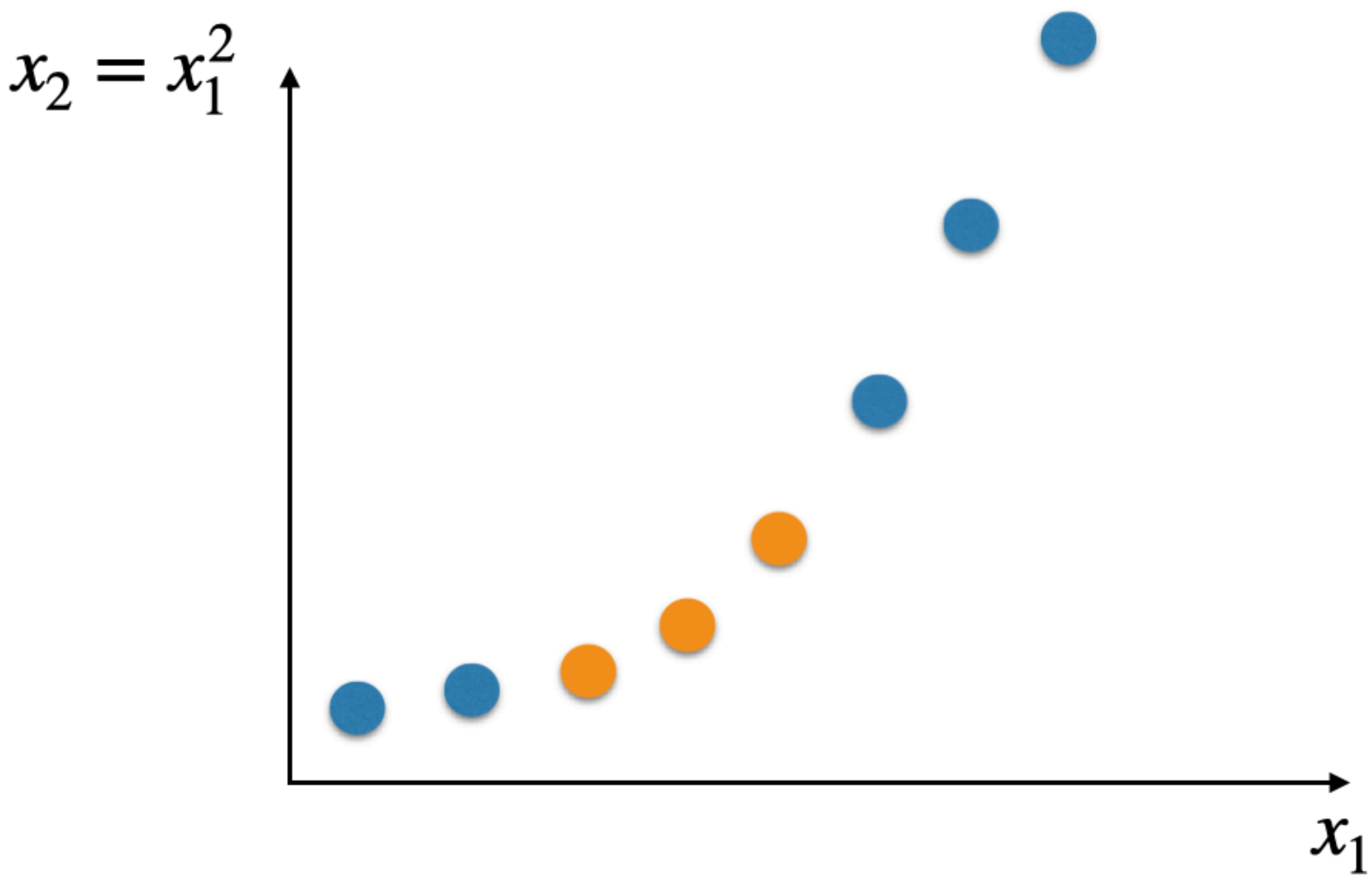


Nonlinear Transformations

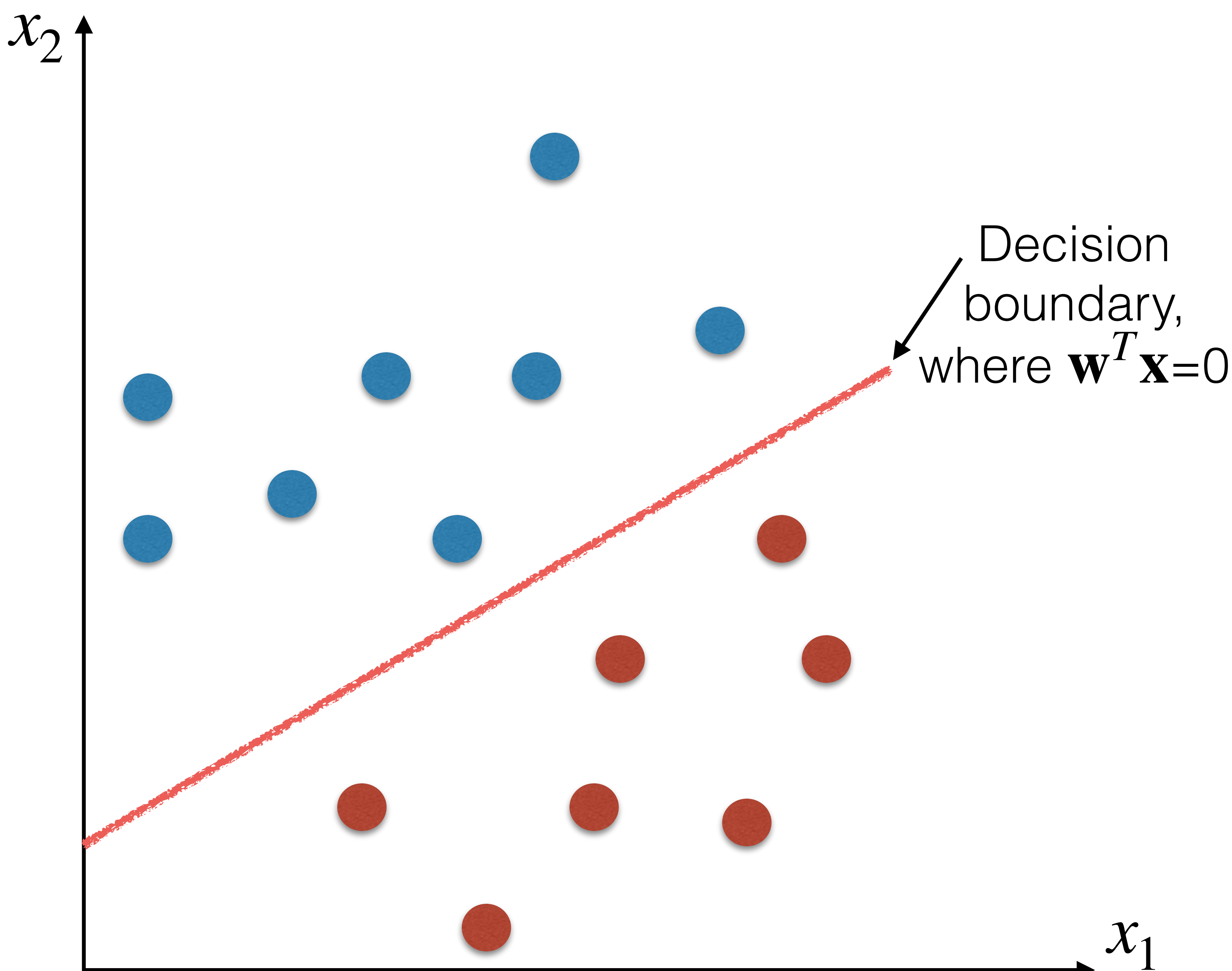
Leandro L. Minku

Outline

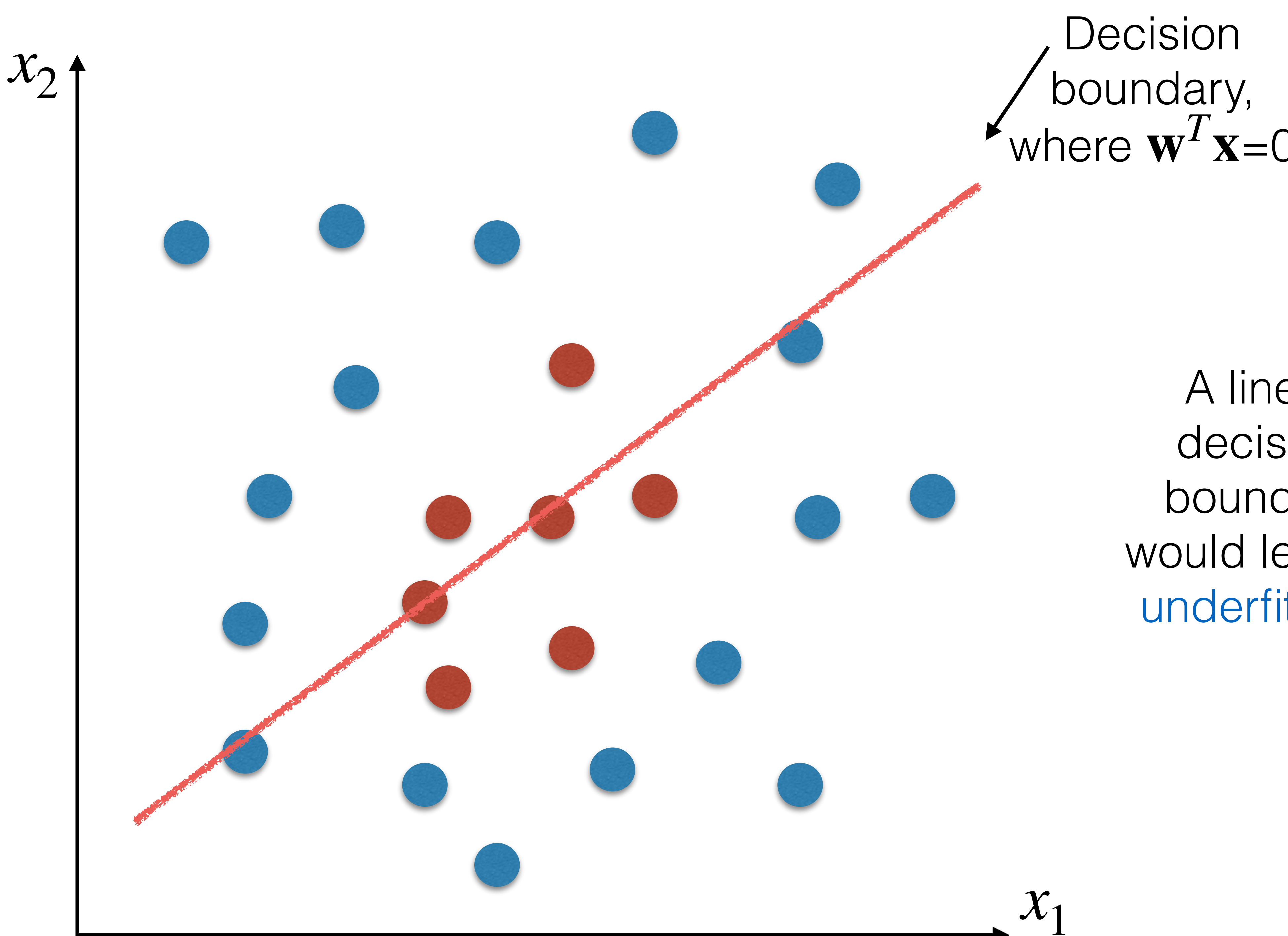
- The need for nonlinear transformations
- Intuition behind nonlinear transformations
- Adopting nonlinear transformations
- Advantages and potential caveats of nonlinear transformations

Linearly Separable Problems

$\text{logit}(p_1) = \mathbf{w}^T \mathbf{x}$

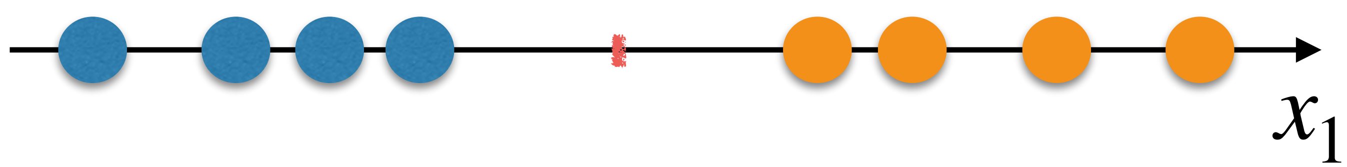


Nonlinearly Separable Problems

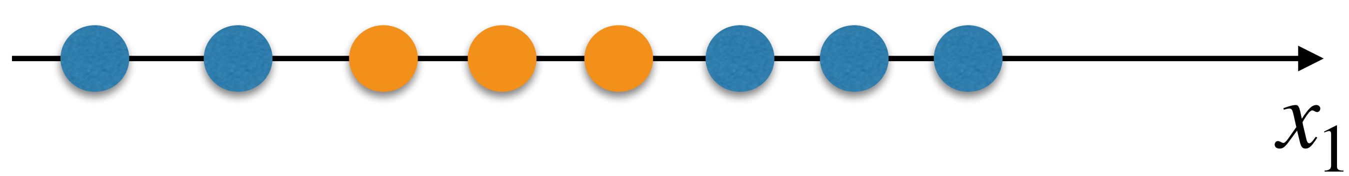


A linear decision boundary would lead to underfitting.

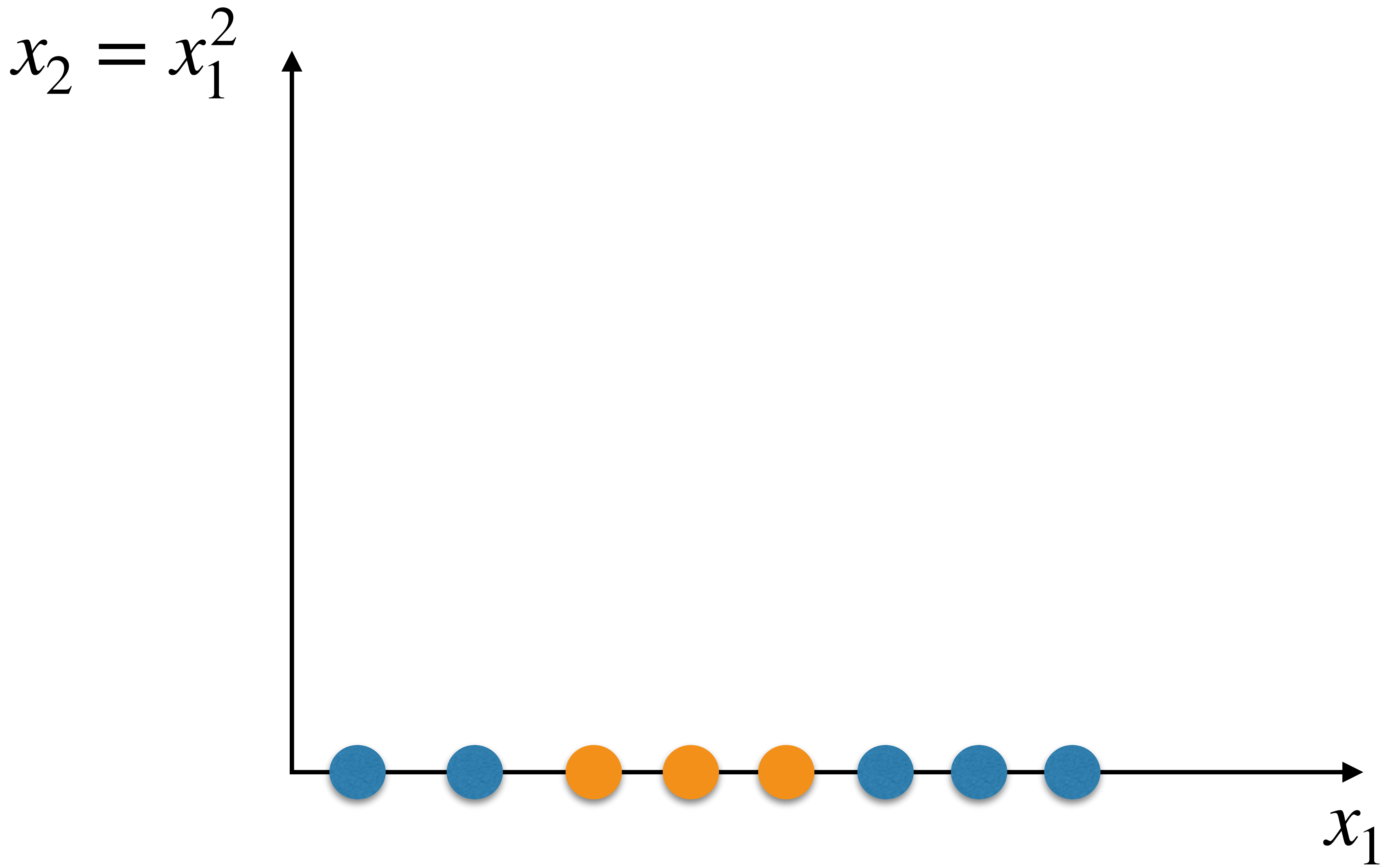
Linearly Separable Problems



Non-Linearly Separable Problems

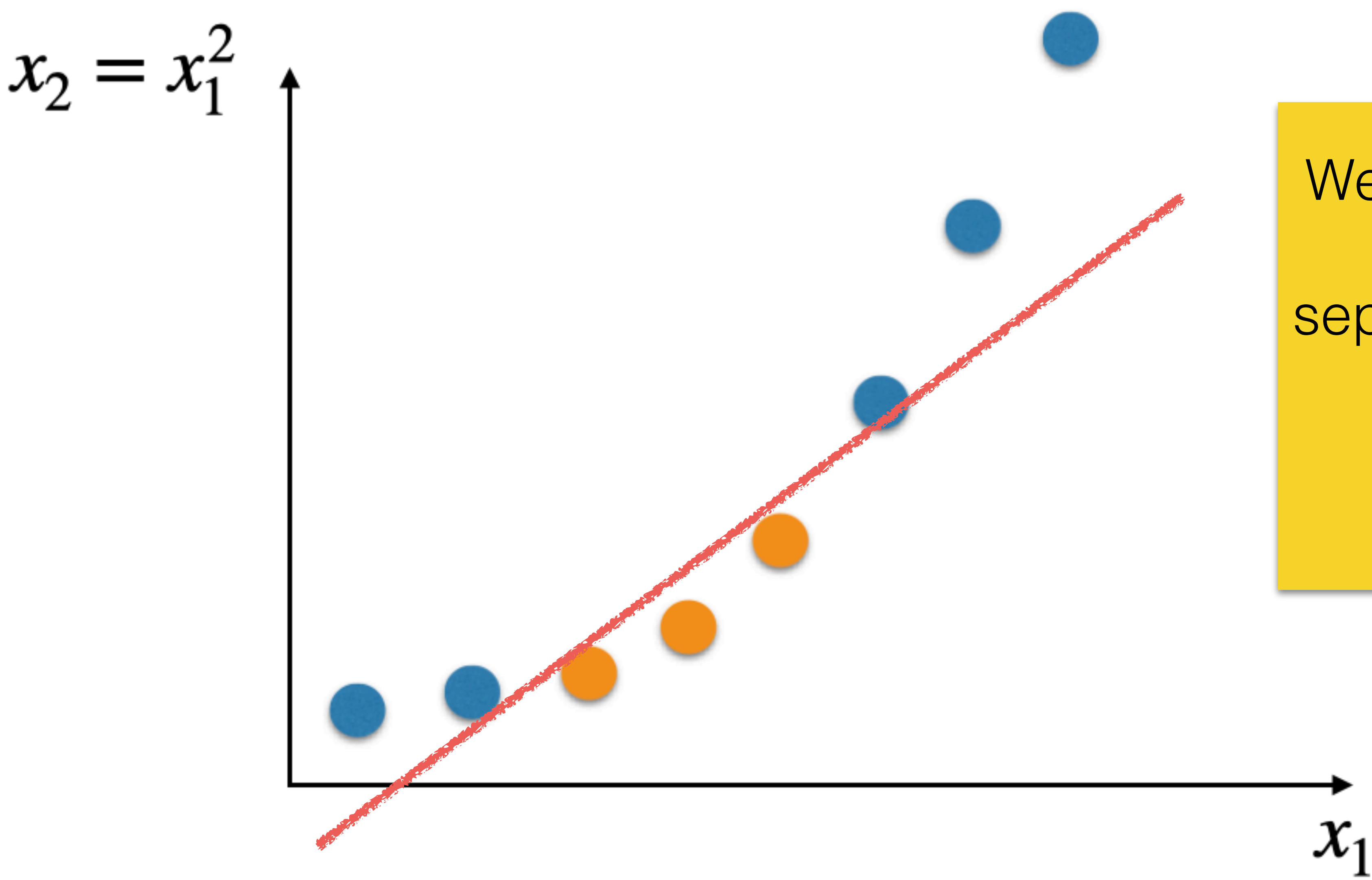


Nonlinear Transformation / Basis Expansion



Higher dimensional embedding / feature space: $\phi(\mathbf{x}) = (x_1, x_1^2)^T$

Nonlinear Transformation / Basis Expansion



We can now learn a linear model to separate the data on the higher dimensional embedding.

feature transform / basis expansion

basis functions

Higher dimensional embedding / feature space: $\phi(\mathbf{x}) = (x_1, x_1^2)^T$

Decision Boundaries Corresponding to Polynomials of Order p in the Original Space

- What feature transform could we use to make the problem linearly separable in the higher dimensional embedding?

- For a polynomial decision boundary of degree p in the original space, create a feature transform that includes all terms of order $\leq p$ that can be created based on the input variables $\mathbf{x}$.

- Example for polynomial of order 2 and a problem with 1 input variable:

$$\mathbf{x} = (1, x_1) \rightarrow \phi(\mathbf{x}) = (1, x_1, x_1^2)^T$$

- Example for polynomial of order 2 and a problem with 2 input variables:

$$\mathbf{x} = (1, x_1, x_2)^T \rightarrow \phi(\mathbf{x}) = (1, x_1, x_2, x_1^2, x_2^2, x_1 x_2)^T$$

Decision Boundaries Represented by Polynomials of Order p in the Original Space

- Create a nonlinear transform that includes all terms of order $\leq p$ that can be created based on the input variables $\mathbf{x}$.

- Example for polynomial of order 3 and a problem with 2 input variables:

$$\mathbf{x} = (1, x_1, x_2)^T \rightarrow \phi(\mathbf{x}) = (1, x_1, x_2, x_1^2, x_2^2, x_1x_2, x_1^3, x_2^3, x_1x_2^2, x_1^2x_2)^T$$

If we follow this idea, any decision boundary that is a polynomial of order p in $\mathbf{x}$ is linear in $\phi(\mathbf{x})$.

So, we can adopt linear models in the higher dimensional embedding formed by $\phi(\mathbf{x})$, to learn decision boundaries corresponding to polynomials of order p in $\mathbf{x}$.

Example

Consider that we need a quadratic decision boundary for a problem with 1 input variable:

$$w_0 x_0 + w_1 x_1 + w_2 x_1^2 = 0 \qquad \text{where } x_0 = 1$$

Nonlinear transform:

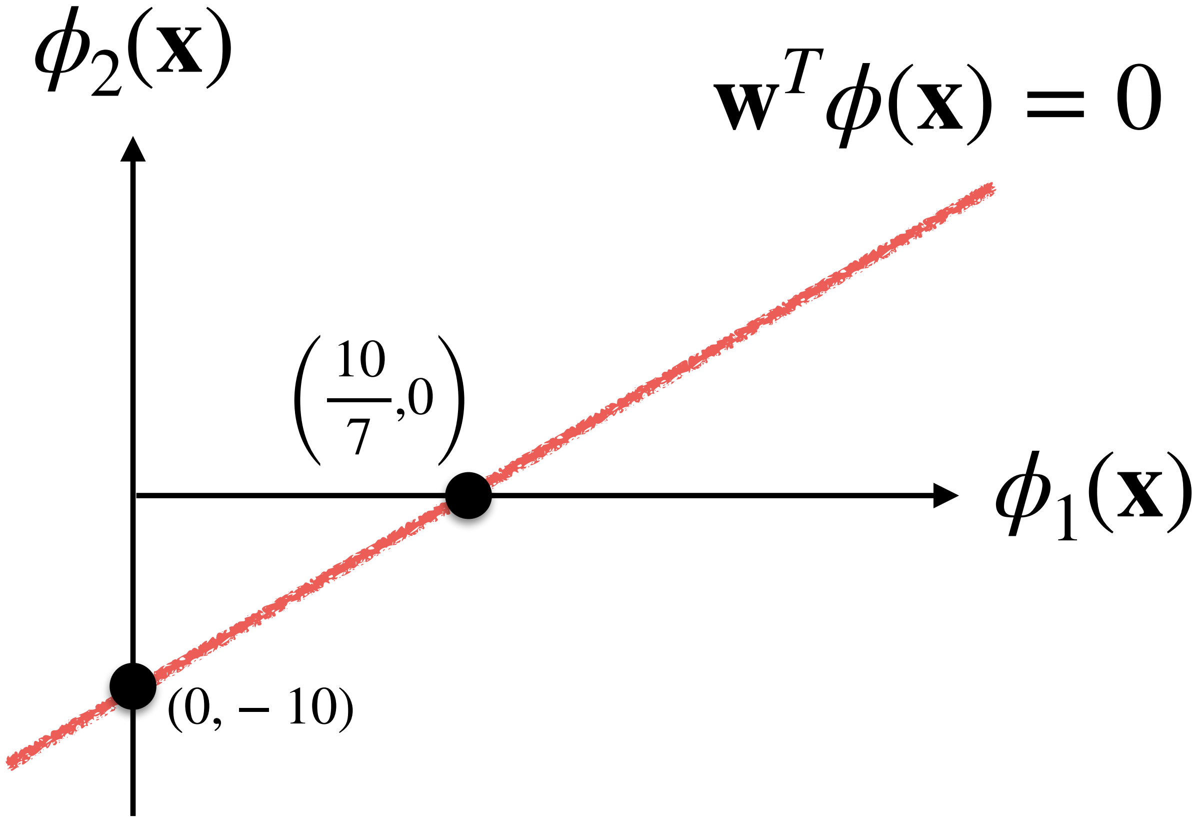
$$\mathbf{x} = (1, x_1)^T \rightarrow \phi(\mathbf{x}) = (1, x_1, x_1^2)^T$$

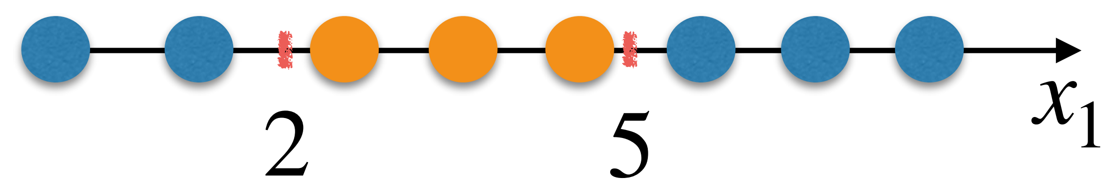
Linear decision boundary in the feature space corresponds to a quadratic decision boundary in the original space:

$$\mathbf{w}^T \phi(\mathbf{x}) = 0 \qquad \mathbf{w}^T = (w_0, w_1, w_2)$$

$$w_0 \times 1 + w_1 \phi_1(\mathbf{x}) + w_2 \phi_2(\mathbf{x}) = 0 \qquad w_0 \times 1 + w_1 x_1 + w_2 x_1^2 = 0$$

Illustration for $\mathbf{w}^T = (10, -7, 1)$, $\phi(\mathbf{x}) = (1, x_1, x_1^2)^T$





$$w_0 \times 1 + w_1\phi_1(\mathbf{x}) + w_2\phi_2(\mathbf{x}) = 0$$

$$10 \times 1 - 7\phi_1(\mathbf{x}) + 1\phi_2(\mathbf{x}) = 0$$

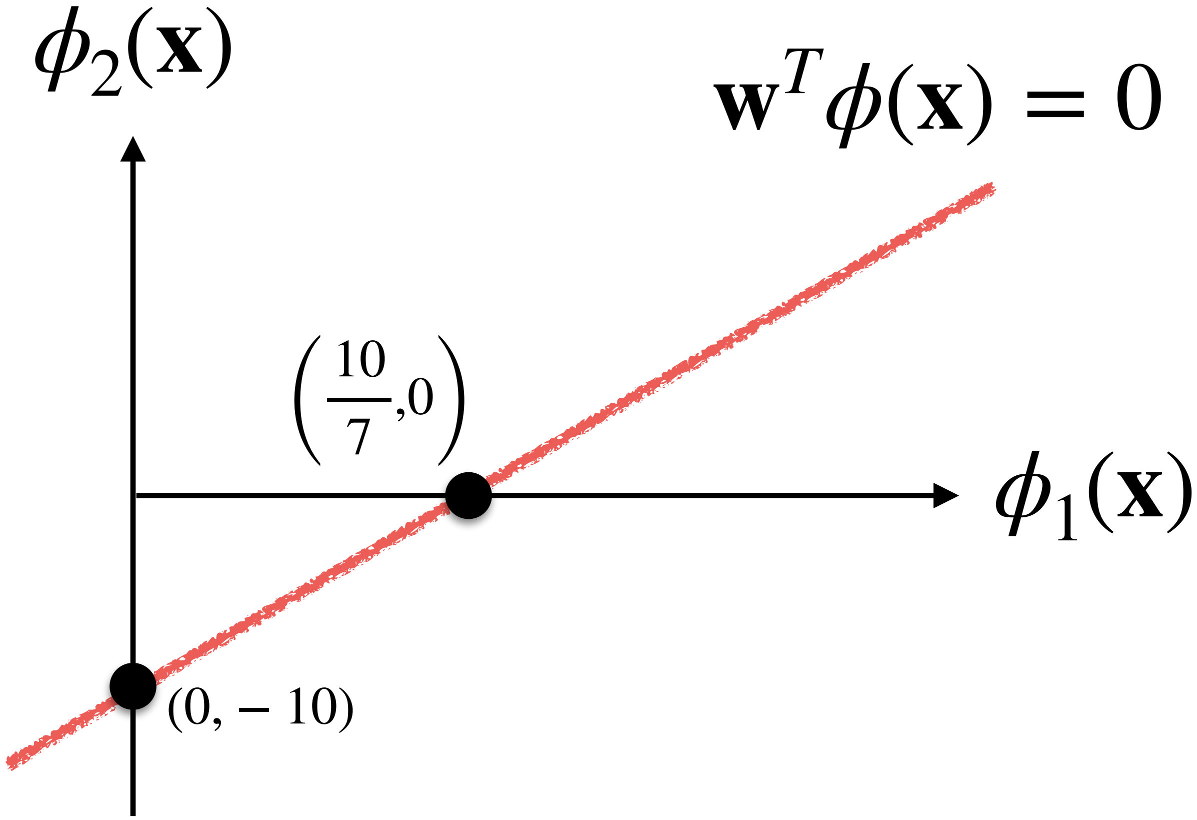
$$10 - 7\phi_1(\mathbf{x}) + 1\phi_2(\mathbf{x}) = 0$$

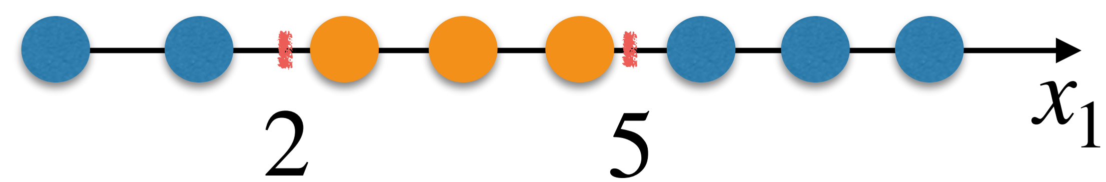
$$w_2 x_1^2 + w_1 x_1 + w_0 \times 1 = 0$$

$$1x_1^2 - 7x_1 + 10 = 0$$

$$x_1 = \frac{7 \pm \sqrt{(-7)^2 - 4 \times 1 \times 10}}{2 \times 1}$$

Illustration for $\mathbf{w}^T = (10, -7, 1),\ \phi(\mathbf{x}) = (1, x_1, x_1^2)^T$





When we include a basis function in the transformation, the decision boundary in the original space can include that in a term!

$$w_2 x_1^2 + w_1 x_1 + w_0 \times 1 = 0$$

$$1x_1^2 - 7x_1 + 10 = 0$$

$$x_1 = \frac{7 \pm \sqrt{(-7)^2 - 4 \times 1 \times 10}}{2 \times 1}$$

Other Nonlinear Transformations

- The previous slides showed nonlinear transformations to create polynomial decision boundaries in the original space.
  - So, they include terms of degree up to p.

- However, we can use any other nonlinear transformations that we wish to adopt, including non-polynomial ones.

- E.g:
  - $\mathbf{x} = (1, x_1) \rightarrow \phi(\mathbf{x}) = (1, x_1, e^{x_1})^T$

Dimensionality of the Feature Space

- Most of the time, we will be transforming the problem into a higher dimensional space.
- However, this is not necessarily the case.
- E.g., if we don't need a term with x_1^2 and a term with $x_1 x_2$ to form the decision boundary in the original space, we don't need to include them in the nonlinear transformation:
  - $\mathbf{x} = (1, x_1, x_2)^T \rightarrow \phi(\mathbf{x}) = (1, x_1, x_2^2)^T$
- In practice, we will often not know beforehand which terms are needed, so we will often be transforming the problem to a higher dimensional embedding.

Adopting Nonlinear Transformations in Logistic Regression

$$\text{logit}(p_1) = \mathbf{w}^T\mathbf{x} \qquad p_1 = p(1 \mid \mathbf{x}, \mathbf{w}) = \frac{e^{(\mathbf{w}^T\mathbf{x})}}{1 + e^{(\mathbf{w}^T\mathbf{x})}}$$

$$\downarrow$$

$$\text{logit}(p_1) = \mathbf{w}^T\phi(\mathbf{x}) \qquad p_1 = p(1 \mid \phi(\mathbf{x}), \mathbf{w}) = \frac{e^{(\mathbf{w}^T\phi(\mathbf{x}))}}{1 + e^{(\mathbf{w}^T\phi(\mathbf{x}))}}$$

Adopting Nonlinear Transformations in Logistic Regression

Given $\mathscr{T} = \{(\mathbf{x}^{(1)}, y^{(1)}), (\mathbf{x}^{(2)}, y^{(2)}), \cdots, (\mathbf{x}^{(N)}, y^{(N)})\}$, $\underset{\mathbf{w}}{\operatorname{argmin}} E(\mathbf{w})$

$$E(\mathbf{w}) = -\sum_{i=1}^{N} y^{(i)} \ln p(1 \,|\, \mathbf{x}^{(i)}, \mathbf{w}) + (1 - y^{(i)}) \ln (1 - p(1 \,|\, \mathbf{x}^{(i)}, \mathbf{w}))$$

$$\downarrow$$

Given $\mathscr{T} = \{(\phi(\mathbf{x}^{(1)}), y^{(1)}), (\phi(\mathbf{x}^{(2)}), y^{(2)}), \cdots, (\phi(\mathbf{x}^{(N)}), y^{(N)})\}$, $\underset{\mathbf{w}}{\operatorname{argmin}} E(\mathbf{w})$

$$E(\mathbf{w}) = -\sum_{i=1}^{N} y^{(i)} \ln p(1 \,|\, \phi(\mathbf{x}^{(i)}), \mathbf{w}) + (1 - y^{(i)}) \ln (1 - p(1 \,|\, \phi(\mathbf{x}^{(i)}), \mathbf{w}))$$

Adopting Nonlinear Transformations in Logistic Regression

$$\nabla_E(\mathbf{w}) = \sum_{i=1}^{N} (p(1 \mid \mathbf{x}^{(i)}, \mathbf{w}) - y^{(i)})\mathbf{x}^{(i)}$$

$$H_E(\mathbf{w}) = \sum_{i=1}^{N} p(1 \mid \mathbf{x}^{(i)}, \mathbf{w})(1 - p(1 \mid \mathbf{x}^{(i)}, \mathbf{w}))\mathbf{x}^{(i)}\mathbf{x}^{(i)^T}$$

$$\Downarrow$$

$$\nabla_E(\mathbf{w}) = \sum_{i=1}^{N} (p(1 \mid \phi(\mathbf{x}^{(i)}), \mathbf{w}) - y^{(i)})\phi(\mathbf{x}^{(i)})$$

$$H_E(\mathbf{w}) = \sum_{i=1}^{N} p(1 \mid \phi(\mathbf{x}^{(i)}), \mathbf{w})(1 - p(1 \mid \phi(\mathbf{x}^{(i)}), \mathbf{w}))\phi(\mathbf{x}^{(i)})\phi(\mathbf{x}^{(i)})^T$$

Adopting Nonlinear Transformations

1. Choose a nonlinear transformation.

2. Apply it to the training examples so that they have the format $(\phi(\mathbf{x}), y)$.

3. Create a linear model based on the transformed training examples (using the same learning algorithms we've learned so far).

4. Determine the (nonlinear) model by replacing $\phi_i(\mathbf{x})$ with the corresponding value that depends on $\mathbf{x}$.

Adopting Nonlinear Transformations: Example

1. Choose a nonlinear transformation.

   $\mathbf{x} = (1, x_1) \rightarrow \phi(\mathbf{x}) = (1, x_1, x_1^2)^T$

2. Apply it to the training examples so that they have the format $(\phi(\mathbf{x}), y)$.

   $\mathcal{T} = \{(\phi(\mathbf{x}^{(1)}), y^{(1)}), (\phi(\mathbf{x}^{(2)}), y^{(2)}), \cdots, (\phi(\mathbf{x}^{(N)}), y^{(N)})\}$

3. Create a linear model based on the transformed training examples (using the same learning algorithms we've learned so far).

   Given $\mathcal{T}$, $\underset{\mathbf{w}}{\operatorname{argmin}}\, E(\mathbf{w})$

4. Determine the (nonlinear) model by replacing $\phi_i(\mathbf{x})$ with the corresponding value that depends on $\mathbf{x}$.

   $w_0 \times 1 + w_1\phi_1(\mathbf{x}) + w_2\phi_2(\mathbf{x}) = 0 \rightarrow \; w_0 \times 1 + w_1 x_1 + w_2 x_1^2 = 0$

Is Logistic Regression Still a Linear Model If We Adopt Nonlinear Transformations?

- Usually, when we refer to the linearity of a model, we are referring to linearity with respect to its parameters.

- The logistic regression model is still linear with respect to its parameters $\mathbf{w}$.

- The adoption of non-linear transformations of the input variables transforms the problem into a different space, where it can hopefully be solved with a linear model.

- Logistic regression is still finding just a linear model (in the embedding), even though this model leads to a nonlinear decision boundary in the original space.

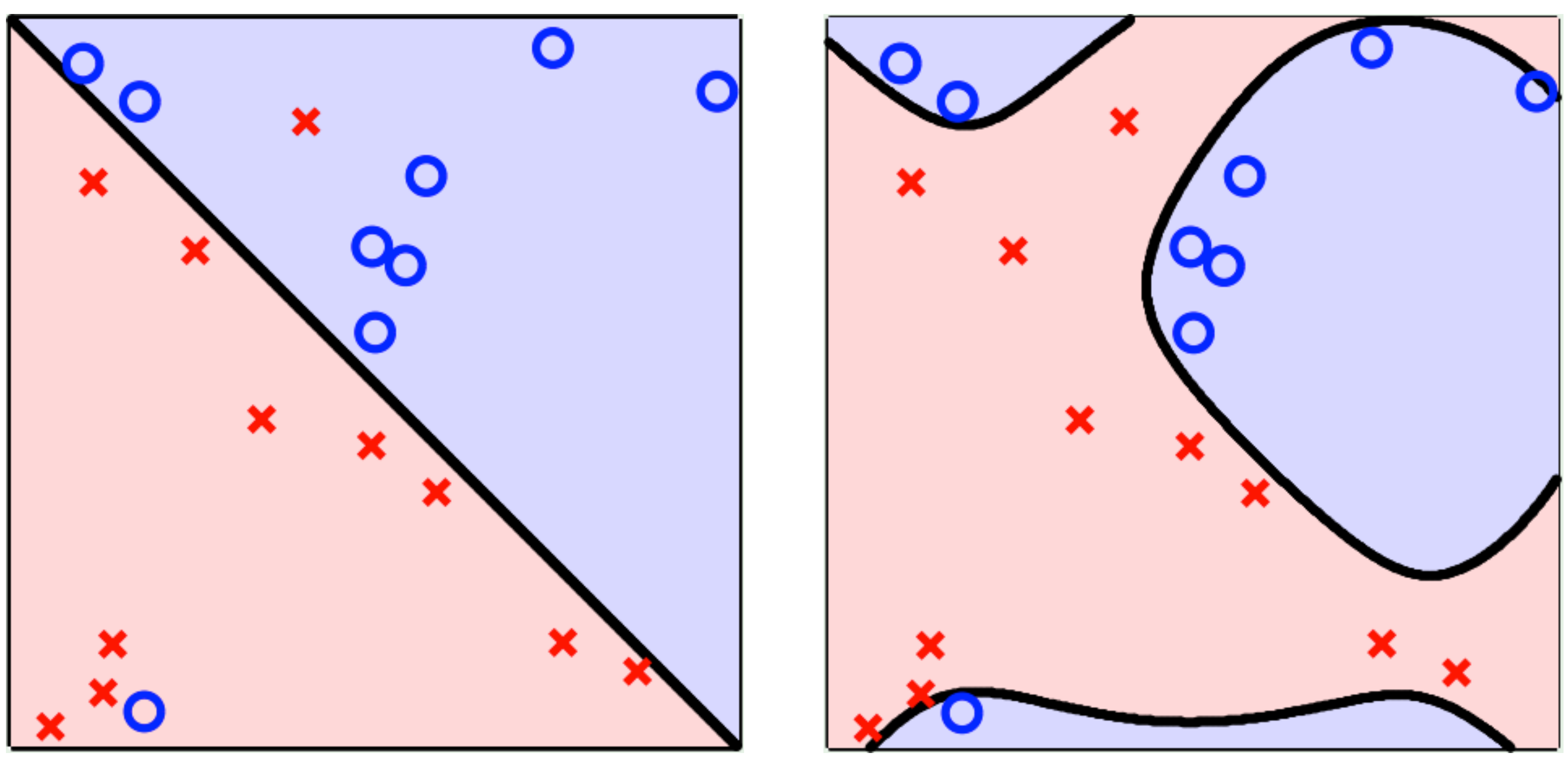
- When using nonlinear transformations, one can say that logistic regression is linear in its parameters, despite being nonlinear in the original problem's input variables.

Advantages of Linear Models

- Linear models are often associated to relatively efficient learning algorithms.
- They can be robust and have good generalisation properties.

Caveats of Nonlinear Transforms

- The number of dimensions may become very high.
- Choosing a nonlinear transformation that fits the training examples well does not necessarily mean that there will be good generalisation. It may lead to overfitting.



Linear Fit   4th Order Polynomial

Figure from: Abu-Mostafa et al's Learning from Data: A Short Course.

Summary

- We can create nonlinear transformations to obtain a (higher dimensional) embedding where our problems become linearly separable, even if they were not linearly separable in the original space.

- We can then adopt our original logistic regression to create a linear decision boundary in this (higher dimensional) embedding.

- This idea can is also applicable to other linear models.

Tutorial Poll

- Available in Week 3 on Teams

Leandro Minku (Computer Science) via Polls: Leandro Minku (Computer Science) sent ...
posted in Machine Learning 2024 - Teaching / Week 3 - Nonlinear transform, SVM primal on 14 October 2024 09:30