Assignment-1 Report

Introduction

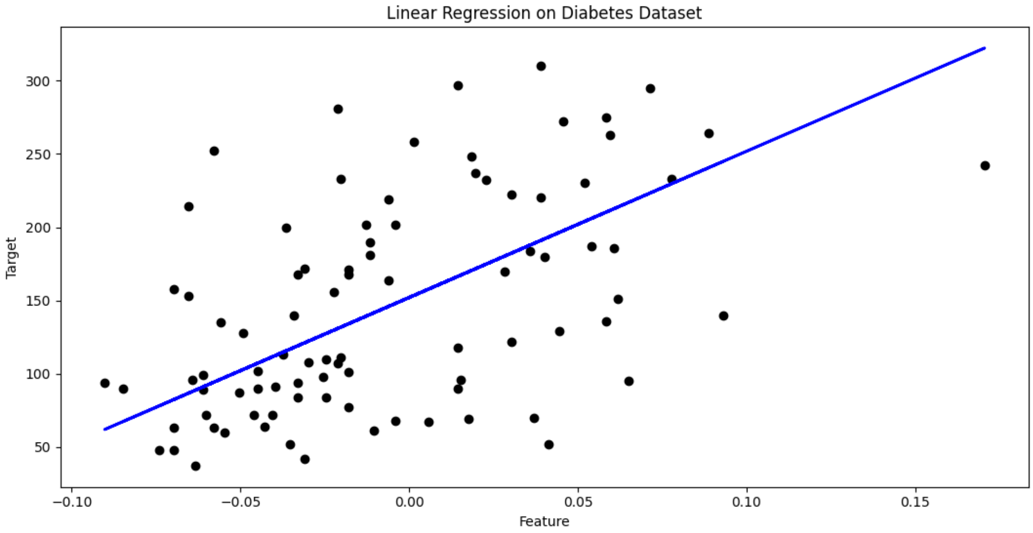
This report contains two experiments using datasets available in scikit-learn.

Linear Regression (Diabetes)

Code:

```
# linear_regression_diabetes.py
import matplotlib.pyplot as plt
import numpy as np
from sklearn.datasets import load_diabetes
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LinearRegression
from sklearn.metrics import mean_squared_error, r2_score

# Load dataset
diabetes = load_diabetes()
X = diabetes.data[:, np.newaxis, 2]   # take just one feature for visualization
y = diabetes.target

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

# Train model
model = LinearRegression()
model.fit(X_train, y_train)

# Predict
y_pred = model.predict(X_test)

# Evaluate
print("Mean squared error:", mean_squared_error(y_test, y_pred))
print("R2 score:", r2_score(y_test, y_pred))

# Plot
plt.scatter(X_test, y_test, color="black")
plt.plot(X_test, y_pred, color="blue", linewidth=2)
plt.title("Linear Regression on Diabetes Dataset")
plt.xlabel("Feature")
plt.ylabel("Target")
plt.show()
```

OUTPUT



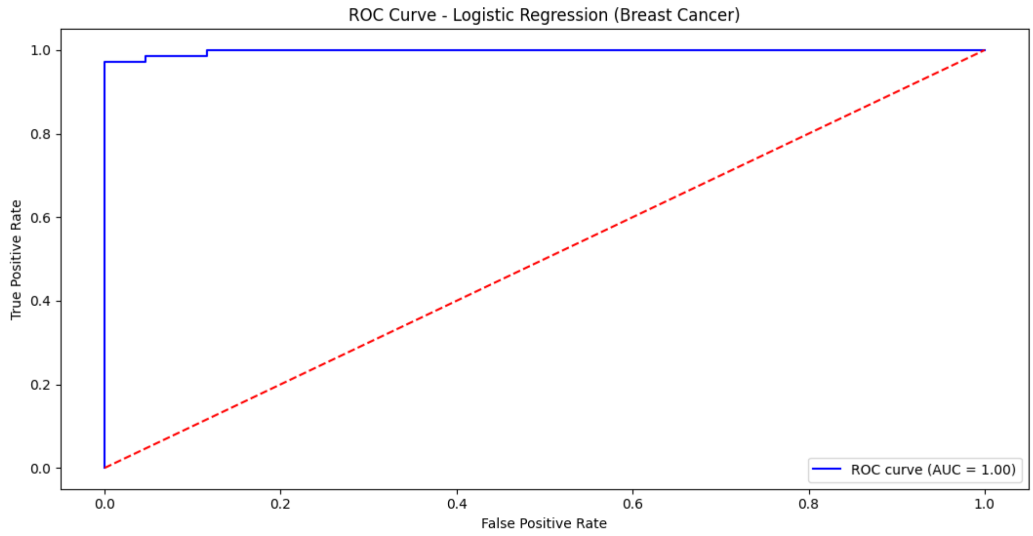
Logistic Regression (Breast Cancer)

Code:

```
# logistic_regression_breastcancer.py
import matplotlib.pyplot as plt
```

```
from sklearn.datasets import load_breast_cancer
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LogisticRegression
from sklearn.metrics import confusion_matrix, classification_report, roc_curve, auc

# Load dataset
cancer = load_breast_cancer()
X = cancer.data
y = cancer.target

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

# Train model
model = LogisticRegression(max_iter=5000)
model.fit(X_train, y_train)

# Predict
y_pred = model.predict(X_test)


# Evaluate
print("Confusion Matrix:\n", confusion_matrix(y_test, y_pred))
print("Report:\n", classification_report(y_test, y_pred))

# ROC Curve
y_prob = model.predict_proba(X_test)[:, 1]
fpr, tpr, _ = roc_curve(y_test, y_prob)
roc_auc = auc(fpr, tpr)

plt.plot(fpr, tpr, color="blue", label=f"ROC curve (AUC = {roc_auc:.2f})")
plt.plot([0, 1], [0, 1], color="red", linestyle="--")
plt.xlabel("False Positive Rate")
plt.ylabel("True Positive Rate")
plt.title("ROC Curve - Logistic Regression (Breast Cancer)")
plt.legend()
plt.show()
```

OUTPUT



Conclusion

Both models demonstrate simple applications of Linear and Logistic Regression using scikit-learn datasets.