ELL-409

ASSIGNMENT 1: LINEAR REGRESSION AND LINEAR CLASSIFICATION

LINEAR CLASSIFICATION:

Question 1: Implementing Linear Classification Build a linear classification model using PyTorch for MNIST and CIFAR-10datasets. Use the following steps to guide your implementation:

1. Load the dataset: Load a dataset of your choice using PyTorch's data loading utilities (e.g., **torchvision.datasets**).
2. Data preprocessing: Normalize the dataset and split it into training and testing sets.
3. Model architecture: Define a linear classification model using PyTorch's **nn.Module**. Your model should consist of a linear layer followed by an appropriate activation function.
4. Loss function and optimizer: Choose a suitable loss function for classification tasks and an optimizer for gradient descent (e.g., CrossEntropyLoss and SGD).
5. Training loop: Implement a training loop that iterates over the training dataset for a specified number of epochs. For each iteration, calculate the forward pass, compute the loss, perform backpropagation, and update the model's parameters.
6. Evaluation: After training, evaluate your model on the testing dataset and calculate the accuracy.
7. Results and analysis: Provide the final accuracy of your model and discuss any observations or insights you have gained from the results.

Question 2: Hyperparameter Tuning and Regularization Extend the linear classification model from Question 1 to explore the impact of hyperparameters and regularization techniques. Follow these steps:

1. Implement learning rate schedules: Modify your training loop to include a learning rate scheduler (e.g., StepLR or ReduceLROnPlateau) and experiment with different learning rate decay strategies.
2. Introduce L2 regularization: Modify the model architecture to include L2 regularization (weight decay) on the linear layer. Train the model with different regularization strengths and observe its effect on overfitting and accuracy.
3. Experiment with batch sizes: Explore the effect of batch size on training convergence and computational efficiency. Train the model with different batch sizes and analyze the trade-offs.
4. Compare different optimizers: Choose two additional optimizers (e.g., Adam and RMSprop) and compare their performance with the previously used optimizer (SGD). Report the accuracy and training time for each optimizer.
5. Write a conclusion: Summarize your findings from the experiments. Which hyperparameters and regularization techniques seemed to have the most impact on the model's performance? What did you learn about the importance of careful tuning?

Remember to provide clear code implementations, comments, and explanations for each step in your solutions.

LINEAR REGRESSION:

Question 3: Problem Statement: Predicting Sales Revenue

Dataset Description:

You are provided with a dataset containing information about sales revenue in a retail business. The dataset includes the following columns:

X1: A numerical feature representing a sales-related metric.

X2:A binary categorical feature (0 or 1) representing a certain retail promotion.

X3: A categorical feature with three categories (A, B, or C) representing the store location.

y:The target variable, representing the sales revenue (numerical).

Problem Description:

Your task is to build a linear regression model to predict the sales revenue based on the provided features. The goal is to understand how different factors, such as the sales-related metric, promotion, and store location, influence sales revenue.

Tasks:

1. Data Preprocessing:

Load and preprocess the dataset, including encoding categorical variables if necessary.

2. Exploratory Data Analysis (EDA):

Perform exploratory data analysis to understand the distribution of features and correlations with the target variable.

3. Feature Selection:

Determine which independent variables (features) are most relevant for predicting sales revenue. You can use correlation coefficients or other methods for feature selection.

4. Linear Regression Model:

Build a linear regression model to predict sales revenue based on the selected features.

5. Model Evaluation:

Evaluate the model's performance using regression evaluation metrics such as Mean Absolute Error (MAE), Mean Squared Error (MSE), and R-squared.

6. Model Interpretation:

Interpret the coefficients of the linear regression model to understand how each feature influences sales revenue.

7. Predictions:

Use the trained model to make sales revenue predictions for new scenarios

Deliverables:

- Jupyter Notebook and a report documenting your data preprocessing, exploratory data analysis, model building, evaluation, and interpretation.
- Visualizations to support your analysis.
- A predictive model for estimating sales revenue based on the provided features.

This problem will test your ability to work with real-world data, perform data analysis, build and evaluate a linear regression model, and communicate your findings effectively.

Submission Deadline: 10th sept 2023 (10 pm) on Moodle.