University of Colorado **Boulder**

Department of Computer Science

CSCI 5622: Machine Learning

Chenhao Tan

Lecture 15: Duality & Kernels

Slides adapted from Chris Ketelsen, Jordan Boyd-Graber, and Noah Smith

Administrivia

- HW4 is released
- Final project started!

Outline

- Duality
- Kernels

Outline

- Duality
- Kernels

Binary classification

Given: $S_{\text{train}} = \{(\boldsymbol{x}_i, y_i)\}_{i=1}^{m}$ training examples, $\boldsymbol{x}_i \in \mathbb{R}^d$, $y_i \in \{-1, 1\}$
Goal: Find hypothesis function $h : X \to Y$
Linear SVM: learn a linear decision rule of the form $\boldsymbol{w} \cdot \boldsymbol{x} + b$

Optimizing the objective function

$$\min_{\boldsymbol{w},b} \frac{1}{2}||\boldsymbol{w}||^2$$

subject to $y_i(\boldsymbol{w} \cdot \boldsymbol{x}_i + b) \geq 1, i \in [1, m]$
This is a quadratic objective function with linear inequality constraints. Many off-the-shelf optimization methods are available.

The Method of Lagrange Multipliers

Constrained problem (Primal problem)

$$\min_{\boldsymbol{x}} f(\boldsymbol{x})$$

$$\text{s.t. } g_i(\boldsymbol{x}) \geq 0, i \in [1, n]$$

Lagrange Multiplier

$$\mathscr{L}(\boldsymbol{x}, \boldsymbol{\alpha}) = f(\boldsymbol{x}) - \sum_{i=1}^{n} \alpha_i g_i(\boldsymbol{x}),$$

$$\alpha_i \geq 0, i \in [1, n]$$

Lagrange Multiplier

p^*: the optimal value in the primal problem
We claim that

$\min_x f(x)$
s.t. $g_i(x) \geq 0$

$$p^* = \min_{\boldsymbol{x}} \max_{\boldsymbol{\alpha}} \mathscr{L}(\boldsymbol{x}, \boldsymbol{\alpha}) = \min_{\boldsymbol{x}} \max_{\boldsymbol{\alpha}} f(\boldsymbol{x}) - \sum_{i=1}^{n} \alpha_i g_i(\boldsymbol{x})$$

p^*: the optimal value in the primal problem
We claim that

$$p^* = \min_{\boldsymbol{x}} \max_{\boldsymbol{\alpha}} \mathscr{L}(\boldsymbol{x}, \boldsymbol{\alpha}) = \min_{\boldsymbol{x}} \max_{\boldsymbol{\alpha}} f(\boldsymbol{x}) - \sum_{i=1}^{n} \alpha_i g_i(\boldsymbol{x})$$

This is because

$\alpha \geq 0$

$$\max -\boldsymbol{\alpha} \boldsymbol{y} = \begin{cases} 0 & \boldsymbol{y} \geq 0 \\ +\infty & \text{otherwise} \end{cases}$$

What happens if we reverse min and max:

$$\max_{\boldsymbol{\alpha}} \min_{\boldsymbol{x}} \mathscr{L}(\boldsymbol{x}, \boldsymbol{\alpha}) \leq \underline{\min_{\boldsymbol{x}} \max_{\boldsymbol{\alpha}}} \mathscr{L}(\boldsymbol{x}, \boldsymbol{\alpha})$$

The left leads to the dual problem.

Primal vs. Dual

Primal problem

$$\min_{\boldsymbol{w},b} \frac{1}{2}||\boldsymbol{w}||^2$$

s.t. $y_i(\boldsymbol{w} \cdot \boldsymbol{x}_i + b) \geq 1, i \in [1, m]$

Derive the function for dual problem.
Replace w, b with stationarity conditions.
(There will be detailed derivations for the soft-margin case later.)

Primal vs. Dual

Primal problem

$$\min_{\boldsymbol{w},b} \frac{1}{2}||\boldsymbol{w}||^2$$

$$\text{s.t. } y_i(\boldsymbol{w} \cdot \boldsymbol{x}_i + b) \geq 1, i \in [1, m]$$

$\min_{w,b} \max_{\alpha}$

Dual problem

$$\max_{\boldsymbol{\alpha}} \underline{\sum_{i=1}^{m} \alpha_i - \frac{1}{2} \sum_{i=1}^{m} \sum_{j=1}^{m} \alpha_i \alpha_j y_i y_j (\boldsymbol{x}_j \cdot \boldsymbol{x}_i)}$$

$$\text{s.t. } \alpha_i \geq 0, i \in [1, m]$$

$$\sum_i \alpha_i y_i = 0$$

$\max_{\alpha} \min_{\underline{w,b}}$

Karush-Kuhn-Tucker (KKT) conditions

Primal and dual feasibility

$$y_i(\boldsymbol{w} \cdot \boldsymbol{x}_i + b) \geq 1, \alpha_i \geq 0$$

Karush-Kuhn-Tucker (KKT) conditions

Primal and dual feasibility

$$y_i(\boldsymbol{w} \cdot \boldsymbol{x}_i + b) \geq 1, \alpha_i \geq 0$$

Karush-Kuhn-Tucker (KKT) conditions

Primal and dual feasibility

$$y_i(\boldsymbol{w} \cdot \boldsymbol{x}_i + b) \geq 1, \alpha_i \geq 0$$

Stationarity

$$\boldsymbol{w} = \sum_{i=1}^{m} \alpha_i y_i \boldsymbol{x}_i, \sum_{i=1}^{m} \alpha_i y_i = 0$$

Karush-Kuhn-Tucker (KKT) conditions

Primal and dual feasibility

$$y_i(\boldsymbol{w} \cdot \boldsymbol{x}_i + b) \geq 1, \alpha_i \geq 0$$

Stationarity

$$\boldsymbol{w} = \sum_{i=1}^{m} \alpha_i y_i \boldsymbol{x}_i, \sum_{i=1}^{m} \alpha_i y_i = 0$$

Remember that two properties about support vector machine directly follows from this:

- Only support vectors affect the weights ($\alpha_i > 0$).
- There must be both positive and negative support vectors.

Karush-Kuhn-Tucker (KKT) conditions

Primal and dual feasibility

$$y_i(\boldsymbol{w} \cdot \boldsymbol{x}_i + b) \geq 1, \alpha_i \geq 0$$

Stationarity

$$\boldsymbol{w} = \sum_{i=1}^{m} \alpha_i y_i \boldsymbol{x}_i, \sum_{i=1}^{m} \alpha_i y_i = 0$$

Complementary slackness

$$\alpha_i = 0 \vee y_i(\boldsymbol{w} \cdot \boldsymbol{x}_i + b) = 1$$

What is the dual problem of soft-margin SVM?

$f(x)$

$$\min_{w,b,\xi} \underline{\frac{1}{2}||\boldsymbol{w}||^2 + C\sum_{i=1} \xi_i}$$

subject to

$$y_i(\boldsymbol{w} \cdot \boldsymbol{x}_i + b) \geq 1 - \xi_i, i \in [1, m] \quad g_i(x)$$

$$\xi_i \geq 0, i \in [1, m] \quad 2m$$

New Lagrangian

$$\mathcal{L}(\boldsymbol{w}, b, \boldsymbol{\xi}, \boldsymbol{\alpha}, \boldsymbol{\beta}) = \frac{1}{2}||\boldsymbol{w}||^2 + C\sum_{i=1}^{m}\xi_i$$
$$- \sum_{i=1}^{m}\alpha_i\left[y_i(\boldsymbol{w}\cdot\boldsymbol{x}_i + b) - 1 + \xi_i\right]$$
$$- \sum_{i=1}^{m}\beta_i\xi_i$$

$\frac{\partial L}{\partial w} = 0$ $\quad w - \sum_i \alpha_i y_i x_i = 0 \Rightarrow w = \sum_i \alpha_i y_i x_i$

$\frac{\partial L}{\partial b} = 0$ $\quad -\sum_{i=1}^{m} \alpha_i y_i = 0 \Rightarrow \sum_i \alpha_i y_i = 0$

$\frac{\partial L}{\partial \xi} = 0$ $\quad \frac{\partial L}{\partial \xi_i} = C - \alpha_i - \beta_i = 0 \Rightarrow \alpha_i + \beta_i = C$

New Lagrangian

$$
\begin{aligned}
\mathscr{L}(\boldsymbol{w}, b, \boldsymbol{\xi}, \boldsymbol{\alpha}, \boldsymbol{\beta}) =& \frac{1}{2}||\boldsymbol{w}||^2 + C\sum_{i=1}^{m}\xi_i \\
& - \sum_{i=1}^{m}\alpha_i\left[y_i(\boldsymbol{w}\cdot\boldsymbol{x}_i + b) - 1 + \xi_i\right] \\
& - \sum_{i=1}^{m}\beta_i\xi_i
\end{aligned}
$$

Taking the gradients ($\nabla_{\boldsymbol{w}}\mathscr{L}, \nabla_b\mathscr{L}, \nabla_{\xi_i}\mathscr{L}$) and solving for zero gives us

$$
\boldsymbol{w} = \sum_{i=1}^{m}\alpha_i y_i \boldsymbol{x}_i \qquad \sum_{i=1}^{m}\alpha_i y_i = 0 \qquad \alpha_i + \beta_i = C
$$

New Lagrangian

$$\begin{aligned}\mathscr{L}(\boldsymbol{w}, b, \boldsymbol{\xi}, \boldsymbol{\alpha}, \boldsymbol{\beta}) =& \frac{1}{2}||\boldsymbol{w}||^2 + C\sum_{i=1}^{m}\xi_i \\ &- \sum_{i=1}^{m}\alpha_i\left[y_i(\boldsymbol{w}\cdot\boldsymbol{x}_i + b) - 1 + \xi_i\right] \\ &- \sum_{i=1}^{m}\beta_i\xi_i\end{aligned}$$

Taking the gradients ($\nabla_{\boldsymbol{w}}\mathscr{L}, \nabla_b\mathscr{L}, \nabla_{\xi_i}\mathscr{L}$) and solving for zero gives us

$$\boldsymbol{w} = \sum_{i=1}^{m}\alpha_i y_i \boldsymbol{x}_i \qquad \sum_{i=1}^{m}\alpha_i y_i = 0 \qquad \alpha_i + \beta_i = C$$

New Lagrangian

$$
\begin{aligned}
\mathscr{L}(\boldsymbol{w}, b, \boldsymbol{\xi}, \boldsymbol{\alpha}, \boldsymbol{\beta}) = & \frac{1}{2}||\boldsymbol{w}||^2 + C\sum_{i=1}^{m}\xi_i \\
& - \sum_{i=1}^{m}\alpha_i\left[y_i(\boldsymbol{w}\cdot\boldsymbol{x}_i + b) - 1 + \xi_i\right] \\
& - \sum_{i=1}^{m}\beta_i\xi_i
\end{aligned}
$$

Taking the gradients ($\nabla_{\boldsymbol{w}}\mathscr{L}, \nabla_b\mathscr{L}, \nabla_{\xi_i}\mathscr{L}$) and solving for zero gives us

$$\boldsymbol{w} = \sum_{i=1}^{m}\alpha_i y_i \boldsymbol{x}_i \qquad \sum_{i=1}^{m}\alpha_i y_i = 0 \qquad \alpha_i + \beta_i = C$$

New Lagrangian

$$\begin{aligned}\mathscr{L}(\boldsymbol{w}, b, \boldsymbol{\xi}, \boldsymbol{\alpha}, \boldsymbol{\beta}) = & \frac{1}{2}||\boldsymbol{w}||^2 + C\sum_{i=1}^{m}\xi_i \\ & - \sum_{i=1}^{m}\alpha_i\left[y_i(\boldsymbol{w}\cdot\boldsymbol{x}_i + b) - 1 + \xi_i\right] \\ & - \sum_{i=1}^{m}\beta_i\xi_i\end{aligned}$$

Taking the gradients ($\nabla_{\boldsymbol{w}}\mathscr{L}, \nabla_b\mathscr{L}, \nabla_{\xi_i}\mathscr{L}$) and solving for zero gives us

$$\boldsymbol{w} = \sum_{i=1}^{m}\alpha_i y_i \boldsymbol{x}_i \qquad \sum_{i=1}^{m}\alpha_i y_i = 0 \qquad \alpha_i + \beta_i = C$$

Simplifying dual objective

$$\boldsymbol{w} = \sum_{i=1}^{m} \alpha_i y_i \boldsymbol{x}_i$$

$$\sum_{i=1}^{m} \alpha_i y_i = 0$$

$$\alpha_i + \beta_i = C$$

Simplifying dual objective

$$\boldsymbol{w} = \sum_{i=1}^{m} \alpha_i y_i \boldsymbol{x}_i \qquad \sum_{i=1}^{m} \alpha_i y_i = 0 \qquad \alpha_i + \beta_i = C$$

$$\begin{aligned}
\mathscr{L}(\boldsymbol{w}, b, \boldsymbol{\xi}, \boldsymbol{\alpha}, \boldsymbol{\beta}) =& \frac{1}{2}||\boldsymbol{w}||^2 + C\sum_{i=1}^{m} \xi_i \\
& - \sum_{i=1}^{m} \alpha_i \left[y_i(\boldsymbol{w} \cdot \boldsymbol{x}_i + b) - 1 + \xi_i\right] \\
& - \sum_{i=1}^{m} \beta_i \xi_i
\end{aligned}$$

Dual Problem

$$\max_{\boldsymbol{\alpha}} \sum_{i=1}^{m} \alpha_i - \frac{1}{2} \sum_{i=1}^{m} \sum_{j=1}^{m} \alpha_i \alpha_j y_i y_j (\boldsymbol{x}_j \cdot \boldsymbol{x}_i)$$

$$\text{s.t. } C \geq \alpha_i \geq 0, i \in [1, m]$$

$$\sum_i \alpha_i y_i = 0$$

$\alpha_i + \beta_i = C$

$\beta_i \geq 0$

Dual Problem

$$\max_{\boldsymbol{\alpha}} \sum_{i=1}^{m} \alpha_i - \frac{1}{2} \sum_{i=1}^{m} \sum_{j=1}^{m} \alpha_i \alpha_j y_i y_j (\boldsymbol{x}_j \cdot \boldsymbol{x}_i)$$

$$\text{s.t. } C \geq \alpha_i \geq 0, i \in [1, m]$$

$$\sum_i \alpha_i y_i = 0$$

Karush-Kuhn-Tucker (KKT) conditions

Primal and dual feasibility

$$y_i(\boldsymbol{w} \cdot \boldsymbol{x}_i + b) \geq 1 - \xi_i, \xi_i \geq 0, C \geq \alpha_i \geq 0, \beta_i \geq 0$$

Karush-Kuhn-Tucker (KKT) conditions

Primal and dual feasibility

$$y_i(\boldsymbol{w} \cdot \boldsymbol{x}_i + b) \geq 1 - \xi_i, \xi_i \geq 0, C \geq \alpha_i \geq 0, \beta_i \geq 0$$

Stationarity

$$\boldsymbol{w} = \sum_{i=1}^{m} \alpha_i y_i \boldsymbol{x}_i, \sum_{i=1}^{m} \alpha_i y_i = 0, \alpha_i + \beta_i = C$$

Karush-Kuhn-Tucker (KKT) conditions

Primal and dual feasibility

$$y_i(\boldsymbol{w} \cdot \boldsymbol{x}_i + b) \geq 1 - \xi_i,\ \xi_i \geq 0,\ C \geq \alpha_i \geq 0,\ \beta_i \geq 0$$

Stationarity

$$\boldsymbol{w} = \sum_{i=1}^{m} \alpha_i y_i \boldsymbol{x}_i,\ \sum_{i=1}^{m} \alpha_i y_i = 0,\ \alpha_i + \beta_i = C$$

Complementary slackness

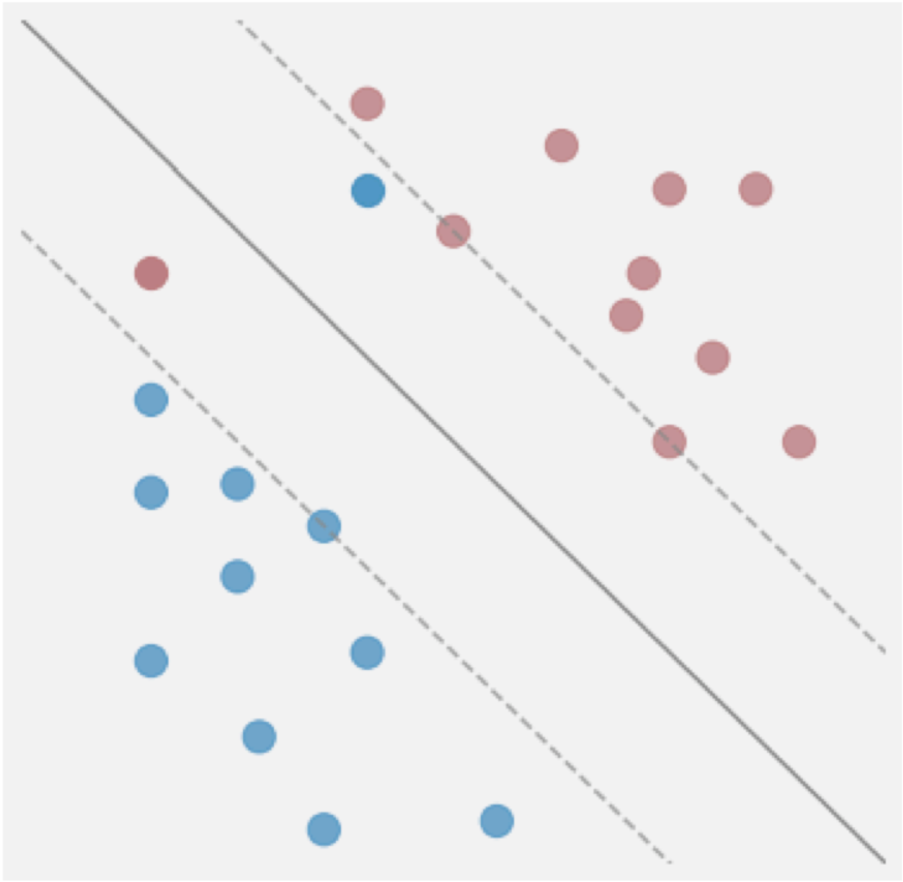
$$\underline{\alpha_i}[\underline{y_i(\boldsymbol{w} \cdot \boldsymbol{x}_i + b) - 1 + \xi_i}] = 0,\ \beta_i \xi_i = 0$$

$\beta_i = 0 \lor \xi_i = 0$

More on Complementary Slackness

$$\alpha_i[y_i(\boldsymbol{w} \cdot \boldsymbol{x}_i + b) - 1 + \xi_i] = 0, \beta_i \xi_i = 0$$

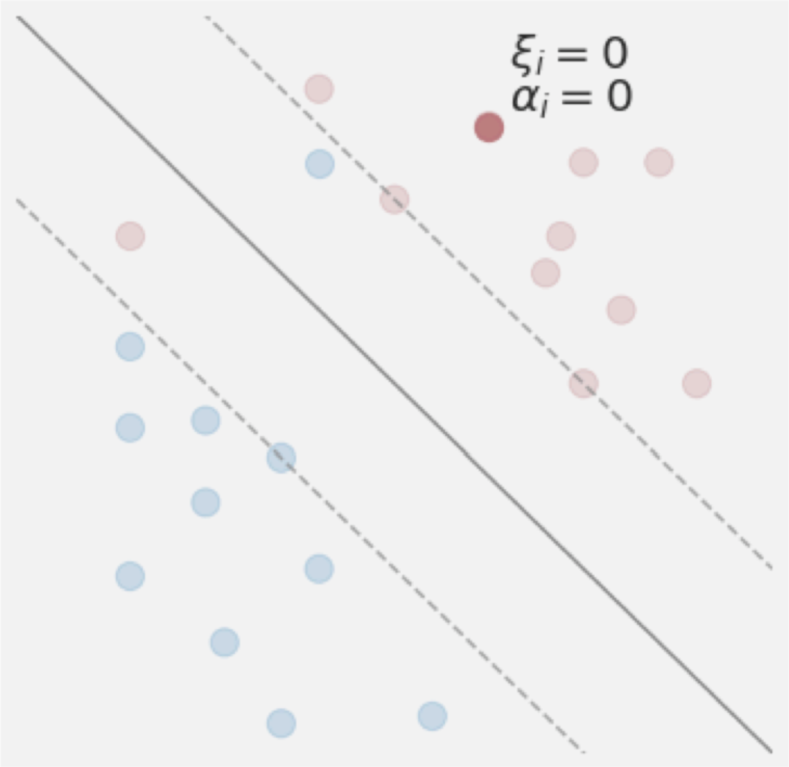
Also, $\alpha_i + \beta_i = C$



More on Complementary Slackness

$$\alpha_i[y_i(\boldsymbol{w} \cdot \boldsymbol{x}_i + b) - 1 + \xi_i] = 0, \beta_i \xi_i = 0$$

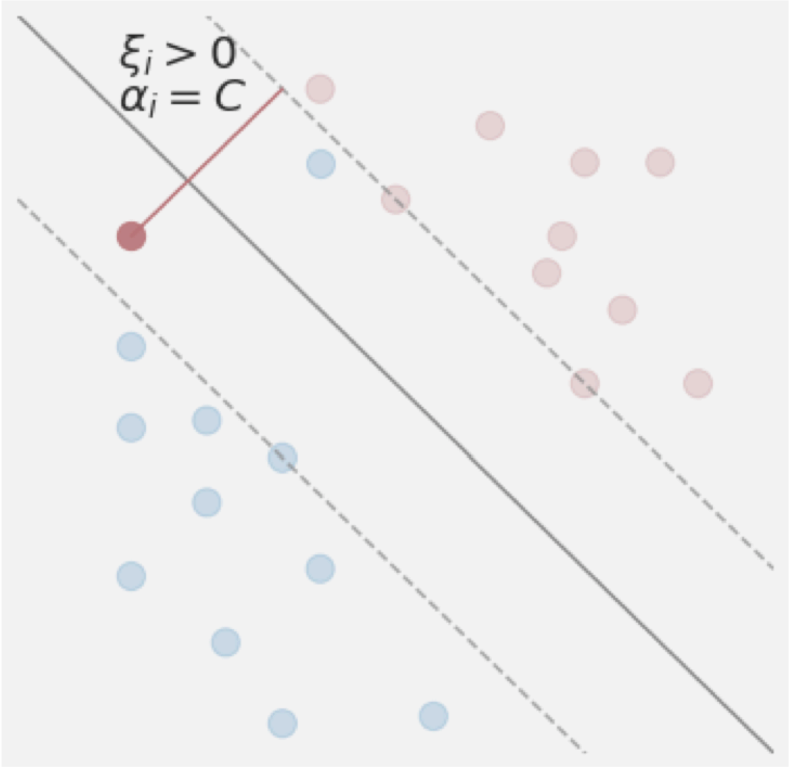
- $\boldsymbol{x}_i$ satisfies the margin,
  $y_i(\boldsymbol{w} \cdot \boldsymbol{x}_i + b) > 1 \Rightarrow \alpha_i = 0$ $\Rightarrow \beta_i = C$



More on Complementary Slackness

$$\alpha_i[y_i(\boldsymbol{w} \cdot \boldsymbol{x}_i + b) - 1 + \xi_i] = 0, \beta_i \xi_i = 0$$

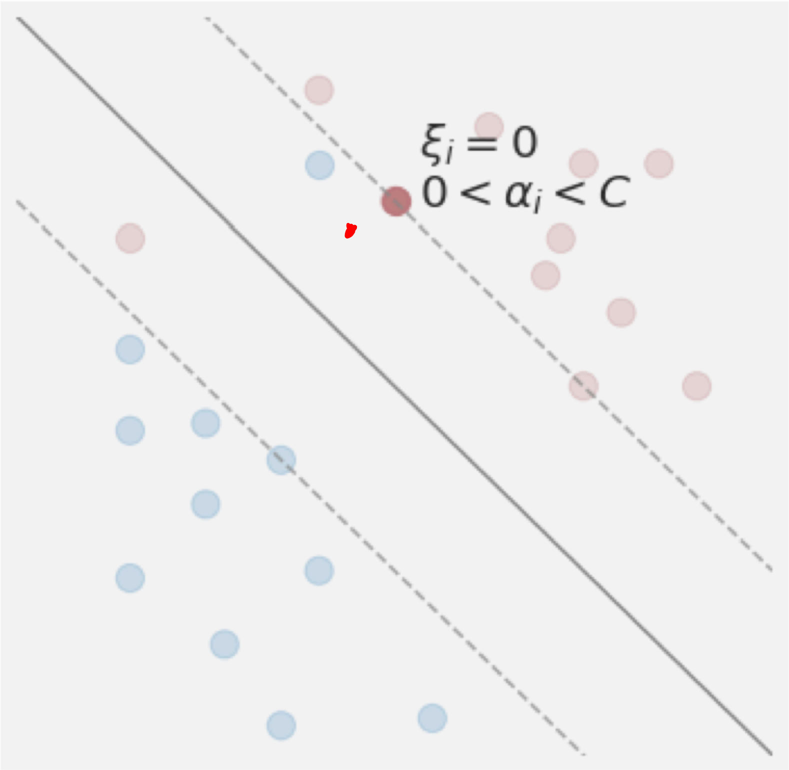
- $\boldsymbol{x}_i$ satisfies the margin,
  $y_i(\boldsymbol{w} \cdot \boldsymbol{x}_i + b) > 1 \Rightarrow \alpha_i = 0$
- $\boldsymbol{x}_i$ does not satisfy the margin,
  $y_i(\boldsymbol{w} \cdot \boldsymbol{x}_i + b) < 1 \Rightarrow \alpha_i = C$



More on Complementary Slackness

$$\alpha_i[y_i(\boldsymbol{w} \cdot \boldsymbol{x}_i + b) - 1 + \xi_i] = 0, \beta_i \xi_i = 0$$

$\sum_i \alpha_i y_i = 0$



- $\boldsymbol{x}_i$ satisfies the margin,
  $y_i(\boldsymbol{w} \cdot \boldsymbol{x}_i + b) > 1 \Rightarrow \alpha_i = 0$
- $\boldsymbol{x}_i$ does not satisfy the margin,
  $y_i(\boldsymbol{w} \cdot \boldsymbol{x}_i + b) < 1 \Rightarrow \alpha_i = C$
- $\boldsymbol{x}_i$ is on the margin,
  $y_i(\boldsymbol{w} \cdot \boldsymbol{x}_i + b) = 1 \Rightarrow 0 \leq \alpha_i \leq C$

Karush-Kuhn-Tucker (KKT) conditions

Primal and dual feasibility

$$y_i(\boldsymbol{w} \cdot \boldsymbol{x}_i + b) \geq 1 - \xi_i, \xi_i \geq 0, C \geq \alpha_i \geq 0, \beta_i \geq 0$$

Stationarity

$$\boldsymbol{w} = \sum_{i=1}^{m} \alpha_i y_i \boldsymbol{x}_i, \sum_{i=1}^{m} \alpha_i y_i = 0, \alpha_i + \beta_i = C$$

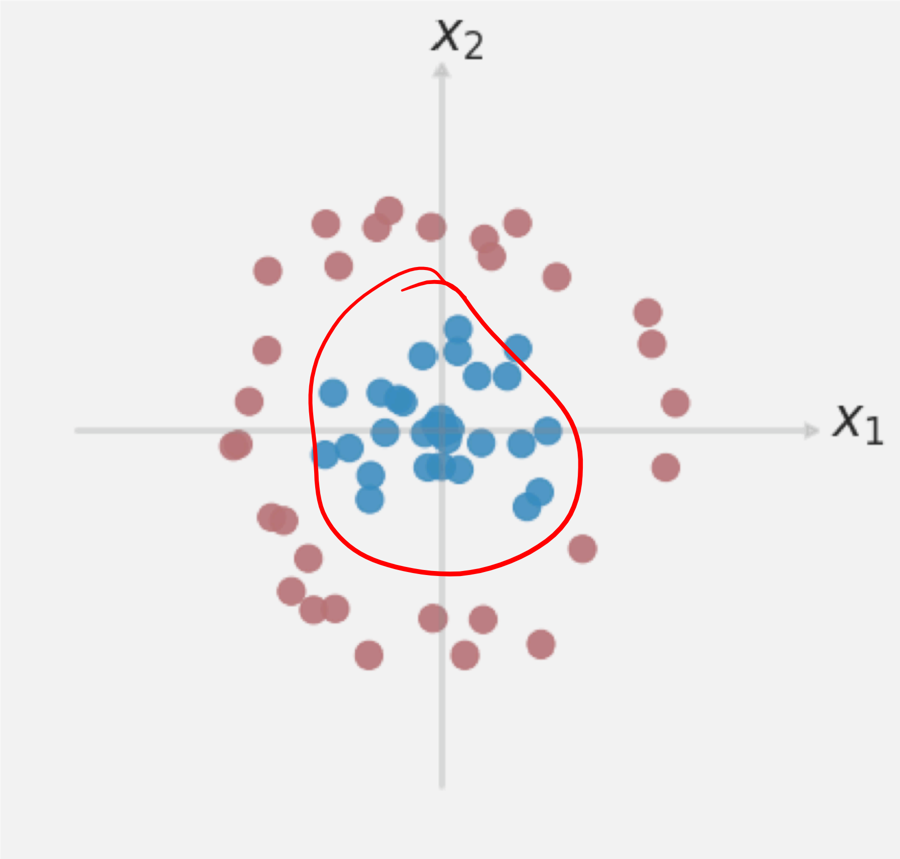
Complementary slackness

$$\alpha_i[y_i(\boldsymbol{w} \cdot \boldsymbol{x}_i + b) - 1 + \xi_i] = 0, \beta_i \xi_i = 0$$

Outline

- Duality
- Kernels

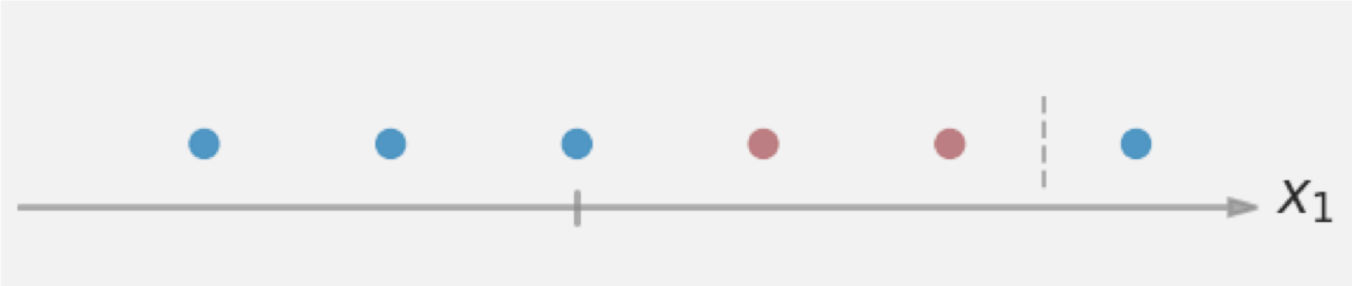
Can you solve this with linear separator?



$x_1^2 + x_2^2$

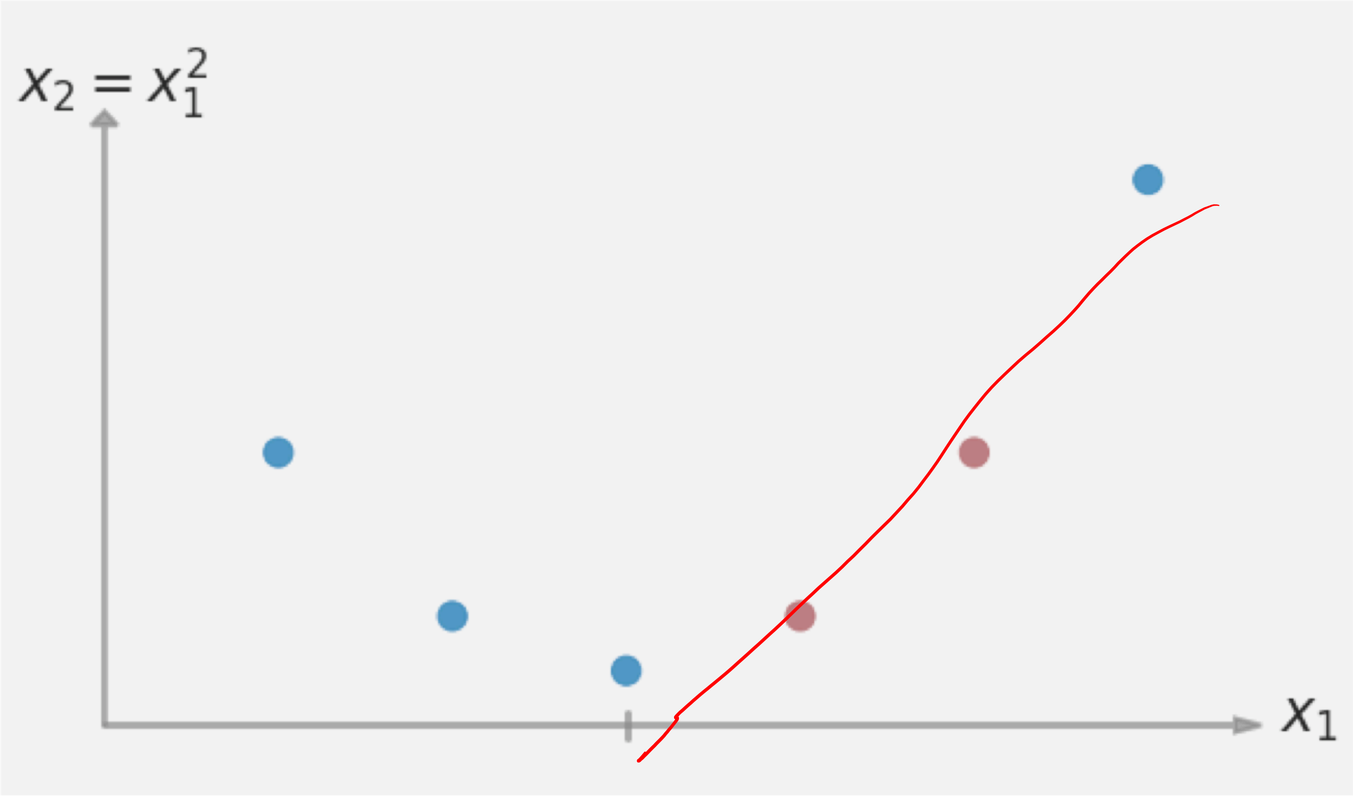
Can you solve this with linear separator?

What can we do if the data is clearly not linearly separable?



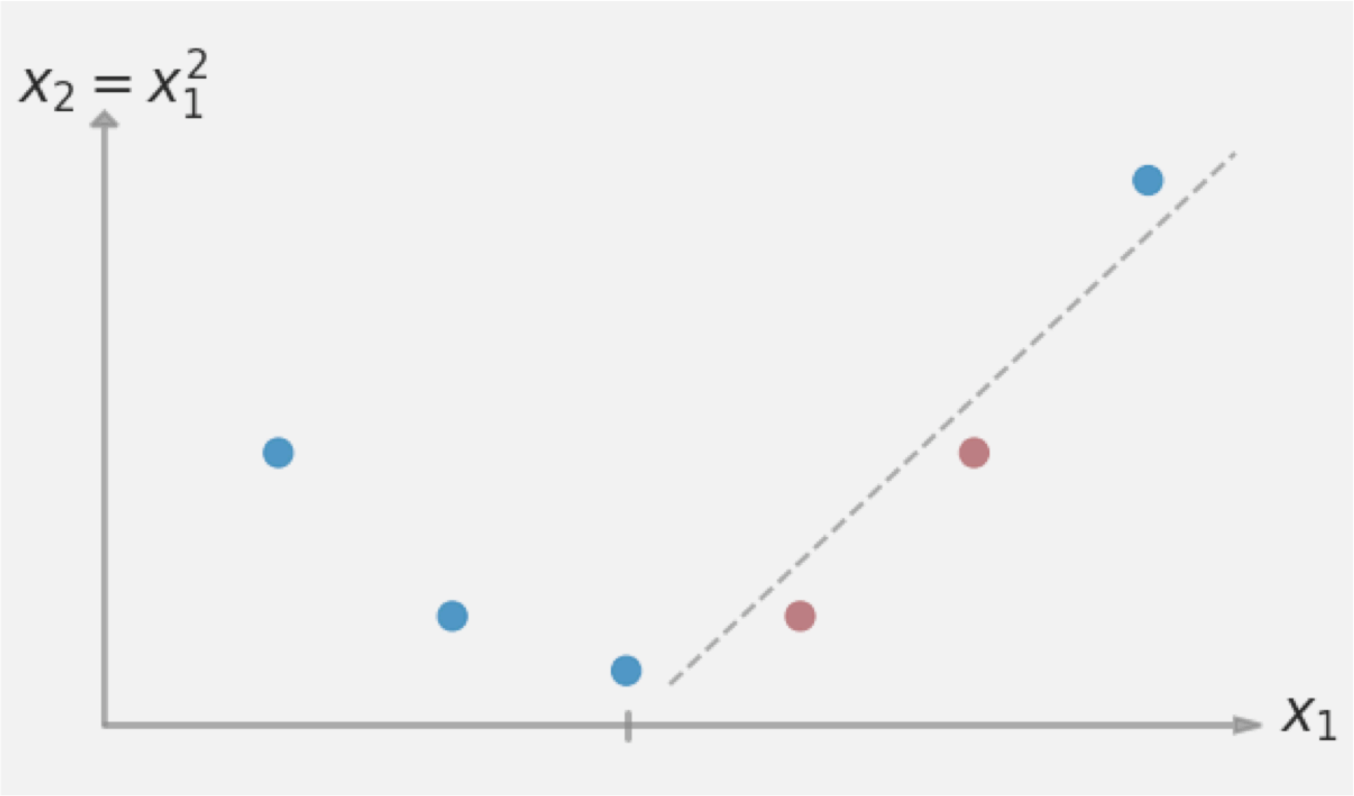
Can you solve this with linear separator?

Add a dimension.



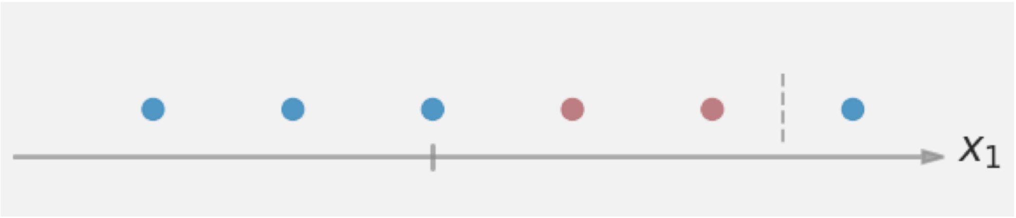
Can you solve this with linear separator?

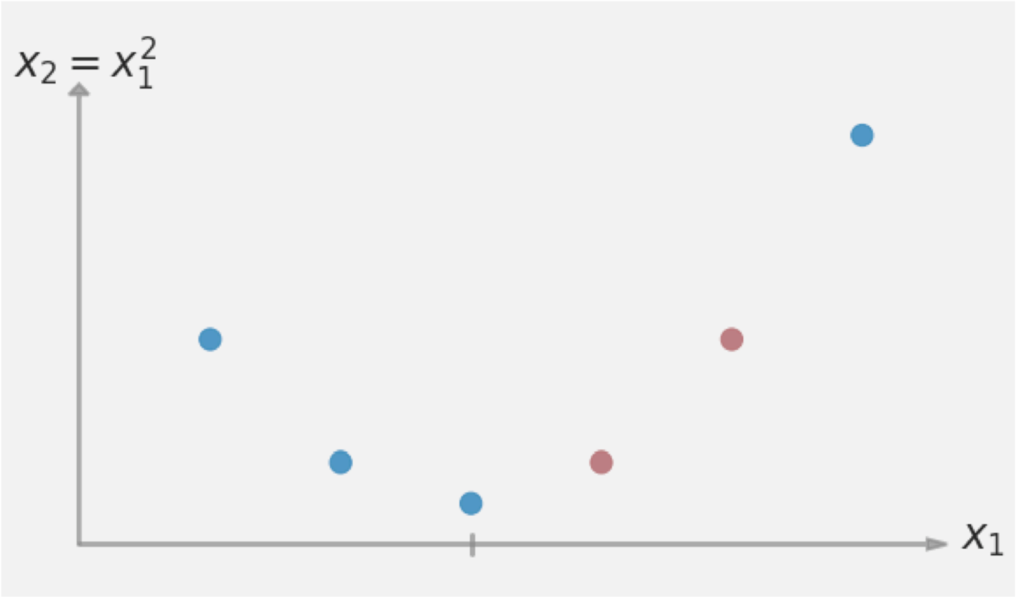
Add a dimension.



Derived features

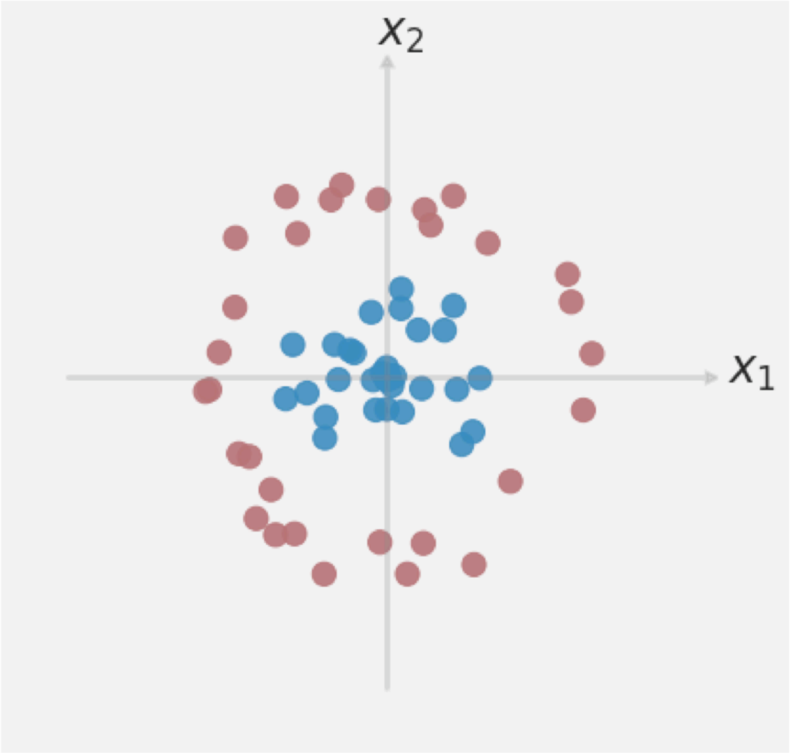
We started with the original feature vector, $\boldsymbol{x} = (x_1)$,
and we created a new derived feature vector, $\phi(\boldsymbol{x}) = (x_1, x_1^2)$.





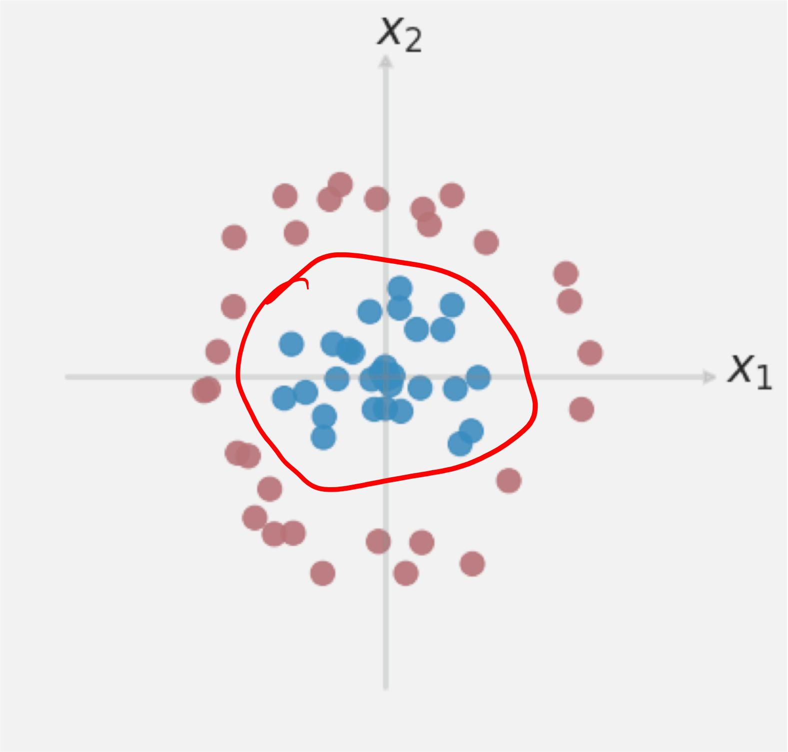
What about the previous problem?

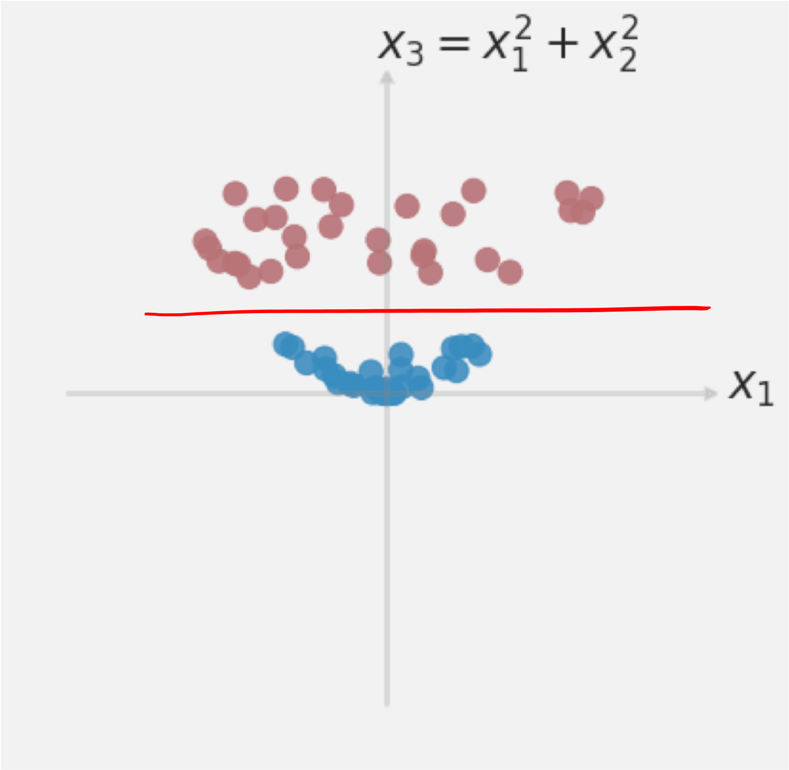
Definitely not separable in two dimensions.



What about the previous problem?

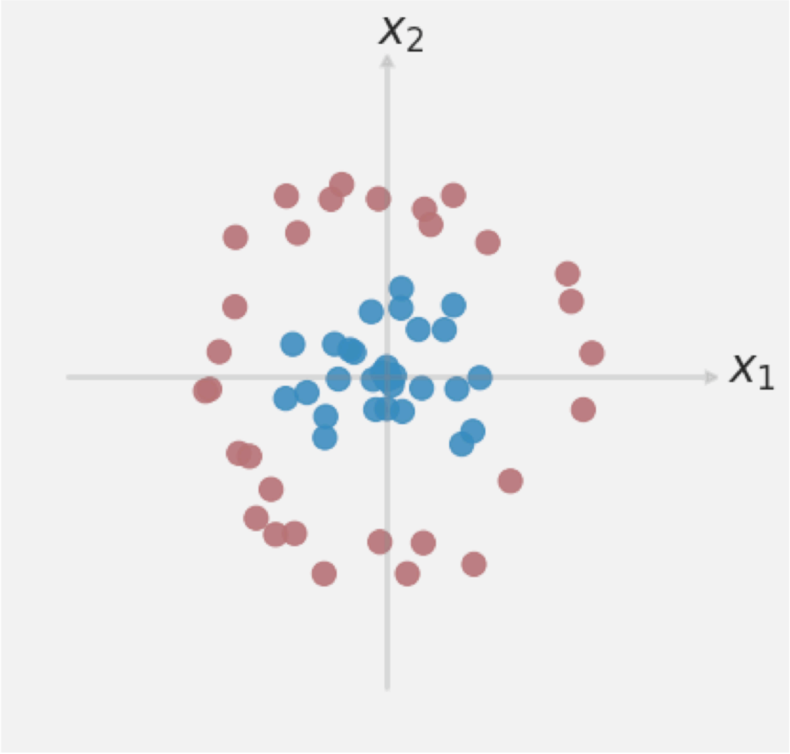
Definitely not separable in two dimensions.
But in three dimensions, it becomes easily separable.

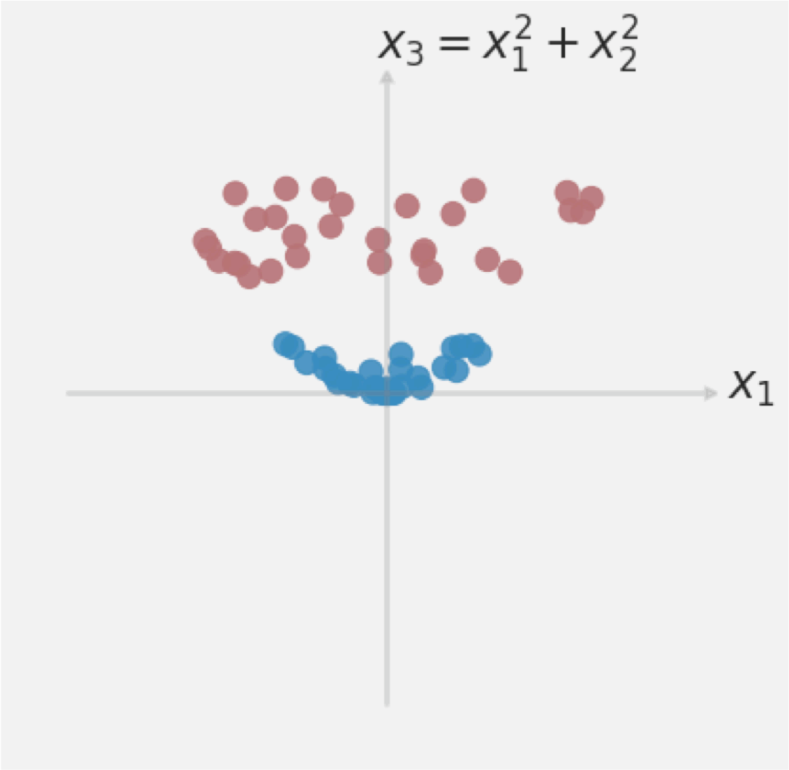




Derived features

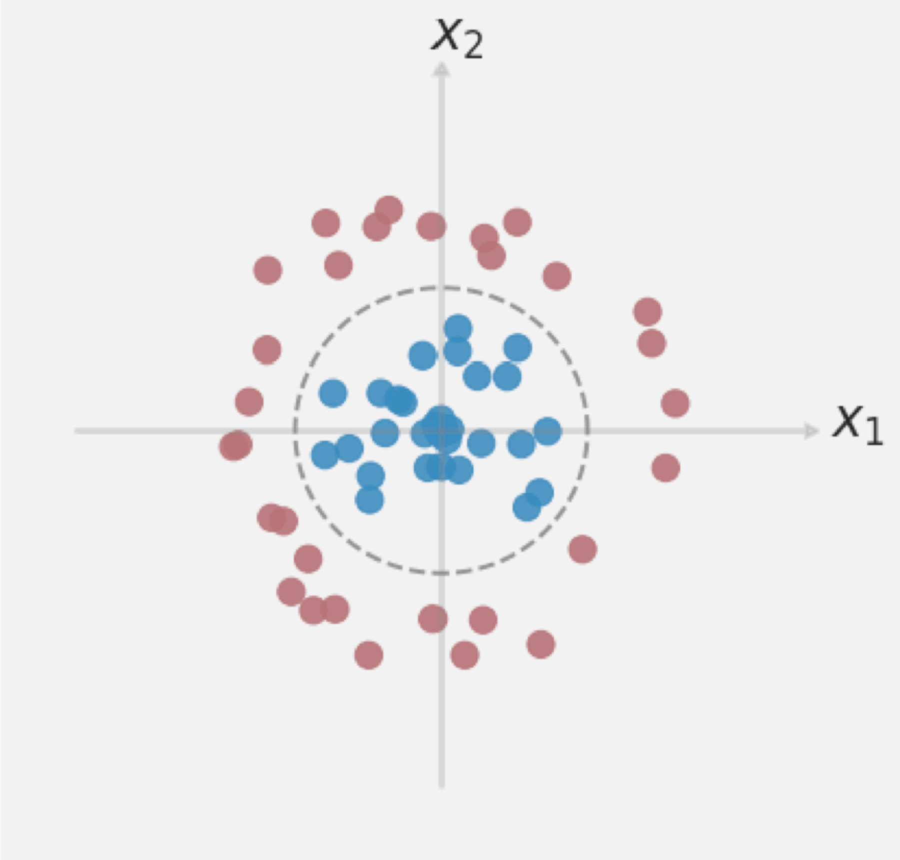
We started with the original feature vector, $\boldsymbol{x} = (x_1, x_2)$,
and we created a new derived feature vector, $\phi(\boldsymbol{x}) = (x_1, x_2, x_1^2 + x_2^2)$.





Derived features

We started with the original feature vector, $\boldsymbol{x} = (x_1, x_2)$,
and we created a new derived feature vector, $\phi(\boldsymbol{x}) = (x_1, x_2, x_1^2 + x_2^2)$.



What's special about SVMs?

$$\max_{\boldsymbol{\alpha}} \sum_{i=1}^{m} \alpha_i - \frac{1}{2} \sum_{i=1}^{m} \sum_{j=1}^{m} \alpha_i \alpha_j y_i y_j \underline{(\boldsymbol{x_i} \cdot \boldsymbol{x_j})}$$

$(\phi(x_i) \cdot \phi(x_j))$

What's special about SVMs?

$$\max_{\boldsymbol{\alpha}} \sum_{i=1}^{m} \alpha_i - \frac{1}{2} \sum_{i=1}^{m} \sum_{j=1}^{m} \alpha_i \alpha_j y_i y_j (\boldsymbol{x_i} \cdot \boldsymbol{x_j})$$

- This dot product is basically just how much x_i looks like x_j. Can we generalize that?

What’s special about SVMs?

$$\max_{\alpha} \sum_{i=1}^{m} \alpha_i - \frac{1}{2} \sum_{i=1}^{m} \sum_{j=1}^{m} \alpha_i \alpha_j y_i y_j (\boldsymbol{x_i} \cdot \boldsymbol{x_j})$$

- This dot product is basically just how much x_i looks like x_j. Can we generalize that?
- Kernels!

What's special about SVMs?

$$\max_{\boldsymbol{\alpha}} \sum_{i=1}^{m} \alpha_i - \frac{1}{2} \sum_{i=1}^{m} \sum_{j=1}^{m} \alpha_i \alpha_j y_i y_j (\boldsymbol{x_i} \cdot \boldsymbol{x_j})$$

$$\max_{\boldsymbol{\alpha}} \sum_{i=1}^{m} \alpha_i - \frac{1}{2} \sum_{i=1}^{m} \sum_{j=1}^{m} \alpha_i \alpha_j y_i y_j (\phi(\boldsymbol{x_i}) \cdot \phi(\boldsymbol{x_j}))$$

$$\max_{\boldsymbol{\alpha}} \sum_{i=1}^{m} \alpha_i - \frac{1}{2} \sum_{i=1}^{m} \sum_{j=1}^{m} \alpha_i \alpha_j y_i y_j K(\boldsymbol{x_i} \cdot \boldsymbol{x_j})$$

What does the kernel trick buy us?

Polynomial kernel:

$$K(\mathbf{x}, \mathbf{x}') = (\mathbf{x} \cdot \mathbf{x}' + c)^d$$

$\phi(x) \cdot \phi(x')$

$x = (x_1, x_2)$

$\phi(x)$

$(x \cdot x' + 1)^2$

$\phi(x) = (x_1, x_2, 1)$

$= (x_1 x_1' + x_2 x_2' + 1)^2$

$= x_1^2 x_1'^2 + x_2^2 x_2'^2 + 2 x_1 x_1' x_2 x_2' + 2 x_1 x_1' + 2 x_2 x_2' + 1$

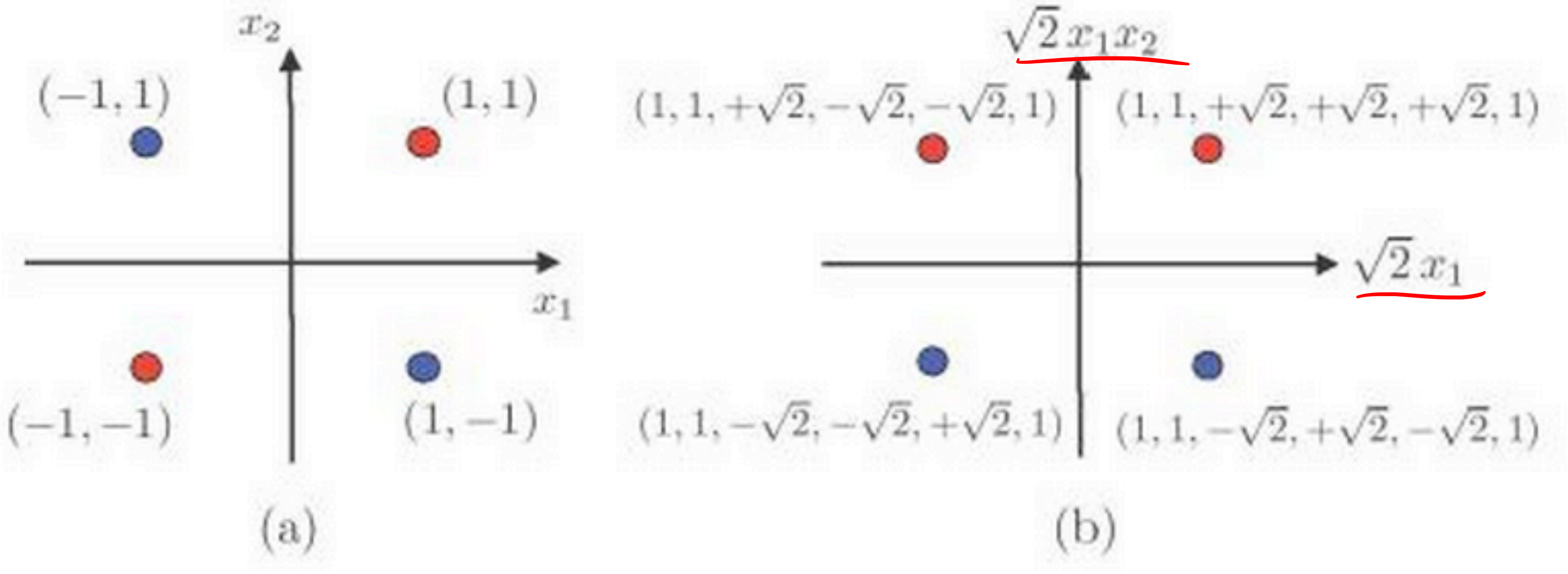
$\phi(x) = (x_1^2, x_2^2, \sqrt{2} x_1 x_2, \sqrt{2} x_1, \sqrt{2} x_2, 1)$

What does the kernel trick buy us?

Polynomial kernel:

$$K(\boldsymbol{x}, \boldsymbol{x}') = (\boldsymbol{x} \cdot \boldsymbol{x}' + c)^d$$

When $d = 2, c = 1$:



What does the kernel trick buy us?

Polynomial kernel:

$$K(\boldsymbol{x}, \boldsymbol{x}') = (\boldsymbol{x} \cdot \boldsymbol{x}' + 1)^2$$

What is the corresponding $\phi(\boldsymbol{x})$, where $\boldsymbol{x} \in \mathbb{R}^k$?
What is the complexity of storing $\phi(\boldsymbol{x})$ and computing $\phi(\boldsymbol{x}) \cdot \phi(\boldsymbol{x}')$?
What about using the kernel function?

What's a kernel?

- A function $K : \mathcal{X} \times \mathcal{X} \mapsto \mathrm{R}$ is a kernel over $\mathcal{X}$.
- This is equivalent to taking the dot product $\langle \phi(x_1), \phi(x_2) \rangle$ for some mapping
- **Mercer's Theorem**: So long as the function is continuous and symmetric, then K admits an expansion of the form

$$K(x, x') = \sum_{n=0}^{\infty} a_n \phi_n(x) \phi_n(x')$$

What's a kernel?

- A function $K : \mathcal{X} \times \mathcal{X} \mapsto \mathrm{R}$ is a kernel over $\mathcal{X}$.
- This is equivalent to taking the dot product $\langle \phi(x_1), \phi(x_2) \rangle$ for some mapping
- **Mercer's Theorem**: So long as the function is continuous and symmetric, then K admits an expansion of the form

$$K(x, x') = \sum_{n=0}^{\infty} a_n \phi_n(x) \phi_n(x')$$

- The computational cost is just in computing the kernel

Kernel Matrix

The important property of the kernel matrix $\boldsymbol{K} = [K(x_i, x_j)]_{ij} \in \mathbb{R}^{m \times m}$ is symmetric positive semidefinite.

Kernel Matrix

The important property of the kernel matrix $\boldsymbol{K} = [K(x_i, x_j)]_{ij} \in \mathbb{R}^{m \times m}$ is symmetric positive semidefinite.

$$\boldsymbol{K}^T = \boldsymbol{K}$$

Kernel Matrix

The important property of the kernel matrix $\boldsymbol{K} = [K(x_i, x_j)]_{ij} \in \mathbb{R}^{m \times m}$ is symmetric positive semidefinite.

$$\boldsymbol{K}^T = \boldsymbol{K}$$

$$\underline{\forall \boldsymbol{x}, \boldsymbol{x}^T \boldsymbol{K} \boldsymbol{x} \geq 0}$$

ΣK_i

Kernel Matrix

The important property of the kernel matrix $\boldsymbol{K} = [K(x_i, x_j)]_{ij} \in \mathbb{R}^{m \times m}$ is symmetric positive semidefinite.

$$\boldsymbol{K}^T = \boldsymbol{K}$$

$$\forall \boldsymbol{x}, \boldsymbol{x}^T \boldsymbol{K} \boldsymbol{x} \geq 0$$

Also known as Gram matrix.

Gaussian Kernel

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

Gaussian Kernel

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

which can be rewritten as

$$K(x,x') = \sum_n \frac{(x \cdot x')^n}{\sigma^n n!}$$

(All polynomials!)

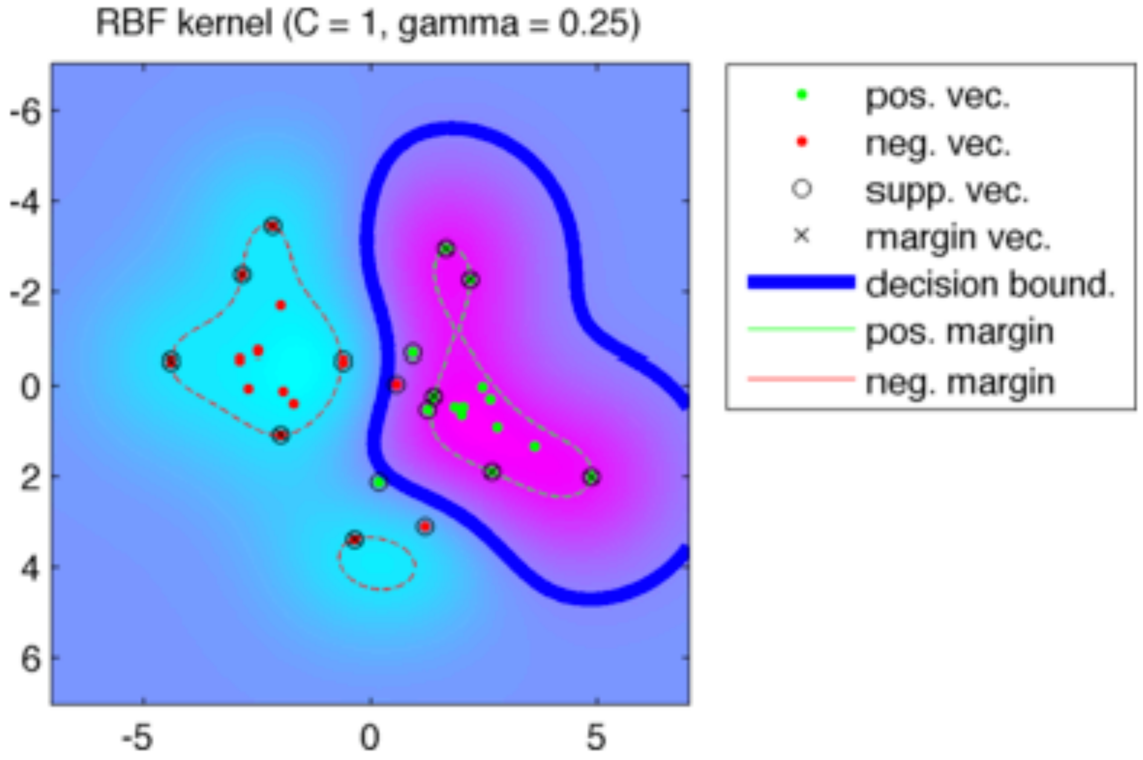
Gaussian Kernel

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

which can be rewritten as

$$K(x,x') = \sum_n \frac{(x \cdot x')^n}{\sigma^n n!}$$

(All polynomials!)



How does it affect optimization

$$\max_{\boldsymbol{\alpha}} \sum_{i=1}^{m} \alpha_i - \frac{1}{2} \sum_{i=1}^{m} \sum_{i=1}^{m} \alpha_i \alpha_j y_i y_j (\boldsymbol{x_i} \cdot \boldsymbol{x}_j)$$

$$\max_{\boldsymbol{\alpha}} \sum_{i=1}^{m} \alpha_i - \frac{1}{2} \sum_{i=1}^{m} \sum_{i=1}^{m} \alpha_i \alpha_j y_i y_j K(\boldsymbol{x_i}, \boldsymbol{x}_j)$$

- Replace all dot product with kernel evaluations $K(x_1, x_2)$
- Makes computation more expensive, overall structure is the same

Examples

- Switch to notebooks

Recap

- This completes our discussion of SVMs
- Workhorse method of machine learning
- Flexible, fast, effective