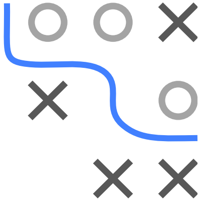
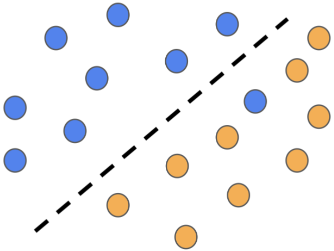
# Introduction to Machine Learning



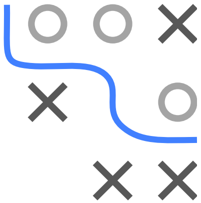
## Classification
## Linear Classifiers



**Learning goals**

- Linear classifier
- Linear decision boundaries
- Linear separability

# LINEAR CLASSIFIERS



Important subclass of classification models.

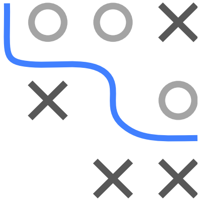
Definition: If discriminant(s) $f_k(\mathbf{x})$ can be written as affine linear function(s) (possibly through a rank-preserving, monotone transformation $g$):

$$g(f_k(\mathbf{x})) = \boldsymbol{w}_k^\top \mathbf{x} + b_k,$$

we will call the classifier **linear**.

- $\boldsymbol{w}_k$ and $b_k$ do not necessarily refer to parameters $\boldsymbol{\theta}_k$, although they often coincide; discriminant simply must be writable in an affine-linear way
- reasons for the transformation is that we only care about the position of the decision boundary

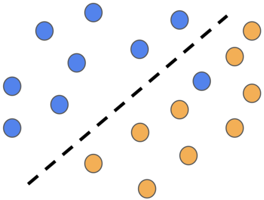
# LINEAR DECISION BOUNDARIES



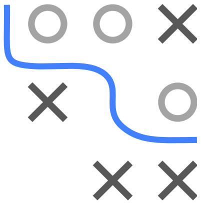
We can also easily show that the decision boundary between classes $i$ and $j$ is a hyperplane. For every $\mathbf{x}$ where there is a tie in scores:

$$\begin{aligned} f_i(\mathbf{x}) &= f_j(\mathbf{x}) \\ g(f_i(\mathbf{x})) &= g(f_j(\mathbf{x})) \\ \boldsymbol{w}_i^\top \mathbf{x} + b_i &= \boldsymbol{w}_j^\top \mathbf{x} + b_j \\ (\boldsymbol{w}_i - \boldsymbol{w}_j)^\top \mathbf{x} + (b_i - b_j) &= 0 \end{aligned}$$

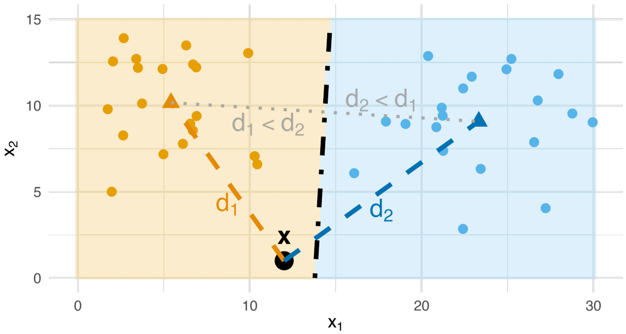
This represents a **hyperplane** separating two classes:



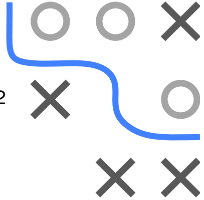
# EXAMPLE: 2 CLASSES WITH CENTROIDS



- Model binary problem with centroid $\mu_k$ per class as "parameters"
- Don’t really care how the centroids are estimated;
  could use class means, but the following doesn’t depend on it
- Classify point **x** by assigning it to class $k$ of nearest centroid



# EXAMPLE: 2 CLASSES WITH CENTROIDS



Let's calculate the decision boundary:

$$d_1 = ||\mathbf{x} - \boldsymbol{\mu}_1||^2 = \mathbf{x}^\top \mathbf{x} - 2\mathbf{x}^\top \boldsymbol{\mu}_1 + \boldsymbol{\mu}_1^\top \boldsymbol{\mu}_1 = \mathbf{x}^\top \mathbf{x} - 2\mathbf{x}^\top \boldsymbol{\mu}_2 + \boldsymbol{\mu}_2^\top \boldsymbol{\mu}_2 = ||\mathbf{x} - \boldsymbol{\mu}_2||^2 = d_2$$

Where $d$ is measured using Euclidean distance. This implies:

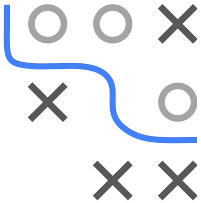
$$-2\mathbf{x}^\top \boldsymbol{\mu}_1 + \boldsymbol{\mu}_1^\top \boldsymbol{\mu}_1 = -2\mathbf{x}^\top \boldsymbol{\mu}_2 + \boldsymbol{\mu}_2^\top \boldsymbol{\mu}_2$$

Which simplifies to:

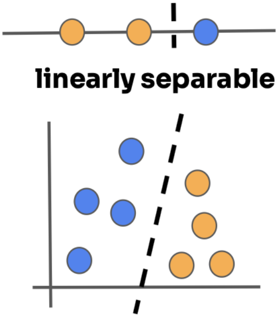
$$2\mathbf{x}^\top (\boldsymbol{\mu}_2 - \boldsymbol{\mu}_1) = \boldsymbol{\mu}_2^\top \boldsymbol{\mu}_2 - \boldsymbol{\mu}_1^\top \boldsymbol{\mu}_1$$

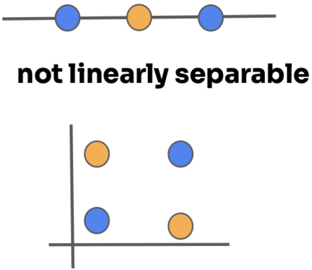
Thus, it's a linear classifier!

# LINEAR SEPARABILITY

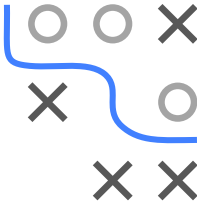


If there exists a linear classifier that perfectly separates the classes of some dataset, the data are called **linearly separable**.

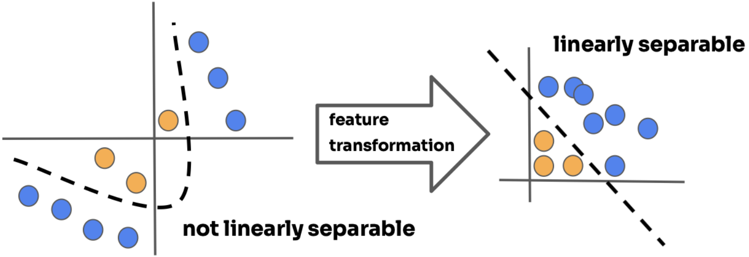


# FEATURE TRANSFORMATIONS



Note that linear classifiers can represent **non-linear** decision boundaries in the original input space if we use derived features like higher order interactions, polynomial features, etc.




Here we used absolute values to find suitable derived features.