102692854  0.0730813725 -0.7129778804  0.0307739289  0.0138722128
##          popn          size       numelec      numregim
##  0.0093429334 -0.0001899975 -0.0160783349  0.1917349158
```

```
# Assumption checks:
par(mfrow=c(1,3))
plot(fitted(africaglm),residuals(africaglm), main='fitted values vs. residuals')
plot(log(fitted(africaglm)),residuals(africaglm), main='logarithm of fitted values \nvs. residuals')
plot(fitted(africaglm),residuals(africaglm,type="response"), main='fitted values vs. response \n residua
```



Performing visual checks on the residuals of the model shows some odd relationships between the relationships and the fitted values, as the variance of the residuals doesn't seem to increase for higher fitted values. This is expected under a Poisson distribution, as higher fitted values correspond to higher variances as lambda is modeled differently for each observation. The first plot also shows some collinearity between variables such as `popn` and `pollib`.

Section 4

```
summary(glm(miltcoup~oligarchy+pollib+parties+pctvote+popn+size+numelec+numregim,
            family=poisson,data=africa))
```

```
##
## Call:
## glm(formula = miltcoup ~ oligarchy + pollib + parties + pctvote +
##     popn + size + numelec + numregim, family = poisson, data = africa)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.3443  -0.9542  -0.2587   0.3905   1.6953
##
## Coefficients:
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept) -0.5102693  0.9053301  -0.564  0.57301
## oligarchy    0.0730814  0.0345958   2.112  0.03465 *
## pollib      -0.7129779  0.2725635  -2.616  0.00890 **
## parties      0.0307739  0.0111873   2.751  0.00595 **
## pctvote      0.0138722  0.0097526   1.422  0.15491
## popn         0.0093429  0.0065950   1.417  0.15658
## size        -0.0001900  0.0002485  -0.765  0.44447
## numelec     -0.0160783  0.0654842  -0.246  0.80605
## numregim     0.1917349  0.2292890   0.836  0.40303
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 1)
```

```
##
##     Null deviance: 65.945  on 35  degrees of freedom
## Residual deviance: 28.668  on 27  degrees of freedom
## AIC: 111.48
##
## Number of Fisher Scoring iterations: 6
```

```
# `numelec` has the highest p-value, and is removed.
summary(glm(miltcoup~oligarchy+pollib+parties+pctvote+popn+size+numregim,
            family=poisson,data=africa))
```

```
##
## Call:
## glm(formula = miltcoup ~ oligarchy + pollib + parties + pctvote +
##     popn + size + numregim, family = poisson, data = africa)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.3997  -0.9381  -0.2666   0.4220   1.6998
##
## Coefficients:
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept) -0.6078028  0.8239267  -0.738  0.46070
## oligarchy    0.0781368  0.0277656   2.814  0.00489 **
## pollib      -0.6773897  0.2290130  -2.958  0.00310 **
## parties      0.0296786  0.0102888   2.885  0.00392 **
## pctvote      0.0131290  0.0092895   1.413  0.15756
## popn         0.0089313  0.0063746   1.401  0.16120
## size        -0.0002021  0.0002436  -0.830  0.40682
## numregim     0.1758198  0.2210498   0.795  0.42639
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 1)
##
##     Null deviance: 65.945  on 35  degrees of freedom
## Residual deviance: 28.728  on 28  degrees of freedom
## AIC: 109.54
##
## Number of Fisher Scoring iterations: 5
```

```
# `numregim` is removed next.
summary(glm(miltcoup~oligarchy+pollib+parties+pctvote+popn+size,
            family=poisson,data=africa))
```

```
##
## Call:
## glm(formula = miltcoup ~ oligarchy + pollib + parties + pctvote +
##     popn + size, family = poisson, data = africa)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.3522  -0.9651  -0.1945   0.4833   1.6179
```

```
##
## Coefficients:
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept) -0.1126871  0.5163030  -0.218 0.827228
## oligarchy    0.0859620  0.0259100   3.318 0.000908 ***
## pollib      -0.6894029  0.2278572  -3.026 0.002481 **
## parties      0.0291944  0.0101954   2.863 0.004190 **
## pctvote      0.0141588  0.0091980   1.539 0.123723
## popn         0.0062736  0.0053994   1.162 0.245272
## size        -0.0001950  0.0002425  -0.804 0.421378
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 1)
##
##     Null deviance: 65.945  on 35  degrees of freedom
## Residual deviance: 29.363  on 29  degrees of freedom
## AIC: 108.17
##
## Number of Fisher Scoring iterations: 5
```

```
# removing `size`
summary(glm(miltcoup~oligarchy+pollib+parties+pctvote+popn,
            family=poisson,data=africa))
```

```
##
## Call:
## glm(formula = miltcoup ~ oligarchy + pollib + parties + pctvote +
##     popn, family = poisson, data = africa)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.4109  -0.9943  -0.1399   0.5516   1.6125
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept) -0.244466   0.495708  -0.493  0.62190
## oligarchy    0.083168   0.025437   3.270  0.00108 **
## pollib      -0.652830   0.221234  -2.951  0.00317 **
## parties      0.029800   0.010294   2.895  0.00379 **
## pctvote      0.013842   0.009282   1.491  0.13591
## popn         0.005587   0.005378   1.039  0.29883
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 1)
##
##     Null deviance: 65.945  on 35  degrees of freedom
## Residual deviance: 30.044  on 30  degrees of freedom
## AIC: 106.85
##
## Number of Fisher Scoring iterations: 5
```

```r
# removing `popn`
summary(glm(miltcoup~oligarchy+pollib+parties+pctvote,
            family=poisson,data=africa))
```

```
##
## Call:
## glm(formula = miltcoup ~ oligarchy + pollib + parties + pctvote,
##     family = poisson, data = africa)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.5456  -0.9841  -0.1881   0.5948   1.6705
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept) -0.093657   0.463279  -0.202  0.83979
## oligarchy    0.095358   0.022421   4.253 2.11e-05 ***
## pollib      -0.666615   0.217564  -3.064  0.00218 **
## parties      0.025630   0.009502   2.697  0.00699 **
## pctvote      0.012134   0.009056   1.340  0.18031
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 1)
##
##     Null deviance: 65.945  on 35  degrees of freedom
## Residual deviance: 31.081  on 31  degrees of freedom
## AIC: 105.89
##
## Number of Fisher Scoring iterations: 5
```

```r
# removing `pctvote`
summary(glm(miltcoup~oligarchy+pollib+parties,
            family=poisson,data=africa))
```

```
##
## Call:
## glm(formula = miltcoup ~ oligarchy + pollib + parties, family = poisson,
##     data = africa)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.3583  -1.0424  -0.2863   0.6278   1.7517
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)  0.251377   0.372689   0.674  0.50000
## oligarchy    0.092622   0.021779   4.253 2.11e-05 ***
## pollib      -0.574103   0.204383  -2.809  0.00497 **
## parties      0.022059   0.008955   2.463  0.01377 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
```

```
## (Dispersion parameter for poisson family taken to be 1)
##
##     Null deviance: 65.945  on 35  degrees of freedom
## Residual deviance: 32.856  on 32  degrees of freedom
## AIC: 105.66
##
## Number of Fisher Scoring iterations: 5
```

The remaining parameters appear significant, as their p-value is lower than 0.05. By examining the collinearity of the remaining variables using the plot below, it appears that none of the remaining variables are excessively collinear.

```
plot(africa[,1:4])
```