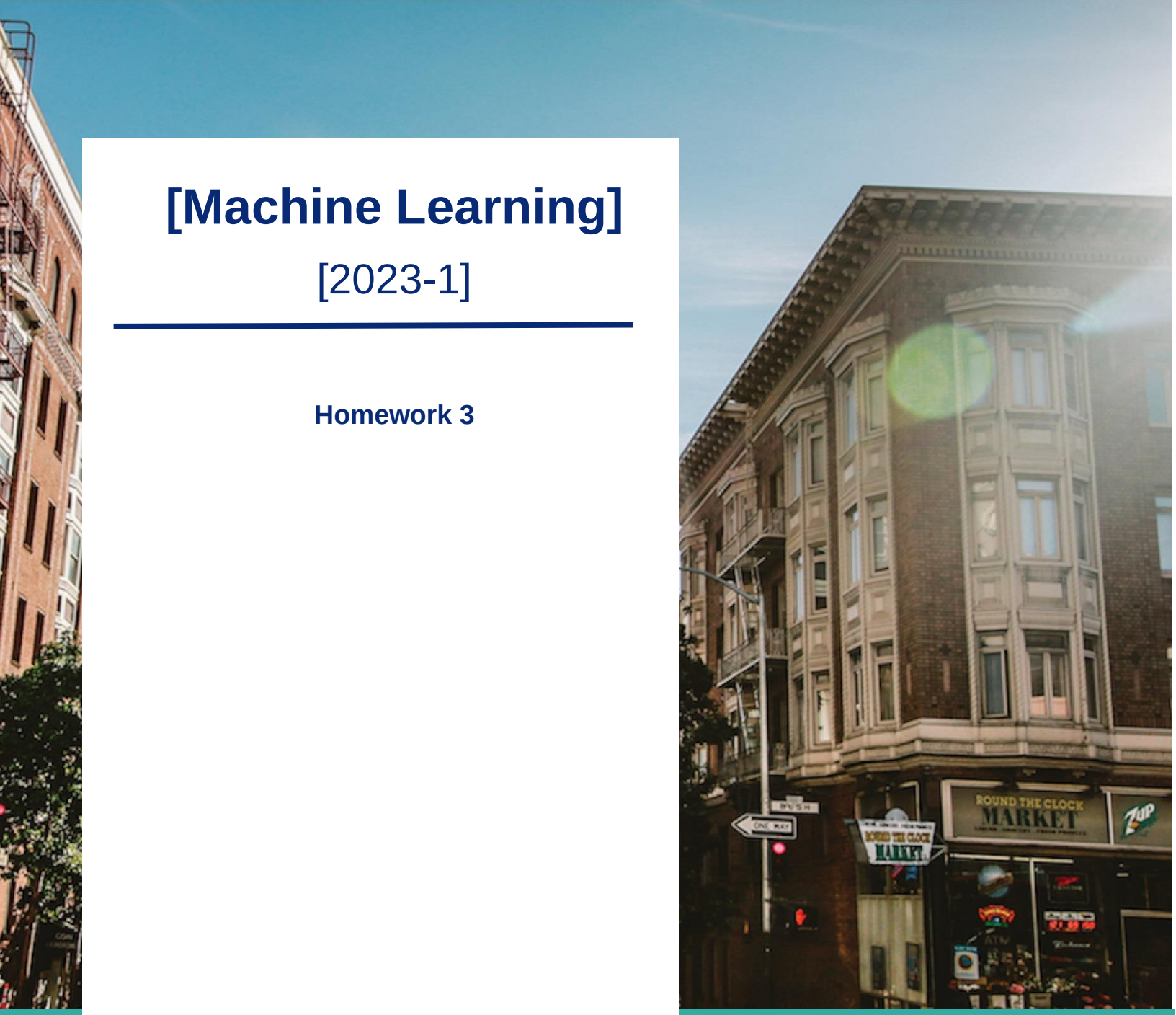


[Machine Learning]

[2023-1]

Homework 3

[Due Date] 2023.05.24

Student ID : 2018112007

Name : 이승현

Professor : Juntae Kim

1. Explain the differences between K-means and DBSCAN clustering algorithms, and discuss the advantages and disadvantages. (10 pts)

Your Answer
K-means 는 중심기반(Center-based) 클러스터링 방법으로 "유사한 데이터는 중심점(centroid)을 기반으로 분포할 것이다"는 가정을 기반으로 한다. n 개의 데이터와 k(<=n)개의 중심점(centroid)이 주어졌을때 각 그룹 내의 데이터와 중심점 간의 비용(거리)을 최소화하는 방향으로 계속 업데이트를 해줌으로써 그룹화를 수행하는 기법이다. k-means 의 장점 1. 구현이 간단하고, 속도가 빠르다 2. 대용량 데이터셋에도 적용가능하다. 3. Local minimum 으로 수렴한다. k-means 의 단점 1. 초기에 랜덤으로 설정하는 중심값(Initial point)에 따라 데이터셋 군집화 성능에 큰 영향을 미친다. 2. 노이즈와 아웃라이어의 데이터가 군집화 성능에 큰 영향을 미친다. 3. 원형의 cluster 만 찾을 수 있다. DBSCAN 는 밀도기반(Density-based) 클러스터링 방법으로 "유사한 데이터는 서로 근접하게 분포할 것이다"는 가정을 기반으로 한다. K-means 와 달리 처음에 그룹의 수(k)를 설정하지 않고 자동적으로 최적의 그룹 수를 찾아나간다. DBSCAN 의 장점 1. 클러스터 개수 정의 불필요 2. 데이터의 밀도를 계산해서 클러스터링

3. 비선형 경계 클러스터링 가능

4. 노이즈에 강함

5. 클러스터 경계에 있는 애매한 점이 감소함

DBSCAN 의 단점

1. 데이터를 사용하는 순서에 따라 클러스터링 차이
2. 고차원 데이터에서 적절한 반경 e 를 찾기 어려움

2. Explain what are the 'one-hot representation' and 'vector representation' of a word. Select one of the existing word embedding algorithms and explain how it works to obtain an appropriate vector representation. (10 pts)

Your Answer

One-hot representation 은 단어를 벡터로 표현하는 방법 중 하나이다. 이 방법은 단어를 고유한 인덱스에 해당하는 위치에 1 의 값을 갖고, 나머지는 0 의 값을 갖는 벡터로 표현한다. 각 단어는 전체 단어 사전 크기의 길이를 가진 벡터로 표현되며, 해당하는 단어의 인덱스 위치에만 1 의 값을 갖는다. 이 표현 방식은 단어 간의 관계나 의미를 고려하지 않고 각 단어를 독립적인 개체로 취급한다.

Vector representation 은 단어를 고정된 크기의 실수 벡터로 표현하는 방법이다. 이 방법은 단어를 고차원 공간 상의 점으로 나타내어 단어의 의미와 관련된 정보를 담을 수 있게 한다. 벡터 표현은 단어 간의 의미적 유사성을 계산하거나 단어를 특정 작업에 활용할 수 있는 잠재적 의미를 포착하는 데 유용하다.

단어 벡터 표현을 위한 알고리즘 중 하나는 'Word2Vec'이다. Word2Vec 은 신경망 기반의 언어 모델로, 대규모의 텍스트 데이터로 사전 학습된 임베딩을 생성한다. Word2Vec 은 주변 단어들의 패턴을 학습하여 단어를 밀집된 벡터로 표현한다.

Word2Vec 은 'Skip-gram'과 'CBOW(Continuous Bag of Words)' 두 가지 모델을 제공한다. Skip-gram 모델은 주어진 단어로부터 주변에 있는 단어를 예측하는 방식으로 학습된다. 반면에 CBOW 모델은 주어진 주변 단어들을 통해 해당하는 단어를 예측하는 방식으로 학습된다.

Word2Vec 모델은 단어를 벡터로 변환하기 위해 다층 신경망을 사용하며, 주어진 문맥 정보를 활용하여 단어의 의미를 파악한다. 모델은 단어 간의 유사성을 포착하기 위해 벡터 공간에서 단어 간의 거리를 계산하거나, 벡터 간의 유사성을 측정하는 코사인 유사도 등을 활용할 수 있다.

Word2Vec 은 대량의 텍스트 데이터로 사전 학습된 임베딩을 제공하며, 이를 특정 작업에 활용하거나 필요한 작업에 맞게 추가적인 학습을 통해 세부 조정할 수 있다. 이를 통해 단어의 의미를 잘 반영하고 유용한 표현을 얻을 수 있다.

3. Explain what 'Vanishing Gradient Problem' is. Compare the sigmoid function and ReLU function in the context of the vanishing gradient and computation efficiency. (10 pts)

Your Answer
Vanishing Gradient Problem 은 심층 신경망에서 발생하는 문제이다. 이 문제는 Backpropagation 알고리즘에서 gradients 가 극도로 작아져서 사실상 사라지는 현상을 말한다. 신경망은 Backpropagation 을 사용하여 가중치를 조정하고 학습을 진행하는데, 기울기가 사라지면 가중치 업데이트가 거의 이루어지지 않아 학습이 제대로 이루어지지 않는 문제가 발생한다. 시그모이드 함수와 ReLU 함수를 비교하면 다음과 같은 차이점이 있다. 시그모이드 함수는 다음과 같이 정의된다: f(x) = 1 / (1 + exp(-x)) ReLU 함수는 다음과 같이 정의된다: f(x) = max(0, x) 1. Vanishing Gradient Problem 시그모이드 함수는 입력 값이 크거나 작을 때 기울기가 매우 작아지는 특징을 가지고 있다. 이는 Backpropagation 과정에서 이전 레이어로 전파되는 기울기가 매우 작아지거나 사라지는 원인이 된다. 반면에 ReLU 함수는 입력 값이 양수인 경우 기울기가 1 이므로 사라지는 기울기 문제가 발생하지 않는다. 2. 계산 효율성: 시그모이드 함수는 지수 함수(exp)를 사용하여 계산되기 때문에 계산 비용이 크다. 반면에 ReLU 함수는 입력이 양수인 경우에는 단순히 입력 값을 반환하므로 계산이 더 효율적이다. 따라서, 사라지는 기울기 문제와 계산 효율성 측면에서 시그모이드 함수와 ReLU 함수는 다른 특징을 가지고 있다.

4. Explain the difference between Sigmoid and Softmax function. For following x, compute both Sigmoid(**x**) and Softmax(**x**) (10 pts)

x = [-0.5, 1.2, -0.1, 2.4]

Your Answer

Sigmoid 함수와 Softmax 함수는 모두 비선형 활성화 함수로 사용되는 함수이다. 하지만 두 함수는 사용되는 문맥과 동작 방식에서 차이가 있다.

Sigmoid 함수는 입력 값을 0 과 1 사이의 확률 값으로 변환하는 함수이다. Sigmoid 함수의 수식은 다음과 같다: f(x) = 1 / (1 + exp(-x)). Sigmoid 함수는 각 입력 값에 대해 독립적으로 적용되며, 출력 값은 해당 입력의 확률로 해석할 수 있다. 이 함수는 이진 분류 문제에서 주로 사용되며, 확률 값을 기반으로 클래스에 대한 예측을 수행할 수 있다.

Softmax 함수는 입력 벡터의 각 요소를 양수로 변환하고, 모든 요소의 합이 1 이 되도록 정규화하는 함수이다. Softmax 함수의 수식은 다음과 같다: f(x_i) = exp(x_i) / sum(exp(x_j)), 여기서 i 는 입력 벡터의 요소 인덱스이고 j 는 모든 요소의 인덱스를 나타낸다. Softmax 함수는 다중 클래스 분류 문제에서 주로 사용되며, 각 클래스에 대한 확률 값을 얻을 수 있다. Softmax 함수를 통해 출력된 값은 클래스 간 상대적인 확률을 나타내며, 가장 높은 값에 해당하는 클래스를 선택할 수 있다.

이제 주어진 x 에 대해 Sigmoid(x)와 Softmax(x)를 계산하면 다음과 같다.

x = [-0.5, 1.2, -0.1, 2.4]

1. Sigmoid(x):
Sigmoid 함수를 x 의 각 요소에 적용한다.
Sigmoid(-0.5) = 0.3775
Sigmoid(1.2) = 0.7685
Sigmoid(-0.1) = 0.4750
Sigmoid(2.4) = 0.9168

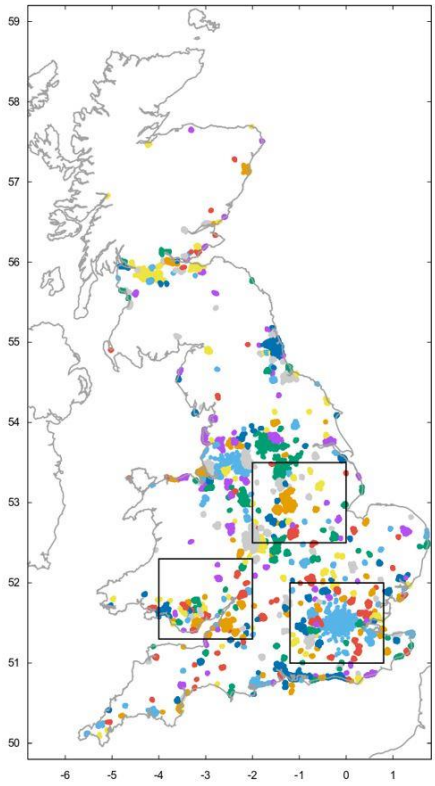
따라서, Sigmoid(x) = [0.3775, 0.7685, 0.4750, 0.9168]

2. Softmax(x):
Softmax 함수를 x 의 각 요소에 적용하여 정규화된 확률 값을 계산한다.
Softmax(-0.5) = 0.0577
Softmax(1.2) = 0.1992
Softmax(-0.1) = 0.0759
Softmax(2.4) = 0.6672

정규화된 값들의 합은 1 이 된다.

따라서, Softmax(x) = [0.0577, 0.1992, 0.0759, 0.6672]

5. The “urbanGB-simple.csv” is the coordinates (longitude and latitude) of 1000 road accidents occurred in urban areas in Great Britain. Perform k-means and DBSCAN clustering on this dataset. For k-means, find proper *k* by using distortion and silhouette analysis. For DBSCAN, find proper *epsilon* and *minPts* so that you can get the similar result to the K-means. Plot the clusters and outliers. (20 pts)



Code
import pandas as pd import numpy as np df = pd.read_csv("urbanGB-simple.csv") X = df.values from sklearn.preprocessing import StandardScaler sc = StandardScaler() X = sc.fit_transform(X)

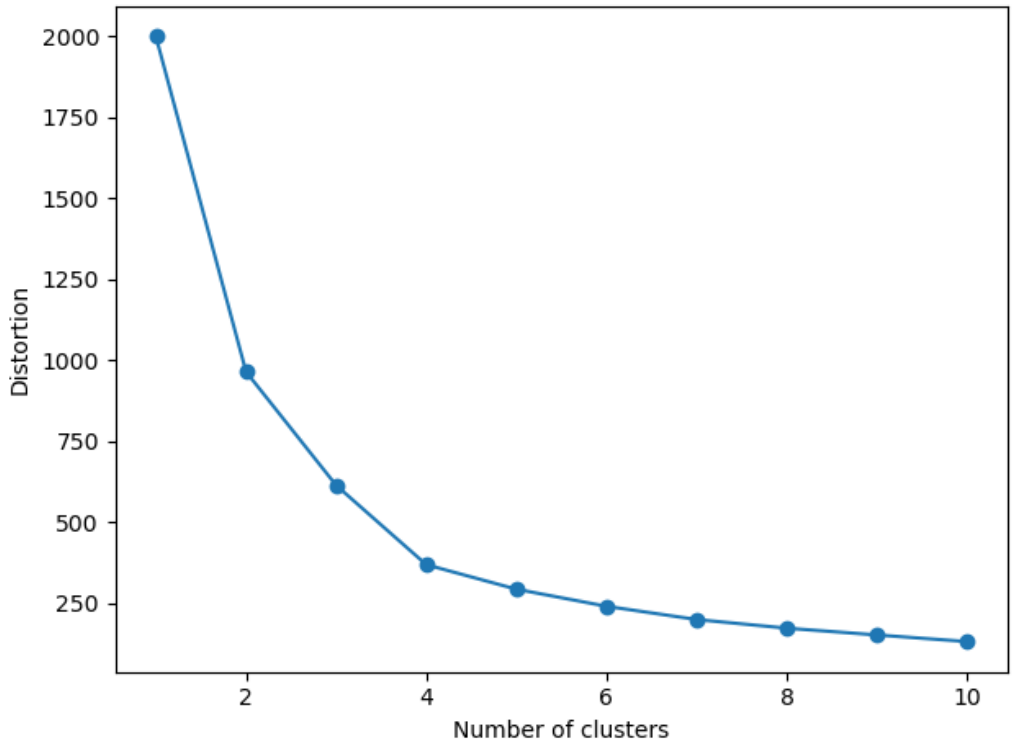
```
from sklearn.cluster import KMeans
import matplotlib.pyplot as plt
distortions = []
for i in range(1, 11):
    km = KMeans(n_clusters=i, init='k-means++', max_iter=300, random_state=0)

    km.fit(X)
    distortions.append(km.inertia_)

plt.plot(range(1, 11), distortions, marker='o')
plt.xlabel('Number of clusters')
plt.ylabel('Distortion')
plt.tight_layout()

plt.show()

from sklearn.metrics import silhouette_score
silhouette_scores = []
for i in range(2, 11):
    km = KMeans(n_clusters=i, init='k-means++', max_iter=300, random_state=0)

    y_km = km.fit_predict(X)
    silhouette_scores.append(silhouette_score(X, y_km, metric='euclidean'))

plt.plot(range(2, 11), silhouette_scores, marker='o')
plt.xlabel('Number of clusters')
plt.ylabel('Silhouette score')
plt.tight_layout()

plt.show()

km = KMeans(n_clusters=4, init='k-means++', max_iter=300, random_state=0)
y_km = km.fit_predict(X)

plt.scatter(X[y_km == 0, 0], X[y_km == 0, 1],
            c='blue', marker='o', s=40,
            label='cluster 1')
plt.scatter(X[y_km == 1, 0], X[y_km == 1, 1],
```

```
            c='red', marker='s', s=40,
            label='cluster 2')
plt.scatter(X[y_km == 2, 0], X[y_km == 2, 1],
            c='green', marker='x', s=40,
            label='cluster 3')
plt.scatter(X[y_km == 3, 0], X[y_km == 3, 1],
            c='cyan', marker='d', s=40,
            label='cluster 4')

plt.title('K-means')
plt.legend()
plt.tight_layout()
plt.show()

from sklearn.cluster import DBSCAN
db = DBSCAN(eps=0.3, min_samples=14)
y_db = db.fit_predict(X)

plt.scatter(X[y_db == 0, 0], X[y_db == 0, 1],
            c='blue', marker='o', s=40,
            label='cluster 1')
plt.scatter(X[y_db == 1, 0], X[y_db == 1, 1],
            c='red', marker='s', s=40,
            label='cluster 2')
plt.scatter(X[y_db == 2, 0], X[y_db == 2, 1],
            c='green', marker='x', s=40,
            label='cluster 3')
plt.scatter(X[y_db == 3, 0], X[y_db == 3, 1],
            c='cyan', marker='d', s=40,
            label='cluster 4')
plt.scatter(X[y_db == -1, 0], X[y_db == -1, 1],
            c='black', marker='v', s=40,
            label='outlier')

plt.title('Density based clustering(DBSCAN)')
plt.legend()
plt.tight_layout()
plt.show()
```

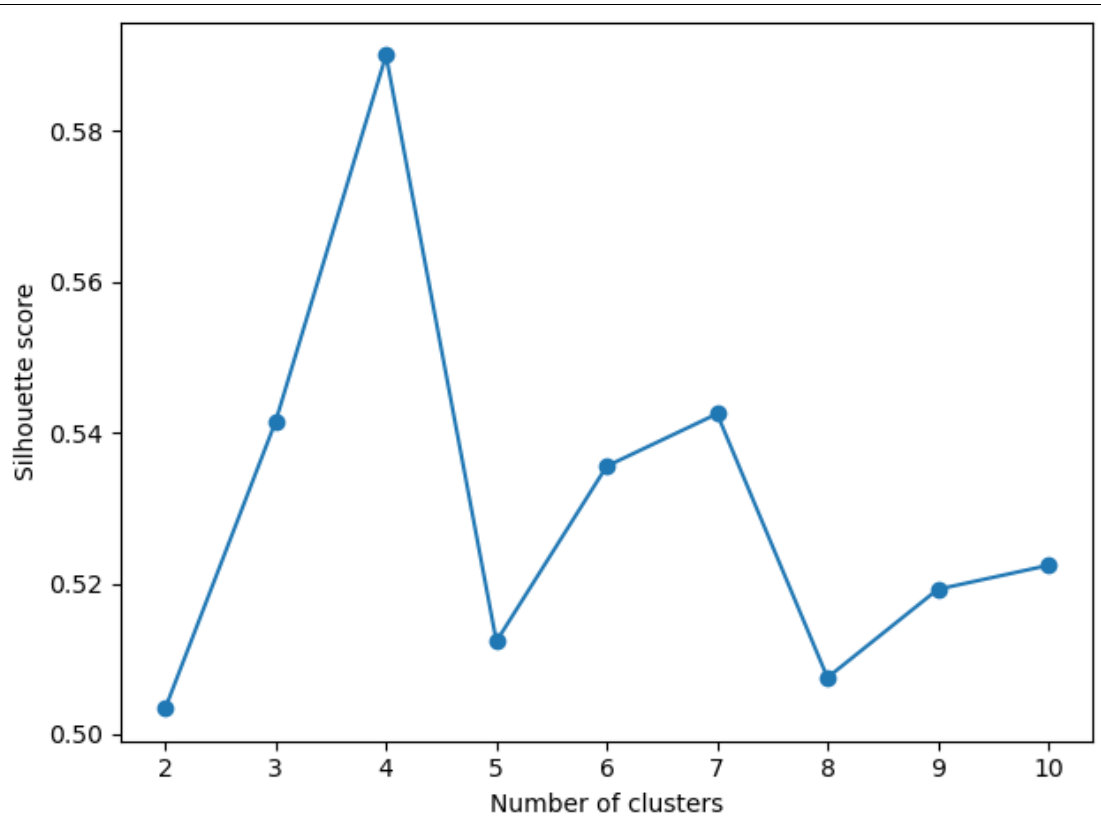
Result(Captured images)

1. k-means

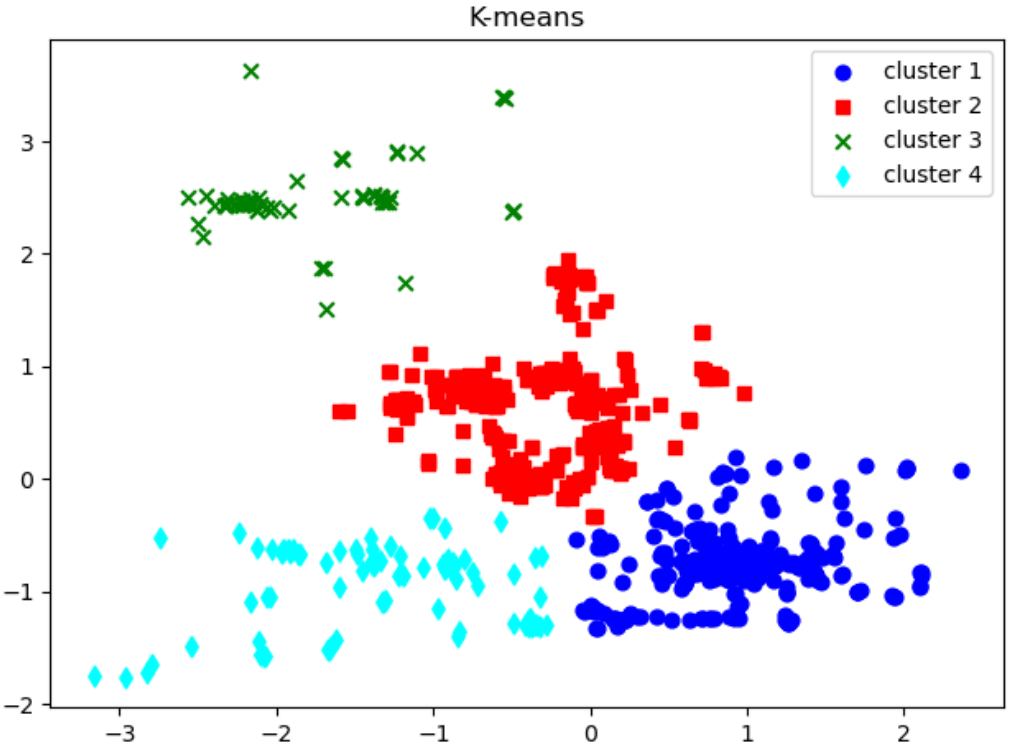
(1) distortion



(2) silhouette score

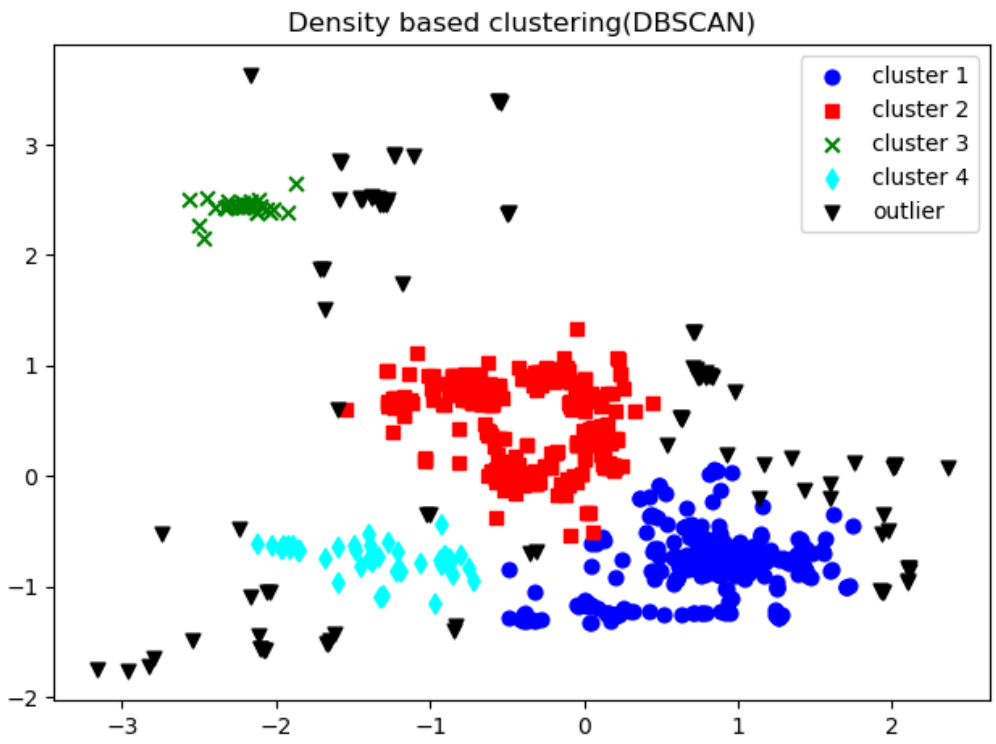


(3) Plot the clusters



2. DBSCAN

(1) Plot the clusters and outliers



Description

urbanGB-simple.csv 를 로드한 후, dataset 을 표준화한 뒤 모델의 학습을 진행하였다.
For 문을 이용해 k 값을 변화시키면서 distortion 과 silhouette score 값의 변화를 그래프로 plot 하여 적절한 k 값을 구하고자 했다. Distortion 그래프에서 갑자기 꺾이는 지점과 silhouette score 그래프에서 값이 가장 큰 값을 종합해서 살펴보면 k-means 에서 k 의 값은 4 가 적절한 것을 볼 수 있다.
그 후, k = 4 일 때 k-means 를 수행한 결과를 plot 하였더니 거의 균일하게 클러스터가 나눠진 모습을 볼 수 있다. 위의 plot 한 이미지를 살펴보면 파란색 점이 cluster 1, 빨간색 사각형이 cluster 2, 초록색 X 가 cluster 3, 하늘색 다이아몬드가 cluster 4 로 설정되어 있는데 균등하게 나뉜 모습을 다시 한번 확인할 수 있다.

DBSCAN 을 수행할 때 위의 k-means 처럼 클러스터의 개수가 3 개가 되도록 최적의 *epsilon* and *minPts* 을 찾은 결과 epsilon 은 0.3, minPts 는 14 가 나왔다. 이를 토대로 cluster 를 plot 하였더니 비교적 outlier 가 많이 나왔지만 k-means 와 유사한 모양을 가지고 있는 것을 확인할 수 있다.
위의 plot 한 이미지를 살펴보면 파란색 점이 cluster 1, 빨간색 사각형이 cluster 2, 초록색 X 가 cluster 3, 하늘색 다이아몬드가 cluster 4, 검은색 역 삼각형이 outlier 로 설정되어 있는데 cluster 1 과 2 가 k-mean 와 가장 유사하면서도 크기가 가장 크고, cluster 4, 2 순으로 크기가 작아지는 양상을 볼 수 있다. 또한 outlier 가 DBSCAN 에서 생기는 것을 볼 수 있다..

6. 20newsgroup dataset is a news dataset consisting of 20 categories. Apply Logistic Regression and Decision Tree Classifier on 20newsgroups dataset to build a document classification model. In this question, we only use samples from 4 categories. (30pts)

   Follow this process:

   1) Load the data. Use fetch_20newsgroups().

```
from sklearn.datasets import fetch_20newsgroups

categories = ['alt.atheism', 'soc.religion.christian',
              'comp.graphics', 'sci.med']

news = fetch_20newsgroups(categories=categories,
                          shuffle=False,
                          random_state=42)
X = news.data
y = news.target
```

```
X[0]
```

```
"From: keith@cco.caltech.edu (Keith Allan Schneider)\nSubject: Re: <
d.caltech.edu\n\nbobbe@vice.ICO.TEK.COM (Robert Beauchaine) writes:\
ral hypothesis, you have to successfully argue that\n>domestication
animals exhibit behaviors not found in the wild.  I\ndon't think tha
years\nto produce certain behaviors, etc.\n\nkeith\n"
```

```
from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test = train_test_split(X, y,
                                                    test_size=0.3,
                                                    stratify=y,
                                                    random_state=1)
```

   2) Make preprocessor function & porter stemmer tokenizer function.

   3) Apply the TF-IDF(TFidfVectorizer) on the data.

   - Use the preprocessor function & porter stemmer tokenizer
   - Use stop-words
   - Drop terms occurred in more than 10% of docs
   - Drop terms occurred in less than 10 docs

   4) Train models(Logistic Regression, Decision Tree) using TF-IDF vectors.

- Check the accuracies of the model
- Find out what are the most important words for this classification

5) Predict the categories of following 4 sentences

'The outbreak was declared a global pandemic by the World Health Organization (WHO) on 11 March.',

'Today, computer graphics is a core technology in digitalphotography, film, video games, cell phone and computer displays,and many specialized applications.',

'Arguments for atheism range from philosophical to social and historical approaches.',

'The Bible is a compilation of many shorter books written at different times by a variety of authors, and later assembled into the biblical canon.'

Code

```
from sklearn.datasets import fetch_20newsgroups

categories = ['alt.atheism', 'soc.religion.christian', 'comp.graphics', 'sci.med']
news = fetch_20newsgroups(categories=categories, shuffle=False, random_state=42)
X = news.data
y = news.target

X[0]

from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.3, stratify=y, random_state=1)

import re
def preprocessor(text):
    text = re.sub('<[^>]*>','', text)
    text = re.sub('[₩W]+',' ', text)
    text = text.lower()
```

```
    return text

from nltk.tokenize import word_tokenize
from nltk.stem.porter import PorterStemmer

stemmer = PorterStemmer()

def tokenizer_stemmer(text):
    text_tokens = word_tokenize(text)
    return [stemmer.stem(word) for word in text_tokens]

from nltk.corpus import stopwords
import nltk
nltk.download('stopwords')

stop = stopwords.words('english')

from sklearn.feature_extraction.text import TfidfVectorizer

tfidf = TfidfVectorizer(strip_accents=None,
                        lowercase=False,
                        preprocessor=preprocessor,
                        tokenizer=tokenizer_stemmer,
                        stop_words=stop,
                        min_df=10,
                        max_df=0.1
                       )
X_train_vector = tfidf.fit_transform(X_train)
X_test_vector = tfidf.transform(X_test)

from sklearn.linear_model import LogisticRegression

lr = LogisticRegression(penalty='l2', verbose=1)
lr.fit(X_train_vector, y_train)

print(f'train accuracy = {lr.score(X_train_vector, y_train)}')
print(f'test accuracy = {lr.score(X_test_vector, y_test)}')
```

```
import numpy as np
max_val = np.max(lr.coef_, axis=1)
idx = np.where(np.max(max_val, axis=0) == max_val)
print(f'most important term : {tfidf.get_feature_names_out()[np.where(lr.coef_[idx[0][0]]
==  np.max(lr.coef_[idx[0][0]], axis = 0))][0]}')

from sklearn.tree import DecisionTreeClassifier

tree = DecisionTreeClassifier(max_depth=20)
tree.fit(X_train_vector, y_train)

print(f'train accuracy = {tree.score(X_train_vector, y_train)}')
print(f'test accuracy = {tree.score(X_test_vector, y_test)}')

importances = tree.feature_importances_
indices = np.argsort(importances)[::-1]

for f in range(10):
    print("%2d. %-30s %f" % (f+1,
                             [w for w, n in tfidf.vocabulary_.items() if n == indices[f]],
                             importances[indices[f]]))

tweets = ['The outbreak was declared a global pandemic by the World Health Organization (WHO) on
11 March.',
          'Today, computer graphics is a core technology in digitalphotography, film, video games, cell
phone and computer displays,and many specialized applications.',
          'Arguments for atheism range from philosophical to social and historical approaches.',
          'The Bible is a compilation of many shorter books written at different times by a variety of
authors, and later assembled into the biblical canon.'
]

tweets_tfidf = tfidf.transform(tweets)

y_pred = lr.predict(tweets_tfidf)

for i in range(len(tweets)):
    if y_pred[i] == 0:
        print(tweets[i], "--> Negative")
```

```
    else:
        print(tweets[i], "--> Positive")

tweets = ['The outbreak was declared a global pandemic by the World Health Organization (WHO) on 11 March.',
          'Today, computer graphics is a core technology in digitalphotography, film, video games, cell phone and computer displays,and many specialized applications.',
          'Arguments for atheism range from philosophical to social and historical approaches.',
          'The Bible is a compilation of many shorter books written at different times by a variety of authors, and later assembled into the biblical canon.'
]

tweets_tfidf = tfidf.transform(tweets)

y_pred = tree.predict(tweets_tfidf)

for i in range(len(tweets)):
    if y_pred[i] == 0:
        print(tweets[i], "--> Negative")
    else:
        print(tweets[i], "--> Positive")
```

Result(Captured images)

1. Logistic regression
   (1) Accuracy

```
train accuracy = 0.9911336288790373
test accuracy = 0.9542772861356932
```

   (2) Most important term

```
most important term : graphic
```

   (3) Prediction

```
The outbreak was declared a global pandemic by the World Health Organization (WHO) on 11 March. --> Positive
Today, computer graphics is a core technology in digitalphotography, film, video games, cell phone and computer displays,and many specialized applications. --> Positive
Arguments for atheism range from philosophical to social and historical approaches. --> Negative
The Bible is a compilation of many shorter books written at different times by a variety of authors, and later assembled into the biblical canon. --> Positive
```

2. Decision tree
   (1) Accuracy

```
train accuracy = 0.8176060797973401
test accuracy = 0.7389380530973452
```

(2) Most important term

```
 1. ['graphic']                    0.136507
 2. ['christ']                     0.092242
 3. ['keith']                      0.078496
 4. ['islam']                      0.064112
 5. ['church']                     0.062738
 6. ['file']                       0.053615
 7. ['pitt']                       0.048363
 8. ['doctor']                     0.041248
 9. ['atheism']                    0.038496
10. ['faith']                      0.034632
```

(3) Prediction

```
The outbreak was declared a global pandemic by the World Health Organization (WHO) on 11 March. --> Positive
Today, computer graphics is a core technology in digitalphotography, film, video games, cell phone and computer displays,and many specialized applications. --> Positive
Arguments for atheism range from philosophical to social and historical approaches. --> Negative
The Bible is a compilation of many shorter books written at different times by a variety of authors, and later assembled into the biblical canon. --> Positive
```

Description

fetch_20newsgroups() 함수를 이용하여 데이터를 로드하고, 테스트 데이터를 전체 데이터셋의 30%만 사용한다. 그 후 preprocessing 함수와 porter stemmer tokenizer 함수를 정의한 다음 Tfidfvectorizer 에 인수로 전달하여 변환을 진행하기 전, preprocessing 과 stemming 을 진행할 수 있도록 한다. 그리고 stop word 를 제외시키고, 문서에서 10% 이상 나타나거나, 10 개 미만의 문서에서 나타나는 용어에 대해 제외하고 변환을 진행한다. 그리고 logistic regression 과 decision tree 을 변환한 데이터셋을 이용해 학습하고 예측을 진행한다.

1. Logistic regression
   (1) Logistic regression 모델의 정확도는 train data 의 경우 약 0.99, test data 의 경우 0.92 이다.
   (2) 학습을 진행한 후, weight 값을 비교하여 most important term 을 구해보면 graphic 이 나온다.
   (3) 4 개의 문장에 대하여 예측을 진행하면 각각 positive, positive, negative, positive 가 나온다.

2. Decision tree
   (1) Max depth 를 20 으로 설정하여 depth 가 너무 커지는 것을 방지하였다.
   (2) Decision tree 모델의 정확도는 train data 의 경우 약 0.82, test data 의 경우 0.74 로 logistic regression 보다 낮은 모습을 확인할 수 있다.
   (3) 학습을 진행한 후 tree.feature_importances_을 내림차순으로 정렬하고, 노드를 잘 구분하는 most important term 을 구해보면 graphic 이 나온다.

(4) 4 개의 문장에 대하여 예측을 진행하면 각각 positive, positive, negative, positive 가 나온다. 이는 logistic regression 과 같은 결과를 보인다.

Note

1. Summit the file to e-class as pdf.
2. Specify your file name as “hw3_<StudentID>_<Name>.pdf”