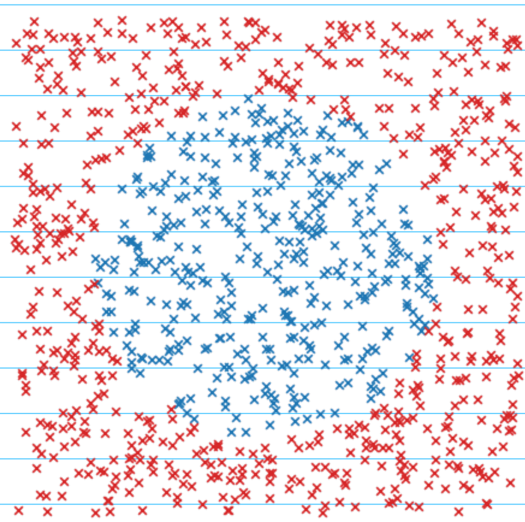
Kernel Methods

So far: we know how to compute "linear" predictors.

→ Linear Regression
→ Ridge Regression
→ LASSO
→ Logistic Regression

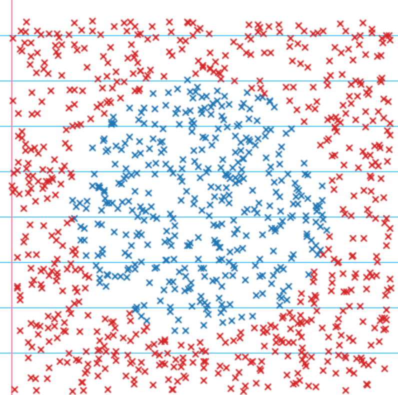
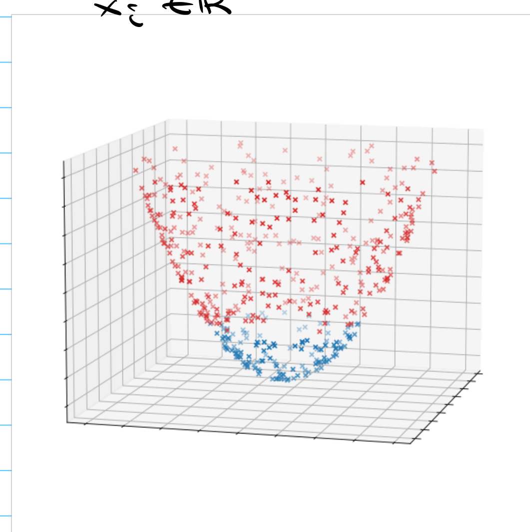
} Closed form or gradient descent.



① Feature maps :

" If a linear predictor will not work, then "lift" the dataset to higher dimension and apply a linear method on the lifted dataset. "

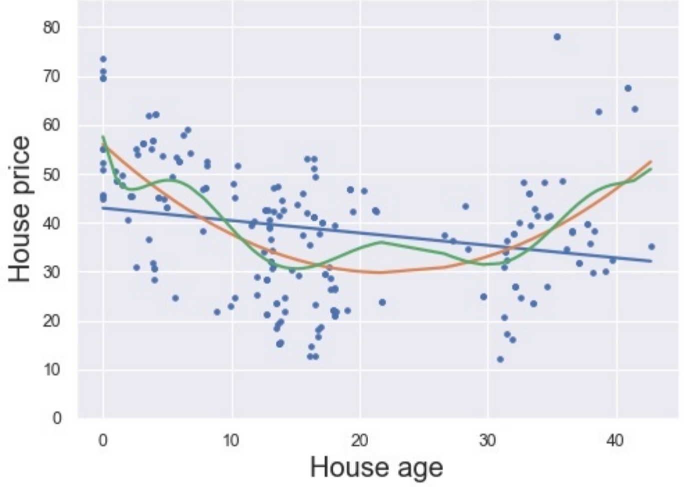
$(X_1, Y_1) \ldots (X_n, Y_n)$ $\qquad x_i \in \mathbb{R}^2$

$$x \in \mathbb{R}^2 \xrightarrow{\Phi} \Phi(x) = (x, \|x\|^2) \in \mathbb{R}^3$$

$\rightsquigarrow$ Linear classification on $(\Phi(X_1), Y_1) \ldots (\Phi(X_n), Y_n)$ has a great performance.

• Revisiting Polynomial Regression:



— $d = 1$

— $d = 2$

— $d = 8$

$X_1 \; ... \; X_n \in \mathbb{R}$
$Y_1 \; ... \; Y_n \in \mathbb{R}$

$\rightsquigarrow$ Linear predictor ? ☹

Use a FEATURE MAP $\Phi : \mathbb{R} \longrightarrow \mathbb{R}^{d+1}$

$\Phi(x) = (1, x, x^2 \dots x^d)$.

Let $a = (a_0, a_1 \dots a_d) \in \mathbb{R}^{d+1}$

Then $\langle a, \Phi(x) \rangle = a_0 + a_1 x + \dots + a_d x^d$
$= P_a(x)$

→ Polynomial regression aims at minimizing:

$$a \in \mathbb{R}^{d+1} \longmapsto \frac{1}{n} \sum_{i=1}^{n} |y_i - P_a(x_i)|^2$$

$$= \frac{1}{n} \sum_{i=1}^{n} |y_i - \langle a, \Phi(x_i) \rangle|^2$$

$$= \frac{1}{n} \| \mathbb{Y} - \tilde{\mathbb{X}} a \|^2$$

$$\mathbb{Y} = \begin{pmatrix} y_1 \\ \vdots \\ y_n \end{pmatrix} \qquad \tilde{\mathbb{X}} = \begin{bmatrix} \Phi(x_1) \\ \vdots \\ \Phi(x_n) \end{bmatrix} = \underbrace{\begin{bmatrix} 1 & x_1 & x_1^2 & \cdots & x_1^d \\ \vdots & & & & \\ 1 & x_n & x_n^2 & \cdots & x_n^d \end{bmatrix}}_{d} \; n$$

→ Polynomial regression

$\Longleftrightarrow$ Linear regression on

$\Phi(x_1) \; - - - - \; \Phi(x_n)$

$y_1 \; - - - - \; y_n$.

Once again ... ① Lift on higher dimension using a feature map

② Apply a linear technique on the transformed dataset.

More Generally : $\mathcal{X}$ general set

$\mathcal{Y} = \mathbb{R}$

$\Phi : \mathcal{X} \to \mathbb{R}^D$ feature map

$(X_1, Y_1), \dots (X_n, Y_n)$ training sample.

Linear regression on $(\Phi(X_1), Y_1) \dots (\Phi(X_n), Y_n)$

$$R_n : a \in \mathbb{R}^D \longmapsto \frac{1}{n} \sum_{i=1}^{n} |Y_i - \langle a, \Phi(X_i) \rangle|^2$$

$$= \frac{1}{n} \| \mathbb{Y} - \tilde{\mathbb{X}} a \|^2$$

$$\tilde{\mathbb{X}} = \begin{pmatrix} \Phi(X_1) \\ \vdots \\ \Phi(X_n) \end{pmatrix} \quad n \times D.$$

$$\hat{a} = \underbrace{(\tilde{\mathbb{X}}^T \tilde{\mathbb{X}})^{-1}}_{D \times D} \tilde{\mathbb{X}}^T \mathbb{Y}$$

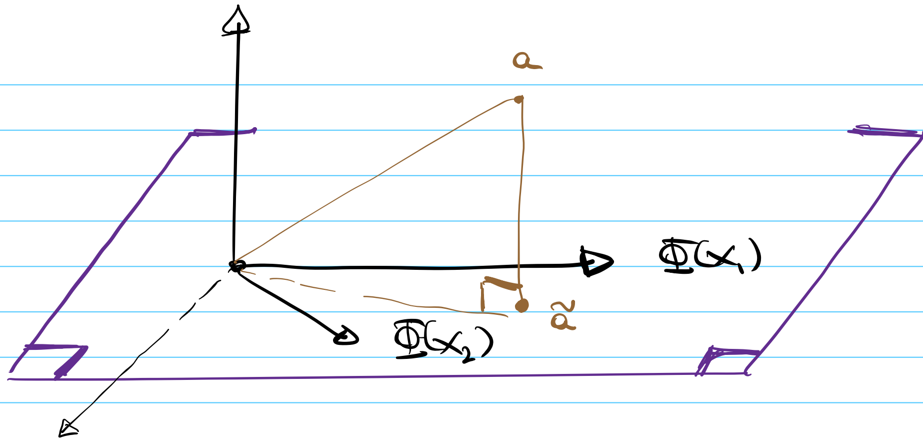
* if $n \geq D$
compute pseudo inverse

Computation cost ? $O(D^3)$

What if $D \gg 1$? ☹

$\Rightarrow$ The Kernel Trick

Let $\tilde{a}$ be the orthogonal projection of a on $\mathrm{Span}(\Phi(X_1) \dots \Phi(X_n)) = E$



$$\rightsquigarrow \quad \langle a, \Phi(x_i)\rangle = \langle \tilde{a}, \Phi(x_i)\rangle$$

$$\Rightarrow \quad \mathbb{X} a = \mathbb{X} \tilde{a} .$$

We may consider only vectors $\tilde{a} \in E$.

$$\Rightarrow \quad \tilde{a} = \sum_{j=1}^{n} b_j \, \Phi(X_j) \quad \text{for some } b_1, \dots, b_n \in \mathbb{R}.$$

$$\frac{1}{n} \sum_{i=1}^{n} |Y_i - \langle \Phi(x_i), \tilde{a} \rangle|^2$$

$$= \frac{1}{n} \sum_{i=1}^{n} |Y_i - \langle \Phi(x_i), \sum_{j=1}^{n} b_j \, \Phi(x_j) \rangle|^2$$

$$= \frac{1}{n} \sum_{i=1}^{n} |Y_i - \sum_{j=1}^{n} b_j \langle \Phi(x_i), \Phi(x_j) \rangle|^2$$

$$= \frac{1}{n} \| \mathbb{Y} - \mathbb{G} b \|^2 \qquad b = \begin{pmatrix} b_1 \\ \vdots \\ b_n \end{pmatrix} \in \mathbb{R}^n$$

where

$$\mathbb{G} = \begin{bmatrix} \langle \Phi(x_1), \Phi(x_1) \rangle & \cdots & \langle \Phi(x_1), \Phi(x_n) \rangle \\ \vdots & & \vdots \\ \langle \Phi(x_n), \Phi(x_1) \rangle & \cdots & \langle \Phi(x_n), \Phi(x_n) \rangle \end{bmatrix}$$

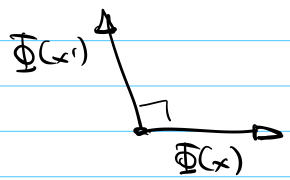
GRAM MATRIX
$n \times n$

$\rightsquigarrow$ if $\mathbb{G}$ invertible, $\hat{b} = \mathbb{G}^{-1} Y$

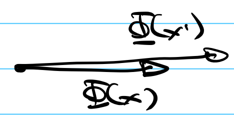
cost $O(n^3)$. $\ll O(D^3)$ if $n \ll D$ || Lift in very high dim

② Reproducing Kernel Hilbert Spaces

The Gram matrix $\mathbb{G}$ measures the proximity between the observations.



x and x' not similar



x and x' similar

$\rightarrow$ What if we replace $\langle \Phi(x_i), \Phi(x_j) \rangle$ by a general "measure of similarity" $k(x_i, x_j)$.

Ex: RBF / Gaussian kernel

$$k_\sigma(x, x') = \exp\left(-\frac{\|x - x'\|^2}{2\sigma^2}\right)$$

If $k(x,x') = \langle \Phi(x), \Phi(x') \rangle$ for some feature map Φ, then the previous discussion applies.

Observations $x_1 \dots x_n$ $\Rightarrow$ Lift $\Phi(x_1) \dots \Phi(x_n)$ $\Rightarrow$ Gram matrix $G = (k(x_i, x_j))$

$\Rightarrow$ Linear Regression $\hat{b} = G^{-1} Y$

$\rightarrow$ How can we know that k can be written in this way?

- Assume $k(x,x') = \langle \Phi(x), \Phi(x') \rangle$.

Then, $\forall\, \alpha_1 \dots \alpha_n \in \mathbb{R}$, $x_1 \dots x_n \in \mathcal{X}$,

$$0 \leq \left\| \sum_{i=1}^{n} \alpha_i \Phi(x_i) \right\|^2$$

$$= \sum_{i=1}^{n} \sum_{j=1}^{m} \alpha_i \alpha_j \langle \Phi(x_i), \Phi(x_j) \rangle$$

$$\left[\, 0 \leq \sum_{i=1}^{n} \sum_{j=1}^{m} \alpha_i \alpha_j k(x_i, x_j) \,\right] \quad (*)$$

$\rightsquigarrow$ If there exists feature map, then k satisfies $(*)$ $\forall\, \alpha_1 \dots \alpha_n \in \mathbb{R}$, $x_1 \dots x_n \in \mathcal{X}$

$\Rightarrow$ This condition is also sufficient!

Hilbert spaces : $\mathcal{H}$ vector space with a DOT PRODUCT

① Symmetry : $\forall x,y \in \mathcal{H}$, $\langle x,y\rangle_{\mathcal{H}} = \langle y,x\rangle_{\mathcal{H}}$

② Linearity : $\forall x,y,z \in \mathcal{H}$, $\lambda,\mu \in \mathbb{R}$

$$\langle x, \lambda y + \mu z\rangle_{\mathcal{H}} = \lambda\langle x,y\rangle_{\mathcal{H}} + \mu\langle x,z\rangle_{\mathcal{H}}$$

③ Positive definiteness : $\forall x \in \mathcal{H}$, $\langle x,x\rangle_{\mathcal{H}} \geq 0$
with $=0$ iff $x=0$. $\|x\|_{\mathcal{H}} = \sqrt{\langle x,x\rangle_{\mathcal{H}}}$ is a norm.

④ Completeness : For every continuous linear map $L : \mathcal{H} \to \mathbb{R}$, $\exists\, h \in \mathcal{H}$ such that $\forall x \in \mathcal{H}$, $L(x) = \langle h, x\rangle$.

ensures that $\left\{ \begin{array}{l} \text{infinite sums} \\ \text{projections} \end{array} \right.$ are well-defined.

" Hilbert space $\simeq$ Like $\mathbb{R}^d$ but possibly of infinite dimension... "

Example : $L_2(\mathbb{R}) = \{ f : \mathbb{R} \to \mathbb{R},\ \int |f(x)|^2 dx < \infty \}$

$$\langle f,g\rangle_{L_2(\mathbb{R})} = \int f(x)g(x)\,dx$$

Def: Let $k : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$. We say that k is a (positive definite) kernel $\Leftrightarrow$

① k is symmetric: $\forall x, x' \in \mathcal{X}$, $k(x,x') = k(x',x)$

② $\forall d_1 \dots d_n \in \mathbb{R}$, $x_1 \dots x_n \in \mathcal{X}$

$$\sum_{1 \le i,j \le n} d_i d_j \, k(x_i, x_j) \geq 0 .$$

THEOREM: If k is a kernel, then there exists $\begin{cases} \text{Hilbert space } \mathcal{H} & \text{RKHS} \\ \Phi : \mathcal{X} \to \mathcal{H} & \text{Feature map} \end{cases}$

such that $\forall x, x' \in \mathcal{X}$,

$$k(x,x') = \langle \Phi(x), \Phi(x') \rangle_{\mathcal{H}} .$$

proof sketch:

$$\mathcal{H}_0 = \left\{ \sum_{i=1}^{n} d_i k(x_i, \cdot) ; \; n \in \mathbb{N}, \; x_i \in \mathcal{X}, \; d_i \in \mathbb{R} \right\}$$

$\hookrightarrow$ vector space.

$$\Phi : \mathcal{X} \to \mathcal{H}_0$$
$$x \mapsto k(x, \cdot)$$

Define $\left\langle \sum_{i=1}^{n} \alpha_i k(x_i,\cdot), \sum_{j=1}^{m} \beta_j k(x_j,\cdot) \right\rangle_{\mathcal{H}_0}$

$$:= \sum_{i=1}^{n} \sum_{j=1}^{m} \alpha_i \beta_j \, k(x_i, x_j)$$

This is a dot product

By construction: $\langle \Phi(x), \Phi(x') \rangle_{\mathcal{H}_0}$

$$= \langle k(x,\cdot), k(x',\cdot) \rangle_{\mathcal{H}_0} = k(x,x').$$

Pb : $\mathcal{H}_0$ is not complete.

It can be completed using a process called completion. $\boxtimes$

How to construct kernels ?

k_1, k_2 kernels on $\mathcal{X}$.

1. $k_1 + k_2$ is a kernel.
2. $k_1 \cdot k_2$ is a kernel.
3. $\mathcal{X} = \mathbb{R}^d$ $\quad k(x,x') = k(x-x')$

k is a kernel if the Fourier transform

$$\mathcal{F}[k](\xi) = \int e^{-2\pi i \langle x, \xi \rangle} k(x)\,dx \geq 0.$$

proof:

(1)

(2) See Lecture notes

(3) $k(x) = \int e^{2i\pi\langle x,\xi\rangle} \mathcal{F}[k](\xi)d\xi$

$$\sum_{ij} d_i d_j k(x_i - x_j) = \int \sum_{ij} d_i d_j \underbrace{\underbrace{e^{2i\pi\langle x_i - x_j,\xi\rangle}}_{e^{2i\pi\langle x_i,\xi\rangle}\,\overline{e^{2i\pi\langle x_j,\xi\rangle}}}}_{} \mathcal{F}[k](\xi)d\xi$$

$$= \sum z_i \bar{z}_j = \|\sum z_i\|^2$$

where $z_i = d_i e^{2i\pi\langle x_i,\xi\rangle}$

Examples:

(1) $\Phi : \mathcal{X} \to \mathcal{H}$ any map. $k(x,x') = \langle\Phi(x),\Phi(x')\rangle_{\mathcal{H}}$ is a kernel.

(2) $k(x,x') = \langle x,x'\rangle^{\alpha}$ $\quad \alpha \in \mathbb{N}$

(3) Radial Basis Function (RBF) kernel

$$k_\sigma(x,x') = \exp\left(-\frac{\|x-x'\|^2}{2\sigma^2}\right)$$

$\rightsquigarrow \mathcal{F}[k_\sigma](\xi) = \sqrt{2\pi\sigma^2}\, e^{-2\pi\sigma^2\|\xi\|^2} \geq 0$

⑧ Kernel Ridge Regression

Classic Ridge Regression :

$x_1 \ldots x_n \in \mathbb{R}^d$
$y_1 \ldots y_n \in \mathbb{R}$

Regularization term

Minimize $\beta \in \mathbb{R}^d \longmapsto \frac{1}{n}\|\mathbb{X}\beta - \mathbb{Y}\|^2 + \lambda\|\beta\|^2$

$$\mathbb{X} = \underbrace{\begin{bmatrix} x_1 \\ \vdots \\ x_n \end{bmatrix}}_{d} \qquad \mathbb{Y} = \begin{bmatrix} y_1 \\ \vdots \\ y_n \end{bmatrix}$$

$$\rightsquigarrow \boxed{\hat{\beta} = (\mathbb{X}^T\mathbb{X} + \lambda n I_d)^{-1}\mathbb{X}^T\mathbb{Y}}$$

Let's kernalize !

$x_1, \ldots x_n \in \mathcal{X}$ ⟶ any set
$y_1 \ldots y_n \in \mathbb{R}$

k kernel on $\mathcal{X}$.
↳ Φ feature map $\mathcal{X} \to \mathcal{H}$

Kernel Ridge Regression :

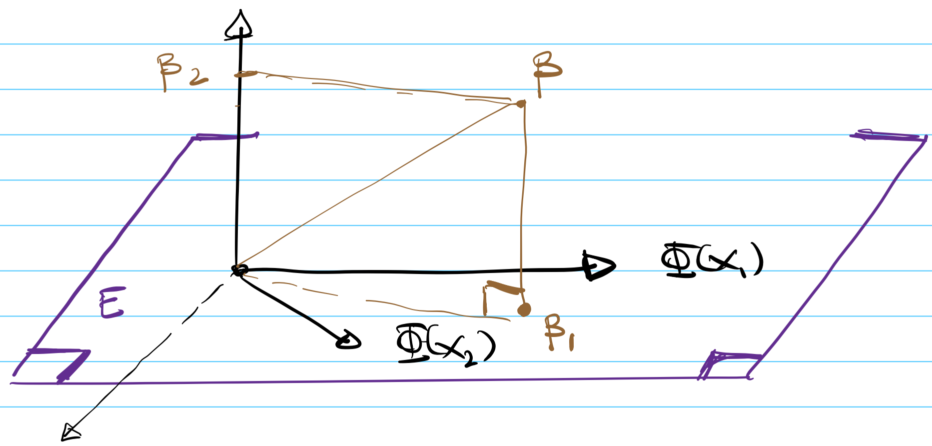
$$(*) \quad \boxed{\text{Minimize } \beta \in \mathcal{H} \longmapsto \frac{1}{n}\sum_{i=1}^{n} |\langle \Phi(x_i), \beta\rangle - y_i|^2 + \lambda\|\beta\|^2_{\mathcal{H}}}$$

$\rightsquigarrow$ How can we find the minimum?

Representer Theorem:

The minimum of $(*)$ is attained at $\beta \in \mathcal{H}$ of the form $\sum_{i=1}^{n} a_i \Phi(x_i)$.

proof: Let $E = \mathrm{Span}(\Phi(x_1) \dots \Phi(x_n))$



$$\beta = \underbrace{\beta_1}_{\in E} + \underbrace{\beta_2}_{\in E^\perp} \quad (= \text{orthogonal of } E)$$

$\rightsquigarrow \langle \beta, \Phi(x_i) \rangle = \langle \beta_1, \Phi(x_i) \rangle$

$\rightsquigarrow \|\beta\|_{\mathcal{H}}^2 = \|\beta_1\|_{\mathcal{H}}^2 + \|\beta_2\|_{\mathcal{H}}^2$

$$\Rightarrow \quad \frac{1}{n} \sum_{i=1}^{n} |\langle \Phi(x_i), \beta \rangle - y_i|^2 + \lambda \|\beta\|_{\mathcal{H}}^2$$

$$= \frac{1}{n} \sum_{i=1}^{n} |\langle \Phi(x_i), \beta_1 \rangle - y_i|^2 + \lambda \|\beta_1\|_{\mathcal{H}}^2 + \underbrace{\lambda \|\beta_2\|_{\mathcal{H}}^2}_{\geq 0} \quad \boxtimes$$

$\rightsquigarrow$ We minimize over $a = (a_1, \dots, a_n) \in \mathbb{R}^n$

$\beta_a = \sum_{j=1}^{n} a_j \Phi(x_j)$

$$\frac{1}{n}\sum_{i=1}^{n} |\langle \Phi(x_i), \beta_a\rangle - y_i|^2 + \lambda \|\beta_a\|_{\mathcal{H}}^2$$

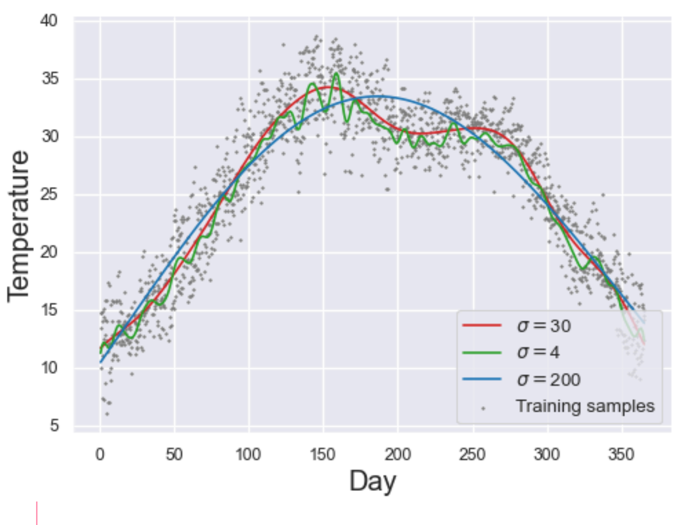
$$= \frac{1}{n}\sum_{i=1}^{n} \Big| \sum_{j=1}^{n} a_j \underbrace{\langle \Phi(x_i), \Phi(x_j)\rangle}_{k(x_i, x_j)} - y_i \Big|^2$$

$\mathbb{G} = (k(x_i, x_j))$

$$+ \lambda \sum_{i,j} a_i a_j \langle \Phi(x_i), \Phi(x_j)\rangle$$

$$= \underbrace{\frac{1}{n}\sum_{i=1}^{n} |\mathbb{G}_i^T a - y_i|^2}_{= \|\mathbb{G}a - \mathbb{y}\|^2} + \lambda a^T \mathbb{G} a$$

$\rightarrow$ Minimizer $\hat{a} = (\mathbb{G} + \lambda n I_n)^{-1} \mathbb{y}$.



x_i = day

$k_\sigma(x, x') = e^{-\frac{(x-x')^2}{2\sigma^2}}$

$\Downarrow$

Compute $\mathbb{G}$

$\Downarrow$

Compute $\hat{a}$

⚠ Choice of σ is critical!

" $k_\sigma(x,x') \lesssim 1$ if $\|x-x'\| \lesssim \sigma$ "

Here $\sigma = 30$: " if two days are at distance $\lesssim$ 30 days, they should be treated similarly. "

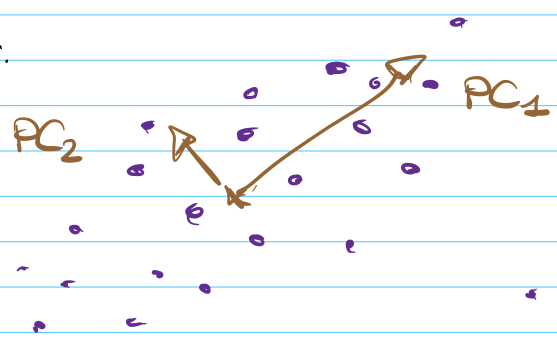
Computations ? $\hat{\alpha} = (\underbrace{\mathbb{G} + \lambda n I_n}_{n \times n \text{ matrix}})^{-1} \mathbb{Y}$.

$\rightsquigarrow O(n^3) =$ a lot!! (but some tricks, see HW)

Kernel methods are tractable for moderate n : $n \lesssim 50000$.

(4) Kernel PCA :

Classic PCA :



$x_1, \ldots, x_n \in \mathbb{R}^d$ $\quad \mathcal{X} = \begin{bmatrix} x_1 \\ \vdots \\ x_n \end{bmatrix}$

Principal components = first k eigenvectors of $\mathcal{X}\mathcal{X}^T = G$

$\hookrightarrow$ $n \times n$ matrix.

$\rightarrow$ Best k-dimensional LINEAR fit of $x_1 \ldots x_n$

$\rightsquigarrow$ Dimension Reduction

$\rightarrow$ What if no good LINEAR fit?

$\rightarrow$ What if $x_i \notin \mathbb{R}^d$

Kernel PCA $\quad$ k kernel on $\mathcal{X}$

$\hookrightarrow$ $\Phi : \mathcal{X} \rightarrow \mathcal{H}$ feature map.

$\Rightarrow$ Apply PCA on $\Phi(x_1), \ldots, \Phi(x_n)$.

$\rightsquigarrow$ First k eigenvectors of

$$G = (\langle \Phi(x_i), \Phi(x_j) \rangle)_{ij}$$
$$= (k(x_i, x_j))_{ij}$$

Example: Word2Vec : $\mathcal{X}$ = set of words

$\Phi : \mathcal{X} \to \mathbb{R}^{25}$ $\quad k(x,x') = \langle \Phi(x), \Phi(x') \rangle$

$x_1, x_2 \dots x_n$ = words

$\leadsto$ either a country or an emotion

germany
kenya

anxious
joy