Big Data Term Project Report

과목명	빅데이터
교수명	김무철 교수님
제출일	2023/12/12
팀구성	20184757 주영석 20194702 김의진

Introduction

MLB 는 2022 년 총수익이 108 억 달러(13 조 4662 억)원이 발생하였으며 이는 지구상에서 2 번째로 수익이 많이 발생하는 스포츠리그라고 알려져있다. 이렇게 거대한 자본이 움직이는데 그중 크게 화제가 되는 것은 선수들의 몸값이다. 각 구단에서는 선수들의 가능성을 여러방면에서 평가하여 조건을 제시하게 되는데 해당 분야의 전문가들이 모였지만 부상과 같이 고려하기 힘든 변수들에 의해 큰 금액을 주고 데려온 선수지만 제 값을 못해주는 소위 "먹튀"라는 명칭이 붙은 선수들도 종종 있다.

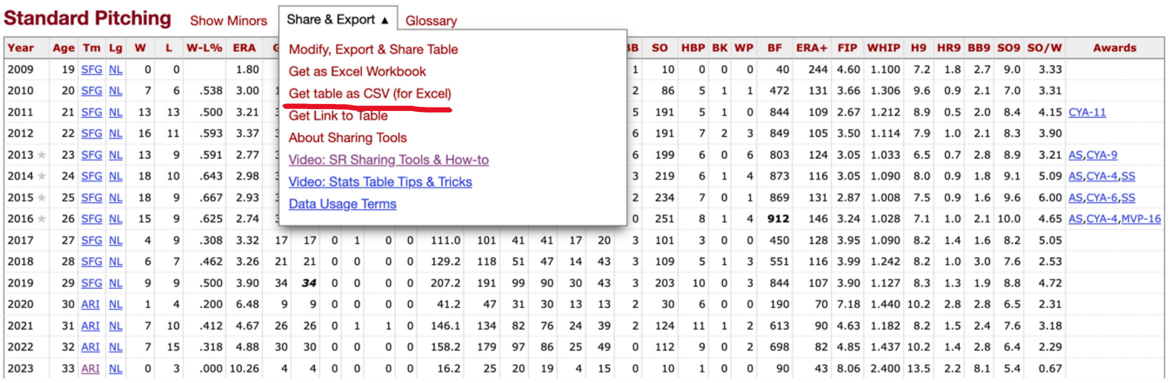
야구의 별칭은 "통계의 스포츠"이다. 그만큼 타 종목에 비해 다양한 지표들을 만들고 기본적인 통계부터 다양한 정규화를 통해 선수의 능력을 수치화하기 위해 많은 노력을 하고 있다. 실제로 많은 구단에서 유망주, 혹은 자유 계약 신분이 되는 선수들의 가능성을 측정할 때 특정 지표를 우선적으로 보는 등 단순하게 산술적인 계산이나, 심지어는 감독, 코치진의 개인적인 지도 가지관이나 야구이념 등이 선수의 평가에 영향을 주곤 한다.

기존에 타자들의 지표를 학습시켜 성적을 예측했던 시도는 있었으나 투수에 대해서는 시도가 잘 이루어 지지 않았고 왜 타자를 대상으로만 학습을 진행했는지에 대한 언급이 없었기에 이번 탐구를 통해 투수들에 대해 다양하게 정의된 여러 지표들사이에 복합적으로 이루어진 연관성을 Fully Connected Neural Network, 시즌마다 쌓이는 지표이기에 Time Series 형태의 데이터 분석에 용이한 Long-Short Term Momory Network 와 같은 모델을 적용시켜 특정 선수의 이전시즌 활동을 기반으로 향후 3 년의 지표를 예상해보는 탐구를 진행하고자 한다.

Implementation

Data Collection & Preprocessing

데이터 수집 대상을 선정할때는 최우선적으로는 매 시즌 꾸준하게 정규이닝을 던져준 선발투수를 최우선으로 생각했으며, Jacon deGrom, Clayton Kershaw와 같이 리그 최상위권 투수들 뿐만 아니라 Ted Lilly, Matt Cain와 같이 압도적으로 리그를 호령한건 아니지만 꾸준하게 성적을 내주었다고 판단되는 투수들 또한 명단에 포함시켰다.



<Madison Bumgarner's Pitching Data Table>

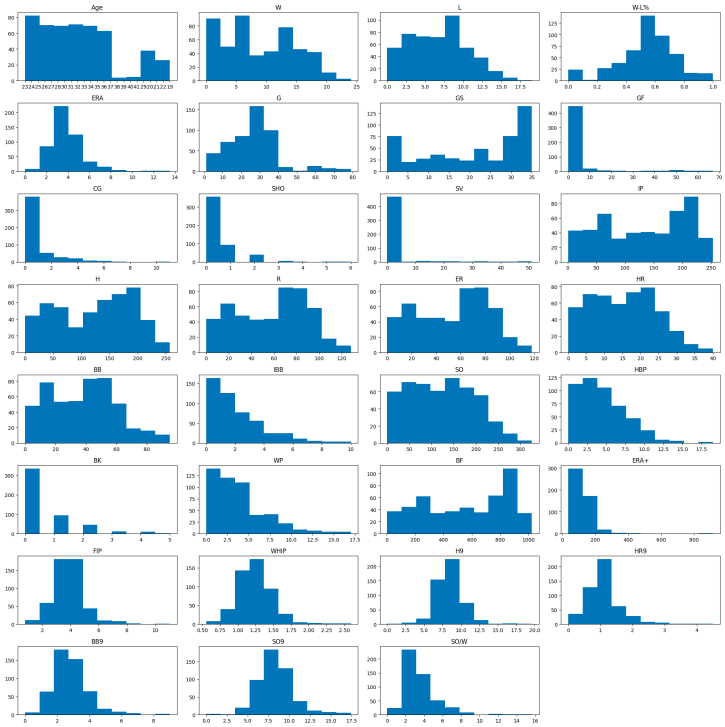
```
Year,Age,Tm,Lg,W,L,W-L%,ERA,G,GS,GF,CG,SHO,SV,IP,H,R,ER,HR,BB,IBB,SO,HBP,BK,WP,BF,ERA+,FIP,WHIP,H9,HR9,BB9,SO9,SO/W,Awards,Year-additional
2009,19,SFG,NL,0,0,,1.80,4,1,1,0,0,0,10.0,8,2,2,2,3,1,10,0,0,0,40,244,4.60,1.100,7.2,1.8,2.7,9.0,3.33,,-9999
2010,20,SFG,NL,7,6,.538,3.00,18,18,0,0,0,0,111.0,119,40,37,11,26,2,86,5,1,1,472,131,3.66,1.306,9.6,0.9,2.1,7.0,3.31,,-9999
2011,21,SFG,NL,13,13,.500,3.21,33,33,0,0,0,0,204.2,202,82,73,12,46,5,191,5,1,0,844,109,2.67,1.212,8.9,0.5,2.0,8.4,4.15,CYA-11,-9999
2012,22,SFG,NL,16,11,.593,3.37,32,32,0,2,1,0,208.1,183,87,78,23,49,6,191,7,2,3,849,105,3.50,1.114,7.9,1.0,2.1,8.3,3.90,,-9999
2013,23,SFG,NL,13,9,.591,2.77,31,31,0,0,0,0,201.1,146,68,62,15,62,6,199,6,0,6,803,124,3.05,1.033,6.5,0.7,2.8,8.9,3.21,ASCYA-9,-9999
2014,24,SFG,NL,18,10,.643,2.98,33,33,0,4,2,0,217.1,194,81,72,21,43,3,219,6,1,4,873,116,3.05,1.090,8.0,0.9,1.8,9.1,5.09,ASCYA-4SS,-9999
2015,25,SFG,NL,18,9,.667,2.93,32,32,0,4,2,0,218.1,181,73,71,21,39,2,234,7,0,1,869,131,2.87,1.008,7.5,0.9,1.6,9.6,6.00,ASCYA-6SS,-9999
2016,26,SFG,NL,15,9,.625,2.74,34,34,0,4,1,0,226.2,179,79,69,26,54,0,251,8,1,4,912,146,3.24,1.028,7.1,1.0,2.1,10.0,4.65,ASCYA-4MVP-16,-9999
2017,27,SFG,NL,4,9,.308,3.32,17,17,0,1,0,0,111.0,101,41,41,17,20,3,101,3,0,0,450,128,3.95,1.090,8.2,1.4,1.6,8.2,5.05,,-9999
2018,28,SFG,NL,6,7,.462,3.26,21,21,0,0,0,0,129.2,118,51,47,14,43,3,109,5,1,3,551,116,3.99,1.242,8.2,1.0,3.0,7.6,2.53,,-9999
2019,29,SFG,NL,9,9,.500,3.90,34,34,0,0,0,0,207.2,191,99,90,30,43,3,203,10,0,3,844,107,3.90,1.127,8.3,1.3,1.9,8.8,4.72,,-9999
2020,30,ARI,NL,1,4,.200,6.48,9,9,0,0,0,0,41.2,47,31,30,13,13,2,30,6,0,0,190,70,7.18,1.440,10.2,2.8,2.8,6.5,2.31,,-9999
2021,31,ARI,NL,7,10,.412,4.67,26,26,0,1,1,0,146.1,134,82,76,24,39,2,124,11,1,2,613,90,4.63,1.182,8.2,1.5,2.4,7.6,3.18,,-9999
2022,32,ARI,NL,7,15,.318,4.88,30,30,0,0,0,0,158.2,179,97,86,25,49,0,112,9,0,2,698,82,4.85,1.437,10.2,1.4,2.8,6.4,2.29,,-9999
2023,33,ARI,NL,0,3,.000,10.26,4,4,0,0,0,0,16.2,25,20,19,4,15,0,10,1,0,0,90,43,8.06,2.400,13.5,2.2,8.1,5.4,0.67,,-9999
15 Yrs,15 Yrs,15 Yrs,15
Yrs,134,124,.519,3.47,358,355,1,16,7,0,2209.1,2007,933,853,258,544,38,2070,89,8,29,9098,110,3.63,1.155,8.2,1.1,2.2,8.4,3.81,,-9999
162 Game Avg.,162 Game Avg.,162 Game Avg.,162 Game
Avg.,13,12,.519,3.47,34,34,0,2,1,0,211,191,89,81,25,52,4,197,8,1,3,868,110,3.63,1.155,8.2,1.1,2.2,8.4,3.81,,pitch_glossary.shtml#162_avg
,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,-9999
SFG (11 yrs),SFG (11 yrs),SFG (11 yrs),SFG (11
yrs),119,92,.564,3.13,289,286,1,15,6,0,1846.0,1622,703,642,192,428,34,1794,62,7,25,7507,120,3.32,1.111,7.9,0.9,2.1,8.7,4.19,,-9999
ARI (4 yrs),ARI (4 yrs),ARI (4 yrs),ARI (4
yrs),15,32,.319,5.23,69,69,0,1,1,0,363.1,385,230,211,66,116,4,276,27,1,4,1591,80,5.17,1.379,9.5,1.6,2.9,6.8,2.38,,-9999
```

<Extracted CSV data (Madison Bumgarner)>

초기에는 BeautifulSoup 을 사용해서 한번의 여러 선수의 데이터를 한번에 크롤링하고자 했으나 URL 의 구조가 선수명이 아닌 선수에게 배정된 고유키로 이루어져 있어 웹에서 제공하는 툴을 사용하여 데이터를 얻었다.

Data Preprocessing

우선, 데이터에서 숫자가 아닌 데이터를 (팀, 리그, 수상) 제거하고, 학습시키는데 관계 없을 것으로 추측되는 데이터를 (년도) 제거했다. 다음으로, 데이터 분포를 확인해 보았을 때 특정 구간에 분포가 매우 많고 값의 범위가 넓은 데이터의 경우 로그를 취해 분포가 기울어진 정도를 조정해 주었다. 해당하는 데이터로는 GF, CG, SHO, SV, IBB, HBP, WP, ERA+가 있다.



<Data 분포>

다음으로, 값이 없는 열을 제거한 후, 모든 feature에 대해 정규화 작업을 진행했다. 이런 전처리 과정을 거쳐 얻은 데이터는 32명의 투수에 대한 31개의 feature 데이터다. 각 투수마다 선수 생활의 길이는 다르다.

3년간의 모든 지표 데이터를 입력하고, 향후 3년간의 성적 지표를 예측하는 것을 목표로 하고 있으므로, 전처리 이후의 데이터에서 두 가지 방법으로 데이터를 얻을 수 있다. 첫째, 선수 생활 중 가장 초반 6년의 데이터를 가져오는 방법이다. 이 방법으로는 32명의 투수에서 각각 하나의 데이터밖에 얻지 못하므로, 총 데이터의 크기가 32개로 매우 적다. 둘째, sliding window를 사용하여 가능한 모든 6년 길이의 시퀀스를 뽑아내는 것이다. 이 방법을 통해 데이터 327개를 얻을 수 있다.

Model

투수들의 3 년간의 지표 데이터를 기반으로 향후 3 년의 투수의 ERA, WHIP 지표를 예측하고자 한다. 이 때, 이전 3 년간의 지표 데이터는 시간의 흐름에 따라 표현되는 데이터, 즉, 시간 축에 나열할 수 있는 데이터이다. 이후 3 년의 지표를 예상할 때, 이전 3 년의 기록에서 모델이 어떠한 관계를 발견하고 학습하는 것을 목표로 하기 때문에, 모델은 시계열 분석에 적합한 순환신경망 (Recurrent Neural Network) 모델, 그 중에서도 현재 인공지능 분야에서 가장 널리 이용되는 장단기 메모리를 (Long Short-Term Memory, LSTM) 기반으로 하는 딥러닝 모델을 사용하고자 한다. 또한, 비교 분석을 위해 일반적인 딥러닝 모델인 Fully Connected Neural Network 도 동일한 데이터를 이용하여 학습시킨다.

우선, LSTM 모델의 경우, 각 데이터 세트를 위해 두 가지 모델을 구성하였다. 첫번째 모델은 투수의 첫 3 년간의 기록만을 사용하는 경우의 모델로, 사용할 수 있는 데이터의 양이 적기 때문에 모델의 크기를 상대적으로 작게 구성하였다. LSTM 레이어의 개수는 2 개, 은닉 상태의 크기는 128 로 구성했으며, 입력 시퀀스 중 마지막 시간 순서의 입력이 들어간 이후 LSTM 레이어 중 입력을 받지 않는 LSTM 의 은닉 상태를 Linear 레이어에 통과시킨 값을 출력하도록 설정했다. 두번째 모델은 투수의 지표 기록에서 sliding window 를 적용시켜 가능한 모든 3 년간의 입력과 이후 3 년의 예측 지표 데이터를 사용하는 경우의 모델로, 첫번째 모델에 비해 상대적으로 큰 모델로 구성하였다. LSTM 레이어의 개수는 동일하게 2 개이지만, 은닉 상태의 크기를 256 으로 구성했고, 출력은 마찬가지로 LSTM 의 은닉 상태를 사용했는데, 이전과 달리 두 개의 fully connected layer 와 batch normalization, dropout 을 사용한 네트워크에 통과시킨 값을 사용하였다.

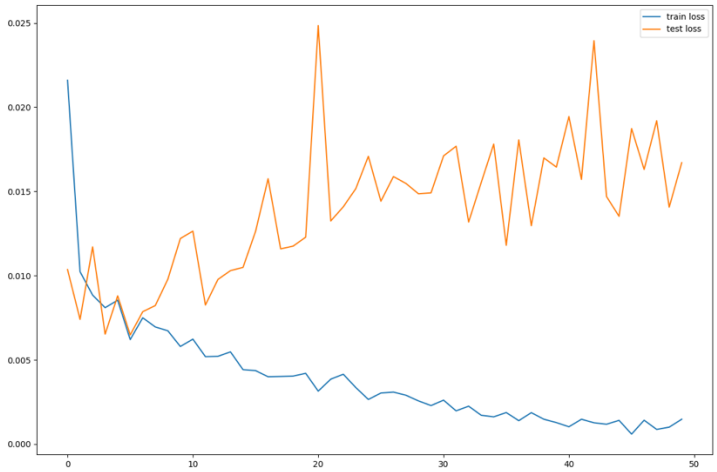
다음으로, Fully Connected Neural Network 도 LSTM 모델과 마찬가지로 각 데이터 세트를 위해 두 가지 모델로 구성하였다. 첫번째 모델은 3 개의 레이어를 가지는 fully connected network 이며, 은닉 층의 크기는 각각 256, 128 이다. 두번째 모델은 4 개의 레이어를 가지는 fully connected network 이며, 은닉 층의 크기는 각각 512, 256, 128 이다. 두번째 모델에서는 batch normalization 과 drouput 또한 사용하였다.

LSTM 모델과 Fully Connected Neural Network 모델 모두 출력 층이 아닌 Linear (Fully Connected) 레이어에는 LeakyReLU 를 활성화 함수로 (activation function) 사용하였다. 또한, 두 종류의 모델 모두 Adam (betas = (0.5, 0.9)) optimizer 를 사용했고, learning rate 는 0.005 로 동일하게 설정했다. 손실 함수로는 (loss function) 평균제곱오차 (Mean Squared Error, MSE)를 사용했다. 학습 에포크로는 LSTM 모델 중 큰 모델의 경우 100, 나머지는 50 을 사용했다.

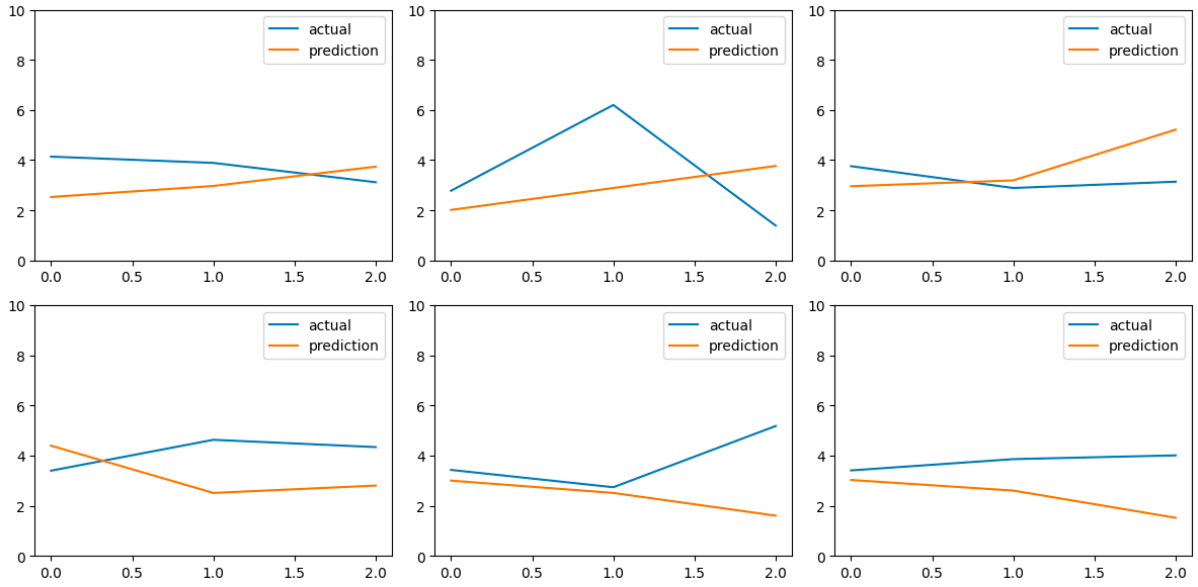
Results

1. LSTM Model 1 (Training data created through first 6 years)

1-1) ERA 예측

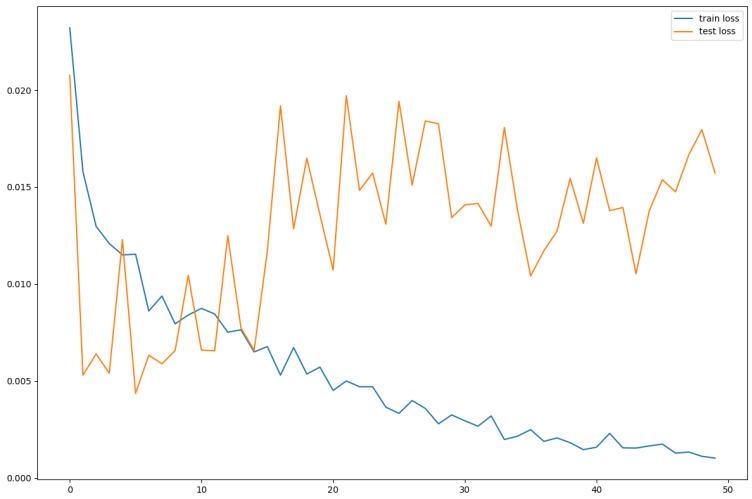


<Loss curve>

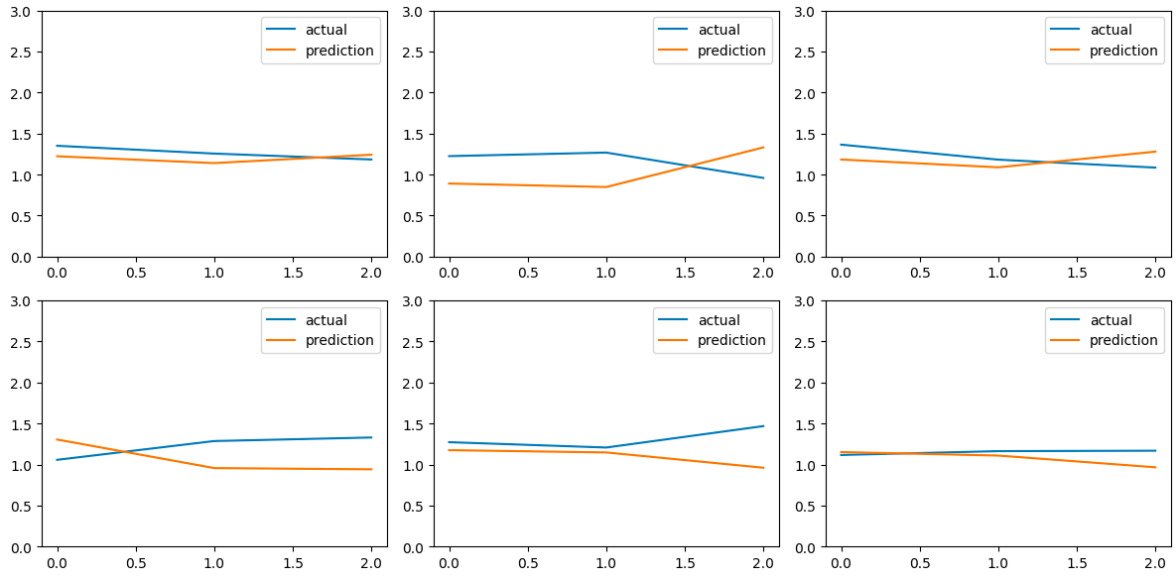


<Test data prediction>

1-2) WHIP 예측



<Loss curve>



<Test data prediction>

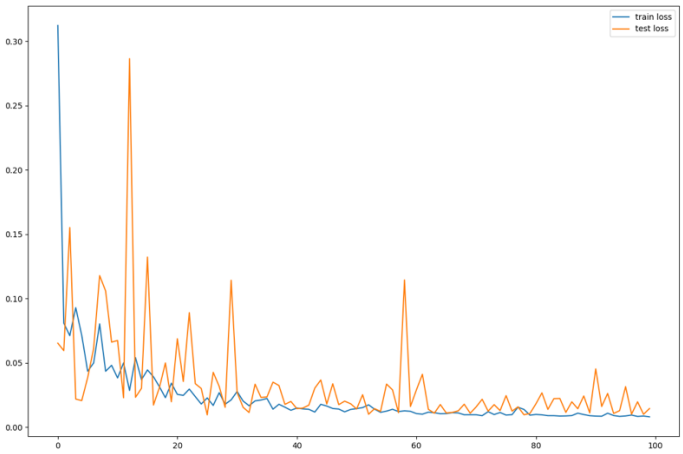
1-3) 결과 분석

두 지표에 대한 학습을 시켰을 때, 모두 loss curve 를 보면 훈련 데이터셋이 아닌 데이터셋에서의 loss 값이 훈련 데이터셋보다 높고, converge 되지 않는 모습을 보인다. 또한, 테스트 데이터셋에 대해 모델이 예측한 결과를 눈으로 확인해보면, 실제 데이터와 다른 양상으로 결과를 예측하는 모습을 보인다. 이와 달리, 훈련

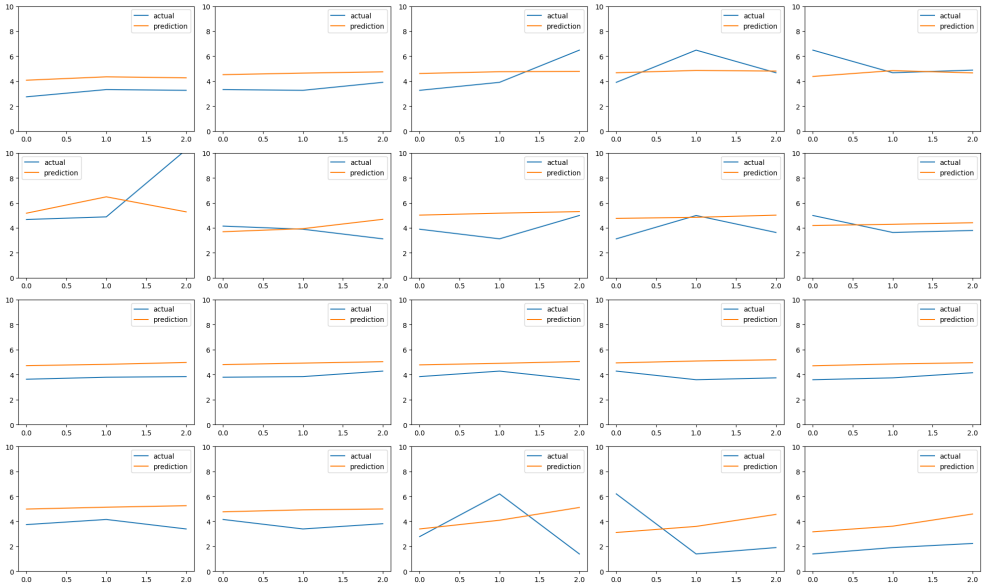
데이터셋에 대한 예측은 매우 정확한 양상을 보여주는데, 이는 모델이 훈련 데이터셋에 대해 오버피팅 되었음을 알 수 있다. 이러한 모습은 loss curve 에서 테스트 데이터셋에 대한 loss 값이 높아지기 전에 학습을 종료 시켜도 동일하게 나타났다. 즉, 모델이 데이터에 대해 일반화 시켜 학습시키는 것에 실패하는 것으로 보인다.

2. LSTM Model 2 (Training data created through sliding window)

2-1) ERA 예측

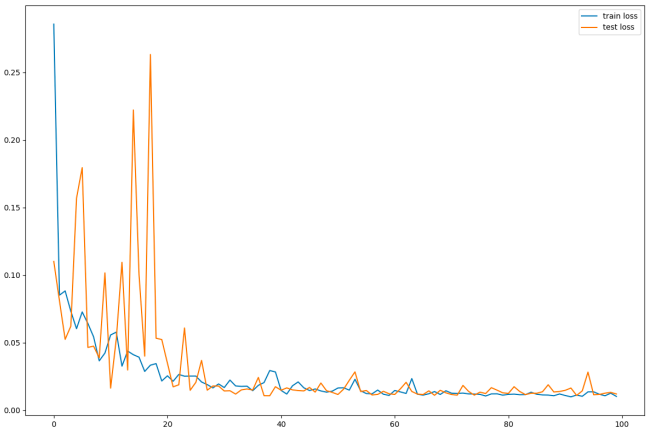


<Loss curve>

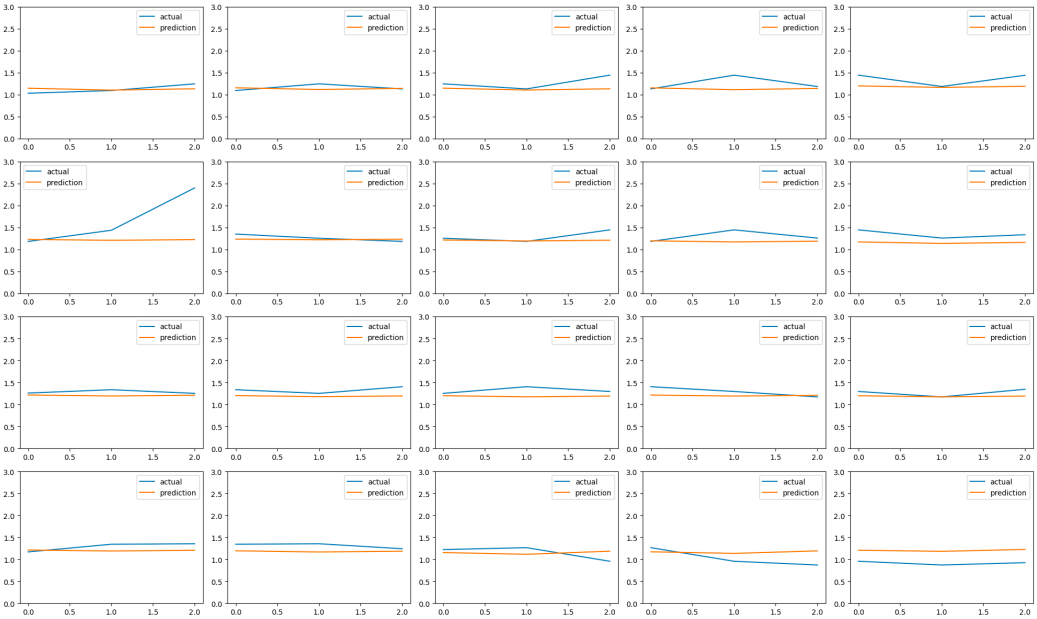


<Test data prediction>

2-2) WHIP 예측



<Loss curve>



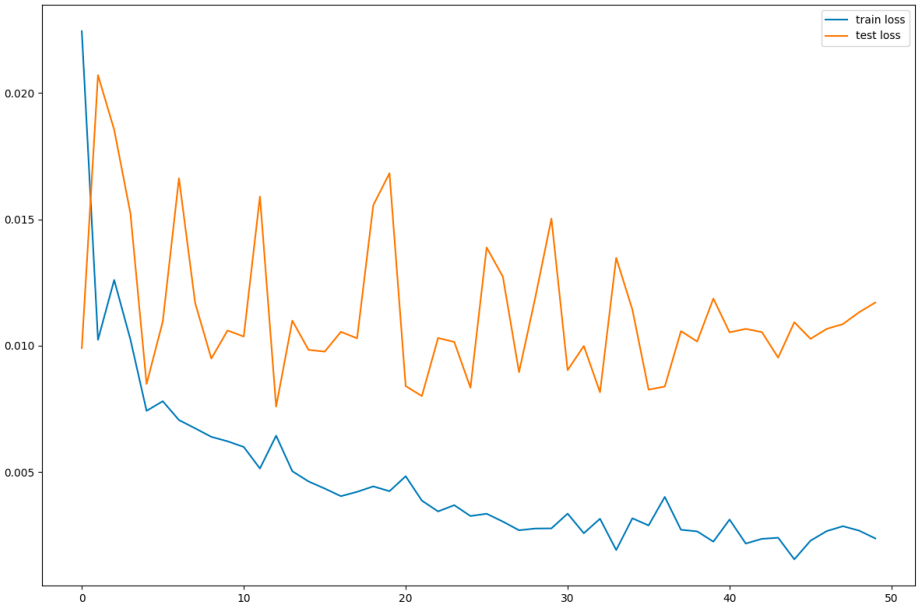
<Test data prediction>

2-3) 결과 분석

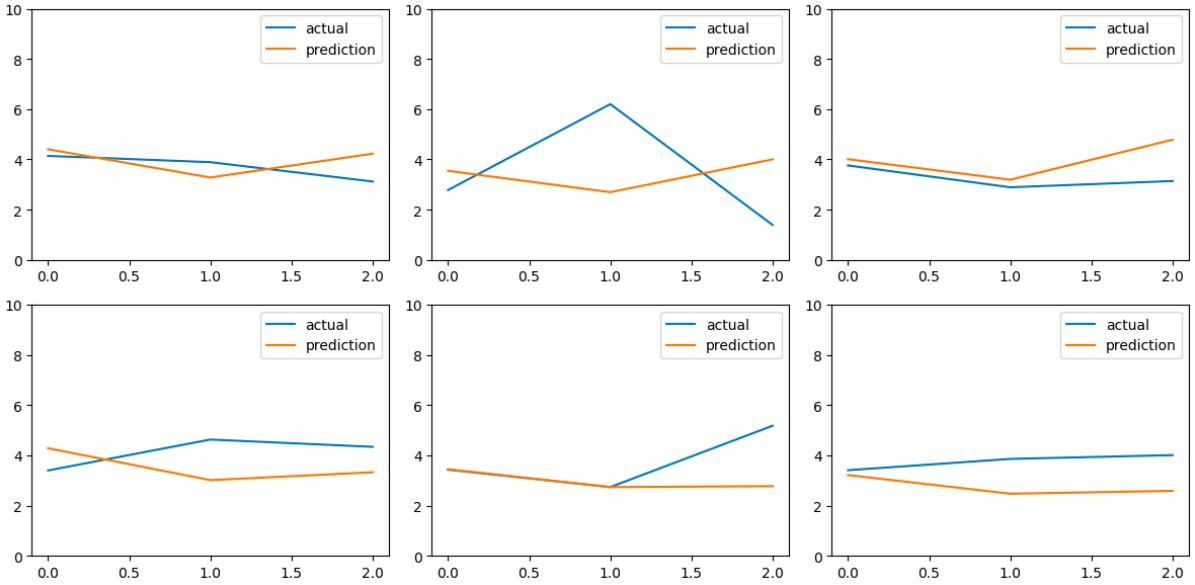
두 지표에 대한 학습을 시켰을 때, loss curve 를 보면 훈련이 잘 되는 것처럼 보인다. 그러나, 테스트 데이터셋에 대해 모델이 예측한 결과를 확인해보면, 입력 데이터에 관계없이 모델이 출력으로 항상 비슷하거나 동일한 출력을 내는 것을 알 수 있다.

3. FC Model 1 (Training data created through first 6 years)

3-1) ERA 예측

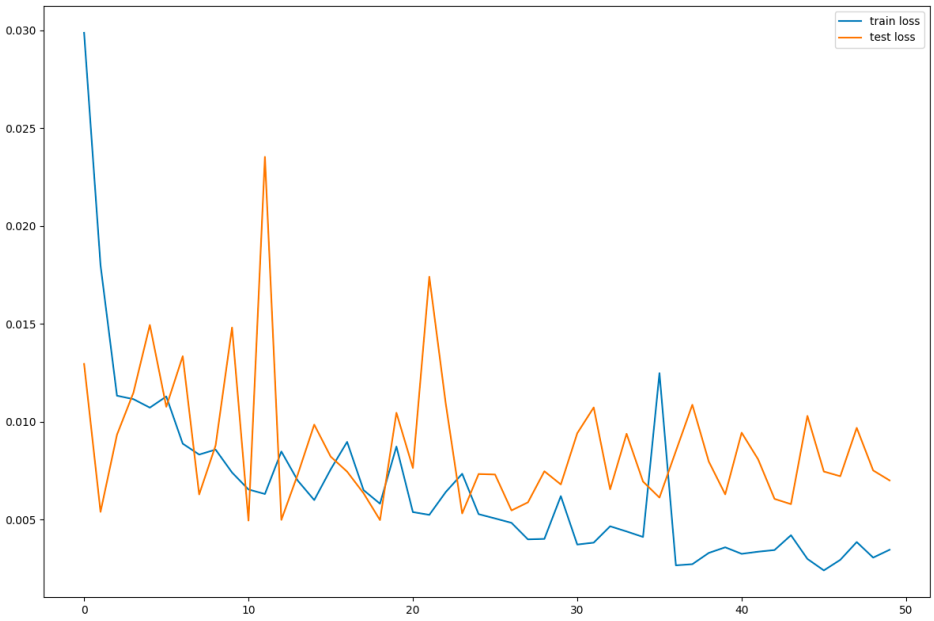


<Loss curve>

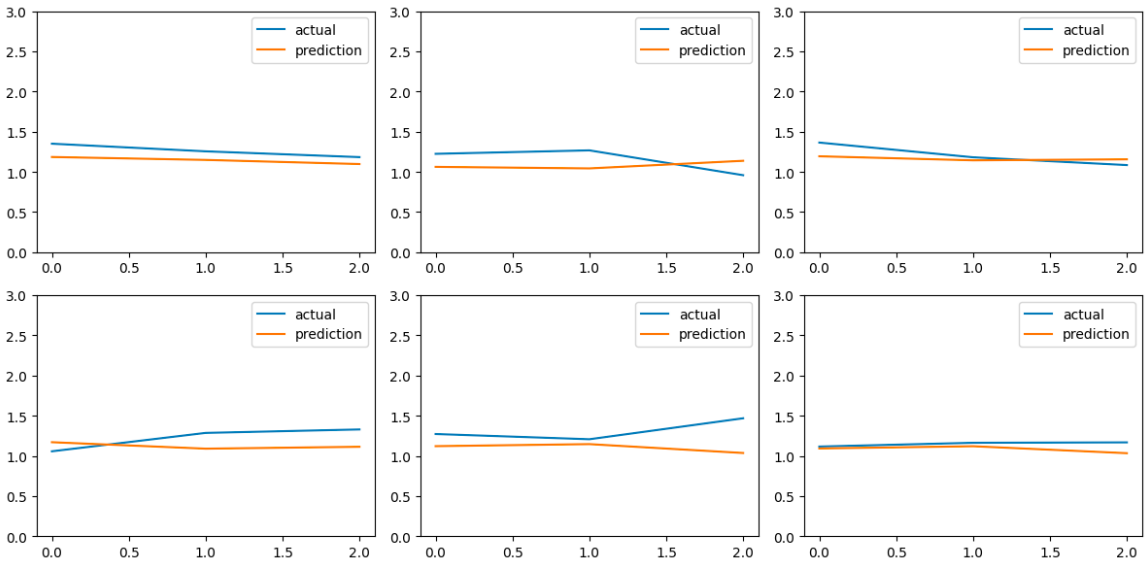


<Test data prediction>

3-2) WHIP prediction



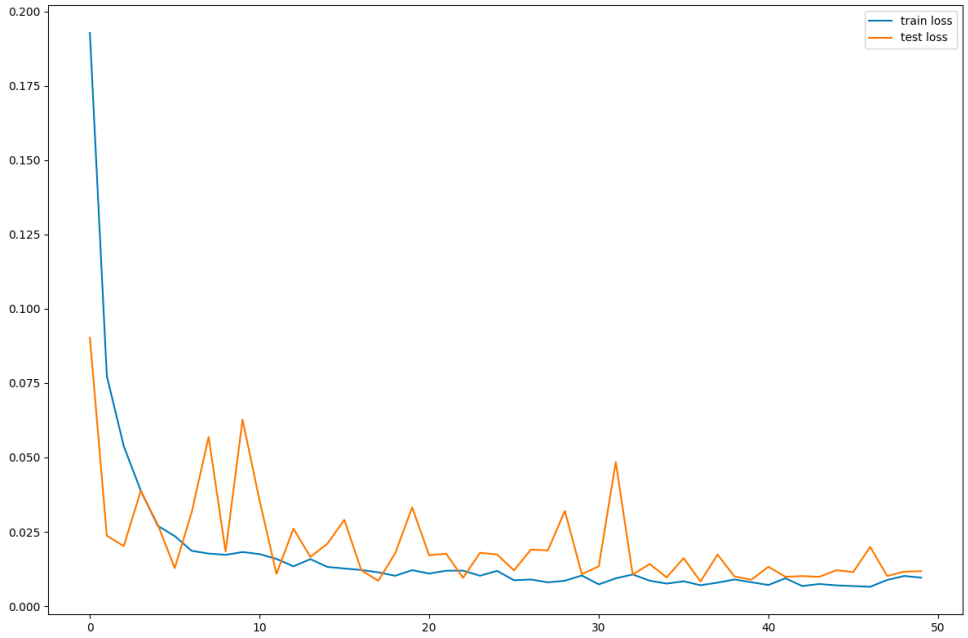
<Loss curve>



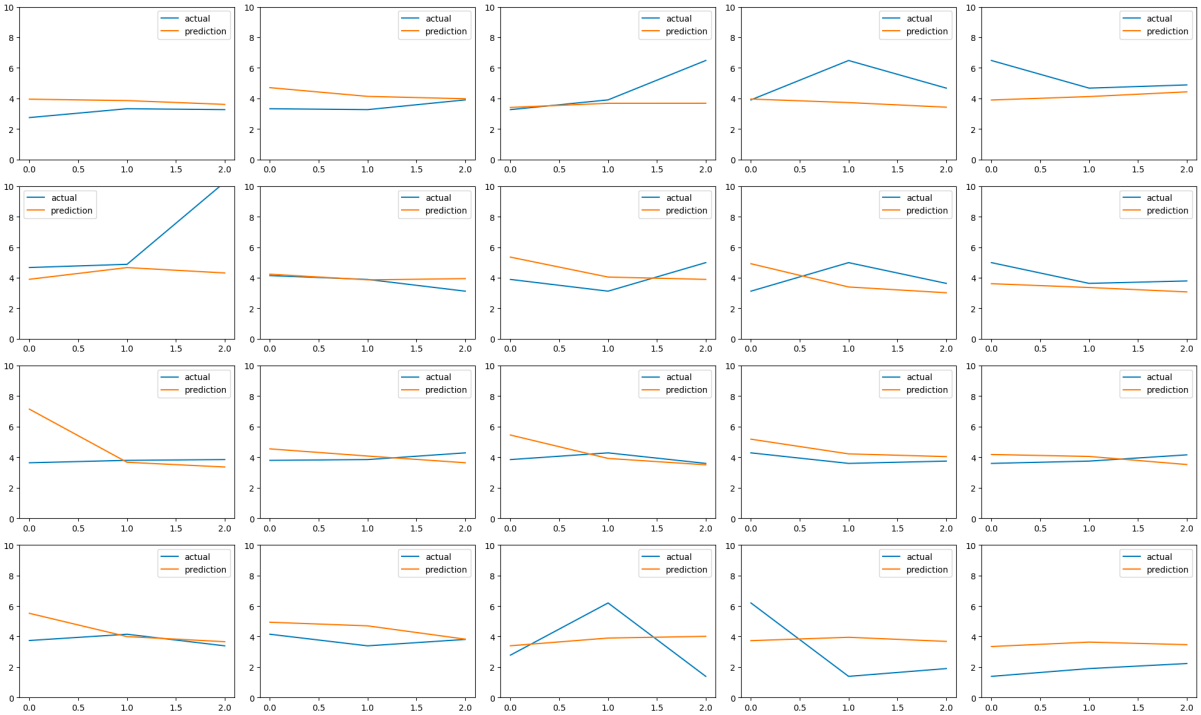
<Test data prediction>

4. FC Model 2 (Training data created through sliding window)

4-1) ERA 예측

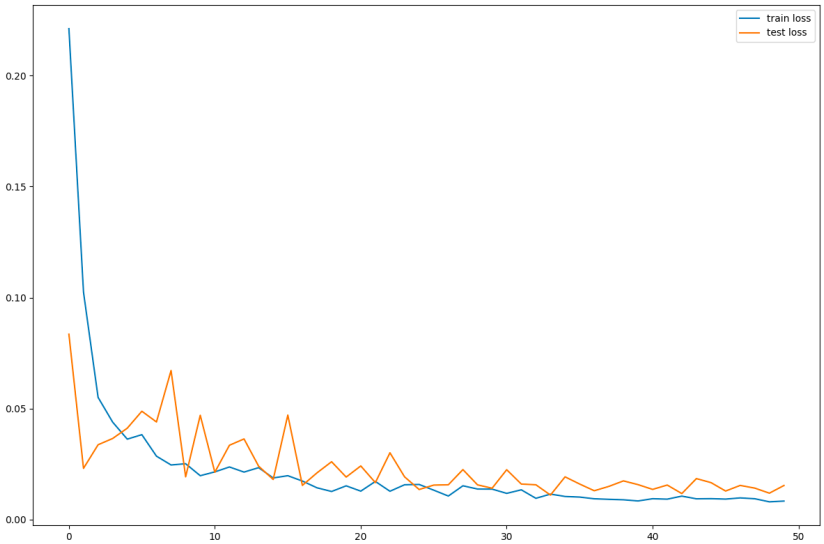


<Loss curve>

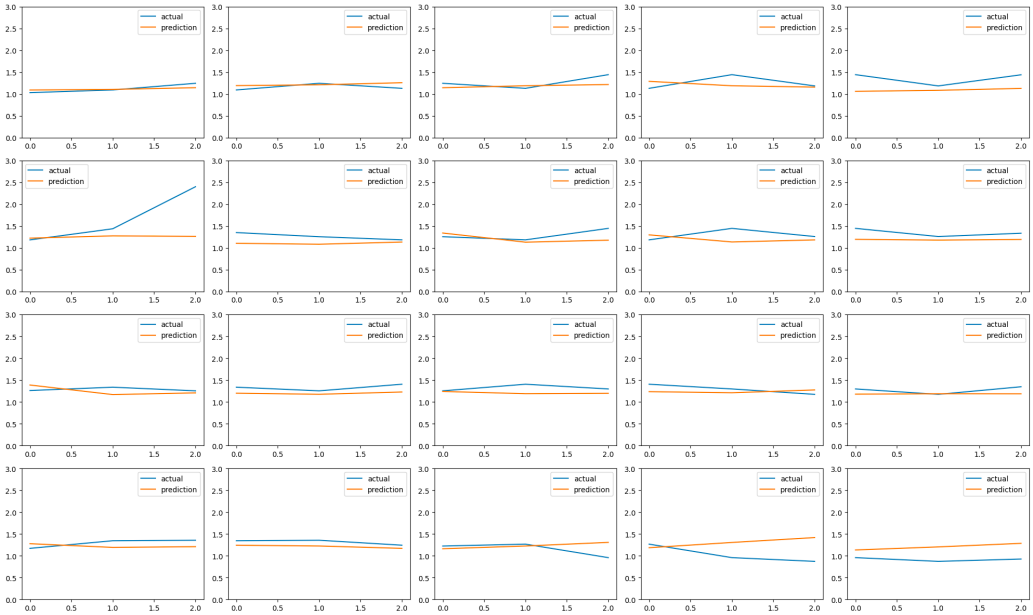


<Test data prediction>

4-2) WHIP 예측



<Loss curve>



<Test data prediction>

4-3) 결과 분석

Fully Connected Network 모델 모두 LSTM 모델과 동일한 양상을 보인다. 초기 6 년만을 이용한 데이터에 대해서는 훈련 데이터셋에 오버피팅 되고, 일반화 성능이 떨어진다. Sliding window 를 통해 만든 데이터셋에 대해서는, 모델이 입력에 관계없이 항상 비슷한 결과를 생성한다.

Discussion

1. 시행 결과

   LSTM 모델과 FC 모델 모두 각각의 데이터에 대해 비슷한 양상을 보여주었다. 두 모델 모두 데이터셋에 대해 일반화하여 학습하는 것에 실패한 것으로 보인다. 어쩌면 3 년간의 지표 데이터와 추후 3 년간의 지표 간의 관계성이 없는 것일 수 있다.

2. 데이터 문제

   타 스포츠에 비해선 많은 지표들이 계산, 기록되는 야구지만 Time Series 로 볼 정도로 장기간 꾸준하게 선수생활을 이어간 선수가 많지 않다는 점과 그런 선수의 기록이라고 하더라도 데이터의 구조가 시즌단위로 나뉘어져 있어 길이가 20 년을 넘지 못하기 때문에 인공지능 모델에 학습시키기에는 턱없이 부족함

   또한 예측하고자 했던 ERA, WHIP 와 같은 지표가 선수들 간의 편차가 크지 않아 데이터로부터 학습을 진행할 때 모델이 수렴하는 정도가 미흡하다는 문제도 있었다.

Data availability

https://github.com/robinjoo1015/big-data-2023-2

Reference

Hsuan-Cheng Sun, Tse-Yu Lin and Yen-Lung Tsai. Performance Prediction in Major League Baseball by Long Short-Term Memory Networks 2022