KAGGLE COMPETITION

Multi-Class Prediction of Obesity Risk

CONTENT

KAGGLE COMPETITION

1. Introduction

1. INTRODUCTION

Overview of Competition

Multi-Class Prediction of Obesity Risk
Playground Series - Season 4, Episode 2

Competition Timeline	February 1st, 2024 ~ February 29th, 2024(11:59 PM UTC)
Prizes	1st - Choice of Kaggle merchandise 2nd - Choice of Kaggle merchandise 3rd - Choice of Kaggle merchandise
Duration Participation	4 days (February 26th, 2024 ~ February 29th, 2024)
Participants	1 person
Kaggle Notebook	[🌱Beginner Friendly] Obesity Risk Prediction(92%)

1. INTRODUCTION

Development Environment

Language

python™

Library

NumPy

OPTUNA

pandas

LightGBM

scikit learn

dmlc XGBoost

1. INTRODUCTION

I. What is LightGBM?

LightGBM(Light Gradient Boosting Machine) is a framework for tree-based learning algorithms using the gradient boosting technique.

1. INTRODUCTION

II. The Features of LightGBM

a. Algorithms of LightGBM

- Leaf-Wise Tree Growth algorithm
- Histogram-Based Splitting

b. Sampling methods of LightGBM

- Gradient-based One-Side Sampling(GOSS)
- Exclusive Feature Bundling(EFB)

c. Training with Category Type Variables

- Facilitating the understanding of dataset characteristics with categorical variables.

1. INTRODUCTION

III. Hyperparameter Tuning

```
param = {"objective": "multiclass",
    "metric": "multi_logloss",
    "verbosity": -1,
    "boosting_type": "gbdt",
    "random_state": 42,
    "num_class": 7,
    'learning_rate': 0.030962211546832760,
    'n_estimators': 500,
    'lambda_l1': 0.009667446568254372,
    'lambda_l2': 0.040186414373018000,
    'max_depth': 10,
    'colsample_bytree': 0.40977129346872643,
    'subsample': 0.9535797422450176,
    'min_child_samples': 26}
```

Hyperparameter tuning for optimal performance!

1. INTRODUCTION

Goal

The goal of this competition is to use various factors to predict obesity risk in individuals, which is related to cardiovascular disease.

Process

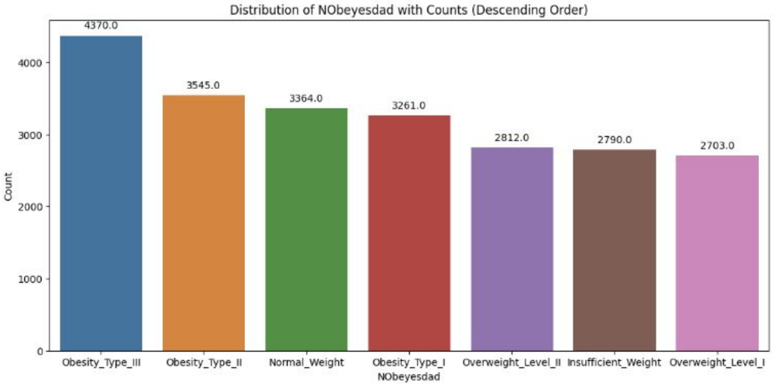
EDA → **Statistical Analysis** → **Data Processing** → **Modeling** → **Prediction**

2. EDA

2. EDA

EDA X
=> Pattern does not exist.

Data Description

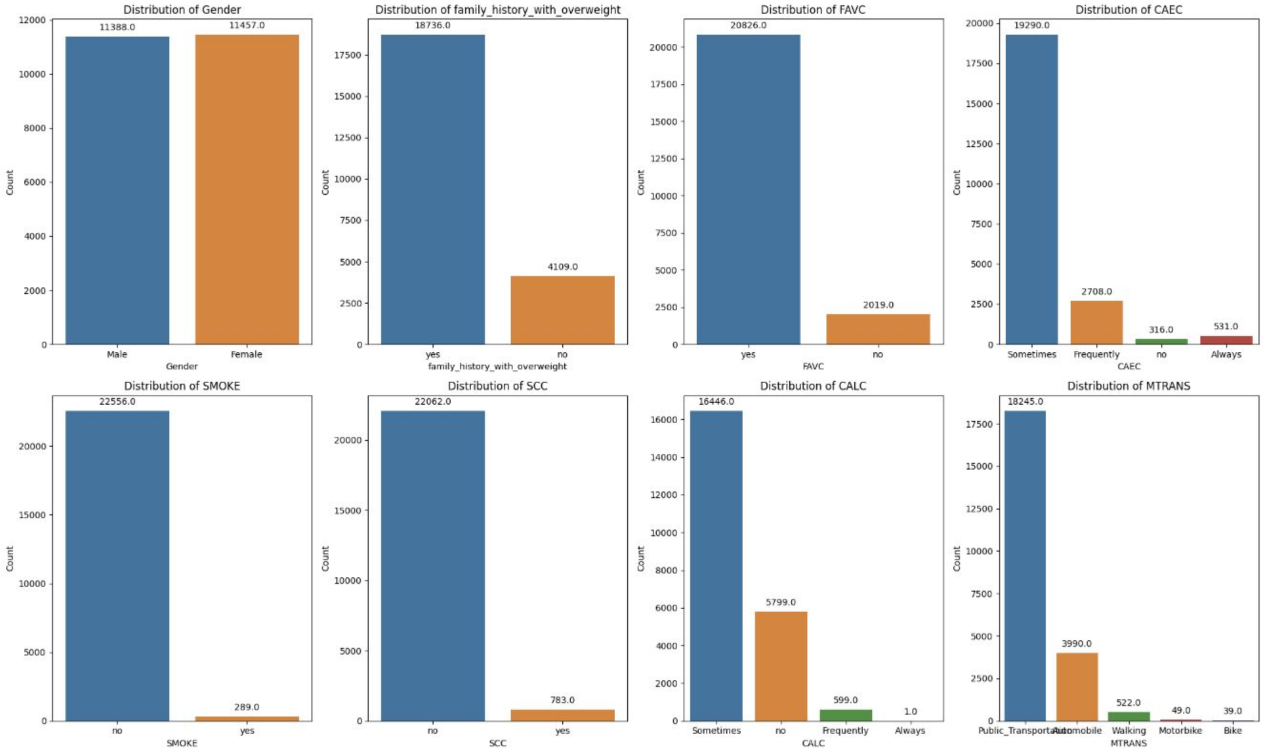


Column	Full Form	Description
'id'	id	Unique for each person(row)
'Gender'	Gender	person's Gender
'Age'	Age	Dtype is float. Age is between 14 years to 61 years
'Height'	Height	Height is in meter it's between 1.45m to 1.98m
'Weight'	Weight	Weight is between 39 to 165. I think it's in KG.
'family_history_with_overweight'	family history with overweight	yes or no question
'FAVC'	Frequent consumption of high calorie food	it's yes or no question. i think question they asked is do you consume high calorie food
'FCVC'	Frequency of consumption of vegetables	Similar to FAVC. this is also `yes or no` question
'NCP'	Number of main meals	dtype is float, NCP is between 1 & 4. I think it should be 1,2,3,4 but our data is synthetic so it's taking float values
'CAEC'	Consumption of food between meals	takes 4 values `Sometimes`, `Frequently`, `no` & `Always`
'SMOKE'	Smoke	yes or no question. i think the question is "Do you smoke?"
'CH2O'	Consumption of water daily	CH2O takes values between 1 & 3. again it's given as float may be because of synthetic data. it's values should be 1,2 or 3
'SCC'	Calories consumption monitoring	yes or no question
'FAF'	Physical activity frequency	FAF is between 0 to 3, 0 means no physical activity and 3 means high workout. and again, in our data it's given as float
'TUE'	Time using technology devices	TUE is between 0 to 2. I think question will be "How long you have been using technology devices to track your health." in our data it's given as float
'CALC'	Consumption of alcohol	Takes 3 values: `Sometimes`, `no`, `Frequently`
'MTRANS'	Transportation used	MTRANS takes 5 values `Public_Transportation`, `Automobile`, `Walking`, `Motorbike`, & `Bike`
'NObeyesdad'	TARGET	This is our target, takes 7 values, and in this comp. we have to give the class name (Not the Probability, which is the case in most comp.)

2. EDA

Categorical Features

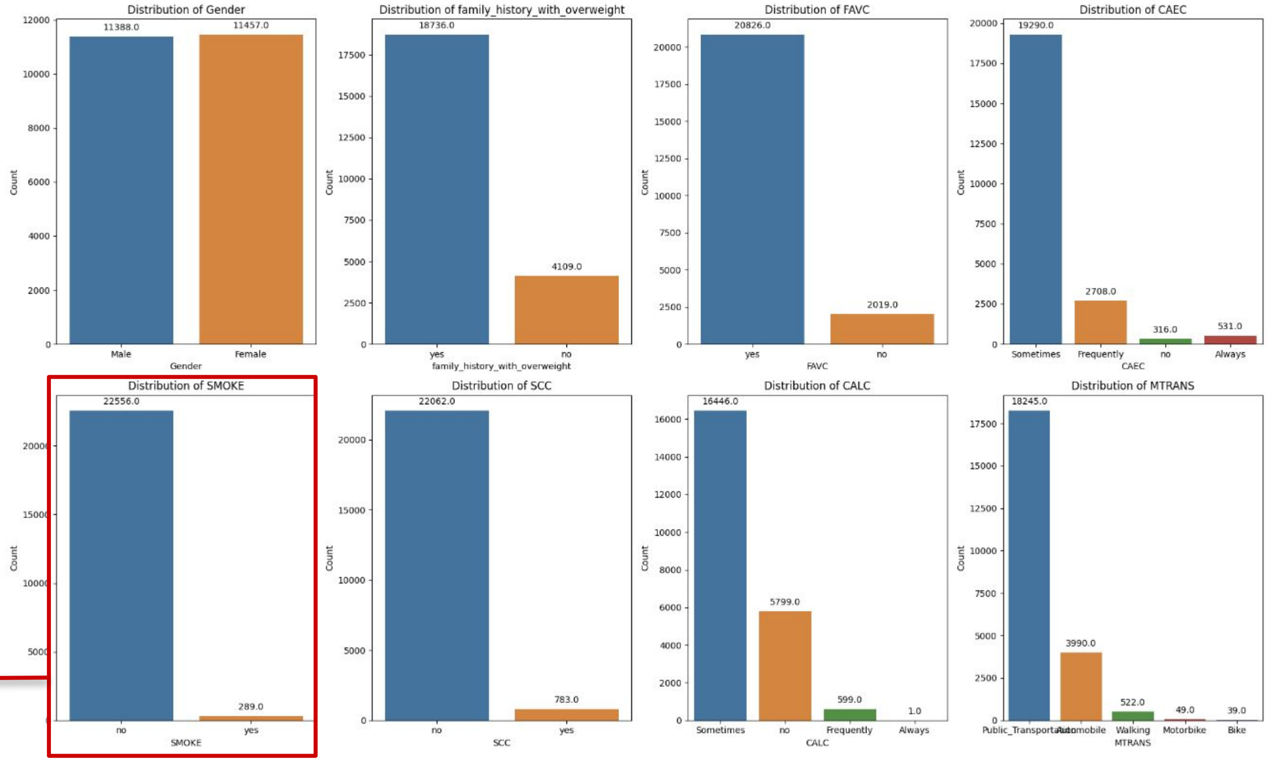
	Column
0	Gender
1	family_history_with_overweight
2	FAVC
3	CAEC
4	SMOKE
5	SCC
6	CALC
7	MTRANS



2. EDA

Categorical Features

	Column
0	Gender
1	family_history_with_overweight
2	FAVC
3	CAEC
4	SMOKE
5	SCC
6	CALC
7	MTRANS



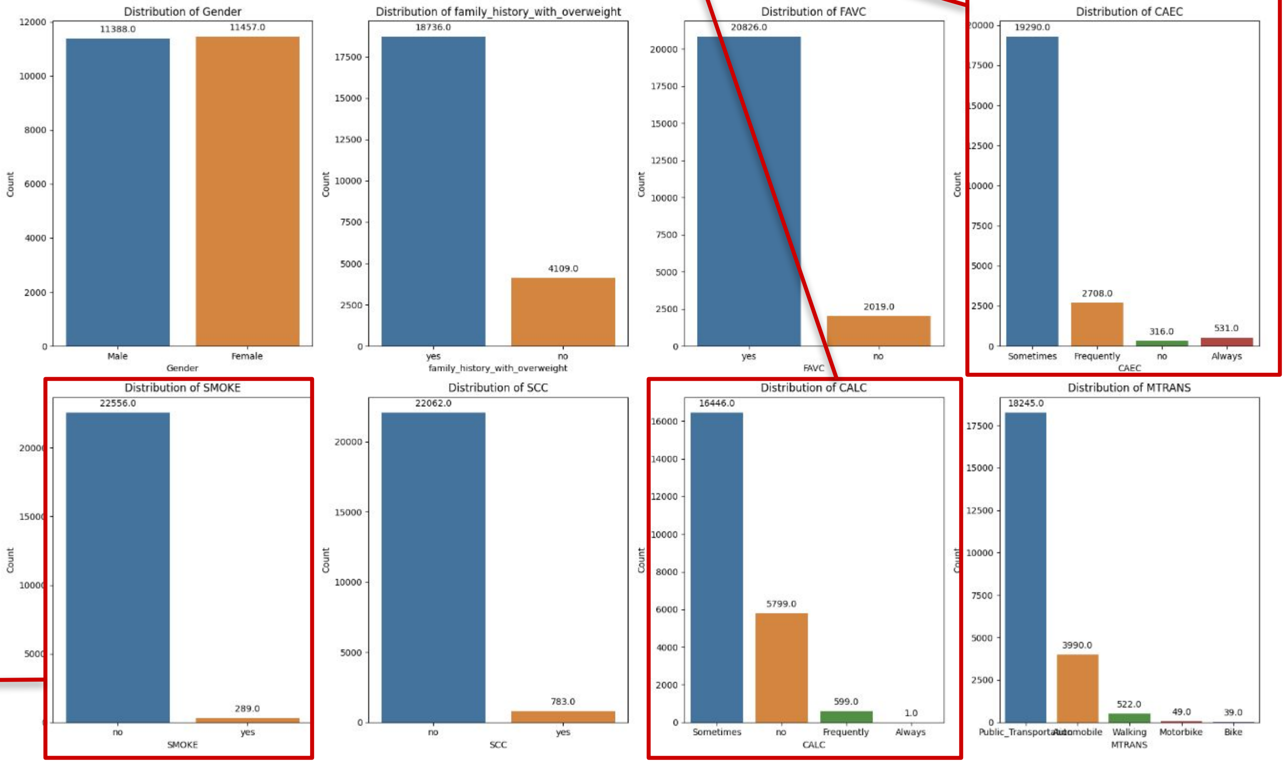
Big difference in the ratio of no and yes. Is it trustworthy?

2. EDA

Categorical Features

Using the characteristics of numbers by changing the frequency of 'CAEC', 'CALC' to numbers?

	Column
0	Gender
1	family_history_with_overweight
2	FAVC
3	CAEC
4	SMOKE
5	SCC
6	CALC
7	MTRANS

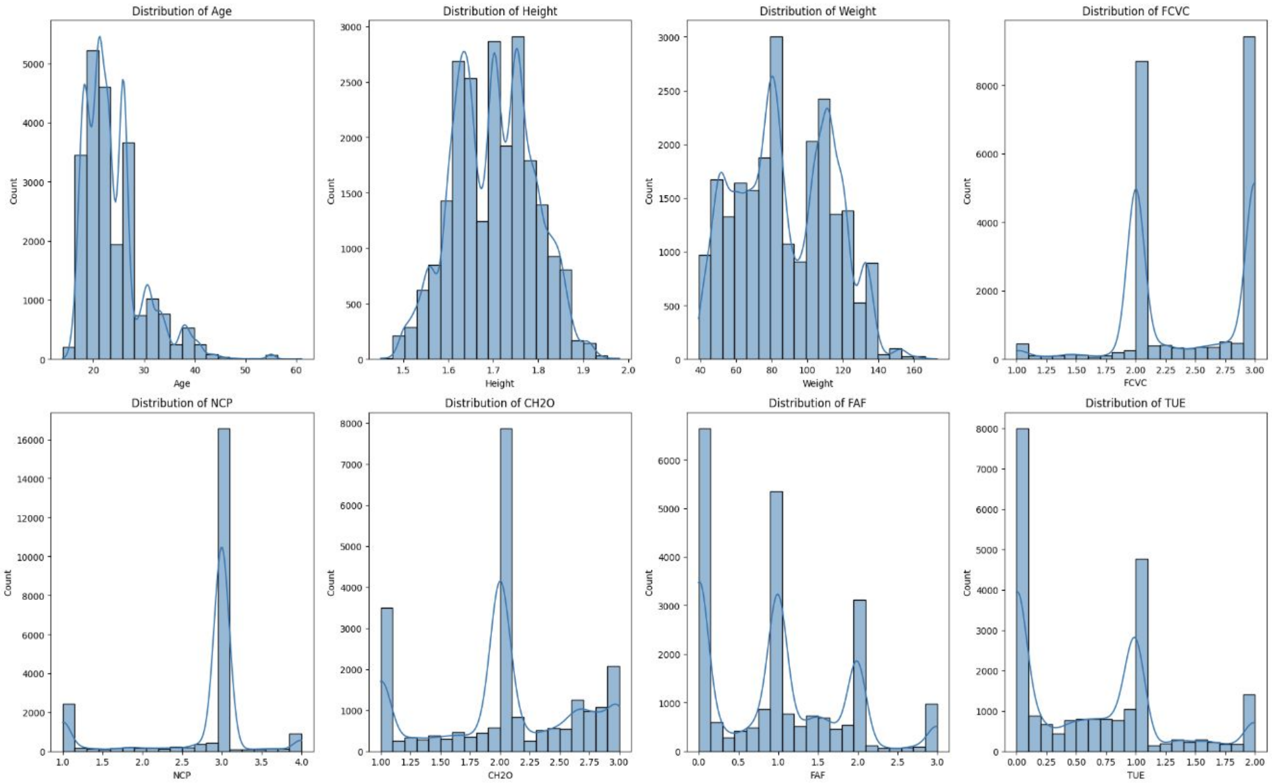


Big difference in the ratio of no and yes. Is it trustworthy?

2. EDA

Numerical Features

	Column
0	Age
1	Height
2	Weight
3	FCVC
4	NCP
5	CH2O
6	FAF
7	TUE

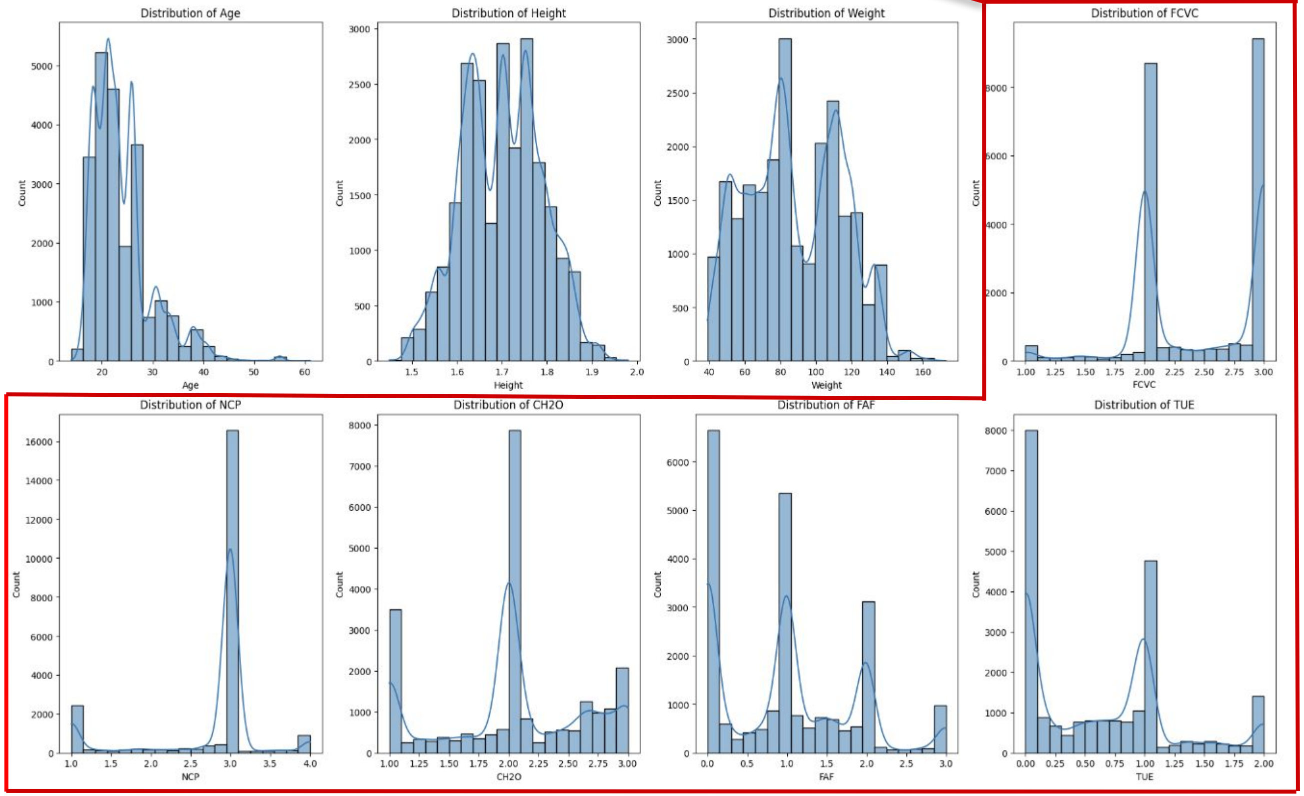


2. EDA

'FCVC', 'NCP', 'CH2O', 'FAF', 'TUE'
look similar to categorical distribution.

Numerical Features

	Column
0	Age
1	Height
2	Weight
3	FCVC
4	NCP
5	CH2O
6	FAF
7	TUE



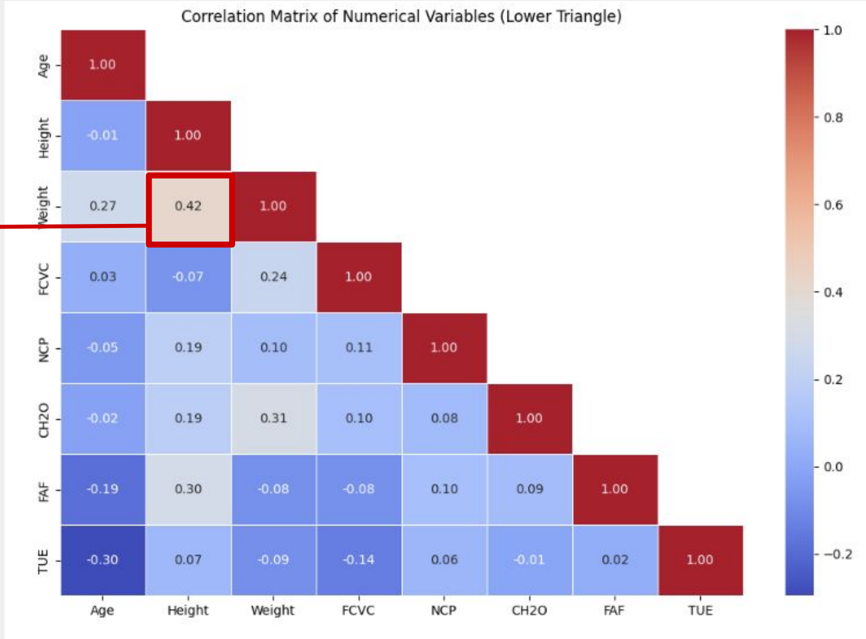
2. EDA

Correlation Matrix

The correlation between height and weight variables is higher than that of other variables.

It is worth creating a BMI variable using height and weight variables.

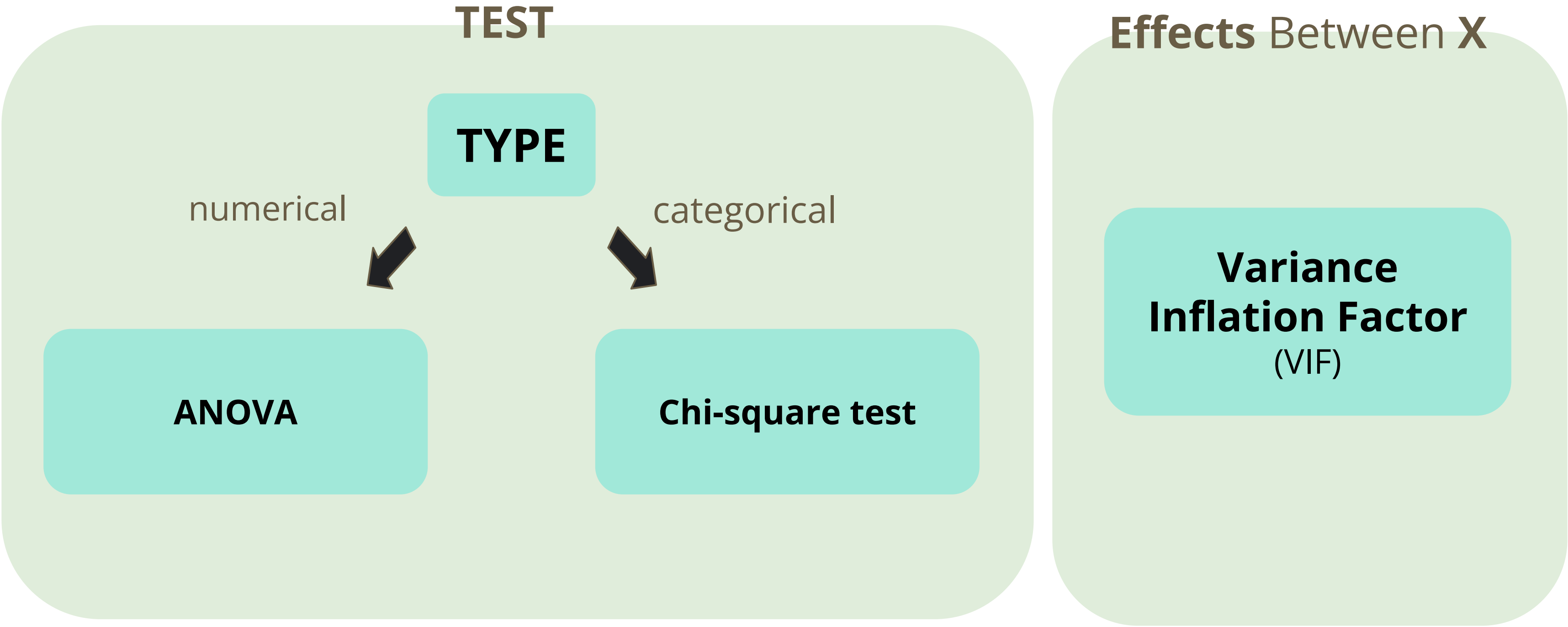
=> *BMI = Weight / Height ** 2*



3. Statistical Analysis

1) ANOVA
2) Chi-square test
3) Variance Inflation Factor

3. Statistical Analysis



3. Statistical Analysis

TEST

TYPE

numerical → **ANOVA**

categorical → **Chi-square test**

Effects Between **X**

Variance Inflation Factor
(VIF)

3. Statistical Analysis

TEST

TYPE

numerical → **ANOVA**

categorical → **Chi-square test**

Effects Between **X**

Variance Inflation Factor
(VIF)

3. Statistical Analysis

Analysis of Variance(ANOVA) - User-Defined Functions_Numerical Features

Homogeneity of Variance
(levene-test)

X

Welch's ANOVA

post hoc-test

Interpreting & Visualization

O

ANOVA

post hoc-test

Interpreting & Visualization

3. Statistical Analysis

OUTPUT

```
levene_anova('NCP')
```

homogeneity

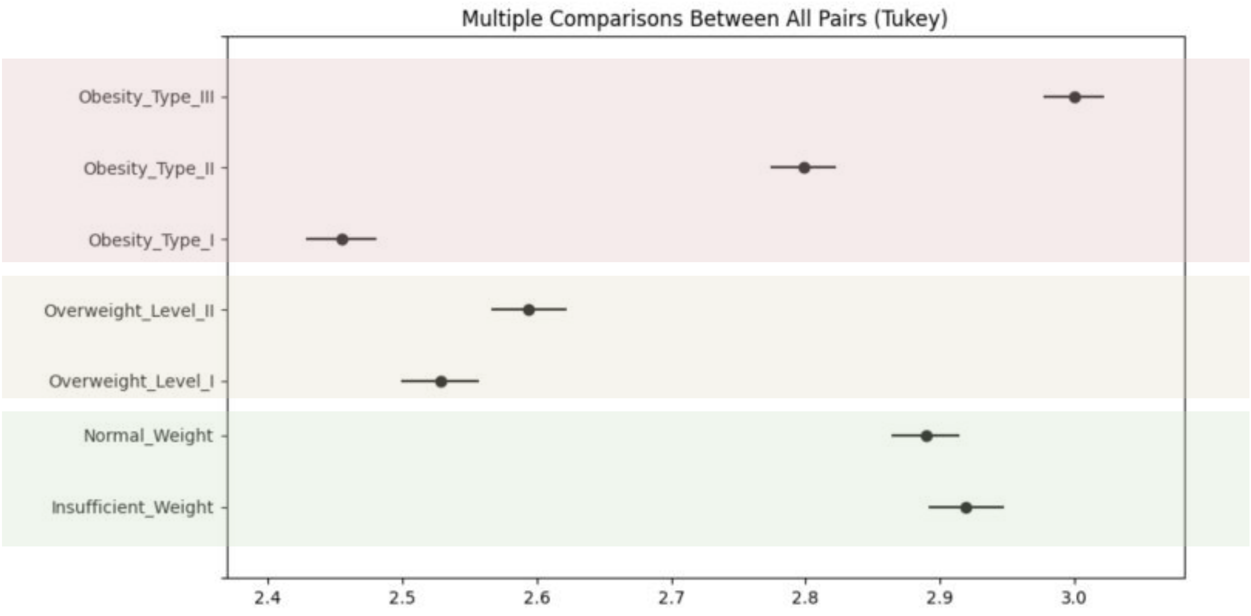
ANOVA

post hoc

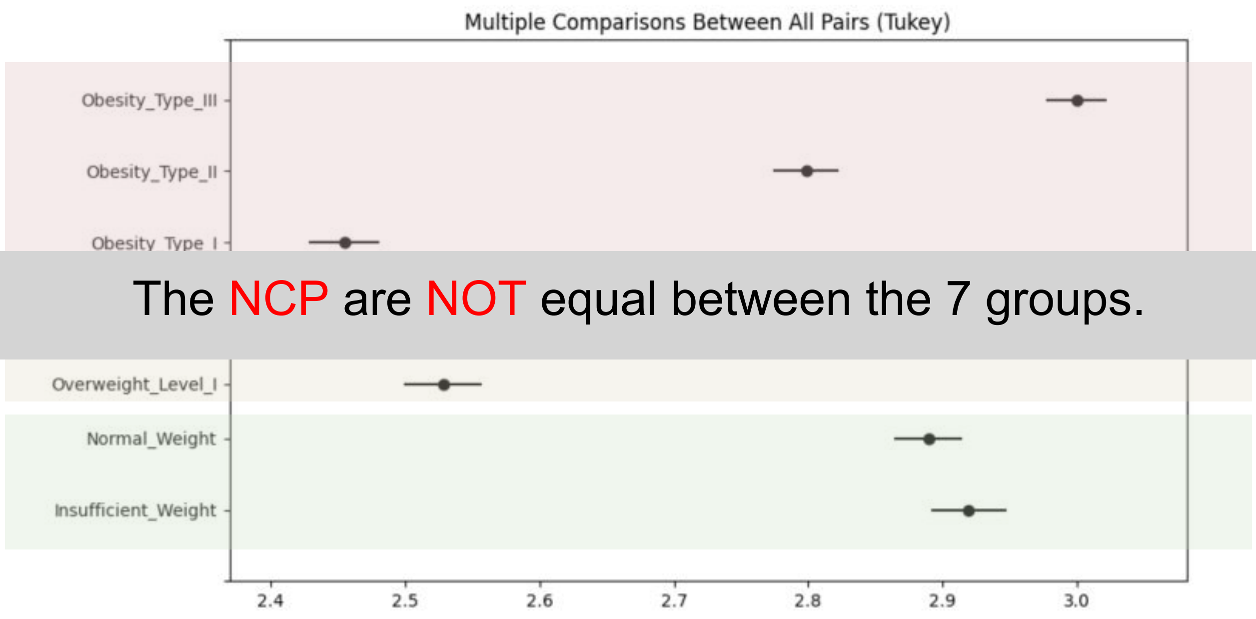
```
MESSAGE: At least one of the variances among the groups is different.
MESSAGE: Reject the null hypothesis that the NCP are equal between the 7 groups
             Multiple Comparison of Means - Tukey HSD, FWER=0.05
==============================================================================
       group1              group2        meandiff p-adj   lower   upper  reject
------------------------------------------------------------------------------
Insufficient_Weight       Normal_Weight    -0.03 0.6487 -0.0836  0.0236  False
Insufficient_Weight      Obesity_Type_I   -0.465    0.0 -0.5193 -0.4107   True
Insufficient_Weight     Obesity_Type_II  -0.1211    0.0 -0.1741 -0.0682   True
Insufficient_Weight    Obesity_Type_III   0.0804 0.0001  0.0298   0.131   True
Insufficient_Weight  Overweight_Level_I  -0.3914    0.0 -0.4481 -0.3346   True
Insufficient_Weight Overweight_Level_II  -0.3254    0.0 -0.3816 -0.2693   True
```

3. Statistical Analysis

OUTPUT

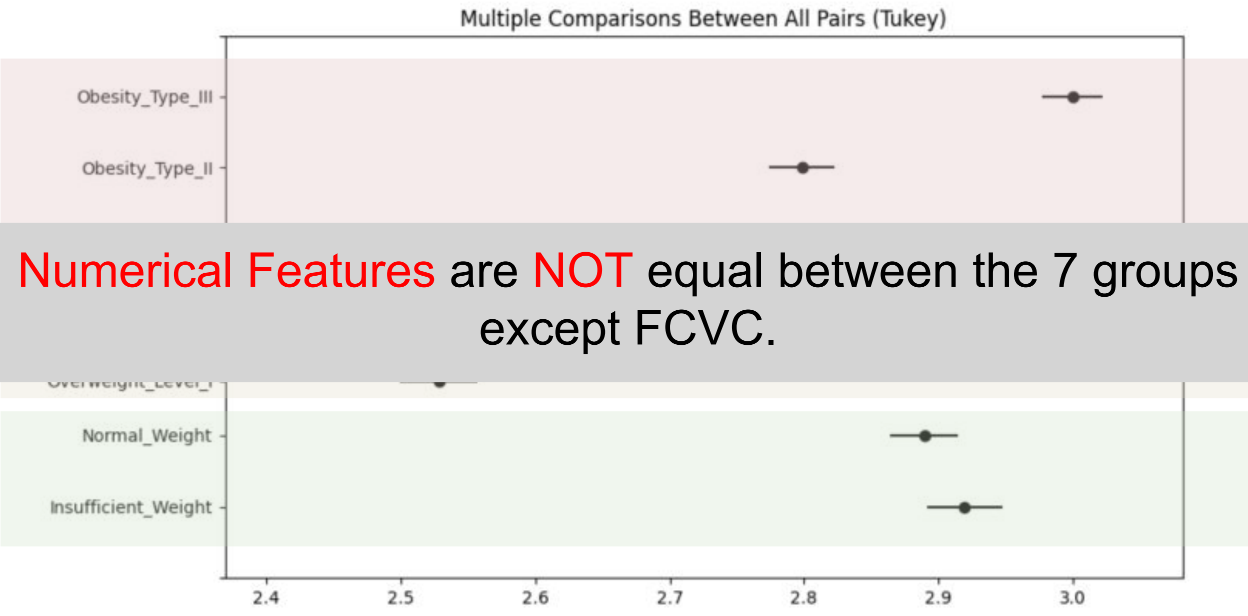


3. Statistical Analysis



The NCP are NOT equal between the 7 groups.

3. Statistical Analysis



Numerical Features are NOT equal between the 7 groups except FCVC.

3. Statistical Analysis

TEST

TYPE

numerical → **ANOVA**

categorical → **Chi-square test**

Effects Between **X**

Variance Inflation Factor
(VIF)

3. Statistical Analysis

Independence test(Chi-square test) - UDF_Categorical Features

Cross-tabulation

↓

Interpreting Results

3. Statistical Analysis

Independence test(Chi-square test) - UDF_Categorical Features

Cross-tabulation

Family history with overweight and the obesity risk is dependent.

Interpreting Results

3. Statistical Analysis

Independence test(Chi-square test) - UDF_Categorical Features

Cross-tabulation

Each Categorical Features on the obesity risk
is dependent.

Interpreting Results

3. Statistical Analysis

TEST

TYPE

numerical → **ANOVA**

categorical → **Chi-square test**

Effects Between **X**

Variance Inflation Factor
(VIF)

3. Statistical Analysis

Variance Inflation Factor (VIF)- UDF

Calculating VIF

↓

Showing Results

3. Statistical Analysis

Variance Inflation Factor (VIF)- UDF

	feature	VIF
0	id	3.984522
1	Age	21.488351
2	Height	75.103311
3	Weight	16.335162
4	FCVC	22.252531
5	NCP	17.411882
6	CH2O	13.719299
7	FAF	2.700994
8	TUE	2.320133

3. Statistical Analysis

High VIF

Problems

1. Interpretation of the model
2. Model stability
3. Statistical significance

Solution

1. **Variable Selection**
2. Dimensionality Reduction
3. Variable Transformation
4. **Tree-based Model**

4. Machine Learning

4. Machine learning

	Model	Data processing	Data preprocessing	Hyperparameter	accuracy	accuracy_rate	recall	precision	F1 Score	F1 Score_rate
case1	LGBMClassifier	X			90.6069%				89.5777%	
case2	Random Forest	X			89.5713%	▼1.0356%			88.4281%	▼1.1496%
case3	LGBMClassifier	O	standard Scaler OneHotEncoder	n_iter = 3, cv=2,	90.6792%	▲0.0723%	89.6663%	89.6914%	89.6618%	▲0.0841%
case4	LGBMClassifier	O	Robust Scaler OneHotEncoder	n_iter = 3, cv=2,	90.6310%	▲0.0241%	89.6101%	89.6371%	89.6026%	▲0.0841%
case5	XGBoost	O	Robust Scaler OneHotEncoder		90.2697%	▼0.3372%	89.2019%	89.2315%	89.1924%	▼0.3853%
case6	Random Forest	O	Robust Scaler OneHotEncoder		90.2697%	▼0.3372%	87.0129%	87.0738%	87.0110%	▼2.5667%
case7	LGBMClassifier	O	standard Scaler OneHotEncoder	"clf__num_leaves": cv=2,	90.5347%	▼0.0723%	89.4956%	89.5202%	89.4910%	▼0.0867%
case8	LGBMClassifier	O	standard Scaler OneHotEncoder	n_iter = 5, cv=2,	90.6792%	▲0.0723%	89.6663%	89.6914%	89.6618%	▲0.0841%
case9	LGBMClassifier	O	standard Scaler OneHotEncoder	n_iter = 5, cv=3,	90.6792%	▲0.0723%	89.6663%	89.6914%	89.6618%	▲0.0841%
case10	LGBMClassifier	O	**RobustScaler** OneHotEncoder	n_iter = 5, **cv=4,**	90.6310%	▲0.0241%	89.6101%	89.6371%	89.6026%	▲0.0249%
case11	LGBMClassifier	O	RobustScaler OneHotEncoder	n_iter = 3, cv=5, random_state = 30	90.5347%	▼0.0723%	89.5015%	89.5539%	89.5072%	▼0.0705%
case12	LGBMClassifier	O	RobustScaler OneHotEncoder	n_iter = 3, cv=2, num_class = 7	90.6310%	▲0.0241%	89.6101%	89.6371%	89.6026%	▲0.0249%
case13	LGBMClassifier	O	standard Scaler OneHotEncoder	n_iter = 3, cv=2, num_class = 7	90.6792%	▲0.0723%	89.6663%	89.6914%	89.6618%	▲0.0841%

→ LGBMClassifier models perform better than
Random Forest and XGBoost models under the same conditions

4. Machine learning

DATA

- 'FCVC', 'NCP', 'CH2O', 'FAF', 'TUE' categorization
- StandardScaler, OneHotEncoder

MODEL

- Using Pipeline, LGBMClassifier, optuna(n_trials=100)

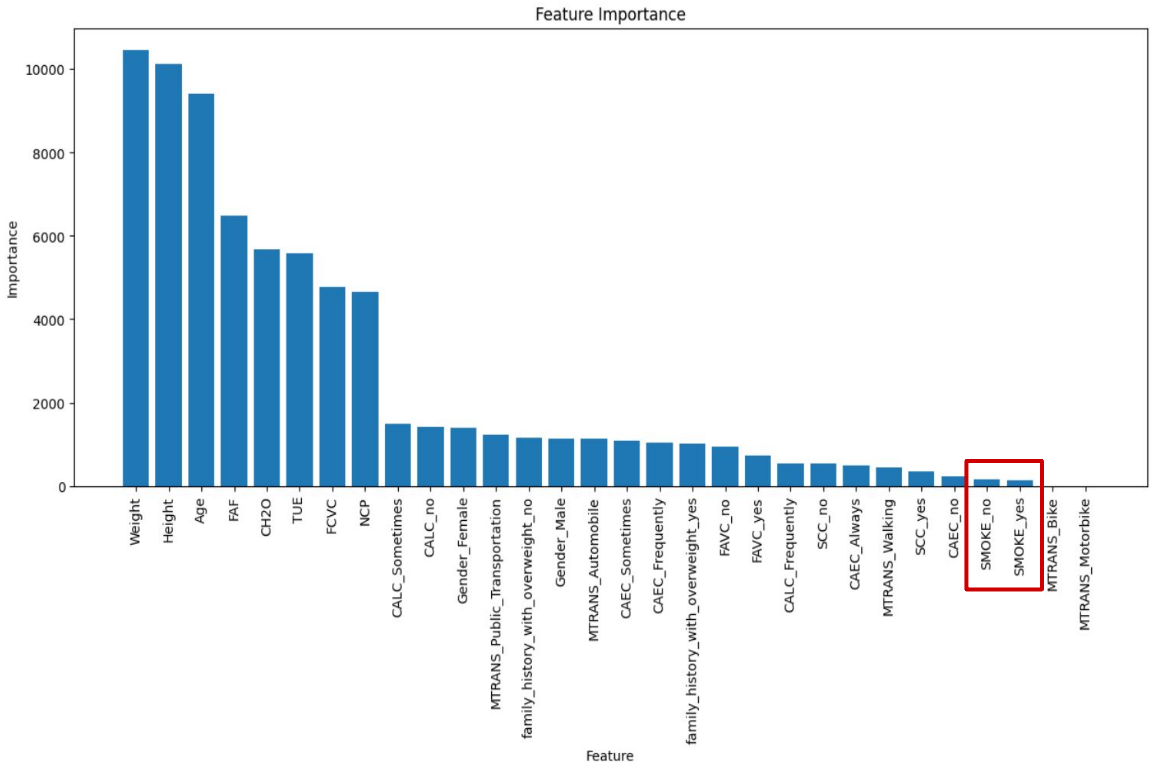
+ Find the **optimal params** : 0.90281

+ y-value **label encoding** : 0.90426

+ label encoding & **BMI columns create** : 0.8992

```
train['BMI'] = train['Weight'] / (train['Height'] ** 2)
test['BMI'] = test['Weight'] / (test['Height'] ** 2)
```

4. Machine learning



- Delete SMOKE variables with low feature importance

4. Machine learning

DATA

- StandardScaler, OneHotEncoder, LabelEncoder(y_value)

MODEL

- Using Pipeline, LGBMClassifier, optuna(n_trials=50)

+ **Delete SMOKE** : 0.90715

+ **No preprocessing** : 0.91112

+ **Reduce run time**(12.4s[50.1s→37.7s]) : 0.91112

+ **Modifying params**(n_trials=100) : 0.90932

4. Machine learning

	Gender	Age	Height	Weight
0	Male	24.443011	1.699998	81.669950
1	Female	18.000000	1.560000	57.000000
2	Female	18.000000	1.711460	50.165754
3	Female	20.952737	1.710730	131.274851
4	Male	31.641081	1.914186	93.798055

	Gender	Age	Height	Weight
0	Male	20.0	1.699998	81.669950
1	Female	10.0	1.560000	57.000000
2	Female	10.0	1.711460	50.165754
3	Female	20.0	1.710730	131.274851
4	Male	30.0	1.914186	93.798055

DATA
Age grouping

DATA

- Age grouping
- StandardScaler, OneHotEncoder, LabelEncoder(y_value)

MODEL

- Using Pipeline, LGBMClassifier, optuna

\+ Find the optimal params : 0.90751

4. Machine learning

DATA

- 'CAEC', 'CALC' mapping Always, Frequently, Sometimes, no → (4, 3, 2, 1)
- StandardScaler, pandas_get_dummies, LabelEncoder(y_value)

MODEL

- Using LGBMClassifier, optuna(n_trial: 100)

\+ predict : 0.90751

DATA

- Create BMI & Delete SMOKE
- StandardScaler, LabelEncoder(object), LabelEncoder(y_value)

MODEL

- Using LGBMClassifier, optuna(n_trial: 120)

\+ predict_proba : 0.87391

4. Machine learning

DATA

- Add original_data_Use to increase the number of data
- StandardScaler, LabelEncoder (object, y_value)

MODEL

- Using LGBMClassifier, optuna(n_trials=100, Adjusting thresholds)

```
param = {"objective": "multiclass",
    "metric": "multi_logloss",
    "verbosity": -1,
    "boosting_type": "gbdt",
    "random_state": 42,
    "num_class": 7,
    'learning_rate': 0.030962211546832760,
    'n_estimators': 500,
    'lambda_l1': 0.009667446568254372,
    'lambda_l2': 0.040186414373018000,
    'max_depth': 10,
    'colsample_bytree': 0.40977129346872643,
    'subsample': 0.9535797422450176,
    'min_child_samples': 26}
```

```
threshold= {'threshold_0': 0.724201213234911, 'threshold_1': 0.6161299800571379, 'threshold_2': 0.
29138887902587174, 'threshold_3': 0.3145837593497076, 'threshold_4': 0.8469398340837189, 'threshol
d_5': 0.6800824438387787, 'threshold_6': 0.35886959729223455}
```

⇒ **SELECTED** : 0. 92196

4. Machine learning

	Model	Data processing	Data preprocessing	Hyperparameter	Public Score	Public Score_rate
case1	LGBMClassifier Pipeline	O_categorization	Standard Scaler OneHotEncoder	optuna	90.281%	**baseline score**
case2	LGBMClassifier Pipeline	O_categorization	Standard Scaler OneHotEncoder **LabelEncoder(y)**	optuna	90.426%	▲0.145%
case3	LGBMClassifier Pipeline	O_categorization, create BMI	Standard Scaler OneHotEncoder LabelEncoder(y)	optuna	89.920%	▼0.361%
case4	LGBMClassifier Pipeline	O_delete SMOKE	Standard Scaler OneHotEncoder LabelEncoder(y)	optuna	90.715%	▲0.434%
case5	LGBMClassifier Pipeline	X	Standard Scaler OneHotEncoder LabelEncoder(y)	optuna	91.112%	▲0.831%
case6	LGBMClassifier Pipeline	X	Standard Scaler OneHotEncoder LabelEncoder(y)	optuna Modification	90.932%	▲0.651%
case7	LGBMClassifier Pipeline	O_Age grouping	Standard Scaler OneHotEncoder LabelEncoder(y)	optuna	90.751%	▲0.470%
case8	LGBMClassifier	O_mapping	Standard Scaler pandas_get_dummies LabelEncoder(y)	optuna	90.751%	▲0.470%
case9	LGBMClassifier	O_create BMI, delete SMOKE	Standard Scaler LabelEncoder(object) LabelEncoder(y)	optuna	87.391%	▼2.890%
case10	LGBMClassifier	O **add original_data**	Standard Scaler LabelEncoder(object) LabelEncoder(y)	optuna **Adjusting thresholds**	92.196%	▲1.915%

Final model

Summary

- Data preprocessing, such as case3 and case9, does not score well

- Better score for case10 with increased number of data

5. Conclusion

5. Conclusion

- Final model Kaggle Leaderboard Public

Public Private

This leaderboard is calculated with approximately 20% of the test data. The final results will be based on the other 80%, so the final standings may be different.

#	Team	Members	Score	Entries	Last	Solution
167	yellayujin		0.92196	14	5d	

→ Ranked **167**th out of a total of **3589** participating teams

5. Conclusion

WHAT WE LEARNED

- It is important to process data for use in predictive models.
- It is important to know the data through statistical analysis.
- While experimenting with various combinations,
  we have gained experience in machine learning.
- Predictions can be made using the LightGBM and pipeline methods.

5. Conclusion

- Final model Kaggle Leaderboard Private

Public Private

The private leaderboard is calculated with approximately 80% of the test data.
This competition has completed. This leaderboard reflects the final standings.

#	△	Team	Members	Score	Entries	Last	Solution
1075	▾ 908	yellayujin		0.90643	14	5d	

→ Ranked **1075**th out of a total of **3589** participating teams

→ Ranked 908 down

5. Conclusion

Regret Points

- There is a big difference between the public score and the private score. This suggests that overflitting has occurred.

Public Score : Calculated with approximately 20% of the test data.

Private Score : Calculated with approximately 80% of the test data.

5. Conclusion

Regret Points

Adjusting the size of test data
(from 0.2 to 0.3)

→

Private Score
from
0.90643
to
0.90661

5. Conclusion

Regret Points

Other solutions available

1. Cross-validation
2. Considering using other models
3.

5. Conclusion

Statistical Theory

- **Data Selection**
- **The Curse of Dimensionality**
  - Preprocessing (binning, derived variables)
  - Variable Selection
- **Hyperparameter Tuning**

5. Conclusion

Statistical Theory

- **Data Selection**
- **The Curse of Dimensionality**
  - Preprocessing (binning, derived variables)
  - Variable Selection
- **Hyperparameter Tuning**

Practical Situations

Thank You