Data Frames II

INFO 201

Today’s Objectives

Review basics of Data Frames

Learn how to handle ***factor*** variables

Practice asking questions about datasets using Data Frames

Text Mining South Park

Some inspiration...

link

Review

Review Questions

How are ***lists*** different from ***vectors***?

How are ***single v.s. double brackets*** different when working with lists?

How could you select the number 35 from this data frame?

	age	height
1	35	71
2	36	65
3	37	60
4	38	62

```
people[1, 1]
people[1, 'age']
people$age[1]
people[1, people$height > 66]
people$age[people$height > 66]
```

Factor Variables

Levels of measurement

Nominal (labels): *Fruits*: apples, bananas, oranges, etc.

- Operations: $=, \neq$

Ordinal (ordered): *Grade of meat*: Grade A, Grade AA, Grade AAA, etc.

- Operations: $=, \neq, \leq, \geq, <, >$

Interval (arbitrary 0): *Dates*: 05/15/2012, 04/17/2015, etc.

- Operations: $=, \neq \leq, \geq, <, >, \pm$

Ratio (zero fixed): *Length*: 1 in, 1.5 in, 2 in, etc.

- Operations: $=, \neq \leq, \geq, <, >, \pm, \div$

wiki

Factors

Allow you to more efficiently store ***nominal*** (categorical) data

R stores ***integers*** and maps those to a set of ***labels***

Very useful for statistical analyses and visualization

Can be challenging to work with

```r
# Create a factor variable
x <- factor(c('Jane', 'Ella', 'Mario'))
print(x)
[1] Jane  Ella  Mario
Levels: Ella Jane Mario

# Look at structure with str function
str(x)
Factor w/ 3 levels "Ella","Jane",..: 2 1 3

# Unable to create new values!
x[1] <- 'Mike'
Warning message:
In `[<-.factor`(`*tmp*`, 1, value = "Mike") :
  invalid factor level, NA generated
```

Considerations for Factor data

```r
data <- read.csv('filename.csv', stringsAsFactors = FALSE)
```

Avoid factor data (unless intentionally using it!)

module-8

Working with Data

module 8 exercise-5

module 8 exercise-6

Upcoming...

By Tuesday: Be confident with **module 8**

Due Tuesday, 10/18 (***before class***): a3-using-data