Koraci za izradu projekta u R-u

Završni ispit: odbrana projekta u R-u

# **Korak 1: Izbor seta podataka za projekat**

|  |  |
| --- | --- |
|  | Za projekat možete koristiti bilo koji set podataka, bilo da je ugrađeni (built-in) R dataset, sa raznih internet izvora (npr. Kaggle) ili je vaš sopstveni set podataka. |

Neki izvori za preuzimanje gotovih setova podataka:

* R setovi podataka po paketima:
  + <https://vincentarelbundock.github.io/Rdatasets/datasets.html>
  + <http://www-eio.upc.edu/~pau/cms/rdata/datasets.html>
* Kaggle setovi podataka:
  + <https://www.kaggle.com/datasets>

# **Korak 2: Analiza seta podataka**

|  |  |
| --- | --- |
|  | U projektu je potrebno ukratko opisati set podataka, navesti link odakle je preuzet, dati opis atributa i ostalo što je relevantno za razumevanje seta podataka i definisanje problema ili pretpostavki (hipoteza). U nastavku se prikazuje primer opisa seta podataka BigMart Sales. |

Prvo poglavlje u projektu je analiza konkretnog seta podataka. Navodi se:

* Kratak opis seta podataka
* Opis atributa
* Struktura podataka, primer prvog head seta...
* Kakva vrsta analize bi mogla da se sprovede, da li je to predviđanje vrednosti nekog atributa ili analiza korelacija između, radi definisanja regresije itd. Navesti problem koji se razmatra.

# **Korak 3: Čišćenje seta podataka**

|  |  |
| --- | --- |
|  | U projektu je potrebno tesitrati set na nedostajuće ili pogrešne vrednosti i ukoliko postoje, primeniti odgovarajuću strategiju čišćenja podataka. |

Ukoliko ste preuzeli set podataka koji je već podeljen na trening i test set, u ovom koraku bi ih trebalo spojiti radi čišćenja i redukcije dimenzionih vrednosti. Npr:

data\_combined <- rbind(BigMartSales\_Train,BigMartSales\_Test)

U poglavlju bi trebalo prikazati da li se u vašem setu podataka nalaze NA vrednosti ili neki drugi netačni podaci. Ukoliko postoje, opisati postupak kako ste to rešili, da li ste ih samo isključili iz dalje analize ili ste ih obrisali ili ste u njih stavili srednju vrednost skupa.

# **Korak 4: Normalizacija i redukcija dimenzionalnosti podataka**

|  |  |
| --- | --- |
|  | U projektu je potrebno analizirati set i ukoliko je potrebno normalizovati ga i primeniti neki od algoritama za redukciju ili smanjenje dimenzionalnosti seta podataka. |

U ovom delu projekta prvo ćete prikazati, uglavnom vizuelno, analizu postojećeg seta podataka. Ova analiza će pomoći u razumevanju prirode podataka, distribuiranosti podataka, korelacija i dr. U daljem tekstu se navodi primer jedne analize. Ukoliko su uočene neke nepravilnosti tada ćete pokušati da popravite, tj. normalizijete podatke. Nakon toga ukoliko bude bilo potrebe primenićete neko od algoritama za smanjenje dimenzionalnosti seta podataka. Na kraju tako pripremljen set podataka, možete podeliti na trening i test set.

Primer:

### **Independent numeric variables**

**First**, we wil find out the number of numerical predictors in our dataset. We will use packages:

library(data.table) # used for reading and manipulation of data

library(dplyr) # used for data manipulation and joining

train\_numeric= dplyr::select\_if(BigMartSales\_Train, is.numeric)

names(train\_numeric)

[1] "Item\_Weight"

[2] "Item\_Visibility"

[3] "Item\_MRP"

[4] "Outlet\_Establishment\_Year"

[5] "Item\_Outlet\_Sales"

plot\_weight=ggplot(data\_combined)+geom\_histogram(aes(Item\_Weight), binwidth =0.5, color="black", fill="lightblue")

plot\_weight

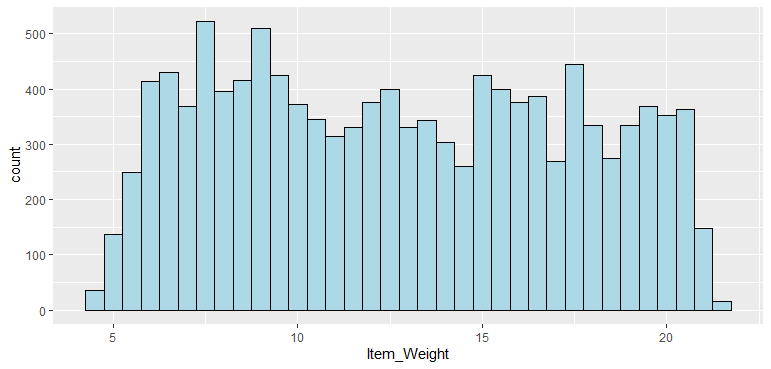


Fig.1.

**Observations**: There seems to be no clear-cut pattern in Item\_Weight.

plot\_visibility <- ggplot(data\_combined) + geom\_histogram(aes(Item\_Visibility), binwidth=0.005, color="black", fill="grey")

plot\_visibility

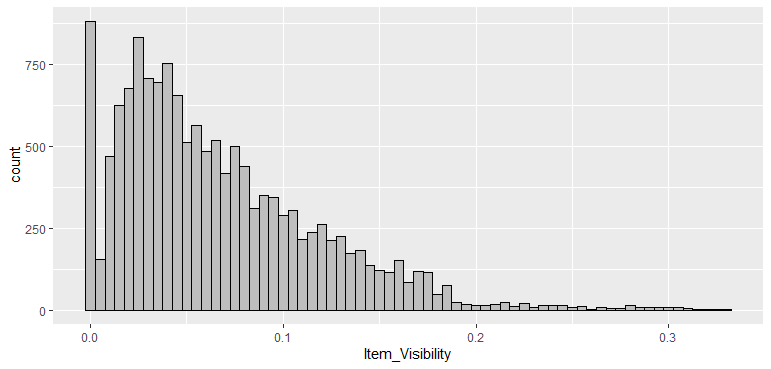


Fig.2.

**Observations**: Item\_Visibility is right-skewed and should be transformed to curb its skewness.

### **Independent categorical variables**

library(dplyr)

plot\_outletSize <- ggplot(data\_combined %>% group\_by(Outlet\_Size) %>% summarise(Count = n())) + geom\_bar(aes(Outlet\_Size, Count), stat = "identity", fill = "coral1") + geom\_label(aes(Outlet\_Size, Count, label = Count), vjust = 0.5, size =2.5) + theme(axis.text.x = element\_text(angle = 45, hjust = 1))

plot\_outletSize

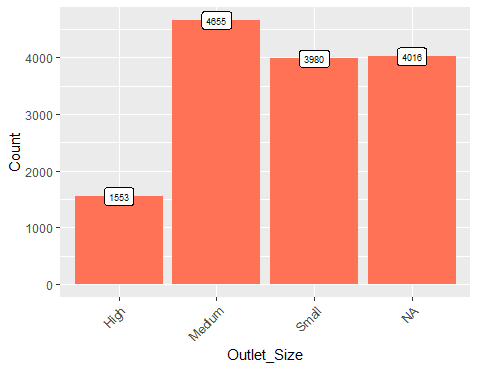


Fig.3.

**Observations:** In Outlet\_Size’s plot, for 4016 observations, Outlet\_Size is blank or missing.

plot\_fatContent <- ggplot(data\_combined %>% group\_by(Item\_Fat\_Content) %>% summarise(Count = n())) + geom\_bar(aes(Item\_Fat\_Content, Count), stat = "identity", fill = "coral1") + geom\_label(aes(Item\_Fat\_Content, Count, label = Count), vjust = 0.5, size = 2.5) + theme(axis.text.x = element\_text(angle = 45, hjust = 1))

plot\_fatContent

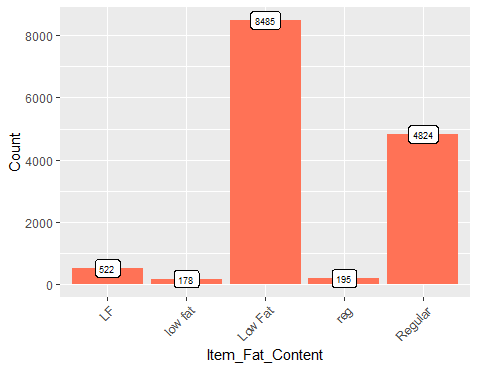


Fig.4.

**Observations:**

* ‘LF’, ‘low fat’, and ‘Low Fat’ are the same category and can be combined into one. Similarly we can combine ‘reg’ and ‘Regular’ into one.
* Some of the items are non-food items, but all the items are categorized either as low fat or regular, which is incorrect. Therefore, we need to assign separate category to non-food items.

## **Multivariate Analysis**

The objective is to discover hidden relationships between the independent variable and the target variable and use those findings in missing data imputation and feature engineering. We will use scatter plots for the continuous or numeric variables and violin plots for the categorical variables.

**Research Question 1 (RQ 1):** Is there any pattern between Item\_Weight and Item\_Outlet\_Sales plot?

plot1 <- ggplot(BigMartSales\_Train) + geom\_point(aes(Item\_Weight, Item\_Outlet\_Sales), colour = "skyblue", alpha = 0.3) + theme(axis.title = element\_text(size = 8.5))

plot1

Warning message:

Removed 1463 rows containing missing values (geom\_point).

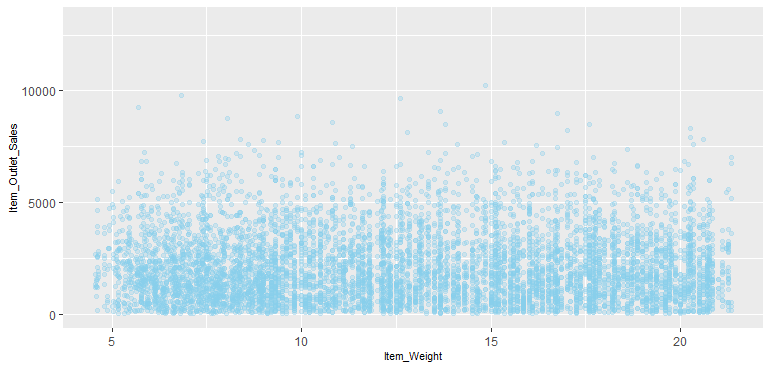


Fig.5.

**Observation**: There is no obvious pattern between Item\_Weight and Item\_Outlet\_Sales plot.

**RQ 2:** Is there any pattern between Item\_Visibility and Item\_Outlet\_Sales plot?

plot2 <- ggplot(BigMartSales\_Train)+geom\_point(aes(Item\_Visibility,Item\_Outlet\_Sales), colour = "skyblue", alpha = 0.3) +theme(axis.title = element\_text(size = 8.5))

plot2

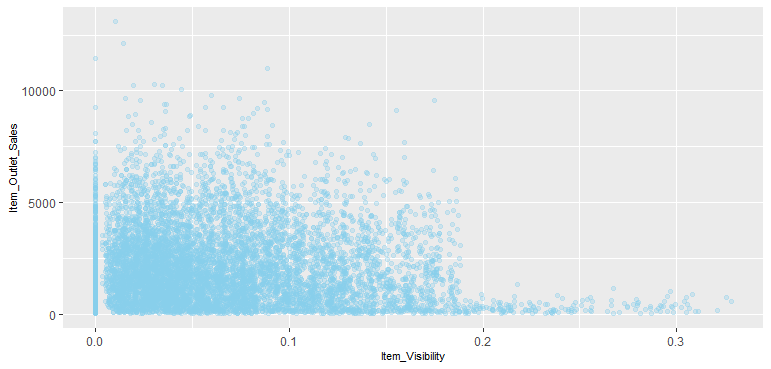


Fig.6.

**Observations**: Item\_Visibility vs Item\_Outlet\_Sales indicates that the more visible a product is the less higher its sales will be. This might be due to the fact that a great number of daily use products, which do not need high visibility, control the top of the sales chart. Furthermore, there is a concerning number of products with visibility zero.

**RQ 3:** Is there any pattern between Item\_MRP (Maximum Retail Price) and Item\_Outlet\_Sales plot?

plot3 <- ggplot(BigMartSales\_Train) + geom\_point(aes(Item\_MRP, Item\_Outlet\_Sales), colour = "skyblue", alpha = 0.3) + theme(axis.title = element\_text(size = 8.5))

plot3

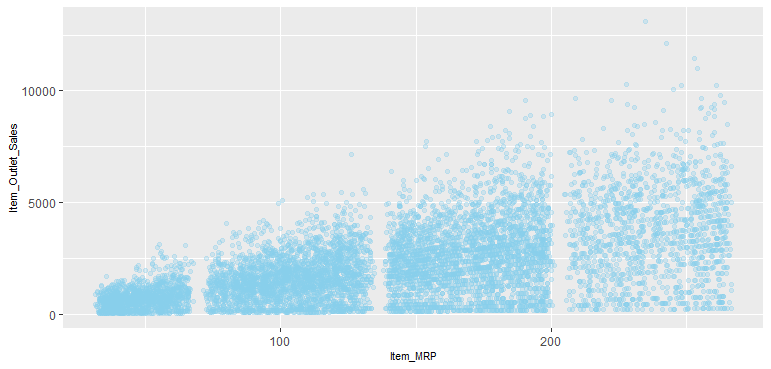


Fig.7.

**Observations**: In the third plot of Item\_MRP vs Item\_Outlet\_Sales, we can clearly see 4 segments of prices that can be used in feature engineering to create a new variable.

**Dealing with categorical variables.** Most of the machine learning algorithms produce better results with numerical variables, therefore, it is essential to treat the categorical variables. One way to do that is to completely remove the categorical variables, but that would be enormous loss of information. We will convert categorical variables into numeric ones using two techniques:

* **Label encoding** – means converting each category of a categorical variable to a number. This is more suitable for ordinal variables – categorical variables with some order.
* **One hot encoding** – means that each category of a categorical variable is converted into a new binary column (1/0).

### **Normalization**

Now, we will scale our numerical predictors. We use Z-score normalization[[1]](#footnote-1).

data\_combined$Item\_Weight <- scale(data\_combined$Item\_Weight,center= TRUE, scale=TRUE)

data\_combined$Item\_Visibility <- scale(data\_combined$Item\_Visibility, center= TRUE, scale=TRUE)

data\_combined$Item\_MRP <- scale(data\_combined$Item\_MRP, center= TRUE, scale=TRUE)

data\_combined$Outlet\_Age <- scale(data\_combined$Outlet\_Age, center= TRUE, scale=TRUE)

str(data\_combined)

### **Splitting the dataset into training and test**

Now, we split the data\_combined back into train and test data for building our model[[2]](#footnote-2).

smp\_size=floor(0.60\*nrow(data\_combined))

train\_ind=sample(seq\_len(nrow(data\_combined)),size=smp\_size)

train\_set=data\_combined[train\_ind,]

test\_set=data\_combined[-train\_ind,]

dim(train\_set)

[1] 8522 27

dim(test\_set)

[1] 5682 27

# removing Item\_Outlet\_Sales from test set

test\_set$Item\_Outlet\_Sales=NULL

### **Correlated Variables**

We will check for the correlation among the variables to decide whether we need to reduce dimensions of our dataset or not. It is not desirable to have correlated features if we are using linear regressions.

# select numeric variables

train\_num=train\_set%>%select\_if(is.numeric)

corMatrix=cor(train\_num)

corrplot(corMatrix,order = "FPC",method = "color",type = "lower", tl.cex = 0.6, tl.col = "black")

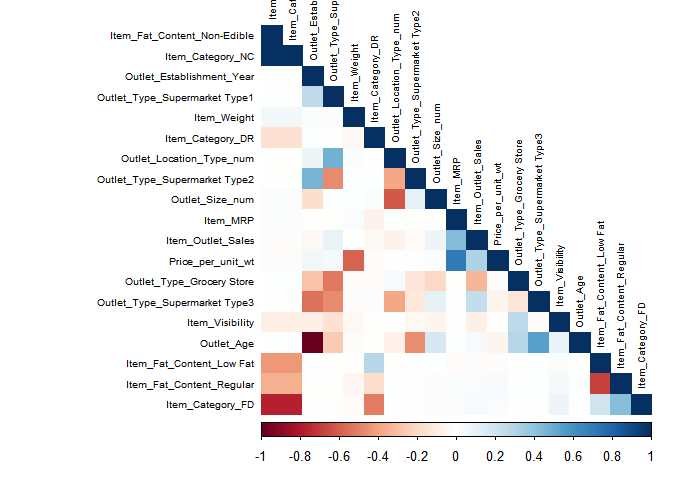


Fig. 8.

Positive correlations are displayed in blue and negative correlations in red color. Since, there is insignificant correlation (less large intensity color blocks) among the variables, we don’t need to perform principal component analysis on our dataset.

# **Korak 5: Primena algoritma za mašinsko učenje**

|  |  |
| --- | --- |
|  | U ovom delu ćete primeniti jedan, a možete i više algoritama za mašinsko učenje. |

Pored primene algoritma, objasnićete do kojih rezultata ste došli.

# **Korak 6: Testiranje**

|  |  |
| --- | --- |
|  | Na osnovu prethodno definisanog modela, a do kog ste došli primenom odgovarajućeg algoritma za mašinsko učenje, primenićete ga na test podatke i objasniti rezultate do kojih ste došli. |

U ovom delu se primenjuje podešen model do kog ste došli u prethodnom koraku, na test podatke i videti npr. koliki je procenat tačnosti dobijenih rezultata.

# **Korak 7: Zaključna razmatranja**

|  |  |
| --- | --- |
|  | Dobijene rezultate ćete ukratko opisati u zaključku rada. |

1. Read more: <https://www.codecademy.com/articles/normalization>. [↑](#footnote-ref-1)
2. Splitting applied from this source: <https://rpubs.com/ID_Tech/S1>. [↑](#footnote-ref-2)