ADS2 Week5-Getting and Cleaning Data

2019-10-14

Wanlu Liu

wanluliu@intl.zju.edu.cn

Lab website: labw.org

https://www.acaps.org/sites/acaps/files/resources/files/acaps_technical_brief_data_cleaning_april_2016_0.pdf

Some REVIEW

- Last Week, we learnt **shareable data-set**, what is it? What a shareable data-set should contain?
- You should also have some idea about **synthetic dataset**, what is it?

https://www.memecenter.com/fun/3938961/life-is-hard-so-make-it-easy

Learning Objectives

This week we will learn – how to **get** – **clean** data.

1. Describe the features of tidy data
2. Use different data types and data structures in R
3. Understand the steps of data cleaning
4. Clean a real-world dataset according to tidy/data cleaning principles

Do you still remember – Tidy dataset?

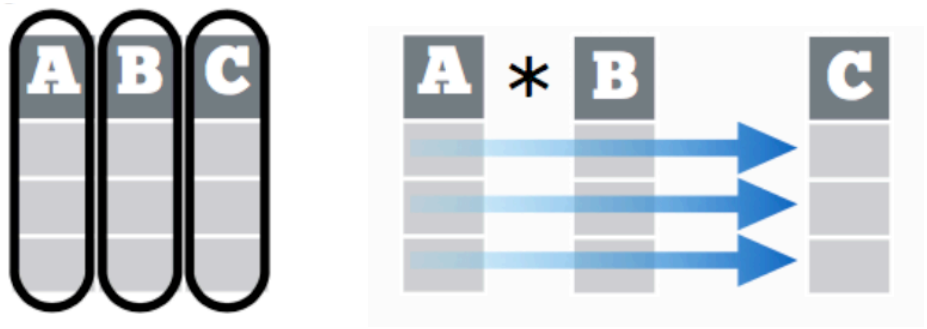
1. Each variable forms a column.
2. Each observation forms a row.
3. Each type of observational unit forms a table.
4. Fonts have been harmonized
5. Text is aligned to the left, numbers to the right
6. There are no blank rows
7. Column headers are clear and visually distinct.

A table is tidy if:

Each **variable** is in its own **column**

Each **observation**, or **case**, is in its own **row**

Tidy data:



Makes variables easy to access as vectors

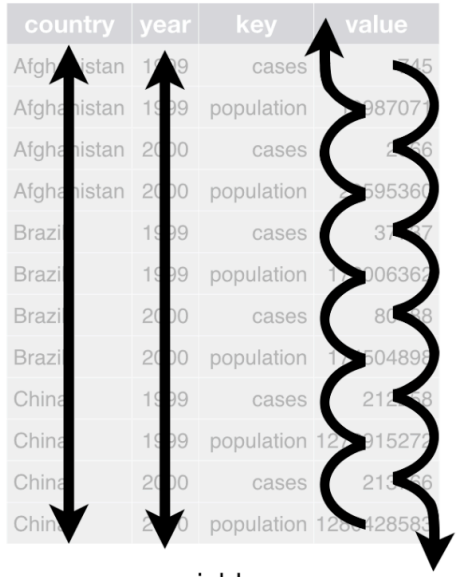
Preserves cases during vectorized operations

Do you still remember - Long vs Wide

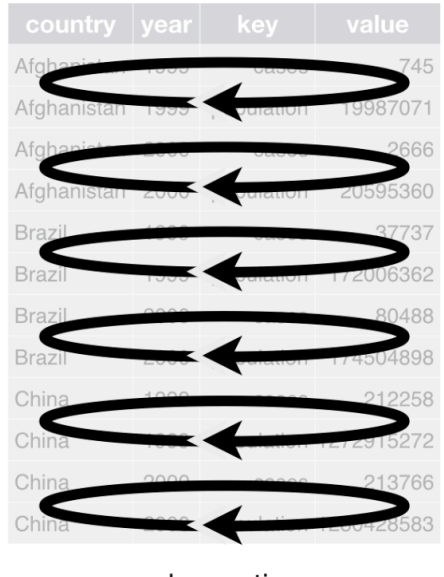
- Long

country	year	key	value
Afghanistan	1999	cases	745
Afghanistan	1999	population	19987071
Afghanistan	2000	cases	2666
Afghanistan	2000	population	20595360
Brazil	1999	cases	37737
Brazil	1999	population	172006362
Brazil	2000	cases	80488
Brazil	2000	population	174504898
China	1999	cases	212258
China	1999	population	1272915272
China	2000	cases	213766
China	2000	population	1280428583

table2



variables



observations

- Wide (tidy)

country	year	cases	population
Afghanistan	1999	745	19987071
Afghanistan	2000	2666	20595360
Brazil	1999	37737	172006362
Brazil	2000	80488	174504898
China	1999	212258	1272915272
China	2000	213766	1280428583

table1

variables

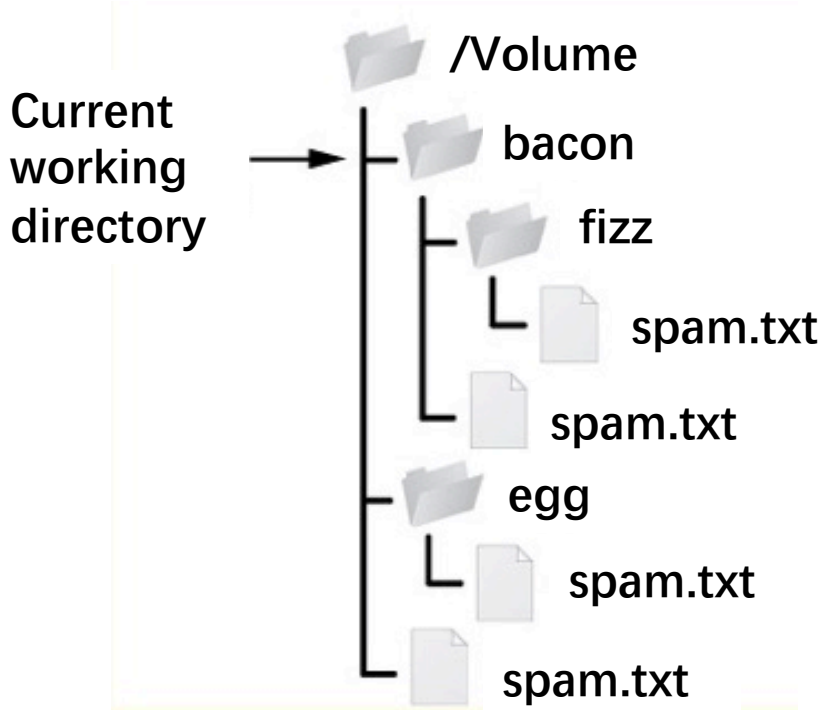
observations

Import data to R

Various functions have been implemented in R to import data including:

- read.delim() tab-delimited files with period as decimal separator.
- read.delim2() tab-delimited files with comma as decimal separator.
- read.csv() for comma separated values with period as decimal separator.
- read.csv2() for semicolon separated values with comma as decimal separator.
- read.table() The read.function() is the most flexible function to read tabular data that is stored in a textual format. In fact, the other read-functions mentioned above all eventually use read.table with some fixed parameters and possibly after some preprocessing.

Set working directory – Absolute vs. Relative Paths



Relative Paths		Absolute Paths
../	→	/Volume
./	→	/Volume/bacon
./fizz	→	/Volume/bacon/fizz
./fizz/spam.txt	→	/Volume/bacon/fizz/spam.txt
./spam.txt	→	/Volume/bacon/spam.txt
../egg	→	/Volume/egg
../egg/spam.txt	→	/Volume/egg/spam.txt
../spam.txt	→	/Volume/spam.txt

Note: this is linux/unix style, windows have different syntax

https://automatetheboringstuff.com/chapter8/

R example in import data

```
1  #set up working directory
2  #this is a absolute or relative directory?
3  setwd("/Users/wanluliu/Desktop/ADS2Week5/")
4
5  #import working data
6  #my data is in xxx.csv format, so I use read.csv() function
7  data=read.csv("Rdata_diamonds_samples100_mdf.csv")
8  head(data)
9
```

9:1 (Top Level) R Script

Console Terminal Jobs

~/Desktop/ADS2Week5/

```
> data=read.csv("Rdata_diamonds_samples100_mdf.csv")
> head(data)
  ID carat       cut color clarity depth table price    x    y    z
1 58  0.24 Very Good     G    VVS2  62.0    56   449 4.00 4.03 2.49
2  8  0.27 Very Good     D     VS1  60.4    59   470 4.15 4.20 2.52
3 53  0.29 Very Good     E    VVS1  60.9    61   629 4.23 4.27 2.59
4  1  0.30   Premium     E     VS2  61.7    60   570 4.28 4.31 2.65
5 42  0.30     Ideal     G     VS2  63.0    55   675 4.31 4.29 2.71
6 80  0.30     Ideal     H     SI1  62.2    53  1105 4.29 4.32 2.68
>
```

R data types & Structure

R Data types

- Character: “a” , “swc”
- Numeric (real or decimal): 2, 15.5
- Integer: 2L (the L tells R to store this as an integer)
- Logical: TRUE/FALSE
- Complex: 1+4i (complex numbers with real and imaginary parts)

R Data Structure

- Atomic vector
- List
- Matrix
- Data frame
- factors Factors have levels.

R provides many functions to examine features of vectors and other objects, for example
class(), typeof(), length(),attributes()

Objects can have attributes. Attributes are part of the object. These include:
names(), dimnames(), dim(), class(),attributes()

Dimensions	Homogenous	Heterogeneous
1-D	Atomic vector	list
2-D	matrix	Data frame

https://swcarpentry.github.io/r-novice-inflammation/13-supp-data-structures/

R data types

```
> head(data)
  ID carat       cut color clarity depth table price    x    y    z
1 58  0.24 Very Good     G    VVS2  62.0    56   449 4.00 4.03 2.49
2  8  0.27 Very Good     D     VS1  60.4    59   470 4.15 4.20 2.52
3 53  0.29 Very Good     E    VVS1  60.9    61   629 4.23 4.27 2.59
4  1  0.30   Premium     E     VS2  61.7    60   570 4.28 4.31 2.65
5 42  0.30     Ideal     G     VS2  63.0    55   675 4.31 4.29 2.71
6 80  0.30     Ideal     H     SI1  62.2    53  1105 4.29 4.32 2.68
>
```

```
1  #R data type
2  #check the data type of ID column
3  typeof(data$ID)
4  class(data$ID)
5  #check the data type of carat column
6  class(data$carat)
7  #check the data type of cut column
8  class(data$cut)
```

What you guess the results would be?

What you guess the results would be?

What you guess the results would be?

```
21  #difference between numeric vs integer
22  var1=c(2,3,5,6)
23  class(var1)
24  var2=c(2.0,3.9,5.1,6.9)
25  class(var2)
26  var3=c(2L,3L,5L,6L)
27  class(var3)
```

What's your guess?

What's your guess?

What's your guess?

```
18  #check the data type of cut column
19  head(data$cut)
20  class(data$cut)
21  #What's the difference between -
22  #character and factor?
23  cut.chr=as.character(data$cut)
24  head(cut.chr)
25  class(cut.chr)
```

R data structure 1-D : Vector vs List

- Generate vectors

```
36  #create numeric vector
37  var.num=c(2.0,3.9,5.1,6.9)
38  class(var.num)
39  #create integer vector
40  var.int=c(2L,3L,5L,6L)
41  class(var.int)
42  #create character vector
43  var.char=c("I","am","super","happy")
44  class(var.char)
45  #create factor vector
46  var.fac=factor(c("mid","mid","high","low"),
47                 levels=c("low","mid","high"))
48  class(var.fac)
```

```
> var.num=c(2.0,3.9,5.1,6.9)
> class(var.num)
[1] "numeric"
> var.int=c(2L,3L,5L,6L)
> class(var.int)
[1] "integer"
> var.char=c("I","am","super","happy")
> class(var.char)
[1] "character"
> var.fac=factor(c("mid","mid","high","low"),
+                levels=c("low","mid","high"))
> class(var.fac)
[1] "factor"
```

List is 1-D data that can contain different type of vectors, or even data frame/matrix!

- Generate list

```
49  #combine all vector above to create list
50  data.list=list(var.num,var.int,var.char,var.fac)
51  data.list[[1]]
52  data.list[[1]][1]
53  data.list[[3]]
54  data.list[[3]][3]
55  length(data.list)
56  dim(data.list)
57
58
59
60
```

```
> data.list=list(var.num,var.int,var.char,var.fac)
> data.list[[1]]
[1] 2.0 3.9 5.1 6.9
> data.list[[1]][1]
[1] 2
> data.list[[3]]
[1] "I"     "am"    "super" "happy"
> data.list[[3]][3]
[1] "super"
> length(data.list)
[1] 4
> dim(data.list)
NULL
```

R data structure 2-D : data frame vs matrix

- One of the most commonly used data structure in R
- 2-Dimension
- Heterogenous (can contain numeric, factor, character at the same time)

- What is the difference of data frame vs matrix?

```
#convert dataframe to matrix
data.matrix=as.matrix(data)
#look at the header of the matrix
head(data.matrix)
#check the matrix class
class(data.matrix[,1])
class(data.matrix[,2])
```

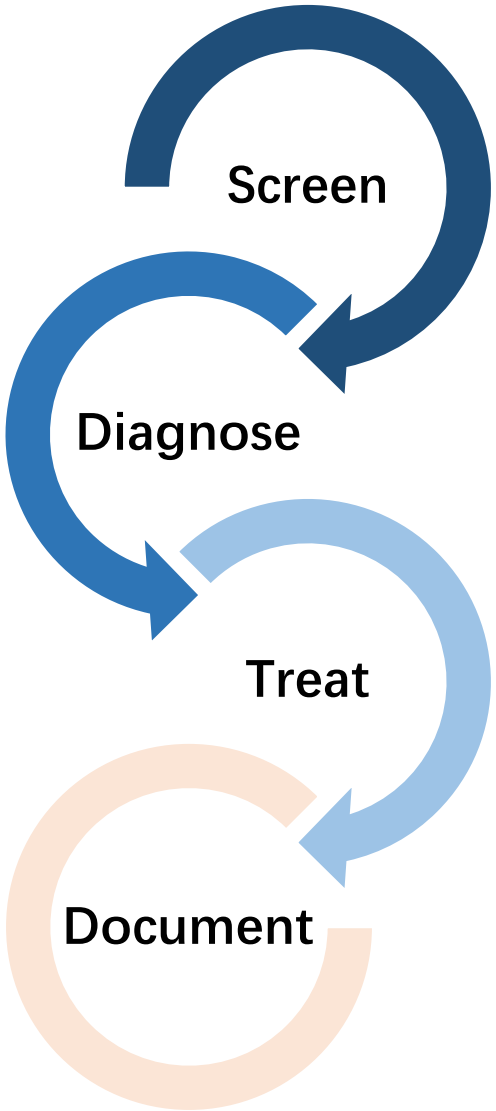
Matrix only contains data of the same type! (homogenous)

```
> head(data)
  ID carat       cut color clarity depth table price    x    y    z
1 58  0.24 Very Good     G    VVS2  62.0    56   449 4.00 4.03 2.49
2  8  0.27 Very Good     D     VS1  60.4    59   470 4.15 4.20 2.52
3 53  0.29 Very Good     E    VVS1  60.9    61   629 4.23 4.27 2.59
4  1  0.30   Premium     E     VS2  61.7    60   570 4.28 4.31 2.65
5 42  0.30     Ideal     G     VS2  63.0    55   675 4.31 4.29 2.71
6 80  0.30     Ideal     H     SI1  62.2    53  1105 4.29 4.32 2.68
>
```

```
#R data structure
#check the data type of data
class(data)
#check the dimension of data
dim(data)
#check the names of data
names(data)
#check the number of rows
nrow(data)
#check the number of cols
ncol(data)
#look at specific position
data[1,1]
data[5,3]
data[2,4]
```

```
> dim(data)
 1] 103  11
 names(data)
[1] "ID"      "carat"   "cut"     "color"
[5] "clarity" "depth"   "table"   "price"
[9] "x"       "y"       "z"
 nrow(data)
1] 103
 ncol(data)
1] 11
 data[1,1]
1] 58
 data[5,3]
1] Ideal
 Levels: Fair Good Idea Ideal ... Very Good
 data[2,4]
1] D
evels: D E F G H I J
```

The four steps of data cleaning



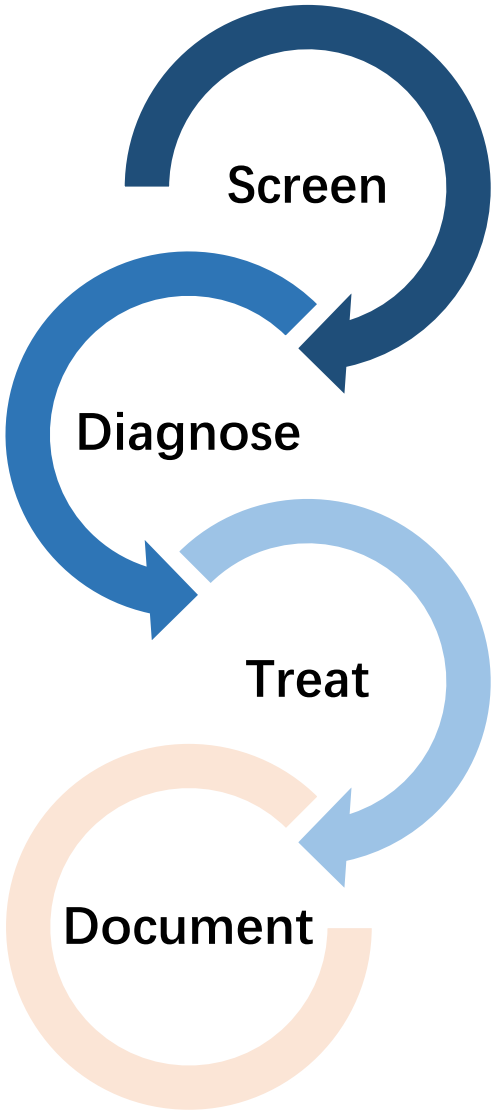
Screening involves systematically looking for suspect features in assessment questionnaires, databases, or analysis datasets.

The **diagnosis** (identifying the nature of the defective data) and **treatment** (deleting, editing or leaving the data as it is) phases of data cleaning requires an in depth understanding of all types and sources of errors possible during data collection and entry processes.

Documenting changes entails leaving an audit trail of errors detected, alterations, additions and error checking and will allow a return to the original value if required.

Van den Broeck J, Argeseanu Cunningham S, Eeckels R, Herbst K (2005) and Arthur D. Chapman https://www.acaps.org/sites/acaps/files/resources/files/acaps_technical_brief_data_cleaning_april_2016_0.pdf

The four steps of data cleaning



- Lack of data
- Excess of data
- Outliers
- Inconsistencies
- Strange patterns
- Suspect analysis results

- Data is missing
- Error
- Valid records: True experiments
- No diagnosis, still suspect

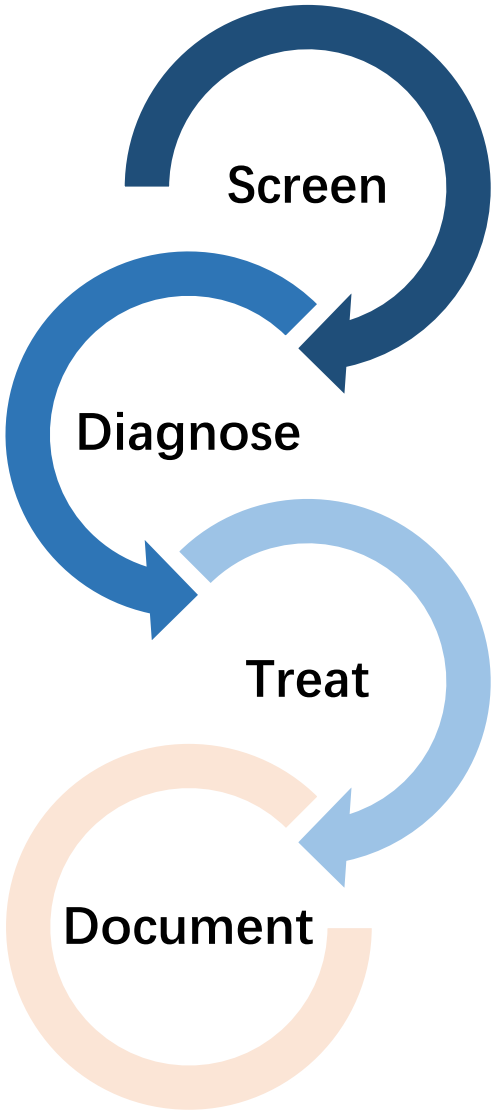
- Leave unchanged
- Correct
- Delete

- Maintain change log
- Archive raw data and old values

- **Lack of data :**Do some columns/rows have far fewer answers compared to others?
- **Excess of data:** Are there duplicate entries or more answers than originally allowed?
- **Outliers/Inconsistencies:** Are there values that are so far beyond the typical distribution that they seem potentially erroneous?
- **Strange patterns:** Are there patterns that suggest that the respondent or enumerator has not answered or recorded questions honestly? (i.e. several questionnaires with the exact same answers)?
- **Suspect analysis results:** Do the answers to some questions seem counterintuitive or extremely unlikely?

Van den Broeck J, Argeseanu Cunningham S, Eeckels R, Herbst K (2005) and Arthur D. Chapman https://www.acaps.org/sites/acaps/files/resources/files/acaps_technical_brief_data_cleaning_april_2016_0.pdf

The four steps of data cleaning



- Lack of data
- Excess of data
- Outliers
- Inconsistencies
- Strange patterns
- Suspect analysis results

- Data is missing
- Error
- Valid records: True experiments
- No diagnosis, still suspect

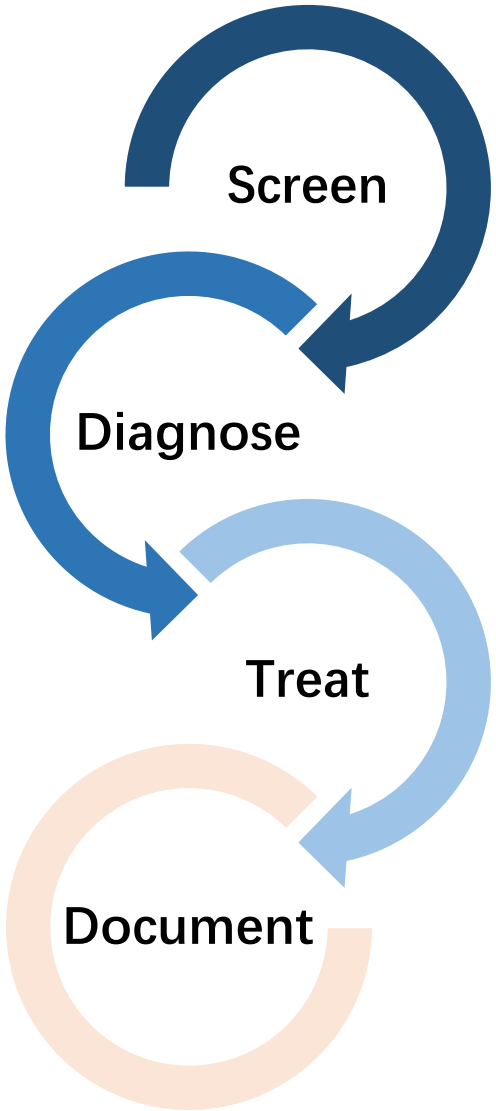
- Leave unchanged
- Correct
- Delete

- Maintain change log
- Archive raw data and old values

- **Missing data:** Answers omitted by the respondent (nonresponse), questions skipped by the enumerator or dropout
- **Errors:** Typos or answers that indicate the question was misunderstood
- **Ture extreme:** An answer that seems high but can be justified by other answers (i.e. the respondent working 60 hours a week because they work a full-time job and a part-time job at the same time)
- **Ture normal:** A valid record
- **No diagnosis, still suspect:** Make a judgement call on how to treat this data during the treatment phase

Van den Broeck J, Argeseanu Cunningham S, Eeckels R, Herbst K (2005) and Arthur D. Chapman https://www.acaps.org/sites/acaps/files/resources/files/acaps_technical_brief_data_cleaning_april_2016_0.pdf

The four steps of data cleaning



- Lack of data
- Excess of data
- Outliers
- Inconsistencies
- Strange patterns
- Suspect analysis results

- Data is missing
- Error
- Valid records: True experiments
- No diagnosis, still suspect

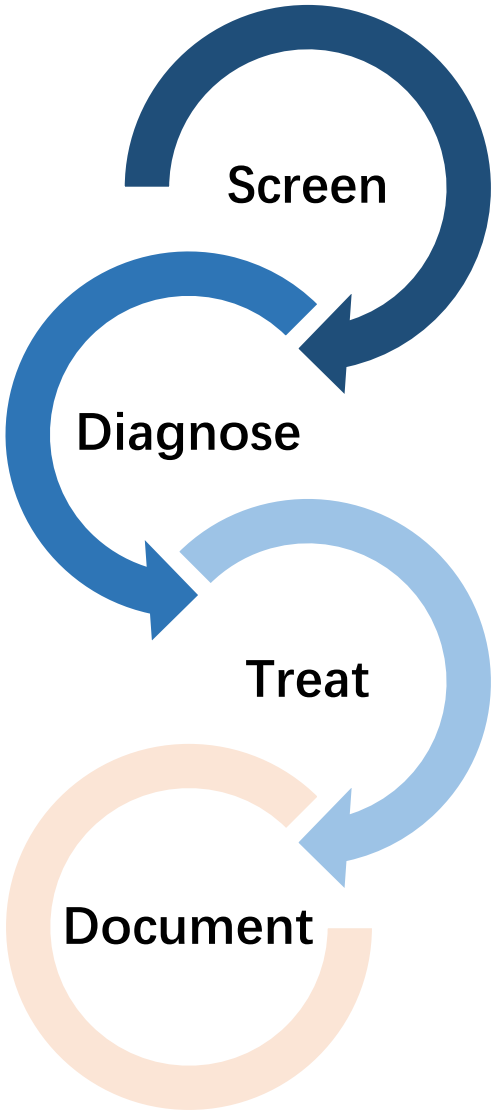
- Leave unchanged
- Correct
- Delete

- Maintain change log
- Archive raw data and old values

- **Leave it unchanged:** The larger the sample size, the less one suspect response will affect the analysis; the smaller the sample size, the more difficult the decision.
- **Correct the data:** If the respondent's original intent can be determined, correct the answer
- **Delete data:** Delete just this one cell or delete the entire record? Be careful of **"cherry picking"!** Can add another label to the data(1=suspicious record, 0=not suspicious).
- **If time and resources allows, re-measure** the suspect or erroneous values.

https://www.acaps.org/sites/acaps/files/resources/files/acaps_technical_brief_data_cleaning_april_2016_0.pdf

The four steps of data cleaning



- Lack of data
- Excess of data
- Outliers
- Inconsistencies
- Strange patterns
- Suspect analysis results

- Data is missing
- Error
- Valid records: True experiments
- No diagnosis, still suspect

- Leave unchanged
- Correct
- Delete

- Maintain change log
- Archive raw data and old values

Documentation of errors, alterations, additions and error checking:

- Maintain data quality
- Avoid duplication of error checking by different data cleaners.
- Recover data cleaning errors
- Determine the fitness of the data for use.
- Inform users who may have used the data knowing
- what changes have been made since they last accessed the data

Create a change log within the workbook, where all information related to modified fields is sourced. Within the change log, store the following information:

- Table
- Column, Row
- Date changed -> Changed by
- Old value -> New value
- Comments

Make sure to document

- Procedures implemented
- By whom
- How many responses were affected

ALWAYS make this information available when sharing the dataset internally or externally (i.e. by enclosing the change log in a separate worksheet)

https://www.acaps.org/sites/acaps/files/resources/files/acaps_technical_brief_data_cleaning_april_2016_0.pdf

Data cleaning demo with R

Referred to ADS2Week5_lectureCode.html