서울시립대학교 2017년도 2학기

컴퓨터과학부

2012920053 정주안

# 통계데이터마이닝 – 박창이 교수님 \_KOSPI data에 대한 PCA 및DNN 학습

1. 요약
   1. 프로젝트를 위한 데이터로 KOSPI 주식 데이터를 선택했다. 주식 데이터를 이용해 적절한 회귀모형을 만들기 위해, a – 14 일 ~ a - 1 일 데이터에 대한 Volume 과 전날 대비 주가 데이터를 이용해 a일의 주가를 predictor로 이용하는 데이터 구조를 만들고 그 주성분을 추출해 차원을 축소시키기 위해 PCA를 진행했다. 그후 DNN 회귀 모델을 학습시켜보았다.
2. 데이터 전처리
   1. Python의 pandas API를 이용해 yahoo finance server에서 KOSPI전체에 대한 일별 데이터를 받았다. 다음 코드는 그 과정의 일부이다.

|  |
| --- |
| 1. for li in KOSPIList[1:]:  2.    print(li)  3.    try:  4.        tmpDf = listModule.data.DataReader(  5.            li,  6.            'yahoo',  7.            listModule.START  8.        )  9.    except listModule.\_utils.RemoteDataError  as RDE:  10.        print("ERROR with " + li + ": " + RDE)  11.        continue  12.        pass  13.    else:  14.        tmpDf.to\_csv(listModule.dataPath + li + '.csv')  15.        gotList.append(li + '.csv') |

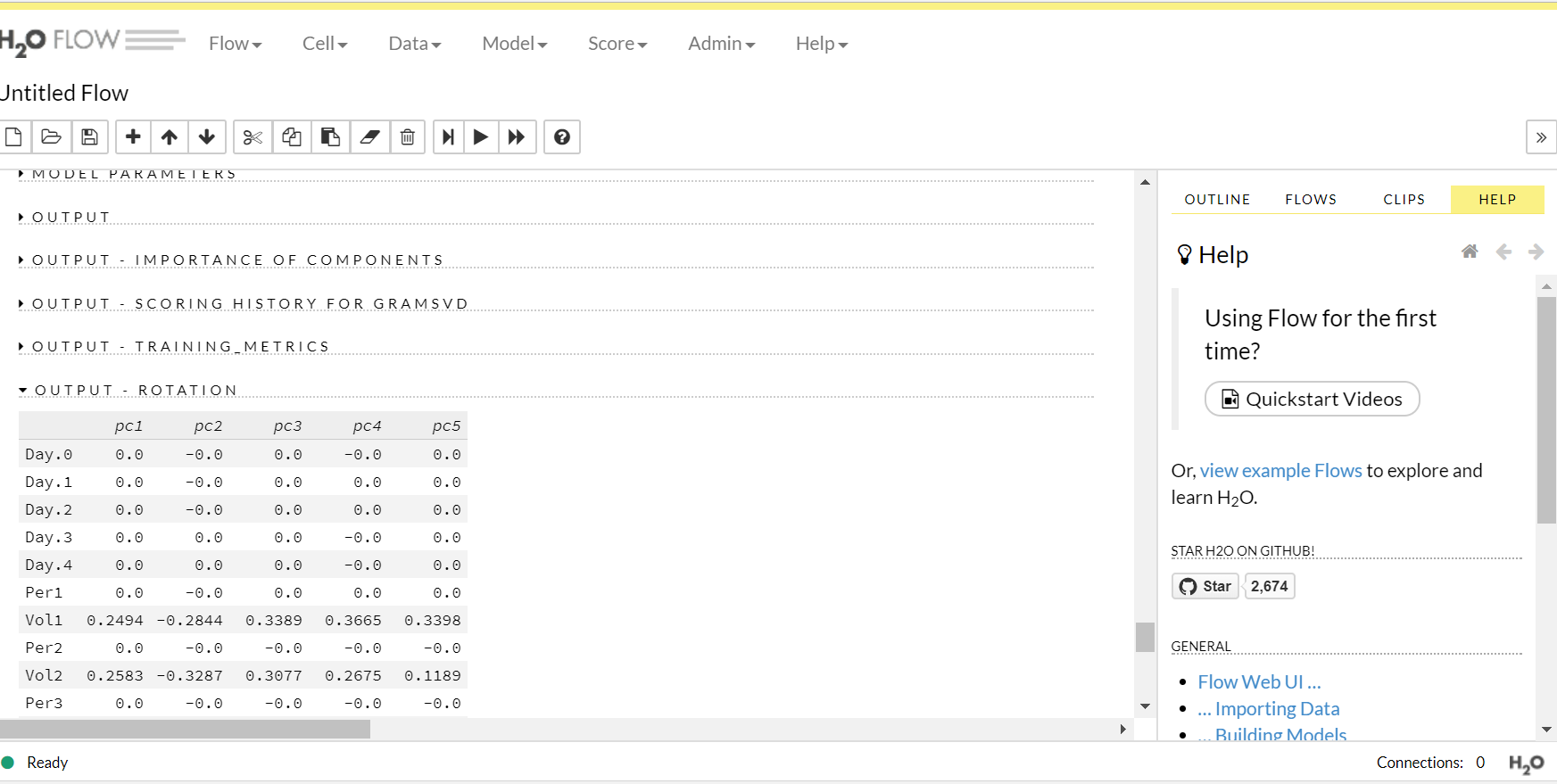
이에 대한 전체 코드는 project gitub [1]/Data preprocess/getData.py 에서 참조할 수 있다.

* 1. 이 데이터를 14일을 기준으로 Shingling하고, 각 날짜의 거래 Volume과 전날대비 주가수치, 그리고 14일의 다음날의 전날대비 주가수치를 계산했다.

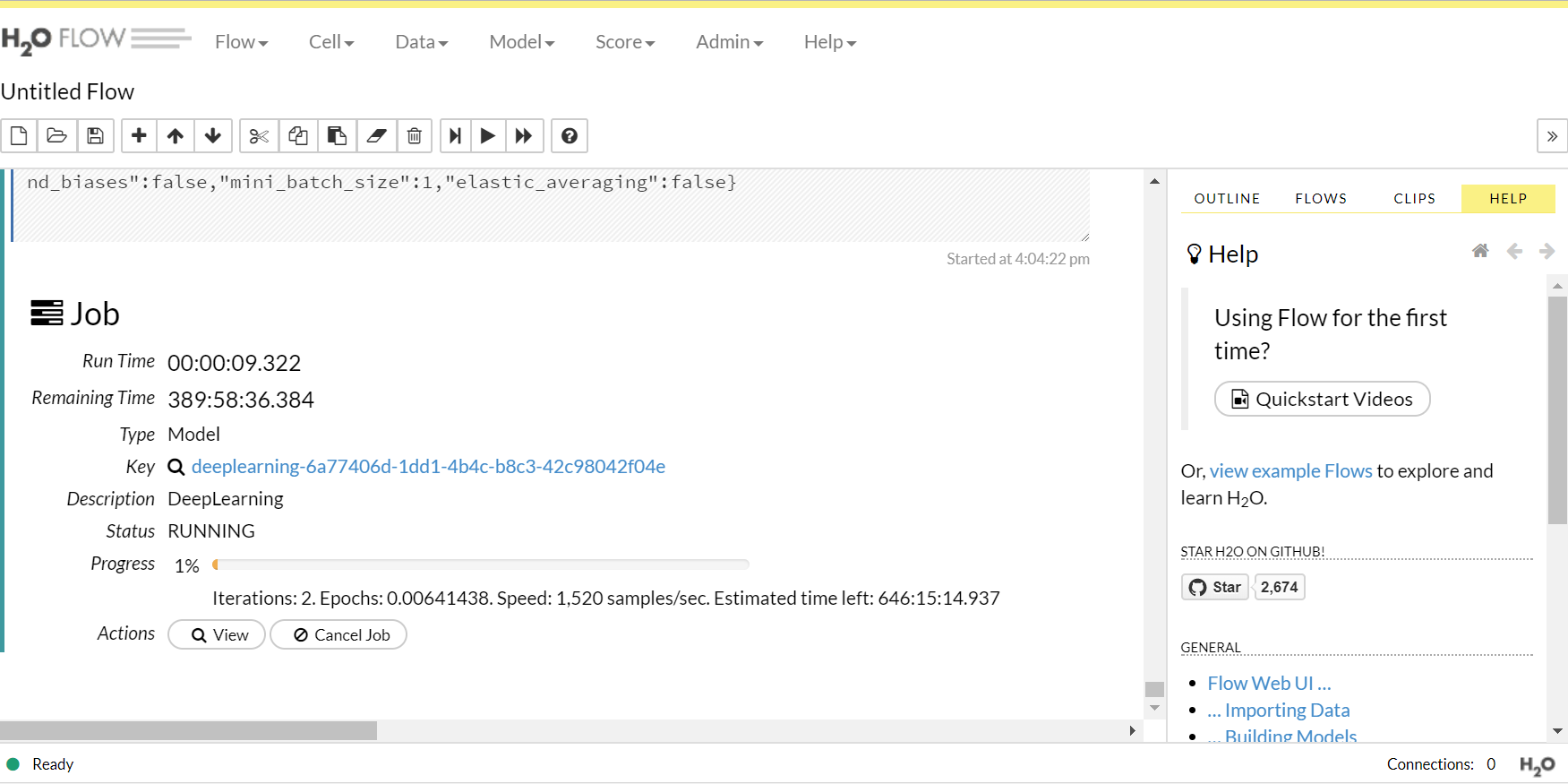
|  |
| --- |
| def getClosePerOpenAndVol(code, cmpBase):     tmpKOSPIData = LM.csv.reader(open(LM.dataPath + code))     tmpValueList = []     KOSPIDataList = list(tmpKOSPIData)[1:]     for idx, val in enumerate(KOSPIDataList):         try:             if val[1] != '':                 tmpPer = ((float(val[4]) - float(val[1]))/float(val[1]))                 tmp = list()                 tmpI = 0                 if (idx - LM.N >= 0):                     tmpI = idx - LM.N                     for i in range(tmpI, idx):                         if KOSPIDataList[i][1] == '':                             break                         # if len(KOSPIDataList[i]) == 7:                         #     tmpWeek = datetime.datetime.strptime(str(KOSPIDataList[i][0]), "%Y-%m-%d").weekday()                         #     KOSPIDataList[i].insert(1, DAY[tmpWeek])                         tmp.append([100\*((float(KOSPIDataList[i][4]) - float(KOSPIDataList[i][1]))/float(KOSPIDataList[i][1])), KOSPIDataList[i][6]])                     if len(tmp) == 14:                         tmp.append(100\*tmpPer)                         tmp.append(datetime.datetime.strptime(str(val[0]), "%Y-%m-%d").weekday())                         tmpValueList.append(tmp)         except ValueError as e:             print(e)     return tmpValueList |

이에 대한 전체 코드는 project gitub [1]/Data preprocess/classifyAndSaveModule.py 에서 참조할 수 있다.

* 1. R 환경에서 구동가능한 Machine learning library인 H2O를 이용해, PCA를 진행하여 다음과 같이 5개의 주성분을 얻었다.

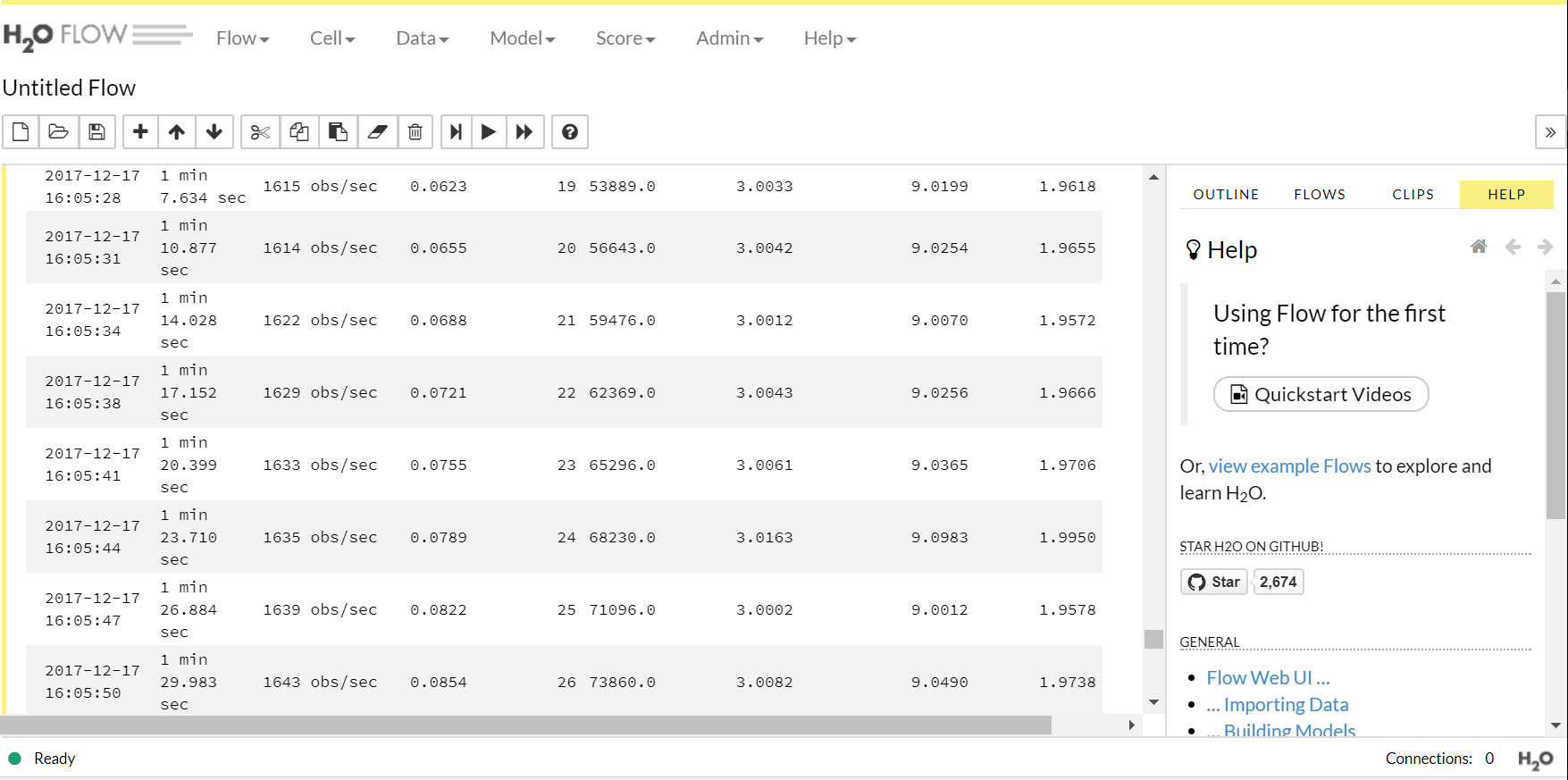


* 1. H2O에서 지원하는 DNN을 이용해 다음과 같이 학습하였다.

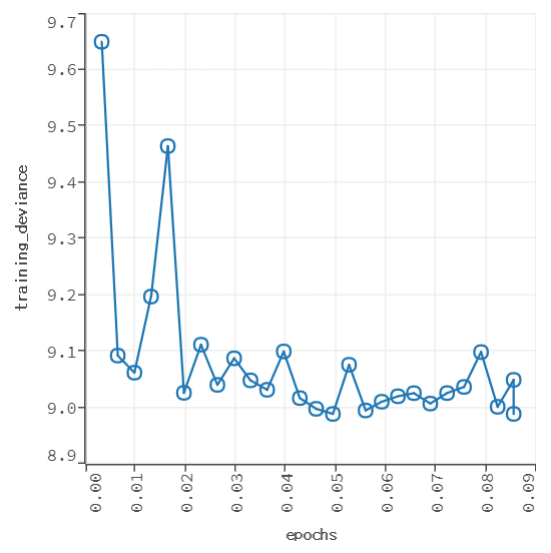


이때 사용한 네트워크는 maxout with dropout / size: 256\*512\*128 의 일반적인 DNN이다.

* 1. 다음은 학습결과 Iteration에 대한 사진이다.



* 1. 다음은 학습결과 Training deviance-epochs의 그래프는 다음과 같다.



그래프에서 알 수 있듯이, 학습이 특별한 효과없이 종료되었다. RMSE 또한 3.1 ~ 2.9를 왔다갔다 하는 형상을 보이며 그 이상의 학습은 진행하지 못했다.

1. 결론  
    14일간의 거래량과 전날 대비 주가 데이터를 이용해 다음 날의 주가를 예측하는 DNN 모델을 만들어보았다. KOSPI 데이터를 얻어 원하는 형태로 전처리를 진행한 후, PCA를 이용해 데이터의 차원을 축소시켰다. 그 결과를 DNN을 이용해 학습시켰으나 특별한 진전없이 학습이 종료되었다. 이는 애초에 데이터의 x-y 관계가 규칙적이지 않기 때문으로 보인다. 즉 14일간의 거래량과 전날대비 주가 데이터로는 다음 날의 주가를 예측하기 어려운 것으로 결론지을 수 있다.
2. 참고  
   [1] Project GITHUB: <https://github.com/johnnyapu15/DataMiningGrad2017Proj>