# Logistic Regression report

```
library(dplyr)
library(Amelia)
library(ggplot2)
library(forcats)
library(caTools)
library(caret)
library(Metrics)

# load data
df <- read.csv("adult_sal.csv")
head(df)

# Drop column X
df <- select(df,-X)

str(df)
summary(df)
```

## Data Cleaning

```
# Data Cleaning
table(df$type_employer)
any(is.na(df$type_employer))

# Drop "?" 1770 value
# df <- df[!df$type_employer == "?","type_employer"]
# Never-worked and Without-pay
count(df[df$type_employer == "?",])
# Combine Never-worked and Without-pay called "Unemployed"
"Unemployed" -> df[df$type_employer %in% c("Without-pay","Never-worked"),"type_employer"]

# Combine State-gov and Local-gov called "SL-gov"
"SL-gov" -> df[df$type_employer %in% c("State-gov","Local-gov"),"type_employer"]

# Combine Self-emp-inc and Self-emp-not-inc called "self-emp"
"self-emp" -> df[df$type_employer %in% c("Self-emp-inc","Self-emp-not-inc"),"type_employer"]

adult$type_employer <- sapply(adult$type_employer,unemp)
table(adult$marital)

group_emp <- function(job){
    if (job=='Local-gov' | job=='State-gov'){
        return('SL-gov')
    }else if (job=='Self-emp-inc' | job=='Self-emp-not-inc'){
        return('self-emp')
    }else{
        return(job)
    }
}
```

```
adult$type_employer <- sapply(adult$type_employer,group_emp)
table(adult$type_employer)


# Marital Column
table(df$marital)

# Reduce this to three groups
"Married" -> df[df$marital %in% c("Married-AF-spouse","Married-civ-spouse","Married-spouse-absent"),"marital"]
"Not-Married" -> df[df$marital %in% c("Separated","Widowed","Divorced"),"marital"]

adult$marital <- sapply(adult$marital,group_marital)
table(adult$marital)


# Country Column
table(df$country)

as <- c('China','Hong','India','Iran','Cambodia','Japan','Laos',
       'Philippines' ,'Vietnam' ,'Taiwan', 'Thailand')
naca <- c('Canada','United-States','Puerto-Rico' )
eu <- c('England' ,'France', 'Germany' ,'Greece','Holand-Netherlands','Hungary',
        'Ireland','Italy','Poland','Portugal','Scotland','Yugoslavia')
saca <- c('Columbia','Cuba','Dominican-Republic','Ecuador',
          'El-Salvador','Guatemala','Haiti','Honduras',
          'Mexico','Nicaragua','Outlying-US(Guam-USVI-etc)','Peru',
          'Jamaica','Trinadad&Tobago')
oth <- c('South','?')

"Asia" -> df[df$country %in% as,"country"]
"North.America" -> df[df$country %in% naca,"country"]
"Europe" -> df[df$country %in% eu,"country"]
"Latin.and.South.America" -> df[df$country %in% saca,"country"]
"Other" -> df[df$country %in% oth,"country"]
```

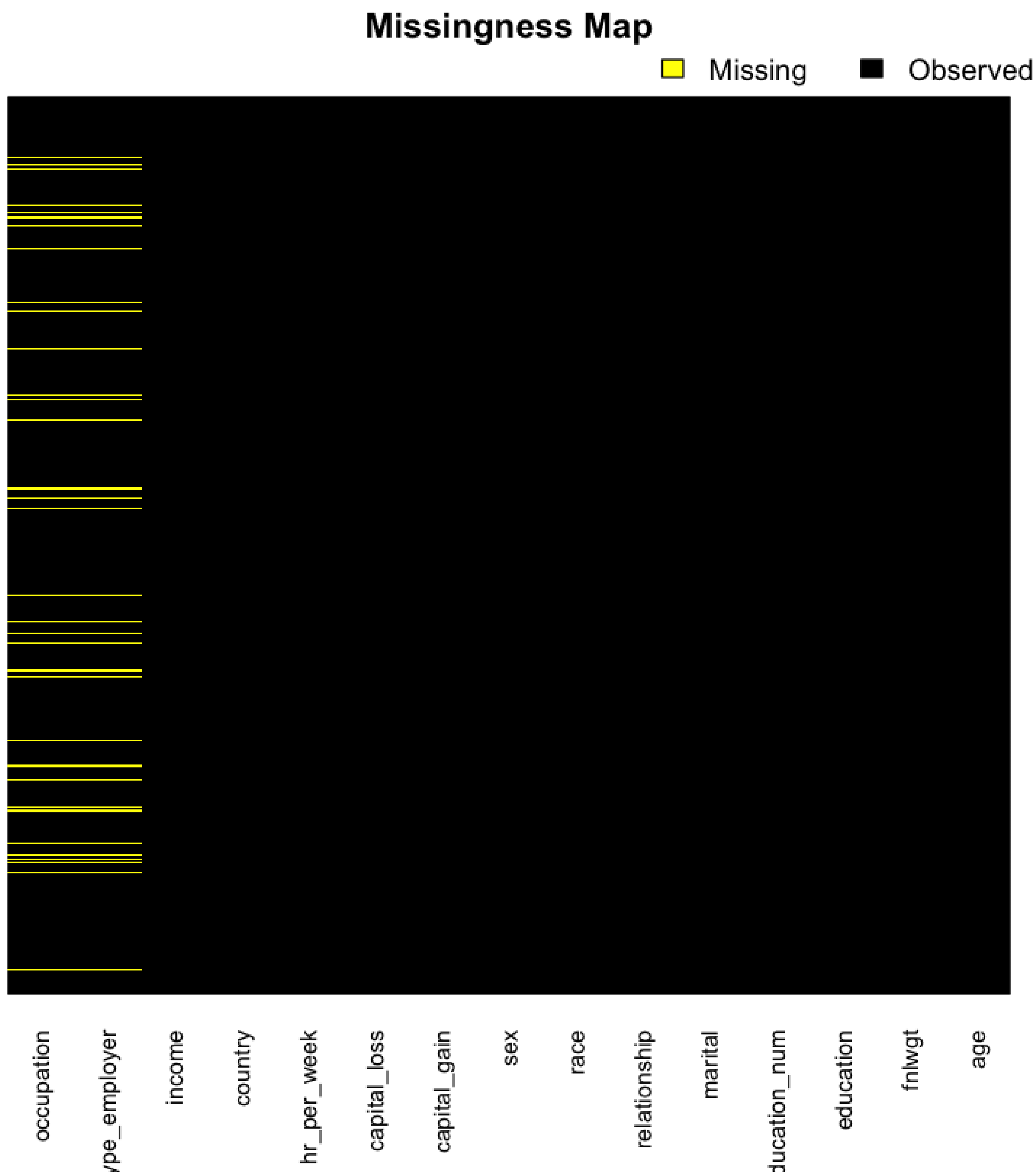
→ ตรวจสอบให้แน่ใจว่าคอลัมน์ใด ๆ ที่เราเปลี่ยนแปลงมีระดับปัจจัยด้วย factor()

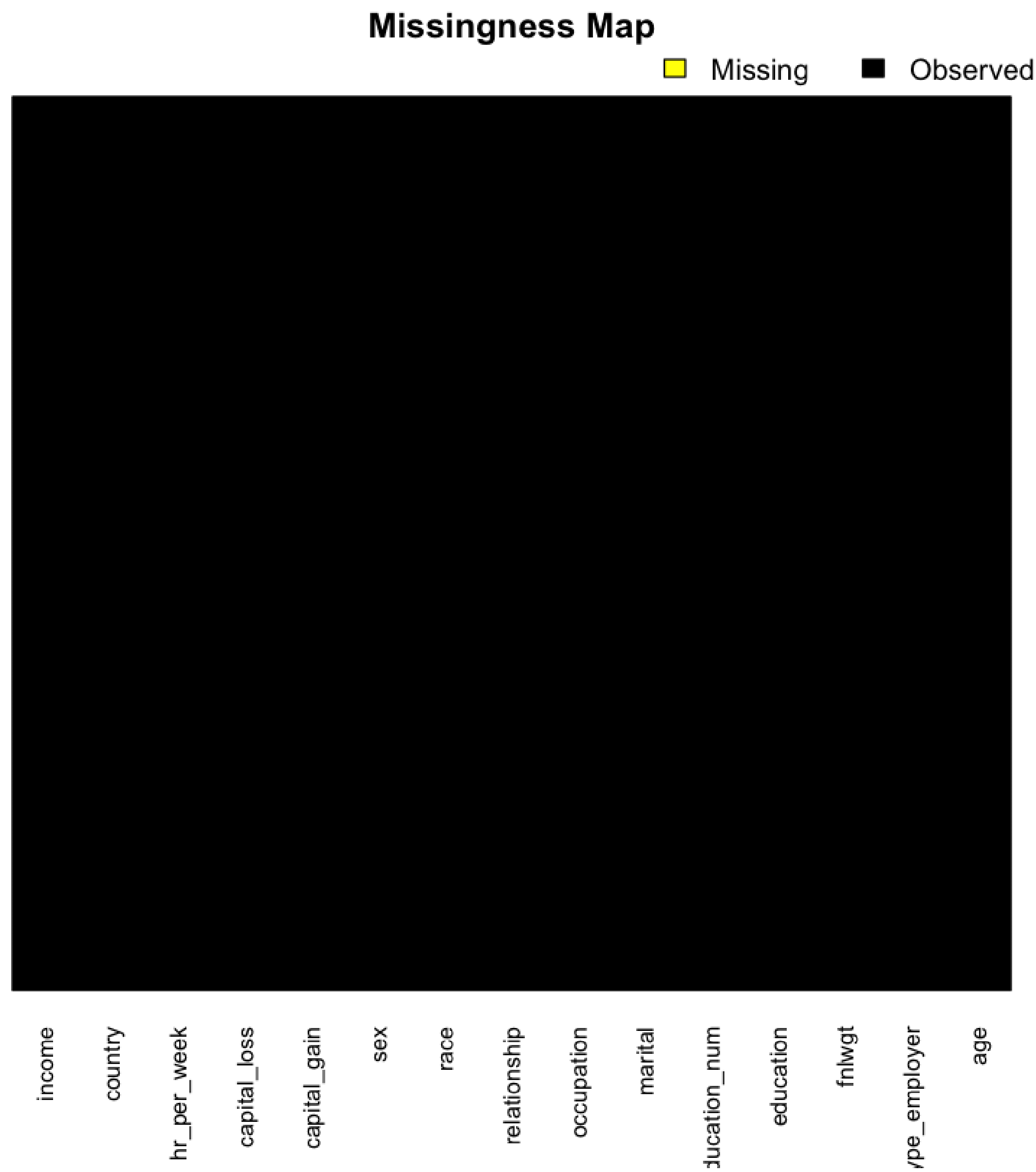
```
# convert "?" to NA value
NA -> df[df$type_employer == "?","type_employer"]
NA -> df[df$occupation == "?","occupation"]
table(df$type_employer)
table(df$occupation)

# drop levels factor
df <- droplevels(df)
str(df)

# change character to factor levels
# df$type_employer <- factor(df$type_employer)
# df$marital <- factor(df$marital)
factor_df <- function(dt){
  out <- dt
  for (i in 1:length(dt)) {

    if (class(dt[[i]]) == "character"){
      out[[i]] <- factor(dt[[i]])
    }

    else{
      out[[i]] <- dt[[i]]
    }
```

```
  }
  return(out)
}
df <- factor_df(df)
str(df)
```

## Missing Data

```
library(Amelia)
# missmap check NA
missmap(df, main="Missings Map",
        col=c("yellow", "black"),y.at=c(1),y.labels = c(''))

# drop NA
df <- na.omit(df)
any(is.na(df))
```


Missingness Map
Missing
Observed
occupation
ype_employer
income
country
hr_per_week
capital_loss
capital_gain
sex
race
relationship
marital
ducation_num
education
fnlwgt
age

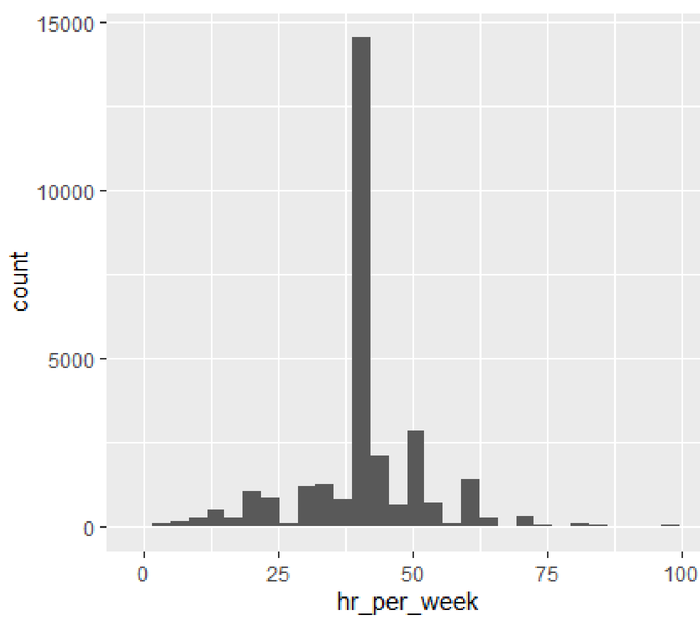


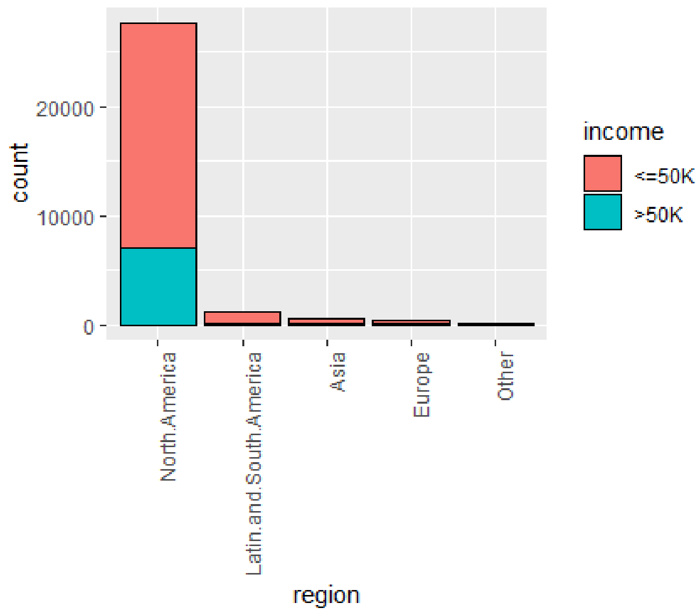

## EDA → Exploratory Data Analysis

→ explored it using visualization

```
library(ggplot2)
library(dplyr)

# EDA
str(df)
# create a histogram of ages, colored by income
pl <- ggplot(df, aes(age))
pl + geom_histogram(bins = 70,aes(fill=income), color= "black")
# Plot a histogram of hours worked per week
pl2  <- ggplot(df, aes(hr_per_week))
```

```
pl2 + geom_histogram()

# Rename the country column to region column (Rename the column name)
colnames(df)[colnames(df) == "country"] = "region"
head(df)

# Create a barplot of region with the fill color defined by income class
# Plot a histogram of hours worked per week
pl3 <- ggplot(df,aes(x = fct_infreq(region)))
pl3 + geom_bar(aes(fill = income), color= "black")
```




## Building a Model

Now it's time to build a model to classify people into two groups: Above or Below 50k in Salary

```
# Building a Model
# Logistic Regression
head(df)

# Train Test Split
set.seed(24)
split = sample.split(df$income, SplitRatio = 0.70)
train = subset(df, split == TRUE)
test = subset(df, split == FALSE)

# Training the Model
model <- glm(formula=income ~ . , family = binomial(logit),data = train)
summary(model)

# step()
# iteratively tries to remove predictor variables from the model
# attempt to delete variables that do not significantly add to the fit
new.model <- step(model)
summary(new.model)


#predict
p <- predict(new.model,newdata = test,type= "response")
result <- ifelse(p > 0.5,TRUE,FALSE)
test.con <- ifelse(test$income == ">50K",TRUE, FALSE)

# confusion matrix
con <- table(result,test.con)[2:1, 2:1]
#            test.con
# result  TRUE FALSE
#   TRUE  1356   469
#   FALSE  896  6327
```

```
# convert con to dataframe
con.df <- as.data.frame(con)

# Accuracy
# error <- mean(result != test.con)
# ac <- (1-error)*100
# sum(con.df$Freq[c(1,2)]
TP <- con.df$Freq[1]
TN <- con.df$Freq[4]
FP <- con.df$Freq[3]
FN <- con.df$Freq[2]
ac <- (TN + TP)/sum(con.df$Freq)

# Recall : TP/(TP+FN)
rc <- TP/(TP + FN)

# Precision : TP/(TP+FP)
pt <- TP/(TP+FP)

cat("Accuracy:",round(ac*100,2),"%\nRecall:",round(rc*100,2),"%\nprecision:",round(pt*100,2),"%")

# check confusion matrix -> caret
confusionMatrix(con)
recall(test.con,result)
```

- Accuracy: 84.9 %
- Recall: 60.2 %
- precision: 74.3 %