Machine Learning Project

Customer Personality Analysis

Mario Avolio
880995

Rocco Gianni Rapisarda
845197

8 gennaio 2022

Sommario

L'apprendimento automatico e la statistica sono discipline strettamente collegate. Secondo Michael I. Jordan, le idee dell'apprendimento automatico, dai principi metodologici agli strumenti teorici, sono stati sviluppati prima in statistica. In questo elaborato si riporta l'attività di sviluppo e sperimentazione di diversi modelli di **Machine Learning** per l'analisi approfondita dei clienti ideali per una generica azienda. La concentrazione è stata focalizzata soprattutto sul **Clustering** mediante l'algoritmo **K-Means**, sebbene nel corso della trattazione si esporranno anche altre metodologie utilizzate per l'analisi dei dati.

1 Descrizione del dominio di riferimento e obiettivi dell'elaborato

Molto spesso lo sviluppo di nuovi prodotti o servizi è attivato dall'imitazione dei concorrenti e da analisi di mercato generiche, mentre al cliente si dedica poca attenzione. L'acquisto è prima di tutto un'esperienza ed è necessario comprendere quali bisogni la guidano: solo così ogni segmento di mercato individuato sarà connesso con la capacità dell'azienda di soddisfare le aspettative dei clienti, comprese quelle inespresse. Progettare, sviluppare e vendere prodotti non connessi con il proprio target rappresenta un costo insostenibile, mentre è necessario progettare uno sviluppo in linea con la *customer satisfaction*. Per questo motivo Customer Personality Analysis riguarda un'analisi dettagliata dei clienti ideali per una generica azienda. Il compito fondamentale è quello di aiutare un'attività commerciale a comprendere meglio i propri compratori al fine di rendere più semplice la modifica e la scelta dei propri prodotti, in relazione alle esigenze richieste dagli acquirenti. L'obiettivo che ha spinto ad analizzare questo insieme di dati è inerente alle **diverse** personalità e comportamenti che gli acquirenti assumono durante il ruolo di potenziali clienti aziendali. Per questo motivo le aziende non possono adottare lo stesso approccio per ogni tipologia di plausibile compratore.

2 Scelte di design, ipotesi e assunzioni

TODO

3 Descrizione del training set

3.1 Attributi

Si fornisce la descrizione originale degli attributi analizzati.

People

- ID: Customer's unique identifier
- Year_Birth: Customer's birth year
- Education: Customer's education level
- Marital_Status: Customer's marital status
- Income: Customer's yearly household income
- Kidhome: Number of children in customer's household
- Teenhome: Number of teenagers in customer's household
- Dt_Customer: Date of customer's enrollment with the company
- Recency: Number of days since customer's last purchase
- Complain: 1 if the customer complained in the last 2 years, 0 otherwise

Products

- MntWines: Amount spent on wine in last 2 years
- MntFruits: Amount spent on fruits in last 2 years
- MntMeatProducts: Amount spent on meat in last 2 years
- MntFishProducts: Amount spent on fish in last 2 years
- MntSweetProducts: Amount spent on sweets in last 2 years
- MntGoldProds: Amount spent on gold in last 2 years

Promotion

- NumDealsPurchases: Number of purchases made with a discount
- AcceptedCmp1: 1 if customer accepted the offer in the 1st campaign, 0 otherwise
- AcceptedCmp2: 1 if customer accepted the offer in the 2nd campaign, 0 otherwise
- AcceptedCmp3: 1 if customer accepted the offer in the 3rd campaign, 0 otherwise
- AcceptedCmp4: 1 if customer accepted the offer in the 4th campaign, 0 otherwise
- AcceptedCmp5: 1 if customer accepted the offer in the 5th campaign, 0 otherwise
- Response: 1 if customer accepted the offer in the last campaign, 0 otherwise

Place

- NumWebPurchases: Number of purchases made through the company's website
- NumCatalogPurchases: Number of purchases made using a catalogue
- NumStorePurchases: Number of purchases made directly in stores
- NumWebVisitsMonth: Number of visits to company's website in the last month

La tabella 1 fornisce un'iniziale descrizione della tipologia di variabili presenti nel dataset.

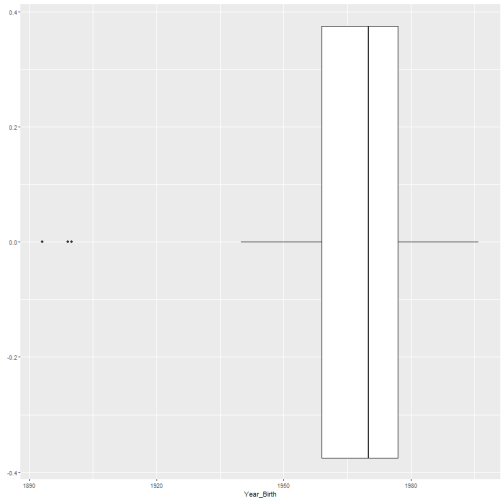
	sapply(customers, class)
ID	integer
Year_Birth	integer
Education	character
Marital_Status	character
Income	integer
Kidhome	integer
Teenhome	integer
Dt_Customer	character
Recency	integer
MntWines	integer
MntFruits	integer
MntMeatProducts	integer
MntFishProducts	integer
MntSweetProducts	integer
MntGoldProds	integer
NumDealsPurchases	integer
NumWebPurchases	integer
NumCatalogPurchases	integer
NumStorePurchases	integer
NumWebVisitsMonth	integer
AcceptedCmp3	integer
AcceptedCmp4	integer
AcceptedCmp5	integer
AcceptedCmp1	integer
AcceptedCmp2	integer
Complain	integer
Z_CostContact	integer
Z_Revenue	integer
Response	integer

Tabella 1: Output funzione *sapply(customers, class)*

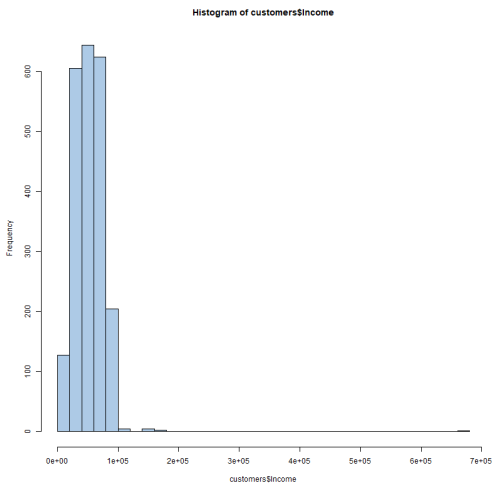
3.2 Prime analisi

Prima di fornire un'analisi dettagliata degli elementi del dataset si è ritenuto necessario effettuare una prima ispezione di alto livello, senza entrare nel dettaglio di ciascun attributo. Il dataset viene importato mediante la funzione:

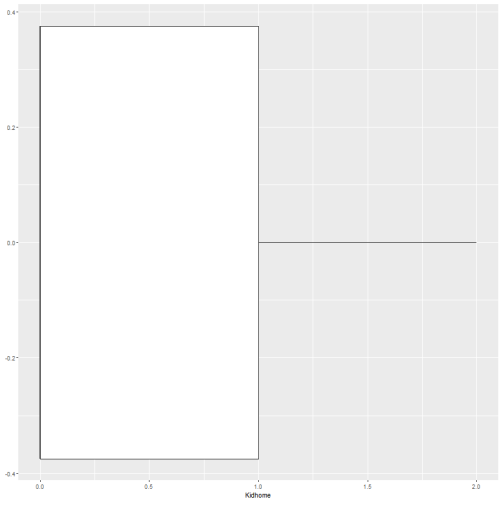
```
customers <- read.csv(paste(getwd(),"/Data/marketing_campaign.csv",sep = ""),
    header=TRUE, sep="\t", stringsAsFactors=F) # use TAB as separator!
```



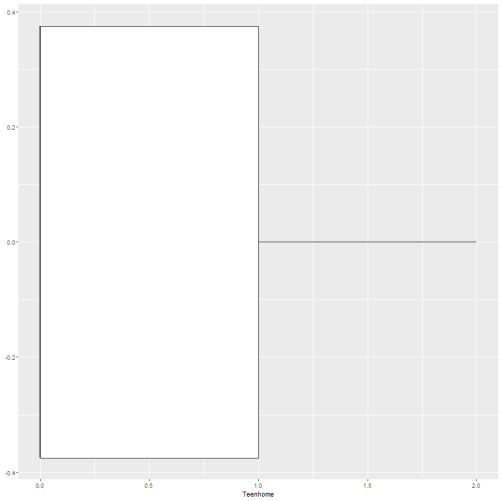
(a) BoxPlot Year_Birth



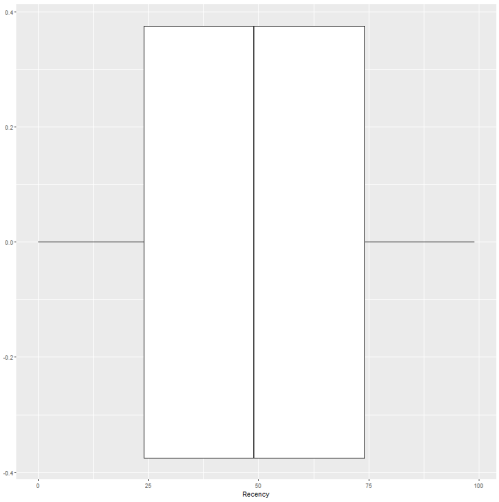
(b) BoxPlot Income



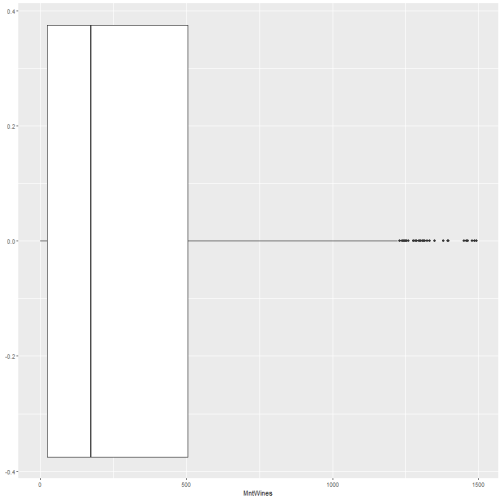
(c) BoxPlot KidHome



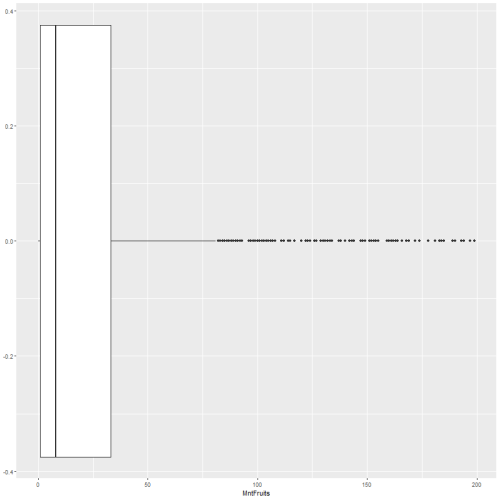
(d) BoxPlot TeenHome



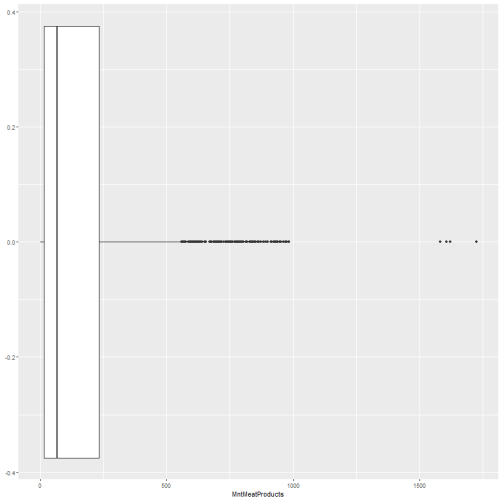
(e) BoxPlot Recency



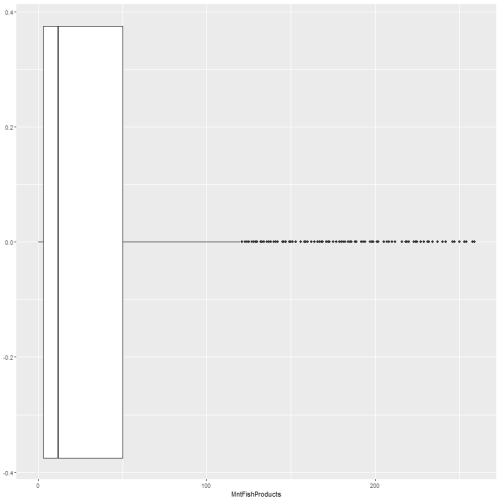
(f) BoxPlot MntWines



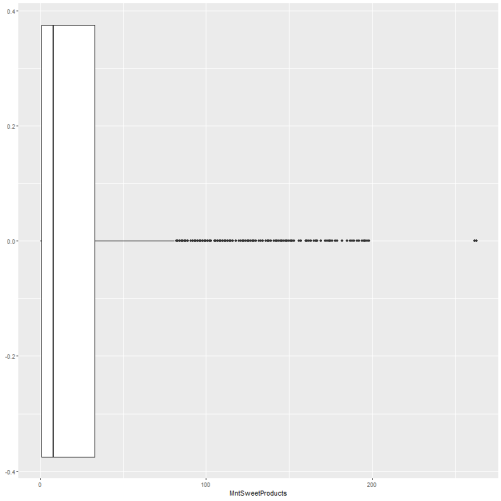
(g) BoxPlot MntFruits



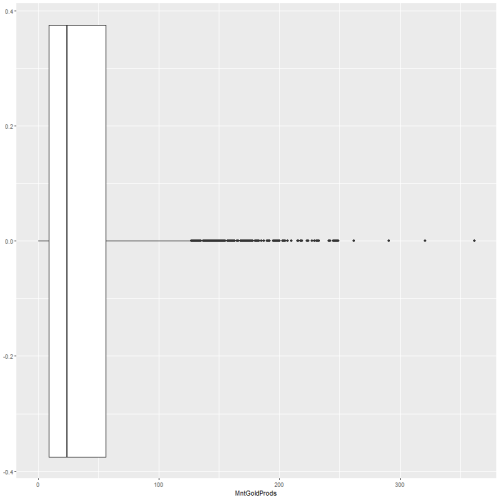
(h) BoxPlot MntMeatProducts



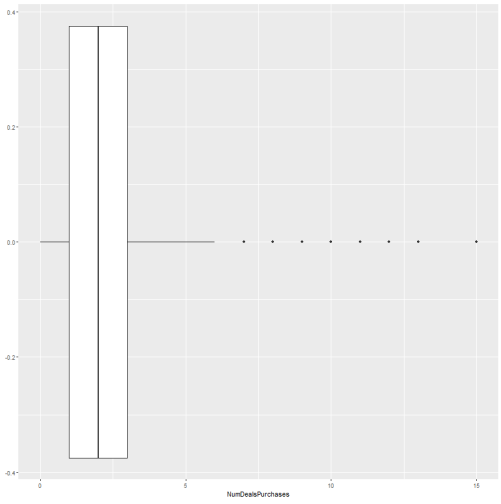
(i) BoxPlot MntFishProducts



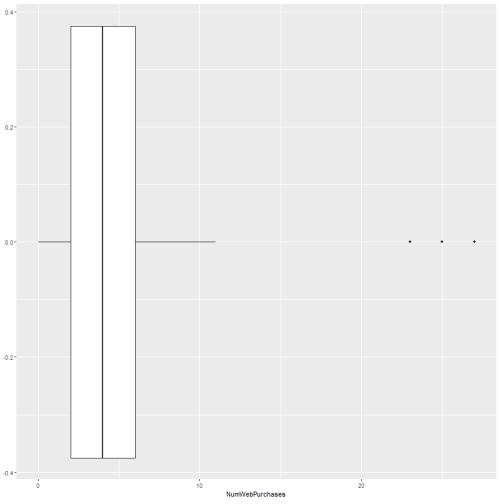
(j) BoxPlot MntSweetProducts



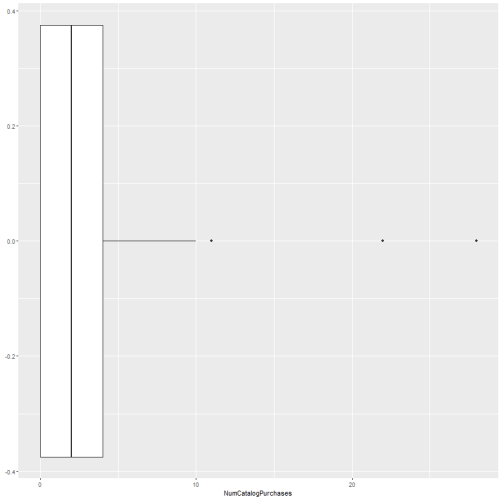
(k) BoxPlot MntGoldProds



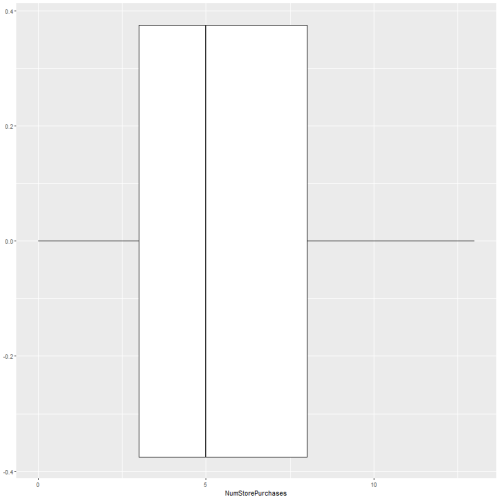
(l) BoxPlot numDealsPurchases



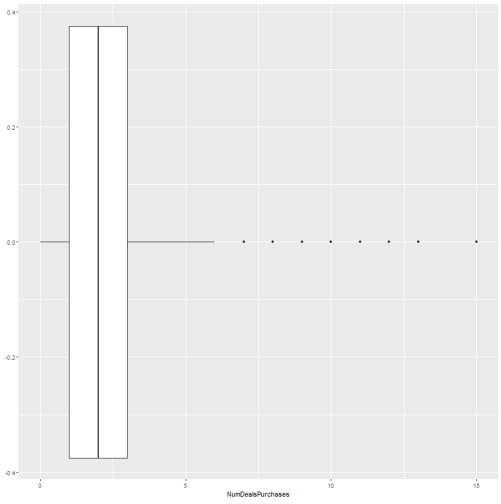
(m) BoxPlot NumWebPurchases



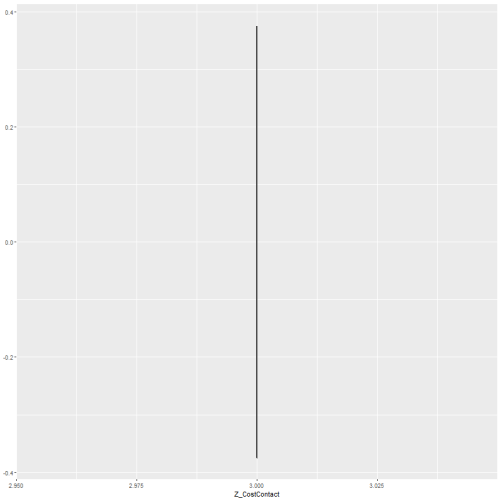
(n) BoxPlot NumCatalogPurchases



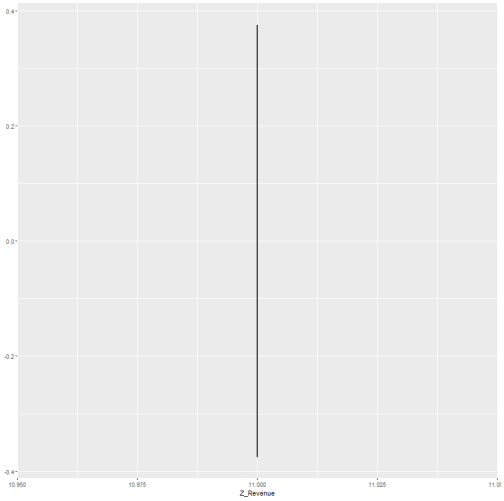
(o) BoxPlot NumStorePurchases



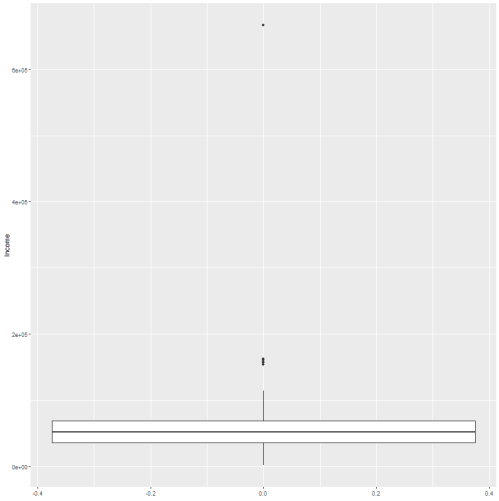
(p) BoxPlot NumDealsPurchases



(q) BoxPlot Z_CostContact



(r) BoxPlot Z_Revenue



(s) BoxPlot Income

Durante questa prima indagine sono stati effettuati controlli di base sulle variabili. In particolare, dalle analisi dei boxplot riportati rispettivamente nelle figure 1(r) e 1(q) è doveroso notare la mancanza di **varianza** di tali variabili, questo aspetto sarà successivamente preso in considerazione durante la fase di preprocessing.
Inoltre durante l'analisi delle figure 1(s) e 1(b), mediante un apposito *warning* durante l'esecuzione del codice R, si è notata la presenza di alcuni valori mancanti.

```
hist(customers$Income,40,col="#adcae6")
ggplot(customers, aes(y = Income)) + geom_boxplot()
n_miss(customers) # counting the total number of missing values in the data
miss_var_summary(customers) # Summarizing missingness in each variable
```

La tabella 2 mostra l'output del comando *miss_var_summary(customers)*, dove si possono notare 24 valori mancanti nella variabile *Income*.

	variable	n_miss	pct_miss
1	Income	24	1.07
2	ID	0	0.00
3	Year_Birth	0	0.00
4	Education	0	0.00
5	Marital_Status	0	0.00
6	Kidhome	0	0.00
7	Teenhome	0	0.00
8	Dt_Customer	0	0.00
9	Recency	0	0.00
10	MntWines	0	0.00
11	MntFruits	0	0.00
12	MntMeatProducts	0	0.00
13	MntFishProducts	0	0.00
14	MntSweetProducts	0	0.00
15	MntGoldProds	0	0.00
16	NumDealsPurchases	0	0.00
17	NumWebPurchases	0	0.00
18	NumCatalogPurchases	0	0.00
19	NumStorePurchases	0	0.00
20	NumWebVisitsMonth	0	0.00
21	AcceptedCmp3	0	0.00
22	AcceptedCmp4	0	0.00
23	AcceptedCmp5	0	0.00
24	AcceptedCmp1	0	0.00
25	AcceptedCmp2	0	0.00
26	Complain	0	0.00
27	Z_CostContact	0	0.00
28	Z_Revenue	0	0.00
29	Response	0	0.00

Tabella 2: Output funzione *miss_var_summary(customers)*

3.3 Data Preprocessing

La fase di *data preprocessing* è risultata fondamentale per la buona riuscita dell'analisi dei dati preposti. Essa si basa su quattro principali stadi:

- Refactor del dataset
- Risoluzione dei valori mancanti nella variabile *income*
- Splitting del dataset in *trainingSet* e *testSet*
- Feature Scaling

3.3.1 Refactor del Dataset

In questa fase ci si è concentrati su diversi aspetti migliorativi. In primo luogo si è voluto essettuare un'azione di incorporamento di dati. La motivazione è dovuta principalmente alla presenza di dati

ridondati all'interno di molte variabili, in particolare si vuole porre l'attenzione su: *Marital_Status* e *Education*. Difatti utilizzando la funzione *unique*, come riportato nelle tabelle 4 e 3, si è potuto rilevare la presenza di troppi elementi superflui.

	unique(customers$Marital_Status)
1	Single
2	Together
3	Married
4	Divorced
5	Widow
6	Alone
7	Absurd
8	YOLO

Tabella 3: Output *unique*(*customers*$*Marital_Status*)

	unique(customers$Education)
1	Graduation
2	PhD
3	Master
4	Basic
5	2n Cycle

Tabella 4: Output *unique*(*customers*$*Education*)

Come mostrato dal codice sottostante, l'obiettivo è stato quello di diminuire, per favorire l'analisi mediante i diversi algoritmi utilizzati, in numero delle categorie di valori per le relative variabili.

```
# ------------------------------------ COLLAPSING
#Collapsing marital Status into two categories: Single & Couple
unique(customers$Marital_Status)
customers <- mutate(customers, Marital_Status = replace(Marital_Status, Marital_Status
    == "Divorced" | Marital_Status == "Widow" | Marital_Status == "Alone" |
    Marital_Status == "Absurd" | Marital_Status == "YOLO", "Single"))
customers <- mutate(customers, Marital_Status = replace(Marital_Status, Marital_Status
    == "Together" | Marital_Status == "Married", "Couple"))

#Collapsing the Education into two Categories: graduate and non-graduate
unique(customers$Education)
customers <- mutate(customers, Education = replace(Education, Education == "Graduation"|
    Education == "PhD" | Education == "Master", "graduate"))
customers <- mutate(customers, Education = replace(Education, Education == "Basic"|
    Education == "2n Cycle", "non-graduate"))
# ------------------------------------
```

In particolar modo si è deciso di fornire due possibili valori riassuntivi per ciascuna variabile, come indicato nelle tabelle 5 e 6.

La fase di *refactoring* si è anche occupata della convesione in *factor* degli elementi *character* all'interno dell'insieme di dati. Il codice sottostante mostra la procedura seguita. Le tabelle 8 e 7 mostrano il risultato di tale procedura.

```
# ------------------------------------ CONVERSION
```

	unique(customers$Education)
1	graduate
2	non-graduate

Tabella 5: Output *unique*(*customers*$*Education*) dopo la procedura di *collapsing* dei dati

	unique(customers$Marital_Status)
1	Single
2	Couple

Tabella 6: Output *unique*(*customers*$*Marital_Status*) dopo la procedura di *collapsing* dei dati

```
#Converting them to factors
customers <- mutate(customers, Marital_Status = as.factor(Marital_Status), Education =
    as.factor(Education))

# Encoding the categorical features to numeric
customers <- mutate(customers, Education = case_when(Education == "graduate" ~ 1,
                                                     Education == "non-graduate" ~ 0))
customers <- mutate(customers, Marital_Status = case_when(Marital_Status == "Couple" ~ 1,
                                                          Marital_Status == "Single" ~ 0))
# -------------------------------------
```

	head(customers$Marital_Status)
1	0.00
2	0.00
3	1.00
4	1.00
5	1.00
6	1.00

Tabella 7: Output *head*(*customers*$*Marital_Status*)

	head(customers$Education)
1	1.00
2	1.00
3	1.00
4	1.00
5	1.00
6	1.00

Tabella 8: Output *head*(*customers*$*Education*)

Questa fase si è anche occupata della creazione di nuove variabili partendo da quelle già presenti all'interno da quelle già esistenti. Si ponga l'attenzione in particolar modo alle **categorie** di variabili *Mnt*, *Accepted*, *KidHome*, *TeenHome*. Tali categorie possono essere sommate per creare nuove variabili riassuntive. Il codice seguente e la tabella 9 ne riportano un esempio:

```
# ------------------------------------ TOTAL
#Creating a new variable:Total_spent
```

```
customers <- mutate(customers, Total_spent = MntWines + MntFruits + MntMeatProducts +
    MntFishProducts + MntSweetProducts + MntGoldProds)

# Details about previous campains also combined. Creating a new variable:Total_Campains
customers <- mutate(customers, Total_Campains = AcceptedCmp1 + AcceptedCmp2 +
    AcceptedCmp3 + AcceptedCmp4 + AcceptedCmp5)

# These variables can be combined and we can get the no of children for the customers.
    Creating a new variable:Total_Childs
customers <- mutate(customers, Total_Childs = Kidhome + Teenhome)
# ------------------------------------
```

	Total_spent	Total_Campains	Total_Childs
1	1617	0	0
2	27	0	2
3	776	0	0
4	53	0	1
5	422	0	1
6	716	0	1

Tabella 9: Primi valori delle variabili Total_spent & Total_Campains & Total_Childs

Per finire è doveroso sottolineare che in questa sezioni ci si è anche occupati dell'eliminazione delle variabili superflue, come *Z_CostContact* e *Z_Revenue* che non hanno varianza, e della sostituzione della varibile *Year_Birth* con *Age*.

```
# we can calculate customer age from the birth year. It will be more usefull to our
    analysis.
# creating a new variable Age from Year of Birth
thisYear <- as.numeric(format(as.Date(Sys.Date(), format="%d-%m-%Y"),"%Y"))
thisYear
customers <- mutate(customers, Age = thisYear - Year_Birth)


#Dropping some redundant features
customers <- select(customers, - ID, - Year_Birth, - Z_CostContact, - Z_Revenue,
    -Dt_Customer)
```

3.4 EDA

Dopo aver eseguito il preprocessing dei dati si è passati ad un'analisi esplorativa dei dati mutando i valori che può assumere una certa variabile tramite la funzione *cut*. Questa operazione è stata eseguita per *Age*, *Income* e Total spent.

```
# Age Range
ageRange <- cut(trainingSet$Age, breaks = c(24, 64, Inf), include.lowest = T,
              ordered_result = T, labels = c("Adult", "Senior"))

trainingSet <- mutate(trainingSet, Age_range = ageRange)

# Income Range
incomeRange <- cut(trainingSet$Income,
                calculateBreaksFromSummary(trainingSet$Income),
```

```
                labels = c("low", "low medium", "medium high", "high"))

trainingSet <- mutate(trainingSet, Income_range = incomeRange)

# Spent Range
spentRange <- cut(trainingSet$Total_spent,
                  calculateBreaksFromSummary(trainingSet$Total_spent),
                  labels = c("low", "low medium", "medium high", "high"))

trainingSet <- mutate(trainingSet, Spent_range = spentRange)
```

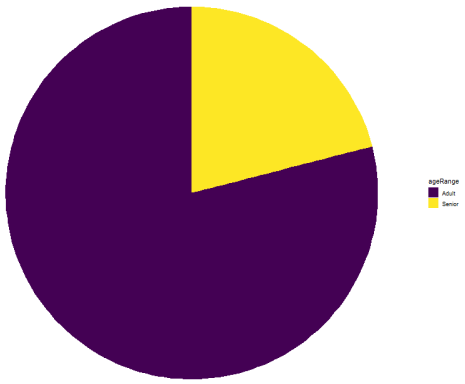


Figura 1: Age pie Chart

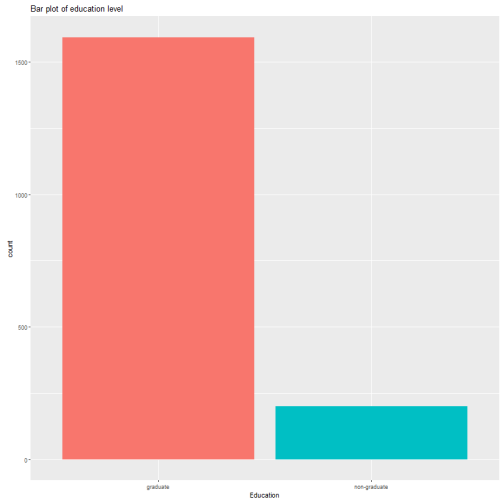


Figura 2: Education bar plot

Il primo grafico a torta è relativo alla variabile Age, il colore viola rappresenta il valore *Adult* mentre il colore giallo *Senior*. Dal grafico si ricava che la maggior parte degli individui è *Adult*. Il secondo rappresenta la variabile *income*, il dataset è equi-distribuito in questo caso. La distribuizione dei valori che assume la variabile *Total spent* è raffigurata nella Figura 3. Da essa si ricava

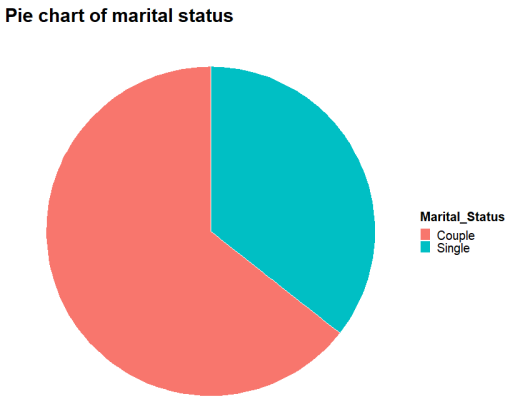


Figura 3: Income pie Chart

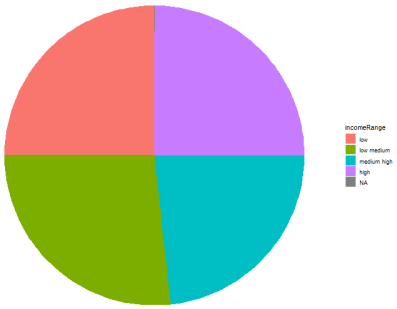


Figura 4: Total_Spent pie Chart

che low e high sono simili mentre quello più frequente è *low medium*.
In seguito si sono seguite delle analisi per le variabili children, total spent e campaign.
La maggior parte delle istanze del dataset ha 1 figlio, la variabile Total_Children è la somma tra la variabile KidHome e TeenHome. Questa informazione si è ricavata eseguendo il codice riportato i seguito.

3.4.1 Total Children

Facendo il bar-plot della variabile *Total_Children* si nota che la maggior parte delle istanze presenti nel dataset ha 1 figlio.

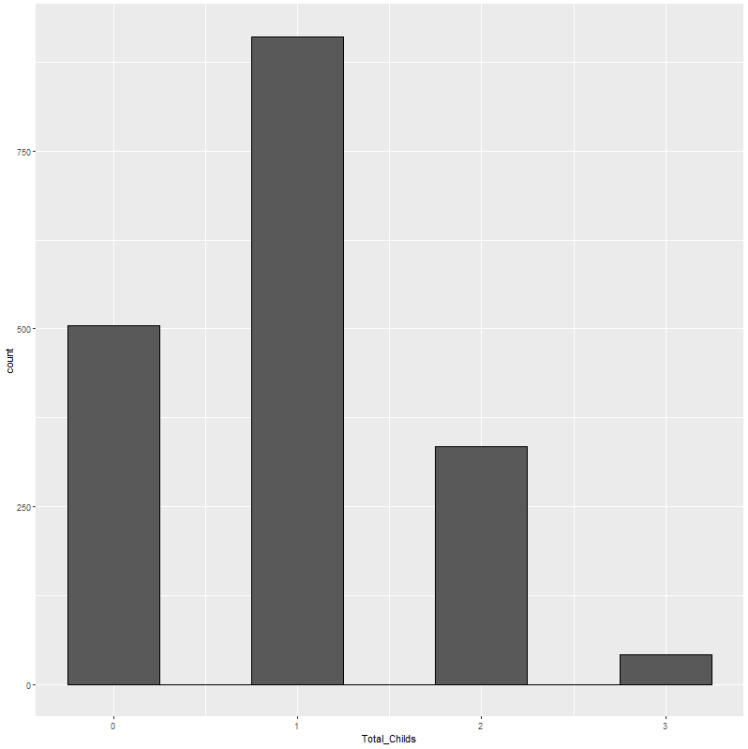


Figura 5: hist plot di total_children

3.4.1.1 Total_Childen e Age

Si è analizzata anche la relazione tra il numero totale di figli, *Total_children* ed *Age* ed è risultato che tra gli *Adult* il numero di figli più frequente è 1 e che rispetto ai *Senior* hanno più figli.

```
age_children_histogram <- ggplot(trainingSet, aes(x=Total_Spent)) +
    geom_histogram(aes(fill=Age_range), binwidth = 0.5, colour = "Black")
age_children_histogram + facet_grid(Age_range~.)
```

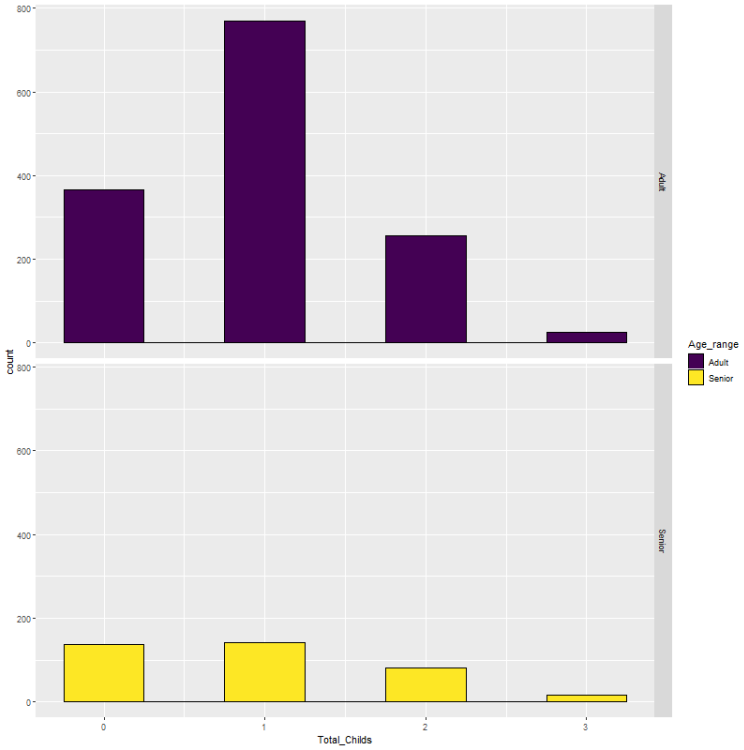


Figura 6: hist plot di Total_Children e Age

3.4.1.2 Total_Children e Marital_Status

Nel pre processing i valori che può assumere la variabile *Marital_Status*sono stati collassati in due possibili valori. *Marital_Status* è 0 se l'istanza ha come valore dell'attributo *Marital_Status* è *single*, *widow* oppure *divorced*. E' uguale a 1 se il valore assunto da *Marital_Status* è *couple* oppure *together*. Rappresentando il diagramma a barre considerando *Total_Children*e *Marital_Status* si ricava che la maggior parte delle istanze ha un figlio. Il numero di istanze con *Marital_Status* pari a 0 è minore rispetto a quelle con valore uguale a 1 per i casi di zero e due figli, mentre per il caso di tre figli sono molto simili.

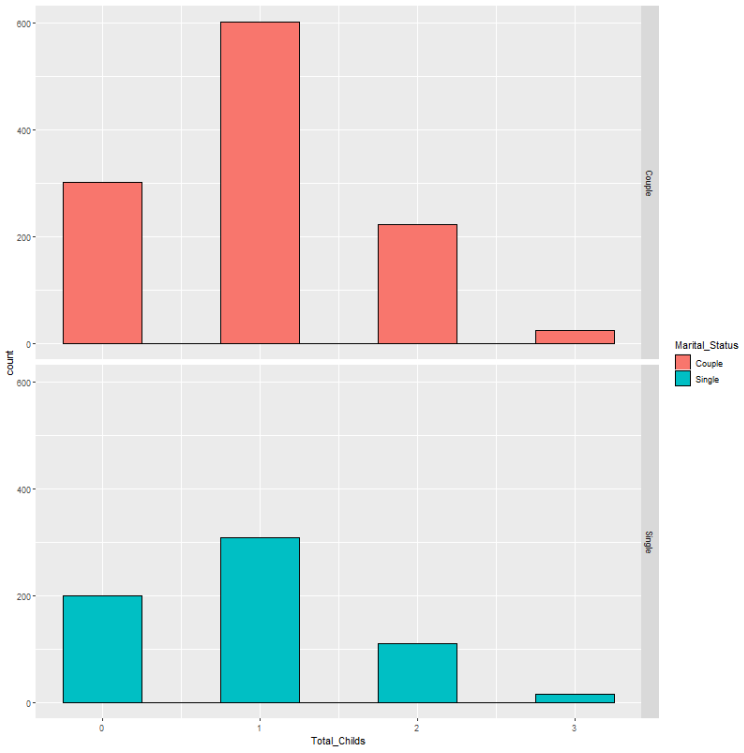


Figura 7: hist plot di Total_Children e Marital_Status

3.4.1.3 Total_Children e Education

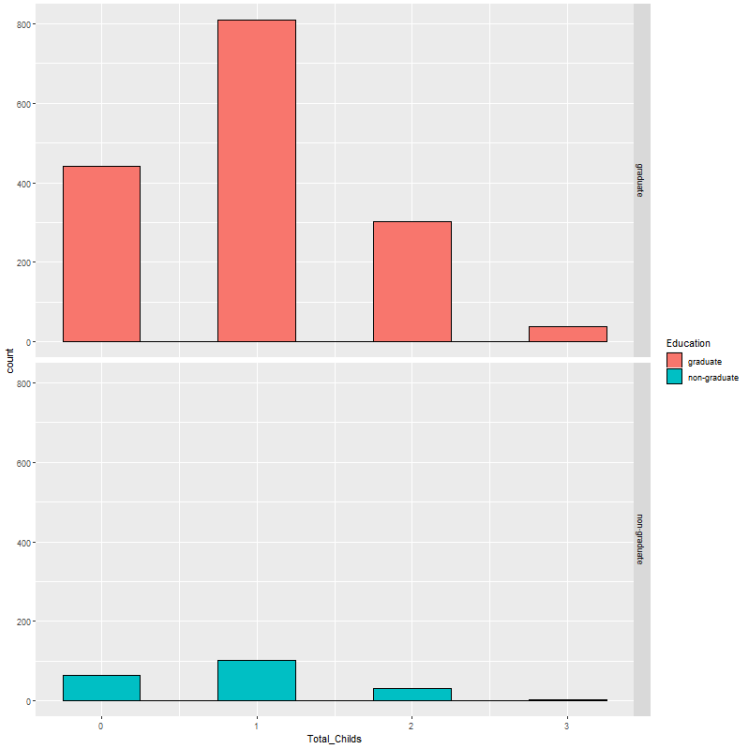


Figura 8: hist plot di Total_Children e Education

3.4.1.4 Total_Children e Income

Sia dall' hist plot che dal jitte rplot si nota che più è basso il guadagno più si hanno figli. Nel caso di due figli ci sono più casi di persone con stipendio *low medium* rispetto a chi ha uno stipendio *low*. Nel caso di istanze con stipendio *high* è comune che si abbiano figli.

3.4.1.5 Total_Children jitter+boxplot

3.4.2 Total Spent

La maggior parte degli individui spende meno di 500$.

Facendo il bar-plot della variabile *Total_Spent* si nota che le istanze presenti nel dataset spendono più per il vino e per la carne.

Per poter produrre il grafico della Figura 12 sono stati sommati i totali che ogni istanza ha speso per un certo prodotto.

3.4.2.1 Total Spent e Age

La maggior parte degli individui spende meno di 500$.

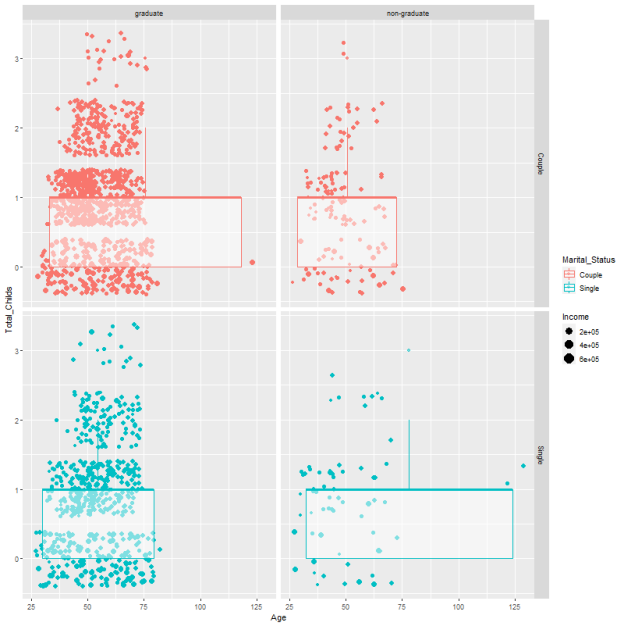


Figura 9: jitter plot di Total_Children e Income

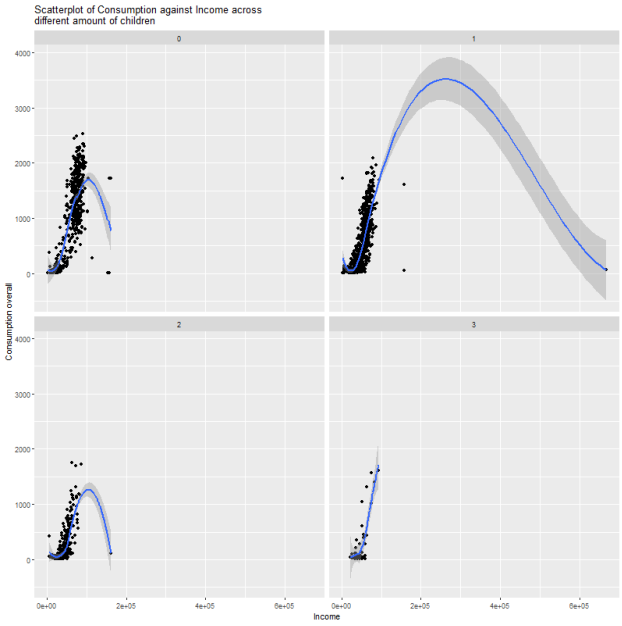


Figura 10: scatter plot , Income e Total_Children

3.4.2.2 Total Spent e Marital_Status

In questo caso sembrano simili. la maggior parte delle coppie spende un totale inferiore a 500. La situazione in single è un po' più rilassata sarà che la maggior parte degli individui nel set di dati sono coppia

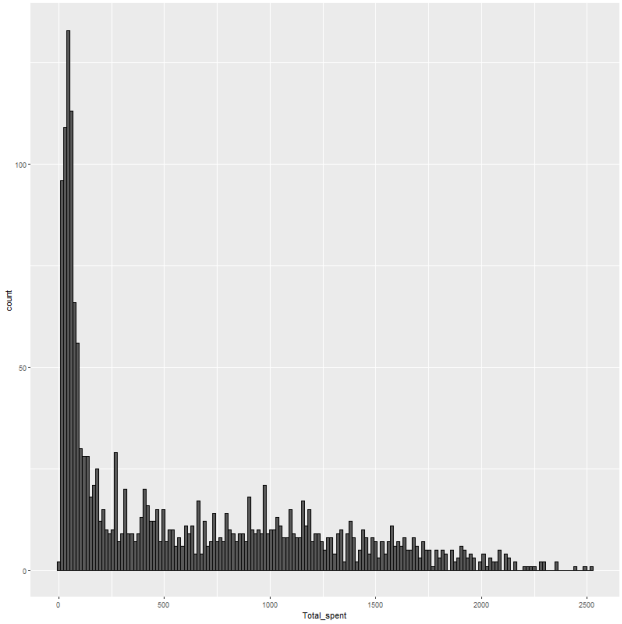


Figura 11: hist plot di total_spent

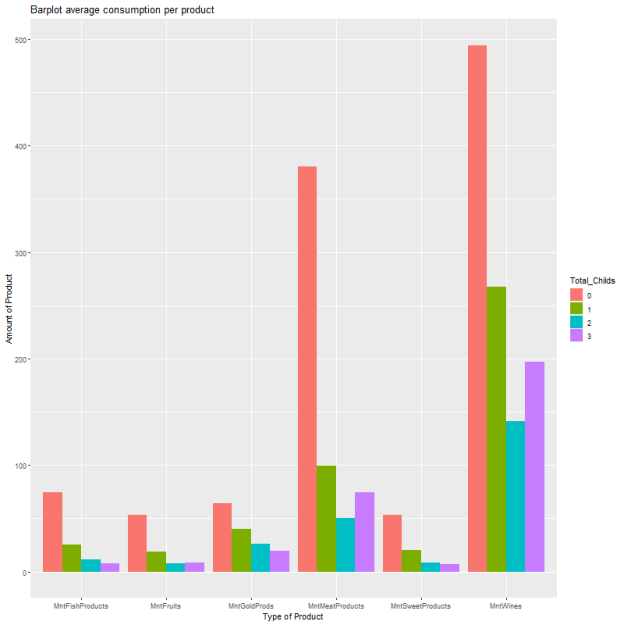


Figura 12: hist plot di total_spent distinguedo tipi di prodotto

3.4.2.3 Total Spent ed Education

I laureati in genere spendono più dei non laureati. la maggior parte dei non laureati spende da 0 a 1500. da 1500 ci sono più casi di laureati che non laureati.

Il grafico della densità è molto simile a quello della variabile *Marital_Status*.

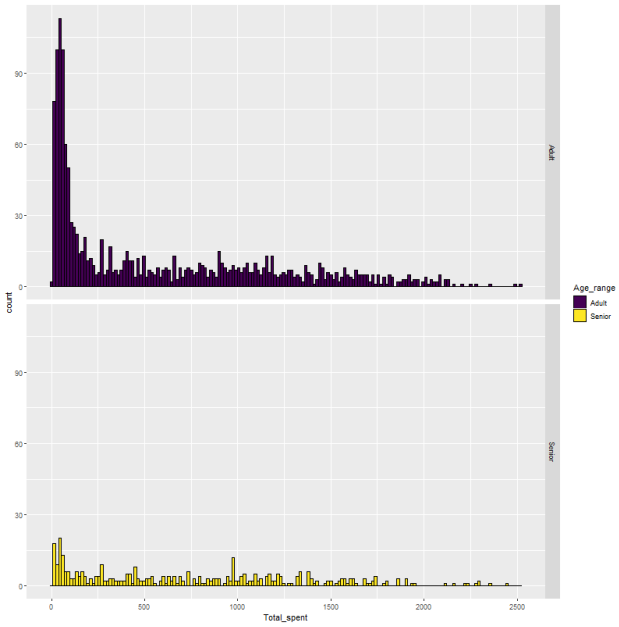


Figura 13: hist plot di Total_Spent e Age

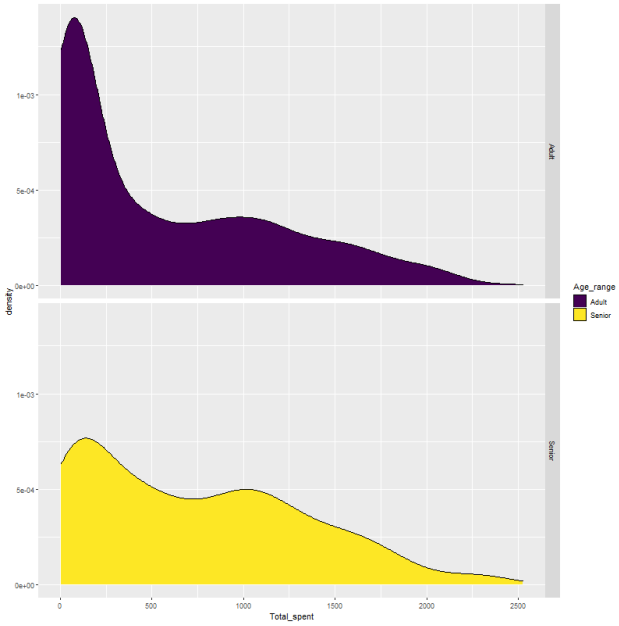


Figura 14: density plot di Total_Spent e Age

3.4.2.4 Total Spent e Total_Children

Di seguito il grafico della densità.

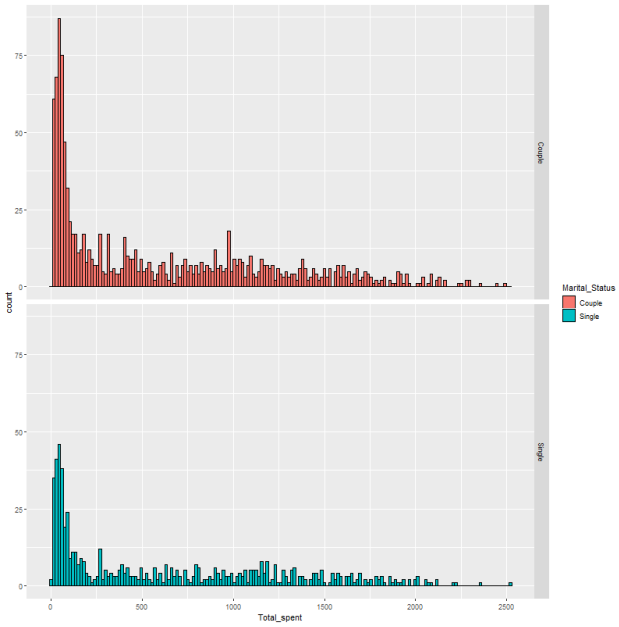


Figura 15: hist plot di Total_Spent e Marital_Status

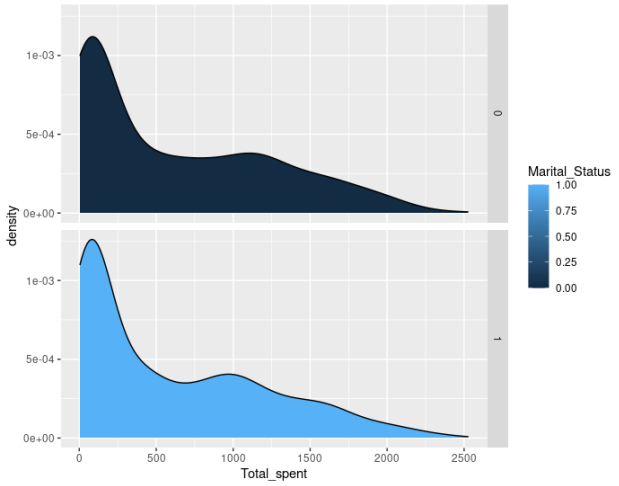


Figura 16: density plot di Total_Spent e Marital_Status

3.4.2.5 Total Spent e Income

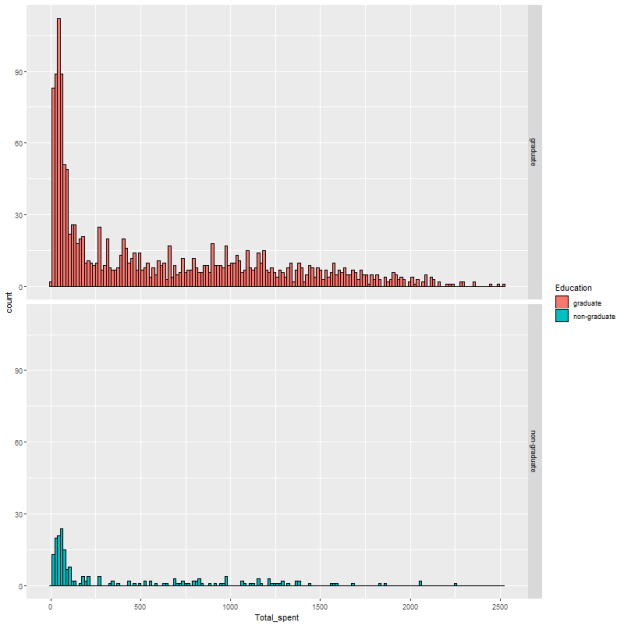


Figura 17: hist plot di Total_Spent e Education

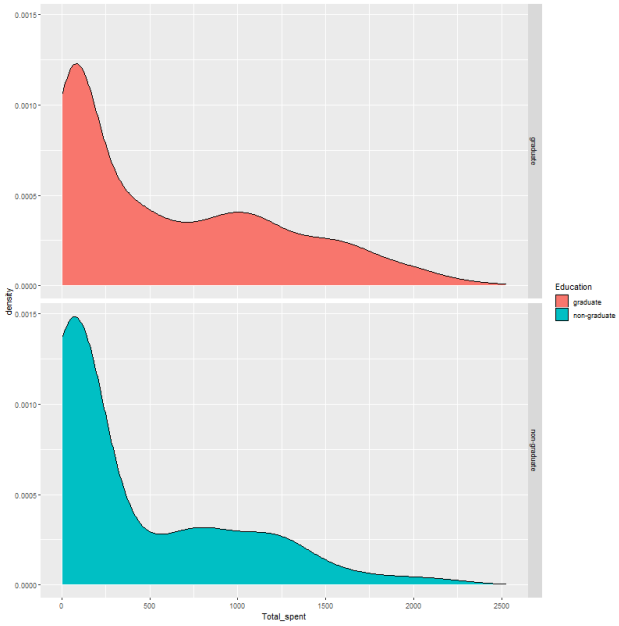


Figura 18: density plot di Total_Spent e Education

3.4.2.6 Total Spent jitter+boxplot

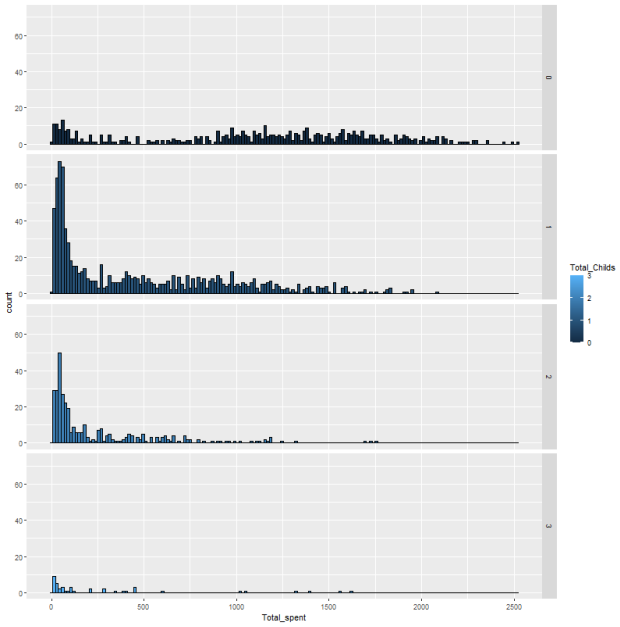


Figura 19: hist plot di Total_Spent e Total_Children

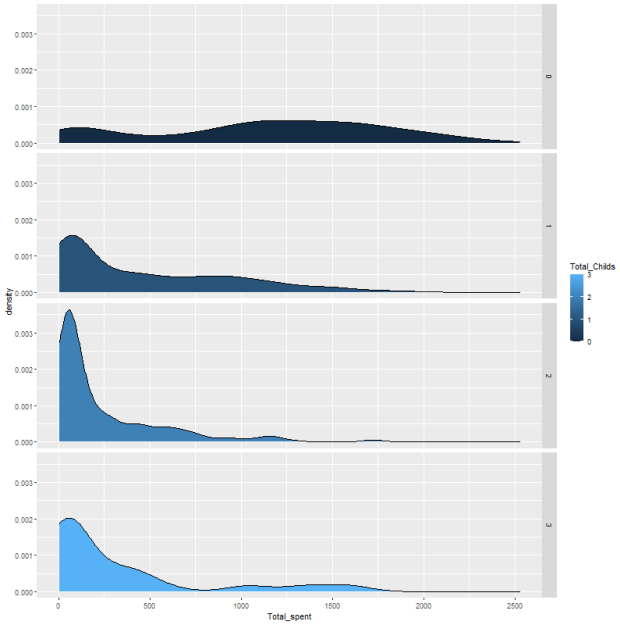


Figura 20: density plot di Total_Spent e Total_Children

3.4.3 Campaign Analysis

Si è eseguita un'analisi anche sulle campagne accettate da ogni individuo.

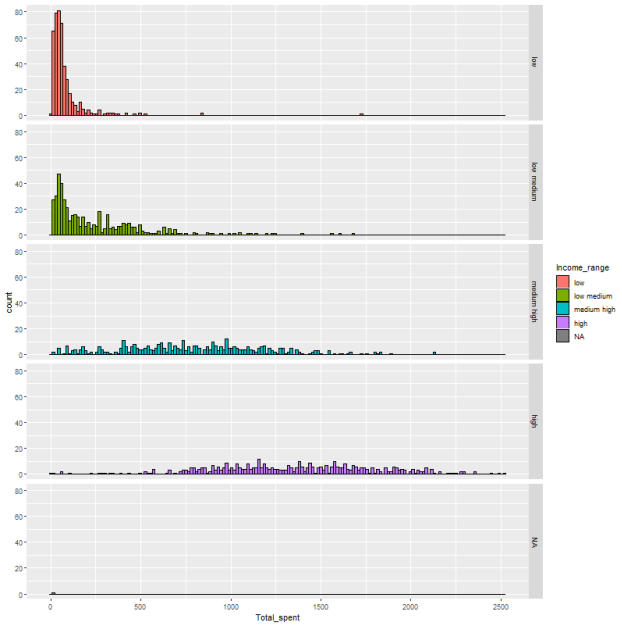


Figura 21: hist plot di Total_Spent e Income

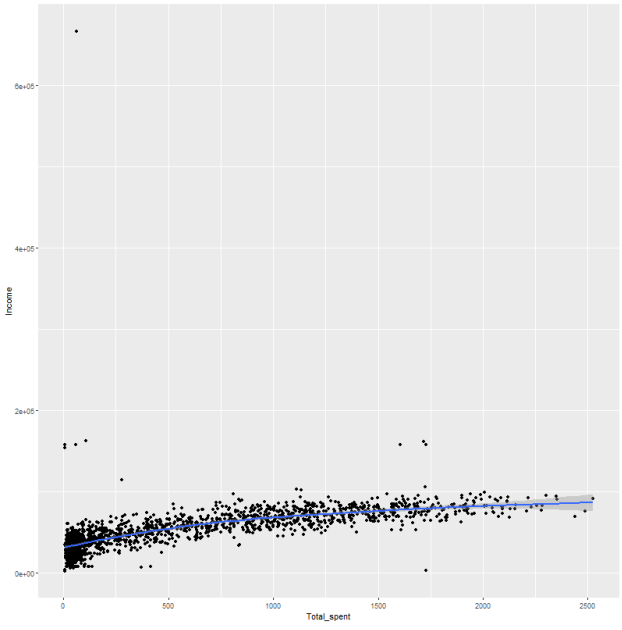


Figura 22: jitter plot di Total_Spent e Income

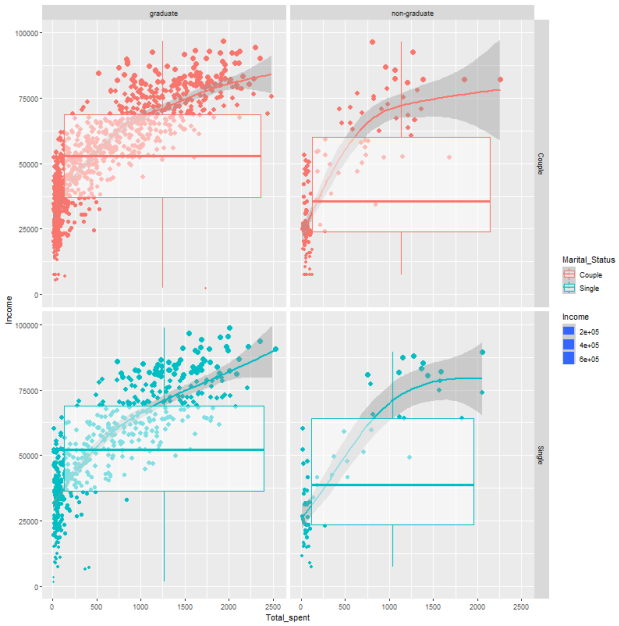


Figura 23: jitter boxplot di Total_Spent, Income e Marital_Status.

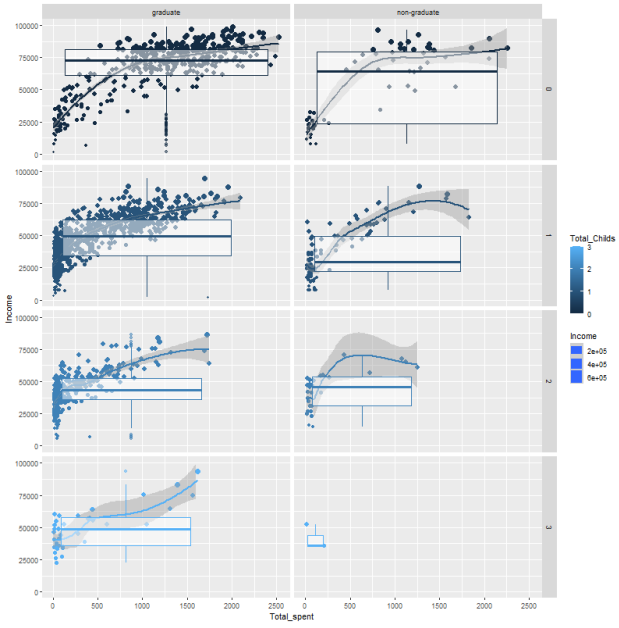


Figura 24: jitter boxplot di Total_Spent, Income e Total_Children.

3.4.3.1 Total_Campaign

3.4.3.2 Total Spent e Age

3.4.3.3 Total Spent e Marital_Status

3.4.3.4 Total Spent ed Education

3.4.3.5 Total Spent e Total_Children

3.4.3.6 Total Spent e Income

3.4.3.7 Total Spent jitter+boxplot

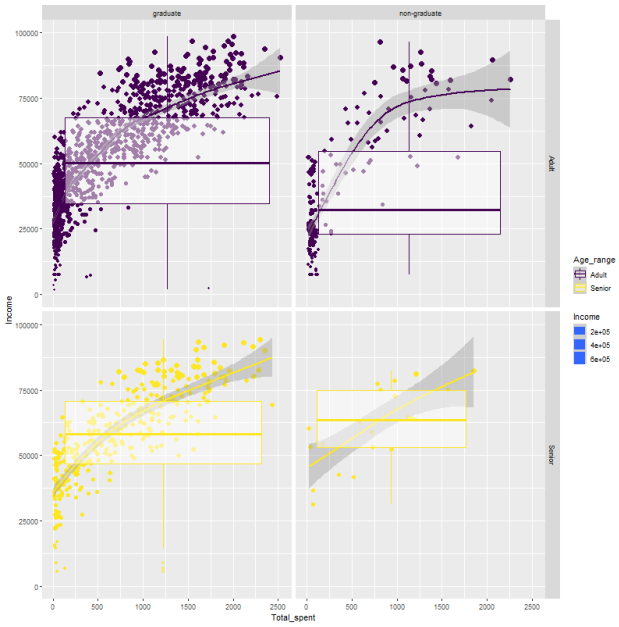


Figura 25: jitter boxplot di Total_Spent, Income ed Age.

3.5 PCA

La PCA è stata prevalentemente sfruttata al fine di ridurre il numero elevato di variabili che descrivono l'insieme di dati a un numero minore di variabili latenti, limitando il più possibile la perdita di informazioni. Il codice seguente mostra parte del codice riportato durante l'analisi dei dati.

```
pca <- PCA(trainingSet_scaled, graph = FALSE)

#Getting the variance of the first 9 new dimensions
pca$eig[,2][1:9]

#Getting the cummulative variance
pca$eig[,3][1:5]

#Getting the most correlated variables
dimdesc(pca, axes = 1:2)

# get eigenvalue
get_eigenvalue(pca)

# visualize pca
fviz_eig(pca, addlabels = TRUE, ylim = c(0, 50))
fviz_contrib(pca, choice = "var", axes = 1, top = 5)
fviz_pca_biplot(pca)

#Creating a factor map for the variable contributions
fviz_pca_var(pca, col.var = "contrib", repel = TRUE)

fviz_pca_var(pca, select.var = list(contrib = 5), col.var = "contrib", repel = TRUE)
```

Da esso si vuole dare particolare attenzione alla funzione *get_eigenvalue(pca)* che fornire le informazioni rappresentate nella tabella 10. Si vuole anche fornire un riferimento grafico a quest'ultima tramite l'output della funzione *fviz_eig(pca, addlabels = TRUE, ylim = c(0, 50))* descritto dalla figura 26. Da essa si può notare che le prime 5 dimensioni fornite dalla PCA forniscono il 70% della varianza cumulativa, per questo motivo si è deciso di prendere in considerazione tali dimensioni. Inoltre si vuole sottolineare l'importanza della prima dimensione che riesce a spiegare più del 40% della varianza dei dati. In particolare la tabella 11 vuole far notare il contributo di ciascun at-

	eigenvalue	variance.percent	cumulative.variance.percent
Dim.1	6.98	41.06	41.06
Dim.2	1.75	10.32	51.38
Dim.3	1.15	6.78	58.16
Dim.4	1.05	6.19	64.35
Dim.5	1.00	5.87	70.22
Dim.6	0.79	4.66	74.88
Dim.7	0.66	3.91	78.79
Dim.8	0.63	3.70	82.49
Dim.9	0.57	3.33	85.82
Dim.10	0.47	2.77	88.59
Dim.11	0.42	2.49	91.08
Dim.12	0.39	2.30	93.38
Dim.13	0.35	2.05	95.43
Dim.14	0.31	1.81	97.24
Dim.15	0.25	1.46	98.69
Dim.16	0.22	1.31	100.00
Dim.17	0.00	0.00	100.00

Tabella 10: Output funzione *get_eigenvalue(pca)*

tributo del dataset nella creazione delle dimensioni della pca. Da essa possiamo notare le cinque principali variabili che hanno contribuito maggiormente nella creazione della prima dimensione della *principal component analysis*: Total_spent, MntMeatProducts, NumCatalogPurchases, MntWines e MntFishProducts. La figura 27 ne mostra un grafico più esplicativo. Le funzioni *fviz_pca_var* hanno permesso di analizzare graficamente le dimensioni che spie

4 Modelli utilizzati

TODO

4.1 K-Means

TODO

5 Esperimenti

TODO

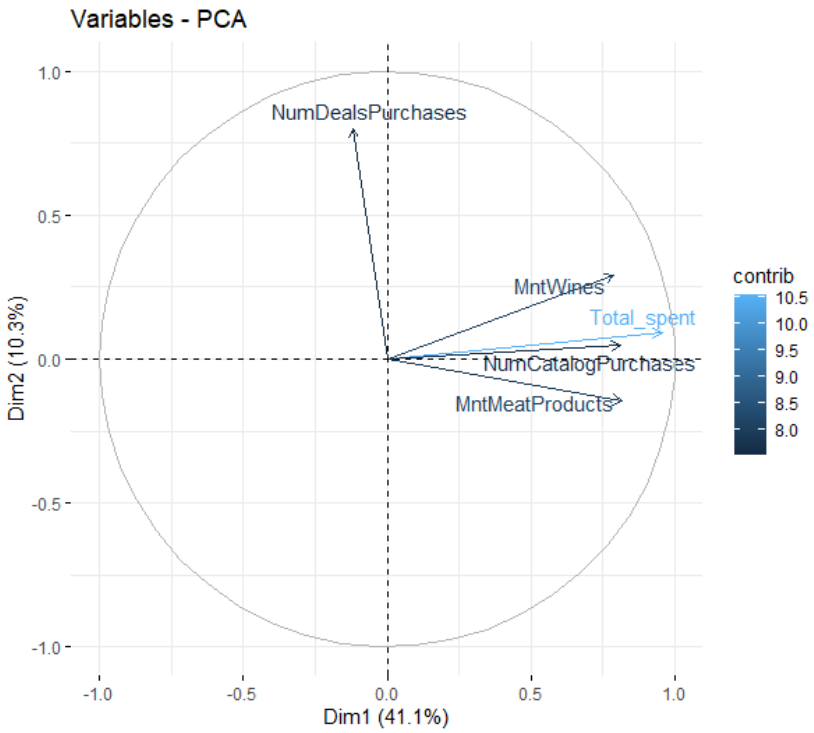


Figura 26: Output funzione *fviz_eig(pca, addlabels = TRUE, ylim = c(0, 50))*

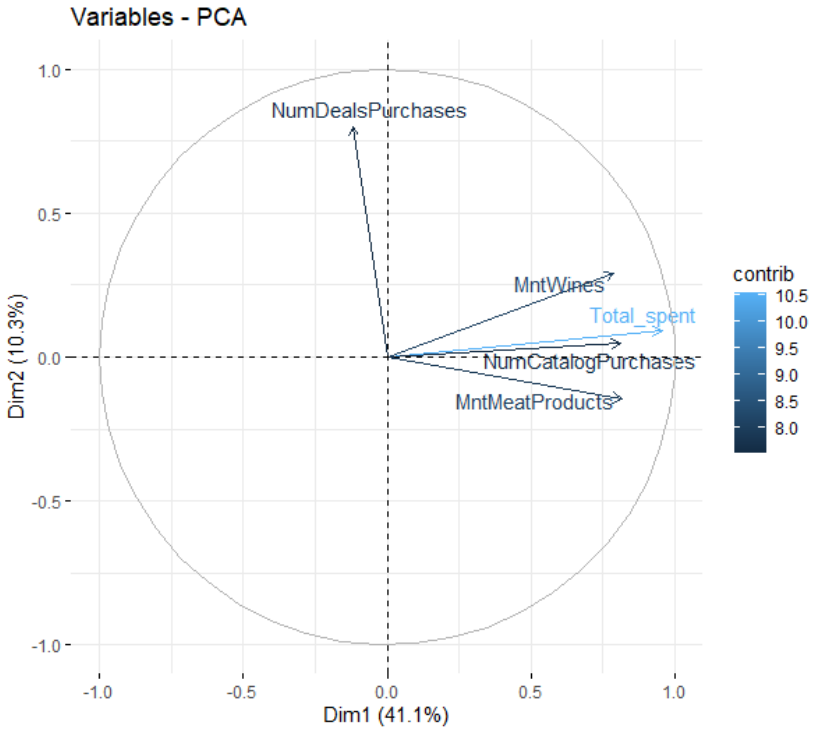


Figura 27: Output funzione *fviz_contrib(pca, choice = "var", axes = 1, top = 5)*

	Dim.1	Dim.2	Dim.3	Dim.4	Dim.5
Income	7.31	0.16	2.59	3.75	0.92
Recency	0.00	0.01	1.24	9.79	87.66
MntWines	8.89	4.77	9.07	1.61	0.88
MntFruits	6.99	1.23	11.85	0.00	0.35
MntMeatProducts	9.55	1.20	0.11	0.00	0.12
MntFishProducts	7.56	1.32	8.51	0.16	0.33
MntSweetProducts	6.94	0.87	9.33	0.06	0.15
MntGoldProds	4.69	2.38	6.52	1.19	0.31
NumDealsPurchases	0.21	36.53	3.87	0.16	0.01
NumWebPurchases	4.30	17.88	0.97	2.21	0.00
NumCatalogPurchases	9.43	0.13	0.70	0.20	0.23
NumStorePurchases	7.53	4.02	0.50	0.32	0.37
NumWebVisitsMonth	5.78	8.73	0.35	12.79	1.36
Total_spent	13.06	0.49	0.82	0.58	0.34
Total_Campains	2.68	0.01	35.64	12.53	3.01
Total_Childs	4.79	14.87	0.20	2.76	0.15
Age	0.28	5.40	7.71	51.88	3.81

Tabella 11: Output *pcavarcontrib*

5.1 Analisi dei risultati ottenuti

6 Conclusion

Appendix: