Automatic binning for visual predictive checks

Christian Sonehag[1], Niklas Olofsson[1], Rasmus Simander[1],
Rikard Nordgren[2] and Kajsa Harling[2]

[1] Department of Scientific Computing, Uppsala University, Uppsala, Sweden
[2] Pharmacometrics group, Department of Pharmaceutical Biosciences, Uppsala University, Uppsala, Sweden

Introduction

Visual predictive checks (VPC) [1] is a model diagnostic that requires binning of the observations in the dimension of the independent variable, usually time. A fast and objective method for binning is desirable. An automatic binning algorithm has previously been proposed for VPCs [2]. In this study we explore a novel automatic algorithm that is based on K-means clustering with a data density function to penalize adding a bin edge where the data is dense, and implement the algorithm in Perl Speaks NONMEM, PsN [3].

Method

Placing the bin edges

A commonly used way to divide a data set into K bins is to use the K-means clustering algorithm [4] which for a given K seeks to minimize $O_0(K) = \Sigma(W)$ where W_j is the sum of squared distances between points in bin j to the mean in bin j, and $\Sigma(W)$ is the sum of all W_j. Figure 1 shows the result of K-means clustering on Data set 1 (data from [2]), a simulated PK dataset with six clusters. The algorithm fails to separate the first two clusters and instead places a bin edge in the last cluster.

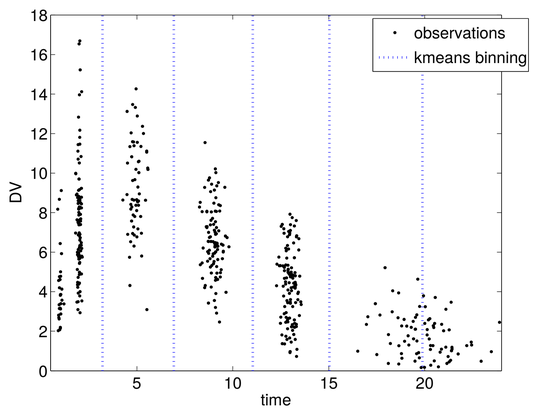

Figure 1. Data set 1, simulated PK data, k-means binning using K=6.

We propose a novel algorithm that for a given Kseeks to find the bin edges that minimizes the objective function $O(K) = \Sigma(W) + \alpha \cdot \Sigma\, \Phi(e_i)$, where $\Sigma(W)$ is the standard K-means objective function, Φ is a data density function, $\Sigma\, \Phi(e_i)$ is the sum of the data density function values at the bin edges e_i, and alpha is a scaling factor. The data density function Φ is obtained by kernel density estimation using a Gaussian kernel [4]. That is, a Gaussian density function is placed at each data point, and the sum of the density functions is computed over the range of the data. The bandwidth for the kernel is initially chosen as $\sigma(k) \cdot n_k^{-0.2}$, where $\sigma(k)$ is the standard deviation of the data in bin k and n_k is the number of data points in the bin [5]. If the kurtosis of the bin data resembles the kurtosis of data that originate from a single cluster of Gaussian distributed measurements, this choice of band width is kept, but if the kurtosis is small, indicating that the bin data originate from more than one Gaussian distribution, the initial band width is decreased to increase the resolution of the data density function and decrease the penalty for moving a bin edge into this area. An initial binning is required to obtain the data density function. To obtain an initial binning we minimize $O_0 = \Sigma(W)$. The data density function is computed based on this initial binning, and after possible adjustment of the bandwidth, the data density function is not updated even if bin edges are moved in a later stage of the algorithm. The same optimization algorithm is used both for minimizing $\Sigma(W)$ to get an initial binning and for minimizing $O(K)$ after the the data density function is fixed.

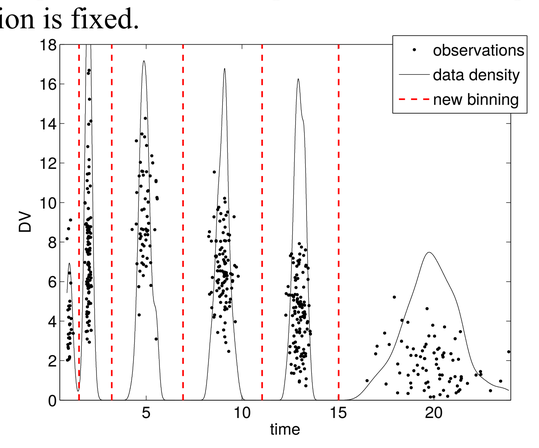

Figure 2. Data set 1, simulated PK data, new binning algorithm using K=6.

The algorithm iterates between two stages. In stage 1 we try to move the bin edges one by one within its two neighbors to decrease the objective function. When no bin edge can favorably be moved in this manner, the algorithm switches to stage 2. Here we attempt to take out the bin edges one by one and place them between two other bin edges to decrease the objective function. If there is any movement of a edge that results in a decreased objective function then we perform that movement and return to stage 1, otherwise we stop. Figure 2 shows the data density function and the result of the new binning algorithm on the simulated data set.

Choosing the number of bins

We need to choose the optimal K. We tried using K that minimized O(K), the K that maximized the Calinski and Harabaszs [6] function, and the ratio between the two.

Results

The best values of the different parameters of the algorithm were judged to be $\alpha = 7.8 \cdot \text{argmax}(W)$, cutoff C = 2.5 to classify the kurtosis as Gaussian/non-Gaussian and factor F = 0.25 as the bandwidth reduction factor for non-Gaussian bin data. The best criterion for selecting K was found to be the ratio between the objective function and the Calinski and Harabaszs function. Figure 3 shows the result of fully automatic binning on Data set 2, real data moxonidine PK [7] where subjects have been sampled five times at three visits, i.e. 15 samples per individual in total.

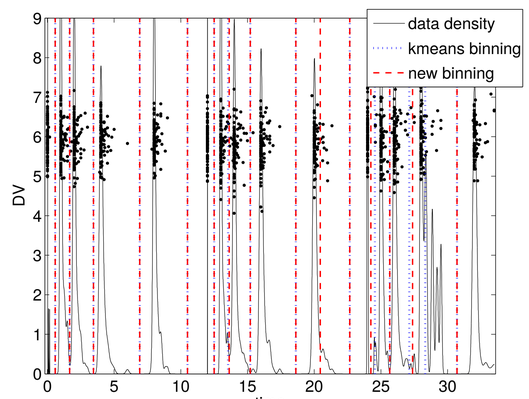

Figure 3. Data set 2: moxonidine PK data with fully automatic binning.

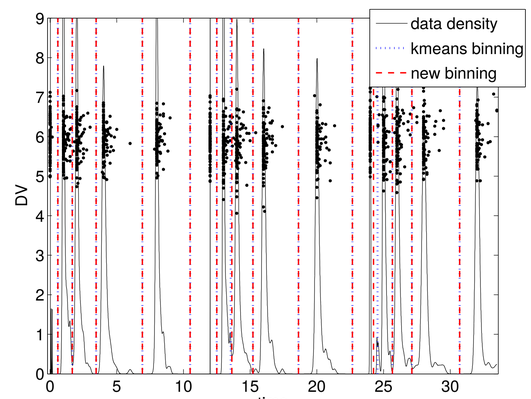

Figure 4. Data set 2, moxonidine PK data with K=15

The algorithm chooses 16 bins for this data set, which is one more bin than the ideal 15. Figure 4 shows the binning when K is manually set to 15. When time after dose is used as the independent variable for Data set 2, the fully automatic binning fails and chooses K=21 . This is caused by the adaptive bandwidth of the gaussian kernel becoming very small and the resolution of the data density function becoming too high (not shown). Figure 5 shows the binning result when the independent variable is time after dose and K is manually set to 5, the number of sampling times after each dose. In this case the algorithm places the bin edges in appropriate places.

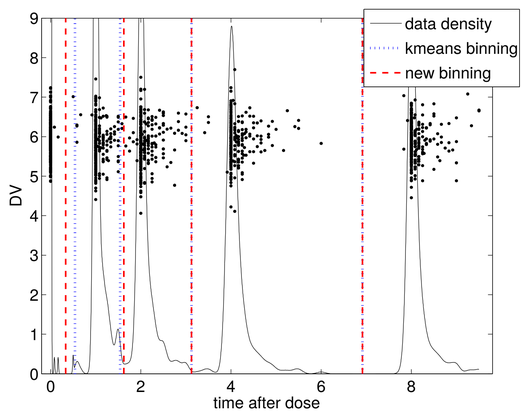

Figure 5. Data set 2, moxonidine PK data, K=5.

Conclusion

We have developed a binning algorithm that can accurately place bin edges given the number of bins K. The algorithm cannot always find an appropriate value of K. When using the algorithm in PsN's vpc program it is important to restrict the range of K to reasonable values given the study protocol/data set. Examples:
`vpc run1.mod -auto_bin=8,12 (`*`additional options`*`)`
makes the algorithm try K from 8 to 12.
`vpc run1.mod -auto_bin=9 (`*`additional options`*`)`
makes the algorithm use K=9.

References

[1] Karlsson M, Holford N. A Tutorial on Visual Predictive Checks, PAGE Meeting, Marseille, 2008.
[2] Lavielle M, Bleakley K. Automatic data binning for improved visual diagnosis of pharmacometric models, J Pharmacokinet Pharmacodyn. 2011 Dec;38(6):861-71
[3] Lindbom L, Ribbing J, Jonsson EN. Perl-speaks-NONMEM (PsN)--a Perl module for NONMEM related programming. Comput Methods Programs Biomed. 75(2):85-94.
[4] MacQueen JB. Some methods for classification and analysis of multivariate observations. Proceedings of 5th Berkeley Symposium on Mathematical Statistics and Probability 1. University of California Press. pp. 281–297.
[5] Silverman BW. Density Estimation for Statistics and Data Analysis, 1998.
[6] Milligan G, Cooper M. An examination of procedures for determining the number of clusters in a data set, Psychometrika-vol. 50, no 2, 159- 179. June 1985
[7] Karlsson MO, Jonsson EN, Wiltse CG, Wade JR. Assumption testing in population pharmacokinetic models: illustrated with an analysis of moxonidine data from congestive heart failure patients. J Pharmacokinet Biopharm. 1998;26:207–46.