1. ***Briefly describe how ChiMerge works for the following simple dataset attributes.[4]***

|  |  |  |
| --- | --- | --- |
| ***X*** | ***Y*** | ***Class*** |
| ***1*** | ***2*** | ***A*** |
| ***3*** | ***4*** | ***B*** |
| ***5*** | ***6*** | ***A*** |
| ***7*** | ***8*** | ***B*** |
| ***9*** | ***10*** | ***A*** |
| ***11*** | ***12*** | ***B*** |
| ***13*** | ***14*** | ***A*** |

Chi-merge is a supervised discretization algorithm that bins the data/feature by comparing two (or more) subsequent sorted value list of the feature for statistical independence. If the chi-squared score corresponding to the values in comparison suggests independence, then they are merged together as part of the same bin. The process continues until a stop criterion is met which usually is either the number of intervals or a chi-score for some arbitrary ‘X’ percentage of independence (usually 0.9, 0.95 or .99).

**Data-set inference**

* Applies generally to both X and Y as they are the same distribution
* The features to be discretized are already neatly sorted in ascending order
* Class labels are only 2, A and B and hence degrees of freedom, df = 1
* From chi-tables, for 90% probability of independence, chi-squared, ꭕ2 <= 2.71
  + This indicates a p-value which is not significant (0.1)
* The starting number of intervals is 7 in each case (X and Y)
* The class label is alternating in the 7 intervals
  + This means that the intervals are all equally independent
  + In other words, the intervals are already discrete

**Scope of chi-merge**

* Given that X, Y can already be considered binned/discretized, let us see what happens on applying chi-merge with a stopping criterion of number of intervals
  + The only clubbing stop criterion that makes sense to avoid a meaningless single interval (in which case the feature can as well probably be dropped) is based in number of intervals
* Calculating ꭕ2 on all different adjacent intervals, we get the same value (**2**) since each interval is completely similar. On merging intervals, we will see alternating patterns of ꭕ2 as is the quality of the data-set provided (with alternating class values)
  + This means as the number of samples increase to a larger value, ꭕ2 between the last 2 intervals (we would at least keep a minimum of 2 intervals to say that we are discretizing) would be ***M/(M+1)*** where ***M*** is the number of samples
  + This makes sense as there would only be 1 sample value differing between the last 2 discretized intervals (either A or B)
  + Further this value also approaches **1** as ***M*** increases by a large value again suggesting the difference in frequency of class label is only 1, (p-value converges around 0.32)
  + The chi-score value of **2** also makes sense in saying that the number of discrete items here with the degrees of freedom 1 is **2**
* Even with the usage of chi-merge for this data-set with intervals of 3 and 5 instead of 2 for comparison – this will give a ꭕ2 of 2/3 and 2/5 respectively for each interval which is less and less significance and more and more independence
* The upper bound of ꭕ2 at the 90% threshold will never be arrived at as we started with a very discrete set, and the chi-value from the first iteration will be the maximum value corresponding to a p-value of around 0.16 which is not significant
  + This also means that chi-merge is not significant and any unsupervised discretization works fairly as well in abstracting this data-set
* If the intervals are 7 – the table remains and the intervals are same
  + [1,1], [3,3], [5,5], [7,7], [9,9], [11,11], [13,13] for **X**
  + [2,2], [4,4], [6,6], [8,8], [10,10], [12,12], [14,14] for **Y**
* Looping on intervals and merging the smallest ranges we get the below for **X** –
  + 7 {**4/2**}: [1,1], [3,3], [5,5], [7,7], [9,9], [11,11], [13,13]
  + 6 {**3/4**}: [1,3], [5,5], [7,7], [9,9], [11,11], [13,13]
  + 5 {**4/3**}: [1,5], [7,7], [9,9], [11,11], [13,13]
  + 4 {**5/6**}: [1,7], [9,9], [11,11], [13,13]
  + 3 {**6/5**}: [1,9], [11,11], [13,13]
  + 2 {**7/8**}: [1,11], [13,13]
* Looping on intervals and merging the smallest ranges we get the below for **Y** –
  + 7 {**4/2**}: [2,2], [4,4], [6,6], [8,8], [10,10], [12,12], [14,14]
  + 6 {**3/4**}: [2,4], [6,6], [8,8], [10,10], [12,12], [14,14]
  + 5 {**4/3**}: [2,6], [8,8], [10,10], [12,12], [14,14]
  + 4 {**5/6**}: [2,8], [10,10], [12,12], [14,14]
  + 3 {**6/5**}: [2,10], [12,12], [14,14]
  + 2 {**7/8**}: [2,12], [14,14]
* ꭕ2 values for ever iteration in given in flower braces along the row for the merged intervals
  + Rest of the intervals would still have the same ꭕ2 value of **2**
* Merging into one whole interval would give ꭕ2 as **7**
  + This is meaningful as there are 7 discrete alternating intervals
  + For **X** {**7**}: [1,13]
  + For **Y** {**7**}: [2,14]
  + This calculation follows similar lines and is show in the python notebook

**Final Thoughts**

* An approach for better feature selection and discretization would be to club X and Y as one feature and then apply discretization
* This would result in lesser features and better interval coverage
* Calculations for ꭕ2 are provided as a picture in the next page (from manual calculation) as it is tough to do it in word without spending significant amount of time

**Appendix (ꭕ2 Derivations)**

