Q1. [5 points]

You are tasked with developing a time series forecasting model for predicting energy consumption. Given the sequential nature of time series data, explain how you would apply boosting algorithms like AdaBoost and Gradient Boosting to this problem. Detail the process of handling residuals, updating weights, and fitting weak learners in the context of time-dependent data. What specific considerations would you take into account to ensure the models capture temporal dependencies effectively?

Answer:

Applying boosting algorithms:

- AdaBoost:
    i. Initialize Weights: Start by assigning equal weights to all instances in the training dataset.
    ii. Train Weak Classifiers: Iteratively train weak classifiers (usually simple models like decision stumps). In each iteration:
        a. Train a classifier using the weighted instances.
        b. Calculate the error of the classifier based on these weights.
        c. Assign more weight to instances that were misclassified, making it more likely for future classifiers to focus on these.
    iii. Calculate Classifier Weights: Assign a weight to each classifier based on its accuracy. More accurate classifiers have higher weights.
    iv. Combine Classifiers: Aggregate the decisions of all the classifiers, weighted by their respective accuracies, to make the final prediction
- Gradient Boosting:
    i. Initial Prediction: Start with a basic prediction for each instance. This could be something simple like the mean of the target values. This is your initial prediction for all instances.
    ii. Calculate Residuals: Compute the residuals for each instance. The residual is the difference between the actual target value and your current prediction. Essentially, it represents how much your model is off for each instance.
    iii. Train Model on Residuals: Train a new model, not on the original target, but on these residuals. The goal of this model is to predict these residuals based on the input features. This means that this model is trying to learn the errors of the previous model.
    iv. Update Predictions: Use the predictions from this new model (which are predictions of the residuals) to update your initial predictions. You adjust your existing predictions by adding the predicted residuals (usually scaled by a learning rate) to them. This gives you a new set of improved predictions.
    v. Recalculate Residuals: With your updated predictions, calculate new residuals. These new residuals are again the differences between the actual target values and your updated

predictions.

vi. Repeat: Train another model on these new residuals and use its output to update your predictions again. This cycle continues for a fixed number of iterations, or until the improvements become negligibly small.

vii. Final Model: The ensemble of all these sequentially trained models, each correcting the predecessor's errors, forms your final predictive model.

- to ensure the models capture temporal dependencies effectively
  Given that Boosting operates sequentially, it is crucial to respect the time order of the data. This means ensuring that models are trained on past data to predict future values, preventing any leakage of future information into the model.

Q2. [5 points]

You are tasked with developing a neural network model for a specialized image classification task with limited labeled data. Transfer learning is a viable approach, and you plan to use a pre-trained model as a starting point.

Question:

Explain the process of transfer learning in the context of deep learning. How would you select an appropriate pre-trained model and customize it for your task? Discuss the factors influencing your decisions on freezing layers, replacing the top layer, and training strategies.

Answer:

- Transfer learning
  is a machine learning technique where a model developed for a task is reused as the starting point for a model on a second task. It's especially powerful in the field of deep learning due to the amount of time, resources, and data required to train deep learning models from scratch.
- Selecting an Appropriate Pre-Trained Model
  The choice of pre-trained model can significantly impact performance. It's essential to select a model pre-trained on a sufficiently large and diverse dataset that has learned a broad range of features. The architecture of the model (e.g., ResNet, Inception, VGG) should also be appropriate for the complexity of the new task.
- Customizing Pre-trained Models
  - Replacing the Top Layer: The most common customization is replacing the top layer(s) of the model (which make the final classification decision) with new layers tailored to the new task. For instance, changing the output layer to match the number of classes in the new task.
  - Training Strategy: Deciding whether to freeze the weights of the pre-trained layers during training or allow them to update (fine-tuning). If the dataset for the new task is small, it's usually better to freeze most of the pre-trained layers to prevent overfitting.

    - Data Preprocessing: The input data should be preprocessed in the same way as the data used to train the pre-trained model. This ensures the features extracted by the pre-trained layers are relevant and meaningful for the new task.

Q3. [5 points]

You are tasked with detecting anomalies in network traffic data to identify potential security threats. The dataset consists of various features, such as packet size, source IP, destination IP, and time intervals between packets. Traditional clustering algorithms have not performed well due to the irregular and sparse nature of the anomalies.

Question:

Explain how you would implement DBSCAN to detect anomalies in this network traffic data. Detail the process of selecting appropriate values for ε (epsilon) and MinPts, how you would interpret the results to distinguish between normal traffic and anomalies, and the advantages of using DBSCAN over other clustering methods in this context. Additionally, discuss any preprocessing steps you would take to prepare the data for DBSCAN.

Answer:

- Selecting the right values for ε and MinPts is challenging and can significantly impact the clustering results.
    - Epsilon (ϵ): The maximum distance between two points to be considered neighbors.
    - MinPts: The minimum number of points required to form a dense region (core point).
- Algorithm Steps
    - Classification of Points: Each point is classified as a core, border, or noise point based on ε and MinPts.
        - Core Point: A point with at least MinPts neighbors within ϵ.
        - Border Point: A point that is not a core point but is within ϵ distance of a core point.
        - Noise Point: A point that is neither a core nor a border point
- distinguish between normal traffic and anomalies
    - consider all the points that are classified with Noise Point as anomalies
- Advantages
    - No Assumption on Cluster Shapes: Effectively handles clusters of arbitrary shapes and sizes.
    - Robustness to Outliers: Naturally identifies and ignores noise or outlier data points.
    - Minimal Input Parameters: Requires only two parameters (ε and MinPts), making it relatively easy to configure.

- Preprocessing steps
    - Dimensionality reduction: consider using PCA and t-SNE to reduce the dimensionality in the data to Eliminates redundant and irrelevant features, improving model performance

1. Which of the following statements about bagging and pasting is correct? **(A)**
   A) Bagging involves sampling with replacement, while pasting involves sampling without replacement.
   B) Bagging results in higher bias compared to pasting due to less diverse training subsets.
   C) Both bagging and pasting allow sampling the same training instance multiple times for the same predictor.
   D) Pasting typically results in higher model variance compared to bagging due to more correlated predictors.
2. In the context of out-of-bag (oob) evaluation used in bagging, which of the following statements is true? **(B)**
   A) Oob instances are the training instances that are used multiple times by a predictor during training.
   B) Oob instances are the training instances not seen by a given predictor during training, allowing for an internal validation set.
   C) Oob evaluation increases the overall bias of the model by excluding 37% of the training instances.
   D) Oob evaluation requires a separate validation set to evaluate the ensemble's performance accurately.
3. The Inception module is a key innovation of the GoogLeNet architecture. What is the primary purpose of the 1x1 convolutions in the Inception module? **(B)**
   A) To perform max pooling
   B) To reduce the computational cost and number of parameters
   C) To increase the dimensionality of the feature maps
   D) To replace ReLU activations
4. ResNet introduced residual learning to address a specific problem associated with training deep neural networks. What problem does residual learning primarily solve? **(B)**
   A) Overfitting on small datasets
   B) Vanishing gradient problem
   C) High computational cost
   D) Insufficient feature extraction
5. Which of the following is a key challenge associated with using hierarchical clustering for large datasets? **(C)**

A) Difficulty in handling non-linear relationships

B) Requirement to pre-specify the number of clusters

C) High computational complexity

D) Inability to detect outliers

6. What is the primary objective of unsupervised learning? **(B)**

A) To predict future outcomes based on historical data

B) To model the underlying structure or distribution in the data

C) To classify data points into predefined categories

D) To optimize a reward signal in a sequential decision-making process

7. In reinforcement learning, what is the main objective of the agent? **(C)**

A) To minimize the immediate reward

B) To maximize the immediate reward

C) To maximize the expected cumulative reward over time

D) To minimize the expected cumulative reward over time

8. Which of the following statements best describes a stochastic policy in reinforcement learning? **(B)**

A) It maps each state to a specific action with certainty

B) It provides a probability distribution over actions for each state

C) It always selects the action with the highest estimated value

D) It selects actions randomly, ignoring the state

9. Which of the following is a primary advantage of using Long Short-Term Memory (LSTM) networks over traditional Recurrent Neural Networks (RNNs)? **(C)**

A) LSTMs use a simpler architecture with fewer gates than RNNs.

B) LSTMs can handle sequences of arbitrary lengths without any issues.

C) LSTMs address the vanishing and exploding gradient problems, improving the learning of long-term dependencies.

D) LSTMs can process sequences in parallel, leading to more efficient training.

10. What is the main benefit of incorporating attention mechanisms in sequence-to-sequence (Seq2Seq) models? **(B)**

A) Attention mechanisms reduce the computational complexity of training Seq2Seq models.

B) Attention mechanisms allow the model to focus on different parts of the input sequence when generating each word of the output, improving performance on long sequences.

C) Attention mechanisms eliminate the need for an encoder-decoder architecture in Seq2Seq models.

D) Attention mechanisms ensure that the model can handle fixed-size input sequences more efficiently.