Kernel machines

Nathanaël Carraz Rakotonirina

Mathématiques Informatique et Statistique Appliquées (MISA)
Université d'Antananarivo

Motivations

- We want to use features that are more appropriate (instead of just the raw inputs) for a given problem.
- Instead of operating on the inputs $\boldsymbol{x}$, we operate on features $\phi(\boldsymbol{x})$ (using the feature mapping ϕ) which can result in non-linear models with more capacity.
- We want to use these features efficiently in our models.

Kernel trick

Given a feature mapping ϕ, we define the corresponding **kernel** to be

$$\begin{aligned} \mathcal{K} : \mathcal{X} \times \mathcal{X} &\to \mathbb{R}^+ \\ (\boldsymbol{x}, \boldsymbol{x}') &\mapsto \mathcal{K}(\boldsymbol{x}, \boldsymbol{x}') = \phi(\boldsymbol{x})^\top \phi(\boldsymbol{x}') \end{aligned}$$

The **kernel trick** is about replacing the dot products $\boldsymbol{x}^\top \boldsymbol{x}'$ in our model with the kernel function $\mathcal{K}(\boldsymbol{x}, \boldsymbol{x}')$. By doing so

- The model would now be learning using the features ϕ.
- There is a way to efficiently calculate $\mathcal{K}(\boldsymbol{x}, \boldsymbol{x}')$ without having to explicitly find and compute the feature vectors $\phi(\boldsymbol{x})$ (which can be very expensive).

How to find a kernel that is valid or corresponds to some feature mapping ϕ ?

Mercer kernel

A **Mercer kernel** or **positive definite kernel** is a function $\mathcal{K} : \mathcal{X} \times \mathcal{X} \to \mathbb{R}^+$ such that:

- It is symmetric $\mathcal{K}(\boldsymbol{x}, \boldsymbol{x}') = \mathcal{K}(\boldsymbol{x}', \boldsymbol{x})$
- For any set of (unique) points $\{\boldsymbol{x}_i\}_{i=1}^N$, and any numbers $c_i \in \mathbb{R}$

$$\sum_{i=1}^{N} \sum_{j=1}^{N} \mathcal{K}(\boldsymbol{x}_i, \boldsymbol{x}_j) c_i c_j \geq 0$$

There is another way to define it. Given a set of N points, the **Gram matrix $\boldsymbol{K}$** is an $N \times N$ symmetric matrix with entries $\boldsymbol{K}_{i,j} = \mathcal{K}(\boldsymbol{x}_i, \boldsymbol{x}_j)$. $\mathcal{K}$ is a Mercer kernel iff the Gram matrix $\boldsymbol{K}$ is positive definite for any set of (distinct) points $\{\boldsymbol{x}_i\}_{i=1}^N$.

Mercer theorem

A kernel $\mathcal{K} : \mathcal{X} \times \mathcal{X} \to \mathbb{R}^+$ can be computed by an inner product of some feature vectors iff for any set of poins $\{\boldsymbol{x}_i\}_{i=1}^N$, it is postive definite.

Example of Mercer kernels

- Linear : $\mathcal{K}(\boldsymbol{x}, \boldsymbol{x}') = \boldsymbol{x}^\top \boldsymbol{x}'$
- Quadratic : $\mathcal{K}(\boldsymbol{x}, \boldsymbol{x}') = (\boldsymbol{x}^\top \boldsymbol{x}' + b)^2$
- Polynomial : $\mathcal{K}(\boldsymbol{x}, \boldsymbol{x}') = (\boldsymbol{x}^\top \boldsymbol{x}' + b)^p$
- Gaussian (RBF) $\mathcal{K}(\boldsymbol{x}, \boldsymbol{x}') = \exp\big(- \frac{||\boldsymbol{x}-\boldsymbol{x}'||^2}{2\sigma^2}\big)$
- Laplacian : $\mathcal{K}(\boldsymbol{x}, \boldsymbol{x}') = \exp\big(- \frac{|\boldsymbol{x}-\boldsymbol{x}'|}{\sigma}\big)$

Some kernels may contain hyperparameters that needs to be tuned using cross-validation.

Making new kernels

Given valid kernels $\mathcal{K}_1(\boldsymbol{x}, \boldsymbol{x}')$ and $\mathcal{K}_2(\boldsymbol{x}, \boldsymbol{x}')$, we can create a new kernel using:

- $\mathcal{K}(\boldsymbol{x}, \boldsymbol{x}') = \mathcal{K}_1(\boldsymbol{x}, \boldsymbol{x}') + \mathcal{K}_2(\boldsymbol{x}, \boldsymbol{x}')$
- $\mathcal{K}(\boldsymbol{x}, \boldsymbol{x}') = \mathcal{K}_1(\boldsymbol{x}, \boldsymbol{x}') \times \mathcal{K}_2(\boldsymbol{x}, \boldsymbol{x}')$
- $\mathcal{K}(\boldsymbol{x}, \boldsymbol{x}') = c\mathcal{K}_1(\boldsymbol{x}, \boldsymbol{x}')$ for any constant $c > 0$
- $\mathcal{K}(\boldsymbol{x}, \boldsymbol{x}') = f(\boldsymbol{x})\mathcal{K}_1(\boldsymbol{x}, \boldsymbol{x}')f(\boldsymbol{x}')$ for any function f
- $\mathcal{K}(\boldsymbol{x}, \boldsymbol{x}') = q(\mathcal{K}_1(\boldsymbol{x}, \boldsymbol{x}'))$ for any function polynomial q with non-negative coefficients
- $\mathcal{K}(\boldsymbol{x}, \boldsymbol{x}') = \exp(\mathcal{K}_1(\boldsymbol{x}, \boldsymbol{x}'))$
- $\mathcal{K}(\boldsymbol{x}, \boldsymbol{x}') = \boldsymbol{x}^\top \boldsymbol{A}\boldsymbol{x}'$ for any positive semi-definite matrix $\boldsymbol{A}$

More

Explore further

- Other examples of kernels
- Kernels for structured inputs (strings, time series, graphs, images)
- Kernel PCA
- Kernel ridge regression
- Automatically Choosing a Kernel