



## Case Study 1 Data Preprocessing

### 1. Bank Data

The Bank Dataset contains 11 independent variables specifically age, region, income, sex, married, children, car, save\_act, current\_act, and mortgage and one response variable which answers the question: “Did the customer buy a PEP (Personal Equity Plan) after the last mailing?” with a yes/no response. We will analyze this data beforehand using descriptive analytics and preprocess the data for use in various data mining algorithms.

### 2. Generating Descriptive Analytics

#### 2.1. Initialize R: Setting the Working Directory

The working directory is the main directory in which R does analysis. Usually before starting any analysis with R, we set the working directory to a folder where all the data is stored.

2.1.1. Open R Studio

2.1.2. On the file explorer tab click on Files.



2.1.3. Click on Explore

2.1.4. Go to the Desktop Folder -> Module 3 Datasets

2.1.5. Click on More. . Click on Set as Working Directory.

#### 2.2. Load the Bank Dataset into R.

2.2.1. Click on File-> New File -> R Script.

2.2.2. In the new tab script , type the following code:

```
• bankdata = read.csv("bankdata.csv")
```

2.2.3. Put the cursor at the end of the code and click on Run . As a result, the data is loaded in the Environment

#### 2.3. Descriptive Analytics and Visualization

We want to analyze the all attribute columns in terms of Mean, Standard Deviation, Median, Mode, Variance, Range, Minimum, Maximum, Sum and Count.

2.3.1. To calculate for the descriptive statistics, type the following lines of code.

```
• library(pastecs)  
• options(scipen=100, digits=2)  
• write.csv(stat.desc(bankdata), file = "NumericalStatistics.csv")  
• write.csv(summary(bankdata), file = "CategoricalStatistics.csv")
```

2.3.2. Highlight all lines that were typed in step 2.3.1 and click on Run . As a result, the descriptive statistics results are saved in the Desktop -> Module 3 Datasets -> Case 1 Folder.

2.3.3. Answer the following questions:


2.3.3.1. What is the range of values of the Age variable? What is the minimum, maximum and middle value?



2.3.3.2. How many customers have a savings account? Current account?

2.3.4. Let say we want a CrossTab Report for the relationship of the variable “Married” and the Number of Children. Type the following line of code.


- `xtabs(~married+children,data=bankdata)`

2.3.5. Highlight the line that was typed in step 2.3.4 and click on Run  Run. Verify the result as follows:

	children			
married	0	1	2	3
NO	83	46	50	25
YES	180	89	84	43

2.3.6. Now, we would like to calculate the means of Age, Income and Children by PEP, Married and has Car. To do this, type the following lines of code

- `library(reshape)`
- `bankdata.m = melt(bankdata, id=c("pep", "married", "car"), measure=c("age", "income", "children"))`
- `bankdata.c = cast(bankdata.m, pep + married + car ~ variable, mean)`
- `write.csv(bankdata.c , file = "bankdataByPepStatusCar.csv")`

2.3.7. Highlight all lines that were typed in step 2.3.6 and click on Run  Run. As a result, the pivot analysis results are saved in the Desktop -> Module 3 Datasets -> Case 1 Folder. Verify the result by opening the file bankdataByPepStatuscar.csv.

	pep	married	car	age	income	children
1	NO	NO	NO	36.54762	21758.49	1.5
2	NO	NO	YES	39.2619	23635.47	1.5
3	NO	YES	NO	40.12698	25031.17	0.833333
4	NO	YES	YES	41.65517	26355.49	1.008621
5	YES	NO	NO	44.38333	30565.43	0.8
6	YES	NO	YES	45.98333	31752.53	0.783333
7	YES	YES	NO	43.39474	28293.14	1.052632
8	YES	YES	YES	46.73077	32145.53	1.076923

2.3.7.1. As Age increases, what pattern do you see in terms of buying a PEP?

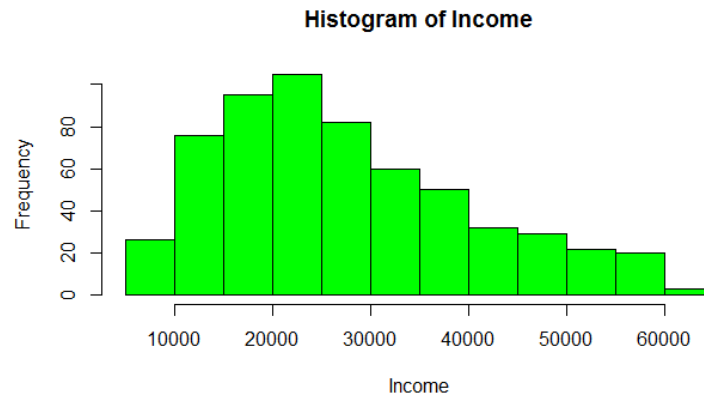
2.3.7.2. In terms of the number of children what pattern do you see in terms of buying a PEP? For Being Married?

2.3.8. Now, we would like to calculate for a histogram of the Income variable. Type the following code:

- `hist(bankdata$income,breaks=15, col="green",xlab="Income",main="Histogram of Income")`



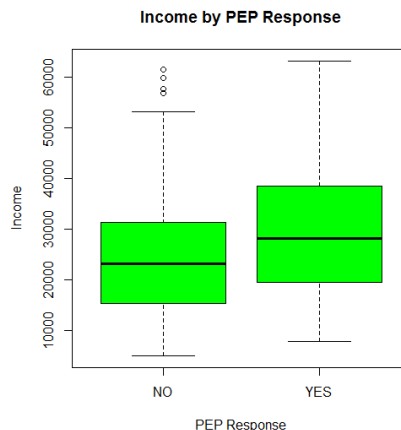
2.3.9. Highlight the line that was typed in step 2.3.8 and click on Run . Click on Zoom to view the result. The histogram should look like this:



2.3.10. We would like to calculate for a box plot of the Income variable by PEP. Type the following code:

- ```
boxplot(income~pep,data=bankdata, main="Income by PEP Response",  
xlab="PEP Response", ylab="Income", col="green")
```

2.3.11. Highlight the line that was typed in step 2.3.10 and click on Run . Click on Zoom to view the result. The box plot should look like this:



2.3.11.1. What can you generalize from this Box Plot?

---

2.3.11.2. Are there any outliers?

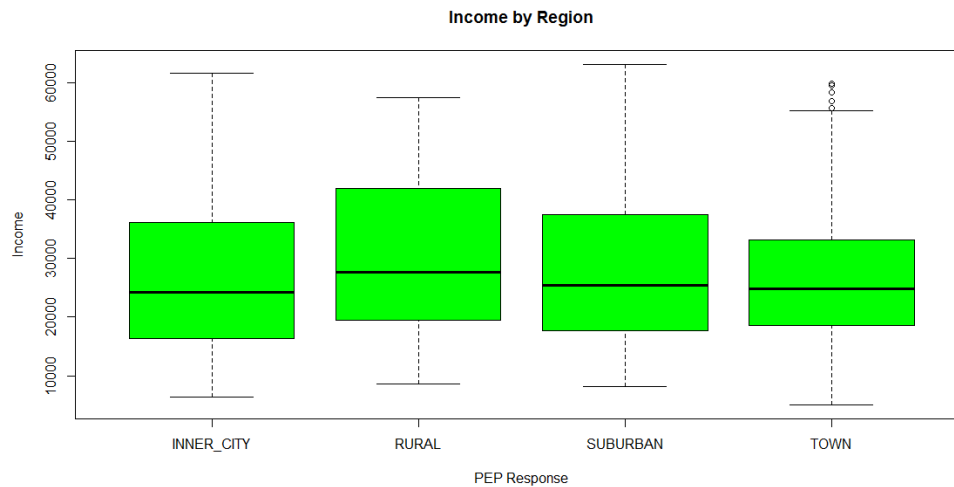
---

2.3.12. We would like to generate a box plot of the Income variable by region. Type the following code:

- ```
boxplot(income~region,data=bankdata, main="Income by Region",  
xlab="Region", ylab="Income", col="green")
```



2.3.13. Highlight the line that where typed in step 2.3.12 and click on Run . Click on Zoom to view the result. The box plot should look like this:

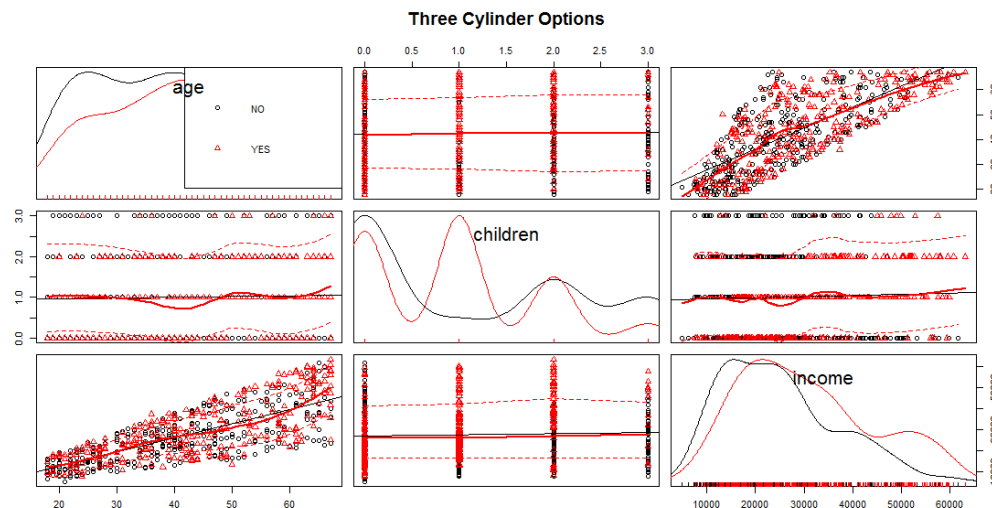


2.3.13.1. What can you generalize from this Box Plot?

2.3.14. To plot a scatter plot matrix of the Income, Age and # of Children, type the following code:

- `library(car)`
- `scatterplotMatrix(~age+children+income|pep, data=bankdata, main="Age Children and Income by PEP")`

2.3.15. Highlight the line that where typed in step 2.3.14 and click on Run . Click on Zoom to view the result. The box plot should look like this:






### 2.3.15.1. What can you generalize from this Scatter Plot?

## 2.4. Data Transformation

We want to transform certain variables into a different format as an input to the various data mining methodologies.


2.4.1. To Normalize the Income Column into a [0,1] scale, type the following code:

- `IncomeData = bankdata[,5]`
- `NormalizedIncomeData = (IncomeData - min(IncomeData)) / (max(IncomeData) - min(IncomeData))`
- `bankdata = cbind(bankdata, NormalizedIncomeData)`
- `View(bankdata)`

2.4.2. Highlight the line that where typed in step 2.4.1 and click on Run . The result of the code from step 2.4.1 is the same bankdata but with a new column NormalizedIncomeData at the End.


2.4.3. Suppose that we want to create an equal depth(frequency) variable for Income where the new variable could take in "Low", "Medium" and "High." Type the following code:

- `bins=3`
- `cutpoints=quantile(IncomeData, (0:bins)/bins)`
- `DiscreteIncome = cut(IncomeData, cutpoints, include.lowest=TRUE, dig.lab=5, labels=c("Low", "Med", "High"))`
- `bankdata = cbind(bankdata, DiscreteIncome)`
- `View(bankdata)`

2.4.4. Highlight the line that where typed in step 2.4.3 and click on Run . The result of the code from step 2.4.3 is the same bankdata but with a new column DiscreteIncome at the end representing the discretized Income.

2.4.5. Suppose that we want to create dummy variables for the four values of Region. Type the following code:

- `indicators=model.matrix(~ region - 1, data = bankdata)`
- `bankdata = cbind(bankdata, indicators)`
- `View(bankdata)`

2.4.6. Highlight the line that where typed in step 2.4.5 and click on Run . The result of the code from step 2.4.5 is the same bankdata but with four new columns specifically regionINNER\_CITY, regionRURAL, regionSUBURBAN, regionTOWN.

## 2.5. Data Sampling

2.5.1. To sample 100 rows of the bank data without replacement type the following:

- `Samplebankdata = bankdata[sample(nrow(bankdata), 100, replace = FALSE), ]`
- `View(Samplebankdata)`

2.5.2. The result is a subset sample of the bankdata dataset.