Datasets Link : https://github.com/vijay-2155/Dm-lab

✅ EXPERIMENT 1 — Information Gain & Attribute Selection

What is Information Gain?

Information Gain (IG) measures how much a feature reduces the uncertainty (entropy) of the class.
Higher IG → more useful attribute.

Where is it used?

✔ Decision Tree algorithms (ID3, C4.5)
✔ Feature selection
✔ Selecting the best splitting attribute

Steps in WEKA

1. Open **Explorer** → **Preprocess**
2. Load dataset
3. Set **Class** attribute
4. Go to **Select Attributes**
5. Choose
   - Evaluator → **InfoGainAttributeEval**
   - Search → **Ranker**
6. Click **Start**
7. Read ranking to identify best attribute

✅ EXPERIMENT 2 — J48 Decision Tree Classification

What is J48?

J48 is WEKA’s implementation of C4.5 decision tree.
It splits data using Gain Ratio and creates human-readable rules.

Where is it used?

✔ Classification problems
✔ Interpretable models
✔ Medical & Financial decisions

Steps in WEKA

1. Explorer → Preprocess → load dataset
2. Set **Class attribute**
3. Go to **Classify** tab
4. Select **trees** → **J48**
5. Use **10-fold cross-validation**
6. Click **Start**
7. View decision tree and accuracy

✅ EXPERIMENT 3 — ID3 Decision Tree

What is ID3?

ID3 uses **Information Gain** to split nodes and works only on **categorical attributes**.

Where is it used?

✔ Introductory ML
✔ Decision support systems
✔ Educational datasets

Steps in WEKA

1. Install ID3:
    - Tools → Package Manager → install "simpleEducationalLearningSchemes"
2. Load dataset with **nominal attributes**
3. Set class
4. Classify → choose **ID3**
5. Click **Start**
6. View tree and rules

✅ EXPERIMENT 4 — k-Nearest Neighbor (k-NN / IBk)

What is k-NN?

A lazy classifier that predicts class based on the **k most similar instances**.

Where is it used?

✔ Pattern recognition
✔ Student performance
✔ Recommender systems

Steps in WEKA

1. Open dataset
2. Set class
3. Classify → choose **lazy** → **IBk**
4. Set **k = 1 or 3**
5. Run **10-fold cross-validation**
6. To classify a new record:
    - Test options → **Supplied test set**
    - Load test file
7. Click **Start**

✅ EXPERIMENT 5 — Naive Bayes Classification

What is Naive Bayes?

A probabilistic classifier based on **Bayes' Theorem** that assumes **independence** of features.

Where is it used?

✔ Text classification
✔ Spam filtering
✔ Medical diagnosis

Steps in WEKA

1. Load dataset
2. Set class

3. Classify → choose **bayes** → **NaiveBayes**
4. Enable **Output predictions** to view probabilities
5. Click **Start**

✅ EXPERIMENT 6 — Feature Selection on Iris Dataset

What is Feature Selection?

Selecting the most important attributes to improve model accuracy and reduce complexity.

Where is it used?

✔ Classification tasks
✔ Dimensionality reduction
✔ Performance optimization

Steps in WEKA

1. Preprocess → load dataset
2. Go to **Select Attributes**
3. Choose:
   - Evaluator → **InfoGainAttributeEval** or **GainRatioAttributeEval**
   - Search → **Ranker**
4. Click **Start**
5. Note top-ranked features

✅ EXPERIMENT 7 — Data Pre-processing (Customer Dataset)

What is Pre-processing?

Cleaning and transforming data before modeling.

Where used?

✔ All ML pipelines
✔ Before ANN, k-NN, SVM
✔ Handling missing values and scaling

Steps in WEKA

1. Load dataset
2. Apply filters:
    - **ReplaceMissingValues**
    - **Normalize** (0–1 scaling)
    - **Standardize** (Z-score)
    - **Discretize** (bin numeric to categorical)
    - **Remove** unwanted attributes
    - **NominalToBinary** for encoding
3. Save cleaned dataset

✅ EXPERIMENT 8 — Pre-processing on Iris Dataset

Concept

Same as Experiment 7, but Iris dataset has:

- Numeric attributes
- 1 categorical label

Steps in WEKA

1. Replace missing values
2. Normalize numerical attributes
3. Convert nominal (Species) into binary → **NominalToBinary**
4. Save processed dataset

✅ EXPERIMENT 9 — Backpropagation Neural Network (MLP)

What is Backpropagation?

A training algorithm for neural networks that:

- Propagates error backward
- Updates weights

- Minimizes loss

Where used?

✔ Deep learning
✔ Predicting performance
✔ Non-linear modeling

Steps in WEKA

1. Load dataset
2. Set **Class = Output attribute**
3. Classify → choose:

   `functions → MultilayerPerceptron`

4. Set:
   - HiddenLayers = a
   - Learning rate, momentum, epochs
5. Click **Start**
6. Analyze accuracy & confusion matrix

✅ EXPERIMENT 10 — k-Means Clustering

What is k-Means?

An unsupervised algorithm that groups data into **k clusters** based on similarity.

Where used?

✔ Customer segmentation
✔ Pattern grouping
✔ Marketing analytics

Steps in WEKA

1. Load dataset
2. Go to **Cluster** tab
3. Choose **SimpleKMeans**
4. Set:
   - k = desired number of clusters
   - Distance = Euclidean
5. Click **Start**

6. View centroids and cluster assignments

✅ EXPERIMENT 11 — Apriori Association Rule Mining

What is Apriori?

Algorithm to find:

- Frequent itemsets
- Association rules such as *Milk* → *Bread*

Where used?

✔ Market basket analysis
✔ Retail
✔ E-commerce recommendations

Steps in WEKA

1. Load dataset with Yes/No items
2. Go to **Associate**
3. Choose **Apriori**
4. Set:
    - Min support
    - Min confidence
5. Click **Start**
6. View frequent itemsets & rules

✅ EXPERIMENT 12 — FP-Growth

What is FP-Growth?

A faster alternative to Apriori using:

- FP-tree
- Frequent pattern compression

Where used?

✔ Very large transaction datasets
✔ Retail analytics
✔ Web usage mining

Steps in WEKA

1. Load transactional dataset
2. Associate → choose **FPGrowth**
3. Set:
    - minSupport
    - numRules
4. Click **Start**
5. Analyze extracted rules

✅ EXPERIMENT 13 — Compare J48, k-NN, Naive Bayes

What is Model Comparison?

Evaluating multiple classifiers using:

- Accuracy
- Precision
- Recall
- Confusion matrix

Why do it?

✔ Identify best algorithm
✔ Understand data behavior
✔ Improve prediction quality

Steps in WEKA

1. Load dataset
2. For each classifier:
    - Classify → choose
        - **J48**
        - **IBk** (k-NN)

- **NaiveBayes**

   - Use **10-fold cross-validation**
   - Record:
     - Accuracy
     - Precision
     - Recall
     - Confusion matrix

3. Compare results in a table
4. Write conclusion on best algorithm
