Clustering Single Cells with Noisy Observations

April 30, 2018

General notation

S is the $n \times n$ similarity matrix (symmetrized from a probability transition matrix) and is our main goal for optimization. We compute S by averaging the information of $P^{(i)}$ which are noisy similarity matrices observed from the data m different times based on either different kernels or different sets of features.

Model 1 : Multiview with one kernel

Problem set up

We divide the data set into m groups of features, and compute the similarity matrix for each data using a carefully selected kernel (we can use the general kernel formulation in SIMLR paper and choose the best one). Then we impose sparsity on the error matrix, the difference between the unobserved true similarity matrix S and the noisy matrices $P^{(i)}$.

Objective

$$\underset{S}{\operatorname{argmin}} \|S\|_* + \lambda \sum_i \|E^{(i)}\|_1$$

such that $P^{(i)} = S + E^{(i)}, S \geq 0, S1 = 1$

The Augmented Lagrangian formulation with the auxiliary variables is like below.

$$\mathcal{L}(S, Q, E^{(i)}) = \|Q\|_* + \lambda \sum_{i=1}^{m} \|E^{(i)}\|_1 + \sum_{i=1}^{m} \langle Y^{(i)}, S + E^{(i)} - P^{(i)} \rangle + \frac{\mu}{2} \|S + E^{(i)} - P^{(i)}\|_F^2 + \langle Z, S - Q \rangle + \frac{\mu}{2} \|S - Q\|_F^2$$

Model 2 : Multi-kernel with convex relaxation

Problem set up

From here on, we observe multiple similarity matrices $P^{(i)}$ by using different kernels. This assumes that the data has several nonlinear structure combined, and we aim to capture all those nonlinear patterns by using many different kernels. The actual performance was not great, and the algorithm tends to push the rank of S down to 1, but I write three variations of optimization goal for record-keeping.

Objective 1

$$\underset{S}{\operatorname{argmin}} \, \tau \|S\|_* + \sum_{i=1}^{m} \|S - P^{(i)}\|_F^2$$

s.t. $S \geq 0, S1 = 1$

$$\mathcal{L}(S) = \tau\|Q\|_* + \sum_{i=1}^{m} \|E^{(i)}\|_F^2 + \langle Y, S - Q\rangle + \frac{\mu}{2}\|S - Q\|_F^2 + \sum_i \langle W^{(i)}, S + E^{(i)} - P^{(i)}\rangle + \frac{\mu}{2}\|S + E^{(i)} - P^{(i)}\|_F^2$$

Objective 2

$$\operatorname*{argmin}_S \tau\|S\|_* + \sum_{i=1}^{m} \lambda_i \|S - P^{(i)}\|_F^2 + \sum_{i=1}^{m} \lambda_i^2$$

such that $S \geq 0, S1 = 1, \sum_i \lambda_i = 1, \lambda_i \geq 0$

$$\mathcal{L}(S) = \tau\|Q\|_* + \sum_{i=1}^{m} \lambda_i \|E^{(i)}\|_F^2 + \langle Y, S - Q\rangle + \frac{\mu}{2}\|S - Q\|_F^2 + \sum_i \langle W^{(i)}, S + E^{(i)} - P^{(i)}\rangle + \frac{\mu}{2}\|S + E^{(i)} - P^{(i)}\|_F^2 + \sum_{i=1}^{m} \lambda_i^2$$

Objective 3

$$\operatorname*{argmin}_S \|S\|_* + \sum_{i=1}^{m} \frac{S - P^{(i)}}{2n^2\sigma_i} + \frac{\sigma_i}{2}$$

where $S \geq 0, S1 = 1$, and there are no constraints on σ_i

$$\mathcal{L} = \tau\|Q\|_* + \sum_{i=1}^{m} \frac{\|E^{(i)}\|_F^2}{2n^2\sigma_i} + \frac{\sigma_i}{2} + \langle Y, S - Q\rangle + \frac{\mu}{2}\|S - Q\|_F^2 + \sum_{i=1}^{m} \langle Z^{(i)}, P^{(i)} - S - E^{(i)}\rangle + \sum_{i=1}^{m} \frac{\mu}{2}\|P^{(i)} - S - E^{(i)}\|_F^2$$

Model 3 : Multi-kernel without convex relaxation

Problem set up

Without minimizing the trace norm $\|\cdot\|_*$, we use Eckart-Young-Mirsky theorem and constrain S to have rank less than or equal to a user-defined integer r, which corresponds to the expected number of clusters. We try giving different weights to different kernels to capture non-linear structure well.

Objective 0

$$\operatorname*{argmin}_S \sum_{i=1}^{m} \|S - P^{(i)}\|_F^2$$

such that $rank(S) \leq r, S \geq 0, S1 = 1$

Objective 1

$$\operatorname*{argmin}_{S,w} \frac{1}{2}\sum_{i=1}^{m} w_i \|S - P^{(i)}\|_F^2 + \sum_{i=1}^{m} w_i^2 \tag{1}$$

s.t. $rank(S) \leq r, S \geq 0, S1 = 1, \sum_{i=1}^{m} w_i = 1$

$$\mathcal{L} = \sum_{i=1}^{m} \frac{1}{2} w_i \|Q^{(i)} - P^{(i)}\|_F^2 + \sum_{i=1}^{m} w_i^2 + \sum_{i=1}^{m} \langle Y^{(i)}, S - Q^{(i)}\rangle + \frac{\mu}{2}\|S - Q^{(i)}\|_F^2$$

- **update** Q

$$\operatorname*{argmin}_{Q^{(i)}} \mathcal{L} = \operatorname*{argmin}_{Q^{(i)}} \left\| Q^{(i)} - \frac{1}{w_i + \mu} \left(\mu S + P^{(i)} w_i + Y^{(i)} \right) \right\|_F^2$$

  such that $rank(Q^{(i)}) \leq r$. That means, we take SVD of $\frac{1}{w_i+\mu}\mu S + P^{(i)} w_i + Y^{(i)}$ and push all singular values to zero except the first r.

- **update** S

$$\operatorname*{argmin}_{S} \mathcal{L} = \operatorname*{argmin}_{S} \left\| S - \sum_{i=1}^{m} \frac{1}{m} \left(Q^{(i)} - \frac{Y^{(i)}}{\mu} \right) \right\|_F^2$$

  such that $rank(S) \leq r$. That means, we take SVD of $\frac{1}{m}\left(Q^{(i)} - \frac{Y^{(i)}}{\mu} \right)$ and truncate all singular values after the first r ones.

- **update** w

$$\operatorname*{argmin}_{w} \mathcal{L} = \operatorname*{argmin}_{w} \sum_{i=1}^{m} \frac{1}{2} w_i \| Q^{(i)} - P^{(i)} \|_F^2 + \sum_{i=1}^{m} w_i^2$$

  where $\sum_{i=1}^{m} w_i = 1$ and $w \geq 0$. We can solve this using the interior point quadratic programming method.

- **update** Y

$$Y_{new}^{(i)} = Y^{(i)} + \mu(S - Q^{(i)})$$

- **update** μ

$$\mu_{new} = \mu\rho$$

  where ρ is a user-defined multiplication factor.

Objective 1 Modified

$$\operatorname*{argmin}_{S,w} \frac{1}{2} \sum_{i=1}^{m} w_i \| S - P^{(i)} \|_F^2 + \sum_{i=1}^{m} w_i^2 \qquad (2)$$

s.t. $rank(S) \leq r, S \geq 0, S1 = 1, \sum_{i=1}^{m} w_i = 1$

$$\mathcal{L} = \sum_{i=1}^{m} \frac{1}{2} w_i \| Q^{(i)} - P^{(i)} \|_F^2 + \sum_{i=1}^{m} w_i^2 + \sum_{i=1}^{m} \langle Y^{(i)}, S - Q^{(i)} \rangle + \frac{\mu}{2} \| S - Q^{(i)} \|_F^2$$

- **update** Q

$$\operatorname*{argmin}_{Q^{(i)}} \mathcal{L} = \operatorname*{argmin}_{Q^{(i)}} \left\| Q^{(i)} - \frac{1}{w_i + \mu} \left(\mu S + P^{(i)} w_i + Y^{(i)} \right) \right\|_F^2$$

  such that $rank(Q^{(i)}) \leq r$. That means, we take SVD of $\frac{1}{w_i+\mu}\mu S + P^{(i)} w_i + Y^{(i)}$ and push all singular values to zero except the first r.

- **update** S

$$\operatorname*{argmin}_{S} \mathcal{L} = \operatorname*{argmin}_{S} \left\| S - \sum_{i=1}^{m} \frac{1}{m} \left(Q^{(i)} - \frac{Y^{(i)}}{\mu} \right) \right\|_F^2$$

  such that $rank(S) \leq r$. That means, we take SVD of $\frac{1}{m}\left(Q^{(i)} - \frac{Y^{(i)}}{\mu} \right)$ and truncate all singular values after the first r ones.

- **update** w
$$\underset{w}{\operatorname{argmin}} \mathcal{L} = \underset{w}{\operatorname{argmin}} \sum_{i=1}^{m} \frac{1}{2} w_i \|Q^{(i)} - P^{(i)}\|_F^2 + \sum_{i=1}^{m} w_i^2$$
where $\sum_{i=1}^m w_i = 1$ and $w \geq 0$. We can solve this using the interior point quadratic programming method.

- **update** Y
$$Y_{new}^{(i)} = Y^{(i)} + \mu(S - Q^{(i)})$$

- **update** μ
$$\mu_{new} = \mu\rho$$
where ρ is a user-defined multiplication factor.

Objective 2

$$\underset{S,\sigma}{\operatorname{argmin}} \sum_{i=1}^{m} \frac{\|S - P^{(i)}\|_F^2}{2n^2\sigma_i} + \frac{\sigma_i}{2} \tag{3}$$

s.t. $rank(S) \leq r, S \geq 0, S1 = 1$

$$\mathcal{L} = \sum_{i=1}^{m} \frac{\|Q^{(i)} - P^{(i)}\|_F^2}{2n^2\sigma_i} + \frac{\sigma_i}{2} + \sum_{i=1}^{m} \langle Y^{(i)}, S - Q^{(i)} \rangle + \frac{\mu}{2}\|S - Q^{(i)}\|_F^2$$

- **update** Q
$$\underset{Q^{(i)}}{\operatorname{argmin}} \mathcal{L} = \underset{Q^{(i)}}{\operatorname{argmin}} \left\| Q^{(i)} - \frac{\sigma_i n^2}{\mu\sigma_i n^2 + 1} \left(\frac{P^{(i)}}{\sigma_i n^2} + Y^{(i)} + \mu S \right) \right\|_F^2$$
We take SVD of $\frac{\sigma_i n^2}{\mu\sigma_i n^2+1} \left(\frac{P^{(i)}}{\sigma_i n^2} + Y^{(i)} + \mu S \right)$ and take the first r singular values only.

- **update** S
$$\underset{S}{\operatorname{argmin}} \mathcal{L} = \underset{S}{\operatorname{argmin}} \left\| S - \sum_{i=1}^{m} \frac{1}{m} \left(Q^{(i)} - \frac{Y^{(i)}}{\mu} \right) \right\|_F^2$$
and we take SVD of $\frac{1}{m} \left(Q^{(i)} - \frac{Y^{(i)}}{\mu} \right)$ and take the first r singular values only.

- **update** σ
$$\underset{\sigma_i}{\operatorname{argmin}} \mathcal{L} = \frac{\|S - P^{(i)}\|_F^2}{n^2}$$

Objective 3

$$\underset{S,w}{\operatorname{argmin}} \sum_{i=1}^{m} \frac{\|S - P^{(i)}\|_F^2}{\sigma_i^2} w_i + \sum_{i=1}^{m} w_i^2 \tag{4}$$

s.t. $rank(S) \leq r, S \geq 0, S1 = 1, \sum_{i=1}^m w_i = 1$

How to assign weights to different kernels

In this section we compare the objectives (1), (2), and (3).

(1) is straightforwardly assigning different weights to different kernels, where the term $\sum_{i=1}^{m} w_i^2$ makes the optimization for w smooth. The sum of weights equal to 1. This is very close to SIMLR except that they use $wlogw$ instead of w^2 to make the function smooth.

(2) is inspired by scaled lasso. Scaled lasso divides the error by $2n\sigma_i$, but here we divide by $2n^2\sigma_i$ because we are dealing with a $n \times n$ matrix instead of a $n \times 1$ vector. Dividing by $2n^2\sigma_i$ is essentially the same as assigning different weights w_i to each kernel as in (1), but the constraint is different. (1) constrains the sum of the weights to be 1. Translating this to σ would be $\sum_i \frac{1}{2n\sigma_i} = 1$. However, scaled lasso instead penalizes on $\frac{\sigma_i}{2}$. Intuitively, this is reducing the noise level for each kernel. (2) is jointly convex in terms of S and σ, and when σ is optimized through KKT condition,

$$\sigma_i^2 = \frac{\|S - P^{(i)}\|_F^2}{n^2}$$

which is element-wise average of the squared error. During our last meeting, we somehow concluded that this 'maximizes' the goal, but that was because we forgot to square the Frobenius norm. The optimization goal is indeed jointly convex, and above σ_i will 'minimize' the function.

(3) does not optimize over σ. Instead, we compute σ^2 to be the $\|S - P^{(i)}\|_F^2/n^2$ (which coincides with the solution of (2)). Then, for all i,

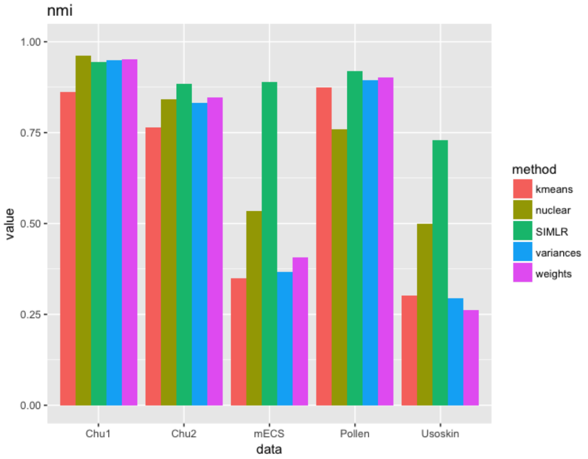
$$\frac{\|S - P^{(i)}\|_F^2}{\sigma_i^2} = \frac{\sum_{j=1}^{n}\sum_{k=1}^{n}(S_{j,k} - P_{j,k})^2}{\sum_{j=1}^{n}\sum_{k=1}^{n}(S_{j,k} - P_{j,k})^2/n^2} = n^2$$

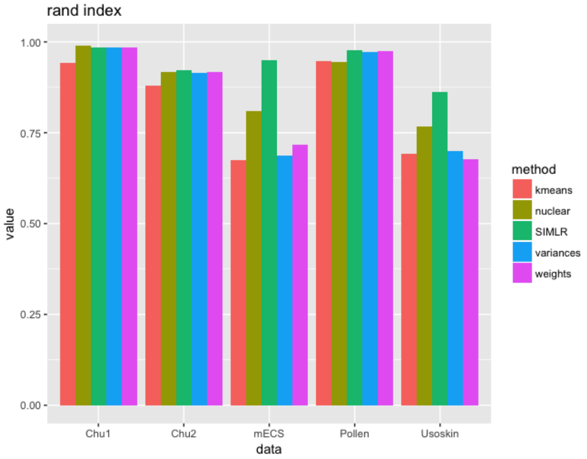
So essentially the optimization problem becomes

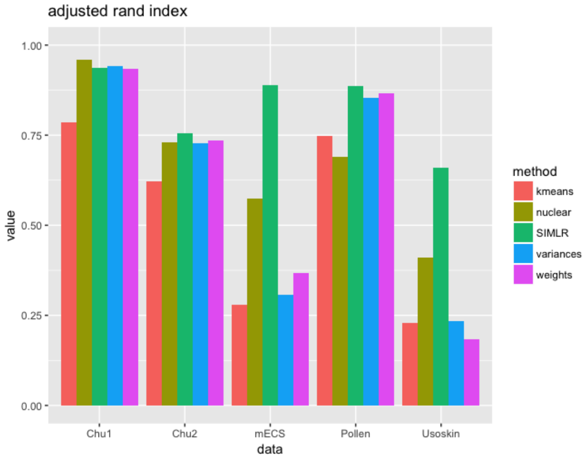
$$\operatorname*{argmin}_{S,w} \sum_{i=1}^{m} n^2 w_i + \sum_{i=1}^{m} w_i^2$$

making it not informative of S or $P^{(i)}$.

Performance comparison







Re-visit nuclear norm

$$\min_S \sum_{i=1}^m w_i^2 \|S - P^{(i)}\|_F^2 + \tau \|S\|_* \text{ such that } S1 = 1, S \geq 0, \sum_i w_i = 1, w_i \geq 0$$

$$\mathcal{L} = \tau \|Q\|_* + \sum_{i=1}^m w_i \|Q - P^{(i)}\|_F^2 + \langle Y, S - Q \rangle + \frac{\mu}{2} \|S - Q\|_F^2 + \sum_i w_i^2$$

- **update Q**

$$\operatorname*{argmin}_Q \tau \|Q\|_* + \sum_{i=1}^m \frac{w_i}{2} \|Q - P^{(i)}\|_F^2 + \langle Y, S - Q \rangle + \frac{\mu}{2} \|S - Q\|_F^2$$

$$= \operatorname*{argmin}_Q \tau \|Q\|_* + \frac{\mu + 2}{2} \left\| Q - \frac{1}{(\mu + 2)} \left(\sum_i w_i P_i + \mu S + Y \right) \right\|_F^2$$

So we take SVD of $\frac{1}{\mu+2} \left(\sum_i w_i P_i + \mu S + Y \right)$ and take the soft threshold on the singular values using $\frac{\tau}{\mu+2}$

- **update S**

$$\operatorname*{argmin}_S \left\| S - \left(Q - \frac{Y}{\mu} \right) \right\|_F^2$$

Therefore project onto the domain $S \geq 0$, $S1 = 1$ using $Q - \frac{Y}{\mu}$.

Include rank information through Laplacian matrix

$$\min_{S,J,w} \tau \|S\|_* + \gamma tr(J^T D^{-1/2}(I - S)D^{-1/2}J) + \frac{1}{2}\sum_i w_i \|S - P_i\|_F^2 + \frac{1}{2}\sum_i w_i^2$$

$$\mathcal{L} = \tau \|Q\|_* + \gamma tr(J^T D^{-1/2}(I - Q)D^{-1/2}J) + \frac{1}{2}\sum_i w_i \|Q - P_i\|_F^2 + \frac{1}{2}\sum_i w_i^2 + \langle Y, S - Q \rangle + \frac{\mu}{2}\|S - Q\|_F^2$$

- **update Q**

$$\operatorname*{argmin}_Q \tau \|Q\|_* + \frac{\mu+1}{2}\|Q - \frac{1}{\mu+1}(\gamma \tilde{J}\tilde{J}^T + \sum_i w_i P_i + Y + \mu S)\|_F^2$$

Therefore, do the SVD of $\gamma \tilde{J}\tilde{J}^T + \sum_i w_i P_i + Y + \mu S$ and do the soft thresholding with $\frac{\tau}{\mu+1}$

- **update S**

$$\operatorname*{argmin}_S \frac{\mu}{2}\|S - (Q - \frac{Y}{\mu})\|_F^2$$

- **update J**

$$\operatorname*{argmax}_J tr(J^T D^{-1/2}(S - I)D^{-1/2}J)$$

So J is the first r eigenvalues of $D^{-1/2}(S - I)D^{-1/2}$

Record of Optimization Goals

1. objective_1_c

$$\min_{S,w} \frac{1}{2} \sum_{i=1}^{m} w_i \|S - P_i\|_F^2 + \sum_{i=1}^{m} w_i^2$$

$S1 = 1, S \geq 0, rank(S) \leq r, w \geq 0, \sum_{i=1}^{m} w_i = 1$

2. objective_2_c

$$\min_{S,\sigma} \sum_{i=1}^{m} \frac{\|S - P_i\|_F^2}{2n^2\sigma_i} + \frac{\sigma_i}{2}$$

$S1 = 1, S \geq 0, rank(S) \leq r$

3. objective_3

$$\min_{S,w} \sum_{i=1}^{m} w_i \|S - P_i\|_F^2 + \tau \|S\|_* + \frac{1}{2} \sum_{i} w_i^2$$

$S1 = 1, S \geq 0, \sum w_i = 1, w \geq 0$

4. objective_4

$$\min_{S,J,w} \sum_{i=1}^{m} w_i \|S - P_i\|_F^2 + \tau \|S\|_* + \gamma tr(J^T \tilde{L} J) + \frac{1}{2} \sum_{i} w_i^2$$

$S1 = 1, S \geq 0, \sum w_i = 1, w \geq 0$

5. objective_5

$$\min_{S,J,w} \sum_{i=1}^{m} w_i \|S - P_i\|_F^2 + \tau \|S\|_* + \gamma tr(J^T \tilde{L} J) + \gamma \|S\|_F^2 + \frac{1}{2} \sum_{i} w_i^2$$

$S1 = 1, S \geq 0, \sum w_i = 1, w \geq 0, J^T J = I_r$ Note that it's equivalent to

$$\min_{S,J,w} \sum_{i=1}^{m} w_i \|S - P_i\|_F^2 + \tau \|S\|_* + \gamma \|S - \frac{1}{2} J J^T\|_F^2 + \frac{1}{2} \sum_{i} w_i^2$$

with the same constraints. An extra penalty term is pushing S towards rank r matrix JJ^T.

6. obectjve_6

$$\min_{S} \sum_{i} w_i \|S - P_i\|_F^2 + \sum_{i} w_i^2$$

$S1 = 1, S \geq 0, \sum w_i = 1, w \geq 0, rank(L_S) = n - k$

7. nuclear_scaledlasso

$$\min_{S,w} \sum_{i=1}^{m} \frac{\|S - P_i\|_F^2}{2n^2\sigma_i} + \tau \|S\|_* + \frac{1}{2} \sum_{i} \frac{\sigma_i}{2}$$

$S1 = 1, S \geq 0$

8. Laplacian1

$$\tau\|I - L\|_* + \sum_i \frac{\|L - M_i\|_F^2}{2n^2\sigma_i} + \sum_i \frac{\sigma_i}{2}$$

L is the true Laplacian matrix, M_i is the i'th observed Laplacian matrix. We use **unnormalized** similarity matrix to construct Laplacian so that the use of row sums when constructing the Laplacian accounts for the local structure.

9. Laplacian2

$$\tau\|I - L\|_* + \sum_i w_i\|L - M_i\|_F^2 + \sum_i w_i^2$$

$w_i \geq 0$, $\sum_i w_i = 0$ Same as above except that weight is used instead of scaled lasso.

- **nucs**

$$\min_{S,w}\sum_{i=1}^{m}\frac{\|S-P_i\|_F^2}{2n^2\sigma_i}+\tau\|S\|_*+\frac{1}{2}\sum_i\frac{\sigma_i}{2}$$

  $S1=1, S\geq 0$

- **rank**

$$\min_{S,J,w}\sum_{i=1}^{m}w_i\|S-P_i\|_F^2+\tau\|S\|_*+\gamma tr(J^T\tilde{L}J)+\frac{1}{2}\sum_i w_i^2$$

  $S1=1, S\geq 0, \sum w_i=1, w\geq 0$

- **kmeans** Kmeans on the entire feature set
- **zsimlr** Software SIMLR with its default initialization

Initialization: how to build P

- **Build similarity matrices K_ℓ with different kernel parameters**
  Start with data matrix $X \in \mathbb{R}^{n \times p}$ where n is the number of cells and p is the number of features. First, build Euclidean distance matrix $D \in \mathbb{R}^{n \times n}$ where

  $$D_{ij} = \|x_i - x_j\|_2^2$$

  x_i and x_j are cells i and j taken from the i'th and j'th rows of X. Since we are using multiple kernels with different sets of parameters, use index $\ell \in \{1, 2, \cdots, L\}$ to denote that the kernel is built with parametersk_ℓ and σ_ℓ. The default initialization is $k_\ell \in \{10, 15, 20, 25, 30\}$ and $\sigma_\ell \in \{2, 1.9, 1.8, 1.7, 1.6, 1.5, 1.4, 1.3, 1.2, 1.1, 1\}$. Now, for each ℓ, build $\mu_\ell \in \mathbb{R}^{n \times 1}$ vector that averages the distance to k nearest neighbors.

  $$\mu_{i\ell} = \frac{\sum_{j \in KNN_\ell(x_i)} \|x_i - x_j\|_2^2}{k_\ell}$$

  Now we can build the variance matrix ϵ_ℓ:

  $$\epsilon_{i,j,\ell} = \frac{(\mu_i + \mu_j)\sigma_\ell}{2}$$

  μ and ϵ account for the local structure. If two nodes are relatively isolated from other nodes, the distance between them is adjusted to be smaller. Finally, we use ϵ_ℓ to build the similarity matrix $K_\ell \in \mathbb{R}^{n \times n}$:

  $$\begin{aligned} K_{i,j,\ell} &= N(D_{ij}; 0, \epsilon_{i,j,\ell}) \\ &= \frac{1}{\sqrt{2\pi}\,\epsilon_{i,j,\ell}} exp\left(-\frac{D_{ij}}{2\epsilon_{i,j,\ell}^2}\right) \end{aligned}$$

- **Build normalized distance matrix C**
  Build distance matrix C that again adjusts for the local structure:

  $$C_{i,j,\ell} = K_{i,i,\ell} + K_{j,j,\ell} - 2K_{i,j,\ell}$$

- **Using knn, build sparse similarity matrices P**
  For each ℓ (dropping the subscript ℓ in this section), only take the closest $k+1$ neighbors' distances and indices using C and build two matrices

  $$M, I \in \mathbb{R}^{n \times (k+1)}$$

  The i'th row of matrix M has an ordered distance for point i's $k+1$ nearest neighbors, and I has the corresponding indices of those neighbors. Then, convert this back to a similarity matrix by subtracting each column from the $(k+1)$'th column of M so that the $(k+1)$'th column becomes a vector of 0's. Removing this last column, now we have $\tilde{M} \in \mathbb{R}^{n \times k}$ that has the largest similarity in the first column and the smallest similarity in the k'th column. Then normalize $\tilde{M}$ so that each row sums to 1. The similarity measures are all 0's except these k nearest neighbors. Now create similarity matrix A by assigning these nonnegative similarity measures to its correct indices of the neighbors. Now, $A \in \mathbb{R}^{n \times n}$ is a transition probability matrix that is element-wise non-negative and has row sums of 1. Then finally create symmetric similarity matrix P from A.

  $$P = (A + A^T)/2$$

Optimization algorithm

We solve the following to get the underlying true similarity matrix S.

$$\min_{S,\sigma} \sum_{i=1}^{m} \frac{\|S - P_i\|_F^2}{2n^2\sigma_i} + \tau\|S\|_1 + \sum_i \frac{\sigma_i}{2}$$

$S1 = 1, S \geq 0.$
We employ the scaled lasso to give different weights to each observed similarity P_i and to control the noise level simultaneously.

$$\mathcal{L} = \sum_{i=1}^{m} \frac{1}{2n^2\sigma_i}\|Q - P_i\|_F^2 + \tau\|Q\|_1 + \sum_i \frac{\sigma_i}{2} + \langle Y, S - Q\rangle + \frac{\mu}{2}\|S - Q\|_F^2$$

- **update** Q
  Defining $\phi = \mu + \sum_i \frac{1}{n^2\sigma_i}$,

  $$\mathcal{L}(Q) = \frac{\phi}{2}\left\| Q - \frac{1}{\phi}\left(\sum_i \frac{P_i}{n^2\sigma_i} + Y + \mu S\right)\right\|_F^2$$

  Then soft-threshold the matrix element-wise.

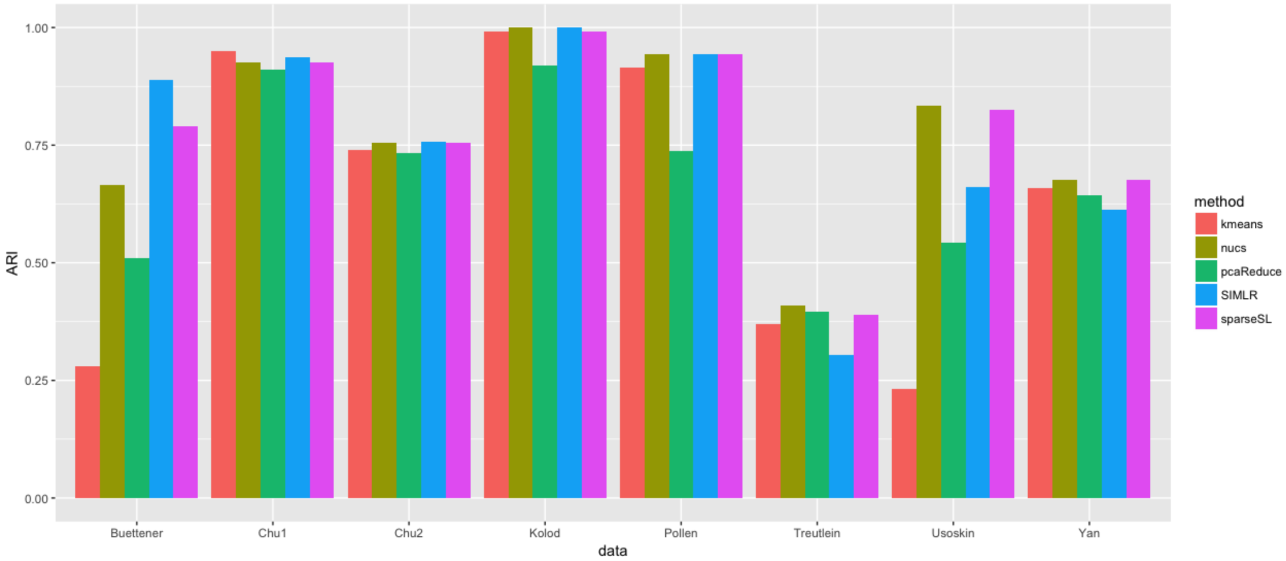
- **update** S

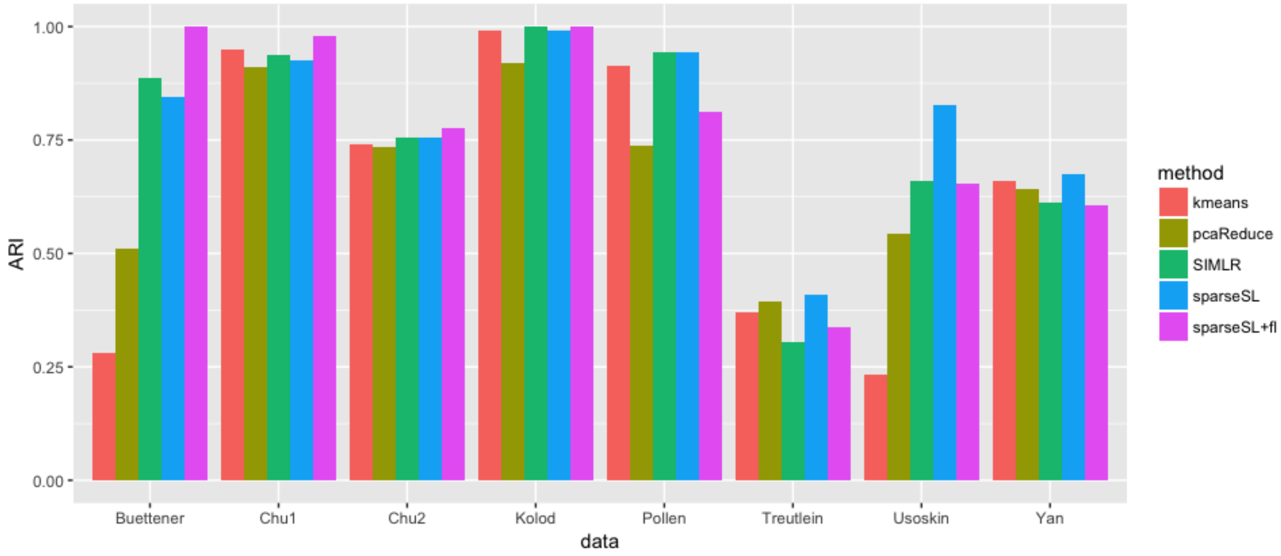
  $$\mathcal{L}(S) = \left\| S - \left(Q - \frac{Y}{\mu}\right)\right\|_F^2$$

  where $S \geq 0$ and $S1 = 1$

- **update** σ

  $$\sigma_i = \frac{\|P_i - Q\|_F^2}{n^2}$$





Tuning parameter τ