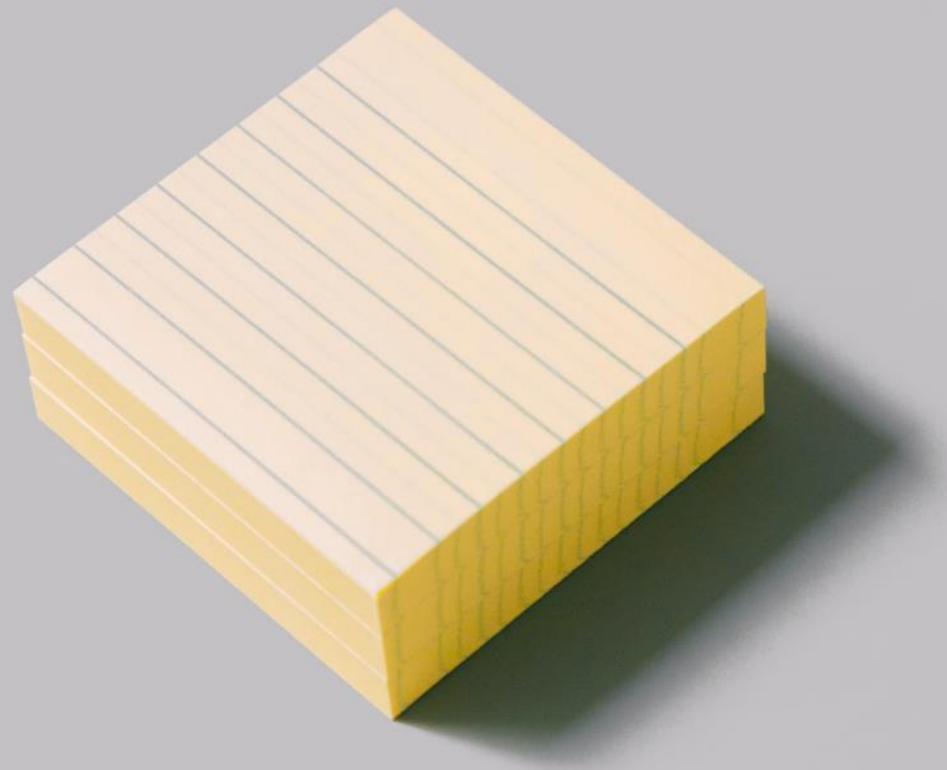


Kaggle Multi-Classification Model Competition

Contents

1. Introduction to the competition
2. Evaluation metrics
3. Modeling Process

1. RandomForest
2. XGBoost
3. CatBoost
4. LightGBM
5. Final Model Selection

1. Introduction to the competition

KAGGLE · PLAYGROUND PREDICTION COMPETITION · 5 DAYS AGO

Late Submission

Multi-Class Prediction of Obesity Risk

Playground Series - Season 4, Episode 2

Topic	Obesity Risk Prediction
Type	Playground
Submission	Simple Competition (CSV file submission)
Host	Kaggle
Problem Type	Multiclass classification
Data Type	Tabular Data
Evaluation Metric	Accuracy
Participating Teams	3587 teams
Duration	24.02.01 ~ 24.02.29, 11:59 PM UTC

Competition Duration

Predicting individuals' obesity risk levels based on various factors such as diverse lifestyle habits and health data.

Participatnts	4 Persosns
Part. Period	7days (2.23~2.29)

1. Introduction to the competition

Dataset Introduction

- Target: Individuals aged 14 to 61 in the countries of Mexico, Peru, and Colombia
- Content: Estimates of obesity levels reflecting various dietary habits and physical conditions
- Creation: Generated through training of deep learning models based on the original obesity risk dataset
  -> Using the original dataset improves performance

Training Dataset

Qty of Data Points: 20,758
Qty of Columns: 18

+

Original Dataset

Qty of Data Points: 2,111
Qty of Columns: 17

→

Combined Dataset

Qty of Data Points: 22,869
Qty of Columns: 17
(excluding the ID column)

No missing values or
duplicate entries.

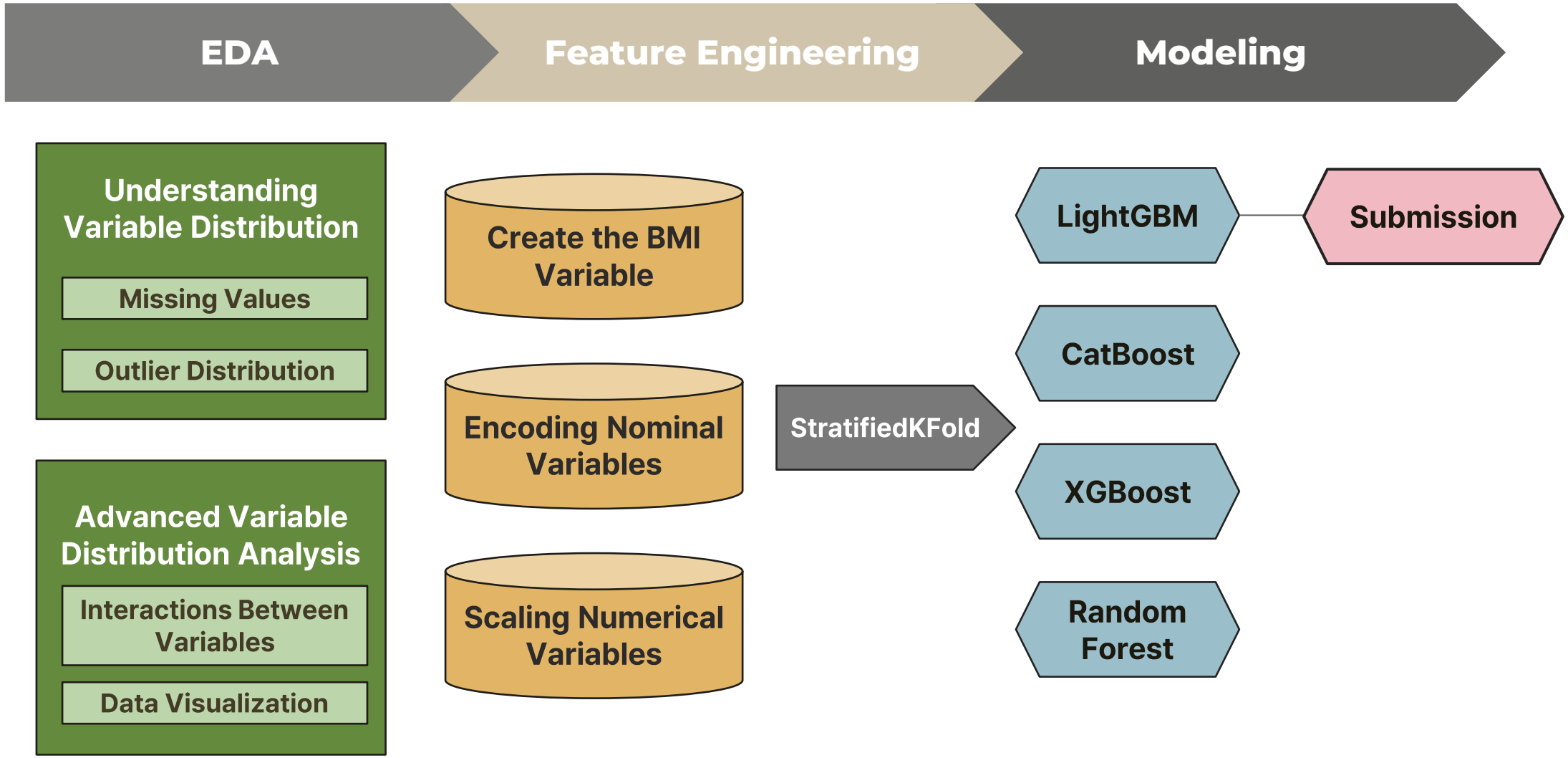
2. Evaluation metrics

Accuracy

- **One of the most basic and intuitive performance evaluation metrics in classification problems.**
- Accuracy $= \frac{Number\ of\ correct\ predictions}{Total\ number\ of\ cases} = \frac{TP+TN}{TP+TN+FP+FN}$

		Actual Correct Label	
		True	False
Predicted Label	True	TP (True Positive)	FP (False Positive)
	False	FN (False Negative)	TN (True Negative)

3. Modeling Process



Part 2

E.D.A

Part 2

1. Imformation about Features in the data

Numeric Variables

Categorical Variables

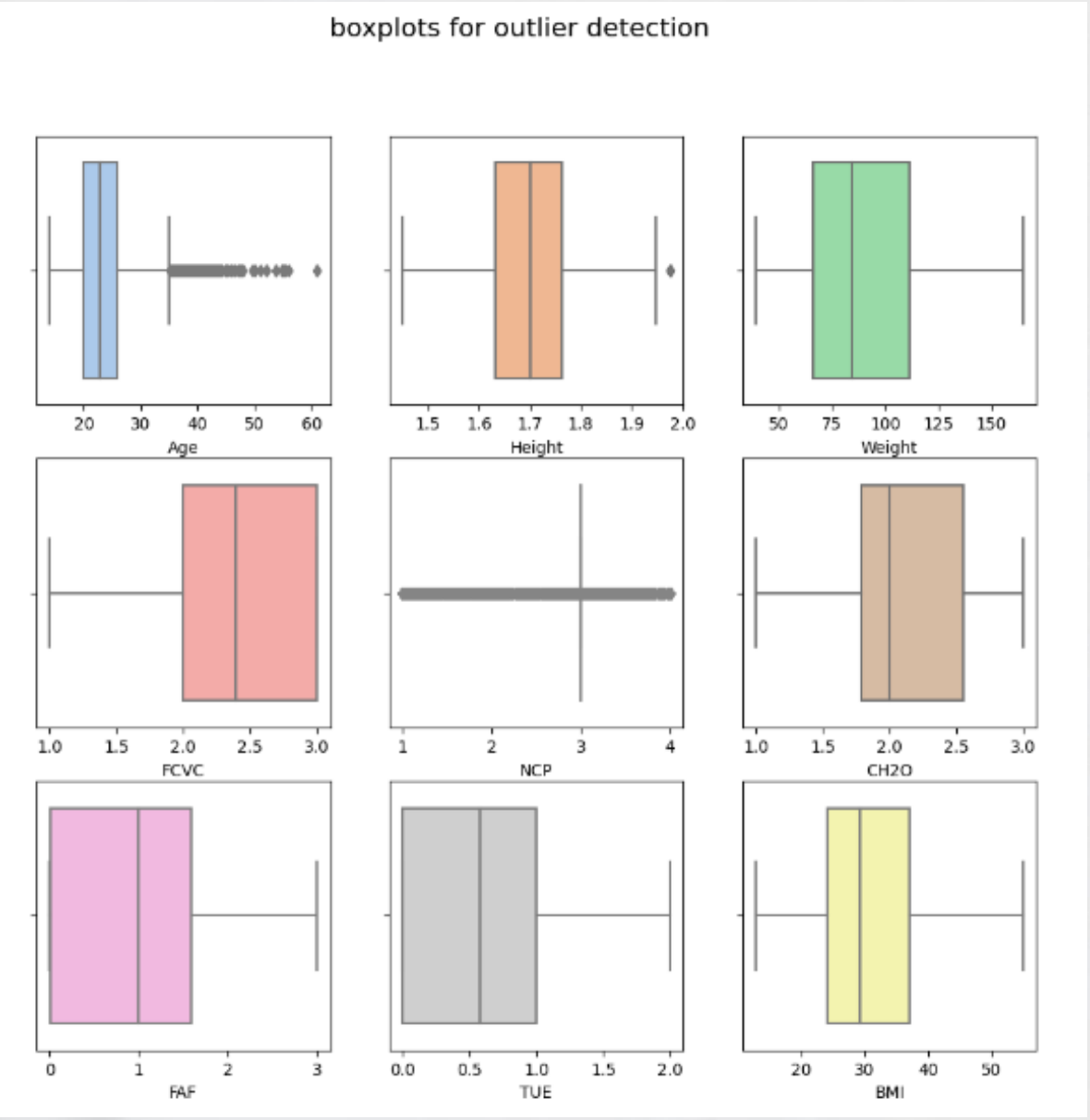
Name	Content	Name	Content
~~id~~	~~Index~~	Gender	Gender
Age	age	Family_history_with_overweight	Presence of Obesity in Family
Height	Height (m^2)	FAVC	Preference for Hight-Calorie Foods
Weight	Weight(Kg)	CAEC	Habits of Snacking Between Meals
FCVC	Vegetable Consumption Frequency per Day	SMOKE	Smoking Status
NCP	Number of Main Meals per Day	SCC	Monitoring Caloric Intake
CH2O	Amount of Water Conumed per Day	CALC	Frequency of Alcohol Consumption
FAF	Weekly Freqeuncy of Physical Activity	MTRANS	Primary Mode of Transportation
TUE	Device Usage Time	NObeyesdad	Obesity Level, Target Variable

Part 2

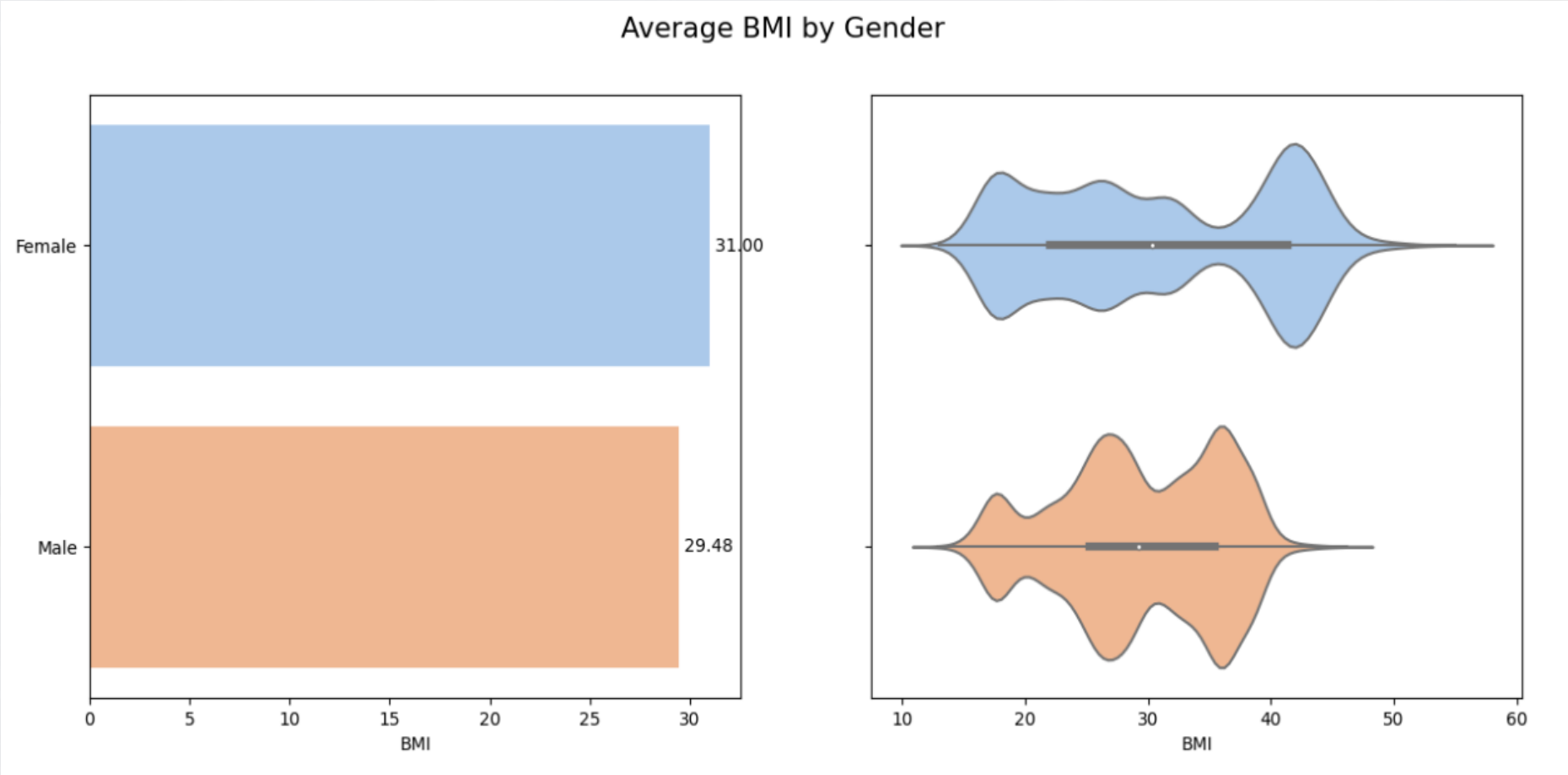
2. EDA

Boxplot visualization to identify outliers

- Found many outliers in ‘Age’
- However, the minimum value is 14 and the maximum is 61, which is in the range within the data description -> considered normal values



2. EDA

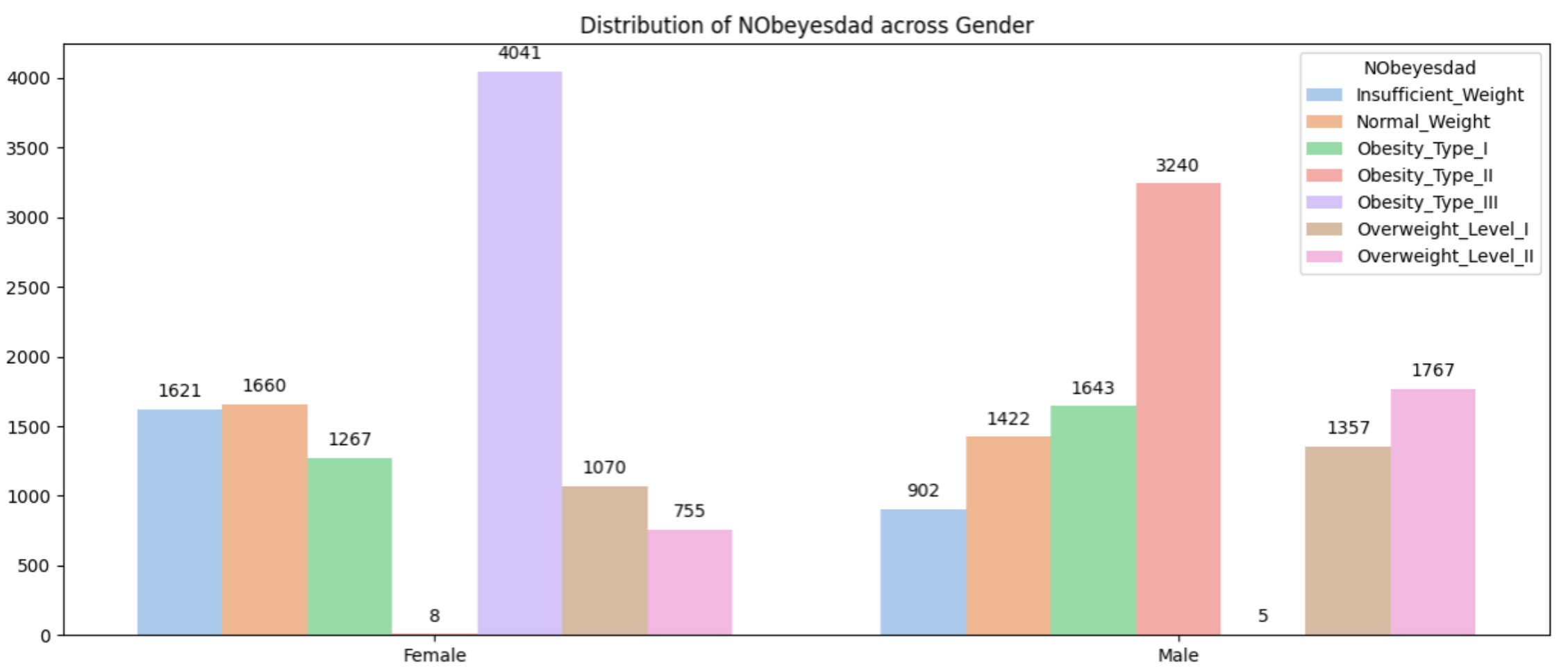


Visualizing average BMI by gender

BMI is slightly higher in the female group

Part 2

2. EDA



Visualize Nobeyesdad distribution by gender

- **Obesisty_Type_III: Mostly observed in women**
- **Obesisty_Type_II: observed primarily in men**

Part 3

MODEL EVALUATION

Part 3

Key algorithms for machine learning

RandomForest Classifier 001

XGBoost 002

CatBoost 003

LightGBM 004

Part 3

1. Random Forest

Preprocessing : StandScaler, OneHotEncoder >> Hyperparameter tuning >> Ensemble

final score : 0.90028
2174/3587

제목	모델 알고리즘	데이터 재가공	feature engineering	교차 검증	하이퍼 파라미터 튜닝	정확도
RandomForest 1	RandomForest Classifier	StandardSclaer : 수치형 데이터 표준화, OneHotEncoder : 범주형 데이터	BMI	**StratifiedKFold**(n_splits=5, shuffle=True, random_state=42) 0.898640994184376		**0.898**
RandomForest 1-1	RandomForest Classifier	StandardSclaer : 수치형 데이터 표준화, OneHotEncoder : 범주형 데이터	BMI	StratifiedKFold(n_splits=5, shuffle=True, random_state=42), 0.9009052302553556	param_dist ={'n_estimators': 120, 'min_samples_split': 5, 'min_samples_leaf': 1, 'max_depth': None}	**0.902**
RandomForest 1-2. ensemble	RandomForest Classifier, GradientBoostingClassifier	StandardSclaer : 수치형 데이터 표준화, OneHotEncoder : 범주형 데이터	BMI		**RandomForestClassifier** : param_dist ={'n_estimators': 120, 'min_samples_split': 5, 'min_samples_leaf': 1, 'max_depth': None}, **GradientBoostingClassifier** : param_dist ={n_estimators : 140, min_samples_split : 9, min_samples_leaf :2, max_depth :5}	**0.905**

Part 3

2. XGBoost

Preprocessing : StandScaler, OneHotEncoder >> Hyperparameter tuning

final score : 0.91076
813/3587

제목	모델 알고리즘	데이터 재가공	feature engineering	교차 검증	하이퍼 파라미터 튜닝	정확도
XGBoost 1	XGBoost	StandardScaler: 수치형 데이터 스케일링OneHotEncoder: 범주형 데이터 인코딩	BMI, 연령대(10단위)	0.904552878	{'classifier__learning_rate': 0.1, 'classifier__max_depth': 5, 'classifier__n_estimators': 200}	**0.906069**
XGBoost 2	XGBoost	StandardScaler: 수치형 데이터 스케일링OneHotEncoder: 범주형 데이터 인코딩	BMI, 연령대(10단위)	StratifiedKFold(n_splits=5, shuffle=True, random_state=42) 0.9035891316836775	classifier__n_estimators': randint(100, 1000), 'classifier__learning_rate': uniform(0.01, 0.6), 'classifier__max_depth': randint(3, 10), 'classifier__colsample_bytree': uniform(0.5, 0.5), 'classifier__subsample': uniform(0.5, 0.5)	**0.908**
XGBoost3	XGBoost	StandardScaler: 수치형 데이터 스케일링OneHotEncoder: 범주형 데이터 인코딩	BMI	StratifiedKFold(n_splits=5, shuffle=True, random_state=42) 0.9081657944146503	pipeline = Pipeline(steps=[('preprocessor', preprocessor), ('classifier', XGBClassifier(subsample=0.7, n_estimators=900, max_depth=4, learning_rate=0.03, colsample_bytree=0.5, use_label_encoder=False, eval_metric='mlogloss'))])	**0.91076**

Part 3

3. Catboost

Preprocessing : FunctionTranformer >> Hyperparameter tuning

제목	모델 알고리즘	데이터 재가공	feature engineering	교차 검증	하이퍼 파라미터 튜닝	정확도
Catboost Model	Catboost	FunctionTransformer(age_rounder:Age반올림/height_rounder:Height반올림/extract_features:BMI구하기/col_rounder:FCVC,NCP,CH2O,FAF,TUE반올림) / .select_dtypes(include=['int64','float64']).columns.tolist(): 수치형 데이터 인코딩 / .select_dtypes(include=['object']).columns.tolist() & .remove('NObeyesdad'): 범주형 데이터 인코딩	BMI	StratifiedKFold(n_splits=5, shuffle=True, random_state=42)'	CB = make_pipeline(CatBoostClassifier(**params, cat_features=categorical_columns)) params = {'learning_rate': 0.13762007048684638, 'depth': 5, 'l2_leaf_reg': 5.285199432056192, 'bagging_temperature': 0.6029582154263095, 'random_seed': RANDOM_SEED, 'verbose': False, 'task_type':"GPU", 'iterations':1000}	**0.911**

Part 3

4. LightGBM

Preprocessing : LabelEncoder >> Hyperparameter tuning

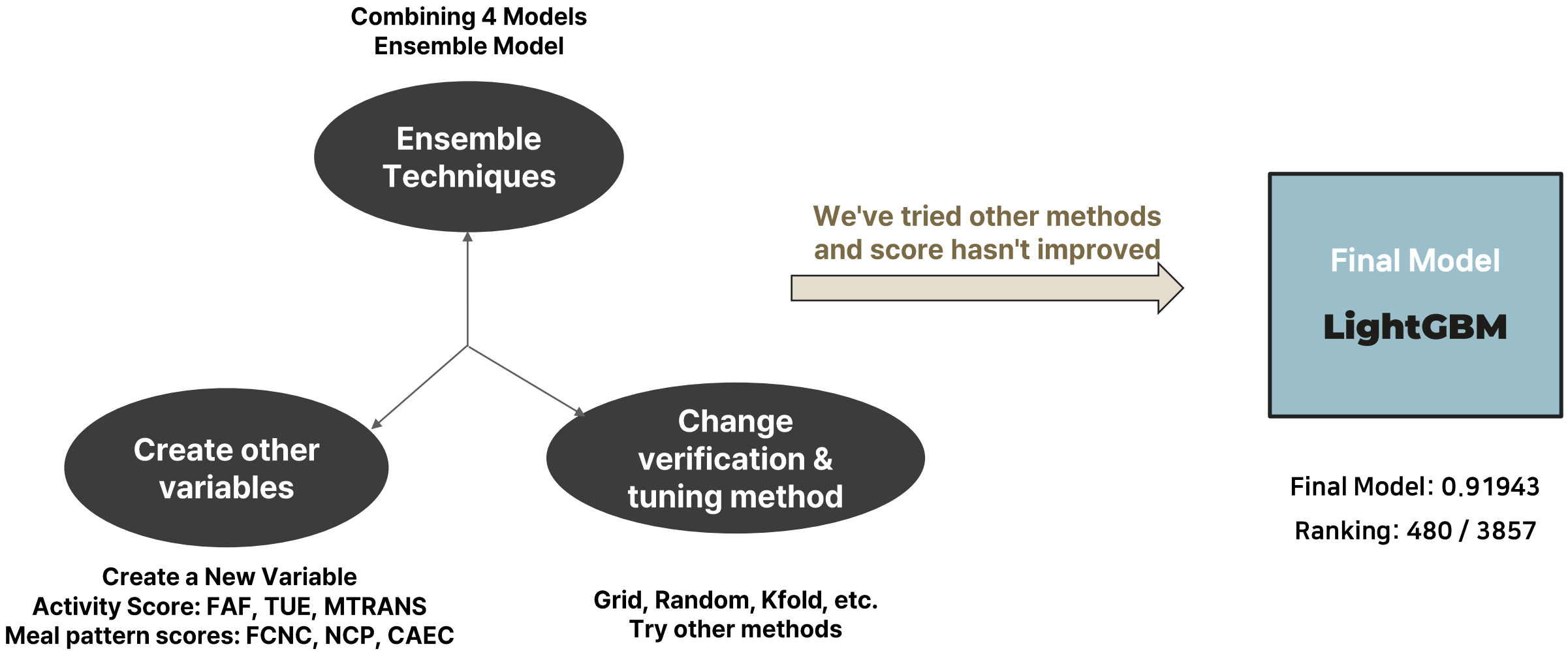
final score : 0.91943
480/3587

1	제목	모델 알고리즘	데이터 재가공	feature engineering	교차 검증	하이퍼 파라미터 튜닝	정확도

2	**LightGBM**	LightGBM	결측치 제거, 중복값 제거,
			LabelEncoder() 사용
			scale_cols = ['Age','Height',
			'Weight','FCVC','NCP','CH2O','
			FAF','TUE']for c in scale_cols:
			X_train[c] =
			X_train[c].pow(0.5)
			X_test[c] =
			X_test[c].pow(0.5)
				BMI	StratifiedKFold(n_spl
					its=5,random_state=
					4,shuffle=True)
						lgbm_params = {
						"objective": "multiclass",
						"metric": "multi_logloss",
						"verbosity": -1,
						"boosting_type": "gbdt",
						"random_state": 42,
						"num_class": 7,
						'learning_rate': 0.031,
						'n_estimators': 550,
						'lambda_l1': 0.010,
						'lambda_l2': 0.040,
						'max_depth': 20,
						'colsample_bytree': 0.413,
						'subsample': 0.97,
						'min_child_samples': 25,
						'class_weight':'balanced'
						}
							0.91943

Part 3

5. Final Model Selction



Conclusion and Insigths Summary

Data Analysis

- Data is structured and easy to analyze with no missing or duplicate values
- Over-representation of non-smokers
- Due to the hypothetical nature of the data, the number of meals and frequency of vegetable intake are decimalized, requiring caution against overfitting

Feature Engineering

- Generated BMI: it was the main variable that helped predict, and was the second most important variable after Weight when visualizing variable importance by model
- Encoding nominal variable labels
- Numerical variable root transformation: Adjusts data distribution closer to a normal distribution, reduces the impact of extreme values
- Wished there were other variables besides BMI that could have been helpful

Model

- LightGBM: Unlike most Gradient Boosting models, the leaf-wise growth strategy reduces the risk of overfitting, which may have contributed to the high accuracy.
- Potential for improvement: Performance could be improved with other ensemble combinations, stacking methods, etc.

Thank you!

Team Leader: Jigeon Park

Team member: Suhyeon Kim

Team member: Indong Yang

Team member: Indong Song