7/5

Paul
Peyssard
MSc DSAI

Inference Theory exam

Exercice 1

Let consider the simple linear model defined by the matricial writting:

$$\underline{Y} = \underline{X}\beta + U \quad / \quad \beta = \begin{pmatrix} \beta_0 \\ \beta_1 \end{pmatrix},$$

We see that:

$$\hat{\beta_1} = \frac{\Sigma(X_i - \bar{X}_n)(Y_i - \bar{Y}_n)}{\Sigma(X_i - \bar{X}_n)^2}$$

For cleanliness, $\Sigma = \sum_{i=1}^{n}$

1) Prove $\hat{B_1} = B_1 + \frac{\Sigma(X_i - \bar{X}_n)\varepsilon_i}{\Sigma(X_i - \bar{X}_n)^2}$

We can define $\hat{B_1} = \frac{\Sigma(X_i - \bar{X}_n)(\beta_0 + \beta_1 X_i + \varepsilon_i - \beta_0 - \beta_1 \bar{X}_n - \bar{\varepsilon}_i)}{\Sigma(X_i - \bar{X}_n)^2}$

$$= \frac{\Sigma B_1 (X_i - X_n)^2}{\Sigma(X_i - \bar{X}_n)^2} + \frac{\Sigma(X_1 - X_n)\varepsilon_1}{\Sigma(X_1 - \bar{X}_n)^2} \quad \text{with} \quad E(\varepsilon_i) = 0$$

$$\Rightarrow B_1 + \frac{\Sigma(X_i - \bar{X}_n)\varepsilon_i}{\Sigma(X_i - \bar{X}_n)^2}$$

2) Deduce that $\hat{B}_1$ is an unbiased estimator of B_1

We have from previous question : $\hat{B}_1 = B_1 + \frac{\sum (x_i - \bar{x}_n)\varepsilon_i}{\sum_{i=1}^{n} (x_i - \bar{x}_n)^2}$

$E[B_1 + \sum \phi_i \varepsilon_i]$ / $\phi_i = \frac{x_i - \bar{x}_n}{\sum (x_i - \bar{x}_n)^2}$

$= B_1 + \sum \phi_i E[\varepsilon_i] = B_1$ using linearity properties.

$\Rightarrow B_1 - B_1 = 0 = \text{bias}$

$\Rightarrow \hat{B}_1$ is unbiased

3) Compute ~~[illegible]~~ variance of $\hat{B}_1$

$$V(\hat{B}_1) = V\left(B_1 + \frac{\sum (x_i - \bar{x}_n)\varepsilon_i}{\sum (x_i - \bar{x}_n)^2}\right)$$

$$= V\left(\frac{\sum (x_i - \bar{x})\varepsilon_i}{\sum (x_i - \bar{x})^2}\right)$$

$$= \frac{1}{\left(\sum (x_i - \bar{x})^2\right)^2} \times \left(V\left(\sum (x_i - \bar{x})\varepsilon_i\right)\right)$$

$$= \frac{1}{\left(\sum (x_i - \bar{x})^2\right)^2} \times \left(\sum (x_i - \bar{x})\right)^2 \times V(\varepsilon_i)$$

$$= \frac{1}{\sum(x_i-\bar{x})^2} \times V(\varepsilon_i) = \frac{\sigma^2}{\sum(x_i-\bar{x})^2}$$

4) Under the assumption:
- $\varepsilon_i \sim \mathcal{N}(0,\sigma^2)$ for $i \in \{1,\dots,n\}$
- independence of the ε_i,

Prove that

$$\hat{\beta} = \begin{pmatrix}\hat{\beta}_0\\ \hat{\beta}_1\end{pmatrix} \sim \mathcal{N}(\beta, \sigma^2 V)$$

with $V = \frac{1}{\sum(x_i-\bar{x}_n)^2} \times \begin{pmatrix} \frac{\sum x_i^2}{n} & -\bar{x}_n \\ -\bar{x}_n & 1 \end{pmatrix}$

Plus, we have: $\hat{\beta}_0 = \bar{y}_n - \hat{\beta}_1 \bar{x}_n$

$$\hat{\beta}_1 = \beta_1 + \frac{\sum(x_i-\bar{x}_n)\varepsilon_i}{\sum(x_i-\bar{x}_n)^2}$$

which follow normal law

$$\Rightarrow V[\hat{\beta}_1] = \frac{\sigma^2}{\sum(x_i-\bar{x}_n)^2}, \quad V[\hat{\beta}_0] = \frac{\sigma^2 \sum x_i^2}{n\sum(x_i-\bar{x}_n)^2}$$

$$\Rightarrow \operatorname{cov}(\hat{\beta}_0, \hat{\beta}_1) = \frac{-\sigma^2 \bar{x}_n}{\sum(x_i-\bar{x}_n)^2}$$

$$\Rightarrow V = \begin{bmatrix} V(\hat{\beta}_0) & \operatorname{cov}(\hat{\beta}_0,\hat{\beta}_1) \\ \operatorname{cov}(\hat{\beta}_0,\hat{\beta}_1) & V(\hat{\beta}_1) \end{bmatrix}$$

4/5

$$= \begin{bmatrix} \frac{\sigma^2 \sum x_i^2}{n \sum (x_i - \bar{x}_n)^2} & \frac{-\sigma^2 \bar{x}_n}{\sum (x_i - \bar{x}_n)^2} \\ \frac{-\sigma^2 \bar{x}_n}{\sum (x_i - \bar{x}_n)^2} & \frac{\sigma^2}{\sum (x_i - \bar{x}_n)^2} \end{bmatrix}$$

$$= \frac{1}{\sum (x_i - \bar{x}_n)^2} \begin{bmatrix} \frac{\sum x_i^2}{n} & -\bar{x}_n \\ -\bar{x}_n & 1 \end{bmatrix}$$

5) Prove the expression of the CI for y_{n+1} which is

$$\left[\hat{y}_{n+1} \pm \hat{\sigma}_n t_{n-2,\, 1-\alpha/2} \sqrt{1 + \frac{1}{n} + \frac{(x_{n+1} - \bar{x}_n)^2}{\sum (x_i - \bar{x}_n)^2}} \right]$$

where $\hat{y}_{n+1} = \hat{\beta}_0 + \hat{\beta}_1 x_{n+1}$, $\hat{\sigma}_n^2 = \frac{1}{n-2} \sum_{i=1}^{n} (y_i - \hat{y}_i)^2$
and $P(T(n-2) \leq t_{n-2,\, 1-\alpha/2}) = 1 - \alpha/2$

$$E[y_{n+1} - \hat{y}_{n+1}] = E[\beta_0 - \hat{\beta}_0 + (\beta_1 - \hat{\beta}_1) x_{n+1} + \varepsilon_{n+1}]$$

$$= 0 \quad \text{from previous question, bias} = 0 \ldots$$

then

$$V(y_{n+1} - \hat{y}_{n+1}) = \sigma^2 \left(1 + \frac{1}{n} + \frac{(x_{n+1} - \bar{x}_n)^2}{\sum (x_i - \bar{x}_n)^2} \right)$$

As ε_i, σ^2 are independent and $\frac{\hat{\sigma}_n^2 (n-1)}{\sigma^2} \sim \chi^2_{n-2}$

$$\Rightarrow \frac{y_{n+1} - \hat{y}_{n+1}}{\hat{\sigma}_n \sqrt{1 + \frac{1}{n} + \frac{(x_{n+1} - x_n)^2}{\sum (x_i - \bar{x}_n)^2}}} \sim T(n-2)$$

5/5

$$\Rightarrow P\left(-t_{n-2;\,1-\alpha/2} \leq T \leq t_{n-2,\,1-\alpha/2}\right) = 1-\alpha$$

$$\Rightarrow -t_{n-2,\,1-\alpha/2} \leq t_i \leq t_{n-2,\,1-\alpha/2}$$

$$\Rightarrow -t_{n-2,\,1-\alpha/2} \leq \frac{y_{n+1} - \hat{y}_{n+1}}{\sqrt{V(\hat{y}_{n+1})}} \leq t_{n-2,\,1-1/\alpha}$$

$$\Rightarrow \hat{y}_{n+1} \pm t_{n-2,\,1-\alpha/2}\sqrt{V(\hat{y}_{n+1})}$$

$$\Rightarrow -t_{n-2,\,1-\alpha/2}\sqrt{V(\hat{y}_{n+1})} \leq y_{n+1} - \hat{y}_{n+1} \leq t_{n-2,\,1-\alpha/2} + \sqrt{V(\hat{y}_{n+1})}$$

$$\Rightarrow \hat{y}_{n+1} - t_{n-2,\,1-\alpha/2}\sqrt{V(\hat{y}_{n+1})} \leq y \leq \hat{y}_{n+1} + t_{n-2,\,1-\alpha/2}\sqrt{V(\hat{y}_{n+1})}$$

exam_inference_theory_exo2_and_3

Paul Peyssard

2023-02-14

Libraries

```
library(ggplot2)
library(dplyr)
```

```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
library(stats)
```

Exercise 2 :

Consider the mtcars database in the R software. The mpg variable is the response variable, all the others are explanatory variables.

Question 1) Write your code to compute B_hat, sigma_square_hat

and to decide if a linear model is a good model

```
#import mtcars
data("mtcars")

#x=all the variable but mpg, y=mpg
x <- as.matrix(mtcars[, -1])
y <- as.matrix(mtcars$mpg)


#we combine x with a column of 1s to represent the intercept in a linear regression model.
x <- cbind(rep(1, nrow(x)), x)

#Compute B_hat with linear regression,  t() returns the transpose matrix
B_hat <- solve(t(x) %*% x) %*% t(x) %*% y

print(B_hat)
```

```
##             [,1]
##      12.30337416
## cyl  -0.11144048
## disp  0.01333524
## hp   -0.02148212
## drat  0.78711097
## wt   -3.71530393
## qsec  0.82104075
## vs    0.31776281
## am    2.52022689
## gear  0.65541302
## carb -0.19941925
```

```
#Compute sigma_square_hat
  #firstly, the residual of lr :

residuals<- y-x%*%B_hat

  #Secondly, sigma_square_hat
sigma_square_hat<-t(residuals)%*%residuals/(nrow(x)-ncol(x))
print(sigma_square_hat)
```

```
##          [,1]
## [1,] 7.023544
```

```
#Compute the explained variance of lr
exp_var <- sum((x %*% B_hat - mean(y))^2)

#Compute the total variance of lr
tot_var<-sum((y-mean(y))^2)

#Compute the coefficient of determination : r_squared
r_squared<-exp_var/tot_var

print(r_squared)
```

```
## [1] 0.8690158
```

Here, we have an r_square of 0.86 which is very close to 1. It means that the linear regression is quite good in this case and it captures a large portion of the response variable.

Question 2) Compare your results with the one produced by the lm function

```
#Compute the same with built-in functions
compare <- lm(mtcars$mpg ~ ., data = mtcars[,-1])
#Extract R_squared from summary
r_squared2 <- summary(compare)$r.squared

print(r_squared2)
```

```
## [1] 0.8690158
```

Here, we can see that my previous result r_squared is the same as the result of lm function r_squared2

if we want more detail about lm to compare deeply, we can print the whole summary

```
summary(compare)
```

```
##
## Call:
## lm(formula = mtcars$mpg ~ ., data = mtcars[, -1])
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -3.4506 -1.6044 -0.1196  1.2193  4.6271
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 12.30337   18.71788   0.657   0.5181
## cyl         -0.11144    1.04502  -0.107   0.9161
## disp         0.01334    0.01786   0.747   0.4635
## hp          -0.02148    0.02177  -0.987   0.3350
## drat         0.78711    1.63537   0.481   0.6353
## wt          -3.71530    1.89441  -1.961   0.0633 .
## qsec         0.82104    0.73084   1.123   0.2739
## vs           0.31776    2.10451   0.151   0.8814
## am           2.52023    2.05665   1.225   0.2340
## gear         0.65541    1.49326   0.439   0.6652
## carb        -0.19942    0.82875  -0.241   0.8122
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.65 on 21 degrees of freedom
## Multiple R-squared:  0.869,  Adjusted R-squared:  0.8066
## F-statistic: 13.93 on 10 and 21 DF,  p-value: 3.793e-07
```

Question 3) Perform by yourself, without a dedicated function, a

backward procedure to perform variable selection

```
#Set the initial pvalues threshold
pval_threshold <- 0.05

#Initialize an empty vector to store the p-values for each feature
pvals <- rep(NA, ncol(x) + 1)

#Fit the initial linear regression model using all features from mtcars
  #Here I recompute b_hat,residuals... to have a better view in the cell oh what is happening ev
en if we already did it previously
b_hat <- solve(t(x) %*% x) %*% t(x) %*% y
residuals <- y - x %*% b_hat
sse <- t(residuals) %*% residuals
sigma_square_hat <- as.integer(sse / (nrow(x) - ncol(x)))

#Compute the variance-covariance matrix :
vcov <- solve(t(x) %*% x) * sigma_square_hat
se <- sqrt(diag(vcov))
t_stat <- b_hat / se

#Compute p values. pt() compute the CDF of t-distribution
pvals <- 2 * pt(abs(t_stat), df = nrow(x) - ncol(x), lower.tail = FALSE)

#Iterate over all features until all p-values are below the threshold and that we cannot go furt
her
while (max(pvals[-1]) > pval_threshold) {
#Find the index of the feature with the highest p-value
  to_remove <- which.max(pvals[-1]) + 1

#Remove the non wanted feature from X
  x <- x[, -to_remove]

#Compute the new coefficients
  b_hat <- solve(t(x) %*% x) %*% t(x) %*% y

#Compute the residuals and sum of squared errors
  residuals <- y - x %*% b_hat
  sse <- t(residuals) %*% residuals

#Compute the variance-covariance matrix
  vcov <- solve(t(x) %*% x) * as.integer(sse / (nrow(x) - ncol(x)))

#Compute the standard errors and t-stat for each coef
  se <- sqrt(diag(vcov))
  t_stat <- b_hat / se
#Compute the two-sided p-value
  pvals <- 2 * pt(abs(t_stat), df = nrow(x) - ncol(x), lower.tail = FALSE)
}
```

```
#Final set of features :
cat("set of Selected features:", colnames(x)[-1], "\n")
```

```
## set of Selected features: wt qsec am
```

```
#FInal intercept
cat("Intercept : ",pvals[1],"\n")
```

```
## Intercept :  0.1763172
```

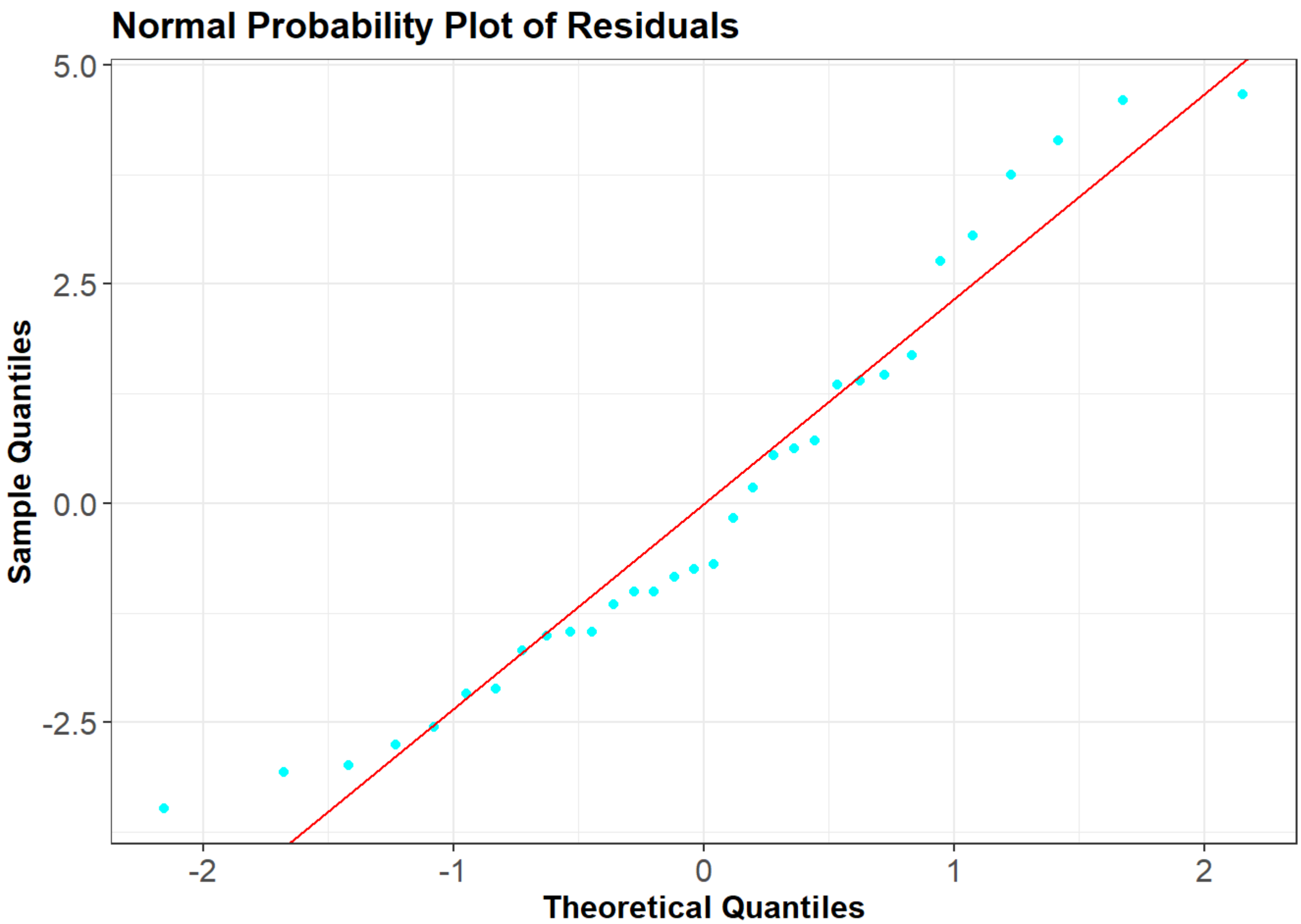
We can aslo print the intercept and value together.

```
print(pvals)
```

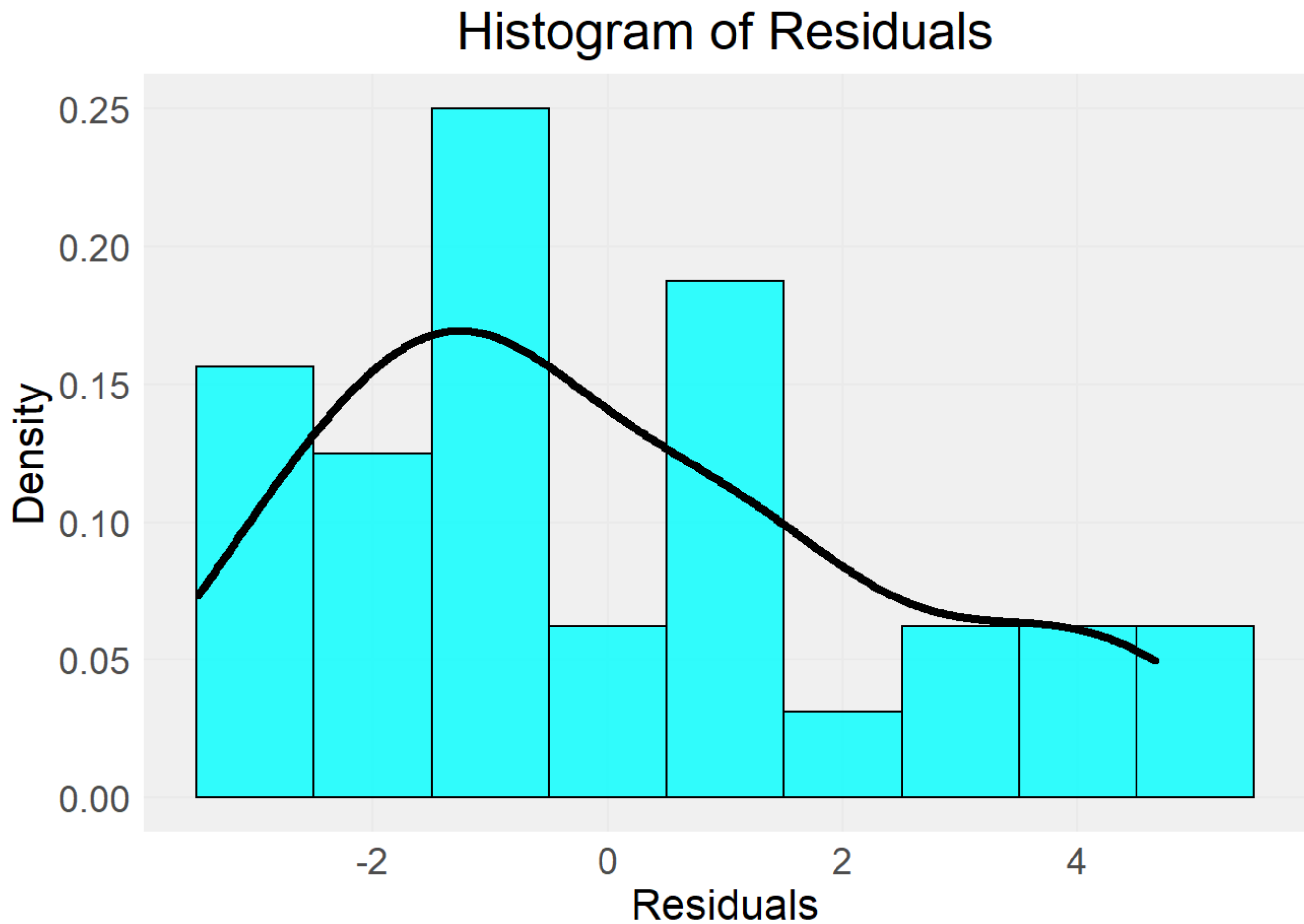
```
##              [,1]
##      1.763172e-01
## wt   6.565999e-06
## qsec 2.068865e-04
## am   4.593985e-02
```

Question 4) Verify if the Gaussian assumption onto the noise is verified. Why this is important to see with respect to question 3 ?

```
#To verify the gaussian assumption, we can plot the qqplot and the histogram of the residuals
data.frame(residuals) %>% ggplot(aes(sample = residuals)) +
  geom_qq(color = "cyan") +
  geom_abline(intercept = mean(residuals), slope = sd(residuals), color = "red") +
  ggtitle("Normal Probability Plot of Residuals") +
  xlab("Theoretical Quantiles") +
  ylab("Sample Quantiles") +
    theme_bw() +
  theme(plot.title = element_text(size = 14, face = "bold"),
        axis.text = element_text(size = 12),
        axis.title = element_text(size = 12, face = "bold"))
```




```
data.frame(residuals) %>% ggplot(aes(x = residuals)) +
  geom_histogram(aes(y = ..density..), binwidth = 1, color = "black", fill = "#00FFFF", alpha =
0.8) +
  geom_density(color = "black", size = 1.5) +
  theme_minimal() +
  labs(x = "Residuals", y = "Density", title = "Histogram of Residuals") +
  theme(plot.title = element_text(size = 20, hjust = 0.5),
        axis.title = element_text(size = 16),
        axis.text = element_text(size = 14),
        panel.grid.major = element_line(color = "#EAEAEA"),
        panel.grid.minor = element_blank(),
        panel.background = element_rect(fill = "#F0F0F0", color = NA),
        panel.border = element_blank())
```




```
shapiro.test(residuals)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  residuals
## W = 0.9411, p-value = 0.08043
```

Conclusion :

The qqplot and the histogram of the residuals show that the residuals follow a Gaussian distribution. This is important for the backward procedure in question 2.3, because it assumes that the residuals are normally distributed.

By visually confirming that the residuals follow a normal distribution, we can ensure that the assumptions of the backward procedure are satisfied. This means that the results of the variable selection procedure are more likely to be reliable and accurate, and that the selected variables are more likely to be associated with the response variable.

Plus, Based on the Shapiro-Wilk test and its p-values greater than 0.05, we can assume with reasonable confidence that the noise in the data follows a normal distribution.

In other words, the assumption of normality for the residuals is a crucial assumption for linear regression models, and it is important to check this assumption before proceeding with variable selection or any other inference procedure. By doing so, we can avoid biased or unreliable results, and ensure that the model is appropriately specified and valid for the data at hand.

Exercise 3 :

What kind of variable can you assign to the observations created in the object a ? Convince me with graphics and a test

```
#library(stats)

#n = number of samples generated
n = 10000

#lambda
l = 4.7

#Create n random number from uniform distribution (runif) and applies (-1/l)*log(i) for each ele
ment n  --> We can assume an Exponential distribution
a = (-1/l)*(log(runif(n)))

a[0:10]
```

```
##  [1] 0.31848963 0.12033373 0.04449831 0.53436059 0.01549886 0.27278314
##  [7] 0.48405410 0.75740430 0.15162310 0.07761952
```

a is a list of 10000 generated sample from an exponential distribution. I will prove this assumptions with different plots.

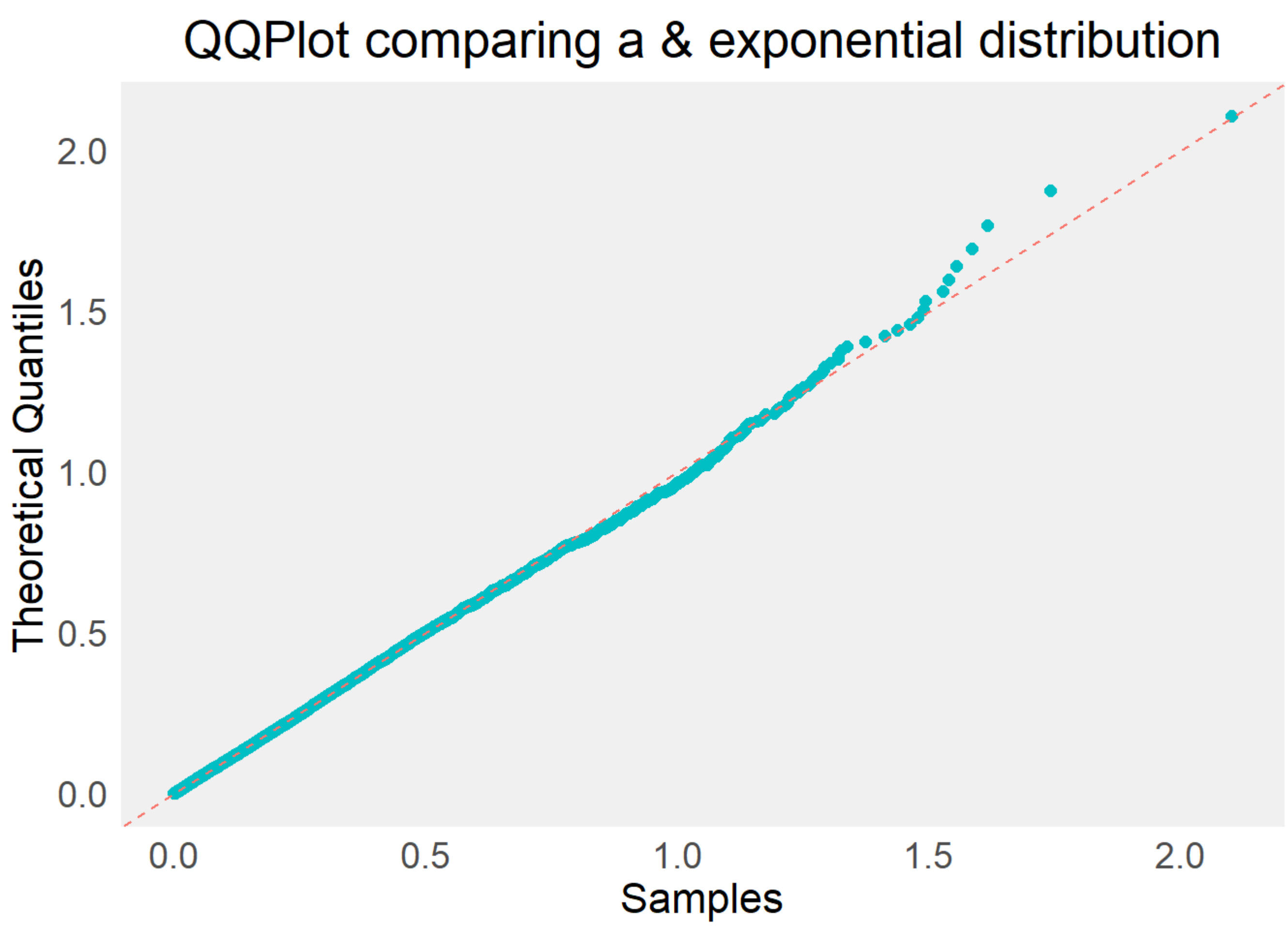
Firstly, qqplot :

```
#Generate the exponential distribution quantiles and sort them in order to compare
exp_distrib <- qexp(ppoints(n), 1)
exp_distrib_sorted <- sort(exp_distrib)

#Sort the generated data
a_sort <- sort(a)

#Convert exponential distribution into dataframe
exp_data <- data.frame(Sample = a_sort, exp_distrib = exp_distrib_sorted)

#qqplot
exp_data %>% ggplot(aes(x = Sample, y = exp_distrib)) +
  geom_point(size = 2, color = "#00BFC4") +
  geom_abline(slope = 1, intercept = 0, linetype = "dashed", color = "#F8766D") +
  theme_minimal() +
  labs(x = "Samples", y = "Theoretical Quantiles", title = "QQPlot comparing a & exponential dis
tribution") +
  theme(plot.title = element_text(size = 20, hjust = 0.5),
        axis.title = element_text(size = 16),
        axis.text = element_text(size = 14),
        panel.grid.major = element_blank(),
        panel.grid.minor = element_blank(),
        panel.background = element_rect(fill = "#F0F0F0", color = NA),
        panel.border = element_blank())
```

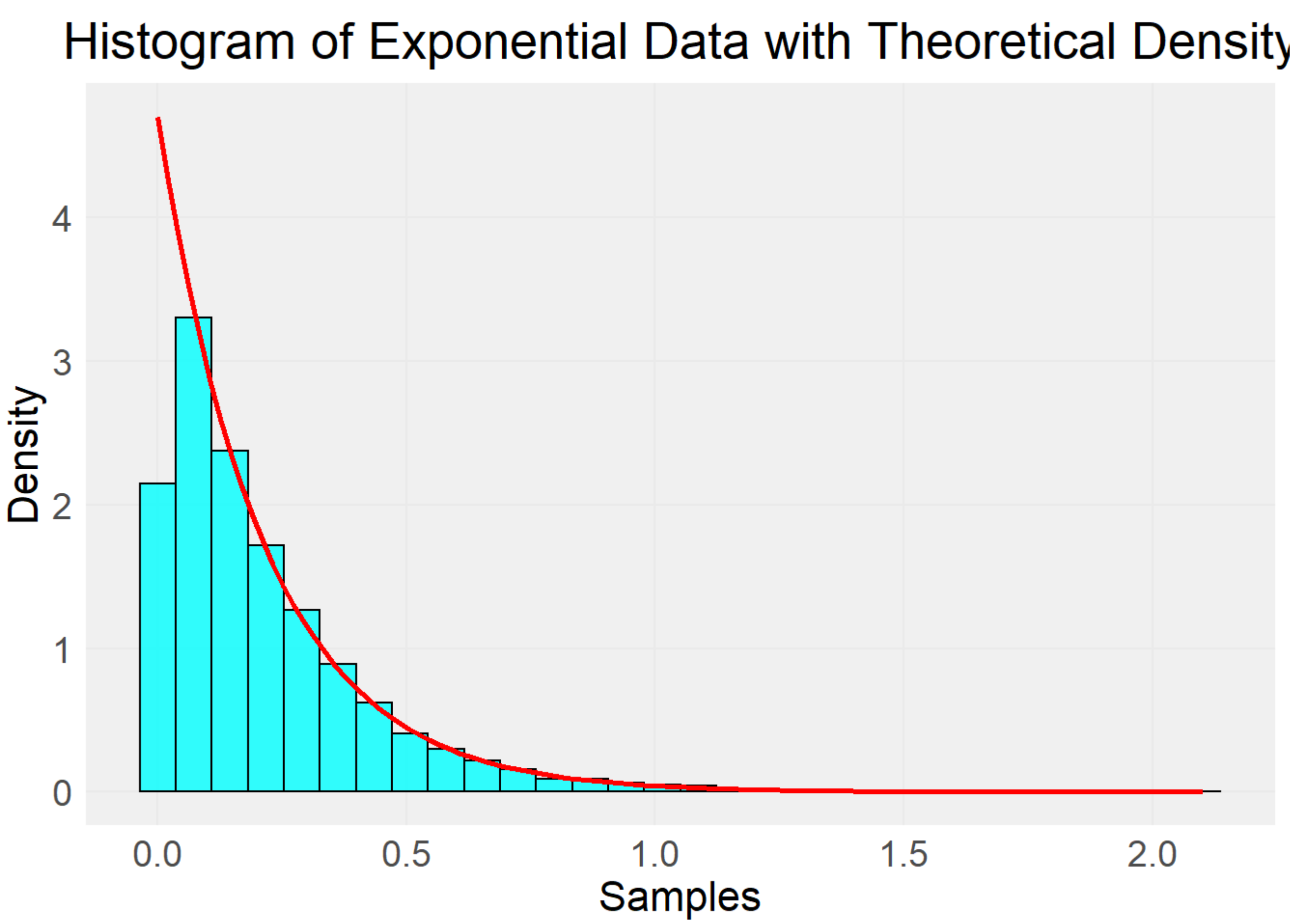


Secondly, histogram :

```
a_data <- data.frame(a)

a_data %>% ggplot(aes(x = a)) +
  geom_histogram(aes(y = ..density..), color = "BLACK", fill = "CYAN", alpha = 0.8) +
  stat_function(fun = dexp, args = list(rate = 1), color = "RED", size = 1) +
  theme_minimal() +
  labs(x = "Samples", y = "Density", title = "Histogram of Exponential Data with Theoretical Den
sity") +
  theme(plot.title = element_text(size = 20, hjust = 0.5),
        axis.title = element_text(size = 16),
        axis.text = element_text(size = 14),
        panel.grid.major = element_line(color = "#EAEAEA"),
        panel.grid.minor = element_blank(),
        panel.background = element_rect(fill = "#F0F0F0", color = NA),
        panel.border = element_blank())
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



With the qqplot and the histogram, we can see that the generated data and have an exponential distribution.

I will now proceed to a Kolmogorov-Smirnov test in order to have another proof of my assumptions.

```
ks.test(a, "pexp", 1)
```

```
##
##  Asymptotic one-sample Kolmogorov-Smirnov test
##
## data:  a
## D = 0.0095947, p-value = 0.316
## alternative hypothesis: two-sided
```

Conclusion :

To confirm that the observations created in the object a follow an exponential distribution with lambda = 4.7, we performed both visual and statistical tests. The visual tests involved creating a histogram and Q-Q plot of the data, while the statistical test involved using the Kolmogorov-Smirnov test.

The histogram of the data shows a peak around 0 and a long right tail, which is consistent with what we expect from an exponential distribution.

The Q-Q plot shows that the data points are approximately linear, which also indicates that the distribution of the data is close to the theoretical exponential distribution.

Finally, the Kolmogorov-Smirnov test provides a statistical measure of the correlation between the data and the theoretical distribution, and the test result (p-value =0.8131 > 0.05) suggests that we do not have enough evidence to reject the null hypothesis that the data follows an exponential distribution with lambda = 4.7.

Therefore, we can conclude that the observations created in the object a can be assigned to an exponential distribution with lambda = 4.7.