A kNN-LWPLSR pipeline

matthieu.lesnoff@cirad.fr

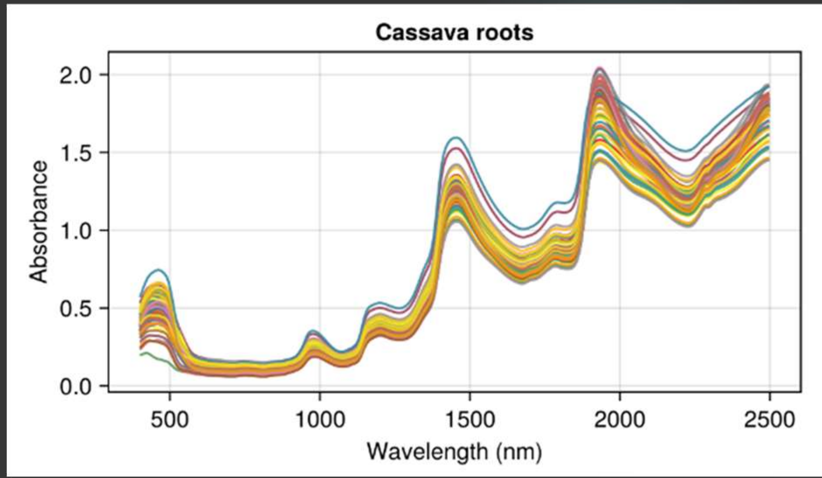
Cirad, UMR Selmet

The algorithm is

- useful when non-linearity between X and Y

  (data heterogeneity, etc.)

- Very performant for NIR data



Available pipelines

Journal of CHEMOMETRICS

RESEARCH ARTICLE

Comparison of locally weighted PLS strategies for regression and discrimination on agronomic NIR data

Matthieu Lesnoff, Maxime Metz, Jean-Michel Roger

First published: 16 January 2020 | https://doi.org/10.1002/cem.3209 | Citations: 30

ELSEVIER

Chemometrics and Intelligent Laboratory Systems

Volume 244, 15 January 2024, 105031

Averaging a local PLSR pipeline to predict chemical compositions and nutritive values of forages and feed from spectral near infrared data

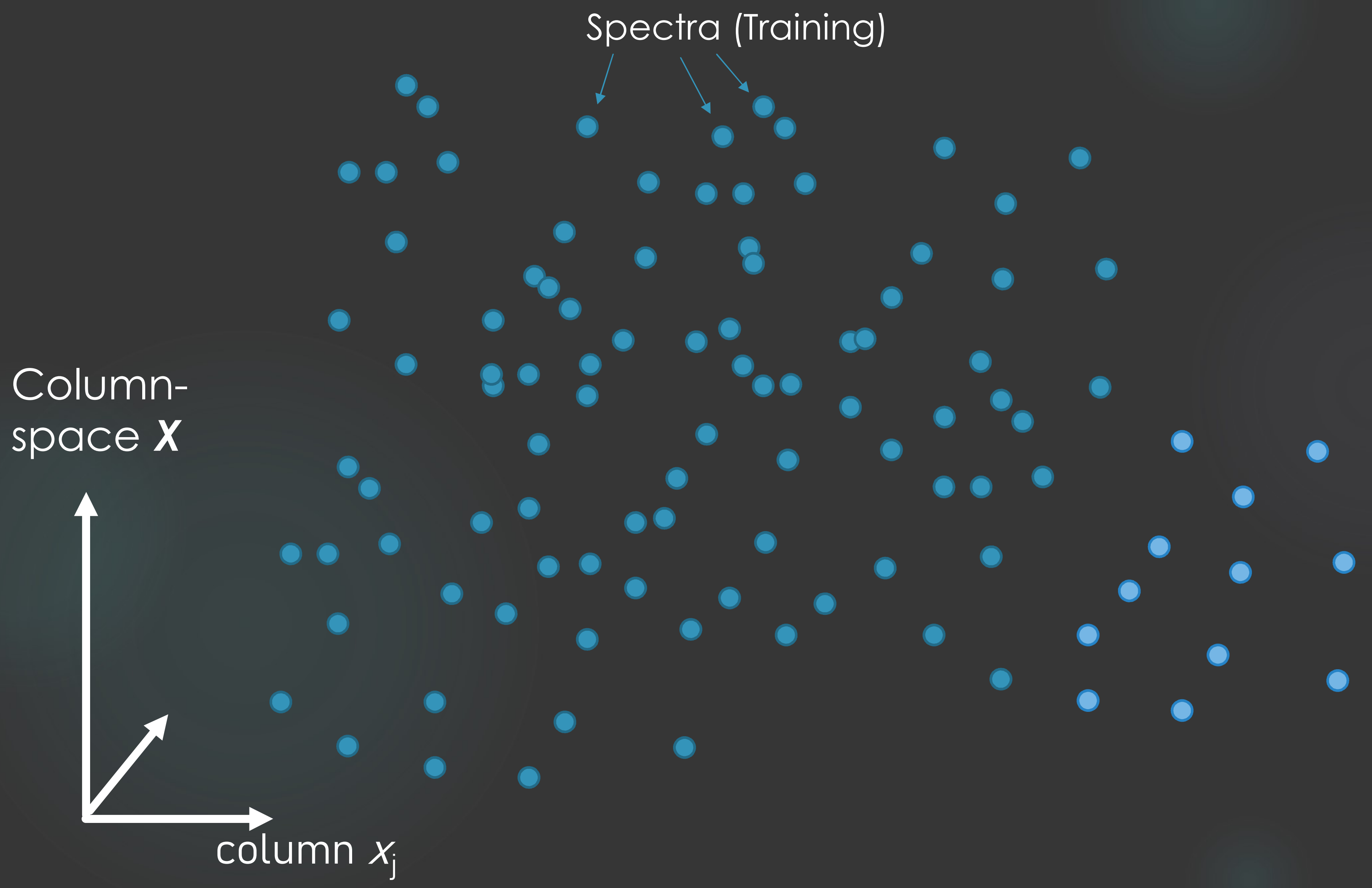
Matthieu Lesnoff [a b c]

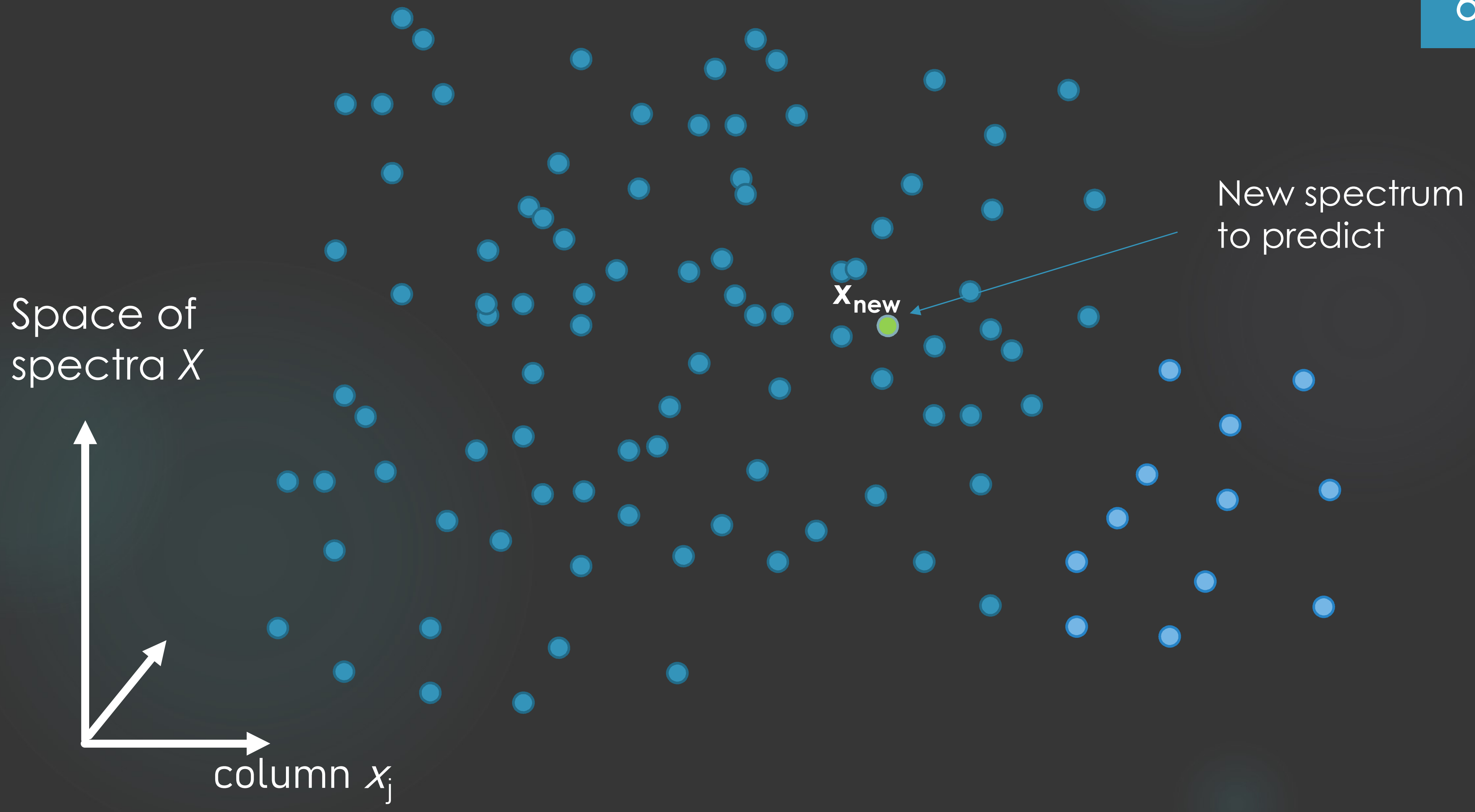
Two steps

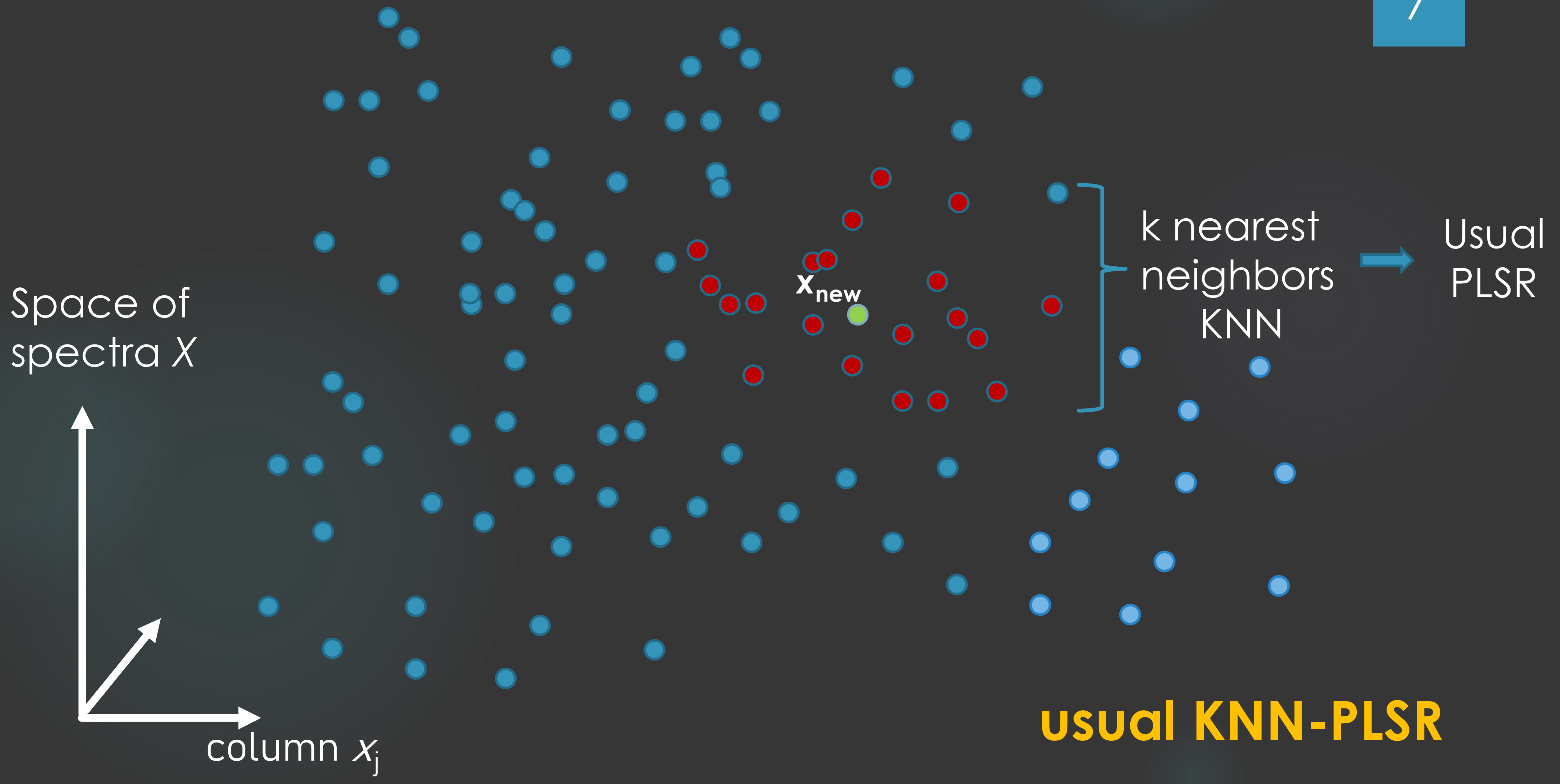
KNN : Selection of k nearest neighbors

LWPLSR : Locally weighted partial least squares regression on the neighborhood

(If discrimination $\Rightarrow$ LWPLSDA)

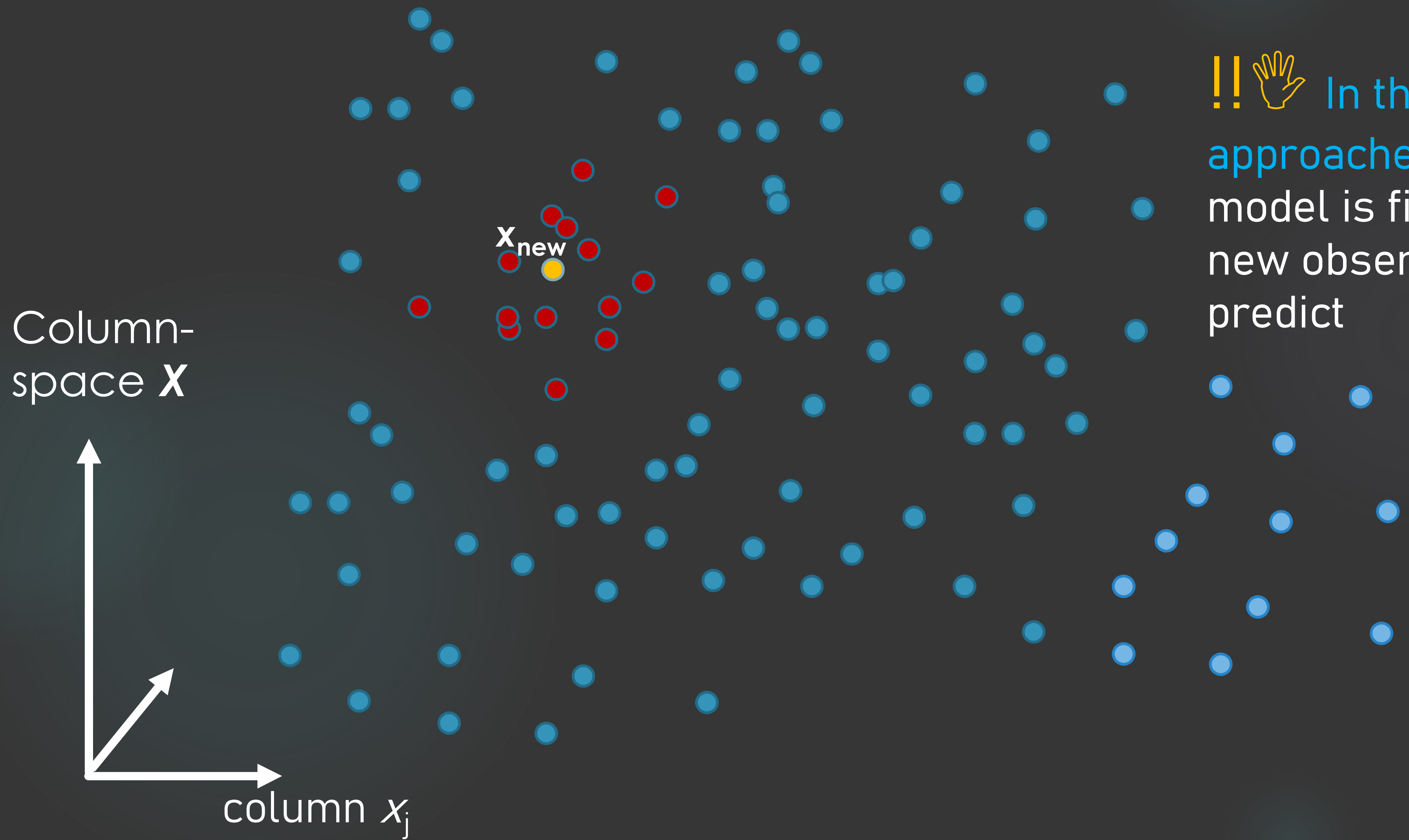

Spectra (Training)
Column-space ***X***
column x_j


Space of
spectra X
column xj
xnew
New spectrum
to predict



usual KNN-PLSR

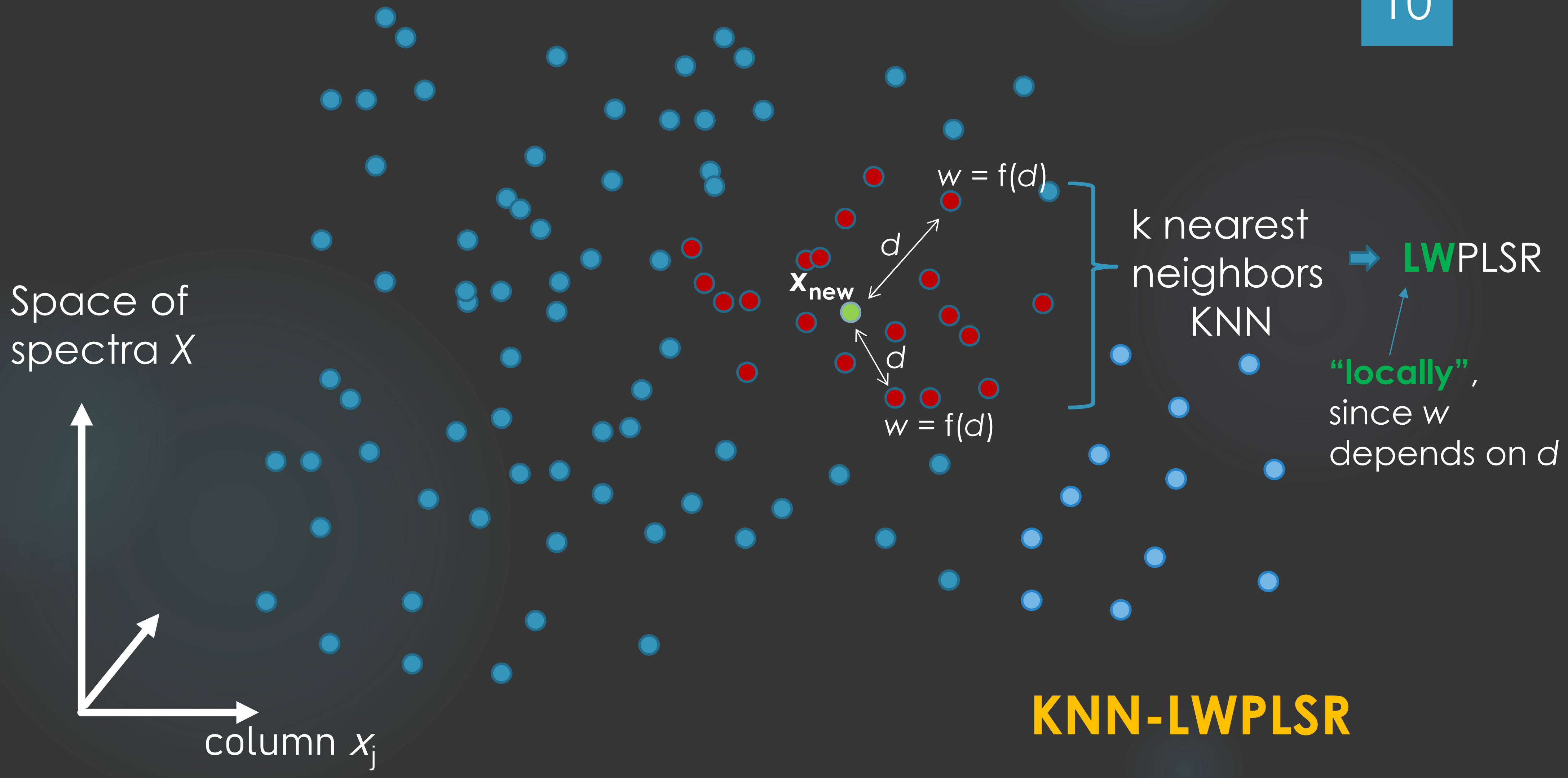
!!✋ In the local approaches, a new model is fitted for each new observation to predict



- Usual PLSR $\max_{\boldsymbol{t}} \mathrm{Cov}(\boldsymbol{t}, \boldsymbol{y})^2 = \sum_{i=1}^{n} \left(\frac{\mathbf{1}}{\boldsymbol{n}} t_i y_i\right)^2$

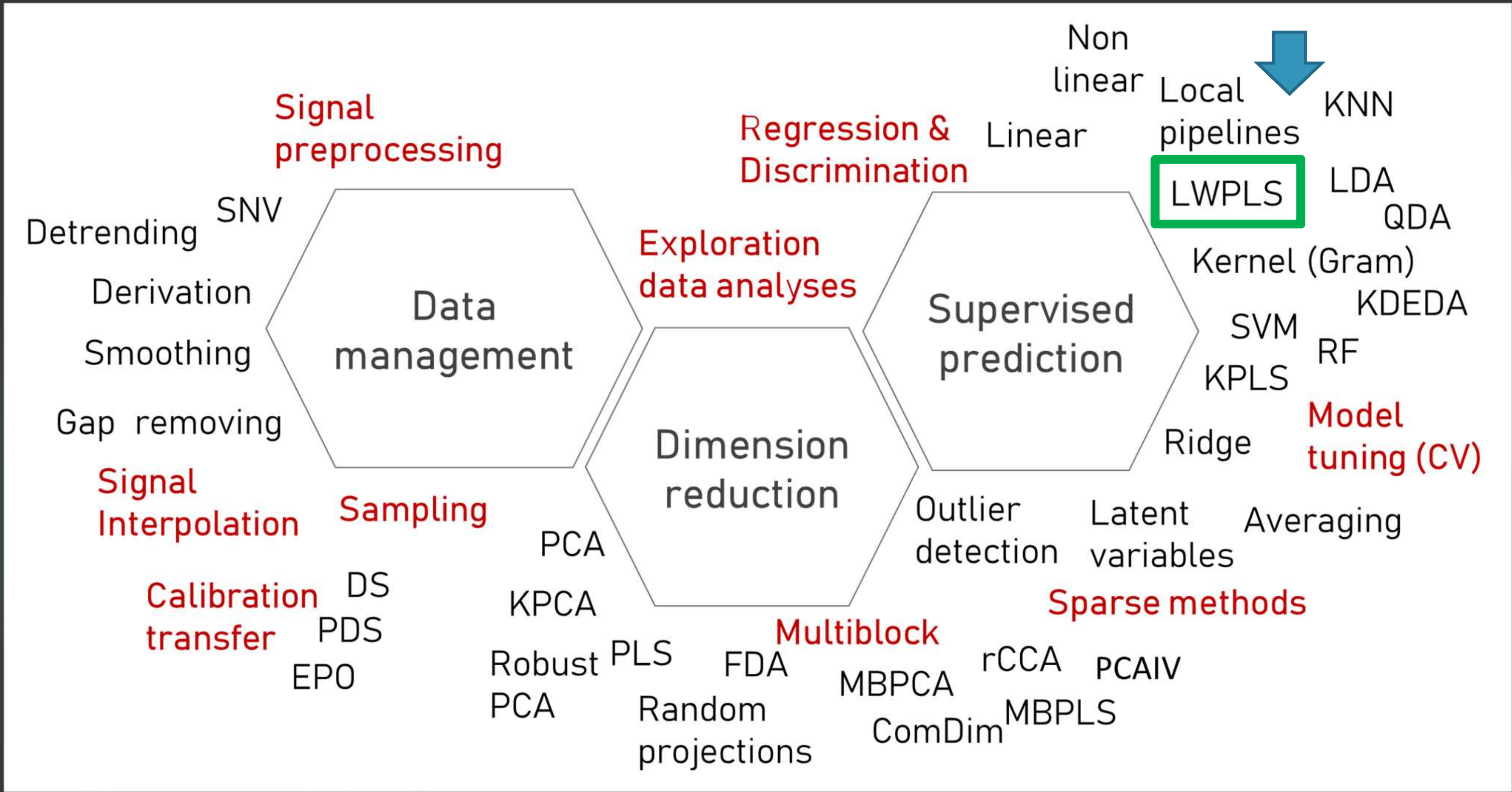
Extension:

- Weighted PLSR (WPLSR) $\max_{\boldsymbol{t}} \mathrm{Cov}_{\boldsymbol{W}}(\boldsymbol{t}, \boldsymbol{y})^2 = \sum_{i=1}^{n} (\boldsymbol{w_i} t_i y_i)^2$


Space of
spectra X
column x_j
x_{new}
d
d
w = f(d)
w = f(d)
k nearest
neighbors
KNN
LWPLSR
"locally",
since w
depends on d

KNN-LWPLSR

Implementation in Package Jchemo



https://mlesnoff.github.io/Jchemo.jl/dev/domains

Function `lwplsr`

Five arguments

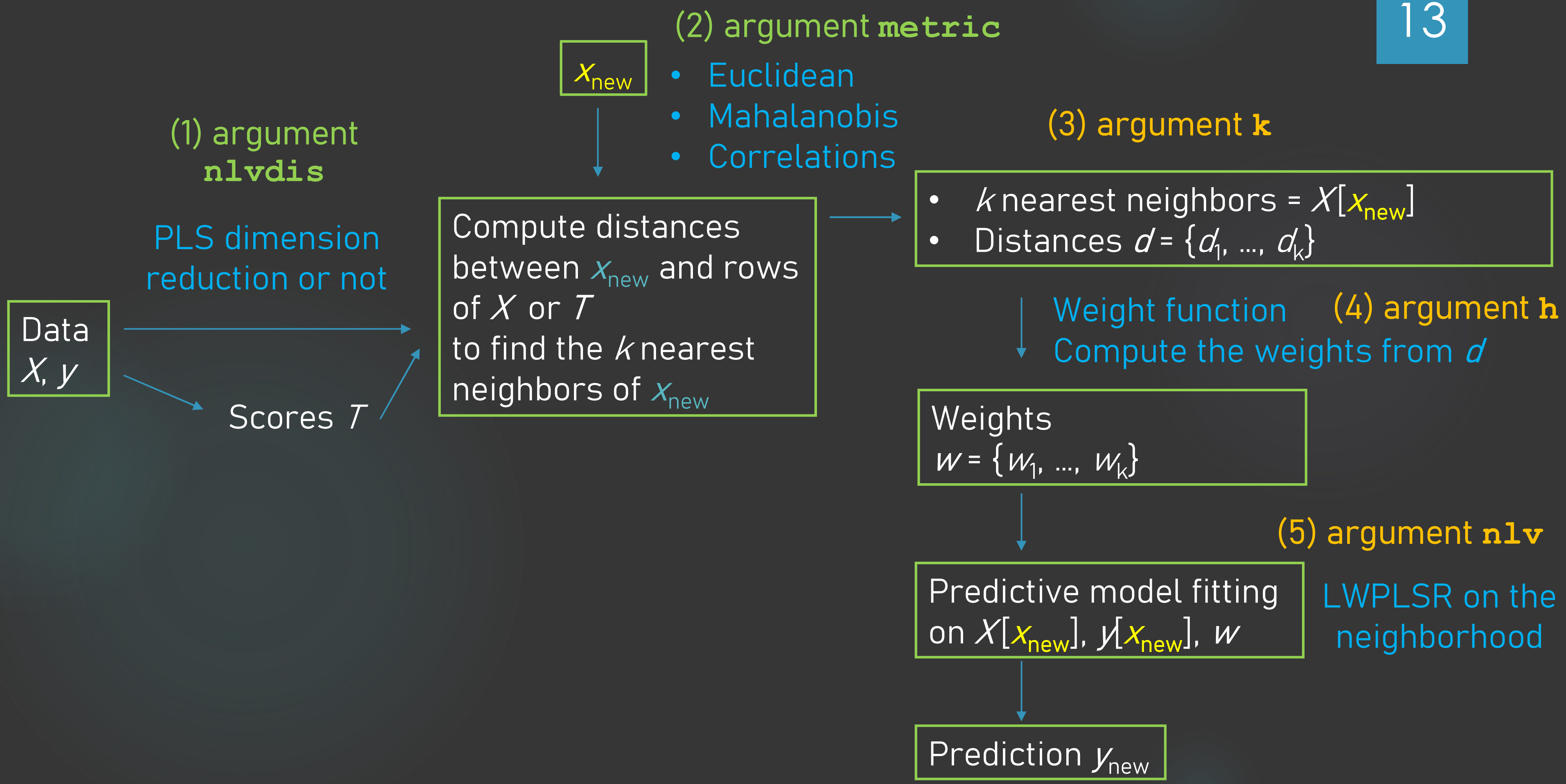
1) `nlvdis` : Space used to compute the distances. X (nlvdis = 0) or nlvdis (>0) global PLS scores (matrix T)

2) `metric` : Metric used to compute the distances (Euclidean, Mahalanobis, correlations)

3) `h` : Sharpness of the weight function

4) `k` : Nb. neighbors

5) `nlv` : Nb. LVs for each local PLSR model



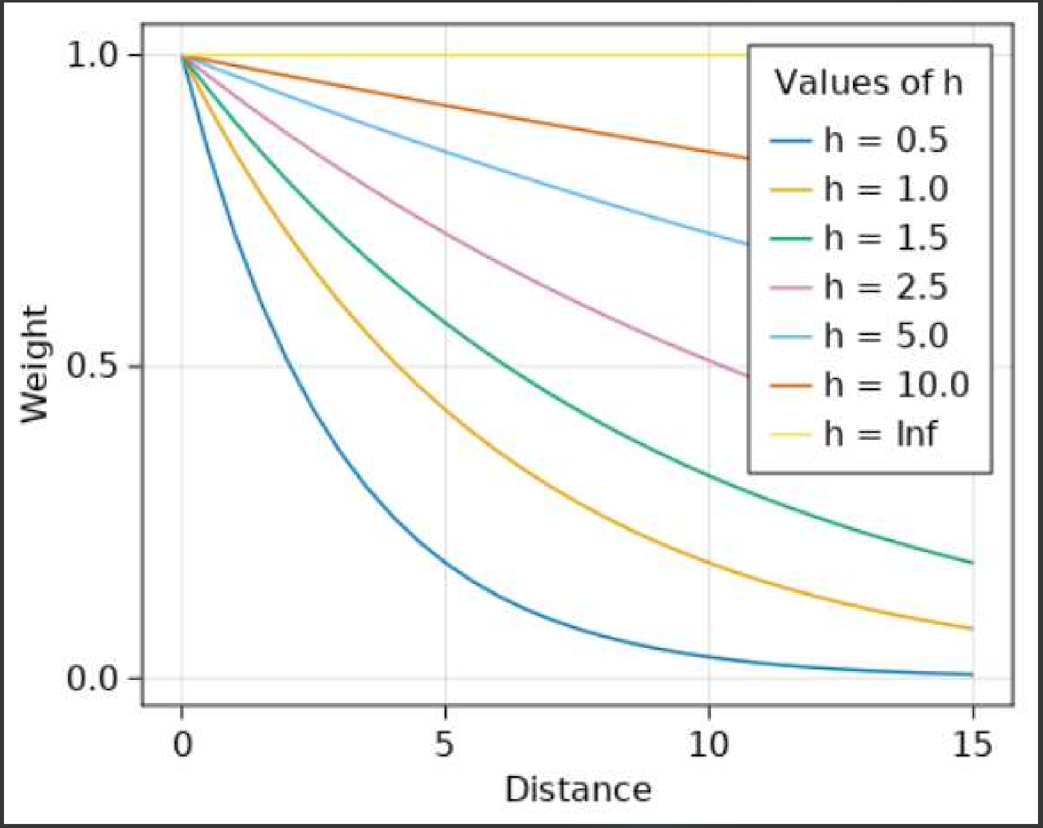
(1) argument **nlvdis**
PLS dimension reduction or not
Data
X, y
Scores T
x_{new}
(2) argument **metric**
• Euclidean
• Mahalanobis
• Correlations
Compute distances between x_{new} and rows of X or T to find the k nearest neighbors of x_{new}
(3) argument **k**
• k nearest neighbors = $X[x_{new}]$
• Distances $d = \{d_1, \ldots, d_k\}$
Weight function
Compute the weights from d
(4) argument **h**
Weights
$w = \{w_1, \ldots, w_k\}$
(5) argument **nlv**
Predictive model fitting on $X[x_{new}]$, $y[x_{new}]$, w
LWPLSR on the neighborhood
Prediction y_{new}

Weight function implemented in `lwplsr`

This is an adaptation from *Kim S, Kano M, Nakagawa H, Hasebe S. Estimation of active pharmaceutical ingredients content using locally weighted partial least squares and statistical wavelength selection. Int J Pharm. 2011;421(2):269-274.* https://doi.org/10.1016/j.ijpharm.2011.10.007

$j = 1, \ldots, k$ neighbors of x_{new}

- $w_j = exp \frac{-d_j}{h \times mad\{d_1,\ldots,d_k\}}$
- $w_j = w_j \,/\, \text{maximum}\{w_1, \ldots, w_k\}$



```
## function lwplsr

nlvdis = 20 ; metric = :mah
h = 1 ; k = 500 ; nlv = 15

mod = model(lwplsr; nlvdis, metric, h, k, nlv)
fit!(mod, X, y)

res = predict(mod, Xnew)
```

arguments

Keyword arguments:

- `nlvdis` : Number of latent variables (LVs) to consider in the global PLS used for the dimension reduction before computing the dissimilarities. If `nlvdis = 0`, there is no dimension reduction.
- `metric` : Type of dissimilarity used to select the neighbors and to compute the weights. Possible values are: `:eucl` (Euclidean distance), `:mah` (Mahalanobis distance).
- `h` : A scalar defining the shape of the weight function computed by function `wdist`. Lower is h, sharper is the function. See function `wdist` for details (keyword arguments `criw` and `squared` of `wdist` can also be specified here).
- `k` : The number of nearest neighbors to select for each observation to predict.
- `tolw` : For stabilization when very close neighbors.
- `nlv` : Nb. latent variables (LVs) for the local (i.e. inside each neighborhood) models.
- `scal` : Boolean. If `true`, each column of `X` and `Y` is scaled by its uncorrected standard deviation for the global dimension reduction and the local models.