Homework 2

Model Based Statistical Learning

12/8/2022

This homework is to be sent by mail to aude.sportisse@inria.fr before January 25 2023.

- The first part consists in comparing imputation methods on a data set.
- The second part is the implementation of the EM algorithm for a logistic regression with missing values in the outcome variable y.

Comparison of imputation methods

The goal is to compare imputation methods on the SBS5242 dataset containing missing values, which is a synthetic subset (but realistic) of the Austrian structural business statistics data.

```
library(VIM)
data(SBS5242)
XNA <- SBS5242
```

- Visualize the missing values with the function **aggr** of the **VIM** package and compute the global percentage of missing values in the dataset.
- Introduce 30% synthetic missing values in the dataset. You can use the function **synthetic_MCAR** (defined below) or create your own function. Explain why the synthetic missing values introduced as such are MCAR.
- Compare three imputation methods (mean imputation, another single imputation, multiple imputation) by computing the MSE for the synthetic missing values.
- To analyse the stochasticity implied by the introduction of missing values, repeat the procedure (introduction of synthetic missing values and imputation) several times and plot the distribution of the MSEs by showing a boxplot.

```
###This function takes as input
# XNA: a dataset containing missing values
# percNA: the percentage of missing values we want to introduce in addition.
###This functions returns a list containing
# XNA_new: a dataset containing the old and the new added missing values,
# mask: a vector with the indexes of the new added missing values

synthetic_MCAR <- function(XNA,percNA){
  trueNA <- which(is.na(XNA))
  nbNA <- round(sum(dim(XNA))*percNA)
  syntheticNA <- sample(setdiff(1:sum(dim(XNA)),trueNA),nbNA,replace=FALSE)
  XNA_new <- XNA
  XNA_new[syntheticNA] <- NA
  return(list(XNA_new=XNA_new,mask=syntheticNA))
}
```

EM algorithm for logistic regression

Let us consider an i.i.d. sample $(y_i, X_i)_{i=1,\dots,n}$.

$y_i \in \{0,1\}$ is the outcome variable, $X_i \in \mathbb{R}^d$ are the covariates, $\beta \in \mathbb{R}^d$ is the regression parameter.

We assume the logistic regression:

$$\mathbb{P}(y_i = 1 | X_i; \beta) = \frac{1}{1+\exp(-X_i^T \beta)}$$

In the sequel, we consider that y contains some MCAR values and we derive an EM algorithm to estimate β.

- Write the observed log-likelihood. Note that in this case, we can maximize it with numerical methods (for example using the R function **glm** and argument **family=binomial**).
- Although maximizing the observed likelihood can be done easily, we will also derive an EM algorithm to maximize it. Write the full log-likelihood.
- Show that the E-step can be written as follows

$$Q(\beta; \beta^{(r)}) = \sum_{i=1}^{n} Q_i(\beta; \beta^{(r)})$$

where

$$Q_i(\beta; \beta^{(r)}) = \begin{cases} \sum_{y_i \in \{0,1\}} p(y_i | x_i; \beta^{(r)}) \log p(y_i | x_i; \beta) & \text{if } y_i \text{ is missing} \\ \log p(y_i | x_i; \beta) & \text{otherwise.} \end{cases}$$

- Code the EM algorithm.

Hint 1: remark that $Q(\beta; \beta^{(r)})$ can be seen as a weighted complete data log-likelihood ($\sum_{i=1}^{n} \sum_{k=1}^{n_i} \omega_{y_k} \log p(y_k | x_k; \beta)$) based on a sample size $N = \sum_{i=1}^{n} n_i$, where $n_i = 2$ if y_i is missing and $n_i = 1$ otherwise. The weights are $\omega_{y_k} = p(y_k | x_k; \beta^{(r)})$ if y_k is missing and $\omega_{y_k} = 1$ if y_k is observed.

*Hint 2: For the M-step, you can simply use the function **glm** with the argument **weights**.*

- Apply the EM algorithm for the synthetic data defined below. Compare the following estimators for β by computing the MSE: (i) without missing values (y, X), (ii) with missing values (y_{NA}, X) by using only the rows which do not contain missing values, (iii) with missing values (y_{NA}, X) by using the EM algorithm. Note that in (i) and (ii), you just have to use the function **glm**. Here, you can consider that the intercept is null. What do we notice for estimators (ii) and (iii) ?

```
library(mvtnorm)
set.seed(1)

d <- 3
beta_true <- c(0.1,0.5,0.9)

n <- 1000
mu <- rep(0,d)
Sigma <- diag(1,d)+matrix(0.5,d,d)

X <- rmvnorm(n, mean=mu, sigma=Sigma) #multivariate Gaussian variable
logit_link <- 1/(1+exp(-X%*%beta_true))
y <- (runif(n)<logit_link)*1

#### Introduction of MCAR values

nb_missingvalues <- 0.2*n*d
```

```
missing_idx <- sample(1:n,nb_missingvalues)
yna <- y
yna[missing_idx] <- NA
```

- Apply the EM algorithm for the cancer prostate dataset (only quantitative variables) and compare the different estimators for β. Here, consider that the intercept is **not null**.

```
set.seed(1)

canpros <- read.table(file = 'cancerprostate.csv',header = T, sep = ";")
#head(canpros)
quanti_var <- c(1,2,6,7)
canpros <- canpros[,quanti_var] # we use only the quantitative variables.

n <- dim(canpros)[1]
y <- canpros$Y
X <- cbind(rep(1,n),canpros$age,canpros$acide,canpros$log.acid)
d <- dim(X)[2]

#### Introduction of MCAR values

nb_missingvalues <- 0.2*n*d
missing_idx <- sample(1:n,nb_missingvalues)

yna <- y
yna[missing_idx] <- NA
```

- Briefly discuss the link with semi-supervised learning.