Kernel-Based Learning & Multivariate Modeling

MIRI Master

Lluís A. Belanche
belanche@cs.upc.edu

Soft Computing Research Group

Universitat Politècnica de Catalunya

2019-2020

Kernel-Based Learning & Multivariate Modeling

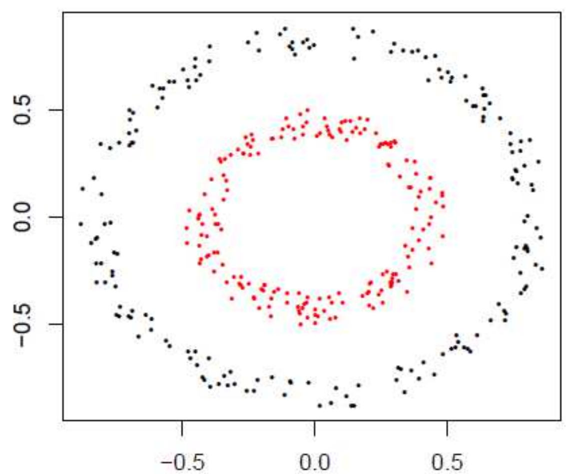
Syllabus

Sep 10 Introduction to kernel-based learning
Sep 17 The SVM for classification, regression & novelty detection (I)
Oct 01 The SVM for classification, regression & novelty detection (II)
Oct 08 Kernel design (I): theoretical issues
Oct 15 Kernel design (II): practical issues
Oct 22 Kernelizing ML & stats algorithms
Oct 29 Advanced topics

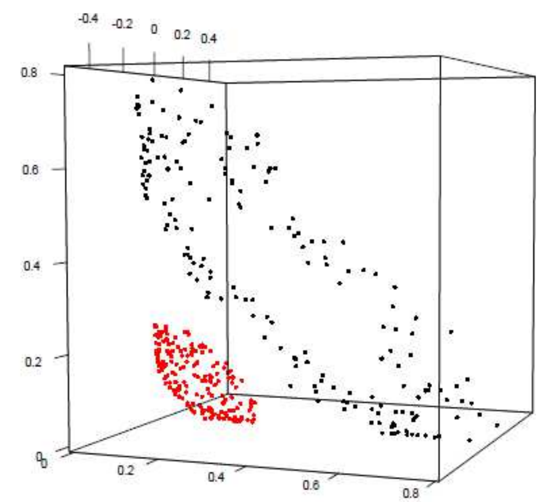
Kernel design (I): theoretical issues

General feature maps

Recall the idea of mapping input data into some Hilbert space (called the *feature space*) via a non-linear mapping $\phi : \mathcal{X} \to \mathcal{H}$



(a) Input Space (data not linearly separable)



(b) Feature Space (data linearly separable)

Kernel design (I): theoretical issues

Hilbert spaces

An abstract complete **vector space** endowed with an inner product:

Inner product requires symmetry, bilinearity and PSD-ness

Completeness means all Cauchy sequences converge to an element within the space (w.r.t. the norm induced by the inner product)

Kernel design (I): theoretical issues

Characterization of Kernels

Given a function $k : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$, which properties make it a valid kernel function for ML?

$\Rightarrow$ existence of a map $\phi : \mathcal{X} \to \mathcal{H}$ s.t.

1. $\mathcal{H}$ is a Hilbert space and

2. $k(\boldsymbol{x}, \boldsymbol{x}') = \left\langle \phi(\boldsymbol{x}), \phi(\boldsymbol{x}') \right\rangle_{\mathcal{H}}$ holds?

Kernel design (I): theoretical issues

Characterization of Kernels

A symmetric function k is called **positive semi-definite** (PSD) in $\mathcal{X}$ if:

for every $n \in \mathbb{N}$, and every choice $\boldsymbol{x}_1, \cdots, \boldsymbol{x}_n \in \mathcal{X}$,

the Gram matrix $\mathbf{K} = (k_{ij})$, where $k_{ij} = k(\boldsymbol{x}_i, \boldsymbol{x}_j)$, is PSD.

Theorem. A function $k : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ admits the existence of a map $\phi : \mathcal{X} \to \mathcal{H}$ s.t. $\mathcal{H}$ is a Hilbert space and $k(\boldsymbol{x}, \boldsymbol{x'}) = \left\langle \phi(\boldsymbol{x}), \phi(\boldsymbol{x'}) \right\rangle_{\mathcal{H}}$ if and only if k is a symmetric and PSD function in $\mathcal{X}$.

Kernel design (I): theoretical issues

On positive semi-definiteness

There are many **equivalent characterizations** of the PSD property for real symmetric matrices. Here are some: $A_{n\times n}$ is PSD if and only if ...

1. all of its eigenvalues are non-negative

2. the determinants of all of its leading principal minors are non-negative

3. there is a PSD matrix B such that $BB^\top = A$ (this matrix is unique, denoted with $B = A^{1/2}$, and called the *principal square root* of A)

4. $\forall \boldsymbol{c} \in \mathbb{R}^n,\ \boldsymbol{c}^\top A \boldsymbol{c} \geq 0$

Kernel design (I): theoretical issues

Generating the inner product

Given a kernel k symmetric and PSD, consider the space of functions:

$$\begin{aligned} \phi :& \mathcal{X} \to \mathbb{R}^{\mathcal{X}} \\ & \boldsymbol{x} \mapsto \phi(\boldsymbol{x}) : k(\boldsymbol{x}, \cdot) \end{aligned}$$

Define the (soon-to-be) vector space

$$\mathcal{H}_{\text{pre}} := \text{span}\Big\{\phi(\boldsymbol{x}) / \ \boldsymbol{x} \in \mathcal{X}\Big\}$$

$$= \Big\{f(\cdot) = \sum_{i=1}^{n} \alpha_i k(\boldsymbol{x}_i, \cdot) / \ n \in \mathbb{N}, \boldsymbol{x}_i \in \mathcal{X}, \alpha_i \in \mathbb{R}\Big\}$$

Kernel design (I): theoretical issues

Generating the inner product

Let $f, g \in \mathcal{H}_{\text{pre}}$; define an **inner product** in $\mathcal{H}_{\text{pre}}$ as

$$\langle f, g \rangle = \left\langle \sum_{i=1}^{n} \alpha_i k(\boldsymbol{x}_i, \cdot), \sum_{j=1}^{m} \beta_j k(\boldsymbol{x}'_j, \cdot) \right\rangle := \sum_{i=1}^{n} \sum_{j=1}^{m} \alpha_i \beta_j k(\boldsymbol{x}_i, \boldsymbol{x}'_j)$$

Note that $\langle f, k(\boldsymbol{x}, \cdot) \rangle = \sum_{i=1}^{n} \alpha_i k(\boldsymbol{x}_i, \boldsymbol{x}) = f(\boldsymbol{x})$

This is called the **reproducing property** of the kernel

Kernel design (I): theoretical issues

Generating the inner product

Let's check we have a valid inner product space:

1. $\langle f, g \rangle = \langle g, f \rangle$ (symmetry)

2. $\langle f, g \rangle = \sum_{i=1}^{n} \alpha_i g(\boldsymbol{x}_i) = \sum_{j=1}^{m} \beta_j f(\boldsymbol{x}'_j)$ (bilinearity)

3. $\langle f, f \rangle \geq 0$ with equality iff f is the zero function (PSD-ness)

This inner product satisfies the Cauchy-Schwartz inequality:

$$|\langle f, g \rangle| \leq \sqrt{\langle f, f \rangle} \cdot \sqrt{\langle g, g \rangle}, \ \forall f, g \in \mathcal{H}_{\text{pre}}$$

Kernel design (I): theoretical issues

Generating the inner product

1. Once we have an inner product, we have a **norm** $\|f\| := \sqrt{\langle f, f \rangle}$

2. Moreover, we have a **metric** $d(f, g) := \|f - g\|$

3. For any metric space, one can construct a **complete** metric space which contains the former as a dense subspace*; if completion is applied to an inner product space, the result is a Hilbert space $\mathcal{H}$

(*): Let (X, d) be a metric space, and $X_0 \subset X$. Then X_0 is dense in X if and only if $\forall x \in X$ there is a sequence of points $x_n \in X_0$ that has limit x.

Kernel design (I): theoretical issues

The Kernel Trick

Such a space is called a **Reproducing Kernel Hilbert Space** (RKHS)

Given the mapping $\phi : \mathcal{X} \to \mathcal{H}$, the **kernel trick** consists in performing the mapping and the inner product simultaneously by defining its associated kernel function:

$$k(\boldsymbol{x}, \boldsymbol{x'}) = \left\langle \phi(\boldsymbol{x}), \phi(\boldsymbol{x'}) \right\rangle_{\mathcal{H}}, \;\; \boldsymbol{x}, \boldsymbol{x'} \in \mathcal{X}$$

This way it is possible to compute inner products in $\mathcal{H}$ without explicitly performing/knowing the map (*e.g.* Gram matrices, the OSH)

Kernel design (I): theoretical issues

The Kernel Trick: an example

Take $k(\boldsymbol{x}, \boldsymbol{x'}) = \langle \boldsymbol{x}, \boldsymbol{x'} \rangle^q$, for $\boldsymbol{x}, \boldsymbol{x'} \in \mathbb{R}^d$. What is the underlying feature map ϕ?

$\Rightarrow$ Answer: the space spanned by all products of exactly q dimensions of $\mathbb{R}^d$.

Example: $\boldsymbol{x}, \boldsymbol{x'} \in \mathbb{R}^3$, and $q = 2$:

$$
\begin{aligned}
k(\boldsymbol{x}, \boldsymbol{x'}) &= \langle \boldsymbol{x}, \boldsymbol{x'} \rangle^2 = \left\langle \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix}, \begin{pmatrix} x'_1 \\ x'_2 \\ x'_3 \end{pmatrix} \right\rangle^2 \\
&= (x_1 x'_1 + x_2 x'_2 + x_3 x'_3)^2 = (x_1 x'_1 + x_2 x'_2)^2 + 2(x_1 x'_1 + x_2 x'_2) x_3 x'_3 + (x_3 x'_3)^2 \\
&= \left\langle \begin{pmatrix} x_1^2 \\ \sqrt{2} x_1 x_2 \\ \sqrt{2} x_1 x_3 \\ \sqrt{2} x_2 x_3 \\ x_2^2 \\ x_3^2 \end{pmatrix}, \begin{pmatrix} (x'_1)^2 \\ \sqrt{2} x'_1 x'_2 \\ \sqrt{2} x'_1 x'_3 \\ \sqrt{2} x'_2 x'_3 \\ (x'_2)^2 \\ (x'_3)^2 \end{pmatrix} \right\rangle \\
&= \langle \phi(\boldsymbol{x}), \phi(\boldsymbol{x'}) \rangle
\end{aligned}
$$

Kernel design (I): theoretical issues

Popular choices for the Kernel

Polynomial kernels (relation to GLDs)

$$k(\boldsymbol{x}, \boldsymbol{x'}) = (a \langle \boldsymbol{x}, \boldsymbol{x'} \rangle + c)^q, \;\; q \in \mathbb{N}, a > 0, c \geq 0 \in \mathbb{R}$$

Gaussian kernels (relation to RBFNNs)

$$k(\boldsymbol{x}, \boldsymbol{x'}) = \exp\left(-\gamma \|\boldsymbol{x} - \boldsymbol{x'}\|^2\right), \;\; \gamma > 0 \in \mathbb{R}$$

Laplacian kernels (relation to ???)

$$k(\boldsymbol{x}, \boldsymbol{x'}) = \exp\left(-\gamma \|\boldsymbol{x} - \boldsymbol{x'}\|\right), \;\; \gamma > 0 \in \mathbb{R}$$

Sigmoidal kernels (relation to MLPs)

$$k(\boldsymbol{x}, \boldsymbol{x'}) = g(a \langle \boldsymbol{x}, \boldsymbol{x'} \rangle + c)$$

with g a sigmoidal (*e.g.*, logistic, tanh, ...) and particular choices for a, c

Kernel design (I): theoretical issues

Kernel construction

Which **operations** (*e.g.*, products, sums, composition, etc) on kernels produce new kernels? (*closure properties*)

Example:

Consider functions $p : \mathbb{R} \to \mathbb{R}$.

If k is a kernel, when is $p \circ k$ a kernel?

Kernel design (I): theoretical issues

Closure properties

- Inner products: finite (sums), infinite countable (series) or infinite uncountable (integrals)
- Scalar operations, sums and direct sums
- Products and tensor products
- Limits of point-wise convergent sequences
- Composition with certain analytic functions
- Normalization

Kernel design (I): theoretical issues

Inner products

1. Let $f_1, \ldots, f_n : \mathcal{X} \to \mathbb{R}$ be a finite collection of functions:

$$k(\boldsymbol{x}, \boldsymbol{x}') = \sum_{i=1}^{n} f_i(\boldsymbol{x}) \cdot f_i(\boldsymbol{x}')$$

2. Let $\{f_n\}_n$ be a sequence of functions $\mathcal{X} \to \mathbb{R}$; if the series is convergent:

$$k(\boldsymbol{x}, \boldsymbol{x}') = \sum_{n=1}^{\infty} f_n(\boldsymbol{x}) \cdot f_n(\boldsymbol{x}')$$

3. Let $f : \mathcal{X} \times W \to \mathbb{R}$ be a parameterized (indexed) set of functions; if the integral is well-defined:

$$k(\boldsymbol{x}, \boldsymbol{x}') = \int_W f(\boldsymbol{x}; \boldsymbol{w}) \cdot f(\boldsymbol{x}'; \boldsymbol{w})\, d\boldsymbol{w}$$

Kernel design (I): theoretical issues

Scalar operations, sums and direct sums

Take $k_1, k_2 : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ and $k' : \mathcal{Y} \times \mathcal{Y} \to \mathbb{R}$ kernels

- $a \cdot k_1(\boldsymbol{x}, \boldsymbol{x'}) + b, \ a > 0, b \geq 0$

- $k_+ : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ defined as

$$k_+(\boldsymbol{x}, \boldsymbol{x'}) = k_1(\boldsymbol{x}, \boldsymbol{x'}) + k_2(\boldsymbol{x}, \boldsymbol{x'})$$

- $k_\oplus : (\mathcal{X} \times \mathcal{Y}) \times (\mathcal{X} \times \mathcal{Y}) \to \mathbb{R}$ defined as

$$k_\oplus((\boldsymbol{x}, \boldsymbol{y}), (\boldsymbol{x'}, \boldsymbol{y'})) = k_1(\boldsymbol{x}, \boldsymbol{x'}) + k'(\boldsymbol{y}, \boldsymbol{y'})$$

Kernel design (I): theoretical issues

Products and tensor products

Take $k_1, k_2 : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ and $k' : \mathcal{Y} \times \mathcal{Y} \to \mathbb{R}$ kernels

- $k_\cdot : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ defined as

$$k_\cdot(\boldsymbol{x}, \boldsymbol{x'}) = k_1(\boldsymbol{x}, \boldsymbol{x'}) \cdot k_2(\boldsymbol{x}, \boldsymbol{x'})$$

- $k_\odot : (\mathcal{X} \times \mathcal{Y}) \times (\mathcal{X} \times \mathcal{Y}) \to \mathbb{R}$ defined as

$$k_\odot((\boldsymbol{x}, \boldsymbol{y}), (\boldsymbol{x'}, \boldsymbol{y'})) = k_1(\boldsymbol{x}, \boldsymbol{x'}) \cdot k'(\boldsymbol{y}, \boldsymbol{y'})$$

Kernel design (I): theoretical issues

Limits of sequences

Let $\{k_n\}_n : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ be a sequence of kernels; if, for all $\boldsymbol{x}, \boldsymbol{x'} \in \mathcal{X}$, the limit exists,

then $k_\infty : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ defined as

$$k_\infty(\boldsymbol{x}, \boldsymbol{x'}) := \operatorname*{lím}_{n\to\infty} k_n(\boldsymbol{x}, \boldsymbol{x'}),\ \forall \boldsymbol{x}, \boldsymbol{x'} \in \mathcal{X}$$

is a valid kernel.

Kernel design (I): theoretical issues

Composition with analytic functions

Theorem. Let f be a real analytic function with radius of convergence $R > 0$ s.t. all the coefficients in its power series expansion are non-negative. Let $k : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ be a kernel fulfilling $|k(\boldsymbol{x}, \boldsymbol{x'})| < R$.

Then $k_f : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ given by $k_f(\boldsymbol{x}, \boldsymbol{x'}) := f(k(\boldsymbol{x}, \boldsymbol{x'}))$ is a valid kernel.

Example: $f(z) = \exp(z)$

A real function f is *analytic* in an open set $\Omega \subset \mathbb{R}$ iff for every $x_0 \in \Omega$ there is a neighborhood of x_0 for which the Taylor series expansion of f in x_0 coincides with $f(x)$.

Kernel design (I): theoretical issues

Operations in feature space

Norms in feature space:

$$||\phi(\boldsymbol{x})||_{\mathcal{H}} = \sqrt{\langle \phi(\boldsymbol{x}), \phi(\boldsymbol{x}) \rangle_{\mathcal{H}}} = \sqrt{k(\boldsymbol{x}, \boldsymbol{x})}$$

Norms of linear combinations in feature space:

$$\Big|\Big| \sum_i \alpha_i \phi(\boldsymbol{x}_i) \Big|\Big|_{\mathcal{H}}^2 = \langle K\boldsymbol{\alpha}, \boldsymbol{\alpha} \rangle = \boldsymbol{\alpha}^\top K \boldsymbol{\alpha}$$

Kernel design (I): theoretical issues

Operations in feature space

Distances in feature space:

$$||\phi(\boldsymbol{x}) - \phi(\boldsymbol{x'})||_{\mathcal{H}} = \sqrt{\langle \phi(\boldsymbol{x}), \phi(\boldsymbol{x}) \rangle_{\mathcal{H}} + \left\langle \phi(\boldsymbol{x'}), \phi(\boldsymbol{x'}) \right\rangle_{\mathcal{H}} - 2 \left\langle \phi(\boldsymbol{x}), \phi(\boldsymbol{x'}) \right\rangle_{\mathcal{H}}}$$

and then $d_{\mathcal{H}}(\boldsymbol{x}, \boldsymbol{x'}) := \sqrt{k(\boldsymbol{x}, \boldsymbol{x}) + k(\boldsymbol{x'}, \boldsymbol{x'}) - 2k(\boldsymbol{x}, \boldsymbol{x'})}$ is an Euclidean metric (distance) when ϕ is injective (otherwise it would be a pseudo-metric).

Kernel design (I): theoretical issues

Normalizing kernels

If k is a kernel, then so is:

$$k_n(\boldsymbol{x}, \boldsymbol{x'}) := \frac{k(\boldsymbol{x}, \boldsymbol{x'})}{\sqrt{k(\boldsymbol{x}, \boldsymbol{x})} \cdot \sqrt{k(\boldsymbol{x'}, \boldsymbol{x'})}}$$

Moreover, $|k_n(\boldsymbol{x}, \boldsymbol{x'})| \leq 1$ and $k_n(\boldsymbol{x}, \boldsymbol{x}) = 1$.

The effect is to project each point onto the unit sphere, since

$$1 = k_n(\boldsymbol{x}, \boldsymbol{x}) = \langle \phi_n(\boldsymbol{x}), \phi_n(\boldsymbol{x}) \rangle = ||\phi_n(\boldsymbol{x})||^2$$

Kernel design (I): theoretical issues

General linear kernel

Theorem. If $A_{d\times d}$ is a PSD matrix, then the function $k : \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}$ given by $k(\boldsymbol{x}, \boldsymbol{x'}) = \boldsymbol{x}^\top A \boldsymbol{x'}$ is a kernel.

Proof. Since A is PSD we can write it in the form $A = BB^\top$. For every $n \in \mathbb{N}$, and every choice $\boldsymbol{x}_1, \cdots, \boldsymbol{x}_n \in \mathbb{R}^d$, we form the matrix $\mathbf{K} = (k_{ij})$, where $k_{ij} = k(\boldsymbol{x}_i, \boldsymbol{x}_j) = \boldsymbol{x}_i^\top A \boldsymbol{x}_j$. Then for every $\boldsymbol{c} \in \mathbb{R}^n$:

$$\sum_{i=1}^{n}\sum_{j=1}^{n} c_i c_j k_{ij} = \sum_{i=1}^{n}\sum_{j=1}^{n} c_i c_j \boldsymbol{x}_i^\top A \boldsymbol{x}_j = \sum_{i=1}^{n}\sum_{j=1}^{n} c_i c_j (B^\top \boldsymbol{x}_i)^\top (B^\top \boldsymbol{x}_j)$$

$$= \left\| \sum_{i=1}^{n} c_i (B^\top \boldsymbol{x}_i) \right\|^2 \geq 0. \qquad \text{Note that } \phi(\boldsymbol{x}) = B^\top \boldsymbol{x}$$

Kernel design (I): theoretical issues

Polynomial kernels

1. If k is a kernel and p is a (non-zero) polynomial of degree q with non-negative coefficients, then the function

$$k_p(\boldsymbol{x}, \boldsymbol{x}') := p(k(\boldsymbol{x}, \boldsymbol{x}'))$$

   is also a kernel.

2. The special case where k is linear and $p(z) = (az + c)^q, a > 0, c \geq 0 \in \mathbb{R}$ leads to the so-called **polynomial kernel**

Kernel design (I): theoretical issues

Translation invariant and radial kernels

We say that a kernel $k : \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}$ is:

Translation invariant if it has the form $k(\boldsymbol{x}, \boldsymbol{x'}) = T(\boldsymbol{x} - \boldsymbol{x'})$, where $T : \mathbb{R}^d \to \mathbb{R}$ is a differentiable function

Radial if it has the form $k(\boldsymbol{x}, \boldsymbol{x'}) = t(\|\boldsymbol{x} - \boldsymbol{x'}\|)$, where $t : [0, \infty) \to [0, \infty)$ is a differentiable function

Radial kernels fulfill $k(\boldsymbol{x}, \boldsymbol{x}) = t(0)$.

Kernel design (I): theoretical issues

The Gaussian kernel

Consider the function $t(z) = \exp(-\gamma z^2), \gamma > 0$. The resulting radial kernel is known as the **Gaussian RBF kernel**:

$$k(\boldsymbol{x}, \boldsymbol{x'}) = \exp(-\gamma \|\boldsymbol{x} - \boldsymbol{x'}\|^2)$$

Many people refer to it simply as "the RBF kernel"

You can also find it as:

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

Kernel design (I): theoretical issues

Using the exponential

1. If k is a kernel and $\gamma > 0$, then the function

$$k(\boldsymbol{x}, \boldsymbol{x'}) = \exp(\gamma k(\boldsymbol{x}, \boldsymbol{x'}))$$

   is also a kernel.

2. If k is a kernel and $\gamma > 0$, then the function

$$k(\boldsymbol{x}, \boldsymbol{x'}) = \exp\Big(- \gamma[k(\boldsymbol{x}, \boldsymbol{x}) + k(\boldsymbol{x'}, \boldsymbol{x'}) - 2k(\boldsymbol{x}, \boldsymbol{x'})]\Big)$$

   is also a kernel.

Kernel design (I): theoretical issues

Characterization of Kernels

A symmetric function k is called **conditionally positive semi-definite** (CPSD) in $\mathcal{X}$ if for every $n \in \mathbb{N}$, and every choice $x_1, \cdots, x_n \in \mathcal{X}$, the matrix $\mathbf{K} = (k_{ij})$, where $k_{ij} = k(\boldsymbol{x}_i, \boldsymbol{x}_j)$, is CPSD.

A real symmetric matrix $A_{n \times n}$ is CPSD if and only if $\forall \boldsymbol{c} \in \mathbb{R}^n$ such that $\boldsymbol{c}^\top \mathbf{1} = 0$, $\boldsymbol{c}^\top A \boldsymbol{c} \geq 0$.

It turns out that it suffices for a kernel to be CPSD! Since the class of CPSD kernels is larger than that of PSD kernels:

1. a larger set of kernel functions are usable by kernel machines

2. a larger set of learning algorithms are prone to kernelization