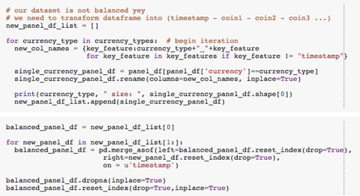
4주차 강의

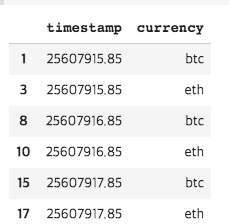
# 1. [과제] Limit Orderbook Data에 기반하여 FNN, CNN, RNN model을 통한 예측 모델을 작성해 봅시다 (Day 1)

## Question: balanced panel을 만드는 부분이 잘 이해가 되지 않았습니다.



이 부분의 코드에 대해 질문을 주셨는데요

저희의 orderbook 데이터 모양을 보시면 currecy-time level의 데이터임을 보실 수 있습니다. 저희가 분석에서 사용하려면, 해당 데이터를 currency-time level이 아니라, time level 만으로 바꿔줄 필요가 있고요



예를 들어서 위 그림에서 보시면 동일한 timestamp 2개가 연달아 있지만, currency가 다른 2개의 데이터 있음을 보실수 있습니다

이와 같이 같은 timestamp 이더라도, currency가 다르면 다른 row에 있는 모양으로 되어 있고요 (edited)

하지만 regression, deep learning 등의 분석을 할때는 추정하고자 하는 모델이 주로 Yt = Xt + ... 의 형태로 같은 timestamp의 시점에 대한 분석을 하고 있습니다. 따라서 timestamp가 같으면 모두 가로로 와줘야하는데 지금은 그렇지 않은 상황입니다. 위 코드를 통해서 변형해준 결과는



같은 timestamp에 대해서는, 모든 currency가 같은 row에 오도록 변형해 주었습니다

## Question. day1에 hands on lab에서 Serieal Correlation Test에서 컬럼명을 t-1과 t+1이던데,,, t-1과 t시점의 상관관계가 아닌 t-1과 t+1의 상관관계를 확인하는건가요? 그리고 t-1시점과 t+1시점의 상관관계 결과가 -0.079179인데, t시점에 비해 t+1시점에 -0.079179이므로 상관관계가 0에 가까우므로 상관관계가 없다고 판단하면 되는거같은데...단순히 이 상관관계를 파악하는 것이 Serieal Correlation Test의 목적인가요?? 이 실험에서는 상관관계가 0에 가까운데 만약에 1에 가깝다거나 -1에 가깝다면 앞으로의 실험방향에서 다르게 설정하거나 추가로 해야할게 있을까요??

serial correlation에 대해서 질문을 주셨는데요, 해당 column은 t-1 t가 맞습니다 ㅠ

그리고, 해당 값을 테스트하는 목적은, 과연 과거의 값이, 미래를 얼마나 설명해 주는지를 테스트하여 과거의 몇번쨰 term 까지를 끌고 와야 할지에 대해 조사해 보기 위한 목적이고요

예를 들어서, 가장 간단한 모형을 고려해보시면 Xt= Xt-1 + Xt-2 ... Xt-N 의 몇 시점까지를 고려해야 할지 테스트 하는 용도로 쓸수 있습니다

## Question. bin이 quantity와 price를 같이 보기 위해 수정한 변수라는건 알겠는데, 이게 왜 그런지 이해가 안갑니다... bin에 대한 개념과 ask\_strength, bin\_strength개념 추가 설명 부탁드립니다.

지금의 bin에 따른 변수들은 CS230 Project 리포트를 모사하기 위해 동일한 방식으로 만들어져 있는데요, Current Price를 가장 큰 bid와 가장 작은 ask 값의 average 값으로 보고 해당 값 (현재 가격)의 10%~ -10%까지 price 범위 내에서, 0.1% 크기의 bin 200개를 만들어 주었습니다

저희가 fundamental하게 생각해 볼수 있는것이 bid (구매 요청)가 많으면 수요가 많다는 뜻이니까 가격이 오른다는 의미로 볼수도 있고, ask (판매 요청)가 많으면 가격이 떨어진다고 예측해볼수도 있습니다

또한 가격 범위에 따라서 bid와 ask volume을 본 이유는 현재가와 유사한 가격대에 bid, ask를 준 경우가, 훨씬 먼 가격대에 bid, ask를 준 경우보다 영향력이 클것이라고 예상해 볼수 있기때문에 예를 들어서, 현재가가 100인데 101과 99에 판매 / 구매 요청을 한 경우와, 현재가가 100인데 200에 판다와 1에 산 다는 요청을 한 경우, 다른 영향이 있을 것이라고 보고, 가격 범위에 따라서 volume을 측정해 주었습니다

bin 설정에 대해서 조금 더 얘기드리자면, 여기서 사실 민감한 부분이 bin을 설정함에 따라서 discontinuous 하게 price에 따른 volume이 짤리는데 이 부분은 smoothing을 통해서 해결할 수 있고요, bin size 자체도 중요하지만 지금의 study에서는 0.1%로 설정하였습니다.

bin을 조금 더 구체적인 예제로 보여드리자면, 현재가격 \*0.9/ ... / 현재가격 \*0.998/ 현재가격 \*0.999 / 현재가격 / 현재가격\*1.001 / 현재가격 \*1.002 / ... / 현재가격 \*1.1 이러한 가격 범위에 따라서 해당 범위 내의 bid 혹은 ask volume을 측정해 주었습니다.

# \*2. [과제] Limit Orderbook Data에 기반하여 FNN, CNN, RNN model을 통한 예측 모델을 작성해 봅시다 (Day 1)\*

## Question. 1D CNN의 출력을 LSTM으로 넣을때 어떠한 이점을 예상할수 있는것인가?

CNN을 통해서 Input의 Local Pattern을 찾아주는 역할을 할수 있습니다.

일종의 Feature Extraction의 용도로 CNN을 붙힌 뒤에 RNN을 쓰고 있는 것 이고요

# 3. Stacked Autoencoder는 어떠한 용도로 사용될 수 있나요? (Day 3, 4)

많은 분들이 좋은 답을 주셨는데, stacked autoencoder를 통해서 크게는 1. Denoising 2. feature extraction의 용도로 사용하실 수 있습니다.

stacked autoencoder를 자세히 보시면 더 적은 정보량으로, 원레의 정보를 복구해내는 것을 보실 수 있는데요

이를 통해서, 데이터를 압축하는 과정에서 일반적이지 않은 패턴은 지워진다고 보시면 되고요, 여러 데이터에 걸쳐 유사하게 나타나는 패턴만 살아남는다고 보시면 됩니다.

예를 들어서, input은 200개 인데 마지막 layer의 값은 30~40개 정도이면 200개의 값이 가진 정보를 사실은 훨씬 적은 개수로 표현할 수 있게 됩니다.

저희 마지막 모델 (day4, 5)에서는 stacked autoencoder의 encoded input을 다음 LSTM의 input으로 사용합니다.

denoise & abstaction이 충분히 된 데이터를 사용하려는 목적이고요.

### 1D CNN보다 SAE가 feature extraction관점에서는 더 좋은것인가보네요? (최명환)

최종 Performance를 통해서 Feature extraction 정도를 판단하기에는 어려운 점이 있고요, 해당 부분을 판단하기 위해서는 보다 적절한 evaluation measure들의 개발을 필요로 합니다.

다만 두개의 목적이 조금 다른 부분이 1d cnn을 쓰시면 인접한 데이터에 있는 local pattern에 대한 feature를 찾는데 이점이 있으며 stacked autoencoder는 local pattern을 걸러내는 역할을 하기 보다는 여러 데이터에 걸쳐서 유사한 패턴만을 남기고, correlation이 강한 데이터 간의 pattern만 남기고 있습니다

같은 feature extraction이긴 하지만, 그 작동 원리가 조금 다른것 같습니다

### Day5에서 고민해본건데. SAE 하고 1D CNN하고 더 feature extraction에 도움이 되는건가요? 아님 too much인건지. 오히려 (최명환)

네 맞습니다. 두개의 feature extraction의 결과를 concat layer를 추가한 뒤에 lstm에 넘겨주시는 것도 좋은 방법입니다!

### 순서는 상관없을까요? 1D CNN과 SAE 혹은 SAE 다음 1D CNN.. 이건 테스트 해봐야알겠죠? :;; (최명환)

Serial하게 넣어주시는 방법 이외에도 (순서 상관이 있는 방법) 병렬적으로 처리해 주실수 있습ㄴ이다 니다

input -> 1d cnn 의 결과와, input -> sae의 결과를, concat 한뒤에 lstm에 넣어주시는 편이 좋을것도 같습니다

그런데 조금 다른 얘기긴 하지만, 개인적으로는 같은 Performance면 간단한 모델이 더 좋은 모델인 것 같습니다. 따라서 꼭 복잡한 모델이 test set에서 더 좋은 performance를 보이는 것은 아닌것 같고요 (테스트가 필요하지만)

# 4. [과제] 가장 간단한 거래 전략에 기반하여, 직접 작성한 Model의 Profit을 공유해 주세요. (100을 투자하여, 상승 예측 시 구매, 하락 예측시 판매 했을 때 테스트 기간 동안의 수익) (Day 5)

성과가 가장 많이 나오신 분이 189.3 까지 나오셨다고 하네요 ㅎㅎ 다른 분들은 혹시 어디까지 나오셨나요~?

저는 참고로 이번 수업에 사용한 모델 중에서 어떤(?!) 모델은 consistent하게 200 이상이 나오는게 있었습니다. tip을 드리자면, 꼭 복잡한 모델이 더 좋은 성능을 내지는 않고요 제가 과제로 내드린 모델을 보다 간단하게 만들어 보시면 더 좋은 모델도 찾으실 수 있을 것 같습니다.

해당 일자에 나온 코드 질문 중에서

### Question. 코드에서 Wavelets Denoising 실행하는 위치가 가장 좋은 위치는 어디인가요? train X를 다 만들고 나서? 아님 data 받고 preprocessing 쯤... 저번주 Wavelet Denoising 실습했는데 그때와, day5에서 실행하는 위치는 약간 다름.

저희가 이전에 다룬 모델은 Price를 denoise하면 모양이 어떻게 달라질지 보여드린 것 이고요. 이번에 wavelets은 price 쪽 (outcome)이 아니라 inputs에 사용해서 input 데이터를 denoising하는데 사용하였습니다.

### Question. LSTM대신 CNN-RNN으로도 학습할수 있다고 day4 때 공유된 문서에 되어있는데, encoder 출력시 shape이 (N,60,20) 인데 이걸 CNN으로 넣을때 2D CNN으로는 못넣는거죠? input shape을 어떻게 구성해야할지 감이 안옴. 1D는 input shape (60,20)으로 하면 될것 같음.

해당 부분에 대해 답을 드리자면, keras layer 중 reshape을 통해서 2D conv의 모양에 맞도록 channel을 추가해주시면 사용하실 수 있습니다. keras.layers.Reshape(target\_shape)