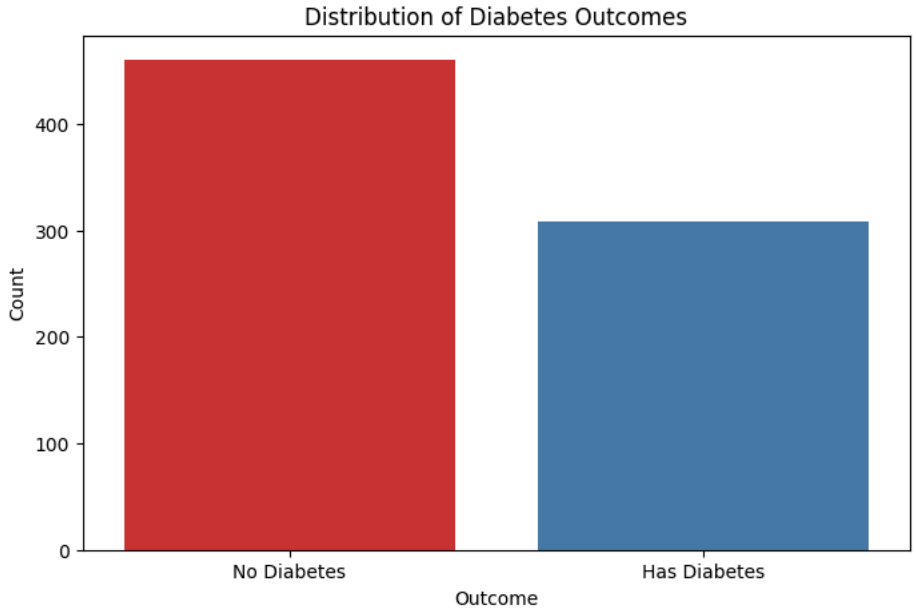
```
# Step 1: Import necessary libraries
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
from sklearn.model_selection import train_test_split
from sklearn.naive_bayes import GaussianNB
from sklearn.metrics import accuracy_score, confusion_matrix, roc_curve, auc

# Step 2: Load the CSV data
from google.colab import files
uploaded = files.upload()
df = pd.read_csv(next(iter(uploaded)))

# Step 3: Explore the data
print(df.head())

# Step 4: Visualize the Outcome column

plt.figure(figsize=(8, 5))
sns.countplot(x='Outcome', hue='Outcome', data=df, palette='Set1', legend=False)
plt.title('Distribution of Diabetes Outcomes')
plt.xticks([0, 1], ['No Diabetes', 'Has Diabetes'])
plt.ylabel('Count')
plt.show()

# Step 5: Prepare the feature variables (X) and the target label (y)
X = df.drop('Outcome', axis=1)
y = df['Outcome']

# Step 6: Split the data into training and test sets
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)

# Step 7: Create a Gaussian Naive Bayes model and train it
model = GaussianNB()
model.fit(X_train, y_train)

# Step 8: Make predictions on the test set
y_pred = model.predict(X_test)

# Step 9: Evaluate the model's performance
accuracy = accuracy_score(y_test, y_pred)
print(f"Model Accuracy: {accuracy * 100:.2f}%")

# Step 10: Predict on a new patient (example)
new_patient = [[5, 116, 74, 0, 0, 25.6, 0.201, 30]]  # Example patient data

# Convert to DataFrame with feature names
new_patient_df = pd.DataFrame(new_patient, columns=X.columns)

# Predict using the DataFrame (with correct feature names)
predicted = model.predict(new_patient_df)
print(f"Predicted Value for new patient: {'Has Diabetes' if predicted[0] == 1 else 'No Diabetes'}")

# Step 11: Visualize confusion matrix
conf_matrix = confusion_matrix(y_test, y_pred)
sns.heatmap(conf_matrix, annot=True, fmt='d', cmap='Blues', xticklabels=['No Diabetes', 'Has Diabetes'], yticklabels=['No Diabetes', 'Has
plt.title('Confusion Matrix')
plt.ylabel('True Label')
plt.xlabel('Predicted Label')
plt.show()
```

Choose Files diabetes v2.csv

- **diabetes v2.csv**(text/csv) - 23936 bytes, last modified: 10/24/2024 - 100% done

```
Saving diabetes v2.csv to diabetes v2 (14).csv
   Pregnancies  Glucose  BloodPressure  SkinThickness  Insulin   BMI  \
0            6      148             72             35      0.0  33.6
1            1       85             66             29      0.0  26.6
2            8       18             64              0      0.0  23.3
3            1       89             66             23     94.0  28.1
4            0      137             40             35    168.0  43.1

   DiabetesPedigreeFunction  Age  Outcome
0                     0.627   50        1
1                     0.351   31        0
2                     0.672   32        1
3                     0.167   21        0
4                     2.288   33        1
```




```
Model Accuracy: 74.03%
Predicted Value for new patient: No Diabetes
```

New Section