Lab 3 - Parallelizing k-means
Stat 215A, Fall 2017

3032101346

October 23, 2017

1 Introduction

There is a zoo of methods for determining the appropriate number of groups (hereafter referred to as k) to use in clustering a dataset. Claims of an 'optimal' method for choosing k should be met with suspicious eyes, for optimality refers to a specific objective, which varies from problem to problem. When the clusters are used only for subsequent prediction, cross-validation often yields results; however, when the clusters are to be interpreted as distinct groups with strong intra-cluster cohesion, predictive accuracy is no longer the most relevant measure.

If all of the points belong to a single cluster ($k = 1$), then the data is, in a sense, maximally compressed; conversely, if all of the points belong to their own cluster ($k = n$), then the total error in the cluster assignments is zero (if the error does not penalize large k). Either extreme usually defeats the purpose of clustering, and standard methods for selecting k in k-means trade off between the two. The *elbow method* adds clusters until it reaches rapidly diminishing gains in explained variance. The *silhouette method* summarizes the silhouette plots of various clusterings, say by the average value, choosing k optimally. *Information criteria* such as AIC and BIC can be applied by viewing k-means as a scaling limit of isotropic Gaussian mixture modeling. Similarly, scaling limits of Bayesian nonparametric methods [2] such as DP-means lead to their own penalties on k, whereas fully Bayesian approaches treat k as fixed but unknown and hence model the data as a mixture of finite mixtures [3]. The gap-statistic method [4] maximizes the gap in the sum of squares objective relative to a reference distribution. Density-based clustering algorithms such as DBSCAN automate the selection process and can discover non-linearly separable clusters, whereas visualization-based methods (such as a dendrogram) more directly loop in the analyst in determining k.

Ben-Hur et al. [1] propose choosing k based on the likelihood of clusterings on two subsamples producing similar results. One measure of similarity—called correlation—is based on whether two data points (present in both subsamples) co-occur in both clusterings. Specifically, let $\ell^{(1)}$ and $\ell^{(2)}$ denote the labelings on the overlap on the two subsamples, and define

$$\texttt{corr}\left(\ell^{(1)}, \ell^{(2)}\right) = \frac{\sum_{ij} 1_{\ell_i^{(1)}=\ell_j^{(1)} \,\&\, \ell_i^{(2)}=\ell_j^{(2)}}}{\sqrt{\left(\sum_{ij} 1_{\ell_i^{(1)}=\ell_j^{(1)}}\right)\left(\sum_{ij} 1_{\ell_i^{(2)}=\ell_j^{(2)}}\right)}} \tag{1}$$

Note that computing the denominator can be further simplified since $\sum_{ij} 1_{\ell_i^{(1)}=\ell_j^{(1)}} = \sum_k \left(\sum_i 1_{\ell_i^{(1)}=k}\right)^2$.

In this lab, we look at the stability-based method of [1] for clustering a Dialect Survey conducted by Bert Vaux, containing the responses of 45152 individuals across America to 68 questions regarding features of spoken language. We cluster subsamples of the 45152×468 one-hot encoded response matrix `ling` and repeatedly calculate the correlation measure of similarity between pairs of labelings ($N = 100$ times for each value of k). We then inspect the empirical distribution of the similarity measure to see how large it is on average and how much it varies across iterations.

2 Implementation

Ben-Hur et al. [1] refer to the algorithm described in Section 1 as the model explorer algorithm. We implement the model explorer algorithm in the `code` directory in the repository containing this report. The script `parallel.R` runs the algorithm in parallel for each $k \in \{2, \ldots, 10\}$. Each subsample uses 80% of the `ling` data. The inner loop over $N = 100$ iterations is entirely contained in `ClusterSim.R`. We ran the script `parallel.R` on the SCF cluster by submitting `job.sh`. The job use 9 CPUs and terminated in under 2 hours.

Getting the job to run in a reasonable amount of time required care for the computational bottlenecks. The built in k-means algorithm took about $1.7k$ seconds to run on one subsample of $q = \lfloor 0.8n \rceil = 36122$ datapoints. Subsampling was a much faster operation, but the other computationally intensive aspect of the model-explorer algorithm was computing the correlation similarity measure $\texttt{corr}\left(\ell^{(1)}, \ell^{(2)}\right)$ on the overlap of the two subsamples. Formula (1) indicates that the correlation may be computed via a double for loop, i.e. q^2 operations to compute the numerator and $\min(2kq, q^2)$ operations for the denominator. Such an algorithm is space-efficient, as it only requires tracking running totals; however, in R, this algorithm was prohibitively slow. We implemented this algorithm in C++ and linked it up with the model-explorer algorithm using RCpp. We also implemented a space-inefficient algorithm using comembership matrices, as described in [1]. In Table 1, we compare the timing of these functions on cluster labels of length 5000, and see that the C++ version is much faster.

	min	median	max
R	1.88	1.94	2.35
C++	0.03	0.04	0.04

Table 1: Comparison of timing of C++ implementation of `Corr` to R version. The correlation between random labelings of length 5000 was computed 10 times, and we report the minimum, median and maximum computing time in **seconds**. For inputs of this size, the C++ implementation is two orders of magnitude faster and is memory-efficient.

Finally, we note that our notation in this report matches the variable names in the code as well as in the lab assignment description, except that we refer to cluster labelings as `cl` for short in the code.

3 Results

The empirical distribution of correlation score from the output of `parallel.R` (as described in Section 2) is depicted in Figure 1 on the following page. The minimum correlation observed is just below 0.4 and the maximum is above 0.99. Overall, the empirical distributions appear to be stochastically decreasing for $k \geq 3$, meaning more and more mass is moved to the left end of the plot. This trend does not hold for $k = 2$, however. The histogram for $k = 2$ is roughly bimodal, with one mode centered well above 0.9 and the other around 0.7. Moreover, the modes have roughly equal mass. This suggests that there are two rough neighborhoods for the cluster labeling, and half of the time, the two runs of k-means converge to the same neighborhood and half of the time they converge to opposite neighborhoods.

The empirical cdf for $k = 3$ dominates every other curve for almost every value of the correlation. The difference is so striking, especially in the plot of the histograms, that we need not worry about the randomness in measuring the similarity distribution for $N = 100$ pairs of subsamples. Any summary of these empirical distributions, such as mean or median or maximum mode, would thus lead to the same conclusion that $k = 3$ is the most stable in the sense of [1]. Hence we would choose $k = 3$ according to the procedure outlined in [1]. We complicate this process and consider what sort of conclusions it could yield in our discussion in Section 4.

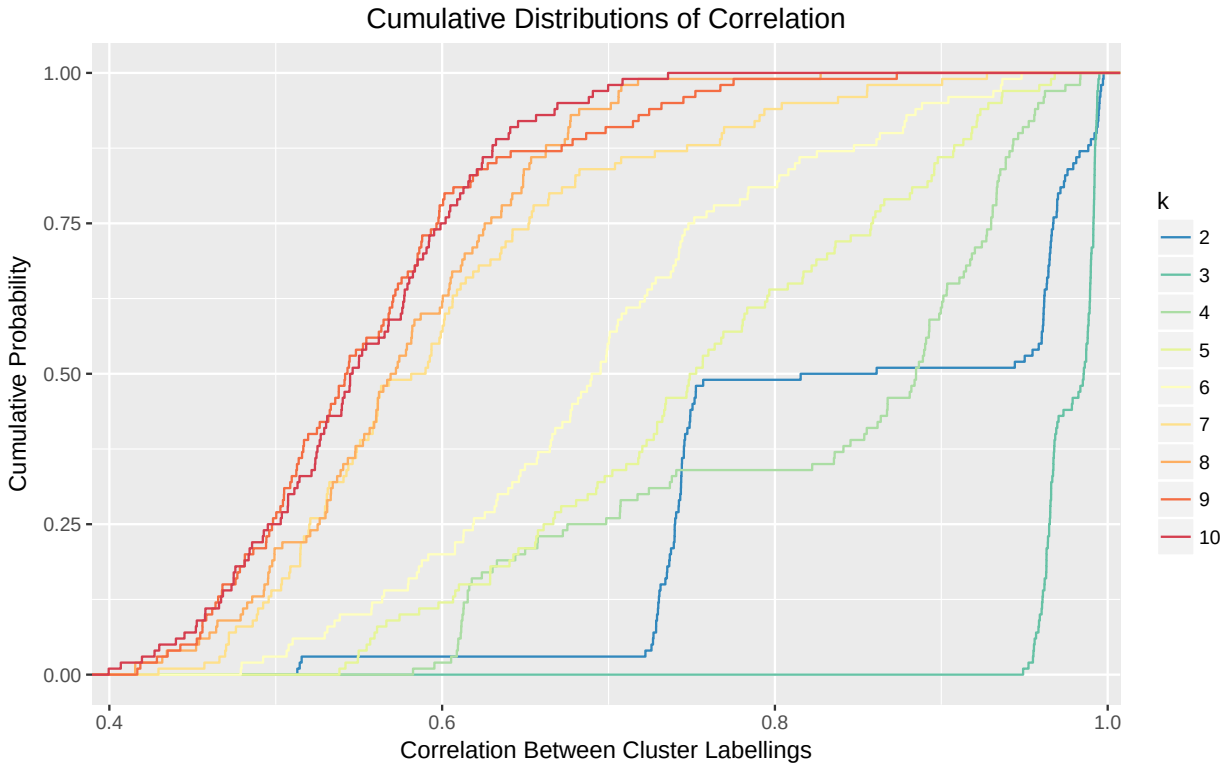
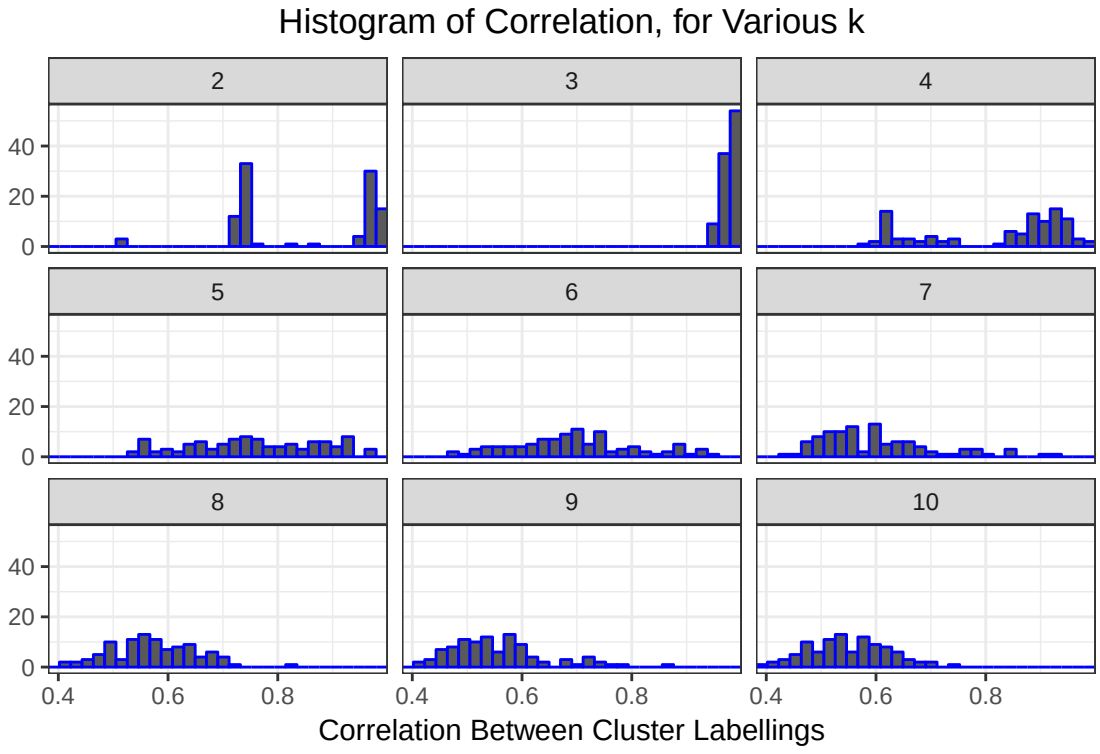


Figure 1: Above: overlay of the empirical cumulative distributions of the correlation similarity measure for increasing values of k on the `ling` binary response dataset; below: histogram of correlation. Using $k = 3$ clusters gives the most concentrated distribution with high similarity.



4 Discussion

In Figure 2 below, we plot the survey respondents colored by cluster assignment using $k = 3$ means:

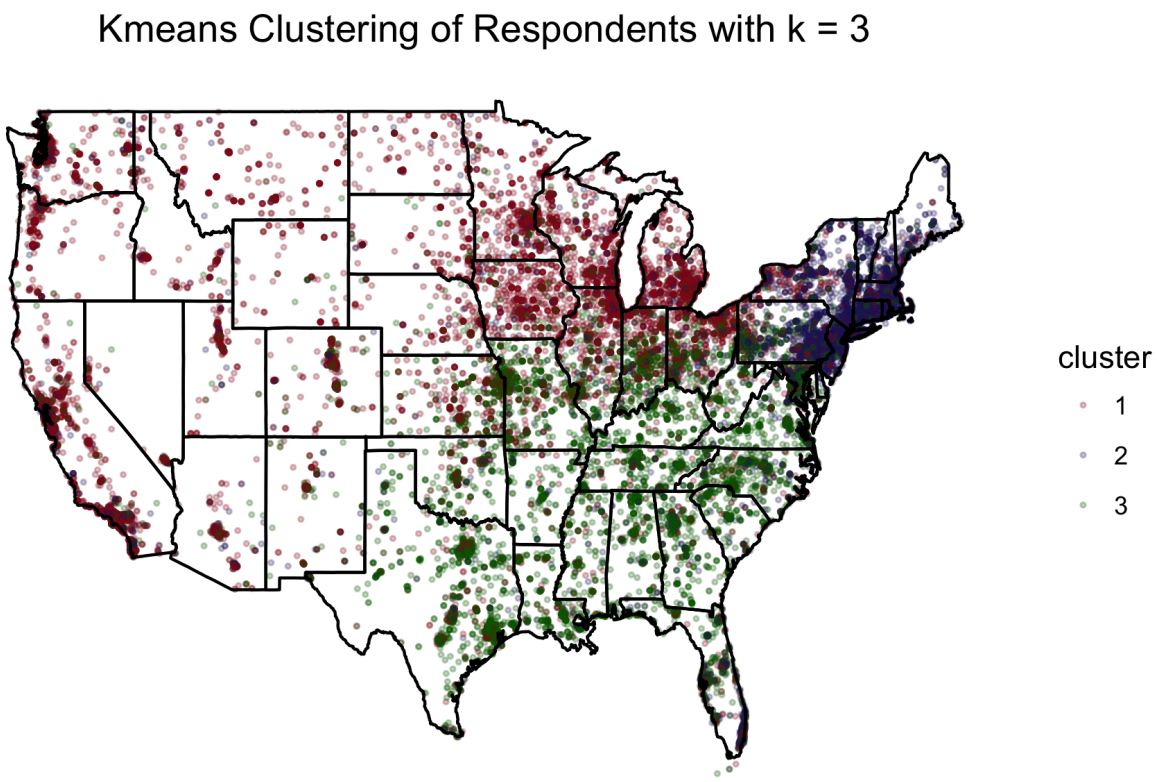


Figure 2: Coloring respondents by cluster using $k = 3$ means. Recall that k-means does not have knowledge of the lat-long information. In this section we scrutinize the claim of whether our method for the selection of k is a reliable source of insight into "linguistic differences whose distribution is determined primarily by geography." [5]

The results of Figure 1 offer a compelling story about how to choose k using the data we were provided: the result is nontrivial (in that it did not compel us to choose the absolute smallest number of clusters possible) but the trend is clear (in that the empirical distributions are stochastically decreasing for $k \geq 3$). Even better still, Ben-Hur et al. essentially conjecture that similarities will only concentrate close to 1 when there is real structure captured by a partition into k groups. We could, at this point, announce that our measure of stability (correlation on pairs of subsamples) was all that we care about, that $k = 3$ empirically minimizes this quantity, and so we will wipe our hands with the whole endeavor. Problem solved.

Not so fast! We are attempting to make a claim about the number of clusters or groupings in over $40,000$ Americans' responses to 68 linguistic survey questions; actually, we would like to infer something about meaningful linguistic groupings of **all** Americans. This is the whole point of dialectometry, not measuring the stability of a particular algorithm we used for exploratory analysis. Figure 2 *does* suggest that the clustering did something interesting on a macro-level. There are several reasons why we might want to distrust the results of [1]; in particular, why there could be many more than $k = 3$ distinct groups of respondents in the linguistic survey data. We discuss them in order of increasing severity.

The first issue is a small point, that we technically are not interested in choosing k to maximize the stability as measured in [1]. In particular, taking $k = 1$ or $k = n$ would yield a perfectly stable cluster every time, assuming our similarity metric is invariant to label switching (which they are and should be). That may seem pedantic: it may seem like we wouldn't want to do clustering if either of these were even plausible. But since n is so large, taking $k = n - 1000$ would yield almost the exact same conclusion about stability even without subsampling. [1] addresses this issue by taking k to be the largest number of clusters after which the clustering becomes unstable. This becomes a problem in our case, however, as $k = 2$ itself is unstable. One might dismiss these comments by saying k very small or very large are edge cases not meant to be interpreted in the same respect as the other k we would find more meaningful, but it points at the difficulty of actually selecting $k \in \{1, \ldots, n\}$. We emphasize that all of these are possible in the "true underlying k" sense. It is possible that there are over 40000 distinct linguistic groups in America, or that there are no meaningful groupings to be had. Our data and Figure 2 do seem to stand as strong evidence to the contrary, however.

For the second point we consider a different example. Consider data coming from a Gaussian mixture with $k = 4$ components $\{\mu_k\}_{k=1}^4$ all with isotropic covariances $\sigma^2 I$ with σ^2 known. Imagine that $\|\mu_1 - \mu_2\|_2$ and $\|\mu_3 - \mu_4\|_2$ are small relative to $\frac{\sigma}{\sqrt{n}}$ (where n is the number of observations) but the pairs are well separated from each other. This setting is a mix of high and low signal-to-noise ratio, but in particular telling μ_1 and μ_2 apart (or telling μ_3 and μ_4 apart) in this setting would be very difficult, and our process for selecting k would most definitely find $k = 2$ rather than $k = 4$. We could play the same trick with many more pairs (or triples, or quadriples, etc.) of means and find that we in trying to distinguish between, say, $k = 3$ and $k = 1000$, we are very firm in our finding that $k = 3$ even though we know (having generated the data) that in reality $k = 1000$.

Perhaps our best hope in light of this second point is to collect lots of data and pray that we have escaped that regime (which depends implicitly on the dimension so is only harshened by the fact that we have 468 columns in the `ling` binary response matrix). But this hopelessness betrays all of our analysis of the `ling` dataset in Lab 2. Therein, we saw some outstanding differences in the Miami dialect with the rest of the Southern states, but we also saw many questions on which they disagreed with the Northeast. In Southeastern Pennsylvania, we saw how stubbornly Philadelphians held onto calling sandwiches "hoagies" despite every major city around them disagreeing, and a host of questions on which the Philadelphia area was in far stronger agreement than with the rest of the Northeast. We saw how scattered the Western half of the coastal US looked with so little data in comparison, and how states like Wyoming and Idaho had a surprising amount of variation even with relatively few respondents.

All told, there is too much richness in this data, too many subsets of questions which did seem to matter strongly to certain areas, too much local variation to wipe it all away and declare $k = 3$. Using the method of Ben-Hur et al. [1], we should take comfort knowing that our macro-level breakdown of respondents into 3 groups is a fairly stable result, such that we might even use it in a subsequent hierarchical analysis. But it is a step too far to suggest that this data only has $k = 3$ clusters, let alone that Americans can be grouped into one of 3 dialects.

5 References

[1] Asa Ben-Hur, André Elisseeff, and Isabelle Guyon (2001). A stability based method for discovering structure in clustered data. *Pacific symposium on biocomputing*, volume 7.

[2] Brian Kulis and Michael I. Jordan (2012). Revisiting k-means: new algorithms via Bayesian nonparametrics. *Proceedings of the 29th international conference on machine learning.*

[3] Jeffrey W. Miller and Matthew T. Harrison (2017). Mixture models with a prior on the number of components. *Journal of the American Statistical Association.*

[4] Rob Tibshirani, Guenther Walther, and Trevor Hastie (2001). Estimating the number of clusters in a dataset via the gap statistic. *Journal of the Royal Statistical Society.*

[5] John Nerbonne and William Kretzschmar (2003). Introducing computational techniques in dialectometry. *Computers and the Humanities.*