Name: Syeda Saba Hussain Batch: DS2311
Internship#0523

Answers to PFA of Worksheet Set 2
Machine Learning

1)

R^2 gives a measure of the proportion of variability explained.
RSS gives a measure of the magnitude of the residuals or errors.

R-squared (R2) is generally considered a better measure of a regression model's goodness of fit than the Residual Sum of Squares (RSS). Here are some key reasons why:

a. R-squared measures the proportion of variance in the dependent variable that the independent variables can explain. Values range from 0 to 1, with higher values indicating more variance explained and thus better model fit. RSS does not have an intuitive interpretability.

b. R-squared can be meaningfully compared across different model specifications, even if they have a different number of parameters or predictors. RSS cannot be easily compared in this way as its scale and values will change dramatically based on the specific model.

c. R-squared is easy to interpret - it gives the proportion of response variable fluctuation that is explained by the model. RSS simply provides the sum of squares of residuals without an intuitive sense of what constitutes a good value.

e. R-squared is scale-invariant and accounts for degrees of freedom whereas RSS - it depends heavily on sample size, number of predictors, scale of the response variable etc.

While both measure model fit, R-squared gives an intuitive 0-100% measure of variance explained that allows standardized comparisons regardless of data or model specifics. This makes it a much better indicator of model goodness of fit than Residual Sum of Squares.

---

2) **Total Sum of Squares (TSS)**

- □ This measures the total variance in the response (y) variable.
- □ It is calculated by summing the squares of the deviations of each y value from the mean y.

Explained Sum of Squares (ESS)

- □ This measures the amount of variance in y that is explained by the regression model.
- □ It is calculated by summing the squares of the deviations of the predicted y values from the mean y.

Residual Sum of Squares (RSS)

- □ This measures the amount of variance in y that is NOT explained by the model.
- □ It is calculated by summing the squares of the residuals (the deviations of the actual y values from the predicted y values).

The equation relating the three sums of squares is: TSS = ESS + RSS

It means that the Total Sum of Squares is equal to the Explained Sum of Squares (the variance explained by the model) + the Residual Sum of Squares (the unexplained variance).

The relative size of the ESS and RSS indicates how well the model fits the data. A large ESS and a small RSS indicate a good model fit, explaining a large portion of the variance in the response variable.

---

3) Regularization is an important concept in machine learning that helps prevent overfitting and improves generalizability:

a) Overfitting is a key problem in machine learning where a model fits the training data very well, but fails to generalize to new unseen data. This happens when the model is too complex relative to the amount and noisiness of the training data.

b) Regularization addresses overfitting by introducing additional constraints and penalties into the optimization of a machine learning model. This restricts model complexity and makes the model favour simpler explanations.

c) Some examples of regularization include L1 and L2 regularization which add a penalty proportional to the sum of absolute values of parameters (L1) or sum of squares of parameters (L2) to the loss function. This forces many model parameters to shrink towards zero.

d) Regularization helps smooth the model, enhance generalizability, avoid extremes by discouraging large parameters, enable feature and parameter selection, and improve performance on unseen data.

e) The amount of regularization is a key hyperparameter that must be carefully tuned - too little could still cause overfitting whereas too much could lead to an overly simplistic model that underfits the training data.

---

4) The Gini impurity index is a metric used to measure how often a randomly chosen element from a set would be incorrectly labeled if it were randomly labeled according to the distribution of labels in the subset. It is commonly used as a splitting criterion when creating decision trees for classification tasks.

---

5) Yes, unregularized decision trees are very prone to overfitting. There are a few key reasons for this:

a) Decision trees can grow very deep and complex to perfectly fit the training data, capturing all the noise and outliers. Without regularization, the tree will keep splitting nodes to reduce impurity/error and will overfit.

b) They are non-parametric models, so their complexity is not intrinsically controlled or limited except by the data itself. Therefore decision trees can create arbitrarily complex trees to overfit noisy data.

c) Decision trees are high variance but low bias models, so they can learn very flexible boundaries but have less bias control. Their variance needs external control via regularization.

d) Each split node reduction in impurity is done to minimize a simple loss function like Gini impurity or entropy. But this greedy splitting criteria often overfits due to focusing too localally.

e) Full-grown overfit trees tend to have very low impurities at their leaf nodes, but perform poorly on test data. Without regularization, most of the patterns may be just noise or outliers rather than signal.

---

6) An ensemble technique in machine learning refers to combining multiple learning models together to produce better predictive performance compared to a single model. Some key points about ensemble methods:

a) They train multiple base models (e.g. decision trees, neural nets) independently on the same training set.

b) These base models are typically more simple/weak learners with high bias but low variance.
c) The predictions from all base models are then combined through techniques like voting, averaging, or other methods to produce the overall ensemble prediction.
d) Ensemble techniques include:

- Bootstrap Aggregating (Bagging)
- Boosting
- Stacking
- Bucket of Models
- Majority Voting Ensemble

e) Ensembles reduce variance without increasing bias. They average out the biases of individual models to reduce overall error and have better generalization ability.
f) They produce more accurate and robust predictions compared to any of the individual constituent models. The key is that the base models make independent errors which cancel out when predictions are aggregated.

---

7) The key differences between the ensemble techniques of Bagging and Boosting are:

a) Bagging trains base models in parallel using random subsets of the training data samples, while boosting trains base models sequentially putting more focus on misclassified examples after each iteration.
b) Bagging combines base model predictions by model averaging or voting, while boosting combines by additive weighting according to each base model's competence.
c) Bagging aims to reduce variance by using randomly drawn subsets to decorrelate the base models. In contrast, boosting aims to reduce both bias and variance by focusing on harder examples.
d) Bagging uses independent base model training, while boosting tweaks the data weights iteratively after each base model based on its performance.
e) Common bagging methods are Random Forests and Extra Trees, while common boosting methods are AdaBoost, Gradient Boosting, and XGBoost.
f) Bagging is generally less prone to overfitting compared to boosting since it averages over many models trained independently.

---

8) The out-of-bag (OOB) error is a method for getting an unbiased estimate of the test set error for a random forest model without needing to use a separate validation set. the OOB error is an internally calculated estimate of a random forest model's

generalization performance based on the cases left out of each tree's bootstrap sample during training. It is a very useful metric in RF model building.

---

9) K-fold cross-validation is a systematic and widely used resampling technique that partitions the dataset into K subsets, trains and evaluates the model on different subsets iteratively, and provides a comprehensive assessment of the model's generalization performance. By leveraging multiple validation sets and averaging performance metrics, K-fold cross-validation helps in building more reliable, robust, and generalizable machine learning models suitable for various applications and datasets.

---

10) Hyperparameter tuning is the process of selecting the optimal set of hyperparameters for a machine learning model to optimize its performance. Hyperparameters are parameters that are not learned from the data but are set before training the model. They govern the learning process, model complexity, and generalization capabilities of the algorithm. Examples of hyperparameters include learning rate, regularization strength, tree depth in decision trees, number of layers and neurons in neural networks, and kernel type and parameters in support vector machines.

---

11) Using a large learning rate in gradient descent and other optimization algorithms can introduce various issues, including divergence, overshooting, unstable dynamics, poor generalization, numerical instabilities, slow or no convergence, and sensitivity to initialization. To mitigate these issues and optimize the training process effectively, practitioners typically employ techniques like learning rate scheduling, adaptive learning rate methods (e.g., Adam, RMSprop), gradient clipping, regularization, and careful hyperparameter tuning to determine an appropriate learning rate and ensure stable, efficient, and reliable model training and convergence.

---

12)
Logistic regression can't effectively model non-linear relationships because:

- It assumes a linear decision boundary (a straight line or hyperplane).
- It's unable to capture complex patterns or interactions between features.

Using it on non-linear data can lead to:

- Poor performance and inaccurate predictions.
- Oversimplified understanding of the data.

---

13)
AdaBoost and Gradient Boosting are ensemble learning techniques that combine multiple weak learners to create strong predictive models, but they differ in their

underlying algorithms, training processes, objectives, and performance characteristics. AdaBoost focuses on correcting misclassifications by sequentially adjusting the data weights and emphasizing errors, while Gradient Boosting aims to minimize a predefined loss function by iteratively optimizing the model's predictions, reducing residuals, and creating a strong ensemble model capable of capturing complex relationships and patterns in the data. Depending on the task, data characteristics, and requirements, practitioners can choose between AdaBoost and Gradient Boosting and leverage their unique strengths, capabilities, and features to develop accurate, robust, and efficient machine learning models suitable for various applications and domains.

---

14) The bias-variance trade-off is a critical concept in supervised machine learning that describes the relationship between a model's complexity, flexibility, and generalization performance. By understanding and balancing bias and variance, practitioners can develop accurate, robust, and reliable predictive models that capture the underlying patterns and relationships in the data, avoid overfitting or underfitting, and achieve optimal performance on new, unseen data. Through careful model selection, tuning, evaluation, and validation, the bias-variance trade-off guides the development and optimization of machine learning models suitable for various applications, datasets, and requirements, ensuring effective and efficient solutions to real-world problems.

---

15)

1. Linear Kernel:

The Linear kernel is the simplest and most basic kernel used in SVMs. It computes the dot product between the input feature vectors in the original space, effectively constructing a linear decision boundary in the feature space.

Functionality: The Linear kernel is suitable for linearly separable or nearly linearly separable datasets where the classes can be effectively separated by a straight line or hyperplane in the original feature space.

2. Radial Basis Function (RBF) Kernel:

The Radial Basis Function (RBF) kernel, also known as the Gaussian kernel, transforms the input feature vectors into an infinite-dimensional space by computing the radial basis function of the Euclidean distance between the feature vectors.

Functionality: The RBF kernel is versatile and capable of capturing non-linear relationships and complex patterns in the data by creating non-linear decision boundaries. It is particularly effective for handling non-linearly separable datasets and achieving high-dimensional representation, allowing SVMs to model intricate relationships and achieve superior performance on various tasks.

3. Polynomial Kernel:
The Polynomial kernel transforms the input feature vectors into a higher-dimensional space using polynomial functions, enabling SVMs to capture non-linear relationships and construct polynomial decision boundaries.
Functionality: The Polynomial kernel is useful for handling non-linearly separable datasets and creating polynomial decision boundaries of varying degrees (e.g., linear, quadratic, cubic) to model complex relationships and patterns in the data. By adjusting the degree parameter, practitioners can control the complexity and flexibility of the decision boundary, ensuring optimal performance and generalization for specific applications and datasets.

Statistics Worksheet 5

1. d) expected
2. c) frequencies
3. c) 6
4. b) Chi-squared distribution
5. c) F Distribution
6. b) Hypothesis
7. a) Null Hypothesis
8. a) Two tailed
9. b) Research Hypothesis
10. a) np
