Document Analysis Assignment2-ML Written

Q2 Clustering with Naive Bayes

a

K-MEANS($\{\overrightarrow{x_1}, \overrightarrow{x_2}, \ldots\ \overrightarrow{x_n}\}, K$)

1. ($\{\overrightarrow{s_1}, \overrightarrow{s_2}, \ldots\ \overrightarrow{s_n}\}$) $\leftarrow$ SelectRandonSeeds($\{\overrightarrow{x_1}, \overrightarrow{x_2}, \ldots\ \overrightarrow{x_n}\}$, k)
2. D $\leftarrow$ $D_1, D_2, \ldots\ D_k$ //D represent the whole training set, Dk represent the set of same cluster.
3. for k $\leftarrow$ 1 to K
4. $D_k \leftarrow (y_k,\ \overrightarrow{s_k})$ // init D by label each random seed as one cluster, y_k is the label of documents, represent a class.
5. while stopping criterion has not been met
6. V, prior[K], condprob[M][K] $\leftarrow$ $TrainNB(D)$ //Recomputation, return the vocabulary set V, class probability list prior[K], and term probability of each class list condprob[M][K], M represent number of terms construct the vector.
7. for n $\leftarrow$ 1 to N
8. for k $\leftarrow$ 1 to K
9. $y_k \leftarrow arg^{max}_{y_k}$[log prior[$D_k$] + $\sum_{i=1}^{m} log$(condprob[t_i][k])] // reassignment, m is the number of term in the vector
10. $D_k \leftarrow (y_k,\ \overrightarrow{x_n})$
11. return D

Train NB algorithm [1]:

TrainNB(D)

1. V $\leftarrow$ ExtractVocabulary(D)
2. N $\leftarrow$ CountDocs(D)
3. for each y_k
4. $N_k \leftarrow$ CountDocs(D_k)
5. $prior[c] \leftarrow N_c/N$
6. $text_c \leftarrow$ ConcatenateTextOfAllDocsInClass(D_k, y_k)
7. for each t $\in$ V
8. condprob[t][c] $\leftarrow \frac{T_{ct}+1}{\sum_{t'}(T_{ct'}+1)}$
9. return V, prior, condprob

b

No, they will not converge the same answers.

Supervised method and unsupervised use different method to detect the outlier, which will cause differences between the outcome clusters.

Q3 Logistic Regression

1. As the money offered increased geometrically, the number of males who will offer their privacy shows a nearly "linear" growth.
2. Observed proportion = 18/20 = 90%
3. Observed odds (males selling their privacy)= P/(1-P)= $\frac{18/20}{1-18/20}$= 0.9:0.1 = 9:1
4. $\widehat{y_{16}} = log\left(\frac{p_{16}}{1-p_{16}}\right) = -2.81856 + 1.25895 * log_2 16 = -2.81856 + 1.25895 * 4 = 2.21724$

   $\frac{p_{16}}{1-p_{16}} = e^{2.21724} = 9.18$

   So the predicted odds of for a sum of 16 is 9.18

   $P_{16} = 9.18 * (1 - P_{16})$

   $P_{16} = 90.19\%$

   The predicted proportion of males taking the deal is 90.19%.

5. $\widehat{y_{32}} = log\left(\frac{p_{32}}{1-p_{32}}\right) = -2.81856 + 1.25895 * log_2 32 = -2.81856 + 1.25895 * 5 = 3.47619$

   $\frac{p_{32}}{1-p_{32}} = e^{3.47619} = 32.336$

   $\frac{\frac{p_{32}}{1-p_{32}}}{\frac{p_{16}}{1-p_{16}}} = 3.52$

   If we increase the sum from 16 to 32, the predicted odds will increase from 9.18 to 32.336, which is (3.52-1) = 2.52 multiple increase.

6. According to the plot, usually females value their privacy more than males do. Especially in the interval offered-money increased from 2,000 to 16,000, the growth of males number are faster than females number who provide their privacy.
7. As can be seen from plot, when the proportion of males taking the deal is 0.5, the logarithm (base 2) of offered money is around 2.2, which means:
   $log_2(offeredmoney) = 2.2$
   $\text{offeredmoney} = 2^{2.2} \approx 4.59$
   We should offer around 4.59 thousand of money to obtain the data of 50% of the males.
8. Similar as sub question 7, when the proportion of males taking the deal is 0.5, the logarithm (base 2) of offered money is around 3.3, then we have:
   $log_2(offeredmoney) = 3.3$

   $\text{offeredmoney} = 2^{3.3} \approx 9.85$

We should offer around 9.85 thousand of money to obtain the data of 50% of the males.

9. Because "we can make this a linear function of x without fear of nonsensical results." [2]
As described in textbook,
If we want to model the conditional probability of a numerical predictor variable x with a binary response, we can choose p(x), log p(x) or log(p/1-p) be a linear function of x. However, as linear functions are unbounded but P must be between 0 and 1, p(x) is not a good idea.
log p(x) has the similar problem, logarithms are unbounded in only direction, and linear functions are not.
Logit transformation is the easiest modification of log p which has an unbounded range.

Q4 Hierarchical Classification

1. We can use the decision tree induction [3] for this hierarchical.
Train the hierarchical classifier:
    a. Take all the training documents set D have been assigned to node as input.
    b. For k mutually exclusive classes, split to k1 = k/b, k2 = k/b, … kb = k/b classes as different training set to train b classifiers as the first level of classifiers, we get classifiers c1, c2, …. cb.
    c. For each subclass in first level, redo Step b to get the second level classifiers. That is, for k1, we get c11, c12, … c1b under classifier c1. Same as k2, k3, … kb.
    d. If the documents in node ni under classifier j are all in the same class, stop split this node ni in the next loop.
    e. Repeat step c, d until all the leaf nodes are composed by documents of the same class.

    Test document use the trained hierarchical classifier:
    a. Take test document d1 as input.
    b. Traverse the first level classifier c1, c2 … cb, choose the classifier ci of the largest similarity with d1.
    c. Traverse the sub classifier under ci, find the classifier cij of the largest similarity with d1.
    d. Repeat this process until the similarity between classifier and d1 reach a stop point.
    e. Label d1 with the stop class.
2. **Training time complexity:** $O\big((m * v + b * v) * d\big)$ or $O(m * v * d + n * v)$. The training algorithm will traverse the m training documents in every level, which is d for maximum. And also, it will traverse each nodes, which is n or for maximum b*d, as we have b branches at most for each level and d level all together.
Testing time complexity: $O(v * b^d)$. To classify one document, we will go at most b^d nodes.

The algorithm is computationally fesible for large-scale training and testing in practice because both training and testing time complexity are linear to the scale of training and testing set.

References

[1] P. R. H. S. Christopher D. Manning, An Introduction to Information Retrieval, New York : Cambridge University Press, 2008.

[2] "Chapter 12 Logistic Regression," [Online]. Available: http://www.stat.cmu.edu/~cshalizi/uADA/12/lectures/ch12.pdf. [Accessed 24 8 2016].

[3] M. K. J. P. Jiawei Han, Data Mining Concepts and Techniques, Waltham: Morgan Kaufmann, 2012.