Kernelizations

David Rosenberg

New York University

February 21, 2015

Linear SVM

- The SVM prediction function is the solution to

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

- Found it's equivalent to solve the dual problem to get α^*:i

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

- Notice: x's only show up as inner products with other x's.

Kernelization

Definition

We say a machine learning method is **kernelized** if all references to inputs $x \in \mathcal{X}$ are through an inner product between pairs of points $\langle x, y \rangle$ for $x, y \in \mathbf{R}^d$.

So far, we've only partially kernelized SVM

We've shown that the training portion is kernelized. Later we'll show the prediction portion is also kernelized.

SVM Dual Problem

- x's only show up in pairs of inner products: $x_j^T x_i = \langle x_j, x_i \rangle$:

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

- Then primal optimal solution is given as:

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

and for any $\alpha_i \in \left(0, \frac{c}{n}\right)$,

$$b^* = y_i - x_i^T w^*.$$

SVM: Kernelizing b

- We found that for any j with $\alpha_j \in \left(0, \frac{c}{n}\right)$:

$$\begin{aligned} b^* &= y_j - x_j^T w^* \\ &= y_j - x_j^T \left(\sum_{i=1}^{n} \alpha_i^* y_i x_i \right). \\ &= y_j - \sum_{i=1}^{n} \alpha_i^* y_i \langle x_j, x_i \rangle . \end{aligned}$$

- What about kernelizing w^*?

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

- Not obvious...
- But we really only care about kernelizing the predictions $f^*(x)$.

SVM: Kernelizing Predictions $f^*(x)$

- For any j with $\alpha_j \in \left(0, \frac{c}{n}\right)$:

$$
\begin{aligned}
f^*(x) &= x^T w^* + b^* \\
&= x^T \left(\sum_{i=1}^{n} \alpha_i^* y_i x_i \right) + b^* \\
&= \sum_{i=1}^{n} \alpha_i^* y_i \langle x_i, x \rangle + \left(y_j - \sum_{i=1}^{n} \alpha_i^* y_i \langle x_j, x_i \rangle \right)
\end{aligned}
$$

- We now have a fully kernelized version of SVM.
- Can we kernelize the primal version of the SVM?

Kernelizing the SVM Primal Problem

- Primal SVM

$$\min_{w\in\mathbf{R}^d, b\in\mathbf{R}} \frac{1}{2}\|w\|^2 + \frac{c}{n}\sum_{i=1}^{n}\left(1-y_i\left[w^Tx_i+b\right]\right)_+.$$

- From our study of the dual, found that

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

- So w^* is a linear combination of the input vectors.
- Restrict to optimization to w of the form

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

Some Vectorization

- **Design matrix** $X \in \mathbf{R}^{n \times d}$ has input vectors as rows:

$$X = \begin{pmatrix} -x_1- \\ \vdots \\ -x_n- \end{pmatrix}.$$

- The contraint on w looks like

$$w = \begin{pmatrix} w_1 \\ \vdots \\ w_d \end{pmatrix} = \begin{pmatrix} | & \cdots & | \\ x_1 & \cdots & x_n \\ | & \cdots & | \end{pmatrix} \begin{pmatrix} \beta_1 \\ \vdots \\ \beta_n \end{pmatrix} = X^T \beta.$$

- So replace all w with $X^T\beta$, with $\beta \in \mathbf{R}^n$ unrestricted.

The Kernel Matrix (or the Gram Matrix)

Definition

For a set of $\{x_1, \ldots, x_n\}$ and an inner product $\langle \cdot, \cdot \rangle$on the set, the **kernel matrix** or the **Gram matrix** is defined as

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

Then for the standard Euclidean inner product $\langle x_i, x_j \rangle = x_i^T x_j$, we have

$$K = XX^T$$

Some Vectorization

- Regularization Term:

$$\|w\|^2 = w^T w = \beta^T X X^T \beta = \beta^T K \beta$$

- Prediction on training point x_i:

$$\begin{aligned} f(x_i) &= b + x_i^T w \\ &= b + x_i^T \left(\sum_{j=1}^{n} \beta_j x_j \right) \\ &= b + \sum_{j=1}^{n} \beta_j K_{ij} \end{aligned}$$

Kernelized Primal SVM

- Putting it together, kernelized primal SVM is

$$\min_{\beta\in\mathbf{R}^n, b\in\mathbf{R}} \frac{1}{2}\beta^T K\beta + \frac{c}{n}\sum_{i=1}^{n}\left(1 - y_i\left[b + \sum_{j=1}^{n}\beta_j K_{ij}\right]\right)_+ .$$

- We can write this as a differentiable, constrained optimization problem:

$$\begin{array}{ll} \text{minimize} & \frac{1}{2}\beta^T K\beta + \frac{c}{n}\mathbf{1}^T\xi \\ \text{subject to} & \xi \succeq 0 \\ & \xi \succeq \left(\mathbf{1} - Y\left[b + K\beta\right]\right), \end{array}$$

where $Y = \operatorname{diag}(y_1, \ldots, y_n)$, $\mathbf{1}$ is a column vector of 1's, and $\succeq$ represent element-wise vector inequality.

Kernelized Primal SVM: Kernel Trick

- Kernelized primal SVM is

$$\min_{\beta\in\mathbf{R}^n, b\in\mathbf{R}} \frac{1}{2}\beta^T K\beta + \frac{c}{n}\sum_{i=1}^{n}\left(1-y_i\left[b+\sum_{j=1}^{n}\beta_j K_{ij}\right]\right)_+ .$$

- We derived this with $K = XX^T$, which corresponds to the linear kernel.
- Suppose we have another kernel defined in terms of a map ϕ, i.e.

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

  then we can just plug in the corresponding kernel matrix K_ϕ to the optimization problem above.
- What kernels can be written as an inner product of feature vectors?

Ridge Regression

- Recall the ridge regression objective:

$$J(w) = \|Xw - y\|^2 + \lambda \|w\|^2.$$

- Differentiating and setting equal to zero ,we get

$$\left(X^T X + \lambda I\right) w \quad = \quad X^T y$$

- On board to review?

Kernelizing Ridge Regression

- So we have, for $\lambda > 0$:

$$
\begin{aligned}
(X^TX + \lambda I)w &= X^Ty \\
\lambda w &= X^Ty - X^TXw \\
w &= \frac{1}{\lambda}X^T(y - Xw) \\
w &= X^T\alpha
\end{aligned}
$$

  for $\alpha = \lambda^{-1}(y - Xw) \in \mathbf{R}^n$.

- So w is "**in the span of the data**":

$$
w = \begin{pmatrix} | & \cdots & | \\ x_1 & \cdots & x_n \\ | & \cdots & | \end{pmatrix} \begin{pmatrix} \alpha_1 \\ \vdots \\ \alpha_n \end{pmatrix} = \alpha_1 x_1 + \cdots \alpha_n x_n
$$

Kernelizing Ridge Regression

- So plugging in $w = X^T\alpha$ to

$$\begin{aligned}
\alpha &= \lambda^{-1}(y - Xw) \\
\lambda\alpha &= y - XX^T\alpha \\
XX^T\alpha + \lambda\alpha &= y \\
\left(XX^T + \lambda I\right)\alpha &= y \\
\alpha &= (\lambda I + XX^T)^{-1}y
\end{aligned}$$

- So we have α. How to do prediction?

$$\begin{aligned}
Xw &= X\left(X^T\alpha\right) \\
&= \left(XX^T\right)(\lambda I + XX^T)^{-1}y
\end{aligned}$$

- To predict on new data, need the "cross-kernel" matrix, between new and old data.

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 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?

Additive Closure

- Suppose k_1 and k_2 are psd kernels with feature maps ϕ_1 and ϕ_2 ,respectively.
- Then
$$k_1(w,x)+k_2(w,x)$$
is a psd kernel.
- Proof: Concatenate the feature vectors to get
$$\phi(x)=(\phi_1(x),\phi_2(x)).$$
Then ϕ is a feature map for k_1+k_2.

Closure under Positive Scaling

- Suppose k is a psd kernel with feature maps ϕ.
- Then for any $\alpha > 0$,
$$\alpha k$$
is a psd kernel.
- Proof: Note that
$$\phi(x) = \sqrt{\alpha}\phi(x)$$
is a feature map for αk.

Scalar Function Gives a Kernel

- For any function $f(x)$,

  $$k(w,x) = f(w)f(x)$$

  is a kernel.

- Proof: Let $f(x)$ be the feature mapping. (It maps into a 1-dimensional feature space.)

  $$\langle f(x), f(w) \rangle = f(x)f(w) = k(w,x).$$

Closure under Hadamard Products

- Suppose k_1 and k_2 are psd kernels with feature maps ϕ_1 and ϕ_2, respectively.
- Then

$$k_1(w,x)k_2(w,x)$$

  is a psd kernel.
- Proof: Take the outer product of the feature vectors:

$$\phi(x) = \phi_1(x)\left[\phi_2(x)\right]^T.$$

  Note that $\phi(x)$ is a matrix.
- Continued...

Closure under Hadamard Products

- Then

$$
\begin{aligned}
\langle \phi(x), \phi(w) \rangle &= \sum_{i,j} \phi(x)\phi(w) \\
&= \sum_{i,j} \left[\phi_1(x) \left[\phi_2(x)\right]^T \right]_{ij} \left[\phi_1(w) \left[\phi_2(w)\right]^T \right]_{ij} \\
&= \sum_{i,j} \left[\phi_1(x)\right]_i \left[\phi_2(x)\right]_j \left[\phi_1(w)\right]_i \left[\phi_2(w)\right]_j \\
&= \left(\sum_i \left[\phi_1(x)\right]_i \left[\phi_1(w)\right]_i \right) \left(\sum_j \left[\phi_2(x)\right]_j \left[\phi_2(w)\right]_j \right) \\
&= k_1(w,x) k_2(w,x)
\end{aligned}
$$