Lecture 14: Support Vector Machine (SVM)

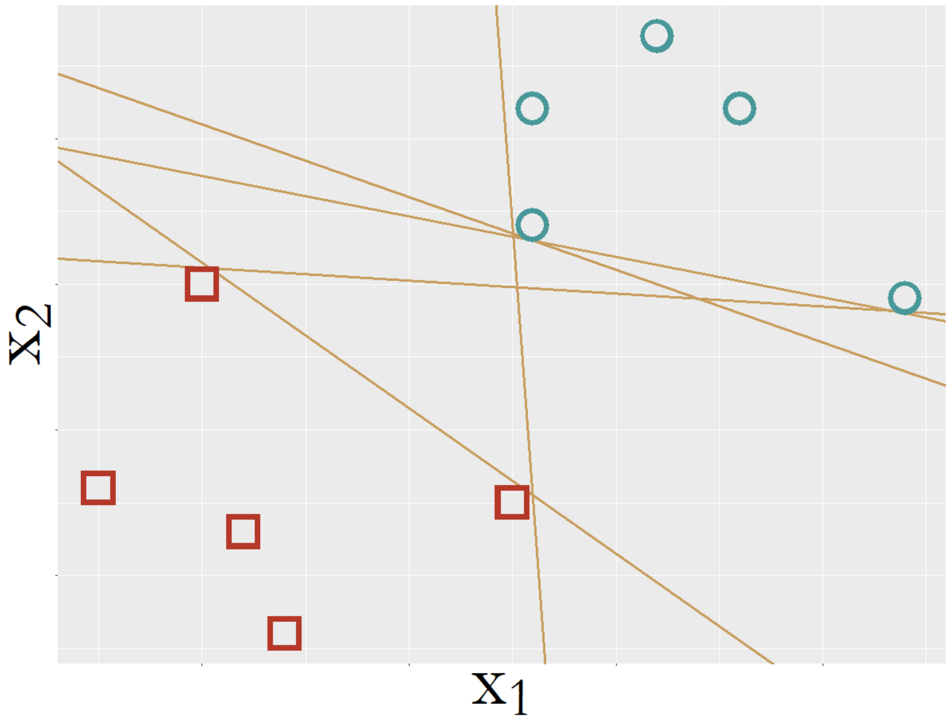
Instructor: Prof. Shuai Huang

Industrial and Systems Engineering

University of Washington

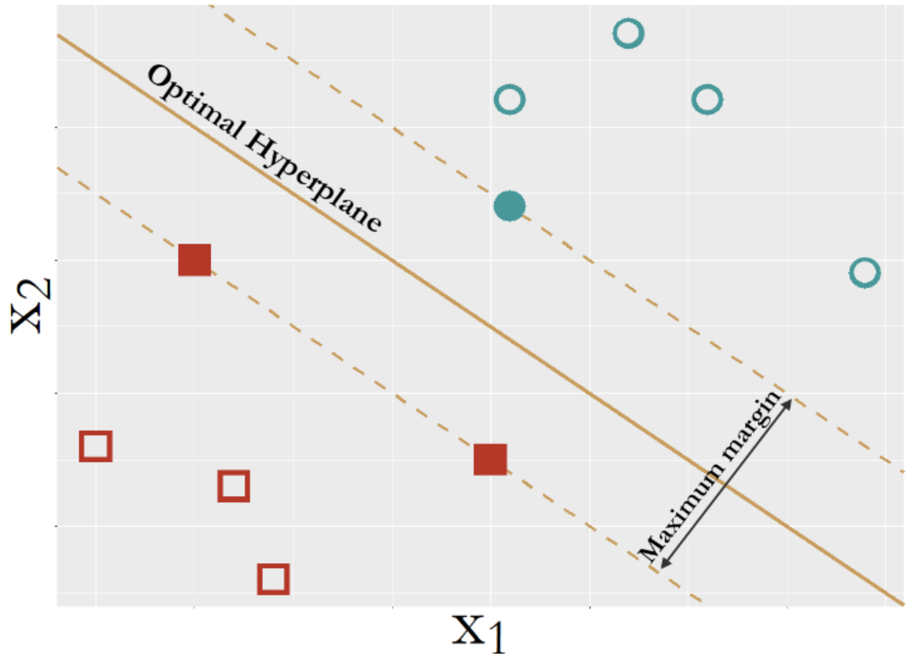
What ambiguity the SVM ties to solve

- Which model should we use?



The model with maximum margin

- SVM is essentially a preference over models that have maximum margin



Can this idea lead to mathematic tractability?

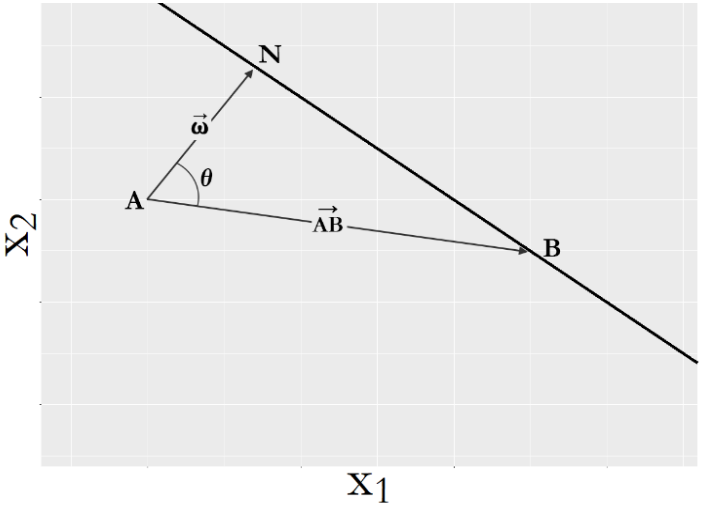
- The goal is to identify a model, $\boldsymbol{w}^T\boldsymbol{x} + b$, using which we can make binary classification:

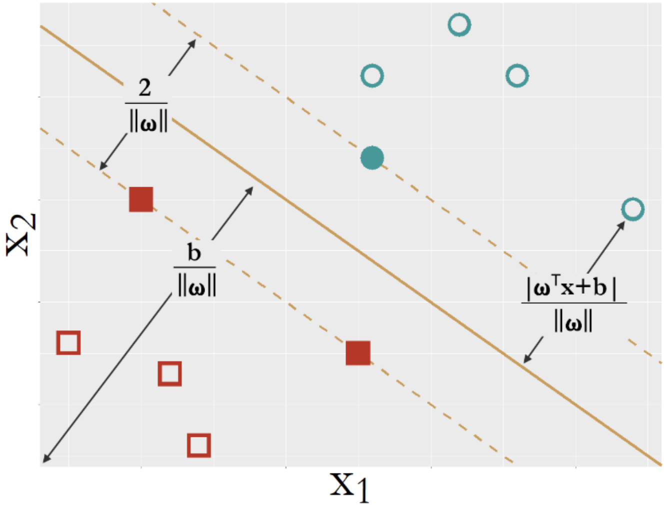
  If $\boldsymbol{w}^T\boldsymbol{x} + b > 0$, then $y = 1$; Otherwise, $y = -1$.

- The final SVM formulation is:

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

Subject to: $y_n(\boldsymbol{w}^T\boldsymbol{x}_n + b) \geq 1$ for $n = 1,2, \dots, N$.





Solve for SVM

- To solve this problem, first, we can use the method of Lagrange multiplier:
$$L(\boldsymbol{w}, b, \boldsymbol{\alpha}) = \frac{1}{2}\|\boldsymbol{w}\| - \sum_{n=1}^{N} \alpha_n \left[y_n\left(\boldsymbol{w}^T \boldsymbol{x}_n + b\right) - 1\right].$$
- This could be rewritten as
$$L(\boldsymbol{w}, b, \boldsymbol{\alpha}) = \frac{1}{2}\boldsymbol{w}^T \boldsymbol{w} - \sum_{n=1}^{N} \alpha_n y_n \boldsymbol{w}^T \boldsymbol{x}_n - b\sum_{n=1}^{N} \alpha_n y_n + \sum_{n=1}^{N} \alpha_n.$$
- Differentiating $L(\boldsymbol{w}, b, \boldsymbol{\alpha})$ with respect to $\boldsymbol{w}$ and b, and setting to zero yields:
$$\boldsymbol{w} = \sum_{n=1}^{N} \alpha_n y_n \boldsymbol{x}_n, \;\; \sum_{n=1}^{N} \alpha_n y_n = 0.$$
- Then, we can rewrite $L(\boldsymbol{w}, b, \boldsymbol{\alpha})$ as
$$L(\boldsymbol{w}, b, \boldsymbol{\alpha}) = \sum_{n=1}^{N} \alpha_n - \frac{1}{2}\sum_{n=1}^{N}\sum_{m=1}^{N} \alpha_n \alpha_m y_n y_m \boldsymbol{x}_n^T \boldsymbol{x}_m.$$
- This is because that:
$$\frac{1}{2}\boldsymbol{w}^T \boldsymbol{w} = \frac{1}{2}\boldsymbol{w}^T \sum_{n=1}^{N} \alpha_n y_n \boldsymbol{x}_n = \frac{1}{2}\sum_{n=1}^{N} \alpha_n y_n \boldsymbol{w}^T \boldsymbol{x}_n =$$
$$\frac{1}{2}\sum_{n=1}^{N} \alpha_n y_n \left(\sum_{n=1}^{N} \alpha_n y_n \boldsymbol{x}_n\right)^T \boldsymbol{x}_n = \frac{1}{2}\sum_{n=1}^{N}\sum_{m=1}^{N} \alpha_n \alpha_m y_n y_m \boldsymbol{x}_n^T \boldsymbol{x}_m.$$

The dual form of SVM

- Finally, we can derive the model of SVM by solving its dual form problem:

$$\max_{\boldsymbol{\alpha}} \sum_{n=1}^{N} \alpha_n - \frac{1}{2} \sum_{n=1}^{N} \sum_{m=1}^{N} \alpha_n \alpha_m y_n y_m \boldsymbol{x}_n^T \boldsymbol{x}_m,$$

Subject to: $\alpha_n \geq 0$ for $n = 1,2, \ldots, N$ and $\sum_{n=1}^{N} \alpha_n y_n = 0.$

- This is a **quadratic programming** problem that can be solved using many existing packages.

The support points

- The learned model parameters could be represented as:

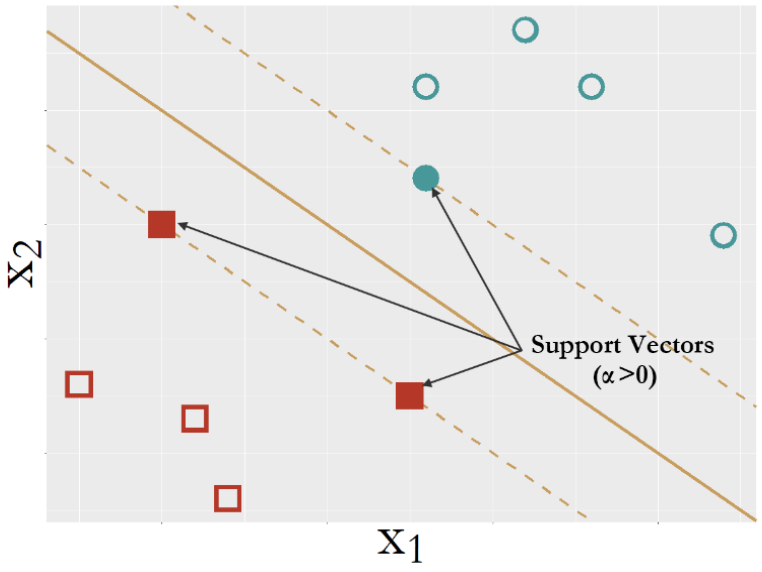
$$\hat{\boldsymbol{w}} = \sum_{n=1}^{N} \alpha_n y_n \boldsymbol{x}_n \text{ and } \hat{b} = 1 - \hat{\boldsymbol{w}}^T \boldsymbol{x}_n \text{ for any } \boldsymbol{x}_n \text{ whose } \alpha_n > 0.$$

- And we know that, based on the KKT condition:

$$\alpha_n \left[y_n \left(\boldsymbol{w}^T \boldsymbol{x}_n + b \right) - 1 \right] = 0 \text{ for } n = 1,2, \dots, N.$$

- Thus, for any data point, e.g., the nth data point, it is either

$$\alpha_n = 0 \text{ or } y_n \left(\boldsymbol{w}^T \boldsymbol{x}_n + b \right) - 1 = 0.$$

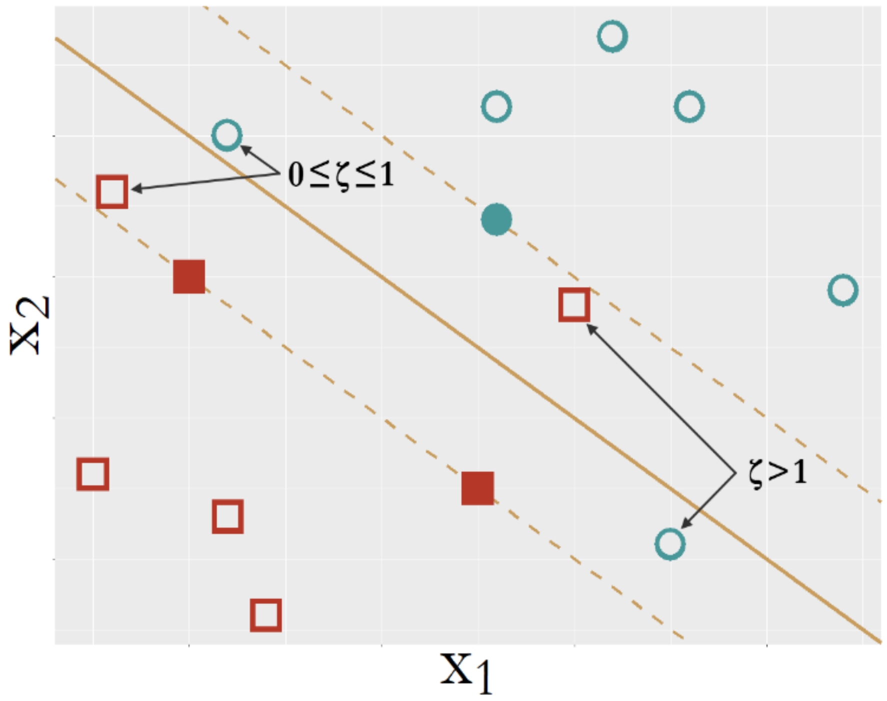


Extension to non-separable cases

- Introduce the slack variables:

  $$y_n(\boldsymbol{w}^T\boldsymbol{x}_n + b) \geq 1 - \xi_n \text{ for } n = 1,2,\dots,N.$$

- The data points that are within the margins will have the corresponding slack variables as $0 \leq \xi_n \leq 1$
- The data points that are on the wrong side of the decision line have the corresponding slack variables as $\xi_n > 1$.



The revised SVM formulation

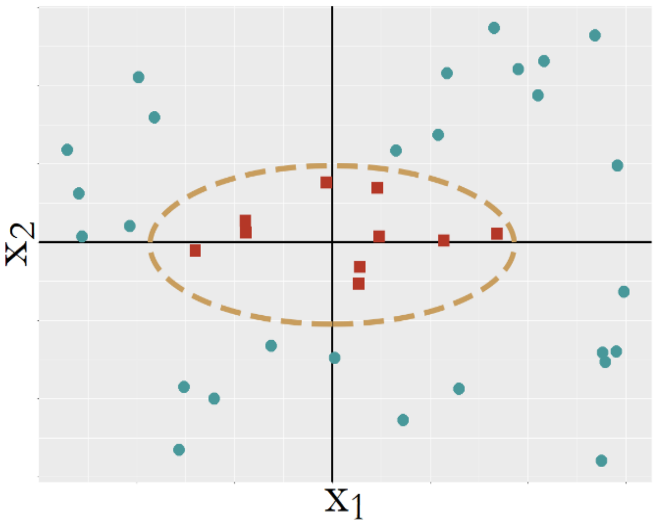
- The corresponding formulation of the SVM model becomes:

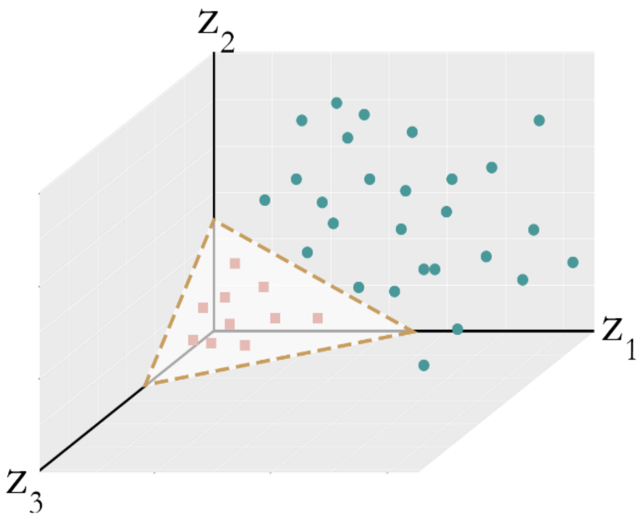
$$\min_{\boldsymbol{w}} \frac{1}{2} \|\boldsymbol{w}\| + C \sum_{n=1}^{N} \xi_n,$$

Subject to: $y_n\left(\boldsymbol{w}^T \boldsymbol{x}_n + b\right) \geq 1 - \xi_n$ and $\xi_n \geq 0$, for $n = 1,2, \ldots, N$.

Extension to nonlinear cases

- Main idea: transformation from $\boldsymbol{x}$ to $\boldsymbol{z}$
- An example: $z_1 = x_1^2, z_2 = \sqrt{2}x_1x_2, z_3 = x_2^2.$
- But not in all times the transformations can be made explicit





Assume the transformation exists

- The dual formulation of SVM on the transformed variables is:

$$\max_{\boldsymbol{\alpha}} \sum_{n=1}^{N} \alpha_n - \frac{1}{2} \sum_{n=1}^{N} \sum_{m=1}^{N} \alpha_n \alpha_m y_n y_m \boldsymbol{z}_n^T \boldsymbol{z}_m,$$

Subject to: $0 \leq \alpha_n \leq C$ for $n = 1,2, \ldots, N$ and $\sum_{n=1}^{N} \alpha_n y_n = 0$.

- What matters here is really the inner product of the transformed vectors
- Thus, we can write it up as $\boldsymbol{z}_n^T \boldsymbol{z}_m = K(\boldsymbol{x}_n, \boldsymbol{x}_m)$. This is called the "**kernel function**". A kernel function is a function that theoretically entails a transformation $\boldsymbol{z} = \phi(\boldsymbol{x})$ such that $K(\boldsymbol{x}_n, \boldsymbol{x}_m)$ implies that it can be written as an inner product $K(\boldsymbol{x}_n, \boldsymbol{x}_m) = \phi(\boldsymbol{x})^T \phi(\boldsymbol{x})$.

The revised SVM formulation

- With a given kernel function, SVM learns the model by solving the following optimization problem:

$$\max_{\boldsymbol{\alpha}} \sum_{n=1}^{N} \alpha_n - \frac{1}{2} \sum_{n=1}^{N} \sum_{m=1}^{N} \alpha_n \alpha_m y_n y_m K(\boldsymbol{x}_n, \boldsymbol{x}_m),$$

Subject to: $0 \leq \alpha_n \leq C$ for $n = 1,2, \dots, N$ and $\sum_{n=1}^{N} \alpha_n y_n = 0$.

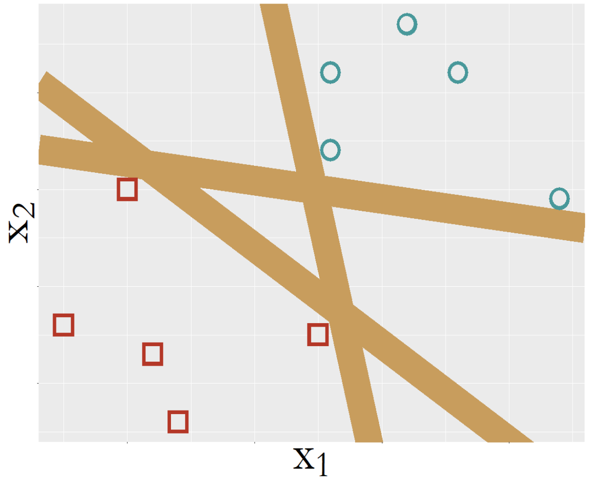
- However, in the kernel space, it will no longer to possible to write up the parameter $\boldsymbol{w}$ the same way as in linear models.
- For any new data point, denoted as $\boldsymbol{x}_*$, the learned SVM model predict on it as

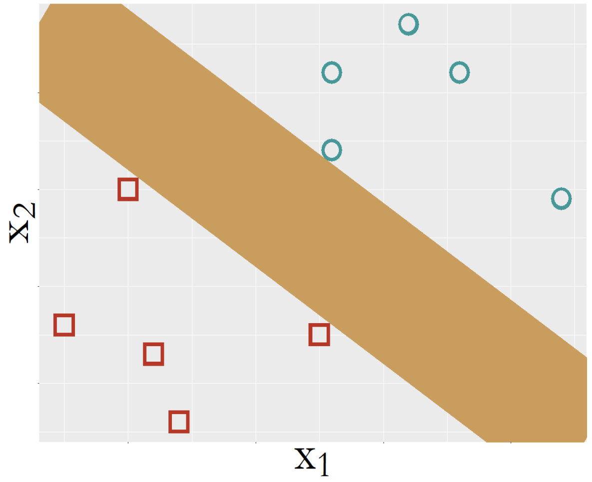
If $\sum_{n=1}^{N} \alpha_n y_n K(\boldsymbol{x}_n, \boldsymbol{x}_*) + b > 0$, then $y = 1$;

Otherwise, $y = -1$.

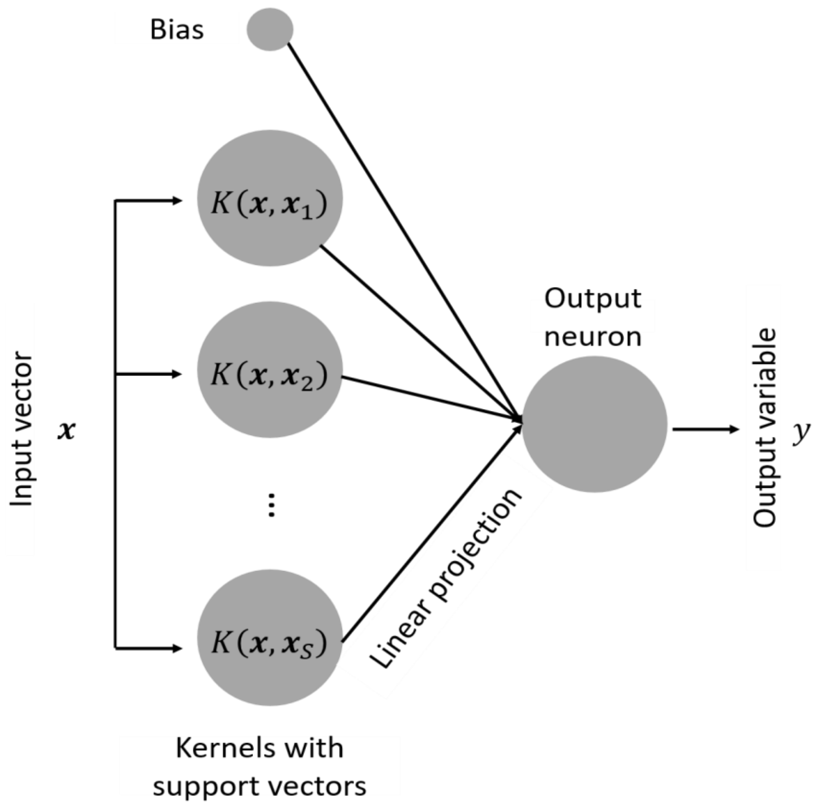
Is SVM a more complex model?

- In statistical learning theory, a more complex model has larger VC-dimension. In intuitive language, that means, a more complex model has more mathematical capacity to encode a richer signal. Thus, it could be very flexible and sensitive to data distributions
- However, for SVM …





SVM is a neural network model



R lab

- Download the markdown code from course website
- Conduct the experiments
- Interpret the results
- Repeat the analysis on other datasets