HW6

William Florez (wflorez)

28 de junio de 2017

HW6

Question 1

I tried to adapt (http://simpy.readthedocs.io/en/latest/examples/carwash.html) but I couln'd make this good. I don't get how to generate randomness into the model. I tried with CHECKTIME * expon.rvs(scale = 0.75). But I was lost with that. But I think that is a good try.

```
Passenger 0 arrives at the BoardingChecks at 0.00.
Passenger 1 arrives at the BoardingChecks at 0.00.
Passenger 2 arrives at the BoardingChecks at 0.00.
Passenger 3 arrives at the BoardingChecks at 0.00.
Passenger 0 enters the BoardingChecks at 0.00.
Passenger 0 leaves the BoardingChecks at 1.00.
Passenger 0 enters the Security Checks at 1.00.
Passenger 0 leaves the Security Checks at 2.00.
Passenger 1 enters the BoardingChecks at 2.00.
Passenger 1 leaves the BoardingChecks at 3.00.
Passenger 1 enters the Security Checks at 3.00.
Passenger 1 leaves the Security Checks at 4.00.
Passenger 2 enters the BoardingChecks at 4.00.
Passenger 4 arrives at the BoardingChecks at 5.00.
Passenger 2 leaves the BoardingChecks at 5.00.
Passenger 2 enters the Security Checks at 5.00.
Passenger 2 leaves the Security Checks at 6.00.
Passenger 3 enters the BoardingChecks at 6.00.
Passenger 3 leaves the BoardingChecks at 7.00.
Passenger 3 enters the Security Checks at 7.00.
Passenger 3 leaves the Security Checks at 8.00.
Passenger 4 enters the BoardingChecks at 8.00.
Passenger 4 leaves the BoardingChecks at 9.00.
Passenger 4 enters the Security Checks at 9.00.
Passenger 5 arrives at the BoardingChecks at 10.00.
Passenger 4 leaves the Security Checks at 10.00.
Passenger 5 enters the BoardingChecks at 10.00.
Passenger 5 leaves the BoardingChecks at 11.00.
Passenger 5 enters the Security Checks at 11.00.
Passenger 5 leaves the Security Checks at 12.00.
Passenger 6 arrives at the BoardingChecks at 17.00.
Passenger 6 enters the BoardingChecks at 17.00.
Passenger 6 leaves the BoardingChecks at 18.00.
Passenger 6 enters the Security Checks at 18.00.
Passenger 6 leaves the Security Checks at 19.00.
Passenger 7 arrives at the BoardingChecks at 23.00.
```

Passenger 7 enters the BoardingChecks at 23.00.
Passenger 7 leaves the BoardingChecks at 24.00.
Passenger 7 enters the Security Checks at 24.00.
Passenger 7 leaves the Security Checks at 25.00.
Passenger 8 arrives at the BoardingChecks at 29.00.
Passenger 8 enters the BoardingChecks at 29.00.
Passenger 8 leaves the BoardingChecks at 30.00.
Passenger 8 enters the Security Checks at 30.00.
Passenger 8 leaves the Security Checks at 31.00.
Passenger 9 arrives at the BoardingChecks at 35.00.
Passenger 9 enters the BoardingChecks at 35.00.
Passenger 9 leaves the BoardingChecks at 36.00.
Passenger 9 enters the Security Checks at 36.00.
Passenger 9 leaves the Security Checks at 37.00.
Passenger 10 arrives at the BoardingChecks at 40.00.
Passenger 10 enters the BoardingChecks at 40.00.
Passenger 10 leaves the BoardingChecks at 41.00.
Passenger 10 enters the Security Checks at 41.00.
Passenger 10 leaves the Security Checks at 42.00.
Passenger 11 arrives at the BoardingChecks at 49.00.
Passenger 11 enters the BoardingChecks at 49.00.
Passenger 11 leaves the BoardingChecks at 50.00.
Passenger 11 enters the Security Checks at 50.00.
Passenger 11 leaves the Security Checks at 51.00.
Passenger 12 arrives at the BoardingChecks at 54.00.
Passenger 12 enters the BoardingChecks at 54.00.
Passenger 12 leaves the BoardingChecks at 55.00.
Passenger 12 enters the Security Checks at 55.00.
Passenger 12 leaves the Security Checks at 56.00.
Passenger 13 arrives at the BoardingChecks at 63.00.
Passenger 13 enters the BoardingChecks at 63.00.
Passenger 13 leaves the BoardingChecks at 64.00.
Passenger 13 enters the Security Checks at 64.00.
Passenger 13 leaves the Security Checks at 65.00.
Passenger 14 arrives at the BoardingChecks at 71.00.
Passenger 14 enters the BoardingChecks at 71.00.
Passenger 14 leaves the BoardingChecks at 72.00.
Passenger 14 enters the Security Checks at 72.00.
Passenger 14 leaves the Security Checks at 73.00.
Passenger 15 arrives at the BoardingChecks at 76.00.
Passenger 15 enters the BoardingChecks at 76.00.
Passenger 15 leaves the BoardingChecks at 77.00.
Passenger 15 enters the Security Checks at 77.00.
Passenger 15 leaves the Security Checks at 78.00.
Passenger 16 arrives at the BoardingChecks at 81.00.
Passenger 16 enters the BoardingChecks at 81.00.
Passenger 16 leaves the BoardingChecks at 82.00.
Passenger 16 enters the Security Checks at 82.00.
Passenger 16 leaves the Security Checks at 83.00.
Passenger 17 arrives at the BoardingChecks at 86.00.
Passenger 17 enters the BoardingChecks at 86.00.
Passenger 17 leaves the BoardingChecks at 87.00.
Passenger 17 enters the Security Checks at 87.00.
Passenger 17 leaves the Security Checks at 88.00.
Passenger 18 arrives at the BoardingChecks at 92.00.
Passenger 18 enters the BoardingChecks at 92.00.

```
Passenger 18 leaves the BoardingChecks at 93.00.
Passenger 18 enters the Security Checks at 93.00.
Passenger 18 leaves the Security Checks at 94.00.
Passenger 19 arrives at the BoardingChecks at 98.00.
Passenger 19 enters the BoardingChecks at 98.00.
Passenger 19 leaves the BoardingChecks at 99.00.
Passenger 19 enters the Security Checks at 99.00.
Passenger 19 leaves the Security Checks at 100.00.
Passenger 20 arrives at the BoardingChecks at 107.00.
Passenger 20 enters the BoardingChecks at 107.00.
Passenger 20 leaves the BoardingChecks at 108.00.
Passenger 20 enters the Security Checks at 108.00.
Passenger 20 leaves the Security Checks at 109.00.
Passenger 21 arrives at the BoardingChecks at 116.00.
Passenger 21 enters the BoardingChecks at 116.00.
Passenger 21 leaves the BoardingChecks at 117.00.
Passenger 21 enters the Security Checks at 117.00.
Passenger 21 leaves the Security Checks at 118.00.
Passenger 22 arrives at the BoardingChecks at 121.00.
Passenger 22 enters the BoardingChecks at 121.00.
Passenger 22 leaves the BoardingChecks at 122.00.
Passenger 22 enters the Security Checks at 122.00.
Passenger 22 leaves the Security Checks at 123.00.
Passenger 23 arrives at the BoardingChecks at 130.00.
Passenger 23 enters the BoardingChecks at 130.00.
Passenger 23 leaves the BoardingChecks at 131.00.
Passenger 23 enters the Security Checks at 131.00.
Passenger 23 leaves the Security Checks at 132.00.
Passenger 24 arrives at the BoardingChecks at 136.00.
Passenger 24 enters the BoardingChecks at 136.00.
Passenger 24 leaves the BoardingChecks at 137.00.
Passenger 24 enters the Security Checks at 137.00.
Passenger 24 leaves the Security Checks at 138.00.
Passenger 25 arrives at the BoardingChecks at 145.00.
Passenger 25 enters the BoardingChecks at 145.00.
Passenger 25 leaves the BoardingChecks at 146.00.
Passenger 25 enters the Security Checks at 146.00.
Passenger 25 leaves the Security Checks at 147.00.
```

```
import random
import scipy
from scipy.stats import expon
import simpy


RANDOM_SEED = 42
NUM_MACHINES = 1  # Number of machines in the BoardingChecks
CHECKTIME = 1      # Minutes it takes to bording check below I adjusted with exponential dist
CHECKTIME2 = 1 # Minutes it takes to security checke below I adjusted with exponential dist
T_INTER = 1       # Create a passenger every ~n minutes, below I adjusted it to a exponential dist
SIM_TIME = 150     # Simulation time in minutes


class BoardingChecks(object):
    """A BoardingChecks has a limited number of machines (``NUM_MACHINES``) to
    checks ID passenger in parallel.
```

```
    """
    def __init__(self, env, num_machines, CHECKTIME, CHECKTIME2):
        self.env = env
        self.machine = simpy.Resource(env, num_machines)
        self.CHECKTIME = CHECKTIME
        self.CHECKTIME2 = CHECKTIME2

    def checkId(self, passenger):
        yield self.env.timeout(CHECKTIME)


    def checkSecurity(self, passenger):
        yield self.env.timeout(CHECKTIME2)



def passenger(env, name, cw):
    """The passenger process (each passenger has a ``name``) arrives at the BoardingChecks
    (``cw``) and requests a check machine and then security check.



    """
    print('%s arrives at the BoardingChecks at %.2f.' % (name, env.now))
    with cw.machine.request() as request:
        yield request

        print('%s enters the BoardingChecks at %.2f.' % (name, env.now))
        yield env.process(cw.checkId(name))

        print('%s leaves the BoardingChecks at %.2f.' % (name, env.now))

        print('%s enters the Security Checks at %.2f.' % (name, env.now))
        yield env.process(cw.checkSecurity(name))

        print('%s leaves the Security Checks at %.2f.' % (name, env.now))

def setup(env, num_machines, CHECKTIME, CHECKTIME2, t_inter):

    boardingChecks = BoardingChecks(env, num_machines, CHECKTIME * expon.rvs(scale = 0.75),
CHECKTIME2 * expon.rvs(scale = 0.5))

    # Create 4 initial passenger
    for i in range(4):
        env.process(passenger(env, 'Passenger %d' % i, boardingChecks))

    # Create more passengers while the simulation is running
    while True:
        yield env.timeout(random.randint(t_inter - 2, t_inter + 2))
        i += 1
        env.process(passenger(env, 'Passenger %d' % i, boardingChecks))


# Setup and start the simulation
```

```
print('BoardingChecks')
random.seed(RANDOM_SEED)  # This helps reproducing the results

# Create an environment and start the setup process
env = simpy.Environment()
env.process(setup(env, NUM_MACHINES, CHECKTIME, CHECKTIME2, T_INTER))

# Execute!
env.run(until=SIM_TIME)
```

Question 2

```
require(dplyr)

## Loading required package: dplyr

##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union

require(mice)

## Loading required package: mice

require(knitr)

## Loading required package: knitr

library(VIM)

## Loading required package: colorspace

## Loading required package: grid

## Loading required package: data.table

##
## Attaching package: 'data.table'

## The following objects are masked from 'package:dplyr':
##
##     between, first, last

## VIM is ready to use.
##  Since version 4.0.0 the GUI is in its own package VIMGUI.
##
##           Please use the package to use the new (and old) GUI.
```

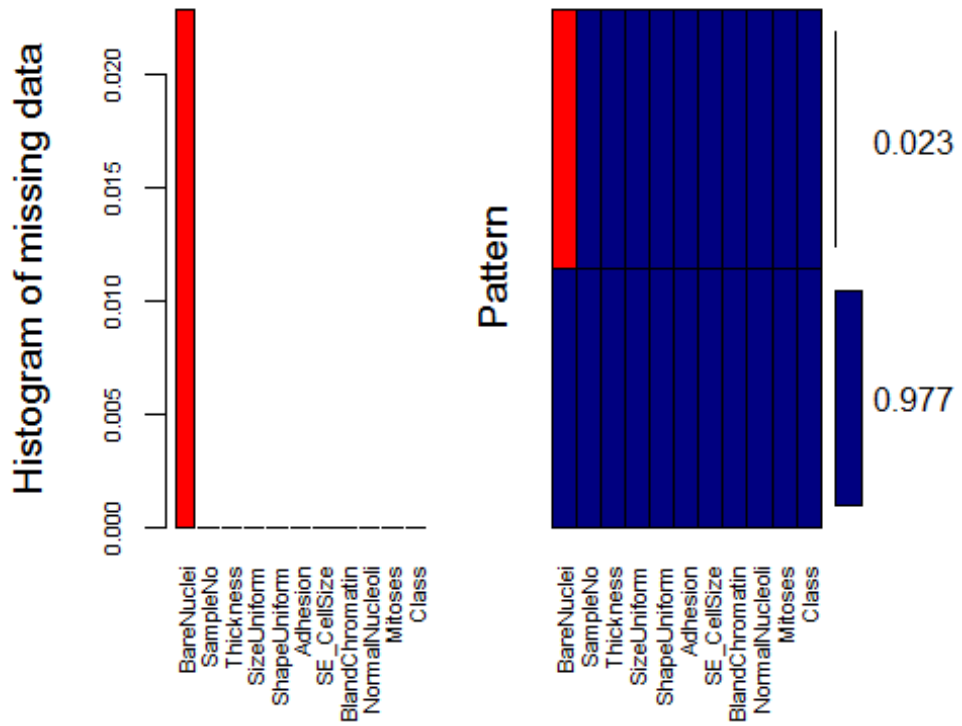
```
## Suggestions and bug-reports can be submitted at:
https://github.com/alexkowa/VIM/issues

##
## Attaching package: 'VIM'

## The following object is masked from 'package:datasets':
##
##     sleep

breast.data <- read.csv("http://archive.ics.uci.edu/ml/machine-learning-
databases/breast-cancer-wisconsin/breast-cancer-wisconsin.data",
                        header = F,  na.strings = "?")

colnames(breast.data) <- c("SampleNo",
                           "Thickness",
                           "SizeUniform",
                           "ShapeUniform",
                           "Adhesion",
                           "SE_CellSize",
                           "BareNuclei",
                           "BlandChromatin",
                           "NormalNucleoli",
                           "Mitoses",
                           "Class")

summary(breast.data)

##     SampleNo          Thickness       SizeUniform      ShapeUniform
##  Min.   :   61634   Min.   : 1.000   Min.   : 1.000   Min.   : 1.000
##  1st Qu.:  870688   1st Qu.: 2.000   1st Qu.: 1.000   1st Qu.: 1.000
##  Median : 1171710   Median : 4.000   Median : 1.000   Median : 1.000
##  Mean   : 1071704   Mean   : 4.418   Mean   : 3.134   Mean   : 3.207
##  3rd Qu.: 1238298   3rd Qu.: 6.000   3rd Qu.: 5.000   3rd Qu.: 5.000
##  Max.   :13454352   Max.   :10.000   Max.   :10.000   Max.   :10.000
##
##     Adhesion       SE_CellSize       BareNuclei     BlandChromatin
##  Min.   : 1.000   Min.   : 1.000   Min.   : 1.000   Min.   : 1.000
##  1st Qu.: 1.000   1st Qu.: 2.000   1st Qu.: 1.000   1st Qu.: 2.000
##  Median : 1.000   Median : 2.000   Median : 1.000   Median : 3.000
##  Mean   : 2.807   Mean   : 3.216   Mean   : 3.545   Mean   : 3.438
##  3rd Qu.: 4.000   3rd Qu.: 4.000   3rd Qu.: 6.000   3rd Qu.: 5.000
##  Max.   :10.000   Max.   :10.000   Max.   :10.000   Max.   :10.000
##                                    NA's   :16
##  NormalNucleoli      Mitoses           Class
##  Min.   : 1.000   Min.   : 1.000   Min.   :2.00
##  1st Qu.: 1.000   1st Qu.: 1.000   1st Qu.:2.00
##  Median : 1.000   Median : 1.000   Median :2.00
##  Mean   : 2.867   Mean   : 1.589   Mean   :2.69
##  3rd Qu.: 4.000   3rd Qu.: 1.000   3rd Qu.:4.00
```

```
##  Max.   :10.000   Max.   :10.000   Max.   :4.00
##
```

```
str(breast.data)
```

```
## 'data.frame':    699 obs. of  11 variables:
##  $ SampleNo      : int  1000025 1002945 1015425 1016277 1017023
1017122 1018099 1018561 1033078 1033078 ...
##  $ Thickness     : int  5 5 3 6 4 8 1 2 2 4 ...
##  $ SizeUniform   : int  1 4 1 8 1 10 1 1 1 2 ...
##  $ ShapeUniform  : int  1 4 1 8 1 10 1 2 1 1 ...
##  $ Adhesion      : int  1 5 1 1 3 8 1 1 1 1 ...
##  $ SE_CellSize   : int  2 7 2 3 2 7 2 2 2 2 ...
##  $ BareNuclei    : int  1 10 2 4 1 10 10 1 1 1 ...
##  $ BlandChromatin: int  3 3 3 3 3 9 3 3 1 2 ...
##  $ NormalNucleoli: int  1 2 1 7 1 7 1 1 1 1 ...
##  $ Mitoses       : int  1 1 1 1 1 1 1 1 5 1 ...
##  $ Class         : int  2 2 2 2 2 4 2 2 2 2 ...
```

```
#Checking % of NAs
aggr_plot <- aggr(breast.data, col=c('navyblue','red'), numbers=TRUE,
sortVars=TRUE, labels=names(breast.data), cex.axis=.7, gap=3,
ylab=c("Histogram of missing data","Pattern"))
```




```
##
##  Variables sorted by number of missings:
##        Variable      Count
##      BareNuclei 0.02288984
```

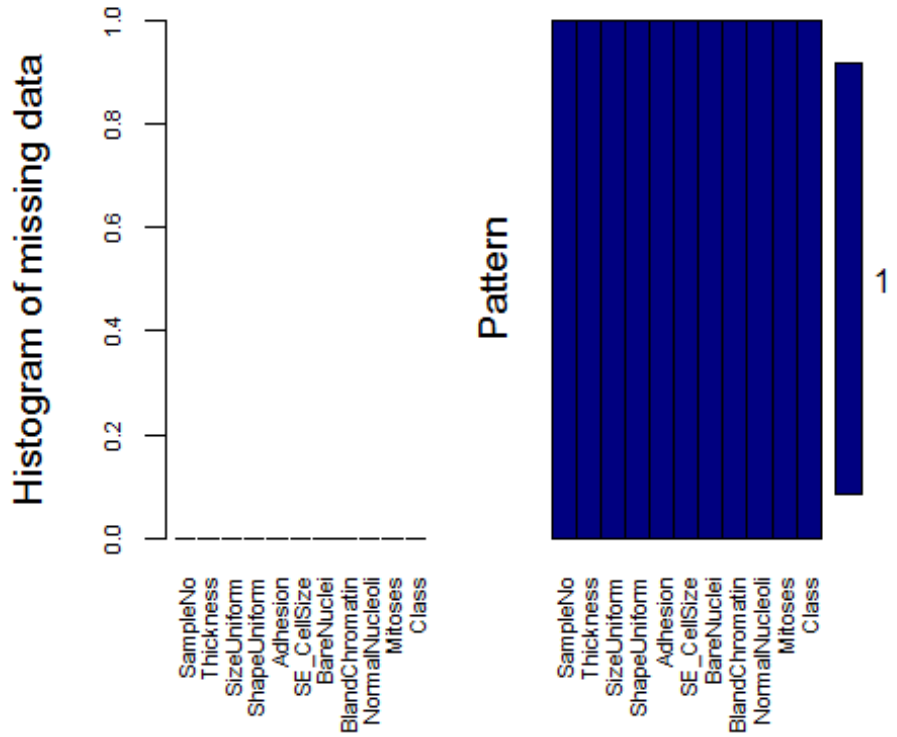
```
##        SampleNo 0.00000000
##       Thickness 0.00000000
##     SizeUniform 0.00000000
##    ShapeUniform 0.00000000
##        Adhesion 0.00000000
##     SE_CellSize 0.00000000
##  BlandChromatin 0.00000000
##  NormalNucleoli 0.00000000
##         Mitoses 0.00000000
##           Class 0.00000000
```

```r
# Setting seed.
set.seed(123)
# mean values
mean.imputation <- mice(breast.data, method = "pmm", maxit = 10)
```

```
##
##  iter imp variable
##   1   1  BareNuclei
##   1   2  BareNuclei
##   1   3  BareNuclei
##   1   4  BareNuclei
##   1   5  BareNuclei
##   2   1  BareNuclei
##   2   2  BareNuclei
##   2   3  BareNuclei
##   2   4  BareNuclei
##   2   5  BareNuclei
##   3   1  BareNuclei
##   3   2  BareNuclei
##   3   3  BareNuclei
##   3   4  BareNuclei
##   3   5  BareNuclei
##   4   1  BareNuclei
##   4   2  BareNuclei
##   4   3  BareNuclei
##   4   4  BareNuclei
##   4   5  BareNuclei
##   5   1  BareNuclei
##   5   2  BareNuclei
##   5   3  BareNuclei
##   5   4  BareNuclei
##   5   5  BareNuclei
##   6   1  BareNuclei
##   6   2  BareNuclei
##   6   3  BareNuclei
##   6   4  BareNuclei
##   6   5  BareNuclei
##   7   1  BareNuclei
##   7   2  BareNuclei
```

```
##    7    3  BareNuclei
##    7    4  BareNuclei
##    7    5  BareNuclei
##    8    1  BareNuclei
##    8    2  BareNuclei
##    8    3  BareNuclei
##    8    4  BareNuclei
##    8    5  BareNuclei
##    9    1  BareNuclei
##    9    2  BareNuclei
##    9    3  BareNuclei
##    9    4  BareNuclei
##    9    5  BareNuclei
##    10   1  BareNuclei
##    10   2  BareNuclei
##    10   3  BareNuclei
##    10   4  BareNuclei
##    10   5  BareNuclei
```

```
complete.data.mean <- complete(mean.imputation,1)

#Checking % of NAs with mean imputation
aggr_plot <- aggr(complete.data.mean, col=c('navyblue','red'),
numbers=TRUE, sortVars=TRUE, labels=names(breast.data), cex.axis=.7,
gap=3, ylab=c("Histogram of missing data","Pattern"))
```



```
##
##  Variables sorted by number of missings:
##        Variable Count
##        SampleNo     0
##       Thickness     0
##     SizeUniform     0
##    ShapeUniform     0
##        Adhesion     0
##     SE_CellSize     0
##      BareNuclei     0
##  BlandChromatin     0
##  NormalNucleoli     0
##         Mitoses     0
##           Class     0
```

```
# regressed values
regressed.imputation <- mice(breast.data, method = "norm.predict", maxit
= 10)
```

```
##
##  iter imp variable
##   1   1  BareNuclei
##   1   2  BareNuclei
##   1   3  BareNuclei
##   1   4  BareNuclei
##   1   5  BareNuclei
##   2   1  BareNuclei
##   2   2  BareNuclei
##   2   3  BareNuclei
##   2   4  BareNuclei
##   2   5  BareNuclei
##   3   1  BareNuclei
##   3   2  BareNuclei
##   3   3  BareNuclei
##   3   4  BareNuclei
##   3   5  BareNuclei
##   4   1  BareNuclei
##   4   2  BareNuclei
##   4   3  BareNuclei
##   4   4  BareNuclei
##   4   5  BareNuclei
##   5   1  BareNuclei
##   5   2  BareNuclei
##   5   3  BareNuclei
##   5   4  BareNuclei
##   5   5  BareNuclei
##   6   1  BareNuclei
##   6   2  BareNuclei
##   6   3  BareNuclei
##   6   4  BareNuclei
```

```
##   6   5  BareNuclei
##   7   1  BareNuclei
##   7   2  BareNuclei
##   7   3  BareNuclei
##   7   4  BareNuclei
##   7   5  BareNuclei
##   8   1  BareNuclei
##   8   2  BareNuclei
##   8   3  BareNuclei
##   8   4  BareNuclei
##   8   5  BareNuclei
##   9   1  BareNuclei
##   9   2  BareNuclei
##   9   3  BareNuclei
##   9   4  BareNuclei
##   9   5  BareNuclei
##   10   1  BareNuclei
##   10   2  BareNuclei
##   10   3  BareNuclei
##   10   4  BareNuclei
##   10   5  BareNuclei
```

```
complete.data.imputation <- complete(mean.imputation,1)

#Checking % of NAs with regressed values
aggr_plot <- aggr(complete.data.imputation, col=c('navyblue','red'),
numbers=TRUE, sortVars=TRUE, labels=names(breast.data), cex.axis=.7,
gap=3, ylab=c("Histogram of missing data","Pattern"))
```

```
##
##  Variables sorted by number of missings:
##        Variable Count
##        SampleNo     0
##       Thickness     0
##     SizeUniform     0
##    ShapeUniform     0
##        Adhesion     0
##     SE_CellSize     0
##      BareNuclei     0
##  BlandChromatin     0
##  NormalNucleoli     0
##         Mitoses     0
##           Class     0
```

```
# Impute using pertubation
ind <- is.na(breast.data[,7])
data.nona <- breast.data[!ind, ]
complete.data.perturbation = breast.data
perturbation  = rnorm(nrow(data.nona ))*0.1
model <- lm(BareNuclei + perturbation ~ ., data = data.nona )
complete.data.perturbation[, 7] = round(predict(model, breast.data),
```

```
digits = 0)

#Checking % of NAs with regressed+perturbation values
aggr_plot <- aggr(complete.data.perturbation, col=c('navyblue','red'),
numbers=TRUE, sortVars=TRUE, labels=names(breast.data), cex.axis=.7,
gap=3, ylab=c("Histogram of missing data","Pattern"))

##
##  Variables sorted by number of missings:
##        Variable Count
##        SampleNo     0
##       Thickness     0
##     SizeUniform     0
##    ShapeUniform     0
##        Adhesion     0
##     SE_CellSize     0
##      BareNuclei     0
##  BlandChromatin     0
##  NormalNucleoli     0
##         Mitoses     0
##           Class     0
```

Question 3

A linear optimization problem for could be *Portfolio Asset Allocation*. I work in investment strategy for voluntary pension fund. We need to rebalance the portfolios periodically and we need to allocate resources wisely. For that I implemented an investment portfolio optimizer.

As Target:
Weight for n Assets for allocate money

As input:
Daily % Price Change for n Assets
Expected 12 months returns

As Constrains: Asset Max Weight
Asset Min Weight
Sector Max Weight
Sector Min Weight
Max volatility allowed
Min Volatility Allowed
All weight of asset sum = 1