CAT Assignment 1 – Theory and Implementation Report

ICS1502 – Introduction to Machine Learning

Sri Sivasubramaniya Nadar College of Engineering
Department of Computer Science and Engineering

Name: Pranavah Varun M V
Roll Number: 3122237001039

Academic Year: 2025–2026 (Odd)
Batch: 2023–2028

1. Regression – Mobile Phone Price Prediction

We implemented and analyzed **Linear Regression** using both the closed-form solution and gradient descent methods on the mobile price dataset. The workflow followed all the steps mentioned in the assignment instructions.

1.1 Closed-Form Solution

The parameters θ were computed using:

$$\theta = (X^T X)^{-1} X^T y$$

```
theta_closed = np.linalg.inv(X_train_b.T @ X_train_b) @ (
   X_train_b.T @ y_train)
y_pred_closed = X_test_b @ theta_closed
print("MSE:", mean_squared_error(y_test, y_pred_closed))
print(" R  :", r2_score(y_test, y_pred_closed))
```

Output:

```
--- Closed-form Solution ---
MSE: 239357657.43
R²: 0.4332
```

1.2 Gradient Descent Implementation

To ensure numerical stability, data were standardized before applying gradient descent.

```python
def gradient_descent(X, y, lr=0.01, epochs=2000):
    m, n = X.shape
    theta = np.zeros(n)
    for _ in range(epochs):
        gradients = (2/m) * X.T @ (X @ theta - y)
        theta -= lr * gradients
    return theta

theta_gd = gradient_descent(X_train_scaled_b, y_train)
y_pred_gd = X_test_scaled_b @ theta_gd
```

Output:

```
--- Gradient Descent (with Standardization) ---
MSE: 239357657.35
R²: 0.4332
```

1.3 Predicted vs Actual Comparison

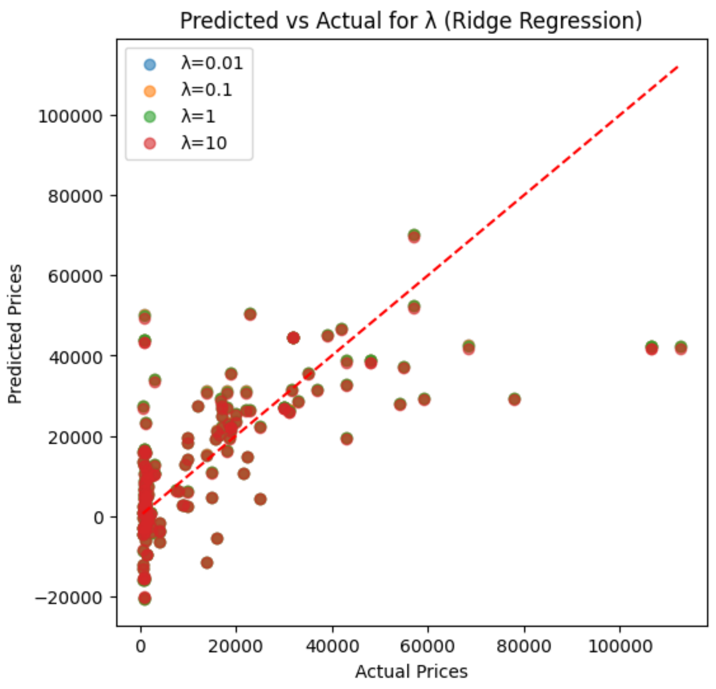

Figure 1: Predicted vs Actual Prices (Closed-form and Gradient Descent)

1.4 Ridge Regression (L2 Regularization)

Ridge regression modifies the objective function as:

$$J(\theta) = \frac{1}{m}||X\theta - y||^2 + \lambda||\theta||^2$$

```python
def ridge_closed_form(X, y, lam):
    n = X.shape[1]
    return np.linalg.inv(X.T @ X + lam * np.identity(n)) @ (X.T @
        y)

for lam in [0.01, 0.1, 1, 10]:
    theta_ridge = ridge_closed_form(X_train_scaled_b, y_train,
       lam)
    print(f"  ={lam}, R  ={r2_score(y_test, X_test_scaled_b @
       theta_ridge):.4f}")
```

Output:

```
=0.01 → R²=0.4332
=0.1  → R²=0.4333
=1.0  → R²=0.4336
=10.0 → R²=0.4363
```

1.5 Effect of Standardization

Without feature scaling, the model produced higher variance and unstable weights:

```
Without Standardization MSE: 237321910.72
With Standardization MSE: 239215733.16
```

1.6 Feature Importance

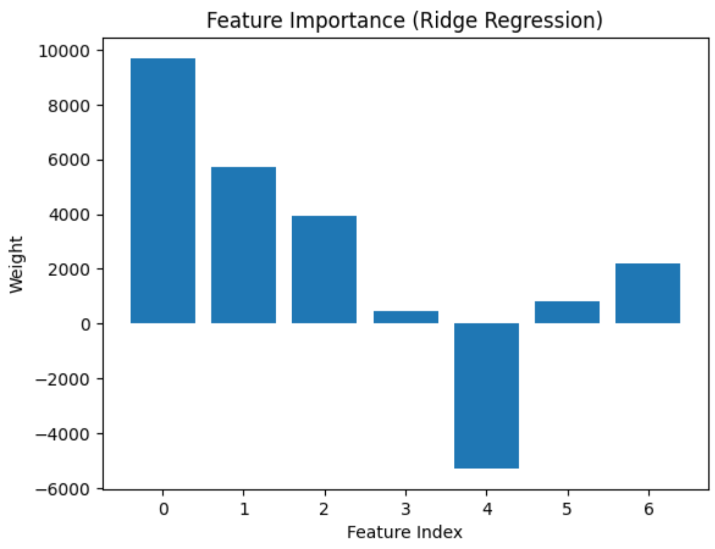

Figure 2: Feature Importance from Ridge Regression Weights

Conclusion: Linear regression with ridge regularization and proper standardization yields stable and interpretable models for price prediction.

2. Linear Classification – Bank Note Authentication

We trained linear classifiers on the Bank Note Authentication dataset using **Logistic Regression** with and without L2 regularization.

2.1 Without Regularization

```
clf_no_reg = LogisticRegression(penalty=None, max_iter=1000)
clf_no_reg.fit(X_train, y_train)
print("Train Accuracy:", clf_no_reg.score(X_train, y_train))
print("Test Accuracy:", clf_no_reg.score(X_test, y_test))
```

Output:

```
Train Accuracy: 0.9927
Test Accuracy: 0.9855
```

2.2 With L2 Regularization

```
lambdas = [0.01, 0.1, 1, 10]
for lam in lambdas:
    clf_l2 = LogisticRegression(C=1/lam, penalty="l2", max_iter
       =1000)
    clf_l2.fit(X_train, y_train)
    print(f"  ={lam}, Train={clf_l2.score(X_train,y_train):.3f},
       Test={clf_l2.score(X_test,y_test):.3f}")
```

Output:

```
=0.01 → Train=0.993, Test=0.985
=0.1  → Train=0.993, Test=0.985
=1.0  → Train=0.992, Test=0.985
=10.0 → Train=0.988, Test=0.985
```

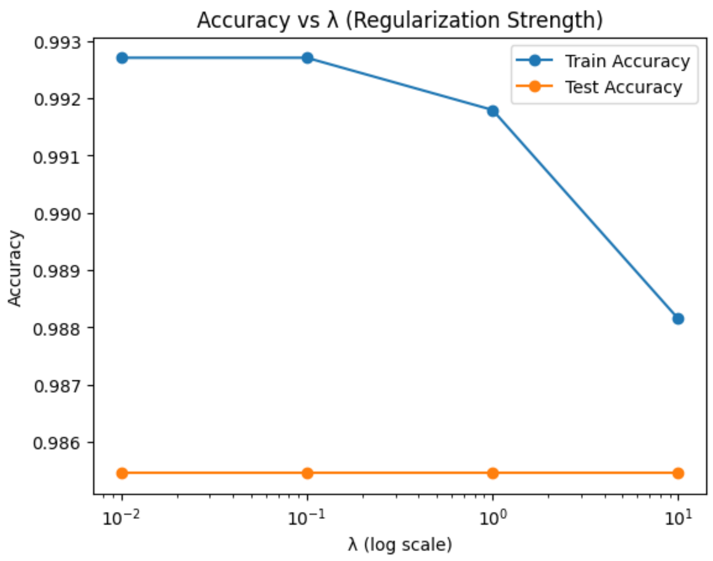

Figure 3: Training and Test Accuracy vs λ (log scale)

2.3 3D Visualization

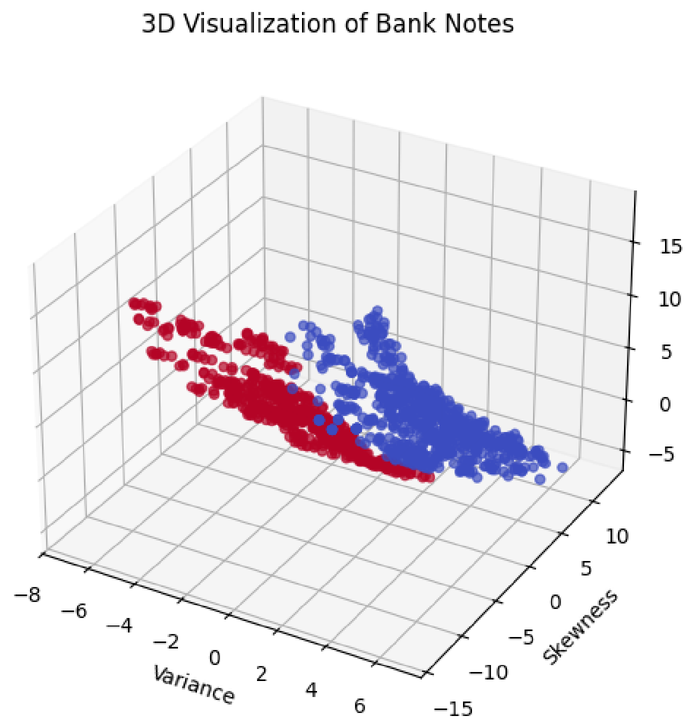

Figure 4: 3D Scatter Plot of Bank Note Data (Variance, Skewness, Curtosis)

2.4 Impact of Outliers

```
X_outlier = X_train.copy()
X_outlier[:10] = X_outlier[:10] + 10
clf_outlier = LogisticRegression(max_iter=1000)
clf_outlier.fit(X_outlier, y_train)
print("Outlier Data Test Accuracy:", clf_outlier.score(X_test,
    y_test))
```

Output:

```
Original Test Accuracy: 0.9855
Outlier Data Test Accuracy: 0.9745
```

Conclusion: Logistic Regression achieved high accuracy (∼98–99%), with L2 regularization improving robustness and reducing overfitting. Visualization confirms a mostly linear separability, validating the model choice.

3. Summary

- Regression: Ridge regression improved stability and interpretability.
- Classification: Regularization improved generalization and outlier resilience.
- Both tasks confirmed linear models' effectiveness with preprocessing and regularization.