# MLEA: Machine Learning

## Kernels methods

Réda DEHAK
reda.dehak@lrde.epita.fr

EPITA

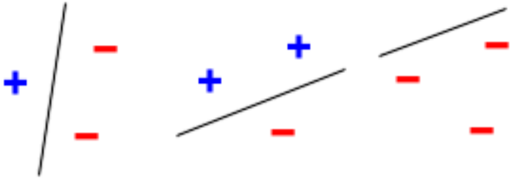
October 23, 2017

1. VC Dimension
2. Feature Space
3. The Kernel Trick
4. Using SVM on a Toy Problem

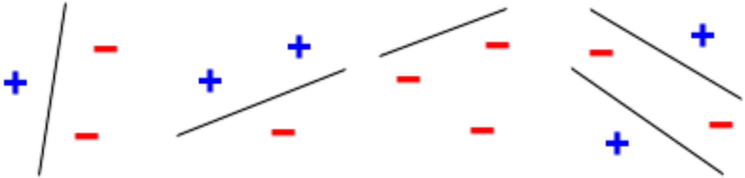
# Vapnik-Chervonenkis dimension

- measure of the capacity of a statistical classification algorithm (how complicated it can be)
- cardinality of the largest set of points that the algorithm can shatter, for each possible combination of labels
- Core concept in Vapnik-Chervonenkis theory

# Example



# Example



VC dimension for a linear classifier handling data of dimension N: N + 1

# VC dim-based model selection

## Definition

TestingSet Error $\leq$ TrainingSet Error $+ \sqrt{\frac{h(\log(2R/h)+1)-log(\mu/4)}{R}}$
With $h$ the VC dimension of the classification model
$R$ the size of the trainingset
True with probability $1-\mu$

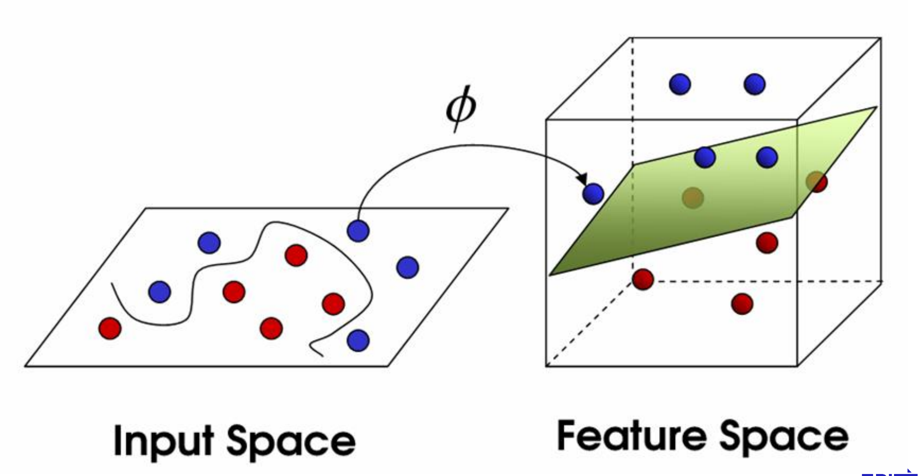
1. VC Dimension
2. Feature Space
3. The Kernel Trick
4. Using SVM on a Toy Problem

# Separability of patterns

**Cover's Theorem (1965)**

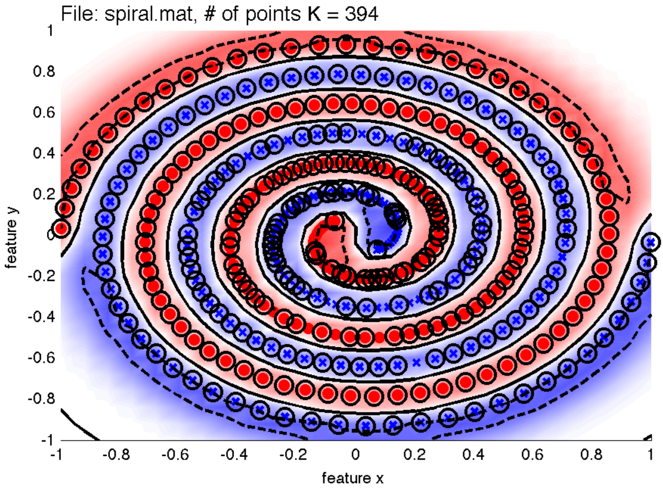
A complex pattern-classification problem cast in a high-dimensional space nonlinearly is more likely to be linearly separable than in low dimensional space

# Feature Space projection



# the two spirals

Separated by a hyperplane in feature space (gaussian kernels)



# Feature and Feature Space

**Definition**

A function $\phi_i : \chi \to \mathbb{R}$ that maps each object $x \in \chi$ to a real value $\phi_i(x)$ is called a feature.
To be valid, $\forall x \in \chi, \phi_i(x)$ should be finite.

# Feature and Feature Space

## Definition

A function $\phi_i : \chi \rightarrow \mathbb{R}$ that maps each object $x \in \chi$ to a real value $\phi_i(x)$ is called a feature.
To be valid, $\forall x \in \chi, \phi_i(x)$ should be finite.
Combining n features $\phi_1 ... \phi_n$ results in a feature mapping $\phi : \chi \rightarrow \kappa \subseteq \mathbb{R}^n$ and the space $\kappa$ is called a feature space.

1 VC Dimension

2 Feature Space

3 The Kernel Trick

4 Using SVM on a Toy Problem

# Kernel

## Definition

Suppose we are given a feature mapping $\phi : \chi \rightarrow \kappa \subseteq \mathbb{R}^n$
The corresponding kernel is the inner product function
$K : \chi \times \chi \rightarrow \mathbb{R}$ such as:

$$K(x_i, x_j) =< \phi(x_i), \phi(x_j) >$$

# The kernel trick for SVM

- Replace dot product by kernel functions
- the dot product is done in the feature space

# SVM reformulation

- Minimize $Q(\alpha) = \sum \alpha_i - \frac{1}{2} \sum \sum \alpha_i \alpha_j y_i y_j \langle x_i, x_j \rangle$
  $\Rightarrow$

# SVM reformulation

- Minimize $Q(\alpha) = \sum \alpha_i - \frac{1}{2} \sum \sum \alpha_i \alpha_j y_i y_j \langle x_i, x_j \rangle$
  $\Rightarrow$Minimize $Q(\alpha) = \sum \alpha_i - \frac{1}{2} \sum \sum \alpha_i \alpha_j y_i y_j K(x_i, x_j)$
- Decision function: $f(x) = w^T x + b$

# SVM reformulation

- Minimize $Q(\alpha) = \sum \alpha_i - \frac{1}{2} \sum \sum \alpha_i \alpha_j y_i y_j \langle x_i, x_j \rangle$
  $\Rightarrow$Minimize $Q(\alpha) = \sum \alpha_i - \frac{1}{2} \sum \sum \alpha_i \alpha_j y_i y_j K(x_i, x_j)$
- Decision function: $f(x) = w^T x + b = \sum \alpha_i y_i \langle x_i, x \rangle + b$

# SVM reformulation

- Minimize $Q(\alpha) = \sum \alpha_i - \frac{1}{2} \sum \sum \alpha_i \alpha_j y_i y_j \langle x_i, x_j \rangle$
  $\Rightarrow$Minimize $Q(\alpha) = \sum \alpha_i - \frac{1}{2} \sum \sum \alpha_i \alpha_j y_i y_j K(x_i, x_j)$
- Decision function: $f(x) = w^T x + b = \sum \alpha_i y_i \langle x_i, x \rangle + b$
  $\Rightarrow$

# SVM reformulation

- Minimize $Q(\alpha) = \sum \alpha_i - \frac{1}{2} \sum \sum \alpha_i \alpha_j y_i y_j \langle x_i, \, x_j \rangle$
  $\Rightarrow$Minimize $Q(\alpha) = \sum \alpha_i - \frac{1}{2} \sum \sum \alpha_i \alpha_j y_i y_j K(x_i, x_j)$
- Decision function: $f(x) = w^T x + b = \sum \alpha_i y_i \langle x_i, \, x \rangle + b$
  $\Rightarrow \; f(x) = \sum \alpha_i y_i K(x_i, x) + b$

# Example: XOR problem revisited

4 points dataset :

| **Data** | **Class** |
|---|---|
| $(-1,-1)$ | -1 |
| $(-1,+1)$ | $+1$ |
| $(+1,-1)$ | $+1$ |
| $(+1,+1)$ | -1 |

Find the Large margin SVM corresponding to the dataset using the polynomial Kernel $K(x,y) = (1 + x^t y)^2$.

# Example: XOR problem revisited

4 points dataset :

| **Data** | **Class** |
|---|---|
| $(-1,-1)$ | -1 |
| $(-1,+1)$ | $+1$ |
| $(+1,-1)$ | $+1$ |
| $(+1,+1)$ | -1 |

Find the Large margin SVM corresponding to the dataset using the polynomial Kernel $K(x,y) = (1 + x^t y)^2$.
Consider $x = [x_1, x_2]^t$ and $y = [y_1, y_2]^t$.
The feature mapping is quite explicit:

$$K(x,y) = 1 + x_1^2 y_1^2 + 2x_1 x_2 y_1 y_2 + x_2^2 y_2^2 + 2x_1 y_1 + 2x_2 y_2$$

The projection of $x$ in the feature space is :

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

# Common Kernels

- Linear : $K(x, y) = x^T y$
- Polynomial : $K(x, y) = (x^T y + 1)^d$
- Laplacian RBF : $exp\,(-\gamma\|x - y\|) = exp\left(-\frac{\|x-y\|}{\sigma^2}\right)$
- RBF : $K(x, y) = exp(-\gamma\|x - y\|^2) = exp(-\frac{\|x-y\|^2}{\sigma^2})$

# Common Kernels

- Linear : $K(x,y) = x^T y$
- Polynomial : $K(x,y) = (x^T y + 1)^d$
- Laplacian RBF : $exp\left(-\gamma\|x-y\|\right) = exp\left(-\frac{\|x-y\|}{\sigma^2}\right)$
- RBF : $K(x,y) = exp(-\gamma\|x-y\|^2) = exp(-\frac{\|x-y\|^2}{\sigma^2})$
- RTFL...

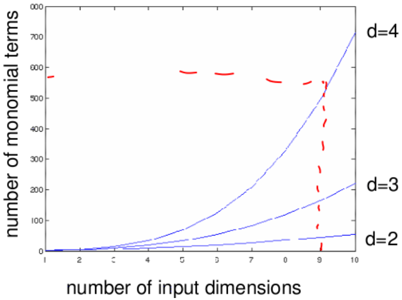
# Exercise: Kernel VC Dimension

Give the VC dimension of a SVM using:

- Linear Kernel
- RBF Kernel
- Polynomial Kernel

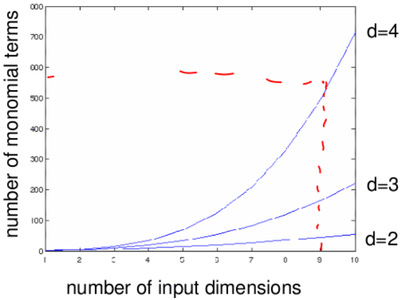
# Polynomial Kernel Mapping

$$dim = \begin{pmatrix} d+m \\ d \end{pmatrix} = \frac{(d+m)!}{d!m!}$$



# Polynomial Kernel Mapping

$$dim = \binom{d+m}{d} = \frac{(d+m)!}{d!m!}$$




- implicit mapping not necessary
- allow to project the data to infinite feature space

# Positive semi-definite Kernel Function

## Definition

A kernel function $K$ is said to be positive semi-definite if

$$\forall (x_1, .., x_n) \in \chi^n, \forall (c_1, .., c_n) \in \mathbb{R}^n \neq 0$$

$$\sum_{i,j} K(x_i, x_j) c_i c_j \geq 0$$

# Gram matrix

## Definition

Given a kernel K: $\chi \times \chi \rightarrow \mathbb{R}$ and a set $\mathbf{x} = (x_1, ..., x_m) \in \chi^m$ of m objects in $\chi$, we call the $m \times m$ matrix $\mathbf{G}$ with:

$$\mathbf{G_{ij}} = K(x_i, x_j)$$

the Gram matrix of K at x.

# Lemma

## Definition

A $N \times N$ matrix $\mathcal{M}$ is said to be positive semi-definite if and only if:

$$\forall v \in \mathbb{R}^N, v^T \mathcal{M} v \geq 0$$

## Lemma

Given a positive semi-definite kernel function $K$

$$\forall \mathbf{x} = (x_1, ..., x_m) \in \chi^m$$

the corresponding Gram Matrix **G** will be positive semi-definite.

# Mercer Theorem

**Theorem**

If K is a positive semi-definite kernel then there exists a function $\phi$ such that:

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

# Kernel Properties

Let $K_1$ and $K_2$ be kernels over $\chi$, $a \geq 0$, $B$ a symmetric positive semi-definite matrix, and $K_3$ a kernel over $\mathbb{R}^n$. Prove the following functions are kernels:

- $K(x, y) = K_1(x, y) + K_2(x, y)$

# Kernel Properties

Let $K_1$ and $K_2$ be kernels over $\chi$, $a \geq 0$, $B$ a symmetric positive semi-definite matrix, and $K_3$ a kernel over $\mathbb{R}^n$. Prove the following functions are kernels:

- $K(x, y) = K_1(x, y) + K_2(x, y)$
- $K(x, y) = a\, K_1(x, y)$

# Kernel Properties

Let $K_1$ and $K_2$ be kernels over $\chi$, $a \geq 0$, $B$ a symmetric positive semi-definite matrix, and $K_3$ a kernel over $\mathbb{R}^n$. Prove the following functions are kernels:

- $K(x, y) = K_1(x, y) + K_2(x, y)$
- $K(x, y) = a\,K_1(x, y)$
- $K(x, y) = K_1(x, y)K_2(x, y)$

# Kernel Properties

Let $K_1$ and $K_2$ be kernels over $\chi$, $a \geq 0$, $B$ a symmetric positive semi-definite matrix, and $K_3$ a kernel over $\mathbb{R}^n$. Prove the following functions are kernels:

- $K(x, y) = K_1(x, y) + K_2(x, y)$
- $K(x, y) = a\, K_1(x, y)$
- $K(x, y) = K_1(x, y)K_2(x, y)$
- $K(x, y) = f(x)f(y)$

# Kernel Properties

Let $K_1$ and $K_2$ be kernels over $\chi$, $a \geq 0$, $B$ a symmetric positive semi-definite matrix, and $K_3$ a kernel over $\mathbb{R}^n$. Prove the following functions are kernels:

- $K(x, y) = K_1(x, y) + K_2(x, y)$
- $K(x, y) = a\,K_1(x, y)$
- $K(x, y) = K_1(x, y)K_2(x, y)$
- $K(x, y) = f(x)f(y)$
- $K(x, y) = K_3(\phi(x), \phi(y))$

# Kernel Properties

Let $K_1$ and $K_2$ be kernels over $\chi$, $a \geq 0$, $B$ a symmetric positive semi-definite matrix, and $K_3$ a kernel over $\mathbb{R}^n$. Prove the following functions are kernels:

- $K(x, y) = K_1(x, y) + K_2(x, y)$
- $K(x, y) = a\, K_1(x, y)$
- $K(x, y) = K_1(x, y) K_2(x, y)$
- $K(x, y) = f(x) f(y)$
- $K(x, y) = K_3(\phi(x), \phi(y))$
- $K(x, y) = x^t B y$

# Kernel Properties

Let $K_1$ and $K_2$ be kernels over $\chi$, $a \geq 0$, $B$ a symmetric positive semi-definite matrix, and $K_3$ a kernel over $\mathbb{R}^n$. Prove the following functions are kernels:

- $K(x, y) = K_1(x, y) + K_2(x, y)$
- $K(x, y) = a\, K_1(x, y)$
- $K(x, y) = K_1(x, y) K_2(x, y)$
- $K(x, y) = f(x) f(y)$
- $K(x, y) = K_3(\phi(x), \phi(y))$
- $K(x, y) = x^t B y$
- $K(x, y) = \frac{K(x,y)}{\sqrt{K(x,x)K(y,y)}}$

# Kernel Properties

Let $K_1$ and $K_2$ be kernels over $\chi$, $a \geq 0$, $B$ a symmetric positive semi-definite matrix, and $K_3$ a kernel over $\mathbb{R}^n$. Prove the following functions are kernels:

- $K(x, y) = K_1(x, y) + K_2(x, y)$
- $K(x, y) = a\, K_1(x, y)$
- $K(x, y) = K_1(x, y)K_2(x, y)$
- $K(x, y) = f(x)f(y)$
- $K(x, y) = K_3(\phi(x), \phi(y))$
- $K(x, y) = x^t By$
- $K(x, y) = \frac{K(x,y)}{\sqrt{K(x,x)K(y,y)}}$

# Kernel Properties

Prove the following functions are kernels:

- $K(x, y) = p(K_1(x, y))$ with $p(x)$ a polynomial with positive coefficients

# Kernel Properties

Prove the following functions are kernels:

- $K(x, y) = p(K_1(x, y))$ with $p(x)$ a polynomial with positive coefficients
- $K(x, y) = exp(K_1(x, y))$

# Kernel Properties

Prove the following functions are kernels:

- $K(x, y) = p(K_1(x, y))$ with $p(x)$ a polynomial with positive coefficients
- $K(x, y) = exp(K_1(x, y))$
  Help: $exp(x) = \sum_{i=0}^{\infty} \frac{x^i}{i!}$

# Kernel Properties

Prove the following functions are kernels:

- $K(x, y) = p(K_1(x, y))$ with $p(x)$ a polynomial with positive coefficients
- $K(x, y) = exp(K_1(x, y))$
  Help: $exp(x) = \sum_{i=0}^{\infty} \frac{x^i}{i!}$
- $K(x, y) = exp(-\frac{\|x-y\|^2}{2\sigma^2})$

# Kernels in practice

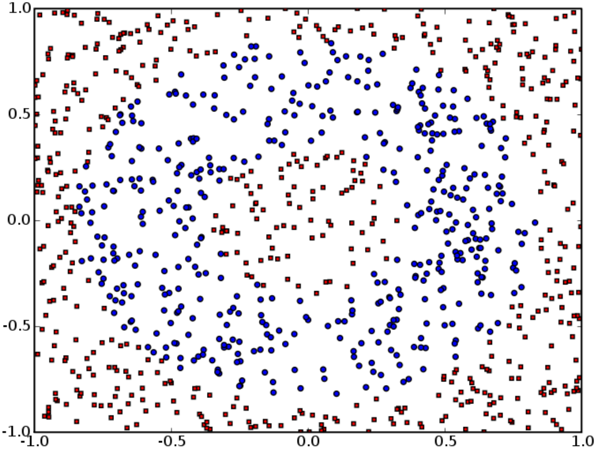
- Data should be centered and reduced
- if the number of feature is large, non linear mapping may not improve performances

# Kernels in practice

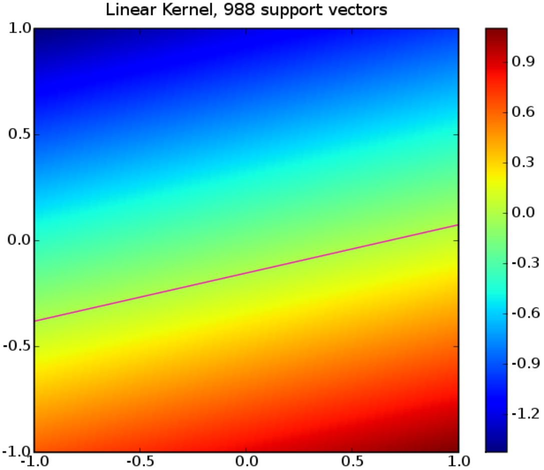
- Data should be centered and reduced
- if the number of feature is large, non linear mapping may not improve performances
- moreover it will be much slower!!

1. VC Dimension
2. Feature Space
3. The Kernel Trick
4. Using SVM on a Toy Problem

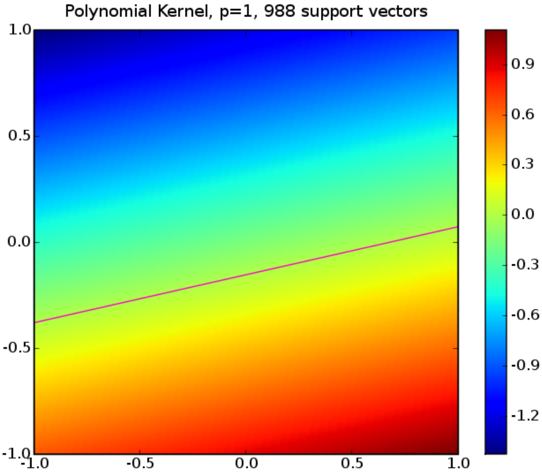
# Donut...



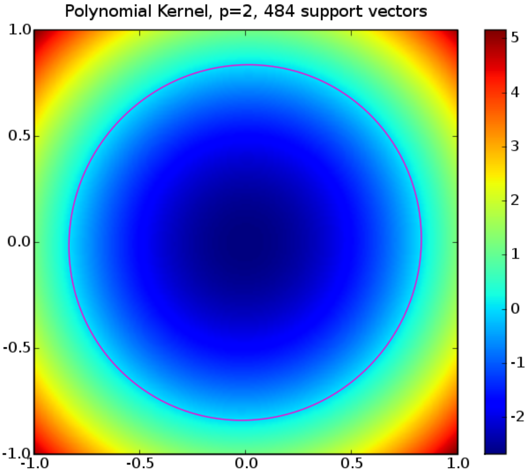
# Linear Kernel



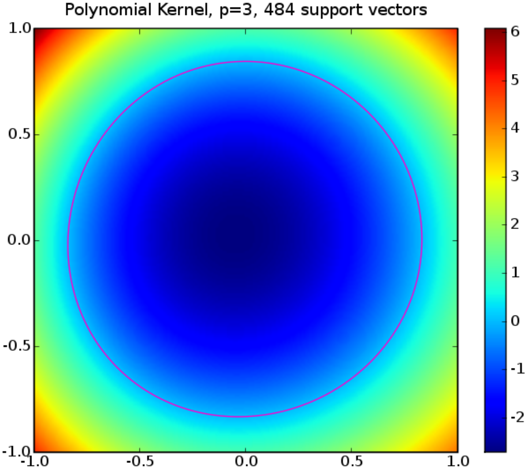
# Polynomial Kernel



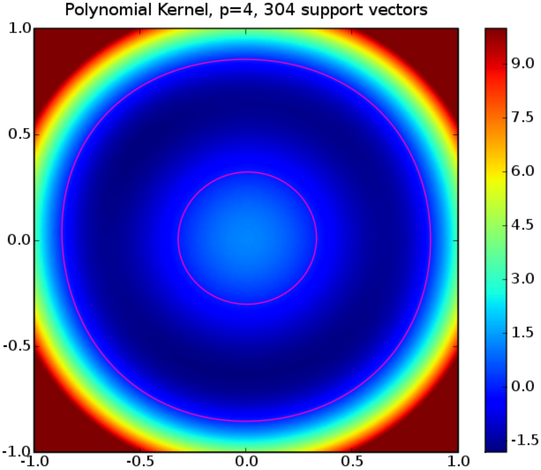
# Polynomial Kernel



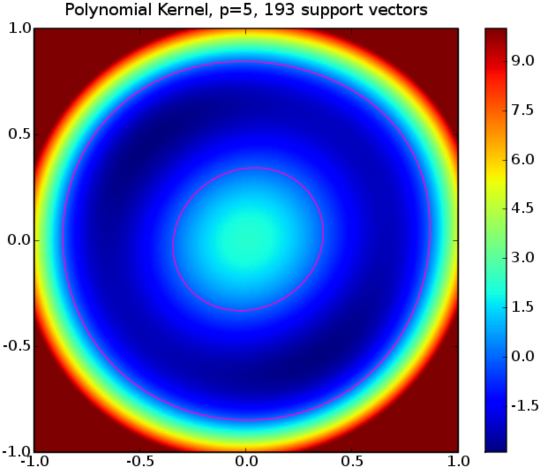
# Polynomial Kernel



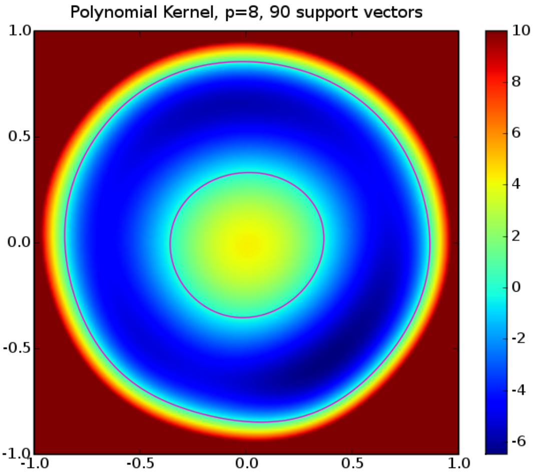
# Polynomial Kernel



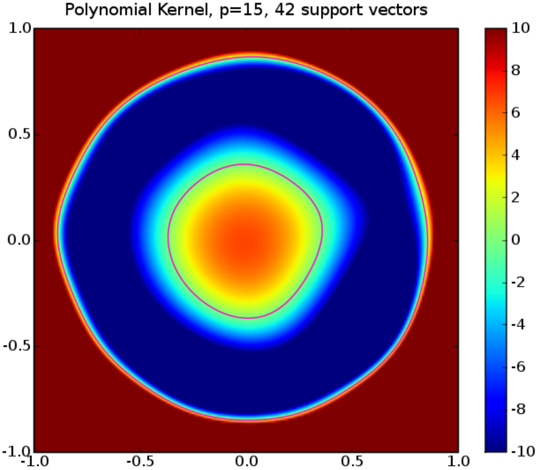
# Polynomial Kernel



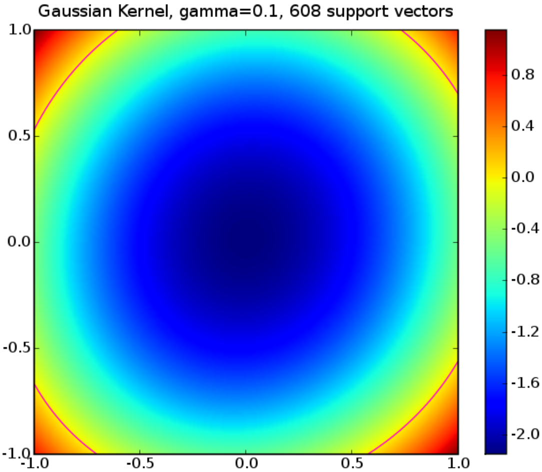
# Polynomial Kernel



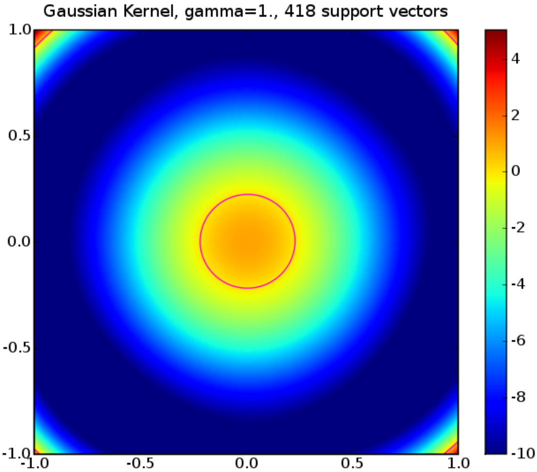
# Polynomial Kernel



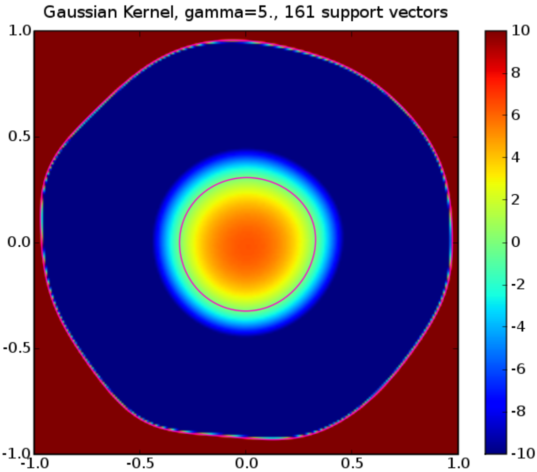
# Gaussian Kernel



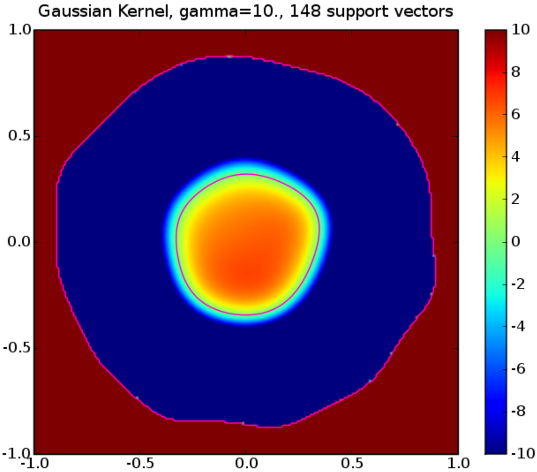
# Gaussian Kernel



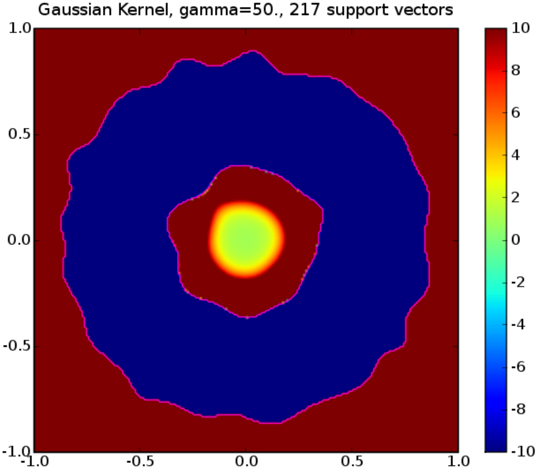
# Gaussian Kernel



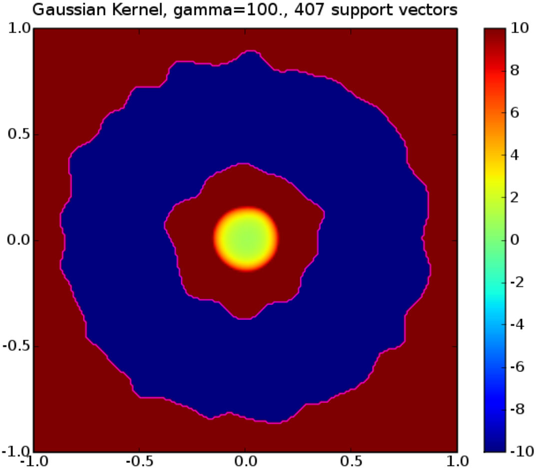
# Gaussian Kernel



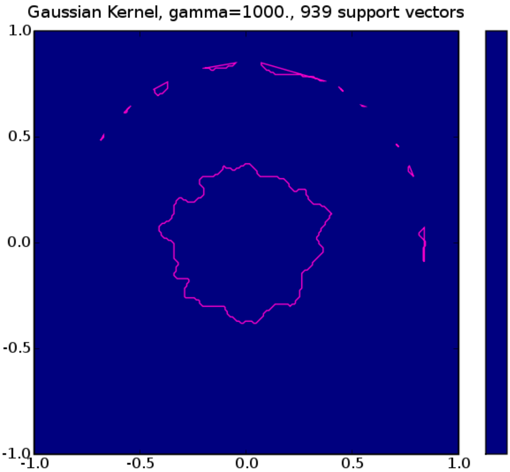
# Gaussian Kernel



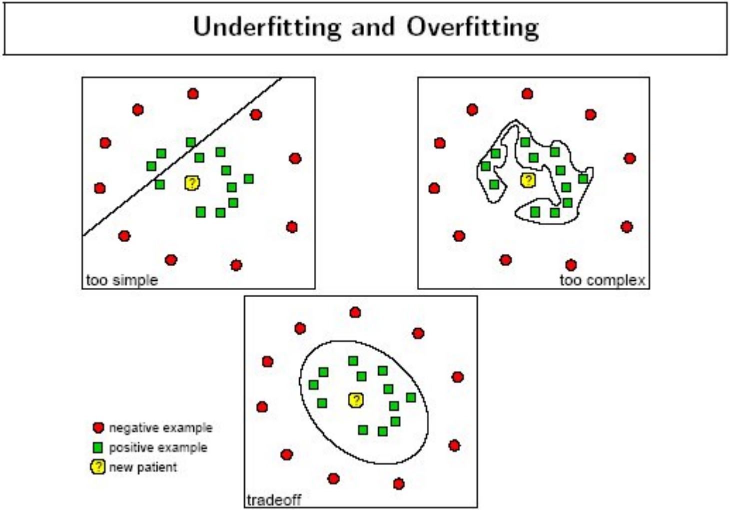
# Gaussian Kernel



# Gaussian Kernel



# SVM parameters should be chosen wisely



# Kernel trick

- Is there other algorithms such that we don't need to know $\phi$?
- Compute $L_2$ distance in feature space

# Kernel trick

- Is there other algorithms such that we don't need to know $\phi$?
- Compute $L_2$ distance in feature space

$$\begin{aligned}
\|\phi(x) - \phi(y)\|^2 &= (\phi(x) - \phi(y))^T(\phi(x) - \phi(y)) \\
&= \phi(x)^T\phi(x) + \phi(y)^T\phi(y) - 2\phi(x)^T\phi(y) \\
&= K(x,x) + K(y,y) - 2K(x,y)
\end{aligned}$$

# Kernel KMeans

# Kernel KMeans

1. Compute Class center

$$\mu_c = \frac{1}{|C_c|} \sum_{x_i \in C_c} \phi(x_i)$$

# Kernel KMeans

1. Compute Class center

$$\mu_c = \frac{1}{|C_c|} \sum_{x_i \in C_c} \phi(x_i)$$

2. Compute distance between example and Class center

$$\begin{aligned}
\|\phi(x) - \mu_c\|^2 &= (\phi(x) - \mu_c)^T(\phi(x) - \mu_c) \\
&= \phi(x)^T\phi(x) - 2\phi(x)^T\mu_c + \mu_c^T\mu_c \\
&= K(x,x) - \frac{2}{|C_c|}\sum_{x_i \in C_c} \phi(x)^T\phi(x_i) \\
&\quad + \frac{1}{|C_c|^2}\sum_{x_i \in C_c}\sum_{x_j \in C_c} \phi(x_i)^T\phi(x_j)
\end{aligned}$$

# Kernel KMeans

1. Compute Class center

$$\mu_c = \frac{1}{|C_c|} \sum_{x_i \in C_c} \phi(x_i)$$

2. Compute distance between example and Class center

$$\|\phi(x) - \mu_c\|^2 = K(x,x) - \frac{2}{|C_c|} \sum_{x_i \in C_c} K(x, x_i) + \frac{1}{|C_c|^2} \sum_{x_i \in C_c} \sum_{x_j \in C_c} K(x_i, x_j)$$

# Exercice: Kernel Principal Component Analysis

Howto?