빅데이터 입문
Term Project

체스 rating과 흑백 승리 여부

목차

- 요약
- 목표
- 데이터
- 실험
- 향후 과제

요약

- 체스의 실력을 나타내는 척도 중 하나인 ELO rating(게임에서 사용되는 MMR과 유사) 등을 이용해 흑, 백 중 승자를 예측
- 데이터는 kaggle의 5-million-chess-game-results-november-2019를 사용
- 데이터 패턴을 학습할 모델은 딥러닝이 아닌 기계학습을 사용

목표

- 체스는 실력이 높을수록 ELO rating이 높게 설정된다.
- 그렇다면 먼저두는 백색 말 플레이어의 ELO rating이 흑색 말 플레이어보다 높다면 백색 플레이어가 이길 것이라고 예측할 수 있다.
- 하지만 만약 흑색 플레이어의 ELO rating이 더 높다면 어느 정도일 때 승리가 보장될까
- 또 ELO rating 이외에 승패에 영향을 미치는 요인은 무엇이 있을까
- 위 궁금증을 ML을 이용해 확인

데이터

- 데이터는 kaggle의 5-million-chess-game-results-november-2019를 사용 https://www.kaggle.com/timhanewich/5-million-chess-game-results-november-2019
- 해당 데이터는 체스 게임 이름, 날짜, 리플레이 사이트, 결과, 흑백 플레이어의 ELO rating, 진행 라운드, 플레이어의 이름 등의 500만개의 데이터가 있다.
- 이중 관심있는 데이터는 ELO rating, 결과, 흑백 플레이어의 ELO rating 차이, ECO(체스의 오프닝)이 있다.
- 데이터는 1건의 누락데이터(missing)를 제외하고는 맞지 않거나(mismatched) 누락된(missing)데이터는 없다.(kaggle이 제공하는 데이터 통계)

데이터 전처리

- 프로젝트는 ELO rating과 흑백 승패 관계이지만, 실제로는 그 외에도 승패에 영향을 미치는 요소가 있을 수 있기 때문에 다른 데이터도 포함해 사용할 수 있다.
- 데이터의 colum의 개수는 200개를 넘지만 대부분이 턴별 기보로 이루어졌다.
- 따라서 기보, 리플레이사이트, 플레이어 이름, 날짜 등은 제외하였다.
- 레이블로 사용될 결과의 경우 백 승리, 흑 승리, 무승부가 각각 '1-0', '0-1', '1/2-1/2'로 표기되어 있기 때문에 사용할 언어, 모듈 등에 따라 원핫인코딩 등의 전처리가 필요할 수 있다.

언어 및 모델 선택

- 기존에는 python의 scikit-learn을 사용할 예정이었지만 matlab의 Statistics and Machine Learning Toolbox 등을 사용하였다.
- 사용된 모델은 svm과 랜덤 포레스트 모델을 사용하였다.
- 승패를 맞추는 문제이기 때문에 각 모델은 분류 모델로 사용하였다.

실험

- 최초에는 500만개의 데이터, 불필요 특징을 제외하고 남은 5개의 특징(흑백 ELO, 흑백 ELO차이, ECO)을 가지고 svm 학습 <- 학습에 너무 오랜 시간이 소요됨
- 500만개 중 1%(5만개)를 복원추출 후 svm 학습 <- 마찬가지로 학습에 너무 오랜 시간이 소요됨, 아마 특징의 개수가 지나치게 많은 것으로 예상
- 데이터 5만개, 특징 2개(흑백 ELO)만 가지고 svm 학습 <- 이 또한 학습에 너무 오랜 시간이 소요됨, 아마 svm은 여러 클래스에 대해 다중분류할 수 없기 때문에 일대일 전략으로 학습하다보니(matlab default) 각각에 대해 학습하는데 오래걸리는 것으로 예상됨. 즉 데이터가 너무 많음
- 500만개 중 0.2%(1만개), 특징 2개로 svm 학습

실험

- 모델의 검증은 다시 500만개의 데이터 중 2000개의 데이터를 복원추출해 test set 구성
- 훈련 데이터를 포함하는 500만개의 데이터를 복원추출했기 때문에 위험할 수 있지만 모집단이 워낙 커 무관할것으로 생각
- 추출한 test set으로 정확도(올바르게 예측한 비율) 계산

```
svm_result =

    0.5125
```

- 정확도가 0.5125로 지나치게 낮게 나옴 <- 모델의 문제인가?

실험

- 모델을 랜덤 포레스트 모델로 바꿔 실험

```
RF_result =

    0.5555
```

- svm보다 낫지만 여전히 지나치게 낮은 정확도
- 훈련데이터와 검증데이터가 서로 겹치지 않도록 검증데이터를 추출해도 여전히 정확도가 낮게 나옴

1-0	50%
0-1	46%
Other (191974)	4%

향후 과제

- i) 익숙하지 않은 matlab을 사용해 코드에 문제가 있을 수 있음 <- python을 이용해 코드를 작성해 재실험
- ii) 실제 ELO rating 단독으로는 승패에 영향이 적을 수 있음_실제 경기는 ELO rating이 크게 차이나지 않도록 경기를 치룸 <- 다른 특징을 추가
- iii) 모델이 단순해 해당 특징에 나타나는 패턴을 잡아낼 수 없음 <- 더욱 정교한 모델(DNN) 등을 사용
- iv) 해당 데이터에 있는 특징들로는 label과 연관된 패턴이 없음 <- 시각화를 통해 데이터 형태를 확인
- 최종적으로는 500만개의 데이터를 빅데이터 처리 기법을 사용해 처리하는 것이 목적
- Matlab 코드 깃허브 : https://github.com/mental-disaster/chess

참고자료

- 최대진 교수님 빅데이터 입문 수업
- Kaggle 5 Millon Chess Game Result – November 2019
  https://www.kaggle.com/timhanewich/5-million-chess-game-results-november-2019
- 매트랩 서포트 벡터 머신 분류
  https://kr.mathworks.com/help/stats/support-vector-machine-classification.html?s_tid=CRUX_lftnav
- 매트랩 분류 앙상블
  https://kr.mathworks.com/help/stats/classification-ensembles.html?s_tid=CRUX_lftnav