**THRISHA S**

**22CSR226**

**ASSIGNMENT 1:**

|  |  |  |  |
| --- | --- | --- | --- |
| STUDENT ID | STUDY HOURS | PRACTICE TESTS | EXAM SCORE |
| 1 | 5 | 2 | 80 |
| 2 | 3 | 1 | 65 |
| 3 | 7 | 3 | 90 |
| 4 | 4 | 2 | 75 |
| 5 | 6 | 3 | 85 |

**FEATURE:**

Individual measurable properties or characteristics used as inputs to the model

Eg: the features are study hours and practice tests

**LABEL:**

The output variable that the model aims to predict

Eg: the label is exam score

**PREDICTION:**

The output from the model after it has been trained

Eg: study hour and practice test will predict an exam score

**OUTLIER:**

A data point that differs significantly from other observations

Eg: if one student had an exam score of 100 while others are between 65-90 then this score will be considered an outlier

**TEST DATA:**

The portion of the data used to evaluate the models performance

Eg: students 4 and 5 can be used as test data to evaluate the model trained on the other students

**TRAINING DATA:**

The portion of the data used to train the model

Eg: Student 1,2,3 can be used as training data

**MODEL:**

A mathematical representation of the relationship between features and labels

Eg: a linear regression model that predicts exam scores based on study hours and practice tests

**VALIDATION DATA:**

A subset of the training data used to tune hyperparameters and avoid overfitting

Eg: using students 3s data to validate the model trained on students 1 and 2

**HYPERPARAMETER:**

Parameters whose values are set before the learning process begins

Eg: learning rate or the number of epochs

**EPOCH:**

One complete pass through the entire training dataset

Eg: if we iterate through all the students data once during training it counts as one epoch

**LOSS FUNCTION:**

A method to evaluate how well the models predictions match the actual data

Eg: mean squared error(mse) could be used to measure the difference between the predicted and actual exam scores

**LEARNING RATE:**

A hyperparameter that controls how much to change the model in response to the estimated error each time the model weights are updated

Eg: a learning rate of 0.01 means the model is updated slowly while a learning rate of 1 means the model is updated quickly

**OVERFITTING:**

When a model learns the training data too well including noise and outliers leading to poor performance on new data

Eg: model that perfectly predicts the training data but performs poorly on the test data

**UNDERFITTING:**

When a model is too simple to capture the underlying pattern in the data leading to poor performance on both training and new data

Eg: a model that performs poorly on both the training and test data

**REGULARIZATION:**

Techniques used to reduce overfitting by penalizing complex models

Eg: adding a penalty to the loss function for large coefficients in a linear regression model

**CROSS-VALIDATION:**

A technique to evaluate the models performance by dividing the data into several subsets and training/testing the model on different combinations of these subsets

Eg: 5-fold cross-validation involves splitting the data into 5 parts, training the model on 4 parts and testing on the remaining part, repeating this process 5 times

**FEATURE ENGINEERING:**

The process of creating new features or modifying existing ones to improve model performance

Eg: creating a new feature total study time by multiplying study hours with practice test

**DIMENSIONALITY REDUCTION:**

Techniques to reduce the number of features while retaining important information

Eg: principal component analysis(pca) to reduce study hours and practice test into a single combined feature

**BIAS:**

The error introduced by approximating a real-world problem which may be complex by a much simpler model

Eg: a high bias model may consistently predict exam scores far from the actual scores

**VARIANCE:**

The error introduced by the models sensitivity to small fluctuations in the training set

Eg: a high variance model may predict exam scores very accurately for training data but poorly for test data