MCIS6273 Data Mining (Prof. Maull) / Fall 2023 / HW2

Points Possible	Due Date	Time Commitment (estimated)
40	Sunday, December 10 @ Midnight	*up to* 24 hours

- **GRADING:** Grading will be aligned with the completeness of the objectives.
- **INDEPENDENT WORK:** Copying, cheating, plagiarism and academic dishonesty *are not tolerated* by University or course policy. Please see the syllabus for the full departmental and University statement on the academic code of honor.

OBJECTIVES

- Perform data engineering on assignment dataset.
- [supervised learning] Perform K-means analysis on real-world data.
- [supervised learning/advances] Listen to this podcast about the future of search and advances in supervised learning.

WHAT TO TURN IN

You are being encouraged to turn the assignment in using the provided Jupyter Notebook. To do so, make a directory in your Lab environment called `homework/hw2`. Put all of your files in that directory. Then zip that directory, rename it with your name as the first part of the filename (e.g. `maull_hw2_files.zip`), then download it to your local machine, then upload the `.zip` to Blackboard.

If you do not know how to do this, please ask, or visit one of the many tutorials out there on the basics of using zip in Linux.

If you choose not to use the provided notebook, you will still need to turn in a `.ipynb` Jupyter Notebook and corresponding files according to the instructions in this homework.

ASSIGNMENT TASKS

(30%) Perform data engineering on assignment dataset.

In this part of the assignment we will be introduced to a specific dataset that will be very interesting and unique.

In the US (and most other counties), regulatory agencies control and inspect food and other products that come into the country and or otherwise imported for sale by wholesalers and retailers.

One such area that we are going to explore is in what is called "import refusal". Import refusal refers to the regulary mechanism which rejects products from importation into the coutry by way of the US FDA (Food and Drug Administration) – specificially if an inspected regulated product is not in compliance with FDA standards, the owner/cosignee is allowed to respond to the refusal and either show evidence that the product is in compliance (i.e. the FDA made a mistake) or produce a plan to bring the product into compliance – otherwise the product is exported back to the owner/cosignee or destroyed.

The mechanism used to track these refusals is called the IRR or "Import Refusal Report".

You can (and might want to) read more about this IRR here:

- https://www.fda.gov/industry/fda-import-process/import-refusals

We will explore the data in this report with some unsupervised learning mechanisms, but before we do, will do perform some standard data engineering to bring the report into alignment with the tools required to do what we'd like later.

§ Task: Load and unzip the compressed ZIP import refusal report for 2014-present. You can use Jupyter "magics" (the easiest way), or you can write Python code to load the ZIP file, and unzip it. If you choose that method, you may like to make use of:

- Python Requests library
- Python zip/unzip `zipfile` library

You will find the file on this page:

- https://www.accessdata.fda.gov/scripts/ImportRefusals/index.cfm

And to get the URL of the ZIP file, select the 2014-present file and open your browser developer tools, click Download and see the actual URL (watch network traffic tab), or (less fun), use this URL:

- 2014-present.zip

§ Task: Now that you have the data, you will read the CSV file. Produced a file called `"country_violations_2014-2023.csv"` which contains just the counts of the violations **grouped by** `['ISO_CNTRY_CODE','PROVINCE_STATE','CITY_NAME']`.

- you will need to use `groupby().count()` and restrict the columns to a single column (`ENTRY_NUM` will do) using `.loc()`

§ Task: Produce a CSV file which includes the same data as `"country_violations_2014-2023.csv"` except it groups by `'YEAR', 'MONTH', 'ISO_CNTRY_CODE','PROVINCE_STATE','CITY_NAME'`. You will might like to take the column `REFUSAL_DATE` and break it into a `YEAR` and `MONTH` column of its own, then do the grouping. You new file should be called `"country_violations_year_month_2014-2023.csv"`.

§ Task: Which city, country, province had the most violations in a single month? How many? Which month and year?

§ Task: What are the 10 most frequent products in the IRR for 2018 (using `'PRDCT_CODE_DESC_TEXT'`)?

§ Task: BONUS (+1 point) What was the company associated with the largest violation in a single month?

(50%) [supervised learning] Perform K-means analysis on real-world data.

The goal of cluster analysis is to extract data patterns from data that does not contain (or is not used for) training instances. We call this *unsupervised learning*, since there is no training data to build models from.

Instead, we use some of the commonly studied *distance metrics* to develop a notion of similarity. Indeed, we are trying to optimize for instances of a cluster to maximize *intra*-cluster (within-cluster) similarity, while *inter*-cluster similarity is minimized – put another way, instances that belong to a cluster should look close to one another.

There are many clustering algorithms, but one of the most robust and useful is the K-means algorithm.

§ Task: Prepare the data such that you have three datasets where the `REFUSAL_CHARGES`, `ISO_CNTRY_CODE` and `CITY_NAME` are the columns (features).

You will need to use the `LabelBinarizer()` (for countries and cities) and `MultiLabelBinarizer()` (for the charges) of the sklearn libraries.

See:

- `sklearn.preprocessing.LabelBinarizer()`
- `sklearn.preprocessing.MultiLabelBinarizer()`

§ Task: Merge the three Dataframes above into one – the final Dataframe should have 435 columns. These represent the features that will allow clustering to occur. This will allow us to see the clusters that emerge along those categories of features in the data. With a bit more information, we might come to the conclusion that some of these features should be removed.

§ Task: You will now take the dataset from the first part and begin the process of clustering.

To be successful, please study the following:

- K-Means in scikit-learn
- K-Means example notebook

You will set three K to 5, 10 and 12. You will need to report the centroids for each cluster and in words how you would describe that cluster. I will give more guidance on this.

§ Task: *(Perform elbow analysis to find optimal cluster size)*

In the previous part, we chose the cluster size K. Another way to do this is to analyze the change in within cluster sum of squares and determine when such value fails to change significantly. In other words, when the addition of another cluster fails to significantly change the within cluster sum of squares, then you can be confident more clusters won't make a difference (increasing K will no longer be relevant).

This is often referred to as "Elbow Analysis" or the "Elbow Method" because you will visually find the elbow in a plot of the sum of squares and choose K based on that.

Study the following code, implement it, and find the optimal K based on it.

Your answer must include:

- the elbow graph
- the optimal K
- the reanalysis of the previous answer based on the optimal K (re-run your clusters and report their centroid characteristics)

Here is the code to help you:

```
max_clusters = 15
css = [] # within cluster sum of squares

for k in range(1,max_clusters):
  kmeans = KMeans(n_clusters=k, 'k-means++', max_iter=200, n_init=10, random_state=0)
  kmeans.fit(d) # where d is the dataset you have standardized in the first part of this
  css.append(kmeans.inertia_)

# now make a line plot of all the values in css
...
```

(20%) [supervised learning/advances] Listen to this podcast about the future of search and advances in supervised learning.

We are, as you know, entering an entirely new era of human-machine interaction.

LLMs, popularized by ChatGPT, Bard and others, are pushing new paradigms of interaction with machines, fueling what many are calling a breakthrough in Artificial Intelligence unlike any seen in prior advances in the field (AI, has after all, been the intense study of computer scientists since the 1950s).

One obvious area where this is going to be immediately obvious is in search. For some time, we have been using search engines with the "poke and hope" method – typing in some keywords and hoping we get what we are looking for. Have you noticed that this method doesn't work that well? Have you also noticed that sometimes you spend more time trying to find the "right" keywords instead of getting to what you want with just the words you have to express what you reall mean?

One reason for this is that our mental models for what we are searching for are incomplete – we often don't have enough domain knowledge to phrase the question in a way that would yield answers even remotely close to what we would like …

That is all about to change.

You will listen to this 59 minute podcast interview with the CEO of
Perplexity AI – a company focused on using AI to improve learning. Search might be the first test case to demonstrate what is coming in many other areas of the interesting use cases of AI.

Listen to this podcast:

- Machine Learning Street Talk (MLST): *Perplexity AI*: The Future of Search; May 8, 2023; Interview with Aravind Srinivas, CEO and co founder of Perplexity AI. You will find a variety of sources of the interview (pick one):
  - Apple Podcasts
  - Player.fm
  - Spotify
  - Youtube (**no ads**, no tracking, viewed through DDG)
  - Youtube direct (ads, account tracking, etc.)

Find out what all the hullabuloo is about.

§ Task: Summarize the 3-5 main takeaways of the interview? Be brief.

§ Task: Provide 3-5 sentences expressing your reactions to the interview? Be direct, succinct and precise.

§ Task: What *one thing* did you find most interesting or surprising in the interview?

§ Task: Provide *one* criticism or concern of the work of Perplexity AI, and expand on that criticism with a few sentences explaining why you feel your criticism/concern is warranted.