ECON 3640–001

Problem Set 1 - Solutions

Marcio Santetti

Spring 2022

Problem 1

With your own words, define and give an example of each of the following statistical terms.

(a) Population;

(b) Sample;

(c) Parameter;

(d) Statistic.

ANSWER: Suppose we want to compute the average course evaluations (ranging from 0 to 10) from *all* students at the University of Utah. This is our **population** of interest. However, it may be very costly (in terms of time, budget, and response rates) to obtain replies from each student. Thus, we may select a **sample** of 200 students from each school/department, and run a survey with them. The population **parameter** we are interested in is the average course evaluation, but what we end up with is with this sample's **statistic**, i.e., their course evaluations. Through statistical inference, we may evaluate whether this was an apporpriate sample or not for such purpose.

Problem 2

A student received the following letter grades on the 14 quizzes she took during a semester: *A, A-, B+, A-, A, C, A, A, A-, B, B+, C, A*, and *A*.

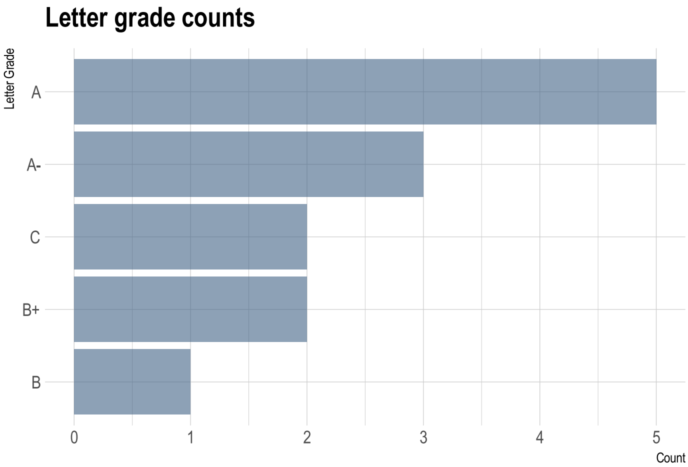
(a) What is the `type` of these data? Explain.

ANSWER: These are **categorical** data.

(b) Use your best judgment to *illustrate* these data.

ANSWER:

```r
library(tidyverse)
library(hrbrthemes)

grades <- tibble(
  letter_grade = c("A", "A-", "B+", "A-", "A", "C", "A", "A", "A-", "B", "B+", "C", "A")
)

grades %>%
  count(letter_grade, sort = TRUE) %>%
  mutate(letter_grade = fct_reorder(letter_grade, n)) %>%
  ggplot(aes(y = letter_grade, x = n)) +
  geom_col(fill = "#426386", alpha = 0.6) +
  labs(x = "Count", y = "Letter Grade", title = "Letter grade counts") +
  theme_ipsum()
```



Problem 3

A sample of 12 people was asked how much change they had in their pockets and wallets. The responses (in cents) were:

52 25 15 0 104 44 60 30 33 81 40 5

(a) Determine the *mean*, *median* and *mode* for these data. Show your calculations.

ANSWER: Sample **mean**:

$$\bar{x} = \frac{\sum_{i=1}^{12} x_i}{n} = \frac{52 + 25 + \ldots + 5}{12} = 40.8 \text{ cents}$$

- Sample **median**:

```
change ← tibble(
  how_much = c(52, 25, 15, 0, 104, 44, 60, 30, 33, 81, 40, 5)
)

change %>%
  count(how_much) # putting values in ascending order.
```

```
## # A tibble: 12 × 2
##    how_much     n
##       <dbl> <int>
##  1        0     1
##  2        5     1
##  3       15     1
##  4       25     1
##  5       30     1
##  6       33     1
##  7       40     1
##  8       44     1
##  9       52     1
## 10       60     1
## 11       81     1
## 12      104     1
```

$$\text{Sample median} = \frac{33 + 40}{2} = 36.5 \text{ cents}$$

- Sample **mode**: No value is repeated, Thus, this data set has **no mode**.

(b) Determine the first, second, and third *quartiles* of these data.

ANSWER:

$$L_{25} = (n+1)\frac{25}{100} = (13) \cdot 0.25 = 3.25$$

The 1st quartile lies between the 3rd (15) and 4th (25) positions (with the values in ascending order). More specifically, it lies on the 3rd location plus one-quarter of the distance between the 3rd and the 4th. Let's compute this additional distance:

$$0.25 \times (25 - 15) = 2.5$$

Thus, the 1st quartile is 15 + 2.5 = **17.5** cents.

$$L_{50} = (n+1)\frac{50}{100} = (13) \cdot 0.5 = 6.5$$

The second quartile lies between the 6th and 7th positions, which are 33 and 40, respectively.

$$0.5 \times (40 - 33) = 3.5$$

Thus, the second quartile (Q2) is 33 + 3.5 = **36.5** cents. Not surprisingly, it equals the sample **median**.

$$L_{75} = (n+1)\frac{75}{100} = (13) \cdot 0.75 = 9.75$$

The third quartile lies between the 9th and 10th positions, which are 52 and 60, respectively.

$$0.75 \times (60 - 52) = 6$$

Thus, the third quartile (Q3) is 52 + 6 = **58** cents.

IIMPORTANT: R locates percentiles using a different **method**. Above, we used the methodology seen in class. But you may try the following in R:

```
change %>%
  summarize(quartiles = quantile(how_much))
```

```
## # A tibble: 5 × 1
##   quartiles
##       <dbl>
## 1       0
## 2      22.5
## 3      36.5
## 4      54
## 5     104
```

The median value remains the same. However, for Q1 and Q3, R uses a different methodology, consisting of subtracting the excess distances (2.5 for Q1 and 6 for Q3, as above) from the final position.

Thus, the `quantile` function gives 22.5 as the first quartile because it takes the 4th position, 25, and subtracts the excess distance, 2.5, from it, giving us 22.5.

Similarly, the `quantile` function gives 54 as the third quartile because it takes the 10th position, 60, and subtracts the excess distance, 6, from it, giving us 54. **Both methodologies are correct, and you may choose whichever you prefer**.

(c) Compute the *variance* and *standard deviation* for these data. Show your calculations.

ANSWER:

- Sample variance:

$$s_x^2 = \frac{1}{n-1}\left[\sum_{i=1}^{n} x_i^2 - \frac{\left(\sum_{i=1}^{n} x_i\right)^2}{n}\right] = \frac{1}{12}\left[(0)^2 + (5)^2 + \ldots + (104)^2 - \frac{(489)^2}{13}\right] = 923.11 \text{ cents}^2$$

- Sample standard deviation:

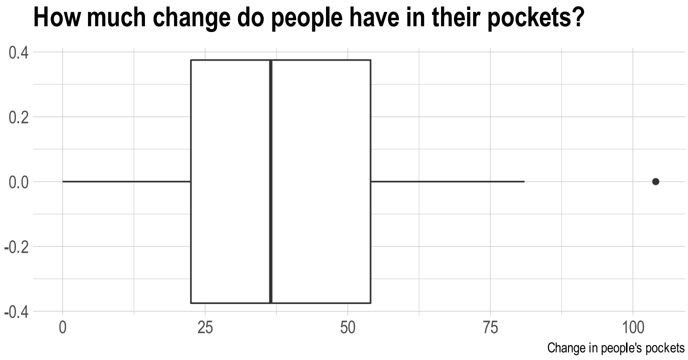
$$s_x = \sqrt{s_x^2} = \sqrt{923.11} = 30.38 \text{ cents}$$

(d) Draw a *box plot* for these data.

ANSWER:

Notice that the first and second quartiles are obtained using the **second methodology** explained in part (b)'s answer.

```
change %>%
  ggplot(aes(x = how_much)) +
  geom_boxplot() +
  labs(x = "Change in people's pockets",
       title = "How much change do people have in their pockets?") +
  theme_ipsum()
```



Problem 4

From the `wooldridge` package, load the `mroz` data set. The data come from `Mroz (1987)`. Just as with any data set from this package, its documentation contains all variable names and descriptions. Simply google "wooldridge R package" and you will find it.

First, transform it into a `tibble`. Just follow the code below:

```
library(tidyverse)
library(wooldridge)

data("mroz")

mroz_tibble <- mroz %>% as_tibble()
```

Then, answer the following questions:

(a) What is the mode (i.e., most frequent) educational attainment— in years —in this sample?

```
mroz_tibble %>%
  count(educ, sort=TRUE)
```

```
## # A tibble: 13 × 2
##     educ     n
##    <int> <int>
##  1    12   381
##  2    16    57
##  3    14    51
##  4    17    46
##  5    10    44
##  6    13    44
##  7    11    43
##  8     8    30
##  9     9    25
## 10    15    14
## 11     7     8
## 12     6     6
## 13     5     4
```

We can see that **12** years of education is the educational attainment that gets repeated the most. Therefore, this is the sample mode.

ANSWER:

(b) Select the first five rows of this data set and manually compute the covariance, correlation coefficient, and coefficient of determination (R^2) between educational attainment (`educ`) and hourly earnings (`wage`). Interpret these results. **Hint**: `head(5)`.

```
mroz_tibble %>%
  select(educ, wage) %>%
  head(5) # selecting the first five rows.
```

```
## # A tibble: 5 × 2
##    educ  wage
##   <int> <dbl>
## 1    12  3.35
## 2    12  1.39
## 3    12  4.55
## 4    12  1.10
## 5    14  4.59
```

ANSWER:

- Sample **covariance**:

$$s_{xy} = \frac{1}{n-1}\left[\sum_{i=1}^{n} x_i y_i - \frac{\sum_{i=1}^{n} x_i \sum_{i=1}^{n} y_i}{n}\right] = \frac{1}{4}\left[[(12 \times 3.35) + (12 \times 1.39) + \ldots + (14 \times 4.59)] - \frac{62 \times 15}{5}\right] = 0.798$$

- Sample **correlation**:

$$r = \frac{s_{xy}}{s_x s_y} = \frac{0.798}{(0.894) \times (1.68)} = 0.532$$

- Sample **R**^{2}:

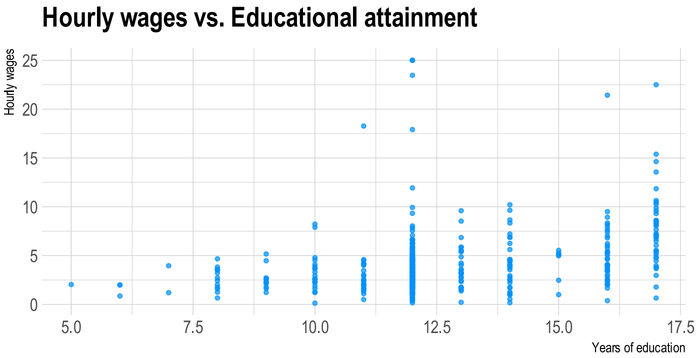
$$R^2 = (0.532)^2 = 0.283 \text{ or } 28.3\%$$

Interpreting these 3 results: For these 5 observations, educational attainment and wages show a positive association (given by the *covariance*), showing some linearity (given by the positive *correlation* coefficient). 28.3% of variations (changes) in wages are explained by variations (changes) in educational attainment.

(c) Use the most appropriate visual descriptive technique to illustrate the association between the two variables from part (b). Do not forget to make your plot informative to a wide audience (i.e., label your axes and give it a nice title).

ANSWER: Using the entire data set…

```
mroz_tibble %>%
  ggplot(aes(x = educ, y = wage)) +
  geom_point(color = "#0091fa", alpha = 0.7, size = 1) +
  labs(x = "Years of education",
       y = "Hourly wages",
       title = "Hourly wages vs. Educational attainment") +
  theme_ipsum()
```



Problem 5

Still using the `mroz` data set, run the following:

```
mroz_tibble %>%
  select(inlf, hours, huswage) %>%
  head()
```

```
## # A tibble: 6 × 3
##    inlf hours huswage
##   <int> <int>   <dbl>
## 1     1  1610    4.03
## 2     1  1656    8.44
## 3     1  1980    3.58
## 4     1   456    3.54
## 5     1  1568   10   
## 6     1  2032    6.71
```

Notice that the `inlf` variable is defined here as an integer (`int`), but what it is actually doing is serving as a *binary* indicator, which equals 1 if the woman interviewed is in the labor force, and 0 if not. Thus, if we want to use this variable for the upcoming plot, we should transform it into a factor (`fct`) class object.

Just do the following:

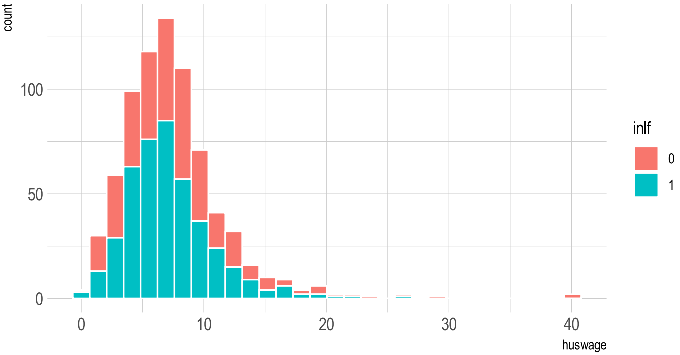
```
mroz_tibble <- mroz_tibble %>%
  mutate(inlf = as_factor(inlf))
```

and check out whether the variable is now a factor. The `as_factor()` function is part of the `tidyverse`.

(a) Draw a histogram of husband wages (`huswage`), comparing the difference between whether the interviewee is in the labor force or not. **Hint**: you may either use the `fill` argument within the `aes()` environment, or use the `facet_wrap()` function to do that. Interpret your results.

ANSWER:

```
mroz_tibble %>%
  ggplot(aes(x = huswage)) +
  geom_histogram(aes(fill = inlf), color = "white") +
  theme_ipsum()
```



This histogram distinguishes between women that are in the labor force and those who are not. Even though the distribution of husband wages does not change across groups, we observe a larger count of data for families where women are not in the labor force. However, the average wage is basically the same for both groups.

(b) In this sample, how many interviewees are in the labor force? How many aren't?

ANSWER:

```
mroz_tibble %>%
  count(inlf, sort=TRUE)
```

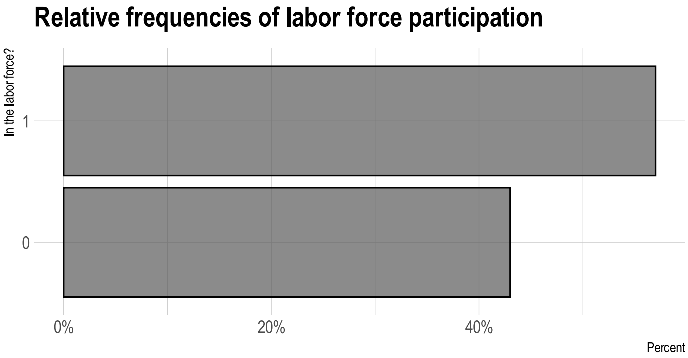
```
## # A tibble: 2 × 2
##   inlf      n
##   <fct> <int>
## 1 1       428
## 2 0       325
```

428 individuals are in the labor force, while 325 are not.

(c) From your answer to part (b), illustrate the relative frequencies (i.e., %) for each case either with a bar or pie chart.

ANSWER:

```
library(scales)

mroz_tibble %>%
  count(inlf, sort=TRUE) %>%
  mutate(lf_share = round(n/sum(n), 2)) %>% # the "round()" function rounds the decimal points
  ggplot(aes(y = inlf, x = lf_share)) +
  geom_col(color = "black", alpha = 0.7) +
  scale_x_continuous(labels = percent_format()) +
  labs(x = "Percent",
       y = "In the labor force?",
       title = "Relative frequencies of labor force participation") +
  theme_ipsum()
```



Problem 6

During these pandemic times, you have probably come across the `moving average` term. It simply consists of calculating a mean that is adjusted over a specified time window. For instance, a 7-day moving (or rolling) average computes the mean value of a variable over the previous 7 days, and it gets adjusted as time moves on.

This application allows us to smooth out short-term oscillations in a data set. Applying this in R is very simple, and we'll get there soon.

(a) First, import the `covid-cases-22.csv` data set into your R environment. Call it `covid_cases`.

ANSWER:

```
covid_cases ← read_csv("covid-cases-22.csv")
```

(b) Now, we need to convert the `period` column into a `date` object. It will be imported as a character (`chr`). You need to use the `lubridate` package, specifically built to deal with dates and times. Check out the code below:

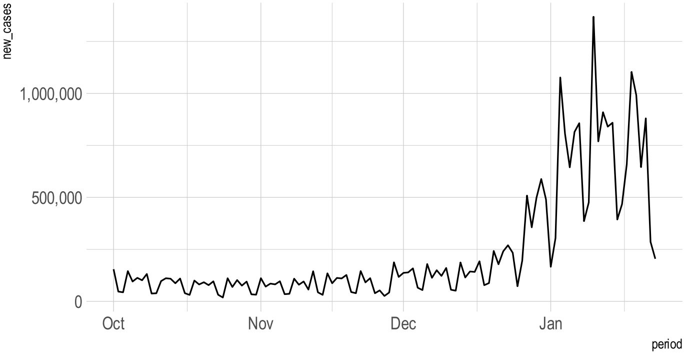
```
library(lubridate) # make sure to have it installed first.

covid_cases ← covid_cases %>%
  mutate(period = mdy(period))
```

The `mdy` function simply converts a character string defined by `d`ay-`m`onth-`y`ear into a `date` object.

(c) Find out how to plot the `new_cases` variable over time using `ggplot2`.

ANSWER:

```
covid_cases %>%
  ggplot(aes(x = period, y = new_cases)) +
  geom_line() +  # for line plots.
  scale_y_continuous(labels = comma) +
  theme_ipsum()
```



(d) To calculate moving averages, one may use the `RcppRoll` package. Its `roll_mean()` function does the job. So, create a new column in your data set, called `new_cases_ma`, defined by the 14-day moving average for the `new_cases` series. You will use the `roll_mean()` function and 3 arguments: `n`, which is the number of days you want your moving-average window to be; `align`, which you will set equal to "right"; and `fill`, which you will set to `NA`. The latter guarantees that the first values (for which you will not be able to calculate the moving average) will be filled out with `NA` values.

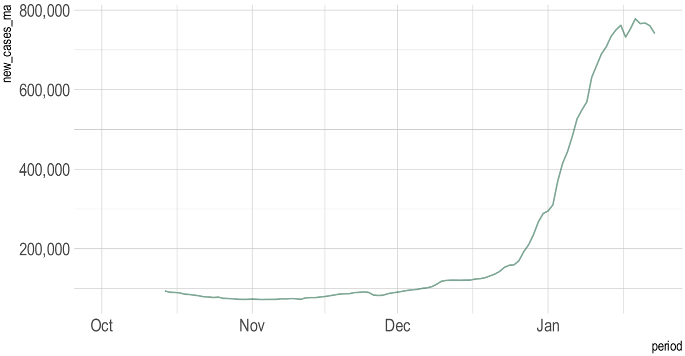
ANSWER:

```r
library(RcppRoll)

covid_cases <- covid_cases %>%
  mutate(new_cases_ma = roll_mean(new_cases, n = 14, align = "right", fill = NA))
```

(e) Lastly, plot this new variable from part (d) over time and compare it with your plot from part (c).

ANSWER:

```r
covid_cases %>%
  ggplot(aes(x = period, y = new_cases_ma)) +
  geom_line(color = "#317256", alpha = 0.6) +
  scale_y_continuous(labels = comma) +
  theme_ipsum()
```


new_cases_ma
Oct
Nov
Dec
Jan
period