Data Warehousing & Data Mining – Practical Lab Manual with Solutions

Practical 1: Visualization Features in WEKA

Solution:

1. Open WEKA → Explorer → Open File → Select dataset (e.g., iris.arff).

2. Go to the "Visualize" tab.

3. Scatter plot matrix appears showing relationships between attributes.

4. Identify patterns:

- Iris-setosa is clearly separable in PetalLength vs PetalWidth.

- Iris-versicolor and virginica overlap slightly.

Conclusion: Visualization helps detect separability and outliers.

Practical 2: Data Preprocessing & Association Rule Mining

Solution:

1. Load dataset → Preprocess tab.

2. Apply filters:

- Remove → delete specific attributes.

- Normalize → scale attributes.

3. To perform association mining:

- Associate tab → Choose Apriori.

- Set minimum support = 0.1, confidence = 0.9.

4. Click Start.

Sample Output Rule:

petalwidth < 0.3 ⇒ class=setosa (confidence 1.0)

Conclusion: Apriori finds strong rules that describe attribute relationships.

Practical 3: Classification on Datasets

Solution:

1. Load dataset → Classify tab.

2. Choose classifier → J48 (Decision Tree).

3. Test Options:

- Use Training Set

- Percentage split = 70%

4. Start.

Sample Result:

Accuracy: 94%

Confusion Matrix:

a b c

a 50 0 0

b 2 44 4

c 0 3 47

Conclusion: J48 performs strongly on structured datasets like Iris.

Practical 4: Clustering on Datasets

Solution:

1. Load dataset → Cluster tab.

2. Choose SimpleKMeans → set k = 3.

3. Start.

Output:

Cluster 0: mostly setosa

Cluster 1: mixed virginca/versicolor

Cluster 2: mixed virginca/versicolor

Conclusion: K-means clusters based on attribute similarity but is unsupervised; labels are not used.

Practical 5: German Credit Data

Solution:

1. Load german_credit.arff → Preprocess.

2. Handle missing values (ReplaceMissingValues filter).

3. Classify → Choose NaiveBayes or J48.

4. Start.

Result example:

Accuracy: 72%

Conclusion: German Credit dataset is noisy and complex, giving lower accuracy than Iris.

Practical 6: Decision Tree with Cross Validation

Solution:

1. Load dataset → Classify tab.

2. Choose J48 → Test Option: 10-fold cross-validation.

3. Record accuracy.

Compare:

- Training set accuracy: 98%

- Cross-validation accuracy: 94%

Reason: Training accuracy is higher due to overfitting.

Conclusion: Cross-validation gives realistic performance measures.

Practical 7: Check Bias Against Attributes

Solution:

1. Load German Credit dataset.

2. Remove attribute 'foreign_worker'.

3. Train model → Record accuracy.

4. Remove attribute 'personal_status'.

Accuracy Comparison:

- Original: 72%

- Without foreign_worker: 72% (no change)

- Without personal_status: 70% (slight drop)

Conclusion: No major bias found against foreign workers. Personal status has some predictive importance.

Practical 8: Attribute Combination Testing

Solution:

Steps:

1. Use Attribute Selection Filter → Ranker + InfoGain.

2. Select top attributes: credit_amount, duration, checking_status.

Model Accuracy:

- Using all attributes: 72%

- Using top 5 attributes: 71%

Conclusion: Reduced attributes give similar performance; some attributes are redundant.

Practical 9: Decision Tree vs Cross Validation

Solution:

Normal Training:

Accuracy: 97%

Cross-validation:

Accuracy: 93%

Difference Reason:

- Training uses same data for testing → overly optimistic.

- CV uses unseen data → more realistic.

Conclusion: Cross-validation should be preferred.

Practical 10: Decision Tree Complexity vs Bias

Solution:

Steps:

1. In J48 settings → change pruning confidence:

  - High confidence (0.5) → Larger tree (low bias, high variance).

  - Low confidence (0.1) → Smaller tree (high bias, low variance).

Observed:

- Larger tree accuracy (training): 98%

- Smaller tree accuracy (CV): 93%

Conclusion: Higher complexity reduces bias but increases variance; balanced pruning gives best results.