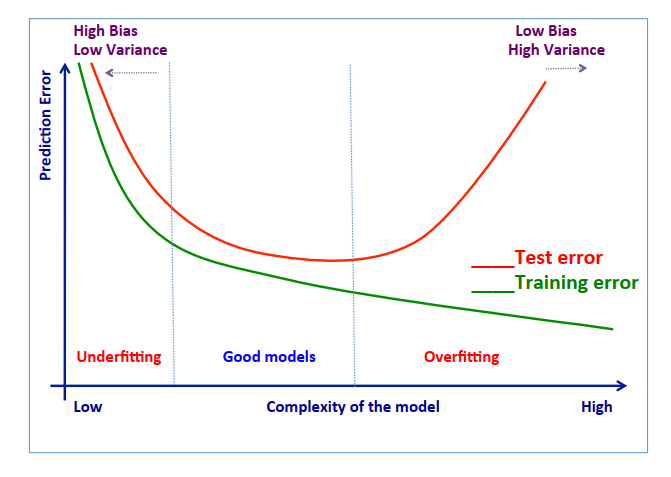
**AI Final Exam**

**Exam Date: 25-Feb-2024 9:00am**

**Submitted Date: 25-Feb-2024 12:00pm**

Theory (10pts)

1. Explain in detail what is Training Set, Validation Set, and Testing Set?
2. Regarding the figure below, what is Underfitting, Good Models, Overfitting?
3. In case of Overfitting existing, how to fix it and please explain in details?
4. Below Confusion Matrix, please calculate the value of these term and explain:

|  |  |  |  |
| --- | --- | --- | --- |
|  |  | Actual | |
|  |  | Orange | Banana |
| Prediction | Orange | 45 | 5 |
| Banana | 20 | 60 |

* 1. Accuracy
  2. Precision
  3. Recall
  4. specificity
  5. FI measure

**AI Final Exam**

**Exam Date: 25-Feb-2024 9:00am**

**Submitted Date: 25-Feb-2024 12:00pm**

NAME: THY SOMNANG

CLASS: 29/32 15E

**ANS**

* + - 1. **The Training Set** It is the set of data that is used to train and make the model learn the hidden features/patterns in the data.

The training set should have a diversified set of inputs so that the model is trained in all scenarios and can predict any unseen data sample that may appear in the future.

**The Validation Set**The validation set is a set of data, separate from the training set, that is used to validate our model performance during training**.**

The model is trained on the training set, and, simultaneously, the model evaluation is performed on the validation set after every epoch**.**

The main idea of splitting the dataset into a validation set is to prevent our model from overfitting i.e., the model becomes really good at classifying the samples in the training set but cannot generalize and make accurate classifications on the data it has not seen before.

**The Test Set**

The test set is a separate set of data used to test the model after completing the training.

Training data/validation/test

Evaluate model on validation Set

Train model

on Training Set

Confirm results

On Test Set

Pick model that does best on Validation Set

Tweak model according

To results Validation Set

* + - 1. Regarding the figure below, what is Underfitting, Good Models, Overfitting.

In the case of Unsupervised or Supervised machine learning, we are using a training set of data and a testing set of data. Data that are used to create the function model and data that are used to test if this model corresponds to the reality.

In order to know how good our model is we calculate Etrain.. The function that will calculate the number of error that our model f did on the training set of data.

Etrain(f) = Somme[de i=1 à n](loss(yi,f(xi)))

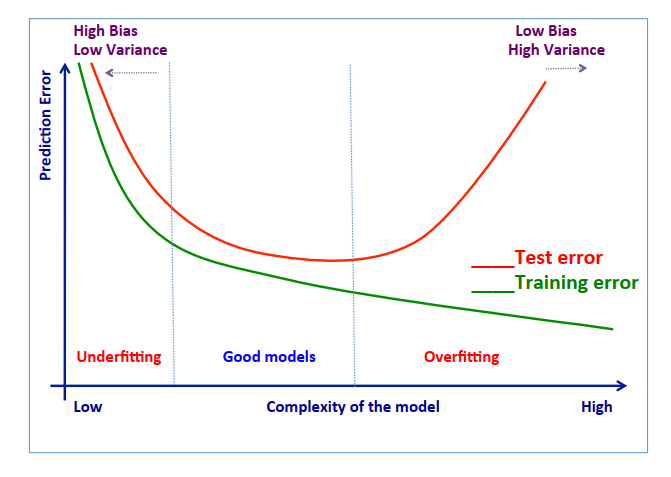
With for example the function loss(yi,f(xi)) = 1 (if sign(yi) is different from sign(f(xi)) or if not = 0

How can we minimize the function E. How can we be as close as possible to the reality?How to find the best function f?

if the model choose it’s too general or to specific we will not be able to minimize the function E.

For example, we want a function that recognize if a picture is a car or not. A function too general could be "picture with 4 wheels, superior to 1,4 meters tall". A truck would be recognize as a car. Or a function too specific: "picture with 4 wheels, 1m84 tall, red color…" would not recognize a white sportive car as a car.

We need to find the right balance between underfitting and overfitting model :



By at least having a model that work well on the training data set

Use simple model: reduce the number of features, do model selection, use regularization, do a cross-validation, avoiding high degree polynomials.

3. Overfitting is a common problem in machine learning where a model learns the training data too well, including noise and random fluctuations, to the extent that it performs poorly on unseen data.

Increase the Amount of Training Data:

One of the most effective ways to combat overfitting is to increase the size of the training dataset. With more data, the model is less likely to memorize noise and instead learns more generalizable patterns.

Cross-validation is a technique used to assess the generalization performance of a model. By splitting the data into multiple subsets (folds), training the model on some folds, and testing it on others, we can get a better estimate of how well the model will perform on unseen data. This helps in detecting overfitting and tuning hyperparameters accordingly.

Feature Selection:

Overfitting can occur when the model learns from irrelevant or redundant features in the training data. Feature selection techniques such as forward selection, backward elimination, or regularization methods (like L1 or L2 regularization) can help in selecting the most informative features and reducing overfitting.

Regularization:

Regularization methods add a penalty term to the loss function during training, discouraging the model from fitting the training data too closely.

L1 Regularization (Lasso): Adds the absolute value of the magnitude of coefficients as penalty term.

L2 Regularization (Ridge): Adds the square of the magnitude of coefficients as penalty term.

Elastic Net Regularization: A combination of L1 and L2 regularization.

By tuning the regularization parameter (lambda), you can control the amount of regularization applied to the model.

1. Below Confusion Matrix, please calculate the value of these term and explain:

|  |  |  |  |
| --- | --- | --- | --- |
|  |  | Actual | |
|  |  | Orange | Banana |
| Prediction | Orange | 45 | 5 |
| Banana | 20 | 60 |

a. Accuracy: Accuracy measures the overall correctness of the model.

Accuracy

=

True Positives

+

True Negatives

Total Population

Accuracy=

Total Population

True Positives + True Negatives

Accuracy

=

45

+

60

45

+

5

+

20

+

60

=

105

130

≈

0.8077

Accuracy=

45+5+20+60

45+60

=

130

105

≈0.8077

b. Precision: Precision measures the accuracy of the positive predictions.

Precision

=

True Positives

True Positives

+

False Positives

Precision=

True Positives + False Positives

True Positives

Precision

=

45

45

+

20

=

45

65

≈

0.6923

Precision=

45+20

45

=

65

45

≈0.6923

c. Recall: Recall measures how many of the actual positives our model captures through labeling it as positive.

Recall

=

True Positives

True Positives

+

False Negatives

Recall=

True Positives + False Negatives

True Positives

Recall

=

45

45

+

5

=

45

50

=

0.9

Recall=

45+5

45

=

50

45

=0.9

d. Specificity: Specificity measures how many of the actual negatives our model captures by labeling them as negative.

Specificity

=

True Negatives

True Negatives

+

False Positives

Specificity=

True Negatives + False Positives

True Negatives

Specificity

=

60

60

+

5

=

60

65

≈

0.9231

Specificity=

60+5

60

=

65

60

≈0.9231

e. F1 measure: The F1 score conveys the balance between the precision and the recall of the model.

F1 Score

=

2

×

Precision

×

Recall

Precision

+

Recall

F1 Score=2×

Precision + Recall

Precision × Recall

e. F1 measure: The F1 score conveys the balance between the precision and the recall of the model.

F1 Score

=

2

×

Precision

×

Recall

Precision

+

Recall

F1 Score=2×

Precision + Recall

Precision × Recall

These metrics provide a comprehensive understanding of the model's performance in terms of its predictive capabilities.