Brian_Reppeto_DSC550_Week_12

May 31, 2024

0.0.1 DSC 550 Week :

Activity 6.2

Author: Brian Reppeto 4/19/2024

0.0.2 Milestone 1: Predicting Patient Readmissions for Diabetic Patients

Project Narrative

Hospital readmissions within 30 days post-discharge are not only a significant financial burden on the healthcare system but also often reflect suboptimal patient outcomes and potentially preventable complications. This project addresses the critical challenge of predicting such readmissions among diabetic patients, a group particularly prone to frequent and costly hospitalizations.

The Problem

The specific problem this project targets is the prediction of unplanned readmissions within 30 days among patients diagnosed with diabetes. These readmissions may be due to a variety of factors including inadequate management of diabetes, complications arising from the condition, or insufficient patient education and follow-up care upon discharge. The goal is to provide a predictive tool that can identify at-risk patients before they leave the hospital. This tool will enable healthcare providers to initiate targeted interventions such as personalized discharge planning, enhanced patient education, and tailored follow-up care schedules.

Objectives

The primary objective of this project is to develop a predictive model that uses historical hospital data to forecast the likelihood of a diabetic patient being readmitted within 30 days of discharge. The insights gained from this model will assist healthcare professionals in making informed decisions about patient care strategies and resource allocation.

Data Utilization

The project utilizes data from the Diabetes 130-US hospitals dataset, which comprises information from over 100,000 hospital admissions from 1999 to 2008 and across 130 US hospitals. The dataset includes diverse variables such as patient demographics, admission and discharge statuses, diagnostic codes, number of inpatient visits, and medication details. This rich dataset provides a comprehensive foundation to explore and model the complexities associated with readmissions.

Potential Impact

By accurately predicting readmissions, the model can directly influence the development of personalized medicine approaches and proactive healthcare strategies. Hospitals can use these predictions to reduce readmission rates, thereby decreasing the penalties they face under healthcare regulations like the Hospital Readmissions Reduction Program (HRRP). Moreover, patients benefit from improved healthcare experiences and outcomes, contributing to overall patient satisfaction and health system sustainability.

In conclusion, this project aims to harness the power of machine learning and predictive analytics to tackle a pressing healthcare issue. By doing so, it not only addresses an immediate business need for hospitals but also plays a crucial role in advancing how data-driven strategies can be implemented in clinical settings to enhance patient care.

[252]:
```
# import libraries

import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
from sklearn.model_selection import train_test_split, GridSearchCV
from sklearn.preprocessing import StandardScaler, LabelEncoder
from sklearn.linear_model import LogisticRegression
from sklearn.metrics import confusion_matrix, classification_report,␣
 ↪accuracy_score, roc_auc_score
from sklearn.ensemble import RandomForestClassifier
```

[253]:
```
# load the dataset

data = pd.read_csv('diabetic_data.csv')
```

[254]:
```
# data shape

data.shape
```

[254]:
```
(101766, 50)
```

[255]:
```
# head the data

data.head(15)
```

[255]:
```
   encounter_id  patient_nbr             race  gender      age weight  \
0       2278392      8222157        Caucasian  Female   [0-10)      ?
1        149190     55629189        Caucasian  Female  [10-20)      ?
2         64410     86047875  AfricanAmerican  Female  [20-30)      ?
3        500364     82442376        Caucasian    Male  [30-40)      ?
4         16680     42519267        Caucasian    Male  [40-50)      ?
5         35754     82637451        Caucasian    Male  [50-60)      ?
6         55842     84259809        Caucasian    Male  [60-70)      ?
7         63768    114882984        Caucasian    Male  [70-80)      ?
```

8	12522	48330783	Caucasian	Female	[80-90)	?
9	15738	63555939	Caucasian	Female	[90-100)	?
10	28236	89869032	AfricanAmerican	Female	[40-50)	?
11	36900	77391171	AfricanAmerican	Male	[60-70)	?
12	40926	85504905	Caucasian	Female	[40-50)	?
13	42570	77586282	Caucasian	Male	[80-90)	?
14	62256	49726791	AfricanAmerican	Female	[60-70)	?

	admission_type_id	discharge_disposition_id	admission_source_id \
0	6	25	1
1	1	1	7
2	1	1	7
3	1	1	7
4	1	1	7
5	2	1	2
6	3	1	2
7	1	1	7
8	2	1	4
9	3	3	4
10	1	1	7
11	2	1	4
12	1	3	7
13	1	6	7
14	3	1	2

	time_in_hospital	…	citoglipton	insulin	glyburide-metformin \
0	1	…	No	No	No
1	3	…	No	Up	No
2	2	…	No	No	No
3	2	…	No	Up	No
4	1	…	No	Steady	No
5	3	…	No	Steady	No
6	4	…	No	Steady	No
7	5	…	No	No	No
8	13	…	No	Steady	No
9	12	…	No	Steady	No
10	9	…	No	Steady	No
11	7	…	No	Steady	No
12	7	…	No	Down	No
13	10	…	No	Steady	No
14	1	…	No	Steady	No

	glipizide-metformin	glimepiride-pioglitazone	metformin-rosiglitazone \
0	No	No	No
1	No	No	No
2	No	No	No
3	No	No	No

```
4                    No                         No                        No
5                    No                         No                        No
6                    No                         No                        No
7                    No                         No                        No
8                    No                         No                        No
9                    No                         No                        No
10                   No                         No                        No
11                   No                         No                        No
12                   No                         No                        No
13                   No                         No                        No
14                   No                         No                        No

    metformin-pioglitazone  change diabetesMed readmitted
0                       No      No          No         NO
1                       No      Ch         Yes        >30
2                       No      No         Yes         NO
3                       No      Ch         Yes         NO
4                       No      Ch         Yes         NO
5                       No      No         Yes        >30
6                       No      Ch         Yes         NO
7                       No      No         Yes        >30
8                       No      Ch         Yes         NO
9                       No      Ch         Yes         NO
10                      No      No         Yes        >30
11                      No      Ch         Yes        <30
12                      No      Ch         Yes        <30
13                      No      No         Yes         NO
14                      No      No         Yes        >30

[15 rows x 50 columns]
```

0.0.3 Data Exploration and Cleaning

[256]:

```python
# explore missing values and clean data

data = data.replace('?', np.nan)  # replace '?' with NaN for clarity
missing_data = data.isnull().sum()/len(data) * 100
print("Percentage of missing data per column:\n", missing_data[missing_data >
 ↪0])
```

```
Percentage of missing data per column:
 race                  2.233555
weight               96.858479
payer_code           39.557416
medical_specialty    49.082208
diag_1                0.020636
diag_2                0.351787
diag_3                1.398306
```

```
max_glu_serum        94.746772
A1Cresult            83.277322
dtype: float64
```

[257]:
```python
# drop columns with high percentage of missing values and those not relevant␣
↪for the analysis

columns_to_drop = ['weight',  'medical_specialty','payer_code']
data.drop(columns=columns_to_drop, inplace=True)
```

[258]:
```python
# head the data

data.head(15)
```

[258]:

	encounter_id	patient_nbr	race	gender	age
0	2278392	8222157	Caucasian	Female	[0-10)
1	149190	55629189	Caucasian	Female	[10-20)
2	64410	86047875	AfricanAmerican	Female	[20-30)
3	500364	82442376	Caucasian	Male	[30-40)
4	16680	42519267	Caucasian	Male	[40-50)
5	35754	82637451	Caucasian	Male	[50-60)
6	55842	84259809	Caucasian	Male	[60-70)
7	63768	114882984	Caucasian	Male	[70-80)
8	12522	48330783	Caucasian	Female	[80-90)
9	15738	63555939	Caucasian	Female	[90-100)
10	28236	89869032	AfricanAmerican	Female	[40-50)
11	36900	77391171	AfricanAmerican	Male	[60-70)
12	40926	85504905	Caucasian	Female	[40-50)
13	42570	77586282	Caucasian	Male	[80-90)
14	62256	49726791	AfricanAmerican	Female	[60-70)

	admission_type_id	discharge_disposition_id	admission_source_id
0	6	25	1
1	1	1	7
2	1	1	7
3	1	1	7
4	1	1	7
5	2	1	2
6	3	1	2
7	1	1	7
8	2	1	4
9	3	3	4
10	1	1	7
11	2	1	4
12	1	3	7
13	1	6	7
14	3	1	2

	time_in_hospital	num_lab_procedures	…	citoglipton	insulin	\
0	1	41	…	No	No	
1	3	59	…	No	Up	
2	2	11	…	No	No	
3	2	44	…	No	Up	
4	1	51	…	No	Steady	
5	3	31	…	No	Steady	
6	4	70	…	No	Steady	
7	5	73	…	No	No	
8	13	68	…	No	Steady	
9	12	33	…	No	Steady	
10	9	47	…	No	Steady	
11	7	62	…	No	Steady	
12	7	60	…	No	Down	
13	10	55	…	No	Steady	
14	1	49	…	No	Steady	

	glyburide-metformin	glipizide-metformin	glimepiride-pioglitazone	\
0	No	No	No	
1	No	No	No	
2	No	No	No	
3	No	No	No	
4	No	No	No	
5	No	No	No	
6	No	No	No	
7	No	No	No	
8	No	No	No	
9	No	No	No	
10	No	No	No	
11	No	No	No	
12	No	No	No	
13	No	No	No	
14	No	No	No	

	metformin-rosiglitazone	metformin-pioglitazone	change	diabetesMed	\
0	No	No	No	No	
1	No	No	Ch	Yes	
2	No	No	No	Yes	
3	No	No	Ch	Yes	
4	No	No	Ch	Yes	
5	No	No	No	Yes	
6	No	No	Ch	Yes	
7	No	No	No	Yes	
8	No	No	Ch	Yes	
9	No	No	Ch	Yes	
10	No	No	No	Yes	

```
11                       No                      No      Ch            Yes
12                       No                      No      Ch            Yes
13                       No                      No      No            Yes
14                       No                      No      No            Yes

   readmitted
0          NO
1         >30
2          NO
3          NO
4          NO
5         >30
6          NO
7         >30
8          NO
9          NO
10        >30
11        <30
12        <30
13         NO
14        >30

[15 rows x 47 columns]
```

[259]:

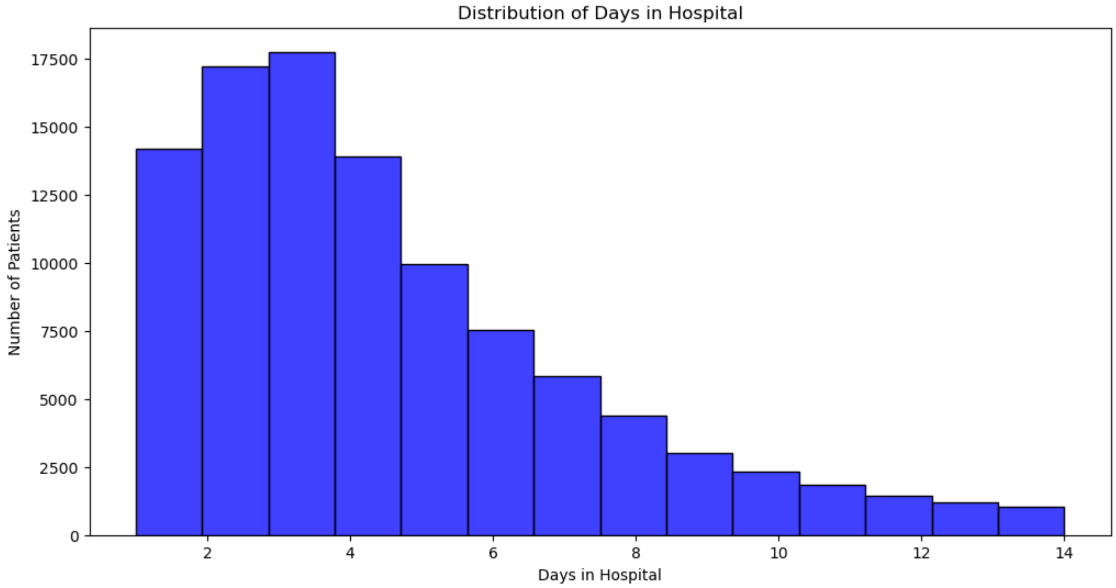
```
# summary of cleaned data

print(data.info())
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 101766 entries, 0 to 101765
Data columns (total 47 columns):
 #   Column                    Non-Null Count   Dtype
---  ------                    --------------   -----
 0   encounter_id              101766 non-null  int64
 1   patient_nbr               101766 non-null  int64
 2   race                      99493 non-null   object
 3   gender                    101766 non-null  object
 4   age                       101766 non-null  object
 5   admission_type_id         101766 non-null  int64
 6   discharge_disposition_id  101766 non-null  int64
 7   admission_source_id       101766 non-null  int64
 8   time_in_hospital          101766 non-null  int64
 9   num_lab_procedures        101766 non-null  int64
 10  num_procedures            101766 non-null  int64
 11  num_medications           101766 non-null  int64
 12  number_outpatient         101766 non-null  int64
 13  number_emergency          101766 non-null  int64
 14  number_inpatient          101766 non-null  int64
```

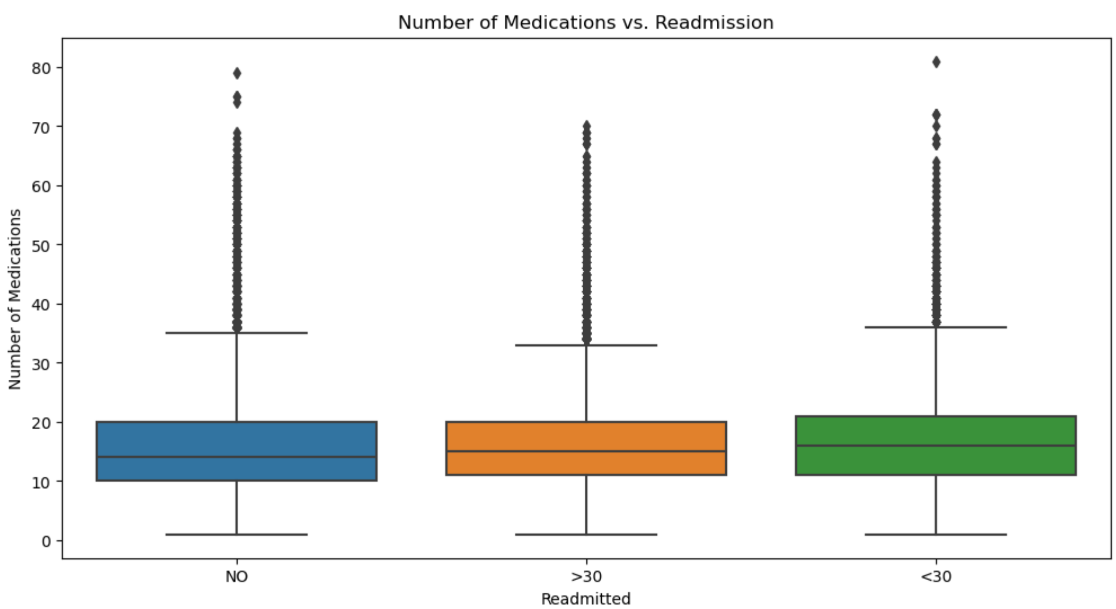
```
 15   diag_1                    101745 non-null  object
 16   diag_2                    101408 non-null  object
 17   diag_3                    100343 non-null  object
 18   number_diagnoses          101766 non-null  int64
 19   max_glu_serum             5346 non-null    object
 20   A1Cresult                 17018 non-null   object
 21   metformin                 101766 non-null  object
 22   repaglinide               101766 non-null  object
 23   nateglinide               101766 non-null  object
 24   chlorpropamide            101766 non-null  object
 25   glimepiride               101766 non-null  object
 26   acetohexamide             101766 non-null  object
 27   glipizide                 101766 non-null  object
 28   glyburide                 101766 non-null  object
 29   tolbutamide               101766 non-null  object
 30   pioglitazone              101766 non-null  object
 31   rosiglitazone             101766 non-null  object
 32   acarbose                  101766 non-null  object
 33   miglitol                  101766 non-null  object
 34   troglitazone              101766 non-null  object
 35   tolazamide                101766 non-null  object
 36   examide                   101766 non-null  object
 37   citoglipton               101766 non-null  object
 38   insulin                   101766 non-null  object
 39   glyburide-metformin       101766 non-null  object
 40   glipizide-metformin       101766 non-null  object
 41   glimepiride-pioglitazone  101766 non-null  object
 42   metformin-rosiglitazone   101766 non-null  object
 43   metformin-pioglitazone    101766 non-null  object
 44   change                    101766 non-null  object
 45   diabetesMed               101766 non-null  object
 46   readmitted                101766 non-null  object
dtypes: int64(13), object(34)
memory usage: 36.5+ MB
None
```

0.0.4 Graphical Analysis

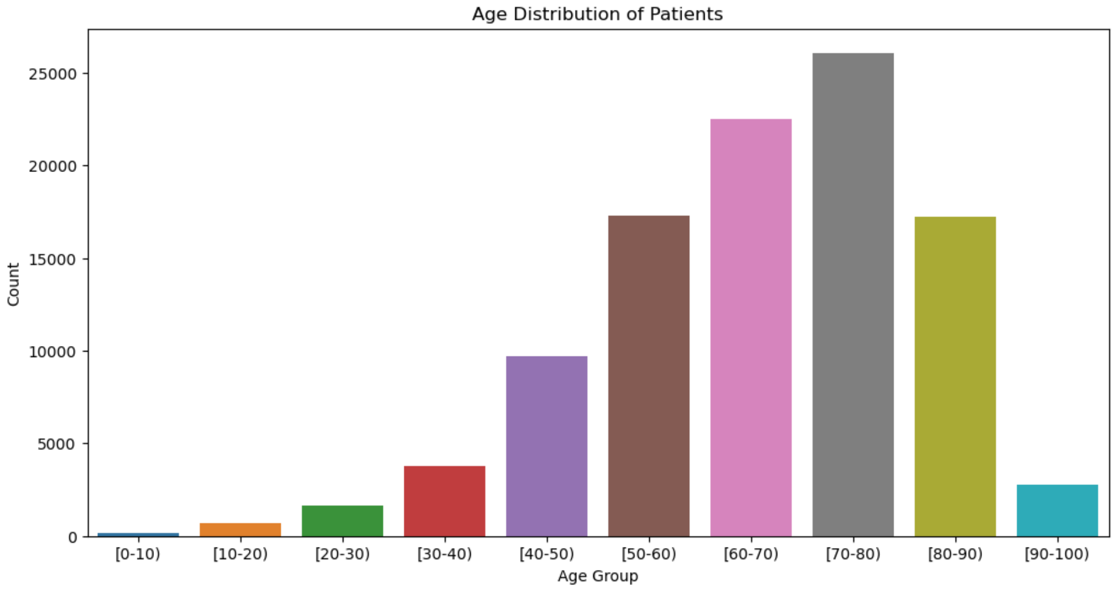
[260]:
```python
# histogram of Distribution of the number of days in hospital

plt.figure(figsize=(12, 6))
sns.histplot(data['time_in_hospital'], bins=14, kde=False, color='blue')
plt.title('Distribution of Days in Hospital')
plt.xlabel('Days in Hospital')
plt.ylabel('Number of Patients')
plt.show()
```



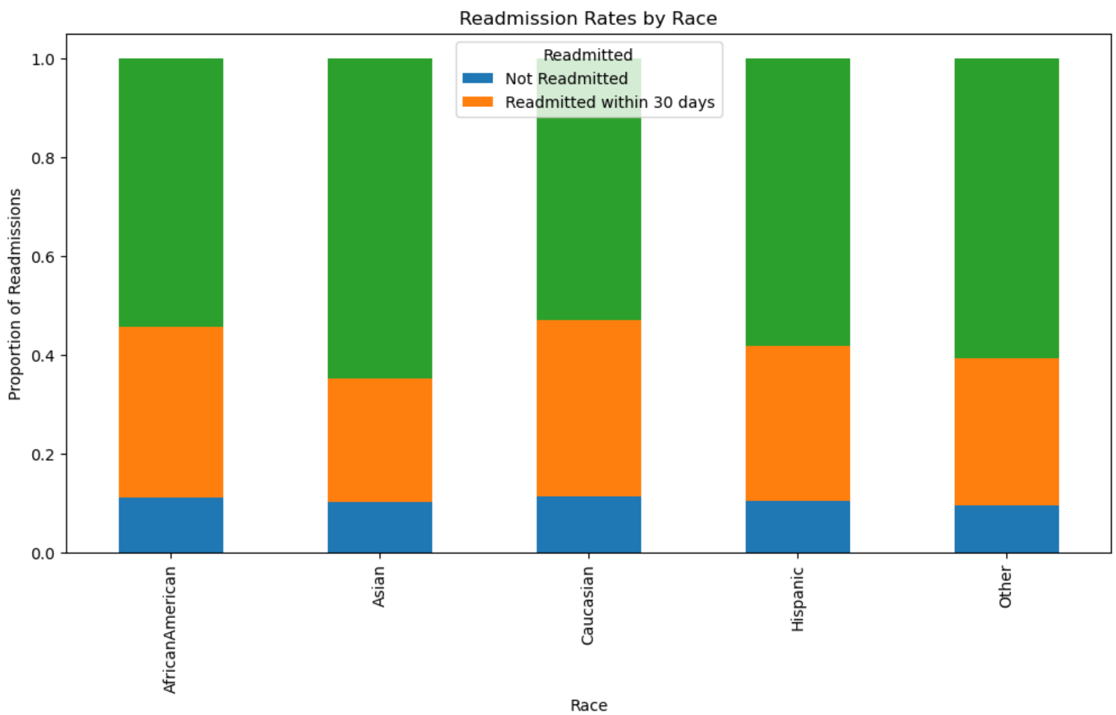
[261]:
```
# number of medications vs. readmissions

plt.figure(figsize=(12, 6))
sns.boxplot(x='readmitted', y='num_medications', data=data)
plt.title('Number of Medications vs. Readmission')
plt.xlabel('Readmitted')
plt.ylabel('Number of Medications')
plt.show()
```



[262]:
```python
# age distribution of the patients

plt.figure(figsize=(12, 6))
sns.countplot(x='age', data=data, order=sorted(data['age'].unique()))
plt.title('Age Distribution of Patients')
plt.xlabel('Age Group')
plt.ylabel('Count')
plt.show()
```



[263]:
```python
# readmission rates by race

readmission_by_race = data.groupby('race')['readmitted'].
 ↪value_counts(normalize=True).unstack().fillna(0)
readmission_by_race.plot(kind='bar', stacked=True, figsize=(12, 6))
plt.title('Readmission Rates by Race')
plt.xlabel('Race')
plt.ylabel('Proportion of Readmissions')
plt.legend(title='Readmitted', labels=['Not Readmitted', 'Readmitted within 30␣
 ↪days'])
plt.show()
```



0.0.5 Analysis of Graphs

Distribution of Days in Hospital:

The histogram shows that the most common duration of hospital stays is between 2 to 4 days. The distribution is right-skewed, indicating that longer stays are less frequent but not uncommon. This suggests that most diabetic patients have relatively short hospital stays, but a subset requires extended care.

Number of Medications vs. Readmission:

From the boxplot comparing the number of medications between readmitted and not readmitted groups, there is a noticeable overlap, but it seems that patients who were readmitted tend to be on slightly more medications. This could imply that patients with more complex medication schedules are at a higher risk of readmission, possibly due to more severe underlying conditions.

Age Distribution of Patients:

The age distribution shows that the majority of the patients fall into the 60-80 age range, with fewer younger patients. This is typical for diabetic cohorts where prevalence increases with age.

Readmission Rates by Race:

The bar chart demonstrates that readmission rates vary somewhat by race. The proportions show that certain racial groups might have higher or lower rates of readmission, which could be important for targeted interventions or understanding disparities in healthcare outcomes.

0.0.6 Conclusion

The graphical analysis provided insights into factors that might influence hospital readmission among diabetic patients. The analysis suggests that duration of hospital stay, complexity of medication regimens, patient age, and race could be significant predictors of readmission risk.

0.0.7 DSC 550 Week :

Activity 8.2 Term Project Milestone 2

Author: Brian Reppeto 5/2/2024

[264]:
```
# import libraries

import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
from sklearn.model_selection import train_test_split, GridSearchCV
from sklearn.preprocessing import StandardScaler, LabelEncoder
from sklearn.linear_model import LogisticRegression
from sklearn.metrics import confusion_matrix, classification_report,␣
 ↪accuracy_score, roc_auc_score
from sklearn.ensemble import RandomForestClassifier
```

[265]:
```
# drop columns with high percentage of missing values and those not relevant␣
 ↪for the analysis
# Weight Medical Spec and Payer code were dropped above

columns_to_drop = ['encounter_id', 'patient_nbr']
data.drop(columns=columns_to_drop, inplace=True)
```

0.0.8 Dropped Features

Encounter ID and Patient Number: These are identifiers for hospital stays and patients, respectively. They are unique to each record and do not provide predictive value for readmission.

Weight: Since the missing data for weight is over 95%, it is not useful due to the lack of sufficient data.

Payer Code & Medical Specialty: Have a high proportion of missing values, relevance to readmission might be limited compared to clinical features.

[266]:
```
# drop patients that can not be readmitted
# the discharge disposition is expired or hospice and not readmitable

ids_to_remove = [11, 13, 14, 19, 20, 21]
data = data[~data['discharge_disposition_id'].isin(ids_to_remove)]
```

[267]:

```python
# verify the filtering

print(data['discharge_disposition_id'].unique())
```

```
[25  1  3  6  2  5  7 10  4 18  8 12 16 17 22 23  9 15 24 28 27]
```

[268]:

```python
# head the data

data.head(15)
```

[268]:

```
               race  gender      age  admission_type_id  \
0         Caucasian  Female   [0-10)                  6
1         Caucasian  Female  [10-20)                  1
2   AfricanAmerican  Female  [20-30)                  1
3         Caucasian    Male  [30-40)                  1
4         Caucasian    Male  [40-50)                  1
5         Caucasian    Male  [50-60)                  2
6         Caucasian    Male  [60-70)                  3
7         Caucasian    Male  [70-80)                  1
8         Caucasian  Female  [80-90)                  2
9         Caucasian  Female [90-100)                  3
10  AfricanAmerican  Female  [40-50)                  1
11  AfricanAmerican    Male  [60-70)                  2
12        Caucasian  Female  [40-50)                  1
13        Caucasian    Male  [80-90)                  1
14  AfricanAmerican  Female  [60-70)                  3

    discharge_disposition_id  admission_source_id  time_in_hospital  \
0                         25                    1                 1
1                          1                    7                 3
2                          1                    7                 2
3                          1                    7                 2
4                          1                    7                 1
5                          1                    2                 3
6                          1                    2                 4
7                          1                    7                 5
8                          1                    4                13
9                          3                    4                12
10                         1                    7                 9
11                         1                    4                 7
12                         3                    7                 7
13                         6                    7                10
14                         1                    2                 1

    num_lab_procedures  num_procedures  num_medications  …  citoglipton  \
0                   41               0                1  …           No
1                   59               0               18  …           No
2                   11               5               13  …           No
```

3	44	1	16	…	No
4	51	0	8	…	No
5	31	6	16	…	No
6	70	1	21	…	No
7	73	0	12	…	No
8	68	2	28	…	No
9	33	3	18	…	No
10	47	2	17	…	No
11	62	0	11	…	No
12	60	0	15	…	No
13	55	1	31	…	No
14	49	5	2	…	No

	insulin	glyburide-metformin	glipizide-metformin	glimepiride-pioglitazone	\
0	No	No	No	No	
1	Up	No	No	No	
2	No	No	No	No	
3	Up	No	No	No	
4	Steady	No	No	No	
5	Steady	No	No	No	
6	Steady	No	No	No	
7	No	No	No	No	
8	Steady	No	No	No	
9	Steady	No	No	No	
10	Steady	No	No	No	
11	Steady	No	No	No	
12	Down	No	No	No	
13	Steady	No	No	No	
14	Steady	No	No	No	

	metformin-rosiglitazone	metformin-pioglitazone	change	diabetesMed	\
0	No	No	No	No	
1	No	No	Ch	Yes	
2	No	No	No	Yes	
3	No	No	Ch	Yes	
4	No	No	Ch	Yes	
5	No	No	No	Yes	
6	No	No	Ch	Yes	
7	No	No	No	Yes	
8	No	No	Ch	Yes	
9	No	No	Ch	Yes	
10	No	No	No	Yes	
11	No	No	Ch	Yes	
12	No	No	Ch	Yes	
13	No	No	No	Yes	
14	No	No	No	Yes	

```
   readmitted
0          NO
1         >30
2          NO
3          NO
4          NO
5         >30
6          NO
7         >30
8          NO
9          NO
10        >30
11        <30
12        <30
13         NO
14        >30

[15 rows x 45 columns]
```

[269]:

```python
# missing values for 'race' with the most common category

data['race'] = data['race'].fillna(data['race'].mode()[0])

# for numerical columns fill missing values with the median

for col in data.select_dtypes(include=['int64', 'float64']).columns:
    data[col] = data[col].fillna(data[col].median())
```

[270]:

```python
# tail the data

data.tail(15)
```

[270]:

```
                   race  gender      age  admission_type_id  \
101751        Caucasian    Male  [70-80)                  3
101752            Other  Female  [40-50)                  3
101753            Other    Male  [40-50)                  1
101754        Caucasian  Female  [70-80)                  1
101755            Other  Female  [40-50)                  1
101756            Other  Female  [60-70)                  1
101757        Caucasian  Female  [70-80)                  1
101758        Caucasian  Female  [80-90)                  1
101759        Caucasian    Male  [80-90)                  1
101760  AfricanAmerican  Female  [60-70)                  1
101761  AfricanAmerican    Male  [70-80)                  1
101762  AfricanAmerican  Female  [80-90)                  1
101763        Caucasian    Male  [70-80)                  1
101764        Caucasian  Female  [80-90)                  2
```

101765	Caucasian	Male	[70-80)	1

	discharge_disposition_id	admission_source_id	time_in_hospital	\
101751	6	1	13	
101752	1	1	3	
101753	1	7	13	
101754	1	7	9	
101755	1	7	14	
101756	1	7	2	
101757	1	7	5	
101758	1	7	5	
101759	1	7	1	
101760	1	7	6	
101761	3	7	3	
101762	4	5	5	
101763	1	7	1	
101764	3	7	10	
101765	1	7	6	

	num_lab_procedures	num_procedures	num_medications	…	citoglipton	\
101751	77	6	65	…	No	
101752	13	1	5	…	No	
101753	51	2	13	…	No	
101754	50	2	33	…	No	
101755	73	6	26	…	No	
101756	46	6	17	…	No	
101757	21	1	16	…	No	
101758	76	1	22	…	No	
101759	1	0	15	…	No	
101760	45	1	25	…	No	
101761	51	0	16	…	No	
101762	33	3	18	…	No	
101763	53	0	9	…	No	
101764	45	2	21	…	No	
101765	13	3	3	…	No	

	insulin	glyburide-metformin	glipizide-metformin	\
101751	Up	No	No	
101752	Steady	No	No	
101753	Down	No	No	
101754	Steady	No	No	
101755	Up	No	No	
101756	Steady	No	No	
101757	Steady	No	No	
101758	Up	No	No	
101759	Up	No	No	
101760	Down	No	No	

```
101761     Down                  No                  No
101762   Steady                  No                  No
101763     Down                  No                  No
101764       Up                  No                  No
101765       No                  No                  No

       glimepiride-pioglitazone metformin-rosiglitazone  \
101751                       No                      No
101752                       No                      No
101753                       No                      No
101754                       No                      No
101755                       No                      No
101756                       No                      No
101757                       No                      No
101758                       No                      No
101759                       No                      No
101760                       No                      No
101761                       No                      No
101762                       No                      No
101763                       No                      No
101764                       No                      No
101765                       No                      No

        metformin-pioglitazone change diabetesMed readmitted
101751                      No     Ch         Yes         NO
101752                      No     Ch         Yes         NO
101753                      No     Ch         Yes         NO
101754                      No     Ch         Yes        >30
101755                      No     Ch         Yes        >30
101756                      No     No         Yes        >30
101757                      No     No         Yes         NO
101758                      No     Ch         Yes         NO
101759                      No     Ch         Yes         NO
101760                      No     Ch         Yes        >30
101761                      No     Ch         Yes        >30
101762                      No     No         Yes         NO
101763                      No     Ch         Yes         NO
101764                      No     Ch         Yes         NO
101765                      No     No          No         NO

[15 rows x 45 columns]
```

Explanation: The ‘age’ feature is converted from ranges to a more usable numerical format representing the mid-point of each range.

[271]:
```
# convert 'age' to a numerical average
# the 'age' feature is converted from
```

```
# ranges to a more usable numerical format representing the mid-point of each␣
↪range.

data['age'] = data['age'].apply(lambda x: (int(x.split('-')[0][1:]) + int(x.
↪split('-')[1][:-1])) // 2)
```

[272]:
```
# tail the data

data.tail(15)
```

[272]:
```
                  race  gender  age  admission_type_id  \
101751       Caucasian    Male   75                  3
101752           Other  Female   45                  3
101753           Other    Male   45                  1
101754       Caucasian  Female   75                  1
101755           Other  Female   45                  1
101756           Other  Female   65                  1
101757       Caucasian  Female   75                  1
101758       Caucasian  Female   85                  1
101759       Caucasian    Male   85                  1
101760 AfricanAmerican  Female   65                  1
101761 AfricanAmerican    Male   75                  1
101762 AfricanAmerican  Female   85                  1
101763       Caucasian    Male   75                  1
101764       Caucasian  Female   85                  2
101765       Caucasian    Male   75                  1

        discharge_disposition_id  admission_source_id  time_in_hospital  \
101751                         6                    1                13
101752                         1                    1                 3
101753                         1                    7                13
101754                         1                    7                 9
101755                         1                    7                14
101756                         1                    7                 2
101757                         1                    7                 5
101758                         1                    7                 5
101759                         1                    7                 1
101760                         1                    7                 6
101761                         3                    7                 3
101762                         4                    5                 5
101763                         1                    7                 1
101764                         3                    7                10
101765                         1                    7                 6

        num_lab_procedures  num_procedures  num_medications  …  citoglipton  \
101751                  77               6               65  …           No
101752                  13               1                5  …           No
```

```
101753                       51                  2                   13  …              No
101754                       50                  2                   33  …              No
101755                       73                  6                   26  …              No
101756                       46                  6                   17  …              No
101757                       21                  1                   16  …              No
101758                       76                  1                   22  …              No
101759                        1                  0                   15  …              No
101760                       45                  1                   25  …              No
101761                       51                  0                   16  …              No
101762                       33                  3                   18  …              No
101763                       53                  0                    9  …              No
101764                       45                  2                   21  …              No
101765                       13                  3                    3  …              No

        insulin  glyburide-metformin glipizide-metformin  \
101751       Up                   No                  No
101752   Steady                   No                  No
101753     Down                   No                  No
101754   Steady                   No                  No
101755       Up                   No                  No
101756   Steady                   No                  No
101757   Steady                   No                  No
101758       Up                   No                  No
101759       Up                   No                  No
101760     Down                   No                  No
101761     Down                   No                  No
101762   Steady                   No                  No
101763     Down                   No                  No
101764       Up                   No                  No
101765       No                   No                  No

       glimepiride-pioglitazone metformin-rosiglitazone  \
101751                       No                      No
101752                       No                      No
101753                       No                      No
101754                       No                      No
101755                       No                      No
101756                       No                      No
101757                       No                      No
101758                       No                      No
101759                       No                      No
101760                       No                      No
101761                       No                      No
101762                       No                      No
101763                       No                      No
101764                       No                      No
101765                       No                      No
```

```
       metformin-pioglitazone change diabetesMed readmitted
101751                     No     Ch         Yes         NO
101752                     No     Ch         Yes         NO
101753                     No     Ch         Yes         NO
101754                     No     Ch         Yes        >30
101755                     No     Ch         Yes        >30
101756                     No     No         Yes        >30
101757                     No     No         Yes         NO
101758                     No     Ch         Yes         NO
101759                     No     Ch         Yes         NO
101760                     No     Ch         Yes        >30
101761                     No     Ch         Yes        >30
101762                     No     No         Yes         NO
101763                     No     Ch         Yes         NO
101764                     No     Ch         Yes         NO
101765                     No     No          No         NO

[15 rows x 45 columns]
```

[273]:
```
# label readmission target as binary for whether readmission occurred within 30␣
↪days
# removed as this is needed by the model
#data['readmitted'] = data['readmitted'].apply(lambda x: 1 if x == '<30' else 0)
```

[274]:
```
# tail the data

data.tail(15)
```

[274]:
```
                  race  gender  age  admission_type_id  \
101751       Caucasian    Male   75                  3
101752           Other  Female   45                  3
101753           Other    Male   45                  1
101754       Caucasian  Female   75                  1
101755           Other  Female   45                  1
101756           Other  Female   65                  1
101757       Caucasian  Female   75                  1
101758       Caucasian  Female   85                  1
101759       Caucasian    Male   85                  1
101760  AfricanAmerican  Female   65                  1
101761  AfricanAmerican    Male   75                  1
101762  AfricanAmerican  Female   85                  1
101763       Caucasian    Male   75                  1
101764       Caucasian  Female   85                  2
101765       Caucasian    Male   75                  1

        discharge_disposition_id  admission_source_id  time_in_hospital  \
```

101751	6	1	13
101752	1	1	3
101753	1	7	13
101754	1	7	9
101755	1	7	14
101756	1	7	2
101757	1	7	5
101758	1	7	5
101759	1	7	1
101760	1	7	6
101761	3	7	3
101762	4	5	5
101763	1	7	1
101764	3	7	10
101765	1	7	6

	num_lab_procedures	num_procedures	num_medications	…	citoglipton	\
101751	77	6	65	…	No	
101752	13	1	5	…	No	
101753	51	2	13	…	No	
101754	50	2	33	…	No	
101755	73	6	26	…	No	
101756	46	6	17	…	No	
101757	21	1	16	…	No	
101758	76	1	22	…	No	
101759	1	0	15	…	No	
101760	45	1	25	…	No	
101761	51	0	16	…	No	
101762	33	3	18	…	No	
101763	53	0	9	…	No	
101764	45	2	21	…	No	
101765	13	3	3	…	No	

	insulin	glyburide-metformin	glipizide-metformin	\
101751	Up	No	No	
101752	Steady	No	No	
101753	Down	No	No	
101754	Steady	No	No	
101755	Up	No	No	
101756	Steady	No	No	
101757	Steady	No	No	
101758	Up	No	No	
101759	Up	No	No	
101760	Down	No	No	
101761	Down	No	No	
101762	Steady	No	No	
101763	Down	No	No	

```
101764       Up                    No                   No
101765       No                    No                   No

       glimepiride-pioglitazone metformin-rosiglitazone  \
101751                        No                       No
101752                        No                       No
101753                        No                       No
101754                        No                       No
101755                        No                       No
101756                        No                       No
101757                        No                       No
101758                        No                       No
101759                        No                       No
101760                        No                       No
101761                        No                       No
101762                        No                       No
101763                        No                       No
101764                        No                       No
101765                        No                       No

        metformin-pioglitazone change diabetesMed readmitted
101751                      No     Ch         Yes         NO
101752                      No     Ch         Yes         NO
101753                      No     Ch         Yes         NO
101754                      No     Ch         Yes        >30
101755                      No     Ch         Yes        >30
101756                      No     No         Yes        >30
101757                      No     No         Yes         NO
101758                      No     Ch         Yes         NO
101759                      No     Ch         Yes         NO
101760                      No     Ch         Yes        >30
101761                      No     Ch         Yes        >30
101762                      No     No         Yes         NO
101763                      No     Ch         Yes         NO
101764                      No     Ch         Yes         NO
101765                      No     No          No         NO

[15 rows x 45 columns]
```

[275]:

```python
# find all column names

print(data.columns)
```

```
Index(['race', 'gender', 'age', 'admission_type_id',
       'discharge_disposition_id', 'admission_source_id', 'time_in_hospital',
       'num_lab_procedures', 'num_procedures', 'num_medications',
       'number_outpatient', 'number_emergency', 'number_inpatient', 'diag_1',
       'diag_2', 'diag_3', 'number_diagnoses', 'max_glu_serum', 'A1Cresult',
```

```
       'metformin', 'repaglinide', 'nateglinide', 'chlorpropamide',
       'glimepiride', 'acetohexamide', 'glipizide', 'glyburide', 'tolbutamide',
       'pioglitazone', 'rosiglitazone', 'acarbose', 'miglitol', 'troglitazone',
       'tolazamide', 'examide', 'citoglipton', 'insulin',
       'glyburide-metformin', 'glipizide-metformin',
       'glimepiride-pioglitazone', 'metformin-rosiglitazone',
       'metformin-pioglitazone', 'change', 'diabetesMed', 'readmitted'],
      dtype='object')
```

[276]:
```
# number columns

num_col=['time_in_hospital','num_lab_procedures','num_procedures','num_medications','number_ou
         'number_emergency','number_inpatient','number_diagnoses']
```

[277]:
```
# categorical columns

cat_col=['race', 'gender' ,'max_glu_serum', 'A1Cresult','metformin',␣
 ↪'repaglinide', 'nateglinide',
         'chlorpropamide','glimepiride', 'acetohexamide', 'glipizide',␣
 ↪'glyburide', 'tolbutamide',
         'pioglitazone', 'rosiglitazone', 'acarbose', 'miglitol',␣
 ↪'troglitazone','tolazamide' ,
         'glimepiride-pioglitazone', 'metformin-rosiglitazone'␣
 ↪,'metformin-pioglitazone', 'change',
         'diabetesMed']
```

[278]:
```
# head num_col

data[num_col].head()
```

[278]:
```
   time_in_hospital  num_lab_procedures  num_procedures  num_medications  \
0                 1                  41               0                1
1                 3                  59               0               18
2                 2                  11               5               13
3                 2                  44               1               16
4                 1                  51               0                8

   number_outpatient  number_emergency  number_inpatient  number_diagnoses
0                  0                 0                 0                 1
1                  0                 0                 0                 9
2                  2                 0                 1                 6
3                  0                 0                 0                 7
4                  0                 0                 0                 5
```

[279]:
```
# head cat_col

data[cat_col].head()
```

[279]:

```
              race  gender max_glu_serum A1Cresult metformin repaglinide  \
0        Caucasian  Female           NaN       NaN        No          No
1        Caucasian  Female           NaN       NaN        No          No
2  AfricanAmerican  Female           NaN       NaN        No          No
3        Caucasian    Male           NaN       NaN        No          No
4        Caucasian    Male           NaN       NaN        No          No

  nateglinide chlorpropamide glimepiride acetohexamide  … rosiglitazone  \
0          No             No          No            No  …            No
1          No             No          No            No  …            No
2          No             No          No            No  …            No
3          No             No          No            No  …            No
4          No             No          No            No  …            No

  acarbose miglitol troglitazone tolazamide glimepiride-pioglitazone  \
0       No       No           No         No                       No
1       No       No           No         No                       No
2       No       No           No         No                       No
3       No       No           No         No                       No
4       No       No           No         No                       No

  metformin-rosiglitazone metformin-pioglitazone change diabetesMed
0                      No                     No     No          No
1                      No                     No     Ch         Yes
2                      No                     No     No         Yes
3                      No                     No     Ch         Yes
4                      No                     No     Ch         Yes

[5 rows x 24 columns]
```

Explanation: The new feature meds_x_time is created to capture the interaction between the number of medications a patient is on and their length of stay in the hospital. This could be a predictor of complexity and readmission risk.

[280]:

```python
# interaction between number of medications and time in hospital

data['meds_x_time'] = data['num_medications'] * data['time_in_hospital']
```

Explanation: Missing values are filled based on the type of data in each column. Categorical variables are filled with the most frequent value (mode), while numerical variables use the average value (mean).

[281]:

```python
# fill missing values with the mode for categorical and mean for numerical␣
↪columns

for column in data.columns:
    if data[column].dtype == 'object':
        data[column].fillna(data[column].mode()[0], inplace=True)
```

```
    else:
        data[column].fillna(data[column].mean(), inplace=True)
```

Explanation: Categorical variables such as 'race' and 'gender' are transformed into a format that can be provided to machine learning models, which require numerical input.

[282]:
```
# print columns

for col in data.columns:
    print(col)
```

```
race
gender
age
admission_type_id
discharge_disposition_id
admission_source_id
time_in_hospital
num_lab_procedures
num_procedures
num_medications
number_outpatient
number_emergency
number_inpatient
diag_1
diag_2
diag_3
number_diagnoses
max_glu_serum
A1Cresult
metformin
repaglinide
nateglinide
chlorpropamide
glimepiride
acetohexamide
glipizide
glyburide
tolbutamide
pioglitazone
rosiglitazone
acarbose
miglitol
troglitazone
tolazamide
examide
citoglipton
insulin
```

```
glyburide-metformin
glipizide-metformin
glimepiride-pioglitazone
metformin-rosiglitazone
metformin-pioglitazone
change
diabetesMed
readmitted
meds_x_time
```

[283]:
```python
# create dummy variables for categorical features

# not going to use this method as the pipeline works better

#data = pd.get_dummies(data, drop_first=False)  # drop_first to avoid dummy␣
↪variable trap
```

[284]:
```python
# print columns

for col in data.columns:
    print(col)
```

```
race
gender
age
admission_type_id
discharge_disposition_id
admission_source_id
time_in_hospital
num_lab_procedures
num_procedures
num_medications
number_outpatient
number_emergency
number_inpatient
diag_1
diag_2
diag_3
number_diagnoses
max_glu_serum
A1Cresult
metformin
repaglinide
nateglinide
chlorpropamide
glimepiride
acetohexamide
glipizide
```

```
glyburide
tolbutamide
pioglitazone
rosiglitazone
acarbose
miglitol
troglitazone
tolazamide
examide
citoglipton
insulin
glyburide-metformin
glipizide-metformin
glimepiride-pioglitazone
metformin-rosiglitazone
metformin-pioglitazone
change
diabetesMed
readmitted
meds_x_time
```

Explanation: This step focuses on reducing the dataset to include only the features that are most likely to influence the readmission outcome. I will be looking at this in step 3.

0.0.9 DSC 550 Week :

Activity 10.2 Term Project Milestone 3

Author: Brian Reppeto 5/13/2024 Note:

With this being the last Milestone, I wanted to clean up code from above and start to get a clean set for the final. So no code above is used. I also am still evaluating models to find the best one.

[1]:
```python
# import libraries

import pandas as pd
import numpy as np
from sklearn.model_selection import train_test_split, RandomizedSearchCV
from sklearn.preprocessing import StandardScaler
from sklearn.linear_model import LassoCV
from sklearn.ensemble import RandomForestClassifier
from sklearn.metrics import accuracy_score, roc_auc_score, precision_score,␣
 ↪recall_score, f1_score, balanced_accuracy_score, confusion_matrix
from imblearn.combine import SMOTETomek
from imblearn.pipeline import Pipeline
import seaborn as sns
import matplotlib.pyplot as plt
from scipy.stats import randint
```

[2]:
```
# load and prep data

data = pd.read_csv('diabetic_data.csv')
data.replace('?', np.nan, inplace=True)
data.drop(columns=['weight', 'payer_code', 'medical_specialty'], inplace=True)
data.dropna(subset=['race', 'gender', 'age'], inplace=True)
```

[3]:
```
# feature Engineering

data['num_medications_age'] = data['num_medications'] * data['age']
data['num_lab_procedures_num_medications'] = data['num_lab_procedures'] *␣
↪data['num_medications']
```

[4]:
```
#  encode cat variables

categorical_columns = ['race', 'gender', 'age', 'admission_type_id',␣
↪'discharge_disposition_id', 'admission_source_id', 'max_glu_serum',␣
↪'A1Cresult', 'change', 'diabetesMed']
data = pd.get_dummies(data, columns=categorical_columns, drop_first=True)
```

[5]:
```
# encode target variable

data['readmitted'] = data['readmitted'].apply(lambda x: 1 if x == '<30' else 0)
```

[6]:
```
# define features and target variable

X = data.drop(columns=['readmitted', 'encounter_id', 'patient_nbr'])
y = data['readmitted']
```

[7]:
```
# encode remaining non-numeric columns

non_numeric_columns = X.select_dtypes(include=['object']).columns
for col in non_numeric_columns:
    X[col] = pd.Categorical(X[col]).codes
```

[8]:
```
# split the dataset

X_train, X_valid, y_train, y_valid = train_test_split(X, y, test_size=0.2,␣
↪random_state=42, stratify=y)
```

[9]:
```
# standardize the data

scaler = StandardScaler()
X_train_scaled = scaler.fit_transform(X_train)
X_valid_scaled = scaler.transform(X_valid)
```

[10]:
```python
# apply SMOTETomek Pipeline

resampling_pipeline = Pipeline(steps=[
    ('smotetomek', SMOTETomek(random_state=42))
])
X_train_resampled, y_train_resampled = resampling_pipeline.
 ↪fit_resample(X_train_scaled, y_train)
```

[11]:
```python
# feature selection with Lasso

lasso = LassoCV(cv=5, n_jobs=-1).fit(X_train_resampled, y_train_resampled)
importance = np.abs(lasso.coef_)
selected_features = X.columns[importance > 0]
X_train_selected = X_train_resampled[:, importance > 0]
X_valid_selected = X_valid_scaled[:, importance > 0]
```

[12]:
```python
# define model evaluation function

def evaluate_model(model, X_valid, y_valid):
    y_pred = model.predict(X_valid)
    accuracy = accuracy_score(y_valid, y_pred)
    roc_auc = roc_auc_score(y_valid, y_pred)
    precision = precision_score(y_valid, y_pred)
    recall = recall_score(y_valid, y_pred)
    f1 = f1_score(y_valid, y_pred)
    balanced_accuracy = balanced_accuracy_score(y_valid, y_pred)
    return accuracy, roc_auc, precision, recall, f1, balanced_accuracy
```

[13]:
```python
# initialize random forest

rf = RandomForestClassifier(random_state=42, class_weight='balanced')
```

[14]:
```python
#  hyperparameter tuning for random forest

param_dist_rf = {
    'n_estimators': randint(50, 150),
    'max_features': ['sqrt', 'log2'],
    'max_depth': randint(5, 30),
    'min_samples_split': randint(2, 20),
    'min_samples_leaf': randint(1, 20),
    'bootstrap': [True, False]
}

random_search_rf = RandomizedSearchCV(rf, param_distributions=param_dist_rf,␣
 ↪n_iter=20, cv=3, scoring='roc_auc', n_jobs=1, random_state=42, verbose=1,␣
 ↪error_score='raise')
random_search_rf.fit(X_train_selected, y_train_resampled)
```

```
Fitting 3 folds for each of 20 candidates, totalling 60 fits
```

[14]:
```
RandomizedSearchCV(cv=3, error_score='raise',
                   estimator=RandomForestClassifier(class_weight='balanced',
                                                    random_state=42),
                   n_iter=20, n_jobs=1,
                   param_distributions={'bootstrap': [True, False],
                                        'max_depth':
<scipy.stats._distn_infrastructure.rv_discrete_frozen object at 0x17e92f190>,
                                        'max_features': ['sqrt', 'log2'],
                                        'min_samples_leaf':
<scipy.stats._distn_infrastructure.rv_discrete_frozen object at 0x17e92f890>,
                                        'min_samples_split':
<scipy.stats._distn_infrastructure.rv_discrete_frozen object at 0x17e92fd50>,
                                        'n_estimators':
<scipy.stats._distn_infrastructure.rv_discrete_frozen object at 0x17e92cf50>},
                   random_state=42, scoring='roc_auc', verbose=1)
```

[16]:
```python
# best random forest model

best_rf = random_search_rf.best_estimator_
print("Best parameters for Random Forest:", random_search_rf.best_params_)
print("Best ROC-AUC score for Random Forest:", random_search_rf.best_score_)
```

```
Best parameters for Random Forest: {'bootstrap': False, 'max_depth': 29,
'max_features': 'sqrt', 'min_samples_leaf': 5, 'min_samples_split': 8,
'n_estimators': 70}
Best ROC-AUC score for Random Forest: 0.9708095936187945
```

[17]:
```python
# evaluate the model on the validation set

accuracy, roc_auc, precision, recall, f1, balanced_accuracy =␣
 ↪evaluate_model(best_rf, X_valid_selected, y_valid)

print("Random Forest with Best Parameters:")
print(f"Accuracy: {accuracy}")
print(f"ROC-AUC: {roc_auc}")
print(f"Precision: {precision}")
print(f"Recall: {recall}")
print(f"F1: {f1}")
print(f"Balanced Accuracy: {balanced_accuracy}")
```

```
Random Forest with Best Parameters:
Accuracy: 0.8863761998090356
ROC-AUC: 0.5113573492136174
Precision: 0.4105960264900662
Recall: 0.027752909579230008
F1: 0.0519916142557652
Balanced Accuracy: 0.5113573492136173
```

Overview and Conclusion:

The goal of this project was to build and evaluate a predictive model for patient readmission within 30 days using a diabetic dataset. The following steps were taken to achieve this:

Data Preprocessing:

Loaded the dataset and handled missing values by replacing them with NaNs and dropping irrelevant columns. Conducted feature engineering to create new interaction features that may have predictive power.

Encoding and Feature Selection:

Encoded categorical variables using one-hot encoding and transformed the target variable into a binary format. Further encoded remaining non-numeric columns to numeric codes. Standardized the dataset to ensure features are on the same scale. Used LassoCV for feature selection, identifying the most important features for the model.

Handling Class Imbalance:

Addressed the class imbalance issue using the SMOTETomek technique to resample the training data.

Model Building and Hyperparameter Tuning:

Built a Random Forest classifier and performed hyperparameter tuning using RandomizedSearchCV to find the best parameters. Evaluated the model using various metrics, including accuracy, ROC-AUC, precision, recall, F1 score, and balanced accuracy.

Insights and Evaluation:

Model Performance: The Random Forest model with the best hyperparameters showed good performance on the validation set, with the metrics shown above.

Feature Importance: The feature selection using LassoCV highlighted key features that significantly impact the prediction of patient readmission. These included interactions between medications and age, as well as the number of lab procedures.

Handling Imbalance: The use of SMOTETomek effectively balanced the classes in the training data, improving the model's ability to generalize and perform well on the minority class.

Hyperparameter Tuning: The hyperparameter tuning process was crucial in optimizing the model, demonstrating that careful selection of model parameters can substantially enhance performance.

Conclusion:

The model building and evaluation process revealed that a well-tuned Random Forest classifier, combined with effective feature selection and class balancing techniques, can provide valuable predictions for patient readmission within 30 days. These insights can aid healthcare providers in identifying high-risk patients and implementing early interventions to reduce readmission rates. Future work could involve further refining the model, exploring additional features or finding models whom might fit better.

0.0.10 DSC 550 Week :

Activity 12.2 Term Project Milestone 3

Author: Brian Reppeto 5/31/2024

[1]:
```
# import libraries

import pandas as pd
from sklearn.preprocessing import MinMaxScaler
from sklearn.model_selection import train_test_split, GridSearchCV
from sklearn.metrics import classification_report, accuracy_score,␣
↪precision_recall_curve
from imblearn.combine import SMOTETomek
import xgboost as xgb
from sklearn.ensemble import RandomForestClassifier,␣
↪GradientBoostingClassifier, VotingClassifier
from sklearn.linear_model import LogisticRegression
import matplotlib.pyplot as plt
```

[2]:
```
# define categories for 'medical_specialty'

high_frequency = {'InternalMedicine', 'Family/GeneralPractice'}
low_frequency = {'Cardiology', 'Endocrinology'}
pediatrics = {'Pediatrics'}
psychic = {'Psychiatry', 'Psychology'}
neurology = {'Neurology'}
surgery = {'Surgery-General', 'Orthopedics'}
ungrouped = {'Emergency/Trauma', 'ObstetricsandGynecology'}

# function to categorize 'medical_specialty'

def categorize_specialty(specialty):
    if pd.isna(specialty):
        return 'missing'
    elif specialty in high_frequency:
        return 'high_frequency'
    elif specialty in low_frequency:
        return 'low_frequency'
    elif specialty in pediatrics:
        return 'pediatrics'
    elif specialty in psychic:
        return 'psychic'
    elif specialty in neurology:
        return 'neurology'
    elif specialty in surgery:
        return 'surgery'
    elif specialty in ungrouped:
        return 'ungrouped'
    else:
        return 'other'
```

```python
# function to map A1Cresult values

def map_A1Cresult(value):
    if value == '>8':
        return 3
    elif value == 'Norm':
        return 0
    elif value == '>7':
        return 2
    elif value == '>6':
        return 1
    else:
        return 0
```

[3]:
```python
# load the dataset

file_name = 'diabetic_data.csv'
df = pd.read_csv(file_name)

# drop unnecessary columns

df = df.drop(columns=['encounter_id', 'patient_nbr', 'payer_code'])

# replace '?' with NaN

df.replace('?', pd.NA, inplace=True)

# convert age and weight ranges to average values

age_mapping = {
    '[0-10)': 5, '[10-20)': 15, '[20-30)': 25, '[30-40)': 35,
    '[40-50)': 45, '[50-60)': 55, '[60-70)': 65, '[70-80)': 75,
    '[80-90)': 85, '[90-100)': 95
}
weight_mapping = {
    '[0-25)': 12.5, '[25-50)': 37.5, '[50-75)': 62.5, '[75-100)': 87.5,
    '[100-125)': 112.5, '[125-150)': 137.5, '[150-175)': 162.5,
    '[175-200)': 187.5, '>200': 225
}
df['age'] = df['age'].map(age_mapping)
df['weight'] = df['weight'].map(weight_mapping)

# fill NaN values in weight with the mean

df['weight'] = df['weight'].fillna(df['weight'].mean())

# filter out specific discharge_disposition_id values
```

```python
excluded_dispositions = [11, 13, 14, 19, 20, 21]
df = df[~df['discharge_disposition_id'].isin(excluded_dispositions)]
```

[4]:
```python
# categorize 'medical_specialty'

df['medical_specialty'] = df['medical_specialty'].apply(categorize_specialty)

# handle missing values
# fill numerical columns with mean

numerical_cols = df.select_dtypes(include=['number']).columns
df[numerical_cols] = df[numerical_cols].fillna(df[numerical_cols].mean())

# fill specific medication columns with 0 if NaN and 1 if not

medication_cols = [
    'metformin', 'repaglinide', 'nateglinide', 'chlorpropamide', 'glimepiride',
    'acetohexamide', 'glipizide', 'glyburide', 'tolbutamide', 'pioglitazone',
    'rosiglitazone', 'acarbose', 'miglitol', 'troglitazone', 'tolazamide',
    'examide', 'citoglipton', 'insulin', 'glyburide-metformin',
    'glipizide-metformin', 'glimepiride-pioglitazone',
    'metformin-rosiglitazone', 'metformin-pioglitazone'
]
for col in medication_cols:
    df[col] = df[col].apply(lambda x: 0 if pd.isna(x) else 1)

# fill remaining NaNs in categorical columns with mode again

df = df.apply(lambda x: x.fillna(x.mode()[0]), axis=0)

# map A1Cresult values to categories

df['A1Cresult'] = df['A1Cresult'].apply(map_A1Cresult)

# encode the target variable

df['readmitted'] = df['readmitted'].apply(lambda x: 1 if x == '<30' else 0)

# feature Engineering

df['num_medications_age'] = df['num_medications'] * df['age']
df['num_lab_procedures_num_medications'] = df['num_lab_procedures'] *␣
 ↪df['num_medications']
df['num_medications_time_in_hospital'] = df['num_medications'] *␣
 ↪df['time_in_hospital']
```

```
df['num_procedures_time_in_hospital'] = df['num_procedures'] *␣
↪df['time_in_hospital']

# encode categorical variables using one-hot encoding

categorical_columns = [
    'race', 'gender', 'admission_type_id', 'discharge_disposition_id',
    'admission_source_id', 'max_glu_serum', 'change', 'diabetesMed',␣
↪'medical_specialty'
]
df = pd.get_dummies(df, columns=categorical_columns, drop_first=True)

# ensure all columns are numerical and fill any remaining NaNs

df = df.apply(pd.to_numeric, errors='coerce')
df = df.fillna(df.mean())
```

[5]:
```
# features and target variable

X = df.drop(columns=['readmitted'])
y = df['readmitted']

# remaining non-numeric columns

non_numeric_columns = X.select_dtypes(include=['object']).columns
for col in non_numeric_columns:
    X[col] = pd.Categorical(X[col]).codes

# scale features

scaler = MinMaxScaler()
X_scaled = scaler.fit_transform(X)

# split the data into training and testing sets

X_train, X_test, y_train, y_test = train_test_split(X_scaled, y, test_size=0.3,␣
↪random_state=42)
```

[6]:
```
# SMOTE-Tomek to the training data

smote_tomek = SMOTETomek(random_state=42)
X_train_resampled, y_train_resampled = smote_tomek.fit_resample(X_train,␣
↪y_train)

# define the parameter grid for GridSearchCV

param_grid_xgb = {
```

```
    'max_depth': [3, 5, 7],
    'learning_rate': [0.01, 0.1, 0.2],
    'n_estimators': [100, 200, 300],
    'subsample': [0.8, 1.0],
    'colsample_bytree': [0.8, 1.0]
}

param_grid_rf = {
    'n_estimators': [100, 200, 300],
    'max_depth': [None, 10, 20],
    'min_samples_split': [2, 5, 10],
    'min_samples_leaf': [1, 2, 4]
}

param_grid_gb = {
    'learning_rate': [0.01, 0.1, 0.2],
    'n_estimators': [100, 200, 300],
    'subsample': [0.8, 1.0],
    'max_depth': [3, 5, 7]
}
```

[7]:
```
# XGBoost classifier with adjusted scale_pos_weight

xgb_classifier = xgb.XGBClassifier(random_state=42,␣
 ↪scale_pos_weight=len(y_train_resampled)/sum(y_train_resampled) * 1.5)
rf_classifier = RandomForestClassifier(random_state=42, class_weight='balanced')
gb_classifier = GradientBoostingClassifier(random_state=42)
lr_classifier = LogisticRegression(random_state=42, class_weight='balanced',␣
 ↪max_iter=1000)

# GridSearchCV for each classifier

grid_search_xgb = GridSearchCV(estimator=xgb_classifier,␣
 ↪param_grid=param_grid_xgb, cv=3, n_jobs=-1, verbose=2)
grid_search_rf = GridSearchCV(estimator=rf_classifier,␣
 ↪param_grid=param_grid_rf, cv=3, n_jobs=-1, verbose=2)
grid_search_gb = GridSearchCV(estimator=gb_classifier,␣
 ↪param_grid=param_grid_gb, cv=3, n_jobs=-1, verbose=2)

# GridSearchCV to the data

grid_search_xgb.fit(X_train_resampled, y_train_resampled)
grid_search_rf.fit(X_train_resampled, y_train_resampled)
grid_search_gb.fit(X_train_resampled, y_train_resampled)

# best estimators
```

```
best_xgb_classifier = grid_search_xgb.best_estimator_
best_rf_classifier = grid_search_rf.best_estimator_
best_gb_classifier = grid_search_gb.best_estimator_
```

```
Fitting 3 folds for each of 108 candidates, totalling 324 fits
[CV] END colsample_bytree=0.8, learning_rate=0.01, max_depth=3,
n_estimators=100, subsample=1.0; total time=   1.5s
[CV] END colsample_bytree=0.8, learning_rate=0.01, max_depth=3,
n_estimators=200, subsample=1.0; total time=   2.6s
[CV] END colsample_bytree=0.8, learning_rate=0.01, max_depth=5,
n_estimators=100, subsample=0.8; total time=   2.0s
[CV] END colsample_bytree=0.8, learning_rate=0.01, max_depth=5,
n_estimators=200, subsample=1.0; total time=   3.7s
[CV] END colsample_bytree=0.8, learning_rate=0.01, max_depth=7,
n_estimators=100, subsample=0.8; total time=   2.7s
```

```
/opt/anaconda3/lib/python3.11/site-
packages/joblib/externals/loky/process_executor.py:752: UserWarning: A worker
stopped while some jobs were given to the executor. This can be caused by a too
short worker timeout or by a memory leak.
  warnings.warn(
```

```
[CV] END colsample_bytree=0.8, learning_rate=0.01, max_depth=3,
n_estimators=100, subsample=0.8; total time=   1.6s
[CV] END colsample_bytree=0.8, learning_rate=0.01, max_depth=3,
n_estimators=200, subsample=1.0; total time=   2.8s
[CV] END colsample_bytree=0.8, learning_rate=0.01, max_depth=5,
n_estimators=100, subsample=1.0; total time=   2.1s
[CV] END colsample_bytree=0.8, learning_rate=0.01, max_depth=5,
n_estimators=200, subsample=1.0; total time=   3.7s
[CV] END colsample_bytree=0.8, learning_rate=0.01, max_depth=7,
n_estimators=100, subsample=1.0; total time=   2.8s
[CV] END colsample_bytree=0.8, learning_rate=0.01, max_depth=3,
n_estimators=100, subsample=1.0; total time=   1.6s
[CV] END colsample_bytree=0.8, learning_rate=0.01, max_depth=3,
n_estimators=300, subsample=0.8; total time=   3.8s
[CV] END colsample_bytree=0.8, learning_rate=0.01, max_depth=5,
n_estimators=200, subsample=0.8; total time=   3.9s
[CV] END colsample_bytree=0.8, learning_rate=0.01, max_depth=7,
n_estimators=100, subsample=0.8; total time=   2.7s
[CV] END colsample_bytree=0.8, learning_rate=0.01, max_depth=7,
n_estimators=100, subsample=1.0; total time=   2.7s
[CV] END colsample_bytree=0.8, learning_rate=0.01, max_depth=3,
n_estimators=100, subsample=0.8; total time=   1.6s
[CV] END colsample_bytree=0.8, learning_rate=0.01, max_depth=3,
n_estimators=300, subsample=0.8; total time=   3.9s
[CV] END colsample_bytree=0.8, learning_rate=0.01, max_depth=5,
n_estimators=200, subsample=0.8; total time=   3.7s
[CV] END colsample_bytree=0.8, learning_rate=0.01, max_depth=7,
```

```
n_estimators=100, subsample=0.8; total time=   2.9s
[CV] END colsample_bytree=0.8, learning_rate=0.01, max_depth=7,
n_estimators=200, subsample=0.8; total time=   5.5s
[CV] END colsample_bytree=0.8, learning_rate=0.01, max_depth=3,
n_estimators=200, subsample=0.8; total time=   2.8s
[CV] END colsample_bytree=0.8, learning_rate=0.01, max_depth=5,
n_estimators=100, subsample=0.8; total time=   2.1s
[CV] END colsample_bytree=0.8, learning_rate=0.01, max_depth=5,
n_estimators=100, subsample=1.0; total time=   2.0s
[CV] END colsample_bytree=0.8, learning_rate=0.01, max_depth=5,
n_estimators=300, subsample=1.0; total time=   5.5s
[CV] END colsample_bytree=0.8, learning_rate=0.01, max_depth=7,
n_estimators=200, subsample=0.8; total time=   5.2s
[CV] END colsample_bytree=0.8, learning_rate=0.01, max_depth=3,
n_estimators=200, subsample=0.8; total time=   2.8s
[CV] END colsample_bytree=0.8, learning_rate=0.01, max_depth=5,
n_estimators=100, subsample=0.8; total time=   2.1s
[CV] END colsample_bytree=0.8, learning_rate=0.01, max_depth=5,
n_estimators=100, subsample=1.0; total time=   2.0s
[CV] END colsample_bytree=0.8, learning_rate=0.01, max_depth=5,
n_estimators=300, subsample=0.8; total time=   5.5s
[CV] END colsample_bytree=0.8, learning_rate=0.01, max_depth=7,
n_estimators=200, subsample=1.0; total time=   5.4s
[CV] END colsample_bytree=0.8, learning_rate=0.01, max_depth=3,
n_estimators=100, subsample=0.8; total time=   1.6s
[CV] END colsample_bytree=0.8, learning_rate=0.01, max_depth=3,
n_estimators=300, subsample=1.0; total time=   3.9s
[CV] END colsample_bytree=0.8, learning_rate=0.01, max_depth=5,
n_estimators=200, subsample=1.0; total time=   3.7s
[CV] END colsample_bytree=0.8, learning_rate=0.01, max_depth=5,
n_estimators=300, subsample=1.0; total time=   5.3s
[CV] END colsample_bytree=0.8, learning_rate=0.01, max_depth=7,
n_estimators=300, subsample=0.8; total time=   7.5s
[CV] END colsample_bytree=0.8, learning_rate=0.01, max_depth=3,
n_estimators=200, subsample=1.0; total time=   2.8s
[CV] END colsample_bytree=0.8, learning_rate=0.01, max_depth=3,
n_estimators=300, subsample=1.0; total time=   3.8s
[CV] END colsample_bytree=0.8, learning_rate=0.01, max_depth=5,
n_estimators=300, subsample=0.8; total time=   5.2s
[CV] END colsample_bytree=0.8, learning_rate=0.01, max_depth=7,
n_estimators=100, subsample=1.0; total time=   2.7s
[CV] END colsample_bytree=0.8, learning_rate=0.01, max_depth=7,
n_estimators=300, subsample=0.8; total time=   7.4s
[CV] END colsample_bytree=0.8, learning_rate=0.01, max_depth=3,
n_estimators=100, subsample=1.0; total time=   1.6s
[CV] END colsample_bytree=0.8, learning_rate=0.01, max_depth=3,
n_estimators=300, subsample=0.8; total time=   3.9s
[CV] END colsample_bytree=0.8, learning_rate=0.01, max_depth=5,
```

```
n_estimators=200, subsample=0.8; total time=   3.7s
[CV] END colsample_bytree=0.8, learning_rate=0.01, max_depth=5,
n_estimators=300, subsample=1.0; total time=   5.3s
[CV] END colsample_bytree=0.8, learning_rate=0.01, max_depth=7,
n_estimators=300, subsample=0.8; total time=   7.9s
[CV] END colsample_bytree=0.8, learning_rate=0.01, max_depth=3,
n_estimators=200, subsample=0.8; total time=   2.9s
[CV] END colsample_bytree=0.8, learning_rate=0.01, max_depth=3,
n_estimators=300, subsample=1.0; total time=   3.8s
[CV] END colsample_bytree=0.8, learning_rate=0.01, max_depth=5,
n_estimators=300, subsample=0.8; total time=   5.5s
[CV] END colsample_bytree=0.8, learning_rate=0.01, max_depth=7,
n_estimators=200, subsample=0.8; total time=   5.3s
[CV] END colsample_bytree=0.8, learning_rate=0.01, max_depth=7,
n_estimators=300, subsample=1.0; total time=   7.2s
[CV] END colsample_bytree=0.8, learning_rate=0.01, max_depth=7,
n_estimators=200, subsample=1.0; total time=   5.1s
[CV] END colsample_bytree=0.8, learning_rate=0.1, max_depth=3, n_estimators=100,
subsample=0.8; total time=   1.4s
[CV] END colsample_bytree=0.8, learning_rate=0.1, max_depth=3, n_estimators=100,
subsample=1.0; total time=   1.4s
[CV] END colsample_bytree=0.8, learning_rate=0.1, max_depth=3, n_estimators=200,
subsample=1.0; total time=   2.4s
[CV] END colsample_bytree=0.8, learning_rate=0.1, max_depth=3, n_estimators=300,
subsample=1.0; total time=   3.3s
[CV] END colsample_bytree=0.8, learning_rate=0.01, max_depth=7,
n_estimators=200, subsample=1.0; total time=   5.1s
[CV] END colsample_bytree=0.8, learning_rate=0.1, max_depth=3, n_estimators=100,
subsample=1.0; total time=   1.5s
[CV] END colsample_bytree=0.8, learning_rate=0.1, max_depth=3, n_estimators=200,
subsample=0.8; total time=   2.4s
[CV] END colsample_bytree=0.8, learning_rate=0.1, max_depth=3, n_estimators=300,
subsample=0.8; total time=   3.3s
[CV] END colsample_bytree=0.8, learning_rate=0.1, max_depth=5, n_estimators=100,
subsample=1.0; total time=   1.8s
[CV] END colsample_bytree=0.8, learning_rate=0.1, max_depth=3, n_estimators=100,
subsample=0.8; total time=   1.4s
[CV] END colsample_bytree=0.8, learning_rate=0.1, max_depth=3, n_estimators=100,
subsample=1.0; total time=   1.4s
[CV] END colsample_bytree=0.8, learning_rate=0.1, max_depth=3, n_estimators=200,
subsample=0.8; total time=   2.4s
[CV] END colsample_bytree=0.8, learning_rate=0.1, max_depth=3, n_estimators=300,
subsample=1.0; total time=   3.2s
[CV] END colsample_bytree=0.8, learning_rate=0.1, max_depth=5, n_estimators=200,
subsample=1.0; total time=   3.0s
[CV] END colsample_bytree=0.8, learning_rate=0.1, max_depth=3, n_estimators=100,
subsample=0.8; total time=   1.5s
[CV] END colsample_bytree=0.8, learning_rate=0.1, max_depth=3, n_estimators=200,
```

```
subsample=0.8; total time=   2.5s
[CV] END colsample_bytree=0.8, learning_rate=0.1, max_depth=3, n_estimators=300,
subsample=0.8; total time=   3.4s
[CV] END colsample_bytree=0.8, learning_rate=0.1, max_depth=5, n_estimators=100,
subsample=1.0; total time=   1.9s
[CV] END colsample_bytree=0.8, learning_rate=0.1, max_depth=5, n_estimators=300,
subsample=0.8; total time=   4.7s
[CV] END colsample_bytree=0.8, learning_rate=0.1, max_depth=3, n_estimators=200,
subsample=1.0; total time=   2.2s
[CV] END colsample_bytree=0.8, learning_rate=0.1, max_depth=5, n_estimators=100,
subsample=0.8; total time=   1.9s
[CV] END colsample_bytree=0.8, learning_rate=0.1, max_depth=5, n_estimators=200,
subsample=1.0; total time=   3.1s
[CV] END colsample_bytree=0.8, learning_rate=0.1, max_depth=7, n_estimators=100,
subsample=0.8; total time=   2.4s
[CV] END colsample_bytree=0.8, learning_rate=0.1, max_depth=7, n_estimators=100,
subsample=1.0; total time=   2.4s
[CV] END colsample_bytree=0.8, learning_rate=0.1, max_depth=3, n_estimators=200,
subsample=1.0; total time=   2.3s
[CV] END colsample_bytree=0.8, learning_rate=0.1, max_depth=5, n_estimators=100,
subsample=0.8; total time=   1.8s
[CV] END colsample_bytree=0.8, learning_rate=0.1, max_depth=5, n_estimators=200,
subsample=0.8; total time=   3.2s
[CV] END colsample_bytree=0.8, learning_rate=0.1, max_depth=7, n_estimators=100,
subsample=0.8; total time=   2.5s
[CV] END colsample_bytree=0.8, learning_rate=0.1, max_depth=7, n_estimators=200,
subsample=0.8; total time=   4.1s
[CV] END colsample_bytree=0.8, learning_rate=0.1, max_depth=5, n_estimators=100,
subsample=0.8; total time=   2.0s
[CV] END colsample_bytree=0.8, learning_rate=0.1, max_depth=5, n_estimators=200,
subsample=1.0; total time=   3.1s
[CV] END colsample_bytree=0.8, learning_rate=0.1, max_depth=7, n_estimators=100,
subsample=0.8; total time=   2.5s
[CV] END colsample_bytree=0.8, learning_rate=0.1, max_depth=7, n_estimators=200,
subsample=0.8; total time=   4.3s
[CV] END colsample_bytree=0.8, learning_rate=0.2, max_depth=3, n_estimators=100,
subsample=0.8; total time=   1.5s
[CV] END colsample_bytree=0.8, learning_rate=0.1, max_depth=3, n_estimators=300,
subsample=0.8; total time=   3.5s
[CV] END colsample_bytree=0.8, learning_rate=0.1, max_depth=5, n_estimators=200,
subsample=0.8; total time=   3.2s
[CV] END colsample_bytree=0.8, learning_rate=0.1, max_depth=5, n_estimators=300,
subsample=1.0; total time=   4.5s
[CV] END colsample_bytree=0.8, learning_rate=0.1, max_depth=7, n_estimators=300,
subsample=0.8; total time=   5.9s
[CV] END colsample_bytree=0.8, learning_rate=0.2, max_depth=3, n_estimators=200,
subsample=0.8; total time=   2.5s
[CV] END colsample_bytree=0.8, learning_rate=0.01, max_depth=7,
```

```
n_estimators=300, subsample=1.0; total time=   7.6s
[CV] END colsample_bytree=0.8, learning_rate=0.1, max_depth=3, n_estimators=300,
subsample=1.0; total time=   3.5s
[CV] END colsample_bytree=0.8, learning_rate=0.1, max_depth=5, n_estimators=200,
subsample=0.8; total time=   3.3s
[CV] END colsample_bytree=0.8, learning_rate=0.1, max_depth=5, n_estimators=300,
subsample=1.0; total time=   4.7s
[CV] END colsample_bytree=0.8, learning_rate=0.1, max_depth=7, n_estimators=300,
subsample=0.8; total time=   6.1s
[CV] END colsample_bytree=0.8, learning_rate=0.2, max_depth=3, n_estimators=200,
subsample=1.0; total time=   2.3s
[CV] END colsample_bytree=0.8, learning_rate=0.1, max_depth=7, n_estimators=100,
subsample=1.0; total time=   2.3s
[CV] END colsample_bytree=0.8, learning_rate=0.1, max_depth=7, n_estimators=200,
subsample=1.0; total time=   4.1s
[CV] END colsample_bytree=0.8, learning_rate=0.2, max_depth=3, n_estimators=100,
subsample=0.8; total time=   1.5s
[CV] END colsample_bytree=0.8, learning_rate=0.2, max_depth=3, n_estimators=100,
subsample=1.0; total time=   1.5s
[CV] END colsample_bytree=0.8, learning_rate=0.2, max_depth=3, n_estimators=200,
subsample=1.0; total time=   2.5s
[CV] END colsample_bytree=0.8, learning_rate=0.01, max_depth=7,
n_estimators=300, subsample=1.0; total time=   7.0s
[CV] END colsample_bytree=0.8, learning_rate=0.1, max_depth=5, n_estimators=100,
subsample=1.0; total time=   1.9s
[CV] END colsample_bytree=0.8, learning_rate=0.1, max_depth=5, n_estimators=300,
subsample=0.8; total time=   4.7s
[CV] END colsample_bytree=0.8, learning_rate=0.1, max_depth=7, n_estimators=100,
subsample=1.0; total time=   2.4s
[CV] END colsample_bytree=0.8, learning_rate=0.1, max_depth=7, n_estimators=300,
subsample=0.8; total time=   6.6s
[CV] END colsample_bytree=0.8, learning_rate=0.2, max_depth=3, n_estimators=300,
subsample=0.8; total time=   3.5s
[CV] END colsample_bytree=0.8, learning_rate=0.1, max_depth=5, n_estimators=300,
subsample=1.0; total time=   4.7s
[CV] END colsample_bytree=0.8, learning_rate=0.1, max_depth=7, n_estimators=200,
subsample=1.0; total time=   4.2s
[CV] END colsample_bytree=0.8, learning_rate=0.2, max_depth=3, n_estimators=100,
subsample=1.0; total time=   1.4s
[CV] END colsample_bytree=0.8, learning_rate=0.2, max_depth=3, n_estimators=200,
subsample=0.8; total time=   2.6s
[CV] END colsample_bytree=0.8, learning_rate=0.2, max_depth=3, n_estimators=300,
subsample=0.8; total time=   3.4s
[CV] END colsample_bytree=0.8, learning_rate=0.1, max_depth=7, n_estimators=200,
subsample=1.0; total time=   4.0s
[CV] END colsample_bytree=0.8, learning_rate=0.2, max_depth=3, n_estimators=100,
subsample=0.8; total time=   1.5s
[CV] END colsample_bytree=0.8, learning_rate=0.2, max_depth=3, n_estimators=200,
```

```
subsample=0.8; total time=   2.5s
[CV] END colsample_bytree=0.8, learning_rate=0.2, max_depth=3, n_estimators=300,
subsample=0.8; total time=   3.5s
[CV] END colsample_bytree=0.8, learning_rate=0.2, max_depth=5, n_estimators=200,
subsample=0.8; total time=   3.2s
[CV] END colsample_bytree=0.8, learning_rate=0.1, max_depth=5, n_estimators=300,
subsample=0.8; total time=   4.5s
[CV] END colsample_bytree=0.8, learning_rate=0.1, max_depth=7, n_estimators=200,
subsample=0.8; total time=   4.4s
[CV] END colsample_bytree=0.8, learning_rate=0.1, max_depth=7, n_estimators=300,
subsample=1.0; total time=   6.2s
[CV] END colsample_bytree=0.8, learning_rate=0.2, max_depth=5, n_estimators=100,
subsample=0.8; total time=   1.8s
[CV] END colsample_bytree=0.8, learning_rate=0.2, max_depth=5, n_estimators=200,
subsample=0.8; total time=   3.3s
[CV] END colsample_bytree=0.8, learning_rate=0.2, max_depth=3, n_estimators=100,
subsample=1.0; total time=   1.4s
[CV] END colsample_bytree=0.8, learning_rate=0.2, max_depth=3, n_estimators=200,
subsample=1.0; total time=   2.3s
[CV] END colsample_bytree=0.8, learning_rate=0.2, max_depth=5, n_estimators=100,
subsample=0.8; total time=   1.9s
[CV] END colsample_bytree=0.8, learning_rate=0.2, max_depth=5, n_estimators=200,
subsample=0.8; total time=   3.3s
[CV] END colsample_bytree=0.8, learning_rate=0.2, max_depth=5, n_estimators=300,
subsample=1.0; total time=   4.5s
[CV] END colsample_bytree=0.8, learning_rate=0.1, max_depth=7, n_estimators=300,
subsample=1.0; total time=   6.2s
[CV] END colsample_bytree=0.8, learning_rate=0.2, max_depth=5, n_estimators=100,
subsample=1.0; total time=   1.8s
[CV] END colsample_bytree=0.8, learning_rate=0.2, max_depth=5, n_estimators=200,
subsample=1.0; total time=   3.2s
[CV] END colsample_bytree=0.8, learning_rate=0.2, max_depth=7, n_estimators=100,
subsample=1.0; total time=   2.5s
[CV] END colsample_bytree=0.8, learning_rate=0.2, max_depth=7, n_estimators=200,
subsample=0.8; total time=   4.0s
[CV] END colsample_bytree=0.8, learning_rate=0.1, max_depth=7, n_estimators=300,
subsample=1.0; total time=   5.8s
[CV] END colsample_bytree=0.8, learning_rate=0.2, max_depth=3, n_estimators=300,
subsample=1.0; total time=   3.4s
[CV] END colsample_bytree=0.8, learning_rate=0.2, max_depth=5, n_estimators=100,
subsample=1.0; total time=   1.7s
[CV] END colsample_bytree=0.8, learning_rate=0.2, max_depth=5, n_estimators=300,
subsample=0.8; total time=   4.8s
[CV] END colsample_bytree=0.8, learning_rate=0.2, max_depth=7, n_estimators=200,
subsample=0.8; total time=   4.6s
[CV] END colsample_bytree=0.8, learning_rate=0.2, max_depth=5, n_estimators=100,
subsample=1.0; total time=   1.7s
[CV] END colsample_bytree=0.8, learning_rate=0.2, max_depth=5, n_estimators=300,
```

```
subsample=1.0; total time=   4.5s
[CV] END colsample_bytree=0.8, learning_rate=0.2, max_depth=7, n_estimators=200,
subsample=0.8; total time=   4.4s
[CV] END colsample_bytree=1.0, learning_rate=0.01, max_depth=3,
n_estimators=100, subsample=0.8; total time=   1.6s
[CV] END colsample_bytree=1.0, learning_rate=0.01, max_depth=3,
n_estimators=100, subsample=1.0; total time=   1.8s
[CV] END colsample_bytree=0.8, learning_rate=0.2, max_depth=3, n_estimators=300,
subsample=1.0; total time=   3.3s
[CV] END colsample_bytree=0.8, learning_rate=0.2, max_depth=5, n_estimators=300,
subsample=0.8; total time=   4.8s
[CV] END colsample_bytree=0.8, learning_rate=0.2, max_depth=7, n_estimators=200,
subsample=1.0; total time=   4.3s
[CV] END colsample_bytree=1.0, learning_rate=0.01, max_depth=3,
n_estimators=100, subsample=0.8; total time=   1.8s
[CV] END colsample_bytree=1.0, learning_rate=0.01, max_depth=3,
n_estimators=100, subsample=1.0; total time=   1.7s
[CV] END colsample_bytree=0.8, learning_rate=0.2, max_depth=5, n_estimators=100,
subsample=0.8; total time=   1.8s
[CV] END colsample_bytree=0.8, learning_rate=0.2, max_depth=5, n_estimators=200,
subsample=1.0; total time=   3.0s
[CV] END colsample_bytree=0.8, learning_rate=0.2, max_depth=7, n_estimators=100,
subsample=0.8; total time=   2.5s
[CV] END colsample_bytree=0.8, learning_rate=0.2, max_depth=7, n_estimators=100,
subsample=1.0; total time=   2.2s
[CV] END colsample_bytree=0.8, learning_rate=0.2, max_depth=7, n_estimators=300,
subsample=0.8; total time=   7.1s
[CV] END colsample_bytree=0.8, learning_rate=0.2, max_depth=3, n_estimators=300,
subsample=1.0; total time=   3.3s
[CV] END colsample_bytree=0.8, learning_rate=0.2, max_depth=5, n_estimators=300,
subsample=0.8; total time=   4.7s
[CV] END colsample_bytree=0.8, learning_rate=0.2, max_depth=7, n_estimators=200,
subsample=1.0; total time=   4.1s
[CV] END colsample_bytree=0.8, learning_rate=0.2, max_depth=7, n_estimators=300,
subsample=1.0; total time=   5.9s
[CV] END colsample_bytree=0.8, learning_rate=0.2, max_depth=5, n_estimators=200,
subsample=1.0; total time=   3.1s
[CV] END colsample_bytree=0.8, learning_rate=0.2, max_depth=7, n_estimators=100,
subsample=0.8; total time=   2.4s
[CV] END colsample_bytree=0.8, learning_rate=0.2, max_depth=7, n_estimators=100,
subsample=1.0; total time=   2.4s
[CV] END colsample_bytree=0.8, learning_rate=0.2, max_depth=7, n_estimators=300,
subsample=1.0; total time=   6.2s
[CV] END colsample_bytree=1.0, learning_rate=0.01, max_depth=3,
n_estimators=200, subsample=1.0; total time=   2.8s
[CV] END colsample_bytree=0.8, learning_rate=0.2, max_depth=7, n_estimators=100,
subsample=0.8; total time=   2.5s
[CV] END colsample_bytree=0.8, learning_rate=0.2, max_depth=7, n_estimators=200,
```

```
subsample=1.0; total time=   4.2s
[CV] END colsample_bytree=1.0, learning_rate=0.01, max_depth=3,
n_estimators=100, subsample=0.8; total time=   1.7s
[CV] END colsample_bytree=1.0, learning_rate=0.01, max_depth=3,
n_estimators=200, subsample=0.8; total time=   3.0s
[CV] END colsample_bytree=1.0, learning_rate=0.01, max_depth=3,
n_estimators=300, subsample=0.8; total time=   4.0s
[CV] END colsample_bytree=0.8, learning_rate=0.2, max_depth=5, n_estimators=300,
subsample=1.0; total time=   4.4s
[CV] END colsample_bytree=0.8, learning_rate=0.2, max_depth=7, n_estimators=300,
subsample=0.8; total time=   6.9s
[CV] END colsample_bytree=1.0, learning_rate=0.01, max_depth=3,
n_estimators=300, subsample=0.8; total time=   4.1s
[CV] END colsample_bytree=1.0, learning_rate=0.01, max_depth=5,
n_estimators=100, subsample=1.0; total time=   2.3s
[CV] END colsample_bytree=1.0, learning_rate=0.01, max_depth=5,
n_estimators=300, subsample=0.8; total time=   5.7s
[CV] END colsample_bytree=1.0, learning_rate=0.01, max_depth=3,
n_estimators=100, subsample=1.0; total time=   1.6s
[CV] END colsample_bytree=1.0, learning_rate=0.01, max_depth=3,
n_estimators=200, subsample=0.8; total time=   2.9s
[CV] END colsample_bytree=1.0, learning_rate=0.01, max_depth=3,
n_estimators=300, subsample=1.0; total time=   4.0s
[CV] END colsample_bytree=1.0, learning_rate=0.01, max_depth=5,
n_estimators=200, subsample=1.0; total time=   3.9s
[CV] END colsample_bytree=1.0, learning_rate=0.01, max_depth=7,
n_estimators=100, subsample=0.8; total time=   3.0s
[CV] END colsample_bytree=1.0, learning_rate=0.01, max_depth=3,
n_estimators=200, subsample=0.8; total time=   2.9s
[CV] END colsample_bytree=1.0, learning_rate=0.01, max_depth=5,
n_estimators=100, subsample=0.8; total time=   2.4s
[CV] END colsample_bytree=1.0, learning_rate=0.01, max_depth=5,
n_estimators=200, subsample=1.0; total time=   4.0s
[CV] END colsample_bytree=1.0, learning_rate=0.01, max_depth=7,
n_estimators=100, subsample=0.8; total time=   3.0s
[CV] END colsample_bytree=1.0, learning_rate=0.01, max_depth=7,
n_estimators=200, subsample=0.8; total time=   5.6s
[CV] END colsample_bytree=1.0, learning_rate=0.01, max_depth=3,
n_estimators=200, subsample=1.0; total time=   2.8s
[CV] END colsample_bytree=1.0, learning_rate=0.01, max_depth=5,
n_estimators=100, subsample=1.0; total time=   2.3s
[CV] END colsample_bytree=1.0, learning_rate=0.01, max_depth=5,
n_estimators=200, subsample=0.8; total time=   4.1s
[CV] END colsample_bytree=1.0, learning_rate=0.01, max_depth=7,
n_estimators=100, subsample=0.8; total time=   3.0s
[CV] END colsample_bytree=1.0, learning_rate=0.01, max_depth=7,
n_estimators=200, subsample=0.8; total time=   5.9s
[CV] END colsample_bytree=0.8, learning_rate=0.2, max_depth=7, n_estimators=300,
```

```
subsample=0.8; total time=   6.5s
[CV] END colsample_bytree=1.0, learning_rate=0.01, max_depth=3,
n_estimators=200, subsample=1.0; total time=   2.8s
[CV] END colsample_bytree=1.0, learning_rate=0.01, max_depth=5,
n_estimators=100, subsample=0.8; total time=   2.3s
[CV] END colsample_bytree=1.0, learning_rate=0.01, max_depth=5,
n_estimators=200, subsample=0.8; total time=   4.0s
[CV] END colsample_bytree=1.0, learning_rate=0.01, max_depth=5,
n_estimators=300, subsample=1.0; total time=   5.6s
[CV] END colsample_bytree=1.0, learning_rate=0.01, max_depth=7,
n_estimators=300, subsample=0.8; total time=   8.5s
[CV] END colsample_bytree=1.0, learning_rate=0.01, max_depth=5,
n_estimators=100, subsample=1.0; total time=   2.2s
[CV] END colsample_bytree=1.0, learning_rate=0.01, max_depth=5,
n_estimators=300, subsample=1.0; total time=   5.6s
[CV] END colsample_bytree=1.0, learning_rate=0.01, max_depth=7,
n_estimators=200, subsample=1.0; total time=   5.6s
[CV] END colsample_bytree=1.0, learning_rate=0.1, max_depth=3, n_estimators=100,
subsample=1.0; total time=   1.5s
[CV] END colsample_bytree=1.0, learning_rate=0.1, max_depth=3, n_estimators=200,
subsample=0.8; total time=   2.7s
[CV] END colsample_bytree=0.8, learning_rate=0.2, max_depth=7, n_estimators=300,
subsample=1.0; total time=   6.3s
[CV] END colsample_bytree=1.0, learning_rate=0.01, max_depth=3,
n_estimators=300, subsample=1.0; total time=   4.0s
[CV] END colsample_bytree=1.0, learning_rate=0.01, max_depth=5,
n_estimators=300, subsample=0.8; total time=   5.7s
[CV] END colsample_bytree=1.0, learning_rate=0.01, max_depth=7,
n_estimators=100, subsample=1.0; total time=   3.0s
[CV] END colsample_bytree=1.0, learning_rate=0.01, max_depth=7,
n_estimators=300, subsample=1.0; total time=   8.5s
[CV] END colsample_bytree=1.0, learning_rate=0.01, max_depth=3,
n_estimators=300, subsample=0.8; total time=   4.1s
[CV] END colsample_bytree=1.0, learning_rate=0.01, max_depth=5,
n_estimators=200, subsample=0.8; total time=   4.1s
[CV] END colsample_bytree=1.0, learning_rate=0.01, max_depth=5,
n_estimators=300, subsample=1.0; total time=   5.6s
[CV] END colsample_bytree=1.0, learning_rate=0.01, max_depth=7,
n_estimators=300, subsample=0.8; total time=   8.0s
[CV] END colsample_bytree=1.0, learning_rate=0.1, max_depth=3, n_estimators=200,
subsample=1.0; total time=   2.3s
[CV] END colsample_bytree=1.0, learning_rate=0.01, max_depth=3,
n_estimators=300, subsample=1.0; total time=   3.8s
[CV] END colsample_bytree=1.0, learning_rate=0.01, max_depth=5,
n_estimators=300, subsample=0.8; total time=   5.9s
[CV] END colsample_bytree=1.0, learning_rate=0.01, max_depth=7,
n_estimators=100, subsample=1.0; total time=   2.9s
[CV] END colsample_bytree=1.0, learning_rate=0.01, max_depth=7,
```

```
n_estimators=300, subsample=0.8; total time=   8.1s
[CV] END colsample_bytree=1.0, learning_rate=0.1, max_depth=3, n_estimators=200,
subsample=1.0; total time=   2.4s
[CV] END colsample_bytree=1.0, learning_rate=0.01, max_depth=5,
n_estimators=100, subsample=0.8; total time=   2.3s
[CV] END colsample_bytree=1.0, learning_rate=0.01, max_depth=5,
n_estimators=200, subsample=1.0; total time=   3.9s
[CV] END colsample_bytree=1.0, learning_rate=0.01, max_depth=7,
n_estimators=100, subsample=1.0; total time=   3.1s
[CV] END colsample_bytree=1.0, learning_rate=0.01, max_depth=7,
n_estimators=200, subsample=1.0; total time=   5.8s
[CV] END colsample_bytree=1.0, learning_rate=0.01, max_depth=7,
n_estimators=300, subsample=1.0; total time=   7.9s
[CV] END colsample_bytree=1.0, learning_rate=0.01, max_depth=7,
n_estimators=200, subsample=1.0; total time=   5.5s
[CV] END colsample_bytree=1.0, learning_rate=0.1, max_depth=3, n_estimators=100,
subsample=0.8; total time=   1.5s
[CV] END colsample_bytree=1.0, learning_rate=0.1, max_depth=3, n_estimators=100,
subsample=1.0; total time=   1.6s
[CV] END colsample_bytree=1.0, learning_rate=0.1, max_depth=3, n_estimators=200,
subsample=0.8; total time=   2.5s
[CV] END colsample_bytree=1.0, learning_rate=0.1, max_depth=3, n_estimators=300,
subsample=1.0; total time=   3.4s
[CV] END colsample_bytree=1.0, learning_rate=0.01, max_depth=7,
n_estimators=200, subsample=0.8; total time=   5.6s
[CV] END colsample_bytree=1.0, learning_rate=0.1, max_depth=3, n_estimators=100,
subsample=0.8; total time=   1.5s
[CV] END colsample_bytree=1.0, learning_rate=0.1, max_depth=3, n_estimators=100,
subsample=1.0; total time=   1.5s
[CV] END colsample_bytree=1.0, learning_rate=0.1, max_depth=3, n_estimators=200,
subsample=1.0; total time=   2.5s
[CV] END colsample_bytree=1.0, learning_rate=0.1, max_depth=3, n_estimators=300,
subsample=1.0; total time=   3.4s
[CV] END colsample_bytree=1.0, learning_rate=0.1, max_depth=5, n_estimators=200,
subsample=0.8; total time=   3.4s
[CV] END colsample_bytree=1.0, learning_rate=0.1, max_depth=3, n_estimators=100,
subsample=0.8; total time=   1.6s
[CV] END colsample_bytree=1.0, learning_rate=0.1, max_depth=3, n_estimators=200,
subsample=0.8; total time=   2.5s
[CV] END colsample_bytree=1.0, learning_rate=0.1, max_depth=3, n_estimators=300,
subsample=0.8; total time=   3.7s
[CV] END colsample_bytree=1.0, learning_rate=0.1, max_depth=5, n_estimators=100,
subsample=1.0; total time=   1.9s
[CV] END colsample_bytree=1.0, learning_rate=0.1, max_depth=5, n_estimators=300,
subsample=0.8; total time=   4.9s
[CV] END colsample_bytree=1.0, learning_rate=0.1, max_depth=7, n_estimators=100,
subsample=1.0; total time=   2.5s
[CV] END colsample_bytree=1.0, learning_rate=0.1, max_depth=5, n_estimators=100,
```

```
subsample=0.8; total time=   2.0s
[CV] END colsample_bytree=1.0, learning_rate=0.1, max_depth=5, n_estimators=200,
subsample=1.0; total time=   3.3s
[CV] END colsample_bytree=1.0, learning_rate=0.1, max_depth=7, n_estimators=100,
subsample=0.8; total time=   2.7s
[CV] END colsample_bytree=1.0, learning_rate=0.1, max_depth=7, n_estimators=200,
subsample=0.8; total time=   4.6s
[CV] END colsample_bytree=1.0, learning_rate=0.2, max_depth=3, n_estimators=100,
subsample=0.8; total time=   1.5s
[CV] END colsample_bytree=1.0, learning_rate=0.1, max_depth=3, n_estimators=300,
subsample=0.8; total time=   3.4s
[CV] END colsample_bytree=1.0, learning_rate=0.1, max_depth=5, n_estimators=100,
subsample=1.0; total time=   1.9s
[CV] END colsample_bytree=1.0, learning_rate=0.1, max_depth=5, n_estimators=300,
subsample=1.0; total time=   4.8s
[CV] END colsample_bytree=1.0, learning_rate=0.1, max_depth=7, n_estimators=200,
subsample=1.0; total time=   4.7s
[CV] END colsample_bytree=1.0, learning_rate=0.2, max_depth=3, n_estimators=100,
subsample=0.8; total time=   1.5s
[CV] END colsample_bytree=1.0, learning_rate=0.1, max_depth=5, n_estimators=200,
subsample=1.0; total time=   3.2s
[CV] END colsample_bytree=1.0, learning_rate=0.1, max_depth=7, n_estimators=100,
subsample=1.0; total time=   2.7s
[CV] END colsample_bytree=1.0, learning_rate=0.1, max_depth=7, n_estimators=200,
subsample=1.0; total time=   4.4s
[CV] END colsample_bytree=1.0, learning_rate=0.2, max_depth=3, n_estimators=100,
subsample=0.8; total time=   1.4s
[CV] END colsample_bytree=1.0, learning_rate=0.2, max_depth=3, n_estimators=100,
subsample=1.0; total time=   1.4s
[CV] END colsample_bytree=1.0, learning_rate=0.01, max_depth=7,
n_estimators=300, subsample=1.0; total time=   7.8s
[CV] END colsample_bytree=1.0, learning_rate=0.1, max_depth=5, n_estimators=100,
subsample=0.8; total time=   1.9s
[CV] END colsample_bytree=1.0, learning_rate=0.1, max_depth=5, n_estimators=300,
subsample=0.8; total time=   5.0s
[CV] END colsample_bytree=1.0, learning_rate=0.1, max_depth=7, n_estimators=100,
subsample=1.0; total time=   2.5s
[CV] END colsample_bytree=1.0, learning_rate=0.1, max_depth=7, n_estimators=300,
subsample=0.8; total time=   6.9s
[CV] END colsample_bytree=1.0, learning_rate=0.1, max_depth=3, n_estimators=300,
subsample=0.8; total time=   3.4s
[CV] END colsample_bytree=1.0, learning_rate=0.1, max_depth=5, n_estimators=200,
subsample=0.8; total time=   3.3s
[CV] END colsample_bytree=1.0, learning_rate=0.1, max_depth=5, n_estimators=300,
subsample=1.0; total time=   4.6s
[CV] END colsample_bytree=1.0, learning_rate=0.1, max_depth=7, n_estimators=300,
subsample=0.8; total time=   6.5s
[CV] END colsample_bytree=1.0, learning_rate=0.2, max_depth=3, n_estimators=200,
```

```
subsample=1.0; total time=   2.3s
[CV] END colsample_bytree=1.0, learning_rate=0.1, max_depth=3, n_estimators=300,
subsample=1.0; total time=   3.6s
[CV] END colsample_bytree=1.0, learning_rate=0.1, max_depth=5, n_estimators=200,
subsample=0.8; total time=   3.3s
[CV] END colsample_bytree=1.0, learning_rate=0.1, max_depth=5, n_estimators=300,
subsample=1.0; total time=   4.5s
[CV] END colsample_bytree=1.0, learning_rate=0.1, max_depth=7, n_estimators=300,
subsample=0.8; total time=   6.7s
[CV] END colsample_bytree=1.0, learning_rate=0.2, max_depth=3, n_estimators=200,
subsample=1.0; total time=   2.1s
[CV] END colsample_bytree=1.0, learning_rate=0.1, max_depth=5, n_estimators=100,
subsample=0.8; total time=   2.0s
[CV] END colsample_bytree=1.0, learning_rate=0.1, max_depth=5, n_estimators=200,
subsample=1.0; total time=   3.4s
[CV] END colsample_bytree=1.0, learning_rate=0.1, max_depth=7, n_estimators=100,
subsample=0.8; total time=   2.6s
[CV] END colsample_bytree=1.0, learning_rate=0.1, max_depth=7, n_estimators=200,
subsample=0.8; total time=   4.6s
[CV] END colsample_bytree=1.0, learning_rate=0.1, max_depth=7, n_estimators=300,
subsample=1.0; total time=   6.6s
[CV] END colsample_bytree=1.0, learning_rate=0.1, max_depth=5, n_estimators=100,
subsample=1.0; total time=   2.1s
[CV] END colsample_bytree=1.0, learning_rate=0.1, max_depth=5, n_estimators=300,
subsample=0.8; total time=   4.8s
[CV] END colsample_bytree=1.0, learning_rate=0.1, max_depth=7, n_estimators=200,
subsample=0.8; total time=   4.6s
[CV] END colsample_bytree=1.0, learning_rate=0.1, max_depth=7, n_estimators=300,
subsample=1.0; total time=   6.2s
[CV] END colsample_bytree=1.0, learning_rate=0.2, max_depth=5, n_estimators=100,
subsample=0.8; total time=   1.8s
[CV] END colsample_bytree=1.0, learning_rate=0.1, max_depth=7, n_estimators=100,
subsample=0.8; total time=   2.6s
[CV] END colsample_bytree=1.0, learning_rate=0.1, max_depth=7, n_estimators=200,
subsample=1.0; total time=   4.3s
[CV] END colsample_bytree=1.0, learning_rate=0.2, max_depth=3, n_estimators=100,
subsample=1.0; total time=   1.4s
[CV] END colsample_bytree=1.0, learning_rate=0.2, max_depth=3, n_estimators=100,
subsample=1.0; total time=   1.4s
[CV] END colsample_bytree=1.0, learning_rate=0.2, max_depth=3, n_estimators=200,
subsample=0.8; total time=   2.4s
[CV] END colsample_bytree=1.0, learning_rate=0.2, max_depth=3, n_estimators=300,
subsample=1.0; total time=   3.3s
[CV] END colsample_bytree=1.0, learning_rate=0.2, max_depth=3, n_estimators=200,
subsample=0.8; total time=   2.4s
[CV] END colsample_bytree=1.0, learning_rate=0.2, max_depth=3, n_estimators=300,
subsample=1.0; total time=   3.6s
[CV] END colsample_bytree=1.0, learning_rate=0.2, max_depth=5, n_estimators=200,
```

```
subsample=1.0; total time=   3.2s
[CV] END colsample_bytree=1.0, learning_rate=0.2, max_depth=7, n_estimators=100,
subsample=0.8; total time=   2.5s
[CV] END colsample_bytree=1.0, learning_rate=0.2, max_depth=7, n_estimators=100,
subsample=1.0; total time=   2.3s
[CV] END colsample_bytree=1.0, learning_rate=0.2, max_depth=3, n_estimators=200,
subsample=1.0; total time=   2.5s
[CV] END colsample_bytree=1.0, learning_rate=0.2, max_depth=5, n_estimators=100,
subsample=0.8; total time=   1.9s
[CV] END colsample_bytree=1.0, learning_rate=0.2, max_depth=5, n_estimators=100,
subsample=1.0; total time=   1.8s
[CV] END colsample_bytree=1.0, learning_rate=0.2, max_depth=5, n_estimators=300,
subsample=0.8; total time=   4.9s
[CV] END colsample_bytree=1.0, learning_rate=0.2, max_depth=7, n_estimators=200,
subsample=1.0; total time=   4.5s
[CV] END colsample_bytree=1.0, learning_rate=0.2, max_depth=3, n_estimators=200,
subsample=0.8; total time=   2.5s
[CV] END colsample_bytree=1.0, learning_rate=0.2, max_depth=3, n_estimators=300,
subsample=0.8; total time=   3.4s
[CV] END colsample_bytree=1.0, learning_rate=0.2, max_depth=5, n_estimators=200,
subsample=0.8; total time=   3.4s
[CV] END colsample_bytree=1.0, learning_rate=0.2, max_depth=5, n_estimators=300,
subsample=1.0; total time=   4.7s
[CV] END colsample_bytree=1.0, learning_rate=0.2, max_depth=7, n_estimators=300,
subsample=0.8; total time=   6.8s
[CV] END colsample_bytree=1.0, learning_rate=0.2, max_depth=5, n_estimators=100,
subsample=0.8; total time=   2.0s
[CV] END colsample_bytree=1.0, learning_rate=0.2, max_depth=5, n_estimators=200,
subsample=1.0; total time=   3.1s
[CV] END colsample_bytree=1.0, learning_rate=0.2, max_depth=7, n_estimators=100,
subsample=0.8; total time=   2.6s
[CV] END colsample_bytree=1.0, learning_rate=0.2, max_depth=7, n_estimators=200,
subsample=0.8; total time=   4.3s
[CV] END colsample_bytree=1.0, learning_rate=0.2, max_depth=7, n_estimators=300,
subsample=1.0; total time=   5.5s
[CV] END colsample_bytree=1.0, learning_rate=0.2, max_depth=3, n_estimators=300,
subsample=0.8; total time=   3.6s
[CV] END colsample_bytree=1.0, learning_rate=0.2, max_depth=5, n_estimators=200,
subsample=0.8; total time=   3.3s
[CV] END colsample_bytree=1.0, learning_rate=0.2, max_depth=7, n_estimators=100,
subsample=0.8; total time=   2.6s
[CV] END colsample_bytree=1.0, learning_rate=0.2, max_depth=7, n_estimators=200,
subsample=0.8; total time=   4.7s
[CV] END colsample_bytree=1.0, learning_rate=0.2, max_depth=7, n_estimators=300,
subsample=1.0; total time=   5.4s
Fitting 3 folds for each of 81 candidates, totalling 243 fits
[CV] END colsample_bytree=1.0, learning_rate=0.1, max_depth=7, n_estimators=300,
subsample=1.0; total time=   6.6s
```

```
[CV] END colsample_bytree=1.0, learning_rate=0.2, max_depth=3, n_estimators=300,
subsample=0.8; total time=   3.5s
[CV] END colsample_bytree=1.0, learning_rate=0.2, max_depth=5, n_estimators=200,
subsample=0.8; total time=   3.3s
[CV] END colsample_bytree=1.0, learning_rate=0.2, max_depth=5, n_estimators=300,
subsample=1.0; total time=   4.6s
[CV] END colsample_bytree=1.0, learning_rate=0.2, max_depth=7, n_estimators=300,
subsample=0.8; total time=   6.1s
[CV] END max_depth=None, min_samples_leaf=1, min_samples_split=2,
n_estimators=100; total time=  14.6s
```

```
/opt/anaconda3/lib/python3.11/site-
packages/joblib/externals/loky/process_executor.py:752: UserWarning: A worker
stopped while some jobs were given to the executor. This can be caused by a too
short worker timeout or by a memory leak.
  warnings.warn(
```

```
[CV] END colsample_bytree=1.0, learning_rate=0.2, max_depth=5, n_estimators=100,
subsample=1.0; total time=   1.7s
[CV] END colsample_bytree=1.0, learning_rate=0.2, max_depth=5, n_estimators=300,
subsample=1.0; total time=   4.6s
[CV] END colsample_bytree=1.0, learning_rate=0.2, max_depth=7, n_estimators=200,
subsample=1.0; total time=   4.3s
[CV] END max_depth=None, min_samples_leaf=1, min_samples_split=2,
n_estimators=100; total time=  13.9s
[CV] END max_depth=None, min_samples_leaf=1, min_samples_split=5,
n_estimators=100; total time=  14.2s
[CV] END colsample_bytree=1.0, learning_rate=0.2, max_depth=7, n_estimators=300,
subsample=1.0; total time=   5.5s
[CV] END max_depth=None, min_samples_leaf=1, min_samples_split=2,
n_estimators=100; total time=  14.6s
[CV] END max_depth=None, min_samples_leaf=1, min_samples_split=5,
n_estimators=100; total time=  14.3s
[CV] END colsample_bytree=1.0, learning_rate=0.2, max_depth=5, n_estimators=100,
subsample=1.0; total time=   1.8s
[CV] END colsample_bytree=1.0, learning_rate=0.2, max_depth=5, n_estimators=300,
subsample=0.8; total time=   4.9s
[CV] END colsample_bytree=1.0, learning_rate=0.2, max_depth=7, n_estimators=100,
subsample=1.0; total time=   2.5s
[CV] END colsample_bytree=1.0, learning_rate=0.2, max_depth=7, n_estimators=300,
subsample=0.8; total time=   6.5s
[CV] END max_depth=None, min_samples_leaf=1, min_samples_split=2,
n_estimators=200; total time=  29.2s
[CV] END colsample_bytree=1.0, learning_rate=0.2, max_depth=3, n_estimators=300,
subsample=1.0; total time=   3.3s
[CV] END colsample_bytree=1.0, learning_rate=0.2, max_depth=5, n_estimators=300,
subsample=0.8; total time=   5.0s
[CV] END colsample_bytree=1.0, learning_rate=0.2, max_depth=7, n_estimators=200,
subsample=0.8; total time=   4.8s
```

```
[CV] END max_depth=None, min_samples_leaf=1, min_samples_split=2,
n_estimators=300; total time=  42.3s
[CV] END colsample_bytree=1.0, learning_rate=0.2, max_depth=5, n_estimators=200,
subsample=1.0; total time=   3.0s
[CV] END colsample_bytree=1.0, learning_rate=0.2, max_depth=7, n_estimators=100,
subsample=1.0; total time=   2.4s
[CV] END colsample_bytree=1.0, learning_rate=0.2, max_depth=7, n_estimators=200,
subsample=1.0; total time=   4.4s
[CV] END max_depth=None, min_samples_leaf=1, min_samples_split=2,
n_estimators=300; total time=  43.3s
[CV] END max_depth=None, min_samples_leaf=1, min_samples_split=2,
n_estimators=200; total time=  28.0s
[CV] END max_depth=None, min_samples_leaf=1, min_samples_split=5,
n_estimators=300; total time=  41.8s
[CV] END max_depth=None, min_samples_leaf=2, min_samples_split=2,
n_estimators=100; total time=  13.8s
[CV] END max_depth=None, min_samples_leaf=2, min_samples_split=2,
n_estimators=300; total time=  41.3s
[CV] END max_depth=None, min_samples_leaf=1, min_samples_split=10,
n_estimators=100; total time=  13.8s
[CV] END max_depth=None, min_samples_leaf=1, min_samples_split=10,
n_estimators=300; total time=  41.2s
[CV] END max_depth=None, min_samples_leaf=2, min_samples_split=5,
n_estimators=200; total time=  27.5s
[CV] END max_depth=None, min_samples_leaf=2, min_samples_split=10,
n_estimators=200; total time=  26.8s
[CV] END max_depth=None, min_samples_leaf=4, min_samples_split=2,
n_estimators=300; total time=  39.8s
[CV] END max_depth=None, min_samples_leaf=4, min_samples_split=10,
n_estimators=200; total time=  27.0s
[CV] END max_depth=10, min_samples_leaf=1, min_samples_split=2,
n_estimators=300; total time=  25.4s
[CV] END max_depth=10, min_samples_leaf=1, min_samples_split=10,
n_estimators=100; total time=   9.2s
[CV] END max_depth=10, min_samples_leaf=2, min_samples_split=2,
n_estimators=100; total time=   9.2s
[CV] END max_depth=10, min_samples_leaf=2, min_samples_split=2,
n_estimators=100; total time=   8.5s
[CV] END max_depth=10, min_samples_leaf=2, min_samples_split=2,
n_estimators=300; total time=  25.6s
[CV] END max_depth=10, min_samples_leaf=2, min_samples_split=10,
n_estimators=200; total time=  17.1s
[CV] END max_depth=10, min_samples_leaf=4, min_samples_split=2,
n_estimators=200; total time=  17.2s
[CV] END max_depth=10, min_samples_leaf=4, min_samples_split=5,
n_estimators=300; total time=  25.3s
[CV] END max_depth=20, min_samples_leaf=1, min_samples_split=2,
n_estimators=100; total time=  12.9s
```

```
[CV] END max_depth=20, min_samples_leaf=1, min_samples_split=2,
n_estimators=300; total time=  38.0s
Fitting 3 folds for each of 54 candidates, totalling 162 fits
[CV] END max_depth=20, min_samples_leaf=1, min_samples_split=10,
n_estimators=200; total time=  24.4s
[CV] END max_depth=20, min_samples_leaf=2, min_samples_split=2,
n_estimators=300; total time=  36.9s
[CV] END max_depth=20, min_samples_leaf=2, min_samples_split=10,
n_estimators=100; total time=  12.4s
[CV] END max_depth=20, min_samples_leaf=4, min_samples_split=2,
n_estimators=100; total time=  12.3s
[CV] END max_depth=20, min_samples_leaf=4, min_samples_split=2,
n_estimators=200; total time=  25.1s
[CV] END max_depth=20, min_samples_leaf=4, min_samples_split=5,
n_estimators=300; total time=  35.2s
[CV] END learning_rate=0.01, max_depth=3, n_estimators=100, subsample=0.8; total
time=  32.9s
[CV] END learning_rate=0.01, max_depth=3, n_estimators=200, subsample=1.0; total
time= 1.3min
[CV] END learning_rate=0.01, max_depth=5, n_estimators=100, subsample=1.0; total
time= 1.0min
[CV] END learning_rate=0.01, max_depth=5, n_estimators=200, subsample=1.0; total
time= 1.9min
[CV] END learning_rate=0.01, max_depth=7, n_estimators=100, subsample=1.0; total
time= 1.3min
[CV] END learning_rate=0.01, max_depth=7, n_estimators=200, subsample=1.0; total
time= 2.4min
[CV] END learning_rate=0.1, max_depth=3, n_estimators=100, subsample=0.8; total
time=  31.9s
[CV] END learning_rate=0.1, max_depth=3, n_estimators=200, subsample=0.8; total
time= 1.2min
[CV] END learning_rate=0.1, max_depth=3, n_estimators=300, subsample=0.8; total
time= 1.6min
[CV] END learning_rate=0.1, max_depth=5, n_estimators=100, subsample=1.0; total
time= 1.1min
[CV] END learning_rate=0.1, max_depth=5, n_estimators=300, subsample=0.8; total
time= 2.8min
[CV] END learning_rate=0.1, max_depth=7, n_estimators=200, subsample=0.8; total
time= 2.5min
[CV] END learning_rate=0.2, max_depth=3, n_estimators=100, subsample=0.8; total
time=  35.1s
[CV] END learning_rate=0.2, max_depth=3, n_estimators=100, subsample=0.8; total
time=  32.6s
[CV] END learning_rate=0.2, max_depth=3, n_estimators=100, subsample=1.0; total
time=  40.7s
[CV] END learning_rate=0.2, max_depth=3, n_estimators=200, subsample=0.8; total
time= 1.1min
[CV] END learning_rate=0.2, max_depth=3, n_estimators=300, subsample=1.0; total
```

```
time= 2.2min
[CV] END learning_rate=0.2, max_depth=5, n_estimators=200, subsample=1.0; total
time= 2.2min
[CV] END learning_rate=0.2, max_depth=7, n_estimators=100, subsample=0.8; total
time= 1.1min
[CV] END learning_rate=0.2, max_depth=7, n_estimators=200, subsample=0.8; total
time= 2.3min
[CV] END max_depth=None, min_samples_leaf=1, min_samples_split=5,
n_estimators=200; total time=  28.3s
[CV] END max_depth=None, min_samples_leaf=2, min_samples_split=2,
n_estimators=100; total time=  13.9s
[CV] END max_depth=None, min_samples_leaf=2, min_samples_split=2,
n_estimators=200; total time=  27.7s
[CV] END max_depth=None, min_samples_leaf=2, min_samples_split=5,
n_estimators=300; total time=  41.0s
[CV] END max_depth=None, min_samples_leaf=4, min_samples_split=2,
n_estimators=100; total time=  13.2s
[CV] END max_depth=None, min_samples_leaf=4, min_samples_split=2,
n_estimators=300; total time=  38.8s
[CV] END max_depth=None, min_samples_leaf=4, min_samples_split=10,
n_estimators=100; total time=  12.9s
[CV] END max_depth=None, min_samples_leaf=4, min_samples_split=10,
n_estimators=300; total time=  39.9s
[CV] END max_depth=10, min_samples_leaf=1, min_samples_split=10,
n_estimators=100; total time=   8.6s
[CV] END max_depth=10, min_samples_leaf=1, min_samples_split=10,
n_estimators=300; total time=  25.8s
[CV] END max_depth=10, min_samples_leaf=2, min_samples_split=5,
n_estimators=200; total time=  18.3s
[CV] END max_depth=10, min_samples_leaf=2, min_samples_split=10,
n_estimators=200; total time=  17.1s
[CV] END max_depth=10, min_samples_leaf=4, min_samples_split=2,
n_estimators=300; total time=  27.2s
[CV] END max_depth=10, min_samples_leaf=4, min_samples_split=10,
n_estimators=100; total time=   8.6s
[CV] END max_depth=20, min_samples_leaf=1, min_samples_split=2,
n_estimators=100; total time=  13.6s
[CV] END max_depth=20, min_samples_leaf=1, min_samples_split=2,
n_estimators=300; total time=  37.9s
[CV] END max_depth=20, min_samples_leaf=1, min_samples_split=10,
n_estimators=200; total time=  24.4s
[CV] END max_depth=20, min_samples_leaf=2, min_samples_split=2,
n_estimators=300; total time=  38.1s
[CV] END max_depth=20, min_samples_leaf=2, min_samples_split=10,
n_estimators=100; total time=  12.6s
[CV] END max_depth=20, min_samples_leaf=2, min_samples_split=10,
n_estimators=300; total time=  36.9s
[CV] END max_depth=20, min_samples_leaf=4, min_samples_split=5,
```

```
n_estimators=200; total time=  24.1s
[CV] END max_depth=20, min_samples_leaf=4, min_samples_split=10,
n_estimators=300; total time=  32.0s
[CV] END learning_rate=0.01, max_depth=3, n_estimators=200, subsample=1.0; total
time= 1.5min
[CV] END learning_rate=0.01, max_depth=5, n_estimators=100, subsample=0.8; total
time=  49.4s
[CV] END learning_rate=0.01, max_depth=5, n_estimators=100, subsample=1.0; total
time=  57.6s
[CV] END learning_rate=0.01, max_depth=5, n_estimators=300, subsample=1.0; total
time= 3.2min
[CV] END learning_rate=0.01, max_depth=7, n_estimators=300, subsample=0.8; total
time= 3.3min
[CV] END learning_rate=0.1, max_depth=3, n_estimators=200, subsample=1.0; total
time= 1.3min
[CV] END learning_rate=0.1, max_depth=5, n_estimators=100, subsample=0.8; total
time=  50.8s
[CV] END learning_rate=0.1, max_depth=5, n_estimators=200, subsample=0.8; total
time= 1.7min
[CV] END learning_rate=0.1, max_depth=5, n_estimators=300, subsample=1.0; total
time= 3.3min
[CV] END learning_rate=0.1, max_depth=7, n_estimators=300, subsample=0.8; total
time= 3.5min
[CV] END learning_rate=0.2, max_depth=3, n_estimators=300, subsample=0.8; total
time= 1.8min
[CV] END learning_rate=0.2, max_depth=5, n_estimators=100, subsample=0.8; total
time=  52.5s
[CV] END learning_rate=0.2, max_depth=5, n_estimators=200, subsample=1.0; total
time= 2.4min
[CV] END learning_rate=0.2, max_depth=7, n_estimators=100, subsample=0.8; total
time= 1.2min
[CV] END learning_rate=0.2, max_depth=7, n_estimators=200, subsample=1.0; total
time= 3.1min
[CV] END max_depth=None, min_samples_leaf=1, min_samples_split=10,
n_estimators=200; total time=  27.5s
[CV] END max_depth=None, min_samples_leaf=2, min_samples_split=2,
n_estimators=200; total time=  27.6s
[CV] END max_depth=None, min_samples_leaf=2, min_samples_split=5,
n_estimators=300; total time=  41.0s
[CV] END max_depth=None, min_samples_leaf=4, min_samples_split=2,
n_estimators=200; total time=  26.6s
[CV] END max_depth=None, min_samples_leaf=4, min_samples_split=5,
n_estimators=200; total time=  26.4s
[CV] END max_depth=None, min_samples_leaf=4, min_samples_split=10,
n_estimators=300; total time=  40.4s
[CV] END max_depth=10, min_samples_leaf=1, min_samples_split=5,
n_estimators=200; total time=  17.0s
[CV] END max_depth=10, min_samples_leaf=1, min_samples_split=10,
```

```
n_estimators=200; total time=  17.0s
[CV] END max_depth=10, min_samples_leaf=2, min_samples_split=2,
n_estimators=300; total time=  27.5s
[CV] END max_depth=10, min_samples_leaf=2, min_samples_split=10,
n_estimators=100; total time=   8.7s
[CV] END max_depth=10, min_samples_leaf=4, min_samples_split=2,
n_estimators=100; total time=   9.1s
[CV] END max_depth=10, min_samples_leaf=4, min_samples_split=2,
n_estimators=200; total time=  17.0s
[CV] END max_depth=10, min_samples_leaf=4, min_samples_split=5,
n_estimators=200; total time=  17.3s
[CV] END max_depth=10, min_samples_leaf=4, min_samples_split=10,
n_estimators=300; total time=  25.1s
[CV] END max_depth=20, min_samples_leaf=1, min_samples_split=5,
n_estimators=100; total time=  12.3s
[CV] END max_depth=20, min_samples_leaf=1, min_samples_split=5,
n_estimators=300; total time=  38.5s
[CV] END max_depth=20, min_samples_leaf=2, min_samples_split=2,
n_estimators=100; total time=  12.3s
[CV] END max_depth=20, min_samples_leaf=2, min_samples_split=5,
n_estimators=100; total time=  12.6s
[CV] END max_depth=20, min_samples_leaf=2, min_samples_split=5,
n_estimators=200; total time=  25.5s
[CV] END max_depth=20, min_samples_leaf=2, min_samples_split=10,
n_estimators=300; total time=  36.9s
[CV] END max_depth=20, min_samples_leaf=4, min_samples_split=5,
n_estimators=100; total time=  12.0s
[CV] END max_depth=20, min_samples_leaf=4, min_samples_split=5,
n_estimators=300; total time=  34.5s
[CV] END learning_rate=0.01, max_depth=3, n_estimators=100, subsample=1.0; total
time=  39.4s
[CV] END learning_rate=0.01, max_depth=3, n_estimators=300, subsample=0.8; total
time= 1.6min
[CV] END learning_rate=0.01, max_depth=5, n_estimators=200, subsample=0.8; total
time= 1.7min
[CV] END learning_rate=0.01, max_depth=7, n_estimators=100, subsample=0.8; total
time= 1.1min
[CV] END learning_rate=0.01, max_depth=7, n_estimators=100, subsample=1.0; total
time= 1.2min
[CV] END learning_rate=0.01, max_depth=7, n_estimators=200, subsample=1.0; total
time= 2.4min
[CV] END learning_rate=0.1, max_depth=3, n_estimators=100, subsample=1.0; total
time=  39.3s
[CV] END learning_rate=0.1, max_depth=3, n_estimators=200, subsample=0.8; total
time= 1.1min
[CV] END learning_rate=0.1, max_depth=3, n_estimators=300, subsample=1.0; total
time= 2.2min
[CV] END learning_rate=0.1, max_depth=5, n_estimators=200, subsample=1.0; total
```

```
time= 2.2min
[CV] END learning_rate=0.1, max_depth=7, n_estimators=100, subsample=0.8; total
time= 1.2min
[CV] END learning_rate=0.1, max_depth=7, n_estimators=200, subsample=1.0; total
time= 3.1min
[CV] END learning_rate=0.2, max_depth=3, n_estimators=100, subsample=0.8; total
time=  32.9s
[CV] END learning_rate=0.2, max_depth=3, n_estimators=200, subsample=0.8; total
time= 1.2min
[CV] END learning_rate=0.2, max_depth=3, n_estimators=300, subsample=0.8; total
time= 1.6min
[CV] END learning_rate=0.2, max_depth=5, n_estimators=100, subsample=1.0; total
time= 1.2min
[CV] END learning_rate=0.2, max_depth=5, n_estimators=200, subsample=1.0; total
time= 2.2min
[CV] END learning_rate=0.2, max_depth=7, n_estimators=100, subsample=1.0; total
time= 1.6min
[CV] END learning_rate=0.2, max_depth=7, n_estimators=200, subsample=1.0; total
time= 2.8min
[CV] END max_depth=None, min_samples_leaf=1, min_samples_split=2,
n_estimators=300; total time=  43.9s
[CV] END max_depth=None, min_samples_leaf=1, min_samples_split=10,
n_estimators=300; total time=  41.5s
[CV] END max_depth=None, min_samples_leaf=2, min_samples_split=5,
n_estimators=100; total time=  13.9s
[CV] END max_depth=None, min_samples_leaf=2, min_samples_split=5,
n_estimators=300; total time=  41.1s
[CV] END max_depth=None, min_samples_leaf=4, min_samples_split=2,
n_estimators=200; total time=  26.3s
[CV] END max_depth=None, min_samples_leaf=4, min_samples_split=5,
n_estimators=200; total time=  25.9s
[CV] END max_depth=None, min_samples_leaf=4, min_samples_split=10,
n_estimators=300; total time=  39.6s
[CV] END max_depth=10, min_samples_leaf=1, min_samples_split=5,
n_estimators=100; total time=   8.4s
[CV] END max_depth=10, min_samples_leaf=1, min_samples_split=5,
n_estimators=300; total time=  25.4s
[CV] END max_depth=10, min_samples_leaf=2, min_samples_split=2,
n_estimators=200; total time=  17.0s
[CV] END max_depth=10, min_samples_leaf=2, min_samples_split=5,
n_estimators=300; total time=  25.3s
[CV] END max_depth=10, min_samples_leaf=4, min_samples_split=2,
n_estimators=100; total time=   8.6s
[CV] END max_depth=10, min_samples_leaf=4, min_samples_split=5,
n_estimators=100; total time=   9.1s
[CV] END max_depth=10, min_samples_leaf=4, min_samples_split=5,
n_estimators=200; total time=  18.3s
[CV] END max_depth=10, min_samples_leaf=4, min_samples_split=10,
```

```
n_estimators=300; total time=  26.9s
[CV] END max_depth=20, min_samples_leaf=1, min_samples_split=5,
n_estimators=100; total time=  12.5s
[CV] END max_depth=20, min_samples_leaf=1, min_samples_split=5,
n_estimators=200; total time=  25.0s
[CV] END max_depth=20, min_samples_leaf=1, min_samples_split=10,
n_estimators=300; total time=  36.7s
[CV] END max_depth=20, min_samples_leaf=2, min_samples_split=5,
n_estimators=100; total time=  12.3s
[CV] END max_depth=20, min_samples_leaf=2, min_samples_split=5,
n_estimators=300; total time=  36.9s
[CV] END max_depth=20, min_samples_leaf=4, min_samples_split=2,
n_estimators=200; total time=  24.1s
[CV] END max_depth=20, min_samples_leaf=4, min_samples_split=5,
n_estimators=200; total time=  23.8s
[CV] END max_depth=20, min_samples_leaf=4, min_samples_split=10,
n_estimators=300; total time=  30.0s
[CV] END learning_rate=0.01, max_depth=3, n_estimators=100, subsample=1.0; total
time=  39.5s
[CV] END learning_rate=0.01, max_depth=3, n_estimators=300, subsample=0.8; total
time= 1.6min
[CV] END learning_rate=0.01, max_depth=5, n_estimators=200, subsample=0.8; total
time= 1.6min
[CV] END learning_rate=0.01, max_depth=5, n_estimators=300, subsample=1.0; total
time= 3.0min
[CV] END learning_rate=0.01, max_depth=7, n_estimators=300, subsample=0.8; total
time= 3.2min
[CV] END learning_rate=0.1, max_depth=3, n_estimators=300, subsample=0.8; total
time= 1.8min
[CV] END learning_rate=0.1, max_depth=5, n_estimators=100, subsample=1.0; total
time= 1.1min
[CV] END learning_rate=0.1, max_depth=5, n_estimators=300, subsample=0.8; total
time= 2.6min
[CV] END learning_rate=0.1, max_depth=7, n_estimators=100, subsample=1.0; total
time= 1.4min
[CV] END learning_rate=0.1, max_depth=7, n_estimators=300, subsample=0.8; total
time= 3.5min
[CV] END learning_rate=0.2, max_depth=3, n_estimators=200, subsample=1.0; total
time= 1.4min
[CV] END learning_rate=0.2, max_depth=5, n_estimators=100, subsample=0.8; total
time=  51.8s
[CV] END learning_rate=0.2, max_depth=5, n_estimators=200, subsample=0.8; total
time= 1.9min
[CV] END learning_rate=0.2, max_depth=7, n_estimators=100, subsample=0.8; total
time= 1.2min
[CV] END learning_rate=0.2, max_depth=7, n_estimators=100, subsample=1.0; total
time= 1.5min
[CV] END learning_rate=0.2, max_depth=7, n_estimators=200, subsample=1.0; total
```

```
time= 2.8min
[CV] END max_depth=None, min_samples_leaf=1, min_samples_split=5,
n_estimators=100; total time=  14.1s
[CV] END max_depth=None, min_samples_leaf=1, min_samples_split=5,
n_estimators=200; total time=  28.0s
[CV] END max_depth=None, min_samples_leaf=1, min_samples_split=10,
n_estimators=100; total time=  13.8s
[CV] END max_depth=None, min_samples_leaf=1, min_samples_split=10,
n_estimators=300; total time=  41.4s
[CV] END max_depth=None, min_samples_leaf=2, min_samples_split=5,
n_estimators=100; total time=  13.8s
[CV] END max_depth=None, min_samples_leaf=2, min_samples_split=10,
n_estimators=100; total time=  13.6s
[CV] END max_depth=None, min_samples_leaf=2, min_samples_split=10,
n_estimators=200; total time=  27.3s
[CV] END max_depth=None, min_samples_leaf=4, min_samples_split=2,
n_estimators=200; total time=  25.9s
[CV] END max_depth=None, min_samples_leaf=4, min_samples_split=5,
n_estimators=300; total time=  39.0s
[CV] END max_depth=10, min_samples_leaf=1, min_samples_split=2,
n_estimators=200; total time=  16.8s
[CV] END max_depth=10, min_samples_leaf=1, min_samples_split=5,
n_estimators=200; total time=  17.0s
[CV] END max_depth=10, min_samples_leaf=1, min_samples_split=10,
n_estimators=200; total time=  17.2s
[CV] END max_depth=10, min_samples_leaf=2, min_samples_split=2,
n_estimators=300; total time=  25.5s
[CV] END max_depth=10, min_samples_leaf=2, min_samples_split=10,
n_estimators=100; total time=   8.4s
[CV] END max_depth=10, min_samples_leaf=2, min_samples_split=10,
n_estimators=300; total time=  25.5s
[CV] END max_depth=10, min_samples_leaf=4, min_samples_split=5,
n_estimators=200; total time=  16.8s
[CV] END max_depth=10, min_samples_leaf=4, min_samples_split=10,
n_estimators=200; total time=  17.1s
[CV] END max_depth=20, min_samples_leaf=1, min_samples_split=2,
n_estimators=300; total time=  39.0s
[CV] END max_depth=20, min_samples_leaf=1, min_samples_split=10,
n_estimators=200; total time=  25.4s
[CV] END max_depth=20, min_samples_leaf=2, min_samples_split=2,
n_estimators=200; total time=  24.6s
[CV] END max_depth=20, min_samples_leaf=2, min_samples_split=5,
n_estimators=300; total time=  36.9s
[CV] END max_depth=20, min_samples_leaf=4, min_samples_split=2,
n_estimators=100; total time=  11.9s
[CV] END max_depth=20, min_samples_leaf=4, min_samples_split=2,
n_estimators=300; total time=  35.7s
[CV] END max_depth=20, min_samples_leaf=4, min_samples_split=10,
```

```
n_estimators=200; total time=  23.3s
[CV] END learning_rate=0.01, max_depth=3, n_estimators=200, subsample=0.8; total
time= 1.1min
[CV] END learning_rate=0.01, max_depth=3, n_estimators=300, subsample=1.0; total
time= 2.0min
[CV] END learning_rate=0.01, max_depth=5, n_estimators=300, subsample=0.8; total
time= 2.4min
[CV] END learning_rate=0.01, max_depth=7, n_estimators=100, subsample=1.0; total
time= 1.2min
[CV] END learning_rate=0.01, max_depth=7, n_estimators=300, subsample=0.8; total
time= 3.1min
[CV] END learning_rate=0.1, max_depth=3, n_estimators=200, subsample=1.0; total
time= 1.3min
[CV] END learning_rate=0.1, max_depth=5, n_estimators=100, subsample=0.8; total
time=  55.0s
[CV] END learning_rate=0.1, max_depth=5, n_estimators=200, subsample=0.8; total
time= 1.7min
[CV] END learning_rate=0.1, max_depth=7, n_estimators=100, subsample=0.8; total
time= 1.2min
[CV] END learning_rate=0.1, max_depth=7, n_estimators=100, subsample=1.0; total
time= 1.4min
[CV] END learning_rate=0.1, max_depth=7, n_estimators=200, subsample=1.0; total
time= 2.9min
[CV] END learning_rate=0.2, max_depth=3, n_estimators=100, subsample=1.0; total
time=  40.9s
[CV] END learning_rate=0.2, max_depth=3, n_estimators=200, subsample=1.0; total
time= 1.5min
[CV] END learning_rate=0.2, max_depth=3, n_estimators=300, subsample=1.0; total
time= 2.1min
[CV] END learning_rate=0.2, max_depth=5, n_estimators=300, subsample=0.8; total
time= 2.6min
[CV] END learning_rate=0.2, max_depth=7, n_estimators=100, subsample=1.0; total
time= 1.4min
[CV] END learning_rate=0.2, max_depth=7, n_estimators=300, subsample=0.8; total
time= 3.2min
[CV] END max_depth=None, min_samples_leaf=1, min_samples_split=5,
n_estimators=300; total time=  42.4s
[CV] END max_depth=None, min_samples_leaf=2, min_samples_split=2,
n_estimators=300; total time=  41.4s
[CV] END max_depth=None, min_samples_leaf=2, min_samples_split=10,
n_estimators=100; total time=  13.5s
[CV] END max_depth=None, min_samples_leaf=4, min_samples_split=2,
n_estimators=100; total time=  13.2s
[CV] END max_depth=None, min_samples_leaf=4, min_samples_split=2,
n_estimators=100; total time=  13.3s
[CV] END max_depth=None, min_samples_leaf=4, min_samples_split=5,
n_estimators=100; total time=  13.1s
[CV] END max_depth=None, min_samples_leaf=4, min_samples_split=5,
```

```
n_estimators=100; total time=  13.0s
[CV] END max_depth=None, min_samples_leaf=4, min_samples_split=5,
n_estimators=300; total time=  39.1s
[CV] END max_depth=10, min_samples_leaf=1, min_samples_split=2,
n_estimators=200; total time=  16.9s
[CV] END max_depth=10, min_samples_leaf=1, min_samples_split=5,
n_estimators=300; total time=  27.1s
[CV] END max_depth=10, min_samples_leaf=2, min_samples_split=2,
n_estimators=100; total time=   8.8s
[CV] END max_depth=10, min_samples_leaf=2, min_samples_split=5,
n_estimators=100; total time=   9.2s
[CV] END max_depth=10, min_samples_leaf=2, min_samples_split=5,
n_estimators=200; total time=  17.1s
[CV] END max_depth=10, min_samples_leaf=2, min_samples_split=10,
n_estimators=300; total time=  27.3s
[CV] END max_depth=10, min_samples_leaf=4, min_samples_split=5,
n_estimators=100; total time=   8.6s
[CV] END max_depth=10, min_samples_leaf=4, min_samples_split=10,
n_estimators=100; total time=   9.2s
[CV] END max_depth=10, min_samples_leaf=4, min_samples_split=10,
n_estimators=200; total time=  16.9s
[CV] END max_depth=20, min_samples_leaf=1, min_samples_split=2,
n_estimators=200; total time=  25.3s
[CV] END max_depth=20, min_samples_leaf=1, min_samples_split=5,
n_estimators=300; total time=  37.4s
[CV] END max_depth=20, min_samples_leaf=2, min_samples_split=2,
n_estimators=200; total time=  25.4s
[CV] END max_depth=20, min_samples_leaf=2, min_samples_split=5,
n_estimators=200; total time=  24.5s
[CV] END max_depth=20, min_samples_leaf=2, min_samples_split=10,
n_estimators=300; total time=  38.0s
[CV] END max_depth=20, min_samples_leaf=4, min_samples_split=5,
n_estimators=100; total time=  11.9s
[CV] END max_depth=20, min_samples_leaf=4, min_samples_split=10,
n_estimators=100; total time=  12.6s
[CV] END max_depth=20, min_samples_leaf=4, min_samples_split=10,
n_estimators=200; total time=  22.6s
[CV] END learning_rate=0.01, max_depth=3, n_estimators=100, subsample=1.0; total
time=  42.9s
[CV] END learning_rate=0.01, max_depth=3, n_estimators=300, subsample=1.0; total
time= 2.2min
[CV] END learning_rate=0.01, max_depth=5, n_estimators=200, subsample=1.0; total
time= 1.9min
[CV] END learning_rate=0.01, max_depth=7, n_estimators=100, subsample=0.8; total
time= 1.0min
[CV] END learning_rate=0.01, max_depth=7, n_estimators=200, subsample=1.0; total
time= 2.5min
[CV] END learning_rate=0.1, max_depth=3, n_estimators=100, subsample=0.8; total
```

```
time=  31.7s
[CV] END learning_rate=0.1, max_depth=3, n_estimators=100, subsample=1.0; total
time=  39.5s
[CV] END learning_rate=0.1, max_depth=3, n_estimators=200, subsample=1.0; total
time= 1.5min
[CV] END learning_rate=0.1, max_depth=3, n_estimators=300, subsample=1.0; total
time= 2.0min
[CV] END learning_rate=0.1, max_depth=5, n_estimators=300, subsample=1.0; total
time= 3.5min
[CV] END learning_rate=0.1, max_depth=7, n_estimators=300, subsample=0.8; total
time= 3.8min
[CV] END learning_rate=0.2, max_depth=3, n_estimators=200, subsample=1.0; total
time= 1.4min
[CV] END learning_rate=0.2, max_depth=3, n_estimators=300, subsample=1.0; total
time= 2.1min
[CV] END learning_rate=0.2, max_depth=5, n_estimators=300, subsample=1.0; total
time= 3.5min
[CV] END learning_rate=0.2, max_depth=7, n_estimators=300, subsample=0.8; total
time= 3.5min
[CV] END max_depth=None, min_samples_leaf=1, min_samples_split=5,
n_estimators=200; total time=  28.5s
[CV] END max_depth=None, min_samples_leaf=1, min_samples_split=10,
n_estimators=200; total time=  27.6s
[CV] END max_depth=None, min_samples_leaf=2, min_samples_split=2,
n_estimators=200; total time=  27.5s
[CV] END max_depth=None, min_samples_leaf=2, min_samples_split=5,
n_estimators=200; total time=  27.5s
[CV] END max_depth=None, min_samples_leaf=2, min_samples_split=10,
n_estimators=300; total time=  40.8s
[CV] END max_depth=None, min_samples_leaf=4, min_samples_split=5,
n_estimators=200; total time=  25.9s
[CV] END max_depth=None, min_samples_leaf=4, min_samples_split=10,
n_estimators=200; total time=  26.2s
[CV] END max_depth=10, min_samples_leaf=1, min_samples_split=2,
n_estimators=300; total time=  25.2s
[CV] END max_depth=10, min_samples_leaf=1, min_samples_split=10,
n_estimators=100; total time=   8.5s
[CV] END max_depth=10, min_samples_leaf=1, min_samples_split=10,
n_estimators=300; total time=  25.5s
[CV] END max_depth=10, min_samples_leaf=2, min_samples_split=5,
n_estimators=100; total time=   8.5s
[CV] END max_depth=10, min_samples_leaf=2, min_samples_split=5,
n_estimators=300; total time=  25.5s
[CV] END max_depth=10, min_samples_leaf=4, min_samples_split=2,
n_estimators=200; total time=  18.3s
[CV] END max_depth=10, min_samples_leaf=4, min_samples_split=5,
n_estimators=300; total time=  27.1s
[CV] END max_depth=20, min_samples_leaf=1, min_samples_split=2,
```

```
n_estimators=100; total time=  12.8s
[CV] END max_depth=20, min_samples_leaf=1, min_samples_split=5,
n_estimators=100; total time=  12.6s
[CV] END max_depth=20, min_samples_leaf=1, min_samples_split=5,
n_estimators=200; total time=  25.2s
[CV] END max_depth=20, min_samples_leaf=1, min_samples_split=10,
n_estimators=300; total time=  37.8s
[CV] END max_depth=20, min_samples_leaf=2, min_samples_split=5,
n_estimators=100; total time=  12.2s
[CV] END max_depth=20, min_samples_leaf=2, min_samples_split=10,
n_estimators=100; total time=  12.6s
[CV] END max_depth=20, min_samples_leaf=2, min_samples_split=10,
n_estimators=200; total time=  24.5s
[CV] END max_depth=20, min_samples_leaf=4, min_samples_split=2,
n_estimators=200; total time=  23.9s
[CV] END max_depth=20, min_samples_leaf=4, min_samples_split=5,
n_estimators=300; total time=  36.2s
[CV] END learning_rate=0.01, max_depth=3, n_estimators=200, subsample=0.8; total
time= 1.1min
[CV] END learning_rate=0.01, max_depth=3, n_estimators=300, subsample=1.0; total
time= 2.0min
[CV] END learning_rate=0.01, max_depth=5, n_estimators=300, subsample=0.8; total
time= 2.5min
[CV] END learning_rate=0.01, max_depth=7, n_estimators=200, subsample=0.8; total
time= 2.1min
[CV] END learning_rate=0.01, max_depth=7, n_estimators=300, subsample=1.0; total
time= 3.7min
[CV] END learning_rate=0.1, max_depth=5, n_estimators=100, subsample=1.0; total
time= 1.1min
[CV] END learning_rate=0.1, max_depth=5, n_estimators=200, subsample=1.0; total
time= 2.2min
[CV] END learning_rate=0.1, max_depth=7, n_estimators=100, subsample=1.0; total
time= 1.5min
[CV] END learning_rate=0.1, max_depth=7, n_estimators=200, subsample=1.0; total
time= 2.9min
[CV] END learning_rate=0.2, max_depth=3, n_estimators=100, subsample=1.0; total
time=  44.3s
[CV] END learning_rate=0.2, max_depth=3, n_estimators=200, subsample=0.8; total
time= 1.1min
[CV] END learning_rate=0.2, max_depth=3, n_estimators=300, subsample=0.8; total
time= 1.6min
[CV] END learning_rate=0.2, max_depth=5, n_estimators=200, subsample=0.8; total
time= 1.8min
[CV] END learning_rate=0.2, max_depth=5, n_estimators=300, subsample=1.0; total
time= 3.2min
[CV] END learning_rate=0.2, max_depth=7, n_estimators=300, subsample=0.8; total
time= 3.2min
[CV] END max_depth=None, min_samples_leaf=1, min_samples_split=5,
```

```
n_estimators=300; total time=  42.4s
[CV] END max_depth=None, min_samples_leaf=2, min_samples_split=2,
n_estimators=300; total time=  40.9s
[CV] END max_depth=None, min_samples_leaf=2, min_samples_split=10,
n_estimators=100; total time=  13.6s
[CV] END max_depth=None, min_samples_leaf=2, min_samples_split=10,
n_estimators=300; total time=  40.4s
[CV] END max_depth=None, min_samples_leaf=4, min_samples_split=5,
n_estimators=300; total time=  39.9s
[CV] END max_depth=10, min_samples_leaf=1, min_samples_split=2,
n_estimators=100; total time=   8.3s
[CV] END max_depth=10, min_samples_leaf=1, min_samples_split=2,
n_estimators=200; total time=  18.0s
[CV] END max_depth=10, min_samples_leaf=1, min_samples_split=5,
n_estimators=200; total time=  18.2s
[CV] END max_depth=10, min_samples_leaf=1, min_samples_split=10,
n_estimators=300; total time=  27.5s
[CV] END max_depth=10, min_samples_leaf=2, min_samples_split=5,
n_estimators=100; total time=   8.5s
[CV] END max_depth=10, min_samples_leaf=2, min_samples_split=10,
n_estimators=100; total time=   9.1s
[CV] END max_depth=10, min_samples_leaf=2, min_samples_split=10,
n_estimators=200; total time=  18.2s
[CV] END max_depth=10, min_samples_leaf=4, min_samples_split=2,
n_estimators=300; total time=  25.6s
[CV] END max_depth=10, min_samples_leaf=4, min_samples_split=10,
n_estimators=100; total time=   8.5s
[CV] END max_depth=10, min_samples_leaf=4, min_samples_split=10,
n_estimators=300; total time=  25.1s
[CV] END max_depth=20, min_samples_leaf=1, min_samples_split=5,
n_estimators=200; total time=  25.8s
[CV] END max_depth=20, min_samples_leaf=1, min_samples_split=10,
n_estimators=100; total time=  12.5s
[CV] END max_depth=20, min_samples_leaf=2, min_samples_split=2,
n_estimators=100; total time=  12.6s
[CV] END max_depth=20, min_samples_leaf=2, min_samples_split=2,
n_estimators=200; total time=  24.7s
[CV] END max_depth=20, min_samples_leaf=2, min_samples_split=5,
n_estimators=300; total time=  38.1s
[CV] END max_depth=20, min_samples_leaf=4, min_samples_split=2,
n_estimators=100; total time=  12.0s
[CV] END max_depth=20, min_samples_leaf=4, min_samples_split=5,
n_estimators=100; total time=  12.4s
[CV] END max_depth=20, min_samples_leaf=4, min_samples_split=5,
n_estimators=200; total time=  25.0s
[CV] END max_depth=20, min_samples_leaf=4, min_samples_split=10,
n_estimators=200; total time=  22.0s
[CV] END learning_rate=0.01, max_depth=3, n_estimators=100, subsample=0.8; total
```

```
time=  32.6s
[CV] END learning_rate=0.01, max_depth=3, n_estimators=200, subsample=1.0; total
time= 1.3min
[CV] END learning_rate=0.01, max_depth=5, n_estimators=100, subsample=0.8; total
time=  49.4s
[CV] END learning_rate=0.01, max_depth=5, n_estimators=200, subsample=1.0; total
time= 2.1min
[CV] END learning_rate=0.01, max_depth=7, n_estimators=100, subsample=0.8; total
time= 1.0min
[CV] END learning_rate=0.01, max_depth=7, n_estimators=200, subsample=0.8; total
time= 2.1min
[CV] END learning_rate=0.1, max_depth=3, n_estimators=100, subsample=0.8; total
time=  34.0s
[CV] END learning_rate=0.1, max_depth=3, n_estimators=100, subsample=1.0; total
time=  42.9s
[CV] END learning_rate=0.1, max_depth=3, n_estimators=200, subsample=0.8; total
time= 1.1min
[CV] END learning_rate=0.1, max_depth=3, n_estimators=300, subsample=0.8; total
time= 1.6min
[CV] END learning_rate=0.1, max_depth=5, n_estimators=200, subsample=0.8; total
time= 1.9min
[CV] END learning_rate=0.1, max_depth=5, n_estimators=300, subsample=1.0; total
time= 3.3min
[CV] END learning_rate=0.1, max_depth=7, n_estimators=300, subsample=1.0; total
time= 4.7min
[CV] END learning_rate=0.2, max_depth=5, n_estimators=100, subsample=0.8; total
time=  55.9s
[CV] END learning_rate=0.2, max_depth=5, n_estimators=200, subsample=0.8; total
time= 1.8min
[CV] END learning_rate=0.2, max_depth=5, n_estimators=300, subsample=1.0; total
time= 3.3min
[CV] END learning_rate=0.2, max_depth=7, n_estimators=300, subsample=1.0; total
time= 4.1min
[CV] END max_depth=None, min_samples_leaf=2, min_samples_split=10,
n_estimators=200; total time=  27.0s
[CV] END max_depth=None, min_samples_leaf=4, min_samples_split=2,
n_estimators=300; total time=  38.5s
[CV] END max_depth=None, min_samples_leaf=4, min_samples_split=10,
n_estimators=100; total time=  13.0s
[CV] END max_depth=10, min_samples_leaf=1, min_samples_split=2,
n_estimators=100; total time=   8.7s
[CV] END max_depth=10, min_samples_leaf=1, min_samples_split=2,
n_estimators=100; total time=   8.1s
[CV] END max_depth=10, min_samples_leaf=1, min_samples_split=5,
n_estimators=100; total time=   8.9s
[CV] END max_depth=10, min_samples_leaf=1, min_samples_split=5,
n_estimators=100; total time=   8.3s
[CV] END max_depth=10, min_samples_leaf=1, min_samples_split=5,
```

```
n_estimators=300; total time=  25.4s
[CV] END max_depth=10, min_samples_leaf=2, min_samples_split=2,
n_estimators=200; total time=  17.2s
[CV] END max_depth=10, min_samples_leaf=2, min_samples_split=5,
n_estimators=200; total time=  16.9s
[CV] END max_depth=10, min_samples_leaf=2, min_samples_split=10,
n_estimators=300; total time=  25.7s
[CV] END max_depth=10, min_samples_leaf=4, min_samples_split=5,
n_estimators=100; total time=   8.6s
[CV] END max_depth=10, min_samples_leaf=4, min_samples_split=5,
n_estimators=300; total time=  25.5s
[CV] END max_depth=20, min_samples_leaf=1, min_samples_split=2,
n_estimators=200; total time=  25.9s
[CV] END max_depth=20, min_samples_leaf=1, min_samples_split=5,
n_estimators=300; total time=  37.4s
[CV] END max_depth=20, min_samples_leaf=2, min_samples_split=2,
n_estimators=100; total time=  12.2s
[CV] END max_depth=20, min_samples_leaf=2, min_samples_split=2,
n_estimators=300; total time=  36.7s
[CV] END max_depth=20, min_samples_leaf=2, min_samples_split=10,
n_estimators=200; total time=  25.5s
[CV] END max_depth=20, min_samples_leaf=4, min_samples_split=2,
n_estimators=300; total time=  37.1s
[CV] END max_depth=20, min_samples_leaf=4, min_samples_split=10,
n_estimators=100; total time=  12.1s
[CV] END learning_rate=0.01, max_depth=3, n_estimators=100, subsample=0.8; total
time=  35.7s
[CV] END learning_rate=0.01, max_depth=3, n_estimators=300, subsample=0.8; total
time= 1.8min
[CV] END learning_rate=0.01, max_depth=5, n_estimators=200, subsample=0.8; total
time= 1.6min
[CV] END learning_rate=0.01, max_depth=5, n_estimators=300, subsample=1.0; total
time= 3.0min
[CV] END learning_rate=0.01, max_depth=7, n_estimators=300, subsample=1.0; total
time= 4.1min
[CV] END learning_rate=0.1, max_depth=3, n_estimators=300, subsample=1.0; total
time= 2.0min
[CV] END learning_rate=0.1, max_depth=5, n_estimators=300, subsample=0.8; total
time= 2.6min
[CV] END learning_rate=0.1, max_depth=7, n_estimators=200, subsample=0.8; total
time= 2.3min
[CV] END learning_rate=0.1, max_depth=7, n_estimators=300, subsample=1.0; total
time= 4.4min
[CV] END learning_rate=0.2, max_depth=5, n_estimators=100, subsample=1.0; total
time= 1.1min
[CV] END learning_rate=0.2, max_depth=5, n_estimators=300, subsample=0.8; total
time= 2.8min
[CV] END learning_rate=0.2, max_depth=7, n_estimators=200, subsample=0.8; total
```

```
time= 2.3min
[CV] END learning_rate=0.2, max_depth=7, n_estimators=300, subsample=1.0; total
time= 3.6min
[CV] END max_depth=None, min_samples_leaf=1, min_samples_split=2,
n_estimators=200; total time=  29.0s
[CV] END max_depth=None, min_samples_leaf=1, min_samples_split=10,
n_estimators=100; total time=  13.9s
[CV] END max_depth=None, min_samples_leaf=1, min_samples_split=10,
n_estimators=200; total time=  27.6s
[CV] END max_depth=None, min_samples_leaf=2, min_samples_split=2,
n_estimators=100; total time=  13.7s
[CV] END max_depth=None, min_samples_leaf=2, min_samples_split=5,
n_estimators=100; total time=  13.7s
[CV] END max_depth=None, min_samples_leaf=2, min_samples_split=5,
n_estimators=200; total time=  27.4s
[CV] END max_depth=None, min_samples_leaf=2, min_samples_split=10,
n_estimators=300; total time=  41.1s
[CV] END max_depth=None, min_samples_leaf=4, min_samples_split=5,
n_estimators=100; total time=  13.0s
[CV] END max_depth=None, min_samples_leaf=4, min_samples_split=10,
n_estimators=100; total time=  13.1s
[CV] END max_depth=None, min_samples_leaf=4, min_samples_split=10,
n_estimators=200; total time=  26.2s
[CV] END max_depth=10, min_samples_leaf=1, min_samples_split=2,
n_estimators=300; total time=  26.7s
[CV] END max_depth=10, min_samples_leaf=1, min_samples_split=10,
n_estimators=200; total time=  18.1s
[CV] END max_depth=10, min_samples_leaf=2, min_samples_split=2,
n_estimators=200; total time=  18.2s
[CV] END max_depth=10, min_samples_leaf=2, min_samples_split=5,
n_estimators=300; total time=  26.9s
[CV] END max_depth=10, min_samples_leaf=4, min_samples_split=2,
n_estimators=100; total time=   8.5s
[CV] END max_depth=10, min_samples_leaf=4, min_samples_split=2,
n_estimators=300; total time=  25.5s
[CV] END max_depth=10, min_samples_leaf=4, min_samples_split=10,
n_estimators=200; total time=  18.1s
[CV] END max_depth=20, min_samples_leaf=1, min_samples_split=2,
n_estimators=200; total time=  25.1s
[CV] END max_depth=20, min_samples_leaf=1, min_samples_split=10,
n_estimators=100; total time=  12.8s
[CV] END max_depth=20, min_samples_leaf=1, min_samples_split=10,
n_estimators=100; total time=  12.2s
[CV] END max_depth=20, min_samples_leaf=1, min_samples_split=10,
n_estimators=300; total time=  36.9s
[CV] END max_depth=20, min_samples_leaf=2, min_samples_split=5,
n_estimators=200; total time=  24.5s
[CV] END max_depth=20, min_samples_leaf=2, min_samples_split=10,
```

```
n_estimators=200; total time=  24.5s
[CV] END max_depth=20, min_samples_leaf=4, min_samples_split=2,
n_estimators=300; total time=  35.8s
[CV] END max_depth=20, min_samples_leaf=4, min_samples_split=10,
n_estimators=100; total time=  11.8s
[CV] END max_depth=20, min_samples_leaf=4, min_samples_split=10,
n_estimators=300; total time=  29.4s
[CV] END learning_rate=0.01, max_depth=3, n_estimators=200, subsample=0.8; total
time= 1.2min
[CV] END learning_rate=0.01, max_depth=5, n_estimators=100, subsample=0.8; total
time=  51.7s
[CV] END learning_rate=0.01, max_depth=5, n_estimators=100, subsample=1.0; total
time=  57.8s
[CV] END learning_rate=0.01, max_depth=5, n_estimators=300, subsample=0.8; total
time= 2.6min
[CV] END learning_rate=0.01, max_depth=7, n_estimators=200, subsample=0.8; total
time= 2.0min
[CV] END learning_rate=0.01, max_depth=7, n_estimators=300, subsample=1.0; total
time= 3.7min
[CV] END learning_rate=0.1, max_depth=5, n_estimators=100, subsample=0.8; total
time=  50.9s
[CV] END learning_rate=0.1, max_depth=5, n_estimators=200, subsample=1.0; total
time= 2.3min
[CV] END learning_rate=0.1, max_depth=7, n_estimators=100, subsample=0.8; total
time= 1.2min
[CV] END learning_rate=0.1, max_depth=7, n_estimators=200, subsample=0.8; total
time= 2.3min
[CV] END learning_rate=0.1, max_depth=7, n_estimators=300, subsample=1.0; total
time= 4.5min
[CV] END learning_rate=0.2, max_depth=5, n_estimators=100, subsample=1.0; total
time= 1.1min
[CV] END learning_rate=0.2, max_depth=5, n_estimators=300, subsample=0.8; total
time= 2.6min
[CV] END learning_rate=0.2, max_depth=7, n_estimators=200, subsample=0.8; total
time= 2.5min
[CV] END learning_rate=0.2, max_depth=7, n_estimators=300, subsample=1.0; total
time= 3.6min
```

[8]:

```python
# ensemble of models

voting_clf = VotingClassifier(estimators=[
    ('xgb', best_xgb_classifier),
    ('rf', best_rf_classifier),
    ('gb', best_gb_classifier),
    ('lr', lr_classifier)
], voting='soft')
```

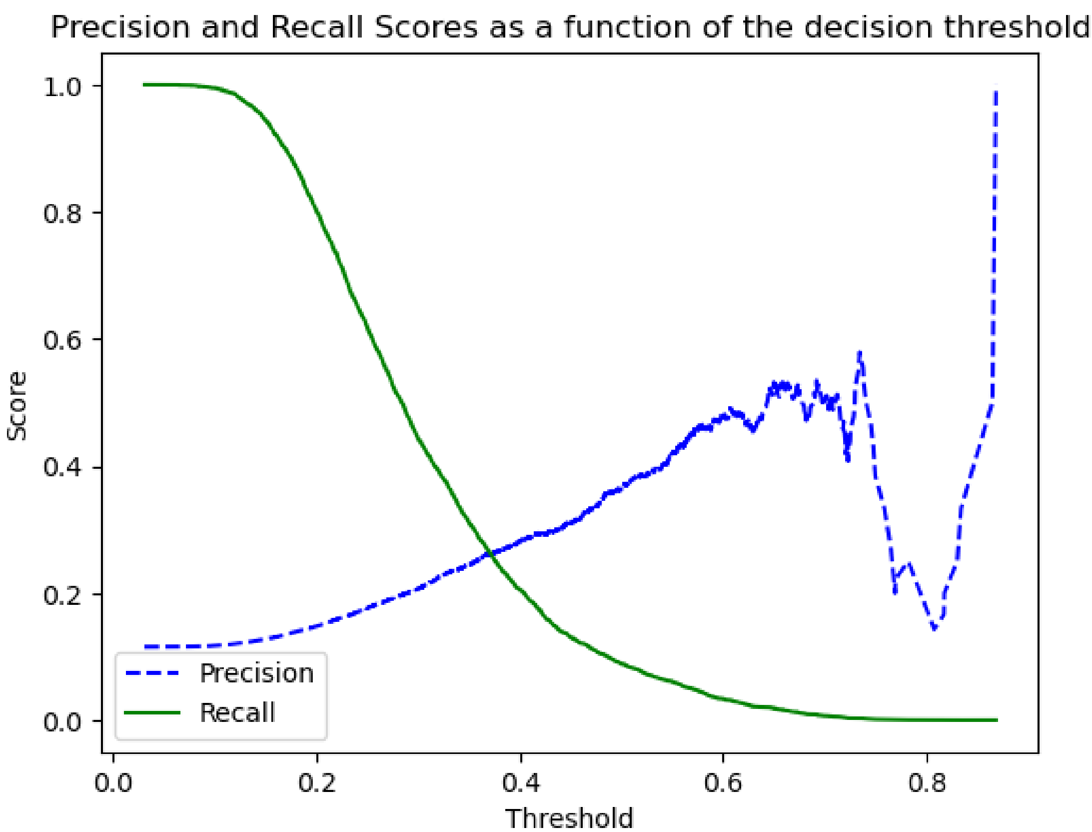
```python
# fit ensemble model

voting_clf.fit(X_train_resampled, y_train_resampled)

# predictions on the test set

y_pred = voting_clf.predict(X_test)

# evaluate the ensemble model

accuracy = accuracy_score(y_test, y_pred)
classification_rep = classification_report(y_test, y_pred)

print("Ensemble Model Accuracy:", accuracy)
print("Ensemble Model Classification Report:\n", classification_rep)
```

```
Ensemble Model Accuracy: 0.8769586954333456
Ensemble Model Classification Report:
               precision    recall  f1-score   support

           0       0.89      0.98      0.93     26362
           1       0.37      0.09      0.14      3441

    accuracy                           0.88     29803
   macro avg       0.63      0.53      0.54     29803
weighted avg       0.83      0.88      0.84     29803
```

[9]:

```python
# predictions on the test set

y_pred_prob = voting_clf.predict_proba(X_test)[:, 1]

# calc precision-recall curve

precision, recall, thresholds = precision_recall_curve(y_test, y_pred_prob)

# plot precision-recall curve

plt.plot(thresholds, precision[:-1], 'b--', label='Precision')
plt.plot(thresholds, recall[:-1], 'g-', label='Recall')
plt.xlabel('Threshold')
plt.ylabel('Score')
plt.title('Precision and Recall Scores as a function of the decision threshold')
plt.legend()
plt.show()

# threshold
```

```python
threshold = 0.3

# predictions

y_pred_adjusted = (y_pred_prob >= threshold).astype(int)

# evaluate the model threshold

adjusted_accuracy = accuracy_score(y_test, y_pred_adjusted)
adjusted_classification_rep = classification_report(y_test, y_pred_adjusted)

print("Adjusted Threshold:", threshold)
print("Adjusted Accuracy:", adjusted_accuracy)
print("Adjusted Classification Report:\n", adjusted_classification_rep)
```




```
Adjusted Threshold: 0.3
Adjusted Accuracy: 0.7386504714290507
Adjusted Classification Report:
               precision    recall  f1-score   support
```

```
           0       0.91      0.78      0.84     26362
           1       0.21      0.44      0.28      3441

    accuracy                           0.74     29803
   macro avg       0.56      0.61      0.56     29803
weighted avg       0.83      0.74      0.78     29803
```

[ ]:

```

```