Instructor: Nara Yoon, Ph.D. (yoonnx@jmu.edu)

Spring 2024 | Thursdays 4-7pm, Hart 3008

Data Analytics for Managerial Leadership

WK 12: Descriptive analysis

PLAN FOR TODAY

Overview of core concepts

- Descriptive statistics
- Census ACS data with tidycensus
- Descriptive statistics with ACS data

Getting descriptive stats

LOAD DATA (iris)

```
iris <- iris

head(iris)
```

```
##   Sepal.Length Sepal.Width Petal.Length Petal.Width Species
## 1          5.1         3.5          1.4         0.2  setosa
## 2          4.9         3.0          1.4         0.2  setosa
## 3          4.7         3.2          1.3         0.2  setosa
## 4          4.6         3.1          1.5         0.2  setosa
## 5          5.0         3.6          1.4         0.2  setosa
## 6          5.4         3.9          1.7         0.4  setosa
```

summarise_at()

When to use summarise_at()?

In iris dataset – among those 5 columns,

we are interested in the summary stats of **selected columns**

- *first four* columns in the iris dataset (i.e., sepal length, sepal width, petal length, petal width)

CENTRAL TENDENCY: **MEAN**

```
iris %>%
    summarise_at(1:4,mean)
```

```
##   Sepal.Length Sepal.Width Petal.Length Petal.Width
## 1     5.843333    3.057333        3.758    1.199333
```

← Instead of using column name,
we can also use column “order” from 1 to 4

Here, we look at the first four columns

CENTRAL TENDENCY: **MEDIAN**

```
iris %>%
    summarise_at(1:4,median)
```

```
##   Sepal.Length Sepal.Width Petal.Length Petal.Width
## 1          5.8           3         4.35         1.3
```

DATA SPREAD: **RANGE**

```
iris %>%
    summarise_at(1:4,range)
```

```
##   Sepal.Length Sepal.Width Petal.Length Petal.Width
## 1          4.3         2.0          1.0         0.1
## 2          7.9         4.4          6.9         2.5
```

DATA SPREAD: **VARIANCE**

```
iris %>%
    summarise_at(1:4,var)
```

```
##   Sepal.Length Sepal.Width Petal.Length Petal.Width
## 1    0.6856935   0.1899794     3.116278   0.5810063
```

DATA SPREAD: **STANDARD DEVIATION**

```
iris %>%
    summarise_at(1:4,sd)
```

```
##   Sepal.Length Sepal.Width Petal.Length Petal.Width
## 1    0.8280661   0.4358663     1.765298   0.7622377
```

Descriptive analysis

GETTING CENSUS API KEY

Use the link to request api key

Activate your key before using it for analysis

cf) More info on getting data via API

INSTALL *tidycensus*

```
install.packages("tidycensus")
```

```
library(tidycensus)
library(tidyverse)
```

```
## ── Attaching packages ───────────────────
```

```
## ✓ ggplot2 3.4.0      ✓ purrr   0.3.5
## ✓ tibble  3.1.8      ✓ dplyr   1.0.10
## ✓ tidyr   1.2.1      ✓ stringr 1.5.0
## ✓ readr   2.1.2      ✓ forcats 0.5.2
```

LOADING ACS DATA: **load_variables()**

Today, we are interested in ACS 2015 5yr estimate

```r
variables <- load_variables(2015, "acs5", cache=T)  # default is 5yr data
head(variables)
```

```
## # A tibble: 6 × 4
##   name        label                                concept                     geogr…¹
##   <chr>       <chr>                                <chr>                       <chr>
## 1 B00001_001 Estimate!!Total                       UNWEIGHTED SAMPLE COU… block …
## 2 B00002_001 Estimate!!Total                       UNWEIGHTED SAMPLE HOU… block …
## 3 B01001_001 Estimate!!Total                       SEX BY AGE                  block …
## 4 B01001_002 Estimate!!Total!!Male                 SEX BY AGE                  block …
## 5 B01001_003 Estimate!!Total!!Male!!Under 5 years SEX BY AGE                  block …
## 6 B01001_004 Estimate!!Total!!Male!!5 to 9 years  SEX BY AGE                  block …
```

CF) CHECKING NAMES OF VARIABLES

Variable names for ACS 2015 5yr estimate
https://api.census.gov/data/2015/acs/acs5/variables.html

Among many variables, we only look at “total population (B00001_001E)” in every states in 2015

GETTING POPULATION DATA, BY STATE

```
state_pop <- get_acs(geography = "state", variables = "B00001_001E", year=2015)
```

```
## Getting data from the 2011-2015 5-year ACS
```

```
head(state_pop)
```

```
## # A tibble: 6 × 4
##   GEOID NAME       variable   estimate
##   <chr> <chr>      <chr>         <dbl>
## 1 01    Alabama    B00001_001   408374
## 2 02    Alaska     B00001_001   101973
## 3 04    Arizona    B00001_001   480058
## 4 05    Arkansas   B00001_001   248496
## 5 06    California B00001_001  2851991
## 6 08    Colorado   B00001_001   424486
```

GETTING POPULATION DATA, BY COUNTY IN ONE STATE

```
ny_county_pop <- get_acs(geography = "county", state = "New York", year=2015, variables = "B00001_001E")
```

```
## Getting data from the 2011-2015 5-year ACS
```

```
head(ny_county_pop)
```

```
## # A tibble: 6 × 4
##   GEOID NAME                         variable   estimate
##   <chr> <chr>                        <chr>         <dbl>
## 1 36001 Albany County, New York      B00001_001    24553
## 2 36003 Allegany County, New York    B00001_001    10694
## 3 36005 Bronx County, New York       B00001_001    91893
## 4 36007 Broome County, New York      B00001_001    18679
## 5 36009 Cattaraugus County, New York B00001_001    14851
## 6 36011 Cayuga County, New York      B00001_001    12390
```

Descriptive analysis

For the sake of time, we use downloaded ACS dataset in the posit site

LOAD PACKAGES

```
library(tidyverse)
library(dplyr)
library(readr)
```

LOAD DATA (acs dataset)

```
dp02 <- read_csv("ACS_DP02_2010_2015_merged.csv")
```

```
dp03 <- read_csv("ACS_DP03_2010_2015_merged.csv")
```

```
dp05 <- read_csv("ACS_DP05_2010_2015_merged.csv")
```

MERGE DATASET:
WHAT DOES OUR DATA LOOK LIKE?

```
dp <- dp02 %>%
      left_join(dp03, by=c("County code","County name","Year")) %>%
      left_join(dp05, by=c("County code","County name","Year"))
```

```
head(dp,10)  # showing the first 10 observations only
```

```
## # A tibble: 10 × 17
##    Count…¹ Count…² Year Perce…³ Perce…⁴ Per c…⁵ Perce…⁶ Popul…⁷ Media…⁸ Perce…⁹
##      <dbl> <chr>  <dbl>   <dbl>   <dbl>   <dbl>   <dbl>   <dbl>   <dbl>   <dbl>
##  1   36001 Albany  2010    37.6     6     30863    12.6  304032    38.1    13.8
##  2   36003 Allega… 2010    18.6     8.4   20058    16.5   49105    37.1    14.8
##  3   36005 Bronx   2010    17.6    12.1   17575    28.4 1365725    32.6    10.4
##  4   36007 Broome  2010    25.1     6.8   24314    15.5  200804    39.8    16.2
##  5   36009 Cattar… 2010    18.1     8.4   20824    16      80776    40.3    15.1
##  6   36011 Cayuga  2010    18.4     6.5   22959    12.1   80431    40.7    15
##  7   36013 Chauta… 2010    20.3     8.1   21033    17.1  135263    40.3    16.3
##  8   36015 Chemung 2010    20.9     7.9   23457    15.2   88708    40.6    15.5
##  9   36017 Chenan… 2010    17       8.1   22036    13.6   51045    42.2    16
## 10   36019 Clinton 2010    21.7     8.1   22660    13.3   82380    38.1    12.8
```

RENAME VARIABLES

```
dp_merge <- dp_merge %>%
            rename ("cty_code" ="County code",
                    "cty_name" = "County name",
                    "yr"= "Year",
                    "pct_bachelor" = "Percent bachelor's degree or higher (population 25 yrs and over)",
                    "pct_unemp" = "Percent unemployed",
                    "income" = "Per capita income (2010 inflation-adjusted dollars)",
                    "pct_poverty" = "Percent of people whose income in the past 12 months is below poverty level"
,
                    "pop" = "Population",
                    "age"= "Median age",
                    "pct_65" ="Percent 65 yrs and over",
                    "pct_female" = "Percent female",
                    "pct_white" = "Percent white")
```

WHAT SUMMARY STATISTICS DO WE WANT TO GET?

Central tendency

- Mean
- Median

Spread

- Variance
- Standard deviation
- Range

SUMMARIZE THE WHOLE DATASET: summary()

```
summary(dp_merge)
```

```
##     cty_code       cty_name               yr        pct_bachelor
##  Min.   :36001   Length:372         Min.   :2010   Min.   :13.10
##  1st Qu.:36031   Class :character   1st Qu.:2011   1st Qu.:19.27
##  Median :36062   Mode  :character   Median :2012   Median :23.30
##  Mean   :36062                      Mean   :2012   Mean   :25.72
##  3rd Qu.:36093                      3rd Qu.:2014   3rd Qu.:29.93
##  Max.   :36123                      Max.   :2015   Max.   :59.90
##    pct_unemp          income        pct_poverty         pop
##  Min.   : 2.300   Min.   :17575   Min.   : 5.00   Min.   :   4760
##  1st Qu.: 6.900   1st Qu.:23383   1st Qu.:11.50   1st Qu.:  51062
##  Median : 7.900   Median :25227   Median :13.55   Median :  90646
##  Mean   : 8.008   Mean   :27344   Mean   :13.77   Mean   : 313669
##  3rd Qu.: 8.900   3rd Qu.:29312   3rd Qu.:15.90   3rd Qu.: 234239
##  Max.   :15.200   Max.   :64993   Max.   :30.70   Max.   :2595259
##       age            pct_65        pct_female      pct_white
##  Min.   :29.50   Min.   :10.40   Min.   :44.10   Min.   :23.00
##  1st Qu.:38.80   1st Qu.:13.80   1st Qu.:50.00   1st Qu.:84.03
##  Median :40.70   Median :15.20   Median :50.80   Median :92.70
##  Mean   :40.35   Mean   :15.25   Mean   :50.66   Mean   :87.76
##  3rd Qu.:42.10   3rd Qu.:16.50   3rd Qu.:51.80   3rd Qu.:96.10
##  Max.   :52.10   Max.   :25.80   Max.   :54.70   Max.   :98.80
```

SUMMARIZE SELECTED VARIABLES: summarise_at()

Since we have a 6-year period longitudinal panel,
it can be more useful to have stats by year
(2010, 2011, 2012, 2013, 2014, 2015)

Here, we are interested in the statistics of selected variables
– from "% bachelors and above" to "age"

Since we are interested in only selected variables,
we use summarise_at()

MEAN

```
dp_merge %>% group_by(yr) %>% summarise_at(vars("pct_bachelor":"age"), mean) %>% round(.,2)  # rounding to show 2
 digits
```

```
## # A tibble: 6 × 7
##      yr pct_bachelor pct_unemp income pct_poverty     pop   age
##   <dbl>        <dbl>     <dbl>  <dbl>       <dbl>   <dbl> <dbl>
## 1  2010         25.1      7.12 26156.        12.9 310157.  39.7
## 2  2011         25.3      7.65 26838.        13.3 311330.  40.0
## 3  2012         25.5      8.22 27260.        13.6 312873.  40.3
## 4  2013         25.8      8.74 27608.        14   314307.  40.5
## 5  2014         26.2      8.46 28012.        14.3 316038.  40.7
## 6  2015         26.5      7.86 28190.        14.5 317309.  40.9
```

MEDIAN

```
dp_merge %>% group_by(yr) %>% summarise_at(vars("pct_bachelor":"age"), median) %>% round(.,2)
```

```
## # A tibble: 6 × 7
##      yr pct_bachelor pct_unemp income pct_poverty   pop   age
##   <dbl>        <dbl>     <dbl>  <dbl>       <dbl> <dbl> <dbl>
## 1  2010         22.4      6.9  23848.        13.0 91181  40.0
## 2  2011         22.8      7.6  24612.        13.2 91180.  40.5
## 3  2012         23.3      8    24814         13.4 91128.  40.6
## 4  2013         23.2      8.4  25328         13.6 91050  41
## 5  2014         23.4      8.1  25746.        13.8 90784  41.2
## 6  2015         23.6      7.55 25891         14.6 90342.  41.2
```

VARIANCE

```r
dp_merge %>% group_by(yr) %>% summarise_at(vars("pct_bachelor":"age"), var) %>% round(.,2)
```

```
## # A tibble: 6 × 7
##      yr pct_bachelor pct_unemp    income pct_poverty          pop   age
##   <dbl>        <dbl>     <dbl>     <dbl>       <dbl>        <dbl> <dbl>
## 1  2010         78.7      2.38 47525923.        14.4 276955666612.  9.43
## 2  2011         80.9      2.58 50116338.        15.6 280176970365. 10.2
## 3  2012         82.7      2.73 50133098.        16.6 284742716334. 10.8
## 4  2013         81.8      2.93 49240196.        17.0 289300456031  11.6
## 5  2014         82.3      3.06 50313554.        17.3 294902494535  12.4
## 6  2015         82.7      2.61 51793948.        17.1 299463233271. 12.7
```

STANDARD DEVIATION

```
dp_merge %>% group_by(yr) %>% summarise_at(vars("pct_bachelor":"age"), sd) %>% round(.,2)
```

```
## # A tibble: 6 × 7
##      yr pct_bachelor pct_unemp income pct_poverty     pop   age
##   <dbl>        <dbl>     <dbl>  <dbl>       <dbl>   <dbl> <dbl>
## 1  2010         8.87      1.54  6894.        3.79 526266.  3.07
## 2  2011         9         1.61  7079.        3.95 529317.  3.19
## 3  2012         9.09      1.65  7080.        4.07 533613.  3.28
## 4  2013         9.05      1.71  7017.        4.12 537867.  3.41
## 5  2014         9.07      1.75  7093.        4.16 543049.  3.52
## 6  2015         9.09      1.62  7197.        4.14 547232.  3.56
```

RANGE

```
dp_merge %>% group_by(yr) %>% summarise_at(vars("pct_bachelor":"age"), range) %>% round(.,2)
```

```
## # A tibble: 12 × 7
## # Groups:   yr [6]
##         yr pct_bachelor pct_unemp income pct_poverty     pop   age
##      <dbl>        <dbl>     <dbl>  <dbl>       <dbl>   <dbl> <dbl>
##  1  2010         14.3       2.3  17575         5      4908  29.6
##  2  2010         57        12.1  59149        28.4 2466782  50.4
##  3  2011         13.7       2.8  17992         5.2    4870  29.5
##  4  2011         57.7      13    61290        28.5 2486119  50.9
##  5  2012         13.1       4.9  18048         5.8    4835  29.8
##  6  2012         58.1      14.2  61951        29.3 2512740  51
##  7  2013         13.6       5.8  18171         5.8    4813  29.9
##  8  2013         58.9      15.2  62498        29.8 2539789  51.6
##  9  2014         14.2       5.3  18269         5.6    4783  29.8
## 10  2014         59.3      15    63610        30.5 2570801  52.1
## 11  2015         15.2       5.1  18456         5.3    4760  30.2
## 12  2015         59.9      14    64993        30.7 2595259  51.9
```