
A statistical test for learning the k in k -means

Greg Hamerly

Department of Computer Science and Engineering
University of California, San Diego
La Jolla, California 92093-0114
ghamerly@cs.ucsd.edu

Abstract

Practical machine learning requires that the practitioner choose a model and some parameters for the model. In data clustering with k -means, the choice of k reflects prior knowledge about the structure of the dataset to be clustered. However, having to choose k is not always easy or obvious, and true learning should learn k as well as the parameters for each cluster. In this paper we develop a new way of choosing k based on a χ^2 test of the variance of clusters. The simple idea is that using two centers to represent a true cluster that looks Gaussian is bad. Our test builds on the framework of [?], but is a more flexible test that produces better results. We perform synthetic and real-world experiments to illustrate the performance of our algorithm.

1 Introduction

Data clustering is the task of dividing a set of data points into several partitions, where the points in each partition are similar to themselves, and different from points in other clusters. Data clustering is a useful tool for data mining, compression, unsupervised learning, probability estimation, and other tasks in machine learning and statistics. However, most clustering algorithms require the practitioner to provide the number of clusters (called k), and there is not always a clear answer on what k should be. Figure 1 shows an example where k has been improperly chosen. Thus, choosing k can be a somewhat ad-hoc decision based on prior knowledge, assumptions, and practical experience.

Several algorithms have been proposed to determine k automatically, based on model complexity penalties like the Bayesian information criterion (BIC) [?], or based on a description length metric [?]. Both of these are wrapper methods around clustering algorithms and rely on splitting and/or merging rules for centers to grow/shrink k as the algorithm proceeds. Other research in agglomerative clustering algorithms suggests choosing k based on the “stability” of the merging tree (dendrogram) of distances between points. This too requires some prior knowledge. However, using these methods is not any more intuitive than having to choose k . This work builds on work by [?] by providing a more intuitive splitting rule based on a statistical test, the χ^2 test.

Our method is based on a simple observation: when two centers are used to describe one cluster (i.e. points drawn from a Gaussian distribution), the centers form a poor description of that data (see Figure 2), because two centers are being used to describe one true

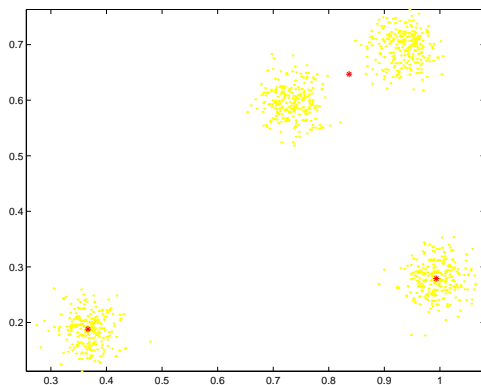


Figure 1: This shows a dataset (yellow dots) with 3 final centers (red dots) found by k -means, where k has been improperly chosen by the user. We can see that the center closest to the top should be split into two to fit the two clusters around it, while the other two centers should not be split. This intuition is based on the assumption that true clusters should be Gaussian-shaped.

cluster. The most common center-based clustering algorithms (e.g. k -means and Gaussian Expectation-Maximization) assume that true clusters have approximately Gaussian distributions. Intuition tells us that only one center should be used to describe a true cluster. In this paper we describe the algorithm framework for learning k , we develop the statistical test for deciding whether to split a center into two centers, and we provide experiments on real data to illustrate the behavior of our algorithm.

2 The X -means algorithm

Pelleg and Moore [?] proposed a regularization framework for learning k , which they call X -means. The algorithm scores each clustering model based on the Bayesian Information Criterion (BIC) [?]:

$$BIC(M) = \mathcal{L}(D) - \frac{p}{2} \log R$$

where $\mathcal{L}(D)$ is the log-likelihood of the dataset according to model M , and p and R are the model complexity parameters. Then they choose the model with the best BIC score.

Along with deciding between models, the algorithm must also be able to search through the space of values for k efficiently. Given lower and upper bounds for k , called k_{\min} and k_{\max} , and starting with $k = k_{\min}$,

1. Run k -means on the entire data set.
2. Split each “parent” center into two “children” centers.
3. Run 2-means on each pair of children on only the data belonging to their parent.
4. For each parent, if $BIC(\text{parent}) > BIC(\text{children})$, then keep the parent. Otherwise, replace the parent with its children.

This process is repeated until $k = k_{\max}$ or no more children are added. The algorithm splits a parent by choosing a random direction vector and placing two children at equal distances away from the parent on that random vector. The distance from the parent is proportional to the variance of the parent’s data. Additionally, in their implementation Pelleg and Moore

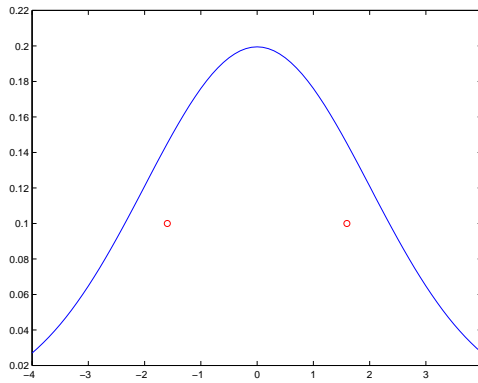


Figure 2: Two k -means centers (red circles) in their final positions for clustering data drawn from a univariate Gaussian distribution (pdf represented by the blue line). This is the situation we want to avoid. Intuitively, only one center should be used to represent this data.

provide a key parameter `num_splits`, which determines how many times the entire loop above is run. This parameter is misleading and determines how much of the search space to examine and can drastically change the behavior of the algorithm, as we will see.

Pelleg and Moore compared X -means with a BIC-scored version of k -means where the k -means algorithm was run for several reasonable values of k , choosing the one with the highest BIC score. They showed that X -means performed better than k -means at finding the low-distortion clusterings (according to the k -means quality metric of within-cluster variance). However, they also showed that X -means tended to underestimate the true number of clusters, and that BIC-scored k -means with a parameter sweep of k did a better job of estimating k than did X -means.

The X -means algorithm is also efficient with respect to the size of the dataset, because it uses a data structure called the k d-tree and knowledge about the hard assignment of k -means to ignore certain regions of the data, without losing any accuracy. This efficiency is good, but our primary concern is choosing k .

3 A new test for splitting centers

Following the example of Pelleg and Moore, we have developed a new decision criterion that is simpler and more intuitive than the BIC. This criterion makes the assumption that each true cluster is elliptical and distributed according to a multivariate Gaussian distribution, and that only one center should be used for each true cluster. These are appropriate assumptions for k -means clustering. In fact, the assumptions are more loose than those imposed by k -means, since k -means clustering assumes spherical clusters.

The intuition we propose is that if a cluster is distributed according to a single Gaussian distribution, we should be able to detect if two centers are attempting to model the cluster based on the variance of each center. We will show that we can know the minimum expected variance of two k -means centers clustering one Gaussian. Using this knowledge, we can develop a statistical test to determine when to split centers in X -means.

Theorem 1 *Given a univariate Gaussian distribution with mean μ and variance σ^2 , the minimum k -means variance for $k = 2$ centers occurs when $c_1 = \mu - \frac{2\sigma}{\sqrt{2\pi}}$ and $c_2 =$*

$\mu + \frac{2\sigma}{\sqrt{2\pi}}$. The variance around each center is $\text{Var}(c_x) = \sigma^2(1 - \frac{2}{\pi})$ for a total variance of $\text{Var}(c_1) + \text{Var}(c_2) = 2\sigma^2(1 - \frac{2}{\pi})$.

For the proof of this theorem, please refer to Appendix A. Now that we know the minimum variance for two centers in a Gaussian distribution with mean μ and covariance Σ , we can form a statistical test to determine whether to split a center into two children. We use the χ^2 distribution, where

$$\chi^2 = \sum_{i=1}^n \frac{z_i^2 - \bar{z}}{\sigma^2}$$

forms a χ^2 value with $n - 1$ degrees of freedom (since we use one degree of freedom to compute \bar{z}). We take σ^2 from Equation 2. Our test hypotheses are:

- $H_0: \sigma^2 \geq \sigma_0^2$
This is the null hypothesis that says that the variance of the two children centers (σ^2) is no better than the variance of two centers on a similar Gaussian distribution (σ_0^2). If we accept H_0 , then we should not split the center in question into two centers, because the test does not support it.
- $H_A: \sigma^2 < \sigma_0^2$
This is the alternative hypothesis which says that σ^2 is significantly better (smaller) than the variance of two centers on a similar Gaussian distribution (σ_0^2). Accepting H_A means we should split the center in question into two centers, because they improve the variance more than it would have two centers in one cluster.

Then we choose the significance level of the test, α , which is the probability with which we will make a mistake (i.e. reject H_0 when we should not reject it). To perform the test, we compute χ^2 and compare it with $\chi_{\alpha, n-1}^2$. If $\chi^2 < \chi_{\alpha, n-1}^2$, then we reject H_0 and accept H_A , which means we should split the center into two.

A benefit of using this statistical test (over simply comparing the values of two variances) is that when the number of sample points is small, the test tends to reject the alternative hypothesis. In other words, this test is able to find cluster structure at many levels of resolution in the data, but when the cluster size is too small (too few data points), the test becomes less confident. Therefore it will not split clusters with a very small number of points, which would happen if we were simply comparing two variances. The statistical test lends confidence to our predictions.

3.1 Adapting to multivariate data

We have shown the best way to represent one Gaussian with two centers and to perform a test of significance on an empirical measurement. Now we must adapt the to the case of multivariate data. For this, we consider the statistic of the distortion of the dataset $X = x_i, 1 \leq i \leq n$ belonging to a center c :

$$\text{dist}(X) = \sum_{i=1}^n \|x_i - c\|^2 \tag{1}$$

This is the k -means metric of distortion for data belonging to a center. It is similar to the univariate variance, however we use the Euclidean norm $\|x\| = \sqrt{\sum_d x_d^2}$ to take the length of multivariate data.

We claim without proof that the minimum distortion for one of two centers in a multivariate

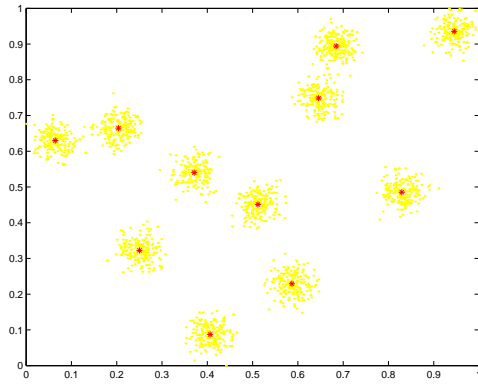


Figure 3: Here the X -means algorithm using our statistical test has found the correct number of centers and placed the centers in the right locations. Here we let X -means search for the correct k starting from $k = 1$, and we use $\alpha = 0.01$.

Gaussian distribution with mean μ and covariance Σ is

$$\text{Var}(c_x) = \frac{1}{2} \sum_{i=1}^d \lambda_i - \max_j \lambda_j \left(\frac{1}{2} - \frac{2}{\pi} \right) \quad (2)$$

where λ_i is the i th eigenvalue of Σ . This allows us to adapt the test to the multivariate case; we can use the same statistical test as before, but on multivariate data.

When we use the X -means algorithm, we must make a decision of *how* to split a parent center into two children – that is, how they will be initialized. Pelleg and Moore choose a *random* direction vector that goes through the parent center, and place the two children on opposite sides of the parent, equally far apart along that vector, with distance proportional to the distortion of the parent. Our splitting method is more intuitive. We know the eigenvectors v_j and eigenvalues λ_j of the data’s covariance, Σ . The direction of largest covariance is along v_m , where $m = \arg \max_j \lambda_j$. In the case of two centers modeling the same Gaussian, the minimum-variance solution will place both centers along v_m , giving the center placements

$$c_1 = \mu - \sqrt{\frac{2\lambda_m}{\pi}} \quad c_2 = \mu + \sqrt{\frac{2\lambda_m}{\pi}}$$

Since λ_m is the variance in the direction of v_m . Therefore, finding the eigenvalues and eigenvectors of Σ gives the σ_0 used in the statistical test for splitting a center, and it also gives good initial positions for the children centers, which allows k -means to perform less update iterations.

4 Experiments

Figure 3 shows an example of the X -means algorithm with our statistical test, which correctly finds the 11 true clusters. For this experiment we started with one center located at the centroid of the data set, and ran the X -means algorithm with our statistical test for splitting centers. For the value of α , we used 0.01. We have rarely had to adjust the value of α .

Figure 4 shows the results on a dataset of 1,037 datapoints in 2 dimensions which represents a trace of the execution of the program `gzip`. The original dataset had 8,163 dimensions.

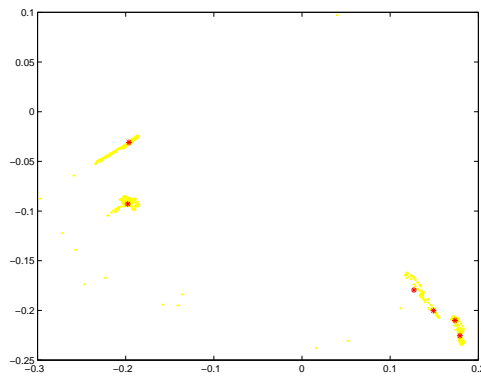


Figure 4: Here the X -means algorithm using our statistical test has found $k = 6$ clusters in the dataset `gzip`, using the parameter $\alpha = 0.01$ and starting from one center and splitting. This dataset has 2 dimensions (using random linear projection to reduce the dimension from 8,163), and 1,037 datapoints.

Table 1: The number of centers that Pelleg and Moore’s X -means algorithm found in the 2-dimensional dataset `gzip`, as a function of the parameter `num_splits` given to the algorithm. The number $k = 50$ appears to be a hard limit in the software for the dataset. Note that it is not clear how to pick the value of `num_splits` based on this table.

<code>num_splits</code>	1	2	3	4	5	6	7	8	9	10
k found by P&M	1	2	3	5	9	14	24	36	50	50

Table 2: The number of centers that X -means with our statistical test found in the same 2-dimensional dataset. It is easy to pick a good value of α , since in most statistical tests we want fairly high confidence (low α). In practice, we would recommend choosing $\alpha \leq 0.05$.

α	0.001	0.01	0.02	0.03	0.04	0.05	0.1	0.25
k found by χ^2	6	6	7	8	10	10	28	44

Each dimension represents a basic block of execution from the program’s binary image. A value x in dimension y means that basic block y was executed y times in a given timeslice of execution. Each datapoint represents a timeslice of 100 million executed instructions. We reduce the dimension of the data drastically by using a random linear projection to 2 dimensions, as in [?]. After this random projection, we use X -means to cluster the data and to find k .

In Tables 1 and 2 we see that depending on the parameter given to the algorithms (for Pelleg and Moore, it is `num_splits`, for our test it is α), we can obtain a different number for k . However, it is far more intuitive how to choose α than how to choose `num_splits`, since the former is the standard statistical test parameter, while the latter is a heuristic which must reflect some notion of previous knowledge of the data. Note that in this dataset, there is no “correct” value of k , so we cannot compare the clustering results of the two methods.

5 Discussion and conclusions

We have shown that a statistical test based on the distortion of a data set can be used to determine whether that data should be split into 2 clusters, or should remain as one. This statistical test is very intuitive, since its purpose is to determine whether two split centers are representing one cluster (in which case we do not want to keep the split), or if they are actually representing more than one cluster (in which case we do want to keep the split). This test takes as a parameter the confidence level α , which can be set in the same way as a typical χ^2 test. The system works well on real 2-d datasets at finding the correct k , and the locations of the true cluster centers.

References

A Proof of theorem 1

Assume we have one univariate Gaussian probability distribution, centered at μ with variance σ^2 . The probability of x according to this Gaussian is

$$p(x; \mu, \sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp \frac{-(x - \mu)^2}{2\sigma^2}$$

Now if we have a point $c > \mu$ and we want to measure the variance of data generated from the Gaussian from this new point, we want to calculate the expectation of the squared distance to the point c :

$$\begin{aligned} E[(x - c)^2] &= 2 \int_{\mu}^{\infty} (x - c)^2 p(x; \mu, \sigma^2) dx \\ &= 2 \int_{\mu}^{\infty} (x - c)^2 \frac{1}{\sqrt{2\pi\sigma^2}} \exp \frac{-(x - \mu)^2}{2\sigma^2} dx \end{aligned}$$

Note that here we have made the assumption that we want the variance with respect to only half of the Gaussian (which is why we integrate with lower bound μ rather than $-\infty$). Thus we multiply the integral by two, to re-normalize $p(x; \mu, \sigma^2)$.

Using the following substitutions,

$$\begin{aligned} y &= x - \mu & z &= \mu - c \\ dy &= dx & m &= \frac{1}{\sqrt{2\pi\sigma^2}} \end{aligned}$$

we obtain the following equations:

$$\begin{aligned} E[(x - c)^2] &= 2m \int_0^{\infty - \mu} (y + \mu - c)^2 \exp \frac{-y^2}{2\sigma^2} dy \\ &= 2m \int_0^{\infty - \mu} (y + z)^2 \exp \frac{-y^2}{2\sigma^2} dy \\ &= 2m \int_0^{\infty - \mu} (y^2 + 2yz + z^2) \exp \frac{-y^2}{2\sigma^2} dy \\ &= 2m \left[\int_0^{\infty - \mu} y^2 \exp \frac{-y^2}{2\sigma^2} dy + 2z \int_0^{\infty - \mu} y \exp \frac{-y^2}{2\sigma^2} dy + \right. \\ &\quad \left. z^2 \int_0^{\infty - \mu} \exp \frac{-y^2}{2\sigma^2} dy \right] \end{aligned}$$

Taking each term in turn and using standard definitions:

$$\begin{aligned}
 2m \int_0^{\infty-\mu} y^2 \exp \frac{-y^2}{2\sigma^2} dy &= \sigma^2 \\
 4zm \int_0^{\infty-\mu} y \exp \frac{-y^2}{2\sigma^2} dy &= 4zm(-\sigma^2 \exp \frac{-y^2}{2\sigma^2}) \Big|_0^{\infty-\mu} \\
 &= \frac{4(\mu - c)}{\sqrt{2\pi\sigma^2}} \sigma^2 \\
 &= \frac{4\sigma(\mu - c)}{\sqrt{2\pi}} \\
 2z^2m \int_0^{\infty-\mu} \exp \frac{-y^2}{2\sigma^2} dy &= z^2 \\
 &= (\mu - c)^2
 \end{aligned}$$

Now we can recombine the terms:

$$\begin{aligned}
 E[(x - c)^2] &= \sigma^2 + \frac{4\sigma(\mu - c)}{\sqrt{2\pi}} + (\mu - c)^2 \\
 &= \sigma^2 + (\mu - c)^2 + \frac{4\sigma(\mu - c)}{\sqrt{2\pi}}
 \end{aligned}$$

This is then the variance around point c for half of a Gaussian.

Now we want to find the c for which this expectation is minimized. Taking the derivative with respect to c , we obtain:

$$\frac{d}{dc} E[(x - c)^2] = 2c - 2\mu - \frac{4\sigma}{\sqrt{2\pi}}$$

Setting this to zero gives the minimum solution:

$$c = \mu + \frac{2\sigma}{\sqrt{2\pi}}$$

Now if we place the minimum solution for c back into the expected variance, then we get:

$$\begin{aligned}
 \min_c E[(x - c)^2] &= \sigma^2 + \left(\mu - \left(\mu + \frac{2\sigma}{\sqrt{2\pi}} \right) \right)^2 + \frac{4\sigma \left(\mu - \left(\mu + \frac{2\sigma}{\sqrt{2\pi}} \right) \right)}{\sqrt{2\pi}} \\
 &= \sigma^2 + \frac{4\sigma^2}{2\pi} - \frac{8\sigma^2}{2\pi} \\
 &= \sigma^2 - \frac{2\sigma^2}{\pi} \\
 &= \sigma^2 \left(1 - \frac{2}{\pi} \right)
 \end{aligned}$$

This is the minimum expected variance for one center which is modeling half of a Gaussian. For two centers which model both halves, the minimum expected variance is simply twice that, or $2\sigma^2 \left(1 - \frac{2}{\pi} \right)$. ■