기계학습기초

Titanic – Machine Learning from Disaster

홍민재(201804126)

# 1. 요약 Abstract

# 기계학습기초 수업의 과제로 타이타닉 생존자 예측을 하게 되었다. 본 연구는 타이타닉 탑승자들이 생존하는 데 있어 어떤 변수들이 영향을 미쳤는지 분석하고, 생존자를 예측하는 데 있어서 (수업 내용 중) 최적의 데이터 전처리와 분류 모델을 제안한다. 데이터 전처리는 생존자 예측을 위해 주어진 변수 Pclass, Name, Sex, Age, SibSp(siblings and spouse), Parch(parent and children), Ticket, Fare, Cabin, Embarked 중 Ticket, Cabin을 제외하고 모두 생존에 영향을 미치는 주요 변수로 판단하였고, 분류 모델은 RandomForestClassifier와 DecisionTreeClassifier를 사용하였다. 또한 각 모델의 하이퍼파라미터를 조정함으로써, 정확도를 높일 수 있는 것을 확인하였다.

# 2. 서론 Introduction

본 연구는 캐글에서 제공하는 데이터 세트를 이용한다. 3장에서는 데이터 특징을 파악하고 활용하여 전처리하는 방법을 설명하고, 머신러닝 기법을 이용하여 생존을 예측하는 방법에 관해 설명한다. 4장에서는 결과를 서술하고, 마지막으로 5장에서 결론과 향후 연구과제를 서술하고자 한다.

# 3. 방법론 Methods

3.1 데이터 분석

캐글에서 Train, Test 데이터를 받아 데이터에 대한 정보를 확인한 결과 Train 데이터 변수로 Passengerld(승객의 고유 번호), Survived(생존 여부), Pclass(객실 등급-승객의 사회적, 경제적 지위), Name(이름), Sex(성별), Age(나이), SibSp(동반한 형제자매와 배우자의 수), Parch(동반한 부모 자식의 수), Ticket(티켓의 고유넘버), Fare(티켓의 요금), Cabin(객실 번호), Embarked(승선한 항)이 있고 총 데이터는 891개가 있는데 Name(이름)은 891개중 714개만 채워져있고 Cabin(객실 번호)는 891개 중 204개만 채워져 있고 Embarked(승선한 항)은 891개 중 889개만 채워져있고 나머지는 특성들은 891개 중 891개의 데이터가 채워져 있었다. 그림[3-1]을 통해 확인할 수 있다. Test 데이터 특성으로는 Train 데이터의 Survived(생존 여부)를 제외하고 모두 존재했고 Age(나이)는 418개 중에 332개만 채워져 있고 Fare(티켓의 요금)은 418개 중에 417개만 채워져 있고 Cabin(객실 번호)은 418개 중에 91개만 채워져 있고 나머지 특성들은 418개 중에 418개의 데이터가 채워져 있었다. 그림[3-2]을 통해 확인할 수 있다.

데이터 변수의 타입으로는 Passengerld(승객의 고유 번호)는 int 타입, Pclass(객실 등급-승객의 사회적, 경제적 지위)는 int 타입, Name(이름)는 object 타입, Sex(성별)은 object 타입, Age(나이)는 float 타입, SibSp(동반한 형제자매와 배우자의 수)는 int 타입, Parch(동반한 부모 자식의 수)는 int타입, Ticket(티켓의 고유넘버) object 타입, Fare(티켓의 요금) float 타입, Cabin(객실 번호) object 타입, Embarked(승선한 항)는 object 타입으로 되어있었다. 그림[3-1]과 그림[3-2] 통해 확인할 수 있다.

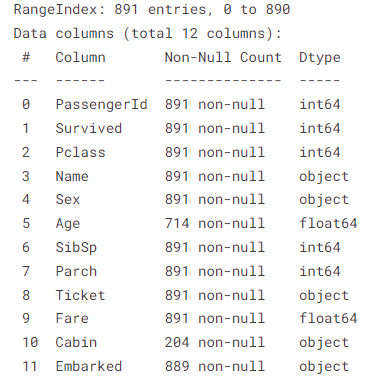
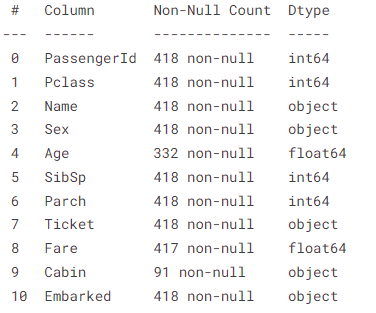


그림 [3-2]

그림 [3-1]

Train 데이터의 생존 확률로는 38.4%로 많이 저조한 편이었다. 그림 [3-3] 통해 확인할 수 있다.

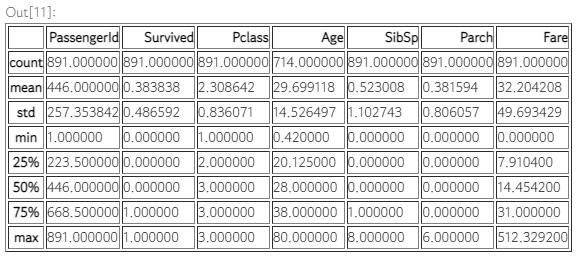
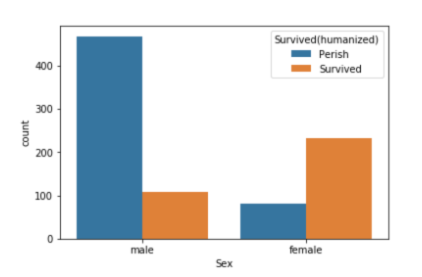


그림 [3-3]

Train 데이터의 남녀 비율은 남성이 577명 여성이 314명으로 나타났다 이에 따른 생존율 비교로는 여성이 74%로 생존하였고 남성이 18%로 생존하였다 그림 [3-4] 통해 알 수 있다.

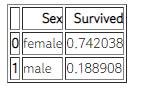
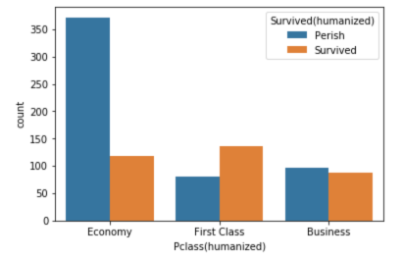
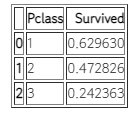


그림 [3-4]

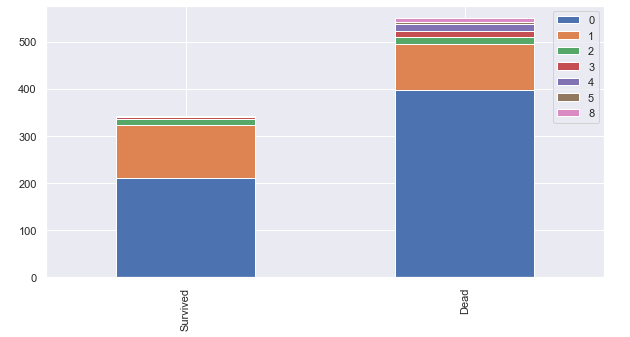
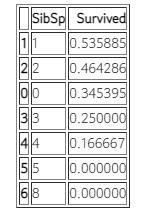
Train 데이터의 Pclass(객실 등급)에 따른 생존율 비교로는 Pclass중 1등급(first class)이 62%로 생존하였고 2등급(business)이 47%로 생존하였고 3등급(economy)이 24%로 등급이 좋을수록 생존율이 높았다 그림[3-5] 통해 알 수 있다.



****

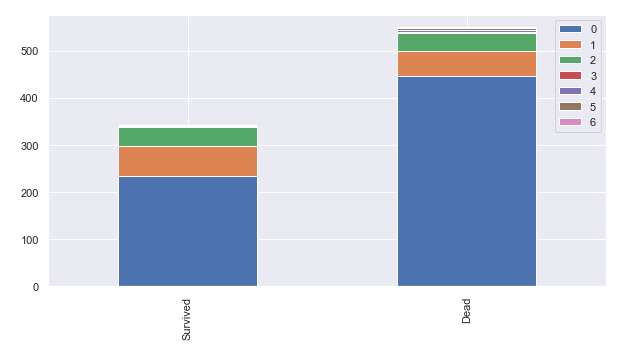
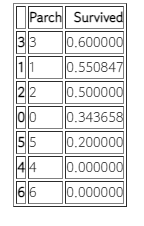
**그림[3-5]**

Train 데이터의 SibSp(동반한 형제자매와 배우자 수)에 따른 생존율 비교로는 SibSp=0인 경우 34%로 생존하였고 SibSp=1인 경우 53%로 생존하였고 SibSp=2인 경우 46%로 생존하였고 SibSp=3인 경우 25%로 생존하였고 SibSp=4인 경우 16%로 생존하였고 SibSp=5인 경우와 SibSp=8인 경우 0%로 생존하였다. SibSp가 1 그리고 2를 제외하고는 숫자가 높을수록 생존 확률이 줄어들었다. 그림[3-6] 통해 알 수 있다.

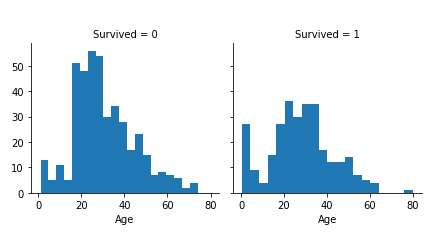


그림[3-6]

Train 데이터의 Parch(동반한 부모 자식 수)에 따른 생존율 비교로는 Parch=0인 경우 34%로 생존하였고 Parch=1인 경우 55%로 생존하였고 Parch=2인 경우 50%로 생존하였고 Parch=3인 경우 60%로 생존하였고 Parch=4인 경우와 Parch=6인 경우 0%로 생존하였고 Parch=5인 경우 20%로 생존하였다. Parch가 1 그리고 2 그리고 3을 제외하고는 동행이 많을수록 생존 확률이 낮았다. 그림[3-7] 통해 알 수 있다.



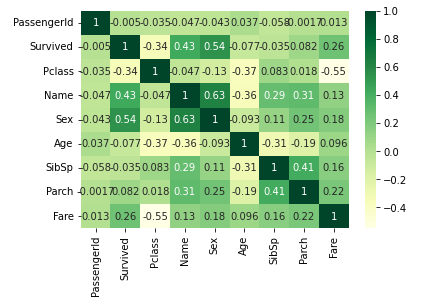
**그림[3-7]**



Train 데이터의 Age(나이)에 따른 생존율 비교로는 4세 이하 유아의 생존율이 높았고 15~25세 승객들의 생존율이 높았다. 그림[3-8] 통해 알 수 있다. 나이에 관해 다른 변수들을 연관 지어서 생존 여부를 조사할 필요가 있었다.

그림[3-8]

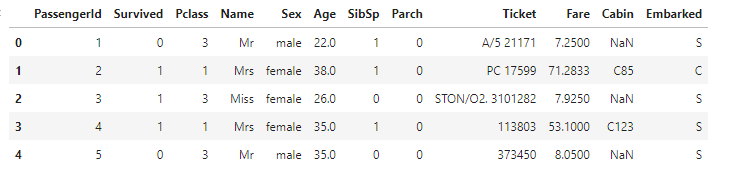
Train 데이터의 상관분석을 통해 Age(나이)에 대한 상관관계를 알 수 있었다. 그림[3-9] 통해 Age(나이), Name(이름), Pclass(객실 등급), SibSp(동반한 형제자매와 배우자의 수), Parch(동반한 부모 자식) 간에 음의 상관관계가 있음을 알 수 있었다.



**그림[3-9]**

3.2 데이터 전처리

Train, Test 데이터의 Name(이름)이 풀네임으로 적혀 있어 이니셜로 대체하였다. 그림[3-10] 통해 알 수 있다.

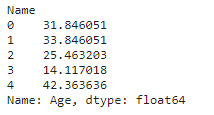


**그림[3-10]**

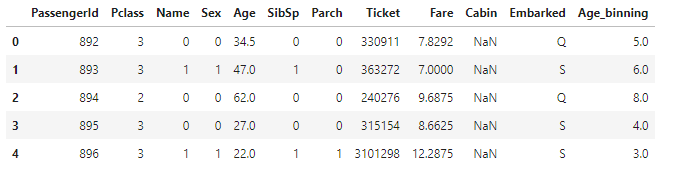
Name(이름)에 들어간 이니셜을 확인한 결과 이니셜이 너무 많아 Miss, Mr, Mrs, Ms, Other로 대체하였다 그리고 학습을 위해 이니셜을 Mr=0, Mrs=1, Miss=2, Master=3, Other=4로 숫자로 바꾸었다.

Train, Test 데이터의 Sex(성별)이 female과 male로 적혀있어 남성은 0으로 여성은 1로 변경하였다 그림[3-12] 통해 알 수 있다.

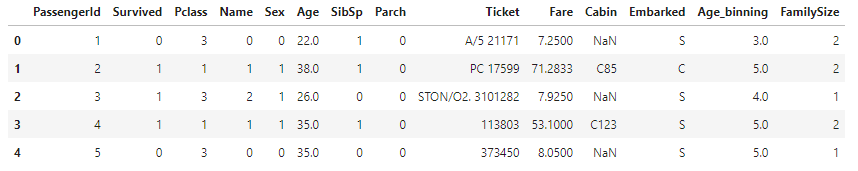
Train, Test 데이터의 Age(나이)와 가장 상관관계가 큰 Name(이름)으로 Age(나이)의 평균값을 구하였다. Pclass(객실 등급)이 상관관계가 가장 컸지만, 등급이 3개뿐이라 체계적으로 나누기가 어려워 Name(이름)에 따라 나누게 되었다. 그림[3-11] 통해 알 수 있다.

그림[3-8]를 통해 20~30대가 가장 생존율이 높은 걸 알 수 있는데 위 그림을 바탕으로 구간을 설정하였다. 나이가 10살 이하는 0그룹에 나이가 10~15는 1그룹에 나이가 15~20은 2그룹에 나이가 20~25느 3그룹에 나이가 25~30은 4그룹에 나이가 30~40은 5그룹에 나이가 40~50은 6그룹에 나이가 50~60은 7그룹에 나이가 60 이상은 8그룹으로 설정하였다. 그림[3-12]을 통해 'Age\_binning'이라는 변수를 만들어 처리한 걸 확인할 수 있다.

그림[3-11]

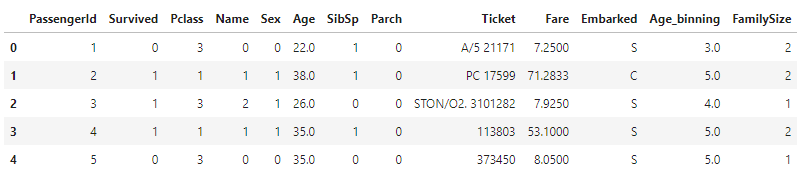
  **그림[3-12]**

Train, Test 데이터의 'FamilySize'라는 변수를 만들어 SibSp(동반한 형제자매와 배우자 수)와 Parch(동반한 부모 자식)를 합치고 본인을 더하여 넣어주었다. 그림[3-13] 통해 알 수 있다.

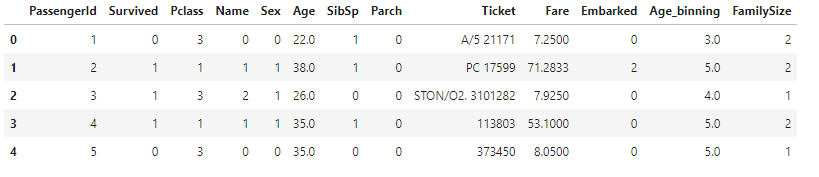
**그림[3-13]**

Train, Test 데이터의 Cabin(객실 번호)는 결측값이 너무 많아 데이터의 해당 변수를 제거하였다. 그림[3-14] 통해 알 수 있다.

그림 [3-11]

**그림[3-14]**

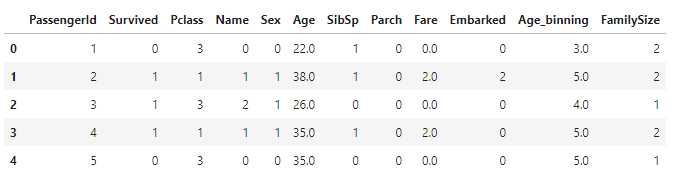
Train, Test 데이터의 Embarked(승선한 항)은 결측값을 채우기 위해 다른 변수들에 대한 상관관계를 확인하였지만, 결측값이 상당히 적고 SibSp(동반한 형제자매와 배우자 수)와 Parch(동반한 부모 자식 수)도 없기 때문에 추측하기 힘들어 빈도수가 제일 높은 'S'로 채워 넣었다. Embarked(승선한 항)은 문자열로 되어있어 'S'=0로 'Q'=1로 'C'=2로 숫자로 변환하였다. 그림[3-15] 통해 알 수 있다.



**그림[3-15]**

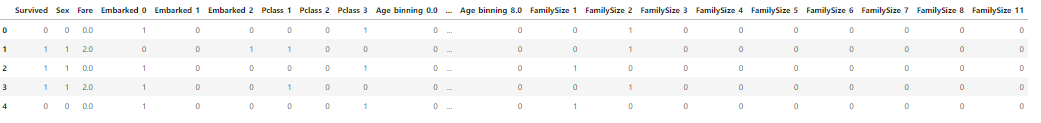
Train, Test 데이터의 Ticket(티켓의 고유넘버)는 생존 여부와 상관이 없다고 판단하여 데이터의 해당 변수를 제거하였다.

Train, Test 데이터의 Fare(티켓의 요금)은 결측값이 하나밖에 없으므로 중윗값으로 대체하고 변숫값들에 대해 Fare이 10 이하인 경우는 0으로 10~30인 경우는 1로 30~100인 경우 2로 100 이상은 3으로 구간화를 하였다. 그림[3-16] 통해 알 수 있다.



**그림[3-16]**

Train, Test 데이터의 변수 중에 범주화 데이터가 아닌 경우가 있어 Embarked(승선한 항), Pclass(객실 등급), Age\_binning, FamilySize(형제 자매 + 배우자 수 + 부모 자식 + 본인)을 One-Hot Encoding을 진행하였다. One-Hot Encoding이란 모델 학습을 할 때 데이터에 연속성이 없다는 것을 알려주기 위한 과정이다. 이를 통해 연속성이 없다는 것을 확실히 하여 알고리즘의 오류를 줄일 수 있다. 그림[3-17] 통해 알 수 있다.



**그림[3-17]**

3.3 Machine Learning

3.3.1 Decision Tree

데이터의 전처리 된 것을 확인해보면 특성들이 상당히 많은 걸 알 수 있다.이러한 특성들을 하나의 가중치 값들로 나타내기보다는 데이터의 특성을 세세히 나누어 분류 정확도를 높이는 게 현명한 판단이라고 생각되어서 결정트리를 결정하게 되었다. Decision Tree란 의사결정트리 또는 의사결정나무라고 하고 분류와 회귀 모두 가능한 지도 학습 모델 중 하나이다. 특정 기준에 따라 데이터를 구분하는 모델로 한 번의 분기 때마다 변수 영역을 두 개로 구분하여 학습하는 게 특징이다. 학습의 정확도를 올리기 위해서 하이퍼파라미터를 조정하였는데 max\_depth와 class\_weight를 조절하여 정확도를 측정하였는데 max\_depth=9와 class\_weight={0:5,1:1}인 경우가 학습한 것 중에 제일 높게 나왔다. max\_depth이란 트리의 최대 깊이를 지정하여 오버피팅을 막고 학습 시간을 조절할 수 있다. class\_weight란 데이터에서 클래스의 불균형 문제가 존재하는데 각 클래스가 갖고 있는 데이터의 양이 차이가 큰 경우 클래스 불균형이라고 부른다 이러한 경우 학습이 안 되는 경우가 발생하다 이를 위하여 원하는 타겟에 대해서 못 맞출 경우 가중치를 부여하여 좀 더 정확도를 올릴 수 있다.

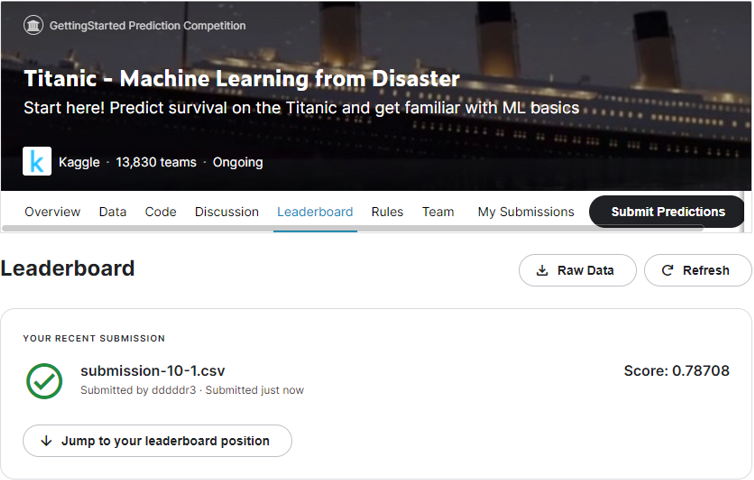
3.3.2 RandomForest

데이터의 특성들이 정형 데이터로 이루어져 있어서 정형 데이터를 다루는데 가장 뛰어난 성과를 지내는 알고리즘으로 앙상블 학습 즉 랜덤포레스트를 결정하였다. 랜덤포레스트란 앙상블 학습의 대표 주자 중 하나로 안정적인 성능 덕분에 널리 사용되고 있는 모델 중 하나이다. 랜덤 포레스트는 결정 트리를 랜덤하게 만들어 결정 트리의 숲을 만들고 각 결정 트리의 예측을 사용해 최종 예측을 만드는 모델이다. 학습의 정확도를 올리기 위해서 하이퍼파라미터를 조정하였는데 class\_weight를 조절하여 정확도를 측정하였는데 class\_weight={0:14,1:1}인 경우가 학습한 것 중에 제일 높게 나왔다.

# 4. 결과 Results

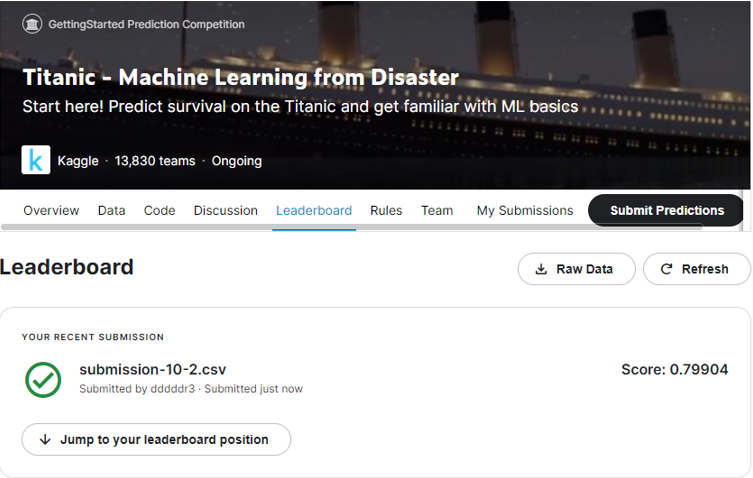
위 Method를 사용하여 나온 결과를 객관적으로 작성하였고 그림, 표 등을 적극 이용하여 정량적, 정성적 결과를 작성했다. 본 연구에서는 3.2에서 제시한 데이터 전처리 방식을 바탕으로 3.3에서 제시한 머신러닝 기법을 통해 나온 캐글의 submit 결과를 확인하였다. 그림[4-1], 그림[4-2] 통해 알 수 있다.

DecisionTreeRandomForest



**그림[4-1]**

RandomForest



**그림[4-2]**

# 5. 결론 Conclusion

본 연구는 캐글의 타이타닉 데이터셋을 바탕으로 데이터를 전처리하고, 전처리한 데이터를 바탕으로 수업 중 배운 머신러닝 기법을 통해 생존자를 예측하여 캐글에서 높은 점수를 받는 데 목적을 두었다.

Random Forest의 submit 최고점은 0.79904로, 다른 머신러닝 기법들(Decision Tree : 0.78708, K Nearest Neighbor : 0.77 등)에 비해 가장 좋은 성능을 보이는 것으로 나타났다.

향후 과제로는 더 알맞은 데이터 전처리 방법을 고안하고, 더 적합한 머신러닝 기법을 배워 더 높은 점수가 나올 수 있도록 할 것이다.