NSL\_5 데이터 분석

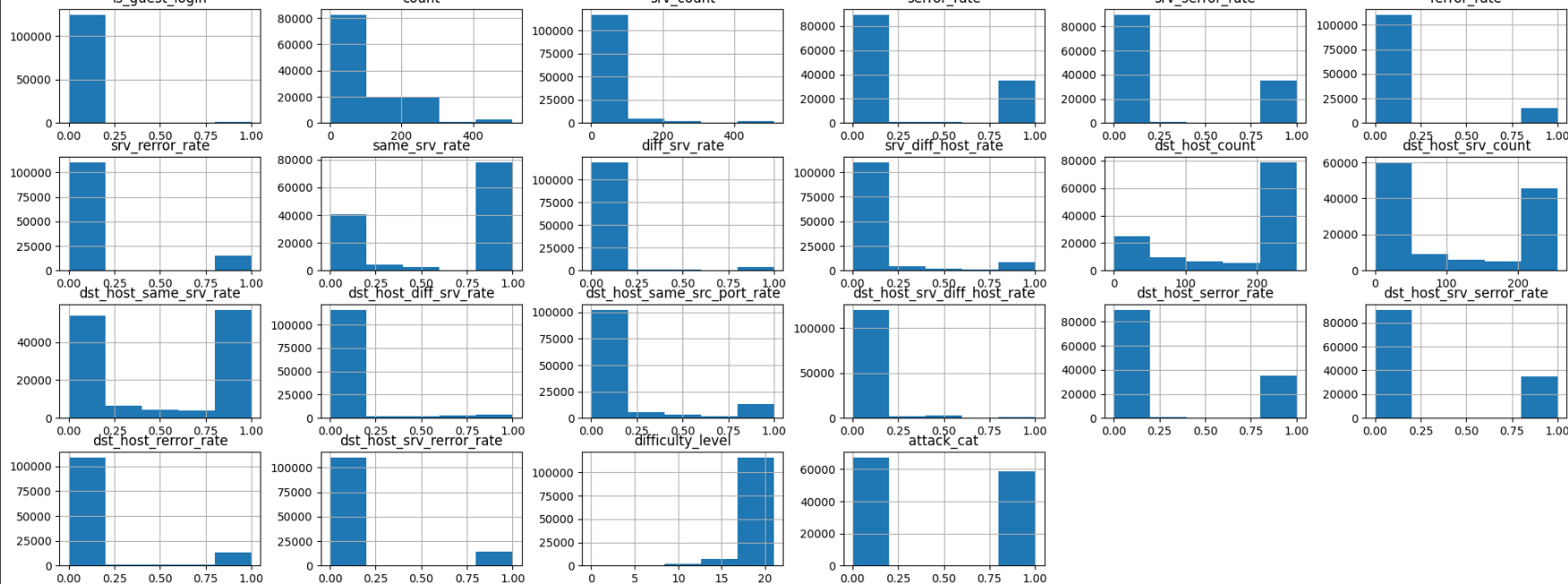
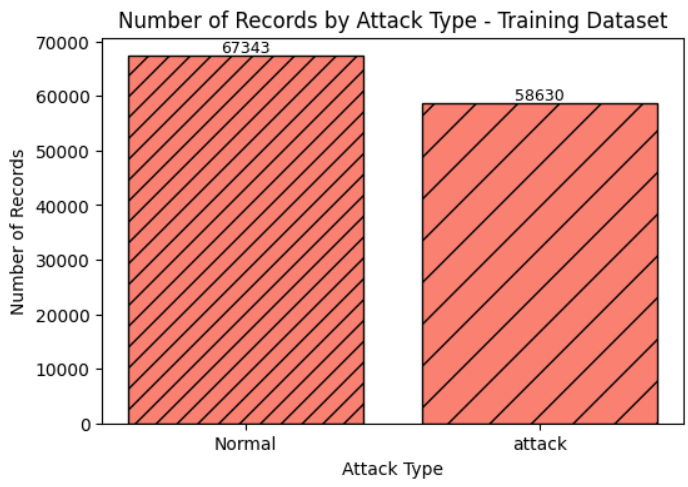
빅데이터응용보안

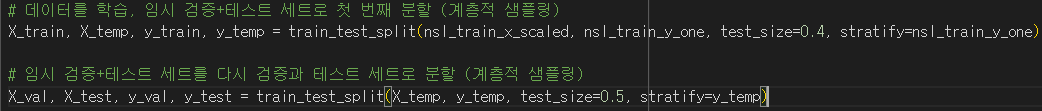
202220664 사이버보안학과

허한빈

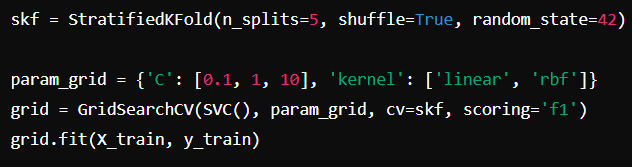
훈련 데이터를 보았을 때 normal 데이터와 attack 데이터 크기 차이가 그다지 크지 않고, 이중 분류로 진행했기 때문에 모델의 복잡도 감소와 과적합 위험 감소 등의 장점이 있을 거라 예상하였다.

그러나 전체적으로 모델의 평가가 AUC 0.79, f1socre 0.74 라는 다소 기대에 미치지 못하는 결과값들이 나왔다. 이후 원인을 추정했을 때 첫번째로 데이터가 distribution이 양극단적으로 구성되어 있다는 점을 보았다. 훈련과 시험 데이터 분포가 다소 차이가 있어 이를 해결하기 위해 계층적 샘플링을 시도했다. 훈련&학습 데이터로 나누어져 있는 데이터를 합쳐 공격&정상 데이터 비율을 맞췄을 때 RF 기준으로 accuracy 0.791로 0.02% 상승 결과를 얻었다.





SVM: KFold를 사용하여 데이터를 나누고, GridSearch을 사용하여 최적의 C와 kernel 파라미터을 찾으며 조정하였다.



**KNN**: StandardScaler 를 사용하여 스케일링된 데이터를 가지고 n\_neighbors 계수를 조정하였다.

**CNN**: 입력층에서 input shape를 데이터셋에 맞춰 입력하는 부분과 출력층에서 클래스를 동일하게 하는 부분을 중점적으로 보았다. nsl\_val\_x\_scaled.shape # (37792, 59)에 맞춰 입력층 59, 출력층 이진분류 클래스에 맞춰 조정하였다. 이후 Conv2D와 MaxPooling2D 레이어를 여러 개 시도해보면서 다시 모델을 검증했다.





DNN: dropout 값 조정.

XGboost: gridserach를 통한 최적의 파라미터 조정.

SVM:

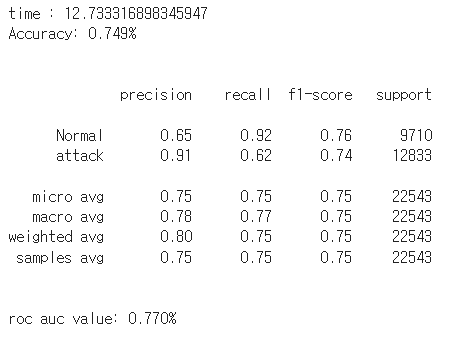


Figure 1 SVM 선형 커널 결과

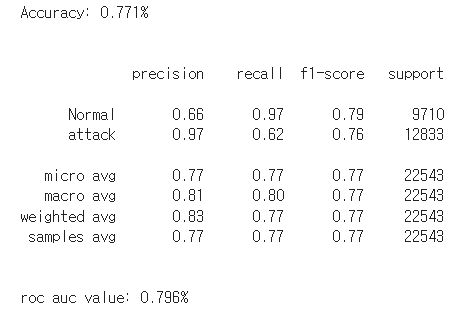


Figure 2 RF 결과

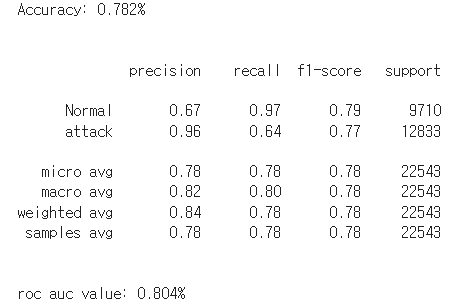


Figure 3 KNN result  
n\_neighbors=3

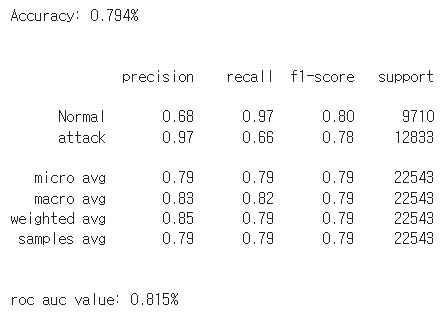


Figure 4 XGBoost result  
 gamma=5, max\_depth=12, n\_estimators=32, random\_state=42

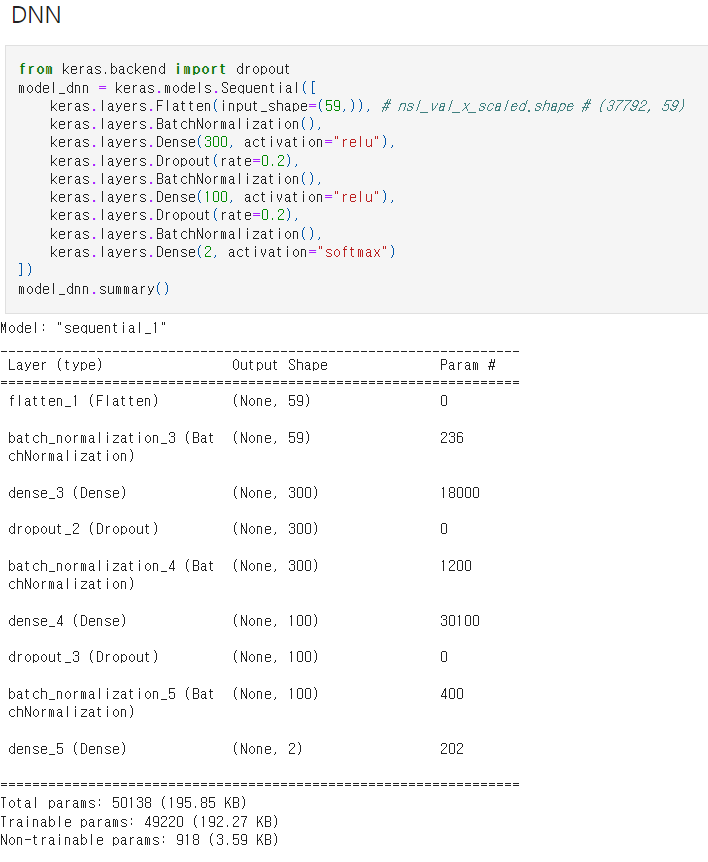


Figure 5 DNN

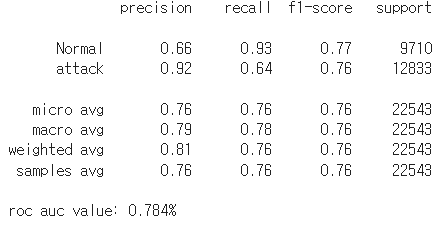


Figure 6 DNN result

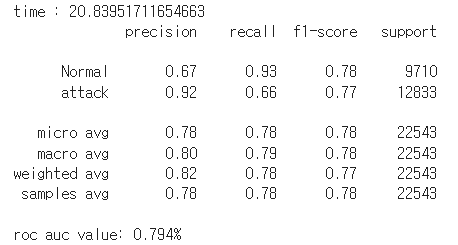


Figure 7 CNN result