Advanced Learning and Ensemble Techniques

Mr. Mehul A. Jadhav
Department of Computer Studies (MCA)
Vivekanand College, Kolhapur

17 July 2025

Agenda

Introduction to Ensemble Learning

Concept: Combines multiple models to improve predictive performance and reduce overfitting.

Types:

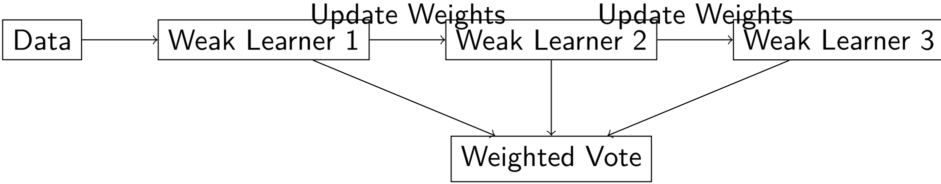
- Bagging (e.g., Random Forest)
- Boosting (e.g., AdaBoost, Gradient Boosting)

AdaBoost

Adaptive Boosting: Sequentially trains weak learners, assigning higher weights to misclassified instances.

Applications:

- Face detection
- Binary classification tasks



Gradient Boosting

Concept: Builds models sequentially, minimizing a loss function (e.g., mean squared error) using gradient descent.

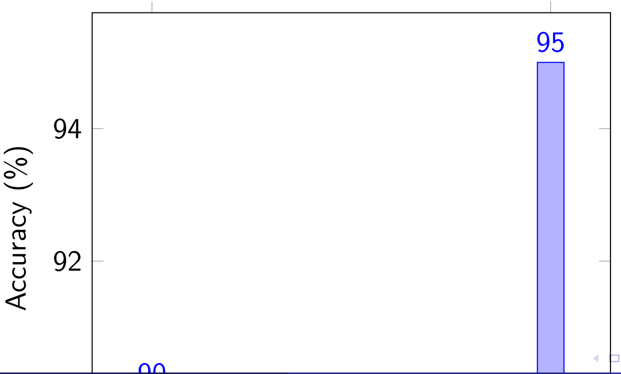
Applications:

- Ranking systems
- Regression tasks

Extreme Gradient Boosting: Optimized gradient boosting with regularization and parallel processing.

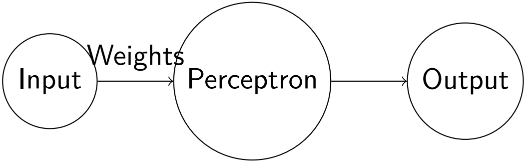
Advantages:

- High performance
- Handles missing data



Perceptron

Concept: Basic neural network unit, computes weighted sum and applies activation function.

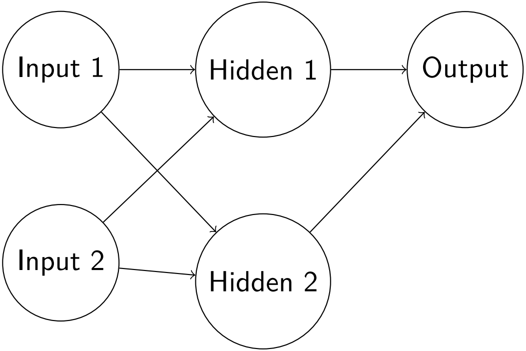


Multilayer Perceptron (MLP)

Concept: Deep neural network with multiple layers of neurons.

Applications:

- Image recognition
- Natural language processing



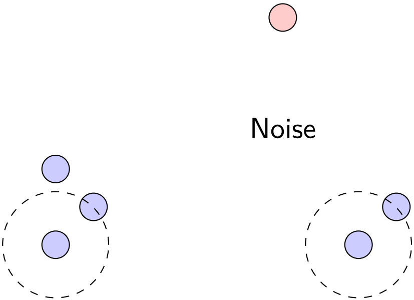
Density-Based Spatial Clustering of Applications with Noise:

- Clusters based on density of data points.
- Identifies outliers as noise.

Applications:

- Anomaly detection
- Spatial data analysis

DBSCAN Visualization



Dense regions form clusters; outliers are noise.

t-SNE

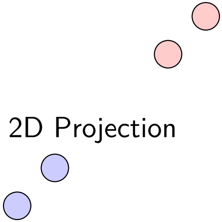
t-Distributed Stochastic Neighbor Embedding:

- Non-linear dimensionality reduction for visualization.
- Preserves local structure of high-dimensional data.

Applications:

- Visualizing high-dimensional datasets
- Feature embedding

t-SNE Visualization

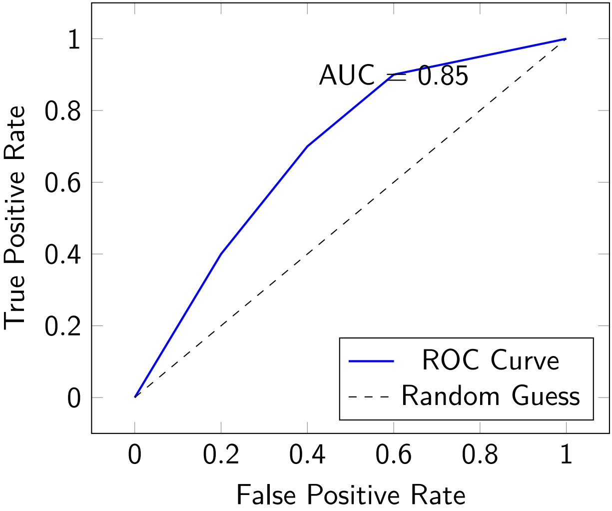


High-dimensional data projected to 2D space.

Model Evaluation Metrics

- **Accuracy:** Fraction of correct predictions.
- **Precision:** True positives / (True positives + False positives).
- **Recall:** True positives / (True positives + False negatives).
- **F1-Score:** Harmonic mean of precision and recall.
- **ROC-AUC:** Area under the Receiver Operating Characteristic curve.

ROC Curve



AUC = 0.85
ROC Curve
Random Guess
True Positive Rate
False Positive Rate
0
0.2
0.4
0.6
0.8
1

AdaBoost Implementation

Implementation (Python): `sklearn.ensemble.AdaBoostClassifier`

Key Parameters:

- `n_estimators`: Number of weak learners.
- `learning_rate`: Weight adjustment rate.

Use Cases: Classification tasks, boosting weak models.

Gradient Boosting Implementation

Implementation (Python):
`sklearn.ensemble.GradientBoostingClassifier`

Key Parameters:

- `n_estimators`: Number of boosting stages.
- `max_depth`: Maximum tree depth.

Use Cases: Predictive modeling, ranking.

DBSCAN Implementation

Implementation (Python): `sklearn.cluster.DBSCAN`

Key Parameters:

- `eps`: Maximum distance between points in a cluster.
- `min_samples`: Minimum points to form a cluster.

Use Cases: Outlier detection, spatial clustering.

Multilayer Perceptron (MLP) Implementation

Implementation (Python): `sklearn.neural`$_n$*etwork.MLPClassifier*

Key Parameters:

- `hidden_layer_sizes`: Number of neurons in each layer.
- `activation`: Activation function (e.g., relu).

Use Cases: Image classification, NLP.

t-SNE Implementation

Implementation (Python): `sklearn.manifold.TSNE`

Key Parameters:

- `n_components`: Dimensions of the embedded space.
- `perplexity`: Balance between local and global structure.

Use Cases: Data visualization, feature analysis.

Conclusion

Advanced techniques like ensemble learning, neural networks, DBSCAN, and t-SNE enhance predictive and analytical capabilities.

Model evaluation metrics ensure robust performance assessment.

References

- Scikit-learn Documentation: Ensemble Methods and Clustering
- GeeksforGeeks: Neural Networks and t-SNE
- Towards Data Science: Model Evaluation Metrics