

NH 투자증권

주식 보유기간

예측

장민수 / 최우철 / 허지혜 / 진서영



**CHAPTER
01**

대회 설명 및
데이터 설명

**CHAPTER
03**

모델링 정리

**CHAPTER
02**

데이터 전처리,
시각화

**CHAPTER
04**

결과 정리





CHAPTER 91



Contents

01. DATA

NH 투자증권의 데이터를 이용하여
주식 보유 기간을 예측하는 REGRESSOR 문제

제공 DATA = 정형 DATA

cus_info.csv
(1,000건)

고객 및 주 거래계좌 정보

stk_bnc_hist.csv
(2,573,839건)

국내주식 잔고이력

iem_info.csv
(3,078건)

종목 정보

stk_hld_train.csv
(681,472건)

16년 1월 ~ 20년 12월
사이 고객의 국내주식
거래가 종료 된 건

stk_hld_test.csv
(70,596건)

20년 12월 이전에 매수
하고 21년 이후에 고객
이 전량 매도한 국내주
식 보유기간 예측

Contents

01. DATA

1) cus_info.csv : 고객 및 주거래계좌 정보

act_id : 계좌 ID

sex_dit_d : 성별

cus_age_stn_cd : 연령대

ivs_icn_cd : 투자성향

cus_aet_stn_cd : 주거래상품군

lsg_sgm_cd : Life Style

tco_cus_grd_cd : 서비스 등급

tot_ivs_te_sgm_cd : 총 투가기간

mrz_btp_dit_cd : 주거래업종구분

sex_dit_cd	cus_age_stn_cd	ivs_icn_cd	cus_aet_stn_cd
1	4	99	1
1	6	4	4
2	7	4	3
2	6	4	4
1	5	2	2

mrz_pdt_tp_sgm_cd	lsg_sgm_cd	tco_cus_grd_cd	tot_ivs_te_sgm_cd
2	3	3	6
2	5	2	6
2	5	5	6
8	5	3	6
2	5	5	5

mrz_btp_dit_cd
16
1
9
16
16

범주형 변수

shape : (1000,10)

1) cus_info.csv

sex_dit_cd

```
cus['sex_dit_cd'].unique()
```

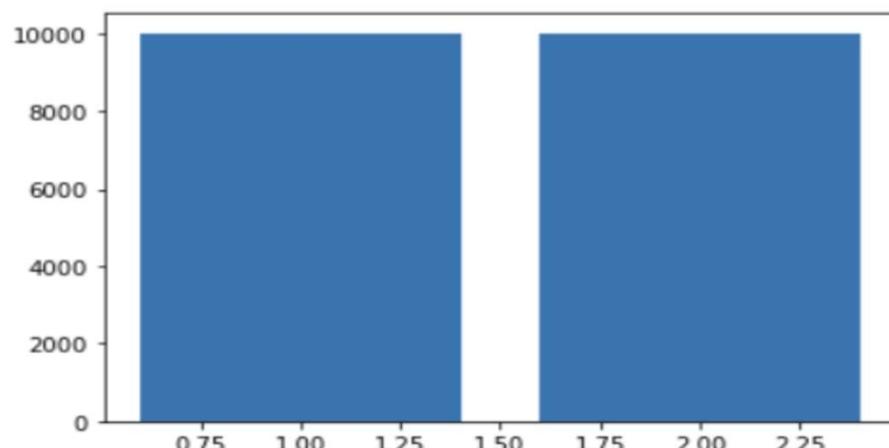
```
array([1, 2], dtype=int64)
```

```
cus['sex_dit_cd'].value_counts()
```

```
1    5985  
2    4015  
Name: sex_dit_cd, dtype: int64
```

```
import matplotlib.pyplot as plt  
plt.bar(cus['sex_dit_cd'],cus.index)
```

```
<BarContainer object of 10000 artists>
```



cus_age_stn_cd

```
cus['cus_age_stn_cd'].unique()
```

```
array([4, 6, 7, 5, 8, 9, 2, 3, 1], dtype=int64)
```

연령대 구분

1 : 20세 ~ 25세 미만

2 : 25세 ~ 30세 미만

3 : 30세 ~ 35세 미만

4 : 35세 ~ 40세 미만

5 : 40세 ~ 45세 미만

6 : 45세 ~ 50세 미만

7 : 50세 ~ 55세 미만

8 : 55세 ~ 60세 미만

9 : 60세 ~ 65세 미만

ivs_icn_cd

계좌 가입시 설문조사로 결정되는 투자성향

- 1: 안정형
- 2: 안정추구형
- 3: 위험중립형
- 4: 적극투자형
- 5: 공격투자형
- 9: 전문투자가형
- 00: 정보제공미동의
- 99: 미정의

1) cus_info.csv



MRZ_PDT_TP_SGM_CD



고객의 월말 기준 잔고가 가장 많은 상품의 유형으로 분류, 잔고
가 동일할 경우 상품소분류코드가 작은 순으로 우선 인식

- 01: Only CMA
- 02: 국내주식
- 03: 해외주식
- 04: 선물옵션
- 05: 금속
- 06: 국내채권
- 07: 해외채권
- 08: 펀드
- 09: ELS/DLS
- 10: 신탁_퇴직연금
- 11: RP
- 12: 발행어음
- 14: WRAP
- 15: 신용대출
- 99: 미정의

LSG_SGM_CD

고객 *Life stage*, 업종 등 고려하여 유사 특성을 보유한
고객 클래스 도출

- 02: 사회초년생 (20-29세)
- 03: 가족형성기_남자 (30-39세 & 남자)
- 04: 가족형성기_여자 (30-39세 & 여자)
- 05: 가족성숙기_직장인 (40-59세 & 직장인 & 남자)
- 06: 가족성숙기_주부 (40-59세 & 주부 & 여자)
- 07: 가족성숙기_남자 (40-59세 & 기타 & 남자)
- 08: 가족성숙기_여자 (40-59세 & 기타 & 여자)
- 09: 은퇴기 (60-69세)

TOT_IVS_TE_SGM_CD



자산 및 수익 기준 고객의 등급을 부여

- 01: 탑클래스 (자산1)10억이상 or 수익기여도2) 5백만원 이상)
- 02: 골드 (자산3억이상 or 수익기여도 3백만원 이상)
- 03: 로얄 (자산1억이상 or 수익기여도 1백만원 이상)
- 04: 그린 (자산3천이상 or 수익기여도 5십만원 이상)
- 05: 블루 (자산1천이상 or 수익기여도 1십만원 이상)
- 09: 등급 미정의
- 99: 미정의 (결측치)

MRZ_BTP_DIT_CD

계좌를 개설한 이래 고객이 100만원 이상 보유한 개월 수

- 01: 6개월 미만
- 02: 6개월-1년 미만
- 03: 1년-3년 미만
- 04: 3년-5년 미만
- 05: 5년-10년 미만
- 06: 10년 이상

Contents

01. DATA

2) STK_BNC_HIST.csv (국내 주식 잔고 이력)

BSE_DT : 기준일자

IEM_CD : 종목 코드

BNC_QTY : 잔고 수량

TOT_AET_AMT : 잔고 금액

STK_PAR_PR : 액면가

bse_dt	iem_cd	bnc_qty	tot_aet_amt	stk_par_pr	stk_p
20200820	A008770	40.0	2828000.0	5000.0	70700.0
20200623	A008770	20.0	1390000.0	5000.0	69500.0
20160104	A005940	311.0	2982490.0	5000.0	9590.0
20200814	A005930	40.0	2320000.0	100.0	58000.0
20200623	A005930	20.0	1028000.0	100.0	51400.0
...
20200806	A035720	1.0	364000.0	500.0	364000.0
20200813	A035720	0.0	0.0	500.0	0.0
20200819	A035720	1.0	376500.0	500.0	376500.0
20200825	A035720	0.0	0.0	500.0	0.0
20200901	A035720	1.0	401500.0	500.0	401500.0

shape : (2573839,7)

연속형 변수

Contents

01. DATA

3) iem_info.csv(3,078건): 종목 정보

iem_cd: 종목코드

iem_krl_nm: 종목한글명

btp_cfc_cd: 종목업종

mkt_pr_tal_scl_tp_cd: 시가총액 규모유형

stk_dit_cd: 시장구분

iem_krl_nm	btp_cfc_cd	mkt_pr_tal_scl_tp_cd	stk_dit_cd
동화약품	8	2	99
하이트진로	14	2	1
성창기업지주	5	3	99
유유제약2우B	8	99	99
노루홀딩스우	2	99	99
...
에코프로에이치엔	14	99	99
KODEX K-미래차액티브	14	99	99
KBSTAR Fn컨택트대표	14	99	99
KBSTAR 비메모리반도체액티브	14	99	99
KOSEF 립소글로벌퓨처모빌리티 MSCI	14	99	99

shape : (3078,5)

3) iem_info.csv



btp_cfc_cd : 종목업종

- '01: 건설
- 02: 금융
- 03: 기계
- 04: 통신
- 05: 서비스
- 06: 운송
- 07: 유통
- 08: 의료
- 09: 전기
- 10: 제조
- 11: 철강
- 12: 화학
- 13: IT
- 14: 기타



mkt_pr_tal_scl_tp_cd:

시가총액 규모유형

- 01: 대형주
- 02: 중형주
- 03: 소형주
- 99: 기타



skt_dit_cd : 시장 구분

- 01: 코스피200
- 02: 코스닥150
- 99: 기타

Contents

01. DATA

4) stk_hld_train.csv(681,472건): 16년 1월 ~ 20년 12

월 사이 고객의 국내주식 거래가 종료 된 건

act_id: 계좌 ID

iem_cd: 종목코드

byn_dt: 매수일자

hold_d: 보유기간(일)

5) stk_hld_test.csv(681,472건): 20년 12월 이전에 매수하고

21년 이후에 고객이 전량 매도한 국내주식 보유기간 예측

act_id: 계좌 ID

iem_cd: 종목코드

byn_dt: 매수일자

hist_d : 과거 보유일

hold_d: 보유기간(일)

act_id	iem_cd	byn_dt	hold_d	hist_d
le554601a981b...	A006360	20180726	11	6.0
le554601a981b...	A005930	20180131	80	48.0
le554601a981b...	A005070	20180517	5	3.0
le554601a981b...	A003520	20201112	22	13.0
le554601a981b...	A002310	20180905	324	194.0

shape : (681472,5)

act_id	iem_cd	byn_dt	hist_d	submit_id	hold_d
a981b...	A032640	20200522	153	IDX00001	0
a981b...	A160600	20190823	335	IDX00002	0
a981b...	A234340	20200611	139	IDX00003	0
a981b...	A131760	20200120	236	IDX00004	0
a981b...	A293490	20201217	9	IDX00005	0

예측값

shape : (70596,6)



독립 변수의 다중공선성

```
train_data_ind = train_data[['act_id', 'iem_cd', 'byn_dt', 'hist_d', 'sex_dit_cd',
    'cus_age_stn_cd', 'ivs_icn_cd', 'cus_aet_stn_cd', 'mrz_pdt_tp_sgm_cd',
    'lsg_sgm_cd', 'tco_cus_grd_cd', 'tot_ivs_te_sgm_cd', 'mrz_btp_dit_cd',
    'iem_krl_nm', 'btp_cfc_cd', 'mkt_pr_tal_scl_tp_cd', 'stk_dit_cd']]
```

다중공선성 판단 방법?! 분산팽창요인(vif)로 판단

VIF < 5 : 안전

5 < VIF < 10 : 주의

VIF > 10 : 위험

	VIF Factor	features
0	2.124624	byn_dt
1	1.223324	hist_d
2	1.776789	sex_dit_cd
3	66.137102	cus_age_stn_cd
4	1.082220	ivs_icn_cd
5	41.979781	cus_aet_stn_cd
6	1.045923	mrz_pdt_tp_sgm_cd
7	67.214753	lsg_sgm_cd
8	42.355930	tco_cus_grd_cd
9	2.977520	tot_ivs_te_sgm_cd
10	1.035454	mrz_btp_dit_cd
11	1.240561	btp_cfc_cd
12	19.150351	mkt_pr_tal_scl_tp_cd
13	19.374849	stk_dit_cd



	VIF Factor	features
0	2.094662	byn_dt
1	1.207306	hist_d
2	1.117505	sex_dit_cd
3	1.032483	ivs_icn_cd
4	38.910402	cus_aet_stn_cd
5	1.036862	mrz_pdt_tp_sgm_cd
6	40.171591	tco_cus_grd_cd
7	2.521807	tot_ivs_te_sgm_cd
8	1.031703	mrz_btp_dit_cd
9	1.239270	btp_cfc_cd
10	19.101323	mkt_pr_tal_scl_tp_cd
11	19.355828	stk_dit_cd



	VIF Factor	features
0	2.088766	byn_dt
1	1.162900	hist_d
2	1.044705	sex_dit_cd
3	1.029682	ivs_icn_cd
4	1.337092	cus_aet_stn_cd
5	1.019169	mrz_pdt_tp_sgm_cd
6	2.307791	tot_ivs_te_sgm_cd
7	1.030150	mrz_btp_dit_cd
8	1.239262	btp_cfc_cd
9	19.084511	mkt_pr_tal_scl_tp_cd
10	19.336187	stk_dit_cd



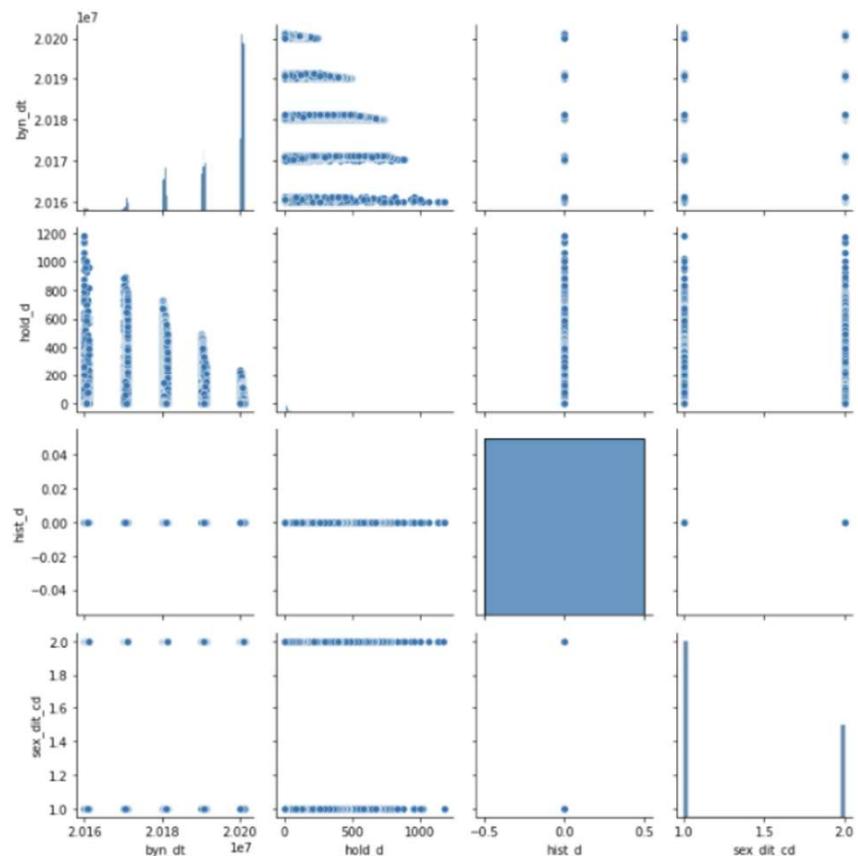
	VIF Factor	features
0	2.076399	byn_dt
1	1.144611	hist_d
2	1.043605	sex_dit_cd
3	1.029162	ivs_icn_cd
4	1.333666	cus_aet_stn_cd
5	1.015625	mrz_pdt_tp_sgm_cd
6	2.305657	tot_ivs_te_sgm_cd
7	1.027351	mrz_btp_dit_cd
8	1.177739	btp_cfc_cd
9	1.237258	mkt_pr_tal_scl_tp_cd

cus_age_stn_cd, cus_aet_stn_cd, tco_cus_grd_cd, lsg_sgm_cd, mkt_pr_tal_scl_tp_cd, stk_dit_cd 제거

상관 분석

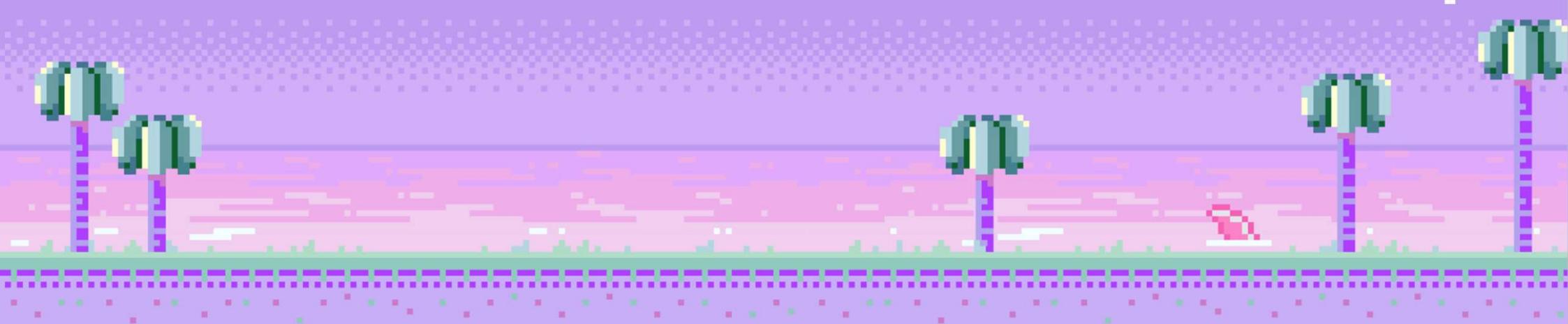
```
: train_data.corr(method='spearman')
```

	byn_dt	hold_d	hist_d	sex_dit_cd	cus_age_stn_cd	ivs_icn_cd	cus_aet_stn_cd	mrz_pdt_tp_sg
byn_dt	1.000000	-0.116926	-0.117317	0.037952	-0.089407	-0.035309	0.007733	0.0
hold_d	-0.116926	1.000000	0.996857	0.045810	-0.020726	0.037023	0.095405	0.0
hist_d	-0.117317	0.996857	1.000000	0.045872	-0.020808	0.037021	0.095247	0.0
sex_dit_cd	0.037952	0.045810	0.045872	1.000000	0.040283	0.005511	-0.025869	-0.0
cus_age_stn_cd	-0.089407	-0.020726	-0.020808	0.040283	1.000000	0.112196	0.236191	-0.0
ivs_icn_cd	-0.035309	0.037023	0.037021	0.005511	0.112196	1.000000	0.041533	-0.0
cus_aet_stn_cd	0.007733	0.095405	0.095247	-0.025869	0.236191	0.041533	1.000000	0.0
mrz_pdt_tp_sgm_cd	0.012778	0.031722	0.031600	-0.007100	-0.042702	-0.029718	0.036781	1.0
lsg_sgm_cd	-0.079624	-0.013337	-0.013249	0.157892	0.870308	0.105534	0.240262	-0.0
tco_cus_grd_cd	0.000105	-0.049126	-0.049130	0.077932	-0.225983	-0.030502	-0.845494	-0.0
tot_ivs_te_sgm_cd	-0.390369	0.040282	0.040306	-0.080534	0.251123	0.054722	0.212786	-0.0
mrz_btp_dit_cd	-0.039158	-0.001339	-0.001165	0.007446	0.026743	-0.017749	0.001363	0.0
btp_cfc_cd	-0.020771	-0.023331	-0.023242	-0.006086	0.009228	-0.006605	0.005154	-0.0
mkt_pr_tal_scl_tp_cd	-0.074276	-0.062687	-0.062231	-0.035280	-0.038464	-0.022178	-0.049224	-0.0
stk_dit_cd	-0.044741	-0.086372	-0.085933	-0.026677	-0.030021	-0.017343	-0.051567	-0.0



상관 분석 결과, 선형 회귀 분석을 쓰기에는 선형성이 매우 낮기에 비선형 회귀 분석을 쓰기로 하였다.

CHAPTER 02



Contents

02

1. 주최 측 힌트에서 보유기간 - 20년 12월까지의 보유기간이 146 이하가 되어야 해서 hist_d 값을 0.877을 곱해 생성.

The screenshot shows a Jupyter Notebook interface with a pink-themed header. The title of the notebook is "데이터 전처리 1". The code cell In [5] contains the following Python code:

```
# Hint : Hold_d(보유기간) - hist_d( '20년 12월 31일까지의 최근 보유기간) ≤ 146
train["hist_d"] = train["hold_d"]*0.877
train.hist_d = np.trunc(train["hist_d"])
```

The code cell In [6] contains:

```
max(train['hold_d']-train['hist_d'])
```

The output cell Out [6] shows:

```
146.0
```

Contents

02

2. train data에 고객정보와 주식정보를 병합

데이터 전처리 2

```
In [7]: train.head(3)
Out[7]:
   act_id    iem_cd    byn_dt  hold_d  hist_d
0  0ad104dbed99be0cd858aa772765ddedade554601a981b...  A006360  20180726    11    9.0
1  0ad104dbed99be0cd858aa772765ddedade554601a981b...  A005930  20180131     80   70.0
2  0ad104dbed99be0cd858aa772765ddedade554601a981b...  A005070  20180517      5    4.0

In [8]: # train과 test에 고객정보(cus_info)와 주식정보(iem_info)를 추가하겠습니다.
train_data = pd.merge(train, cus, how = "left", on = ["act_id"])
train_data = pd.merge(train_data, iem, how = "left", on = ["iem_cd"])

test_data = pd.merge(test, cus, how = "left", on = ["act_id"])
test_data = pd.merge(test_data, iem, how = "left", on = ["iem_cd"])

In [9]: train_data.head(3)
Out[9]:
   act_id    iem_cd    byn_dt  hold_d  hist_d  sex_dit_cd  cus_age_stn_cd  lvs_icn_cd  cus_aet_stn_cd  mrz_pdt
0  0ad104dbed99be0cd858aa772765ddedade554601a981b...  A006360  20180726    11    9.0       1         9        3          2
1  0ad104dbed99be0cd858aa772765ddedade554601a981b...  A005930  20180131     80   70.0       1         9        3          2
2  0ad104dbed99be0cd858aa772765ddedade554601a981b...  A005070  20180517      5    4.0       1         9        3          2
```

Contents

02

3. hist_d를 통해 hold_d를 예측해야 하므로, train data에서 있던 hold_d 열을 제거. 또한 라벨 인코더를 통해 복잡한 형태의 데이터를 숫자형으로 바꾸어 주었다.

데이터 전처리 3

```
In [10]: # train_data에서 Y값을 추출한 후 hold_d column을 지워주겠습니다.  
train_label = train_data["hold_d"]  
train_data.drop(["hold_d"], axis = 1, inplace = True)  
  
In [11]: # 추가적으로 약간의 전처리를 통해 train data와 test data를 구성하겠습니다.  
hist["stk_p"] = hist["tot_aet_qty"] / hist["bnc_qty"]  
hist.fillna(0)  
  
train_data = pd.merge(train_data, hist, how = "left", on = ["act_id", "iem_cd"])  
train_data = train_data[(train_data["byn_dt"] == train_data["bse_dt"])]  
train_data.reset_index(drop = True, inplace = True)  
  
test_data = pd.merge(test_data, hist, how = "left", on = ["act_id", "iem_cd"])  
test_data = test_data[(test_data["byn_dt"] == test_data["bse_dt"])]  
test_data.reset_index(drop = True, inplace = True)  
  
train_data = train_data.drop(["act_id", "iem_cd", "byn_dt", "bse_dt"], axis = 1)  
test_data = test_data.drop(["act_id", "iem_cd", "byn_dt", "submit_id", "hold_d", "bse_dt"], axis = 1)  
  
L_encoder = LabelEncoder()  
L_encoder.fit(iem["iem_krl_nm"])  
train_data["iem_krl_nm"] = L_encoder.transform(train_data["iem_krl_nm"])  
test_data["iem_krl_nm"] = L_encoder.transform(test_data["iem_krl_nm"])
```

Contents

02

전처리 후의 train / test data 모습

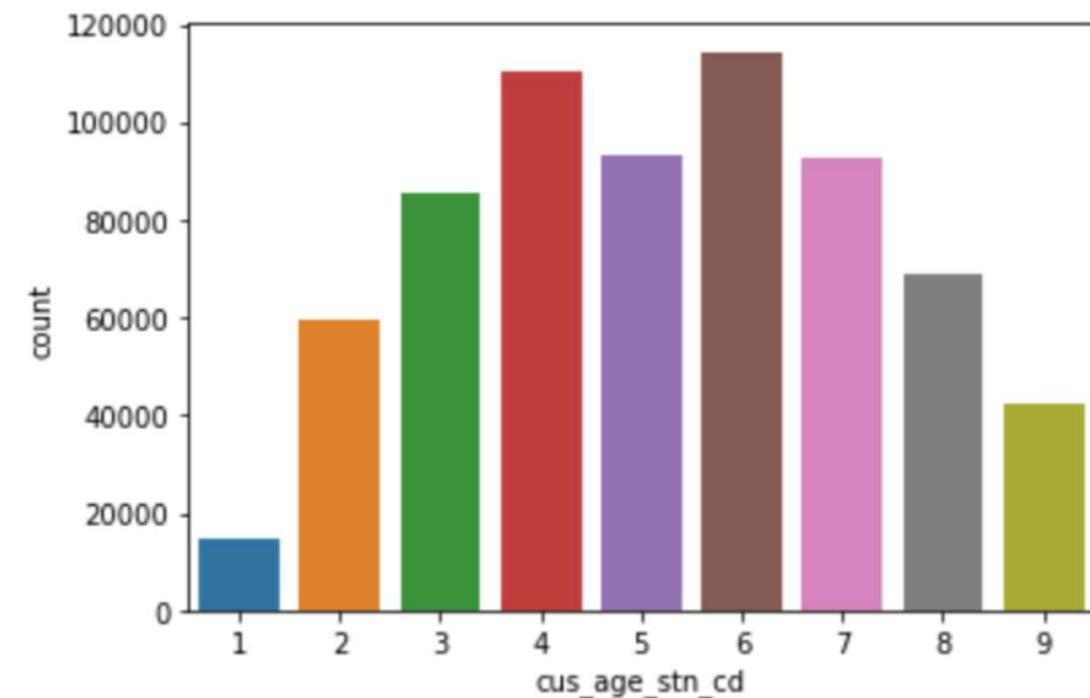
데이터 전처리 결과

```
In [12]: train_data.head(3)
Out[12]:
   hist_d  sex_dit_cd  cus_age_stn_cd  ivs_icn_cd  cus_aet_stn_cd  mrz_pdt_tp_sgm_cd  lsg_sgm_cd  tco_cus_grd_cd  tot_ivs_te_sgm_cd  mrz_btp_dit_cd  item
0      9.0           1              9            3             2                  2                 9               5                  5                 8
1     70.0           1              9            3             2                  2                 9               5                  5                 8
2      4.0           1              9            3             2                  2                 9               5                  5                 8
```

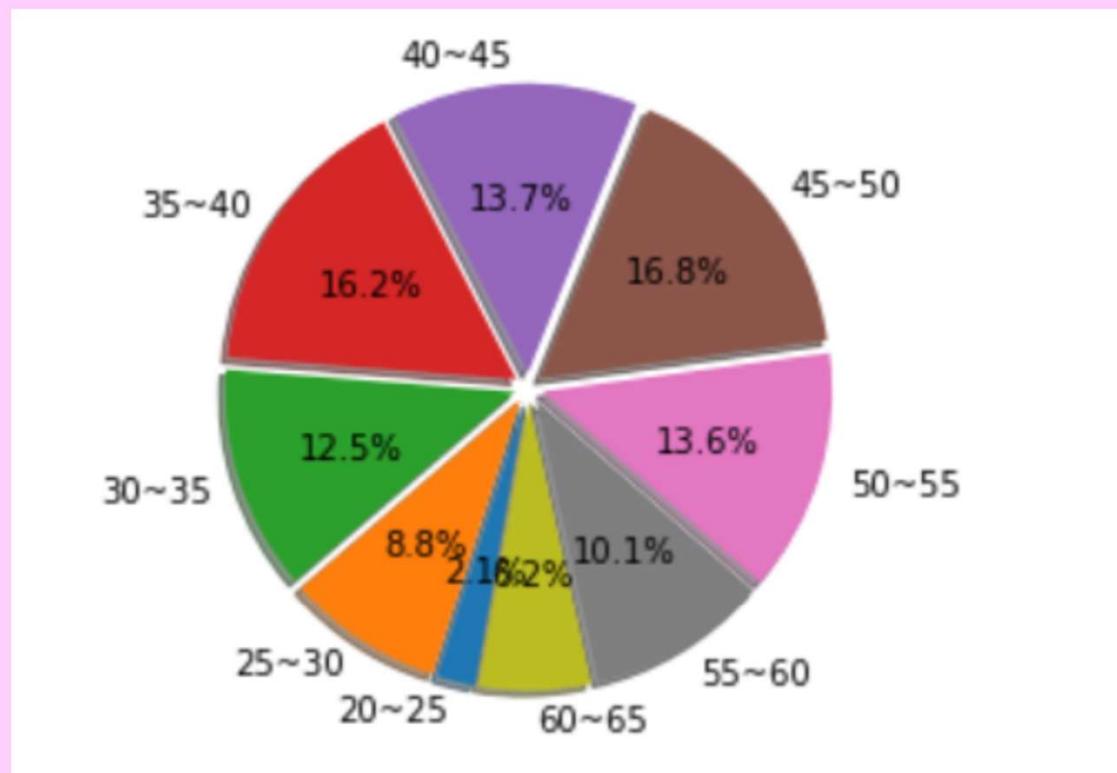
```
In [13]: test_data.head(3)
Out[13]:
   hist_d  sex_dit_cd  cus_age_stn_cd  ivs_icn_cd  cus_aet_stn_cd  mrz_pdt_tp_sgm_cd  lsg_sgm_cd  tco_cus_grd_cd  tot_ivs_te_sgm_cd  mrz_btp_dit_cd  item
0     153           1              9            3             2                  2                 9               5                  5                 8
1     335           1              9            3             2                  2                 9               5                  5                 8
2     139           1              9            3             2                  2                 9               5                  5                 8
```

```
In [14]: train_data.reset_index(drop = True, inplace=True)
train_label.reset_index(drop = True, inplace=True)
```

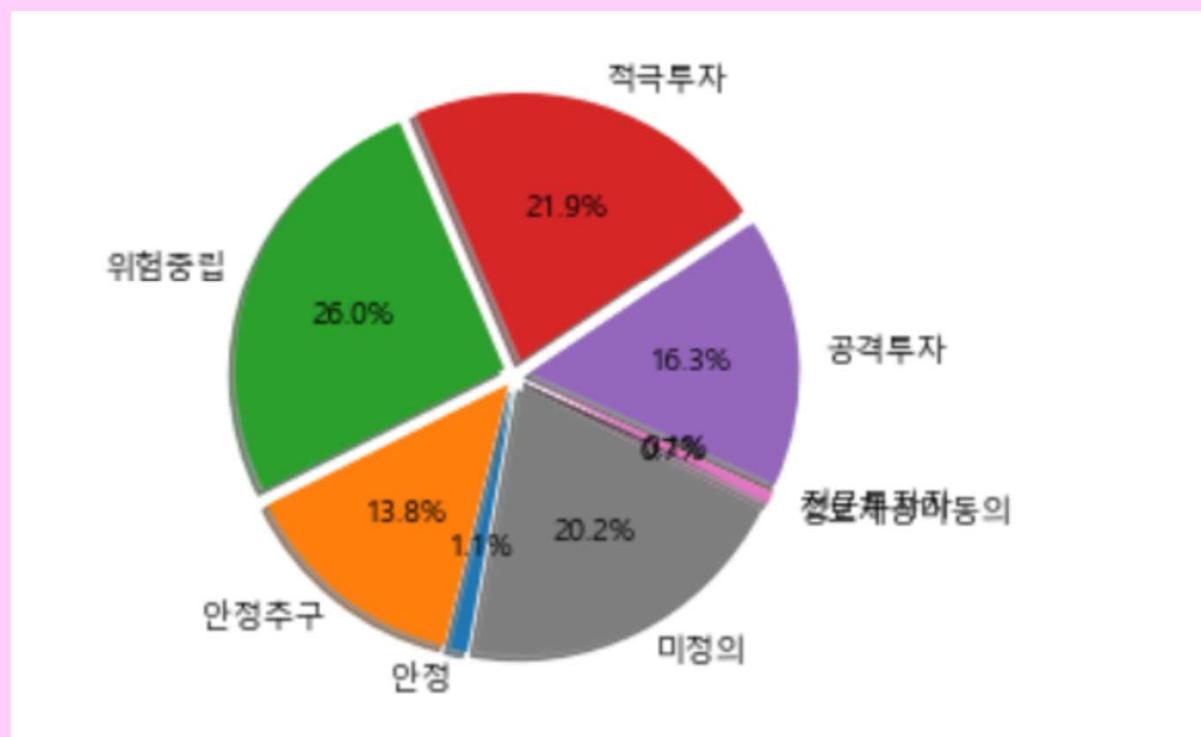
시각화. 연령대별 투자자 수



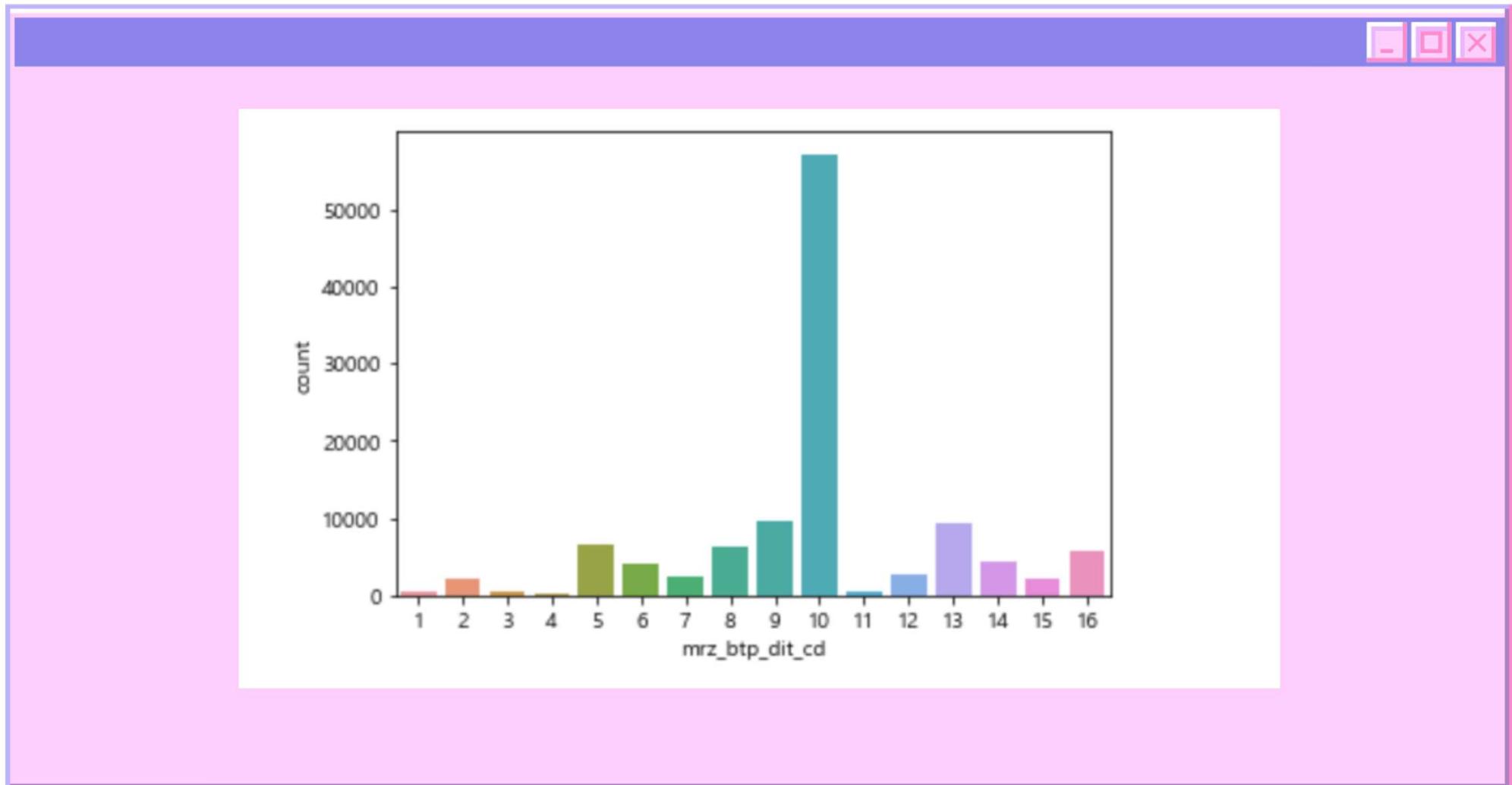
시각화. 연령대별 투자자 비율



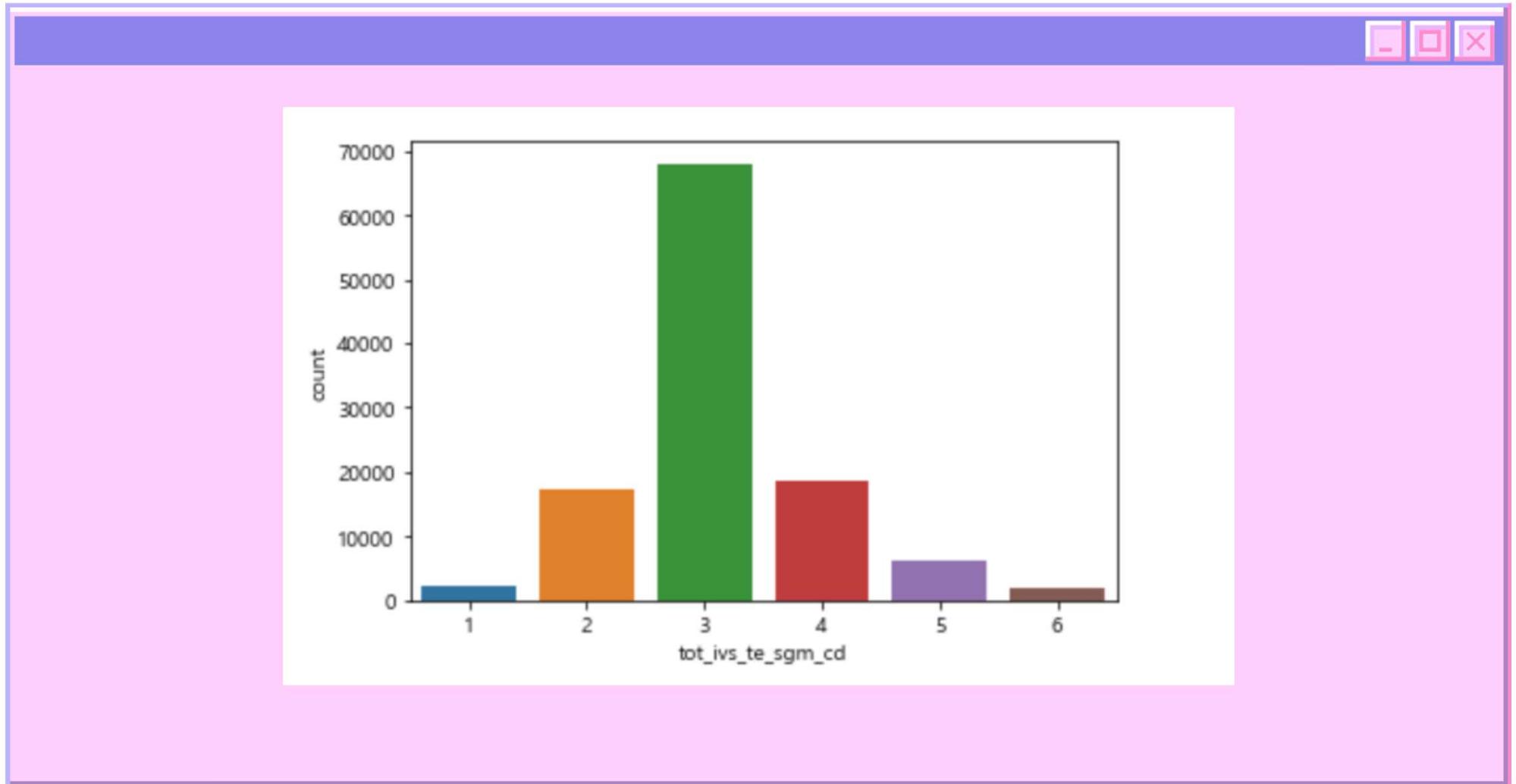
시각화. 45~50세 투자자들의 투자 성향



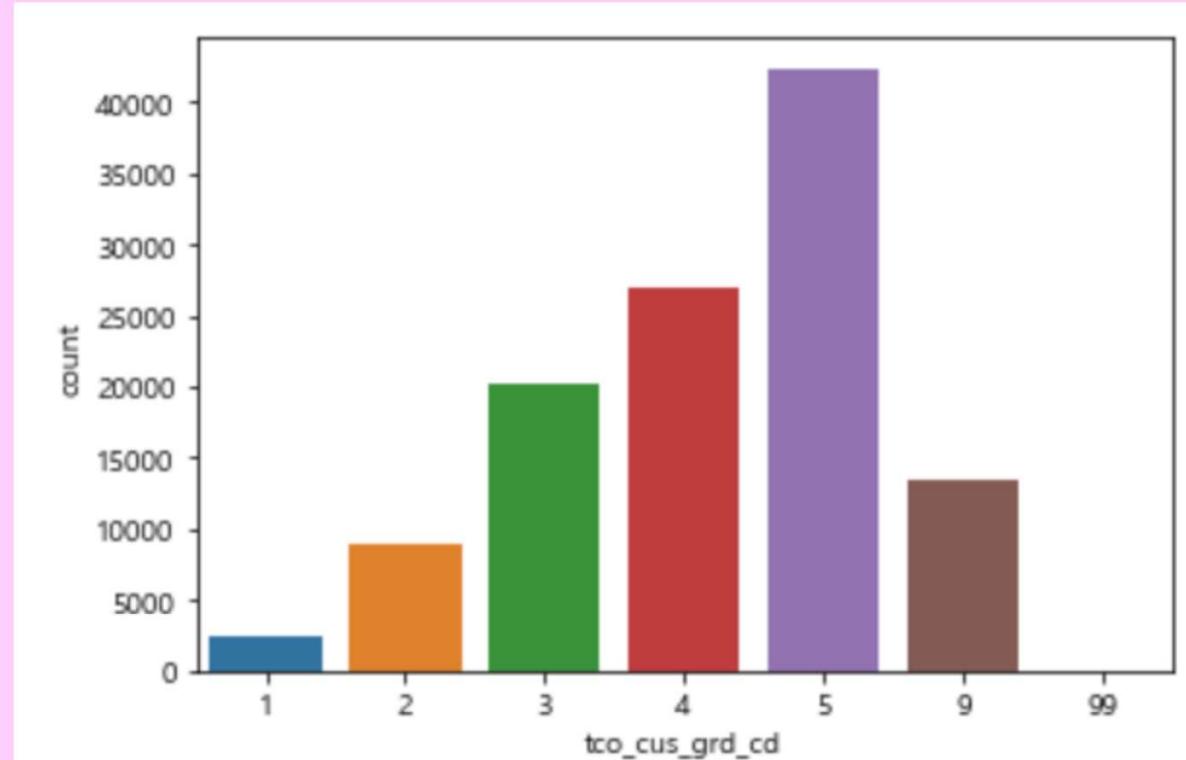
시각화. 45~50세 투자자들의 투자 종목



시각화. 45~50세 투자자들의 총 투자기간



시각화. 45~50세 투자자들의 고액등급





CHAPTER 93

MODELING

Deep Learning, RandomForest, LGBM, XGBR

MODELING

Deep Learning: 머신 러닝의 특정한 한 분야로서 연속된 층layer에서 점진적으로 의미 있는 표현을 배우는 데 강점이 있으며, 데이터로부터 표현을 학습하는 새로운 방식

- **파라미터**: Learning Rate, Cost Function, Regularization parameter, Mini-batch Size, Training Loop, Hidden Unit, Weight Initialization

RandomForest: 분류, 회귀 분석 등에 사용되는 양상을 학습 방법의 일종으로, 훈련 과정에서 구성한 다수의 결정 트리로부터 부류(분류) 또는 평균 예측치(회귀 분석)를 출력함

- **파라미터**: n_estimators, min_samples_split, min_samples_leaf, max_features, max_depth, max_leaf_nodes

MODELING

Deep Learning, RandomForest, LGBM, XGBR

MODELING

LGBM: 트리 기반의 학습 알고리즘인 Gradient Boosting 방식의 프레임 워크

- 파라미터: learning_rate, num_iterations, max_depth, boosting

XGBR: XGBoost는 Extreme Gradient Boosting의 약자

> Gradient Boost 을 병렬 학습이 지원되도록 구현한 라이브러리가 XGBoost

> Regression, Classification 문제를 모두 지원

- 파라미터: base_score, learning_rate, max_depth

CHAPTER 34

RESULT

Deep Learning, RandomForest, LGBM, XGBR

Result					
	train mse	train mae	test mse	test mae	제출 rmse
Deep Learning	34.3478	1.5709	1.9161	0.7433	80.3384
RandomForest	0.0216	0.0137	0.0543	0.0281	82.9963
LGBM	0.8604	0.3666	0.3055	0.339	77.3799
XGBR	428.0202		429.202	7.0703	82.5963

RESULT

Result

WINNER 1% 4% 10%

#	팀	팀 멤버	점수	제출
143	jihyheo		77.37993	7
1	Team CoH		53.75366	122

RESULT

각자 한 개의 모델을 선택해서 스스로 공부하는 과정에서 하이퍼 파라미터 설정, 모델 메커니즘 등을 배울 수 있는 시간이 됐다.

결과 부분에서 몇 가지 아쉬운 부분이 있었다. 파라미터 조정과 mae를 구하는 과정에서 난 오류를 해결하지 못한 게 아쉬웠다.



THANK YOU?