Explore and summarize data using R

Ken Mwai

Data exploration

- Except when a full census is taken, we collect data on a sample a population.
- Data we collect can either be categorical or continous
- Then you can derive new variables using `mutate`

How can you summarize data?

- *Summary statistics* , also known as *descriptive statistics*, is the first step in the analysis of data.
- For a continuous variable you can summarize by
  - *Measures of central tendency* : Mean,Median ,Mode
  - *Measures of dispersion*: variance, SD, mad, min,max , IQR
- We mostly report an *Measures of central tendency* with its associated *Measures of dispersion*
- For categorical variables you can do a count or frequency
  - proportions or percentages of the total number of individuals

Summary statistics for continuous {mean,sd}

- The *mean* or the *average* is the sum total of all the data point values of a numerical variable divided by the total number of data point values.
- In R we use the function `mean()` is used to calculate the mean

$$\bar{x} = \frac{1}{n}\sum_{i=1}^{n} x_i \tag{1}$$

```
x <- c(10, 11, 12,13,14,15)
mean(x)

y <- c(10, 11, 12,13,14,15 ,NA)
mean(y)
```

- With missing values you have to exclude them in the calculation

```
y <- c(10, 11, 12,13,14,15 ,NA)
## with missing it returns an error
mean(y)
```

```
## [1] NA
```

```
## exclude missing values her
mean(y , na.rm=T)
```

```
## [1] 12.5
```

- **variance** is the average difference between each value and the **mean**.
- The **standard deviation** is the square root of the **variance**
- The SD is what is mostly reported

$$\sigma = \sqrt{\frac{\sum_{i=1}^{n} \left(x_i - \bar{x}\right)^2}{n-1}} \tag{2}$$

- Why sd and not variance?
- In R we use the function `sd()` is used to calculate the mean

```
x <- c(10, 11, 12,13,14,15)
sd(x)
y <- c(10, 11, 12,13,14,15 ,NA)
sd(y , na.rm=T)
```

Summary statistics for continuous {median,min,max,mad}

- The _max__ and the **min** are the minimum and maximum values of a given variable respectively
- In R we use the function `mix()` and `max()` to find the minimum and maximum value.

```
x <- c(10, 11, 12,13,14,15)
min(x)
max(x)
y <- c(10, 11, 12,13,14,15 ,NA)
min(y , na.rm=T)
max(y, na.rm=T )
```

- The **median** is the midway value; half of the distribution lies below the **median** and half above it

$$\bar{x} = \frac{n+1}{2} th \quad value \quad of \quad ordered \quad values \tag{3}$$

- The median is great for values that are not symmetric around the mean. For normally distributed values the median will be equal to mean.
- NB: A histogram can help in checking the distribution
- In R we use the function `median()` is used to calculate the median

```
x <- c(10, 11, 12,13,14,15)
median(x)
y <- c(10, 11, 12,13,14,15 ,NA)
median(y , na.rm=T)
```

- However for the below vector a `mean` would be a misrepresentation so we would report median

```
z <- c(10, 11, 12,13,14,15,NA,100,200)
mean(z , na.rm=T)
median(z, na.rm=T)
```

- **SD** rely on the mean value to compute the average distance of scores away from the center
  - The squared differences are used, thus SD is sensitive to outliers
- **MAD** is a resistant measure of variability
  - Relies on the median as the estimate of the center of the distribution
  - Relies on the absolute difference rather than the squared difference
- **MAD** is the median of the absolute deviations from the median

$$\text{MAD} = \text{median}(|X_i - \text{median}(X)|)$$

- In R we use the function `mad()` is used to calculate the mad

```
z <- c(10, 11, 12,13,14,15,NA,100,200)
mad(z , na.rm=T)
```

Vectors are not giving lets use a dataframe!!

- For the examples here I will use the cleaned data set
- Lets I want to calculate a measure of central tendency
  1. Check the distribution then decide
  2. Check if there is any missing values
  3. Calculate the measure

Mean of age

```
hospital_df_merged %>%
  summarise(mean = mean(age, na.rm = T))
```

Mean and SD of age

```
hospital_df_merged %>%
  summarise(mean = mean(age, na.rm = T) ,
            sd = sd(age,na.rm=T))
```

Median, mad, min and max

```
hospital_df_merged %>%
  summarise(n=n(),
            median = median(age, na.rm = T) ,
            mad = mad(age,na.rm=T) ,
            min = min(age, na.rm=T) ,
            max =  max(age, na.rm=T) )
```

```
hospital_df_merged %>%
  summarise(n=n(),
            median = median(age, na.rm = T) ,
            mad = mad(age,na.rm=T) ,
            min = min(age, na.rm=T) ,
            max =  max(age, na.rm=T) ,
            ## what is IQR
            #difference between the third and the first quartile values.
            IQR = IQR(age, na.rm=T))
```

Summary statistics by groups

```
hospital_df_merged %>%
  group_by(gender) %>%
  summarise(n=n(),
            median = median(age, na.rm = T) ,
            mad = mad(age,na.rm=T) ,
            min = min(age, na.rm=T) ,
            max =  max(age, na.rm=T),
            mean = mean(age, na.rm=T) ,
            sd = sd(age,na.rm=T))
```

Summary stats using the `rstatix` package

- A friendly approach and it is `tidyverse` and `piping` friendly

```
hospital_df_merged %>%
  rstatix::get_summary_stats(age, wt_kg)
```

```
## # A tibble: 2 × 13
##   variable     n   min   max median    q1    q3   iqr   mad  mean    sd    se    ci
##   <fct>    <dbl> <dbl> <dbl>  <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>
## 1 age       6378     0    84     13     6    23    17  11.9  16.1  12.6 0.158 0.31
## 2 wt_kg     6474   -11   111     54    41    66    25  17.8  52.7  18.6 0.231 0.453
```

```
hospital_df_merged %>%
  rstatix::get_summary_stats(age, wt_kg , type = "common")
```

```
## # A tibble: 2 × 10
##   variable     n   min   max median   iqr  mean    sd    se    ci
##   <fct>    <dbl> <dbl> <dbl>  <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>
## 1 age       6378     0    84     13    17  16.1  12.6 0.158 0.31 
## 2 wt_kg     6474   -11   111     54    25  52.7  18.6 0.231 0.453
```

```
hospital_df_merged %>%
  filter(gender!="") %>%
  group_by(gender) %>%
  rstatix::get_summary_stats(age, wt_kg , type = "common") %>%
  flextable::flextable() %>%
  theme_apa()
```

gender	variable	n	min	max	median	iqr	mean	sd	se	ci
f	age	3,080.00	0.00	52.00	11.00	14.00	12.73	9.62	0.17	0.34
f	wt_kg	3,080.00	-11.00	97.00	47.00	22.00	45.87	16.88	0.30	0.60
m	age	3,087.00	0.00	84.00	17.00	20.00	19.57	14.27	0.26	0.50
m	wt_kg	3,087.00	7.00	111.00	63.00	20.00	59.61	17.67	0.32	0.62

Summary of categorical variables

- For categorical variables you can do a count or frequency
  - proportions or percentages of the total number of individuals
- The `n()`, `count`, `tally()` functions are utilized to count for categorical variables

```
hospital_df_merged %>%
  count(gender)
```

```
##   gender    n
## 1         307
## 2      f 3080
## 3      m 3087
```

```
hospital_df_merged %>%
  group_by(gender) %>%
  tally()
```

```
## # A tibble: 3 × 2
##   gender     n
##   <chr>  <int>
## 1 ""       307
## 2 "f"     3080
## 3 "m"     3087
```

```
hospital_df_merged %>%
  group_by(gender) %>%
  summarise(n=n() )
```

```
## # A tibble: 3 × 2
##   gender     n
##   <chr>  <int>
## 1 ""       307
## 2 "f"     3080
## 3 "m"     3087
```

```
hospital_df_merged %>%
  group_by(gender) %>%
  summarise(n=n()) %>%
   mutate(freq = n / sum(n),
          percent= freq*100)
```

```
## # A tibble: 3 × 4
##   gender     n   freq percent
##   <chr>  <int>  <dbl>   <dbl>
## 1 ""       307 0.0474    4.74
## 2 "f"     3080 0.476    47.6
## 3 "m"     3087 0.477    47.7
```

```
hospital_df_merged %>%
  group_by(gender) %>%
  tabyl(cough)
```

```
##  cough    n   percent
##         250 0.0386160
##     no  858 0.1325301
##    yes 5366 0.8288539
```

```
hospital_df_merged %>%
  group_by(gender) %>%
  tabyl(cough) %>%
  adorn_pct_formatting()
```

```
##  cough    n percent
##         250    3.9%
##     no  858   13.3%
##    yes 5366   82.9%
```

Apply reshape in summary

```
tbl1 <- hospital_df_merged %>%
  group_by(gender,cough) %>%
summarise(n=n() , mean_age=mean(age,na.rm=T))
tbl1
```

```
## # A tibble: 9 × 4
## # Groups:   gender [3]
##   gender cough     n mean_age
##   <chr>  <chr> <int>    <dbl>
## 1 ""     ""       10     16.4
## 2 ""     "no"     39     17.9
## 3 ""     "yes"   258     14.4
## 4 "f"    ""      127     12.6
## 5 "f"    "no"    428     11.8
## 6 "f"    "yes"  2525     12.9
## 7 "m"    ""      113     18.5
## 8 "m"    "no"    391     20.1
## 9 "m"    "yes"  2583     19.5
```

```
tbl1 %>%
pivot_wider(names_from = gender,
            values_from = n:mean_age)
```

```
## # A tibble: 3 × 7
##   cough    n_   n_f   n_m mean_age_ mean_age_f mean_age_m
##   <chr> <int> <int> <int>     <dbl>      <dbl>      <dbl>
## 1 ""       10   127   113      16.4       12.6       18.5
## 2 "no"     39   428   391      17.9       11.8       20.1
## 3 "yes"   258  2525  2583      14.4       12.9       19.5
```

gtsummary package

```
##gtsummary  package
hospital_df_merged %>%
  select(gender, outcome) %>%    # keep variables of interest
  tbl_summary(by = outcome)
```

Characteristic	****, N = 1,459[1]	**Death**, N = 2,849[1]	**Recover**, N = 2,166[1]
gender			
	80 (5.5%)	140 (4.9%)	87 (4.0%)
f	697 (48%)	1,339 (47%)	1,044 (48%)
m	682 (47%)	1,370 (48%)	1,035 (48%)

[1] n (%)

Comparison of means

- In a study of the determinants of weight, we may wish to compare the mean weight of individuals that had a cough and those that did not.
- Compares only 2 ggroups
- In general we compare the mean of $\bar{x_c ough ed}$ and $\bar{x_{!cough ed}}$
- `Ttest` is used to compare means: **However** take note of these assumptions - Independence of observations - Normality of observations - Homogeneity of variances

```
hospital_df_merged <- hospital_df_merged %>%
    filter(case_id!="") %>%
  filter(gender!="")
t.test(data=hospital_df_merged ,
       wt_kg ~ gender)
```

```
##
##  Welch Two Sample t-test
##
## data:  wt_kg by gender
## t = -31.216, df = 6153.1, p-value < 2.2e-16
## alternative hypothesis: true difference in means between group f and group m is not equal to 0
## 95 percent confidence interval:
##  -14.59925 -12.87397
## sample estimates:
## mean in group f mean in group m
##        45.86948        59.60609
```

A clean output

```
t_test_tab <- t.test(data=hospital_df_merged ,
       wt_kg ~ gender)

broom::tidy(t_test_tab)
```

```r
df_test <- hospital_df_merged %>% filter(cough!="")
t.test(data=df_test ,
       wt_kg ~ cough)
```

```
##
##  Welch Two Sample t-test
##
## data:  wt_kg by cough
## t = -1.5934, df = 1078.8, p-value = 0.1114
## alternative hypothesis: true difference in means between group no and group yes is not equal to 0
## 95 percent confidence interval:
##  -2.5481709  0.2642594
## sample estimates:
##  mean in group no mean in group yes
##          51.82418          52.96613
```

gtsummary package

```
hospital_df_merged %>%
  filter(cough!="") %>%   ## remove missing cough
  select(wt_kg, cough) %>%             # keep variables of interest
  tbl_summary(                              # produce summary table
    statistic = wt_kg ~ "{mean} ({sd})", # specify what statistics to show
    by = cough) %>%                        # specify the grouping variable
  add_p(wt_kg ~ "t.test")                # specify what tests to perform
```

Characteristic	**no**, N = 819[1]	**yes**, N = 5,108[1]	**p-value**[2]
wt_kg	52 (19)	53 (19)	0.11

[1] Mean (SD)

[2] Welch Two Sample t-test

Comparison of proportions