API-201 ABC REVIEW SESSION #9

Friday, December 2

Table of Contents

Exercise 1 - Dplyr functions

Download the data using this link.

0. Upload the Excel file WEO-2018.xlsx to Google Colab and run the next lines of code to load the data and examine the first rows of the dataset.

```
library(tidyverse)
library(readxl)
weo_data <- read_excel(path = "WEO-2018.xlsx", sheet = 1)
head(weo_data)
```

country	continent	pop_1992	pop_1993	pop_1994	pop_1995	pop_1996	pop_1997	pop_1998
<chr>	<chr>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>
Afghanistan	Asia	NA	NA	NA	NA	NA	NA	NA
Albania	Europe	3.217	3.201	3.137	3.141	3.168	3.148	3.129
Algeria	Africa	26.271	26.894	27.496	28.060	28.566	29.045	29.507
Angola	Africa	13.459	13.863	14.279	14.707	15.148	15.603	16.071
Antigua and Barbuda	North America	0.062	0.063	0.065	0.067	0.068	0.070	0.072
Argentina	South America	33.420	33.917	34.353	34.779	35.196	35.604	36.005

1. Imagine that you wanted to show the five countries with the highest GDP per capita in 2017. Which of the following code snippets would accomplish that?

a. weo_data %>%
mutate(rgdppc_2017=rgdp_2017/pop_2017) %>%
arrange(desc(rgdppc_2017)) %>%
filter(country,continent,rgdppc_2017) %>%
head(5)

b. weo_data %>%
mutate(rgdppc_2017=rgdp_2017/pop_2017) %>%
arrange(desc(rgdppc_2017)) %>%
select(country,continent,rgdppc_2017) %>%
head(5)

c. weo_data %>%
summarise(rgdppc_2017=rgdp_2017/pop_2017) %>%
arrange(desc(rgdppc_2017)) %>%
select(country,continent,rgdppc_2017) %>%
head(5)

d. weo_data %>%
summarise(rgdppc_2017=rgdp_2017/pop_2017) %>%
select(desc(rgdppc_2017)) %>%
filter(country,continent,rgdppc_2017) %>%
head(5)

```
# Your answer here!

# START

weo_data %>%
    mutate(rgdppc_2017=rgdp_2017/pop_2017) %>%
    arrange(desc(rgdppc_2017)) %>%
    select(country,continent,rgdppc_2017) %>%
    head(5)
# Option b is correct.

# END
```

A tibble: 5 × 3

country	continent	rgdppc_2017

2. Suppose you want to calculate GDP growth between 2017 and 2018 and create a dataset called `growth_data` that includes the country, the continent, and the GDP growth rate. Which of the following code snippets would accomplish that?

a. weo_data %>%
summarise(rgdp_growth=rgdp_2018/rgdp_2017 - 1) %>%
select(country,continent,rgdp_growth)
b. weo_data %>%
mutate(rgdp_growth=rgdp_2018/rgdp_2017 - 1) %>%
filter(country,continent,rgdp_growth)
c. weo_data %>%
mutate(rgdp_growth=rgdp_2018/rgdp_2017 - 1) %>%
select(country,continent,rgdp_growth)
d. growth_data <- weo_data %>%
mutate(rgdp_growth=rgdp_2018/rgdp_2017 - 1) %>%
select(country,continent,rgdp_growth)

```
# Your answer here!

# START

growth_data <- weo_data %>%
    mutate(rgdp_growth=rgdp_2018/rgdp_2017 - 1) %>%
    select(country,continent,rgdp_growth)
# Option d is correct.

# END
```

3. Suppose you want to show the five countries in Europe with highest GDP growth between 2017-2018. Which of the following snippets would accomplish that?

a. growth_data %>%
select(continent == "Europe") %>%
arrange(desc(rgdp_growth)) %>%
head(5)
b. growth_data %>%
filter(continent == "Europe") %>%
arrange(desc(rgdp_growth)) %>%
head(5)
c. growth_data %>%
filter("Europe") %>%
arrange(desc(rgdp_growth)) %>%

head(5)
d. growth_data %>%
select("Europe") %>%
arrange(desc(rgdp_growth)) %>%
head(5)

```
# Your answer here!

# START

growth_data %>%
    filter(continent == "Europe") %>%
    arrange(desc(rgdp_growth)) %>%
    head(5)
# Option b is correct.

# END
```

A tibble: 5 × 3

country	continent	rgdp_growth
<chr>	<chr>	<dbl>
Malta	Europe	0.05716901
Romania	Europe	0.05100430
Ireland	Europe	0.04506535
Georgia	Europe	0.04458567
Turkey	Europe	0.04410933

4. Suppose you want to use this new dataset to compute the average growth rate by continent and then sort the continents in descending order. Which of the following code snippets would accomplish that?
a. growth_data %>%
select(continent) %>%
summarise(av_growth = mean(rgdp_growth)) %>%
arrange(desc(av_growth))
b. growth_data %>%
summarise(av_growth = mean(rgdp_growth)) %>%
arrange(desc(av_growth))
c. growth_data %>%
group_by(continent) %>%
summarise(av_growth = mean(rgdp_growth)) %>%
arrange(desc(av_growth))
d. growth_data %>%
group_by(continent) %>%

mutate(av_growth = mean(rgdp_growth)) %>%
arrange(desc(av_growth))

```
# Your answer here!

# START

growth_data %>%
    group_by(continent) %>%
    summarise(av_growth = mean(rgdp_growth)) %>%
    arrange(desc(av_growth))

# Option c is correct.

# END
```

A tibble: 6 × 2

continent	av_growth
<chr>	<dbl>
Asia	0.03922983
Africa	0.03786809
Europe	0.03157609
Oceania	0.02384227
North America	0.01809262
South America	0.01444411

Exercise 2: Bayes rule

Suppose that your team is interested in testing the performance of a new spam filter. This filter analyzes incoming emails and quarantines those that are assessed as potential spam. Emails can either be spam (S) or not spam (NS), and the filter can either move emails to a separate folder for quarantine (Q) or leave them in the main inbox folder (NQ). The team is excited to announce that 75% of spam emails are placed in quarantine. However, they also find that 10% of not spam emails are held in quarantine. Suppose that the typical proportion of email that is spam is 5%.

a. Somone in your team says: given that such high proportion of spam emails are held in quarantine, most emails moved to quarantine have to be spam. Do you agree with this claim?

`Your answer here!`

START

Below is the information that we were given:

$$P(Q|S) = 0.8$$
$$P(NQ \mid S) = 1 - 0.8 = 0.2$$
$$P(Q \mid NS) = 0.1$$
$$P(NQ \mid NS) = 1 - 0.1 = 0.9$$
$$P(S) = 0.05$$
$$P(NS) = 1 - 0.05 = 0.95$$

To answer this question, we need to calculate $P(S|Q)$. We will use two approaches to solve this question: the first one is filling a 2x2 probability table and the second one consists on applying the Bayes Rule formula. Let's start with approach 1. We want to fill the table below with joint probabilities:

	Spam	Not Spam	Row sums
Q	P(Q & S)	P(Q & NS)	P(Q)
NQ	P(NQ & S)	P(NQ & NS)	P(NQ)
Col sums	P(S)	P(NS)	1

Suppose there are 1,000 emails. We know that 50 of them are spam. Out of those 50, 40 are held in quarantine. There are 950 not spam emails, and we know that 10% of them are held in quarantine, which is 95.

	Spam	Not Spam	Row sums
Q	40	95	135
NQ	10	855	865
Col sums	50	950	1000

What we just did is equivalent to calculating all 4 joint probabilities:

$$P(Q\&S) = P(Q|S) * P(S) = 0.8 * 0.05 = 0.04$$
$$P(Q\&NS) = P(Q|NS) * P(NS) = 0.1 * 0.95 = 0.095$$
$$P(NQ\&S) = P(NQ|S) * P(S) = (1 - 0.8) * 0.05 = 0.01$$
$$P(NQ\&NS) = P(NQ|NS) * P(NS) = 0.9 * 0.95 = 0.855$$

The question tells us that the filter placed the email in quarantine, so we know this case falls in either the top left or top right quadrants of the table. Now we calculate the probability of spam, given that the filter put the email in quarantine:

$$P(S|Q) = \frac{P(Q\&S)}{P(Q)}$$
$$= \frac{\text{top left quadrant}}{\text{top left quadrant} + \text{top right quadrant}}$$
$$= \frac{40}{40 + 95} = 0.296$$

Alternatively, we can directly apply Bayes Rule (approach 2):

$$
\begin{aligned}
P(S \mid Q) &= \frac{P(Q\&S)}{P(Q)} \\
&= \frac{P(Q\&S)}{P(Q\&S) + P(Q\&NS)} \\
&= \frac{P(Q \mid S) * P(S)}{P(Q \mid S) * P(S) + P(Q \mid NS) * P(NS)} \\
&= \frac{0.8 * 0.05}{0.8 * 0.05 + (0.1) * 0.95} = 0.296
\end{aligned}
$$

The claim of your teammate is incorrect.

END

b. Calculate the probability that an email is not spam if the email was not placed in quarantine: P(NS|NQ).

`Your answer here!`

START

Using the information from the table, we calculate:

$$
\begin{aligned}
P(NS|NQ) &= \frac{P(NQ\&NS)}{P(NQ)} \\
&= \frac{\text{bottom right quadrant}}{\text{bottom left quadrant} + \text{bottom right quadrant}} \\
&= \frac{855}{10 + 855} = 0.988
\end{aligned}
$$

END

Exercise 3: Discrete random variables

The number of tornadoes in a year is a random variable with the probability distribution given below.

Number of tornadoes	Probability
0	0.850
1	0.100
2	0.030
3	0.015
4	0.005
5 or more	0

1. Calculate the expected number of tornadoes in a year.

`Your answer here!`

START

For a discrete random variable X, its expected value can be calculated as $E[X] = \sum x \cdot p(x)$. In this case, the expected number of tornadoes in a year is equal to 0.225.

END

2. What is the expected number of tornadoes in 10 years?

`Your answer here!`

START

Let X_t be a discrete random variable for the number of tornadoes in year *t*. Then the number of tornadoes in 10 years is $Y = X_1 + X_2 + \ldots + X_{10}$. By **linearity of expectations**, the expectation of a sum is equal to the sum of expectations:

$$\begin{aligned} E[Y] &= E[X_1 + X_2 + \ldots + X_{10}] \\ &= E[X_1] + E[X_2] + \ldots + E[X_{10}] \\ &= 10 \times E[X_t] \\ &= 10 \times 0.225 \\ &= 2.25 \end{aligned}$$

We would expect the city to be affected by 2.25 tornadoes in 10 years.

END

3. What is the probability of being affected by at least one tornado in 10 years?

Your answer here!

START

The probability of any tornadoes in 10 years is equal to 1 minus the probability of no tornadoes in 10 years, which is easier to calculate.

$$\Pr(\text{Any Tornadoes}) = 1 - \Pr(\text{No Tornadoes})$$

If we assume that tornado risk is independent across years (e.g. exposure to a tornado in year 1 is unrelated to exposure in year 2, 3, 4, ...) then we can write the probability of no tornadoes in 10 years as the following:

$$\Pr(\text{No Tornadoes}) = 0.85^{10} = 0.197$$

From the first equation, the answer is 80%:

$$\Pr(\text{Any Tornadoes}) = 1 - 0.197 = 0.803$$

END

4. These calculations can also be obtained in R through simulation. Read the code below.

```
# Expected number of tornadoes in ayera
num_tornadoes <- c(0, 1, 2, 3, 4, 5)
p <- c(.85, .1, .03, .015, .005, 0)
sum(num_tornadoes*p)
```

0.225

```
# Expected number of tornadoes in a year (simulation)
mean(sample(num_tornadoes,10000, prob = p, replace = TRUE))
```

0.2344

```
# Expected number of tornadoes in 10 years (simulation)
mean(replicate(10000,sum(sample(num_tornadoes,10, prob = p, replace = TRUE))))
```

2.2439

```
# Prob at least one tornado in 10 years
mean(replicate(10000,sum(sample(num_tornadoes,10, prob = p, replace = TRUE)))>0)
```

0.8008

Exercise 4: Normal distribution + decision tree

Suppose a risk-neutral farmer is considering insuring against rainfall conditions. Rainfall is normally distributed with mean 500mm and standard deviation 50mm. If rainfall is less than 450mm or greater than 550mm, the farmer's crop will fail.

a. What is the probability the farmer's crop will fail?

Hint: Approximately 68% of the area under the normal distribution curve falls within 2 standard deviations of the mean.

Your answer here!

START

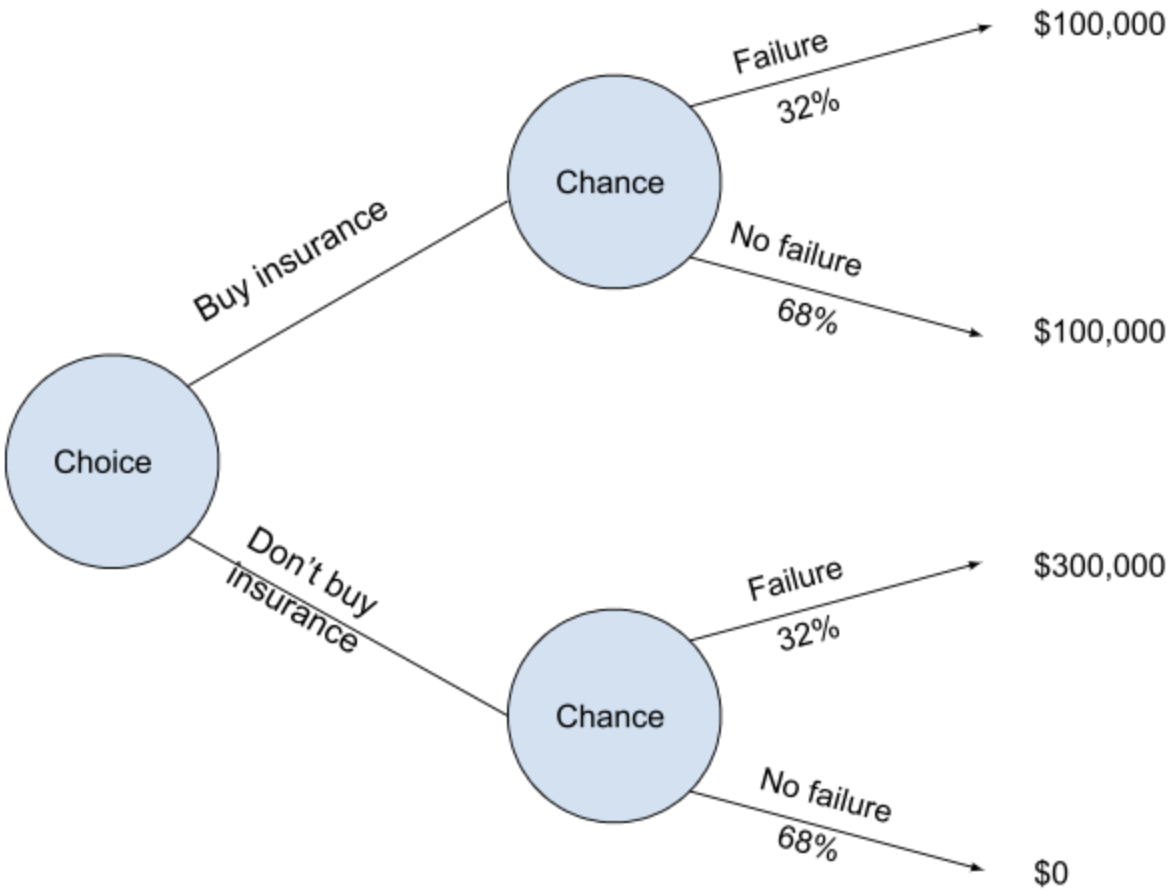
The crop will fail if rainfall is less than 450mm or greater than 550mm. 68% of outcomes fall between the points of failure. Thus the probability of crop failure is $1 - 0.68 = 0.32$.

END

b. Now suppose the cost of crop failure is \$300,000. Insurance costs \$100,000 to buy. If crops fail, the farmer is reimbursed the entire cost of crop failure. Draw out a decision tree. Should she buy insurance?

Your answer here!

START



She should not buy insurance.

The expected cost of insurance is \$100,000 because she pays that much whether crops fail or not. The expected cost of uninsurance is $\$300,000 \times \Pr(Fail) = \$96,000$. The expected cost of insurance is higher than uninsurance, so on average, she will incur less cost by remaining unisured.

END

c. What is the highest price the farmer would be willing to pay for insurance?

Your answer here!

START

Buy if and only if the cost of insurance is *less* than the expected cost of uninsurance: X < 96,000.

END

Exercise 5 - Dplyr functions (optional)

Just like we did in the R primers, for this exercise we are going to use full baby name data provided by the SSA. This includes all names with at least 5 uses. `babynames` is a data frame with five variables: `year`, `sex`, `name`, `n` and `prop` (n divided by total number of applicants in that year, which means proportions are of people of that sex with that name born in that year).

```
# install.packages("babynames")
library(tidyverse)
library(babynames)
```

```
── Attaching packages ─────────────────────────────────────── tidyverse 1.3.1 ──

✔ ggplot2 3.3.6     ✔ purrr   0.3.4
✔ tibble  3.1.7     ✔ dplyr   1.0.9
✔ tidyr   1.2.0     ✔ stringr 1.4.0
✔ readr   2.1.2     ✔ forcats 0.5.1

── Conflicts ────────────────────────────────────────── tidyverse_conflicts() ──
✖ dplyr::filter() masks stats::filter()
✖ dplyr::lag()    masks stats::lag()
```

1. Examine the first ten rows of data. What is the unit of analysis of this dataset?

```
# Your answer here!

# START

head(babynames, 10)

# END
```

A tibble: 10 × 5

year	sex	name	n	prop
1880	F	Elizabeth	1939	0.01986579

2. What were the five most popular girl names in 2000? Report the number of babies.

Hint: these are the steps to follow:

a. Filter babynames to just girls born in 2000.

b. Select the name and n columns from the result.

c. Arrange those columns so that the most popular names appear near the top.

```
# Your answer here!

# START

babynames %>%
    filter(year == 2000 & sex == "F") %>%
    select(name, n) %>%
    arrange(desc(n)) %>%
    head(5)

# END
```

A tibble: 5 × 2

name	n
<chr>	<int>
Emily	25953
Hannah	23080
Madison	19967
Ashley	17997
Sarah	17697

3. What were the five most popular boy names in 2017? Report the proportions.

```
# Your answer here!

# START

babynames %>%
    filter(year == 2017 & sex == "M") %>%
    select(name, prop) %>%
    arrange(desc(prop)) %>%
    head(5)

# END
```

A tibble: 5 × 2

name	prop
<chr>	<dbl>
Liam	0.00953909
Noah	0.00933433
William	0.00759134

4. Plot the evolution of the popularity of the name *Arya*.

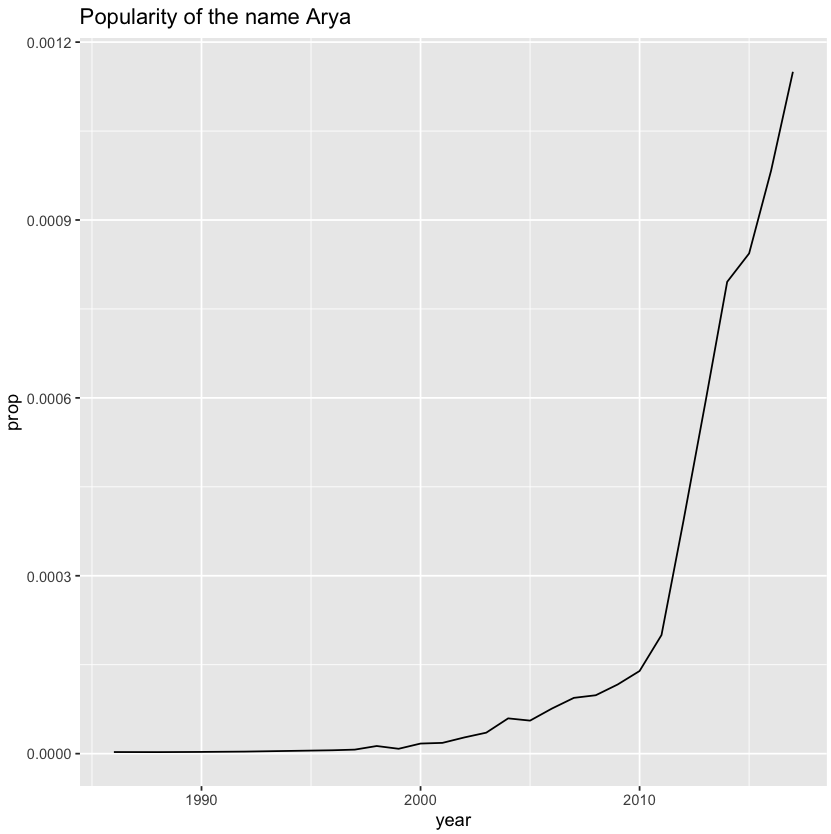
```
# Your answer here!

# START

arya_name <- babynames %>%
  filter(name == "Arya", sex == "F") %>%
  select(year, prop)

ggplot(arya_name) +
    geom_line(aes(x = year, y = prop)) +
    labs(title = "Popularity of the name Arya")

# END
```



5. Calculate for your name: (1) The total number of children that had your name, (2) The maximum number of kids with that name in a single year, (3) The mean number of kids named that per year, and (4) The first time a kid was named that in this dataset.

```
# Your answer here!

# START

babynames %>%
  filter(name == "Guillermo", sex == "M") %>%
  summarise(total = sum(n), max=max(n), mean=mean(n), year=first(year))

#END
```

A tibble: 1 × 4

total	max	mean	year
<int>	<int>	<dbl>	<dbl>
29698	693	253.8291	1885

6. Our first measure of *popularity* is the total number of children of a single gender given a name. Display the ten more popular names and the proportion of times a kid received that name.

```
# Your answer here!

# START

babynames %>%
    group_by(sex,name) %>%
    summarise(total = sum(n)) %>%
    mutate(prop=total/sum(total)) %>%
    arrange(desc(total)) %>%
    head(10)

#END
```

`summarise()` has grouped output by 'sex'. You can override using the `.groups` argument.

A grouped_df: 10 × 4

sex	name	total	prop

7. Under the second definition, a name is *popular* if it consistently ranks among the top names from year to year. Display the ten most popular names.

```
# Your answer here!

# START

babynames %>%
    group_by(year, sex) %>%
    mutate(rank = min_rank(desc(prop))) %>%
    group_by(name, sex) %>%
    summarise(score = median(rank)) %>%
    arrange(score) %>%
    head(10)

# END
```

`summarise()` has grouped output by 'name'. You can override using the `.groups` argument.

A grouped_df: 10 × 3

name	sex	score
<chr>	<chr>	<dbl>
Mary	F	1.0
James	M	3.0
John	M	3.0
William	M	4.0
Robert	M	6.0
Michael	M	7.5
Charles	M	9.0
Elizabeth	F	10.0
Joseph	M	10.0
Thomas	M	11.0