1. 과제 설명

이번 과제는 주어진 데이터를 기반으로 피싱 사이트를 분류하는 모델의 제작이다. SVM과 Naive bayes의 2가지 알고리즘을 사용한다. 데이터 세트에는 과제 PPT의 Dataset Info를 참고바란다. 알고리즘에 사용된 변수들은 각각 X에 들어간 칼럼은 Dataset Info에 들어간 값들이고 Y 값에 들어가는 칼럼은 [status]이다. Status는 해당 url로 접속했을 때 정상적인 사이트라면 legitimate를, 피싱이라면 phising이라는 문자열을 담고있다. 마지막으로, 그림, 표 등을 포함하며 각 알고리즘을 분석하고 평가하는 내용을 보고서로 작성해 제출해야 한다.

2. 각 알고리즘 분석과 결과

2-1. 두 알고리즘 동일 적용 사항

아래의 2개의 알고리즘에 적용되는 데이터 칼럼 중 url 관련 칼럼을 제거했다. 이는 먼저, url이 피싱 사이트 판단에 관련 없는 자료이기 때문이다. url 값은 사이트마다 다르게 설정되어있고, 다른 칼럼은 수치로 된 값인 반면에, url 값은 문자열로 되어있어서 분류가 어렵다. 또한, 정규화를 위해서는 모든 칼럼 값이 수치로 이루어져야 하는데, url 값을 수치화시킬 명확한 기준이 없기에 제거했다.

2-2. SVM

첫번째로, SVM을 사용하여 피싱 사이트를 분류하는 모델을 만들었다. 위에서 주어진 데이터 세트를 train data 70%, test data 30%로 나누고 test data에 대한 성능을 평가한다.

SVM은 Hyperplane의 Margin을 최대화 하는데 목적이 있는 모델로서, 일반화에 유리한 경향을 보인다. 이를 위해 StandardScaler를 이용하여 관련 데이터들을 정규화했다. 해당 모델은 수많은 주어진 데이터를 통해 새로 들어온 데이터의 칼럼들을 분석하여 새로이 들어온 데이터의 url이 피싱인지 정상적인 사이트인지 확인한다.

해당 모델은

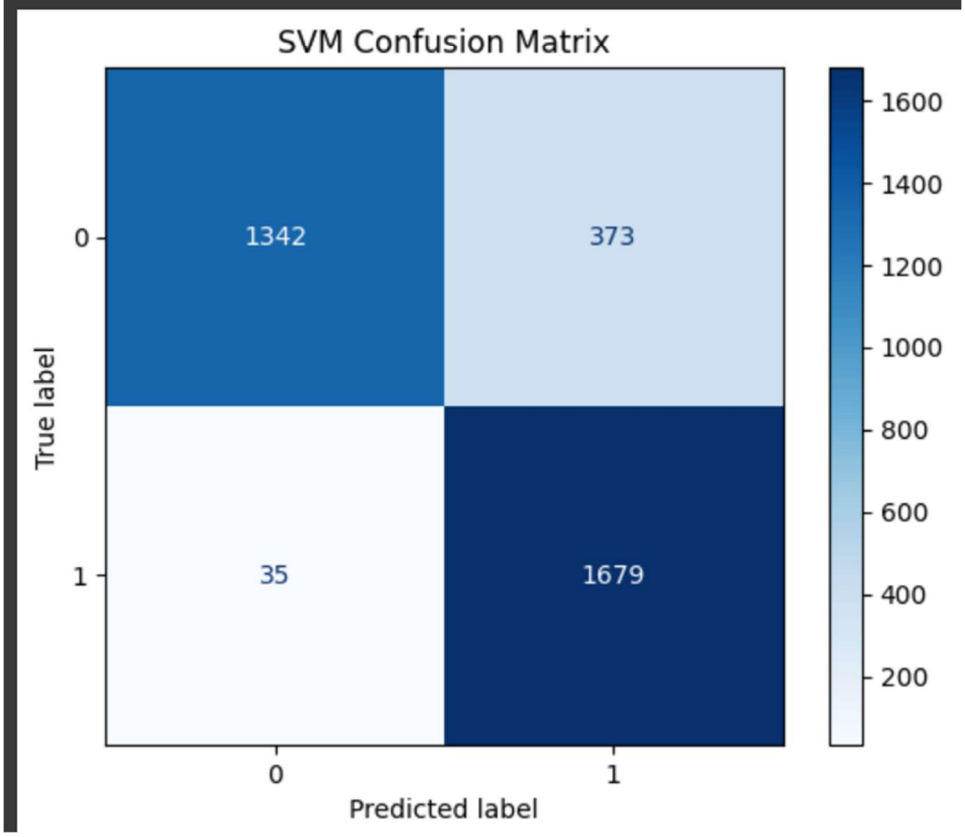
```
Svm = ss.SVC(kernel='rbf', C=1, gamma=0.5, random_state=50)
```

위의 코드를 통해 생성된다. 여기서 kernel은 커널 함수를 지정하는 매개변수로서, 'linear', 'poly', 'sigmoid'등이 있지만 기본값인 rbf를 사용한다. 다음으로 C는 데이터가 margin을 위반하는 것을 합한 값을 의미한다. 이 값은 데이터에 얼마나 더 민감하게 반응하는지에 비례한다. 고로 C를 아주 큰 값으로 하게 된다면 overfitting으로 새로운 데이터가 입력되었을 때의 성능이 저하될 수 있다. gamma값은 특정 데이터

가 미치는 영향의 범위를 의미한다. 이 값이 작을수록 부드럽고 일반화된 결정 경계를 만든다.

SVM result

	precision	recall	f1-score	support
0	0.97	0.78	0.87	1715
1	0.82	0.98	0.89	1714
accuracy			0.88	3429
macro avg	0.90	0.88	0.88	3429
weighted avg	0.90	0.88	0.88	3429



위의 그림을 통해 0(legitimate)와 1(phising)이 약 88%의 정확도로 구분된 것을 확인할 수 있다.

2-2. Naïve bayes

두번째로, Naïve bayes를 사용하여 피싱 사이트를 분류하는 모델을 만들었다. 위에서 주어진 데이터 세트 또한 train data 70%, test data 30%로 나누고 test data에 대한 성능을 평가한다.

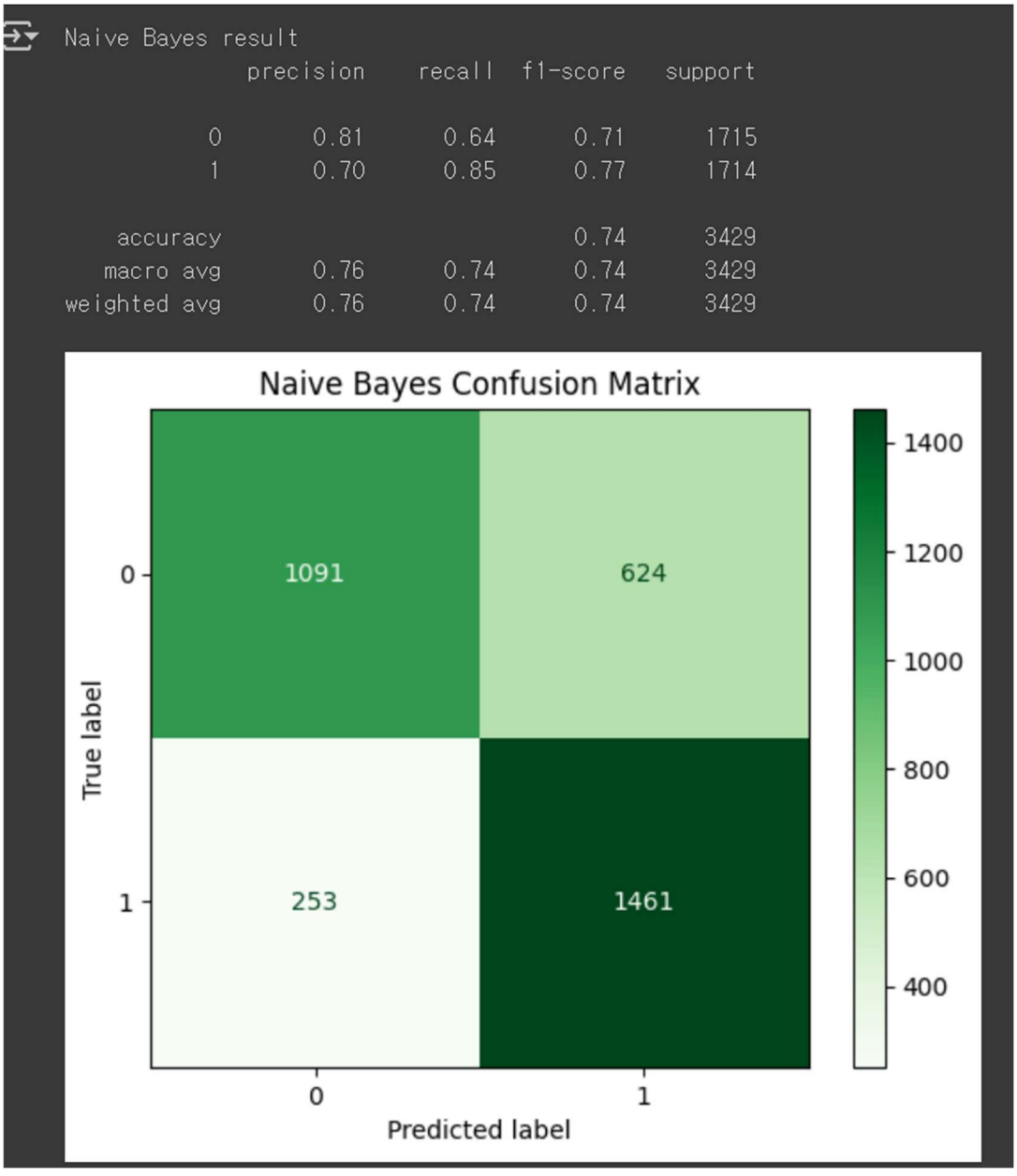
Naïve bayes는 아래의 수식을 기반으로 한 모델로서,

$$P(Y|X) = \frac{P(X|Y)P(Y)}{P(X)} \quad 이때 P(X \cap Y) = P(Y)P(X|Y) = P(X)P(Y|X)$$

조건부 확률을 기반으로 하여 값들을 분류한다. 각 항은 아래와 같은 의미를 지닌다.

- P(Y|X) 사후확률, 사건 X가 발생한 후 갱신된 사건 Y의 확률
- P(Y) 사전확률, 사건 X가 발생하기 전에 가지고 있던 사건 Y의 확률
- P(X|Y) 가능도, 사건 Y가 발생한 경우 사건 X의 확률
- P(X) 정규화 상수, 확률의 크기 조정

이를 이용해 nb = sn.GaussianNB() 의 코드에서 사용하여 분류를 진행한다.



위의 SVM과 달리 정확도가 약 74%로 낮은 것을 확인할 수 있다.

3. 결론

Naïve bayes가 svm보다 정확도가 낮은 이유는 여러가지가 있지만 먼저, 독립성의 문제이다. Naïve bayes는 조건부 확률을 이용하는데 이때, 각 속성들이 조건부 독립이라고 가정한다. 현실 데이터에서는 위의 가정이 성립하지 않게되고, 이는 분류 정확도의 저하로 이어진다. 데이터 칼럼에서 suspicious_tld는 의심스러운 최상위 도메인 사용 여부를 나타내는 값으로, 이는 피싱 사이트와 독립이 아닌 연관이 있을 수 있다. 고로, 분류 정확도가 낮게 측정되었다.

다음으로는, 결정 경계 모양이다. Naïve bayes는 선형 결정 경계를 가지는 모델로 svm처럼 비선형 경계를 가지는 모델보다 정확하게 데이터를 분리하지 못할 수도 있다. 위의 SVC 함수에서 kernel 값을 여러 개의 옵션 중에서 선택할 수 있었던 것처럼 svm이 다양하고 정확한 분류를 할 수 있다.

따라서 SVM은 복잡한 경계나 고차원 데이터에 사용하면 좋고, Naïve Bayes는 독립적인 성질은 가진 데이터나, 단어 기반의 데이터에 사용하면 좋은 성능을 발휘한다.