Kernel-based Regression

Matthias Rupp

Fritz Haber Institute of the Max Planck Society, Berlin, Germany

2017 NYU Shanghai Summer School on
Machine Learning in the Molecular Sciences

June 12–16, Shanghai, China

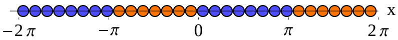
Outline

1. Kernel learning
   kernel trick, kernels
2. Kernel ridge regression
   Gaussian process regression
3. Model building
   validation, hyperparameters, overfitting

The kernel trick

Idea:

- **Transform** samples into higher-dimensional space
- **Implicitly** compute inner products there
- Rewrite linear algorithm to use only inner products



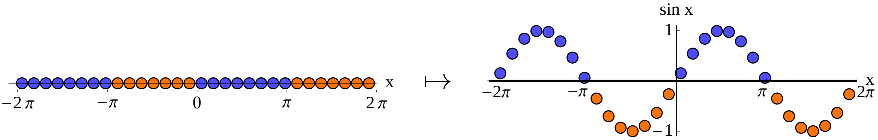
Input space $\mathcal{X}$

Schölkopf, Smola: Learning with Kernels, 2002; Hofmann et al.: *Ann. Stat.* 36, 1171, 2008.

The kernel trick

Idea:

- **Transform** samples into higher-dimensional space
- **Implicitly** compute inner products there
- Rewrite linear algorithm to use only inner products



Input space $\mathcal{X}$ $\quad\xrightarrow{\phi}\quad$ Feature space $\mathcal{H}$

$$k : \mathcal{X} \times \mathcal{X} \to \mathbb{R}, \qquad k(x, z) = \langle \phi(x), \phi(z) \rangle$$

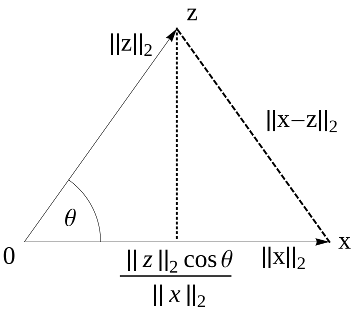
Schölkopf, Smola: Learning with Kernels, 2002; Hofmann et al.: *Ann. Stat.* 36, 1171, 2008.

Kernel functions

Kernels correspond to **inner products**.

If $k : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ is symmetric positive semi-definite, then $k(x, z) = \langle \phi(x), \phi(z) \rangle$ for some $\phi : \mathcal{X} \to \mathcal{H}$.

Inner products encode information about lengths and angles:
$||x - z||^2 = \langle x, x \rangle - 2 \langle x, z \rangle + \langle z, z \rangle , \qquad \cos\theta = \frac{\langle x,z \rangle}{||x|| \, ||z||}$.



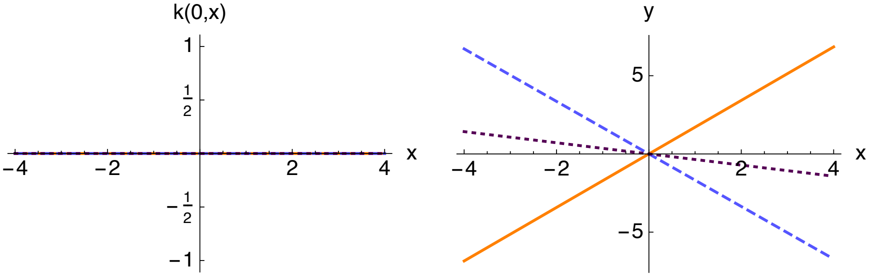
- well characterized function class
- closure properties
- access data only by $\boldsymbol{K}_{ij} = k(x_i, x_j)$
- $\mathcal{X}$ can be any non-empty set

Example: quadratic kernel

$\rightarrow$ blackboard

Examples of kernel functions

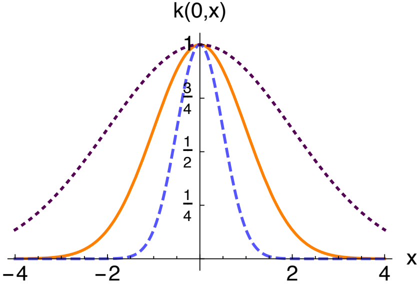
Linear kernel $k(\boldsymbol{x}, \boldsymbol{z}) = \langle \boldsymbol{x}, \boldsymbol{z} \rangle$

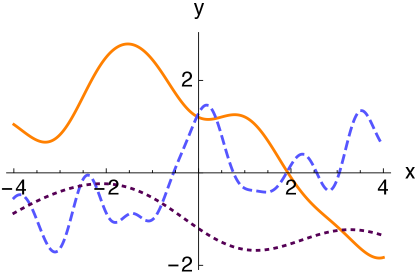


- recovers original linear model

Examples of kernel functions

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

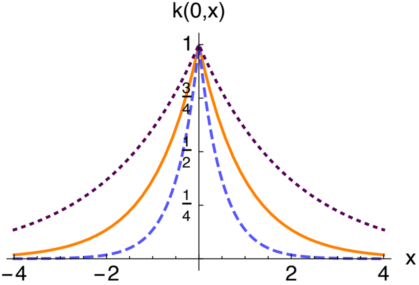


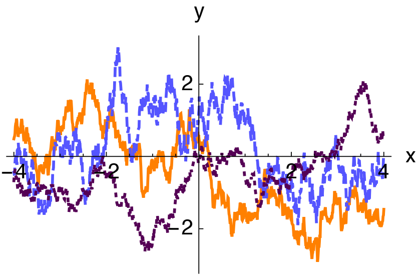


- length scale σ
- infinite dimensional feature space
- universal local approximator

Examples of kernel functions

Laplacian kernel $k(\boldsymbol{x}, \boldsymbol{z}) = \exp\left(-\frac{\|\boldsymbol{x} - \boldsymbol{z}\|_1}{\sigma}\right)$





- length scale σ

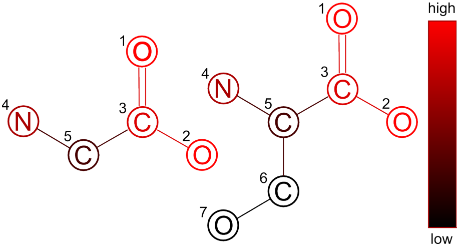
Example of a graph kernel

Iterative (graph) similarity optimal assignment kernel (ISOAK)

- $|V| \times |V'|$ matrix X of pairwise vertex similarities
- „two vertices are similar if their neighbours are similar"
- recursive definition; iterative computation
- find assignment $\rho : V \to V'$ such that $\sum_{i=1}^{|V|} X_{i,\rho(i)}$ is maximal

$10^2 X_{ij}$	1	2	3	4	5	6	7
1	98	50	00	00	00	00	50
2	50	98	11	34	16	17	89
3	00	11	96	14	68	78	13
4	00	34	14	91	13	20	38
5	00	24	67	17	81	77	20

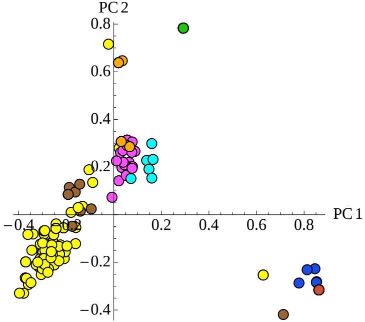
Pairwise atom similarities



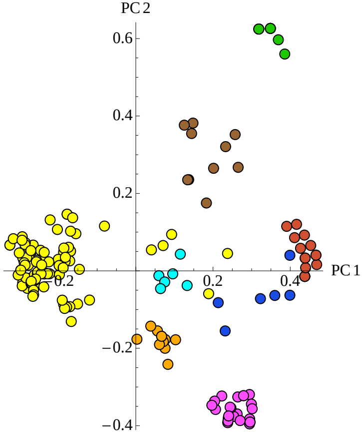
Glycine Serine

Rupp, Proschak, Schneider: *J. Chem. Inf. Model.*, 2280, 2007

Example of clustering with a graph kernel



Linear PCA with CATS2D



Kernel PCA with ISOAK

● tyrosines, ● TZDs, ● indoles, ● oxadiazoles, ● fatty acids,
● tertiary amides, ● tyrosines N, ● TZD-fatty acid hybrids

From linear regression to kernel ridge regression

- linear regression $\rightarrow$ blackboard
  problem, model form, optimization problem, solution
- ridge regression $\rightarrow$ blackboard
  correlated inputs, overfitting, "ridge" penalization, meaning
- kernel ridge regression $\rightarrow$ blackboard
  kernel trick, solution

Rupp, *Int. J. Quant. Chem.*, 1058, 2015
Hastie, Tibshirani, Friedman: Elements of Statistical Learning, Springer, 2009, pp. 168–169

Comparison of linear and kernel ridge regression

Ridge regression

Minimizing

$$\min_{\boldsymbol{\beta} \in \mathbb{R}^d} \sum_{i=1}^{n} \left(f(\boldsymbol{x_i}) - y_i\right)^2 + \lambda ||\boldsymbol{\beta}||^2$$

yields

$$\beta = \left(\boldsymbol{X}^T\boldsymbol{X} + \lambda\boldsymbol{I}\right)^{-1}\boldsymbol{X}^T\boldsymbol{y}$$

for models

$$f(\boldsymbol{x}) = \sum_{i=1}^{d} \beta_i \boldsymbol{x}_i$$

Kernel ridge regression

Minimizing

$$\min_{\boldsymbol{\alpha} \in \mathbb{R}^n} \sum_{i=1}^{n} \left(f(\boldsymbol{x_i}) - y_i\right)^2 + \lambda ||f||_{\mathcal{H}}^2$$

yields

$$\boldsymbol{\alpha} = \left(\boldsymbol{K} + \lambda\boldsymbol{I}\right)^{-1}\boldsymbol{y}.$$

for models

$$f(\boldsymbol{x}) = \sum_{i=1}^{n} \alpha_i k(\boldsymbol{x_i}, \boldsymbol{x})$$

Representer theorem

Kernel models have form

$$f(\boldsymbol{z}) = \sum_{i=1}^{n} \alpha_i k(\boldsymbol{x_i}, \boldsymbol{z})$$

due to the **representer theorem**:

Any function minimizing a regularized risk functional

$$\ell\Big(\big(\boldsymbol{x_i}, y_i, f(\boldsymbol{x_i})\big)_{i=1}^{n}\Big) + g(\|f\|)$$

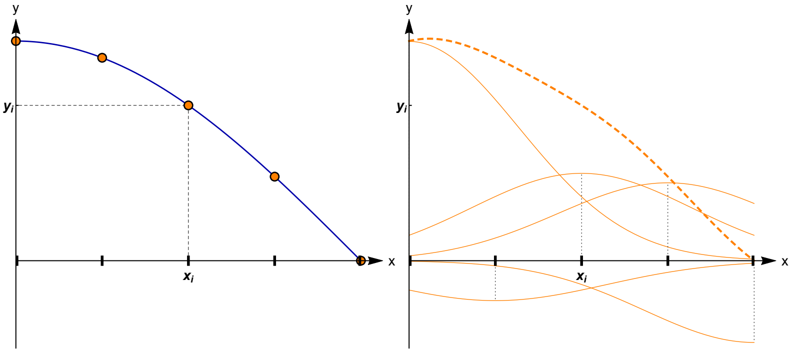
admits to above representation.

Intuition:

- model lives in space spanned by training data
- weighted sum of basis functions

Schölkopf, Herbrich & Smola, COLT 2001

The basis function picture

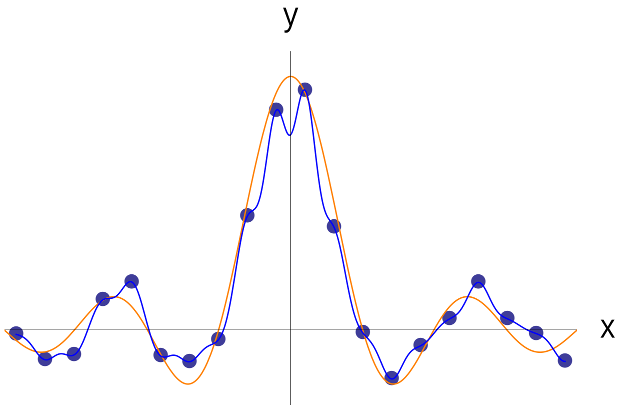


— learned $y = \cos(x)$ — basis functions

• training samples (x_i, y_i) - - - prediction f

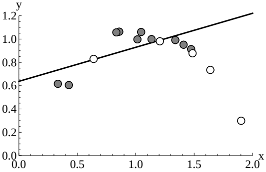
Vu et al, *Int J Quant Chem* 115: 1115, 2015

How regularization helps against overfitting

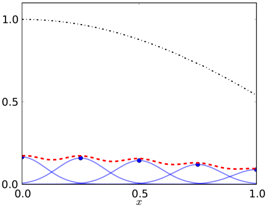


Effect of regularization

Underfitting

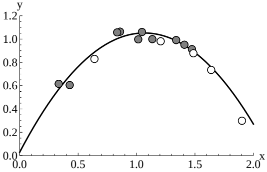


0.123 / 0.443

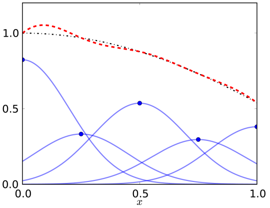


λ too large

Fitting

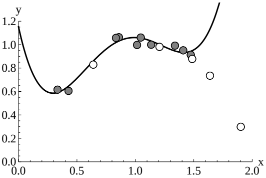


0.044 / 0.068

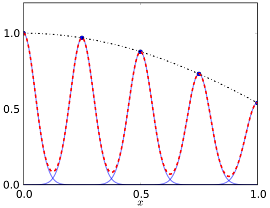


λ right

Overfitting



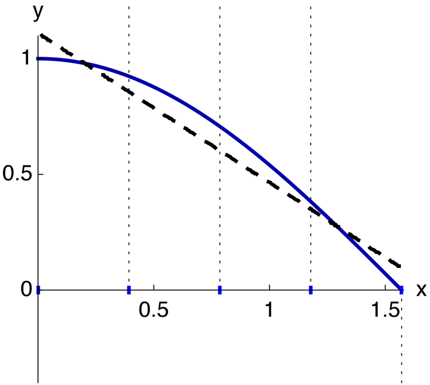
0.036 / 0.939



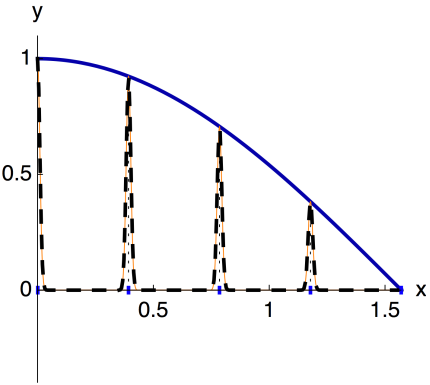
λ too small

Rupp, PhD thesis, 2009; Vu et al, *Int. J. Quant. Chem.*, 1115, 2015

Overfitting and underfitting in the limit



underfitting



overfitting

Centering in kernel feature space

Centering $\boldsymbol{X}$ and $\boldsymbol{y}$ is equivalent to having a bias term b.

For kernel models, center in kernel feature space:

$$\tilde{k}(\boldsymbol{x}, \boldsymbol{z}) = \left\langle \phi(\boldsymbol{x}) - \frac{1}{n}\sum_{i=1}^{n} \phi(\boldsymbol{x_i}), \phi(\boldsymbol{z}) - \frac{1}{n}\sum_{i=1}^{n} \phi(\boldsymbol{x_i}) \right\rangle$$

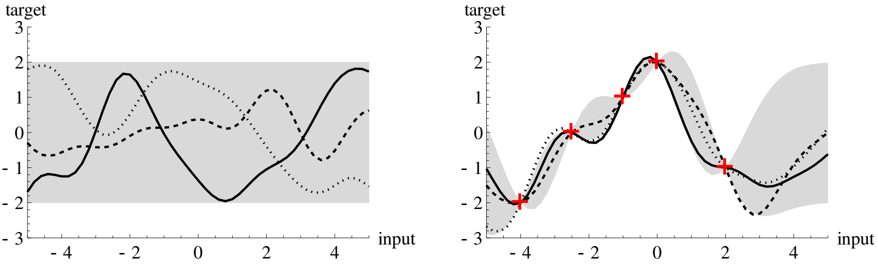
$$\Rightarrow \tilde{\boldsymbol{K}} = \Big(\boldsymbol{I} - \frac{1}{n}\mathbf{1}\Big)\boldsymbol{K}\Big(\boldsymbol{I} - \frac{1}{n}\mathbf{1}\Big)$$

Some kernels like Gaussian and Laplacian kernels do not need centering

Poggio *et al.*, Tech. Rep., 2001

Gaussian process regression

- generalization of multivariate normal distribution to functions
- determined by mean function and covariance function = kernel
- conditioning of prior on training data yields posterior distribution
- variance as confidence estimates for predictions



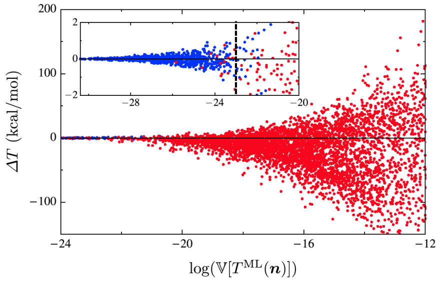
Rasmussen, Williams, MIT Press, 2006

Predictive variance

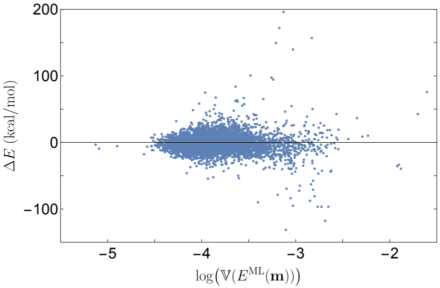
"It is not the estimate [...] that matters so much as the degree of confidence with the opinion"

Taleb, Random House, 2004

Works for some datasets, fails for others



Snyder et al, *Phys Rev Lett* 108, 2012



unpublished

Other kernel regression algorithms

- (kernel) support vector machines (SVM)
  Steinwart, Christmann, Springer, 2008
- kernel partial least squares (PLS)
  Rosipal, Trejo: *J. Mach. Learn. Res.*, 97, 2001
- **kernel ridge regression (KRR)**
  Hastie, Tibshirani, Friedman, Springer, 2009
- **Gaussian process regression (GPR)**
  Rasmussen, Williams, MIT Press, 2006

Summary

- the kernel trick: implicit transformation to high-dimensional spaces
- kernel ridge regression: regularized regression with kernels
- validation: avoid over-fitting by following the golden rule