Christopher LaJon Morgan
Sep 14,2013
CS478: Brother Christophe

1. **Using the Cardiology dataset, do the following:**
   - **Build a neural network (using the Multilayer Perceptron/Backpropagation algorithm) that predicts whether a patient has a heart condition. Record the 10-fold cross-validation accuracy of your model as A_1. Cross-validation is a form of model validation where a dataset is split into folds, and the learning algorithm is trained on all but one fold and tested on the remaining fold; the process is repeated for each fold and accuracy is averaged over all folds.**

     81.8482%

   - **Create a new attribute *coarseBloodPressure*, with values: 1 if *blood pressure* is less than or equal to 120, 2 if *blood pressure* is greater than 120 but less than or equal to 150, and 3 if *blood pressure* is greater than 150.**

     Done

   - **Build a neural network (using the same algorithm as above) that predicts whether a patient has a heart condition, using the new attribute*coarseBloodPressure* instead of the original *blood pressure*. Record the 10-fold cross-validation accuracy of your model as A_2.**

     81.1881%

   - **Compare A_1 and A_2. Any comments?**

     The two runs appear to be arguable the same. It appears that we didn't really gain any information by binning the different blood pressures. It appears we might have even lost some classifying power by binning the data.

   - **Remove all records whose value of the attribute *resting ecg* is Abnormal.**

     Done

   - **Construct a decision tree that predicts whether a patient has a heart condition, given the attributes *age*, *sex*, *chest pain type*, *coarseBloodPressure*,*angina*, *peak*, and *slope*. Insert the confusion matrix obtained with 10-fold cross-validation in your report.**

     Using J48:
     === Confusion Matrix ===

     ```
      a   b   <-- classified as
     90  48 |   a = Sick
     35 130 |   b = Healthy
     ```

2. **Using the CPU dataset, do the following:**
   - **Cluster the data (using the simple k-Means algorithm, with k=3) and report on the nature and composition of the extracted clusters.**

     Clusters seem to have an unequal distribution with 74% of the data in cluster 2 falling in class 53.01, 16% of the data in cluster 1 falling in class 354.14, and 10% of the data in cluster 0 falling in class 25.1. It also appears that as MMax, CACH, CHMIN, CHMAX, MMIN increase the data moves from cluster 0 to 2 to1.

   - **Discretize the attributes *MMIN*, *MMAX*, *CACH*, *CHMIN* and *CHMAX* using 3 buckets in one step. Use the binning by frequency approach. Find associations among these attributes only (i.e., remove the other ones), using the Apriori algorithm, with support 0.1, confidence 0.95 and top 4 rules only being displayed. Insert the results in your report.**

     Assert: support 0.1 == only lowerBoundMinSupport 0.1

     It appears that a small CACH and CHMAX imply a small CHMIN. Higher MMIN, MMAX, and CHMAX, with potentially a higher CACH, imply a higher CHMIN. It also seems that a large MMAX, CACH, and CHMIN imply a large CHMAX.

     It is also interesting to note that as one decreases the confidence level required, the number of constraints on the left hand side of the rule decrease. For example, with less confidence it looks like a large CACH directly implies a large MMAX.

     === Run information ===

     Scheme: weka.associations.Apriori -N 4 -T 0 -C 0.95 -D 0.05 -U 1.0 -M 0.1 -S -1.0 -c -1
     Relation: cpu-weka.filters.unsupervised.attribute.Discretize-F-B3-M-1.0-Rfirst-last-weka.filters.unsupervised.attribute.Remove-R1-weka.filters.unsupervised.attribute.Remove-R6
     Instances: 209
     Attributes: 5
     MMIN
     MMAX
     CACH
     CHMIN
     CHMAX
     === Associator model (full training set) ===

     Apriori
     =======

     Minimum support: 0.15 (31 instances)

Minimum metric <confidence>: 0.95
Number of cycles performed: 17

Generated sets of large itemsets:

Size of set of large itemsets L(1): 15

Size of set of large itemsets L(2): 29

Size of set of large itemsets L(3): 12

Size of set of large itemsets L(4): 4

Best rules found:

1. MMIN='(2150-inf)' CACH='(20-inf)' CHMAX='(15.5-inf)' 35 ==> CHMIN='(5.5-inf)' 35 conf:(1)
2. MMIN='(2150-inf)' MMAX='(14000-inf)' CHMAX='(15.5-inf)' 32 ==> CHMIN='(5.5-inf)' 32 conf:(1)
3. MMAX='(14000-inf)' CACH='(20-inf)' CHMIN='(5.5-inf)' 36 ==> CHMAX='(15.5-inf)' 35 conf:(0.97)
4. CACH='(-inf-0.5]' CHMAX='(-inf-5.5]' 33 ==> CHMIN='(-inf-1.5]' 32 conf:(0.97)

- **Using the original CPU dataset, list the eigenvalues associated with the attributes selected by the Principal Components Analysis method, when the amount of variance you wish covered by the subset of attributes is 75%.**

Eigenvalues
3.35674
0.82936
0.73923