Kernel

Announcements

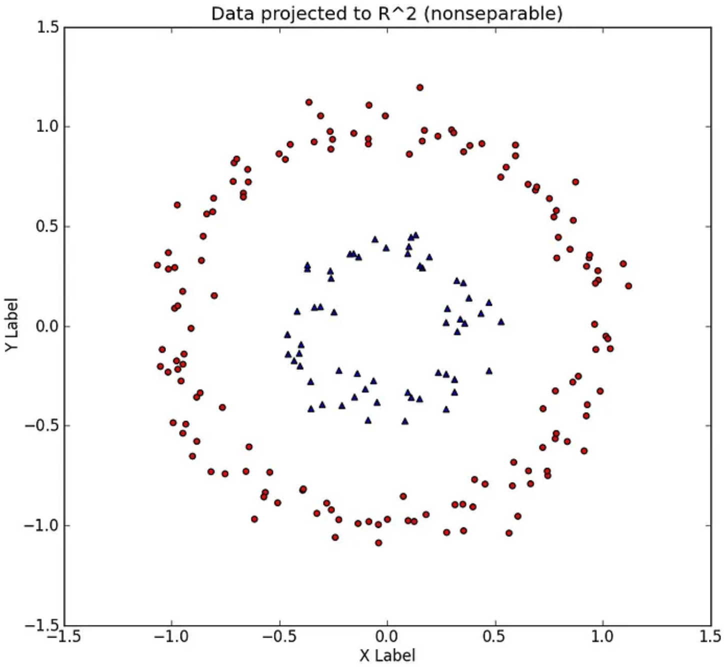
HW5 and P5 are released (due in one week)

Objective today (and next Tuesday)

Use kernels to design nonlinear ML models (regression & classification)

Objective today (and next Tuesday)

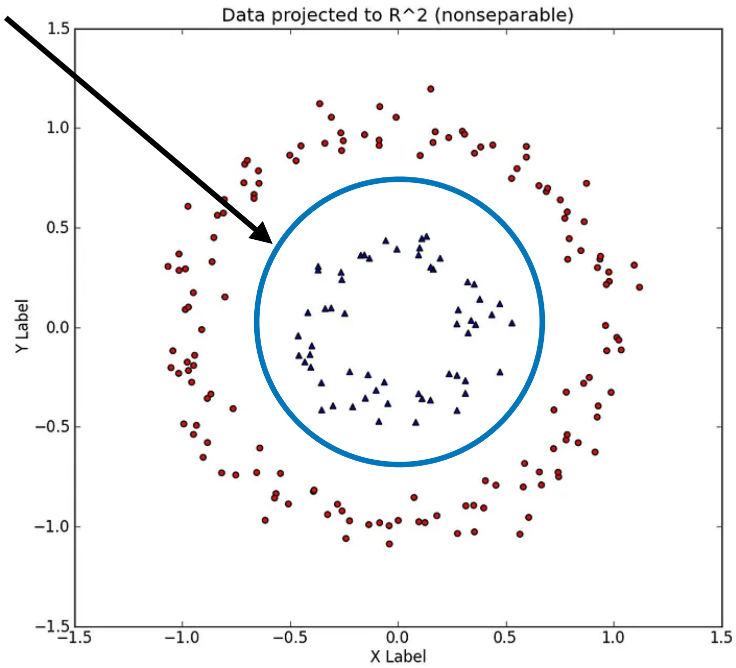
Use kernels to design nonlinear ML models (regression & classification)



Objective today (and next Tuesday)

Use kernels to design nonlinear ML models (regression & classification)

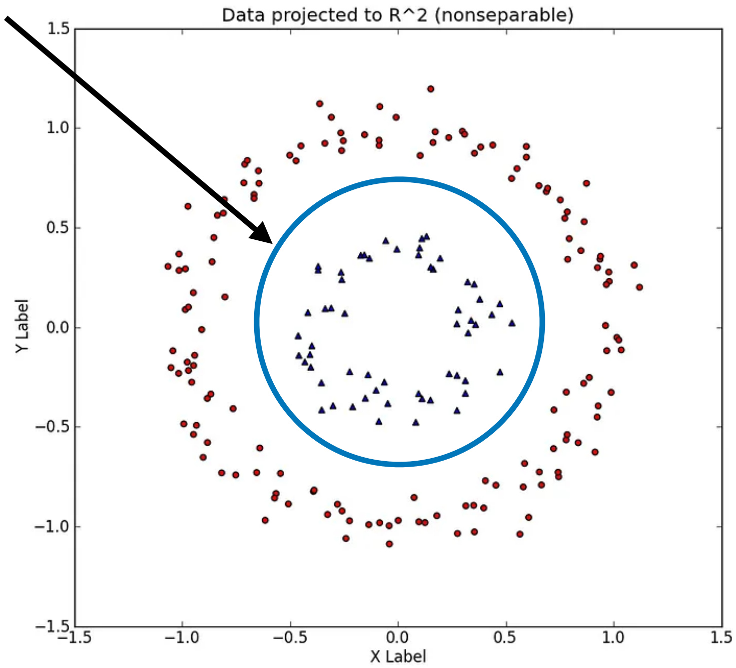
Goal: Non-linear decision boundary



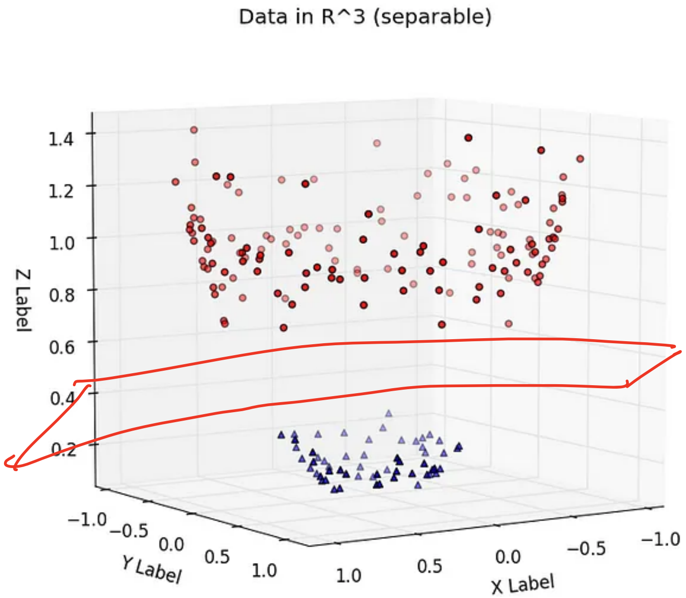
Objective today (and next Tuesday)

Use kernels to design nonlinear ML models (regression & classification)

Goal: Non-linear decision boundary



Our approach



Outline

1. A new perspective on ridge linear regression

2. Feature mapping and Kernel

3. Kernel trick and demo of kernel regression

Linear regression revisited

Dataset $\mathscr{D} = \{\mathbf{x}_i, y_i\}, \mathbf{x}_i \in \mathbb{R}^d, y_i \in \mathbb{R}$

Linear regression revisited

Dataset $\mathscr{D} = \{\mathbf{x}_i, y_i\}, \mathbf{x}_i \in \mathbb{R}^d, y_i \in \mathbb{R}$

Ridge Linear regression solves the following problem:

$$\arg\min_{w} \sum_{i=1}^{n} (w^\top \mathbf{x}_i - y_i)^2 + \lambda \|w\|_2^2$$

Linear regression revisited

Dataset $\mathscr{D} = \{\mathbf{x}_i, y_i\}, \mathbf{x}_i \in \mathbb{R}^d, y_i \in \mathbb{R}$

Ridge Linear regression solves the following problem:

$$\arg\min_{w} \sum_{i=1}^{n} (w^\top \mathbf{x}_i - y_i)^2 + \lambda \|w\|_2^2$$

$$X = \begin{bmatrix} | & & | \\ x_1 & \cdots & x_n \\ | & & | \end{bmatrix}$$

Closed-form solution exists, i.e.,

$$\hat{w} = (XX^\top + \lambda I)^{-1} XY$$

$$Y = \begin{bmatrix} y_1 \\ \vdots \\ y_n \end{bmatrix}$$

Linear regression revisited

$$\boxed{\text{Claim: } \hat{w} = (XX^\top + \lambda I)^{-1} XY \in \text{Span}(X)}$$

$$\hat{w} = \sum_{i=1}^{n} \alpha_i x_i$$

Linear regression revisited

$\boxed{\text{Claim: } \hat{w} = (XX^\top + \lambda I)^{-1} XY \in \text{Span}(X)}$

An intuitive proof: GD (or SGD)

Linear regression revisited

$$\boxed{\text{Claim: } \hat{w} = (XX^\top + \lambda I)^{-1}XY \in \text{Span}(X)}$$

An intuitive proof: GD (or SGD)

$$w_0 = \mathbf{0}, w^{t+1} = w^t - \eta\left[\underbrace{\sum_{i=1}^{n}(\mathbf{x}_i^\top w^t - y_i)\mathbf{x}_i}_{(+)} + \lambda w^t\right]$$

$w^t \in \text{span}(X)$ $\qquad$ $(+) \in \text{span}(X)$

A new perspective of linear regression

Since we know optimal solution lives in span(X), we can re-parameterize

$$w = \sum_{i=1}^{n} \alpha_i \mathbf{x}_i = X\alpha, \ \alpha_i \in \mathbb{R}, \forall i$$

We will learn α instead

A new perspective of linear regression

Since we know optimal solution lives in span(X), we can re-parameterize

$$w = \sum_{i=1}^{n} \alpha_i \mathbf{x}_i = X\alpha, \ \ \alpha_i \in \mathbb{R}, \forall i$$

$$w = \sum_{i=1}^{n} \alpha_i x_i$$

$$\arg\min_{w} \sum_{i=1}^{n} \left\| X^\top w - Y \right\|_2^2 + \lambda \|w\|_2^2$$

Original formulation

A new perspective of linear regression

Since we know optimal solution lives in span(X), we can re-parameterize

$$w = \sum_{i=1}^{n} \alpha_i \mathbf{x}_i = X\alpha, \quad \alpha_i \in \mathbb{R}, \forall i$$

$$\arg\min_{w} \sum_{i=1}^{n} \left\| X^\top w - Y \right\|_2^2 + \lambda \|w\|_2^2 \quad \Longrightarrow \quad \arg\min_{\alpha} \left\| X^\top X\alpha - Y \right\|_2^2 + \|X\alpha\|_2^2$$

Original formulation

New formulation w/ α as our variables

A new perspective of linear regression

n # of data points

$$\arg\min_{\alpha} \| X^\top X\alpha - Y \|_2^2 + \lambda \|X\alpha\|_2^2$$

$\alpha \in \mathbb{R}^n$

$w = X\alpha$

A new perspective of linear regression

$$\arg\min_{\alpha} \left\| X^\top X\alpha - Y \right\|_2^2 + \lambda\|X\alpha\|_2^2$$

Solution:

$$\alpha = \overbrace{\left(X^\top X + \lambda I\right)}^{n\times n}{}^{-1} \overset{\mathbb{R}^n}{Y} \in \mathbb{R}^n$$

① $\alpha \in \mathbb{R}^n$

② $X = \begin{bmatrix} | & & | \\ x_1 & \cdots & x_n \\ | & & | \end{bmatrix} \in \mathbb{R}^{d\times n}$

$$X^\top X = \begin{bmatrix} - & x_1^\top & - \\ - & x_2^\top & - \\ & \vdots & \\ - & x_i^\top & - \\ & \vdots & \\ - & x_n^\top & - \end{bmatrix} \begin{bmatrix} | & | & & | & & | \\ x_1 & x_2 & \cdots & x_j & \cdots & x_n \\ | & | & & | & & | \end{bmatrix}$$

$X^\top X \in \mathbb{R}^{n\times n}$

$$= \begin{array}{c c} & j \\ i & \begin{bmatrix} & \vdots & \\ \cdots & x_i^\top x_j & \cdots \\ & \vdots & \end{bmatrix} \end{array}_{n\times n}$$

A new perspective of linear regression

$$\arg\min_{\alpha} \left\| X^\top X\alpha - Y \right\|_2^2 + \lambda \|X\alpha\|_2^2$$

Solution:

$$\alpha = \left(X^\top X + \lambda I\right)^{-1} Y \in \mathbb{R}^n$$

$$X^\top X \in \mathbb{R}^{n\times n},\ (X^\top X)_{i,j} = \mathbf{x}_i^\top \mathbf{x}_j = \langle \mathbf{x}_i, \mathbf{x}_j \rangle$$

A new perspective of linear regression

When we make prediction on a test example $\mathbf{x} \in \mathbb{R}^d$, we have:

$$\hat{w}^\top \mathbf{x} = (\sum_{i=1}^{n} \underline{\alpha_i \mathbf{x}_i})^\top \mathbf{x} = \sum_{i=1}^{n} \alpha_i \cdot \langle \mathbf{x}_i, \mathbf{x} \rangle$$

$$\hat{\omega} = \sum_{i=1}^{n} \alpha_i x_i$$

A new perspective of linear regression

When we make prediction on a test example $\mathbf{x} \in \mathbb{R}^d$, we have:

$$\hat{w}^\top \mathbf{x} = (\sum_{i=1}^{n} \alpha_i \mathbf{x}_i)^\top \mathbf{x} = \sum_{i=1}^{n} \alpha_i \cdot \langle \mathbf{x}_i, \mathbf{x} \rangle$$

$$\hat{W} = (\underbrace{XX^T}_{d \times d} + \lambda I)^{-1} X Y$$

Notice a theme here:

Linear regression can be done by just using inner product of features

$\langle \mathbf{x}, \mathbf{z} \rangle, \mathbf{x} \in \mathbb{R}^d, \mathbf{z} \in \mathbb{R}^d$

Outline

1. A new perspective on ridge linear regression

2. Feature mapping and Kernel

3. Kernel trick and demo of kernel regression

Feature mapping

Define $\phi(\mathbf{x}) \in \mathbb{R}^m$ as a feature mapping (often $m > d$)

Feature mapping

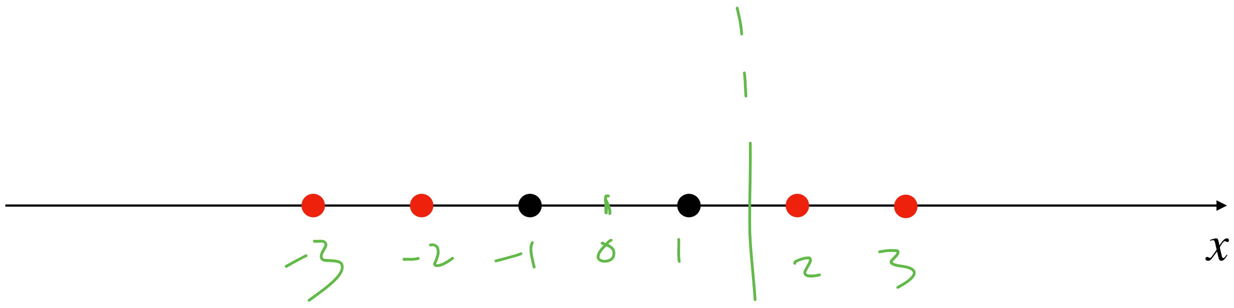
Define $\phi(\mathbf{x}) \in \mathbb{R}^m$ as a feature mapping (often $m > d$)

Ex 1: $\mathbf{x} \in \mathbb{R}$, $\phi(\mathbf{x}) = [x, x^2]^\top \in \mathbb{R}^2$ $= \begin{bmatrix} x \\ x^2 \end{bmatrix}$

Feature mapping

Define $\phi(\mathbf{x}) \in \mathbb{R}^m$ as a feature mapping (often $m > d$)

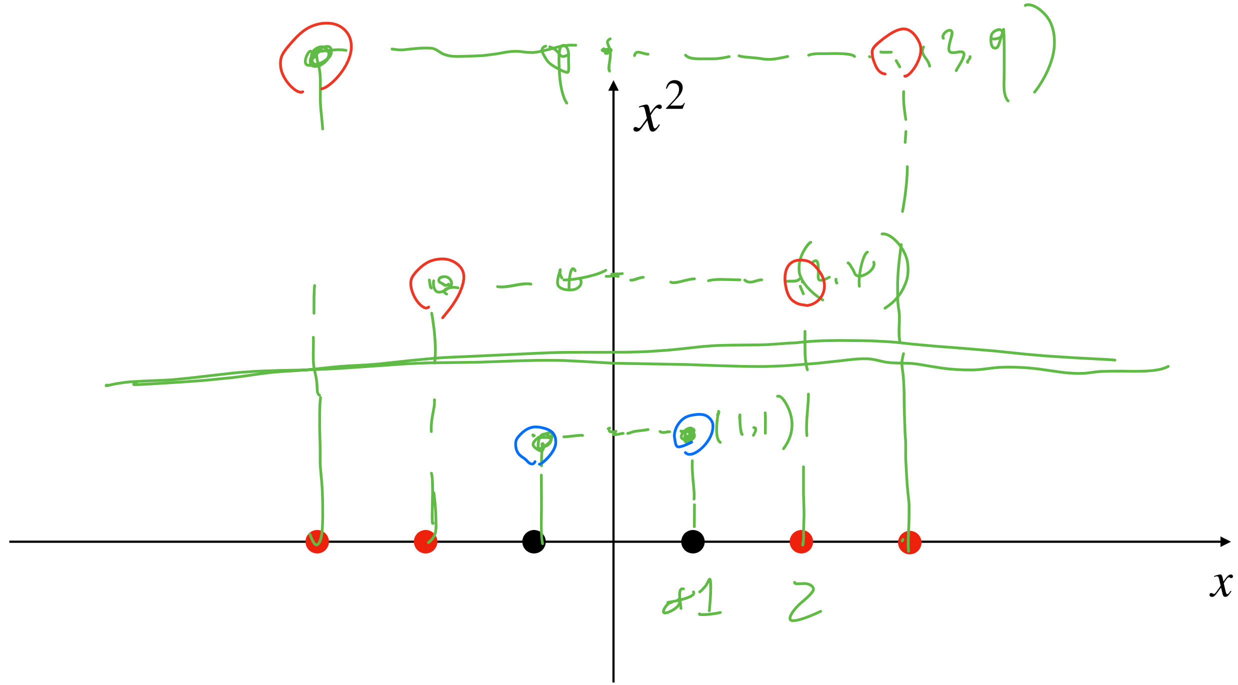
Ex 1: $\mathbf{x} \in \mathbb{R}$, $\phi(\mathbf{x}) = [x, x^2]^\top \in \mathbb{R}^2$



Feature mapping

Define $\phi(\mathbf{x}) \in \mathbb{R}^m$ as a feature mapping (often $m > d$)

Ex 1: $\mathbf{x} \in \mathbb{R}$, $\phi(\mathbf{x}) = [x, x^2]^\top \in \mathbb{R}^2$



Feature mapping

Define $\phi(\mathbf{x}) \in \mathbb{R}^m$ as a feature mapping (often $m > d$)

Ex 2: quadratic
feature mapping ϕ

$$\mathbf{x} = [x_1, x_2]^\top, \quad \in R^2$$

$$\phi(\mathbf{x}) = [1, x_1, x_2, x_1^2, x_2^2, x_1 x_2]^\top$$

Feature mapping

Define $\phi(\mathbf{x}) \in \mathbb{R}^m$ as a feature mapping (often $m > d$)

Ex 2: cubic feature mapping ϕ

$$\mathbf{x} = [x_1, x_2]^\top,$$

$$\phi(\mathbf{x}) = [1, x_1, x_2, x_1^2, x_2^2, x_1 x_2, x_1^3, x_2^3, x_1 x_2^2, x_1^2 x_2]^\top$$

Feature mapping

Define $\phi(\mathbf{x}) \in \mathbb{R}^m$ as a feature mapping (often $m > d$)

Ex 2: cubic feature mapping ϕ

$$\mathbf{x} = [x_1, x_2]^\top,$$

$$\phi(\mathbf{x}) = [1, x_1, x_2, x_1^2, x_2^2, x_1x_2, x_1^3, x_2^3, x_1x_2^2, x_1^2x_2]^\top$$

Q: in general, for $\mathbf{x} \in \mathbb{R}^d$, and a p-th order polynomial feature ϕ, what's the dimension of $\phi(\mathbf{x})$?

$$x = \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{pmatrix} \quad p = 3$$

$$\begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_4 \\ x_1^2 \\ \vdots \\ x_4^2 \\ \vdots \\ x_1^3 \\ \vdots \\ x_4^3 \\ x_1^2 x_2 \\ x_1^2 x_4 \\ \vdots \end{bmatrix}$$

Feature mapping

Define $\phi(\mathbf{x}) \in \mathbb{R}^m$ as a feature mapping (often $m > d$)

Ex 2: cubic feature mapping ϕ

$$\mathbf{x} = [x_1, x_2]^\top,$$

$$\phi(\mathbf{x}) = [1, x_1, x_2, x_1^2, x_2^2, x_1x_2, x_1^3, x_2^3, x_1x_2^2, x_1^2x_2]^\top$$

Q: in general, for $\mathbf{x} \in \mathbb{R}^d$, and a p-th order polynomial feature ϕ, what's the dimension of $\phi(\mathbf{x})$?

at least $\binom{d}{p} \approx d^p$

$d = 100$
$p = 10$

Feature mapping

Define $\phi(\mathbf{x}) \in \mathbb{R}^m$ as a feature mapping (often $m > d$)

Ex 2: cubic feature mapping ϕ

$$\mathbf{x} = [x_1, x_2]^\top,$$

$$\phi(\mathbf{x}) = [1, x_1, x_2, x_1^2, x_2^2, x_1x_2, x_1^3, x_2^3, x_1x_2^2, x_1^2x_2]^\top$$

Q: in general, for $\mathbf{x} \in \mathbb{R}^d$, and a p-th order polynomial feature ϕ, what's the dimension of $\phi(\mathbf{x})$?

at least $\binom{d}{p}$

Dim of $\phi(\mathbf{x})$ can be very large!

Fit linear functions in the high-dim feature space

The feature mapping $\phi(\mathbf{x}) \in \mathbb{R}^m$ allows us to perform linear regression in the ϕ space

Fit linear functions in the high-dim feature space

The feature mapping $\phi(\mathbf{x}) \in \mathbb{R}^m$ allows us to perform linear regression in the ϕ space

Ex: cubic feature mapping ϕ

$$\mathbf{x} = [x_1, x_2]^\top, \;\; \phi(\mathbf{x}) = [1, x_1, x_2, x_1^2, x_2^2, x_1 x_2, x_1^3, x_2^3, x_1 x_2^2, x_1^2 x_2]^\top$$

$$w^T \phi(x) = w_0 + w_1 x_1 \cdots w x_1^2 + w' x_2^2$$

$$+ \cdots w'' x_1^3$$

$$+ \cdots w x_1^2 x_2$$

Fit linear functions in the high-dim feature space

The feature mapping $\phi(\mathbf{x}) \in \mathbb{R}^m$ allows us to perform linear regression in the ϕ space

Ex: cubic feature mapping ϕ

$$\mathbf{x} = [x_1, x_2]^\top, \;\; \phi(\mathbf{x}) = [1, x_1, x_2, x_1^2, x_2^2, x_1x_2, x_1^3, x_2^3, x_1x_2^2, x_1^2x_2]^\top$$

$w^\top \phi(\mathbf{x})$ now can represent a 3-order polynomials!

Fit linear functions in the high-dim feature space

The feature mapping $\phi(\mathbf{x}) \in \mathbb{R}^m$ allows us to perform linear regression in the ϕ space

Ex: cubic feature mapping ϕ

$$\mathbf{x} = [x_1, x_2]^\top, \ \ \phi(\mathbf{x}) = [1, x_1, x_2, x_1^2, x_2^2, x_1x_2, x_1^3, x_2^3, x_1x_2^2, x_1^2x_2]^\top$$

$w^\top \phi(\mathbf{x})$ now can represent a 3-order polynomials!

To fit a 3-order polynomial in $\mathbf{x}$, we can instead do linear regression in $\phi(\mathbf{x})$

$$(x_1 \cdots x_n) \rightarrow \begin{bmatrix} | & & | \\ \phi(x_1) & \cdots & \phi(x_n) \\ | & & | \end{bmatrix}$$

Fit linear functions in the high-dim feature space

Perform linear regression in ϕ space, i.e.,

$$\min_{w} \sum_{i=1}^{n} \left(w^\top \phi(\mathbf{x}_i) - y_i\right)^2 + \lambda \|w\|_2^2$$

Fit linear functions in the high-dim feature space

Perform linear regression in ϕ space, i.e.,

$$\min_w \sum_{i=1}^n \left(w^\top \phi(\mathbf{x}_i) - y_i\right)^2 + \lambda \|w\|_2^2$$

Linear in ϕ, but high-order poly in $\mathbf{x}$

Fit linear functions in the high-dim feature space

Perform linear regression in ϕ space, i.e.,

$$\min_{w} \sum_{i=1}^{n} \left(w^\top \phi(\mathbf{x}_i) - y_i\right)^2 + \lambda \|w\|_2^2$$

Linear in ϕ, but high-order poly in $\mathbf{x}$

What is the potential problem of doing this?

$\phi(x)$

d^p

d = dim of x

p - ~~ord~~ order poly

Fit linear functions in the high-dim feature space

Perform linear regression in ϕ space, i.e.,

$$\min_{w} \sum_{i=1}^{n} \left(w^\top \phi(\mathbf{x}_i) - y_i\right)^2 + \lambda \|w\|_2^2$$

Linear in ϕ, but high-order poly in $\mathbf{x}$

What is the potential problem of doing this?

This is where the new perspective of linear regression and kernels come to rescue!

Kernel

Kernel $k(\mathbf{x}, \mathbf{z}) \in \mathbb{R}$

$x^T z$

Kernel

Kernel $k(\mathbf{x}, \mathbf{z})$

A valid kernel is a kernel such that $\exists \phi, k(\mathbf{x}, \mathbf{z}) = \phi(\mathbf{x})^\top \phi(\mathbf{z}), \forall \mathbf{x}, \mathbf{z}$

$$K(x, z) = x^T z$$

$$= \langle x, z \rangle$$

ϕ: identically

Kernel

Kernel $k(\mathbf{x}, \mathbf{z})$

A valid kernel is a kernel such that $\exists \phi, k(\mathbf{x}, \mathbf{z}) = \phi(\mathbf{x})^\top \phi(\mathbf{z}), \forall \mathbf{x}, \mathbf{z}$

Ex: quadratic kernel

$$k(\mathbf{x}, \mathbf{z}) = (\mathbf{x}^\top \mathbf{z} + 1)^2$$

$\phi(x)$??

$\phi(x)^T \phi(z) = (x^T z + 1)^2$

Kernel

Kernel $k(\mathbf{x}, \mathbf{z})$

A valid kernel is a kernel such that $\exists \phi,\ k(\mathbf{x}, \mathbf{z}) = \phi(\mathbf{x})^\top \phi(\mathbf{z}), \forall \mathbf{x}, \mathbf{z}$

Ex: quadratic kernel

$$k(\mathbf{x}, \mathbf{z}) = (\mathbf{x}^\top \mathbf{z} + 1)^2$$

$$\phi(\mathbf{x}) = [1, \sqrt{2}x_1, \sqrt{2}x_2, x_1^2, x_2^2, \sqrt{2}x_1x_2]^\top$$

$$\phi(x)^T \phi(z) = (x^T z + 1)^2$$

Kernel

Kernel $k(\mathbf{x}, \mathbf{z})$

A valid kernel is a kernel such that $\exists \phi, k(\mathbf{x}, \mathbf{z}) = \phi(\mathbf{x})^\top \phi(\mathbf{z}), \forall \mathbf{x}, \mathbf{z}$

Ex: quadratic kernel

$x \in \mathbb{R}^d$

$$k(\mathbf{x}, \mathbf{z}) = (\mathbf{x}^\top \mathbf{z} + 1)^2$$

$$\phi(\mathbf{x}) = [1, \sqrt{2}x_1, \sqrt{2}x_2, x_1^2, x_2^2, \sqrt{2}x_1x_2]^\top$$

$(x^\top z + 1)^2 \leftarrow O(d)$

Q: what's the computation of $k(\mathbf{x}, \mathbf{z})$?

Kernel

Kernel $k(\mathbf{x}, \mathbf{z})$

A valid kernel is a kernel such that $\exists \phi, k(\mathbf{x}, \mathbf{z}) = \phi(\mathbf{x})^\top \phi(\mathbf{z}), \forall \mathbf{x}, \mathbf{z}$

Ex: quadratic kernel

$$k(\mathbf{x}, \mathbf{z}) = (\mathbf{x}^\top \mathbf{z} + 1)^2$$

$$\phi(\mathbf{x}) = [1, \sqrt{2}x_1, \sqrt{2}x_2, x_1^2, x_2^2, \sqrt{2}x_1x_2]^\top$$

Q: what's the computation of $k(\mathbf{x}, \mathbf{z})$?

Q: what's the computation of $\phi(\mathbf{x})^\top \phi(\mathbf{z})$?

$k(x, z)$

$= \phi(x)^\top \phi(z)$

$O(d^2)$

$\phi(x) \in \mathbb{R}^{d^2}$

Kernel

$\phi(x) \in R^{d^3}$

Kernel $k(\mathbf{x}, \mathbf{z})$ Computation $(\phi(x)^\top \phi(z)) \leftarrow O(d^3)$??

A valid kernel is a kernel such that $\exists \phi, k(\mathbf{x}, \mathbf{z}) = \phi(\mathbf{x})^\top \phi(\mathbf{z}), \forall \mathbf{x}, \mathbf{z}$

Ex: quadratic kernel

$$k(\mathbf{x}, \mathbf{z}) = (\mathbf{x}^\top \mathbf{z} + 1)^2$$

$$\phi(\mathbf{x}) = [1, \sqrt{2}x_1, \sqrt{2}x_2, x_1^2, x_2^2, \sqrt{2}x_1x_2]^\top$$

Q: what's the computation of $k(\mathbf{x}, \mathbf{z})$?

Q: what's the computation of $\phi(\mathbf{x})^\top \phi(\mathbf{z})$?

computation $O(d)$

Ex: cubic feature mapping ϕ

$$k(\mathbf{x}, \mathbf{z}) = (\mathbf{x}^\top \mathbf{z} + 1)^3$$

$$\phi(x) = \begin{bmatrix} 1 \\ \sqrt{3}x_1 \\ \sqrt{3}x_2 \\ \vdots \\ x_d \\ x_1^2 \\ \vdots \\ x_d^2 \\ \vdots \\ x_1^3 \\ \vdots \\ x_d^3 \end{bmatrix}$$

$\phi(x)^\top \phi(z)$

$= k(x, z)$

Kernel

Kernel $k(\mathbf{x}, \mathbf{z})$

A valid kernel is a kernel such that $\exists \phi, k(\mathbf{x}, \mathbf{z}) = \phi(\mathbf{x})^\top \phi(\mathbf{z}), \forall \mathbf{x}, \mathbf{z}$

Ex: quadratic kernel

$$k(\mathbf{x}, \mathbf{z}) = (\mathbf{x}^\top \mathbf{z} + 1)^2$$

$$\phi(\mathbf{x}) = [1, \sqrt{2}x_1, \sqrt{2}x_2, x_1^2, x_2^2, \sqrt{2}x_1x_2]^\top$$

Q: what's the computation of $k(\mathbf{x}, \mathbf{z})$?

Q: what's the computation of $\phi(\mathbf{x})^\top \phi(\mathbf{z})$?

Ex: cubic feature mapping ϕ

$$k(\mathbf{x}, \mathbf{z}) = (\mathbf{x}^\top \mathbf{z} + 1)^3$$

$p = 20$

Generalizing to p-th order polynomials:

$$k(\mathbf{x}, \mathbf{z}) = (\mathbf{x}^\top \mathbf{z} + 1)^p$$

$O(d)$

$\phi(x) \in R^{(d^p)}$

$\phi(x)^\top \phi(z) \approx O(d^p)$

$\phi(x)^T \phi(z)$

Kernel

pdf of Gaussian

Gaussian Kernel: $k(\mathbf{x}, \mathbf{z}) = \exp\left(-\|\mathbf{x} - \mathbf{z}\|_2^2/\sigma^2\right)$

The mapping $\phi(\mathbf{x})$ is infinite-dimensional

$\phi(x) \in R^{\infty}$

Kernel

Gaussian Kernel: $k(\mathbf{x}, \mathbf{z}) = \exp\left(-\|\mathbf{x} - \mathbf{z}\|_2^2/\sigma^2\right)$

The mapping $\phi(\mathbf{x})$ is infinite-dimensional

$d=1$

Ex: $\mathbf{x} \in \mathbb{R}$, the mapping $\phi(\mathbf{x})$:

i-th polynomial

$$\phi(\mathbf{x}) = \left[\ldots, \underbrace{\frac{1}{\sqrt{i!}}\exp\left(-\frac{x^2}{2\sigma^2}\right)}_{\text{i-th element}} x^i, \ldots\right]^\top \in \mathbb{R}^\infty$$

Kernel

Gaussian Kernel: $k(\mathbf{x}, \mathbf{z}) = \exp\left(-\|\mathbf{x} - \mathbf{z}\|_2^2/\sigma^2\right)$

2. Linear function $w^\top \phi(\mathbf{x})$ can model any indefinitely differentiable function f

exp(x) sin(x)

Kernel

Gaussian Kernel: $k(\mathbf{x}, \mathbf{z}) = \exp\left(-\|\mathbf{x} - \mathbf{z}\|_2^2/\sigma^2\right)$

2. Linear function $w^\top \phi(\mathbf{x})$ can model any indefinitely differentiable function f

Why? ϕ contains all polynomials, and f can be written as
an infinite Taylor series..

$x = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$

$z = \begin{bmatrix} z_1 \\ z_2 \end{bmatrix}$

Summary so far

1. Feature mapping $\phi(\mathbf{x})$ lifts $\mathbf{x}$ into high-dimensional space (e.g., high-order polynomials)

Summary so far

1. Feature mapping $\phi(\mathbf{x})$ lifts $\mathbf{x}$ into high-dimensional space (e.g., high-order polynomials)

2. A kernel $k(\mathbf{x}, \mathbf{z})$ is a symmetric function, such that there exists a ϕ, so that

$$k(\mathbf{x}, \mathbf{z}) = \phi(\mathbf{x})^\top \phi(\mathbf{z})$$

Summary so far

1. Feature mapping $\phi(\mathbf{x})$ lifts $\mathbf{x}$ into high-dimensional space (e.g., high-order polynomials)

2. A kernel $k(\mathbf{x}, \mathbf{z})$ is a symmetric function, such that there exists a ϕ, so that

$$k(\mathbf{x}, \mathbf{z}) = \phi(\mathbf{x})^\top \phi(\mathbf{z})$$

3. Kernel allows us to compute $\langle \phi(\mathbf{x}), \phi(\mathbf{z}) \rangle$ without ever explicitly computing ϕ

($k(\mathbf{x}, \mathbf{z})$ is easy to compute but $\langle \phi(\mathbf{x}), \phi(\mathbf{z}) \rangle$ is hard to compute)