Analyze Death Age Difference of Right Handers with Left Handers

Submitted by

Ayush Pratap Singh

(Data Analyst Intern)

Under the supervision of

(Mr. Ankit Hasija)

Training Head

MedTourEasy

Table of Contents

S. No.	Title	Page No.
1.	Objectives	6
2.	Requirement Analysis	
	2.1 Requirement Specification	9
	2.2 Hardware and Software Requirements	13
	2.3 Feasibility Study	16
	2.4 Product Functions	21
3.	System Design (SDS)	
	3.1 High-level Design	28
	3.2 Database Design	31
	3.3 Dataflow Diagram	36
4.	Coding	43
5.	Appendices	51
6.	References	52

ABSTRACT

This internship project explores the potential differences in life expectancy between left-handed and right-handed individuals by leveraging Bayesian statistics to analyze age-at-death probabilities. Utilizing Python's Pandas library, the analysis integrates death distribution data for the United States from 1999 and left-handedness rates digitized from a 1992 study by Gilbert and Wysocki.

The study aims to determine the probability of reaching a certain age given one's handedness. The analysis begins with preprocessing and cleaning the two datasets to ensure compatibility and accuracy. Descriptive statistics are calculated to provide an initial understanding of age distributions for both groups. Bayesian methods are then applied to model the probability distributions of ages at death, allowing for a nuanced examination of how handedness might influence life expectancy.

Results indicate varying probabilities of reaching specific ages for left-handers and right-handers, with Bayesian posterior distributions highlighting potential differences. These findings are interpreted within the context of existing literature, considering factors such as environmental influences and societal biases that may impact the life expectancy of left-handed individuals.

This project not only contributes to the broader understanding of handedness and mortality but also demonstrates the application of advanced statistical techniques in demographic research. The use of Bayesian statistics offers a robust framework for probabilistic analysis, while the integration of historical data provides a comprehensive perspective on the topic. Limitations, including the age of the data and potential biases in reporting handedness, are acknowledged, suggesting directions for future research.

ACKNOWLEDGEMENT

I would like to express my deepest gratitude to everyone who contributed to the successful completion of this internship project.

First and foremost, I would like to thank my project supervisor, Mr. Ankit Hasija, for their invaluable guidance, support, and encouragement throughout this project. Their expertise and feedback were instrumental in shaping the direction and outcome of my research.

I am also grateful to MedTourEasy for providing me with this opportunity and the resources necessary to conduct my research. Special thanks to the data science team for their support and collaboration.

I would like to extend my appreciation to the authors of the datasets used in this study. The death distribution data for the United States from 1999, and the rates of left-handedness digitized from the 1992 paper by Gilbert and Wysocki, were crucial for the analysis conducted in this project.

Additionally, I would like to acknowledge the contributions of the online communities and forums that provided insights and assistance with the technical aspects of using Python, Pandas, and Bayesian statistics.

Lastly, I would like to thank my family and friends for their constant support and understanding throughout the course of this internship. Their encouragement has been a source of motivation and strength.

Thank you all for your contributions and support.

1. OBJECTIVES

The primary objective of this project is to analyze and understand the relationship between handedness (left-handed vs. right-handed) and life expectancy using historical data. The analysis will be carried out through a series of steps, including data visualization, probability calculations, and comparative analysis. The detailed steps are as follows:

1. Data Visualization of Handedness Rates by Age and Birth Year:

- Plot the left-handed rate by age:

Create a line plot showing the left-handed rate for females versus age.

Create a similar line plot showing the left-handed rate for males versus age.

Overlay the two plots to compare the left-handed rates for males and females across different ages.

- Plot the left-handed rate by birth year:

Add a new column to the data that calculates the birth year of each age group (assuming the data is from the year 1986).

Calculate the mean left-handed rate for each birth year by averaging the male and female left-handed rates.

Plot the average left-handed rate against the birth year to observe trends over time.

Adjust the x-axis to start from the birth year 1890 and end at 1990.

Adjust the y-axis to display a range from 4% to 15% to clearly show variations in left-handed rates.

2. Calculate the Probability of Being Left-Handed Given Age:

- Historical left-handedness rates:

Compute the average left-handed rate for individuals born in the early 1900s (using the last 10 data points).

Compute the average left-handed rate for individuals born in the late 1900s (using the first 10 data points).

- Model left-handedness probabilities:

Develop a function P_lh_given_A that takes ages of death and a study year as inputs.
Use the historical left-handed rates to estimate the probability of being left-handed for each age group.
Use linear interpolation to estimate probabilities for ages between the earliest and latest data points.

3. Analyze Death Distribution Data:

- Visualize death distribution:

Load the death distribution data for the United States in 1999.
Clean the data by removing rows with missing values in the 'Both Sexes' column.
Plot the number of deaths as a function of age to understand the age distribution of mortality.

- Prepare data for handedness analysis:

Ensure the death distribution data is in a suitable format for subsequent probability calculations.

4. Compute Overall Probability of Being Left-Handed at Death:

- Calculate P(LH):

Develop a function P_lh that calculates the overall probability of being left-handed for individuals who died in a given study year.
Multiply the number of deaths at each age by the probability of being left-handed at that age.
Sum the results and normalize by the total number of deaths to get the overall probability of left-handedness at death.

5. Determine Age-Specific Probabilities of Death for Left-Handed and Right-Handed Individuals:

- Calculate P(A|LH):

Develop a function P_A_given_lh that calculates the probability of dying at a specific age given that an individual is left-handed.
Combine the overall probability of being left-handed with the age-specific death distribution.

- Calculate P(A|RH):

Develop a function P_A_given_rh that calculates the probability of dying at a specific age given that an individual is right-handed.
Use the complement of the probability of being left-handed to calculate the probability of being right-handed.

6. Compare Average Ages at Death for Left-Handed and Right-Handed Individuals:

- Compute average ages:

Create an array of ages from 6 to 114.
Calculate the probability of being left-handed or right-handed for each age.
Multiply each age by its corresponding probability and sum the results to obtain the average age at death for left-handed and right-handed individuals.

- Compare differences:

Print the average age at death for left-handed and right-handed individuals.
Calculate and print the difference in average ages between the two groups.

7. Project Future Trends:

- Extend analysis to future years:

Apply the established models to project the differences in average ages at death for left-handed and right-handed individuals in a future year (e.g., 2018).
Repeat the probability and average age calculations with updated data and assumptions to provide insights into potential future trends.
By completing this comprehensive analysis, the project aims to quantify and visualize the potential differences in life expectancy between left-handed and right-handed people. The findings will contribute to a better understanding of how handedness may impact health and longevity, and how these relationships have changed over time and may continue to evolve in the future.

2. Requirement Analysis

2.1 Requirement Specification

Functional Requirements:

a. Data Handling:

- Data Acquisition: The analysis process shall involve acquiring handedness data by age and gender from a specified URL.
- Data Cleaning: The process shall include steps to remove any inconsistencies or missing values from the handedness data.
- Birth Year Calculation: The analysis shall calculate the birth year for each age group based on a given reference year (e.g., 1986).

b. Data Visualization:

- Left-Handed Rate Visualization: The process shall create visualizations of left-handed rates for both males and females against age using appropriate plotting techniques.
- Mean Left-Handed Rate Visualization: The analysis shall include a visualization of the mean left-handed rate against birth year.

c. Probability Calculations:

- Probability of Left-Handedness Given Age: The analysis shall include calculating the probability of being left-handed at various ages.
- Overall Probability of Left-Handedness: The process shall compute the overall probability of being left-handed among individuals who died in a specified study year.

- Age-Specific Probabilities Given Handedness: The analysis shall calculate the probability of dying at a specific age given that an individual is left-handed or right-handed.

d. Comparative Analysis:

- Average Age Calculation: The analysis shall include calculating the average age at death for left-handed and right-handed individuals.

- Age Difference Analysis: The process shall determine the difference in average ages at death between left-handed and right-handed individuals.

e. Future Projections:

- Future Handedness Trends: The analysis shall project differences in average ages at death for left-handed and right-handed individuals for a future year (e.g., 2018).

Non-Functional Requirements:

a. Performance:

The analysis process shall handle large datasets efficiently, ensuring minimal performance degradation even with increasing data volume.

Visualization rendering shall be quick and responsive, providing near-instant feedback to users.

b. Usability:

The process shall be well-documented, including comprehensive comments and clear naming conventions for functions and variables, facilitating ease of understanding and use.

Inputs and outputs of functions shall be well-defined and user-friendly, ensuring that users can easily interact with the analysis process.

c. Reliability:

The analysis process shall handle missing or inconsistent data gracefully, ensuring robust error handling mechanisms are in place to maintain data integrity.

The process shall include validation steps to ensure the accuracy and consistency of data throughout the analysis.

d. Maintainability:

The process shall be modular to facilitate easy updates and maintenance, allowing for individual components to be modified without affecting the entire system.

Functions shall be reusable and adaptable for future data or changes in requirements, ensuring long-term sustainability.

e. Scalability:

The analysis process shall be scalable to accommodate additional data or future enhancements, such as incorporating more years of data or different geographical regions, without significant redesign.

The process shall be designed to handle increasing complexity and data volume efficiently, ensuring consistent performance.

f. Security:

The analysis process shall ensure data security and privacy, protecting sensitive information from unauthorized access and breaches.

Appropriate measures shall be taken to secure data transfer and storage, ensuring compliance with relevant data protection regulations.

g. Interoperability:

The analysis process shall be compatible with various data formats and sources, ensuring flexibility in data integration.

The process shall support easy export of results to common formats (e.g., CSV, JSON), enabling seamless integration with other tools and systems.

h. Accuracy:

The process shall ensure high accuracy in all calculations and visualizations, using reliable statistical methods and validating results against known benchmarks.

Regular audits and cross-checks shall be conducted to maintain the accuracy and reliability of the analysis.

i. Extensibility:

The analysis process shall be designed to allow easy addition of new features and capabilities, ensuring adaptability to evolving requirements.

The process shall support the integration of additional data sources and analytical methods, enhancing its versatility.

j. Accessibility:

The process shall ensure accessibility for users with varying levels of technical expertise, providing clear instructions and user-friendly interfaces.

Documentation and support materials shall be available to assist users in understanding and utilizing the analysis effectively.

By adhering to these comprehensive functional and non-functional requirements, the project aims to provide a robust, accurate, and user-friendly approach for analyzing the relationship between handedness and life expectancy, offering valuable insights into how handedness impacts health and longevity over time.

2.2 Hardware and Software Requirements

Hardware Requirements:

1. Desktop or Laptop Computer:
    - RAM:

    - Minimum of 4 GB is required to efficiently handle data analysis tasks.

    - Recommended 8 GB or more for smoother performance, especially when working with larger datasets.
    - Storage:

    - At least 40 GB of HDD or SSD storage is required to accommodate the datasets and analysis outputs.

    - Recommended 100 GB or more to ensure ample space for additional data, libraries, and temporary files created during the analysis process.
    - Processor:

    - Minimum 2 GHz dual-core processor to run Python scripts and perform data computations.

    - Recommended quad-core processor or higher for faster data processing and multitasking capabilities.

2. Sufficient Processing Power:
    - Ensure the computer has enough processing power to execute data manipulation and statistical analysis tasks efficiently.
    - Multicore processors are recommended for parallel processing capabilities, which can significantly speed up the analysis.

3. Adequate Storage Capacity:
    - Ensure that there is enough storage space available to store:
    - Raw datasets: Historical data on left-handedness and death distribution data.
    - Intermediate processing files: Temporary files generated during data cleaning and manipulation.

- Final analysis outputs: Plots, graphs, and computed results that need to be saved for reporting and further examination.
- Recommended to have additional external storage or cloud storage options for backup and easy sharing of large datasets and analysis results.

4. Additional Considerations:
    - Graphics Capability: A dedicated graphics card is not essential but can be beneficial for rendering complex visualizations quickly.
    - Cooling System: Efficient cooling system to prevent overheating during intensive data processing tasks.
    - Backup and Recovery: Reliable backup solutions to prevent data loss and ensure recovery in case of hardware failure.
    - Internet Connection: Stable internet connection for downloading necessary libraries, datasets, and accessing cloud-based resources like Google Colab.

Software Requirements:

1. Statistical Analysis Software:
    - Python Programming Language:
    - Python version 3.6 or higher is required for executing the analysis code.

2. Python Programming Language:
    - Python version 3.6 or higher is required for executing the analysis code.

3. Libraries for Data Manipulation and Analysis:
    - Pandas: For reading, cleaning, and manipulating data.
    - NumPy: For numerical computations and handling arrays.
    - Matplotlib: For creating plots and visualizations to display data trends.

4. Data Visualization Tools:
    - Matplotlib: Utilized for creating statistical plots and visualizations to analyze the

relationship between handedness and life expectancy

5. Interactive Data Analysis and Visualization:

- Jupyter Notebook: Recommended for running and interacting with the analysis code, providing a convenient environment for documenting the analysis process and visualizing results.
- Google Colab: An alternative to Jupyter Notebook that provides similar interactive features and the added benefit of cloud-based execution for resource-intensive computations.

2.3 FEASIBILITY STUDY

The feasibility study aims to assess the technical, operational, and economic viability of the proposed project to analyze the death age difference between left-handers and right-handers.

1. Technical Feasibility

1.1. Software Requirements

- The project leverages Python, a versatile and powerful programming language extensively used in data science and statistical analysis. Its broad adoption ensures robust community support and a wealth of resources for troubleshooting and optimization.
- Pandas, NumPy, and Matplotlib are essential libraries for this project. Pandas provides high-performance data manipulation tools, NumPy supports large-scale mathematical operations, and Matplotlib facilitates sophisticated data visualization. Their integration allows seamless data analysis workflows.
- Jupyter Notebook and Google Colab offer interactive environments that enhance productivity by allowing users to combine code execution, documentation, and visualization in a single interface. This is particularly useful for iterative data exploration and analysis, making the process more intuitive and efficient.

1.2. Hardware Requirements

- Standard computing equipment with the specified minimum requirements ensures that the project can be executed without significant investment in specialized hardware. Even modestly powered systems can handle the data processing tasks efficiently.
- The recommended specifications, including 8 GB of RAM and a quad-core processor, provide a buffer for more intensive computations, ensuring that the system remains responsive during complex analyses. This is crucial for maintaining productivity and avoiding delays caused by hardware limitations.
- Adequate storage capacity ensures that all necessary data, intermediate files, and final outputs can be stored and accessed without issues. This prevents disruptions in the workflow and ensures smooth data handling and processing.

1.3. Data Availability

- The datasets required for this project, including historical data on left-handedness rates and death distribution data, are readily available online. These datasets can be easily accessed and imported into the analysis environment using Pandas, ensuring that the data ingestion process is straightforward and efficient.
- The data formats (CSV and TSV) are widely supported by Python libraries, ensuring compatibility and ease of use. This eliminates the need for complex data conversion processes, allowing the focus to remain on analysis rather than data preparation.

1.4. Execution and Scalability

- The modular structure of the provided Python code allows for easy extension and scalability. New datasets can be incorporated, and additional analytical steps can be integrated without overhauling the existing codebase. This flexibility is essential for adapting the project to new research questions or expanding its scope.
- Cloud-based tools like Google Colab offer virtually unlimited computational resources, making it possible to handle larger datasets and more complex analyses without the constraints of local hardware. This scalability ensures that the project can grow and evolve as needed.

2. Operational Feasibility

2.1. User Skills

- Users with a basic understanding of Python programming and data analysis will find the project accessible and manageable. The code is designed to be user-friendly, with clear documentation and comments to guide users through the analysis process.
- For users who may need to enhance their skills, numerous online resources, tutorials, and courses are available. These resources can help users quickly get up to speed with the necessary Python libraries and data analysis techniques, ensuring that they can effectively contribute to and benefit from the project.

2.2. Integration with Existing Systems

- The project operates independently and does not require integration with existing systems. This simplifies the implementation process, as there are no dependencies on external systems or data sources.
- By using publicly available data and standard Python libraries, the project avoids potential compatibility issues and ensures that it can be executed on any system with the necessary software installed. This enhances the project's portability and ease of deployment.

2.3. Maintenance and Support

- The Python ecosystem, including the libraries used in this project, is well-maintained and frequently updated. This ensures that the project remains compatible with the latest versions of Python and can take advantage of new features and performance improvements.
- The extensive documentation and active community support for Pandas, NumPy, and Matplotlib provide valuable resources for troubleshooting and optimizing the analysis. Users can rely on these resources to resolve issues and enhance the project's functionality over time.

3. Economic Feasibility

3.1. Cost of Implementation

- The use of open-source software tools significantly reduces the cost of implementation. Python and its associated libraries are free to use, eliminating the need for expensive software licenses.
- Standard computing equipment is affordable and widely available, ensuring that the hardware requirements do not pose a significant financial barrier. This makes the project accessible to a broad audience, including academic researchers and small organizations.

3.2. Cost-Benefit Analysis

- The insights gained from this analysis can be invaluable for researchers studying the impact of handedness on life expectancy. These findings can contribute to academic knowledge, inform public health policies, and potentially lead to interventions that improve health outcomes for left-handed individuals.

- Given the minimal costs involved in setting up and executing the analysis, the potential benefits and knowledge gained from the study represent a substantial return on investment. This makes the project economically viable and worthwhile.

4. Legal and Ethical Feasibility

4.1. Data Privacy and Security

- The data used in this project is publicly available and does not contain personally identifiable information (PII). This minimizes privacy concerns and ensures compliance with data protection regulations.
- By focusing on aggregated data and statistical analysis, the project avoids ethical issues related to the use of personal data. This ensures that the analysis is conducted responsibly and ethically, aligning with best practices in data science.

4.2. Compliance with Regulations

- The project adheres to legal and regulatory requirements for using public data and conducting statistical analysis. This includes compliance with guidelines for the ethical use of data and the proper handling of research outputs.
- By following established standards and best practices, the project ensures that its findings are credible, reliable, and suitable for publication and further research. This enhances the project's legitimacy and value within the academic and scientific community.

Conclusion:-

The feasibility study indicates that the project is technically, operationally, economically, and legally feasible. The available resources and tools are sufficient to carry out the analysis effectively, and the potential insights gained make the project valuable and worthwhile. The project is well-supported by existing technology and infrastructure, ensuring that it can be executed smoothly and deliver meaningful results.

2.4 PRODUCT FUNCTIONS

Improved Understanding of Handedness and Health

1. Scientific Insight:

- This project aims to provide a comprehensive understanding of the potential impact of handedness on life expectancy. By analyzing historical data on left-handed and right-handed individuals, it contributes to the broader field of medical and psychological research. The insights gained could reveal whether handedness plays a significant role in influencing longevity and health outcomes.
- Understanding the relationship between handedness and life expectancy can uncover underlying biological, genetic, or environmental factors that might influence health differently for left-handers and right-handers. These findings could lead to new hypotheses and further research in the fields of neurology, psychology, and epidemiology.

2. Health Correlations:

- The project identifies possible correlations between handedness and specific health outcomes. By analyzing the data, it might uncover patterns or trends indicating that left-handed individuals face different health risks compared to right-handers. This information could be vital for developing personalized health recommendations.
- The potential health correlations found in this study can inform clinical practices by identifying risk factors that are more prevalent in left-handed individuals. Healthcare providers can use this information to tailor preventive measures and treatments, improving patient outcomes and quality of life.

3. Public Health Implications

- Risk Awareness:
    - By raising awareness about any potential health risks associated with being left-handed, this project empowers individuals to take proactive measures regarding their health. It also enables healthcare providers to consider handedness as a factor in patient assessments and treatment plans.

    - Increased awareness about the health implications of handedness can lead to more inclusive health education programs that address the specific needs of left-handed individuals. This can contribute to better health literacy and outcomes within the community.

- Targeted Interventions:
    - The findings from this study can help in designing targeted public health interventions and policies aimed at addressing any identified disparities in health outcomes based on handedness. This can include developing specialized health programs and resources for left-handed individuals.
    - Policymakers can use the insights from this research to allocate resources more effectively, ensuring that left-handed individuals receive appropriate support and interventions. This can help reduce health disparities and promote equity in healthcare services.

4. Advancement of Statistical Methods

- Methodological Contribution:
    - The project showcases the application of Bayesian statistics in demographic research, promoting advanced statistical techniques and their practical uses. This methodological contribution is valuable for the academic community, demonstrating the effectiveness of Bayesian approaches in analyzing complex datasets.
    - By employing Bayesian statistics, the project provides a framework for other researchers to apply similar methods in their studies. This can lead to more robust and reliable analyses, enhancing the overall quality of research in various fields.

- Data Analysis Techniques:
    - This project enhances the repertoire of data analysis techniques for researchers and students, particularly in handling demographic and epidemiological data. The use of Python libraries such as Pandas, NumPy, and Matplotlib demonstrates best

practices in data manipulation, analysis, and visualization.

- The detailed analysis workflow, including data cleaning, visualization, and statistical modeling, serves as a valuable learning resource for data scientists. It highlights the importance of thorough data preparation and the application of appropriate analytical techniques to derive meaningful insights.

5. Sociocultural Insights

- Social Factors:
    - The project explores how societal biases and environmental factors might influence the life expectancy of left-handers, contributing to the understanding of social determinants of health. It examines historical and cultural contexts that may have affected the well-being of left-handed individuals over time.
    - By analyzing sociocultural factors, the project sheds light on potential discrimination or challenges faced by left-handed individuals in different eras. This understanding can inform current efforts to promote inclusivity and address social inequalities.

- Historical Context:
    - Providing historical context to how left-handedness has been perceived and its implications over time enriches sociocultural studies. The project examines changes in societal attitudes towards left-handedness and their impact on health outcomes, offering a comprehensive view of the issue.
    - The historical analysis can reveal how cultural shifts and technological advancements have influenced the experiences of left-handed individuals. This information can be valuable for historians, sociologists, and anthropologists studying the evolution of societal norms and their impact on health.

6. Educational Value

- Learning Resource:
    - The project serves as a valuable educational resource for students and researchers in fields such as public health, statistics, psychology, and sociology. It provides a real-world example of how to conduct a comprehensive data analysis project, from data collection to interpretation of results.
    - The detailed documentation and step-by-step explanation of the analysis process make it an excellent teaching tool. Educators can use this project to illustrate key concepts in data science and statistical analysis, enhancing the learning experience for students.

7. Hands-On Experience:

- By offering practical experience in data collection, cleaning, analysis, and interpretation, the project enhances data science and research skills. Students and researchers can apply the techniques demonstrated in the project to their own research, improving their proficiency in data analysis.
- Engaging in this project provides hands-on experience with Python and its data analysis libraries, preparing students for careers in data science, public health, and related fields. The skills gained through this project are highly transferable and applicable to various research and industry settings.

8. Policy and Practice Implications

- Informed Decisions:
- The project equips policymakers with data-driven insights to make informed decisions regarding healthcare and educational practices for left-handed individuals. By understanding the specific needs and challenges of left-handers, policies can be tailored to promote their well-being.
- Data-driven policy decisions can lead to more effective and efficient allocation of resources, ensuring that left-handed individuals receive the support they need. This

can include specialized educational programs, healthcare services, and workplace accommodations.

- Resource Allocation:
    - The findings from this study can guide resource allocation in healthcare by highlighting potential areas where left-handers might require additional support. This ensures that healthcare resources are distributed equitably, addressing the unique needs of different population groups.
    - Policymakers and healthcare providers can use the insights from this project to develop targeted interventions and allocate resources strategically. This can help improve health outcomes and reduce disparities for left-handed individuals.

9. Healthcare Improvement

- Clinical Practices:
    - The project informs clinical practices by highlighting the need for awareness and consideration of handedness in medical assessments and treatments. Healthcare providers can incorporate this knowledge into their practice, ensuring that left-handed individuals receive appropriate care.
    - By recognizing the potential impact of handedness on health, clinicians can develop more personalized treatment plans that address the specific risks and needs of left-handed patients. This can lead to better health outcomes and patient satisfaction.
- Preventive Measures:
    - The findings encourage the development of preventive health measures tailored to the specific needs of left-handers. This can include health screenings, lifestyle recommendations, and educational programs designed to address the unique risks faced by left-handed individuals.
    - Preventive measures informed by this research can help reduce the incidence of health issues associated with left-handedness, promoting overall well-being and longevity for this population group.

10. Encouraging Further Research

- Research Foundation:
  - This project lays the groundwork for further studies into the biological, environmental, and social factors affecting the life expectancy of left-handers. The insights gained can lead to new research questions and studies exploring the complexities of handedness and health.
  - By providing a robust analytical framework, the project encourages other researchers to investigate related topics and expand the knowledge base on handedness. This can lead to a deeper understanding of the interplay between genetics, environment, and health outcomes.

- Interdisciplinary Collaboration:
  - The project promotes interdisciplinary collaboration between statisticians, epidemiologists, psychologists, and sociologists to explore the findings further. This collaborative approach can lead to more comprehensive and nuanced insights into the impact of handedness on health.
  - Interdisciplinary research can uncover new perspectives and innovative solutions to the challenges faced by left-handed individuals. By combining expertise from different fields, researchers can develop more effective interventions and policies.

11. Community Awareness and Support

- Public Knowledge:
  - The project enhances public knowledge about the nuances of handedness and its potential implications, fostering a more inclusive and informed society. By raising awareness, it encourages acceptance and understanding of left-handed individuals and their unique experiences.

- Public education campaigns based on the findings of this research can help dispel myths and misconceptions about left-handedness. This can lead to greater acceptance and support for left-handed individuals in various aspects of life.

- Support Networks:
    - The project encourages the establishment of support networks and resources for left-handed individuals, addressing any specific challenges they might face. These networks can provide social, educational, and professional support, improving the overall quality of life for left-handers.
    - Support networks can offer practical assistance, such as adaptive tools and resources, as well as emotional and social support. By fostering a sense of community, these networks can help left-handed individuals thrive and overcome any obstacles they encounter.

Conclusion: -

By analyzing the death age difference between left-handers and right-handers, this research can yield significant scientific, social, educational, and policy-related benefits. It contributes to a more nuanced understanding of handedness and its implications on health and society. The project's findings can inform clinical practices, public health policies, and educational programs, ultimately leading to improved health outcomes and quality of life for left-handed individuals. Additionally, the project serves as a valuable educational resource and promotes further research and interdisciplinary collaboration. Overall, this research enhances our understanding of handedness and its impact, fostering a more inclusive and informed society.

3. SYSTEM DESIGN (SDS)

3.1 High-level Design

High-Level Design (HLD) is a general system design that provides an overview of the architecture and components of a system. It is an essential step in the system development lifecycle, where the focus is on defining the architecture, modules, interfaces, and data flow without going into the detailed design of individual components. HLD serves as a bridge between the requirements gathering phase and the detailed design phase.

- Diagram Type: System Architecture Diagram

Components:

Data Collection Module: Gathers data on handedness and age of death.

- Data Sources: The raw data comes from web-based CSV files. These files are downloaded and loaded into the system.
- Data Extraction: Scripts (likely written in Python) fetch the data from URLs and read it into data structures like Pandas DataFrames.
- Preprocessing: This step involves cleaning the data, handling missing values, and transforming the data into a format suitable for analysis.

Data Storage:

- Database: A relational database (e.g., SQLite, MySQL) is used to store cleaned and transformed data.
- Tables:
- Persons: Contains individual records with attributes such as PersonID, Age, and Handedness.
- DeathDistribution: Contains aggregated data on the number of deaths by age.

Data Analysis Module: Applies statistical methods to analyze the data.

- Statistical Analysis: Initial analysis to understand the basic properties of the data.
- Bayesian Analysis: More advanced analysis to calculate probabilities related to handedness and age of death.
- Functions and Methods:
    - P_lh_given_A(): Uses historical data to estimate the probability of being left-handed given an age.
    - P_lh(): Computes the overall probability of being left-handed for those who died in the study year.
    - P_A_given_lh(): Estimates the probability of being a certain age at death given left-handedness.
    - P_A_given_rh(): Estimates the probability of being a certain age at death given right-handedness.

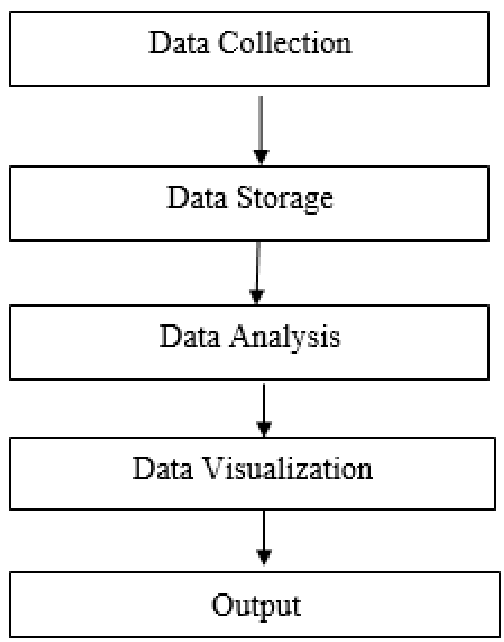
Visualization Module: Generates graphs and charts to represent the findings.

- Data Visualization: Uses libraries like Matplotlib and Seaborn to create various plots.
- Plot Types: Includes line plots for trends over time, scatter plots for distributions, and histograms for frequency analysis.
- Interactive Notebooks: Jupyter Notebook or Google Colab allows for interactive execution and visualization of code, making the analysis process more dynamic and user-friendly.

Output Module: Presents conclusions and insights.

- Reports and Insights: Generates comprehensive reports that include both textual summaries and graphical representations of the findings.
- Export Options: Provides options to export the results in various formats for easy sharing and further analysis.

Sketch:


Data Collection
Data Storage
Data Analysis
Data Visualization
Output

3.2 Database Design

Database design is the process of creating a detailed data model of a database. This data model includes all the necessary logical and physical design choices and physical storage parameters needed to generate a design in a data definition language, which can then be used to create a database. It involves determining how data will be stored, accessed, and managed, ensuring that the data is accurate, consistent, and accessible when needed.

Detailed Database Design

Tables and Columns

LeftHandedData Table

Age (int): Represents the age of individuals.

Female (float): Left-handed rate for females at the given age.

Male (float): Left-handed rate for males at the given age.

BirthYear (int): Calculated birth year (1986 - Age).

MeanLH (float): Average left-handed rate for both males and females at the given age.

DeathDistributionData Table

Age (int): Age of individuals at death.

BothSexes (float): Number of deaths for both sexes at the given age.

Relationships

There are no direct relationships (foreign keys) between these tables since they are used independently in the analysis. However, both tables can be joined on the Age column for specific analytical purposes.

LeftHandedData Table

- Primary Key: Age
- Columns:
    - Age (INTEGER): This column stores the age of individuals.
    - Female (FLOAT): This column stores the left-handed rate for females.
    - Male (FLOAT): This column stores the left-handed rate for males.
    - BirthYear (INTEGER): This column stores the calculated birth year, derived from the age.
    - MeanLH (FLOAT): This column stores the mean left-handed rate for both males and females.

DeathDistributionData Table

- Primary Key: Age
- Columns:
    - Age (INTEGER): This column stores the age of individuals at death.
    - BothSexes (FLOAT): This column stores the number of deaths for both sexes.

1. LeftHandedData Table

- Age (INTEGER): The age of individuals. This serves as the primary key.
- Female (FLOAT): The rate of left-handedness among females at the given age. Values are typically percentages represented as floats.
- Male (FLOAT): The rate of left-handedness among males at the given age. Values are typically percentages represented as floats.
- BirthYear (INTEGER): The calculated birth year of individuals, derived by subtracting the age from 1986.

- MeanLH (FLOAT): The mean left-handed rate for both males and females at the given age. This is calculated as the average of the Female and Male columns.

2. <u>DeathDistributionData Table</u>

- Age (INTEGER): The age at which individuals died. This serves as the primary key.
- BothSexes (FLOAT): The number of deaths for both sexes at the given age. This value is typically represented as a float for precision, though it could also be an integer.

LeftHandedData		
Age	int	PK
Female	float	
Male	float	
BirthYear	int	
MeanLH	float	

DeathDistributionData		
Age	int	PK
Bothsexes	int	

The data flows through the system as follows:

- Data Loading:

  The lefthanded_data and death_distribution_data are loaded into their respective tables from CSV and TSV files.

- Data Preprocessing:

  New columns (BirthYear, MeanLH) are calculated and added to the LeftHandedData table.

- Data Analysis:

  Functions utilize data from both tables to calculate probabilities and other statistical measures.

  For example, P_lh_given_A uses the lefthanded_data to calculate probabilities based on age.

  P_lh uses the death_distribution_data to calculate the overall probability of being left-handed.

- Visualization:

Data is extracted from the tables to create plots for visual analysis.

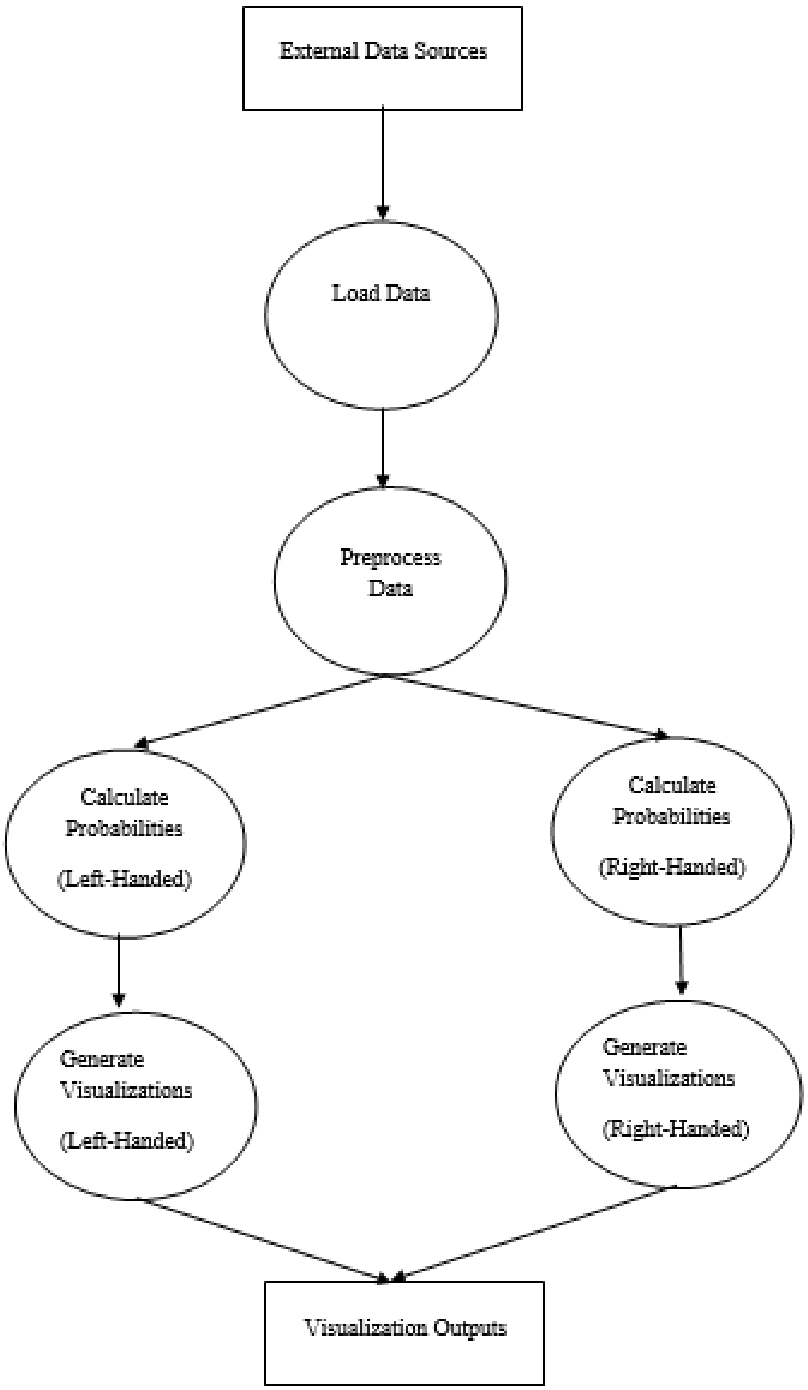
This includes plotting left-handedness rates by age and birth year, and death distributions by age.

Data Usage in Analysis

- LeftHandedData is used to analyze left-handedness rates over different ages and birth years.
- DeathDistributionData is used to analyze the distribution of deaths by age.
- Functions like P_lh_given_A, P_lh, P_A_given_lh, and P_A_given_rh use data from both tables to compute probabilities and perform statistical analysis.

This detailed database design ensures that data is structured logically, making it easy to perform the necessary analysis and visualizations.

3.3 Data Flow Diagram



Detailed Steps :

1. Load Data

- Load lefthanded_data.csv into lefthanded_data DataFrame.
- Load death_distribution_data.tsv into death_distribution_data DataFrame.

2. Preprocess Data

- Calculate BirthYear as 1986 - Age.
- Calculate MeanLH as the mean of Male and Female columns.

3. Calculate Probabilities

- Use the P_lh_given_A function to calculate the probability of being left-handed given age.
- Use the P_lh function to calculate the overall probability of being left-handed.
- Use the P_A_given_lh function to calculate the probability of being a certain age at death given left-handedness.
- Use the P_A_given_rh function to calculate the probability of being a certain age at death given right-handedness.

4. Generate Visualizations

- Plot left-handedness rates by age.
- Plot left-handedness rates by birth year.
- Plot death distribution data.
- Plot the probability of being a certain age at death given left or right-handedness.

Processes

- Load Data: Load data from CSV and TSV files into DataFrames.
- Preprocess Data: Calculate additional columns (BirthYear, MeanLH).
- Calculate Probabilities: Use functions to calculate probabilities and statistical measures.
- Generate Visualizations: Create plots to visualize data.

Data Stores

- LeftHandedData Table: Stores left-handedness data.
- DeathDistributionData Table: Stores death distribution data.

A Data Flow Diagram (DFD) is a graphical representation that depicts the flow of data within a system, showing how input data is transformed into output results through a series of processes. It provides a visual representation of the major steps and data involved in the system. DFDs are used for modeling and analyzing the flow of data in a system, particularly in software development, business process re-engineering, and systems analysis.

Purpose of a DFD

A Data Flow Diagram (DFD) is a graphical representation used to visualize the flow of data within a system. It illustrates how data moves from input to output, through processes, data stores, and external entities. The primary purposes of a DFD include:

1. Understanding and Analyzing the System:

   DFDs provide a clear and concise way to understand the workings of a system by breaking down its processes and showing how data flows between them. This helps in analyzing the system's functionality and identifying areas for improvement.

2. Documentation:

   DFDs serve as a useful documentation tool that can be referenced by developers, analysts, and other stakeholders. It ensures that everyone has a consistent understanding of the system's data flow.

3. Communication:

   DFDs act as a visual communication tool that can be easily understood by non-technical stakeholders, such as business managers or clients. This helps in ensuring that all parties have a clear understanding of the system's operations.

4. Design and Development:

   During the design and development phases, DFDs help in planning the architecture of the system. They aid in identifying necessary components, their interactions, and data dependencies, thus guiding the development process.

5. Error Detection and Troubleshooting:

   By visualizing the flow of data, DFDs can help in identifying bottlenecks, redundancies, or errors in the system. This makes it easier to troubleshoot issues and optimize the system.

6. Requirements Specification:

   DFDs can be used during the requirements gathering phase to depict what the system should do. This helps in ensuring that the system meets the user's needs and aligns with business objectives.

7. System Validation:

   DFDs help validate that the design of a system meets the requirements specified by stakeholders. By mapping out the data flow, it can be confirmed whether all required data inputs, processes, and outputs are accounted for.

8. Facilitating System Integration:

   When integrating multiple systems or subsystems, DFDs help in understanding how data flows between them. This ensures seamless integration and proper data exchange, reducing integration errors.

9. Simplifying Complex Systems:

   DFDs break down complex systems into simpler, manageable components. This hierarchical decomposition helps in understanding each part of the system in detail without being overwhelmed by its complexity.

10. Support for Process Improvement:

    By providing a clear view of existing processes and data flows, DFDs support efforts to improve processes. They help in identifying inefficiencies, redundant steps, and opportunities for process automation.

11. Training and Onboarding:

    DFDs serve as an educational tool for new team members. They provide a visual representation of the system, making it easier for newcomers to understand how the system operates and how data flows within it.

12. Regulatory Compliance:

    In industries with strict regulatory requirements, DFDs can help ensure that data flows and processes comply with legal and regulatory standards. They provide a documented trail of data movement and processing.

13. Project Management:

    DFDs aid project managers in planning, monitoring, and controlling project activities related to system design and implementation. They help in visualizing the project scope and identifying dependencies and critical paths.

14. Risk Management:

By understanding the data flow and system processes, potential risks such as data breaches, loss, or inaccuracies can be identified. DFDs help in implementing measures to mitigate these risks effectively.

15. Enhancing Collaboration:

DFDs promote better collaboration among cross-functional teams, including developers, analysts, designers, and business stakeholders. The visual nature of DFDs ensures that everyone is on the same page regarding system functionality.

16. Cost Estimation:

During the planning phase, DFDs assist in estimating the cost of system development and maintenance by outlining the required processes, data stores, and interactions, helping in budget allocation and resource planning.

17. System Maintenance and Upgrades:

For existing systems, DFDs help in understanding the current data flows and processes, which is crucial when making updates or performing maintenance tasks. They provide a reference to ensure changes do not disrupt existing functionalities.

18. User Requirements Capture: DFDs help in capturing and visualizing user requirements more accurately by showing how user inputs are processed and transformed into outputs. This ensures that the final system aligns closely with user expectations.

19. Benchmarking and Performance Measurement:

DFDs can be used to benchmark existing processes against industry standards or best practices. They help in measuring performance and identifying areas for performance enhancement.

20. Modeling Different Scenarios:

DFDs allow for the modeling of different scenarios and what-if analyses, helping stakeholders understand the impact of potential changes or new features on the overall system.

21. Supporting Agile Development:

In Agile development environments, DFDs can be used to quickly map out and iterate on system processes and data flows. This helps teams to adapt to changes in requirements and continuously improve the system design throughout the development cycle.

22. Facilitating User Experience (UX) Design:

DFDs help UX designers understand how users interact with the system and how their inputs are processed. This understanding is critical for designing intuitive and user-friendly interfaces that align with the underlying data flows and processes.

23. Improving Data Governance:

DFDs provide a clear visualization of where and how data is stored, processed, and transmitted within a system. This is essential for establishing effective data governance practices, ensuring data quality, security, and compliance with data protection regulations.

Overall, DFDs are a valuable tool in systems analysis and design, offering a clear and organized method for representing the flow of data through various processes within a system.

4. CODING

1. Where are the old left-handed people?

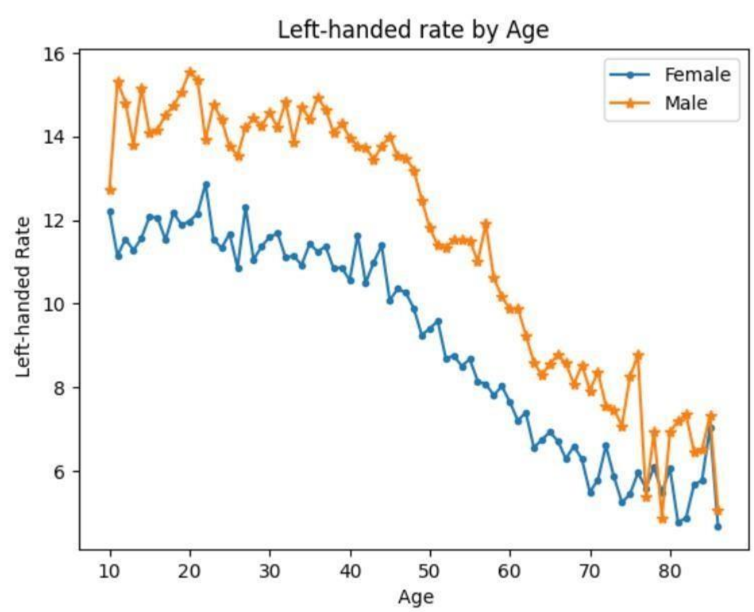
In this notebook, we will explore this phenomenon using age distribution data to see if we can reproduce a difference in average age at death purely from the changing rates of left-handedness over time, refuting the claim of early death for left-handers. This notebook uses `pandas` and Bayesian statistics 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 Libraries
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
%matplotlib inline
data_url_1= "https://gist.githubusercontent.com/mbonsma/8da0990b71ba9a09f7de395574e54df1/raw/aec88b30af87fad8d45da7e774223f91dad09e88/lh_data.csv"
lefthanded_data = pd.read_csv(data_url_1)
```

```
fig, ax = plt.subplots() # create figure and axis objects
ax.plot('Age', 'Female', data = lefthanded_data, marker = '.') # plot "Female" vs. "Age"
ax.plot('Age', 'Male', data = lefthanded_data, marker = '*') # plot "Male" vs. "Age"
ax.legend() # add a Legend
ax.set_xlabel(' Age') # Label x-axis
ax.set_ylabel(' Left-handed Rate') # Label y-axis
plt.title("Left-handed rate by Age") # Add title to the plot
plt.show() # Display the figure
```



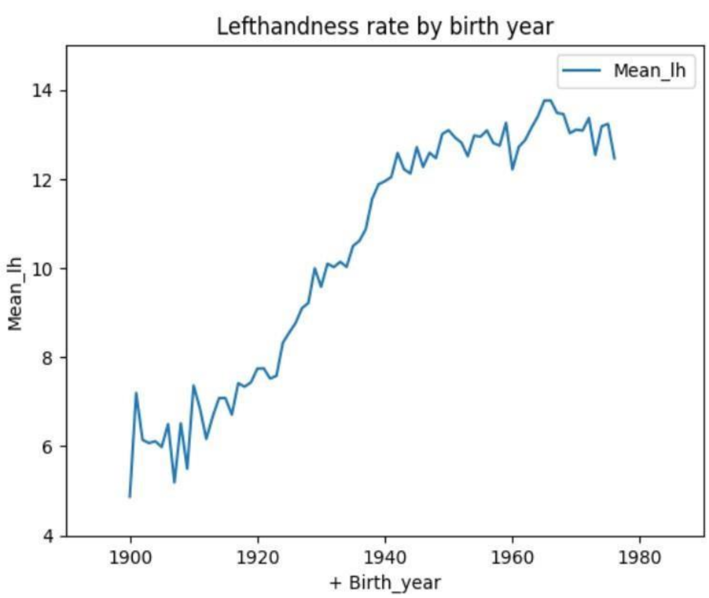
2. Rates of left-handedness over time

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
lefthanded_data['Birth_year'] = 1986 - lefthanded_data['Age']
# create a new column for the average of male and female
lefthanded_data['Mean_lh'] = lefthanded_data[['Male', 'Female']].mean(axis=1)
```

```
# create a plot of the 'Mean_Lh' column vs. 'Birth_year
fig, ax = plt.subplots()
ax.plot("Birth_year", 'Mean_lh', data = lefthanded_data) # plot Birth_year vs. Mean_Lh
#Set origin to zero
ax.set_xlim(1890, 1990) # Set x-axis range
ax.set_ylim(4, 15) # Set y-axis range
ax.legend()
ax.set_xlabel('+ Birth_year') # Label x-axis
ax.set_ylabel('Mean_lh') # Label y-axis
plt.title('Lefthandness rate by birth year') # Plot title
plt.show() #displays plot
```



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.

```
# Creating P(LH/A) function
def P_lh_given_A(ages_of_death, study_year = 1990):
  #Use the mean of the 10 Last and 18 first points for Left-handedness rates before and after the start
  early_1900s_rate = lefthanded_data['Mean_lh'] [-10:].mean()
  late_1900s_rate = lefthanded_data['Mean_lh'][:10].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] = early_1900s_rate / 100
  P_return[ages_of_death< youngest_age] = late_1900s_rate / 100
  P_return[np.logical_and((ages_of_death <= oldest_age), (ages_of_death >= youngest_age))] = middle_rates / 100
  return P_return
```

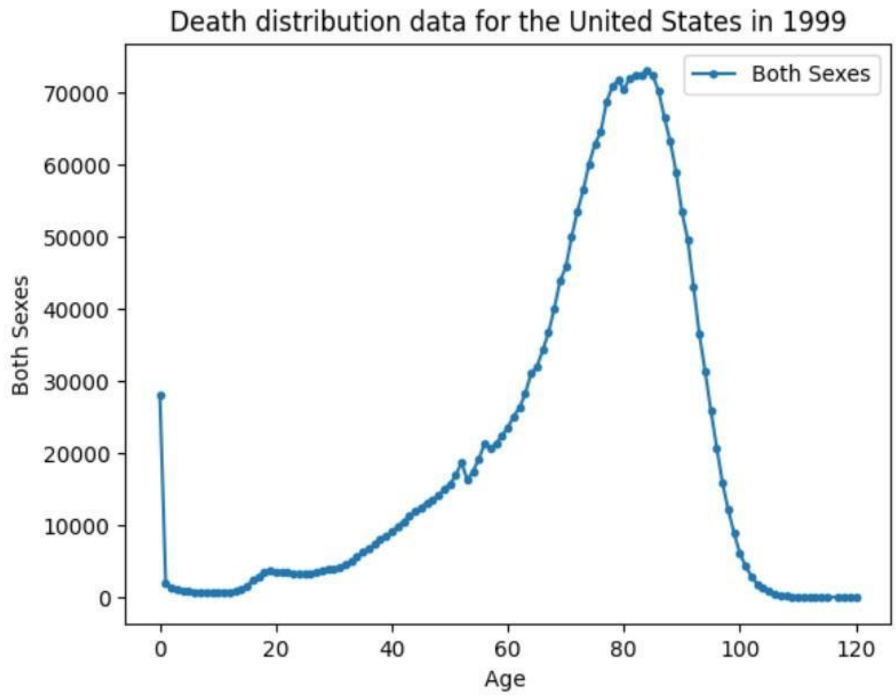
4. When do people normally die?

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_vs00199_table310.tsv"
#Load death distribution data
death_distribution_data = pd.read_csv(data_url_2, sep='\t', skiprows=[1])
#drop NaN values from the Both Sexes column
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('Age', 'Both Sexes', data = death_distribution_data, marker='.') # plot 'Both Sexes' vs. 'Age'
ax.legend()
plt.title('Death distribution data for the United States in 1999')
ax.set_xlabel('Age ')
ax.set_ylabel('Both Sexes')
plt.show()
```

```
plt.show()
```



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"""
    # multiply number of dead people by P_lh_given_A
    p_list = death_distribution_data['Both Sexes']* P_lh_given_A(death_distribution_data['Age'], study_year)
    #calculate the sum of p_List\
    p = np.sum(p_list)
    # normalize to total number of people (sum of death_distribution_data['Both Sexes'])
    return p/np.sum(death_distribution_data['Both Sexes'])
    print(P_lh(death_distribution_data))
0.07766387615350638
```

```
0.07766387615350638
```

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['Both Sexes'] [ages_of_death] / np.sum(death_distribution_data['Both Sexes'])
      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
      return P_lh_A*P_A/P_left
```

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

And now for right-handers.

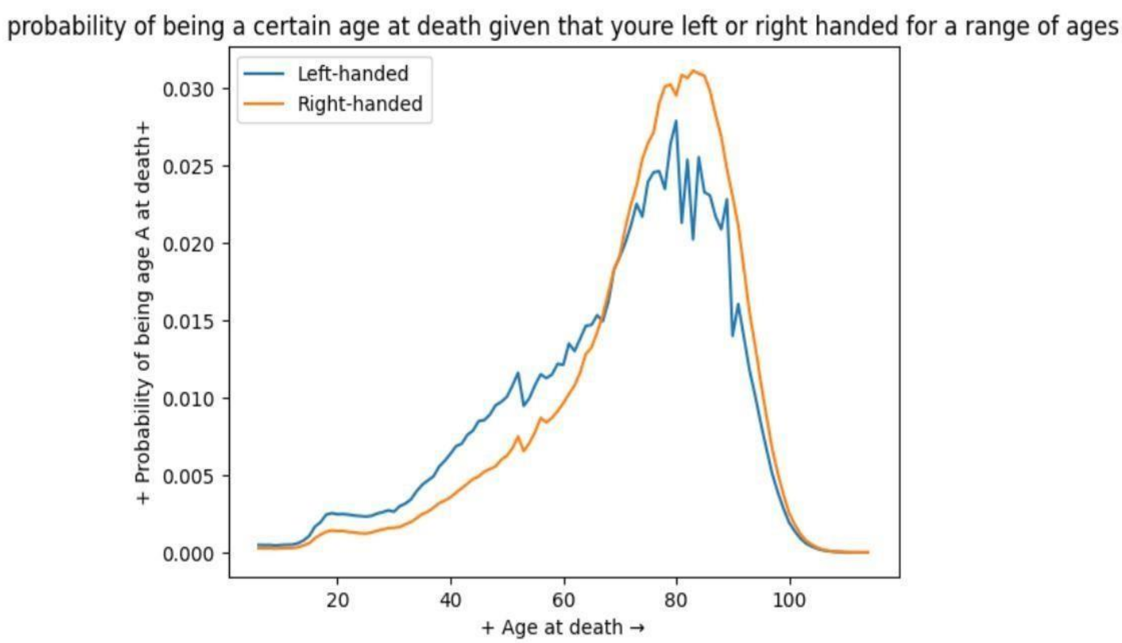
```
def P_A_given_rh(ages_of_death, death_distribution_data, study_year = 1990):
  """ The overall probability of being a particular age_of_death given that you're right-handed """
  P_A = death_distribution_data['Both Sexes'] [ages_of_death]/np.sum(death_distribution_data['Both Sexes'])
  P_right = 1 - P_lh(death_distribution_data, study_year) # 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_A1P_Lh_A
  return P_rh_A*P_A/P_right
```

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
    plt.title('probability of being a certain age at death given that youre left or right handed for a range of ages')
    ax.set_xlabel ("+ Age at death →")
    ax.set_ylabel("+ Probability of being age A at death+")
    plt.show()
```



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.

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

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

```python
# 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(left_handed_probability))
average_rh_age =  np.nansum(ages*np.array(right_handed_probability))

# print the average ages for each group
# ... YOUR CODE FOR TASK 9 ...
print("Average age of lefthanded" + str(average_lh_age))
print("Average age of righthanded" + str(average_rh_age))

# print the difference between the average ages
print("The difference in average ages is " + str(round(average_rh_age - average_lh_age, 1)) + " years.")
```

```
Average age of lefthanded67.24503662801027
Average age of righthanded72.79171936526477
The difference in average ages is 5.5 years.
```

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, 2018)
    right_handed_probability_2018 = P_A_given_rh(ages, death_distribution_data, 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 " +
          str(round(average_rh_age_2018 - average_lh_age_2018, 1)) + " years.")
```

```
The difference in average ages is 2.3 years.
```

APPENDICES

Appendix A: Data Sources

Provide a description of the data sources used, including URLs and brief descriptions.

Appendix A1: Left-Handedness Data

- URL: Left-Handedness Data
- Description: This dataset contains information on the rates of left-handedness by age and gender.

Appendix A2: Death Distribution Data

- URL: Death Distribution Data
- Description: This dataset contains the death distribution data for the United States in 1999, categorized by age and gender.

Appendix A3: Derived Right-Handedness Data

- Description: The right-handedness data is derived from the left-handedness data. Since the data assumes binary handedness, the probability of being right-handed is calculated as $1-P(\text{left-handed})$.

References

- Google Data Analytics Professional certificate - https://www.coursera.org/professional-certificates/google-data-analytics
- A case study on left-handedness and life expectancy - https://archpublichealth.biomedcentral.com/articles/10.1186/s13690-023-01156-6#:~:text=Using%20a%20simulation%20that%20assumed,to%20changes%20in%20the%20rate
- The development of handedness in left/right asymmetry - https://pubmed.ncbi.nlm.nih.gov/2209459/
- MedTourEasy official website - https://www.medtoureasy.com/
- Data Analytics: Become A Master In Data Analytics, by Richard Dor