Cost-sensitive classification and regression
(+ anything I can talk about in about 45')

Emilio Carrizosa
Instituto de Matemáticas de la Universidad de Sevilla
Paris, March 27 2019

www.grupo.us.es/gioptim

www.riseneeds.eu

@ needs_project

- Sandra Benítez-Peña
- Rafael Blanquero
- M. Asunción Jiménez-Cordero
- Alba V Olivares-Nadal
- Pepa Ramírez-Cobo
- Remedios Sillero-Denamiel

Supervised classification

Supervised classification. The framework

- Given: set I of individuals, each $i \in I$ with associated

Supervised classification. The framework

- Given: set I of individuals, each $i \in I$ with associated
  - A vector $\mathbf{x}_i \in \mathcal{X}$, assumed here $\mathcal{X} \subset \mathbb{R}^m$
  - A label y_i, assumed here to be in $\{-1, +1\}$
- Seen as a sample from $(\mathbf{X}, Y)$, with unknown distribution

Supervised classification. The framework

- Given: set I of individuals, each $i \in I$ with associated
  - A vector $\mathbf{x}_i \in \mathcal{X}$, assumed here $\mathcal{X} \subset \mathbb{R}^m$
  - A label y_i, assumed here to be in $\{-1, +1\}$
- Seen as a sample from $(\mathbf{X}, Y)$, with unknown distribution
- Goal: to infer from I a classifier $\varphi : \mathcal{X} \longrightarrow \{-1, 1\}$ so that we can classify any object just by knowing $\mathbf{x}$

Supervised classification. The framework

- Given: set I of individuals, each $i \in I$ with associated
  - A vector $\mathbf{x}_i \in \mathcal{X}$, assumed here $\mathcal{X} \subset \mathbb{R}^m$
  - A label y_i, assumed here to be in $\{-1, +1\}$
- Seen as a sample from $(\mathbf{X}, Y)$, with unknown distribution
- Goal: to infer from I a classifier $\varphi : \mathcal{X} \longrightarrow \{-1, 1\}$ so that we can classify any object just by knowing $\mathbf{x}$
- Linear classifier $\varphi : \mathbf{x} = (x_1, \ldots, x_m) \longmapsto \{-1, 1\}$:
  - score function:

$$\mathbf{x} = (x_1, \ldots, x_m) \longmapsto \omega_1 x_1 + \ldots + \omega_n x_m + \beta$$

  - $\varphi(x) = \begin{cases} 1, & \text{if } \omega_1 x_1 + \ldots + \omega_n x_m + \beta > 0 \\ -1, & \text{else} \end{cases}$

Supervised classification. The framework

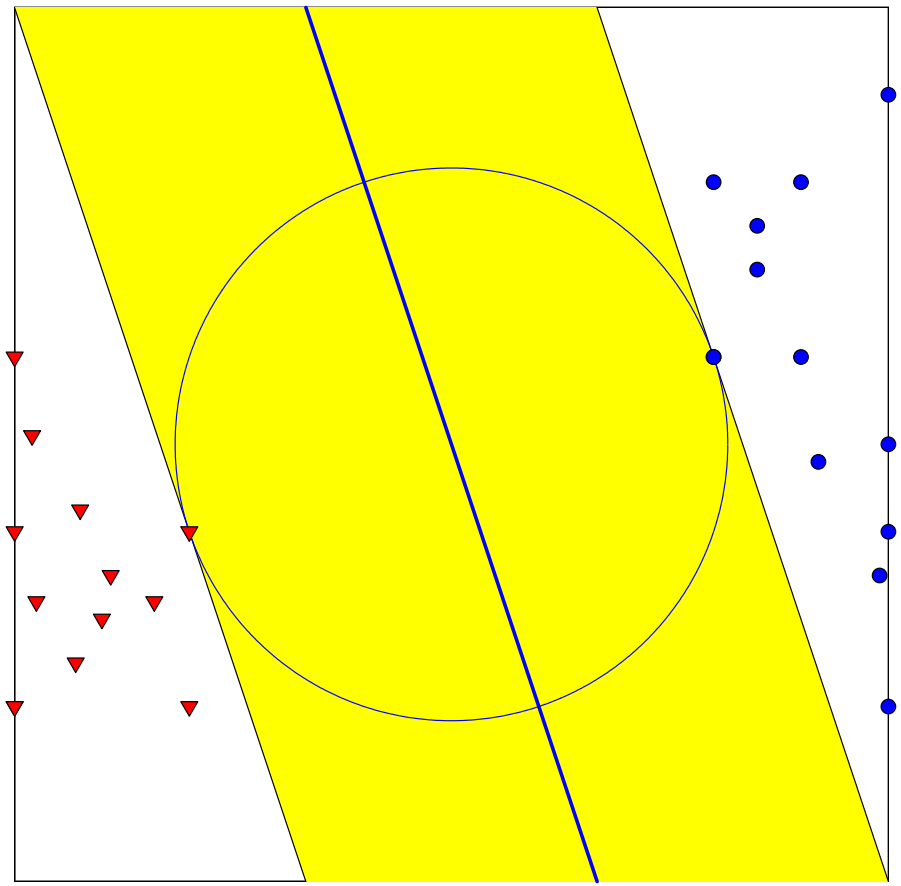
- Given: set I of individuals, each $i \in I$ with associated
  - A vector $\mathbf{x}_i \in \mathcal{X}$, assumed here $\mathcal{X} \subset \mathbb{R}^m$
  - A label y_i, assumed here to be in $\{-1, +1\}$
- Seen as a sample from $(\mathbf{X}, Y)$, with unknown distribution
- Goal: to infer from I a classifier $\varphi : \mathcal{X} \longrightarrow \{-1, 1\}$ so that we can classify any object just by knowing $\mathbf{x}$
- Linear classifier $\varphi : \mathbf{x} = (x_1, \ldots, x_m) \longmapsto \{-1, 1\}$:
  - score function:

$$\mathbf{x} = (x_1, \ldots, x_m) \longmapsto \omega_1 x_1 + \ldots + \omega_n x_m + \beta$$

  - $\varphi(x) = \begin{cases} 1, & \text{if } \omega_1 x_1 + \ldots + \omega_n x_m + \beta > 0 \\ -1, & \text{else} \end{cases}$
- Problem: how to infer from I the coefficients $\omega = (\omega_1, \ldots, \omega_n), \beta$?

SVM

- Roughly speaking, SVM finds the hyperplane $\omega_1 x_1 + \ldots + \omega_m x_m + \beta = 0$ separating most the sets $\{\mathbf{x}_i : i \in I, y_i = 1\}$ and $\{\mathbf{x}_i : i \in I, y_i = -1\}$



SVM

Convex quadratic optimization problem with linear constraints:


C., and Romero Morales, "Supervised classification and mathematical optimization", *Computers & Operations Research*, 2013.


Duarte Silva, "Optimization approaches to supervised classification", *EJOR*, 2017.

SVM

Convex quadratic optimization problem with linear constraints:

$$\begin{array}{lll} \min_{\omega,\beta,\xi} & \|\omega\|^2 + C\sum_{i\in I}\xi_i & \\ \text{s.t.} & y_i\left(\omega^\top \mathbf{x}_i + \beta\right) \geq 1 - \xi_i & i \in I \\ & \xi_i \geq 0 & i \in I \end{array}$$


C., and Romero Morales, "Supervised classification and mathematical optimization", *Computers & Operations Research*, 2013.


Duarte Silva, "Optimization approaches to supervised classification", *EJOR*, 2017.

SVM

Convex quadratic optimization problem with linear constraints:

$$\begin{array}{lll} \min_{\omega,\beta,\xi} & \|\omega\|^2 + C\sum_{i\in I}\xi_i & \\ \text{s.t.} & y_i\left(\omega^\top \mathbf{x}_i + \beta\right) \geq 1 - \xi_i & i \in I \\ & \xi_i \geq 0 & i \in I \end{array}$$

$$\begin{array}{lll} \max_{\lambda} & \sum_{i\in I}\lambda_i - \frac{1}{2}\sum_{ij}\lambda_i y_i \lambda_j y_j \mathbf{x}_i^\top \mathbf{x}_j & \\ \text{s.t.} & \sum_{i\in I}\lambda_i y_i = 0 & \\ & 0 \leq \lambda_i \leq \frac{C}{2} & i \in I \end{array}$$


C., and Romero Morales, "Supervised classification and mathematical optimization", *Computers & Operations Research*, 2013.


Duarte Silva, "Optimization approaches to supervised classification", *EJOR*, 2017.

SVM

Convex quadratic optimization problem with linear constraints:

$$\begin{array}{lll} \min_{\omega,\beta,\xi} & \|\omega\|^2 + C\sum_{i\in I}\xi_i & \\ \text{s.t.} & y_i\left(\omega^\top \mathbf{x}_i + \beta\right) \geq 1 - \xi_i & i \in I \\ & \xi_i \geq 0 & i \in I \end{array}$$

$$\begin{array}{lll} \max_{\lambda} & \sum_{i\in I}\lambda_i - \frac{1}{2}\sum_{ij}\lambda_i y_i \lambda_j y_j K(\mathbf{x}_i, \mathbf{x}_j) & \\ \text{s.t.} & \sum_{i\in I}\lambda_i y_i = 0 & \\ & 0 \leq \lambda_i \leq \frac{C}{2} & i \in I \end{array}$$


C., and Romero Morales, "Supervised classification and mathematical optimization", *Computers & Operations Research*, 2013.


Duarte Silva, "Optimization approaches to supervised classification", *EJOR*, 2017.

Kernels

$K(\mathbf{x}_i, \mathbf{x}_j) = \ldots$

- $\mathbf{x}_i^\top \mathbf{x}_j$ (linear kernel)
- $(1 + \mathbf{x}_i^\top \mathbf{x}_j)^d$ (polynomial kernel)
- $e^{-\gamma \|\mathbf{x}_i - \mathbf{x}_j\|^2}$ (gaussian kernel)
- $\sum_k \theta_k e^{-\gamma_k \|\mathbf{x}_i - \mathbf{x}_j\|^2}$
- . . . many more (not only for $\mathbf{x}$ in a dot product space)

Cristianini and Shawe-Taylor. *An introduction to support vector machines and other kernel-based learning methods*, 2000.

Hofmann, Schölkopf and Smola, "Kernel methods in Machine Learning", *Annals of Statistics*, 2008.

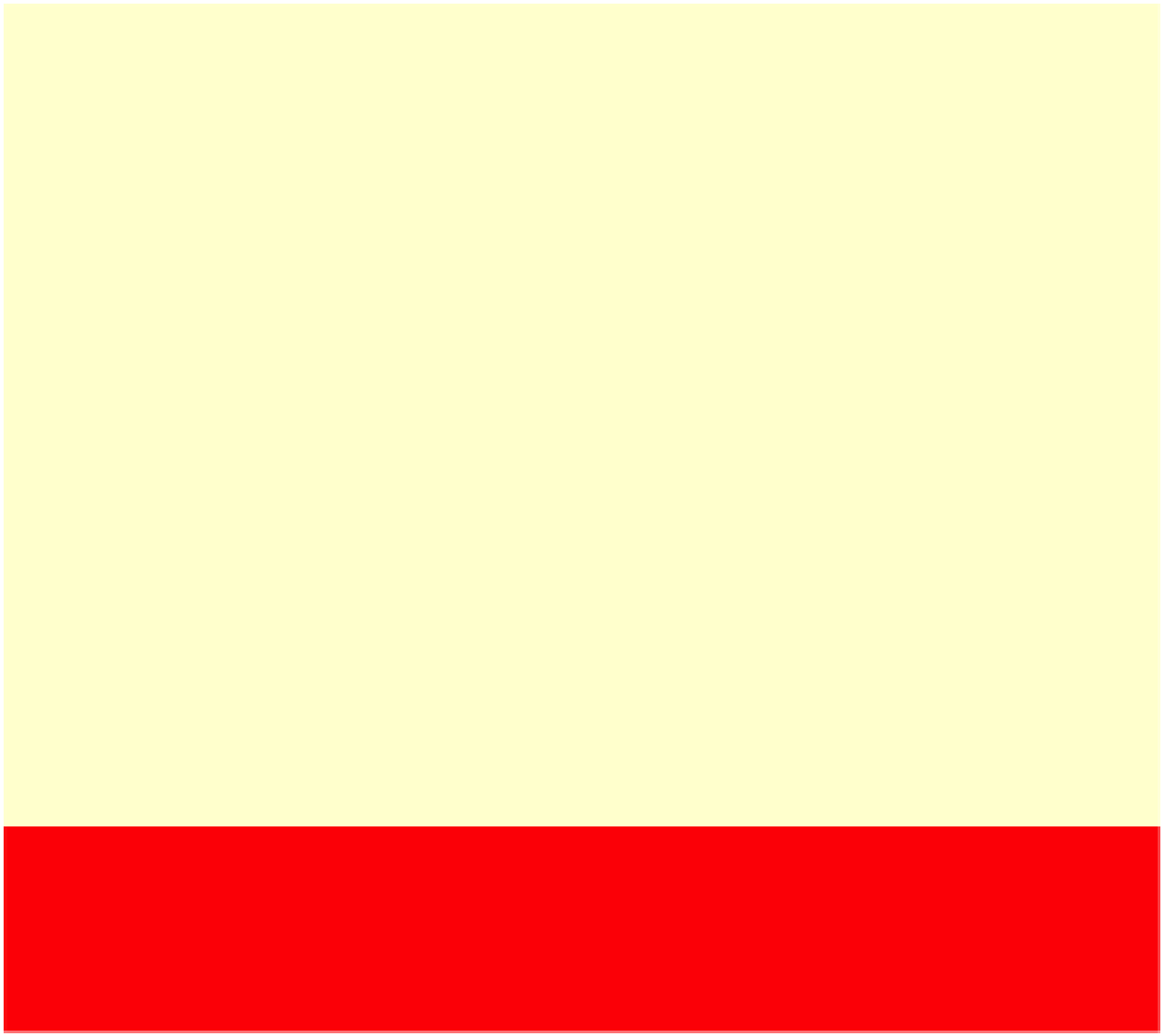
Parameters tuning

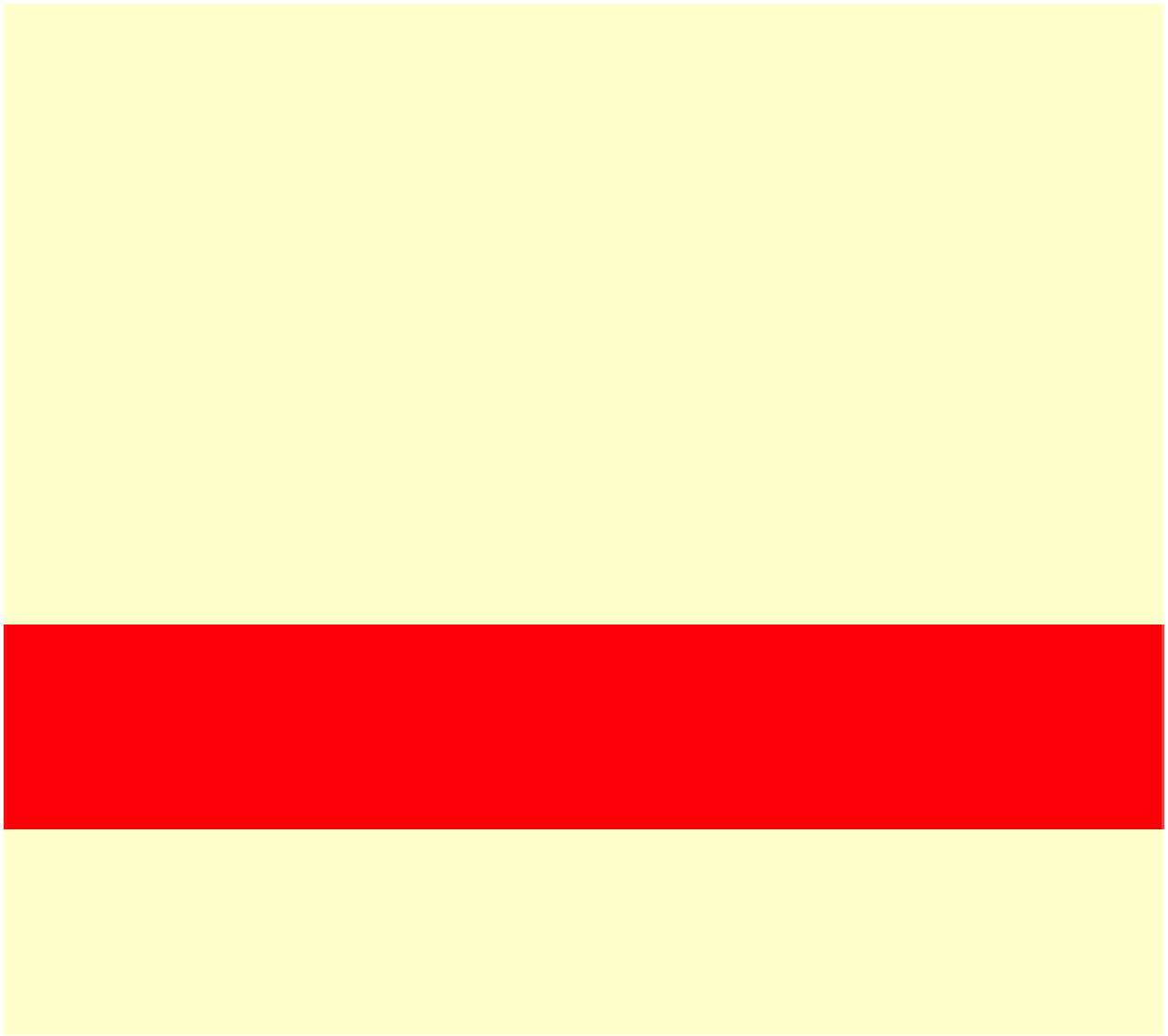
$$\begin{array}{lll} \max_\lambda & \sum_{i\in I}\lambda_i - \frac{1}{2}\sum_{ij}\lambda_i y_i \lambda_j y_j e^{-\gamma\|\mathbf{x}_i-\mathbf{x}_j\|^2} & \\ \text{s.t.} & \sum_{i\in I}\lambda_i y_i = 0 & (P_{I,C,\gamma}) \\ & 0 \le \lambda_i \le \frac{C}{2} \qquad\qquad i \in I & \end{array}$$

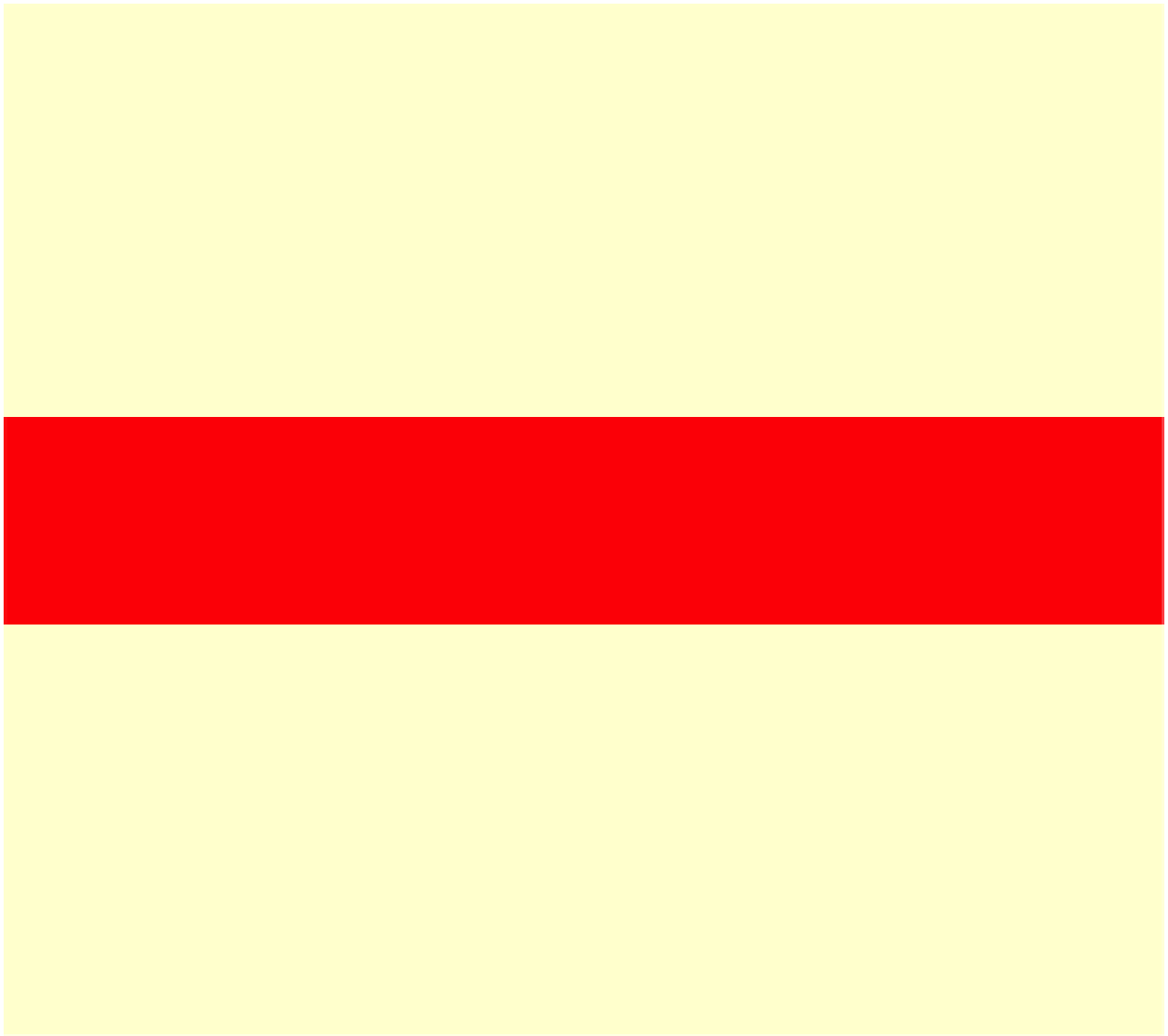
C, γ: k-fold crossvalidation

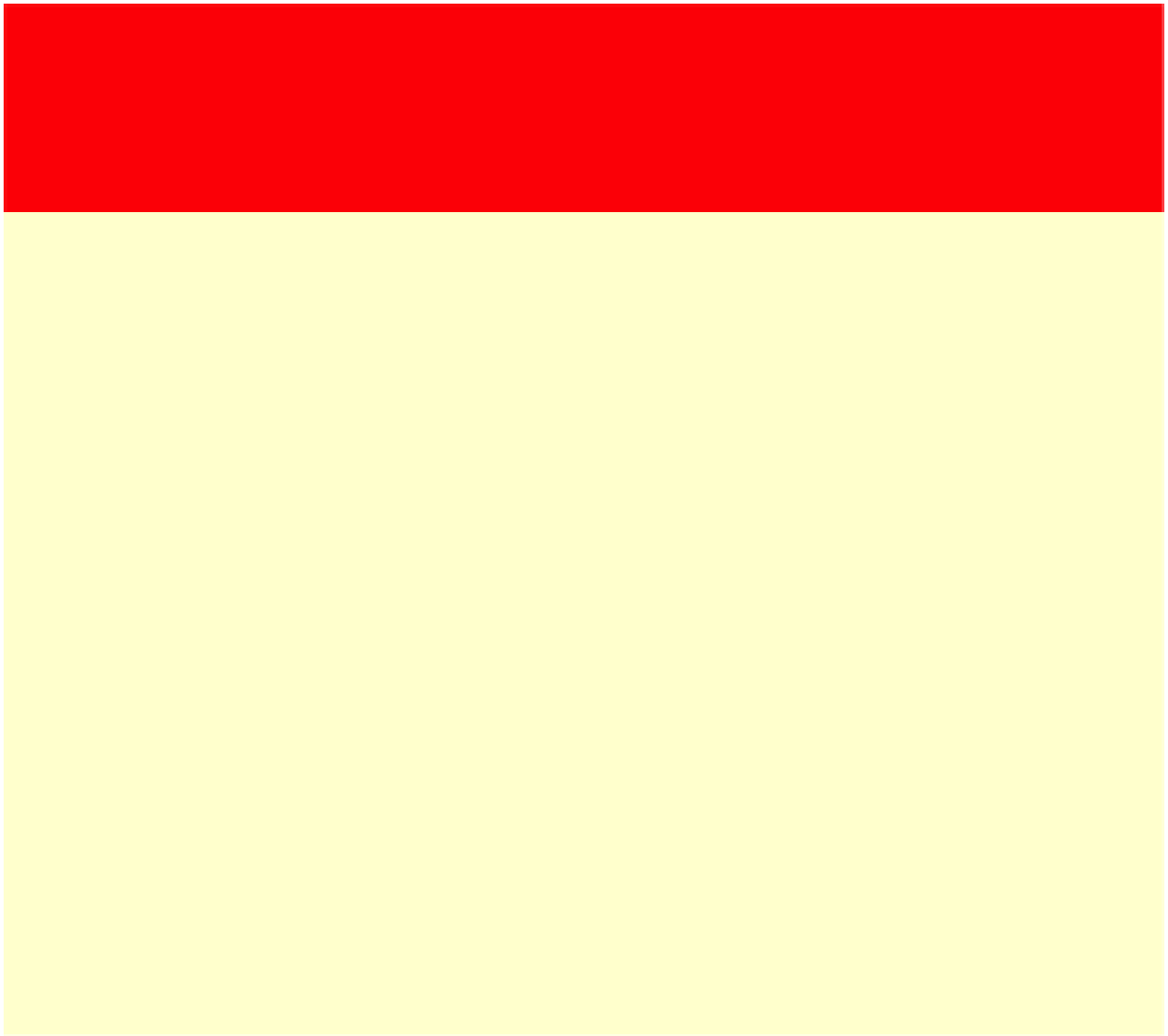
- I : split in k blocks of similar size, $I_1, \ldots, I_k$
- for each pair C, γ in a grid (e.g. $2^{-12} \ldots, 2^{12}$), estimate $acc(C, \gamma)$:
  - for each $i = 1, \ldots, k$
    - solve $(P_{I\setminus I_i,C,\gamma})$, yielding λ^i, β (via KKT)
    - calculate $acc(C, \gamma, I_i)$, fraction of correctly classified in I_i if classifier with λ^i, β were used
  - $acc(C, \gamma) = \frac{1}{k}\sum_{i=1}^{k} acc(C, \gamma, I_i)$

Kohavi, "A study of cross-validation and bootstrap for accuracy estimation and model selection", *IJCAI*, 1995.



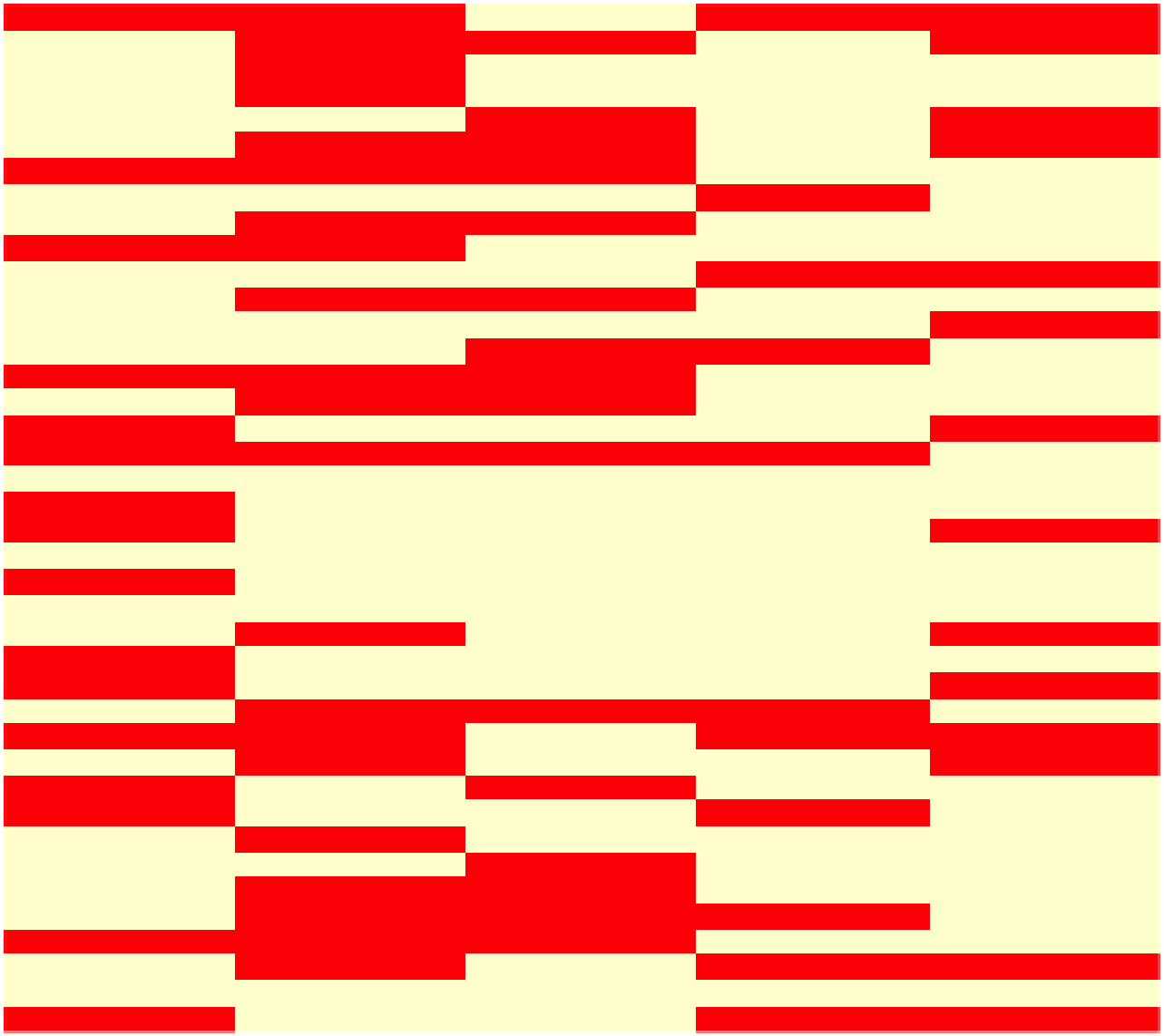


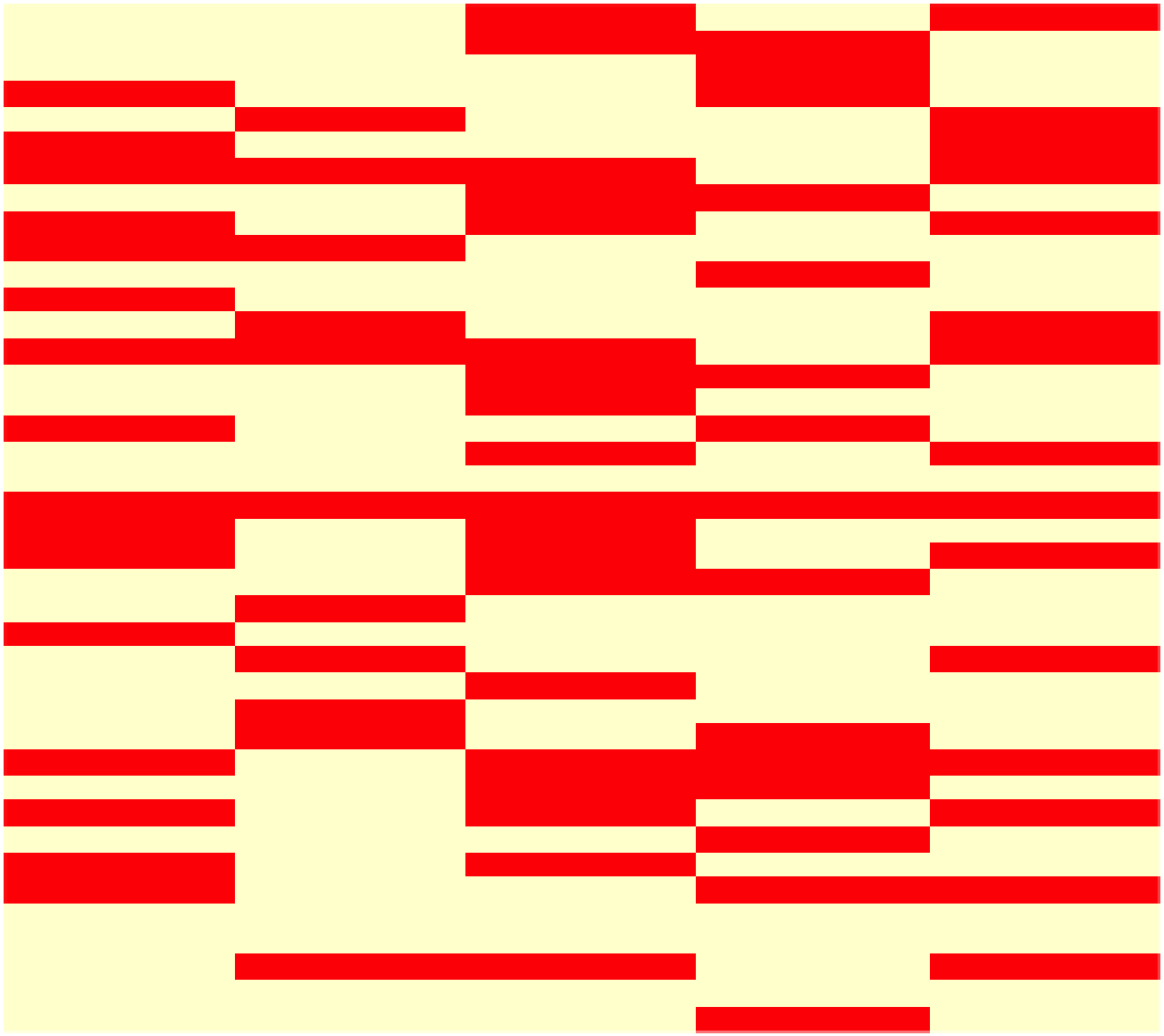


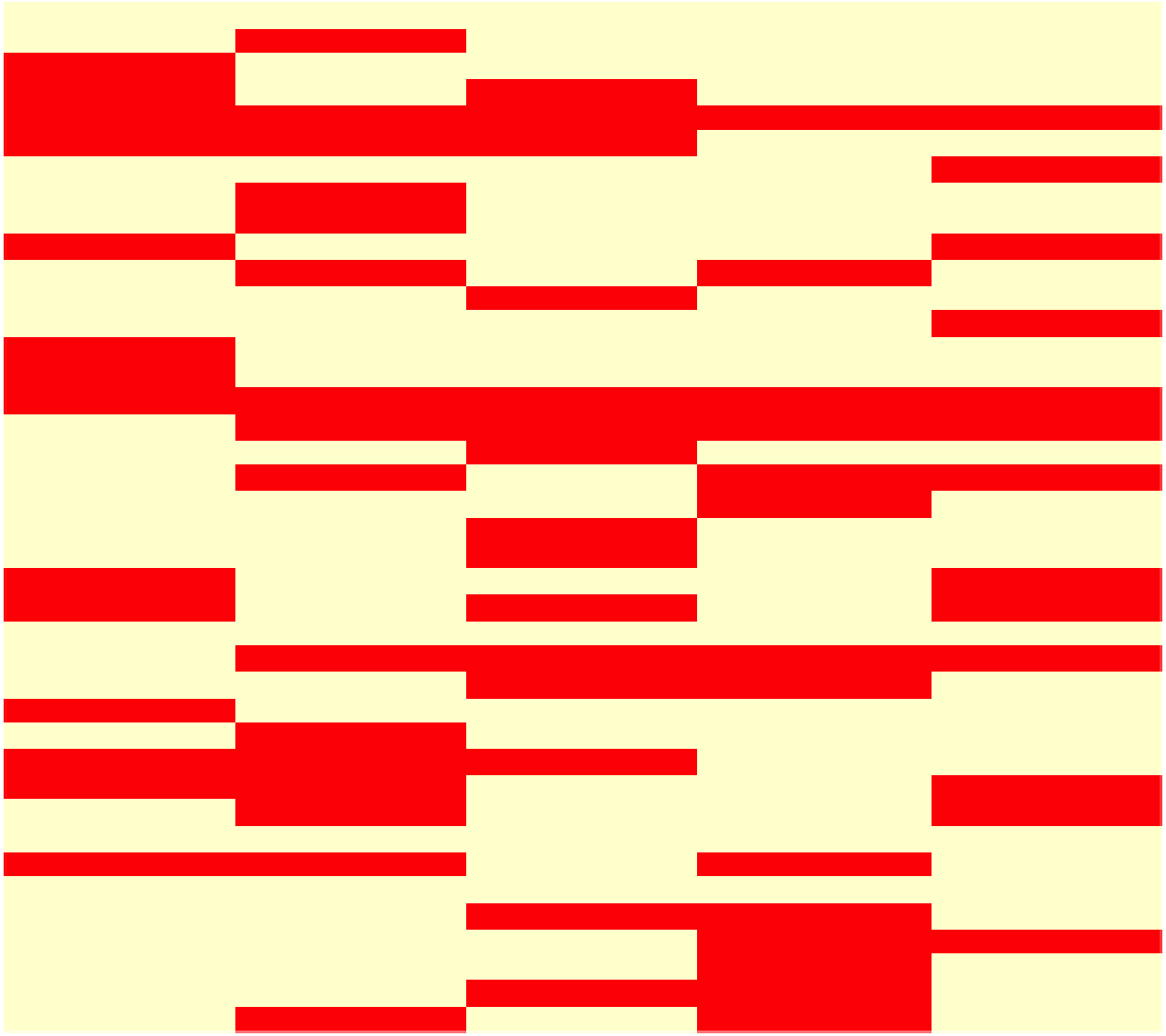


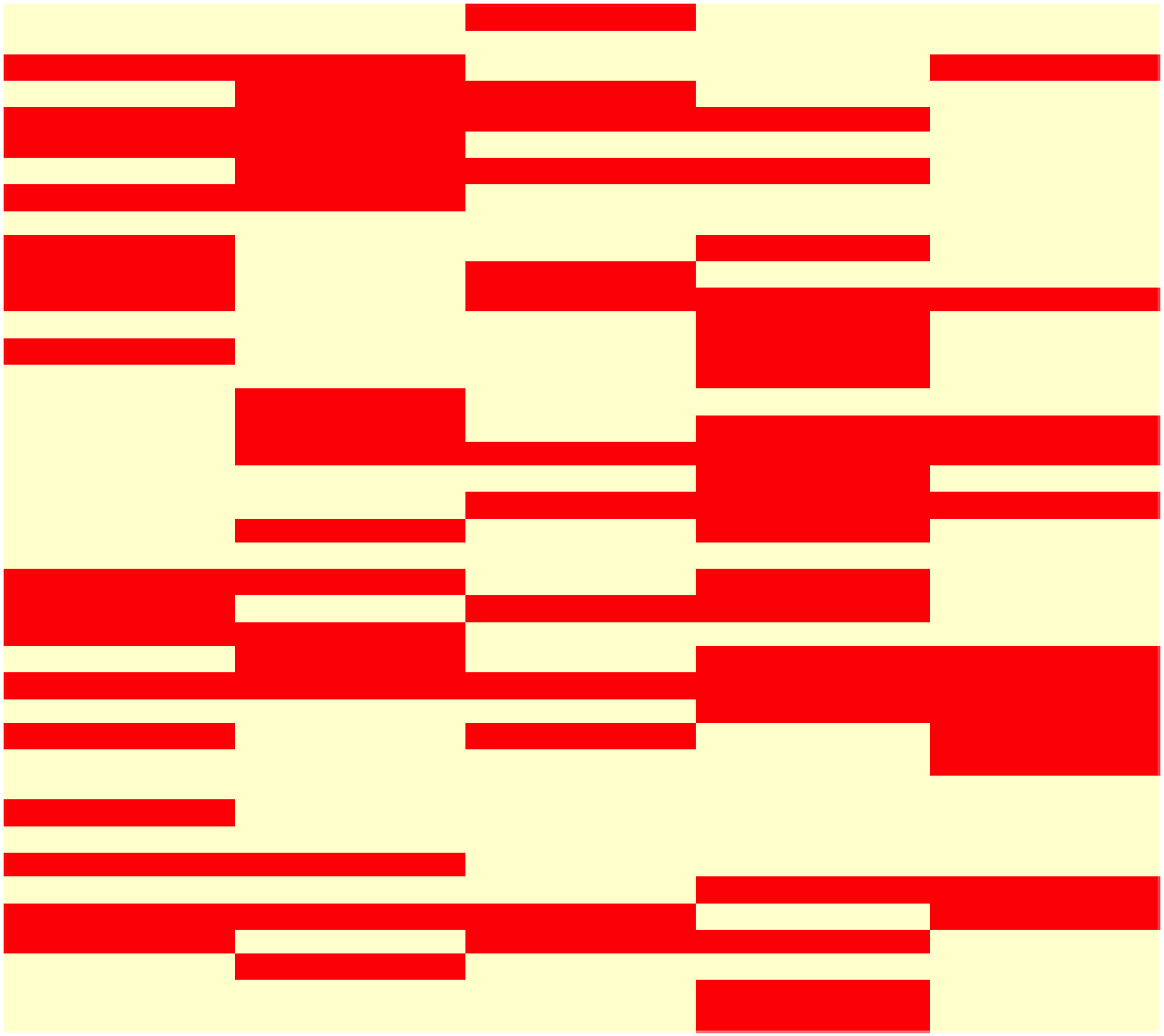
Bootstrap

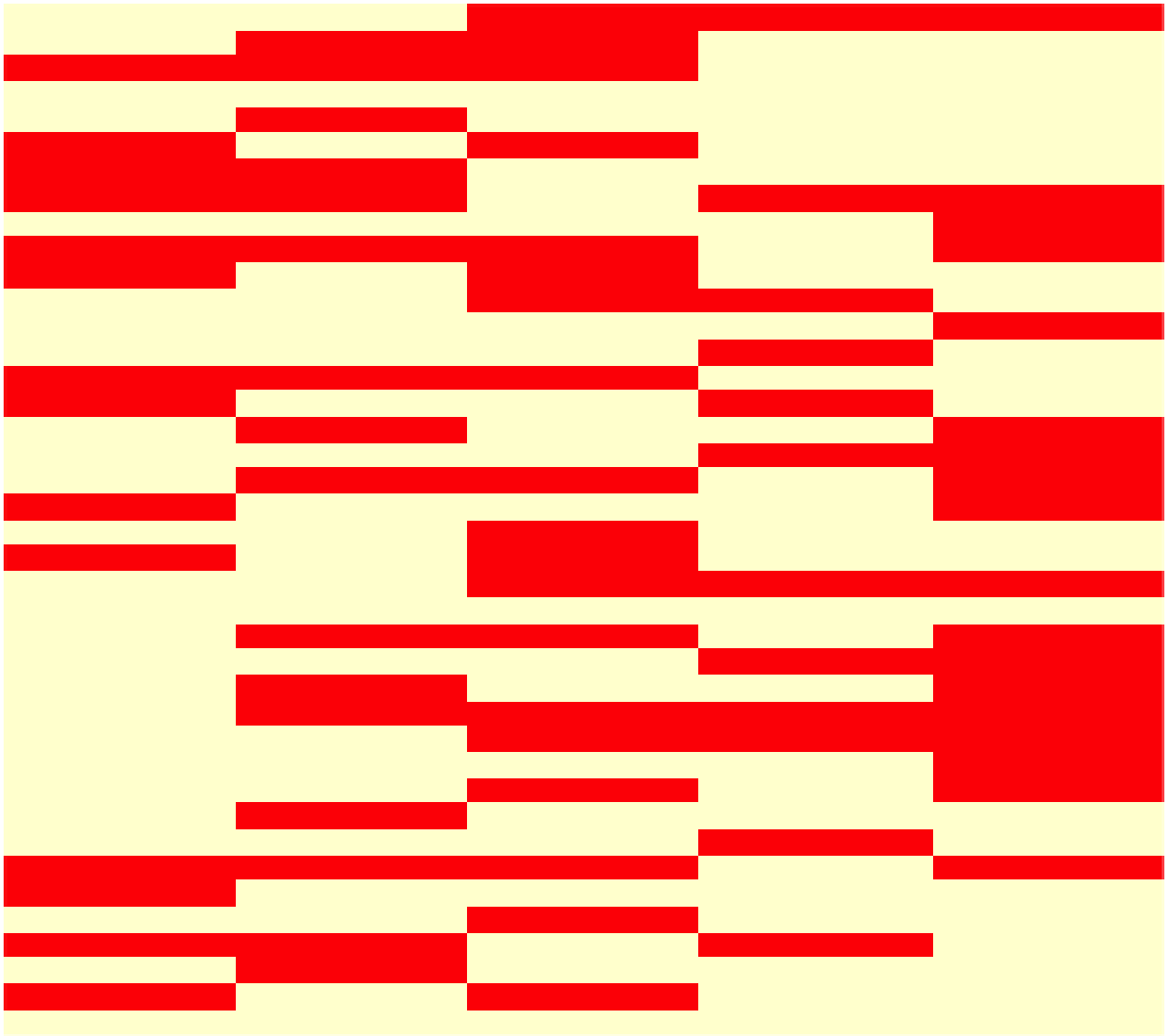
1. Take B (e.g. $B = 100$, $B = 1000$)
2. Foe each $b \in \{1, \ldots, B\}$:
   1. Generate a sample with replacement of same size than training sample
   2. Build the model on the bootstrap sample
   3. Estimate performance π_b on out-of-bag sample
3. Average π_b











The performance measure

$$
\begin{array}{lll}
\min_{\omega,\beta,\xi} & \|\omega\|^2 + C \sum_{i \in I} \xi_i & \\
\text{s.t.} & y_i \left(\omega^\top \mathbf{x}_i + \beta \right) \geq 1 - \xi_i & i \in I \\
& \xi_i \geq 0 & i \in I
\end{array}
$$

The performance measure

$$
\begin{array}{lllll}
\min_{\omega,\beta,\xi} & \|\omega\|^2 + C \sum_{i \in I} \xi_i & & & \\
\text{s.t.} & y_i \left(\omega^\top \mathbf{x}_i + \beta \right) \geq 1 - \xi_i & i \in I & & \omega, \beta, C \\
& \xi_i \geq 0 & i \in I & &
\end{array}
$$

The performance measure

$$\begin{array}{lll} \min_{\omega,\beta,\xi} & \|\omega\|^2 + C\sum_{i\in I}\xi_i & \\ \text{s.t.} & y_i\left(\omega^\top \mathbf{x}_i + \beta\right) \geq 1 - \xi_i & i \in I \qquad \omega, \beta, C \\ & \xi_i \geq 0 & i \in I \end{array}$$

The performance measure

- $\{(\mathbf{x}_i, y_i) : i \in I\}$: seen as a random sample of $(\mathbf{X}, Y)$
- Accuracy: $acc = P(Y(\omega^\top \mathbf{X} + \beta) > 0)$

The performance measure

$$\begin{array}{lllll} \min_{\omega,\beta,\xi} & \|\omega\|^2 + C\sum_{i\in I}\xi_i & & & \\ \text{s.t.} & y_i\left(\omega^\top \mathbf{x}_i + \beta\right) \geq 1 - \xi_i & i \in I & & \omega, \beta, C \\ & \xi_i \geq 0 & i \in I & & \end{array}$$

The performance measure

- $\{(\mathbf{x}_i, y_i) : i \in I\}$: seen as a random sample of $(\mathbf{X}, Y)$
- Accuracy: $acc = P(Y(\omega^\top \mathbf{X} + \beta) > 0)$
- Distribution of $(\mathbf{X}, Y)$: Unknown

(Asymmetric) costs

- Benítez-Peña, Blanquero, C., Ramírez-Cobo, "On Support Vector Machines under a multiple-cost scenario". Advances in Data Analysis and Classification, 2017.
- C., Martín-Barragán, Romero Morales. "Multi-group support vector machines with measurement costs: A biobjective approach". Discrete Applied Mathematics, 2008.
- He, Ma. *Imbalanced learning: foundations, algorithms, and applications.* Wiley, 2013.
- Maldonado, Pérez, Bravo. "Cost-based feature selection for support vector machines: An application in credit scoring". EJOR, 2017.
- Prati, Batista, Duarte Silva. "Class imbalance revisited: a new experimental setup to assess the performance of treatment methods". Knowledge and Information Systems. 2015.
- Turney. "Types of cost in inductive concept learning". 2002.

Performance measures

$$\begin{array}{lll} \min_{\omega,\beta,\xi} & \|\omega\|^2 + C\sum_{i\in I}\xi_i & \\ \text{s.t.} & y_i\left(\omega^\top \mathbf{x}_i + \beta\right) \geq 1 - \xi_i & i \in I \\ & \xi_i \geq 0 & i \in I \end{array} \qquad \omega, \beta, C$$

Performance measures $\pi(\omega, \beta)$:

- Accuracy: $acc = P(Y(\omega^\top \mathbf{X} + \beta) > 0)$

Performance measures

$$\begin{array}{lll} \min_{\omega,\beta,\xi} & \|\omega\|^2 + C\sum_{i\in I}\xi_i & \\ \text{s.t.} & y_i\left(\omega^\top \mathbf{x}_i + \beta\right) \geq 1 - \xi_i & i \in I \qquad \omega, \beta, C \\ & \xi_i \geq 0 & i \in I \end{array}$$

Performance measures $\pi(\omega, \beta)$:

- Accuracy: $acc = P(Y(\omega^\top \mathbf{X} + \beta) > 0)$
- Sensitivity: $TPR = P(\omega^\top \mathbf{X} + \beta > 0 | Y = 1)$
- Specificity: $TNR = P(\omega^\top \mathbf{X} + \beta < 0 | Y = -1)$
- Youden's index:
  $J = TPR + TNR - 1 = P(\omega^\top \mathbf{X} + \beta > 0 | Y = 1) + P(\omega^\top \mathbf{X} + \beta < 0 | Y = -1) - 1$
- Positive Predictive Value: $PPV = P(Y = 1 | \omega^\top \mathbf{X} + \beta > 0)$
- Negative Predictive Value: $NPV = P(Y = -1 | \omega^\top \mathbf{X} + \beta < 0)$
- ...

- Performance measures $\pi_\ell(\omega, \beta), \ell \in L$
- Threshold values γ_ℓ for π_ℓ, $\ell \in L$
- I : training sample $\{(\mathbf{x}_i, y_i) : i \in I\}$

- Performance measures $\pi_\ell(\omega, \beta), \ell \in L$
- Threshold values γ_ℓ for π_ℓ, $\ell \in L$
- I : training sample $\{(\mathbf{x}_i, y_i) : i \in I\}$
- J : anchor sample $\{(\mathbf{x}_j, y_j) : j \in J\}$ ($J \cap I = \emptyset$)

- Performance measures $\pi_\ell(\omega, \beta), \ell \in L$
- Threshold values γ_ℓ for π_ℓ, $\ell \in L$
- I : training sample $\{(\mathbf{x}_i, y_i) : i \in I\}$
- J : anchor sample $\{(\mathbf{x}_j, y_j) : j \in J\}$ ($J \cap I = \emptyset$)
- Estimates of performance measures: $\widehat{\pi}_\ell(\omega, \beta; J), \ell \in L$

- Performance measures $\pi_\ell(\omega, \beta), \ell \in L$
- Threshold values γ_ℓ for π_ℓ, $\ell \in L$
- I : training sample $\{(\mathbf{x}_i, y_i) : i \in I\}$
- J : anchor sample $\{(\mathbf{x}_j, y_j) : j \in J\}$ ($J \cap I = \emptyset$)
- Estimates of performance measures: $\widehat{\pi}_\ell(\omega, \beta; J), \ell \in L$
- Desired: $\pi_\ell(\omega, \beta) \geq \gamma_\ell$, $\ell \in L$

- Performance measures $\pi_\ell(\omega, \beta), \ell \in L$
- Threshold values γ_ℓ for π_ℓ, $\ell \in L$
- I : training sample $\{(\mathbf{x}_i, y_i) : i \in I\}$
- J : anchor sample $\{(\mathbf{x}_j, y_j) : j \in J\}$ ($J \cap I = \emptyset$)
- Estimates of performance measures: $\widehat{\pi}_\ell(\omega, \beta; J), \ell \in L$
- Desired: $\pi_\ell(\omega, \beta) \geq \gamma_\ell$, $\ell \in L$
- Imposed: $\widehat{\pi}_\ell(\omega, \beta; J) \geq \gamma_\ell^*$, $\ell \in L$

- Performance measures $\pi_\ell(\omega, \beta), \ell \in L$
- Threshold values γ_ℓ for π_ℓ, $\ell \in L$
- I : training sample $\{(\mathbf{x}_i, y_i) : i \in I\}$
- J : anchor sample $\{(\mathbf{x}_j, y_j) : j \in J\}$ ($J \cap I = \emptyset$)
- Estimates of performance measures: $\widehat{\pi}_\ell(\omega, \beta; J), \ell \in L$
- Desired: $\pi_\ell(\omega, \beta) \geq \gamma_\ell$, $\ell \in L$
- Imposed: $\widehat{\pi}_\ell(\omega, \beta; J) \geq \gamma_\ell^*$, $\ell \in L$

Standard approach

$$
\begin{array}{lll}
\min_{\omega,\beta,\xi} & \|\omega\|^2 + C \sum_{i \in I} \xi_i & \\
\text{s.t.} & y_i \left(\omega^\top \mathbf{x}_i + \beta\right) \geq 1 - \xi_i & i \in I \\
& \xi_i \geq 0 & i \in I
\end{array}
$$

- Performance measures $\pi_\ell(\omega, \beta), \ell \in L$
- Threshold values γ_ℓ for π_ℓ, $\ell \in L$
- I : training sample $\{(\mathbf{x}_i, y_i) : i \in I\}$
- J : anchor sample $\{(\mathbf{x}_j, y_j) : j \in J\}$ ($J \cap I = \emptyset$)
- Estimates of performance measures: $\widehat{\pi}_\ell(\omega, \beta; J), \ell \in L$
- Desired: $\pi_\ell(\omega, \beta) \geq \gamma_\ell$, $\ell \in L$
- Imposed: $\widehat{\pi}_\ell(\omega, \beta; J) \geq \gamma_\ell^*$, $\ell \in L$

Constrained approach

$$
\begin{array}{lll}
\min_{\omega,\beta,\xi} & \|\omega\|^2 + C \sum_{i \in I} \xi_i & \\
\text{s.t.} & y_i \left(\omega^\top \mathbf{x}_i + \beta\right) \geq 1 - \xi_i & i \in I \\
& \xi_i \geq 0 & i \in I \\
& \widehat{\pi}_\ell(\omega, \beta; J) \geq \gamma_\ell^* & \ell \in L
\end{array}
$$

Adding constraints to an SVM model

$$
\begin{array}{lll}
\min_{\omega,\beta,\xi} & \|\omega\|^2 + C\sum_{i\in I}\xi_i & \\
\text{s.t.} & y_i\left(\omega^\top \mathbf{x}_i + \beta\right) \geq 1 - \xi_i & i \in I \\
& \xi_i \geq 0 & i \in I \\
& (\omega, \beta) \in \Omega &
\end{array}
$$

Ω : some (polyhedral) regions forced to be in one side of the separating hyperplane

Adding constraints to an SVM model

$$
\begin{array}{lll}
\min_{\omega,\beta,\xi} & \|\omega\|^2 + C\sum_{i\in I}\xi_i & \\
\text{s.t.} & y_i\left(\omega^\top \mathbf{x}_i + \beta\right) \geq 1 - \xi_i & i \in I \\
& \xi_i \geq 0 & i \in I \\
& (\omega, \beta) \in \Omega &
\end{array}
$$

Ω : some (polyhedral) regions forced to be in one side of the separating hyperplane

- C., and Plastria, "Optimal expected-distance separating halfspace", *Maths of OR*, 2008.
- Fung, , Mangasarian, and Shavlik, "Knowledge-based support vector machine classifiers". In *Advances in NIPS*, 2002
- Lauer and Bloch, "Incorporating prior knowledge in support vector machines for classification: A review", *Neurocomputing*, 2008.
- Mangasarian, "Knowledge-based linear programming", *SIAM Journal on Optimization*, 2005.
- Mangasarian and Wild, "Nonlinear knowledge-based classification", *IEEE Transactions on Neural Networks*, 2008.

- Desired: $\pi_\ell(\omega, \beta) \geq \gamma_\ell$, $\ell \in L$
- Imposed: $\widehat{\pi}_\ell(\omega, \beta; J) \geq \gamma^*_\ell$, $\ell \in L$

- Desired: $\pi_\ell(\omega, \beta) \geq \gamma_\ell$, $\ell \in L$
- Imposed: $\widehat{\pi}_\ell(\omega, \beta; J) \geq \gamma_\ell^*$, $\ell \in L$
- γ_ℓ^* : so that H_0 cannot be rejected in the test hypothesis

$$\begin{cases} H_0: & \pi_\ell(\omega, \beta) \geq \gamma_\ell \\ H_1: & \pi_\ell(\omega, \beta) < \gamma_\ell \end{cases}$$

- Desired: $\pi_\ell(\omega, \beta) \geq \gamma_\ell$, $\ell \in L$
- Imposed: $\widehat{\pi}_\ell(\omega, \beta; J) \geq \gamma_\ell^*$, $\ell \in L$
- γ_ℓ^* : so that H_0 cannot be rejected in the test hypothesis

$$\begin{cases} H_0 : & \pi_\ell(\omega, \beta) \geq \gamma_\ell \\ H_1 : & \pi_\ell(\omega, \beta) < \gamma_\ell \end{cases}$$

Building γ_ℓ^*

- Hoeffding's inequality: for $Z_1, \ldots, Z_n$ *i.i.d.*, $Be(p)$, $P(\bar{Z} - p \geq c) \leq e^{-2nc^2}$.
- $100(1-\alpha)\%$ CI for p :

$$\left(\bar{Z} - \sqrt{\frac{\log \alpha}{-2n}}, 1 \right)$$

- Imposing $p_0 \in CI$ means

$$\bar{Z} \geq p_0 + \sqrt{\frac{\log \alpha}{-2n}}$$

- Desired: $\pi_\ell(\omega, \beta) \geq \gamma_\ell$, $\ell \in L$
- Imposed: $\widehat{\pi}_\ell(\omega, \beta; J) \geq \gamma_\ell^*$, $\ell \in L$
- γ_ℓ^* : so that H_0 cannot be rejected in the test hypothesis

$$\begin{cases} H_0 : & \pi_\ell(\omega, \beta) \geq \gamma_\ell \\ H_1 : & \pi_\ell(\omega, \beta) < \gamma_\ell \end{cases}$$

Building γ_ℓ^*

- Hoeffding's inequality: for $Z_1, \ldots, Z_n$ *i.i.d.*, $Be(p)$, $P(\bar{Z} - p \geq c) \leq e^{-2nc^2}$.
- $100(1-\alpha)\%$ CI for p :

$$\left(\bar{Z} - \sqrt{\frac{\log \alpha}{-2n}}, 1 \right)$$

- Imposing $p_0 \in CI$ means

$$\bar{Z} \geq p_0 + \sqrt{\frac{\log \alpha}{-2n}}$$

- $\gamma_\ell^* = \gamma_\ell + \sqrt{\dfrac{\log \alpha}{-2|J|}}$

Feasibility?

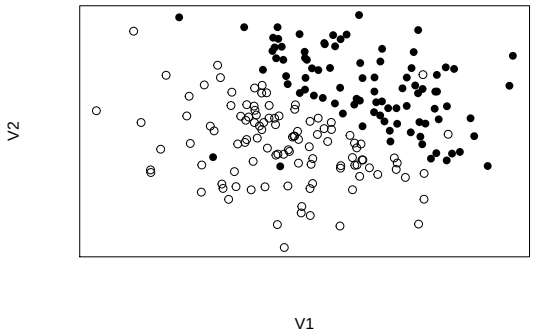
Always feasible:

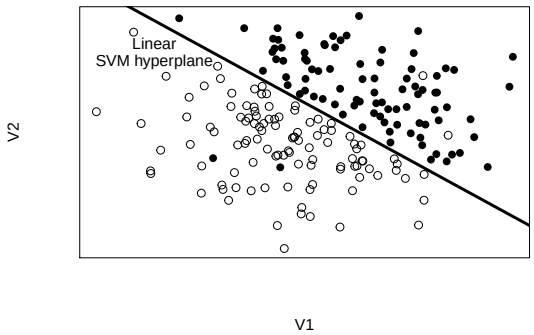
$$\begin{array}{lll} \min_{\omega,\beta,\xi} & \|\omega\|^2 + C\sum_{i\in I}\xi_i & \\ \text{s.t.} & y_i\left(\omega^\top \mathbf{x}_i + \beta\right) \geq 1 - \xi_i & i \in I \\ & \xi_i \geq 0 & i \in I \end{array}$$

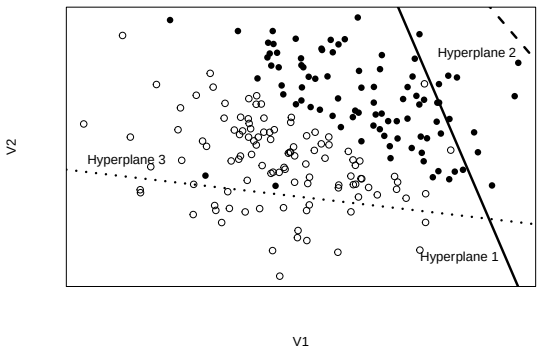
Maybe unfeasible:

$$\begin{array}{lll} \min_{\omega,\beta,\xi} & \|\omega\|^2 + C\sum_{i\in I}\xi_i & \\ \text{s.t.} & y_i\left(\omega^\top \mathbf{x}_i + \beta\right) \geq 1 - \xi_i & i \in I \\ & \xi_i \geq 0 & i \in I \\ & \widehat{\pi}_\ell(\omega, \beta; J) \geq \gamma_\ell^* & \ell \in L \end{array}$$

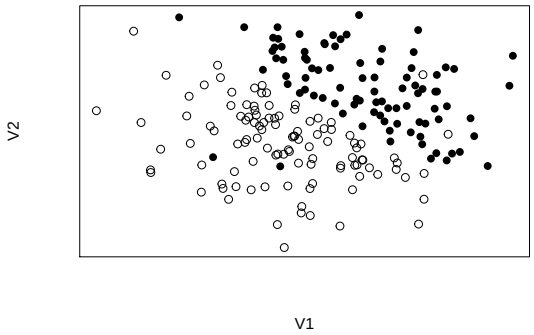
Coping with (eventual) unfeasibility

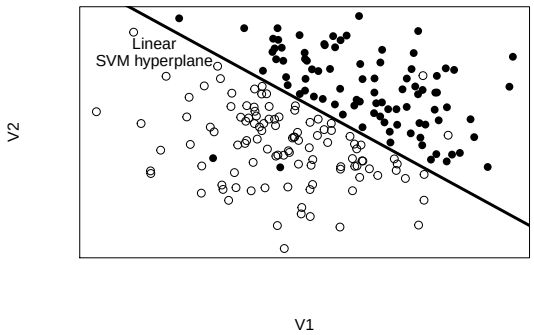


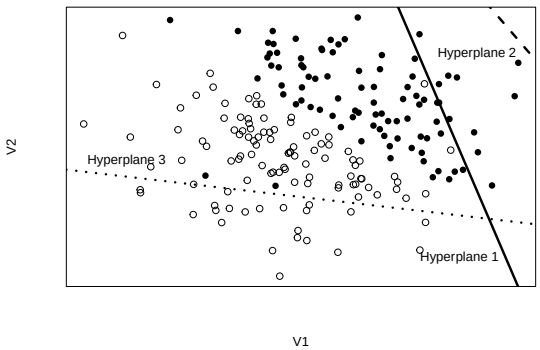




Coping with (eventual) unfeasibility







Playing with VC dimension

- Map the dataset $\{\mathbf{x}_i : i \in I\}$ onto $\{\Phi(\mathbf{x})_i : i \in I\}$ so that the transformed set is separable

$$
\begin{array}{lll}
\min_{\omega,\beta,\xi} & \|\omega\|^2 + C\sum_{i\in I}\xi_i & \\
\text{s.t.} & y_i\left(\omega^\top \mathbf{x}_i + \beta\right) \geq 1 - \xi_i & i \in I \\
& \xi_i \geq 0 & i \in I \\
& \widehat{\pi}_\ell(\omega, \beta; J) \geq \gamma_\ell^* & \ell \in L
\end{array}
$$

$$
\begin{array}{lll}
\min_{\omega,\beta,\xi} & \|\omega\|^2 + C\sum_{i\in I}\xi_i & \\
\text{s.t.} & y_i\left(\omega^\top \mathbf{x}_i + \beta\right) \geq 1 - \xi_i & i \in I \\
& \xi_i \geq 0 & i \in I \\
& \widehat{\pi}_\ell(\omega, \beta; J) \geq \gamma_\ell^* & \ell \in L
\end{array}
\qquad
z_j = \begin{cases} 1, & \text{if } y_j\left(\omega^\top \mathbf{x}_j + \beta\right) \geq 1 \\ 0, & \text{else} \end{cases}
\qquad j \in J
$$

$$
\begin{array}{lll}
\min_{\omega,\beta,\xi} & \|\omega\|^2 + C\sum_{i\in I}\xi_i & \\
\text{s.t.} & y_i\left(\omega^\top \mathbf{x}_i + \beta\right) \geq 1 - \xi_i & i \in I \\
& \xi_i \geq 0 & i \in I \\
& \widehat{\pi}_\ell(\omega, \beta; J) \geq \gamma_\ell^* & \ell \in L
\end{array}
\qquad
z_j = \begin{cases} 1, & \text{if } y_j\left(\omega^\top \mathbf{x}_j + \beta\right) \geq 1 \\ 0, & \text{else} \end{cases}
\qquad j \in J
$$

$$
\widehat{TPR}(\omega, \beta; J) \geq \gamma
$$

$$
\sum_{j\in J: y_j = 1} z_j \geq \gamma \, \#\left(\{j \in J : y_j = 1\}\right)
$$

$$
\begin{array}{lll}
\min_{\omega,\beta,\xi} & \|\omega\|^2 + C\sum_{i\in I}\xi_i & \\
\text{s.t.} & y_i\left(\omega^\top \mathbf{x}_i + \beta\right) \geq 1 - \xi_i & i \in I \\
& \xi_i \geq 0 & i \in I \\
& \widehat{\pi}_\ell(\omega, \beta; J) \geq \gamma_\ell^* & \ell \in L
\end{array}
\qquad
z_j = \begin{cases} 1, & \text{if } y_j\left(\omega^\top \mathbf{x}_j + \beta\right) \geq 1 \\ 0, & \text{else} \end{cases}
\qquad j \in J
$$

$$
\widehat{TNR}(\omega, \beta; J) \geq \gamma
$$

$$
\sum_{j\in J: y_j = -1} z_j \geq \gamma \, \#\left(\{j \in J : y_j = -1\}\right)
$$

$$
\begin{array}{lll}
\min_{\omega,\beta,\xi} & \|\omega\|^2 + C\sum_{i\in I}\xi_i & \\
\text{s.t.} & y_i\left(\omega^\top \mathbf{x}_i + \beta\right) \geq 1 - \xi_i & i \in I \\
& \xi_i \geq 0 & i \in I \\
& \widehat{\pi}_\ell(\omega, \beta; J) \geq \gamma_\ell^* & \ell \in L
\end{array}
\qquad
z_j = \begin{cases} 1, & \text{if } y_j\left(\omega^\top \mathbf{x}_j + \beta\right) \geq 1 \\ 0, & \text{else} \end{cases}
\qquad j \in J
$$

$$
\widehat{J}(\omega, \beta; J) \geq \gamma
$$

$$
\sum_{j\in J} z_j \geq \gamma \,\#\,(J)
$$

$$
\begin{array}{lll}
\min_{\omega,\beta,\xi} & \|\omega\|^2 + C\sum_{i\in I}\xi_i & \\
\text{s.t.} & y_i\left(\omega^\top \mathbf{x}_i + \beta\right) \geq 1 - \xi_i & i \in I \\
& \xi_i \geq 0 & i \in I \\
& \widehat{\pi}_\ell(\omega, \beta; J) \geq \gamma_\ell^* & \ell \in L
\end{array}
\qquad
z_j = \begin{cases} 1, & \text{if } y_j\left(\omega^\top \mathbf{x}_j + \beta\right) \geq 1 \\ 0, & \text{else} \end{cases}
\qquad j \in J
$$

$$
\widehat{TPR_m}(\omega, \beta; J) \geq \gamma
$$

$$
\sum_{j\in J: u_j = m} z_j \geq \gamma \, \#\left(\{j \in J : u_j = m\}\right)
$$

$$
\begin{array}{lll}
\min_{\omega,\beta,\xi} & \|\omega\|^2 + C\sum_{i\in I}\xi_i & \\
\text{s.t.} & y_i\left(\omega^\top \mathbf{x}_i + \beta\right) \geq 1 - \xi_i & i \in I \\
& \xi_i \geq 0 & i \in I \\
& \widehat{\pi}_\ell(\omega, \beta; J) \geq \gamma_\ell^* & \ell \in L
\end{array}
\qquad
z_j = \begin{cases} 1, & \text{if } y_j\left(\omega^\top \mathbf{x}_j + \beta\right) \geq 1 \\ 0, & \text{else} \end{cases}
\qquad j \in J
$$

$$
\widehat{PPV}(\omega, \beta; J) \geq \gamma
$$

$$
(1-\gamma)\, prev_+ \sum_{j\in J: y_j = 1} z_j - \gamma\,(1 - prev_+) \sum_{j\in J: y_j = -1} (1 - z_j) \geq 0
$$

$$
\begin{array}{lll}
\min_{\omega,\beta,\xi} & \|\omega\|^2 + C\sum_{i\in I}\xi_i & \\
\text{s.t.} & y_i\left(\omega^\top \mathbf{x}_i + \beta\right) \geq 1 - \xi_i & i \in I \\
& \xi_i \geq 0 & i \in I \\
& \widehat{\pi}_\ell(\omega, \beta; J) \geq \gamma^*_\ell & \ell \in L
\end{array}
\qquad
z_j = \begin{cases} 1, & \text{if } y_j\left(\omega^\top \mathbf{x}_j + \beta\right) \geq 1 \\ 0, & \text{else} \end{cases}
\qquad j \in J
$$

$$
\widehat{NPV}(\omega, \beta; J) \geq \gamma
$$

$$
(1-\gamma)\, prev_- \sum_{j\in J: y_j=-1} z_j - \gamma\,(1-prev_-) \sum_{j\in J: y_j=1} (1-z_j) \geq 0
$$

$$
\begin{array}{lll}
\min_{\omega,\beta,\xi} & \|\omega\|^2 + C\sum_{i\in I}\xi_i & \\
\text{s.t.} & y_i\left(\omega^\top \mathbf{x}_i + \beta\right) \geq 1 - \xi_i & i \in I \\
& \xi_i \geq 0 & i \in I \\
& \widehat{\pi}_\ell(\omega, \beta; J) \geq \gamma_\ell^* & \ell \in L
\end{array}
\qquad
z_j = \begin{cases} 1, & \text{if } y_j\left(\omega^\top \mathbf{x}_j + \beta\right) \geq 1 \\ 0, & \text{else} \end{cases}
\qquad j \in J
$$

$$\widehat{\pi}_\ell(\omega, \beta; J) \geq \gamma_\ell^*$$

$$\mathbf{a}_\ell^\top \mathbf{z} \geq b_\ell$$

$$
\begin{array}{lll}
\min_{\omega,\beta,\xi} & \|\omega\|^2 + C\sum_{i\in I}\xi_i & \\
\text{s.t.} & y_i\left(\omega^\top \mathbf{x}_i + \beta\right) \geq 1 - \xi_i & i \in I \\
& \xi_i \geq 0 & i \in I \\
& \widehat{\pi}_\ell(\omega, \beta; J) \geq \gamma_\ell^* & \ell \in L
\end{array}
\qquad
z_j = \begin{cases} 1, & \text{if } y_j\left(\omega^\top \mathbf{x}_j + \beta\right) \geq 1 \\ 0, & \text{else} \end{cases}
\qquad j \in J
$$

$$\widehat{\pi}_\ell(\omega, \beta; J) \geq \gamma_\ell^*$$

$$\mathbf{a}_\ell^\top \mathbf{z} \geq b_\ell$$

$$
\begin{array}{lll}
\min_{\omega,\beta,\xi,\mathbf{z}} & \|\omega\|^2 + C\sum_{i\in I}\xi_i & \\
\text{s.t.} & y_i\left(\omega^\top \mathbf{x}_i + \beta\right) \geq 1 - \xi_i & i \in I \\
& \xi_i \geq 0 & i \in I \\
& \mathbf{a}_\ell^\top z \geq b_\ell & \ell \in L \\
& z_\ell \in \{0, 1\} & \ell \in L \\
& y_j\left(\omega^\top \mathbf{x}_j + \beta\right) \geq 1 - M(1 - z_j) & j \in J
\end{array}
$$

$$
\begin{array}{lll}
\min_{\omega,\beta,\xi,\mathbf{z}} & \|\omega\|^2 + C\sum_{i\in I}\xi_i & \\
\text{s.t.} & y_i\left(\omega^\top \mathbf{x}_i + \beta\right) \geq 1 - \xi_i & i \in I \\
& \xi_i \geq 0 & i \in I \\
& \mathbf{a}_l^\top z \geq b_l & l \in L \\
& z_\ell \in \{0,1\} & \ell \in L \\
& y_j\left(\omega^\top \mathbf{x}_j + \beta\right) \geq 1 - M(1 - z_j) & j \in J
\end{array}
$$

- Denote $J(z) = \{j \in J : z_j = 1\}$

$$
\begin{array}{lll}
\min_{\mathbf{z}} & & \\
\text{s.t.} & z_\ell \in \{0,1\} & \ell \in L \\
& \mathbf{a}_\ell^\top z \geq b_\ell & l \in L
\end{array}
\qquad
\begin{array}{lll}
\min_{\omega,\beta,\xi} & \omega^\top \omega + C \sum_{i \in I} \xi_i & \\
\text{s.t.} & y_i \left(\omega^\top \mathbf{x}_i + \beta\right) \geq 1 - \xi_i & i \in I \\
& y_j \left(\omega^\top \mathbf{x}_j + \beta\right) \geq 1 & j \in J(\mathbf{z}) \\
& \xi_i \geq 0 & i \in I
\end{array}
$$

- Denote $J(z) = \{j \in J : z_j = 1\}$

$$
\begin{array}{lllll}
\min_{\mathbf{z}} & & & \min_{\omega,\beta,\xi} & \omega^\top \omega + C \sum_{i \in I} \xi_i \\
\text{s.t.} & z_\ell \in \{0,1\} & \ell \in L & \text{s.t.} & y_i \left(\omega^\top \mathbf{x}_i + \beta\right) \geq 1 - \xi_i \quad i \in I \\
& \mathbf{a}_\ell^\top z \geq b_\ell & l \in L & & y_j \left(\omega^\top \mathbf{x}_j + \beta\right) \geq 1 \quad j \in J(\mathbf{z}) \\
& & & & \xi_i \geq 0 \quad i \in I
\end{array}
$$

KKT conditions for inner problem ($\mathbf{z}$ fixed)

$$
\begin{array}{rcll}
\omega & = & \sum_{s \in I} \lambda_s y_s \mathbf{x}_s + \sum_{t \in J(\mathbf{z})} \mu_t y_t \mathbf{x}_t & \\
0 & = & \sum_{s \in I} \lambda_s y_s + \sum_{t \in J(\mathbf{z})} \mu_t y_t & \\
0 & \leq & \lambda_s \leq C/2 & s \in I \\
0 & \leq & \mu_t & t \in J(\mathbf{z})
\end{array}
$$

- Denote $J(z) = \{j \in J : z_j = 1\}$

$$
\begin{array}{lllll}
\min_{\mathbf{z}} & & & \min_{\omega,\beta,\xi} & \omega^\top \omega + C \sum_{i \in I} \xi_i \\
\text{s.t.} & z_\ell \in \{0,1\} & \ell \in L & \text{s.t.} & y_i \left(\omega^\top \mathbf{x}_i + \beta\right) \geq 1 - \xi_i \quad i \in I \\
& \mathbf{a}_\ell^\top z \geq b_\ell & l \in L & & y_j \left(\omega^\top \mathbf{x}_j + \beta\right) \geq 1 \quad j \in J(\mathbf{z}) \\
& & & & \xi_i \geq 0 \quad i \in I
\end{array}
$$

KKT conditions for inner problem ($\mathbf{z}$ fixed)

$$
\begin{array}{rcll}
\omega & = & \sum_{s \in I} \lambda_s y_s \mathbf{x}_s + \sum_{t \in J} \mu_t y_t \mathbf{x}_t & \\
0 & = & \sum_{s \in I} \lambda_s y_s + \sum_{t \in J} \mu_t y_t & \\
0 & \leq & \lambda_s \leq C/2 & s \in I \\
0 & \leq & \mu_t \leq M z_t & t \in J
\end{array}
$$

(Partial) Dual

$$
\begin{array}{lll}
\min_{\lambda,\mu,\beta,\xi,\mathbf{z}} & \left(\sum_{s\in I}\lambda_s y_s \mathbf{x}_s + \sum_{t\in J}\mu_t y_t \mathbf{x}_t\right)^\top \left(\sum_{s\in I}\lambda_s y_s \mathbf{x}_s + \sum_{t\in J}\mu_t y_t \mathbf{x}_t\right) & \\
& +C\sum_{i\in I}\xi_i & \\
\text{s.t.} & z_\ell \in \{0,1\} & \ell \in L \\
& \mathbf{a}_\ell^\top z \geq b_\ell & \ell \in L \\
& y_i\left(\left(\sum_{s\in I}\lambda_s y_s \mathbf{x}_s + \sum_{t\in J}\mu_t y_t \mathbf{x}_t\right)^\top \mathbf{x}_i + \beta\right) \geq 1-\xi_i & i \in I \\
& y_j\left(\left(\sum_{s\in I}\lambda_s y_s \mathbf{x}_s + \sum_{t\in J}\mu_t y_t \mathbf{x}_t\right)^\top \mathbf{x}_j + \beta\right) \geq 1-M(1-z_j) & j \in J \\
& \xi_i \geq 0 & i \in I \\
& 0 \leq \lambda_i \leq C/2 & i \in I \\
& 0 \leq \mu_j \leq M z_j & j \in J
\end{array}
$$

(Partial) Dual: The kernel trick

$$
\begin{array}{lll}
\min & \sum_{s,s' \in I} \lambda_s y_s \lambda_{s'} y_{s'} K(\mathbf{x}_s, \mathbf{x}_{s'}) + \sum_{t,t' \in J} \mu_t y_t \mu_{t'} y_{t'} K(\mathbf{x}_t, \mathbf{x}_{t'}) & \\
& +2 \sum_{s \in I, t \in J} \lambda_s y_s \lambda_t y_t K(\mathbf{x}_s, \mathbf{x}_t) + C \sum_{i \in I} \xi_i & \\
\text{s.t.} & z_\ell \in \{0,1\} & \ell \in L \\
& \mathbf{a}_\ell^\top z \geq b_\ell & \ell \in L \\
& y_i \left(\sum_{s \in I} \lambda_s y_s K(\mathbf{x}_s, \mathbf{x}_i) + \sum_{t \in J} \mu_t y_t K(\mathbf{x}_t, \mathbf{x}_i) + \beta \right) \geq 1 - \xi_i & i \in I \\
& y_j \left(\sum_{s \in I} \lambda_s y_s K(\mathbf{x}_s, \mathbf{x}_j) + \sum_{t \in J} \mu_t y_t K(\mathbf{x}_t, \mathbf{x}_j) + \beta \right) \geq 1 - M(1 - z_j) & j \in J \\
& \xi_i \geq 0 & i \in I \\
& 0 \leq \lambda_i \leq C/2 & i \in I \\
& 0 \leq \mu_j \leq M z_j & j \in J
\end{array}
$$

(Partial) Dual: The kernel trick

$$
\begin{array}{lll}
\min & \sum_{s,s' \in I} \lambda_s y_s \lambda_{s'} y_{s'} K(\mathbf{x}_s, \mathbf{x}_{s'}) + \sum_{t,t' \in J} \mu_t y_t \mu_{t'} y_{t'} K(\mathbf{x}_t, \mathbf{x}_{t'}) & \\
& +2 \sum_{s \in I, t \in J} \lambda_s y_s \lambda_t y_t K(\mathbf{x}_s, \mathbf{x}_t) + C \sum_{i \in I} \xi_i & \\
\text{s.t.} & z_\ell \in \{0,1\} & \ell \in L \\
& \mathbf{a}_\ell^\top z \geq b_\ell & \ell \in L \\
& y_i \left(\sum_{s \in I} \lambda_s y_s K(\mathbf{x}_s, \mathbf{x}_i) + \sum_{t \in J} \mu_t y_t K(\mathbf{x}_t, \mathbf{x}_i) + \beta \right) \geq 1 - \xi_i & i \in I \\
& y_j \left(\sum_{s \in I} \lambda_s y_s K(\mathbf{x}_s, \mathbf{x}_j) + \sum_{t \in J} \mu_t y_t K(\mathbf{x}_t, \mathbf{x}_j) + \beta \right) \geq 1 - M(1 - z_j) & j \in J \\
& \xi_i \geq 0 & i \in I \\
& 0 \leq \lambda_i \leq C/2 & i \in I \\
& 0 \leq \mu_j \leq M z_j & j \in J
\end{array}
$$

Parameters involved:

- C, to be tuned
- M, to be fixed

(Partial) Dual: The kernel trick

$$
\begin{array}{lll}
\min & \sum_{s,s' \in I} \lambda_s y_s \lambda_{s'} y_{s'} K(\mathbf{x}_s, \mathbf{x}_{s'}) + \sum_{t,t' \in J} \mu_t y_t \mu_{t'} y_{t'} K(\mathbf{x}_t, \mathbf{x}_{t'}) & \\
& +2 \sum_{s \in I, t \in J} \lambda_s y_s \lambda_t y_t K(\mathbf{x}_s, \mathbf{x}_t) + C \sum_{i \in I} \xi_i & \\
\text{s.t.} & z_\ell \in \{0,1\} & \ell \in L \\
& \mathbf{a}_\ell^\top z \geq b_\ell & \ell \in L \\
& y_i \left(\sum_{s \in I} \lambda_s y_s K(\mathbf{x}_s, \mathbf{x}_i) + \sum_{t \in J} \mu_t y_t K(\mathbf{x}_t, \mathbf{x}_i) + \beta \right) \geq 1 - \xi_i & i \in I \\
& y_j \left(\sum_{s \in I} \lambda_s y_s K(\mathbf{x}_s, \mathbf{x}_j) + \sum_{t \in J} \mu_t y_t K(\mathbf{x}_t, \mathbf{x}_j) + \beta \right) \geq 1 - M(1 - z_j) & j \in J \\
& \xi_i \geq 0 & i \in I \\
& 0 \leq \lambda_i \leq C/2 & i \in I \\
& 0 \leq \mu_j \leq M z_j & j \in J
\end{array}
$$

Parameters involved:

- C, to be tuned
- M, to be fixed?

(Partial) Dual: The kernel trick

$$
\begin{array}{lll}
\min & \sum_{s,s' \in I} \lambda_s y_s \lambda_{s'} y_{s'} K(\mathbf{x}_s, \mathbf{x}_{s'}) + \sum_{t,t' \in J} \mu_t y_t \mu_{t'} y_{t'} K(\mathbf{x}_t, \mathbf{x}_{t'}) & \\
& +2 \sum_{s \in I, t \in J} \lambda_s y_s \lambda_t y_t K(\mathbf{x}_s, \mathbf{x}_t) + C \sum_{i \in I} \xi_i & \\
\text{s.t.} & z_\ell \in \{0,1\} & \ell \in L \\
& \mathbf{a}_\ell^\top z \geq b_\ell & \ell \in L \\
& y_i \left(\sum_{s \in I} \lambda_s y_s K(\mathbf{x}_s, \mathbf{x}_i) + \sum_{t \in J} \mu_t y_t K(\mathbf{x}_t, \mathbf{x}_i) + \beta \right) \geq 1 - \xi_i & i \in I \\
& y_j \left(\sum_{s \in I} \lambda_s y_s K(\mathbf{x}_s, \mathbf{x}_j) + \sum_{t \in J} \mu_t y_t K(\mathbf{x}_t, \mathbf{x}_j) + \beta \right) \geq 1 - M(1 - z_j) & j \in J \\
& \xi_i \geq 0 & i \in I \\
& 0 \leq \lambda_i \leq C/2 & i \in I \\
& 0 \leq \mu_j \leq M z_j & j \in J
\end{array}
$$

Parameters involved:

- C, to be tuned
- M, to be fixed?

Straightforward extension to several anchors

Experiments

- RBF kernel, parameters tuned by grid search
- Python + Gurobi
- $M = 100$, time.limit $= 300$ sec

Experiments

- RBF kernel, parameters tuned by grid search
- Python + Gurobi
- $M = 100$, time.limit $= 300$ sec

Data sets

Name	$\lvert\Omega\rvert$	V	$\lvert\Omega_+\rvert$	(%)
`wisconsin`	567	30	357	(62.7%)
`australian`	690	14	383	(55.5%)
`votes`	435	16	267	(61.4%)
`german`	1000	45	700	(70%)

Results. Increasing TNR (0.025)

Name		SVM		CSVM	
		Mean	Std	Mean	Std
wisconsin	TPR	0.990	0.017	0.945	0.045
	TNR	**0.948**	**0.049**	**0.965**	**0.037**
australian	TPR	0.863	0.079	0.772	0.081
	TNR	**0.830**	**0.071**	**0.903**	**0.050**
votes	TPR	0.963	0.040	0.846	0.097
	TNR	**0.951**	**0.031**	**0.978**	**0.038**
german	TPR	0.905	0.036	0.791	0.063
	TNR	**0.405**	**0.114**	**0.547**	**0.141**

Results. Increasing TPR (0.025)

Name		SVM		CSVM	
		Mean	Std	Mean	Std
wisconsin	TPR	**0.990**	**0.017**	**0.989**	**0.018**
	TNR	0.948	0.049	0.856	0.153
australian	TPR	**0.863**	**0.079**	**0.910**	**0.047**
	TNR	0.830	0.071	0.694	0.092
votes	TPR	**0.963**	**0.040**	**0.978**	**0.026**
	TNR	0.951	0.031	0.922	0.040
german					

Data sets

Name	$\lvert\Omega\rvert$	V	$\lvert\Omega_+\rvert$	(%)
`wisconsin`	567	30	357	(62.7%)
`australian`	690	14	383	(55.5%)
`votes`	435	16	267	(61.4%)
`german`	1000	45	700	(70%)

G-mean criterion

	SVM		CSVM (TNR$\geq$ 0.65)		CSVM (TNR $\geq$ 0.7)	
	Mean	Std	Mean	Std	Mean	Std
TPR	0.905	0.036	0.668	0.111	0.683	0.073
TNR	0.405	0.114	0.671	0.164	0.690	0.103

Feature selection

Benítez-Peña, Blanquero, C., Ramírez-Cobo, Cost-sensitive feature selection for support vector machines. Computers & OR, 2018.

Aim

- Find a minimum-cost (e.g. minimum-cardinality) set of features
  - Attaining $\widehat{\pi}_\ell(\omega, \beta) \geq \gamma_\ell^*$, $\ell \in L$
  - Hoping $\pi_\ell(\omega, \beta; I) \geq \gamma_\ell$, $\ell \in L$
- Once identified the features, solve an SVM

Feature selection. Linear kernel

$$
\begin{aligned}
\min_{\boldsymbol{w},\beta,z,\zeta} \quad & \sum_{k=1}^{N} \delta_k z_k \\
s.t. \quad & y_i(\boldsymbol{w}^\top x_i + \beta) \geq 1 - L(1-\zeta_i), && \forall i \in I \\
& \textstyle\sum_{i\in I} \zeta_i(1-y_i) \geq \lambda_{-1} \sum_{i\in I}(1-y_i) \\
& \textstyle\sum_{i\in I} \zeta_i(1+y_i) \geq \lambda_{1} \sum_{i\in I}(1+y_i) \\
& |w_k| \leq M z_k && \forall k \in 1,\ldots,N \\
& \zeta_i \in \{0,1\} && \forall i \in I \\
& z_k \in \{0,1\} && \forall k \in 1,\ldots,N
\end{aligned}
$$

Results. Linear kernel

Name		SVM		FS		Reduction
		Mean	Std	Mean	Std	
wisconsin	TPR	0.992	0.013	0.975	0.023	30 → 6.2 (0.919 Std)
	TNR	0.943	0.051	0.947	0.048	
votes	TPR	0.955	0.038	0.96	0.034	32 → 9.3 (1.16 Std)
	TNR	0.947	0.059	0.945	0.052	
nursery	TPR	1	0	1	0	19 → 1 (0 Std)
	TNR	1	0	1	0	
australian	TPR	0.769	0.083	0.772	0.074	34 → 5.75 (1.89 Std)
	TNR	0.912	0.05	0.924	0.053	
careval	TPR	0.96	0.022	0.962	0.018	15 → 11 (0 Std)
	TNR	0.948	0.024	0.935	0.039	

Results. Radial kernel

Supervised classification. The framework

- Given: set I of individuals, each $i \in I$ with associated
  - A vector $\mathbf{x}_i \in \mathcal{X}$, assumed here $\mathcal{X} \subset \mathbb{R}^m$
  - A label y_i, assumed here to be in $\{-1, +1\}$
- Seen as a sample from $(\mathbf{X}, Y)$, with unknown distribution
- Goal: to infer from I a classifier $\varphi : \mathcal{X} \longrightarrow \{-1, 1\}$ so that we can classify any object just by knowing $\mathbf{x}$

Supervised classification. The framework

- Given: set I of individuals, each $i \in I$ with associated
  - A vector $\mathbf{x}_i \in \mathcal{X}$, assumed here $\mathcal{X} \subset \mathbb{R}^m$
  - A label y_i, assumed here to be in $\{-1, +1\}$
- Seen as a sample from $(\mathbf{X}, Y)$, with unknown distribution
- Goal: to infer from I a classifier $\varphi : \mathcal{X} \longrightarrow \{-1, 1\}$ so that we can classify any object just by knowing $\mathbf{x}$

Inferring probabilities

I.o. φ, estimate $P(y = 1 \mid x)$, e.g.

$$P(y = 1 \mid x) = \frac{1}{1 + \exp(Af(x) + B)}$$


Hastie and Tibshirani. Classification by pairwise coupling. In Advances in neural information processing systems, 1998.


Platt, Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods. In Advances in large margin classifiers, 2000.

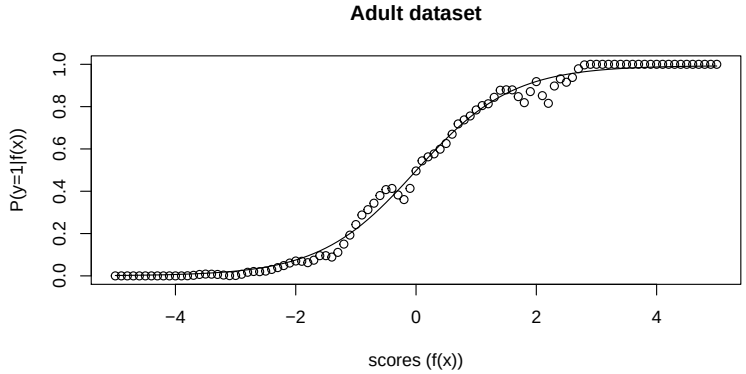

Sollich, Bayesian methods for support vector machines: Evidence and predictive class probabilities. Machine learning, 2002.

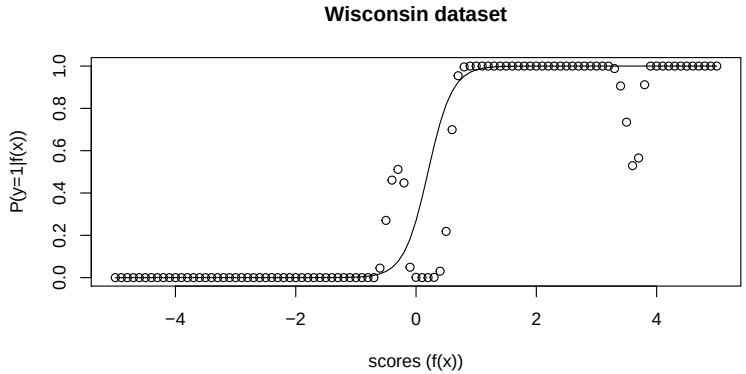

Tao, Wu, Wang, and Wang , Posterior probability support vector machines for unbalanced data. IEEE Transactions on Neural Networks, 2005.

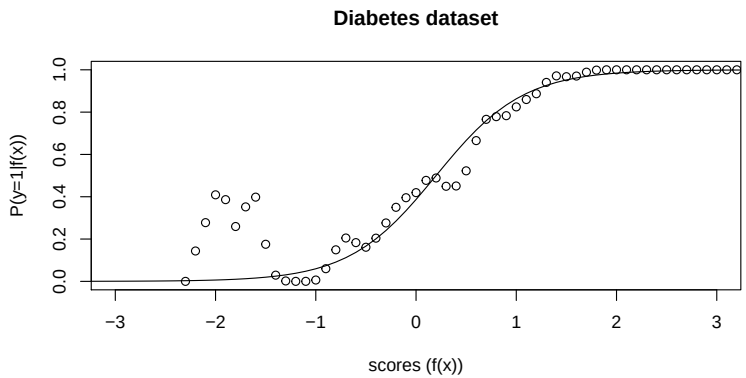

Vapnik, Statistical Learning Theory, 1998.


Wahba, Multivariate function and operator estimation, based on smoothing splines and reproducing kernels. In Santa Fe Institute Studies in the Sciences of Complexity-Proceedings, 1992


Wahba, Support vector machines, reproducing kernel Hilbert spaces and the randomized gacv. Advances in Kernel Methods-Support Vector Learning, 1999.

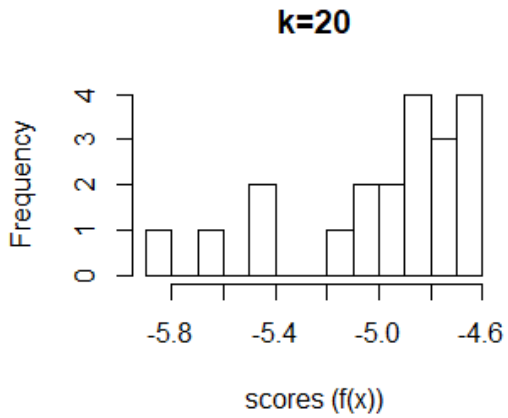

Adult dataset
P(y=1|f(x))
0.0
0.2
0.4
0.6
0.8
1.0
−4
−2
0
2
4
scores (f(x))

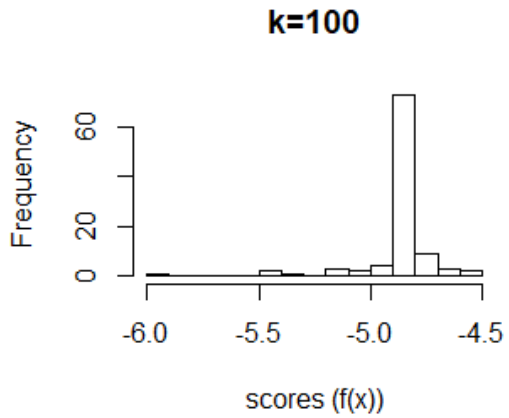

Wisconsin dataset
P(y=1|f(x))
0.0
0.2
0.4
0.6
0.8
1.0
−4
−2
0
2
4
scores (f(x))

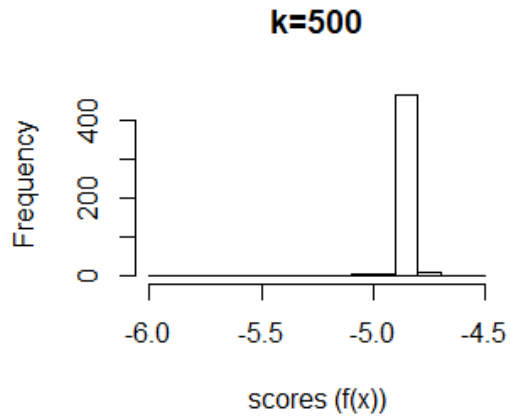

Diabetes dataset
P(y=1|f(x))
0.0
0.2
0.4
0.6
0.8
1.0
−3
−2
−1
0
1
2
3
scores (f(x))

Inferring probabilities. A frequentist approach

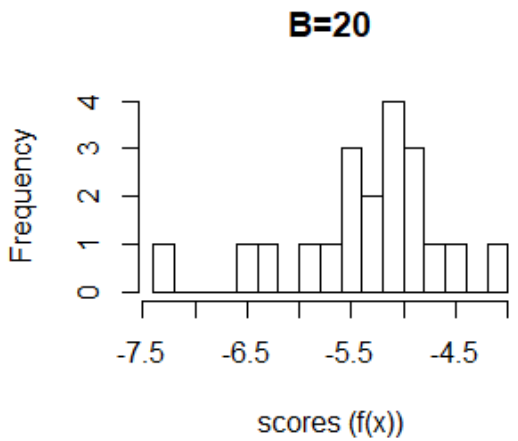
k-fold CV

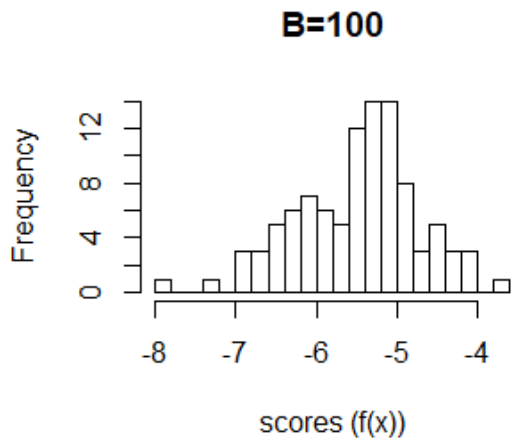


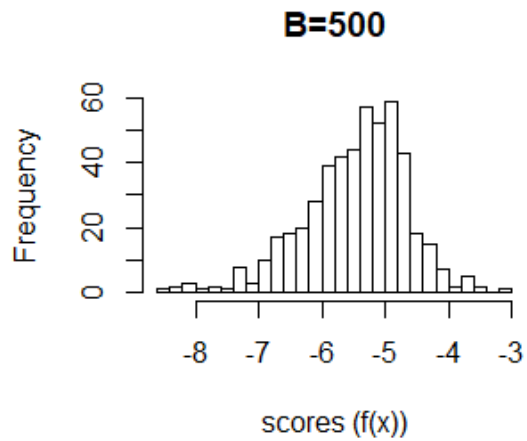


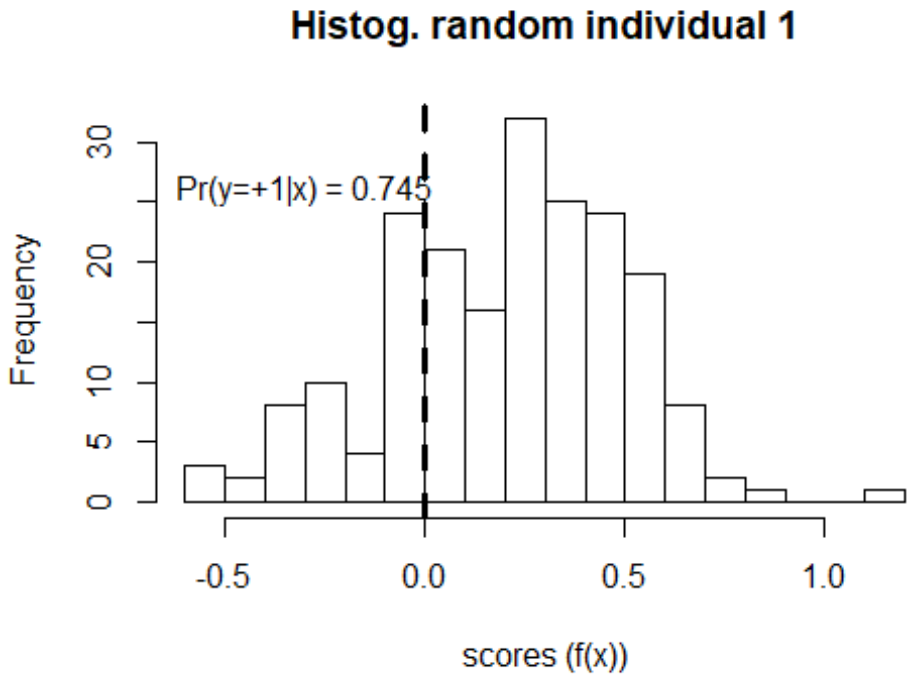


bootstrapped





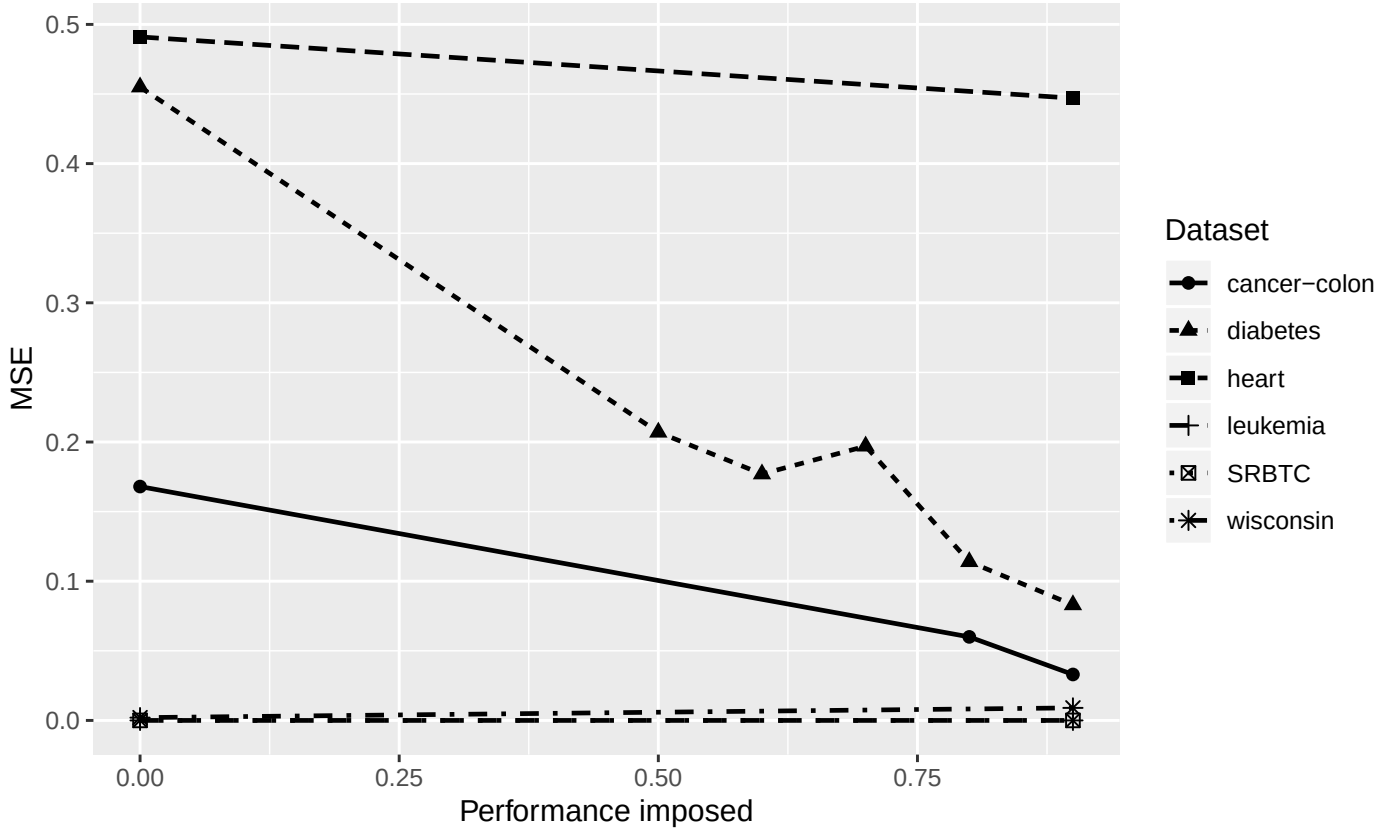



Histog. random individual 1
Pr(y=+1|x) = 0.745
Frequency
0
5
10
20
30
-0.5
0.0
0.5
1.0
scores (f(x))

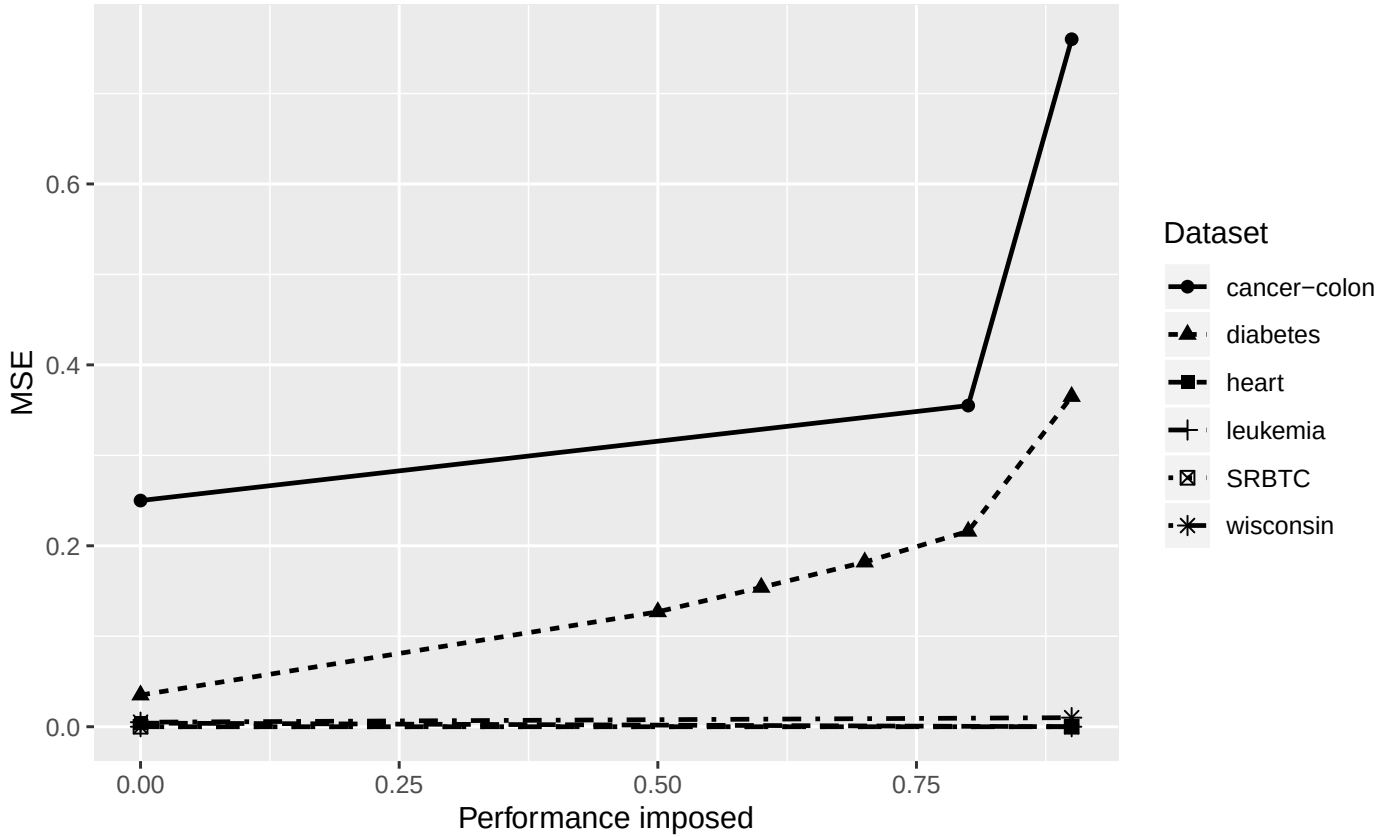
MSE (linear SVM)

Dataset	*Bootstrapped*	*Sollich*	*Platt*	*Tao et al.* ($r = 0, \sqrt{10}, \sqrt{20}, \sqrt{30}$)
`wisconsin`	**0.003**	0.064	0.021	0.028, 0.019, 0.034, 0.055
`cancer-colon`	**0.201**	0.241	0.252	0.208, 0.208, 0.208, 0.208
`diabetes`	0.192	0.190	**0.157**	0.229, 0.233, 0.234, 0.234
`leukemia`	**0**	0.239	0.01	0.029, 0.029, 0.029, 0.029
`SRBTC`	0.01	0.237	0.011	**0**, **0**, **0**, **0**
`heart`	0.143	0.158	**0.121**	0.15, 0.118, 0.207, 0.218
`adult`	0.144	0.128	**0.068**	0.148, 0.078, 0.142, 0.167

MSE for the positive class probability predictions



MSE for the negative class probability predictions



(Sparse and cost-sensitive) linear models using MINLO

$$\min_{\alpha \in \mathcal{A}} \sum_{i=1}^{m} \left(Y_i - \sum_{j=1}^{N} \alpha_j X_{ij} \right)^2$$

$\mathcal{A}$ modelling, among other things, which features are selected

(Sparse and cost-sensitive) linear models using MINLO

$$\min_{\alpha \in \mathcal{A}} \sum_{i=1}^{m} \left(Y_i - \sum_{j=1}^{N} \alpha_j X_{ij} \right)^2$$

$\mathcal{A}$ modelling, among other things, which features are selected


Bertsimas and King, *Operations Research*, 2015


Bertsimas, King and Mazumder, *Annals of Statistics*, 2016


C., Olivares-Nadal, Ramírez-Cobo, *Biostatistics*, 2017

Sparsity in linear models via convex optim

$$Y_i = \sum_{j=1}^{N} \alpha_j X_{ij} + e_i \qquad i = 1, \ldots, m$$

Sparsity in linear models via convex optim

$$Y_i = \sum_{j=1}^{N} \alpha_j X_{ij} + e_i \qquad i = 1, \ldots, m$$

$$\min_{\alpha} \sum_{i=1}^{m} \left(Y_i - \sum_{j=1}^{N} \alpha_j X_{ij} \right)^2$$

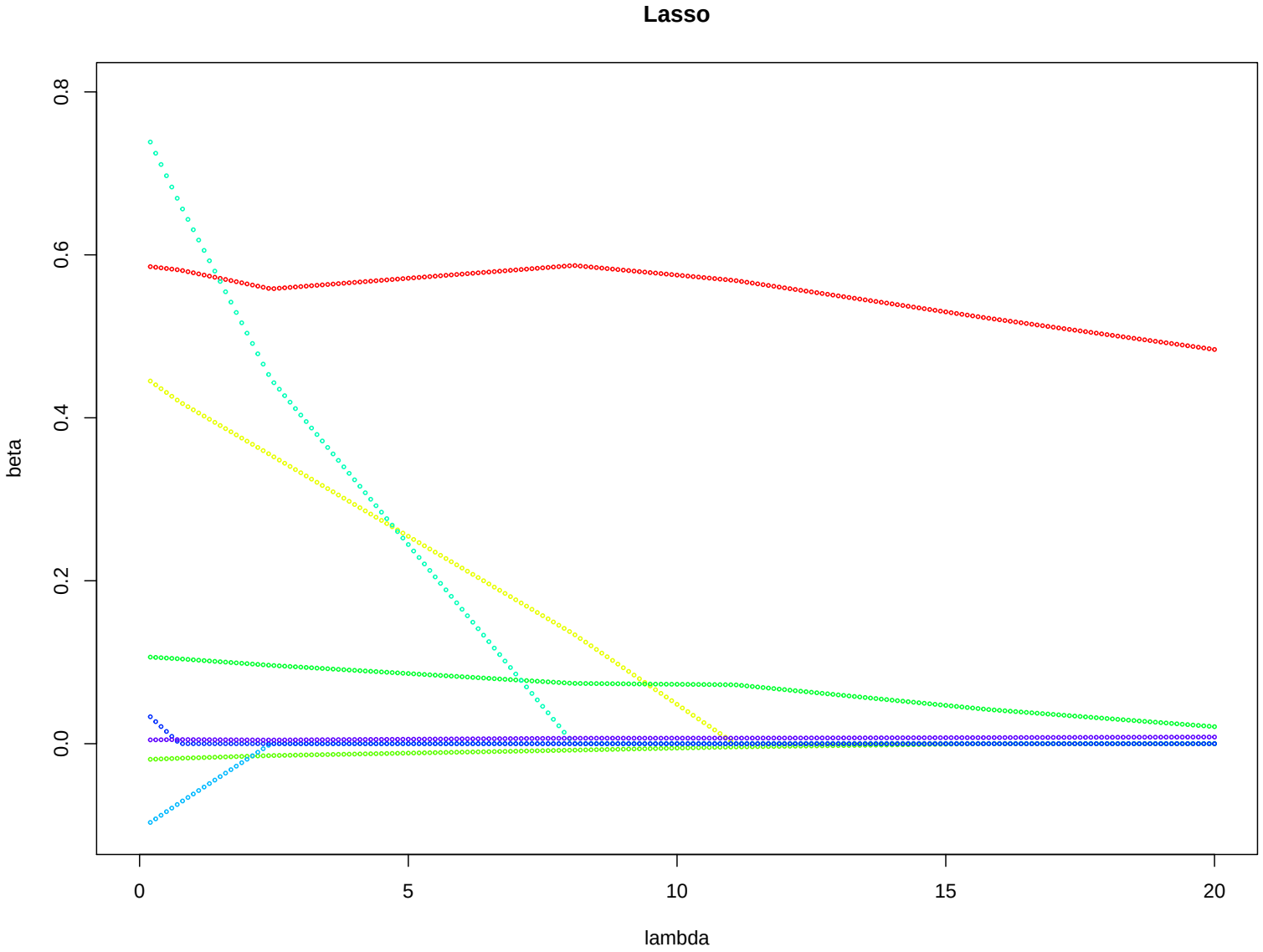
Sparsity in linear models via convex optim

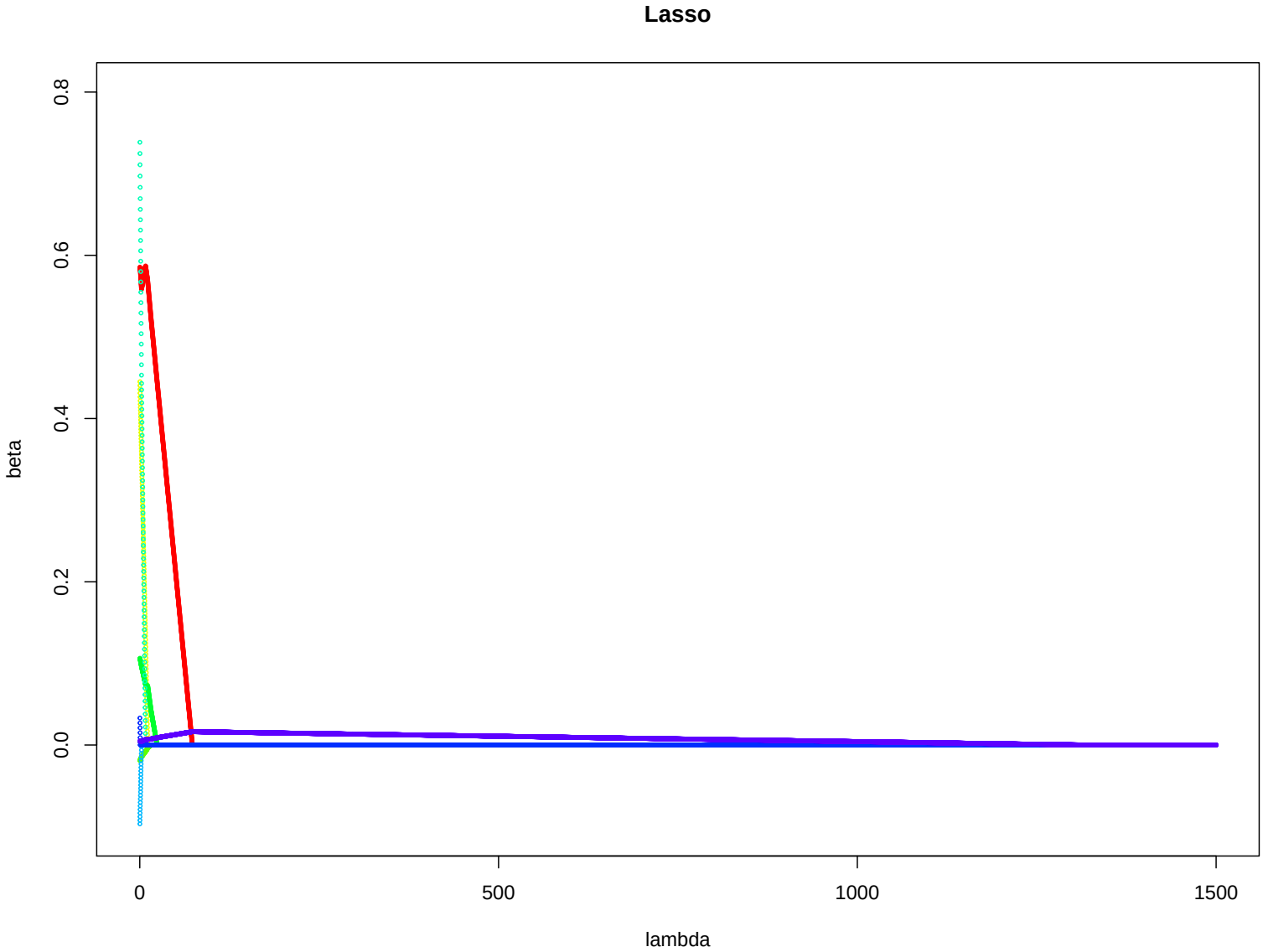
$$Y_i = \sum_{j=1}^{N} \alpha_j X_{ij} + e_i \qquad i = 1, \ldots, m \qquad\qquad \min_{\alpha} \sum_{i=1}^{m} \left(Y_i - \sum_{j=1}^{N} \alpha_j X_{ij} \right)^2$$

Making the model sparse. The lasso

$$\min_{\alpha} \sum_{i=1}^{m} \left(Y_i - \sum_{j=1}^{N} \alpha_j X_{ij} \right)^2 + \lambda \|\alpha\|_1$$

- R. Tibshirani, "Regression shrinkage and selection via the lasso", *J. of the Royal Statistical Society - B* , 1996
- $\approx$ 27.500 cites in Scholar


Lasso
beta
0.0
0.2
0.4
0.6
0.8
0
5
10
15
20
lambda


Lasso
beta
0.0
0.2
0.4
0.6
0.8
0
500
1000
1500
lambda

$$\min_{\alpha} \sum_{i=1}^{m} \left(Y_i - \sum_{j=1}^{N} \alpha_j X_{ij} \right)^2 + \lambda \|\alpha\|_1$$

$$\min_{\alpha} \sum_{i=1}^{m} \left(Y_i - \sum_{j=1}^{N} \alpha_j X_{ij} \right)^2 + \lambda \|\alpha\|_1$$

- Lasso and relatives implemented in several packages in `R` (e.g. `lars`, `elasticnet`, ...) and Python (`scikit-learn`)

$$\min_{\alpha} \sum_{i=1}^{m} \left(Y_i - \sum_{j=1}^{N} \alpha_j X_{ij} \right)^2 + \lambda \|\alpha\|_1$$

- Lasso and relatives implemented in several packages in `R` (e.g. `lars`, `elasticnet`, ...) and Python (`scikit-learn`)
- Records treated homogeneously. No control of errors on subpopulations, in case of heterogeneous data

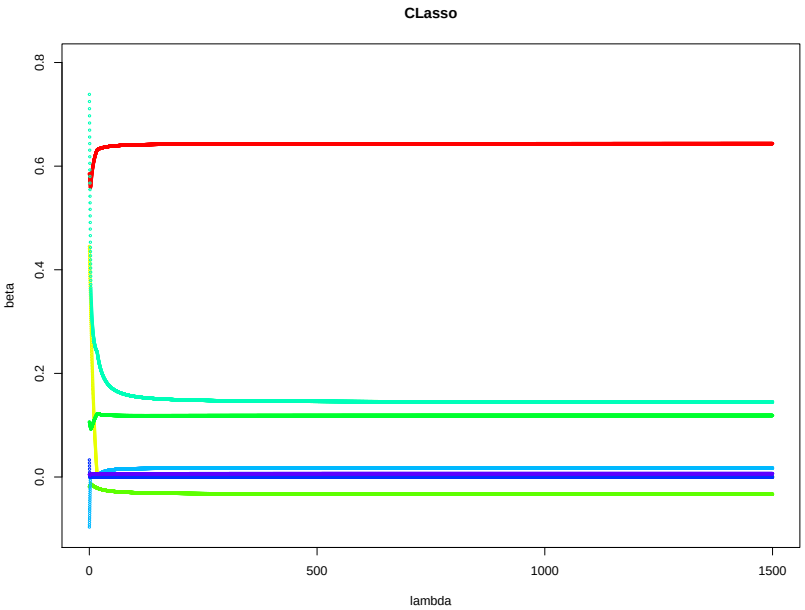
$$\min_{\alpha} \sum_{i=1}^{m} \left(Y_i - \sum_{j=1}^{N} \alpha_j X_{ij} \right)^2 + \lambda \|\alpha\|_1$$

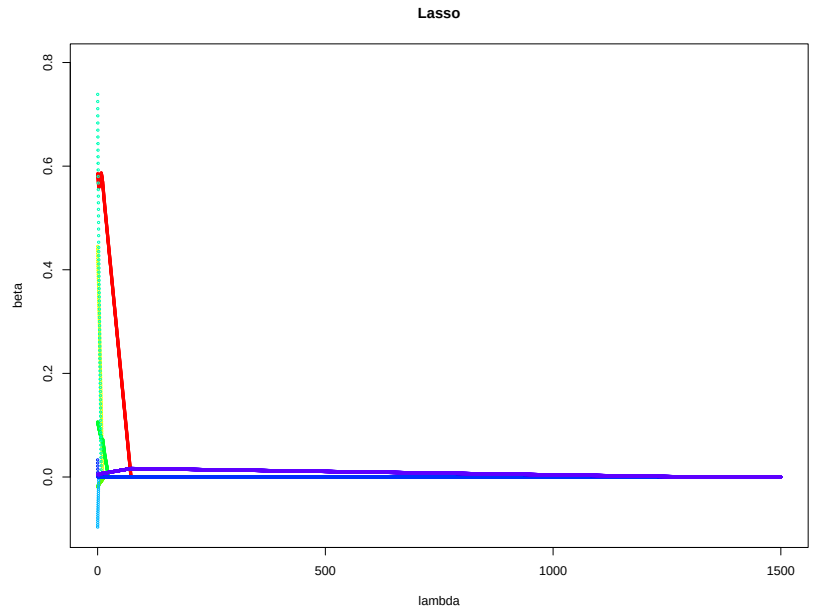
- Lasso and relatives implemented in several packages in `R` (e.g. `lars`, `elasticnet`, ...) and Python (`scikit-learn`)
- Records treated homogeneously. No control of errors on subpopulations, in case of heterogeneous data
- New Mathematical Optimization problem:

$$\begin{array}{ll} \min_{\alpha} & \sum_{i=1}^{m} \left(Y_i - \sum_{j=1}^{N} \alpha_j X_{ij} \right)^2 + \lambda \|\alpha\|_1 \\ \text{s.t.} & \sum_{i \in S_h} \left(Y_i - \sum_{j=1}^{N} \alpha_j X_{ij} \right)^2 \leq (1 + \tau_h) SSE_h \qquad \forall h \end{array}$$

$$\min_{\alpha} \quad \sum_{i=1}^{m} \left(Y_i - \sum_{j=1}^{N} \alpha_j X_{ij} \right)^2 + \lambda \|\alpha\|_1$$

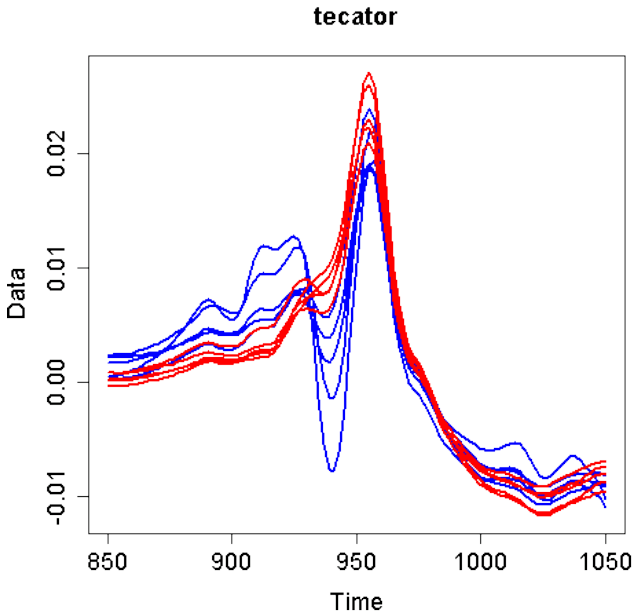
$$\text{s.t.} \quad \sum_{i \in S_h} \left(Y_i - \sum_{j=1}^{N} \alpha_j X_{ij} \right)^2 \leq (1 + \tau_h) SSE_h \qquad \forall h$$





. . . but we don't know how to (easily) build the path

Classification with functional data



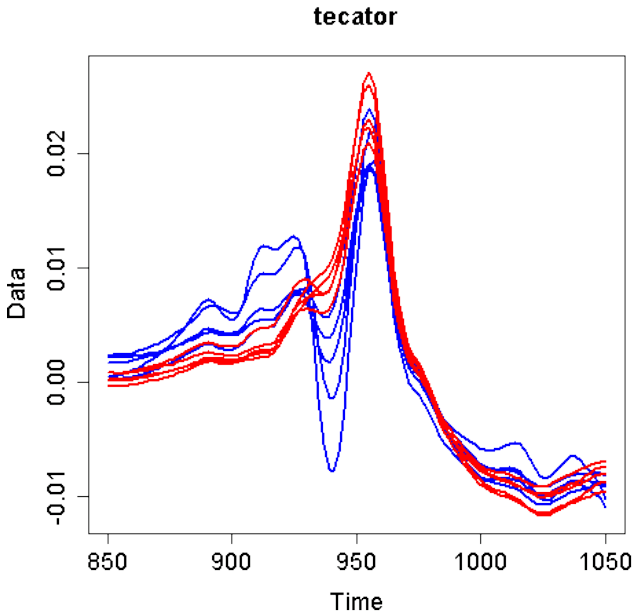
- $\mathbf{x} \in \mathcal{C}^0([0,T])$

Ferraty and Vieu. *Nonparametric functional data analysis: theory and practice*, 2006.

Ramsay and Silverman. *Functional data analysis*, 2006.

Febrero-Bande and Oviedo de la Fuente. "Statistical computing in functional data analysis: the r package fda.usc". *Journal of Statistical Software*, 2012.

Classification with functional data



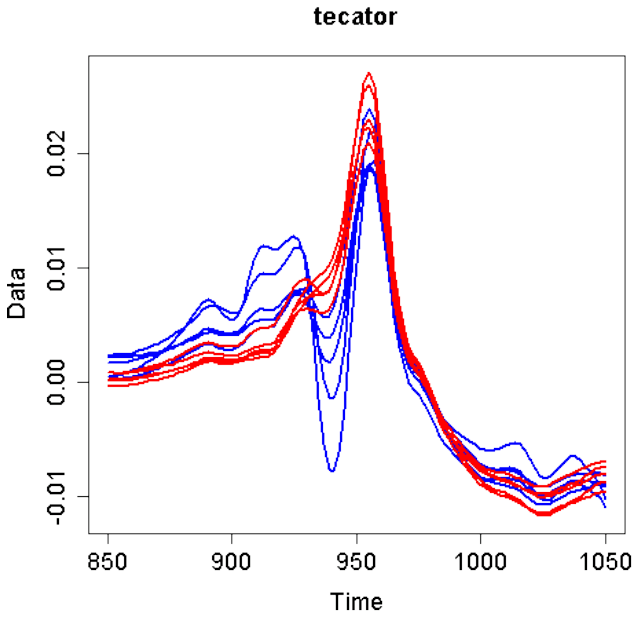
- $\mathbf{x} \in \mathcal{C}^0([0,T])$
- $\mathbf{x} \approx (\mathbf{x}(t_1), \ldots, \mathbf{x}(t_m)) \in \mathbb{R}^m$

Ferraty and Vieu. *Nonparametric functional data analysis: theory and practice*, 2006.

Ramsay and Silverman. *Functional data analysis*, 2006.

Febrero-Bande and Oviedo de la Fuente. "Statistical computing in functional data analysis: the r package fda.usc". *Journal of Statistical Software*, 2012.

Classification with functional data



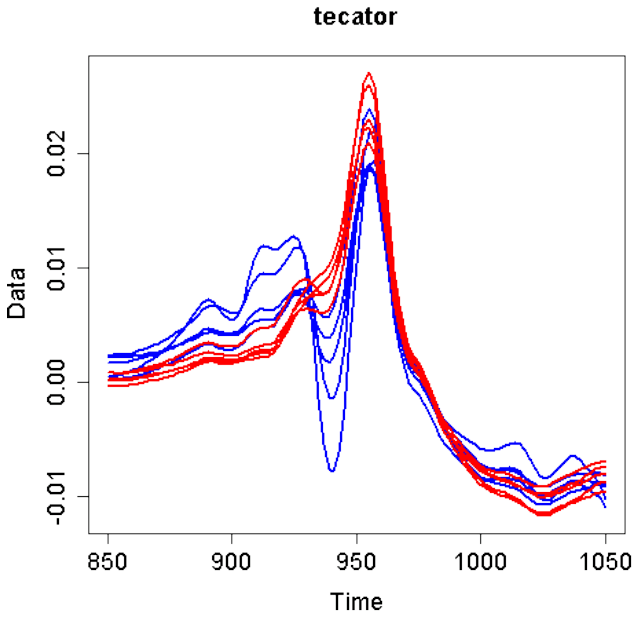
- $\mathbf{x} \in \mathcal{C}^0([0,T])$
- $\mathbf{x} \approx (\mathbf{x}(t_1), \ldots, \mathbf{x}(t_m)) \in \mathbb{R}^m$
- $(\mathbf{x}(t_1), \ldots, \mathbf{x}(t_m)) \approx \mathbf{x} \in \mathcal{C}^0([0,T])$

Ferraty and Vieu. *Nonparametric functional data analysis: theory and practice*, 2006.

Ramsay and Silverman. *Functional data analysis*, 2006.

Febrero-Bande and Oviedo de la Fuente. "Statistical computing in functional data analysis: the r package fda.usc". *Journal of Statistical Software*, 2012.

Classification with functional data



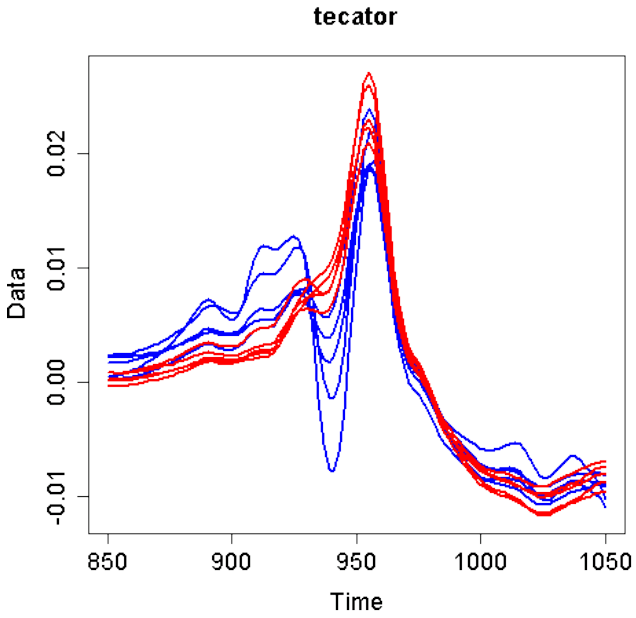
- $\mathbf{x} \in \mathcal{C}^0([0,T])$
- $\mathbf{x} \approx (\mathbf{x}(t_1), \ldots, \mathbf{x}(t_m)) \in \mathbb{R}^m$
- $(\mathbf{x}(t_1), \ldots, \mathbf{x}(t_m)) \approx \mathbf{x} \in \mathcal{C}^0([0,T]) \approx (\mathbf{x}(t_1), \ldots, \mathbf{x}(t_m))$

Ferraty and Vieu. *Nonparametric functional data analysis: theory and practice*, 2006.

Ramsay and Silverman. *Functional data analysis*, 2006.

Febrero-Bande and Oviedo de la Fuente. "Statistical computing in functional data analysis: the r package fda.usc". *Journal of Statistical Software*, 2012.

Classification with functional data

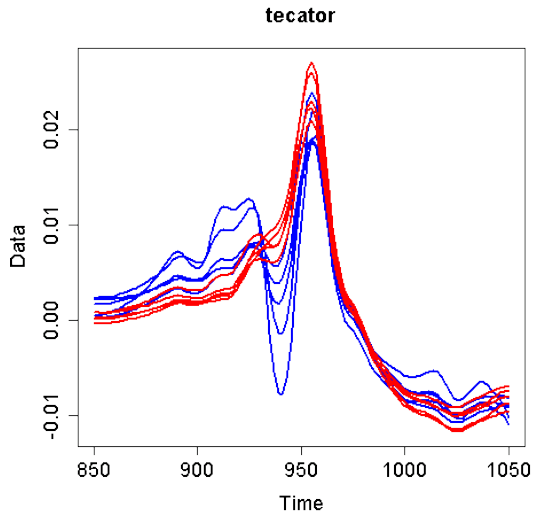


- $\mathbf{x} \in \mathcal{C}^0([0,T])$
- $\mathbf{x} \approx (\mathbf{x}(t_1), \ldots, \mathbf{x}(t_m)) \in \mathbb{R}^m$
- $(\mathbf{x}(t_1), \ldots, \mathbf{x}(t_m)) \approx \mathbf{x} \in \mathcal{C}^0([0,T]) \approx (\mathbf{x}(t_1), \ldots, \mathbf{x}(t_m))$

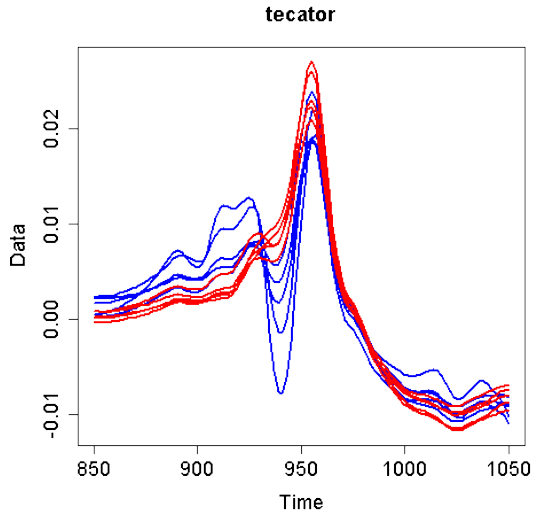
Muñoz and González. "Representing functional data using support vector machines". *Pattern Recognition Letters*, 2010.

Rossi and Villa. "Support vector machine for functional data classification". *Neurocomputing*, 2006.

Gaussian kernel for functional data (I)



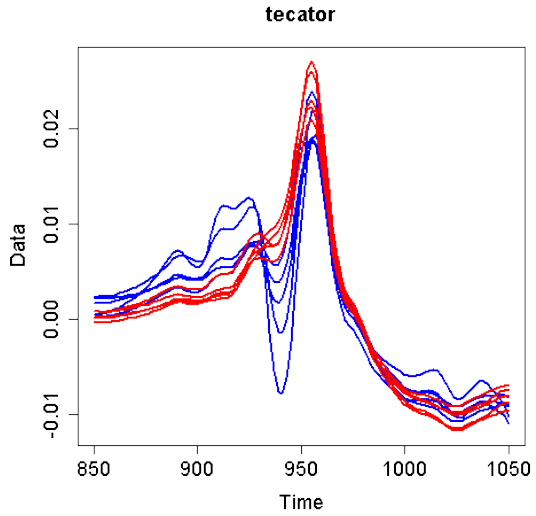
Gaussian kernel for functional data (I)



$$K(\mathbf{x}_i, \mathbf{x}_j) = e^{-\gamma \|\mathbf{x}_i - \mathbf{x}_j\|^2}$$

- $\|\mathbf{x}_i - \mathbf{x}_j\|^2 = \int_0^T (\mathbf{x}_i(t) - \mathbf{x}_j(t))^2 \, dt$

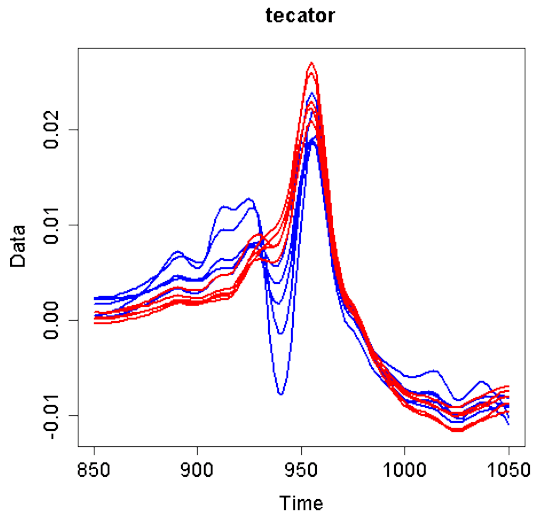
Gaussian kernel for functional data (I)



$$K(\mathbf{x}_i, \mathbf{x}_j) = e^{-\gamma \|\mathbf{x}_i - \mathbf{x}_j\|^2}$$

- $\|\mathbf{x}_i - \mathbf{x}_j\|^2 = \int_0^T (\mathbf{x}_i(t) - \mathbf{x}_j(t))^2 \, dt \approx \sum_k \eta_k \, (\mathbf{x}_i(t_k) - \mathbf{x}_j(t_k))^2$

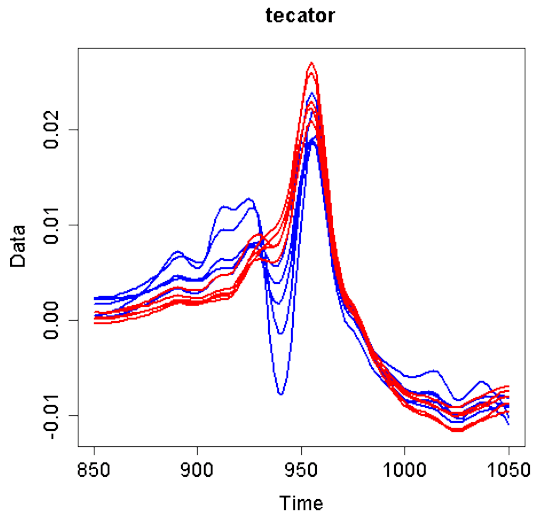
Gaussian kernel for functional data (I)



$$K(\mathbf{x}_i, \mathbf{x}_j) = e^{-\gamma \|\mathbf{x}_i - \mathbf{x}_j\|^2}$$

- $\|\mathbf{x}_i - \mathbf{x}_j\|^2 = \int_0^T (\mathbf{x}_i(t) - \mathbf{x}_j(t))^2 \, dt \approx \sum_k \eta_k \, (\mathbf{x}_i(t_k) - \mathbf{x}_j(t_k))^2$
- $K(\mathbf{x}_i, \mathbf{x}_j) = e^{-\int_0^T \gamma (\mathbf{x}_i(t) - \mathbf{x}_j(t))^2 \, dt} \approx e^{-\sum_k \gamma \eta_k (\mathbf{x}_i(t_k) - \mathbf{x}_j(t_k))^2}$

Gaussian kernel for functional data (I)



$$K(\mathbf{x}_i, \mathbf{x}_j) = e^{-\gamma \|\mathbf{x}_i - \mathbf{x}_j\|^2}$$

- $\|\mathbf{x}_i - \mathbf{x}_j\|^2 = \int_0^T (\mathbf{x}_i(t) - \mathbf{x}_j(t))^2 \, dt \approx \sum_k \eta_k \, (\mathbf{x}_i(t_k) - \mathbf{x}_j(t_k))^2$
- $K(\mathbf{x}_i, \mathbf{x}_j) = e^{-\int_0^T \gamma (\mathbf{x}_i(t) - \mathbf{x}_j(t))^2 \, dt} \approx e^{-\sum_k \gamma \eta_k (\mathbf{x}_i(t_k) - \mathbf{x}_j(t_k))^2}$

Gaussian kernel with functional bandwidth

$$K(\mathbf{x}_i, \mathbf{x}_j) = e^{-\int_0^T \gamma(t) (\mathbf{x}_i(t) - \mathbf{x}_j(t))^2 \, dt}$$

Gaussian kernel with functional bandwidth

$$K(\mathbf{x}_i, \mathbf{x}_j) = e^{-\int_0^T \gamma(t)(\mathbf{x}_i(t)-\mathbf{x}_j(t))^2\, dt}$$

A possible model for γ

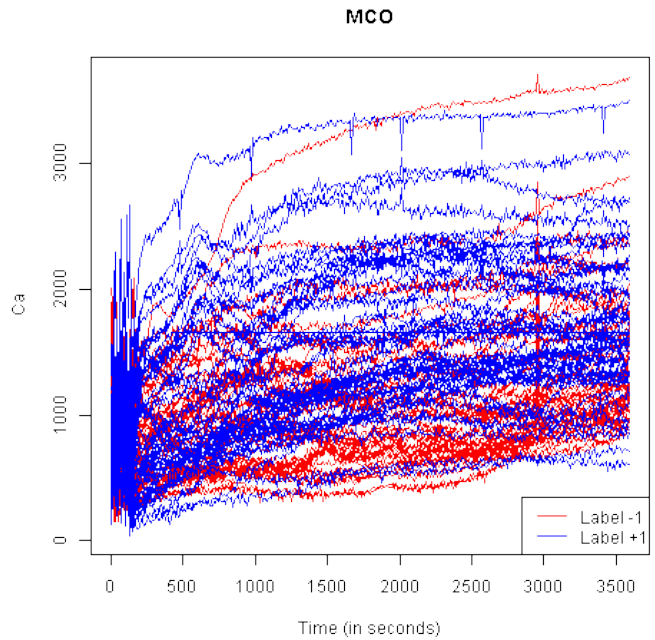
$$\gamma(t) = \begin{cases} \gamma_1, & \text{if } 0 \leq t \leq \tau_1 \\ \gamma_2, & \text{if } \tau_1 < t \leq \tau_2 \\ \dots & \dots \\ \gamma_H, & \text{if } \tau_{H-1} < t \leq T \end{cases}$$

- $\gamma_1, \dots, \gamma_H \geq 0$
- $0 \leq \tau_1 \leq \dots \leq \tau_{H-1} \leq T$

Blanquero, C., Jiménez-Cordero, Martín-Barragán. Functional-bandwidth kernel for Support Vector Machine with Functional Data: An alternating optimization algorithm. EJOR, 2019

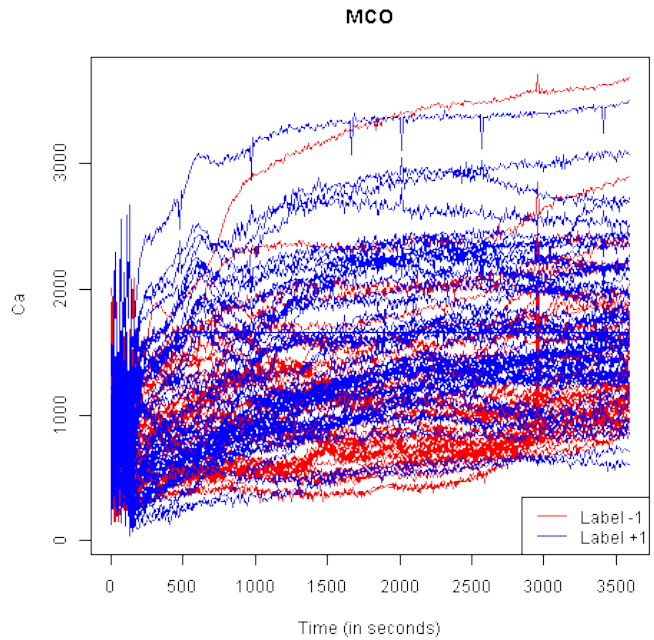
An example: Mitochondrial calcium data set

- 360 time instants in $[0, T]$, $T = 3590$
- 44 mice in treatment (+1), 45 control (-1)



An example: Mitochondrial calcium data set

- 360 time instants in $[0, T]$, $T = 3590$
- 44 mice in treatment (+1), 45 control (-1)



Out of sample accuracy estimates

$$\gamma(t) = \gamma \qquad \gamma(t) = \begin{cases} \gamma_1, & \text{if } 0 \le t \le \tau_1 \\ \gamma_2, & \text{if } \tau_1 < t \le T \end{cases}$$

	-1	$+1$	-1	$+1$
-1 :	37.55%	10.96%	42.58%	7.56%
$+1$	12.09%	35.61%	7.23%	42.95%

Parameters tuning (basic gaussian kernel)

$$\begin{array}{lll} \max_\lambda & \sum_{i\in I}\lambda_i - \frac{1}{2}\sum_{ij}\lambda_i y_i \lambda_j y_j e^{-\gamma\|\mathbf{x}_i-\mathbf{x}_j\|^2} & \\ \text{s.t.} & \sum_{i\in I}\lambda_i y_i = 0 & (P_{I,C,\gamma}) \\ & 0 \le \lambda_i \le \frac{C}{2} \qquad\qquad i \in I & \end{array}$$

C, γ: k-fold crossvalidation

- I : split in k blocks of similar size, $I_1, \ldots, I_k$
- for each pair C, γ in a grid (e.g. $2^{-12} \ldots, 2^{12}$), estimate $acc(C, \gamma)$:
  - for each $i = 1, \ldots, k$
    - solve $(P_{I\setminus I_i,C,\gamma})$, yielding λ^i, β (via KKT)
    - calculate $acc_i(C, \gamma)$, fraction of correctly classified in I_i if classifier with λ^i, β were used
  - $acc(C, \gamma) = \frac{1}{k}\sum_{i=1}^{k} acc_i(C, \gamma)$

Parameters tuning (basic gaussian kernel)

$$\begin{array}{lll} \max_\lambda & \sum_{i\in I}\lambda_i - \frac{1}{2}\sum_{ij}\lambda_i y_i \lambda_j y_j e^{-\gamma\|\mathbf{x}_i-\mathbf{x}_j\|^2} & \\ \text{s.t.} & \sum_{i\in I}\lambda_i y_i = 0 & (P_{I,C,\gamma}) \\ & 0 \le \lambda_i \le \frac{C}{2} \qquad\qquad i \in I & \end{array}$$

C, γ: k-fold crossvalidation

- I : split in k blocks of similar size, $I_1, \ldots, I_k$
- for each pair C, γ in a grid (e.g. $2^{-12} \ldots, 2^{12}$), estimate $acc(C, \gamma)$:
  - for each $i = 1, \ldots, k$
    - solve $(P_{I\setminus I_i,C,\gamma})$, yielding λ^i, β (via KKT)
    - calculate $acc_i(C, \gamma)$, fraction of correctly classified in I_i if classifier with λ^i, β were used
  - $acc(C, \gamma) = \frac{1}{k}\sum_{i=1}^{k} acc_i(C, \gamma)$

Unfeasible for functional bandwidth kernel!!!

Parameters tuning (functional bandwidth kernel)

$$\theta = (\gamma_1, \ldots, \gamma_H | \tau_1, \ldots, \tau_{H-1})$$

Parameters tuning (functional bandwidth kernel)

$$\theta = (\gamma_1, \ldots, \gamma_H | \tau_1, \ldots, \tau_{H-1})$$

$$\widehat{y}_{I,C,\theta}(\mathbf{x}) = \sum_{i \in I} y_i \lambda_i^C K_\theta(\mathbf{x}, \mathbf{x}_i) + \beta^C$$

Parameters tuning (functional bandwidth kernel)

$$\theta = (\gamma_1, \ldots, \gamma_H | \tau_1, \ldots, \tau_{H-1})$$

$$\widehat{y}_{I,C,\theta}(\mathbf{x}) = \sum_{i \in I} y_i \lambda_i^C K_\theta(\mathbf{x}, \mathbf{x}_i) + \beta^C$$

$$\begin{array}{lll} \max_\lambda & \sum_{i \in I} \lambda_i - \frac{1}{2} \sum_{ij} \lambda_i y_i \lambda_j y_j K_\theta(\mathbf{x}_i, \mathbf{x}_j) & \\ \text{s.t.} & \sum_{i \in I} \lambda_i y_i = 0 & \\ & 0 \leq \lambda_i \leq \frac{C}{2} & i \in I \end{array} \qquad (P_{I,C,\theta})$$

Parameters tuning (functional bandwidth kernel)

$$\theta = (\gamma_1, \ldots, \gamma_H | \tau_1, \ldots, \tau_{H-1}) \qquad \widehat{y}_{I,C,\theta}(\mathbf{x}) = \sum_{i \in I} y_i \lambda_i^C K_\theta(\mathbf{x}, \mathbf{x}_i) + \beta^C$$

$$\begin{array}{lll} \max_\lambda & \sum_{i \in I} \lambda_i - \frac{1}{2} \sum_{ij} \lambda_i y_i \lambda_j y_j K_\theta(\mathbf{x}_i, \mathbf{x}_j) & \\ \text{s.t.} & \sum_{i \in I} \lambda_i y_i = 0 & (P_{I,C,\theta}) \\ & 0 \leq \lambda_i \leq \frac{C}{2} & i \in I \end{array}$$

Randomly split sample I into I_1, I_2 and I_3.
for C in grid **do**
    Alternating Procedure
    repeat
        1. Fixed θ, find λ^C solving $(P_{I_1,C,\theta})$
        2. Fixed λ, find θ maximizing correlation of y and $\widehat{y}_{I,C,\theta}$ in I_2.
    until stopping criteria
end for
Return as C the one with best misclassification rate in I_3.
Return as λ and θ those associated with C

Parameters tuning (functional bandwidth kernel)

$$\theta = (\gamma_1, \ldots, \gamma_H | \tau_1, \ldots, \tau_{H-1}) \qquad \widehat{y}_{I,C,\theta}(\mathbf{x}) = \sum_{i \in I} y_i \lambda_i^C K_\theta(\mathbf{x}, \mathbf{x}_i) + \beta^C$$

$$\begin{array}{lll} \max_\lambda & \sum_{i \in I} \lambda_i - \frac{1}{2} \sum_{ij} \lambda_i y_i \lambda_j y_j K_\theta(\mathbf{x}_i, \mathbf{x}_j) & \\ \text{s.t.} & \sum_{i \in I} \lambda_i y_i = 0 & (P_{I,C,\theta}) \\ & 0 \leq \lambda_i \leq \frac{C}{2} & i \in I \end{array}$$

Randomly split sample I into I_1, I_2 and I_3.
for C in grid **do**
  Alternating Procedure
  repeat
    1. Fixed θ, find λ^C solving $(P_{I_1,C,\theta})$
    2. Fixed λ, find θ maximizing correlation of y and $\widehat{y}_{I,C,\theta}$ in I_2.
  until stopping criteria
end for
Return as C the one with best misclassification rate in I_3.
Return as λ and θ those associated with C

$$\widehat{y}_{I,C,\theta}(\mathbf{x}) = \sum_{i \in I} y_i \lambda_i^C K_\theta(\mathbf{x}, \mathbf{x}_i) + \beta$$

$$K_\theta(\mathbf{x}, \mathbf{x}_i) = e^{-\sum_{h=1}^{H} \int_{\tau_{h-1}}^{\tau_h} \gamma_h (\mathbf{x}(t) - \mathbf{x}_i(t))^2 \, dt}$$

$$\widehat{y}_{I,C,\theta}(\mathbf{x}) = \sum_{i \in I} y_i \lambda_i^C K_\theta(\mathbf{x}, \mathbf{x}_i) + \beta$$

$$K_\theta(\mathbf{x}, \mathbf{x}_i) = e^{-\sum_{h=1}^{H} \int_{\tau_{h-1}}^{\tau_h} \gamma_h (\mathbf{x}(t) - \mathbf{x}_i(t))^2 \, dt}$$

Smooth optimization problem

- chain rule
- $K : \mathcal{C}^1$ for $\mathbf{x} : \mathcal{C}^0$
- $K : \mathcal{C}^3$ for $\mathbf{x} : \mathcal{C}^2$ (as generated by cubic spline)

Parameters tuning (functional bandwidth kernel)
Improved

Model H

$$\gamma(t) = \begin{cases} \gamma_1, & \text{if } 0 \leq t \leq \tau_1 \\ \gamma_2, & \text{if } \tau_1 < t \leq \tau_2 \\ \dots & \dots \\ \gamma_H, & \text{if } \tau_{H-1} < t \leq T \end{cases}$$

Nested heuristic

C., Martín-Barragán, Romero Morales, *Computers & OR*, 2014

Parameters tuning (functional bandwidth kernel)
Improved

Model H

$$\gamma(t) = \begin{cases} \gamma_1, & \text{if } 0 \leq t \leq \tau_1 \\ \gamma_2, & \text{if } \tau_1 < t \leq \tau_2 \\ \dots & \dots \\ \gamma_H, & \text{if } \tau_{H-1} < t \leq T \end{cases}$$

Nested heuristic

C., Martín-Barragán, Romero Morales, *Computers & OR*, 2014

Model 1

$\gamma(t) = \gamma_1$

Parameters tuning (functional bandwidth kernel)
Improved

Model H

$$\gamma(t) = \begin{cases} \gamma_1, & \text{if } 0 \leq t \leq \tau_1 \\ \gamma_2, & \text{if } \tau_1 < t \leq \tau_2 \\ \dots & \dots \\ \gamma_H, & \text{if } \tau_{H-1} < t \leq T \end{cases}$$

Nested heuristic

C., Martín-Barragán, Romero Morales, *Computers & OR*, 2014

Model 1

$\gamma(t) = \gamma_1$

Model 2

$$\gamma(t) = \begin{cases} \gamma_1, & \text{if } 0 \leq t \leq \tau_1 \\ \gamma_2, & \text{if } \tau_1 < t \leq T \end{cases}$$

Parameters tuning (functional bandwidth kernel)
Improved

Model H

$$\gamma(t) = \begin{cases} \gamma_1, & \text{if } 0 \leq t \leq \tau_1 \\ \gamma_2, & \text{if } \tau_1 < t \leq \tau_2 \\ \dots & \dots \\ \gamma_H, & \text{if } \tau_{H-1} < t \leq T \end{cases}$$

Nested heuristic

C., Martín-Barragán, Romero Morales, *Computers & OR*, 2014

Model 1

$\gamma(t) = \gamma_1$

Model 2

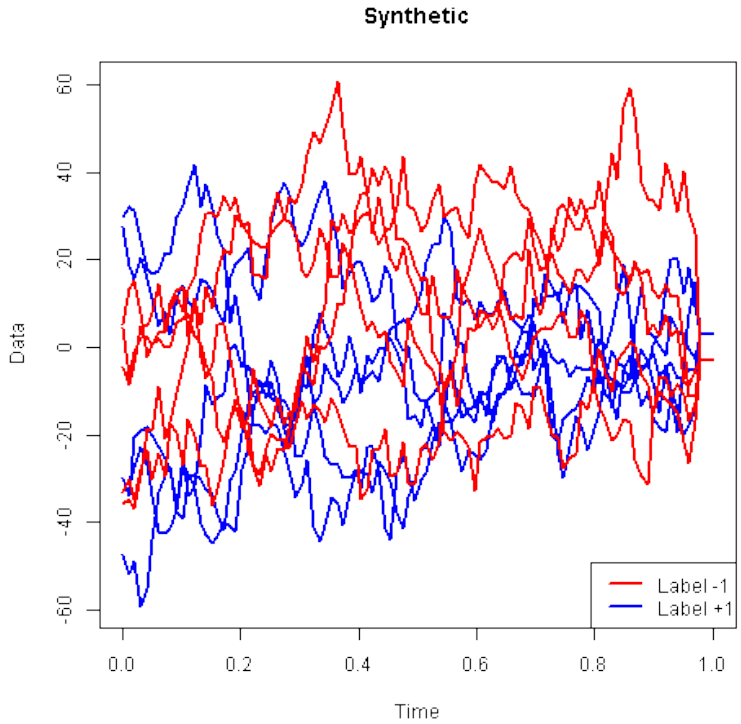
$$\gamma(t) = \begin{cases} \gamma_1, & \text{if } 0 \leq t \leq \tau_1 \\ \gamma_2, & \text{if } \tau_1 < t \leq T \end{cases}$$

Model 3

$$\gamma(t) = \begin{cases} \gamma_1, & \text{if } 0 \leq t \leq \tau_1 \\ \gamma_2, & \text{if } \tau_1 < t \leq \tau_2 \\ \gamma_3, & \text{if } \tau_2 < t \leq T \end{cases} \quad \dots$$

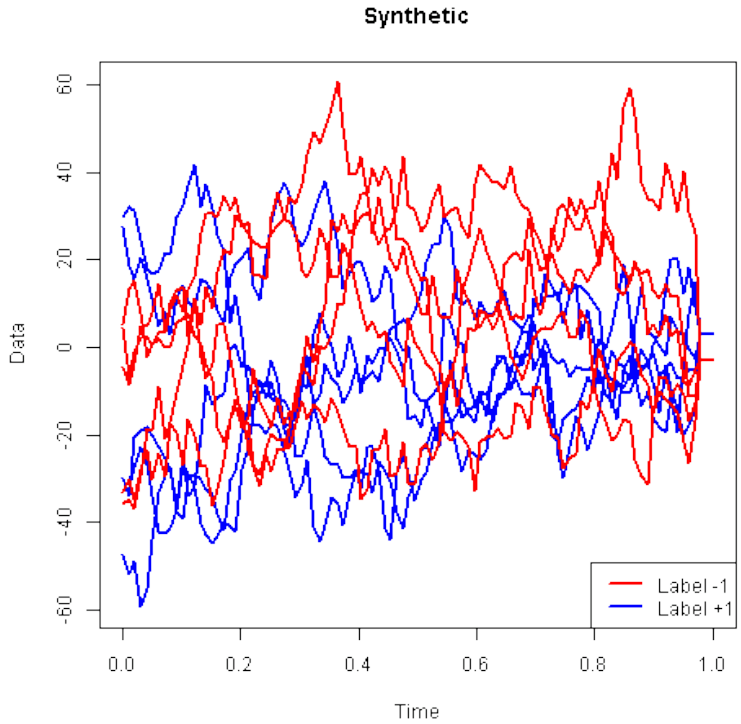
A test example

15,000 functions like these



A test example

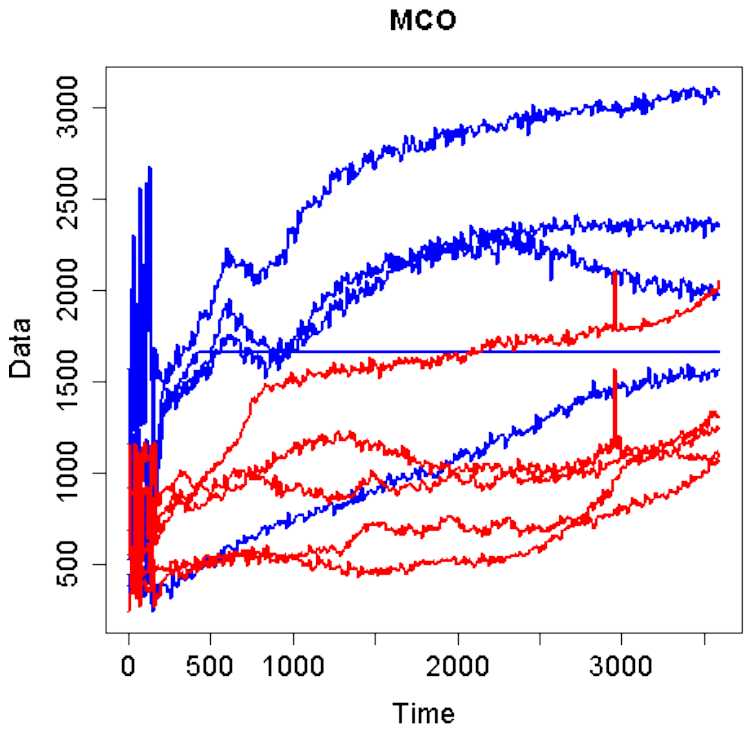
15,000 functions like these



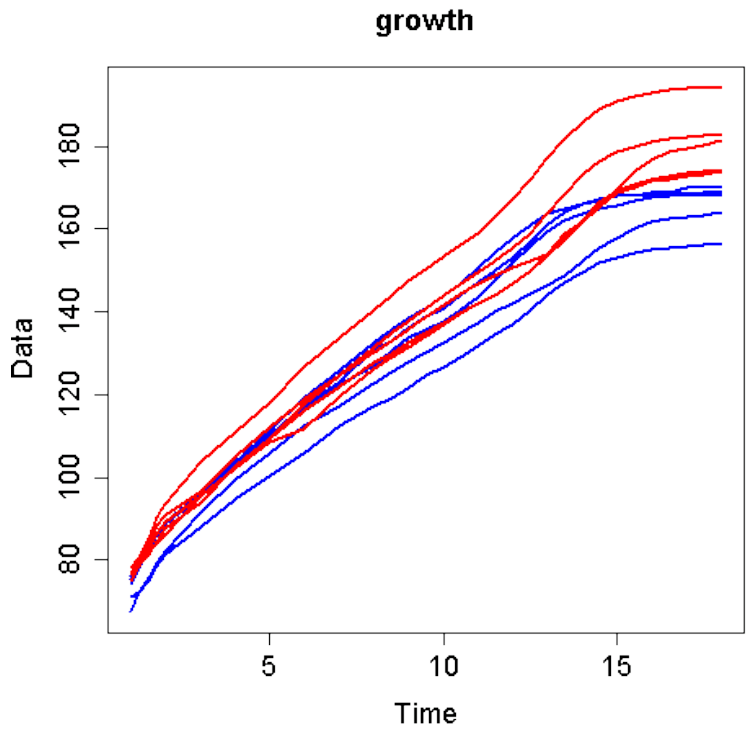
	1 (classic SVM)	$H = 2$	$H = 3$	$H = 4$
% misc	32.95	0	0	0

	#records	#time instants	#records label -1	#records label +1
MCO	89	360	44	45
growth	93	31	54	39
phoneme	200	150	100	100
rain	35	365	15	20
regions	35	365	20	15
tecator	215	100	77	138

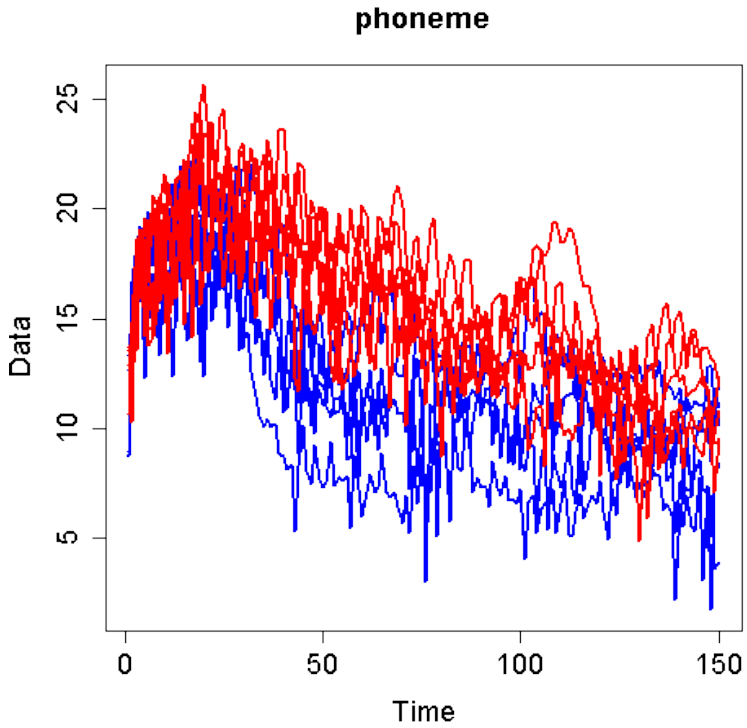
	#records	#time instants	#records label -1	#records label +1
MCO	89	360	44	45
growth	93	31	54	39
phoneme	200	150	100	100
rain	35	365	15	20
regions	35	365	20	15
tecator	215	100	77	138



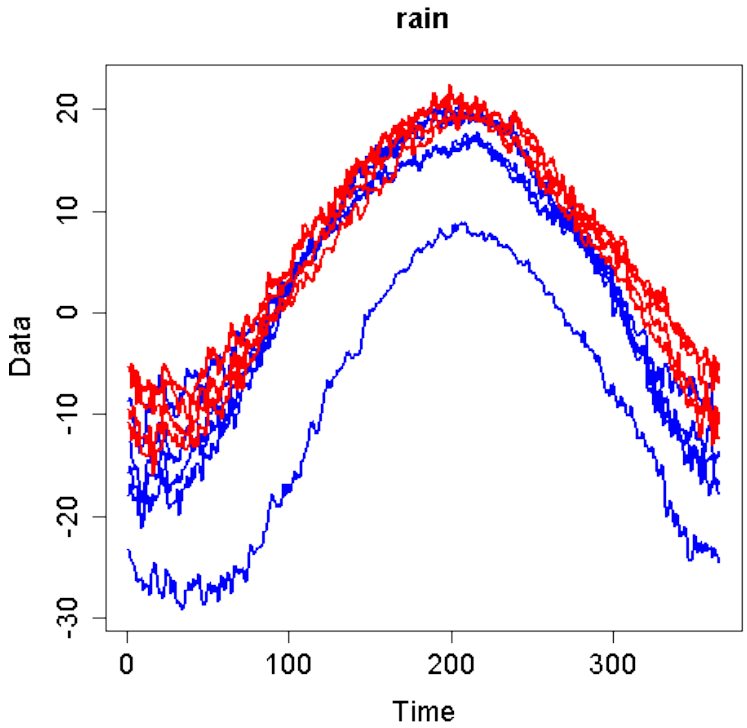
	#records	#time instants	#records label -1	#records label +1
MCO	89	360	44	45
growth	93	31	54	39
phoneme	200	150	100	100
rain	35	365	15	20
regions	35	365	20	15
tecator	215	100	77	138



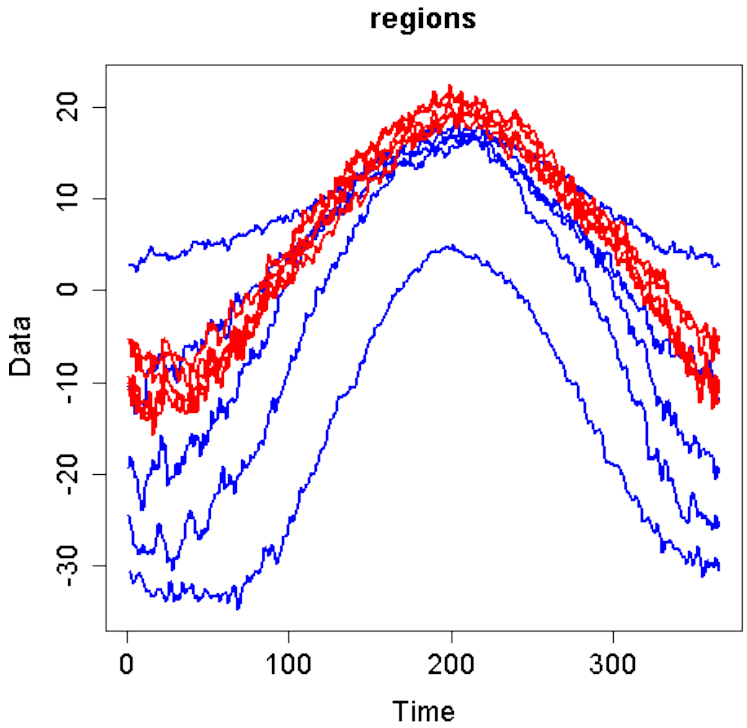
	#records	#time instants	#records label -1	#records label +1
MCO	89	360	44	45
growth	93	31	54	39
phoneme	200	150	100	100
rain	35	365	15	20
regions	35	365	20	15
tecator	215	100	77	138



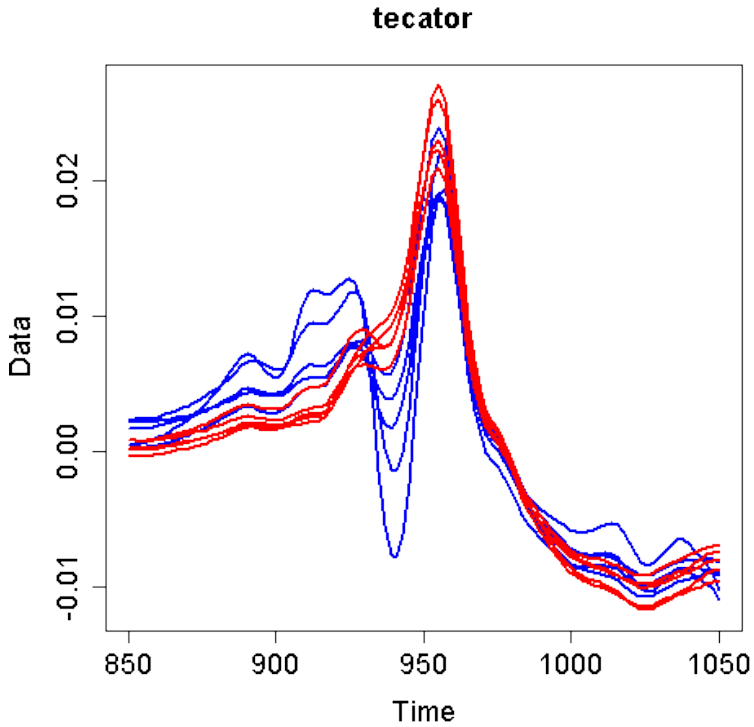
	#records	#time instants	#records label -1	#records label +1
MCO	89	360	44	45
growth	93	31	54	39
phoneme	200	150	100	100
rain	35	365	15	20
regions	35	365	20	15
tecator	215	100	77	138



	#records	#time instants	#records label -1	#records label +1
MCO	89	360	44	45
growth	93	31	54	39
phoneme	200	150	100	100
rain	35	365	15	20
regions	35	365	20	15
tecator	215	100	77	138



	#records	#time instants	#records label -1	#records label +1
MCO	89	360	44	45
growth	93	31	54	39
phoneme	200	150	100	100
rain	35	365	15	20
regions	35	365	20	15
tecator	215	100	77	138

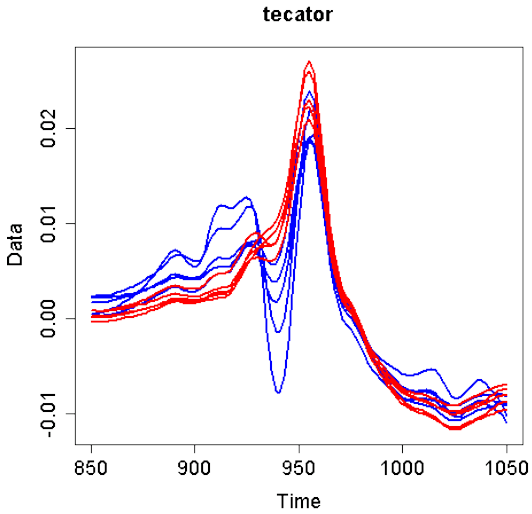


	#records	#time instants	#records label -1	#records label +1
MCO	89	360	44	45
growth	93	31	54	39
phoneme	200	150	100	100
rain	35	365	15	20
regions	35	365	20	15
tecator	215	100	77	138

% misclassification rate (out-of-sample)

	$H = 1$	$H = 2$	$H = 3$	$H = 4$
MCO	20.80	14.73	11.05	10.37
growth	5.64	4.67	4.35	4.19
phoneme	19.88	18.08	17.63	17.11
rain	28.40	22.84	22.42	21.59
regions	19.46	16.43	16.02	16.51
tecator	3.47	2.92	2.64	2.29

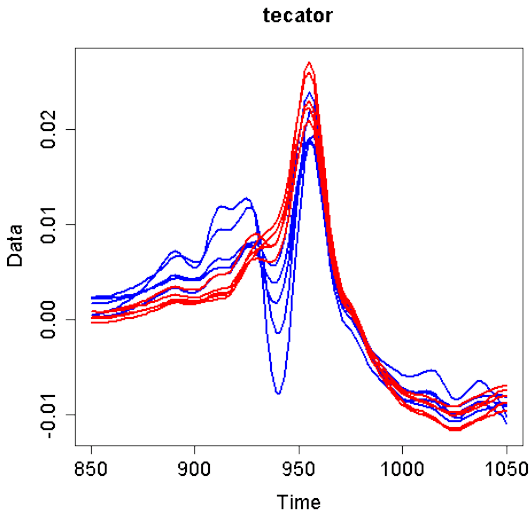
Gaussian kernel for functional data (II)



$$K(\mathbf{x}_i, \mathbf{x}_j) = e^{-\gamma \|\mathbf{x}_i - \mathbf{x}_j\|^2}$$

- $K(\mathbf{x}_i, \mathbf{x}_j) = e^{-\int_0^T \gamma (\mathbf{x}_i(t) - \mathbf{x}_j(t))^2 \, dt}$

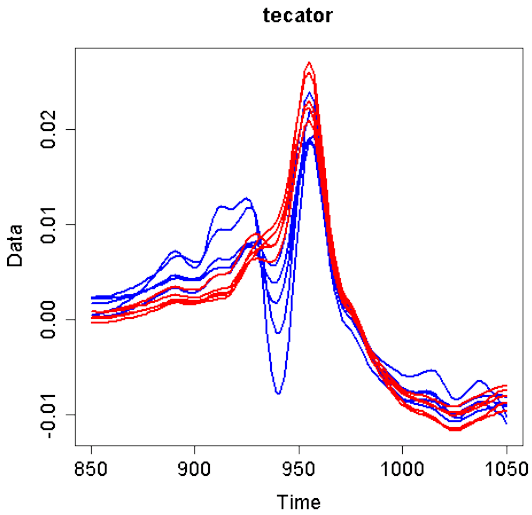
Gaussian kernel for functional data (II)



$$K(\mathbf{x}_i, \mathbf{x}_j) = e^{-\gamma \|\mathbf{x}_i - \mathbf{x}_j\|^2}$$

- $K(\mathbf{x}_i, \mathbf{x}_j) = e^{-\int_0^T \gamma (\mathbf{x}_i(t) - \mathbf{x}_j(t))^2 \, dt}$
- $K(\mathbf{x}_i, \mathbf{x}_j) = e^{-\int_0^T \gamma(t) (\mathbf{x}_i(t) - \mathbf{x}_j(t))^2 \, dt}$

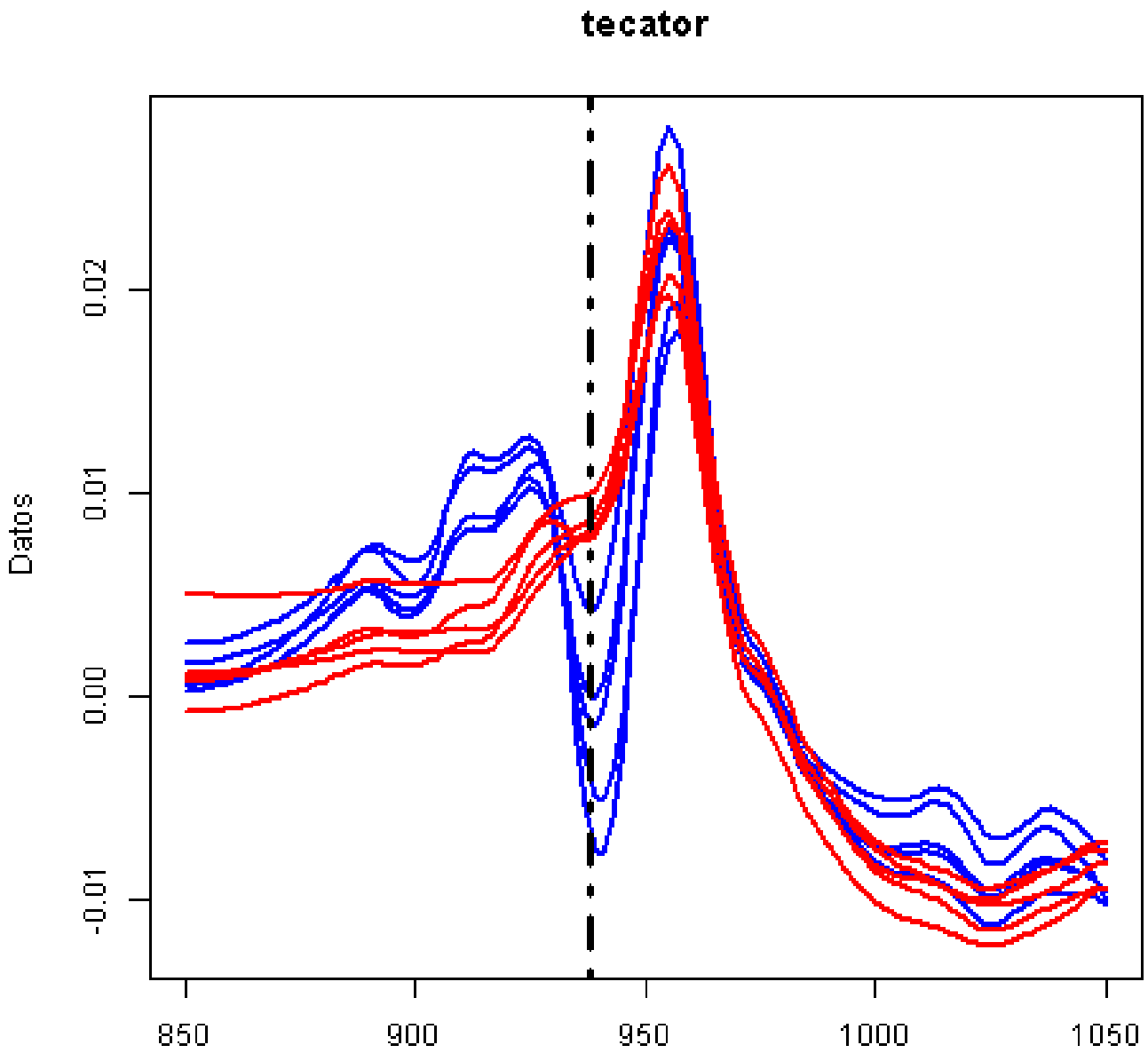
Gaussian kernel for functional data (II)



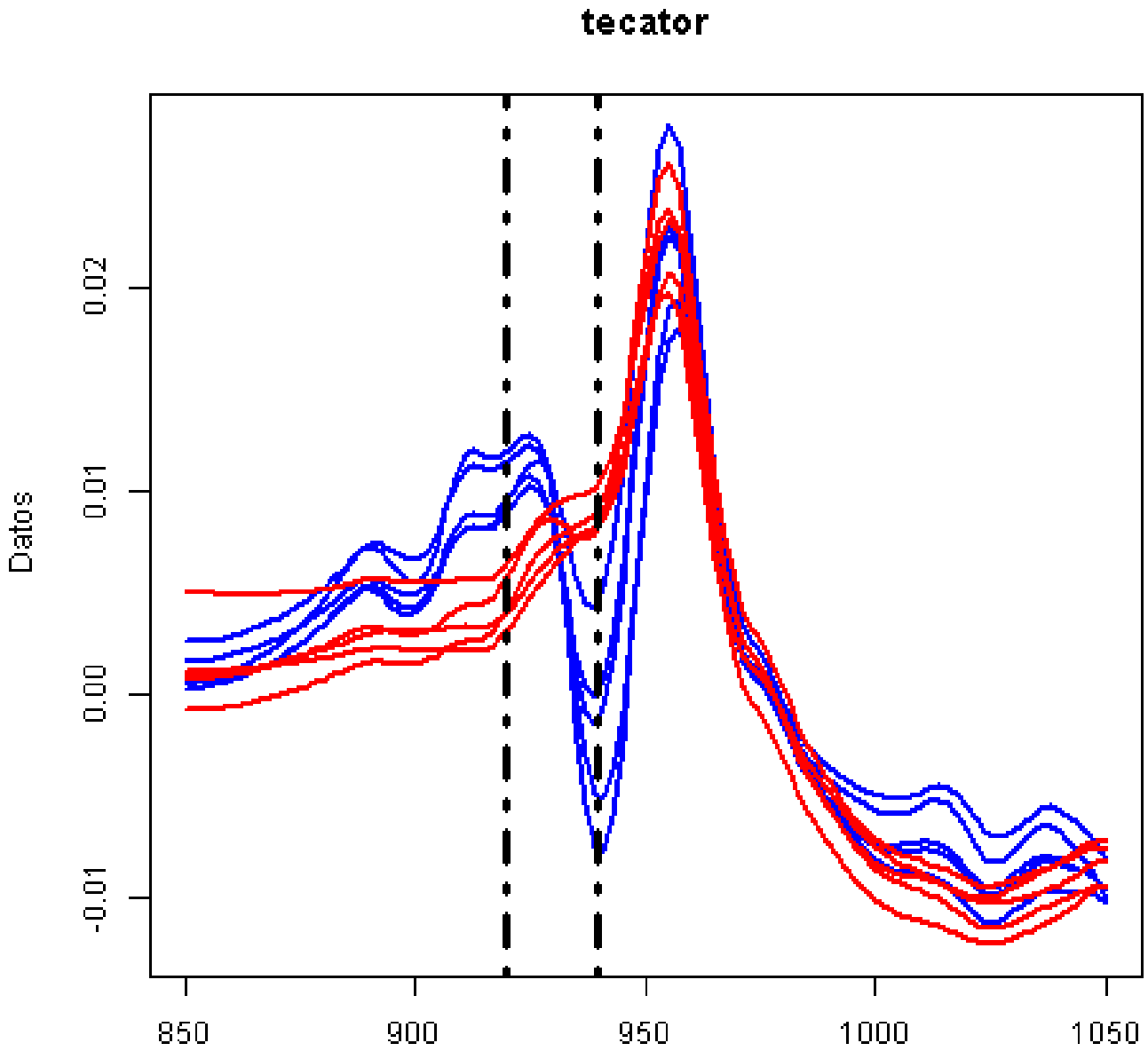
$$K(\mathbf{x}_i, \mathbf{x}_j) = e^{-\gamma \|\mathbf{x}_i - \mathbf{x}_j\|^2}$$

- $K(\mathbf{x}_i, \mathbf{x}_j) = e^{-\int_0^T \gamma (\mathbf{x}_i(t) - \mathbf{x}_j(t))^2 \, dt}$
- $K(\mathbf{x}_i, \mathbf{x}_j) = e^{-\int_0^T \gamma(t) (\mathbf{x}_i(t) - \mathbf{x}_j(t))^2 \, dt}$
- $K(\mathbf{x}_i, \mathbf{x}_j) = e^{-\sum_{h=1}^{H} \gamma (\mathbf{x}_i(\tau_h) - \mathbf{x}_j(\tau_h))^2}$
  - $\gamma \geq 0$
  - $0 \leq \tau_{h-1} \leq \tau_h - \delta, \; h = 1, \ldots, H$

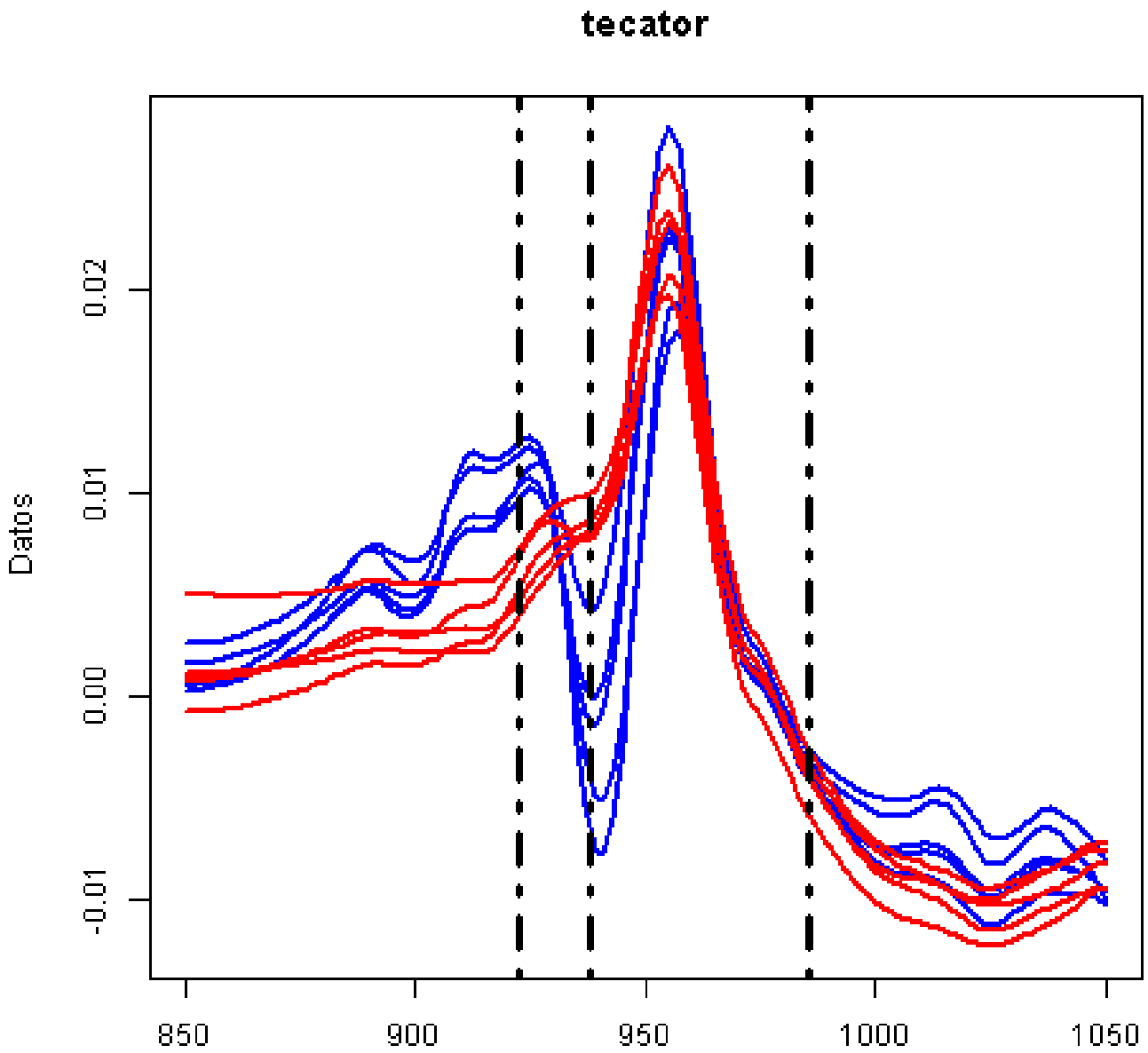
SVM with functional data



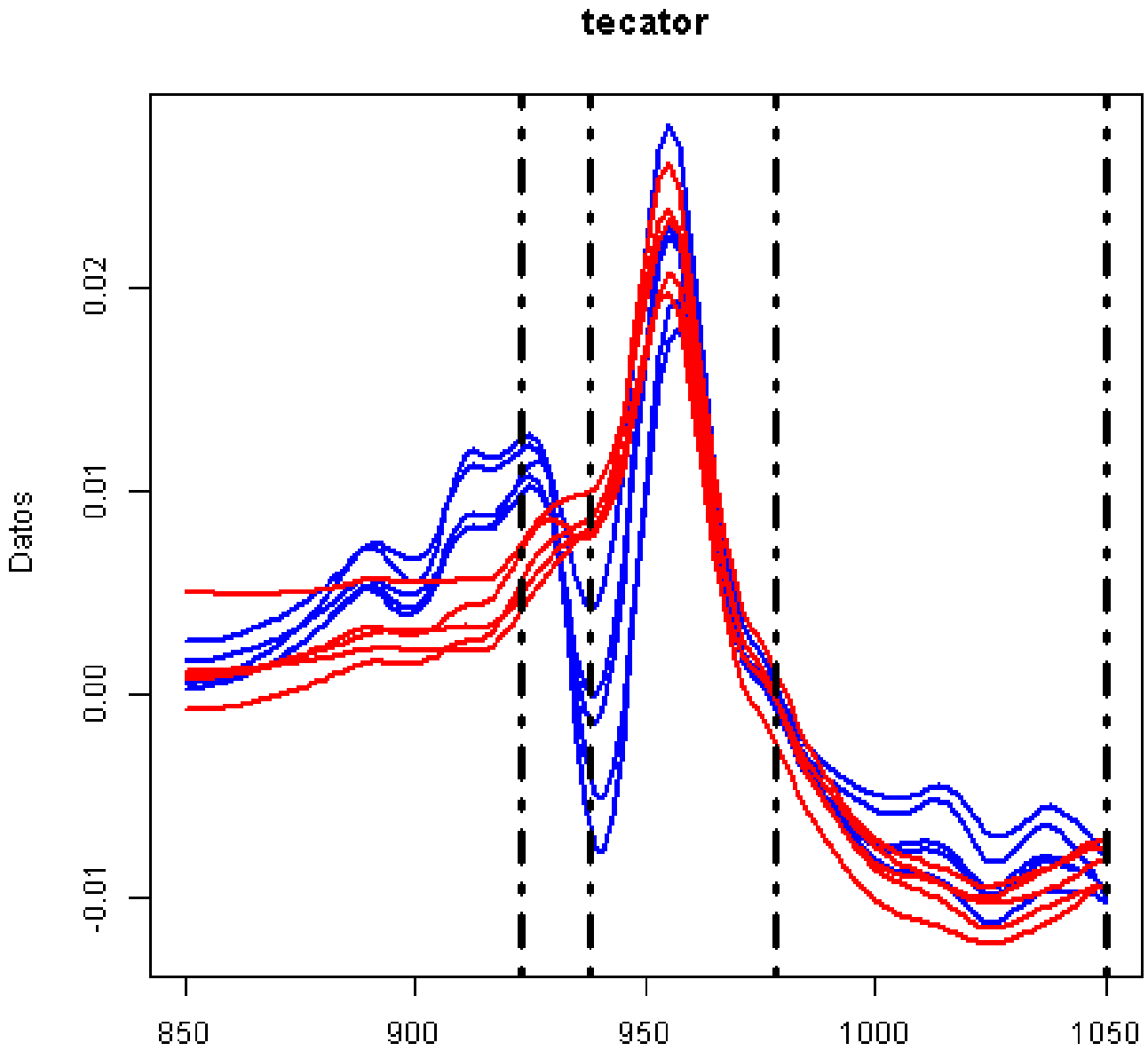
SVM with functional data



SVM with functional data



SVM with functional data



Parameters tuning (time instants selection)

Blanquero, C., Jiménez-Cordero, Martín-Barragán. Variable selection in classification for multivariate functional data. Information Sciences, 2019.

$$\theta = (\gamma | \tau_1, \ldots, \tau_{H-1})$$

Parameters tuning (time instants selection)

Blanquero, C., Jiménez-Cordero, Martín-Barragán. Variable selection in classification for multivariate functional data. Information Sciences, 2019.

$$\theta = (\gamma | \tau_1, \ldots, \tau_{H-1}) \qquad \widehat{y}_{I,C,\theta}(\mathbf{x}) = \sum_{i \in I} y_i \lambda_i^C K_\theta(\mathbf{x}, \mathbf{x}_i) + \beta^C$$

Randomly split sample I into I_1, I_2 and I_3.
for C in grid **do**
  Alternating Procedure
  repeat
    1. Fixed θ, find λ^C solving $(P_{I_1,C,\theta})$
    2. Fixed λ, find θ maximizing correlation of y and $\widehat{y}_{I,C,\theta}$ in I_2.
  until stopping criteria
end for
Return as C the one with best misclassification rate in I_3.
Return as λ and θ those associated with C

% misclassification rate (out-of-sample)

	SVM	$H = 1$	$H = 2$	$H = 3$	$H = 4$
MCO	20.80	29.02	18.64	18.14	18.81
$growth$	5.64	13.22	4.67	4.03	3.87
$phoneme$	19.88	18.00	16.96	16.36	16.20
$rain$	28.40	10.75	11.66	11.66	10
$regions$	19.46	20.75	10.26	8.10	7.23
$tecator$	3.47	4.66	2.22	2.08	1.52

Many thanks!!!

ecarrizosa@us.es