**Abstract**

The US Department of Energy distributes large amounts of funds to early career scientists every year. This study aims to study the distribution of the recipients of such funding. It delves into the kind of institutions receiving funding, the gender disparity among recipients, the locations where most recipients are from, and variance in the number of recipients under each program area. These results can help us better understand the diverse groups being funded and where more funding needs to go.

**Introduction**

Lots of money is provided as funds to early career scientists by the US Department of Energy. Awardees are selected based on peer review by outside scientific experts. Often, diversity of recipients is not really considered. Looking at the diversity of recipients can help us understand the groups we need to fund more. Understanding the groups that are a minority and why they are a minority is significant in ensuring that all groups are properly represented.

**Method**

The code scrapes data from a list of pdfs linked on the award website. The data is about the recipients, the institution they belong to, their gender, and their location.

The libraries used in this code are:

* requests: to fetch web pages and download files
* BeautifulSoup: to parse HTML contents to find elements
* os: to perform file system operations like creating directories or checking if a file exists
* urllib.parse.urljoin: form URLs by joining relative paths to URLs
* tqdm: progress bar for loops
* pdfplumber: extract text from PDF files
* re: regular expressions for searching and extracting patterns from text
* pandas: data manipulation and storage in tabular manner
* gender\_guesser: library that guesses gender based on first name
* logging: to log messages for debugging and progress tracking
* scholarly: access Google Scholar to retrieve information

The code follows four steps to extract and classify the data from the pdf files.

* Step 1: Fetching PDF links
  + Send a GET request to the base URL
  + Raise an exception if there was an error accessing the URL
  + Use BeautifulSoup to parse through the content of the website and find the links for pdf files
  + Storing the links in a list
  + download\_pdf method:
    - Takes in two parameters: the URL of the file to be downloaded and the path it must be saved in
    - Makes a request to the URL, stream=True ensures that data is downloaded in chunks
    - If the URL is found, it opens a new file at that path in “wb” mode
    - PDF is downloaded in chunks. Chunk size is defined by chunk\_size
    - Each chunk is written to the file
  + download\_pdfs method takes in a list of PDF URLs and the directory the PDFs must be saved at
* Step 2: Extracting text from the PDFs
  + extract\_text\_to\_txt method
    - Opens a PDF file
    - Going through each page of the PDF and extracting text from it
    - Writing the extracted text to a new file and saving it
  + extract\_texts
    - Going through each pdf and extracting the text from it
* Step 3: Extracting Names and Institutions
  + Initializing the gender detector and setting it to not be case sensitive
  + Defining lists of national labs, public universities, private universities, and ivy leagues
  + categorize\_institution method
    - Categorizes each institution accordingly
  + classify\_gender method
    - Used gender\_guesser to guess the gender of the awardee
  + extract\_entries\_from\_text method
    - Extracts the name of the awardee and their institution from the given text
    - Split
  + extract\_entries\_from\_text method
  + Extracts the name of the awardee and their institution from the given text
    - Splits each PDF file based on new line character (new line character indicates a new page)
    - A list called entries is created to store (Name, Institution) pairs
    - Storing a list of keywords to avoid checking for departments of an institution
    - Checking for lines that start with Dr.
    - Extracting their name
    - Looking for a line with a zip code and moving one or two lines above that to find the institution name
  + extract\_and\_classify\_data method
    - Identifies text files within the directory and loops through each of them
    - Extracts the names and institutions from the text files
    - Guesses the gender based on first name
    - Classifies institution
    - Creates a DataFrame to store each entry
* Step 4: Execution
  + Calling all the methods we defined
  + Converting the DataFrame to a .csv file
    - index=False ensures that the Data Frame’s index isn’t included in the csv file
  + if \_\_name\_\_ == "\_\_main\_\_“
    - Ensures that the main method is called only when the script is run directly, not when it is imported as a module in another script

**Results and Discussion**

**Conclusion**

**Acknowledgements**