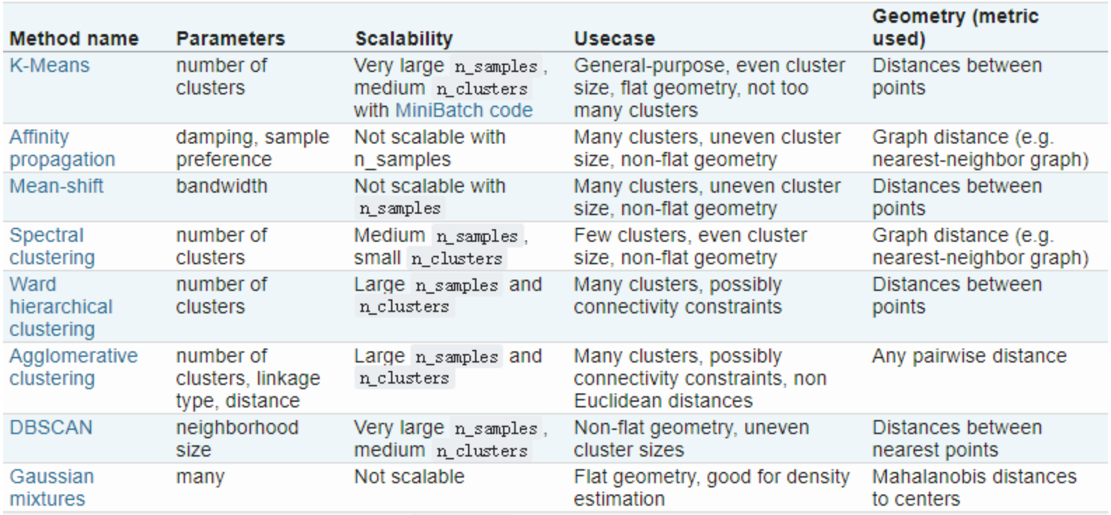
The Third Experimental Report

一、 实验内容

---测试 sklearn 中以下聚类算法在 tweets 数据集上的聚类效果。

---使用 NMI(Normalized Mutual Information)作为评价指标。



Method name	Parameters	Scalability	Usecase	Geometry (metric used)
K-Means	number of clusters	Very large `n_samples`, medium `n_clusters` with MiniBatch code	General-purpose, even cluster size, flat geometry, not too many clusters	Distances between points
Affinity propagation	damping, sample preference	Not scalable with n_samples	Many clusters, uneven cluster size, non-flat geometry	Graph distance (e.g. nearest-neighbor graph)
Mean-shift	bandwidth	Not scalable with `n_samples`	Many clusters, uneven cluster size, non-flat geometry	Distances between points
Spectral clustering	number of clusters	Medium `n_samples`, small `n_clusters`	Few clusters, even cluster size, non-flat geometry	Graph distance (e.g. nearest-neighbor graph)
Ward hierarchical clustering	number of clusters	Large `n_samples` and `n_clusters`	Many clusters, possibly connectivity constraints	Distances between points
Agglomerative clustering	number of clusters, linkage type, distance	Large `n_samples` and `n_clusters`	Many clusters, possibly connectivity constraints, non Euclidean distances	Any pairwise distance
DBSCAN	neighborhood size	Very large `n_samples`, medium `n_clusters`	Non-flat geometry, uneven cluster sizes	Distances between nearest points
Gaussian mixtures	many	Not scalable	Flat geometry, good for density estimation	Mahalanobis distances to centers

二、 数据集

The Tweets dataset is in format of JSON like follows:

{"text": "centrepoint winter white gala london", "cluster": 65}

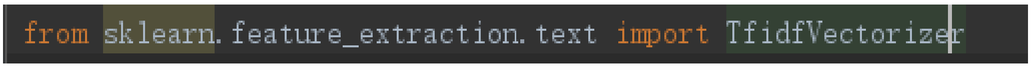
{"text": "mourinho seek killer instinct", "cluster": 96}

{"text": "roundup golden globe won seduced johansson voice", "cluster": 72}

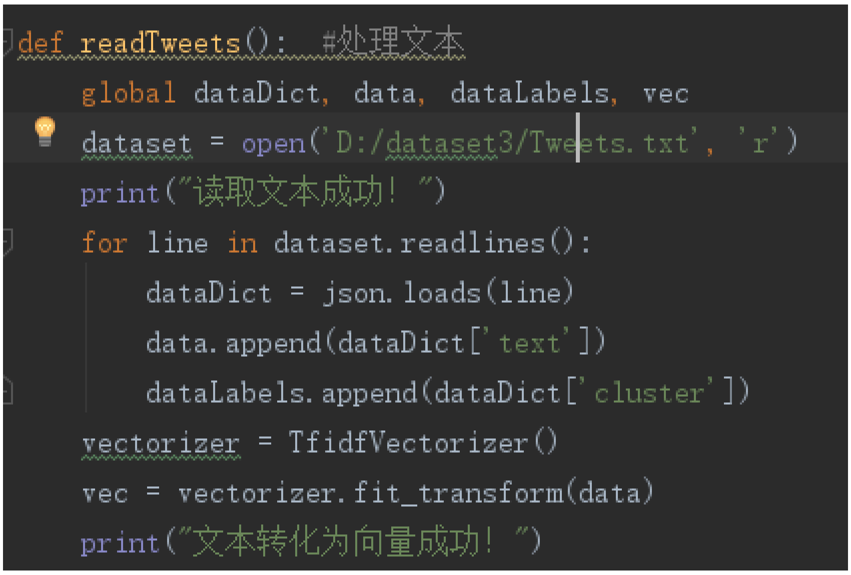
{"text": "travel disruption mount storm cold air sweep south florida", "cluster": 140}

三、 实验过程

（1） 先用读取数据集里的内容，然后用 sklearn 里的 TfidfVectorizer 函数处理数据，进行向量化。

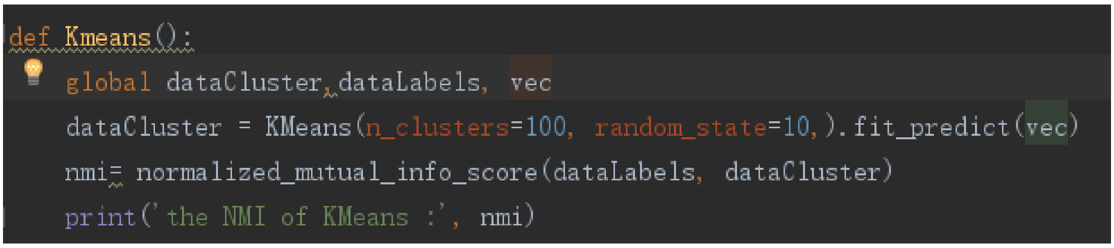


```python
from sklearn.feature_extraction.text import TfidfVectorizer
```



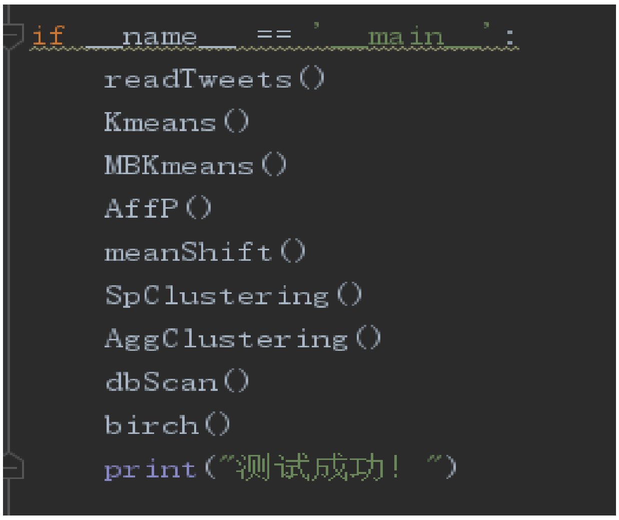
```python
def readTweets():  #处理文本
    global dataDict, data, dataLabels, vec
    dataset = open('D:/dataset3/Tweets.txt', 'r')
    print("读取文本成功！")
    for line in dataset.readlines():
        dataDict = json.loads(line)
        data.append(dataDict['text'])
        dataLabels.append(dataDict['cluster'])
    vectorizer = TfidfVectorizer()
    vec = vectorizer.fit_transform(data)
    print("文本转化为向量成功！")
```

（2） 依次编写函数，用 nmi 来测试每个聚类方法的性能。这里取 cluster 的数目为 100。



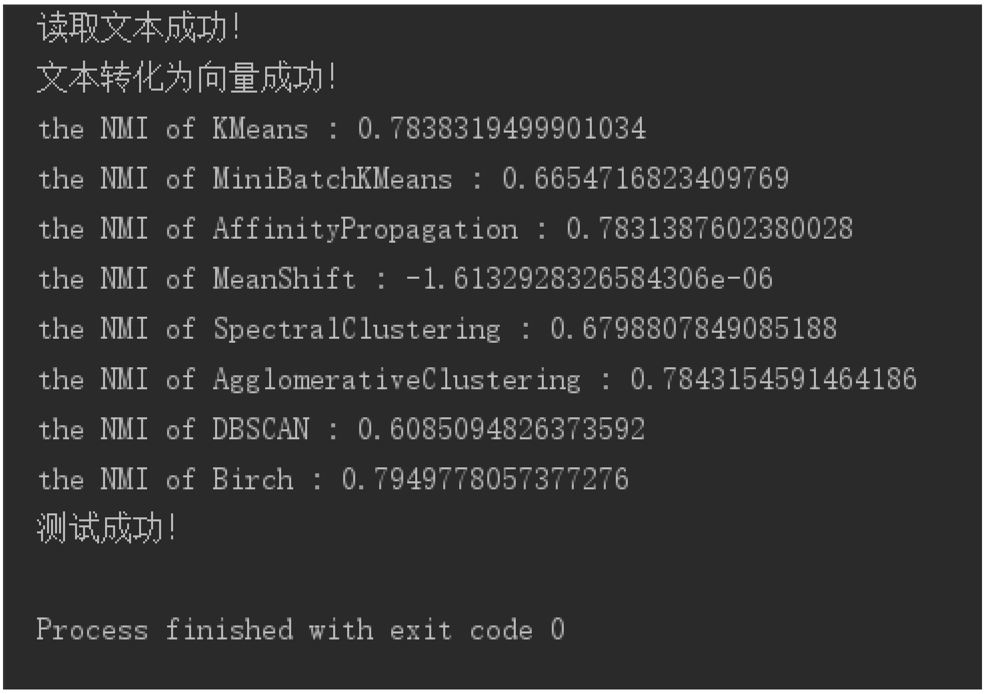
```python
def Kmeans():
    global dataCluster,dataLabels, vec
    dataCluster = KMeans(n_clusters=100, random_state=10,).fit_predict(vec)
    nmi= normalized_mutual_info_score(dataLabels, dataCluster)
    print('the NMI of KMeans :', nmi)
```

（3） 在 main 函数里依次运行八个聚类函数。



```python
if __name__ == '__main__':
    readTweets()
    Kmeans()
    MBKmeans()
    AffP()
    meanShift()
    SpClustering()
    AggClustering()
    dbScan()
    birch()
    print("测试成功！")
```

四、 实验结果



```
读取文本成功!
文本转化为向量成功!
the NMI of KMeans : 0.7838319499901034
the NMI of MiniBatchKMeans : 0.6654716823409769
the NMI of AffinityPropagation : 0.7831387602380028
the NMI of MeanShift : -1.6132928326584306e-06
the NMI of SpectralClustering : 0.6798807849085188
the NMI of AgglomerativeClustering : 0.7843154591464186
the NMI of DBSCAN : 0.6085094826373592
the NMI of Birch : 0.7949778057377276
测试成功!

Process finished with exit code 0
```

五、 实验心得

Scikit-learn(sklearn)是一种简单有效的数据挖掘和数据分析工具，它对常用的机器学习方法进行了封装，比如回归、降维、分类、聚类等方法。本次实验是对 sklearn 里的八种聚类方法，k 均值聚类、层次聚类、谱聚类、吸引子传播等进行测试。使用标准互信息作为聚类方法的评估指标。由实验结果可以看出 Birch 方法表现最好。

但不知为何，MeanShift 方法的结果为负数，有点不理解。

总的来说，通过本次实验，对于聚类有个更深层次的理解，也体会到了 sklearn 的便利。有时间可以多阅读 Scikit-learn API 来学习一下 sklearn