

ad_click

December 21, 2025

```
[116]: # Import necessary libraries
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
from scipy import stats
from sklearn.base import BaseEstimator, TransformerMixin
from sklearn.model_selection import train_test_split
from sklearn.metrics import accuracy_score, precision_score, recall_score, \
    f1_score, roc_curve, auc, \
    confusion_matrix, classification_report, roc_auc_score
from sklearn.metrics import roc_auc_score, log_loss, precision_recall_curve, \
    classification_report
from sklearn.preprocessing import StandardScaler
from sklearn.linear_model import LogisticRegression
from sklearn.neighbors import KNeighborsClassifier
from sklearn.tree import DecisionTreeClassifier
from sklearn.ensemble import RandomForestClassifier, \
    GradientBoostingClassifier, AdaBoostClassifier
from sklearn.svm import SVC
from sklearn.naive_bayes import GaussianNB
from sklearn.compose import ColumnTransformer
from sklearn.model_selection import GridSearchCV, RandomizedSearchCV
from sklearn.pipeline import Pipeline
import requests
import io
from imblearn.over_sampling import SMOTE
import warnings
warnings.filterwarnings("ignore")
from xgboost import XGBClassifier
from lightgbm import LGBMClassifier
sns.set_theme(style="dark")
pd.set_option('display.max_columns', 0)
plt.style.use('ggplot')
pd.options.display.float_format = '{:.2f}'.format
from sklearn.preprocessing import OneHotEncoder, StandardScaler
```

```
[117]: df = pd.read_csv('./Ad_click_prediction_train.csv')
print(df.shape)
df.head()
```

(463291, 15)

```
[117]: session_id      DateTime  ...  var_1  is_click
0      140690  2017-07-02 00:00  ...      0          0
1      333291  2017-07-02 00:00  ...      0          0
2      129781  2017-07-02 00:00  ...      0          0
3      464848  2017-07-02 00:00  ...      1          0
4       90569  2017-07-02 00:01  ...      1          0
```

[5 rows x 15 columns]

```
[118]: df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 463291 entries, 0 to 463290
Data columns (total 15 columns):
#   Column                                Non-Null Count  Dtype
---  -
0   session_id                            463291 non-null  int64
1   DateTime                              463291 non-null  object
2   user_id                               463291 non-null  int64
3   product                               463291 non-null  object
4   campaign_id                           463291 non-null  int64
5   webpage_id                            463291 non-null  int64
6   product_category_1                    463291 non-null  int64
7   product_category_2                    97437 non-null   float64
8   user_group_id                         445048 non-null  float64
9   gender                                445048 non-null  object
10  age_level                             445048 non-null  float64
11  user_depth                            445048 non-null  float64
12  city_development_index                 338162 non-null  float64
13  var_1                                 463291 non-null  int64
14  is_click                              463291 non-null  int64
dtypes: float64(5), int64(7), object(3)
memory usage: 53.0+ MB
```

```
[119]: df.columns
```

```
[119]: Index(['session_id', 'DateTime', 'user_id', 'product', 'campaign_id',
        'webpage_id', 'product_category_1', 'product_category_2',
        'user_group_id', 'gender', 'age_level', 'user_depth',
        'city_development_index', 'var_1', 'is_click'],
        dtype='object')
```

```
[120]: # Calculate the count of each class label in the "is_click" column
class_counts = df["is_click"].value_counts()

# Print the class distribution
print("Class Distribution:")
for class_label, count in class_counts.items():
    print(f"Class {class_label}: {count} samples")

print("CTR: ",(class_counts[1]/len(df))*100)
```

```
Class Distribution:
Class 0: 431960 samples
Class 1: 31331 samples
CTR: 6.762704218299081
```

```
[121]: for col in df.columns:
        print(col,": ",df[col].nunique())
```

```
session_id : 463291
DateTime   : 8610
user_id    : 150347
product    : 10
campaign_id : 10
webpage_id : 9
product_category_1 : 5
product_category_2 : 29
user_group_id : 13
gender     : 2
age_level  : 7
user_depth : 3
city_development_index : 4
var_1      : 2
is_click   : 2
```

```
[122]: print("Missing Values Percentage:\n",(df.isna().sum()/len(df))*100)
nan_cols = df.columns[df.isna().sum()!=0]
print("#"*100)
print("Columns Which has NaN values",list(nan_cols))
```

```
Missing Values Percentage:
  session_id      0.00
DateTime         0.00
user_id          0.00
product          0.00
campaign_id      0.00
webpage_id       0.00
product_category_1 0.00
product_category_2 78.97
user_group_id     3.94
```

```

gender                3.94
age_level             3.94
user_depth            3.94
city_development_index 27.01
var_1                 0.00
is_click              0.00
dtype: float64
#####
#####
Columns Which has NaN values ['product_category_2', 'user_group_id', 'gender',
'age_level', 'user_depth', 'city_development_index']

```

Extracting required features from DateTime column

```

[123]: def extract_features_from_datetime(data, col):
        def get_time_of_day(hour):
            if 6 <= hour <= 11:
                return 'morning'
            elif 12 <= hour <= 17:
                return 'afternoon'
            elif 18 <= hour <= 21:
                return 'evening'
            else:
                return 'night'

        data[col] = pd.to_datetime(data[col])
        data['day_in_week'] = data[col].dt.dayofweek
        data['day_in_month'] = data[col].dt.day
        data['month'] = data[col].dt.month
        data['hour'] = data[col].dt.hour
        data['time_of_day'] = data['hour'].apply(get_time_of_day)
        return data

df = extract_features_from_datetime(df, 'DateTime')

```

```

[124]: df['gender'] = df['gender'].map({'Female':0, 'Male':1})

```

0.0.1 Exploratory Data Analysis

- As we can see there is a huge imbalance in target variable we need to handle this

Univariate Analysis

```

[125]: df.hist(figsize=(20,15), bins=50)

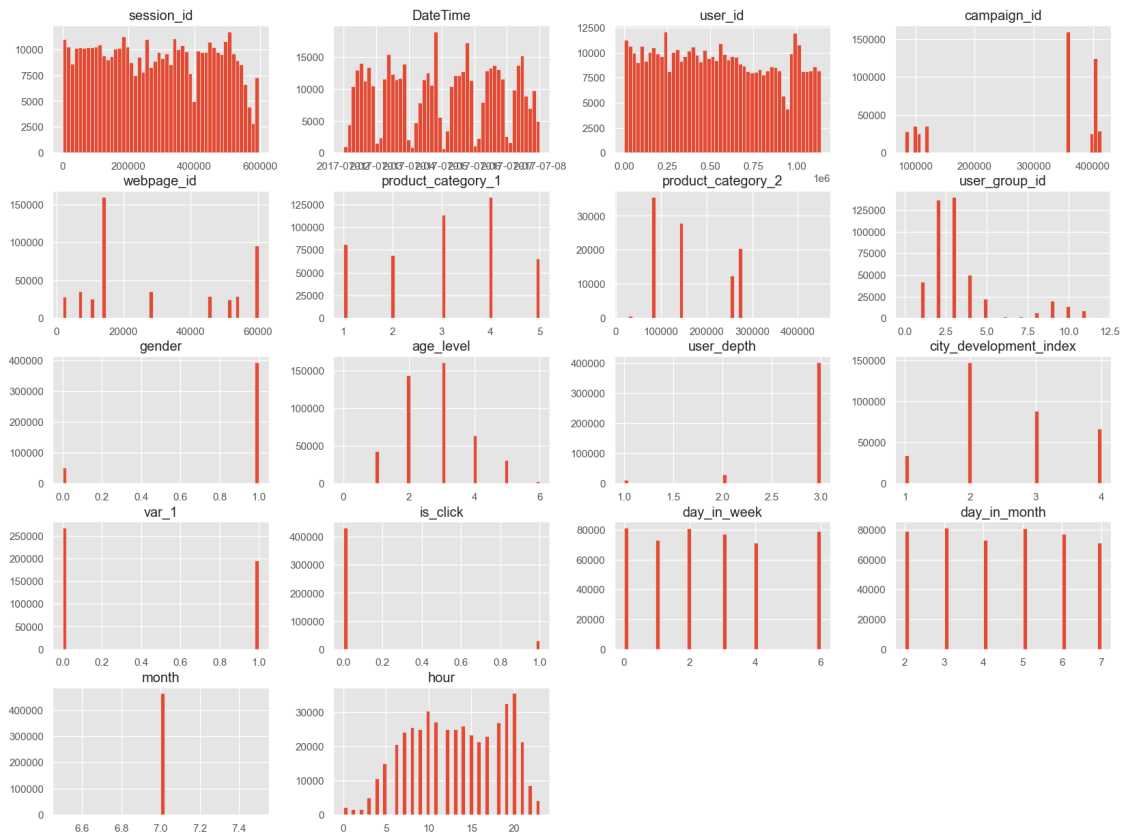
```

```

[125]: array([[<Axes: title={'center': 'session_id'}>,
               <Axes: title={'center': 'DateTime'}>,
               <Axes: title={'center': 'user_id'}>,
               <Axes: title={'center': 'campaign_id'}>],

```

```
[<Axes: title={'center': 'webpage_id'}>,
 <Axes: title={'center': 'product_category_1'}>,
 <Axes: title={'center': 'product_category_2'}>,
 <Axes: title={'center': 'user_group_id'}>],
 [<Axes: title={'center': 'gender'}>,
 <Axes: title={'center': 'age_level'}>,
 <Axes: title={'center': 'user_depth'}>,
 <Axes: title={'center': 'city_development_index'}>],
 [<Axes: title={'center': 'var_1'}>,
 <Axes: title={'center': 'is_click'}>,
 <Axes: title={'center': 'day_in_week'}>,
 <Axes: title={'center': 'day_in_month'}>],
 [<Axes: title={'center': 'month'}>,
 <Axes: title={'center': 'hour'}>, <Axes: >, <Axes: >]],
 dtype=object)
```

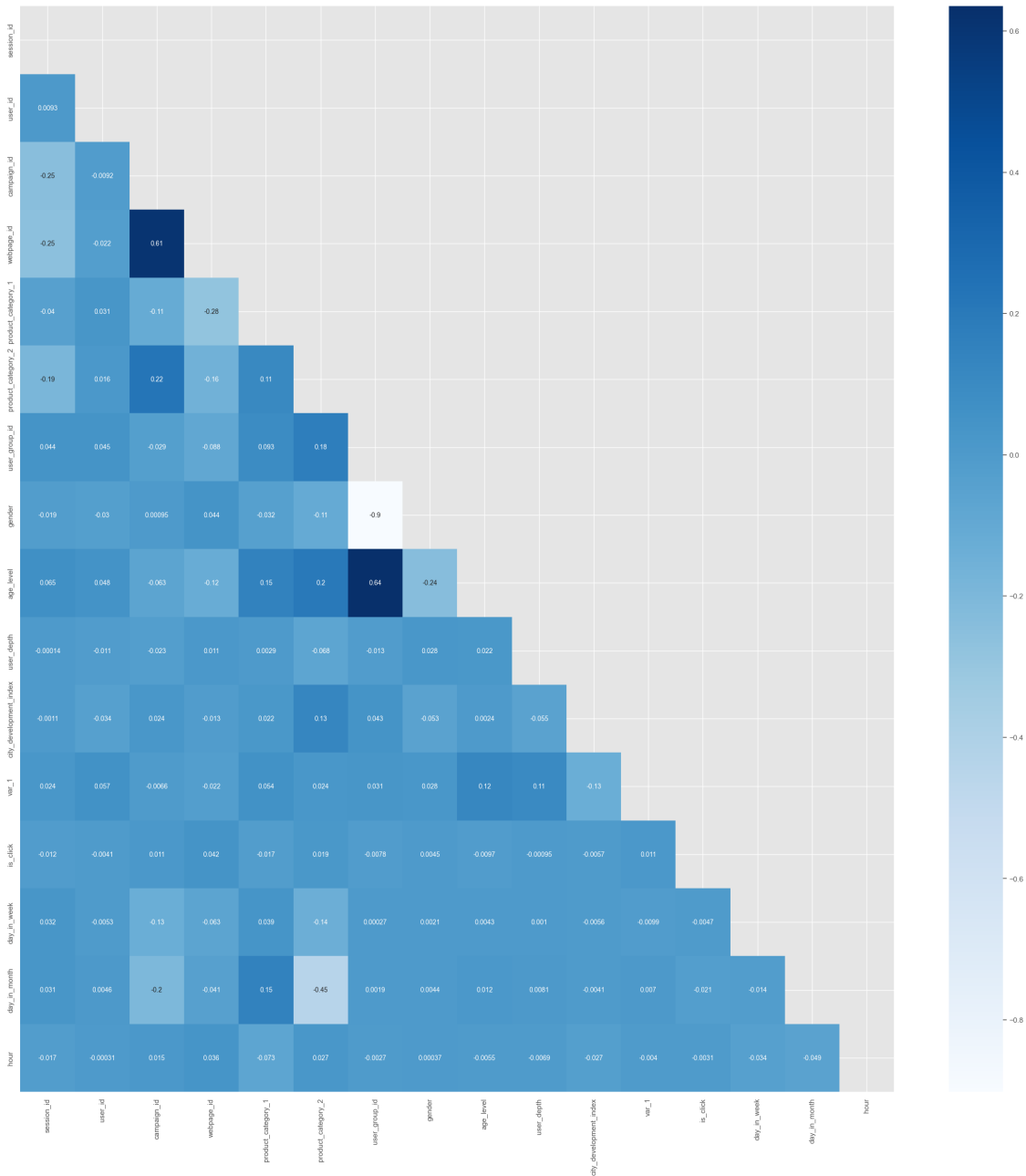


- As we can see is_click and gender columns are having severe imbalance, var_1 columns has decent imbalance
- There is only 7th month related data, so month columns can be dropped because it doesn't have any significance
- user_depth columns also is highly skewed, most of the datapoint are of value 3

- We need to handle class imbalance

```
[126]: df.drop('month',axis=1,inplace=True)
```

```
[127]: corr_mat = df.corr(numeric_only=True)
plt.figure(figsize=(30, 30))
mask = np.zeros_like(corr_mat)
mask[np.triu_indices_from(mask)] = True
corr_heatmap=sns.heatmap(df.corr(numeric_only=True), cmap="Blues", annot=True,
↪mask=mask)
```



- Most features are only very weakly correlated with each other, so there is little multicollinearity and no need to drop features just to “fix correlation”.
- Overall, the heatmap confirms that predictive signal must come from non-linear interactions rather than simple linear relationships, which matches the modest gains you see even with boosted trees.

0.1 Bivariate Analysis

Date and time related Analysis

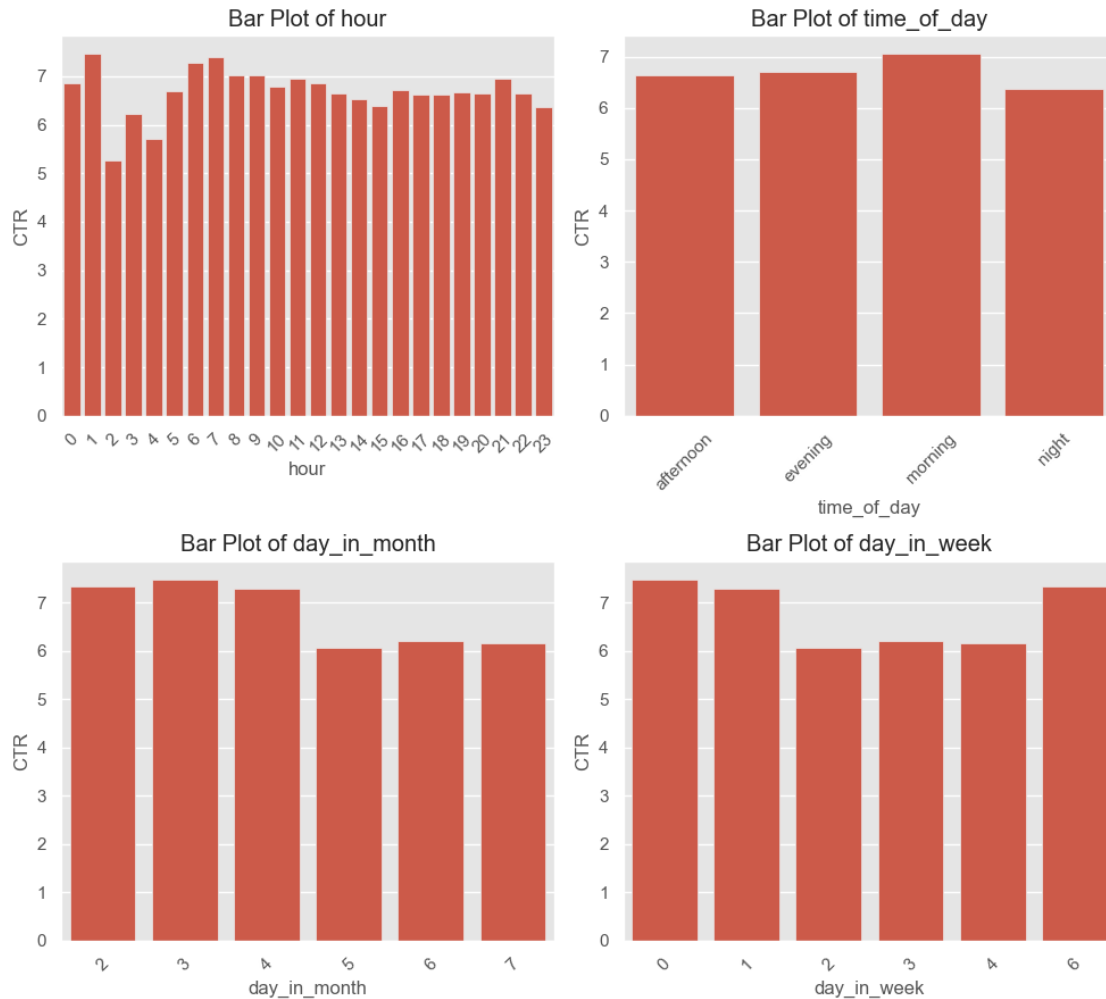
```
[128]: def calculate_ctr(group):
        clicks = group['is_click'].sum()
        impressions = group['session_id'].count()
        return np.round((clicks/impressions)*100,2)

[129]: def visualize_group_wise_ctr(df, groups,num_cols=2, size=(10,9)):
        # Determine the number of columns and rows for subplots
        num_rows = (len(groups) + 1) // num_cols # Calculate the number of rows
        ↪needed
        fig, axes = plt.subplots(num_rows, num_cols, figsize=size) # Adjust the
        ↪figure size as needed
        axes = axes.flatten() # Flatten the 2D array of axes for easy iteration

        # Iterate through categorical columns and create bar plots
        for i, column in enumerate(groups):
            ax = axes[i]
            group_wise_ctr = df.groupby(column).apply(calculate_ctr).
            ↪reset_index(name='CTR')
            sns.barplot(data=group_wise_ctr, x=column,y='CTR', ax=ax)
            ax.set_title(f'Bar Plot of {column}')
            ax.set_xlabel(column)
            ax.tick_params(axis='x', labelrotation=45)
            ax.set_ylabel('CTR')

        # Adjust the layout and display the plots
        plt.tight_layout()
        plt.show()

[130]: groups = ['hour','time_of_day','day_in_month','day_in_week']
        visualize_group_wise_ctr(df,groups)
```



- Hour: CTR is fairly stable but slightly higher in the early morning and around 7–10 AM, with a mild dip during late night and some mid-day hours.
- Time of day: Morning has the highest CTR, followed by evening/afternoon; night performs noticeably worse.
- Day in month: CTR is a bit higher in the first few days (2–4) and slightly lower around days 5–7, though variation is modest.
- Day of week: CTR peaks on day 0 and 6 (likely Sunday and Monday), and is lower mid-week, indicating better performance on weekends.

User features relative analysis

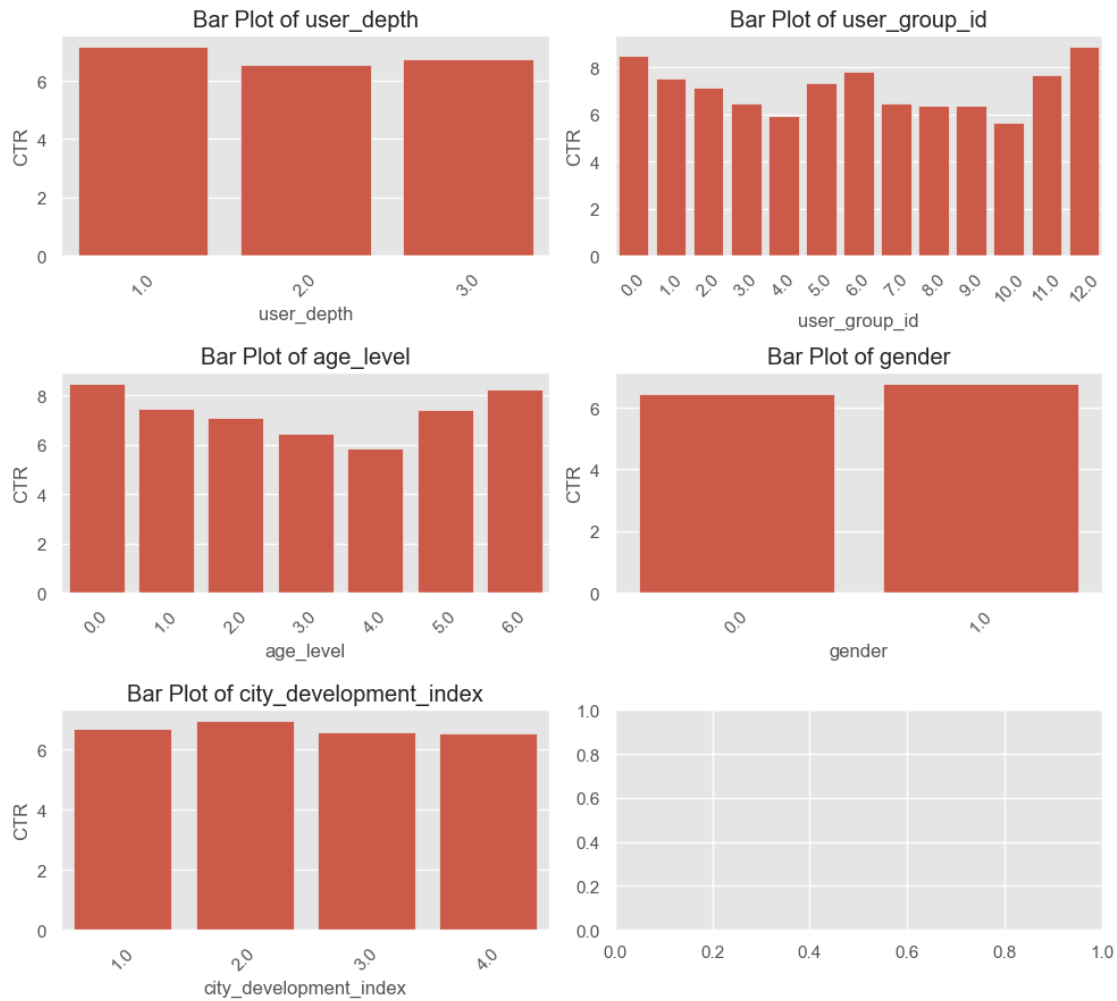
```
[131]: df.columns
```

```
[131]: Index(['session_id', 'DateTime', 'user_id', 'product', 'campaign_id',  
         'webpage_id', 'product_category_1', 'product_category_2',
```



```
'user_group_id', 'gender', 'age_level', 'user_depth',
'city_development_index', 'var_1', 'is_click', 'day_in_week',
'day_in_month', 'hour', 'time_of_day'],
dtype='object')
```

```
[132]: groups = [
    ↪ ['user_depth', 'user_group_id', 'age_level', 'gender', 'city_development_index']
visualize_group_wise_ctr(df, groups)
```



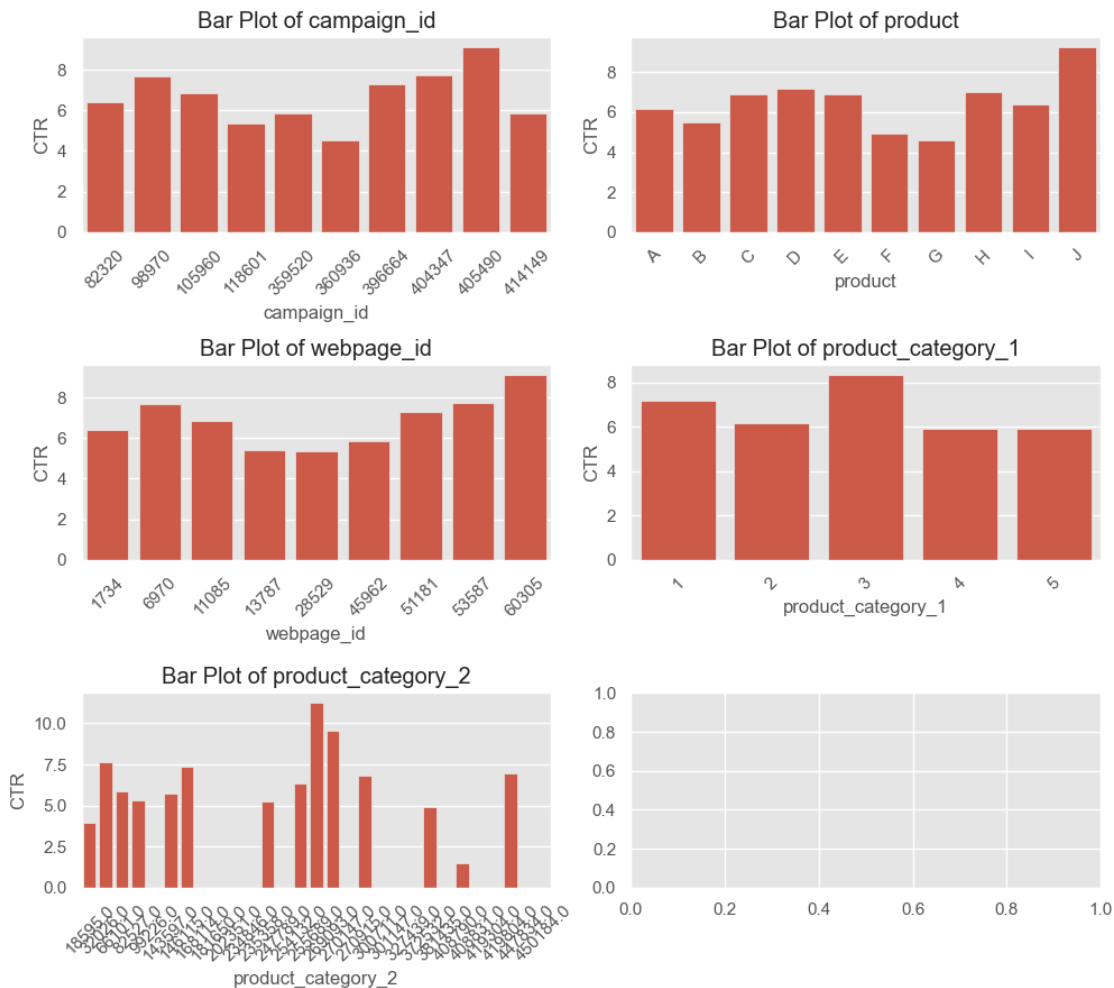
- User depth: CTR is very similar across depth levels, so depth alone does not strongly differentiate click propensity.
- User group: Certain user_group_ids (especially at the ends of the scale) show noticeably higher CTR, suggesting some segments are more responsive than others and are valuable for targeting.
- Age level: Extremes (youngest and oldest buckets) have higher CTR, with a dip in middle

age bands, indicating mild U-shaped behaviour by age.

- Gender: CTR is almost identical across genders, so gender adds little signal in this dataset.
- City development index: CTR is fairly flat across city tiers, implying limited difference in click behaviour by city development level.

Campaign Related Analysis

```
[133]: groups = [
    ↪ ['campaign_id', 'product', 'webpage_id', 'product_category_1', 'product_category_2']
visualize_group_wise_ctr(df, groups, num_cols=2)
```



- campaign_id: CTR varies noticeably by campaign; some campaigns (e.g., the right-most one) perform clearly better, indicating strong creative/targeting differences across campaigns.
- product: Products show distinct CTR levels, with certain products (e.g., “J”) attracting substantially more clicks than others, so item choice matters a lot.

- webpage_id: Different placements/pages yield different CTRs; a few webpages are clear high performers, suggesting valuable inventory for ad serving.
- product_category_1: Category 3 has the highest CTR, while others are similar or slightly lower, implying some product categories are inherently more engaging.
- product_category_2: CTR is highly uneven across secondary categories, with some rare categories showing very high CTR, though small sample sizes may inflate these rates

0.2 Feature Engineering

Interaction Features

```
[134]: df.columns
```

```
[134]: Index(['session_id', 'DateTime', 'user_id', 'product', 'campaign_id',
        'webpage_id', 'product_category_1', 'product_category_2',
        'user_group_id', 'gender', 'age_level', 'user_depth',
        'city_development_index', 'var_1', 'is_click', 'day_in_week',
        'day_in_month', 'hour', 'time_of_day'],
        dtype='object')
```

```
[135]: def add_interaction_features(df):
        df['user_product_interaction'] = df['user_id'].astype(str) + '_' +
        df['product'].astype(str)
        df['product_age'] = df['product'].astype(str) + '_' + df['age_level'].
        astype(str)
        df['product_cat_1_age'] = df['product_category_1'].astype(str) + '_' +
        df['age_level'].astype(str)
        df['campaign_webpage'] = df['campaign_id'].astype(str) + '_' +
        df['webpage_id'].astype(str)
        df['gender_age'] = df['gender'].astype(str) + '_' + df['age_level'].
        astype(str)
        return
        ['campaign_webpage', 'gender_age', 'user_product_interaction', 'product_age', 'product_cat_1_age']

interaction_features = add_interaction_features(df)
df[['campaign_webpage', 'gender_age', 'user_product_interaction', 'product_age', 'product_cat_1_age']]
    head()
```

```
[135]:   campaign_webpage  gender_age  ...  product_age  product_cat_1_age
0      359520_13787    0.0_4.0  ...      C_4.0          4_4.0
1      105960_11085    0.0_2.0  ...      C_2.0          5_2.0
2      359520_13787    0.0_2.0  ...      C_2.0          4_2.0
3      359520_13787    1.0_3.0  ...      I_3.0          3_3.0
4      405490_60305    1.0_2.0  ...      C_2.0          3_2.0
```

[5 rows x 5 columns]

```
[136]: df.groupby('campaign_webpage')['is_click'].mean().describe()
```

```
[136]: count    10.00
      mean     0.07
      std      0.01
      min      0.05
      25%      0.06
      50%      0.07
      75%      0.08
      max      0.09
      Name: is_click, dtype: float64
```

- CTR varies from 4% to 9.1%, and have a moderate variance, low cardinality
- Campaign performance depends on placement (webpage).

```
[137]: df.groupby('gender_age')['is_click'].mean().describe()
```

```
[137]: count    15.00
      mean     0.07
      std      0.02
      min      0.03
      25%      0.06
      50%      0.07
      75%      0.08
      max      0.10
      Name: is_click, dtype: float64
```

- CTR varies from 3% to 10% good indicator to keep this feature, variance is about 1.5% which is moderate
- If the CTR distribution across interaction groups has meaningful variance, the interaction feature is useful.

```
[138]: df.groupby('product_age')['is_click'].mean().describe()
```

```
[138]: count    80.00
      mean     0.07
      std      0.04
      min      0.00
      25%      0.05
      50%      0.06
      75%      0.08
      max      0.33
      Name: is_click, dtype: float64
```

- CTR varies from 5% to 33% that means some product is loved by user of certain age level

```
[139]: df.groupby('user_product_interaction')['is_click'].mean().describe()
```

```
[139]: count    245846.00
      mean         0.08
      std         0.24
```

```

min          0.00
25%          0.00
50%          0.00
75%          0.00
max          1.00
Name: is_click, dtype: float64

```

- There are so many users who don't like certain products and there are some users who click everytime for certain products

0.3 Data Preprocessing

Handling NaN values

```

[140]: def replace_nan(df):
        # since these two columns are having 78% and 28% missing values, so
        ↪ replacing nan with -1 which acts
        # as a new category for missing values
        df['product_category_2'] = df['product_category_2'].fillna(-1).
        ↪ astype('int64')

        for col in ['age_level', 'user_depth', 'user_group_id', 'gender']:
            mode_val = df[col].mode()[0]
            df[col] = df[col].fillna(mode_val)

        # city_development_index: treat as ordinal category with missing level 0
        df['city_development_index'] = df['city_development_index'].fillna(0).
        ↪ astype('int64')

```

```

[141]: replace_nan(df)
df.columns[df.isna().sum() != 0]

```

```

[141]: Index([], dtype='object')

```

Time-based train/validation split

```

[142]: target_col = 'is_click'

cut = int(len(df) * 0.8)
df_train = df.iloc[:cut].copy()
df_val    = df.iloc[cut:].copy()

print("Original CTR:", df['is_click'].mean())
print("Train CTR:", df_train['is_click'].mean())
print("Validation CTR:", df_val['is_click'].mean())

```

```

Original CTR: 0.06762704218299082
Train CTR: 0.06913596235619159
Validation CTR: 0.06159142662882181

```

Aggregate Features

```
[143]: def build_agg_tables(train, entity_cols, target, global_ctr, alpha):
    """
    From train, build one aggregate table per entity col.
    Returns dict: {col_name: agg_df}
    Each agg_df has columns: [col, f'{col}_impr', f'{col}_ctr'].
    """
    agg_tables = {}

    for col in entity_cols:
        grp = train.groupby(col)[target].agg(['count', 'sum']).reset_index()
        grp.columns = [col, f'{col}_impr', f'{col}_clicks']

        # optional: raw CTR for analysis
        grp[f'{col}_ctr_raw'] = grp[f'{col}_clicks'] / grp[f'{col}_impr']

        # smoothed CTR
        grp[f'{col}_ctr'] = (grp[f'{col}_clicks'] + alpha * global_ctr) / (
            grp[f'{col}_impr'] + alpha
        )

        # keep only what you need for merging
        agg_tables[col] = grp[[col, f'{col}_impr', f'{col}_ctr']]

    return agg_tables

global_ctr = df_train[target_col].mean()
alpha = 50
agg_entities = ['user_group_id', 'age_level', 'product',
                'campaign_id', 'webpage_id', 'product_category_1']

agg_tables = build_agg_tables(df_train, agg_entities, target_col, global_ctr,
                               alpha)

def merge_agg_features(df, agg_tables, entity_cols, global_ctr):
    """
    Merge precomputed aggregate tables into any df (train/val/test).
    """
    df = df.copy()
    for col in entity_cols:
        table = agg_tables[col]
        df = df.merge(table, on=col, how='left')
        df[f'{col}_impr'] = df[f'{col}_impr'].fillna(0)
        df[f'{col}_ctr'] = df[f'{col}_ctr'].fillna(global_ctr)
    return df

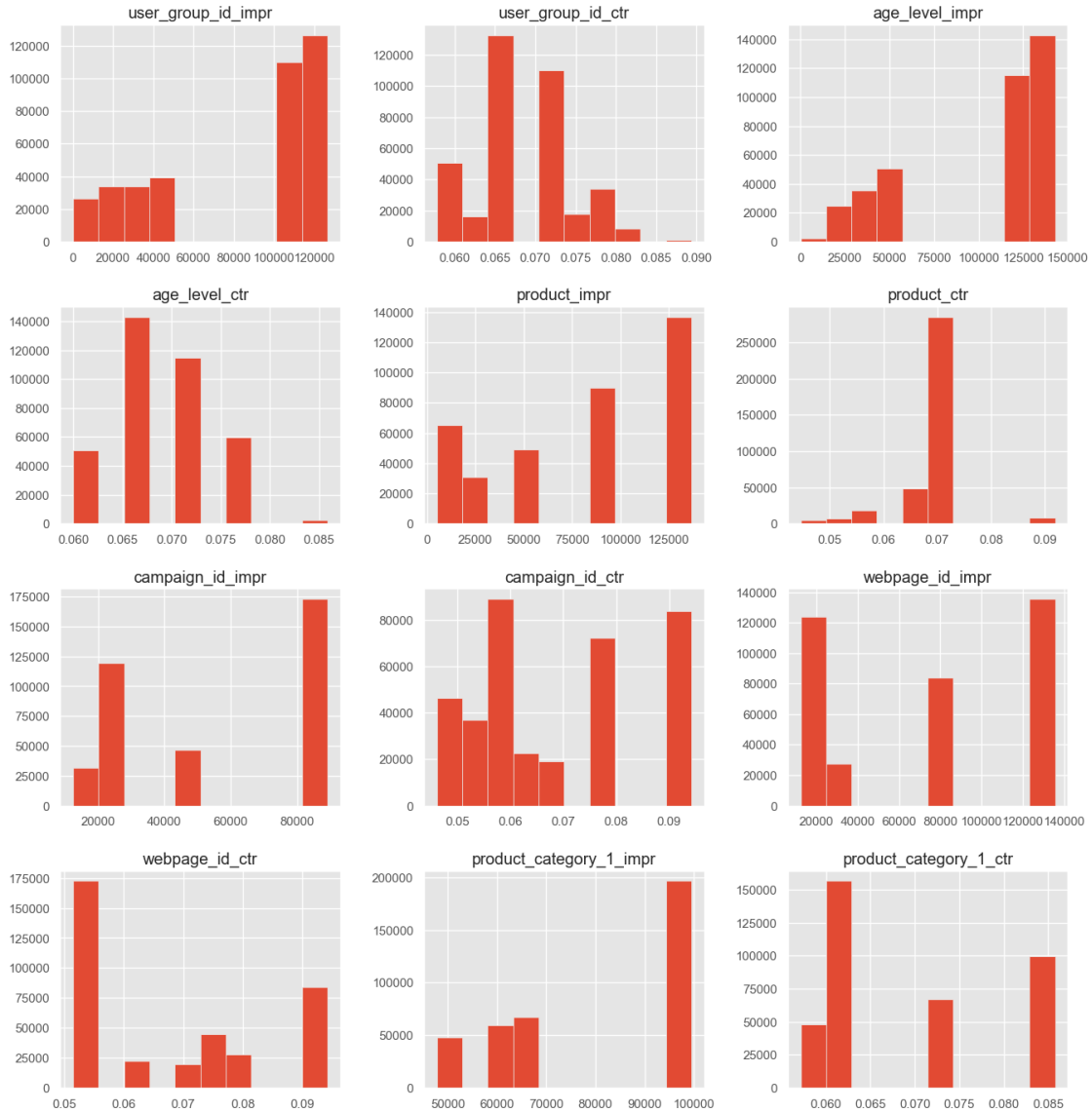
# train/val
```

```
df_train = merge_agg_features(df_train, agg_tables, agg_entities, global_ctr)
df_val    = merge_agg_features(df_val,    agg_tables, agg_entities, global_ctr)
```

- To avoid data leakage of test data in calculating the aggregated features, using only train dataframe and merge that data on test dataframe

```
[29]: df_train[df_train.columns[df_train.columns.str.endswith(('ctr', 'impr'))]].  
      ↪ hist(figsize=(16,17))
```

```
[29]: array([[<Axes: title={'center': 'user_group_id_impr'}>,  
             <Axes: title={'center': 'user_group_id_ctr'}>,  
             <Axes: title={'center': 'age_level_impr'}>],  
            [<Axes: title={'center': 'age_level_ctr'}>,  
             <Axes: title={'center': 'product_impr'}>,  
             <Axes: title={'center': 'product_ctr'}>],  
            [<Axes: title={'center': 'campaign_id_impr'}>,  
             <Axes: title={'center': 'campaign_id_ctr'}>,  
             <Axes: title={'center': 'webpage_id_impr'}>],  
            [<Axes: title={'center': 'webpage_id_ctr'}>,  
             <Axes: title={'center': 'product_category_1_impr'}>,  
             <Axes: title={'center': 'product_category_1_ctr'}>]], dtype=object)
```



Encoding of Categorical Variables

Temporal Features encoding

- Doing a cyclic encoding because in temporal features like hour 24 is close to 0, and weekday 6 is close to 0 it is a cyclic relation, if we don't do cyclic encoding using sin or cos model doesn't understand this

```
[30]: def add_cyclic_features(
      df,
      cyclic_config,
      drop_original=True
```



```

):
    df = df.copy()

    for col, period in cyclic_config.items():
        if col not in df.columns:
            continue

        df[f"{col}_sin"] = np.sin(2 * np.pi * df[col] / period)
        df[f"{col}_cos"] = np.cos(2 * np.pi * df[col] / period)

        if drop_original:
            df.drop(columns=[col], inplace=True)

    return df

cyclic_features = {
    "hour": 24,
    "day_in_week": 7,
    "day_in_month": 31,
}

df_train = add_cyclic_features(df_train, cyclic_features)
df_val = add_cyclic_features(df_val, cyclic_features)

```

- Hour, day_of_week and similar time features are periodic. Using integers 0–23 or 0–6 makes endpoints (23 vs 0, 6 vs 0) look far apart and breaks the circular structure. Cyclical encoding maps these features to sin/cos on the unit circle, so adjacent times remain close in feature space while using only two columns per feature. This helps models learn daily/weekly patterns without artificial discontinuities and often improves performance for non-tree models.

```

[31]: X_train = df_train.drop(columns='is_click')
      y_train = df_train['is_click']
      X_val = df_val.drop(columns='is_click')
      y_val = df_val['is_click']

```

Feature Selection

- remove session_id because it is constant and doesn't have any significant values in the dataset
- remove DateTime because we have already extracted required features from it
- remove user_id to avoid overfitting for each user and generalise the model, it is also having too many unique values

```

[32]: X_train.drop(columns=['session_id', 'DateTime', 'user_id'], inplace=True)
      X_val.drop(columns=['session_id', 'DateTime', 'user_id'], inplace=True)

```

Handle Imbalance

- Since dataset have too many categorical columns we cannot use SMOTE technique because this may include new ids etc

- We can use scale positive weights parameter while training XGBoost/LightGBM models
- And choose appropriate metrics like f1 score, precision, recall and AUC-ROC, should not use accuracy
- Using Random Over Sampling technique also increases impressions which are not real.
- SMOTE technique may help in reducing false negative by synthetically adding points in minority class (is_click=1), but SMOTE cannot be used in CTR dataset as we have seen above it is creating new categories which are not real.

0.4 Model Building And Evaluation

```
[33]: def evaluate_model(name, y_true, y_proba):
    auc = roc_auc_score(y_true, y_proba)
    ll = log_loss(y_true, y_proba)

    prec, rec, thr = precision_recall_curve(y_true, y_proba)
    f1_scores = 2 * prec * rec / (prec + rec + 1e-8)
    best_idx = np.argmax(f1_scores)
    best_thr = thr[best_idx]
    best_f1 = f1_scores[best_idx]

    y_pred = (y_proba >= best_thr).astype(int)

    print(f"\n=== {name} ===")
    print(f"AUC: {auc:.4f}")
    print(f"LogLoss: {ll:.4f}")
    print(f"Best F1 threshold: {best_thr:.3f}")
    print(f"Best F1: {best_f1:.4f}")
    print(classification_report(y_true, y_pred))

    return {
        "model": name,
        "auc": auc,
        "logloss": ll,
        "best_thr": best_thr,
        "best_f1": best_f1
    }
```

Logistic Regression

- Interaction features are having high cardinality so calculating aggregates for them, and dropping these columns

```
[34]: def add_interaction_aggregates(X_train, y_train, X_test, combo_cols, alpha=50):
    # create temp train df with target
    train = X_train.copy()
    train['__target__'] = y_train.values
```

```

global_ctr = train['__target__'].mean()

for col in combo_cols:
    grp = (
        train
        .groupby(col)['__target__']
        .agg(['count', 'sum'])
        .reset_index()
    )
    grp.columns = [col, f'{col}_impr', f'{col}_clicks']

    grp[f'{col}_ctr'] = (grp[f'{col}_clicks'] + alpha * global_ctr) / (
        grp[f'{col}_impr'] + alpha
    )

    # merge into train / test
    train = train.merge(
        grp[[col, f'{col}_impr', f'{col}_ctr']],
        on=col, how='left'
    )
    X_test = X_test.merge(
        grp[[col, f'{col}_impr', f'{col}_ctr']],
        on=col, how='left'
    )

    # handle unseen combos in test
    train[f'{col}_impr'] = train[f'{col}_impr'].fillna(0)
    train[f'{col}_ctr'] = train[f'{col}_ctr'].fillna(global_ctr)
    X_test[f'{col}_impr'] = X_test[f'{col}_impr'].fillna(0)
    X_test[f'{col}_ctr'] = X_test[f'{col}_ctr'].fillna(global_ctr)

# drop temp target and return X versions only
train = train.drop(columns='__target__')
return train, X_test

```

```

[35]: num_scaled = [
    # demographics / behavior
    'age_level',
    'user_depth',
    'city_development_index',
    'var_1',

    # high-cardinality IDs summarized by aggregates
    'user_group_id_impr', 'user_group_id_ctr',
    'age_level_impr', 'age_level_ctr',
    'product_impr', 'product_ctr',
    'campaign_id_impr', 'campaign_id_ctr',

```

```

'webpage_id_impr', 'webpage_id_ctr',
'product_category_1_impr', 'product_category_1_ctr',

# interaction aggregates (if you added them later, include their *_impr /
↳*_ctr here)
'campaign_webpage_ctr',
'campaign_webpage_impr',
'gender_age_ctr',
'gender_age_impr',
'user_product_interaction_ctr',
'user_product_interaction_impr',
'product_age_ctr',
'product_age_impr',
'product_cat_1_age_ctr',
'product_cat_1_age_impr',

# cyclic encodings
'hour_sin', 'hour_cos',
'day_in_week_sin', 'day_in_week_cos',
'day_in_month_sin', 'day_in_month_cos'
]
cat_onehot = [
    'product_category_1',
    'product_category_2',
    'user_group_id',
    'gender',
    'time_of_day'
]

```

```

[38]: X_train_lr = X_train.copy()
      X_val_lr   = X_val.copy()

      X_train_lr, X_val_lr = add_interaction_aggregates(
          X_train_lr, y_train,
          X_val_lr,
          combo_cols=interaction_features,
          alpha=50
      )

      preprocess_lr = ColumnTransformer(
          transformers=[
              ("cat", OneHotEncoder(handle_unknown="ignore"), cat_onehot),
              ("num", "passthrough", num_scaled)
          ]
      )

      pos_weight_lr = (y_train == 0).sum() / (y_train == 1).sum()

```

```

logreg = LogisticRegression(
    class_weight={0: 1.0, 1: pos_weight_lr}
)

pipe_lr = Pipeline(steps=[
    ("preprocess", preprocess_lr),
    ("model", logreg)
])

param_grid_lr = {
    "model__C": [0.001, 0.01, 0.1, 1, 10]
}

gs_lr = GridSearchCV(
    estimator=pipe_lr,
    param_grid=param_grid_lr,
    scoring="roc_auc",
    cv=3,
    verbose=1,
    n_jobs=-1
)

gs_lr.fit(X_train_lr, y_train)

best_lr = gs_lr.best_estimator_
print("Best LogReg C:", gs_lr.best_params_)

y_val_proba_lr = best_lr.predict_proba(X_val_lr)[:, 1]
res_lr = evaluate_model("LogReg_GS", y_val, y_val_proba_lr)

```

Fitting 3 folds for each of 5 candidates, totalling 15 fits

```

/Users/manishdiddi/ml-env/lib/python3.12/site-
packages/sklearn/linear_model/_logistic.py:469: ConvergenceWarning: lbfgs failed
to converge (status=1):
STOP: TOTAL NO. OF ITERATIONS REACHED LIMIT.

```

Increase the number of iterations (max_iter) or scale the data as shown in:

<https://scikit-learn.org/stable/modules/preprocessing.html>

Please also refer to the documentation for alternative solver options:

https://scikit-learn.org/stable/modules/linear_model.html#logistic-regression

```

n_iter_i = _check_optimize_result(
/Users/manishdiddi/ml-env/lib/python3.12/site-
packages/sklearn/linear_model/_logistic.py:469: ConvergenceWarning: lbfgs failed
to converge (status=1):
STOP: TOTAL NO. OF ITERATIONS REACHED LIMIT.

```

Increase the number of iterations (max_iter) or scale the data as shown in:

<https://scikit-learn.org/stable/modules/preprocessing.html>

Please also refer to the documentation for alternative solver options:

https://scikit-learn.org/stable/modules/linear_model.html#logistic-regression

```
n_iter_i = _check_optimize_result(
/Users/manishdiddi/ml-env/lib/python3.12/site-
packages/sklearn/linear_model/_logistic.py:469: ConvergenceWarning: lbfgs failed
to converge (status=1):
STOP: TOTAL NO. OF ITERATIONS REACHED LIMIT.
```

Increase the number of iterations (max_iter) or scale the data as shown in:

<https://scikit-learn.org/stable/modules/preprocessing.html>

Please also refer to the documentation for alternative solver options:

https://scikit-learn.org/stable/modules/linear_model.html#logistic-regression

```
n_iter_i = _check_optimize_result(
/Users/manishdiddi/ml-env/lib/python3.12/site-
packages/sklearn/linear_model/_logistic.py:469: ConvergenceWarning: lbfgs failed
to converge (status=1):
STOP: TOTAL NO. OF ITERATIONS REACHED LIMIT.
```

Increase the number of iterations (max_iter) or scale the data as shown in:

<https://scikit-learn.org/stable/modules/preprocessing.html>

Please also refer to the documentation for alternative solver options:

https://scikit-learn.org/stable/modules/linear_model.html#logistic-regression

```
n_iter_i = _check_optimize_result(
/Users/manishdiddi/ml-env/lib/python3.12/site-
packages/sklearn/linear_model/_logistic.py:469: ConvergenceWarning: lbfgs failed
to converge (status=1):
STOP: TOTAL NO. OF ITERATIONS REACHED LIMIT.
```

Increase the number of iterations (max_iter) or scale the data as shown in:

<https://scikit-learn.org/stable/modules/preprocessing.html>

Please also refer to the documentation for alternative solver options:

https://scikit-learn.org/stable/modules/linear_model.html#logistic-regression

```
n_iter_i = _check_optimize_result(
/Users/manishdiddi/ml-env/lib/python3.12/site-
packages/sklearn/linear_model/_logistic.py:469: ConvergenceWarning: lbfgs failed
to converge (status=1):
STOP: TOTAL NO. OF ITERATIONS REACHED LIMIT.
```

Increase the number of iterations (max_iter) or scale the data as shown in:

<https://scikit-learn.org/stable/modules/preprocessing.html>

Please also refer to the documentation for alternative solver options:

```
https://scikit-learn.org/stable/modules/linear_model.html#logistic-
regression
n_iter_i = _check_optimize_result(
/Users/manishdiddi/ml-env/lib/python3.12/site-
packages/sklearn/linear_model/_logistic.py:469: ConvergenceWarning: lbfgs failed
to converge (status=1):
STOP: TOTAL NO. OF ITERATIONS REACHED LIMIT.
```

Increase the number of iterations (max_iter) or scale the data as shown in:

```
https://scikit-learn.org/stable/modules/preprocessing.html
Please also refer to the documentation for alternative solver options:
https://scikit-learn.org/stable/modules/linear_model.html#logistic-
regression
n_iter_i = _check_optimize_result(
/Users/manishdiddi/ml-env/lib/python3.12/site-
packages/sklearn/linear_model/_logistic.py:469: ConvergenceWarning: lbfgs failed
to converge (status=1):
STOP: TOTAL NO. OF ITERATIONS REACHED LIMIT.
```

Increase the number of iterations (max_iter) or scale the data as shown in:

```
https://scikit-learn.org/stable/modules/preprocessing.html
Please also refer to the documentation for alternative solver options:
https://scikit-learn.org/stable/modules/linear_model.html#logistic-
regression
n_iter_i = _check_optimize_result(
/Users/manishdiddi/ml-env/lib/python3.12/site-
packages/sklearn/linear_model/_logistic.py:469: ConvergenceWarning: lbfgs failed
to converge (status=1):
STOP: TOTAL NO. OF ITERATIONS REACHED LIMIT.
```

Increase the number of iterations (max_iter) or scale the data as shown in:

```
https://scikit-learn.org/stable/modules/preprocessing.html
Please also refer to the documentation for alternative solver options:
https://scikit-learn.org/stable/modules/linear_model.html#logistic-
regression
n_iter_i = _check_optimize_result(
/Users/manishdiddi/ml-env/lib/python3.12/site-
packages/sklearn/linear_model/_logistic.py:469: ConvergenceWarning: lbfgs failed
to converge (status=1):
STOP: TOTAL NO. OF ITERATIONS REACHED LIMIT.
```

Increase the number of iterations (max_iter) or scale the data as shown in:

```
https://scikit-learn.org/stable/modules/preprocessing.html
Please also refer to the documentation for alternative solver options:
https://scikit-learn.org/stable/modules/linear_model.html#logistic-
regression
n_iter_i = _check_optimize_result(
/Users/manishdiddi/ml-env/lib/python3.12/site-
```

```
packages/sklearn/linear_model/_logistic.py:469: ConvergenceWarning: lbfgs failed
to converge (status=1):
STOP: TOTAL NO. OF ITERATIONS REACHED LIMIT.
```

Increase the number of iterations (max_iter) or scale the data as shown in:

<https://scikit-learn.org/stable/modules/preprocessing.html>

Please also refer to the documentation for alternative solver options:

https://scikit-learn.org/stable/modules/linear_model.html#logistic-regression

```
n_iter_i = _check_optimize_result(
/Users/manishdiddi/ml-env/lib/python3.12/site-
packages/sklearn/linear_model/_logistic.py:469: ConvergenceWarning: lbfgs failed
to converge (status=1):
STOP: TOTAL NO. OF ITERATIONS REACHED LIMIT.
```

Increase the number of iterations (max_iter) or scale the data as shown in:

<https://scikit-learn.org/stable/modules/preprocessing.html>

Please also refer to the documentation for alternative solver options:

https://scikit-learn.org/stable/modules/linear_model.html#logistic-regression

```
n_iter_i = _check_optimize_result(
/Users/manishdiddi/ml-env/lib/python3.12/site-
packages/sklearn/linear_model/_logistic.py:469: ConvergenceWarning: lbfgs failed
to converge (status=1):
STOP: TOTAL NO. OF ITERATIONS REACHED LIMIT.
```

Increase the number of iterations (max_iter) or scale the data as shown in:

<https://scikit-learn.org/stable/modules/preprocessing.html>

Please also refer to the documentation for alternative solver options:

https://scikit-learn.org/stable/modules/linear_model.html#logistic-regression

```
n_iter_i = _check_optimize_result(
/Users/manishdiddi/ml-env/lib/python3.12/site-
packages/sklearn/linear_model/_logistic.py:469: ConvergenceWarning: lbfgs failed
to converge (status=1):
STOP: TOTAL NO. OF ITERATIONS REACHED LIMIT.
```

Increase the number of iterations (max_iter) or scale the data as shown in:

<https://scikit-learn.org/stable/modules/preprocessing.html>

Please also refer to the documentation for alternative solver options:

https://scikit-learn.org/stable/modules/linear_model.html#logistic-regression

```
n_iter_i = _check_optimize_result(
```

```
Best LogReg C: {'model__C': 1}
```

```
=== LogReg_GS ===
```

```
AUC: 0.5634
```


LogLoss: 0.7114

Best F1 threshold: 0.523

Best F1: 0.1296

	precision	recall	f1-score	support
0	0.95	0.58	0.72	86952
1	0.07	0.51	0.13	5707
accuracy			0.57	92659
macro avg	0.51	0.55	0.42	92659
weighted avg	0.89	0.57	0.68	92659

Decision Tree Classifier

```
[41]: cat_features = [  
    'product',  
    'campaign_id',  
    'webpage_id',  
    'product_category_1',  
    'product_category_2',  
    'user_group_id',  
    'gender',  
    'time_of_day',  
    'user_product_interaction',  
    'product_age',  
    'product_cat_1_age',  
    'campaign_webpage',  
    'gender_age'  
]
```

```
[42]: X_train_tree = X_train.copy()  
X_val_tree = X_val.copy()  
  
for col in cat_features:  
    X_train_tree[col] = X_train_tree[col].astype('category').cat.codes  
    X_val_tree[col] = X_val_tree[col].astype('category').cat.codes  
  
pos_weight = (y_train == 0).sum() / (y_train == 1).sum()  
  
dt = DecisionTreeClassifier(  
    random_state=42,  
    class_weight={0: 1.0, 1: pos_weight} # handle 6.7% CTR imbalance  
)  
  
param_grid_dt = {  
    "max_depth": [5, 10, None],  
    "min_samples_split": [2, 50, 200],  
}
```

```

    "min_samples_leaf": [1, 20, 100],
    "max_features": [None, "sqrt", 0.5],
}

gs_dt = GridSearchCV(
    estimator=dt,
    param_grid=param_grid_dt,
    scoring="roc_auc",
    cv=3,
    n_jobs=-1,
    verbose=1
)

gs_dt.fit(X_train_tree, y_train)

best_dt = gs_dt.best_estimator_
print("Best DecisionTree params:", gs_dt.best_params_)

y_val_proba_dt = best_dt.predict_proba(X_val_tree)[: , 1]
res_dt = evaluate_model("DecisionTree_GS", y_val, y_val_proba_dt)

```

Fitting 3 folds for each of 81 candidates, totalling 243 fits
 Best DecisionTree params: {'max_depth': 5, 'max_features': 'sqrt',
 'min_samples_leaf': 20, 'min_samples_split': 2}

=== DecisionTree_GS ===

AUC: 0.5379

LogLoss: 0.6557

Best F1 threshold: 0.431

Best F1: 0.1211

	precision	recall	f1-score	support
0	0.95	0.31	0.47	86952
1	0.07	0.74	0.12	5707
accuracy			0.34	92659
macro avg	0.51	0.53	0.29	92659
weighted avg	0.89	0.34	0.44	92659

Random Forest Classifier

```

[44]: rf = RandomForestClassifier(
    n_estimators=500,
    random_state=42,
    n_jobs=-1,
    class_weight={0: 1.0, 1: pos_weight}
)

```

```

param_dist_rf = {
    "max_depth": [None, 10, 20],
    "min_samples_split": [2, 50, 200],
    "min_samples_leaf": [1, 20, 100],
    "max_features": ["sqrt", 0.3, 0.5],
    "bootstrap": [True, False],
}

rs_rf = RandomizedSearchCV(
    estimator=rf,
    param_distributions=param_dist_rf,
    n_iter=20,
    scoring="roc_auc",
    cv=3,
    n_jobs=-1,
    verbose=1,
    random_state=42
)

rs_rf.fit(X_train_tree, y_train)

best_rf = rs_rf.best_estimator_
print("Best RandomForest params:", rs_rf.best_params_)

y_val_proba_rf = best_rf.predict_proba(X_val_tree)[: , 1]
res_rf = evaluate_model("RandomForest_RS", y_val, y_val_proba_rf)

```

Fitting 3 folds for each of 20 candidates, totalling 60 fits
 Best RandomForest params: {'min_samples_split': 200, 'min_samples_leaf': 100, 'max_features': 'sqrt', 'max_depth': 20, 'bootstrap': False}

```

=== RandomForest_RS ===
AUC: 0.5698
LogLoss: 0.5972
Best F1 threshold: 0.473
Best F1: 0.1329

```

	precision	recall	f1-score	support
0	0.95	0.67	0.79	86952
1	0.08	0.42	0.13	5707
accuracy			0.66	92659
macro avg	0.51	0.55	0.46	92659
weighted avg	0.89	0.66	0.75	92659

XGBoost

```

[45]: X_train_xgb = X_train.copy()
      X_val_xgb   = X_val.copy()

      # For XGBoost, convert categoricals to integer codes
      for col in cat_features:
          X_train_xgb[col] = X_train_xgb[col].astype('category').cat.codes
          X_val_xgb[col]   = X_val_xgb[col].astype('category').cat.codes

      pos_weight_xgb = (y_train == 0).sum() / (y_train == 1).sum() #_
      ↪scale_pos_weight[web:129]

      xgb_est = XGBClassifier(
          objective="binary:logistic",
          eval_metric="auc",
          tree_method="hist",
          scale_pos_weight=pos_weight_xgb,
          use_label_encoder=False,
          n_estimators=500,
          random_state=42,
          n_jobs=-1
      )

      param_dist_xgb = {
          "max_depth": [3, 5, 7, 9],
          "learning_rate": [0.01, 0.05, 0.1, 0.2],
          "n_estimators": [200, 300, 500, 800],
          "subsample": [0.6, 0.8, 1.0],
          "colsample_bytree": [0.6, 0.8, 1.0],
          "min_child_weight": [1, 5, 10],
          "gamma": [0, 1]
      }

      rs_xgb = RandomizedSearchCV(
          estimator=xgb_est,
          param_distributions=param_dist_xgb,
          n_iter=20,
          scoring="roc_auc",
          cv=5,
          verbose=1,
          random_state=42,
          n_jobs=-1
      )

      rs_xgb.fit(X_train_xgb, y_train)

      best_xgb = rs_xgb.best_estimator_
      print("Best XGB params:", rs_xgb.best_params_)

```

```
y_val_proba_xgb = best_xgb.predict_proba(X_val_xgb)[: , 1]
res_xgb = evaluate_model("XGBoost_RS", y_val, y_val_proba_xgb)
```

Fitting 5 folds for each of 20 candidates, totalling 100 fits

```
/Users/manishdiddi/ml-env/lib/python3.12/site-packages/xgboost/training.py:199:
UserWarning: [03:43:23] WARNING:
/Users/runner/work/xgboost/xgboost/src/learner.cc:790:
Parameters: { "use_label_encoder" } are not used.
```

```
    bst.update(dtrain, iteration=i, fobj=obj)
/Users/manishdiddi/ml-env/lib/python3.12/site-packages/xgboost/training.py:199:
UserWarning: [03:43:23] WARNING:
/Users/runner/work/xgboost/xgboost/src/learner.cc:790:
Parameters: { "use_label_encoder" } are not used.
```

```
    bst.update(dtrain, iteration=i, fobj=obj)
/Users/manishdiddi/ml-env/lib/python3.12/site-packages/xgboost/training.py:199:
UserWarning: [03:43:23] WARNING:
/Users/runner/work/xgboost/xgboost/src/learner.cc:790:
Parameters: { "use_label_encoder" } are not used.
```

```
    bst.update(dtrain, iteration=i, fobj=obj)
/Users/manishdiddi/ml-env/lib/python3.12/site-packages/xgboost/training.py:199:
UserWarning: [03:43:23] WARNING:
/Users/runner/work/xgboost/xgboost/src/learner.cc:790:
Parameters: { "use_label_encoder" } are not used.
```

```
    bst.update(dtrain, iteration=i, fobj=obj)
/Users/manishdiddi/ml-env/lib/python3.12/site-packages/xgboost/training.py:199:
UserWarning: [03:43:23] WARNING:
/Users/runner/work/xgboost/xgboost/src/learner.cc:790:
Parameters: { "use_label_encoder" } are not used.
```

```
    bst.update(dtrain, iteration=i, fobj=obj)
/Users/manishdiddi/ml-env/lib/python3.12/site-packages/xgboost/training.py:199:
UserWarning: [03:43:24] WARNING:
/Users/runner/work/xgboost/xgboost/src/learner.cc:790:
Parameters: { "use_label_encoder" } are not used.
```

```
    bst.update(dtrain, iteration=i, fobj=obj)
/Users/manishdiddi/ml-env/lib/python3.12/site-packages/xgboost/training.py:199:
UserWarning: [03:43:24] WARNING:
/Users/runner/work/xgboost/xgboost/src/learner.cc:790:
Parameters: { "use_label_encoder" } are not used.
```

```
    bst.update(dtrain, iteration=i, fobj=obj)
```

```
/Users/manishdiddi/ml-env/lib/python3.12/site-packages/xgboost/training.py:199:
UserWarning: [03:43:24] WARNING:
/Users/runner/work/xgboost/xgboost/src/learner.cc:790:
Parameters: { "use_label_encoder" } are not used.
```

```
bst.update(dtrain, iteration=i, fobj=obj)
/Users/manishdiddi/ml-env/lib/python3.12/site-packages/xgboost/training.py:199:
UserWarning: [03:43:24] WARNING:
/Users/runner/work/xgboost/xgboost/src/learner.cc:790:
Parameters: { "use_label_encoder" } are not used.
```

```
bst.update(dtrain, iteration=i, fobj=obj)
/Users/manishdiddi/ml-env/lib/python3.12/site-packages/xgboost/training.py:199:
UserWarning: [03:43:24] WARNING:
/Users/runner/work/xgboost/xgboost/src/learner.cc:790:
Parameters: { "use_label_encoder" } are not used.
```

```
bst.update(dtrain, iteration=i, fobj=obj)
/Users/manishdiddi/ml-env/lib/python3.12/site-packages/xgboost/training.py:199:
UserWarning: [03:43:24] WARNING:
/Users/runner/work/xgboost/xgboost/src/learner.cc:790:
Parameters: { "use_label_encoder" } are not used.
```

```
bst.update(dtrain, iteration=i, fobj=obj)
/Users/manishdiddi/ml-env/lib/python3.12/site-packages/xgboost/training.py:199:
UserWarning: [03:43:24] WARNING:
/Users/runner/work/xgboost/xgboost/src/learner.cc:790:
Parameters: { "use_label_encoder" } are not used.
```

```
bst.update(dtrain, iteration=i, fobj=obj)
/Users/manishdiddi/ml-env/lib/python3.12/site-packages/xgboost/training.py:199:
UserWarning: [03:43:32] WARNING:
/Users/runner/work/xgboost/xgboost/src/learner.cc:790:
Parameters: { "use_label_encoder" } are not used.
```

```
bst.update(dtrain, iteration=i, fobj=obj)
/Users/manishdiddi/ml-env/lib/python3.12/site-packages/xgboost/training.py:199:
UserWarning: [03:43:32] WARNING:
/Users/runner/work/xgboost/xgboost/src/learner.cc:790:
Parameters: { "use_label_encoder" } are not used.
```

```
bst.update(dtrain, iteration=i, fobj=obj)
/Users/manishdiddi/ml-env/lib/python3.12/site-packages/xgboost/training.py:199:
UserWarning: [03:43:32] WARNING:
/Users/runner/work/xgboost/xgboost/src/learner.cc:790:
Parameters: { "use_label_encoder" } are not used.
```

```
bst.update(dtrain, iteration=i, fobj=obj)
```

```
/Users/manishdiddi/ml-env/lib/python3.12/site-packages/xgboost/training.py:199:
UserWarning: [03:43:33] WARNING:
/Users/runner/work/xgboost/xgboost/src/learner.cc:790:
Parameters: { "use_label_encoder" } are not used.
```

```
    bst.update(dtrain, iteration=i, fobj=obj)
/Users/manishdiddi/ml-env/lib/python3.12/site-packages/xgboost/training.py:199:
UserWarning: [03:43:33] WARNING:
/Users/runner/work/xgboost/xgboost/src/learner.cc:790:
Parameters: { "use_label_encoder" } are not used.
```

```
    bst.update(dtrain, iteration=i, fobj=obj)
/Users/manishdiddi/ml-env/lib/python3.12/site-packages/xgboost/training.py:199:
UserWarning: [03:43:33] WARNING:
/Users/runner/work/xgboost/xgboost/src/learner.cc:790:
Parameters: { "use_label_encoder" } are not used.
```

```
    bst.update(dtrain, iteration=i, fobj=obj)
/Users/manishdiddi/ml-env/lib/python3.12/site-packages/xgboost/training.py:199:
UserWarning: [03:43:33] WARNING:
/Users/runner/work/xgboost/xgboost/src/learner.cc:790:
Parameters: { "use_label_encoder" } are not used.
```

```
    bst.update(dtrain, iteration=i, fobj=obj)
/Users/manishdiddi/ml-env/lib/python3.12/site-packages/xgboost/training.py:199:
UserWarning: [03:43:33] WARNING:
/Users/runner/work/xgboost/xgboost/src/learner.cc:790:
Parameters: { "use_label_encoder" } are not used.
```

```
    bst.update(dtrain, iteration=i, fobj=obj)
/Users/manishdiddi/ml-env/lib/python3.12/site-packages/xgboost/training.py:199:
UserWarning: [03:43:33] WARNING:
/Users/runner/work/xgboost/xgboost/src/learner.cc:790:
Parameters: { "use_label_encoder" } are not used.
```

```
    bst.update(dtrain, iteration=i, fobj=obj)
/Users/manishdiddi/ml-env/lib/python3.12/site-packages/xgboost/training.py:199:
UserWarning: [03:43:34] WARNING:
/Users/runner/work/xgboost/xgboost/src/learner.cc:790:
Parameters: { "use_label_encoder" } are not used.
```

```
    bst.update(dtrain, iteration=i, fobj=obj)
/Users/manishdiddi/ml-env/lib/python3.12/site-packages/xgboost/training.py:199:
UserWarning: [03:43:37] WARNING:
/Users/runner/work/xgboost/xgboost/src/learner.cc:790:
Parameters: { "use_label_encoder" } are not used.
```

```
    bst.update(dtrain, iteration=i, fobj=obj)
```

```
/Users/manishdiddi/ml-env/lib/python3.12/site-packages/xgboost/training.py:199:
UserWarning: [03:43:37] WARNING:
/Users/runner/work/xgboost/xgboost/src/learner.cc:790:
Parameters: { "use_label_encoder" } are not used.
```

```
bst.update(dtrain, iteration=i, fobj=obj)
/Users/manishdiddi/ml-env/lib/python3.12/site-packages/xgboost/training.py:199:
UserWarning: [03:43:37] WARNING:
/Users/runner/work/xgboost/xgboost/src/learner.cc:790:
Parameters: { "use_label_encoder" } are not used.
```

```
bst.update(dtrain, iteration=i, fobj=obj)
/Users/manishdiddi/ml-env/lib/python3.12/site-packages/xgboost/training.py:199:
UserWarning: [03:43:37] WARNING:
/Users/runner/work/xgboost/xgboost/src/learner.cc:790:
Parameters: { "use_label_encoder" } are not used.
```

```
bst.update(dtrain, iteration=i, fobj=obj)
/Users/manishdiddi/ml-env/lib/python3.12/site-packages/xgboost/training.py:199:
UserWarning: [03:43:37] WARNING:
/Users/runner/work/xgboost/xgboost/src/learner.cc:790:
Parameters: { "use_label_encoder" } are not used.
```

```
bst.update(dtrain, iteration=i, fobj=obj)
/Users/manishdiddi/ml-env/lib/python3.12/site-packages/xgboost/training.py:199:
UserWarning: [03:43:43] WARNING:
/Users/runner/work/xgboost/xgboost/src/learner.cc:790:
Parameters: { "use_label_encoder" } are not used.
```

```
bst.update(dtrain, iteration=i, fobj=obj)
/Users/manishdiddi/ml-env/lib/python3.12/site-packages/xgboost/training.py:199:
UserWarning: [03:43:43] WARNING:
/Users/runner/work/xgboost/xgboost/src/learner.cc:790:
Parameters: { "use_label_encoder" } are not used.
```

```
bst.update(dtrain, iteration=i, fobj=obj)
/Users/manishdiddi/ml-env/lib/python3.12/site-packages/xgboost/training.py:199:
UserWarning: [03:43:46] WARNING:
/Users/runner/work/xgboost/xgboost/src/learner.cc:790:
Parameters: { "use_label_encoder" } are not used.
```

```
bst.update(dtrain, iteration=i, fobj=obj)
/Users/manishdiddi/ml-env/lib/python3.12/site-packages/xgboost/training.py:199:
UserWarning: [03:43:46] WARNING:
/Users/runner/work/xgboost/xgboost/src/learner.cc:790:
Parameters: { "use_label_encoder" } are not used.
```

```
bst.update(dtrain, iteration=i, fobj=obj)
```



```
/Users/manishdiddi/ml-env/lib/python3.12/site-packages/xgboost/training.py:199:
UserWarning: [03:43:46] WARNING:
/Users/runner/work/xgboost/xgboost/src/learner.cc:790:
Parameters: { "use_label_encoder" } are not used.
```

```
bst.update(dtrain, iteration=i, fobj=obj)
/Users/manishdiddi/ml-env/lib/python3.12/site-packages/xgboost/training.py:199:
UserWarning: [03:43:46] WARNING:
/Users/runner/work/xgboost/xgboost/src/learner.cc:790:
Parameters: { "use_label_encoder" } are not used.
```

```
bst.update(dtrain, iteration=i, fobj=obj)
/Users/manishdiddi/ml-env/lib/python3.12/site-packages/xgboost/training.py:199:
UserWarning: [03:43:46] WARNING:
/Users/runner/work/xgboost/xgboost/src/learner.cc:790:
Parameters: { "use_label_encoder" } are not used.
```

```
bst.update(dtrain, iteration=i, fobj=obj)
/Users/manishdiddi/ml-env/lib/python3.12/site-packages/xgboost/training.py:199:
UserWarning: [03:43:52] WARNING:
/Users/runner/work/xgboost/xgboost/src/learner.cc:790:
Parameters: { "use_label_encoder" } are not used.
```

```
bst.update(dtrain, iteration=i, fobj=obj)
/Users/manishdiddi/ml-env/lib/python3.12/site-packages/xgboost/training.py:199:
UserWarning: [03:43:53] WARNING:
/Users/runner/work/xgboost/xgboost/src/learner.cc:790:
Parameters: { "use_label_encoder" } are not used.
```

```
bst.update(dtrain, iteration=i, fobj=obj)
/Users/manishdiddi/ml-env/lib/python3.12/site-packages/xgboost/training.py:199:
UserWarning: [03:43:54] WARNING:
/Users/runner/work/xgboost/xgboost/src/learner.cc:790:
Parameters: { "use_label_encoder" } are not used.
```

```
bst.update(dtrain, iteration=i, fobj=obj)
/Users/manishdiddi/ml-env/lib/python3.12/site-packages/xgboost/training.py:199:
UserWarning: [03:43:54] WARNING:
/Users/runner/work/xgboost/xgboost/src/learner.cc:790:
Parameters: { "use_label_encoder" } are not used.
```

```
bst.update(dtrain, iteration=i, fobj=obj)
/Users/manishdiddi/ml-env/lib/python3.12/site-packages/xgboost/training.py:199:
UserWarning: [03:43:56] WARNING:
/Users/runner/work/xgboost/xgboost/src/learner.cc:790:
Parameters: { "use_label_encoder" } are not used.
```

```
bst.update(dtrain, iteration=i, fobj=obj)
```

```
/Users/manishdiddi/ml-env/lib/python3.12/site-packages/xgboost/training.py:199:
UserWarning: [03:43:58] WARNING:
/Users/runner/work/xgboost/xgboost/src/learner.cc:790:
Parameters: { "use_label_encoder" } are not used.
```

```
    bst.update(dtrain, iteration=i, fobj=obj)
/Users/manishdiddi/ml-env/lib/python3.12/site-packages/xgboost/training.py:199:
UserWarning: [03:43:58] WARNING:
/Users/runner/work/xgboost/xgboost/src/learner.cc:790:
Parameters: { "use_label_encoder" } are not used.
```

```
    bst.update(dtrain, iteration=i, fobj=obj)
/Users/manishdiddi/ml-env/lib/python3.12/site-packages/xgboost/training.py:199:
UserWarning: [03:43:58] WARNING:
/Users/runner/work/xgboost/xgboost/src/learner.cc:790:
Parameters: { "use_label_encoder" } are not used.
```

```
    bst.update(dtrain, iteration=i, fobj=obj)
/Users/manishdiddi/ml-env/lib/python3.12/site-packages/xgboost/training.py:199:
UserWarning: [03:43:58] WARNING:
/Users/runner/work/xgboost/xgboost/src/learner.cc:790:
Parameters: { "use_label_encoder" } are not used.
```

```
    bst.update(dtrain, iteration=i, fobj=obj)
/Users/manishdiddi/ml-env/lib/python3.12/site-packages/xgboost/training.py:199:
UserWarning: [03:44:02] WARNING:
/Users/runner/work/xgboost/xgboost/src/learner.cc:790:
Parameters: { "use_label_encoder" } are not used.
```

```
    bst.update(dtrain, iteration=i, fobj=obj)
/Users/manishdiddi/ml-env/lib/python3.12/site-packages/xgboost/training.py:199:
UserWarning: [03:44:02] WARNING:
/Users/runner/work/xgboost/xgboost/src/learner.cc:790:
Parameters: { "use_label_encoder" } are not used.
```

```
    bst.update(dtrain, iteration=i, fobj=obj)
/Users/manishdiddi/ml-env/lib/python3.12/site-packages/xgboost/training.py:199:
UserWarning: [03:44:02] WARNING:
/Users/runner/work/xgboost/xgboost/src/learner.cc:790:
Parameters: { "use_label_encoder" } are not used.
```

```
    bst.update(dtrain, iteration=i, fobj=obj)
/Users/manishdiddi/ml-env/lib/python3.12/site-packages/xgboost/training.py:199:
UserWarning: [03:44:04] WARNING:
/Users/runner/work/xgboost/xgboost/src/learner.cc:790:
Parameters: { "use_label_encoder" } are not used.
```

```
    bst.update(dtrain, iteration=i, fobj=obj)
```

```
/Users/manishdiddi/ml-env/lib/python3.12/site-packages/xgboost/training.py:199:
UserWarning: [03:44:07] WARNING:
/Users/runner/work/xgboost/xgboost/src/learner.cc:790:
Parameters: { "use_label_encoder" } are not used.
```

```
    bst.update(dtrain, iteration=i, fobj=obj)
/Users/manishdiddi/ml-env/lib/python3.12/site-packages/xgboost/training.py:199:
UserWarning: [03:44:07] WARNING:
/Users/runner/work/xgboost/xgboost/src/learner.cc:790:
Parameters: { "use_label_encoder" } are not used.
```

```
    bst.update(dtrain, iteration=i, fobj=obj)
/Users/manishdiddi/ml-env/lib/python3.12/site-packages/xgboost/training.py:199:
UserWarning: [03:44:07] WARNING:
/Users/runner/work/xgboost/xgboost/src/learner.cc:790:
Parameters: { "use_label_encoder" } are not used.
```

```
    bst.update(dtrain, iteration=i, fobj=obj)
/Users/manishdiddi/ml-env/lib/python3.12/site-packages/xgboost/training.py:199:
UserWarning: [03:44:09] WARNING:
/Users/runner/work/xgboost/xgboost/src/learner.cc:790:
Parameters: { "use_label_encoder" } are not used.
```

```
    bst.update(dtrain, iteration=i, fobj=obj)
/Users/manishdiddi/ml-env/lib/python3.12/site-packages/xgboost/training.py:199:
UserWarning: [03:44:09] WARNING:
/Users/runner/work/xgboost/xgboost/src/learner.cc:790:
Parameters: { "use_label_encoder" } are not used.
```

```
    bst.update(dtrain, iteration=i, fobj=obj)
/Users/manishdiddi/ml-env/lib/python3.12/site-packages/xgboost/training.py:199:
UserWarning: [03:44:11] WARNING:
/Users/runner/work/xgboost/xgboost/src/learner.cc:790:
Parameters: { "use_label_encoder" } are not used.
```

```
    bst.update(dtrain, iteration=i, fobj=obj)
/Users/manishdiddi/ml-env/lib/python3.12/site-packages/xgboost/training.py:199:
UserWarning: [03:44:11] WARNING:
/Users/runner/work/xgboost/xgboost/src/learner.cc:790:
Parameters: { "use_label_encoder" } are not used.
```

```
    bst.update(dtrain, iteration=i, fobj=obj)
/Users/manishdiddi/ml-env/lib/python3.12/site-packages/xgboost/training.py:199:
UserWarning: [03:44:11] WARNING:
/Users/runner/work/xgboost/xgboost/src/learner.cc:790:
Parameters: { "use_label_encoder" } are not used.
```

```
    bst.update(dtrain, iteration=i, fobj=obj)
```

```
/Users/manishdiddi/ml-env/lib/python3.12/site-packages/xgboost/training.py:199:
UserWarning: [03:44:11] WARNING:
/Users/runner/work/xgboost/xgboost/src/learner.cc:790:
Parameters: { "use_label_encoder" } are not used.
```

```
    bst.update(dtrain, iteration=i, fobj=obj)
/Users/manishdiddi/ml-env/lib/python3.12/site-packages/xgboost/training.py:199:
UserWarning: [03:44:12] WARNING:
/Users/runner/work/xgboost/xgboost/src/learner.cc:790:
Parameters: { "use_label_encoder" } are not used.
```

```
    bst.update(dtrain, iteration=i, fobj=obj)
/Users/manishdiddi/ml-env/lib/python3.12/site-packages/xgboost/training.py:199:
UserWarning: [03:44:14] WARNING:
/Users/runner/work/xgboost/xgboost/src/learner.cc:790:
Parameters: { "use_label_encoder" } are not used.
```

```
    bst.update(dtrain, iteration=i, fobj=obj)
/Users/manishdiddi/ml-env/lib/python3.12/site-packages/xgboost/training.py:199:
UserWarning: [03:44:14] WARNING:
/Users/runner/work/xgboost/xgboost/src/learner.cc:790:
Parameters: { "use_label_encoder" } are not used.
```

```
    bst.update(dtrain, iteration=i, fobj=obj)
/Users/manishdiddi/ml-env/lib/python3.12/site-packages/xgboost/training.py:199:
UserWarning: [03:44:14] WARNING:
/Users/runner/work/xgboost/xgboost/src/learner.cc:790:
Parameters: { "use_label_encoder" } are not used.
```

```
    bst.update(dtrain, iteration=i, fobj=obj)
/Users/manishdiddi/ml-env/lib/python3.12/site-packages/xgboost/training.py:199:
UserWarning: [03:44:14] WARNING:
/Users/runner/work/xgboost/xgboost/src/learner.cc:790:
Parameters: { "use_label_encoder" } are not used.
```

```
    bst.update(dtrain, iteration=i, fobj=obj)
/Users/manishdiddi/ml-env/lib/python3.12/site-packages/xgboost/training.py:199:
UserWarning: [03:44:15] WARNING:
/Users/runner/work/xgboost/xgboost/src/learner.cc:790:
Parameters: { "use_label_encoder" } are not used.
```

```
    bst.update(dtrain, iteration=i, fobj=obj)
/Users/manishdiddi/ml-env/lib/python3.12/site-packages/xgboost/training.py:199:
UserWarning: [03:44:21] WARNING:
/Users/runner/work/xgboost/xgboost/src/learner.cc:790:
Parameters: { "use_label_encoder" } are not used.
```

```
    bst.update(dtrain, iteration=i, fobj=obj)
```

```
/Users/manishdiddi/ml-env/lib/python3.12/site-packages/xgboost/training.py:199:
UserWarning: [03:44:21] WARNING:
/Users/runner/work/xgboost/xgboost/src/learner.cc:790:
Parameters: { "use_label_encoder" } are not used.
```

```
bst.update(dtrain, iteration=i, fobj=obj)
/Users/manishdiddi/ml-env/lib/python3.12/site-packages/xgboost/training.py:199:
UserWarning: [03:44:22] WARNING:
/Users/runner/work/xgboost/xgboost/src/learner.cc:790:
Parameters: { "use_label_encoder" } are not used.
```

```
bst.update(dtrain, iteration=i, fobj=obj)
/Users/manishdiddi/ml-env/lib/python3.12/site-packages/xgboost/training.py:199:
UserWarning: [03:44:22] WARNING:
/Users/runner/work/xgboost/xgboost/src/learner.cc:790:
Parameters: { "use_label_encoder" } are not used.
```

```
bst.update(dtrain, iteration=i, fobj=obj)
/Users/manishdiddi/ml-env/lib/python3.12/site-packages/xgboost/training.py:199:
UserWarning: [03:44:22] WARNING:
/Users/runner/work/xgboost/xgboost/src/learner.cc:790:
Parameters: { "use_label_encoder" } are not used.
```

```
bst.update(dtrain, iteration=i, fobj=obj)
/Users/manishdiddi/ml-env/lib/python3.12/site-packages/xgboost/training.py:199:
UserWarning: [03:44:23] WARNING:
/Users/runner/work/xgboost/xgboost/src/learner.cc:790:
Parameters: { "use_label_encoder" } are not used.
```

```
bst.update(dtrain, iteration=i, fobj=obj)
/Users/manishdiddi/ml-env/lib/python3.12/site-packages/xgboost/training.py:199:
UserWarning: [03:44:23] WARNING:
/Users/runner/work/xgboost/xgboost/src/learner.cc:790:
Parameters: { "use_label_encoder" } are not used.
```

```
bst.update(dtrain, iteration=i, fobj=obj)
/Users/manishdiddi/ml-env/lib/python3.12/site-packages/xgboost/training.py:199:
UserWarning: [03:44:24] WARNING:
/Users/runner/work/xgboost/xgboost/src/learner.cc:790:
Parameters: { "use_label_encoder" } are not used.
```

```
bst.update(dtrain, iteration=i, fobj=obj)
/Users/manishdiddi/ml-env/lib/python3.12/site-packages/xgboost/training.py:199:
UserWarning: [03:44:25] WARNING:
/Users/runner/work/xgboost/xgboost/src/learner.cc:790:
Parameters: { "use_label_encoder" } are not used.
```

```
bst.update(dtrain, iteration=i, fobj=obj)
```

```
/Users/manishdiddi/ml-env/lib/python3.12/site-packages/xgboost/training.py:199:
UserWarning: [03:44:25] WARNING:
/Users/runner/work/xgboost/xgboost/src/learner.cc:790:
Parameters: { "use_label_encoder" } are not used.
```

```
    bst.update(dtrain, iteration=i, fobj=obj)
/Users/manishdiddi/ml-env/lib/python3.12/site-packages/xgboost/training.py:199:
UserWarning: [03:44:25] WARNING:
/Users/runner/work/xgboost/xgboost/src/learner.cc:790:
Parameters: { "use_label_encoder" } are not used.
```

```
    bst.update(dtrain, iteration=i, fobj=obj)
/Users/manishdiddi/ml-env/lib/python3.12/site-packages/xgboost/training.py:199:
UserWarning: [03:44:26] WARNING:
/Users/runner/work/xgboost/xgboost/src/learner.cc:790:
Parameters: { "use_label_encoder" } are not used.
```

```
    bst.update(dtrain, iteration=i, fobj=obj)
/Users/manishdiddi/ml-env/lib/python3.12/site-packages/xgboost/training.py:199:
UserWarning: [03:44:30] WARNING:
/Users/runner/work/xgboost/xgboost/src/learner.cc:790:
Parameters: { "use_label_encoder" } are not used.
```

```
    bst.update(dtrain, iteration=i, fobj=obj)
/Users/manishdiddi/ml-env/lib/python3.12/site-packages/xgboost/training.py:199:
UserWarning: [03:44:30] WARNING:
/Users/runner/work/xgboost/xgboost/src/learner.cc:790:
Parameters: { "use_label_encoder" } are not used.
```

```
    bst.update(dtrain, iteration=i, fobj=obj)
/Users/manishdiddi/ml-env/lib/python3.12/site-packages/xgboost/training.py:199:
UserWarning: [03:44:30] WARNING:
/Users/runner/work/xgboost/xgboost/src/learner.cc:790:
Parameters: { "use_label_encoder" } are not used.
```

```
    bst.update(dtrain, iteration=i, fobj=obj)
/Users/manishdiddi/ml-env/lib/python3.12/site-packages/xgboost/training.py:199:
UserWarning: [03:44:30] WARNING:
/Users/runner/work/xgboost/xgboost/src/learner.cc:790:
Parameters: { "use_label_encoder" } are not used.
```

```
    bst.update(dtrain, iteration=i, fobj=obj)
/Users/manishdiddi/ml-env/lib/python3.12/site-packages/xgboost/training.py:199:
UserWarning: [03:44:31] WARNING:
/Users/runner/work/xgboost/xgboost/src/learner.cc:790:
Parameters: { "use_label_encoder" } are not used.
```

```
    bst.update(dtrain, iteration=i, fobj=obj)
```

```
/Users/manishdiddi/ml-env/lib/python3.12/site-packages/xgboost/training.py:199:
UserWarning: [03:44:31] WARNING:
/Users/runner/work/xgboost/xgboost/src/learner.cc:790:
Parameters: { "use_label_encoder" } are not used.
```

```
    bst.update(dtrain, iteration=i, fobj=obj)
/Users/manishdiddi/ml-env/lib/python3.12/site-packages/xgboost/training.py:199:
UserWarning: [03:44:31] WARNING:
/Users/runner/work/xgboost/xgboost/src/learner.cc:790:
Parameters: { "use_label_encoder" } are not used.
```

```
    bst.update(dtrain, iteration=i, fobj=obj)
/Users/manishdiddi/ml-env/lib/python3.12/site-packages/xgboost/training.py:199:
UserWarning: [03:44:31] WARNING:
/Users/runner/work/xgboost/xgboost/src/learner.cc:790:
Parameters: { "use_label_encoder" } are not used.
```

```
    bst.update(dtrain, iteration=i, fobj=obj)
/Users/manishdiddi/ml-env/lib/python3.12/site-packages/xgboost/training.py:199:
UserWarning: [03:44:31] WARNING:
/Users/runner/work/xgboost/xgboost/src/learner.cc:790:
Parameters: { "use_label_encoder" } are not used.
```

```
    bst.update(dtrain, iteration=i, fobj=obj)
/Users/manishdiddi/ml-env/lib/python3.12/site-packages/xgboost/training.py:199:
UserWarning: [03:44:31] WARNING:
/Users/runner/work/xgboost/xgboost/src/learner.cc:790:
Parameters: { "use_label_encoder" } are not used.
```

```
    bst.update(dtrain, iteration=i, fobj=obj)
/Users/manishdiddi/ml-env/lib/python3.12/site-packages/xgboost/training.py:199:
UserWarning: [03:44:32] WARNING:
/Users/runner/work/xgboost/xgboost/src/learner.cc:790:
Parameters: { "use_label_encoder" } are not used.
```

```
    bst.update(dtrain, iteration=i, fobj=obj)
/Users/manishdiddi/ml-env/lib/python3.12/site-packages/xgboost/training.py:199:
UserWarning: [03:44:32] WARNING:
/Users/runner/work/xgboost/xgboost/src/learner.cc:790:
Parameters: { "use_label_encoder" } are not used.
```

```
    bst.update(dtrain, iteration=i, fobj=obj)
/Users/manishdiddi/ml-env/lib/python3.12/site-packages/xgboost/training.py:199:
UserWarning: [03:44:35] WARNING:
/Users/runner/work/xgboost/xgboost/src/learner.cc:790:
Parameters: { "use_label_encoder" } are not used.
```

```
    bst.update(dtrain, iteration=i, fobj=obj)
```

```
/Users/manishdiddi/ml-env/lib/python3.12/site-packages/xgboost/training.py:199:
UserWarning: [03:44:38] WARNING:
/Users/runner/work/xgboost/xgboost/src/learner.cc:790:
Parameters: { "use_label_encoder" } are not used.
```

```
bst.update(dtrain, iteration=i, fobj=obj)
/Users/manishdiddi/ml-env/lib/python3.12/site-packages/xgboost/training.py:199:
UserWarning: [03:44:38] WARNING:
/Users/runner/work/xgboost/xgboost/src/learner.cc:790:
Parameters: { "use_label_encoder" } are not used.
```

```
bst.update(dtrain, iteration=i, fobj=obj)
/Users/manishdiddi/ml-env/lib/python3.12/site-packages/xgboost/training.py:199:
UserWarning: [03:44:39] WARNING:
/Users/runner/work/xgboost/xgboost/src/learner.cc:790:
Parameters: { "use_label_encoder" } are not used.
```

```
bst.update(dtrain, iteration=i, fobj=obj)
/Users/manishdiddi/ml-env/lib/python3.12/site-packages/xgboost/training.py:199:
UserWarning: [03:44:39] WARNING:
/Users/runner/work/xgboost/xgboost/src/learner.cc:790:
Parameters: { "use_label_encoder" } are not used.
```

```
bst.update(dtrain, iteration=i, fobj=obj)
/Users/manishdiddi/ml-env/lib/python3.12/site-packages/xgboost/training.py:199:
UserWarning: [03:44:39] WARNING:
/Users/runner/work/xgboost/xgboost/src/learner.cc:790:
Parameters: { "use_label_encoder" } are not used.
```

```
bst.update(dtrain, iteration=i, fobj=obj)
/Users/manishdiddi/ml-env/lib/python3.12/site-packages/xgboost/training.py:199:
UserWarning: [03:44:40] WARNING:
/Users/runner/work/xgboost/xgboost/src/learner.cc:790:
Parameters: { "use_label_encoder" } are not used.
```

```
bst.update(dtrain, iteration=i, fobj=obj)
/Users/manishdiddi/ml-env/lib/python3.12/site-packages/xgboost/training.py:199:
UserWarning: [03:44:43] WARNING:
/Users/runner/work/xgboost/xgboost/src/learner.cc:790:
Parameters: { "use_label_encoder" } are not used.
```

```
bst.update(dtrain, iteration=i, fobj=obj)
/Users/manishdiddi/ml-env/lib/python3.12/site-packages/xgboost/training.py:199:
UserWarning: [03:44:45] WARNING:
/Users/runner/work/xgboost/xgboost/src/learner.cc:790:
Parameters: { "use_label_encoder" } are not used.
```

```
bst.update(dtrain, iteration=i, fobj=obj)
```



```

/Users/manishdiddi/ml-env/lib/python3.12/site-packages/xgboost/training.py:199:
UserWarning: [03:44:45] WARNING:
/Users/runner/work/xgboost/xgboost/src/learner.cc:790:
Parameters: { "use_label_encoder" } are not used.

```

```

    bst.update(dtrain, iteration=i, fobj=obj)
/Users/manishdiddi/ml-env/lib/python3.12/site-packages/xgboost/training.py:199:
UserWarning: [03:44:45] WARNING:
/Users/runner/work/xgboost/xgboost/src/learner.cc:790:
Parameters: { "use_label_encoder" } are not used.

```

```

    bst.update(dtrain, iteration=i, fobj=obj)
/Users/manishdiddi/ml-env/lib/python3.12/site-packages/xgboost/training.py:199:
UserWarning: [03:44:46] WARNING:
/Users/runner/work/xgboost/xgboost/src/learner.cc:790:
Parameters: { "use_label_encoder" } are not used.

```

```

    bst.update(dtrain, iteration=i, fobj=obj)
/Users/manishdiddi/ml-env/lib/python3.12/site-packages/xgboost/training.py:199:
UserWarning: [03:44:46] WARNING:
/Users/runner/work/xgboost/xgboost/src/learner.cc:790:
Parameters: { "use_label_encoder" } are not used.

```

```

    bst.update(dtrain, iteration=i, fobj=obj)
/Users/manishdiddi/ml-env/lib/python3.12/site-packages/xgboost/training.py:199:
UserWarning: [03:44:47] WARNING:
/Users/runner/work/xgboost/xgboost/src/learner.cc:790:
Parameters: { "use_label_encoder" } are not used.

```

```

    bst.update(dtrain, iteration=i, fobj=obj)

Best XGB params: {'subsample': 0.6, 'n_estimators': 300, 'min_child_weight': 1,
'max_depth': 3, 'learning_rate': 0.1, 'gamma': 1, 'colsample_bytree': 1.0}

```

```

=== XGBoost_RS ===

```

```

AUC: 0.5587

```

```

LogLoss: 0.6328

```

```

Best F1 threshold: 0.487

```

```

Best F1: 0.1284

```

	precision	recall	f1-score	support
0	0.95	0.62	0.75	86952
1	0.07	0.46	0.13	5707
accuracy			0.61	92659
macro avg	0.51	0.54	0.44	92659
weighted avg	0.89	0.61	0.71	92659

LightGBM

```
[46]: import lightgbm as lgb

X_train_lgb = X_train.copy()
X_val_lgb    = X_val.copy()

for col in cat_features:
    X_train_lgb[col] = X_train_lgb[col].astype('category')
    X_val_lgb[col]    = X_val_lgb[col].astype('category')

pos_weight = (y_train == 0).sum() / (y_train == 1).sum()

lgb_est = lgb.LGBMClassifier(
    objective='binary',
    metric='auc',
    scale_pos_weight=pos_weight,
    n_estimators=1000,
    random_state=42,
    n_jobs=-1
)

param_dist_lgb = {
    "num_leaves": [31, 50, 100, 150],
    "max_depth": [-1, 5, 8, 12],
    "min_child_samples": [20, 50, 100],
    "learning_rate": [0.005, 0.01, 0.05, 0.1],
    "n_estimators": [200, 400, 600, 800],
    "subsample": [0.7, 0.9],
    "colsample_bytree": [0.7, 0.9],
    "feature_fraction": [0.6, 0.8, 1.0],
    "bagging_fraction": [0.6, 0.8, 1.0],
    "bagging_freq": [1, 3, 5]
}

rs_lgb = RandomizedSearchCV(
    estimator=lgb_est,
    param_distributions=param_dist_lgb,
    n_iter=20,
    scoring="roc_auc",
    cv=5,
    verbose=1,
    random_state=42,
    n_jobs=-1
)

rs_lgb.fit(X_train_lgb, y_train, categorical_feature=cat_features)
```

```

best_lgb = rs_lgb.best_estimator_
print("Best LGB params:", rs_lgb.best_params_)

y_val_proba_lgb = best_lgb.predict_proba(X_val_lgb)[: , 1]
res_lgb = evaluate_model("LightGBM_RS", y_val, y_val_proba_lgb)

```

```

Fitting 5 folds for each of 20 candidates, totalling 100 fits
[LightGBM] [Warning] bagging_freq is set=3, subsample_freq=0 will be ignored.
Current value: bagging_freq=3
[LightGBM] [Warning] bagging_fraction is set=0.6, subsample=0.9 will be ignored.
Current value: bagging_fraction=0.6
[LightGBM] [Warning] feature_fraction is set=0.8, colsample_bytree=0.9 will be
ignored. Current value: feature_fraction=0.8
[LightGBM] [Warning] bagging_freq is set=3, subsample_freq=0 will be ignored.
Current value: bagging_freq=3
[LightGBM] [Warning] bagging_fraction is set=0.6, subsample=0.9 will be ignored.
Current value: bagging_fraction=0.6
[LightGBM] [Warning] feature_fraction is set=0.8, colsample_bytree=0.9 will be
ignored. Current value: feature_fraction=0.8
[LightGBM] [Warning] bagging_freq is set=3, subsample_freq=0 will be ignored.
Current value: bagging_freq=3
[LightGBM] [Warning] bagging_fraction is set=0.6, subsample=0.9 will be ignored.
Current value: bagging_fraction=0.6
[LightGBM] [Warning] feature_fraction is set=0.8, colsample_bytree=0.9 will be
ignored. Current value: feature_fraction=0.8
[LightGBM] [Warning] Categorical features with more bins than the configured
maximum bin number found.
[LightGBM] [Warning] For categorical features, max_bin and max_bin_by_feature
may be ignored with a large number of categories.
[LightGBM] [Warning] Categorical features with more bins than the configured
maximum bin number found.
[LightGBM] [Warning] For categorical features, max_bin and max_bin_by_feature
may be ignored with a large number of categories.
[LightGBM] [Warning] bagging_freq is set=3, subsample_freq=0 will be ignored.
Current value: bagging_freq=3
[LightGBM] [Warning] bagging_fraction is set=0.6, subsample=0.9 will be ignored.
Current value: bagging_fraction=0.6
[LightGBM] [Warning] feature_fraction is set=0.8, colsample_bytree=0.9 will be
ignored. Current value: feature_fraction=0.8
[LightGBM] [Warning] Categorical features with more bins than the configured
maximum bin number found.
[LightGBM] [Warning] For categorical features, max_bin and max_bin_by_feature
may be ignored with a large number of categories.
[LightGBM] [Warning] bagging_freq is set=3, subsample_freq=0 will be ignored.
Current value: bagging_freq=3
[LightGBM] [Warning] bagging_fraction is set=0.6, subsample=0.9 will be ignored.
Current value: bagging_fraction=0.6
[LightGBM] [Warning] feature_fraction is set=0.8, colsample_bytree=0.9 will be

```

ignored. Current value: feature_fraction=0.8
[LightGBM] [Warning] bagging_freq is set=3, subsample_freq=0 will be ignored.
Current value: bagging_freq=3
[LightGBM] [Warning] bagging_fraction is set=0.6, subsample=0.9 will be ignored.
Current value: bagging_fraction=0.6
[LightGBM] [Warning] feature_fraction is set=0.8, colsample_bytree=0.9 will be ignored. Current value: feature_fraction=0.8
[LightGBM] [Info] Number of positive: 20499, number of negative: 276006
[LightGBM] [Warning] bagging_freq is set=3, subsample_freq=0 will be ignored.
Current value: bagging_freq=3
[LightGBM] [Warning] bagging_fraction is set=0.6, subsample=0.9 will be ignored.
Current value: bagging_fraction=0.6
[LightGBM] [Warning] feature_fraction is set=0.8, colsample_bytree=0.9 will be ignored. Current value: feature_fraction=0.8
[LightGBM] [Info] Number of positive: 20499, number of negative: 276006
[LightGBM] [Info] Auto-choosing row-wise multi-threading, the overhead of testing was 0.041338 seconds.
You can set `force_row_wise=true` to remove the overhead.
And if memory is not enough, you can set `force_col_wise=true`.
[LightGBM] [Info] Total Bins 13211
[LightGBM] [Info] Auto-choosing row-wise multi-threading, the overhead of testing was 0.028154 seconds.
You can set `force_row_wise=true` to remove the overhead.
And if memory is not enough, you can set `force_col_wise=true`.
[LightGBM] [Info] Total Bins 13359
[LightGBM] [Info] Number of data points in the train set: 296505, number of used features: 35
[LightGBM] [Info] Number of data points in the train set: 296505, number of used features: 35
[LightGBM] [Info] [binary:BoostFromScore]: pavg=0.069135 -> initscore=-2.600047
[LightGBM] [Info] [binary:BoostFromScore]: pavg=0.069135 -> initscore=-2.600047
[LightGBM] [Info] Start training from score -2.600047
[LightGBM] [Info] Start training from score -2.600047
[LightGBM] [Warning] Categorical features with more bins than the configured maximum bin number found.
[LightGBM] [Warning] For categorical features, max_bin and max_bin_by_feature may be ignored with a large number of categories.
[LightGBM] [Warning] bagging_freq is set=1, subsample_freq=0 will be ignored.
Current value: bagging_freq=1
[LightGBM] [Warning] bagging_fraction is set=0.6, subsample=0.7 will be ignored.
Current value: bagging_fraction=0.6
[LightGBM] [Warning] feature_fraction is set=0.6, colsample_bytree=0.7 will be ignored. Current value: feature_fraction=0.6
[LightGBM] [Warning] bagging_freq is set=3, subsample_freq=0 will be ignored.
Current value: bagging_freq=3
[LightGBM] [Warning] bagging_fraction is set=0.6, subsample=0.9 will be ignored.
Current value: bagging_fraction=0.6
[LightGBM] [Warning] feature_fraction is set=0.8, colsample_bytree=0.9 will be

ignored. Current value: feature_fraction=0.8

[LightGBM] [Info] Number of positive: 20500, number of negative: 276006

[LightGBM] [Info] Auto-choosing row-wise multi-threading, the overhead of testing was 0.011476 seconds.

You can set `force_row_wise=true` to remove the overhead.

And if memory is not enough, you can set `force_col_wise=true`.

[LightGBM] [Info] Total Bins 13717

[LightGBM] [Info] Number of data points in the train set: 296506, number of used features: 35

[LightGBM] [Info] [binary:BoostFromScore]: pavg=0.069139 -> initscore=-2.599998

[LightGBM] [Info] Start training from score -2.599998

[LightGBM] [Warning] Categorical features with more bins than the configured maximum bin number found.

[LightGBM] [Warning] For categorical features, max_bin and max_bin_by_feature may be ignored with a large number of categories.

[LightGBM] [Warning] bagging_freq is set=1, subsample_freq=0 will be ignored.

Current value: bagging_freq=1

[LightGBM] [Warning] bagging_fraction is set=0.6, subsample=0.7 will be ignored.

Current value: bagging_fraction=0.6

[LightGBM] [Warning] feature_fraction is set=0.6, colsample_bytree=0.7 will be ignored. Current value: feature_fraction=0.6

[LightGBM] [Warning] Categorical features with more bins than the configured maximum bin number found.

[LightGBM] [Warning] For categorical features, max_bin and max_bin_by_feature may be ignored with a large number of categories.

[LightGBM] [Warning] bagging_freq is set=1, subsample_freq=0 will be ignored.

Current value: bagging_freq=1

[LightGBM] [Warning] bagging_fraction is set=0.6, subsample=0.7 will be ignored.

Current value: bagging_fraction=0.6

[LightGBM] [Warning] feature_fraction is set=0.6, colsample_bytree=0.7 will be ignored. Current value: feature_fraction=0.6

[LightGBM] [Warning] bagging_freq is set=3, subsample_freq=0 will be ignored.

Current value: bagging_freq=3

[LightGBM] [Warning] bagging_fraction is set=0.6, subsample=0.9 will be ignored.

Current value: bagging_fraction=0.6

[LightGBM] [Warning] feature_fraction is set=0.8, colsample_bytree=0.9 will be ignored. Current value: feature_fraction=0.8

[LightGBM] [Info] Number of positive: 20499, number of negative: 276007

[LightGBM] [Info] Auto-choosing row-wise multi-threading, the overhead of testing was 0.007800 seconds.

You can set `force_row_wise=true` to remove the overhead.

And if memory is not enough, you can set `force_col_wise=true`.

[LightGBM] [Info] Total Bins 13726

[LightGBM] [Info] Number of data points in the train set: 296506, number of used features: 35

[LightGBM] [Info] [binary:BoostFromScore]: pavg=0.069135 -> initscore=-2.600050

[LightGBM] [Info] Start training from score -2.600050

[LightGBM] [Warning] bagging_freq is set=3, subsample_freq=0 will be ignored.

```

Current value: bagging_freq=3
[LightGBM] [Warning] bagging_fraction is set=0.6, subsample=0.9 will be ignored.
Current value: bagging_fraction=0.6
[LightGBM] [Warning] feature_fraction is set=0.8, colsample_bytree=0.9 will be
ignored. Current value: feature_fraction=0.8
[LightGBM] [Info] Number of positive: 20499, number of negative: 276007
[LightGBM] [Info] Auto-choosing row-wise multi-threading, the overhead of
testing was 0.034372 seconds.
You can set `force_row_wise=true` to remove the overhead.
And if memory is not enough, you can set `force_col_wise=true`.
[LightGBM] [Info] Total Bins 13769
[LightGBM] [Info] Number of data points in the train set: 296506, number of used
features: 35
[LightGBM] [Info] [binary:BoostFromScore]: pavg=0.069135 -> initscore=-2.600050
[LightGBM] [Info] Start training from score -2.600050
[LightGBM] [Warning] bagging_freq is set=1, subsample_freq=0 will be ignored.
Current value: bagging_freq=1
[LightGBM] [Warning] bagging_fraction is set=0.6, subsample=0.7 will be ignored.
Current value: bagging_fraction=0.6
[LightGBM] [Warning] feature_fraction is set=0.6, colsample_bytree=0.7 will be
ignored. Current value: feature_fraction=0.6
[LightGBM] [Warning] Categorical features with more bins than the configured
maximum bin number found.
[LightGBM] [Warning] For categorical features, max_bin and max_bin_by_feature
may be ignored with a large number of categories.
[LightGBM] [Warning] bagging_freq is set=1, subsample_freq=0 will be ignored.
Current value: bagging_freq=1
[LightGBM] [Warning] bagging_fraction is set=0.6, subsample=0.7 will be ignored.
Current value: bagging_fraction=0.6
[LightGBM] [Warning] feature_fraction is set=0.6, colsample_bytree=0.7 will be
ignored. Current value: feature_fraction=0.6
[LightGBM] [Info] Number of positive: 20499, number of negative: 276006
[LightGBM] [Warning] Categorical features with more bins than the configured
maximum bin number found.
[LightGBM] [Warning] For categorical features, max_bin and max_bin_by_feature
may be ignored with a large number of categories.
[LightGBM] [Info] Auto-choosing row-wise multi-threading, the overhead of
testing was 0.031591 seconds.
You can set `force_row_wise=true` to remove the overhead.
And if memory is not enough, you can set `force_col_wise=true`.
[LightGBM] [Info] Total Bins 13211
[LightGBM] [Info] Number of data points in the train set: 296505, number of used
features: 35
[LightGBM] [Info] [binary:BoostFromScore]: pavg=0.069135 -> initscore=-2.600047
[LightGBM] [Info] Start training from score -2.600047
[LightGBM] [Warning] bagging_freq is set=1, subsample_freq=0 will be ignored.
Current value: bagging_freq=1
[LightGBM] [Warning] bagging_fraction is set=0.6, subsample=0.7 will be ignored.

```

Current value: bagging_fraction=0.6
[LightGBM] [Warning] feature_fraction is set=0.6, colsample_bytree=0.7 will be ignored. Current value: feature_fraction=0.6
[LightGBM] [Info] Number of positive: 20500, number of negative: 276006
[LightGBM] [Warning] bagging_freq is set=1, subsample_freq=0 will be ignored.
Current value: bagging_freq=1
[LightGBM] [Warning] bagging_fraction is set=0.6, subsample=0.7 will be ignored.
Current value: bagging_fraction=0.6
[LightGBM] [Warning] feature_fraction is set=0.6, colsample_bytree=0.7 will be ignored. Current value: feature_fraction=0.6
[LightGBM] [Info] Number of positive: 20499, number of negative: 276006
[LightGBM] [Warning] bagging_freq is set=1, subsample_freq=0 will be ignored.
Current value: bagging_freq=1
[LightGBM] [Warning] bagging_fraction is set=0.6, subsample=0.7 will be ignored.
Current value: bagging_fraction=0.6
[LightGBM] [Warning] feature_fraction is set=0.6, colsample_bytree=0.7 will be ignored. Current value: feature_fraction=0.6
[LightGBM] [Info] Auto-choosing row-wise multi-threading, the overhead of testing was 0.046281 seconds.
You can set `force_row_wise=true` to remove the overhead.
And if memory is not enough, you can set `force_col_wise=true`.
[LightGBM] [Info] Total Bins 13717
[LightGBM] [Info] Number of data points in the train set: 296506, number of used features: 35
[LightGBM] [Warning] Categorical features with more bins than the configured maximum bin number found.
[LightGBM] [Warning] For categorical features, max_bin and max_bin_by_feature may be ignored with a large number of categories.
[LightGBM] [Info] Auto-choosing col-wise multi-threading, the overhead of testing was 0.056925 seconds.
You can set `force_col_wise=true` to remove the overhead.
[LightGBM] [Info] [binary:BoostFromScore]: pavg=0.069139 -> initscore=-2.599998
[LightGBM] [Info] Total Bins 13359

[LightGBM] [Info] Number of data points in the train set: 296505, number of used features: 35
[LightGBM] [Info] Start training from score -2.599998
[LightGBM] [Info] [binary:BoostFromScore]: pavg=0.069135 -> initscore=-2.600047
[LightGBM] [Info] Start training from score -2.600047
[LightGBM] [Warning] bagging_freq is set=5, subsample_freq=0 will be ignored.
Current value: bagging_freq=5
[LightGBM] [Warning] bagging_fraction is set=1.0, subsample=0.7 will be ignored.
Current value: bagging_fraction=1.0
[LightGBM] [Warning] feature_fraction is set=1.0, colsample_bytree=0.7 will be ignored. Current value: feature_fraction=1.0
[LightGBM] [Warning] bagging_freq is set=1, subsample_freq=0 will be ignored.
Current value: bagging_freq=1
[LightGBM] [Warning] bagging_fraction is set=0.6, subsample=0.7 will be ignored.

Current value: bagging_fraction=0.6

[LightGBM] [Warning] feature_fraction is set=0.6, colsample_bytree=0.7 will be ignored. Current value: feature_fraction=0.6

[LightGBM] [Info] Number of positive: 20499, number of negative: 276007

[LightGBM] [Warning] Categorical features with more bins than the configured maximum bin number found.

[LightGBM] [Warning] For categorical features, max_bin and max_bin_by_feature may be ignored with a large number of categories.

[LightGBM] [Info] Auto-choosing col-wise multi-threading, the overhead of testing was 0.076544 seconds.

You can set `force_col_wise=true` to remove the overhead.

[LightGBM] [Info] Total Bins 13726

[LightGBM] [Info] Number of data points in the train set: 296506, number of used features: 35

[LightGBM] [Info] [binary:BoostFromScore]: pavg=0.069135 -> initscore=-2.600050

[LightGBM] [Info] Start training from score -2.600050

[LightGBM] [Warning] Categorical features with more bins than the configured maximum bin number found.

[LightGBM] [Warning] For categorical features, max_bin and max_bin_by_feature may be ignored with a large number of categories.

[LightGBM] [Warning] bagging_freq is set=5, subsample_freq=0 will be ignored.

Current value: bagging_freq=5

[LightGBM] [Warning] bagging_fraction is set=1.0, subsample=0.7 will be ignored.

Current value: bagging_fraction=1.0

[LightGBM] [Warning] feature_fraction is set=1.0, colsample_bytree=0.7 will be ignored. Current value: feature_fraction=1.0

[LightGBM] [Warning] bagging_freq is set=1, subsample_freq=0 will be ignored.

Current value: bagging_freq=1

[LightGBM] [Warning] bagging_fraction is set=0.6, subsample=0.7 will be ignored.

Current value: bagging_fraction=0.6

[LightGBM] [Warning] feature_fraction is set=0.6, colsample_bytree=0.7 will be ignored. Current value: feature_fraction=0.6

[LightGBM] [Info] Number of positive: 20499, number of negative: 276007

[LightGBM] [Info] Auto-choosing col-wise multi-threading, the overhead of testing was 0.113844 seconds.

You can set `force_col_wise=true` to remove the overhead.

[LightGBM] [Info] Total Bins 13769

[LightGBM] [Info] Number of data points in the train set: 296506, number of used features: 35

[LightGBM] [Info] [binary:BoostFromScore]: pavg=0.069135 -> initscore=-2.600050

[LightGBM] [Info] Start training from score -2.600050

[LightGBM] [Warning] bagging_freq is set=5, subsample_freq=0 will be ignored.

Current value: bagging_freq=5

[LightGBM] [Warning] bagging_fraction is set=1.0, subsample=0.7 will be ignored.

Current value: bagging_fraction=1.0

[LightGBM] [Warning] feature_fraction is set=1.0, colsample_bytree=0.7 will be ignored. Current value: feature_fraction=1.0

[LightGBM] [Info] Number of positive: 20499, number of negative: 276006


```

[LightGBM] [Info] Auto-choosing row-wise multi-threading, the overhead of
testing was 0.046327 seconds.
You can set `force_row_wise=true` to remove the overhead.
And if memory is not enough, you can set `force_col_wise=true`.
[LightGBM] [Info] Total Bins 13211
[LightGBM] [Info] Number of data points in the train set: 296505, number of used
features: 35
[LightGBM] [Info] [binary:BoostFromScore]: pavg=0.069135 -> initscore=-2.600047
[LightGBM] [Info] Start training from score -2.600047
[LightGBM] [Warning] Categorical features with more bins than the configured
maximum bin number found.
[LightGBM] [Warning] For categorical features, max_bin and max_bin_by_feature
may be ignored with a large number of categories.
[LightGBM] [Warning] bagging_freq is set=5, subsample_freq=0 will be ignored.
Current value: bagging_freq=5
[LightGBM] [Warning] bagging_fraction is set=1.0, subsample=0.7 will be ignored.
Current value: bagging_fraction=1.0
[LightGBM] [Warning] feature_fraction is set=1.0, colsample_bytree=0.7 will be
ignored. Current value: feature_fraction=1.0
[LightGBM] [Info] Number of positive: 20499, number of negative: 276006
[LightGBM] [Info] Auto-choosing row-wise multi-threading, the overhead of
testing was 0.008631 seconds.
You can set `force_row_wise=true` to remove the overhead.
And if memory is not enough, you can set `force_col_wise=true`.
[LightGBM] [Info] Total Bins 13359
[LightGBM] [Info] Number of data points in the train set: 296505, number of used
features: 35
[LightGBM] [Info] [binary:BoostFromScore]: pavg=0.069135 -> initscore=-2.600047
[LightGBM] [Info] Start training from score -2.600047
[LightGBM] [Warning] bagging_freq is set=3, subsample_freq=0 will be ignored.
Current value: bagging_freq=3
[LightGBM] [Warning] feature_fraction is set=0.8, colsample_bytree=0.9 will be
ignored. Current value: feature_fraction=0.8
[LightGBM] [Warning] bagging_fraction is set=0.6, subsample=0.9 will be ignored.
Current value: bagging_fraction=0.6
[LightGBM] [Warning] bagging_freq is set=3, subsample_freq=0 will be ignored.
Current value: bagging_freq=3
[LightGBM] [Warning] feature_fraction is set=0.8, colsample_bytree=0.9 will be
ignored. Current value: feature_fraction=0.8
[LightGBM] [Warning] bagging_fraction is set=0.6, subsample=0.9 will be ignored.
Current value: bagging_fraction=0.6
[LightGBM] [Warning] bagging_freq is set=3, subsample_freq=0 will be ignored.
Current value: bagging_freq=3
[LightGBM] [Warning] feature_fraction is set=0.8, colsample_bytree=0.9 will be
ignored. Current value: feature_fraction=0.8
[LightGBM] [Warning] bagging_fraction is set=0.6, subsample=0.9 will be ignored.
Current value: bagging_fraction=0.6
[LightGBM] [Warning] bagging_freq is set=3, subsample_freq=0 will be ignored.

```

Current value: bagging_freq=3
[LightGBM] [Warning] feature_fraction is set=0.8, colsample_bytree=0.9 will be ignored. Current value: feature_fraction=0.8
[LightGBM] [Warning] bagging_fraction is set=0.6, subsample=0.9 will be ignored. Current value: bagging_fraction=0.6
[LightGBM] [Warning] bagging_freq is set=3, subsample_freq=0 will be ignored. Current value: bagging_freq=3
[LightGBM] [Warning] feature_fraction is set=0.8, colsample_bytree=0.9 will be ignored. Current value: feature_fraction=0.8
[LightGBM] [Warning] bagging_fraction is set=0.6, subsample=0.9 will be ignored. Current value: bagging_fraction=0.6
[LightGBM] [Warning] bagging_freq is set=5, subsample_freq=0 will be ignored. Current value: bagging_freq=5
[LightGBM] [Warning] bagging_fraction is set=1.0, subsample=0.7 will be ignored. Current value: bagging_fraction=1.0
[LightGBM] [Warning] feature_fraction is set=1.0, colsample_bytree=0.7 will be ignored. Current value: feature_fraction=1.0
[LightGBM] [Warning] Categorical features with more bins than the configured maximum bin number found.
[LightGBM] [Warning] For categorical features, max_bin and max_bin_by_feature may be ignored with a large number of categories.
[LightGBM] [Warning] bagging_freq is set=5, subsample_freq=0 will be ignored. Current value: bagging_freq=5
[LightGBM] [Warning] bagging_fraction is set=1.0, subsample=0.7 will be ignored. Current value: bagging_fraction=1.0
[LightGBM] [Warning] feature_fraction is set=1.0, colsample_bytree=0.7 will be ignored. Current value: feature_fraction=1.0
[LightGBM] [Info] Number of positive: 20500, number of negative: 276006
[LightGBM] [Info] Auto-choosing row-wise multi-threading, the overhead of testing was 0.128217 seconds.
You can set `force_row_wise=true` to remove the overhead.
And if memory is not enough, you can set `force_col_wise=true`.
[LightGBM] [Info] Total Bins 13717
[LightGBM] [Info] Number of data points in the train set: 296506, number of used features: 35
[LightGBM] [Info] [binary:BoostFromScore]: pavg=0.069139 -> initscore=-2.599998
[LightGBM] [Info] Start training from score -2.599998
[LightGBM] [Warning] bagging_freq is set=5, subsample_freq=0 will be ignored. Current value: bagging_freq=5
[LightGBM] [Warning] bagging_fraction is set=1.0, subsample=0.7 will be ignored. Current value: bagging_fraction=1.0
[LightGBM] [Warning] feature_fraction is set=1.0, colsample_bytree=0.7 will be ignored. Current value: feature_fraction=1.0
[LightGBM] [Warning] Categorical features with more bins than the configured maximum bin number found.
[LightGBM] [Warning] For categorical features, max_bin and max_bin_by_feature may be ignored with a large number of categories.
[LightGBM] [Warning] bagging_freq is set=5, subsample_freq=0 will be ignored.

```

Current value: bagging_freq=5
[LightGBM] [Warning] bagging_fraction is set=1.0, subsample=0.7 will be ignored.
Current value: bagging_fraction=1.0
[LightGBM] [Warning] feature_fraction is set=1.0, colsample_bytree=0.7 will be
ignored. Current value: feature_fraction=1.0
[LightGBM] [Info] Number of positive: 20499, number of negative: 276007
[LightGBM] [Info] Auto-choosing row-wise multi-threading, the overhead of
testing was 0.014515 seconds.
You can set `force_row_wise=true` to remove the overhead.
And if memory is not enough, you can set `force_col_wise=true`.
[LightGBM] [Info] Total Bins 13726
[LightGBM] [Info] Number of data points in the train set: 296506, number of used
features: 35
[LightGBM] [Info] [binary:BoostFromScore]: pavg=0.069135 -> initscore=-2.600050
[LightGBM] [Info] Start training from score -2.600050
[LightGBM] [Warning] bagging_freq is set=5, subsample_freq=0 will be ignored.
Current value: bagging_freq=5
[LightGBM] [Warning] bagging_fraction is set=1.0, subsample=0.7 will be ignored.
Current value: bagging_fraction=1.0
[LightGBM] [Warning] feature_fraction is set=1.0, colsample_bytree=0.7 will be
ignored. Current value: feature_fraction=1.0
[LightGBM] [Warning] bagging_freq is set=5, subsample_freq=0 will be ignored.
Current value: bagging_freq=5
[LightGBM] [Warning] bagging_fraction is set=0.8, subsample=0.7 will be ignored.
Current value: bagging_fraction=0.8
[LightGBM] [Warning] feature_fraction is set=1.0, colsample_bytree=0.9 will be
ignored. Current value: feature_fraction=1.0
[LightGBM] [Warning] Categorical features with more bins than the configured
maximum bin number found.
[LightGBM] [Warning] For categorical features, max_bin and max_bin_by_feature
may be ignored with a large number of categories.
[LightGBM] [Warning] bagging_freq is set=5, subsample_freq=0 will be ignored.
Current value: bagging_freq=5
[LightGBM] [Warning] bagging_fraction is set=1.0, subsample=0.7 will be ignored.
Current value: bagging_fraction=1.0
[LightGBM] [Warning] feature_fraction is set=1.0, colsample_bytree=0.7 will be
ignored. Current value: feature_fraction=1.0
[LightGBM] [Info] Number of positive: 20499, number of negative: 276007
[LightGBM] [Warning] Categorical features with more bins than the configured
maximum bin number found.
[LightGBM] [Warning] For categorical features, max_bin and max_bin_by_feature
may be ignored with a large number of categories.
[LightGBM] [Info] Auto-choosing row-wise multi-threading, the overhead of
testing was 0.029652 seconds.
You can set `force_row_wise=true` to remove the overhead.
And if memory is not enough, you can set `force_col_wise=true`.
[LightGBM] [Info] Total Bins 13769
[LightGBM] [Info] Number of data points in the train set: 296506, number of used

```

```

features: 35
[LightGBM] [Info] [binary:BoostFromScore]: pavg=0.069135 -> initscore=-2.600050
[LightGBM] [Info] Start training from score -2.600050
[LightGBM] [Warning] bagging_freq is set=5, subsample_freq=0 will be ignored.
Current value: bagging_freq=5
[LightGBM] [Warning] bagging_fraction is set=0.8, subsample=0.7 will be ignored.
Current value: bagging_fraction=0.8
[LightGBM] [Warning] feature_fraction is set=1.0, colsample_bytree=0.9 will be
ignored. Current value: feature_fraction=1.0
[LightGBM] [Warning] bagging_freq is set=5, subsample_freq=0 will be ignored.
Current value: bagging_freq=5
[LightGBM] [Warning] bagging_fraction is set=0.8, subsample=0.7 will be ignored.
Current value: bagging_fraction=0.8
[LightGBM] [Warning] feature_fraction is set=1.0, colsample_bytree=0.9 will be
ignored. Current value: feature_fraction=1.0
[LightGBM] [Info] Number of positive: 20499, number of negative: 276006
[LightGBM] [Info] Auto-choosing row-wise multi-threading, the overhead of
testing was 0.008442 seconds.
You can set `force_row_wise=true` to remove the overhead.
And if memory is not enough, you can set `force_col_wise=true`.
[LightGBM] [Info] Total Bins 13211
[LightGBM] [Info] Number of data points in the train set: 296505, number of used
features: 35
[LightGBM] [Info] [binary:BoostFromScore]: pavg=0.069135 -> initscore=-2.600047
[LightGBM] [Info] Start training from score -2.600047
[LightGBM] [Warning] Categorical features with more bins than the configured
maximum bin number found.
[LightGBM] [Warning] For categorical features, max_bin and max_bin_by_feature
may be ignored with a large number of categories.
[LightGBM] [Warning] bagging_freq is set=5, subsample_freq=0 will be ignored.
Current value: bagging_freq=5
[LightGBM] [Warning] bagging_fraction is set=0.8, subsample=0.7 will be ignored.
Current value: bagging_fraction=0.8
[LightGBM] [Warning] feature_fraction is set=1.0, colsample_bytree=0.9 will be
ignored. Current value: feature_fraction=1.0
[LightGBM] [Info] Number of positive: 20499, number of negative: 276006
[LightGBM] [Info] Auto-choosing row-wise multi-threading, the overhead of
testing was 0.007479 seconds.
You can set `force_row_wise=true` to remove the overhead.
And if memory is not enough, you can set `force_col_wise=true`.
[LightGBM] [Info] Total Bins 13359
[LightGBM] [Info] Number of data points in the train set: 296505, number of used
features: 35
[LightGBM] [Info] [binary:BoostFromScore]: pavg=0.069135 -> initscore=-2.600047
[LightGBM] [Info] Start training from score -2.600047
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

```

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]


```

[LightGBM] [Warning] feature_fraction is set=1.0, colsample_bytree=0.9 will be
ignored. Current value: feature_fraction=1.0
[LightGBM] [Warning] Categorical features with more bins than the configured
maximum bin number found.
[LightGBM] [Warning] For categorical features, max_bin and max_bin_by_feature
may be ignored with a large number of categories.
[LightGBM] [Warning] bagging_freq is set=5, subsample_freq=0 will be ignored.
Current value: bagging_freq=5
[LightGBM] [Warning] feature_fraction is set=1.0, colsample_bytree=0.9 will be
ignored. Current value: feature_fraction=1.0
[LightGBM] [Warning] bagging_fraction is set=0.8, subsample=0.7 will be ignored.
Current value: bagging_fraction=0.8
[LightGBM] [Warning] bagging_freq is set=5, subsample_freq=0 will be ignored.
Current value: bagging_freq=5
[LightGBM] [Warning] bagging_fraction is set=0.8, subsample=0.7 will be ignored.
Current value: bagging_fraction=0.8
[LightGBM] [Warning] feature_fraction is set=1.0, colsample_bytree=0.9 will be
ignored. Current value: feature_fraction=1.0
[LightGBM] [Info] Number of positive: 20500, number of negative: 276006
[LightGBM] [Info] Auto-choosing row-wise multi-threading, the overhead of
testing was 0.098404 seconds.
You can set `force_row_wise=true` to remove the overhead.
And if memory is not enough, you can set `force_col_wise=true`.
[LightGBM] [Info] Total Bins 13717
[LightGBM] [Info] Number of data points in the train set: 296506, number of used
features: 35
[LightGBM] [Info] [binary:BoostFromScore]: pavg=0.069139 -> initscore=-2.599998
[LightGBM] [Info] Start training from score -2.599998
[LightGBM] [Warning] bagging_freq is set=5, subsample_freq=0 will be ignored.
Current value: bagging_freq=5
[LightGBM] [Warning] bagging_fraction is set=0.8, subsample=0.7 will be ignored.
Current value: bagging_fraction=0.8
[LightGBM] [Warning] feature_fraction is set=1.0, colsample_bytree=0.9 will be
ignored. Current value: feature_fraction=1.0
[LightGBM] [Warning] Categorical features with more bins than the configured
maximum bin number found.
[LightGBM] [Warning] For categorical features, max_bin and max_bin_by_feature
may be ignored with a large number of categories.
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] bagging_freq is set=5, subsample_freq=0 will be ignored.
Current value: bagging_freq=5
[LightGBM] [Warning] bagging_fraction is set=0.8, subsample=0.7 will be ignored.
Current value: bagging_fraction=0.8
[LightGBM] [Warning] feature_fraction is set=1.0, colsample_bytree=0.9 will be
ignored. Current value: feature_fraction=1.0
[LightGBM] [Info] Number of positive: 20499, number of negative: 276007
[LightGBM] [Info] Auto-choosing row-wise multi-threading, the overhead of

```

testing was 0.057273 seconds.

You can set `force_row_wise=true` to remove the overhead.

And if memory is not enough, you can set `force_col_wise=true`.

[LightGBM] [Info] Total Bins 13726

[LightGBM] [Info] Number of data points in the train set: 296506, number of used features: 35

[LightGBM] [Info] [binary:BoostFromScore]: pavg=0.069135 -> initscore=-2.600050

[LightGBM] [Info] Start training from score -2.600050

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

```

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] bagging_freq is set=1, subsample_freq=0 will be ignored.
Current value: bagging_freq=1
[LightGBM] [Warning] bagging_fraction is set=0.6, subsample=0.9 will be ignored.
Current value: bagging_fraction=0.6
[LightGBM] [Warning] feature_fraction is set=0.8, colsample_bytree=0.9 will be
ignored. Current value: feature_fraction=0.8
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] Categorical features with more bins than the configured
maximum bin number found.
[LightGBM] [Warning] For categorical features, max_bin and max_bin_by_feature
may be ignored with a large number of categories.
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] bagging_freq is set=1, subsample_freq=0 will be ignored.
Current value: bagging_freq=1
[LightGBM] [Warning] bagging_fraction is set=0.6, subsample=0.9 will be ignored.
Current value: bagging_fraction=0.6
[LightGBM] [Warning] feature_fraction is set=0.8, colsample_bytree=0.9 will be
ignored. Current value: feature_fraction=0.8
[LightGBM] [Info] Number of positive: 20499, number of negative: 276006

```


[illegible]

[illegible]

[illegible][illegible]

[illegible]

[illegible]

may be ignored with a large number of categories.

[illegible]

Current value: bagging_freq=1

```
[LightGBM] [Warning] bagging_fraction is set=0.6, subsample=0.9 will be ignored.
```

Current value: bagging fraction=0.6

```
[LightGBM] [Warning] feature_fraction is set=0.8, colsample_bytree=0.9 will be ignored. Current value: feature fraction=0.8
```

```
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
```

```
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
```

```
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
```

```
[LightGBM] [Info] Number of positive: 20499, number of negative: 276007
```

```
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
```

```
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
```

```
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
```

```
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
```

```
[LightGBM] [warning] No further splits with positive gain, best gain:  in
[LightGBM] [Info] Auto-choosing row-wise multi-threading, the overhead of
testing was 0.014129 seconds.
```

You can set ``force row wise=true`` to remove the overhead.

And if memory is not enough, you can set ``force_col_wise=true``.

```
[LightGBM] [Info] Start training from score -2.600050
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] bagging_freq is set=1, subsample_freq=0 will be ignored.
Current value: bagging_freq=1
[LightGBM] [Warning] feature_fraction is set=0.6, colsample_bytree=0.7 will be
ignored. Current value: feature_fraction=0.6
[LightGBM] [Warning] bagging_fraction is set=0.6, subsample=0.7 will be ignored.
Current value: bagging fraction=0.6
```

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]


```
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf  
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf  
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf  
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf  
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf  
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf  
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf  
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf  
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf  
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf  
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf  
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf  
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf  
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf  
[LightGBM] [Warning] No further splits with positive gain, best gain:  
-inf [LightGBM] [Warning] No further splits with positive gain, best gain: -inf  
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
```

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

```
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain:
```

-inf

```
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
```


[illegible]

[illegible]

[illegible]


```

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] bagging_freq is set=1, subsample_freq=0 will be ignored.
Current value: bagging_freq=1
[LightGBM] [Warning] feature_fraction is set=0.8, colsample_bytree=0.9 will be
ignored. Current value: feature_fraction=0.8
[LightGBM] [Warning] bagging_fraction is set=0.6, subsample=0.9 will be ignored.
Current value: bagging_fraction=0.6
[LightGBM] [Warning] Categorical features with more bins than the configured
maximum bin number found.
[LightGBM] [Warning] For categorical features, max_bin and max_bin_by_feature
may be ignored with a large number of categories.
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

```


[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]


```

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] bagging_freq is set=1, subsample_freq=0 will be ignored.
Current value: bagging_freq=1
[LightGBM] [Warning] feature_fraction is set=0.8, colsample_bytree=0.9 will be
ignored. Current value: feature_fraction=0.8
[LightGBM] [Warning] bagging_fraction is set=0.6, subsample=0.9 will be ignored.
Current value: bagging_fraction=0.6
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] bagging_freq is set=5, subsample_freq=0 will be ignored.
Current value: bagging_freq=5
[LightGBM] [Warning] bagging_fraction is set=1.0, subsample=0.7 will be ignored.
Current value: bagging_fraction=1.0
[LightGBM] [Warning] feature_fraction is set=1.0, colsample_bytree=0.9 will be
ignored. Current value: feature_fraction=1.0
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

```

[illegible]

[illegible]


```

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] Categorical features with more bins than the configured
maximum bin number found.
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] For categorical features, max_bin and max_bin_by_feature
may be ignored with a large number of categories.
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] bagging_freq is set=3, subsample_freq=0 will be ignored.
Current value: bagging_freq=3
[LightGBM] [Warning] bagging_fraction is set=0.8, subsample=0.7 will be ignored.
Current value: bagging_fraction=0.8
[LightGBM] [Warning] feature_fraction is set=0.6, colsample_bytree=0.7 will be
ignored. Current value: feature_fraction=0.6
[LightGBM] [Info] Number of positive: 20499, number of negative: 276006
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Info] Auto-choosing row-wise multi-threading, the overhead of

```

testing was 0.039254 seconds.

You can set `force_row_wise=true` to remove the overhead.

And if memory is not enough, you can set `force_col_wise=true`.

[LightGBM] [Info] Total Bins 13211

[LightGBM] [Info] Number of data points in the train set: 296505, number of used features: 35

[LightGBM] [Info] [binary:BoostFromScore]: pavg=0.069135 -> initscore=-2.600047

[LightGBM] [Info] Start training from score -2.600047

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf


```

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] Categorical features with more bins than the configured
maximum bin number found.
[LightGBM] [Warning] For categorical features, max_bin and max_bin_by_feature
may be ignored with a large number of categories.
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] bagging_freq is set=3, subsample_freq=0 will be ignored.
Current value: bagging_freq=3
[LightGBM] [Warning] bagging_fraction is set=0.8, subsample=0.7 will be ignored.
Current value: bagging_fraction=0.8
[LightGBM] [Warning] feature_fraction is set=0.6, colsample_bytree=0.7 will be
ignored. Current value: feature_fraction=0.6

```



```

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] bagging_freq is set=3, subsample_freq=0 will be ignored.
Current value: bagging_freq=3
[LightGBM] [Warning] bagging_fraction is set=0.8, subsample=0.7 will be ignored.
Current value: bagging_fraction=0.8
[LightGBM] [Warning] feature_fraction is set=0.6, colsample_bytree=0.7 will be
ignored. Current value: feature_fraction=0.6
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] Categorical features with more bins than the configured
maximum bin number found.
[LightGBM] [Warning] For categorical features, max_bin and max_bin_by_feature
may be ignored with a large number of categories.
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

```



```

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] bagging_freq is set=3, subsample_freq=0 will be ignored.
Current value: bagging_freq=3
[LightGBM] [Warning] bagging_fraction is set=0.8, subsample=0.7 will be ignored.
Current value: bagging_fraction=0.8
[LightGBM] [Warning] feature_fraction is set=0.6, colsample_bytree=0.7 will be
ignored. Current value: feature_fraction=0.6
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] Categorical features with more bins than the configured
maximum bin number found.
[LightGBM] [Warning] For categorical features, max_bin and max_bin_by_feature
may be ignored with a large number of categories.
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] bagging_freq is set=3, subsample_freq=0 will be ignored.
Current value: bagging_freq=3
[LightGBM] [Warning] bagging_fraction is set=0.8, subsample=0.7 will be ignored.
Current value: bagging_fraction=0.8
[LightGBM] [Warning] feature_fraction is set=0.6, colsample_bytree=0.7 will be
ignored. Current value: feature_fraction=0.6
[LightGBM] [Info] Number of positive: 20499, number of negative: 276007
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Info] Auto-choosing row-wise multi-threading, the overhead of

```


testing was 0.041204 seconds.

You can set `force_row_wise=true` to remove the overhead.

And if memory is not enough, you can set `force_col_wise=true`.

[LightGBM] [Info] Total Bins 13726

[LightGBM] [Info] Number of data points in the train set: 296506, number of used features: 35

[LightGBM] [Info] [binary:BoostFromScore]: pavg=0.069135 -> initscore=-2.600050

[LightGBM] [Info] Start training from score -2.600050

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf


```

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] Categorical features with more bins than the configured
maximum bin number found.
[LightGBM] [Warning] For categorical features, max_bin and max_bin_by_feature
may be ignored with a large number of categories.
[LightGBM] [Warning] bagging_freq is set=3, subsample_freq=0 will be ignored.
Current value: bagging_freq=3
[LightGBM] [Warning] bagging_fraction is set=0.8, subsample=0.7 will be ignored.
Current value: bagging_fraction=0.8
[LightGBM] [Warning] feature_fraction is set=0.6, colsample_bytree=0.7 will be
ignored. Current value: feature_fraction=0.6
[LightGBM] [Info] Number of positive: 20499, number of negative: 276007
[LightGBM] [Info] Auto-choosing row-wise multi-threading, the overhead of
testing was 0.033140 seconds.
You can set `force_row_wise=true` to remove the overhead.
And if memory is not enough, you can set `force_col_wise=true`.
[LightGBM] [Info] Total Bins 13769
[LightGBM] [Info] Number of data points in the train set: 296506, number of used
features: 35
[LightGBM] [Info] [binary:BoostFromScore]: pavg=0.069135 -> initscore=-2.600050
[LightGBM] [Info] Start training from score -2.600050
[LightGBM] [Warning] bagging_freq is set=5, subsample_freq=0 will be ignored.
Current value: bagging_freq=5
[LightGBM] [Warning] feature_fraction is set=1.0, colsample_bytree=0.9 will be
ignored. Current value: feature_fraction=1.0
[LightGBM] [Warning] bagging_fraction is set=1.0, subsample=0.7 will be ignored.
Current value: bagging_fraction=1.0
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] bagging_freq is set=5, subsample_freq=0 will be ignored.
Current value: bagging_freq=5
[LightGBM] [Warning] feature_fraction is set=1.0, colsample_bytree=0.7 will be
ignored. Current value: feature_fraction=1.0
[LightGBM] [Warning] bagging_fraction is set=1.0, subsample=0.7 will be ignored.
Current value: bagging_fraction=1.0
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

```

```

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] bagging_freq is set=3, subsample_freq=0 will be ignored.
Current value: bagging_freq=3
[LightGBM] [Warning] bagging_fraction is set=0.8, subsample=0.9 will be ignored.
Current value: bagging_fraction=0.8
[LightGBM] [Warning] feature_fraction is set=0.8, colsample_bytree=0.9 will be
ignored. Current value: feature_fraction=0.8
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] Categorical features with more bins than the configured
maximum bin number found.
[LightGBM] [Warning] For categorical features, max_bin and max_bin_by_feature
may be ignored with a large number of categories.
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] bagging_freq is set=3, subsample_freq=0 will be ignored.
Current value: bagging_freq=3
[LightGBM] [Warning] bagging_fraction is set=0.8, subsample=0.9 will be ignored.
Current value: bagging_fraction=0.8
[LightGBM] [Warning] feature_fraction is set=0.8, colsample_bytree=0.9 will be
ignored. Current value: feature_fraction=0.8
[LightGBM] [Info] Number of positive: 20499, number of negative: 276006
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Info] Auto-choosing row-wise multi-threading, the overhead of
testing was 0.069823 seconds.
You can set `force_row_wise=true` to remove the overhead.
And if memory is not enough, you can set `force_col_wise=true`.
[LightGBM] [Info] Total Bins 13211
[LightGBM] [Info] Number of data points in the train set: 296505, number of used
features: 35
[LightGBM] [Info] [binary:BoostFromScore]: pavg=0.069135 -> initscore=-2.600047
[LightGBM] [Info] Start training from score -2.600047
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

```


[illegible]

```

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] bagging_freq is set=5, subsample_freq=0 will be ignored.
Current value: bagging_freq=5
[LightGBM] [Warning] feature_fraction is set=1.0, colsample_bytree=0.9 will be
ignored. Current value: feature_fraction=1.0
[LightGBM] [Warning] bagging_fraction is set=1.0, subsample=0.7 will be ignored.
Current value: bagging_fraction=1.0
[LightGBM] [Warning] bagging_freq is set=3, subsample_freq=0 will be ignored.
Current value: bagging_freq=3
[LightGBM] [Warning] bagging_fraction is set=0.8, subsample=0.9 will be ignored.
Current value: bagging_fraction=0.8
[LightGBM] [Warning] feature_fraction is set=0.8, colsample_bytree=0.9 will be
ignored. Current value: feature_fraction=0.8
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] Categorical features with more bins than the configured
maximum bin number found.
[LightGBM] [Warning] For categorical features, max_bin and max_bin_by_feature
may be ignored with a large number of categories.
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

```

```

[LightGBM] [Warning] bagging_freq is set=3, subsample_freq=0 will be ignored.
Current value: bagging_freq=3
[LightGBM] [Warning] feature_fraction is set=0.6, colsample_bytree=0.7 will be
ignored. Current value: feature_fraction=0.6
[LightGBM] [Warning] bagging_fraction is set=0.8, subsample=0.7 will be ignored.
Current value: bagging_fraction=0.8
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] bagging_freq is set=3, subsample_freq=0 will be ignored.
Current value: bagging_freq=3
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] bagging_fraction is set=0.8, subsample=0.9 will be ignored.
Current value: bagging_fraction=0.8
[LightGBM] [Warning] feature_fraction is set=0.8, colsample_bytree=0.9 will be
ignored. Current value: feature_fraction=0.8
[LightGBM] [Info] Number of positive: 20499, number of negative: 276006
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Info] Auto-choosing row-wise multi-threading, the overhead of
testing was 0.047216 seconds.
You can set `force_row_wise=true` to remove the overhead.
And if memory is not enough, you can set `force_col_wise=true`.
[LightGBM] [Info] Total Bins 13359
[LightGBM] [Info] Number of data points in the train set: 296505, number of used
features: 35
[LightGBM] [Info] [binary:BoostFromScore]: pavg=0.069135 -> initscore=-2.600047
[LightGBM] [Info] Start training from score -2.600047
[LightGBM] [Warning] No further splits with positive gain, best gain:
-inf[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

```



```

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] bagging_freq is set=3, subsample_freq=0 will be ignored.
Current value: bagging_freq=3
[LightGBM] [Warning] bagging_fraction is set=0.8, subsample=0.9 will be ignored.
Current value: bagging_fraction=0.8
[LightGBM] [Warning] feature_fraction is set=0.8, colsample_bytree=0.9 will be
ignored. Current value: feature_fraction=0.8
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] Categorical features with more bins than the configured
maximum bin number found.
[LightGBM] [Warning] For categorical features, max_bin and max_bin_by_feature
may be ignored with a large number of categories.
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] bagging_freq is set=5, subsample_freq=0 will be ignored.
Current value: bagging_freq=5
[LightGBM] [Warning] feature_fraction is set=1.0, colsample_bytree=0.7 will be
ignored. Current value: feature_fraction=1.0
[LightGBM] [Warning] bagging_fraction is set=1.0, subsample=0.7 will be ignored.
Current value: bagging_fraction=1.0
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] bagging_freq is set=3, subsample_freq=0 will be ignored.
Current value: bagging_freq=3
[LightGBM] [Warning] bagging_fraction is set=0.8, subsample=0.9 will be ignored.
Current value: bagging_fraction=0.8

```

```

[LightGBM] [Warning] feature_fraction is set=0.8, colsample_bytree=0.9 will be
ignored. Current value: feature_fraction=0.8
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] bagging_freq is set=3, subsample_freq=0 will be ignored.
Current value: bagging_freq=3
[LightGBM] [Warning] bagging_fraction is set=0.8, subsample=0.9 will be ignored.
Current value: bagging_fraction=0.8
[LightGBM] [Warning] feature_fraction is set=0.8, colsample_bytree=0.9 will be
ignored. Current value: feature_fraction=0.8
[LightGBM] [Info] Number of positive: 20500, number of negative: 276006
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Info] Auto-choosing row-wise multi-threading, the overhead of
testing was 0.097801 seconds.
You can set `force_row_wise=true` to remove the overhead.
And if memory is not enough, you can set `force_col_wise=true`.
[LightGBM] [Info] Total Bins 13717
[LightGBM] [Info] Number of data points in the train set: 296506, number of used
features: 35
[LightGBM] [Info] [binary:BoostFromScore]: pavg=0.069139 -> initscore=-2.599998
[LightGBM] [Info] Start training from score -2.599998
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] bagging_freq is set=3, subsample_freq=0 will be ignored.
Current value: bagging_freq=3
[LightGBM] [Warning] bagging_fraction is set=0.8, subsample=0.9 will be ignored.
Current value: bagging_fraction=0.8
[LightGBM] [Warning] feature_fraction is set=0.8, colsample_bytree=0.9 will be
ignored. Current value: feature_fraction=0.8
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] Categorical features with more bins than the configured
maximum bin number found.
[LightGBM] [Warning] For categorical features, max_bin and max_bin_by_feature
may be ignored with a large number of categories.
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] Categorical features with more bins than the configured
maximum bin number found.
[LightGBM] [Warning] For categorical features, max_bin and max_bin_by_feature
may be ignored with a large number of categories.
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

```

```

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] bagging_freq is set=5, subsample_freq=0 will be ignored.
Current value: bagging_freq=5
[LightGBM] [Warning] feature_fraction is set=1.0, colsample_bytree=0.7 will be
ignored. Current value: feature_fraction=1.0
[LightGBM] [Warning] bagging_fraction is set=1.0, subsample=0.7 will be ignored.
Current value: bagging_fraction=1.0
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] bagging_freq is set=3, subsample_freq=0 will be ignored.
Current value: bagging_freq=3
[LightGBM] [Warning] bagging_fraction is set=0.8, subsample=0.9 will be ignored.
Current value: bagging_fraction=0.8
[LightGBM] [Warning] feature_fraction is set=0.8, colsample_bytree=0.9 will be
ignored. Current value: feature_fraction=0.8
[LightGBM] [Warning] bagging_freq is set=3, subsample_freq=0 will be ignored.
Current value: bagging_freq=3
[LightGBM] [Warning] bagging_fraction is set=0.8, subsample=0.9 will be ignored.
Current value: bagging_fraction=0.8
[LightGBM] [Warning] feature_fraction is set=0.8, colsample_bytree=0.9 will be
ignored. Current value: feature_fraction=0.8
[LightGBM] [Info] Number of positive: 20499, number of negative: 276007
[LightGBM] [Info] Number of positive: 20499, number of negative: 276007
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Info] Auto-choosing col-wise multi-threading, the overhead of
testing was 0.026020 seconds.
You can set `force_col_wise=true` to remove the overhead.
[LightGBM] [Info] Total Bins 13726
[LightGBM] [Info] Number of data points in the train set: 296506, number of used
features: 35
[LightGBM] [Info] [binary:BoostFromScore]: pavg=0.069135 -> initscore=-2.600050
[LightGBM] [Info] Start training from score -2.600050
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Info] Auto-choosing row-wise multi-threading, the overhead of
testing was 0.066471 seconds.
You can set `force_row_wise=true` to remove the overhead.
And if memory is not enough, you can set `force_col_wise=true`.
[LightGBM] [Info] Total Bins 13769
[LightGBM] [Info] Number of data points in the train set: 296506, number of used
features: 35
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

```


[illegible]

[illegible]


```

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] Categorical features with more bins than the configured
maximum bin number found.
[LightGBM] [Warning] For categorical features, max_bin and max_bin_by_feature
may be ignored with a large number of categories.
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] bagging_freq is set=1, subsample_freq=0 will be ignored.
Current value: bagging_freq=1
[LightGBM] [Warning] bagging_fraction is set=1.0, subsample=0.9 will be ignored.
Current value: bagging_fraction=1.0
[LightGBM] [Warning] feature_fraction is set=0.6, colsample_bytree=0.9 will be
ignored. Current value: feature_fraction=0.6
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Info] Number of positive: 20499, number of negative: 276006
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Info] Auto-choosing row-wise multi-threading, the overhead of
testing was 0.009566 seconds.
You can set `force_row_wise=true` to remove the overhead.
And if memory is not enough, you can set `force_col_wise=true`.
[LightGBM] [Info] Total Bins 13359
[LightGBM] [Info] Number of data points in the train set: 296505, number of used
features: 35
[LightGBM] [Info] [binary:BoostFromScore]: pavg=0.069135 -> initscore=-2.600047

```



```

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] bagging_freq is set=3, subsample_freq=0 will be ignored.
Current value: bagging_freq=3
[LightGBM] [Warning] feature_fraction is set=0.6, colsample_bytree=0.7 will be
ignored. Current value: feature_fraction=0.6
[LightGBM] [Warning] bagging_fraction is set=0.8, subsample=0.7 will be ignored.
Current value: bagging_fraction=0.8
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] Categorical features with more bins than the configured
maximum bin number found.
[LightGBM] [Warning] For categorical features, max_bin and max_bin_by_feature
may be ignored with a large number of categories.
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] bagging_freq is set=1, subsample_freq=0 will be ignored.
Current value: bagging_freq=1
[LightGBM] [Warning] bagging_fraction is set=1.0, subsample=0.9 will be ignored.
Current value: bagging_fraction=1.0
[LightGBM] [Warning] feature_fraction is set=0.6, colsample_bytree=0.9 will be
ignored. Current value: feature_fraction=0.6
[LightGBM] [Info] Number of positive: 20500, number of negative: 276006
[LightGBM] [Info] Auto-choosing row-wise multi-threading, the overhead of
testing was 0.022943 seconds.
You can set `force_row_wise=true` to remove the overhead.
And if memory is not enough, you can set `force_col_wise=true`.
[LightGBM] [Info] Total Bins 13717
[LightGBM] [Info] Number of data points in the train set: 296506, number of used
features: 35
[LightGBM] [Info] [binary:BoostFromScore]: pavg=0.069139 ->
initscore=-2.599998[LightGBM] [Warning] No further splits with positive gain,
best gain: -inf

```



```

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] bagging_freq is set=1, subsample_freq=0 will be ignored.
Current value: bagging_freq=1
[LightGBM] [Warning] bagging_fraction is set=1.0, subsample=0.9 will be ignored.
Current value: bagging_fraction=1.0
[LightGBM] [Warning] feature_fraction is set=0.6, colsample_bytree=0.9 will be
ignored. Current value: feature_fraction=0.6
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] bagging_freq is set=1, subsample_freq=0 will be ignored.
Current value: bagging_freq=1
[LightGBM] [Warning] bagging_fraction is set=1.0, subsample=0.9 will be ignored.
Current value: bagging_fraction=1.0
[LightGBM] [Warning] feature_fraction is set=0.6, colsample_bytree=0.9 will be
ignored. Current value: feature_fraction=0.6
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] bagging_freq is set=3, subsample_freq=0 will be ignored.
Current value: bagging_freq=3
[LightGBM] [Warning] bagging_fraction is set=0.6, subsample=0.9 will be ignored.
Current value: bagging_fraction=0.6
[LightGBM] [Warning] feature_fraction is set=0.8, colsample_bytree=0.9 will be
ignored. Current value: feature_fraction=0.8
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] Categorical features with more bins than the configured
maximum bin number found.
[LightGBM] [Warning] For categorical features, max_bin and max_bin_by_feature
may be ignored with a large number of categories.

```



```

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] Categorical features with more bins than the configured
maximum bin number found.
[LightGBM] [Warning] For categorical features, max_bin and max_bin_by_feature
may be ignored with a large number of categories.
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] Categorical features with more bins than the configured
maximum bin number found.
[LightGBM] [Warning] For categorical features, max_bin and max_bin_by_feature
may be ignored with a large number of categories.
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] bagging_freq is set=1, subsample_freq=0 will be ignored.
Current value: bagging_freq=1
[LightGBM] [Warning] bagging_fraction is set=1.0, subsample=0.9 will be ignored.
Current value: bagging_fraction=1.0
[LightGBM] [Warning] feature_fraction is set=0.6, colsample_bytree=0.9 will be
ignored. Current value: feature_fraction=0.6
[LightGBM] [Info] Number of positive: 20499, number of negative: 276007
[LightGBM] [Info] Auto-choosing row-wise multi-threading, the overhead of
testing was 0.006566 seconds.
You can set `force_row_wise=true` to remove the overhead.
And if memory is not enough, you can set `force_col_wise=true`.
[LightGBM] [Info] Total Bins 13726
[LightGBM] [Info] Number of data points in the train set: 296506, number of used
features: 35
[LightGBM] [Info] [binary:BoostFromScore]: pavg=0.069135 -> initscore=-2.600050
[LightGBM] [Info] Start training from score -2.600050
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] bagging_freq is set=1, subsample_freq=0 will be ignored.
Current value: bagging_freq=1
[LightGBM] [Warning] bagging_fraction is set=1.0, subsample=0.9 will be ignored.
Current value: bagging_fraction=1.0
[LightGBM] [Warning] feature_fraction is set=0.6, colsample_bytree=0.9 will be
ignored. Current value: feature_fraction=0.6
[LightGBM] [Info] Number of positive: 20499, number of negative: 276007
[LightGBM] [Warning] bagging_freq is set=3, subsample_freq=0 will be ignored.
Current value: bagging_freq=3
[LightGBM] [Warning] bagging_fraction is set=0.6, subsample=0.9 will be ignored.
Current value: bagging_fraction=0.6
[LightGBM] [Warning] feature_fraction is set=0.8, colsample_bytree=0.9 will be
ignored. Current value: feature_fraction=0.8
[LightGBM] [Info] Auto-choosing row-wise multi-threading, the overhead of

```

testing was 0.024902 seconds.

You can set `force_row_wise=true` to remove the overhead.

And if memory is not enough, you can set `force_col_wise=true`.

```
[LightGBM] [Info] Total Bins 13769
[LightGBM] [Info] Number of data points in the train set: 296506, number of used
features: 35
[LightGBM] [Info] [binary:BoostFromScore]: pavg=0.069135 -> initscore=-2.600050
[LightGBM] [Info] Start training from score -2.600050
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Info] Number of positive: 20499, number of negative: 276006
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Info] Auto-choosing col-wise multi-threading, the overhead of
testing was 0.095418 seconds.
You can set `force_col_wise=true` to remove the overhead.
[LightGBM] [Info] Total Bins 13211
[LightGBM] [Info] Number of data points in the train set: 296505, number of used
features: 35
[LightGBM] [Info] [binary:BoostFromScore]: pavg=0.069135 -> initscore=-2.600047
[LightGBM] [Info] Start training from score -2.600047
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] bagging_freq is set=3, subsample_freq=0 will be ignored.
Current value: bagging_freq=3
[LightGBM] [Warning] bagging_fraction is set=0.6, subsample=0.9 will be ignored.
Current value: bagging_fraction=0.6
[LightGBM] [Warning] feature_fraction is set=0.8, colsample_bytree=0.9 will be
ignored. Current value: feature_fraction=0.8
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] bagging_freq is set=3, subsample_freq=0 will be ignored.
Current value: bagging_freq=3
[LightGBM] [Warning] feature_fraction is set=0.8, colsample_bytree=0.9 will be
ignored. Current value: feature_fraction=0.8
[LightGBM] [Warning] bagging_fraction is set=0.8, subsample=0.9 will be ignored.
```

Current value: bagging_fraction=0.8

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] Categorical features with more bins than the configured maximum bin number found.

[LightGBM] [Warning] For categorical features, max_bin and max_bin_by_feature may be ignored with a large number of categories.

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] bagging_freq is set=3, subsample_freq=0 will be ignored.

Current value: bagging_freq=3

[LightGBM] [Warning] bagging_fraction is set=0.6, subsample=0.9 will be ignored.

Current value: bagging_fraction=0.6

[LightGBM] [Warning] feature_fraction is set=0.8, colsample_bytree=0.9 will be ignored. Current value: feature_fraction=0.8

[LightGBM] [Info] Number of positive: 20499, number of negative: 276006

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Info] Auto-choosing row-wise multi-threading, the overhead of testing was 0.106575 seconds.

You can set `force_row_wise=true` to remove the overhead.

And if memory is not enough, you can set `force_col_wise=true`. [LightGBM]

[Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Info] Total Bins 13359

[LightGBM] [Info] Number of data points in the train set: 296505, number of used features: 35

[LightGBM] [Info] [binary:BoostFromScore]: pavg=0.069135 -> initscore=-2.600047

[LightGBM] [Info] Start training from score -2.600047

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

```

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] bagging_freq is set=3, subsample_freq=0 will be ignored.
Current value: bagging_freq=3
[LightGBM] [Warning] bagging_fraction is set=0.6, subsample=0.9 will be ignored.
Current value: bagging_fraction=0.6
[LightGBM] [Warning] feature_fraction is set=0.8, colsample_bytree=0.9 will be
ignored. Current value: feature_fraction=0.8
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] bagging_freq is set=3, subsample_freq=0 will be ignored.
Current value: bagging_freq=3
[LightGBM] [Warning] feature_fraction is set=0.8, colsample_bytree=0.9 will be
ignored. Current value: feature_fraction=0.8
[LightGBM] [Warning] bagging_fraction is set=0.8, subsample=0.9 will be ignored.
Current value: bagging_fraction=0.8
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] Categorical features with more bins than the configured
maximum bin number found.
[LightGBM] [Warning] For categorical features, max_bin and max_bin_by_feature
may be ignored with a large number of categories.
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

```



```

[LightGBM] [Warning] bagging_fraction is set=0.6, subsample=0.9 will be ignored.
Current value: bagging_fraction=0.6
[LightGBM] [Warning] feature_fraction is set=0.8, colsample_bytree=0.9 will be
ignored. Current value: feature_fraction=0.8
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] bagging_freq is set=3, subsample_freq=0 will be ignored.
Current value: bagging_freq=3
[LightGBM] [Warning] feature_fraction is set=0.8, colsample_bytree=0.9 will be
ignored. Current value: feature_fraction=0.8
[LightGBM] [Warning] bagging_fraction is set=0.8, subsample=0.9 will be ignored.
Current value: bagging_fraction=0.8
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] Categorical features with more bins than the configured
maximum bin number found.
[LightGBM] [Warning] For categorical features, max_bin and max_bin_by_feature
may be ignored with a large number of categories.
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] bagging_freq is set=3, subsample_freq=0 will be ignored.
Current value: bagging_freq=3
[LightGBM] [Warning] bagging_fraction is set=0.6, subsample=0.9 will be ignored.
Current value: bagging_fraction=0.6
[LightGBM] [Warning] feature_fraction is set=0.8, colsample_bytree=0.9 will be
ignored. Current value: feature_fraction=0.8
[LightGBM] [Info] Number of positive: 20499, number of negative: 276007
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Info] Auto-choosing row-wise multi-threading, the overhead of
testing was 0.028434 seconds.
You can set `force_row_wise=true` to remove the overhead.
And if memory is not enough, you can set `force_col_wise=true`.
[LightGBM] [Info] Total Bins 13726
[LightGBM] [Info] Number of data points in the train set: 296506, number of used
features: 35
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

```

```

[LightGBM] [Info] [binary:BoostFromScore]: pavg=0.069135 -> initscore=-2.600050
[LightGBM] [Info] Start training from score -2.600050
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] bagging_freq is set=3, subsample_freq=0 will be ignored.
Current value: bagging_freq=3
[LightGBM] [Warning] bagging_fraction is set=0.6, subsample=0.9 will be ignored.
Current value: bagging_fraction=0.6
[LightGBM] [Warning] feature_fraction is set=0.8, colsample_bytree=0.9 will be
ignored. Current value: feature_fraction=0.8
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] Categorical features with more bins than the configured
maximum bin number found.
[LightGBM] [Warning] For categorical features, max_bin and max_bin_by_feature
may be ignored with a large number of categories.

```


[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[LightGBM] [Warning] feature_fraction is set=1.0, colsample_bytree=0.7 will be ignored. Current value: feature_fraction=1.0

[LightGBM] [Warning] No further splits with positive gain, best gain:

-inf [LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] Categorical features with more bins than the configured maximum bin number found.

[LightGBM] [Warning] For categorical features, max_bin and max_bin_by_feature may be ignored with a large number of categories.

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]


```

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] bagging_freq is set=1, subsample_freq=0 will be ignored.
Current value: bagging_freq=1
[LightGBM] [Warning] feature_fraction is set=0.6, colsample_bytree=0.9 will be
ignored. Current value: feature_fraction=0.6
[LightGBM] [Warning] bagging_fraction is set=1.0, subsample=0.9 will be ignored.
Current value: bagging_fraction=1.0
[LightGBM] [Warning] bagging_freq is set=3, subsample_freq=0 will be ignored.
Current value: bagging_freq=3
[LightGBM] [Warning] bagging_fraction is set=0.8, subsample=0.7 will be ignored.
Current value: bagging_fraction=0.8
[LightGBM] [Warning] feature_fraction is set=1.0, colsample_bytree=0.7 will be
ignored. Current value: feature_fraction=1.0
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] Categorical features with more bins than the configured
maximum bin number found.
[LightGBM] [Warning] For categorical features, max_bin and max_bin_by_feature
may be ignored with a large number of categories.
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

```



```

[LightGBM] [Warning] Categorical features with more bins than the configured
maximum bin number found.
[LightGBM] [Warning] For categorical features, max_bin and max_bin_by_feature
may be ignored with a large number of categories.
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] bagging_freq is set=3, subsample_freq=0 will be ignored.
Current value: bagging_freq=3
[LightGBM] [Warning] bagging_fraction is set=0.8, subsample=0.7 will be ignored.
Current value: bagging_fraction=0.8
[LightGBM] [Warning] feature_fraction is set=1.0, colsample_bytree=0.7 will be
ignored. Current value: feature_fraction=1.0
[LightGBM] [Info] Number of positive: 20499, number of negative: 276007
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Info] Auto-choosing row-wise multi-threading, the overhead of
testing was 0.100325 seconds.
You can set `force_row_wise=true` to remove the overhead.
And if memory is not enough, you can set `force_col_wise=true`.
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Info] Total Bins 13769
[LightGBM] [Info] Number of data points in the train set: 296506, number of used
features: 35
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

```


[illegible]

may be ignored with a large number of categories.

```
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] bagging_freq is set=3, subsample_freq=0 will be ignored.
Current value: bagging_freq=3
[LightGBM] [Warning] bagging_fraction is set=1.0, subsample=0.9 will be ignored.
Current value: bagging_fraction=1.0
[LightGBM] [Warning] feature_fraction is set=0.8, colsample_bytree=0.7 will be
ignored. Current value: feature_fraction=0.8
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Info] Number of positive: 20499, number of negative:
276006[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Info] Auto-choosing row-wise multi-threading, the overhead of
testing was 0.015960 seconds.
You can set `force_row_wise=true` to remove the overhead.
And if memory is not enough, you can set `force_col_wise=true`.
[LightGBM] [Info] Total Bins 13211
[LightGBM] [Info] Number of data points in the train set: 296505, number of used
features: 35
[LightGBM] [Info] [binary:BoostFromScore]: pavg=0.069135 -> initscore=-2.600047
[LightGBM] [Info] Start training from score -2.600047
```



```

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] Categorical features with more bins than the configured
maximum bin number found.
[LightGBM] [Warning] For categorical features, max_bin and max_bin_by_feature
may be ignored with a large number of categories.
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] bagging_freq is set=3, subsample_freq=0 will be ignored.
Current value: bagging_freq=3
[LightGBM] [Warning] bagging_fraction is set=1.0, subsample=0.9 will be ignored.
Current value: bagging_fraction=1.0
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] feature_fraction is set=0.8, colsample_bytree=0.7 will be
ignored. Current value: feature_fraction=0.8
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Info] Number of positive: 20499, number of negative: 276006
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Info] Auto-choosing row-wise multi-threading, the overhead of
testing was 0.031044 seconds.
You can set `force_row_wise=true` to remove the overhead.
And if memory is not enough, you can set `force_col_wise=true`.
[LightGBM] [Info] Total Bins 13359
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Info] Number of data points in the train set: 296505, number of used

```



```
features: 35
```

[illegible]


```

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] bagging_freq is set=3, subsample_freq=0 will be ignored.
Current value: bagging_freq=3
[LightGBM] [Warning] bagging_fraction is set=1.0, subsample=0.9 will be ignored.
Current value: bagging_fraction=1.0
[LightGBM] [Warning] feature_fraction is set=0.8, colsample_bytree=0.7 will be
ignored. Current value: feature_fraction=0.8
[LightGBM] [Info] Number of positive: 20499, number of negative: 276007
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Info] Auto-choosing row-wise multi-threading, the overhead of
testing was 0.008423 seconds.
You can set `force_row_wise=true` to remove the overhead.
And if memory is not enough, you can set `force_col_wise=true`.
[LightGBM] [Info] Total Bins 13726
[LightGBM] [Info] Number of data points in the train set: 296506, number of used
features: 35
[LightGBM] [Info] [binary:BoostFromScore]: pavg=0.069135 -> initscore=-2.600050
[LightGBM] [Info] Start training from score -2.600050
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] bagging_freq is set=3, subsample_freq=0 will be ignored.

```

```

Current value: bagging_freq=3
[LightGBM] [Warning] feature_fraction is set=0.8, colsample_bytree=0.9 will be
ignored. Current value: feature_fraction=0.8
[LightGBM] [Warning] bagging_fraction is set=0.6, subsample=0.9 will be ignored.
Current value: bagging_fraction=0.6
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] bagging_freq is set=3, subsample_freq=0 will be ignored.
Current value: bagging_freq=3
[LightGBM] [Warning] bagging_fraction is set=1.0, subsample=0.9 will be ignored.
Current value: bagging_fraction=1.0
[LightGBM] [Warning] feature_fraction is set=0.8, colsample_bytree=0.7 will be
ignored. Current value: feature_fraction=0.8
[LightGBM] [Warning] Categorical features with more bins than the configured
maximum bin number found.
[LightGBM] [Warning] For categorical features, max_bin and max_bin_by_feature
may be ignored with a large number of categories.
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] bagging_freq is set=3, subsample_freq=0 will be ignored.
Current value: bagging_freq=3
[LightGBM] [Warning] feature_fraction is set=0.8, colsample_bytree=0.9 will be
ignored. Current value: feature_fraction=0.8
[LightGBM] [Warning] bagging_fraction is set=0.6, subsample=0.9 will be ignored.
Current value: bagging_fraction=0.6
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] bagging_freq is set=3, subsample_freq=0 will be ignored.
Current value: bagging_freq=3
[LightGBM] [Warning] bagging_fraction is set=1.0, subsample=0.9 will be ignored.
Current value: bagging_fraction=1.0
[LightGBM] [Warning] feature_fraction is set=0.8, colsample_bytree=0.7 will be
ignored. Current value: feature_fraction=0.8
[LightGBM] [Info] Number of positive: 20499, number of negative: 276007
[LightGBM] [Info] Auto-choosing row-wise multi-threading, the overhead of
testing was 0.050930 seconds.
You can set `force_row_wise=true` to remove the overhead.
And if memory is not enough, you can set `force_col_wise=true`.
[LightGBM] [Info] Total Bins 13769
[LightGBM] [Info] Number of data points in the train set: 296506, number of used
features: 35
[LightGBM] [Info] [binary:BoostFromScore]: pavg=0.069135 -> initscore=-2.600050
[LightGBM] [Info] Start training from score -2.600050
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] bagging_freq is set=3, subsample_freq=0 will be ignored.
Current value: bagging_freq=3

```


Current value: bagging_fraction=1.0
[LightGBM] [Warning] feature_fraction is set=0.6, colsample_bytree=0.7 will be ignored. Current value: feature_fraction=0.6
[LightGBM] [Warning] bagging_freq is set=3, subsample_freq=0 will be ignored.
Current value: bagging_freq=3
[LightGBM] [Warning] bagging_fraction is set=1.0, subsample=0.7 will be ignored.
Current value: bagging_fraction=1.0
[LightGBM] [Warning] feature_fraction is set=0.6, colsample_bytree=0.7 will be ignored. Current value: feature_fraction=0.6
[LightGBM] [Info] Number of positive: 20499, number of negative: 276006
[LightGBM] [Info] Auto-choosing row-wise multi-threading, the overhead of testing was 0.026202 seconds.
You can set `force_row_wise=true` to remove the overhead.
And if memory is not enough, you can set `force_col_wise=true`.
[LightGBM] [Info] Total Bins 13211
[LightGBM] [Info] Number of data points in the train set: 296505, number of used features: 35
[LightGBM] [Info] [binary:BoostFromScore]: pavg=0.069135 -> initscore=-2.600047
[LightGBM] [Info] Start training from score -2.600047
[LightGBM] [Warning] Categorical features with more bins than the configured maximum bin number found.
[LightGBM] [Warning] For categorical features, max_bin and max_bin_by_feature may be ignored with a large number of categories.
[LightGBM] [Warning] bagging_freq is set=3, subsample_freq=0 will be ignored.
Current value: bagging_freq=3
[LightGBM] [Warning] bagging_fraction is set=1.0, subsample=0.7 will be ignored.
Current value: bagging_fraction=1.0
[LightGBM] [Warning] feature_fraction is set=0.6, colsample_bytree=0.7 will be ignored. Current value: feature_fraction=0.6
[LightGBM] [Warning] bagging_freq is set=3, subsample_freq=0 will be ignored.
Current value: bagging_freq=3
[LightGBM] [Warning] bagging_fraction is set=1.0, subsample=0.7 will be ignored.
Current value: bagging_fraction=1.0
[LightGBM] [Warning] feature_fraction is set=0.6, colsample_bytree=0.7 will be ignored. Current value: feature_fraction=0.6
[LightGBM] [Info] Number of positive: 20499, number of negative: 276006
[LightGBM] [Info] Auto-choosing row-wise multi-threading, the overhead of testing was 0.022772 seconds.
You can set `force_row_wise=true` to remove the overhead.
And if memory is not enough, you can set `force_col_wise=true`.
[LightGBM] [Info] Total Bins 13359
[LightGBM] [Info] Number of data points in the train set: 296505, number of used features: 35
[LightGBM] [Info] [binary:BoostFromScore]: pavg=0.069135 -> initscore=-2.600047
[LightGBM] [Info] Start training from score -2.600047
[LightGBM] [Warning] Categorical features with more bins than the configured maximum bin number found.
[LightGBM] [Warning] For categorical features, max_bin and max_bin_by_feature

may be ignored with a large number of categories.

[LightGBM] [Warning] bagging_freq is set=3, subsample_freq=0 will be ignored.
Current value: bagging_freq=3

[LightGBM] [Warning] bagging_fraction is set=1.0, subsample=0.7 will be ignored.
Current value: bagging_fraction=1.0

[LightGBM] [Warning] feature_fraction is set=0.6, colsample_bytree=0.7 will be ignored. Current value: feature_fraction=0.6

[LightGBM] [Info] Number of positive: 20500, number of negative: 276006

[LightGBM] [Info] Auto-choosing col-wise multi-threading, the overhead of testing was 0.053463 seconds.
You can set `force_col_wise=true` to remove the overhead.

[LightGBM] [Info] Total Bins 13717

[LightGBM] [Info] Number of data points in the train set: 296506, number of used features: 35

[LightGBM] [Info] [binary:BoostFromScore]: pavg=0.069139 -> initscore=-2.599998

[LightGBM] [Info] Start training from score -2.599998

[LightGBM] [Warning] bagging_freq is set=3, subsample_freq=0 will be ignored.
Current value: bagging_freq=3

[LightGBM] [Warning] bagging_fraction is set=1.0, subsample=0.7 will be ignored.
Current value: bagging_fraction=1.0

[LightGBM] [Warning] feature_fraction is set=0.6, colsample_bytree=0.7 will be ignored. Current value: feature_fraction=0.6

[LightGBM] [Warning] Categorical features with more bins than the configured maximum bin number found.

[LightGBM] [Warning] For categorical features, max_bin and max_bin_by_feature may be ignored with a large number of categories.

[LightGBM] [Warning] bagging_freq is set=3, subsample_freq=0 will be ignored.
Current value: bagging_freq=3

[LightGBM] [Warning] bagging_fraction is set=1.0, subsample=0.7 will be ignored.
Current value: bagging_fraction=1.0

[LightGBM] [Warning] feature_fraction is set=0.6, colsample_bytree=0.7 will be ignored. Current value: feature_fraction=0.6

[LightGBM] [Info] Number of positive: 20499, number of negative: 276007

[LightGBM] [Info] Auto-choosing row-wise multi-threading, the overhead of testing was 0.045445 seconds.
You can set `force_row_wise=true` to remove the overhead.
And if memory is not enough, you can set `force_col_wise=true`.

[LightGBM] [Info] Total Bins 13726

[LightGBM] [Info] Number of data points in the train set: 296506, number of used features: 35

[LightGBM] [Info] [binary:BoostFromScore]: pavg=0.069135 -> initscore=-2.600050

[LightGBM] [Info] Start training from score -2.600050

[LightGBM] [Warning] bagging_freq is set=3, subsample_freq=0 will be ignored.
Current value: bagging_freq=3

[LightGBM] [Warning] feature_fraction is set=1.0, colsample_bytree=0.7 will be ignored. Current value: feature_fraction=1.0

[LightGBM] [Warning] bagging_fraction is set=0.8, subsample=0.7 will be ignored.
Current value: bagging_fraction=0.8

```

[LightGBM] [Warning] bagging_freq is set=3, subsample_freq=0 will be ignored.
Current value: bagging_freq=3
[LightGBM] [Warning] feature_fraction is set=1.0, colsample_bytree=0.7 will be
ignored. Current value: feature_fraction=1.0
[LightGBM] [Warning] bagging_fraction is set=0.8, subsample=0.7 will be ignored.
Current value: bagging_fraction=0.8
[LightGBM] [Warning] bagging_freq is set=3, subsample_freq=0 will be ignored.
Current value: bagging_freq=3
[LightGBM] [Warning] feature_fraction is set=1.0, colsample_bytree=0.7 will be
ignored. Current value: feature_fraction=1.0
[LightGBM] [Warning] bagging_fraction is set=0.8, subsample=0.7 will be ignored.
Current value: bagging_fraction=0.8
[LightGBM] [Warning] bagging_freq is set=3, subsample_freq=0 will be ignored.
Current value: bagging_freq=3
[LightGBM] [Warning] bagging_fraction is set=1.0, subsample=0.7 will be ignored.
Current value: bagging_fraction=1.0
[LightGBM] [Warning] feature_fraction is set=0.6, colsample_bytree=0.7 will be
ignored. Current value: feature_fraction=0.6
[LightGBM] [Warning] bagging_freq is set=1, subsample_freq=0 will be ignored.
Current value: bagging_freq=1
[LightGBM] [Warning] bagging_fraction is set=0.6, subsample=0.9 will be ignored.
Current value: bagging_fraction=0.6
[LightGBM] [Warning] feature_fraction is set=0.6, colsample_bytree=0.7 will be
ignored. Current value: feature_fraction=0.6
[LightGBM] [Warning] Categorical features with more bins than the configured
maximum bin number found.
[LightGBM] [Warning] For categorical features, max_bin and max_bin_by_feature
may be ignored with a large number of categories.
[LightGBM] [Warning] Categorical features with more bins than the configured
maximum bin number found.
[LightGBM] [Warning] For categorical features, max_bin and max_bin_by_feature
may be ignored with a large number of categories.
[LightGBM] [Warning] bagging_freq is set=3, subsample_freq=0 will be ignored.
Current value: bagging_freq=3
[LightGBM] [Warning] bagging_fraction is set=1.0, subsample=0.7 will be ignored.
Current value: bagging_fraction=1.0
[LightGBM] [Warning] feature_fraction is set=0.6, colsample_bytree=0.7 will be
ignored. Current value: feature_fraction=0.6
[LightGBM] [Info] Number of positive: 20499, number of negative: 276007
[LightGBM] [Info] Auto-choosing row-wise multi-threading, the overhead of
testing was 0.007152 seconds.
You can set `force_row_wise=true` to remove the overhead.
And if memory is not enough, you can set `force_col_wise=true`.
[LightGBM] [Info] Total Bins 13769
[LightGBM] [Info] Number of data points in the train set: 296506, number of used
features: 35
[LightGBM] [Info] [binary:BoostFromScore]: pavg=0.069135 -> initscore=-2.600050
[LightGBM] [Info] Start training from score -2.600050

```

```

[LightGBM] [Warning] bagging_freq is set=1, subsample_freq=0 will be ignored.
Current value: bagging_freq=1
[LightGBM] [Warning] bagging_fraction is set=0.6, subsample=0.9 will be ignored.
Current value: bagging_fraction=0.6
[LightGBM] [Warning] feature_fraction is set=0.6, colsample_bytree=0.7 will be
ignored. Current value: feature_fraction=0.6
[LightGBM] [Info] Number of positive: 20499, number of negative: 276006
[LightGBM] [Info] Auto-choosing row-wise multi-threading, the overhead of
testing was 0.005944 seconds.
You can set `force_row_wise=true` to remove the overhead.
And if memory is not enough, you can set `force_col_wise=true`.
[LightGBM] [Info] Total Bins 13211
[LightGBM] [Info] Number of data points in the train set: 296505, number of used
features: 35
[LightGBM] [Info] [binary:BoostFromScore]: pavg=0.069135 -> initscore=-2.600047
[LightGBM] [Info] Start training from score -2.600047
[LightGBM] [Warning] bagging_freq is set=1, subsample_freq=0 will be ignored.
Current value: bagging_freq=1
[LightGBM] [Warning] bagging_fraction is set=0.6, subsample=0.9 will be ignored.
Current value: bagging_fraction=0.6
[LightGBM] [Warning] feature_fraction is set=0.6, colsample_bytree=0.7 will be
ignored. Current value: feature_fraction=0.6
[LightGBM] [Warning] Categorical features with more bins than the configured
maximum bin number found.
[LightGBM] [Warning] For categorical features, max_bin and max_bin_by_feature
may be ignored with a large number of categories.
[LightGBM] [Warning] bagging_freq is set=1, subsample_freq=0 will be ignored.
Current value: bagging_freq=1
[LightGBM] [Warning] bagging_fraction is set=0.6, subsample=0.9 will be ignored.
Current value: bagging_fraction=0.6
[LightGBM] [Warning] feature_fraction is set=0.6, colsample_bytree=0.7 will be
ignored. Current value: feature_fraction=0.6
[LightGBM] [Info] Number of positive: 20499, number of negative: 276006
[LightGBM] [Info] Auto-choosing row-wise multi-threading, the overhead of
testing was 0.026836 seconds.
You can set `force_row_wise=true` to remove the overhead.
And if memory is not enough, you can set `force_col_wise=true`.
[LightGBM] [Info] Total Bins 13359
[LightGBM] [Info] Number of data points in the train set: 296505, number of used
features: 35
[LightGBM] [Info] [binary:BoostFromScore]: pavg=0.069135 -> initscore=-2.600047
[LightGBM] [Info] Start training from score -2.600047
[LightGBM] [Warning] bagging_freq is set=1, subsample_freq=0 will be ignored.
Current value: bagging_freq=1
[LightGBM] [Warning] feature_fraction is set=0.6, colsample_bytree=0.7 will be
ignored. Current value: feature_fraction=0.6
[LightGBM] [Warning] bagging_fraction is set=0.6, subsample=0.9 will be ignored.
Current value: bagging_fraction=0.6

```

```

[LightGBM] [Warning] bagging_freq is set=1, subsample_freq=0 will be ignored.
Current value: bagging_freq=1
[LightGBM] [Warning] feature_fraction is set=0.6, colsample_bytree=0.7 will be
ignored. Current value: feature_fraction=0.6
[LightGBM] [Warning] bagging_fraction is set=0.6, subsample=0.9 will be ignored.
Current value: bagging_fraction=0.6
[LightGBM] [Warning] bagging_freq is set=1, subsample_freq=0 will be ignored.
Current value: bagging_freq=1
[LightGBM] [Warning] bagging_fraction is set=0.6, subsample=0.9 will be ignored.
Current value: bagging_fraction=0.6
[LightGBM] [Warning] feature_fraction is set=0.6, colsample_bytree=0.7 will be
ignored. Current value: feature_fraction=0.6
[LightGBM] [Warning] Categorical features with more bins than the configured
maximum bin number found.
[LightGBM] [Warning] For categorical features, max_bin and max_bin_by_feature
may be ignored with a large number of categories.
[LightGBM] [Warning] bagging_freq is set=1, subsample_freq=0 will be ignored.
Current value: bagging_freq=1
[LightGBM] [Warning] bagging_fraction is set=0.6, subsample=0.9 will be ignored.
Current value: bagging_fraction=0.6
[LightGBM] [Warning] feature_fraction is set=0.6, colsample_bytree=0.7 will be
ignored. Current value: feature_fraction=0.6
[LightGBM] [Info] Number of positive: 20500, number of negative: 276006
[LightGBM] [Info] Auto-choosing row-wise multi-threading, the overhead of
testing was 0.024816 seconds.
You can set `force_row_wise=true` to remove the overhead.
And if memory is not enough, you can set `force_col_wise=true`.
[LightGBM] [Info] Total Bins 13717
[LightGBM] [Info] Number of data points in the train set: 296506, number of used
features: 35
[LightGBM] [Info] [binary:BoostFromScore]: pavg=0.069139 -> initscore=-2.599998
[LightGBM] [Info] Start training from score -2.599998
[LightGBM] [Warning] bagging_freq is set=1, subsample_freq=0 will be ignored.
Current value: bagging_freq=1
[LightGBM] [Warning] bagging_fraction is set=0.6, subsample=0.9 will be ignored.
Current value: bagging_fraction=0.6
[LightGBM] [Warning] feature_fraction is set=0.6, colsample_bytree=0.7 will be
ignored. Current value: feature_fraction=0.6
[LightGBM] [Warning] Categorical features with more bins than the configured
maximum bin number found.
[LightGBM] [Warning] For categorical features, max_bin and max_bin_by_feature
may be ignored with a large number of categories.
[LightGBM] [Warning] bagging_freq is set=1, subsample_freq=0 will be ignored.
Current value: bagging_freq=1
[LightGBM] [Warning] bagging_fraction is set=0.6, subsample=0.9 will be ignored.
Current value: bagging_fraction=0.6
[LightGBM] [Warning] feature_fraction is set=0.6, colsample_bytree=0.7 will be
ignored. Current value: feature_fraction=0.6

```

```

[LightGBM] [Info] Number of positive: 20499, number of negative: 276007
[LightGBM] [Info] Auto-choosing row-wise multi-threading, the overhead of
testing was 0.006607 seconds.
You can set `force_row_wise=true` to remove the overhead.
And if memory is not enough, you can set `force_col_wise=true`.
[LightGBM] [Info] Total Bins 13726
[LightGBM] [Info] Number of data points in the train set: 296506, number of used
features: 35
[LightGBM] [Info] [binary:BoostFromScore]: pavg=0.069135 -> initscore=-2.600050
[LightGBM] [Info] Start training from score -2.600050
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] bagging_freq is set=1, subsample_freq=0 will be ignored.
Current value: bagging_freq=1
[LightGBM] [Warning] feature_fraction is set=0.6, colsample_bytree=0.7 will be
ignored. Current value: feature_fraction=0.6
[LightGBM] [Warning] bagging_fraction is set=0.6, subsample=0.9 will be ignored.
Current value: bagging_fraction=0.6
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] bagging_freq is set=1, subsample_freq=0 will be ignored.
Current value: bagging_freq=1
[LightGBM] [Warning] feature_fraction is set=0.6, colsample_bytree=0.7 will be
ignored. Current value: feature_fraction=0.6
[LightGBM] [Warning] bagging_fraction is set=0.6, subsample=0.9 will be ignored.
Current value: bagging_fraction=0.6
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] bagging_freq is set=1, subsample_freq=0 will be ignored.
Current value: bagging_freq=1
[LightGBM] [Warning] bagging_fraction is set=0.6, subsample=0.9 will be ignored.
Current value: bagging_fraction=0.6
[LightGBM] [Warning] feature_fraction is set=0.6, colsample_bytree=0.7 will be
ignored. Current value: feature_fraction=0.6
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] Categorical features with more bins than the configured
maximum bin number found.
[LightGBM] [Warning] For categorical features, max_bin and max_bin_by_feature
may be ignored with a large number of categories.
[LightGBM] [Warning] bagging_freq is set=1, subsample_freq=0 will be ignored.
Current value: bagging_freq=1
[LightGBM] [Warning] bagging_fraction is set=0.6, subsample=0.9 will be ignored.
Current value: bagging_fraction=0.6
[LightGBM] [Warning] feature_fraction is set=0.6, colsample_bytree=0.7 will be
ignored. Current value: feature_fraction=0.6
[LightGBM] [Info] Number of positive: 20499, number of negative: 276007
[LightGBM] [Info] Auto-choosing row-wise multi-threading, the overhead of
testing was 0.166682 seconds.
You can set `force_row_wise=true` to remove the overhead.
And if memory is not enough, you can set `force_col_wise=true`.

```

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]


```

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] bagging_freq is set=1, subsample_freq=0 will be ignored.
Current value: bagging_freq=1
[LightGBM] [Warning] bagging_fraction is set=1.0, subsample=0.9 will be ignored.
Current value: bagging_fraction=1.0
[LightGBM] [Warning] feature_fraction is set=1.0, colsample_bytree=0.7 will be
ignored. Current value: feature_fraction=1.0
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] Categorical features with more bins than the configured
maximum bin number found.
[LightGBM] [Warning] For categorical features, max_bin and max_bin_by_feature
may be ignored with a large number of categories.
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] bagging_freq is set=1, subsample_freq=0 will be ignored.
Current value: bagging_freq=1
[LightGBM] [Warning] bagging_fraction is set=1.0, subsample=0.9 will be ignored.
Current value: bagging_fraction=1.0
[LightGBM] [Warning] feature_fraction is set=1.0, colsample_bytree=0.7 will be
ignored. Current value: feature_fraction=1.0
[LightGBM] [Info] Number of positive: 20499, number of negative: 276006
[LightGBM] [Info] Auto-choosing row-wise multi-threading, the overhead of
testing was 0.020459 seconds.
You can set `force_row_wise=true` to remove the overhead.
And if memory is not enough, you can set `force_col_wise=true`.
[LightGBM] [Info] Total Bins 13359
[LightGBM] [Info] Number of data points in the train set: 296505, number of used
features: 35
[LightGBM] [Info] [binary:BoostFromScore]: pavg=0.069135 -> initscore=-2.600047
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Info] Start training from score -2.600047

```

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]


```
[LightGBM] [Warning] bagging_fraction is set=1.0, subsample=0.9 will be ignored.
Current value: bagging_fraction=1.0
```

[illegible]

[illegible]


```

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] bagging_freq is set=1, subsample_freq=0 will be ignored.
Current value: bagging_freq=1
[LightGBM] [Warning] bagging_fraction is set=1.0, subsample=0.9 will be ignored.
Current value: bagging_fraction=1.0
[LightGBM] [Warning] feature_fraction is set=1.0, colsample_bytree=0.7 will be
ignored. Current value: feature_fraction=1.0
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] Categorical features with more bins than the configured
maximum bin number found.
[LightGBM] [Warning] For categorical features, max_bin and max_bin_by_feature
may be ignored with a large number of categories.
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

```



```
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] bagging_freq is set=3, subsample_freq=0 will be ignored.
Current value: bagging_freq=3
[LightGBM] [Warning] feature_fraction is set=0.8, colsample_bytree=0.7 will be
ignored. Current value: feature_fraction=0.8
[LightGBM] [Warning] bagging_fraction is set=1.0, subsample=0.9 will be ignored.
Current value: bagging_fraction=1.0
[LightGBM] [Warning] No further splits with positive gain, best gain:
-inf[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
```



```

[LightGBM] [Warning] feature_fraction is set=0.8, colsample_bytree=0.9 will be
ignored. Current value: feature_fraction=0.8
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] Categorical features with more bins than the configured
maximum bin number found.
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] For categorical features, max_bin and max_bin_by_feature
may be ignored with a large number of categories.
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] bagging_freq is set=1, subsample_freq=0 will be ignored.
Current value: bagging_freq=1
[LightGBM] [Warning] bagging_fraction is set=1.0, subsample=0.9 will be ignored.
Current value: bagging_fraction=1.0
[LightGBM] [Warning] feature_fraction is set=0.8, colsample_bytree=0.9 will be
ignored. Current value: feature_fraction=0.8
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] bagging_freq is set=1, subsample_freq=0 will be ignored.
Current value: bagging_freq=1
[LightGBM] [Warning] bagging_fraction is set=1.0, subsample=0.9 will be ignored.
Current value: bagging_fraction=1.0
[LightGBM] [Warning] feature_fraction is set=0.8, colsample_bytree=0.9 will be
ignored. Current value: feature_fraction=0.8
[LightGBM] [Info] Number of positive: 20499, number of negative: 276006
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Info] Auto-choosing row-wise multi-threading, the overhead of

```

```

testing was 0.006080 seconds.
You can set `force_row_wise=true` to remove the overhead.
And if memory is not enough, you can set `force_col_wise=true`.
[LightGBM] [Info] Total Bins 13211
[LightGBM] [Info] Number of data points in the train set: 296505, number of used
features: 35
[LightGBM] [Info] [binary:BoostFromScore]: pavg=0.069135 -> initscore=-2.600047
[LightGBM] [Info] Start training from score -2.600047
[LightGBM] [Warning] Categorical features with more bins than the configured
maximum bin number found.
[LightGBM] [Warning] For categorical features, max_bin and max_bin_by_feature
may be ignored with a large number of categories.
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] Categorical features with more bins than the configured
maximum bin number found.
[LightGBM] [Warning] For categorical features, max_bin and max_bin_by_feature
may be ignored with a large number of categories.
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] bagging_freq is set=1, subsample_freq=0 will be ignored.
Current value: bagging_freq=1
[LightGBM] [Warning] bagging_fraction is set=1.0, subsample=0.9 will be ignored.
Current value: bagging_fraction=1.0
[LightGBM] [Warning] feature_fraction is set=0.8, colsample_bytree=0.9 will be
ignored. Current value: feature_fraction=0.8
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Info] Number of positive: 20499, number of negative: 276006
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Info] Auto-choosing col-wise multi-threading, the overhead of
testing was 0.027874 seconds.
You can set `force_col_wise=true` to remove the overhead.
[LightGBM] [Info] Total Bins 13359
[LightGBM] [Info] Number of data points in the train set: 296505, number of used
features: 35
[LightGBM] [Info] [binary:BoostFromScore]: pavg=0.069135 -> initscore=-2.600047
[LightGBM] [Info] Start training from score -2.600047
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

```



```

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] bagging_freq is set=1, subsample_freq=0 will be ignored.
Current value: bagging_freq=1
[LightGBM] [Warning] bagging_fraction is set=1.0, subsample=0.9 will be ignored.
Current value: bagging_fraction=1.0
[LightGBM] [Warning] feature_fraction is set=0.8, colsample_bytree=0.9 will be
ignored. Current value: feature_fraction=0.8
[LightGBM] [Info] Number of positive: 20500, number of negative: 276006
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Info] Auto-choosing row-wise multi-threading, the overhead of
testing was 0.006063 seconds.
You can set `force_row_wise=true` to remove the overhead.
And if memory is not enough, you can set `force_col_wise=true`.
[LightGBM] [Info] Total Bins 13717
[LightGBM] [Info] Number of data points in the train set: 296506, number of used
features: 35
[LightGBM] [Info] [binary:BoostFromScore]: pavg=0.069139 -> initscore=-2.599998
[LightGBM] [Info] Start training from score -2.599998
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] bagging_freq is set=3, subsample_freq=0 will be ignored.
Current value: bagging_freq=3
[LightGBM] [Warning] feature_fraction is set=0.6, colsample_bytree=0.7 will be
ignored. Current value: feature_fraction=0.6
[LightGBM] [Warning] bagging_fraction is set=1.0, subsample=0.7 will be ignored.
Current value: bagging_fraction=1.0
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

```


[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]


```

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Info] Number of positive: 20499, number of negative: 276007
[LightGBM] [Warning] bagging_freq is set=1, subsample_freq=0 will be ignored.
Current value: bagging_freq=1[LightGBM] [Warning] No further splits with
positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] bagging_fraction is set=1.0, subsample=0.9 will be ignored.
Current value: bagging_fraction=1.0
[LightGBM] [Warning] feature_fraction is set=0.8, colsample_bytree=0.9 will be
ignored. Current value: feature_fraction=0.8
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Info] Number of positive: 20499, number of negative: 276007
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Info] Auto-choosing row-wise multi-threading, the overhead of
testing was 0.025455 seconds.
You can set `force_row_wise=true` to remove the overhead.
And if memory is not enough, you can set `force_col_wise=true`.
[LightGBM] [Info] Total Bins 13769
[LightGBM] [Info] Number of data points in the train set: 296506, number of used
features: 35
[LightGBM] [Info] [binary:BoostFromScore]: pavg=0.069135 -> initscore=-2.600050
[LightGBM] [Info] Start training from score -2.600050
[LightGBM] [Info] Auto-choosing row-wise multi-threading, the overhead of
testing was 0.143687 seconds.
You can set `force_row_wise=true` to remove the overhead.
And if memory is not enough, you can set `force_col_wise=true`.
[LightGBM] [Info] Total Bins 13726
[LightGBM] [Info] Number of data points in the train set: 296506, number of used
features: 35
[LightGBM] [Info] [binary:BoostFromScore]: pavg=0.069135 -> initscore=-2.600050
[LightGBM] [Info] Start training from score -2.600050
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

```

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]


```

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] bagging_freq is set=3, subsample_freq=0 will be ignored.
Current value: bagging_freq=3
[LightGBM] [Warning] bagging_fraction is set=1.0, subsample=0.9 will be ignored.
Current value: bagging_fraction=1.0
[LightGBM] [Warning] feature_fraction is set=0.6, colsample_bytree=0.7 will be
ignored. Current value: feature_fraction=0.6
[LightGBM] [Info] Number of positive: 20499, number of negative: 276006
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Info] Auto-choosing row-wise multi-threading, the overhead of
testing was 0.081098 seconds.
You can set `force_row_wise=true` to remove the overhead.
And if memory is not enough, you can set `force_col_wise=true`.
[LightGBM] [Info] Total Bins 13211
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Info] Number of data points in the train set: 296505, number of used
features: 35
[LightGBM] [Info] [binary:BoostFromScore]: pavg=0.069135 -> initscore=-2.600047
[LightGBM] [Info] Start training from score -2.600047
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

```



```

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] Categorical features with more bins than the configured
maximum bin number found.
[LightGBM] [Warning] For categorical features, max_bin and max_bin_by_feature
may be ignored with a large number of categories.
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] bagging_freq is set=3, subsample_freq=0 will be ignored.
Current value: bagging_freq=3
[LightGBM] [Warning] bagging_fraction is set=1.0, subsample=0.9 will be ignored.
Current value: bagging_fraction=1.0
[LightGBM] [Warning] feature_fraction is set=0.6, colsample_bytree=0.7 will be
ignored. Current value: feature_fraction=0.6

```


[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

```
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] bagging_freq is set=3, subsample_freq=0 will be ignored.
Current value: bagging_freq=3
[LightGBM] [Warning] bagging_fraction is set=1.0, subsample=0.9 will be ignored.
Current value: bagging_fraction=1.0
[LightGBM] [Warning] feature_fraction is set=0.6, colsample_bytree=0.7 will be
ignored. Current value: feature_fraction=0.6
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] bagging_freq is set=1, subsample_freq=0 will be ignored.
Current value: bagging_freq=1
[LightGBM] [Warning] feature_fraction is set=0.8, colsample_bytree=0.9 will be
ignored. Current value: feature_fraction=0.8
[LightGBM] [Warning] bagging_fraction is set=1.0, subsample=0.9 will be ignored.
Current value: bagging_fraction=1.0
[LightGBM] [Warning] No further splits with positive gain, best gain:
-inf[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
```

```

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] Categorical features with more bins than the configured
maximum bin number found.
[LightGBM] [Warning] For categorical features, max_bin and max_bin_by_feature
may be ignored with a large number of categories.
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] bagging_freq is set=3, subsample_freq=0 will be ignored.
Current value: bagging_freq=3
[LightGBM] [Warning] bagging_fraction is set=1.0, subsample=0.9 will be ignored.
Current value: bagging_fraction=1.0
[LightGBM] [Warning] feature_fraction is set=0.6, colsample_bytree=0.7 will be
ignored. Current value: feature_fraction=0.6
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] bagging_freq is set=3, subsample_freq=0 will be ignored.
Current value: bagging_freq=3
[LightGBM] [Warning] bagging_fraction is set=1.0, subsample=0.9 will be ignored.
Current value: bagging_fraction=1.0
[LightGBM] [Warning] feature_fraction is set=0.6, colsample_bytree=0.7 will be
ignored. Current value: feature_fraction=0.6
[LightGBM] [Info] Number of positive: 20499, number of negative: 276007
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Info] Auto-choosing row-wise multi-threading, the overhead of
testing was 0.015419 seconds.
You can set `force_row_wise=true` to remove the overhead.
And if memory is not enough, you can set `force_col_wise=true`.
[LightGBM] [Info] Total Bins 13726
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Info] Number of data points in the train set: 296506, number of used
features: 35
[LightGBM] [Info] [binary:BoostFromScore]: pavg=0.069135 -> initscore=-2.600050
[LightGBM] [Info] Start training from score -2.600050
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

```

```

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] Categorical features with more bins than the configured
maximum bin number found.
[LightGBM] [Warning] For categorical features, max_bin and max_bin_by_feature
may be ignored with a large number of categories.
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] bagging_freq is set=3, subsample_freq=0 will be ignored.
Current value: bagging_freq=3
[LightGBM] [Warning] bagging_fraction is set=1.0, subsample=0.9 will be ignored.
Current value: bagging_fraction=1.0
[LightGBM] [Warning] feature_fraction is set=0.6, colsample_bytree=0.7 will be
ignored. Current value: feature_fraction=0.6
[LightGBM] [Info] Number of positive: 20499, number of negative: 276007
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Info] Auto-choosing col-wise multi-threading, the overhead of
testing was 0.108330 seconds.
You can set `force_col_wise=true` to remove the overhead.
[LightGBM] [Info] Total Bins 13769
[LightGBM] [Info] Number of data points in the train set: 296506, number of used
features: 35
[LightGBM] [Info] [binary:BoostFromScore]: pavg=0.069135 -> initscore=-2.600050
[LightGBM] [Info] Start training from score -2.600050
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

```

[illegible]

```

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] bagging_freq is set=3, subsample_freq=0 will be ignored.
Current value: bagging_freq=3
[LightGBM] [Warning] bagging_fraction is set=1.0, subsample=0.9 will be ignored.
Current value: bagging_fraction=1.0
[LightGBM] [Warning] feature_fraction is set=1.0, colsample_bytree=0.7 will be
ignored. Current value: feature_fraction=1.0
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] Categorical features with more bins than the configured
maximum bin number found.
[LightGBM] [Warning] For categorical features, max_bin and max_bin_by_feature
may be ignored with a large number of categories.
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] bagging_freq is set=3, subsample_freq=0 will be ignored.
Current value: bagging_freq=3
[LightGBM] [Warning] bagging_fraction is set=1.0, subsample=0.9 will be ignored.
Current value: bagging_fraction=1.0
[LightGBM] [Warning] feature_fraction is set=1.0, colsample_bytree=0.7 will be
ignored. Current value: feature_fraction=1.0
[LightGBM] [Info] Number of positive: 20499, number of negative: 276006

```

[illegible]

ignored. Current value: feature_fraction=1.0

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] Categorical features with more bins than the configured maximum bin number found.

[LightGBM] [Warning] For categorical features, max_bin and max_bin_by_feature may be ignored with a large number of categories.

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] bagging_freq is set=3, subsample_freq=0 will be ignored.

Current value: bagging_freq=3

[LightGBM] [Warning] bagging_fraction is set=1.0, subsample=0.9 will be ignored.

Current value: bagging_fraction=1.0

[LightGBM] [Warning] feature_fraction is set=1.0, colsample_bytree=0.7 will be ignored. Current value: feature_fraction=1.0

[LightGBM] [Info] Number of positive: 20499, number of negative: 276006

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Info] Auto-choosing row-wise multi-threading, the overhead of testing was 0.195587 seconds.

You can set `force_row_wise=true` to remove the overhead.
And if memory is not enough, you can set `force_col_wise=true`.

```
[LightGBM] [Info] Total Bins 13359
[LightGBM] [Info] Number of data points in the train set: 296505, number of used
features: 35
[LightGBM] [Info] [binary:BoostFromScore]: pavg=0.069135 -> initscore=-2.600047
[LightGBM] [Info] Start training from score -2.600047
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] bagging_freq is set=3, subsample_freq=0 will be ignored.
Current value: bagging_freq=3
[LightGBM] [Warning] bagging_fraction is set=1.0, subsample=0.9 will be ignored.
Current value: bagging_fraction=1.0
[LightGBM] [Warning] feature_fraction is set=1.0, colsample_bytree=0.7 will be
ignored. Current value: feature_fraction=1.0
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] Categorical features with more bins than the configured
maximum bin number found.
[LightGBM] [Warning] For categorical features, max_bin and max_bin_by_feature
may be ignored with a large number of categories.
```

```

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] bagging_freq is set=3, subsample_freq=0 will be ignored.
Current value: bagging_freq=3
[LightGBM] [Warning] bagging_fraction is set=1.0, subsample=0.9 will be ignored.
Current value: bagging_fraction=1.0
[LightGBM] [Warning] feature_fraction is set=1.0, colsample_bytree=0.7 will be
ignored. Current value: feature_fraction=1.0
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] bagging_freq is set=3, subsample_freq=0 will be ignored.
Current value: bagging_freq=3
[LightGBM] [Warning] bagging_fraction is set=1.0, subsample=0.9 will be ignored.
Current value: bagging_fraction=1.0
[LightGBM] [Warning] feature_fraction is set=1.0, colsample_bytree=0.7 will be
ignored. Current value: feature_fraction=1.0
[LightGBM] [Info] Number of positive: 20500, number of negative: 276006
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Info] Auto-choosing col-wise multi-threading, the overhead of
testing was 0.102598 seconds.
You can set `force_col_wise=true` to remove the overhead.
[LightGBM] [Info] Total Bins 13717
[LightGBM] [Info] Number of data points in the train set: 296506, number of used
features: 35
[LightGBM] [Info] [binary:BoostFromScore]: pavg=0.069139 -> initscore=-2.599998
[LightGBM] [Info] Start training from score -2.599998
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] Categorical features with more bins than the configured
maximum bin number found.

```


[illegible]

[illegible]

[illegible]

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
 [LightGBM] [Warning] No further splits with positive gain, best gain: -inf
 [LightGBM] [Warning] No further splits with positive gain, best gain: -inf
 [LightGBM] [Warning] bagging_freq is set=1, subsample_freq=0 will be ignored.
 Current value: bagging_freq=1
 [LightGBM] [Warning] feature_fraction is set=1.0, colsample_bytree=0.7 will be ignored. Current value: feature_fraction=1.0
 [LightGBM] [Warning] bagging_fraction is set=1.0, subsample=0.9 will be ignored.
 Current value: bagging_fraction=1.0
 [LightGBM] [Warning] No further splits with positive gain, best gain: -inf
 [LightGBM] [Warning] No further splits with positive gain, best gain: -inf
 [LightGBM] [Warning] No further splits with positive gain, best gain: -inf
 [LightGBM] [Warning] No further splits with positive gain, best gain: -inf
 [LightGBM] [Warning] No further splits with positive gain, best gain: -inf
 [LightGBM] [Warning] No further splits with positive gain, best gain: -inf
 [LightGBM] [Warning] No further splits with positive gain, best gain: -inf
 [LightGBM] [Warning] bagging_freq is set=3, subsample_freq=0 will be ignored.
 Current value: bagging_freq=3
 [LightGBM] [Warning] bagging_fraction is set=1.0, subsample=0.9 will be ignored.
 Current value: bagging_fraction=1.0
 [LightGBM] [Warning] No further splits with positive gain, best gain: -inf
 [LightGBM] [Warning] feature_fraction is set=1.0, colsample_bytree=0.7 will be ignored. Current value: feature_fraction=1.0
 [LightGBM] [Warning] No further splits with positive gain, best gain: -inf
 [LightGBM] [Warning] bagging_freq is set=1, subsample_freq=0 will be ignored.
 Current value: bagging_freq=1
 [LightGBM] [Warning] bagging_fraction is set=0.6, subsample=0.9 will be ignored.
 Current value: bagging_fraction=0.6
 [LightGBM] [Warning] feature_fraction is set=0.6, colsample_bytree=0.9 will be ignored. Current value: feature_fraction=0.6
 [LightGBM] [Warning] No further splits with positive gain, best gain: -inf
 [LightGBM] [Warning] No further splits with positive gain, best gain: -inf
 [LightGBM] [Warning] No further splits with positive gain, best gain: -inf
 [LightGBM] [Warning] No further splits with positive gain, best gain: -inf
 [LightGBM] [Warning] Categorical features with more bins than the configured maximum bin number found.
 [LightGBM] [Warning] For categorical features, max_bin and max_bin_by_feature may be ignored with a large number of categories.
 [LightGBM] [Warning] No further splits with positive gain, best gain: -inf
 [LightGBM] [Warning] No further splits with positive gain, best gain: -inf
 [LightGBM] [Warning] No further splits with positive gain, best gain: -inf
 [LightGBM] [Warning] Categorical features with more bins than the configured maximum bin number found.
 [LightGBM] [Warning] For categorical features, max_bin and max_bin_by_feature may be ignored with a large number of categories.
 [LightGBM] [Warning] bagging_freq is set=3, subsample_freq=0 will be ignored.
 Current value: bagging_freq=3
 [LightGBM] [Warning] bagging_fraction is set=1.0, subsample=0.9 will be ignored.

```

Current value: bagging_fraction=1.0
[LightGBM] [Warning] feature_fraction is set=1.0, colsample_bytree=0.7 will be
ignored. Current value: feature_fraction=1.0
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Info] Number of positive: 20499, number of negative: 276007
[LightGBM] [Warning] bagging_freq is set=1, subsample_freq=0 will be ignored.
Current value: bagging_freq=1
[LightGBM] [Warning] bagging_fraction is set=0.6, subsample=0.9 will be ignored.
Current value: bagging_fraction=0.6
[LightGBM] [Warning] feature_fraction is set=0.6, colsample_bytree=0.9 will be
ignored. Current value: feature_fraction=0.6
[LightGBM] [Info] Number of positive: 20499, number of negative: 276006
[LightGBM] [Info] Auto-choosing row-wise multi-threading, the overhead of
testing was 0.048868 seconds.
You can set `force_row_wise=true` to remove the overhead.
And if memory is not enough, you can set `force_col_wise=true`.
[LightGBM] [Info] Total Bins 13769
[LightGBM] [Info] Number of data points in the train set: 296506, number of used
features: 35
[LightGBM] [Info] [binary:BoostFromScore]: pavg=0.069135 -> initscore=-2.600050
[LightGBM] [Info] Start training from score -2.600050
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Info] Auto-choosing row-wise multi-threading, the overhead of
testing was 0.164614 seconds.
You can set `force_row_wise=true` to remove the overhead.
And if memory is not enough, you can set `force_col_wise=true`.
[LightGBM] [Info] Total Bins 13211
[LightGBM] [Info] Number of data points in the train set: 296505, number of used
features: 35
[LightGBM] [Info] [binary:BoostFromScore]: pavg=0.069135 -> initscore=-2.600047
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Info] Start training from score -2.600047
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] bagging_freq is set=3, subsample_freq=0 will be ignored.
Current value: bagging_freq=3
[LightGBM] [Warning] feature_fraction is set=1.0, colsample_bytree=0.7 will be
ignored. Current value: feature_fraction=1.0
[LightGBM] [Warning] bagging_fraction is set=1.0, subsample=0.9 will be ignored.

```

```

Current value: bagging_fraction=1.0
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] bagging_freq is set=1, subsample_freq=0 will be ignored.
Current value: bagging_freq=1
[LightGBM] [Warning] bagging_fraction is set=0.6, subsample=0.9 will be ignored.
Current value: bagging_fraction=0.6
[LightGBM] [Warning] feature_fraction is set=0.6, colsample_bytree=0.9 will be
ignored. Current value: feature_fraction=0.6
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] Categorical features with more bins than the configured
maximum bin number found.
[LightGBM] [Warning] For categorical features, max_bin and max_bin_by_feature
may be ignored with a large number of categories.
[LightGBM] [Warning] bagging_freq is set=1, subsample_freq=0 will be ignored.
Current value: bagging_freq=1
[LightGBM] [Warning] bagging_fraction is set=0.6, subsample=0.9 will be ignored.
Current value: bagging_fraction=0.6
[LightGBM] [Warning] feature_fraction is set=0.6, colsample_bytree=0.9 will be
ignored. Current value: feature_fraction=0.6
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] bagging_freq is set=1, subsample_freq=0 will be ignored.
Current value: bagging_freq=1
[LightGBM] [Warning] bagging_fraction is set=0.6, subsample=0.9 will be ignored.
Current value: bagging_fraction=0.6
[LightGBM] [Warning] feature_fraction is set=0.6, colsample_bytree=0.9 will be
ignored. Current value: feature_fraction=0.6
[LightGBM] [Info] Number of positive: 20499, number of negative: 276006
[LightGBM] [Info] Auto-choosing row-wise multi-threading, the overhead of
testing was 0.035174 seconds.
You can set `force_row_wise=true` to remove the overhead.
And if memory is not enough, you can set `force_col_wise=true`.
[LightGBM] [Info] Total Bins 13359
[LightGBM] [Info] Number of data points in the train set: 296505, number of used
features: 35
[LightGBM] [Info] [binary:BoostFromScore]: pavg=0.069135 -> initscore=-2.600047
[LightGBM] [Info] Start training from score -2.600047
[LightGBM] [Warning] Categorical features with more bins than the configured
maximum bin number found.
[LightGBM] [Warning] For categorical features, max_bin and max_bin_by_feature

```

may be ignored with a large number of categories.

```
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] bagging_freq is set=1, subsample_freq=0 will be ignored.
Current value: bagging_freq=1
[LightGBM] [Warning] bagging_fraction is set=0.6, subsample=0.9 will be ignored.
Current value: bagging_fraction=0.6
[LightGBM] [Warning] feature_fraction is set=0.6, colsample_bytree=0.9 will be
ignored. Current value: feature_fraction=0.6
[LightGBM] [Info] Number of positive: 20500, number of negative: 276006
[LightGBM] [Info] Auto-choosing row-wise multi-threading, the overhead of
testing was 0.007524 seconds.
You can set `force_row_wise=true` to remove the overhead.
And if memory is not enough, you can set `force_col_wise=true`.
[LightGBM] [Info] Total Bins 13717
[LightGBM] [Info] Number of data points in the train set: 296506, number of used
features: 35
[LightGBM] [Info] [binary:BoostFromScore]: pavg=0.069139 -> initscore=-2.599998
[LightGBM] [Info] Start training from score -2.599998
[LightGBM] [Warning] bagging_freq is set=1, subsample_freq=0 will be ignored.
Current value: bagging_freq=1
[LightGBM] [Warning] feature_fraction is set=1.0, colsample_bytree=0.7 will be
ignored. Current value: feature_fraction=1.0
[LightGBM] [Warning] bagging_fraction is set=1.0, subsample=0.9 will be ignored.
Current value: bagging_fraction=1.0
[LightGBM] [Warning] bagging_freq is set=3, subsample_freq=0 will be ignored.
Current value: bagging_freq=3
[LightGBM] [Warning] feature_fraction is set=1.0, colsample_bytree=0.7 will be
ignored. Current value: feature_fraction=1.0
[LightGBM] [Warning] bagging_fraction is set=1.0, subsample=0.9 will be ignored.
Current value: bagging_fraction=1.0
[LightGBM] [Warning] bagging_freq is set=1, subsample_freq=0 will be ignored.
Current value: bagging_freq=1
[LightGBM] [Warning] bagging_fraction is set=0.6, subsample=0.9 will be ignored.
Current value: bagging_fraction=0.6
[LightGBM] [Warning] feature_fraction is set=0.6, colsample_bytree=0.9 will be
ignored. Current value: feature_fraction=0.6
[LightGBM] [Warning] Categorical features with more bins than the configured
maximum bin number found.
[LightGBM] [Warning] For categorical features, max_bin and max_bin_by_feature
may be ignored with a large number of categories.
[LightGBM] [Warning] bagging_freq is set=1, subsample_freq=0 will be ignored.
Current value: bagging_freq=1
[LightGBM] [Warning] bagging_fraction is set=0.6, subsample=0.9 will be ignored.
Current value: bagging_fraction=0.6
[LightGBM] [Warning] feature_fraction is set=0.6, colsample_bytree=0.9 will be
```

ignored. Current value: feature_fraction=0.6

[LightGBM] [Info] Number of positive: 20499, number of negative: 276007

[LightGBM] [Info] Auto-choosing row-wise multi-threading, the overhead of testing was 0.024928 seconds.

You can set `force_row_wise=true` to remove the overhead.

And if memory is not enough, you can set `force_col_wise=true`.

[LightGBM] [Info] Total Bins 13726

[LightGBM] [Info] Number of data points in the train set: 296506, number of used features: 35

[LightGBM] [Info] [binary:BoostFromScore]: pavg=0.069135 -> initscore=-2.600050

[LightGBM] [Info] Start training from score -2.600050

[LightGBM] [Warning] bagging_freq is set=1, subsample_freq=0 will be ignored.

Current value: bagging_freq=1

[LightGBM] [Warning] bagging_fraction is set=0.6, subsample=0.9 will be ignored.

Current value: bagging_fraction=0.6

[LightGBM] [Warning] feature_fraction is set=0.6, colsample_bytree=0.9 will be ignored. Current value: feature_fraction=0.6

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] Categorical features with more bins than the configured maximum bin number found.

[LightGBM] [Warning] For categorical features, max_bin and max_bin_by_feature may be ignored with a large number of categories.

[LightGBM] [Warning] bagging_freq is set=1, subsample_freq=0 will be ignored.

Current value: bagging_freq=1

[LightGBM] [Warning] bagging_fraction is set=0.6, subsample=0.9 will be ignored.

Current value: bagging_fraction=0.6

[LightGBM] [Warning] feature_fraction is set=0.6, colsample_bytree=0.9 will be ignored. Current value: feature_fraction=0.6

[LightGBM] [Info] Number of positive: 20499, number of negative: 276007

[LightGBM] [Info] Auto-choosing row-wise multi-threading, the overhead of testing was 0.088306 seconds.

You can set `force_row_wise=true` to remove the overhead.

And if memory is not enough, you can set `force_col_wise=true`.

[LightGBM] [Info] Total Bins 13769

[LightGBM] [Info] Number of data points in the train set: 296506, number of used features: 35

[LightGBM] [Info] [binary:BoostFromScore]: pavg=0.069135 -> initscore=-2.600050

[LightGBM] [Info] Start training from score -2.600050

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] bagging_freq is set=3, subsample_freq=0 will be ignored.

Current value: bagging_freq=3

[LightGBM] [Warning] feature_fraction is set=1.0, colsample_bytree=0.7 will be ignored. Current value: feature_fraction=1.0

[LightGBM] [Warning] bagging_fraction is set=1.0, subsample=0.9 will be ignored.

Current value: bagging_fraction=1.0

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf


```

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] bagging_freq is set=5, subsample_freq=0 will be ignored.
Current value: bagging_freq=5
[LightGBM] [Warning] bagging_fraction is set=0.8, subsample=0.9 will be ignored.
Current value: bagging_fraction=0.8
[LightGBM] [Warning] feature_fraction is set=0.8, colsample_bytree=0.9 will be
ignored. Current value: feature_fraction=0.8
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] Categorical features with more bins than the configured
maximum bin number found.
[LightGBM] [Warning] For categorical features, max_bin and max_bin_by_feature
may be ignored with a large number of categories.
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] bagging_freq is set=5, subsample_freq=0 will be ignored.
Current value: bagging_freq=5
[LightGBM] [Warning] bagging_fraction is set=0.8, subsample=0.9 will be ignored.
Current value: bagging_fraction=0.8
[LightGBM] [Warning] feature_fraction is set=0.8, colsample_bytree=0.9 will be
ignored. Current value: feature_fraction=0.8
[LightGBM] [Info] Number of positive: 20499, number of negative: 276006
[LightGBM] [Info] Auto-choosing col-wise multi-threading, the overhead of
testing was 0.055728 seconds.
You can set `force_col_wise=true` to remove the overhead.
[LightGBM] [Info] Total Bins 13359
[LightGBM] [Info] Number of data points in the train set: 296505, number of used
features: 35
[LightGBM] [Info] [binary:BoostFromScore]: pavg=0.069135 -> initscore=-2.600047
[LightGBM] [Info] Start training from score -2.600047
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

```


[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]


```

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] bagging_freq is set=1, subsample_freq=0 will be ignored.
Current value: bagging_freq=1
[LightGBM] [Warning] feature_fraction is set=0.6, colsample_bytree=0.9 will be
ignored. Current value: feature_fraction=0.6
[LightGBM] [Warning] bagging_fraction is set=0.6, subsample=0.9 will be ignored.
Current value: bagging_fraction=0.6
[LightGBM] [Warning] bagging_freq is set=1, subsample_freq=0 will be ignored.
Current value: bagging_freq=1
[LightGBM] [Warning] feature_fraction is set=0.6, colsample_bytree=0.9 will be
ignored. Current value: feature_fraction=0.6
[LightGBM] [Warning] bagging_fraction is set=0.6, subsample=0.9 will be ignored.
Current value: bagging_fraction=0.6
[LightGBM] [Warning] bagging_freq is set=1, subsample_freq=0 will be ignored.
Current value: bagging_freq=1
[LightGBM] [Warning] feature_fraction is set=0.6, colsample_bytree=0.9 will be
ignored. Current value: feature_fraction=0.6
[LightGBM] [Warning] bagging_fraction is set=0.6, subsample=0.9 will be ignored.
Current value: bagging_fraction=0.6
[LightGBM] [Warning] bagging_freq is set=3, subsample_freq=0 will be ignored.
Current value: bagging_freq=3
[LightGBM] [Warning] feature_fraction is set=0.6, colsample_bytree=0.7 will be
ignored. Current value: feature_fraction=0.6
[LightGBM] [Warning] bagging_fraction is set=1.0, subsample=0.9 will be ignored.
Current value: bagging_fraction=1.0
[LightGBM] [Warning] bagging_freq is set=1, subsample_freq=0 will be ignored.
Current value: bagging_freq=1
[LightGBM] [Warning] bagging_fraction is set=0.6, subsample=0.9 will be ignored.
Current value: bagging_fraction=0.6
[LightGBM] [Warning] feature_fraction is set=0.8, colsample_bytree=0.9 will be
ignored. Current value: feature_fraction=0.8
[LightGBM] [Warning] Categorical features with more bins than the configured
maximum bin number found.
[LightGBM] [Warning] For categorical features, max_bin and max_bin_by_feature
may be ignored with a large number of categories.
[LightGBM] [Warning] bagging_freq is set=1, subsample_freq=0 will be ignored.
Current value: bagging_freq=1
[LightGBM] [Warning] bagging_fraction is set=0.6, subsample=0.9 will be ignored.
Current value: bagging_fraction=0.6
[LightGBM] [Warning] feature_fraction is set=0.8, colsample_bytree=0.9 will be
ignored. Current value: feature_fraction=0.8
[LightGBM] [Info] Number of positive: 25624, number of negative: 345008
[LightGBM] [Info] Auto-choosing row-wise multi-threading, the overhead of
testing was 0.006095 seconds.
You can set `force_row_wise=true` to remove the overhead.

```

And if memory is not enough, you can set `force_col_wise=true`.

[LightGBM] [Info] Total Bins 13368

[LightGBM] [Info] Number of data points in the train set: 370632, number of used features: 35

[LightGBM] [Info] [binary:BoostFromScore]: pavg=0.069136 -> initscore=-2.600038

[LightGBM] [Info] Start training from score -2.600038

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[LightGBM] [Warning] No further splits with positive gain, best gain: -inf

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]


```
[LightGBM] [Warning] bagging_freq is set=1, subsample_freq=0 will be ignored.
Current value: bagging_freq=1
[LightGBM] [Warning] feature_fraction is set=0.8, colsample_bytree=0.9 will be
ignored. Current value: feature_fraction=0.8
[LightGBM] [Warning] bagging_fraction is set=0.6, subsample=0.9 will be ignored.
Current value: bagging_fraction=0.6
```

```
=== LightGBM_RS ===
```

```
AUC: 0.5648
```

```
LogLoss: 0.6197
```

```
Best F1 threshold: 0.460
```

```
Best F1: 0.1311
```

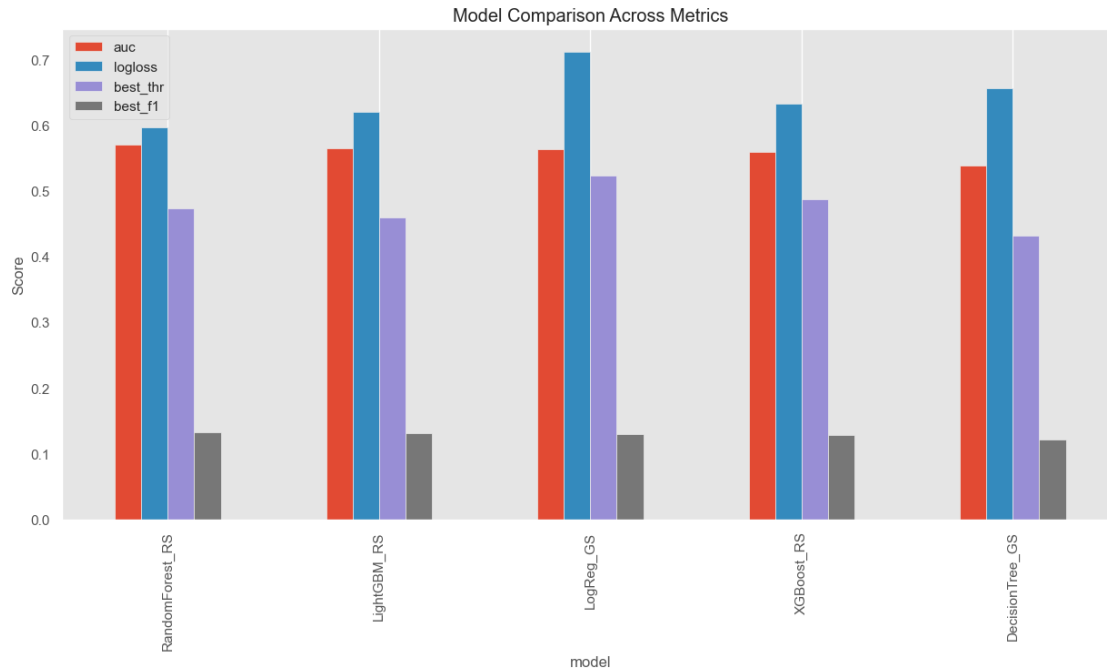
	precision	recall	f1-score	support
0	0.95	0.57	0.71	86952
1	0.07	0.53	0.13	5707
accuracy			0.57	92659
macro avg	0.51	0.55	0.42	92659
weighted avg	0.89	0.57	0.68	92659

```
[48]: results = pd.DataFrame([res_lgb, res_xgb, res_lr, res_rf, res_dt])
print("\nModel comparison (sorted by AUC):")
print(results.sort_values(by=["auc", "logloss", 'best_f1'], ascending=False))
```

```
Model comparison (sorted by AUC):
```

	model	auc	logloss	best_thr	best_f1
3	RandomForest_RS	0.57	0.60	0.47	0.13
0	LightGBM_RS	0.56	0.62	0.46	0.13
2	LogReg_GS	0.56	0.71	0.52	0.13
1	XGBoost_RS	0.56	0.63	0.49	0.13
4	DecisionTree_GS	0.54	0.66	0.43	0.12

```
[54]: results.sort_values(by=["auc", "logloss", 'best_f1'], ascending=False).
      ↪set_index('model').plot(kind="bar", figsize=(15,7))
plt.title("Model Comparison Across Metrics")
plt.ylabel("Score")
plt.grid(axis='y')
plt.show()
```



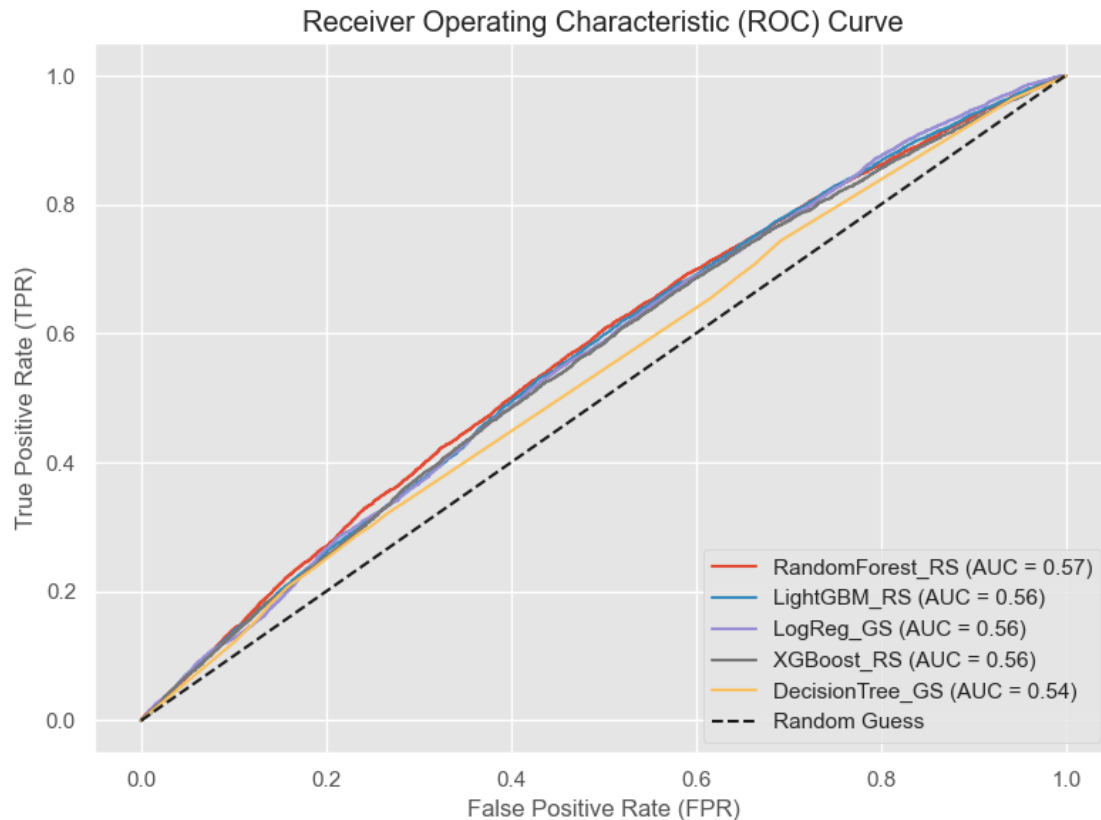
```
[55]: model_probas = {
    "RandomForest_RS": y_val_proba_rf,
    "LightGBM_RS": y_val_proba_lgb,
    "LogReg_GS": y_val_proba_lr,
    "XGBoost_RS": y_val_proba_xgb,
    "DecisionTree_GS": y_val_proba_dt,
}

plt.figure(figsize=(8, 6))

for name, y_proba in model_probas.items():
    fpr, tpr, _ = roc_curve(y_val, y_proba)
    model_auc = auc(fpr, tpr)
    plt.plot(fpr, tpr, label=f"{name} (AUC = {model_auc:.2f})")

# random baseline
plt.plot([0, 1], [0, 1], 'k--', label="Random Guess")

plt.title("Receiver Operating Characteristic (ROC) Curve")
plt.xlabel("False Positive Rate (FPR)")
plt.ylabel("True Positive Rate (TPR)")
plt.legend(loc="lower right")
plt.grid(True)
plt.tight_layout()
plt.show()
```



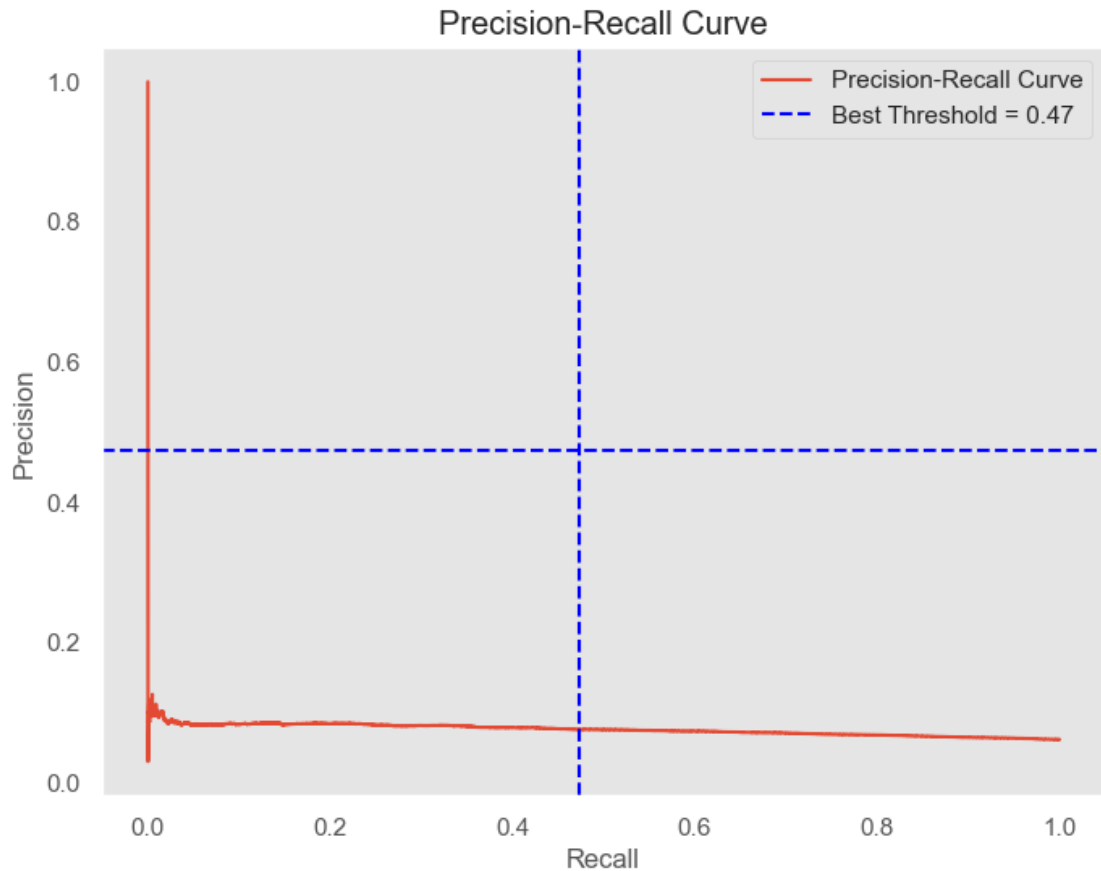
- All models are tightly clustered: AUC is 0.54–0.57 and best F1 0.12–0.13 for every model, so no algorithm clearly dominates.
- RandomForest_RS is marginally best (AUC 0.57, logloss 0.60), with LightGBM_RS, XGBoost_RS, and LogReg_GS just behind; DecisionTree_GS is clearly weakest.
- This consistency across very different models indicates that current features and time-based validation limit achievable performance; further gains will come more from richer user/behavior features than from changing model family.

RandomForest Classifier Precision Recall Curve plot and F1 score plot

```
[64]: y_prob = best_rf.predict_proba(X_val_tree)[: ,1] # Use LGBM or XGB
precision, recall, thresholds = precision_recall_curve(y_val, y_prob)

plt.figure(figsize=(8,6))
plt.plot(recall, precision, label='Precision-Recall Curve')
plt.axvline(res_rf['best_thr'], color='blue', linestyle='--', label=f'Best_
↳Threshold = {np.round(res_rf['best_thr'],2)}')
plt.axhline(res_rf['best_thr'], color='blue', linestyle='--')
plt.xlabel("Recall")
plt.ylabel("Precision")
plt.title("Precision-Recall Curve")
```

```
plt.grid()
plt.legend()
plt.show()
```

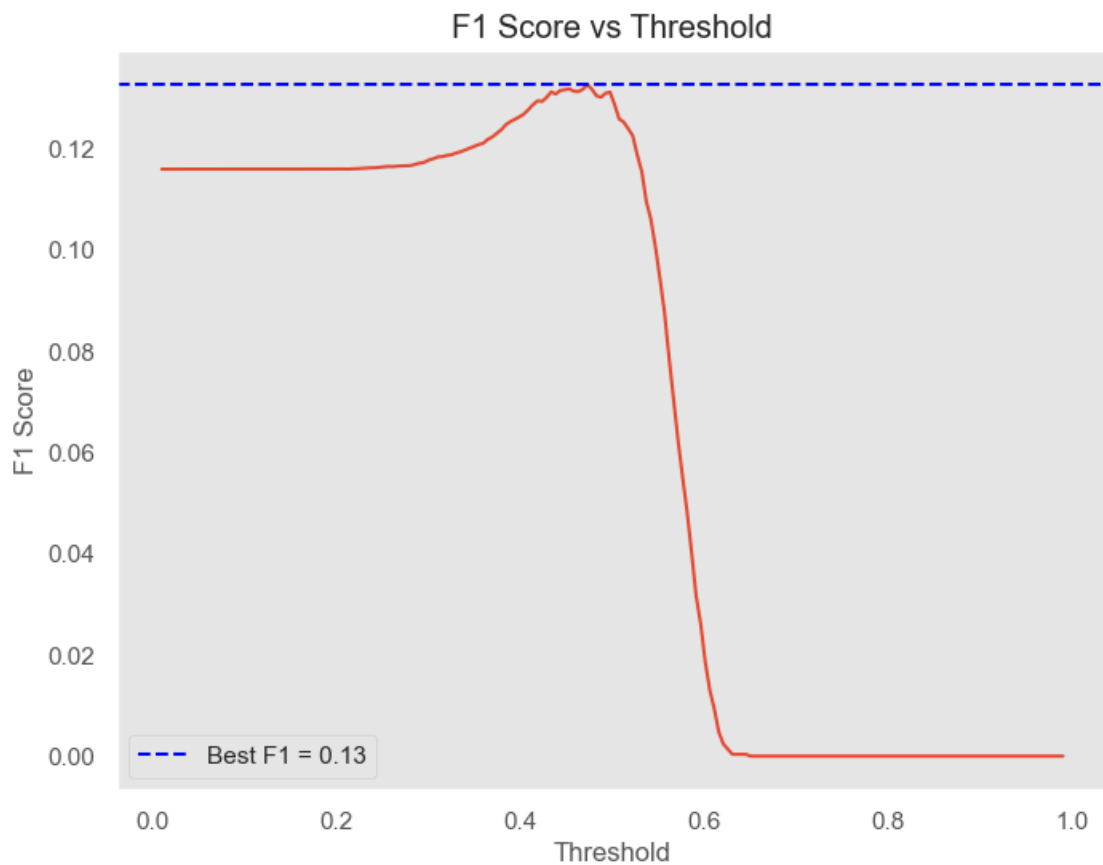


```
[67]: thresholds = np.linspace(0.01, 0.99, 200)
f1_scores = []

for t in thresholds:
    preds = (y_prob >= t).astype(int)
    f1_scores.append(f1_score(y_val, preds))

plt.figure(figsize=(8,6))
plt.plot(thresholds, f1_scores)
plt.axhline(res_rf['best_f1'], color='blue', linestyle='--', label=f'Best F1 = {
    np.round(res_rf['best_f1'],2)}')
plt.xlabel("Threshold")
plt.ylabel("F1 Score")
plt.title("F1 Score vs Threshold")
plt.legend()
```

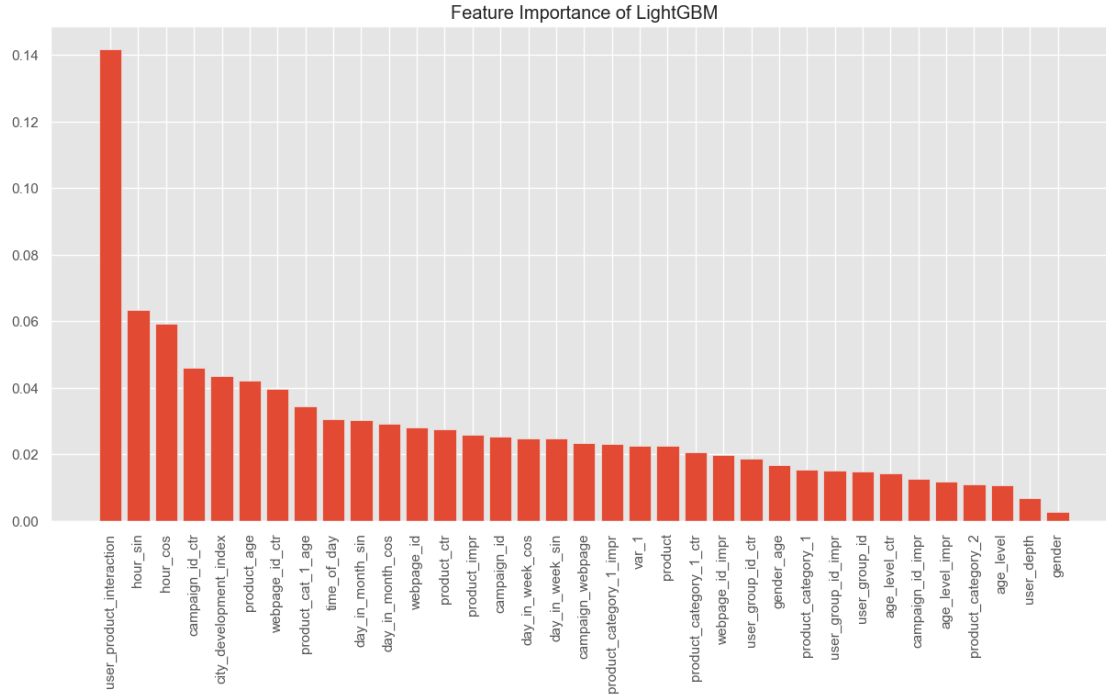
```
plt.grid()
plt.show()
```



Feature Importance

```
[69]: importances = best_rf.feature_importances_
indices = np.argsort(importances[::-1]) # Sort feature importances in
↳ descending order
names = [X_train.columns[i] for i in indices] # Rearrange feature names so they
↳ match the sorted feature importances

plt.figure(figsize=(15, 7)) # Create plot
plt.title("Feature Importance of LightGBM") # Create plot title
plt.bar(range(X_train.shape[1]), importances[indices]) # Add bars
plt.xticks(range(X_train.shape[1]), names, rotation=90) # Add feature names as
↳ x-axis labels
plt.show() # Show plot
```



3. Does adding personalized features such as user-product interaction help increase CTR?

- Yes. The very high importance of user_product_interaction shows that user-product history is one of the strongest drivers of predicted clicks, so personalized interaction features do improve CTR prediction.

4. Based on feature importance, which factors (e.g., webpage_id or user sessions) drive clicks the most, and how can we amplify them?

- The top drivers are user-product interaction, time (hour_sin/hour_cos, time_of_day), campaign/webpage CTRs, product_age interaction and city_development_index, so clicks concentrate where engaged users see relevant products at the right time on strong campaigns/pages; amplifying them means bidding more in those time windows and placements, and serving more impressions for high-CTR user-product and campaign-webpage pairs.

5. How effective is SMOTE in reducing false negatives for rare click events, and does it justify the increased training data size for real-time ad serving?

- For this dataset, class weighting already yields similar recall with simpler training, and SMOTE would increase data size, harm calibration, and add synthetic patterns, so it is unlikely to meaningfully reduce false negatives in a way that justifies the cost for real-time serving.

6. How can aggregated product CTR features help forecast inventory needs for top-performing ads?

- Aggregated product CTR features (product_ctr, product_impr) identify consistently high-performing items over time, which can be extrapolated to forecast future demand and

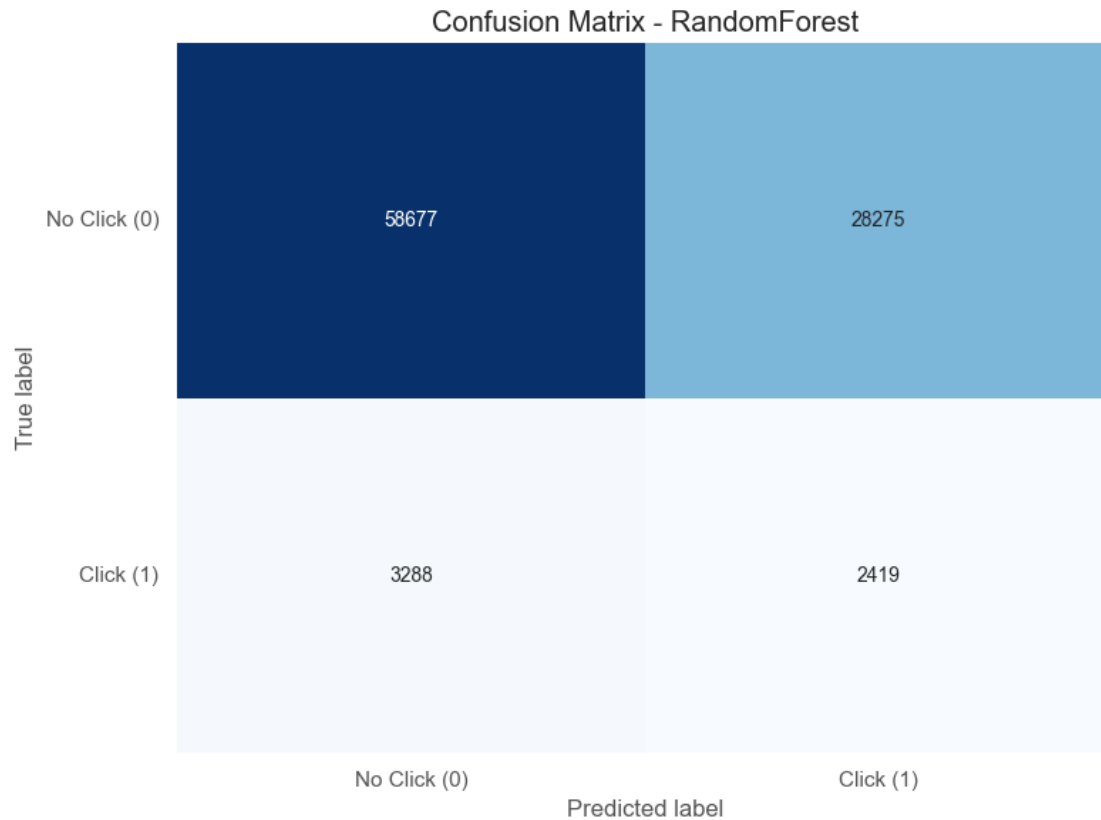
allocate inventory or budget toward those ads before they peak.

7. What user profiles (e.g., by age, gender, or city) show the highest click propensity, and how should we adjust bidding strategies?
 - User profiles defined by `age_level`, `gender`, and `city_development_index` have relatively low importance compared with behavioral and campaign features, so demographic segments only weakly affect click propensity, and bidding strategies should primarily prioritize high-CTR behaviors and contexts, using demographics as secondary fine-tuning rather than primary levers.

Confusion Matrix

```
[82]: y_prob = best_rf.predict_proba(X_val_tree)[:,-1]
y_pred = (y_prob >= res_rf['best_thr']).astype(int)
cm = confusion_matrix(y_val,y_pred)  # y_val_pred at best threshold

plt.figure(figsize=(8, 6))
sns.heatmap(
    cm,
    annot=True,
    fmt="d",
    cmap="Blues",
    cbar=False,
    linewidths=0.0,  # removes grid lines
    linecolor=None
)
plt.xlabel("Predicted label")
plt.ylabel("True label")
plt.title("Confusion Matrix - RandomForest")
plt.xticks(ticks=[0.5, 1.5], labels=["No Click (0)", "Click (1)"])
plt.yticks(ticks=[0.5, 1.5], labels=["No Click (0)", "Click (1)"], rotation=0)
plt.tight_layout()
plt.show()
```

- The model correctly rejects most non-clicks (58.7k TN) but still produces a large number of false positives (28.3k), which matches the low precision in your PR curve.
- It captures a reasonable share of true clicks (2419 TP vs 3238 FN), so recall is moderate, but overall F1 remains low because false positives dominate, reflecting the inherent difficulty and class imbalance in the dataset.

Testdata

- first loading the test data and need to do all the processing that we done for train data

```
[149]: test_df = pd.read_csv('./Ad_Click_predicition_test.csv')
test_df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 128858 entries, 0 to 128857
Data columns (total 14 columns):
#   Column                Non-Null Count  Dtype
---  -
0   session_id            128858 non-null int64
1   DateTime              128858 non-null object
2   user_id               128858 non-null int64
3   product              128858 non-null object
```

```

4  campaign_id          128858 non-null  int64
5  webpage_id           128858 non-null  int64
6  product_category_1   128858 non-null  int64
7  product_category_2   52687 non-null  float64
8  user_group_id        123174 non-null  float64
9  gender               123174 non-null  object
10 age_level            123174 non-null  float64
11 user_depth           123174 non-null  float64
12 city_development_index 94249 non-null  float64
13 var_1               128858 non-null  int64

```

dtypes: float64(5), int64(6), object(3)

memory usage: 13.8+ MB

```

[150]: def my_pipeline(df):

    # extracting required features from datetime columns
    df = extract_features_from_datetime(df, 'DateTime')

    # applying cyclic encoding for time features
    df = add_cyclic_features(df, cyclic_features)

    # mapping female to 0 and male to 1
    df['gender'] = df['gender'].map({'Female':0, 'Male':1})

    # adding interaction feature
    add_interaction_features(df)

    # replace nan values
    replace_nan(df)

    # merging aggregate feature of calculated from train data
    df = merge_agg_features(df, agg_tables, agg_entities, global_ctr)

    # feature selection
    drop_cols = ['session_id', 'DateTime']
    if 'is_click' in df.columns:
        drop_cols.append('is_click')
    if 'user_id' in df.columns:
        drop_cols.append('user_id')

    X_test = df[X_train.columns] # ensure same columns/order as training

    # encode categoricals as in RF training
    X_test_tree = X_test.copy()
    for col in cat_features:
        X_test_tree[col] = X_test_tree[col].astype('category')
        # align categories with training categories

```

```

X_test_tree[col] = X_test_tree[col].cat.set_categories(
    X_train[col].astype('category').cat.categories
)
X_test_tree[col] = X_test_tree[col].cat.codes

# ----- 6. Predict with RandomForest -----
proba = best_rf.predict_proba(X_test_tree)[: , 1]
preds = (proba >= res_rf['best_thr']).astype(int)

return proba, preds

```

```

[155]: proba, is_click = my_pipeline(test_df)
final_pred = pd.concat((test_df, pd.Series(is_click, name='is_click'), pd.
↳ Series(proba, name='click_probability')), axis=1)
print("CTR: ", np.round(final_pred.is_click.mean()*100, 2))

```

CTR: 37.48

```

[161]: final_pred.
↳ sort_values(by='click_probability', ascending=False)[['session_id', 'user_id', 'campaign_id', '
↳ head(5)

```

```

[161]:
    session_id  user_id  campaign_id  webpage_id
111166      262925  1045407         98970         6970
11821       264236   161867         98970         6970
10764       264589   217830         98970         6970
11985       463900  1139971         98970         6970
10721       273966   359288         98970         6970

```

- These are the users, campaign and webpage which are having high probability of click

[]:

[]:

[]:

[]:

[]:

[]:

[]:

[]:

[]:

[]: