Unsupervised dimensionality reduction

Juha Mehtonen, MSc, PhD student

Institute of Biomedicine, School of Medicine, University of Eastern Finland

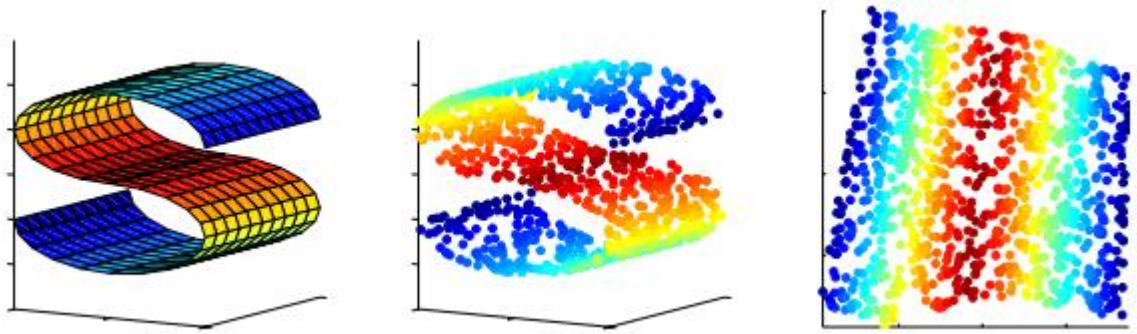
Presentation overview

- Dimensionality reduction
- Dimensionality reduction evaluation
- Hemap

Dimensionality reduction

Dimensionality reduction

- The process of reducing the number of random variables (i.e. features) in a data set.
- Retain as much of the information as possible.
- Feature extraction vs. feature selection.

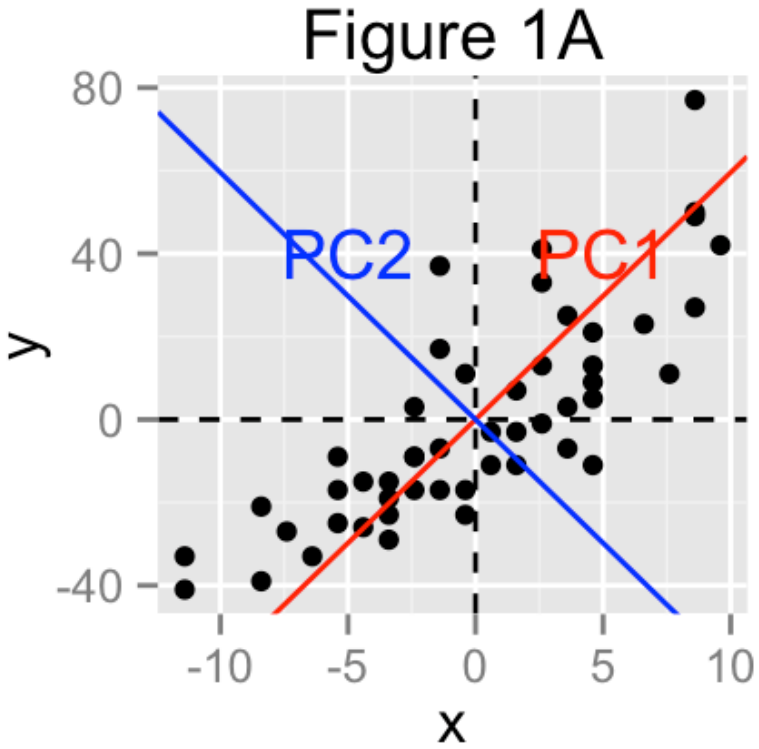


PCA

Principal Component Analysis

PCA – Principal Component Analysis

- Orthogonal projection of data onto lower-dimensional subspace
  - Projection dimensionality < Original data dimensionality
- Maximizes the variance of the projected data.
- Minimizes the mean squared distance between data points and projections.
- In other words: **Project the data to a lower-dimension while preserving as much of the information as possible.**



PCA – Principal Component Analysis

- We want to project $N*D$ matrix X into a $N*m$ matrix Y, where $m < D$.

1. Centralize the data (Subtract the feature means).
2. Calculate covariance matrix C = 1 / (N-1) * X * X'.
   - Diagonals in C are the variance of each feature.
   - Off-diagonals are the covariances of two features.
3. Calculate the eigenvectors of C.
4. Select m largest eigenvectors that correspond to largest m eigenvalues.

PCA – Principal Component Analysis

- Assumptions of PCA

1. Linearity.
2. Mean and variance are sufficient statistics.
   - Gaussian (normal) distribution assumed.
3. Large variances have important dynamics.

t-SNE

t-distributed Stochastic Neighbor Embedding

t-SNE – t-distributed Stochastic Neighbor Embedding

- PCA attempts to retain the global distances between data points through projection.
- However, for achieving good visualization this might not work in all cases.
- Preserving the local distances (neighborhoods) of data points is more suited for visualization.

t-SNE – t-distributed Stochastic Neighbor Embedding

- Each high-dimensional data point x is represented by a low-dimensional data point y.
- Focus on retaining the neighborhood around x.
- Distant datapoints y_i and y_j correspond to dissimilar data points.
- Offers better scalability for large, high-dimensional datasets.

t-SNE – t-distributed Stochastic Neighbor Embedding

- Procedure:

1. Compute a *N*N* similarity matrix from original data points.
2. Define a low-dimensional ”embedding” and it’s *N*N* similarity matrix.
3. Define a ”cost-function”.
4. Learn the low-dimensional representation by minimizing the cost.

t-SNE – t-distributed Stochastic Neighbor Embedding

- The similarity matrices are converted to *joint probabilities* that represent the similarities between data points.
- Similarity in high dimension as *joint probability*:

$$p_{ij} = \frac{exp(-||x_i - x_j||^2/2\sigma^2)}{\sum_{k \neq l} exp(-||x_l - x_k||^2/2\sigma^2)}$$

t-SNE – t-distributed Stochastic Neighbor Embedding

- In high-dimension the probabilites are calculated assuming Gaussian distribution.
- The low-dimensional probabilities are calculated using a heavy-tailed t-distribution:

$$q_{ij} = \frac{(1 + \|y_i - y_j\|^2)^{-1}}{\sum_{k \neq m}(1 + \|y_k - y_m\|^2)^{-1}}$$

t-SNE – t-distributed Stochastic Neighbor Embedding

- As cost-function t-SNE uses *Kullback-Leibler divergence*:

$$C = \sum_i KL(P_i \| Q_i) = \sum_i \sum_j p_{ji} \log \frac{p_{ij}}{q_{ij}}$$

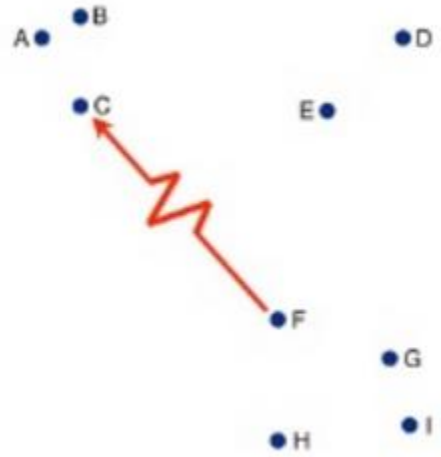
- Used to measure similarities of probability distributions.
- Large p_{ij} modeled by small q_{ij}: Large cost.
- Small p_{ij} modeled by large q_{ij}: Small cost.
- Thus, t-SNE mainly preserves the local similarities in the data set.

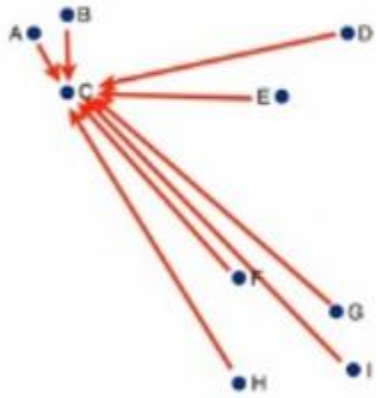
t-SNE – t-distributed Stochastic Neighbor Embedding

- Gradient of the cost function is rather simple:

$$\frac{\partial C}{\partial y} = 4 \sum_{i \neq j} (p_{ij} - q_{ij}) \frac{y_i - y_j}{1 + \|y_i - y_j\|^2}$$

- Physical analogy of attraction and repulsion.





t-SNE – t-distributed Stochastic Neighbor Embedding

- t-SNE has a tendency to form subgroups from the data.
  - This rarely causes problem in practice.
- Interpretation focus should be on the local neighborhoods, or subgroups.
- Distant samples/subgroups can only be interpreted to be dissimilar.
  - Can't interpret based on the actual distance of the samples on the embedding.

SIMLR

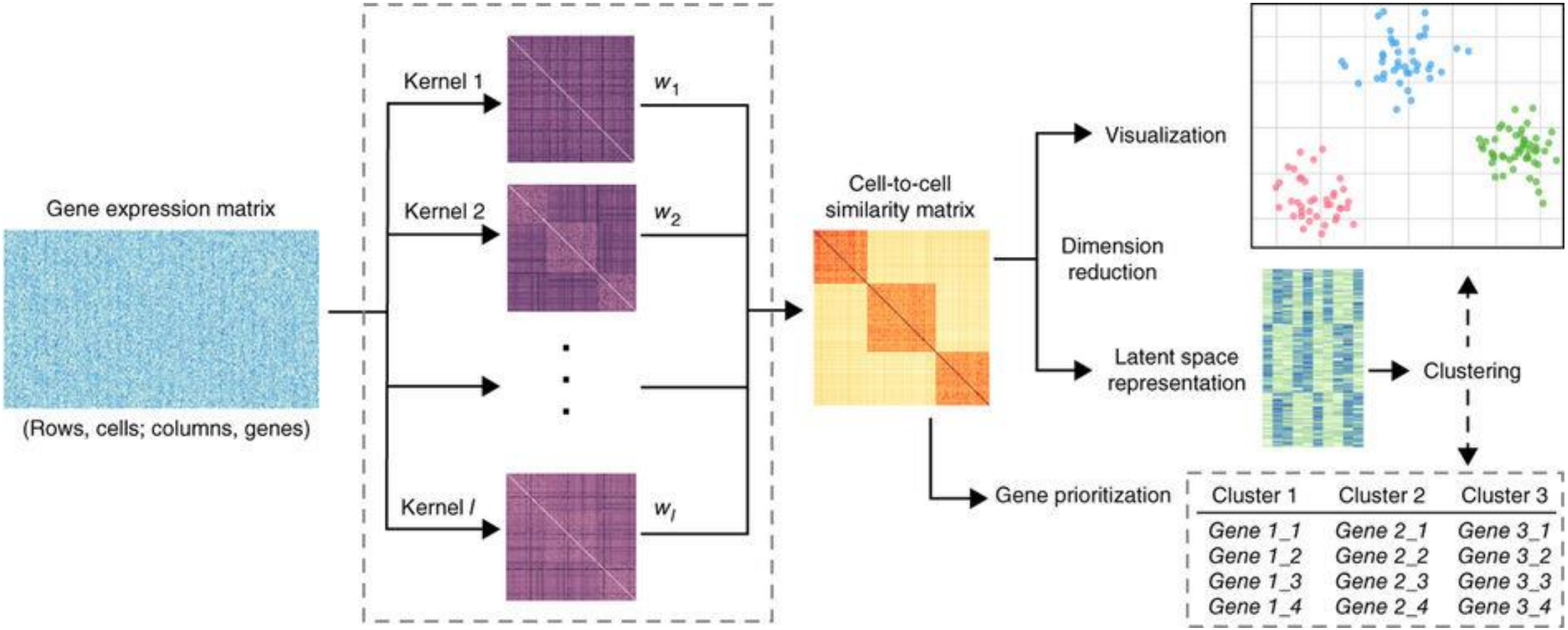
Single-cell Interpretation via Multikernel LeaRning

UEF // University of Eastern Finland

SIMLR – Single-cell Interpretation via Multikernel LeaRning

- Learns a cell-to-cell similarity measure from single-cell RNA-seq data.
- Addresses the challenge of high levels of dropouts.
- The similarity measure is then used to perform
  - Dimensionality reduction
  - Clustering
  - Visualization
- Wang *et al*. 2017 Nature Methods

SIMLR – Single-cell Interpretation via Multikernel LeaRning

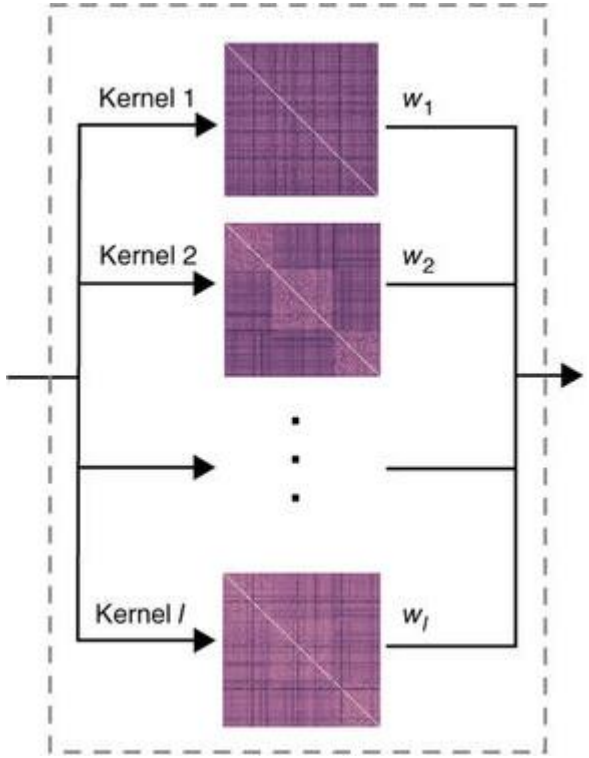


SIMLR – Single-cell Interpretation via Multikernel LeaRning

- l Gaussian kernels used to model the pairwise cell distances.
- General form of the distance between cell i and cell j:

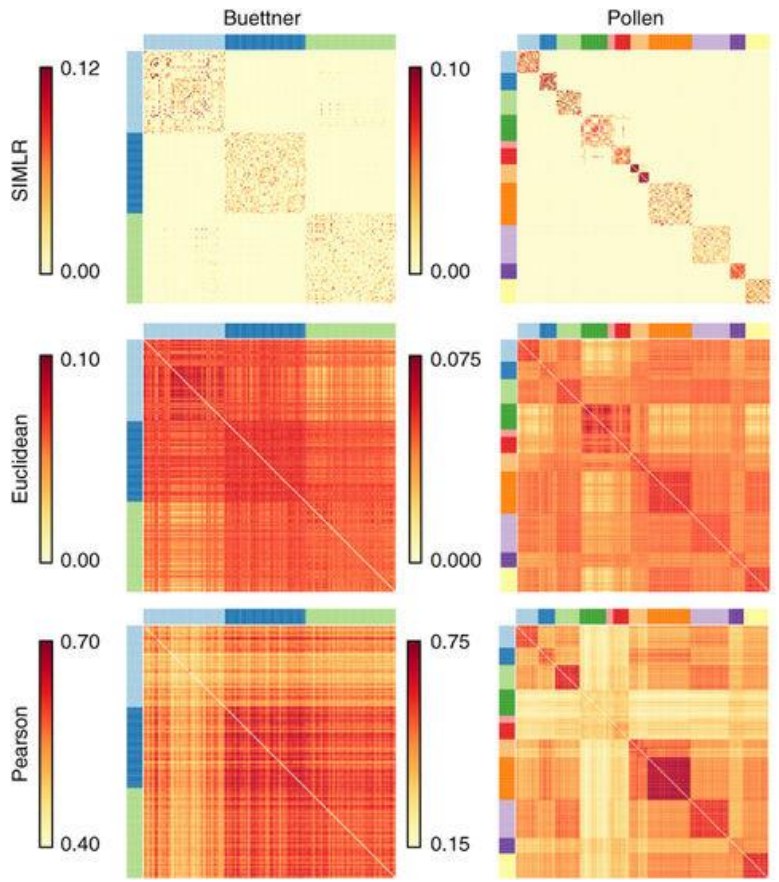
$$D\left(c_i, c_j\right) = 2 - 2\sum_l w_l K_l\left(c_i, c_j\right)$$

- Multiple kernels
  - correspond to different informative representations
  - are often more flexible than a single kernel



SIMLR – Single-cell Interpretation via Multikernel LeaRning

- If *C* separable subpopulations exist, similarity matrix *S* should have an approximate block-diagonal structure with *C* blocks.



SIMLR – Single-cell Interpretation via Multikernel LeaRning

- Optimization:

$$\underset{S,L,w}{\text{minimize}} - \sum_{i,j,l} w_l K_l\left(c_i, c_j\right) S_{ij} + \beta \| S \|_F^2 + \gamma \, \mathbf{tr}(L^T (I_N - S) L) + \rho \sum_l w_l \log w_l$$

- Optimize over S, L and w.
  - **Nonconvex**, but objective function for each variable conditional on the other two variables being fixed is **convex**.
- S: Similarity matrix
- L: rank-enforcing auxiliary matrix
- w: weight vector

SIMLR – Single-cell Interpretation via Multikernel LeaRning

- Optimization:

$$\underset{S,L,w}{\text{minimize}} \; -\sum_{i,j,l} w_l K_l\left(c_i, c_j\right) S_{ij} + \beta \| S \|_F^2 + \gamma \, \mathbf{tr}(L^T (I_N - S) L) + \rho \sum_l w_l \log w_l$$

Forces small similarity for distant cells

- Optimize over S, L and w.
  - **Nonconvex**, but objective function for each variable conditional on the other two variables being fixed is **convex**.
- S: Similarity matrix
- L: rank-enforcing auxiliary matrix
- w: weight vector

SIMLR – Single-cell Interpretation via Multikernel LeaRning

- Optimization:

$$\underset{S,L,w}{\text{minimize}} - \sum_{i,j,l} w_l K_l\left(c_i, c_j\right) S_{ij} + \beta \| S \|_F^2 + \gamma \, \mathbf{tr}(L^T (I_N - S) L) + \rho \sum_l w_l \log w_l$$

Regularize *S*

- Optimize over S, L and w.
  - **Nonconvex**, but objective function for each variable conditional on the other two variables being fixed is **convex**.
- S: Similarity matrix
- L: rank-enforcing auxiliary matrix
- w: weight vector

SIMLR – Single-cell Interpretation via Multikernel LeaRning

- Optimization:

$$\underset{S,L,w}{\text{minimize}} - \sum_{i,j,l} w_l K_l\left(c_i, c_j\right) S_{ij} + \beta \| S \|_F^2 +$$

$$\gamma\, \mathbf{tr}(L^T (I_N - S) L) + \rho \sum_l w_l \log w_l$$

Forces block-structure

- Optimize over S, L and w.
  - **Nonconvex**, but objective function for each variable conditional on the other two variables being fixed is **convex**.
- S: Similarity matrix
- L: rank-enforcing auxiliary matrix
- w: weight vector

SIMLR – Single-cell Interpretation via Multikernel LeaRning

- Optimization:

$$\underset{S,L,w}{\text{minimize}} - \sum_{i,j,l} w_l K_l\left(c_i, c_j\right) S_{ij} + \beta \| S \|_F^2 +$$

$$\gamma\, \mathbf{tr}(L^T (I_N - S) L) + \rho \sum_l w_l \log w_l$$

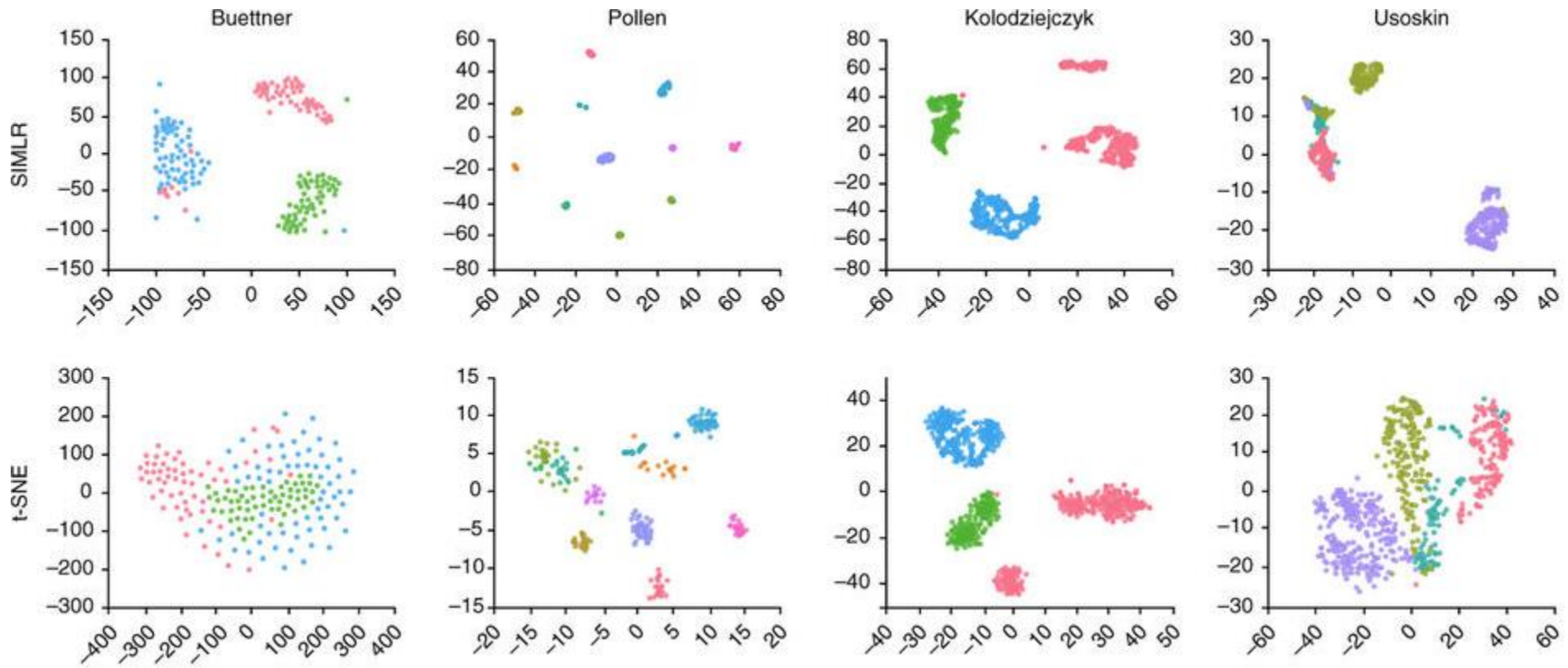
Regularize kernel weights

- Optimize over S, L and w.
  - **Nonconvex**, but objective function for each variable conditional on the other two variables being fixed is **convex**.
- S: Similarity matrix
- L: rank-enforcing auxiliary matrix
- w: weight vector

SIMLR – Single-cell Interpretation via Multikernel LeaRning

- Dimensionality reduction with SIMLR uses t-SNE.
- Euclidean distance matrix calculated in t-SNE.
- Replace this with the (dis)similarity matrix from SIMLR.

SIMLR – Single-cell Interpretation via Multikernel LeaRning



Dimensionality reduction evaluation

Trustworthiness & continuity

- Rank-based measures where ranks are given to each datapoint based on how close they are to another datapoint.
- Unsupervised measures.
- Trustwothiness
  - Are neighbors of x in the embedding actually neighbors in the original data?
- Continuity
  - Are neighbors of x in the original data still neighbors in the embedding?

Trustworthiness & continuity

- Trustworthiness

$$M_{trust}(k) = 1 - A(k) \sum_{i=1}^{N} \sum_{j \in U_k(i)} (r(i,j) - k)$$

- Continuity

$$M_{cont}(k) = 1 - A(k) \sum_{i=1}^{N} \sum_{j \in V_k(i)} (\hat{r}(i,j) - k)$$

- Normalization term

$$A(k) = \begin{cases} \dfrac{2}{Nk(2N - 3k - 1)}, & \text{if } k < \dfrac{N}{2} \\ \dfrac{2}{N(N-k)(N-k-1)}, & \text{if } k \geq \dfrac{N}{2} \end{cases}$$

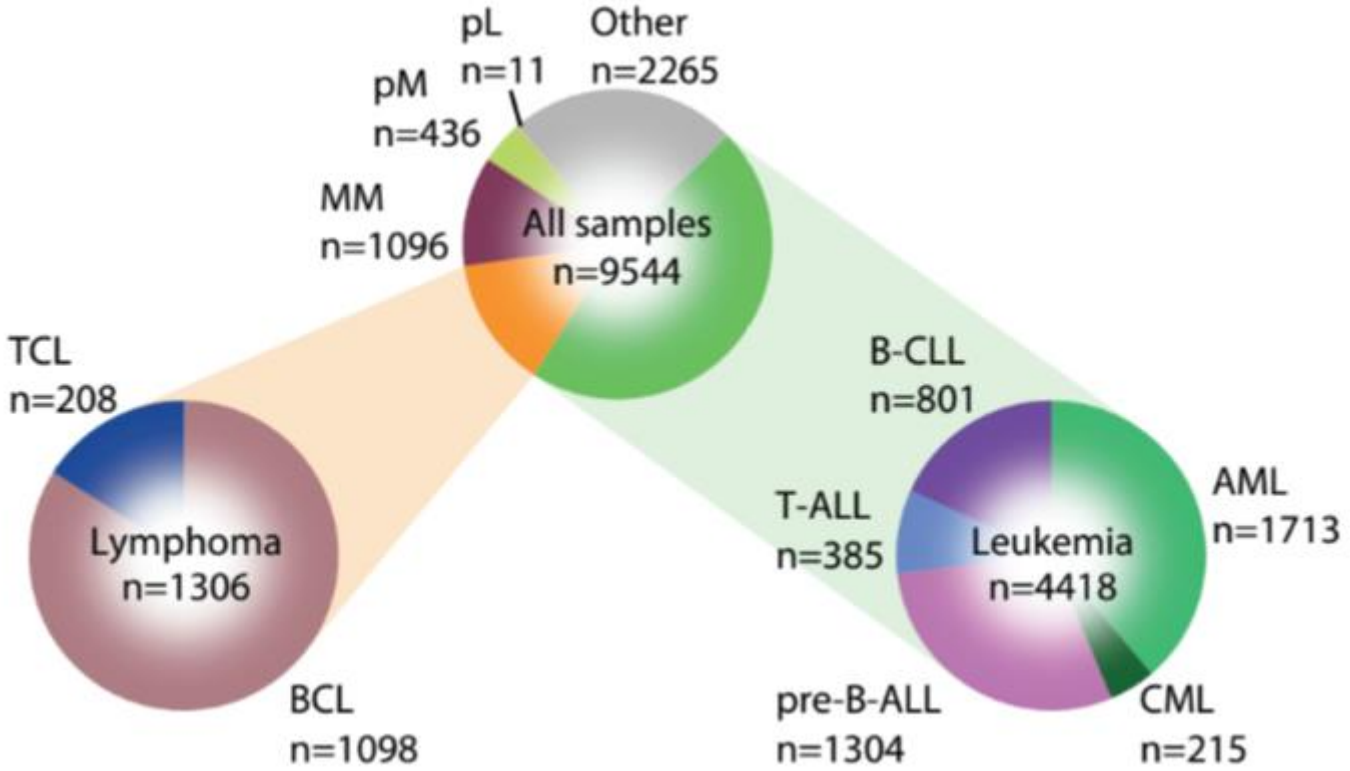
Nearest Neighbor Error

- k-Nearest Neighbor classifier used to evaluate goodness of low-dimensional embedding.
- When classifying, only consider the closest datapoint (k = 1).
- Calculate average classification error using cross validation.
- Requires class labels, i.e. supervised measure.

UNIVERSITY OF EASTERN FINLAND

Hemap

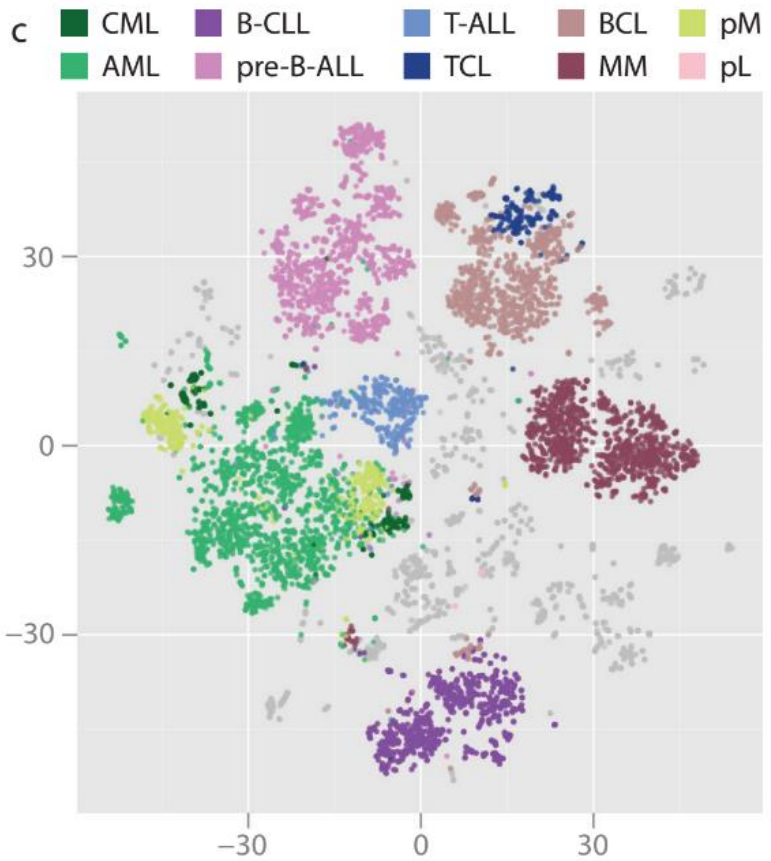
Hemap



Mehtonen *et al*. unpublished

Hemap

- Publicly available microarray data from multiple labs and experiments.
- Put together through bias correction, normalization.
- Embedding with t-SNE following feature selection.
- Mean-shift clustering done on top of embedding reveals subtypes.



Mehtonen *et al*. unpublished

Hemap

- Different dimensionality reduction techniques were tested.

Method	Trustworthiness	Continuity	1-NN error
BH-SNE	**0.9941**	**0.9877**	0.0583
GPLVM (DTC)	0.9710	0.9567	0.1523
GPLVM (FITC)	0.9551	0.9553	0.1702
GPLVM (variational DTC)	0.9792	0.9572	0.1453
LLE	0.9182	0.9645	0.1323
PCA	0.8047	0.9493	0.5128
Sammon mapping	0.83	0.9403	0.4966
SNE	0.6173	0.6406	0.7399
t-SNE	0.9937	0.9833	**0.0479**