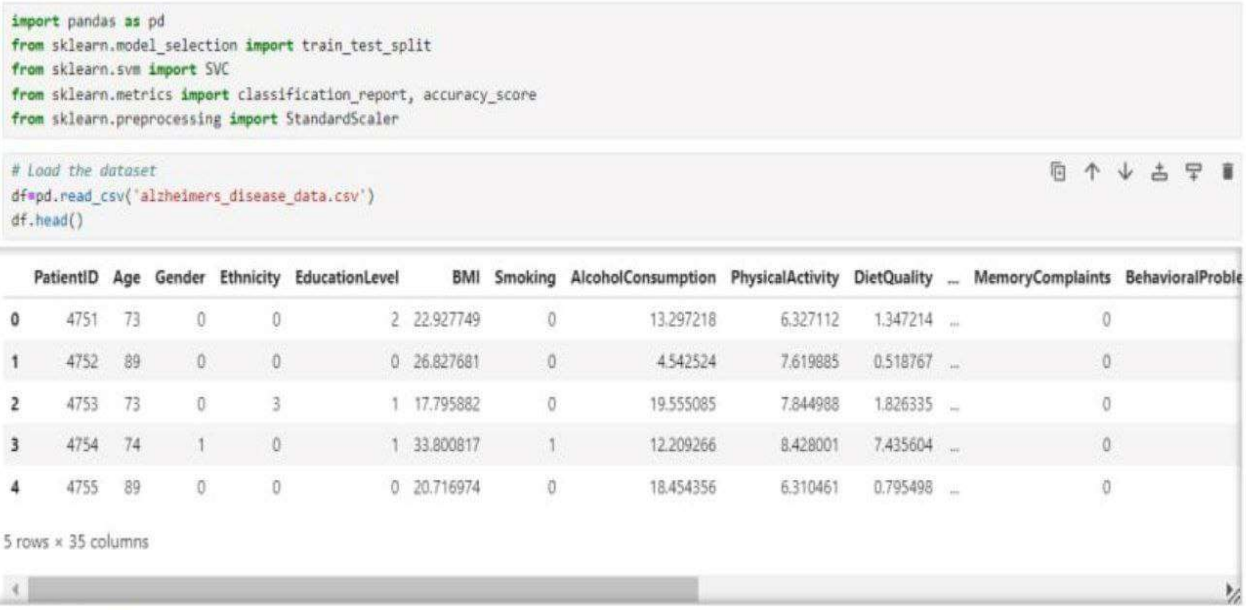
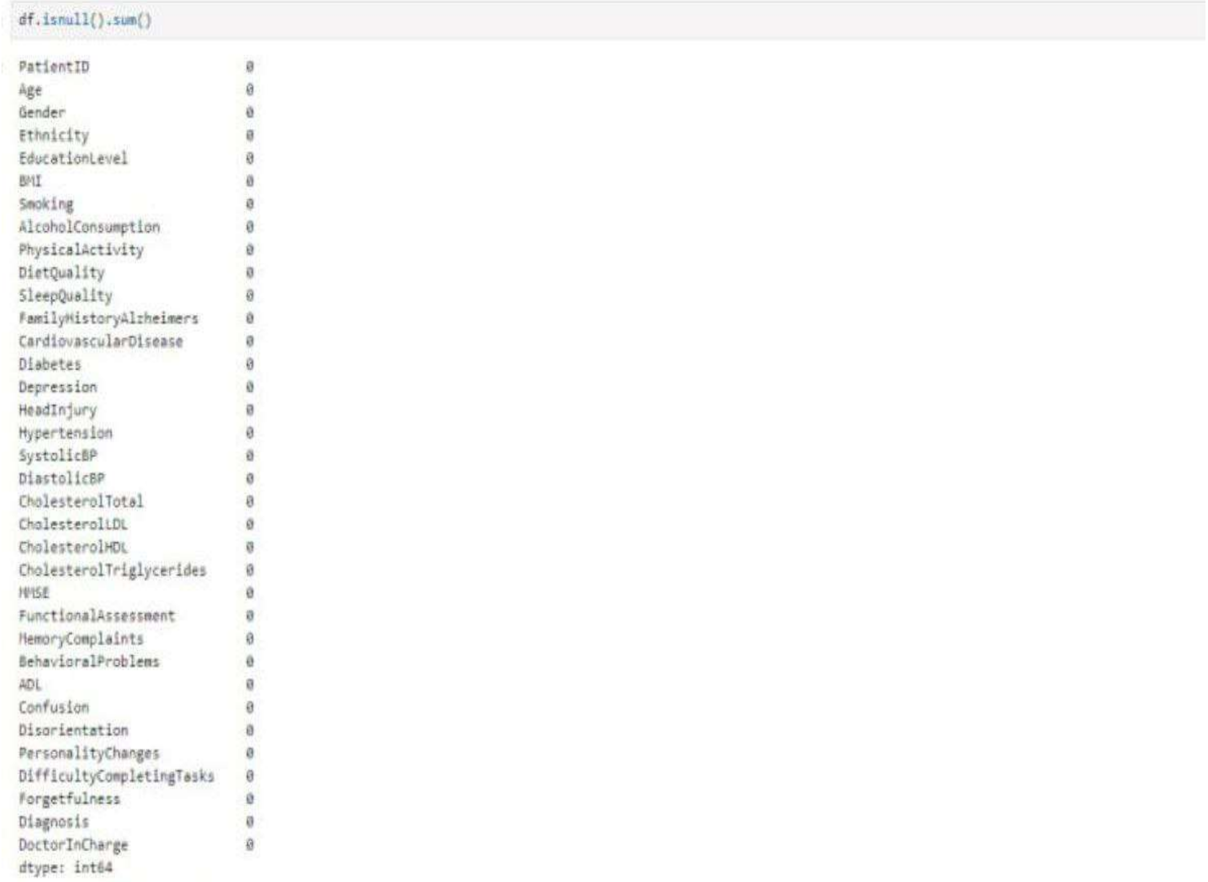


```
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.svm import SVC
from sklearn.metrics import classification_report, accuracy_score
from sklearn.preprocessing import StandardScaler
```

```
# Load the dataset
df=pd.read_csv('alzheimers_disease_data.csv')
df.head()
```

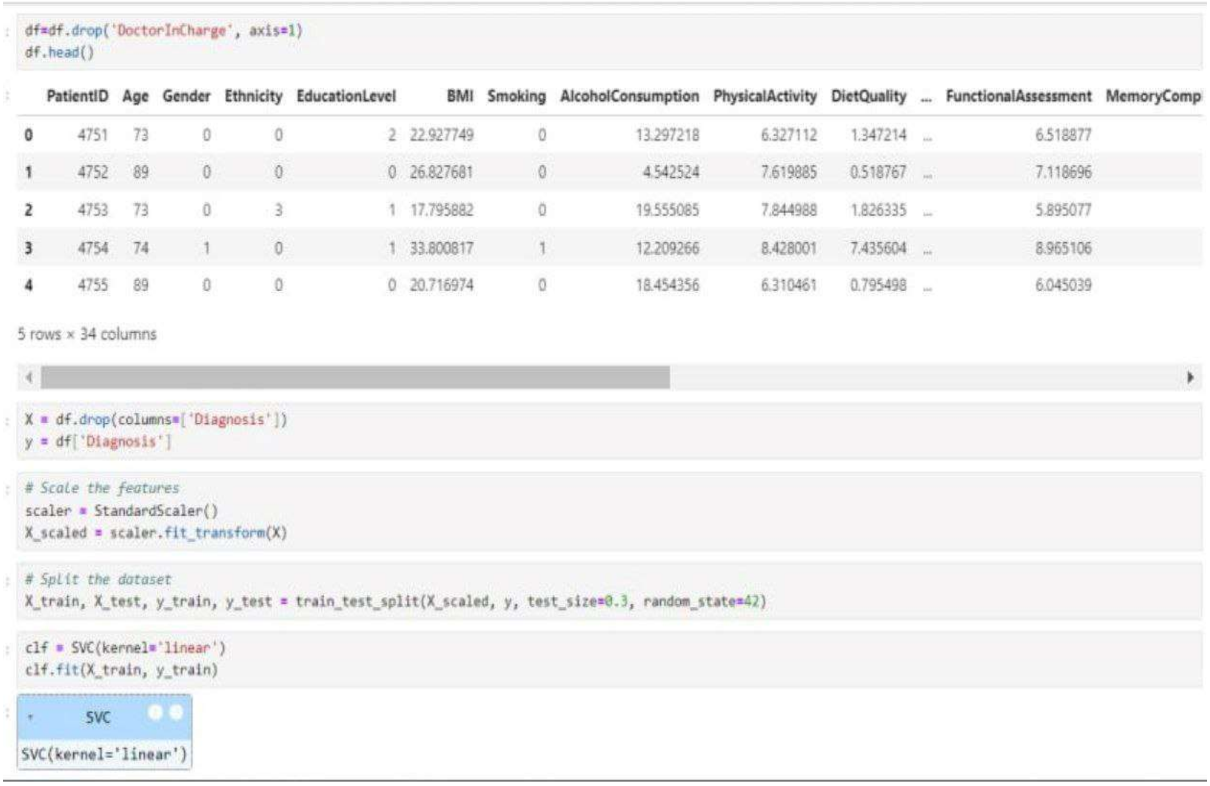
	PatientID	Age	Gender	Ethnicity	EducationLevel	BMI	Smoking	AlcoholConsumption	PhysicalActivity	DietQuality	...	MemoryComplaints	BehavioralProble
0	4751	73	0	0	2	22.927749	0	13.297218	6.327112	1.347214	...	0	
1	4752	89	0	0	0	26.827681	0	4.542524	7.619885	0.518767	...	0	
2	4753	73	0	3	1	17.795882	0	19.555085	7.844988	1.826335	...	0	
3	4754	74	1	0	1	33.800817	1	12.209266	8.428001	7.435604	...	0	
4	4755	89	0	0	0	20.716974	0	18.454356	6.310461	0.795498	...	0	

5 rows × 35 columns



```
df.isnull().sum()
```

```
PatientID                    0
Age                          0
Gender                       0
Ethnicity                    0
EducationLevel               0
BMI                          0
Smoking                      0
AlcoholConsumption           0
PhysicalActivity             0
DietQuality                  0
SleepQuality                 0
FamilyHistoryAlzheimers      0
CardiovascularDisease        0
Diabetes                     0
Depression                   0
HeadInjury                   0
Hypertension                 0
SystolicBP                   0
DiastolicBP                  0
CholesterolTotal             0
CholesterolLDL               0
CholesterolHDL               0
CholesterolTriglycerides     0
MMSE                         0
FunctionalAssessment         0
MemoryComplaints             0
BehavioralProblems           0
ADL                          0
Confusion                    0
Disorientation               0
PersonalityChanges           0
DifficultyCompletingTasks    0
Forgetfulness                0
Diagnosis                    0
DoctorInCharge               0
dtype: int64
```



```
df=df.drop('DoctorInCharge', axis=1)
df.head()
```

	PatientID	Age	Gender	Ethnicity	EducationLevel	BMI	Smoking	AlcoholConsumption	PhysicalActivity	DietQuality	...	FunctionalAssessment	MemoryComp
0	4751	73	0	0	2	22.927749	0	13.297218	6.327112	1.347214	...	6.518877	
1	4752	89	0	0	0	26.827681	0	4.542524	7.619885	0.518767	...	7.118696	
2	4753	73	0	3	1	17.795882	0	19.555085	7.844988	1.826335	...	5.895077	
3	4754	74	1	0	1	33.800817	1	12.209266	8.428001	7.435604	...	8.965106	
4	4755	89	0	0	0	20.716974	0	18.454356	6.310461	0.795498	...	6.045039	

5 rows × 34 columns

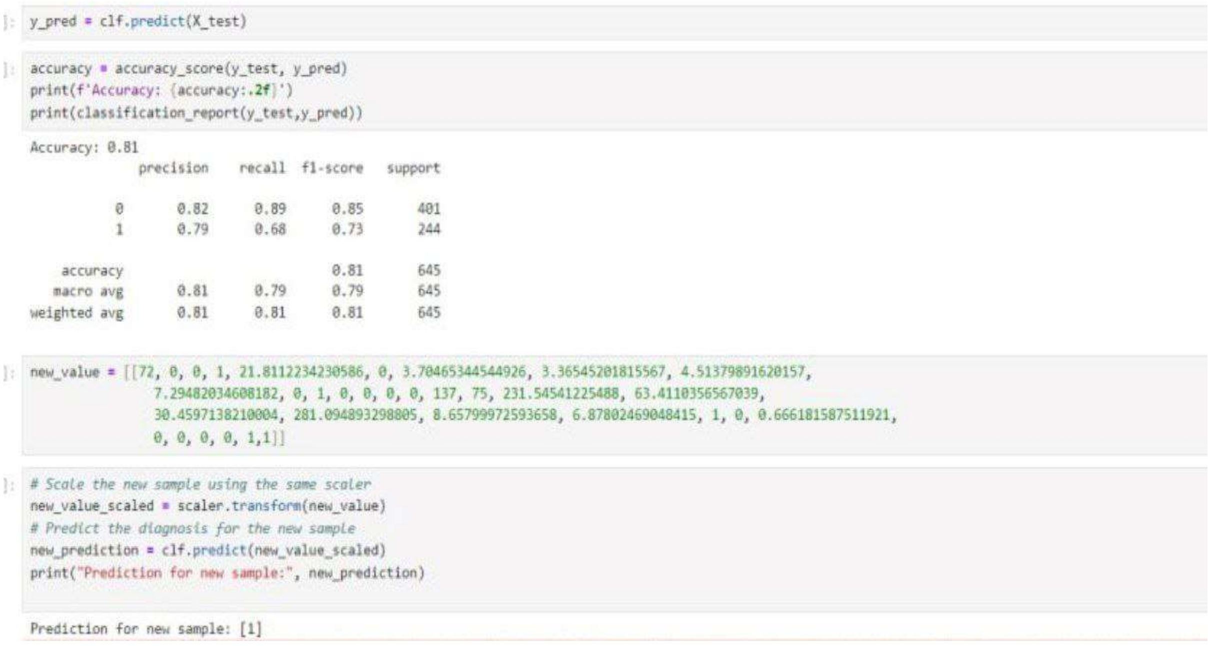
```
X = df.drop(columns=['Diagnosis'])
y = df['Diagnosis']
```

```
# Scale the features
scaler = StandardScaler()
X_scaled = scaler.fit_transform(X)
```

```
# Split the dataset
X_train, X_test, y_train, y_test = train_test_split(X_scaled, y, test_size=0.3, random_state=42)
```

```
clf = SVC(kernel='linear')
clf.fit(X_train, y_train)
```

```
SVC
SVC(kernel='linear')
```



```
y_pred = clf.predict(X_test)
```

```
accuracy = accuracy_score(y_test, y_pred)
print(f'Accuracy: {accuracy:.2f}')
print(classification_report(y_test,y_pred))
```

```
Accuracy: 0.81
              precision    recall  f1-score   support

           0       0.82      0.89      0.85       401
           1       0.79      0.68      0.73       244

    accuracy                           0.81       645
   macro avg       0.81      0.79      0.79       645
weighted avg       0.81      0.81      0.81       645
```

```
new_value = [[72, 0, 0, 1, 21.8112234230586, 0, 3.70465344544926, 3.36545201815567, 4.51379891620157,
               7.29482034608182, 0, 1, 0, 0, 0, 0, 137, 75, 231.54541225488, 63.4110356567039,
               30.4597138210004, 281.094893298805, 8.65799972593658, 6.87802469048415, 1, 0, 0.666181587511921,
               0, 0, 0, 0, 1,1]]
```

```
# Scale the new sample using the same scaler
new_value_scaled = scaler.transform(new_value)
# Predict the diagnosis for the new sample
new_prediction = clf.predict(new_value_scaled)
print("Prediction for new sample:", new_prediction)
```

```
Prediction for new sample: [1]
```