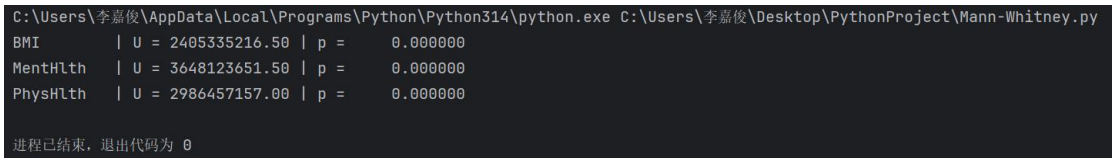
Mann Whitney python 代码

```
from ucimlrepo import fetch_ucirepo
import pandas as pd
from scipy.stats import mannwhitneyu

cdc = fetch_ucirepo(id=891)
df  = cdc.data.features.copy()
df['Diabetes_binary'] = cdc.data.targets

group0 = df[df['Diabetes_binary'] == 0]
group1 = df[df['Diabetes_binary'] == 1]

for col in ['BMI', 'MentHlth', 'PhysHlth']:
    u, p = mannwhitneyu(group0[col].dropna(),
                        group1[col].dropna(),
                        alternative='two-sided')
    print(f'{col:<10} | U = {u:>12.2f} | p = {p:>12.6f}')
```



这里最好加上为什么只选取了 BMI，MentHlth，PhysHlsh 三个变量来做 Mann-Whitney 检验。

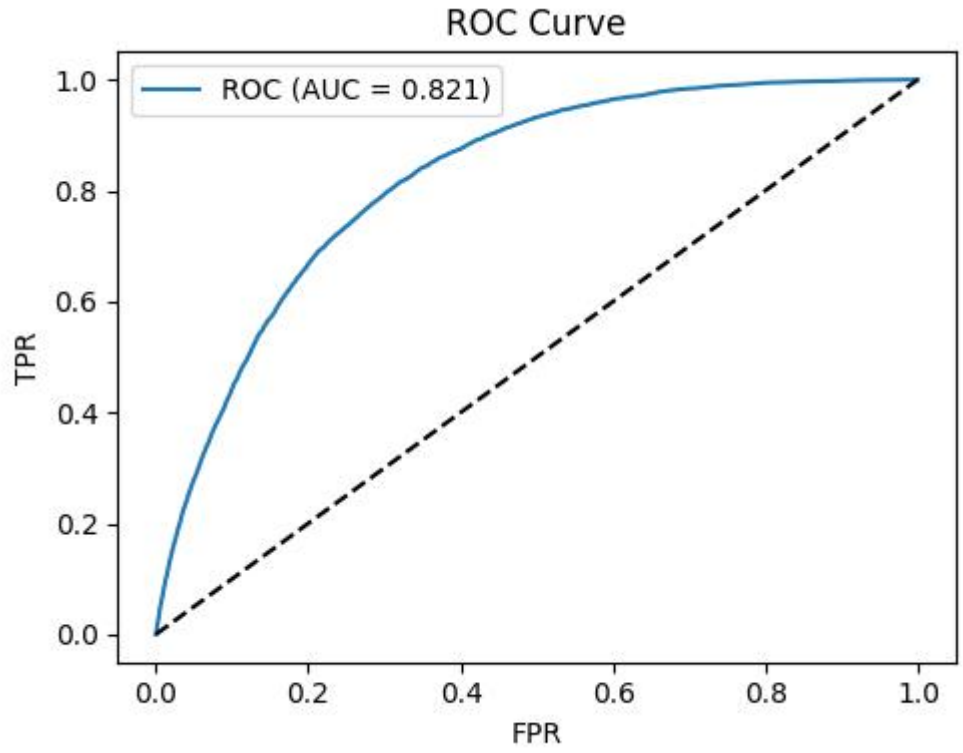
原因：只有这三个变量的取值范围比较大，对于糖尿病指示指标为 0 和 1 的两组数据，考虑这三个变量的位置参数是比较合理的，剩下的变量参数适合做分类而不适合比较位置参数。

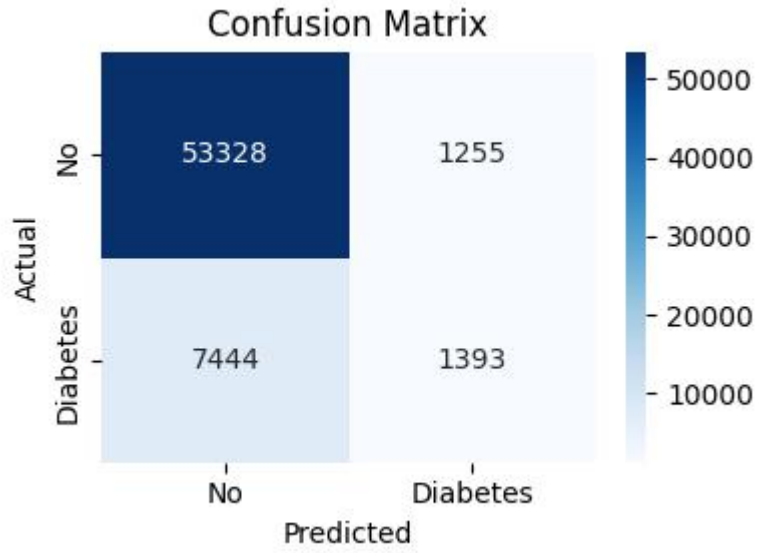
Logistic 回归代码（为什么不做普通线性回归是因为是否为糖尿病的指示变量取值只有两个，不适合做普通线性回归）

```
from ucimlrepo import fetch_ucirepo
import pandas as pd
import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt
from sklearn.linear_model import LogisticRegression
from sklearn.metrics import (classification_report, roc_auc_score,
                             roc_curve, confusion_matrix, accuracy_score)
from sklearn.model_selection import train_test_split

cdc = fetch_ucirepo(id=891)
X = cdc.data.features
y = cdc.data.targets
```

```
X_train, X_test, y_train, y_test = train_test_split(
      X, y, test_size=0.25, random_state=42, stratify=y)

logit = LogisticRegression(max_iter=1000)
logit.fit(X_train, y_train)

y_pred = logit.predict(X_test)
y_pred_proba = logit.predict_proba(X_test)[:, 1]

print(classification_report(y_test, y_pred))
print(f'Accuracy : {accuracy_score(y_test, y_pred):.3f}')
print(f'AUC          : {roc_auc_score(y_test, y_pred_proba):.3f}')

fpr, tpr, _ = roc_curve(y_test, y_pred_proba)
plt.figure(figsize=(5, 4))
sns.lineplot(x=fpr, y=tpr, label=f'ROC (AUC = {roc_auc_score(y_test, y_pred_proba):.3f})')
plt.plot([0, 1], [0, 1], 'k--')
plt.xlabel('FPR')
plt.ylabel('TPR')
plt.title('ROC Curve')
plt.tight_layout()
plt.show()

plt.figure(figsize=(4, 3))
sns.heatmap(confusion_matrix(y_test, y_pred),
                annot=True, fmt='d', cmap='Blues',
                xticklabels=['No', 'Diabetes'],
                yticklabels=['No', 'Diabetes'])
plt.xlabel('Predicted')
plt.ylabel('Actual')
plt.title('Confusion Matrix')
plt.tight_layout()
plt.show()

coef_df = (pd.DataFrame({'Feature': X.columns,
                                'Coefficient': logit.coef_[0],
                                'OR': np.exp(logit.coef_[0])})
             .assign(Abs_OR=lambda d: d['OR'].abs())
             .sort_values('Abs_OR', ascending=False))
print('\nTop 15 most influential factors (OR)')
print(coef_df.head(15).round(3))
```

```
C:\Users\李嘉俊\AppData\Local\Programs\Python\Python314\python.exe C:\Users\李嘉俊\Desktop\PythonProject\Mann-Whitney.py
C:\Users\李嘉俊\AppData\Local\Programs\Python\Python314\Lib\site-packages\sklearn\utils\validation.py:1406: DataConversionWarning: A column-vector y was passe
  y = column_or_1d(y, warn=True)
              precision    recall  f1-score   support

           0       0.88      0.98      0.92     54583
           1       0.53      0.16      0.24      8837

    accuracy                           0.86     63420
   macro avg       0.70      0.57      0.58     63420
weighted avg       0.83      0.86      0.83     63420

Accuracy : 0.863
AUC      : 0.821

Top 15 most influential factors (OR)
                 Feature  Coefficient     OR  Abs_OR
2              CholCheck        1.199  3.316   3.316
0                 HighBP        0.754  2.126   2.126
1               HighChol        0.582  1.790   1.790
13               GenHlth        0.531  1.700   1.700
17                   Sex        0.254  1.289   1.289
6   HeartDiseaseorAttack        0.229  1.258   1.258
5                 Stroke        0.143  1.154   1.154
18                   Age        0.125  1.133   1.133
16              DiffWalk        0.124  1.132   1.132
11         AnyHealthcare        0.092  1.096   1.096
3                   BMI        0.062  1.064   1.064
12           NoDocbcCost        0.019  1.019   1.019
14              MentHlth       -0.003  0.997   0.997
9                Veggies       -0.006  0.994   0.994
15              PhysHlth       -0.007  0.993   0.993

进程已结束，退出代码为 0
```




Confusion Matrix
53328
1255
7444
1393
No
Diabetes
Actual
No
Diabetes
Predicted
50000
40000
30000
20000
10000