Final Report of Internship Program 2024

On

"Analyze Death Age Difference of Right Handers with Left Handers"

MEDTOUREASY

2nd Feb, 2024

Prepared By-

Monica Upadhyay

Acknowledgment

This internship opportunity that I had with MedTourEasy was a great opportunity and shift in my career for learning and understanding the significance of Data Analytics. It helped me in personal as well as professional development.

I have done this project under the guidance of MedTourEasy. I take this opportunity to express our gratitude. First and foremost, I would like to express our sincere gratitude to my mentor Ankit Hasija for his continued support and guidance which led to the completion of the project work. All the interactive conversations I have had with him during this period have been inspiring and rewarding for me. It was a great pleasure to work with him as he was exceptionally cooperative, helpful, modest and caring. This project would not have been completed without his guidance. And I am deeply grateful for MedTourEasy's support and the

opportunity they have
provided me. Their assistance allowed me to focus on my internship and
my professional development, and I am thankful for their commitment to
helping me succeed, their support and love during this journey will live in
my memory.

TABLE OF CONTENTS

Introduction

1.1 About the Company

MedTourEasy, a global healthcare company, provides you the informational
resources needed to evaluate your global options. **MedTourEasy** provides
analytical solutions to our partner healthcare providers globally.
MedTourEasy's mission is to provide access to quality healthcare

for everyone,
regardless of location, time frame or budget.

1.2 Project Description

In this project, we will explore this phenomenon using age distribution data to
see if we can reproduce a difference in average age at death purely from the
changing rates of left-handedness over time, refuting the claim of early death for
left-handers. This notebook uses pandas and Bayesian statistics to analyze the
probability of being a certain age at death given that you are reported as
lefthanded or right-handed.

1.3 Objectives and Deliverables

This project is all about focusing and carrying out the in-depth analysis by
gathering data of right-handers and left-handers from various sources like death
distribution data for the United States from the year 1999 and rates of lefthandedness digitized from a figure in this 1992 paper by Gilbert and Wysocki.
Then, using Python and it's packages like Pandas, NumPy and Matplotlib.Pyplot
to analyze and visualize this project which will provide actionable insights that
can help company make informed decisions.

The project consists of deliverables as follows:

- Rates of left-handedness over time.
- When do people normally die?
- The overall probability of left-handedness
- Age of left and right-handers at death
- Differences of average age of left-handed people and right-handed people
at death.

2. Methodology

2.1 Approach to the Project

First, I have gathered data of right-handers and left-handers from various sources
like death distribution data for the United States from the year 1999 and rates of
left-handedness digitized from a figure in this 1992 paper by Gilbert and
Wysocki.

Then I have loaded the datasets in Python. Then I have done the EDA part using
Python with the help of Pandas a data analysis library. And this analysis was
done with the help of various libraries, functions and formula.

The project followed the following steps to accomplish the desired objectives
and deliverables:

- Gathering Requirements & Defining Problem

- Data collection and importing
- Cleaning the datasets
- Analyze the datasets
- Exploratory Data Analysis (EDA)

2.1 Language and Platform Used
Language: Python

There are many programming languages available, but Python is popularly used
by statisticians, engineers, scientists and analyst to perform data analytics.
Here are some of the reasons why Data Analytics using Python has become
popular:
1. Python is easy to learn and understand and has a simple syntax.
2. The programming language is scalable and flexible.
3. It has a vast collection of libraries for numerical computation and data
manipulation.
4. Python provides libraries for graphics and data visualization

to build plots.
5. It has broad community support to help solve many kinds of queries.

IDE: Google Colab

Colaboratory, or "Colab" for short, is a product from Google Research. Colab allows anybody to write and execute arbitrary python code through the browser, and is especially well suited to machine learning, data analysis and education.
More technically, Colab is a hosted Jupyter notebook service that requires no setup to use, while providing access free of charge to computing resources including GPUs. Colab notebooks allow you to combine executable code and rich text in a single document, along with images, HTML, LaTeX and more. To be precise, Colab is a free Jupyter notebook environment that runs entirely in the cloud. Most importantly, it does not require a setup.

Libraries: Pandas, NumPy, Matplotlib

One of the main reasons why Data Analytics using Python has become the most
preferred and popular mode of data analysis is that it provides a range of libraries.

NumPy: NumPy supports n-dimensional arrays and provides numerical computing
tools. It is useful for Linear algebra and Fourier transform.

Pandas: Pandas provides functions to handle missing data, perform mathematical
operations, and manipulate the data.

Matplotlib: Matplotlib library is commonly used for plotting data points and
creating interactive visualizations of the data.

```
# import libraries
import pandas as pd
import matplotlib.pyplot as plt
```

```
# import library
import numpy as np
```

3. IMPLEMENTATION

3.1 Gathering Requirements and Defining Problem Statement

So, this is the first step where all the instruction and requirements are received from MedTourEasy to understand what needs to be done in this project and all the questions are being asked by MedTourEasy need to be answered to reach the deliverables during this project, after this the final step is the problem statement which is defined which has to be followed while development of the project.

3.2 Data Collection and Importing

The data of Right-Handers and Left-Handers has been collected through various sources, mentioned as follows:

- Death distribution data for the United States from the year 1999.

▪ Rates of left-handedness digitized from a figure in this 1992 paper by
Gilbert and Wysocki.
▪ https://www.cdc.gov/nchs/nvss/mortality_tables.htm

Data importing is something that let us upload the required data into the
programing language from external sources (online websites and data
repositories). Then the data can be manipulated, aggregated, filtered as per
the analysis requirements and needs of the project.

Read_csv: CSV files are the Comma Separated Files. To access data from
the CSV file, we require a function read_csv() from Pandas that retrieves data
in the form of the data frame.

```
# load the data
data_url_1 =
"https://gist.githubusercontent.com/mbonsma/8da0990b71ba9a09f7de395574
e54df1/raw/aec88b30af87fad8d45da7e774223f91dad09e88/lh_data.csv"
lefthanded_data = pd.read_csv(data_url_1)
```

```
# Death distribution data for the United States in 1999
data_url_2 =
"https://gist.githubusercontent.com/mbonsma/2f4076aab6820ca1807f4e29f7
5f18ec/raw/62f3ec07514c7e31f5979beeca86f19991540796/cdc_vs00199_table3
10.tsv"

# load death distribution data
death_distribution_data = pd.read_csv(data_url_2, sep= '\t', skiprows=
[1])
```

3.3 Designing Databases

Once the data is collected and imported into the Python environment, it is necessary to design the structure of the database tables as to clearly recognize the columns, rows and datatype in the data. Once the data is imported into Python environment, it is converted into pandas' data frame which makes it easy to maintain the data in form of tables. The following are the various tables which have been created as pandas' data frame:

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 77 entries, 0 to 76
Data columns (total 3 columns):
 #   Column  Non-Null Count  Dtype
---  ------  --------------  -----
 0   Age     77 non-null     int64
 1   Male    77 non-null     float64
 2   Female  77 non-null     float64
dtypes: float64(2), int64(1)
memory usage: 1.9 KB
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 120 entries, 0 to 120
Data columns (total 4 columns):
 #   Column      Non-Null Count  Dtype
---  ------      --------------  -----
 0   Age         120 non-null    int64
 1   Both Sexes  120 non-null    float64
 2   Male        115 non-null    float64
 3   Female      120 non-null    float64
dtypes: float64(3), int64(1)
memory usage: 4.7 KB
```

Analysis and Observations

4.1. Where are the old left-handed people?

In this notebook, we will explore this phenomenon using age distribution data to see if we can reproduce a difference in average age at death purely from the changing rates of left-handedness over time, refuting the claim of early death for left-handers. This notebook uses `pandas` and Bayesian statistics to analyze the probability of being a certain age at death given that you are reported as left-handed or right-handed.

A National Geographic survey in 1986 resulted in over a million responses that included age, sex, and hand preference for throwing and writing. Researchers Avery Gilbert and Charles Wysocki analyzed this

data and noticed that rates of left-handedness were around 13% for people younger than 40 but decreased with age to about 5% by the age of 80. They concluded based on analysis of a subgroup of people who throw left-handed but write right-handed that this age-dependence was primarily due to changing social acceptability of left-handedness. This means that the rates aren't a factor of *age* specifically but rather of the *year you were born*, and if the same study was done today, we should expect a shifted version of the same distribution as a function of age. Ultimately, we'll see what effect this changing rate has on the apparent mean age of death of left-handed people, but let's start by plotting the rates of left-handedness as a function of age.

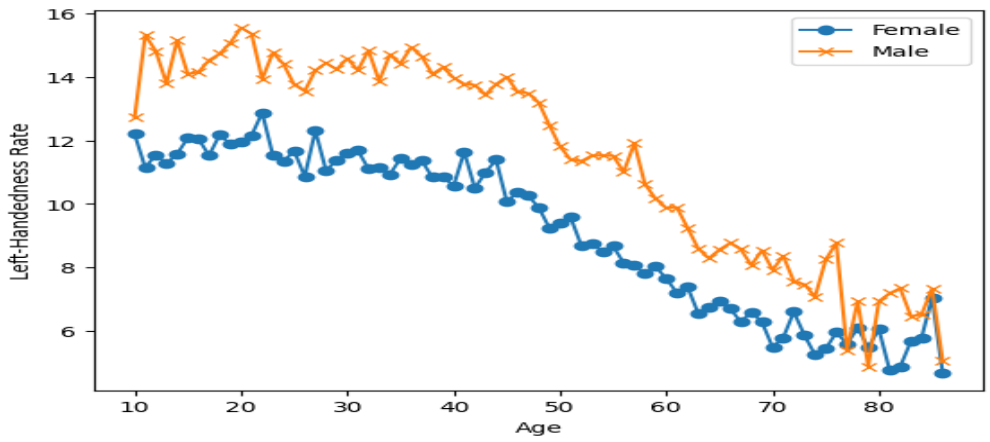
This notebook uses two datasets: death distribution data for the United States from the year 1999 (source website here) and rates of left-handedness digitized from a figure in this 1992 paper by Gilbert and Wysocki.

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import datetime

data_url_1 = "https://gist.githubusercontent.com/mbonsma/8da0990b71ba9a09f7de395574e54df1/raw/aec88b30af87fad8d45da7e774223f91dad09e88/lh_d
lefthanded_data = pd.read_csv(data_url_1)

%matplotlib inline

fig, ax = plt.subplots() # create figure and axis objects
ax.plot(lefthanded_data['Age'], lefthanded_data['Female'], label='Female', marker = 'o') # plot "Female" vs. "Age"
ax.plot(lefthanded_data['Age'], lefthanded_data['Male'], label='Male', marker = 'x') # plot "Male" vs. "Age"
ax.legend() # add a legend
ax.set_xlabel('Age')
ax.set_ylabel('Left-Handedness Rate')
plt.show()
```



4.2. Rates of left-handedness over time

Let's convert this data into a plot of the rates of left-handedness as a function of the year of birth, and average over male and female to get a single rate for both sexes.

Since the study was done in 1986, the data after this conversion will be the percentage of people alive in 1986 who are left-handed as a function of the year they were born.

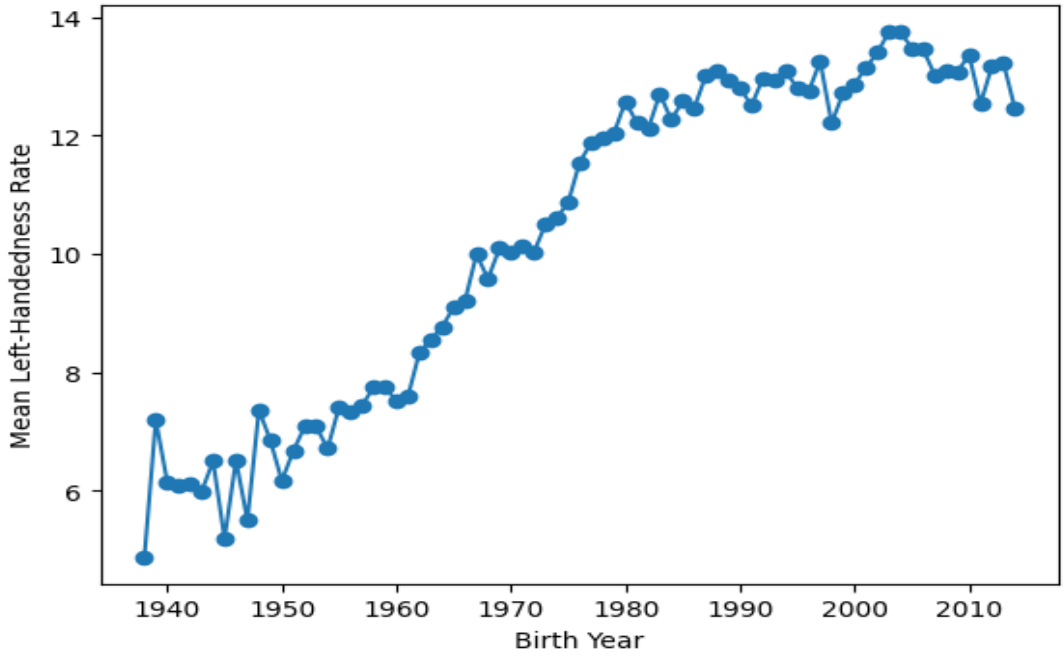
```
# create a new column for birth year of each age
# ... YOUR CODE FOR TASK 2 ...

lefthanded_data['Birth_year'] = 2024 - lefthanded_data['Age']

# create a new column for the average of male and female
# ... YOUR CODE FOR TASK 2 ...

lefthanded_data['Mean_lh'] = (lefthanded_data['Male'] + lefthanded_data['Female']) / 2

# create a plot of the 'Mean_lh' column vs. 'Birth_year'
fig, ax = plt.subplots()
ax.plot(lefthanded_data['Birth_year'], lefthanded_data['Mean_lh'], marker='o') # plot 'Mean_lh' vs. 'Birth_year'
ax.set_xlabel('Birth Year') # set the x label for the plot
ax.set_ylabel('Mean Left-Handedness Rate') # set the y label for the plot
plt.show()
```



4.3. Applying Bayes' rule

The probability of dying at a certain age given that you're left-handed is **not** equal to the probability of being left-handed given that you died at a certain age. This inequality is why we need **Bayes' theorem**, a statement about conditional probability which allows us to update our beliefs after seeing evidence.

We want to calculate the probability of dying at age A given that you're left-handed. Let's write this in shorthand as P(A | LH). We also want the same quantity for right-handers: P(A | RH).

Here's Bayes' theorem for the two events we care about: left-handedness (LH) and dying at age A.

$$P(A | LH) = \frac{P(LH|A) P(A)}{P(LH)}$$

P(LH | A) is the probability that you are left-handed *given that* you died at age A. P(A) is the overall probability of dying at age A, and P(LH) is the overall probability of being left-handed. We will now calculate each of these three quantities, beginning with P(LH | A).

To calculate P(LH | A) for ages that might fall outside the original data, we will need to extrapolate the data to earlier and later years. Since the rates flatten out in the early 1900s and late 1900s, we'll use a few points at each end and take the mean to extrapolate the rates on each end. The number of points used for this is arbitrary, but we'll pick 10 since the data looks flat-ish until about 1910.

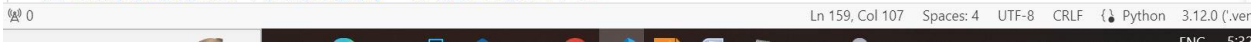
```
# import library
# ... YOUR CODE FOR TASK 3 ...

# create a function for P(LH | A)
def P_lh_given_A(ages_of_death, study_year = 1990):
    """ P(Left-handed | ages of death), calculated based on the reported rates of left-handedness.
    Inputs: numpy array of ages of death, study_year
    Returns: probability of left-handedness given that subjects died in `study_year` at ages `ages_of_death` """

    # Use the mean of the 10 last and 10 first points for left-handedness rates before and after the start
    early_1900s_rate = lefthanded_data.loc[lefthanded_data['Birth_year'].between(study_year - 1890 - 10, study_year - 1890)]['Mean_lh'].mea
    late_1900s_rate = lefthanded_data.loc[lefthanded_data['Birth_year'].between(study_year - 1986, study_year - 1986 + 10)]['Mean_lh'].mean
    middle_rates = lefthanded_data.loc[lefthanded_data['Birth_year'].isin(study_year - ages_of_death)]['Mean_lh']
    youngest_age = study_year - 1986 + 10 # the youngest age is 10
    oldest_age = study_year - 1986 + 86 # the oldest age is 86

    P_return = np.zeros(ages_of_death.shape) # create an empty array to store the results
    # extract rate of left-handedness for people of ages 'ages_of_death'
    #P_return[ages_of_death > oldest_age] = late_1900s_rate / 100
    #P_return[ages_of_death < youngest_age] = early_1900s_rate / 100

    # Use boolean indexing for the middle age range
    middle_mask = np.logical_and((ages_of_death <= oldest_age), (ages_of_death >= youngest_age))
    # Fill the middle values with the mean of middle_rates
    P_return[middle_mask] = middle_rates.mean() / 100
    #P_return[np.logical_and((ages_of_death <= oldest_age), (ages_of_death >= youngest_age))] = middle_rates / 100
    # Assign values for other age ranges
    P_return[ages_of_death > oldest_age] = late_1900s_rate / 100
    P_return[ages_of_death < youngest_age] = early_1900s_rate / 100
```



4.4. When do people normally die?

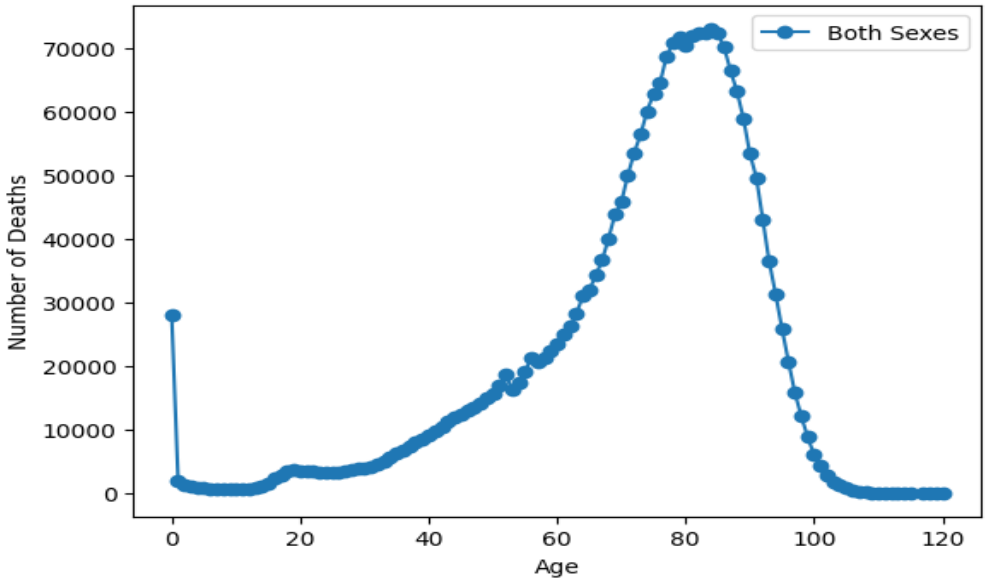
To estimate the probability of living to an age A, we can use data that gives the number of people who died in a given year and how old they were to create a distribution of ages of death. If we normalize the numbers to the total number of people who died, we can think of this data as a probability distribution that gives the probability of dying at age A. The data we'll use for this is from the entire US for the year 1999 - the closest I could find for the time range we're interested in.

In this block, we'll load in the death distribution data and plot it. The first column is the age, and the other columns are the number of people who died at that age.

```
# Death distribution data for the United States in 1999
data_url_2 = "https://gist.githubusercontent.com/mbonsma/2f4076aab6820ca1807f4e29f75f18ec/raw/62f3ec07514c7e31f5979beeca86f19991540796/cdc_

# load death distribution data
# ... YOUR CODE FOR TASK 4 ...
death_distribution_data = pd.read_csv(data_url_2, sep='\t', skiprows=[1])  # Skip the second row as it contains non-tabular data

# drop NaN values from the `Both Sexes` column
# ... YOUR CODE FOR TASK 4 ...
death_distribution_data = death_distribution_data.dropna(subset=['Both Sexes'])
# plot number of people who died as a function of age
fig, ax = plt.subplots()
ax.plot(death_distribution_data['Age'], death_distribution_data['Both Sexes'], label = 'Both Sexes', marker='o') # plot 'Both Sexes' vs. 'A
ax.set_xlabel('Age')
ax.set_ylabel('Number of Deaths')
ax.legend()
plt.show()
```



4.5. The overall probability of left-handedness

In the previous code block we loaded data to give us P(A), and now we need P(LH). P(LH) is the probability that a person who died in our particular study year is left-handed, assuming we know nothing else about them. This is the average left-handedness in the population of deceased people, and we can calculate it by summing up all of the left-handedness probabilities for each age, weighted with the number of deceased people at each age, then divided by the total number of deceased people to

get a probability. In equation form, this is what we're calculating, where N(A) is the number of people who died at age A (given by the dataframe `death_distribution_data`):

$$P(LH) = \frac{\sum_A P(LH|A)N(A)}{\sum_A N(A)}$$

```
def P_lh(death_distribution_data, study_year = 1990): # sum over P_lh for each age group
    """ Overall probability of being left-handed if you died in the study year
    Input: dataframe of death distribution data, study year
    Output: P(LH), a single floating point number """

    # Calculate P_lh_given_A for each age group
    p_list = P_lh_given_A(death_distribution_data['Age'].values, study_year)

    p_list *= death_distribution_data['Both Sexes'].values # multiply number of dead people by P_lh_given_A
    p = np.sum(p_list) # calculate the sum of p_list
    p /= np.sum(death_distribution_data['Both Sexes'].values)

    return p # normalize to total number of people (sum of death_distribution_data['Both Sexes'])

print(P_lh(death_distribution_data))
```

4.6. Putting it all together: dying while left-handed (i)

Now we have the means of calculating all three quantities we need: P(A), P(LH), and P(LH | A). We can combine all three using Bayes' rule to get P(A | LH), the probability of being age A at death (in the study year) given that you're left-handed. To make this answer meaningful, though, we also want to compare it to P(A | RH), the probability of being age A at death given that you're right-handed.

We're calculating the following quantity twice, once for left-handers and once for right-handers.

$$P(A | LH) = \frac{P(LH|A) P(A)}{P(LH)}$$

First, for left-handers.

```
def P_A_given_lh(ages_of_death, death_distribution_data, study_year = 1990):
    """ The overall probability of being a particular `age_of_death` given that you're left-handed """
    P_A = death_distribution_data.loc[death_distribution_data['Age'].isin(ages_of_death)]['Both Sexes'] / np.sum(death_distribution_data['B
    P_left = P_lh(death_distribution_data, study_year) # use P_lh function to get probability of left-handedness overall
    P_lh_A = P_lh_given_A(ages_of_death, study_year) # use P_lh_given_A to get probability of left-handedness for a certain age
    result= P_lh_A*P_A/P_left
    return result
```

4.7. Putting it all together: dying while left-handed (ii)

And now for right-handers.

```
def P_A_given_rh(ages_of_death, death_distribution_data, study_year = 1990):
    """ The overall probability of being a particular `age_of_death` given that you're right-handed """
    P_A = death_distribution_data.loc[death_distribution_data['Age'].isin(ages_of_death)]['Both Sexes'] / np.sum(death_distribution_data['B
    # Calculate the probability of being left-handed overall
    P_left = P_lh(death_distribution_data, study_year)
    P_right = 1 - P_left # either you're left-handed or right-handed, so P_right = 1 - P_left
    P_rh_A = 1 - P_lh_given_A(ages_of_death, study_year) # P_rh_A = 1 - P_lh_A
    result = P_rh_A * P_A / P_right

    return result
```

4.8. Plotting the distributions of conditional probabilities

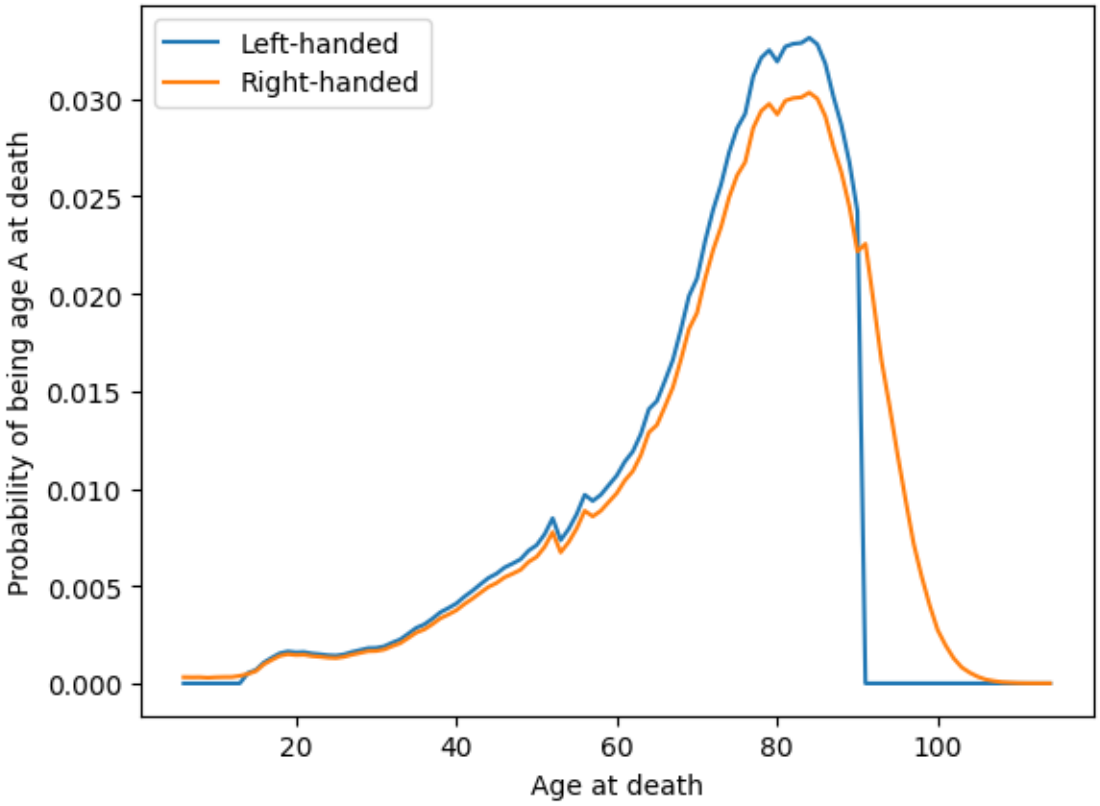
Now that we have functions to calculate the probability of being age A at death given that you're left-handed or right-handed, let's plot these probabilities for a range of ages of death from 6 to 120.

Notice that the left-handed distribution has a bump below age 70: of the pool of deceased people, left-handed people are more likely to be younger.

```
ages = np.arange(6, 115, 1) # make a list of ages of death to plot

# calculate the probability of being left- or right-handed for each
left_handed_probability = P_A_given_lh(ages, death_distribution_data)
right_handed_probability = P_A_given_rh(ages, death_distribution_data)

# create a plot of the two probabilities vs. age
fig, ax = plt.subplots() # create figure and axis objects
ax.plot(ages, left_handed_probability, label = "Left-handed")
ax.plot(ages, right_handed_probability, label = "Right-handed")
ax.legend() # add a legend
ax.set_xlabel("Age at death")
ax.set_ylabel(r"Probability of being age A at death")
plt.show()
```



4.9. Moment of truth: age of left and right-handers at death

Finally, let's compare our results with the original study that found that left-handed people were nine years younger at death on average. We can do this by calculating the mean of these probability distributions in the same way we calculated P(LH) earlier, weighting the probability distribution by age and summing over the result.

$$\text{Average age of left-handed people at death} = \sum_A A P(A | LH)$$

$$\text{Average age of right-handed people at death} = \sum_A A P(A | RH)$$

```
# calculate average ages for left-handed and right-handed groups
# use np.array so that two arrays can be multiplied
average_lh_age =  np.nansum(ages*np.array(P_A_given_lh(ages, death_distribution_data)))
average_rh_age =  np.nansum(ages*np.array(P_A_given_rh(ages, death_distribution_data)))

# print the average ages for each group
# ... YOUR CODE FOR TASK 9 ...

print("Average age for left-handed group: {:.1f} years".format(average_lh_age))
print("Average age for right-handed group: {:.1f} years".format(average_rh_age))

# print the difference between the average ages
print("The difference in average ages is {:.1f} years.".format(abs(average_lh_age - average_rh_age)))
```

```
Average age for left-handed group: 67.3 years
Average age for right-handed group: 72.8 years
The difference in average ages is 5.5 years.
```

4.10. Final comments

We got a pretty big age gap between left-handed and right-handed people purely as a result of the changing rates of left-handedness in the population, which is good news for left-handers: you probably won't die young because of your sinisterness. The reported rates of left-handedness have increased from just 3% in the early 1900s to about 11% today, which means that older people are much more likely to be reported as right-handed than left-handed, and so looking at a sample of recently deceased people will have more old right-handers.

Our number is still less than the 9-year gap measured in the study. It's possible that some of the approximations we made are the cause:

1. We used death distribution data from almost ten years after the study (1999 instead of 1991), and we used death data from the entire United States instead of California alone (which was the original study).
2. We extrapolated the left-handedness survey results to older and younger age groups, but it's possible our extrapolation wasn't close enough to the true rates for those ages.

One thing we could do next is figure out how much variability we would expect to encounter in the age difference purely because of random sampling: if you take a smaller sample of recently deceased people and assign handedness with the probabilities of the survey, what does that distribution look like? How often would we encounter an age gap of nine years using the same data and assumptions? We won't do that here, but it's possible with this data and the tools of random sampling.

To finish off, let's calculate the age gap we'd expect if we did the study in 2018 instead of in 1990. The gap turns out to be much smaller since rates of left-handedness haven't increased for people born after about 1960. Both the National Geographic study and the 1990 study happened at a unique time - the rates of left-handedness had been changing across the lifetimes of most people alive, and the difference in handedness between old and young was at its most striking.

```
# Calculate the probability of being left- or right-handed for all ages
left_handed_probability_2018 = P_A_given_lh(ages, death_distribution_data, study_year=2018)
right_handed_probability_2018 = P_A_given_rh(ages, death_distribution_data, study_year=2018)

# calculate average ages for left-handed and right-handed groups
average_lh_age_2018 = np.nansum(ages*np.array(left_handed_probability_2018))
average_rh_age_2018 = np.nansum(ages*np.array(right_handed_probability_2018))

print("The difference in average ages is {:.1f} years.".format(average_rh_age_2018 - average_lh_age_2018))
```

```
The difference in average ages is 1.5 years.
```

Conclusion and Future Scope

If we compare our results with the original study, we found that left-handed
people were nine years younger at death on average. But as per the

analysis of
this project, in the above calculation, we found that the average age of
Lefthanders is 67.25 and the average age of Righthanders is 72.79 and the
difference in average ages is 5.5 years. And the difference in average ages is 1.5
years if we did the study in 2018 instead of in 1990.

This project aimed at analyzing the analyze death age difference of right handers
with left handers. Currently, the project is done with analysis and all the
questions and tasks has been answered

References

1. https://www.cdc.gov/nchs/data/statab/vs00199_table310.pdf
2. https://www.cdc.gov/nchs/nvss/mortality_tables.htm
3. https://www.ncbi.nlm.nih.gov/pubmed/1528408