Student name: Thien Quoc Nguyen

Title of the Report/Paper: Data clustering with K-Means Algorithm

Applying program: Information Technology - Master

Date: 21-08-2022

Preface

With limited time, limited knowledge and many other problems, mistakes are inevitable. Since the field of machine learning and artificial intelligence is extremely large and requires a lot of research and development time, this project will summarize the most basic parts to introduce the algorithm as well as the initial use to data analysis. Please consider and give suggestions so that I can improve further with future projects.

Table of Contents

I. Introduction to Machine Learning

1. Definition

Artificial Intelligence (AI) is creeping into every area of life that we may not realize in recent years:[1] Self-driving cars from Google and Tesla, Facebook's self-tagging system in photos, Apple's Siri virtual assistant, Amazon's product recommendation system, Netflix's movie recommendation system, Google's AlphaGo Go player DeepMind,… Those are just a few of the many applications of AI/Machine Learning.

In 1959, Arthur Samuel described it this way: "the field of study that gives computers the ability to learn without being explicitly programmed"[2]. That's an old definition. Then in 1997, Tom Mitchell, a famous professor at Carnegie Mellon University - CMU defined more modern and standard as follows: ”A computer program is said to **learn** from experience *E* with respect to some class of **tasks** *T* and **performance** measure *P* if its performance at tasks in *T*, as measured by *P*, improves with experience *E*”[3]. An example: playing chess:

E: experience in playing many chess games

T: task to play chess

P: performance calculating from T if it improves with E

Simply put, Machine Learning is a subfield of Computer Science that has the ability to learn on its own based on input without having to be specifically programmed. In recent years, when the computing power of computers has been raised to a new level with huge amounts of data collected by big technology firms, Machine Learning has come a long way and a new field was born: Deep Learning. Deep Learning has helped humans do things that were impossible a decade ago: classify thousands of different objects in photos, create labels for images, imitate human voices and handwriting, or even compose music.

Any Machine Learning problem can be assigned to two broad categories: *Supervised learning* and *Unsupervised learning*.

2. Supervised learning

[4]Supervised learning is an algorithm that predicts the output (*outcome*) of new data (*new input*) based on the relationship between previously known pairs (*input, outcome*). This data pair is also known as (*data, label*). Supervised learning is the most popular group of Machine Learning

algorithms. Supervised learning problems are divided into two categories: *Regression* and *Classification*.

In *regression*, our task is to try to predict the output continuously, i.e. try to map the inputs to a continuous function.

In *classification*, our task is to try to predict distinct outputs, that is, to try to map input to distinct categories.

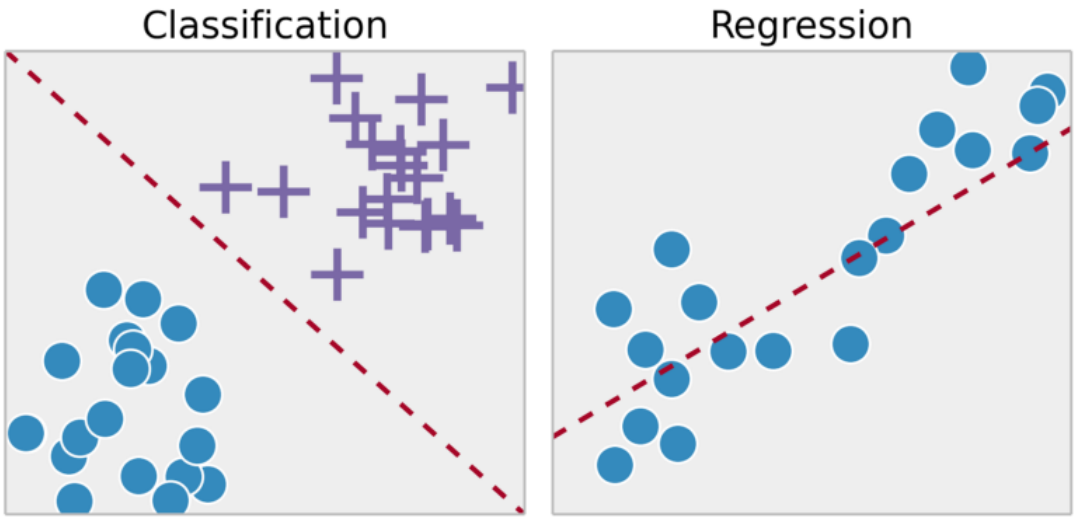

Figure 1. Regression vs Classification. Source: https://towardsdatascience.com/supervised-vs-unsupervised-learning-14f68e32ea8d

Example: Regression – Take a picture of a person, and predict the age based on the basic elements of the picture.

Classification – Given a patient with a tumor, predict whether the tumor is benign or malignant.

3. Unsupervised learning

For a practical problem: grouping customers based on purchasing behavior. In this problem, we don't know the *outcome* or the *label*, only the input data. This algorithm will rely on the structure of the data to perform a certain task, for example, the above problem, clustering or reducing the number of dimensions of the data to facilitate storage and computation. Mathematically, *unsupervised learning* is when we only have input ***X*** without knowing the corresponding label ***Y***. With unsupervised learning there is no response based on the prediction results because there is no true answer.

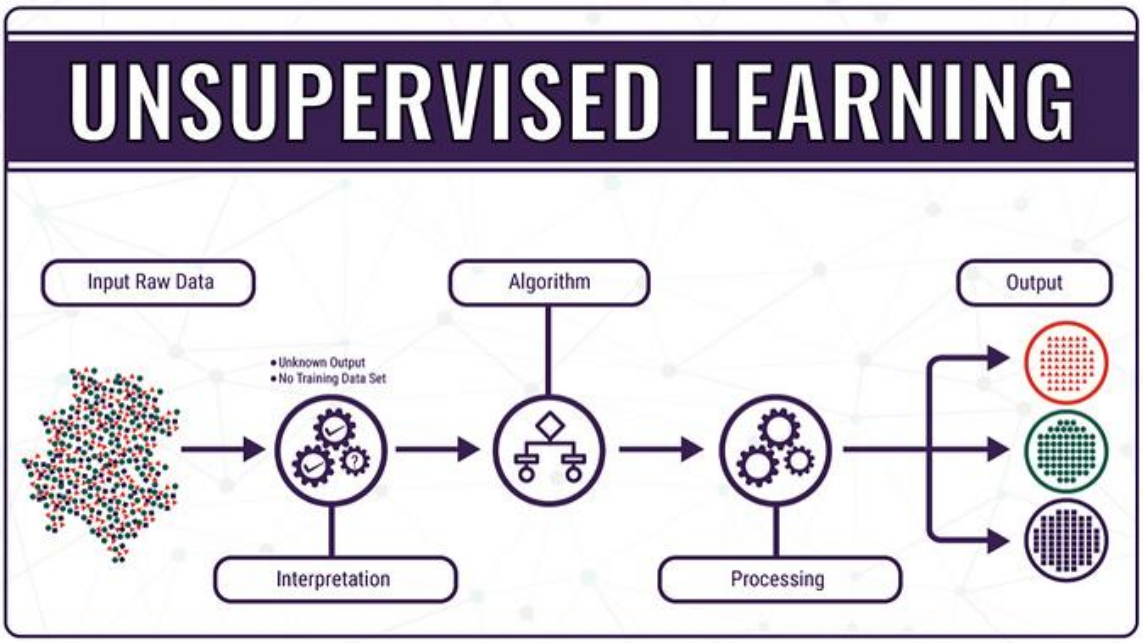

Figure 2. Unsupervised Learning. Source: https://community.singularitynet.io/t/how-singularitynet-is-advancing-unsupervised-language-learning/524

Example: Take a collection of 1,000,000 different genes and find a way to automatically group these genes into groups that are similar, or related to each other by different variables – such as lifespan, location, role,...

[4]Another type of problem also belongs to Unsupervised learning, which is Association. In this problem, we try to infer a rule based on any given data. For example, male customers who buy clothes tend to buy with watches or belts; Avengers: Endgame moviegoers tend to see more Iron Man movies, based on that to create a customer recommendation system (Recommendation System), promoting customer demand.

4. Commonly used programming libraries

[5]Pandas: an open source library, effectively supporting data manipulation. It is also a powerful data processing and analysis toolkit of the python programming language. This library is widely used in both research and development of data science applications as it uses a separate data structure, Dataframe. Pandas provides a lot of functionality for handling and working on this data structure.

[6]Numpy: provides objects and methods for working with multi-dimensional arrays and linear algebra operations.

[7]Scikit-learn (abbreviated as sklearn): an open source library for machine learning - a branch of artificial intelligence, very powerful and popular with the Python community, designed on top of

NumPy and SciPy. Scikit-learn contains most of the most modern machine learning algorithms, accompanied by documentations, always up to date.

[8]Matplotlib. pyplot: is a set of functions that matplotlib can work with as MATLAB. Each pyplot function makes a change in the graph, for example creating a plot frame, plotting on the created frame, or plotting multiple lines on the same graph, labeling the axes, etc. The various states in matplotlib.pyplot store the called functions, which save everything of the current graph and plotted plot area, and the graph function plotted directly on the coordinate axes.

II. Project decription

1. Problem definition

The trend of today's world, data is becoming more and more expansive. Humans do not have enough human resources to analyze and categorize this massive "data block". There are many ways to extract information from data, one of which is *clustering*. Clustering is often used to analyze data that has large or even very large data in terms of *numbers* and *labels* on unknown data. Since labeling large amounts of data is a costly and time-consuming task, we need to find a different approach, which is necessary to extract useful information from the data. Clustering focuses on finding methods for efficient and effective cluster analysis in large databases.[9]

In clustering, it is often necessary to define several *centers* of the cluster in advance to ensure that the *sum of squared errors* between each data point and its potential centers is small during clustering. In this project, a data clustering software using K-means clustering algorithm is one of the solutions being applied in many fields around the world.

2. Introduction to K-means Algorithm

In the implementation of the data clustering algorithm, MacQueen introduced the K-means algorithm in his paper in 1967[10]: K-means clustering is a form of Unsupervised Learning, used when you have unlabeled data (i.e., data with no categories or undefined groups). The purpose of this algorithm is to find groups in the data, with the number of groups represented by the variable ***K***. The algorithm works iteratively to assign each data point to one of the ***K*** groups based on the features provided. The data points are clustered based on feature similarity. The result of the algorithm means:

i. The centers of the clusters K can be used to label new data.

ii. Label for training data (each data point is assigned to a unique cluster).

Brief description of the algorithm steps [11]:

Input: Unlabeled data ***X*** and the number of clusters ***K*** to find.

Output: ***M*** centers and label vectors for each data point ***Y***.

Step 1: Choose any ***K*** points as initial centers. The center can be initialized randomly or choose any point on the data to be the initial center.

$$C^{(0)} = \left\{m_1^{(0)}, m_2^{(0)}, \dots, m_k^{(0)}\right\}$$

Step 2: Assign each data point to the cluster whose center is closest to it. For each data point, we calculate its distance to the centers (by Euclidean Distance). We will assign them to the nearest center. The set of points assigned to the same center will form a cluster.

$$S_i^{(t)} = \left\{ x_p : \left\| x_p - m_i^{(t)} \right\|^2 \leq \left\| x_p - m_j^{(t)} \right\|^2 \right\}, \forall_j, 1 \leq j \leq k$$

Step 3: If the assignment of data to each cluster in step 2 does not change compared to the previous loop, then we stop the algorithm.

Step 4: Update the center for each cluster by calculating the mean of all data points assigned to that cluster after step 2.

$$m_i^{(t+1)} = \frac{1}{\left| S_i^{(t)} \right|} \sum_{x \in S_i^{(t)}} x_j$$

Step 5: Go back to step 2.

3. Evaluation methods and Applications

The purpose of clustering is to find out the inner nature of groups of data. It generates clusters. However, there is no single best criterion for evaluating the effectiveness of clustering analysis, this depends on the purpose of clustering such as: data reduction, “natural clusters”, “useful” clusters, outlier detection…[11]

Clustering techniques can be applied in many fields such as[11]:

- Marketing: Identify customer groups (potential customers, value customers, classify and predict customer behavior, etc.) that use the company's products or services to help the company build an effective business strategy.
- Biology: Grouping animals and plants based on their properties.
- Library: Track readers, books, predict readers' needs...
- Insurance, finance: Grouping users of insurance and financial services, predicting trends of customers, detecting financial frauds.
- WWW: Categorize documents, classify web users…
- Education: Classify students, evaluate students' academic performance based on scores to orient the exam block for students...

There are many ways to evaluate the quality of clustering, depending on the different problems that we use different methods. The commonly used methods are[11]: accuracy score, confusion matrix,

ROC curve, Area Under the Curve, Precision and Recall, F1 score, Top R error, etc. This project will use *F1 score* for evaluation.

F1 score is a harmonic mean of *Precision* and *Recall*[12]. With a way of identifying a class as positive, *Precision* is defined as the ratio of true positive scores among those classified as positive (TP + FP). *Recall* is defined as the ratio of the number of true positives among those that are actually positive (TP + FN).

Prediction \ Reality	Positive	Negative
Positive	TP	FP
Negative	FN	TN

Mathematically, *Precision* and *Recall* are two fractions with the same numerator but different denominators:

$$precision = \frac{TP}{TP + FP} \quad recall = \frac{TP}{TP + FN}$$

High *precision* means that the accuracy of the points found is high. High *Recall* means high True Positive Rate, which means that the rate of missing really positive points is low.

When *Precision* = 1, all the points found are really *positive*, that is, there are no *negative* points mixed into the results. However, *Precision* = 1 does not guarantee that the model is good, as the question is whether the model has found all the *positives*. If a model finds only one *positive* point that it is most certain about, then we cannot call it a good model.

When *Recall* = 1, every *positive* is found. However, this quantity does not measure how many *negatives* are mixed in. If the model classifies every point as *positive*, then surely *Recall* = 1, but it is easy to see that this is an extremely bad model.

A good classification model is one that has both high *Precision* and *Recall*, i.e. as close to 1 as possible. F1 score has a value in the half range $(0,1]$. The higher the F1, the better the classifier. When both *precision* and *recall* are equal to 1 (best possible), $F1 = 1$. When both *precision* and *recall* are low, for example 0.1, $F1 = 0.1$

$$F1\ score = 2.\frac{precision.recall}{precision + recall}$$

Here are a few examples of F1:

precision	recall	F1
1	1	1
0.1	0.1	0.1
0.5	0.5	0.5
1	0.1	0.182
0.3	0.8	0.36

Thus, a classifier with *precision* = *recall* = 0.5 is better than another classifier with *precision* = 0.3, *recall* = 0.8 in this measure.

III. Installation and Testing

1. Clustering simulation data

With 2-dimensional point data (*x, y*), which are separate clusters of points, when running with the K-means algorithm, we can see 100% accurate results through determining F1 score = 1. In addition, the computational time of the algorithm for these separate data clusters is very fast because the centroids of the clusters do not move much.

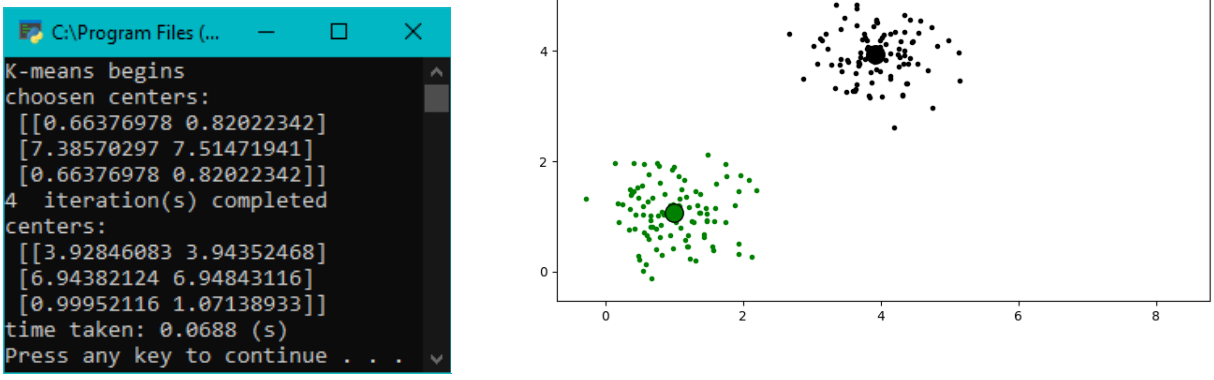

Figure 3. Clustering with separate clusters of points

Not only does the algorithm work with 2- and 3-dimensional data, but the algorithm can also run for multi-dimensional data with the same accuracy and speed as the above results, but graphing is impossible so we will go on. For overlapping data, the algorithm will not run efficiently. The efficiency

of the algorithm will be inversely proportional to the overlap of data clusters. The more overlap, the lower the efficiency.

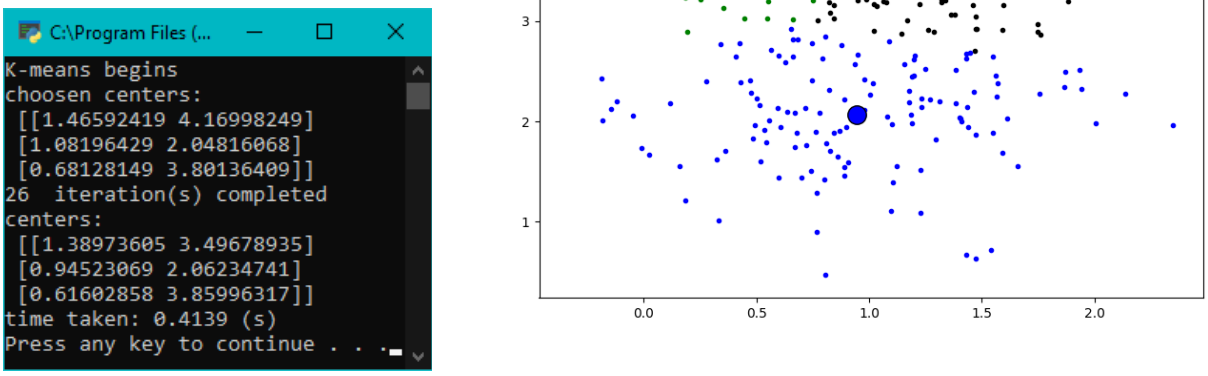

Figure 4. Clustering with overlapped clusters of points

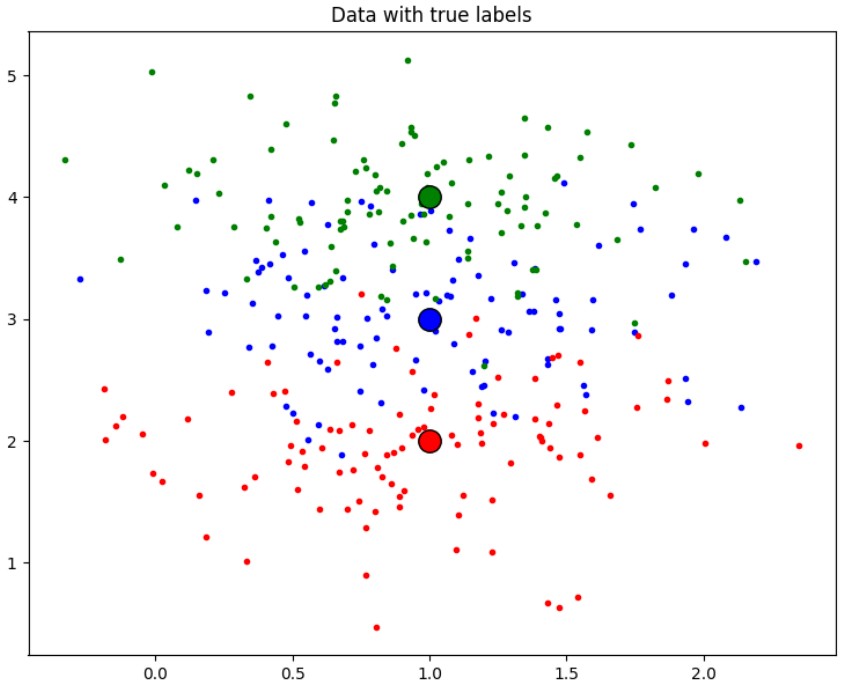

Figure 5. Overlapped data with true labels

2. Customer clustering using K-means algorithm

Dataset is collected on kaggle[13]: Suppose you own a store, through a customer's membership card, you have some customer information like: Customer ID, Age, Gender, monthly income and spending score at the shop. Classify customers based on those characteristics to be able to understand customer groups or plan a reasonable strategy based on that customer group.

Dataset includes 200 customers. Since this data has not been labeled, it is not possible to compare the performance of the algorithm. This report will be based on the label that the algorithm learns and compare with the label that sklearn's K-means algorithm to compare the output.

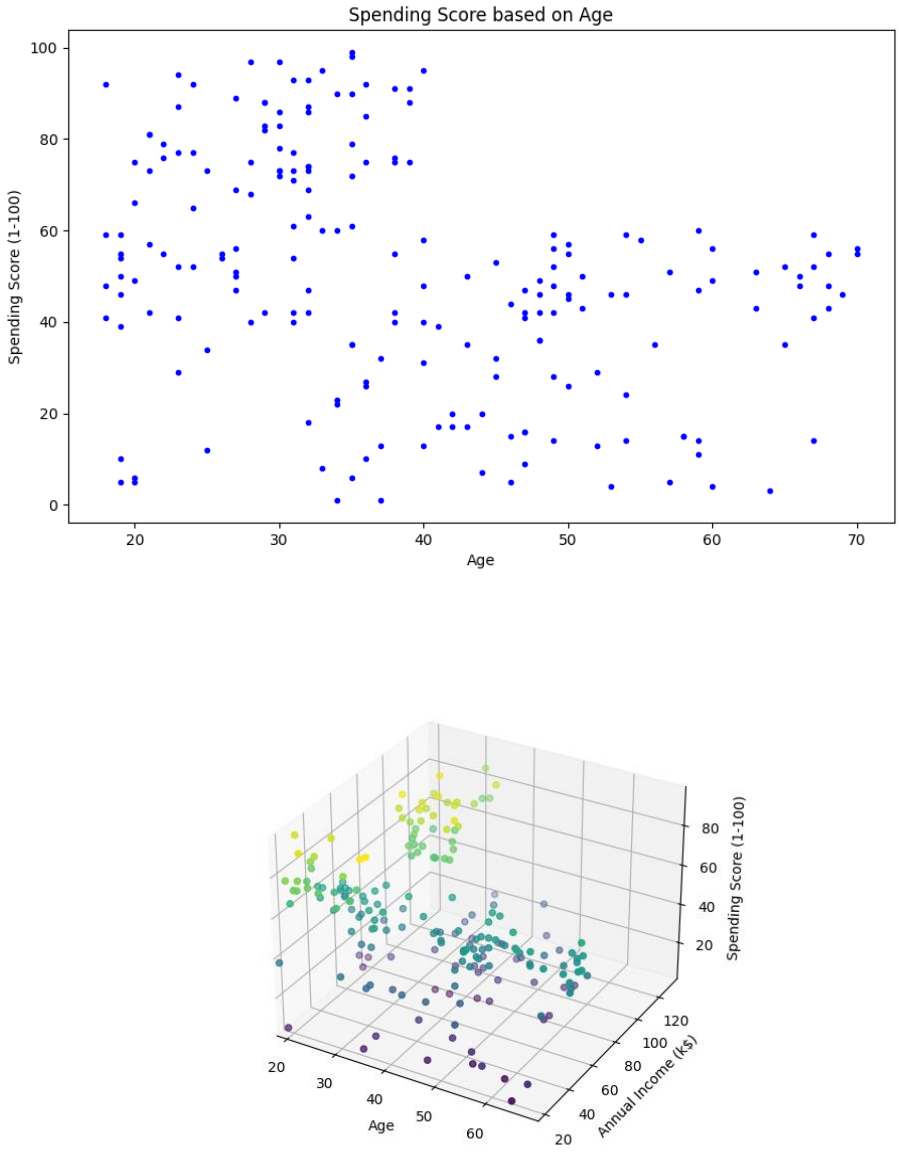

Figure 6. Dataset illustrated through 2 columns of Age and Spending Score

Data needs to be preprocessed: remove the header and the customerID column because these fields can affect the clustering ability of the algorithm, causing undesirable results. Also, if there is any feature with distinct "unit", we should remove to reduce the dispersion of data. For this run, the ***K*** will be set at 3 clusters.

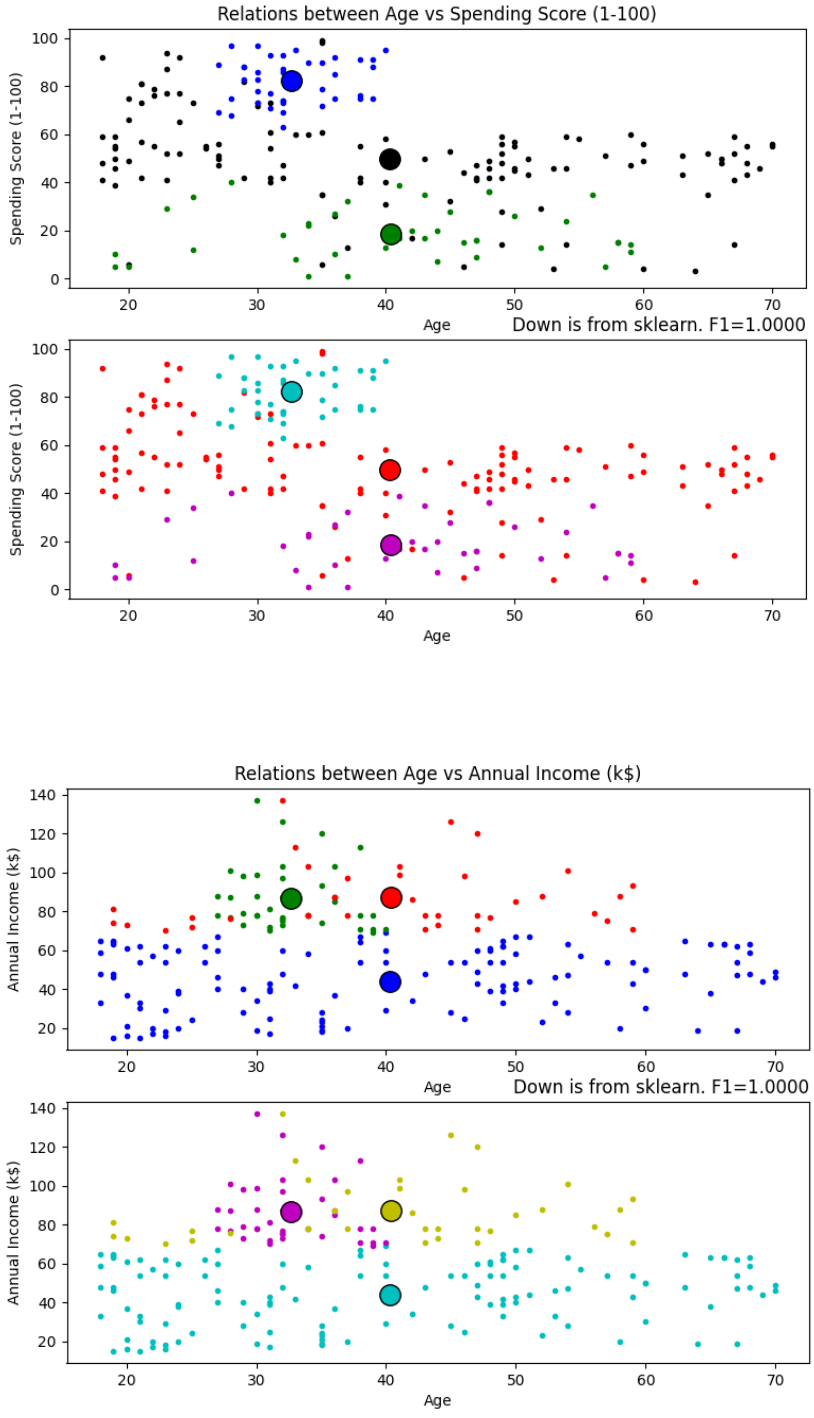

Figure 7.Clustering with real problem data

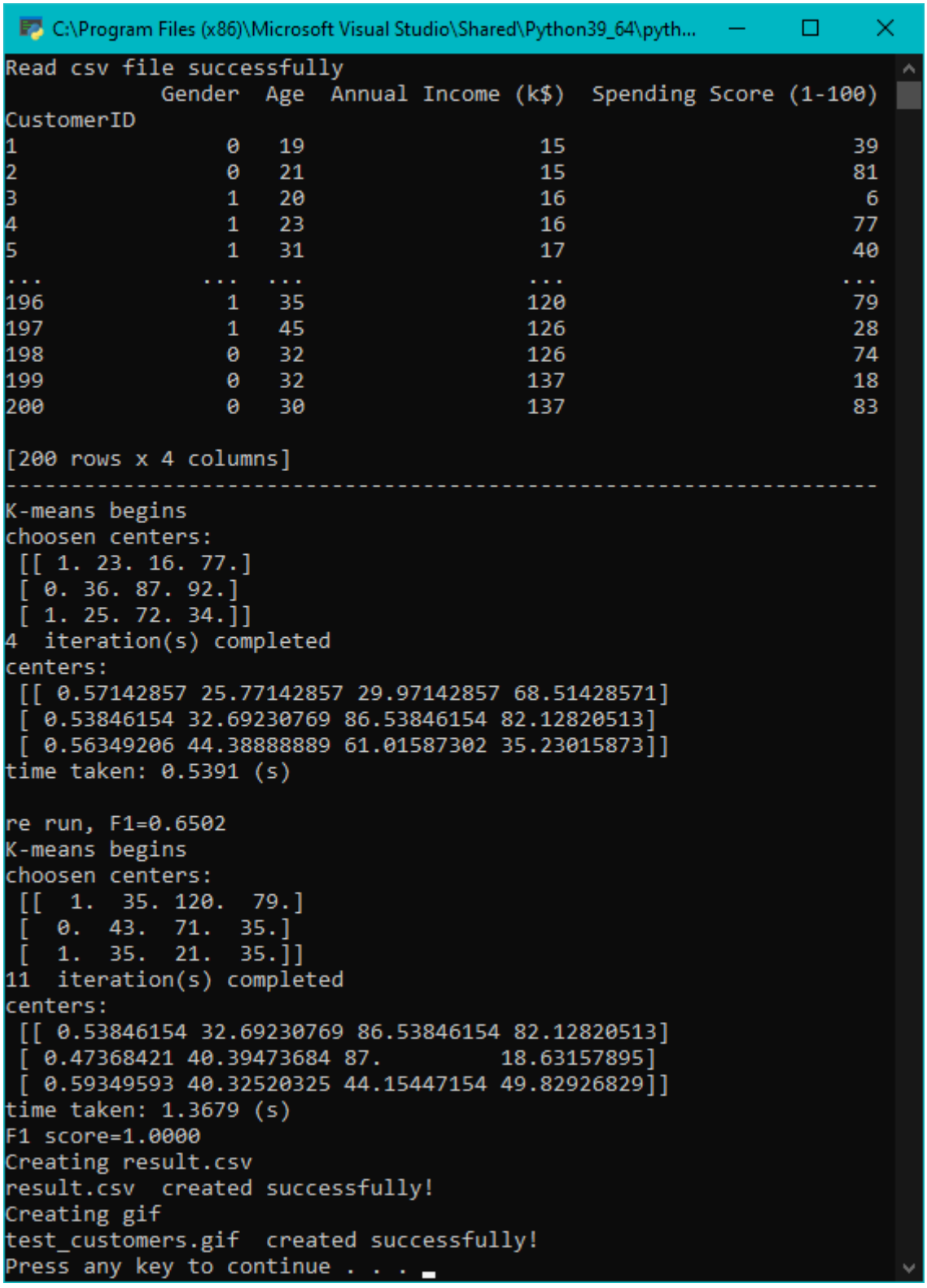

Figure 8. Program output with F1-score = 1

We can see that although the data is somewhat overlapping, the F1 score between the labels of the learned algorithm and the labels from sklearn's K-means algorithm is equal to 1. So the algorithm runs as expected.

However, the performance of the clustering algorithm is highly dependent on the initialization of the cluster center at the beginning of the algorithm. [15]One simple solution is just to run K-Means a couple of times with random initial assignments. We can then select the best result by taking the one with the minimal *sum of distances* from each point to its cluster – the *error* value that we are trying to minimize in the first place.

The algorithm is initialized with ***k*** = 3. But, how can we know if the selected ***k*** is good or not? If we increase ***k***, the elements in the cluster will be smaller, therefore, the *error* value will be smaller. So, just bigger is better? Although we can set ***k*** to be equal to the total number of elements in the dataset, the total *error* will be 0, but that doesn't seem to solve the problem because each element in the dataset will become its own center. This is called *overfitting*.

One way to solve the main problem is to include some *penalties* for the larger number of clusters. So, we're not just trying to minimize the *error* value, but also the *error* + *penalty*. The *error* will gradually converge to 0 as we increase the number of clusters, but so the *penalty* will increase. Balancing these two quantities will give us the optimal result. This solution is called *X-means*, a variant of K-means[14]

IV. Conclusion

The advantage of this project is that it is easy to install, and the accuracy is quite high for separate data. However, for overlapping data, the accuracy will not be high. This is a drawback of the K-means algorithm.

In addition, the algorithm also requires specifying the number of clusters and must initialize the center of each cluster reasonably in order to have a optimal result.

References

[1] Tiep Vu. (2016, December 26). Bài 1: Giới thiệu về Machine Learning/Lesson 1: Introduction to Machine Learning. Retrieved from https://machinelearningcoban.com/2016/12/26/introduce/

[2] Samuel, Arthur (1959). "Some Studies in Machine Learning Using the Game of Checkers". IBM Journal of Research and Development. 3 (3): 210–229.

[3] Mitchell, T. (1997). Machine Learning, McGraw Hill. ISBN 0-07-042807-7, p.2.

[4] Tiep Vu. (2016, December 27). Bài 2: Phân nhóm các thuật toán Machine Learning/Grouping of Machine Learning Algorithms. Retrieved from https://machinelearningcoban.com/2016/12/27/categories/

[5] Nguyễn Văn Hiếu (2018 October 2) Hướng dẫn sử dụng thư viện Pandas trong Python/ Instructions for using the Pandas library in Python. Retrieved from https://viblo.asia/p/huong-dan-su-dung-thu-vien-pandas-trong-python-XL6lAxaDZek

[6] Máy học cho người Việt/Machine learning for Vietnamese (2017 June 1) Numpy. Retrieved from https://ml4vn.blogspot.com/2017/06/numpy.html

[7] Máy học cho người Việt/Machine learning for Vietnamese (2017 June 1) Bắt đầu học Scikit-learn. Retrieved from https://ml4vn.blogspot.com/2017/08/bat-au-voi-scikit-learn-cac-khai-niem.html

[8] Nguyễn Văn Hải. Vẽ đồ thị sử dụng Mathplotlib/Plotting using Matplotlib. Retrieved from http://viet.jnlp.org/home/nguyen-van-hai/nghien-cuu/mlearning/building-machine-learning-system-using-python/chng-1-bt-u-vi-python/ve-do-thi-su-dung-mathplotlib

[9] Han, J., Kamber, M., & Pei, J. (2011). Data mining concepts and techniques third edition. Morgan Kaufmann.

[10] MacQueen, J. (1967, June). Some methods for classification and analysis of multivariate observations. In Proceedings of the fifth Berkeley symposium on mathematical statistics and probability (Vol. 1, No. 14, pp. 281-297).

[11] Tiep Vu. (2017, January 1). Bài 4: K-means Clustering. Retrieved from https://machinelearningcoban.com/2017/01/01/kmeans/

[12] ClustEval | F1-Score. (n.d.). Retrieved from https://clusteval.sdu.dk/1/clustering_quality_measures/18

[13] Mall Customer Segmentation Data. (n.d.). Retrieved from https://www.kaggle.com/vjchoudhary7/customer-segmentation-tutorial-in-python

[14] Pelleg, D., & Moore, A. (2000). X-means: Extending K-means with Efficient Estimation of the Number of Clusters. Carnegie Mellon University, Pittsburgh, PA.

[15] Iliassich, L. (2016, May 19). Clustering algorithms: From start to state of the art. Toptal Engineering Blog. https://www.toptal.com/machine-learning/clustering-algorithms