Experimental Design and Data Analysis: Assignment 2

Andrew Bedard(2566978) & Simone van Gompel(2567525)
Group 19

April 29, 2015

1 Exercise 1

1

Using data from *peruvians.txt* we use the pairs command in R to produce a scatter plot of every variable against every other. The result can be seen in Fig:**??**. From this figure, the variables age and weight seem to have a linear relationship, which would suggest that they are correlated.

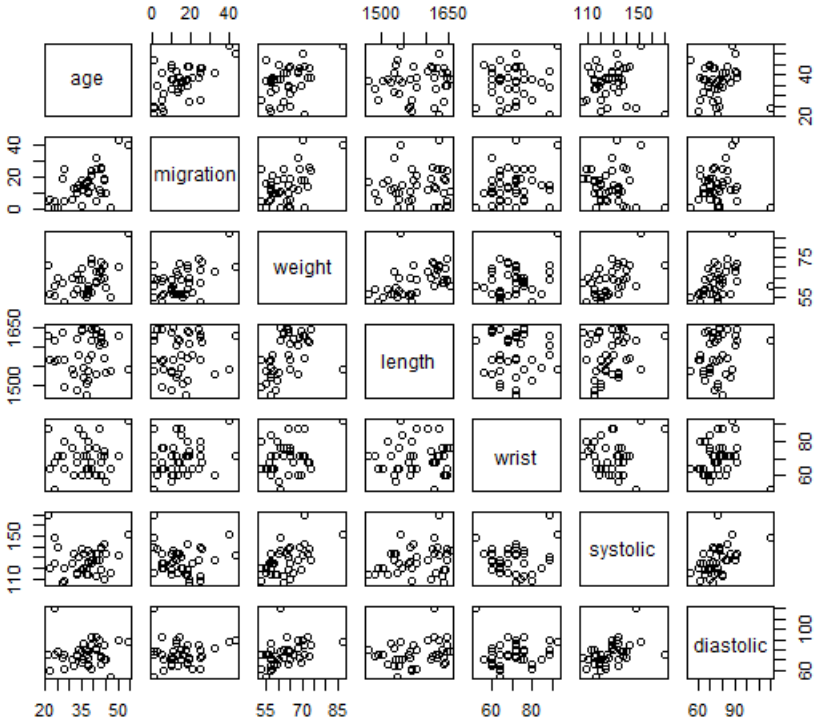

Figure 1: Pairwise scatterplots of the dataset peruvians

2

Using the Spearman rank correlation test for all variables against migration we obtain the following results:

Table 1: Spearman Rank Correlation test between each variable and *migration* data

	age	weight	length	wrist	systolic	diastolic
migration	0.47605753	0.35069559	0.08458432	0.21934983	-0.16842856	0.07514098

The rank correlation between:

- age and migration is 0.47605753, this is visible in Fig**??** in the form of an approximate linear relationship.

- weight and migration is 0.35069559, this is visible in Fig**??** as an approximate linear relationship.
- Length and migration is 0.08458432, as can be seen in Fig **??**, there is no discernible relationships between the two variables.
- wrist and migration is 0.21934983, this result is surprisingly high considering Fig**??**, as any relationship between these two variables is very difficult to see.
- systolic and migration is -0.16842856, which is obvious given the very small downward sloping structure seen in Fig**??**.
- diastolic and migration is 0.07514098, which is unsurprising considering the lack of structure seen in Fig**??**

2 Exercise 2

1

From the data in *clouds.txt*, we can perform various tests to determine whether the two independent samples: values of seeded clouds, and un-seeded clouds are equal.

- The two sample t-test produces: t = 1.9984, df = 33.856, 95% confidence interval of [-4.7405,559.5859] with mean of x = 441.9846 mean of y = 164.5619 and a p-value of 0.05375, which suggests that we accept the null hypothesis, and thus our samples are the same, this is misleading however. Consider Fig**??**, and Fig **??**; The two sample t-test assumes that both our samples come from a normal population, Fig **??** represents a randomly generated normally distributed sample of equal size to both seeded and un-seeded cloud data, clearly Fig**??** is very different, thus this assumption that our samples are from a normal population is not reasonable, and the two sample t-test is not suitable for this data as is.
- The Mann-Whitney test gives: W=473 with a p-value of 0.01383, suggesting that we reject the null hypothesis, and thus, our samples are not equal. The Mann-Whitney test makes no assumptions about the nature of the data but combine our samples to measure their differences, so it is suitable in this case.
- The Kolmogorov-Smirnov test gives: D= 0.4231 with a p-value of 0.01905, suggesting that we reject the null hypothesis and that our samples are not equal. The Kolmogorov-Smirnov test measures the differences in the histograms between the two sets of data, and with no constraints on the requirements for our data, this test is suitable.

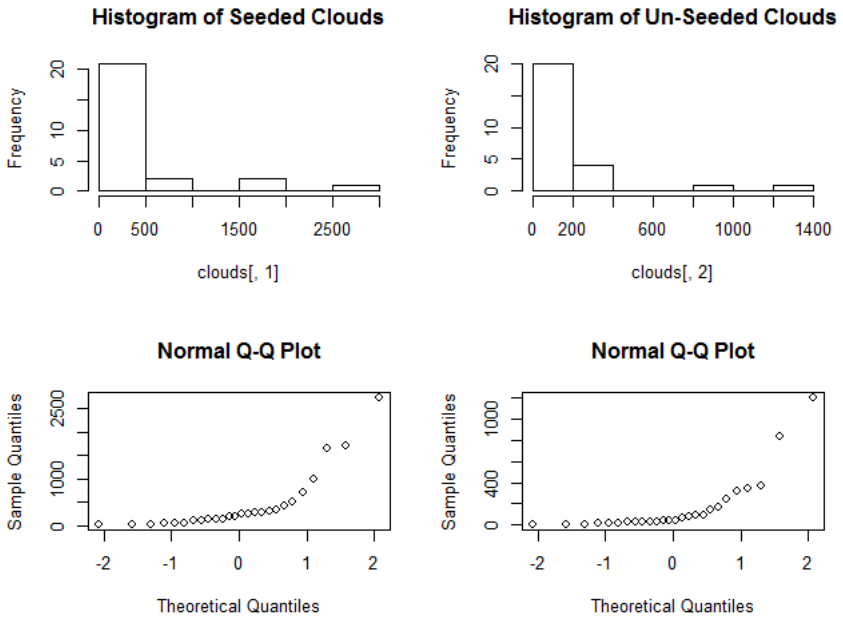

Figure 2: Histograms and QQ-Plots of Seeded Clouds and Un-Seeded Clouds respectively

2

We repeat the previous procedure after taking the square root of each data point from both seeded and un-seeded cloud samples.

- The two sample t-test produces: t=2.4246, df = 43.363, and a p-value of 0.01956, which suggests that we reject the null hypothesis and that our two samples are not equal. In this case the 95% confidence interval is [1.2021,13.0713] with mean x=17.068 and mean y= 9.9313. The transformed data, seen in Fig**??** is still too far from normal for the t-test to be suitable in this case.

- The Mann-Whitney test gives: W=473 with a p-value of 0.01383, suggesting that we reject the null hypothesis, and thus, our samples are not equal. This is the same result as the original data, which is unsurprising considering the test measures the ranks of a combined sample, once again this test is suitable.

- The Kolmogorov-Smirnov test gives: D= 0.4231 with a p-value of 0.01905, suggesting that we reject the null hypothesis and that our samples are not equal. Again, this is the same result obtained with the original data, and

as this test only measures the difference between histograms, it is again suitable.

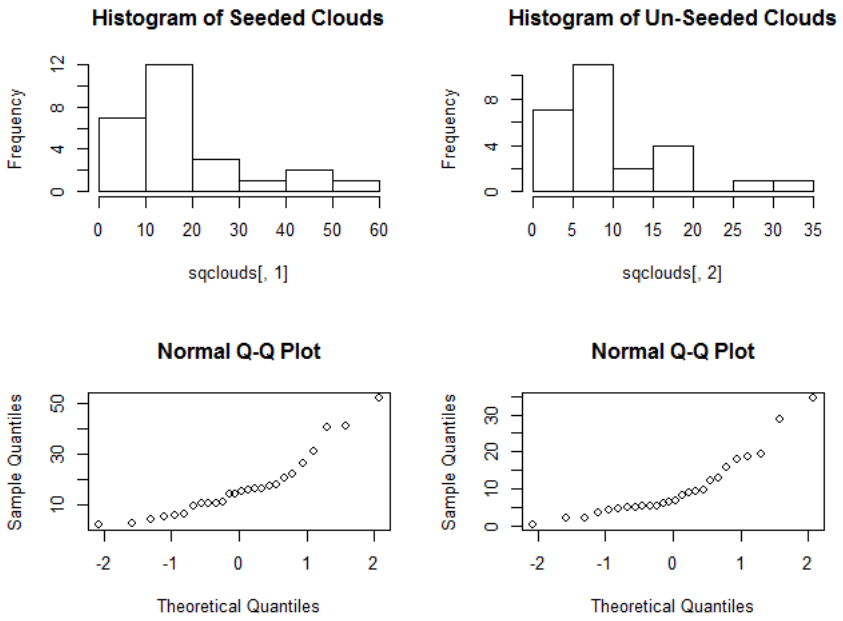

Figure 3: Histograms and QQ-Plots of Seeded Clouds and Un-Seeded Clouds after taking the square root of each data point

3

We repeat the same procedure again, now obtaining the square root of the square root of the original data point in our samples seeded and un-seeded clouds.

- The two sample t-test produces: t=2.5968, df = 48.826, p-value=0.0124, 95% confidence interval [0.2196,1.7236], mean x = 3.8789 and mean y = 2.9073. This suggests that we reject the null hypothesis, and that our samples are not equal. If we observe Fig**??**, we have no reason to believe our data is not normal, thus this test is suitable in this case.
- The Mann-Whitney test gives: W=473 with a p-value of 0.01383, suggesting that we reject the null hypothesis, and thus, our samples are not equal. This is the same result as obtained in both cases before, once again this test is suitable.
- The Kolmogorov-Smirnov test gives: D= 0.4231 with a p-value of 0.01905, suggesting that we reject the null hypothesis and that our samples are not

equal. Again, this is the same result obtained in the previous two cases, and it is again suitable.

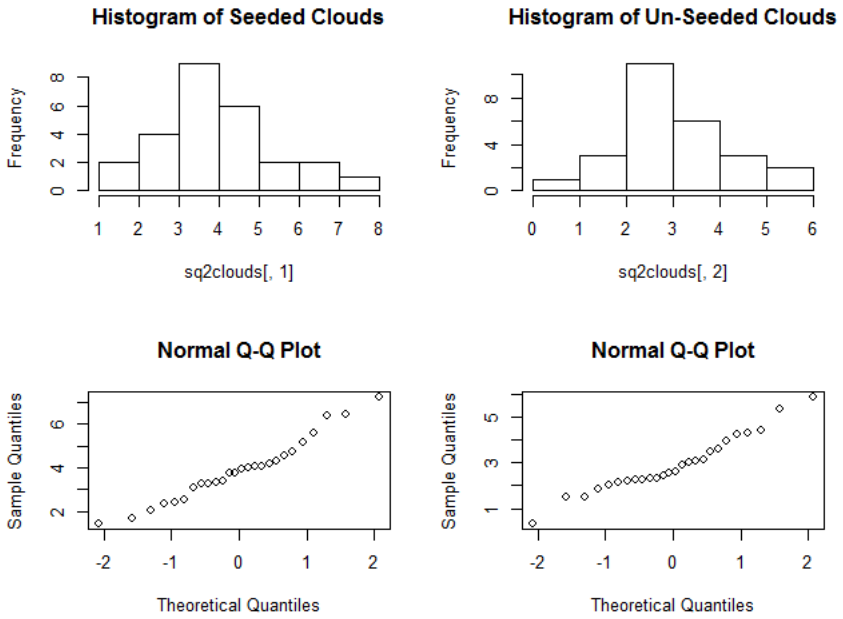

Figure 4: Histograms and QQ-Plots of Seeded Clouds and Un-Seeded Clouds after taking the square root of each data point twice

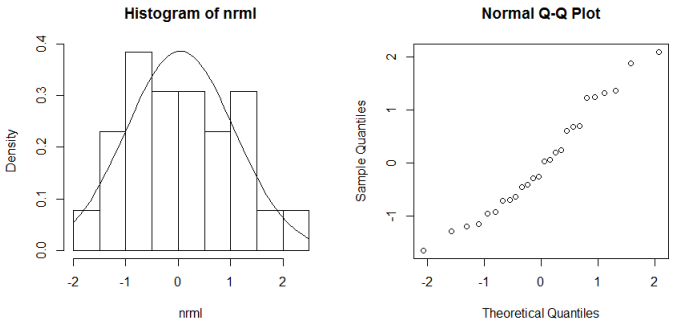

Figure 5: Histogram and QQ-Plot of randomly generated normally distributed data of same size as both Seeded and Un-Seeded Clouds

3 Exercise 3

1

Using data in *genal.txt* which is a collection of minimum values output by a genetic algorithm, based on different mutation factors, we produce a box plot of these minimum values for each factor. As we can see in Fig **??**, the variances appear to be similar across all factors.

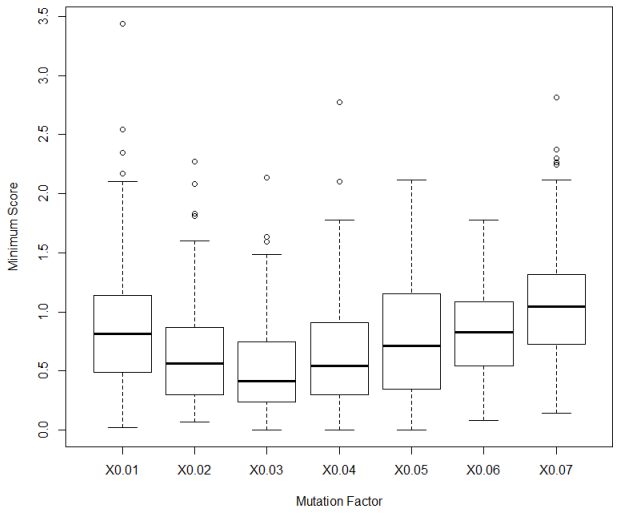

Figure 6: Box Plot of mutation factor and the resulting minimum result obtained

2

Observing the QQ-Plots of each mutation factor in Fig**??**, and comparing these to those of a randomly normally distributed sample of the same size (Fig**??**), it is obvious they differ significantly, and thus it is not a fair assumption that our data from a normal population.

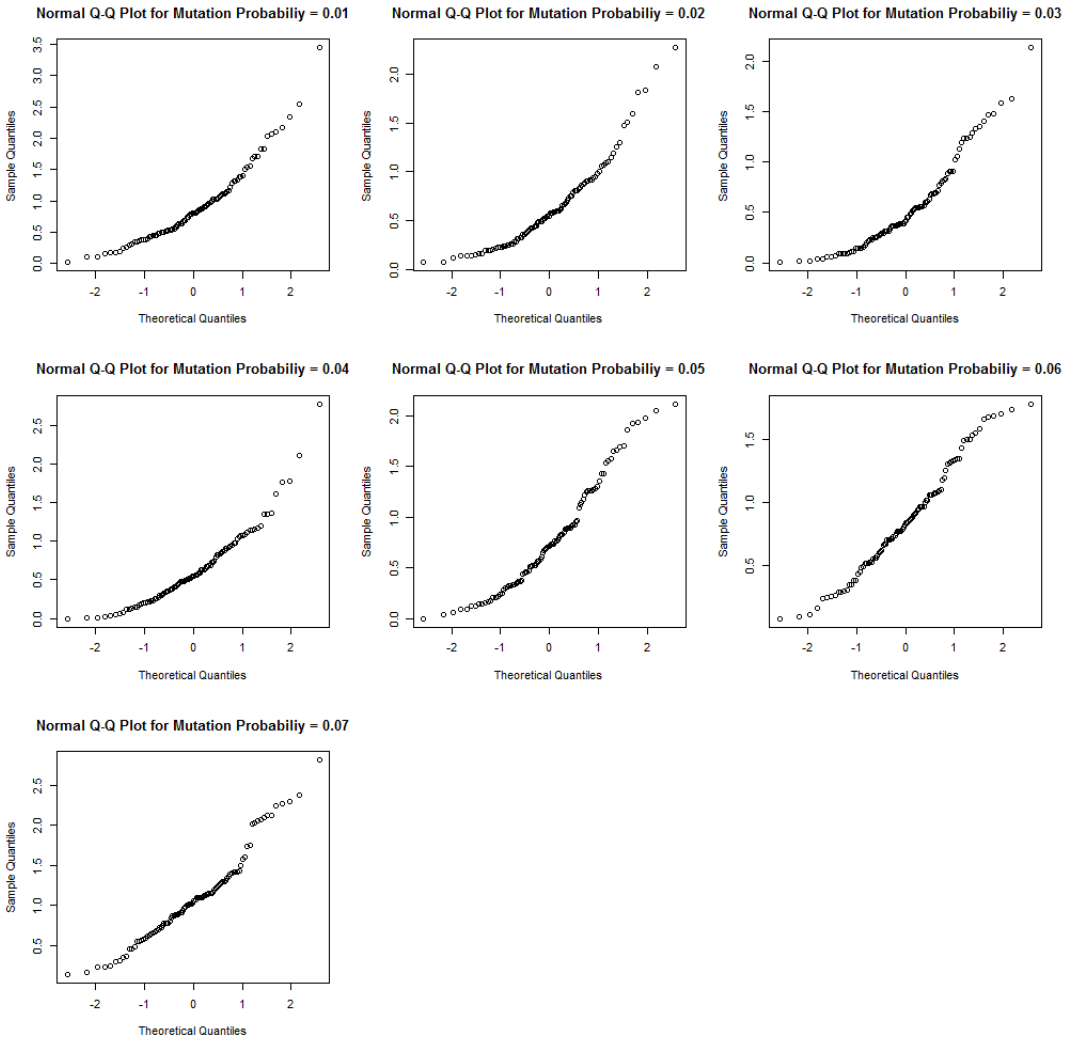

Figure 7: QQ-Plots for each mutation factor

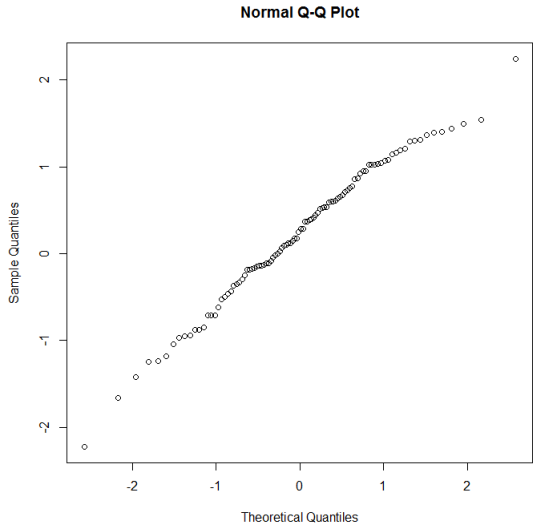

Figure 8: QQ-Plot of randomly generated normally distributed population of the same size as those in *genal.txt*

3

We transform the original data by taking the square root of each data point in our sample, and produce the QQ-Plots as seen in Fig**??**. Comparing these with those of the random normal QQ-Plot Fig **??**, it is clear that they do not differ noticeably, so in this case it is reasonable to assume these samples are from a normal population.

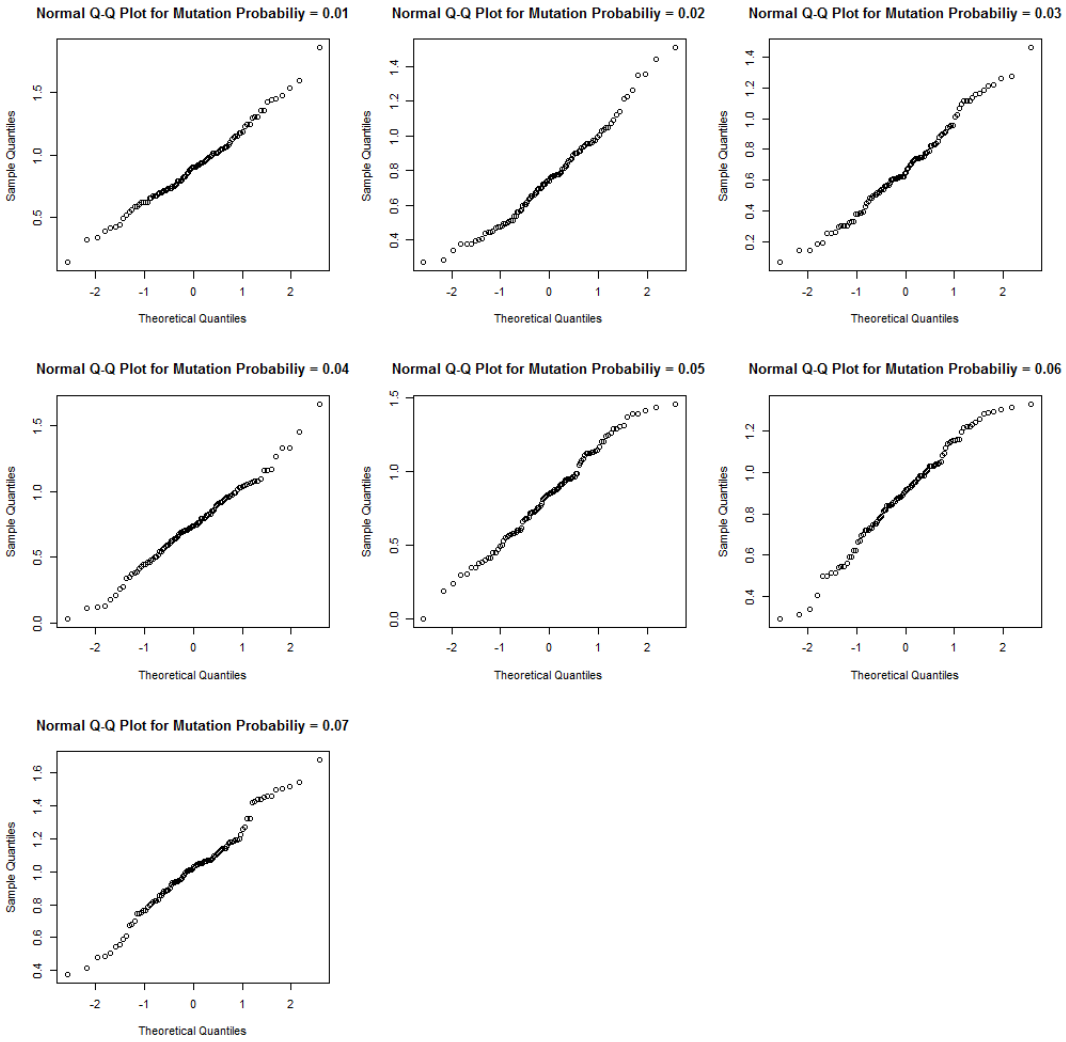

Figure 9: QQ-Plot for each mutation factor after take the square root of each data point

4

Performing the 1-way Anova test on the new square root data, we obtain the following results:

Table 2: Results of 1-way Anova on square root of *genal.txt* data

	Df	Sum	Sq Mean	Sq F Value	Pr(¿F)
Variety	6	7.824	1.30408	15.987	2.2e-16
Residuals	693	56.529	0.08157		

With a p-value of 2.2e-16, this suggests that we reject the nully hypothesis, that is to say that the mutation probability does in fact have an influence on the genetic algorithm.

5

If we use the summary() command on the data we have obtained with the 1-way Anova test, we obtain the estimates:

Table 3: Results of 1-way Anova on square root of *genal.txt* data

mean(X=0.01) - mean(X=)		0.02	0.03	0.04	0.05	0.06	0.0
Estimate	0.903272	-0.145905	-0.222653	-0.161817	-0.074234	-0.007617	0.108

Based on summary() X=0.03 is the best mutation probability, because 0.9033-0.2227 is the smallest value, and we are estimating the smallest value obtained from an optimization where smaller is better.

6

By making a QQ-Plot of the residuals of our 1-way Anova analysis we obtaing Fig**??**. It appears to be normal, which suggests that we made a correct assumption that our data is from a normal population once transformed by taking the square root.

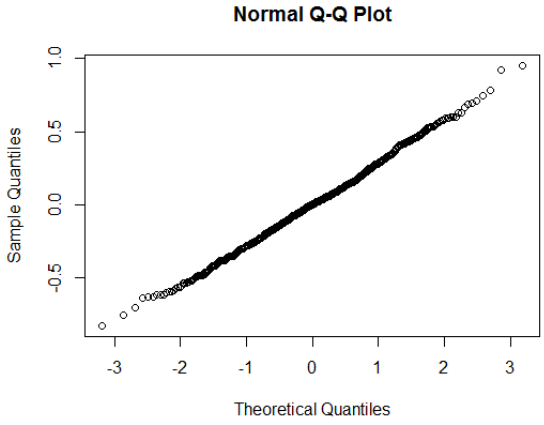

Figure 10: QQ-Plot of the residuals of 1-way Anova test

4 Exercise 4

1

From the data in *dogs.txt*, we make a box plot that measures the plasma epinephrine concentrations of dogs under various anaesthesias, Fig **??**.

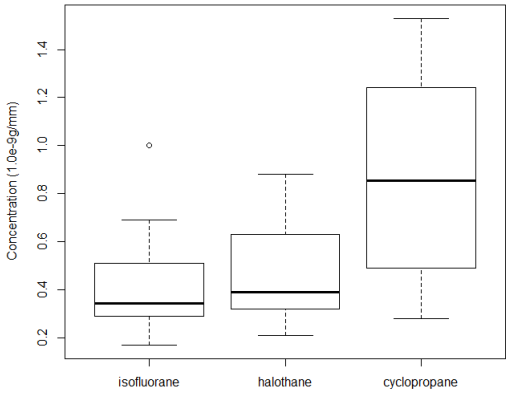

Figure 11: Box plot of plasma epinephrine concentrations under various anaesthesias

2

By making QQ-Plots of our samples for each type of anaesthesia (Fig**??**) we can see that the data appears normally distributed, we do not believe however that this is a reasonable assumption to make as the sample sizes are all so small they can easily be misinterpreted.

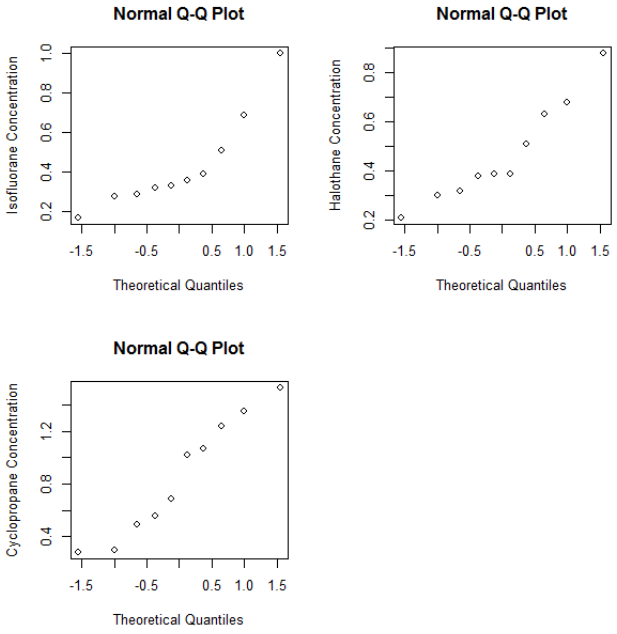

Figure 12: QQ-Plots for data from each type of anaesthesia

3

Using 1-way Anova to test the influence of the different anaesthesias we obtain:

Table 4: Results of 1-way Anova on *dogs.txt* data

	Df	Sum	Sq Mean	Sq F Value	Pr(¿F)
Variety	2	1.0808	0.54040	5.355	0.011
Residuals	27	2.7247	0.10092		

With a p-value=0.011 we should reject the null hypothesis, and thus the different drugs do affect the plasma epinephrine.

Using the summary() command in R on our Anova data again we obtain:

Table 5: Estimated values for plasma epinephrine concentrations for different drugs

mean(isoflourane) - mean( )		halothane	cyclopropane
Estimate	0.4340	0.0350	0.4190

Reject the null, p-value 0.011, estimated value of Isofluorane: 0.4340, of Halothane: 0.469 and of Cyclopropane: 0.853 based on table **??**.

4

The Kruskal-Wallis test for the same hypothesis gives Kruskal-Wallis chi-squared = 5.6442, df=2, p-value=0.05948, which suggests that we do not reject the null hypothesis, and that the different drugs do not have an influence on the plasma epinephrine concentrations.

5

The differences between the 1-way Anova and the Kruskal-Wallis test could be explained by our initial assumption that the 3 samples represented in the data do not in fact come from a normal population, because 1-way Anova assumes that we are sampling from normal populations this could produce conflicting results, though as we see in Figure **??** the residuals do not deviate from the normal significantly. It is perhaps more compelling that the differences between the 1-way Anova and the Kruskal-Wallis are due to the fact that 1-way Anova assumes our populations have equal population variances, and as is observed in Figure **??**, they are obviously not equal. If we calculate these population variances explicitly they are:

- For isofluorane, population variance 0.05967111
- for halothane, population variance 0.04214333
- for cyclopropane, population variance 0.2009344

So clearly the population variance for cyclopropane is too large in comparison to the other two drugs to use 1-way Anova effectively.

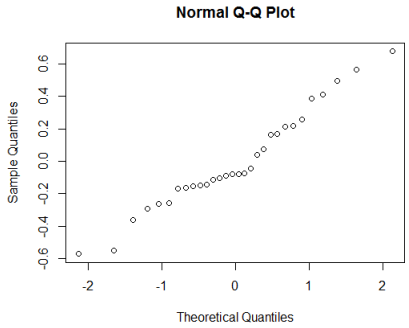

Figure 13: QQ-Plot of residuals of 1-way Anova analysis

5 R-Code

5.1 Exercise 1

```
      #1.1
peru = read.table("peruvians.txt", header=T)
peru = peru[,-c(5,6,7)]
pairs(peru)

#1.2
#This shows the spearman rank correlation against all variables
correl_s = cor(peru, method="spearman")

correl_s[2, ]
```

5.2 Exercise 2

```
      #2.1
clouds = read.table("clouds.txt", header=T)
len = length(clouds[,1])
nrml = rnorm(len)

#Histograms, boxplots, qqplots to see if data is normal
par(mfrow=c(2,2))
hist(clouds[,1])
hist(clouds[,2])
qqnorm(clouds[,1])
qqnorm(clouds[,2])
#These graphs should show the data is not normal

#Two sample t-test
```

```
t.test(clouds[ ,1],clouds[ ,2])
#Mann-Whitney test
wilcox.test(clouds[ ,1],clouds[ ,2])
#Kolmogorov-Smirnov test
ks.test(clouds[,1],clouds[,2])

#2.2
sqclouds = sqrt(clouds)

#Histograms, boxplots, qqplots to see if data is normal
par(mfrow=c(2,2))
hist(sqclouds[,1])
hist(sqclouds[,2])
qqnorm(sqclouds[,1])
qqnorm(sqclouds[,2])
#These graphs should show the data is not normal

#Two sample t-test
t.test(sqclouds[ ,1],sqclouds[ ,2])
#Mann-Whitney test
wilcox.test(sqclouds[ ,1],sqclouds[ ,2])
#Kolmogorov-Smirnov test
ks.test(sqclouds[,1],sqclouds[,2])

#2.3
sq2clouds = sqrt(sqclouds)

#Histograms, boxplots, qqplots to see if data is normal
par(mfrow=c(2,2))
hist(sq2clouds[,1])
hist(sq2clouds[,2])
qqnorm(sq2clouds[,1])
qqnorm(sq2clouds[,2])
#These graphs should show the data is not normal

#Two sample t-test
t.test(sq2clouds[ ,1],sq2clouds[ ,2])
#Mann-Whitney test
wilcox.test(sq2clouds[ ,1],sq2clouds[ ,2])
#Kolmogorov-Smirnov test
ks.test(sq2clouds[,1],sq2clouds[,2])

par(mfrow=c(1,2))
hist(nrml, prob=TRUE)
curve(dnorm(x,mean=mean(nrml),sd=sd(nrml)), add=TRUE)
qqnorm(nrml)
```

5.3 Exercise 3

```
#3.1
genal = read.table("genal.txt", header=T)
len=length(genal[,1])
nrml=rnorm(len)

par(mfrow=c(1,1));boxplot(genal)

#3.2
#Loops through, creating QQ-plot for each mutation probability
par(mfrow=c(3,3))
for (i in 1:7) {
  qqnorm(genal[,i],
         main =paste("Normal Q-Q Plot for Mutation Probabiliy =",
                     i/100))
}
#qqplot of normal random sample of same size as genal[,i]
par(mfrow=c(1,1));qqnorm(nrml)
#3.3
sqgenal=sqrt(genal)

#Loops through, creating QQ-plot for each mutation probability
par(mfrow=c(3,3))
for (i in 1:7) {
  qqnorm(sqgenal[,i],
         main =paste("Normal Q-Q Plot for Mutation Probabiliy =",
                     i/10))
}

#3.4

#Create data-frames for the origional genal data,
#and the squareroot genal data
sqframe=data.frame(yield=as.vector(as.matrix(sqgenal)),
                           variety=factor(rep(1:7,each=100)))

sqaov=lm(yield~variety,data=sqframe)
anova(sqaov)

#3.5
summary(sqaov)

#3.6
par(mfrow=c(1,1));qqnorm(residuals(sqaov))
```

5.4 Exercise 4

```
      #4.1

dogs = read.table("dogs.txt", header=T)
len=length(dogs[,1])
boxplot(dogs)

#4.2
par(mfrow=c(2,2))
qqnorm(dogs[,1],ylab="Isofluorane Concentration")
qqnorm(dogs[,2],ylab="Halothane Concentration")
qqnorm(dogs[,3],ylab="Cyclopropane Concentration")

#4.3
dogframe = data.frame(yield=as.vector(as.matrix(dogs)),
                      variety=factor(rep(1:3,each=10)))
dogaov=lm(yield~variety,data=dogframe)
anova(dogaov)
summary(dogaov)

#Calculate expected value
u1=0.4340;
u2=0.0350+u1
u3=0.4190+u1

print(u1,u2,u3)
#4.4

attach(dogframe)
kruskal.test(yield,variety)
par(mfrow=c(1,1));qqnorm(dogaov$residuals)

#Calculate and print population variances
for (i in 1:3){
  print(sum((dogs[,i]-mean(dogs[,i]))^2)/(len-1))
}
```