# Kernel Methods

## CSci 5525: Machine Learning

Instructor: Arindam Banerjee

October 22, 2008

# Non-linear SVMs

- All important equations have dot-products

# Non-linear SVMs

- All important equations have dot-products
  - Dual is expressed in terms of $\mathbf{x}_i^T \mathbf{x}_j$

# Non-linear SVMs

- All important equations have dot-products
  - Dual is expressed in terms of $\mathbf{x}_i^T \mathbf{x}_j$
  - The predictions are in terms of $\mathbf{x}_i^T \mathbf{x}$

# Non-linear SVMs

- All important equations have dot-products
  - Dual is expressed in terms of $\mathbf{x}_i^T \mathbf{x}_j$
  - The predictions are in terms of $\mathbf{x}_i^T \mathbf{x}$
- How to get a non-linear classifier:

# Non-linear SVMs

- All important equations have dot-products
  - Dual is expressed in terms of $\mathbf{x}_i^T \mathbf{x}_j$
  - The predictions are in terms of $\mathbf{x}_i^T \mathbf{x}$
- How to get a non-linear classifier:
  - Map $\mathbf{x}$ to some (higher dimensional) space $\Phi : \mathbb{R}^d \mapsto \mathcal{H}$

# Non-linear SVMs

- All important equations have dot-products
  - Dual is expressed in terms of $\mathbf{x}_i^T \mathbf{x}_j$
  - The predictions are in terms of $\mathbf{x}_i^T \mathbf{x}$
- How to get a non-linear classifier:
  - Map $\mathbf{x}$ to some (higher dimensional) space $\Phi : \mathbb{R}^d \mapsto \mathcal{H}$
  - The derived feature vectors are $\Phi(\mathbf{x}_i), \forall i$

# Non-linear SVMs

- All important equations have dot-products
  - Dual is expressed in terms of $\mathbf{x}_i^T \mathbf{x}_j$
  - The predictions are in terms of $\mathbf{x}_i^T \mathbf{x}$
- How to get a non-linear classifier:
  - Map $\mathbf{x}$ to some (higher dimensional) space $\Phi : \mathbb{R}^d \mapsto \mathcal{H}$
  - The derived feature vectors are $\Phi(\mathbf{x}_i), \forall i$
  - The dot products are $\langle \Phi(\mathbf{x}_i), \Phi(\mathbf{x}_j) \rangle$

# Non-linear SVMs

- All important equations have dot-products
  - Dual is expressed in terms of $\mathbf{x}_i^T \mathbf{x}_j$
  - The predictions are in terms of $\mathbf{x}_i^T \mathbf{x}$
- How to get a non-linear classifier:
  - Map $\mathbf{x}$ to some (higher dimensional) space $\Phi : \mathbb{R}^d \mapsto \mathcal{H}$
  - The derived feature vectors are $\Phi(\mathbf{x}_i), \forall i$
  - The dot products are $\langle \Phi(\mathbf{x}_i), \Phi(\mathbf{x}_j) \rangle$
- Kernel function allows implicit calculation of dot-products

$$k(\mathbf{x}_i, \mathbf{x}_j) = \langle \Phi(\mathbf{x}_i), \Phi(\mathbf{x}_j) \rangle$$

# Non-linear SVMs

- All important equations have dot-products
  - Dual is expressed in terms of $\mathbf{x}_i^T \mathbf{x}_j$
  - The predictions are in terms of $\mathbf{x}_i^T \mathbf{x}$
- How to get a non-linear classifier:
  - Map $\mathbf{x}$ to some (higher dimensional) space $\Phi : \mathbb{R}^d \mapsto \mathcal{H}$
  - The derived feature vectors are $\Phi(\mathbf{x}_i), \forall i$
  - The dot products are $\langle \Phi(\mathbf{x}_i), \Phi(\mathbf{x}_j) \rangle$
- Kernel function allows implicit calculation of dot-products

$$k(\mathbf{x}_i, \mathbf{x}_j) = \langle \Phi(\mathbf{x}_i), \Phi(\mathbf{x}_j) \rangle$$

- Learn a linear max margin separator in $\mathcal{H}$

# Non-linear SVMs

- All important equations have dot-products
  - Dual is expressed in terms of $\mathbf{x}_i^T \mathbf{x}_j$
  - The predictions are in terms of $\mathbf{x}_i^T \mathbf{x}$
- How to get a non-linear classifier:
  - Map $\mathbf{x}$ to some (higher dimensional) space $\Phi : \mathbb{R}^d \mapsto \mathcal{H}$
  - The derived feature vectors are $\Phi(\mathbf{x}_i), \forall i$
  - The dot products are $\langle \Phi(\mathbf{x}_i), \Phi(\mathbf{x}_j) \rangle$
- Kernel function allows implicit calculation of dot-products

$$k(\mathbf{x}_i, \mathbf{x}_j) = \langle \Phi(\mathbf{x}_i), \Phi(\mathbf{x}_j) \rangle$$

- Learn a linear max margin separator in $\mathcal{H}$
- The final prediction function

$$f(\mathbf{x}) = \sum_{i:\alpha_i > 0} \alpha_i y_i \langle \Phi(\mathbf{x}_i), \Phi(\mathbf{x}) \rangle + b = \sum_{i:\alpha_i > 0} \alpha_i y_i k(\mathbf{x}_i, \mathbf{x}) + b$$

# The Kernel Trick

- Reduces non-linear SVM learning to linear SVM learning

# The Kernel Trick

- Reduces non-linear SVM learning to linear SVM learning
- What functions $k(\mathbf{x}_i, \mathbf{x}_j)$ are valid kernels?

# The Kernel Trick

- Reduces non-linear SVM learning to linear SVM learning
- What functions $k(\mathbf{x}_i, \mathbf{x}_j)$ are valid kernels?
  - Iff $\forall g(\mathbf{x})$ such that $\int g(\mathbf{x})^2 d\mathbf{x} < \infty$,

$$\int k(\mathbf{x}_1, \mathbf{x}_2) g(\mathbf{x}_1) g(\mathbf{x}_2) d\mathbf{x}_1 d\mathbf{x}_2 \geq 0$$

# The Kernel Trick

- Reduces non-linear SVM learning to linear SVM learning
- What functions $k(\mathbf{x}_i, \mathbf{x}_j)$ are valid kernels?
  - Iff $\forall g(\mathbf{x})$ such that $\int g(\mathbf{x})^2 d\mathbf{x} < \infty$,

$$\int k(\mathbf{x}_1, \mathbf{x}_2) g(\mathbf{x}_1) g(\mathbf{x}_2) d\mathbf{x}_1 d\mathbf{x}_2 \geq 0$$

  - The condition is known as Mercer's condition

# The Kernel Trick

- Reduces non-linear SVM learning to linear SVM learning
- What functions $k(\mathbf{x}_i, \mathbf{x}_j)$ are valid kernels?
  - Iff $\forall g(\mathbf{x})$ such that $\int g(\mathbf{x})^2 d\mathbf{x} < \infty$,

$$\int k(\mathbf{x}_1, \mathbf{x}_2) g(\mathbf{x}_1) g(\mathbf{x}_2) d\mathbf{x}_1 d\mathbf{x}_2 \geq 0$$

  - The condition is known as Mercer's condition
- Examples:

# The Kernel Trick

- Reduces non-linear SVM learning to linear SVM learning
- What functions $k(\mathbf{x}_i, \mathbf{x}_j)$ are valid kernels?
  - Iff $\forall g(\mathbf{x})$ such that $\int g(\mathbf{x})^2 d\mathbf{x} < \infty$,

$$\int k(\mathbf{x}_1, \mathbf{x}_2) g(\mathbf{x}_1) g(\mathbf{x}_2) d\mathbf{x}_1 d\mathbf{x}_2 \geq 0$$

  - The condition is known as Mercer's condition
- Examples:
  - Polynomial Kernel: $k(\mathbf{x}_i, \mathbf{x}_j) = (\mathbf{x}_i^T \mathbf{x}_j + 1)^p$

# The Kernel Trick

- Reduces non-linear SVM learning to linear SVM learning
- What functions $k(\mathbf{x}_i, \mathbf{x}_j)$ are valid kernels?
  - Iff $\forall g(\mathbf{x})$ such that $\int g(\mathbf{x})^2 d\mathbf{x} < \infty$,

$$\int k(\mathbf{x}_1, \mathbf{x}_2) g(\mathbf{x}_1) g(\mathbf{x}_2) d\mathbf{x}_1 d\mathbf{x}_2 \geq 0$$

  - The condition is known as Mercer's condition
- Examples:
  - Polynomial Kernel: $k(\mathbf{x}_i, \mathbf{x}_j) = (\mathbf{x}_i^T \mathbf{x}_j + 1)^p$
  - RBF Kernel: $k(\mathbf{x}_i, \mathbf{x}_j) = \exp(-\|\mathbf{x}_i - \mathbf{x}_j\|^2)$

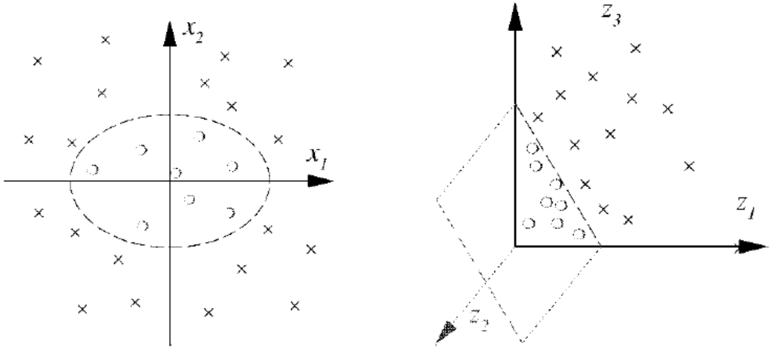
# The Kernel Trick

- Reduces non-linear SVM learning to linear SVM learning
- What functions $k(\mathbf{x}_i, \mathbf{x}_j)$ are valid kernels?
  - Iff $\forall g(\mathbf{x})$ such that $\int g(\mathbf{x})^2 d\mathbf{x} < \infty$,

$$\int k(\mathbf{x}_1, \mathbf{x}_2) g(\mathbf{x}_1) g(\mathbf{x}_2) d\mathbf{x}_1 d\mathbf{x}_2 \geq 0$$

  - The condition is known as Mercer's condition
- Examples:
  - Polynomial Kernel: $k(\mathbf{x}_i, \mathbf{x}_j) = (\mathbf{x}_i^T \mathbf{x}_j + 1)^p$
  - RBF Kernel: $k(\mathbf{x}_i, \mathbf{x}_j) = \exp(-\|\mathbf{x}_i - \mathbf{x}_j\|^2)$
- How to choose a kernel for a given application?

# Example




$$z_1 = x_1^2, \quad z_2 = \sqrt{2}x_1x_2, \quad z_3 = x_2^2$$

# Kernels and Inner Product Spaces

- A kernel $k : \mathcal{X} \times \mathcal{X} \mapsto \mathbb{R}$ is positive semi-definite if

# Kernels and Inner Product Spaces

- A kernel $k : \mathcal{X} \times \mathcal{X} \mapsto \mathbb{R}$ is positive semi-definite if
  - $k$ is symmetric, i.e., $k(\mathbf{x}_i, \mathbf{x}_j) = k(\mathbf{x}_j, \mathbf{x}_i)$

# Kernels and Inner Product Spaces

- A kernel $k : \mathcal{X} \times \mathcal{X} \mapsto \mathbb{R}$ is positive semi-definite if
  - $k$ is symmetric, i.e., $k(\mathbf{x}_i, \mathbf{x}_j) = k(\mathbf{x}_j, \mathbf{x}_i)$
  - For $\{\mathbf{x}_1, \ldots, \mathbf{x}_n\}$, matrix $K(\mathbf{x}_i, \mathbf{x}_j)$ is positive semi-definite

# Kernels and Inner Product Spaces

- A kernel $k : \mathcal{X} \times \mathcal{X} \mapsto \mathbb{R}$ is positive semi-definite if
  - $k$ is symmetric, i.e., $k(\mathbf{x}_i, \mathbf{x}_j) = k(\mathbf{x}_j, \mathbf{x}_i)$
  - For $\{\mathbf{x}_1, \ldots, \mathbf{x}_n\}$, matrix $K(\mathbf{x}_i, \mathbf{x}_j)$ is positive semi-definite
- Inner product space for kernel $k$

# Kernels and Inner Product Spaces

- A kernel $k : \mathcal{X} \times \mathcal{X} \mapsto \mathbb{R}$ is positive semi-definite if
  - $k$ is symmetric, i.e., $k(\mathbf{x}_i, \mathbf{x}_j) = k(\mathbf{x}_j, \mathbf{x}_i)$
  - For $\{\mathbf{x}_1, \ldots, \mathbf{x}_n\}$, matrix $K(\mathbf{x}_i, \mathbf{x}_j)$ is positive semi-definite
- Inner product space for kernel $k$
  - The kernel feature map $\Phi(\mathbf{x}) = k(\cdot, \mathbf{x})$

# Kernels and Inner Product Spaces

- A kernel $k : \mathcal{X} \times \mathcal{X} \mapsto \mathbb{R}$ is positive semi-definite if
  - $k$ is symmetric, i.e., $k(\mathbf{x}_i, \mathbf{x}_j) = k(\mathbf{x}_j, \mathbf{x}_i)$
  - For $\{\mathbf{x}_1, \ldots, \mathbf{x}_n\}$, matrix $K(\mathbf{x}_i, \mathbf{x}_j)$ is positive semi-definite
- Inner product space for kernel $k$
  - The kernel feature map $\Phi(\mathbf{x}) = k(\cdot, \mathbf{x})$
    - Each point $x$ is now a function is some space

# Kernels and Inner Product Spaces

- A kernel $k : \mathcal{X} \times \mathcal{X} \mapsto \mathbb{R}$ is positive semi-definite if
  - $k$ is symmetric, i.e., $k(\mathbf{x}_i, \mathbf{x}_j) = k(\mathbf{x}_j, \mathbf{x}_i)$
  - For $\{\mathbf{x}_1, \ldots, \mathbf{x}_n\}$, matrix $K(\mathbf{x}_i, \mathbf{x}_j)$ is positive semi-definite
- Inner product space for kernel $k$
  - The kernel feature map $\Phi(\mathbf{x}) = k(\cdot, \mathbf{x})$
    - Each point $x$ is now a function is some space
    - The function measures its similarity will all other points

## Kernels and Inner Product Spaces

- A kernel $k : \mathcal{X} \times \mathcal{X} \mapsto \mathbb{R}$ is positive semi-definite if
  - $k$ is symmetric, i.e., $k(\mathbf{x}_i, \mathbf{x}_j) = k(\mathbf{x}_j, \mathbf{x}_i)$
  - For $\{\mathbf{x}_1, \ldots, \mathbf{x}_n\}$, matrix $K(\mathbf{x}_i, \mathbf{x}_j)$ is positive semi-definite
- Inner product space for kernel $k$
  - The kernel feature map $\Phi(\mathbf{x}) = k(\cdot, \mathbf{x})$
    - Each point $x$ is now a function is some space
    - The function measures its similarity will all other points
  - The vector space corresponding to the mapping

$$\text{span}\{\Phi(\mathbf{x}) : \mathbf{x} \in \mathcal{X}\} = \{f(\cdot) = \sum_i \alpha_i k(\cdot, \mathbf{x}_i), \alpha_i \in \mathbb{R}\}$$

# Kernels and Inner Product Spaces

- A kernel $k : \mathcal{X} \times \mathcal{X} \mapsto \mathbb{R}$ is positive semi-definite if
  - $k$ is symmetric, i.e., $k(\mathbf{x}_i, \mathbf{x}_j) = k(\mathbf{x}_j, \mathbf{x}_i)$
  - For $\{\mathbf{x}_1, \ldots, \mathbf{x}_n\}$, matrix $K(\mathbf{x}_i, \mathbf{x}_j)$ is positive semi-definite
- Inner product space for kernel $k$
  - The kernel feature map $\Phi(\mathbf{x}) = k(\cdot, \mathbf{x})$
    - Each point $x$ is now a function is some space
    - The function measures its similarity will all other points
  - The vector space corresponding to the mapping

$$\text{span}\{\Phi(\mathbf{x}) : \mathbf{x} \in \mathcal{X}\} = \{f(\cdot) = \sum_i \alpha_i k(\cdot, \mathbf{x}_i), \alpha_i \in \mathbb{R}\}$$

- For $f = \sum_i \alpha_i k(\cdot, \mathbf{x}_i)$, $g = \sum_j \beta_j k(\cdot, \mathbf{x}_j)$, define inner product

$$\langle f, g \rangle = \sum_{ij} \alpha_i \beta_j k(\mathbf{x}_i, \mathbf{x}_j)$$

# Kernels and Inner Product Spaces

- A kernel $k : \mathcal{X} \times \mathcal{X} \mapsto \mathbb{R}$ is positive semi-definite if
  - $k$ is symmetric, i.e., $k(\mathbf{x}_i, \mathbf{x}_j) = k(\mathbf{x}_j, \mathbf{x}_i)$
  - For $\{\mathbf{x}_1, \ldots, \mathbf{x}_n\}$, matrix $K(\mathbf{x}_i, \mathbf{x}_j)$ is positive semi-definite
- Inner product space for kernel $k$
  - The kernel feature map $\Phi(\mathbf{x}) = k(\cdot, \mathbf{x})$
    - Each point $x$ is now a function is some space
    - The function measures its similarity will all other points
  - The vector space corresponding to the mapping

$$\text{span}\{\Phi(\mathbf{x}) : \mathbf{x} \in \mathcal{X}\} = \{f(\cdot) = \sum_i \alpha_i k(\cdot, \mathbf{x}_i), \alpha_i \in \mathbb{R}\}$$

- For $f = \sum_i \alpha_i k(\cdot, \mathbf{x}_i)$, $g = \sum_j \beta_j k(\cdot, \mathbf{x}_j)$, define inner product

$$\langle f, g \rangle = \sum_{ij} \alpha_i \beta_j k(\mathbf{x}_i, \mathbf{x}_j)$$

  - Symmetric, bilinear, positive semi-definite

# Kernels and Inner Product Spaces

- A kernel $k : \mathcal{X} \times \mathcal{X} \mapsto \mathbb{R}$ is positive semi-definite if
  - $k$ is symmetric, i.e., $k(\mathbf{x}_i, \mathbf{x}_j) = k(\mathbf{x}_j, \mathbf{x}_i)$
  - For $\{\mathbf{x}_1, \ldots, \mathbf{x}_n\}$, matrix $K(\mathbf{x}_i, \mathbf{x}_j)$ is positive semi-definite
- Inner product space for kernel $k$
  - The kernel feature map $\Phi(\mathbf{x}) = k(\cdot, \mathbf{x})$
    - Each point $x$ is now a function is some space
    - The function measures its similarity will all other points
  - The vector space corresponding to the mapping

$$\text{span}\{\Phi(\mathbf{x}) : \mathbf{x} \in \mathcal{X}\} = \{f(\cdot) = \sum_i \alpha_i k(\cdot, \mathbf{x}_i), \alpha_i \in \mathbb{R}\}$$

- For $f = \sum_i \alpha_i k(\cdot, \mathbf{x}_i)$, $g = \sum_j \beta_j k(\cdot, \mathbf{x}_j)$, define inner product

$$\langle f, g \rangle = \sum_{ij} \alpha_i \beta_j k(\mathbf{x}_i, \mathbf{x}_j)$$

  - Symmetric, bilinear, positive semi-definite
  - Satisfies Cauchy-Schwartz inequality

# Reproducing Kernel Hilbert Space

- For any $f$ in the vector space, the reproducing property

$$\langle k(\cdot, \mathbf{x}), f \rangle = \sum_i \alpha_i k(\mathbf{x}, \mathbf{x}_i) = f(\mathbf{x})$$

# Reproducing Kernel Hilbert Space

- For any $f$ in the vector space, the reproducing property

$$\langle k(\cdot, \mathbf{x}), f \rangle = \sum_i \alpha_i k(\mathbf{x}, \mathbf{x}_i) = f(\mathbf{x})$$

- Consider $\mathcal{X} \in \mathbb{R}^d$, Hilbert space $\mathcal{H}$ of functions $f : \mathcal{X} \mapsto \mathbb{R}$

# Reproducing Kernel Hilbert Space

- For any $f$ in the vector space, the reproducing property
$$\langle k(\cdot, \mathbf{x}), f \rangle = \sum_i \alpha_i k(\mathbf{x}, \mathbf{x}_i) = f(\mathbf{x})$$
- Consider $\mathcal{X} \in \mathbb{R}^d$, Hilbert space $\mathcal{H}$ of functions $f : \mathcal{X} \mapsto \mathbb{R}$
- $\mathcal{H}$ is a Reproducing Kernel Hilbert Space if $\exists k : \mathcal{X} \times \mathcal{X} \mapsto \mathbb{R}$

# Reproducing Kernel Hilbert Space

- For any $f$ in the vector space, the reproducing property

$$\langle k(\cdot, \mathbf{x}), f \rangle = \sum_i \alpha_i k(\mathbf{x}, \mathbf{x}_i) = f(\mathbf{x})$$

- Consider $\mathcal{X} \in \mathbb{R}^d$, Hilbert space $\mathcal{H}$ of functions $f : \mathcal{X} \mapsto \mathbb{R}$
- $\mathcal{H}$ is a Reproducing Kernel Hilbert Space if $\exists k : \mathcal{X} \times \mathcal{X} \mapsto \mathbb{R}$
  - $k$ has the reproducing property $\langle k(\cdot, \mathbf{x}), f \rangle = f(\mathbf{x})$

# Reproducing Kernel Hilbert Space

- For any $f$ in the vector space, the reproducing property

$$\langle k(\cdot, \mathbf{x}), f \rangle = \sum_i \alpha_i k(\mathbf{x}, \mathbf{x}_i) = f(\mathbf{x})$$

- Consider $\mathcal{X} \in \mathbb{R}^d$, Hilbert space $\mathcal{H}$ of functions $f : \mathcal{X} \mapsto \mathbb{R}$
- $\mathcal{H}$ is a Reproducing Kernel Hilbert Space if $\exists k : \mathcal{X} \times \mathcal{X} \mapsto \mathbb{R}$
  - $k$ has the reproducing property $\langle k(\cdot, \mathbf{x}), f \rangle = f(\mathbf{x})$
  - $k$ spans $\mathcal{H}$

# Reproducing Kernel Hilbert Space

- For any $f$ in the vector space, the reproducing property

$$\langle k(\cdot, \mathbf{x}), f \rangle = \sum_i \alpha_i k(\mathbf{x}, \mathbf{x}_i) = f(\mathbf{x})$$

- Consider $\mathcal{X} \in \mathbb{R}^d$, Hilbert space $\mathcal{H}$ of functions $f : \mathcal{X} \mapsto \mathbb{R}$
- $\mathcal{H}$ is a Reproducing Kernel Hilbert Space if $\exists k : \mathcal{X} \times \mathcal{X} \mapsto \mathbb{R}$
  - $k$ has the reproducing property $\langle k(\cdot, \mathbf{x}), f \rangle = f(\mathbf{x})$
  - $k$ spans $\mathcal{H}$
- It is necessary that $k$ is symmetric and positive semi-definite

$$k = \sum_{t=1}^{\infty} \lambda_t \psi_t \psi_t^T \quad \Rightarrow \quad K(\mathbf{x}_i, \mathbf{x}_j) = \sum_{t=1}^{\infty} \lambda_t \psi_t(\mathbf{x}_i) \psi_t(\mathbf{x}_j)$$

# Reproducing Kernel Hilbert Space

- For any $f$ in the vector space, the reproducing property

$$\langle k(\cdot, \mathbf{x}), f \rangle = \sum_i \alpha_i k(\mathbf{x}, \mathbf{x}_i) = f(\mathbf{x})$$

- Consider $\mathcal{X} \in \mathbb{R}^d$, Hilbert space $\mathcal{H}$ of functions $f : \mathcal{X} \mapsto \mathbb{R}$
- $\mathcal{H}$ is a Reproducing Kernel Hilbert Space if $\exists k : \mathcal{X} \times \mathcal{X} \mapsto \mathbb{R}$
  - $k$ has the reproducing property $\langle k(\cdot, \mathbf{x}), f \rangle = f(\mathbf{x})$
  - $k$ spans $\mathcal{H}$
- It is necessary that $k$ is symmetric and positive semi-definite

$$k = \sum_{t=1}^{\infty} \lambda_t \psi_t \psi_t^T \quad \Rightarrow \quad K(\mathbf{x}_i, \mathbf{x}_j) = \sum_{t=1}^{\infty} \lambda_t \psi_t(\mathbf{x}_i) \psi_t(\mathbf{x}_j)$$

- Two possible feature representations

# Reproducing Kernel Hilbert Space

- For any $f$ in the vector space, the reproducing property

$$\langle k(\cdot, \mathbf{x}), f \rangle = \sum_i \alpha_i k(\mathbf{x}, \mathbf{x}_i) = f(\mathbf{x})$$

- Consider $\mathcal{X} \in \mathbb{R}^d$, Hilbert space $\mathcal{H}$ of functions $f : \mathcal{X} \mapsto \mathbb{R}$
- $\mathcal{H}$ is a Reproducing Kernel Hilbert Space if $\exists k : \mathcal{X} \times \mathcal{X} \mapsto \mathbb{R}$
  - $k$ has the reproducing property $\langle k(\cdot, \mathbf{x}), f \rangle = f(\mathbf{x})$
  - $k$ spans $\mathcal{H}$
- It is necessary that $k$ is symmetric and positive semi-definite

$$k = \sum_{t=1}^{\infty} \lambda_t \psi_t \psi_t^T \quad \Rightarrow \quad K(\mathbf{x}_i, \mathbf{x}_j) = \sum_{t=1}^{\infty} \lambda_t \psi_t(\mathbf{x}_i) \psi_t(\mathbf{x}_j)$$

- Two possible feature representations
  - $\Phi(\mathbf{x}) = k(\cdot, \mathbf{x})$ such that $\langle k(\cdot, \mathbf{x}_i), k(\cdot, \mathbf{x}_j) \rangle = k(\mathbf{x}_i, \mathbf{x}_j)$

# Reproducing Kernel Hilbert Space

- For any $f$ in the vector space, the reproducing property

$$\langle k(\cdot, \mathbf{x}), f \rangle = \sum_i \alpha_i k(\mathbf{x}, \mathbf{x}_i) = f(\mathbf{x})$$

- Consider $\mathcal{X} \in \mathbb{R}^d$, Hilbert space $\mathcal{H}$ of functions $f : \mathcal{X} \mapsto \mathbb{R}$
- $\mathcal{H}$ is a Reproducing Kernel Hilbert Space if $\exists k : \mathcal{X} \times \mathcal{X} \mapsto \mathbb{R}$
  - $k$ has the reproducing property $\langle k(\cdot, \mathbf{x}), f \rangle = f(\mathbf{x})$
  - $k$ spans $\mathcal{H}$
- It is necessary that $k$ is symmetric and positive semi-definite

$$k = \sum_{t=1}^{\infty} \lambda_t \psi_t \psi_t^T \quad \Rightarrow \quad K(\mathbf{x}_i, \mathbf{x}_j) = \sum_{t=1}^{\infty} \lambda_t \psi_t(\mathbf{x}_i) \psi_t(\mathbf{x}_j)$$

- Two possible feature representations
  - $\Phi(\mathbf{x}) = k(\cdot, \mathbf{x})$ such that $\langle k(\cdot, \mathbf{x}_i), k(\cdot, \mathbf{x}_j) \rangle = k(\mathbf{x}_i, \mathbf{x}_j)$
  - $\Phi(\mathbf{x}) = (\sqrt{\lambda_1}\psi_1(\mathbf{x}), \sqrt{\lambda_2}\psi_2(\mathbf{x}), \ldots)$ with usual inner product

# The Representer Theorem

- Let $\mathcal{H}$ be a reproducing kernel Hilbert space

# The Representer Theorem

- Let $\mathcal{H}$ be a reproducing kernel Hilbert space
- $k : \mathcal{X} \times \mathcal{X} \mapsto \mathbb{R}$ is a symmetric positive semi-definite function

# The Representer Theorem

- Let $\mathcal{H}$ be a reproducing kernel Hilbert space
- $k : \mathcal{X} \times \mathcal{X} \mapsto \mathbb{R}$ is a symmetric positive semi-definite function
- For $f \in \mathcal{H}$, the norm $\|f\|_{\mathcal{H}}^2 = \langle f, f \rangle$

# The Representer Theorem

- Let $\mathcal{H}$ be a reproducing kernel Hilbert space
- $k : \mathcal{X} \times \mathcal{X} \mapsto \mathbb{R}$ is a symmetric positive semi-definite function
- For $f \in \mathcal{H}$, the norm $\|f\|_{\mathcal{H}}^2 = \langle f, f \rangle$
- For any $L : \mathbb{R}^n \mapsto \mathbb{R}$, any non-decreasing $\Omega : \mathbb{R} \mapsto \mathbb{R}$, consider

$$\min_{f \in \mathcal{H}} \left\{ \Omega(\|f\|_{\mathcal{H}}^2) + L(f(\mathbf{x}_1), \ldots, f(\mathbf{x}_n)) \right\}$$

# The Representer Theorem

- Let $\mathcal{H}$ be a reproducing kernel Hilbert space
- $k : \mathcal{X} \times \mathcal{X} \mapsto \mathbb{R}$ is a symmetric positive semi-definite function
- For $f \in \mathcal{H}$, the norm $\|f\|_{\mathcal{H}}^2 = \langle f, f \rangle$
- For any $L : \mathbb{R}^n \mapsto \mathbb{R}$, any non-decreasing $\Omega : \mathbb{R} \mapsto \mathbb{R}$, consider

$$\min_{f \in \mathcal{H}} \{\Omega(\|f\|_{\mathcal{H}}^2) + L(f(\mathbf{x}_1), \ldots, f(\mathbf{x}_n))\}$$

- For some $\alpha_1, \ldots, \alpha_n \in \mathbb{R}$, the minimum is achieved by

$$f(\cdot) = \sum_{i=1}^{n} \alpha_i k(\mathbf{x}_i, \cdot)$$

# The Representer Theorem

- Let $\mathcal{H}$ be a reproducing kernel Hilbert space
- $k : \mathcal{X} \times \mathcal{X} \mapsto \mathbb{R}$ is a symmetric positive semi-definite function
- For $f \in \mathcal{H}$, the norm $\|f\|_{\mathcal{H}}^2 = \langle f, f \rangle$
- For any $L : \mathbb{R}^n \mapsto \mathbb{R}$, any non-decreasing $\Omega : \mathbb{R} \mapsto \mathbb{R}$, consider

$$\min_{f \in \mathcal{H}} \{\Omega(\|f\|_{\mathcal{H}}^2) + L(f(\mathbf{x}_1), \ldots, f(\mathbf{x}_n))\}$$

- For some $\alpha_1, \ldots, \alpha_n \in \mathbb{R}$, the minimum is achieved by

$$f(\cdot) = \sum_{i=1}^{n} \alpha_i k(\mathbf{x}_i, \cdot)$$

- If $\Omega$ is increasing, each minimizer has the above form

# Kernel Logistic Regression

- The kernel trick can be applied elsewhere

# Kernel Logistic Regression

- The kernel trick can be applied elsewhere
  - Several problems are in “dot product” form

# Kernel Logistic Regression

- The kernel trick can be applied elsewhere
  - Several problems are in “dot product” form
  - Extensions to non-vector data types using kernels

# Kernel Logistic Regression

- The kernel trick can be applied elsewhere
  - Several problems are in "dot product" form
  - Extensions to non-vector data types using kernels
- Kernel Logistic Regression (KLR)

# Kernel Logistic Regression

- The kernel trick can be applied elsewhere
  - Several problems are in "dot product" form
  - Extensions to non-vector data types using kernels
- Kernel Logistic Regression (KLR)
  - Log-odds is linear in high-dimensional representation

$$\log\left(\frac{P(1|\mathbf{x})}{P(0|\mathbf{x})}\right) = \langle \mathbf{w}, \Phi(\mathbf{x}) \rangle + w_0$$

# Kernel Logistic Regression

- The kernel trick can be applied elsewhere
  - Several problems are in "dot product" form
  - Extensions to non-vector data types using kernels
- Kernel Logistic Regression (KLR)
  - Log-odds is linear in high-dimensional representation

$$\log\left(\frac{P(1|\mathbf{x})}{P(0|\mathbf{x})}\right) = \langle \mathbf{w}, \Phi(\mathbf{x}) \rangle + w_0$$

  - The regularized KLR minimizes

$$L = \lambda \|\mathbf{w}\|^2 - \sum_i \log P(y_i|\mathbf{x}_i)$$

# Kernel Logistic Regression

- The kernel trick can be applied elsewhere
  - Several problems are in "dot product" form
  - Extensions to non-vector data types using kernels
- Kernel Logistic Regression (KLR)
  - Log-odds is linear in high-dimensional representation

$$\log\left(\frac{P(1|\mathbf{x})}{P(0|\mathbf{x})}\right) = \langle \mathbf{w}, \Phi(\mathbf{x}) \rangle + w_0$$

  - The regularized KLR minimizes

$$L = \lambda \|\mathbf{w}\|^2 - \sum_i \log P(y_i|\mathbf{x}_i)$$

  - From the representer theorem

$$\mathbf{w} = \sum_i \alpha_i \Phi(\mathbf{x}_i) \quad \Rightarrow \quad \log\left(\frac{P(1|\mathbf{x})}{P(0|\mathbf{x})}\right) = \sum_i \alpha_i k(\mathbf{x}_i, \mathbf{x}) + w_0$$

# Kernel Logistic Regression

- The kernel trick can be applied elsewhere
  - Several problems are in "dot product" form
  - Extensions to non-vector data types using kernels
- Kernel Logistic Regression (KLR)
  - Log-odds is linear in high-dimensional representation

$$\log\left(\frac{P(1|\mathbf{x})}{P(0|\mathbf{x})}\right) = \langle \mathbf{w}, \Phi(\mathbf{x}) \rangle + w_0$$

  - The regularized KLR minimizes

$$L = \lambda \|\mathbf{w}\|^2 - \sum_i \log P(y_i|\mathbf{x}_i)$$

  - From the representer theorem

$$\mathbf{w} = \sum_i \alpha_i \Phi(\mathbf{x}_i) \quad \Rightarrow \quad \log\left(\frac{P(1|\mathbf{x})}{P(0|\mathbf{x})}\right) = \sum_i \alpha_i k(\mathbf{x}_i, \mathbf{x}) + w_0$$

- An efficient algorithm can be designed to learn $\alpha_1, \ldots, \alpha_n$

# Kernel Fisher Discriminant

- Recall Fisher's Linear Discriminant

$$
\begin{aligned}
J(\mathbf{w}) &= \frac{\mathbf{w}^T S_B \mathbf{w}}{\mathbf{w}^T S_w \mathbf{w}} \\
S_B &= (m_2 - m_1)(m_2 - m_1)^T \\
S_W &= \sum_{k=1,2} \sum_{i \in C_k} (\mathbf{x}_i - m_k)(\mathbf{x}_i - m_k)^T
\end{aligned}
$$

# Kernel Fisher Discriminant

- Recall Fisher's Linear Discriminant

$$
\begin{aligned}
J(\mathbf{w}) &= \frac{\mathbf{w}^T S_B \mathbf{w}}{\mathbf{w}^T S_w \mathbf{w}} \\
S_B &= (m_2 - m_1)(m_2 - m_1)^T \\
S_W &= \sum_{k=1,2} \sum_{i \in C_k} (\mathbf{x}_i - m_k)(\mathbf{x}_i - m_k)^T
\end{aligned}
$$

- Represent $\mathbf{w}$ in terms of mapped training points:

$$
\begin{aligned}
\mathbf{w} &= \sum_{i=1}^{n} \alpha_i \phi(\mathbf{x}_i) \\
\langle \mathbf{w}, \phi(\mathbf{x}) \rangle &= \sum_{i=1}^{n} \alpha_i K(\mathbf{x}_i, x)
\end{aligned}
$$

# Kernel Fisher Discriminant (Contd.)

The corresponding Rayleigh coefficient

$$J(\alpha) = \frac{(\alpha^T \mu)^2}{\alpha^T N \alpha} = \frac{\alpha^T M \alpha}{\alpha^T N \alpha}$$

where

$$\begin{aligned}
\mu &= \mu_2 - \mu_1 \\
M &= \mu \mu^T \\
\mu_k &= \frac{1}{|C_k|} K \mathbf{1}_k \\
N &= K K^T - \sum_{k=1,2} |C_k| \mu_k \mu_k^T
\end{aligned}$$