Weekly Summary 5

Seline Yang

2/26/2018

1. Complete Case Analysis

Purpose: Remove all data points with missing data before fitting a model.

Summary: It is the process of conducting a statistical analysis by using only the data points with no missing data on relevant variables. Before we decide how to handle missing data, we need to look at the summary to see how many data points is missing. We can impute the missing data instead of removing them for certain cases. If the data are MCAR or MAR, it is ignorable. If the data are MNAR, the missing-data mechanism is non-ignorable.

Pros: These techniques are extremely easy to conduct analysis by ignoring missing data points.

Cons: It would lead to a reduction in sample size. Conduct a complete case analysis can lead to biased estimates.

R Commands:

```
nc_clean <- na.omit(nc)
```

2. Unconditional Mean Imputation

Purpose: To fill in missing data before conducting analysis.

Summary: Replace each missing value with the mean of the observed data for the variable.

Pros: Mean imputation preserves the means of varaibles.

Cons: It tends to weaken the relationship between variables and makes the distribution less variable. It will cause biased regression coefficients and invalid inference even when data are MCAR. It also exaggerates the effective size of the data set, further distorting statistical inference.

R Commands:

```
# Step 1: Create a space to hold the completed data set
nc_imp1 <- nc

# Step 2: Find the mean for fage for all rows for which fage is observed.
mean_fage <- mean(na.omit(nc$fage))

# Step 3: Impute the missing values with this mean
nc_imp1$fage <- ifelse( is.na(nc$fage)==TRUE, mean_fage, nc$fage)
```

3. Conditional Mean Imputation

Purpose: To fill in missing data before conducting analysis.

Summary: We replace each missing value with the mean of the observed data for the variable, but rather than using the global mean(the mean of all observed values for the variable), the technique uses a mean within a group.

Pros: Less bias than unconditional mean imputation.

Cons: The imputed observations tend to be less variable than real data, because they lack residual variation. Bias still exist.

R Commands:

```
# Step 1: Create the data set
nc_impCMI <- nc

# Step 2: Find the mean for fages who are married
married <- which(nc$marital=="married")
mean_fageM <- mean(na.omit(nc$fage[married]))

# Step 3: Find the mean for fages who are NOT married
notmarried <- which(nc$marital=="not married")
mean_fageNM <- mean(na.omit(nc$fage[notmarried]))

# Step 4: Impute the missing values for married men
nc_impCMI$fage <- ifelse( is.na(nc$fage)==TRUE & nc$marital=="married",
                          mean_fageM, nc$fage)

# Step 5: Impute the missing values for married men
nc_impCMI$fage <- ifelse( is.na(nc$fage)==TRUE & nc$marital=="not
                          married", mean_fageNM, nc_impCMI$fage)
```

4. Regression Mean Imputation

Purpose: To fill in missing data before conducting analysis.

Summary: We use a regression model to fill in the missing values.

Pros: When two variables are highly correlated, regression mean has less biased than unconditional/conditional mean imputations with global/group means.

Cons: We have failed to account for uncertainty in the estimation of the regression coefficients used to obtain the imputed values. It improves on unconditional mean imputation, but it generally provides biased estimates and invalid inferences even for missing data that are MCAR.

R Commands:

```
# Step 1: Create the storage space
nc_impReg <- nc

# Step 2 : Choose the model
impmodel_fage <- lm( fage~ mage + marital, data = nc)
#summary(impmodel_fage)

# Step 3: Check the model (check conditions, etc)

# Step 4: Identify which data points are missing
isna_fage <- which( is.na(nc$fage) ==TRUE )
```

```r
# Step 5: Make predictions
predict_fage <- predict( impmodel_fage, nc[isna_fage,])

# Step 6: Fill in the predictions for the missing values
nc_impReg$fage[isna_fage] <-predict_fage
```