기계학습 기반 주가 예측모델

GUI 개발 도구

이진재, 유승호

주식 투자자들은 지속적으로 높은 수익을 얻기를 원한다. 높은 수익을 얻기 위해서 투자자들은 저평가된 종목을 찾거나 (기본적 분석), 특정 chart의 pattern등을 분석하기도 한다. (기술적 분석)

경제지표 또는 개별종목 데이터를 이용하여 미래에 주가가 상승할 것인지 하락할 것인지 기계학습을 통하여 알아낸다. 학습된 데이터를 바탕으로 어떻게 투자를 해야 하는지 사용자에게 의사결정을 지원한다.

Graphic user interface(GUI)

**Gathering**

**Marking**

**Modeling**

**Order create**

**Order generate**

**Visualization**

Scikit-learn, xgboost

zipline

matplotlib

Local DB

**Module**

-Gathering : Fredapi 접속 후 경제지표를 local에 저장

-Marking : 기계학습을 위한 데이터 분류 작업(True or False)

-Modeling : 우수한 모델 탐색

-Order create : 가상매매, 실제매매를 위한 주문서 생성

-Order generate : 매매 시뮬레이션

-Visualization : 시뮬레이션 결과 시각화

**Gathering independent module**

Fredapi를 사용하여 Fred로부터 경제지표를 획득한다. Fred(연방준비제도이사회)는 FRED의 연구부문에서 관리하는 데이터베이스이다. 87개 출처로부터 50만개 이상의 경제지표 데이터를 가지고 있다. 경제지표 데이터는 월간 분석에 사용된다.

**[입력]**

-Fred Api Key : fred로부터 발급받은 개인 api key # 현재는 key 지정 안되있음

-start date : 받아올 경제지표들의 시작 날짜 # 현재는 시작날짜 2000-01-01

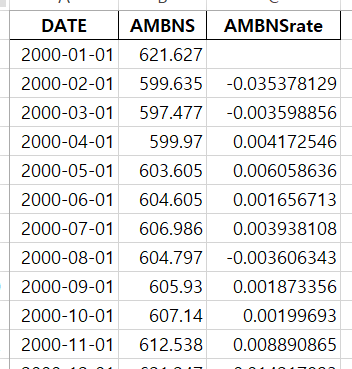
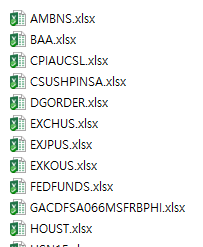
-path : 경제지표들을 저장할 폴더 경로

-column list : 받아올 경제지표들의 목록 #현재는 list를 안받고 30개 전부 받음

**[출력]**

n개의 경제지표가 csv로 저장되어 정해진 경로에 생성됨.

경제지표 별 파일이름은 fred에서 지정한 약자로 되어있다.

[그림 1-1] St. Louis Adjusted Monetary Base(왼), 경제지표들이 들어있는 폴더(오)

Column1 : 날짜(월간)

Column2: 지표

Column3 : 지표 등락률

**[필요한 라이브러리]**

-os

-sys

-datetime

-dateutil

-pandas

-fredapi

Gathering dependent module

Yahoo finance로부터 지수데이터 또는 종목데이터를 받아온다.

[입력]

-종목코드 or 지수이름

-시작날짜, 끝날짜

-저장경로

[출력]

-정해진 경로에 csv파일로 저장됨

[필요한 라이브러리]

**Marking module**

사용자로부터 종목데이터 또는 지수데이터를 입력받아 조건에 맞는 column(label)을 생성한다.(전략을 생성한다) 월간 데이터와 일간 데이터를 선택할 수 있다. 월간 전략과 일간 전략이 구분되어 있다.

**[입력] 월간**

-column option 1 : ???

-column option 1 index :

-column option 2 : ???

-column option 2 index

-column 이름 : 생성될 column 이름

-up/down : 상승인지 하락인지?

-rate : 상승률/하락률

[입력] 일간

-???

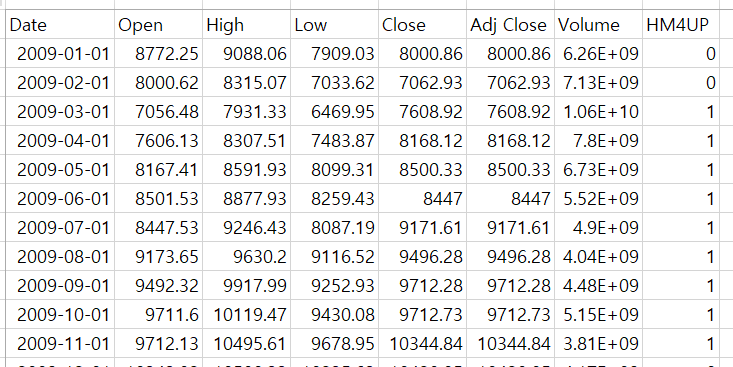
-???

-???

**[출력]**

**입력 옵션에 맞는 column 데이터가 추가된 csv 파일.**

**어떤 옵션으로 생성했는지 marking option log file**



[그림 2-1] 학습데이터 column(HM4UP)이 추가된 csv file

Date : 날짜

Open : 시가

High : 고가

Low : 저가

Close : 종가

Adj Close : 수정 종가

**[필요한 라이브러리]**

-pandas

-numpy

**Modeling Module**

그림 2-1과 같이 학습데이터 column이 추가된 데이터를 가지고 기계학습을 실행한다. 사용되는 feature는 경제지표 또는 다른 데이터가 될 수 있을 것이다. 두 테이블을 join했을 때, 행의 손실은 일어나지 않아야 한다. 시계열데이터는 시간의 흐름도 학습되기 때문에 중간중간에 데이터가 누락되서는 안된다.

[사용되는 기계학습 알고리즘]

-random forest

-knn

-xgboost

-multiple regression

**[입력]**

-algorithm : 사용할 알고리즘

-algorithm option : 알고리즘 옵션

-start date : 시작 날짜

-end date : 끝 날짜

-test size : 테스트 데이터 크기

**[출력]**

여러 알고리즘에 대한 테스트 결과가 들어있는 csv

특정 알고리즘의 예측 값이 들어있는 csv

**[필요한 라이브러리]**

-numpy

-scikit learn

-os

-statsmodels

-xgboost

-warnings

-datetime

-pandas

-itertools

**Create Order module**

특정 모델이 기계학습을 통해서 만들어 낸 예측데이터를 가지고 사용자로부터 어떤 전략으로 매매를 할 것인지 조건들을 입력받는다.

**[입력] 월간**

- 전략 옵션이름

- yahoo 종목 or 지수 코드 이름

- column 이름  
- csv file path : 예측값 column이 추가되어있는 csv 파일 경로

- rate

- 저장 파일 이름

**[입력] 일간**

- .py (파이썬 파일) : 일련의 규칙으로 작성된 조건들이 들어있는 파이썬 파일. 사용자가 작성

- csv file : 예측값 column이 추가되어있는 csv 파일

- column 이름

**[출력]**

-order file : backtesting 환경에서 입력을 받을 수 있는 csv 파일. 종목별 buy와 sell을 언제할 것인지 기술되어 있다. Buy, sell 컬럼이 존재한다.

**[필요한 라이브러리]**

-???

-???

-???

**Generate Order module**

Order create에서 만들어진 주문서를 가지고 가상 매매를 시행한 후, 수익률을 도출한다. 주문서에는 시점별로 buy인지 sell인지 각 column에 표시가 되어있다. Backtesting에 사용되는 파이썬 라이브러리에는 zipline, ta lib, backtrader 등 다양한 라이브러리가 존재한다. 이 모듈에서는 zipline을 사용한다.

[입력]

- order file : 주문서 파일

- 초기자본

- 거래 수수료

- ???

- 날짜

[출력]

-portfolio : 가상 매매 거래내역이 담긴 log 파일

-그래프 : 여러가지 정보로 그래프가 그려져있음

[필요한 라이브러리]

-???

-???

-???

**Visualization**

그래프의 종류

-가상매매 거래내역의 자산변동흐름

-이동평균, 볼린저밴드 등 기술적분석의 지표 그래프

-경제지표 그래프

[그래프 종류]

-캔들차트

-???

-???

**[입력]**

-csv file : 특정 데이터가 들어있는 csv 파일

**[출력]**

-graph : 그래프

**[필요한 라이브러리]**

-matplotlib