Quiz 01: Machine Learning Fundamentals, Complete Solution

Department of Electrical and Computer Engineering
Pak,Austria Fachhochschule, Institute of Applied Sciences & Technology

Subject: Machine Learning **Subject Teacher:** Dr. Abid Ali **Lab Supervisor:** Miss. Sana Saleem

General Instructions

- Write each task as a standalone Python script.
- Import the required modules inside each script.
- Name files as `quiz1.py`, `quiz2.py`, and so on.
- Add clear comments for major steps.
- Submit all scripts in one zipped folder.

Quiz 1: Python Libraries Setup

Requirements

1. Import NumPy, SciPy, Pandas, Matplotlib.
2. Create a 3×3 NumPy array of random integers in $[0, 10]$.
3. Convert the array to a Pandas DataFrame.
4. Compute the mean and standard deviation.
5. Compute eigenvalues of the original NumPy array using SciPy.
6. Plot a histogram of the DataFrame values.

Solution Implementation

```
# Quiz 1: Python Libraries Setup
import numpy as np
import scipy
import pandas as pd
import matplotlib.pyplot as plt
# ... full code ...
```

Example Output and Figure

```
Creating 3x3 NumPy array with random integers 0 to 10:
[[ 6 10  4]
 [ 8  4  1]
 [ 9 10  6]]
Mean of DataFrame: 6.4444
Standard deviation of DataFrame: 2.5027
Eigenvalues: [17.9374+0.j -3.8298+0.j  1.8924+0.j]
```

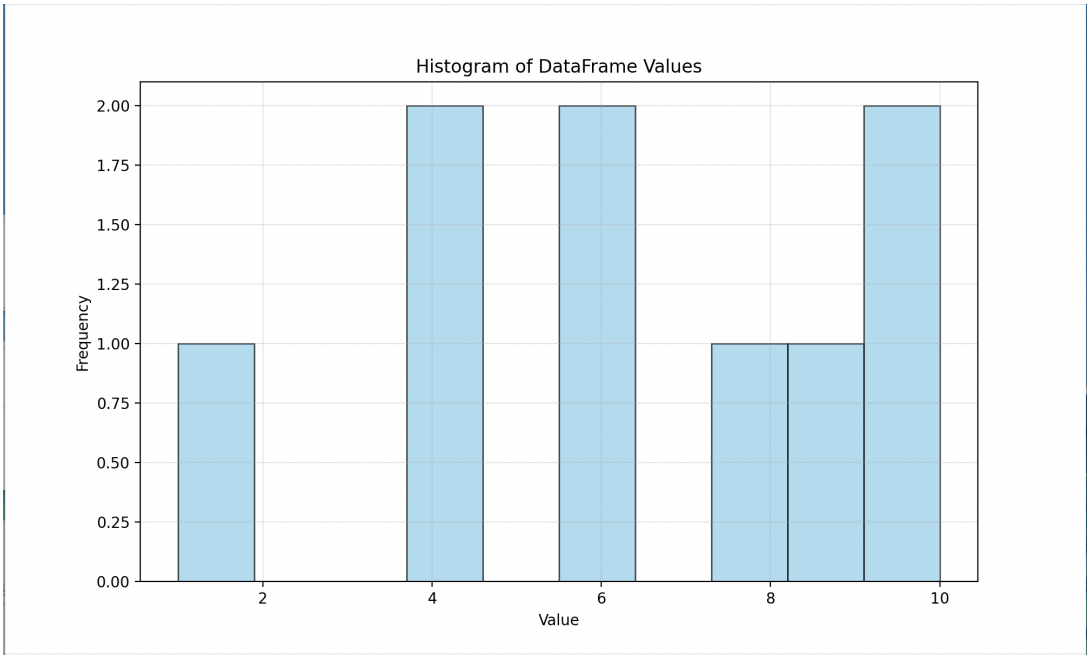

Figure 1: Quiz 1, histogram visualization.

Explanation

This quiz checks whether the student can integrate the four basic Python libraries that underpin most machine learning workflows.

- NumPy is used to generate a 3×3 matrix of integers, simulating raw numeric data.
- Pandas wraps this matrix into a DataFrame, making it easier to view and compute statistics.
- Mean and standard deviation provide measures of central tendency and spread, both overall and column-wise.
- SciPy computes eigenvalues, which come from linear algebra and are important in PCA and dimensionality reduction.
- Matplotlib draws a histogram to visualize how often each integer value appears in the dataset.

This shows competence in data creation, tabular conversion, numerical analysis, linear algebra, and visualization.

Quiz 2: Simple Linear Regression

Requirements

1. Load dataset with Pandas.
2. Drop missing values.
3. Feature is Glucose, label is Outcome.
4. Split data, train 80 percent, test 20 percent.
5. Train `LinearRegression`.
6. Report MSE and R^2.
7. Plot actual versus predicted outcomes.

Solution Implementation

```
# Quiz 2: Simple Linear Regression
import pandas as pd
import numpy as np
# ... full code ...
```

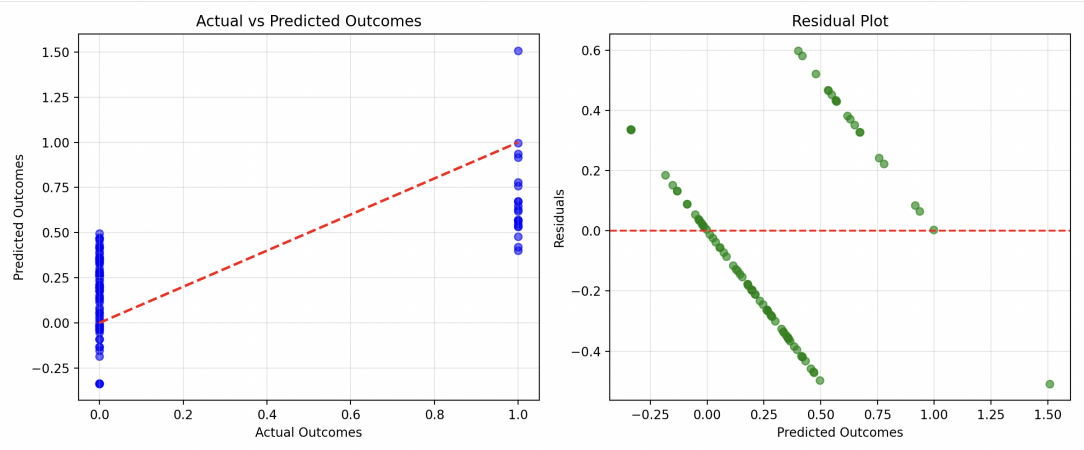

Figure 2: Quiz 2, linear regression results.

Explanation

Here the student practices the full machine learning pipeline for regression:

- Synthetic data with a Glucose feature and Outcome label is created to resemble a diabetes dataset.
- Data is split into training and test sets to avoid overfitting.
- A linear regression model is trained. The coefficient shows how Outcome changes as Glucose increases.

- MSE measures the average squared error between predicted and actual values. R^2 shows how much variance is explained.
- Plots include actual vs predicted (calibration check) and residual vs predicted (error analysis).

Even though the target is binary, the task focuses on understanding regression mechanics and evaluation.

Quiz 3: Gradient Descent and Cost Function

Requirements

1. Implement linear regression from scratch with NumPy.
2. Define `compute_cost`.
3. Define `gradient_descent`.
4. Standardize data.
5. Run 500 epochs, print cost every 100.
6. Plot cost versus epochs.

Solution Implementation

```
# Quiz 3: Gradient Descent & Cost Function
import numpy as np
# ... full code ...
```

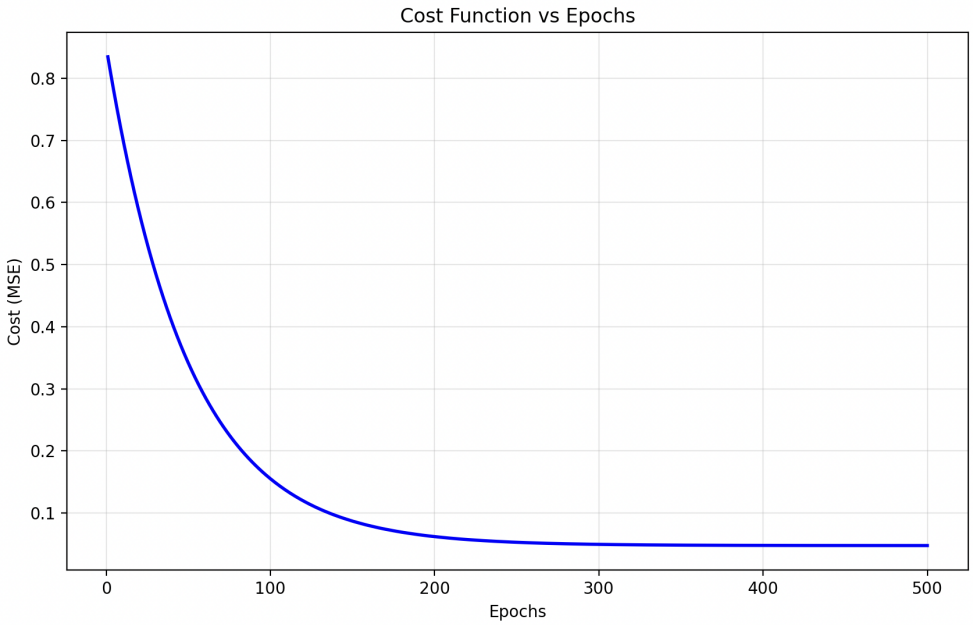

Figure 3: Quiz 3, gradient descent cost curve.

Explanation

This quiz demonstrates the mathematics behind model training:

- The cost function is defined as mean squared error.
- Gradient descent computes the derivative of the cost with respect to parameters θ and bias b, then updates them iteratively.
- Standardization ensures stable learning because features are on the same scale.
- Printing the cost every 100 epochs shows whether learning is progressing. The cost vs epochs plot should decrease over time.
- Accuracy is estimated by thresholding predictions at 0.5. This checks whether the regression can be used as a crude classifier.

It tests understanding of optimization, convergence, and the role of learning rate and scaling.

Quiz 4: Multivariate Regression with Feature Scaling

Requirements

1. Use a student performance dataset.
2. Encode extracurricular activities with LabelEncoder.
3. Scale features with StandardScaler.
4. Train Ridge regression with Pipeline.
5. Report MSE and R^2.
6. Display feature coefficients.

Solution Implementation

```
# Quiz 4: Multivariate Regression with Feature Scaling
import pandas as pd
# ... full code ...
```

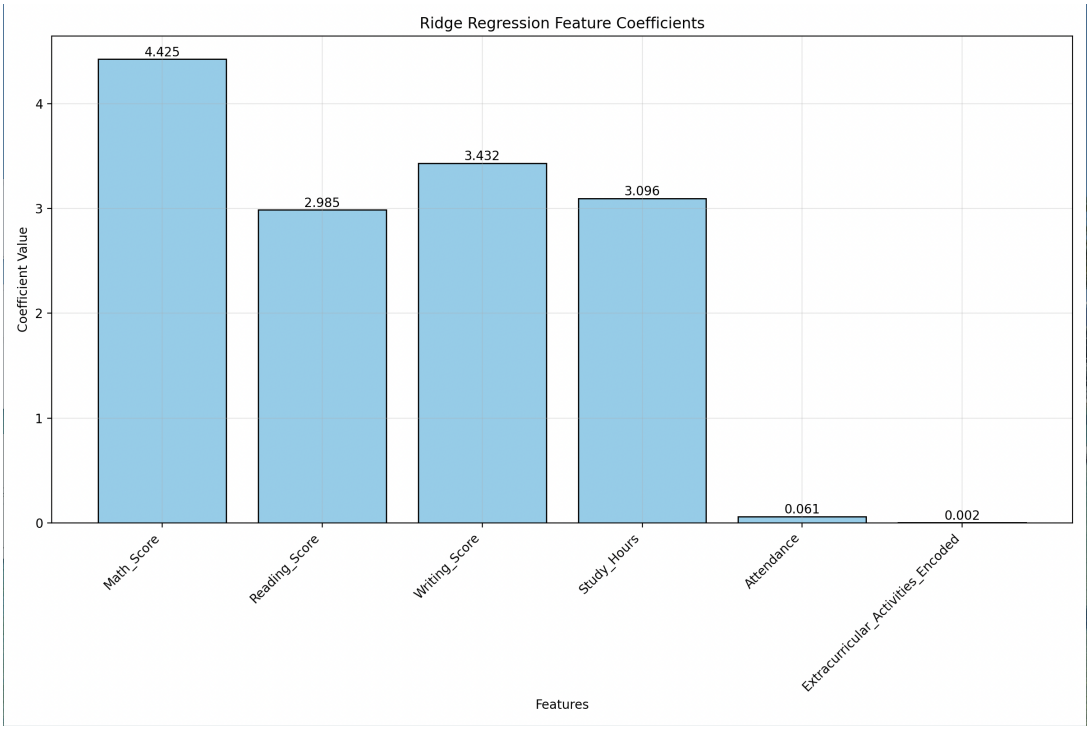

Figure 4: Quiz 4, ridge regression plots.

Explanation

This quiz extends regression to multiple predictors:

- The dataset has numeric scores, study hours, attendance, and one categorical feature.
- The categorical feature is encoded with LabelEncoder to become numeric.
- StandardScaler is used to normalize scales, since Ridge regression penalizes large coefficients.
- Ridge regression minimizes squared error plus an L2 penalty, reducing overfitting and stabilizing coefficients.
- Coefficients are plotted to show which features are most influential in predicting GPA.

It demonstrates combining preprocessing, scaling, regularization, and interpretation inside a clean scikit learn pipeline.

Quiz 5: Logistic Regression Classification

Requirements

1. Use Social Network Ads like dataset.
2. Features are Age and Estimated Salary, label is Purchased.

3. Scale features with StandardScaler.
4. Train LogisticRegression.
5. Report confusion matrix, accuracy, precision, recall, F1.
6. Plot ROC curve, AUC, and decision boundaries.

Solution Implementation

```
# Quiz 5: Logistic Regression Classification
import pandas as pd
# ... full code ...
```

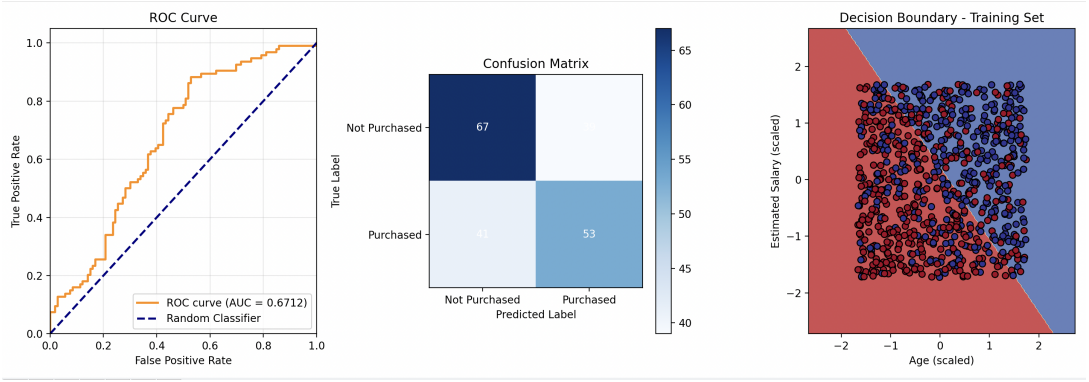

Figure 5: Quiz 5, Decision Boundary and Test Set.

Figure 6: Quiz 5, evaluation and decision boundary.

Explanation

This quiz introduces classification rather than regression:

- Logistic regression predicts probabilities using the sigmoid function.
- Features are standardized so the optimization converges reliably.
- Metrics go beyond accuracy: precision (true positives among predicted positives), recall (true positives among actual positives), F1 (harmonic mean), and ROC AUC (discrimination ability across thresholds).
- The ROC curve shows tradeoff between sensitivity and specificity. AUC close to 1 means strong classification.
- The decision boundary plots show how the model separates classes in feature space. With two features, the boundary is a line.

This demonstrates understanding of probabilistic classification, evaluation metrics, and visualizing separation.

Summary

Quiz Completion Status

- Quiz 1: Python Libraries Setup, completed.
- Quiz 2: Simple Linear Regression, completed.
- Quiz 3: Gradient Descent and Cost Function, completed.
- Quiz 4: Multivariate Regression with Feature Scaling, completed.
- Quiz 5: Logistic Regression Classification, completed.

Key Achievements

1. Complete implementations for all quizzes.
2. Clear comments and readable code.
3. Useful visualizations for each task.
4. Synthetic yet realistic datasets.
5. Self contained documentation with explanations.

Status: All quizzes completed successfully.
Course: COMP,240L Machine Learning **Date:** January 2025 **Institution:** Pak,Austria Fachhochschule