Kaggle Dataset을 이용한

당뇨병 예측

팀장 : 신주용

팀원 : 김주환, 박은영, 이도원, 진광환, 허우영

역할분담

신주용	팀장, 전처리, 모델링 코드 구현, 발표자료 준비
김주환	모델링 코드 도움, 발표자료 준비
박은영	EDA, 데이터 시각화, 발표자료 준비
이도원	EDA, 모델링, 발표
진광환	EDA, 데이터 시각화, 모델링 코드 구현
허우영	EDA, 데이터 시각화, 발표

Contents

01. 주제 및 Column 설명

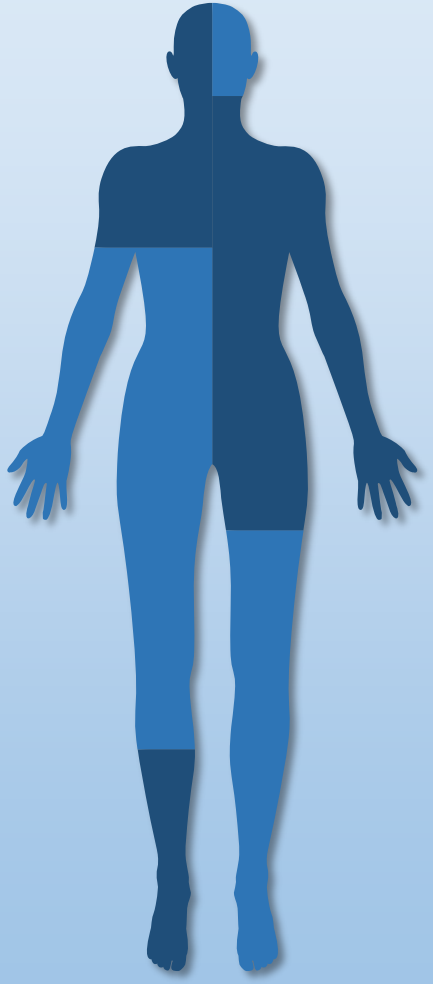
01. 주제소개

Diabetes, Hypertension and Stroke Prediction

70,652 survey responses from cleaned BRFSS 2015

Diabetes.csv만 이용

여러가지 요소들을 이용해 당뇨병을 예측



01. 데이터의 형태

Diabetes Data

	Age	Sex	HighChol	CholCheck	BMI	Smoker	HeartDiseaseorAttack	PhysActivity	Fruits	Veggies	HvyAlcoholConsump	GenHlth	MentHlth	PhysHlth	DiffWalk	Stroke	HighBP	Diabetes
0	13.0	0.0	1.0	1.0	32.0	1.0	1.0	1.0	1.0	1.0	0.0	3.0	0.0	0.0	0.0	0.0	1.0	1.0
1	10.0	0.0	1.0	1.0	22.0	0.0	0.0	1.0	0.0	0.0	0.0	4.0	10.0	20.0	0.0	0.0	1.0	1.0
2	8.0	1.0	1.0	1.0	35.0	0.0	0.0	0.0	1.0	0.0	0.0	3.0	5.0	0.0	0.0	0.0	1.0	1.0
3	9.0	1.0	0.0	1.0	24.0	0.0	1.0	0.0	0.0	0.0	0.0	3.0	0.0	0.0	0.0	1.0	0.0	1.0
4	10.0	0.0	1.0	1.0	34.0	1.0	0.0	0.0	0.0	0.0	0.0	4.0	30.0	0.0	1.0	0.0	1.0	1.0

01. Features 설명

Age	나이
Sex	성별 (0=여성, 1=남성)
HighChol	콜레스테롤 수치가 높은지? (0=no, 1=yes)
BMI	저체중 18.5미만, 정상 18.5~24.9, 과체중 25~29.9, 비만 30~
PhysActivity	physical activity in past 30 days - not including job 30일 내에 운동을 했는지 여부 (0=no, 1=yes)

01. Features 설명

Fruits	하루에 과일을 한번이상 먹는지 (0=no, 1=yes)
Veggies	하루에 채소를 한번이상 먹는지 (0=no, 1=yes)
HvyAlcoholConsump	한 주에 일정 횟수 이상 음주여부(남자: 14번 이상, 여자:7번 이상) (0=no, 1=yes)
GenHlth	Would you say that in general your health is : 평소 자신의 건강상태에 대한 답변 (1=excellent, 2=very good, 3=good, 4=fair, 5=poor
MentHlth	days of poor mental health scale 1-30 days (정신건강이 안 좋은 날 수)

01. Features 설명

PhysHlth	physical illness or injury days in past 30 days scale 1-30 (지난 30일 내 물리적 질환/부상일수)
DiffWalk	Do you have serious difficulty walking or climbing stairs? : 걷기나 계단 오르기에 어려움이 있는지 (0=no, 1=yes)
Stroke	you ever had a stroke. : 뇌졸증 걸린 적이 있는지 (0=no, 1=yes)
HighBP	혈압이 높은 지 (0=no, 1=yes)
HeartDiseaseorAttack	coronary heart disease (CHD) or myocardial infarction (MI) 코로나 심장질환 유무 or 심근경색 유무 (0=no, 1=yes)
Diabetes	당뇨병인지 (0=no, 1=yes)

02. EDA 시각화

02. EDA – 기술통계

각 열의 행수와 자료형

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 56553 entries, 0 to 56552
Data columns (total 18 columns):
 #   Column                Non-Null Count  Dtype  
---  ------                --------------  -----  
 0   Age                   56553 non-null  float64
 1   Sex                   56553 non-null  float64
 2   HighChol              56553 non-null  float64
 3   CholCheck             56553 non-null  float64
 4   BMI                   56553 non-null  float64
 5   Smoker                56553 non-null  float64
 6   HeartDiseaseorAttack  56553 non-null  float64
 7   PhysActivity          56553 non-null  float64
 8   Fruits                56553 non-null  float64
 9   Veggies               56553 non-null  float64
 10  HvyAlcoholConsump     56553 non-null  float64
 11  GenHlth               56553 non-null  float64
 12  MentHlth              56553 non-null  float64
 13  PhysHlth              56553 non-null  float64
 14  DiffWalk              56553 non-null  float64
 15  Stroke                56553 non-null  float64
 16  HighBP                56553 non-null  float64
 17  Diabetes              56553 non-null  float64
dtypes: float64(18)
```

결측치 존재여부

```
1 df.isnull().sum()
```

```
Age                     0
Sex                     0
HighChol                0
CholCheck               0
BMI                     0
Smoker                  0
HeartDiseaseorAttack    0
PhysActivity            0
Fruits                  0
Veggies                 0
HvyAlcoholConsump       0
GenHlth                 0
MentHlth                0
PhysHlth                0
DiffWalk                0
Stroke                  0
HighBP                  0
Diabetes                0
dtype: int64
```

Target의 요소와 각 데이터 수

```
df.Diabetes.value_counts()
```

```
1.0    28371
0.0    28182
Name: Diabetes, dtype: int64
```

02. EDA – 기술통계

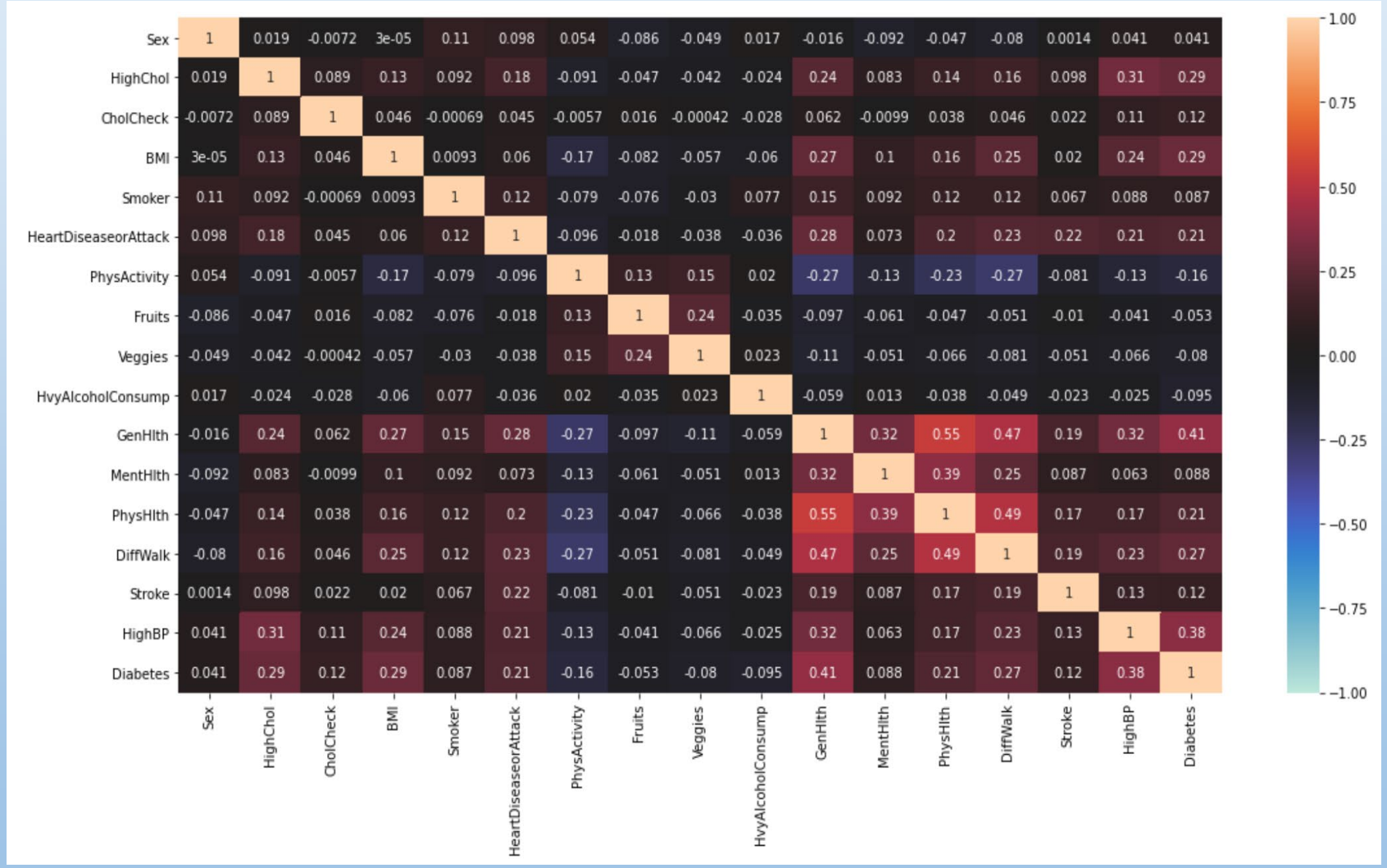
Data Description

	Age	Sex	HighChol	CholCheck	BMI	Smoker	HeartDiseaseorAttack	PhysActivity	Fruits	Veggies
count	56553.000000	56553.000000	56553.000000	56553.000000	56553.000000	56553.000000	56553.000000	56553.000000	56553.000000	56553.000000
mean	8.596131	0.457447	0.526179	0.975174	29.870122	0.475766	0.148922	0.702279	0.611108	0.788800
std	2.847163	0.498190	0.499319	0.155597	7.111446	0.499417	0.356015	0.457260	0.487503	0.408164
min	1.000000	0.000000	0.000000	0.000000	12.000000	0.000000	0.000000	0.000000	0.000000	0.000000
25%	7.000000	0.000000	0.000000	1.000000	25.000000	0.000000	0.000000	0.000000	0.000000	1.000000
50%	9.000000	0.000000	1.000000	1.000000	29.000000	0.000000	0.000000	1.000000	1.000000	1.000000
75%	11.000000	1.000000	1.000000	1.000000	33.000000	1.000000	0.000000	1.000000	1.000000	1.000000
max	13.000000	1.000000	1.000000	1.000000	98.000000	1.000000	1.000000	1.000000	1.000000	1.000000

HvyAlcoholConsump	GenHlth	MentHlth	PhysHlth	DiffWalk	Stroke	HighBP	Diabetes
56553.000000	56553.000000	56553.000000	56553.000000	56553.000000	56553.000000	56553.000000	56553.000000
0.042668	2.838753	3.732587	5.811752	0.252506	0.061995	0.564426	0.501671
0.202109	1.112712	8.140945	10.069800	0.434454	0.241148	0.495836	0.500002
0.000000	1.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
0.000000	2.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
0.000000	3.000000	0.000000	0.000000	0.000000	0.000000	1.000000	1.000000
0.000000	4.000000	2.000000	5.000000	1.000000	0.000000	1.000000	1.000000
1.000000	5.000000	30.000000	30.000000	1.000000	1.000000	1.000000	1.000000

02. EDA 시각화

- 독립변수 간
  상관관계가 보이고 있다.

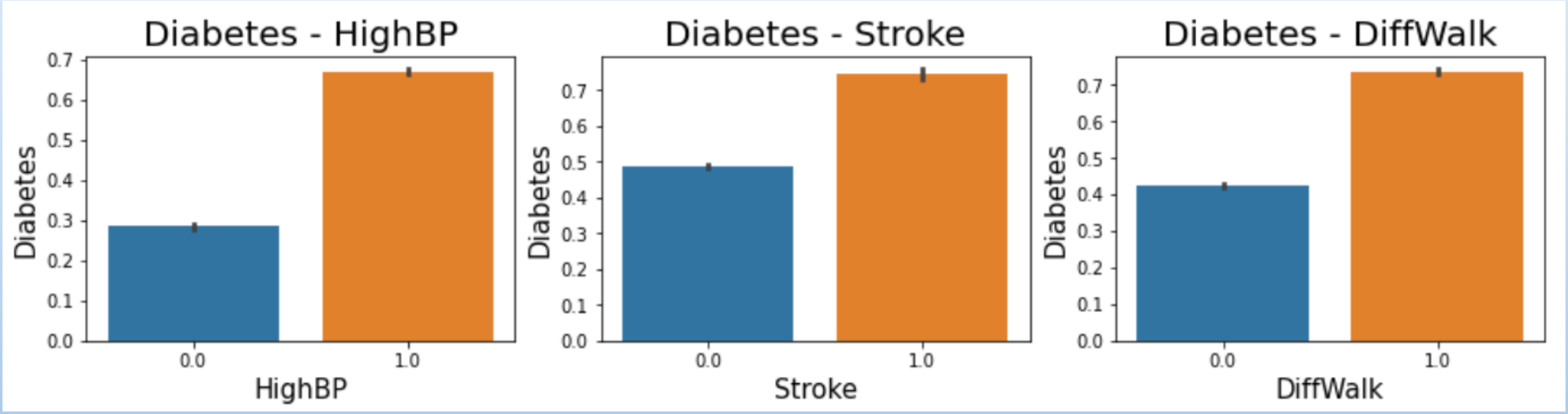


02. EDA 시각화

- 분산팽창요인이 10이하로 낮게 잡혀서 다중공선성이 존재.
- MentHlth, PhysHlth, DiffWalk column과의 상관관계가 매우 높다.
- GenHlth column 자체가 관측자의 주관에 따라 편향된 데이터일 가능성이 있다.

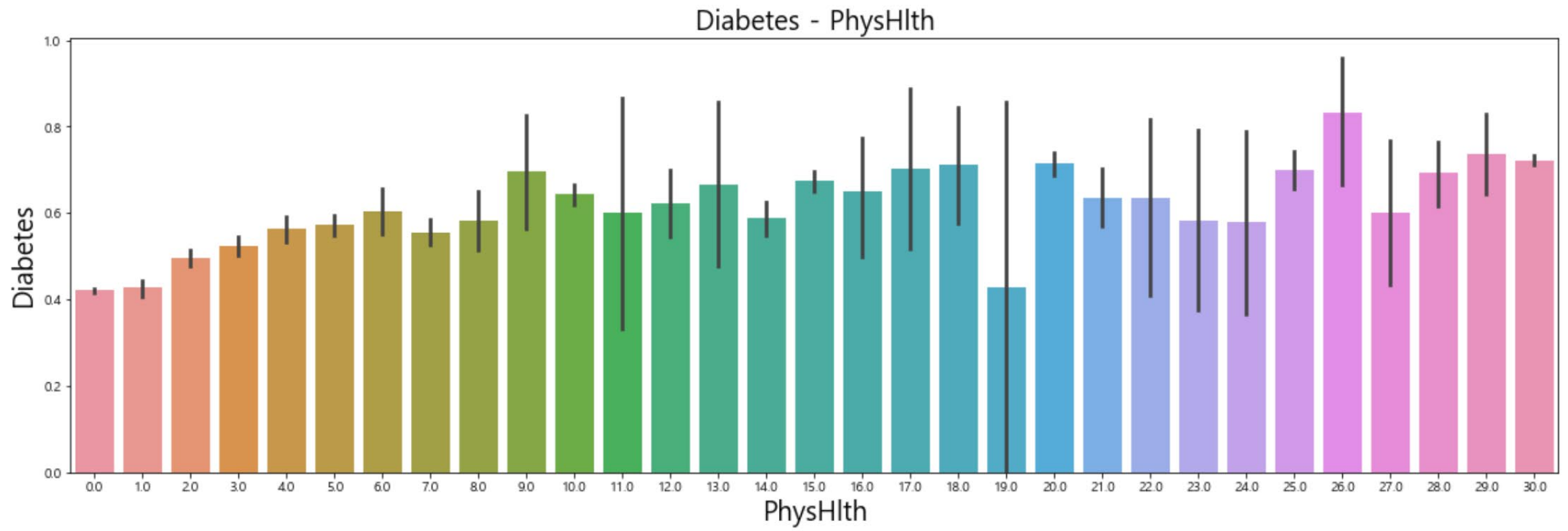
	VIF	Feature
0	9.205191	Age
1	1.877015	Sex
2	2.510930	HighChol
3	2.029575	Smoker
4	1.372672	HeartDiseaseorAttack
5	3.352395	PhysActivity
6	2.762006	Fruits
7	4.568134	Veggies
8	1.063320	HvyAlcoholConsump
9	10.073356	GenHlth
10	1.499843	MentHlth
11	2.224696	PhysHlth
12	2.034970	DiffWalk
13	1.161261	Stroke
14	3.102563	HighBP
15	2.769504	Diabetes

02. EDA 시각화



- 고혈압, 또는 뇌졸증이 있는 경우 당뇨병 발병률이 더 높은 양상을 보임
- 계단을 오를 때 불편함을 겪는 경우 당뇨병 발병률이 더 높은 양상을 보임

02. EDA 시각화

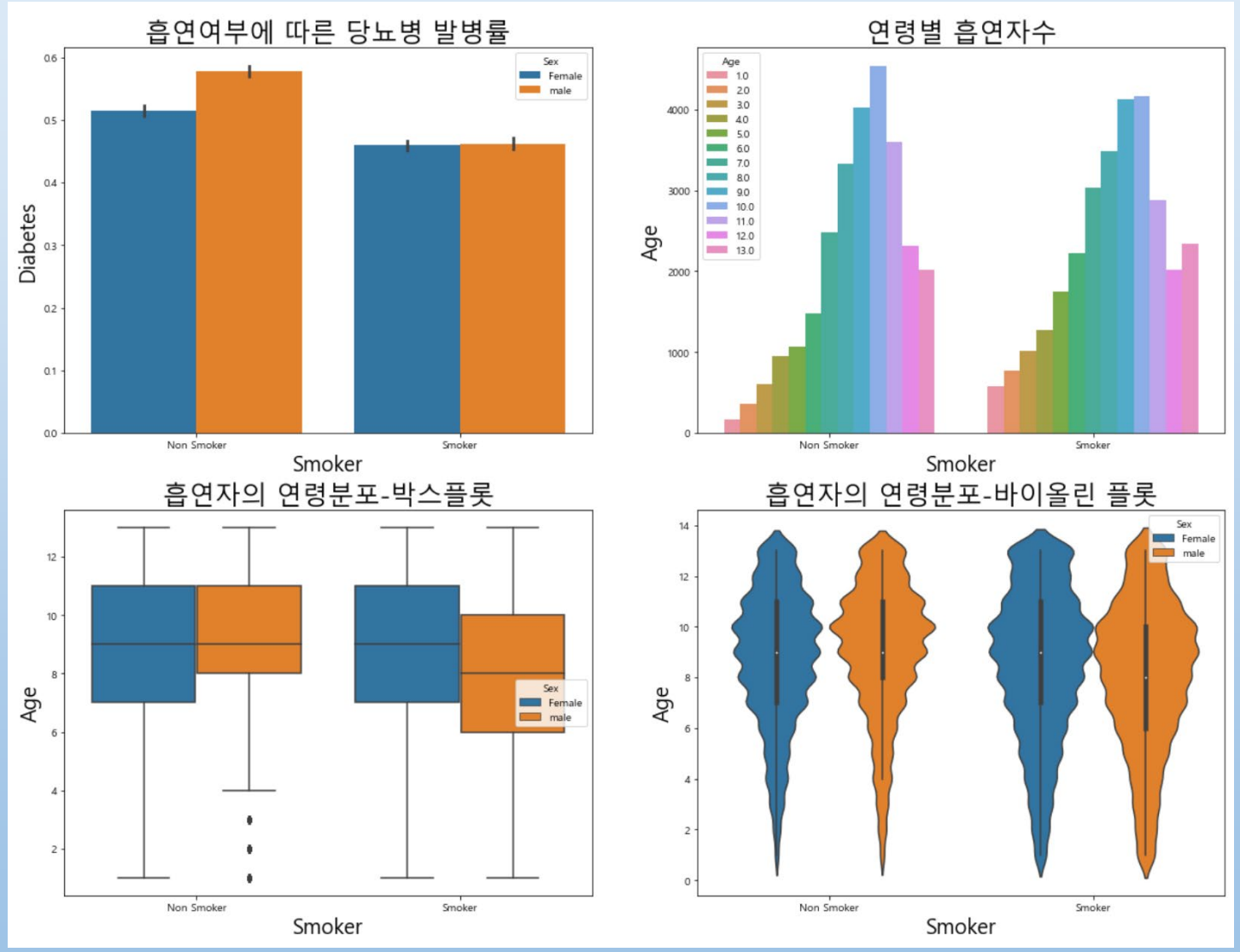


- 지난 한 달 동안 물리적인 부상을 입은 날 수가 증가할수록 당뇨병 발병률 또한 증가하는 양상을 보였다
- 시각화만으로는 상관관계를 파악하기가 어려움.

02. EDA 시각화

- 예상 외로 흡연자가 발병률이 더 낮았으나 실제 비율 상 차이는 미미한 것으로 보임.

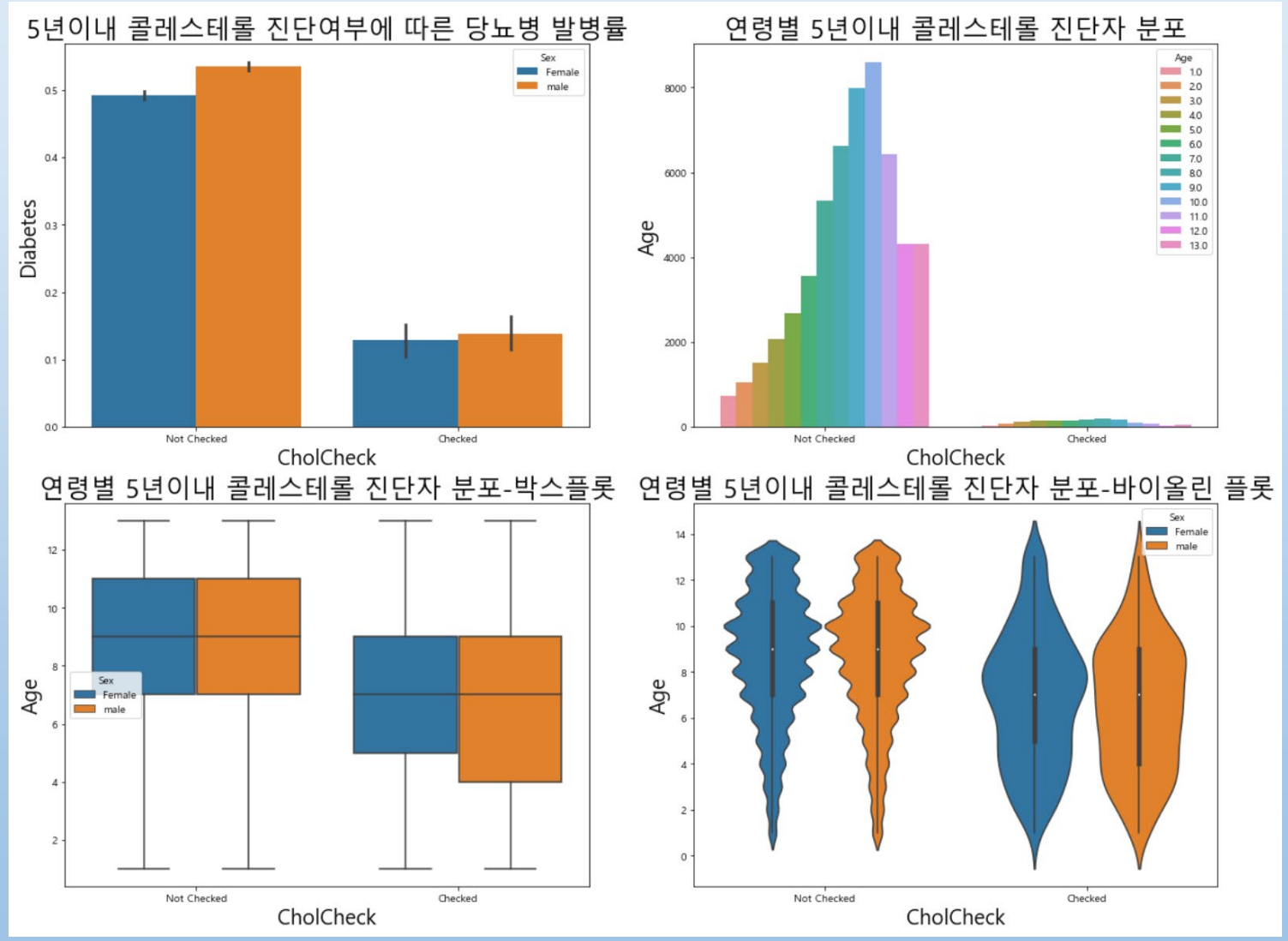
- 일반적으로 여자보다 남자가 당뇨병 발병률이 높은 경향성을 보임



02. EDA 시각화

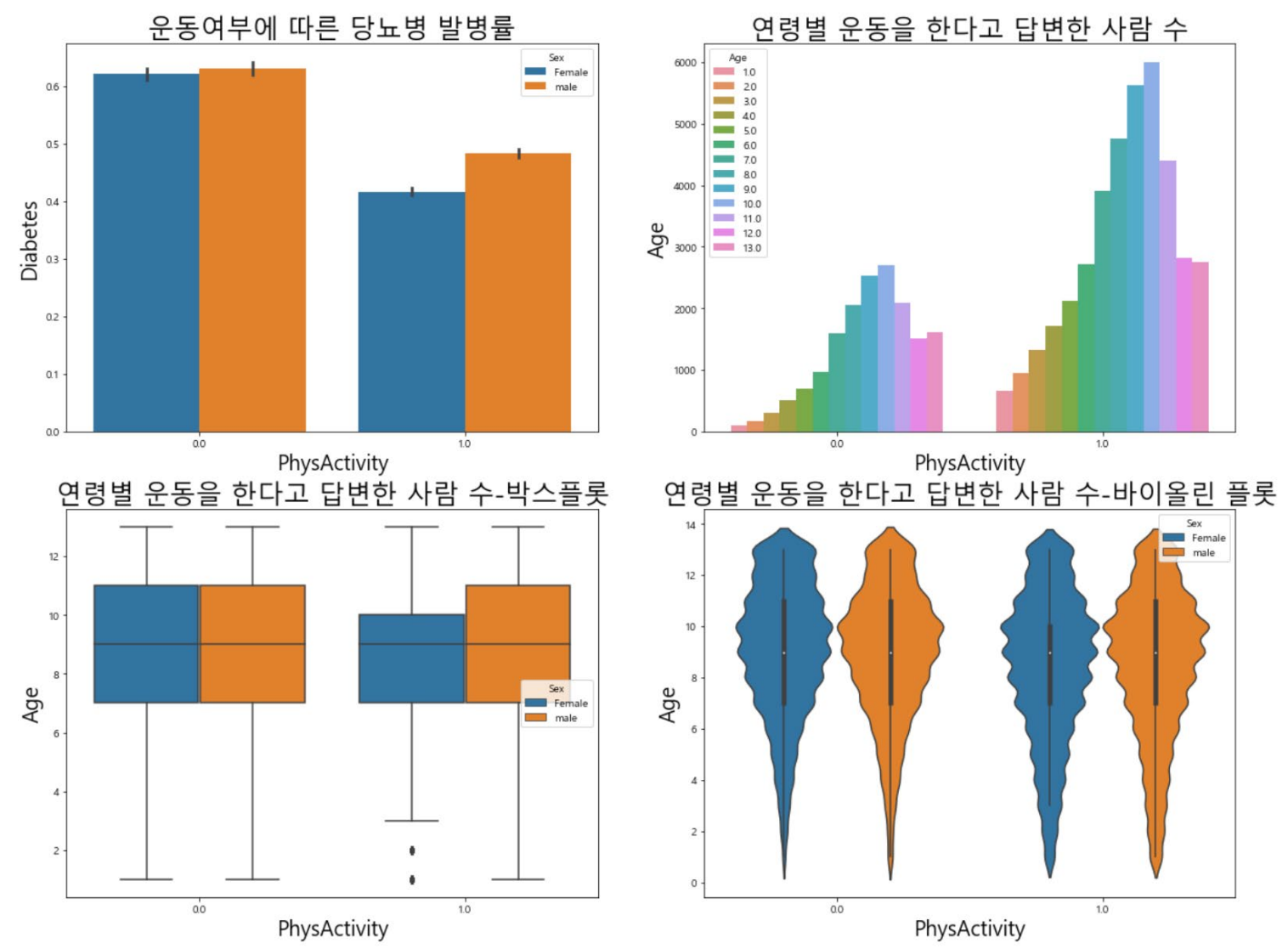
- 5년 이내에 콜레스테롤 진단을 받은 군이 당뇨병 발생률이 현저하게 낮음

- 콜레스테롤 진단을 받은 군은 그렇지 않은 군보다 연령분포가 고른 양상을 보임



02. EDA 시각화

- 30일 이내에 운동을 했을 때 당뇨병 발병률이 더 낮음



03. 머신러닝

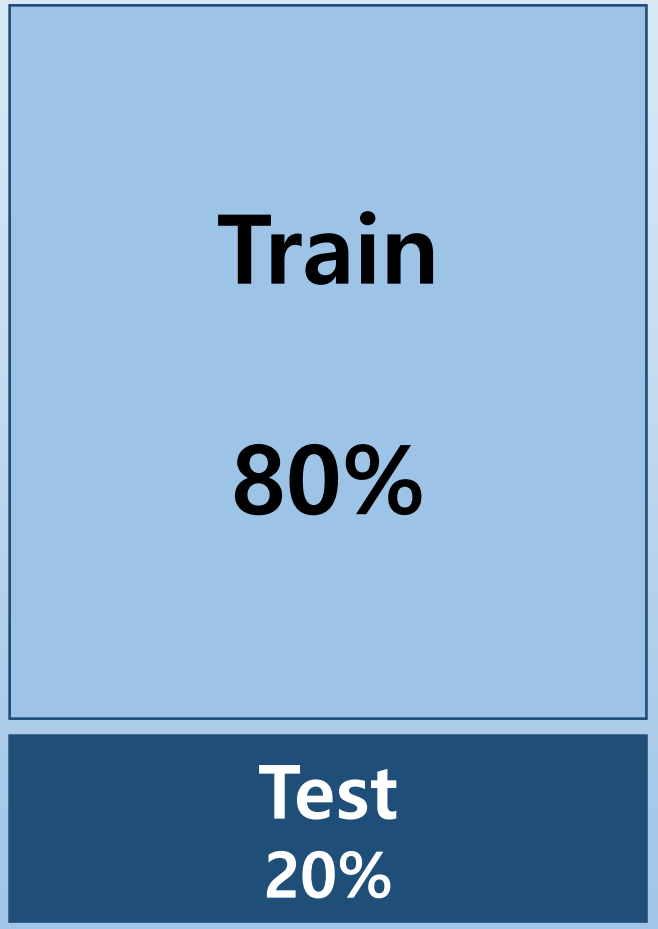
03. 머신러닝 - Preprocess

1. Age 열 삭제

2. train_test_split(test_size=0.2, random_state=33)

3. StandardScaler 적용

4. PCA 적용



Age column 삭제 이유

```
[192]
# Age column 삭제 했을 때
pca.explained_variance_ratio_.sum()
```

```
0.9021610178145532
```

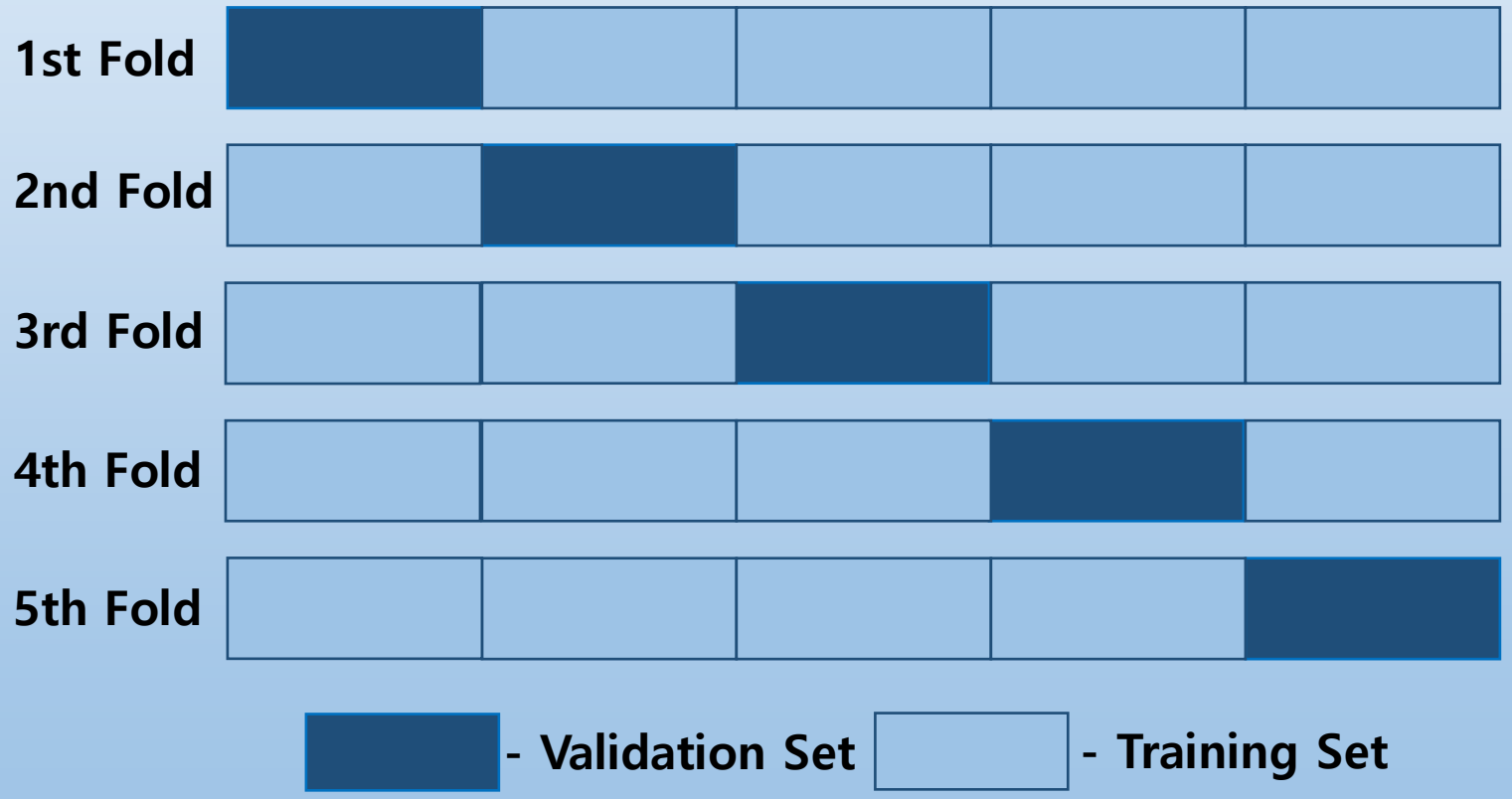
```
[5]
# Age column 삭제 안했을 때
pca.explained_variance_ratio_.sum()
✓ 0.0s
```

```
0.8762144188254798
```

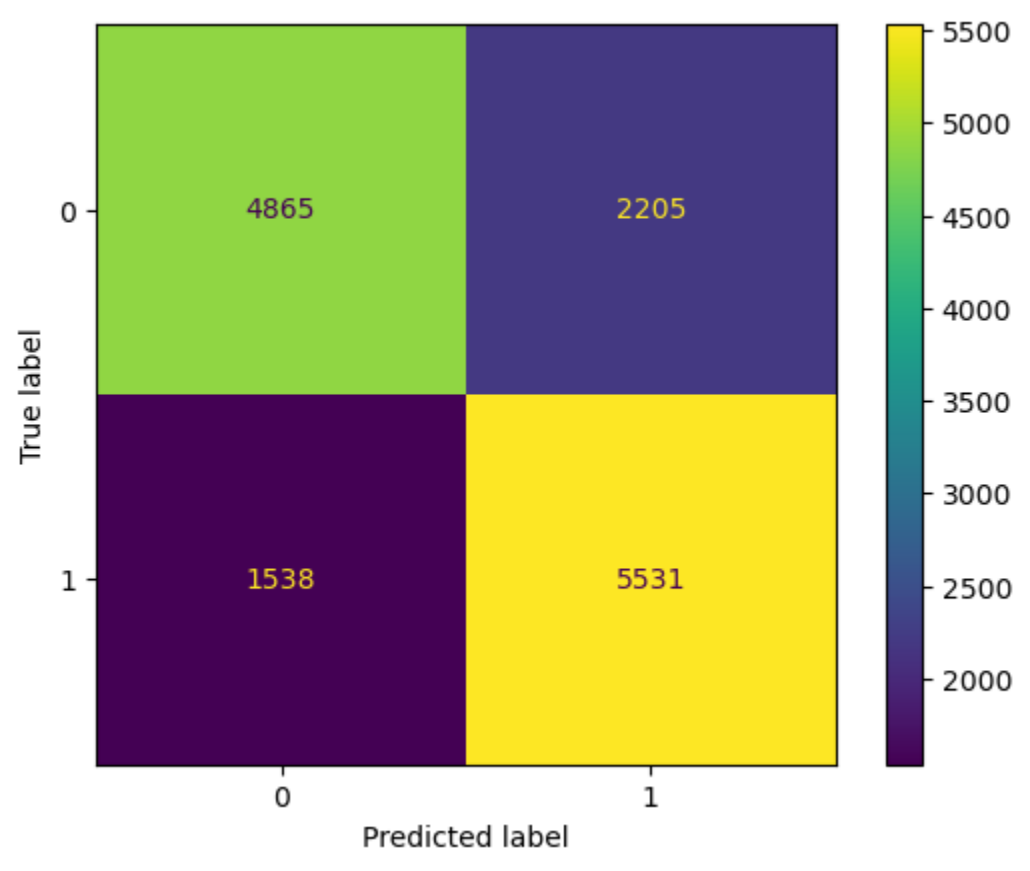
03. 머신러닝 - 사용한 머신러닝 모델

- GBM
- AdaBoost
- Bagging
- RandomForest
- SupportVectorMachine
- LogisticRegression
- XGBoost
- LightGBM

모든 모델에 Kfold n_split=5 적용



03. 머신러닝 - GBM



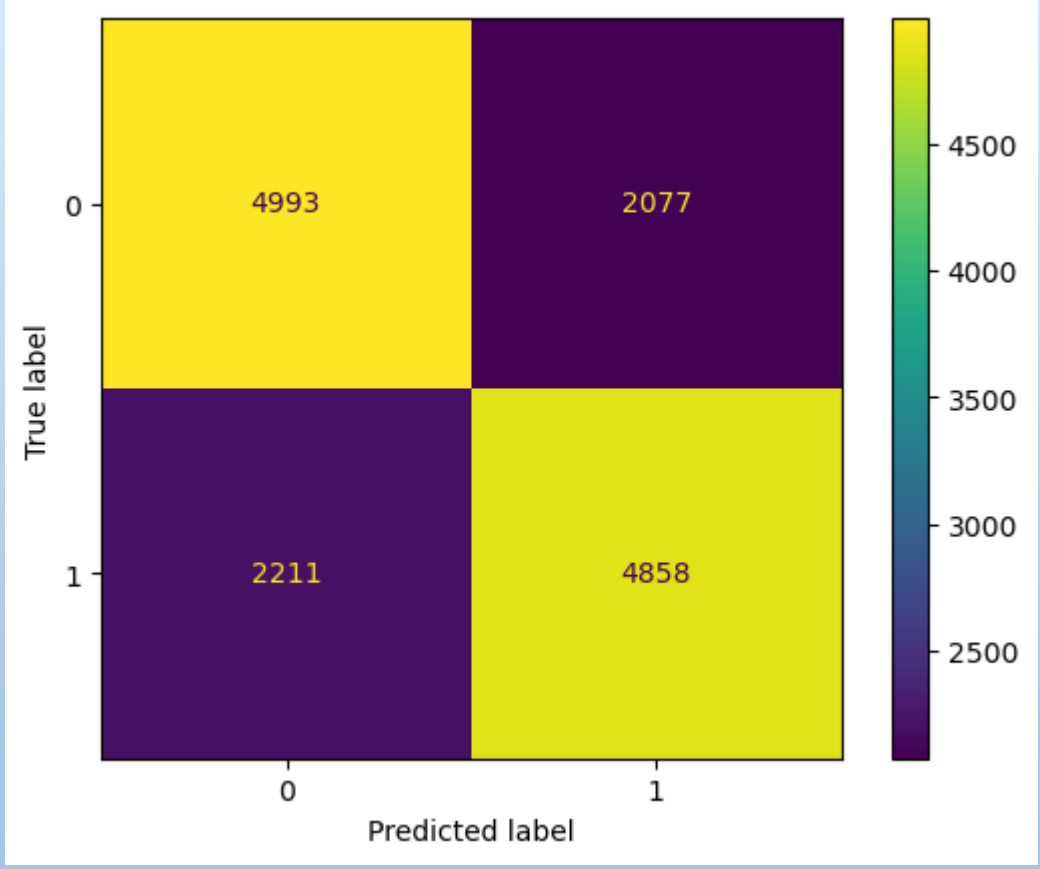
accuracy_score : 0.7352

recall_score : 0.7824

precision_score : 0.7149

f1 score : 0.7471

03. 머신러닝 - Bagging



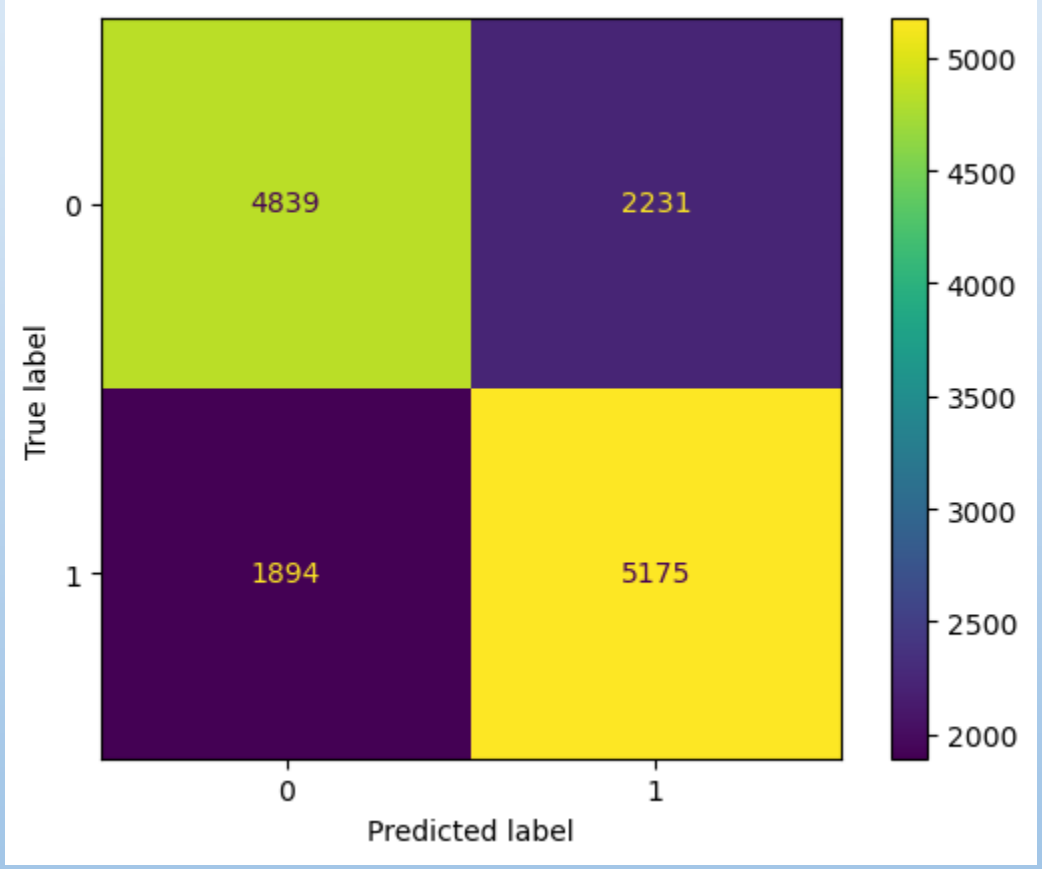
accuracy_score : 0.6967

recall_score : 0.6872

precision_score : 0.7005

f1 score : 0.6938

03. 머신러닝 - RandomForest



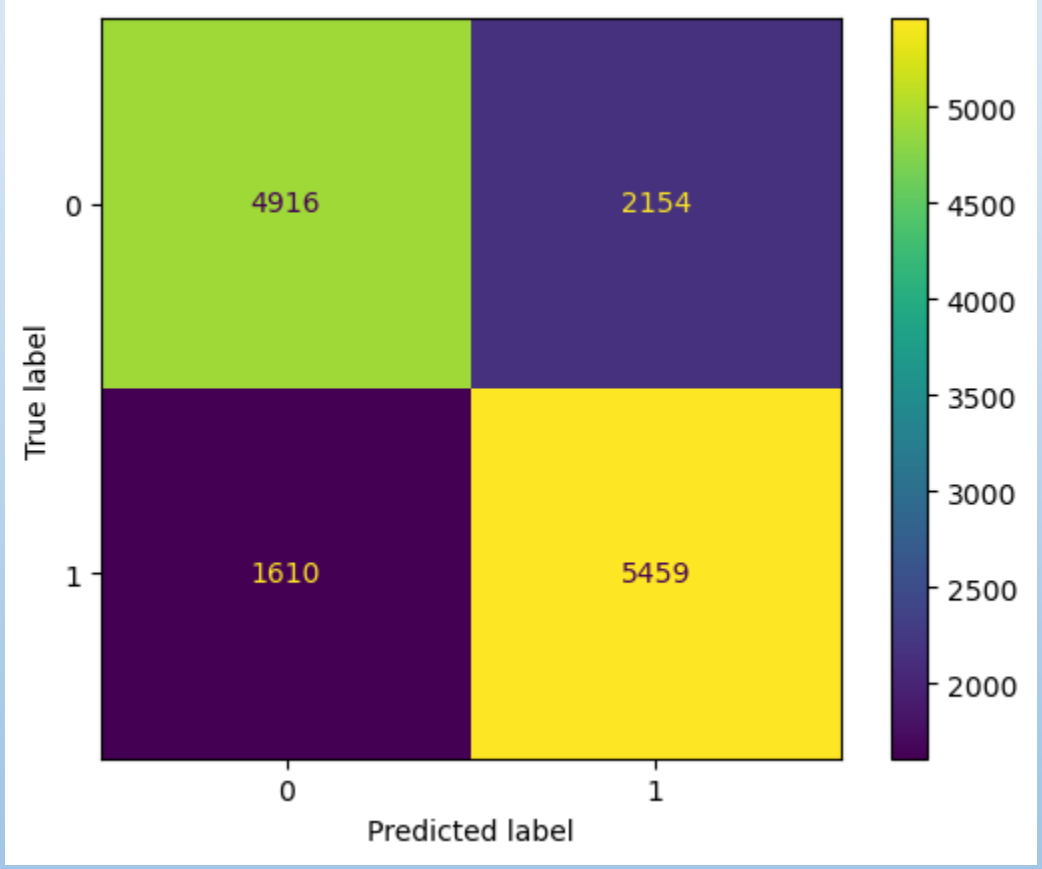
accuracy_score : 0.7082

recall_score : 0.7320

precision_score : 0.6987

f1_score : 0.7150

03. 머신러닝 - SupportVectorMachine



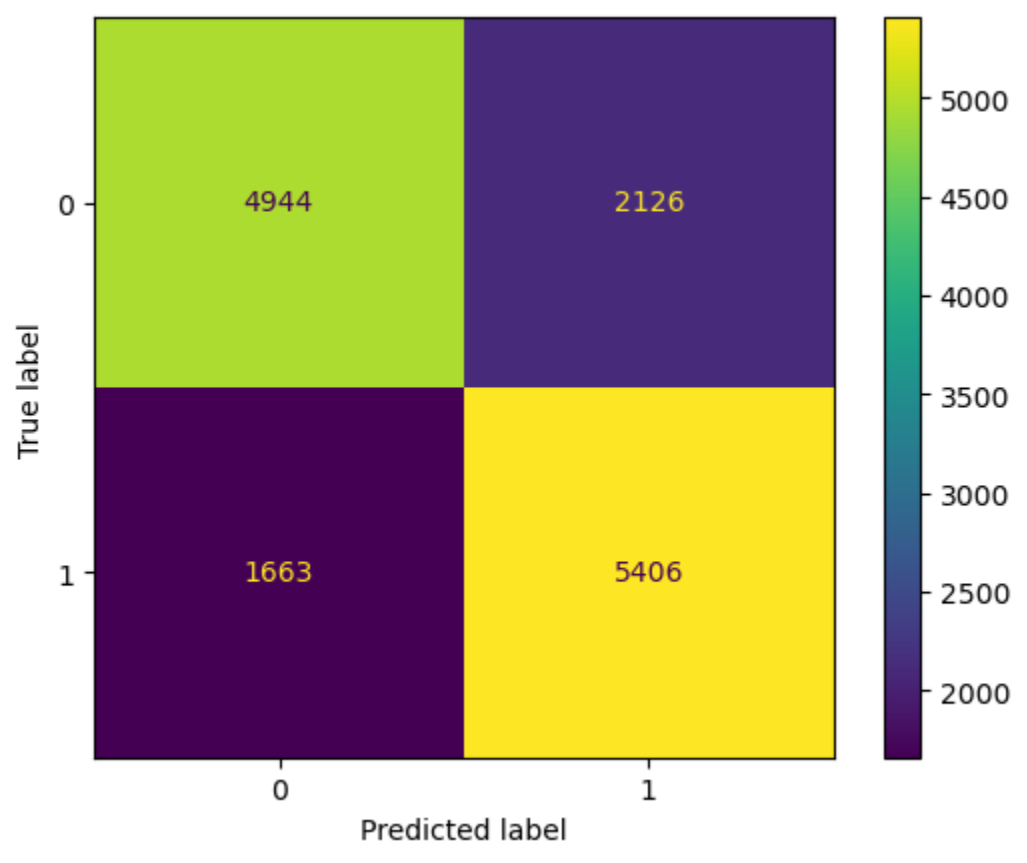
accuracy_score : 0.7337

recall_score : 0.7722

precision_score : 0.7170

f1_score : 0.7436

03. 머신러닝 - AdaBoost



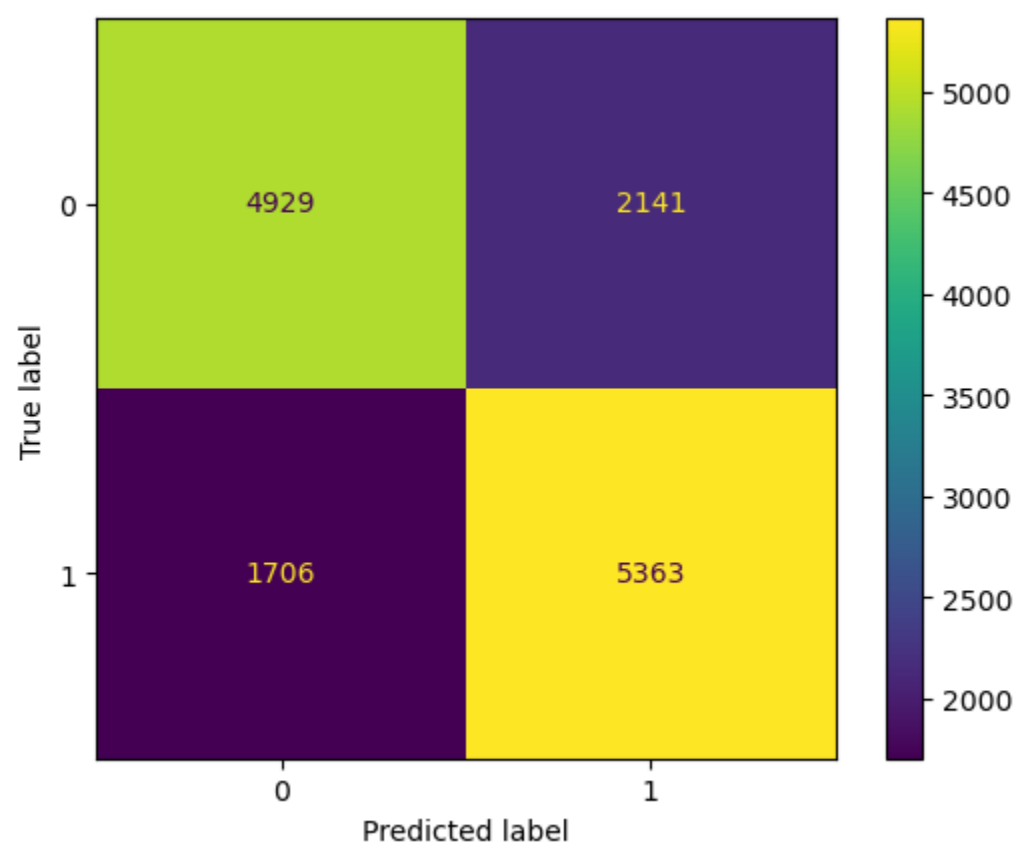
accuracy_score : 0.7320

recall_score : 0.7647

precision_score : 0.7177

f1 score : 0.7404

03. 머신러닝 - LightGBM



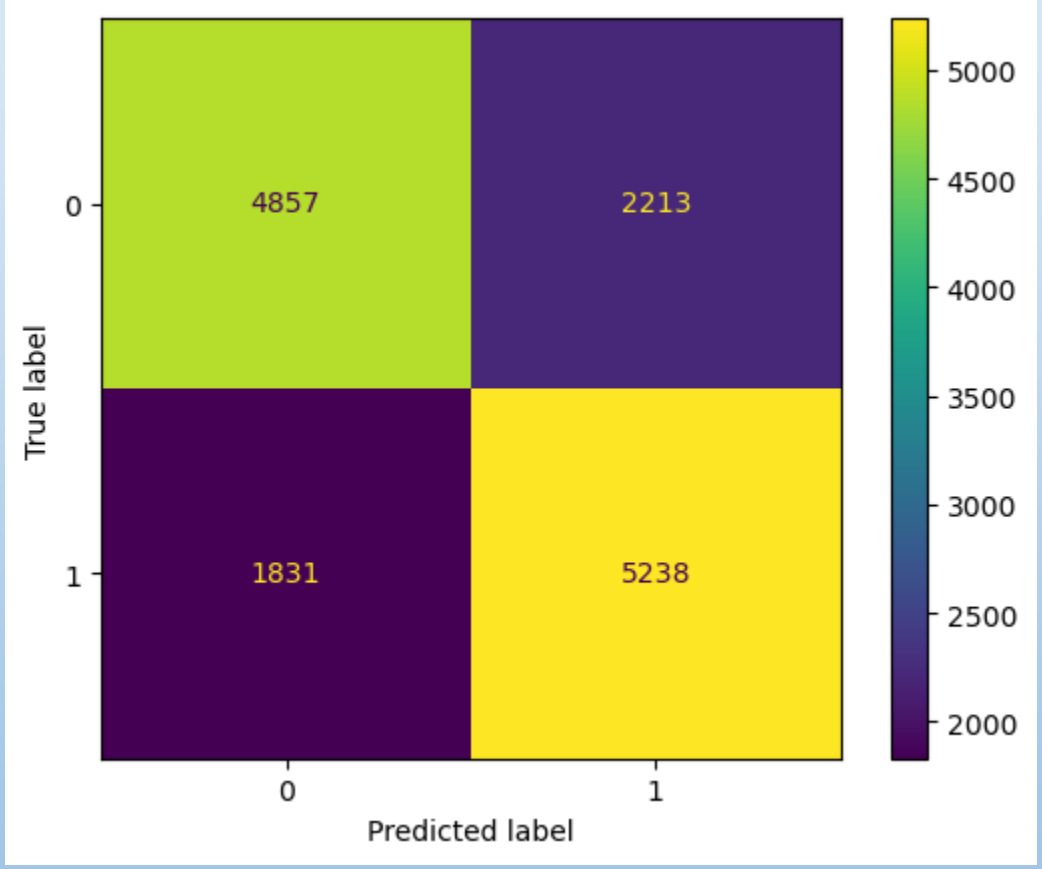
accuracy_score : 0.7279

recall_score : 0.7586

precision_score : 0.7146

f1_score : 0.7360

03. 머신러닝 - XGBoost



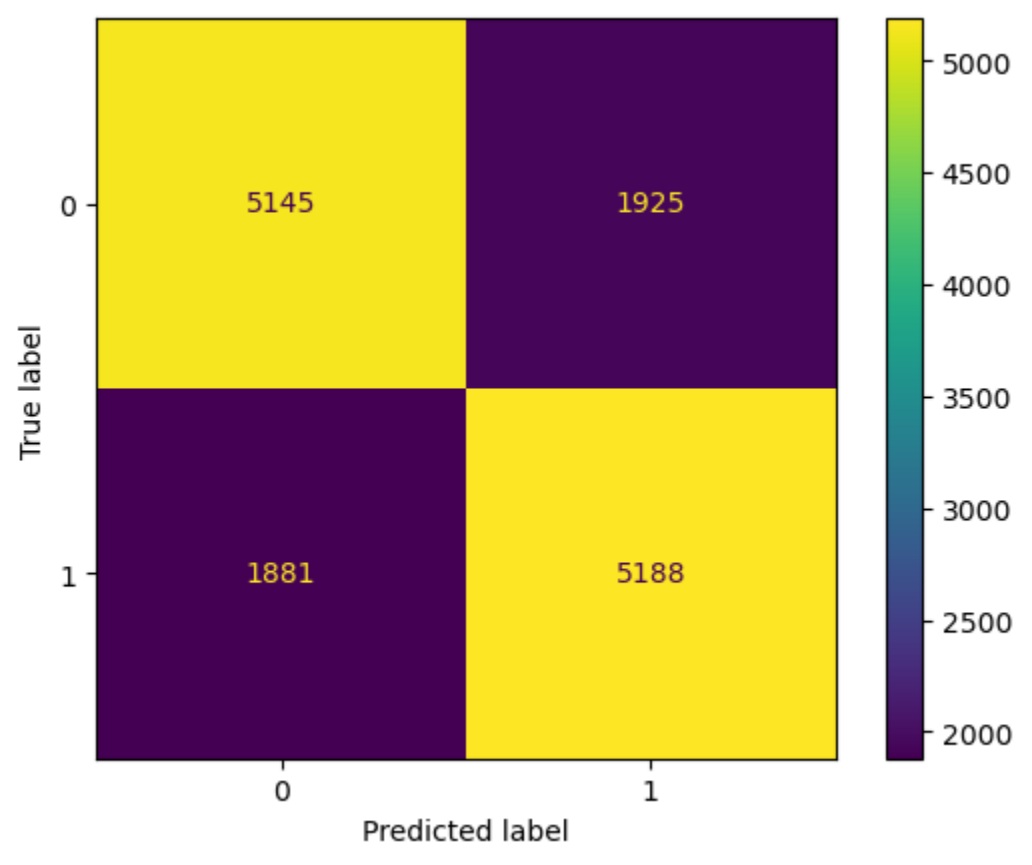
accuracy_score : 0.7139

recall_score : 0.7409

precision_score : 0.7029

f1_score : 0.7214

03. 머신러닝 - LogisticRegression



accuracy_score : 0.7308

recall_score : 0.7339

precision_score : 0.7293

f1_score : 0.7316

03. 머신러닝 - Best ML Models

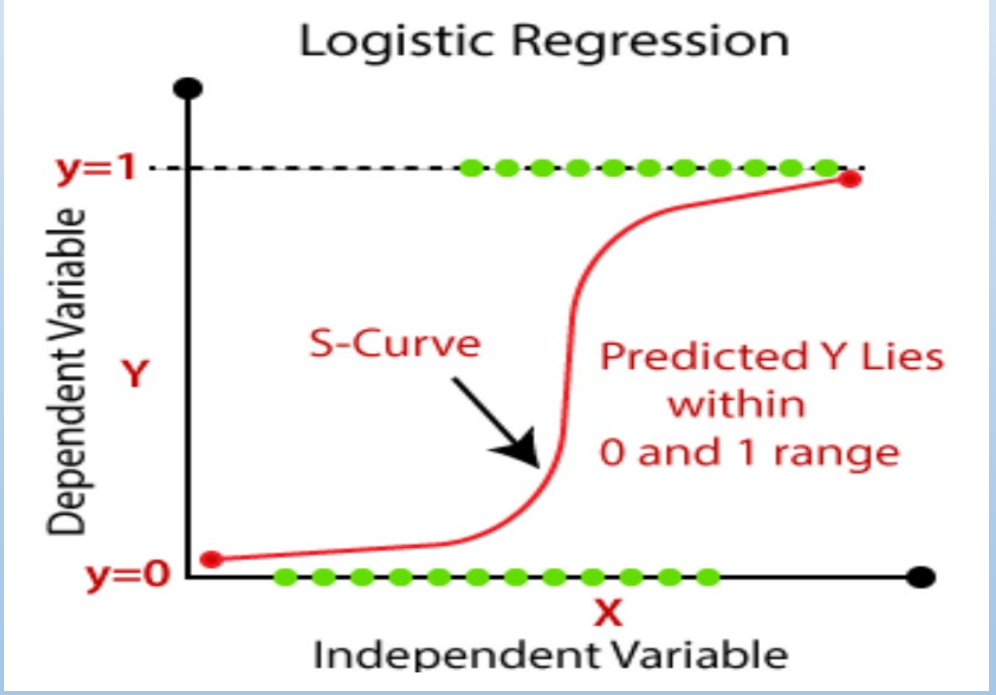
LogisticRegression

accuracy_score : 0.7351

recall_score : 0.7415

precision_score : 0.7321

f1_score : 0.7368



03. 머신러닝 - Pycaret

Top 3 Models - Python AutoML library 활용

1. Gradient Boosting Classifier

Acc : 0.7528 Recall : 0.7977 Prec : 0.7319 F1 : 0.7634

2. Light Gradient Boosting Machine

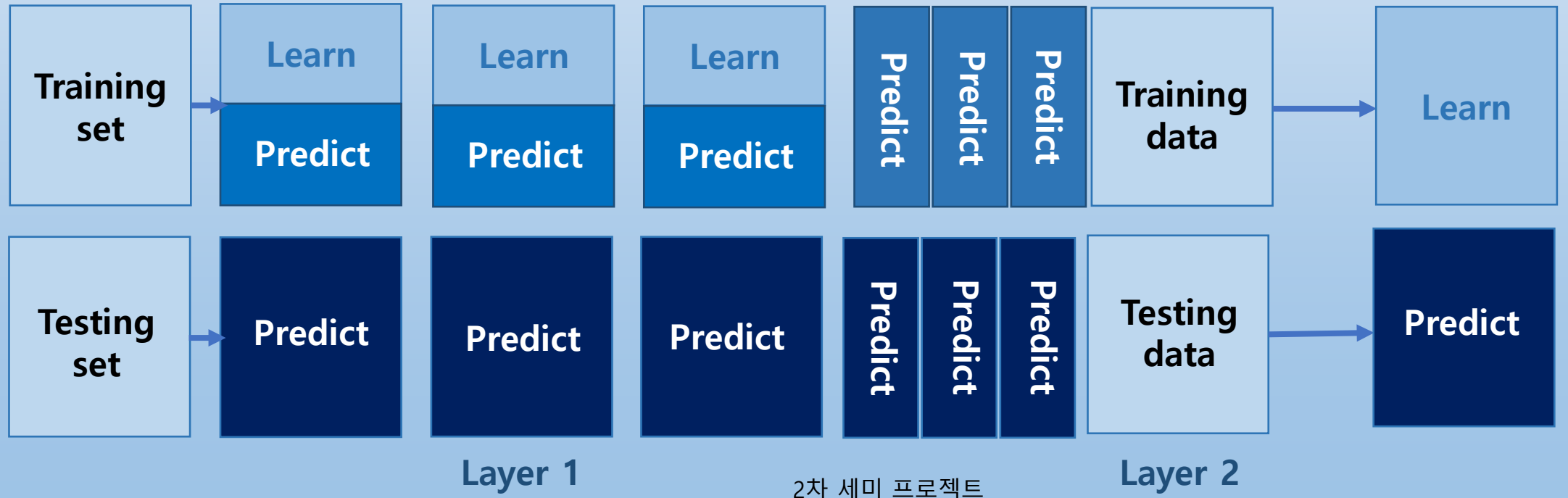
3. Ada Boost Classifier

03. 머신러닝 - Pycaret Blending

Top 3 모델을 혼합하여 더 정확도가 높은 모델을 만들기

블렌딩 평가 - Top 1 모델보다 정확도가 약간 떨어졌다. (-0.0003)

Accuracy : 0.7525, Recall : 0.7966, Prec : 0.7320, F1 : 0.7629



03. 머신러닝 - Pycaret 모델 Finalized

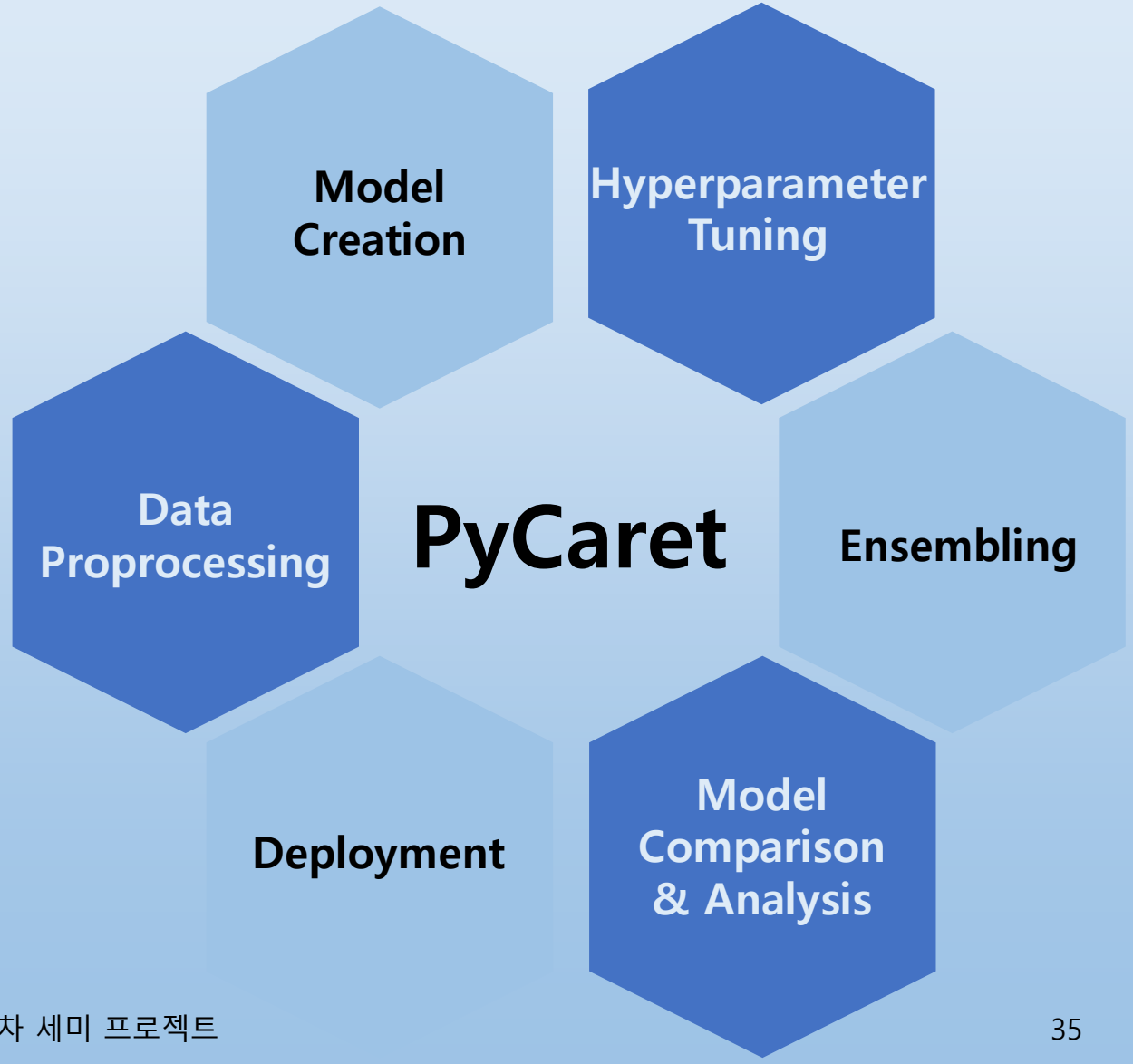
마감된 모델 평가

Accuracy : 0.7570

Recall : 0.7987

Prec : 0.7372

F1 : 0.7667



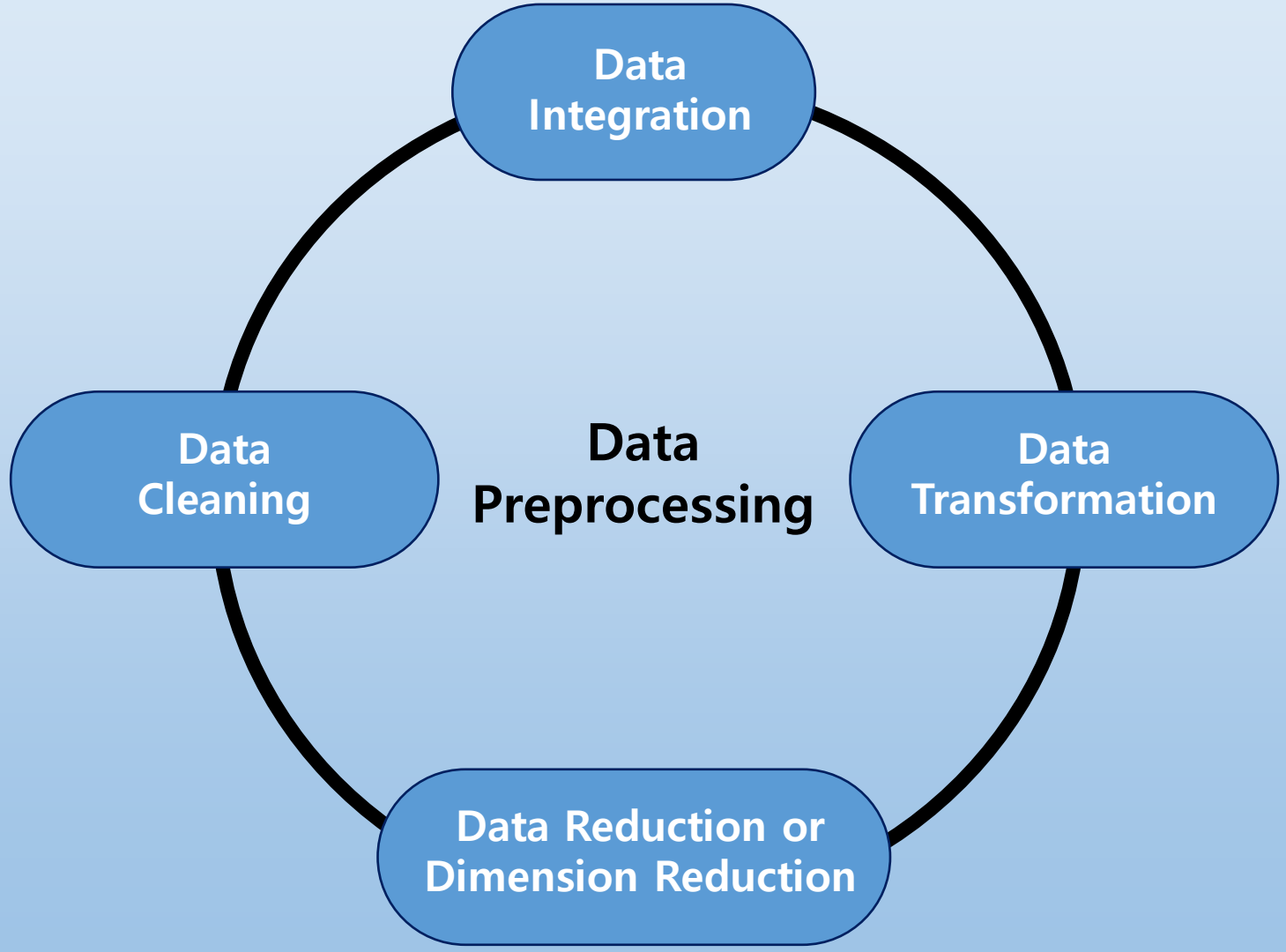
04. 딥러닝

04. 딥러닝 - Preprocess Case 1

1. Age 열 삭제

2. StandardScaler 적용

3. PCA 적용

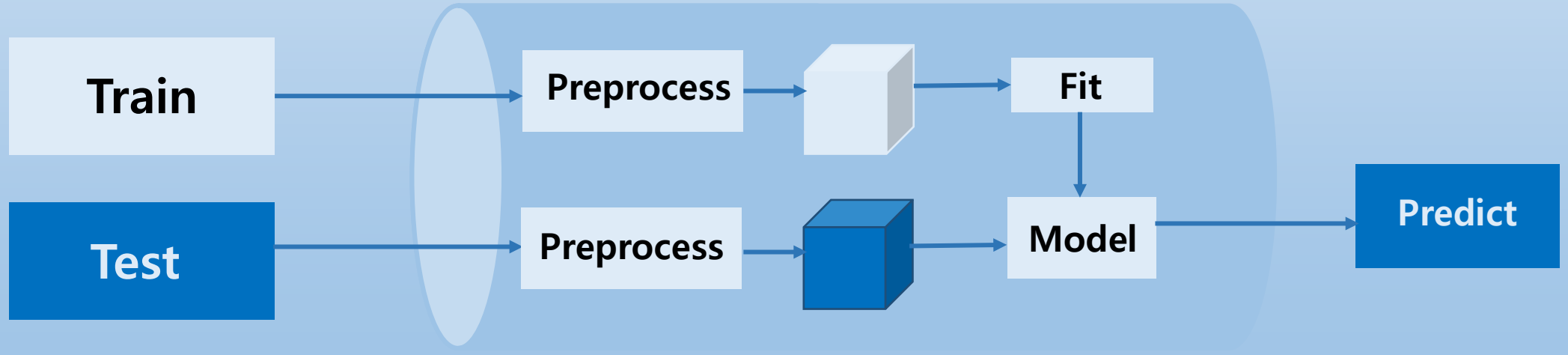


04. 딥러닝 - Preprocess Case 2

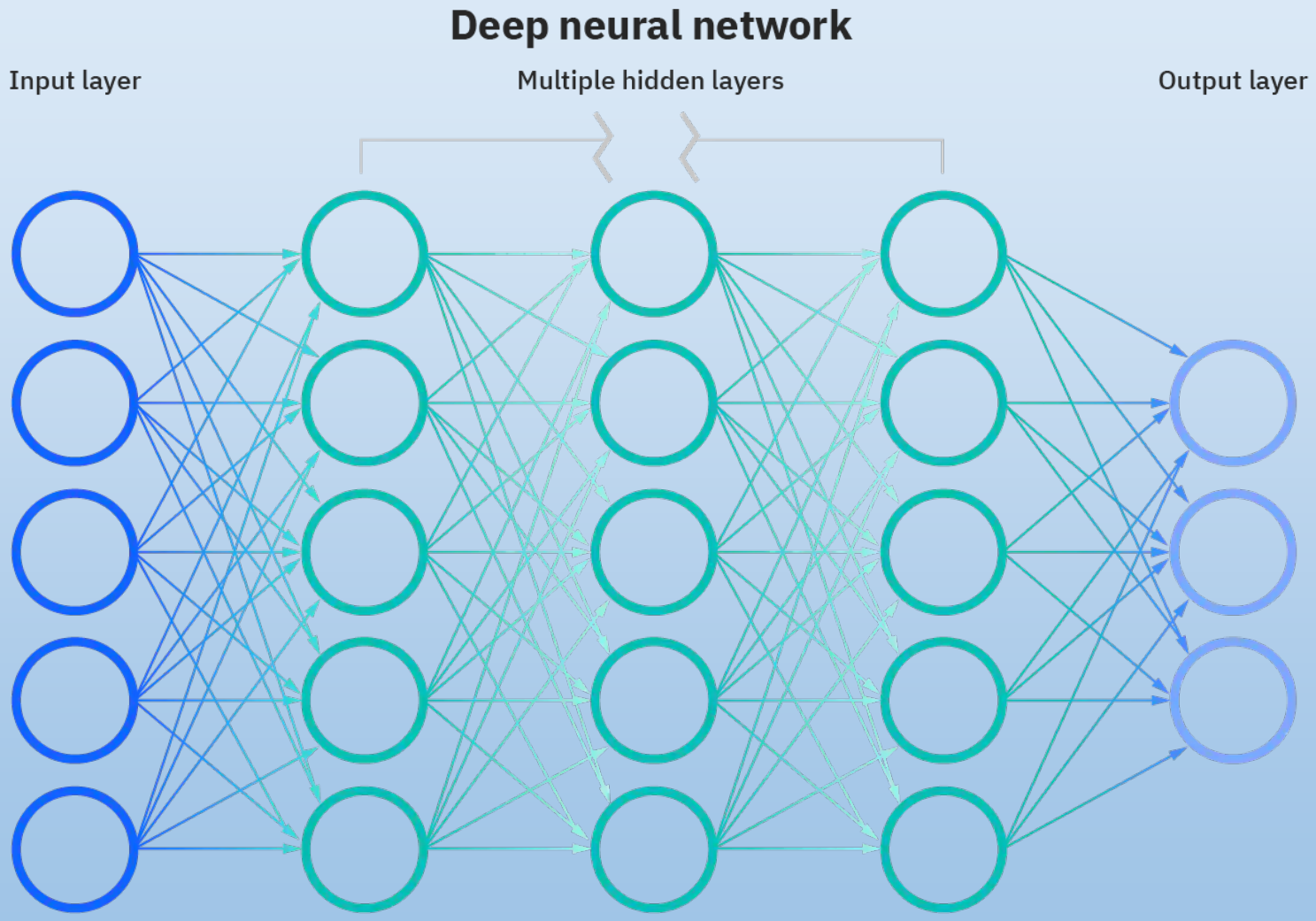
Pipeline 사용

StandardScaler 적용 column : Age, BMI, MentHlth, PhysHlth

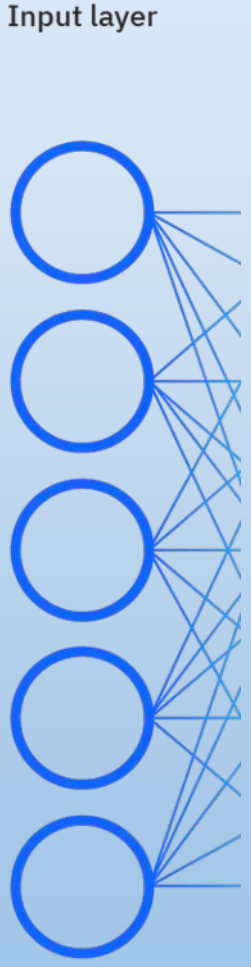
One-Hot Encoding 적용 column : 그 외 모든 열



04. 딥러닝 - 인공신경망 모델



04. 딥러닝 – Layer 1. Input layer (Dense)



Input Dimension = 14

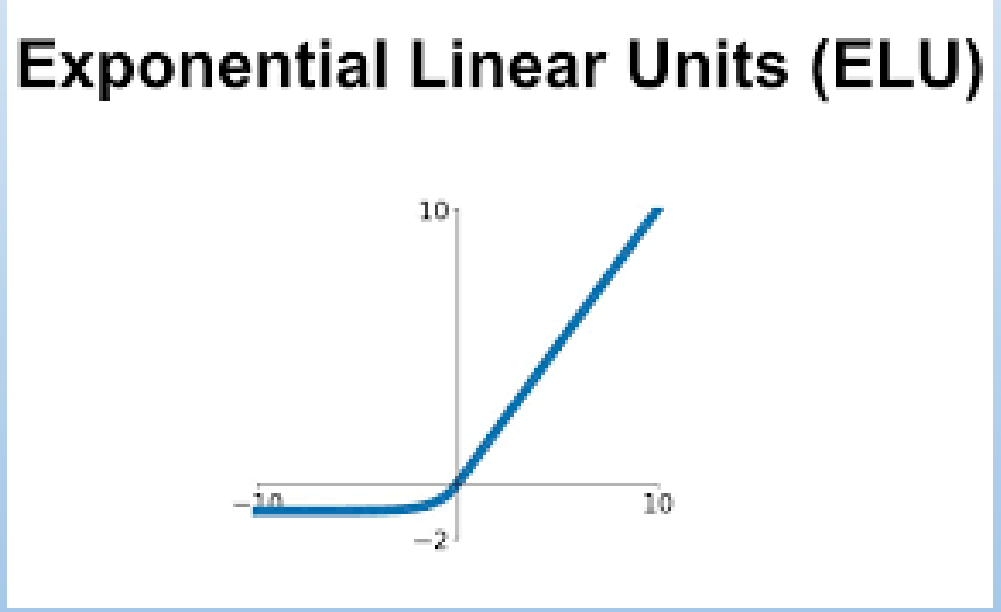
Units = 300

Kernel Initializer = He Uniform

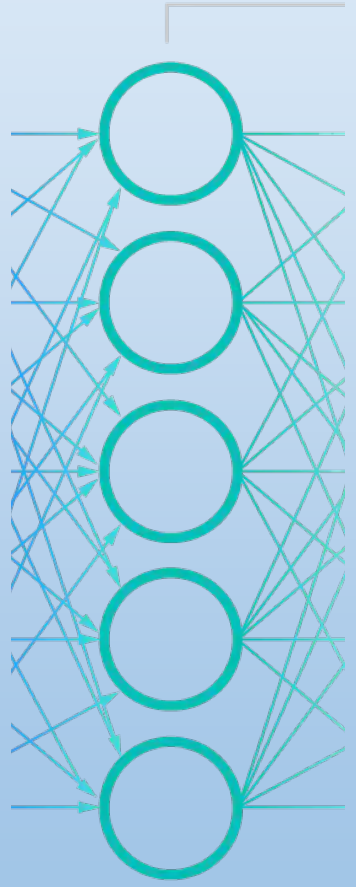
Batch Normalization

Activation = ELU

Drop Out = 0.2



04. 딥러닝 - Layer 2. Hidden layer A (Dense)



Units = 200

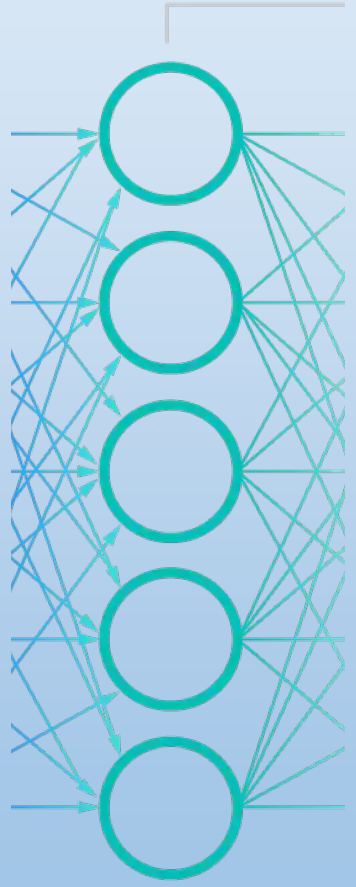
Kernel Initializer = He Uniform

Batch Normalization

Activation = ELU

Drop Out = 0.2

04. 딥러닝 - Layer 3. Hidden layer B (Dense)



Units = 100

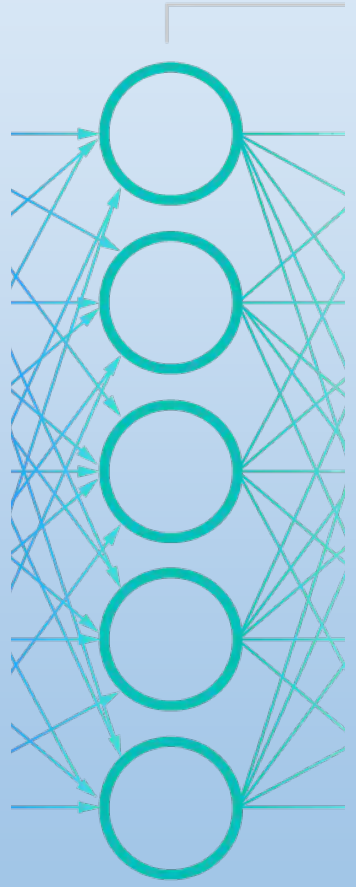
Kernel Initializer = He Uniform

Batch Normalization

Activation = ELU

Drop Out = 0.2

04. 딥러닝. Layer 4. Hidden layer C (Dense)



Units = 50

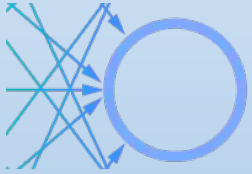
Kernel Initializer = He Uniform

Batch Normalization

Activation = ELU

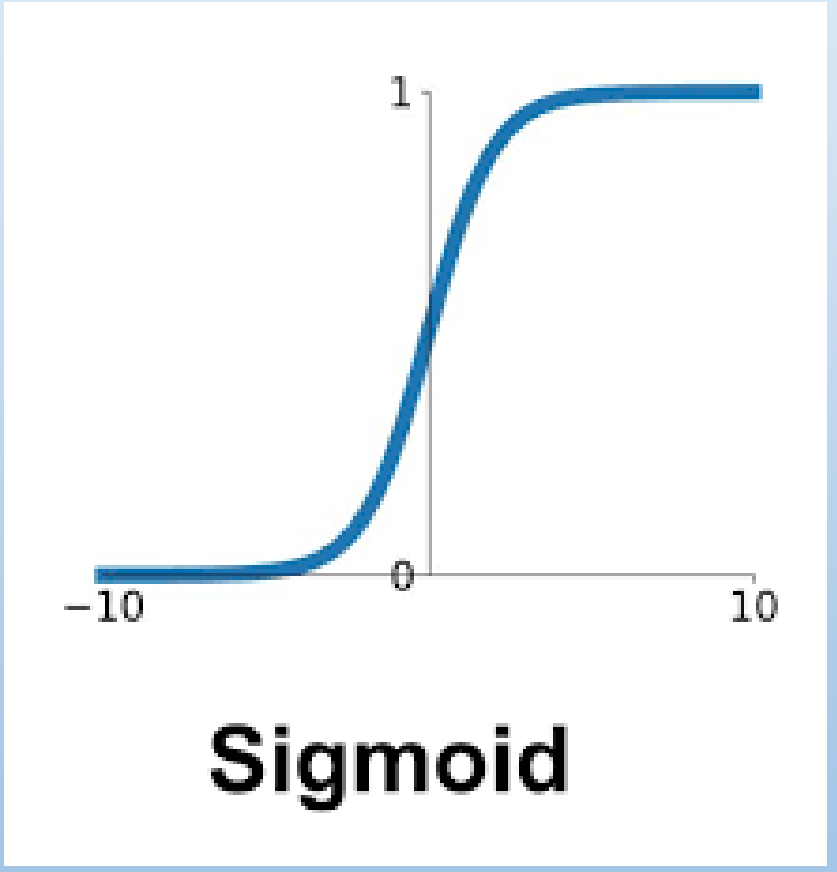
Drop Out = 0.2

04. 딥러닝 - Layer 5. Output layer (Dense)



Units = 1

Activation = Sigmoid



04. 딥러닝 - Compile

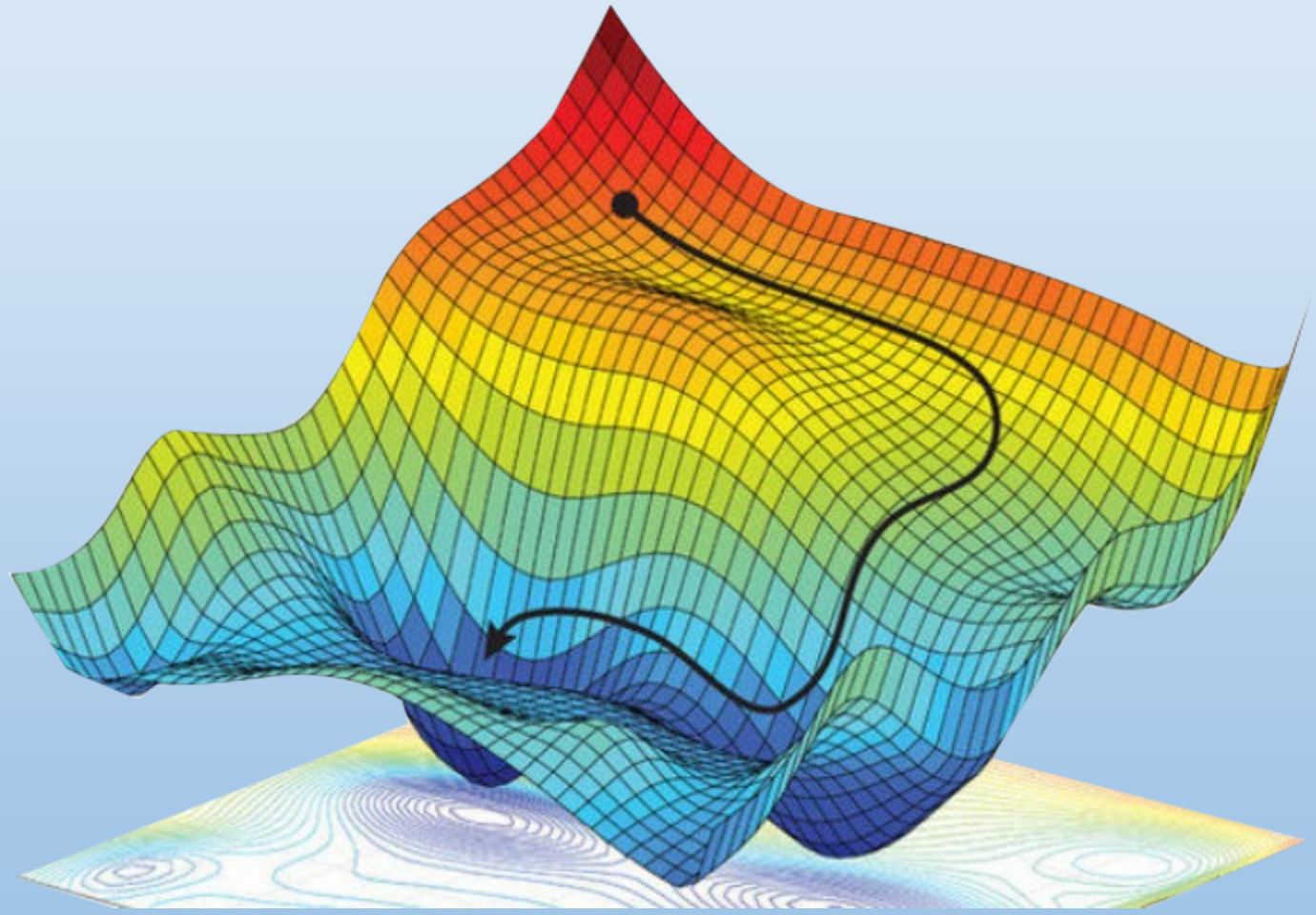
Optimizer = Adam

Loss = Binary Crossentropy

Metrics 1 = Binary_accuracy

Metrics 2 = Recall

Metrics 3 = Precision



04. 딥러닝 - 모델 Fit

Batch Size = 300

Epochs = 20

Validation Split = 0.2

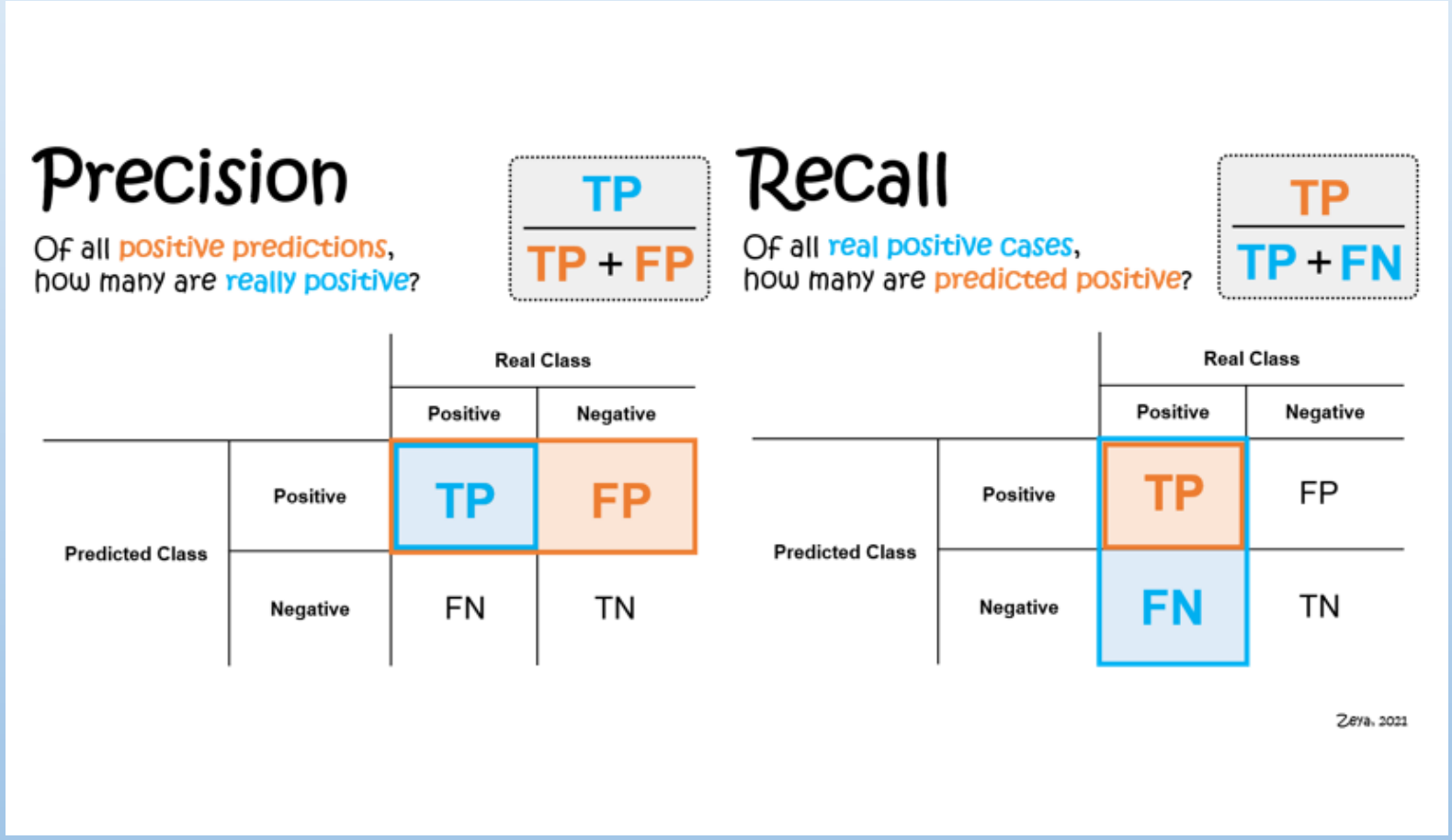
04. 딥러닝 - Case 1 딥러닝 모델 평가

Loss : 0.5246

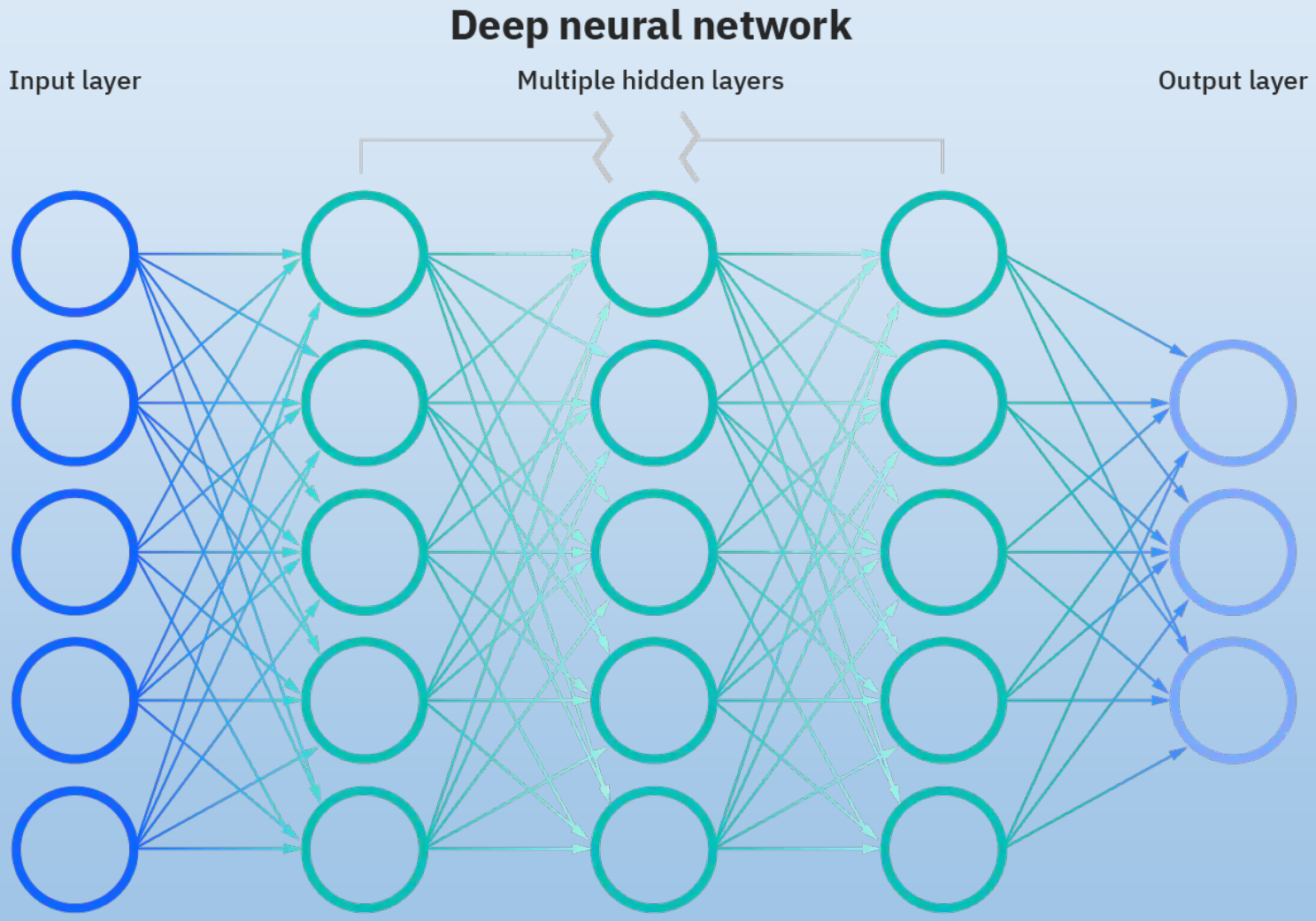
Accuracy : 0.7407

Recall : 0.7954

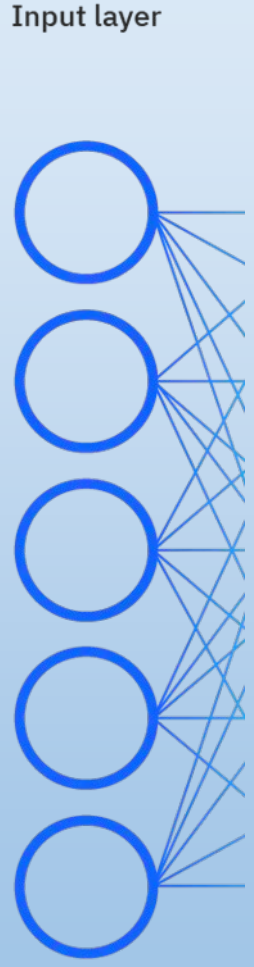
Precision : 0.7169



04. 딥러닝 - 인공신경망 모델



04. 딥러닝 - Layer 1. Input layer (Dense)



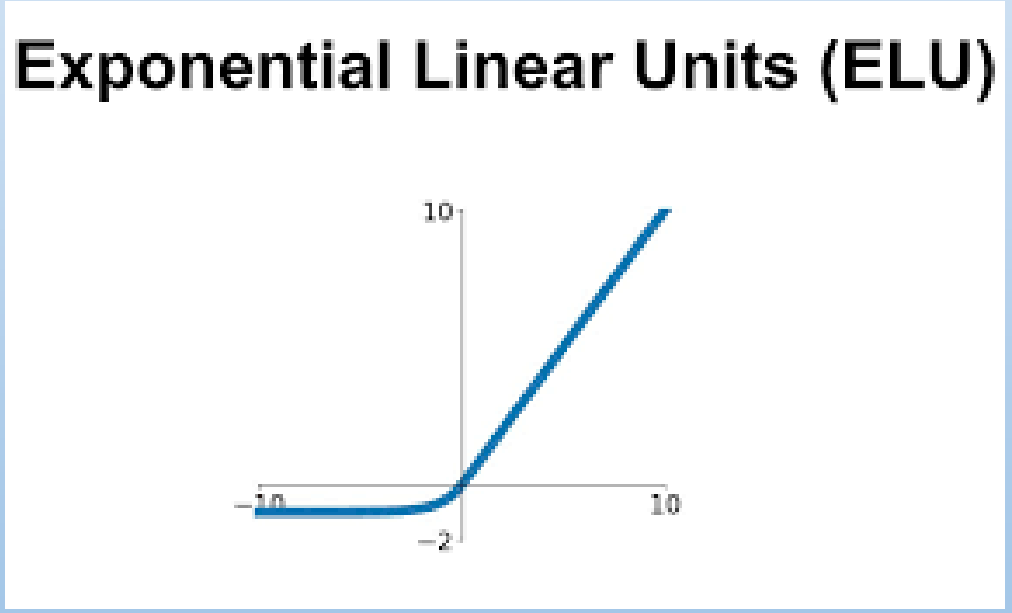
Input Dimension = 33

Units = 256

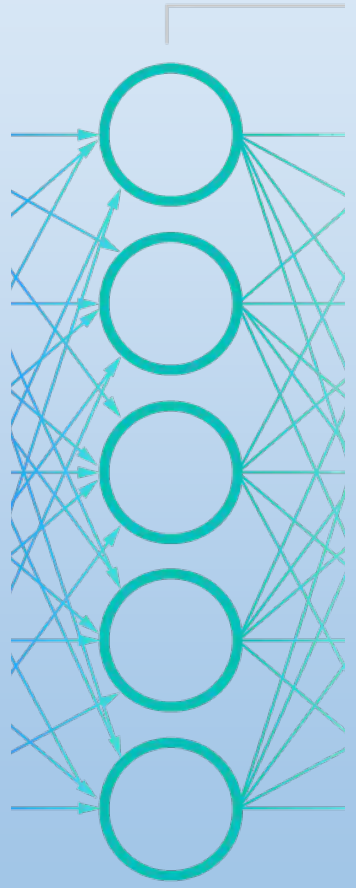
Kernel Initializer = He Normal

Batch Normalization

Activation = ELU



04. 딥러닝 - Layer 2. Hidden layer A (Dense)

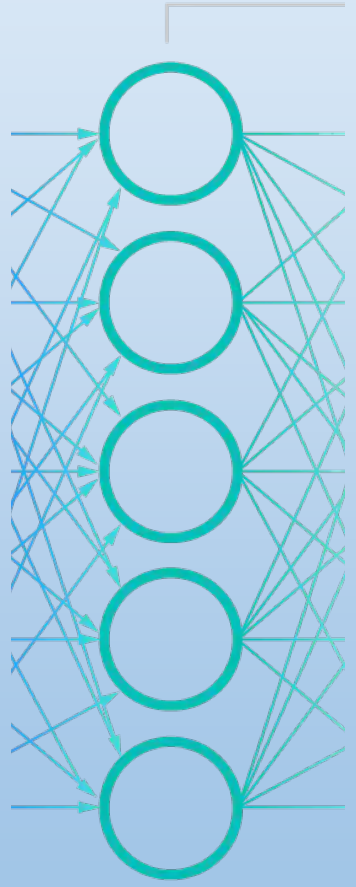


Units = 512

Kernel Initializer = He Normal

Activation = ELU

04. 딥러닝 - Layer 3. Hidden layer B (Dense)

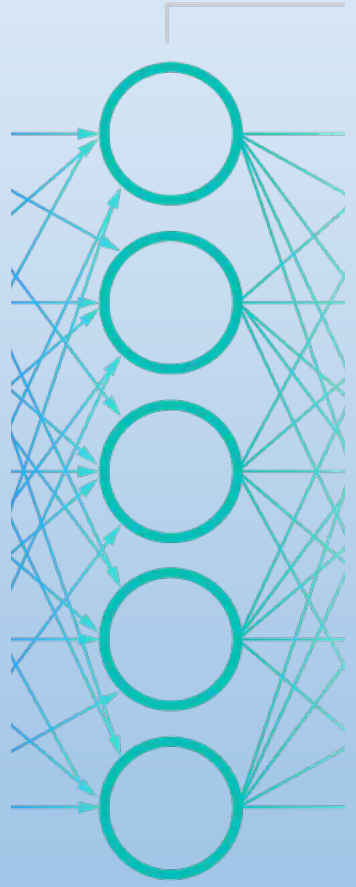


Units = 512

Kernel Initializer = He Normal

Activation = ELU

04. 딥러닝 - Layer 4. Hidden layer C (Dense)



Units = 256

Kernel Initializer = He Normal

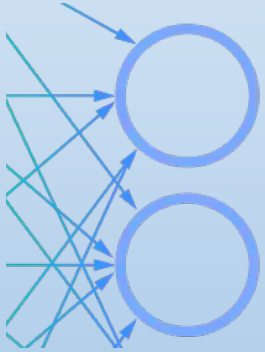
Activation = ELU

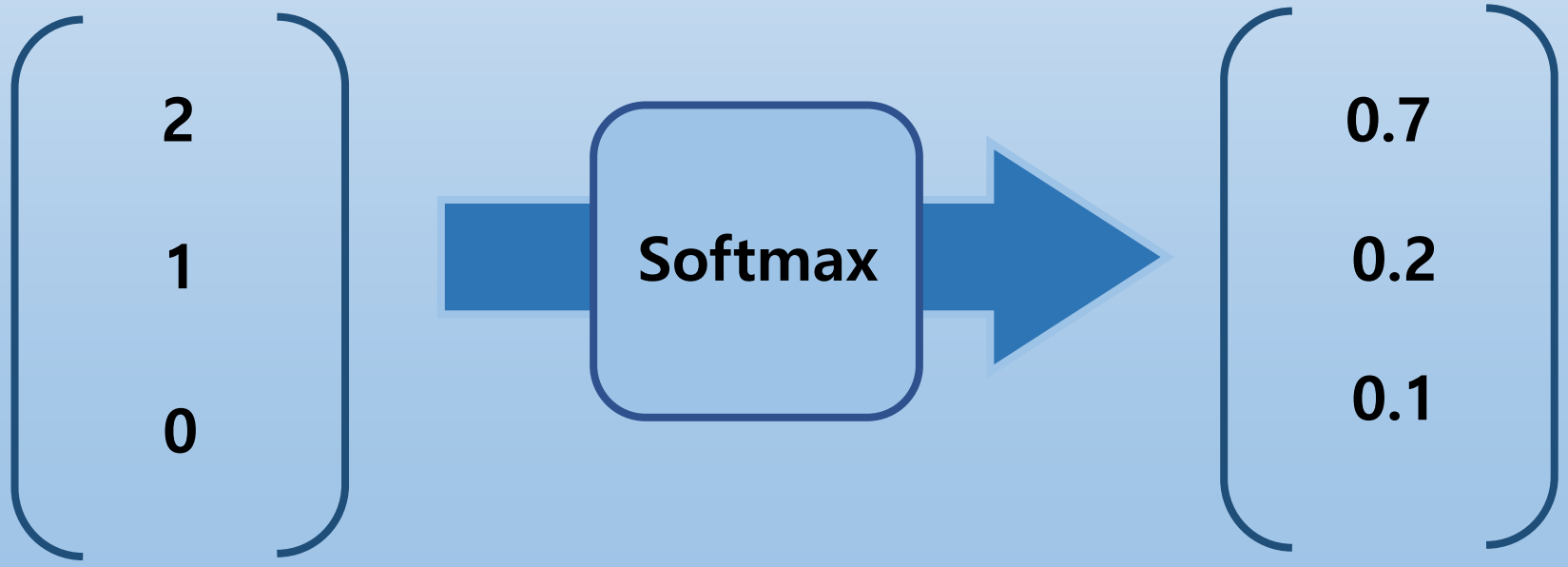
Drop Out = 0.5

04. 딥러닝 - Layer 5. Output layer (Dense)

Units = 2

Activation = Softmax



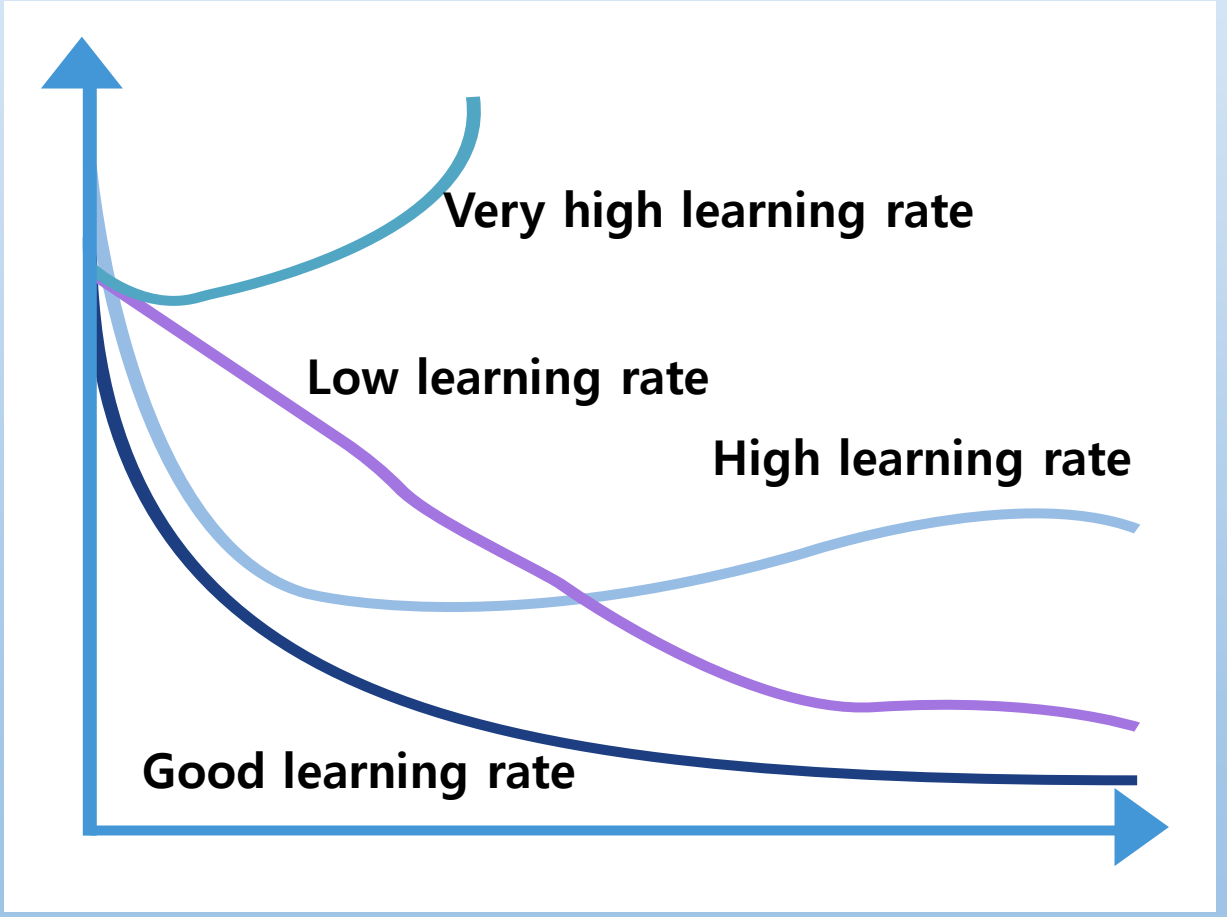


04. 딥러닝 - Compile

Optimizer = Adam (Learning Rate = 0.01)

Loss = Categorical Crossentropy

Metrics 1 = Categorical_accuracy



04. 딥러닝 - Summary

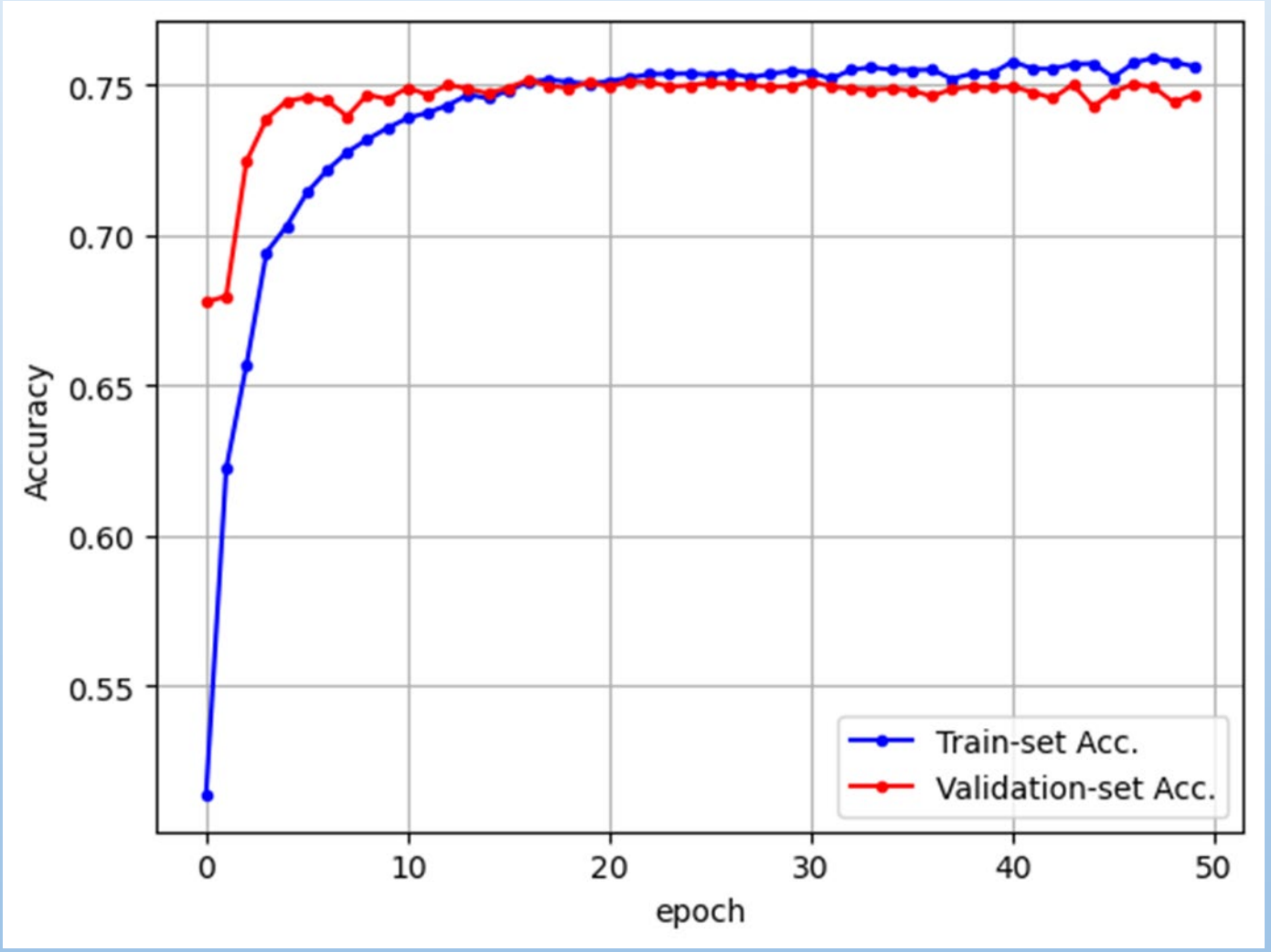
```
Model: "sequential"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
dense (Dense)                (None, 256)               8704
_________________________________________________________________
batch_normalization (BatchNo (None, 256)               1024
_________________________________________________________________
activation (Activation)      (None, 256)               0
_________________________________________________________________
dense_1 (Dense)              (None, 512)               131584
_________________________________________________________________
activation_1 (Activation)    (None, 512)               0
_________________________________________________________________
dense_2 (Dense)              (None, 512)               262656
_________________________________________________________________
activation_2 (Activation)    (None, 512)               0
_________________________________________________________________
dense_3 (Dense)              (None, 256)               131328
_________________________________________________________________
activation_3 (Activation)    (None, 256)               0
_________________________________________________________________
dropout (Dropout)            (None, 256)               0
_________________________________________________________________
dense_4 (Dense)              (None, 2)                 514
=================================================================
Total params: 535,810
Trainable params: 535,298
Non-trainable params: 512
```

04. 딥러닝 - 모델 Fit

Batch Size = 5000

Epochs = 50

Validation Split = 0.3



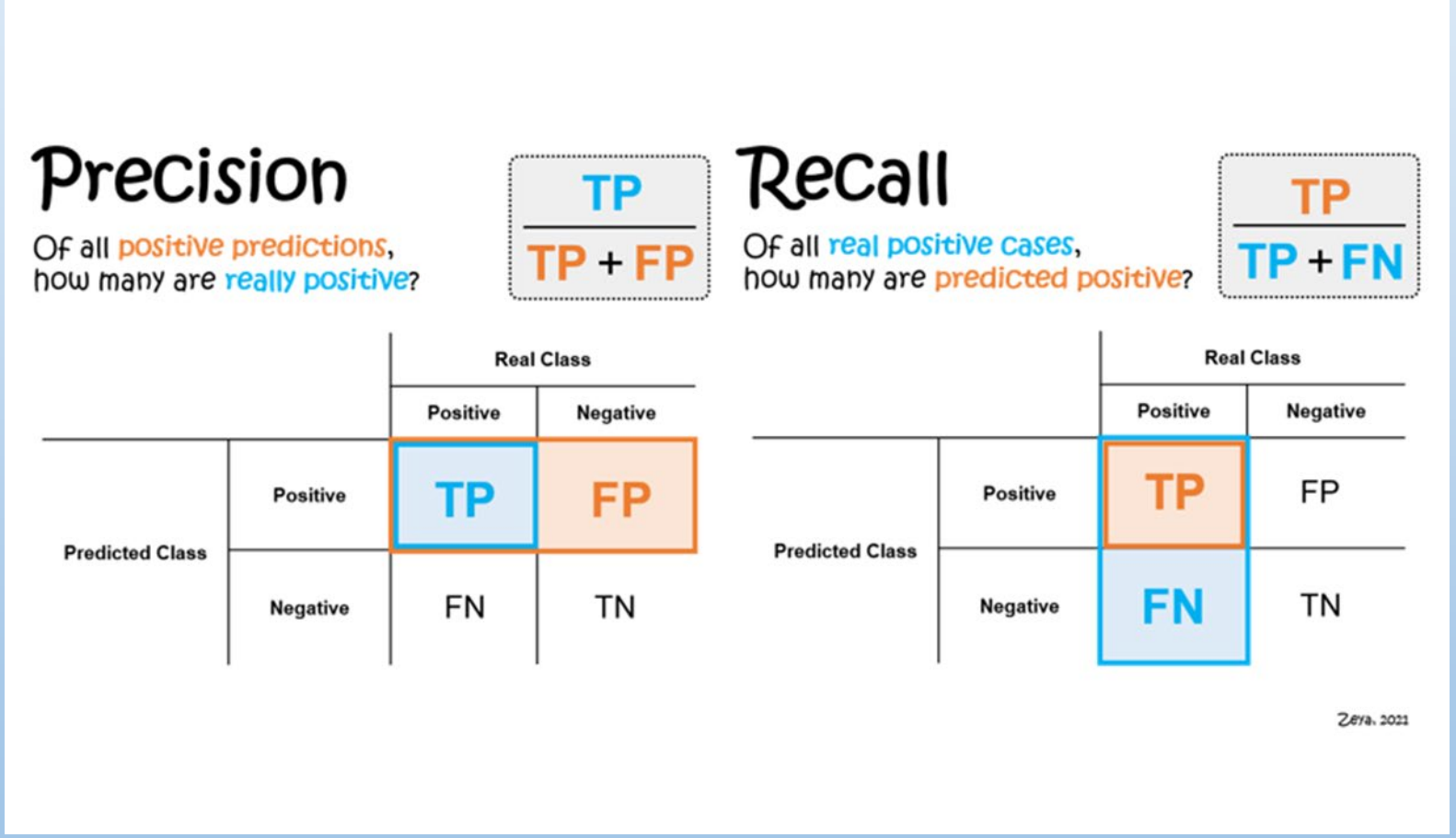
04. 딥러닝 - Case 2 딥러닝 모델 평가

Loss : 0.5025

Accuracy : 0.7533

Recall : 0.8008

Precision : 0.7268



04. 딥러닝 – Keras Tuner

Top-1

```
Trial 09 summary
Hyperparameters:
num_layers: 1
units_0: 256
activation_0: elu
learning_rate: 0.01
units_1: 512
activation_1: relu
units_2: 448
activation_2: elu
Score: 0.7542259097099304
```

Top-2

```
Trial 05 summary
Hyperparameters:
num_layers: 1
units_0: 96
activation_0: relu
learning_rate: 0.0001
units_1: 416
activation_1: relu
units_2: 480
activation_2: elu
Score: 0.754155158996582
```

Top-3

```
Trial 06 summary
Hyperparameters:
num_layers: 3
units_0: 320
activation_0: elu
learning_rate: 0.001
units_1: 96
activation_1: relu
units_2: 448
activation_2: relu
Score: 0.7538015246391296
```

04. 딥러닝 – Keras Tuner

```
Model: "sequential"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 dense (Dense)               (None, 256)               10752

 dense_1 (Dense)             (None, 256)               65792

 dense_2 (Dense)             (None, 2)                 514

=================================================================
Total params: 77,058
Trainable params: 77,058
Non-trainable params: 0
_________________________________________________________________
442/442 [==============================] - 1s 2ms/step - loss: 0.5131 - accuracy: 0.7542
```

```
results
```

```
[0.513058066368103, 0.7542259097099304]
```

05. 결론

05. 결론

1. 머신러닝 중 성능이 가장 높은 모델

- Accuracy : 0.7570, Recall : 0.7987, Prec : 0.7372, F1 : 0.7667 지표가 가장 좋게 나왔으므로 Pycaret 모델이 가장 좋은 모델이라고 판단.

2. 딥러닝 중 성능이 가장 높은 모델

- Loss : 0.5025, Accuracy : 0.7533, Recall : 0.8008, Precision : 0.7268 지표가 가장 좋게 나왔으므로 Case 2(Pipeline적용) 모델이 가장 좋은 모델이라고 판단.

06. 한계 및 발전방향

06. 한계 및 발전방향

1. 한계점

- 데이터가 대부분 이산형으로 정제되어 있어서 EDA에 제약이 많았다.
- 나이 컬럼이 처음에 전처리가 안 되어 있었으면 모델 정확도가 더 높지는 않았을까 판단이 된다.

2. 발전방향

- 전처리 된 데이터가 아닌 것을 선택해서 다음 프로젝트에 전처리를 해 보는 방향으로 잡고, 다양한 EDA로 표현하기로 결정.

Thank you