Processing Data

Therri Usher

Thursday, February 12, 2015

Processing Data

What Is The Difference Between Cleaning and Processing?

I think of cleaning as correcting mistakes in the data so that it can be loaded and processing as making the data suitable for analysis. Some people believe the opposite. Some people believe the terms are interchangeable.

Cleaning and processing data helps strengthen the quality of the data.

Traits Regarding Data Quality

- Accuracy
  - The data was recorded correctly.
- Completeness
  - There are minimal missing values in the data.
- Uniqueness
  - There are no duplicate observations or variables in the data set.
- Timeliness
  - The data is up to date, particularly for longitudinal data sets.
- Consistency
  - The data does not contradict itself.

These qualities are unmeasurable and somewhat vague but good to keep in mind.

Key Aspects of Processing Data

Processing data includes:

- Handling missing values
- Deciding what to do with outliers
- Removing impossible values
- Pinpointing obvious inconsistencies and errors
- Indicating special values, such as missing and null values, correctly
- Removing duplicate values
- Creating new variables to be used in the analysis

Missing Data

There are three kinds of missing data:

- Missing Completely at Random (MCAR): Missing data is independent of all observable and unobservable variables.
  - Ex: Subject's diabetes status could not be measured because every mode of transportation in the city stopped working and he could not make it to the hospital.
  - VERY rare
- Missing at Random (MAR): Missing data is related to a variable but not to the missing values of the variable.
  - Ex: Subject's diabetes status could not be measured because he can only afford public transportation and the buses stopped running that day.
- Not Missing at Random (NMAR): Missing data is dependent on the missing value.
  - Ex: Subject's diabetes status could not be measured because he developed an ulcer on his foot and could not leave the bed.

Possible Solutions For Missing Data

- Deleting observations with missing data, or using complete data
  - The most common technique
  - Very simple
  - Decreases sample size and power of the analysis
  - Potentially introduces bias into the results
- Imputing missing values
  - Methods include: using the mean of observed values, using the last observed value, single imputation using regression, multiple imputation
  - Avoids decrease in sample size and power
  - Only possible for MCAR and MAR
  - No way of knowing if imputed values are correct

Removing Impossible Values

Sometimes, outliers are actually impossible values. If it is evident how the mistake was made, it can be fixed. For instance, if gender is defined as "M" and "F" and someone entered a subject's gender as "MM", it may be safe to say that the subject is male.

However, there are instances where it is impossible to tell what the true value is. In that case, the unrealistic value should be removed, either by setting the value equal to 'NA' in R, or removing the subject completely from the data set if you are using complete data.

Outliers

An outlier is "an observation point that is distant from other observations." - Wikipedia

Perhaps the most common method of detecting outliers is by using the Interquartile Range, defined as $IQR = Q3 - Q1$, where $Q3$ and $Q1$ is the 75th and 25th percentile, respectively.

An outlier is any value that lies outside the interval $[Q1 - 1.5 * IQR, Q3 + 1.5 * IQR]$.

Some analyst advocate for the deletion of outliers. Others do not practice this.

Rectifying Inconsistencies and Errors

Unfortunately, inconsistencies often appear in data. For instance, a subject might report being married but also might report their spouse being deceased. While they are technically no longer married, they may still feel married.

The question becomes "In which value does the inconsistency lie?" We usually have no way of knowing where the error lies when it comes to inconsistent information.

At this point, scientific reasoning comes into play, rather than statistical methods. There is no one correct way to handle situations such as these. Use your best judgment and always be sure to keep your research question in mind.

Discussion Time

Do you think we should impute missing values? Do you think we should delete outliers? Why or why not? Is there any scenario where you might change your mind?

Talk it over with members of your group and be prepared to share your thoughts with the class.

Defining Missing Values in R

```
id <- 1:3
age <- sample(1:100, 3, replace=T)
dat <- as.data.frame(cbind(id,age))
dat
```

```
##   id age
## 1  1  94
## 2  2  24
## 3  3  20
```

```
# Define a value as missing
dat[3,2] <- NA
is.na(dat[3,2])
```

```
## [1] TRUE
```

Defining Missing Values in R

```
# Remove an observation
dat[-(3),]
```

```
##   id age
## 1  1  94
## 2  2  24
```

```
# Remove a variable
dat$age <- NULL
dat
```

```
##   id
## 1  1
## 2  2
## 3  3
```

Defining Missing Values in R

Use NA in R to:

- Designate missing values
- Replacing impossible values and errors

Use NULL in R to:

- Removing impossible values
- Remove data associated with impossible values and errors
- Removing duplicate values

Creating New Variables in R

Often, a data set does not contain the variable you need but contains the information you need to define a variable. It is up to you to define the variable in R, such as in this example:

```
ht <- rnorm(100, mean=125, sd=10)
wt <- rnorm(100, mean=66, sd=2)

bmi <- (wt/ht^2) * 703
```

Useful Functions: If Statements and For Loops

If statements perform specified commands based on the truth of a logical statement. If you want to test multiple logical statements, use if-else-if statements.

```
die.roll <- sample(1:12, 1)

if(die.roll == 7)
{
  print("Winner")
} else if(die.roll == 1)
{
  print("Loser")
} else
{
  print("Roll Again")
}
```

```
## [1] "Roll Again"
```

Useful Functions: If Statements and For Loops

For loops repeat specified commands (that may be indexed by the iterating variable) until the iterating variable falls out of the range.

```
mat <- matrix(nrow=1, ncol=10)
mat
```

```
##      [,1] [,2] [,3] [,4] [,5] [,6] [,7] [,8] [,9] [,10]
## [1,]   NA   NA   NA   NA   NA   NA   NA   NA   NA    NA
```

```
for(i in 1:ncol(mat))
{
  mat[1,i] <- sample(c(0,1), 1)
}

mat
```

```
##      [,1] [,2] [,3] [,4] [,5] [,6] [,7] [,8] [,9] [,10]
## [1,]    1    1    1    0    1    1    0    0    1     0
```

Applying If Statements and For Loops In Processing Data

You will more than likely have to use for loops and if statements in your data. Practice using them.

```
age <- sample(-1:99, 100, replace=T)

for(i in 1:length(age))
{
  if(age[i] < 1)
  {
    age[i] <- NA
  }
}
```

Use the R resources provided to learn more about if statements and for loops.

Active Learning Exercise

For this exercise, we will be using 2007-2008 NHANES Demographic Data and we will focus on the following variables: age in months at screening, race/ethnicity, educational level for adults, and marital status. Complete the following tasks:

- Indicate missing values by replacing them with 'NA' in R. (Refused and don't know are considered missing values.)
- Create an indicator for Hispanic participants.
- Remove any participants who have an age in months considered to be an outlier.
- Verify that all participants under 18 years of age are labeled as "never married."
- Create an copy of the education variable but give the same value to participants who did not graduate high school.

Active Learning Exercise

Use the data documentation (doc file) given on the website to help you find the proper names for each variable and to learn more about what each value for each variable means.

BONUS: Define the new variable for education as a factor with the following levels: No Diploma, HS Diploma, Some College, College Degree. See `http://www.ats.ucla.edu/stat/r/modules/factor_variables.htm` for assistance.