Estimating the number of clusters using Cross Validation

Wei Fu *
Department of IOMS, New York University
and
Patrick O. Perry
Department of IOMS, New York University

April 7, 2016

Abstract

Many clustering methods, including k-means, require the user to specify the number of clusters as an input parameter. A variety of methods have been devised to choose the number of clusters automatically, but they often rely on strong modeling assumptions. We propose a data-driven approach to estimate the number of clusters based on a novel form of cross-validation. This differs from ordinary cross-validation, because clustering is fundamentally an unsupervised learning problem. Simulation and real data analysis results show that our proposed method outperforms existing methods, especially in high-dimensional settings with heavy-tailed data.

Keywords: clustering, unsupervised learning, data-driven method

technometrics tex template (do not remove)

*The authors gratefully acknowledge

1 Introduction

As a main task of exploratory data analysis, clustering organizes unlabeled observations into groups such that observations in same group are more similar compare to those in different group. Clustering is an important topic in unsupervised learning because it can reveal the internal structure of data through grouping, segment the data through partitioning and summarize data for other purposes such as dimension reduction. It has being widely used in various fields such as psychology, biology, statistics and machine learning including pattern recognition, image segmentation etc (Jain et al., 1999).

After being proposed more than 50 years, k-means remains one of the most popular and widely used clustering algorithms (Jain, 2010). Like many other clustering methods, k-means requires an input parameter k, the number of clusters, to be specified by the user. Automatically and quantitatively deciding such parameter is important and yet unsolved problem (Fujita et al., 2014). Various methods have been proposed to tackle this difficulty. One ad hoc approach is to explore the relationship between W_k (within-cluster dispersion) and the number of cluster k for a certain clustering method such as k-means. Since W_k decreases as k increases, one usually find the "elbow" of curve obtain by plotting W_k versus k as the appropriate number of clusters. The example on the top row of Figure 1 demonstrates such approach for data with $k = 4$, where the "elbow" point indeed reveals the true number of clusters. This is based on the idea that under partitioning data set has more impact than over partitioning data set in terms of W_k. However, locating the "elbow" point is somewhat subjective and sometimes is not appropriate to select the optimal k. The second example on the bottom row of Figure 1 shows a situation where there is no clear choice of the "elbow" point – both $k = 2$ and $k = 3$ can be viewed as the "elbow" point. What's more, the true $k = 4$ can never be selected as the optimal k using such approach in this case since it can hardly be viewed as the "elbow" of the curve.

Recently, there are several new proposals to find the k automatically. Gap statistics (Tibshirani et al., 2001) estimates k by comparing the change in within-cluster dispersion with that expected under an appropriate reference null distribution. Specifically, the graph of $\log(W_k)$ is compared with its expectation under an appropriate null reference distribution of the data. The value of k associated with the largest gap between $\log(W_k)$ and the

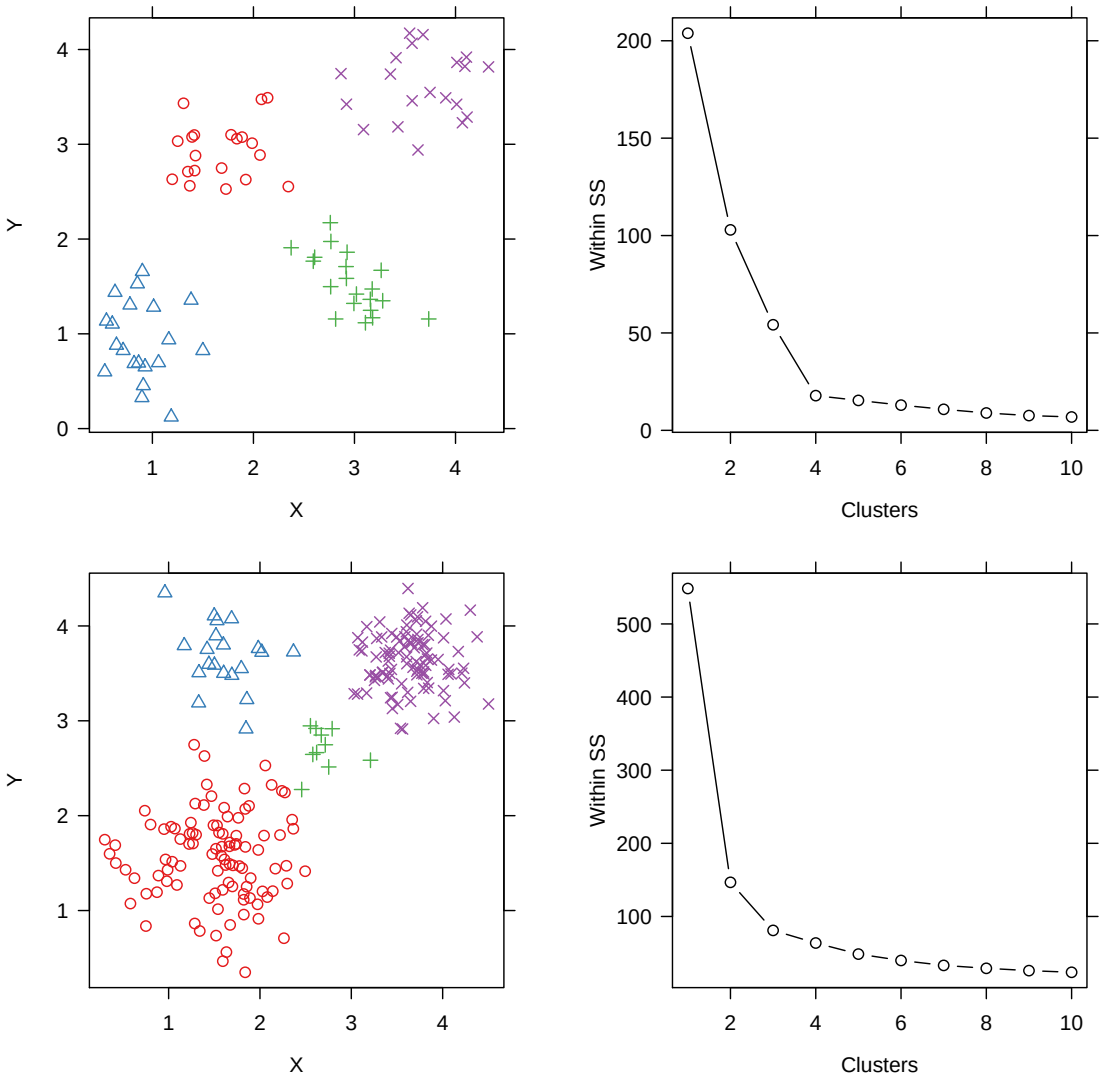


Figure 1: Left panels show the (X, Y) data points; right panels show the corresponding values of the within-cluster sum of squares W_k plotted against the number of clusters, k.

reference curve is selected as optimal k. Sugar and James (2003) proposed an approach which finds the number of clusters based on distortion, a quantity that measures the average distance, per dimension. It's backed by a rigorous theoretical justification based on information-theoretic ideas. Fraley and Raftery (2002)'s Model-based method employs the EM algorithm to estimate the parameters in Gaussian mixture model, and select the best model (k) using BIC criterion. Stability-based criterion is also proposed to locate the best k by some authors such as Ben-Hur et al. (2001), Wang (2010) and Fang and Wang (2012). Chiang and Mirkin (2010) provides a nice review of existing methods for finding the right k in published literature.

Most existing methods are either model based method requires strong modeling assumptions or otherwise lack of clear interpretation and theoretical justification. Although many view selecting the number of clusters as a model selection problem, very few approaches this problem from the prediction point of view. Select model with smallest prediction error via cross-validation is one of the simplest and most widely used model selection techniques in supervised learning. The lack of true class (label) in data set makes the adoption of cross-validation into unsupervised learning problem difficult. A naive algorithm may works as following: with the data split by rows into training set and test set, cluster the training data with parameter k; then predict each observation in test data by the closest cluster center returned in last step. The prediction error is calculated by the difference between the predicted cluster centers and observations' true value, and algorithm picks the k which minimizes such error. Such algorithm will always pick the maximum number of k allowed because more cluster centers means more tight fit to the data space, therefore smaller within-cluster dispersion even if the prediction is evaluated on an independent test set (Hastie et al., 2009).

One exception is Tibshirani and Walther (2005), which selects the optimal k by prediction strength. The strategy is to first cluster the test data and training data into k clusters respectively. Then, for each pair of observations that are assigned to the same test cluster, algorithm determines whether they are also assigned to the same cluster based on the training centers. The intuition here is, if $k = k_0$, the true number of clusters, then the k training set clusters will be similar to the k test clusters, and hence will predict them well.

However, a specifically defined prediction error measure is used in such procedure, which is quite different from the one commonly used in cross-validation procedure in supervised learning. Wang (2010) also uses cross-validation to select the optimal k. Instead of selecting k which minimizes prediction error, such method picks k which minimizes the specifically defined clustering instability. Note that these methods minimize measures that are specifically defined, which makes these methods hard to interpret or compare to other methods through analogy because the underlying measures are unique and not well understood.

The prediction error measure in our proposed method is exactly the same as in supervised learning. Our method is a complete data-driven approach which doesn't rely on strong model assumption. Through novel form of partitioning data set, we effectively transferred an unsupervised learning problem into a supervised leaning problem, which is one of the kind. By doing so, we are able to employing the cross-validation procedure in clustering a similar way as in supervised learning problem, so that much of the intuition from supervised learning carries over. Hence, it's easy for reader to understand the intuition behind our proposed method. Simulation and real data application shows the superior performance of our proposed method compare with existing methods in high-dimension settings and heavy-tailed data. Since the embedded cross-validation procedure is well understood, it also makes our method potentially easily to be extended in future study.

Some proposed methods provide theoretical consistency result for choosing k. Assume data is a mixture of G p-dimension clusters with equal prior, identically distributed with common covariance matrix and finite fourth moments in each dimension, Sugar and James (2003) shows the Jump method will pick the correct k under the conditions that cluster centers are sufficiently separated and an appropriate transformation is used. Specifically, let $\Delta\sqrt{p}$ denotes the minimum Euclidean distance between cluster centers, Sugar and James (2003) shows that the jump will be maximized at $k = G$ given that $\Delta > 6$ and the existence of a positive Y such that

$$\left(\frac{p\Delta^2 W}{9G}\right)^{-Y} + \left(W\left[\frac{2^{2H^*(X)}}{K_{max}^2 2\pi e}\right] - (\frac{\Delta}{6})^2(1-W)\right)^{-Y} < 2$$

and $\left(\frac{p\Delta^2 W}{9G}\right)^{-Y} < 1/2$, where $H^*(X)$ is the minimum entropy of the cluster membership

over each dimension and $W = 1 - \frac{6^4 V_{\mathbf{X}}}{(\Delta^2 - 36)^2}$ with $V_{\mathbf{X}} = var\left(1/p||\mathbf{X} - \mu_j||^2_{\Gamma^{-1}} \mid \mathbf{X} \text{ in jth cluster}\right)$

Tibshirani and Walther (2005) proved that the prediction strength method is consistent under the setting that the observations in p-dimension are uniformly distributed within one unit from their respective cluster centers, with minimum distance between the G cluster centers is four unit. Under such well-separated setting, they show that

$$ps(G) = 1 + o_p(1), \qquad \sup_{k \geq G+1} ps(k) \leq \frac{2}{3} + o_p(1)$$

therefore $\hat{k}$ is consistent in estimating G. When data does come from mixture of Gaussian distribution, the model-based method of Fraley and Raftery (2002) is consistent in estimating k in low dimension.

Let Ψ denotes any given base-clustering algorithm (e.g k-means), and $\Psi(Z, k)$ the clustering obtained by applying Ψ on data Z with parameter k. Also let k_0 be the optimal number of cluster and $d\{\Psi(Z_1, k), \Psi(Z_2, k)\}$ be the measure of distance between clustering $\Psi(Z_1, k)$ and $\Psi(Z_2, k)$ where Z_1 and Z_2 are two independent samples from population. Wang (2010) shows that their proposed method will consistently select $k = k_0$, given such k_0 exists and $d\{\Psi(Z_1, k), \Psi(Z_2, k)\}$ converges to zero at proper rate r_k. However, k_0 is defined as the most stable number for Ψ among all the possible k, instead of the number of components in mixture model or well-separated compact regions commonly seen in literature. Therefore, their consistent result actually is convergent result, i.e. the selected k converges to its limit k_0 given such limit exists.

2 Cross-validation for selecting the number of clusters

Cross-validation is commonly used for model selection in supervised learning problems. In these settings, the data comes in the form of N predictor-response pairs, $(X_1, Y_1), \ldots, (X_N, Y_N)$, with $X_i \in \mathbb{R}^p$ and $Y_i \in \mathbb{R}^q$. The data can be represented as a matrix with N rows and $p+q$ columns. We partition the data into K hold-out "test" subsets, with K typically chosen to be 5 or 10. For each "fold" r in the range $1, \ldots, K$, we permute the rows of the data

matrix to get $\mathfrak{X}$, a matrix with the rth test subset as its trailing rows. We partition $\mathfrak{X}$ as

$$\mathfrak{X} = \begin{bmatrix} X_{\text{train}} & Y_{\text{train}} \\ X_{\text{test}} & Y_{\text{test}} \end{bmatrix}.$$

We use the training rows $[X_{\text{train}}\ Y_{\text{train}}]$ to fit a regression model $\hat{Y} = \hat{Y}(X)$, and then evaluate the performance of this model on the test set, computing the cross-validation error $\|Y_{\text{test}} - \hat{Y}(X_{\text{test}})\|^2$ or some variant thereof. We choose the model with the smallest cross-validation error, averaged over all K folds.

In unsupervised learning problems like factor analysis and clustering, the features of the observations are not naturally partitioned into "predictors" and "responses", so we cannot directly apply the cross-validation procedure described above. For factor analysis, there are at least two versions of cross-validation. Wold (1978) proposed a "speckled" holdout, where in each fold we leave out a subset of the elements of the data matrix. Wold's procedure works well empirically, but does not have any theoretical support, and it requires a factor analysis procedure that can handle missing data. Owen and Perry (2009) proposed a scheme called "bi-cross-validation" wherein each fold designates a subset of the data matrix columns to be response and a subset of the rows to be test data. This generalized a procedure due to Gabriel (2002), who proposed holding out a single column and a single row at each fold. Owen and Perry proved that this procedure is self-consistent, in the sense that it performs the correct model selection in the absence of noise, and Perry (2009) provided more theoretical support.

In this report, we extend the Wold and Gabriel methods to the clustering problem, specifically to choose an appropriate number of clusters for a dataset. We prove that the Gabriel method is self-consistent, and we analyze some of its properties in the presence of noise. We compare these methods to state-of-the-art algorithms, and show that both are competitive.

We now give the details of how to implement the Gabriel cross-validation to locate the optimal cluster number k. The Wold cross-validation algorithm is described in Appendix A.

2.1 Gabriel CV algorithm

We are given a data matrix with N rows and P columns. In each fold of cross-validation, we permute the rows and columns of the data matrix and then partition the rows and columns as $N = n + m$ and $P = p + q$ for non-negative integers n, m, p, and q. We treat the first p columns as "predictors" and the last q columns as "responses"; similarly, we treat the first n rows as "training" and the last m rows as "test". In block form, the permuted data matrix is

$$\mathfrak{X} = \begin{bmatrix} X_{\text{train}} & Y_{\text{train}} \\ X_{\text{test}} & Y_{\text{test}} \end{bmatrix},$$

where $X_{\text{train}} \in \mathbb{R}^{n \times p}$, $Y_{\text{train}} \in \mathbb{R}^{n \times q}$, $X_{\text{test}} \in \mathbb{R}^{m \times p}$, and $Y_{\text{test}} \in \mathbb{R}^{m \times q}$.

Given such a partition of $\mathfrak{X}$, we perform four steps for each value of k, the number of clusters:

1. **Cluster:** Cluster $Y_1, \ldots, Y_n$, the rows of Y_{train}, yielding the assignment rule $\hat{G}^Y : \mathbb{R}^q \to \{1, \ldots, k\}$ and the cluster means $\bar{\mu}_1^Y, \ldots, \bar{\mu}_k^Y$. Set $\hat{G}_i^Y = \hat{G}^Y(Y_i)$ to be the assigned cluster for row i.

2. **Classify:** Take $X_1, \ldots, X_n$, the rows of X_{train} to be predictors, and take $\hat{G}_1^Y, \ldots, \hat{G}_n^Y$ to be corresponding class labels. Use the pairs $\{(X_i, \hat{G}_i^Y)\}_{i=1}^n$ to train a classifier $\hat{G}^X : \mathbb{R}^p \to \{1, \ldots, k\}$.

3. **Predict:** Apply the classifier to $X_{n+1}, \ldots, X_{n+m}$, the rows of X_{test}, yielding predicted classes $\hat{G}_i^X = \hat{G}^X(X_i)$ for $i = n+1, \ldots, n+m$. For each value of i in this range, compute predicted response $\hat{Y}_i = \bar{\mu}^Y(\hat{G}_i^X)$, where $\bar{\mu}^Y(g) = \bar{\mu}_g^Y$.

4. **Evaluate:** Compute the cross-validation error

$$\text{CV}(k) = \frac{1}{m} \sum_{i=n+1}^{n+m} \|Y_i - \hat{Y}_i\|^2,$$

   where $Y_{n+1}, \ldots, Y_{n+m}$ are the rows of Y_{test}.

In principle, we could use any clustering and classification methods in steps 1 and 2. In this report, we use k-means as the clustering algorithm. For the classification step, we compute

the mean value of X for each class; we assign an observation to class g if that class has the closest mean (randomly breaking ties between classes). The classification step is equivalent to linear discriminant analysis with equal class priors and identity noise covariance matrix.

To choose the folds, we randomly partition the rows and columns into K and L subsets, respectively. Each fold is indexed by a pair (r, s) of integers, with $r \in \{1, \ldots, K\}$ and $s \in \{1, \ldots, L\}$. Fold (r, s) treats the rth row subset as "test", and the sth column subset as "response". We typically take $K = 5$ and $L = 2$. For the number of clusters, we select the value of k that minimizes the average of $\mathrm{CV}(k)$ over all $K \times L$ folds (choosing the smallest value of k in the event of a tie).

3 Self-Consistency of Gabriel CV method

This section gives the self-consistency proof of the proposed Gabriel method. Specifically, we will show that under appropriate conditions, in the absence of noise, the Gabriel cross-validation procedure finds the optimal number of clusters.

Because k-means algorithm is essential to the method, we review the procedure here. Given a set of observations $\{x_1, \ldots, x_n\}$, and a specified the number of clusters k, the goal of the k-means procedure is to find a set of k or cluster centers $A = \{a_1, \ldots, a_k\}$ minimizing the within cluster dispersion

$$W(A) = \sum_{i=1}^{n} \min_{a \in A} \|x_i - a\|^2.$$

This implicitly defines a cluster assignment rule

$$g(x) = \arg\min_{g \in \{1,\ldots,k\}} \|x - a_g\|^2,$$

with ties broken arbitrarily. We will assume that the k-means procedure finds an optimal solution, A, but we will not assume that this solution is unique.

It will suffice to analyze a single fold of the cross-validation procedure. As in in section 2.1 we assume that the P variables of the data set have been partitioned into p predictor variables represented in vector X and q response variables represented in vector Y. The

N observations have been divided into two sets: n train observations and m test observations. The following theorem gives conditions for Gabriel CV to recover the true number of clusters in the absence of noise.

Theorem 1. *Let $\{(X_i, Y_i)\}_{i=1}^{n+m}$ be the data from a single fold of Gabriel cross-validation. For any k, let $CV(k)$ be the cross-validation error for this fold, computed as described in Section 2.1. We will assume that there are K true centers $\mu(1), \ldots, \mu(K)$, with the gth cluster center partitioned as $\mu(g) = \big(\mu^X(g), \mu^Y(g)\big)$ for $g = 1, \ldots, K$. Suppose that*

(i) Each observation i has a true cluster $G_i \in \{1, \ldots, K\}$. There is no noise, so that $X_i = \mu^X(G_i)$ and $Y_i = \mu^Y(G_i)$ for $i = 1, \ldots, n+m$.

(ii) The vectors $\mu^X(1), \ldots, \mu^X(K)$ are all distinct.

(iii) The vectors $\mu^Y(1), \ldots, \mu^Y(K)$ are all distinct.

(iv) The training set contains at least one member of each cluster: for all g in the range $1, \ldots, K$, there exists at least one i in the range $1, \ldots, n$ such that $G_i = g$.

(v) The test set contains at least one member of each cluster: for all g in the range $1, \ldots, K$, there exists at least one i in the range $n+1, \ldots, n+m$ such that $G_i = g$.

Then $CV(k) < CV(K)$ for $k < K$, and $CV(k) = CV(K)$ for $k > K$, so that Gabriel CV correctly chooses $k = K$.

This theorem is implied by the following two lemmas.

Lemma 1. *Suppose that the assumptions of Theorem 1 are in force. If $k < K$, then* $\mathrm{CV}(k) > 0$.

Proof. By definition,

$$\mathrm{CV}(k) = \sum_{i=n+1}^{n+m} \|Y_i - \bar{\mu}^Y(\hat{G}_i^X)\|^2,$$

where $\bar{\mu}^Y(g)$ is the center of cluster g returned from applying k-means to $Y_1, \ldots, Y_n$. Assumptions (i) and (v), imply that as i ranges over the test set $n+1, \ldots, n+m$, the response Y_i ranges over all distinct values in $\{\mu^Y(1), \ldots, \mu^Y(K)\}$. Assumption (iii) implies that there are exactly K such distinct values. However, there are only k distinct values of $\bar{\mu}^Y(g)$. Thus, at least one summand $\|Y_i - \bar{\mu}^Y(\hat{G}_i^X)\|^2$ is nonzero. Therefore, $\mathrm{CV}(k) > 0$. □

Lemma 2. *Suppose that the assumptions of Theorem 1 are in force. If $k \geq K$, then* $\mathrm{CV}(k) = 0$.

Proof. From assumptions (i), (iii), and (iv), we know the cluster centers gotten from applying k-means to $Y_1, \ldots, Y_n$ must include $\mu^Y(1), \ldots, \mu^Y(K)$. Without loss of generality, suppose that $\bar{\mu}^Y(g) = \mu^Y(g)$ for $g = 1, \ldots, K$. This implies that $\hat{G}_i^Y = G_i$ for $i = 1, \ldots, n$. Thus, employing assumption (i) again, we get that $\bar{\mu}^X(g) = \mu^X(g)$ for $g = 1, \ldots, K$.

Since assumption (ii) ensures that $\mu^X(1), \ldots, \mu^X(K)$ are all distinct, we must have that $\hat{G}_i^X = G_i$ for all $i = 1, \ldots, m+n$. In particular, this implies that $\bar{\mu}^Y(\hat{G}_i^X) = Y_i$ for $i = 1, \ldots, m+n$, so that $\mathrm{CV}(k) = 0$. □

4 Analysis of Gabriel Cross-Validation with Gaussian Noise

4.1 Single Cluster

Now we we analyze the asymptotic performance of Gabriel Cross-Validation, in the case of Gaussian noise. Our main result is that with single-cluster Gaussian data, if the predictor and response columns of $\mathfrak{X}$ are weakly correlated or independent, then the method will correctly prefer $k = 1$ to $k = 2$ clusters. We first state the result in the case where $\mathfrak{X}$ has two columns, and later generalize this result to higher dimensions.

Proposition 1. *Suppose that $\{(X_i, Y_i)\}_{i=1}^{n+m}$ is data from a single fold of Gabriel cross-validation, where each (X, Y) pair in $\mathbb{R}^2$ is an independent draw from a mean-zero multivariate normal distribution with unit marginal variances and correlation ρ. In this case, the data are drawn from a single cluster; the true number of clusters is* 1. *If $|\rho| < 0.5$, then* $\mathrm{CV}(1) < \mathrm{CV}(2)$ *with probability tending to one as m and n increase.*

Proof. Given $(X_1, Y_1), \ldots, (X_n, Y_n)$, we first apply k-means to $\{Y_i\}_{i=1}^n$. With $k = 1$, the single-cluster centroid will be equal to $\bar{Y}_n$, the sample mean of the $Y_1, \ldots, Y_n$, approximately

equal to $\mathrm{E}(Y) = 0$, with error of size $O_p(n^{-1/2})$. The cross-validation error will be

$$\mathrm{CV}(1) = \frac{1}{m} \sum_{i=n+1}^{n+m} \|Y_i - \bar{Y}_n\|^2 = 1 + O_p(m^{-1/2}) + O_p(n^{-1/2}).$$

Now we will consider the $k = 2$ case. If n is large enough, then Pollard (1981) showed that the centroids $\bar{\mu}_1^Y$ and $\bar{\mu}_2^Y$ will be close to $\mathrm{E}(Y \mid Y > 0) = \sqrt{2/\pi}$ and $\mathrm{E}(Y \mid Y < 0) = -\sqrt{2/\pi}$. We have used Lemma 3 (Appendix B) to compute the expectations. Further, Pollard (1982) showed that the errors will be of size $O_p(n^{-1/2})$.

If $\rho > 0$ and n is large enough, then classification rule learned from $\{(X_i, \hat{G}_i^Y)\}_{i=1}^n$ variables will be determined according to whether $X > 0$; if $\rho < 0$ then the decision is according to whether $X < 0$. More specifically, the decision boundary will be at $0 + O_p(n^{-1/2})$.

In the $\rho > 0$ case, the cross-validation error will be

$$\begin{aligned}\mathrm{CV}(2) &= \frac{1}{m} \sum_{i=n+1}^{n+m} \|(Y_i - \bar{\mu}_1^Y)1\{\hat{G}_i^X = 1\}\|^2 + \|(Y_i - \bar{\mu}_2^Y)1\{\hat{G}_i^X = 2\}\|^2 \\ &= \mathrm{E}[(Y-a)^2 1\{X > 0\}] + \mathrm{E}[(Y+a)^2 1\{X < 0\}] + O_p(m^{-1/2}) + O_p(n^{-1/2}),\end{aligned}$$

where $a = \sqrt{2/\pi}$. From the joint normality of X and Y, it follows that $Y \mid X$ is normal with mean ρX and variance $(1 - \rho^2)$, so that $\mathrm{E}[(Y - a)^2 \mid X] = (\rho X - a)^2 + (1 - \rho^2)$. Applying Lemma 3, we get that for large m and n, the Gabriel cross-validation error is close to $1 + a^2(1 - 2\rho)$.

In the $\rho < 0$ case, a similar calculation shows that CV(2) is close to $1 + a^2(1 + 2\rho)$. In particular, if $|\rho| < 0.5$, then with probability tending to 1 and m and n increase, the asymptotic cross-validation error for $k = 1$ will be smaller than for $k = 2$. □

We confirm this result with a simulation. We perform 10 replicates. In each replicate, we generate 20000 observations from a mean-zero bivariate normal distribution with unit marginal variances and correlation ρ. We perform a single 2×2 fold of Gabriel cross-validation and report the cross-validation mean squared error for the number of clusters k ranging from 1 to 5. Figure 2 shows the cross-validation errors for all 10 replicates. The simulation demonstrates that in the Gabriel cross-validation criterion chooses the correct

answer $k = 1$ whenever $\rho < 0.5$; the criterion chooses $k = 2$ clusters whenever $|\rho| > 0.5$.

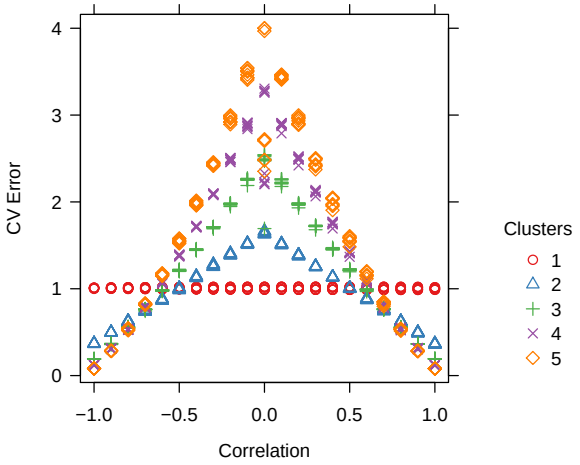


Figure 2: Cross-validation error on 10 replicates, with the number of clusters k ranging from 1 to 5. Data is generated from two-dimensional multivariate normal distribution with correlation ρ. The Gabriel cross-validation criterion chooses the correct answer $k = 1$ whenever $\rho < 0.5$; the criterion chooses $k = 2$ clusters whenever $|\rho| > 0.5$.

The reason why Gabriel CV method tends to select larger k when correlation is high between dimensions is that it resembles the naive method mentioned in section 1 under such circumstance. In fact, if X, Y are perfectly correlated, Gabriel method is equivalent to the naive method.

Proposition 1 gives a very simple condition for the Gabriel method to correctly pick $k = 1$ with single cluster in 2 dimensions. The following proposition generalizes such condition for data in arbitrary dimension.

Proposition 2. *Suppose that* $\{(X_i, Y_i)\}_{i=1}^{n+m}$ *is data from a single fold of Gabriel cross-validation, where each* (X, Y) *pair in* $\mathbb{R}^{p+q}$ *is an independent draw from a mean-zero multivariate normal distribution with covariance matrix* $\Sigma_{XY} = \left(\begin{smallmatrix} \Sigma_{XX} & \Sigma_{XY} \\ \Sigma_{YX} & \Sigma_{YY} \end{smallmatrix}\right)$, *with* Σ_{YY} *has leading eigenvalue* λ_1 *and corresponding eigenvector* u_1. *In this case, the data are drawn from a single cluster; the true number of clusters is* 1. *If* $\frac{\sqrt{\lambda_1}}{2} > \frac{u_1^T \Sigma_{YX} \Sigma_{XY} u_1}{\sqrt{u_1^T \Sigma_{YX} \Sigma_{XX} \Sigma_{XY} u_1}}$, *then* $\mathrm{CV}(1) < \mathrm{CV}(2)$ *with probability tending to one as* m *and* n *increase.*

Proof. Let X and Y be jointly multivariate normal distributed with mean $\mathbf{0}$ and covariance matrix Σ_{XY}, i.e.

$$(X, Y) \sim \mathcal{N}(\mathbf{0}, \Sigma_{XY})$$

where $\Sigma_{XY} = \begin{bmatrix} \Sigma_{XX} & \Sigma_{XY} \\ \Sigma_{YX} & \Sigma_{YY} \end{bmatrix}$.

Let $\Sigma_{YY} = U\Lambda U^T$ be the eigendecomposition of Σ_{YY}, with leading eigenvalue λ_1 and corresponding eigenvector u_1. Then the centroid of k-means applying on $(y_1, .., y_n)$ is on the first PC of Y,

$$E(u_1^T Y | u_1^T Y > 0) = \bar{\mu}_1^Y = \sqrt{2\lambda_1/\pi} u_1$$

and

$$E(u_1^T Y | u_1^T Y < 0) = \bar{\mu}_2^Y = -\sqrt{2\lambda_1/\pi} u_1$$

where $u_1^T Y \sim \mathcal{N}(0, \lambda_1)$.

To compute $\bar{\mu}_1^X = E(X | u_1^T Y > 0)$, we need to know the conditional distribution $X | u_1^T Y$. Since (X, Y) has multivariate normal distribution, $(X, u_1^T Y)$ also has a multivariate normal distribution with mean $\mathbf{0}$ and covariance matrix

$$\Sigma_{X, u_1^T Y} = \begin{bmatrix} \Sigma_{XX} & \Sigma_{XY} u_1 \\ u_1^T \Sigma_{YX} & \lambda_1 \end{bmatrix}$$

The conditional distribution $X | u_1^T Y$ is hence normal with mean

$$\mu_{X | u_1^T Y} = \Sigma_{XY} u_1 \lambda_1^{-1} u_1^T Y$$

Therefore,

$$\begin{aligned} \bar{\mu}_1^X &= E(X \mid u_1^T Y > 0) \\ &= E\left(E[X \mid u_1^T Y] \mid u_1^T Y > 0\right) \\ &= E\left(\Sigma_{XY} u_1 \lambda_1^{-1} u_1^T Y \mid u_1^T Y > 0\right) \\ &= \lambda_1^{-1} \Sigma_{XY} u_1 E(u_1^T Y \mid u_1^T Y > 0) \\ &= \lambda_1^{-1} \Sigma_{XY} u_1 \sqrt{2\lambda_1/\pi} \\ &= \sqrt{2/\lambda_1 \pi} \Sigma_{XY} u_1 \end{aligned}$$

Similar calculation yields $\bar{\mu}_2^X = -\sqrt{2/\lambda_1 \pi} \Sigma_{XY} u_1$. The decision rule to classify any observed

value of X to $\bar{\mu}_1^X$ is therefore

$$(\bar{\mu}_1^X)^T X > 0 \quad \text{or} \quad u_1^T \Sigma_{YX} X > 0$$

Since $u_1^T \Sigma_{YX} X$ is a linear combination of X, it also has normal distribution

$$\mathcal{N}\left(0, u_1^T \Sigma_{YX} \Sigma_{XX} \Sigma_{XY} u_1\right)$$

And $(Y, u_1^T \Sigma_{YX} X)$ also have multivariate normal distribution with mean $\mathbf{0}$ and covariance matrix

$$\begin{bmatrix} \Sigma_{YY} & \Sigma_{YX}\Sigma_{XY}u_1 \\ u_1^T\Sigma_{YX}\Sigma_{XY} & u_1^T\Sigma_{YX}\Sigma_{XX}\Sigma_{XY}u_1 \end{bmatrix}$$

The conditional distribution of $Y|u_1^T \Sigma_{YX} X$ is also multivariate normal with mean

$$\mu_{Y|u_1^T\Sigma_{YX}X} = \Sigma_{YX}\Sigma_{XY}u_1(u_1^T\Sigma_{YX}\Sigma_{XX}\Sigma_{XY}u_1)^{-1}u_1^T\Sigma_{YX}X$$

The Y center for $u_1^T \Sigma_{YX} X > 0$ is

$$\begin{aligned} \hat{\mu}_1^Y &= E(Y|u_1^T\Sigma_{YX}X > 0) \\ &= \Sigma_{YX}\Sigma_{XY}u_1(u_1^T\Sigma_{YX}\Sigma_{XX}\Sigma_{XY}u_1)^{-1}E(u_1^T\Sigma_{YX}X \mid u_1^T\Sigma_{YX}X > 0) \end{aligned}$$

Note that $u_1^T \Sigma_{YX} X$ has normal distribution $\mathcal{N}\left(0, u_1^T \Sigma_{YX} \Sigma_{XX} \Sigma_{XY} u_1\right)$, so

$$E(u_1^T\Sigma_{YX}X \mid u_1^T\Sigma_{YX}X > 0) = \sqrt{2/\pi} \cdot \sqrt{u_1^T\Sigma_{YX}\Sigma_{XX}\Sigma_{XY}u_1}$$

Therefore, we have the Y center for $u_1^T \Sigma_{YX} X > 0$ be

$$\begin{aligned} \hat{\mu}_1^Y &= \sqrt{2/\pi} \cdot \sqrt{u_1^T\Sigma_{YX}\Sigma_{XX}\Sigma_{XY}u_1}\ \ \Sigma_{YX}\Sigma_{XY}u_1(u_1^T\Sigma_{YX}\Sigma_{XX}\Sigma_{XY}u_1)^{-1} \\ &= \frac{\sqrt{2/\pi}}{\sqrt{u_1^T\Sigma_{YX}\Sigma_{XX}\Sigma_{XY}u_1}}\Sigma_{YX}\Sigma_{XY}u_1 \end{aligned}$$

Recall that $\bar{\mu}_1^Y = \sqrt{2\lambda_1/\pi}u_1$, to judge if $CV(2) > CV(1)$, one only need to compare

the distance between $\hat{\mu}_1^Y$ and $\bar{\mu}_1^Y$ with distance between $\hat{\mu}_1^Y$ and grand mean 0. By variance and bias decomposition of prediction MSE, when variance is the same, only bias influence the MSE.

After some linear algebra manipulation, we get $||\hat{\mu}_1^Y - \bar{\mu}_1^Y||^2 > ||\hat{\mu}_1^Y||^2$ or $CV(2) > CV(1)$ iff

$$\frac{\sqrt{\lambda_1}}{2} > \frac{u_1^T \Sigma_{YX} \Sigma_{XY} u_1}{\sqrt{u_1^T \Sigma_{YX} \Sigma_{XX} \Sigma_{XY} u_1}}$$

□

Above equation gives the condition of when Gabriel CV method would correctly choose $k = 1$ over $k = 2$. Although the expression is succinct, it's not straight forward to see how the structure of covariance matrix $\Sigma_{XY} \in \mathbb{R}^{(p+q)\times(p+q)}$ affects the performance of Gabriel CV method. Here, assuming covariance matrix has compound symmetric structure where only the matrix dimension $p + q$ and ρ are variables, i.e.

$$\Sigma_{XY} = \begin{pmatrix} 1 & \rho & \cdots & \rho \\ \rho & 1 & \cdots & \rho \\ \vdots & \vdots & \cdots & \rho \\ \rho & \rho & \cdots & 1 \end{pmatrix}$$

we are able to feel what above equation implies for this specific case. Another reason to use the compound symmetric structure covariance matrix is that it's invariant under the permutation of each column vector in the matrix. Given that Gabriel CV method randomly choose p columns as X (q columns as Y), compound symmetric structure insures that the covariance matrix Σ_{XY} always look the same no matter which p columns selected as X.

If we do 2 fold cross-validation in the column, i.e. $p = q$, then the result in Proposition 2 implies that the boundary value of ρ is

$$\rho^* = \frac{1}{p+1}$$

under the compound symmetric structure given above, where Σ_{XY} has $p+q = 2p$ dimension. If $\rho < \rho^*$, then Gabriel CV prefers $CV(1)$ over $CV(2)$ and vice versa. It means the boundary value ρ^* depends on the dimension of Σ_{XY} linearly.

If dimension is $p+q=2$ with $p=q$, then above boundary condition shows the boundary value $\rho^* = 0.5$, while for dimension $p+q=100$, the boundary value $\rho^* = \frac{1}{51}$. However, such value also depends on the ratio of p over q. Also in $p+q=100$ dimension, if we pick $p=2$ and $q=98$ (leave majority of the columns for clustering), the the boundary value $\rho^* \in (\frac{1}{6}, \frac{1}{7})$.

Theorem 2. *Suppose that* $\{(X_i, Y_i)\}_{i=1}^{n+m}$ *is data from a single fold of Gabriel cross-validation, where each* (X, Y) *pair in* $\mathbb{R}^2$ *is an independent draw from a mean-zero multivariate normal distribution with unit marginal variances and correlation* ρ*. In this case, the data are drawn from a single cluster; the true number of clusters is* 1*. If* $|\rho| < 0.5$*, then* $\mathrm{CV}(1) < \mathrm{CV}(k)$*,* $k > 1$ *with probability tending to one as* m *and* n *increase.*

Proof. Given $(X_1, Y_1), \ldots, (X_n, Y_n)$, let $A_k = \{a_1, a_2, ..., a_k\}$ denotes the set of cluster centers from applying k-means to $\{Y_i\}_{i=1}^n$ with parameter k. Because $\{Y_i\}_{i=1}^n$ is symmetric, it's easy to see that the cluster centers are symmetric around 0, i.e. if $a^* \in A_k$ then $-a^* \in A_k$. The k clusters are symmetric in the same sense. Also, if k is odd, then only one cluster contains both negative and positive Y and it's symmetric with center 0. The rest have pure positive or negative intervals; if k is even, then all clusters have pure positive or negative intervals.

Let a_i denotes a cluster center with interval $[b_i, c_i]$. By above argument, either $0 \leq b_i < c_i$ ($b_i < c_i \leq 0$) or $b_i = -c_i$ which corresponding to $a_i = 0$ with k be odd. By symmetry, it's sufficient to only consider case $0 \leq b_i < c_i$ (with $a_i > 0$) and $b_i = -c_i$ (with $a_i = 0$) with $\rho \geq 0$.

From joint normality of X and Y, it follows that $X \mid Y$ is normal with mean ρY. Therefore, the mean value $\bar{X}_i$ of X in cluster with Y center a_i is

$$
\begin{aligned}
\bar{X}_i &= E[X \mid b_i \leq Y \leq c_i] \\
&= E[E(X \mid Y) \mid b_i \leq Y \leq c_i] \\
&= E[\rho Y \mid b_i \leq Y \leq c_i] \\
&= \rho E[Y \mid b_i \leq Y \leq c_i] \\
&= \rho a_i
\end{aligned}
$$

That is, the mean values of X in the k clusters is $\{\rho a_1, \rho a_2, ..., \rho a_k\}$. Because we assign each observation to the closest $\bar{X}_i$, $i = 1, 2, ..., k$, it's easy to see the corresponding interval on X for $\bar{X}_i = \rho a_i$ is $[\frac{\rho(a_{i-1}+a_i)}{2}, \frac{\rho(a_i+a_{i+1})}{2}]$, where a_{i-1} and a_{i+1} are the two adjacent centers to a_i on Y. Note that $[\frac{(a_{i-1}+a_i)}{2}, \frac{(a_i+a_{i+1})}{2}] = [b_i, c_i]$ by the algorithm definition of k-means. The Y center for such interval on X is

$$
\begin{aligned}
\hat{\mu}_i^Y &= E[Y \mid \frac{\rho\,(a_{i-1}+a_i)}{2} \leq X \leq \frac{\rho\,(a_i+a_{i+1})}{2}] \\
&= E[E(Y \mid X) \mid \frac{\rho\,(a_{i-1}+a_i)}{2} \leq X \leq \frac{\rho\,(a_i+a_{i+1})}{2}] \\
&= E[\rho X \mid \frac{\rho\,(a_{i-1}+a_i)}{2} \leq X \leq \frac{\rho\,(a_i+a_{i+1})}{2}] \\
&= \rho E[X \mid \frac{\rho\,(a_{i-1}+a_i)}{2} \leq X \leq \frac{\rho\,(a_i+a_{i+1})}{2}] \\
&= \rho E[X \mid \rho b_i \leq X \leq \rho c_i] \\
&\leq \rho a_i
\end{aligned}
$$

where the equality holds if $b_i = -c_i$ or $\rho = 1$ on the last step. This is because

$$
\begin{aligned}
E[X \mid \rho b_i \leq X \leq \rho c_i] &= E[Y \mid \rho b_i \leq Y \leq \rho c_i] \\
0 \leq \mathtt{b_i} < \mathtt{c_i},\ 0 \leq \rho \leq 1 \quad &\leq E[Y \mid b_i \leq Y \leq c_i] \\
&= a_i
\end{aligned}
$$

The first equation above is because the same marginal distribution of X and Y.

Note that Gabriel CV method predicts Y to be a_i for $X \in [\frac{\rho(a_{i-1}+a_i)}{2}, \frac{\rho(a_i+a_{i+1})}{2}]$, and if $k = 1$ the predicted $Y = 0$ for all X. Hence, to see if $CV(k) > CV(1)$ for $X \in [\frac{\rho(a_{i-1}+a_i)}{2}, \frac{\rho(a_i+a_{i+1})}{2}]$, one only need to see if $a_i - \hat{\mu}_i^Y > \hat{\mu}_i^Y - 0$. Since $\hat{\mu}_i^Y \leq \rho a_i$, it's clear that $CV(k) > CV(1)$ whenever $\rho < 0.5$. This is true to all segment $[\frac{\rho(a_{i-1}+a_i)}{2}, \frac{\rho(a_i+a_{i+1})}{2}] = [\rho b_i, \rho c_i]$ except the one with $b_i = -c_i$, in which $CV(k) = CV(1)$. Since $k > 1$ and at most one segment has $CV(k) = CV(1)$ while the rest have $CV(k) > CV(1)$ if $\rho < 0.5$, it is clear that the overall prediction error $CV(k) > CV(1)$. By symmetry, we can see $CV(k) > CV(1)$ if $|\rho| < 0.5$. The proof is complete. □

4.2 Two Clusters

We will now analyze a simple two-cluster setting, and derive conditions for Gabriel cross-validation to correctly prefer $k = 2$ clusters to $k = 1$.

Proposition 3. *Suppose that $\{(X_i, Y_i)\}_{i=1}^{n+m}$ is data from a single fold of Gabriel cross-validation, where each (X, Y) pair in $\mathbb{R}^2$ is an independent draw from an equiprobable mixture of two multivariate normal distributions with identity covariance. Suppose that the first mixture component has mean (μ^X, μ^Y), and the second has mean $(-\mu^X, -\mu^Y)$, where $\mu^X > 0$ and $\mu^Y > 0$. If $1+2\Phi(\mu^Y)+\frac{2\varphi(\mu^Y)}{\mu^Y} < 4\Phi(\mu^X)$, then $\mathrm{CV}(2) < \mathrm{CV}(1)$ with probability tending to one as m and n increase.*

Proof. There are two clusters G_1 and G_2, where observations from G_1 are distributed as

$$\mathcal{N}\left(\begin{pmatrix}\mu^X\\ \mu^Y\end{pmatrix}, \mathbf{I}\right)$$

and observations from G_2 are distributed as

$$\mathcal{N}\left(\begin{pmatrix}-\mu^X\\ -\mu^Y\end{pmatrix}, \mathbf{I}\right)$$

where $\mu_1^X > 0$ and $\mu_1^Y > 0$. Let G_i be the true cluster where observation i is generated

from, by assumption

$$P(G_i = G_1) = P(G_i = G_2) = 1/2$$

After applying k-means on $\{Y_i\}_{i=1}^n$ with $k = 2$, if n is large enough, we have the estimated centroids $\bar{\mu}_1^Y$ and $\bar{\mu}_2^Y$ be close to $E(Y \mid Y > 0)$ and $\mathrm{E}(Y \mid Y < 0)$, with errors will be of size $O_p(n^{-1/2})$. Here

$$\begin{aligned} E(Y \mid Y > 0) &= E(Y_1 \mid Y_1 > 0) \cdot P(Y_1 > 0) + E(Y_2 \mid Y_2 > 0) \cdot P(Y_2 > 0) \\ &= 2\varphi(\mu^Y) + 2\mu^Y \Phi(\mu^Y) - \mu^Y \end{aligned} \tag{1}$$

where $Y_1 \sim N(\mu^Y, 1)$ and $Y_2 \sim N(-\mu^Y, 1)$, and we used Lemma 3 (Appendix B). Similarly, we have

$$\begin{aligned} E(Y \mid Y < 0) &= E(Y_1 \mid Y_1 < 0) \cdot P(Y_1 < 0) + E(Y_2 \mid Y_2 < 0) \cdot P(Y_2 < 0) \\ &= -2\varphi(\mu^Y) - 2\mu^Y \Phi(\mu^Y) + \mu^Y \end{aligned} \tag{2}$$

where $\varphi()$ and $\Phi()$ are the standard normal probability and cumulative distribution function respectively.

Same as in single cluster case, the classification rule learned from $\{(X_i, \hat{G}_i^Y)\}_{i=1}^n$ variables will be determined according to whether $X > 0$, with the decision boundary be at $0 + O_p(n^{-1/2})$. By symmetry, the CV error for points from G_1 is same as the points from G_2. Because $P(G = G_1) = P(G = G_2) = 1/2$, the CV error can be calculated solely from G_2, that is

$$\begin{aligned} \mathrm{CV}(2) &= \frac{1}{m} \sum_{i=n+1}^{n+m} \|(Y_i - \bar{\mu}_1^Y) 1\{\hat{G}_i^X = 1\}\|^2 + \|(Y_i - \bar{\mu}_2^Y) 1\{\hat{G}_i^X = 2\}\|^2, \quad Y \sim N(-\mu^Y, 1) \\ &= \mathrm{E}[(Y - a)^2 1\{X > 0\}] + \mathrm{E}[(Y + a)^2 1\{X < 0\}] + O_p(m^{-1/2}) + O_p(n^{-1/2}) \end{aligned}$$

With

$$\begin{aligned}
\mathrm{E}[(Y-a)^2 1\{X>0\}] + \mathrm{E}[(Y+a)^2 1\{X<0\}] =& P(\hat{G}_i^X = 1)\cdot E[(Y-a)^2] + P(\hat{G}_i^X = 2)\cdot E[(Y+a)^2] \\
=& [1-\Phi(\mu^X)][var(Y) + (-\mu^Y - a)^2] + \\
& \Phi(\mu^X)[var(Y) + (-\mu^Y + a)^2] \\
=& [1-\Phi(\mu^X)][1 + (\mu^Y + a)^2] + \Phi(\mu^X)[1 + (\mu^Y - a)^2] \\
=& 1 + (\mu^Y + a)^2 - 4a\Phi(\mu^X)\mu^Y
\end{aligned}$$

where a is given by (1).

When $k = 1$, the result is straight forward since the estimated centroid will approximately equal to 0, with error of size $O_p(n^{-1/2})$. The cross-validation error will be

$$\mathrm{CV}(1) = \frac{1}{m}\sum_{i=n+1}^{n+m} \|Y_i - \bar{Y}_n\|^2 = 1 + (\mu^Y)^2 + O_p(m^{-1/2}) + O_p(n^{-1/2}).$$

So if we have $1 + 2\Phi(\mu^Y) + \frac{2\varphi(\mu^Y)}{\mu^Y} < 4\Phi(\mu^X)$, we have $\mathrm{CV}(2) < \mathrm{CV}(1)$ □

We confirm this result with a simulation. We perform 10 replicates for each pair of (μ^X, μ^Y), where both μ^X and μ^Y take value on grid of $[0, 3]$ with step 0.1. In each replicate, we generate 20000 observations from two multivariate normal distributions with identity covariance, where one has mean (μ^X, μ^Y) and the other one has mean $(-\mu^X, -\mu^Y)$. We perform a single 2×2 fold of Gabriel cross-validation and report the times (out of 10 replicates) when $k = 2$ is selected by the algorithm in stead of $k = 1$. Figure 3 shows the frequency $k = 2$ is selected by the algorithm for each pair of (μ^X, μ^Y). The dark spot means high number (close to 10) is selected by the algorithm, which means algorithm very likely will pick $k = 2$ over $k = 1$ for the corresponding (μ^X, μ^Y). While light spot means algorithm prefer $k = 1$ for the corresponding value of (μ^X, μ^Y). We can see the simulation result perfectly align with the theoretical curve (the black line), which separates the $k = 2$ zone from the $k = 1$ zone. It demonstrates that the Gabriel cross-validation works exactly as it suppose to under such setting. The position of dark spots shows that when the two clusters are reasonably apart (not overlapping too heavily) in both dimensions, the Gabriel cross-validation is asymptotically consistent.

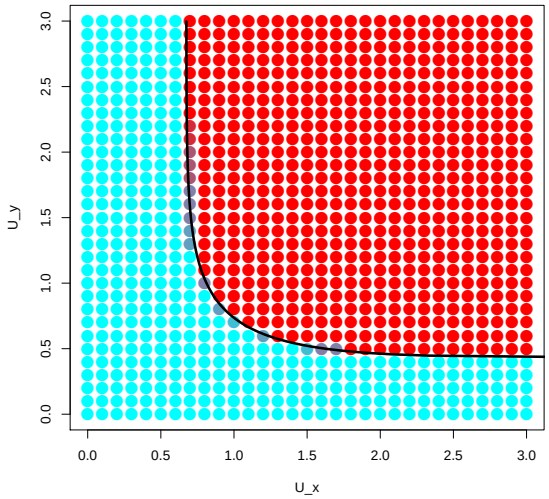


Figure 3: Number of times $k = 2$ is selected out of 10 replicates for each pair of (μ^X, μ^Y). The heat map shows the frequency $k = 2$ is selected by the algorithm, with light means low number (of $k = 2$) is selected and dark color means high number is selected. The black line is the theoretical curve, above which the algorithm suppose to pick $k = 2$ and below which algorithm select $k = 1$.

4.3 Correlation adjustment for Gabriel method

When the correlation between dimensions are high, the proposed Gabriel CV method tends to overestimate the number k. A simply remedy for the high correlation is available if we assume common covariance structure among the k clusters.

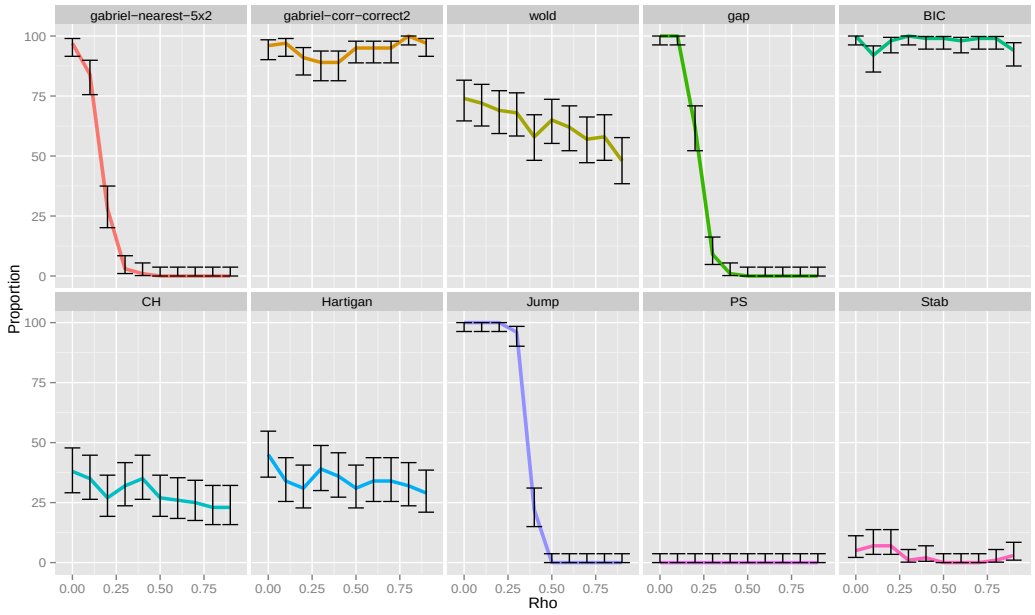
1. Apply Gabriel CV method on the original data $\mathfrak{X}$, get estimated number of cluster $\hat{k}$
2. Estimate the pooled covariance matrix $\hat{\Sigma}$ from the $\hat{k}$ clusters.
3. Let $\hat{\Sigma} = \Gamma\Lambda\Gamma'$ be the Eigendecomposition of $\hat{\Sigma}$, we rescale and rotate the original data $\mathfrak{X}$ to get $\widetilde{\mathfrak{X}} = \mathfrak{X}\Gamma\Lambda^{-1/2}Q$, where Q is a random orthonormal rotation matrix.
4. Apply Gabriel CV method again on the transformed data $\widetilde{\mathfrak{X}}$ to estimate k.

5 Simulation

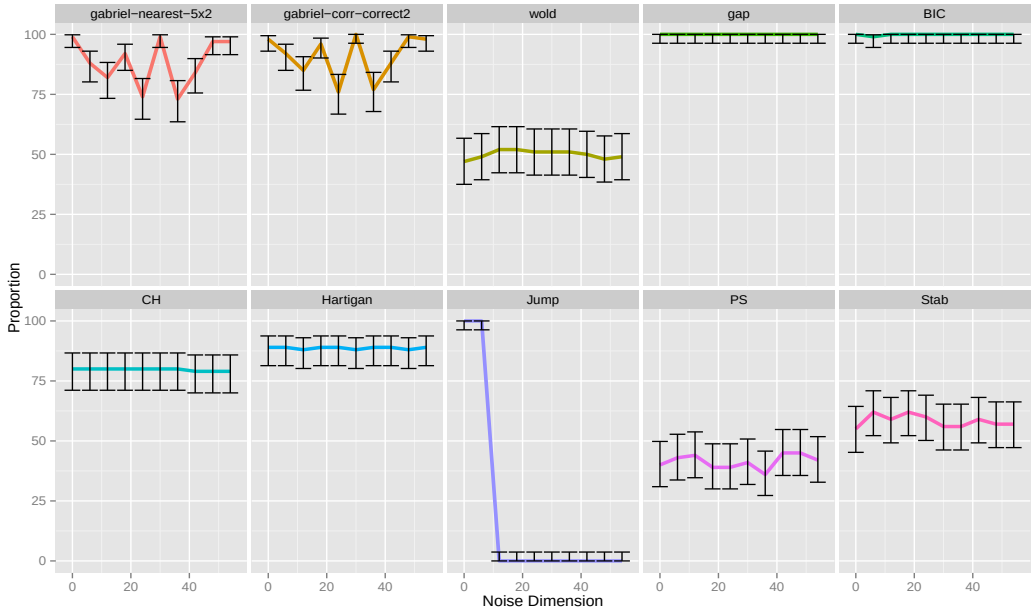
In this section, simulation is performed to evaluate the performance of our proposed methods in locating the "correct" number of clusters. We compare with a basket of existing methods including Gap statistics (Tibshirani et al., 2001), Gaussian mixture model-based clustering (Fraley and Raftery, 2002), CH-index (Caliński and Harabasz, 1974), Hartigan statistics (Hartigan, 1975), Jump method (Sugar and James, 2003), Prediction strength (Tibshirani and Walther, 2005), Bootstrap stability (Fang and Wang, 2012) in following simulation settings. All methods are executed with their default parameter settings. We select $p = q$ and $5-$fold cross-validation in row ($m = \frac{1}{4}n$) as default parameter setting for our proposed Gabriel method. Note that set $p = q$ corresponding to $2-$fold cross-validation in column. Wold method (detailed in Appendix A) and correlation-corrected Gabriel method (section 4.3) are also included for comparison.

Note that in all settings, cluster centers are randomly generated from multivariate normal distribution $\mathcal{N}(\mathbf{0}, \varsigma\mathbf{I})$. All clusters are well-separated, i.e. no overlapping. In fact, any simulated clusters with minimum distance less than one unit was discarded, so there is clear definition of true number of clusters. The parameters ς is chosen such that about half of the random realization were discarded. The idea is borrowed from Tibshirani et al. (2001). The proportion of each method successfully picks the correct k out of 100 simulation trials is reported, along with its corresponding confident interval by Wilson's method (Wilson, 1927).

1. **Correlation between dimensions** – Six clusters in 10 dimensions. Each cluster has 100 or 50 multivariate normal observations with common covariance matrix Σ which has compound symmetric structure with 1 in diagonal and ρ off diagonal. ρ takes value in $\{0, 0.1, ..., 0.9\}$.

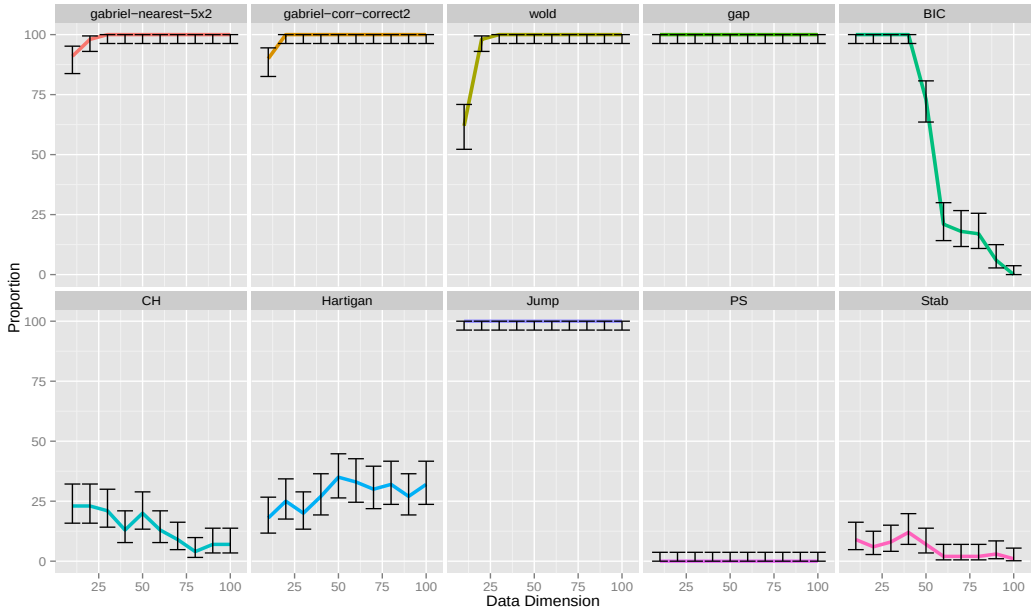


2. **Noise dimensions** — Three clusters in 6 dimensions. Each cluster has 1000 or 500 mean zero multivariate normal observations with identity covariance matrix. We add P dimensions of noise to the data, which is randomly generated from uniform$[0, 1]$. The noise dimension P takes values in $\{0, 6, ..., 54\}$.

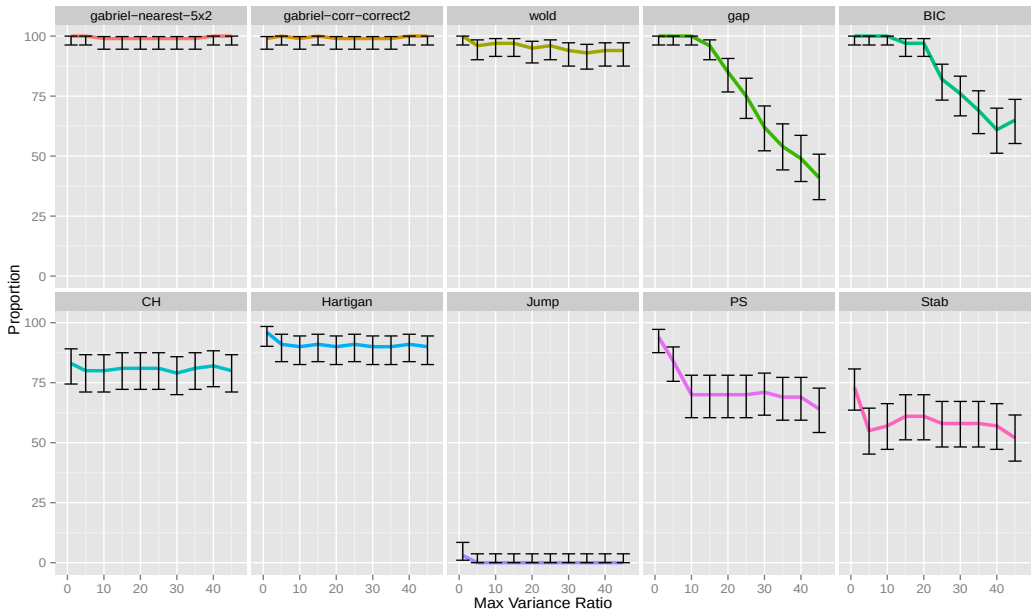


3. **High dimension** — Eight clusters in p dimensions, p takes values in $\{10, 20, ..., 100\}$.

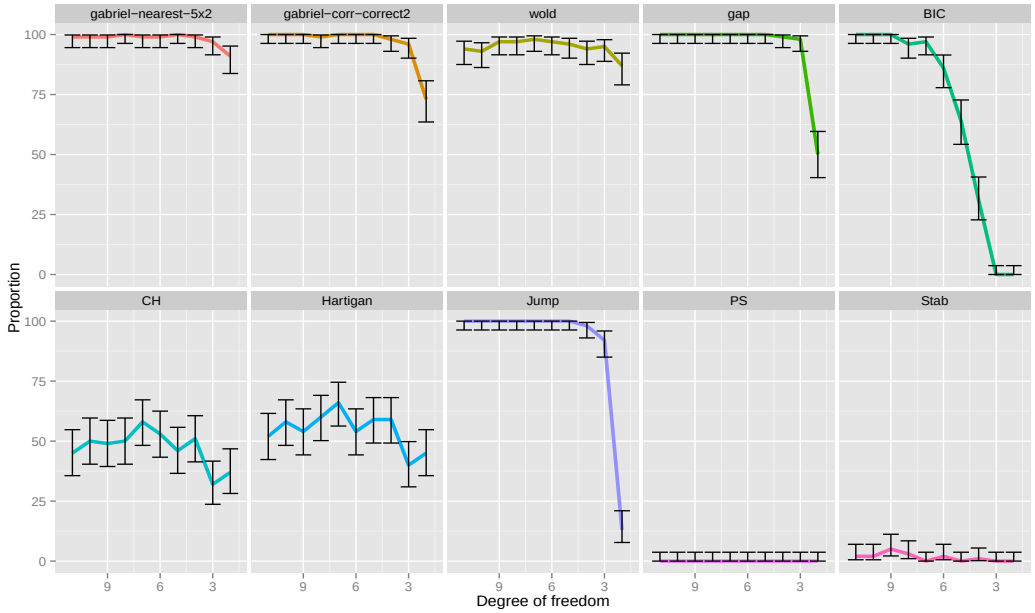
Each cluster has 100 or 50 mean zero multivariate normal observations with identity covariance matrix.



4. **Variance heterogeneity** — Three clusters in 20 dimensions, each has 60 observations in it. Observations are generated from $\mathcal{N}(\mathbf{0}, \sigma_1^2\mathbf{I})$, $\mathcal{N}(\mathbf{0}, \sigma_2^2\mathbf{I})$ and $\mathcal{N}(\mathbf{0}, \sigma_3^2\mathbf{I})$ where $\sigma_1^2 : \sigma_2^2 : \sigma_3^2 = 1 : \frac{1+R}{2} : R$. The maximum ratio R takes values in $\{1, 5, 10, ..., 45\}$



5. **Heavy tail data** — Five clusters in 15 dimensions, each has 80 observations in it. Observations have independent t distribution in each dimension with degree of freedom ν, which takes value in $\{11, 10, ..., 2\}$



From the graph of setting 1, we can see high correlation between dimensions cause problem for most existing methods as well as the proposed Gabriel CV method and Wold method. Higher correlation clearly degenerate most algorithms' ability to find the correct k as data no longer spherical in p-dimension. And the recent proposed methods (Gabriel CV method, Gap statistic, Jump method) seem to be more sensitive to high correlation than those old methods such as CH-index and Hartigan statistics. The only two methods work well in high correlation are the Gaussian model-based BIC method (Fraley and Raftery, 2002) and the correlation-corrected Gabriel method.

Plot of setting 3 shows some methods are insensitive to higher dimension such as Jump method and Gap statistics, while others deteriorate quickly with increasing dimension, most notably the Gaussian model-based BIC method. Note here the data are still generated from mixture of Gaussian. In contrast, our proposed Gabriel CV method as well as its correlation-corrected version actually working better in higher dimension.

Result of setting 4 indicates most previously methods are sensitive to variance heterogeneity among clusters, most notably the Gap statistics and model-based BIC method.

The proposed Gabriel CV method and its correlation-corrected version consistently perform well in estimating k and quite insensitive to variance heterogeneity. The Wold method also performs well in this setting.

The last non-Gaussian setting examines methods' performance in heavy-tail data. With degree of freedom decreases, the tail becomes more flat and the Gaussian assumption becomes more inappropriate. It explains the sharp plunge of Gaussian model-based BIC method in the plot. For most methods, their performances are relatively stable until the tail gets very heavy. In case where degree of freedom is 2, Gap statistics and Jump method dived considerably compare to the Gabriel method and Wold method.

In conclusion, the proposed Gabriel CV method compares well with current state-of-the-art methods and is very robust against variance heterogeneity, high dimension and heavy-tail data. One weakness of the Gabriel CV method is that it's sensitive to the high correlation among dimensions. In such case, its correlation-corrected version will perform well under the assumption of common covariance structure.

6 Real data application

We also applied our proposed method to three real world data sets, two were obtained from the University of California Irvine machine learning repository. The first and third data sets are selected because there are clear number of clusters in those two data sets. The second data set is used as a benchmark data set since it was widely used in literature.

The first one is congress voting data which consists of voting records of 98th United States Congress, 2nd session (Schlimmer, 1987). This data set includes votes for each of the U.S. House of Representatives Congressmen on the 16 key votes identified by the *CQA* (Congressional Quarterly Almanac). For each vote, each Congressman either vote positively "yea" (voted for/paired for/announced for), negatively "nay" (voted against/paired against/announced against) or position unknown "?". We took out those records contain "?" before comparing each algorithm. It results in 232 remaining records, with 124 democrat and 108 republican.

The second data set is the well-known Wisconsin breast cancer data set (Mangasarian et al., 1990). After excluding the records with missing data, this data set consists records

of 683 patients, each with measurements of nine attributes of their biopsy specimens. It is known that there exist at least two groups of patients: 444 patients with benign specimens and 239 patients with malignant specimens.

The third data set is gene expression data of 5 types of brain tumours from de Souto et al. (2008), which contains 42 observations including 10 medulloblastomas, 10 malignant gliomas, 10 atypical teratoid/rhabdoid tumours (AT/RTs), 8 primitive neuroectodermal tumours (PNETs) and 4 normal cerebella. It was originally used by Pomeroy et al. (2002) to show that medulloblastomas are molecularly distinct from other types of brain tumours.

Data was collected with Affymetrix microarrays, which estimates the number of RNA copies found in the cell sample (frozen specimens). Data was preprocessed as following: as a common procedure, all genes with expression level below 10 are set to a minimum level threshold of 10. The maximum threshold is set at $16,000$. Values below or above these thresholds are often unreliable. To remove uninformative genes, for each gene j (attribute), the mean m_j among the samples was calculated. The 10% largest and smallest values are discarded in order to remove extreme values. Based on the mean, following transformation is applied on every value x_{ij} of sample i and gene j

$$y_{ij} = \log_2(x_{ij}/m_j)$$

Genes with expression levels differing by at least l-flod in at least c samples from their mean expression level across the sample were selected. The parameter l and c were chosen so as to produce a filtered data set with at least 10% of the original number of genes (de Souto et al., 2008). Such preprocessing procedure results in 1379 genes selected. After filtering step, data is restored in the original scale (i.e. x_{ij}). So the data set is 42×1379, which serves as a high dimensional example.

Table 1: Number of clusters selected by each algorithm

	Congress Voting	Breast Cancer	Brain Tumours
CH-index	2	2	2
Hartigan	3	3	4
Jump	10	9	1
Prediction strength	2	2	1
Bootstrap stability	2	2	7
Gap	8	10	10
Gaussian-BIC	2	5	2
Gabriel	2	3	5
Gabriel-corr-correct	2	2	5
Wold	2	3	4

All the algorithms executed with their default parameter settings with k ranges from 1 to 10

Since most congressmen vote based on their parties' interest, two parties (Democratic and Republican) represent two clusters in this data set. So the optimal number should be two. Close inspection shows k-means with $k = 2$ separates the two parties very well with the lowest miss-classification error 10.43% ($k = 3$ has 14.78%). Our proposed Gabriel CV method, along with several other algorithms correctly pick $k = 2$ for this data set. Note that CH-index and Bootstrap stability also return $k = 2$, but 2 is the lower bound those methods can select for k. So it's not clear they actually choose $k = 2$ or they hit the lower bound (they would pick $k = 1$ if allowed).

For the breast cancer data, it's known to have 2 clusters based on whether it's benign specimens or malignant specimens. Although the proposed Gabriel method picks $k = 3$, its correlation-corrected version correctly picks $k = 2$. It shows the necessity to correct for the high correlation among data features. However, for this data $k = 3$ is also an acceptable answer, as Fujita et al. (2014) noticed that the malign group is quite heterogeneous and can be further clustered into at least two subgroups.

The brain tumours gene expression data is a typical high-dimension data with $p > n$. It is clear that the correct number of clusters should be 5, the number of brain tumour types. Indeed, k-means results with $k = 5$ show that the 4 normal human cerebella are well separated from brain tumours as they form a single cluster. Medulloblastomas, malignant gliomas and AT/RT tumours also form 3 respective clusters, which means they can be separated from each other. Such results are similar to those found in Pomeroy et al. (2002),

whose analysis is based on the first 3 principle components of the original data. From table 1 we can see only our proposed Gabriel method (as well as its correlation-corrected version) correctly selects $k = 5$, underline how difficult it was to pick the right k when dimension is high.

7 Discussion

In this paper, we proposed a novel approach to estimate the number of clusters. The intuition behind our proposed methods is to transfer the unsupervised learning problem into supervised learning problem via novel form of cross validation. Such approach is quite different from previous methods which utilize the within/between cluster dispersion or stability criterion for selecting the optimal k. Our method utilizes the connection between different dimensions (columns) of data through the uniqueness of each cluster center. We proved the self-consistency for our proposed Gabriel CV method as well as its asymptotic property with Gaussian noise, and showed the robustness of our method by simulation. Our method has very good performance in our limited simulation settings and real data application, and clearly the superior one when data has heterogeneous variance or is heavy-tailed.

Besides no strong modeling assumption is required, our proposed method is robust against data set with variance heterogeneity, unequal number of observations, non-Gaussian noise and high-dimension. Such robustness is important in practice because for any given data, it's hard to tell whether its clusters have different number of observations, is the variance equal for each cluster, or what underlying noise distribution it has. The weakness of our proposed method is that its theory assumes only week correlation between "predictor" columns and "response" columns. In practice, many data sets don't exhibit high correlations between columns, where our proposed method can be safely applied. In case the high correlation does exist, the correlation-corrected version of Gabriel CV method in section 4.3 can be used if we can assume common covariance structure. Other procedure such as leave out most columns for clustering may also be used to reduce its effect. However, the theory for such procedure has not been fully developed and it could be the future research topic.

Another situation where our proposed method cannot be directly used is when the clusters are non-convex. In such situation, k-means itself doesn't work well, for example two concentric circles share the same cluster center (Hastie et al., 2009). However, it's possible that our proposed Gabriel CV method can be used on the transformed data set. In the concentric circles case where spectral clustering is appropriate, our proposed method can be applied on the eigenvector subspace of the graph Laplacian matrix inside the spectral clustering algorithm to find the optimal k. This can also be the future research topic.

APPENDIX

A Wold CV estimation

- For each $k = 1, 2, ..., k_{max}$

  1. Randomly draw some entries in $\mathbf{X}$ missing, keep those hold-out values in vector V_{true}
  2. Impute the missing values with column mean or 0, denote the imputed data as $\mathbf{X}_{new}$
  3. Apply the iterative procedure below until converge or stopping criteria reached
     - Apply K-mean on data set $\mathbf{X}_{new}$ with parameter k
     - Substitute each observation in $\mathbf{X}_{new}$ by its nearest center, get new data $\mathbf{X}^c_{new}$ ($\mathbf{X}_{new}$ keep the same)
     - Replace (impute) those imputed values in $\mathbf{X}_{new}$ with the corresponding entries in $\mathbf{X}^c_{new}$
     - Calculate the difference between the old and newly imputed values, check whether or not they coincide (converge)
  4. Obtain the last imputed entry values of converged $\mathbf{X}_{new}$, denote it by $V_{converge}$
  5. Calculate the prediction error $Error_k = ||V_{true} - V_{converge}||^2$

- For each CV folder, repeat above procedure and obtain the $Error_k$ for each k
- Average $Error_k$ across all folders for each k, and then select the k corresponding to the minimum average $Error_k$

B Technical Lemmas

Lemma 3. *If Z is a standard normal random variable, then*

$$\mathrm{E}(Z \mid a < Z < b) = -\frac{\varphi(b) - \varphi(a)}{\Phi(b) - \Phi(a)}$$

and

$$\mathrm{E}\{(Z-\delta)^2 \mid a < Z < b\} = \delta^2 + 1 - \frac{(b-2\delta)\varphi(b) - (a-2\delta)\varphi(a)}{\Phi(b) - \Phi(a)}$$

for all constants a, b, and δ, where $\varphi(z)$ and $\Phi(z)$ are the standard normal probability density and cumulative distribution functions. These expressions are valid for $a = -\infty$ or $b = \infty$ by taking limits.

Proof. We will derive the expression for the second moment. Integrate to get

$$\begin{aligned}\mathrm{E}[(Z-\delta)^2 1\{Z < b\}] &= \int_{-\infty}^{b} (z-\delta)^2 \varphi(z)\, dz \\ &= (\delta^2+1)\Phi(b) - (b-2\delta)\varphi(b).\end{aligned}$$

Now,

$$\mathrm{E}\{(Z-\delta)^2 \mid a < Z < b\} = \frac{\mathrm{E}[(Z-\delta)^2 1\{Z < b\}] - \mathrm{E}[(Z-\delta)^2 1\{Z < a\}]}{\Phi(b) - \Phi(a)}.$$

□

Lemma 3 has some important special cases:

$$\begin{aligned}\mathrm{E}\{Z \mid Z > 0\} &= 2\varphi(0) = \sqrt{2/\pi}, \\ \mathrm{E}\{(Z-\delta)^2 \mid Z > 0\} &= \delta^2 + 1 - 4\delta\varphi(0), \\ \mathrm{E}\{(Z-\delta)^2 \mid Z < 0\} &= \delta^2 + 1 + 4\delta\varphi(0).\end{aligned}$$

References

Ben-Hur, A., Elisseeff, A., and Guyon, I. (2001). A stability based method for discovering structure in clustered data. In *Pacific symposium on biocomputing*, volume 7, pages 6–17.

Caliński, T. and Harabasz, J. (1974). A dendrite method for cluster analysis. *Communications in Statistics-theory and Methods*, 3(1):1–27.

Chiang, M. M.-T. and Mirkin, B. (2010). Intelligent choice of the number of clusters in k-means clustering: an experimental study with different cluster spreads. *Journal of classification*, 27(1):3–40.

de Souto, M. C., Costa, I. G., de Araujo, D. S., Ludermir, T. B., and Schliep, A. (2008). Clustering cancer gene expression data: a comparative study. *BMC bioinformatics*, 9(1):497.

Fang, Y. and Wang, J. (2012). Selection of the number of clusters via the bootstrap method. *Computational Statistics & Data Analysis*, 56(3):468–477.

Fraley, C. and Raftery, A. E. (2002). Model-based clustering, discriminant analysis, and density estimation. *Journal of the American Statistical Association*, 97(458):611–631.

Fujita, A., Takahashi, D. Y., and Patriota, A. G. (2014). A non-parametric method to estimate the number of clusters. *Computational Statistics & Data Analysis*, 73:27–39.

Gabriel, K. R. (2002). Le biplot–outil d'exploration de données multidimensionelles. *Journal de la Société Francaise de Statistique*, 143:5–55.

Hartigan, J. A. (1975). *Clustering Algorithms*. Wiley.

Hastie, T., Tibshirani, R., and Friedman, J. (2009). *The Elements of Statistical Learning: Data Mining, Inference, and Prediciton*. Springer Series in Statistics. Springer, 2nd edition.

Jain, A. K. (2010). Data clustering: 50 years beyond k-means. *Pattern recognition letters*, 31(8):651–666.

Jain, A. K., Murty, M. N., and Flynn, P. J. (1999). Data clustering: a review. *ACM computing surveys (CSUR)*, 31(3):264–323.

Mangasarian, O. L., Setiono, R., and Wolberg, W. (1990). Pattern recognition via linear programming: Theory and application to medical diagnosis. *Large-scale numerical optimization*, pages 22–31.

Owen, A. B. and Perry, P. O. (2009). Bi-cross-validation of the svd and the nonnegative matrix factorization. *Ann. Appl. Stat.*, 3(2):564–594.

Perry, P. O. (2009). *Cross-Validation for Unsupervised Learning*. PhD thesis, Stanford University.

Pollard, D. (1981). Strong consistency of k-means clustering. *Ann. Stat.*, 9(1):135–140.

Pollard, D. (1982). A central limit theorem for k-means clustering. *Ann. Probab.*, 10(4):919–926.

Pomeroy, S. L., Tamayo, P., Gaasenbeek, M., Sturla, L. M., Angelo, M., McLaughlin, M. E., Kim, J. Y., Goumnerova, L. C., Black, P. M., Lau, C., et al. (2002). Prediction of central nervous system embryonal tumour outcome based on gene expression. *Nature*, 415(6870):436–442.

Schlimmer, J. C. (1987). Concept acquisition through representational adjustment.

Sugar, C. A. and James, G. M. (2003). Finding the number of clusters in a dataset. *Journal of the American Statistical Association*, 98(463).

Tibshirani, R. and Walther, G. (2005). Cluster validation by prediction strength. *Journal of Computational and Graphical Statistics*, 14(3):511–528.

Tibshirani, R., Walther, G., and Hastie, T. (2001). Estimating the number of clusters in a data set via the gap statistic. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 63(2):411–423.

Wang, J. (2010). Consistent selection of the number of clusters via crossvalidation. *Biometrika*, 97(4):893–904.

Wilson, E. B. (1927). Probable inference, the law of succession, and statistical inference. *Journal of the American Statistical Association*, 22(158):209–212.

Wold, S. (1978). Cross-validatory estimation of the number of components in factor and principal components models. *Technometrics*, 20:397–405.