Kernelization

David Rosenberg

New York University

February 24, 2016

Generalizing from SVM

Soft-Margin SVM (no intercept)

- The SVM objective function is

$$\frac{1}{2}\|w\|^2 + \frac{c}{n}\sum_{i=1}^{n}\left(1 - y_i\left[w^T x_i\right]\right)_+ .$$

- We found that the minimizer $w^* \in \mathbf{R}^d$ has the form

$$w^* = \sum_{i=1}^{n} \alpha_i^* x_i .$$

- **Representer Theorem** $\implies$ same result in a much broader context.

Introduce a Feature Map

- Input space: $\mathcal{X}$ (no assumptions).
- Feature space: $\mathcal{H}$ (a Hilbert space, usually $\mathbf{R}^d$) .
- Feature map $\psi : \mathcal{X} \to \mathcal{H}$.
- Featurized SVM objective:

$$\min_{w \in \mathcal{H}} \frac{1}{2}\|w\|^2 + \frac{c}{n}\sum_{i=1}^{n}\left(1 - y_i\left[\langle w, \psi(x_i)\rangle\right]\right)_+ .$$

- Now $\|w\|^2 = \langle w, w \rangle$, where $\langle \cdot, \cdot \rangle$ is inner product for $\mathcal{H}$.
- Note that minimizer $w^* \in \mathcal{H}$. What are predictions $x \mapsto ?$

Generalize

- Featurized SVM objective:

$$\min_{w \in \mathcal{H}} \frac{1}{2}\|w\|^2 + \frac{c}{n}\sum_{i=1}^{n}\left(1 - y_i\left[\langle w, \psi(x_i)\rangle\right]\right)_+ .$$

- **Generalized objective**:

$$\min_{w \in \mathcal{H}} R(\|w\|) + L(\langle w, \psi(x_1)\rangle, \ldots, \langle w, \psi(x_n)\rangle),$$

where

  - $R : \mathbf{R}^{\geqslant 0} \to \mathbf{R}$ is nondecreasing (**Regularization term**)
  - and $L : \mathbf{R}^n \to \mathbf{R}$ is arbitrary. (**Loss term**)

Generalized Objective Function

- **Generalized objective**:

$$\min_{w \in \mathcal{H}} R\left(\|w\|\right) + L\left(\langle w, \psi(x_1)\rangle, \ldots, \langle w, \psi(x_n)\rangle\right),$$

  where

  - $R : \mathbf{R}^{\geqslant 0} \to \mathbf{R}$ is nondecreasing (**Regularization term**), and
  - $L : \mathbf{R}^n \to \mathbf{R}$ is arbitrary (**Loss term**).

- Is ridge regression of this form? What is $R(\cdot)$?
- What if we penalize with $\lambda\|w\|_2$ instead of $\lambda\|w\|_2^2$?
- What if we use lasso regression?

The Representer Theorem

The Representer Theorem

Theorem (Representer Theorem)

Let

$$J(w) = R\left(\|w\|\right) + L\left(\langle w, \psi(x_1)\rangle, \ldots, \langle w, \psi(x_n)\rangle\right),$$

where

- $R : \mathbf{R}^{\geqslant 0} \to \mathbf{R}$ *is nondecreasing (**Regularization term**), and*
- $L : \mathbf{R}^n \to \mathbf{R}$ *is arbitrary (**Loss term**).*

Then there is a minimizer of $J(w)$ of the form

$$w^* = \sum_{i=1}^{n} \alpha_i \psi(x_i).$$

[If R is strictly increasing, then all minimizers have this form.]

The Representer Theorem (Proof)

1. Let w^* be a minimizer.
2. Let $M = \text{span}\left(\psi(x_1), \ldots, \psi(x_n)\right)$. [the **"span of the data"**]
3. Let $w = \text{Proj}_M w^*$. So $\exists \alpha$ s.t. $w = \sum_{i=1}^n \alpha_i \psi(x_i)$.
4. Then $w^\perp := w^* - w$ is orthogonal to M.
5. Projections decrease norms: $\|w\| \leqslant \|w^*\|$.
6. Since R is nondecreasing, $R(\|w\|) \leqslant R(\|w^*\|)$.
7. By (4), $\langle w^*, \psi(x_i) \rangle = \left\langle w + w^\perp, \psi(x_i) \right\rangle = \langle w, \psi(x_i) \rangle$.
8. $L\left(\langle w^*, \psi(x_1) \rangle, \ldots, \langle w^*, \psi(x_n) \rangle\right) = L\left(\langle w, \psi(x_1) \rangle, \ldots, \langle w, \psi(x_n) \rangle\right)$
9. $J(w) \leqslant J(w^*)$.
10. Therefore $w = \sum_{i=1}^n \alpha_i \psi(x_i)$ is also a minimizer.

Q.E.D.

Representer Theorem for Kernelization

Kernelized Predictions

- Consider $w = \sum_{i=1}^{n} \alpha_i \psi(x_i)$.
- How do we make predictions for a given $x \in \mathcal{X}$?

$$
\begin{aligned}
f(x) = \langle w^*, \psi(x) \rangle &= \left\langle \sum_{i=1}^{n} \alpha_i \psi(x_i), \psi(x) \right\rangle \\
&= \sum_{i=1}^{n} \alpha_i \langle \psi(x_i), \psi(x) \rangle \\
&= \sum_{i=1}^{n} \alpha_i k(x_i, x)
\end{aligned}
$$

Kernelized Regularization

- Consider $w=\sum_{i=1}^{n}\alpha_i\psi(x_i)$.
- What does $R(\|w\|)$ look like?

$$
\begin{aligned}
\|w\|^2 &= \langle w,w\rangle \\
&= \left\langle \sum_{i=1}^{n}\alpha_i\psi(x_i),\sum_{j=1}^{n}\alpha_j\psi(x_j)\right\rangle \\
&= \sum_{i,j=1}^{n}\alpha_i\alpha_j\langle\psi(x_i),\psi(x_j)\rangle \\
&= \sum_{i,j=1}^{n}\alpha_i\alpha_j k(x_i,x_j)
\end{aligned}
$$

(You should recognize the last expression as a quadratic form.)

The Kernel Matrix (a.k.a. Gram Matrix)

Definition

The **kernel matrix** for a kernel k on a set $\{x_1, \ldots, x_n\}$ is

$$K = \left(k(x_i, x_j)\right)_{i,j} = \begin{pmatrix} k(x_1, x_1) & \cdots & k(x_1, x_n) \\ \vdots & \ddots & \cdots \\ k(x_n, x_1) & \cdots & k(x_n, x_n) \end{pmatrix} \in \mathbf{R}^{n \times n}.$$

This matrix is also known as the **Gram matrix**.

Kernelized Regularization: Matrix Form

- Consider $w = \sum_{i=1}^{n} \alpha_i \psi(x_i)$.
- What does $R(\|w\|)$ look like?

$$\begin{aligned} \|w\|^2 &= \sum_{i,j=1}^{n} \alpha_i \alpha_j k(x_i, x_j) \\ &= \alpha^T K \alpha \end{aligned}$$

- So $R(\|w\|) = R\left(\sqrt{\alpha^T K \alpha}\right)$.

Kernelized Predictions

- $R(\|w\|) + L(\langle w, \psi(x_1)\rangle, \ldots, \langle w, \psi(x_n)\rangle)$ involves $f(x_i) = \langle w, \psi(x_i)\rangle$.
- Note that

$$
\begin{aligned}
K\alpha &= \begin{pmatrix} k(x_1,x_1) & \cdots & k(x_1,x_n) \\ \vdots & \ddots & \cdots \\ k(x_n,x_1) & \cdots & k(x_n,x_n) \end{pmatrix} \begin{pmatrix} \alpha_1 \\ \vdots \\ \alpha_n \end{pmatrix} \\
&= \begin{pmatrix} \alpha_1 k(x_1,x_1) + \cdots + \alpha_n k(x_1,x_n) \\ \vdots \\ \alpha_1 k(x_n,x_1) + \cdots + \alpha_n k(x_{1,}x_n) \end{pmatrix} \\
&= \begin{pmatrix} f_\alpha(x_1) \\ \vdots \\ f_\alpha(x_n) \end{pmatrix}.
\end{aligned}
$$

Kernelized Objective

- Substituting

$$w = \sum_{i=1}^{n} \alpha_i \psi(x_i)$$

into generalized objective, we get

$$\min_{\alpha \in \mathbf{R}^n} R\left(\sqrt{\alpha^T K \alpha}\right) + L(K\alpha).$$

- No direct access to $\psi(x_i)$.
- All references are via kernel matrix K.
- (Assumes R and L do not hide any references to $\psi(x_i)$.)
- This is the **kernelized objective function.**

Kernelized SVM

- The SVM objective:

$$\min_{w \in \mathcal{H}} \frac{1}{2}\|w\|^2 + \frac{c}{n}\sum_{i=1}^{n}\left(1 - y_i\left[\langle w, \psi(x_i)\rangle\right]\right)_+.$$

- Kernelizing yields

$$\min_{\alpha \in \mathbf{R}^n} \frac{1}{2}\alpha^T K\alpha + \frac{c}{n}\sum_{i=1}^{n}\left(1 - y_i\left(K\alpha\right)_i\right)_+$$

Kernelized Ridge Regression

- Ridge Regression:

$$\min_{w \in \mathbf{R}^d} \frac{1}{n} \sum_{i=1}^{n} \left(w^T x_i - y_i\right)^2 + \lambda \|w\|^2$$

- Featurized Ridge Regression

$$\min_{w \in \mathcal{H}} \frac{1}{n} \sum_{i=1}^{n} \left(\langle w, \psi(x_i) \rangle - y_i\right)^2 + \lambda \|w\|^2$$

- Kernelized Ridge Regression

$$\min_{\alpha \in \mathbf{R}^n} \frac{1}{n} \|K\alpha - y\|^2 + \lambda \alpha^T K \alpha,$$

where $y = (y_1, \ldots, y_n)^T$.

Kernel Examples

SVM Dual

- Recall the SVM dual optimization problem

$$\begin{aligned}
\sup_{\alpha} \quad & \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{n} \alpha_i \alpha_j y_i y_j x_j^T x_i \\
\text{s.t.} \quad & \sum_{i=1}^{n} \alpha_i y_i = 0 \\
& \alpha_i \in \left[0, \frac{c}{n}\right] \; i = 1, \ldots, n.
\end{aligned}$$

- Notice: x's only show up as inner products with other x's.
- Can replace $x_j^T x_i$ by an arbitrary kernel $k(x_j, x_i)$.
- What kernel are we currently using?

Linear Kernel

- Input space: $\mathcal{X} = \mathbf{R}^d$
- Feature space: $\mathcal{H} = \mathbf{R}^d$, with standard inner product
- Feature map
$$\psi(x) = x.$$
- Kernel:
$$k(w, x) = w^T x$$

Quadratic Kernel in $\mathbf{R}^2$

- Input space: $\mathcal{X} = \mathbf{R}^2$
- Feature space: $\mathcal{H} = \mathbf{R}^5$
- Feature map:

$$\psi : (x_1, x_2) \mapsto \left(x_1, x_2, x_1^2, x_2^2, \sqrt{2} x_1 x_2 \right)$$

- Gives us ability to represent conic section boundaries.
- Define kernel as inner product in feature space:

$$\begin{aligned} k(w,x) &= \langle \psi(w), \psi(x) \rangle \\ &= w_1 x_1 + w_2 x_2 + w_1^2 x_1^2 + w_2^2 x_2^2 + 2 w_1 w_2 x_1 x_2 \\ &= w_1 x_1 + w_2 x_2 + (w_1 x_1)^2 + (w_2 x_2)^2 + 2(w_1 x_1)(w_2 x_2) \\ &= \langle w, x \rangle + \langle w, x \rangle^2 \end{aligned}$$

Based on Guillaume Obozinski's Statistical Machine Learning course at Louvain, Feb 2014.

Quadratic Kernel in $\mathbf{R}^d$

- Input space $\mathcal{X} = \mathbf{R}^d$
- Feature space: $\mathcal{H} = \mathbf{R}^D$, where $D = d + \binom{d}{2} \approx d^2/2$.
- Feature map:

$$\phi(x) = (x_1, \ldots, x_d, x_1^2, \ldots, x_d^2, \sqrt{2}x_1x_2, \ldots, \sqrt{2}x_ix_j, \ldots \sqrt{2}x_{d-1}x_d)^T$$

- Still have

$$\begin{aligned} k(w,x) &= \langle \phi(w), \phi(x) \rangle \\ &= \langle x, y \rangle + \langle x, y \rangle^2 \end{aligned}$$

- Computation for inner product with explicit mapping: $O(d^2)$
- Computation for implicit kernel calculation: $O(d)$.

Based on Guillaume Obozinski's Statistical Machine Learning course at Louvain, Feb 2014.

Polynomial Kernel in $\mathbf{R}^d$

- Input space $\mathcal{X} = \mathbf{R}^d$
- Kernel function:
$$k(w,x) = (1 + \langle w,x \rangle)^M$$
- Corresponds to a feature map with all terms up to degree M.
- For any M, computing the kernel has same computational cost
- Cost of explicit inner product computation grows rapidly in M.

Radial Basis Function (RBF) / Gaussian Kernel

- Input space $\mathcal{X} = \mathbf{R}^d$

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

  where σ^2 is known as the bandwidth parameter.
- Does it act like a similarity score?
- Why "radial"?
- Have we departed from our "inner product of feature vector" recipe?
  - Yes and no: corresponds to an infinite dimensional feature vector
- Probably the most common nonlinear kernel.

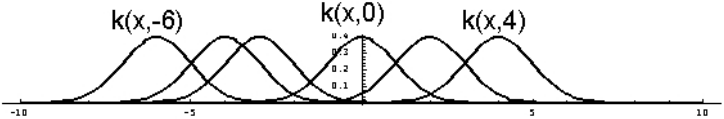
Prediction Functions with RBF Kernel

RBF Basis

- Input space $\mathcal{X} = \mathbf{R}$
- Output space: $\mathcal{Y} = \mathbf{R}$
- RBF kernel $k(w,x) = \exp\left(-(w-x)^2\right)$.
- Suppose we have 6 training examples: $x_i \in \{-6,-4,-3,0,2,4\}$.
- If representer theorem applies, then

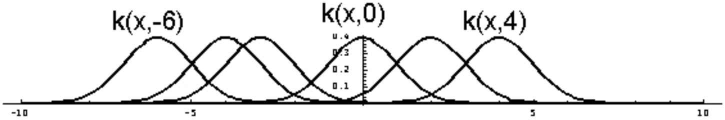
$$f(x) = \sum_{i=1}^{6} \alpha_i k(x_i, x).$$

- f is a linear combination of 6 basis functions of form $k(x_i, \cdot)$:



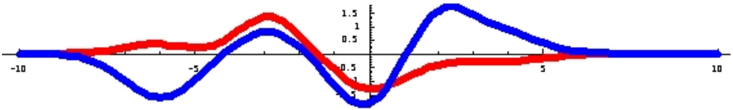
RBF Predictions

- Basis functions



  - Predictions of the form

$$f(x) = \sum_{i=1}^{6} \alpha_i k(x_i, x)$$



- If we have a kernelized algorithm with RBF kernel, prediction functions $x \mapsto \langle w, \psi(x) \rangle$ will look this way.
  - whether we got w from SVM, ridge regression, etc...

When is $k(x, w)$ a kernel function? (Mercer's Theorem)

How to Get Kernels?

1. Explicitly construct $\psi(x) : \mathcal{X} \to \mathbf{R}^d$ and define $k(x, w) = \psi(x)^T \psi(w)$.
2. Directly define the kernel function $k(x, w)$, and verify it corresponds to $\langle \psi(x), \psi(w) \rangle$ for some ψ.

There are many theorems to help us with the second approach

Positive Semidefinite Matrices

Definition

A real, symmetric matrix $M \in \mathbf{R}^{n \times n}$ is **positive semidefinite (psd)** if for any $x \in \mathbf{R}^n$,

$$x^T M x \geqslant 0.$$

Theorem

The following conditions are each necessary and sufficient for M to be positive semidefinite:

- *M has a "square root", i.e. there exists R s.t. $M = R^T R$.*
- *All eigenvalues of M are greater than or equal to 0.*

Positive Semidefinite Function

Definition

A symmetric kernel function $k : \mathcal{X} \times \mathcal{X} \to \mathbf{R}$ is **positive semidefinite (psd)** if for any finite set $\{x_1, \ldots, x_n\} \in \mathcal{X}$, the kernel matrix on this set

$$K = \left(k(x_i, x_j)\right)_{i,j} = \begin{pmatrix} k(x_1, x_1) & \cdots & k(x_1, x_n) \\ \vdots & \ddots & \cdots \\ k(x_n, x_1) & \cdots & k(x_n, x_n) \end{pmatrix}$$

is a positive semidefinite matrix.

Mercer's Theorem

Theorem

A symmetric function $k(w,x)$ can be expressed as an inner product

$$k(w,x) = \langle \psi(w), \psi(x) \rangle$$

*for some ψ if and only if $k(w,x)$ is **positive semidefinite.***

Proof.

[Sketch] Suppose $k(w,x)$ is psd.

1. Let

□

Generating New Kernels from Old

Suppose $k, k_1, k_2 : \mathcal{X} \times \mathcal{X} \to \mathbf{R}$ are psd kernels. Then so are the following:

$$
\begin{aligned}
k_{\text{new}}(w,x) &= k_1(w,x) + k_2(w,x) \\
k_{\text{new}}(w,x) &= \alpha k(w,x) \\
k_{\text{new}}(w,x) &= f(w)f(x) \text{ for any function } f(x) \\
k_{\text{new}}(w,x) &= k_1(w,x)k_2(w,x)
\end{aligned}
$$

are also A symmetric function $k(w,x)$ can be expressed as an inner product

$$k(w,x) = \langle \phi(w), \phi(x) \rangle$$

for some ϕ if and only if $k(w,x)$ is **positive semidefinite.**

- If we start with a psd kernel, can we generate more?