개발 과정 도식

따릉이 수요 예측을 위한 머신러닝 프로그램 구축

대한상공회의소 김덕주

목표:
주요 개발 능력 구현 및 자전거 수요예측

- 파이썬 기반
- 데이터 전처리
  - XML(파싱, 웹데이터 분석),CSV 입.출력, 슬라이싱&조인
- 텐서플로 라이브러리 활용
  - 흐름 구성, GRADIENT DESCENT(경사 하강법) 활용, SESSION 실행, 모델저장
- 머신러닝:
  - KNN 활용 클러스트링(=비슷한 유형의 정류장을 유형화
- 차원축소
- 정규화
- 시그모이드 활용
- 신뢰도 평가
- 외부 라이브러리 구성 및 라이선스 확인

프로그램 동작 과정

데이터 크롤링

- 수집 대상 선정 고려 요소:자전거 임대에 영향을 주는 요소
  - → 논문 참조:Dimitrios Efthymiou(2013),도명식(2014) 등
- 수집 데이터 항목 선정:
  - 기온,습도,강수량,평균풍속,미세먼지 농도,휴일여부,지역
  - 지역은 중랑구로 한정
- 수집 출처 고려요소: 데이터의 신뢰성, 처리용이성
  - 기상청,공공데이터포털, 서울 열린 데이터광장 활용

데이터 수집

OpenApi,urllib을 활용한
XML형식의 데이터 호출

차후 인증키 만료 주의필요

Beautifulsoup4 활용

기존 라이브러리를 활용해
받아온 데이터를 가공한다.

Csv 함수를 활용한
전처리용 .csv 파일 생성

데이터 전처리

데이터 스플릿

필요없는 문자열을 제거한다.

자료 변환

시그모이드 함수:

- 기계오류등으로 인한 극단값을 배제한다.

보통화

데이터를 0~1 값을 가지도록 변형함

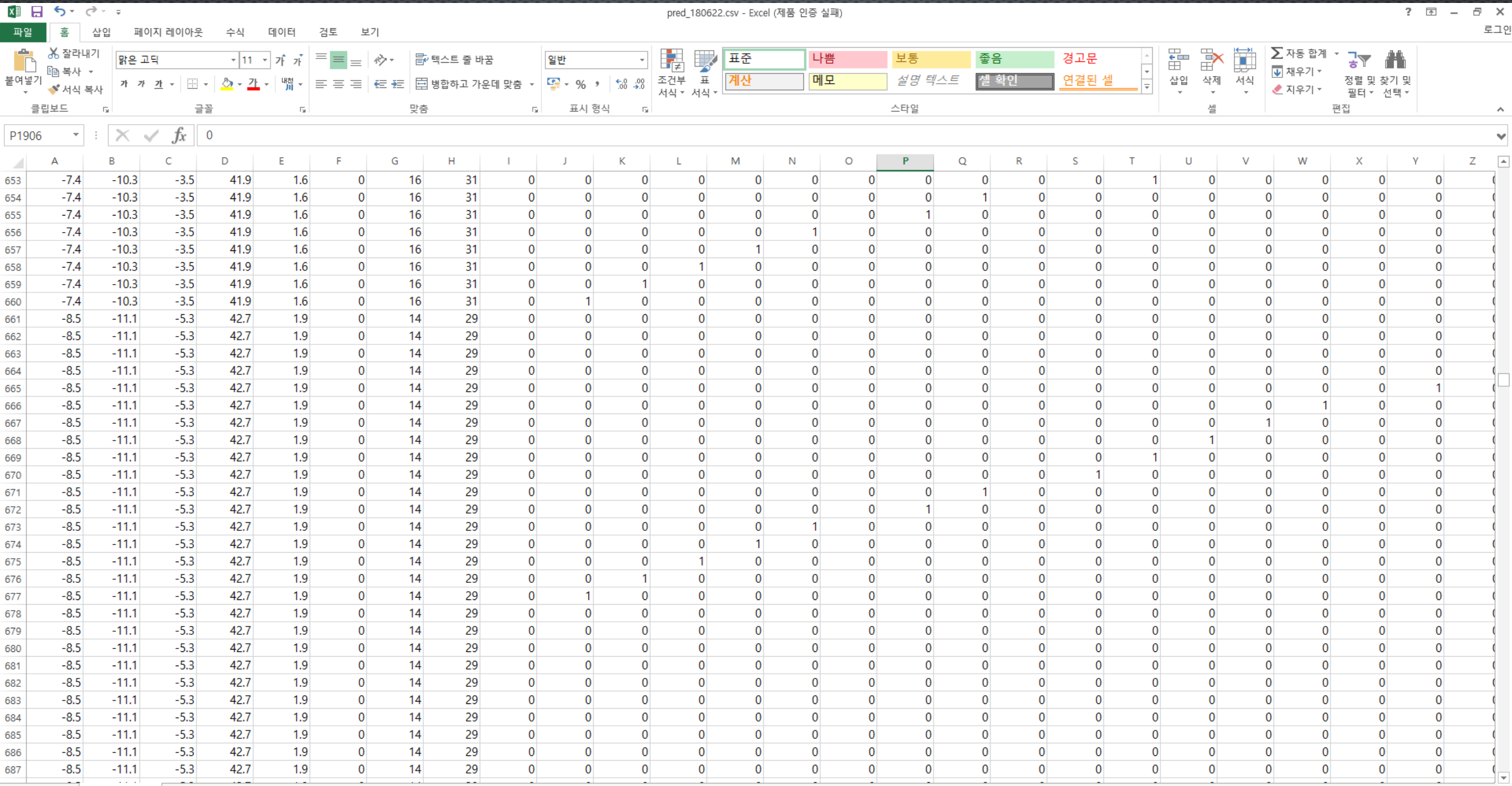
차원 압축

직교하는 데이터를 찾아 최적화

Sklearn의 PCA활용

동작원리:

행렬의 고유값을 활용

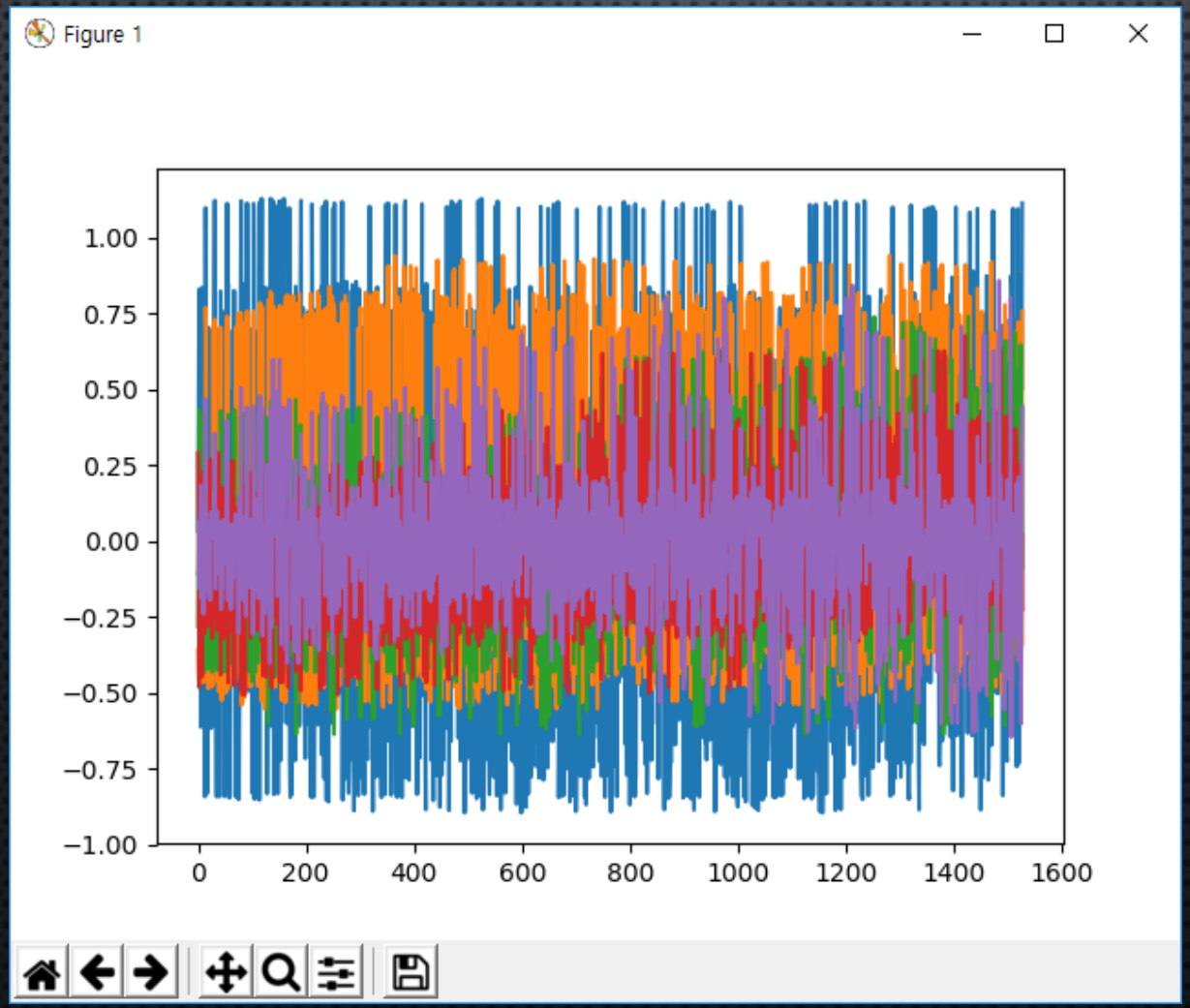


	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
653	-7.4	-10.3	-3.5	41.9	1.6	0	16	31	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0
654	-7.4	-10.3	-3.5	41.9	1.6	0	16	31	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
655	-7.4	-10.3	-3.5	41.9	1.6	0	16	31	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0
656	-7.4	-10.3	-3.5	41.9	1.6	0	16	31	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
657	-7.4	-10.3	-3.5	41.9	1.6	0	16	31	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
658	-7.4	-10.3	-3.5	41.9	1.6	0	16	31	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
659	-7.4	-10.3	-3.5	41.9	1.6	0	16	31	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
660	-7.4	-10.3	-3.5	41.9	1.6	0	16	31	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
661	-8.5	-11.1	-5.3	42.7	1.9	0	14	29	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
662	-8.5	-11.1	-5.3	42.7	1.9	0	14	29	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
663	-8.5	-11.1	-5.3	42.7	1.9	0	14	29	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
664	-8.5	-11.1	-5.3	42.7	1.9	0	14	29	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
665	-8.5	-11.1	-5.3	42.7	1.9	0	14	29	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0
666	-8.5	-11.1	-5.3	42.7	1.9	0	14	29	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0
667	-8.5	-11.1	-5.3	42.7	1.9	0	14	29	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
668	-8.5	-11.1	-5.3	42.7	1.9	0	14	29	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
669	-8.5	-11.1	-5.3	42.7	1.9	0	14	29	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0
670	-8.5	-11.1	-5.3	42.7	1.9	0	14	29	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0
671	-8.5	-11.1	-5.3	42.7	1.9	0	14	29	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
672	-8.5	-11.1	-5.3	42.7	1.9	0	14	29	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0
673	-8.5	-11.1	-5.3	42.7	1.9	0	14	29	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
674	-8.5	-11.1	-5.3	42.7	1.9	0	14	29	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
675	-8.5	-11.1	-5.3	42.7	1.9	0	14	29	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
676	-8.5	-11.1	-5.3	42.7	1.9	0	14	29	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
677	-8.5	-11.1	-5.3	42.7	1.9	0	14	29	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
678	-8.5	-11.1	-5.3	42.7	1.9	0	14	29	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
679	-8.5	-11.1	-5.3	42.7	1.9	0	14	29	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
680	-8.5	-11.1	-5.3	42.7	1.9	0	14	29	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
681	-8.5	-11.1	-5.3	42.7	1.9	0	14	29	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
682	-8.5	-11.1	-5.3	42.7	1.9	0	14	29	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
683	-8.5	-11.1	-5.3	42.7	1.9	0	14	29	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
684	-8.5	-11.1	-5.3	42.7	1.9	0	14	29	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
685	-8.5	-11.1	-5.3	42.7	1.9	0	14	29	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
686	-8.5	-11.1	-5.3	42.7	1.9	0	14	29	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
687	-8.5	-11.1	-5.3	42.7	1.9	0	14	29	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

차원압축

- 개념: 벡터값을 행렬의 고유값을 이용해 차원을 축소하는것
- 적용 알고리즘:PCA
- 적용 고려 대상: 기온(평균,최소,최대), 미세먼지(보통,초미세)
- 효용: '차원의 저주' 극복; 계산 속도 및 정확도 향상,다양한 피쳐 활용 가능
- 실제 라이브러리: PCA(SKLEARN.DECOMPOSTION) OR TENSORFLOW.SQUEEZE()

차원 압축된 데이터의 분포: 5개로 압축



정제된 데이터

- 1909개의 데이터 추출
- 피쳐(독립변인)을 (8+정류소 개수)차원에서 5차원으로 압축
- 5계층으로 나누어 1단위를 검증용으로 활용
- 레이블(종속요인);YDATA; 는 대여와 반납 중 대여만 활용

학습 간 조정 요소

알파값

- 최적값 구하는 것이 학술적으로 불가능
- 임의의 횟수에 의한 개선 추구

학습횟수

10만회 기준

다음 1000회차 시행시 cost값의 0.0001 미만인 경우 중단

학습:

가설함수 h(x)=wx+b

하강경사법 활용

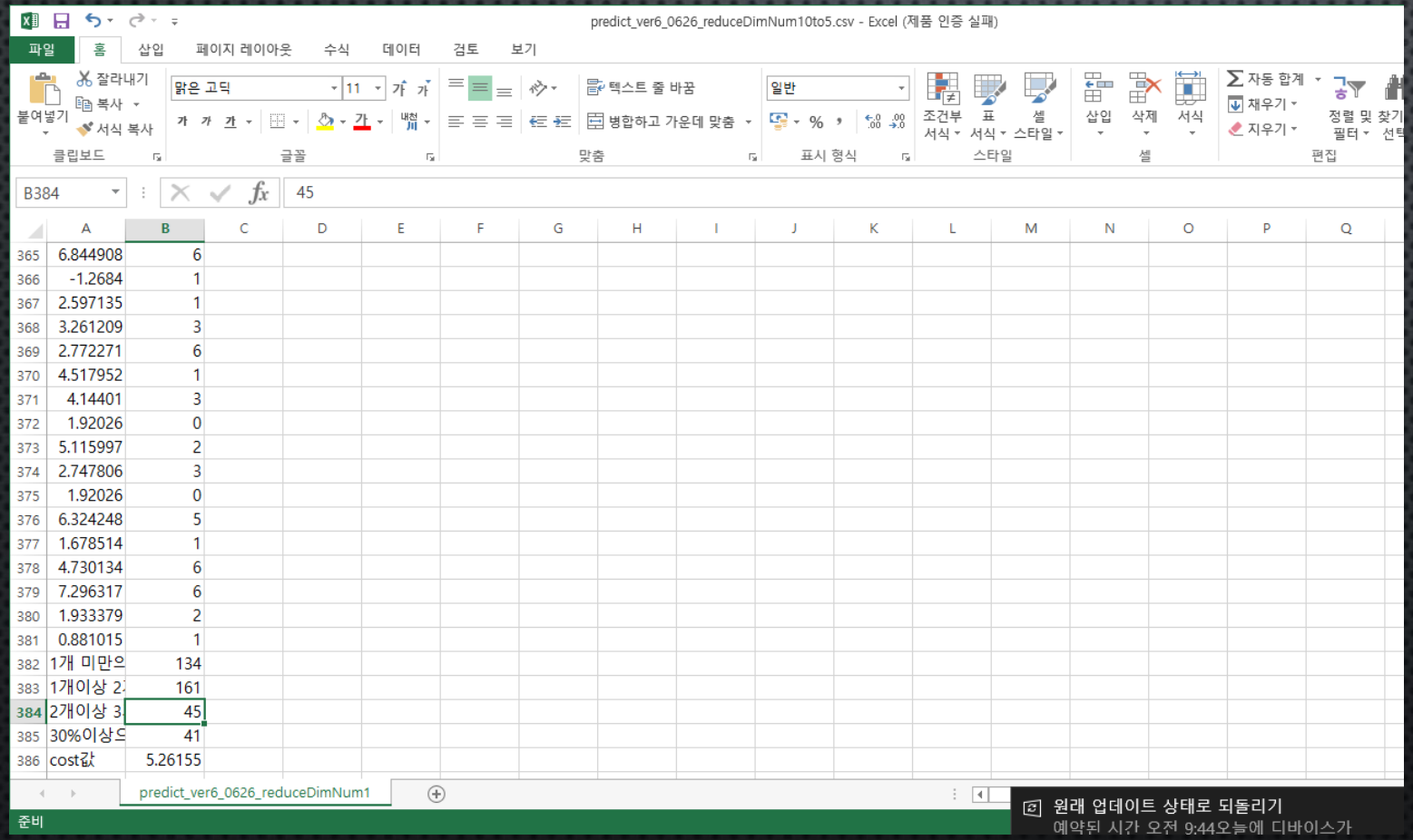
차원축소: 축소된 차원의 수

최적값을 찾기 위해서 조정

결과:출력되는 값들

- Cost
- 테스트 결과 예측값과 레이블 비교

결과



	A	B
365	6.844908	6
366	-1.2684	1
367	2.597135	1
368	3.261209	3
369	2.772271	6
370	4.517952	1
371	4.14401	3
372	1.92026	0
373	5.115997	2
374	2.747806	3
375	1.92026	0
376	6.324248	5
377	1.678514	1
378	4.730134	6
379	7.296317	6
380	1.933379	2
381	0.881015	1
382	1개 미만의	134
383	1개이상 2	161
384	2개이상 3	45
385	30%이상의	41
386	cost값	5.26155

381개 중 295개가 2개 미만의 차이를 보임

bike [D:₩bike] - ...₩rental_pretreatment₩mini_tensor.py [bike] - PyCharm

File Edit View Navigate Code Refactor Run Tools VCS Window Help

bike 〉 rental_pretreatment 〉 mini_tensor.py 〉

mini_tensor

Project

pretreat.py
pyplot_boxplot.py
result_0625
seoulDailyCli.csv
switch.py
대여소 정보 위도경도.csv
대여소별 대여내역(2017년)
연습장
rough
find_kisangchung.py
remove_outlier.py
venv library root
Include
Lib
Scripts
pip-selfcheck.json
pyvenv.cfg
주요문서
~$개발과정.pptx
개발 애로사항
개발과정.pptx
데이터 전처리에 있어서
라이선스 관련 파일.docx
모델학습결과
버그잡기.txt
업그레이드 가능 부분
유의사항

mini_tensor.py | 업그레이드 가능 부분 | pyplot_boxplot.py | pca_func.py | mini_pretreated.py | parctice.py

```
157  bikeCount=sess.run(hf,feed_dict={x:x_test_final})
158
159
160  #
161  ##########기록부
162  output_file = "d:/bike/datasource/predict_ver6_0626_reduceDimNum10to5.csv"
163  filewriter = open(output_file, 'w',newline="")
164  csvWriter = csv.writer(filewriter)
165  count=0
166  count2=0
167  count3=0
168  neg_count=0
169  for i in range(bikeCount.shape[0]):
170      if abs(bikeCount[i][0]-y_test_final[i][0]) <1:
171          count+=1
172      elif abs(bikeCount[i][0]-y_test_final[i][0]) <2:
173          count2+=1
174      elif abs(bikeCount[i][0]-y_test_final[i][0]) <3:
175          count3+=1
176      elif abs(bikeCount[i][0]-y_test_final[i][0])/y_train_final[i][0] >1/4:
177          neg_count+=1
178
179      print(round(bikeCount[i][0]),y_test_final[i][0])
180      csvWriter.writerow([bikeCount[i][0],y_test_final[i][0]])
181  print("count=",count)
182  csvWriter.writerow(["1개 미만의 차이로 맞은 갯수",count])
183  csvWriter.writerow(["1개이상 2개 미만의 차이로 맞은 갯수",count2])
184  csvWriter.writerow(["2개이상 3개 미만의 차이로 맞은 갯수",count3])
185  csvWriter.writerow(["3X이상으로 빗나간 경우",neg_count])
```

for row in x_test

모델학습결과

```
107
108  ----------------------------
109  텐서보드 참조하기
110
111  0626
112  학습사안:
113  그래프는 완벽하게 2차원 그래프를 그린다.
114  알파값은 작을수록 세밀해지나 연산에 필요한컴퓨팅 파워가 기하급수적으로증가
115
116  변경사항
117  데이터 전처리:
118  데이터폴딩->시그모이드->노말라이제이션->PCA
119  학습/평가     이상값 영향 최소화 -> 자전거 대여는 양수 또는 0이므로 배제
120
121  평가:
122  예측값과 레이블값을 비교하도록 설정함
123  평가기준
124
125  결과:
126
127  알파:0.001 & 학습횟수1000001, 차원축소:30개로
128  5만회의 학습 즈음 하였을때 코스트 5.260516즈음하여 고정
129
130  알파:0.0001 & 학습횟수1000001, 차원축소:30개로
131  10만번째에 5.2714753
132  138회 일치
133
134  차원축소 10개
135  100000 cost: 5.2714424
136  500000 cost 5.261073
```

Run mini_tensor

```
예측값
3678000 cost: 5.261073
예측값
3679000 cost: 5.261073
예측값
3680000 cost: 5.261073
예측값
3681000 cost: 5.261073
예측값
3682000 cost: 5.261073
예측값
3683000 cost: 5.261073
예측값
3684000 cost: 5.261073
예측값
```

Event Log

2018-06-25
오전 9:15 IDE and Plugin Updates: PyCharm is ready to update.

오전 11:09 Packages installed successfully: Installed packages: 'mglearn'

2018-06-26
오전 9:15 **IDE and Plugin Updates:** PyCharm is ready to update.

원래 업데이트 상태로 되돌리기
예약된 시간 오전 9:44오늘에 디바이스가 준비되어 있지 않아 Windows를 업데이트 하지 못했습니다.
예약된 시간에 디바이스의 전원이 꺼졌습니다.
지금 다시 시작 | 1시간 대기

IDE and Plugin Updates: PyCharm is ready to update. (today 오전 9:15)

59:19 CRLF UTF-8

느낀점

- 시그모이드 적용 후 정확도가 비약적으로 향상됨
  →OUTLIER에 의한 영향이 컸던것으로 추측됨

한계 및 개선사항

피쳐값 추가

1.정류소의 고도 참조, 성수씨 아이디어
토,일이 아닌 휴일 값 추가
그 전 날 대여횟수를 피쳐로 넣기

코딩
전처리부분에서 기록을 위한 idx 활용 for문을 csv를 활용으로 바꾼다.

정류장 개수에 대해서 원핫인코딩에서의 자동화를 통한 어떤 임의의 정류장 수라도 작동할 수 있도록 활용
출력 결과와 레이블(답) 비교의 히스토그램 그래프 그리기by pyplot
더많은 데이터 확보
이상값(outlier)의 완전 배제
더 많은 학습을 통한 효율성 개선
텐서보드를 활용한 더욱 상세한 학습과정 분석 가능
K-folding 방법의 정석적 활용을 통한 5단위 전부에 대한 검증
각 정류소별 특징을 클러스터링하여 시각화

주요 인용 라이브러리 라이선스

- 파이썬 및 내장 라이브러리: 오픈소스
- 사이킷런(SKLEARN):BCD 라이선스, 상업적 활용가능
  - HTTP://SCIKIT-LEARN.ORG/STABLE/INDEX.HTML
- 텐서플로(TENSORFLOW):APACHE 2.0라이선스
  - HTTPS://WWW.APACHE.ORG/LICENSES/LICENSE-2.0
- MATHPLOT: 단순 상표등록; 로열티 및 이용 제약사항 일절없음
  - HTTPS://MATPLOTLIB.ORG/USERS/LICENSE.HTML#LICENSE-AGREEMENT-FOR-MATPLOTLIB-VERSIONS-PRIOR-TO-1-3-0
- BEAUTIFULSOUP4(HTML파서):MIT라이선스, 파이썬 그 자체와 동일