Linear Dimensionality Reduction in R

Nikolay Oskolkov, MRG Group Leader, LIOS, Riga, Latvia

R course, 09.02.2026

Latvian Institute of Organic Synthesis

@NikolayOskolkov

@oskolkov.bsky.social

Personal homepage:
https://nikolay-oskolkov.com

Image generated by ChatGPT

Session content

Topics we'll cover in this session:

1) Dimensionality reduction is more than just visualization

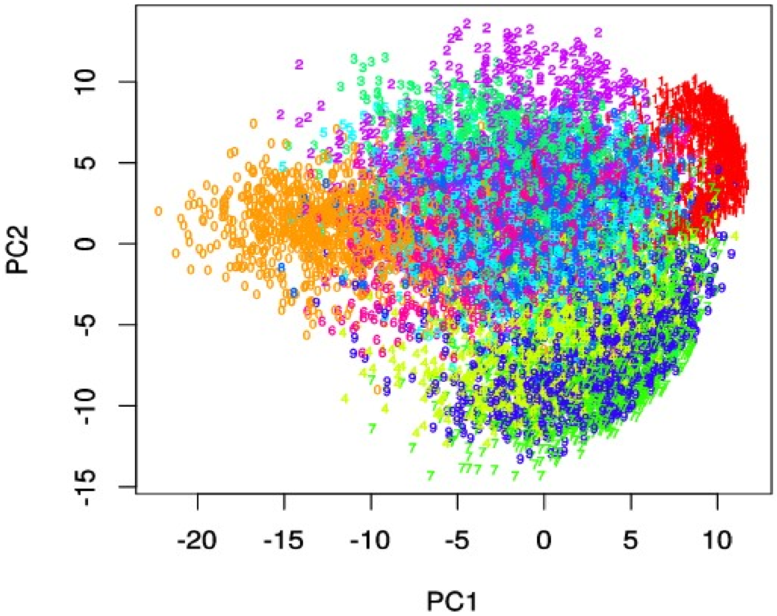
2) Linear and non-linear dimensionality reduction techniques overview

3) Matrix factorization as a key principle of linear dimensionality reduction

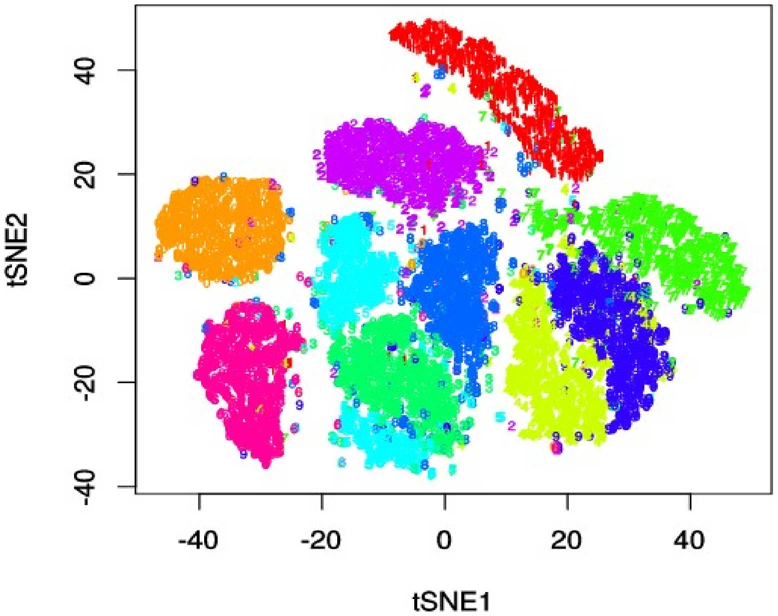
4) Limitations of linear dimensionality reduction methods and need for more

Dimension reduction: more than visualization

PCA PLOT WITH PRCOMP



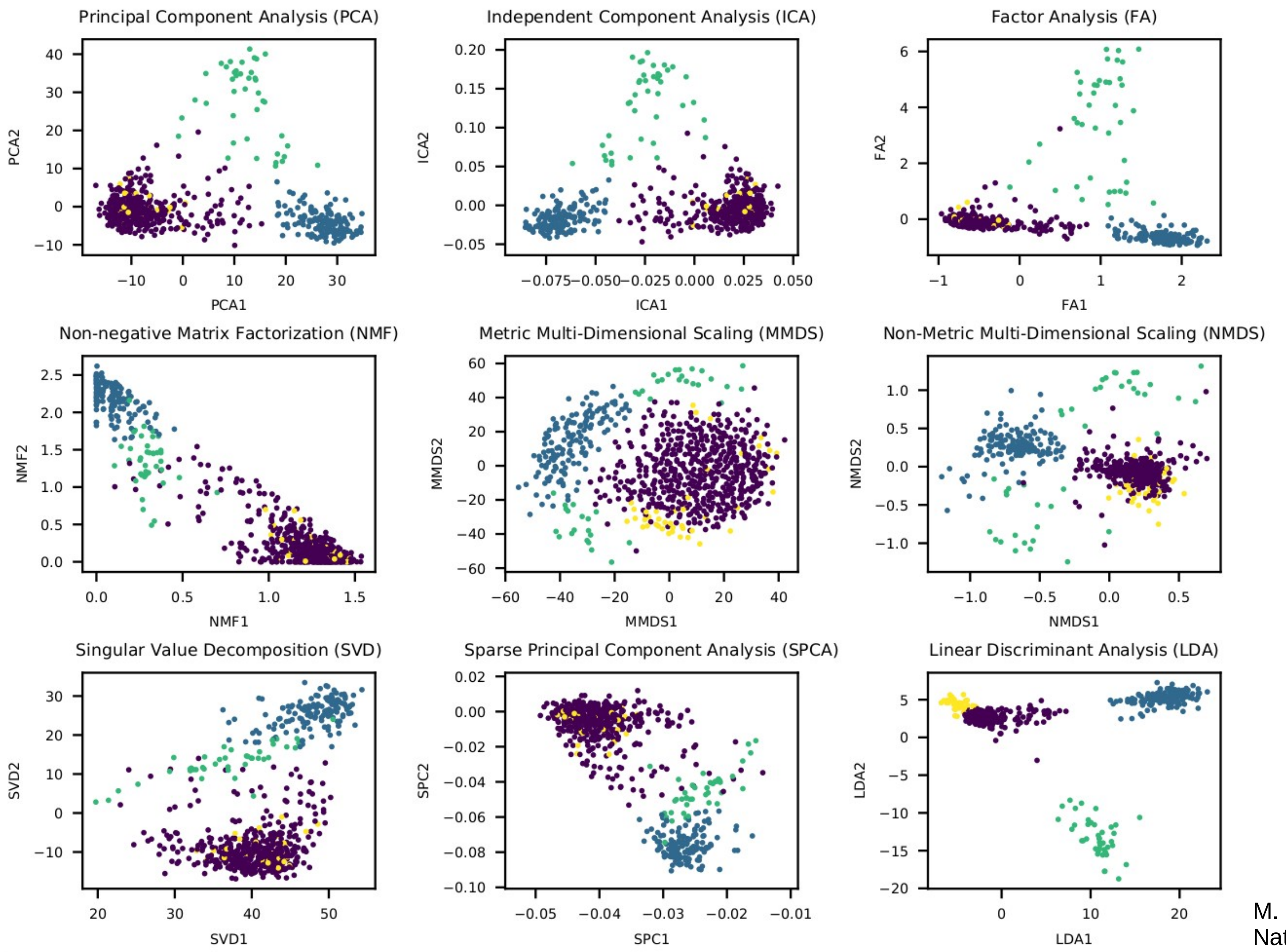
tSNE MNIST



The goal of dimension reduction is not only visualization but also <u>reducing dimensions</u>

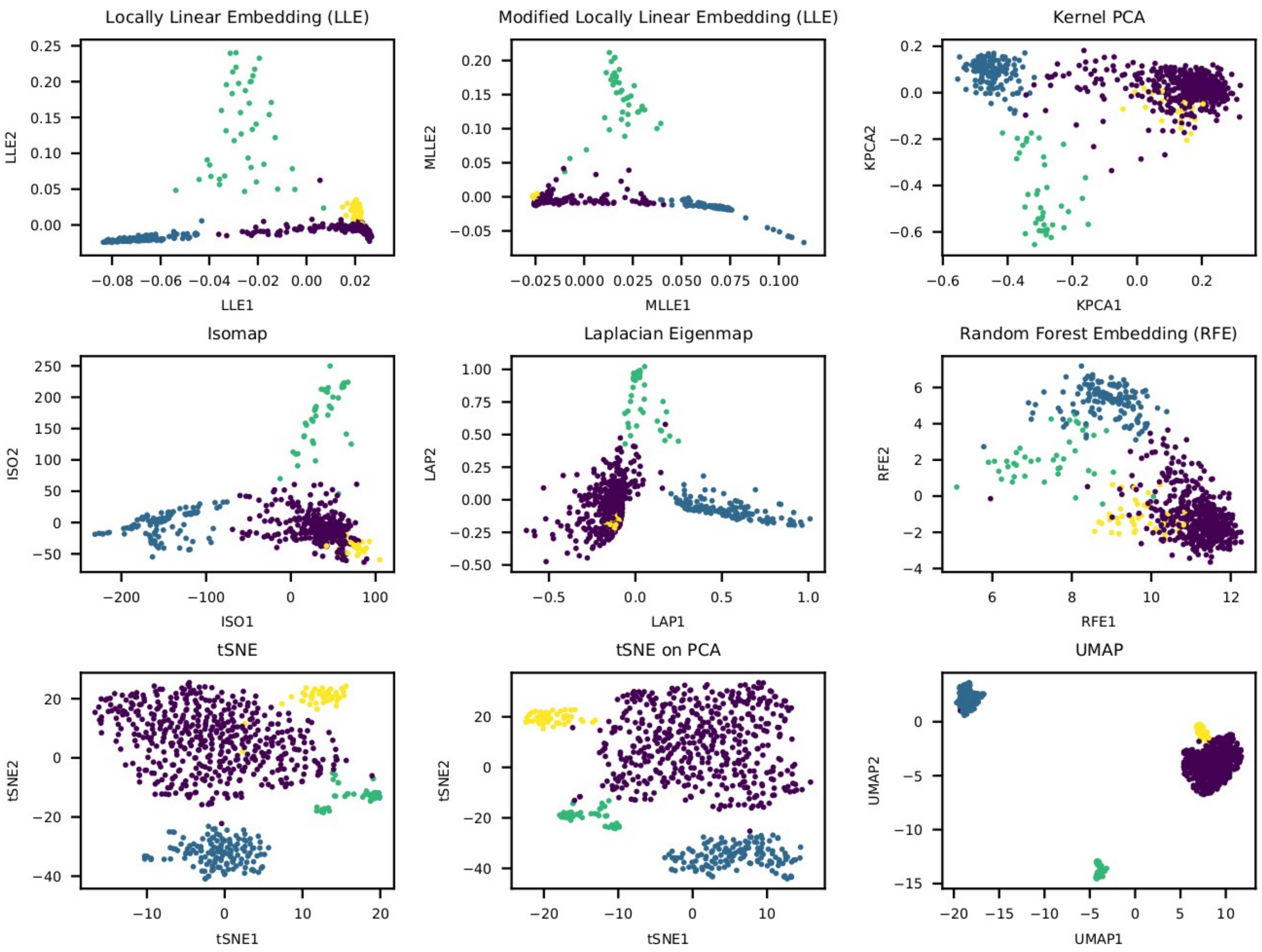
Linear dimensionality reduction

Latvian Institute of Organic Synthesis



M. Bartoschek, N. Oskolkov et al., Nature Communications 2018

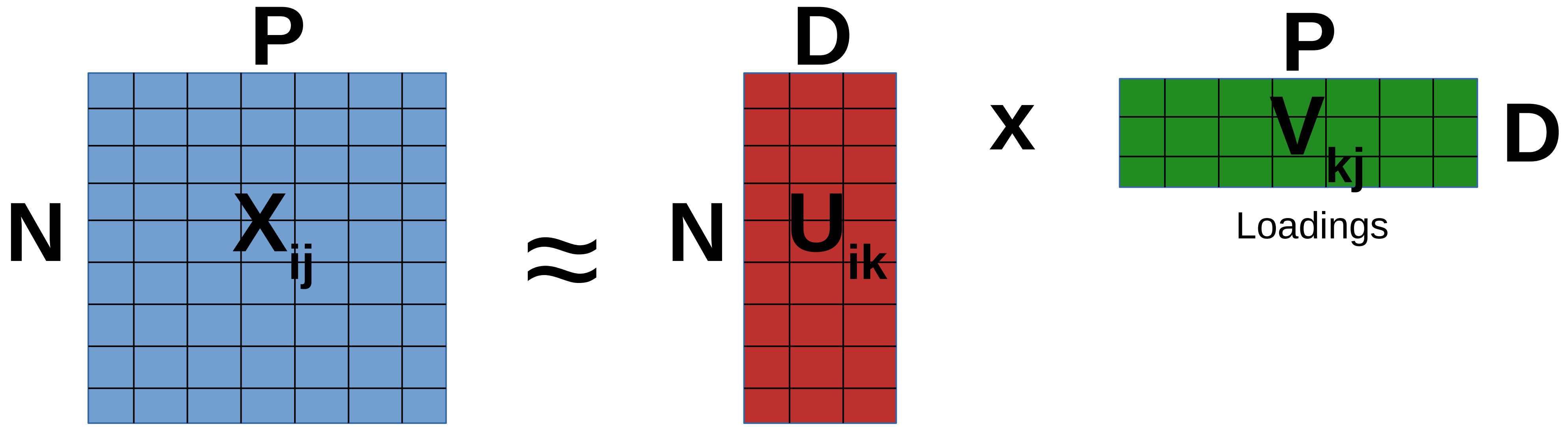
Non-linear dimensionality reduction



M. Bartoschek, N. Oskolkov et al., Nature Communications 2018

Linear dimension reduction: matrix factorization

$$\mathbf{X_{ij}} \approx \mathbf{U_{ik}} \mathbf{V_{kj}}$$



Data

Low-dimensional data representation (embeddings)

Loadings

$$\mathrm{Loss} = \sum_{i=1}^{N} \sum_{j=1}^{P} (\mathbf{X_{ij}} - \mathbf{U_{ik}} \mathbf{V_{kj}})^2$$

PCA dimension reduction algorithm

Coding in R:

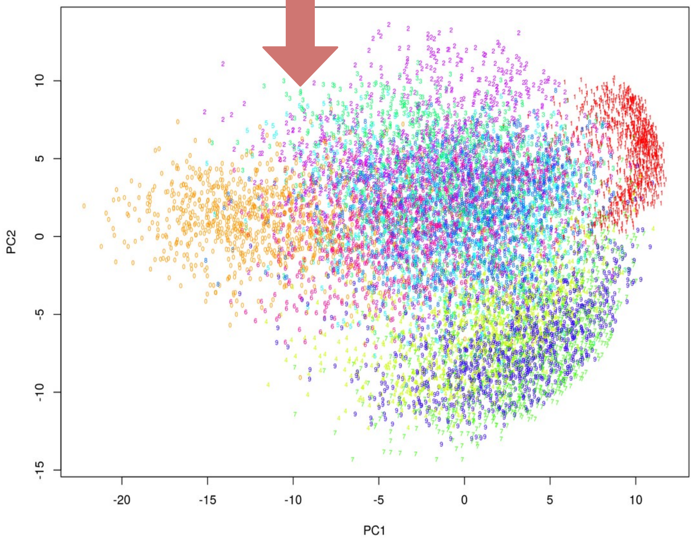
```
data_centered <- scale(data, center = TRUE, scale = FALSE)

covariance <- t(data_centered) %*% data_centered

eig <- eigen(covariance)

plot(eig$vectors[,1:2]);                barplot(eig$values / sum(eig$values))
```

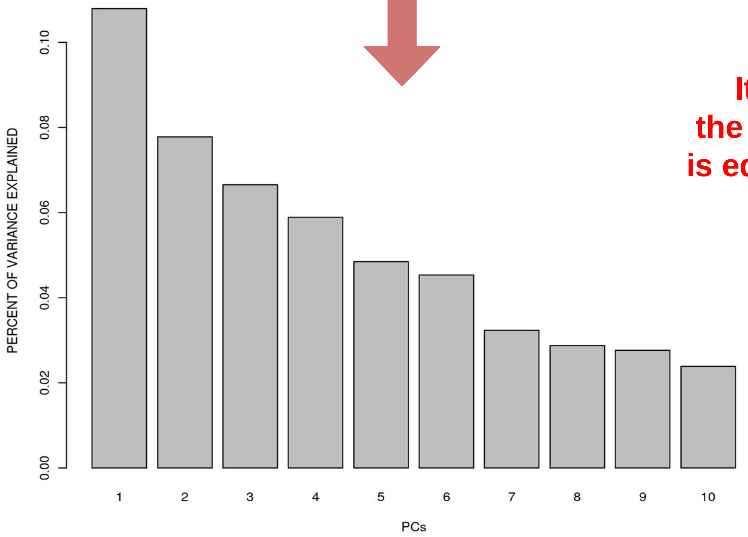
Mathematically:

$M_{ij} = X_{ij} – \mu_j$

$A = (1/N)*M^T*M$

$A*u = \lambda*u$

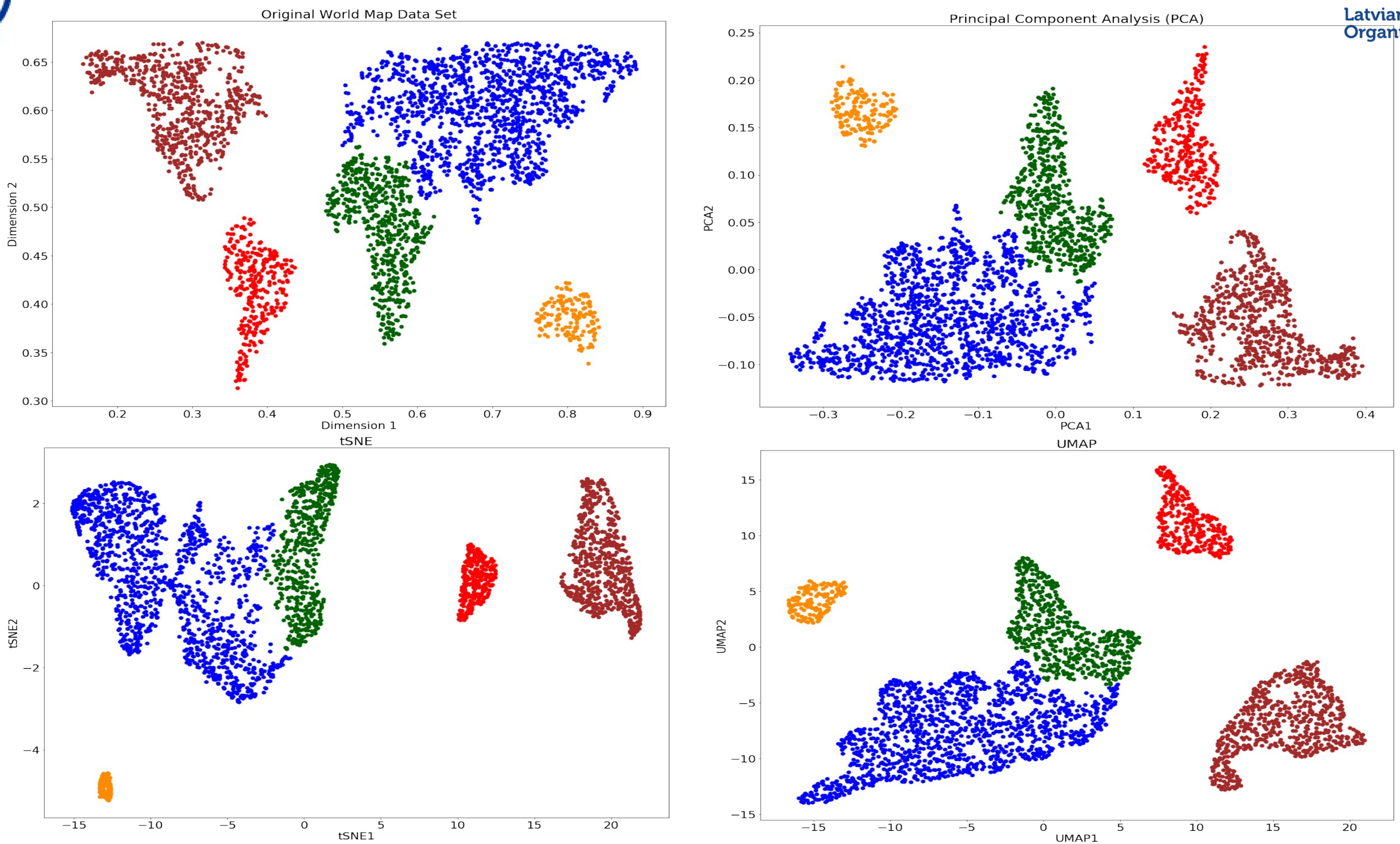




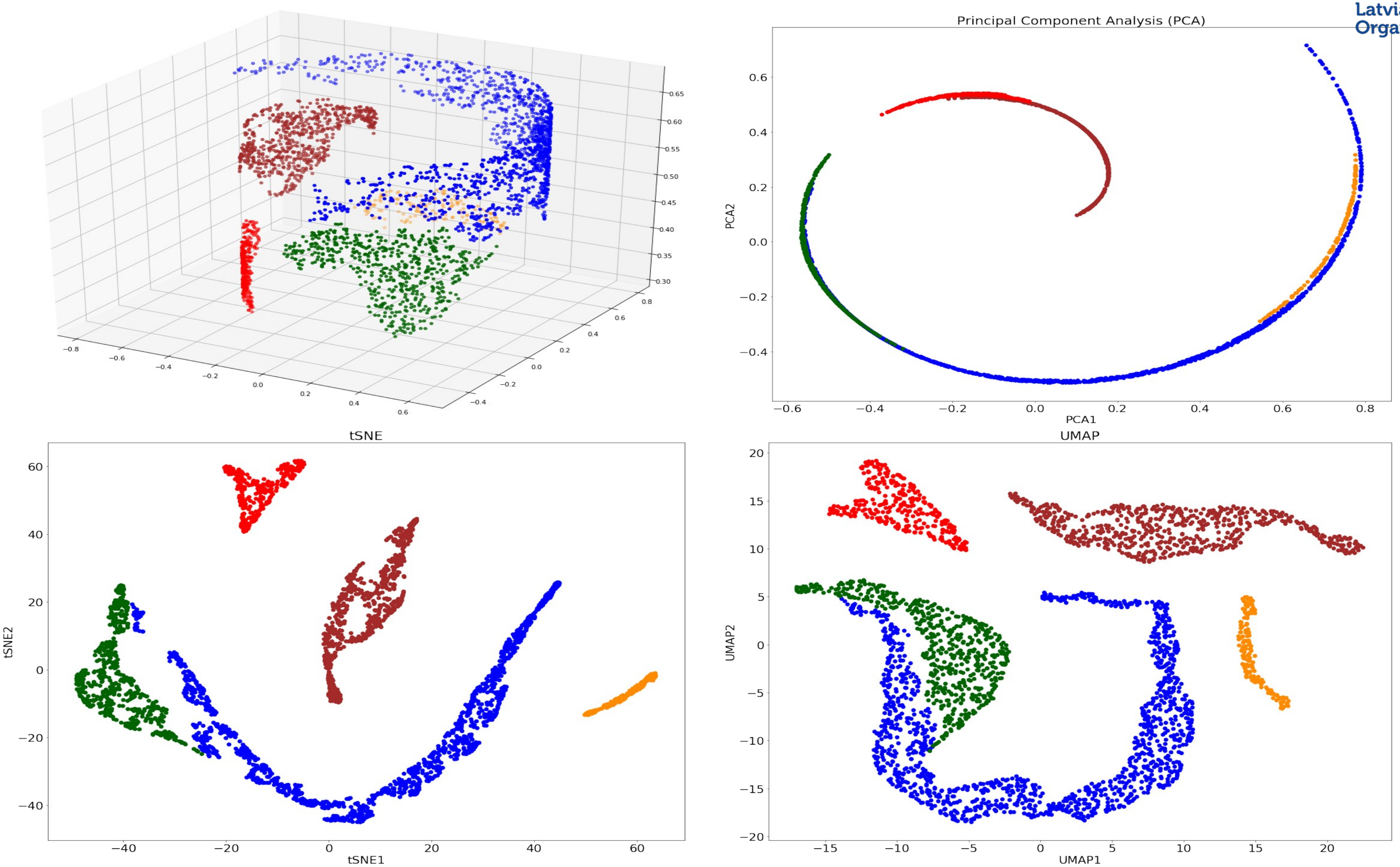
It can be analytically derived that the eigen value decomposition in PCA is equivalent to projecting data on axes of maximal variation in the data

PCA works fine on a linear manifold

Latvian Institute of Organic Synthesis

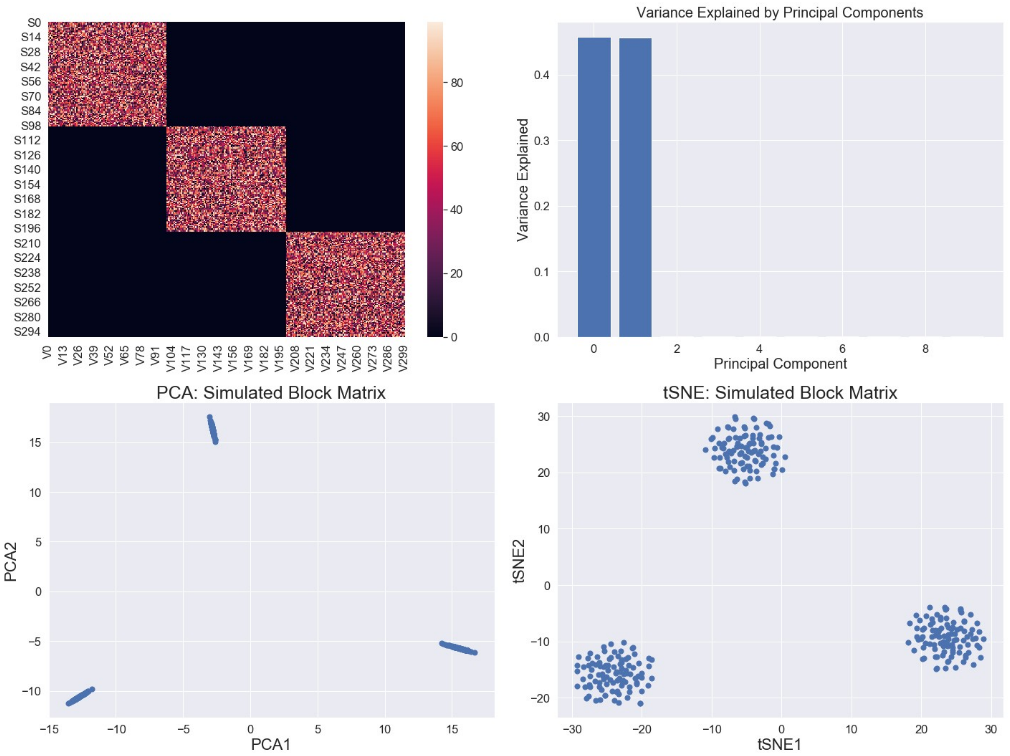


PCA vs. tSNE vs. UMAP on non-linear manifold

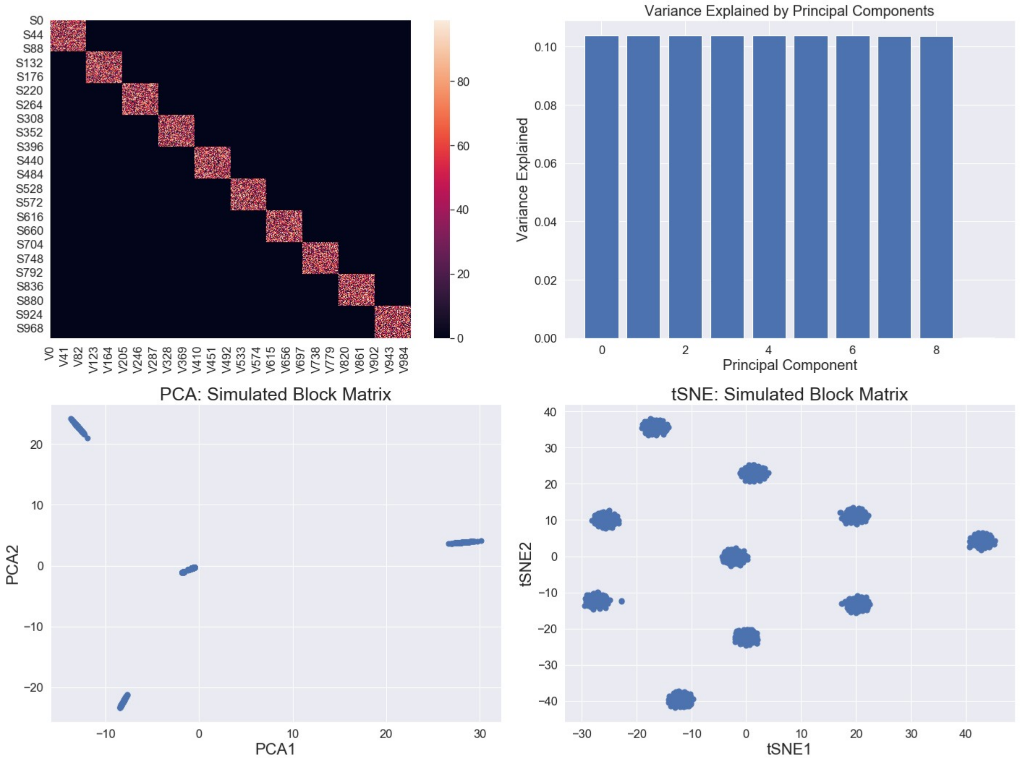


PCA vs. tSNE
when number of populations increases

Three classes of data points



Ten classes of data points



Session summary

Take home messages of the session:

1) Dimensionality reduction function goes beyond simple data visualization as it helps to overcome the curse of dimensionality

2) Matrix factorization is a key principle of linear dimensionality reduction methods

3) Eigen vectors computed via PCA capture directions of maximal variation in the data

4) Data on a non-linear manifold cannot be correctly resolved by PCA, hence tSNE / UMAP are more informative

Acknowledgments: LIOS + TARGETWISE

Latvian Institute of Organic Synthesis

2027
National Development Plan

Funded by
the European Union