Considering Bias in Data

Homework #2

The goal of this assignment is to explore the concept of bias in data using Wikipedia articles. This assignment will consider articles on political figures from different countries. For this assignment, you will combine a dataset of Wikipedia articles with a dataset of country populations, and use a machine learning service called ORES to estimate the quality of each article.

You are expected to perform an analysis of how the coverage of politicians on Wikipedia and the quality of articles about politicians varies among countries. Your analysis will consist of a series of tables that show:

1. The countries with the greatest and least coverage of politicians on Wikipedia compared to their population.
2. The countries with the highest and lowest proportion of high quality articles about politicians.
3. A ranking of geographic regions by articles-per-person and proportion of high quality articles.

You are also expected to write a short reflection on the project that focuses on how both your findings from this analysis and the process you went through to reach those findings helps you understand the causes and consequences of biased data in large, complex data science projects.

Step 1: Getting the Article and Population Data

The first step is getting the data, which lives in several different places. You will need data that lists Wikipedia articles of politicians and data for country populations.

The Wikipedia Category:Politicians by nationality was crawled to generate a list of Wikipedia article pages about politicians from a wide range of countries. This data is in the homework folder as `politicians_by_country.AUG.2024.csv`.

The population data is available in CSV format as `population_by_country_AUG.2024.csv` from the homework folder. This dataset was downloaded from the world population data sheet published by the Population Reference Bureau.

Some Considerations: You should be a little careful with the data. Crawling Wikipedia categories to identify relevant page subsets can result in misleading and/or duplicate category labels. Naturally, the data crawl attempted to resolve these, but not all may have

been caught. As well, Wikipedia categories are folksonomic, meaning there is very little control over how they are applied to pages. This means that the set of pages is very likely some kind of subset, and may have pages that are not actually about individual politicians. You should look for any data inconsistencies and document how you handle inconsistencies that you find.

The `population_by_country_AUG.2024.csv` contains rows that provide cumulative regional population counts. These rows are distinguished by having ALL CAPS values in the 'geography' field (e.g. AFRICA, OCEANIA). These rows should not match the country values in `politicians_by_country.AUG.2024.csv`, but you will want to retain them so that you can report coverage and quality by region as specified in the analysis section below.

Step 2: Getting Article Quality Predictions

Now you need to get the predicted quality scores for each article in the Wikipedia dataset. We're using a machine learning system called ORES. This was originally an acronym for "Objective Revision Evaluation Service" but was simply renamed “ORES”. ORES is a machine learning tool that can provide estimates of Wikipedia article quality. The article quality estimates are, from best to worst:

1. FA - Featured article
2. GA - Good article (also known as A-Class)
3. B - B-Class article
4. C - C-Class article
5. Start - Start-class article
6. Stub - Stub-class article

These class labels were learned based on articles in Wikipedia that were peer-reviewed using the Wikipedia content assessment procedures.These quality classes are a subset of quality assessment categories developed by Wikipedia editors.

ORES requires a specific revision ID of an article to be able to make a label prediction. You can use the API:Info request to get a range of metadata on an article, including the most current revision ID of the article page.

Putting this together, to get a Wikipedia page quality prediction from ORES for each politician’s article page you will need to:

a) read each line of `politicians_by_country.AUG.2024.csv`,
b) make a page info request to get the current page revision, and
c) make an ORES request using the page title and current revision id.

The homework folder contains example code in notebooks to illustrate making a page info request and making an ORES request. This sample code is licensed CC-BY. You can reuse any of the sample code, but make sure you follow the license requirements.

Access to the ORES API will require that you request an API access key. The sample code for making ORES requests includes links to information about how to request a key. A "best practice" for any code that requires an API key is to make sure that the key does not appear in the plain text of the code or notebook. One approach is to embed the key as an environment variable and retrieve the key from that variable. Another approach is to use a code based key manager that stores keys on your local machine. The `apikeys` user module is used in the ORES example code.

As you are running your code, it is possible that you will be unable to get a score for a particular article. If that happens, make sure to maintain a log of articles for which you were not able to retrieve an ORES score. This log can be saved as a separate file, or (if it's only a few articles), simply printed and logged within the notebook. The choice is up to you.

Your notebook should compute and print the score error rate. The error rate is the ratio of the number of articles for which you were not able to get a score divided by the total number of articles. If your request error rate is higher than 1%, then you should review your code, determine what is going wrong, fix it, and rerun your score collection.

Step 3: Combining the Datasets

Some processing of the data will be necessary. In particular, after retrieving and including the ORES data for each article, you'll need to merge the wikipedia data and population data together. Both have fields containing country names for just that purpose. After merging the data, you'll invariably run into entries which cannot be merged. Either the population dataset does not have an entry for the equivalent Wikipedia country, or vice-versa.

Identify all countries for which there are no matches and output a list of those countries, with each country on a separate line called:

```
wp_countries-no_match.txt
```

Consolidate the remaining data into a single CSV file called:

```
wp_politicians_by_country.csv
```

The CSV file should include the following columns of data:

Column
country

region
population
article_title
revision_id
article_quality

Step 4: Analysis

Your analysis will consist of calculating total-articles-per-capita (a ratio representing the number of articles per person) and high-quality-articles-per-capita (a ratio representing the number of high quality articles per person) on a country-by-country and regional basis.

In this analysis a country can only exist in one region. The `population_by_country_AUG.2024.csv` actually represents regions in a hierarchical order. For your analysis always put a country in the closest (lowest in the hierarchy) region.

For this analysis you should consider "high quality" articles to be articles that ORES predicted would be in either the "FA" (featured article) or "GA" (good article) classes.

Also, keep in mind that the `population_by_country_AUG.2024.csv` file provides population in millions. The calculated proportions in this step are likely to be very small numbers. Think carefully about how to represent the results.

Step 5: Results

Your results from this analysis will be produced in the form of data tables. You are being asked to produce six total tables, that show:

1. Top 10 countries by coverage: The 10 countries with the highest total articles per capita (in descending order) .
2. Bottom 10 countries by coverage: The 10 countries with the lowest total articles per capita (in ascending order) .
3. Top 10 countries by high quality: The 10 countries with the highest high quality articles per capita (in descending order) .
4. Bottom 10 countries by high quality: The 10 countries with the lowest high quality articles per capita (in ascending order).

5. Geographic regions by total coverage: A rank ordered list of geographic regions (in descending order) by total articles per capita.
6. Geographic regions by high quality coverage: Rank ordered list of geographic regions (in descending order) by high quality articles per capita.

Embed these tables in your notebook.

Step 6: Write-up, Reflections and Implications

Write several paragraphs that you will include in your README. Your README should include a section header called "Research Implications" after which you will include your write-up paragraphs. One of your paragraphs should reflect on what you have learned, what you found, what (if anything) surprised you about your findings, and/or what theories you have about why any biases might exist (if you find they exist). In addition to any reflections you want to share about the process of the assignment, please respond (briefly) to **at least three** of the questions below:

1. What biases did you expect to find in the data (before you started working with it), and why?
2. What (potential) sources of bias did you discover in the course of your data processing and analysis?
3. What might your results suggest about (English) Wikipedia as a data source?
4. What might your results suggest about the internet and global society in general?
5. Can you think of a realistic data science research situation where using these data (to train a model, perform a hypothesis-driven research, or make business decisions) might create biased or misleading results, due to the inherent gaps and limitations of the data?
6. Can you think of a realistic data science research situation where using these data (to train a model, perform a hypothesis-driven research, or make business decisions) might still be appropriate and useful, despite its inherent limitations and biases?
7. How might a researcher supplement or transform this dataset to potentially correct for the limitations/biases you observed?

This section doesn't need to be particularly long or thorough. Your reflection on the assignment and your answers to the questions above are probably worth a short paragraph for each.

Step 7: Prepare Documentation

Document your code, your notebooks, and write up your repository documentation. Remember to follow best practices for documenting your project.

Step 8: Prepare and Submit your Repository

You will complete this homework by submitting a link to a repository. For this assignment you need to:

1. Create a repository folder named `data-512-homework_2`
2. Copy your notebook(s) into the folder.
3. Copy the CSV data files into the folder.
4. Copy any intermediary/supplementary data files into the folder.
5. Complete and add your README in .txt or .md format and LICENSE file. Your README should document schemas and intermediary files.
6. Set permissions on your repository folder to share it with both TAs and the Instructor. You should share it with the email addresses they listed on the syllabus.
7. Submit the link to your repository through the Homework 2 submission form on Canvas