Supervised Learning - I

By:
Dr. Shikha Deep

First Step in Supervised Learning:

- A simple and widely used approach to estimate a model's performance is to split the available data into two parts: a training set, used to learn the model, and a test set, used to evaluate how well the model performs on unseen examples.
- We give the model lots of example problems where the answers are already known. The model looks for patterns that map the inputs (features) to the correct outputs (labels). Once learned, we hand it new problems and ask it to predict the answers.

Input (X), Output (y), Model, Prediction :

- **Input (X):** the information we feed to the model; features like age, height, pixel values, words, etc.
- **Output (y):** the label or target we want to predict; classes like “spam” or “not spam”, “disease / no disease”.
- **Model:** the method or rule (for example: logistic regression, SVM, decision tree) that learns to turn X → y.

```
┌──────────────────┐
│ Full Dataset │ (X, y)
└──────────────────┘
     │
┌────────┴────────┐
│                │
┌────────────────┐   ┌────────────────┐
│ Training set │    │ Test set │
│ (X_train,y_train)│  │ (X_test,y_test)│
└────────────────┘   └────────────────┘
  │
  ▼
┌────────────────┐
│ Train model │ (learn parameters)
└────────────────┘
  │
  ▼
┌────────────────┐
│ Trained model │
└────────────────┘
  │
  ▼
┌────────────────┐
│ Predict on X_test│ → compare ŷ_test vs y_test
└────────────────┘
```

Training Set :

- Used to **teach** the model.
- The algorithm looks at X_train (inputs) and y_train (correct answers).
- It learns the rules, patterns, or boundaries that connect X → y.
- Analogy: These are the **practice questions** for a student.

Testing Set :

- This portion is kept hidden during training.
- After the model has learned from the training data, we test it on X_test.
- We compare its predictions ŷ_test with the real answers y_test.
- Analogy: This is the **final exam** for the student.

Generalization

Generalization refers to the ability of a learning method to perform well on new, previously unseen data.

- When we train a model, we want it to learn patterns from the training data, not just memorize it.If the model can apply what it learned to new, unseen data, we say it has good generalization.
- A model with good generalization works well both on training data and on unseen test data.
- A model with poor generalization works well only on training data but fails on test data.

Overfitting and Underfitting

Overfitting occurs when a statistical model describes random error or noise instead of the underlying relationship. It performs well on the training data but poorly on new, unseen data.

- Overfitting happens when the model becomes too smart for its own good — it memorizes the training data instead of learning the actual pattern.
- On the training data, accuracy is very high (even close to 100%).But when you give it new test data, the accuracy drops badly.
- Overfitting = "Remembering answers, not understanding the concept."

Overfitting - memorization without understanding.

Signs of Overfitting

- Training accuracy is very high (e.g., 99%).
- Test accuracy is much lower (e.g., 60%).
- Model is unnecessarily complex.
- Predictions on new data are unreliable.

How to Prevent Overfitting

- Use more training data.
- Simplify the model (reduce complexity).
- Use techniques like **regularization** (we'll see later).
- Use **cross-validation** for tuning.

Underfitting occurs when a model is too simple to capture the underlying structure of the data, leading to poor performance both on the training set and unseen data.

- Underfitting happens when the model is too weak or too simple to understand the data properly.
- It cannot even perform well on the training data.
- Naturally, its performance on test data is also poor.
- Underfitting = "Not learning enough."

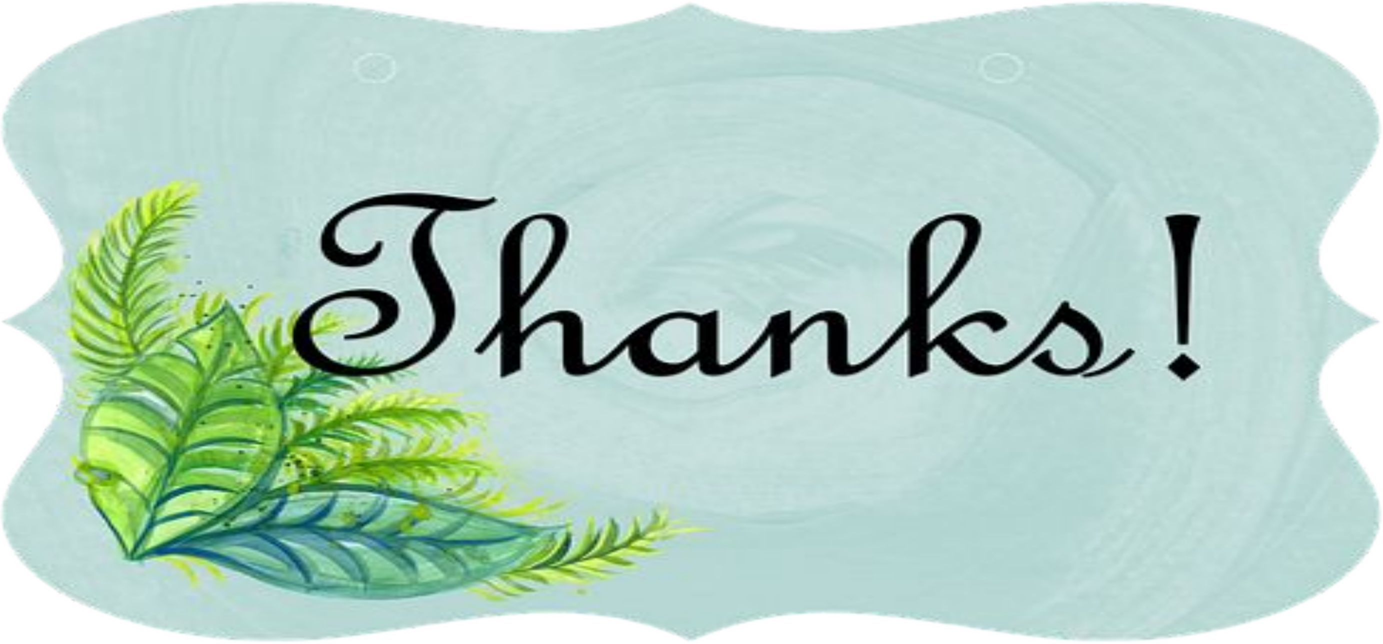
Underfitting — lack of preparation/effort.

Signs of Underfitting

- Training accuracy is low.
- Test accuracy is also low.
- Model is too simplistic (e.g., fitting a straight line to data that is curved).
- High **bias** (systematic error).

How to Fix Underfitting

- Use a more complex model.
- Add more relevant features.
- Reduce data preprocessing that oversimplifies patterns.
- Train the model longer (more epochs in deep learning).


Thanks!