kernel method

Authors

Alexander Jung*, Department of Computer Science, Aalto University, Espoo, Finland
*Corresponding author: alex.jung@aalto.fi

Synonyms

kernel machine

Abstract

A kernel method applies a linear method, such as a linear model or a linear classifier, to transformed feature vectors. The transformation is constructed from a kernel: each feature vector is mapped to a function with domain equal to the original feature space, and inner products between transformed feature vectors reduce to evaluations of a kernel. Kernel methods offer great flexibility in the choice of the feature space, which can consist of text documents or discrete structures such as graphs.

Definition

A kernel method is a machine learning (ML) method that applies a linear method, such as a linear model for regression or a linear classifier for classification, to transformed feature vectors that are constructed from a kernel (Lampert, 2009; Schölkopf and Smola, 2002). Kernel methods are used in computer vision to decide whether an image shows a specific object (Lampert, 2009), and in natural language processing (NLP) to classify text documents by topic (Cristianini and Shawe-Taylor, 2000).

Linear methods learn a hypothesis that combines the features of a data point linearly, e.g., $h(\mathbf{x}) = \mathbf{w}^\top \mathbf{x}$ for the feature space $\mathcal{X} = \mathbb{R}^d$. Such a hypothesis predicts well only if the relation between the feature vectors and the labels of data points is approximately linear. Moreover, a linear method requires numeric feature vectors and is not directly applicable to data points from an arbitrary feature space $\mathcal{X}$, such as text documents or graphs.

Both limitations can be mitigated by a feature transformation $\phi : \mathcal{X} \to \mathcal{H}$ from the original feature space $\mathcal{X}$ to a transformed feature space $\mathcal{H}$, chosen as a Hilbert space. Fig. 1 shows a binary classification problem with the first limitation: no straight line separates the two classes in the original feature space. A suitable feature transformation makes a dataset "more linear": the relation between the transformed feature vectors $\phi(\mathbf{x})$ and the labels is closer to linear in $\mathcal{H}$ than the relation between the original feature vectors $\mathbf{x}$ and the labels is in $\mathcal{X}$.

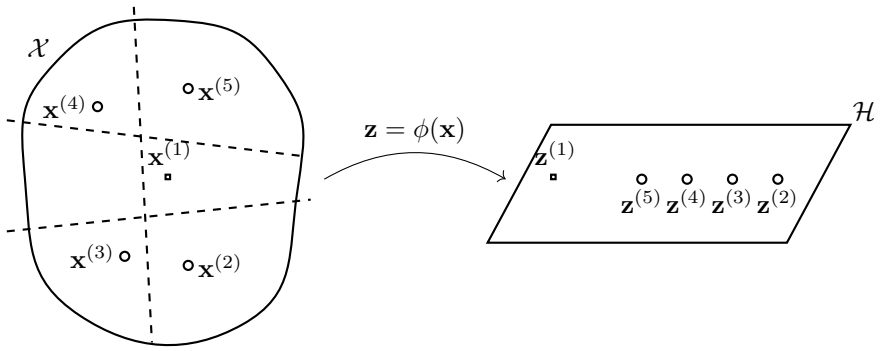

Figure 1: Five data points characterized by feature vectors $\mathbf{x}^{(r)}$ and labels $y^{(r)} \in \{\circ, \square\}$ for $r = 1, \ldots, 5$. The curved boundary indicates the original feature space $\mathcal{X}$ (left); the parallelogram indicates a section of the transformed feature space $\mathcal{H}$ (right), which is a linear space (see Hilbert space). With these feature vectors, there is no way to separate the two classes by a straight line (representing the decision boundary of a linear classifier). The dashed lines illustrate three failed attempts: each places the feature vector $\mathbf{x}^{(1)}$ (class $\square$) on the same side as at least one feature vector of class $\circ$, so a linear classifier with any of these decision boundaries misclassifies at least one data point. In contrast, the transformed feature vectors $\mathbf{z}^{(r)} = \phi\big(\mathbf{x}^{(r)}\big)$ allow the data points to be separated using a linear classifier

A useful feature transformation ϕ often delivers transformed feature vectors in a Hilbert space $\mathcal{H}$ of high (possibly infinite) dimension (Schölkopf and Smola, 2002). Computing and storing the transformed feature vectors $\phi(\mathbf{x})$ explicitly can then be infeasible. However, linear methods access the feature vectors of data points only through inner products between pairs of feature vectors (see linear model). For example, the prediction delivered by a trained support vector machine (SVM) is a weighted sum of inner products between the feature vector of a new data point and the feature vectors in the training set. To apply a linear method to the transformed feature vectors, it is therefore enough to know the inner products $\langle \phi(\mathbf{x}), \phi(\mathbf{x}') \rangle$ for every possible pair of feature vectors $\mathbf{x}, \mathbf{x}' \in \mathcal{X}$. These inner products are captured by a single function of two arguments, the kernel defined by

$$k\big(\mathbf{x}, \mathbf{x}'\big) := \langle \phi(\mathbf{x}), \phi(\mathbf{x}') \rangle, \text{ for } \mathbf{x}, \mathbf{x}' \in \mathcal{X}.$$

The value $k\big(\mathbf{x}, \mathbf{x}'\big)$ quantifies the similarity of the feature vectors $\mathbf{x}$ and $\mathbf{x}'$. As an inner product of transformed feature vectors, the kernel is symmetric, and every matrix of pairwise kernel values is positive semi-definite (psd); these two properties characterize kernels. A linear method that is applied to the transformed feature vectors thus requires only the kernel k; the feature transformation ϕ itself is never evaluated.

Kernel methods reverse this construction: they start from a kernel $k : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ and construct the feature transformation from it. The feature transformation sends a feature vector $\mathbf{x} \in \mathcal{X}$ to the function $k\big(\mathbf{x}, \cdot\big)$, i.e., the transformed feature vector $\mathbf{z} := \phi(\mathbf{x}) = k\big(\mathbf{x}, \cdot\big)$ is itself a function with domain $\mathcal{X}$, for every $\mathbf{x} \in \mathcal{X}$. These functions belong to the reproducing kernel Hilbert space (RKHS) $\mathcal{H}_k$ associated with the kernel k (Aronszajn, 1950). The inner product between the transformed feature vectors of two original feature vectors $\mathbf{x}, \mathbf{x}' \in \mathcal{X}$ is a single kernel evaluation,

$$\langle k\big(\mathbf{x}, \cdot\big), k\big(\mathbf{x}', \cdot\big) \rangle = k\big(\mathbf{x}, \mathbf{x}'\big).$$

More generally, the inner product between the transformed feature vector $k\big(\mathbf{x}, \cdot\big)$ and any function $h \in \mathcal{H}_k$ is a point evaluation,

$$\langle h, k\big(\mathbf{x}, \cdot\big) \rangle = h(\mathbf{x}),$$

which is referred to as the reproducing property of the RKHS $\mathcal{H}_k$ (Aronszajn, 1950). This reproducing property contains $\langle k\big(\mathbf{x}, \cdot\big), k\big(\mathbf{x}', \cdot\big) \rangle = k\big(\mathbf{x}, \mathbf{x}'\big)$ as a special case which is obtained for the choice $h = k\big(\mathbf{x}', \cdot\big)$. The (possibly infinite-dimensional) transformed feature vectors therefore never need to be computed explicitly.

Fig. 2 depicts the RKHS $\mathcal{H}_k$ for a training set with two data points: a hypothesis h is a single vector in $\mathcal{H}_k$. The value $\langle h, k\big(\mathbf{x}, \cdot\big) \rangle = h(\mathbf{x})$ is a linear function of the transformed feature vector $k\big(\mathbf{x}, \cdot\big) \in \mathcal{H}_k$, but the function $\mathbf{x} \mapsto h(\mathbf{x})$ induced on the original feature space $\mathcal{X}$ is nonlinear in general. Linearity in the original feature vector $\mathbf{x}$ may not even be defined, since the feature space $\mathcal{X}$ need not be a vector space, e.g., when it consists of text documents or graphs.

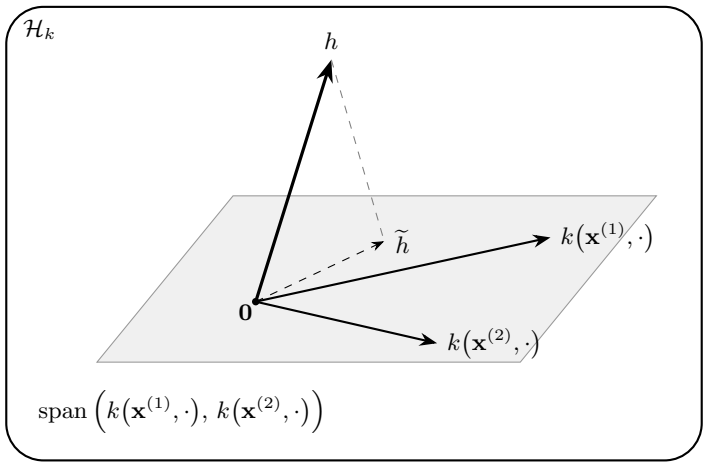

Figure 2: The RKHS $\mathcal{H}_k$ for a training set with two data points having feature vectors $\mathbf{x}^{(1)}, \mathbf{x}^{(2)}$. Their transformed feature vectors $k\big(\mathbf{x}^{(1)},\cdot\big)$ and $k\big(\mathbf{x}^{(2)},\cdot\big)$ span a two-dimensional subspace (shaded parallelogram). A hypothesis h is a vector in $\mathcal{H}_k$; its prediction for a data point with feature vector $\mathbf{x}$ is the inner product $\langle h, k\big(\mathbf{x},\cdot\big)\rangle = h(\mathbf{x})$. This prediction is a linear function of the transformed feature vector $k\big(\mathbf{x},\cdot\big)$, but the function $\mathbf{x} \mapsto h(\mathbf{x})$ induced on $\mathcal{X}$ is nonlinear in general. Replacing h by its orthogonal projection $\widetilde{h}$ (dashed arrow) onto the shaded subspace leaves the predictions for both data points in the training set unchanged and never increases $\|h\|_{\mathcal{H}_k}$

Kernel methods can be formulated as regularized empirical risk minimization (RERM) over the RKHS $\mathcal{H}_k$,

$$\min_{h\in\mathcal{H}_k} \frac{1}{m}\sum_{r=1}^{m} L\left(\big(\mathbf{x}^{(r)}, y^{(r)}\big), h\right) + \alpha\|h\|^2_{\mathcal{H}_k},$$

over a training set with feature vectors $\mathbf{x}^{(r)}$ and labels $y^{(r)}$, for $r = 1, \ldots, m$, and with a regularization parameter $\alpha > 0$ that weights the penalty term $\|h\|^2_{\mathcal{H}_k}$. The loss function $L\left(\big(\mathbf{x}^{(r)}, y^{(r)}\big), h\right)$ depends on h only through the prediction $h\big(\mathbf{x}^{(r)}\big)$, i.e., it is a function of the prediction $h\big(\mathbf{x}^{(r)}\big)$ and the label $y^{(r)}$ only. Since $h \in \mathcal{H}_k$, the reproducing property delivers this prediction as an inner product, $h\big(\mathbf{x}^{(r)}\big) = \langle h, k\big(\mathbf{x}^{(r)},\cdot\big)\rangle$. By the representer theorem, this optimization problem has a minimizer of the form $\widehat{h} = \sum_{r=1}^{m} \beta_r k\big(\mathbf{x}^{(r)},\cdot\big)$ with expansion coefficients $\beta_1, \ldots, \beta_m \in \mathbb{R}$ (Schölkopf and Smola, 2002). Fig. 2 illustrates the underlying projection argument: replacing a hypothesis h by its orthogonal projection $\widetilde{h}$ onto the subspace spanned by $k\big(\mathbf{x}^{(1)},\cdot\big), \ldots, k\big(\mathbf{x}^{(m)},\cdot\big)$ leaves the predictions on the training set unchanged and never increases the penalty term; hence some minimizer $\widehat{h}$ lies in this subspace and has the above form.

Training thus reduces to a convex optimization problem in these m co-

efficients, and the prediction $\widehat{h}(\mathbf{x}) = \sum_{r=1}^{m} \beta_r k\big(\mathbf{x}^{(r)}, \mathbf{x}\big)$ requires only kernel evaluations. Examples of such kernel methods are the kernel SVM, obtained for the hinge loss, and kernel ridge regression, obtained for the squared error loss (Cristianini and Shawe-Taylor, 2000; Hastie et al., 2009, Ch. 12). For $\mathcal{X} = \mathbb{R}^d$, a widely used choice of the kernel k is the Gaussian kernel $k\big(\mathbf{x}, \mathbf{x}'\big) = \exp\big(-\|\mathbf{x} - \mathbf{x}'\|_2^2/(2\sigma^2)\big)$ with bandwidth $\sigma > 0$. Kernels are also available for data points without numeric features, such as strings and graphs (Schölkopf and Smola, 2002). Fig. 3 shows the nonlinear decision boundary $\widehat{h}(\mathbf{x}) = 0$ of a hypothesis $\widehat{h}$ learned by a Gaussian-kernel method with squared error loss from a training set of two concentric rings that no linear classifier separates. Here kernel ridge regression fits the labels $y^{(r)} \in \{-1, +1\}$ and the prediction is the sign of the fitted value, a construction referred to as regularized least-squares classification (Rifkin et al., 2003).

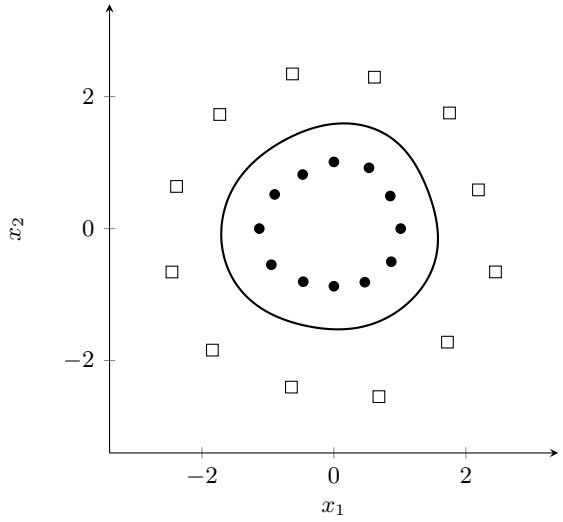

Figure 3: A Gaussian-kernel method (kernel ridge regression with $\sigma = 0.8$, $\alpha = 10^{-3}$) trained on a training set of two concentric rings. Data points with label $y = +1$ are drawn as filled circles, those with $y = -1$ as open squares. The decision boundary $\widehat{h}(\mathbf{x}) = 0$ of the learned hypothesis (solid closed curve) encircles the inner ring. Note that the dataset is not linearly separable in the original feature space. Data generated by `pythondemos/kernelmethod.py`

Kernel methods impose the smoothness assumption through the RKHS norm. That norm $\|h\|_{\mathcal{H}_k}$ quantifies the smoothness of a hypothesis h: it is large for a rapidly varying h and small for a smooth h. Moreover, different kernels impose different kinds of smoothness. The Gaussian kernel gives infinitely differentiable functions and penalizes high-frequency components. The Matérn kernels are a family of kernels indexed by a smoothness parameter $\nu > 0$: their RKHS coincides, with equivalent norms, with the Sobolev space of order determined by ν,

whose functions have derivatives up to that order in the weak sense (Rasmussen and Williams, 2006, Eq. 4.15; Kanagawa et al., 2018, Example 2.6). Because the penalty term $\alpha\|h\|^2_{\mathcal{H}_k}$ in the above RERM is the squared RKHS norm, it implements smoothness as quantified by $\|h\|_{\mathcal{H}_k}$: among hypotheses with the same training error, the RERM objective function is smallest for the hypothesis with the smallest RKHS norm.

The choice of kernel shapes not only training but also the predictions of the learned hypothesis: for localized kernels, such as the Gaussian kernel, the prediction $\widehat{h}(\mathbf{x})$ is dominated by the data points in the training set whose feature vectors are close to $\mathbf{x}$. In this sense, kernel methods implement the smoothness assumption: data points with nearby feature vectors obtain similar predictions. The locality of the predictions also makes kernel methods a soft-weighted analogue of k-nearest neighbors (k-NN): instead of averaging the labels of a fixed number of nearest data points, the prediction weights all data points in the training set by the kernel value $k\big(\mathbf{x}^{(r)}, \mathbf{x}\big)$ (Hastie et al., 2009, Ch. 6).

Cross References

kernel, feature transformation, Hilbert space, RKHS, linear model, linear classifier, SVM, kernel ridge regression, ridge regression, smoothness assumption, k-NN

References

1. Cristianini and Shawe-Taylor (2000). An Introduction to Support Vector Machines and Other Kernel-based Learning Methods. Cambridge Univ. Press, New York, NY, USA.

2. Lampert (2009). Kernel Methods in Computer Vision. Found. Trends Comput. Graph. Vis., 4(3), pp. 193–285.

3. Schölkopf and Smola (2002). Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond. MIT Press, Cambridge, MA, USA.

4. Rasmussen and Williams (2006). Gaussian Processes for Machine Learning. MIT Press, Cambridge, MA, USA.

5. Aronszajn (1950). Theory of Reproducing Kernels. Trans. Am. Math. Soc., 68(3), pp. 337–404.

6. Hastie et al. (2009). The Elements of Statistical Learning: Data Mining, Inference, and Prediction. 2nd ed., Springer Science+Business Media, New York, NY, USA.

7. Kanagawa et al. (2018). Gaussian Processes and Kernel Methods: A Review on Connections and Equivalences. URL: https://arxiv.org/abs/1807.02582.

8. Rifkin et al. (2003). Regularized Least-Squares Classification. In: Advances in Learning Theory: Methods, Model and Applications, pp. 131–154. IOS Press.