Introduction au data mining
L3 MIS - STA 1616 - 2010
V. Monbet
Données manquantes

L'objectif de ce TD est de manipuler et de comparer plusieurs méthodes d'imputation de données manquantes. La première partie propose un travail préliminaire sur des données simulées ; on utilise le logiciel R. Dans la seconde partie, on mène un travail plus avancé dans lequel on va comparer d'une part plusieurs mécanismes de non réponse et d'autre part plusieurs méthodes d'imputation.

A RENDRE Compte rendu de la partie 2.

1 Imputation par la moyenne

Inspirez vous de l'exemple vu en cours et proposez un programme en `R` permettant de mettre en évidence que l'imputation par la moyenne conduit à une sous estimation de la variance des estimateurs dans les deux problèmes suivants :

1. estimation de moments d'ordre un et deux (cas univarié) ;
2. estimation des coefficients d'un modèle linéaire à une variable explicative.

Dans les deux cas vous travaillerez avec des données simulées.

2 Imputation par régression

Le jeu de données utilisé dans cette question est tiré d'une étude sur des nouveaux nés. Il contient 5 variables - 3 sont complètement renseignées.

- `mage` : age de la mère en années.
- `mheight` : taille de la mère en inch.
- `msmoke` : 1 si la mère est fumeuse, 0 sinon.
- `gestwk` : nombre de semaines de gestation (avec données manquantes).
- `bwt` : poids du nouveau né (avec données manquantes).

Les données sont au format `xport`. Pour les importer sous SAS, utilisez le code suivant en remplaçant les '$\cdots$' par les chemins des répertoires où se trouvent votre fichier et votre librairie de travail.

```
LIBNAME TP6 'H:\...';
LIBNAME XP XPORT "H:\...\bwt.xpt";
PROC COPY IN=XP OUT=TP6;
RUN;
```

1. La table bwt.xpt ne contient pas de donnée manquante et va nous servir d'élément de comparaison. Conduisez, sur ce jeu de données, une analyse statistique incluant
   (a) l'estimation de la moyenne et de la variance des variables `gestwk` et `bwt` ;
   (b) l'estimation des coefficients de correlation entre les variables `mheight` et `bwt`, `mage` et `bwt`, `gestwk` et `bwt` ;
   (c) une analyse de la variance pour tester si le fait que la mère soit fumeuse induit une différence significative de poids à la naissance et de durée de gestation.

   Commentez les résultats obtenus.

2. Deux autres jeux de données sont disponibles : bwt_MCAR.xpt avec des données manquantes complètement aléatoirement et bwt_MAR.xpt avec des données manquantes aléatoirement et dont le mécanisme de non réponse dépend des 3 premières variables. Pour ces trois jeux de données répétez les analyses réalisées à la question précédente dans les cas suivants puis analysez et discutez les résultats obtenus.
   (a) On ne remplace pas les données manquantes.
   (b) On impute les données manquantes par la moyenne de la variable sans tenir compte des autres variables.
   (c) On impute les données manquantes par la moyenne de la variable par classe. On réalise donc une classification avant de faire l'imputation. On utilisera par exemple la `PROC CLUSTER` avec l'option `METHOD=MEDIAN` qui utilise la dissymilarité de Gower, adaptée aux cas où le jeu de données comporte des variables continues et des variables catégorielles.
   (d) On impute les données par régression (voir ci-dessous).

   Pour imputer la variable `bwt`, on utilise l'information apportée par les variables `mage`, `mheight` et `msmoke` comme des variables indépendantes pour estimer les paramètres d'une modèle de régression linéaire à partir des individus complètement renseignés.

```
proc glm data=imputed;
MODEL bwt = mage mheight msmoke / solution;
OUTPUT out=regout
p=bwthat;
run;
```

```
quit;
data imputed;
merge imputed regout(keep=bwthat);
if bwt = . then bwtimp=bwthat;
else bwtimp=bwt;
run;
```

Pour imputer la variable `gestwks`, on utilise l'information apportée par les variables `mage`, `mheight`,`msmoke` et les valeurs imputées de la variable `bwt`.

```
proc glm data=imputed;
MODEL gestwk = bwtimp mage mheight msmoke /
solution;
OUTPUT out=regout
p= gestwkshat;
run;
quit;
data imputed;
merge imputed regout(keep= gestwkshat);
if gestwk = . then gestwksimp= gestwkshat;
else gestwksimp = gestwk;
run;
```