Machine learning Assignment

Clustering and RLearning implementations

Team:

Jagathsri B-22IT032

Aananthi N-22IT001

—

Clustering Algorithms

Clustering algorithms are a type of unsupervised machine learning that groups data points into clusters based on similarity. The goal is to organize data in a way that data points in the same cluster are more similar to each other than to those in other clusters. Clustering is widely used in customer segmentation, image recognition, and data analysis.

K-means Clustering

K-means is a popular centroid-based clustering algorithm that aims to partition data into KKK clusters. It operates by iteratively assigning data points to the nearest cluster center (centroid) and then recalculating the centroids based on the new clusters. The process repeats until the centroids stabilize or a maximum number of iterations is reached.

- **Advantages**: Fast and efficient, especially for large datasets.
- **Limitations**: Sensitive to outliers and can form clusters with non-spherical shapes poorly. Requires the number of clusters KKK to be specified in advance.

K-medoids Clustering

K-medoids is similar to K-means but is a **medoid-based algorithm**. Instead of centroids, it chooses actual data points as cluster centers (medoids), which are representative objects within a cluster. K-medoids are more robust to outliers because it minimizes the sum of dissimilarities between data points and the medoid, instead of using Euclidean distance as in K-means.

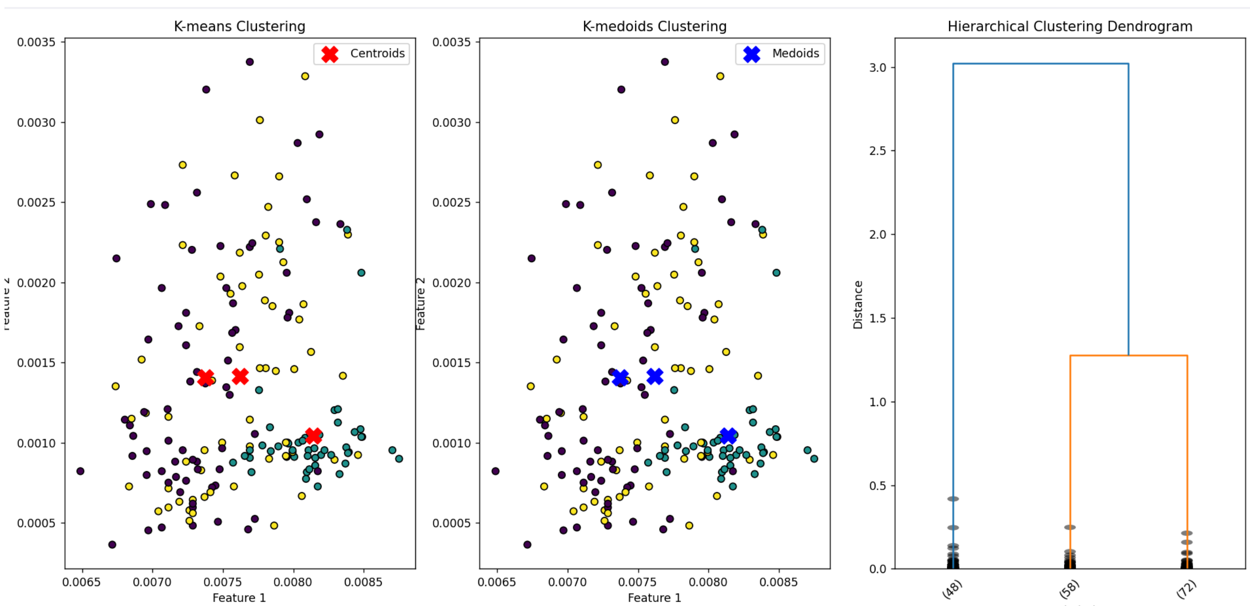
- **Advantages**: More robust to outliers and works better for non-spherical clusters.

- **Limitations**: Generally slower than K-means, especially with large datasets, due to more complex distance calculations.

Agglomerative Clustering

Agglomerative Clustering is a hierarchical clustering method that starts with each data point as an individual cluster and iteratively merges the closest clusters based on a distance metric until a specified number of clusters is reached or all points are combined into a single cluster. It can utilize various distance metrics such as Euclidean or Manhattan distance and different linkage criteria like single, complete, or average linkage to determine how clusters are merged. This technique is useful for exploring the structure of data and is often visualized using dendrograms, making it popular in fields like biology and marketing. However, it can be computationally intensive and sensitive to noise and outliers.

Output:



Reinforcement Learning Algorithms

Reinforcement Learning (RL) is a type of machine learning where an agent learns by interacting with an environment. It takes actions to maximize cumulative rewards over time. RL is unique in its approach to learning from the consequences of actions, making it suitable for applications like robotics, game playing, and recommendation systems. RL algorithms are usually based on a trial-and-error approach and can be model-free or model-based depending on whether they learn a model of the environment.

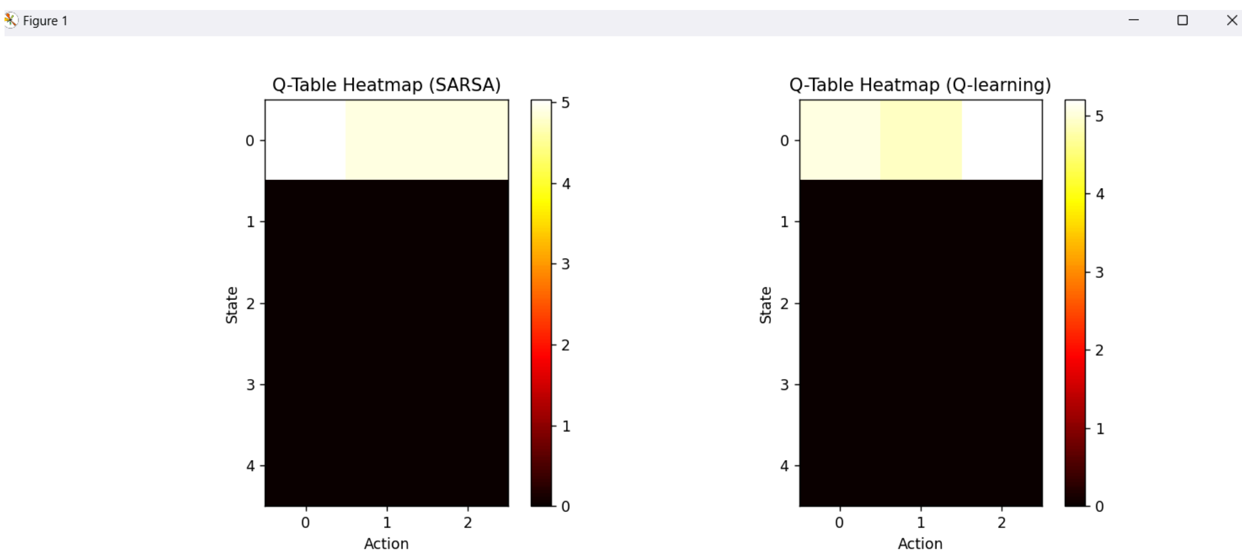
SARSA (State-Action-Reward-State-Action)

SARSA is an on-policy RL algorithm where the agent learns the value of state-action pairs by following a specific policy during training. The update rule for SARSA involves using the next action selected by the current policy to update the Q-value. The name "SARSA" comes from the sequence it follows: (State, Action, Reward, next State, next Action). SARSA's updates are more conservative as it considers the policy's actual behavior, making it well-suited for environments with dynamic risks or changing conditions.

Q-learning

Q-learning is an off-policy RL algorithm that seeks to find the optimal action-selection policy independently of the agent's actions. Unlike SARSA, Q-learning updates the Q-value by choosing the maximum Q-value of possible next actions, making it more aggressive in learning optimal policies. This algorithm works by maximizing future rewards and tends to converge faster on the best policy, especially in stable environments.

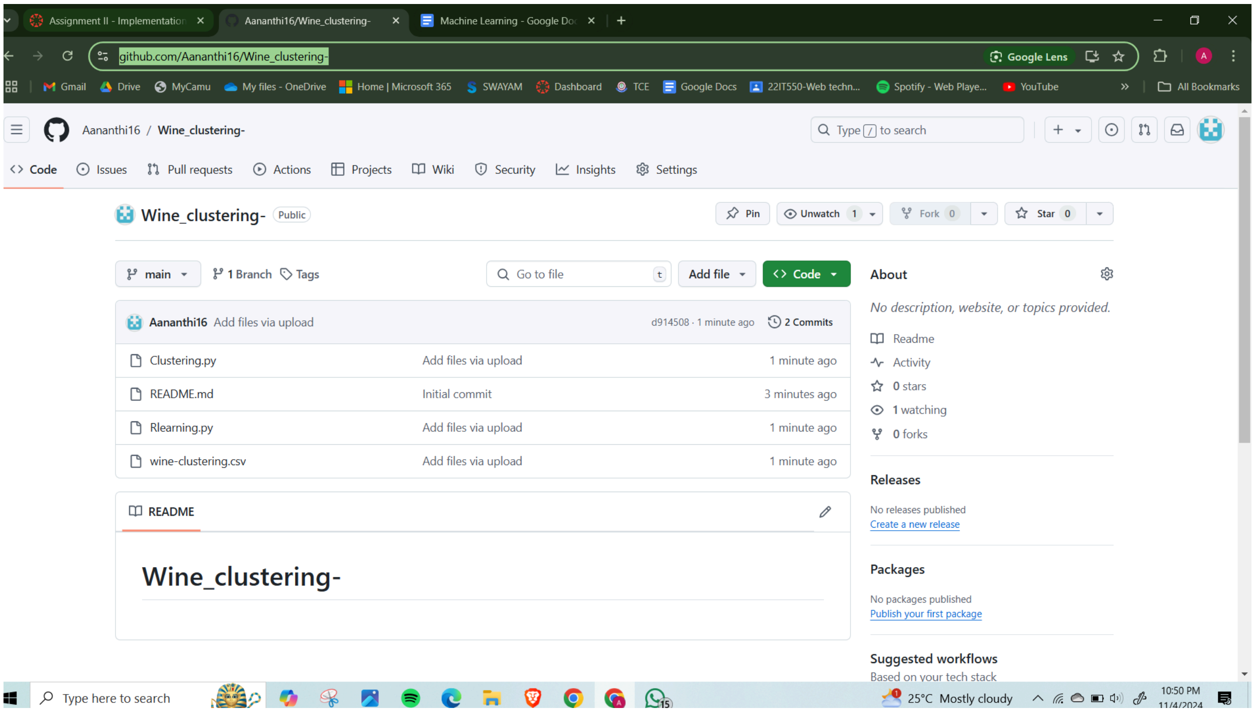
Output:



Git

Github Link:https://github.com/Aananthi16/Wine_clustering-

Screenshots:


Assignment II - Implementation
Aananthi16/Wine_clustering-
Machine Learning - Google Docs
github.com/Aananthi16/Wine_clustering-
Google Lens
Gmail
Drive
MyCamu
My files - OneDrive
Home | Microsoft 365
SWAYAM
Dashboard
TCE
Google Docs
22IT550-Web techn...
Spotify - Web Playe...
YouTube
All Bookmarks
Aananthi16 / Wine_clustering-
Type / to search
Code
Issues
Pull requests
Actions
Projects
Wiki
Security
Insights
Settings
Wine_clustering-
Public
Pin
Unwatch 1
Fork 0
Star 0
main
1 Branch
Tags
Go to file
Add file
Code
Aananthi16 Add files via upload
d914508 · 1 minute ago
2 Commits
Clustering.py
Add files via upload
1 minute ago
README.md
Initial commit
3 minutes ago
Rlearning.py
Add files via upload
1 minute ago
wine-clustering.csv
Add files via upload
1 minute ago
README
Wine_clustering-
About
No description, website, or topics provided.
Readme
Activity
0 stars
1 watching
0 forks
Releases
No releases published
Create a new release
Packages
No packages published
Publish your first package
Suggested workflows
Based on your tech stack
Type here to search
25°C Mostly cloudy
10:50 PM
11/4/2024