Machine Learning

Video 86:

Voting Ensemble | Classification | Voting Classifier | Hard Voting Vs Soft Voting | Part 2

Hard Voting vs. Soft Voting in Ensemble Learning

Both **hard voting** and **soft voting** are techniques used in ensemble learning, particularly in **Voting Classifiers**, where multiple models are combined to make a final prediction.

1. Hard Voting (Majority Voting)

How It Works

- Each classifier in the ensemble makes its own prediction.
- The final prediction is determined by **majority vote** (the class with the most votes wins).
- It does **not** consider probability estimates from individual classifiers—just the raw predictions.

Example:

Suppose we have 3 classifiers predicting a class:

- Model 1 → **Class A**
- Model 2 → **Class B**
- Model 3 → **Class A**

Since **Class A** gets 2 votes and **Class B** gets 1 vote, the final prediction is **Class A**.

Pros & Cons

✅ Simple and robust if all classifiers are well-tuned.
❌ Ignores prediction confidence (e.g., if one model is 90% sure but loses to two models that are 51% sure, it still loses).

2. Soft Voting (Weighted Voting)

How It Works

- Each classifier outputs a probability distribution over classes.
- The final prediction is based on the **average of predicted probabilities** for each class.
- The class with the **highest summed probability** is chosen.

Example:

Suppose we have 3 classifiers predicting probabilities for two classes (A and B):

Model	Probability of A	Probability of B
Model 1	0.60	0.40
Model 2	0.30	0.70

Model	Probability of A	Probability of B
Model 3	0.80	0.20

Average probabilities:

- **Class A**: (0.60 + 0.30 + 0.80) / 3 = **0.566**
- **Class B**: (0.40 + 0.70 + 0.20) / 3 = **0.433**

Since **Class A** has a higher probability, the final prediction is **Class A**.

Pros & Cons

✅ Takes into account model confidence levels.
✅ Works better when models output well-calibrated probabilities.
❌ Requires classifiers that support probability predictions (e.g., Logistic Regression, Random Forest, but not SVM by default).

Key Differences

Feature	Hard Voting	Soft Voting
Decision Based On	Majority class votes	Average probability scores
Uses Probability?	❌ No	✅ Yes
Works with Non-Probabilistic Models?	✅ Yes	❌ No (requires probabilistic output)
Sensitive to Model Confidence?	❌ No	✅ Yes
Best for	Diverse classifiers with similar accuracy	Models with well-calibrated probability estimates

When to Use What?

- **Hard Voting** → When classifiers have similar accuracy and confidence levels.
- **Soft Voting** → When classifiers provide reliable probability estimates and vary in confidence.

Example:

Code link:

https://colab.research.google.com/drive/1fWH2tTh8q7O_XiauQG8jzzcR0FSfS1Oj?usp=sharing

Video 87:

Voting Ensemble | Regression | Part 3

Example:

Code link:
https://colab.research.google.com/drive/1fWH2tTh8q7O_XiauQG8jzzcR0FSfS1Oj?usp=sharing

Video 88:

Bagging | Introduction | Part 1

Bagging (Bootstrap Aggregating) is an ensemble learning technique in machine learning that improves model accuracy by combining multiple models trained on different subsets of the data. It reduces variance and prevents overfitting by averaging predictions (for regression) or using majority voting (for classification). Random Forest is a popular bagging algorithm.

Bootstrapping is a statistical resampling technique used to estimate the distribution of a dataset by repeatedly sampling with replacement. In machine learning, it's used in bagging to create multiple training sets from the original data, helping to improve model stability and reduce overfitting by introducing diversity in training samples.

Aggregation in machine learning refers to the process of combining predictions from multiple models to improve accuracy and robustness. In ensemble methods like bagging, aggregation is done by averaging predictions (for regression) or majority voting (for classification). This helps reduce variance, minimize overfitting, and enhance overall model performance.

Example:

Code link:

https://github.com/campusx-official/bagging-ensemble

Video 89:

Bagging Ensemble | Part 2

Example:

Code Link:

https://github.com/campusx-official/bagging-ensemble/blob/main/bagging_demo.ipynb

Video 90:

Bagging Ensemble | Part 3