```
import pandas as pd
import numpy as np
```

```
from google.colab import files
files=files.upload()
```

```
Choose Files No file chosen  Upload widget is only available when the cell has been executed in the current browser session. Please rerun this cell to enable.
Saving heart.csv to heart (1).csv
```

```
data=pd.read_csv('heart.csv')
data.head()
```

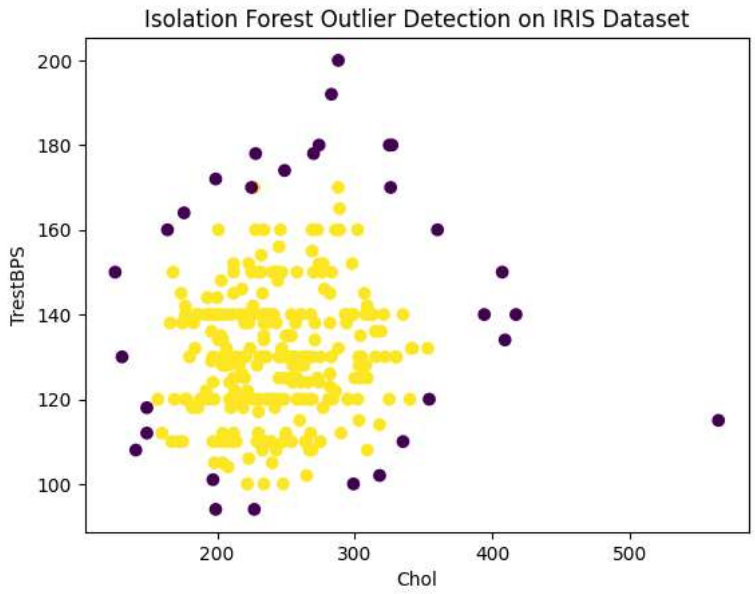
	age	sex	cp	trestbps	chol	fbs	restecg	thalach	exang	oldpeak	slope	ca	thal
0	63	1	3	145	233	1	0	150	0	2.3	0	0	1
1	37	1	2	130	250	0	1	187	0	3.5	0	0	2
2	41	0	1	130	204	0	0	172	0	1.4	2	0	2
3	56	1	1	120	236	0	1	178	0	0.8	2	0	2
4	57	0	0	120	354	0	1	163	1	0.6	2	0	2

```
X=data[['chol','trestbps']]
```

```
from sklearn.ensemble import IsolationForest
clf = IsolationForest(n_estimators=100, contamination=0.1, random_state=42)
clf.fit(X)
y_pred = clf.predict(X)
```

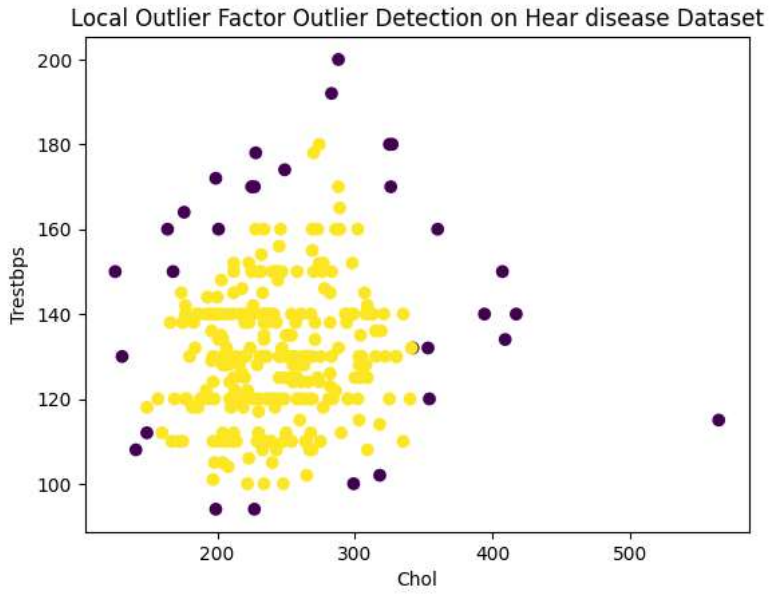
```
/usr/local/lib/python3.10/dist-packages/sklearn/base.py:439: UserWarning: X does not have valid feature names, but IsolationForest was f
  warnings.warn(
```

```
import matplotlib.pyplot as plt
plt.scatter(X.iloc[:, 0], X.iloc[:, 1], c=y_pred, cmap='viridis')
plt.title("Isolation Forest Outlier Detection on IRIS Dataset")
plt.xlabel("Chol")
plt.ylabel("TrestBPS")
plt.show()
```




```
from sklearn.neighbors import LocalOutlierFactor
clf = LocalOutlierFactor(n_neighbors=20, contamination=0.1)
y_pred = clf.fit_predict(X)
```

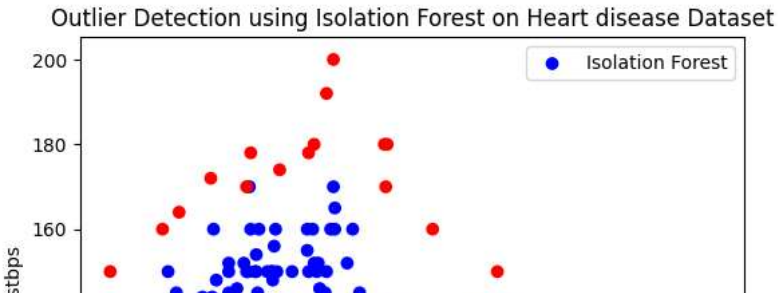
```
plt.scatter(X.iloc[:, 0], X.iloc[:, 1], c=y_pred, cmap='viridis')
plt.title("Local Outlier Factor Outlier Detection on Hear disease Dataset")
plt.xlabel("Chol")
plt.ylabel("Trestbps")
plt.show()
```




```
from sklearn.ensemble import IsolationForest
from sklearn.neighbors import LocalOutlierFactor


y=data['target']


# Fit Isolation Forest model
clf_iso = IsolationForest(contamination=0.1, random_state=42)
y_pred_iso = clf_iso.fit_predict(X)
```

```
/usr/local/lib/python3.10/dist-packages/sklearn/base.py:439: UserWarning: X does not have valid feature names, but IsolationForest was f
  warnings.warn(
```

```
# Fit Local Outlier Factor model
clf_lof = LocalOutlierFactor(contamination=0.1)
y_pred_lof = clf_lof.fit_predict(X)



# Plot Isolation Forest outliers
plt.scatter(X.iloc[:, 0], X.iloc[:, 1], c=np.where(y_pred_iso == -1, 'red', 'blue'), label='Isolation Forest')
plt.title("Outlier Detection using Isolation Forest on Heart disease Dataset")
plt.xlabel("Chol")
plt.ylabel("Trestbps")
plt.legend()
plt.show()
```




```
# Plot Local Outlier Factor outliers
plt.scatter(X.iloc[:, 0], X.iloc[:, 1], c=np.where(y_pred_lof == -1, 'red', 'blue'), label='Local Outlier Factor')
plt.title("Outlier Detection using Isolation Forest on Heart disease Dataset")
plt.xlabel("Chol")
plt.ylabel("Trestbps")
plt.legend()
plt.show()
```