**데이터 전 처리 및 분석 기술 학습**

RELATED WORK

1. *Classification*

* **Random Forest**

Random Forest는 bagging을 이용한 알고리즘이다. 추출한 데이터들을 바탕으로 여러 개의 독립적인 트리를 만들고 각각 개별적으로 학습을 수행하는 방식이다. 마지막에 예측치들에 대한 투표 또는 평균값으로 최종 예측치를 도출한다.

학습에는 Random Forest Classifier 모델을 사용하였으며, 랜덤 포레스트 안에 만들어지는 의사결정 나무 개수인 n estimators 파라미터를 사용했다.

\*bagging: 샘플들을 여러 번 뽑아 각 모델을 학습시켜, Aggregation 결과물을 집계한 것

* **Naive Bayes**

Naive Bayes의 Naive는 순진하다 라는 뜻이며 이는 데이터셋의 모든 특징들이 동등하고 독립적이라는 가정을 반영한 것이다. Bayes는 Bayes 정리를 말한다. Naive bayes는 가장 단순한 지도 학습 중 하나이며, Feature에 따라 Label을 분류하는데 bayes 정리를 사용하는 것이 특징이다. Naïve Bayes 모델은 크게 두가지를 사용하는데, 스팸 메일과 같이 특정 단어가 있다/없다 와 같이 이진 속성을 가지며, 속성이 아주 많은 경우에는 Bernoulli NB를 주로 적용하고, 일반적인 연속 값 속성을 가지는 데이터에는 Gaussian NB를 적용한다. 학습에는 Gaussian NB를 사용하였다.

* **Support Vector Machine**

SVM은 분류문제를 해결하기위해 사용되는 supervised learning 모델 중 하나이다. 주로 분류문제를 다루지만, 회귀 문제에도 사용이 가능하다. SVM은 임의의 두 클래스에 대해 이진분류를 할 때, 서로를 나누는 결정 경계의 margin이 최대화할 수 있도록 학습한다. 여기서 margin이란 결정 경계와 데이터 간의 거리를 의미하고 결정경계는 feature에 대한 n차원의 초평면이 된다. 즉 feature가 2개일 때는 선으로, 3차원일때는 면으로 feature의 개수와 비례해서 늘어나게 된다.

다만 이러한 방식은 Outlier에 의해 적합한 결정 경계를 찾기가 어려울 수 있다. 이처럼 결정 경계와 margin 사이의 이상치를 허용하지 않는 방식을 Hard margin이라고 한다. 반대로 적절하게 이상치를 허용하는 방식은 Soft margin이라 한다.

학습에는 soft margin을 사용하였고, 이상치에 대한 제약조건이 변화할 때 정확도가 얼마나 변하는지를 측정해 보았다.

1. *Clustering*

* **K-Means**

K-Means는 주어진 데이터를 k개의 cluster로 묶는 알고리즘으로, 각 클러스터와 거리차이의 분산을 최소화하는 방식으로 동작한다. Unsupervised learning의 일종으로 따로 labeling 된 데이터를 필요로 하지는 않지만, 몇 개의 cluster로 나눌 것인지에 대한 매개변수를 꼭 설정해 줘야 한다.

K-Means는 ① 먼저 얼마나 많은 cluster가 필요한지 결정하고, ② 초기 centroid를 선택한다. 이 때 centroid의 위치를 임의로 설정할지 수동으로 설정할 지 결정해야 한다. ③ 그리고 모든 데이터를 순회하며 각 데이터마다 가장 가까운 Centroid가 속한 cluster로 assign한다. ④ Centroid는 cluster의 중심으로 이동하고 ⑤ 클러스터에 assign 되는 데이터가 없을 때까지 (Centroid가 움직이지 않을 때까지) ③, ④번을 반복한다.

학습에서는 k값은 데이터에 알맞은 개수로 주었고, centroid를 임의로 설정하여 10번 실행한 결과의 평균 값을 사용했다.

* **DBSCAN**

DBSCAN은 밀도기반 clustering 알고리즘이다. K-means보다는 다소 느릴 수 있으나 따로 cluster의 개수를 미리 지정하지 않는다. 대신 데이터의 반경 (거리) 값인 epsilon과 반경안에 몇 개의 데이터를 포함하는지를 나타내는 값인 min point를 정할 수 있다.

각 instance에 대해서 그 instance가 epsilon 만큼의 반경에 min points 만큼의 instance를 포함하고 있으면 core point라고 하며 클러스터에 포함시킨다.

min points 만큼의 instance를 포함하지 않더라도 core point를 포함하고 있는 instance는 border point라고 부르며 마찬가지로 클러스터에 포함시킨다.

만약 아무런 instance를 포함하지 않는다면 noise point라고 부르며 cluster에서 제외한다.

DBSCAN은 K-means로는 불가능한 cluster도 판별이 가능하고 노이즈에 강하다는 장점을 가진다.

학습에서는 min points와 eps를 변경해보며 언제 가장 좋은 결과가 나오는지를 분석해 보았다.

* **Hierarchical Clustering**

계층적 군집화는 개체들을 가까운 집단부터 차근차근 묶어 나가는 방식이다. 군집화 결과뿐만 아니라 유사한 개체들이 결합되는 Dendrogram을 생성하고 이를 일정 높이에서 잘라 군집의 수를 결정하는 방식이다. ①거리를 계산하는 방법과 ②클러스터 간의 거리를 측정하는 방법에 따라 결과가 달라지는데, ①거리를 측정하는 방법에는 Manhattan distance, Euclidean distance, mahalanobis, 등의 방법이 있으며, ②클러스터 간의 거리를 측정하는 방법에는 single linkage, complete linkage, average linkage 등이 있다. 학습에는 Agglomerative Clustering 모델을 사용하였으며 파라미터의 경우 n clusters(클러스터의 개수), affinity(①거리를 계산하는 방법), linkage(②클러스터 간의 거리를 측정하는 방법) 3가지를 설정해주었다. 정확도 측정에는 조건을 동일하게 설정해주기 위해 affinity는 ‘Euclidean’으로, linkage는 ‘average’(평균 기준법, 한 군집 안에 속한 모든 데이터와 다른 군집에 속한 모든 데이터의 두 집단에 대한 거리 평균을 계산하여 가까운 데이터끼리 군집화)로 통일하여 진행하였다.

**TABLE Ⅰ**

DESCRIPTION OF DATA SETS

|  |  |  |  |
| --- | --- | --- | --- |
|  | # of classes | # of instances | # of features |
| Iris | 4 | 150 | 4 |
| Wine | 3 | 178 | 13 |
| Star | 6 | 240 | 6 |
| Cancer | 2 | 569 | 32 |

Star data의 경우 Color와 Spectral Class가 Object type이여서 2개의 feature를 삭제했다. 나머지 데이터는 따로 수정하기 않고 Scikit learn의 Data Set을 가져와서 사용했다.

**Classification**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | iris | wine | breast\_cancer | star |
| Random forest | 93.3% | 97.8% | 97.2% | 98.3% |
| Naive bayes | 95.5% | 90.7% | 95.3% | 85.4% |
| SVM | 99.2% | 99.3% | 96.9% | 99.0% |

**Clustering**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | iris | wine | breast\_cancer | star |
| K\_Means | 93.05% | 87.76% | 75.41% |  |
| DBSCAN | 67.16% | 59.86% | 37.79% | 52.61% |
| HC | 80.5% | 34.7% | 8.8% | 34.2% |