

## HW3

1. (a)  $k(\vec{x}, \vec{z}) = k_1(\vec{x}, \vec{z}) + k_2(\vec{x}, \vec{z})$

Must be a kernel as sum of two positive semidefinite values is positive semidefinite; thus

$k(\vec{x}, \vec{z})$  is a Mercer kernel.

(b)  $k(\vec{x}, \vec{z}) = k_1(\vec{x}, \vec{z}) - k_2(\vec{x}, \vec{z})$

Not a kernel. Consider

$$k_2(\vec{x}, \vec{z}) = 2k_1(\vec{x}, \vec{z}), \text{ thus}$$

$$k(\vec{x}, \vec{z}) = k_1(\vec{x}, \vec{z}) - 2k_1(\vec{x}, \vec{z}) = -k_1(\vec{x}, \vec{z}), \text{ which is}$$

not positive semidefinite, thus

$k(\vec{x}, \vec{z})$  is not a kernel.

(c) We know  $a > 0, a \in \mathbb{R}$

This is a kernel, as

$$k(\vec{x}, \vec{z}) = a k_1(\vec{x}, \vec{z}) > 0, \text{ thus positive semidefinite.}$$

$\Rightarrow$  Mercer kernel.

~~(d)  $k(\vec{x}, \vec{z}) = -a k_1(\vec{x}, \vec{z})$ ,~~

~~Because  $a > 0, a \in \mathbb{R}$ , we know  $-a k_1(\vec{x}, \vec{z}) < 0$ ,~~

~~thus  $k(\vec{x}, \vec{z})$  is negative semidefinite and thus not a Mercer kernel.~~

(d)  $k(\vec{x}, \vec{z}) = -a k_1(\vec{x}, \vec{z})$

By defn, we know

$$\mathbf{z}^T k_1(\vec{x}, \vec{z}) \mathbf{z} \geq 0, \text{ since } a > 0, a \in \mathbb{R},$$

$$\Rightarrow -a \mathbf{z}^T k_1(\vec{x}, \vec{z}) \mathbf{z} < 0, \text{ so this is not positive}$$

Semidefinite

$\Rightarrow$  Not a Mercer kernel.

$$\begin{aligned}
 (e) \quad k_1(\vec{x}, \vec{z}) &= \sum_i \phi_i^{(1)}(x) \phi_i^{(1)}(z) \\
 k_2(\vec{x}, \vec{z}) &= \sum_j \phi_j^{(2)}(x) \phi_j^{(2)}(z) \\
 k(\vec{x}, \vec{z}) &= k_1(\vec{x}, \vec{z}) k_2(\vec{x}, \vec{z}) = \sum_i \phi_i^{(1)}(x) \phi_i^{(1)}(z) \sum_j \phi_j^{(2)}(x) \phi_j^{(2)}(z) \\
 &\Rightarrow \sum_i \sum_j (\phi_i^{(1)}(x) \phi_j^{(2)}(x)) (\phi_i^{(1)}(z) \phi_j^{(2)}(z))
 \end{aligned}$$

$\Rightarrow$  Sub in  $\phi$  for kernels, thus  $k(\vec{x}, \vec{z})$  are kernels

$$(f) \quad k(\vec{x}, \vec{z}) = f(\vec{x}) f(\vec{z})$$

kernel. ⊗

We want to write  $k(\vec{x}, \vec{z})$  as  $\phi(x)^T \phi(z) \Rightarrow$  since  $f(\vec{x})$  and  $f(\vec{z})$  are scalars, we fit this property if we set  $\phi = f$ .

$$(g) \quad k(\vec{x}, \vec{z}) = k_3(\phi(\vec{x}), \phi(\vec{z}))$$

Because  $k_3$  is a kernel over  $\mathbb{R}^M \times \mathbb{R}^M$ , and  $\phi(\vec{x}) + \phi(\vec{z})$  map  $\Rightarrow \mathbb{R}^M$ ,

$k(\vec{x}, \vec{z})$  must be a kernel.

(h) Since  $p: \mathbb{R} \rightarrow \mathbb{R}$  is a polynomial with positive coefficients, We know the sum returns a kernel (a), scalar multiplication by positive number returns a kernel (c), product returns a kernel (e), and real valued number returns a kernel (f), which the superposition of which returns a polynomial. Thus,  $k(\vec{x}, \vec{z}) = p k_1(\vec{x}, \vec{z})$  is a kernel.

2. (a) To solve this, write  $w^{(n+1)}$  [final iteration of  $w$ ] to observe pattern.

$$w^{n+1} = w^n + \alpha [t^{n+1} - y(x^{n+1}; w^n)] \phi(x^{n+1})$$

expand  $w^n$

$$\Rightarrow w^{n-1} + \alpha [t^n - y(x^n; w^{n-1})] \phi(x^n) + \dots$$

Continue expanding until you hit  $w_0$ , thus, we see

$$w^{n+1} = \sum_{i=1}^n \alpha_i \phi(x^{(i)}), \quad \alpha_i = \alpha [t^i - y(x^i; w^{i-1})]$$

Thus, we write the  $w^{(i)}$  as a linear combinations of  $\phi(x^{(i)})$ s.

$$(b) f(w^{(i)T} \phi(x^{(i+1)}))$$

$$= f\left(\sum_{j=1}^N \alpha_j^T \phi(x^{(j)})^T \phi(x^{(i+1)})\right)$$

Use the kernel trick

$$\Rightarrow f\left(\sum_{j=1}^N \alpha_j^T k(x^{(j)}, x^{(i+1)})\right), \text{ we can use this}$$

kernel trick to efficiently compute the prediction on input  $x^{(i+1)}$ .

$$(c) w^{(i+1)} = w^{(i)} + \alpha [t^{(i+1)} - y(\phi(x^{(i+1)})^T w^{(i)})] \phi(x^{(i+1)})$$

$$= w^{(i)} + \alpha [t^{(i+1)} - f(\sum_{j=1}^N \alpha_j^T k(x^{(j)}, x^{(i+1)}))] \phi(x^{(i+1)})$$

☒

$$3. (a) E(\vec{w}, b) = \frac{1}{2} \|\vec{w}\|^2 + C \sum_{i=1}^N \max(0, 1 - t^{(i)}(w^T x^{(i)} + b))$$

$$= \frac{1}{2} \|\vec{w}\|^2 + C \sum_{i=1}^N \mathbb{I}[t^{(i)}(w^T x^{(i)} + b) < 1] (1 - t^{(i)}(w^T x^{(i)} + b))$$

Solve for two derivatives

$$\nabla_{\vec{w}} E(\vec{w}, b) = \frac{\partial}{\partial \vec{w}} \left( \frac{1}{2} \|\vec{w}\|^2 + C \sum_{i=1}^N \mathbb{I}[t^{(i)}(w^T x^{(i)} + b) < 1] (1 - t^{(i)}(w^T x^{(i)} + b)) \right)$$

$$= \vec{w} + C \sum_{i=1}^N \mathbb{I}[t^{(i)}(w^T x^{(i)} + b) < 1] (-t^{(i)} x^{(i)})$$

$$= \vec{w} - C \sum_{i=1}^N \mathbb{I}[t^{(i)}(w^T x^{(i)} + b) < 1] t^{(i)} x^{(i)} \quad \checkmark$$

$$\frac{\partial}{\partial b} E(\vec{w}, b) = C \sum_{i=1}^N \mathbb{I}[t^{(i)}(w^T x^{(i)} + b) < 1] [-t^{(i)}]$$

$$= -C \sum_{i=1}^N \mathbb{I}[t^{(i)}(w^T x^{(i)} + b) < 1] t^{(i)} \quad \checkmark$$

Thus, proven.  $\square$

(b) Given in code

$$(c) \quad E^{(i)}(\vec{w}, b) = \frac{1}{2N} \|\vec{w}\|^2 + C \max(0, 1 - t^{(i)}(w^T x^{(i)} + b))$$

$$= \frac{1}{2N} \|\vec{w}\|^2 + C I[t^{(i)}(w^T x^{(i)} + b) < 1] (1 - t^{(i)}(w^T x^{(i)} + b))$$

$$\nabla_w E^{(i)}(\vec{w}, b) = \frac{\|\vec{w}\|}{N} + \nabla_w C I[t^{(i)}(w^T x^{(i)} + b) < 1] (1 - t^{(i)}(w^T x^{(i)} + b))$$

$$= \frac{\|\vec{w}\|}{N} - C I[t^{(i)}(w^T x^{(i)} + b) < 1] (t^{(i)} x^{(i)})$$

$$\frac{\partial}{\partial b} E^{(i)}(\vec{w}, b)$$

$$= -C I[t^{(i)}(w^T x^{(i)} + b) < 1] t^{(i)}$$

Thus, proven.  $\square$

(d) Given in code

(e) We conclude SGD converges much faster than BGD. Observe that at 5 iterations, BGD was at ~55% accuracy, but at the same number, SGD was at 96% accuracy.

4 (a) Given in code.

4 (b) Analysis of the two models yields SVM to be greater than NB in terms of accuracy  $\rightarrow$  though the trends of the two do follow a similar path. With more training examples, error decreases.