# \*1. [질문] AI를 통해 Finance의 어떤 문제들을 해결할 수 있을까요? (Day 1)\*

해당 질문 들에 대해서는 Thread를 작성하여 댓글로 토론을 진행하려 합니다. 많은 분들이 해당 질문에 대해서 좋은 답변을 해주셨는데요. 좋은 답변을 선택하여 공유드리자면 다음과 같습니다.

1. Increasing security: security와 관련된 빅데이터 분석을 통한 보안 향상. (Anti-Money Laundering, fraud detection 등등)

2. Reducing processing times: financial documentation 처리에 걸리는 시간을 줄여주는 효과.

3. Algorithmic trading:trading pattern을 찾고, trends를 예측하기 위하여 AI를 적용시켜서, 더 성공적인 stock trading decisions을 내리도록 함.

4. Customer recommendation / Credit lending: consumer data, financial lending 혹은 insurance results등의 자료를 기반으로, 고객들의 관심분야를 추천에서부터, 대출 여부를 평가하는 등의 분야에도 적용 가능함.

5. Porfolio Management

김종호(스터디진행자) [7 days ago]

이 문제에 대해서는 모두들 좋은 답변을 주셔서 추가적으로 2개의 질문을 더 소개하여합니다

### Question1. 머신러닝이 할수 있는 예측에 대하여 궁금합니다. 일부에서는 스포츠 승률이라던지 심지어는 복권당첨도 예측하던데요.

김종호(스터디진행자) [7 days ago]

이 질문에 대해서는 어떻게들 생각하시나요~?

권민(10월딥러닝일요10시) [7 days ago]

제 생각에는 정확도의 문제일 뿐이지 데이터가 주어지는 문제라면 다 어느 정도는 예측이 가능할 것 같아요. 스포츠 승률같은 건 더욱 쉽지 않을까 싶어요

김종호(스터디진행자) [7 days ago]

좋은 답변 감사합니다.

이영훈(10월딥러닝일요10시) [7 days ago]

저도 궁금했습니다. 그런데 머신러닝이란게 기존 방대한 데이터를 가지고 컴퓨터가 자신만의 특정 알고리즘을 만들어서 예측하는 건데, 로또 같은 경우 알고리즘이 있지 않고 완전 운이라서 로또 같은 경우는 궁금하네요..

권민(10월딥러닝일요10시) [7 days ago]

복권 당첨의 경우에는 로또는 해당이 안될 것 같아요

김종호(스터디진행자) [7 days ago]

저는 복권에 대해서는 독립된 환경에서 random하게 추출하다보니 단기적으로는 마치 correlation이 무언가 있는것처럼 보일 수 있어도 superious correlation일 것 같습니다.

곽중헌(10월딥러닝일요10시) [7 days ago]

로또같은 것이라면 결국 패턴이 없다는 결과가 나와야 하는 걸까요?

이영훈(10월딥러닝일요10시) [7 days ago]

그런데 토토같은 경우 이미 토토 공식사이트에서 머신러닝을 이용해 데이터 분석을 해놓은게 있었던 것 같았습니다.

김종호(스터디진행자) [7 days ago]

토토가 혹시 축가인가요? 그 부분은 잘 모르겠습니다, 로또는 Superious correlation으로 안될것 같습니다.

김종호(스터디진행자) [7 days ago]

축가 -> 축구

김인철(10월딥러닝일요10시) [7 days ago]

로또는 누가 어느 숫자에 얼마를 걸었는지와 결과가 연관이 없으니 예측은 의미가 없을 것 같아요.

김종호(스터디진행자) [7 days ago]

머신 러닝은 짧은 시간 동안에 여러 분야에 극적인 영향을 끼치고 있습니다. 예측 할 수 있는 문제의 유형도 시간이 지나면서 Combinatorial하게 넓어지고 있는 것 같습니다.

승포츠 승률 뿐만아니라, 기존에 연구되어 왔던 거의 모든 주제에 있어서 새로운 방법론의 등장으로 인해 새롭게 다루어 지고 있는 것 같습니다.

개인적으로 제 주변에서 재밌게 하고 있는 연구들은,

1. 의료 분야 (쥐의 똥에서 추출한 데이터에 기반해서 쥐의 장 내부에 있는 모든 박테리아의 종류와 양을 예측함),

2. 암 환자들의 모바일 사용 정도로 우울증 상태를 예측,

3. 어떤 식으로 Speech를 한 벤처 기업가가 성공적으로 투자를받을 수 있을지 예측

등등이 있는 것 같습니다. 이외에도 나열하자면 끝이 없는 예측 문제들이 있습니다.

현실에서의 Application이 궁금하시면, Top 10 Applications of Machine Learning (https://www.youtube.com/watch?v=ahRcGObyEZo)을 참고해 보시면 좋을 것 같습니다.

혹시 경제학 분야에서의 적용이 궁금하시면, Stanford의 Susan Athey 교수의 "The Impact of Machine Learning on Economics" 논문을 읽어보시길 권유드립니다. (https://www.nber.org/chapters/c14009.pdf)

YouTube edureka!

Top 10 Applications of Machine Learning | Machine Learning Application Examples | Edureka

권민(10월딥러닝일요10시) [7 days ago]

토토는 로또랑 다르니까요..

김종호(스터디진행자) [7 days ago]

넵 :slightly\_smiling\_face: 맞습니다. 다들 해당 문제에 대해서는 좋은 의견을 가지고 계시네요 ㅎㅎ

김종호(스터디진행자) [7 days ago]

2번째 질문으로는

김종호(스터디진행자) [7 days ago]

### Question2. 과연 딥러닝으로 금융시장을 이길수있는 전략을 도출할수있을까요?

김종호(스터디진행자) [7 days ago]

해당 질문에 대해서는 어떻게 생각하시나요~?

주로 Alpha를 얘기할때는 벤치마크 시장 수익률 (코스피, 코스닥, 나스닥, S&P500 등) 대비 수익률을 얘기합니다

효율적 시장가설에 따르면 알려진 정보로는 수익(alpha)을 낼 수 없습니다. 따라서, 다른 사람에 비해서 Superior한 정보, 방법론 등을 가지고 있는 것이 중요한 것 같습니다.

주식에서 주가 예측 모형으로 가장 유명한 Fama French 3 Factor 모델을 예로 들자면,

해당 모델을 통해서 과거 몇년 동안은 예측률이 있고 수익이 났지만, 해당 모델이 널리 알려진 뒤에는

더 이상 다른 사람에 비해 Superior한 모델이 아니기 때문에 수익이 관측되지 않는다 라는 논문이 있습니다.

또한 주식에서는 큰 주식은 이미 너무나 효율적이여서 예측 모델로 수익을 내기가 쉽지 않고,

작은 주식은 거래량이 적어서 높은 슬리피지로 인해 수익을 만들어 내기 어려운 점이 있습니다.

그래서, 중간 사이즈 크기의 주식에서 오히려 Back test시 수익이 나는 경우도 있는 것 같습니다.

코인의 경우도 마찬가지로 BTC에서는 이미 여러 시장 참가자들에 의해 Alpha가 사라지고,

중간 사이즈의 코인에서의 수익률이 더 나아보기도 합니다.

그런데, 예측 모델의 Accuracy Level을 다른 사람의 모델에 비해서 훨씬 더 끌어올릴 수가 있다면,

(새로운 Factor 사용, 다른 모델 사용)

우월한 방법론으로 Alpha를 찾을 수 있지 않을까 기대하고 있습니다.

하지만, 주의하셔야 할점은 M1.2에서 다루었듯이, Theory 없이 (Causual Relationship) 잘 못된 Factor에 기반한 예측은 심각한 Bias를 만들어 낼 수도 있기 때문에 주의가 필요한 것 같습니다.

# 2. [질문] Deep Learning Predictor를 이용한 방법은 Traditional Predictors에 비해 어떤 이점이 있을까요? (Day2)

해당 질문에 대해서도 많은 분들이 좋은 답변을 달아주셨는데요. 답변을 공유드리자면 다음과 같습니다

1. casual model보다 예측에 더 나은 성능을 보인다.

2. 관련이 있는 모든 데이터들을 포함하도록 input data를 확장시킬 수 있다.

3. 선형모델만으로는 표현할 수 없는 것들을 머신러닝은 비선형으로 표현할 수 있음

4. over-fitting 문제를 쉽게 피할 수 있음. (Cross validation, Dropout for model selection, Regularization techniques등을 통해)

5. 가장 좋은 모델을 만들기 위해 머신러닝은 자동적으로 feature를 선택할 수 있다.

해당 슬라이드 자체에 대해서도 질문이 있었어서 답변을 공유드립니다.

### Question1. 인과추론과 예측의 차이를 예를 들어 설명해주세요. 편향없는 추정과 가장 유사한 상관관계는 어떻게 다른가요

해당 자료를 통해서 말하고 싶은 바는, Machine Learning이 어떤 문제를 효과적으로 풀 수 있고, 어떤 문제에는 효과적이지 않은지에 대해 다루고 싶었습니다. 이는 곧 Financial Time Series Prediction에 있어서 주의해야 할 점이여서 강의 초반부에 집고 넘어가고 싶었고요.

핵심만을 요약드리지면, Machine Learning을 통해서는 Prediction 문제를 효과적으로 풀 수 있는 반면에 Causation에 대한 주장은 할 수 없습니다. 이론적인 배경 없이 어떠한 Factor가 어떠한 결과를 야기했다고 볼 수 없기 떄문에, 이러한 예측은 잘 못된 결과를 만들어 낼 수 있습니다.

조금 더 구체적으로, Prediction과 Causation의 차이를 단적으로 예를 들면, 만약 전자레인지 폭발에 의해 불이나 난 상황에서

1. Causation: 불이 일어난 원인은 전자레인지 폭발에 의한 것입니다.

2. Correlation: 소방관이 몰려 있는 곳에는 불이 났을 수 있습니다. 하지만, 소방관이 불을 일으키진 않았고,

둘 사이에는 상관관계가 높을 뿐 입니다.

3. Prediction: 하지만 소방관이 몰려있다는 것으로 부터, 불이 났을 거라는 예측을 할 수는 있습니다.

따라서, Prediction에 있어서는 Causation과 관련 없이 Correlation이 중요 합니다. 하지만, 만약 소방관이 몰려 있는 것이 불을 일으킨다고 잘 못 해석을 하면, 치명적인 예측 오류를 범하게 됩니다.

Prediction과 Causation은 목표로 하는 역할이 다르며, 저희가 Machine Learning을 통해 풀고자 하는 문제의 목표는 Prediction Level을 높히기 위함입니다.

따라서, 이에 따른 단점 (인과성 부족, 이론 부족)을 인지하신 뒤에, 일반적인 인과 관계 추론 모형 (Regression)보다 어떤 점이 더 나을 수 있는지 (비선형성, Over fitting 문제를 더 쉽게 다를 수 있음 등)를 숙지하셨으면 합니다.

박제권 (10월딥러닝일요10시) [7 days ago]

설명감사합니다. ^^

권민(10월딥러닝일요10시) [7 days ago]

오호... 그렇군요

김종호(스터디진행자) [7 days ago]

또한 Slide 내용 중에서 6~10 page 수식 부분에 대한 설명을 궁금해 하시는 분들이 많더라고요, 그래서 답변을 준비하였습니다.

1. 6 page

- Explanation 모형과 Prediction 모형에 있어서 4가지 큰 차이점을 볼 수 있습니다.

- Theory – Data

- Explanation 모형은 Theory에 기반하여 예측 모형의 형태를 신중하게 설정합니다. 예를 들어 선형 관계이면 왜 선형관계인지, Curve Linear 관계를 지니면 왜 그런지에 대해서 다루며, 입력 데이터와 출력 데이터 사이의 관계가 중요합니다.

- Prediction 모델에서는 예측 함수의 형태는 데이터로 부터 나오며, 입력 데이터와 출력 데이터 사이의 관계는 중요하지 않습니다.

- Causation – Association

- Explanation Model에서는 X가 Y를 Cause합니다

- Prediction Model에서는 X가 Y와 Correlation이 있습니다.

- Bias – Variance

- Explanation 모형에서는 Bias를 최소화한 Factor (Causual Effect)를 찾는 문제를 풀지만

- Prediction에서는 Bias와 Variance의 합을 최소화 하는 모형을 찾습니다.

- Retrospective (in-sample) – Prospective (out-of-sample)

- Predictive modeling은 forward-looking을 목적으로 하는 반면에

- Explanation 모형은 이미 세워진 가설 관계를 검증하는 용도로 사용합니다.

곽중헌(10월딥러닝일요10시) [7 days ago]

사실 사람도 저런 논리 오류를 자주 범하지 않을까요?

고재민(10월딥러닝일요10시) [7 days ago]

논리 오류를 일으키는건 저게 오류라는걸 알 수 있을것 같은데

곽중헌(10월딥러닝일요10시) [7 days ago]

원인을 잘못 해석하는

김종호(스터디진행자) [7 days ago]

맞습니다. Correlation을 마치 Causation마냥 착각해서 잘못된 행동을 하는 경우도 많은것 같습니다.

고재민(10월딥러닝일요10시) [7 days ago]

머신러닝은 블랙박스 시스템이라 어디가 잘못 됐는지 알기 힘들지 않을까요

김종호(스터디진행자) [7 days ago]

넵. 블랙박스 모델이다보니, Interpretability가 상당히 이슈가 되고 연구가 많이 되고 있는 것 같습니다.

김종호(스터디진행자) [7 days ago]

재밌는 주제로

권민(10월딥러닝일요10시) [7 days ago]

머신러닝이라기 보다는... 뉴럴넷이라던가 딥러닝이라던가 하는 용어로 통일하는 게 좋겠네요

김종호(스터디진행자) [7 days ago]

넵 :slightly\_smiling\_face:

김종호(스터디진행자) [7 days ago]

해당 스터디 자료의 7~10page에 대한 설명도

김종호(스터디진행자) [7 days ago]

첨부드립니다.

권민(10월딥러닝일요10시) [7 days ago]

기존의 선형회귀등도 전부 머신러닝에 들어가기땜시..

김종호(스터디진행자) [7 days ago]

2. 7 page

- Bias: Bias is the result of misspecifying the statistical model f.

- Variance: Estimation variance (the third term) is the result of using a sample to estimate f.

- The first term is the error that results even if the model is correctly specified and accurately estimated

- The above decomposition reveals a source of the difference between explanatory and predictive modeling:

- In explanatory modeling the focus is on minimizing bias to obtain the most accurate representation of the underlying theory.

- In contrast, predictive modeling seeks to minimize the combination of bias and estimation variance, occasionally sacrificing theoretical accuracy for improved empirical precision

3. 8 page

- Causual Inference는 Unbiased estimates를 목표로 합니다.

- 이를 검증하기 위해서, 연구 대상 A가 factor X에 영향을 받았을 때와 받지 않았을 때의 차이를 보고 싶은데,

- Observational Data (실험이 아닌)에 기반하면 사실은 연구 대상 A가 factor X에 영향을 받았을 때와,

- X에 노출 된적 없었던 그룹 연구 대상 B가 factor X에 영향을 받지 않았을 때의 차이를 보게 됩니다.

- 완전히 동일한 연구 대상에 2개의 다른 Treatment를 가할 수가 없습니다.

- A -> Treatment -> A'

- Note. we can not observe (A -> No Treatment -> A'') (시간이 지나면서 A도 Treatmnet가 아닌 다른 무언가에 영향을 받음)

- What we want to see: factor x => (A'-A'')

- However, what we actually compare

- B -> No Treatment -> B'(?) (시간이 지나면서 B도 뭔가 바뀜)

- Then, we observe factor x => (A'-B') - (A-B)

4. 9 page

- Prediction은 Out of sample에서의 High Predictive Power를 목표로 합니다.

- 이를 통해서 모든 Correlation을 고려하여 Bias를 줄이며,

- 동시에 Overfitting 문제를 다루어 Variance를 최소화 합니다.

5. 10 page

- 따라서 Prediction 모델과, Causation 모델은 다른 두가지 목적으로 사용됩니다.

- 어떠한 Factor X가 정말로 Outcome Y에 영향을 미쳤는지를 Causual하게 알고 싶으면,

- Causation 모형을 사용하여야 하고

- 예측률이 높은 모형을 만들고 싶으면 Prediction 모형을 쓰면 됩니다.

김인철(10월딥러닝일요10시) [7 days ago]

이건 개그... 이상한 correlation 들을 모아놨네요: http://tylervigen.com/spurious-correlations

tylervigen.com

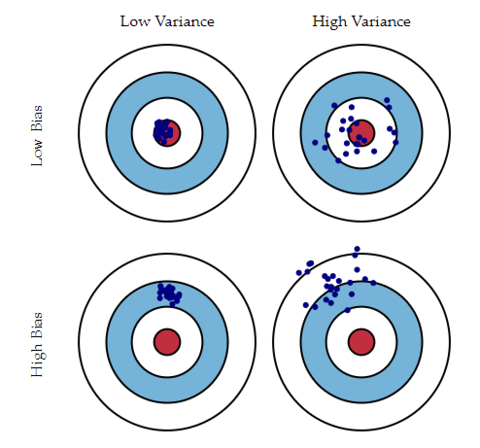
15 Insane Things That Correlate With Each Other

Why do these things correlate? These 15 correlations will blow your mind. (Is this headline sensationalist enough for you to click on it yet?) (41 kB)

김종호(스터디진행자) [7 days ago]

네 machine learning의 범위는 넓게 정의하면 NN말고도 다 포함이 되서, 용어가 햇갈리실 수 있는것 같습니다 ㅎㅎ 재밌는 correlation들 이네요 해당 질문에 대해서 추가적인 질문이 있었는데

### Question3. unbiased estimate란?

이 질문에 대해 답변을 드리자면 다음과 같습니다. unbiased estimator란 모델 추정량의 평균이 실제 모수값과 일치할 때를 뜻합니다. 아래의 자료의 그림 "Graphical illustration of bias and variance" 을 보시면, Low Bias에 대한 이해가 분명해 지실 것 같습니다. 

https://github.com/jonghkim/financial-time-series-prediction/raw/master/Q%26A/bias\_vs\_variance.png

Low Bias일때는 과녁의 중앙에 맞기는 합니다 Variance가 높을 경우에 퍼질수 있긴 하지만요 혹시 해당 주제에 대해서 추가적으로 궁금하신 부분이 있으신가요~? 제가 조금 빠른가요~? 혹시 궁금하신 부분이 있으시면 편하게 질문 바랍니다

# 4. [실습] Google Colab에 접속하여 아래 링크의 Keras 예제 코드를 GPU를 사용하여 실행하여 봅시다 (Day 3, 5)

해당 질문에 대해서는 아래와 같은 질문이 있었습니다

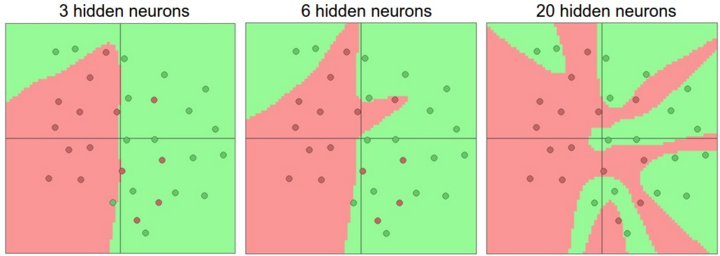
## Question2. 은닉 레이어의 출력 값이 정해지는 기준( (tf.keras.layers.Dense(128, activation=tf.nn.relu) ) 시에 128이 정해지는 기준, 영상에서 레이어 한개는 뉴럴 네트워크 두개이상 딥 뉴럴 네트워크라고 설명을 했는데 같은값으로 은닉 레이어를 두개 만들었는데( model.add(tf.keras.layers.Dense(128, activation=tf.nn.relu)) model.add(tf.keras.layers.Dense(128, activation=tf.nn.relu)) ) 이렇게 할 시 한 개와 어떤 차이점이 있는지. Overfitting과 레이어 갯수의 결정 사이에 대한 연관성 레이어 수와 크기 설정의 문제에 있어서 사용할 아키텍처를 어떻게 결정할까요?

Hidden Layer를 쓰는게 좋을지, 하나의 Hidden Layer를 쓸지, 두 개의 Hidden Layer를 쓸지, 각 층은 얼마나 커야하는지에 대한 고민이 있을 수 있습니다.

간략히 답을 드리자면, 첫째로, 신경망의 크기와 수를 증가 시키면 네트워크의 표현력(Capacity)이 증가한다는 점에 유의하시면 됩니다.

즉, 뉴런이 많을 수록 표현 가능한 공간이 커집니다.

예를 들어, 2차원에서 아래와 같은 2진 분류 문제가 있다고 가정하면, 아래 3개의 신경망(모두 하나의 층만 있음)은 Node의 개수를 달리 했을 때의 결과를 보여 줍니다.



<https://github.com/jonghkim/financial-time-series-prediction/raw/master/Q%26A/layer_sizes.jpeg>

위의 다이어그램에서 뉴런이 더 많은 네트워크는 더 복잡한 기능을 표현할 수 있지만, 이 결과는 blessing과 (더 복잡한 데이터를 분류하는 것을 배울 수 있기 때문에)과 Curse입니다 (훈련 데이터를 Overfitting 하기 쉽기 때문에).

복잡한 모형보다, 3개의 hidden neuron을 가진 모델이 전체 데이터를 큰 두가지 모양으로 구분하고 나머지 점들은 noise로 해석하는데, 더 나은 일반화를 한 모델일 수 있습니다.

추가적으로, 128이라는 숫자는 사실 Convention에 가까운 것 같습니다. 직접 테스트 해보지는 못 했지만, 32의 배수가 GPU가 메모리에서 다루기가 편하며, 속도적인 이점이 있다고도 합니다.

참고: <http://cs231n.github.io/neural-networks-1/>

권민(10월딥러닝일요10시) [7 days ago]

https://playground.tensorflow.org/ 여기서 신경망 뉴런 수와 층 바꾸면 어떻게 돌아가는지 맛보기로 볼수있어요.. 추천합니다..!

김종호(스터디진행자) [7 days ago]

아 넵 좋은 자료내요..! 저도 이 예제를 쓰고 싶었는데 링크를 까먹어서.. ㅎㅎ

김종호(스터디진행자) [7 days ago]

Overfitting에 대한 얘기를 조금 더 드리자면,

딥러닝에서 레이어의 개수 혹은 차원의 개수에 대한 결정은 Overfitting Issues와 큰 관련이 있습니다.

Overfitting은 1. 모델이 너무 복잡하거나 2. 데이터가 너무 적거나 하는 문제로 인해서 발생할 수 있습니다.

1번의 모델이 너무 복잡해서 발생하는 문제는 주로 Cross Validation을 통해 Overfitting이 발생하지 않았는지를 테스트하며, 같은 결과면 간단한 모델이 더 좋은 모델 (오캄의 면도날)의 원칙을 적용하는 것을 권장합니다.

관련한 보다 자세한 이슈는 아래 링크를 통해 공유드립니다. (https://drive.google.com/open?id=1HqCyp94ZC9f7jwhStFnSbuF9TDyrf5iy)

Overfitting을 해결하기 위한 방법은 Module3에서 관련하여 보다 자세히 다룰 예정입니다.

(Learning Curves, Batch Normalization, Dropout and Regularization, Continuous Learning, Hyperparameter Search)

2번의 데이터 샘플에 대한 이슈는, 주어진 데이터에 대해 Train 데이터, Test 데이터의 크기를 결정해야 할 때가 있는데, 둘 사이의 샘플 개수 결정은 경쟁 관계에 있습니다.

최종적인 목적은, generalization error 를 최소화 하기 위함인데, 만약 train 사이즈가 작으면 paramter estimate의 variance 가 높고,

반대로 test 사이즈가 작으면 performace statistic의 variance가 높아집니다.

따라서, 양쪽의 variance가 너무 높지 않도록 설정해야하는데, 이는 sample size에 따라서 그 크기를 결정하는 방식에 차이가 날 것 같습니다.

예를들어 100개의 sample에 대해서 train / test 를 쪼갤때는 어떤 방식을 취해도 parameter estimate variance를 줄이기가 쉽지가 않을 것이고, 100000개의 sample을 train / test 를 쪼갤때는 0.8:0.2 를 취하거나, 0.9:0.1 을 취하든 generalization에 문제가 없을 것으로 보입니다.

variance를 줄이기 위한 방법으로는 train을 비율 (90, 80, 70) 등을 바꿔가면서 각 비율당 여러차래 (ex. 20회 이상) random sampling 하여 mean performance를 비교하여 lower variance를 선택하는 방법을 쓰기도 합니다.

Cross Validation에 대한 설명은 아래 링크를 참조해 보시면 좋을것 같습니다. (https://www.youtube.com/watch?v=TIgfjmp-4BA)

모델이 복잡할때 생길 수 있는 이슈에 대한 예제는 아래의 링크를 참조해 보시면 됩니다. (<https://towardsdatascience.com/beginners-ask-how-many-hidden-layers-neurons-to-use-in-artificial-neural-networks-51466afa0d3e>)

## Question3. 텐서플로우에서 모델 저장하면 checkpoint 파일등 여러개가 만들어지잖아요. 케라스에서는 .h 파일 하나만 생기는겁니까? 혹시 텐서플로우의 .pb 파일에 대응하는 건가요?

다른 분반에서 해당 질문을 주신 분이 있어서 질문을 공유드립니다

Checkpointing 기능은 긴 시간 (days, weeks) 동안의 학습시, train과정에 문제가 있더라도 모든 학습 정보를 잃지 않도록하는데 필수적인 기능이여서, 장기적으로는 꼭 배워둘 필요가 있는 것 같습니다.

질문 주신 Keras의 경우에는 아래의 callback API를 통해 checkpointing 기능을 제공합니다.

해당 기능을 통해서, 저장 위치, checkpoint 이름 (epoch number 혹은 metric으로도 naming 가능),

어떠한 조건 (save\_best\_only, save\_weights\_only, period)을 만족하는 checkpoint를 생성 할지를 설정할 수 있습니다.

https://keras.io/callbacks/#modelcheckpoint

Google Colab에서는 얘기주신 것처럼 training 이후 HDF5 format으로 전체 모델을 직접 저정할 수도 있으며,

callback API를 통해 training 과정 중에 모델을 저장하는 것도 가능합니다.

자세한 checkpoint 저장 방법은 아래의 튜토리얼을 참조해 주세요.

<https://colab.research.google.com/github/tensorflow/models/blob/master/samples/core/tutorials/keras/save_and_restore_models.ipynb#scrollTo=2S4xrNJRilwi>

Model Save Load, Retrain은 Module2.3에서 다룰 예정이기도 합니다

## Question10. AI로 뭔가를 학습 시킬 때 전처리를 할텐데 이런 타임시리즈의 데이타는 어떤식으로 전처리를 하나요? 이동평균선?, 볼린져밴드 등 사용하나요?

해당 질문에서 있었던 방법 (MA, 볼린저벤드)은 주로 Trend와 Cycle을 구분하기 위한 방법으로 보입니다.

사실 두가지 방법에 대한 Practical하게 다루는 곳은 많이 봤지만, 학문적인 Support를 주로 보지는 못 한것 같습니다.

해당 방법과 유사한 방법으로, 주로 학계에서는 Trend와 Cycle을 구분하기 위한 방법으로 Hodrick Prescott Filter를 이용합니다. (원레는 Macro Economy에서 Expansion/Recession을 구분하기 위해 사용하던 방법론)

<https://en.wikipedia.org/wiki/Hodrick%E2%80%93Prescott_filter>

(최근에 흥미롭게 읽었던 논문 중에서는 4차 방정식이 Hodrick Prescott Filter보다 좋으니까 쓰지 말라는 말도 있습니다. https://econweb.ucsd.edu/~jhamilto/hp.pdf)

현재에는 Outcome 변수에 대해서만 설명 드리자면, Preprocessing으로 아래와 같은 과정들을 취하고 있습니다.

핵심은 데이터의 분포를 가능한 Normal Distribution으로 만들어 주려는 것 이고요, 그런 경우에 Significant Change를 잡아내는 방향으로 하고 있습니다.

1. Continuos Value에 대한 예측시에는 Log Return을 예측 값으로 사용 (그냥 Return의 분포는 Kurtosis가 너무 높음)

3. Continuos Value에 대한 예측시 HP Filter 이후에 Cycle 파트에 대한 예측, Trend 파트에 대한 예측

2. Discrete Value (Up/Down) 문제를 풀때는, Outcome 변수를 Significant Change를 상정하여 (1 std 이상 상승/횡보/하락)으로 구분

위의 설명은 Outcome 변수에 대한 것 이지만, 예측에 사용되는 다른 변수들은 그 변수의 특징에 따라서 다른 Preprocessing 과정을 거치고 있습니다.

권민(10월딥러닝일요10시) [7 days ago]

저 궁금한 게 있어요. 주가 예측 같은 경우에는 데이터를 어떤 시간 단위로 나누나요? 데이터 하나를 5분 간격으로 두는지..

김종호(스터디진행자) [7 days ago]

저는 1min, 5min, 10min, 1hour 전부다 해보고 있습니다. 가장 Performance가 좋은 시점을 선택하면되는데 모델마다 차이가 있어서

고재민(10월딥러닝일요10시) [7 days ago]

질문 범위에서 벗어나는 것 같지만.... 1분봉 데이터는 어떤식으로 수집하시나요? 나중에 설명하실 것 같긴 합니다 ㅎㅎ

김종호(스터디진행자) [7 days ago]

코인 데이터는 coinone api를 사용하였습니다

곽중헌(10월딥러닝일요10시) [7 days ago]

5일선에서 안보이던 패턴이 20일선에서 보이는 일도 있죠 ㅎㅎ

김종호(스터디진행자) [7 days ago]

coinone api를 규칙적으로 돌리기 위해서 celery를 사용하였고요

박제권 (10월딥러닝일요10시) [7 days ago]

저도 그거 문제였습니다. 틱데이터를 한땀한땀 매일 매일 모으는 스크립트를 만들기도 했습니다. ㅎㅎ (edited)

김종호(스터디진행자) [7 days ago]

이거를 또 혹시 process가 죽지 않았을까 걱적되서 supervisord를 이용해서 process 관리를 했습니다. 지금은 airflow로 옮겨가고 싶어서 준비중에 있고요 ㅎㅎ

고재민(10월딥러닝일요10시) [7 days ago]

supervisord는 찾아봐야겠네요. 이번 이동평균선, 볼린져밴드 등 질문 관련해서 하나 더 질문이 있는데요

김종호(스터디진행자) [7 days ago]

넵

고재민(10월딥러닝일요10시) [7 days ago]

결국 보조지표를 사용해서 feature를 늘리는 방식을 접근한 논문은 잘 없나요?

김종호(스터디진행자) [7 days ago]

주가에 기반한 보조지표는 학문적으로는 거의 다뤄지지가 않습니다. fundamental factor에 기반한 모형은 논문으로 나오고 있습디ㅏ

Fama French 3 factor / 5 factor 모형

등등과 같이 어떠한 factor가 이론적 배경을 가지고 있는 경우에는 나오는 것 같습니다. 보조지표로 보는게 correlation은 주장할수 있어도 causality에 대한 주장이 어려워서 top journal에서는 받아주지 않는것 같습니다 ㅠ

## Question2. data을 scaling하고 train 하는 이유가 있을것 같아서 문의드립니다. 어떤 모델에서든 항상 0~1값으로 변경해야하는것인지?

해당 질문은 위 thread에서 공유된 적이 있을텐데요 그래도 한번 더 집고 넘어가도록 하겠습니다.

이론적인 면에서는 Normalize는 Neural Network의 Weight, Bias가 조정되는 것으로 모두 잡힐 수 있어야 하지만, Practical 하게는

1. training more stable and faster

2. reduce the chances of getting stuck in local optima

3. 학습과 예측에 사용되는 Domain 데이터 사이의 샘플링 Population 일치 문제 (out of scale from one another)

를 해결 하기 위한 것으로 알려져 있습니다.

1번, 2번의 경우를 보다 자세히 얘기드리자면, gradient descent를 할때는 gradient error vectors와 learning rate를 곱해서 weight를 조정해 주는데,

input feature들 간의 distribution 차이가 나면, 한쪽 feature에 대해서는 잘 작동하는 learning rate가 다른 feature에 대해서는 under compensating하는 문제가 있을 수 있습니다.

(since the learning rate in the update equation of Stochastic Gradient Descent is the same for every parameter) (Elkan, Charles. (2008) Log-linear models and conditional random fields)

이런 이유로 error surface에서 Global minimum을 찾기 위해서는 구모양에 가까운 input feature를 가지는 것이 타원형 구조 (한쪽 feature의 range가 훨씬 큰 경우)보다 빠르게 converge 하는 것으로 알려져 있습니다.

3번 같은 경우에 조금 더 쉬운 예를 들자면, 주식/코인 같은 경우에는 Price를 그대로 입력하는 것이 아니라,

Return, Log Return을 Scaling의 목적으로 사용할 때가 있는데, 학습할때의 Price 값 (ex: KOSPI: 2000, BTC: $3000)의 기준과 예측할 때(ex: KOSPI: 2300, BTC: $7000)의 Price 값의 기준이 크게 차이가 나는 문제를 해결하기 위해서 사용합니다.

(같은 input feature이지만, 시기가 달라서 학습에 관찰되지 않은 데이터가 입력 될 수 있음, out of scale from one another)

또한, 가격에 대한 입력과, Bid/Ask Volume에 대한 입력은 Scale 차이가 상당한데 이러할 경우에, 모든 feature에 기반하여 모델이 작동하기 보다는,

Range가 보다 큰 particular한 feature에 모델이 종속되어 버리는 것 같습니다.

(ex. 모델 내에서 distance metric이 계산될 경우 값의 범위가 큰 feature에 종속)

그리고, 마지막으로 0~1 값으로 변경해야 할지, 다른 옵션 (Mean Normalization, Standardization, Scaling to unit length)를 할지,

혹은 Scailing이 하지 않을지 등에 대한 Feature Scaling 선택 문제가 있을 수 있는데,

풀고자 하는 문제의 특성에 따라서 다른 Scaling 방법을 취해주는 것이 좋을 것 같습니다.

가량 Exact Boundary가 있는 문제의 경우에는 Standardization이 적절하지 않을 수도 있고,

min-max scaling은 long tail 분포가 있는 문제에 대해서는 적절하지 않을 수 있습니다.

권민(10월딥러닝일요10시) [7 days ago]

궁금한 게 있어요. 그때 그 trian이랑 test랑 따로 normalize를 했었는데 그래도 상관없나요? 하나로 concat 해서 안 하고 따로 normalize 해도 상관업는지...

x\_train = tf.keras.utils.normalize(x\_train, axis = 1)

x\_test = tf.keras.utils.normalize(x\_test, axis = 1)

박제권 (10월딥러닝일요10시) [7 days ago]

저는, 어차피 나중에 실제 predict 할 데이터도 따로 normalize 될꺼니까, 라고 실용적으로만 이해했습니다.

권민(10월딥러닝일요10시) [7 days ago]

저는 여기서 의 x\_test가 실제 predict할 데이터라고 받아들였걸랑요..흠..

김종호(스터디진행자) [7 days ago]

좋은 질문이네요..! Predict할때 어차피 새로운 데이터만 가지고 할꺼라 practical 하게는 normalize를 따뤄 해주겠지만, 그 이전에 데이터의 특성에 따라서 전처리를 적절히 해준 이후에 따로하면 되지 않을까 싶습니다.

과연 2개 (train / test)가 완전히 동일한 분포를 지니는 Population으로 부터 추출되었는지를 고려해보면, 적절한 preprocessing과정이 빠지면 이런게 보장이 안되서 train에서 쓴 데이터 분포와 test 데이터 분포가 달라질 수 있을것 같고요

권민(10월딥러닝일요10시) [7 days ago]

아아 네 그리고 생각해보니 이거는 어차피 픽셀의 밝기값이라 상관없을 것 같긴 하네요...ㅋㅋ 감사합니다