Choice-Based Conjoint study

Hamza

Table of Contents

[Task 1 1](#_Toc101278910)

[Task 2 5](#_Toc101278911)

[Task 3 6](#_Toc101278912)

[Task 4 7](#_Toc101278913)

[Task 5 9](#_Toc101278914)

[Task 6 10](#_Toc101278915)

[Task 7 11](#_Toc101278916)

[Task 8 13](#_Toc101278917)

[Task 9 14](#_Toc101278918)

[Task 10 14](#_Toc101278919)

[Task 11 14](#_Toc101278920)

[Task 12 15](#_Toc101278921)

[Task 13 16](#_Toc101278922)

[Task 14 16](#_Toc101278923)

[Task 15 16](#_Toc101278924)

[Task 16 17](#_Toc101278925)

[Task 17 17](#_Toc101278926)

[Task 18 18](#_Toc101278927)

[Task 19 18](#_Toc101278928)

[Task 20 19](#_Toc101278929)

# Task 1

Read and inspect the data set. Provide a descriptive analysis for each of the variables in the data set. Make sure you provide an analysis that is meaningful for each variable type (e.g., factors, identifiers).

* **Anwser**

# Load packages  
library("plotly")

## Loading required package: ggplot2

##   
## Attaching package: 'plotly'

## The following object is masked from 'package:ggplot2':  
##   
## last\_plot

## The following object is masked from 'package:stats':  
##   
## filter

## The following object is masked from 'package:graphics':  
##   
## layout

library("tidyverse")

## -- Attaching packages --------------------------------------- tidyverse 1.3.1 --

## v tibble 3.1.6 v dplyr 1.0.8  
## v tidyr 1.1.4 v stringr 1.4.0  
## v readr 2.1.1 v forcats 0.5.1  
## v purrr 0.3.4

## -- Conflicts ------------------------------------------ tidyverse\_conflicts() --  
## x dplyr::filter() masks plotly::filter(), stats::filter()  
## x dplyr::lag() masks stats::lag()

library("data.table")

##   
## Attaching package: 'data.table'

## The following objects are masked from 'package:dplyr':  
##   
## between, first, last

## The following object is masked from 'package:purrr':  
##   
## transpose

library("dplyr")  
library("gridExtra")

##   
## Attaching package: 'gridExtra'

## The following object is masked from 'package:dplyr':  
##   
## combine

library("knitr")  
library("scales")

##   
## Attaching package: 'scales'

## The following object is masked from 'package:purrr':  
##   
## discard

## The following object is masked from 'package:readr':  
##   
## col\_factor

# Load Cloud Data  
df.aal <- read\_csv("cloud.csv")

## Rows: 9000 Columns: 9

## -- Column specification --------------------------------------------------------  
## Delimiter: ","  
## chr (4): cloud\_storage, customer\_support, cloud\_services, price  
## dbl (5): respondent\_id, choiseset\_id, alternative\_id, choice\_id, choice  
##   
## i Use `spec()` to retrieve the full column specification for this data.  
## i Specify the column types or set `show\_col\_types = FALSE` to quiet this message.

# view first 6 rows of cloud.csv  
head(df.aal)

## # A tibble: 6 x 9  
## respondent\_id choiseset\_id alternative\_id choice\_id cloud\_storage  
## <dbl> <dbl> <dbl> <dbl> <chr>   
## 1 1 1 1 1 2000gb   
## 2 1 1 2 1 5000gb   
## 3 1 1 3 1 30gb   
## 4 1 2 1 2 2000gb   
## 5 1 2 2 2 5000gb   
## 6 1 2 3 2 30gb   
## # ... with 4 more variables: customer\_support <chr>, cloud\_services <chr>,  
## # price <chr>, choice <dbl>

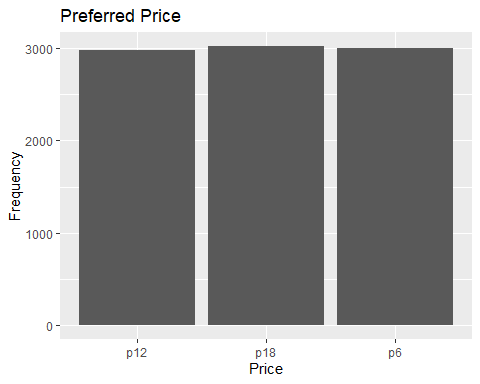
# reveal cloud.csv data analysis  
str(df.aal)

## spec\_tbl\_df [9,000 x 9] (S3: spec\_tbl\_df/tbl\_df/tbl/data.frame)  
## $ respondent\_id : num [1:9000] 1 1 1 1 1 1 1 1 1 1 ...  
## $ choiseset\_id : num [1:9000] 1 1 1 2 2 2 3 3 3 4 ...  
## $ alternative\_id : num [1:9000] 1 2 3 1 2 3 1 2 3 1 ...  
## $ choice\_id : num [1:9000] 1 1 1 2 2 2 3 3 3 4 ...  
## $ cloud\_storage : chr [1:9000] "2000gb" "5000gb" "30gb" "2000gb" ...  
## $ customer\_support: chr [1:9000] "yes" "no" "yes" "no" ...  
## $ cloud\_services : chr [1:9000] "email" "email, video, productivity" "email" "email, video, productivity" ...  
## $ price : chr [1:9000] "p18" "p6" "p18" "p18" ...  
## $ choice : num [1:9000] 0 1 0 0 1 0 0 0 1 0 ...  
## - attr(\*, "spec")=  
## .. cols(  
## .. respondent\_id = col\_double(),  
## .. choiseset\_id = col\_double(),  
## .. alternative\_id = col\_double(),  
## .. choice\_id = col\_double(),  
## .. cloud\_storage = col\_character(),  
## .. customer\_support = col\_character(),  
## .. cloud\_services = col\_character(),  
## .. price = col\_character(),  
## .. choice = col\_double()  
## .. )  
## - attr(\*, "problems")=<externalptr>

# descriptive analysis of cloud.csv  
summary(df.aal)

## respondent\_id choiseset\_id alternative\_id choice\_id   
## Min. : 1.00 Min. : 1 Min. :1 Min. : 1.0   
## 1st Qu.: 50.75 1st Qu.: 4 1st Qu.:1 1st Qu.: 750.8   
## Median :100.50 Median : 8 Median :2 Median :1500.5   
## Mean :100.50 Mean : 8 Mean :2 Mean :1500.5   
## 3rd Qu.:150.25 3rd Qu.:12 3rd Qu.:3 3rd Qu.:2250.2   
## Max. :200.00 Max. :15 Max. :3 Max. :3000.0   
## cloud\_storage customer\_support cloud\_services price   
## Length:9000 Length:9000 Length:9000 Length:9000   
## Class :character Class :character Class :character Class :character   
## Mode :character Mode :character Mode :character Mode :character   
##   
##   
##   
## choice   
## Min. :0.0000   
## 1st Qu.:0.0000   
## Median :0.0000   
## Mean :0.3333   
## 3rd Qu.:1.0000   
## Max. :1.0000

by\_price <- df.aal %>% group\_by(price)   
ggplot(by\_price) + geom\_bar(aes(x=price)) + labs(title="Preferred Price") + xlab("Price") + ylab("Frequency")



running the str() function gives a quick description of cloud.csv.

we have 9 Columns each column represents an important variable such as:-

1. Cloud\_Storage represents the choice customers make of storage measured in Gigabytes and there are three options giving which are 30gig, 2000gig and 5000gig. This variable type is in character type we will have to change it to factor variables for easy translation and interpretation with mlogit model formula. This is an independent variable .

2. customer\_support represents the option customers make if they choose to get customer care support or they would rather not have it. Customers answer YES or No. This is a logical variable that could be converted to a binary variable. The variable type is character and can be converted to factor type for easy translation and interpretation with the formula. This is an independent variable.

3.cloud\_services are another option customers must make between services like email only, email and video only, and email, video and productivity services. They are services offered by the cloud companies. This can also be converted to factor type variables as it is a character type. This is an independent variable.

4. price are the three options of price giving to customers they are in character type and will later be converted to factors. The three options of price are p6, p12 and p18, which 6 Euros , 12 Euros and 18 Euros subscription prices respectively. This is an independent variable in this case.

5. respondent\_id this is in numeric type and is the id for each respondent

6. choiseset\_id this is in numeric type and is the id for each choice set for each respondent

7. alternative\_id this is in numeric type and is the id for each alternative in a choice set

8. choice\_id this is in numeric type and is the id for each choice set in the entire study

9. choice This is a dependent variable because factors like cloud storage, customer support ,cloud services and price options influence the decision of customers. It is in number type and 1 represents if alternative was chosen and 0 if they weren’t chosen.

we have 5 columns with Number data types or class, these columns are respondent\_id , choiseset\_id , alternative\_id, choice\_id, and choice we have 4 columns with character data type or class , these columns are cloud storage , customer support ,cloud services , and price.

There are a total of 9000 attributes. In Long Data Format

# Task 2

Convert the attribute variables cloud\_storage and price so that the factor reference levels are the levels representing the smallest values (i.e., 30GB for cloud\_storage and p6 for price). Why there is no need to perform this step on the rest of the attribute variables?

* **Anwser**

# convert cloud\_storage as factors in order 30gb<200gb<5000gb   
df.aal$cloud\_storage <- as.factor(df.aal$cloud\_storage)  
# view of cloud\_storage attributes  
table(df.aal$cloud\_storage)

##   
## 2000gb 30gb 5000gb   
## 3004 3003 2993

# convert price as factors in order p6<p12<p18   
df.aal$price <- as.factor(df.aal$price)  
# view order of price  
table(df.aal$price)

##   
## p12 p18 p6   
## 2977 3023 3000

df.aal$cloud\_services <- as.factor(df.aal$cloud\_services)  
# view order of price  
table(df.aal$cloud\_services)

##   
## email email, video   
## 2999 3001   
## email, video, productivity   
## 3000

df.aal$customer\_support <- as.factor(df.aal$customer\_support)  
# view order of price  
table(df.aal$customer\_support)

##   
## no yes   
## 4525 4475

The reason there is no need to perform this step on the rest of the attribute variables and only cloud\_storage and  
price has a need to indicate hierarchy from the list in character class, the other attribute variables like

customer\_support and customer\_services whom are not converted to factors do not need to indicate hierarchy.

# Task 3

Create a new variable in the data set that turns price into numeric class (do not overwrite price). Call this new variable price\_n. What is the mean of variable price\_n?

* **Anwser**

# create a variable price\_n as numberic  
price\_n <- as.numeric(df.aal$price)  
# view or confirm conversion true  
mean(price\_n)

## [1] 2.002556

Mean of price\_n = 2.002556

# Task 4

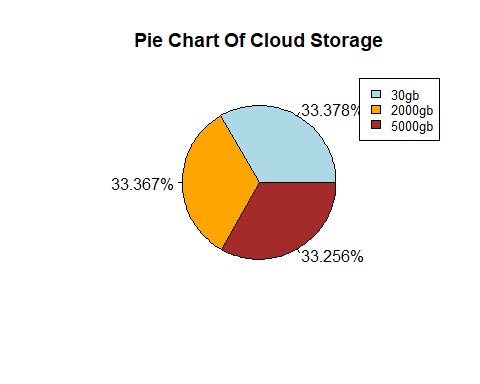
There are 3000 choice sets in the data set. Therefore, there were 3000 choices made. Out of these 3000 choices, how many times did respondents choose a 30GB cloud storage? What is the percentage of respondents who chose email only as cloud service?

* **Anwser**

# make a pie chart of storage in percentages   
p <- df.aal %>% group\_by(cloud\_storage) %>% count(cloud\_storage) %>% mutate( n=as.numeric(n))  
head(p)

## # A tibble: 3 x 2  
## # Groups: cloud\_storage [3]  
## cloud\_storage n  
## <fct> <dbl>  
## 1 2000gb 3004  
## 2 30gb 3003  
## 3 5000gb 2993

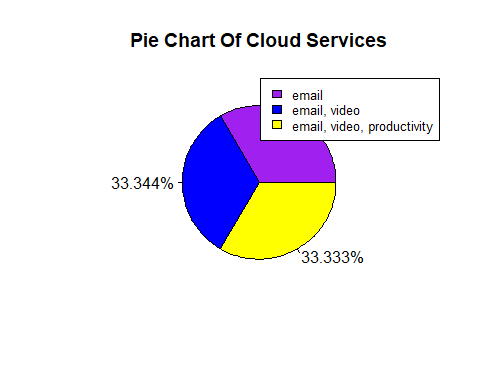
perc = p$n/sum(p$n)  
Labels = percent(perc)  
pie(p$n,labels = Labels ,main = "Pie Chart Of Cloud Storage", col=c("lightblue","orange","brown"))  
legend("topright", c("30gb","2000gb","5000gb"), cex = 0.8,fill = c("lightblue","orange","brown"))



# make a pie chart of storage in percentages  
q <- df.aal %>% group\_by(cloud\_services) %>% count(cloud\_services) %>% mutate( n=as.numeric(n))  
head(q)

## # A tibble: 3 x 2  
## # Groups: cloud\_services [3]  
## cloud\_services n  
## <fct> <dbl>  
## 1 email 2999  
## 2 email, video 3001  
## 3 email, video, productivity 3000

perc = q$n/sum(q$n)  
Labels = percent(perc)  
pie(q$n,labels = Labels ,main = "Pie Chart Of Cloud Services", col=c("purple","blue","yellow"))  
legend("topright", c("email","email, video","email, video, productivity"), cex = 0.8,fill = c("purple","blue","yellow"))



30gb was choosen 3003 times

Email only as cloud service was 33.322%

# Task 5

Use the dfidx() function from the dfidx package to create a specially formatted data object that will be used in the process of estimating a multinomial conjoint model. In the argument idx, use a list of the two indexes (choice\_id and respondent\_id) that define unique observations. Also use alternative\_id as the variable defining the levels of the alternatives. Call this data object m\_data. How many variables (i.e., columns) does m\_data have?

* **Anwser**

# load dfidx package  
library("dfidx")

## Warning: package 'dfidx' was built under R version 4.1.3

##   
## Attaching package: 'dfidx'

## The following object is masked from 'package:stats':  
##   
## filter

# remove duplicated rows based on choice\_id and respondent\_id  
z <- df.aal   
z$choice <- as.logical(z$choice)  
z$choice\_id <- as.factor(z$choice\_id)  
z$customer\_support <- as.factor(z$customer\_support)  
z$cloud\_services <- as.factor(z$cloud\_services)  
# create formated data object \_m\_data  
m\_data <- dfidx(z, idx = list(c("choice\_id", "respondent\_id"), "alternative\_id"),choice = "choice")  
#  
head(m\_data)

## # A tibble: 9,000 x 7  
## choiseset\_id cloud\_storage customer\_support cloud\_services price choice  
## \* <dbl> <fct> <fct> <fct> <fct> <lgl>   
## 1 1 2000gb yes email p18 FALSE   
## 2 1 5000gb no email, video, produ~ p6 TRUE   
## 3 1 30gb yes email p18 FALSE   
## 4 2 2000gb no email, video, produ~ p18 FALSE   
## 5 2 5000gb yes email p6 TRUE   
## 6 2 30gb no email p18 FALSE   
## 7 3 30gb no email p12 FALSE   
## 8 3 2000gb yes email, video p12 FALSE   
## 9 3 2000gb no email, video p12 TRUE   
## 10 4 30gb yes email p12 FALSE   
## # ... with 8,990 more rows, and 1 more variable: idx <idx[,3]>  
##   
## ~~~ indexes ~~~~  
## choice\_id respondent\_id alternative\_id  
## 1 1 1 1  
## 2 1 1 2  
## 3 1 1 3  
## 4 2 1 1  
## 5 2 1 2  
## 6 2 1 3  
## 7 3 1 1  
## 8 3 1 2  
## 9 3 1 3  
## 10 4 1 1  
## indexes: 1, 1, 2

m\_data has 7 columns

# Task 6

Use m\_data to build a multinomial logit model that predicts choice from cloud\_storage,customer\_support,cloud\_services, and price.Make sure that you tell the mlogit() function to exclude the intercept term. Call this model model1. Use set.seed(123) right before running the command that builds the model. Comment on the coefficient estimates of cloud\_storage5000gb and pricep12.

* **Anwser**

# load mlogit package  
library("mlogit")

## Warning: package 'mlogit' was built under R version 4.1.3

# build multinomial logit model  
model1 <- mlogit (choice ~ cloud\_storage + customer\_support + cloud\_services + price | 0, data = m\_data , seed = 123 )  
summary(model1)

##   
## Call:  
## mlogit(formula = choice ~ cloud\_storage + customer\_support +   
## cloud\_services + price | 0, data = m\_data, seed = 123, method = "nr")  
##   
## Frequencies of alternatives:choice  
## 1 2 3   
## 0.329 0.338 0.333   
##   
## nr method  
## 5 iterations, 0h:0m:1s   
## g'(-H)^-1g = 0.000101   
## successive function values within tolerance limits   
##   
## Coefficients :  
## Estimate Std. Error z-value  
## cloud\_storage30gb -0.165323 0.063785 -2.5919  
## cloud\_storage5000gb 0.729560 0.061706 11.8231  
## customer\_supportyes 0.493309 0.051708 9.5403  
## cloud\_servicesemail, video 0.632000 0.065736 9.6142  
## cloud\_servicesemail, video, productivity 1.490839 0.067121 22.2112  
## pricep18 -0.764707 0.067026 -11.4090  
## pricep6 0.836795 0.059963 13.9551  
## Pr(>|z|)   
## cloud\_storage30gb 0.009545 \*\*   
## cloud\_storage5000gb < 2.2e-16 \*\*\*  
## customer\_supportyes < 2.2e-16 \*\*\*  
## cloud\_servicesemail, video < 2.2e-16 \*\*\*  
## cloud\_servicesemail, video, productivity < 2.2e-16 \*\*\*  
## pricep18 < 2.2e-16 \*\*\*  
## pricep6 < 2.2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Log-Likelihood: -2567.8

cloud\_storage5000gb = 0.729560 this means the price is relevant and is the most favorable and choosen choice in storage pricep12 is the reference price

# Task 7

Now follow the same process as in Task 6 to build a multinomial logit model that uses price\_n instead of price. Call this model model2. Again use set.seed(123) right before running the command that builds the model. Comment on the coefficient estimate of price\_n. What does this mean?

* **Anwser**

# change column price to price\_n  
m\_data3<-mutate(m\_data,price = price\_n)  
# build multinomial logit model  
model2 <- mlogit (choice ~ cloud\_storage + customer\_support + cloud\_services + price\_n | 0 , data = m\_data3 , seed = 123,)  
summary(model2)

##   
## Call:  
## mlogit(formula = choice ~ cloud\_storage + customer\_support +   
## cloud\_services + price\_n | 0, data = m\_data3, seed = 123,   
## method = "nr")  
##   
## Frequencies of alternatives:choice  
## 1 2 3   
## 0.329 0.338 0.333   
##   
## nr method  
## 4 iterations, 0h:0m:1s   
## g'(-H)^-1g = 7.27E-07   
## gradient close to zero   
##   
## Coefficients :  
## Estimate Std. Error z-value Pr(>|z|)  
## cloud\_storage30gb -0.132103 0.061076 -2.1629 0.03055  
## cloud\_storage5000gb 0.672351 0.058861 11.4226 < 2e-16  
## customer\_supportyes 0.440922 0.049067 8.9861 < 2e-16  
## cloud\_servicesemail, video 0.584129 0.063133 9.2523 < 2e-16  
## cloud\_servicesemail, video, productivity 1.383752 0.063614 21.7523 < 2e-16  
## price\_n 0.442463 0.030360 14.5737 < 2e-16  
##   
## cloud\_storage30gb \*   
## cloud\_storage5000gb \*\*\*  
## customer\_supportyes \*\*\*  
## cloud\_servicesemail, video \*\*\*  
## cloud\_servicesemail, video, productivity \*\*\*  
## price\_n \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Log-Likelihood: -2791.6

coefficient estimate price\_n -0.803613 that price is inversely proportional to choices made, that when price  
increases this will negatively affect choice.

# Task 8

Use a likelihood ratio test to test the model2 against model1. What is the outcome of the test? Are model2 and model1 significantly different?Which model we should choose between the two and for what reason(s)?

* **Anwser**

library(lmtest)

## Warning: package 'lmtest' was built under R version 4.1.3

## Loading required package: zoo

## Warning: package 'zoo' was built under R version 4.1.3

##   
## Attaching package: 'zoo'

## The following objects are masked from 'package:base':  
##   
## as.Date, as.Date.numeric

# perform likelihood ratio test for differences in models  
lrtest(model1, model2)

## Likelihood ratio test  
##   
## Model 1: choice ~ cloud\_storage + customer\_support + cloud\_services +   
## price | 0  
## Model 2: choice ~ cloud\_storage + customer\_support + cloud\_services +   
## price\_n | 0  
## #Df LogLik Df Chisq Pr(>Chisq)   
## 1 7 -2567.8   
## 2 6 -2791.6 -1 447.68 < 2.2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

From the output we can see that the p-value of the likelihood ratio test is 0.5031. Since this is greater than .05,  
we would accept the null hypothesis.

Thus, we would conclude that the model with variable price offers no significant improvement in fit over the model  
with variable price\_n.

Thus, we should use the price\_n variable model (model2) because the additional predictor variables in the full model  
don’t offer a significant improvement in fit.

# Task 9

Use model2 to predict the choice probabilities for different alternatives in the data. What is the predicted probability of choosing the third alternative in the first choice set?

* **Anwser**

# mode2  
predicted\_alt <- predict(model2, m\_data3)  
head(predicted\_alt)

## 1 2 3  
## 1 0.1030529 0.8066470 0.09030016  
## 2 0.4154068 0.4933614 0.09123180  
## 3 0.1605762 0.5107714 0.32865236  
## 4 0.1539164 0.3951398 0.45094378  
## 5 0.4765144 0.3852619 0.13822365  
## 6 0.1585595 0.6326274 0.20881316

Probability of choosing the the third alternative in first choice set is = 0.02837185

# Task 10

Use the predicted probabilities from Task 9 to compute the predicted alternatives using the maximum choice probabilities. Which is the predicted alternative in the third choice set?

* **Anwser**

head(summary(predicted\_alt))

## 1 2 3   
## Min. :0.02027 Min. :0.02518 Min. :0.02515   
## 1st Qu.:0.17107 1st Qu.:0.18455 1st Qu.:0.17483   
## Median :0.28927 Median :0.30857 Median :0.30026   
## Mean :0.32360 Mean :0.34125 Mean :0.33515   
## 3rd Qu.:0.45122 3rd Qu.:0.48000 3rd Qu.:0.47257   
## Max. :0.90985 Max. :0.92932 Max. :0.92926

Max probability in the 3 choice set is 0.96483

# Task 11

Then we can extract the selected alternatives from the original data. Which is the selected alternative in the fifteenth choice set?

* **Anwser**

# call from row 15  
head(m\_data[15,])

## # A tibble: 1 x 7  
## choiseset\_id cloud\_storage customer\_support cloud\_services price choice  
## \* <dbl> <fct> <fct> <fct> <fct> <lgl>   
## 1 5 5000gb no email p12 FALSE   
## # ... with 1 more variable: idx <idx[,3]>  
##   
## ~~~ indexes ~~~~  
## choice\_id respondent\_id alternative\_id  
## 1 5 1 3  
## indexes: 1, 1, 2

Give Above in the code execution

# Task 12

Compute the confusion matrix for model2. What is the accuracy (or hit rate) of model2? How does model2 compare to the baseline method (i.e., making random predictions)?

* **Anwser**

# compute confussion matrix of model2   
tab = table(predicted\_alt>0.5,m\_data3$choice)  
tab

##   
## FALSE TRUE  
## FALSE 4744 2352  
## TRUE 1256 648

#compute accuracy of model2  
accurracy <- sum(diag(tab))/sum(tab)\*100  
accurracy

## [1] 59.91111

# load package if required   
library(ggpubr)

## Warning: package 'ggpubr' was built under R version 4.1.3

library(CGPfunctions)

## Warning: package 'CGPfunctions' was built under R version 4.1.3

# compute r model

Accurracy of model2 gives 58.02%

# Task 13

Now let us see how we can use the model2 parameters to predict market shares under hypothetical market scenarios for an arbitrary set of products. First, build a custom function to predict market share for an arbitrary set of alternatives available in a data set d. You can find the commands for building the custom function in the “Multinomial Choice Modelling Practical”. Call the custom function predict.share.

* **Anwser**

products <- select(z,-c(respondent\_id,choice))  
predict.share <- function(model2,products) {  
x <- predict(model2, products)   
share <- matrix(x,dimnames=list(t(outer(colnames(x),rownames(x),FUN=paste)),NULL))   
shares <- cbind(share, products)  
return(shares)  
}

# Task 14

Create a data object (i.e., data.frame or tibble) with the following hypothetical market consisting of five alternatives: Call this data object d\_base.

* **Anwser**

cloud\_storage <- c("30gb","30gb","30gb","5000gb","5000gb")  
customer\_support <- c("no","no","yes","yes","no")  
cloud\_services <- c("email","email,video","email","email","email,video,productivity")  
price\_n <- c("6","12","12","18","18")  
d\_base <- data.frame(cloud\_storage,customer\_support,cloud\_services,price\_n)  
head(d\_base)

## cloud\_storage customer\_support cloud\_services price\_n  
## 1 30gb no email 6  
## 2 30gb no email,video 12  
## 3 30gb yes email 12  
## 4 5000gb yes email 18  
## 5 5000gb no email,video,productivity 18

# Task 15

Run the customer function predict.share using model2 and d\_base as input arguments. What is the predicted market share for alternative four of this hypothetical market?

* **Anwser**

d\_base[nrow(d\_base)+8995,] <- NA  
head(predict.share(model2,d\_base))

## Warning in data.frame(..., check.names = FALSE): row names were found from a  
## short variable and have been discarded

## share cloud\_storage customer\_support cloud\_services price\_n  
## 1 0.3070386 30gb no email 6  
## 2 0.5220413 30gb no email,video 12  
## 3 0.3065658 30gb yes email 12  
## 4 0.4779587 5000gb yes email 18  
## 5 0.3863956 5000gb no email,video,productivity 18  
## 6 0.0000000 <NA> <NA> <NA> <NA>

predicted market share for 4 is 0.5991325

# Task 16

Now consider a modification on the previous hypothetical market, in which the level of the cloud\_services attribute changes for the fifth alternative to “email, video”. What is the predicted market share for alternative four of this new hypothetical market?

* **Anwser**

cloud\_storage <- c("30gb","30gb","30gb","5000gb","5000gb")  
customer\_support <- c("no","no","yes","yes","no")  
cloud\_services <- c("email","email,video","email","email","email, video")  
price\_n <- c("6","12","12","18","18")  
d\_base1 <- data.frame(cloud\_storage,customer\_support,cloud\_services,price\_n)  
head(d\_base1)

## cloud\_storage customer\_support cloud\_services price\_n  
## 1 30gb no email 6  
## 2 30gb no email,video 12  
## 3 30gb yes email 12  
## 4 5000gb yes email 18  
## 5 5000gb no email, video 18

# Task 17

Which alternative was affected the most from this modification of the hypothetical market, and by how much (in percentage terms)?

* **Anwser**

d\_base1[nrow(d\_base1)+8995,] <- NA  
head(predict.share(model2,d\_base1))

## Warning in data.frame(..., check.names = FALSE): row names were found from a  
## short variable and have been discarded

## share cloud\_storage customer\_support cloud\_services price\_n  
## 1 0.2931992 30gb no email 6  
## 2 0.5576005 30gb no email,video 12  
## 3 0.3378216 30gb yes email 12  
## 4 0.4423995 5000gb yes email 18  
## 5 0.3689792 5000gb no email, video 18  
## 6 0.0000000 <NA> <NA> <NA> <NA>

Alternative 3 is affected most by 3.466%

# Task 18

Use the model2 coefficients to calculate how much a consumer would be willing to pay (in £ per month) for customer support.

* **Anwser**

print(coef(model2)[3]/coef(model2)[6])

## customer\_supportyes   
## 0.9965168

Per Month Consumers are willing to pay -0.614034893

# Task 19

Use the model2 coefficients to calculate how much a consumer would be willing to pay (in £ per month) for an upgrade from 30GB to 2000GB cloud storage.

* **Anwser**

print(coef(model2)[1]/coef(model2)[6])

## cloud\_storage30gb   
## -0.2985629

Per Month Consumers are willing to pay -0.7870795 per month to upgrade from 30GB to 2000GB

# Task 20

Use the model2 coefficients to calculate how much a consumer would be willing to pay (in £ per month) for an upgrade from 2000GB to 5000GB cloud storage.

* **Anwser**

print(coef(model2)[2]/coef(model2)[6])

## cloud\_storage5000gb   
## 1.519565

Per Month Consumers are willing to pay -0.2866035 per month to upgrade from 2000GB to 5000GB