Experiment 01 – Distance Matrix Construction

AIM: To compute distance matrices using Euclidean and Manhattan distance measures.

THEORY: Distance measures quantify similarity or dissimilarity between data points. Euclidean

distance is the straight line geometric distance, commonly used in clustering and spatial analysis.

Manhattan distance measures the sum of absolute coordinate differences, useful when movement occurs along grid-like paths. A distance matrix stores pairwise distances between all objects,

helping understand closeness patterns before applying algorithms like K-means or hierarchical clustering. Smaller values indicate higher similarity.

APPARATUS / TOOLS REQUIRED: WEKA, Notepad/Excel

STEPS TO PERFORM:

1. Create an ARFF file with the dataset attributes and values.
2. Load the dataset into WEKA ′n Preprocess tab.
3. Navigate to Visualize tab to inspect attribute relationships.
4. Compute Euclidean and Manhattan distances manually for each object pair.
5. Compare manual calculations with WEKA visualization output.

ADDITIONAL NOTES: Distance matrices help analyze relationships before applying clustering algorithms.

CONCLUSION: Both Euclidean and Manhattan matrices were computed and interpreted successfully.

Experiment 02 – K-Means Clustering

AIM: To implement the K-Means clustering algorithm on a dataset.

THEORY: K-Means is an unsupervised clustering technique that partitions data into k groups

based on similarity. It begins by selecting initial centroids, then repeatedly assigns each point to the nearest centroid using Euclidean distance. Centroids are recomputed after every iteration until

cluster membership stabilizes. The method is widely used in pattern recognition, marketing

segmentation, anomaly detection and data compression. K-Means minimizes the within cluster sum of squared distances and is efficient for large datasets.

APPARATUS / TOOLS REQUIRED: WEKA, CSV/Excel

STEPS TO PERFORM:

1. Create a CSV file containing numeric attributes.
2. Load the file in WEKA ′n Preprocess tab.
3. Go to Cluster tab and select SimpleKMeans.
4. Choose number of clusters (k) and distance function.
5. Click Start and observe cluster assignments and centroids.

ADDITIONAL NOTES: K Means requires numeric data and may converge to local minima depending on initial centroids.

CONCLUSION: Cluster centers and memberships were generated successfully, matching expected behavior.

Experiment 03 – Data Pre Processing Techniques

AIM: To preprocess a dataset using attribute selection and handling of missing values.

THEORY: Data preprocessing is essential before applying machine learning. Attribute Selection reduces dimensionality by identifying the most relevant features, improving accuracy and decreasing computation time. Methods like CFS Subset Eval with Best First Search evaluate subsets based on predictive ability and redundancy. Handling missing values is equally important—numeric attributes are often replaced with mean values, while nominal attributes use the most frequent class. Proper preprocessing improves model robustness, reduces noise, and ensures data consistency.

APPARATUS / TOOLS REQUIRED: WEKA, Excel

STEPS TO PERFORM:

1. Load the dataset in WEKA.
2. Use Select Attributes tab ’n choose CFS Subset Eval + Best First.
3. Run selection and view selected features.
4. For missing values ’n Apply ReplaceMissingValues filter.
5. Verify updated dataset using the Viewer.

ADDITIONAL NOTES: Preprocessing improves learning quality by eliminating irrelevant attributes and fixing incomplete data.

CONCLUSION: Selected attributes and cleaned data improved readiness for modeling.

Experiment 04 – Classification Using Naïve Bayes

AIM: To classify a dataset using the Naïve Bayes probabilistic classifier.

THEORY: Naïve Bayes applies Bayes' Theorem assuming conditional independence among features. It computes class probabilities by multiplying prior probabilities with likelihoods of attribute values given each class. Despite its simplicity, it performs remarkably well on categorical datasets such as text classification, medical diagnosis, and spam filtering. The classifier is computationally efficient and handles high dimensional data. It outputs the class having the highest posterior probability.

APPARATUS / TOOLS REQUIRED: WEKA, CSV/Excel

STEPS TO PERFORM:

1. Prepare dataset in CSV/ARFF format.
2. Load dataset in WEKA ’n Preprocess tab.
3. Select Classify tab ’n Choose NaiveBayes.
4. Set class attribute.
5. Run classifier and observe accuracy and predicted class.

ADDITIONAL NOTES: Naïve Bayes works best when features are weakly correlated and dataset is clean.

CONCLUSION: The classifier correctly predicted class labels using computed posterior probabilities.

Experiment 05 – Decision Tree Induction (J48)

AIM: To generate a decision tree using the J48 (C4.5) algorithm.

THEORY: Decision trees split data based on attribute tests that maximize information gain. Entropy measures disorder in data, while information gain quantifies reduction in uncertainty after a split.

J48 recursively selects the attribute with highest gain to form internal nodes and assigns class

labels at leaf nodes. Decision trees are interpretable, easy to visualize, and effective for categorical data. They are widely used in customer behavior prediction and diagnosis systems.

APPARATUS / TOOLS REQUIRED: WEKA, CSV/Excel

STEPS TO PERFORM:

1. Prepare dataset with categorical attributes.
2. Load it into WEKA.
3. Choose J48 from the Classify tab.
4. Set the class attribute and click Start.
5. Right click the result 'n Visualize Tree.

ADDITIONAL NOTES: Decision trees provide clear rule based classification paths.
CONCLUSION: The generated J48 tree matched the manually derived structure.

Experiment 06 – Association Rule Mining Using Apriori

AIM: To generate association rules from a dataset using the Apriori algorithm.

THEORY: Apriori identifies frequent itemsets by iteratively expanding candidate sets and pruning those that do not meet minimum support. It exploits the downward closure property—if an itemset is infrequent, all its supersets are also infrequent. After frequent itemsets are obtained, association

rules are generated using confidence thresholds. Apriori is a foundational algorithm for market basket analysis, recommendation systems and pattern discovery.

APPARATUS / TOOLS REQUIRED: WEKA, ARFF

STEPS TO PERFORM:

1. Create ARFF file containing transactions.
2. Load file in WEKA 'n Associate tab.
3. Select Apriori as the algorithm.
4. Configure minimum support and confidence.
5. Run algorithm and analyze generated rules.

ADDITIONAL NOTES: Apriori may be slow on very large datasets but is effective for small to medium sized data.

CONCLUSION: Strong rules with high support and confidence were generated successfully.

Experiment 07 – Agglomerative Hierarchical Clustering

AIM: To perform hierarchical clustering using the agglomerative approach.

THEORY: Agglomerative clustering is a bottom up hierarchical technique where each data point starts as its own cluster, and the closest clusters are repeatedly merged based on a distance metric such as Euclidean distance. Linkage criteria (single, complete, average) determine how distances between clusters are computed. The result is a dendrogram showing merging steps. This method is useful when the number of clusters is unknown and when visual hierarchy is meaningful.

APPARATUS / TOOLS REQUIRED: WEKA, ARFF

STEPS TO PERFORM:

1. Convert dataset into ARFF format.
2. Load dataset in WEKA.
3. Select Cluster ‘n HierarchicalClusterer.
4. Choose distance function and linkage type.
5. Click Start and visualize dendrogram.

ADDITIONAL NOTES: Agglomerative clustering provides a full cluster hierarchy rather than a fixed number of clusters.

CONCLUSION: The generated dendrogram clearly displayed cluster merging behavior.

Experiment 08 – FP Growth Algorithm

AIM: To generate frequent itemsets and association rules using FP Growth.

THEORY: FP Growth is an advanced frequent pattern mining algorithm that avoids candidate generation by building a compact FP Tree structure from the dataset. Items are ordered by frequency, and transactions are compressed into overlapping prefix paths. The algorithm then recursively mines conditional FP Trees to extract frequent patterns efficiently. FP Growth is faster than Apriori on large datasets and is widely used in market basket analysis and pattern recognition.

APPARATUS / TOOLS REQUIRED: WEKA, ARFF file

STEPS TO PERFORM:

1. Create an ARFF file containing transactional data.
2. Load dataset into WEKA ‘n Preprocess.
3. Go to Associate tab ‘n select FPGrowth.
4. Set minimum support and other parameters.
5. Run algorithm and review frequent itemsets and rules.

ADDITIONAL NOTES: FP Growth is highly efficient due to tree based compression.

CONCLUSION: Frequent itemsets and strong association rules were successfully extracted.