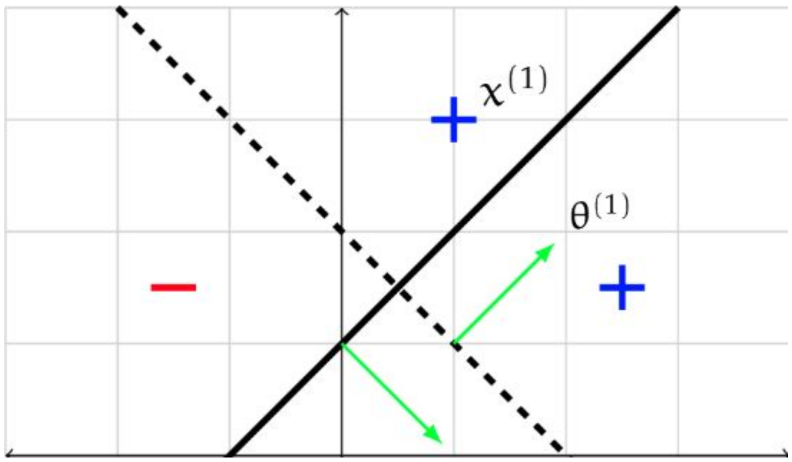
Introduction to Machine Learning



Feature Representations

Review: Linear => Logistic Regression

Data $D = \{x^{(i)}, y^{(i)}\}_{i=1}^{n}, \;\; x^{(i)} \in \mathbb{R}^d, y^{(i)} \in \mathbb{R}$

Hypothesis $h(x;\theta) = \theta^\top x + \theta_0$

Cost $J(\theta) = \frac{1}{n}\sum_{i=1}^{n}(y^{(i)} - (\theta^\top x^{(i)} + \theta_0))^2$

Optimization Analytic solution

Review: Linear => Logistic Regression

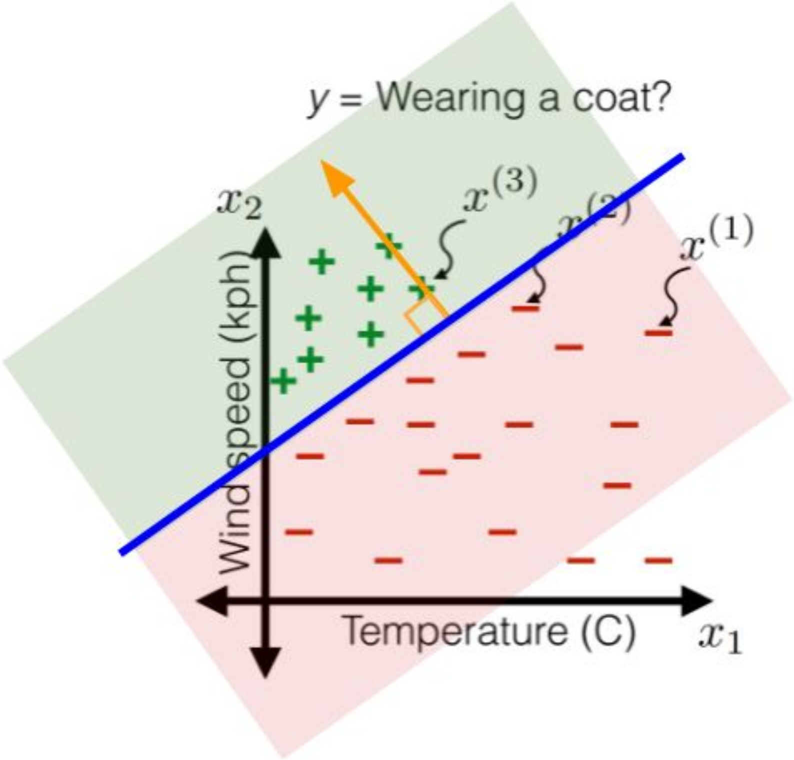
Data $D = \{x^{(i)}, y^{(i)}\}_{i=1}^{n}, \;\; x^{(i)} \in \mathbb{R}^d, \; y^{(i)} \in \{-1, 1\}$

Hypothesis $h(x) = \mathsf{sign}(\theta^\top x + \theta_0)$

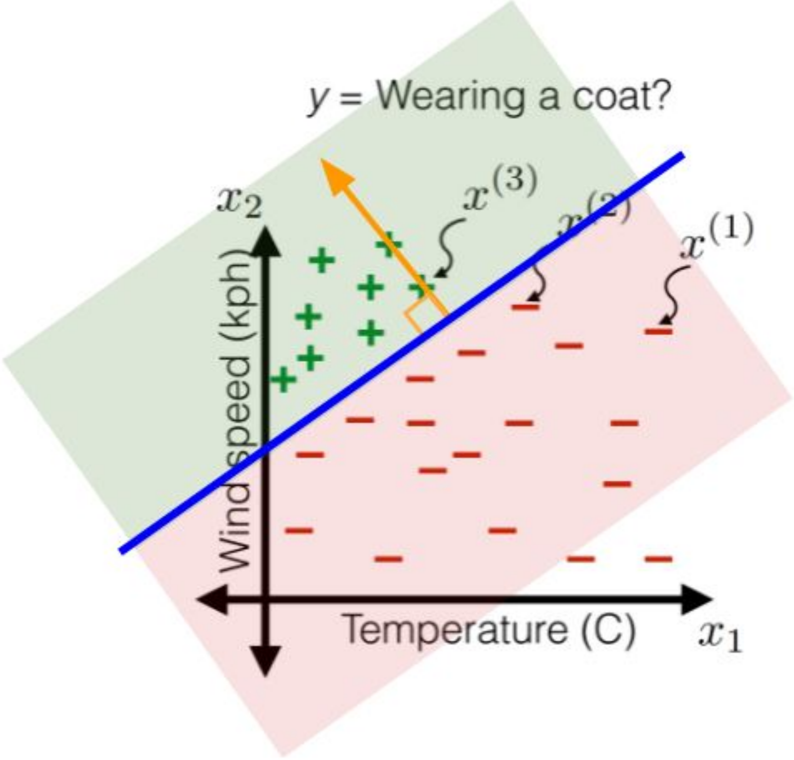
Cost
$$J(\theta) = - \; \frac{1}{n} \sum_{i=1}^{n} \mathbb{1}\left\{ y^{(i)} = +1 \right\} \log \sigma(\theta^\top x^{(i)} + \theta_0) + \mathbb{1}\left\{ y^{(i)} = -1 \right\} \log(1 - \sigma(\theta^\top x^{(i)} + \theta_0))$$

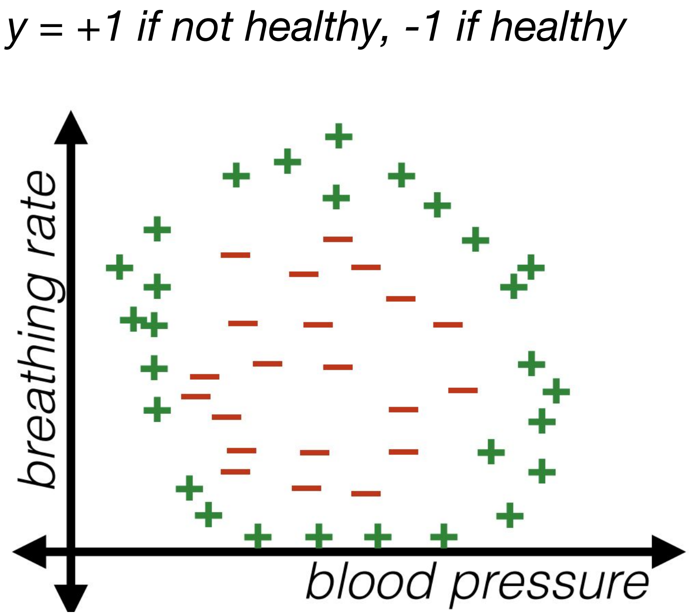
Optimization Gradient Descent

Linear Classifiers

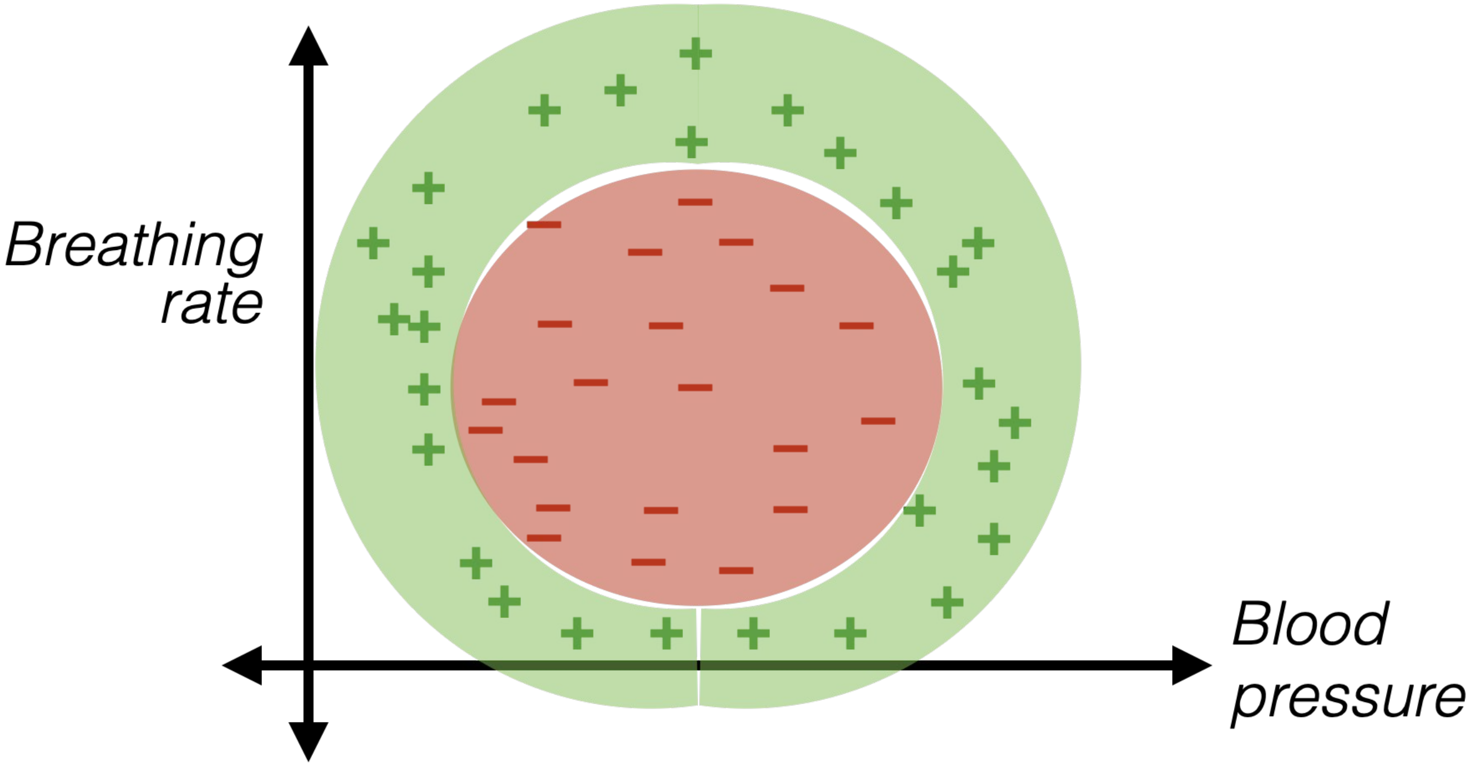


Linear Classifiers?

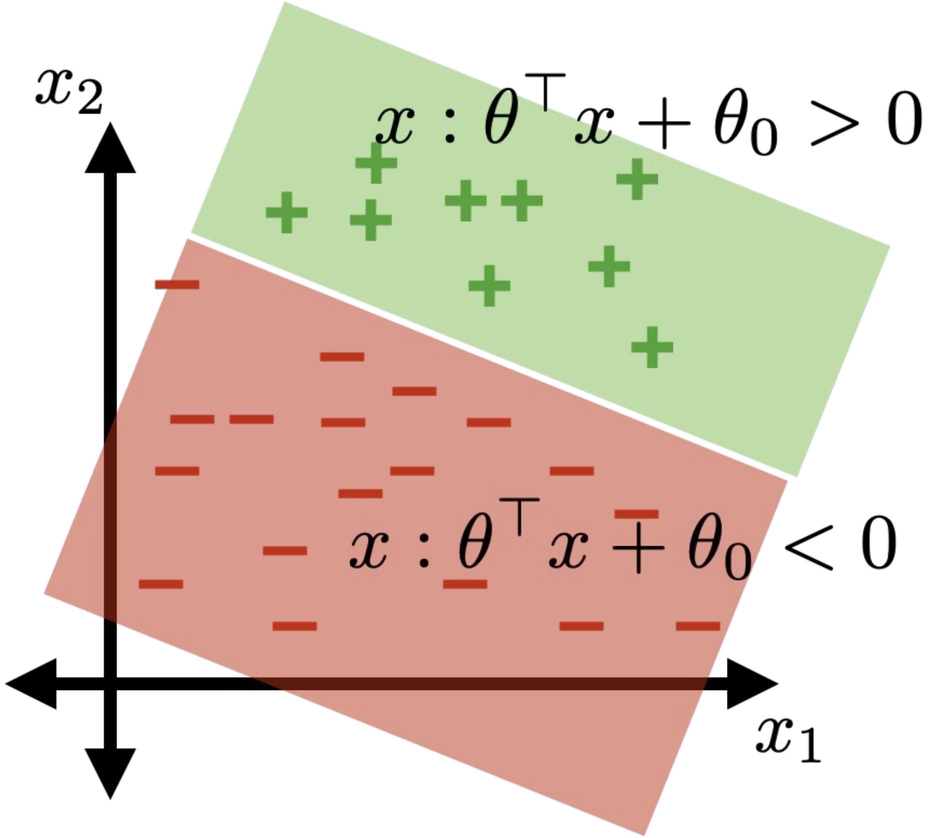


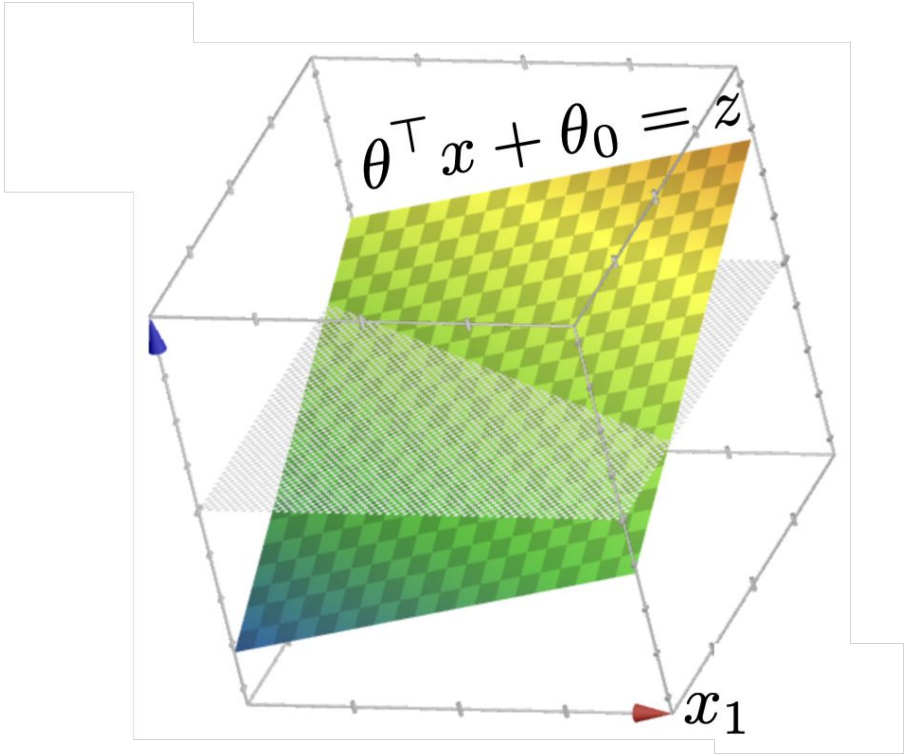


Non-Linear Classifiers

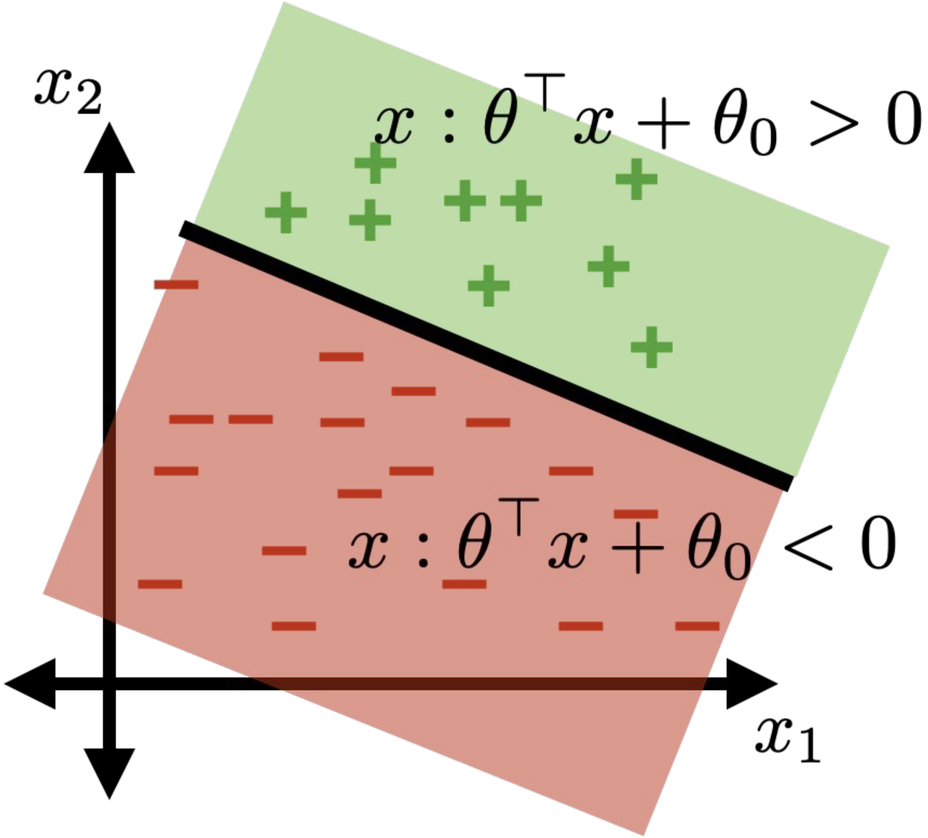


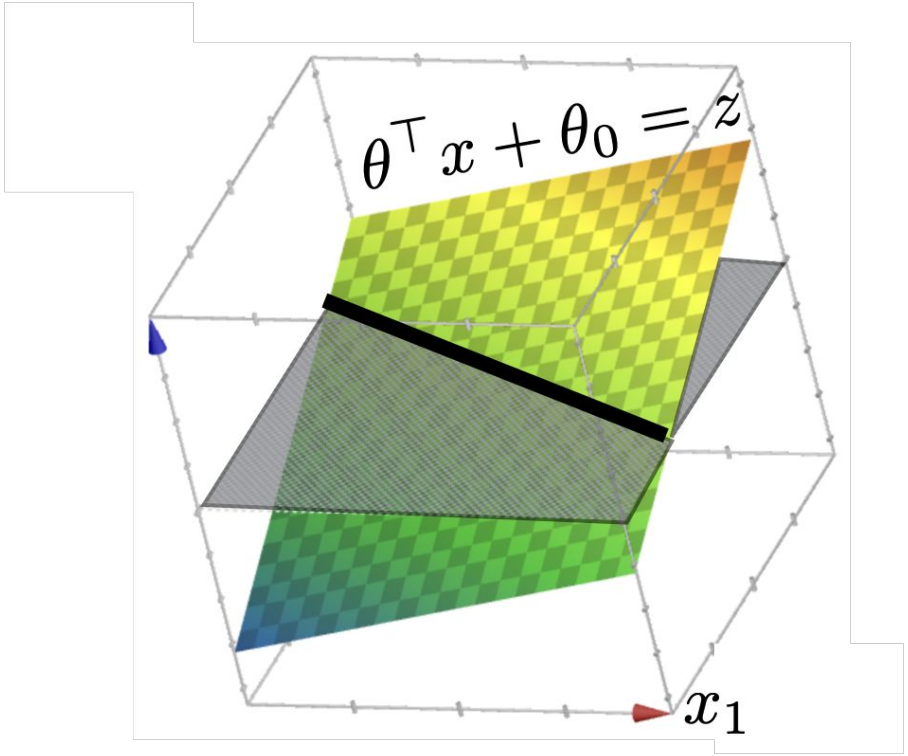
Review: Linear Classification Boundaries



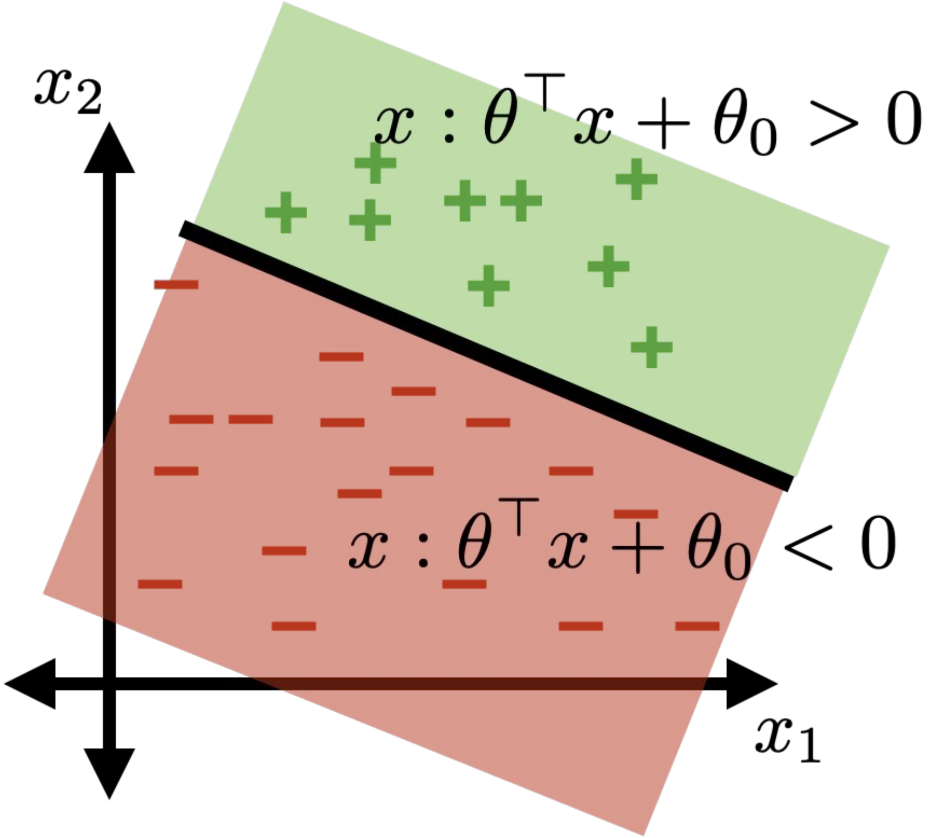


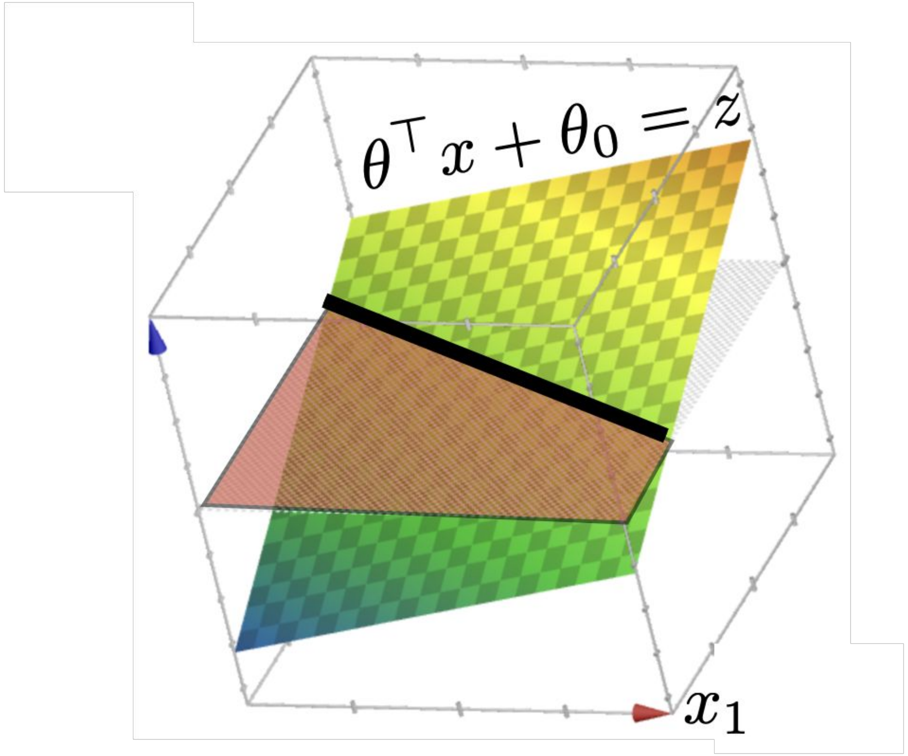
Review: Linear Classification Boundaries



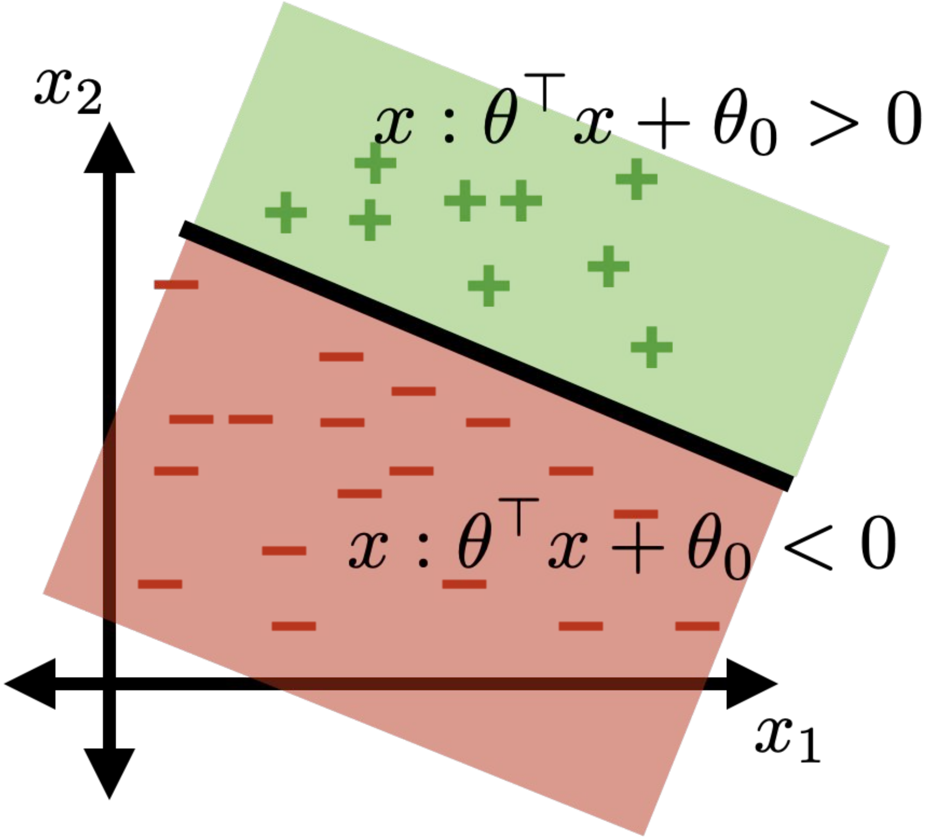


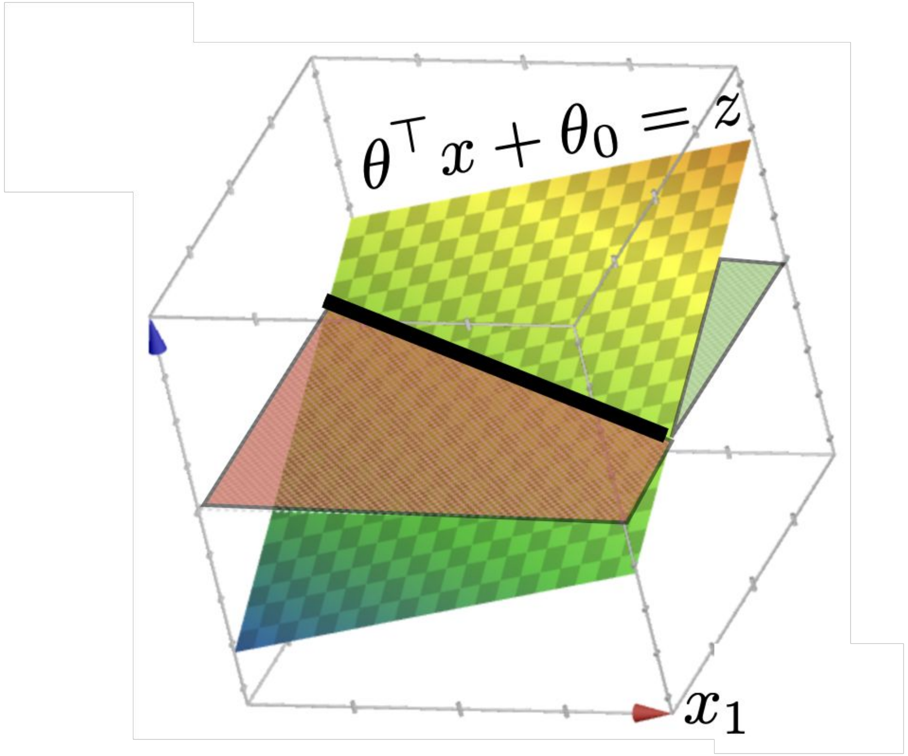
Review: Linear Classification Boundaries



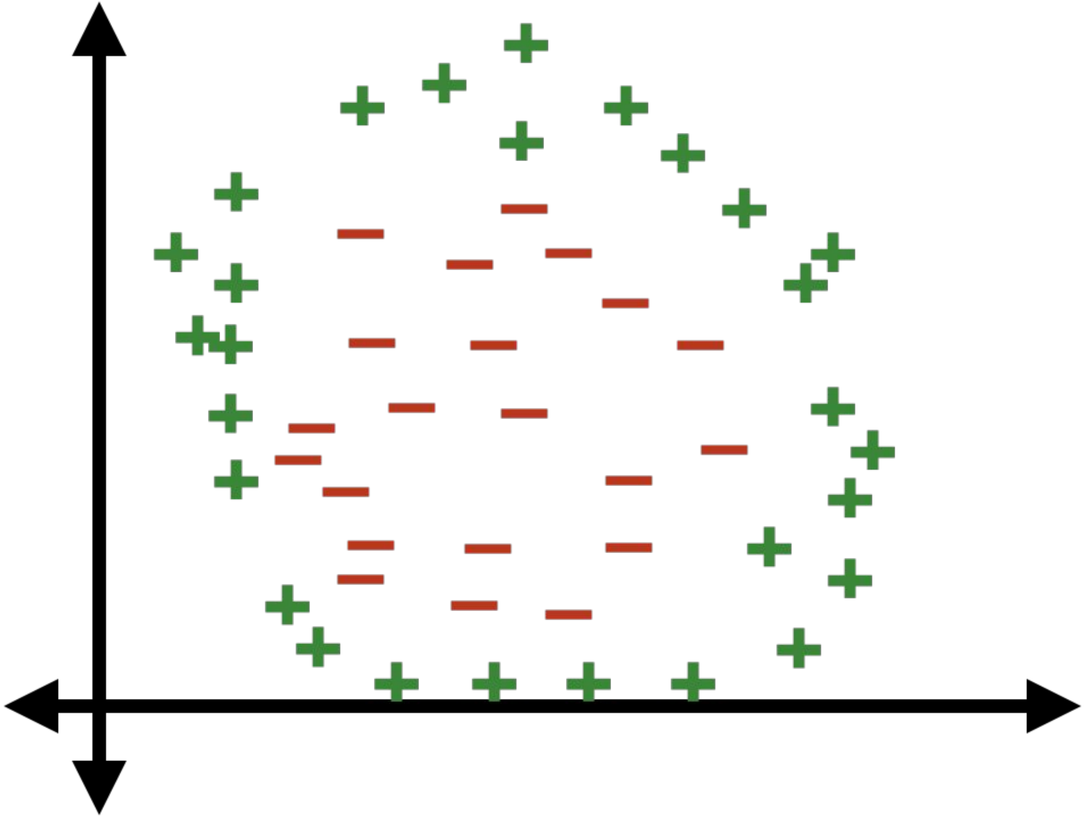


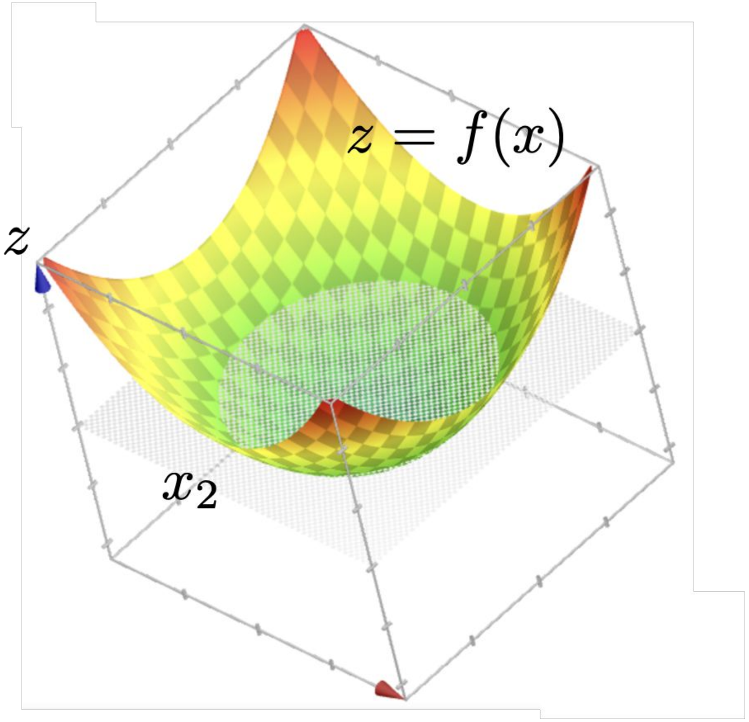
Review: Linear Classification Boundaries



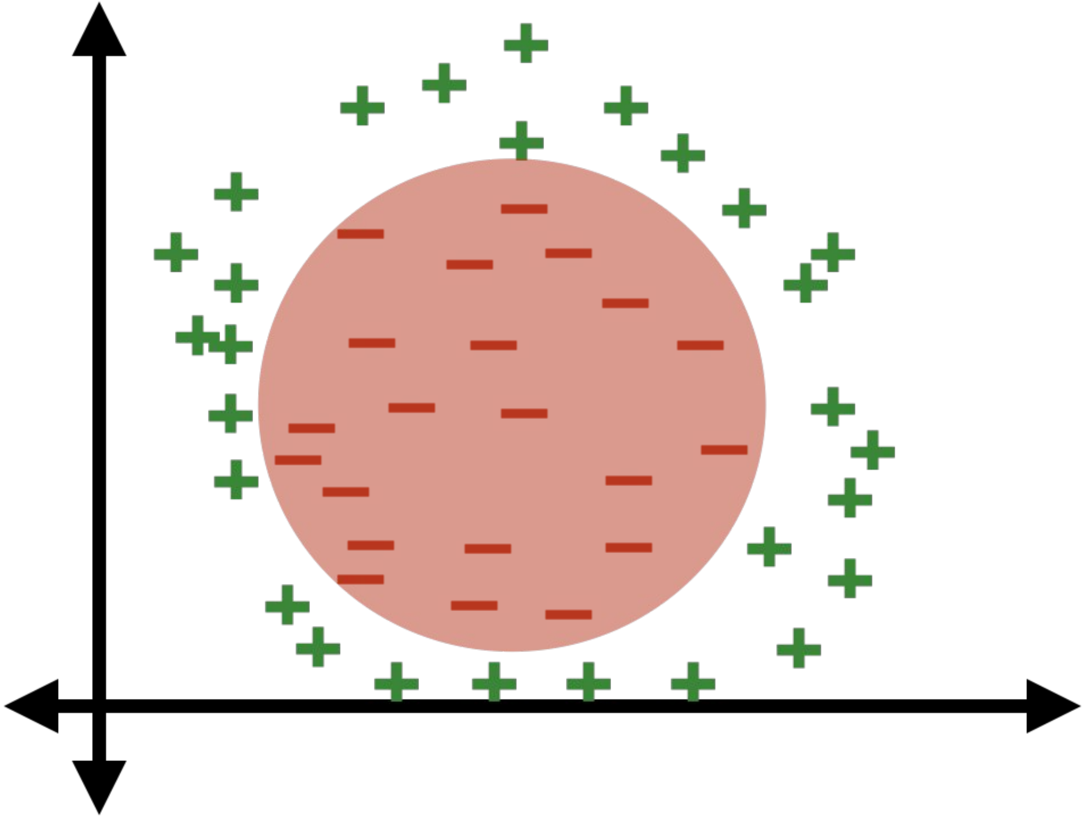


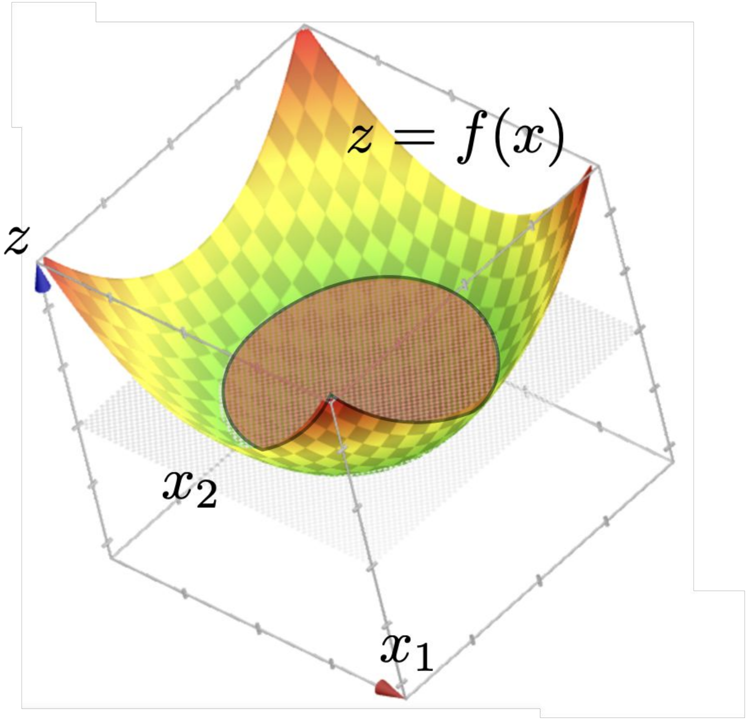
Non-Linear Classification Boundaries



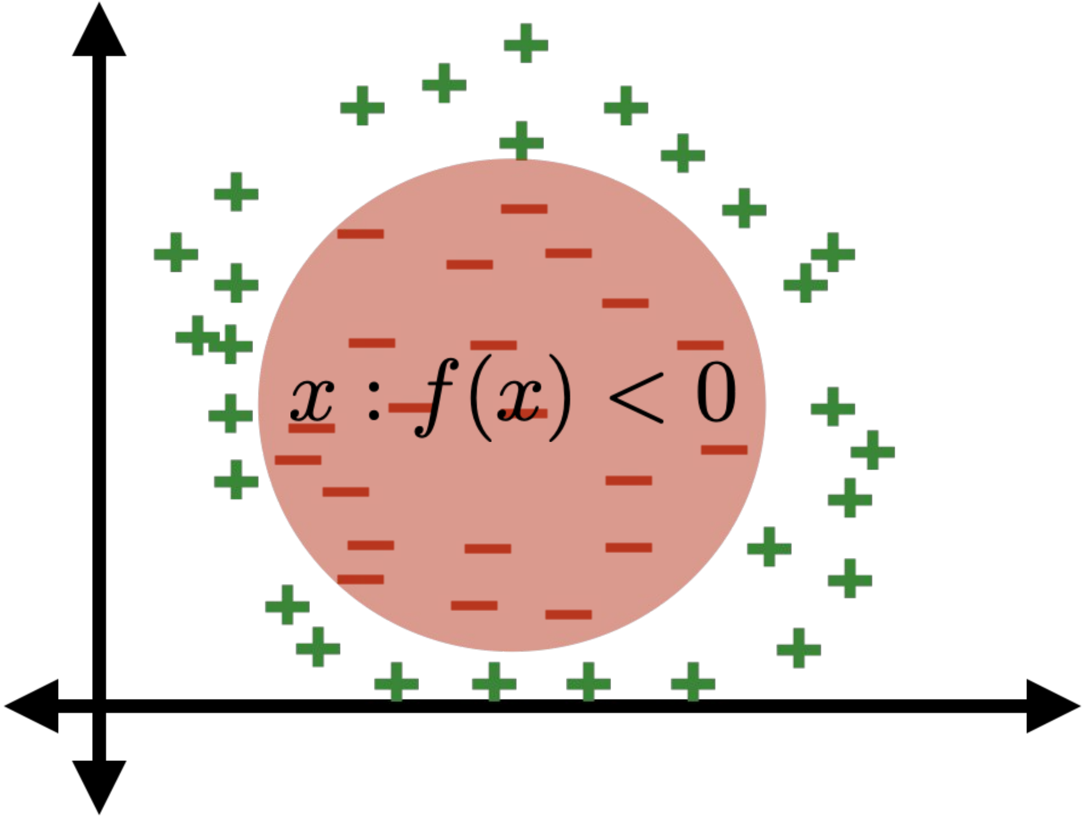


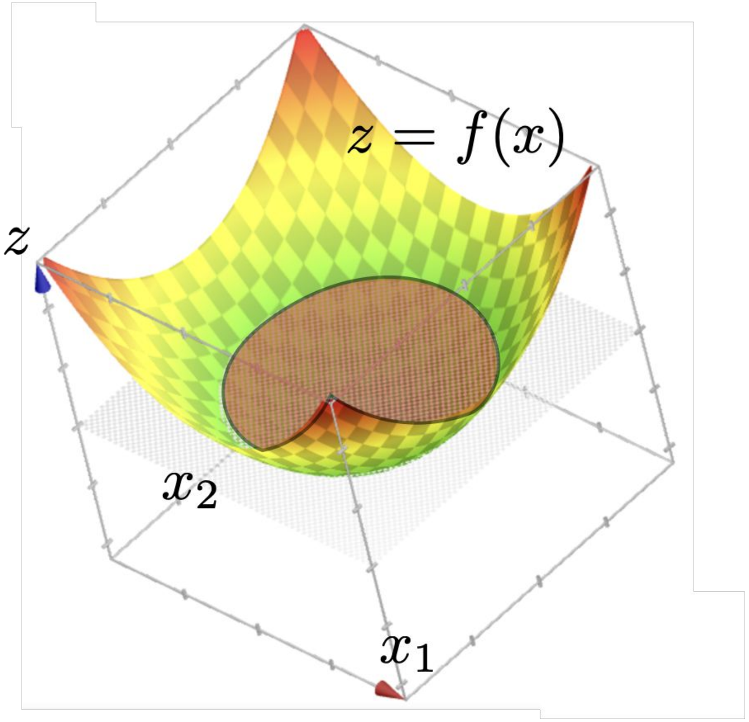
Non-Linear Classification Boundaries



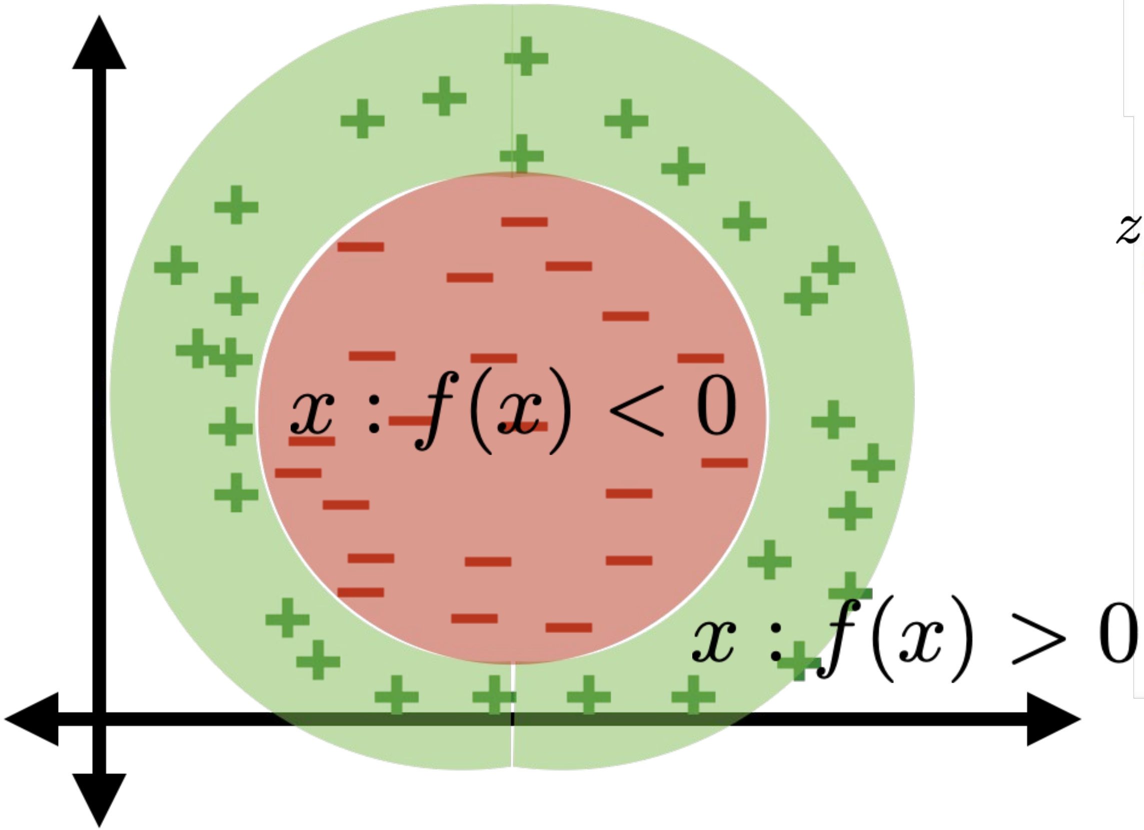


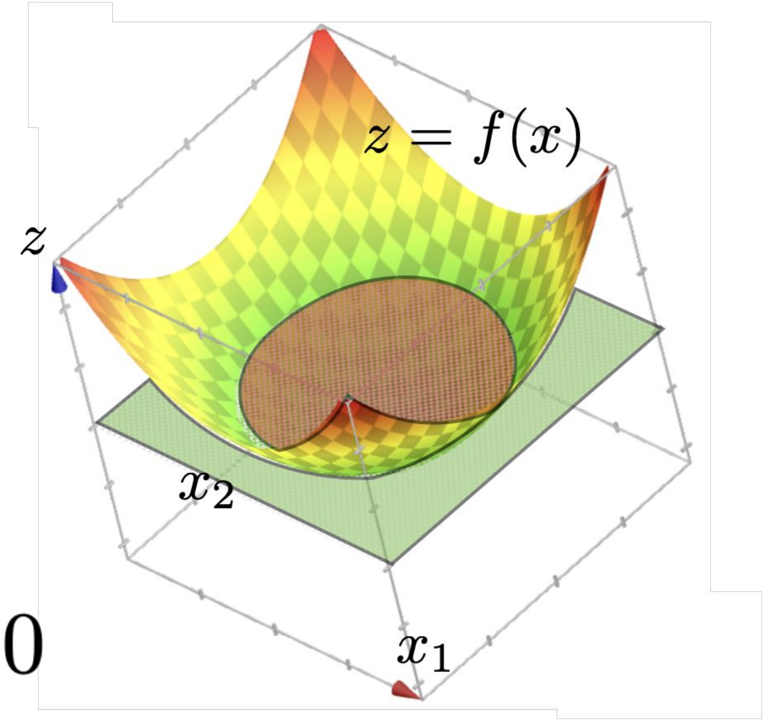
Non-Linear Classification Boundaries





Non-Linear Classification Boundaries





Polynomial basis

Idea: approximate a smooth function with a k-th order Taylor polynomial

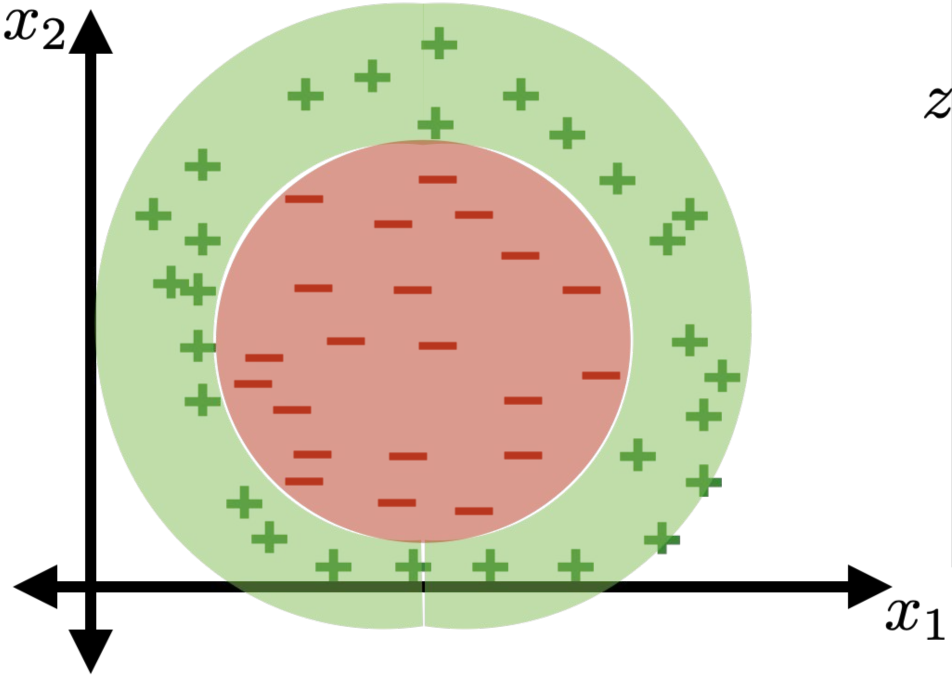
order (*k*)	terms when *d*=1	terms for general d
0	$[1]$	
1	$[1, x_1]$	
2	$[1, x_1, x_1^2]$	
3	$[1, x_1, x_1^2, x_1^3]$	

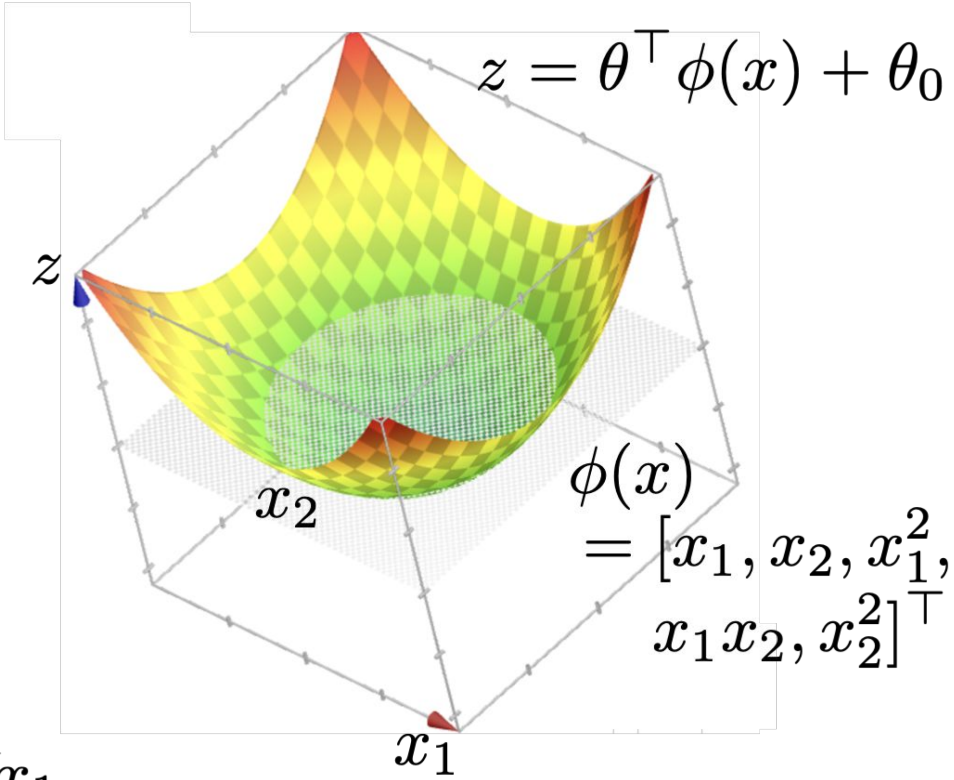
Polynomial basis

Idea: approximate a smooth function with a k-th order Taylor polynomial

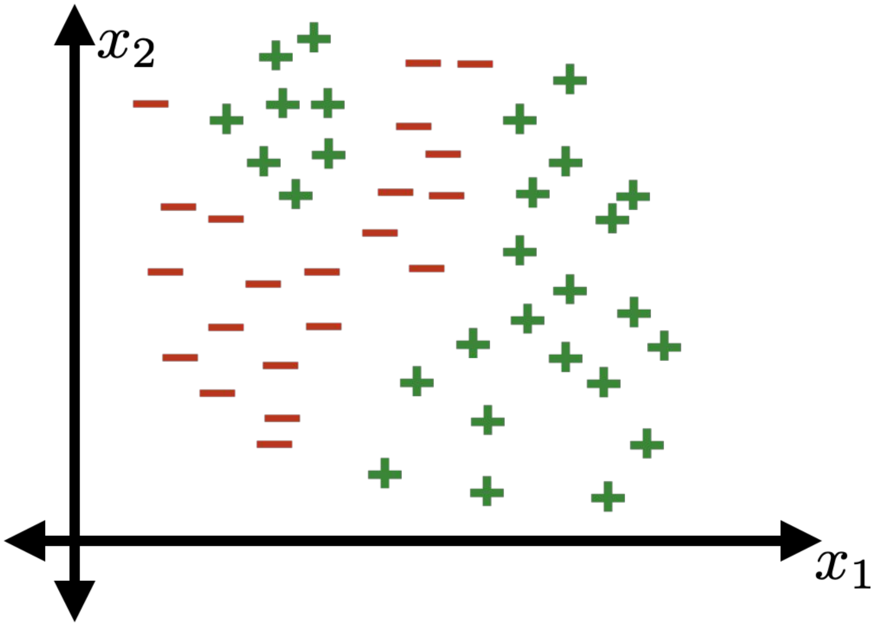
order (k)	terms when d=1	terms for general d
0	$[1]$	$[1]$
1	$[1, x_1]$	$[1, x_1, \ldots, x_d]$
2	$[1, x_1, x_1^2]$	$[1, x_1, \ldots, x_d,$ $x_1^2, x_1x_2, \ldots, x_{d-1}x_d, x_d^2]$
3	$[1, x_1, x_1^2, x_1^3]$	$[1, x_1, \ldots, x_d,$ $x_1^2, x_1x_2, \ldots, x_{d-1}x_d, x_d^2,$ $x_1^3, x_1^2x_2, x_1x_2x_3, \ldots, x_d^3]$

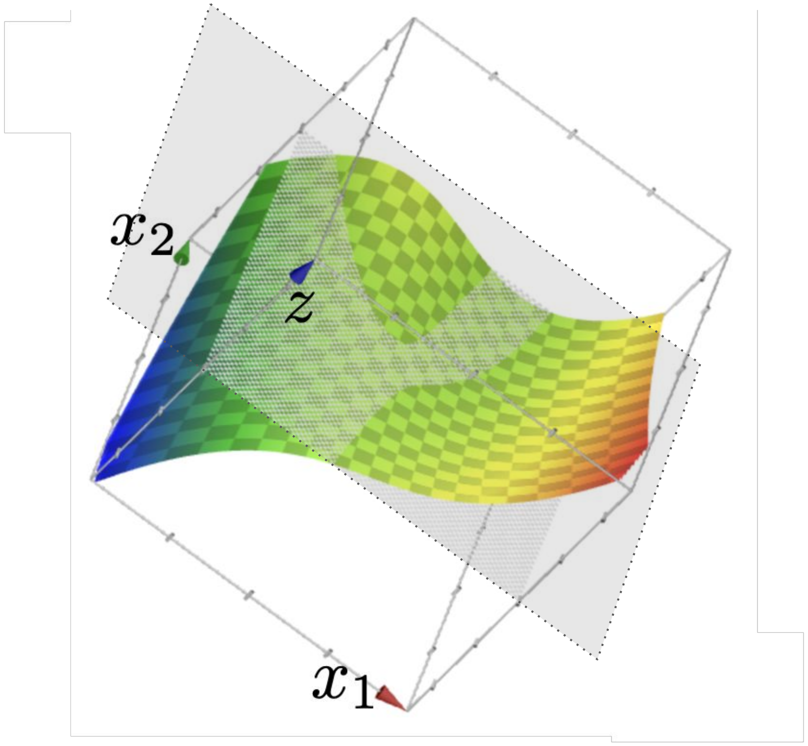
Polynomial basis



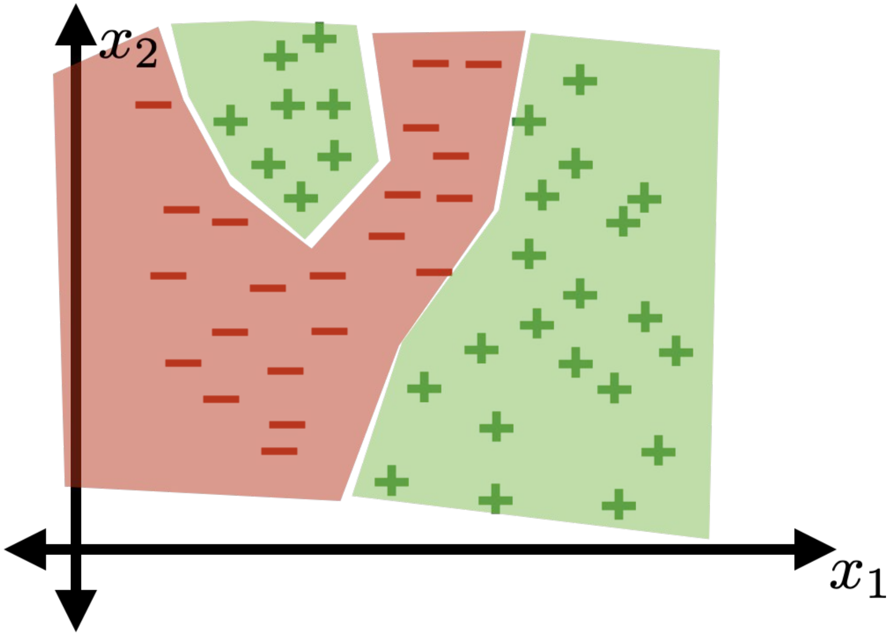


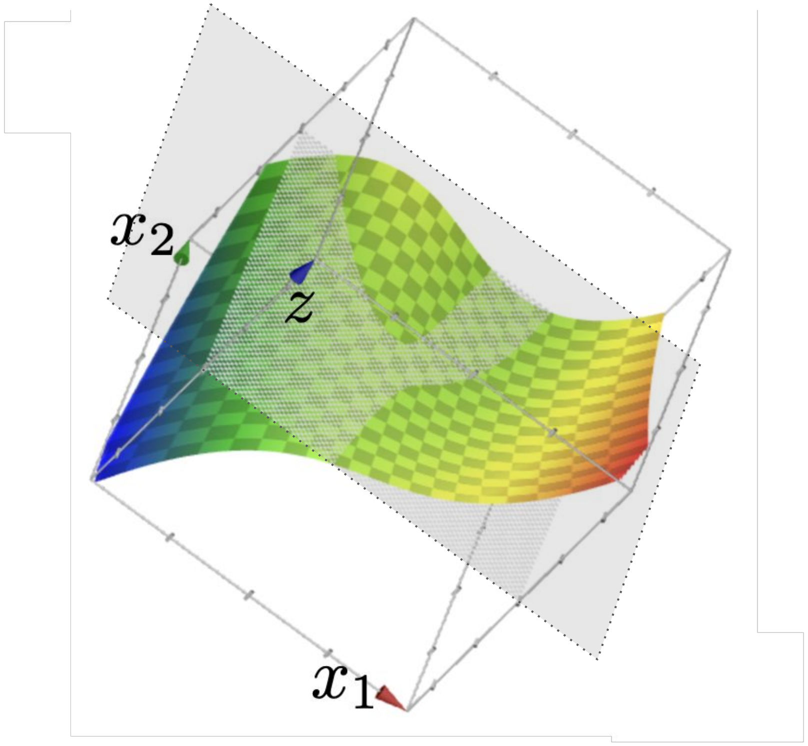
Polynomial basis



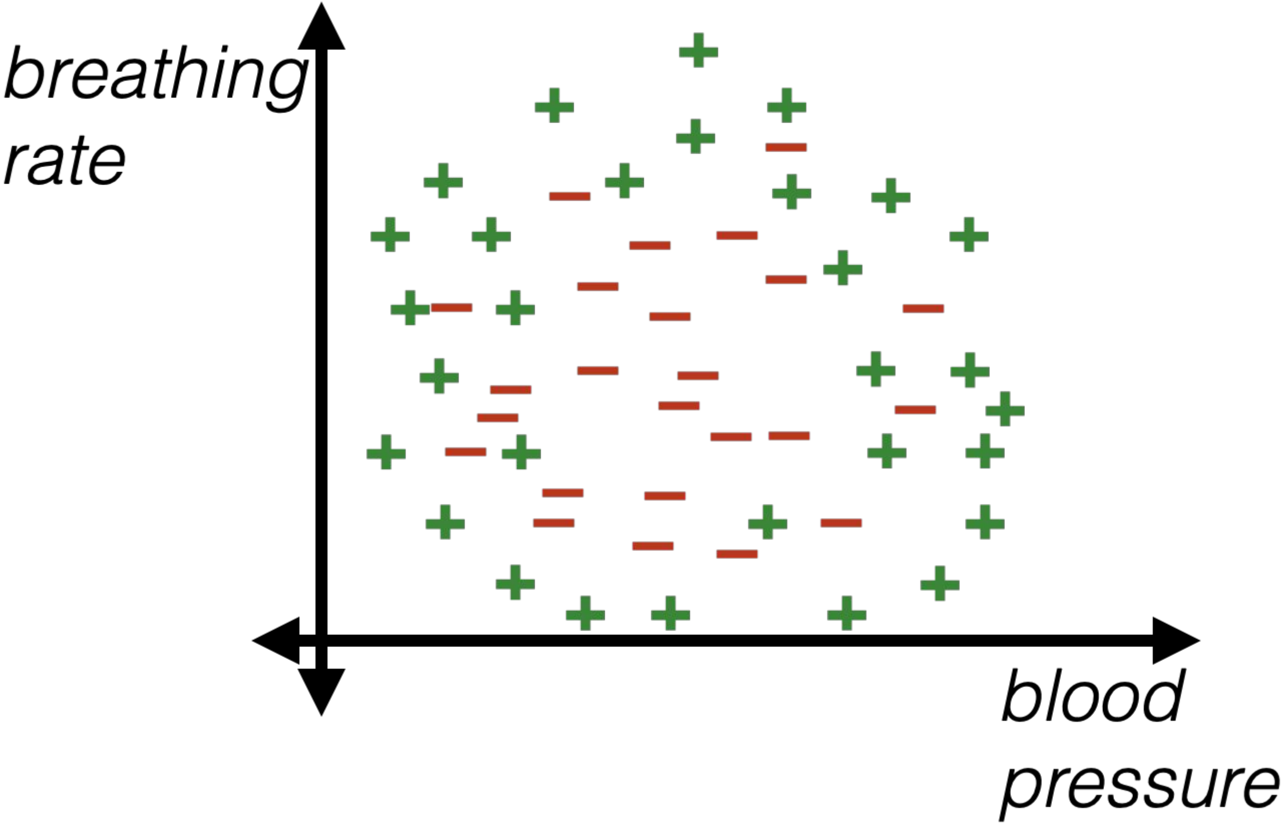


Polynomial basis

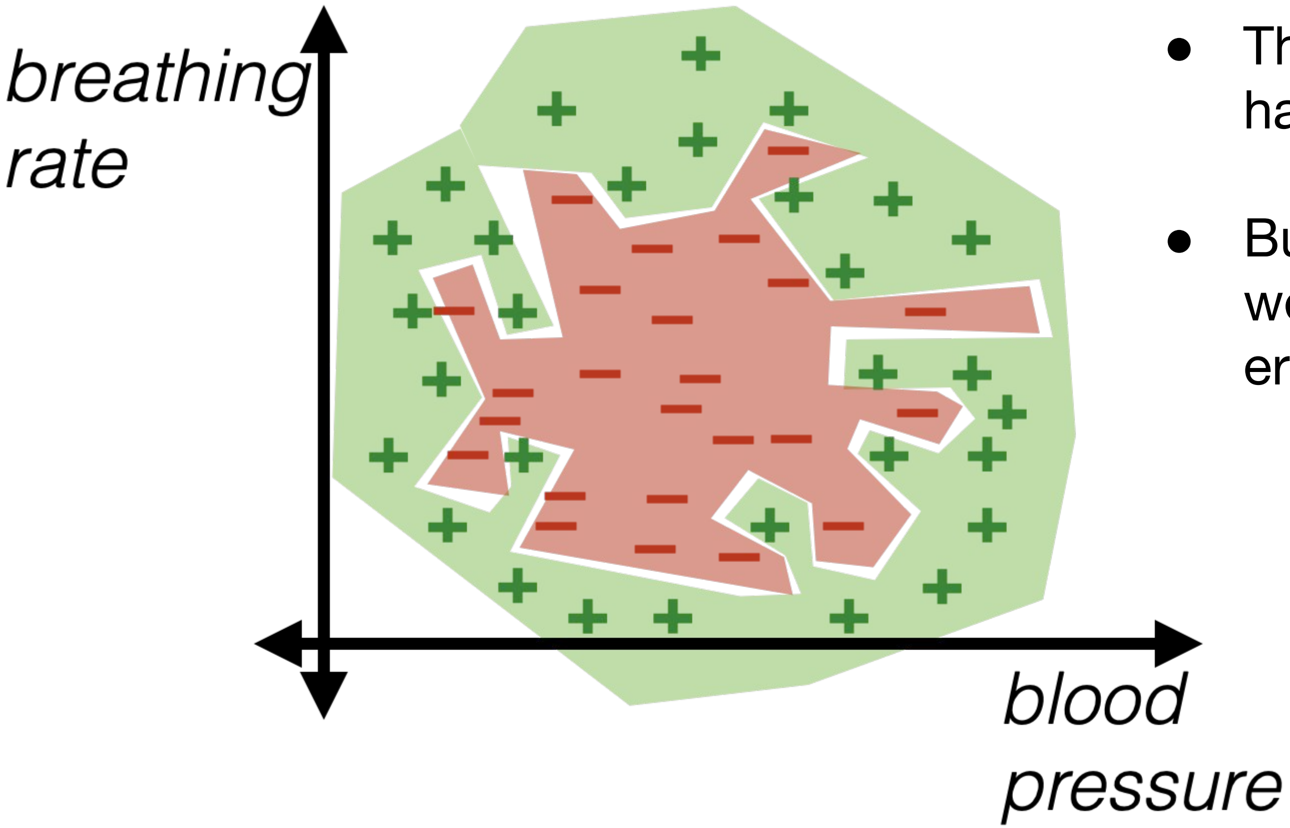




Polynomial basis



Polynomial basis



- This decision boundary has 0 training error!
- But unlikely to generalize well ⇒ high estimation error (i.e., overfitting)

Radial basis functions

- New idea: use “distance” from training points (or a subset thereof) to define features.

$$\phi(x) = \left[f(x, x^{(1)}), f(x, x^{(2)}), \ldots, f(x, x^{(n)}) \right]$$

$$f(x, y) = e^{-\beta ||x-y||^2}$$

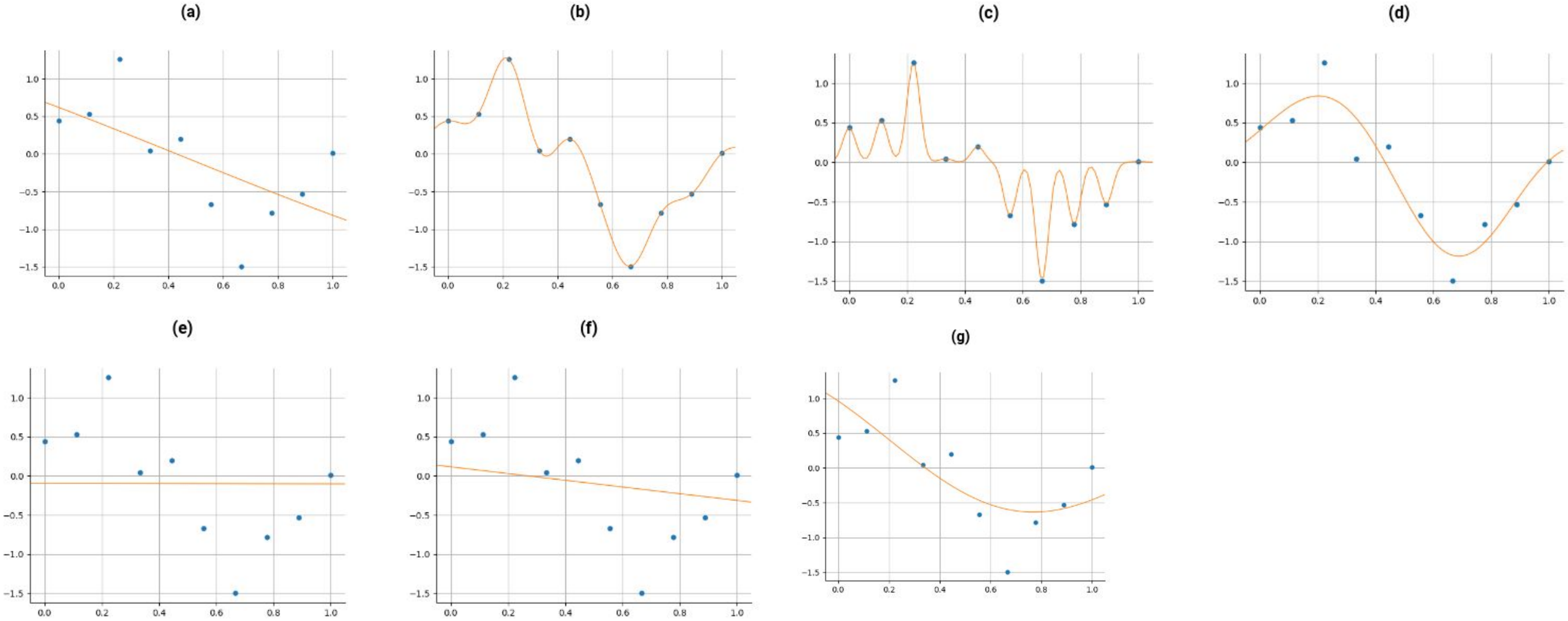
- For what value of x is $f(x, y)$ maximized?

Radial basis functions: HW5 Q5

$$\phi(x) = \left[f(x, x^{(1)}), f(x, x^{(2)}), \ldots, f(x, x^{(n)}) \right]$$

$$f(x, y) = e^{-\beta \|x-y\|^2}$$

$$\beta = \{0.001, 0.01, 0.1, 1, 10, 100, 1000\}$$



Strategies for Encoding Features

Scalar Features:

- Min-max normalization

$$x_{\mathsf{norm}} = \frac{x - \min(X)}{\max(X) - \min(X)}$$

- Standardization

$$x_{\mathsf{std}} = \frac{x - \mathsf{mean}(x)}{\mathsf{std}(X)}$$

Strategies for Encoding Features

Scalar Features:

- Min-max normalization

$$x_{\text{norm}} = \frac{x - \min(X)}{\max(X) - \min(X)}$$

- Standardization

$$x_{\text{std}} = \frac{x - \text{mean}(x)}{\text{std}(X)}$$

Important for regularized regression!

Strategies for Encoding Features

Ordinal Features:

- Ordered values, but differences between values are not meaningful

Strongly disagree	Disagree	Neutral	Agree	Strongly agree
1	2	3	4	5

Strategies for Encoding Features

Ordinal Features:

- Ordered values, but differences between values are not meaningful

Strongly disagree	Disagree	Neutral	Agree	Strongly agree
1	2	3	4	5

- Thermometer code

Strongly disagree	Disagree	Neutral	Agree	Strongly agree
1,0,0,0,0	1,1,0,0,0	1,1,1,0,0	1,1,1,1,0	1,1,1,1,1

Strategies for Encoding Features

Discrete (Categorical) Features:

- School ∈ {`MIT, Harvard, Caltech, …`}
- Job ∈ {`nurse, admin, pharmacist, doctor, social worker`}

Strategies for Encoding Features

Discrete (Categorical) Features:

- School ∈ {`MIT, Harvard, Caltech, …`}
- Job ∈ {`nurse, admin, pharmacist, doctor, social worker`}

"One-hot" Encoding

	ϕ_i	ϕ_{i+1}	ϕ_{i+2}	ϕ_{i+3}	ϕ_{i+4}
nurse	1	0	0	0	0
admin	0	1	0	0	0
pharmacist	0	0	1	0	0
doctor	0	0	0	1	0
social worker	0	0	0	0	1

Strategies for Encoding Features

medicines
pain
beta blockers, pain
beta blockers
none

Strategies for Encoding Features

medicines
pain
beta blockers, pain
beta blockers
none

"One-hot" encoding?

	ϕ_i	ϕ_{i+1}	ϕ_{i+2}	ϕ_{i+3}
pain	1	0	0	0
pain & beta blockers	0	1	0	0
beta blockers	0	0	1	0
no medications	0	0	0	1

Strategies for Encoding Features

medicines
pain
beta blockers, pain
beta blockers
none

Factored encoding!

	ϕ_i	ϕ_{i+1}
pain	1	0
pain & beta blockers	1	1
beta blockers	0	1
no medications	0	0

Choosing good features

How to come up with good features?
- Performanc on validation set
- Domain expertise
- Experience

Choosing good features is super important in real world machine learning!