Homework Programming Assignment 4: Clustering

Due Time: November 4, 2019, 3:00PM

Introduction

In this assignment, you will implement the K-Means clustering algorithm, as well as use clustering to solve real-world problems.

1. Implement K-means algorithm

Please use ***Python 3.6 or 3.7*** to implement your homework assignment.

1.1 Data Description

Both blackbox41 and blackbox42 dataset contains 4000 **two-dimensional** data points. **Each of them contains 4 clusters.** What you need to do is to implement K-Means algorithm to find these 4 clusters among these data points.

1.1.1 Blackbox

The same as last assignment, you are given `Blackbox41.pyc` instead of ready-made data, and you need to generate your own data from that blackbox. When you `ask` the blackbox, it will return an all 4000 data points as type ndarray. Your python script submitted to Vocareum **must import both** blackbox41 and blackbox42 even though blackbox42 is not provided to you.

```
from Blackbox41 import Blackbox41
from Blackbox42 import Blackbox42
```

Remember that your code will be run with 2 different boxes. One way to parse the input argument is

```
blackbox = sys.argv[-1]
if blackbox == 'blackbox41':
    bb = Blackbox41()
elif blackbox == 'blackbox42':
    bb = Blackbox42()
else:
    print('invalid blackbox')
    sys.exit()
```

and then use `bb` as data source. When you develop your algorithm, you only need to care about blackbox41, but you also need to import and add additional parsing logic for blackbox42 since blackbox42 will be used to test your code on Vocareum.

1.1.2 Program

Your program will be run in the following way:

```
python3 Kmeans.py blackbox41
=> results_blackbox41.csv
```

When we grade your model with hidden blackbox42, it should be run:

```
python3 Kmeans.py blackbox42
=> results_blackbox42.csv
```

The `results_blackbox41.csv results_blackbox42.csv` contains “cluster” labels of each data point you K-means algorithm generated, it should have the following format:

```
0
1
2
1
…
0
3
```

In your implementation, ***please do not use any existing machine learning library call***. You must implement the algorithm yourself. Please develop your code yourself and do not copy from other students or from the Internet.

When we grade your algorithm, we will use a hidden blackbox42. Your code will be autograded for technical correctness. Please name your file correctly, or you will wreak havoc on the autograder. **The maximum running time is 3 minutes.**

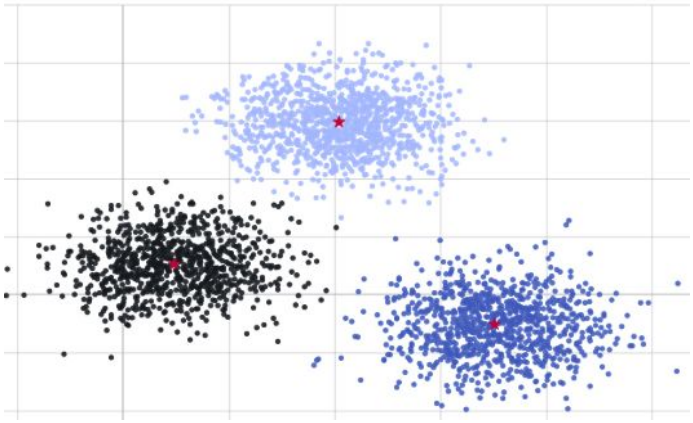
1.2. Submission:

Submit your code to Vocareum

- Submit `KMeans.py` to ***Vocareum***
- After your submission, Vocareum would run two scripts to test your code, a submission script and a grading script. The submission script will test your code with ***only blackbox41***, while the grading script will test your code with another ***blackbox42***.
- The program will terminate and fail if it exceeds the ***3 minutes*** time limit.
- After the submission script/grading script finishes, you can view your submission report immediately to see if your code works, while the grading report will be released after the deadline of the homework.
- You don't have to keep the page open while the scripts are running.

Submit your report to Blackboard

- Create a single **`.zip`** (`Firstname_Lastname_HW4.zip`) which contains:
    - `KMeans.py`
    - `Report.pdf`, a ***brief*** report contains your clustering graph (sample points and centroid of each cluster) for blackbox41**.** Example graph:



- Submit your zipped file to the blackboard.

1.3. Rubric:

50 points in total

- Program correctness(30 points): program always works correctly and meets the specifications within the time limit
- Documentation(10 points): code is well commented and submitted graph is reasonable
- Performance (10 points): the classifier has reasonable performance

2. User Analysis Using Clustering

2.1 Introduction

In this exercise, you'll do a data challenge as a data analyst. Specifically, your task is to apply clustering on real-world users data of Yelp to identify different clusters of users. In this part of the assignment, **you are welcome to use existing machine learning libraries**(e.g. sklearn).

Yelp dataset

Yelp dataset can be found on https://www.kaggle.com/yelp-dataset/yelp-dataset. There are various datasets such as business, reviews and check-ins. But we'll focus on the "users" dataset (yelp_academic_dataset_user.json) this time. Description of attributes is as below.

User_id: user id
Name: user name
Review_count: number of user's reviews
Yelping_since: sign up date of user
Useful: count of receiving "useful" feedback (for the user's reviews)
Funny: count of receiving "funny" feedback (for the user's reviews)
Cool: count of receiving "cool" feedback (for the user's reviews)
Elite: years that the user is part of elite members of Yelp
Friends: list of friends' user id
Fans: number of fans
Average_stars: average star of reviews
Compliment_hot: count of receiving "hot stuff" compliments (for the user)
Compliment_more: count of receiving "write more" compliments (for the user)
Compliment_profile: count of receiving "like your profile" compliments (for the user)
Compliment_cute: count of receiving "cute pic" compliments (for the user)
Compliment_list: count of receiving "great lists" compliments (for the user)
Compliment_note: count of receiving "just a note" compliments (for the user)
Compliment_plain: count of receiving "thank you" compliments (for the user)
Compliment_cool: count of receiving "you are cool" compliments (for the user)
Compliment_funny: count of receiving "you are funny" compliments (for the user)
Compliment_writer: count of receiving "good writer" compliments (for the user)
Compliment_photos: count of receiving "great photo" compliments (for the user)

2.2 Instructions

Suggested steps:

1. Load the first 100,000 lines of data.
2. Preprocessing the data.
3. Standardize the data.
4. Apply Principal Component Analysis to reduce the dimension of data.
5. Apply clustering on the data.
6. Explain your clusters and findings.

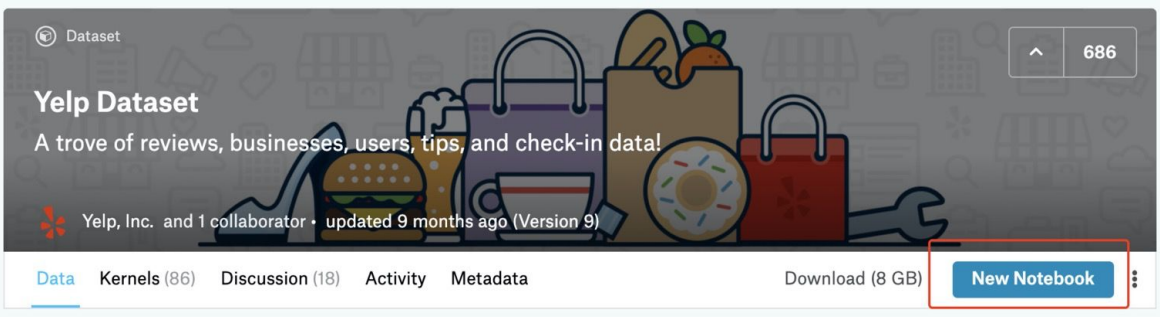
The first 3 steps are easy to understand. For step 4, you should group the attributes according to their covariance matrix as well as your understanding towards the data, and apply PCA to each group. For example, you may find Useful, Funny and cool are highly correlated, so you might want to make them as a group, and use the first principal component of those three attributes as a factor for clustering. You can decide the name of this factor, for instance, review_feedback_factor, that helps you to explain your clusters in the later steps. For step 5, you use the factors you developed in step 4 for clustering. You are welcome to use more than one clustering methods and compare the results. For step 6, you need to name and explain your clusters. For example, you find a cluster's centroid has very high review_feedback_factor and low review_stars_factor, you can name this cluster of users as strict gourmet, and explains: the strict gourmet tend to give strict reviews for restaurants, but are highly helpful to customers.

While doing this assignment, just imagine you are a data analyst in the marketing department of Yelp. Any valuable insights on users may greatly help your company. Do not limit yourself on the above instructions, you are encouraged to use more methods to enhance your findings and theories. The most important thing to keep in mind is to have fun with the data.

You'll need to submit a jupyter notebook (.ipynb) file for this part of the assignment. Conclusion and findings can be written in the markdown cells. Structure your code and markdown reasonably among cells and make sure every cell is runnable.

A sample structured notebook(inf552_hw4_yelp_handout.ipynb) is given to you. You can develop your code based on the given structure.

You can use your local jupyter notebook server or any other servers to finish this assignment, but it is best if you use the **Kaggle Kernel** so that you don't need to download the datasets (they are really huge) and it also reduces the difficulty for grading. To use the Kaggle kernel, simply click **New Notebook** on the Yelp dataset page.



2.3 Suggested Libraries

There are some libraries that you may need to use:

* `numpy.cov` or `pandas.cov` for covariance matrix.
    - https://docs.scipy.org/doc/numpy/reference/generated/numpy.cov.html
    - https://pandas.pydata.org/pandas-docs/stable/reference/api/pandas.DataFrame.cov.html
* `sklearn.decomposition.pca` for PCA.
    - https://scikit-learn.org/stable/modules/generated/sklearn.decomposition.PCA.html
* `sklearn.preprocessing.StandardScaler` for standardization.
    - https://scikit-learn.org/stable/modules/generated/sklearn.preprocessing.StandardScaler.html
* `sklearn.cluster` for clustering.
    - https://scikit-learn.org/stable/modules/clustering.html

The above libraries are only for reference. It is not mandatory to use those libraries in this part of assignment.

2.4 Submission

Include your `Firstname_Lastname_Yelp.ipynb` in the `Firstname_Lastname_HW4.zip` and submit to blackboard.

2.5 Rubric

50 points in total

- Program correctness(20 points): Demonstrated correct use of PCA and at least one clustering method.
- Documentation(10 points): Code is well commented and well structured.
- Report(20 points): The clusters are reasonable and well explained. The findings are insightful.