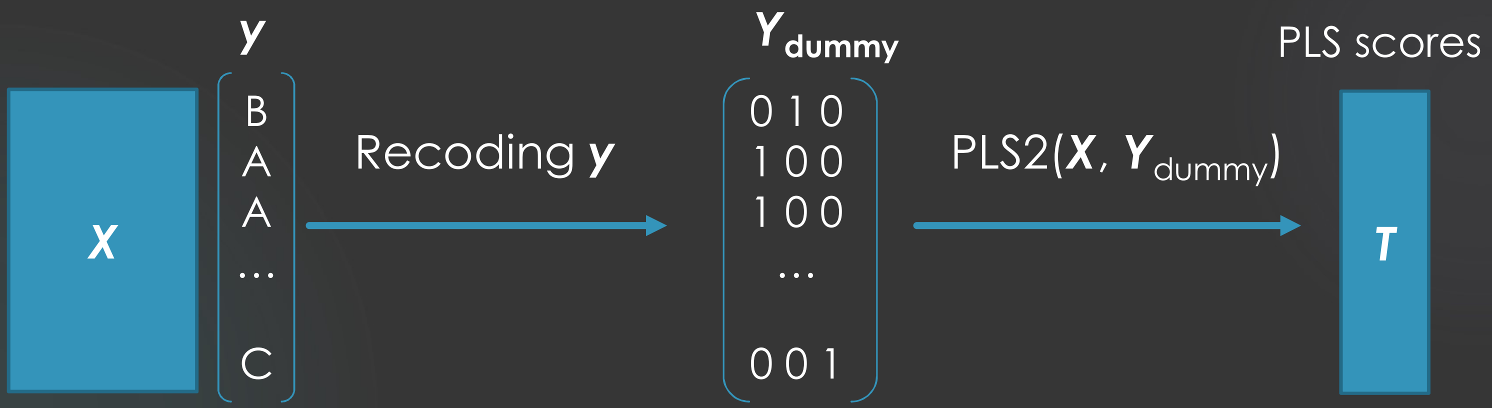
A non parametric PLSDA

matthieu.lesnoff@cirad.fr

Seminar ChemHouse Montpellier 19 June 2023

PLSDA

Step 1)



y
X
B
A
A
…
C
Recoding y
Y dummy
0 1 0
1 0 0
1 0 0
…
0 0 1
PLS2(X, Y dummy)
PLS scores
T

Step 2)

A) Regression $\boldsymbol{Y}_{\mathbf{dummy}}$ on $\boldsymbol{T}$ $\Rightarrow$ PLSR-DA = usual PLSDA

$$\hat{Y}_{dummy}$$

$$\begin{pmatrix} -.2 & 2.7 & -1.5 \\ .3 & .4 & .3 \\ \boxed{.9 \quad -.1 \quad .2} & & \\ & \cdots & \\ -.7 & -.1 & 1.8 \end{pmatrix} \longrightarrow \text{class A}$$

B) Probabilistic DA on ***T***

- Parametric — Assumption on the probability density of ***T***
  - e.g. Gaussian density estimation
    - LDA $\Rightarrow$ PLS-LDA
    - QDA $\Rightarrow$ PLS-QDA

- Non parametric — No assumption on the probability density of ***T***
  - e.g. **Kernel density estimation** (KDE) $\Rightarrow$ PLS-KDE-DA

$$\hat{P}\left(y_i = Class_j\right)$$

$$\begin{pmatrix} .1 & .8 & .1 \\ .4 & .5 & .1 \\ \boxed{.8 \quad .0 \quad .2} & & \\ & \cdots & \\ .2 & .1 & .7 \end{pmatrix} \longrightarrow \text{class A}$$

Illustration on iris data

X

Row	sepal_length Float64	sepal_width Float64	petal_length Float64	petal_width Float64
1	5.1	3.5	1.4	0.2
2	4.9	3.0	1.4	0.2
3	4.7	3.2	1.3	0.2

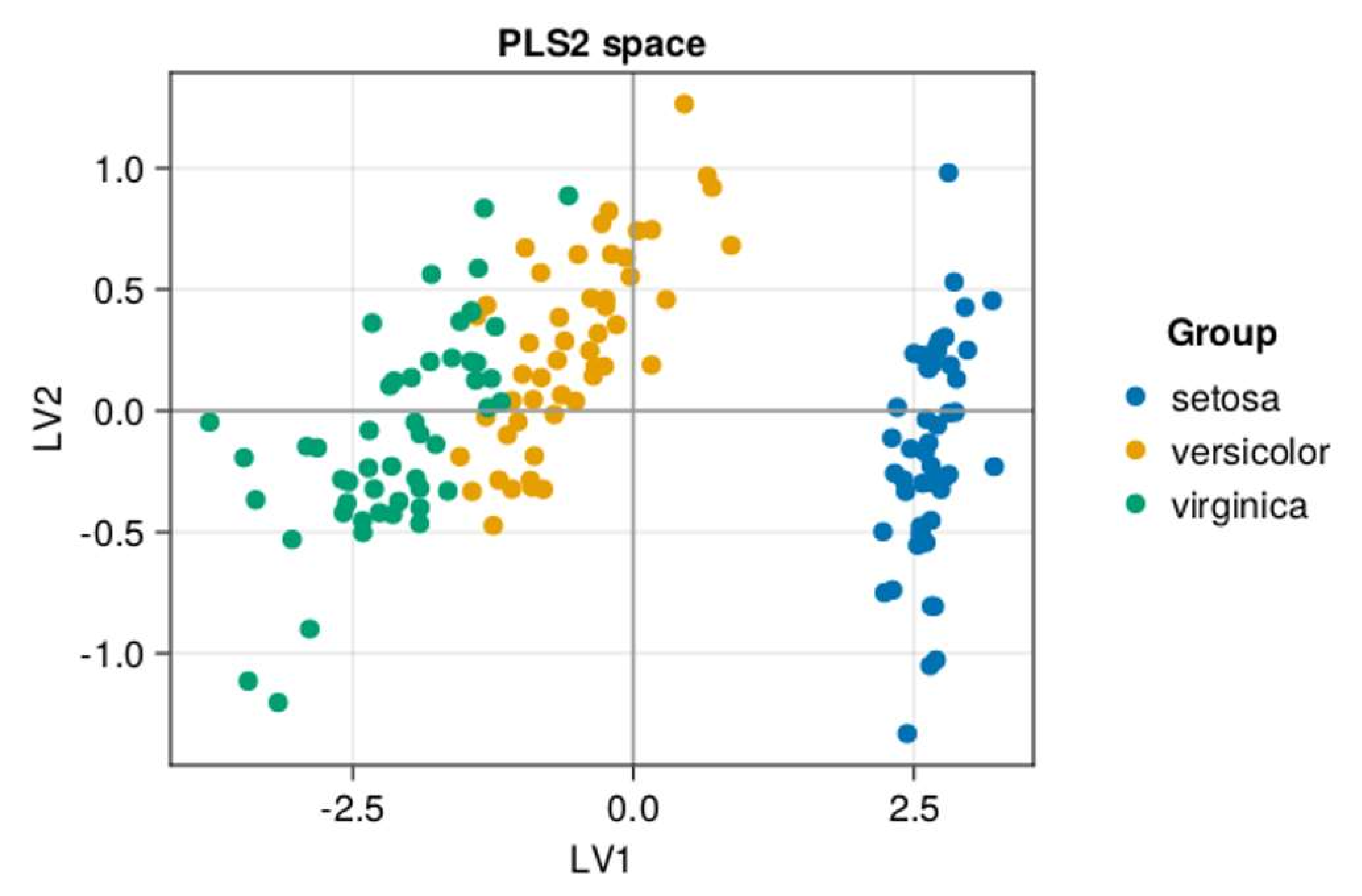
... (150, 4)

y: 3 classes

```
"setosa"     => 50
"versicolor" => 50
"virginica"  => 50
```

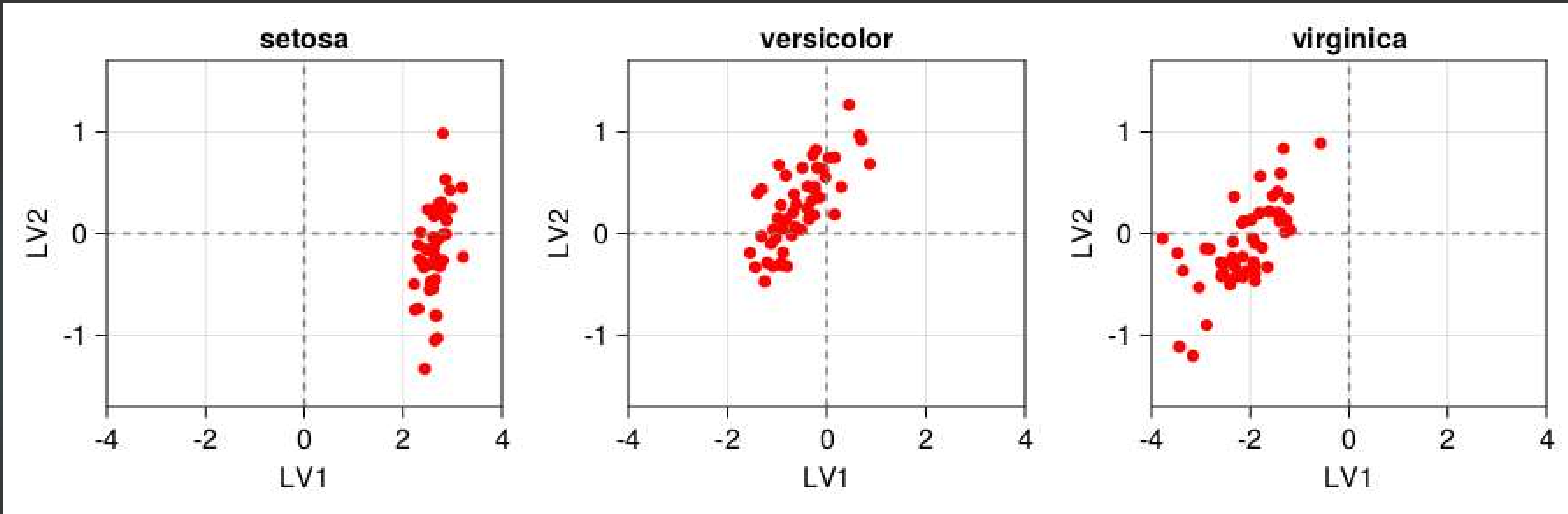
PLS2($\boldsymbol{X}$, $\boldsymbol{Y}_{\text{dummy}}$) nb. LVs = 2

$\Rightarrow$ $\mathbf{T}$ ($n \times 2$)

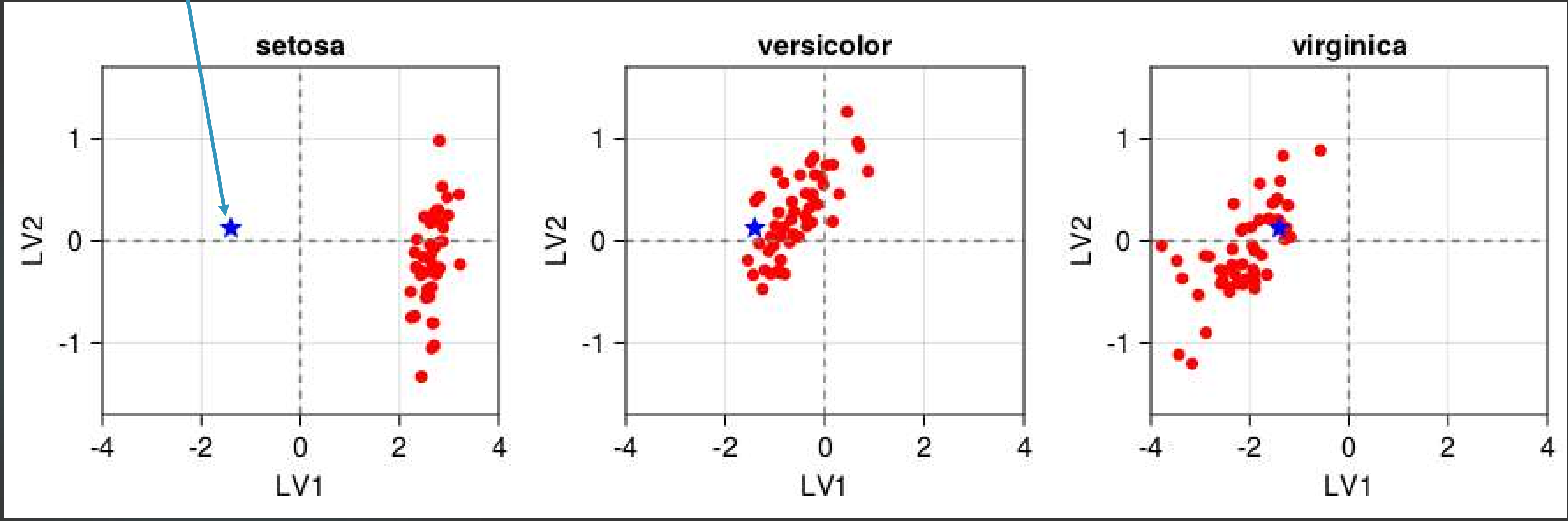

PLS2 space
LV2
1.0
0.5
0.0
-0.5
-1.0
-2.5
0.0
2.5
LV1
Group
setosa
versicolor
virginica

Probabilistic DA

⇒ Estimate the multivariate probability density of $\boldsymbol{T}$ in each class



New observation to predict
= which class?



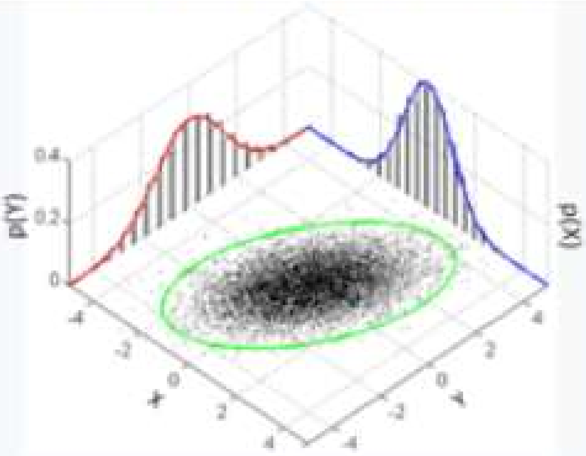
1) Parametric   Gaussian probability density

PDF	$(2\pi)^{-k/2}\det(\boldsymbol\Sigma)^{-1/2}\,\exp\left(-\frac{1}{2}(\mathbf{x}-\boldsymbol\mu)^{\mathsf T}\boldsymbol\Sigma^{-1}(\mathbf{x}-\boldsymbol\mu)\right),$ exists only when $\boldsymbol\Sigma$ is positive-definite

Σ = covariance matrix

Multivariate normal

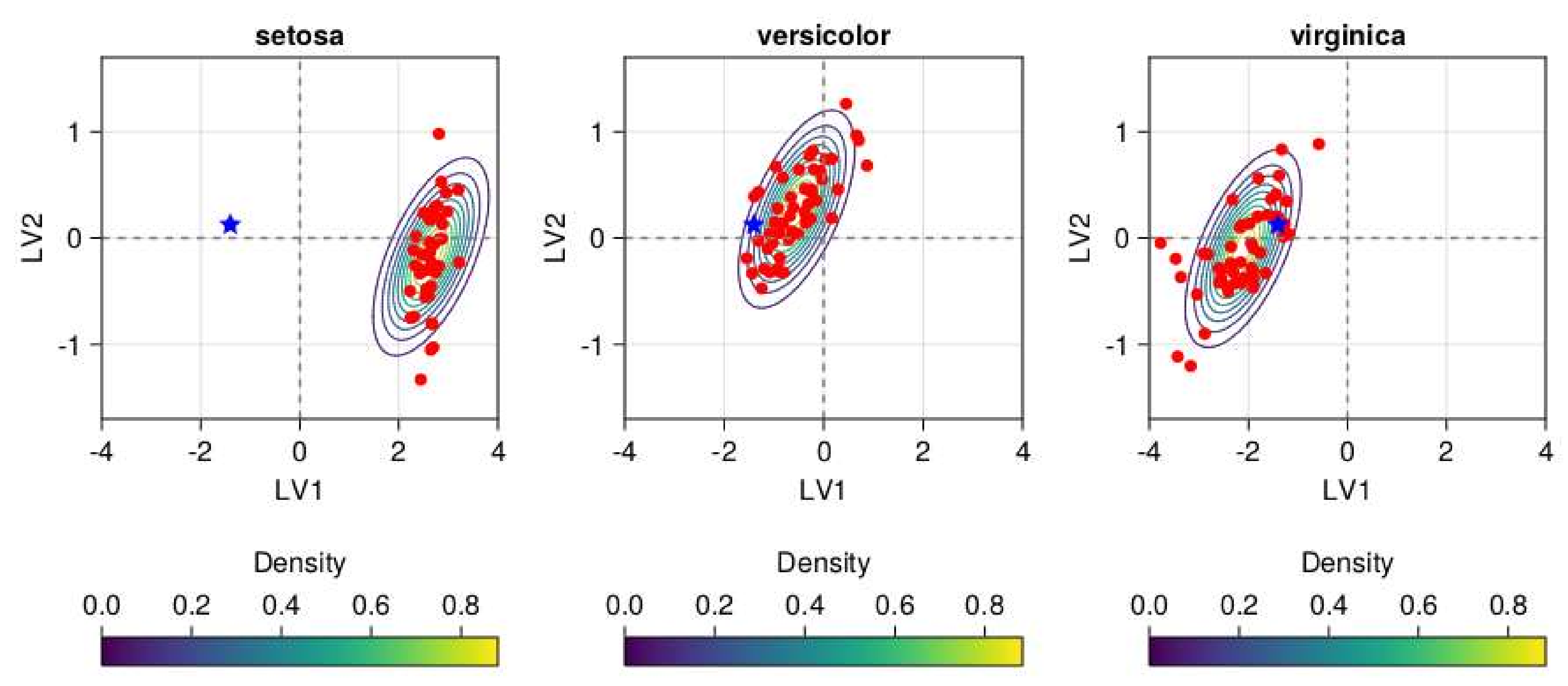
Probability density function



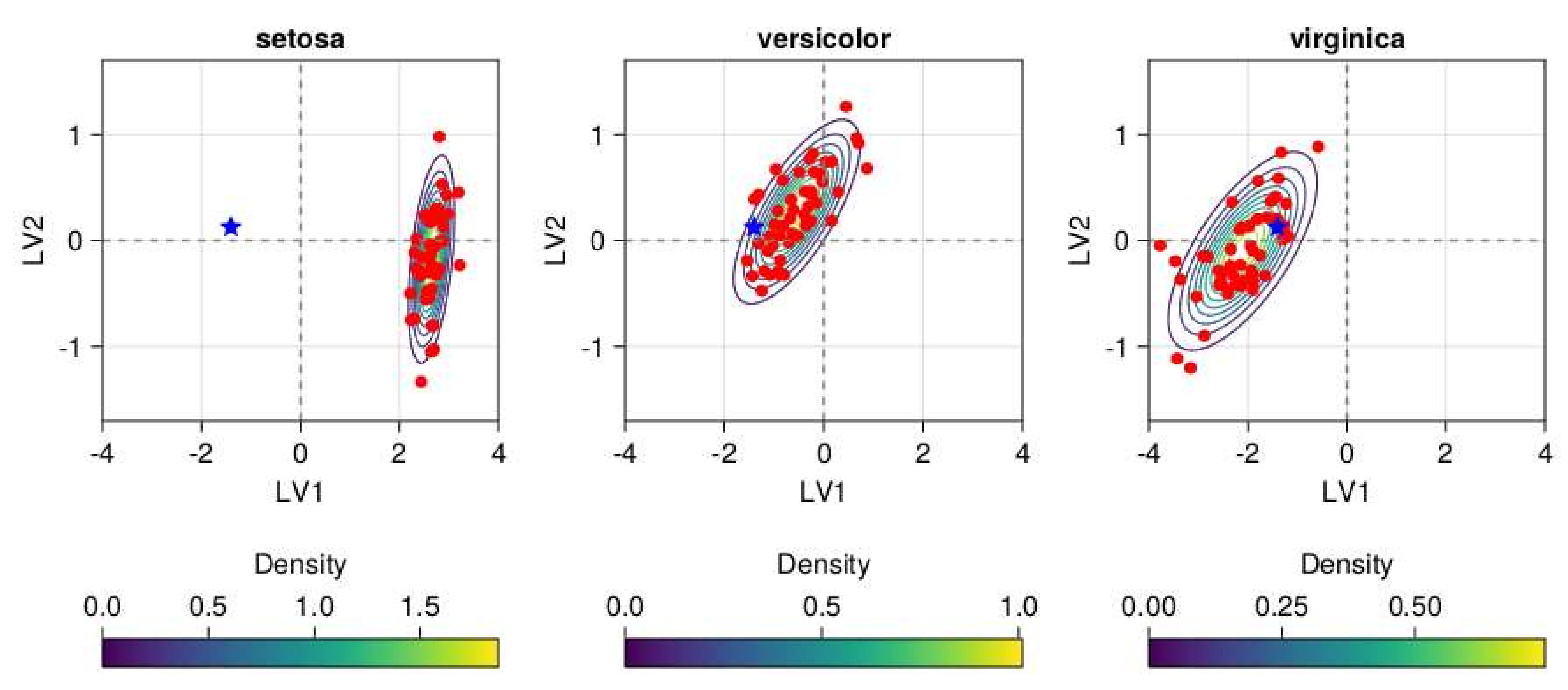
LDA Same Σ for all classes
QDA One Σ per class

https://en.wikipedia.org/wiki/Multivariate_normal_distribution

LDA Same covariance matrix Σ for all classes



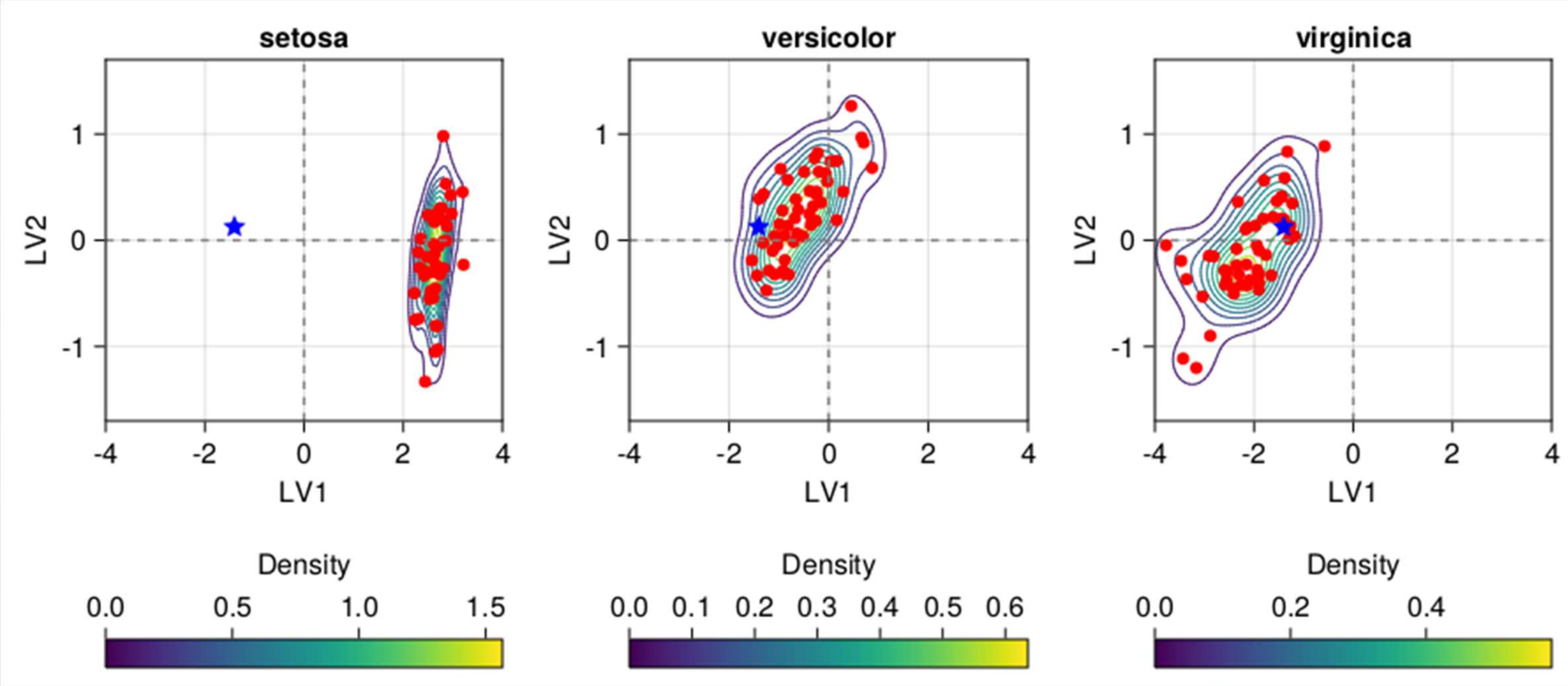
QDA One covariance matrix Σ per class



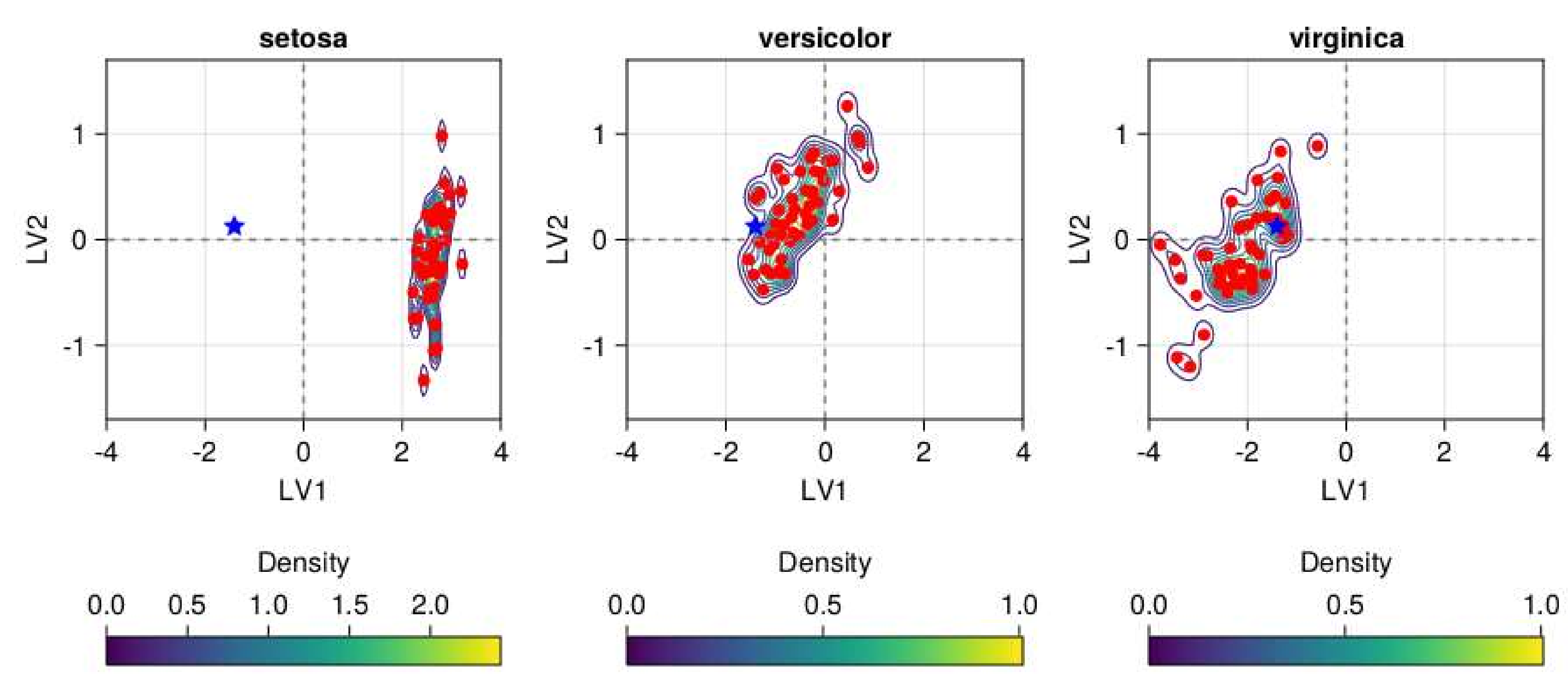
2) Non parametric   Probability density estimated by KDE

Ex: **Multiplicative Gaussian KDE**

Smoothing level a = 1



Smoothing level a = .5



Univariate KDE

$$\hat{f}_K(x) = \frac{1}{nh}\sum_{i=1}^{n} K\left(\frac{x - x_i}{h}\right) = \frac{1}{n}\sum_{i-1}^{n} K_h(x - x_i)\,,$$

Density estimate at obs. **x**

Bandwidth
(= smoothing)

To tune

Kernel function ~ similarity to **x**
(integral sums to 1)

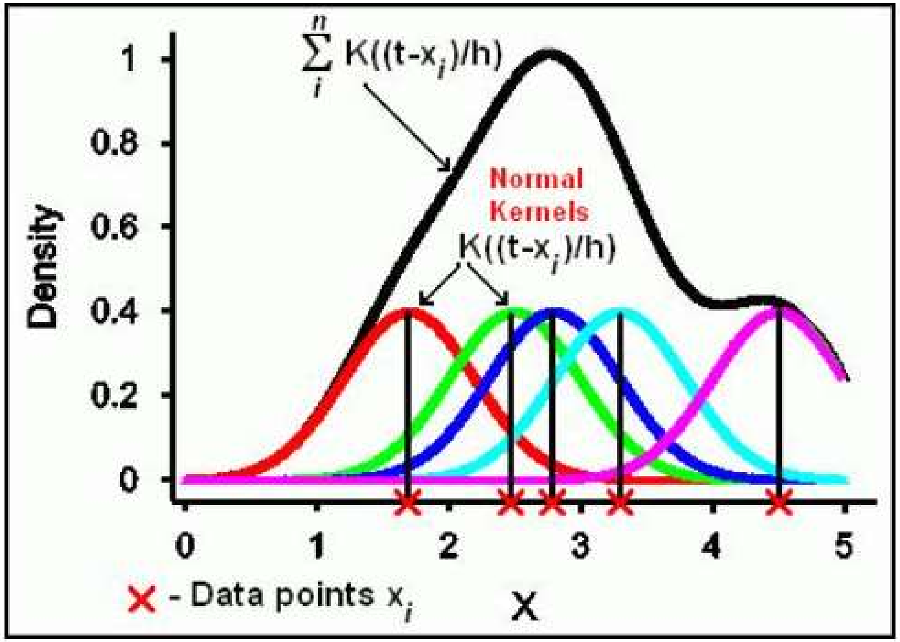
e.g. Gaussian pdf

Scott DW, Sain SR. - Multidimensional Density Estimation. In: Rao CR, Wegman EJ, Solka JL, eds. Handbook of Statistics. Vol 24. Data Mining and Data Visualization. Elsevier; 2005:229-261. doi:10.1016/S0169-7161(04)24009-3.

```
fh(t) = the sum of (K(t-xi)/h)/(nh) from i = 1...n
```

where n denotes the sample size. The choice of kernel function K is not very critical for the resulting estimate fh(t) and so a Gaussian kernel is used.

The following graph showing the sum of the normal kernels at 5 data points illustrates the ideas behind the kernel density estimation.



Univariate KDE

Source: https://genstat.kb.vsni.co.uk/knowledge-base/kernel-density-estimation

Gaussian KDE

kernel K = Gaussian pdf

Univariate

$$\hat{f}(x) = \frac{1}{n}\sum_{i=1}^{n} \frac{1}{h\sqrt{2\pi}} exp\left\{-\frac{1}{2}\left(\frac{x-x_i}{h}\right)^2\right\}$$

$$= \frac{1}{nh}\sum_{i=1}^{n} K(u_i)$$

with $\quad u_i = \dfrac{x-x_i}{h}$

and $\quad K(u_i) = \dfrac{1}{\sqrt{2\pi}} exp\left\{-\dfrac{1}{2} u_i^2\right\}$ pdf Normal(0, 1)

Multivariate

The extension of the kernel estimator to vector-valued data, $\mathbf{x} \in \Re^d$, is straightforward for a normal kernel, $K \sim N(0, \Sigma)$:

$$\hat{f}(\mathbf{x}) = \frac{1}{n(2\pi)^{d/2}|\Sigma|^{1/2}} \sum_{i=1}^{n} \exp[-\frac{1}{2}(\mathbf{x}-\mathbf{x}_i)'\Sigma^{-1}(\mathbf{x}-\mathbf{x}_i)] \,. \qquad (16)$$

Bandwidth matrix

$p \times p$ Positive-definite, symmetric

To tune

Scott & Sain 2005

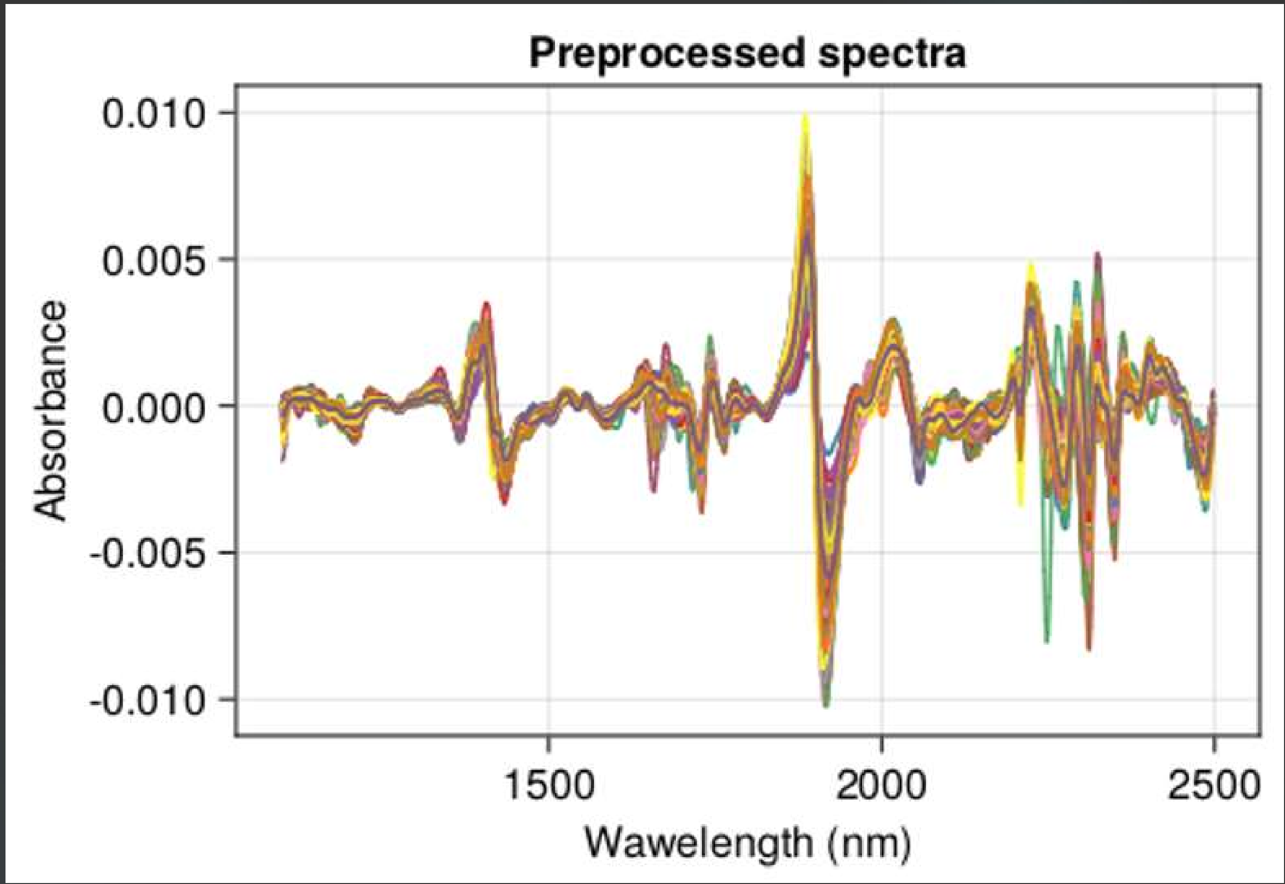
$$\hat{f}(\mathbf{x}) = \frac{1}{n(2\pi)^{d/2}|\Sigma|^{1/2}} \sum_{i=1}^{n} \exp[-\frac{1}{2}(\mathbf{x}-\mathbf{x}_i)'\Sigma^{-1}(\mathbf{x}-\mathbf{x}_i)] \, .$$

Assuming Σ to be diagonal
simplifies a lot the computations and tuning

$\Rightarrow$ Multiplicative Gaussian KDE

(product of univariate KDEs)

PLS-KDE-DA on mixed forages data

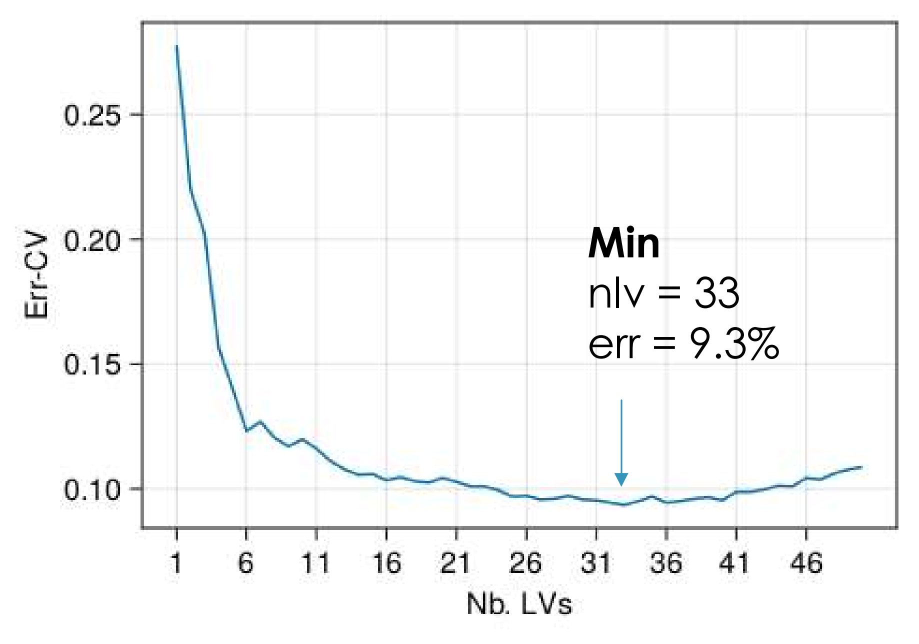


ntot = 485
ntrain = 323 (CV)
ntest = 162

y: 3 classes

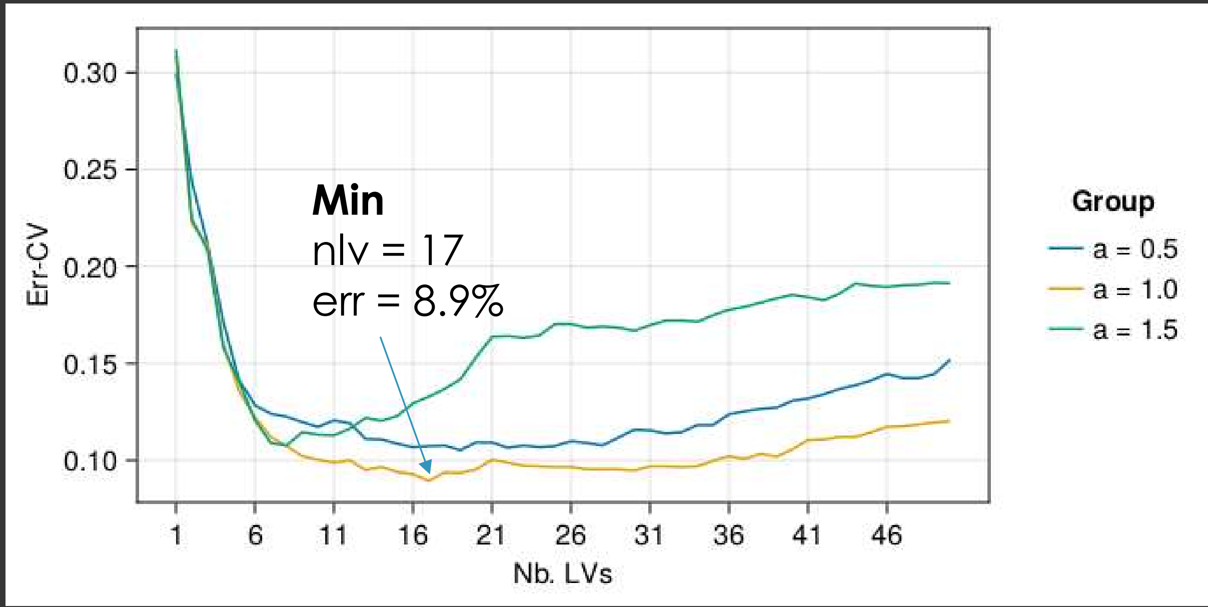
```
"Legume forages"
"Forage trees"
"Cereal and grass forages"
```

PLS-LDA



Err-Test = 8%

PLS-KDE-DA



Err-Test = 7%

Available in package Jchemo

- Functions dmkern, plskdeda

https://github.com/mlesnoff/Jchemo.jl