STATISTICS WORKSHEET-5

Q1 to Q10 are MCQs with only one correct Choose the correct option.

1. Using a goodness of fit, we can assess whether a set of obtained frequencies differ from a set of frequencies.

d. Expected

2. Chi-square is used to analyse

C) Frequency

3. What is the mean of a Chi Square distribution with 6 degrees of freedom?

C) 6

4. Which of these distributions is used for a goodness of fit testing?

b) Chi-square distribution

5) Which of the following distributions is Continuous

C) F Distribution

6) A statement made about a population for testing purpose is called?

b) Hypothesis

7. If the assumed hypothesis is tested for rejection considering it to be true is called?

a) Null Hypothesis

8. If the Critical region is evenly distributed then the test is referred as?

a) Two tailed

9. Alternative Hypothesis is also called as?

b) Research Hypothesis

10) In a Binomial Distribution, the mean value is given by:

a) np

MACHINE LEARNING

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 of goodness of fit model in regression and why?

- **R-squared**: Standardized and ranges from 0 to 1, indicating the proportion of variance in the dependent variable explained by the model. Higher values mean a better fit, making it easy to interpret and compare across models.
- **RSS**: Measures the total variation not explained by the model but is not standardized and depends on the scale of the data, making it harder to interpret.

Conclusion: R-squared is preferred for its clarity and comparability in assessing model fit.

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

In regression analysis, the key metrics TSS, ESS, and RSS help evaluate how well a model fits the data:

1. **Total Sum of Squares (TSS)**: Represents the total variability in the dependent variable, showing how much the data points deviate from their overall mean.
2. **Explained Sum of Squares (ESS)**: Represents the portion of the total variability that the regression model explains. It shows how well the model captures the data's trend.
3. **Residual Sum of Squares (RSS)**: Represents the portion of the total variability that the model doesn't explain. It shows how much the observed data still deviates from the model's predictions.

Relationship:

These metrics are connected by the relationship:

- TSS = ESS + RSS

This means that the total variation in the data (TSS) is split into the part explained by the model (ESS) and the part that remains unexplained (RSS).

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

Regularization in machine learning is essential to prevent overfitting, which occurs when a model performs well on training data but poorly on new, unseen data. It does this by adding a penalty to the model's loss function, discouraging overly complex models that capture noise rather than the underlying patterns. This helps in controlling model complexity, improving generalization, and handling multicollinearity. Techniques like Lasso (L1) and Ridge (L2) regularization can also aid in feature selection and managing the bias-variance tradeoff, resulting in more robust and interpretable models.

4. What is Gini–impurity index?

The Gini impurity index measures the probability of misclassifying a randomly chosen element from a dataset. There are two types: **binary Gini impurity** for two classes and **multiclass Gini impurity** for more than two classes. The formula is $Gini=1-\sum pi^2$Gini=1−∑pi2 , where $p_i$pi is the proportion of each class in a node. Decision trees use Gini impurity to find splits that make nodes purer and improve classification accuracy.

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

Yes, unregularized decision trees are prone to overfitting because they can grow too deep, capturing noise and perfectly fitting the training data, which reduces their ability to generalize to new data.

6. What is an ensemble technique in machine learning?

An ensemble technique in machine learning combines multiple models to enhance accuracy and robustness. Key methods include bagging (e.g., Random Forest), boosting (e.g., XGBoost), and stacking, each of which aggregates predictions in different ways. By leveraging the strengths of various models, ensembles typically outperform individual models.

7. What is the difference between Bagging and Boosting techniques?

Bagging (Bootstrap Aggregating) reduces variance by training multiple models independently on random subsets of data and averaging their predictions. Boosting improves accuracy by sequentially training models where each one focuses on correcting errors made by previous models. Bagging uses parallel training and aggregates predictions, while boosting trains models sequentially and combines their results through weighted voting. Both techniques enhance model performance but differ in their approach to handling data and model errors.

8. What is out-of-bag error in random forests?
Out-of-bag (OOB) error is an estimate of the prediction error of a random forest model. It is calculated by evaluating each training sample on the trees for which it was not used in training (i.e., the "out-of-bag" samples). This method provides a built-in cross-validation estimate of model performance without requiring a separate validation set.

9. What is K-fold cross-validation?
K-fold cross-validation is a model evaluation technique where the dataset is divided into K equally-sized folds. The model is trained K times, each time using K-1 folds for training and the remaining fold for validation. This process helps to estimate model performance more reliably by averaging the results across all folds.

10. What is hyperparameter tuning in machine learning and why is it done?
Hyperparameter tuning involves adjusting the parameters that govern the training process of a machine learning model to optimize its performance. It is done to find the best combination of these parameters that minimizes error and improves the model's accuracy and generalization ability. Effective tuning can significantly enhance model performance.

11. What issues can occur if we have a large learning rate in Gradient Descent?
A large learning rate in Gradient Descent can cause the optimization process to overshoot the minimum of the loss function, leading to divergence or oscillation. This instability may prevent the algorithm from converging to the optimal solution or result in poor generalization. It often necessitates reducing the learning rate or using adaptive learning rate methods.

12. Can we use Logistic Regression for classification of Non-Linear Data? If not, why?
Logistic Regression is inherently a linear classifier, meaning it assumes a linear relationship between the features and the log-odds of the target variable. For non-linear data, it may not capture complex relationships effectively, potentially resulting in poor performance. However, it can be extended with polynomial features or kernel methods to handle non-linearity.

13. Differentiate between Adaboost and Gradient Boosting.
AdaBoost (Adaptive Boosting) adjusts the weight of incorrectly classified instances in each iteration, focusing on harder examples. Gradient Boosting, on the other hand, builds models sequentially by fitting each new model to the residual errors of the previous models. While AdaBoost emphasizes data weighting, Gradient Boosting emphasizes correcting errors through gradient descent.

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

The bias-variance trade-off refers to the balance between a model's ability to generalize well to unseen data (low bias) and its sensitivity to fluctuations in the training data (low variance). High bias can lead to underfitting, while high variance can lead to overfitting. Finding the optimal trade-off is crucial for achieving good model performance.

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

- **Linear Kernel:** Computes the similarity between data points as a linear dot product. It is effective for linearly separable data.
- **RBF (Radial Basis Function) Kernel:** Measures similarity based on the distance between data points, allowing the model to handle non-linear decision boundaries. It uses a Gaussian function to transform the data into a higher-dimensional space.
- **Polynomial Kernel:** Computes similarity based on polynomial functions of the input features, allowing for flexible decision boundaries. It can model interactions between features up to a specified degree.