FLIP ROBO

ASSIGNMENT - 5

MACHINE LEARNING

Mitchelle Khanna | Date:20th July | Subjective| Deadline of Submission: 22nd July 2024

Q1 to Q15 are subjective answer type questions, Answer them briefly.

1. **R-squared or Residual Sum of Squares (RSS) which one of these two is a better measure ofgoodness of fit model in regression and why?**

Ans.

- R-squared tells you how well your model is performing in a way that's easy to interpret. A high R-squared means your model explains a lot of the data's variability, which is a good sign.
- RSS tells you the raw amount of error but isn't as helpful for comparing how well different models fit the data.

For example: Imagine you're trying to predict test scores based on study hours. If you get an R-squared of 0.85, you can proudly say that 85% of the differences in test scores are explained by the study hours. That's a clear and meaningful way to see if your model is on the right track.

On the other hand, just knowing the RSS value doesn't give you a complete picture. It's more useful for seeing how much error is there, but it doesn't tell you how well your model is actually performing compared to others.

In summary, R-squared is usually the better choice because it gives you a percentage of how well your model fits the data, which is easier to understand and compare across different models.

Quick Comparison Table:

Metric	What It Measures	What It Shows
R-squared	How much of the data's variation is explained by your model	A percentage that's easy to understand (higher is better)
RSS	Total amount of prediction error	Just a number, hard to compare or interpret on its own

2. **What are TSS (Total Sum of Squares), ESS (Explained Sum of Squares) and RSS (Residual Sumof Squares) in regression. Also mention the equation relating these three metrics with each other.**

Ans.

In the context of regression analysis, TSS, ESS, and RSS are key metrics used to evaluate the fit of the model:

1. **Total Sum of Squares (TSS):**
    - TSS measures the total variation in the dependent variable (Y). It represents the sum of the squared differences between each observed value and the mean of the observed values.
    - Formula: TSS=∑(Yi−Y¯)2TSS = \sum (Y_i - \overline{Y})^2TSS=∑(Yi−Y)2
2. **Explained Sum of Squares (ESS):**
    - ESS measures the variation explained by the regression model. It represents the sum of

the squared differences between the predicted values ($\hat{Y}$) and the mean of the observed values.
    - Formula: ESS=∑(Yi^−Y¯)2ESS = \sum (\hat{Y_i} - \overline{Y})^2ESS=∑(Yi^−Y)2

3. **Residual Sum of Squares (RSS):**
    - RSS measures the variation that the model fails to explain. It represents the sum of the squared differences between the observed values and the predicted values.
    - Formula: RSS=∑(Yi−Yi^)2RSS = \sum (Y_i - \hat{Y_i})^2RSS=∑(Yi−Yi^)2

These three metrics are related through the following equation: TSS=ESS+RSSTSS = ESS + RSSTSS=ESS+RSS

This relationship shows that the total variation in the data (TSS) is the sum of the variation explained by the model (ESS) and the variation that is not explained (RSS).

3. What is the need of regularization in machine learning?

Ans. Regularization in machine learning is needed to prevent overfitting. Overfitting happens when a model learns the noise in the training data instead of the actual patterns. Regularization techniques, like L1 and L2 regularization, add a penalty for larger coefficients in the model. This helps to keep the model simpler and more generalizable to new, unseen data, leading to better performance and accuracy in real-world applications.

4 What is Gini–impurity index?

Ans. The Gini impurity index is a measure used in decision trees to assess how often a randomly chosen element would be incorrectly labeled if it was randomly labeled according to the distribution of labels in the dataset. In simpler terms, it tells us how mixed the classes are in a node. A lower Gini impurity means a more homogenous node, which is what we aim for when splitting the data to make decisions.

5 Are unregularized decision-trees prone to overfitting? If yes, why?

Ans. Yes, unregularized decision trees are prone to overfitting because they can grow very deep and complex, capturing noise and specific patterns in the training data that don't generalize well to new, unseen data. This makes the model perform well on training data but poorly on test data.

6 What is an ensemble technique in machine learning?

Ans. An ensemble technique in machine learning combines multiple models to improve overall performance. By pooling the strengths of various models, it often achieves better accuracy and robustness compared to using a single model. It's like getting a second (or third) opinion to make a more reliable decision.

7 What is the difference between Bagging and Boosting techniques?

Ans. Bagging and Boosting are both ensemble techniques but they work differently:

- **Bagging** (Bootstrap Aggregating) involves training multiple models independently on different subsets of the data and then averaging their predictions. It helps reduce variance and prevent overfitting. Think of it as gathering multiple opinions to get a balanced view.

- **Boosting** builds models sequentially, where each new model focuses on correcting the errors of the previous ones. It combines their predictions to improve accuracy and reduce bias. It's like refining your approach with each new attempt to get closer to the right answer.

8 What is out-of-bag error in random forests?

Ans. Out-of-bag (OOB) error in random forests is an estimate of the model's performance using data that wasn't included in the training of a particular tree. Each tree is trained on a random subset of the data, and the samples not used for that tree are the "out-of-bag" samples. The OOB error gives a way to evaluate the model's accuracy without needing a separate validation set. It's like using the feedback from parts of your data you didn't directly use for training to check how well your model is doing.

9 What is K-fold cross-validation?

Ans. K-fold cross-validation is a technique to evaluate a model's performance more reliably. It involves splitting the data into KKK equally sized "folds." You train the model KKK times, each time using a different fold as the test set and the remaining K−1K-1K−1 folds for training. This process helps ensure that every data point is used for both training and testing, giving a more balanced view of the model's performance. It's like testing a recipe by sharing it with different people each time to get varied feedback.

10 What is hyper parameter tuning in machine learning and why it is done?

Ans. Hyper parameter tuning is all about fine-tuning the settings of your machine learning model to get the best performance. Unlike model parameters, which the model learns during training, hyper parameters are set before training begins. You adjust them to improve how well the model works, ensuring it makes accurate predictions and performs well on new data. Think of it as tweaking a recipe to get the perfect dish—finding the right balance can make a big difference in the final result.

11 What issues can occur if we have a large learning rate in Gradient Descent?

Ans. If you use a large learning rate in Gradient Descent, you might run into a few problems:

1. **Overshooting:** The updates to the model's weights can be too big, causing you to jump over the optimal solution instead of gradually getting closer to it.
2. **Divergence:** Instead of converging to a solution, the model's performance might get worse and worse because the updates are too aggressive.
3. **Instability:** The training process can become unstable, leading to erratic behavior in the loss function, which makes it hard to track progress.

It's like trying to solve a puzzle by making huge leaps—you're more likely to end up lost than finding the right pieces.

12 Can we use Logistic Regression for classification of Non-Linear Data? If not, why?

Ans. Logistic Regression on its own isn't great for classifying non-linear data. It works best with data where the relationship between the features and the outcome is linear. If your data has complex, non-linear patterns, Logistic Regression might struggle because it tries to fit a straight line (or hyperplane) to the data, which can miss these intricate patterns.
Think of it like trying to fit a straight stick into a curved space—it just doesn't match well. For non-linear data, you might need more flexible models like decision trees or neural networks that can handle curves and complex shapes better.

13 Differentiate between Adaboost and Gradient Boosting.

Ans. Adaboost and Gradient Boosting are both boosting techniques but they have some key differences:

- **Adaboost** (Adaptive Boosting) focuses on correcting the errors made by previous models by giving more weight to the misclassified data points. It builds a series of models where each one tries to fix the mistakes of the one before. It's like getting better at a skill by learning from past mistakes.
- **Gradient Boosting** builds models sequentially, where each new model corrects errors from the previous model but does so by minimizing a specific loss function. It's more flexible because it can handle different types of loss functions and learn complex patterns. It's like gradually improving your approach by focusing on specific areas that need work.

In short, Adaboost emphasizes correcting mistakes in a straightforward way, while Gradient Boosting optimizes the model more deeply and flexibly.

14 What is bias-variance trade off in machine learning?

Ans. The bias-variance trade-off is about finding the right balance between two types of errors in a machine learning model:

- **Bias:** Error due to overly simplistic assumptions in the model. High bias means the model is too simple and may not capture important patterns (underfitting).
- **Variance:** Error due to excessive complexity and sensitivity to small fluctuations in the training data. High variance means the model is too complex and may overfit the training data, capturing noise instead of patterns.

The trade-off is about balancing these errors. Too much bias leads to underfitting, where the model is too basic. Too much variance leads to overfitting, where the model is too complex. The goal is to find a sweet spot where both bias and variance are minimized, leading to a model that generalizes well to new data.

15 Give short description each of Linear, RBF, Polynomial kernels used in SVM.

Ans.

- **Linear Kernel:** Think of this as drawing a straight line to separate different groups in your data. It works well when the data can be divided by a simple line.

- **RBF (Radial Basis Function) Kernel:** This one is like bending and stretching the space to fit your data better. It's great for cases where the data is all jumbled up and needs more flexibility to be separated.

- **Polynomial Kernel:** Imagine fitting a curve or a squiggly line to your data. This kernel helps when the relationship between the features is more complex and needs something beyond just a straight line to capture the patterns.