FALL 2018

TEXAS
The University of Texas at Austin

APPLIED ENGINEERING DATA ANALYSIS, OPTIMIZATION AND VISUALIZATION

Cleaning data

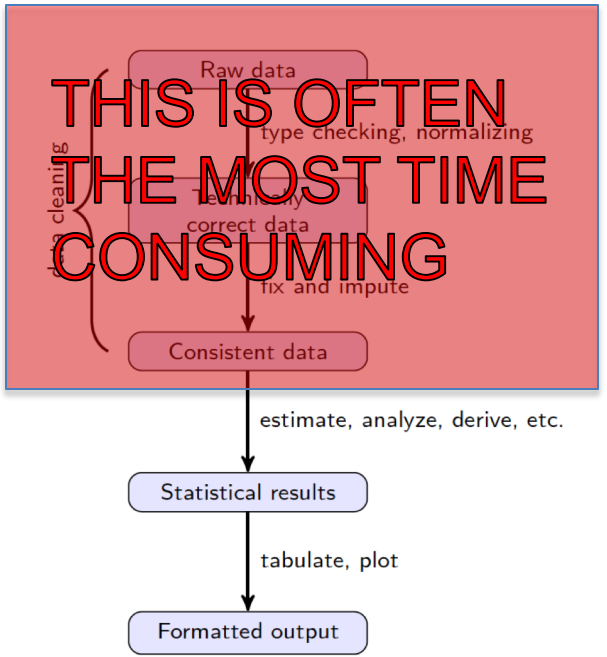
JOSHUA RHODES, PHD
Research Fellow/Adjunct Professor, The University of Texas at Austin

This lecture is based on this text, and my frustrations over the years

- https://cran.r-project.org/doc/contrib/de_Jonge+van_der_Loo-Introduction_to_data_cleaning_with_R.pdf

There are multiple stages of data

- Each step adds value to the dataset
- Each step makes the data more useful
- Typically slowest at front end



Raw data is just that, RAW

- Really Annoying to Work with
- Might be missing headers
- Have contain unknown character/data types
- Have wrong data types
  - Even worse…

Raw data might be in crazy formats

- You might need to get a table from a poorly structured PDF
- It might be in a strange format that is no longer supported but some obscure government agency puts it in the format anyway because they are supposed to by some legislative mandate and they are technically meeting that goal, but they have no incentive to make it easier to use
- It might be unstructured data, pictures, video, sound
  - This is beyond the scope of this class, but if you need this, let's talk

Technically correct data is correct, *technically*

- You can read it into an R data.frame
- Correct names
- Correct types
- Correct labels

But technically correct data are NOT usually ready for analysis

- "stuff" is a data.frame
- But I see some problems here

```
> stuff
       food numbers other_numbers cities should_not_be_NA
1    hotdog       1             3 London           people
2 hamburger       2             4 France           places
3      Stan       3      eleventy      4             <NA>
> 
```

Beware some weird ways that R deals with mixed lists and data.frames

- "stuff" is a data.frame
- But I see some problems here
- Even though "numbers" looks like it is numeric, it is being stored as a factor

```
> stuff
       food numbers other_numbers cities should_not_be_NA
1    hotdog       1             3 London           people
2 hamburger       2             4 France           places
3      Stan       3      eleventy      4             <NA>
> 
```

```
> summary(stuff)
       food   numbers  other_numbers    cities  should_not_be_NA
 hamburger:1   1:1     3       :1     4     :1   people:1
 hotdog    :1   2:1     4       :1     France:1   places:1
 Stan      :1   3:1     eleventy:1     London:1   NA's  :1
> stuff$numbers[1]
[1] 1
Levels: 1 2 3
> stuff$numbers[1] * 2
[1] NA
Warning message:
In Ops.factor(stuff$numbers[1], 2) : '*' not meaningful for factors
```

Converting from Factor to Numeric can be problematic

- Factors are pointers to a location, so a direct translation gives you a location, not the value

```
> x<-10:20
> x
 [1] 10 11 12 13 14 15 16 17 18 19 20
> as.numeric(factor(x))
 [1]  1  2  3  4  5  6  7  8  9 10 11
> as.numeric(as.character(factor(x)))
 [1] 10 11 12 13 14 15 16 17 18 19 20
>
```

Consistent data is the stage where the data are ready for analyses

- At this stage you have built a dataset that you can feel confident using
- You might want to add some more value first
  - Arithmetic between rows
  - Merge with other consistent data

Results follow from consistent data

- You can begin to explore stories that your data can tell

Best practice: save your data at every step

Let’s dig into this a bit