

1) 문제정의

“QRS 파형은 심실 탈분극에 의해 형성되며, 3 개 파로 구성되어 있다. 12 유도 심전도에서 QRS 파형으로 심방 동수, 심장 전기축과 회전 정도를 평가할 수 있고, 심실 내 전도 이상 유무도 알 수 있다. 또한 심실비대, 심근경색증, Wolf-Parkinson-White (WPW)증후군도 진단할 수 있다. 따라서 정상 QRS 파와 비정상 QRS 파형에 대하여 알아보는 것이 심전도를 이해하는 데 중요하다.” [1] 따라서 본 실험에서는 주어진 데이터를 기반으로 정상 QRS 파와 비정상 QRS 파를 구분한 후 이를 탐지하는 방법을 제안하고자 한다.

실험에서 주어진 LUDB 에 있는 데이터 셋은 200 명의 각 환자에 대해 Figure1 과 같이 주어져 있으며, 각 12 개의 lead 에 대하여 P 파, QRS 파, T 파로 구성되어 있다.

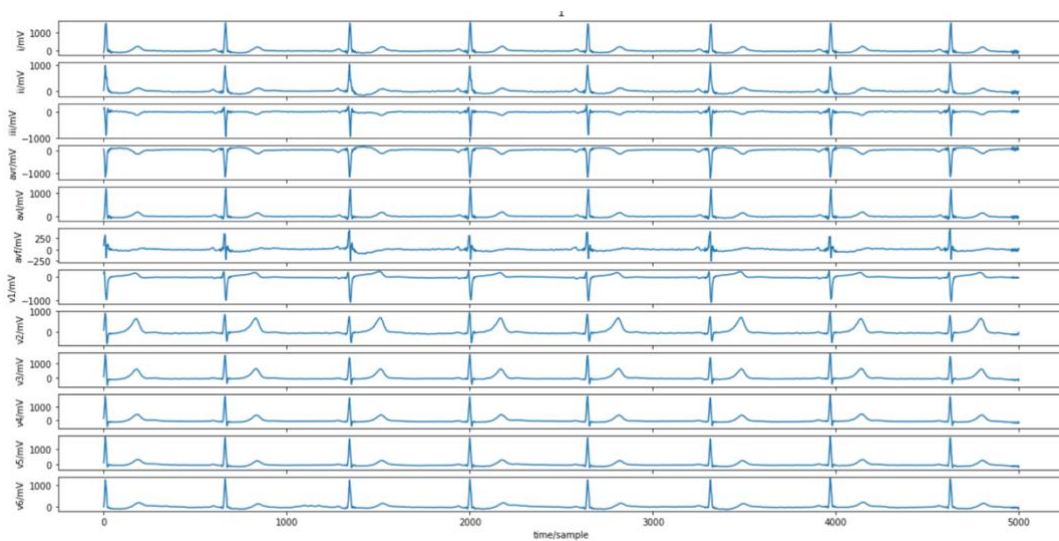


Figure 1

또한 각 환자에 대한 comment 가 있으며, Figure 2 와 같이 정상과 비정상에 대한 라벨링이 되어있다.

Electric axis of the heart	Number of subjects
Normal	75
Left axis deviation	66
Vertical	26
Horizontal	20
Right axis deviation	3
Undetermined	10
<i>Total</i>	200

Figure 2

따라서 실험을 위해 주어진 데이터셋을 사용하여 아래와 같은 가정을 세워 QRS 의 정상과 비정상군을 탐지하는 모델을 제안하고자 한다.

2) 모델을 사용하는 가정

본 실험에서 제안하는 모델을 사용하기 위해 다음과 같은 가정을 세워 탐지 하였다.

가정 1. 12 개의 leads 중 II 의 lead 만 사용하여 정상과 비정상 QRS 를 탐지한다.

가정 2. QRS 의 심방 동수가 120 이상, 60 미만인 경우 비정상이라고 한다.

가정 3. Electric axis of the heart 가 Left axis deviation, Vertical, Horizontal, Right axis deviation 인 경우 비정상 QRS 라고 한다.

가정 1 을 설정한 이유는 선행 연구에 의하면 12 개의 leads 중 II 의 lead 만 사용해도 정확한 결과값이 도출되기 때문에 본 실험에서도 II 의 lead 만을 사용하여 정상과 비 정상군을 탐지하였다 [2].

가정 2 를 설정한 이유는 선행 연구에 의하면 QRS 가 비정상으로 분류될 경우 심방 동수가 120 이상, 60 미만 [3] 이라고 정의하였기 때문에 본 실험에서도 동일한 기준으로 비정상 QRS 를 라벨링 하였다.

가정 3 의 경우도 마찬가지로 선행연구에서 Electric axis of the heart 가 Left axis deviation, Vertical, Horizontal, Right axis deviation 일 경우 비정상이라고 정의하였기 때문에 동일한 기준으로 비정상 QRS 를 라벨링 하였다.

최종적으로 데이터는 아래의 Figure 3 과 같이 비정상 1226 (67.0%), 정상 605 (33.0%)로 라벨링이 되었다.

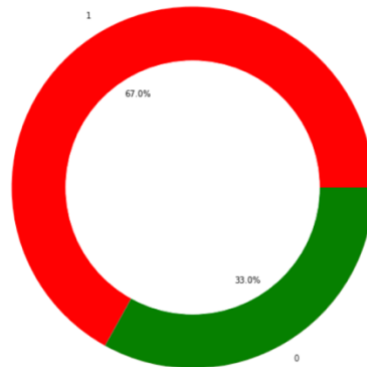


Figure 3

모델에서 사용한 feature 의 경우 onset, offset, peak, duration 을 사용하였다. Onset 의 경우 아래 Figure 4 에서 Q 의 위치정보를 의미하며 peak 의 경우 R 의 위치 정보를, offset 의 경우 S 의 위치정보를 의미한다. 그리고 Duration 은 QRS 의 심방 동수를 의미한다.

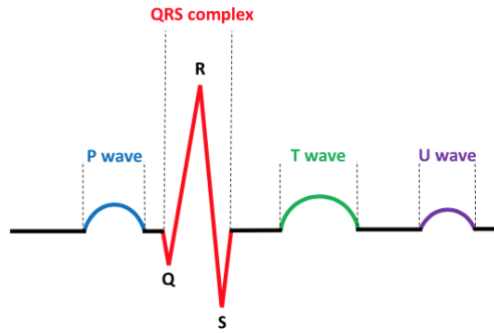


Figure 4

이때, 전체 데이터가 가진 feature 들의 scale 의 차이가 심하게 나기 때문에 모든 데이터 포인트가 동일한 정도의 중요도로 반영될 수 있도록 정규화를 한 후 모델을 학습 시켜 비정상 QRS 를 탐지하는 모델을 제안해 보았다. 정규화의 경우 StandardScaler 를 사용하여 정규화를 진행하였다.

3) 분류의 정확도를 높이기 위해 시도한 내용

분류의 정확도를 높이기 위해 이상치를 제거해 주었다. 정확한 이상치 제거를 위해 전체 feature 에 대한 correlation 그래프를 (Figure 5) 그린 후 라벨링한 결과와 가장 상관관계가 높은 feature 를 뽑았다. 그 결과 duration의 경우 0.75로 강한 상관관계를 보이기 때문에 duration에 대한 이상치를 제거해 주었다 (Figure 6).

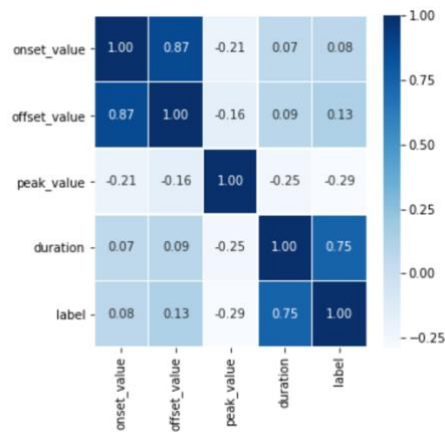


Figure 5

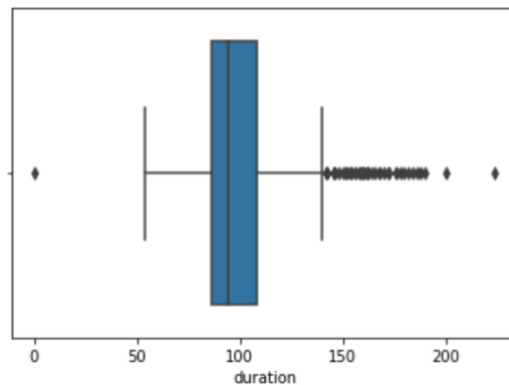


Figure 6

그 결과 1,168 개의 비정상과 605 개의 정상이 남게 되었다.

이때, 1,773 개의 데이터에 대하여 랜덤하게 훈련데이터를 70%를 뽑고 나머지 30%는 테스트 데이터로 모델을 학습시켰다. 정상과 비정상의 경우 데이터 숫자가 맞지 않기 때문에 정확도를 높이기 위해 훈련데이터에서 SMOTE 기법[4]을 사용하여 샘플 수를 맞춰준 후 학습 시킨 다음 테스트 데이터를 갖고 검증을 하였다.

또한 모델의 정확도를 높이기 위해 가장 최고의 성능을 내는 파라미터를 추출하기 위해 GridSearchCV 기법을 사용하여 파라미터를 튜닝하여 추출 하였다. 그 후, SVM, DecisionTree, Logistic Regression 기법 중 가장 성능이 좋은 모델을 선택하였다.

```
from sklearn.tree import DecisionTreeClassifier
model = DecisionTreeClassifier(random_state=42)
grid = {'criterion': ["gini", "entropy"],
        'max_depth': range(1, 10),
        'min_samples_leaf': range(2, 10),
        }
search = GridSearchCV(estimator=model, param_grid=grid, n_jobs=8,
                      cv=5, verbose=1, return_train_score=True)
search.fit(X_train,Y_train)
```

Fitting 5 folds for each of 144 candidates, totalling 720 fits

```
[Parallel(n_jobs=8)]: Using backend LokyBackend with 8 concurrent workers.
[Parallel(n_jobs=8)]: Done 52 tasks      | elapsed:    0.9s
[Parallel(n_jobs=8)]: Done 352 tasks    | elapsed:    4.5s
[Parallel(n_jobs=8)]: Done 720 out of 720 | elapsed:    8.9s finished
```

```
GridSearchCV(cv=5, estimator=DecisionTreeClassifier(random_state=42), n_jobs=8,
             param_grid={'criterion': ['gini', 'entropy'],
                         'max_depth': range(1, 10),
                         'min_samples_leaf': range(2, 10)},
             return_train_score=True, verbose=1)
```

4) 결과의 분석

최종적으로 Decision Tree 의 성능이 제일 높게 나왔으며, 결과는 아래 Figure 7 과 같다. (1:비정상, 0:정상)

	precision	recall	f1-score	support
0	0.4484	0.6420	0.5280	176
1	0.7750	0.6096	0.6824	356
accuracy			0.6203	532
macro avg	0.6117	0.6258	0.6052	532
weighted avg	0.6670	0.6203	0.6313	532

Figure 7

최종적으로 분석한 결과 비정상 환자에 대하여 precision 값은 78.93%의 확률로 계산되었는데, 이는 모델에서 True 로 판단한 경우에 한해서의 정확도로서 $\text{precision} = (\text{실제 비정상 횟수}) / (\text{비정상으로 판단한 횟수})$ 로 계산된다. Recall(57.87%)의 경우 비정상 케이스에 대하여 얼마나 많이 비정상으로 예측 했는지에 대한 정확도로서 계산이 된다. 그리고 Accuracy(61.47%)의

경우 모델이 올바르게 판단한 정도로서 $accuracy = (\text{올바르게 판단 횟수}) / (\text{판단 총 횟수})$ 로 계산이 된다.

추후 이 모델을 보완하고자 한다면 다른 feature 를 추가하여 정상을 질병이라고 진단하는 비율이 조금 있다 하더라도 될 수 있으면 질병이 있는 사람을 보다 잘 진단하기 위하여 Precision 이 낮아지더라도 Recall 를 높여야 할 것이다. 왜냐하면 심각한 질병의 경우 최대한 환자를 잘 찾기 위한 과정이 중요하기 때문이다.

Reference

- [1] Kim, June Soo. "QRS complex." *International Journal of Arrhythmia* 18.2 (2017): 96-99.
- [2] Moskalenko, Viktor, Nikolai Zolotykh, and Grigory Osipov. "Deep Learning for ECG Segmentation." *International Conference on Neuroinformatics*. Springer, Cham, 2019.
- [3] Kashani, Amir, and S. Serge Barold. "Significance of QRS complex duration in patients with heart failure." *Journal of the American College of Cardiology* 46.12 (2005): 2183-2192
- [4] Chawla, Nitesh V., et al. "SMOTE: synthetic minority over-sampling technique." *Journal of artificial intelligence research* 16 (2002): 321-357.