Supplementary Material: CUR matrix decompositions for improved data analysis

Michael W. Mahoney   Petros Drineas

Illustration of our main algorithm

In Figure 0, we provide an illustration of the action of AlgorithmCUR, our main low-rank CUR matrix decomposition algorithm.

Additional data applications

To understand better our CUR matrix decomposition as an exploratory data analysis tool, we analyzed social science data: voting data from the "Second Rehnquist" U.S. Supreme Court [7] and U.S. House of Representatives roll call data for the 107^{th} Congress [5, 6]. These are two datasets to which SVD-based analysis has been applied. The Supreme Court data consists of a 468×9 matrix with entries in $\{-1, +1\}$ encoding the decisions from about 70% of the cases decided by the 9 members of the U.S. Supreme Court from 1994 to 2003 [7]. The House data consists of a $\{-1, 0, +1\}$ encoding of the "yea," not voting, and "nay," voting records on each of the $m = 990$ roll call votes for each of the $n = 444$ representatives (including midterm replacements).

When voting data from the Supreme Court [7]—arguably a data set of such a size and about which too much field specific knowledge is available for random sampling to be appropriate—is plotted on the top two singular vectors, one clearly observes the characteristic "horseshoe" pattern of data that has local ordering information [2]. (See Figures 1(a-c).) If we choose $k = 2$, which captures 79.0% of the variance in the data, then the justice with the highest leverage score is Stevens, followed by Thomas and then Scalia, and the pair of judges which captures the most variance in the data are Rehnquist and Ginsburg, who together capture 65.9% of the data variance. If $k = 1$, which has been interpreted as ordering the justices along a partisan axis and which captures 57.1% of the variance in the data, then the justices with the highest leverage scores are Kennedy and then O'Connor.

In addition, we can pick 3 actual justices (columns of A, in this case corresponding to, *e.g.*, Scalia, Kennedy, and Ginsburg) and then 3 actual cases (rows of A), such that our CUR approximation to A accurately reconstructs 91.0% of the justices' opinions. We do so by first computing the low-rank approximation $A' = CUR$ (or $A' = A_k$ by truncating the SVD if we are interested in eigenjustices and eigencases) and then rounding each element of A' to the nearest element in $\{-1, +1\}$. For comparison, by keeping just the top 2 eigenjustices and eigencases, 92.8% of the entries of the Supreme Court data matrix are accurately predicted [7].

Congressional roll call data are known to be much more homogenous [5, 6]. (See Figures 1(d) and 1(e).) For example, if $k = 2$, which captures 75% of the data variance, then no representative exhibits a particularly high leverage, *e.g.*, none has a leverage score greater than $2k/n$. Indeed, if $k = 2$, the highest leverage score for any representative is only $1.38k/n$. (A value of greater than $2k/n$ is of interest since it has been suggested as a rule of thumb to identify outliers [3].) In this case, uniform random sampling does nearly as well (both in theory and practice) as nonuniform sampling based on the leverage scores, and selecting in a greedy manner representatives with the highest leverage scores does quite poorly. For example, the projection of the members of the U.S. House of Representatives onto the span of two representatives chosen by randomly sampling according to the "leverage score" probabilities is very similar to the plot obtained for the projection onto the "best pair" of representatives (DeLauro (D-CT) and Crenshaw(R-FL)). It is also very similar to the plot obtained for *any* two representatives chosen from as the best of 10 trials, with each trial performed by sampling uniformly at random. Applying the k-means clustering algorithm on the two-dimensional data of this panel, we easily separate the Democrats from the Republicans, with only the four labeled misclassifications. For comparison, k-means on the full dataset misclassifies two or three representatives.

On the other hand, many other data sets exhibit leverage scores that are extremely nonuniform (from the point of view of classical regression diagnostics). Two were discussed in the main text. In addition, consider the full Enron email data. (See Figure 2.) The data matrix is constructed from the PRIVATE collection [1] using log-entropy term weighing and consists of $m = 65,033$ messages and a total of $n = 92,133$ terms. In this case, if, *e.g.*, $k = 12$, then $6,029$ terms (out of $92,133$) have a leverage score greater than $2k/n$, 981 terms have a leverage score greater than $20k/n$, and 29 terms have a leverage score greater than $200k/n$! Consequently, uniform sampling does very poorly, when compared with nonuniform random sampling, but greedily keeping the highest leverage terms—the greatest outliers—does remarkably well at capturing the variance in the data. For example, in Figure 2(b), the rank parameter is fixed, and shown is the best of 10 randomized trials. Results shown are for setting $k = 12$, which captures only 13.3% of the variance of the data; similar results are obtained for other values of k, but note that, e.g., $k = 60$ and $k = 120$ still capture only 22.1% and 27.8%, respectively, of the data variance. Interestingly, greedily keeping the terms with the highest leverage scores also does well in applications of more immediate interest to the data analyst. For example, terms with leverage scores greater than, *e.g.*, ca. $10k/n$ are less correlated with the main axis of variance of data and thus tend to be much more interesting and discriminative if one is interested in tasks such as the detection of time evolving, novel, or outlier topics. (See Figure 2(b) for details.) We saw something similar in the main text.

Finally, see Figures 3, 4, and 5 for additional figures illustrating the application of ALGORITHMCUR to other TechTC datasets that cluster well in the low-dimensional space.

References

[1] M.W. Berry and M. Browne. Email surveillance using non-negative matrix factorization. *Computational and Mathematical Organization Theory*, 11(3):249–264, 2005.

[2] P. Diaconis, S. Goel, and S. Holmes. Horseshoes in multidimensional scaling and kernel methods. *To appear in: Annals of Applied Statistics.*

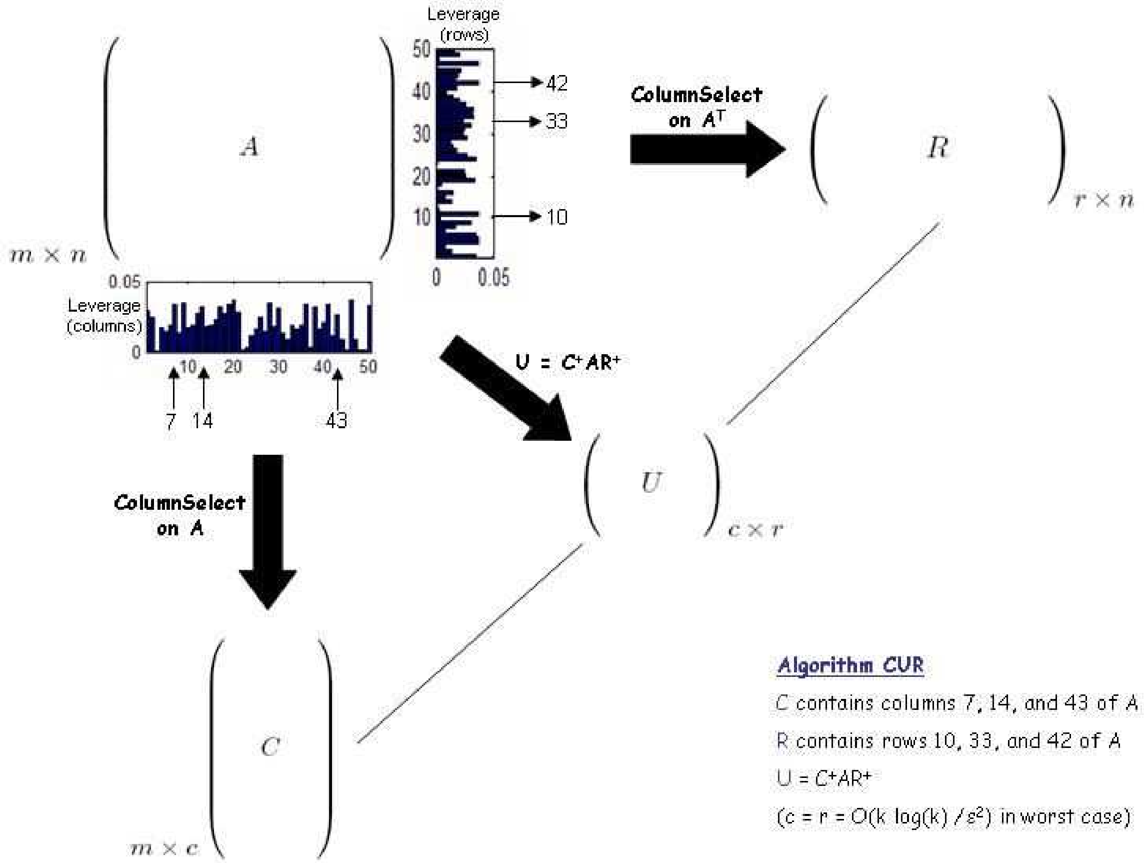
[3] D.C. Hoaglin and R.E. Welsch. The hat matrix in regression and ANOVA. *The American Statistician*, 32(1):17–22, 1978.

[4] B. Klimt and Y. Yang. The Enron corpus: A new dataset for email classification research. In *Proceedings of the 15th European Conference on Machine Learning*, pages 217–226, 2004.

[5] K.T. Poole and H. Rosenthal. Patterns of congressional voting. *American Journal of Political Science*, 35:228–278, 1991.

[6] M.A. Porter, P.J. Mucha, M.E.J. Newman, and C.M. Warmbrand. A network analysis of committees in the U.S. House of Representatives. *Proc. Natl. Acad. Sci. USA*, 102(20):7057–7062, 2005.

[7] L. Sirovich. A pattern analysis of the second Rehnquist U.S. Supreme Court. *Proc. Natl. Acad. Sci. USA*, 100(13):7432–7437, 2003.



Algorithm CUR

C contains columns 7, 14, and 43 of A

R contains rows 10, 33, and 42 of A

$U = C^+AR^+$

($c = r = O(k \log(k) / \varepsilon^2)$ in worst case)

Figure 0: AlgorithmCUR, our main low-rank CUR matrix decomposition algorithm. It takes as input an $m \times n$ real-valued matrix A, a rank parameter k, and an accuracy parameter ϵ. Its output consists of three matrices, C, U, and R such that the error $\|A - CUR\|_F$ is at most $(2+\epsilon)$ times the error of the best rank k approximation.

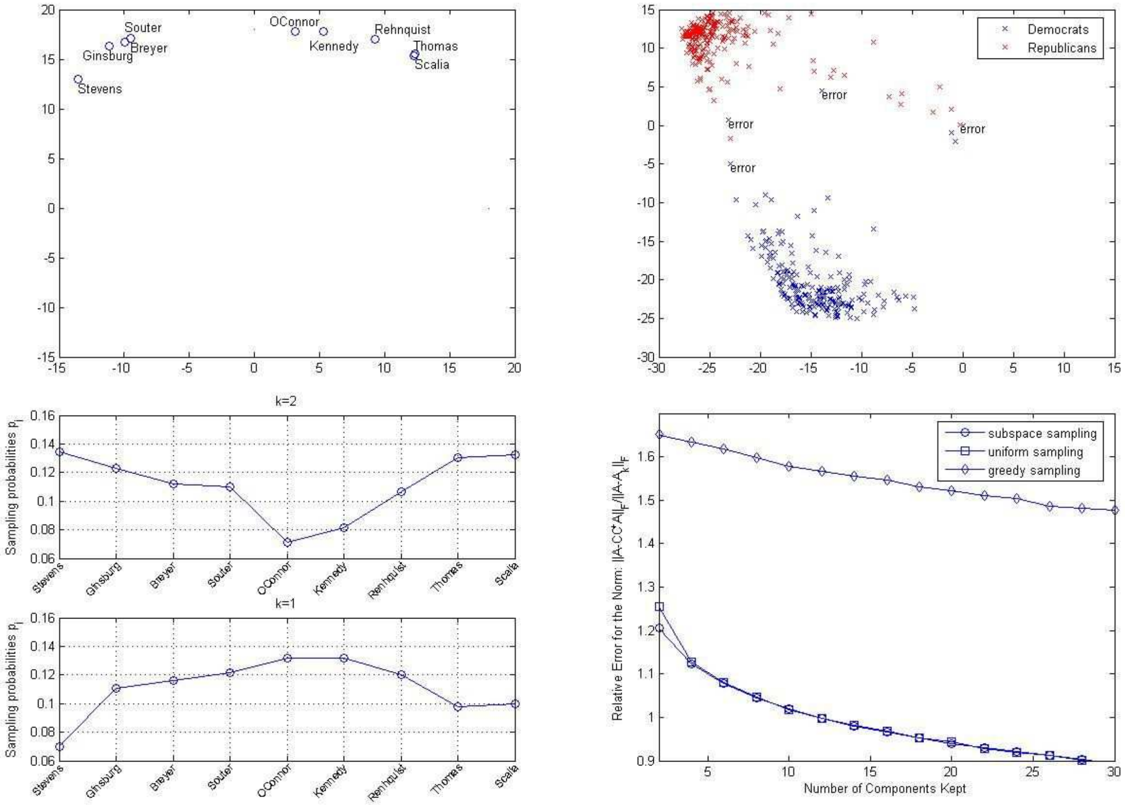


Figure 1: Application to social science data: voting data from the "Second Rehnquist" U.S. Supreme Court [7] and U.S. House of Representatives roll call data for the 107^{th} Congress [5, 6]. (A) Projection of the justices onto the space spanned by the top $k = 2$ singular justices. The x-axis corresponds to the top eigenjustice and the y-axis corresponds to the second eigenjustice. (B) and (C) Statistical leverage scores for each of the judges for the choice of $k = 2$ and $k = 1$. (D) Projection of the members of the U.S. House of Representatives onto the span of two representatives chosen by randomly samping according to the "leverage score" probabilities; shown is the best of 10 independent trials. The x, y-axes correspond to a coordinate system for the subspace spanned by the chosen two representatives. (E) Frobenius norm error for randomly sampling with the "leverage score" probabilities, compared with uniform random sampling and greedily selecting the representatives with the highest leverage scores. For random sampling, shown is the best of 10 independent trials.

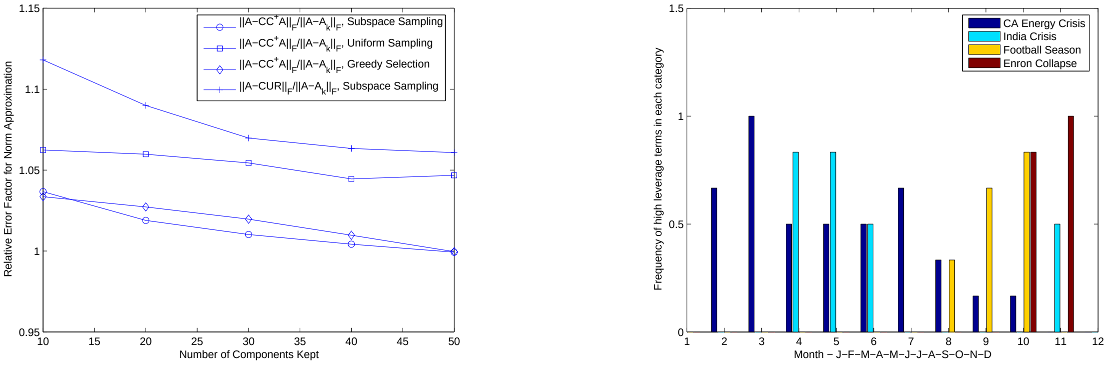


Figure 2: Application to Internet data: term-document data matrix derived from the Enron email corpus [4]. (A) Frobenius norm error for: randomly sampling columns as in AlgorithmCUR; randomly sampling columns and then rows as in AlgorithmCUR; uniform random sampling; and greedily selecting the terms with the highest leverage scores. (B) Detection of Outlier Topics: The full PRIVATE collection for 2001 is decomposed into 12 submatrices, one for each month. Then, for each month, the "leverage scores" are used to order terms from highest to lowest. Shown is a bar chart corresponding to the frequency with which terms in that category appear in the top 50 highest leverage terms. For simplicity six terms are used to define each topic: CA Energy Crisis (power, california, electricity, utilities, market, customers); India Crisis (dabhol, dpc, india(n), lender, mseb, maharashtra); Football Season (texas, fantasy, game, ut, orange, longhorns); and Enron Collapse (partnership, fastow, shares, sec, stock, investor).

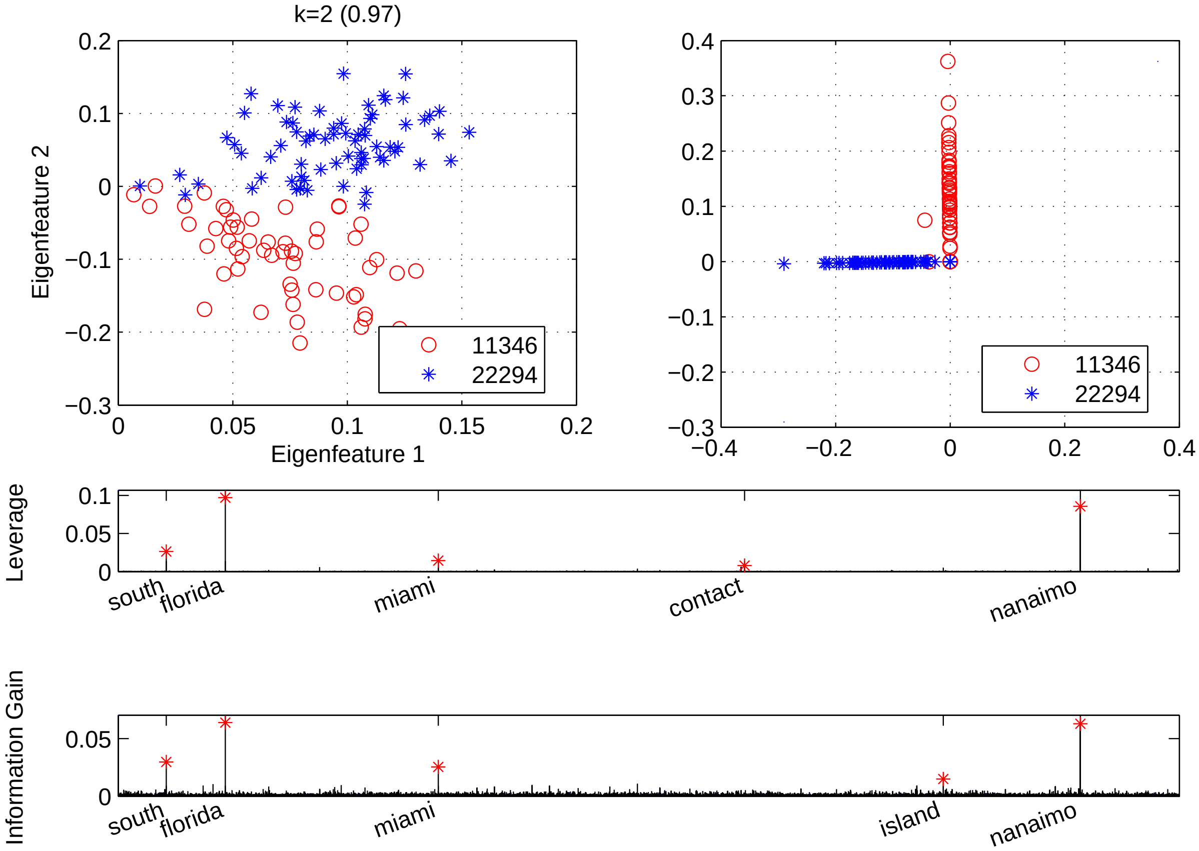


Figure 3: Application of AlgorithmCUR to a TechTC dataset. The matrix consists of 125 documents from TechTC on two topics: `US:Florida` (id:11346) and `Canada:British Columbia:Nanaimo` (id:22294).

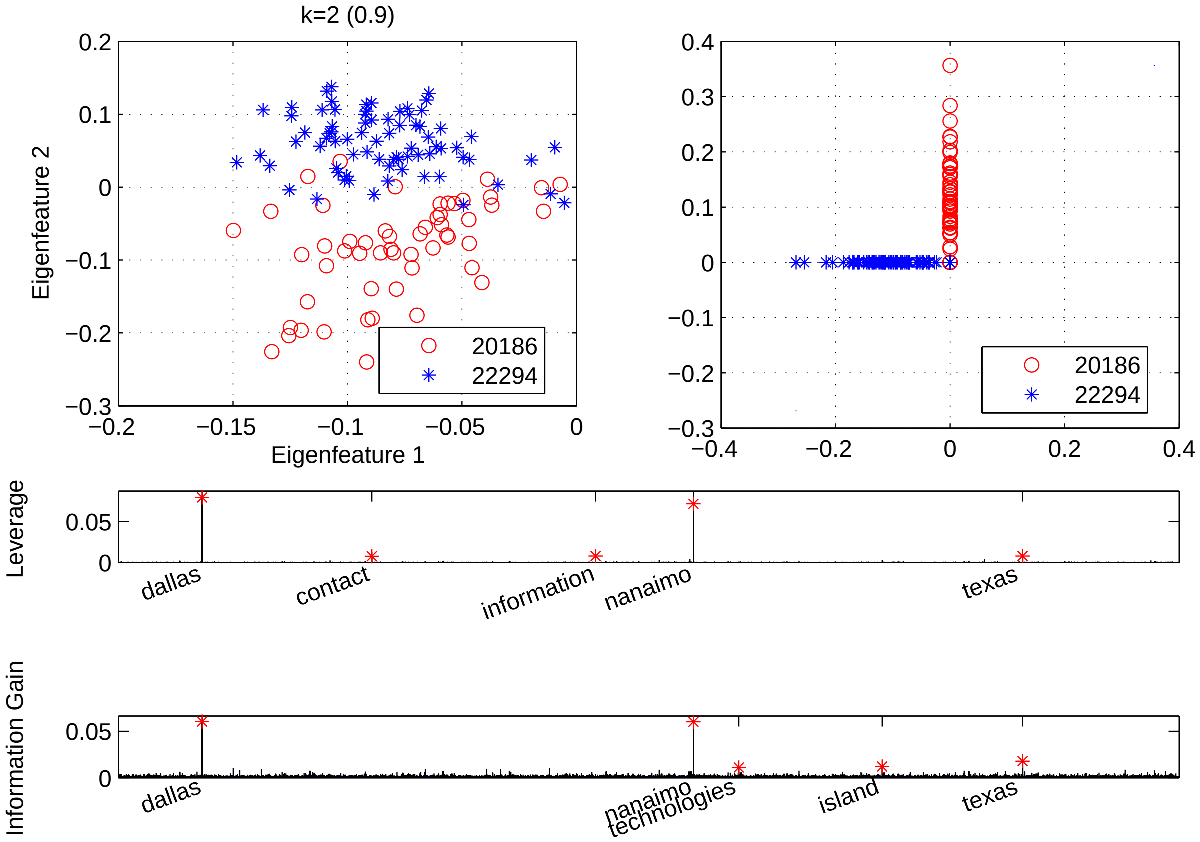


Figure 4: Application of ALGORITHMCUR to a TechTC dataset. The matrix consists of 130 documents from TechTC on two topics: `US:Texas:Dallas` (id:20186) and `Canada:British Columbia:Nanaimo` (id:22294).

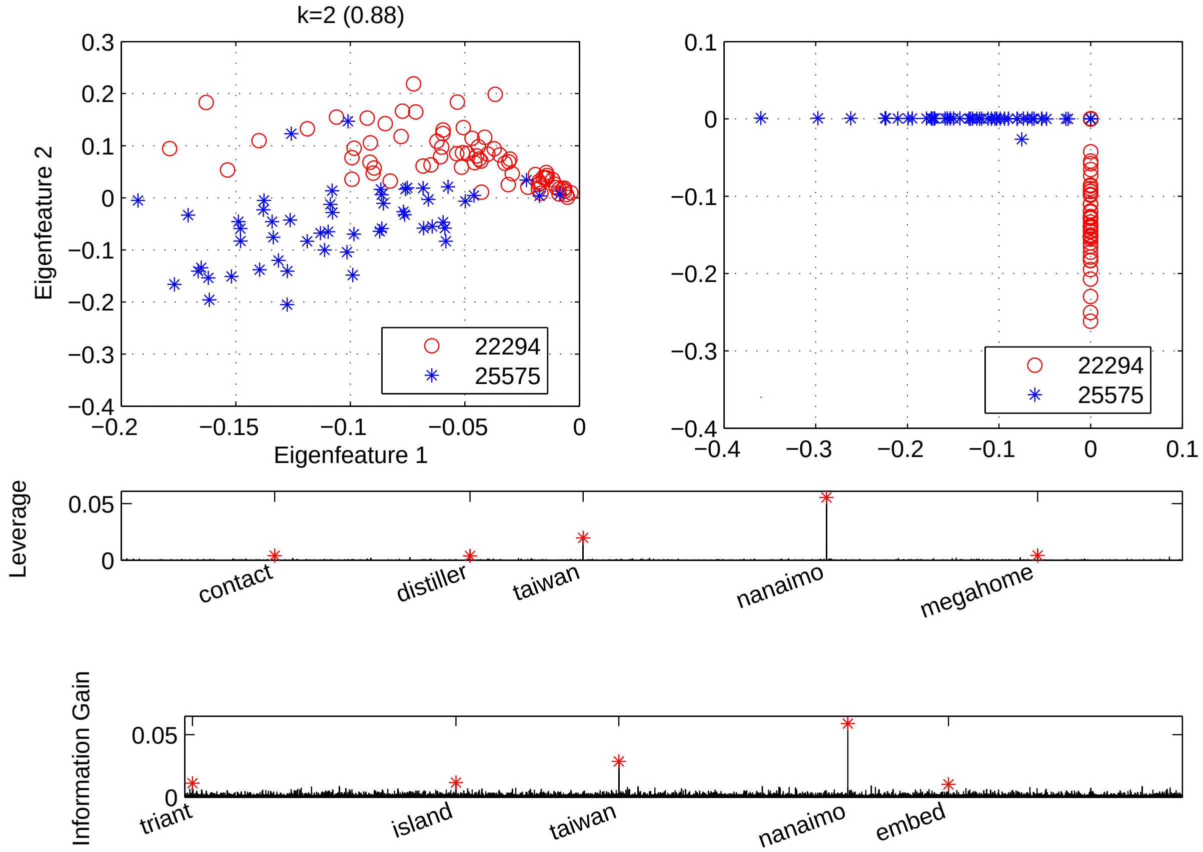


Figure 5: Application of AlgorithmCUR to a TechTC dataset. The matrix consists of 127 documents from TechTC on two topics: Canada:British Columbia:Nanaimo (id:22294) and Asia:Taiwan:Business and Economy (id:25575).