Tidying Data Notes edX

Peter Williams

4/24/2020

- When tidying data, each row should contain 1 observation and each column should be one variable
- Columns can be added with mutate()
- Glimpse of murder before mutate():

```
##     state abb region population total
## 1 Alabama  AL  South    4779736   135
## 2  Alaska  AK   West     710231    19
## 3 Arizona  AZ   West    6392017   232
```

- Glimpse of murder after **mutate()**:
- Data sets can be filtered with the **filter()** function:
- Columns can be selected with **select(df, col1, col2, col3, coln)**

- The **pull()** function can help us to isolate integers from a single observation data frame:

```
us_murder_rate <- murders %>%
  summarize(rate = sum(total) / sum(population) * 100000)
us_murder_rate
```

```
##        rate
## 1 0.5554402
```

```
# Summarize gives us a new data frame, however, using pull() we can extract integers
us_murder_rate %>% pull(rate)
```

```
## [1] 0.5554402
```

- The **arrange()** function can be used to order a dataframe.

- if there is a tie in the order, a second argument can be used to break the tie:

```
murders %>%
  arrange(region, rate) %>%
  head(2)
```

```
##          state abb        region population total      rate
## 1 North Dakota  ND North Central     672591     4 0.5947151
## 2         Iowa  IA North Central    3046355    21 0.6893484
```

```
# This reads: arrange the murders df by region, if regions are the same arrange the region by rate
```

- Another useful organizational function is the **top_n()** function. This function is the combination of head() and arrange()

Tibbles vs. Data Frames

Essentially, a tibble is a modern day data frame, however, there are four major differences between them:
1. Tibbles display better
2. Subsets of tibbles are tibbles

3. Tibbles can have complex entries
4. Tibbles can be grouped