Lecture 4: SVM, PCA

Vladislav Goncharenko
MIPT, 2021

Outline

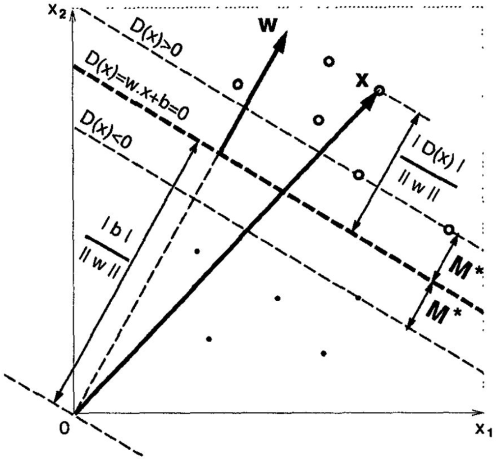
1. Support Vector Machine (SVM)
2. Dimensionality reduction and PCA
3. Validation strategies

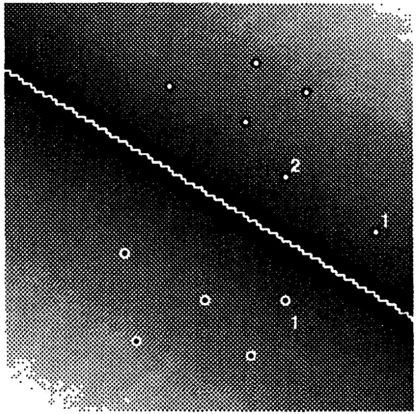
Support Vector Machine

Support Vector Machine

1. History
2. Motivation
3. Solution for separable design
4. Inseparable design, soft margin
5. Kernels
   a. Kernel definition (Hilbert spaces, inner product, positive semidefiniteness)
   b. Kernels properties (addition, infinite sums)
   c. Types of kernels (poly, exponential, gaussian)
6. Current state

History



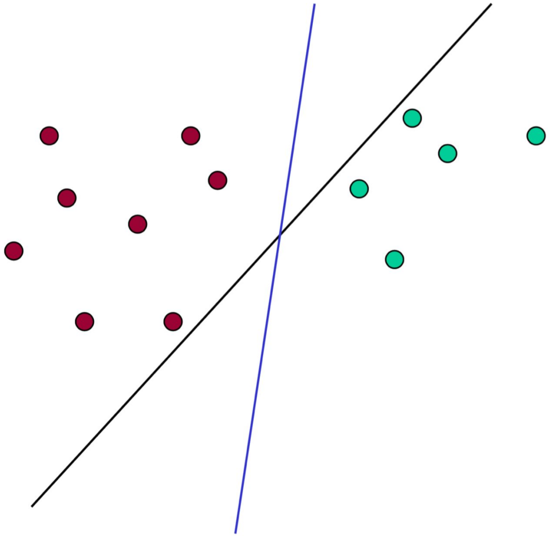


1963: SVM introduced by Soviet mathematicians Vladimir Vapnik and Alexey Chervonenkis

1992: kernel trick (Vapnik, Boser, Guyon)

1995: soft margin (Vapnik, Cortes)

Motivation



Linear separable case

Many separating hyperplanes exist

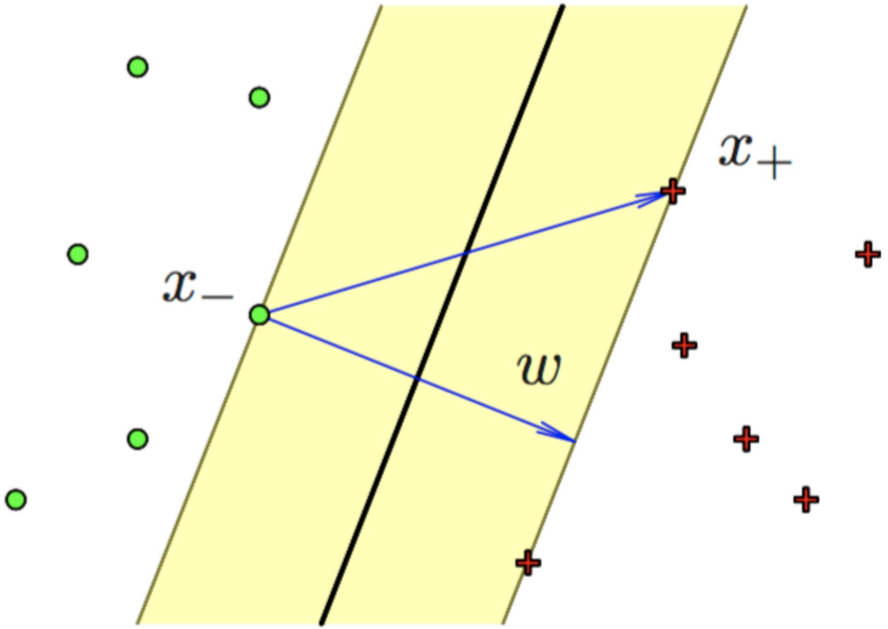
Maximize width

Margin

$$y \in \{1, -1\}$$
$$y_i = 1 : w^T x_i - c > 0$$
$$y_i = -1 : w^T x_i - c < 0$$

$$c_+(w) = \min_{y_i = 1}(w^T x_i)$$
$$c_-(w) = \max_{y_i = -1}(w^T x_i)$$

$$\rho(w) = \frac{c_+(w) - c_-(w)}{2}$$



$$\rho\left(\frac{w_0}{||w_0||}\right) = \frac{1}{||w_0||}$$

Optimization problem

$$y_i = 1 : w^T x_i - c > 0$$
$$y_i = -1 : w^T x_i - c < 0$$
$$M_i = y_i \cdot (w^T x_i - c)$$

$$\rho(w) = \frac{1}{||w||} \to \max_{w,c}$$
$$s.t.\ y_i(w^T x_i - c) \geq 1$$

Convex problem!

$$L(w, c, \alpha) = \frac{1}{2} w^T w - \sum_i \alpha_i (y_i (w^T x_i - c) - 1)$$

Many of them are zeros

Hinge loss

$$L(w, c, \alpha) = \frac{1}{2} w^T w - \sum_i \alpha_i (y_i (w^T x_i - c) - 1)$$

$$L^{\text{hinge}} = (1 - M)_+$$

$$L(w, c, \alpha) = \frac{1}{2} ||w||_2^2 + \sum_i \alpha_i L_i^{\text{hinge}}$$

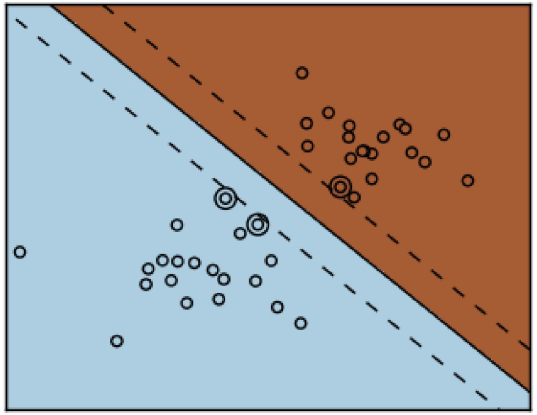
Inseparable case

Let our model mistake, but penalize that mistakes

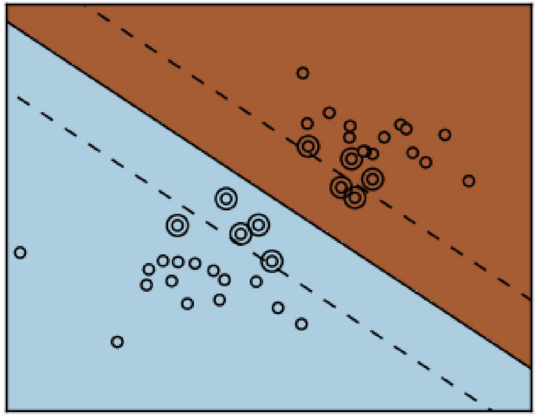
Implemented via margin slack variables

$$\begin{cases} \frac{1}{2}\langle w, w\rangle + C\sum_{i=1}^{\ell}\xi_i \to \min\limits_{w,w_0,\xi}; \\ y_i\big(\langle w, x_i\rangle - w_0\big) \geqslant 1 - \xi_i, \quad i = 1, \ldots, \ell; \\ \xi_i \geqslant 0, \quad i = 1, \ldots, \ell. \end{cases}$$

Big C



Small C



Kernel trick

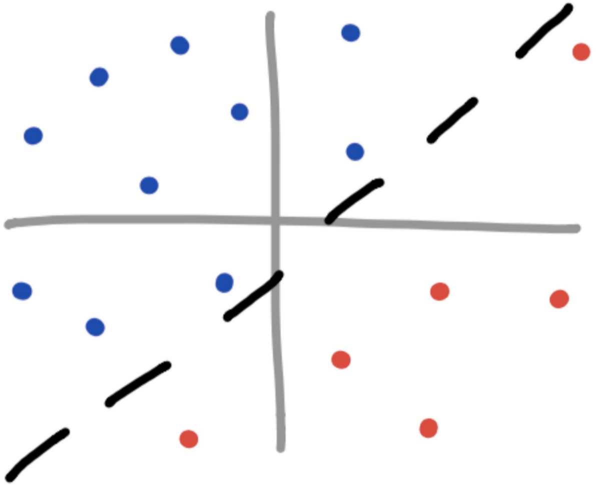
$$y_i = 1 : w^T x_i - c > 0$$

$$y_i = -1 : w^T x_i - c < 0$$

$$\begin{array}{l} x \mapsto \phi(x) \\ w \mapsto \phi(w) \end{array} \implies < w, x > \mapsto < \phi(w), \phi(x) >$$

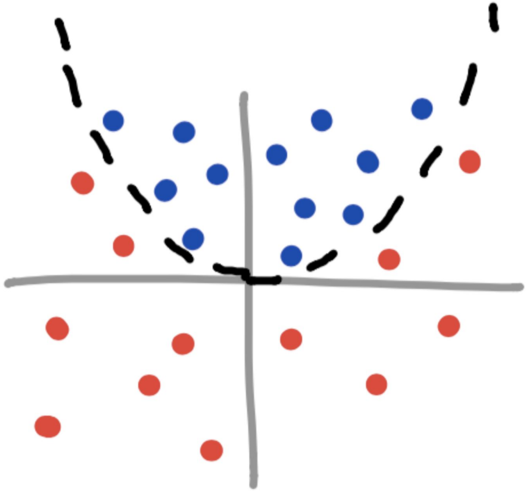
$$K(w, x) = < \phi(w), \phi(x) >$$

Kernel types



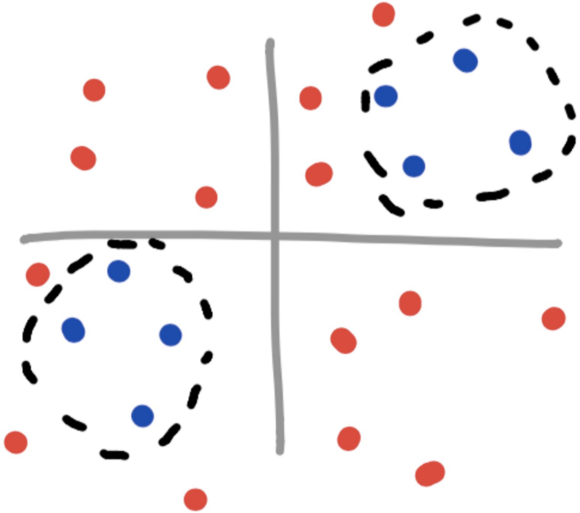
$$K(w,x) =< \mathrm{w}, x >$$

Linear



$$K(w,x) = (\gamma < w, x > +r)^d$$

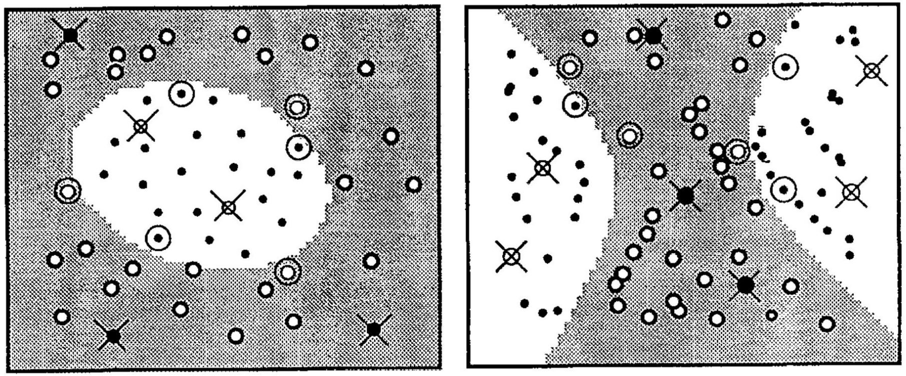
Polynomial



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

Gaussian radial basis function

Current state



Principal Component Analysis

Principal Component Analysis

$$x_1, \ldots, x_n \to g_1, \ldots, g_k, k \leq n$$

$$U : UU^T = I, G = XU$$

$$\hat{X} = GU^T \approx X$$

$$||GU^T - X|| \to \min_{G,U} \; s.t. \; rank(G) \leq k$$

Singular value decomposition

$$||GU^T - X|| \to \min_{G,U} \; s.t. \; rank(G) \leq k$$

$$X = V\Sigma U^T : ||GU^T - V\Sigma U^T||_2 = ||G - V\Sigma||_2$$

$$G = V\Sigma' : ||V\Sigma' - V\Sigma||_2 = ||\Sigma' - \Sigma||_2$$

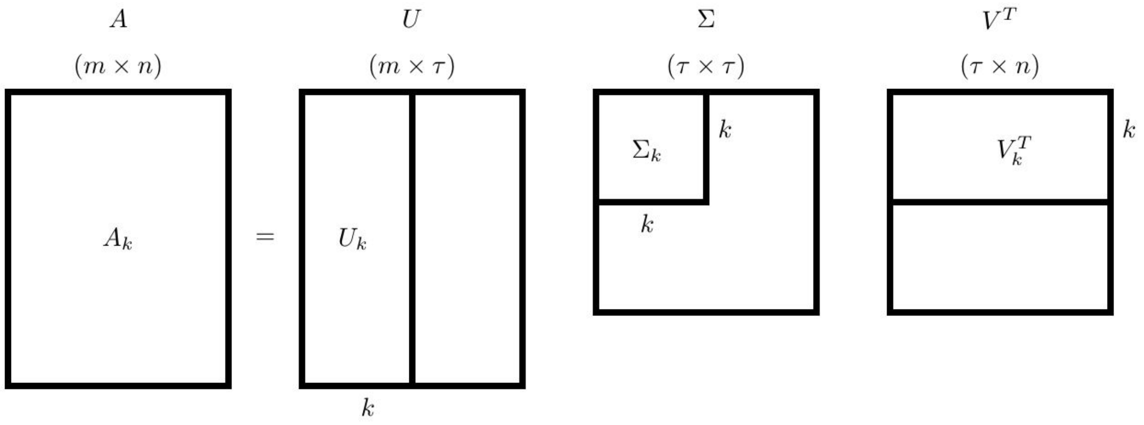
$$||A||_2 = \sigma_{max}(A) : ||\Sigma' - \Sigma||_2 = \sigma_k(\Sigma) = \sigma_k(X)$$

Eckart–Young–Mirsky theorem

Singular value decomposition

$$||GU^T - X|| \to \min_{G,U} \; s.t. \; rank(G) \le k$$

$$X = V\Sigma U^T \qquad \sigma_k(\Sigma) = \sigma_k(X)$$



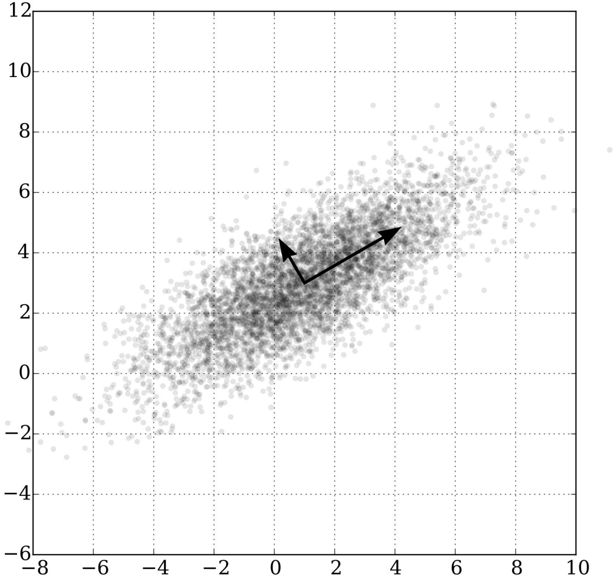
Eckart–Young–Mirsky theorem

Another approach

Residual variance maximization

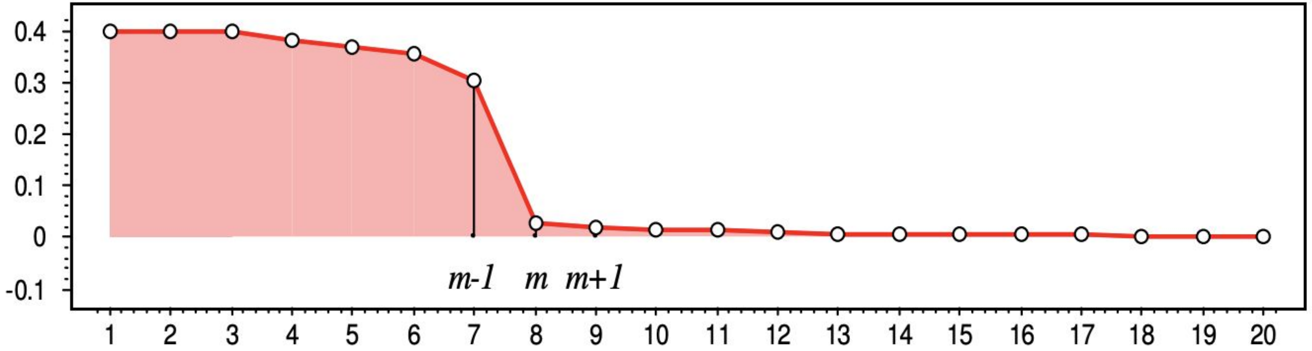
Take new basis vectors greedy

Same result for G and U



Always normalize data before PCA!!!

Dimensionality reduction



Get rid of low-variance components

$$E_m = \frac{\|GU^\intercal - F\|^2}{\|F\|^2} = \frac{\lambda_{m+1} + \cdots + \lambda_n}{\lambda_1 + \cdots + \lambda_n} \leqslant \varepsilon.$$

Dimensionality reduction

Let’s walk through space…

Dimensionality reduction

16 components

Dimensionality reduction

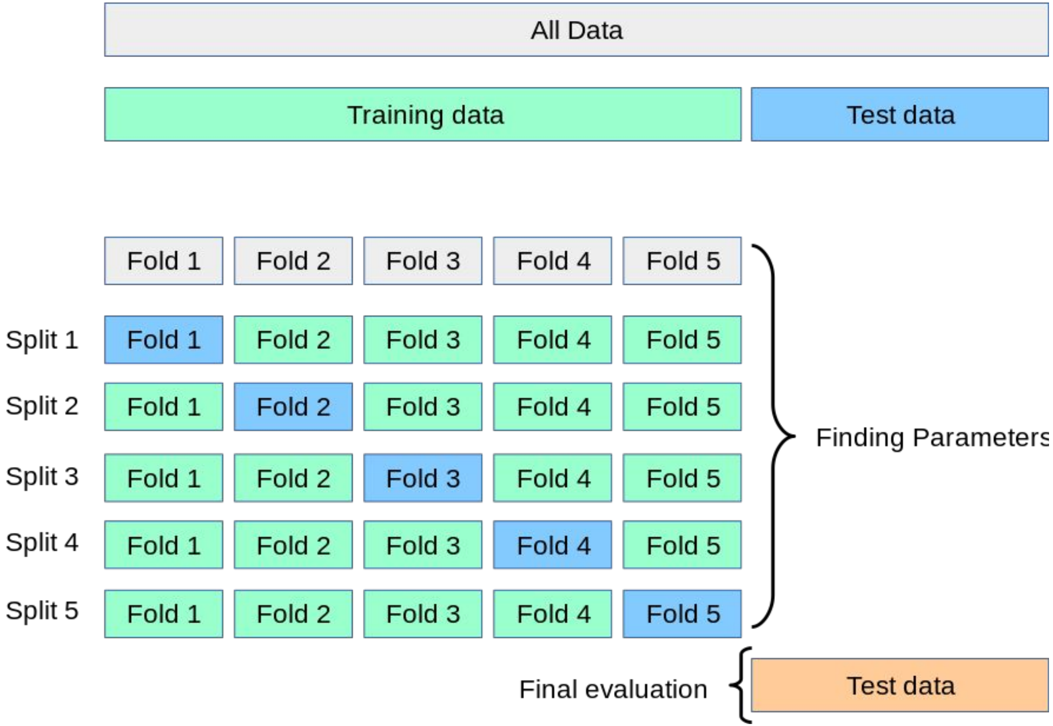
50 components

Dimensionality reduction

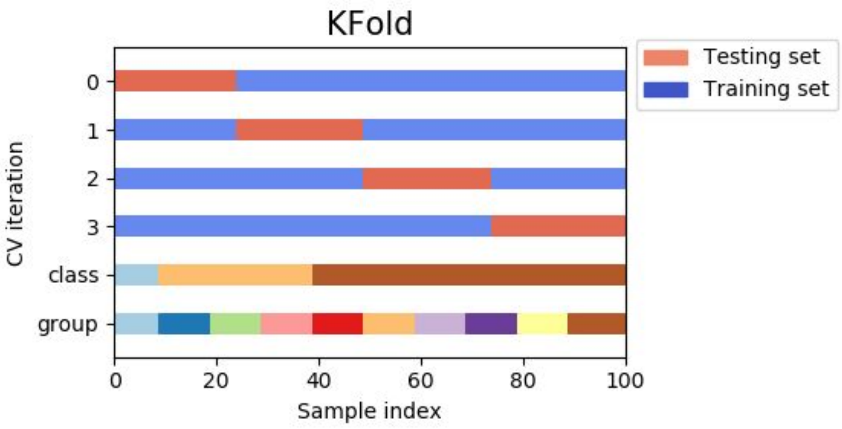
250 components

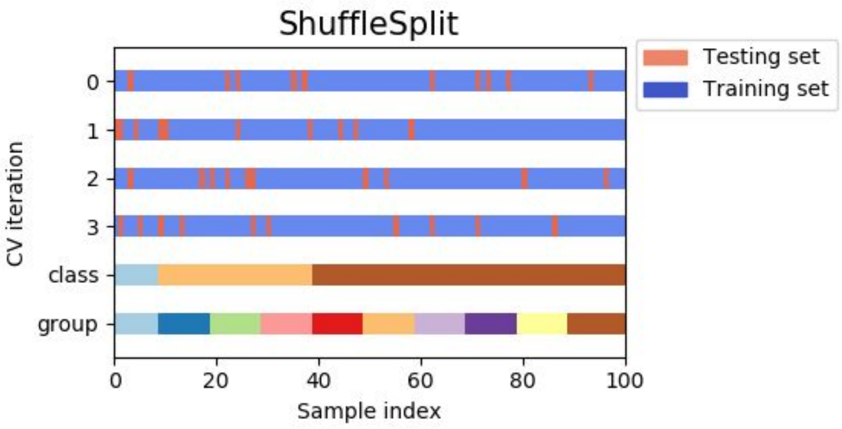
Validation strategies

Validation strategies



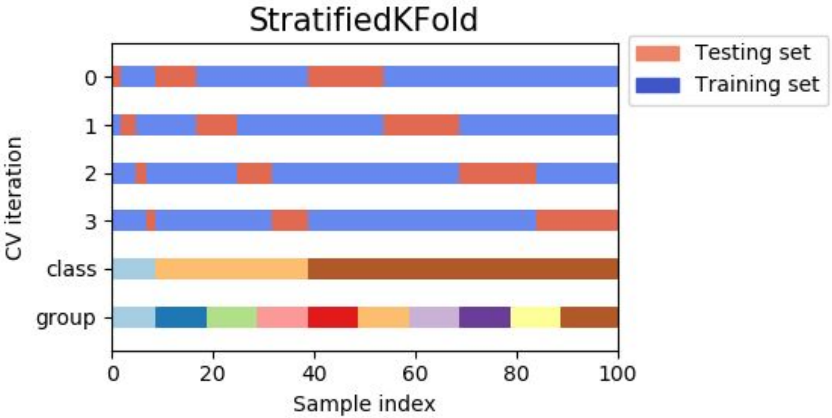
Validation strategies

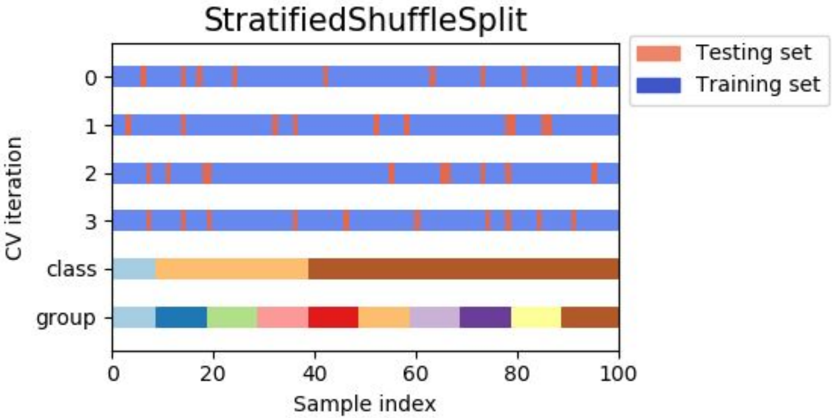




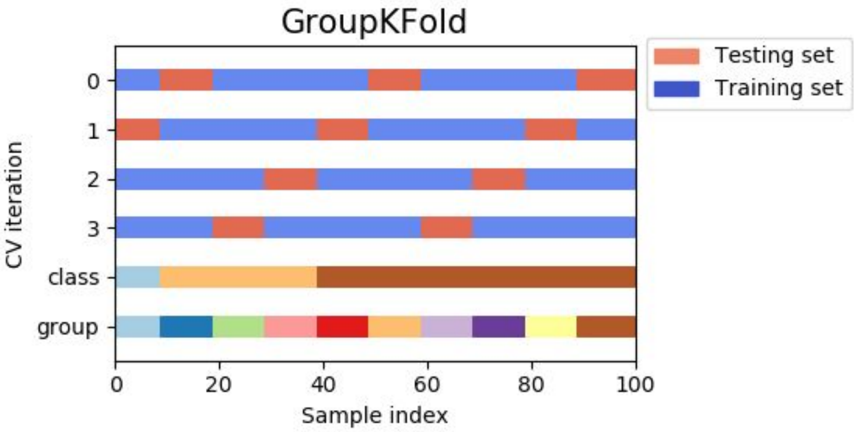
Special case: Leave One Out (LOO) - good for tiny datasets

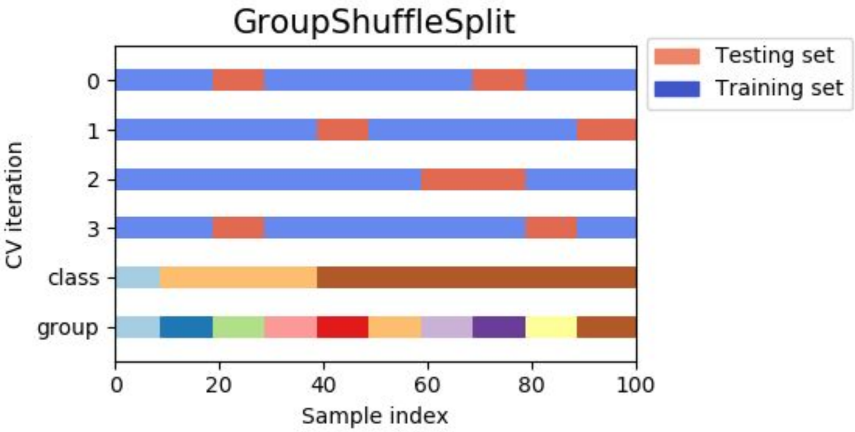
Validation strategies



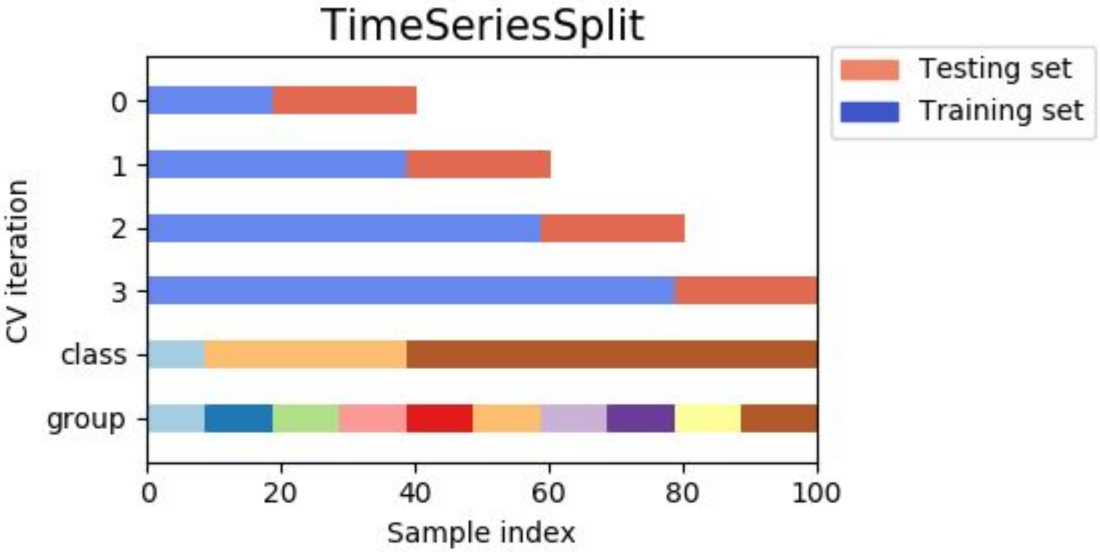


Validation strategies





Special case: timeseries



Never use train_test_split in this case!!!

Thanks for attention!

Maximum Likelihood Estimation

Maximum Likelihood Estimation

What are reasons behind defining “best” linear estimator?

Maximize probability of particular parameter to explain given data

$$L(\theta|X,Y) = P(X,Y|\theta)$$

assuming i.i.d. observations

$$P(X,Y|\theta) = \Pi_{i=1}^{n} P(x^i, y^i|\theta)$$

$$\log L(\theta|X,Y) = \sum_{i=1}^{n} \log P(x^i, y^i|\theta)$$