```
################################################################################
# Project:      Credit Scoring Analysis (petit example)
# Description:  Part 1 - Data processing and cleaning
# Data:         CreditScoring.csv
# By:           Gaston Sanchez
# url:          www.gastonsanchez.com
################################################################################

# remember to change your working directory!!! (don't use mine)
# setwd("/Users/gaston/Documents/Gaston/StatsDataMining")

# import data in file 'CreditScoring.csv'
dd = read.csv("CreditScoring.csv", header=TRUE, stringsAsFactors=FALSE)

# check table dimensions and description of variables
dim(dd)   # (4455, 14)
str(dd)

# see what the data looks like
head(dd)

# let's take a quick summary of the data
summary(dd)

# Questions to keep in mind:
# check maximum and minimum values
# check the scale of the variables (categorical, continuous, etc)
# look for outliers, weird values, missing values


# ==============================================================================
# Check categorical variables
# ==============================================================================

# how many missing values (coded as 0)
table(dd$Status)
table(dd$Home)
table(dd$Marital)
table(dd$Job)

# we'll delete missing values since there are only a few of them
good = dd$Status != 0 & dd$Home !=0 & dd$Marital !=0 & dd$Job != 0
dd = dd[good, ]

# check the new dimensions of the dataset
dim(dd)   # (4446, 14)


# ==============================================================================
# Check continuous variables
# ==============================================================================

# create single vectors for the following variables
income = dd$Income
assets = dd$Assets
debt = dd$Debt
seniority = dd$Seniority

# how many missing values (coded as 99999999)
sum(income == 99999999)    # 31
sum(assets == 99999999)    # 41
sum(debt == 99999999)      # 12
sum(seniority == 99999999) # 0
```

```
# how many zeros (this is just for curiosity)
sum(income == 0)          # 346
sum(assets == 0)          # 1626
sum(debt == 0)            # 3667
sum(seniority == 0)       # 532

# codify missing values as NA
income[income == 99999999 | income == 0] <- NA
assets[assets == 99999999] <- NA
debt[debt == 99999999] <- NA

# since there are a lot of missing values in the continuous variables
# we need to do some imputation
# let's apply knn (k-nearest neighbor) imputation

# load library class
library(class)

# let's do 1nn imputation
# (i.e. look for the most similar neighbor based on the remaining variables)

# imputation with income
dd.aux = dd[,-10]
aux.ok = dd.aux[!is.na(income),]
aux.na = dd.aux[is.na(income),]
knn.income = knn(aux.ok, aux.na, income[!is.na(income)])
income[is.na(income)] = knn.income

# imputation with assets
dd.aux = dd[,-11]
aux.ok = dd.aux[!is.na(assets),]
aux.na = dd.aux[is.na(assets),]
knn.assets = knn(aux.ok, aux.na, assets[!is.na(assets)])
assets[is.na(assets)] = knn.assets

# imputation with debt
dd.aux = dd[,-12]
aux.ok = dd.aux[!is.na(debt),]
aux.na = dd.aux[is.na(debt),]
knn.debt = knn(aux.ok, aux.na, debt[!is.na(debt)])
debt[is.na(debt)] = knn.debt

# let's substitute the imputed variables
dd$Income = income
dd$Assets = assets
dd$Debt = debt

# let's check again the data
# verify that there's no missing and/or weird values
dim(dd)   # (4446, 14)
summary(dd)


# ================================================================================
# Reformat categorical variables
# ================================================================================
# This step has to be done after the knn imputation, otherwise you'll get
# an ugly error message wihtout having no clue at all of what's wrong

# specify categorical variables as factors
dd$Status = as.factor(dd$Status)
dd$Home = as.factor(dd$Home)
dd$Marital = as.factor(dd$Marital)
dd$Records = as.factor(dd$Records)
dd$Job = as.factor(dd$Job)
```

```
# change factor levels (i.e. categories)
levels(dd$Status) = c("good", "bad")
levels(dd$Home) = c("rent", "owner", "priv", "ignore", "parents", "other")
levels(dd$Marital) = c("single", "married", "widow", "separated", "divorced")
levels(dd$Records) = c("no_rec", "yes_rec")
levels(dd$Job) = c("fixed", "partime", "freelance", "others")


# ================================================================================
# Define new variables
# ================================================================================

# let's create two more variables:
# 1) financing ratio
# 2) savings capacity

# add financing ratio
dd$Finrat = 100 * dd$Amount / dd$Price

# what's the distribution of financing ratio
hist(dd$Finrat, col="gray85", main="Financing Ratio")

# add savings capacity
dd$Savings = (dd$Income - dd$Expenses - (dd$Debt/100)) / (dd$Amount / dd$Time)

# what's the distribution of savings capacity
hist(dd$Savings, col="gray85", main="Savings Capacity")


# ================================================================================
# Recoding (categorizing) continuous variables
# ================================================================================

# The next stage involves recoding continuous variables into
# categorical values. This step assumes that you have
# previously explored the distributions of the continuous
# variables in order to determine appropriate cutoffs

seniorityR = cut(dd$Seniority, breaks=c(-1,1,3,8,14,99))
timeR = cut(dd$Time, breaks=c(0,12,24,36,48,99))
ageR = cut(dd$Age, breaks=c(0,25,30,40,50,99))
expensesR = cut(dd$Expenses, breaks=c(0,40,50,60,80,9999))
incomeR = cut(dd$Income, breaks=c(0,80,110,140,190,9999))
assetsR = cut(dd$Assets, breaks=c(-1,0,3000,5000,8000,999999))
debtR = cut(dd$Debt, breaks=c(-1,0,500,1500,2500,999999))
amountR = cut(dd$Amount, breaks=c(0,600,900,1100,1400,99999))
priceR = cut(dd$Price, breaks=c(0,1000,1300,1500,1800,99999))
finratR = cut(dd$Finrat, breaks=c(0,50,70,80,90,100))
savingsR = cut(dd$Savings, breaks=c(-99,0,2,4,6,99))

# let's label the categorized variables
levels(seniorityR) = paste("sen", levels(seniorityR))
levels(timeR) = paste("time", levels(timeR))
levels(ageR) = paste("age", levels(ageR))
levels(expensesR) = paste("exp", levels(expensesR))
levels(incomeR) = paste("inc", levels(incomeR))
levels(assetsR) = paste("asset", levels(assetsR))
levels(debtR) = paste("debt", levels(debtR))
levels(amountR) = paste("am", levels(amountR))
levels(priceR) = paste("priz", levels(priceR))
levels(finratR) = paste("finr", levels(finratR))
levels(savingsR) = paste("sav", levels(savingsR))

# add categorized variables to dataframe
```

```
dd$seniorityR = seniorityR
dd$timeR = timeR
dd$ageR = ageR
dd$expensesR = expensesR
dd$incomeR = incomeR
dd$assetsR = assetsR
dd$debtR = debtR
dd$amountR = amountR
dd$priceR = priceR
dd$finratR = finratR
dd$savingsR = savingsR


# Once we have preprocessed the data, we can save it in a new file
# This is the file we'll be using in the next parts
write.csv(dd, "CleanCreditScoring.csv", row.names=FALSE)
```