МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ

РОССИЙСКОЙ ФЕДЕРАЦИИ

ФГАОУ ВО «СЕВЕРО-КАВКАЗСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

ИНСТИТУТ ИНФОРМАЦИОННЫХ

ТЕХНОЛОГИЙ И ТЕЛЕКОММУНИКАЦИЙ

КАФЕДРА ПРИКЛАДНОЙ ИНФОРМАТИКИ

**Лабораторная работа №5**

по дисциплине

«Искусственный интеллект и машинное обучение»

**Выполнил:**

Сердюков Никита Анатольевич

Студент 2 курса группы \_ПИН-б-о-22-1

Направления подготовки

09.03.03 Прикладная информатика

очной формы обучения

Ставрополь, 2023 г.

Тема: ЛОГИСТИЧЕСКАЯ РЕГРЕССИЯ.

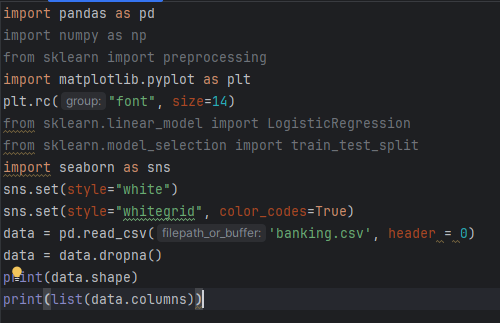
Цель работы: изучение принципов построения информационных систем с использованием линейных методов машинного обучения.

Выполнение работы:

Датасет: bank-additional-full

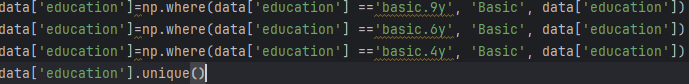
Описание датасета: Данные относятся к кампаниям прямого маркетинга (телефонным звонкам) португальского банковского учреждения. Цель классификации - предсказать, подпишется ли клиент на срочный депозит (переменная y)

1. Загрузка данных



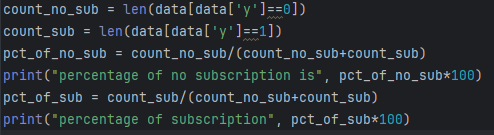


1. Сгруппируем "basic.4y", "basic.9y" и "basic.6y", назовём их "Basic":





1. Корреляция признаков





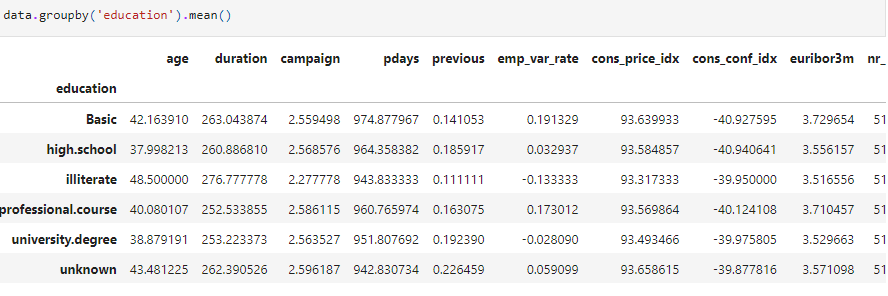
Получаем два значения: без вклада — 88,73458288821988 с вкладом — 11.265417111780131

1. Количество аниме выпущенное за различные года.



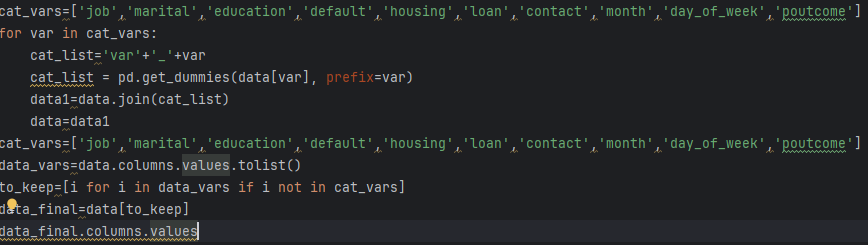
Из полученной таблицы можно сделать выводы:

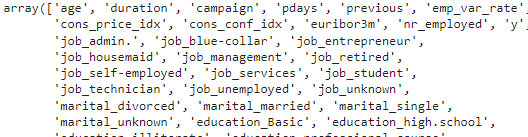
* Cредний возраст среди пенсионеров - 62 года
* Наименьшее среднее количество дней с момента последнего контакта с клиентом наблюдается у студентов и пенсионеров
* Зачастую соглашаются на предложения банка именно студенты и пенсионеры, в то время как обслуживающий персонал и "синии воротнички" - отказываются



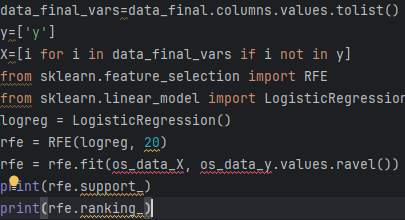
* Средний возраст клиентов, открывших вклад, выше среднего возраста тех, кто этого не сделал
* Количество дней с последнего контакта с клиентом по понятным причинам меньше у клиентов, открывших депозит
* Количество звонков за одну кампанию, меньше у клиентов, которые открыли депозит

1. Создадим три словаря и три матрицы TF-IDF



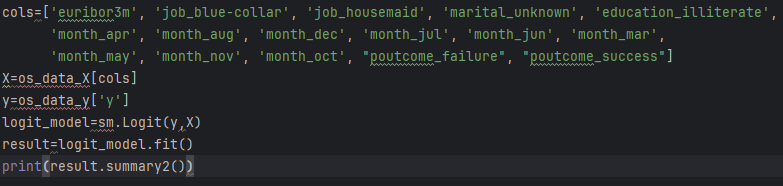


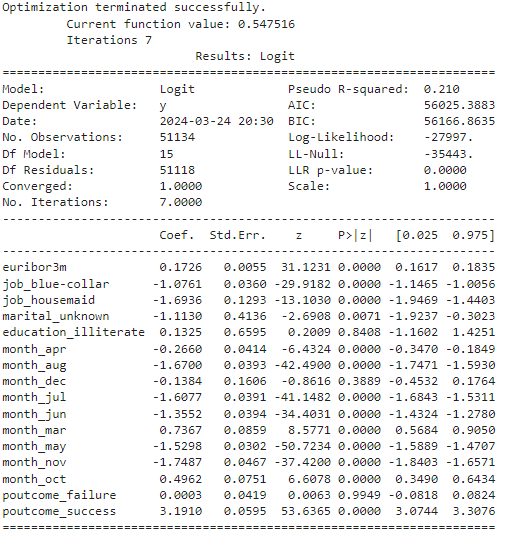
Этот блок кода использует метод RFE из библиотеки scikit-learn для отбора признаков с использованием логистической регрессии. RFE - это техника, которая рекурсивно удаляет наименее значимые признаки и строит модель до достижения указанного количества признаков. В данном случае модель логистической регрессии выбирает лучшие и худшие в плане производительности признаки, а затем отбрасывания признака, рекурсивно рассматривая всё меньшие и меньшие их множества



Метод помог выбрать следующие признаки: "euribor3m", "job\_blue-collar", "job\_housemaid", "marital\_unknown", "education\_illiterate", "default\_no", "default\_unknown", "contact\_cellular", "contact\_telephone", "month\_apr", "month\_aug", "month\_dec", "month\_jul", "month\_jun", "month\_mar", "month\_may", "month\_nov", "month\_oct", "poutcome\_failure", "poutcome\_success"

7.



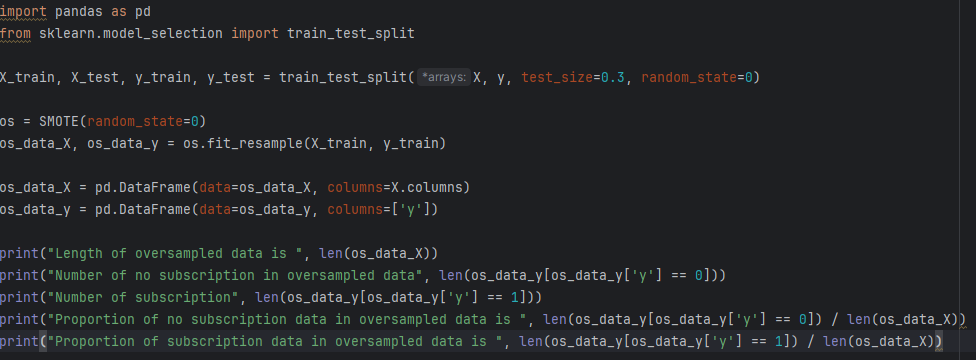


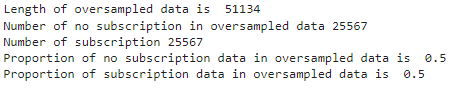
Краткий анализ полученной модели логистической регрессии:

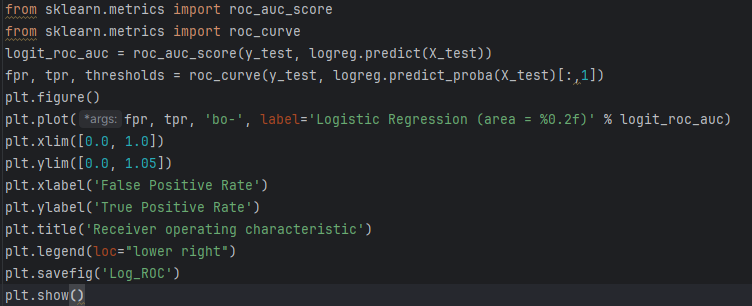
1. Текущее значение функции: Текущее значение функции составляет 0.547516. Это значение может быть связано с функцией правдоподобия или другой метрикой, используемой в процессе оптимизации модели.
2. Итерации: Для достижения оптимальных параметров модели потребовалось 7 итераций.
3. Pseudo R-squared: Значение псевдо R-квадрата составляет 0.210. Это показатель объяснительной способности модели, который указывает на то, насколько хорошо модель объясняет изменчивость зависимой переменной.
4. AIC и BIC: AIC (информационный критерий Акаике) равен 56025.3883, а BIC (критерий шеннона) равен 56166.8635. Эти критерии используются для сравнения моделей, где более низкие значения указывают на более предпочтительные модели.
5. Коэффициенты и p-значения: Для каждого выбранного признака выводятся их коэффициенты, стандартные ошибки, z-значения и p-значения. Эти статистики помогают определить влияние каждого признака на целевую переменную.
6. LLR p-value: Значение p-значения для LLR (likelihood ratio test) равно 0.0000. Это может указывать на статистическую значимость модели в целом.
7. Converged: Значение 1.0000 указывает на успешную сходимость модели.

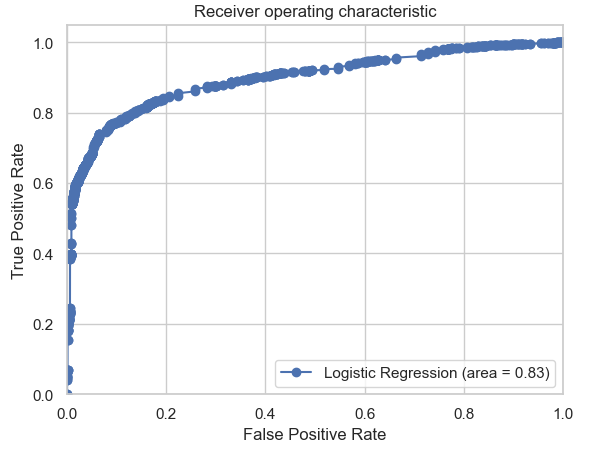
Процесс отбора признаков помогают выявить ключевые факторы, влияющие на целевую переменную, что может быть полезно для предсказаний, принятия решений и понимания важности различных аспектов исследуемого явления.

Обогащение синтетическими данными через SMOTE

1. 



1. 



Листинг:

import pandas as pd  
import numpy as np  
from sklearn import preprocessing  
import matplotlib.pyplot as plt  
plt.rc("font", size=14)  
from sklearn.linear\_model import LogisticRegression  
from sklearn.model\_selection import train\_test\_split  
import seaborn as sns  
sns.set(style="white")  
sns.set(style="whitegrid", color\_codes=True)  
data = pd.read\_csv('banking.csv', header = 0)  
data = data.dropna()  
print(data.shape)  
print(list(data.columns))  
data.head()  
data['education']=np.where(data['education'] =='basic.9y', 'Basic', data['education'])  
data['education']=np.where(data['education'] =='basic.6y', 'Basic', data['education'])  
data['education']=np.where(data['education'] =='basic.4y', 'Basic', data['education'])  
data['education'].unique()  
data['y'].value\_counts()  
count\_no\_sub = len(data[data['y']==0])  
count\_sub = len(data[data['y']==1])  
pct\_of\_no\_sub = count\_no\_sub/(count\_no\_sub+count\_sub)  
print("percentage of no subscription is", pct\_of\_no\_sub\*100)  
pct\_of\_sub = count\_sub/(count\_no\_sub+count\_sub)  
print("percentage of subscription", pct\_of\_sub\*100)  
data.groupby('job').mean()  
data.groupby('education').mean()  
cat\_vars=['job','marital','education','default','housing','loan','contact','month','day\_of\_week','poutcome']  
for var in cat\_vars:  
 cat\_list='var'+'\_'+var  
 cat\_list = pd.get\_dummies(data[var], prefix=var)  
 data1=data.join(cat\_list)  
 data=data1  
cat\_vars=['job','marital','education','default','housing','loan','contact','month','day\_of\_week','poutcome']  
data\_vars=data.columns.values.tolist()  
to\_keep=[i for i in data\_vars if i not in cat\_vars]  
data\_final=data[to\_keep]  
data\_final.columns.values  
data\_final\_vars=data\_final.columns.values.tolist()  
y=['y']  
X=[i for i in data\_final\_vars if i not in y]  
from sklearn.feature\_selection import RFE  
from sklearn.linear\_model import LogisticRegression  
logreg = LogisticRegression()  
rfe = RFE(logreg, 20)  
rfe = rfe.fit(os\_data\_X, os\_data\_y.values.ravel())  
print(rfe.support\_)  
print(rfe.ranking\_)  
cols=['euribor3m', 'job\_blue-collar', 'job\_housemaid', 'marital\_unknown', 'education\_illiterate',  
 'month\_apr', 'month\_aug', 'month\_dec', 'month\_jul', 'month\_jun', 'month\_mar',  
 'month\_may', 'month\_nov', 'month\_oct', "poutcome\_failure", "poutcome\_success"]  
X=os\_data\_X[cols]  
y=os\_data\_y['y']  
logit\_model=sm.Logit(y,X)  
result=logit\_model.fit()  
print(result.summary2())  
from imblearn.over\_sampling import SMOTE  
import pandas as pd  
from sklearn.model\_selection import train\_test\_split  
  
X\_train, X\_test, y\_train, y\_test = train\_test\_split(X, y, test\_size=0.3, random\_state=0)  
  
os = SMOTE(random\_state=0)  
os\_data\_X, os\_data\_y = os.fit\_resample(X\_train, y\_train)  
  
os\_data\_X = pd.DataFrame(data=os\_data\_X, columns=X.columns)  
os\_data\_y = pd.DataFrame(data=os\_data\_y, columns=['y'])  
  
print("Length of oversampled data is ", len(os\_data\_X))  
print("Number of no subscription in oversampled data", len(os\_data\_y[os\_data\_y['y'] == 0]))  
print("Number of subscription", len(os\_data\_y[os\_data\_y['y'] == 1]))  
print("Proportion of no subscription data in oversampled data is ", len(os\_data\_y[os\_data\_y['y'] == 0]) / len(os\_data\_X))  
print("Proportion of subscription data in oversampled data is ", len(os\_data\_y[os\_data\_y['y'] == 1]) / len(os\_data\_X))  
from sklearn.metrics import roc\_auc\_score  
from sklearn.metrics import roc\_curve  
logit\_roc\_auc = roc\_auc\_score(y\_test, logreg.predict(X\_test))  
fpr, tpr, thresholds = roc\_curve(y\_test, logreg.predict\_proba(X\_test)[:,1])  
plt.figure()  
plt.plot(fpr, tpr, 'bo-', label='Logistic Regression (area = %0.2f)' % logit\_roc\_auc)  
plt.xlim([0.0, 1.0])  
plt.ylim([0.0, 1.05])  
plt.xlabel('False Positive Rate')  
plt.ylabel('True Positive Rate')  
plt.title('Receiver operating characteristic')  
plt.legend(loc="lower right")  
plt.savefig('Log\_ROC')  
plt.show()

Вывод: Логистическая регрессия - это статистический метод, используемый для моделирования вероятности возникновения некоторого события в зависимости от набора входных переменных. В отличие от обычной линейной регрессии, которая предсказывает непрерывные значения, логистическая регрессия предсказывает вероятность принадлежности к определенному классу. Она широко используется в задачах классификации, таких как прогнозирование болезни на основе медицинских признаков или определение категории покупателя на основе его поведения.

ROC — другой распространённый в бинарной классификации инструмент. Восходящая прямая линия углового коэффициента 1 здесь представляет кривую рабочей характеристики приёмника для чисто случайного классификатора, при этом график хорошего классификатора держится от восходящей прямой как можно дальше, к левому верхнему углу.