

```

# =====
# STEP 1: 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
from sklearn.linear_model import LinearRegression, LogisticRegression
from sklearn.metrics import mean_absolute_error, mean_squared_error,
r2_score
from sklearn.metrics import accuracy_score, confusion_matrix,
classification_report
from sklearn.preprocessing import LabelEncoder

```

Code for expl

```

# =====
# STEP 2: Load Dataset
# =====

df = pd.read_csv("healthcare-dataset-stroke-data.csv")

print(df.head())

# =====
# STEP 3: Data Cleaning
# =====

# Drop ID column
df.drop("id", axis=1, inplace=True)

```

```

# Fill missing BMI with mean
df["bmi"].fillna(df["bmi"].mean(), inplace=True)

# =====
# STEP 4: Encode Categorical Columns
# =====

le = LabelEncoder()

cat_cols =
["gender", "ever_married", "work_type", "Residence_type", "smoking_status"]

for col in cat_cols:
    df[col] = le.fit_transform(df[col])

print("\nAfter Encoding:")
print(df.head())

# =====
# PART A - LINEAR REGRESSION (Predict BMI)
# =====

print("\n===== LINEAR REGRESSION =====")

X = df.drop("bmi", axis=1)
y = df["bmi"]

X_train, X_test, y_train, y_test = train_test_split(
    X, y, test_size=0.2, random_state=42
)

lr = LinearRegression()
lr.fit(X_train, y_train)

y_pred_lr = lr.predict(X_test)

```

```

print("MAE:", mean_absolute_error(y_test, y_pred_lr))
print("MSE:", mean_squared_error(y_test, y_pred_lr))
print("RMSE:", np.sqrt(mean_squared_error(y_test, y_pred_lr)))
print("R2 Score:", r2_score(y_test, y_pred_lr))

plt.scatter(y_test, y_pred_lr)
plt.xlabel("Actual BMI")
plt.ylabel("Predicted BMI")
plt.title("Linear Regression: BMI Prediction")
plt.show()

# =====
# PART B - LOGISTIC REGRESSION (Predict Stroke)
# =====

print("\n===== LOGISTIC REGRESSION =====")

X = df.drop("stroke", axis=1)
y = df["stroke"]

X_train, X_test, y_train, y_test = train_test_split(
    X, y, test_size=0.2, random_state=42
)

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

y_pred_log = log_model.predict(X_test)

print("Accuracy:", accuracy_score(y_test, y_pred_log))
print("\nConfusion Matrix:\n", confusion_matrix(y_test, y_pred_log))
print("\nClassification Report:\n", classification_report(y_test,
y_pred_log))

sns.heatmap(confusion_matrix(y_test, y_pred_log), annot=True, fmt="d")
plt.xlabel("Predicted")
plt.ylabel("Actual")
plt.title("Confusion Matrix - Stroke Prediction")
plt.show()

```