Kernel Functions for Machine Learning Applications

In recent years, Kernel methods have received major attention, particularly due to the increased popularity of the Support Vector Machines. Kernel functions can be used in many applications as they provide a simple bridge from linearity to non-linearity for algorithms which can be expressed in terms of dot products. In this article, we will list a few kernel functions and some of their properties.

Many of these functions have been incorporated in Accord.NET [https://code.google.com/p/accord/] , a extension framework for the popular AForge.NET Framework [http://code.google.com/p/aforge/] which also includes many other statistics and machine learning tools [http://code.google.com/p/accord/wiki/SampleApplications?tm=6] .

Contents []

Kernel Methods []

Kernel methods are a class of algorithms for pattern analysis [http://en.wikipedia.org/wiki/Pattern_analysis] or recognition [http://en.wikipedia.org/wiki/Pattern_recognition] , whose best known element is the support vector machine [http://en.wikipedia.org/wiki/Support_vector_machine] (SVM). The general task of pattern analysis is to find and study general types of relations [http://en.wikipedia.org/wiki/Relation] (such as clusters [http://en.wikipedia.org/wiki/Cluster_analysis] , rankings [http://en.wikipedia.org/wiki/Ranking] , principal components [http://en.wikipedia.org/wiki/Principal_components] , correlations [http://en.wikipedia.org/wiki/Correlation] , classifications [http://en.wikipedia.org/wiki/Categorization]) in general types of data (such as sequences, text documents, sets of points, vectors, images, graphs, etc) (Wikipedia, 2010a).

The main characteristic of Kernel Methods, however, is their distinct approach to this problem. Kernel methods map the data into higher dimensional spaces in the hope that in this higher-dimensional space the data could become more easily separated or better structured. There are also no constraints on the form of this mapping, which could even lead to infinite-dimensional spaces. This mapping function, however, hardly needs to be computed because of a tool called the kernel trick [http://en.wikipedia.org/wiki/Kernel_trick] .

The Kernel Trick []

The kernel trick is a mathematical tool which can be applied to any algorithm which solely depends on the dot product [http://en.wikipedia.org/wiki/Dot_product] between two vectors. Wherever a dot product is used, it is replaced by a kernel function. When properly applied, those candidate linear algorithms are transformed into a non-linear algorithms (sometimes with little effort or reformulation). Those non-linear algorithms are equivalent to their linear originals operating in the range space of a feature space φ. However, because kernels are used, the φ function does not need to be ever explicitly computed. This is highly desirable, as we noted previously, because this higher-dimensional feature space could even be infinite-dimensional and thus infeasible to compute. There are also no constraints on the nature of the input vectors. Dot products could be defined between any kind of structure, such as trees or strings [http://en.wikipedia.org/wiki/String_kernel] .

Kernel Properties []

Kernel functions must be continuous, symmetric, and most preferably should have a positive (semi-) definite Gram matrix [http://mathworld.wolfram.com/GramMatrix.html] . Kernels which are said to satisfy the Mercer's theorem [http://en.wikipedia.org/wiki/Mercer's_theorem] are positive semi-definite [http://mathworld.wolfram.com/PositiveSemidefiniteMatrix.html] , meaning their kernel matrices have no non-negative Eigen values. The use of a positive definite kernel insures that the optimization problem will be convex and solution will be unique.

However, many kernel functions which aren't strictly positive definite also have been shown to perform very well in practice. An example is the Sigmoid kernel, which, despite its wide use, it is not positive semi-definite for certain values of its parameters. Boughorbel (2005) also experimentally demonstrated that Kernels which are only conditionally positive definite can possibly outperform most classical kernels in some applications [http://perso.lcpc.fr/tarel.jean-philippe/publis/jpt-icme05.pdf] .

Kernels also can be classified as anisotropic stationary, isotropic stationary, compactly supported, locally stationary, nonstationary or separable nonstationary [http://portal.acm.org/citation.cfm?id=944815] . Moreover, kernels can also be labeled scale-invariant or scale-dependant [http://en.wikipedia.org/wiki/Scale_invariance] , which is an interesting property as scale-invariant kernels drive the training process invariant to a scaling of the data [http://www.idiap.ch/~fleuret/papers/RR-4601.pdf] .

Choosing the Right Kernel []

Choosing the most appropriate kernel highly depends on the problem at hand - and fine tuning its parameters can easily become a tedious and cumbersome task. Automatic kernel selection is possible and is discussed in the works by Tom Howley and Michael Madden [http://www2.it.nuigalway.ie/m_madden/profile/pubs/ai_review_05.pdf] .

The choice of a Kernel depends on the problem at hand because it depends on what we are trying to model. A polynomial [http://en.wikipedia.org/wiki/Polynomial] kernel, for example, allows us to model feature conjunctions up to the order of the polynomial. Radial basis functions allows to pick out circles (or hyperspheres) - in constrast with the Linear kernel, which allows only to pick out lines (or hyperplanes [http://en.wikipedia.org/wiki/Hyperplane]).

The motivation behind the choice of a particular kernel can be very intuitive and straightforward depending on what kind of information we are expecting to extract about the data. Please see the final notes on this topic [http://nlp.stanford.edu/IR-book/html/htmledition/nonlinear-svms-1.html] from Introduction to Information Retrieval, by Manning, Raghavan and Schütze [http://nlp.stanford.edu/IR-book/html/htmledition/irbook.html] for a better explanation on the subject.

Kernel Functions []

Below is a list of some kernel functions available from the existing literature. As was the case with previous articles, every LaTeX notation [http://pt.wikipedia.org/wiki/LaTeX] for the formulas below are readily available from their alternate text html tag [http://en.wikipedia.org/wiki/Alt_attribute] . I can not guarantee all of them are perfectly correct, thus use them at your own risk. Most of them have links to articles where they have been originally used or proposed.

1. Linear Kernel []

The Linear kernel is the simplest kernel function. It is given by the inner product <x,y> plus an optional constant **c**. Kernel algorithms using a linear kernel are often equivalent to their non-kernel counterparts, i.e. KPCA [http://crsouza.blogspot.com/2010/01/kernel-principal-component-analysis-in.html] with linear kernel is the same as standard PCA [http://crsouza.blogspot.com/2009/09/principal-component-analysis-in-c.html] .

$$k(x, y) = x^T y + c$$

2. Polynomial Kernel []

The Polynomial kernel is a non-stationary kernel. Polynomial kernels are well suited for problems where all the training data is normalized.

$$k(x, y) = (\alpha x^T y + c)^d$$

Adjustable parameters are the slope **alpha**, the constant term **c** and the polynomial degree **d**.

3. Gaussian Kernel []

The Gaussian kernel is an example of radial basis function kernel.

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

Alternatively, it could also be implemented using

$$k(x, y) = \exp\left(-\gamma||x - y||^2\right)$$

The adjustable parameter **sigma** plays a major role in the performance of the kernel, and should be carefully tuned to the problem at hand. If overestimated, the exponential will behave almost linearly and the higher-dimensional projection will start to lose its non-linear power. In the other hand, if underestimated, the function will lack regularization and the decision boundary will be highly sensitive to noise in training data.

4. Exponential Kernel []

The exponential kernel is closely related to the Gaussian kernel, with only the square of the norm left out. It is also a radial basis function kernel.

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

5. Laplacian Kernel []

The Laplace Kernel is completely equivalent to the exponential kernel, except for being less sensitive for changes in the sigma parameter. Being equivalent, it is also a radial basis function kernel.

$$k(x, y) = \exp\left(-\frac{||x - y||}{\sigma}\right)$$

It is important to note that the observations made about the sigma parameter for the Gaussian kernel also apply to the Exponential and Laplacian kernels.

6. ANOVA Kernel []

The ANOVA kernel is also a radial basis function kernel, just as the Gaussian and Laplacian kernels. It is said to perform well in multidimensional regression problems [http://www.nicta.com.au/research/research_publications?sq_content_src=%2BdXJsPWh0dHBzJTNBJTJGJTJGcHVibGljYXRpb25zLmluc2lkZS5uaWN0YS5jb20uYXUlMkZzZWFyY2glMkZmdWxsdGV4dCUzRmlkJTNEMjYxJmFsbD0x] (Hofmann, 2008).

$$k(x, y) = \sum_{k=1}^{n} \exp(-\sigma(x^k - y^k)^2)^d$$

7. Hyperbolic Tangent (Sigmoid) Kernel []

The Hyperbolic Tangent Kernel is also known as the Sigmoid Kernel and as the Multilayer Perceptron (MLP) kernel. The Sigmoid Kernel comes from the Neural Networks [http://en.wikipedia.org/wiki/Neural_network] field, where the bipolar sigmoid function is often used as an activation function [http://en.wikipedia.org/wiki/Activation_function] for artificial neurons.

$$k(x, y) = \tanh(\alpha x^T y + c)$$

It is interesting to note that a SVM model using a sigmoid kernel function is equivalent to a two-layer, perceptron neural network. This kernel was quite popular for support vector machines due to its origin from neural network theory. Also, despite being only conditionally positive definite, it has been found to perform well in practice [http://perso.lcpc.fr/tarel.jean-philippe/publis/jpt-icme05.pdf] .

There are two adjustable parameters in the sigmoid kernel, the slope **alpha** and the intercept constant **c**. A common value for alpha is 1/N, where N is the data dimension. A more detailed study on sigmoid kernels can be found in the works by Hsuan-Tien and Chih-Jen [http://www.csie.ntu.edu.tw/~cjlin/papers/tanh.pdf] .

8. Rational Quadratic Kernel []

The Rational Quadratic kernel is less computationally intensive than the Gaussian kernel and can be used as an alternative when using the Gaussian becomes too expensive.

9. Multiquadric Kernel []

The Multiquadric kernel can be used in the same situations as the Rational Quadratic kernel. As is the case with the Sigmoid kernel, it is also an example of an non-positive definite kernel.

$$k(x, y) = \sqrt{\|x - y\|^2 + c^2}$$

10. Inverse Multiquadric Kernel []

The Inverse Multi Quadric kernel. As with the Gaussian kernel, it results in a kernel matrix with full rank (Micchelli, 1986 [http://www.springerlink.com/content/w62233k766460945/]) and thus forms a infinite dimension feature space.

$$k(x, y) = \frac{1}{\sqrt{\|x - y\|^2 + c^2}}$$

11. Circular Kernel []

The circular kernel is used in geostatic applications [http://perso.lcpc.fr/tarel.jean-philippe/publis/jpt-icann05b.pdf] . It is an example of an isotropic stationary kernel and is positive definite in R^2.

$$k(x, y) = \frac{2}{\pi} \arccos(-\frac{\|x - y\|}{\sigma}) - \frac{2}{\pi} \frac{\|x - y\|}{\sigma} \sqrt{1 - \left(\frac{\|x - y\|}{\sigma}\right)^2}$$

$$\text{if } \|x - y\| < \sigma, \text{ zero otherwise}$$

12. Spherical Kernel []

The spherical kernel is similar to the circular kernel, but is positive definite in R^3.

$$k(x, y) = 1 - \frac{3}{2} \frac{\|x - y\|}{\sigma} + \frac{1}{2} \left(\frac{\|x - y\|}{\sigma}\right)^3$$

$$\text{if } \|x - y\| < \sigma, \text{ zero otherwise}$$

13. Wave Kernel []

The Wave kernel is also symmetric positive semi-definite [http://www.lib.ncsu.edu/theses/available/etd-02262008-213801/unrestricted/etd.pdf] (Huang, 2008 [http://www.lib.ncsu.edu/theses/available/etd-02262008-213801/unrestricted/etd.pdf]).

$$k(x, y) = \frac{\theta}{\|x - y\|} \sin \frac{\|x - y\|}{\theta}$$

14. Power Kernel []

The Power kernel is also known as the (unrectified) triangular kernel. It is an example of scale-invariant kernel (Sahbi and Fleuret, 2004 [http://hal.archives-ouvertes.fr/docs/00/07/19/84/PDF/RR-4601.pdf]) and is also only conditionally positive definite.

$$k(x, y) = -\|x - y\|^d$$

15. Log Kernel []

The Log kernel seems to be particularly interesting for images, but is only conditionally positive definite.

$$k(x,y) = -log(||x-y||^d + 1)$$

16. Spline Kernel []

The Spline [http://en.wikipedia.org/wiki/Spline_(mathematics)] kernel is given as a piece-wise cubic polynomial, as derived in the works by Gunn (1998) [http://www.svms.org/tutorials/Gunn1998.pdf] .

$$k(x,y) = 1 + xy + xy\ min(x,y) - \frac{x+y}{2}\ min(x,y)^2 + \frac{1}{3}\min(x,y)^3$$

However, what it actually mean is:

$$k(x,y) = \prod_{i=1}^{d} 1 + x_i y_i + x_i y_i \min(x_i, y_i) - \frac{x_i + y_i}{2} \min(x_i, y_i)^2 + \frac{\min(x_i, y_i)^3}{3}$$

With

$$x, y \in R^d$$

17. B-Spline (Radial Basis Function) Kernel []

The B-Spline kernel is defined on the interval [−1, 1]. It is given by the recursive formula:

$$k(x,y) = B_{2p+1}(x-y)$$

$$\text{where } p \in N \text{ with } B_{i+1} := B_i \otimes B_0.$$

In the work by Bart Hamers [ftp://ftp.esat.kuleuven.ac.be/pub/SISTA/hamers/PhD_bhamers.pdf] it is given by:

$$k(x,y) = \prod_{p=1}^{d} B_{2n+1}(x_p - y_p)$$

Alternatively, B_n can be computed using the explicit expression (Fomel, 2000 [http://sepwww.stanford.edu/public/docs/sep105/sergey2/paper_html/node5.html]):

$$B_n(x) = \frac{1}{n!} \sum_{k=0}^{n+1} \binom{n+1}{k} (-1)^k (x + \frac{n+1}{2} - k)^n_+$$

Where x_+ is defined as the truncated power function [http://en.wikipedia.org/wiki/Truncated_power_function] :

$$x^d_+ = \begin{cases} x^d, & \text{if } x > 0 \\ 0, & \text{otherwise} \end{cases}$$

18. Bessel Kernel []

The Bessel [http://en.wikipedia.org/wiki/Bessel_function] kernel is well known in the theory of function spaces of fractional smoothness. It is given by:

$$k(x,y) = \frac{J_{v+1}(\sigma ||x-y||)}{||x-y||^{-n(v+1)}}$$

where J is the Bessel function of first kind [http://en.wikipedia.org/wiki/Bessel_function#Bessel_functions_of_the_first_kind_:_J.CE.B1] . However, in the Kernlab for R documentation [http://rss.acs.unt.edu/Rdoc/library/kernlab/html/dots.html] , the Bessel kernel is said to be:

$$k(x,x') = -Bessel^n_{(nu+1)}(\sigma |x - x'|^2)$$

19. Cauchy Kernel []

The Cauchy kernel comes from the Cauchy distribution [http://en.wikipedia.org/wiki/Cauchy_distribution] (Basak, 2008 [http://figment.cse.usf.edu/~sfefilat/data/papers/WeAT4.2.pdf]). It is a long-tailed kernel and can be used to give long-range influence and sensitivity over the high dimension space.

$$k(x, y) = \frac{1}{1 + \frac{\|x-y\|^2}{\sigma}}$$

20. Chi-Square Kernel []

The Chi-Square kernel comes from the Chi-Square distribution [http://en.wikipedia.org/wiki/Chi-square_distribution] .

$$k(x, y) = 1 - \sum_{i=1}^{n} \frac{(x_i - y_i)^2}{\frac{1}{2}(x_i + y_i)}$$

21. Histogram Intersection Kernel []

The Histogram Intersection Kernel is also known as the Min Kernel and has been proven useful in image classification.

$$k(x, y) = \sum_{i=1}^{n} \min(x_i, y_i)$$

22. Generalized Histogram Intersection []

The Generalized Histogram Intersection kernel is built based on the Histogram Intersection Kernel [http://www-video.eecs.berkeley.edu/Proceedings/ICIP2003/papers/cr1967.pdf] for image classification but applies in a much larger variety of contexts (Boughorbel, 2005 [http://perso.lcpc.fr/tarel.jean-philippe/publis/jpt-icip05.pdf]). It is given by:

$$k(x, y) = \sum_{i=1}^{m} \min(|x_i|^{\alpha}, |y_i|^{\beta})$$

23. Generalized T-Student Kernel []

The Generalized T-Student Kernel has been proven to be a Mercel Kernel [http://ralyx.inria.fr/2004/Raweb/imedia/uid84.html] , thus having a positive semi-definite Kernel matrix (Boughorbel, 2004 [http://ralyx.inria.fr/2004/Raweb/imedia/uid84.html]). It is given by:

$$k(x, y) = \frac{1}{1 + \|x - y\|^d}$$

24. Bayesian Kernel []

The Bayesian kernel could be given as:

$$k(x, y) = \prod_{l=1}^{N} \kappa_l(x_l, y_l)$$

where

$$\kappa_l(a, b) = \sum_{c \in \{0;1\}} P(Y = c \mid X_l = a)\ P(Y = c \mid X_l = b)$$

However, it really depends on the problem being modeled. For more information, please see the work by Alashwal, Deris and Othman [http://www.waset.org/journals/ijci/v5/v5-2-14.pdf] , in which they used a SVM with Bayesian kernels in the prediction of protein-protein interactions.

25. Wavelet Kernel []

The Wavelet kernel (Zhang et al, 2004 [http://see.xidian.edu.cn/faculty/zhangli/publications/WSVM.pdf]) comes from Wavelet theory [http://en.wikipedia.org/wiki/Wavelet] and is given as:

$$k(x, y) = \prod_{i=1}^{N} h(\frac{x_i - c}{a})\ h(\frac{y_i - c}{a})$$

Where **a** and **c** are the wavelet dilation and translation coefficients, respectively (the form presented above is a simplification, please see the original paper for details). A translation-invariant version of this kernel can be given as:

$$k(x, y) = \prod_{i=1}^{N} h(\frac{x_i - y_i}{a})$$

Where in both h(x) denotes a mother wavelet function. In the paper by Li Zhang, Weida Zhou, and Licheng Jiao, the authors suggests a possible h(x) as:

$$h(x) = cos(1.75x)exp(-\frac{x^2}{2})$$

Which they also prove as an admissible kernel function.

Source Code []

The latest version of the source code for almost all of the kernels listed above [https://code.google.com/p/accord/source/browse/trunk#trunk%2FSources%2FAccord.Statistics%2FKernels] is available in the Accord.NET Framework [https://code.google.com/p/accord/] . Some are also available in the sequel of this article, Kernel Support Vector Machines for Classification and Regression in C# [http://crsouza.blogspot.com/2010/04/kernel-support-vector-machines-for.html] . They are provided together with a comprehensive and simple implementation of SVMs (Support Vector Machines) in C#. However, for the latest sources, which may contain bug fixes and other enhancements, please download the most recent version available of Accord.NET.

See also []

- Kernel Support Vector Machines (kSVMs) [http://crsouza.blogspot.com/2010/04/kernel-support-vector-machines-for.html]
- Principal Component Analysis (PCA) [http://crsouza.blogspot.com/2009/09/principal-component-analysis-in-c.html]
- Kernel Principal Component Analysis (KPCA) [http://crsouza.blogspot.com/2010/01/kernel-principal-component-analysis-in.html]
- Linear Discriminant Analysis (LDA) [http://crsouza.blogspot.com/2010/01/linear-discriminant-analysis-in-c.html]
- Non-Linear Discriminant Analysis with Kernels (KDA) [http://crsouza.blogspot.com/2010/01/kernel-discriminant-analysis-in-c.html]
- Logistic Regression Analysis in C# [http://crsouza.blogspot.com/2010/02/logistic-regression-in-c.html]
- The Accord.NET Framework: Scientific Computing in .NET [https://code.google.com/p/accord/]
- Haar-feature object detection in C# (The Viola-Jones Classifier) [http://www.codeproject.com/Articles/441226/Haar-feature-Object-Detection-in-Csharp]
- Handwriting Recognition using Kernel Discriminant Analysis [/KB/recipes/handwriting-kda.aspx]
- Handwriting Recognition Revisited: Kernel Support Vector Machines [http://www.codeproject.com/Articles/106583/Handwriting-Recognition-Revisited-Kernel-Support-V]
- Logistic Regression Analysis [http://crsouza.blogspot.com/2010/02/logistic-regression-in-c.html]

References []

- On-Line Prediction Wiki Contributors. "Kernel Methods." On-Line Prediction Wiki. http://onlineprediction.net/?n=Main.KernelMethods [http://onlineprediction.net/?n=Main.KernelMethods] (accessed March 3, 2010).

- Genton, Marc G. "Classes of Kernels for Machine Learning: A Statistics Perspective [http://jmlr.csail.mit.edu/papers/volume2/genton01a/genton01a.pdf] ." Journal of Machine Learning Research 2 (2001) 299-312.

- Hofmann, T., B. Schölkopf, and A. J. Smola. "Kernel methods in machine learning [http://www.nicta.com.au/research/research_publications?sq_content_src=%2BdXJsPWh0dHBzJTNBJTJGJTJGcHVibGljYXRpb25zLmluc2lkZS5uaWN0YS5jb20uYXUlMkZzZWFyY2glMkZmdWxsdGV4dCUzRmlkJTNEMjYxJmFsbD0x] ." Ann. Statist. Volume 36, Number 3 (2008), 1171-1220.

- Gunn, S. R. (1998, May). "Support vector machines for classification and regression [http://users.ecs.soton.ac.uk/srg/publications/pdf/SVM.pdf] ." Technical report, Faculty of Engineering, Science and Mathematics School of Electronics and Computer Science.

- Karatzoglou, A., Smola, A., Hornik, K. and Zeileis, A. "Kernlab – an R package for kernel Learning [http://statmath.wu.ac.at/~zeileis/papers/Ensemble-2005.pdf] ." (2004).
- Karatzoglou, A., Smola, A., Hornik, K. and Zeileis, A. "Kernlab – an S4 package for kernel methods in R [http://www.jstatsoft.org/v11/i09/paper] ." J. Statistical Software, 11, 9 (2004).
- Karatzoglou, A., Smola, A., Hornik, K. and Zeileis, A. "R: Kernel Functions [http://rss.acs.unt.edu/Rdoc/library/kernlab/html/dots.html] ." Documentation for package 'kernlab' version 0.9-5. http://rss.acs.unt.edu/Rdoc/library/kernlab/html/dots.html [http://rss.acs.unt.edu/Rdoc/library/kernlab/html/dots.html] (accessed March 3, 2010).
- Howley, T. and Madden, M.G. "The genetic kernel support vector machine: Description and evaluation [http://www2.it.nuigalway.ie/m_madden/profile/pubs/ai_review_05.pdf] ". Artificial Intelligence Review. Volume 24, Number 3 (2005), 379-395.
- Shawkat Ali and Kate A. Smith. "Kernel Width Selection for SVM Classification: A Meta-Learning Approach [http://www.infosci-journals.com/downloadPDF/pdf/ITJ2950_WVB69NWMA8.pdf] ." International Journal of Data Warehousing & Mining, 1(4), 78-97, October-December 2005.
- Hsuan-Tien Lin and Chih-Jen Lin. "A study on sigmoid kernels for SVM and the training of non-PSD kernels by SMO-type methods [http://www.csie.ntu.edu.tw/~cjlin/papers/tanh.pdf] ." Technical report, Department of Computer Science, National Taiwan University, 2003.
- Boughorbel, S., Jean-Philippe Tarel, and Nozha Boujemaa. "Project-Imedia: Object Recognition [http://ralyx.inria.fr/2004/Raweb/imedia/uid84.html] ." INRIA - INRIA Activity Reports - RalyX. http://ralyx.inria.fr/2004/Raweb/imedia/uid84.html [http://ralyx.inria.fr/2004/Raweb/imedia/uid84.html] (accessed March 3, 2010).
- Huang, Lingkang. "Variable Selection in Multi-class Support Vector Machine and Applications in Genomic Data Analysis [http://www.lib.ncsu.edu/theses/available/etd-02262008-213801/unrestricted/etd.pdf] ." *PhD Thesis*, 2008.
- Manning, Christopher D., Prabhakar Raghavan, and Hinrich Schütze. "Nonlinear SVMs [http://nlp.stanford.edu/IR-book/html/htmledition/nonlinear-svms-1.html] ." The Stanford NLP (Natural Language Processing) Group. http://nlp.stanford.edu/IR-book/html/htmledition/nonlinear-svms-1.html [http://nlp.stanford.edu/IR-book/html/htmledition/nonlinear-svms-1.html] (accessed March 3, 2010).
- Fomel, Sergey. "Inverse B-spline interpolation [http://sepwww.stanford.edu/public/docs/sep105/sergey2/paper_html/node5.html] ." *Stanford Exploration Project*, 2000. http://sepwww.stanford.edu/public/docs/sep105/sergey2/paper_html/node5.html [http://sepwww.stanford.edu/public/docs/sep105/sergey2/paper_html/node5.html] (accessed March 3, 2010).
- Basak, Jayanta. "A least square kernel machine with box constraints [http://figment.cse.usf.edu/~sfefilat/data/papers/WeAT4.2.pdf] ." *International Conference on Pattern Recognition 2008* 1 (2008): 1-4.
- Alashwal, H., Safaai Deris, and Razib M. Othman. "A Bayesian Kernel for the Prediction of Protein - Protein Interactions [http://www.waset.org/journals/ijci/v5/v5-2-14.pdf] ." *International Journal of Computational Intelligence* 5, no. 2 (2009): 119-124.
- Hichem Sahbi and François Fleuret. "Kernel methods and scale invariance using the triangular kernel [http://hal.archives-ouvertes.fr/docs/00/07/14/40/PDF/RR-5143.pdf] ". INRIA Research Report, N-5143, March 2004.
- Sabri Boughorbel, Jean-Philippe Tarel, and Nozha Boujemaa. "Generalized histogram intersection kernel for image recognition [http://perso.lcpc.fr/tarel.jean-philippe/publis/jpt-icip05.pdf] ". *Proceedings of the 2005 Conference on Image Processing*, volume 3, pages 161-164, 2005.
- Micchelli, Charles. Interpolation of scattered data: Distance matrices and conditionally positive definite functions. [http://www.springerlink.com/content/w62233k766460945/] *Constructive Approximation* 2, no. 1 (1986): 11-22.
- Wikipedia contributors, "Kernel methods," *Wikipedia, The Free Encyclopedia,* http://en.wikipedia.org/w/index.php?title=Kernel_methods&oldid=340911970 [http://en.wikipedia.org/w/index.php?title=Kernel_methods&oldid=340911970] (accessed March 3, 2010).
- Wikipedia contributors, "Kernel trick," *Wikipedia, The Free Encyclopedia,* http://en.wikipedia.org/w/index.php?title=Kernel_trick&oldid=269422477 [http://en.wikipedia.org/w/index.php?title=Kernel_trick&oldid=269422477] (accessed March 3, 2010).

- Weisstein, Eric W. [http://mathworld.wolfram.com/about/author.html] "Positive Semidefinite Matrix." From *MathWorld* [http://mathworld.wolfram.com/] --A Wolfram Web Resource. http://mathworld.wolfram.com/PositiveSemidefiniteMatrix.html [http://mathworld.wolfram.com/PositiveSemidefiniteMatrix.html]
- Hamers B. "Kernel Models for Large Scale Applications [http://www.predictive-analytics.be/files/PhD_bhamers.pdf] ", *Ph.D.* , Katholieke Universiteit Leuven, Belgium, 2004.
- Li Zhang, Weida Zhou, Licheng Jiao. Wavelet Support Vector Machine [http://see.xidian.edu.cn/faculty/zhangli/publications/WSVM.pdf] . IEEE Transactions on System, Man, and Cybernetics, Part B, 2004, 34(1): 34-39.

Citing this work

If you would like, please cite this work as: Souza, César R. "Kernel Functions for Machine Learning Applications." 17 Mar. 2010. Web. <http://crsouza.blogspot.com/2010/03/kernel-functions-for-machine-learning.html>.

Posted 18th March 2010 by César Souza

Labels: Statistics, Mathematics, C#

23 View comments

mwiley63 April 25, 2010 at 1:10 AM

Great article!

I am just learning about perceptrons and kernels, and found this information very helpful.

When will you be able to post some source code? I am trying to program a kernel perceptron right now, but am unable to figure out how to build the kernel function. I am having trouble connecting my example to the theory of the kernel function.

Reply

César Souza April 26, 2010 at 8:48 AM

Hi,

Soon I'm going to post a new article about (Kernel) Support Vector Machines, which will also contain the code for most Kernels presented here.

However, I just updated my Kernel Discriminant Analysis code and also included some of them there. You could try checking it, if you wish.

Good luck,
César

Reply

Uday May 29, 2010 at 7:45 AM

César
Thanks for nice article. I wanted to find the parameter range for each like sigma in Cauchy, Laplace etc do they vary in all the real number range? To optimize it.
Also theta mentioned does it vary from 0 to 360?
Thanks in advance
Uday

Reply

César Souza May 29, 2010 at 10:40 AM

Hi Uday,

In the paper Practical Selection of SVM Parameters and Noise Estimation for SVM Regression the authors have taken sigma values in the range (0.2~0.5)*range(x) for the Gaussian kernel, x being their input data. If the input data was normalized to be in the [0,1] range, then perhaps good choices for sigma would lie in the [0.2,0.5] range.

But this is for the Gaussian kernel. I have linked the original sources for each kernel in the post, perhaps you may found additional information about those kernels in their original paper.

The theta values in the Multiquadric kernels are just positive constants. Perhaps I should change the symbols to something else. For the Wave kernel, perhaps good values would be in the 0~2*pi range.

In most cases, a grid search would be required to find the most suitable Kernel parameters. In this page you can find more directions for parameter tuning in SVMs.

Regards,
César

Reply

Uday May 30, 2010 at 11:57 AM

Thanks César for link and suggestions. For Quadratic, Inverse Quadratic and MutiQuadratic the constant will be like shifting the axis? Do you think moving it in real range 0 to RealMax would have impact on learning?
Uday

Reply

César Souza May 31, 2010 at 2:13 PM

I am not very sure of the exact interpretation of the constant parameter in those Kernels, but its value will probably affect learning. That is why a grid search or a pattern search is advisable in this case.

Regards,
César

Reply

Zhao July 13, 2010 at 5:19 AM

Nice article, very comprehensive for the available kernel functions! Suggestion: the wavelet kernel may be added in the list.

Reply

Zhao July 15, 2010 at 8:24 PM

I am dubious about the form of the ANOVA kernel mentioned in the above article. The summation is right?

Reply

César Souza August 17, 2010 at 9:49 PM

Just cleaning some old comments to avoid confusion. There was a problem with the ANOVA kernel and it has been corrected, thanks to Zhao. A preliminary version of the Wavelet kernel will also be available soon.

Cheers!
César

Reply

Anonymous January 21, 2011 at 3:54 PM

Thanks for the tool; a great work in fact. Can you please add the RBF kernel? it seems to be widely used.
Thanks

Reply

César Souza January 21, 2011 at 7:25 PM

Hi Anonymous,

The RBF is not exactly a kernel but a family of kernels sharing the same common form: k(x,y) = exp(-gamma*||x-y||^2), with gamma being a positive constant. It is the same form depicted in the alternate form of the Gaussian kernel in this post. The most popular member of the RBF family of kernels is the Gaussian kernel, which can be derived from the general RBF form by taking gamma = 1/(sigma^2).

Best regards,
César

Reply

Abhinav Maurya April 11, 2011 at 2:55 AM

The formulation of Circulation Kernel is slightly incorrect. Particularly the square inside the square-root applies to numerator and denominator both.

Reply

Anonymous April 14, 2011 at 12:40 AM

u done great job. awesome. do you have source code for genetic kernel for support vector machine in java or Simple SMO in java

Reply

César Souza April 16, 2011 at 1:50 PM

Hi Abhinav,

Thanks for noticing. I have updated the article with the correction.

@Anonymous: No, unfortunately I do not have any code in Java. Perhaps you could try taking a look on Weka.

Best Regards,
César

Reply

Anonymous September 27, 2011 at 10:45 AM

I have a question about the multiquadratic kernel. Usually a kernel value is larger, if the vectors are more similar to each other. But with the multiquadratic kernel, it is just the opposite. Can this work? Do you have to switch the classification value of the SVM?

Reply

César Souza September 28, 2011 at 1:30 PM

Hi Anonymous,

Indeed, the Multiquadric kernel is not positive definite. Actually, it is a negative-definite kernel. I have included it for theoretical interest in presenting the Inverse Multiquadric kernel, but I have not applied it to any dataset to see how it performs.

However, the interesting thing to note is that some non-positive (semi-)definite kernels have been used with relative success in practice, albeit it isn't so clear in which situations they work. In some cases, an special or modified version of the learning algorithm which can deal with non-PSD kernels is required.

For further information, perhaps you can try taking a look on the work by Hsuan-Tien Lin and Chih-Jen Lin for some detailed discussion on the use of the (non-PSD) sigmoid kernel.

All the best,
César

Reply

Anonymous October 11, 2011 at 3:11 PM

Very nice and helpful article!!
Thanks.

Reply

Android apps development November 19, 2011 at 2:39 AM

Thanks for sharing your info. I really appreciate your efforts and I will be waiting for your further write ups thanks once again.
Android app development| Android app developer|

Reply

Anonymous December 3, 2011 at 6:55 PM

Hi César Souza,

Thanks for this great article. Can you please upload the MATLAB codes for SVM Training & classification with Gaussian Kernel for imagr datasets. I want to classify the images into different emotions.

Reply

madhuri January 30, 2012 at 3:16 AM

Once in my college studies, I got to know about kernel function, and now thru ur blog I remember that and even got to know more about it.Cheers !!
Web Shopping Cart | Free Ecommerce Software

Reply

François-Xavier Thomas March 5, 2012 at 1:30 PM

Just one word : wow!
I have a course on kernel methods in college, but we were lacking this list of known useful kernels, so thanks for taking the time to post this!

Reply

Cool boy August 24, 2012 at 5:18 AM

I'm surely coming again to read these articles and blogs
injection molding machines

Reply

sinan ABO ALCHAMLAT February 6, 2013 at 7:35 AM

Nice article, you done great job, I have a question how i can compare between two function Kernel (Gaussian Kernel, IBS Kernel)

Reply

Enter your comment...

Comment as: Google Account

Publish Preview