택배상품 수요 예측

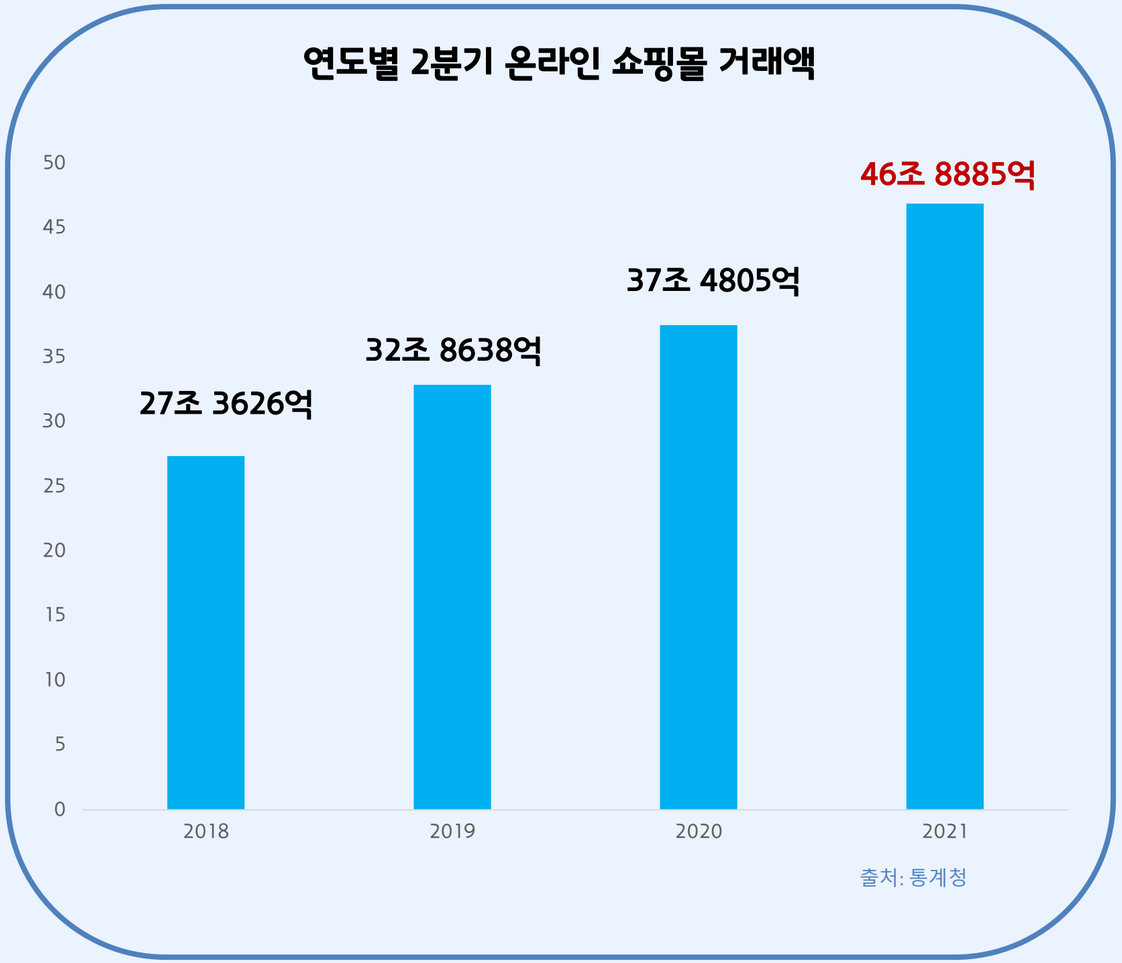
공공 빅데이터 청년 인턴쉽 해커톤

서울 NIA 4그룹1조

윤지의, 김재혁, 안성인, 조승모

" CONTENT "

주제 선정 배경



이커머스 성장 속 풀필먼트 인프라 투자 활발

이커머스 업체들이 배송 역량 강화를 위해 풀필먼트에 대한 투자를 지속적으로 확대하고 있다.

쿠팡은 물류 인프라를 74만 ㎡를 신설했으며, 풀필먼트 센터는 약 90% 가량 증설했다.

신세계 그룹 또한 향후 4년간 1조원 이상을 온라인 풀필먼트에 투자 할 계획이다.

CJ 대한통운도 2023년까지 융합형 풀필먼트 인프라를 현재 8배 이상 수준으로 확장할 예정이다.

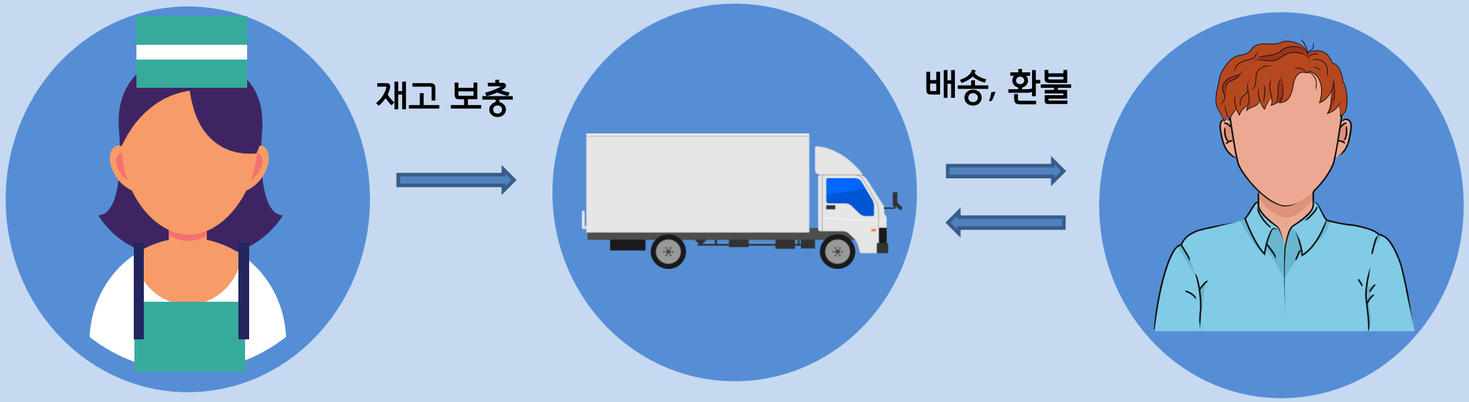
.....

출처:디지털 투데이

주제 선정 배경

상품B를 어디에? 얼만큼?

보관, 선별, 포장



재고 보충

배송, 환불

판매자

풀필먼트 센터

소비자

“재고부족”

손실발생

“과잉재고”

“풀필먼트 서비스” 란?

✓ 풀필먼트는 상품이 물류창고를 거쳐 고객에게 배달 완료까지 전 과정을 판매자 대신 일괄 처리하는 서비스를 말함

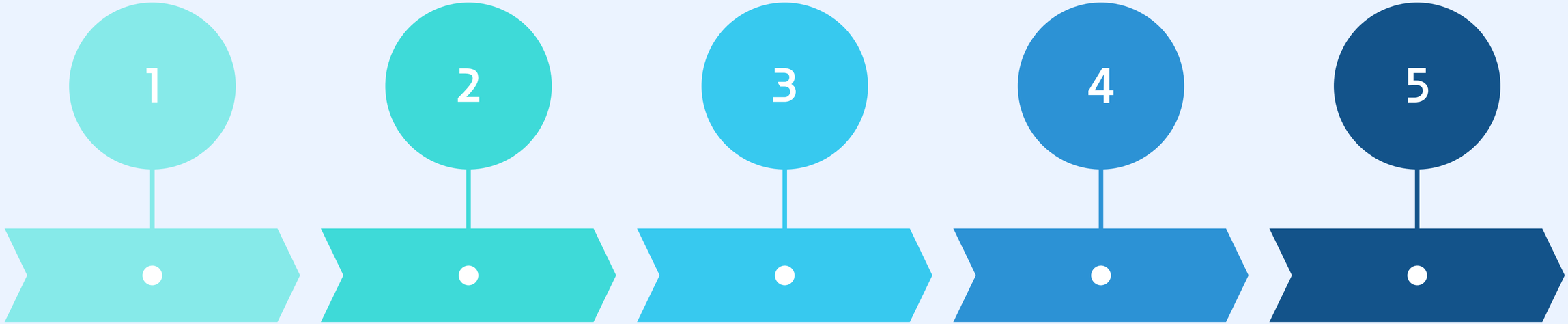
✓ 창고에 재고가 부족하게 되면 판매기회를 잃어 손실발생

✓ 과잉재고 발생 시, 보관비용, 생산비용 등 손실발생

어떤 상품을 어디에 얼만큼 보충 해야 하는가?는 판매자에게 중요한 문제

본 팀은 판매자의 상품별 **주문량을 예측**하고, 분석결과를 **시각자료를 통해 전달**하여 **효율적인 재고 계획 수립에 도움**을 주고자 함

분석 순서도



데이터 EDA 및 전처리

변수생성 및 선택

모델링 및 검증

모델링 최적결과 도출

시각화

사용 데이터

1) 사용 데이터

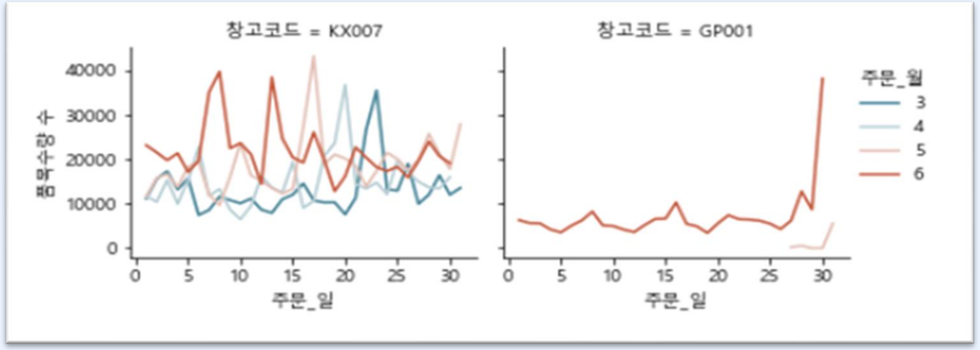
사용 데이터	출처	데이터 기간
풀필먼트 센터 주문 데이터	CJ 대한통운	2021-03 ~ 2021-06
공휴일 데이터	https://www.timeanddate.com/holidays/south-korea/	2021년 기준

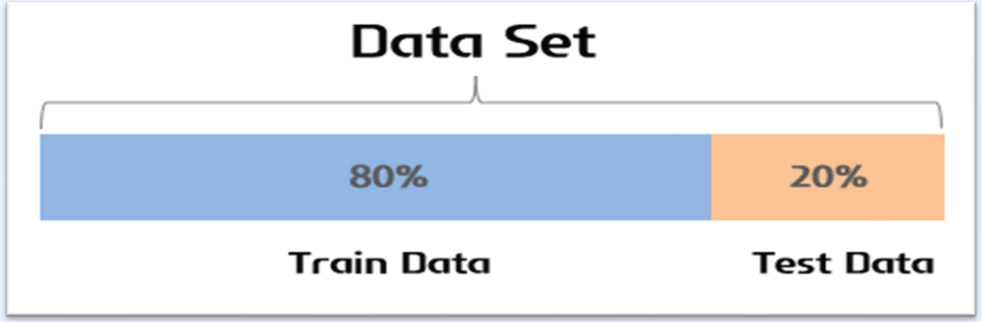
2) 풀필먼트 주문 데이터 주요 변수

주문	품목	배송
고객 주문번호	품목코드	택배사 주문번호
주문유형	품목순번	창고코드
주문날짜	품목명	권역구분
주문수량	품목금액	택배구분
고객사코드	-	수화인, 송화인 주소
주문금액	-	배달터미널코드

데이터 전처리

고객사코드	창고코드	주문유형	주문수량
C1	KX007	7(출고)	3
C1	GP001	8(정상반출)	1
C3	KX007	9(불량반출)	6





품목별 수요 예측이 목적이므로 주문유형(출고, 정상반출, 불량반출)

중 수요와 관련된 **출고 데이터만 추출**

창고코드 GP001은 2021-06월부터 풀필먼트 센터로 지정되어,

그 전 시점의 주문데이터가 없으므로 분석 데이터에서 제외하고

창고코드가 KX007인 데이터만 추출하여 분석을 진행

Train Data: 3,4,5월 주문 데이터 (2021-03-01 ~2021-05-31)

Test Data: 6월 주문 데이터 (2021-06-01~2021-06-30)

변수 생성

고객사,품목코드,권역구분
모두가 같은 데이터 끼리 묶은
class 변수 생성

ANOVA 검정

H0 : 고객사코드, 품목코드, 권역별에 따라 품목수량의 평균 차이가 존재하지 않는다.

H1 : 고객사코드, 품목코드, 권역별에 따라 품목수량의 평균 차이가 존재한다.

p-vaule = 5.284759e-08이므로, 대립가설 채택

즉, 같은 고객사라도 품목코드,권역구분에 따라 주문수량의 차이를 보였다

고객사코드	품목코드	권역구분	주문수량
C1	P1	1	1
C2	P2	4	9
C4	P3	2	8
C4	P3	2	2
C5	P1	4	1

Class	주문수량
V1	1
V2	9
V3	8
V3	2
V4	1

Class	주문수량 평균
V1	5
V2	
V3	
V4	

Class	주문수량 평균
V1	5
V1	5
V2	
V4	

-> train data(3,4,5월)로 class 별 주문수량(Y) 통계치를 계산하여 test set(6월) 독립변수로 매핑하여 예측

변수 생성 - CV, TO%

[Train 데이터]

class	주문수량 평균	주문수량 표준편차	주문수량 합
V1	1.027211	0.200874	151
V2	1.014286	0.119523	71
V3	1.307692	1.1094	17
V4	1.503546	1.234181	212

[Train + Test 데이터]

class	CV	TO%	ADI	판매동향
V1	0.372678	0.000455	13.400000	intermittent
V2	0.119510	0.005235	1.294118	Smooth
V3	0.962091	0.000835	8.428571	Lumpy
V3	0.962091	0.000835	8.428571	Lumpy
V4	0.871420	0.013277	0.451327	Erratic

CLASS별 주문수량 통계

3,4,5월 데이터 (Train Data)로 Class별 주문 수량 통계 구함

df = df.groupby('class')['주문수량'].agg(['주문수량평균' , ',mean'), ('주문수량표준편차' , 'std') , ('주문수량합', 'sum')]).reset_index()

CV, TO%

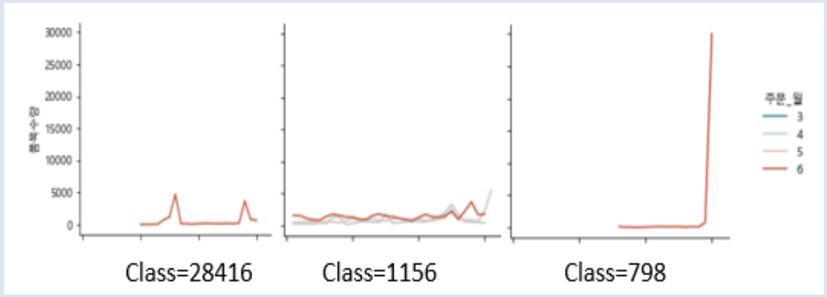
CV: 주문 수량의 변동성을 알기 위한 변수

$$CV = \frac{\text{주문수량 표준편차}}{\text{주문수량 평균}}$$

TO%: 전체 주문 수량 중 각 class의 주문수량 비율

$$\text{TO\%} = \frac{Class\text{의 주문수량 합}}{\text{전체 } Class \text{ 주문수량 합}} \text{ X } 100$$

변수 생성 - ADI



주문 주기가 매우 불규칙한 Class가 많음

-> Class별 주문주기를 학습하기 위한 변수 생성 해야 함

과거주문날짜: 해당 주문날짜 이전 마지막으로 주문된 날짜

주문간격: 주문날짜 - 과거주문날짜

ADI: Class별 평균 주문 간격

[Train 데이터]

class	주문날짜	과거주문날짜	주문간격
V1	2021-04-16	2021-03-23	23
V1	2021-04-20	2021-04-16	4
V1	2021-05-05	2021-04-20	15
V1	2021-05-16	2021-05-05	11
V1	2021-05-30	2021-05-16	14

V1 Class 주문 간격 평균
23+4+15+11+14/5 = 13.400

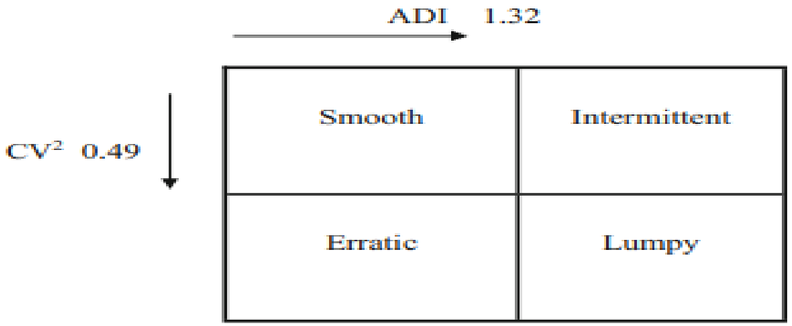
[Train + Test 데이터]

class	CV	TO%	ADI	판매동향
V1	0.372678	0.000455	13.400000	intermittent
V2	0.119510	0.005235	1.294118	Smooth
V3	0.962091	0.000835	8.428571	Lumpy
V3	0.962091	0.000835	8.428571	Lumpy
V4	0.871420	0.013277	0.451327	Erratic

변수 생성 - 판매동향

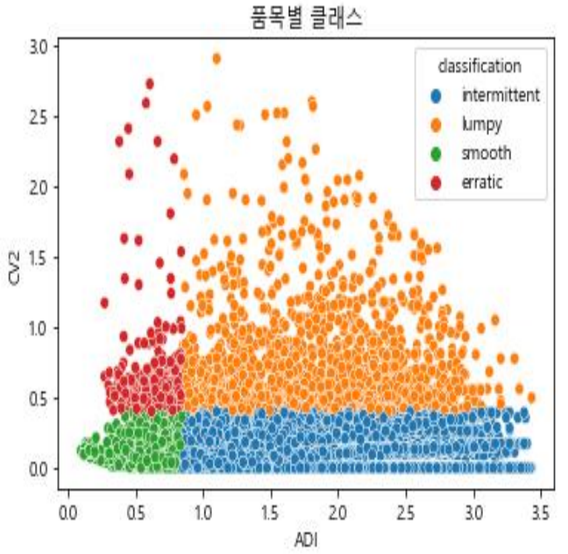
Class별 판매 동향

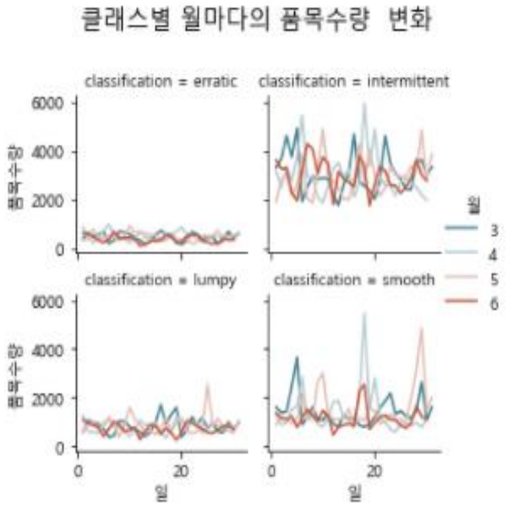
: 앞서 구한 ADI와 CV를 제곱한 값을 사용하여 Class별 판매 동향을 4가지 클래스로 매핑



유형	설명	분류기준
Smooth	수요 발생 간격과 수량이 비교적 규칙적으로 발생하는 품목	ADI<1.32 and cv^2<0.49
Intermittent	수요수량의 변동은 낮지만, 수요 발생 간격이 크게 변화하는 품목	ADI<1.32 and cv^2>=0.49
Erratic	수요수량 변동은 높지만, 수요 발생 간격은 비교적 규칙적인 품목	ADI>=1.32 and cv^2<0.49
Lumpy	특정시기에 한 번에 많은 수량이 발생되는 품목	ADI>=1.32 and cv^2>=0.49

품목별 판매 동향 별 클래스





- Smooth: 169,484개
- Intermittent: 368,826개
- Erratic: 57,231개
- Lumpy : 101,098개

변수 생성

공휴일 , One-Hot Encoding

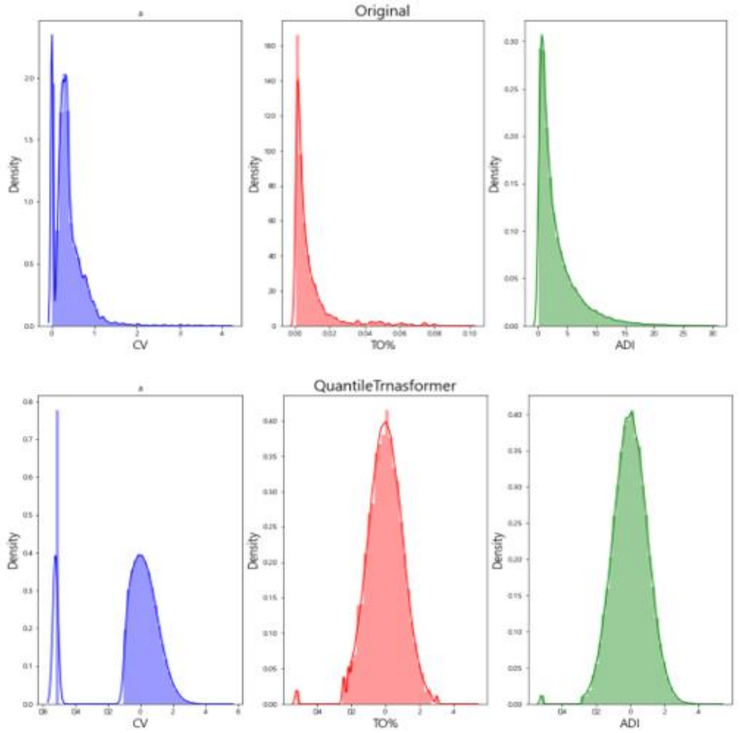
휴일	CV	TO%	ADI	판매동향
0	3.507520	0.012366	0.776569	Erratic
0	0.707107	0.000607	19.25000	Lumpy
1	0.186031	0.016311	0.439614	Smooth
0	0.186031	0.016311	0.439614	Smooth
0	0.186031	0.016311	0.439614	Smooth
1	0.000455	0.000455	10.16667	intermittent

- 휴일: 주문 날짜가 주말 혹은 공휴일인지 여부

- One-Hot Encoding: 범주형 변수들을 인코딩

- 주문요일
- 고객사코드
- 배달권역구분
- 판매 동향
- 주문일
- 품목코드

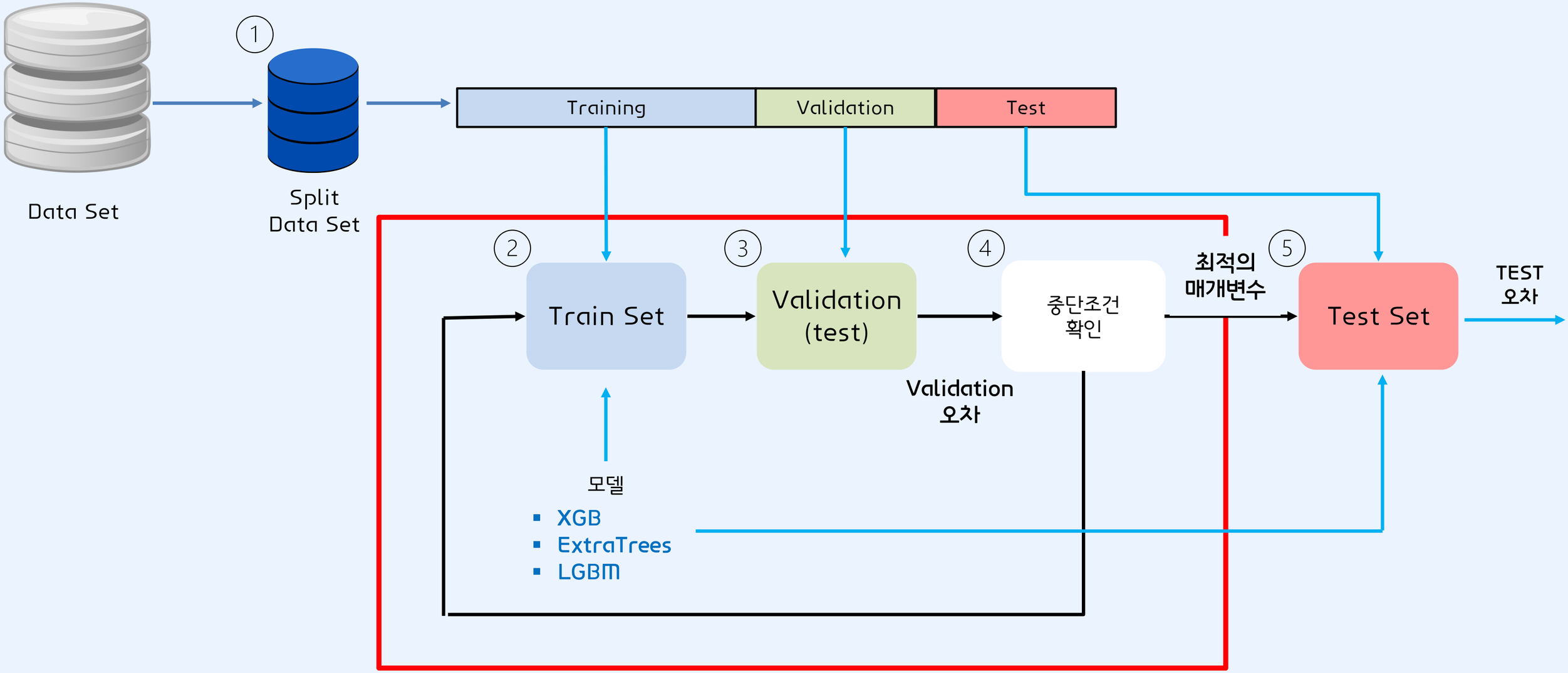
변수 스케일링



- 연속형 변수(CV, TO%, ADI)들의 분포가 불균형

-> QUANTILE TRANSFORM으로 정규분포에 근사하도록 스케일링

모델링 - 순서


Data Set
1
Split
Data Set
Training
Validation
Test
2
3
4
5
Train Set
Validation
(test)
중단조건
확인
최적의
매개변수
Test Set
TEST
오차
Validation
오차
모델
XGB
ExtraTrees
LGBM

모델링 - GridSearchCV

: 모델링에 필요한 최적의 매개변수(하이퍼 파라미터) 찾기

GridSearch(그리드 서치)란?

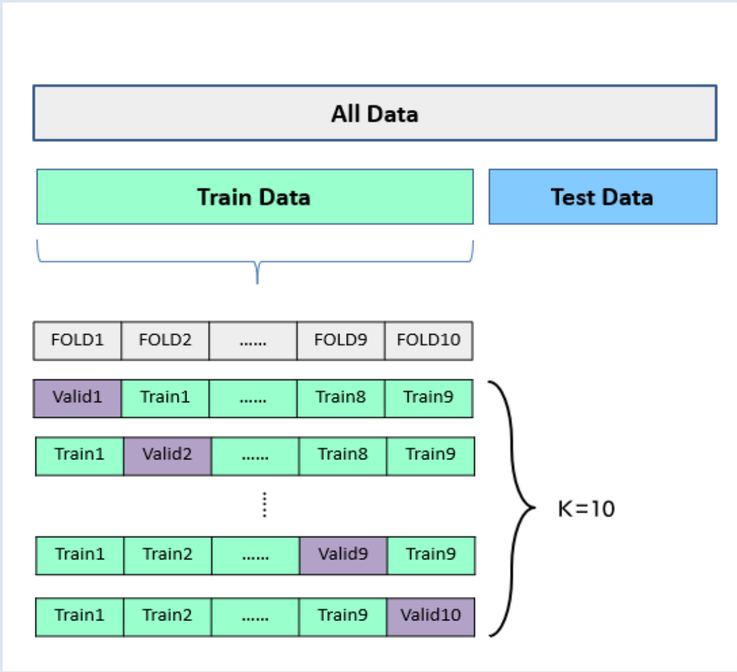
- 관심있는 매개변수들을 대상으로 가능한 모든 조합을 시도하여 최적의 매개변수를 찾는 방법

CV(교차 검증)란?

- 모델에 사용되는 매개변수를 조정하고 과적합을 막기 위해 사용하는 검증 방식

- 데이터를 Train, validation으로 나눌 때 단순히 1번 나누는 것이 아닌 k번 나누고 학습 후 평균값으로 모델 성능 판단

GridSearchCV



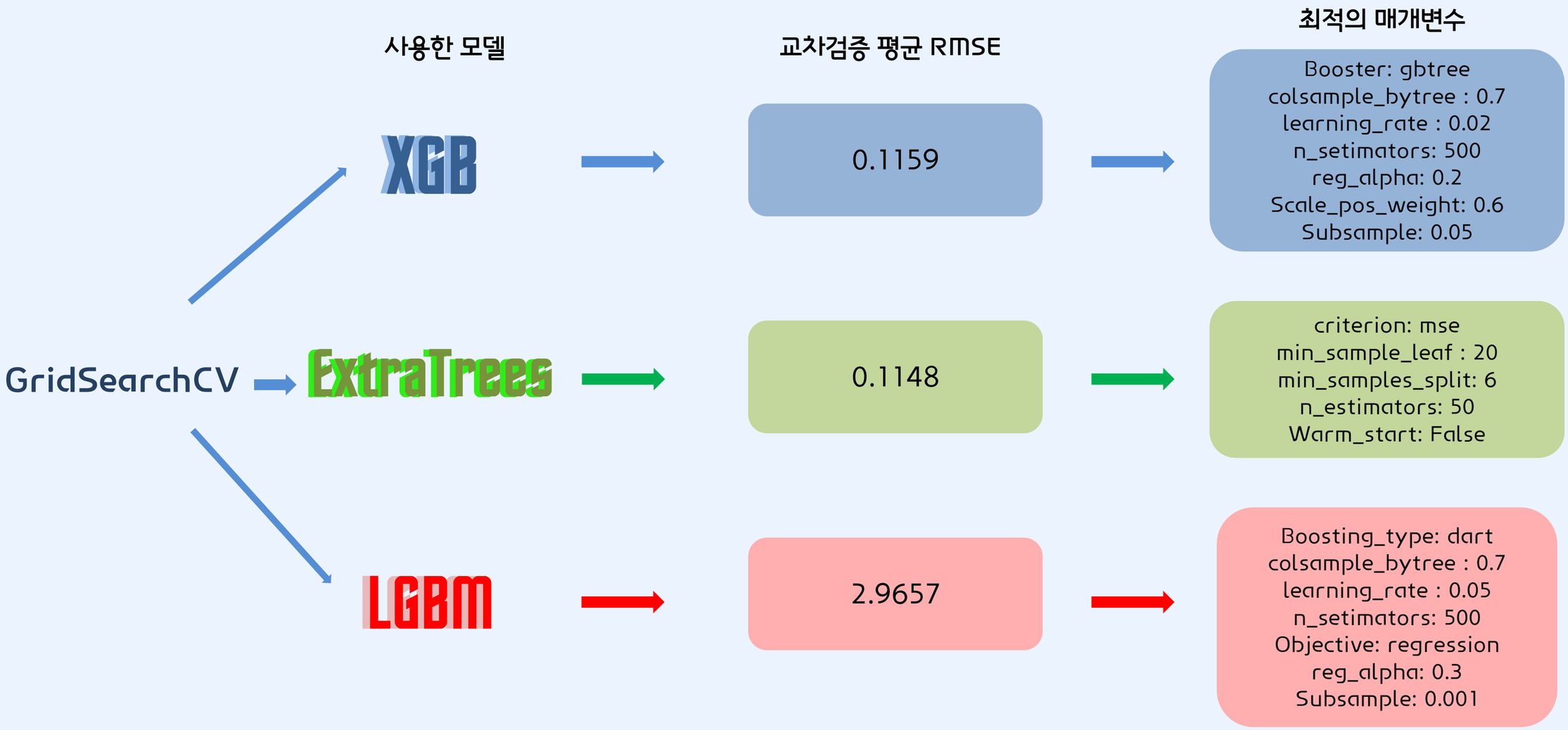
1. Train 데이터를 10등분
2. 1개를 Validation, 9개를 Train으로
3. 매개변수와 Train으로 학습

3. Validation으로 성능 평가

4. validation을 바꿔가며 반복

-> 총 10개의 성능 결과

모델링 - GridSearchCV



모델링 결과 - 모델 선택

	Test 데이터 RMSE	교차검증 평균 RMSE

XGB	`xgb_preds = xgb_estimator.predict(X_test_scaled)`
	`rmse_xgb = np.sqrt(mean_squared_error(y_test,xgb_preds))`
	`print("XGBRegressor test RMSE: %f" % (rmse_xgb ))`
	XGBRegressor test RMSE: 2.334536
		0.1159

ExtraTrees	`ext_preds = ext_estimator.predict(X_test_scaled)`
	`rmse_ext = np.sqrt(mean_squared_error(y_test,ext_preds))`
	`print("ExtraTreesRegressor test RMSE: %f" % (rmse_xgb ))`
	ExtraTreesRegressor test RMSE: 2.334536
		0.1148

LGBM	`lgb_preds = lgb_estimator.predict(lgb_test_x)`
	`rmse_lgb = np.sqrt(mean_squared_error(lgb_test_y,lgb_preds))`
	`print("LGBRegressor test RMSE: %f" % (rmse_lgb ))`
	LGBRegressor test RMSE: 2.262533
		2.9657

Train 데이터 예측 결과는 0.11대로 높게 나왔지만,
Test 데이터 예측 결과는 2.33으로 차이가 많이 남

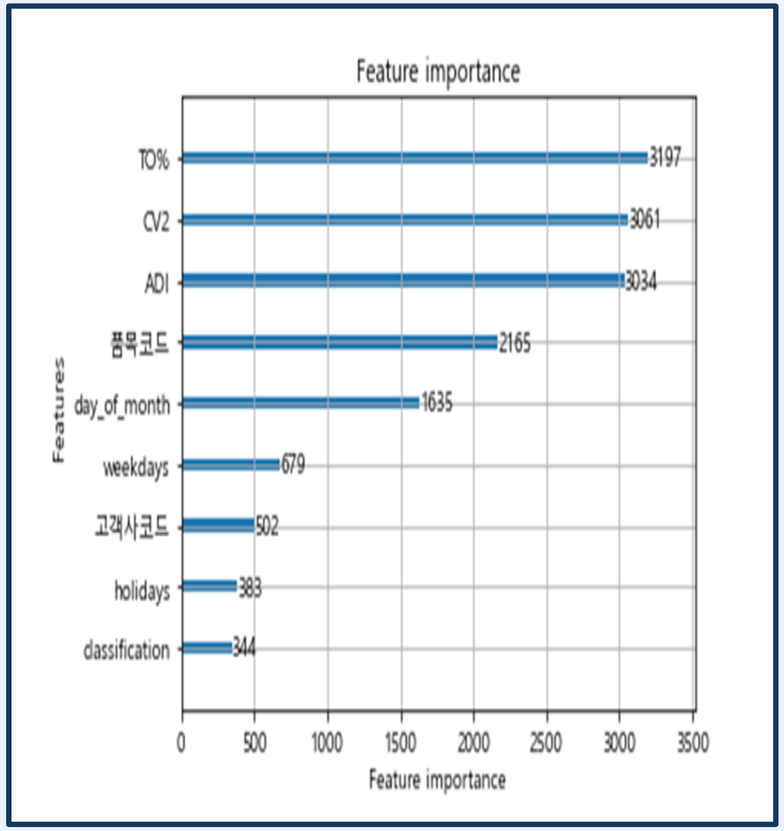
-> **과적합으로 볼 수 있음**

Train 데이터 예측 결과와 Test 데이터 예측 결과가
비슷하면서 성능이 가장 높음

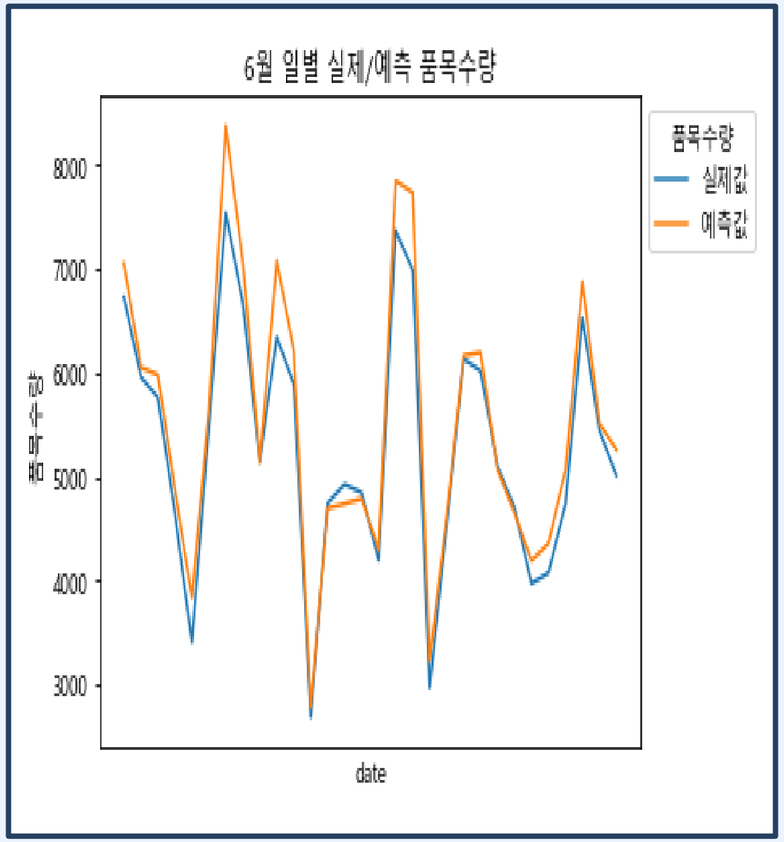
-> 새로운 데이터도 잘 예측하는 좋은 모델이므로
LGBM을 최종 모델로 채택

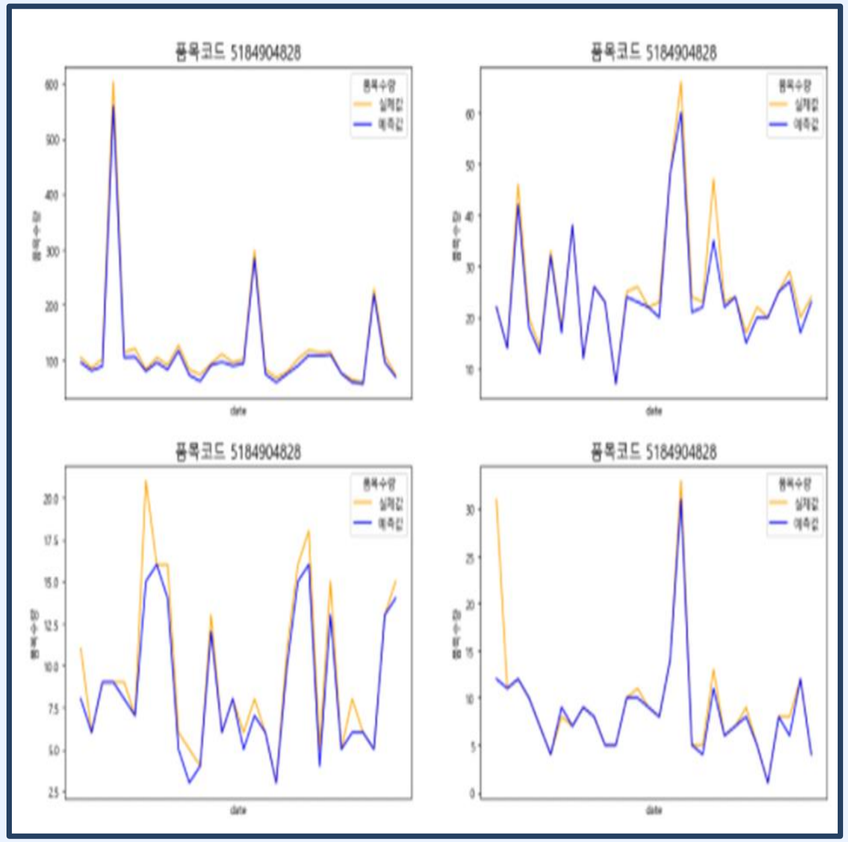
모델링 결과 - LGBM

변수 중요도

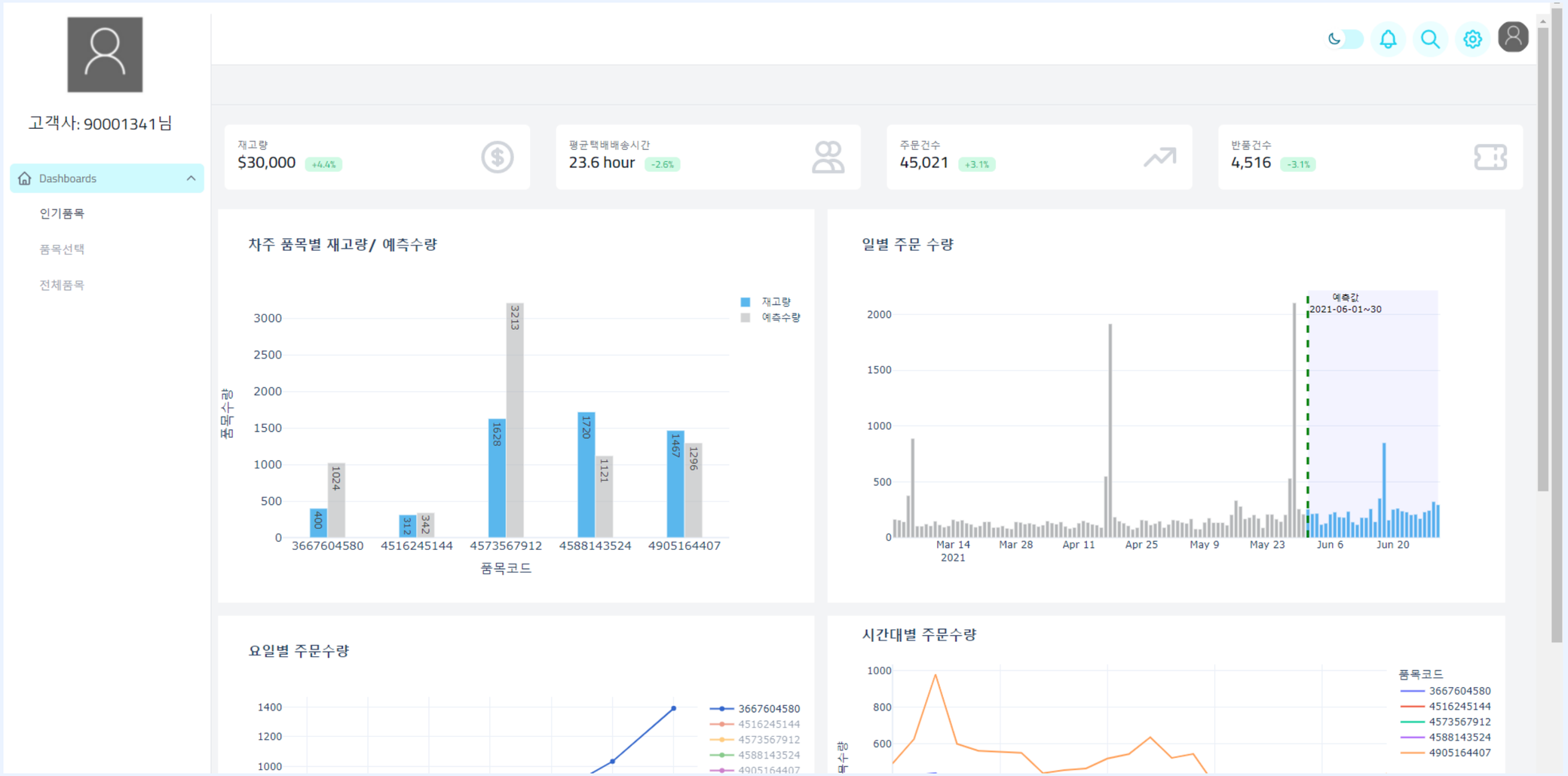


예측 값, 실제 값 비교





시각화


고객사: 90001341님
Dashboards
인기품목
품목선택
전체품목
재고량
$30,000
+4.4%
평균택배배송시간
23.6 hour
-2.6%
주문건수
45,021
+3.1%
반품건수
4,516
-3.1%
차주 품목별 재고량/ 예측수량
재고량
예측수량
품목수량
3000
2500
2000
1500
1000
500
0
400
1024
312
342
1628
3213
1720
1121
1467
1296
3667604580
4516245144
4573567912
4588143524
4905164407
품목코드
일별 주문 수량
예측값
2021-06-01~30
2000
1500
1000
500
0
Mar 14
2021
Mar 28
Apr 11
Apr 25
May 9
May 23
Jun 6
Jun 20
요일별 주문수량
1400
1200
1000
3667604580
4516245144
4573567912
4588143524
4905164407
시간대별 주문수량
1000
800
600
품목코드
3667604580
4516245144
4573567912
4588143524
4905164407

시각화

파레토 ABC 분석

파레토 ABC 분석?

"전체 품목중 상위 20%가 전체 판매 수량의 80%"

즉, 소수의 품목이 전체 판매 수량에 큰 비율을 차지한다는 파레토 기법을

기반으로 효율적인 재고를 위해 품목을 A, B, C등급을 분류하는 분석

ABC 분석 예시

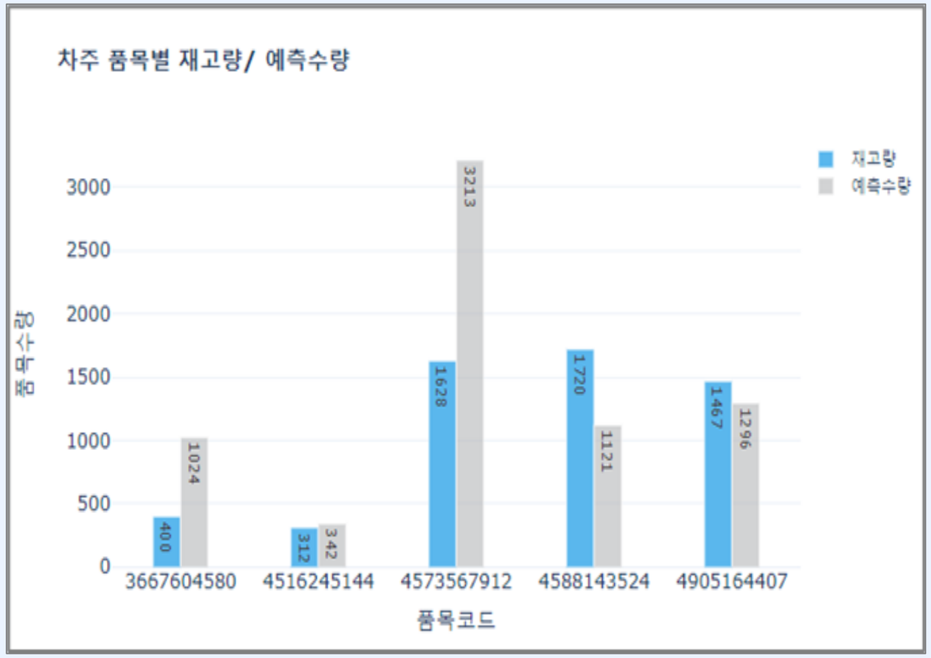
판매자: 90001341

1) A등급 상품: 상위 20% 품목이 전체 판매 수량의 84%를 차지
2) B등급 상품: 다음 30% 품목이 전체 판매 수량의 13%를 차지
3) C등급 상품: 나머지 50% 품목이 전체 판매 수량의 2%를 차지

전체 판매량의 84%를 차지하는 상위 20% 상품 재고 보충을 우선

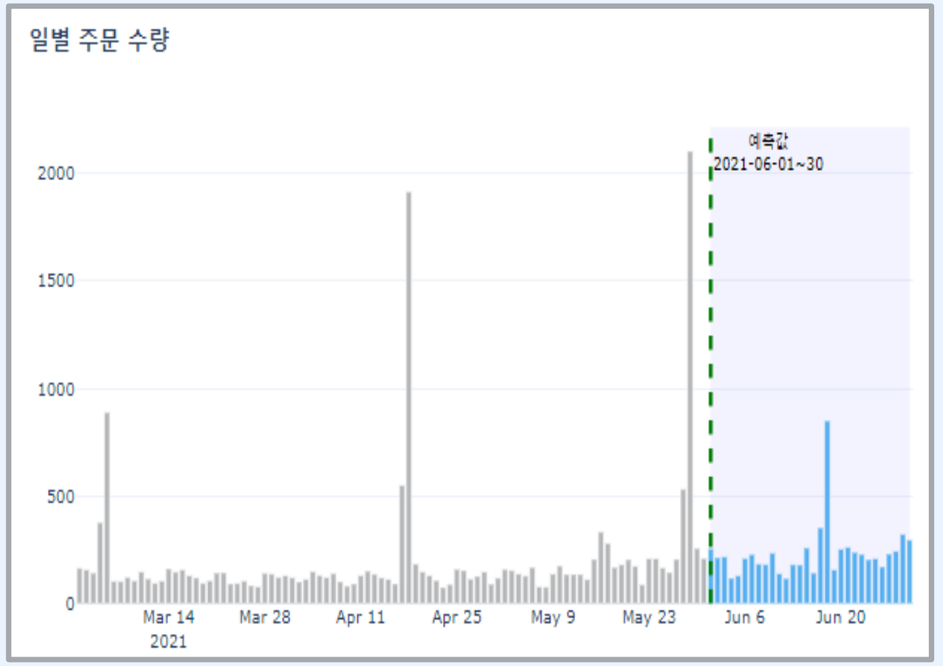
시각화

1) 차주 품목별 재고량 및 예측 수량



- 품목 별 예측수량과 현재 재고량을 비교하여 보충 계획 및 제품 생산 의사결정에 기여

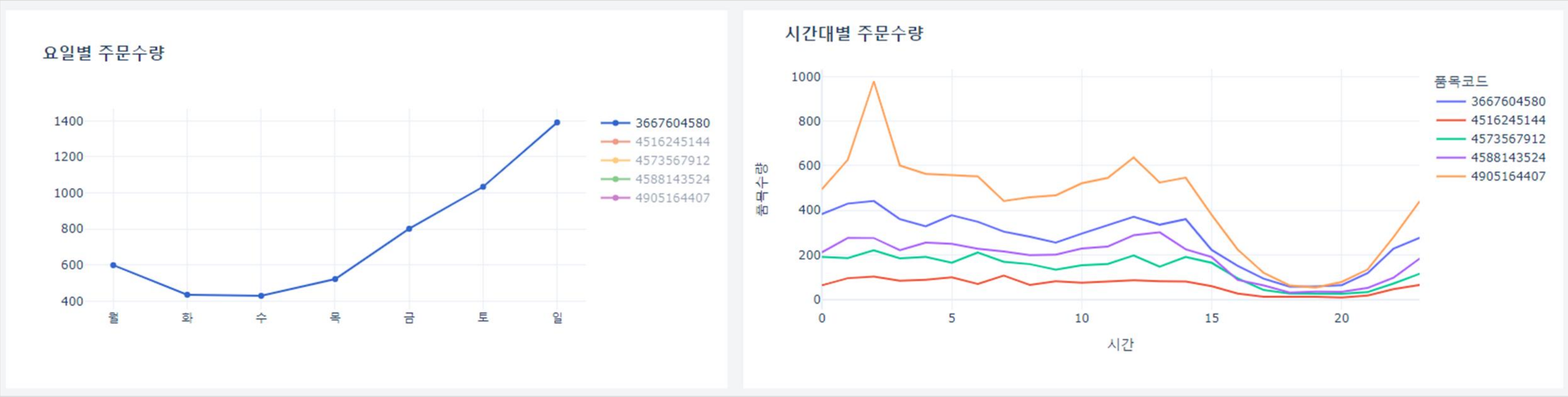
2) 차주 품목별 재고현황 및 예측 수량



- 과거 주문 수량(회색)으로 추세파악 및 예측 주문 수량(파란색)을 기반으로 물동량 선제적 대응 가능

시각화

3) 요일, 품목, 시간대 별 주문수량 및 판매비율



- 요일, 시간대 별 주문의 추세 및 판매 비율을 파악하여 재고 보충 계획 및 판매 트랜드 분석 가능

기대효과

: 수요 예측 결과를 시각자료를 통해 보다 쉽게 전달
-> 판매자는 안정적인 재고관리 및 선제적 대응 가능

✓수요 예측을 통한 안정적인 재고관리

✓파레토 ABC 분석기반으로 상품의 중요도에 따라 효율적인 창고활용 가능

✓풀필먼트 창고 보충횟수를 최적화하여 운송비용 절감

✓예측된 수요를 기반으로 상품 생산, 마케팅 등 비즈니스 의사결정가능

한계

✓ 데이터 비식별화가 많이 진행되어 사용 가능한 속성이 적음

✓ 소비는 상품에 따라 영향 받는 데이터가 많지만, 주문 데이터의 상품에 대한 정보가 현저히 부족하여 사용하지 못함 (ex: 우산 -> 날씨데이터 사용)

✓ 결측치가 다수 존재하는 변수가 많아 다양한 변수를 활용하기에 제한이 됨

✓ 약 4개월의 단기간 데이터이기 때문에 계절변화 및 추세변화 등 시계열 분석에 필요한 조건들을 불만족 시킴

감사합니다