6.036/6.862: Introduction to Machine Learning

Lecture: starts Tuesdays 9:35am (Boston time zone)

Course website: introml.odl.mit.edu

Who's talking? Prof. Tamara Broderick

Questions? Ask on Discourse: discourse.odl.mit.edu

Materials: Will all be available at course website

Today's Plan

I. (More) logistics

II. Machine learning setup

III. Linear classifiers

6.036/6.862: Introduction to Machine Learning

Lecture: starts Tuesdays 9:35am (Boston time zone)

Course website: introml.odl.mit.edu

Who's talking? Prof. Tamara Broderick

Questions? Ask on Discourse: discourse.odl.mit.edu

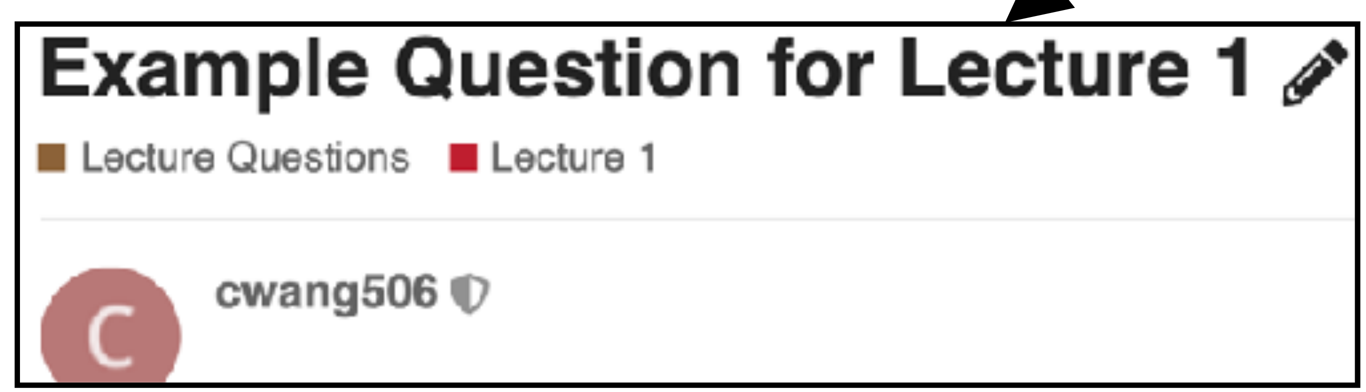
Materials: Will all be available at course website

Today's Plan

I. (More) logistics

II. Machine learning setup

III. Linear classifiers



(set "Lecture 1" category)

Is Introduction to Machine Learning (6.036/6.862) right for you?

Is Introduction to Machine Learning (6.036/6.862) right for you?

Computer Science Prerequisites

Is Introduction to Machine Learning (6.036/6.862) right for you?

Computer Science Prerequisites

- Python programming

Is Introduction to Machine Learning (6.036/6.862) right for you?

Computer Science Prerequisites

- Python programming
- Algorithms (read & understand pseudocode)

Is Introduction to Machine Learning (6.036/6.862) right for you?

Computer Science Prerequisites

- Python programming
- Algorithms (read & understand pseudocode)

Math Prerequisites

Is Introduction to Machine Learning (6.036/6.862) right for you?

Computer Science Prerequisites

- Python programming
- Algorithms (read & understand pseudocode)

Math Prerequisites

- Matrix manipulations (inverse, transpose, multiplication, etc.)

Is Introduction to Machine Learning (6.036/6.862) right for you?

Computer Science Prerequisites

- Python programming
- Algorithms (read & understand pseudocode)

Math Prerequisites

- Matrix manipulations (inverse, transpose, multiplication, etc.)
- Points and planes in dimension > 2

Is Introduction to Machine Learning (6.036/6.862) right for you?

Computer Science Prerequisites

- Python programming
- Algorithms (read & understand pseudocode)

Math Prerequisites

- Matrix manipulations (inverse, transpose, multiplication, etc.)
- Points and planes in dimension > 2
- Gradients

Is Introduction to Machine Learning (6.036/6.862) right for you?

Computer Science Prerequisites

- Python programming
- Algorithms (read & understand pseudocode)

Math Prerequisites

- Matrix manipulations (inverse, transpose, multiplication, etc.)
- Points and planes in dimension > 2
- Gradients
- Basic discrete probability (random variables, independence, conditioning, etc.)

MIT EECS
ELECTRICAL ENGINEERING AND COMPUTER SCIENCE

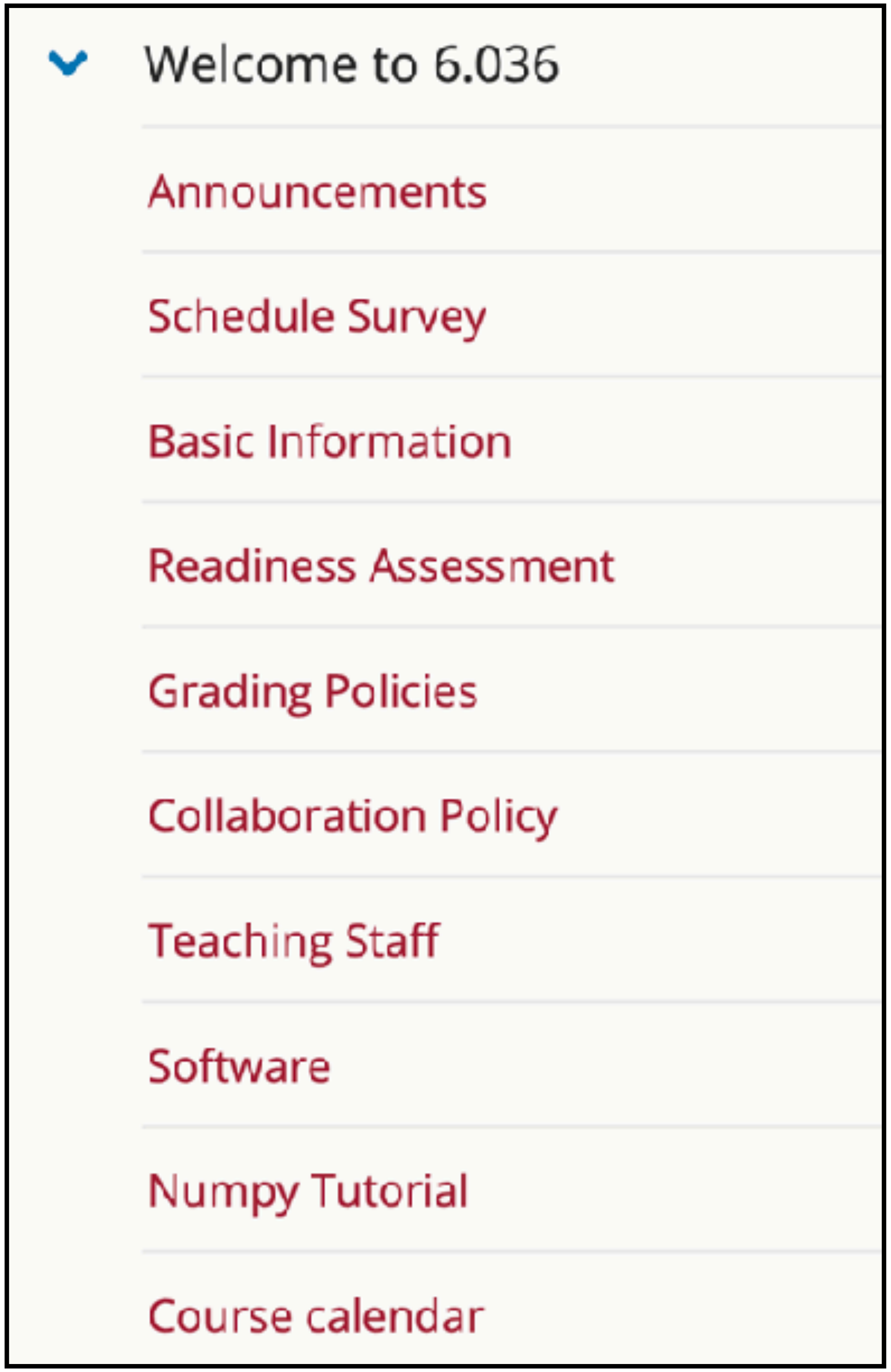
Is Introduction to Machine Learning (6.036/6.862) right for you?

Computer Science Prerequisites

- Python programming
- Algorithms (read & understand pseudocode)

Math Prerequisites

- Matrix manipulations (inverse, transpose, multiplication, etc.)
- Points and planes in dimension > 2
- Gradients
- Basic discrete probability (random variables, independence, conditioning, etc.)



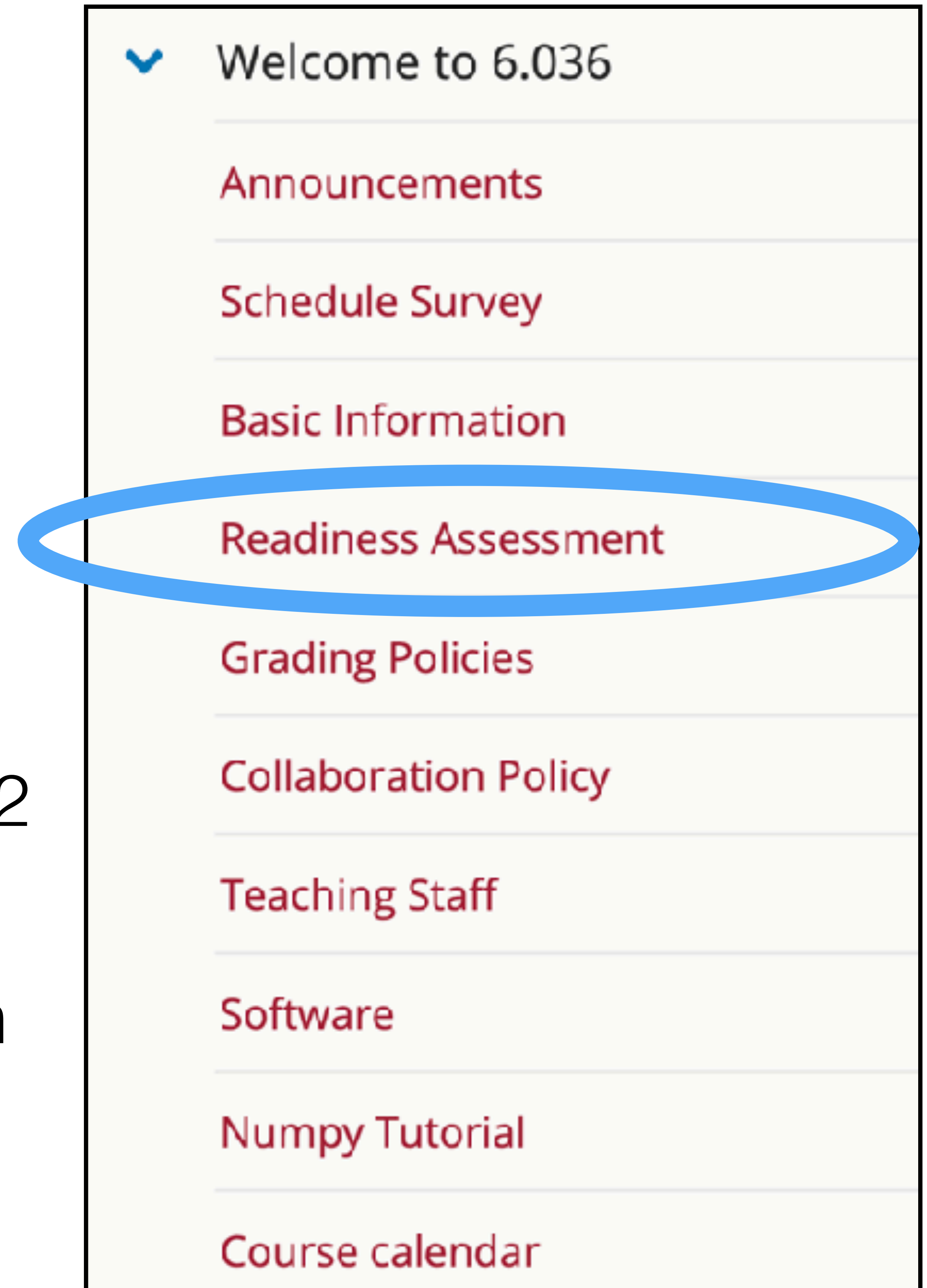
Is Introduction to Machine Learning (6.036/6.862) right for you?

Computer Science Prerequisites

- Python programming
- Algorithms (read & understand pseudocode)

Math Prerequisites

- Matrix manipulations (inverse, transpose, multiplication, etc.)
- Points and planes in dimension > 2
- Gradients
- Basic discrete probability (random variables, independence, conditioning, etc.)

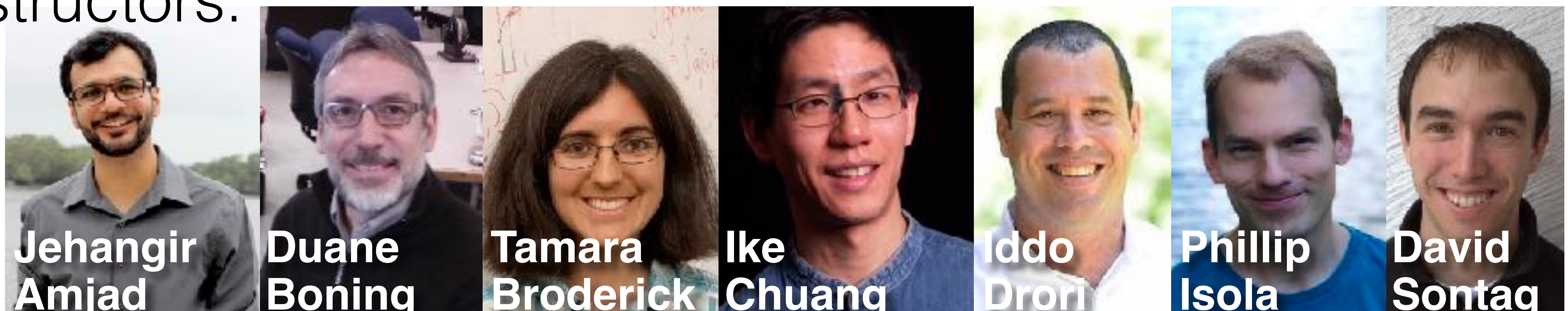


6.036/6.862: Introduction to Machine Learning

6.036/6.862: Introduction to Machine Learning, Staff

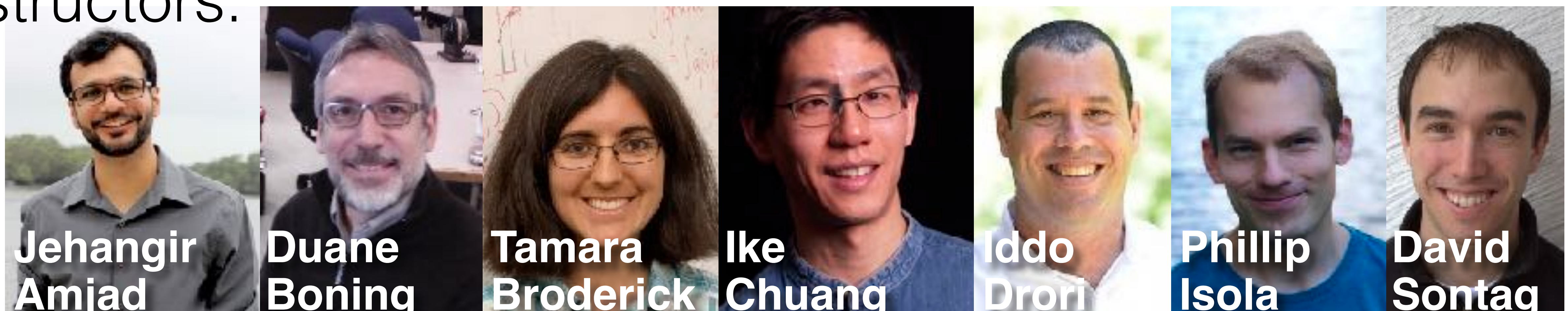
6.036/6.862: Introduction to Machine Learning, Staff

Instructors:

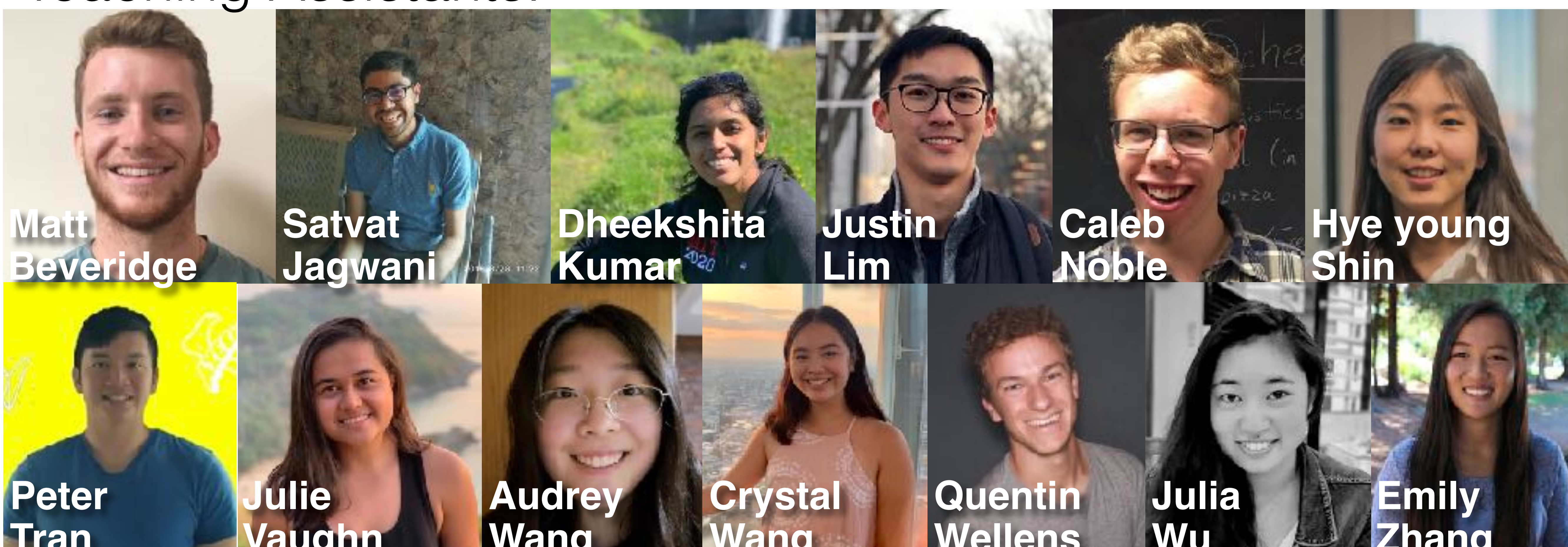


6.036/6.862: Introduction to Machine Learning, Staff

Instructors:

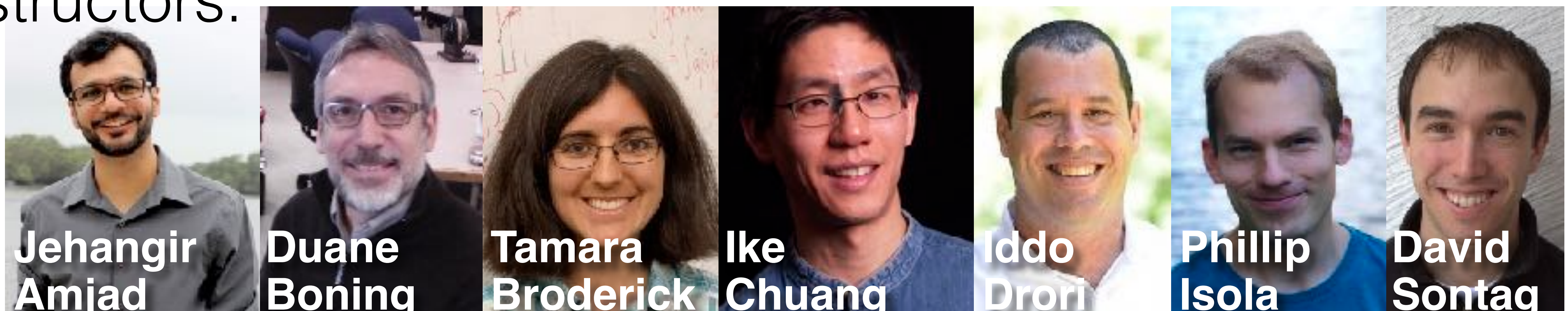


Teaching Assistants:

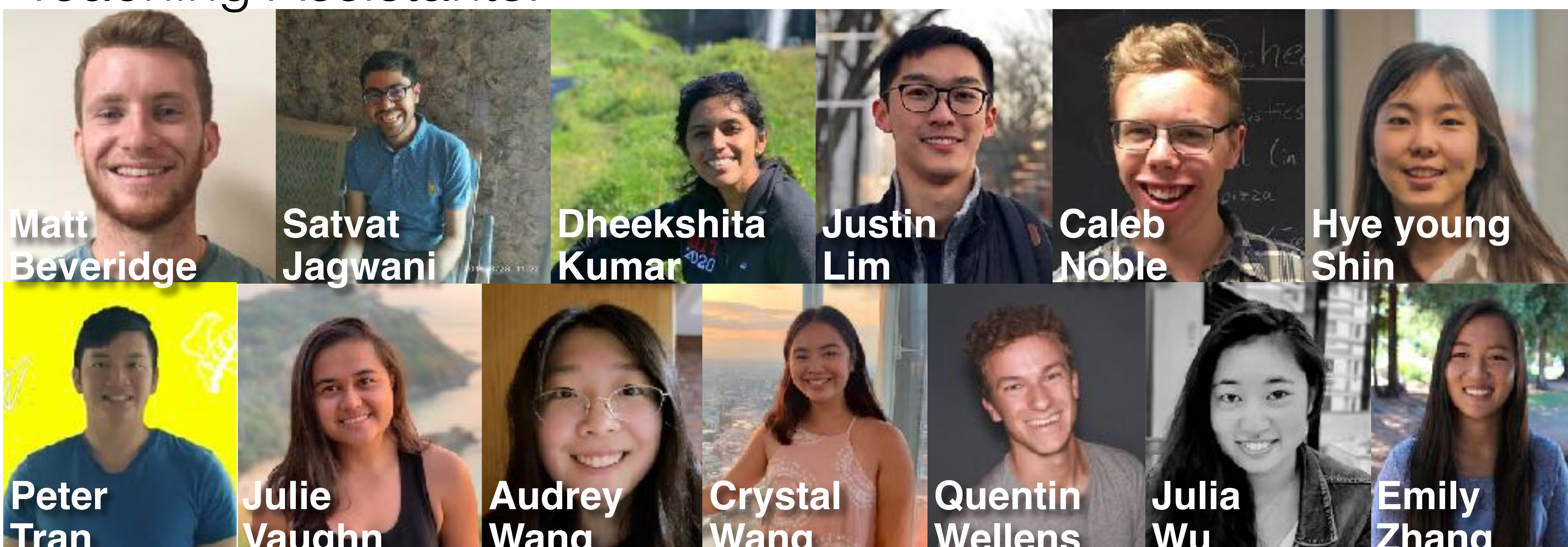


6.036/6.862: Introduction to Machine Learning, Staff

Instructors:



Teaching Assistants:

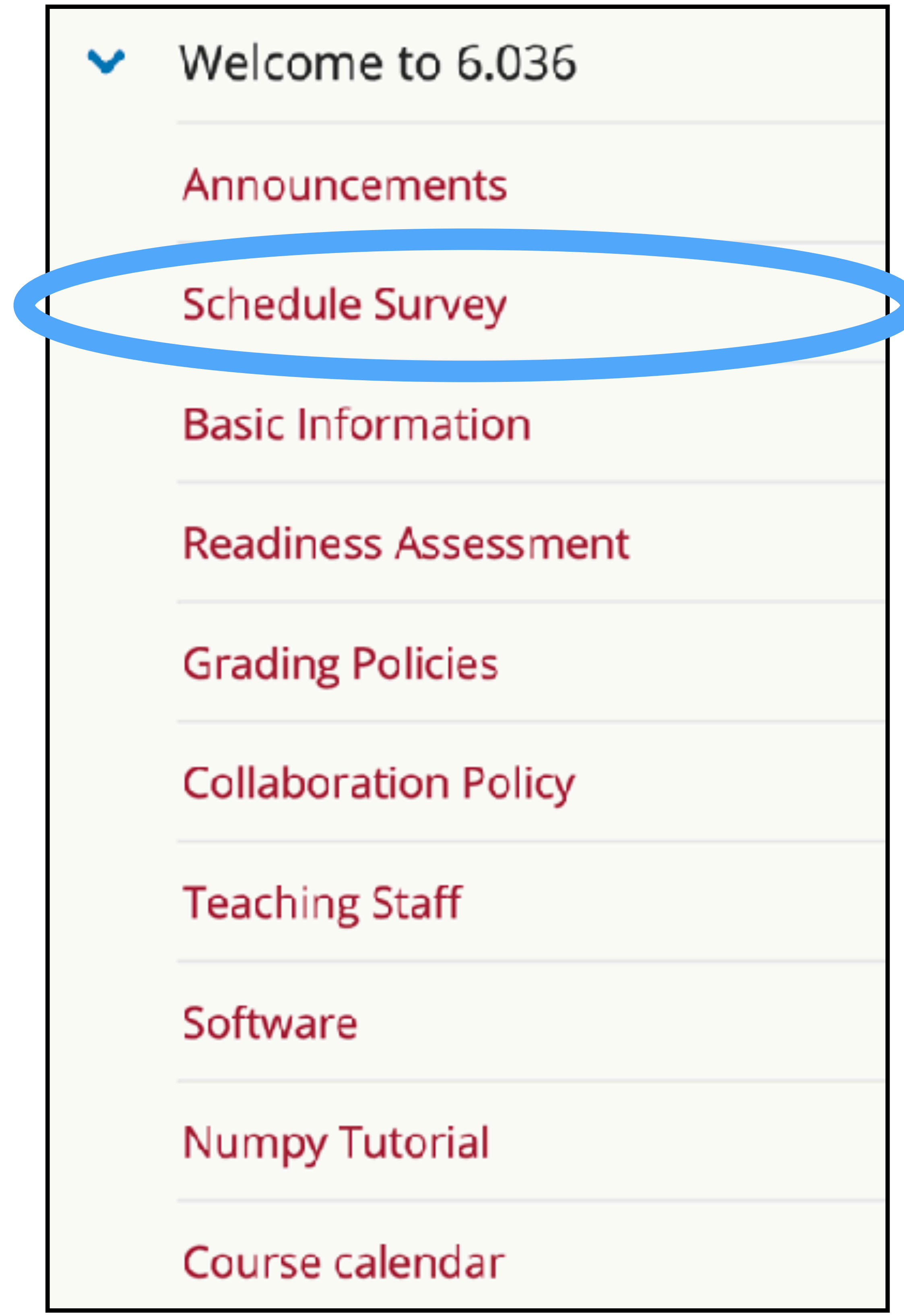


And Lab Assistants!

6.036/6.862: Introduction to Machine Learning, Weekly Plan

- Welcome to 6.036
  - Announcements
  - Schedule Survey
  - Basic Information
  - Readiness Assessment
  - Grading Policies
  - Collaboration Policy
  - Teaching Staff
  - Software
  - Numpy Tutorial
  - Course calendar

6.036/6.862: Introduction to Machine Learning, Weekly Plan



6.036/6.862: Introduction to Machine Learning, Weekly Plan

Welcome to 6.036

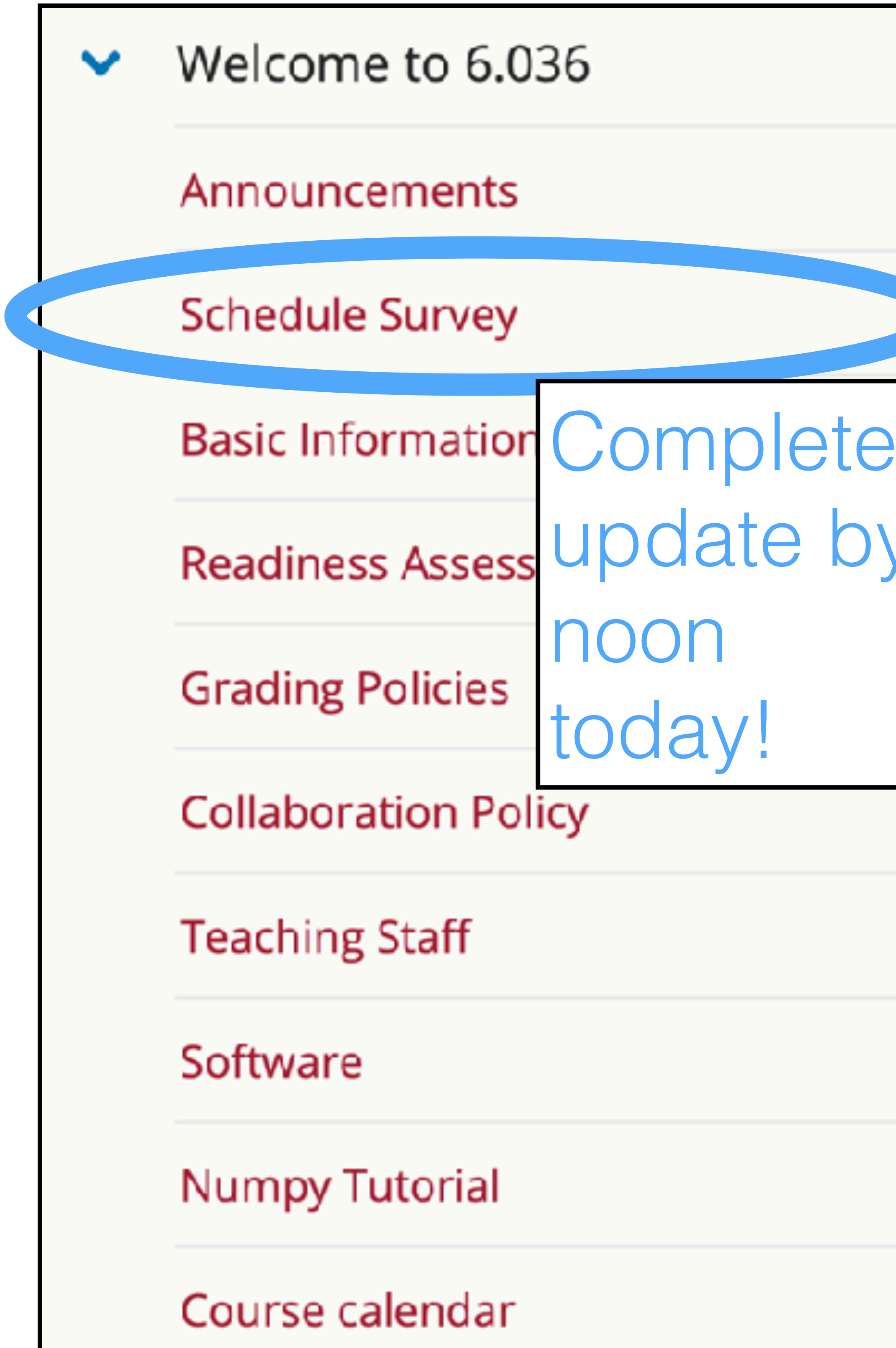
- Announcements
- Schedule Survey
- Basic Information
- Readiness Assess
- Grading Policies
- Collaboration Policy
- Teaching Staff
- Software
- Numpy Tutorial
- Course calendar

Complete/ update by noon today!

MIT EECS ELECTRICAL ENGINEERING AND COMPUTER SCIENCE

6.036/6.862: Introduction to Machine Learning, Weekly Plan

- **Lecture** + course notes



Complete/ update by noon today!

6.036/6.862: Introduction to Machine Learning, Weekly Plan

- **Lecture** + course notes
- **Exercises**
  - Due 9am before lecture

Welcome to 6.036

- Announcements
- Schedule Survey
- Basic Information
- Readiness Assess
- Grading Policies
- Collaboration Policy
- Teaching Staff
- Software
- Numpy Tutorial
- Course calendar

Complete/update by noon today!

6.036/6.862: Introduction to Machine Learning, Weekly Plan

- **Lecture** + course notes
- **Exercises**
  - Due 9am before lecture
- **Lab** (*synchronous, required!*)

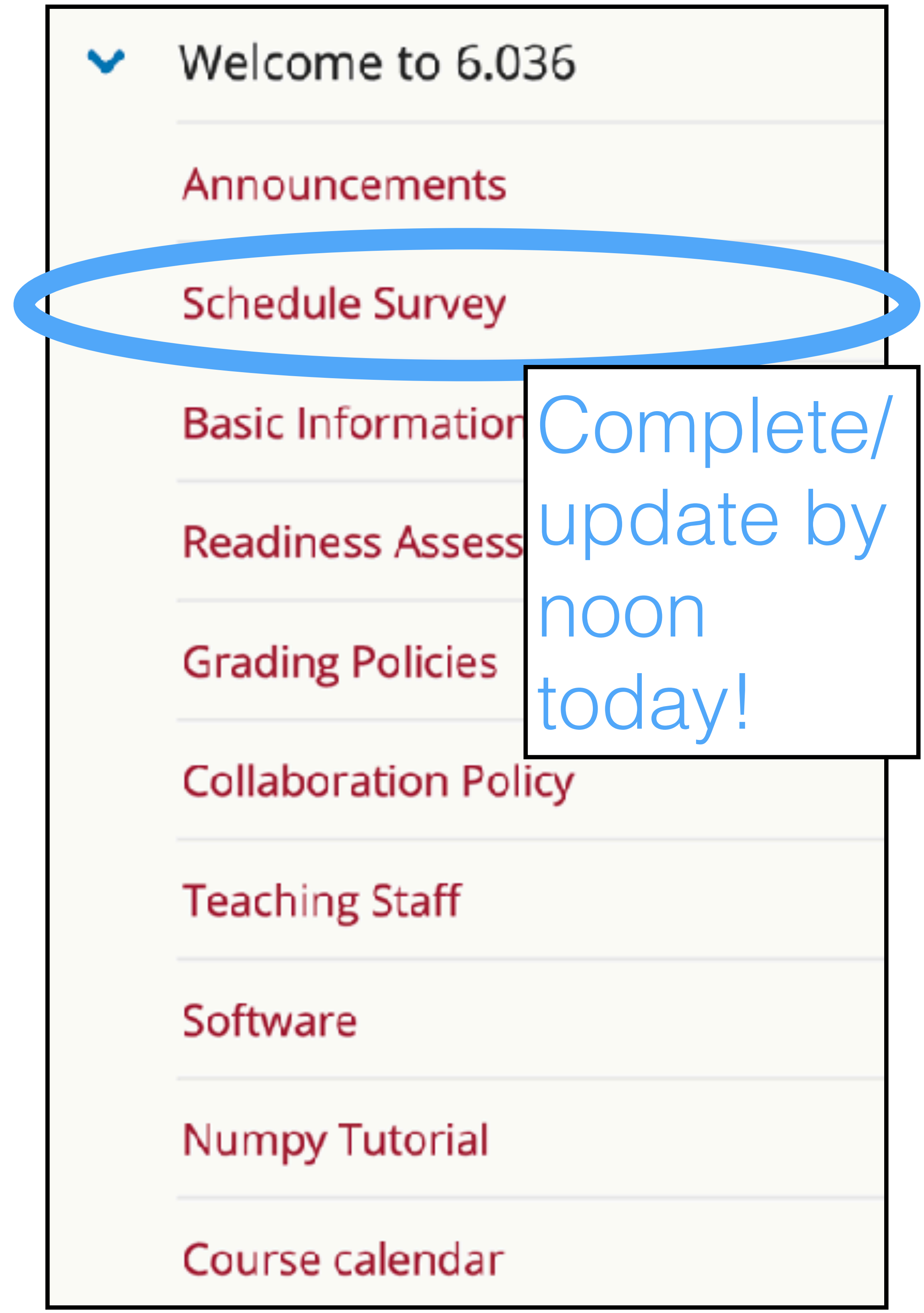
Welcome to 6.036

- Announcements
- Schedule Survey
- Basic Information
- Readiness Assess
- Grading Policies
- Collaboration Policy
- Teaching Staff
- Software
- Numpy Tutorial
- Course calendar

Complete/update by noon today!

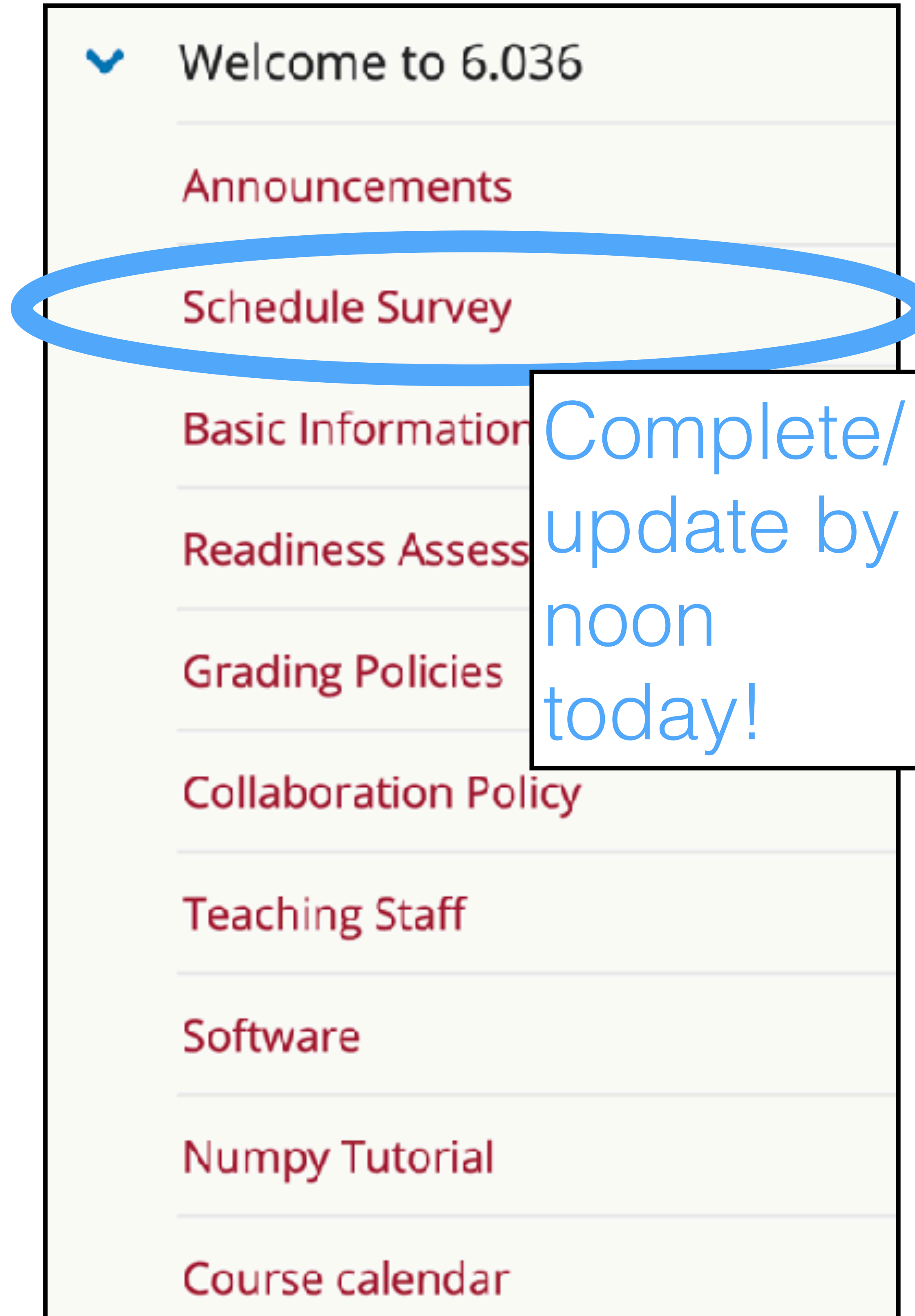
6.036/6.862: Introduction to Machine Learning, Weekly Plan

- **Lecture** + course notes
- **Exercises**
  - Due 9am before lecture
- **Lab** (*synchronous, required!*)
  - (New!) MLyPod: 10 students



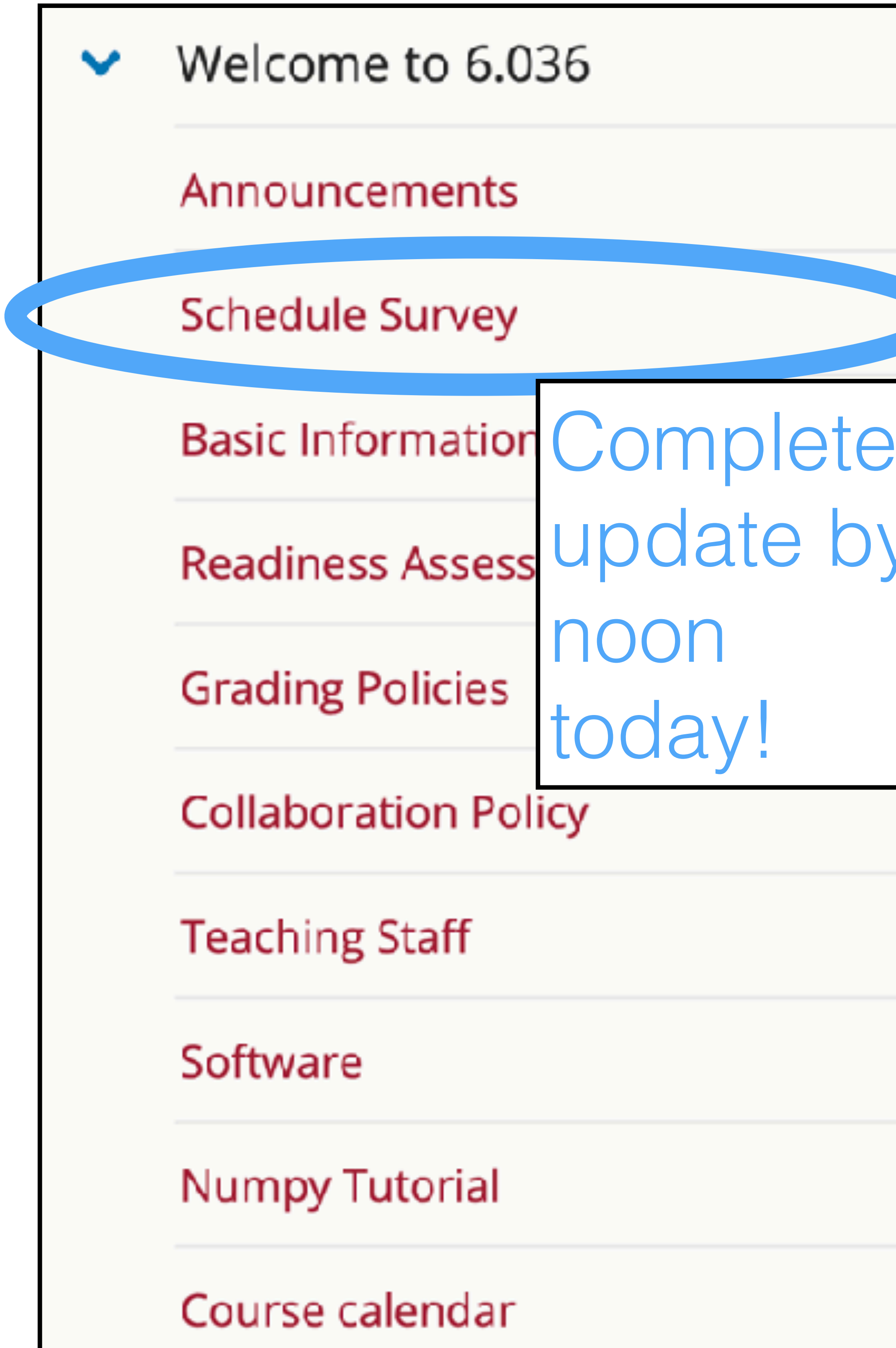
6.036/6.862: Introduction to Machine Learning, Weekly Plan

- **Lecture** + course notes
- **Exercises**
  - Due 9am before lecture
- **Lab** (*synchronous, required!*)
  - (New!) MLyPod: 10 students
  - Work in groups of 2 to 3; check off with staff



6.036/6.862: Introduction to Machine Learning, Weekly Plan

- **Lecture** + course notes
- **Exercises**
  - Due 9am before lecture
- **Lab** (*synchronous, required!*)
  - (New!) MLyPod: 10 students
  - Work in groups of 2 to 3; check off with staff
- **Homework**



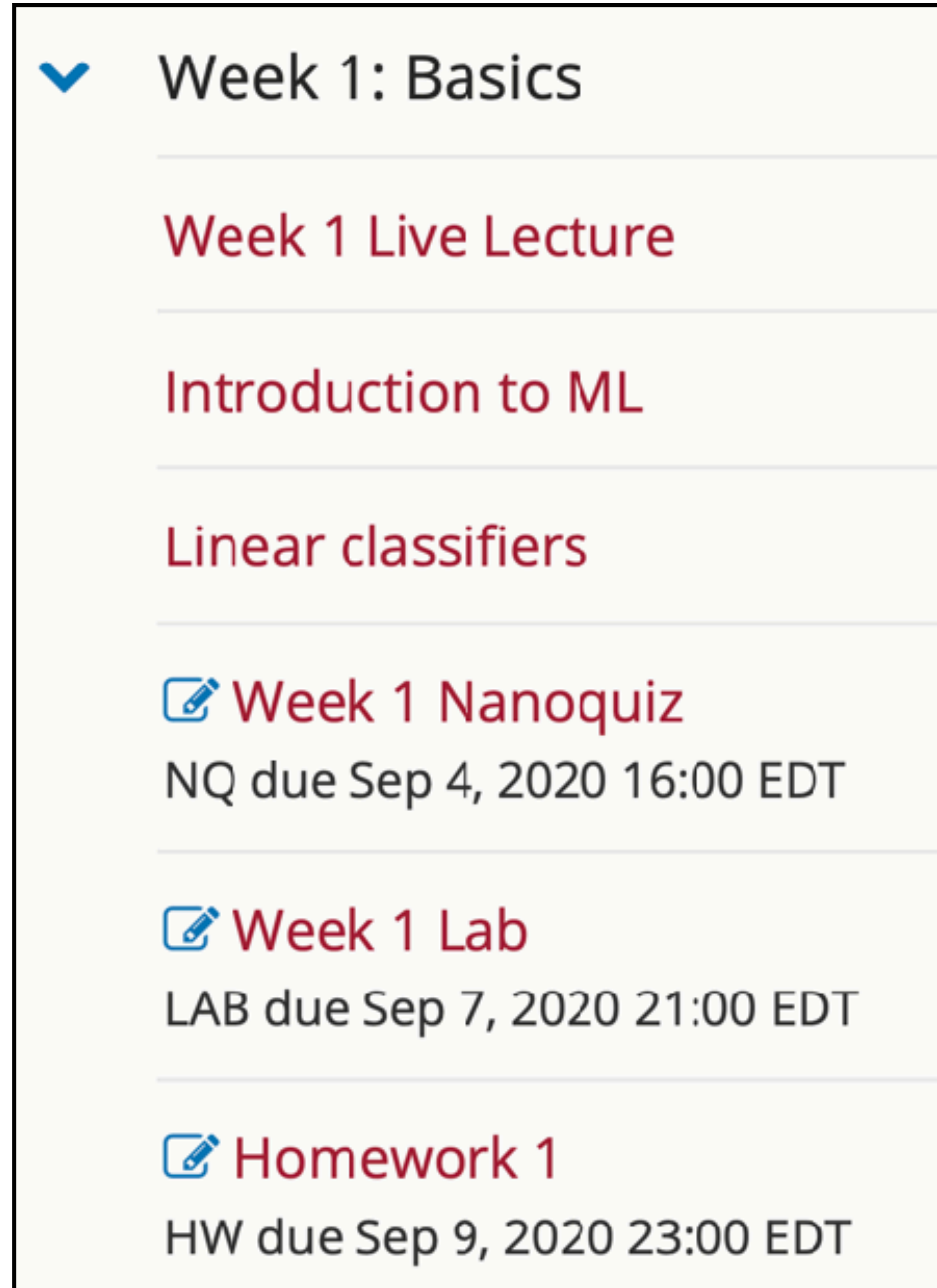
Complete/ update by noon today!

6.036/6.862: Introduction to Machine Learning, Weekly Plan

- **Lecture** + course notes
- **Exercises**
  - Due 9am before lecture
- **Lab** (*synchronous, required!*)
  - (New!) MLyPod: 10 students
  - Work in groups of 2 to 3; check off with staff
- **Homework**

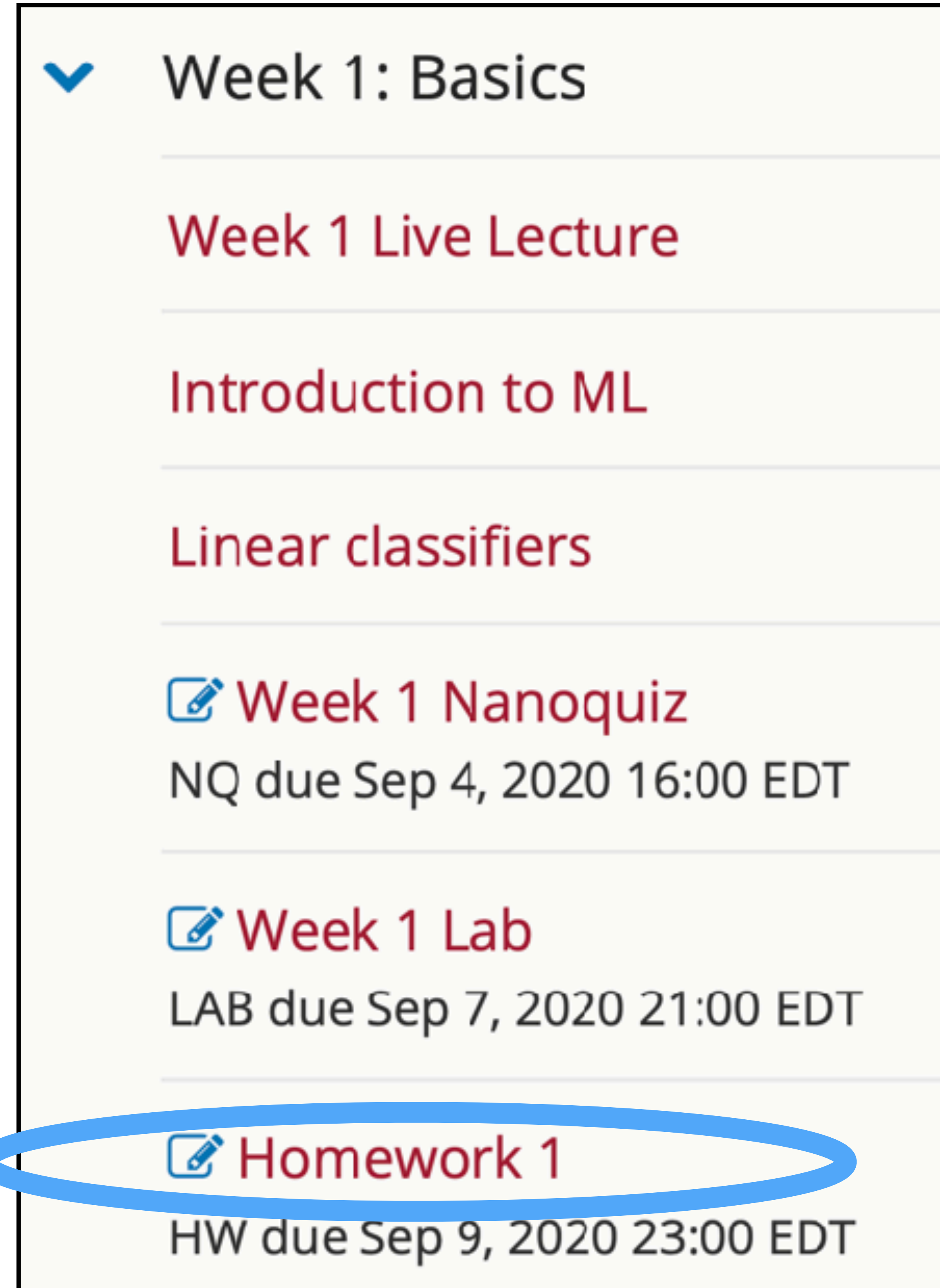
6.036/6.862: Introduction to Machine Learning, Weekly Plan

- **Lecture** + course notes
- **Exercises**
  - Due 9am before lecture
- **Lab** (*synchronous, required!*)
  - (New!) MLyPod: 10 students
  - Work in groups of 2 to 3; check off with staff
- **Homework**



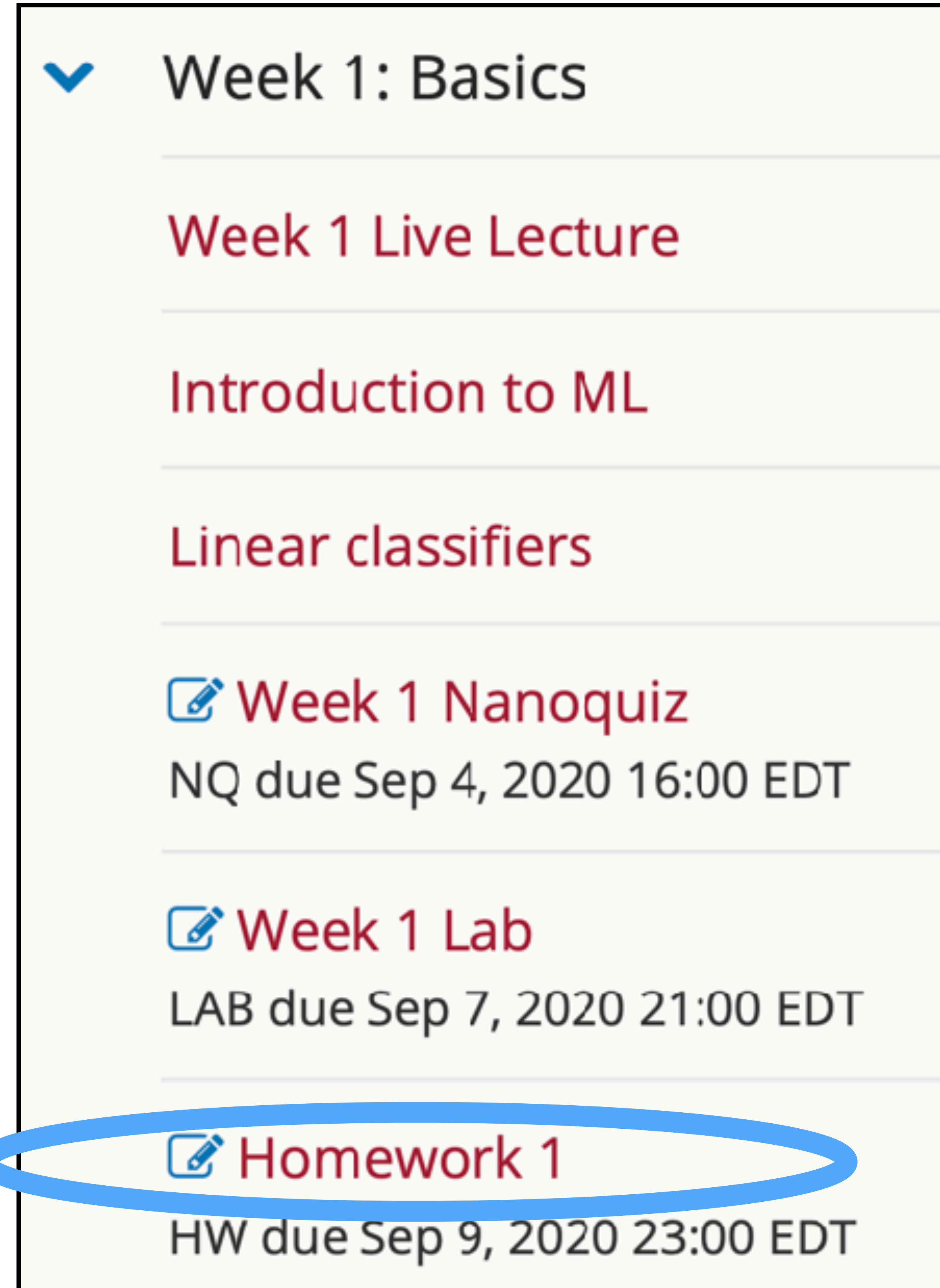
6.036/6.862: Introduction to Machine Learning, Weekly Plan

- **Lecture** + course notes
- **Exercises**
  - Due 9am before lecture
- **Lab** (*synchronous, required!*)
  - (New!) MLyPod: 10 students
  - Work in groups of 2 to 3; check off with staff
- **Homework**



6.036/6.862: Introduction to Machine Learning, Weekly Plan

- **Lecture** + course notes
- **Exercises**
  - Due 9am before lecture
- **Lab** (*synchronous, required!*)
  - (New!) MLyPod: 10 students
  - Work in groups of 2 to 3; check off with staff
- **Homework**
- **Nanoquiz** (no midterm/final)
  - Timed



6.036/6.862: Introduction to Machine Learning, Weekly Plan

- **Lecture** + course notes
- **Exercises**
  - Due 9am before lecture
- **Lab** (*synchronous, required!*)
  - (New!) MLyPod: 10 students
  - Work in groups of 2 to 3; check off with staff
- **Homework**
- **Nanoquiz** (no midterm/final)
  - Timed
- **Office hours**

Week 1: Basics

Week 1 Live Lecture

Introduction to ML

Linear classifiers

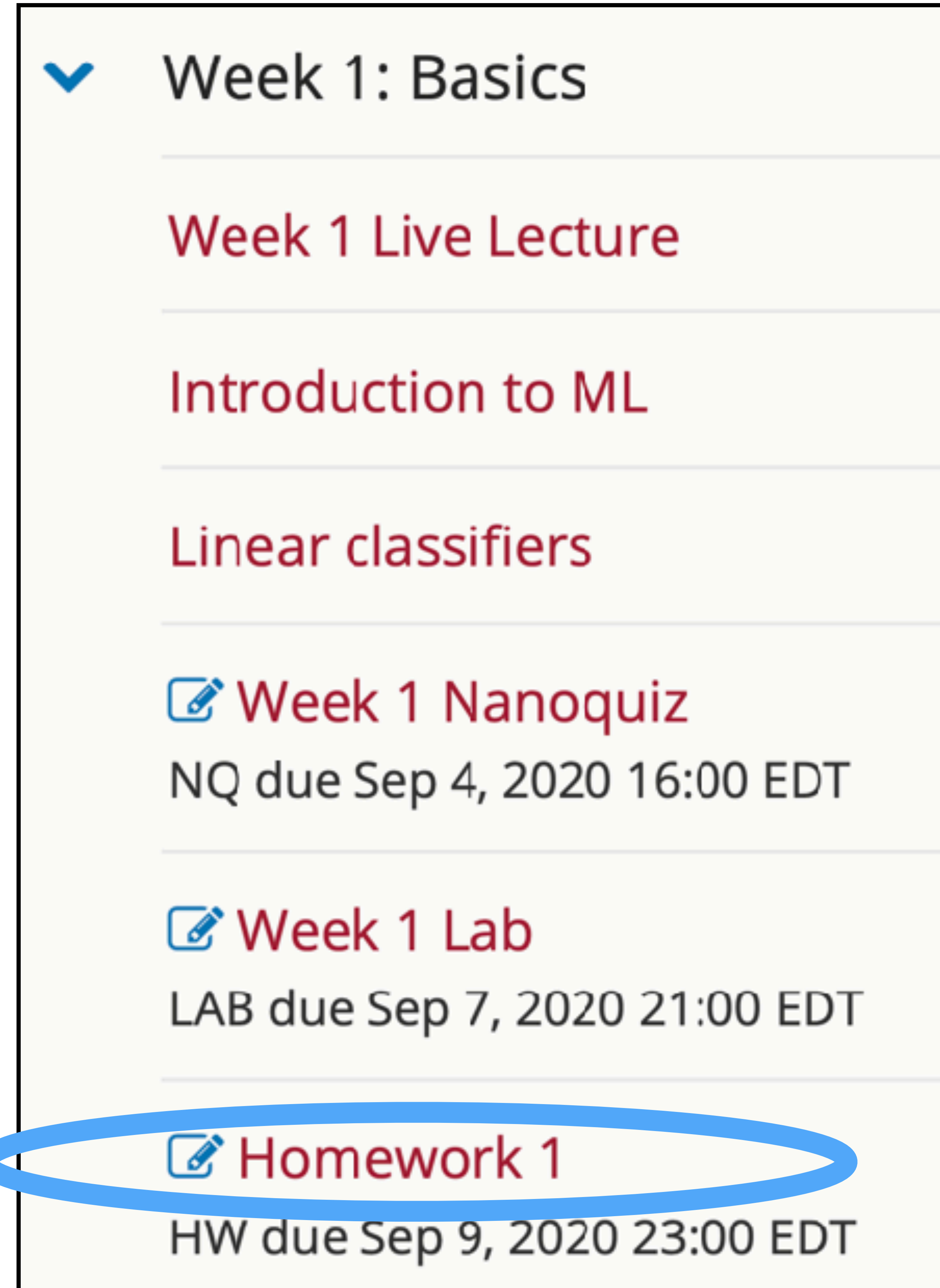
Week 1 Nanoquiz
NQ due Sep 4, 2020 16:00 EDT

Week 1 Lab
LAB due Sep 7, 2020 21:00 EDT

Homework 1
HW due Sep 9, 2020 23:00 EDT

6.036/6.862: Introduction to Machine Learning, Weekly Plan

- **Lecture** + course notes
- **Exercises**
  - Due 9am before lecture
- **Lab** (*synchronous, required!*)
  - (New!) MLyPod: 10 students
  - Work in groups of 2 to 3; check off with staff
- **Homework**
- **Nanoquiz** (no midterm/final)
  - Timed
- **Office hours**
- **6.862:** project (canvas.mit.edu)



Machine learning (ML): why & what

Machine learning (ML): why & what

Machine learning algorithm confirms 50 new exoplanets in historic first

by R. Dallon Adams in Innovation
on August 26, 2020, 9:07 AM PST

A new machine learning technique can be used to sift through massive datasets to discern exoplanets from false positives.

TechRepublic.

[https://www.techrepublic.com/article/machine-learning-algorithm-confirms-50-new-exoplanets-in-historic-first/]

Machine learning (ML): why & what


Machine learning algorithm confirms 50 new exoplanets in historic first

A new
datase

THE LANCET
Child & Adolescent Health

ARTICLES | ONLINE FIRST

A machine-learning algorithm for neonatal seizure recognition: a multicentre, randomised, controlled trial

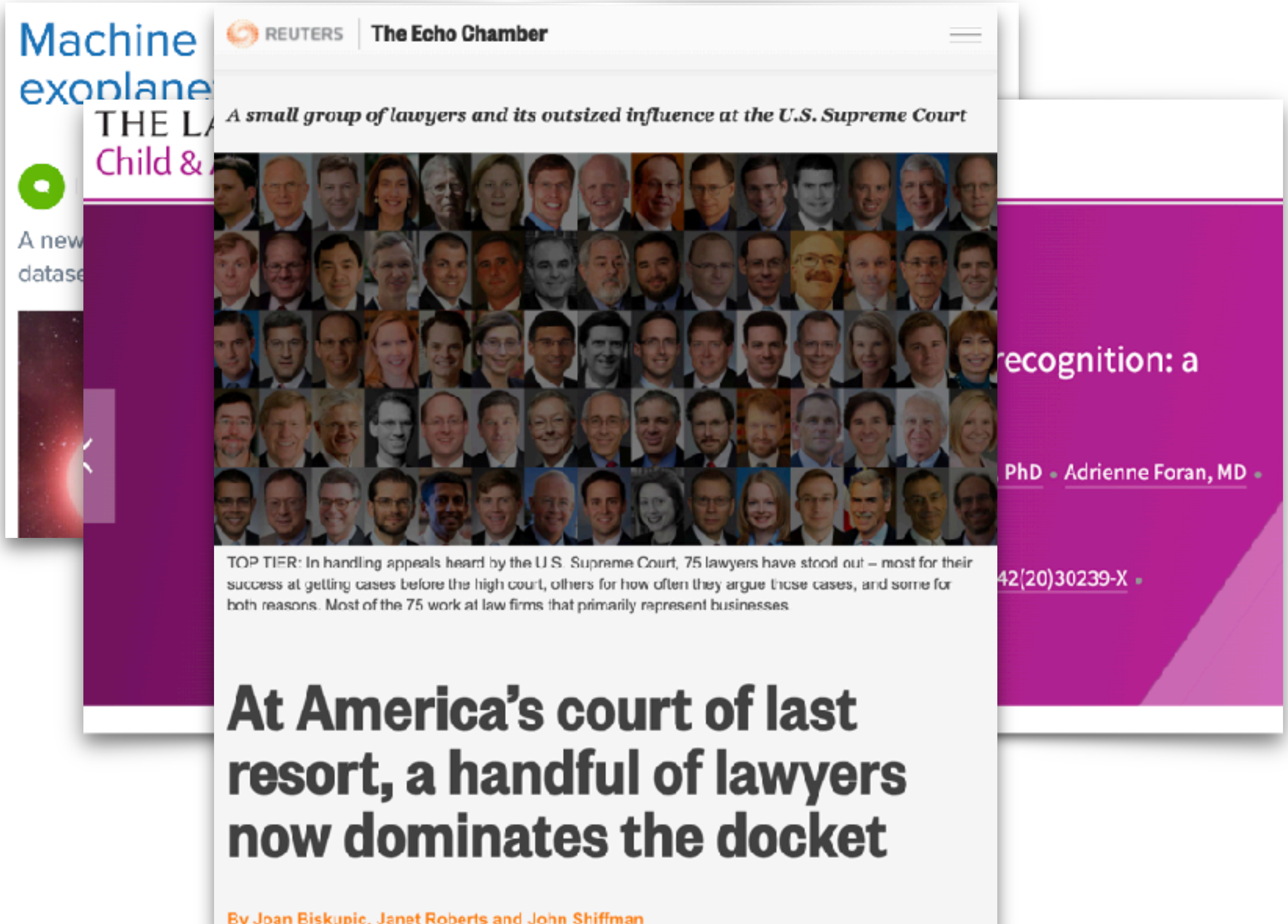
Andreea M Pavel, MD • Janet M Rennie, MD • Linda S de Vries, PhD • Mats Blennow, PhD • Adrienne Foran, MD • Divyen K Shah, MD • et al. Show all authors

Open Access • Published: August 27, 2020 • DOI: https://doi.org/10.1016/S2352-4642(20)30239-X •

Check for updates

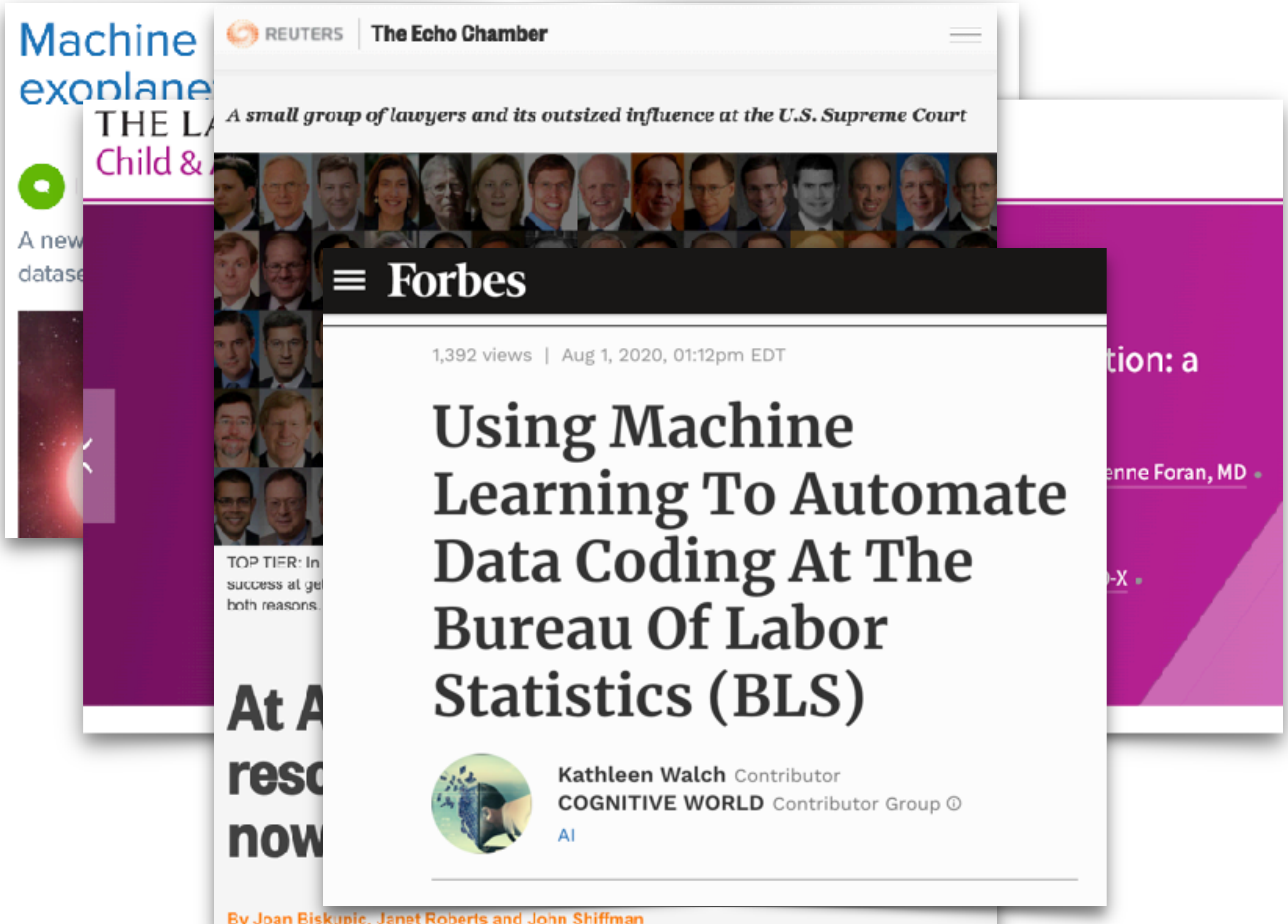
[https://www.thelancet.com/journals/lanchi/article/PIIS2352-4642(20)30239-X/fulltext]

Machine learning (ML): why & what



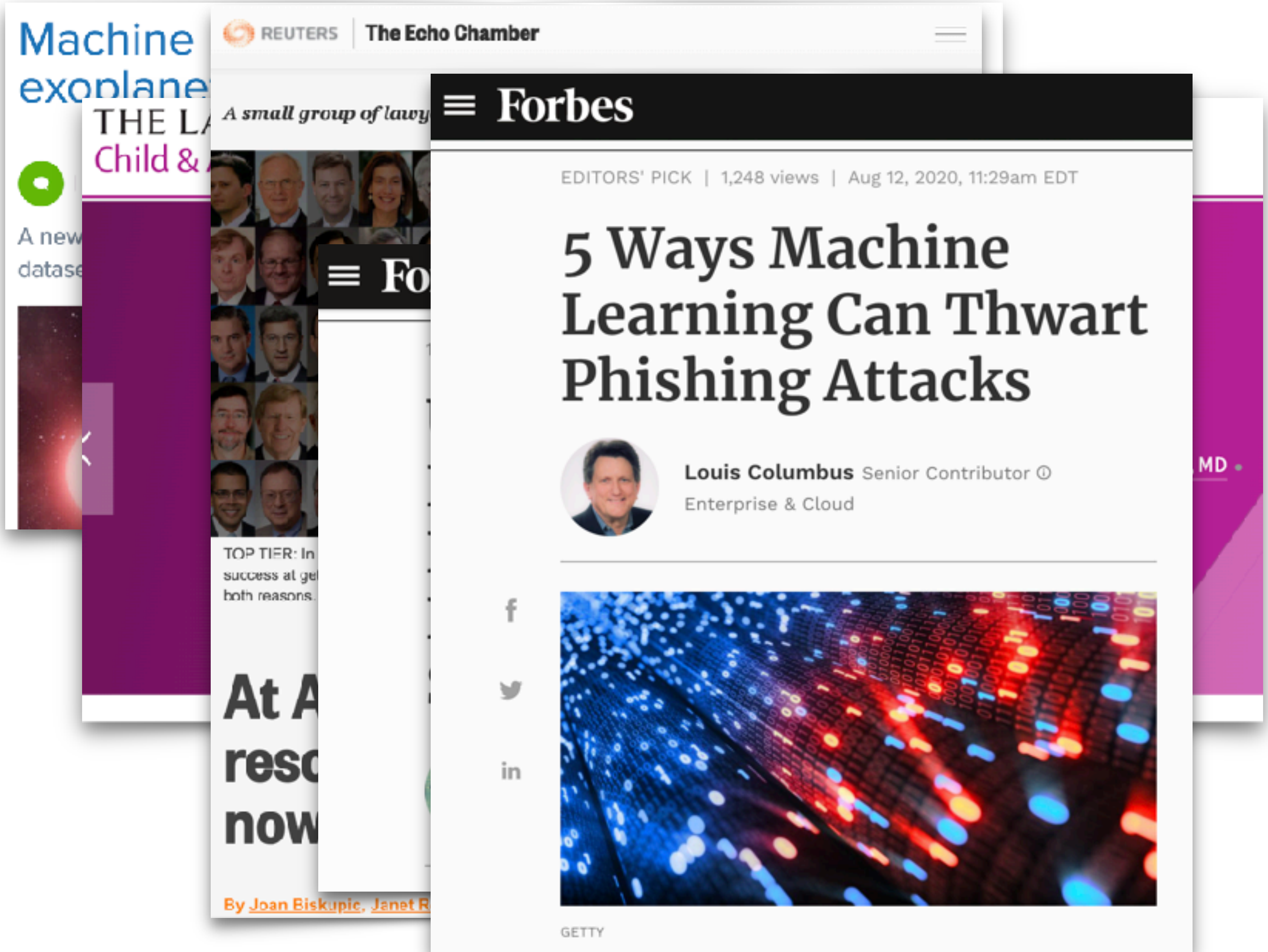
[https://www.reuters.com/investigates/special-report/scotus/]

Machine learning (ML): why & what



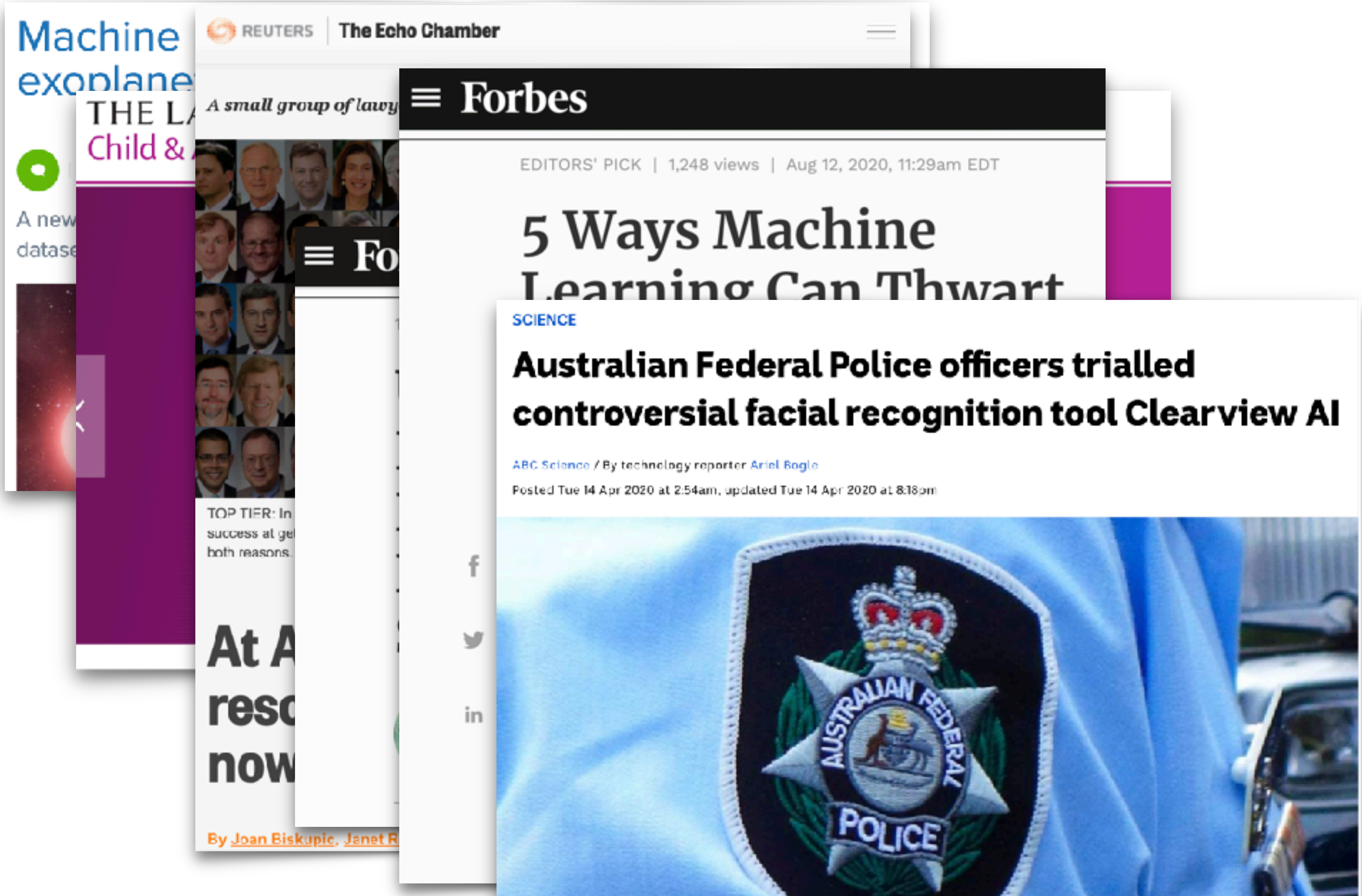
[https://www.forbes.com/sites/cognitiveworld/2020/08/01/using-machine-learning-to-automate-data-coding-at-the-bureau-of-labor-statistics-bls]

Machine learning (ML): why & what



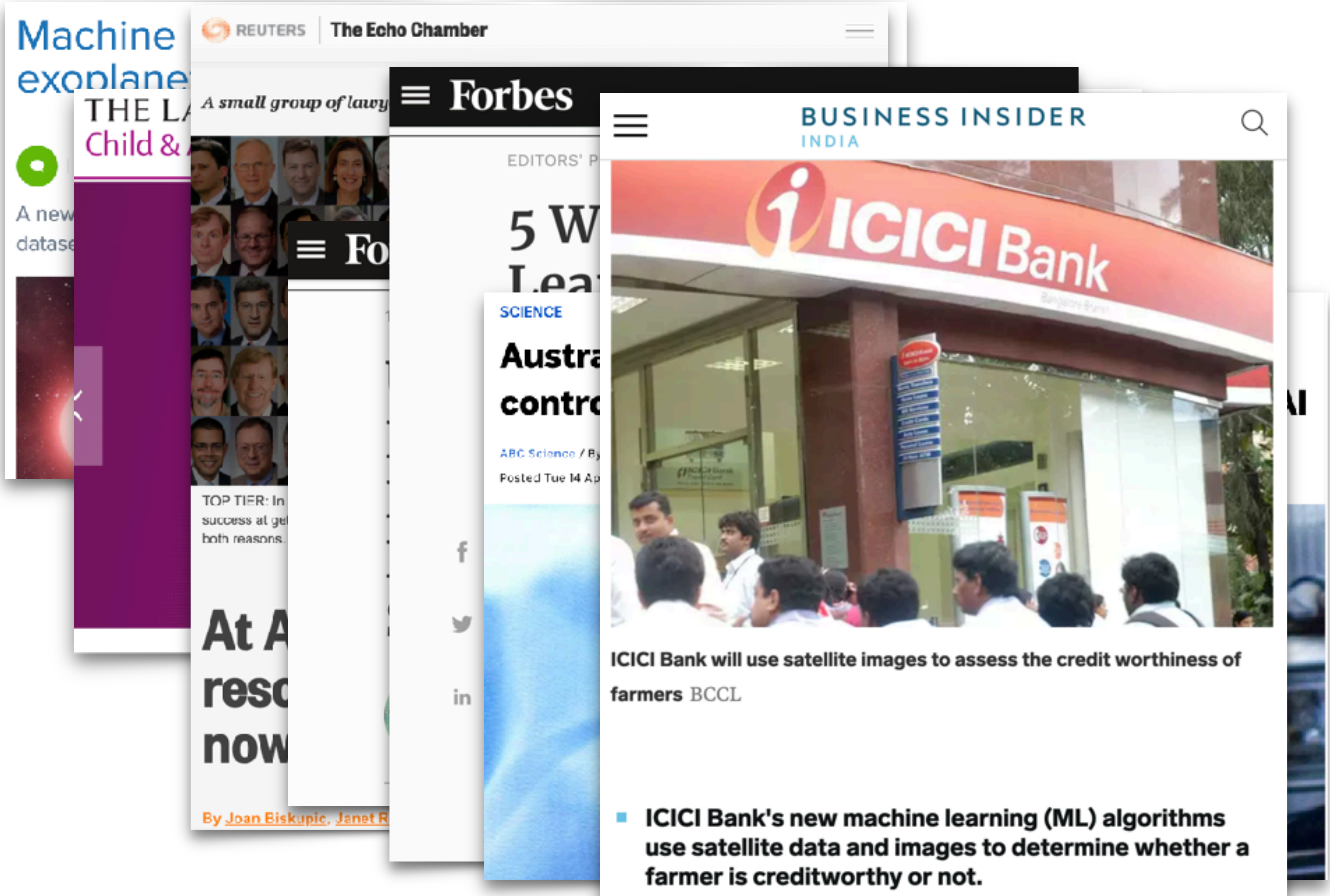
[https://www.forbes.com/sites/louiscolumbus/2020/08/12/5-ways-machine-learning-can-thwart-phishing-attacks]

Machine learning (ML): why & what



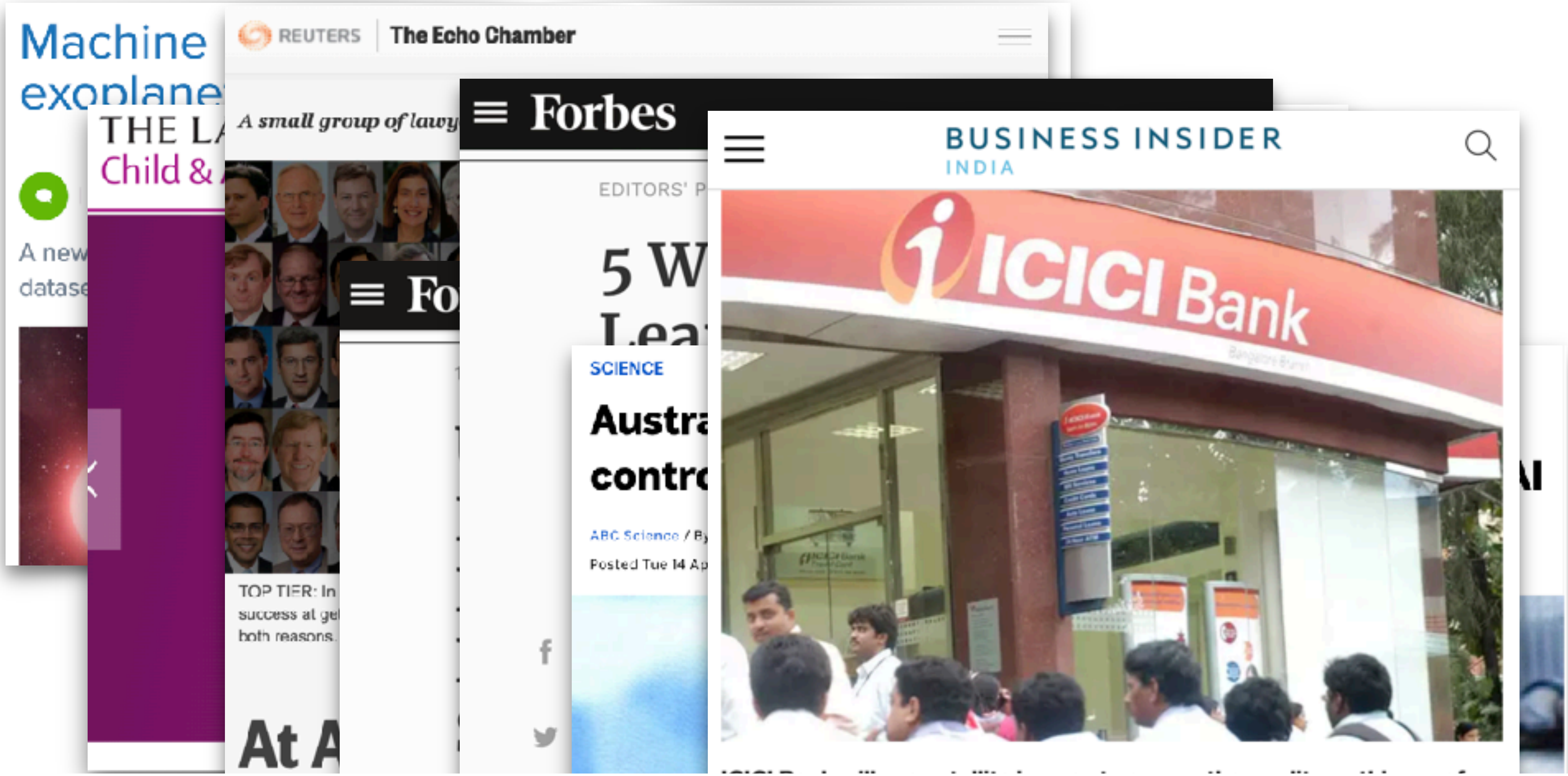
[https://www.abc.net.au/news/science/2020-04-14/clearview-ai-facial-recognition-tech-australian-federal-police/12146894]

Machine learning (ML): why & what

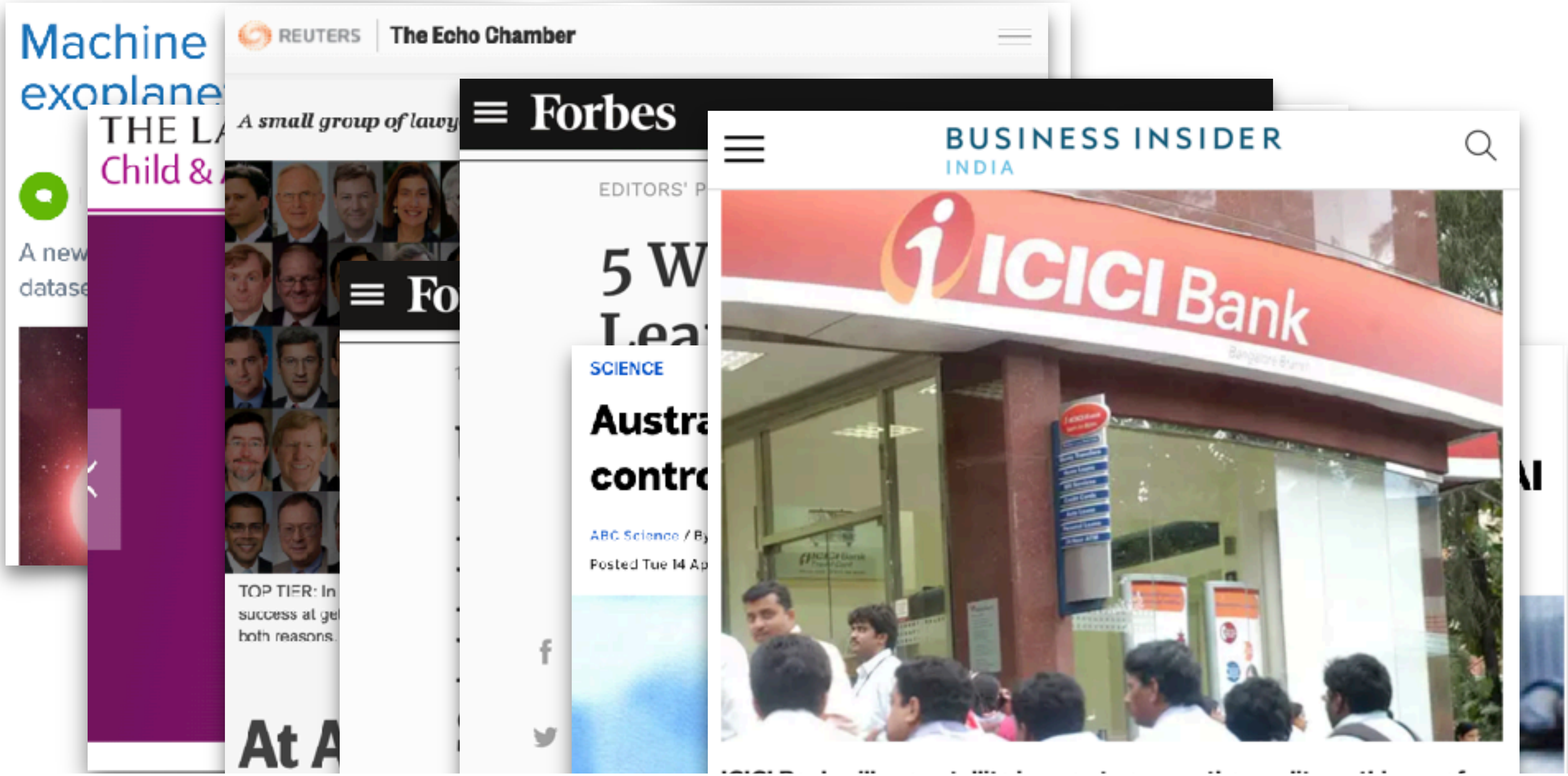


[https://www.businessinsider.in/finance/banks/news/icici-bank-is-looking-to-pump-out-more-kisan-credit-cards-with-satellite-data-and-machine-learning-at-its-side/articleshow/77741696.cms]

Machine learning (ML): why & what

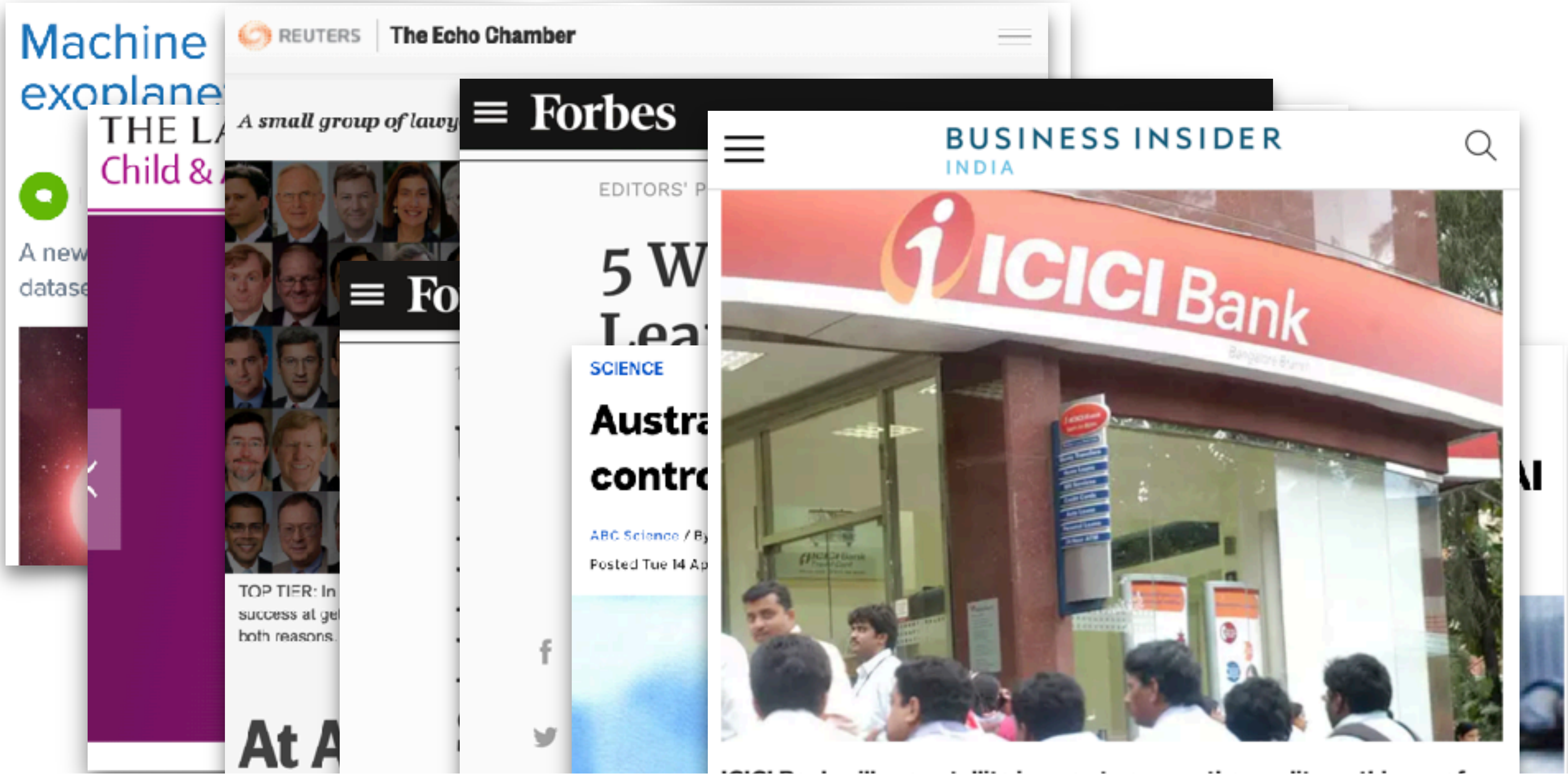


Machine learning (ML): why & what



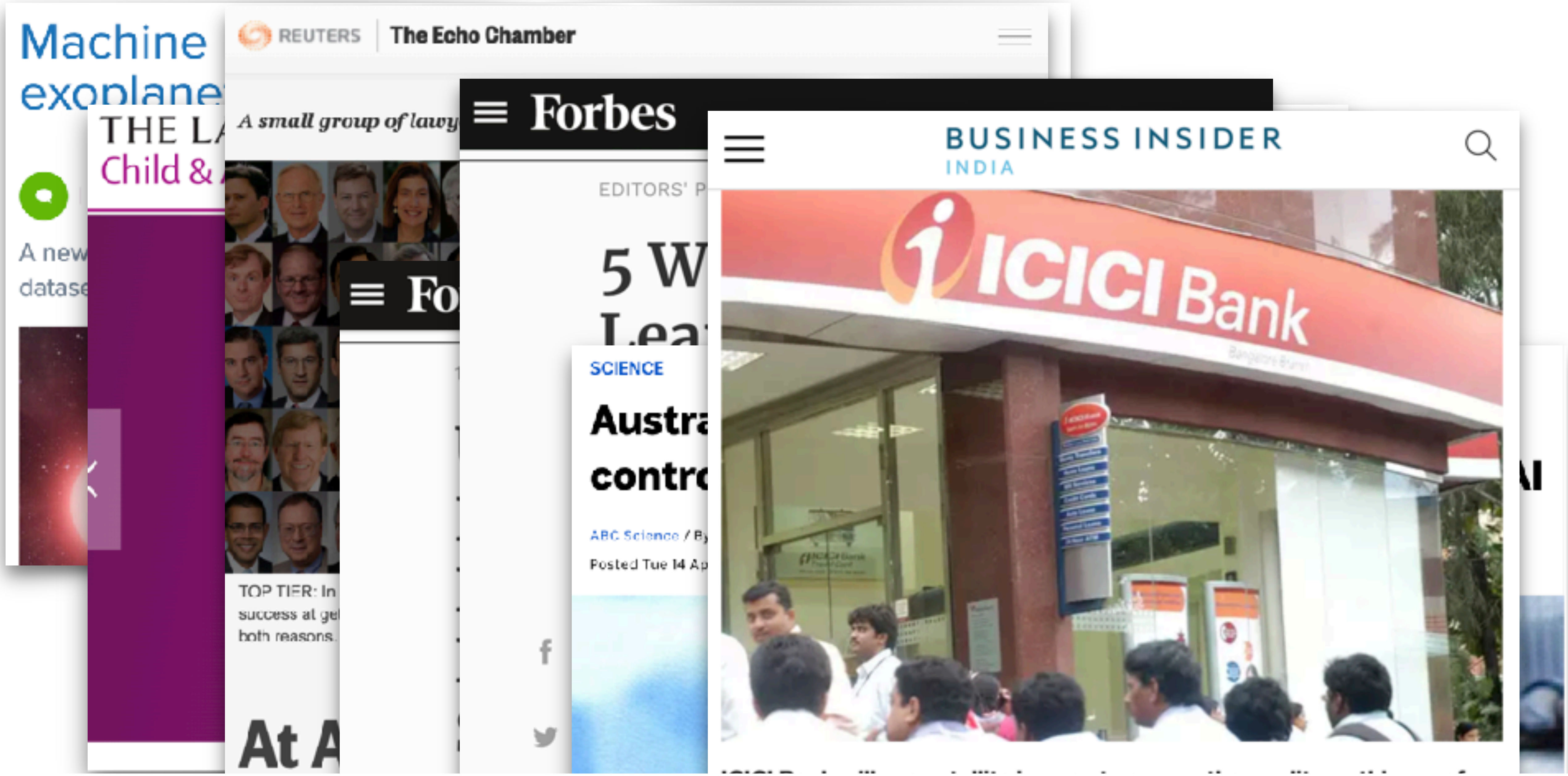
- **What is ML?**

Machine learning (ML): why & what



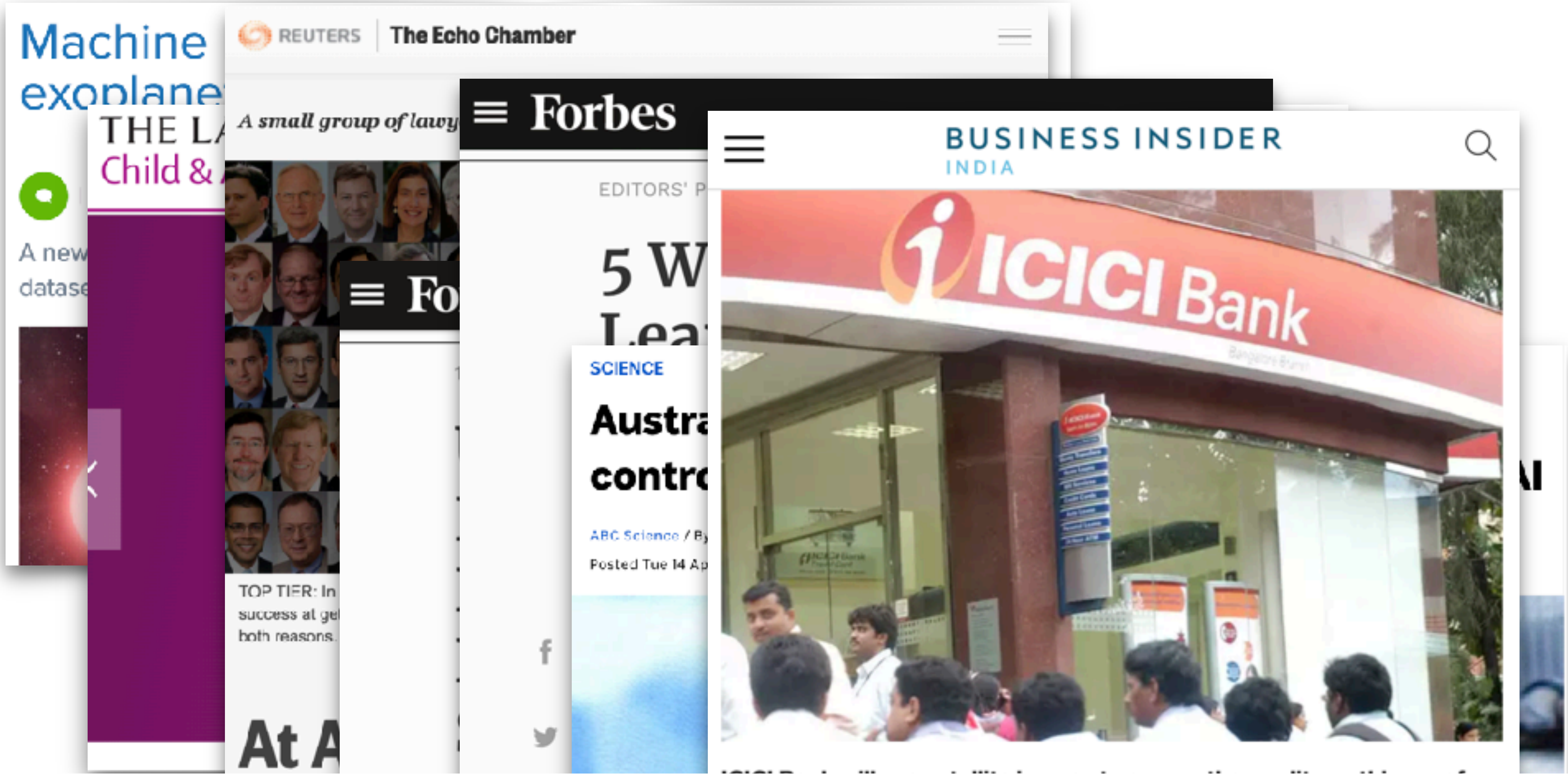
- **What is ML?** A set of methods for making decisions from data. (See the rest of the course!)

Machine learning (ML): why & what



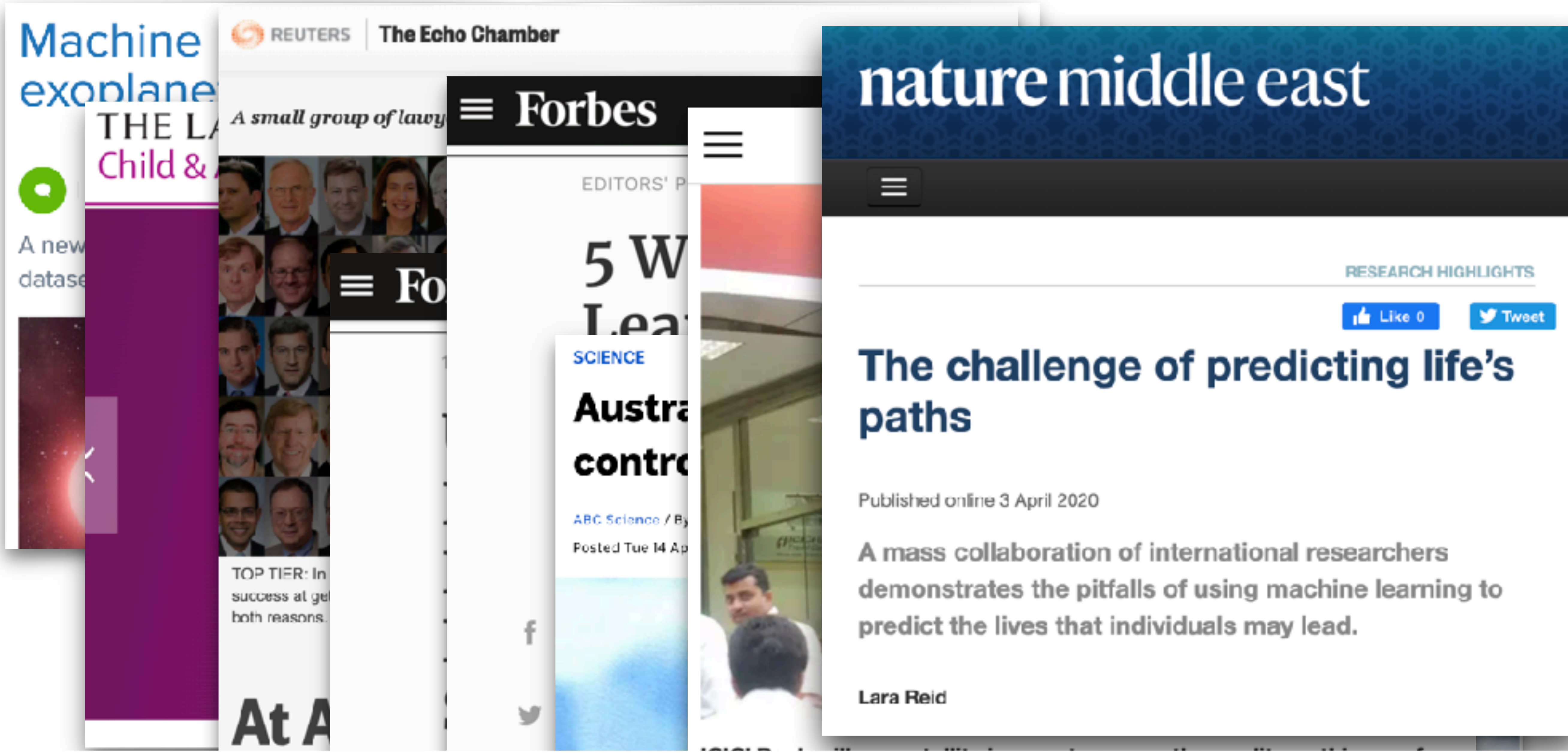
- **What is ML?** A set of methods for making decisions from data. (See the rest of the course!)
- **Why study ML?** To apply; to understand; to evaluate

Machine learning (ML): why & what



- **What is ML?** A set of methods for making decisions from data. (See the rest of the course!)
- **Why study ML?** To apply; to understand; to evaluate
- **Notes:** ML is not magic. ML is built on math.

Machine learning (ML): why & what



- **What is ML?** A set of methods for making decisions from data. (See the rest of the course!)
- **Why study ML?** To apply; to understand; to evaluate
- **Notes:** ML is not magic. ML is built on math.

[https://www.natureasia.com/en/nmiddleeast/article/10.1038/nmiddleeast.2020.46]

Getting started

Getting started

What do we have?

Getting started

What do we have? (Training) data

Getting started

What do we have? (Training) data

- *n* training data points

Getting started

What do we have? (Training) data

- *n* training data points
- For data point $i \in \{1, \ldots, n\}$

Getting started

What do we have? (Training) data

- *n* training data points
- For data point $i \in \{1, \ldots, n\}$
  - Feature vector

Getting started

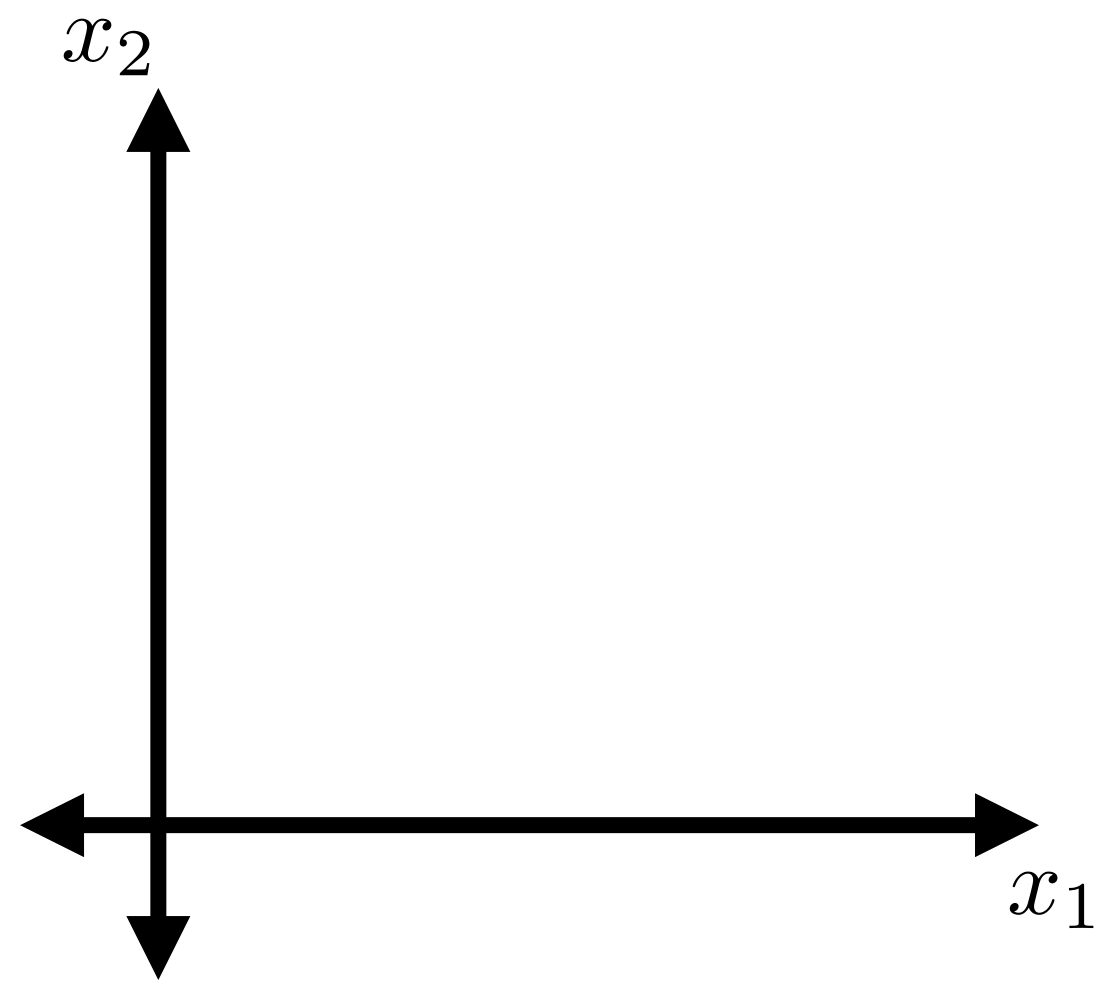
What do we have? (Training) data

- *n* training data points
- For data point $i \in \{1, \ldots, n\}$
  - Feature vector
    $x^{(i)} = (x_1^{(i)}, \ldots, x_d^{(i)})^\top \in \mathbb{R}^d$

Getting started

What do we have? (Training) data

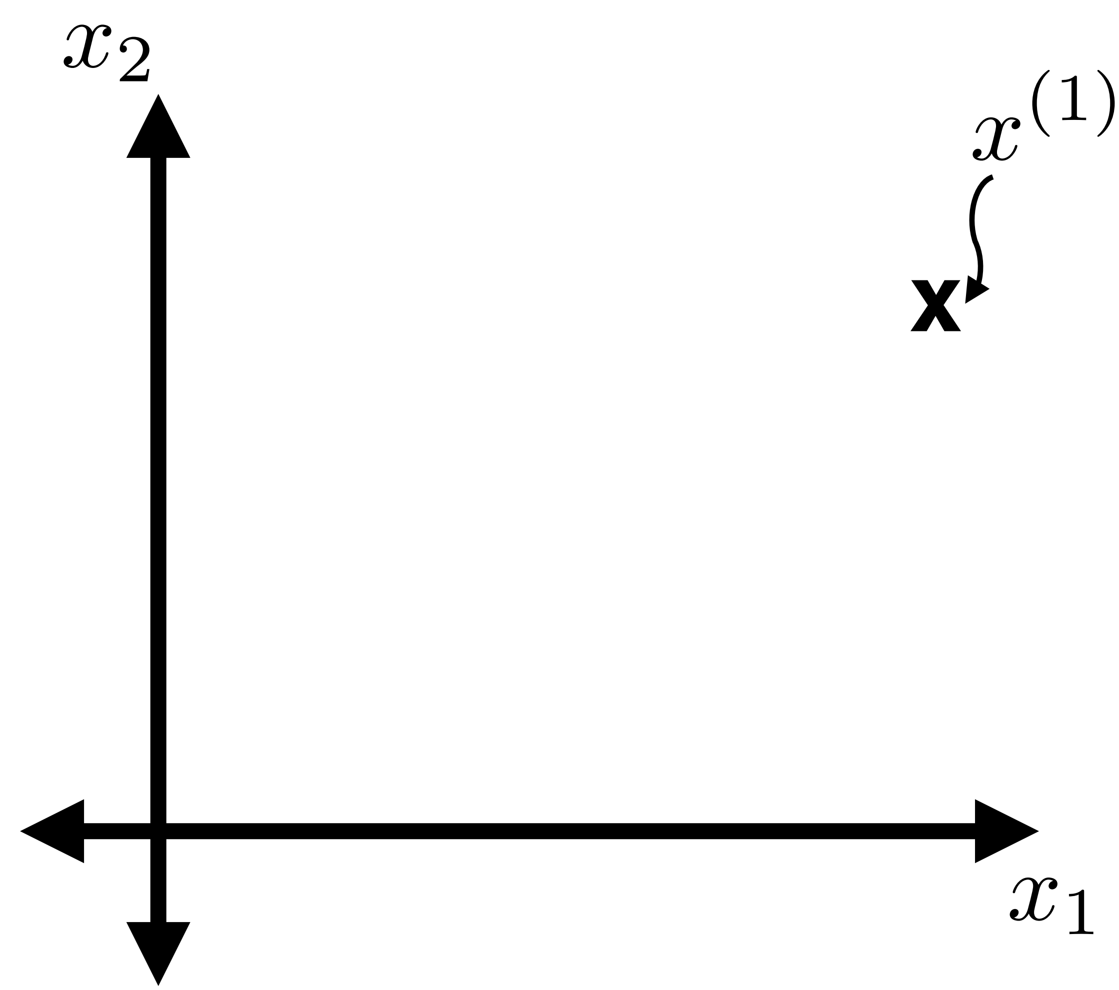
- *n* training data points
- For data point $i \in \{1, \ldots, n\}$
  - Feature vector $x^{(i)} = (x_1^{(i)}, \ldots, x_d^{(i)})^\top \in \mathbb{R}^d$



Getting started

What do we have? (Training) data

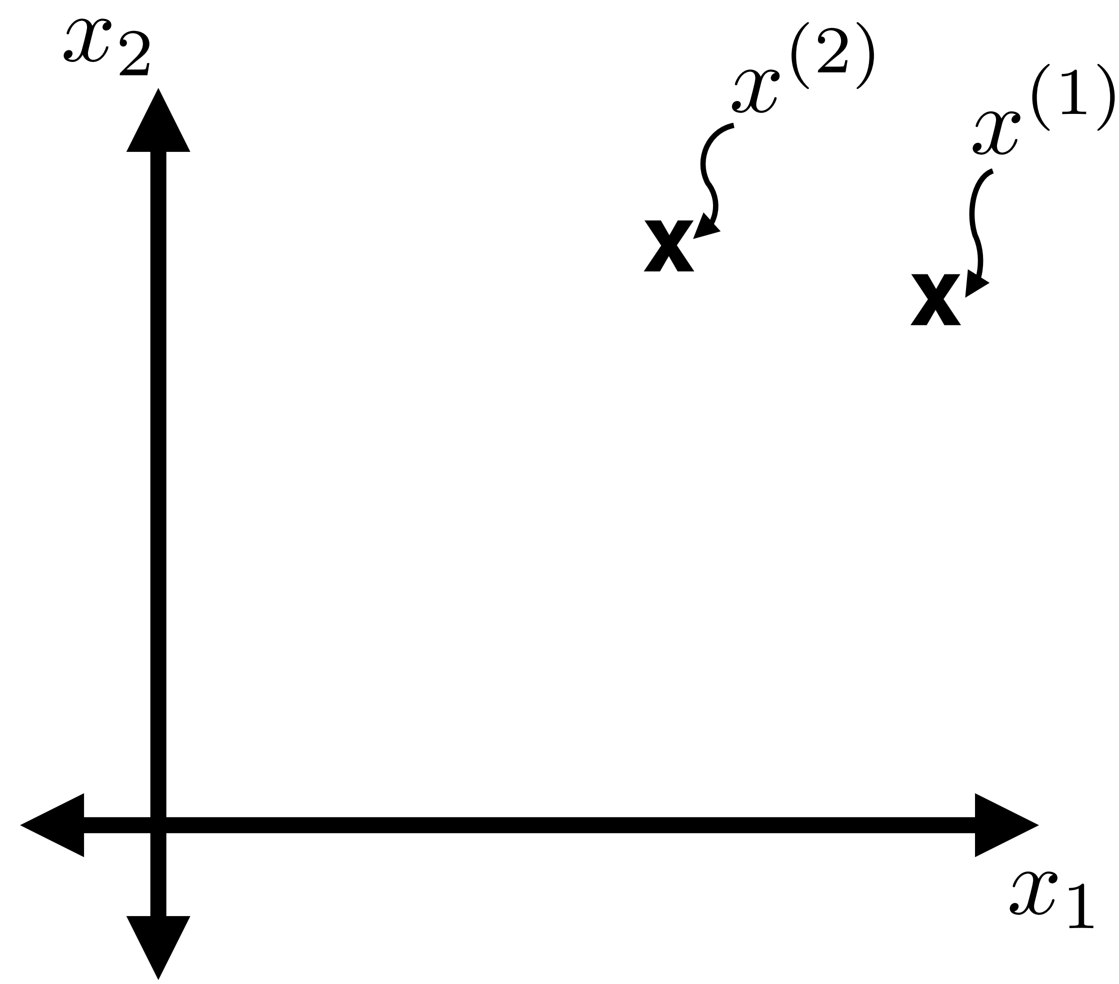
- n training data points
- For data point $i \in \{1, \ldots, n\}$
  - Feature vector $x^{(i)} = (x_1^{(i)}, \ldots, x_d^{(i)})^\top \in \mathbb{R}^d$



Getting started

What do we have? (Training) data

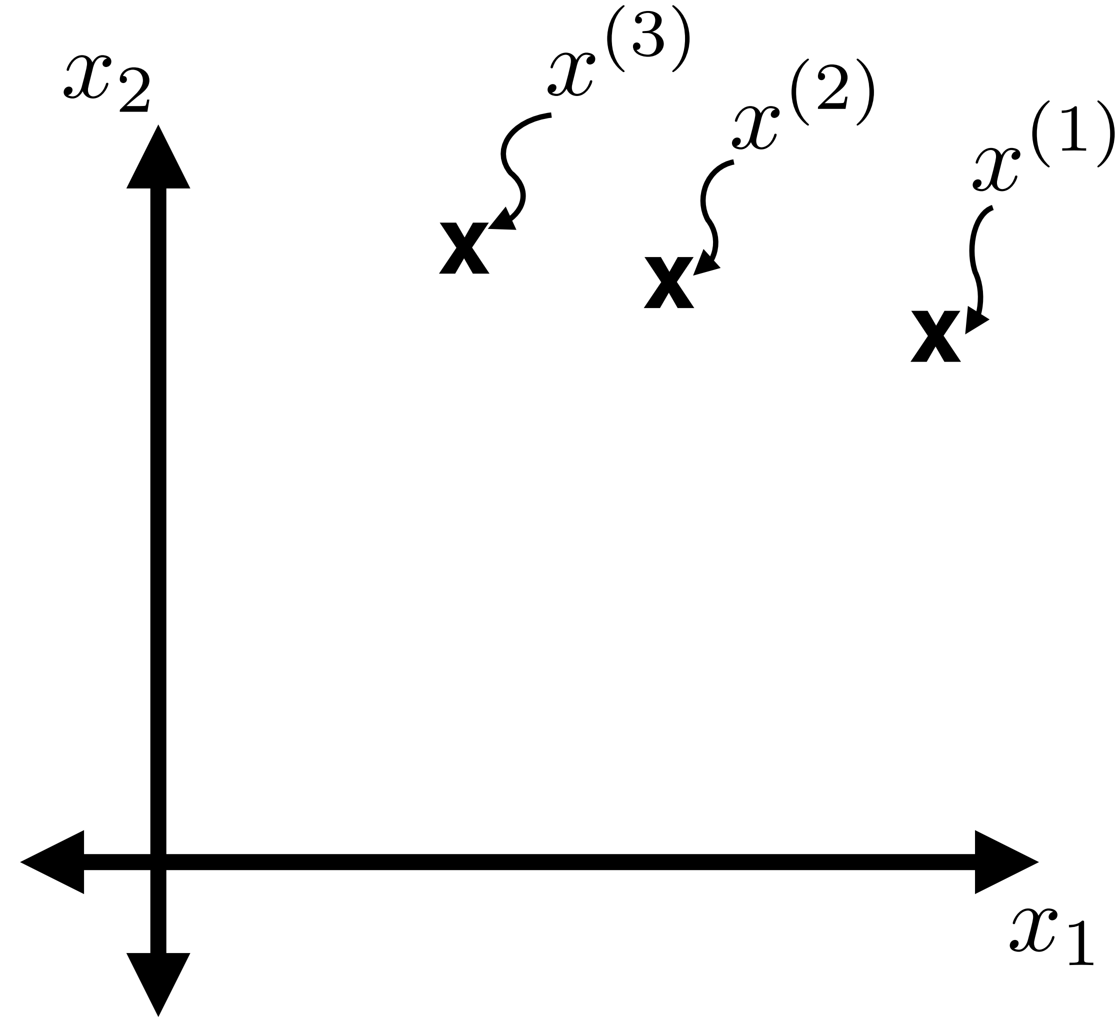
- n training data points
- For data point $i \in \{1, \ldots, n\}$
    - Feature vector
      $x^{(i)} = (x_1^{(i)}, \ldots, x_d^{(i)})^\top \in \mathbb{R}^d$



Getting started

What do we have? (Training) data

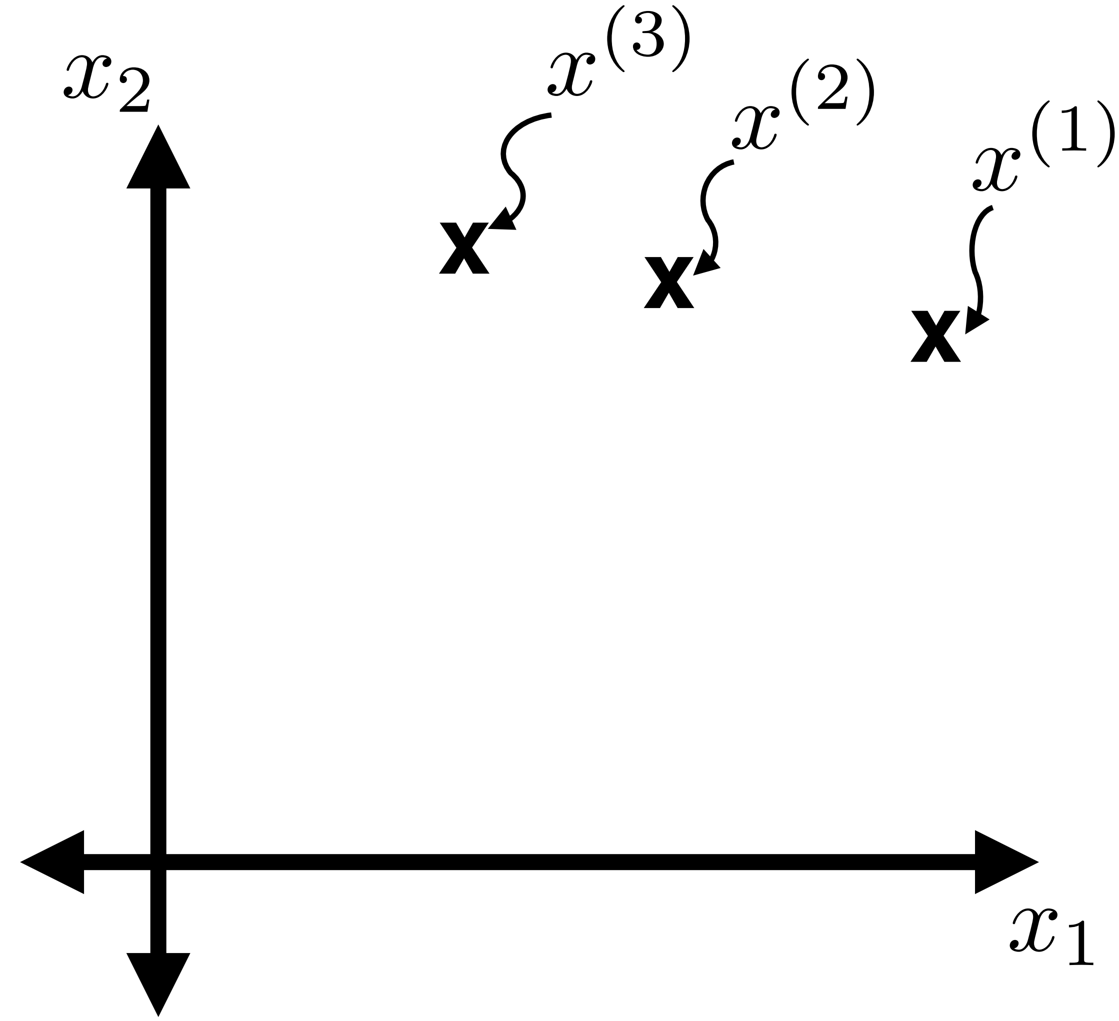
- n training data points
- For data point $i \in \{1, \ldots, n\}$
  - Feature vector
    $x^{(i)} = (x_1^{(i)}, \ldots, x_d^{(i)})^\top \in \mathbb{R}^d$



Getting started

What do we have? (Training) data

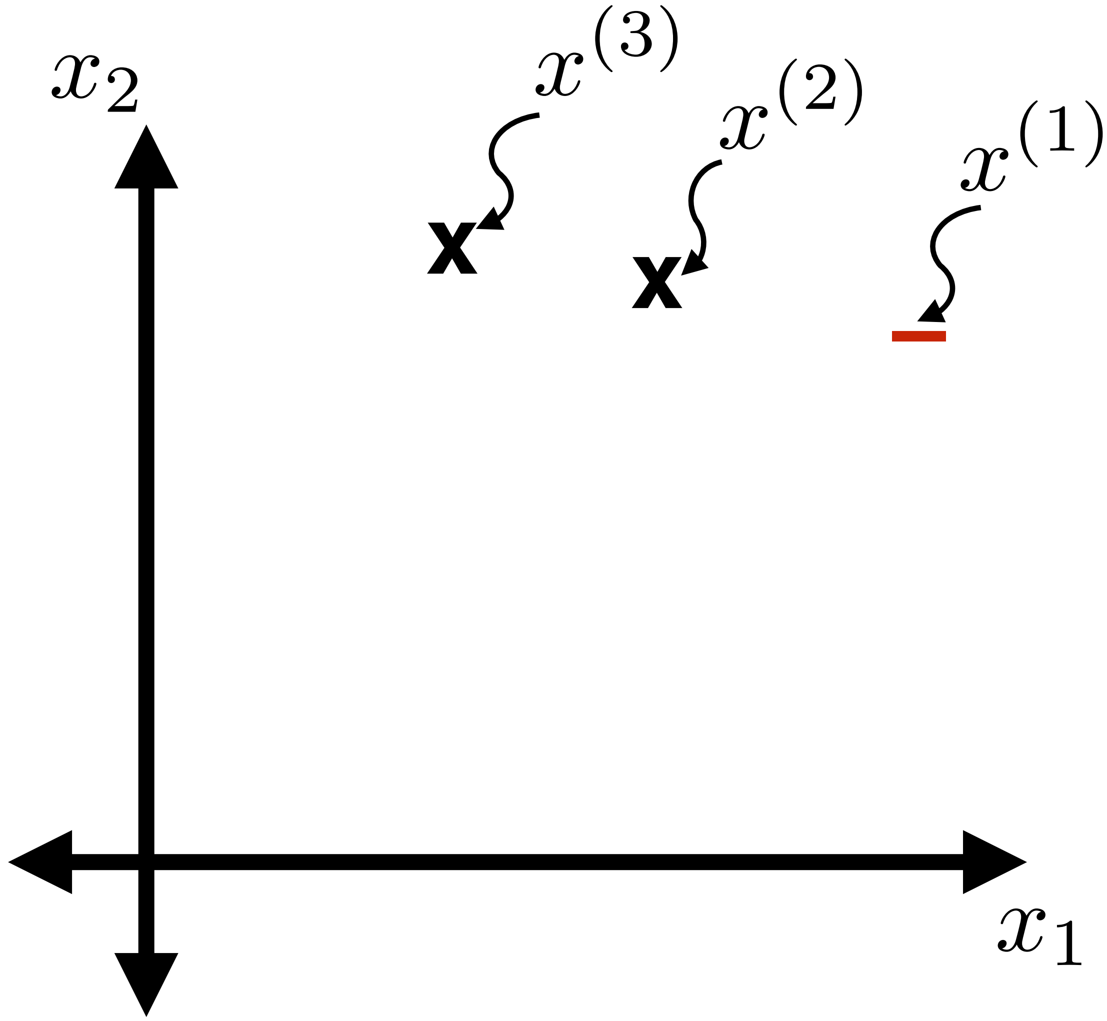
- *n* training data points
- For data point $i \in \{1, \ldots, n\}$
    - Feature vector $x^{(i)} = (x_1^{(i)}, \ldots, x_d^{(i)})^\top \in \mathbb{R}^d$
    - Label $y^{(i)} \in \{-1, +1\}$



Getting started

What do we have? (Training) data

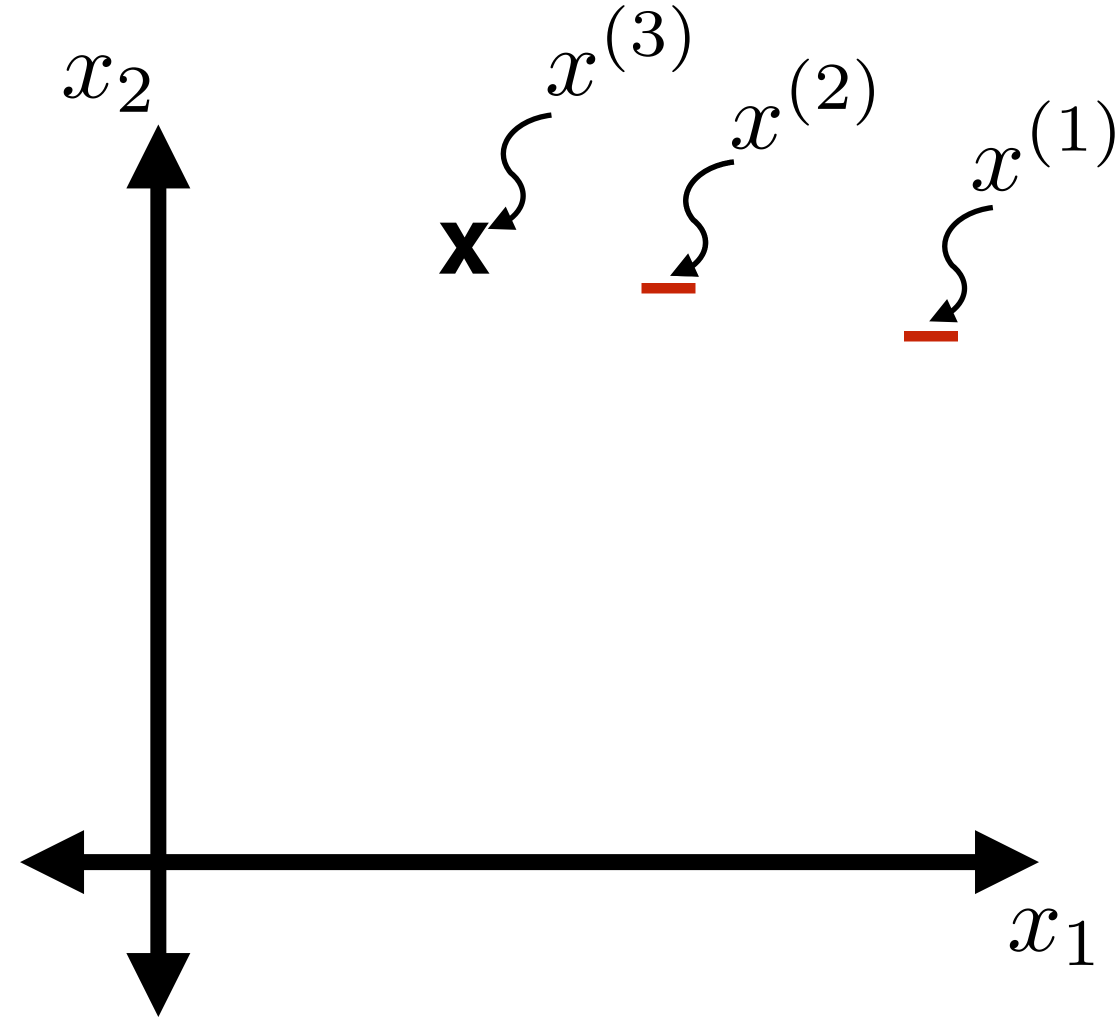
- *n* training data points
- For data point $i \in \{1, \ldots, n\}$
  - Feature vector $x^{(i)} = (x_1^{(i)}, \ldots, x_d^{(i)})^\top \in \mathbb{R}^d$
  - Label $y^{(i)} \in \{-1, +1\}$



Getting started

What do we have? (Training) data

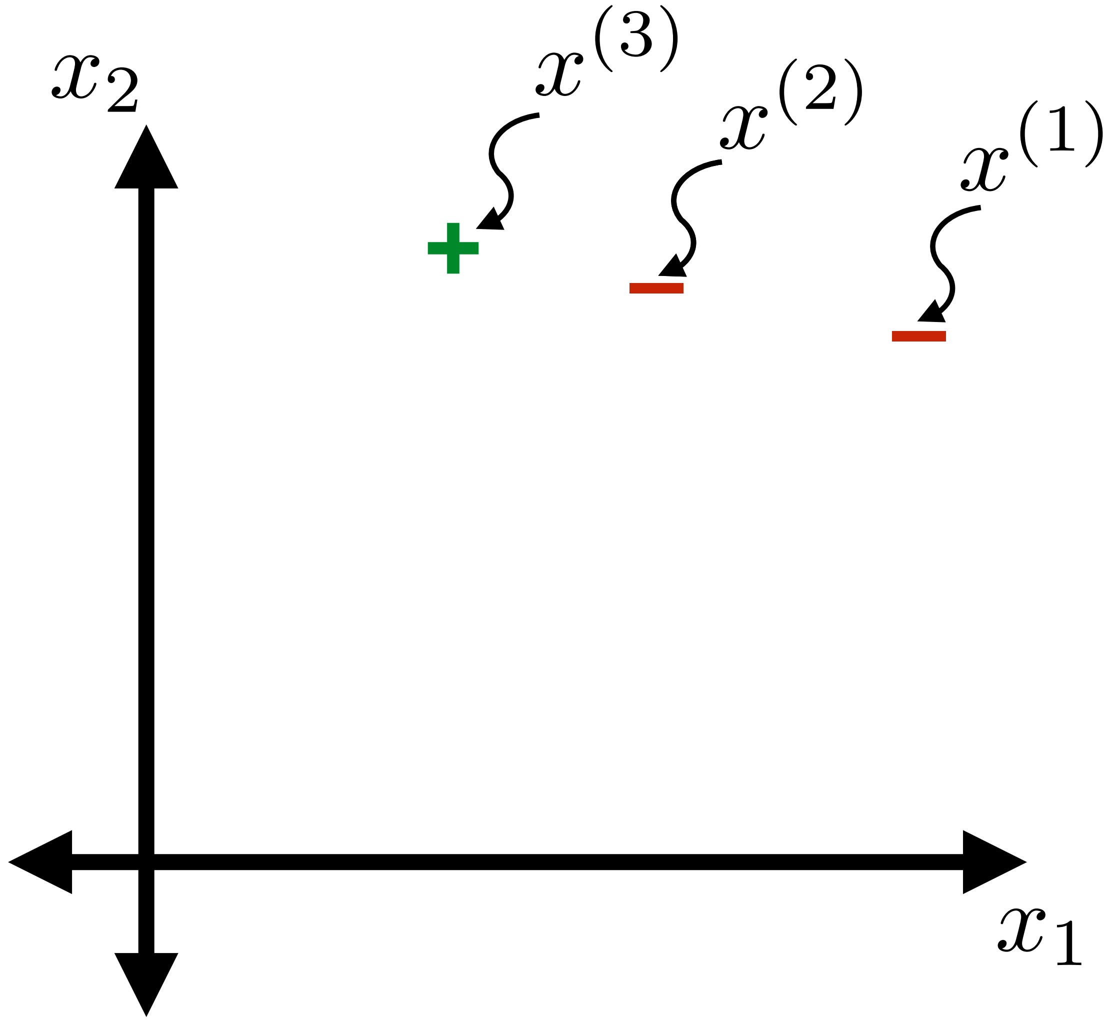
- *n* training data points
- For data point $i \in \{1, \ldots, n\}$
  - Feature vector $x^{(i)} = (x_1^{(i)}, \ldots, x_d^{(i)})^\top \in \mathbb{R}^d$
  - Label $y^{(i)} \in \{-1, +1\}$



Getting started

What do we have? (Training) data

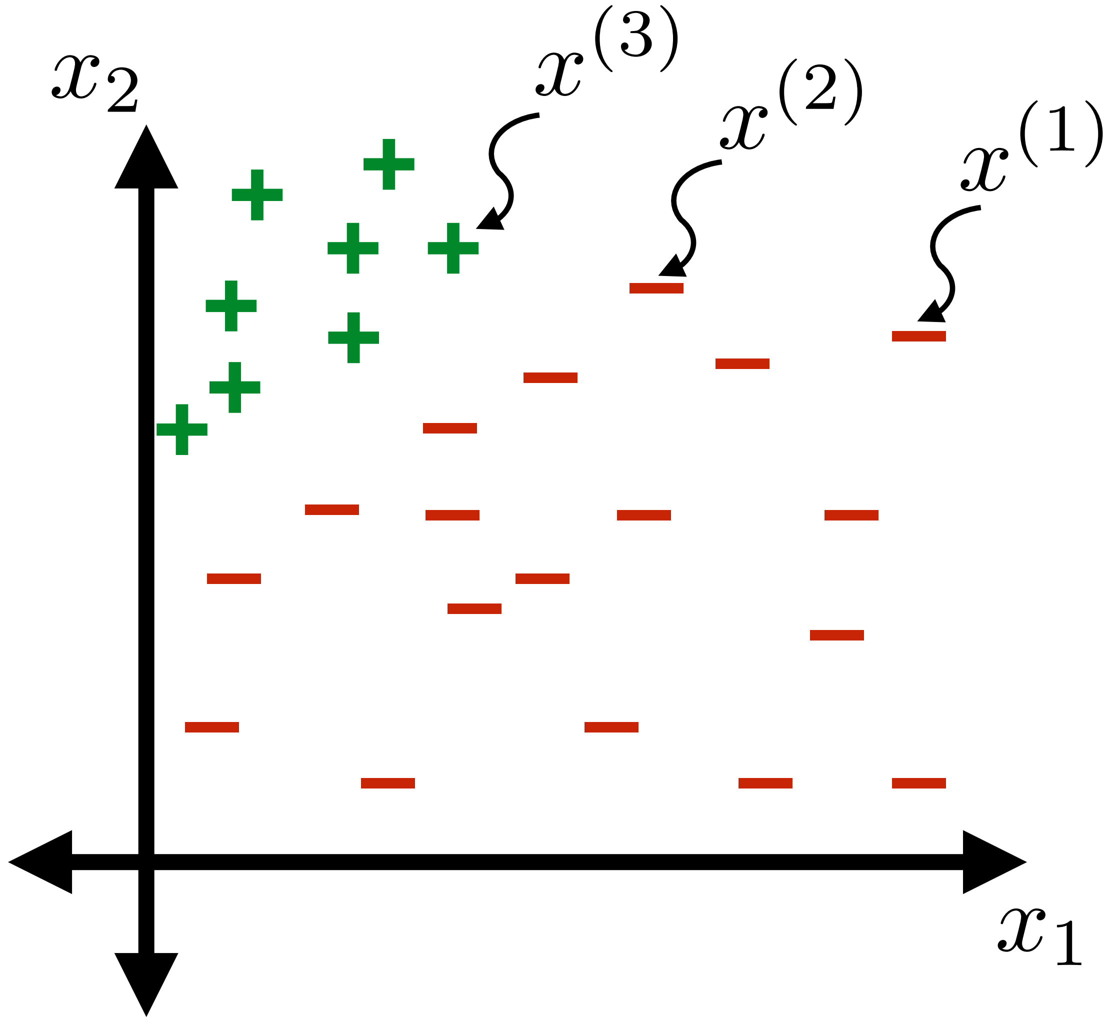
- *n* training data points
- For data point $i \in \{1, \ldots, n\}$
    - Feature vector $x^{(i)} = (x_1^{(i)}, \ldots, x_d^{(i)})^\top \in \mathbb{R}^d$
    - Label $y^{(i)} \in \{-1, +1\}$



Getting started

What do we have? (Training) data

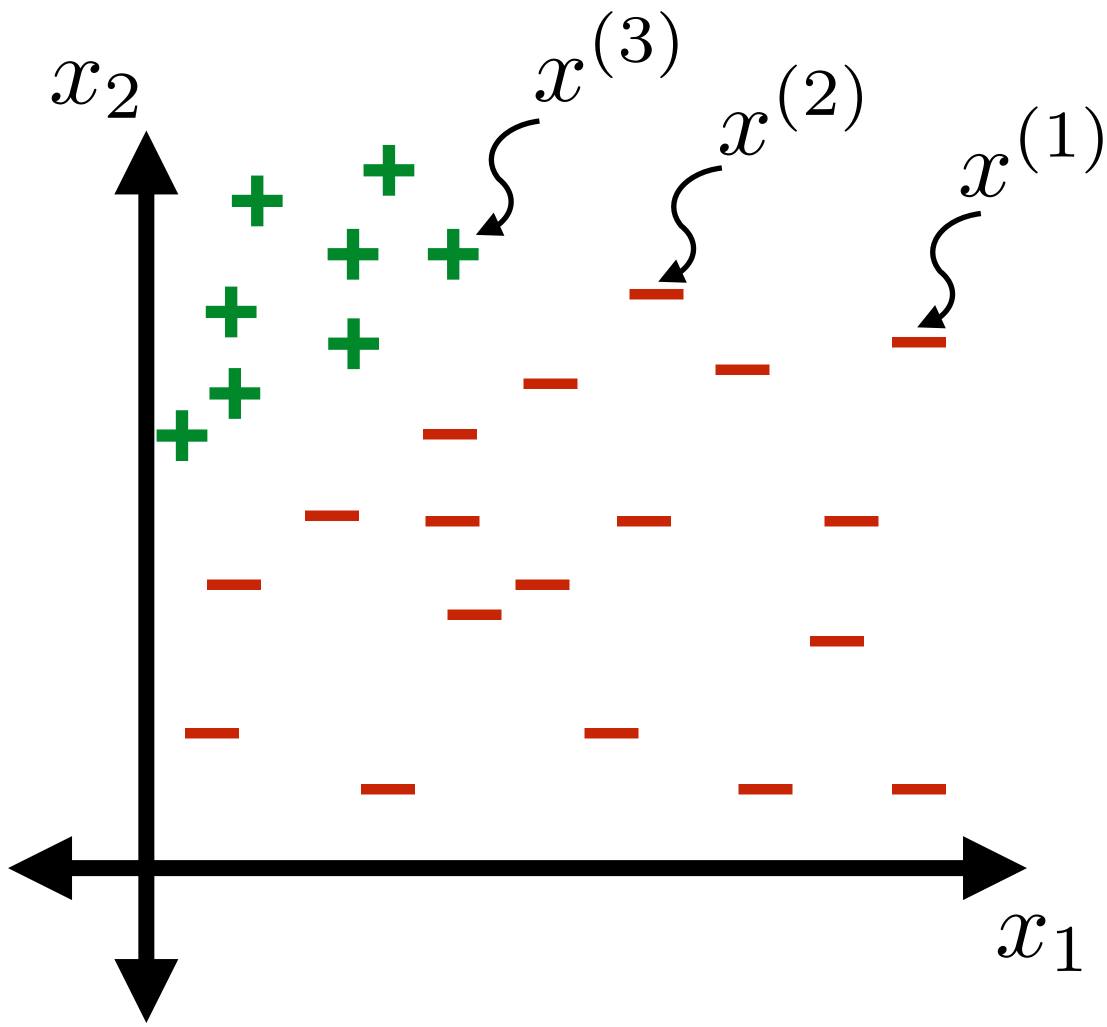
- *n* training data points
- For data point $i \in \{1, \ldots, n\}$
    - Feature vector $x^{(i)} = (x_1^{(i)}, \ldots, x_d^{(i)})^\top \in \mathbb{R}^d$
    - Label $y^{(i)} \in \{-1, +1\}$



Getting started

What do we have? (Training) data

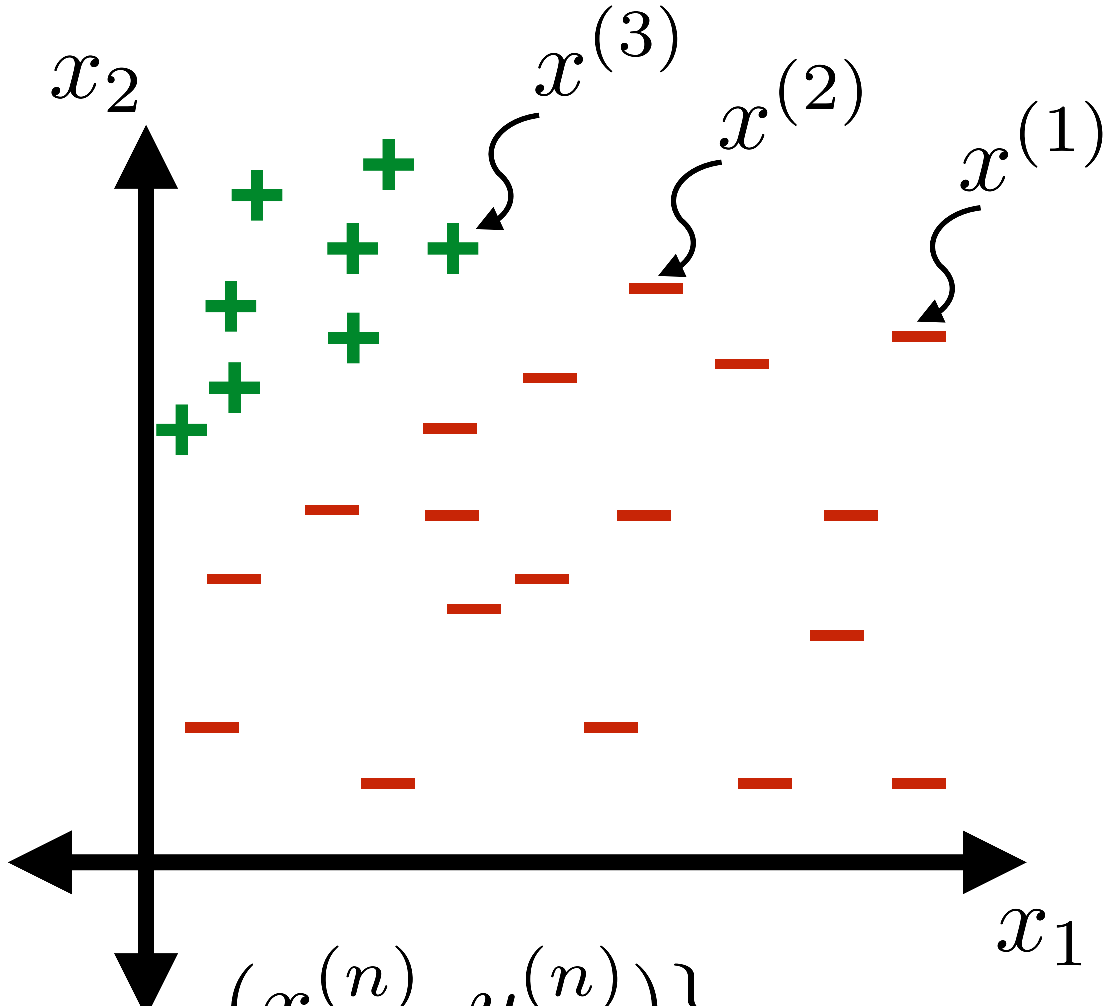
- n training data points
- For data point $i \in \{1, \ldots, n\}$
  - Feature vector
    $x^{(i)} = (x_1^{(i)}, \ldots, x_d^{(i)})^\top \in \mathbb{R}^d$
  - Label $y^{(i)} \in \{-1, +1\}$
- Training data



Getting started

What do we have? (Training) data

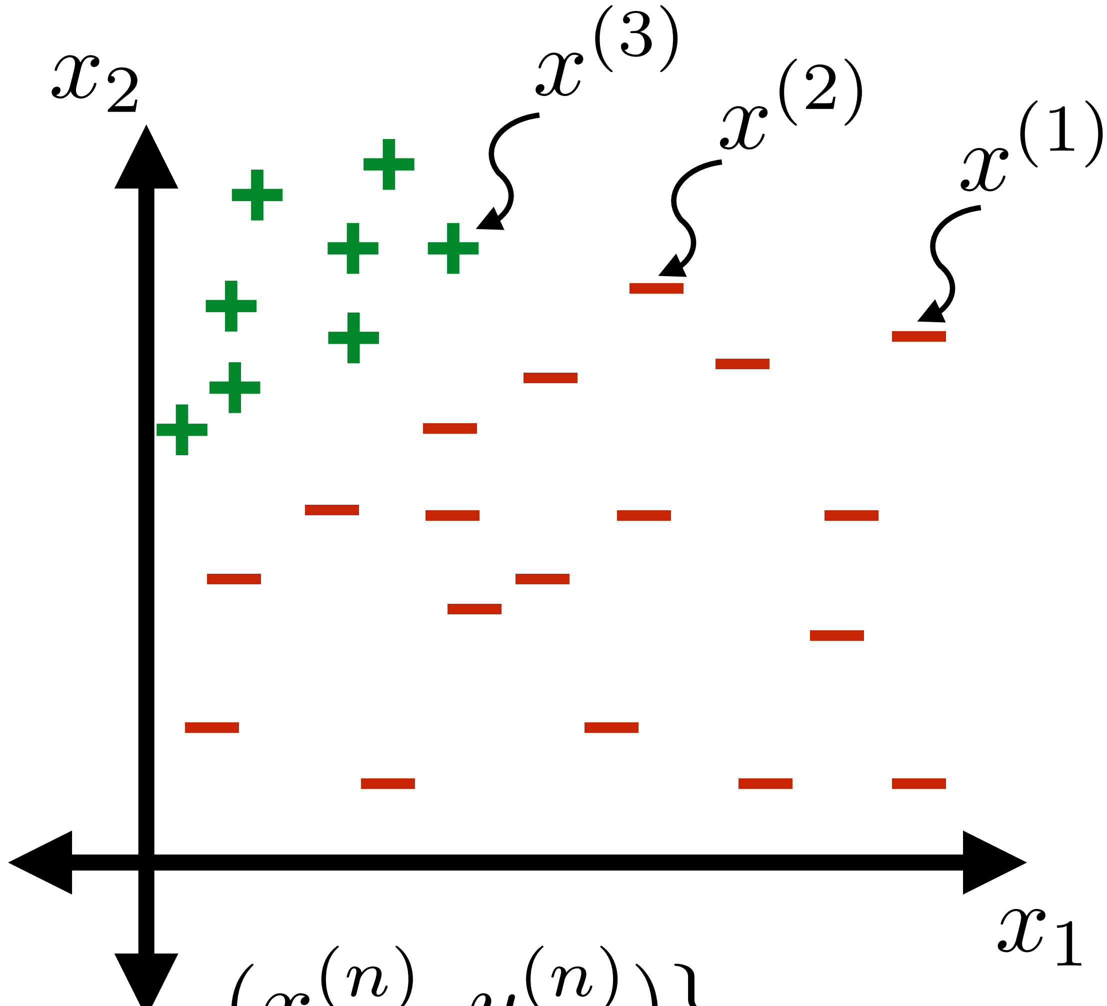
- *n* training data points
- For data point $i \in \{1, \ldots, n\}$
    - Feature vector $x^{(i)} = (x_1^{(i)}, \ldots, x_d^{(i)})^\top \in \mathbb{R}^d$
    - Label $y^{(i)} \in \{-1, +1\}$
- Training data $\mathcal{D}_n = \{(x^{(1)}, y^{(1)}), \ldots, (x^{(n)}, y^{(n)})\}$



Getting started

What do we have? (Training) data

- n training data points
- For data point $i \in \{1, \ldots, n\}$
  - Feature vector $x^{(i)} = (x_1^{(i)}, \ldots, x_d^{(i)})^\top \in \mathbb{R}^d$
  - Label $y^{(i)} \in \{-1, +1\}$
- Training data $\mathcal{D}_n = \{(x^{(1)}, y^{(1)}), \ldots, (x^{(n)}, y^{(n)})\}$

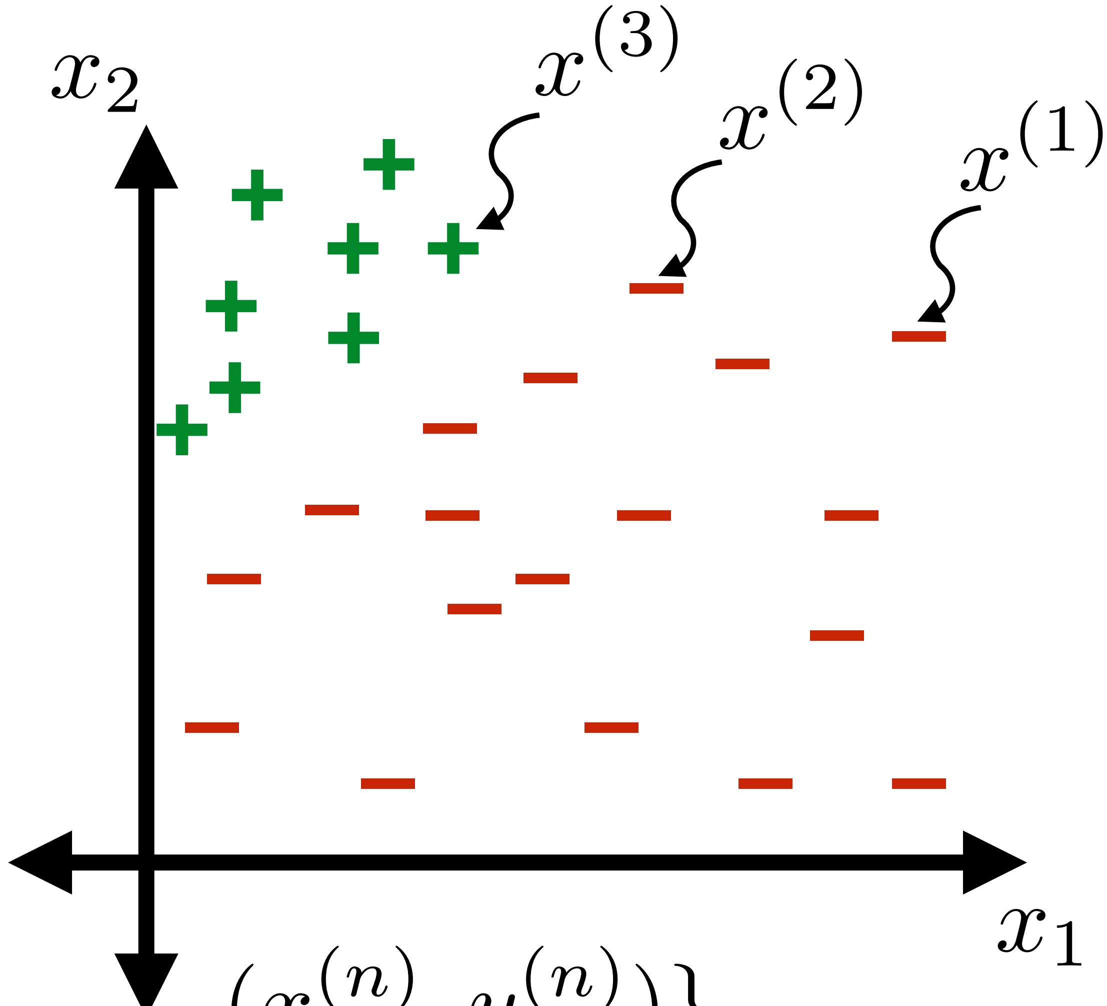


What do we want?

Getting started

What do we have? (Training) data

- n training data points
- For data point $i \in \{1, \ldots, n\}$
  - Feature vector $x^{(i)} = (x_1^{(i)}, \ldots, x_d^{(i)})^\top \in \mathbb{R}^d$
  - Label $y^{(i)} \in \{-1, +1\}$
- Training data $\mathcal{D}_n = \{(x^{(1)}, y^{(1)}), \ldots, (x^{(n)}, y^{(n)})\}$

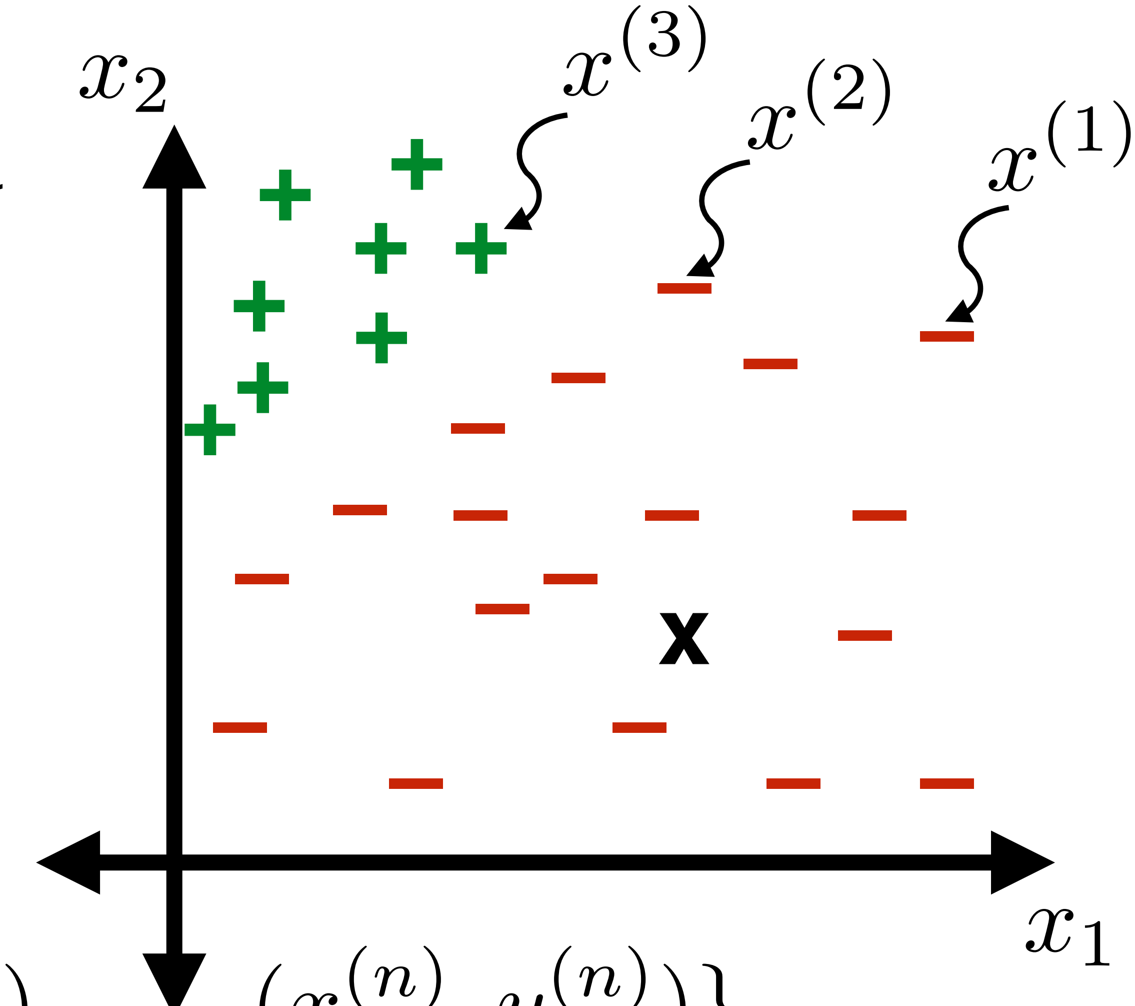


What do we want? A good way to label new points

Getting started

What do we have? (Training) data

- n training data points
- For data point $i \in \{1, \dots, n\}$
    - Feature vector $x^{(i)} = (x_1^{(i)}, \dots, x_d^{(i)})^\top \in \mathbb{R}^d$
    - Label $y^{(i)} \in \{-1, +1\}$
- Training data $\mathcal{D}_n = \{(x^{(1)}, y^{(1)}), \dots, (x^{(n)}, y^{(n)})\}$

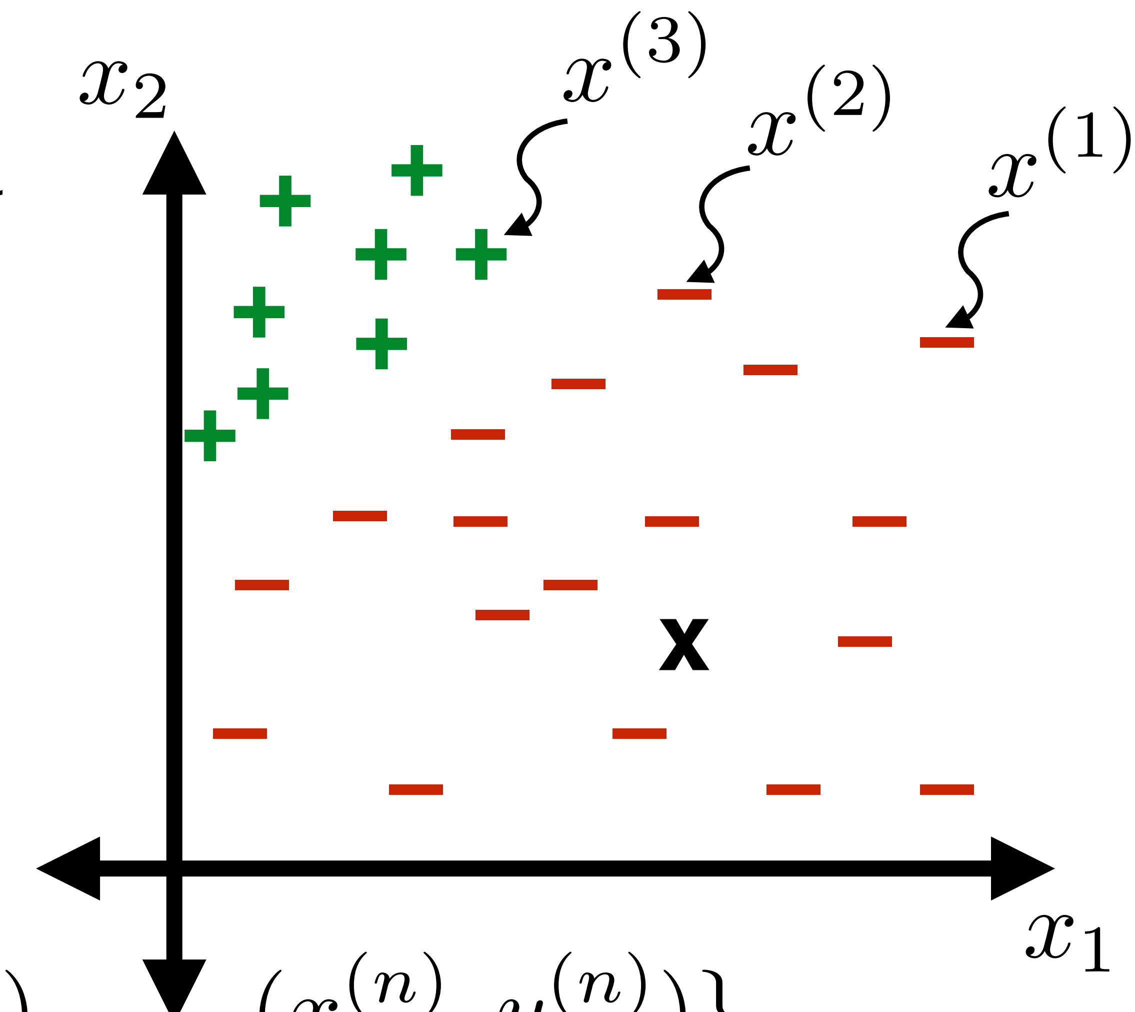


What do we want? A good way to label new points

Getting started

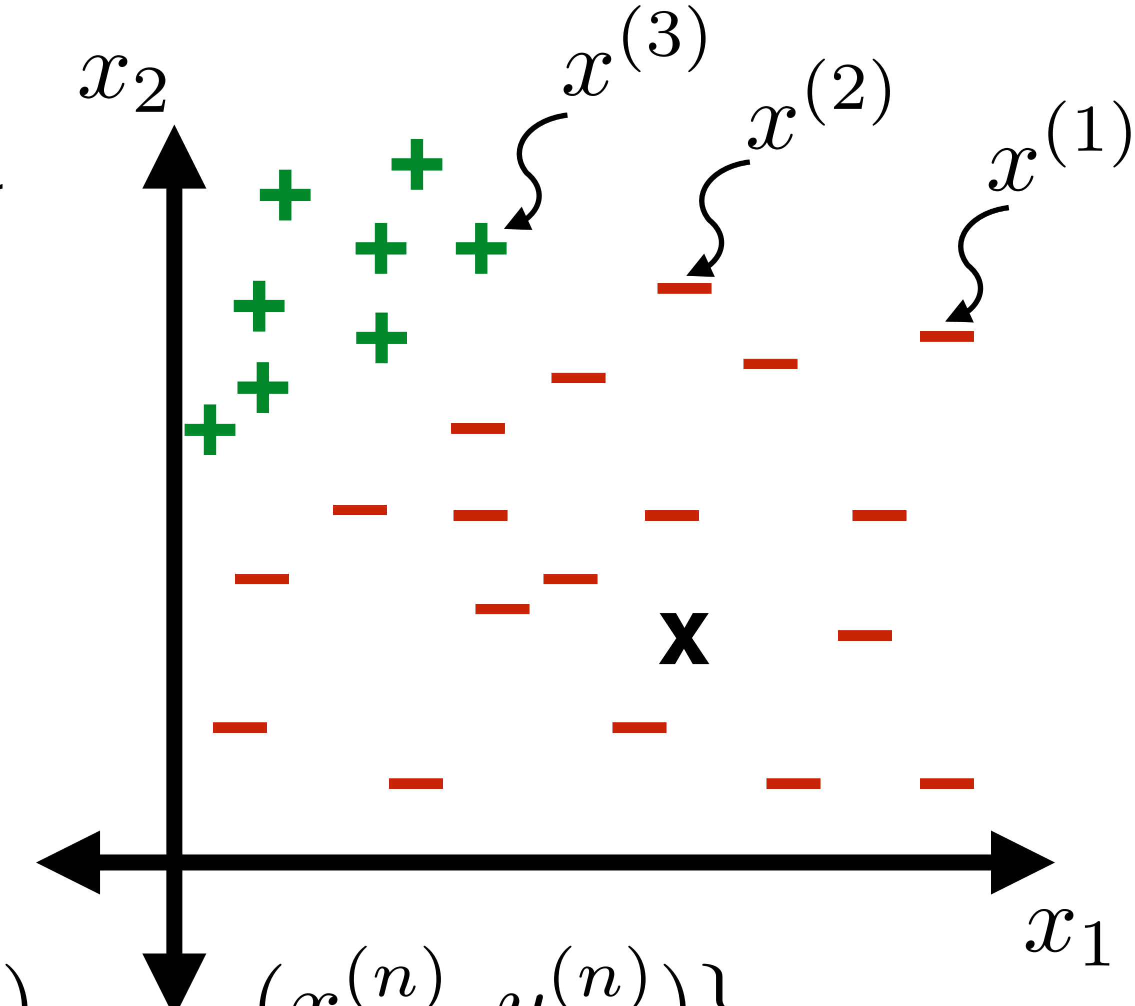
What do we have? (Training) data

- n training data points
- For data point $i \in \{1, \ldots, n\}$
  - Feature vector $x^{(i)} = (x_1^{(i)}, \ldots, x_d^{(i)})^\top \in \mathbb{R}^d$
  - Label $y^{(i)} \in \{-1, +1\}$
- Training data $\mathcal{D}_n = \{(x^{(1)}, y^{(1)}), \ldots, (x^{(n)}, y^{(n)})\}$



What do we want? A good way to label new points

Getting started



What do we have? (Training) data

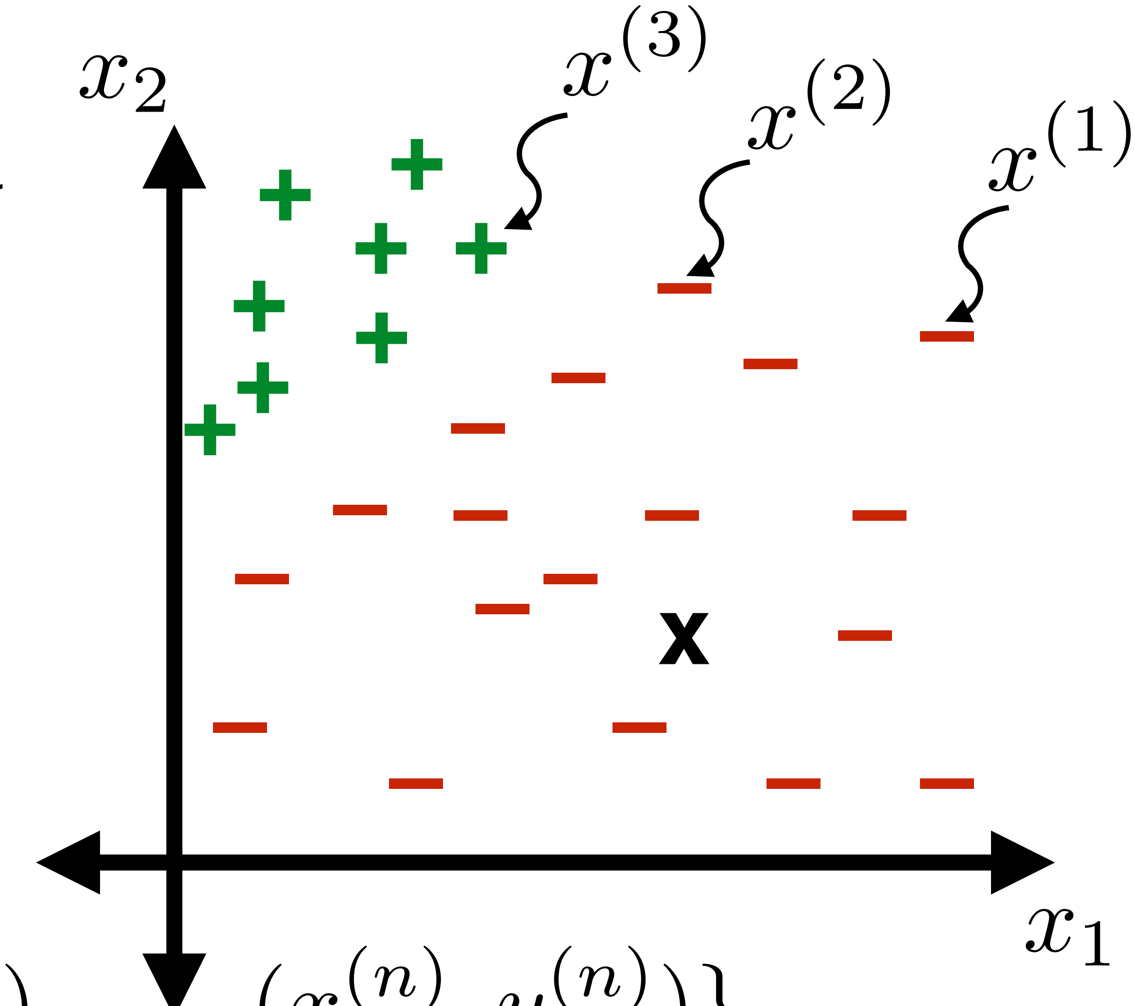
- *n* training data points
- For data point $i \in \{1, \ldots, n\}$
  - Feature vector
    $x^{(i)} = (x_1^{(i)}, \ldots, x_d^{(i)})^\top \in \mathbb{R}^d$
  - Label $y^{(i)} \in \{-1, +1\}$
- Training data $\mathcal{D}_n = \{(x^{(1)}, y^{(1)}), \ldots, (x^{(n)}, y^{(n)})\}$

What do we want? A good way to label new points

Getting started

What do we have? (Training) data

- n training data points
- For data point $i \in \{1, \ldots, n\}$
  - Feature vector $x^{(i)} = (x_1^{(i)}, \ldots, x_d^{(i)})^\top \in \mathbb{R}^d$
  - Label $y^{(i)} \in \{-1, +1\}$
- Training data $\mathcal{D}_n = \{(x^{(1)}, y^{(1)}), \ldots, (x^{(n)}, y^{(n)})\}$

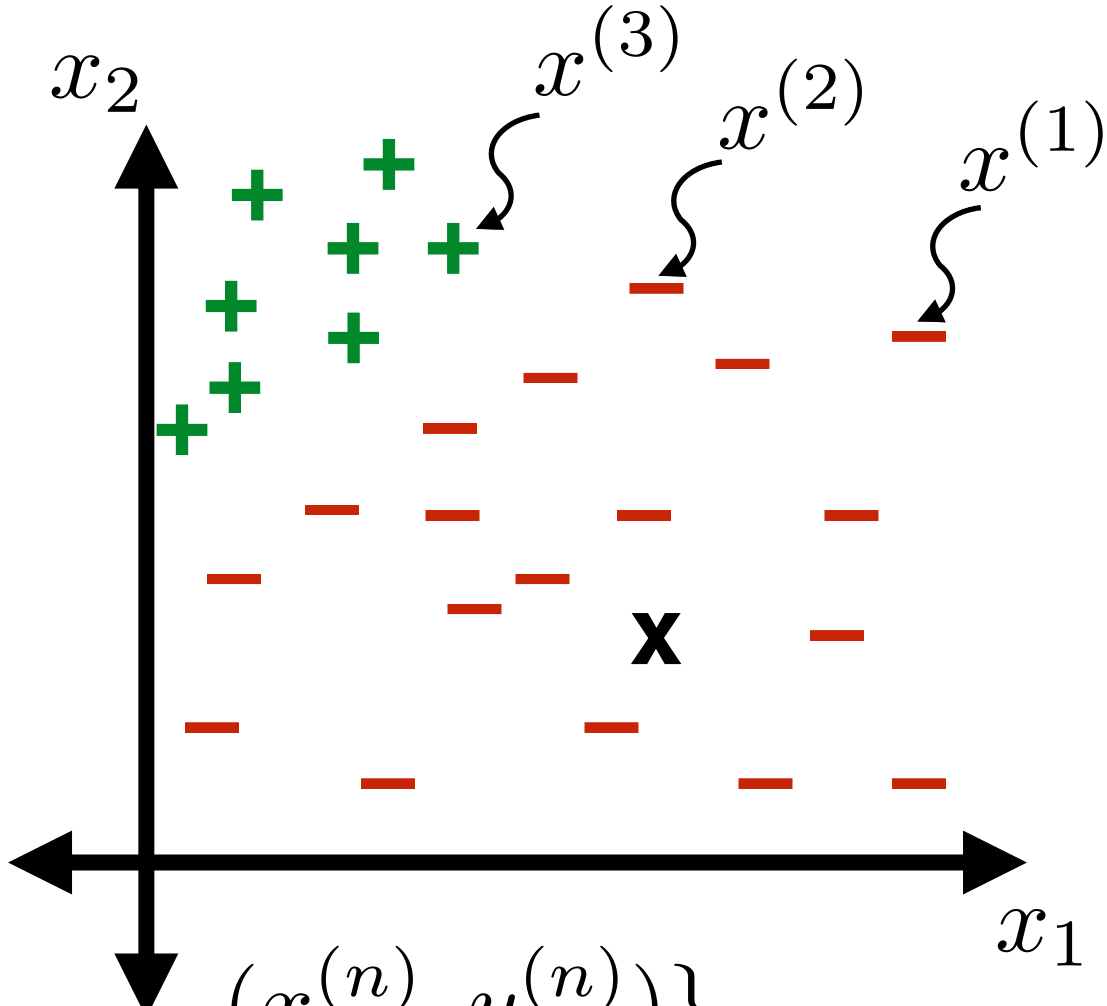


What do we want? A good way to label new points

Getting started

What do we have? (Training) data

- n training data points
- For data point $i \in \{1, \ldots, n\}$
  - Feature vector $x^{(i)} = (x_1^{(i)}, \ldots, x_d^{(i)})^\top \in \mathbb{R}^d$
  - Label $y^{(i)} \in \{-1, +1\}$
- Training data $\mathcal{D}_n = \{(x^{(1)}, y^{(1)}), \ldots, (x^{(n)}, y^{(n)})\}$



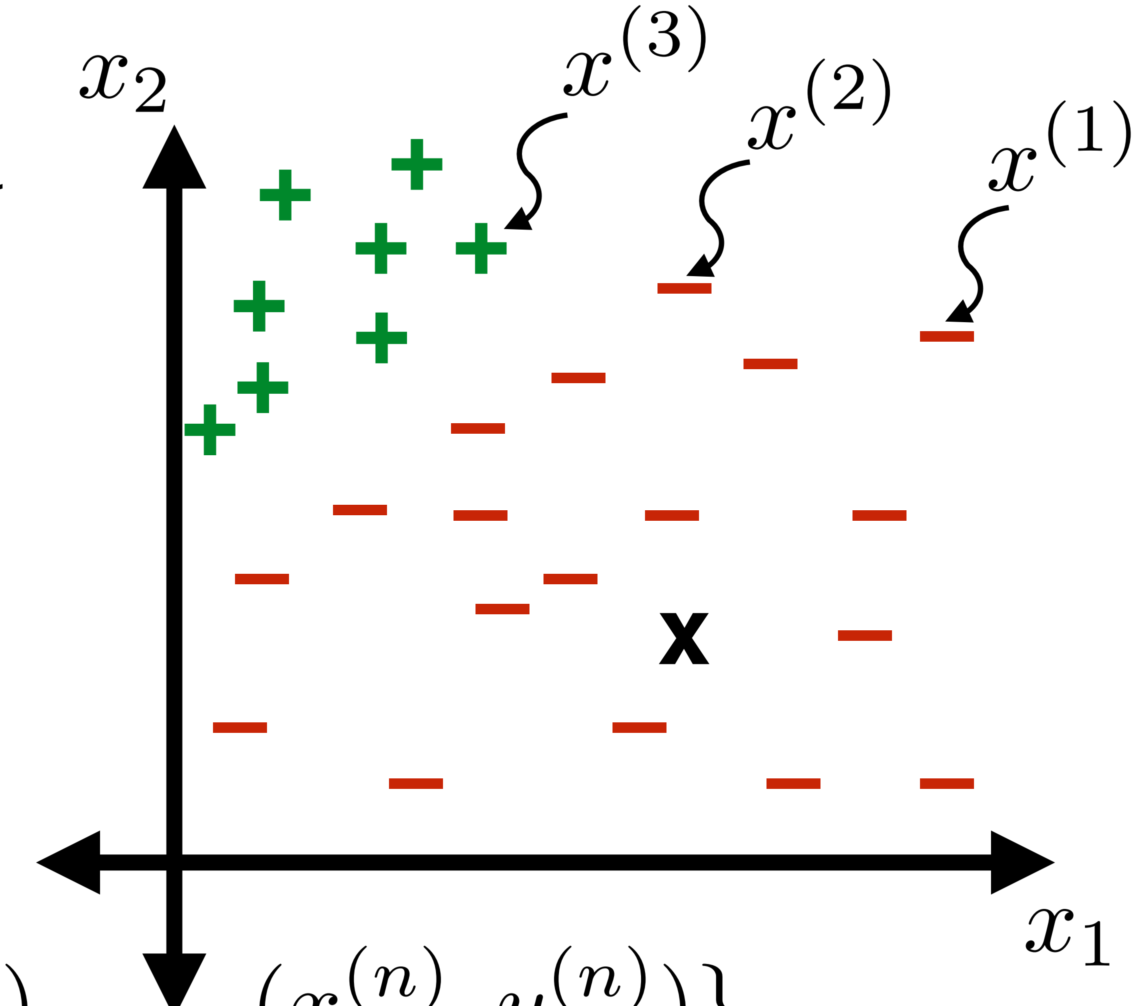
What do we want? A good way to label new points

- How to label?

Getting started

What do we have? (Training) data

- n training data points
- For data point $i \in \{1, \ldots, n\}$
  - Feature vector $x^{(i)} = (x_1^{(i)}, \ldots, x_d^{(i)})^\top \in \mathbb{R}^d$
  - Label $y^{(i)} \in \{-1, +1\}$
- Training data $\mathcal{D}_n = \{(x^{(1)}, y^{(1)}), \ldots, (x^{(n)}, y^{(n)})\}$



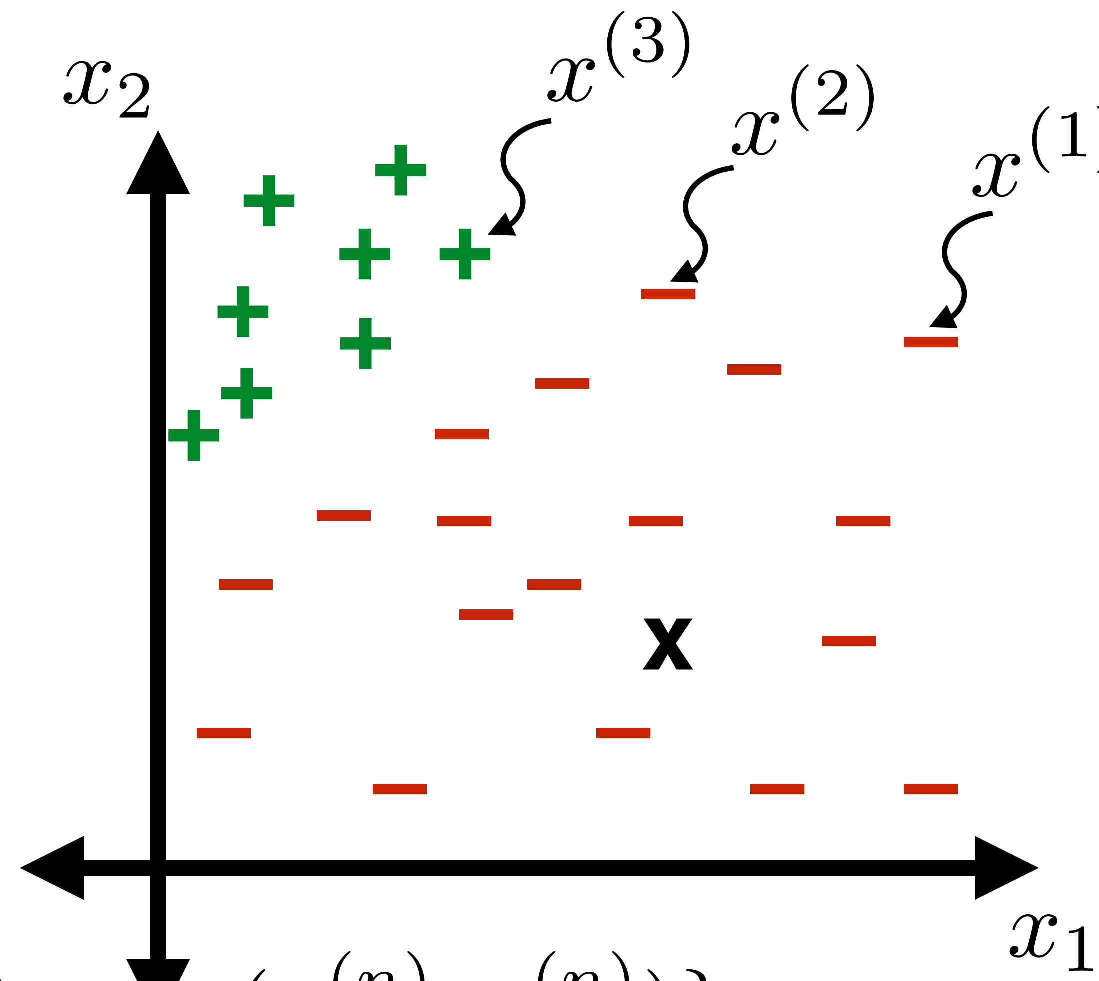
What do we want? A good way to label new points

- How to label? Hypothesis $h : \mathbb{R}^d \rightarrow \{-1, +1\}$

Getting started

What do we have? (Training) data

- n training data points
- For data point $i \in \{1, \ldots, n\}$
  - Feature vector $x^{(i)} = (x_1^{(i)}, \ldots, x_d^{(i)})^\top \in \mathbb{R}^d$
  - Label $y^{(i)} \in \{-1, +1\}$
- Training data $\mathcal{D}_n = \{(x^{(1)}, y^{(1)}), \ldots, (x^{(n)}, y^{(n)})\}$



What do we want? A good way to label new points

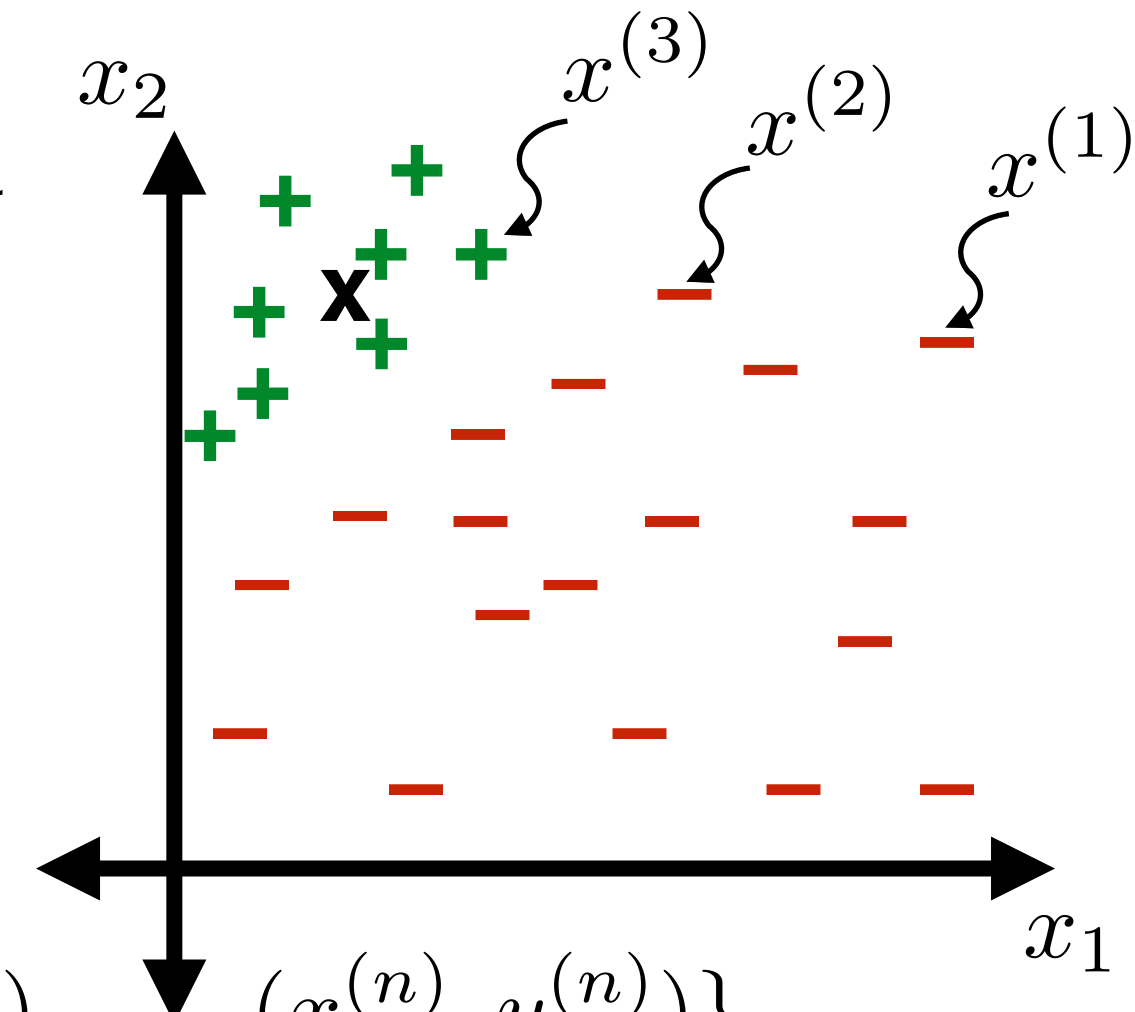
- How to label? Hypothesis $h : \mathbb{R}^d \to \{-1, +1\}$

$$x \rightarrow \boxed{h} \rightarrow y$$

Getting started

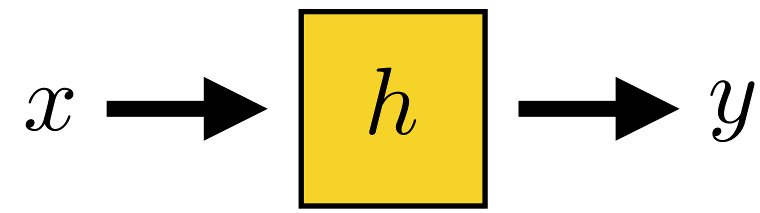
What do we have? (Training) data

- *n* training data points
- For data point $i \in \{1, \ldots, n\}$
  - Feature vector $x^{(i)} = (x_1^{(i)}, \ldots, x_d^{(i)})^\top \in \mathbb{R}^d$
  - Label $y^{(i)} \in \{-1, +1\}$
- Training data $\mathcal{D}_n = \{(x^{(1)}, y^{(1)}), \ldots, (x^{(n)}, y^{(n)})\}$

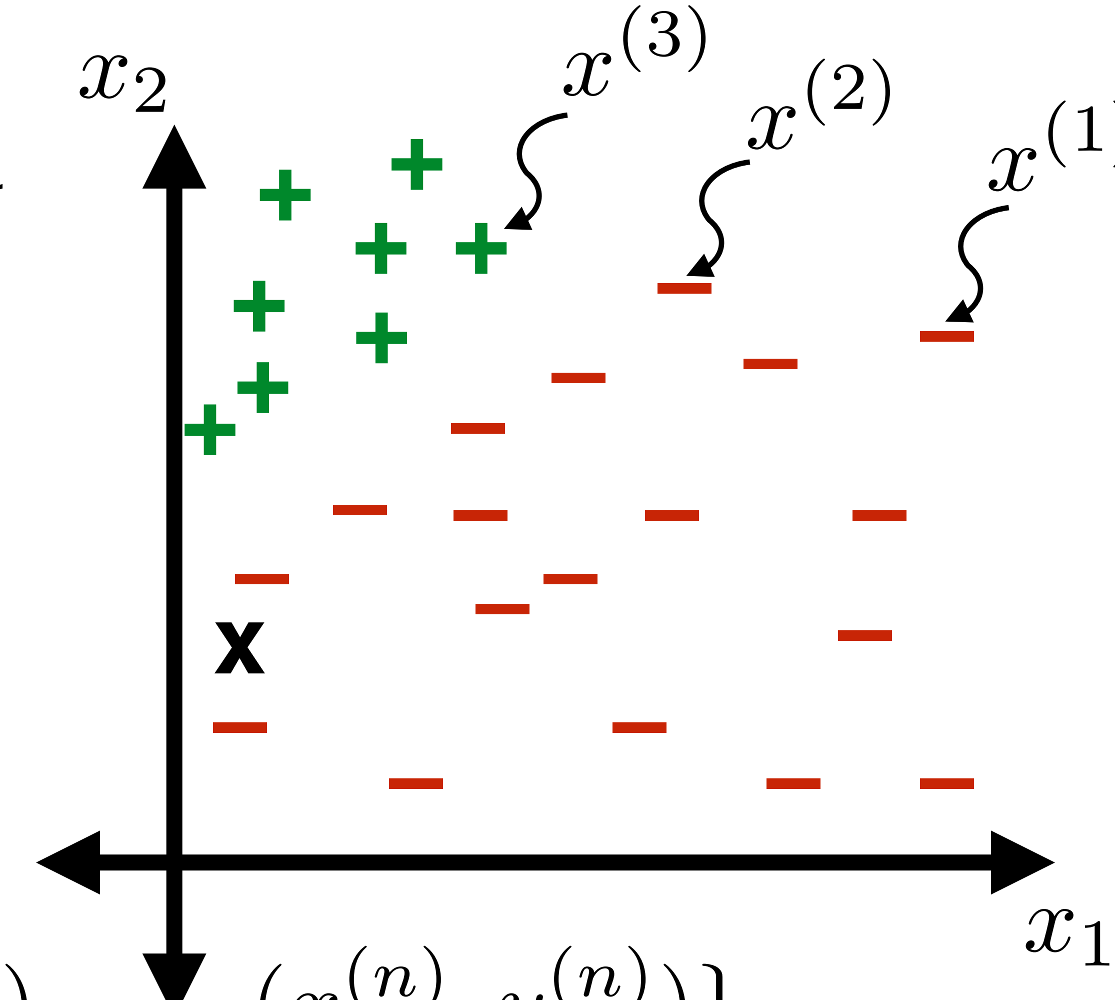


What do we want? A good way to label new points

- How to label? Hypothesis $h : \mathbb{R}^d \to \{-1, +1\}$



Getting started



What do we have? (Training) data

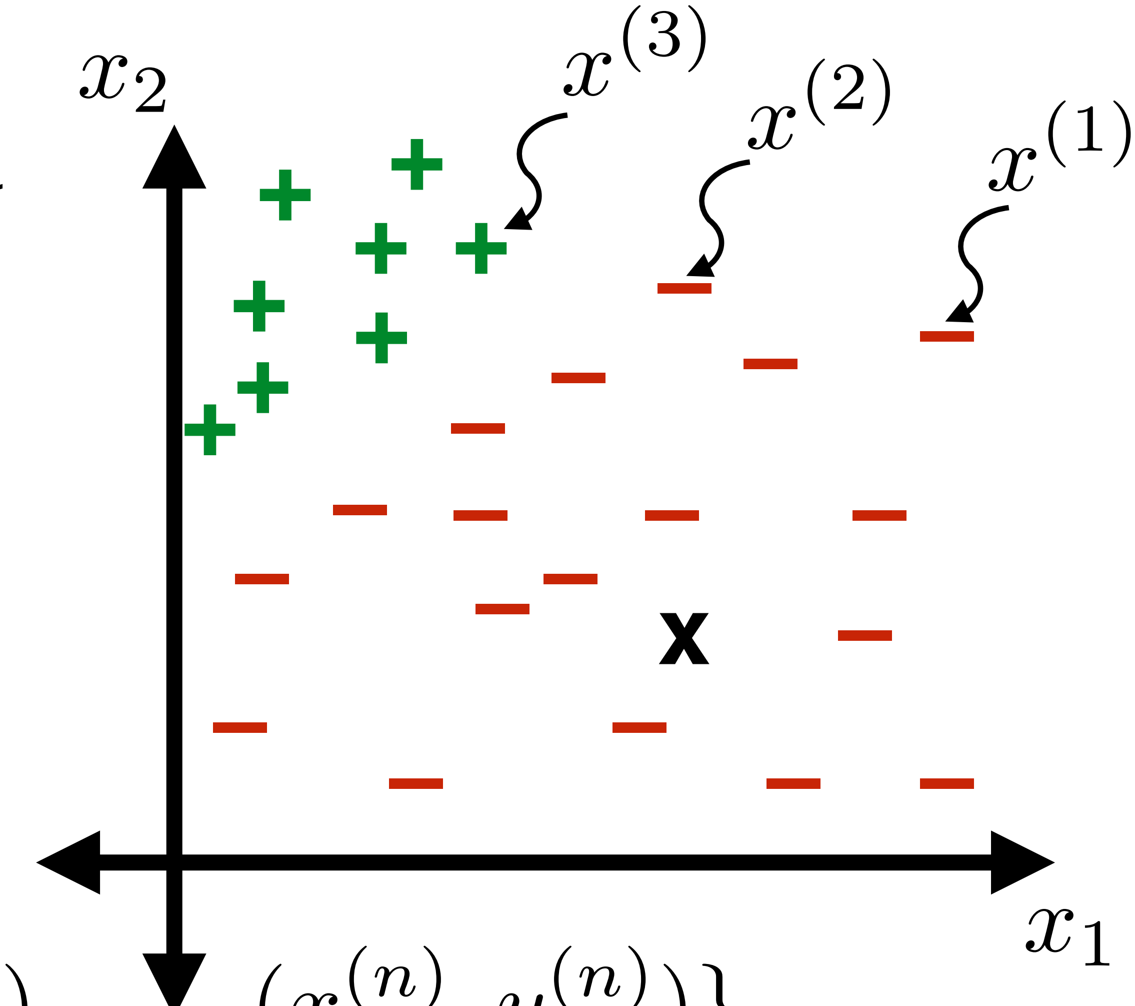
- *n* training data points
- For data point $i \in \{1, \ldots, n\}$
  - Feature vector $x^{(i)} = (x_1^{(i)}, \ldots, x_d^{(i)})^\top \in \mathbb{R}^d$
  - Label $y^{(i)} \in \{-1, +1\}$
- Training data $\mathcal{D}_n = \{(x^{(1)}, y^{(1)}), \ldots, (x^{(n)}, y^{(n)})\}$

What do we want? A good way to label new points

- How to label? Hypothesis $h : \mathbb{R}^d \to \{-1, +1\}$

$$x \rightarrow \boxed{h} \rightarrow y$$

Getting started

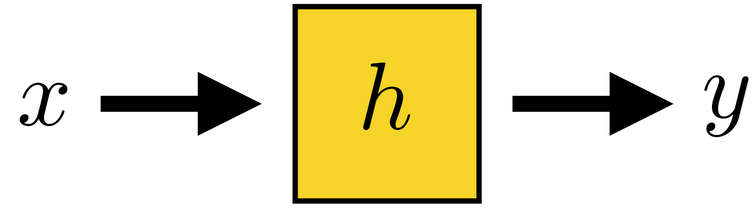


What do we have? (Training) data

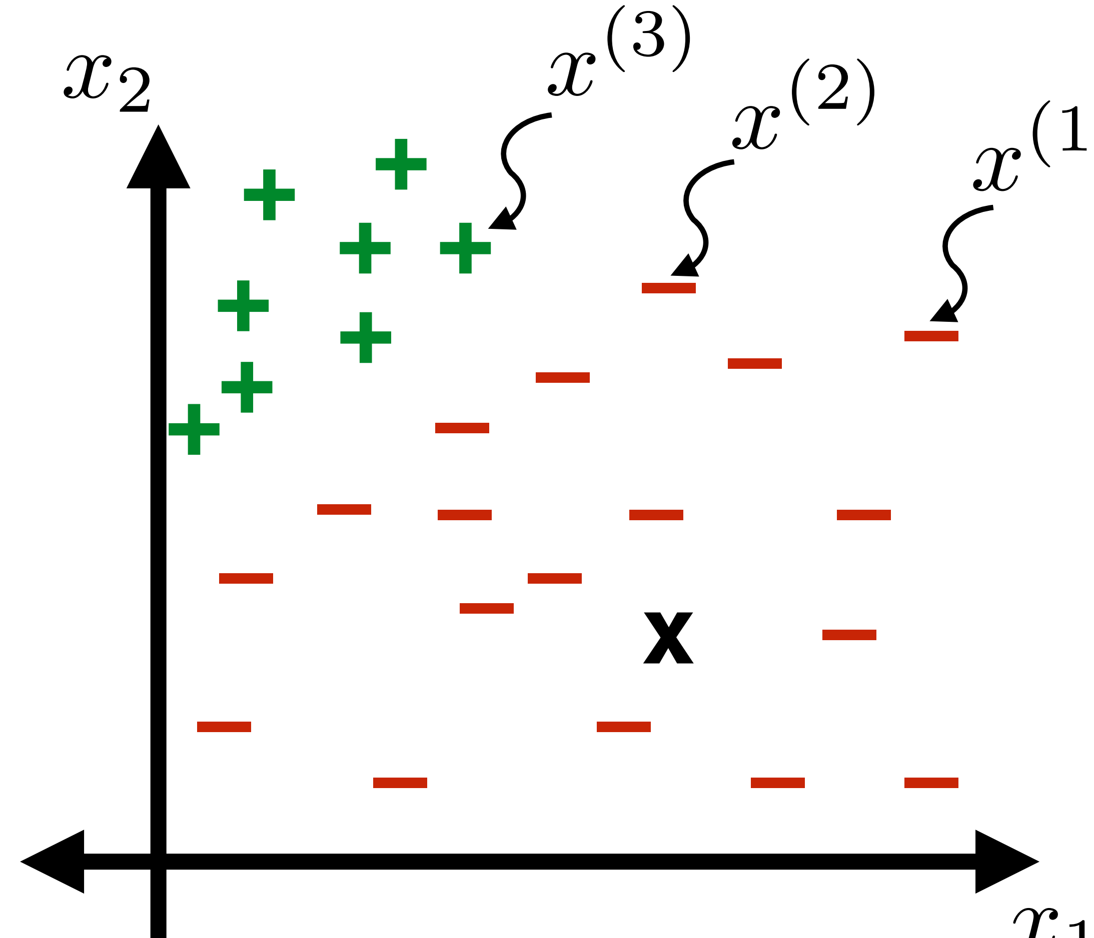
- *n* training data points
- For data point $i \in \{1, \ldots, n\}$
  - Feature vector $x^{(i)} = (x_1^{(i)}, \ldots, x_d^{(i)})^\top \in \mathbb{R}^d$
  - Label $y^{(i)} \in \{-1, +1\}$
- Training data $\mathcal{D}_n = \{(x^{(1)}, y^{(1)}), \ldots, (x^{(n)}, y^{(n)})\}$

What do we want? A good way to label new points

- How to label? Hypothesis $h : \mathbb{R}^d \to \{-1, +1\}$



Getting started



What do we have? (Training) data

- *n* training data points
- For data point $i \in \{1, \ldots, n\}$
  - Feature vector $x^{(i)} = (x_1^{(i)}, \ldots, x_d^{(i)})^\top \in \mathbb{R}^d$
  - Label $y^{(i)} \in \{-1, +1\}$
- Training data $\mathcal{D}_n = \{(x^{(1)}, y^{(1)}), \ldots, (x^{(n)}, y^{(n)})\}$

What do we want? A good way to label new points

- How to label? Hypothesis $h : \mathbb{R}^d \to \{-1, +1\}$

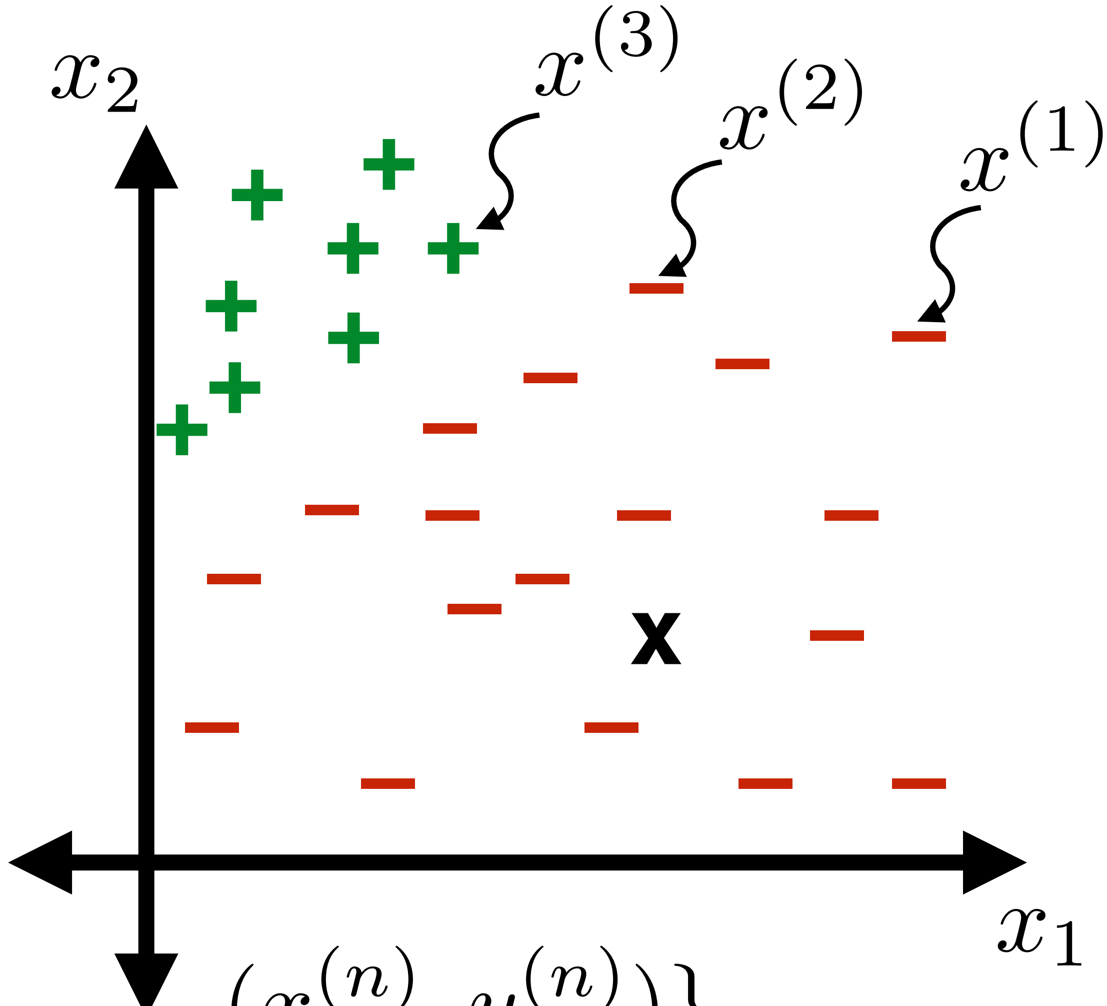
$$x \longrightarrow \boxed{h} \longrightarrow y$$

- Example *h*: For any *x*, *h*(*x*) = +1

Getting started

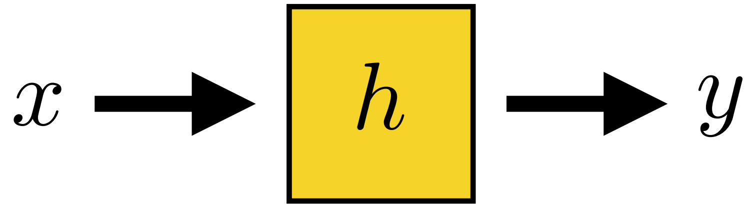
What do we have? (Training) data

- n training data points
- For data point $i \in \{1, \ldots, n\}$
  - Feature vector
    $x^{(i)} = (x_1^{(i)}, \ldots, x_d^{(i)})^\top \in \mathbb{R}^d$
  - Label $y^{(i)} \in \{-1, +1\}$
- Training data $\mathcal{D}_n = \{(x^{(1)}, y^{(1)}), \ldots, (x^{(n)}, y^{(n)})\}$



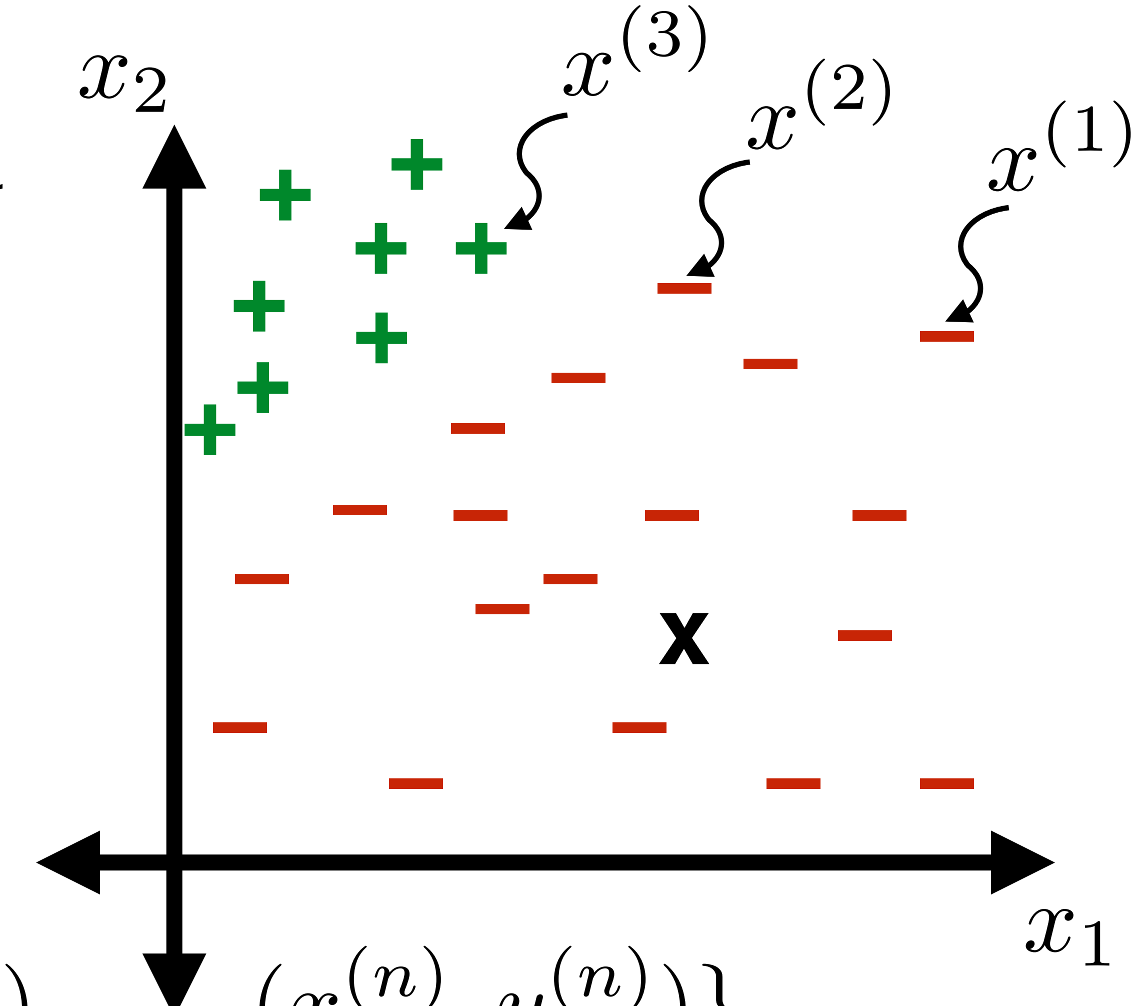
What do we want? A good way to label new points

- How to label? Hypothesis $h : \mathbb{R}^d \to \{-1, +1\}$



  - Example h: For any x, $h(x) = +1$
    - Is this a hypothesis?

Getting started



What do we have? (Training) data

- n training data points
- For data point $i \in \{1, \ldots, n\}$
    - Feature vector
      $x^{(i)} = (x_1^{(i)}, \ldots, x_d^{(i)})^\top \in \mathbb{R}^d$
    - Label $y^{(i)} \in \{-1, +1\}$
- Training data $\mathcal{D}_n = \{(x^{(1)}, y^{(1)}), \ldots, (x^{(n)}, y^{(n)})\}$

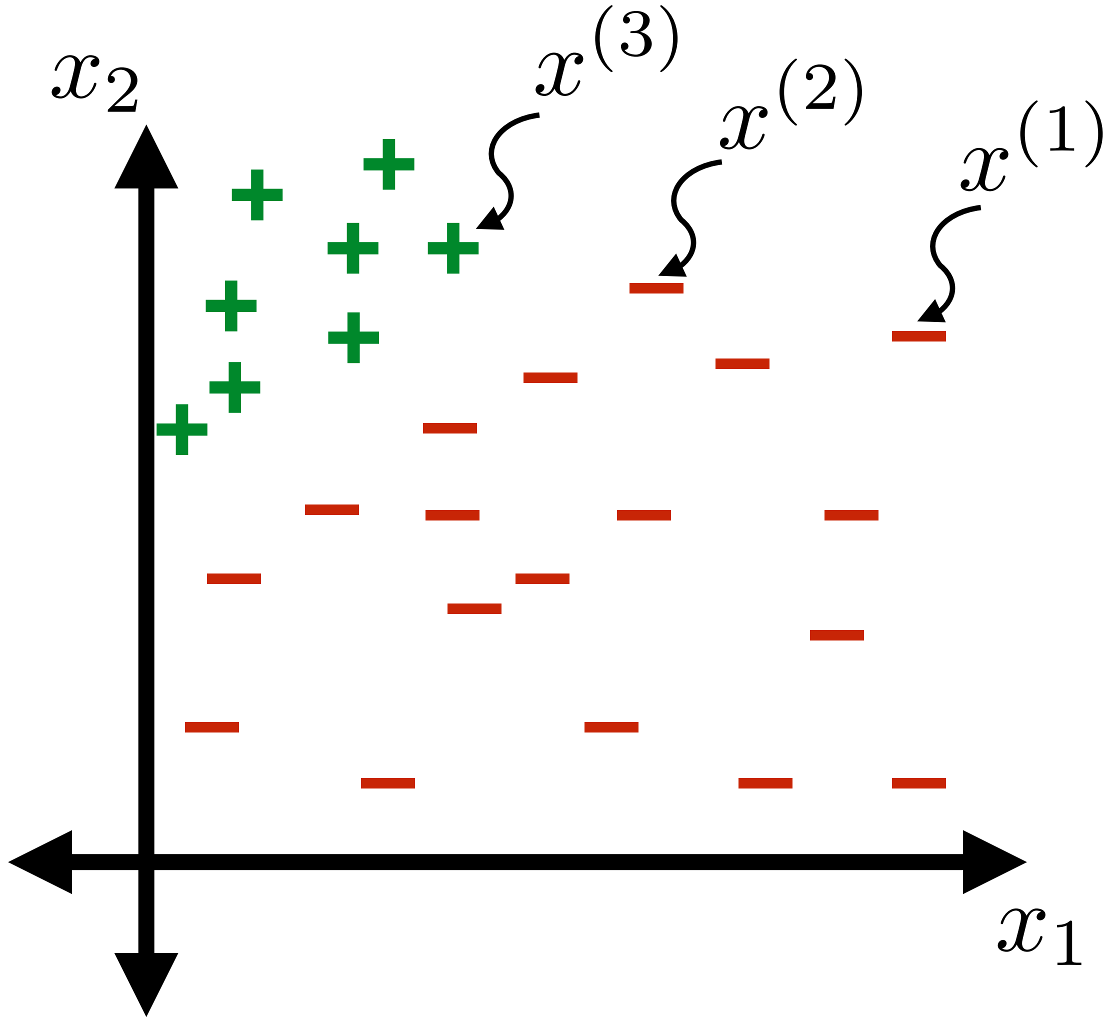
What do we want? A good way to label new points

- How to label? Hypothesis $h : \mathbb{R}^d \rightarrow \{-1, +1\}$

$$x \rightarrow \boxed{h} \rightarrow y$$

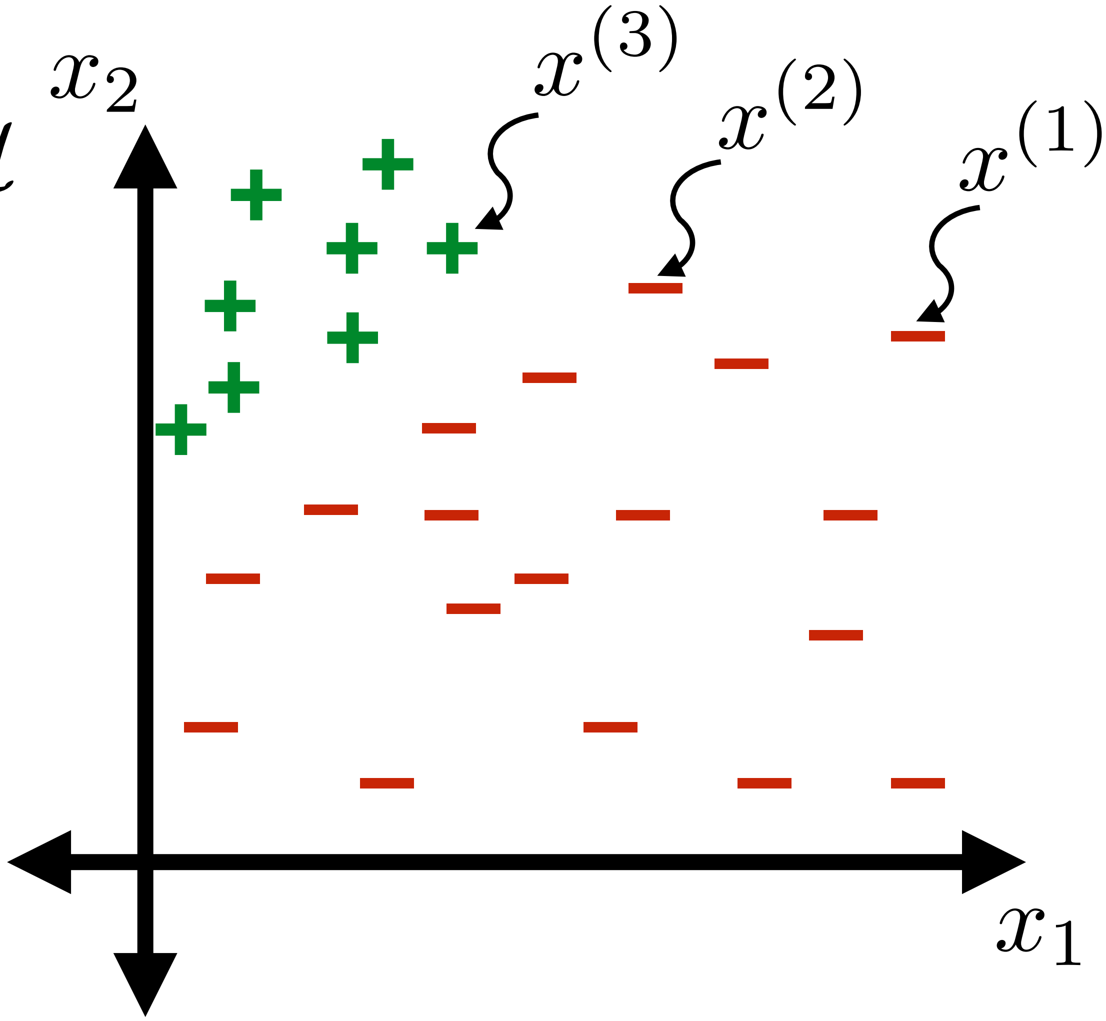
- Example *h*: For any *x*, *h*(*x*) = +1
    - Is this a good hypothesis?

Linear classifiers



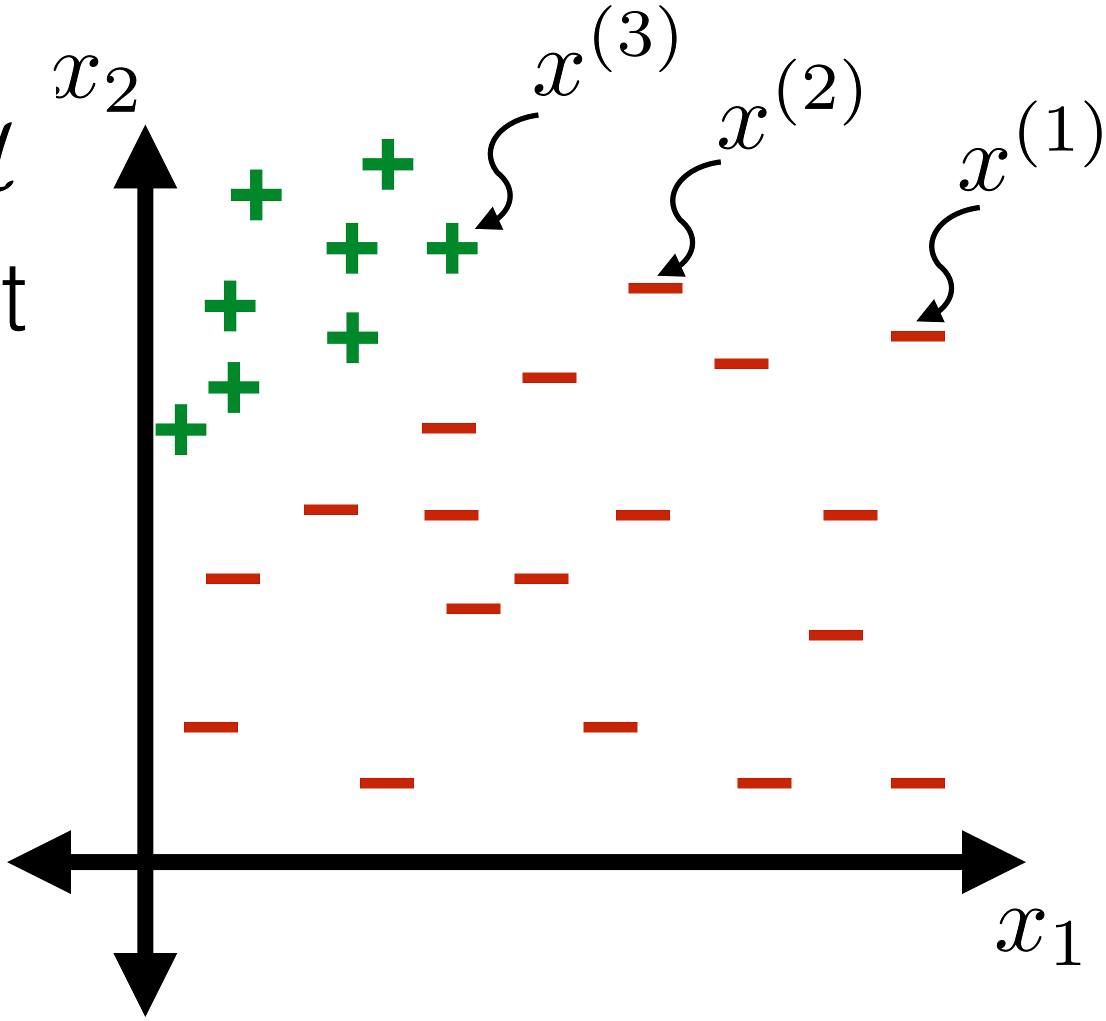
Linear classifiers

- Hypothesis class $\mathcal{H}$: set of $h \in \mathcal{H}$



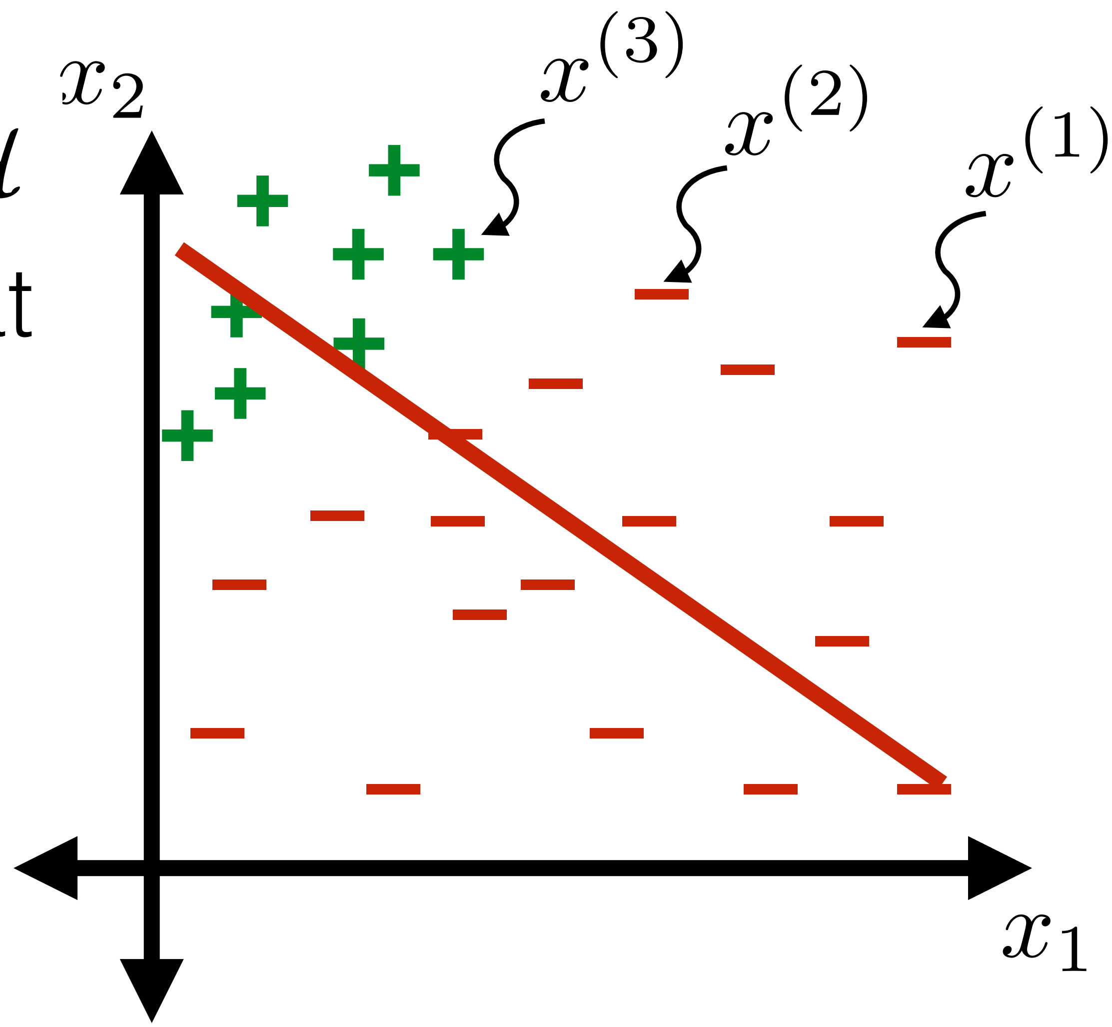
Linear classifiers

- Hypothesis class $\mathcal{H}$: set of $h \in \mathcal{H}$
  - Example $\mathcal{H}$: All hypotheses that label +1 on one side of a line and -1 on the other side



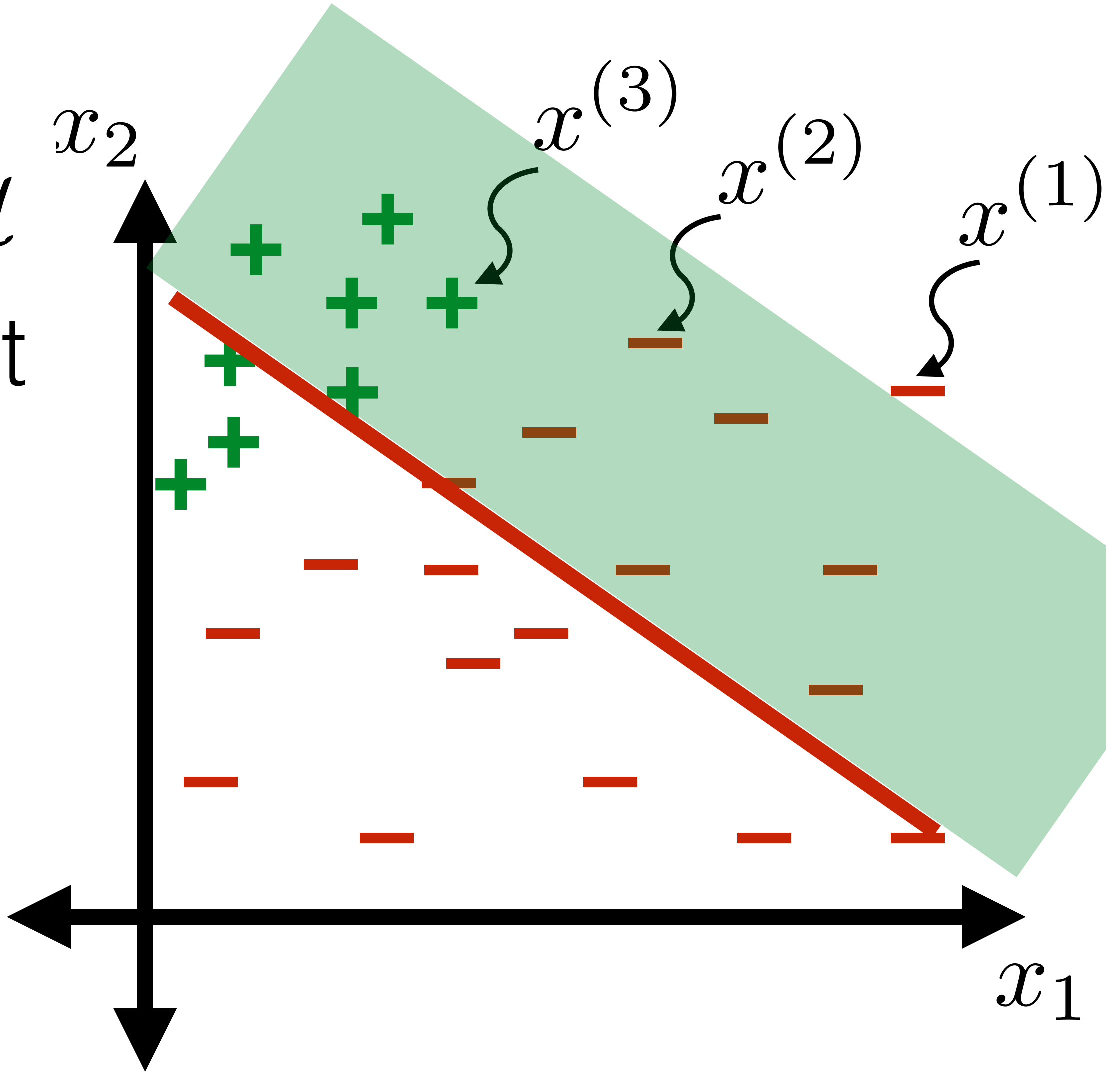
Linear classifiers

- Hypothesis class $\mathcal{H}$: set of $h \in \mathcal{H}$
  - Example $\mathcal{H}$: All hypotheses that label +1 on one side of a line and -1 on the other side



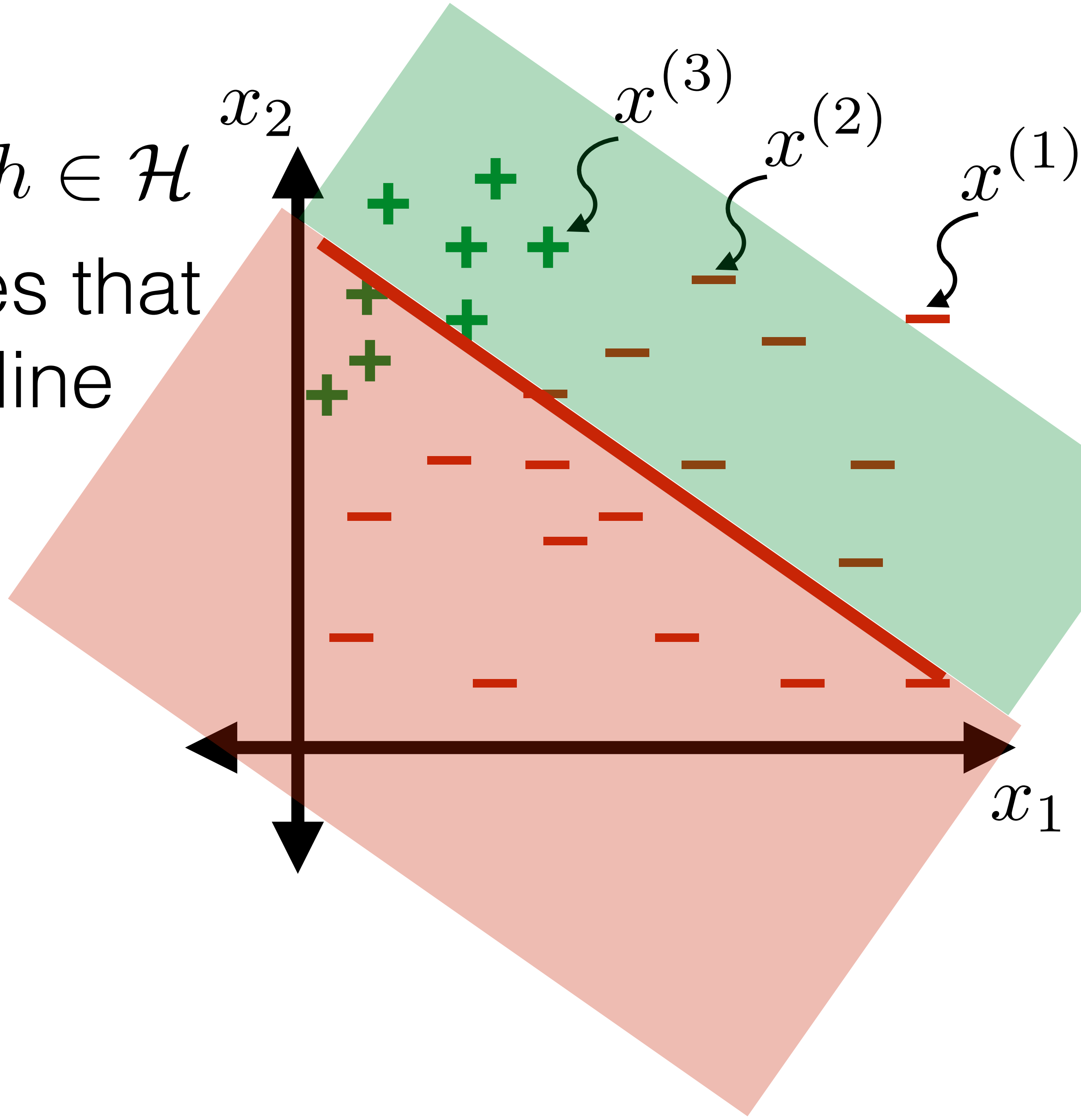
Linear classifiers

- Hypothesis class $\mathcal{H}$: set of $h \in \mathcal{H}$
  - Example $\mathcal{H}$: All hypotheses that label +1 on one side of a line and -1 on the other side



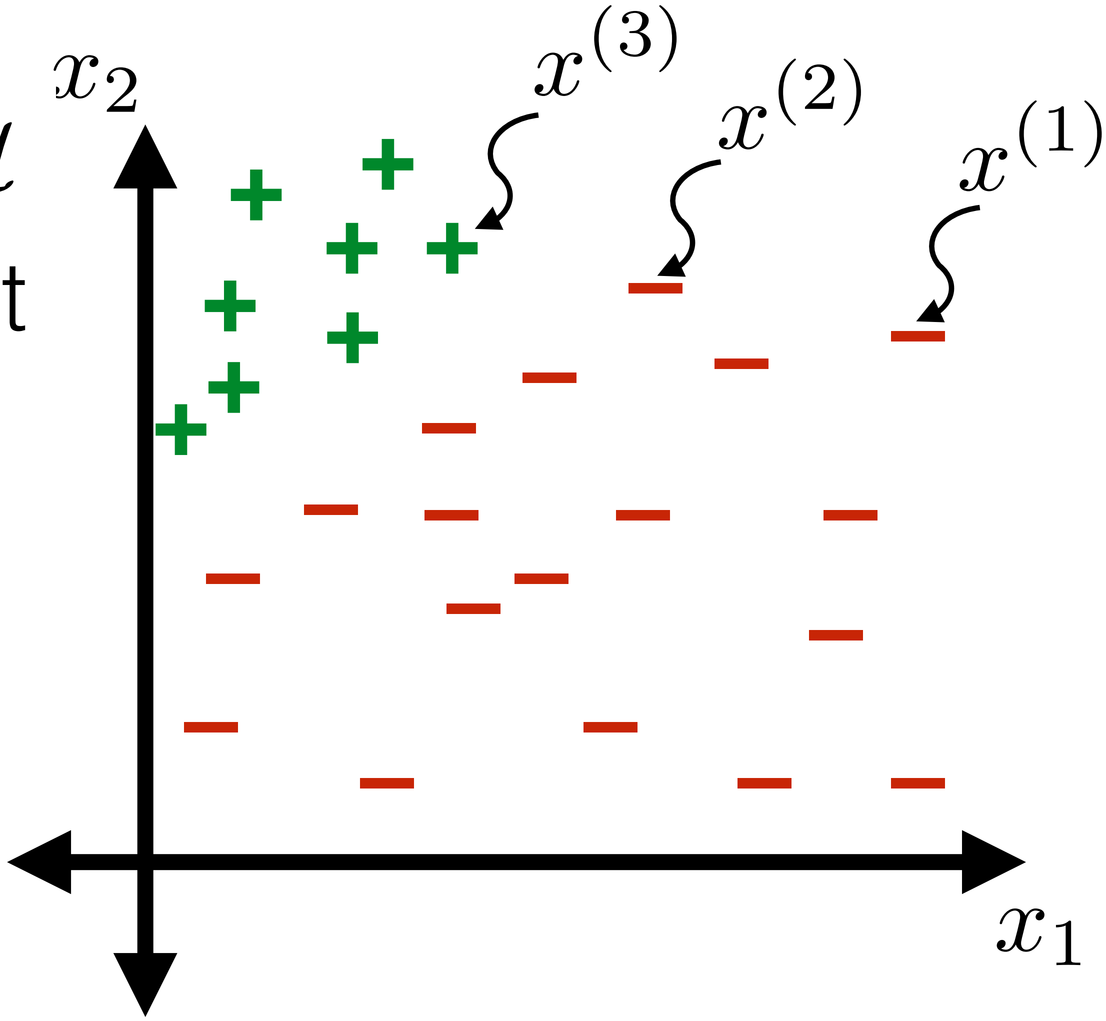
Linear classifiers

- Hypothesis class $\mathcal{H}$: set of $h \in \mathcal{H}$
  - Example $\mathcal{H}$: All hypotheses that label +1 on one side of a line and -1 on the other side



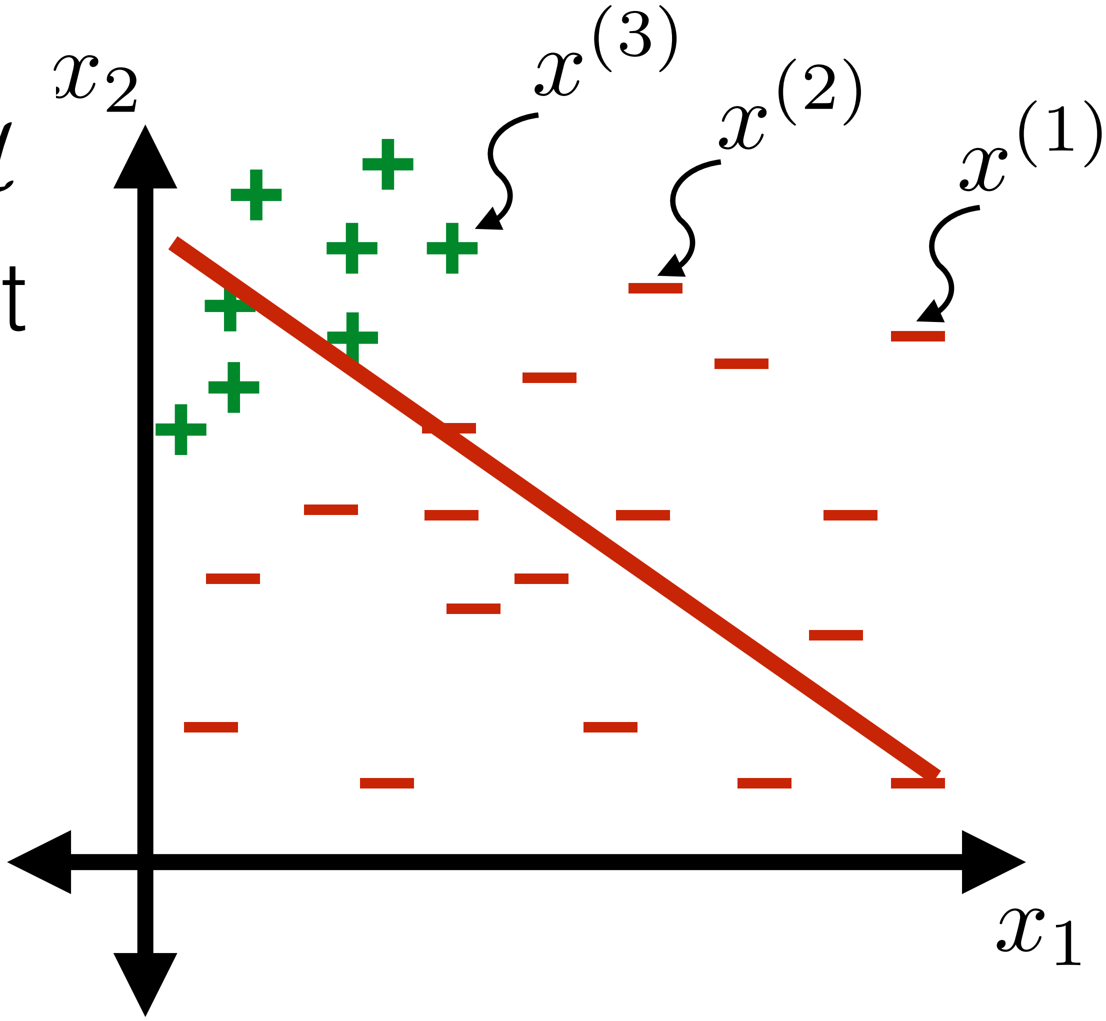
Linear classifiers

- Hypothesis class $\mathcal{H}$: set of $h \in \mathcal{H}$
  - Example $\mathcal{H}$: All hypotheses that label +1 on one side of a line and -1 on the other side



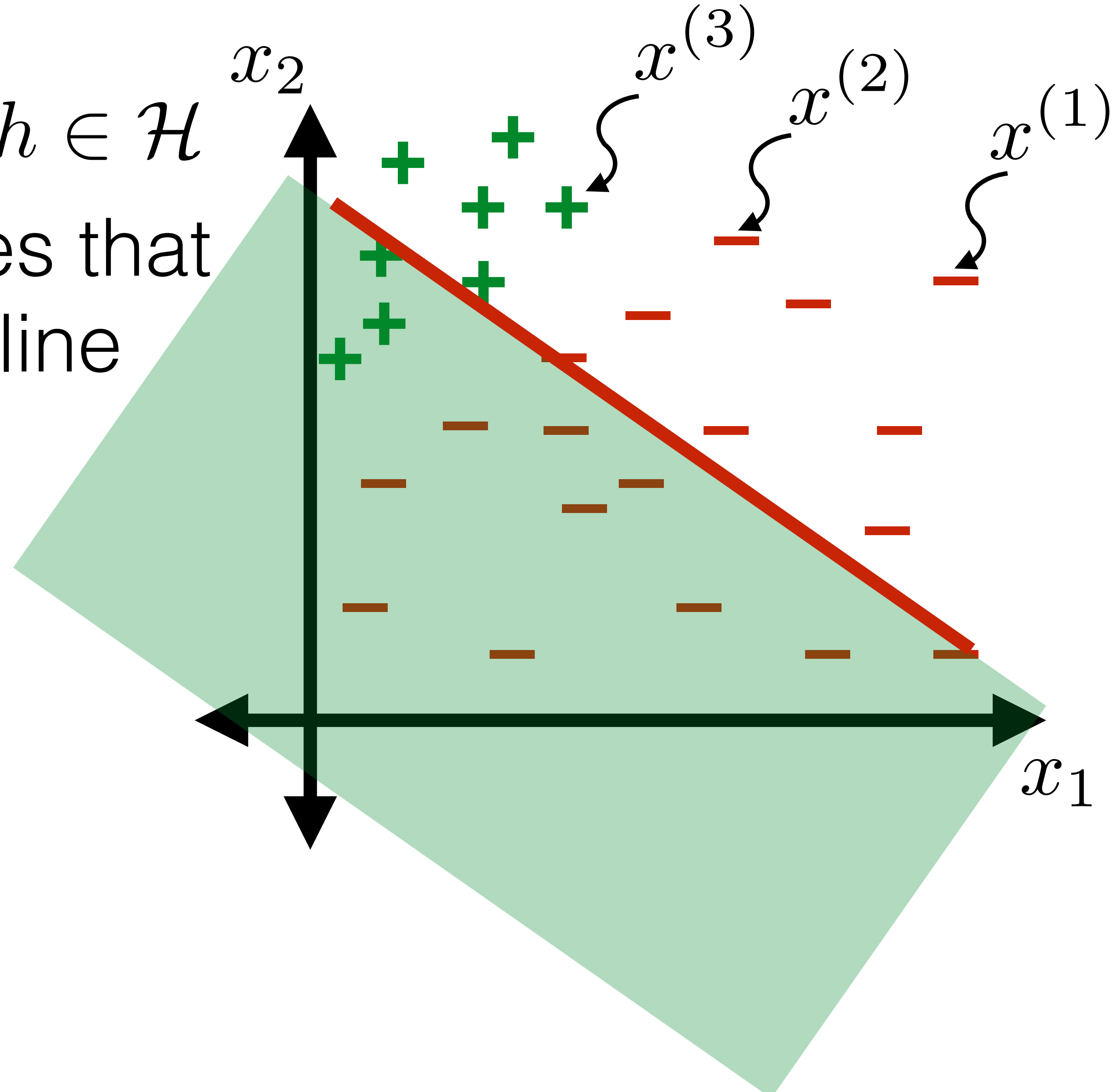
Linear classifiers

- Hypothesis class $\mathcal{H}$: set of $h \in \mathcal{H}$
  - Example $\mathcal{H}$: All hypotheses that label +1 on one side of a line and -1 on the other side



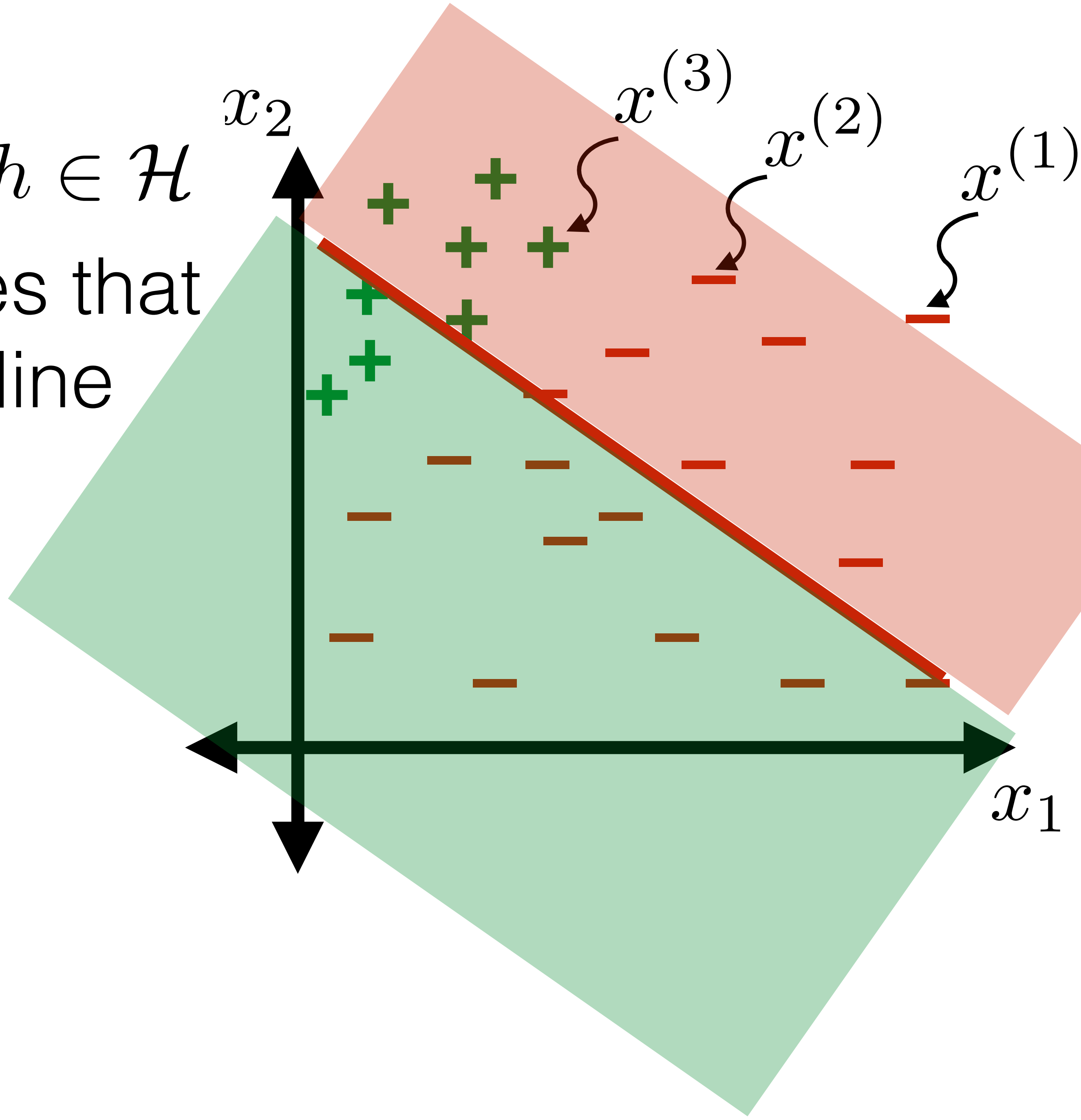
Linear classifiers

- Hypothesis class $\mathcal{H}$: set of $h \in \mathcal{H}$
    - Example $\mathcal{H}$: All hypotheses that label +1 on one side of a line and -1 on the other side



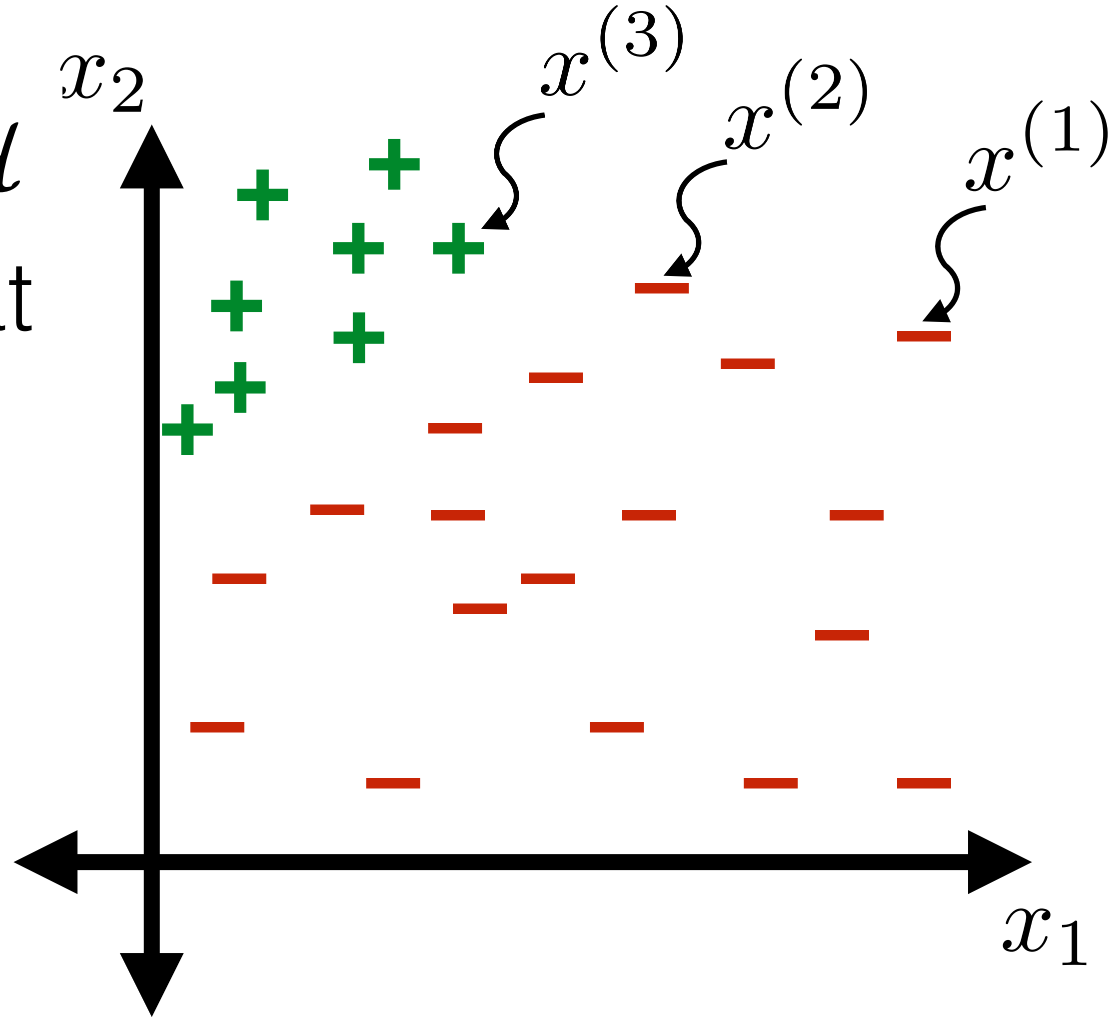
Linear classifiers

- Hypothesis class $\mathcal{H}$: set of $h \in \mathcal{H}$
  - Example $\mathcal{H}$: All hypotheses that label +1 on one side of a line and -1 on the other side



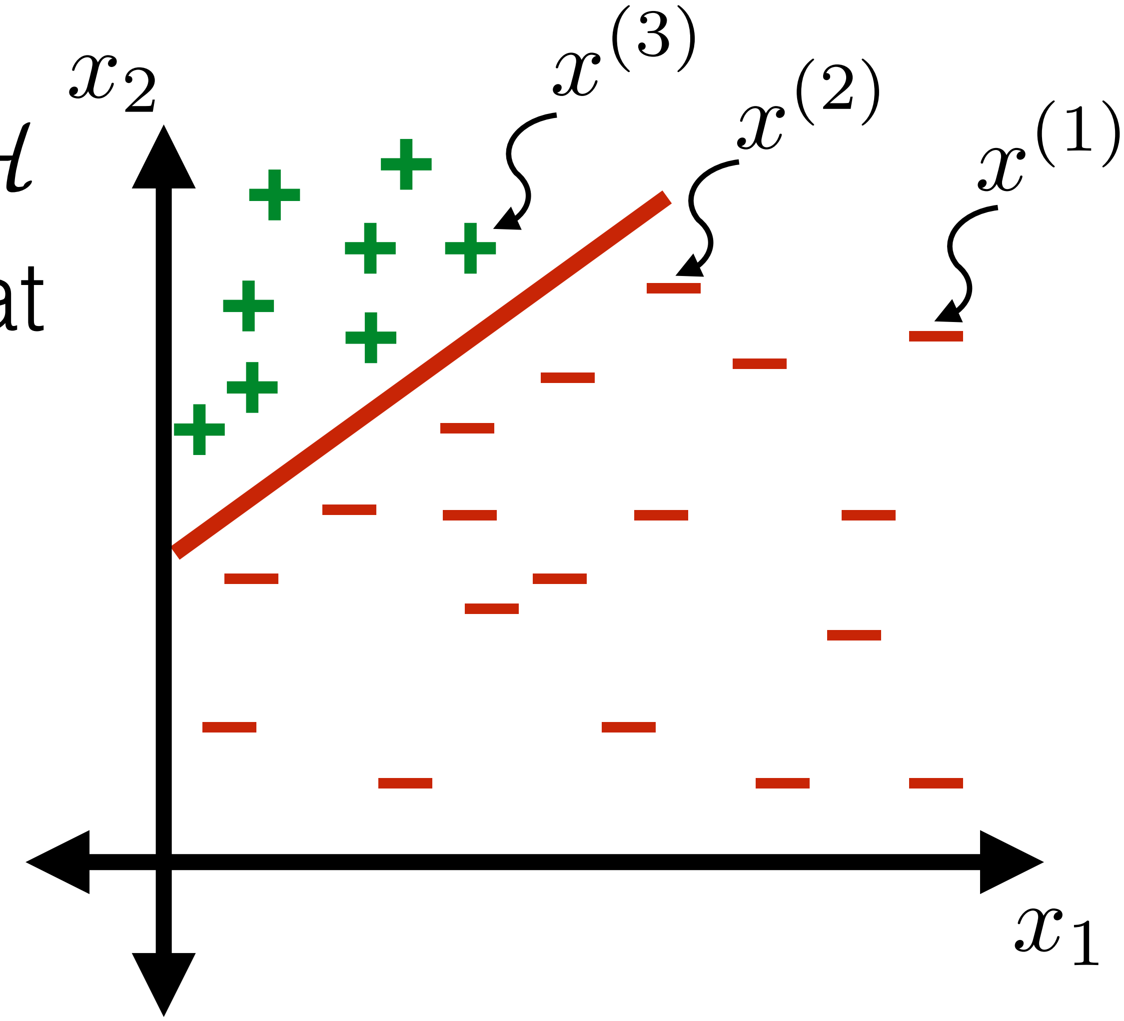
Linear classifiers

- Hypothesis class $\mathcal{H}$: set of $h \in \mathcal{H}$
  - Example $\mathcal{H}$: All hypotheses that label +1 on one side of a line and -1 on the other side



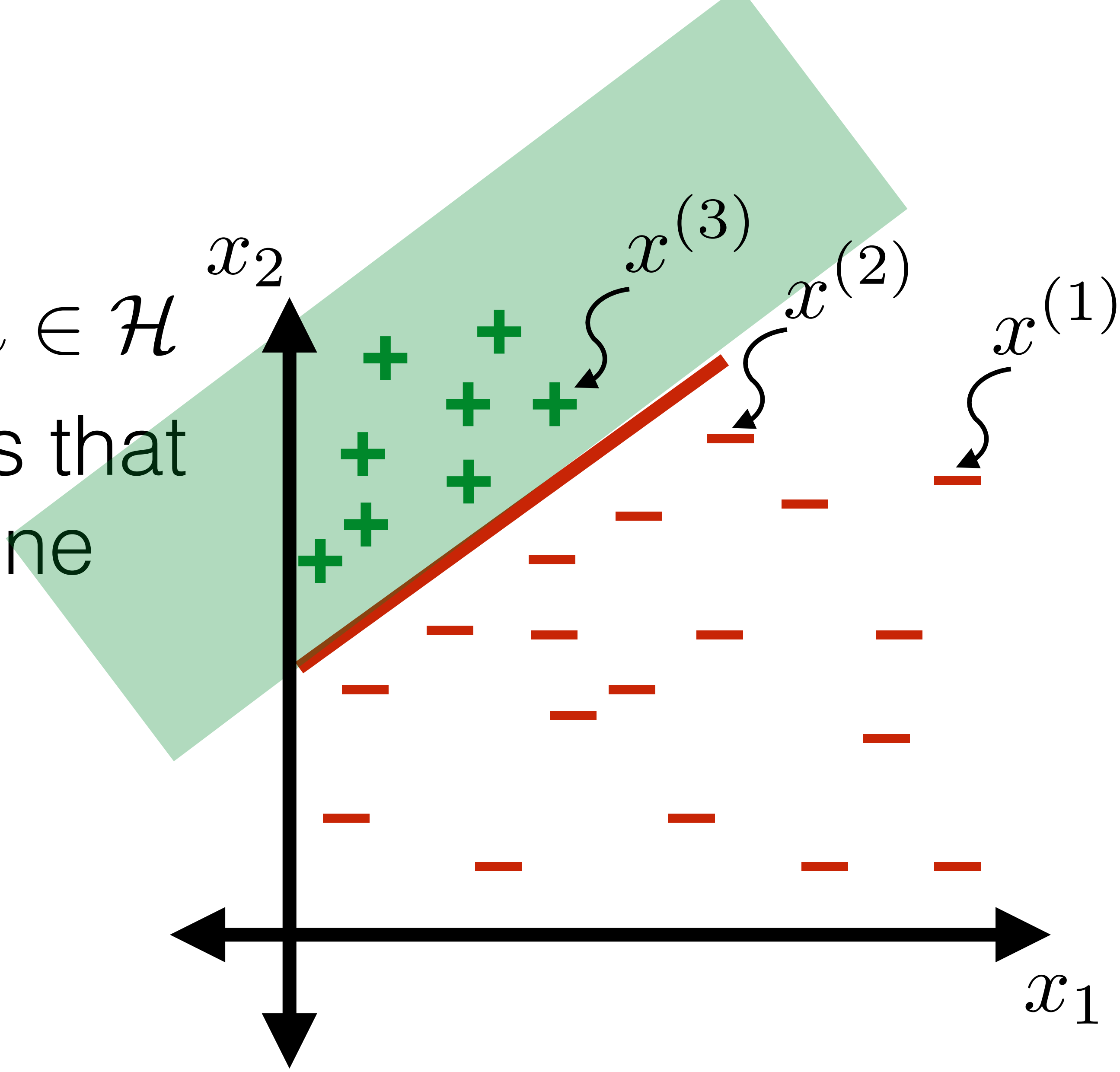
Linear classifiers

- Hypothesis class $\mathcal{H}$: set of $h \in \mathcal{H}$
  - Example $\mathcal{H}$: All hypotheses that label +1 on one side of a line and -1 on the other side



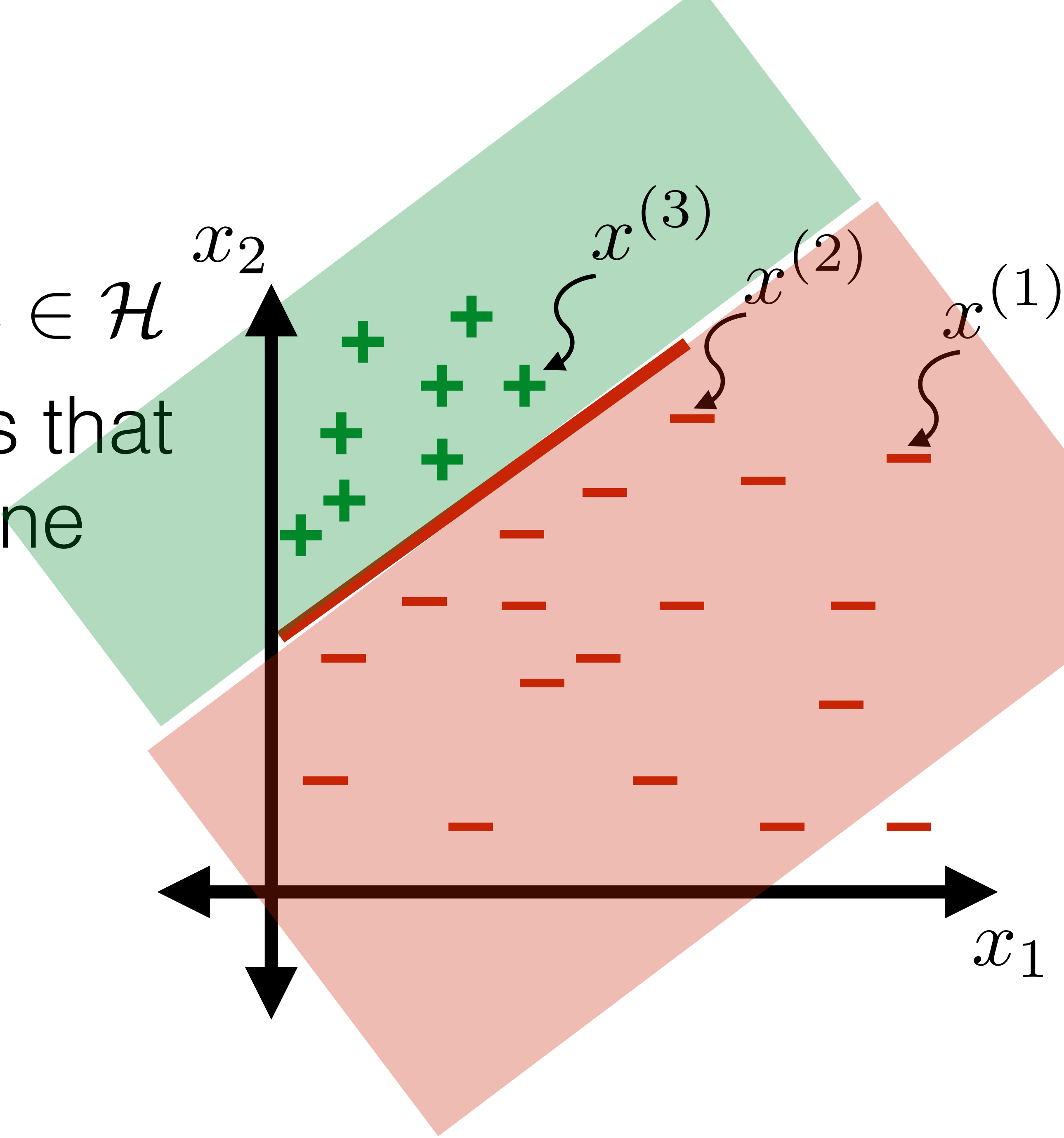
Linear classifiers

- Hypothesis class $\mathcal{H}$: set of $h \in \mathcal{H}$
  - Example $\mathcal{H}$: All hypotheses that label +1 on one side of a line and -1 on the other side



Linear classifiers

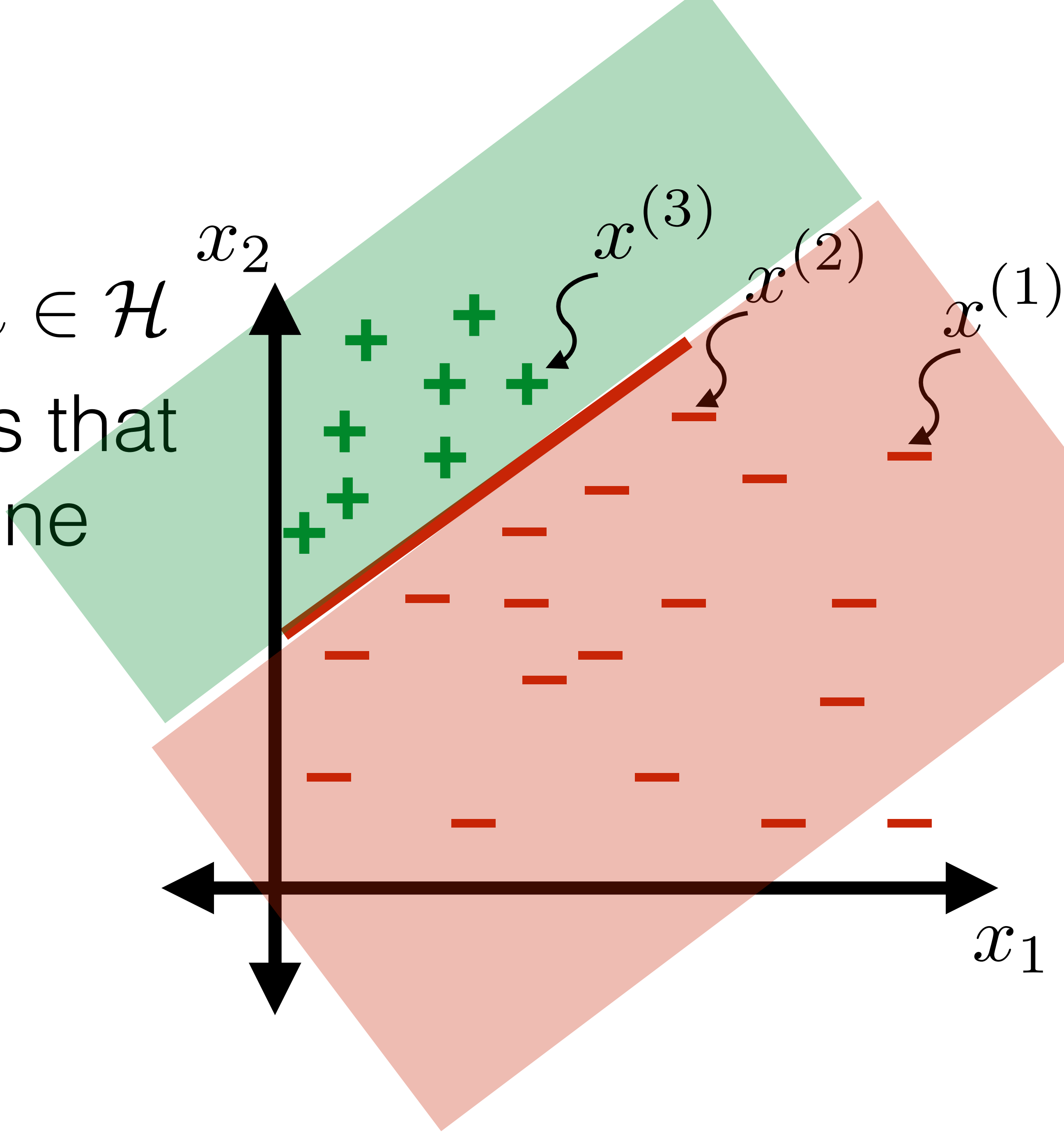
- Hypothesis class $\mathcal{H}$: set of $h \in \mathcal{H}$
  - Example $\mathcal{H}$: All hypotheses that label +1 on one side of a line and -1 on the other side



Linear classifiers

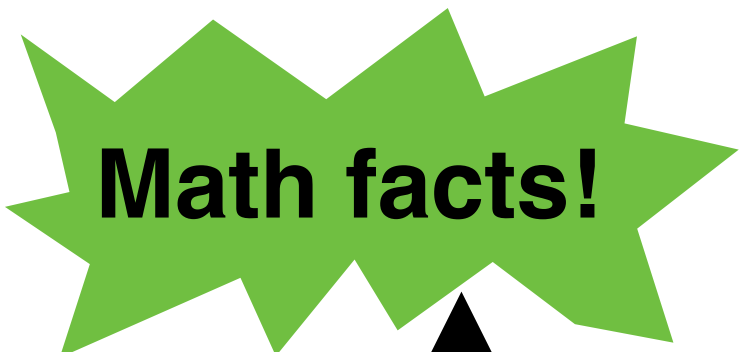
- Hypothesis class $\mathcal{H}$: set of $h \in \mathcal{H}$
  - Example $\mathcal{H}$: All hypotheses that label +1 on one side of a line and -1 on the other side

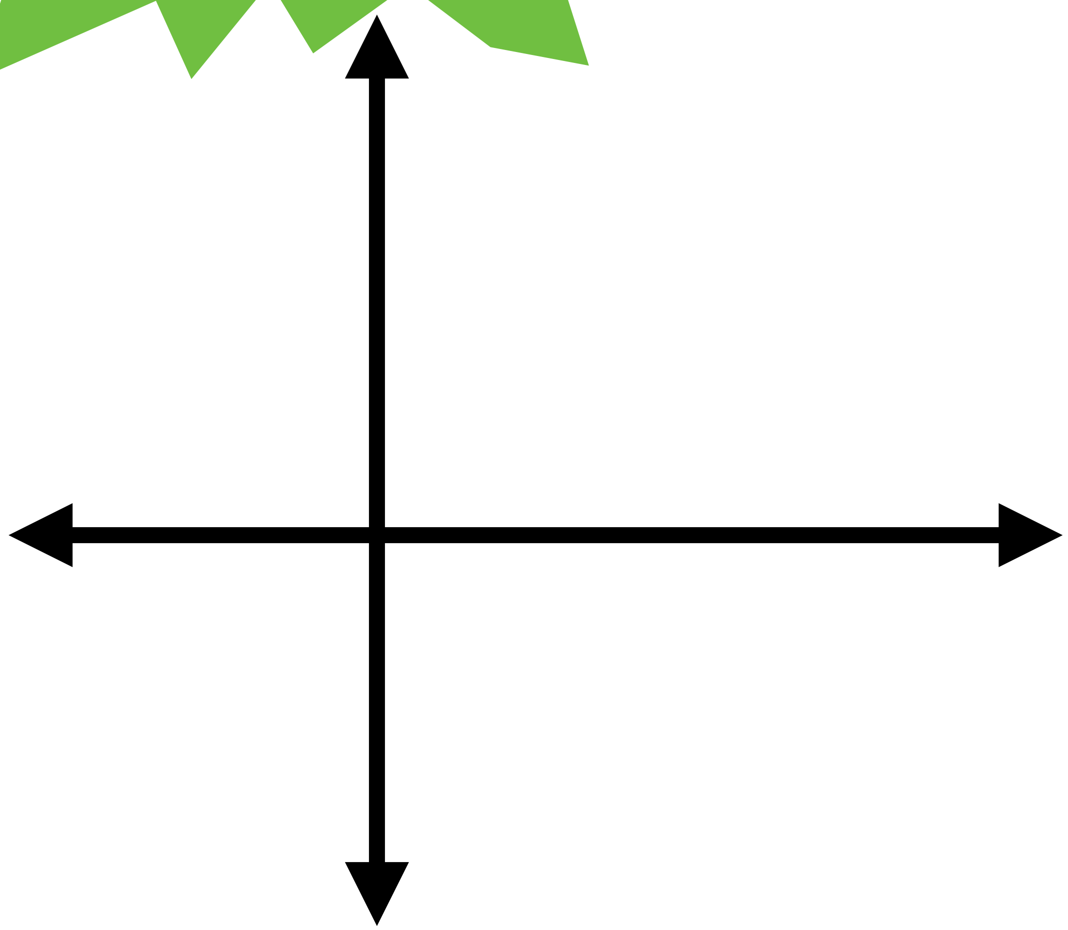
Math facts!

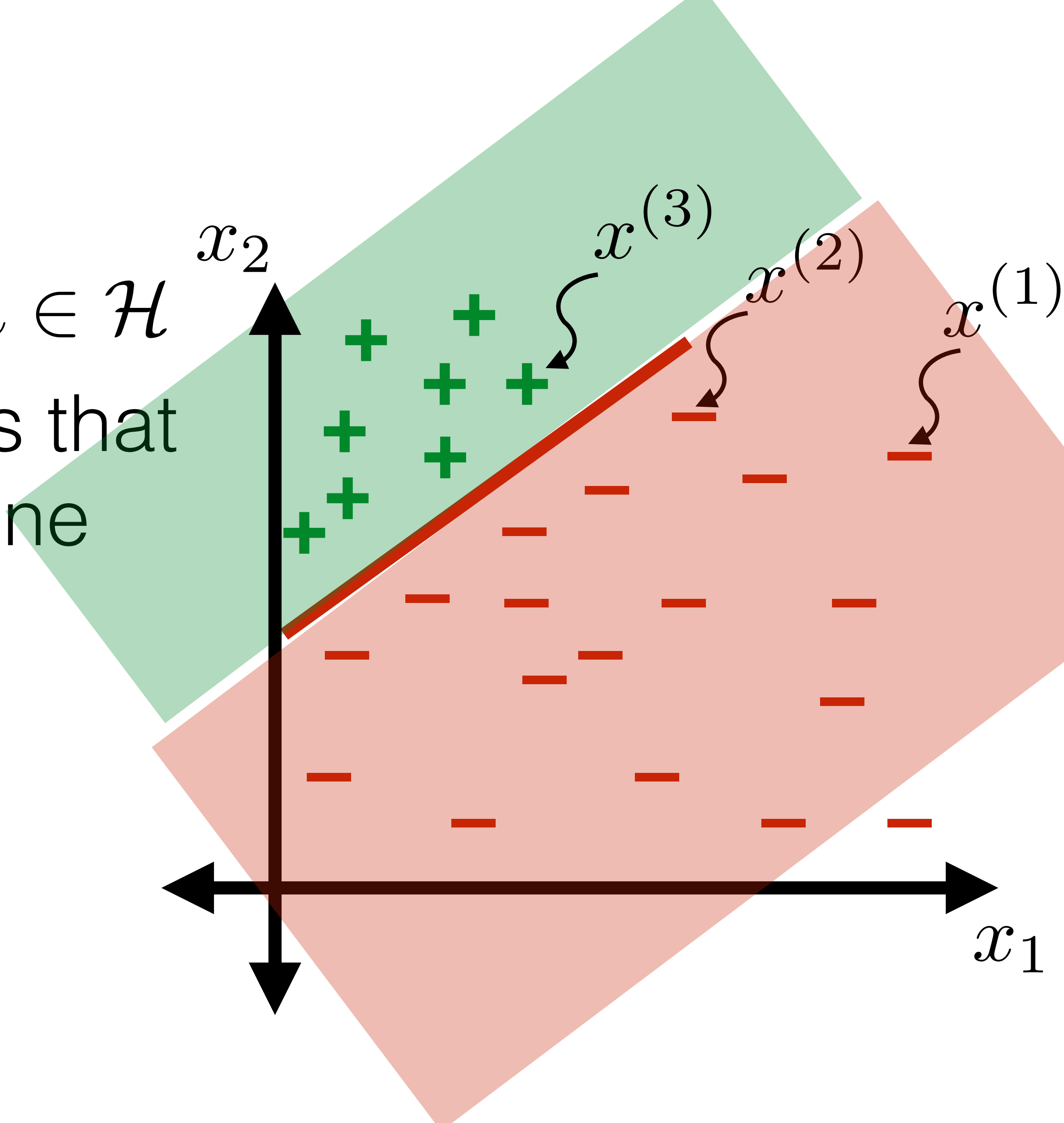


Linear classifiers

- Hypothesis class $\mathcal{H}$: set of $h \in \mathcal{H}$
  - Example $\mathcal{H}$: All hypotheses that label +1 on one side of a line and -1 on the other side

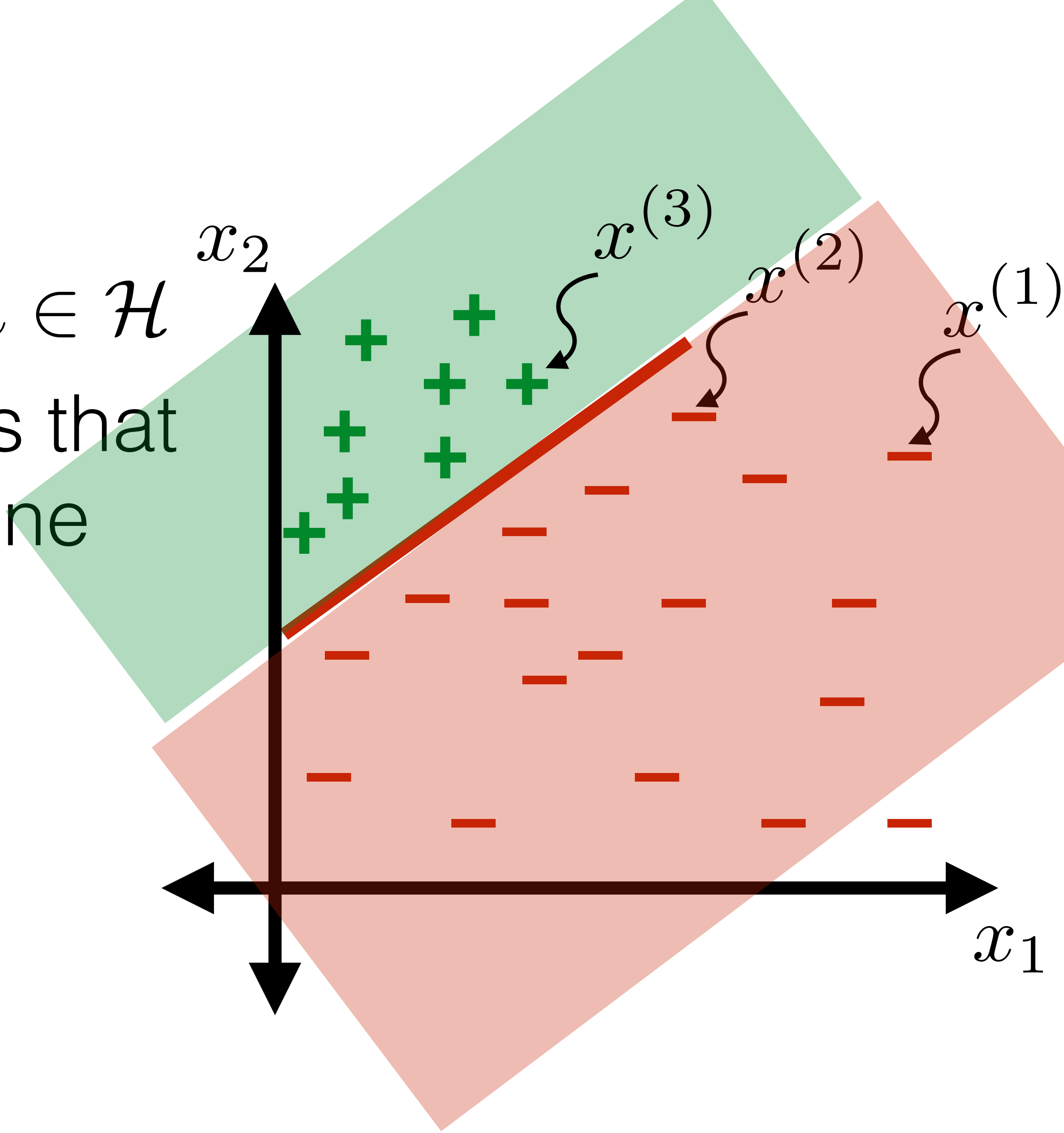


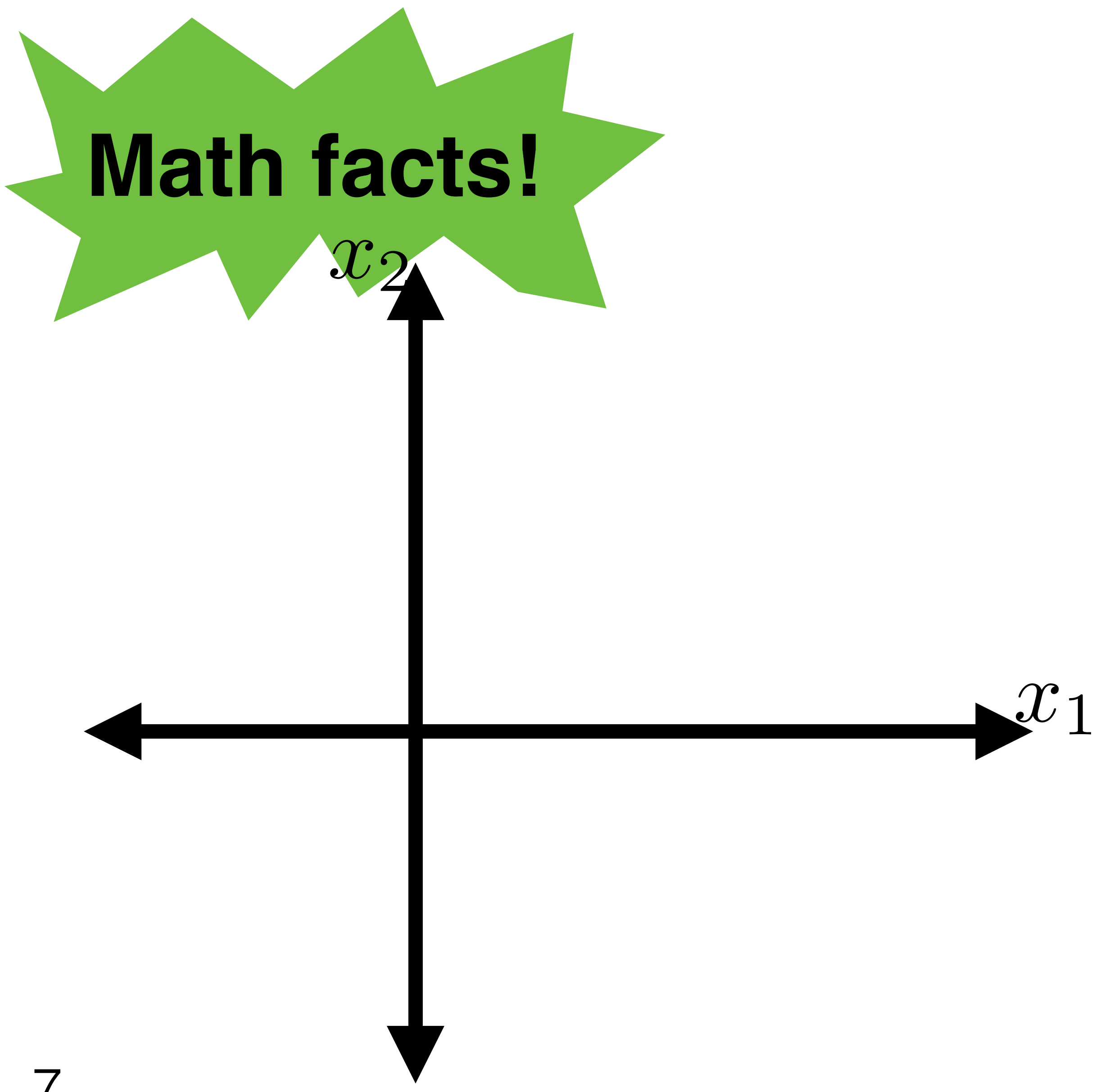




Linear classifiers

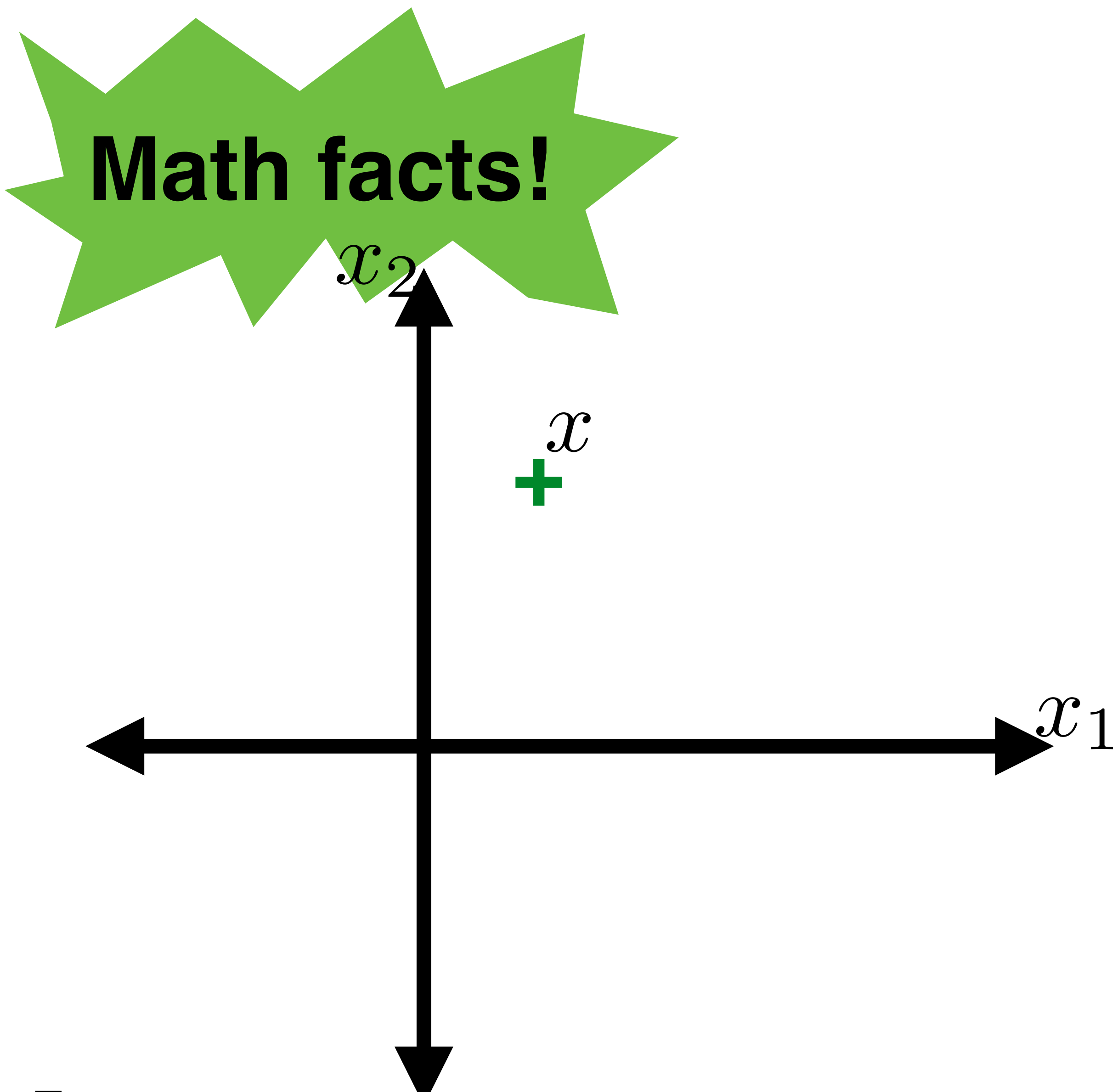
- Hypothesis class $\mathcal{H}$: set of $h \in \mathcal{H}$
  - Example $\mathcal{H}$: All hypotheses that label +1 on one side of a line and -1 on the other side

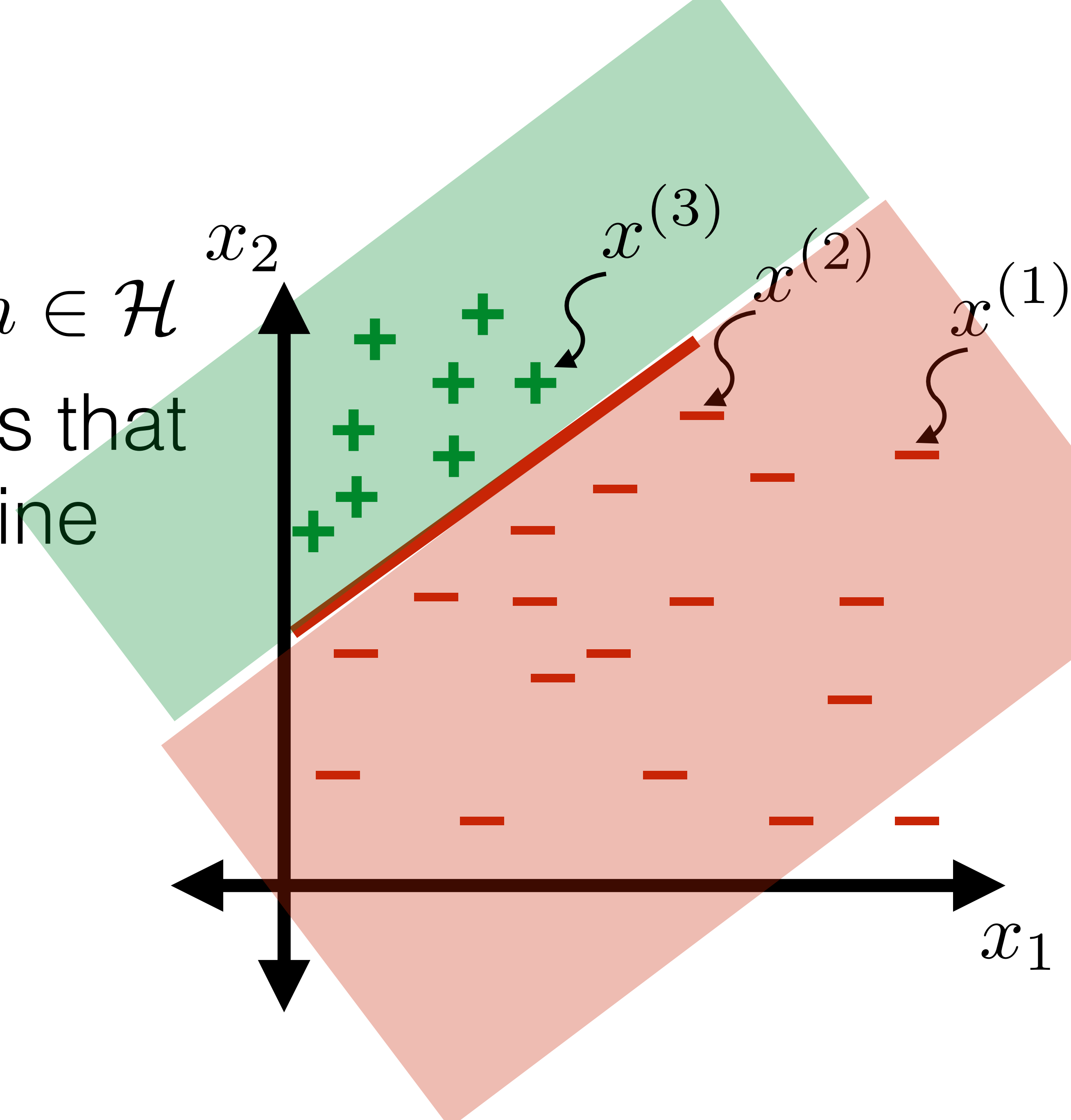




Linear classifiers

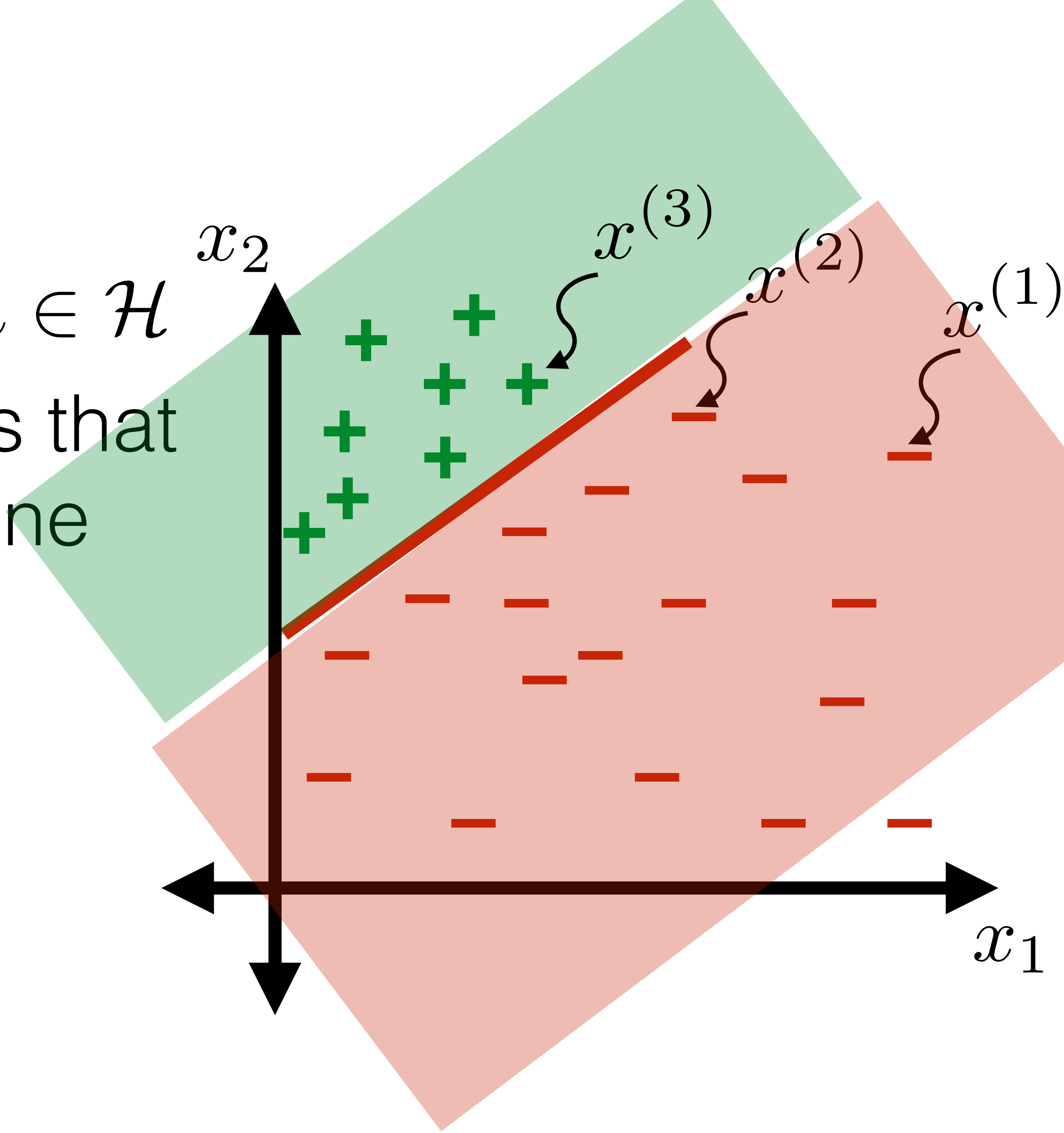
- Hypothesis class $\mathcal{H}$: set of $h \in \mathcal{H}$
  - Example $\mathcal{H}$: All hypotheses that label +1 on one side of a line and -1 on the other side

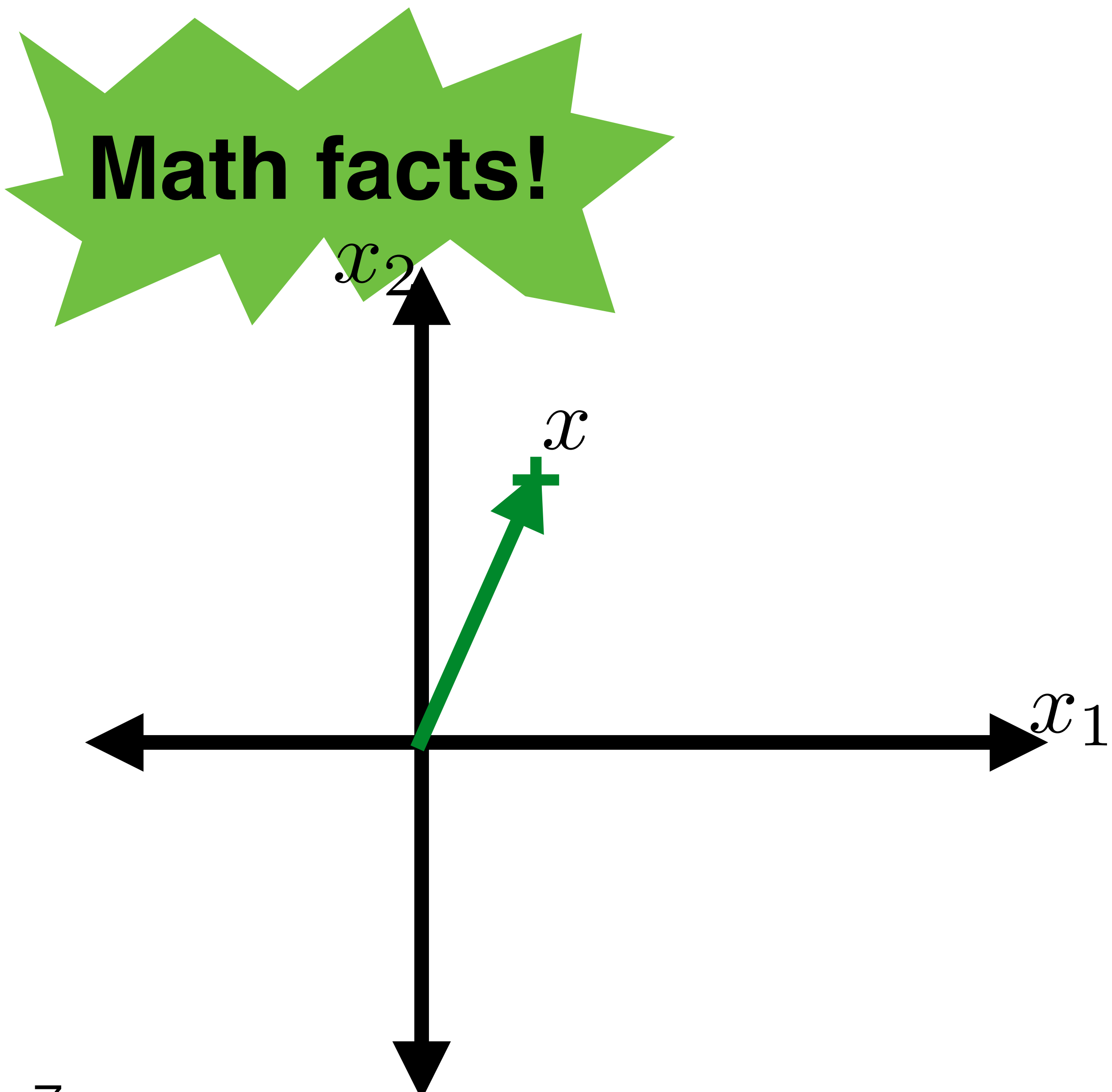




Linear classifiers

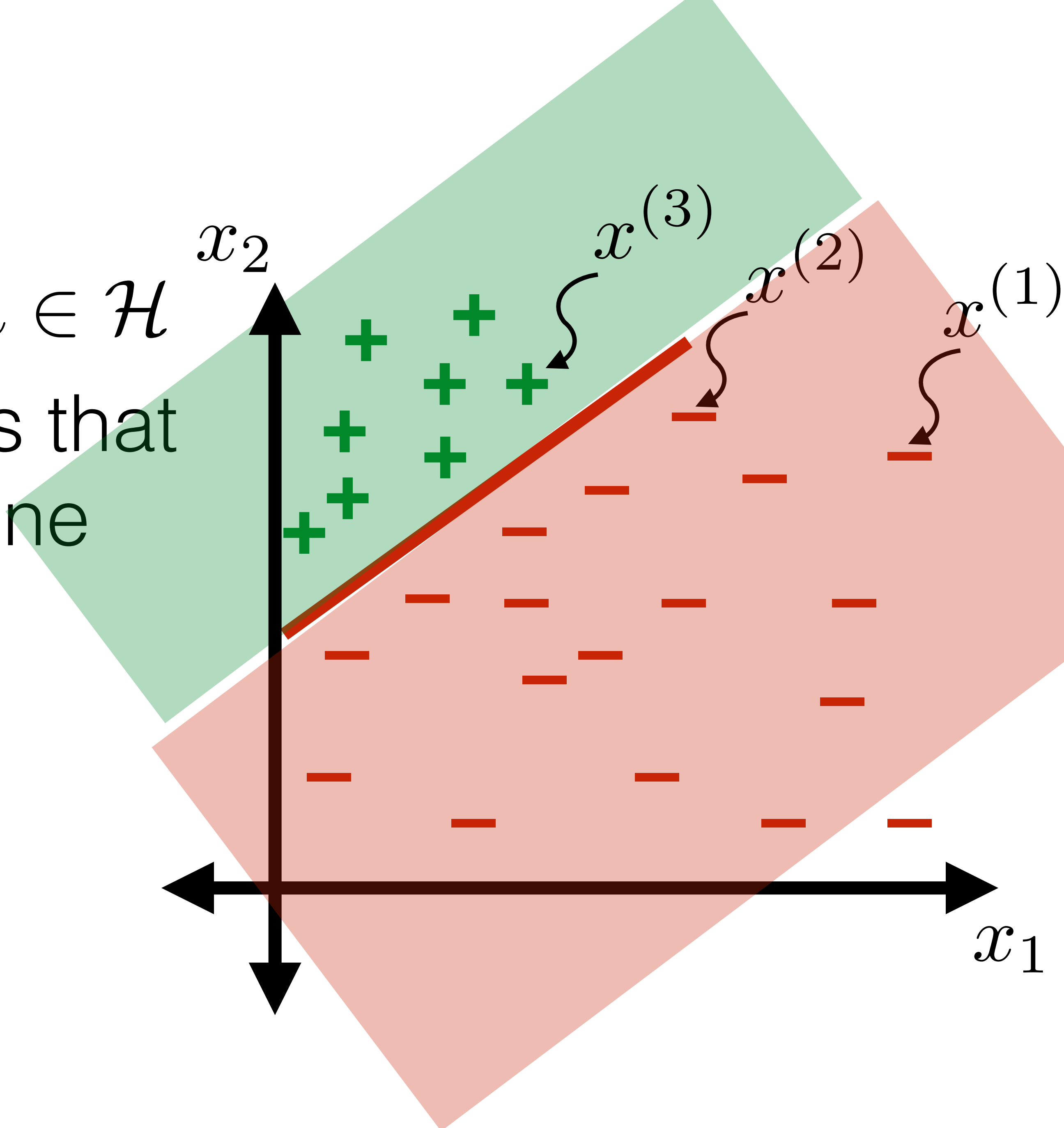
- Hypothesis class $\mathcal{H}$: set of $h \in \mathcal{H}$
  - Example $\mathcal{H}$: All hypotheses that label +1 on one side of a line and -1 on the other side



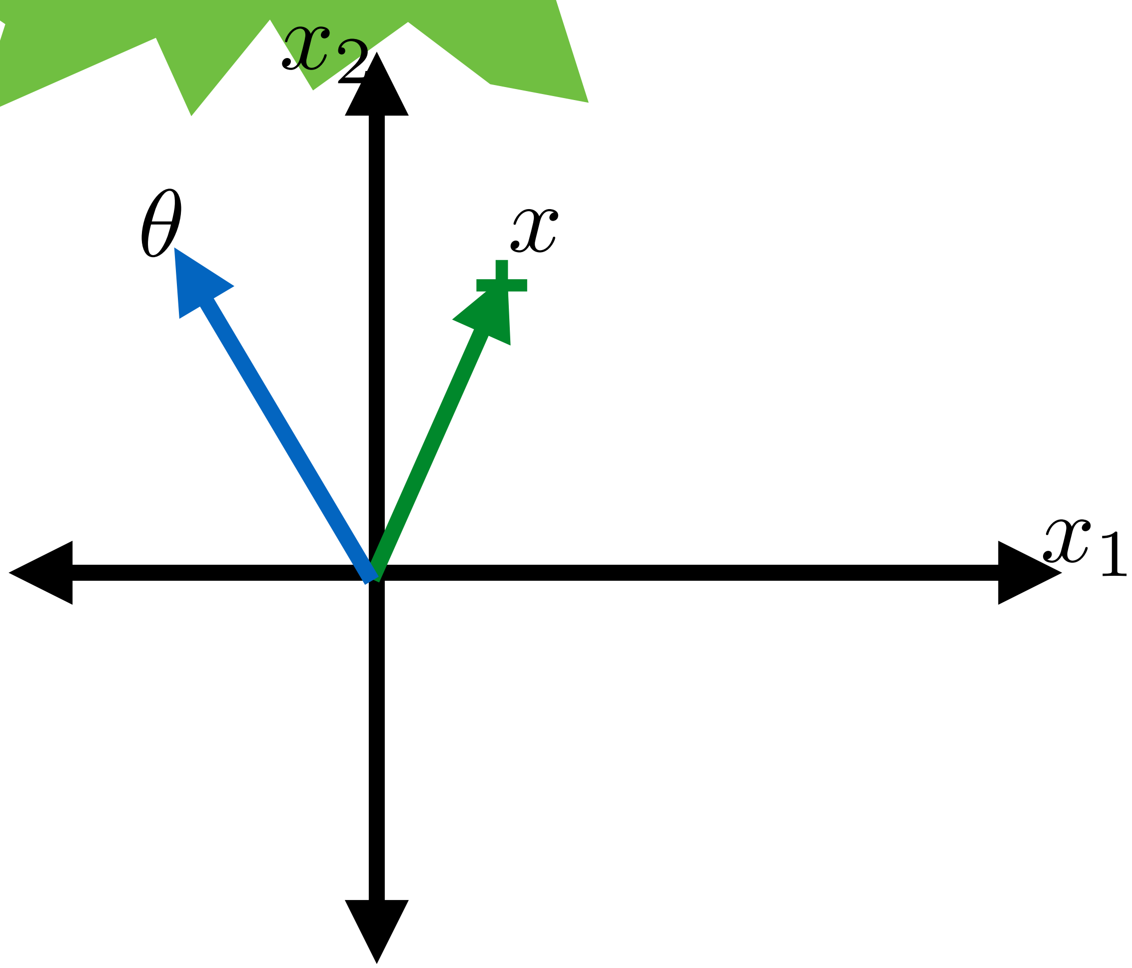


Linear classifiers

- Hypothesis class $\mathcal{H}$: set of $h \in \mathcal{H}$
  - Example $\mathcal{H}$: All hypotheses that label +1 on one side of a line and -1 on the other side

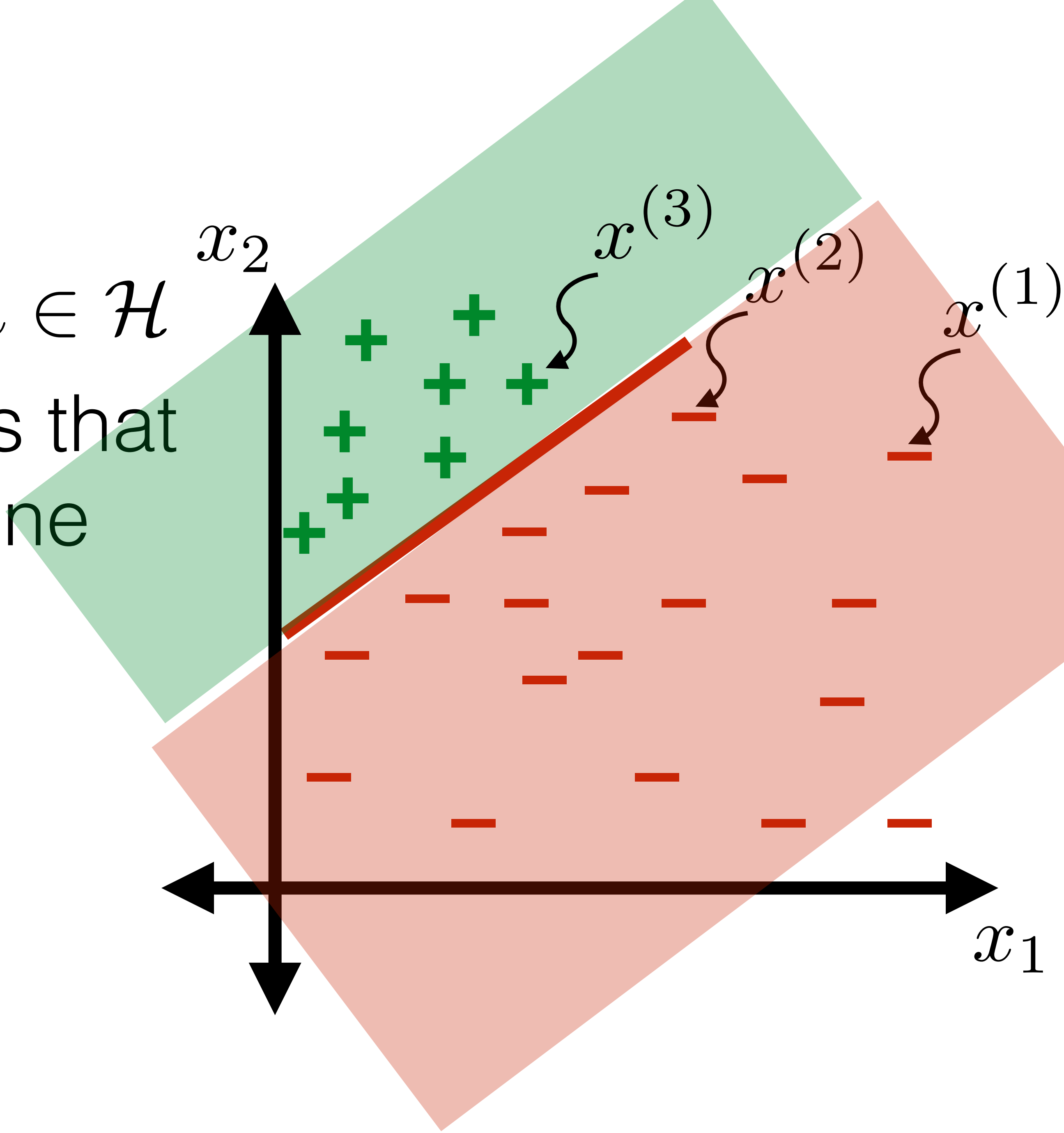


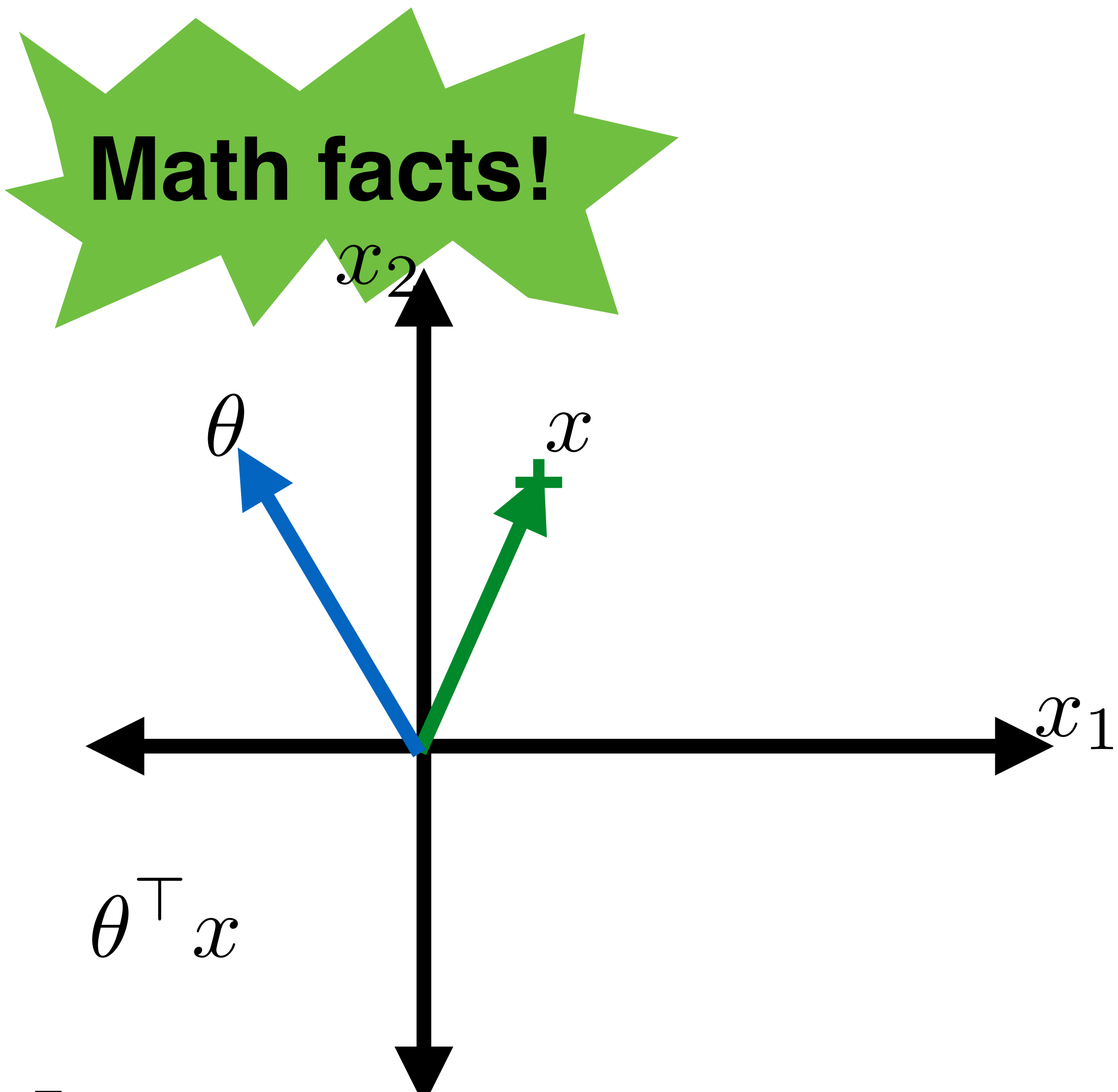
Math facts!



Linear classifiers

- Hypothesis class $\mathcal{H}$: set of $h \in \mathcal{H}$
  - Example $\mathcal{H}$: All hypotheses that label +1 on one side of a line and -1 on the other side

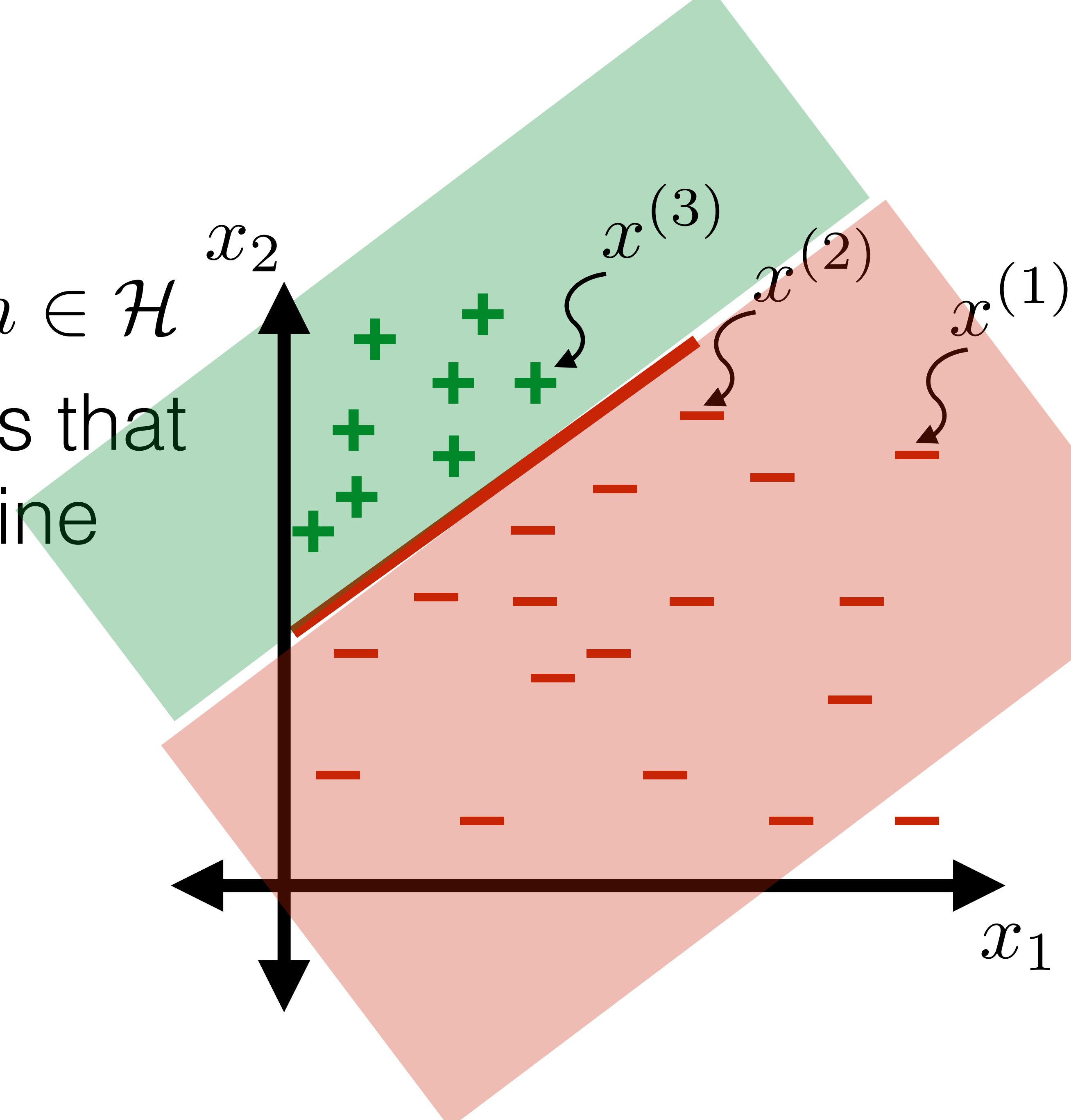




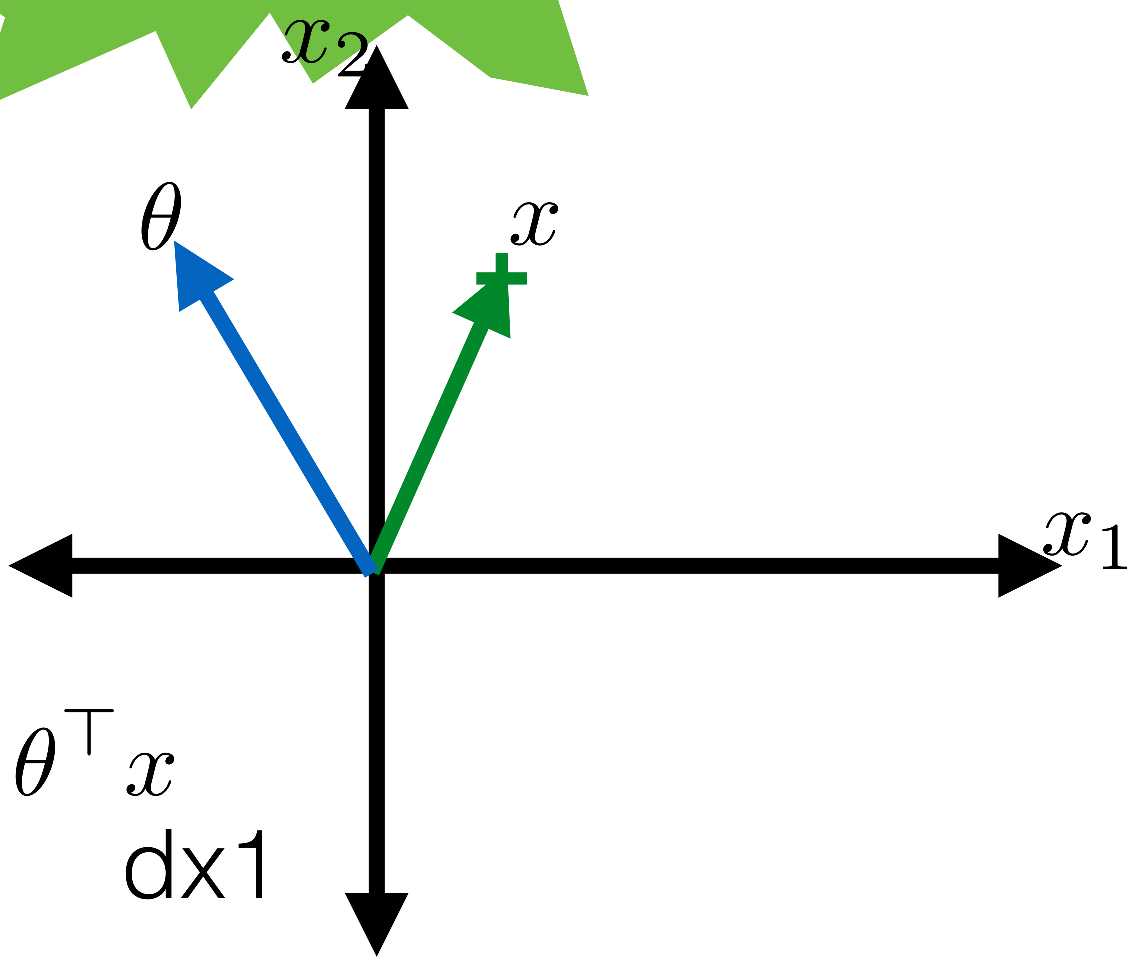
$\theta^\top x$

Linear classifiers

- Hypothesis class $\mathcal{H}$: set of $h \in \mathcal{H}$
  - Example $\mathcal{H}$: All hypotheses that label +1 on one side of a line and -1 on the other side



Math facts!

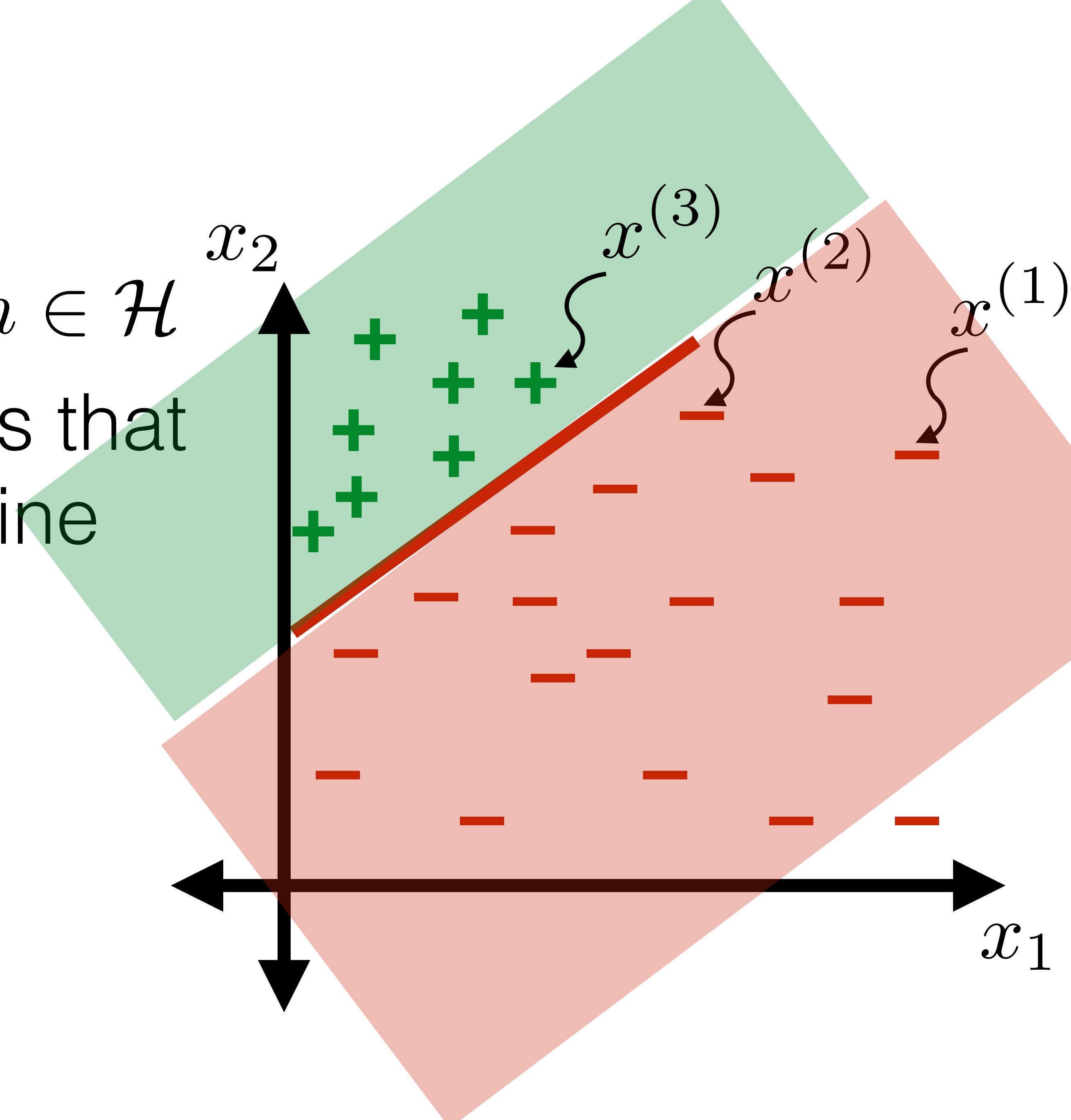


$\theta^\top x$

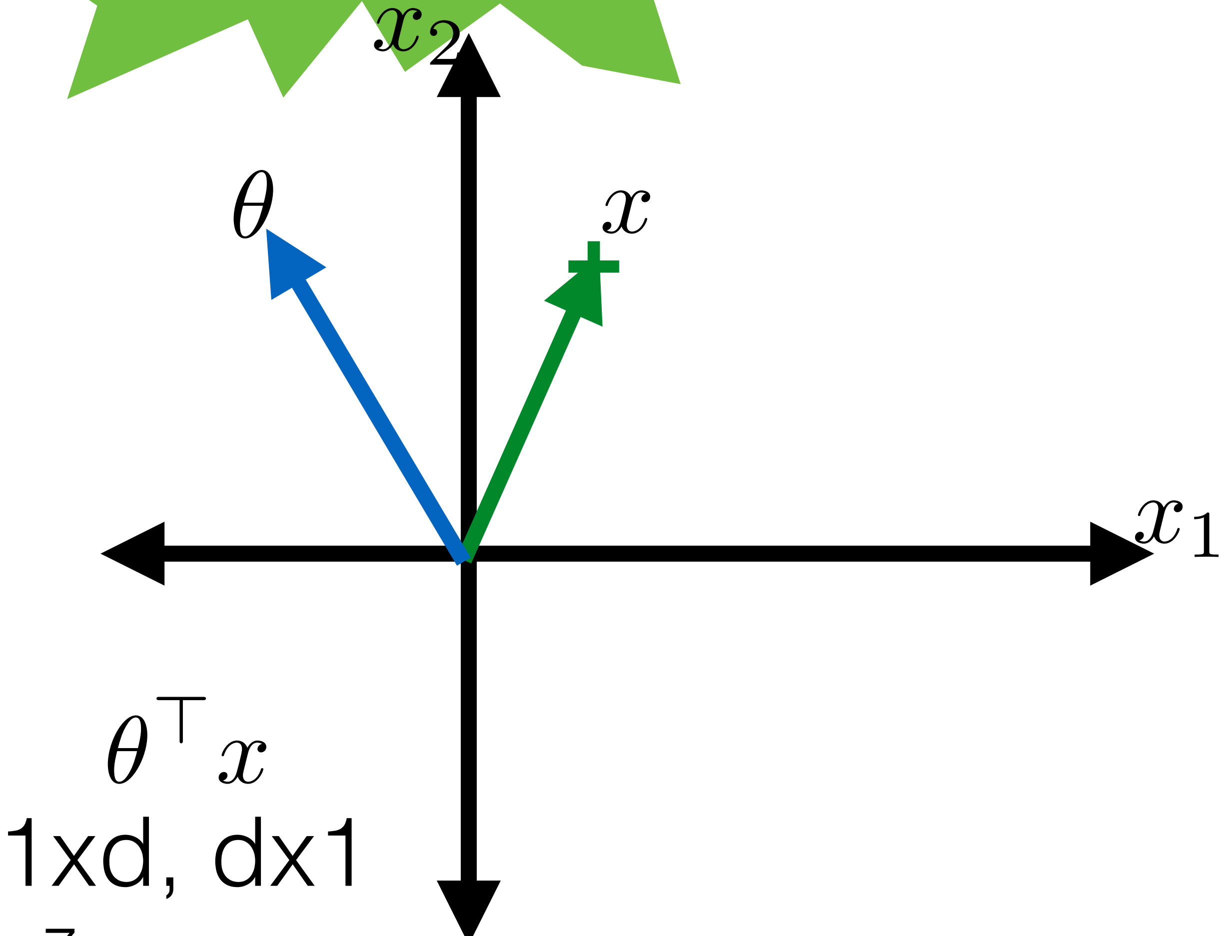
dx1

Linear classifiers

- Hypothesis class $\mathcal{H}$: set of $h \in \mathcal{H}$
  - Example $\mathcal{H}$: All hypotheses that label +1 on one side of a line and -1 on the other side

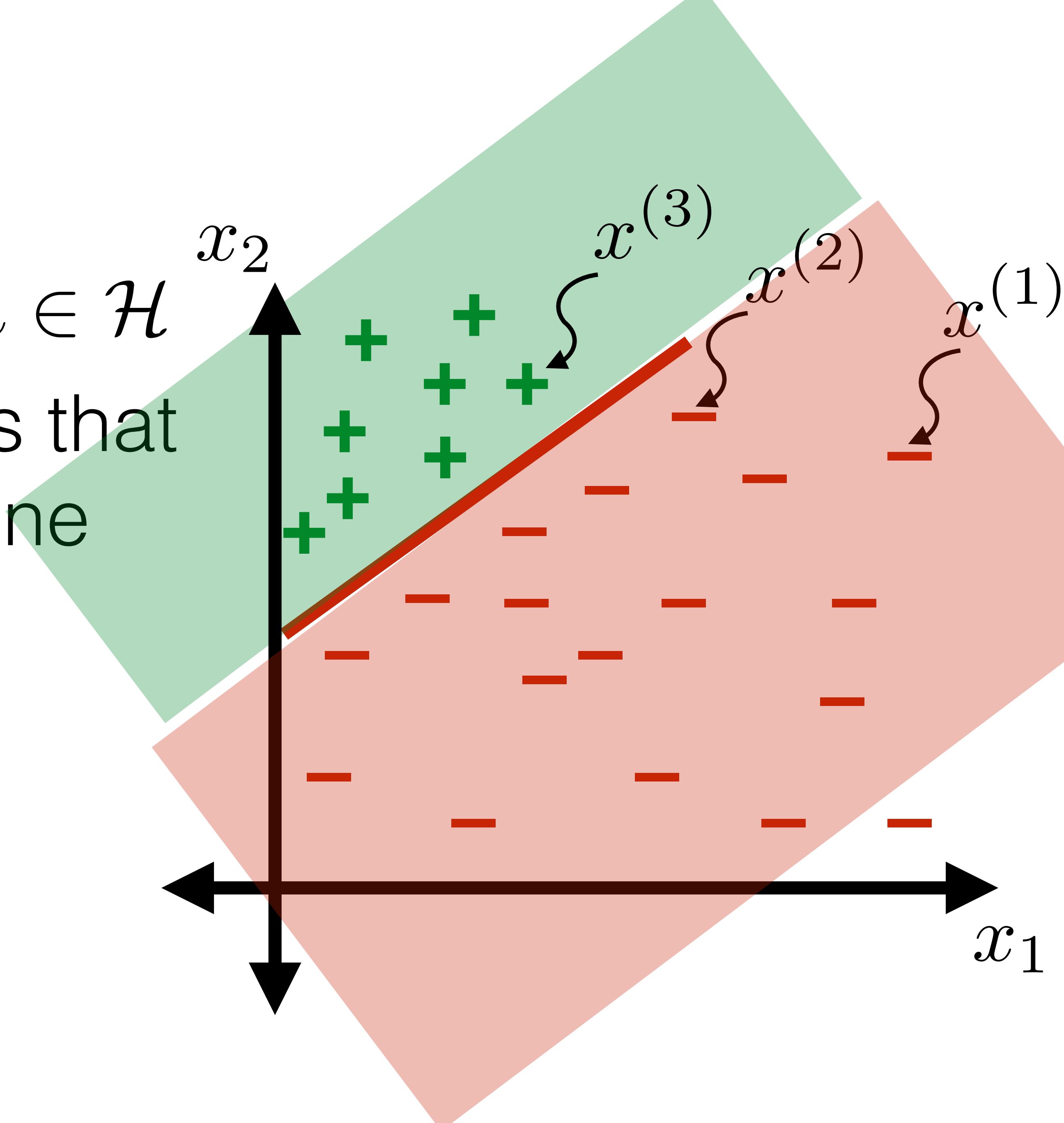


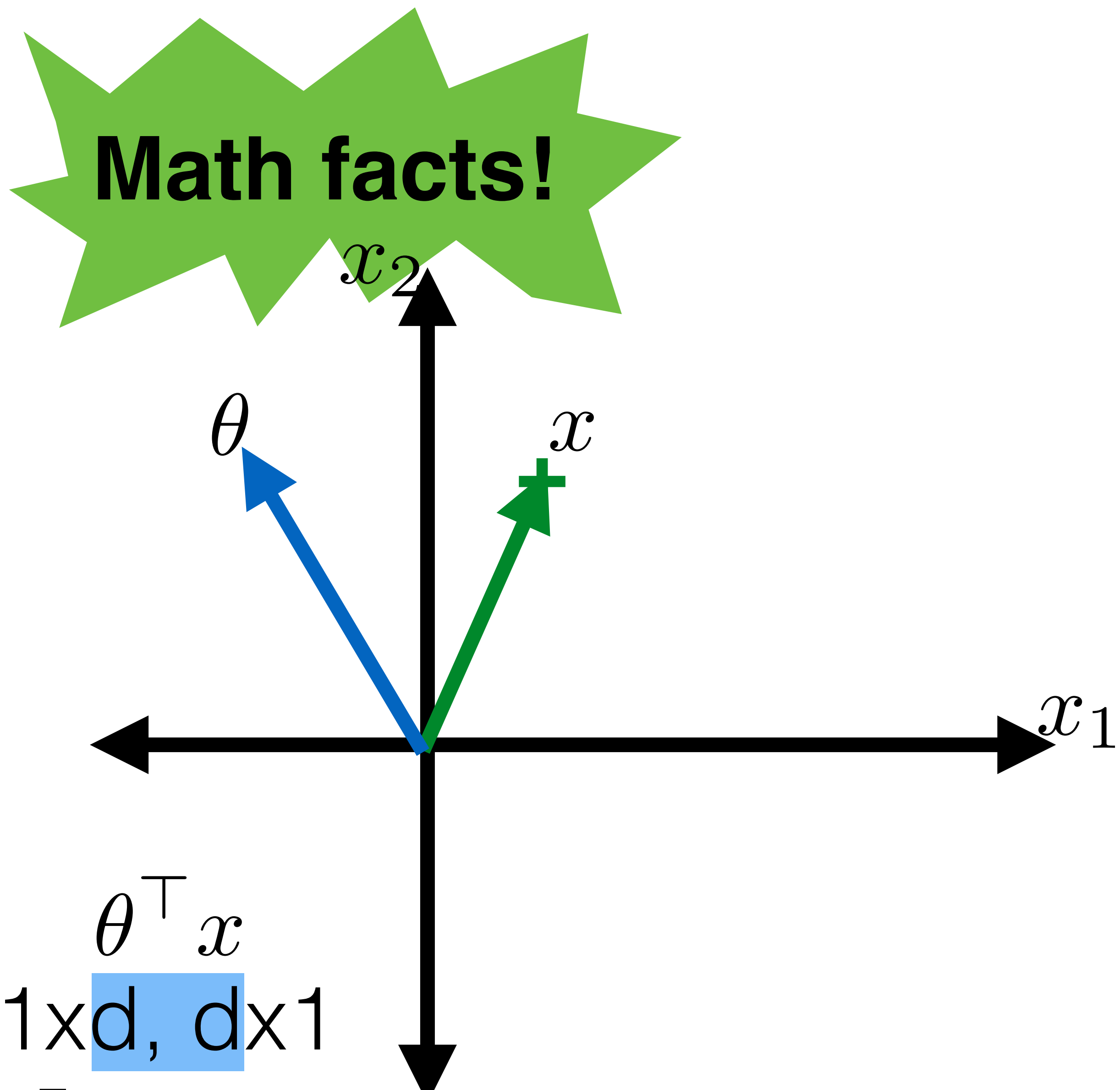
Math facts!



Linear classifiers

- Hypothesis class $\mathcal{H}$: set of $h \in \mathcal{H}$
  - Example $\mathcal{H}$: All hypotheses that label +1 on one side of a line and -1 on the other side



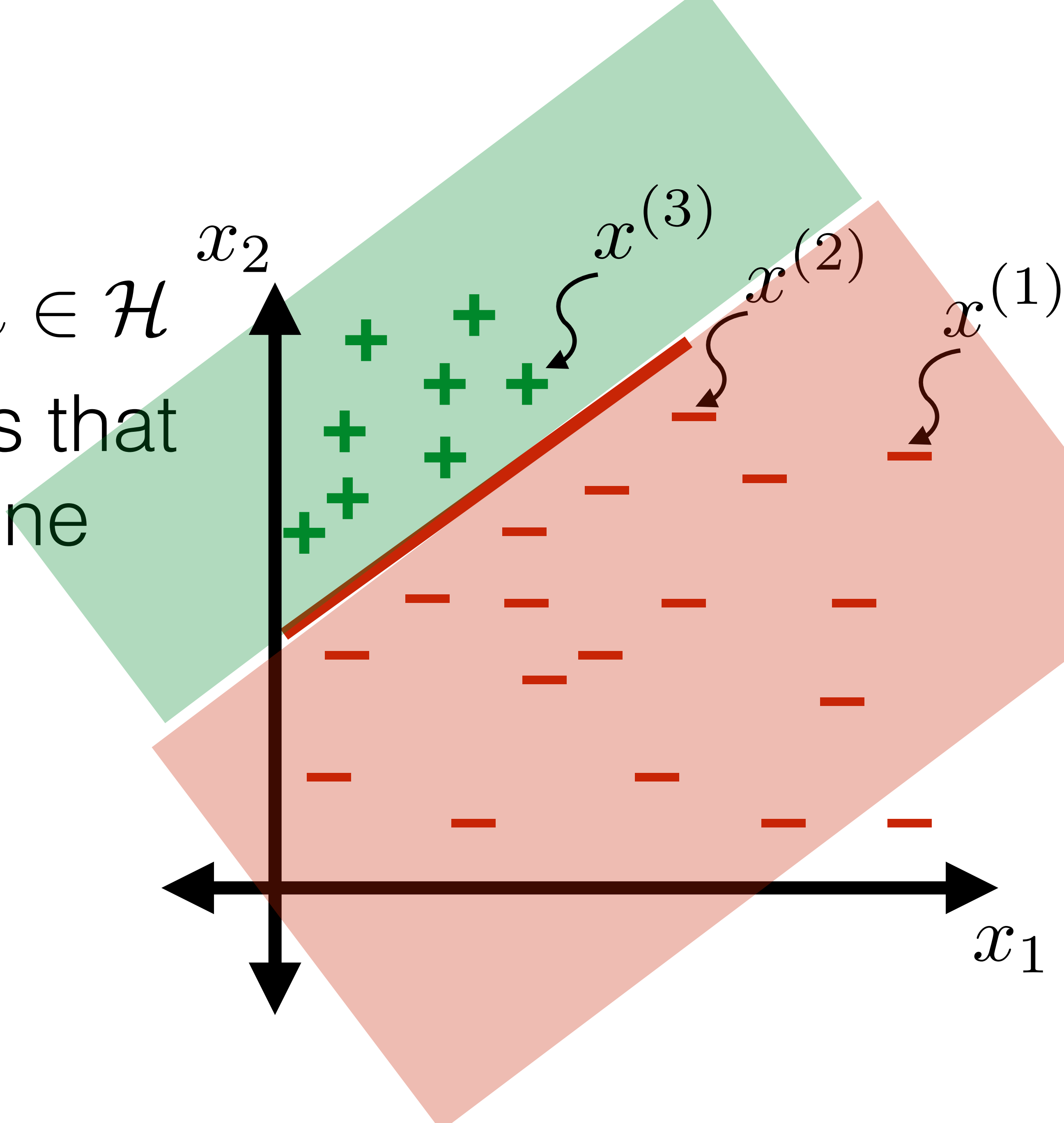


$\theta^\top x$

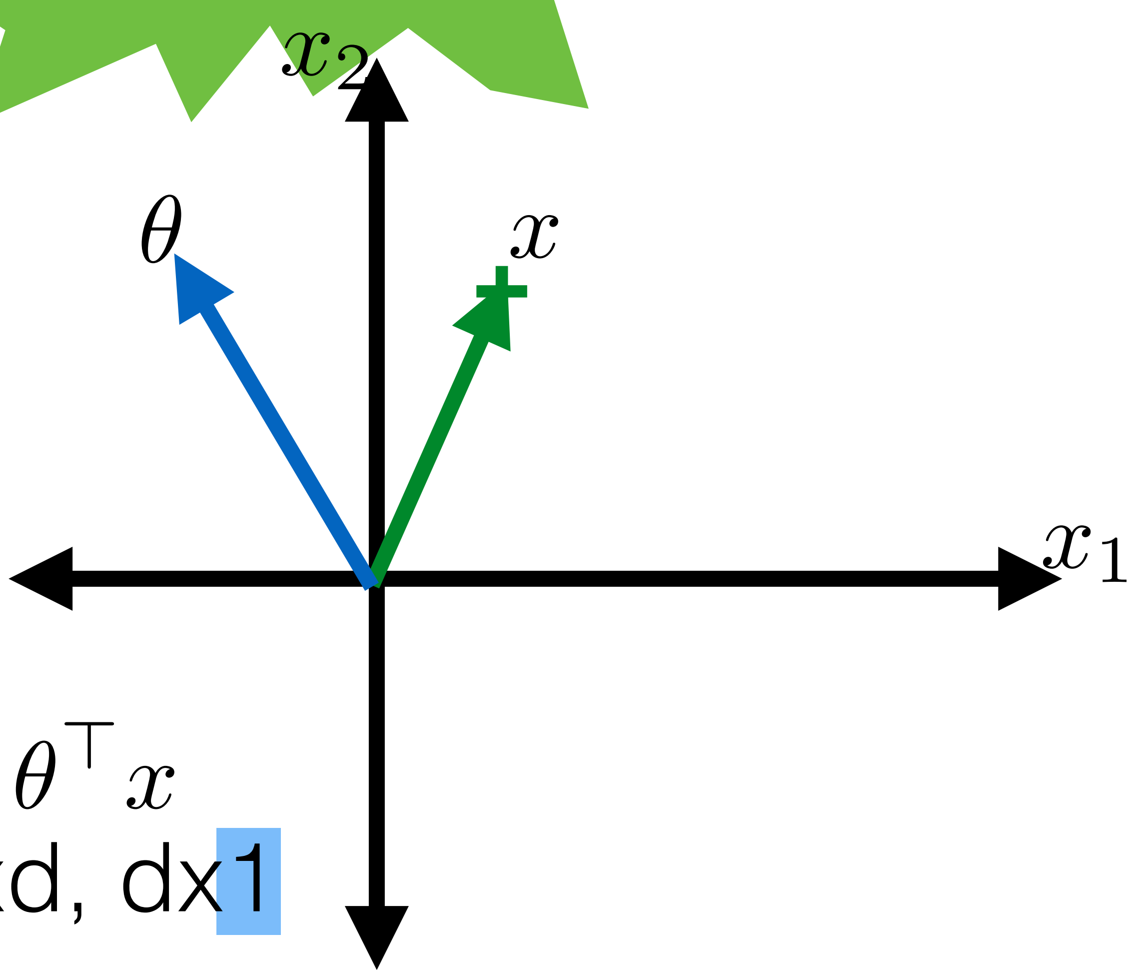
1xd, dx1

Linear classifiers

- Hypothesis class $\mathcal{H}$: set of $h \in \mathcal{H}$
  - Example $\mathcal{H}$: All hypotheses that label +1 on one side of a line and -1 on the other side



Math facts!

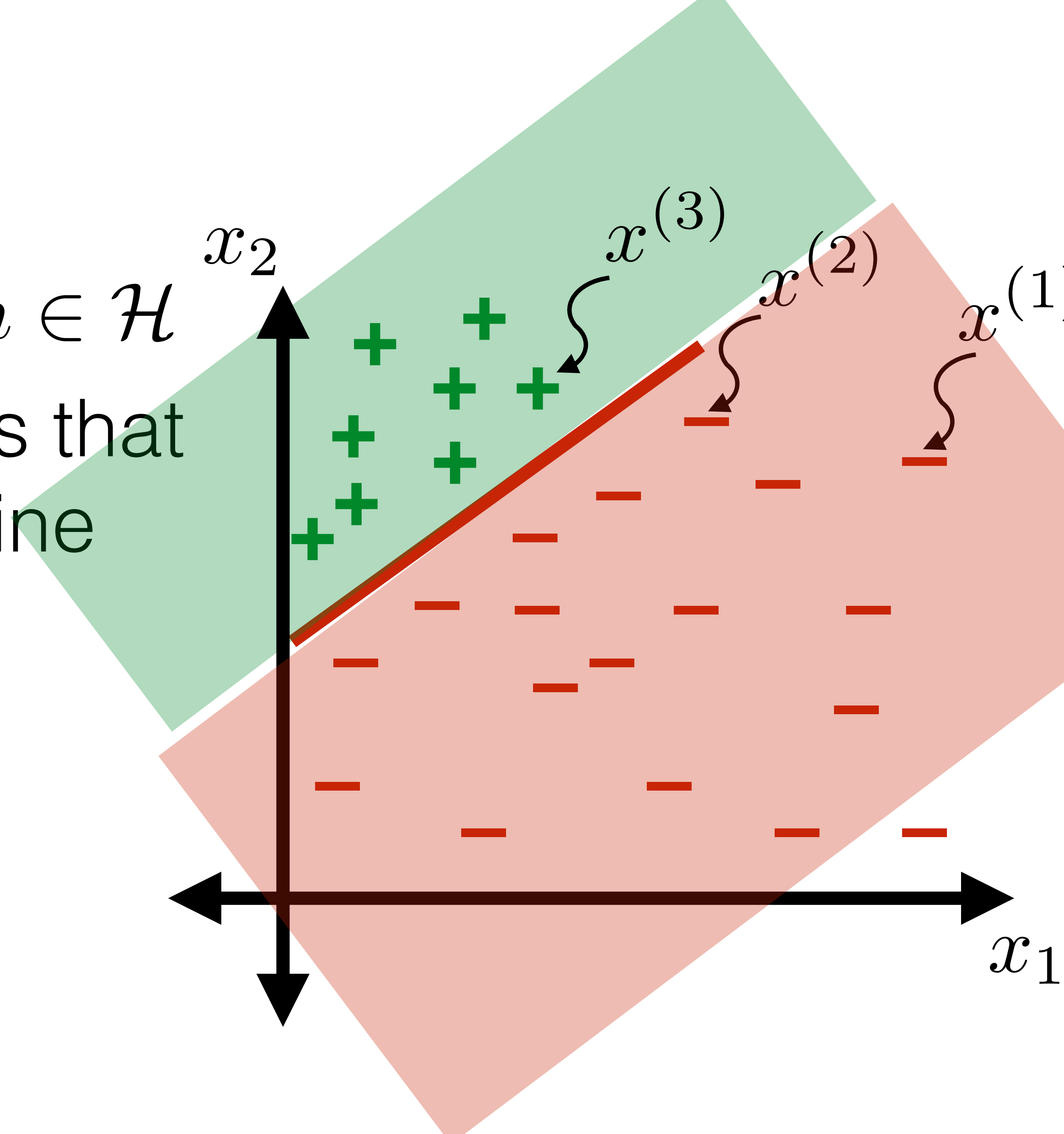


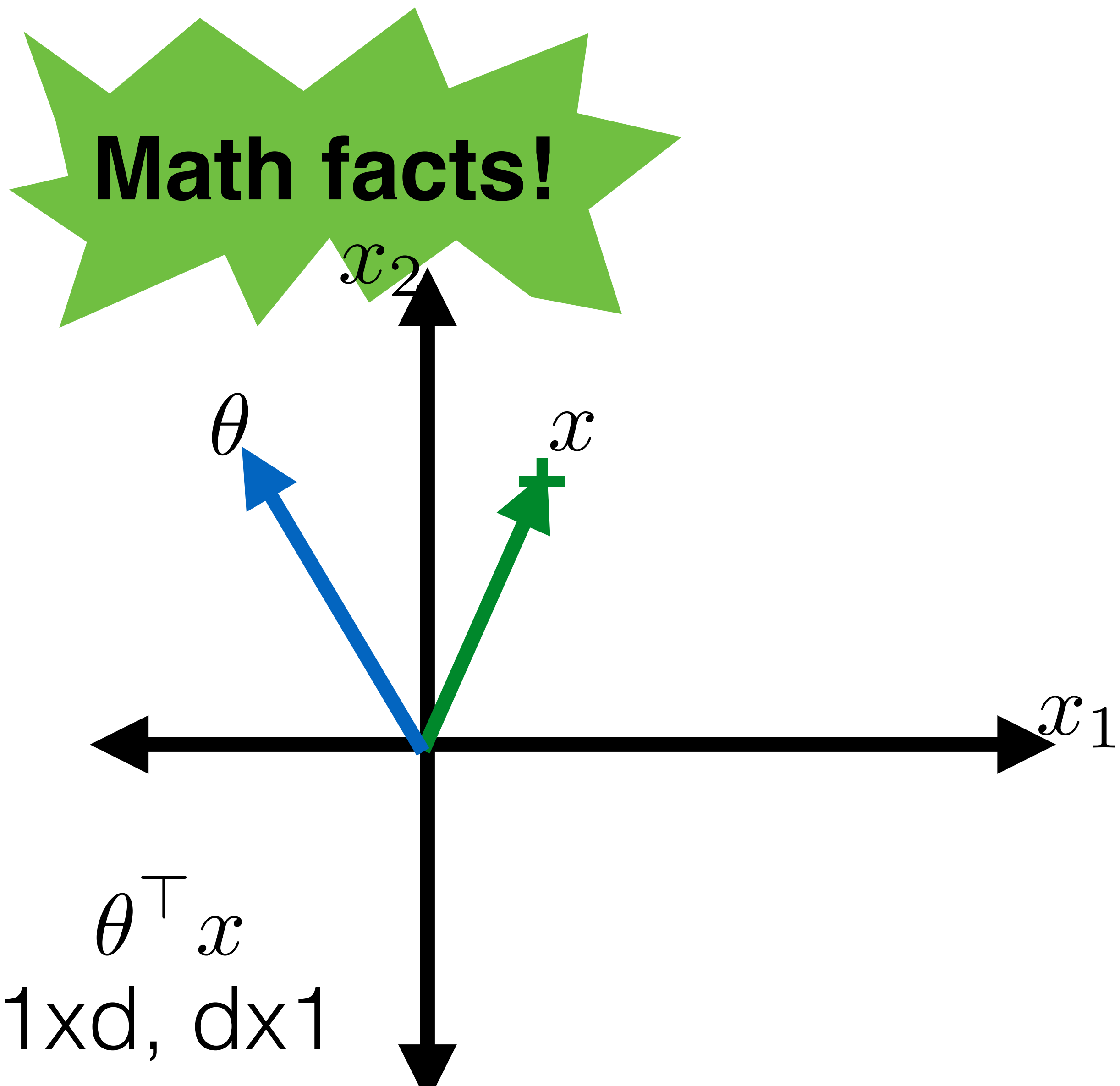
$\theta^\top x$

1xd, dx1

Linear classifiers

- Hypothesis class $\mathcal{H}$: set of $h \in \mathcal{H}$
    - Example $\mathcal{H}$: All hypotheses that label +1 on one side of a line and -1 on the other side



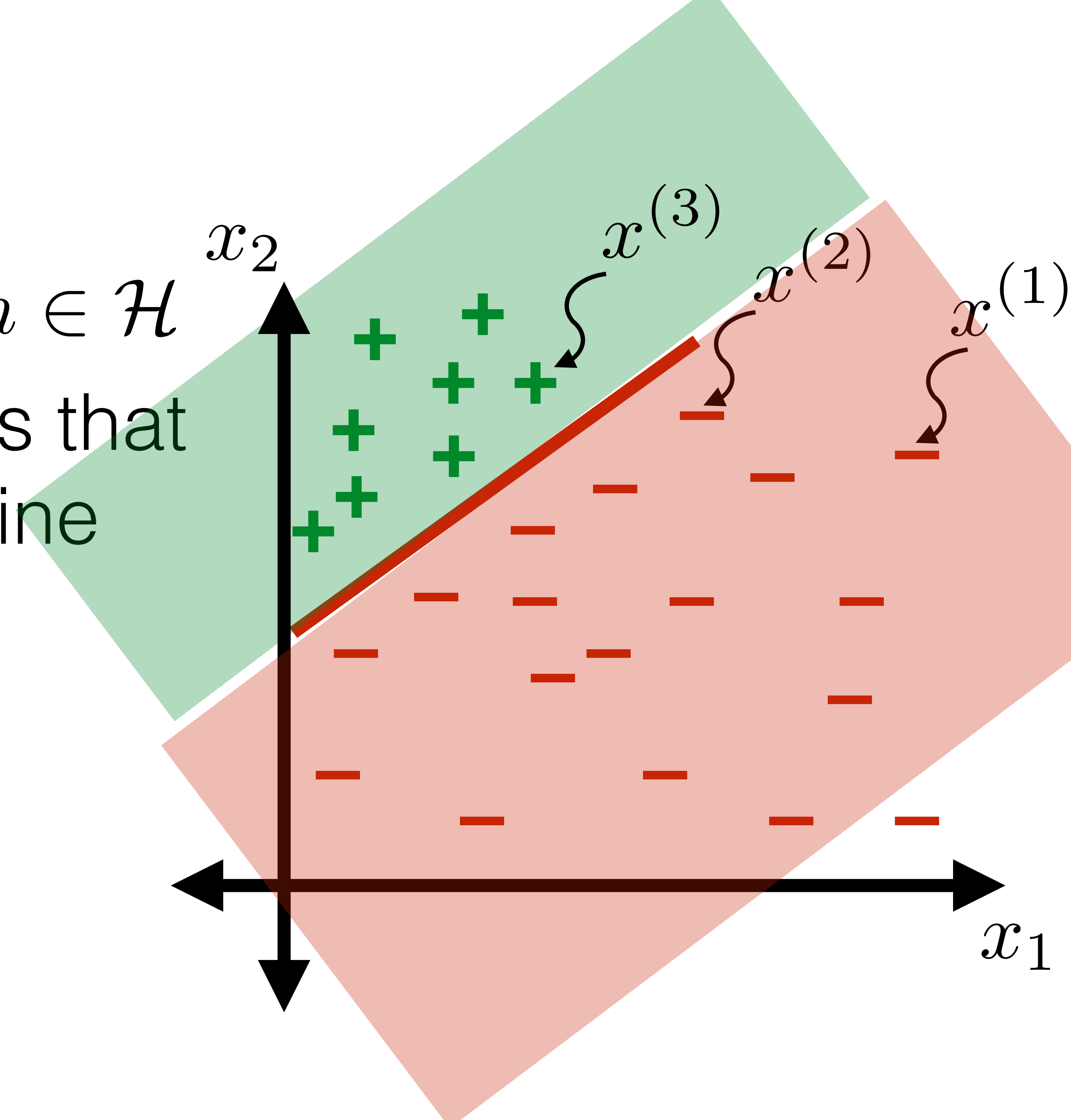


$\theta^\top x$

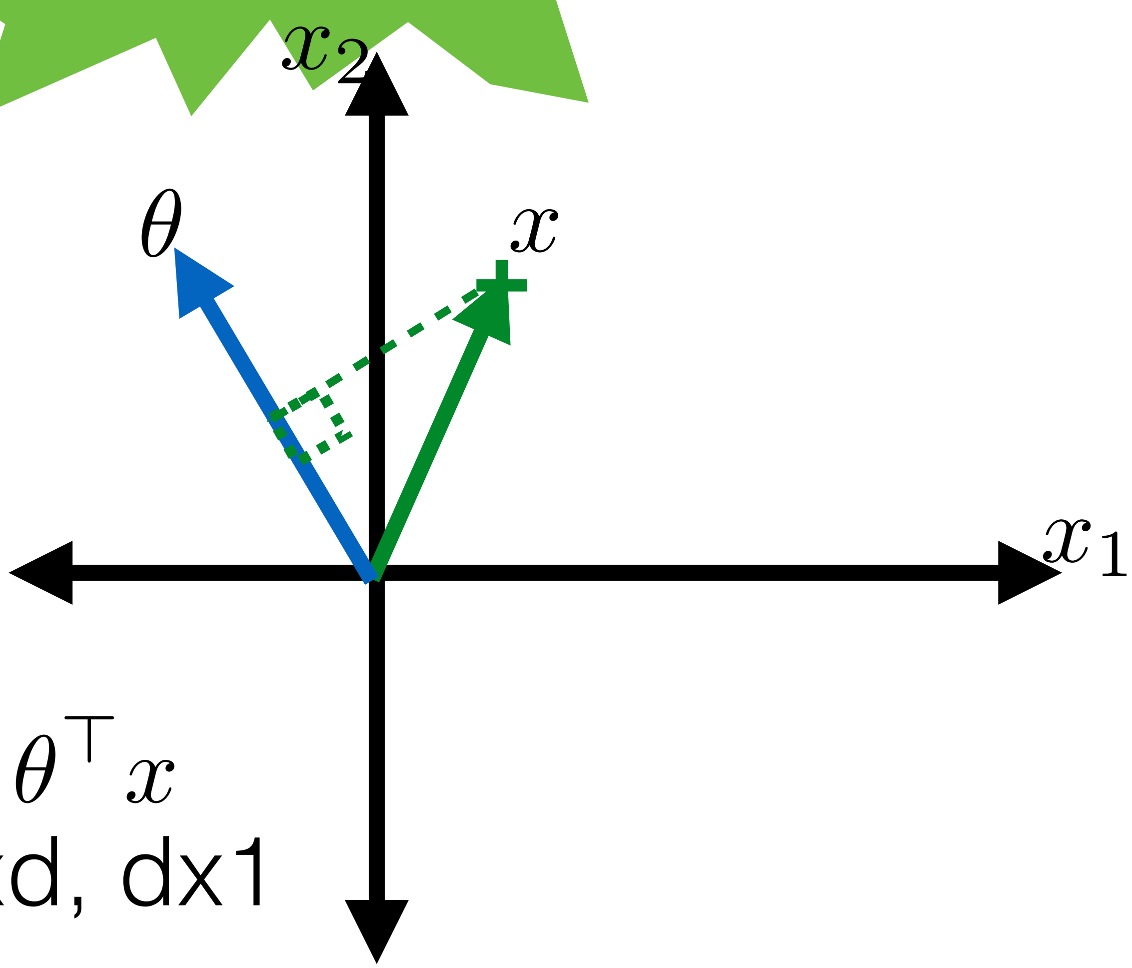
1xd, dx1

Linear classifiers

- Hypothesis class $\mathcal{H}$: set of $h \in \mathcal{H}$
  - Example $\mathcal{H}$: All hypotheses that label +1 on one side of a line and -1 on the other side



Math facts!

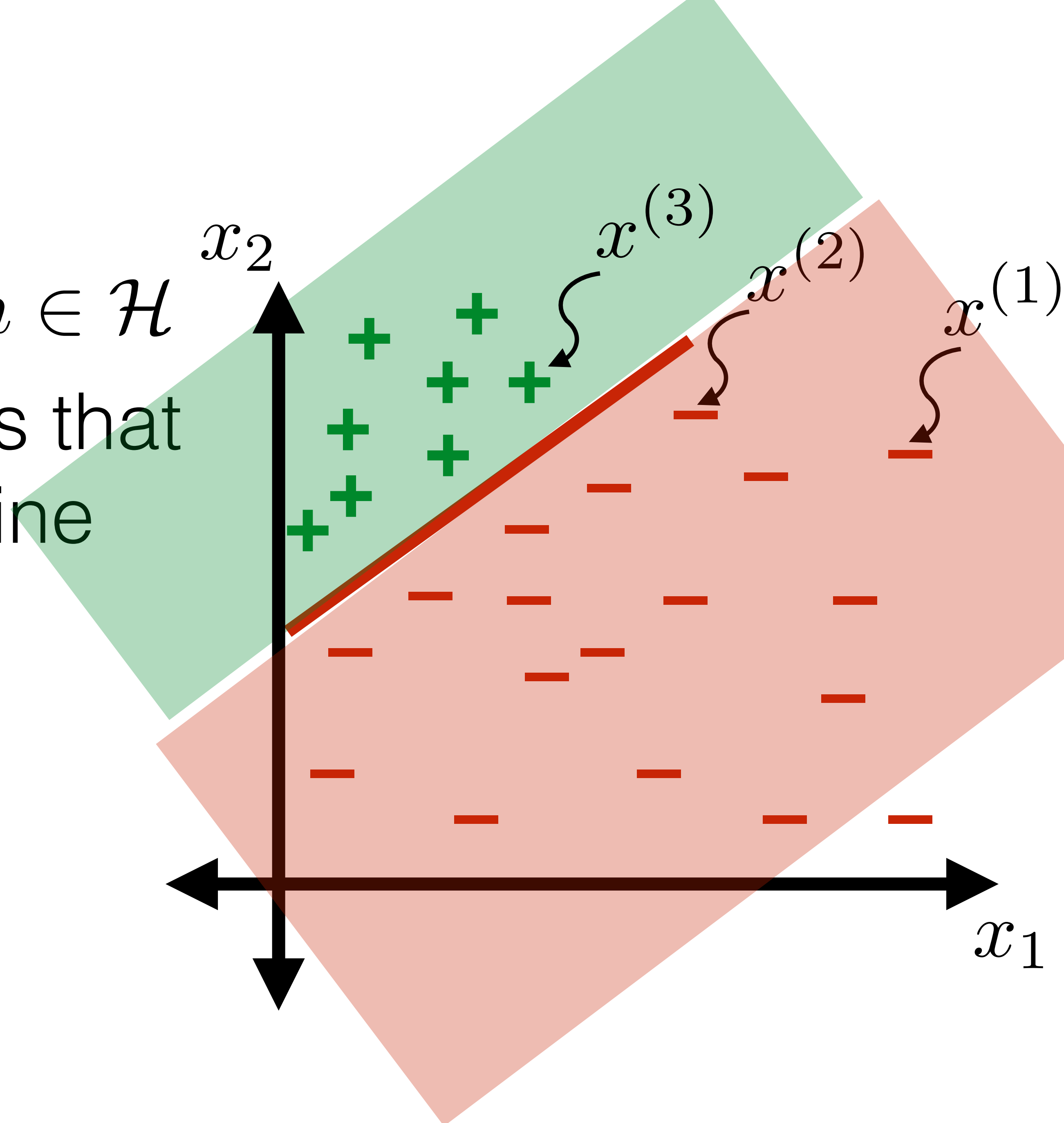


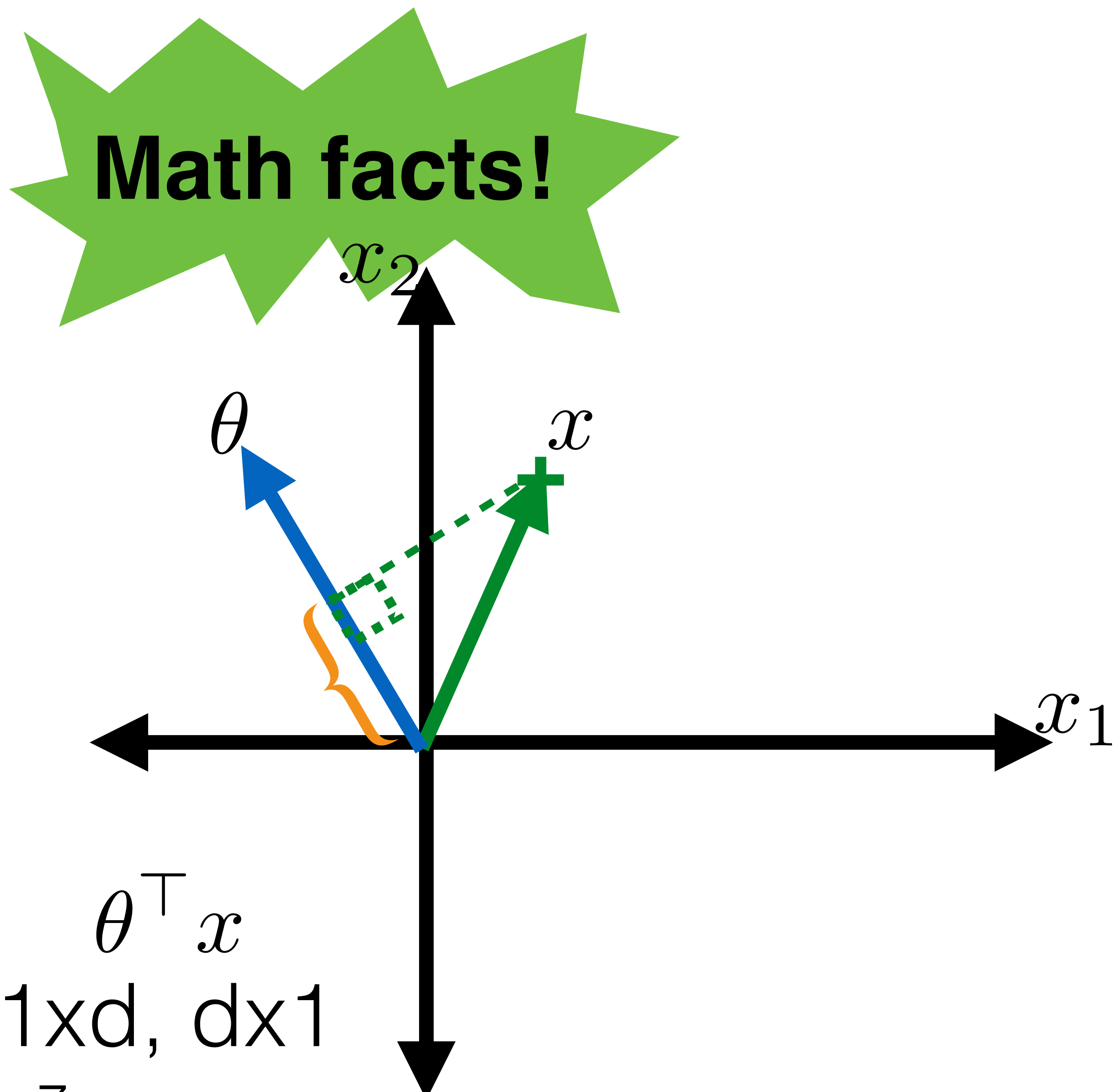
$\theta^\top x$

1xd, dx1

Linear classifiers

- Hypothesis class $\mathcal{H}$: set of $h \in \mathcal{H}$
  - Example $\mathcal{H}$: All hypotheses that label +1 on one side of a line and -1 on the other side

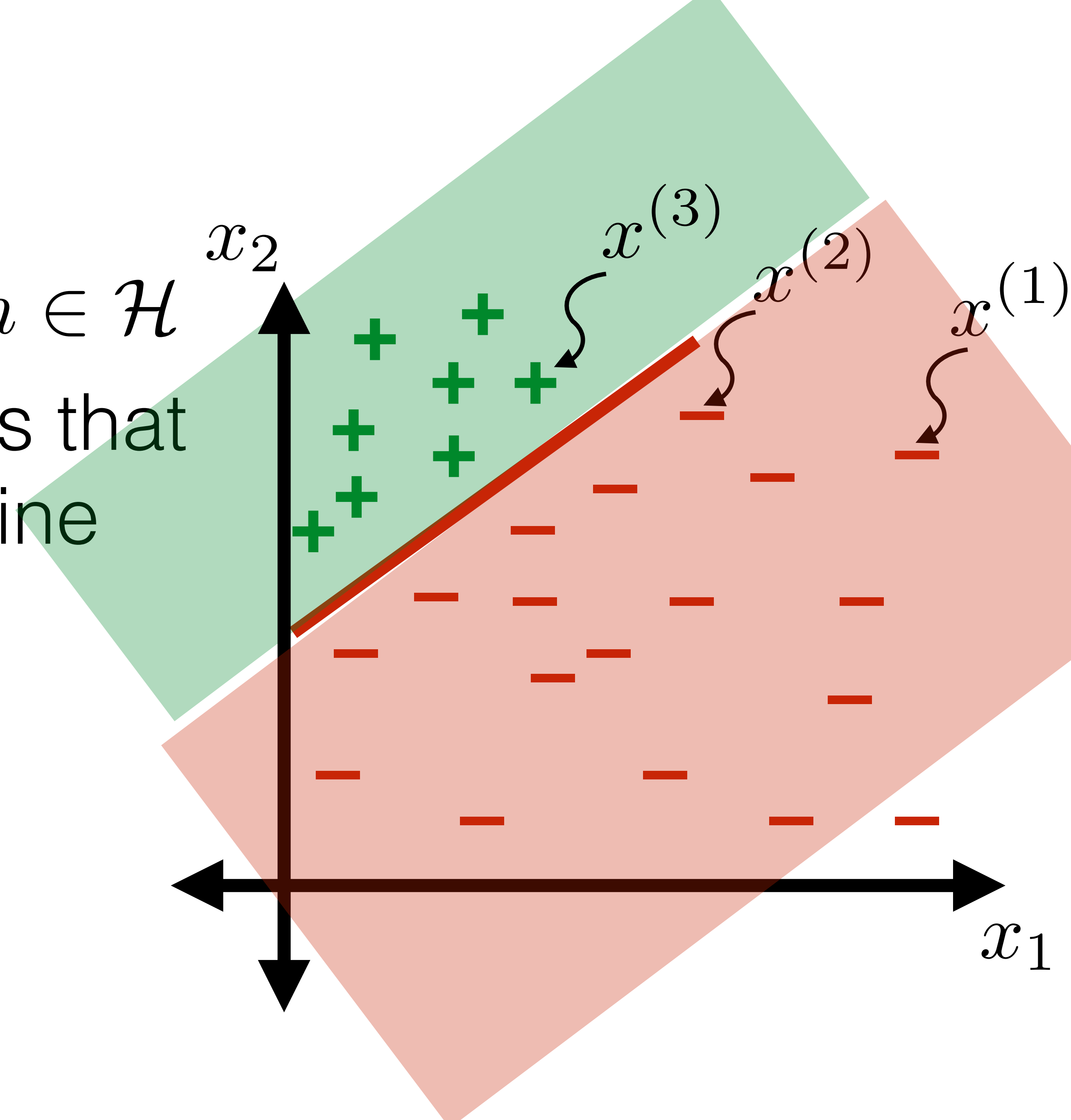


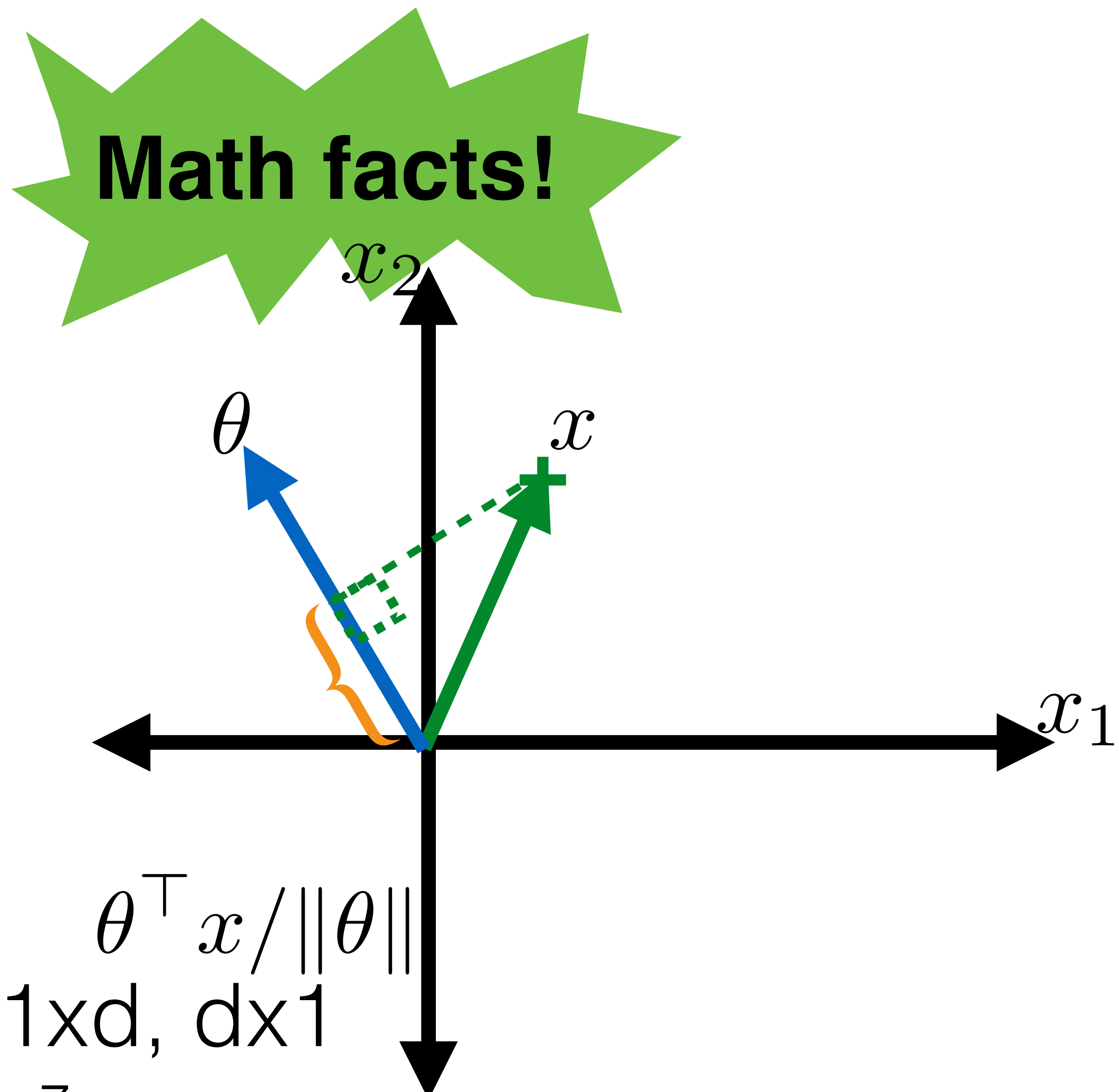


$\theta^\top x$
1xd, dx1

Linear classifiers

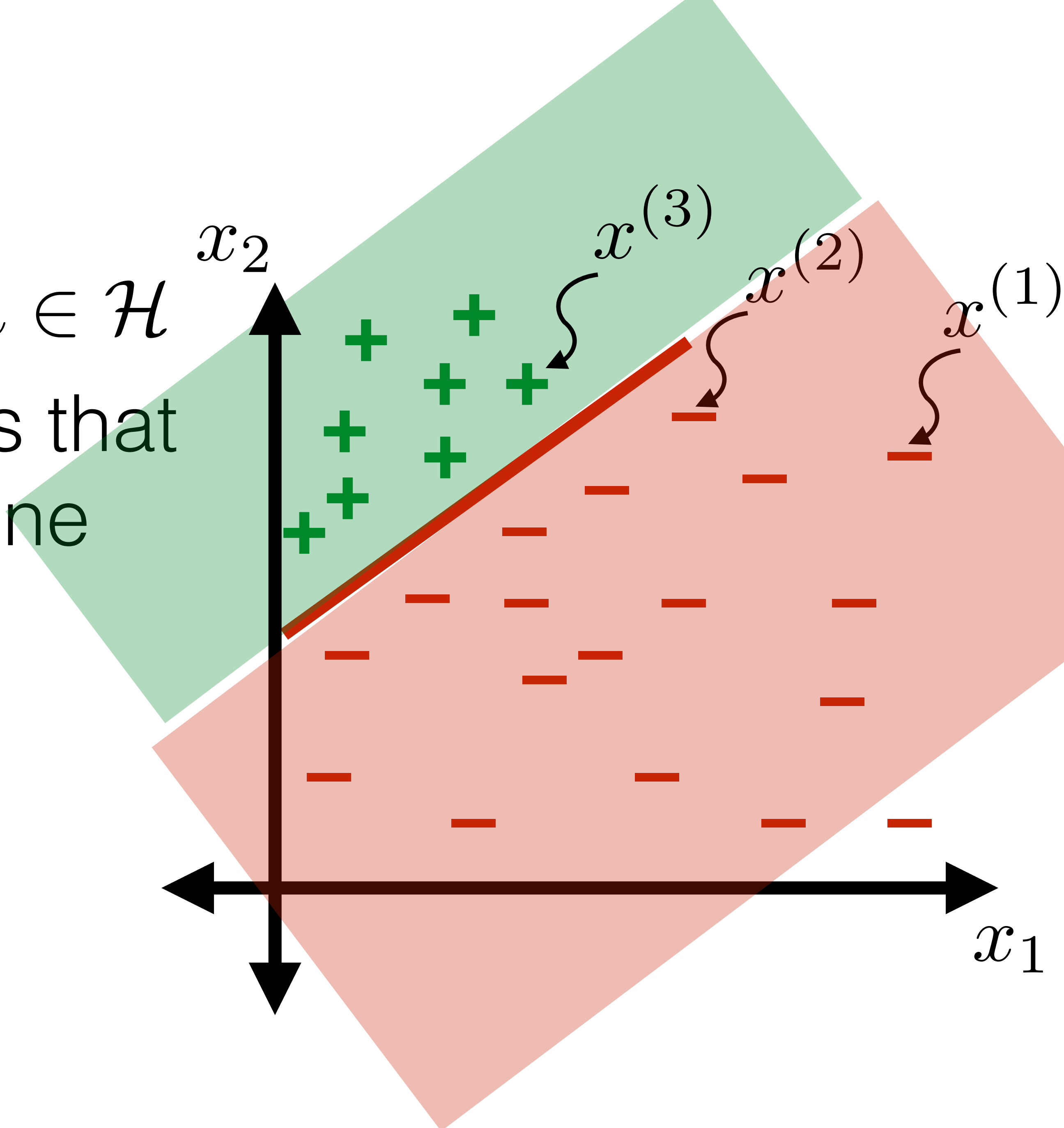
- Hypothesis class $\mathcal{H}$: set of $h \in \mathcal{H}$
  - Example $\mathcal{H}$: All hypotheses that label +1 on one side of a line and -1 on the other side

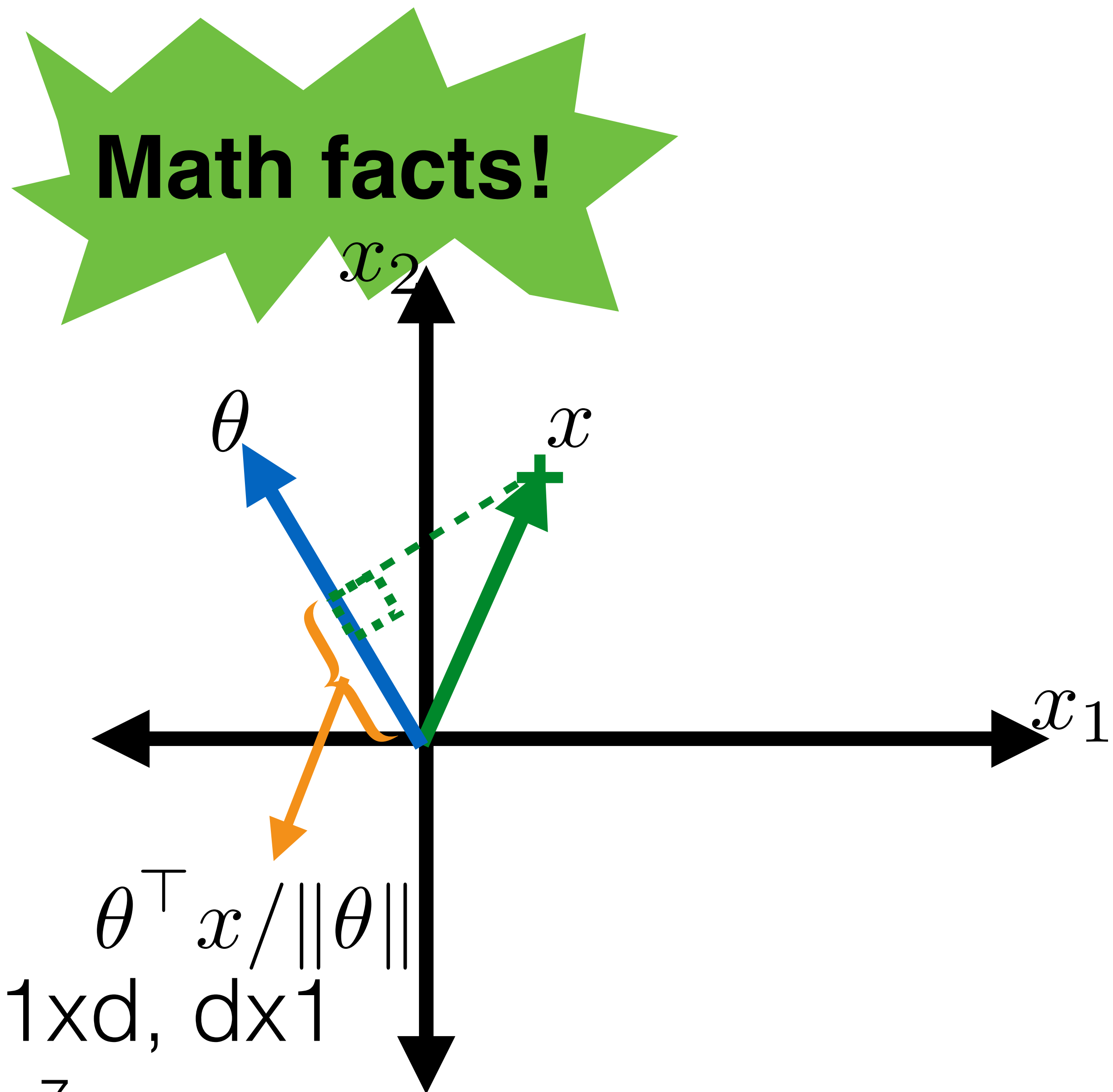




Linear classifiers

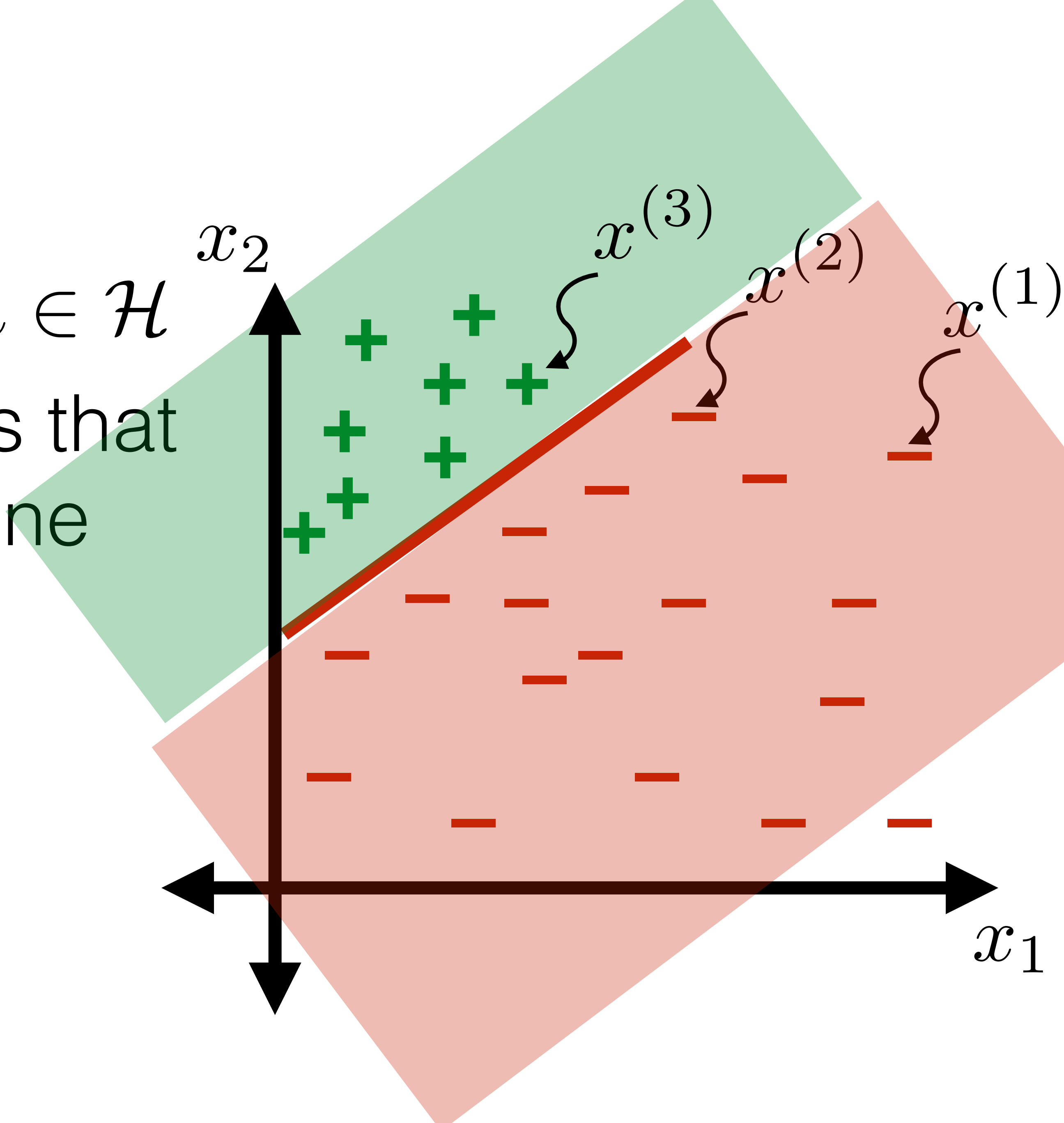
- Hypothesis class $\mathcal{H}$: set of $h \in \mathcal{H}$
  - Example $\mathcal{H}$: All hypotheses that label +1 on one side of a line and -1 on the other side

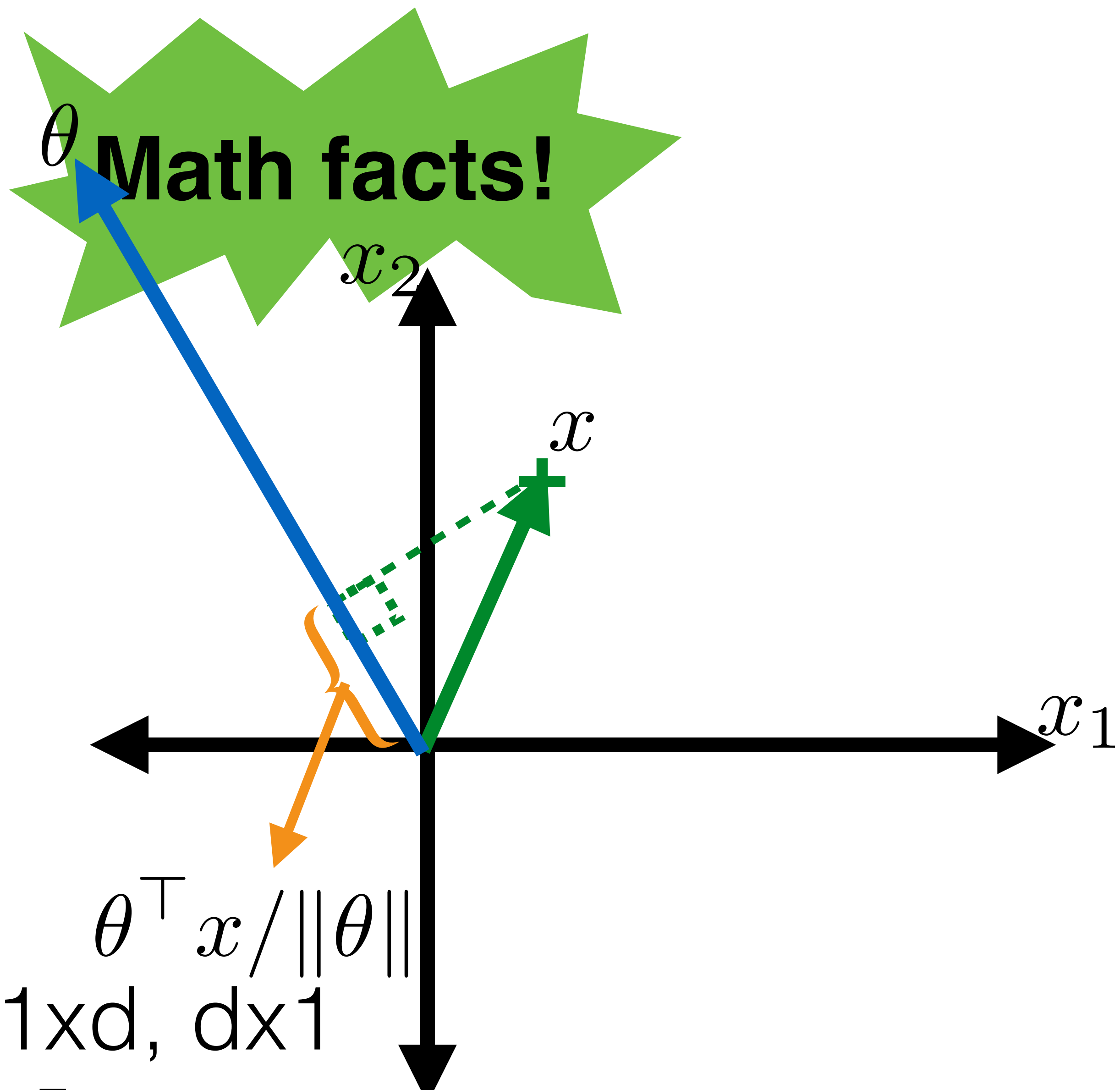




Linear classifiers

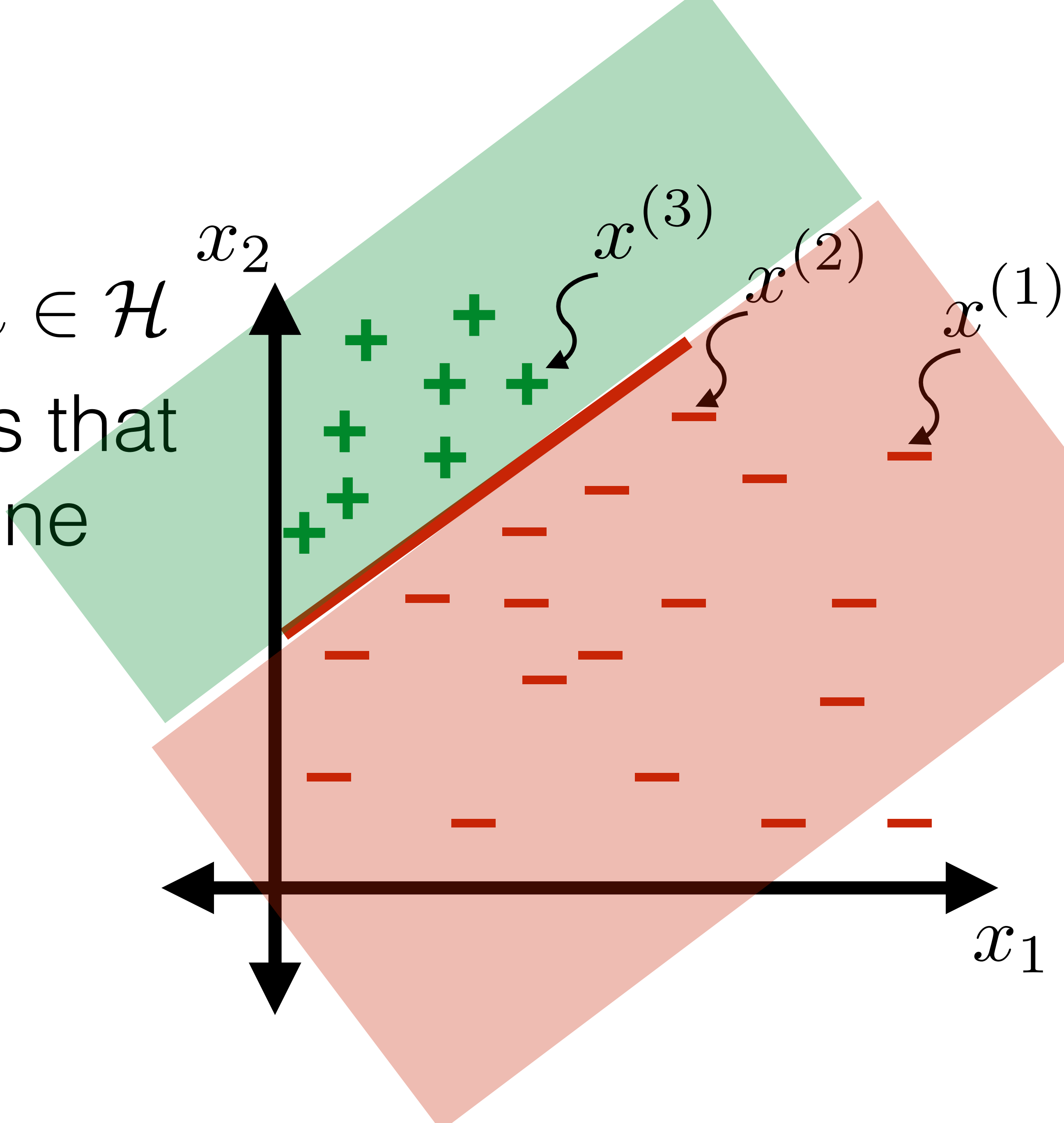
- Hypothesis class $\mathcal{H}$: set of $h \in \mathcal{H}$
  - Example $\mathcal{H}$: All hypotheses that label +1 on one side of a line and -1 on the other side



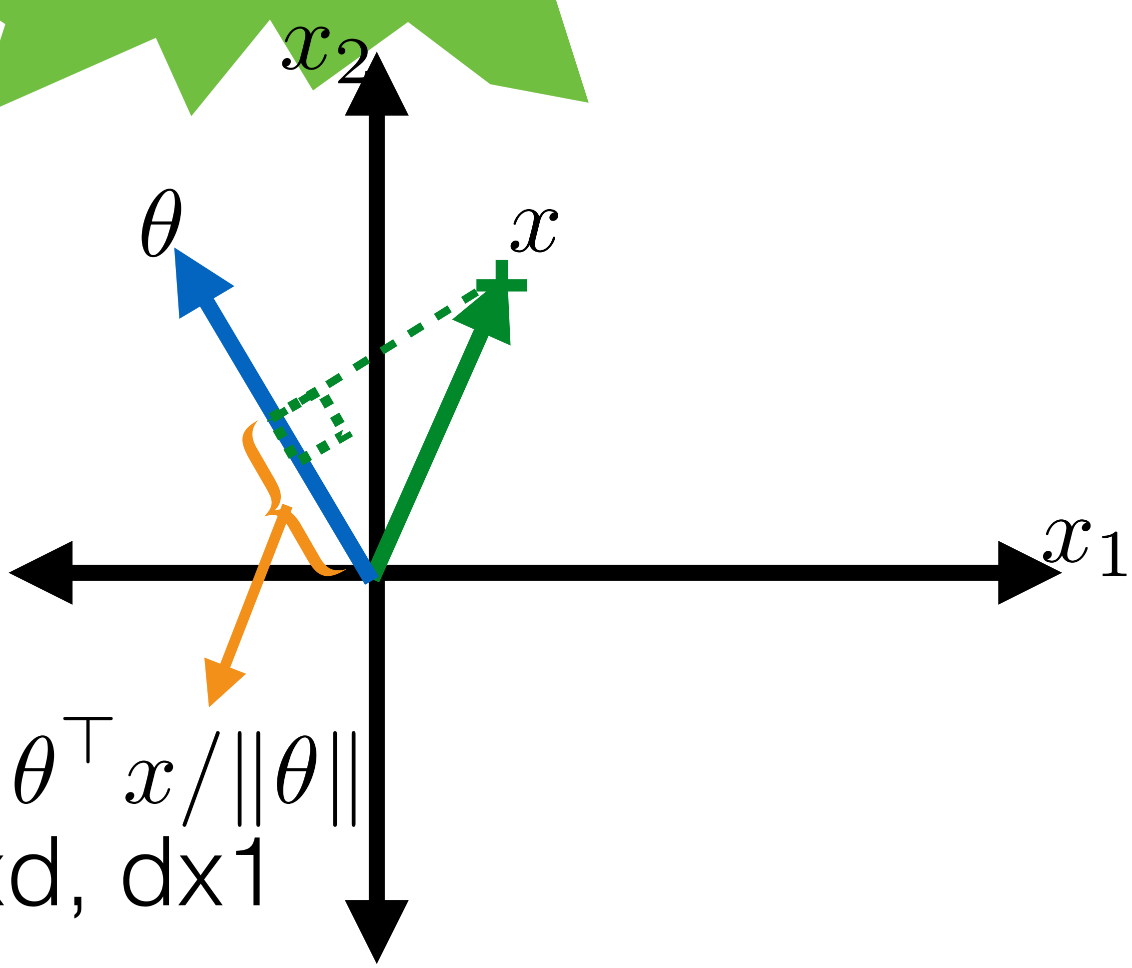


Linear classifiers

- Hypothesis class $\mathcal{H}$: set of $h \in \mathcal{H}$
  - Example $\mathcal{H}$: All hypotheses that label +1 on one side of a line and -1 on the other side



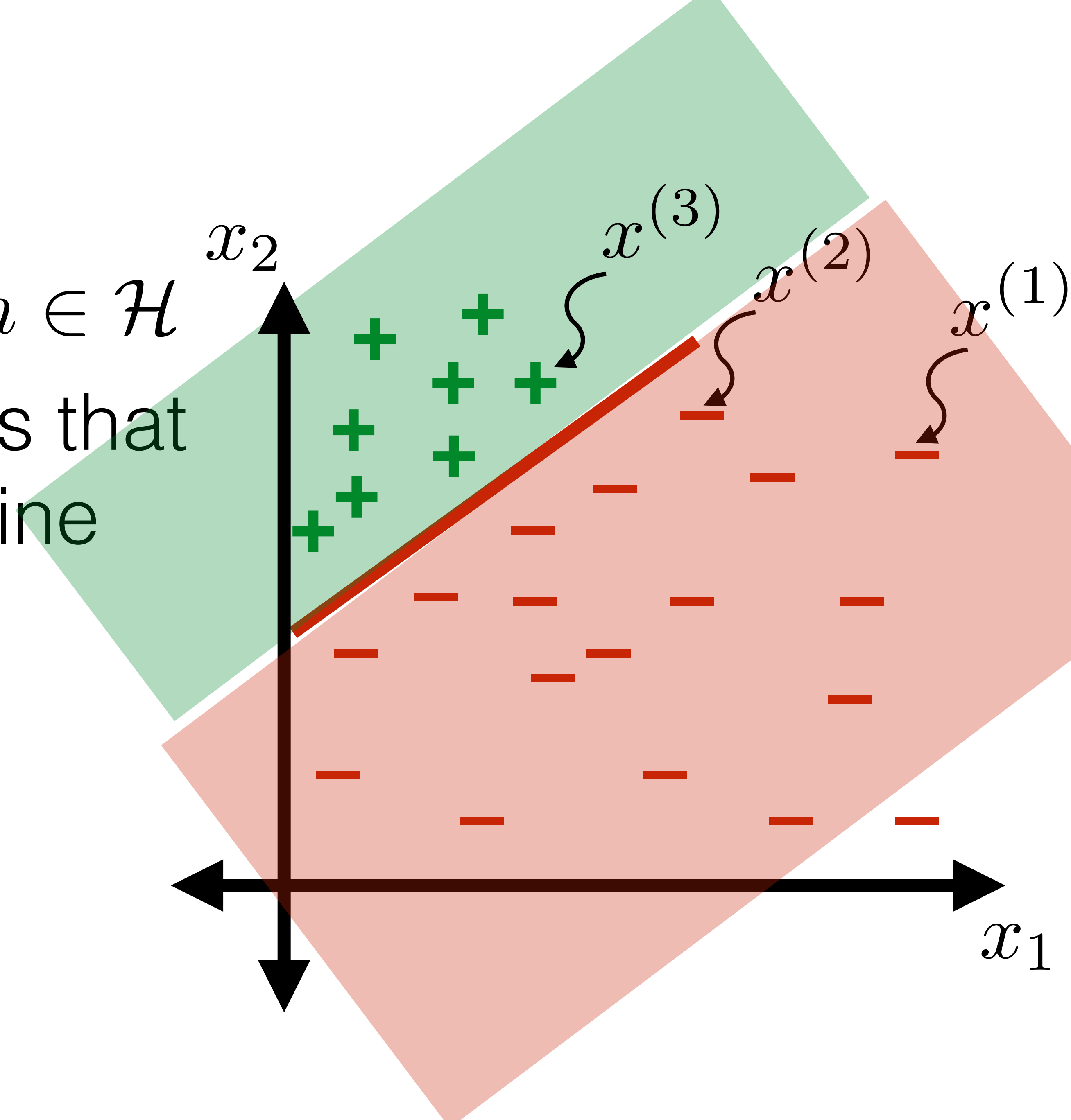
Math facts!

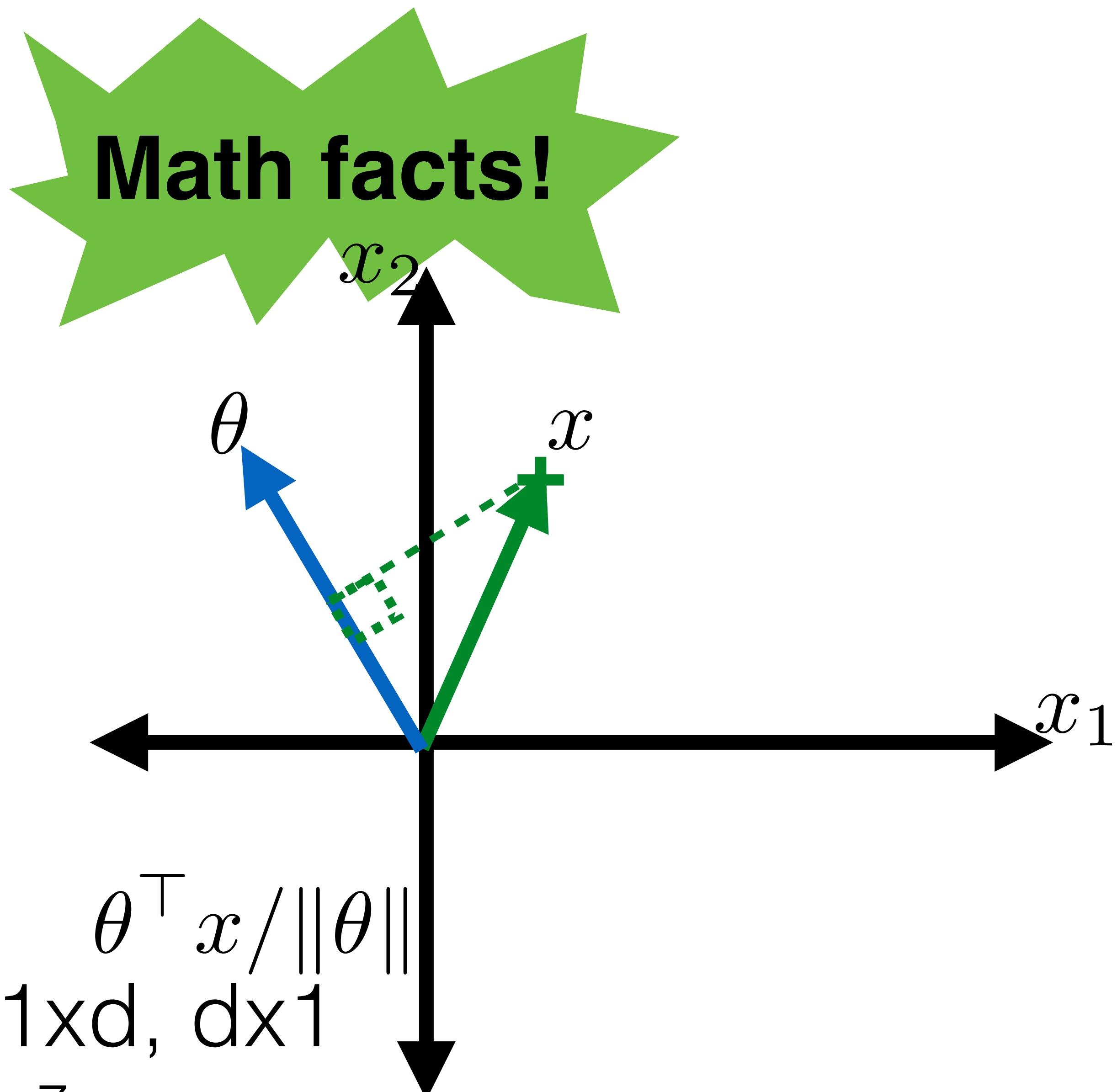


1xd, dx1

Linear classifiers

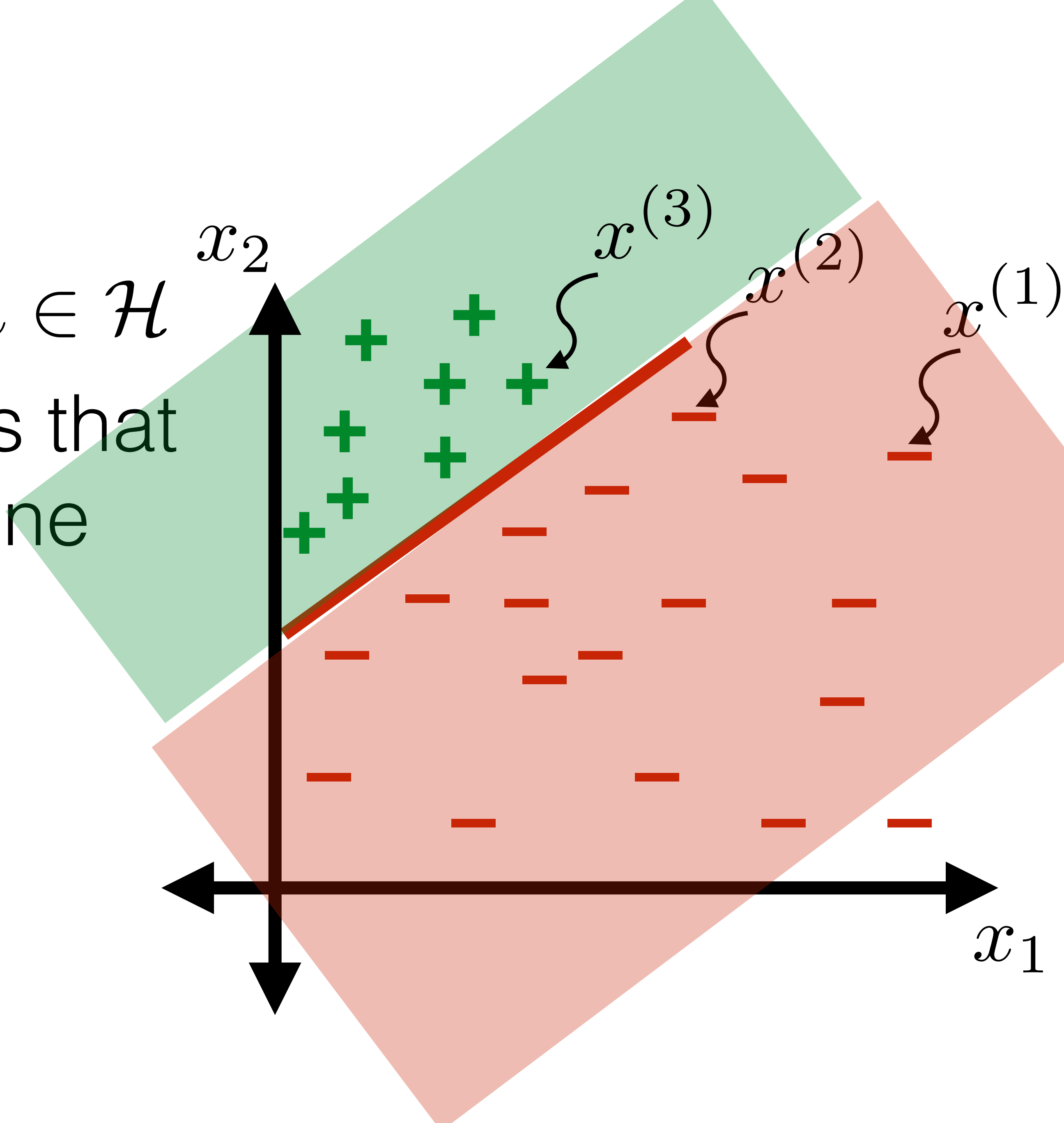
- Hypothesis class $\mathcal{H}$: set of $h \in \mathcal{H}$
  - Example $\mathcal{H}$: All hypotheses that label +1 on one side of a line and -1 on the other side



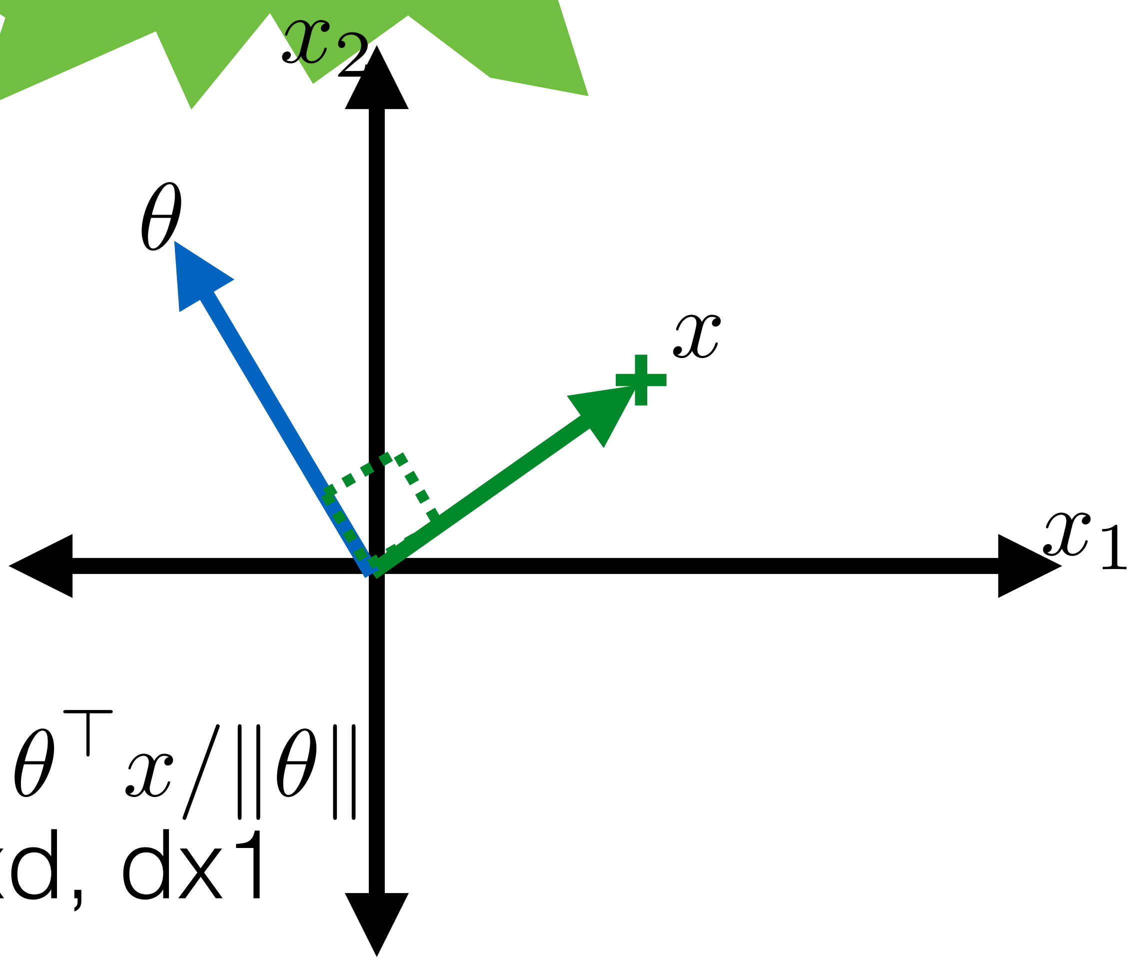


Linear classifiers

- Hypothesis class $\mathcal{H}$: set of $h \in \mathcal{H}$
  - Example $\mathcal{H}$: All hypotheses that label +1 on one side of a line and -1 on the other side



Math facts!

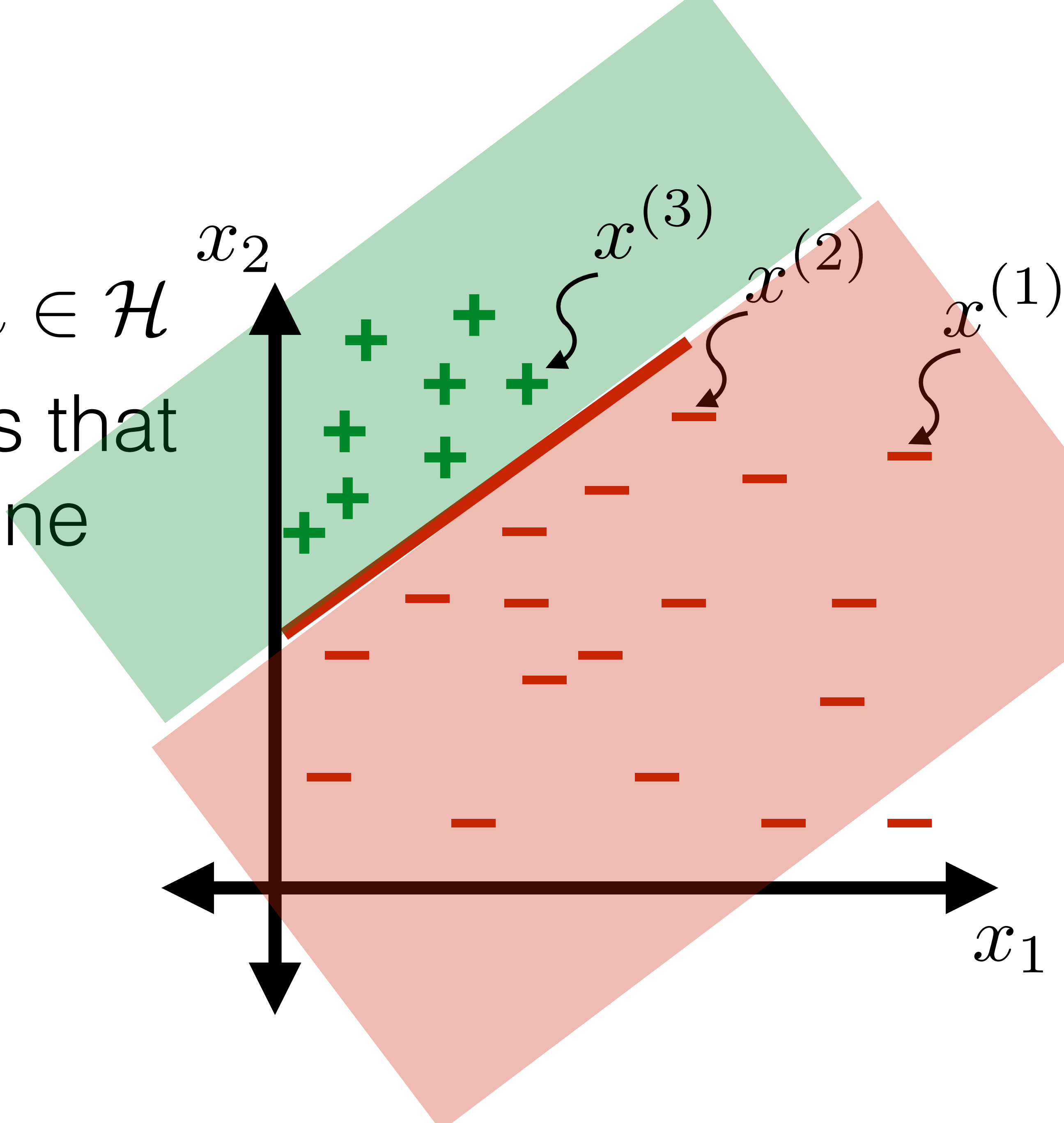


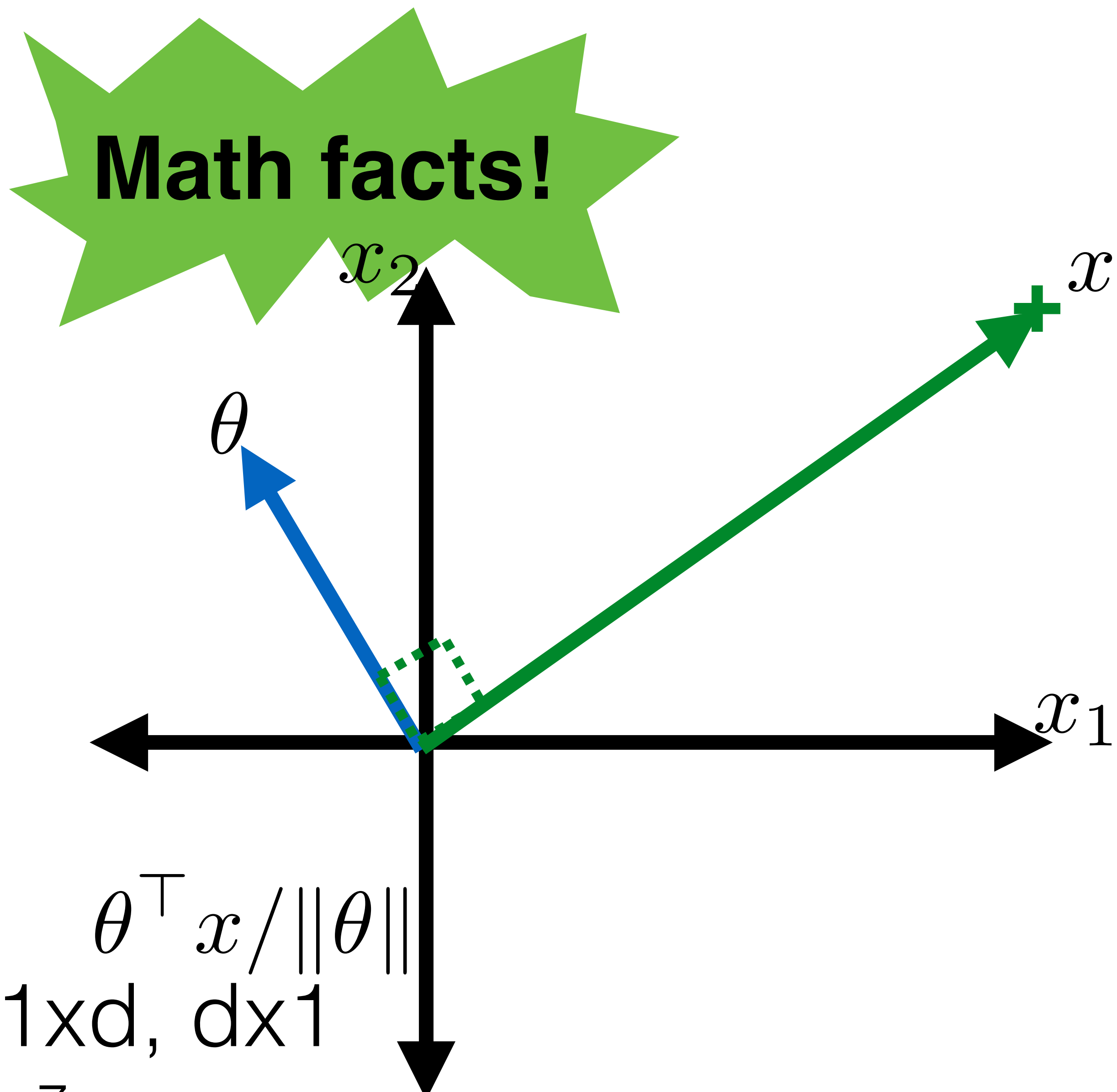
$\theta^\top x / \|\theta\|$

1xd, dx1

Linear classifiers

- Hypothesis class $\mathcal{H}$: set of $h \in \mathcal{H}$
  - Example $\mathcal{H}$: All hypotheses that label +1 on one side of a line and -1 on the other side



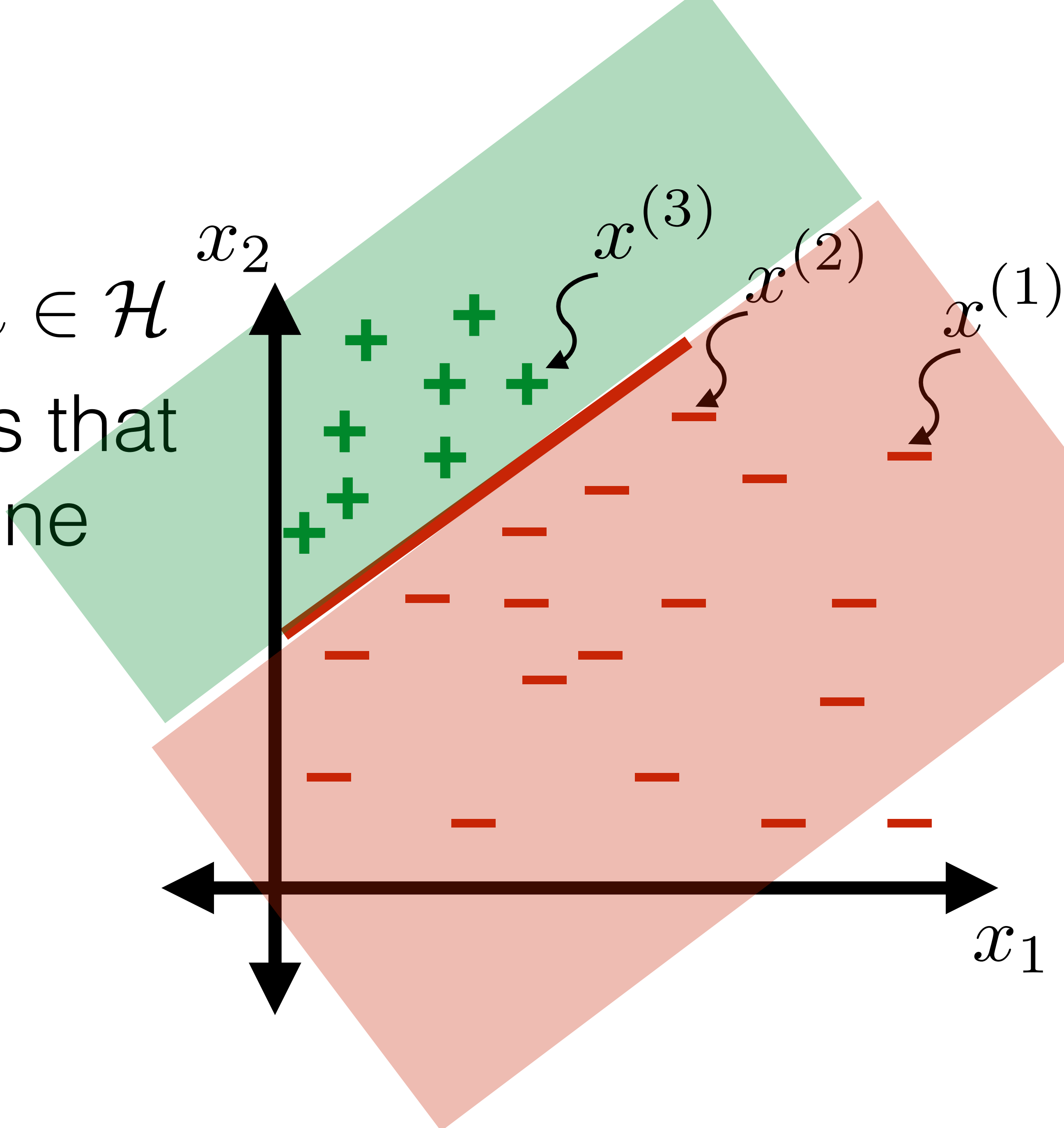


$\theta^\top x / \|\theta\|$

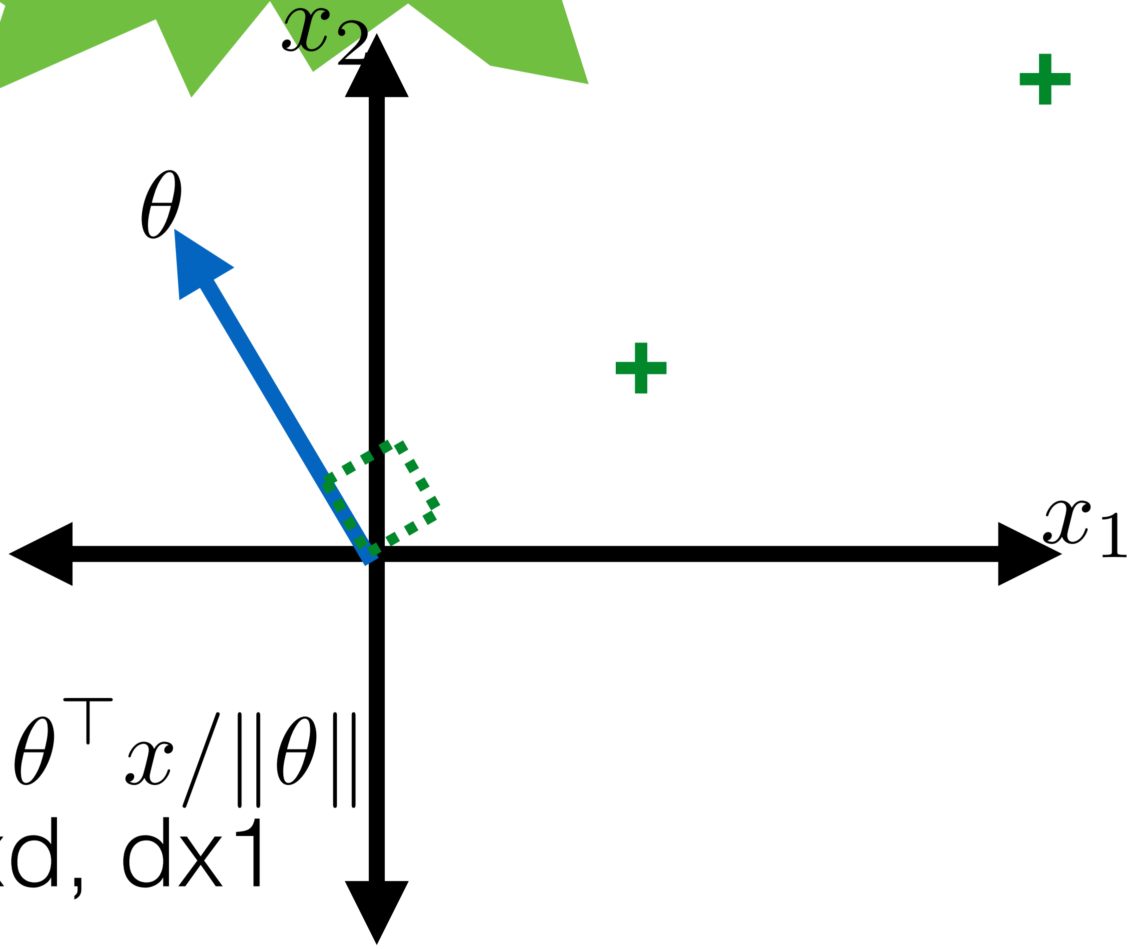
1xd, dx1

Linear classifiers

- Hypothesis class $\mathcal{H}$: set of $h \in \mathcal{H}$
  - Example $\mathcal{H}$: All hypotheses that label +1 on one side of a line and -1 on the other side



Math facts!

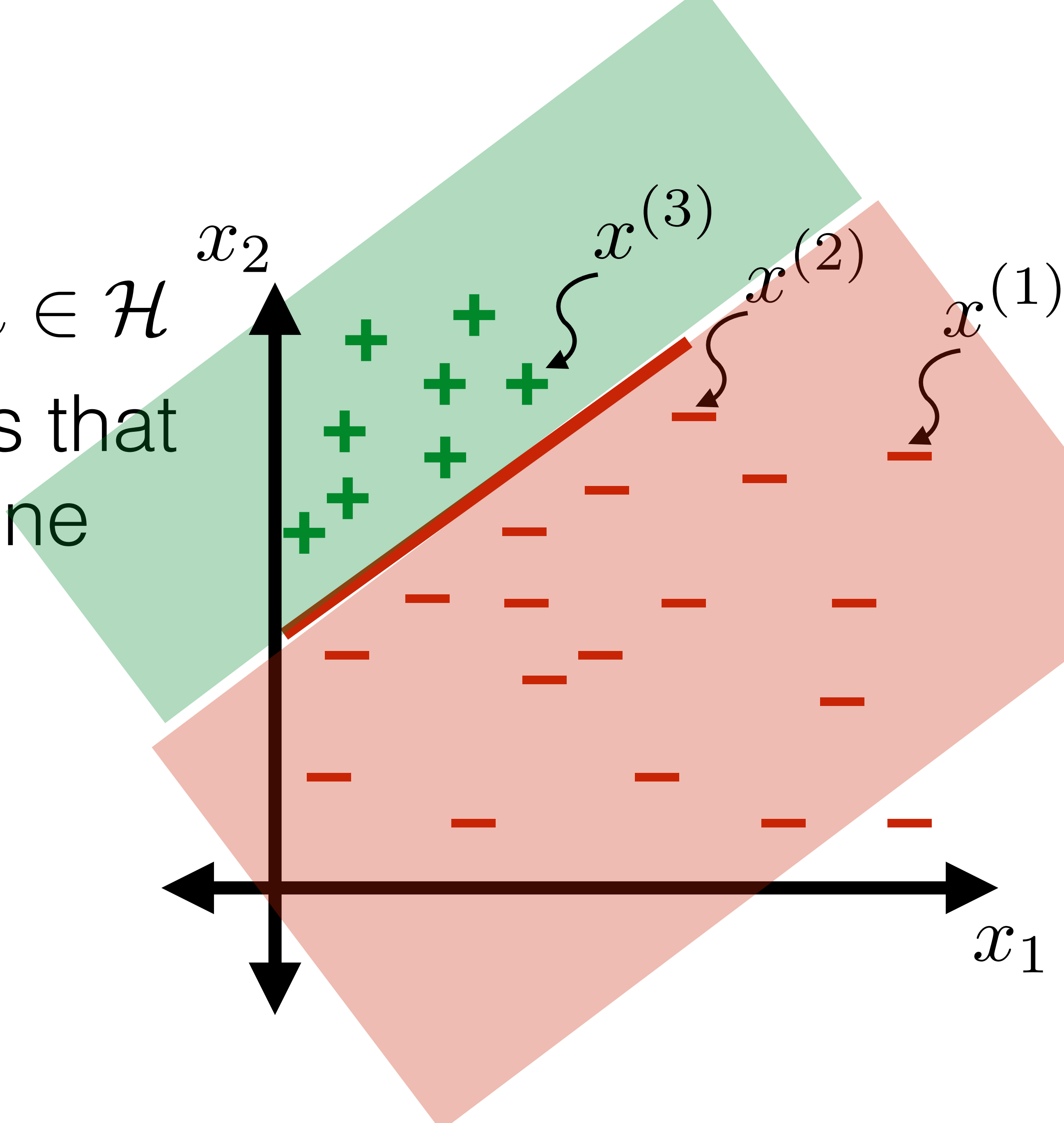


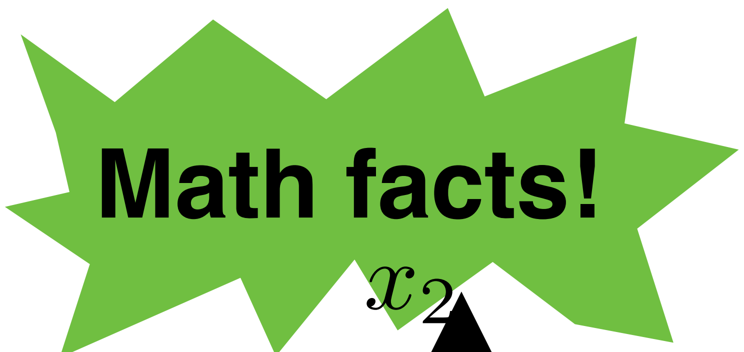
$\theta^\top x / \|\theta\|$

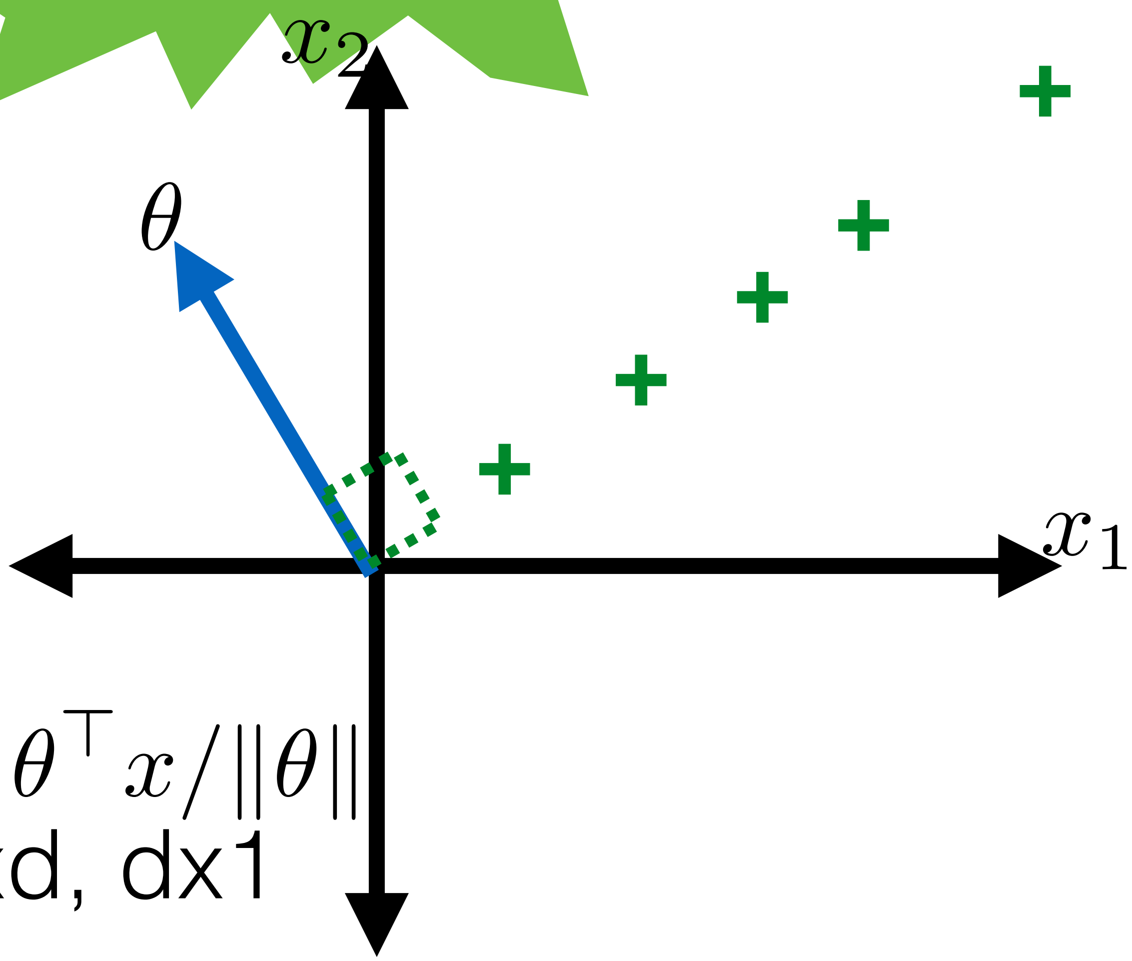
1xd, dx1

Linear classifiers

- Hypothesis class $\mathcal{H}$: set of $h \in \mathcal{H}$
  - Example $\mathcal{H}$: All hypotheses that label +1 on one side of a line and -1 on the other side





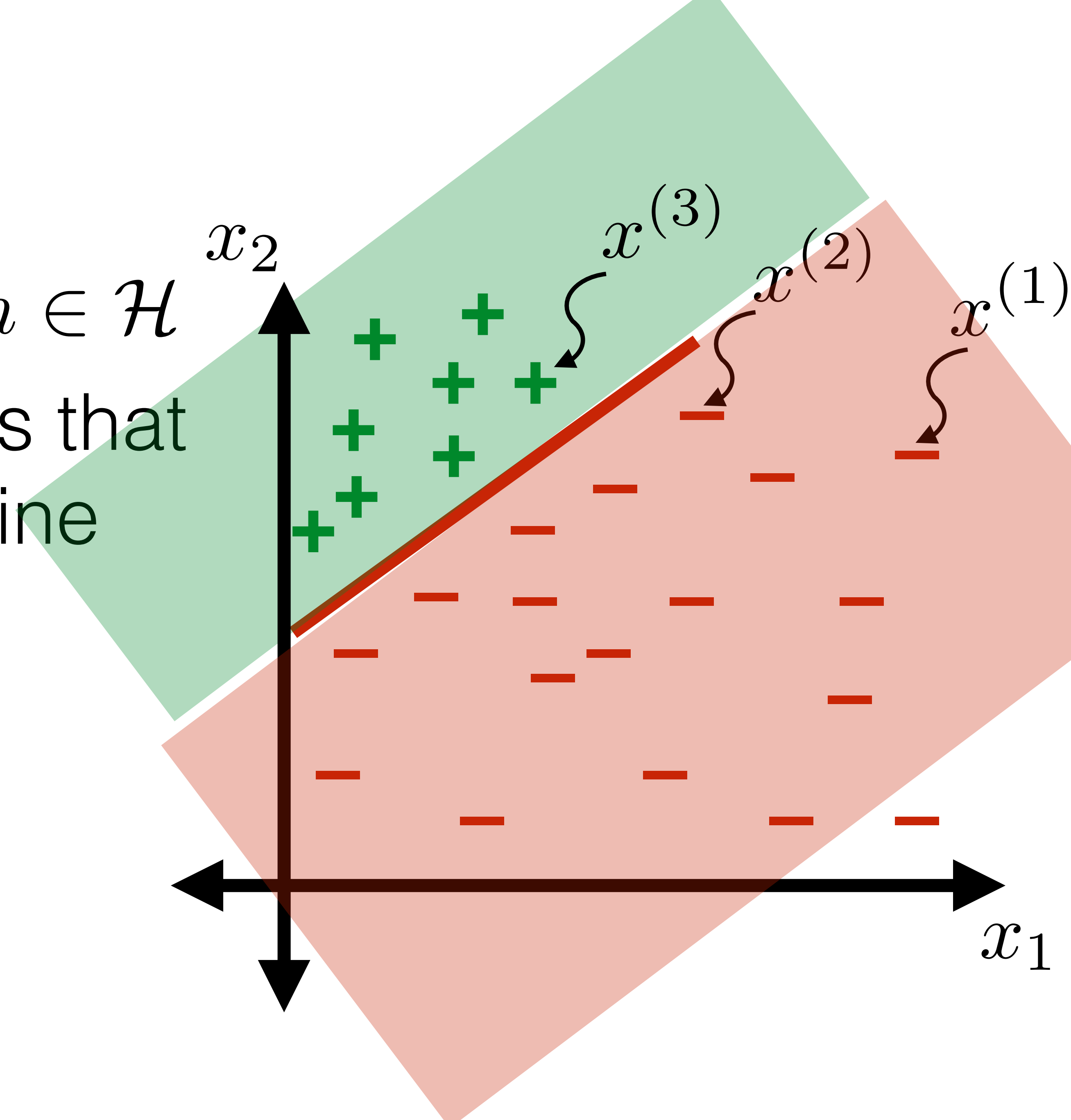


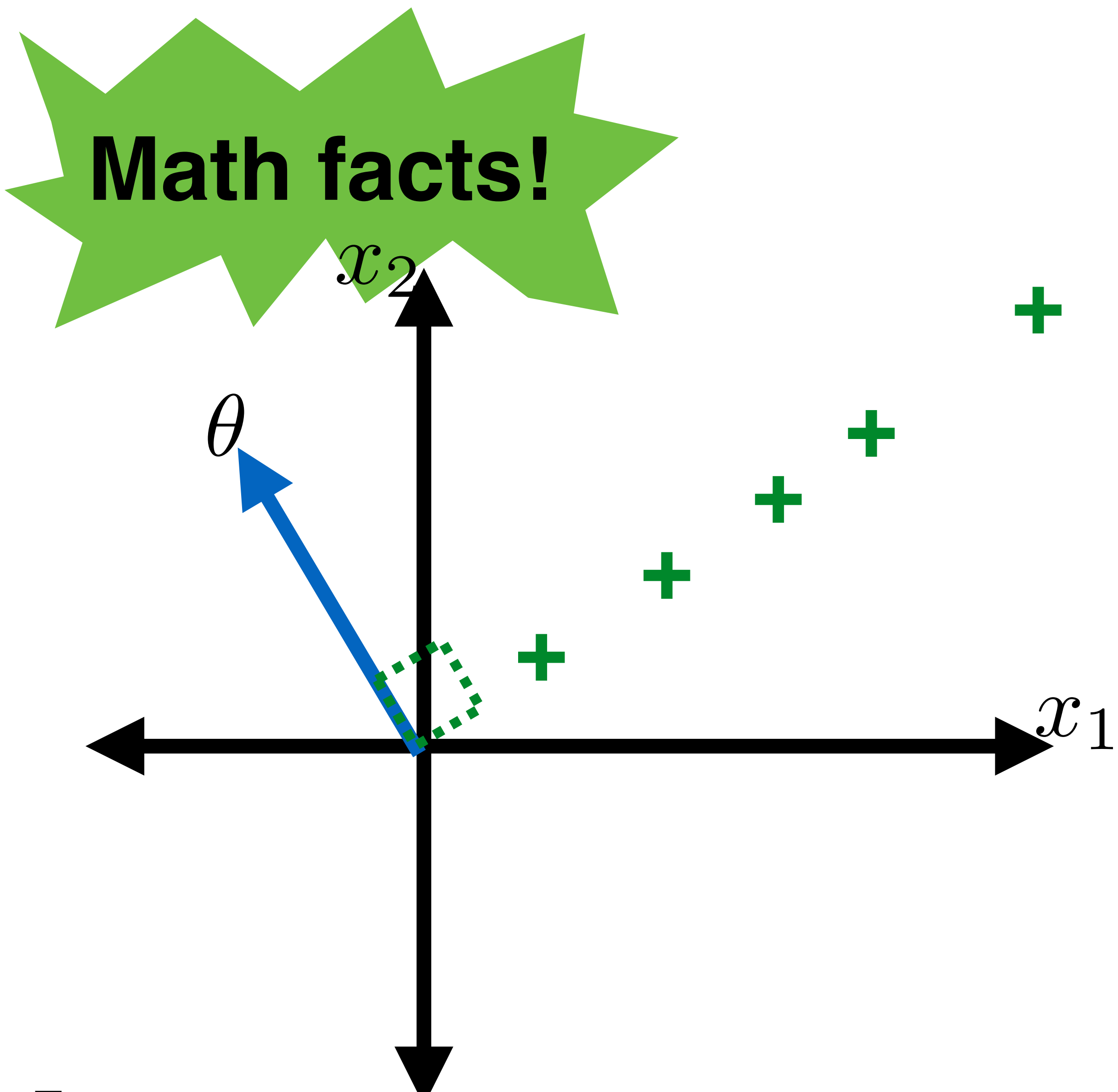
$\theta^\top x / \|\theta\|$

1xd, dx1

Linear classifiers

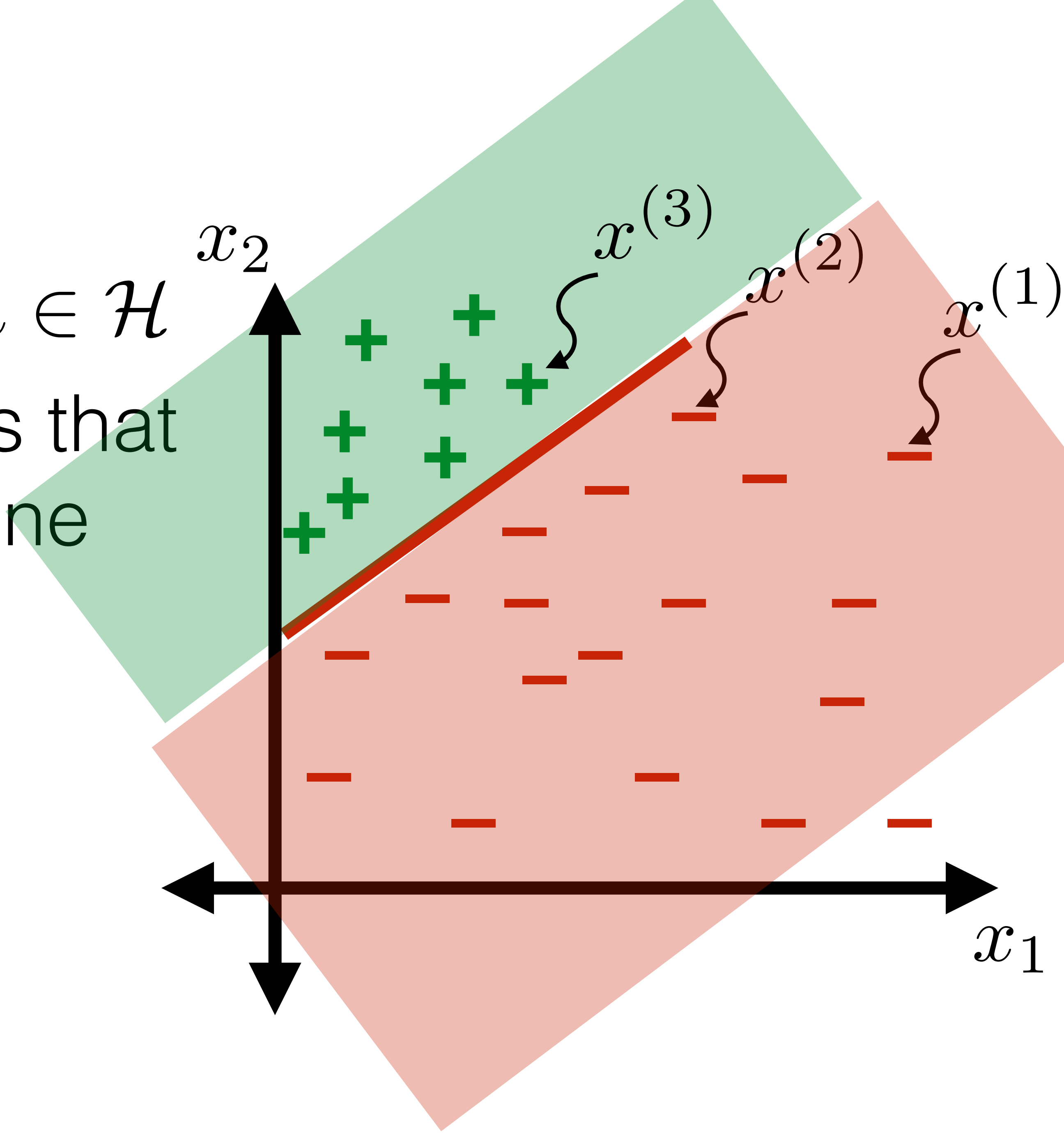
- Hypothesis class $\mathcal{H}$: set of $h \in \mathcal{H}$
  - Example $\mathcal{H}$: All hypotheses that label +1 on one side of a line and -1 on the other side



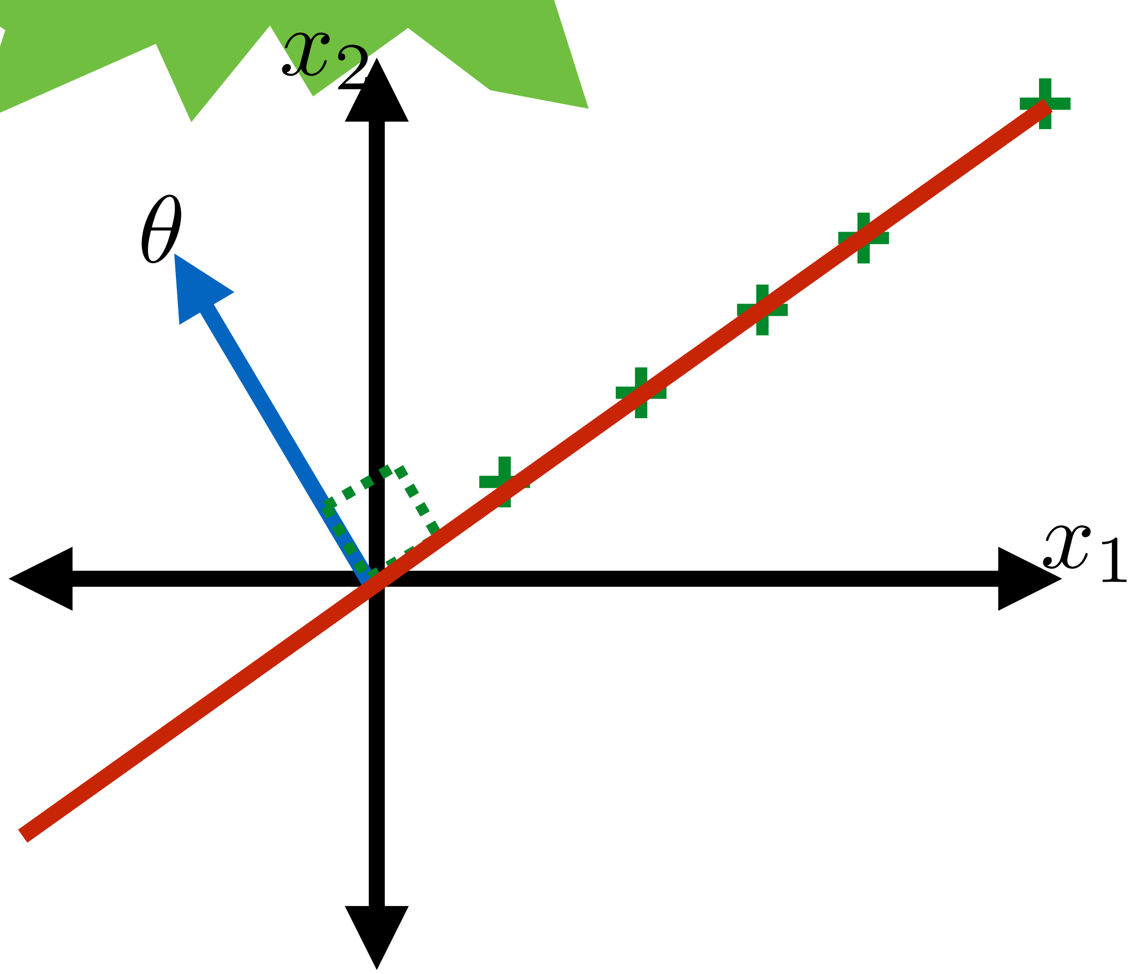


Linear classifiers

- Hypothesis class $\mathcal{H}$: set of $h \in \mathcal{H}$
  - Example $\mathcal{H}$: All hypotheses that label +1 on one side of a line and -1 on the other side

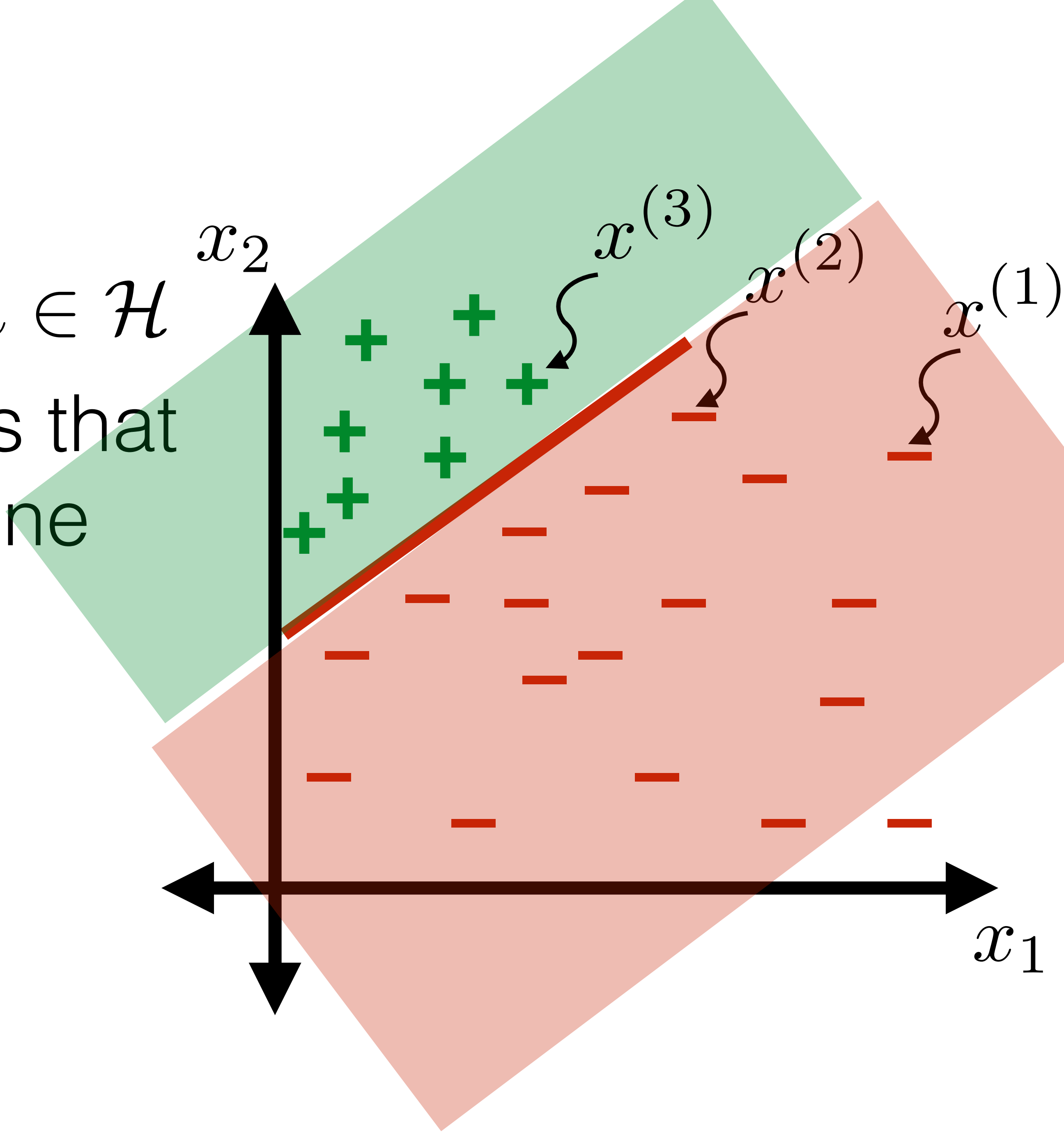


Math facts!

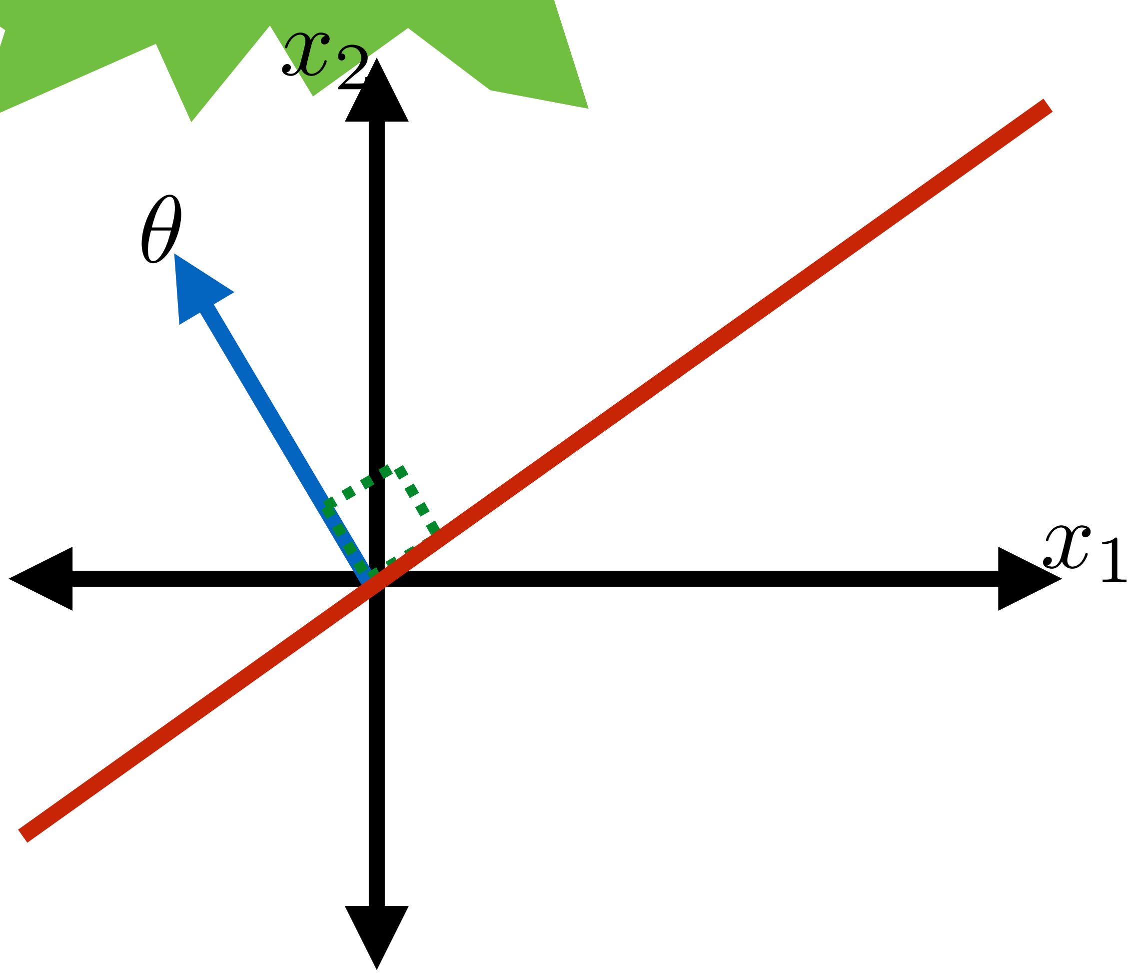


Linear classifiers

- Hypothesis class $\mathcal{H}$: set of $h \in \mathcal{H}$
  - Example $\mathcal{H}$: All hypotheses that label +1 on one side of a line and -1 on the other side

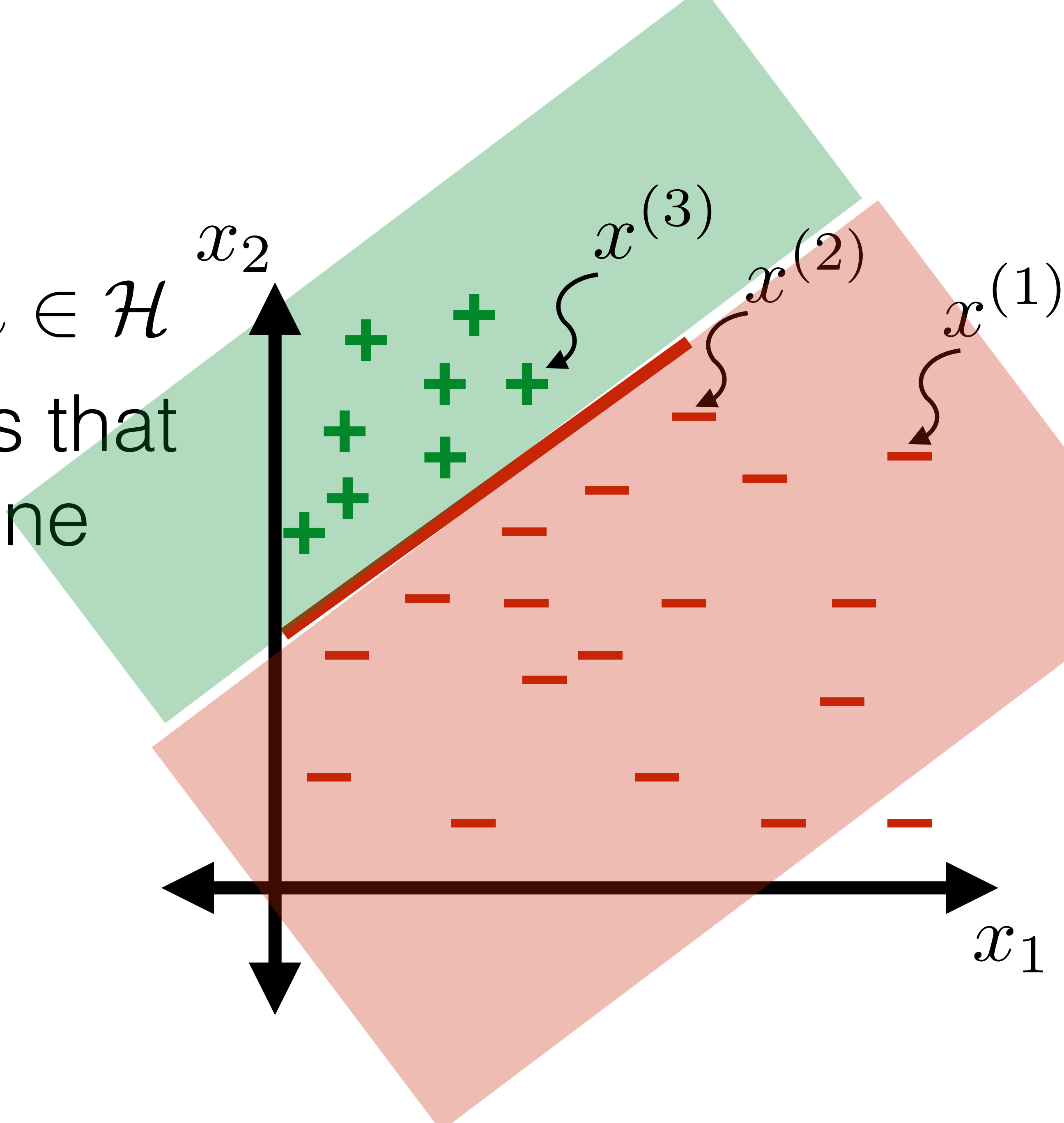


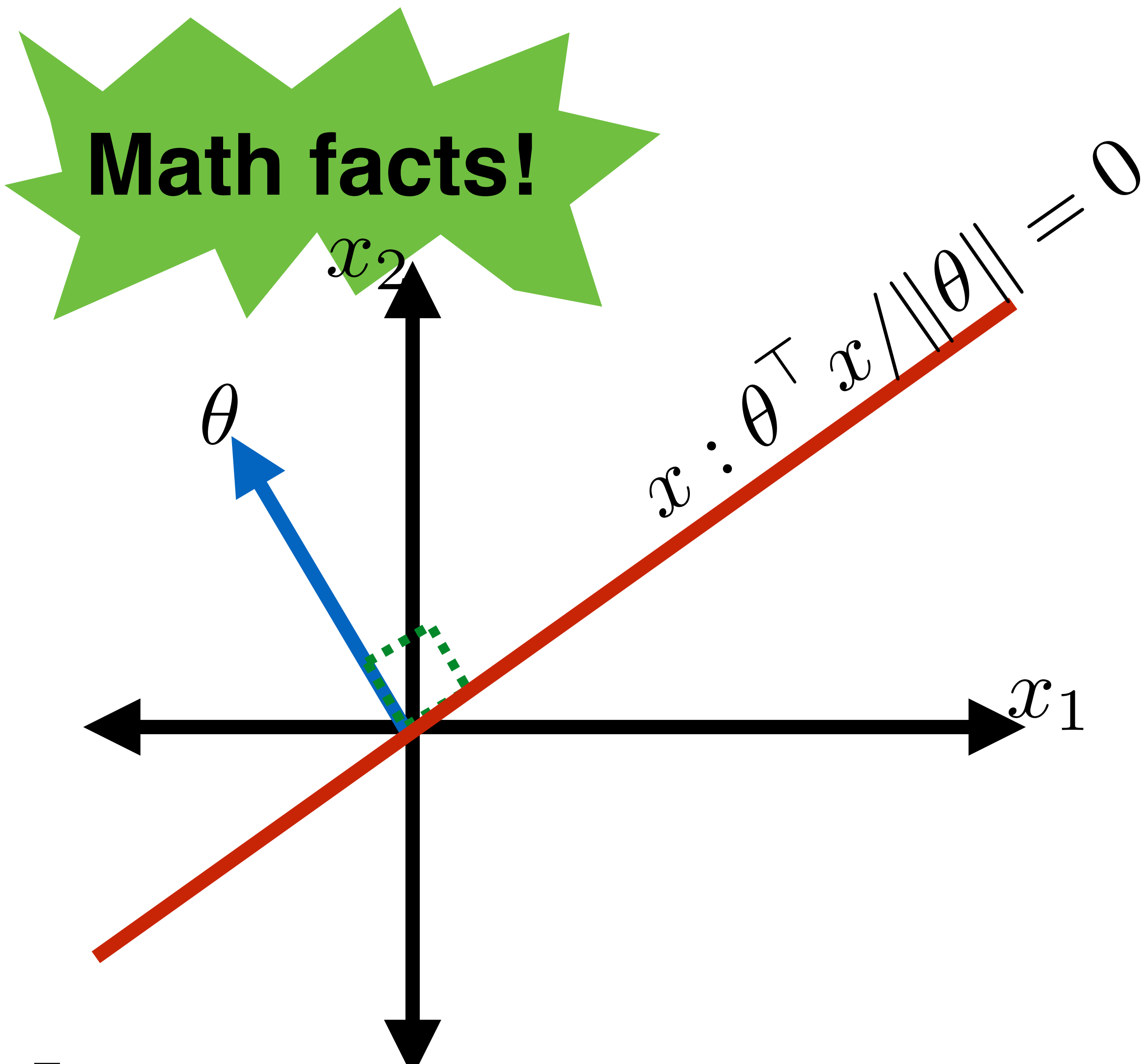
Math facts!



Linear classifiers

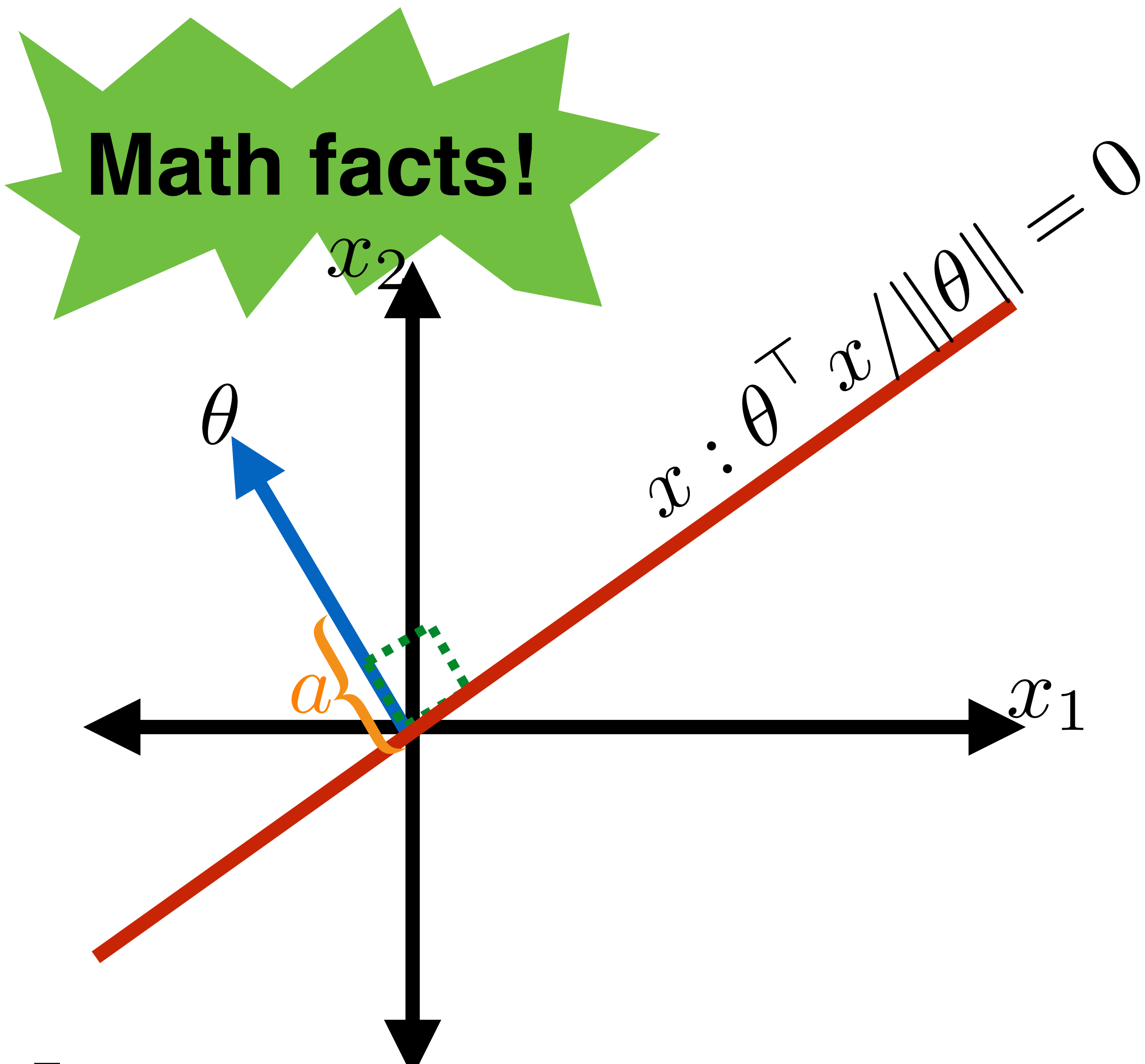
- Hypothesis class $\mathcal{H}$: set of $h \in \mathcal{H}$
  - Example $\mathcal{H}$: All hypotheses that label +1 on one side of a line and -1 on the other side

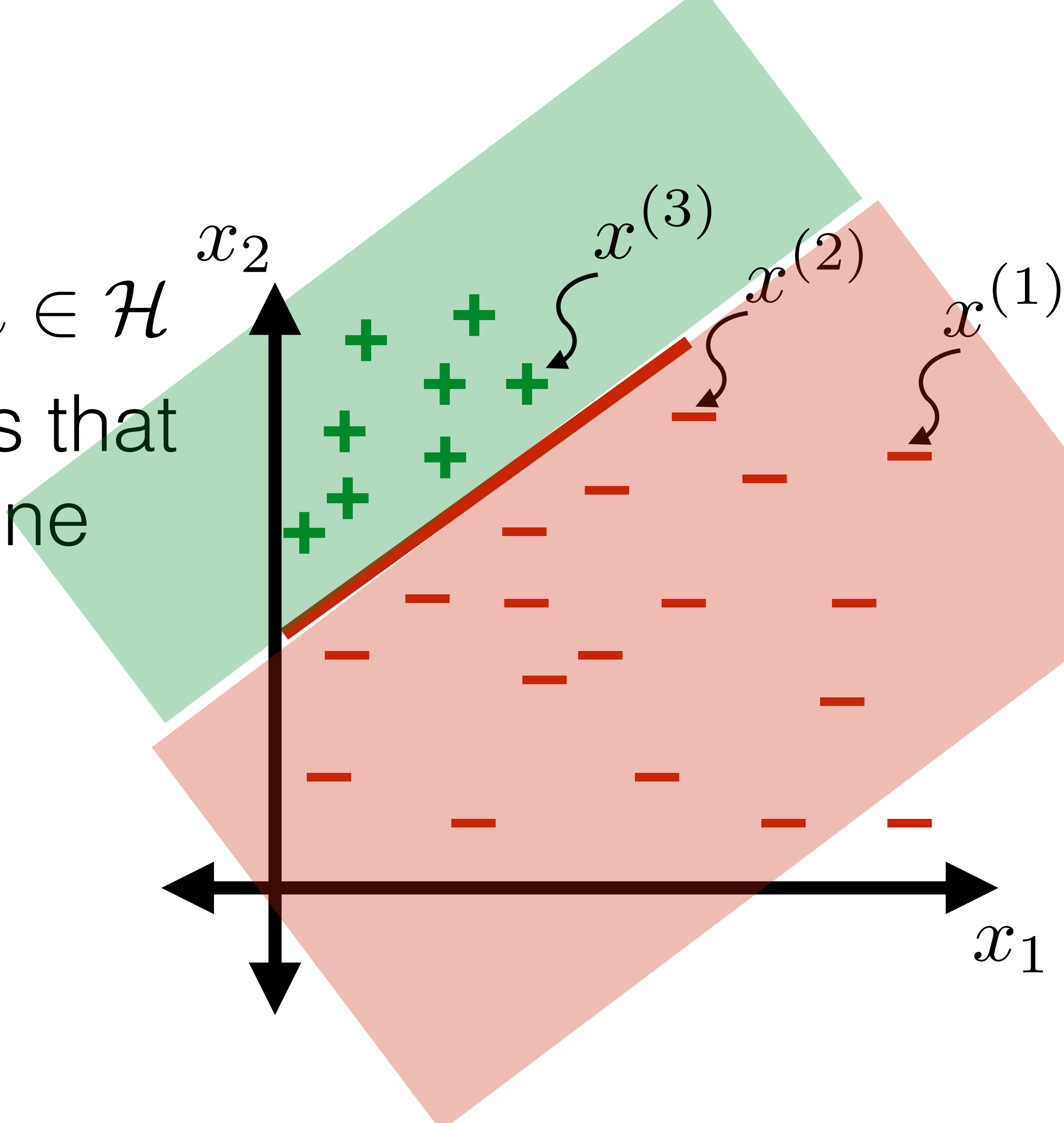




Linear classifiers

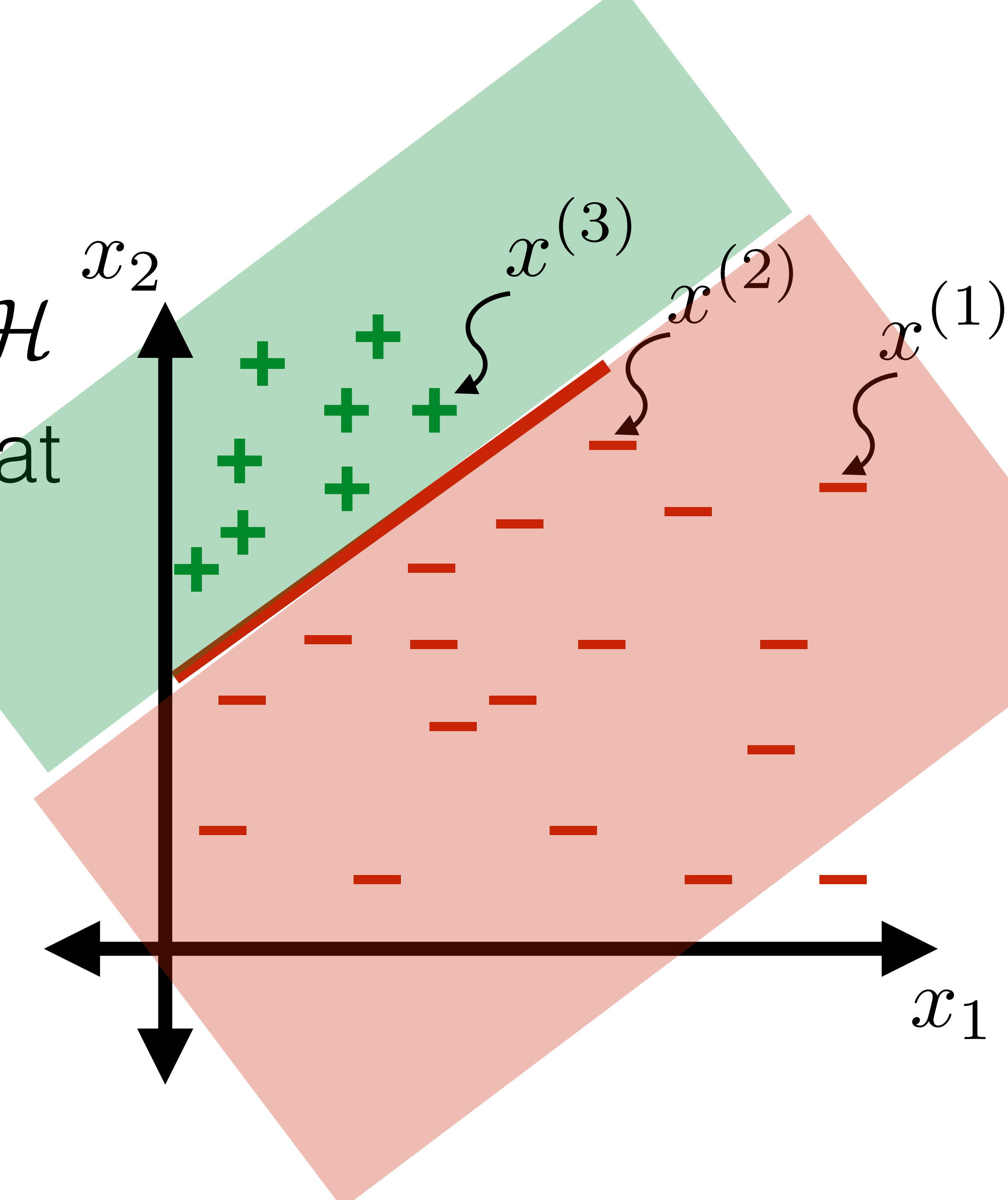
- Hypothesis class $\mathcal{H}$: set of $h \in \mathcal{H}$
    - Example $\mathcal{H}$: All hypotheses that label +1 on one side of a line and -1 on the other side

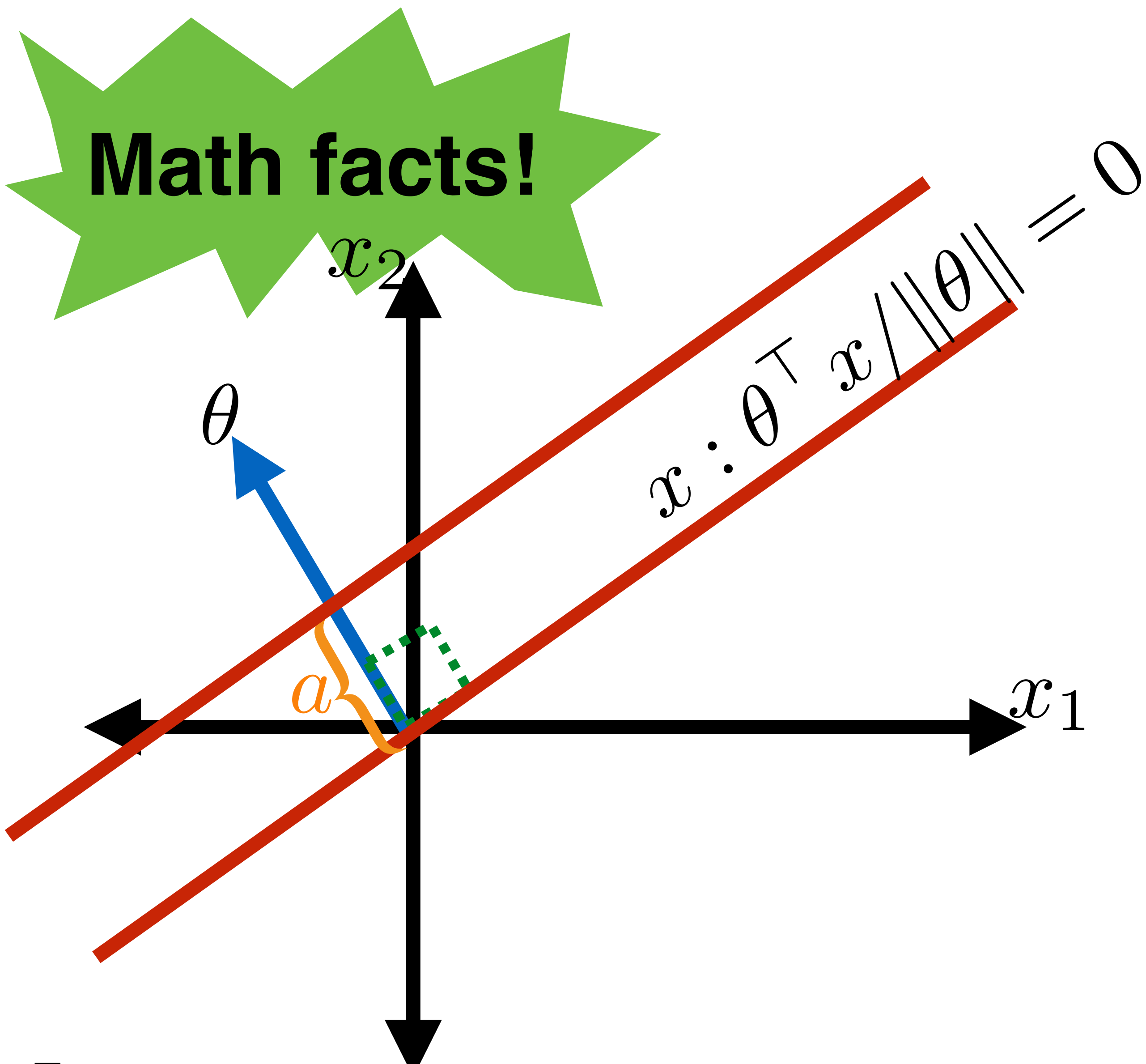




Linear classifiers

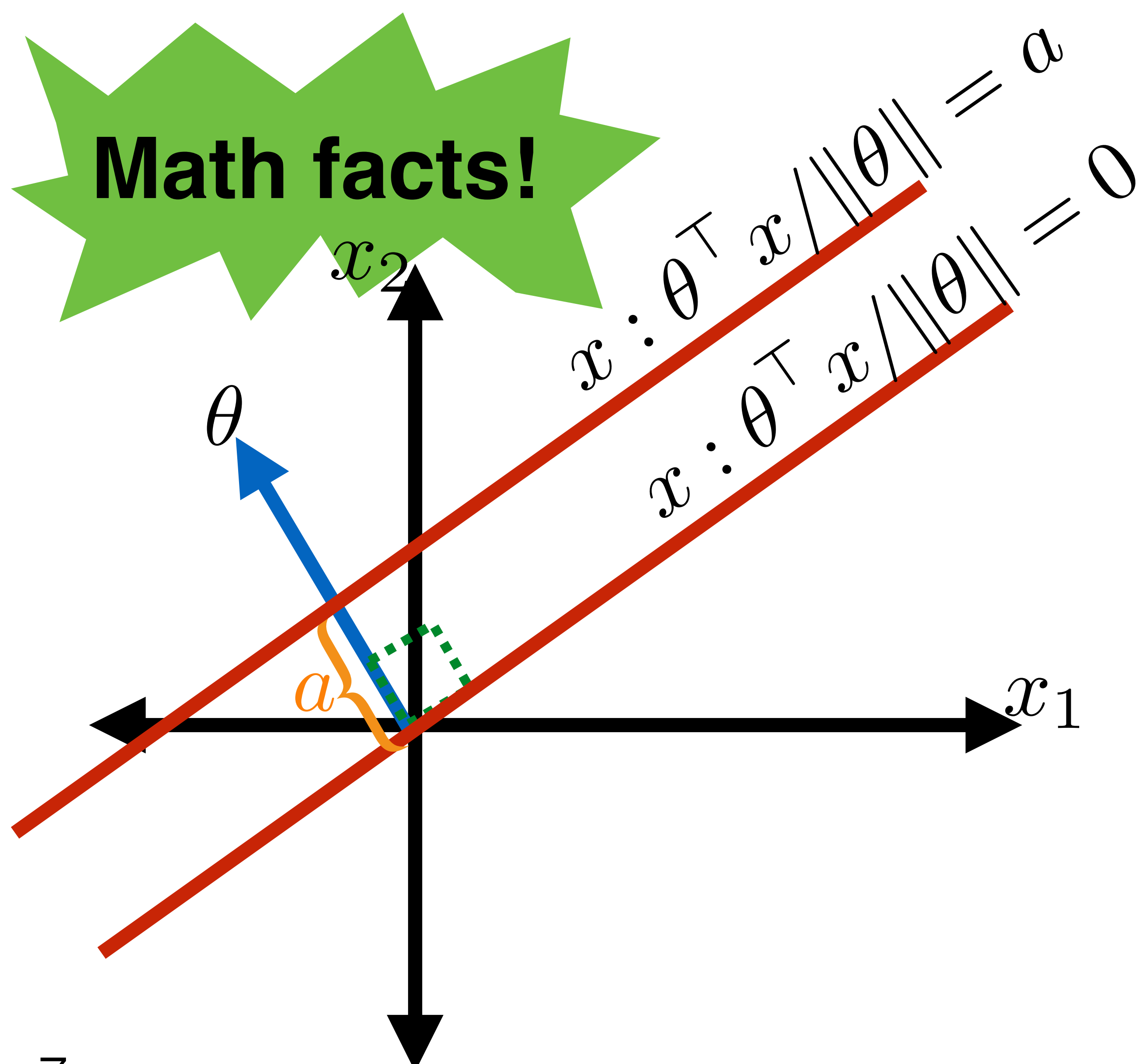
- Hypothesis class $\mathcal{H}$: set of $h \in \mathcal{H}$
    - Example $\mathcal{H}$: All hypotheses that label +1 on one side of a line and -1 on the other side

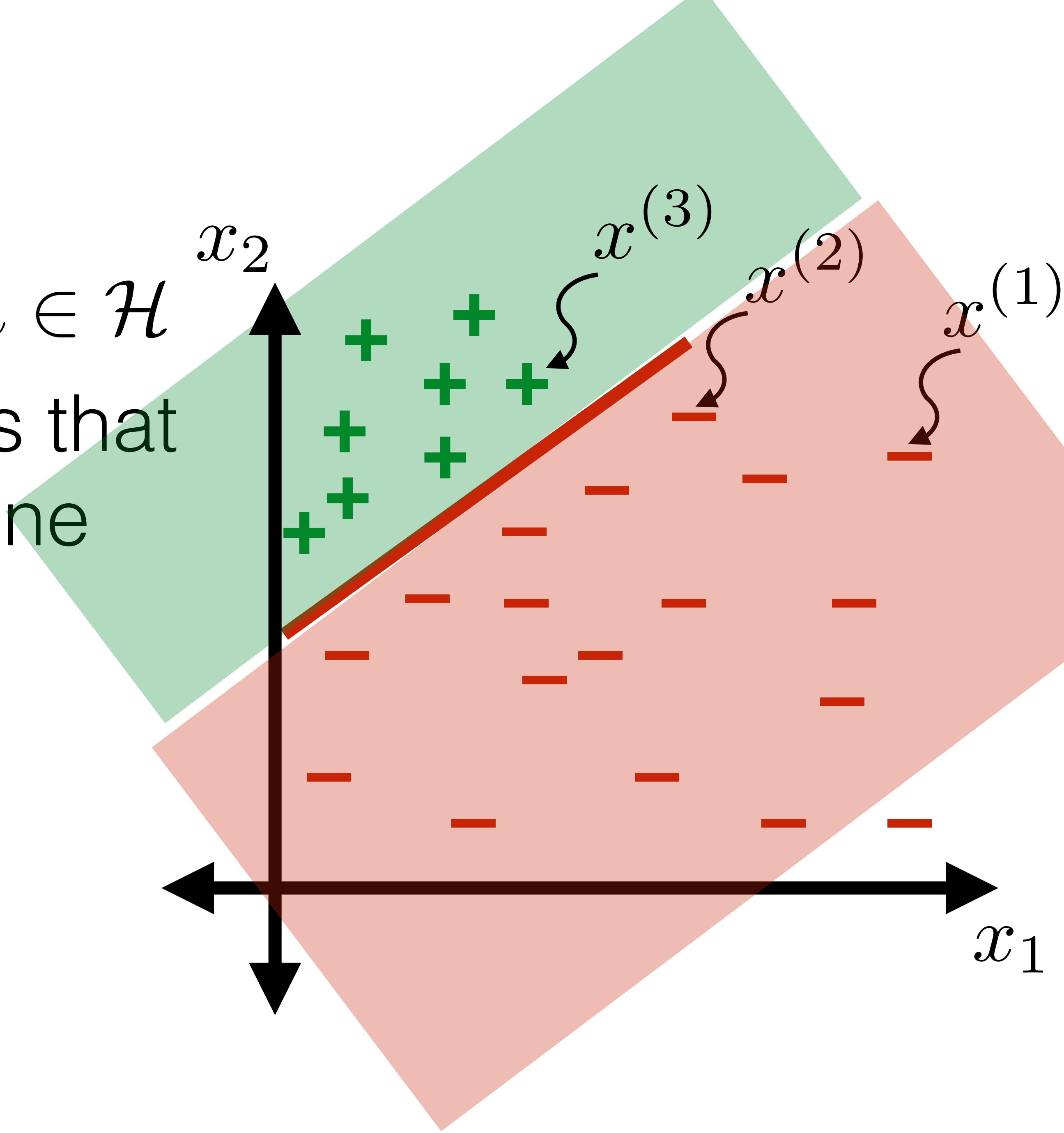




Linear classifiers

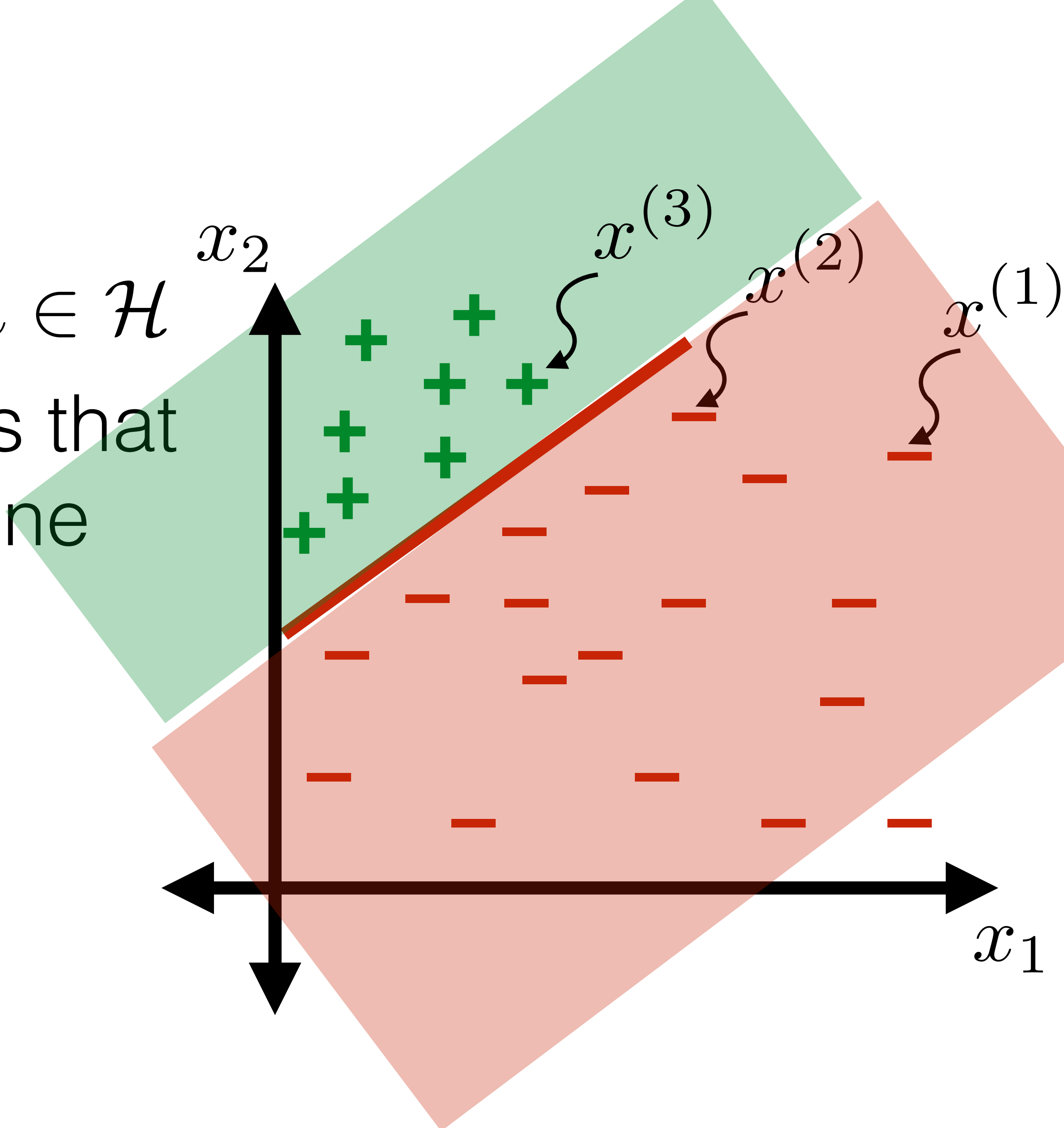
- Hypothesis class $\mathcal{H}$: set of $h \in \mathcal{H}$
  - Example $\mathcal{H}$: All hypotheses that label +1 on one side of a line and -1 on the other side

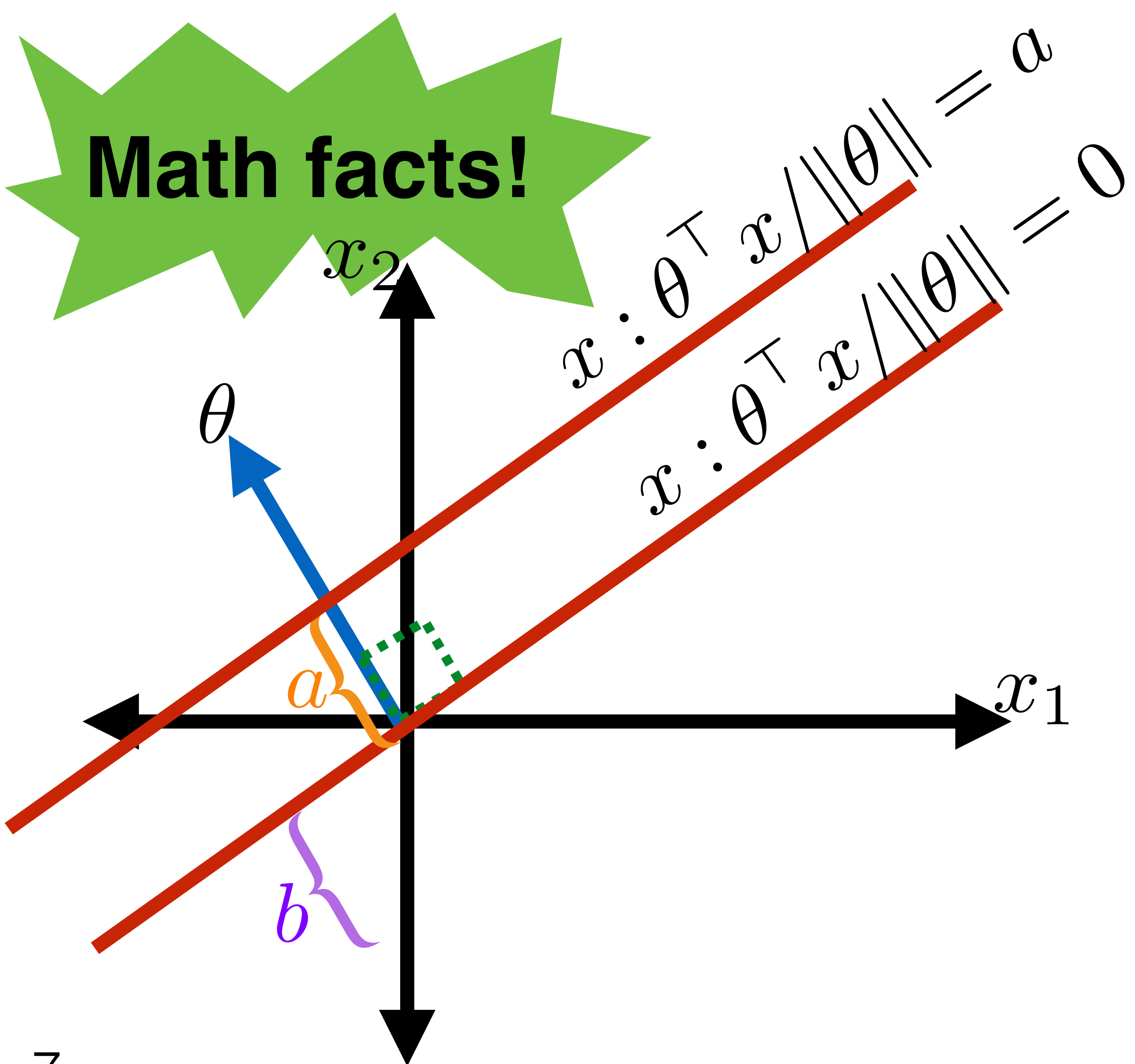




Linear classifiers

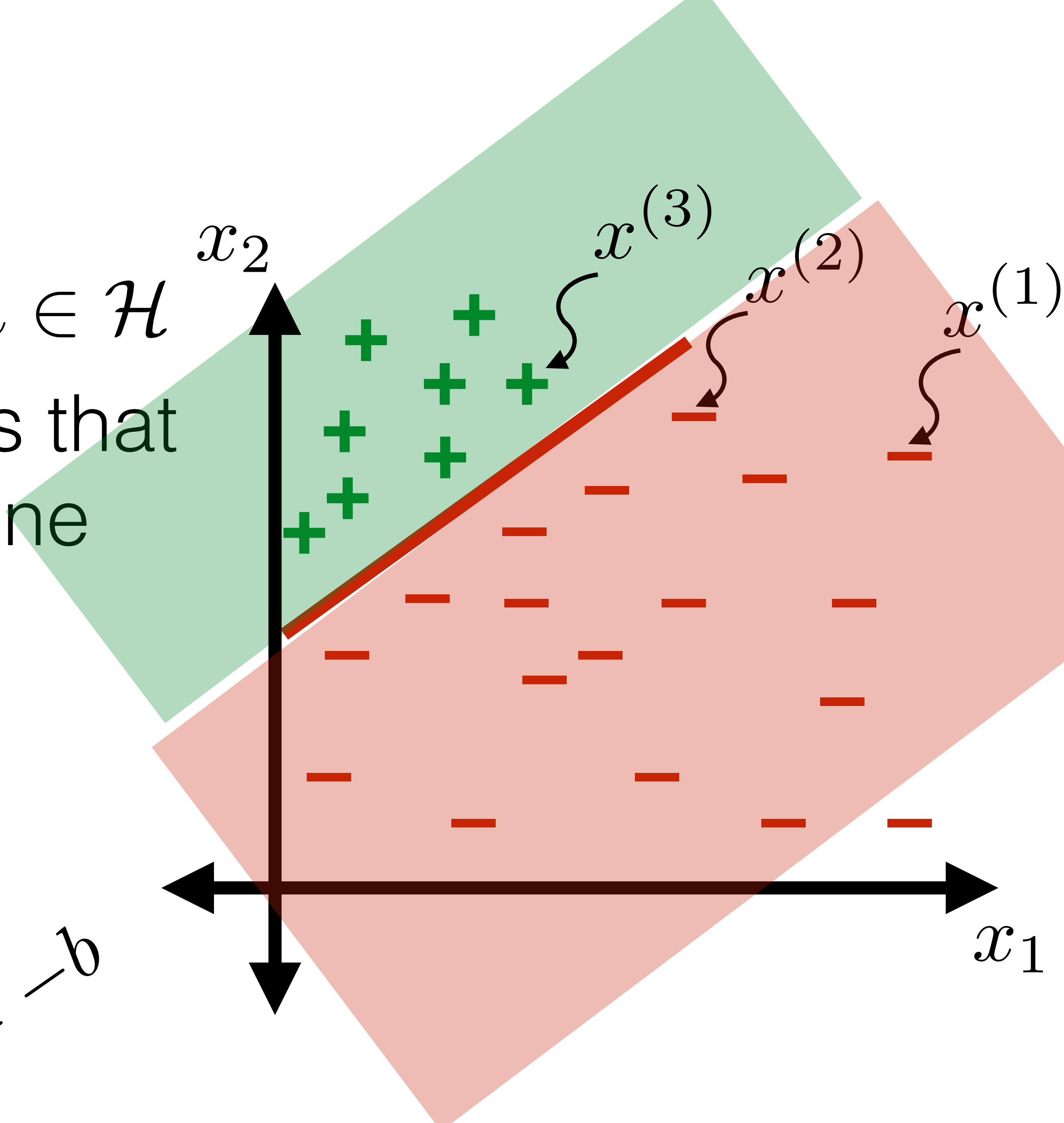
- Hypothesis class $\mathcal{H}$: set of $h \in \mathcal{H}$
  - Example $\mathcal{H}$: All hypotheses that label +1 on one side of a line and -1 on the other side

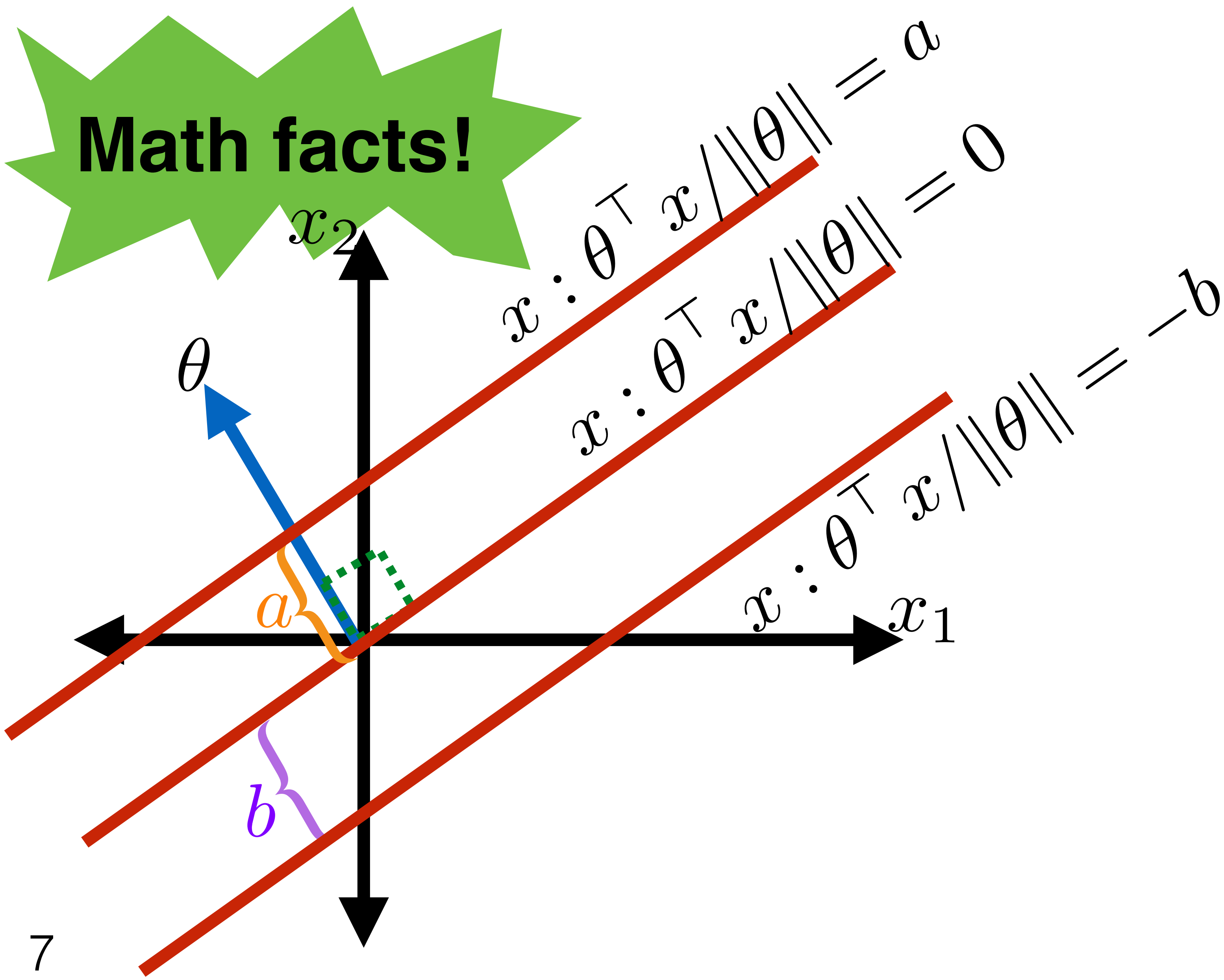




Linear classifiers

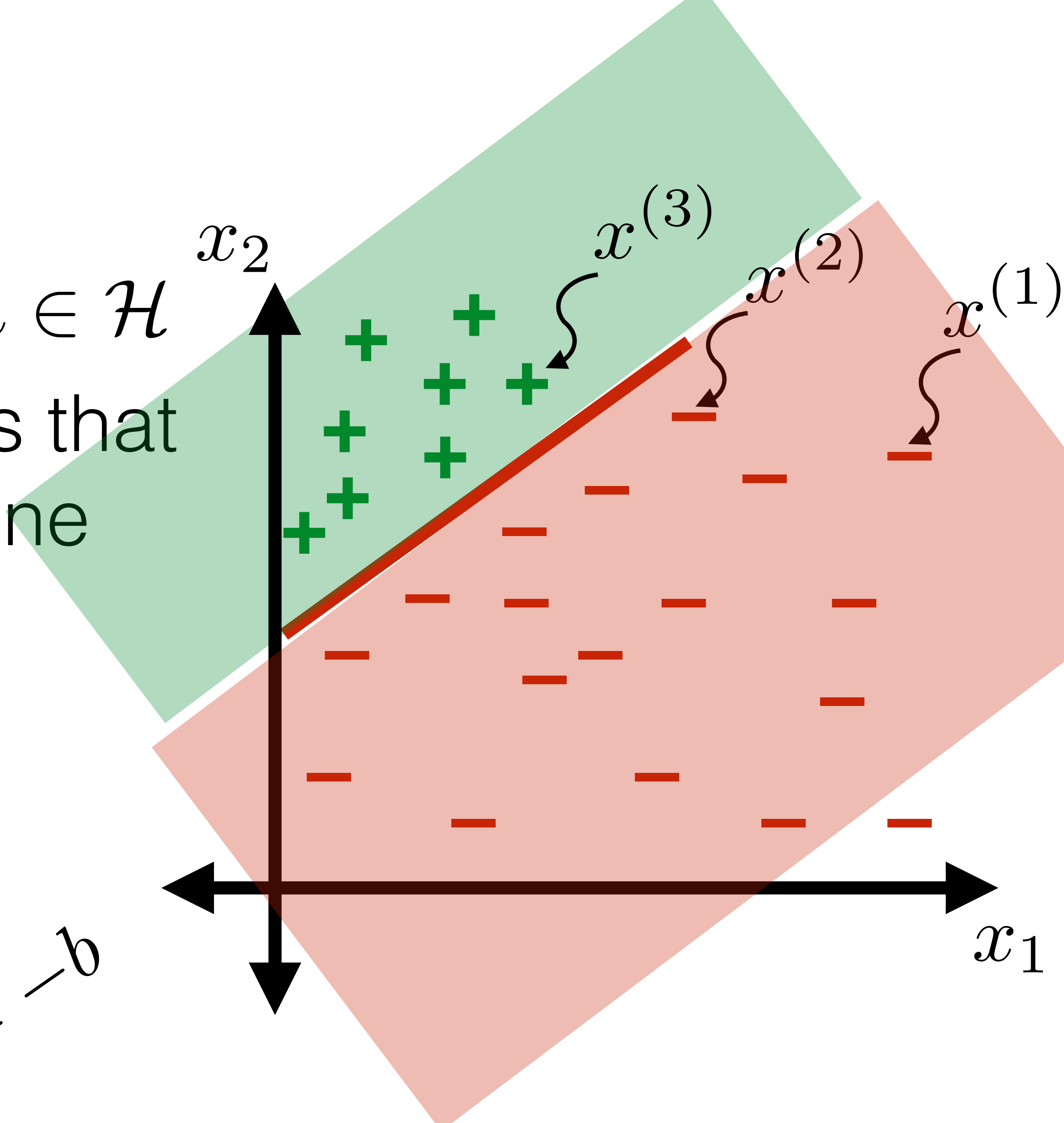
- Hypothesis class $\mathcal{H}$: set of $h \in \mathcal{H}$
  - Example $\mathcal{H}$: All hypotheses that label +1 on one side of a line and -1 on the other side

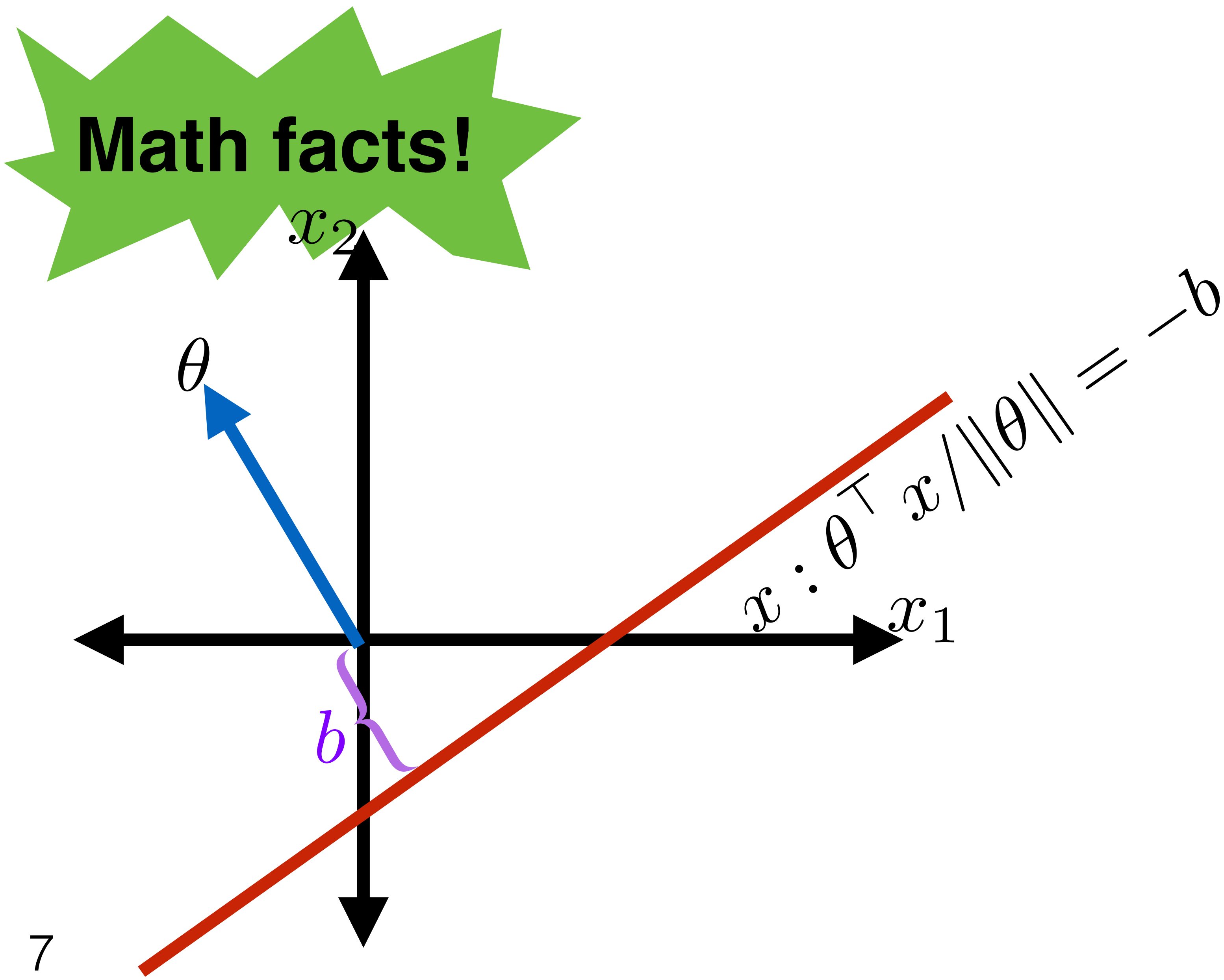




Linear classifiers

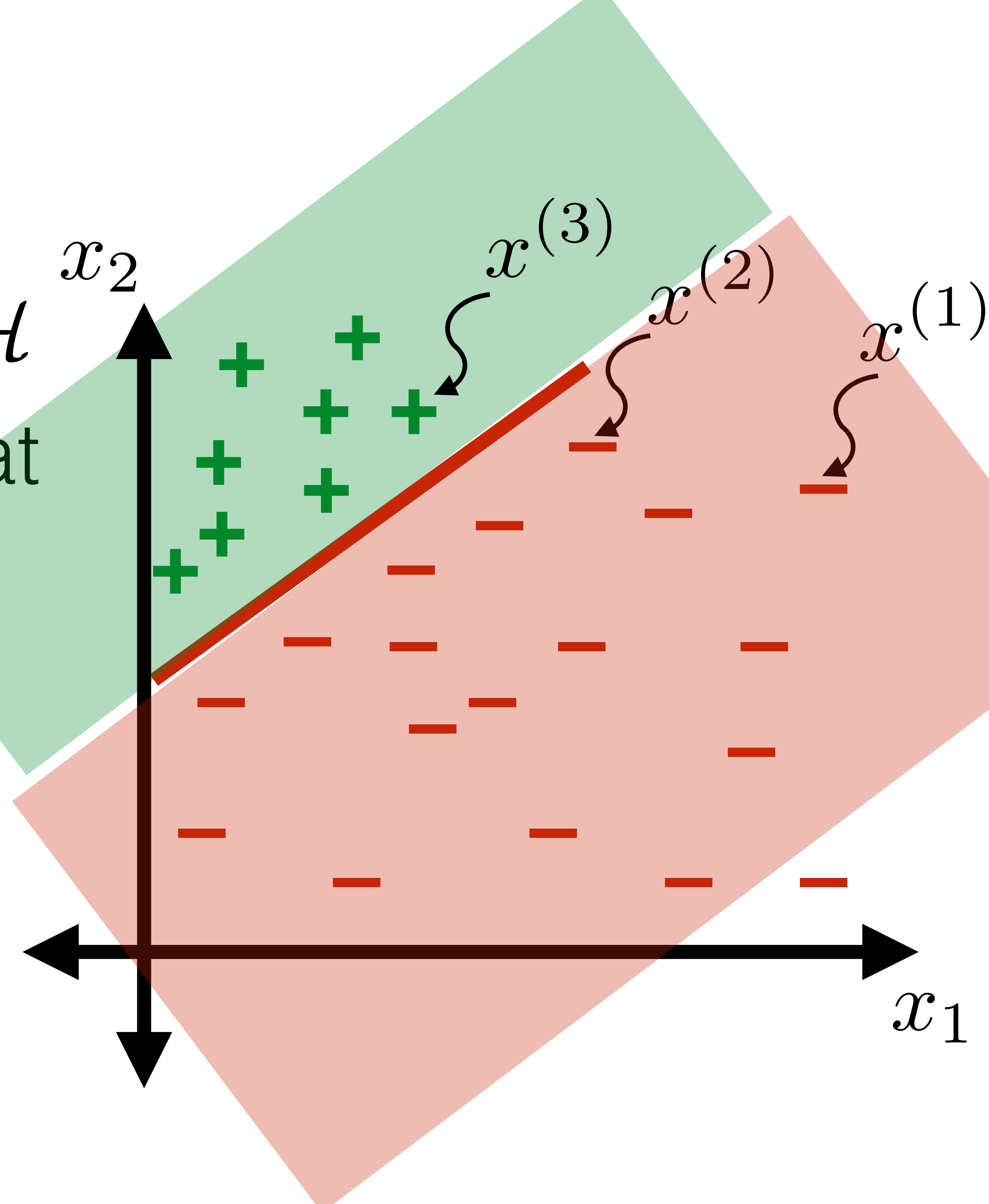
- Hypothesis class $\mathcal{H}$: set of $h \in \mathcal{H}$
  - Example $\mathcal{H}$: All hypotheses that label +1 on one side of a line and -1 on the other side

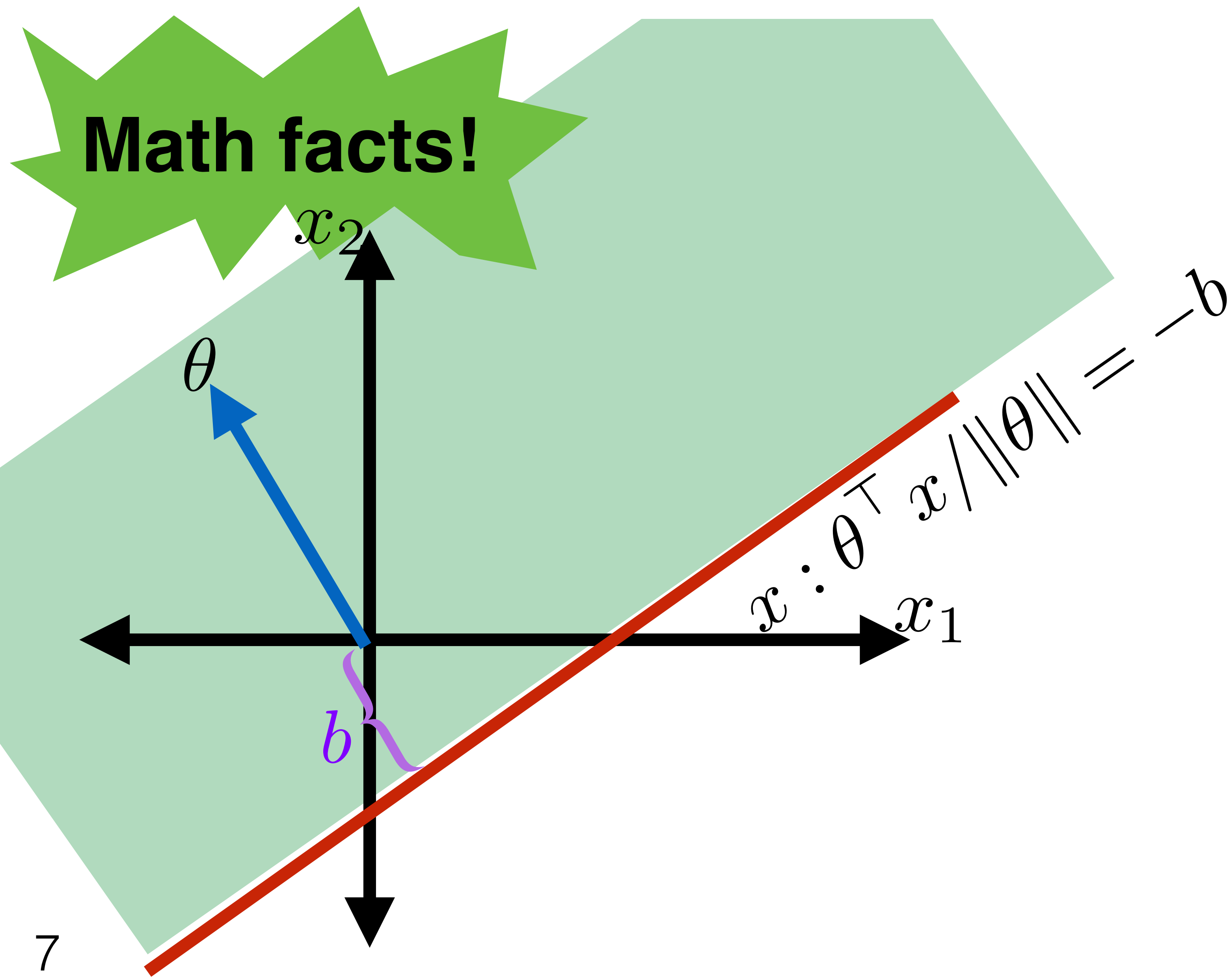




Linear classifiers

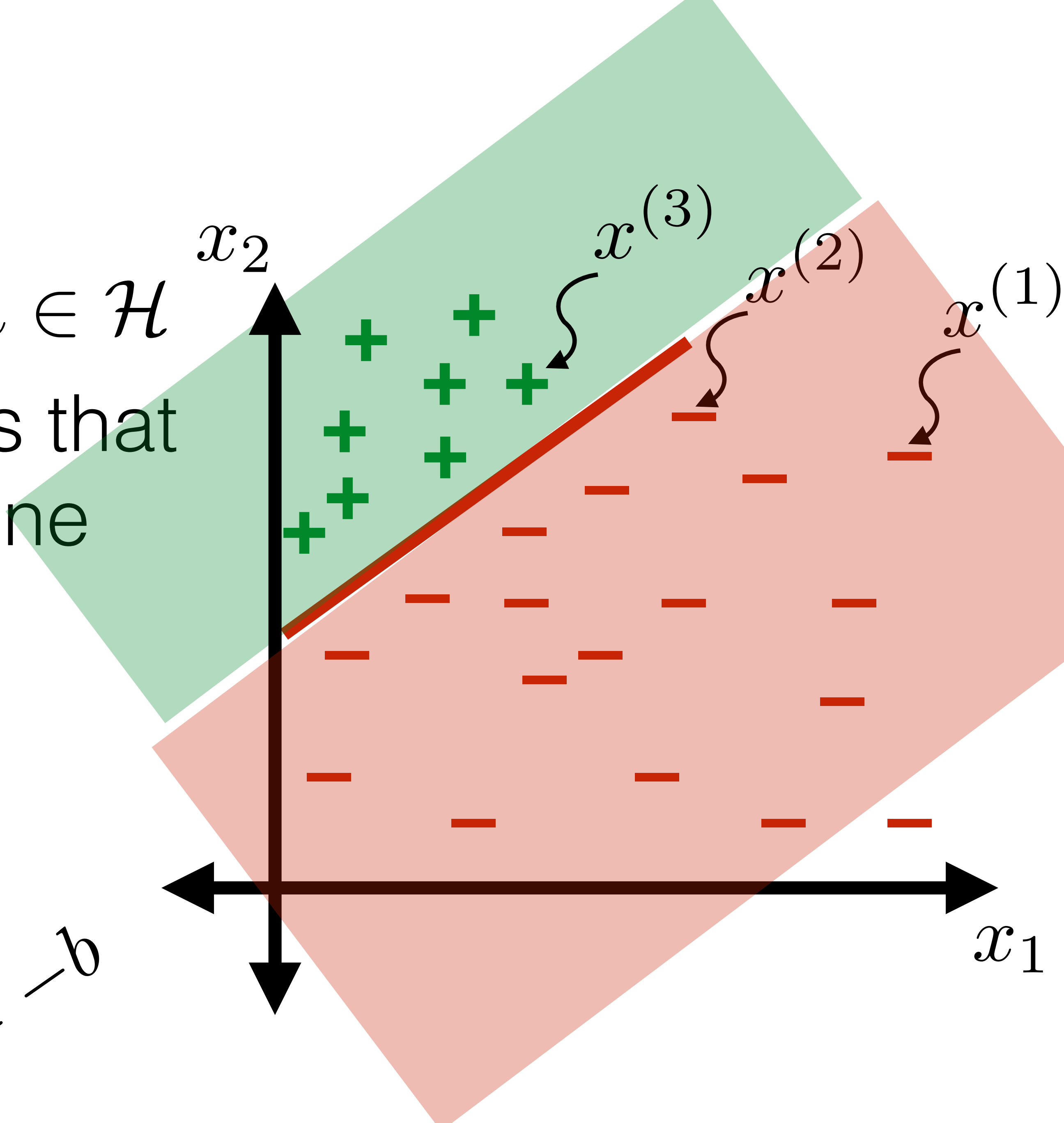
- Hypothesis class $\mathcal{H}$: set of $h \in \mathcal{H}$
  - Example $\mathcal{H}$: All hypotheses that label +1 on one side of a line and -1 on the other side

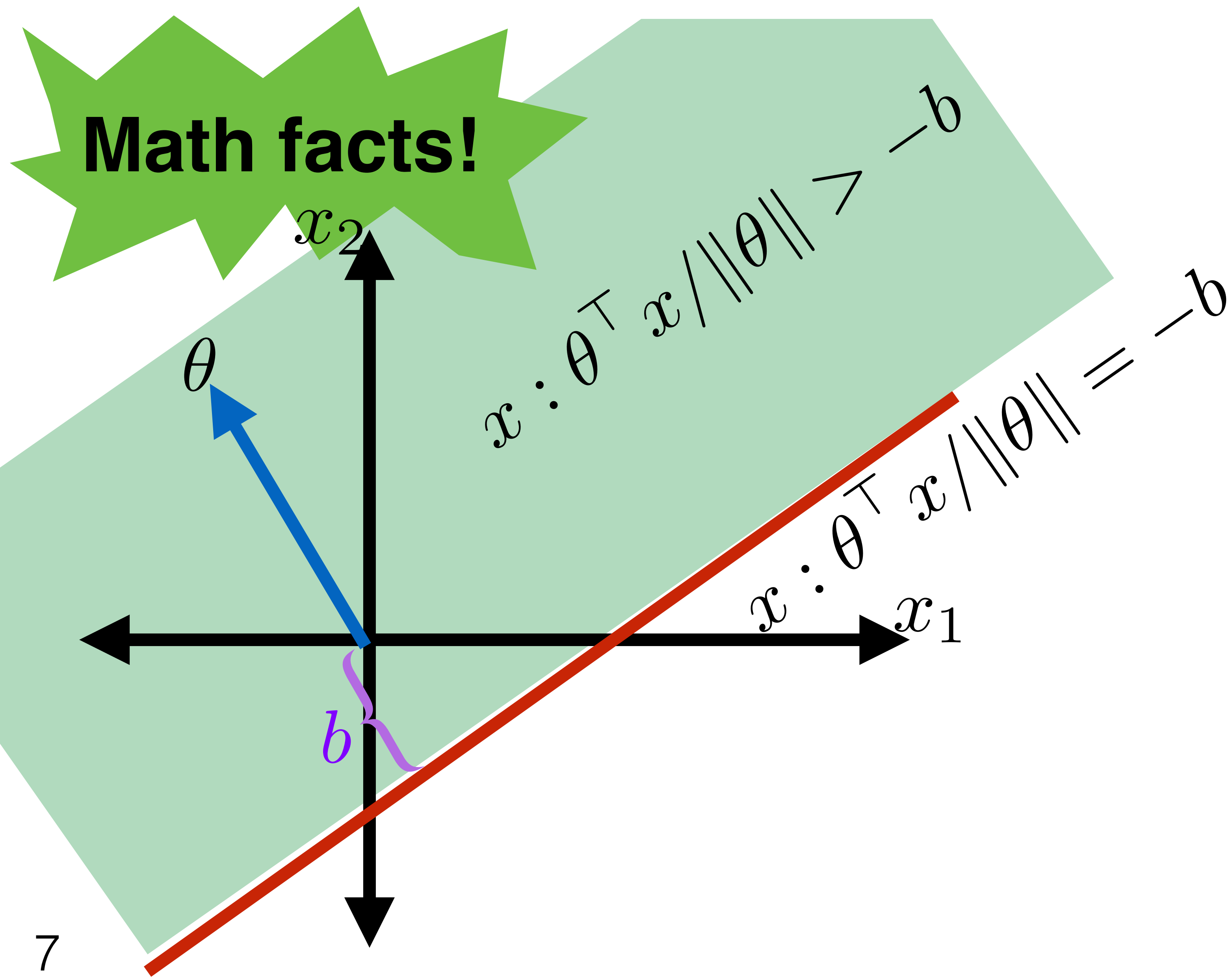




Linear classifiers

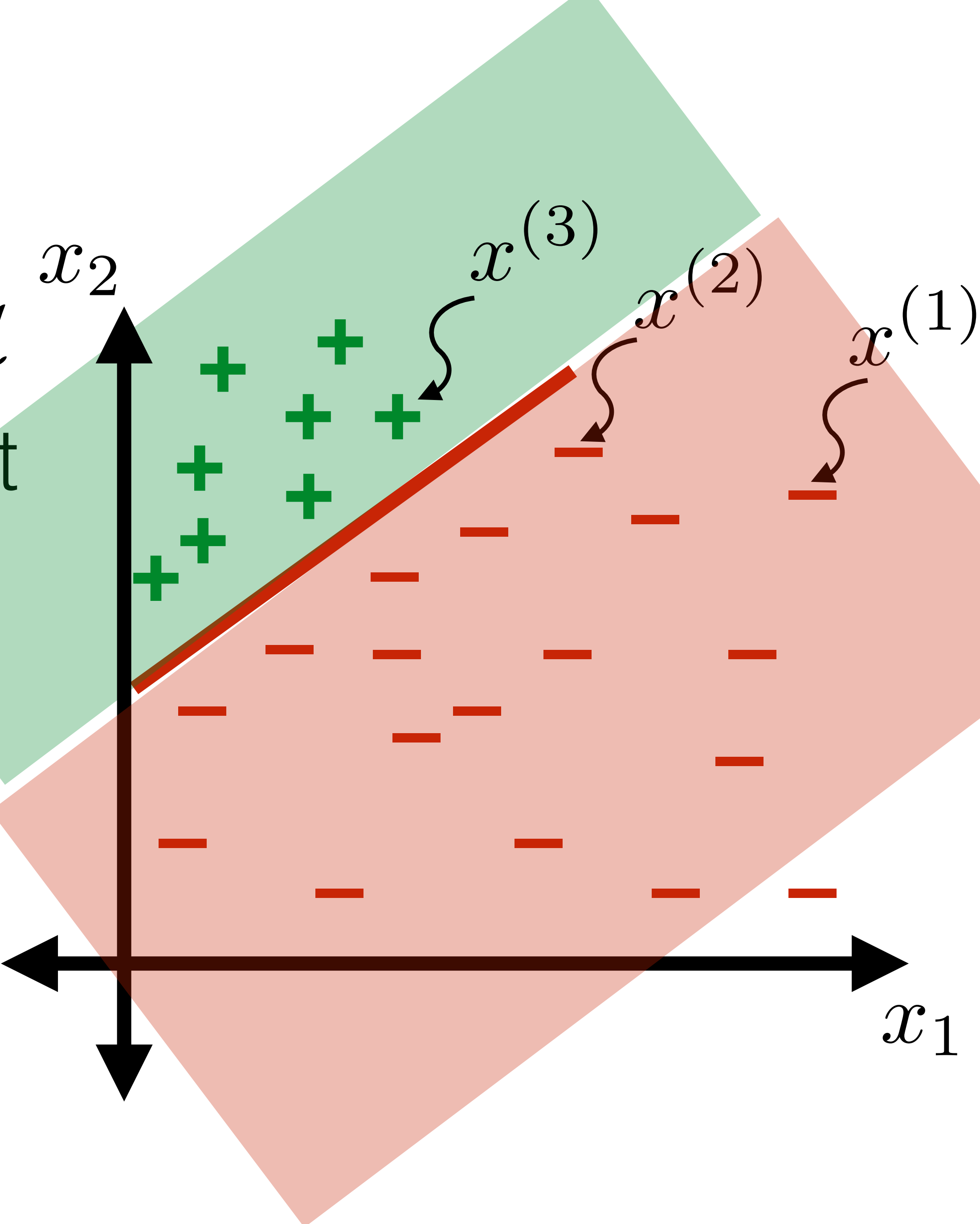
- Hypothesis class $\mathcal{H}$: set of $h \in \mathcal{H}$
  - Example $\mathcal{H}$: All hypotheses that label +1 on one side of a line and -1 on the other side

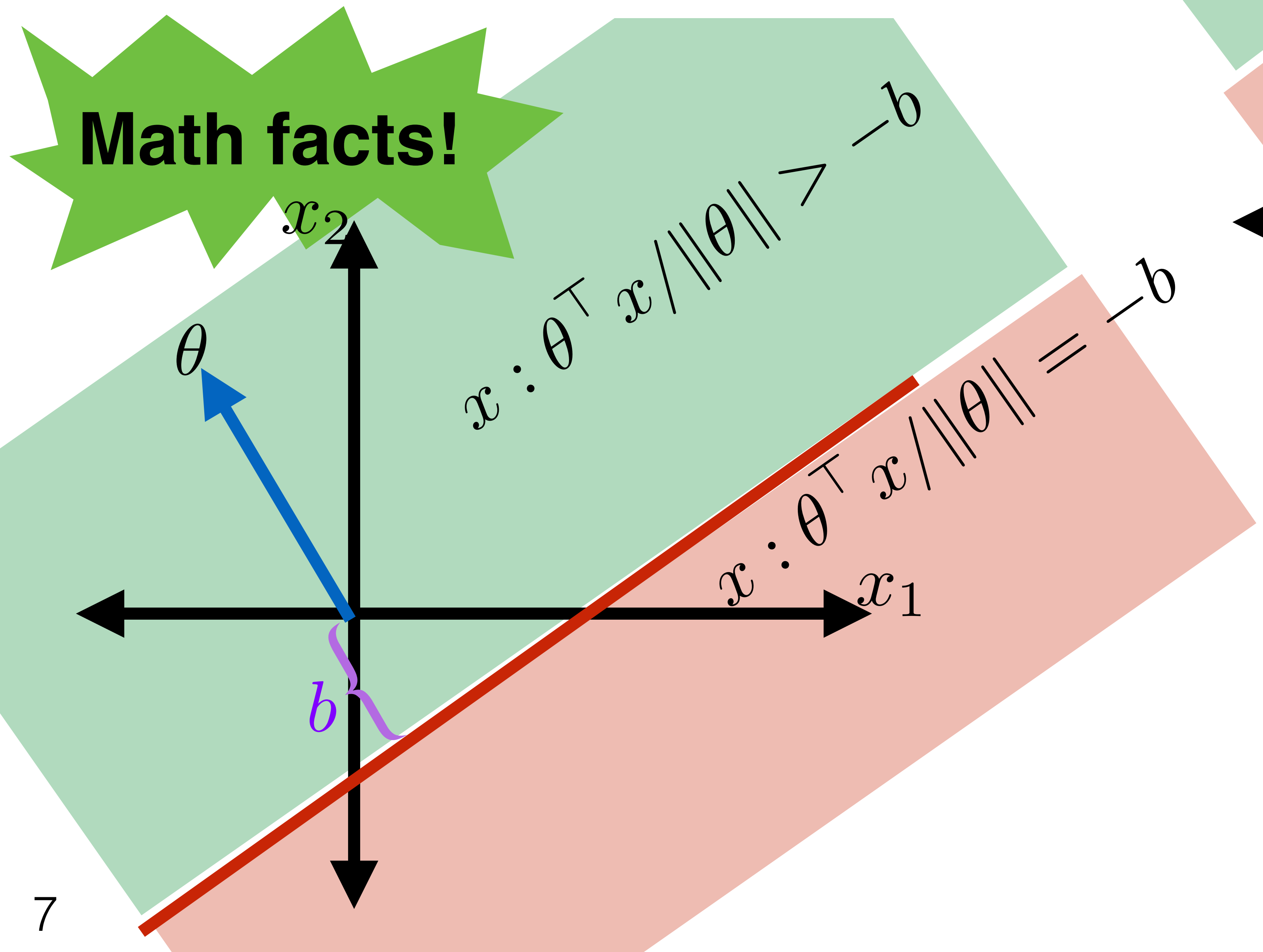




Linear classifiers

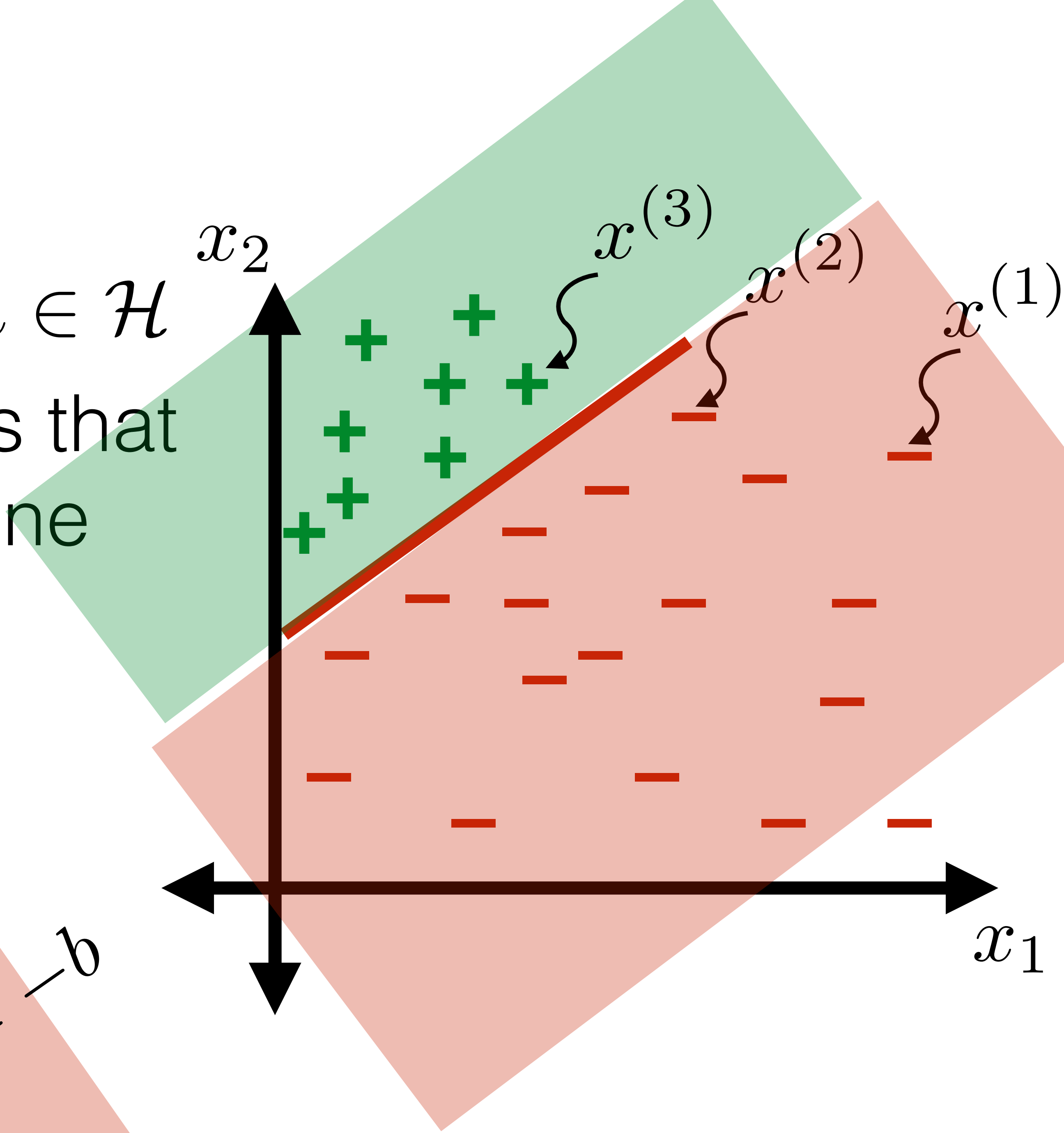
- Hypothesis class $\mathcal{H}$: set of $h \in \mathcal{H}$
    - Example $\mathcal{H}$: All hypotheses that label +1 on one side of a line and -1 on the other side



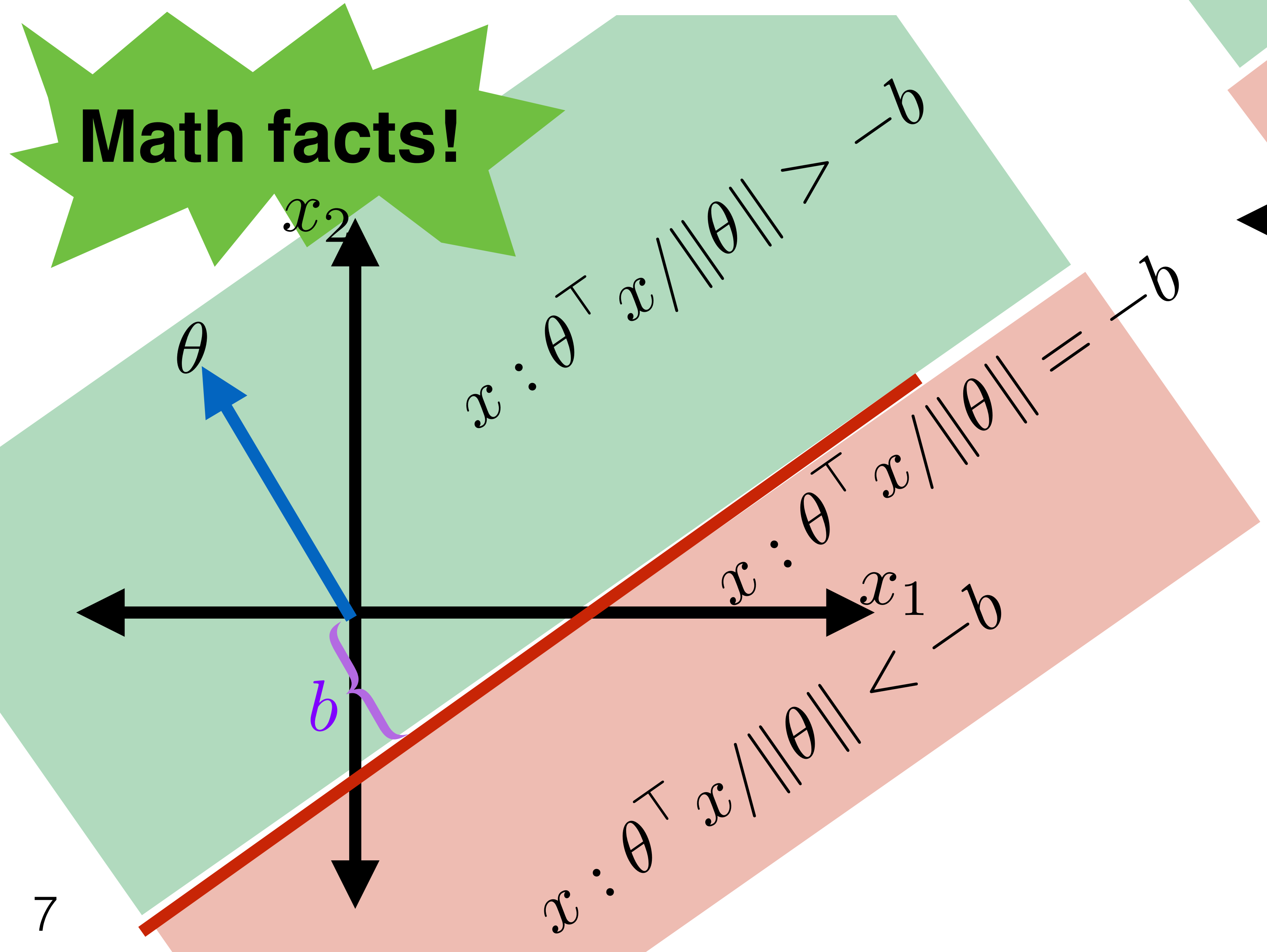


Linear classifiers

- Hypothesis class $\mathcal{H}$: set of $h \in \mathcal{H}$
  - Example $\mathcal{H}$: All hypotheses that label +1 on one side of a line and -1 on the other side

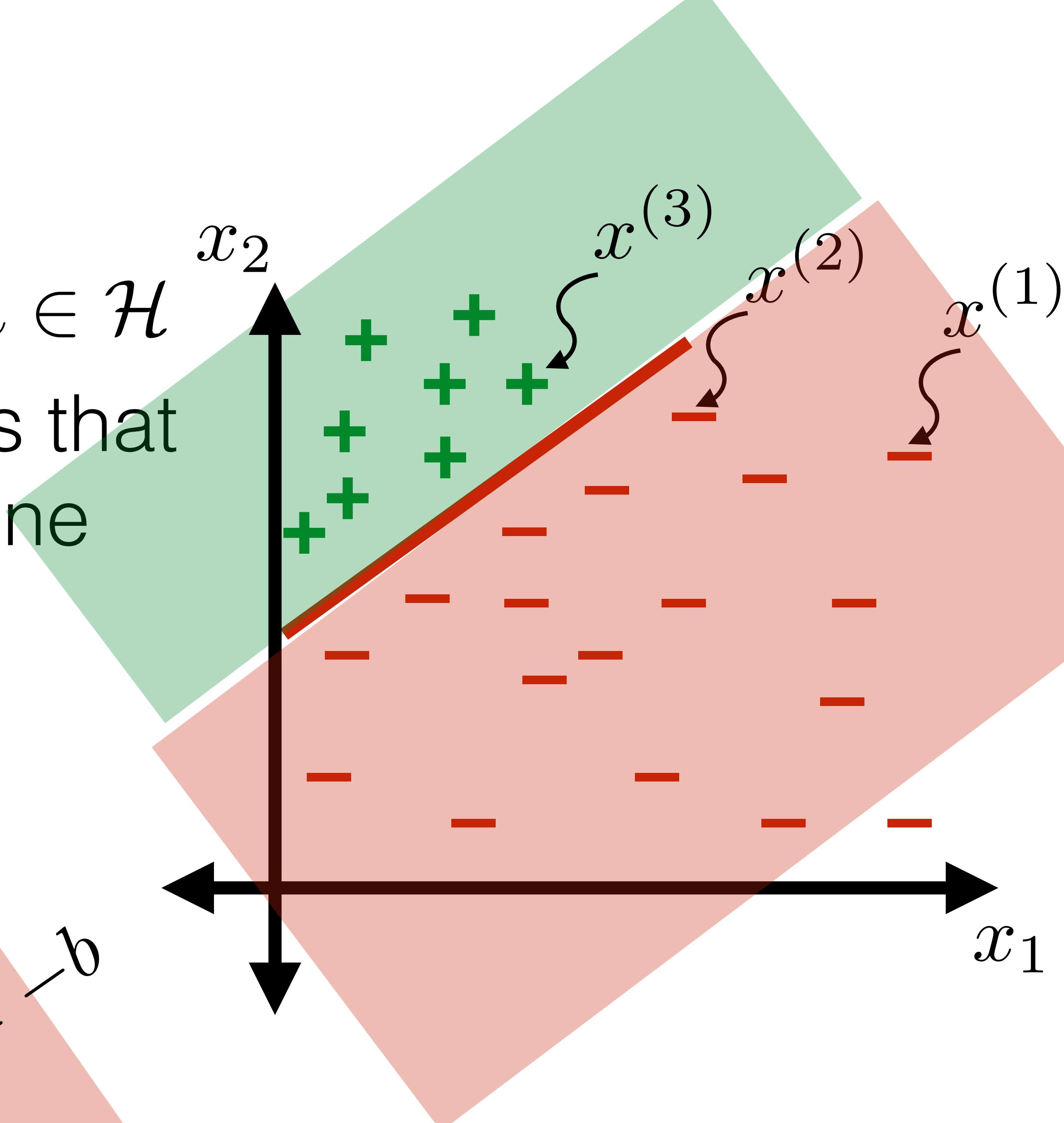


Math facts!

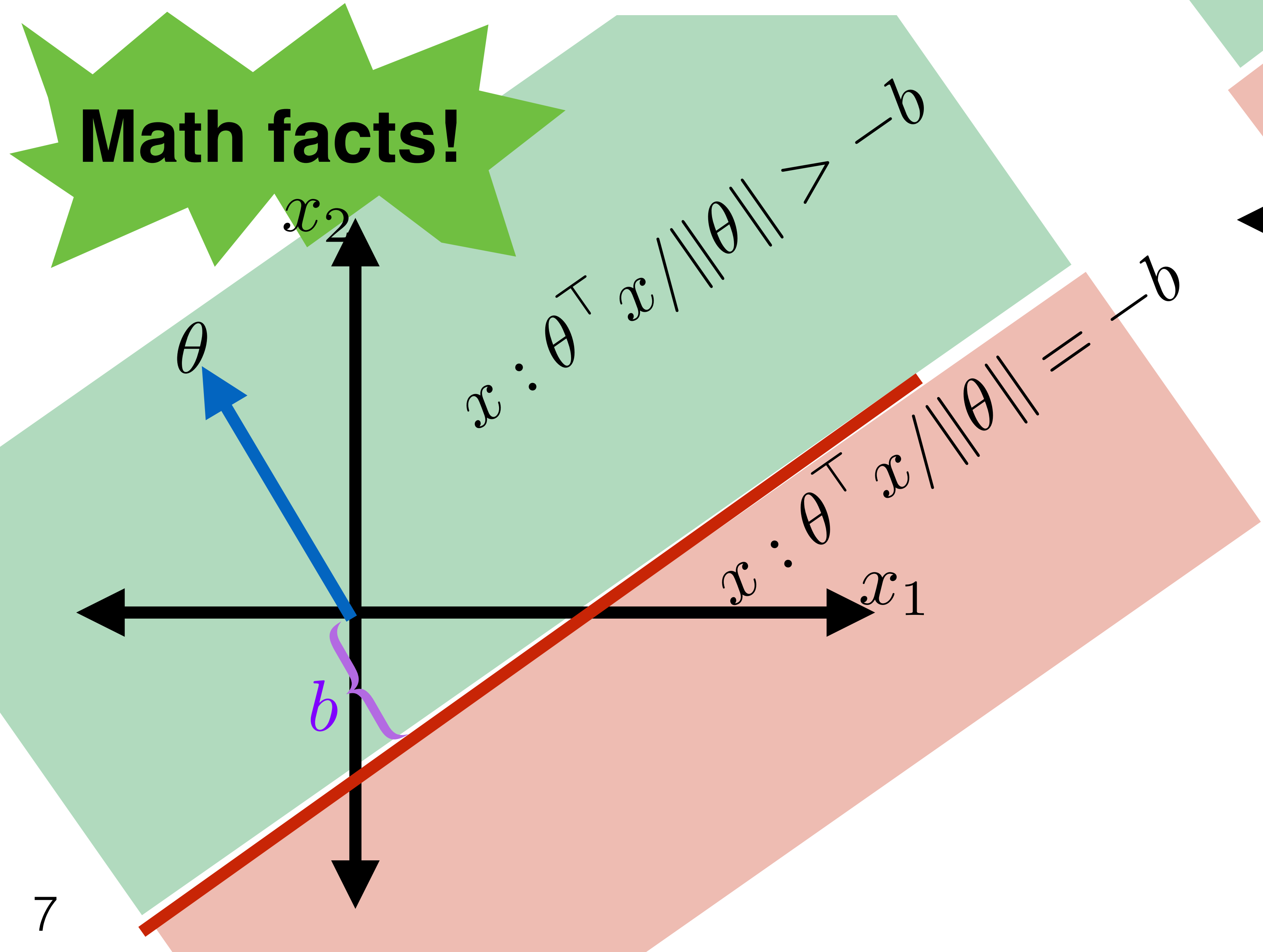


Linear classifiers

- Hypothesis class $\mathcal{H}$: set of $h \in \mathcal{H}$
  - Example $\mathcal{H}$: All hypotheses that label +1 on one side of a line and -1 on the other side

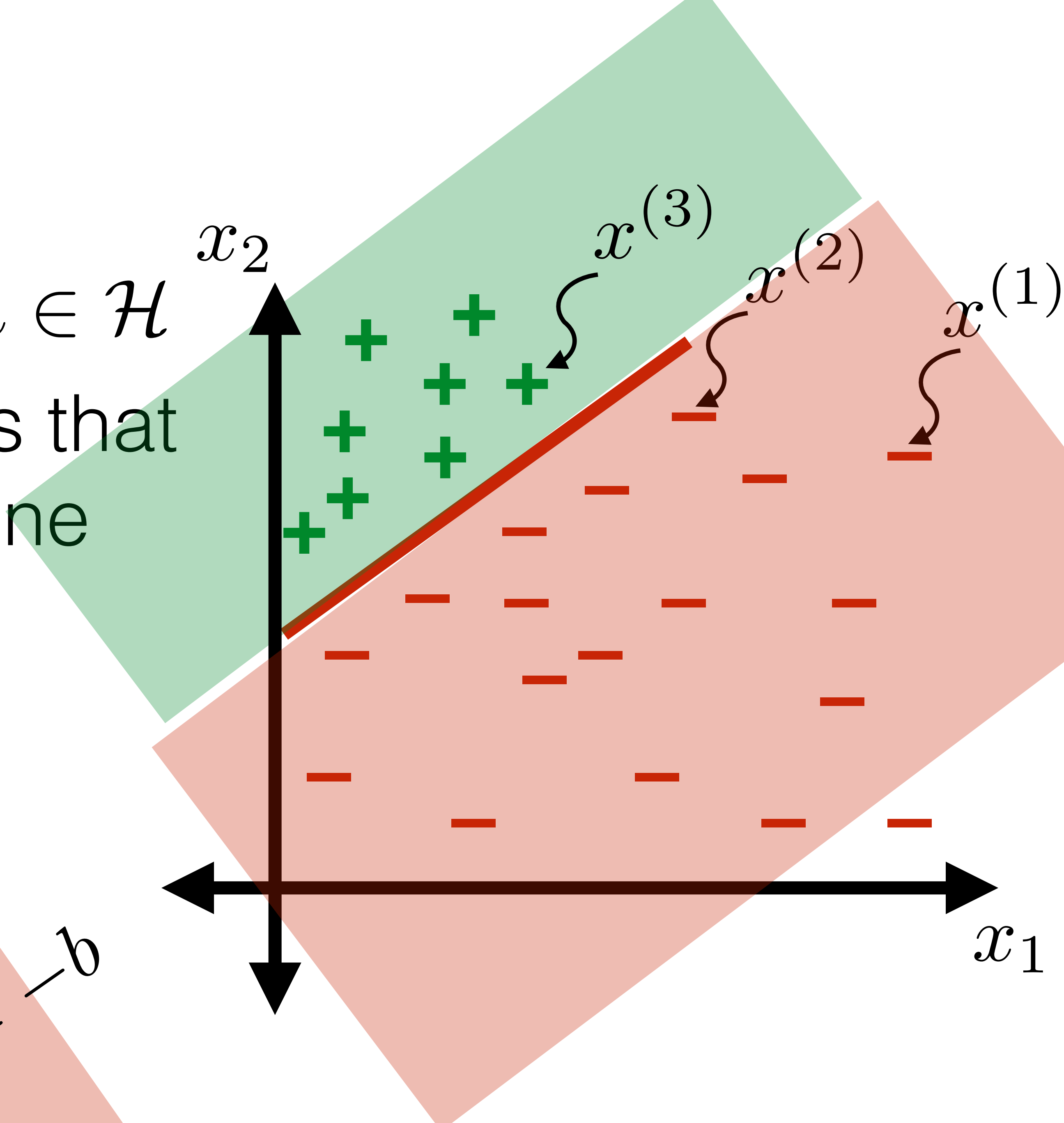


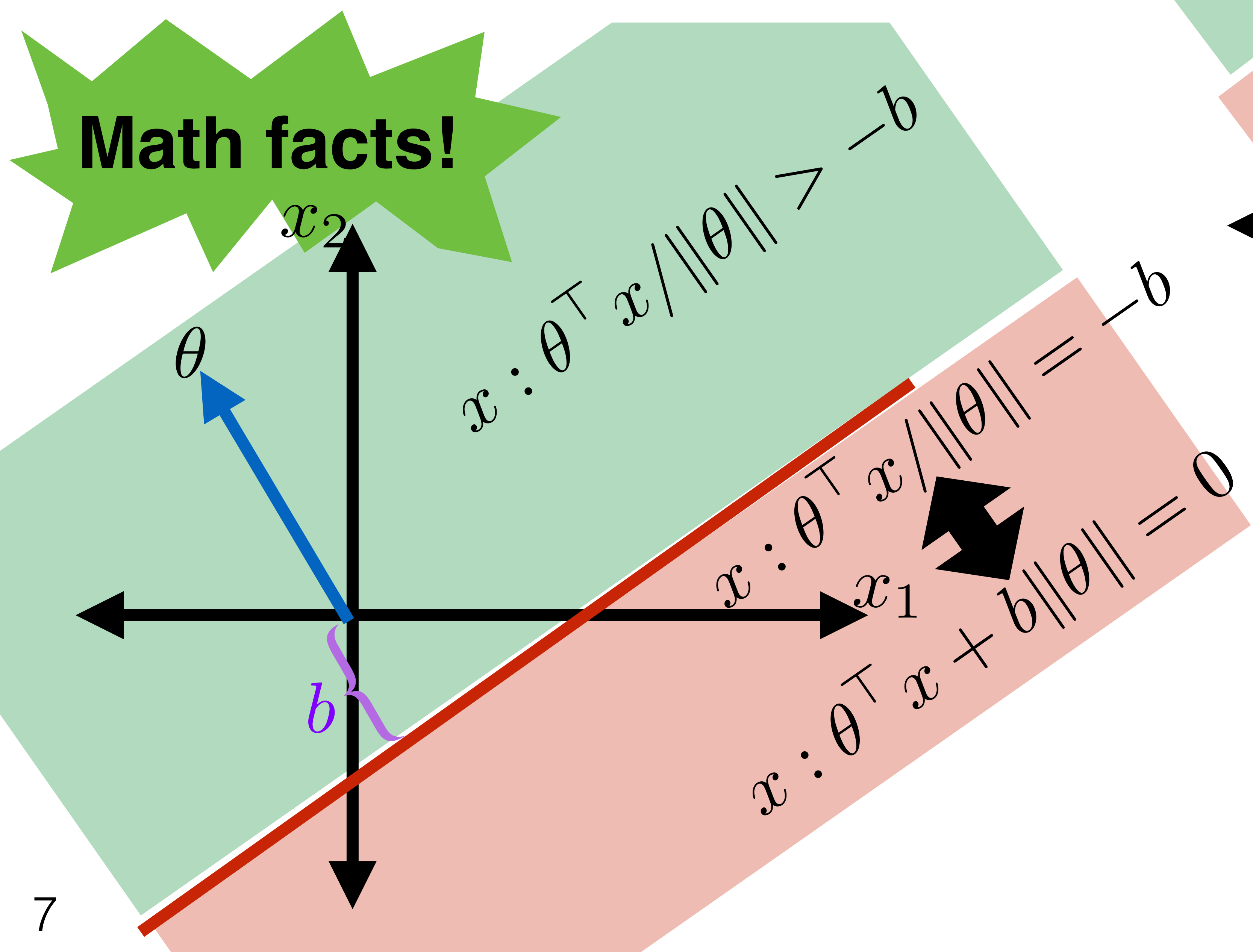
Math facts!



Linear classifiers

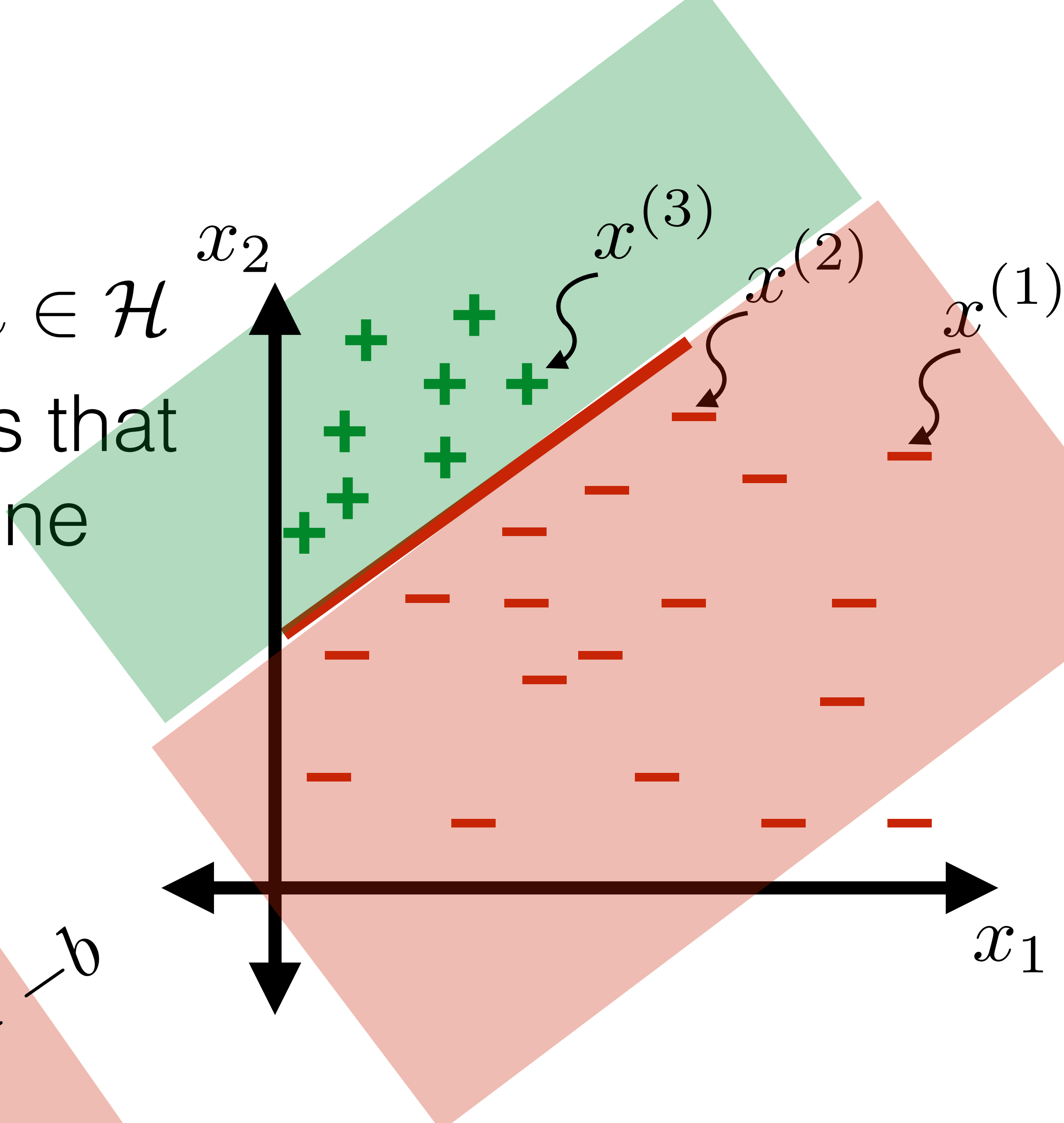
- Hypothesis class $\mathcal{H}$: set of $h \in \mathcal{H}$
    - Example $\mathcal{H}$: All hypotheses that label +1 on one side of a line and -1 on the other side

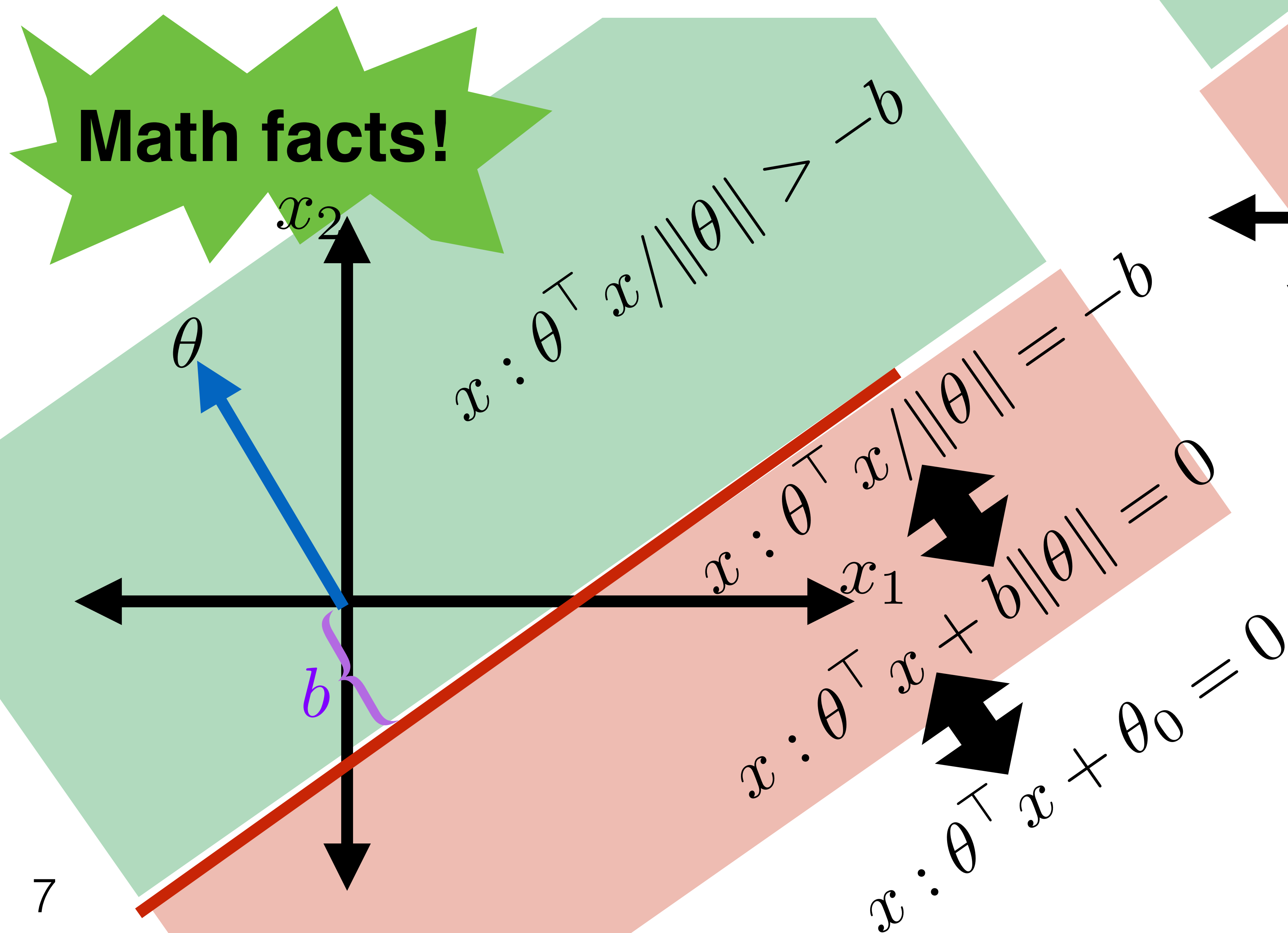




Linear classifiers

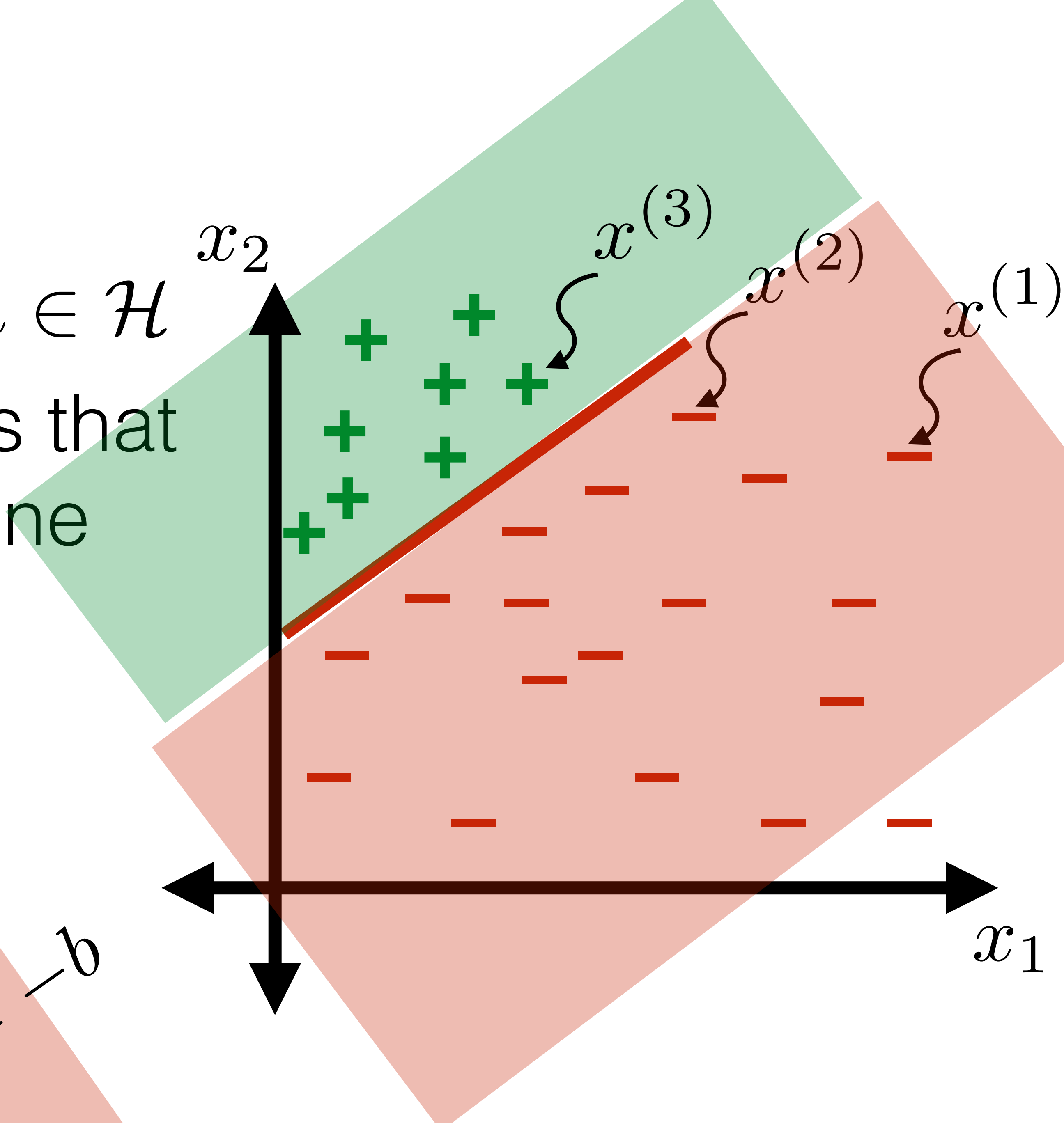
- Hypothesis class $\mathcal{H}$: set of $h \in \mathcal{H}$
  - Example $\mathcal{H}$: All hypotheses that label +1 on one side of a line and -1 on the other side

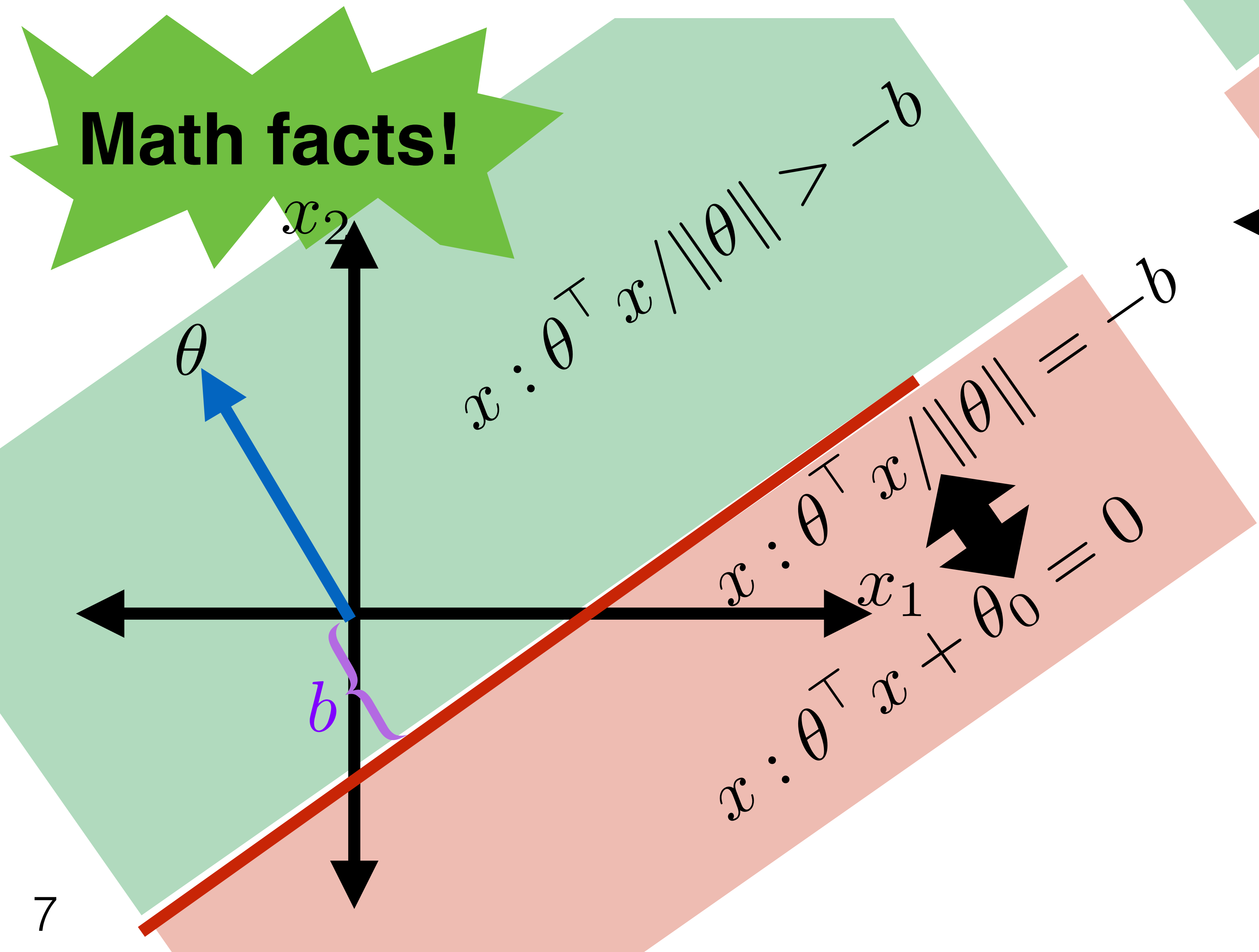




Linear classifiers

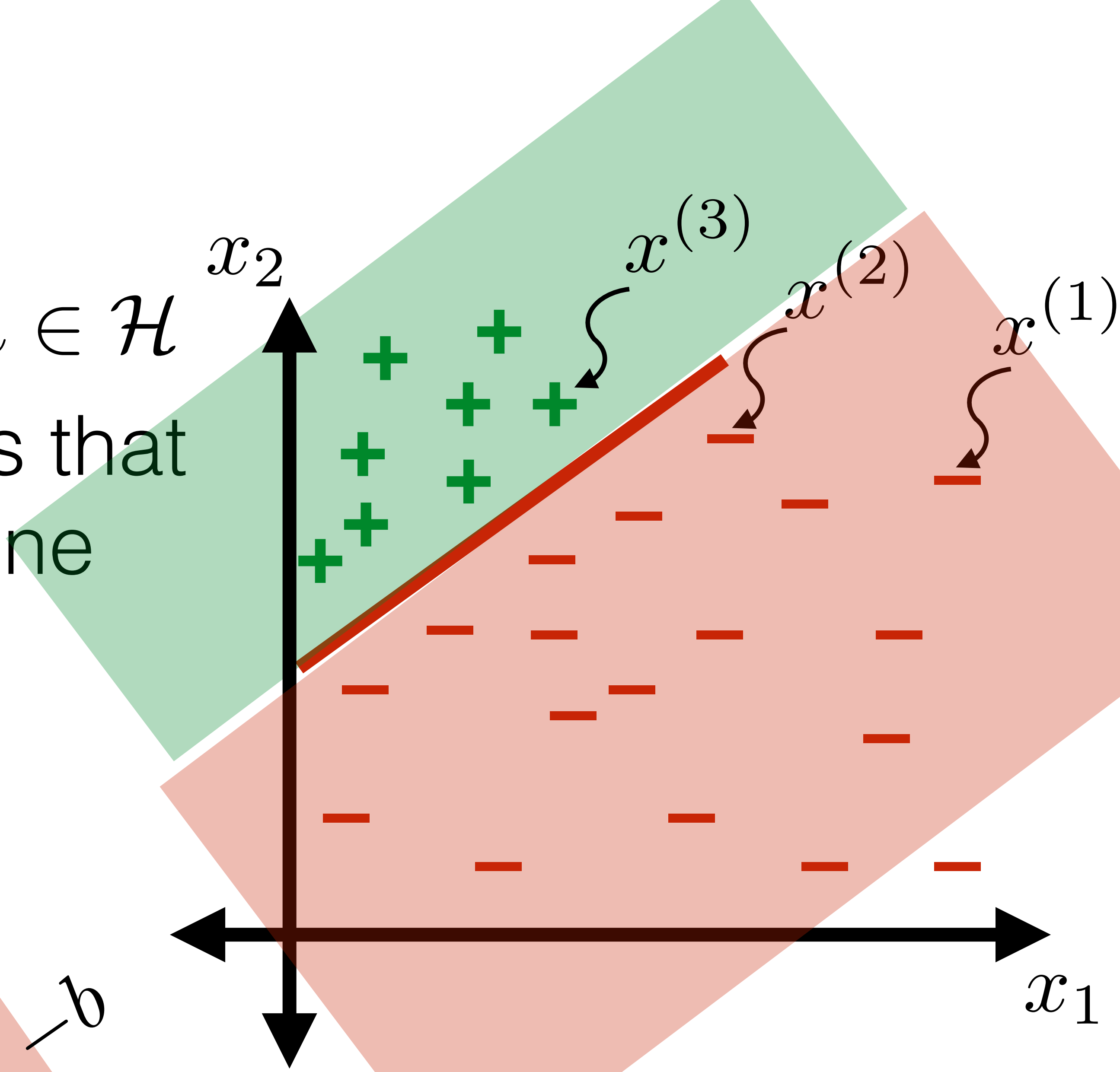
- Hypothesis class $\mathcal{H}$: set of $h \in \mathcal{H}$
  - Example $\mathcal{H}$: All hypotheses that label +1 on one side of a line and -1 on the other side

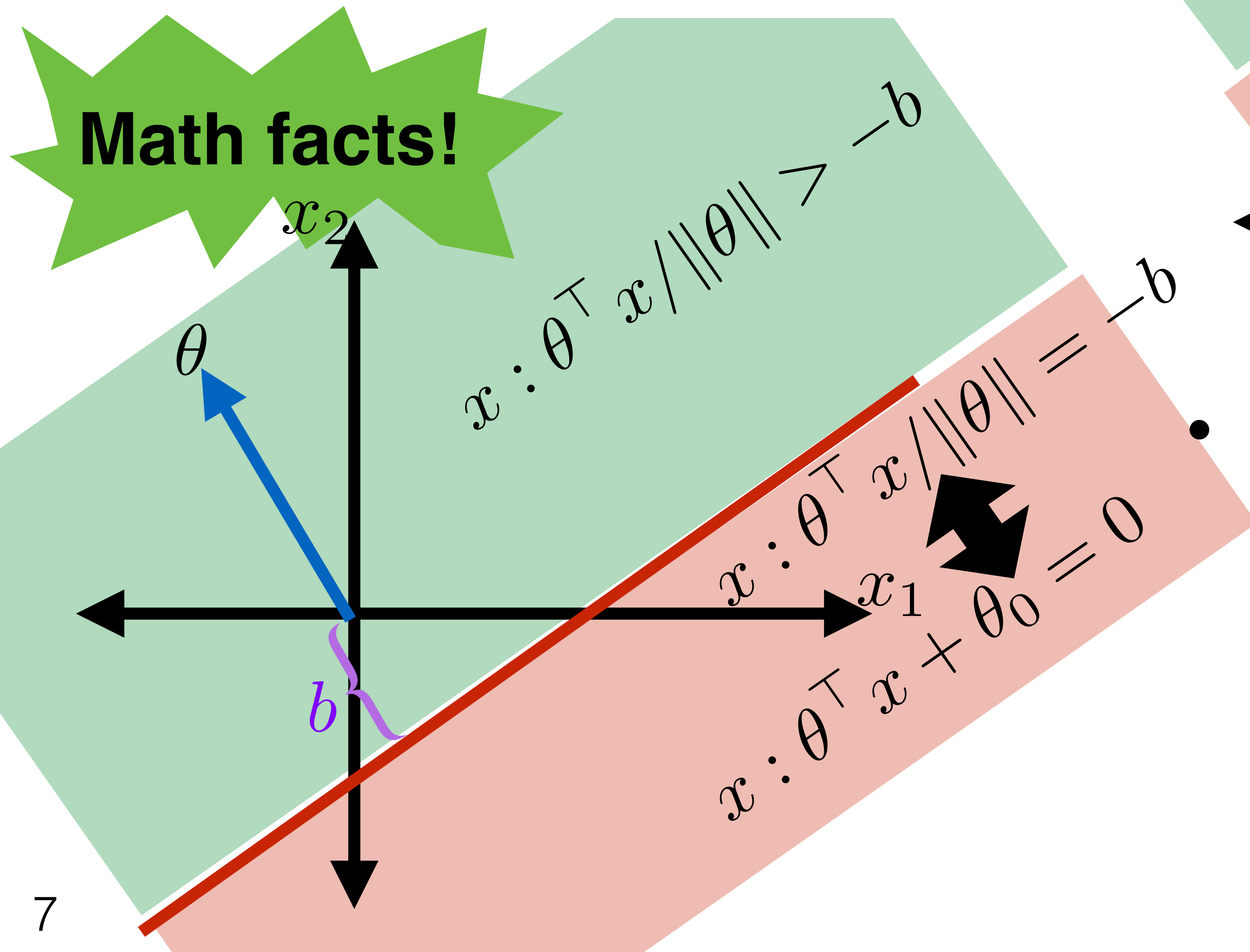




Linear classifiers

- Hypothesis class $\mathcal{H}$: set of $h \in \mathcal{H}$
  - Example $\mathcal{H}$: All hypotheses that label +1 on one side of a line and -1 on the other side

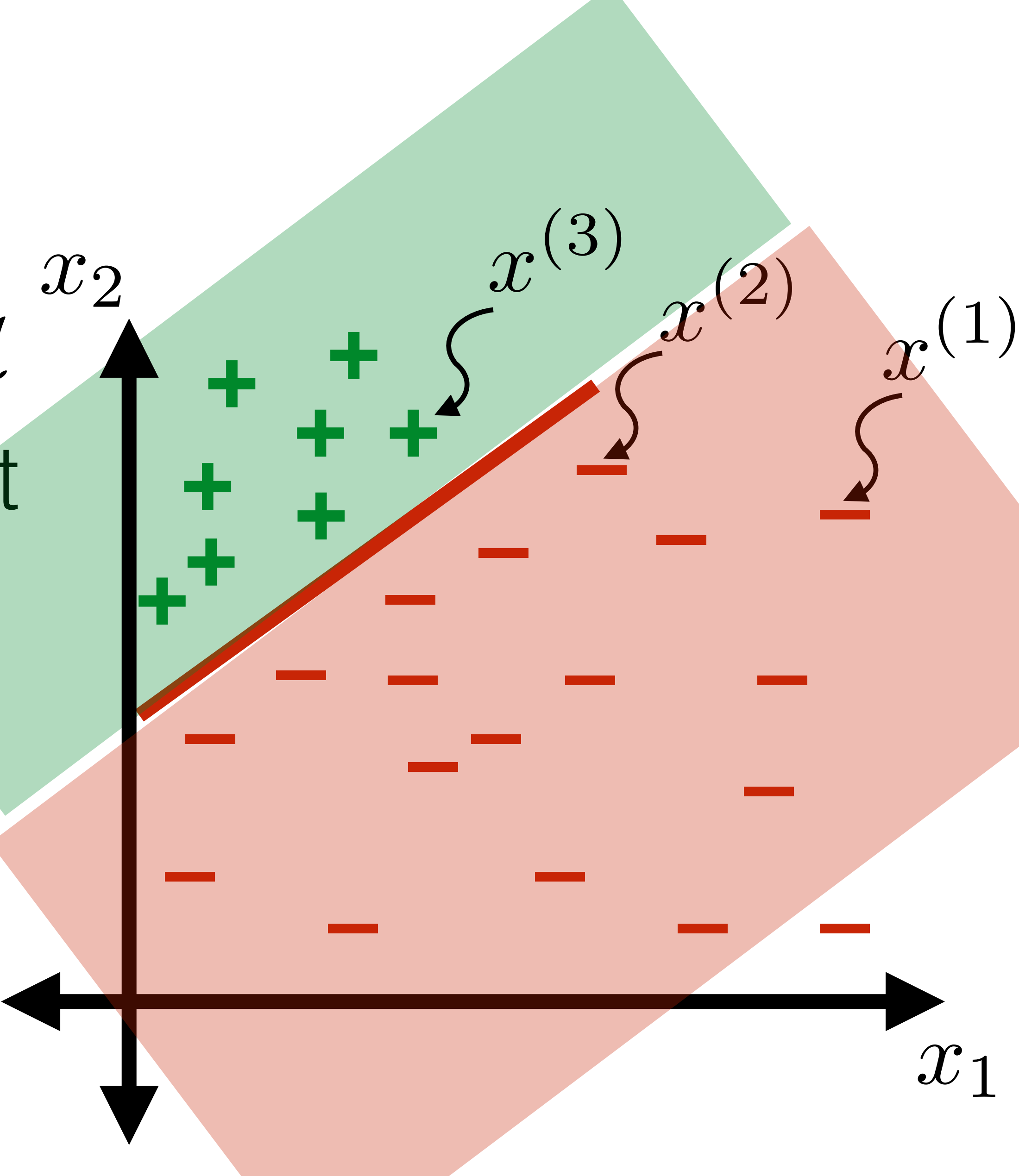




- Linear classifier:

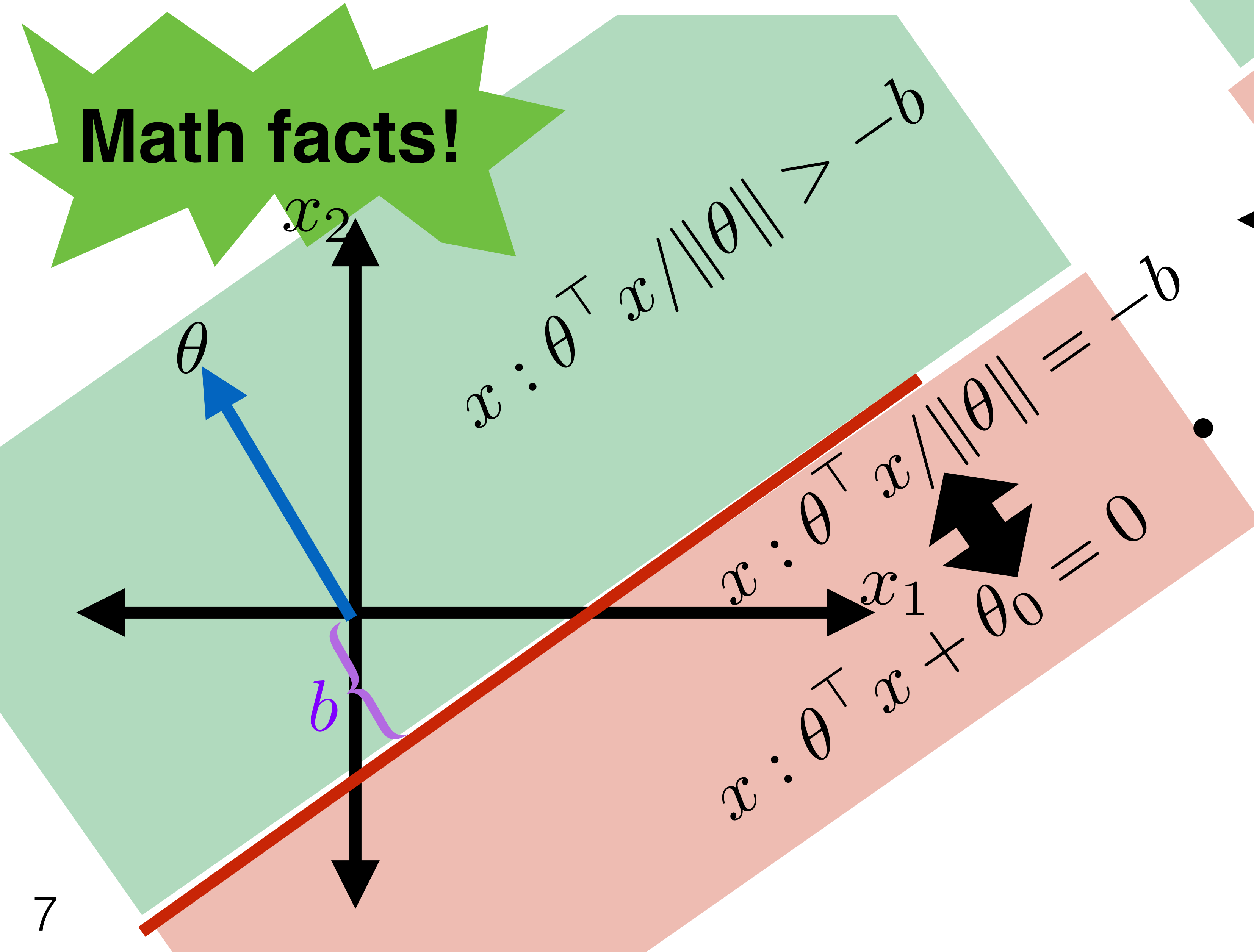
Linear classifiers

- Hypothesis class $\mathcal{H}$: set of $h \in \mathcal{H}$
  - Example $\mathcal{H}$: All hypotheses that label +1 on one side of a line and -1 on the other side



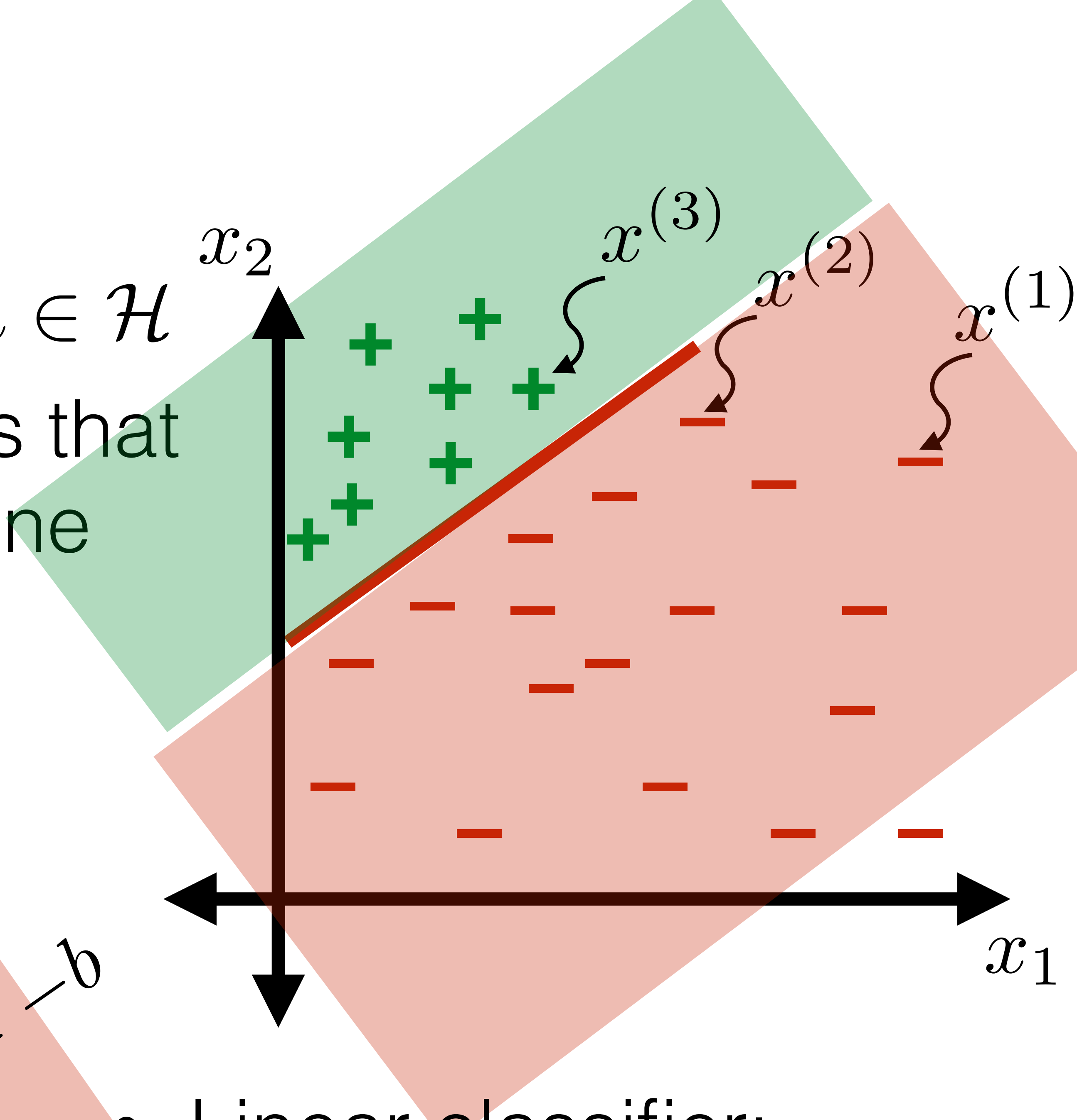
- Linear classifier:

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

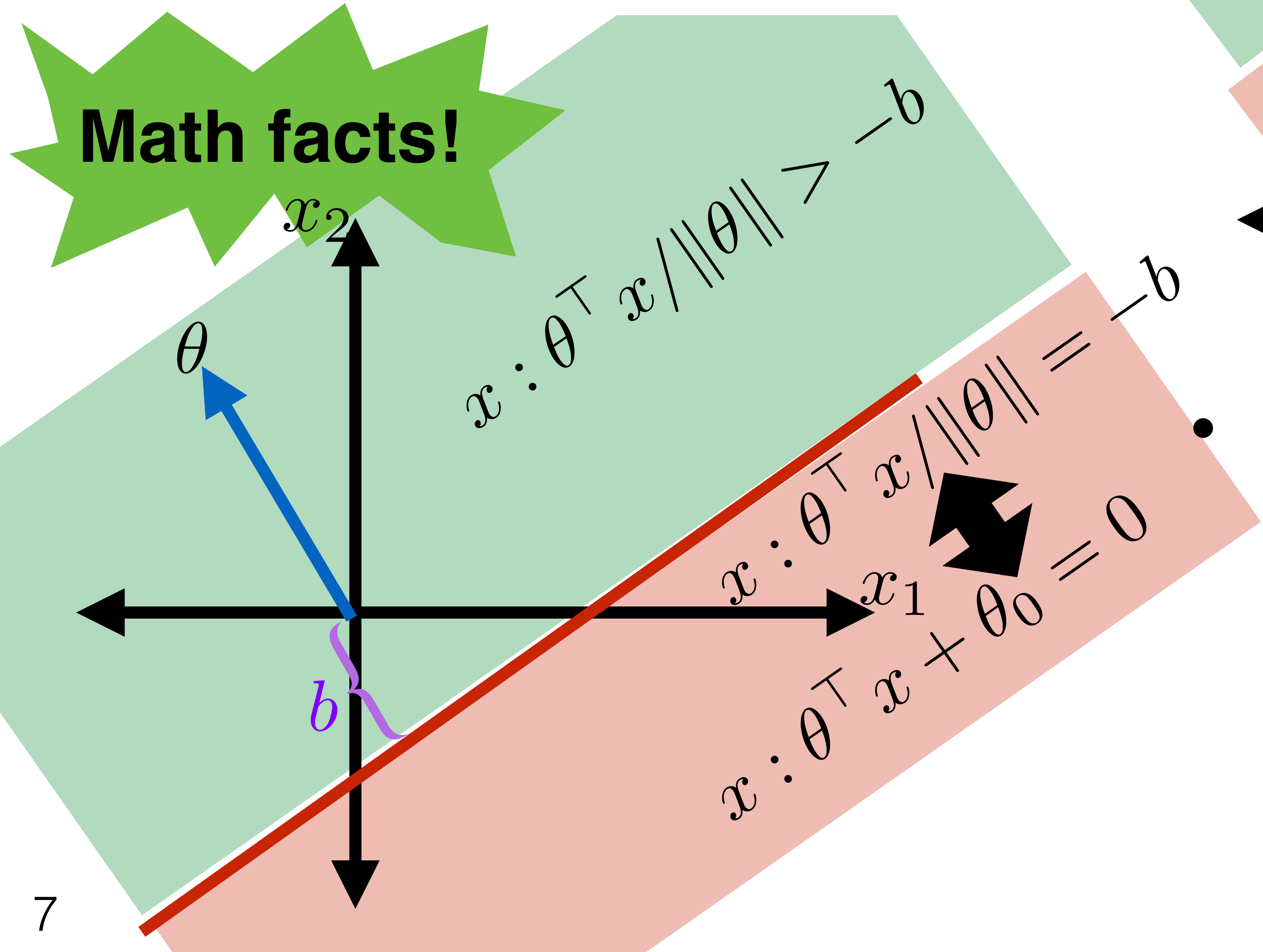


Linear classifiers

- Hypothesis class $\mathcal{H}$: set of $h \in \mathcal{H}$
  - Example $\mathcal{H}$: All hypotheses that label +1 on one side of a line and -1 on the other side



Math facts!



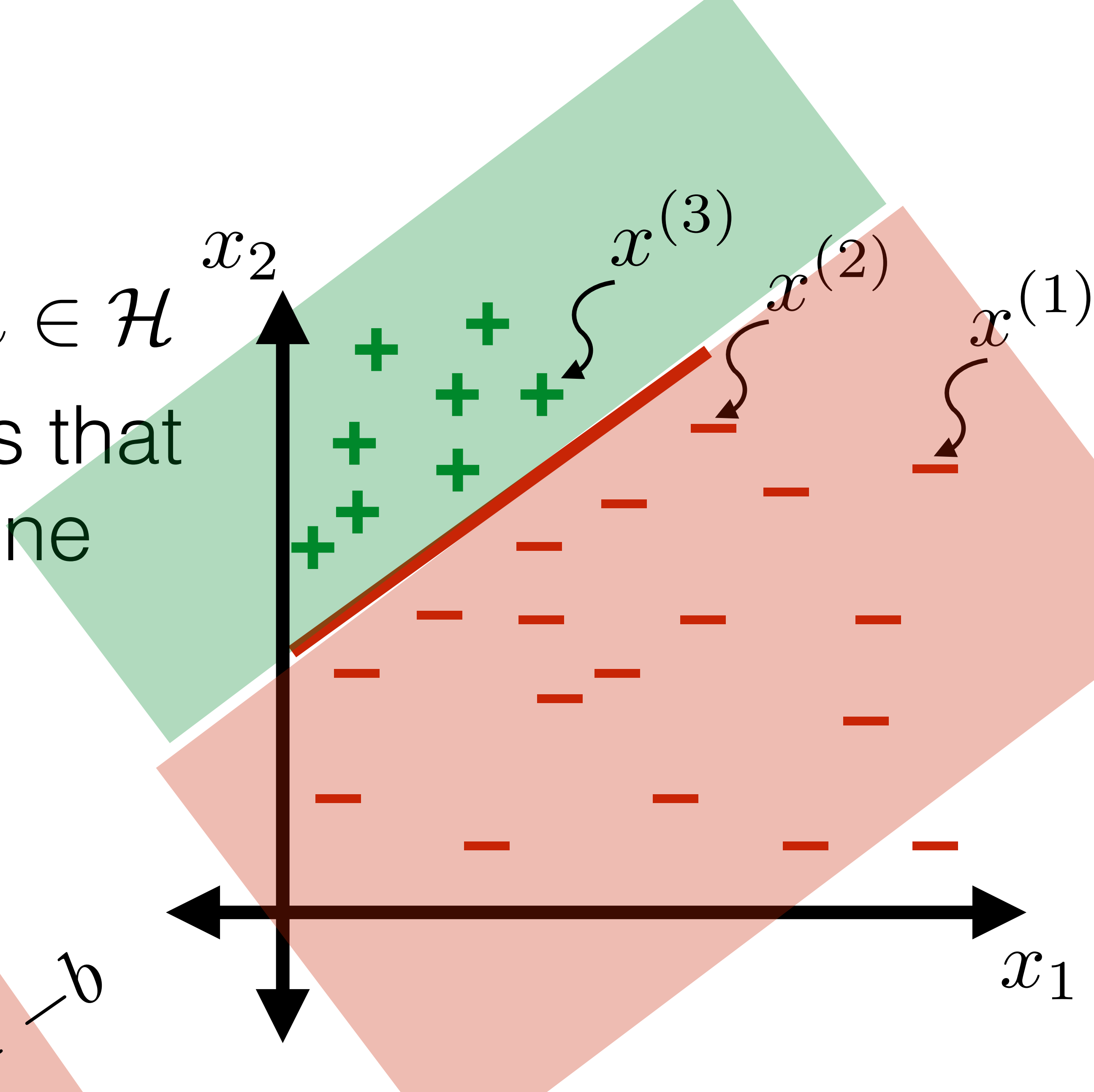
- Linear classifier:

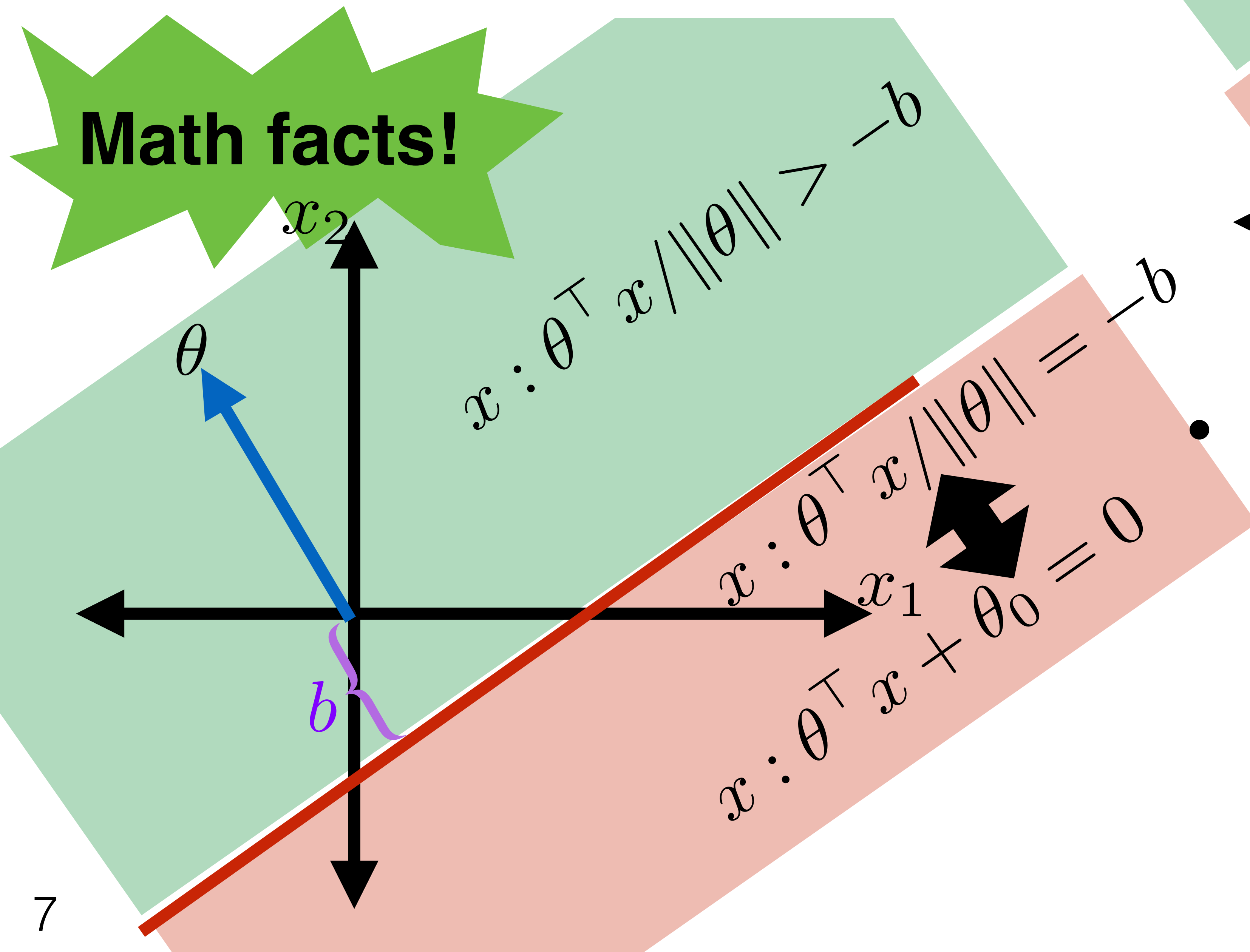
$$h(x) = \text{sign}(\theta^\top x + \theta_0)$$

$$= \begin{cases} +1 \text{ if } \theta^\top x + \theta_0 > 0 \\ -1 \text{ if } \theta^\top x + \theta_0 < 0 \end{cases}$$

Linear classifiers

- Hypothesis class $\mathcal{H}$: set of $h \in \mathcal{H}$
  - Example $\mathcal{H}$: All hypotheses that label +1 on one side of a line and -1 on the other side





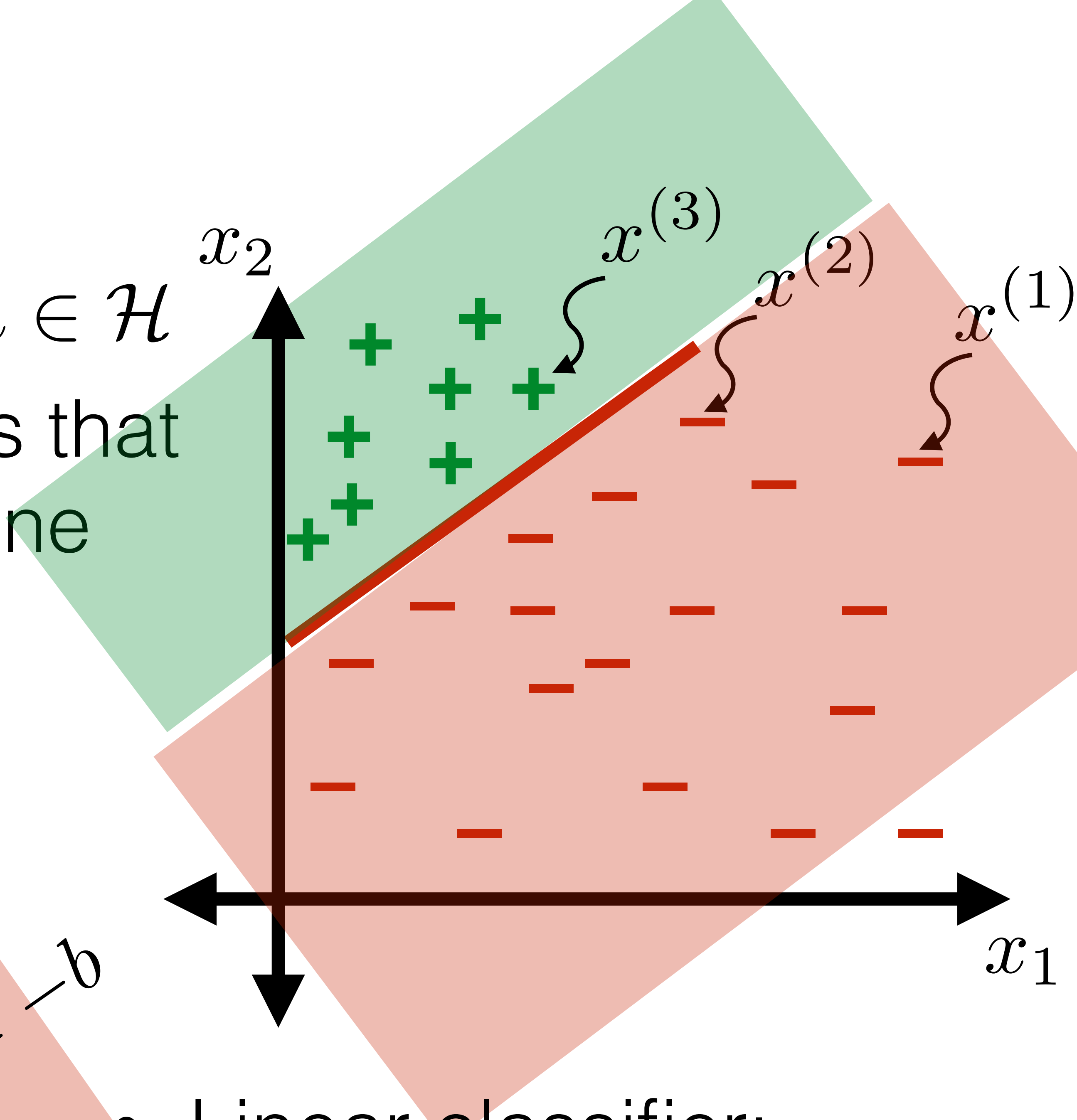
- Linear classifier:

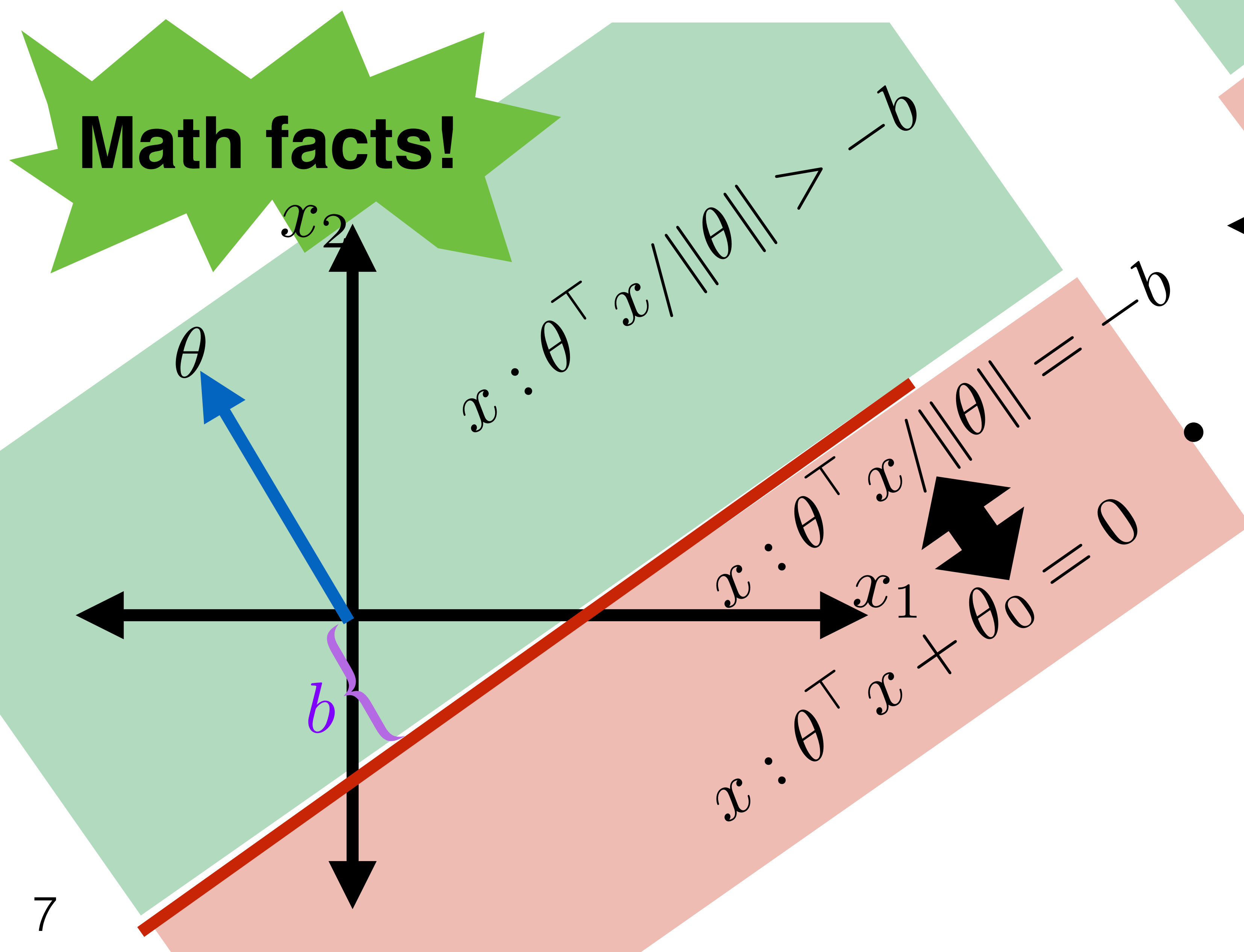
$$h(x) = \text{sign}(\theta^\top x + \theta_0)$$

$$= \begin{cases} +1 \text{ if } \theta^\top x + \theta_0 > 0 \\ -1 \text{ if } \theta^\top x + \theta_0 < 0 \end{cases}$$

Linear classifiers

- Hypothesis class $\mathcal{H}$: set of $h \in \mathcal{H}$
  - Example $\mathcal{H}$: All hypotheses that label +1 on one side of a line and -1 on the other side





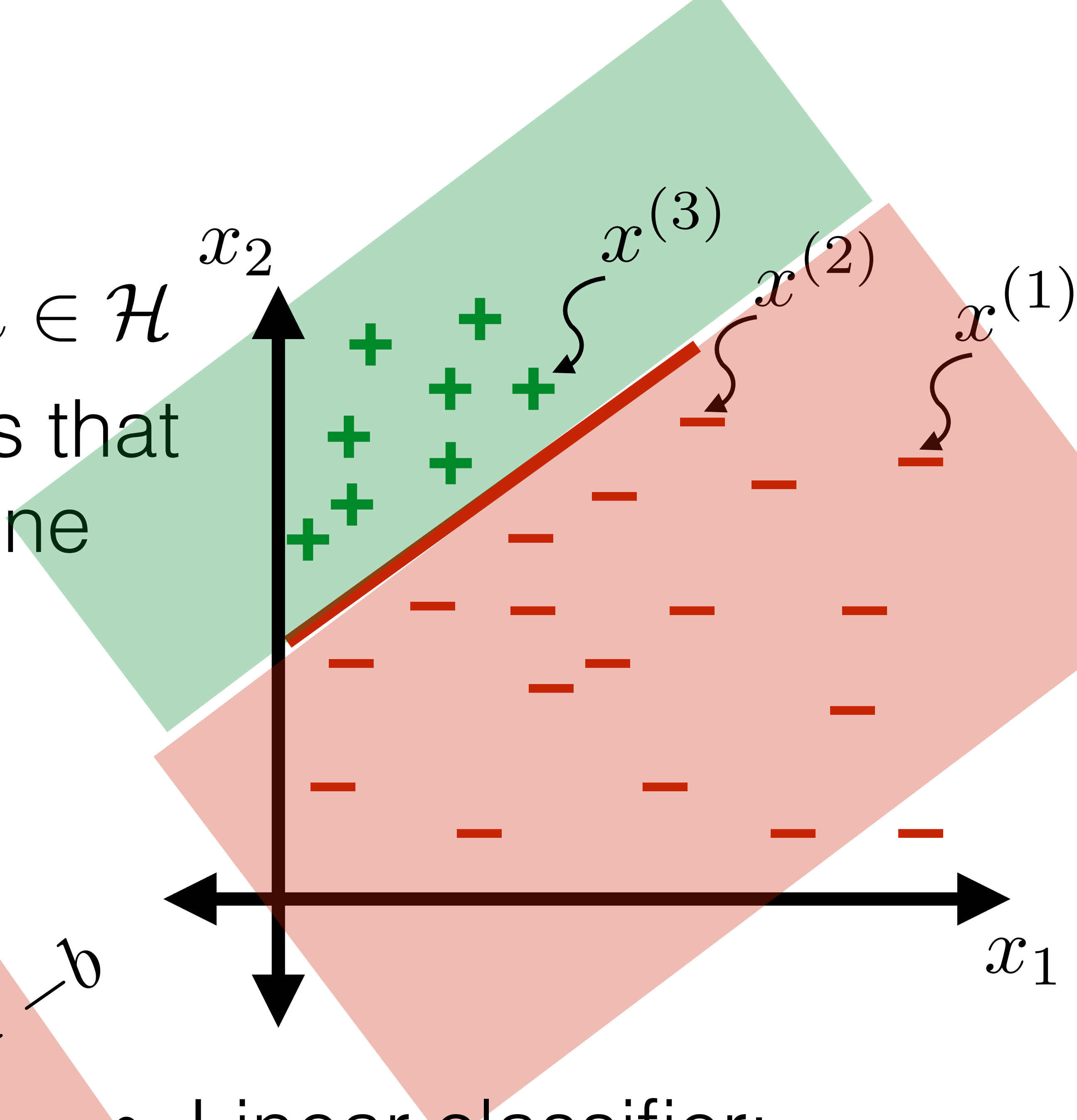
- Linear classifier:

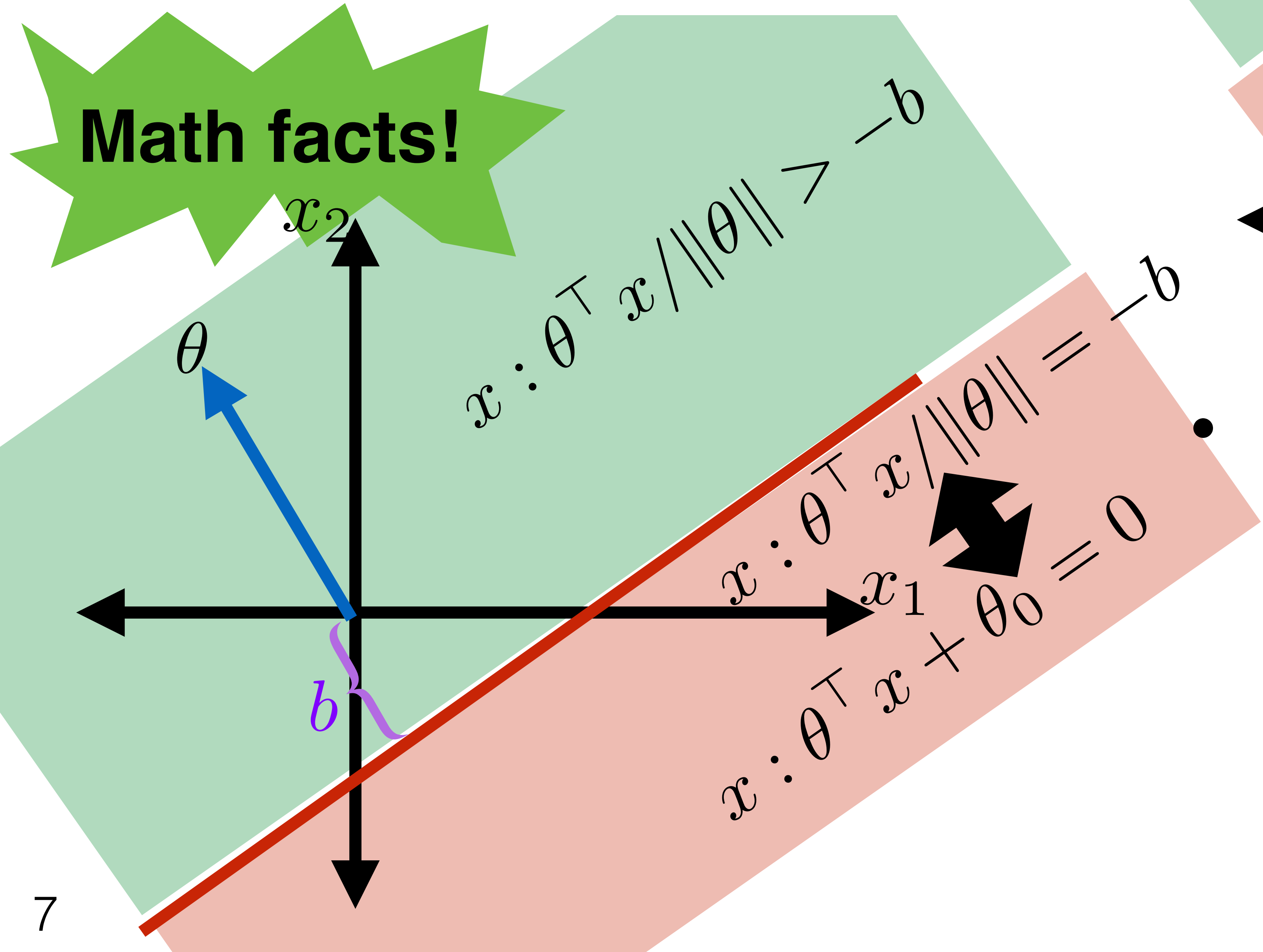
$$h(x) = \text{sign}(\theta^\top x + \theta_0)$$

$$= \begin{cases} +1 \text{ if } \theta^\top x + \theta_0 > 0 \\ -1 \text{ if } \theta^\top x + \theta_0 \leq 0 \end{cases}$$

Linear classifiers

- Hypothesis class $\mathcal{H}$: set of $h \in \mathcal{H}$
    - Example $\mathcal{H}$: All hypotheses that label +1 on one side of a line and -1 on the other side





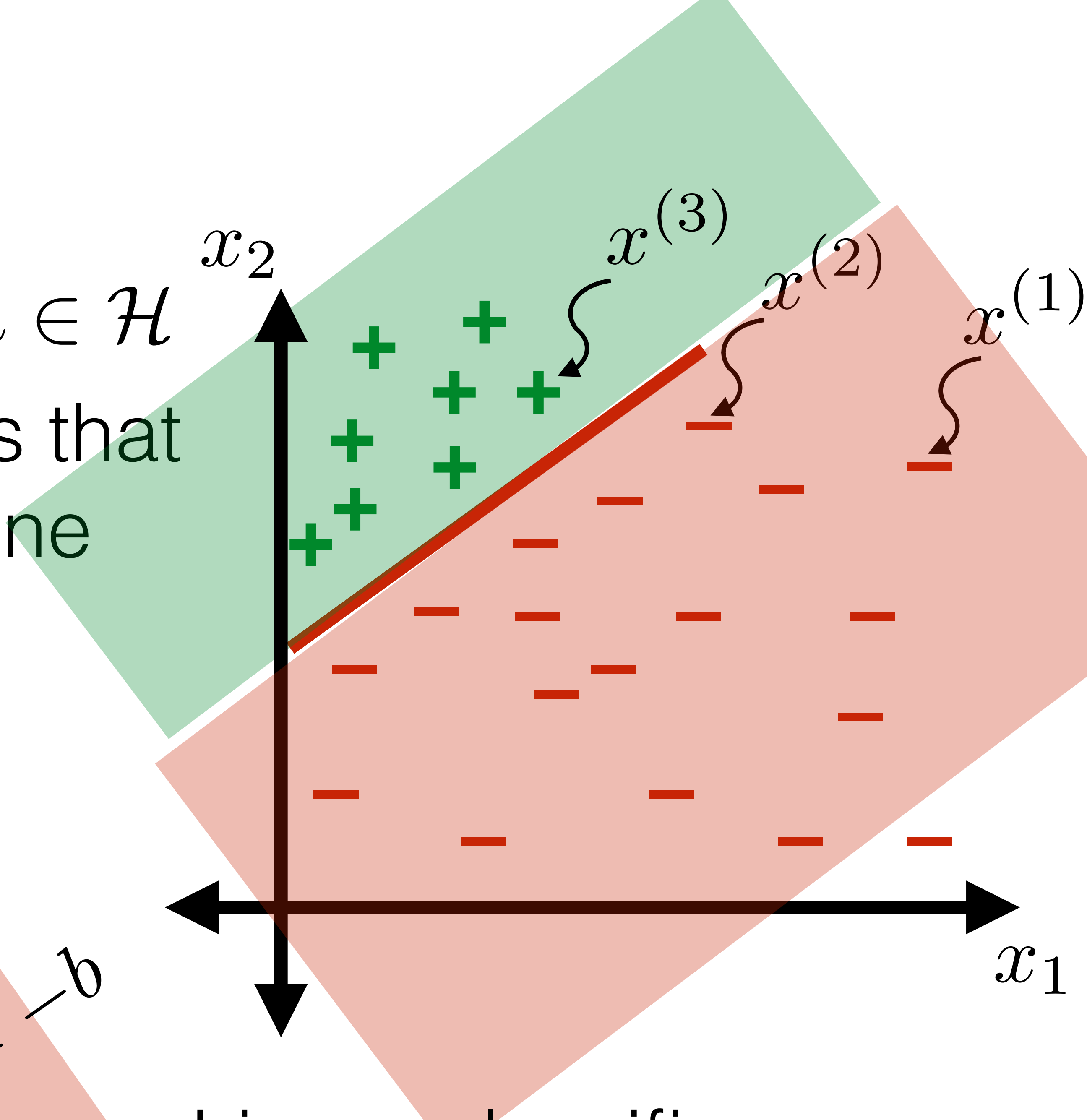
- Linear classifier:

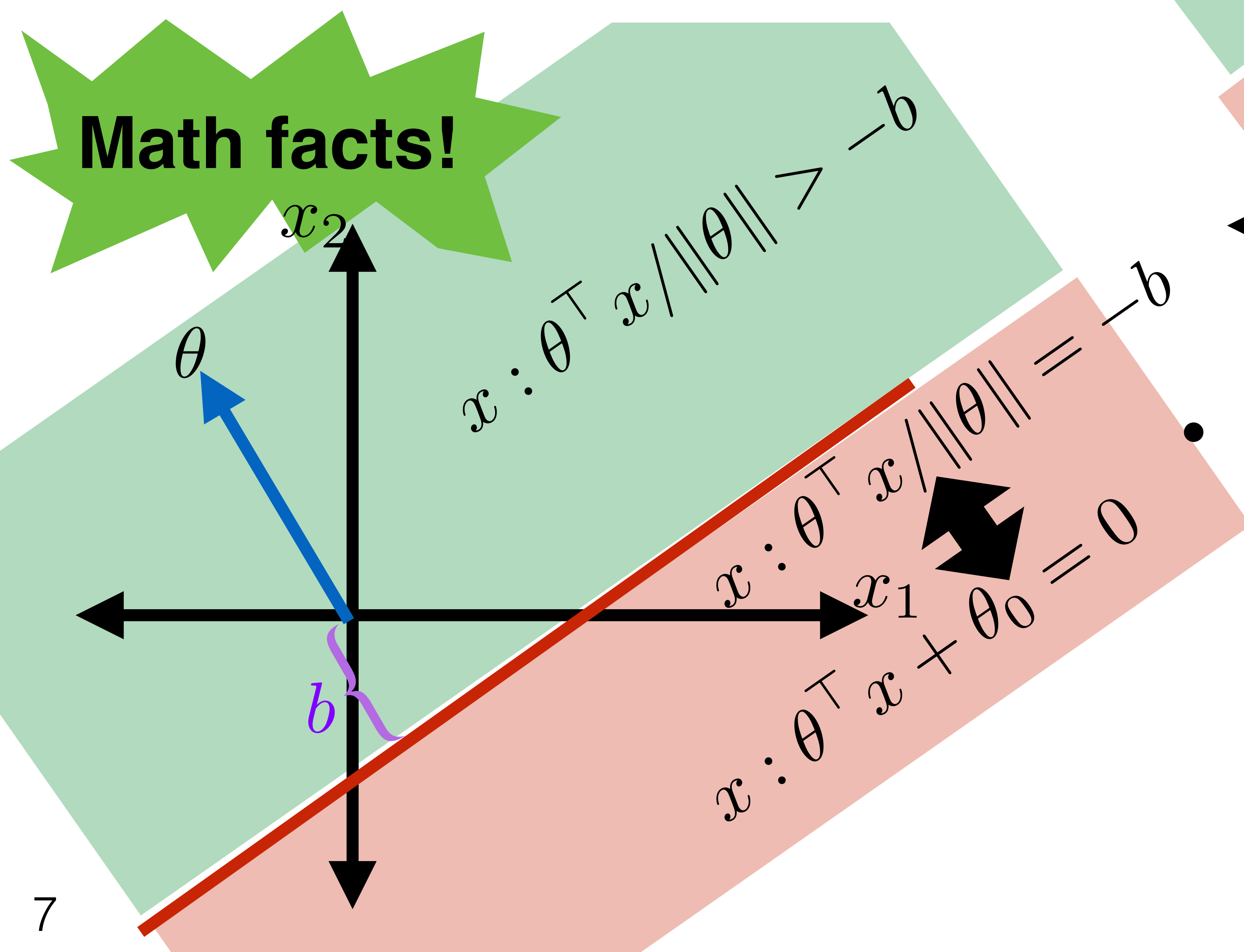
$$h(x) = \text{sign}(\theta^\top x + \theta_0)$$

$$= \begin{cases} +1 \text{ if } \theta^\top x + \theta_0 > 0 \\ -1 \text{ if } \theta^\top x + \theta_0 \leq 0 \end{cases}$$

Linear classifiers

- Hypothesis class $\mathcal{H}$: set of $h \in \mathcal{H}$
  - Example $\mathcal{H}$: All hypotheses that label +1 on one side of a line and -1 on the other side





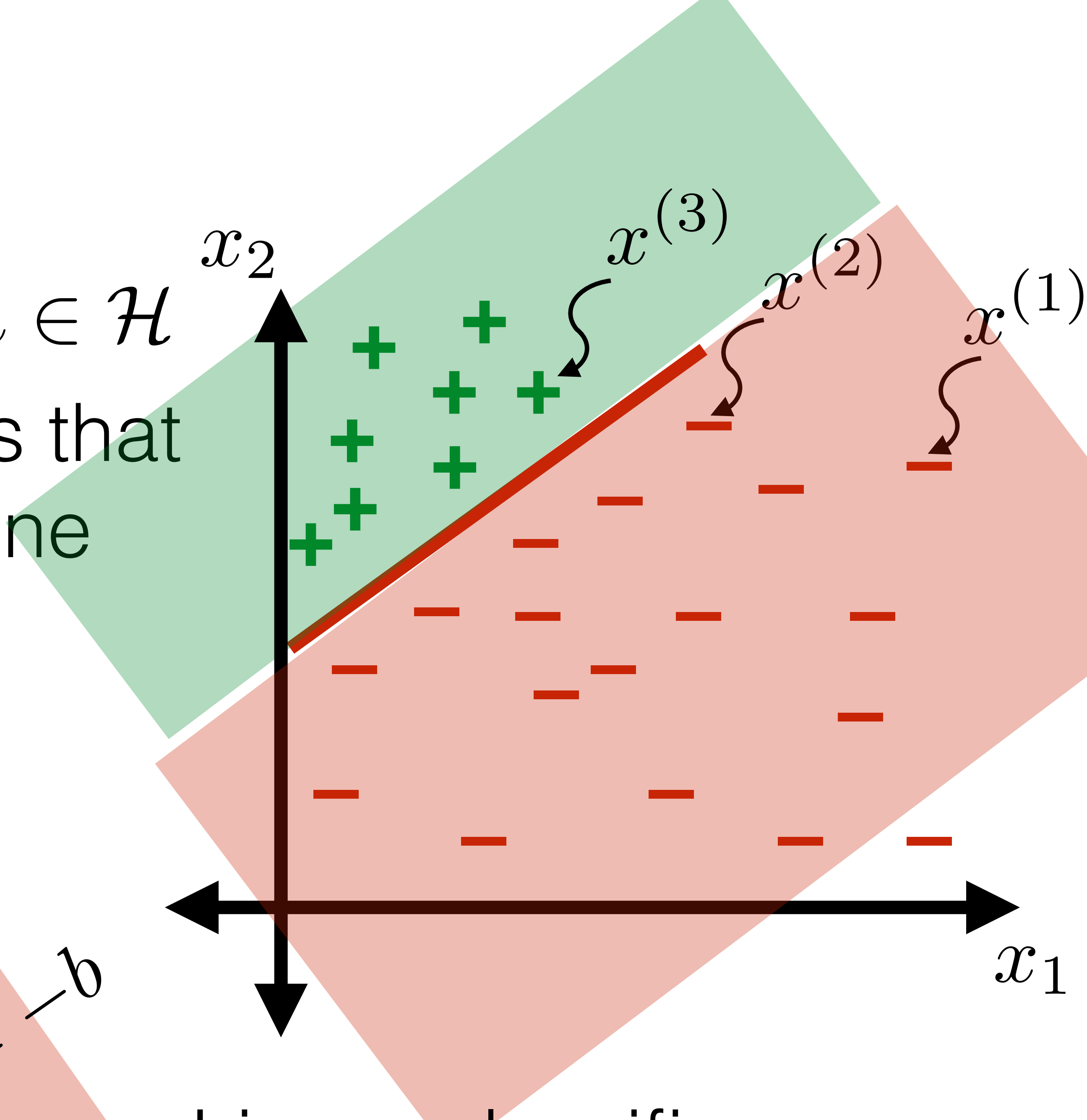
- Linear classifier:

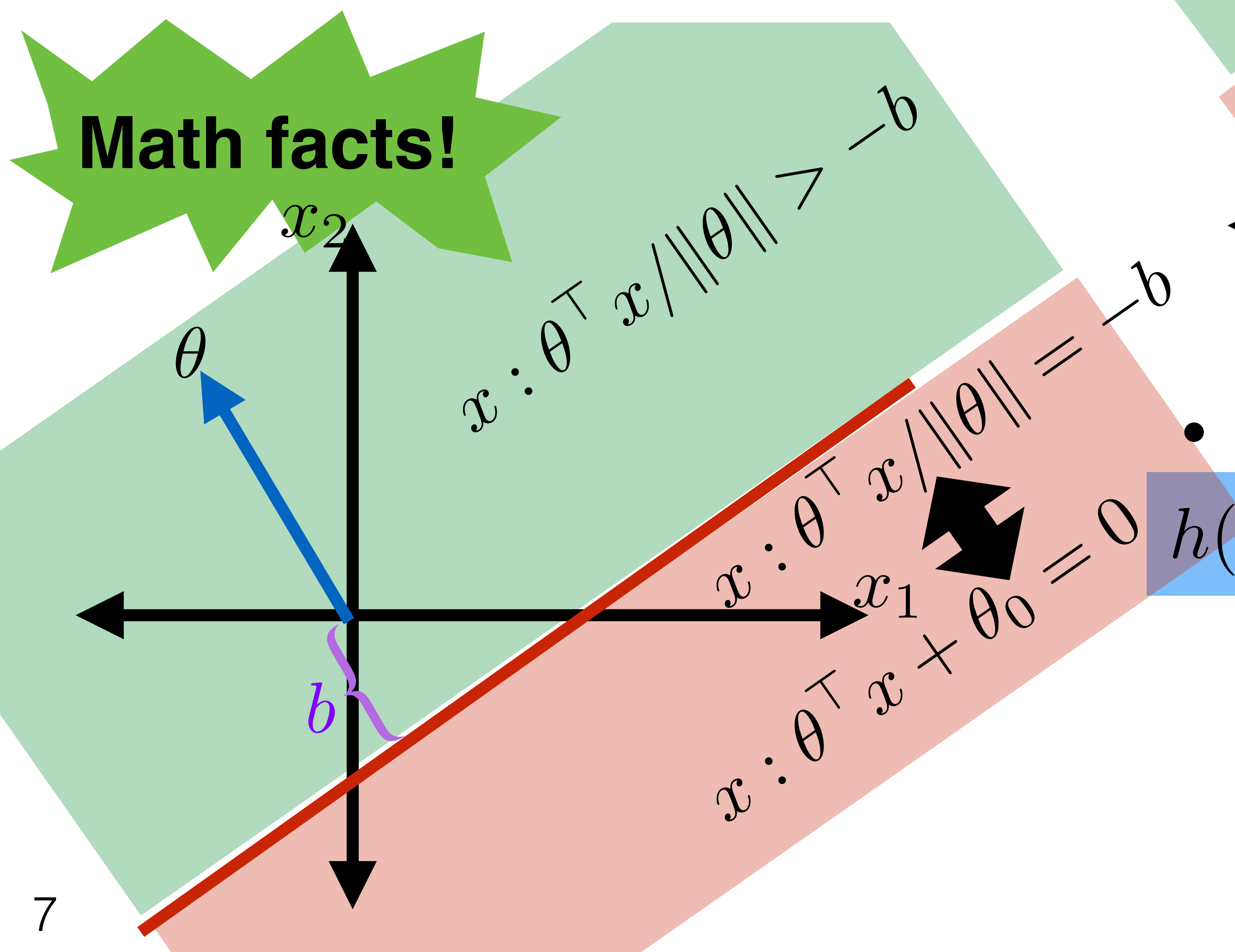
$$\boxed{h(x)} = \text{sign}(\theta^\top x + \theta_0)$$

$$= \begin{cases} +1 \text{ if } \theta^\top x + \theta_0 > 0 \\ -1 \text{ if } \theta^\top x + \theta_0 \leq 0 \end{cases}$$

Linear classifiers

- Hypothesis class $\mathcal{H}$: set of $h \in \mathcal{H}$
  - Example $\mathcal{H}$: All hypotheses that label +1 on one side of a line and -1 on the other side





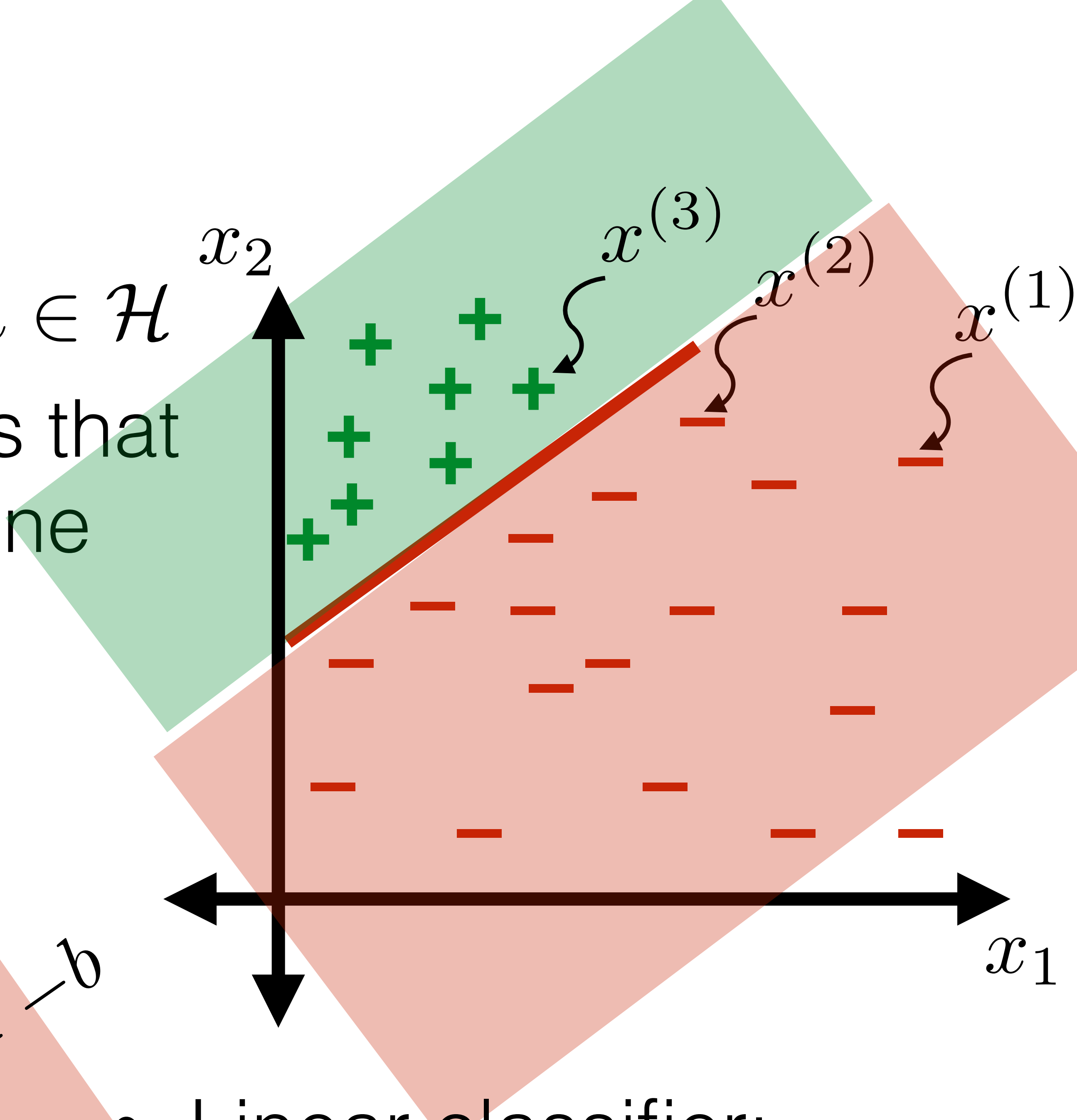
- Linear classifier:

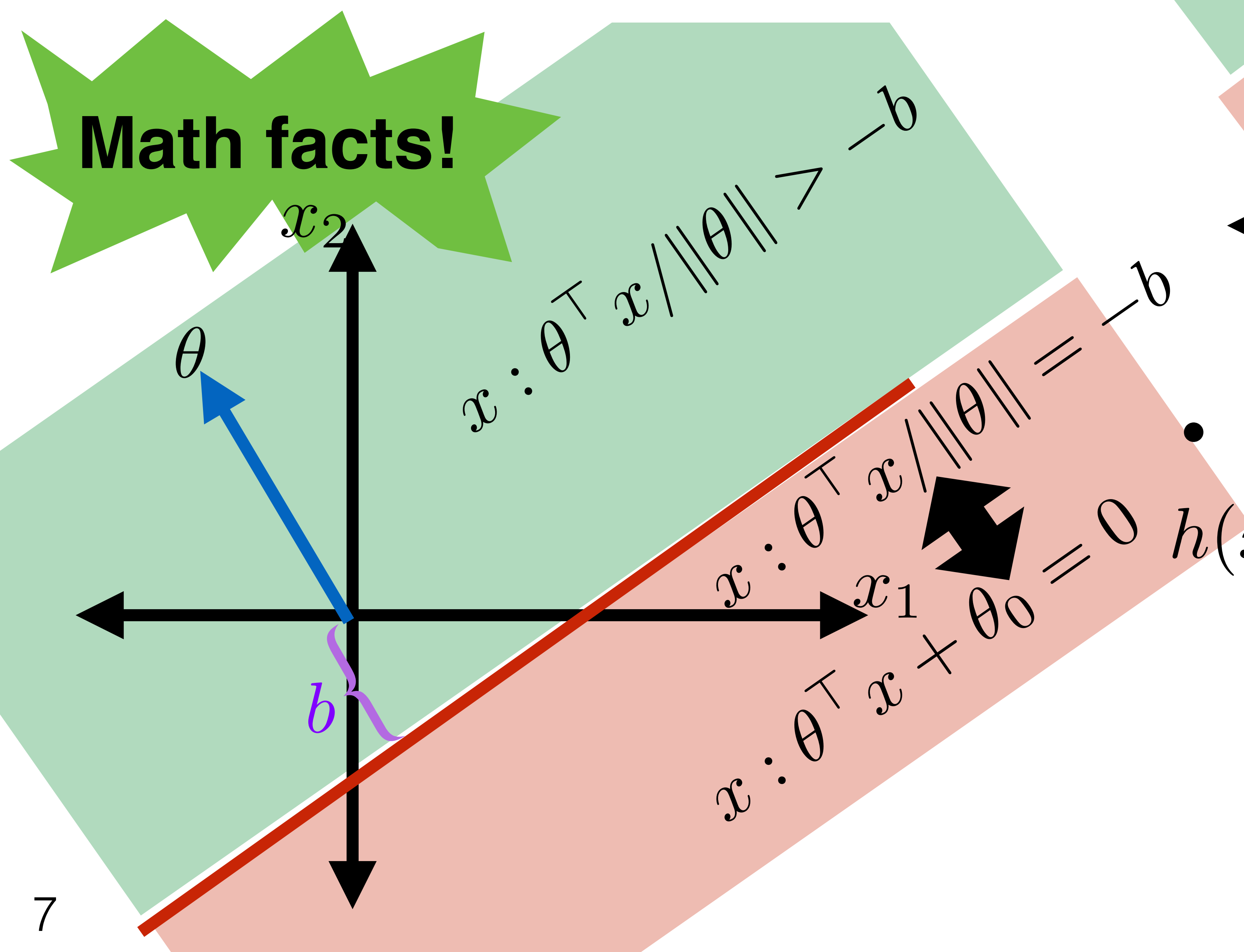
$$\boxed{h(x;\theta,\theta_0)} = \text{sign}(\theta^\top x + \theta_0)$$

$$= \begin{cases} +1 \text{ if } \theta^\top x + \theta_0 > 0 \\ -1 \text{ if } \theta^\top x + \theta_0 \leq 0 \end{cases}$$

Linear classifiers

- Hypothesis class $\mathcal{H}$: set of $h \in \mathcal{H}$
  - Example $\mathcal{H}$: All hypotheses that label +1 on one side of a line and -1 on the other side





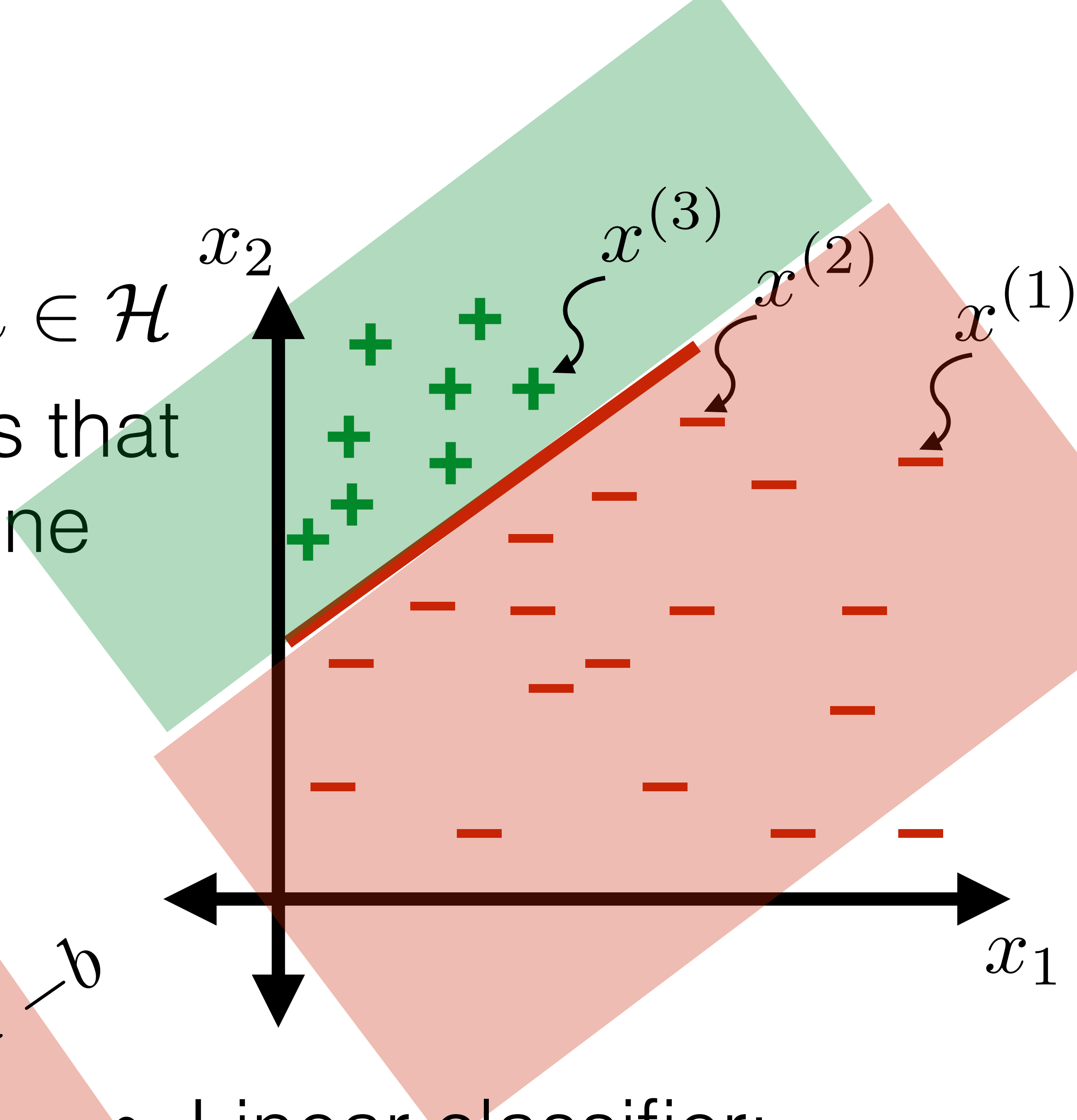
- Linear classifier:

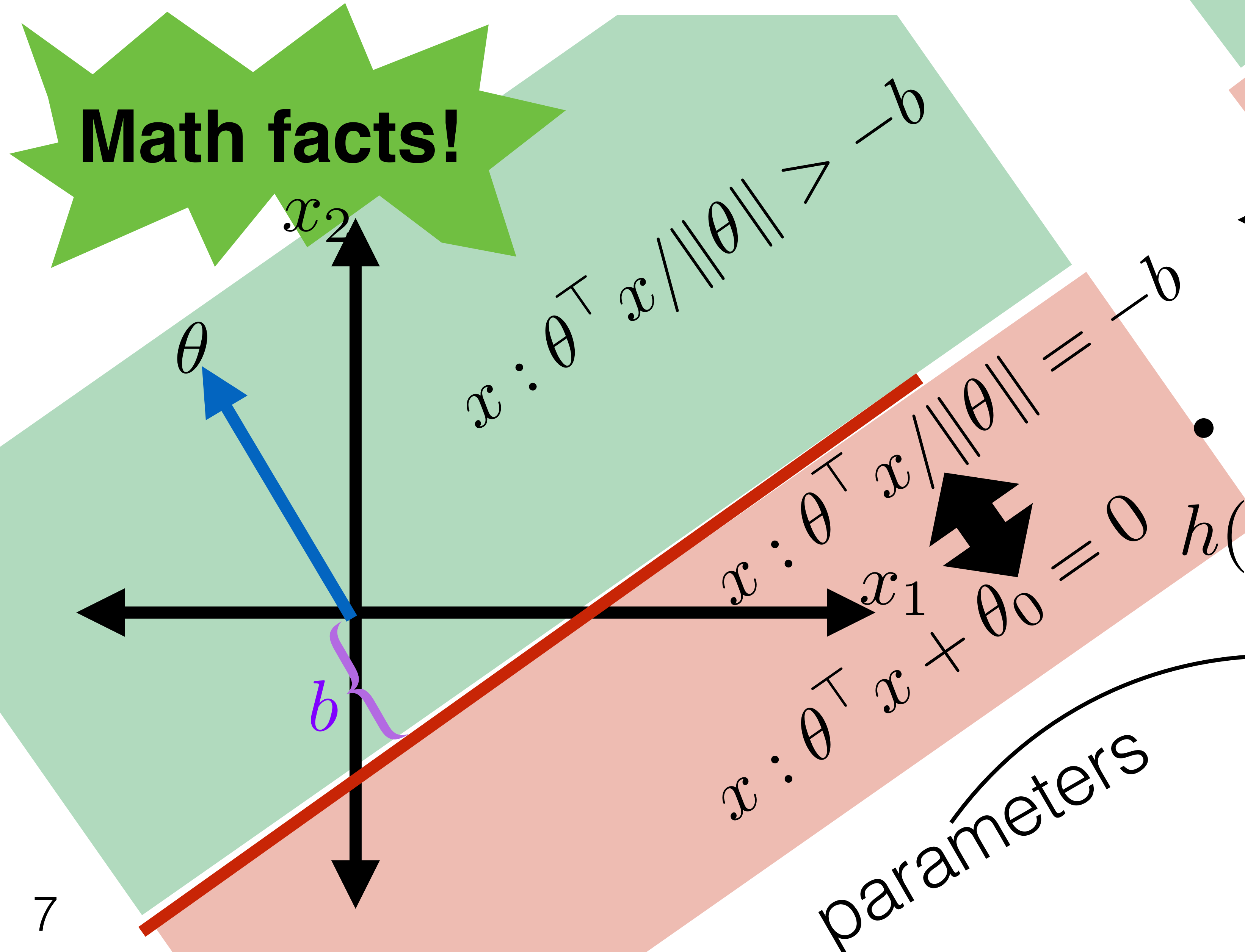
$$h(x; \theta, \theta_0) = \text{sign}(\theta^\top x + \theta_0)$$

$$= \begin{cases} +1 \text{ if } \theta^\top x + \theta_0 > 0 \\ -1 \text{ if } \theta^\top x + \theta_0 \leq 0 \end{cases}$$

Linear classifiers

- Hypothesis class $\mathcal{H}$: set of $h \in \mathcal{H}$
  - Example $\mathcal{H}$: All hypotheses that label +1 on one side of a line and -1 on the other side





- Linear classifier:

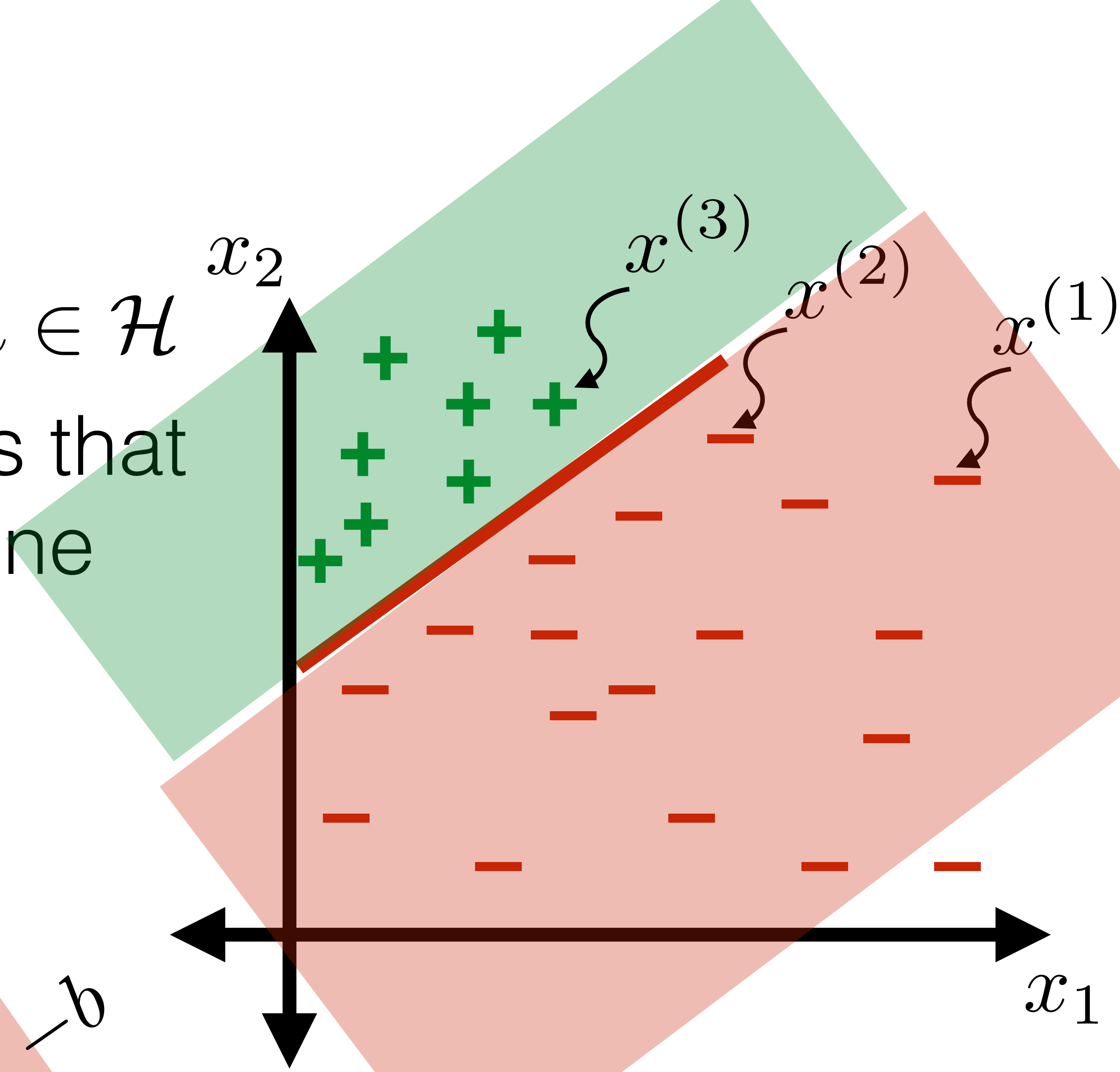
$$h(x; \theta, \theta_0) = \text{sign}(\theta^\top x + \theta_0)$$

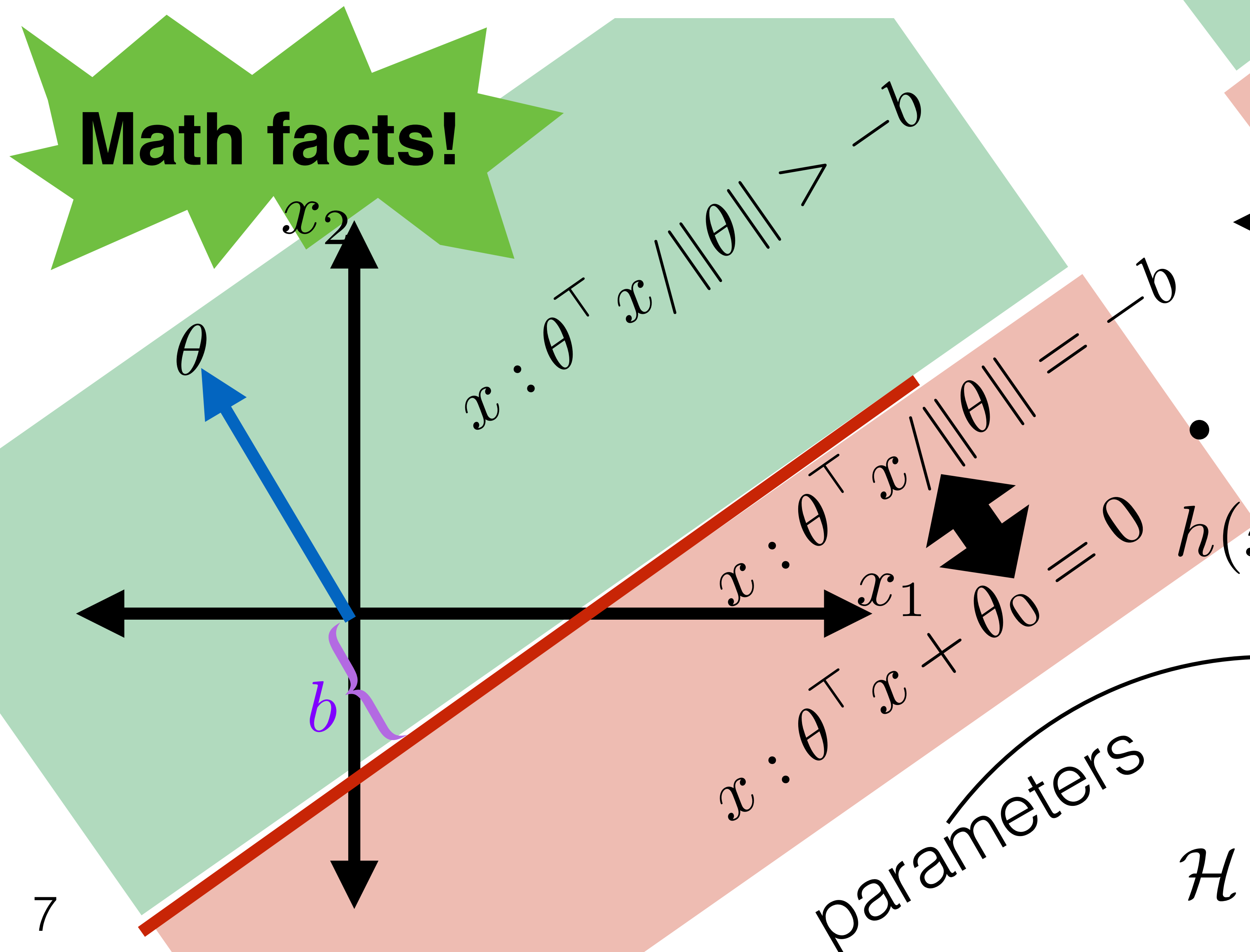
$$= \begin{cases} +1 \text{ if } \theta^\top x + \theta_0 > 0 \\ -1 \text{ if } \theta^\top x + \theta_0 \leq 0 \end{cases}$$

parameters

Linear classifiers

- Hypothesis class $\mathcal{H}$: set of $h \in \mathcal{H}$
  - Example $\mathcal{H}$: All hypotheses that label +1 on one side of a line and -1 on the other side





- Linear classifier:

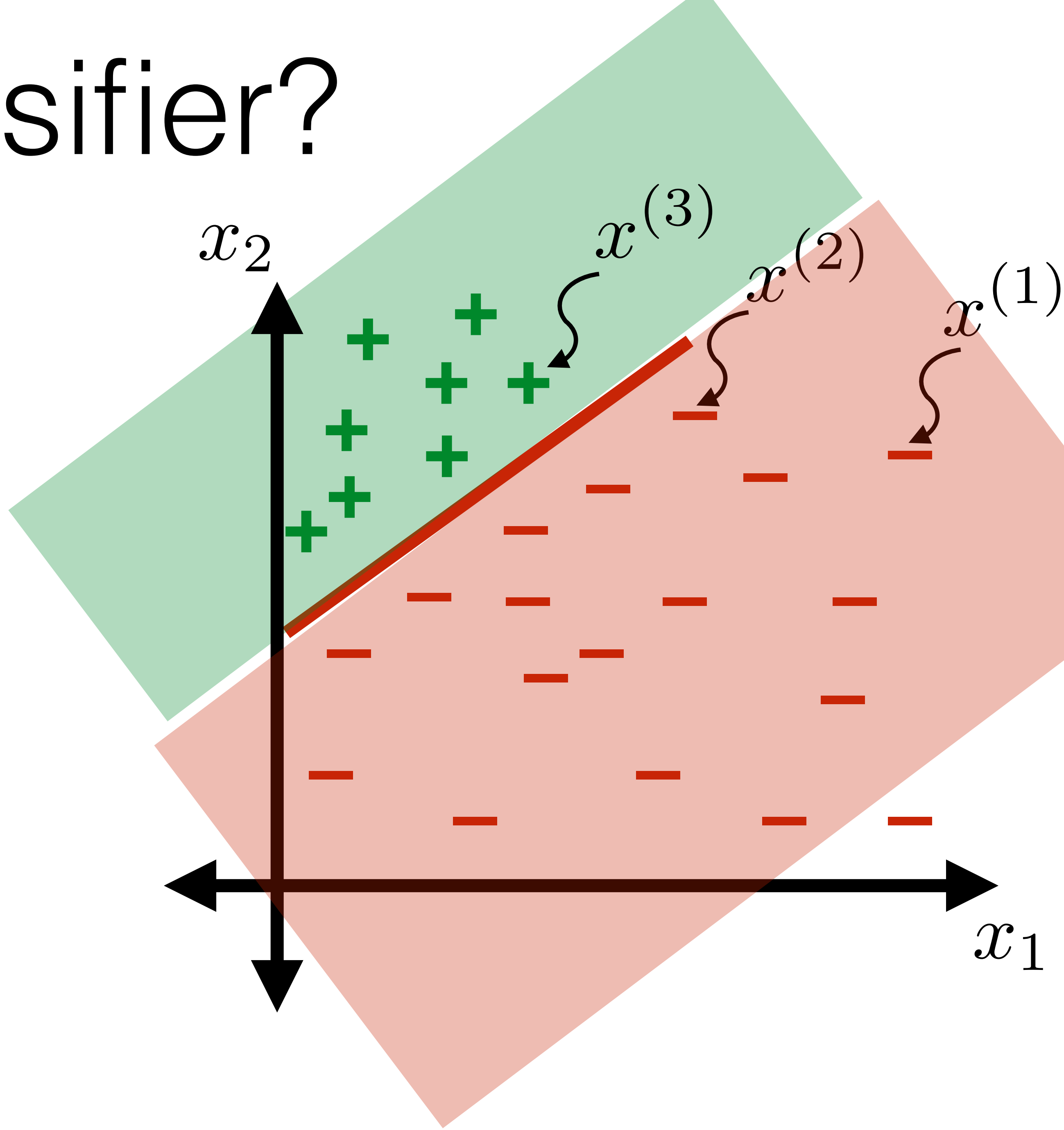
$$h(x; \theta, \theta_0) = \text{sign}(\theta^\top x + \theta_0)$$

$$= \begin{cases} +1 \text{ if } \theta^\top x + \theta_0 > 0 \\ -1 \text{ if } \theta^\top x + \theta_0 \leq 0 \end{cases}$$

parameters

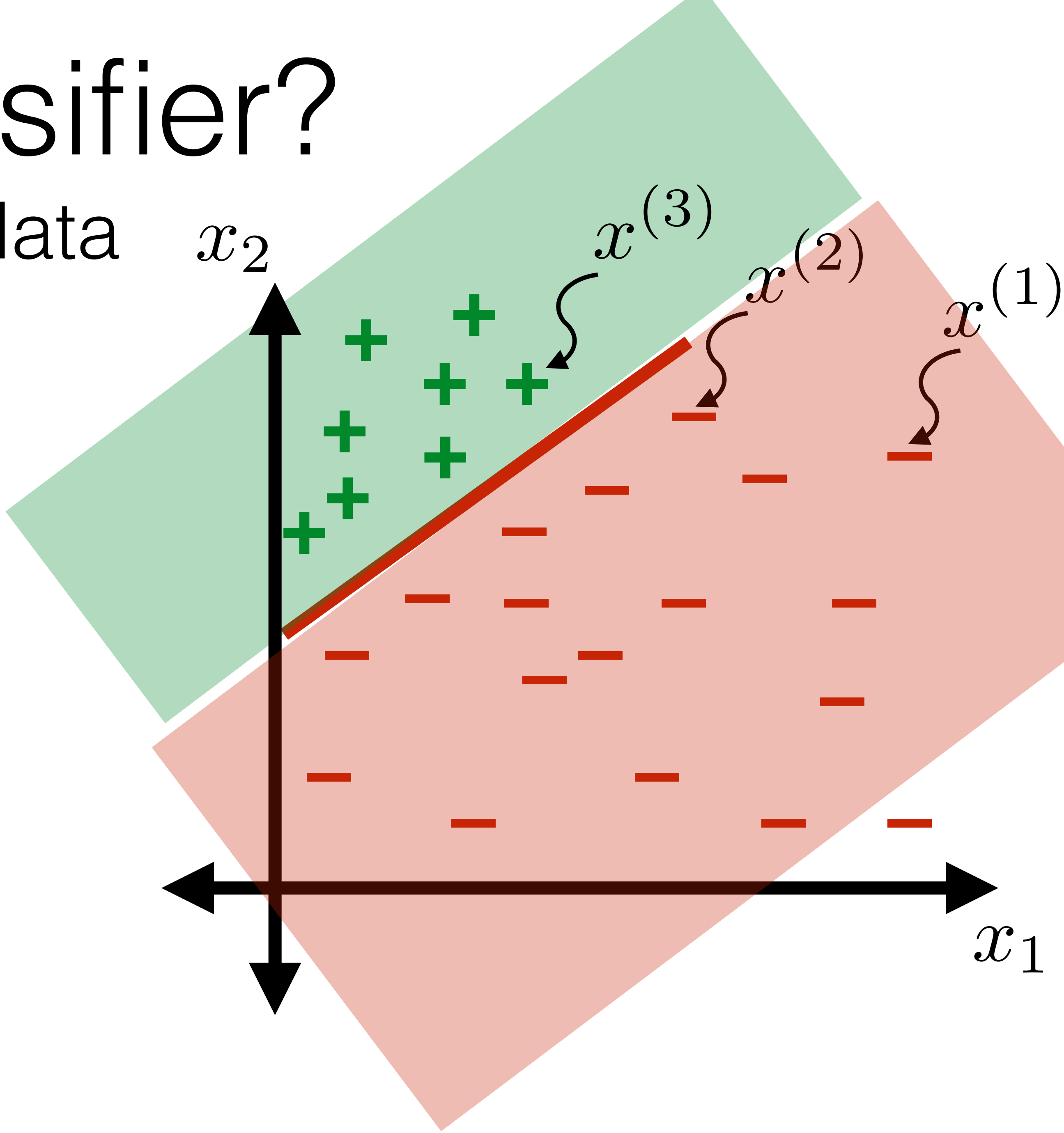
$$\mathcal{H} = \text{set of all such } h$$

How good is a classifier?



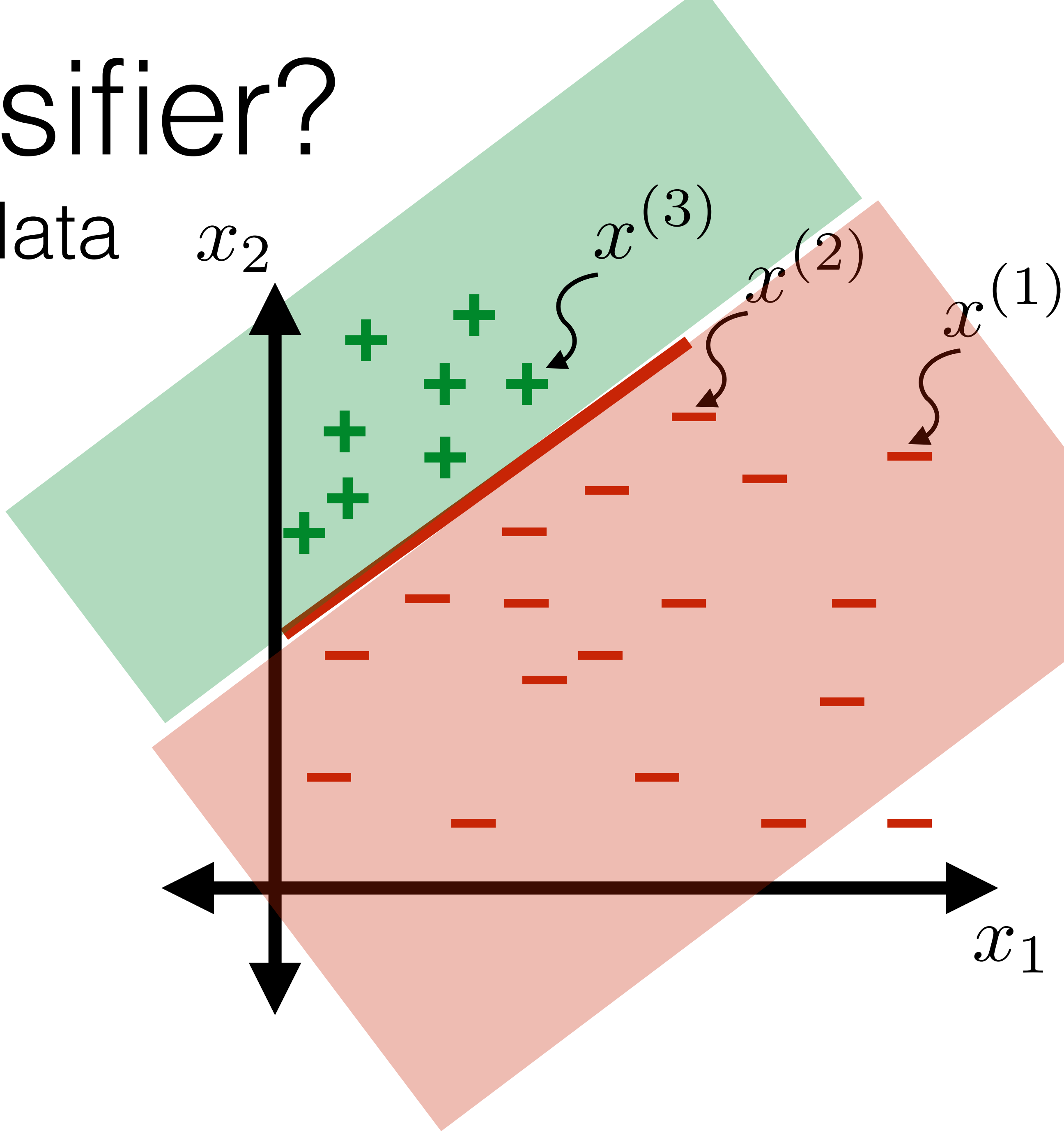
How good is a classifier?

- Should predict well on future data



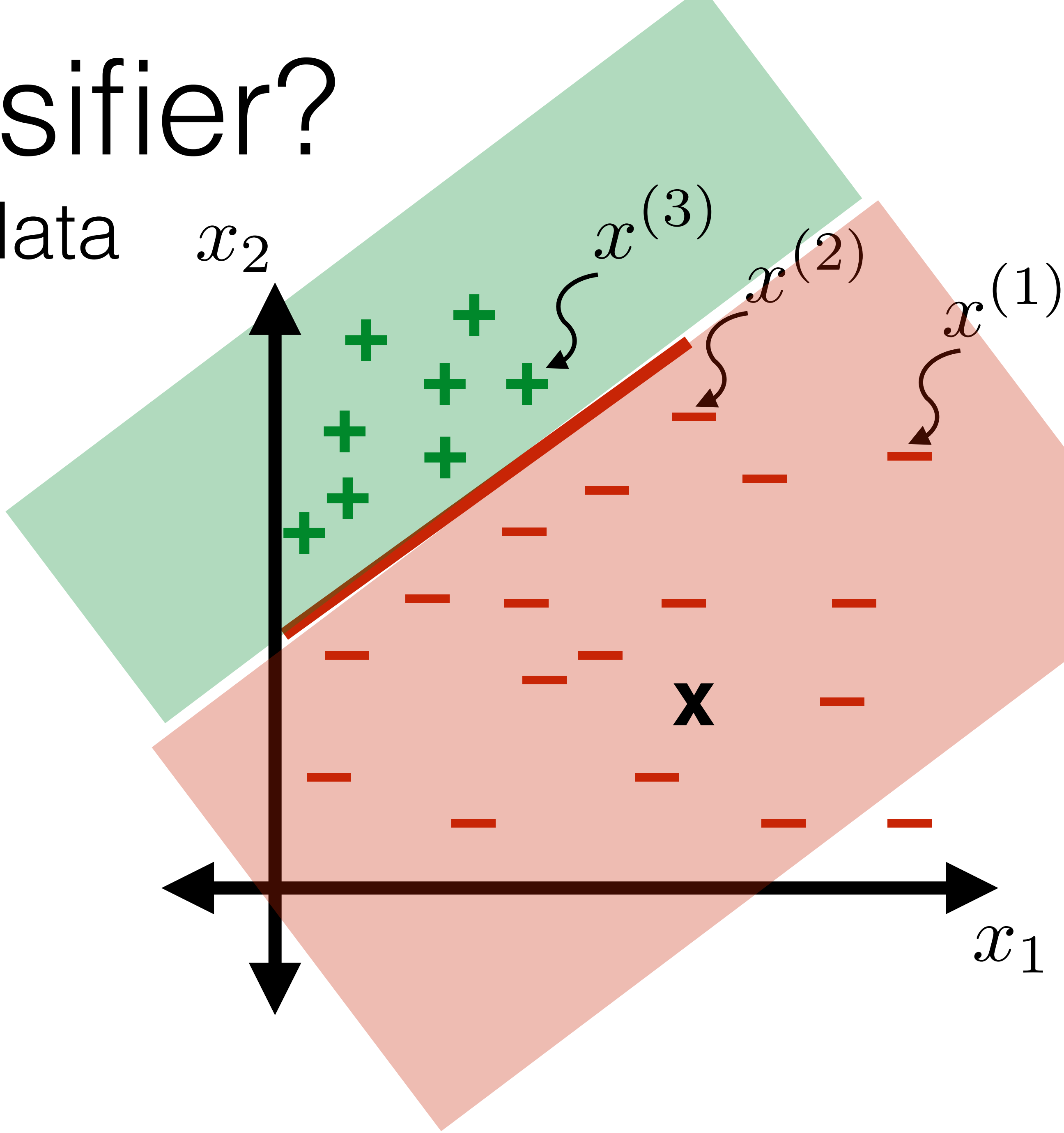
How good is a classifier?

- Should predict well on future data
- How good is a classifier at a single point?



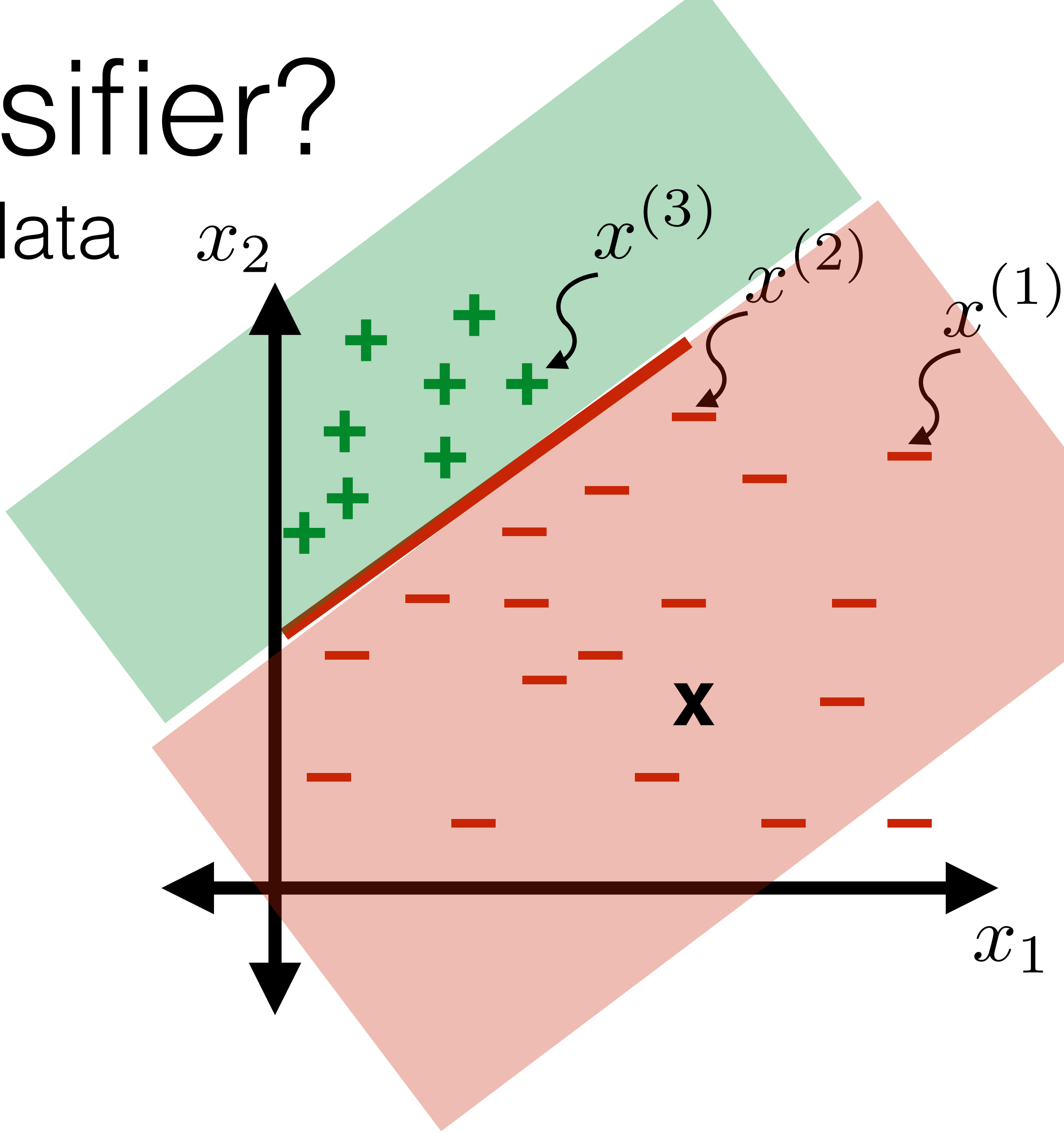
How good is a classifier?

- Should predict well on future data
- How good is a classifier at a single point?



How good is a classifier?

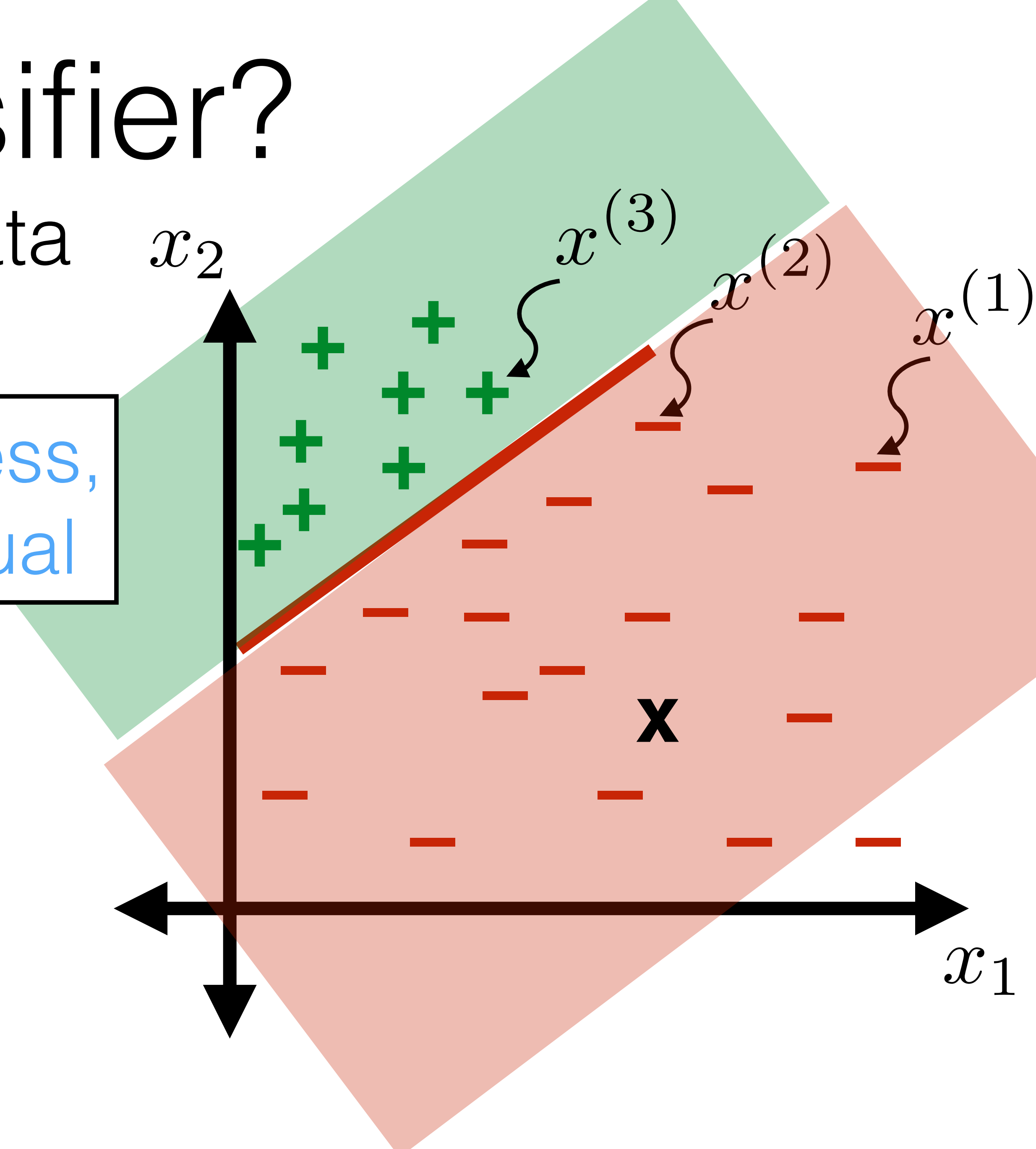
- Should predict well on future data
- How good is a classifier at a single point? Loss $L(g, a)$



How good is a classifier?

- Should predict well on future data
- How good is a classifier at a single point? Loss $L(g, a)$

g: guess,
a: actual

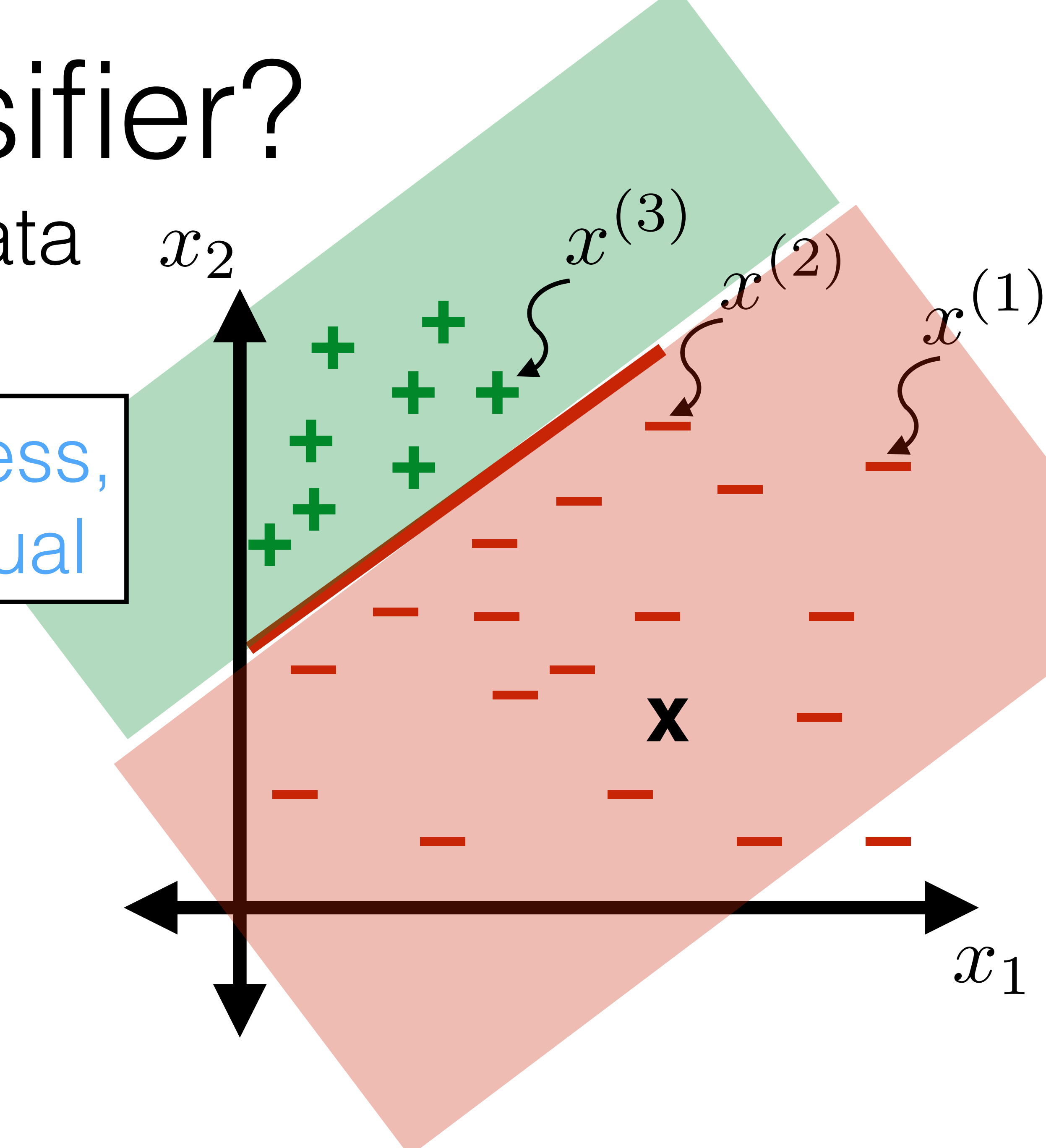


How good is a classifier?

- Should predict well on future data
- How good is a classifier at a single point? Loss $L(g, a)$

g: guess,
a: actual

    - Example: 0-1 loss

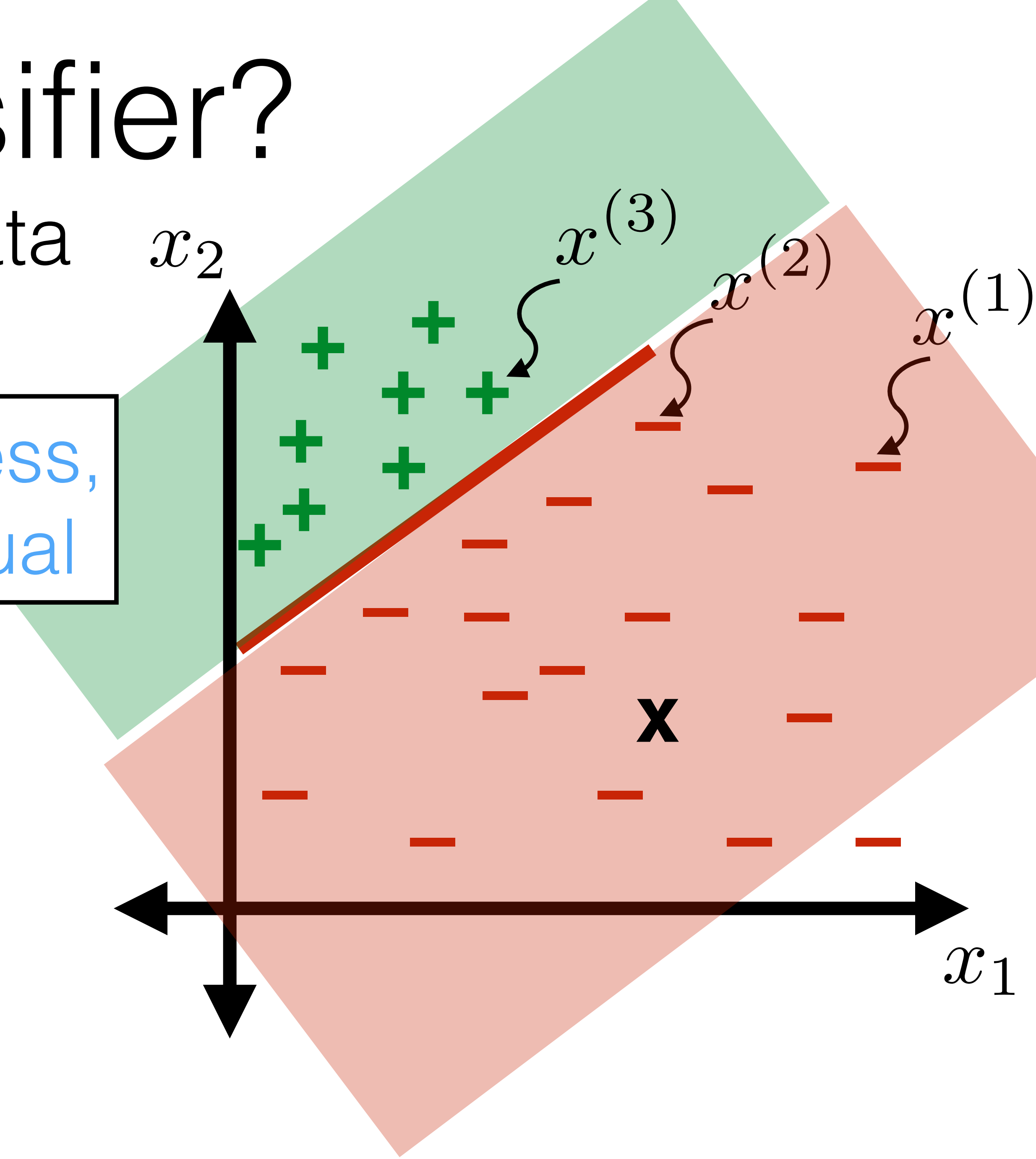


How good is a classifier?

- Should predict well on future data
- How good is a classifier at a single point? Loss $L(g, a)$
    - Example: 0-1 loss

$$L(g,a) = \begin{cases} 0 \text{ if } g = a \\ 1 \text{ else} \end{cases}$$

g: guess,
a: actual

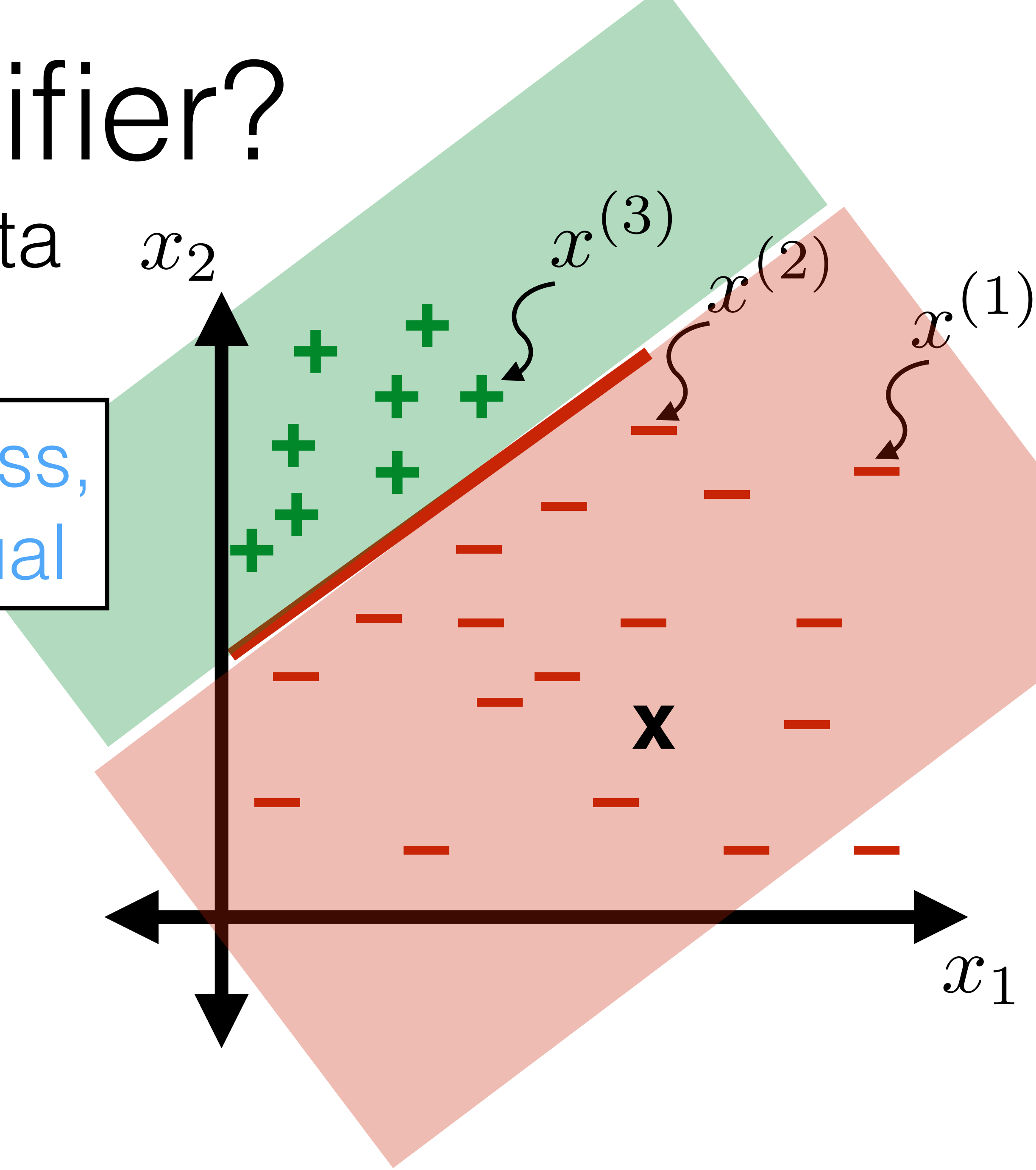


How good is a classifier?

- Should predict well on future data
- How good is a classifier at a single point? Loss $L(g, a)$

g: guess,
a: actual

  - Example: 0-1 loss
$$L(g, a) = \begin{cases} 0 \text{ if } g = a \\ 1 \text{ else} \end{cases}$$
  - Example: asymmetric loss



How good is a classifier?

- Should predict well on future data
- How good is a classifier at a single point? Loss $L(g, a)$

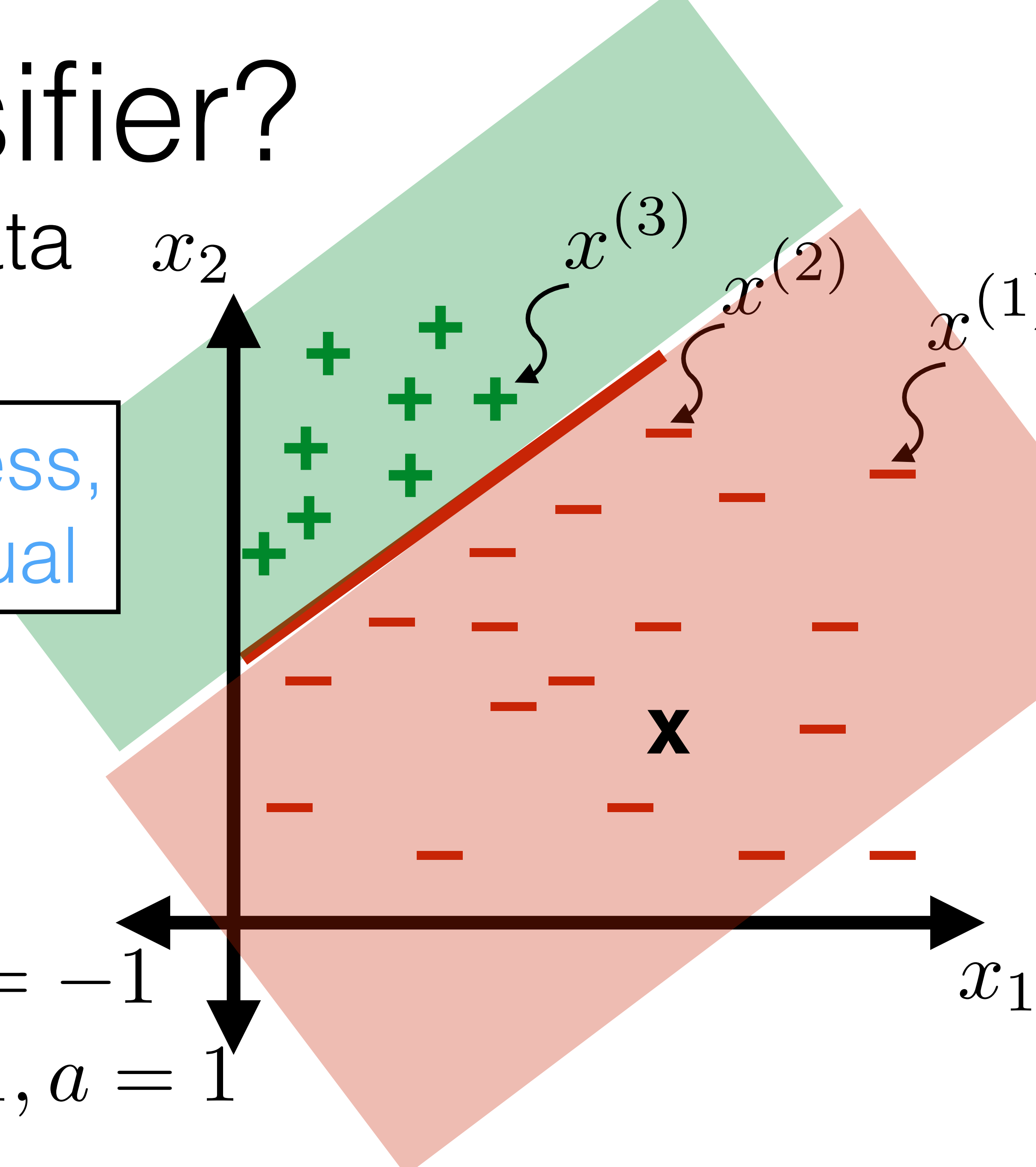
g: guess,
a: actual

  - Example: 0-1 loss

$$L(g, a) = \begin{cases} 0 \text{ if } g = a \\ 1 \text{ else} \end{cases}$$

  - Example: asymmetric loss

$$L(g, a) = \begin{cases} 1 \text{ if } g = 1, a = -1 \\ 100 \text{ if } g = -1, a = 1 \\ 0 \text{ else} \end{cases}$$



How good is a classifier?

- Should predict well on future data
- How good is a classifier at a single point? Loss $L(g, a)$

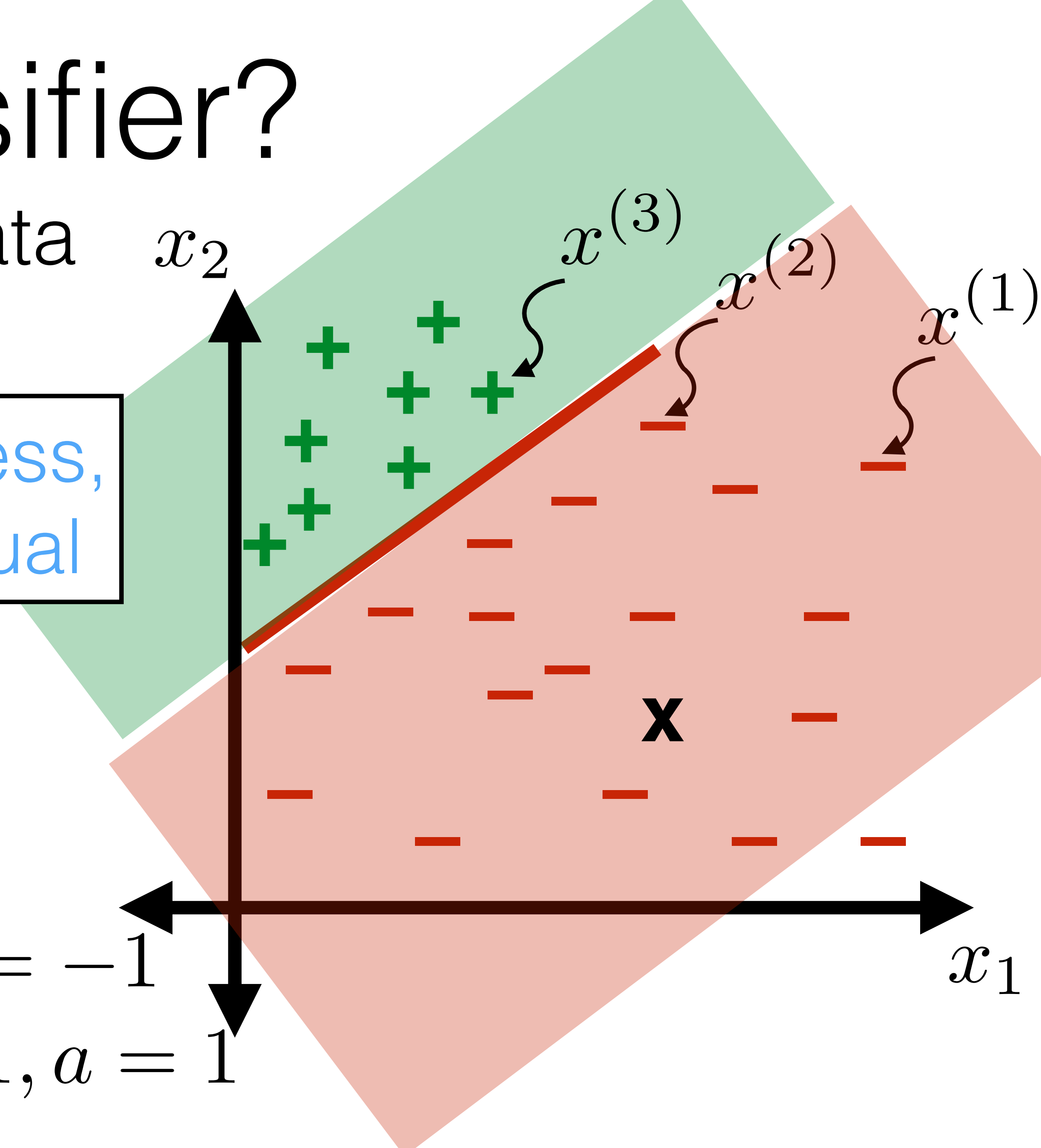
g: guess,
a: actual

  - Example: 0-1 loss

$$L(g, a) = \begin{cases} 0 \text{ if } g = a \\ 1 \text{ else} \end{cases}$$

  - Example: asymmetric loss

$$L(g, a) = \begin{cases} 1 \text{ if } g = 1, a = -1 \\ 100 \text{ if } g = -1, a = 1 \\ 0 \text{ else} \end{cases}$$



- Test error (n' new points):

How good is a classifier?

- Should predict well on future data
- How good is a classifier at a single point? Loss $L(g, a)$

g: guess,
a: actual

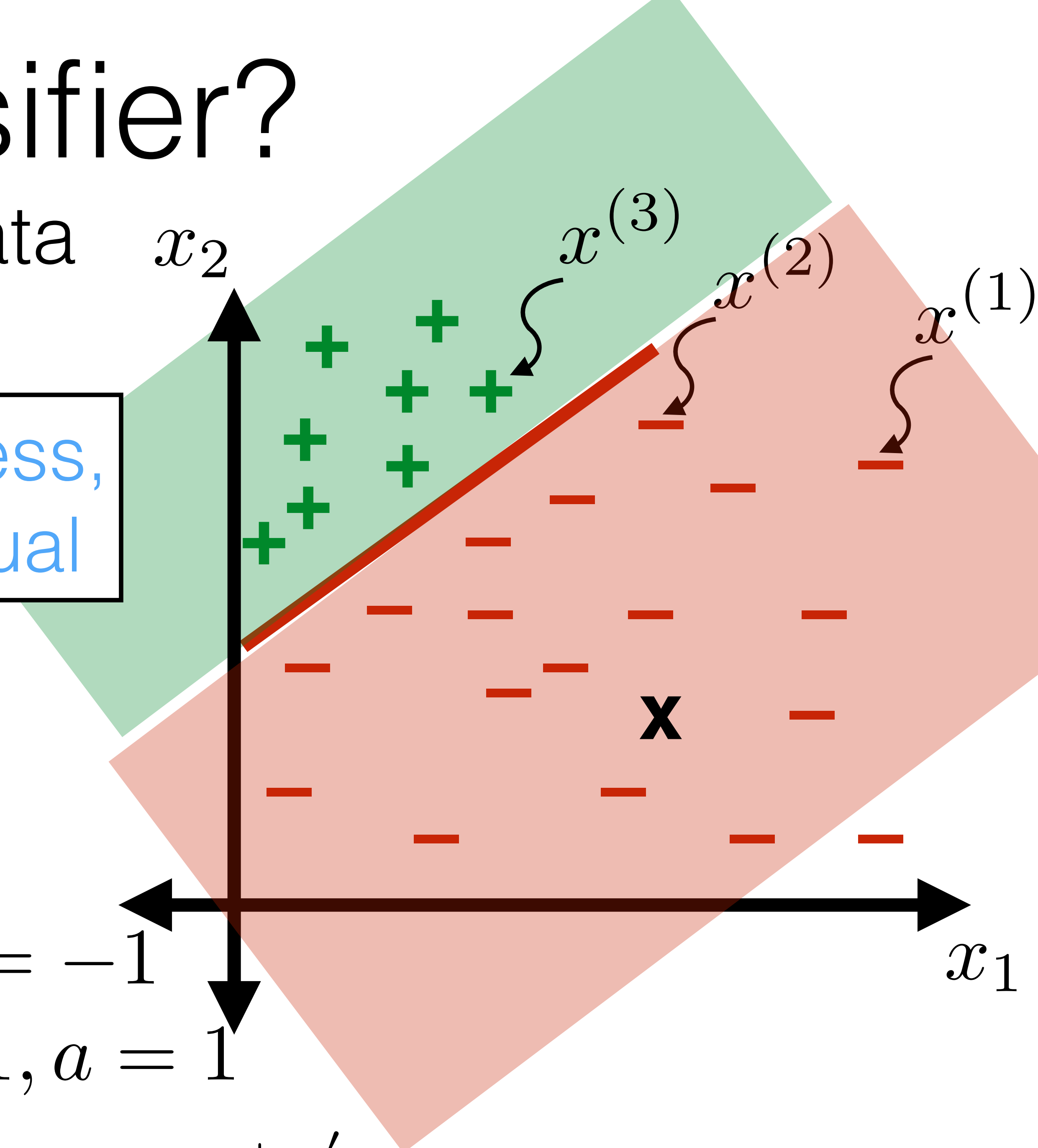
  - Example: 0-1 loss

$$L(g,a) = \begin{cases} 0 \text{ if } g = a \\ 1 \text{ else} \end{cases}$$

  - Example: asymmetric loss

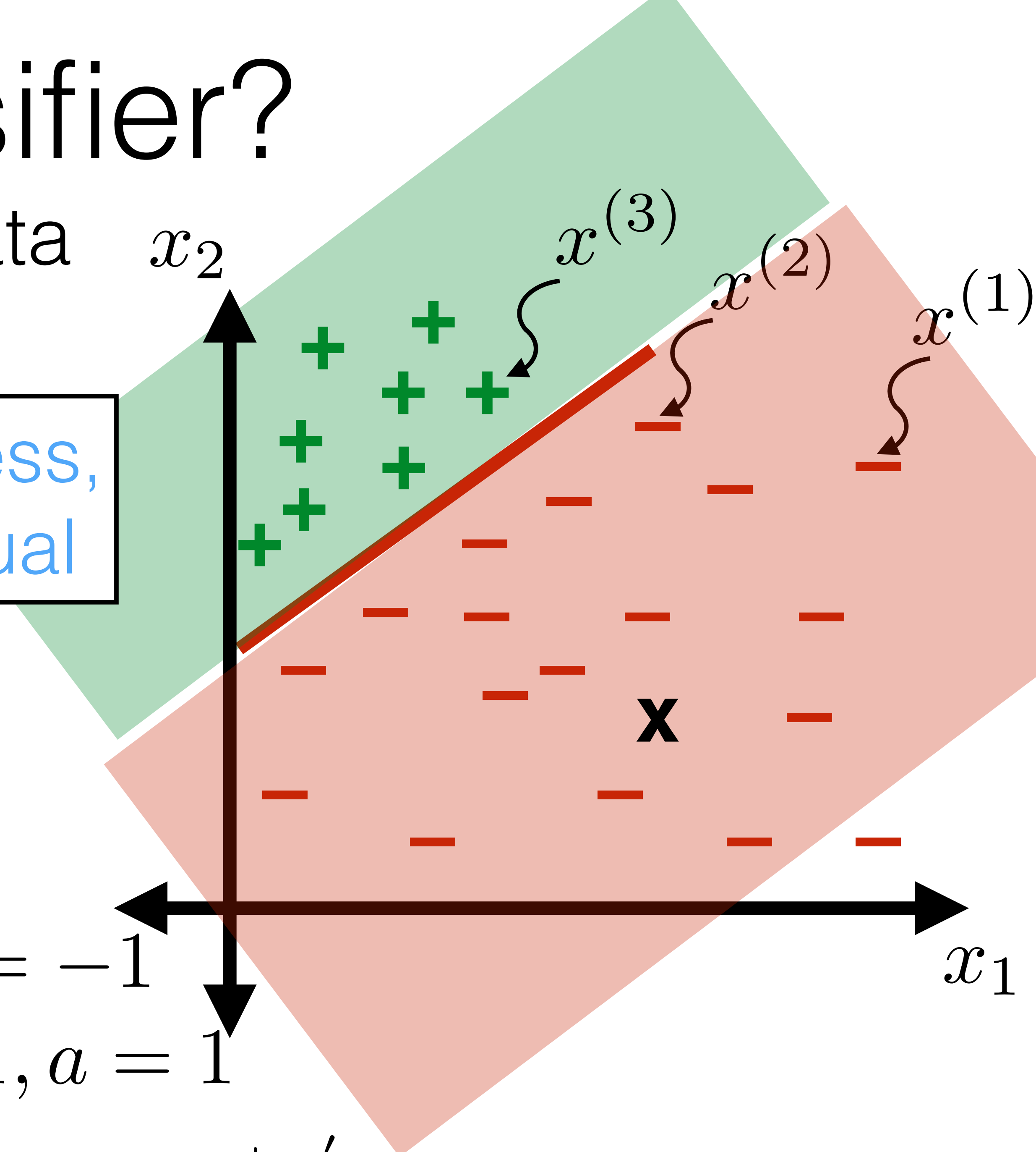
$$L(g,a) = \begin{cases} 1 \text{ if } g = 1, a = -1 \\ 100 \text{ if } g = -1, a = 1 \\ 0 \text{ else} \end{cases}$$

- Test error (n' new points): $\mathcal{E}(h) = \frac{1}{n'} \sum_{i=n+1}^{n+n'} L(h(x^{(i)}), y^{(i)})$



How good is a classifier?

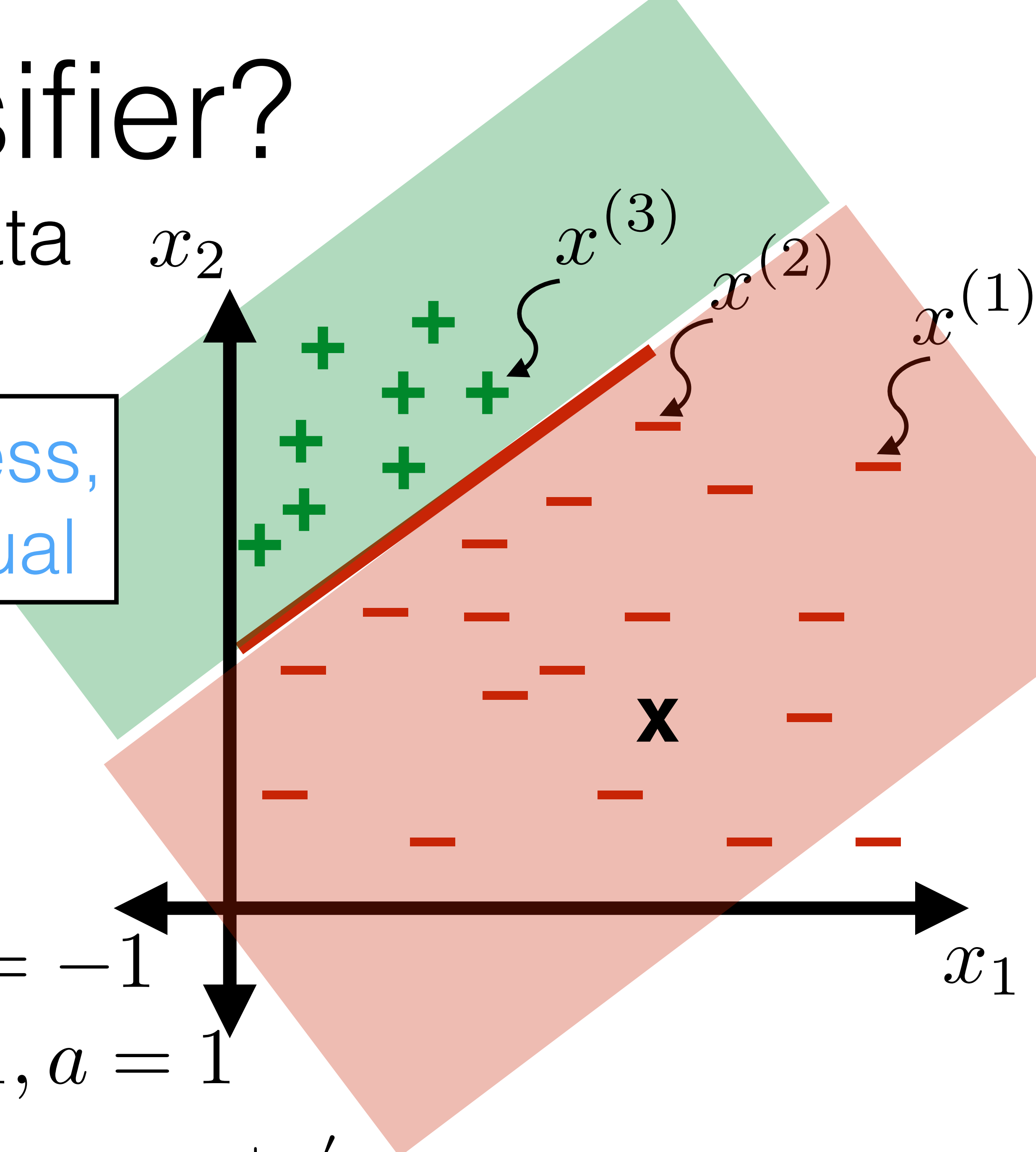
- Should predict well on future data
- How good is a classifier at a single point? Loss $L(g, a)$ g: guess, a: actual
    - Example: 0-1 loss
      $$L(g,a) = \begin{cases} 0 \text{ if } g = a \\ 1 \text{ else} \end{cases}$$
    - Example: asymmetric loss
      $$L(g,a) = \begin{cases} 1 \text{ if } g = 1, a = -1 \\ 100 \text{ if } g = -1, a = 1 \\ 0 \text{ else} \end{cases}$$
- Test error (n' new points): $\mathcal{E}(h) = \frac{1}{n'} \sum_{i=n+1}^{n+n'} L(h(x^{(i)}), y^{(i)})$
- Training error:



How good is a classifier?

- Should predict well on future data
- How good is a classifier at a single point? Loss $L(g, a)$

g: guess,
a: actual



  - Example: 0-1 loss

$$L(g,a) = \begin{cases} 0 \text{ if } g = a \\ 1 \text{ else} \end{cases}$$

  - Example: asymmetric loss

$$L(g,a) = \begin{cases} 1 \text{ if } g = 1, a = -1 \\ 100 \text{ if } g = -1, a = 1 \\ 0 \text{ else} \end{cases}$$

- Test error (n' new points): $\mathcal{E}(h) = \frac{1}{n'} \sum_{i=n+1}^{n+n'} L(h(x^{(i)}), y^{(i)})$
- Training error: $\mathcal{E}_n(h) = \frac{1}{n} \sum_{i=1}^{n} L(h(x^{(i)}), y^{(i)})$

How good is a classifier?

- Should predict well on future data
- How good is a classifier at a single point? Loss $L(g, a)$

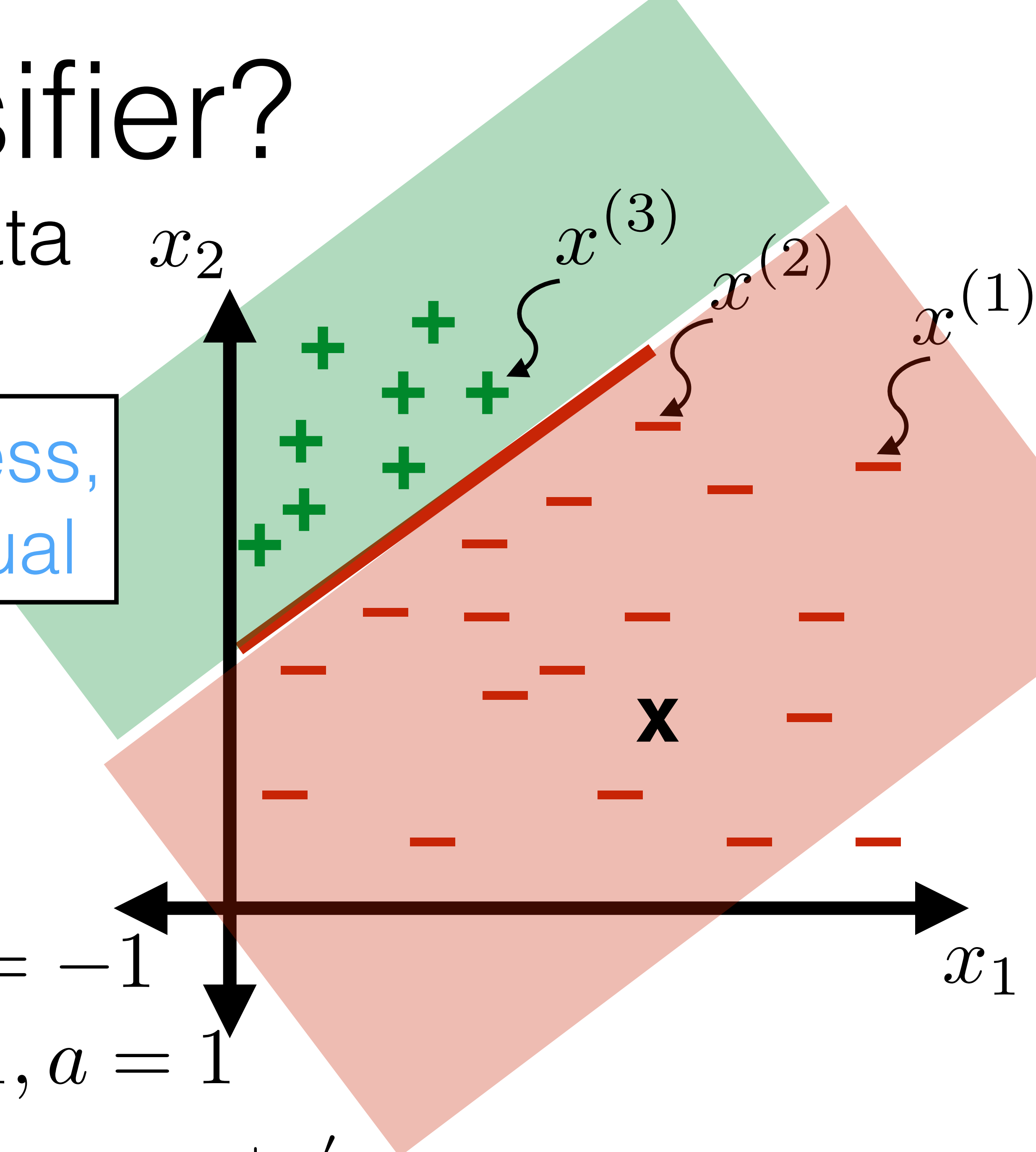
g: guess,
a: actual

  - Example: 0-1 loss

$$L(g,a) = \begin{cases} 0 \text{ if } g = a \\ 1 \text{ else} \end{cases}$$

  - Example: asymmetric loss

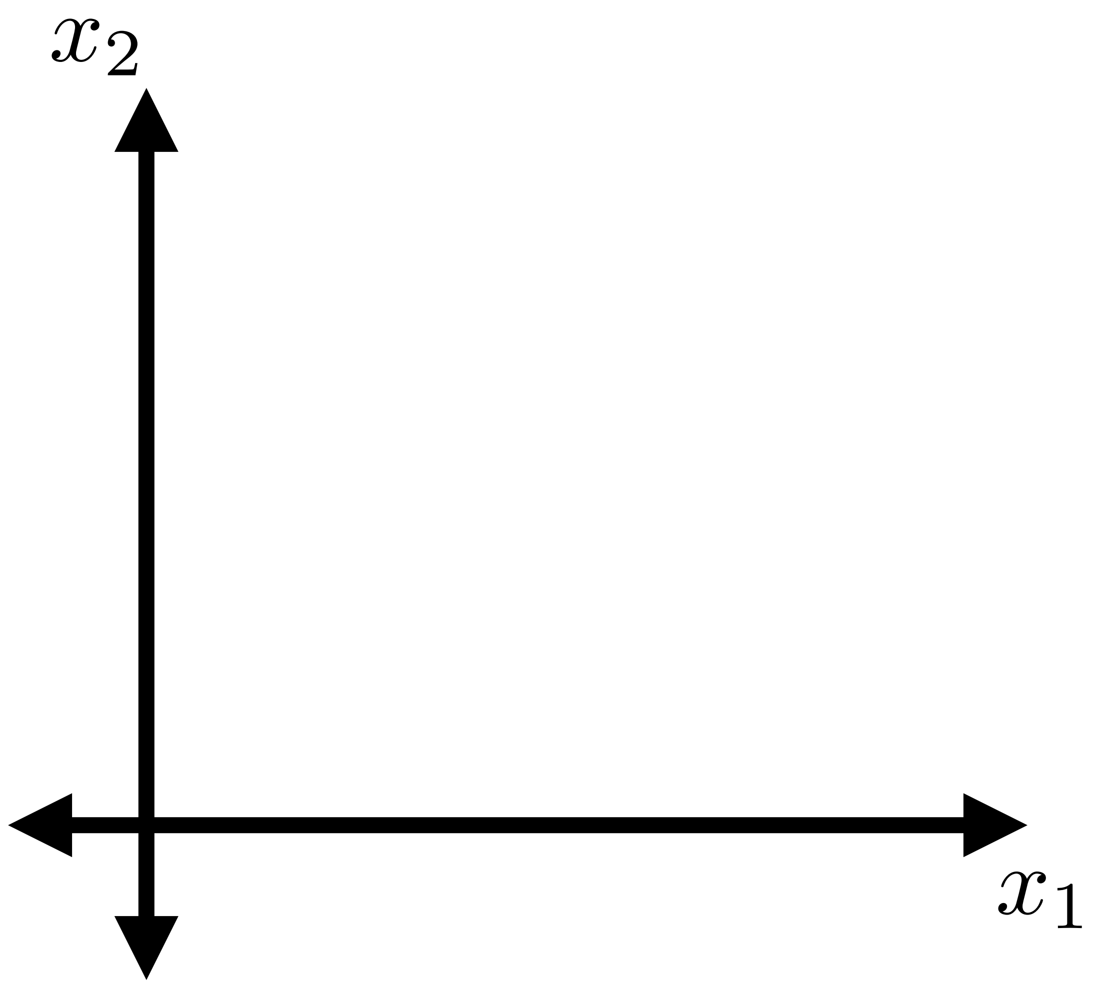
$$L(g,a) = \begin{cases} 1 \text{ if } g = 1, a = -1 \\ 100 \text{ if } g = -1, a = 1 \\ 0 \text{ else} \end{cases}$$



- Test error (n' new points): $\mathcal{E}(h) = \frac{1}{n'} \sum_{i=n+1}^{n+n'} L(h(x^{(i)}), y^{(i)})$
- Training error: $\mathcal{E}_n(h) = \frac{1}{n} \sum_{i=1}^{n} L(h(x^{(i)}), y^{(i)})$
- Prefer h to $\tilde{h}$ if $\mathcal{E}_n(h) < \mathcal{E}_n(\tilde{h})$

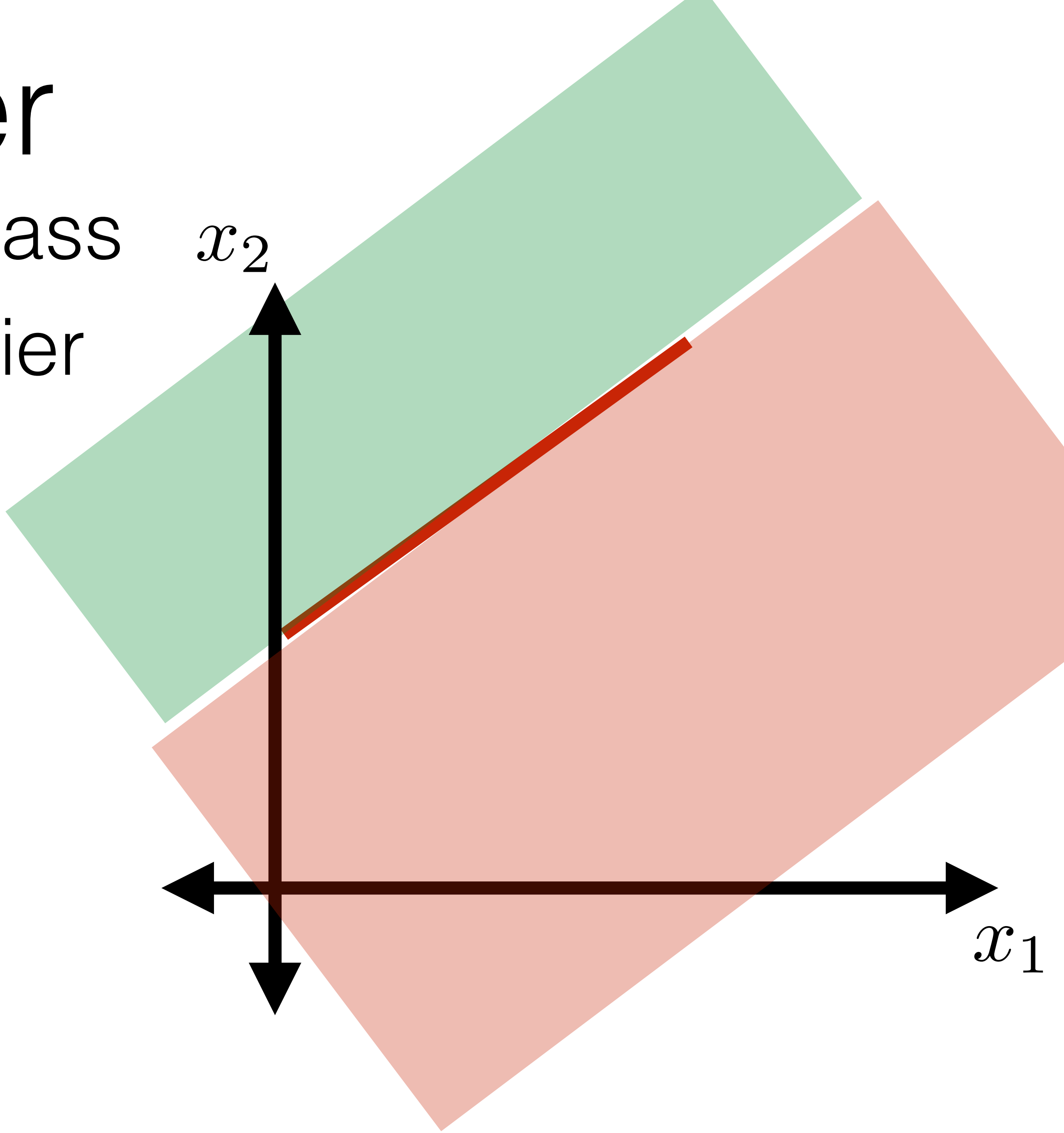
Learning a classifier

- Have data; have hypothesis class
- Want to choose a good classifier
  - Recall: $x \longrightarrow \boxed{h} \longrightarrow y$



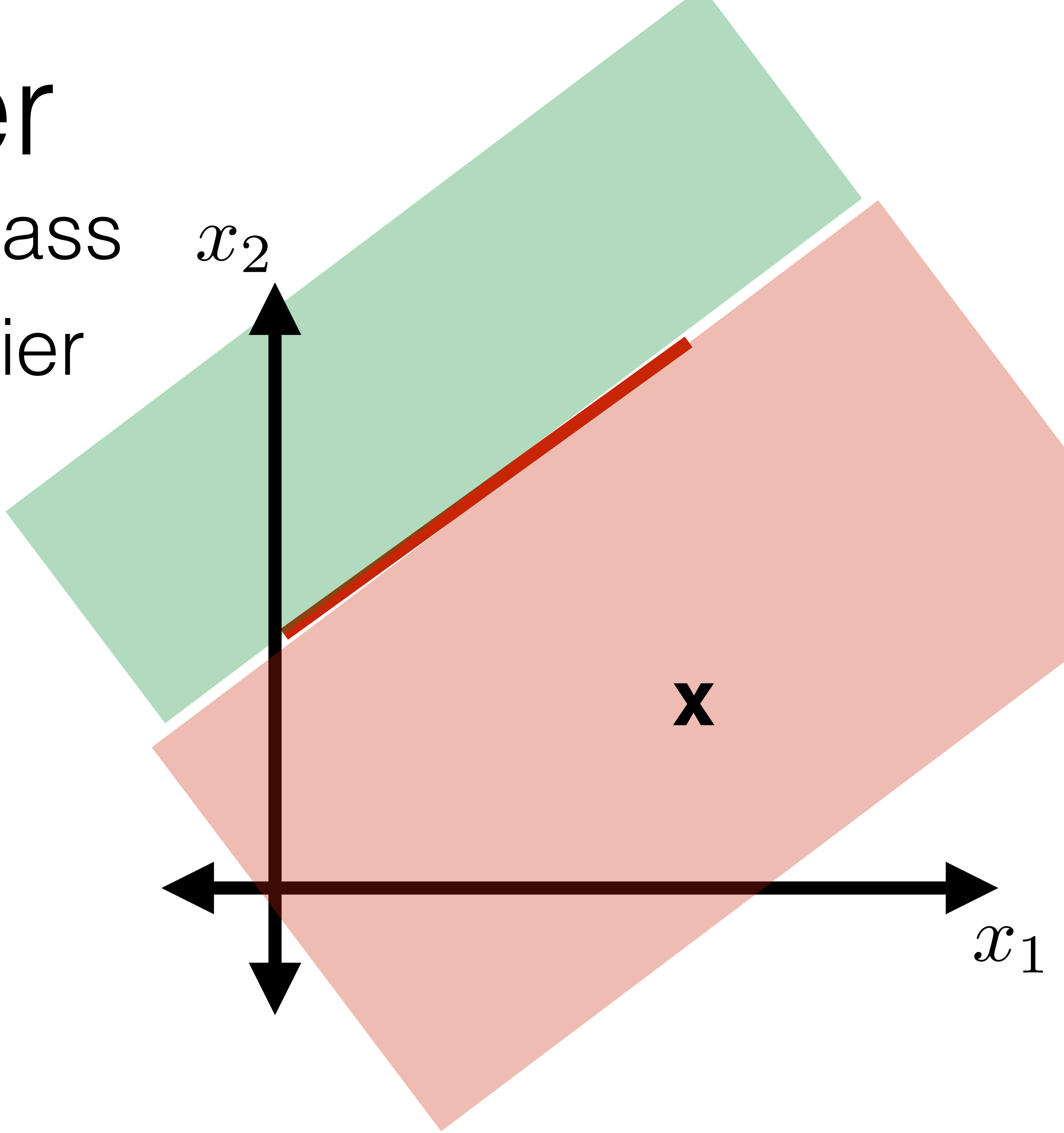
Learning a classifier

- Have data; have hypothesis class
- Want to choose a good classifier
  - Recall: $x \rightarrow \boxed{h} \rightarrow y$



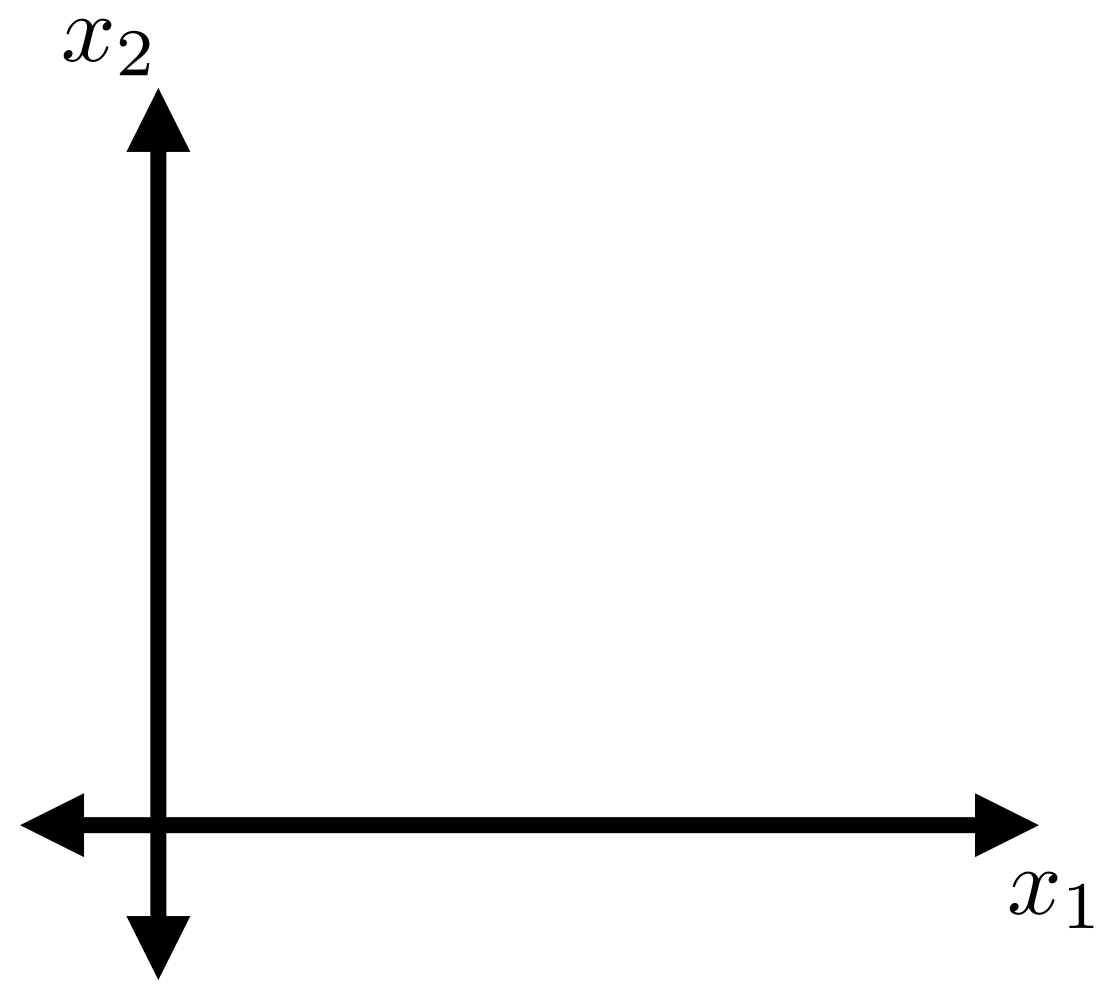
Learning a classifier

- Have data; have hypothesis class
- Want to choose a good classifier
  - Recall: $x \rightarrow \boxed{h} \rightarrow y$



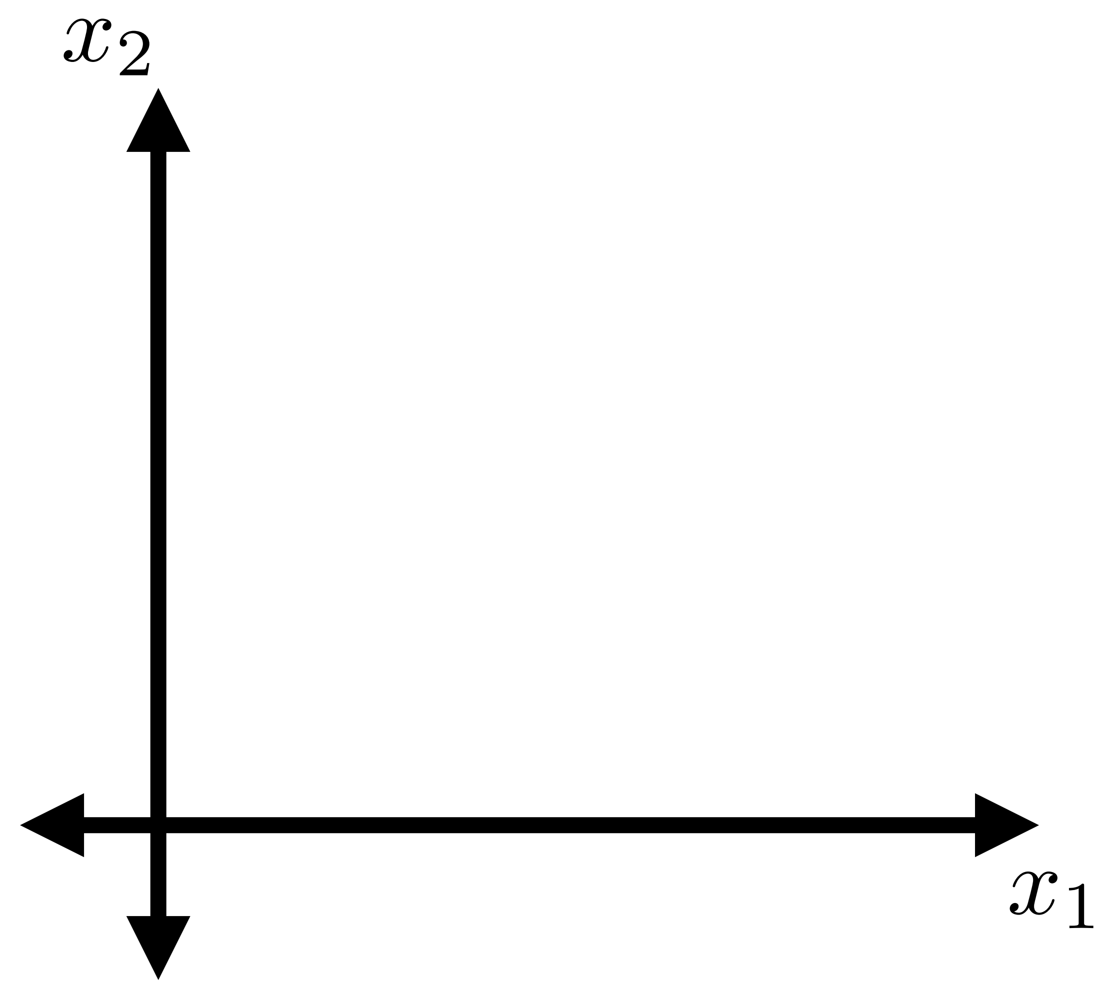
Learning a classifier

- Have data; have hypothesis class
- Want to choose a good classifier
    - Recall: $x \rightarrow \boxed{h} \rightarrow y$



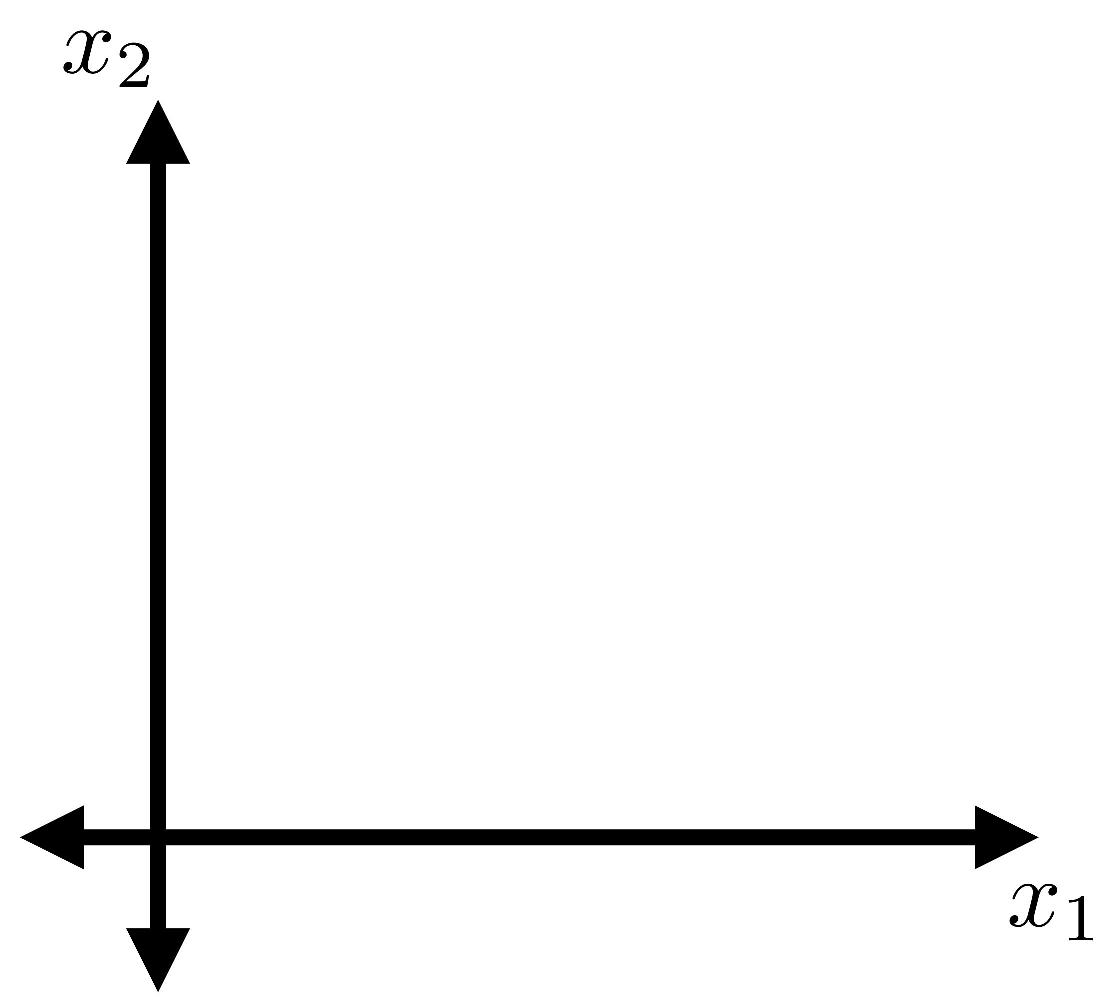
Learning a classifier

- Have data; have hypothesis class
- Want to choose a good classifier
  - Recall: $x \longrightarrow \boxed{h} \longrightarrow y$
  - New:



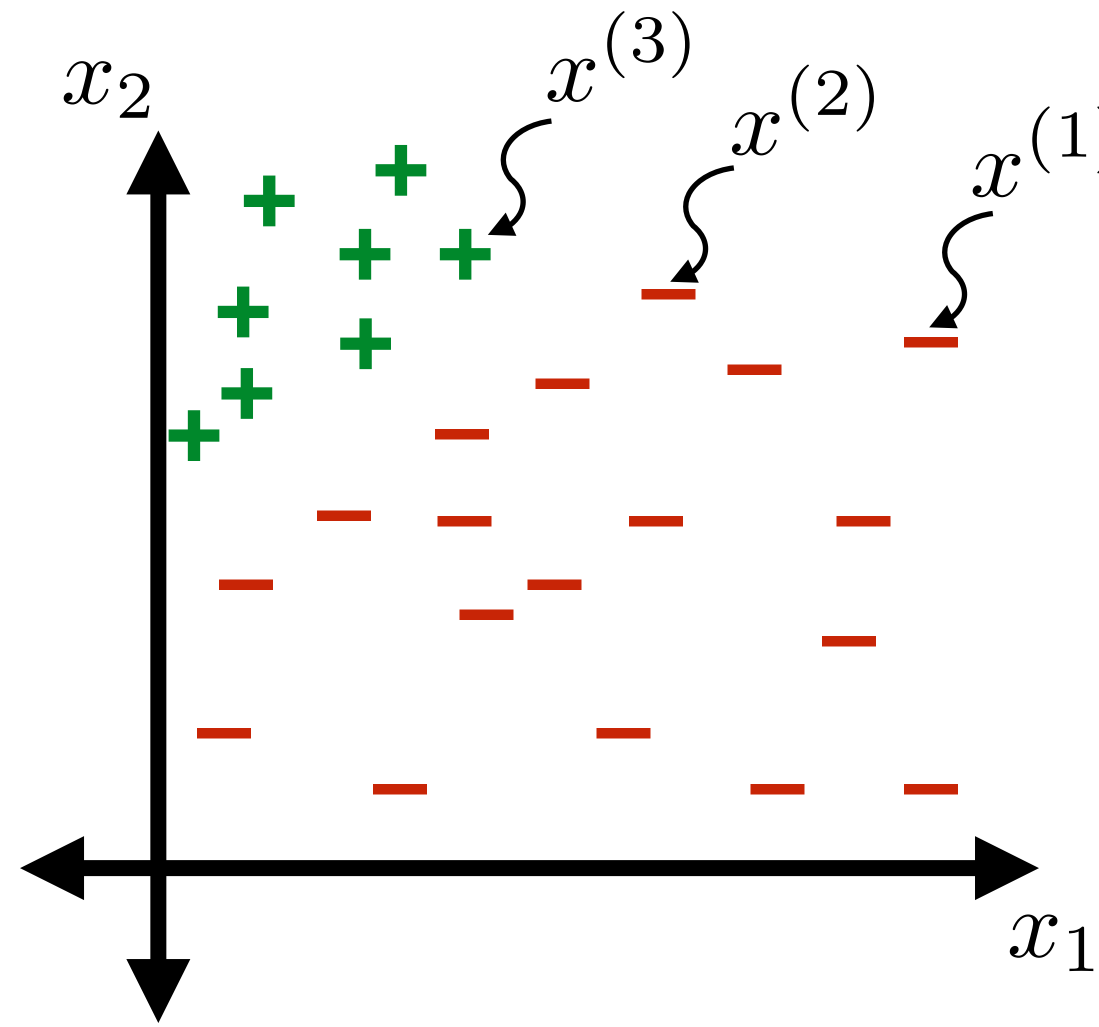
Learning a classifier

- Have data; have hypothesis class
- Want to choose a good classifier
  - Recall: $x \rightarrow \boxed{h} \rightarrow y$
  - New: $\mathcal{D}_n \rightarrow$ learning algorithm $\rightarrow h$



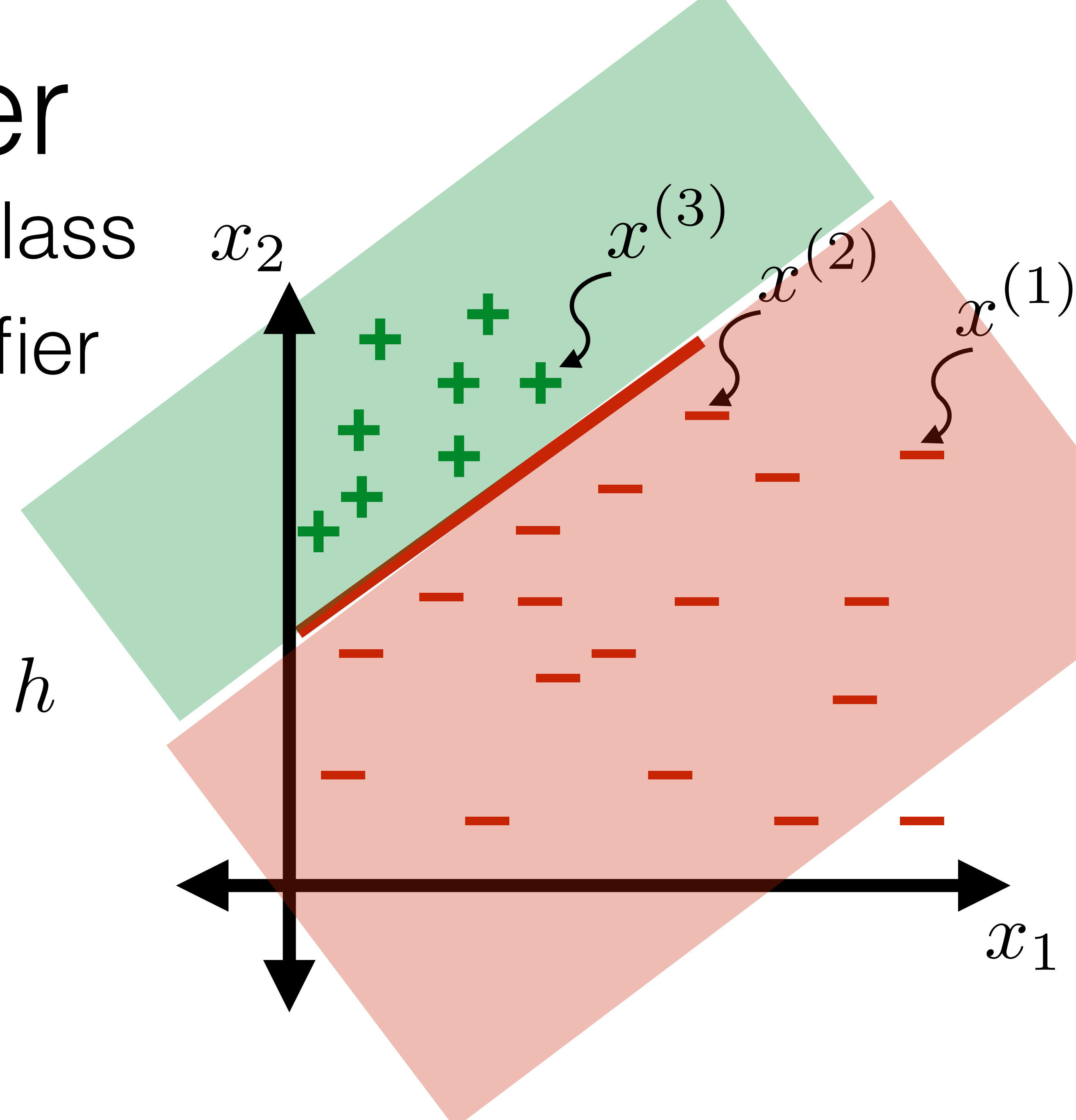
Learning a classifier

- Have data; have hypothesis class
- Want to choose a good classifier
  - Recall: $x \rightarrow \boxed{h} \rightarrow y$
  - New: $\mathcal{D}_n \rightarrow \boxed{\text{learning algorithm}} \rightarrow h$



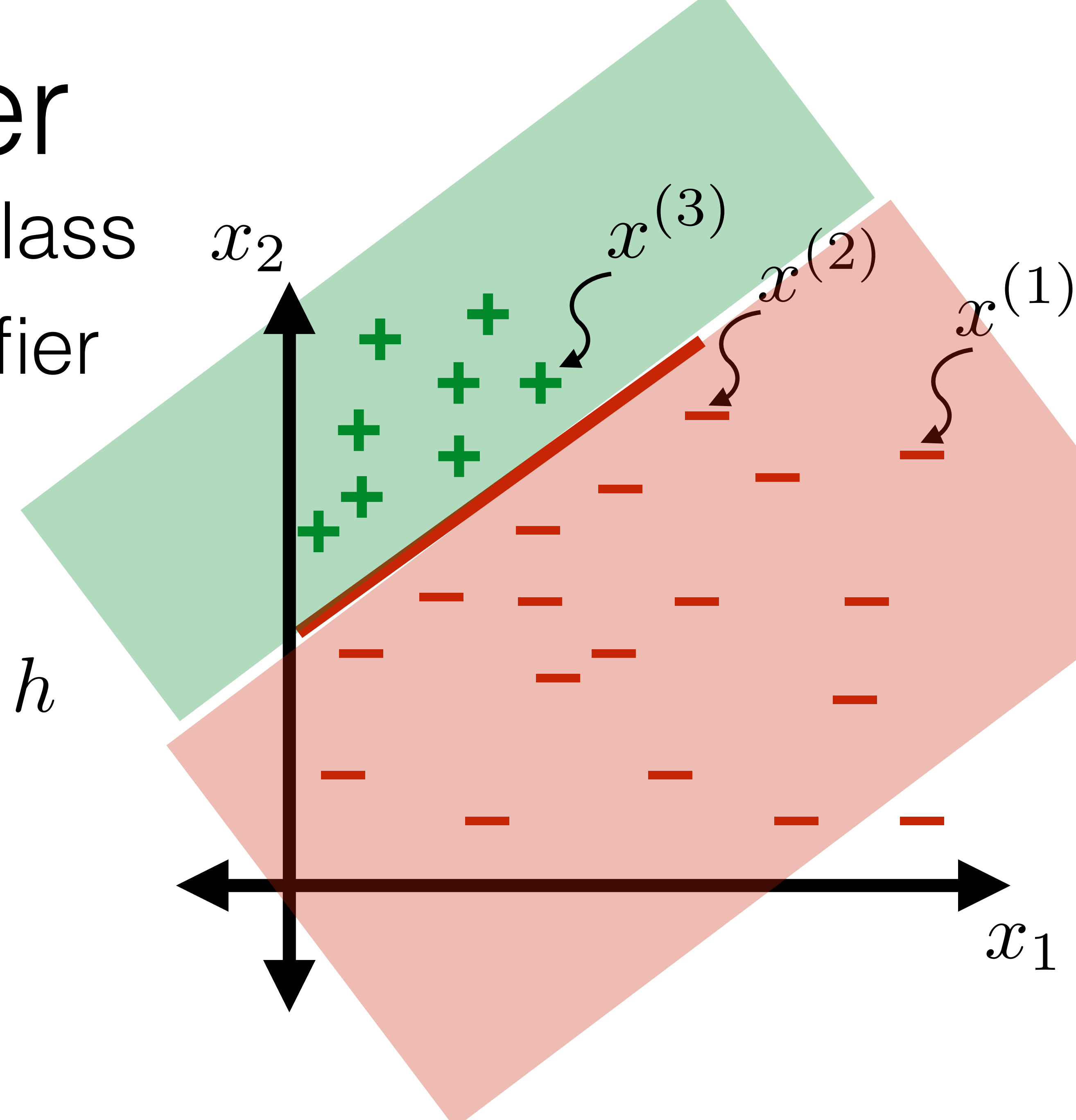
Learning a classifier

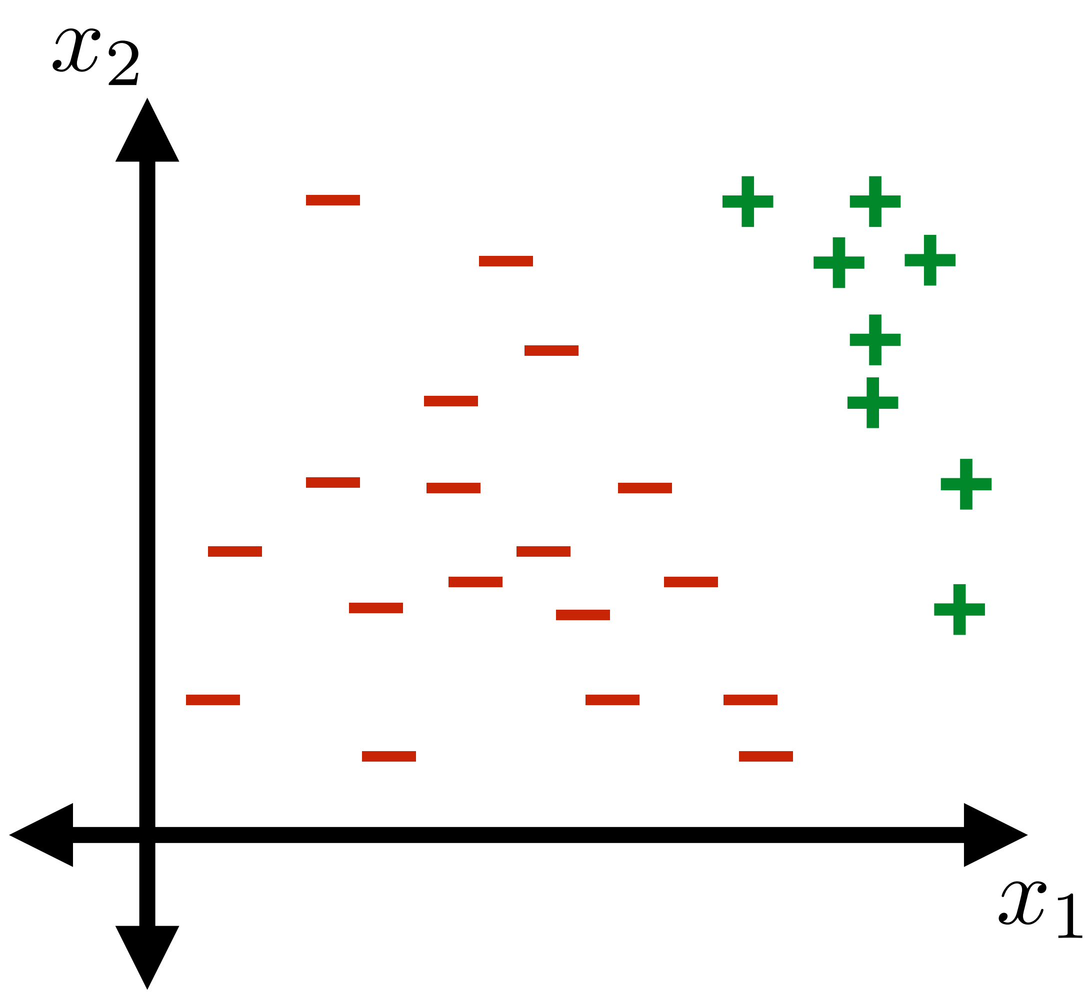
- Have data; have hypothesis class
- Want to choose a good classifier
  - Recall: $x \rightarrow \boxed{h} \rightarrow y$
  - New: $\mathcal{D}_n \rightarrow \boxed{\text{learning algorithm}} \rightarrow h$



Learning a classifier

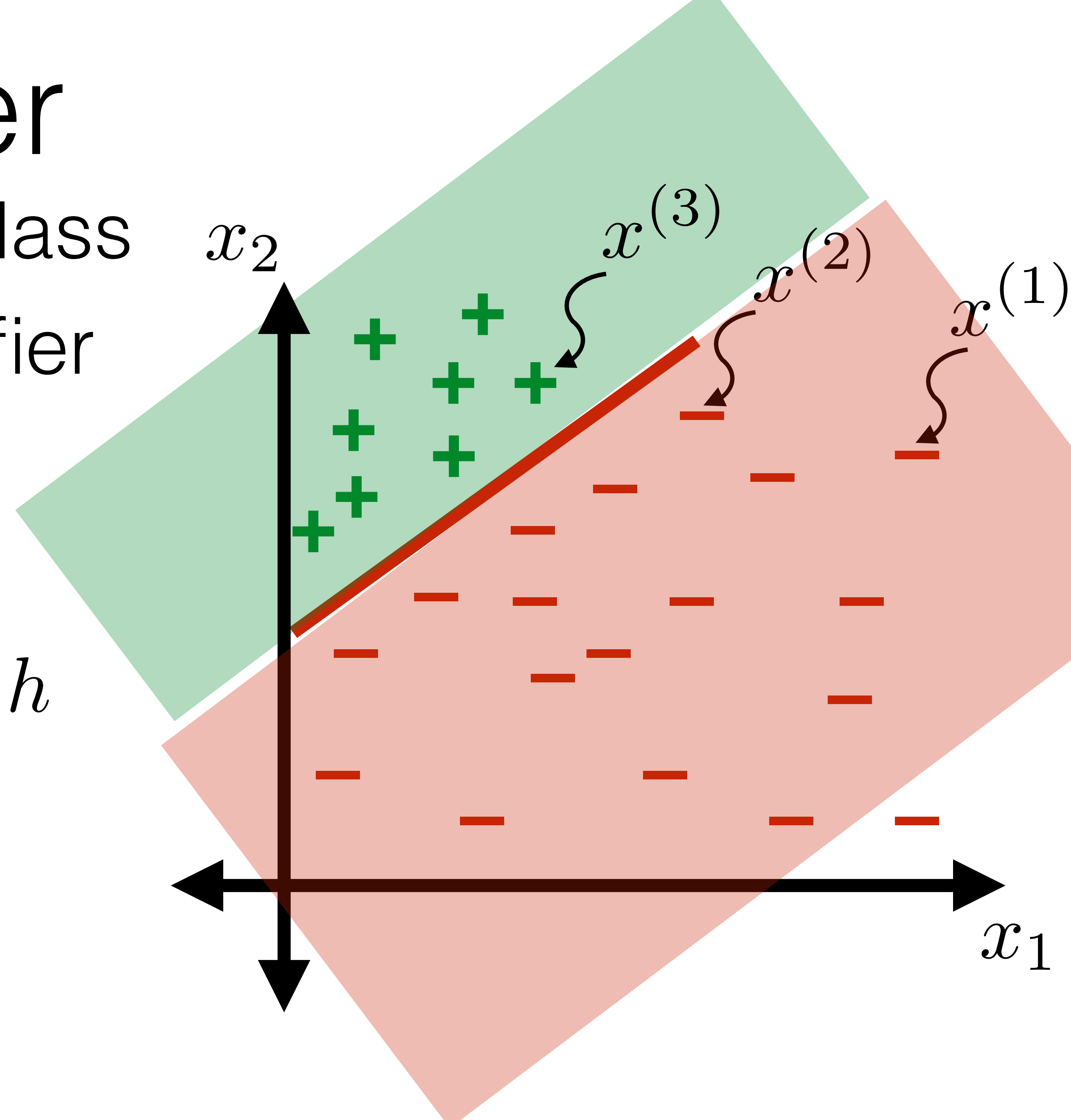
- Have data; have hypothesis class
- Want to choose a good classifier
  - Recall: $x \rightarrow \boxed{h} \rightarrow y$
  - New: $\mathcal{D}_n \rightarrow \boxed{\text{learning algorithm}} \rightarrow h$

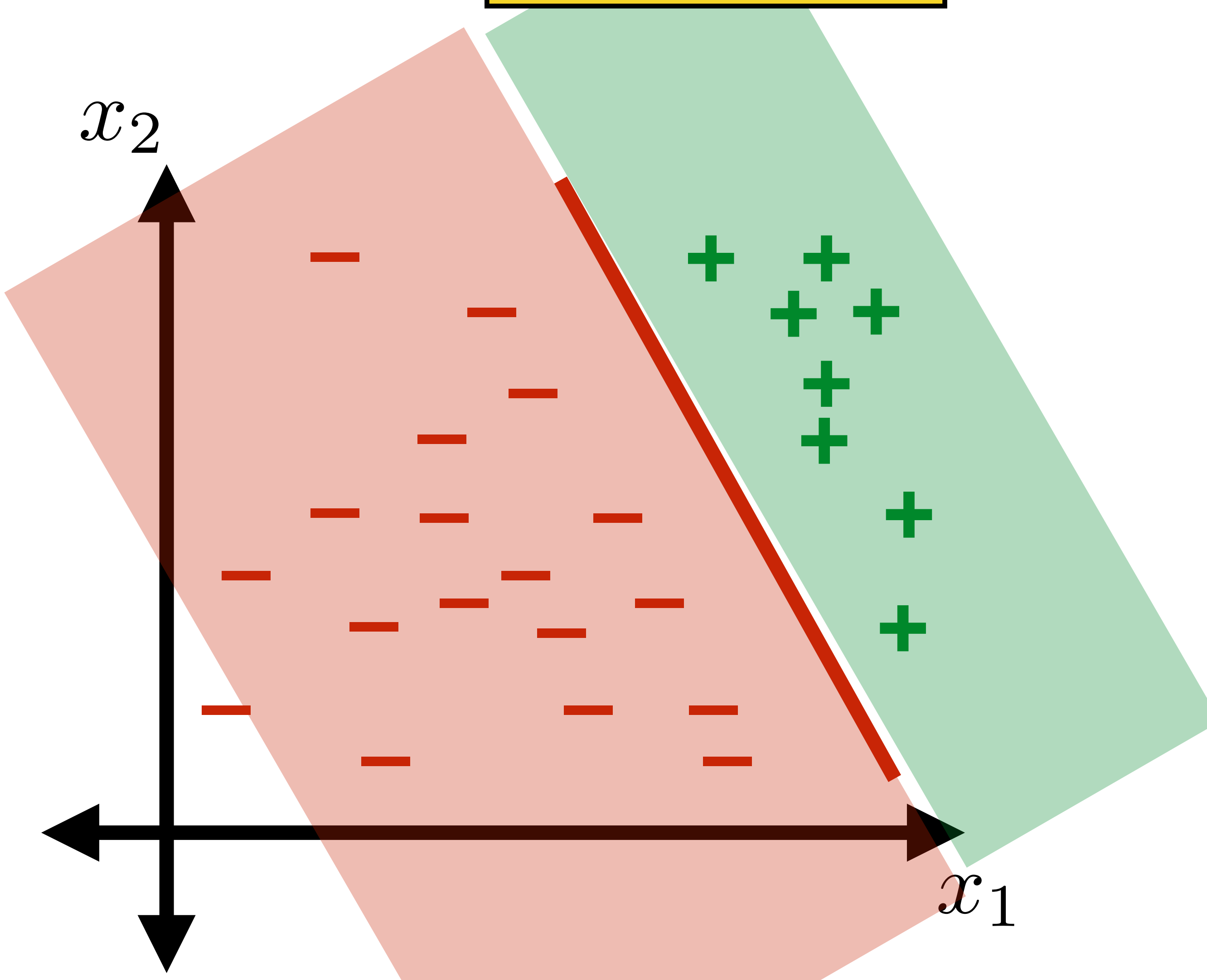




Learning a classifier

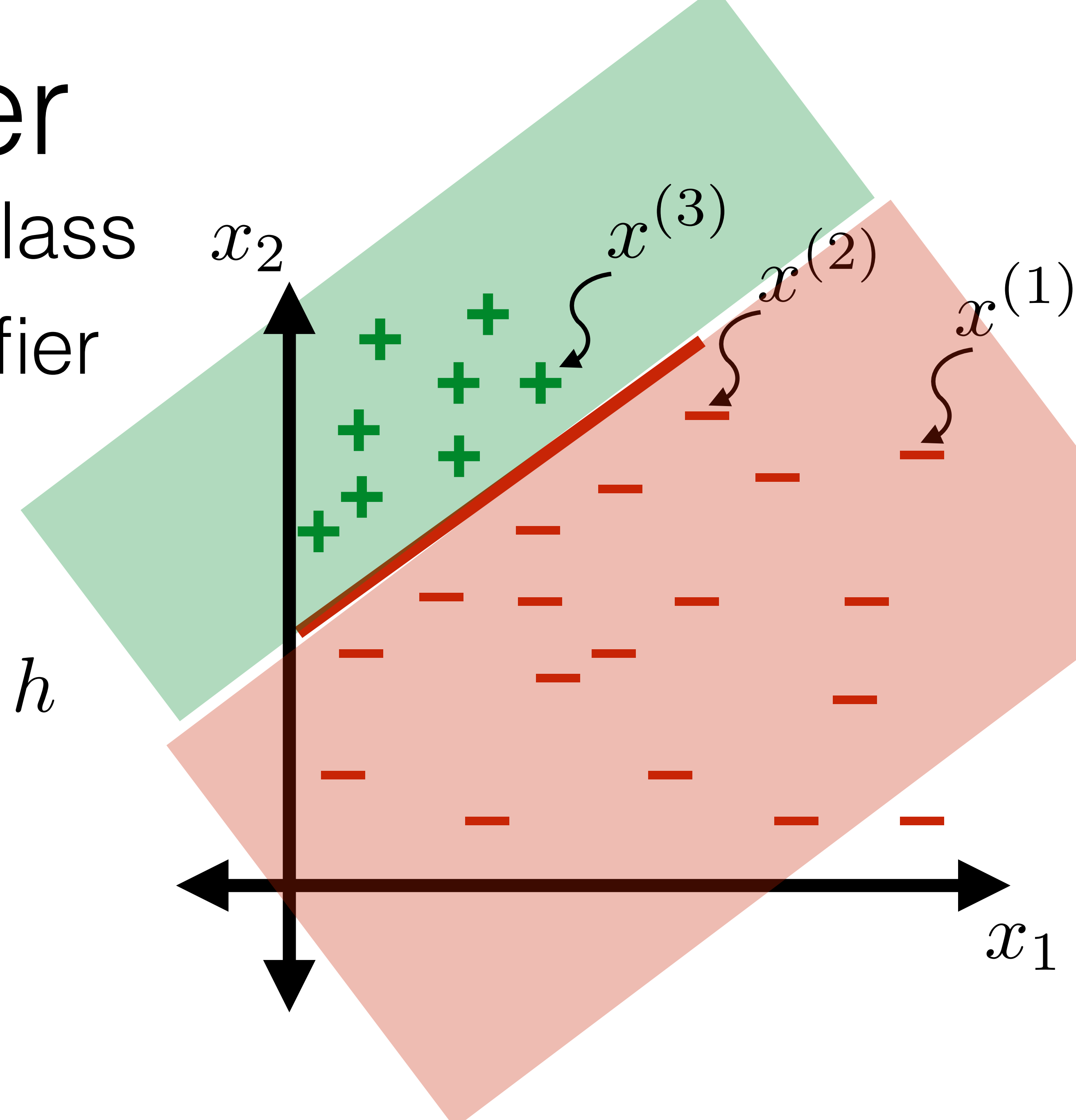
- Have data; have hypothesis class
- Want to choose a good classifier
    - Recall: $x \rightarrow \boxed{h} \rightarrow y$
    - New: $\mathcal{D}_n \rightarrow \boxed{\text{learning algorithm}} \rightarrow h$





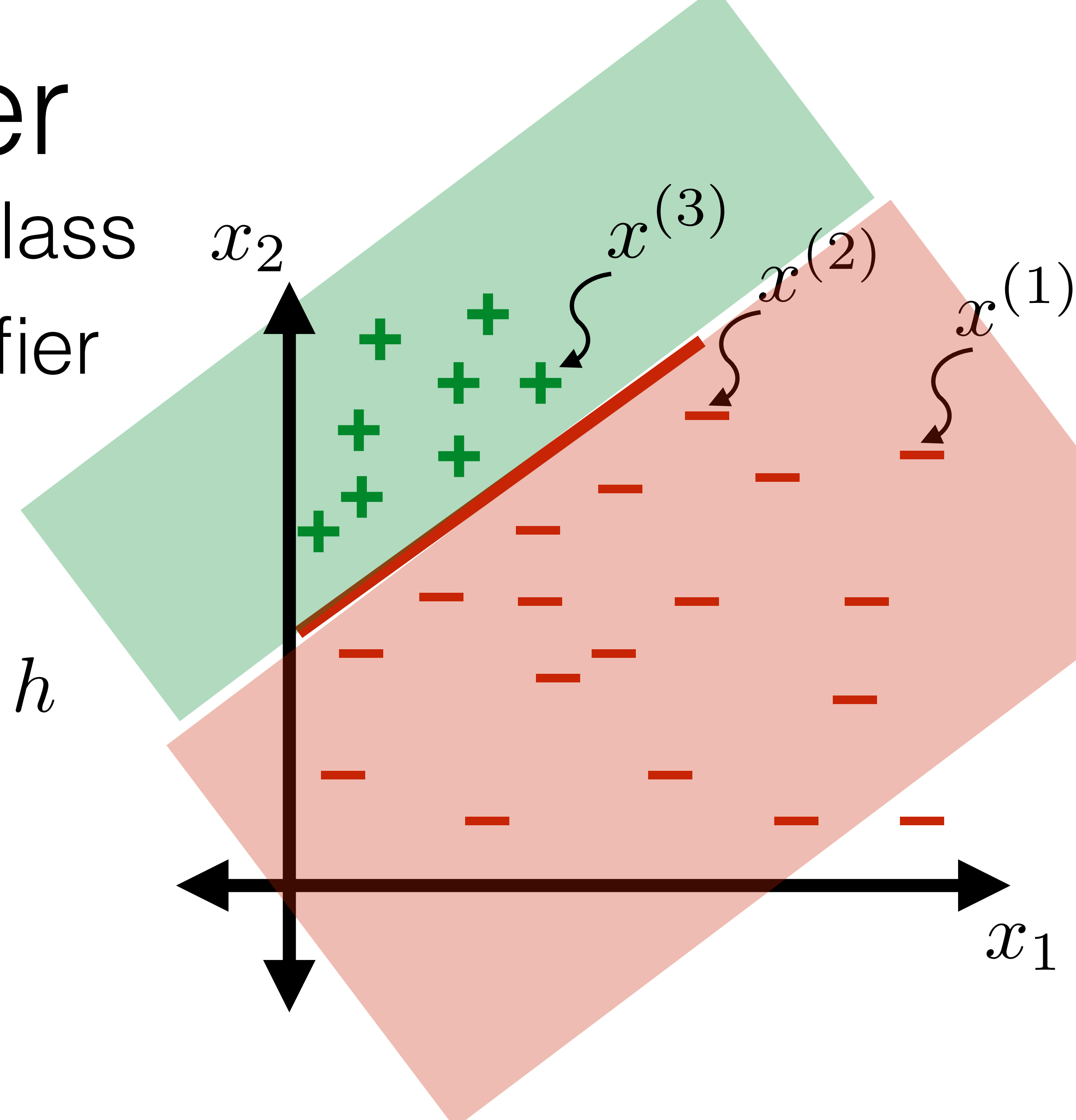
Learning a classifier

- Have data; have hypothesis class
- Want to choose a good classifier
  - Recall: $x \rightarrow \boxed{h} \rightarrow y$
  - New: $\mathcal{D}_n \rightarrow \boxed{\text{learning algorithm}} \rightarrow h$



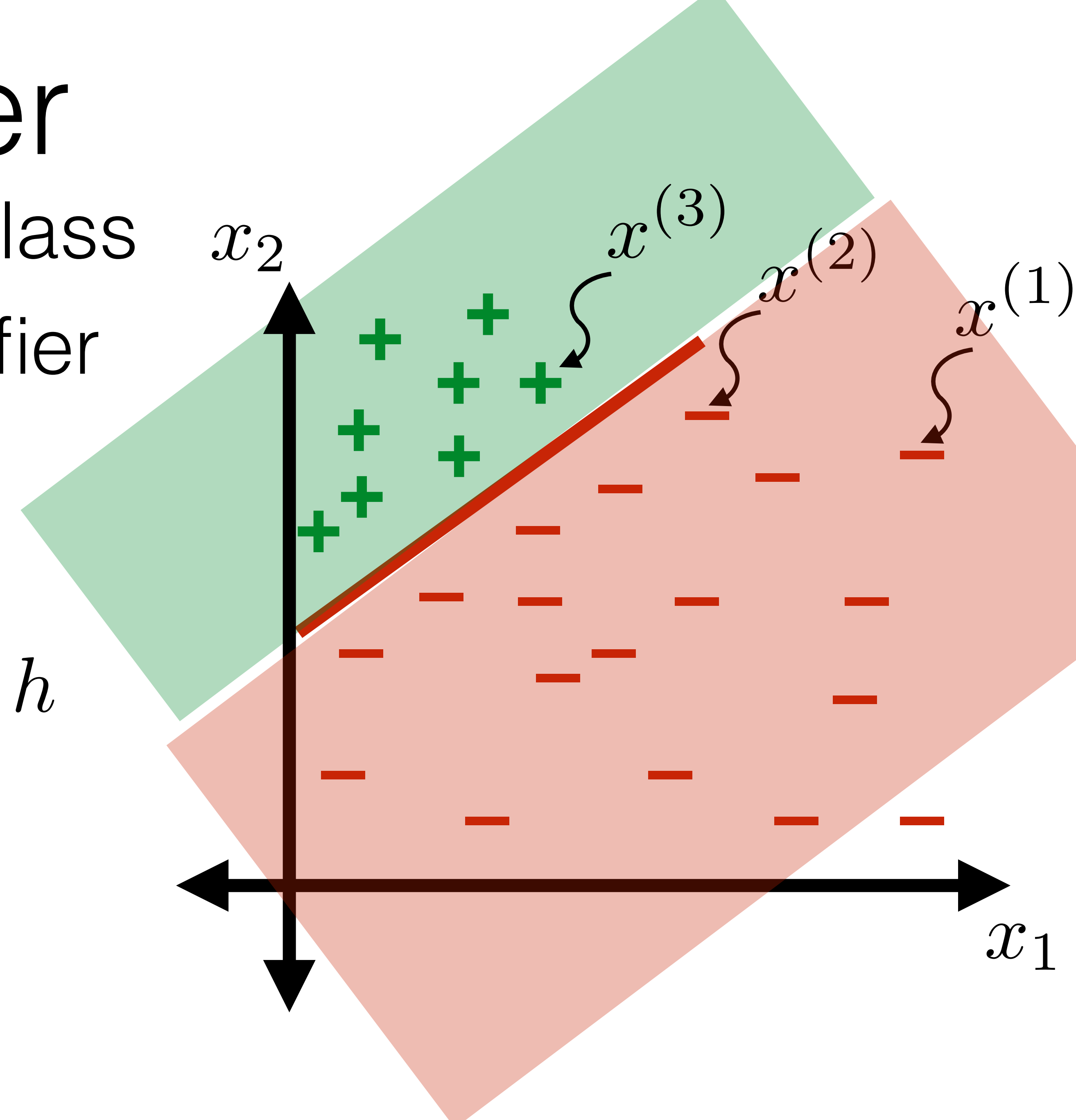
Learning a classifier

- Have data; have hypothesis class
- Want to choose a good classifier
    - Recall: $x \rightarrow \boxed{h} \rightarrow y$
    - New: $\mathcal{D}_n \rightarrow \boxed{\text{learning algorithm}} \rightarrow h$
- Example:



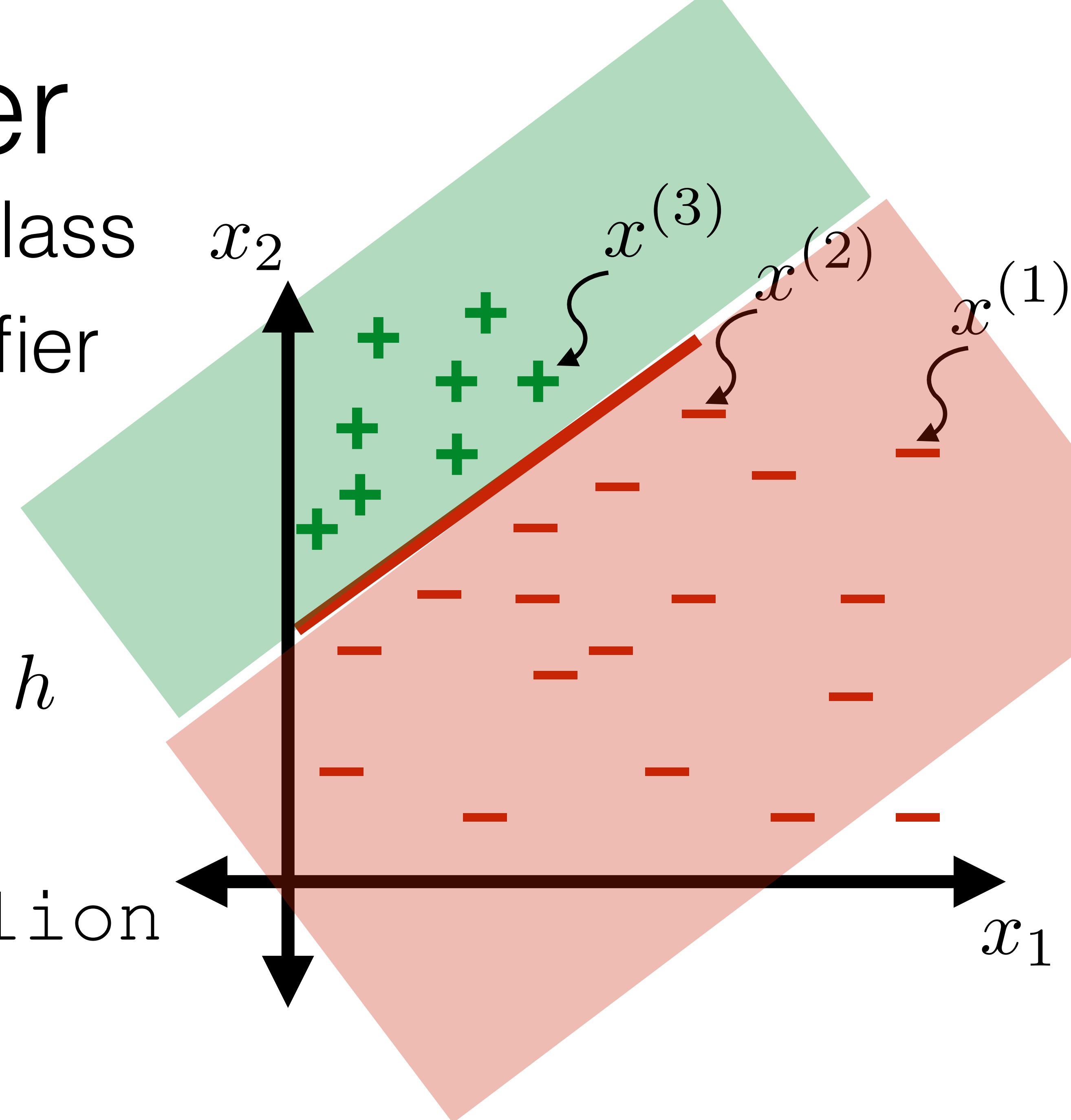
Learning a classifier

- Have data; have hypothesis class
- Want to choose a good classifier
  - Recall: $x \rightarrow \boxed{h} \rightarrow y$
  - New: $\mathcal{D}_n \rightarrow \boxed{\text{learning algorithm}} \rightarrow h$
- Example:



Learning a classifier

- Have data; have hypothesis class
- Want to choose a good classifier
  - Recall: $x \rightarrow \boxed{h} \rightarrow y$
  - New: $\mathcal{D}_n \rightarrow \boxed{\text{learning algorithm}} \rightarrow h$



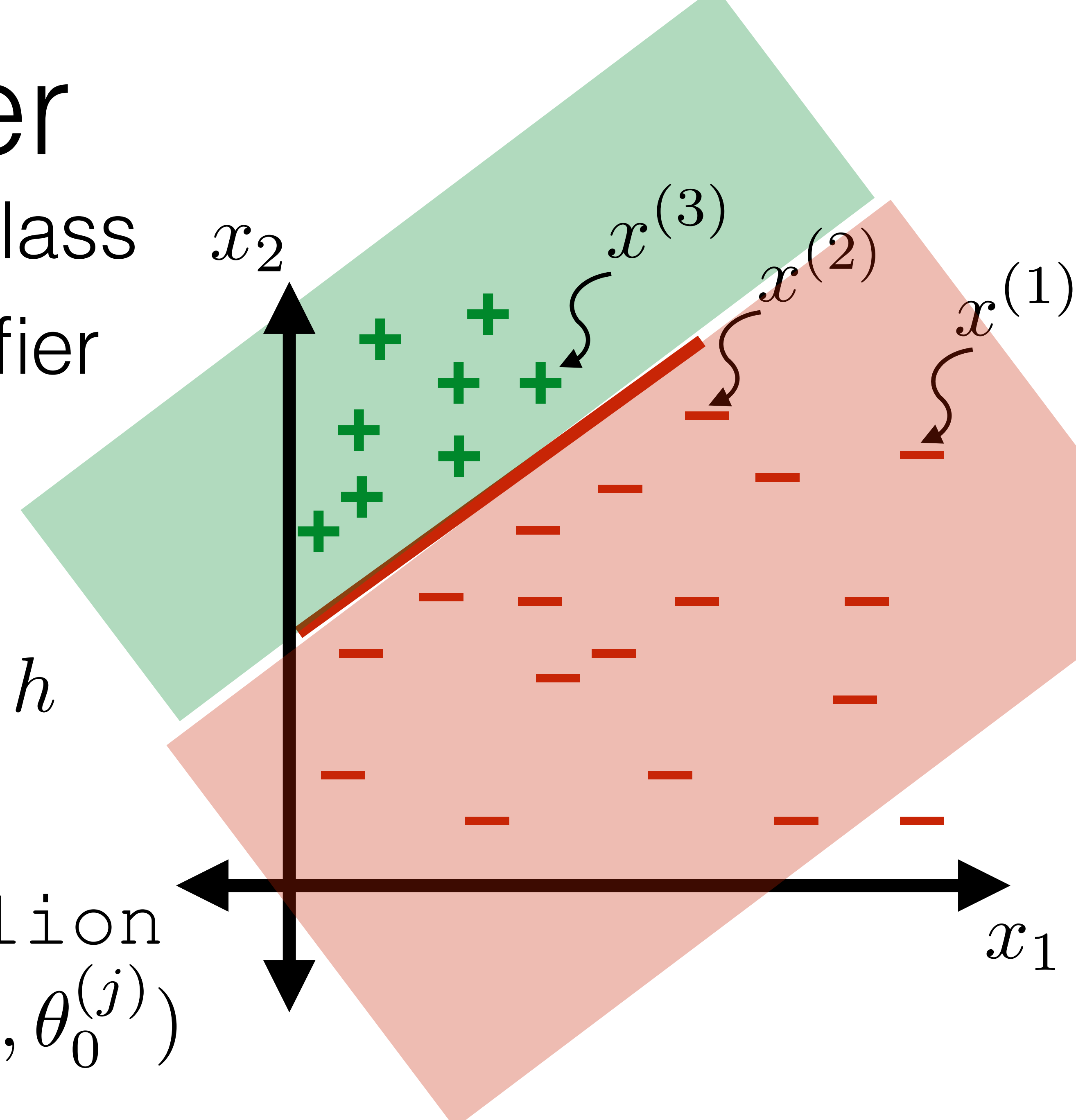
- Example:

```
for j = 1, …, 1 trillion
```

Learning a classifier

- Have data; have hypothesis class
- Want to choose a good classifier
  - Recall: $x \rightarrow \boxed{h} \rightarrow y$
  - New: $\mathcal{D}_n \rightarrow \boxed{\text{learning algorithm}} \rightarrow h$
- Example:

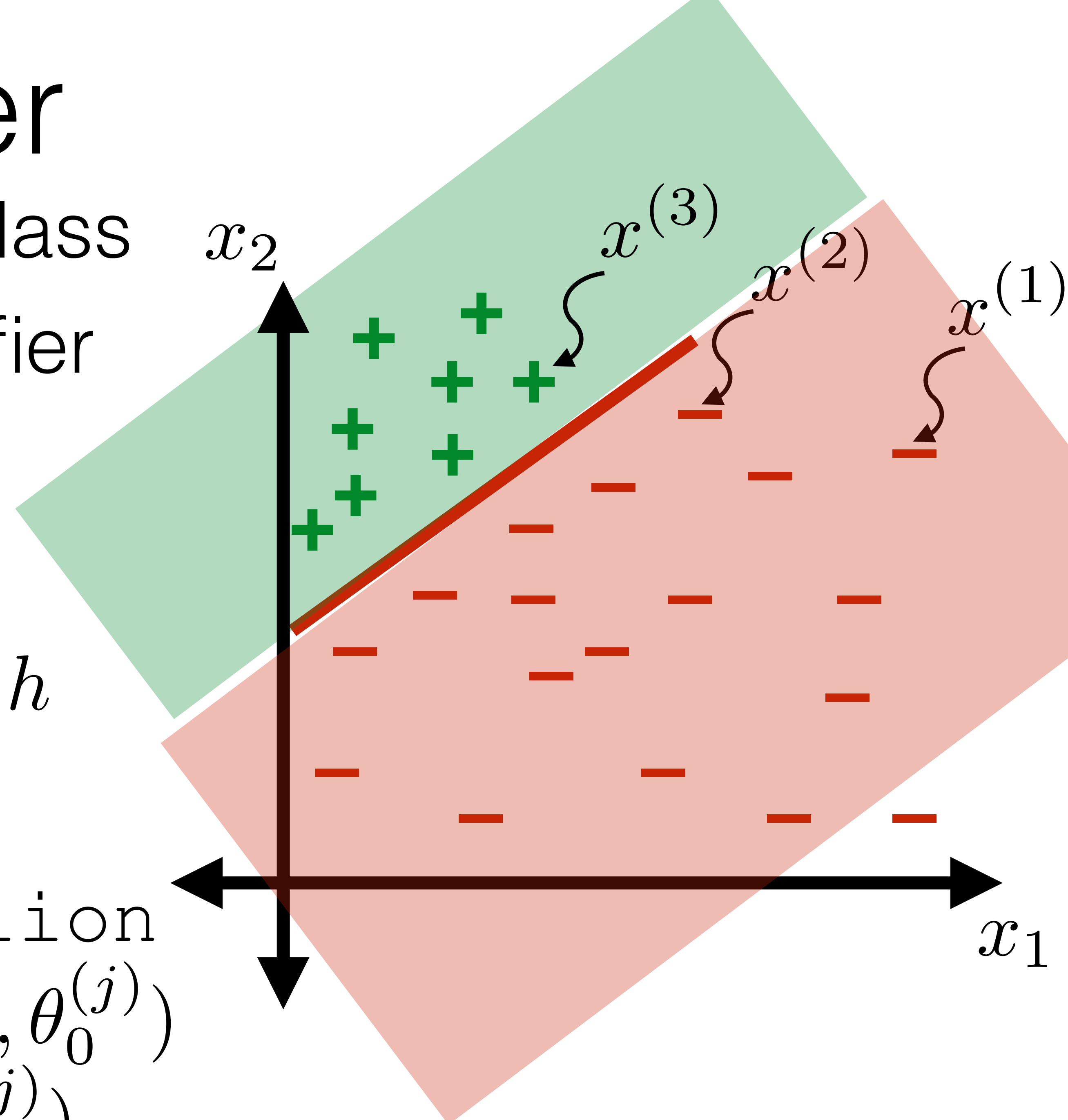
```
for j = 1, …, 1 trillion
  Randomly sample (θ^(j), θ_0^(j))
```



Learning a classifier

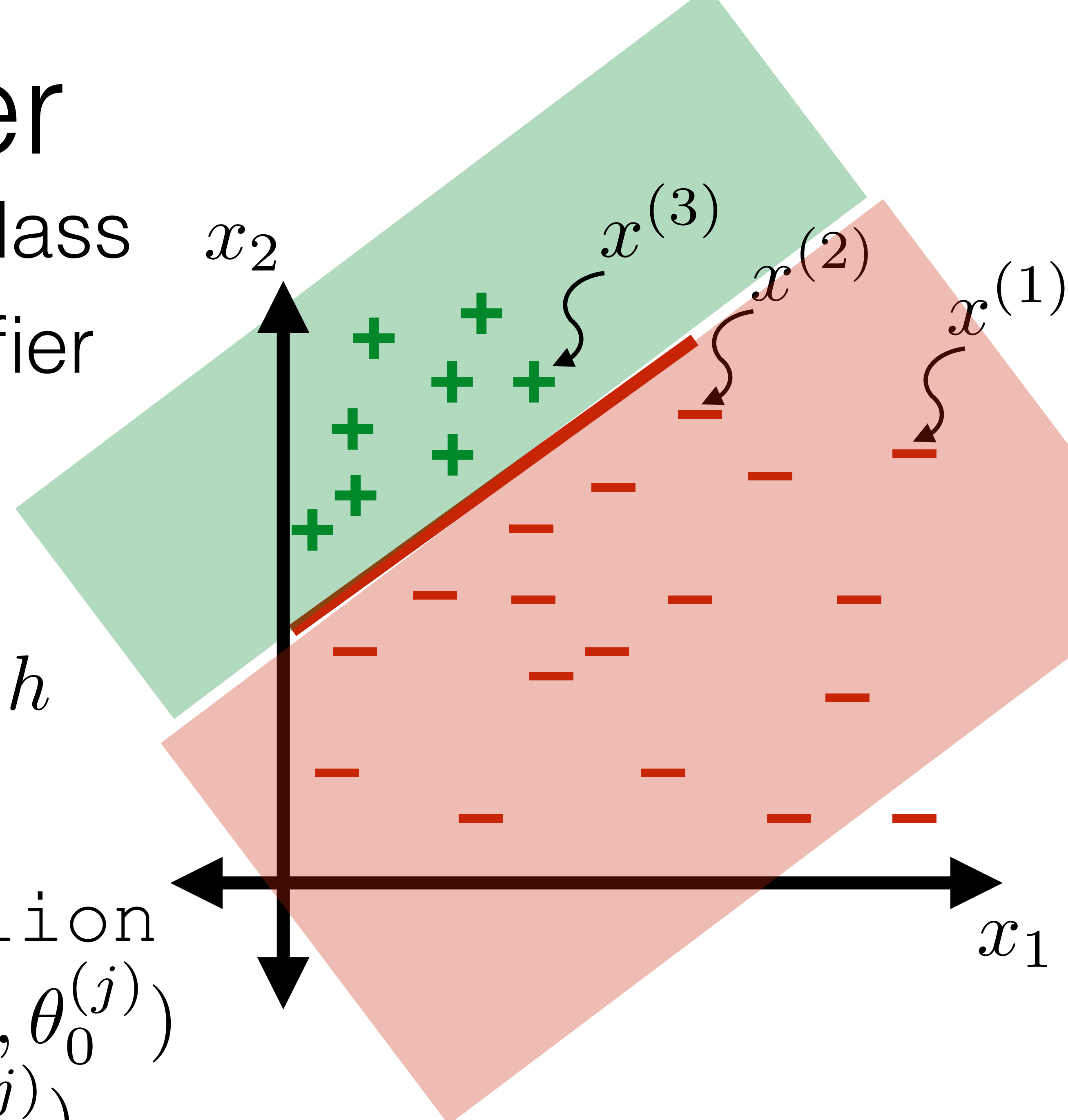
- Have data; have hypothesis class
- Want to choose a good classifier
    - Recall: $x \rightarrow \boxed{h} \rightarrow y$
    - New: $\mathcal{D}_n \rightarrow \boxed{\text{learning algorithm}} \rightarrow h$
- Example:

```
for j = 1, …, 1 trillion
    Randomly sample (θ^(j), θ_0^(j))
    Set h^(j)(x) = h(x; θ^(j), θ_0^(j))
```



Learning a classifier

- Have data; have hypothesis class
- Want to choose a good classifier
  - Recall: $x \rightarrow \boxed{h} \rightarrow y$
  - New: $\mathcal{D}_n \rightarrow$ learning algorithm $\rightarrow h$

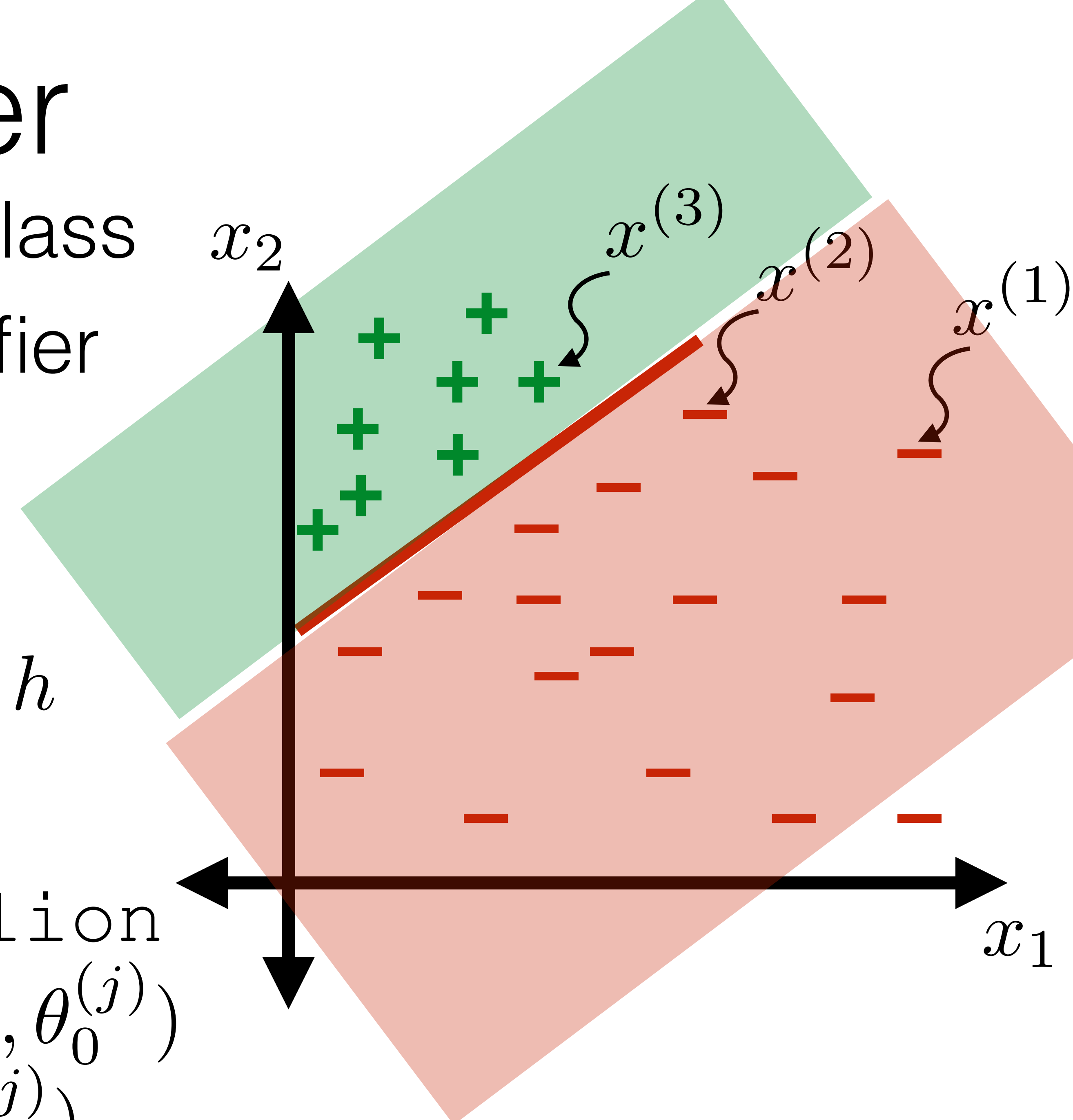


- Example:

for j = 1, …, 1 trillion
  Randomly sample $(\theta^{(j)}, \theta_0^{(j)})$
  Set $h^{(j)}(x) = h(x; \theta^{(j)}, \theta_0^{(j)})$

Learning a classifier

- Have data; have hypothesis class
- Want to choose a good classifier
    - Recall: $x \rightarrow \boxed{h} \rightarrow y$
    - New: $\mathcal{D}_n \rightarrow \boxed{\text{learning algorithm}} \rightarrow h$



- Example:

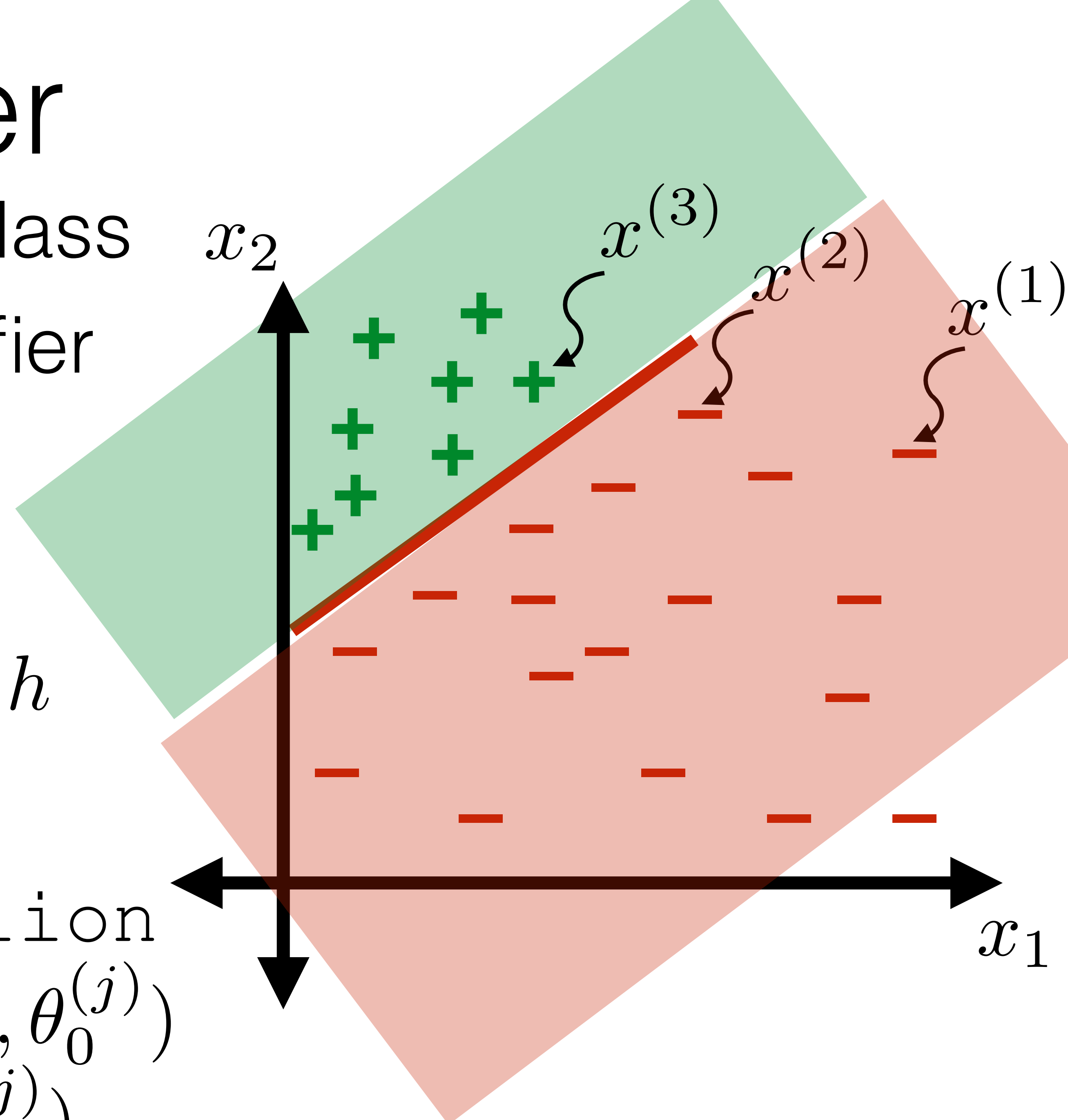
```
for j = 1, …, 1 trillion
  Randomly sample (θ^(j), θ_0^(j))
  Set h^(j)(x) = h(x; θ^(j), θ_0^(j))
Ex_learning_alg
```

Randomly sample $(\theta^{(j)}, \theta_0^{(j)})$

Set $h^{(j)}(x) = h(x; \theta^{(j)}, \theta_0^{(j)})$

Learning a classifier

- Have data; have hypothesis class
- Want to choose a good classifier
  - Recall: $x \rightarrow \boxed{h} \rightarrow y$
  - New: $\mathcal{D}_n \rightarrow \boxed{\text{learning algorithm}} \rightarrow h$

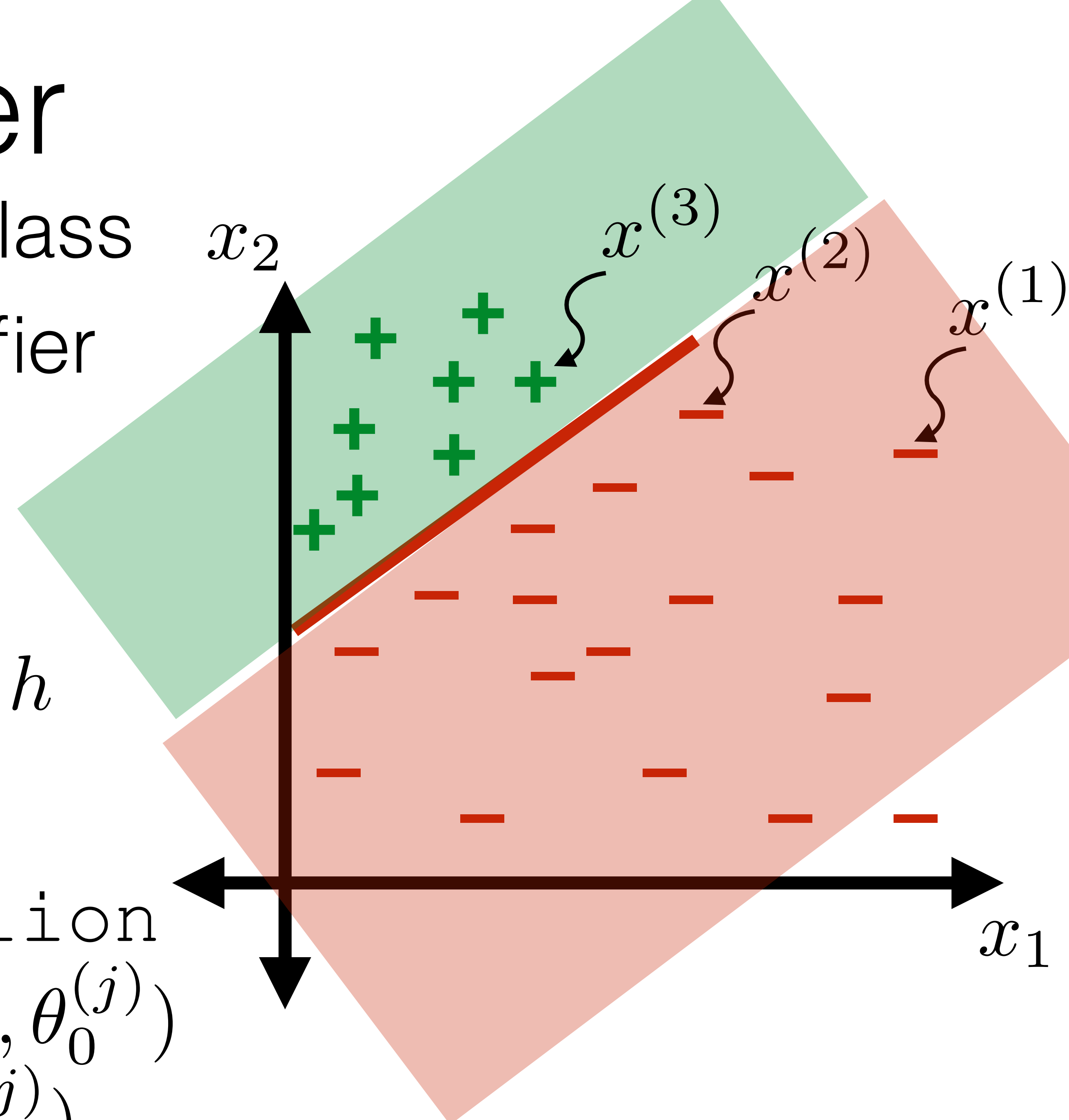


- Example:

for j = 1, …, 1 trillion
  Randomly sample $(\theta^{(j)}, \theta_0^{(j)})$
  Set $h^{(j)}(x) = h(x; \theta^{(j)}, \theta_0^{(j)})$
Ex_learning_alg($\mathcal{D}_n$)

Learning a classifier

- Have data; have hypothesis class
- Want to choose a good classifier
  - Recall: $x \rightarrow \boxed{h} \rightarrow y$
  - New: $\mathcal{D}_n \rightarrow \boxed{\text{learning algorithm}} \rightarrow h$

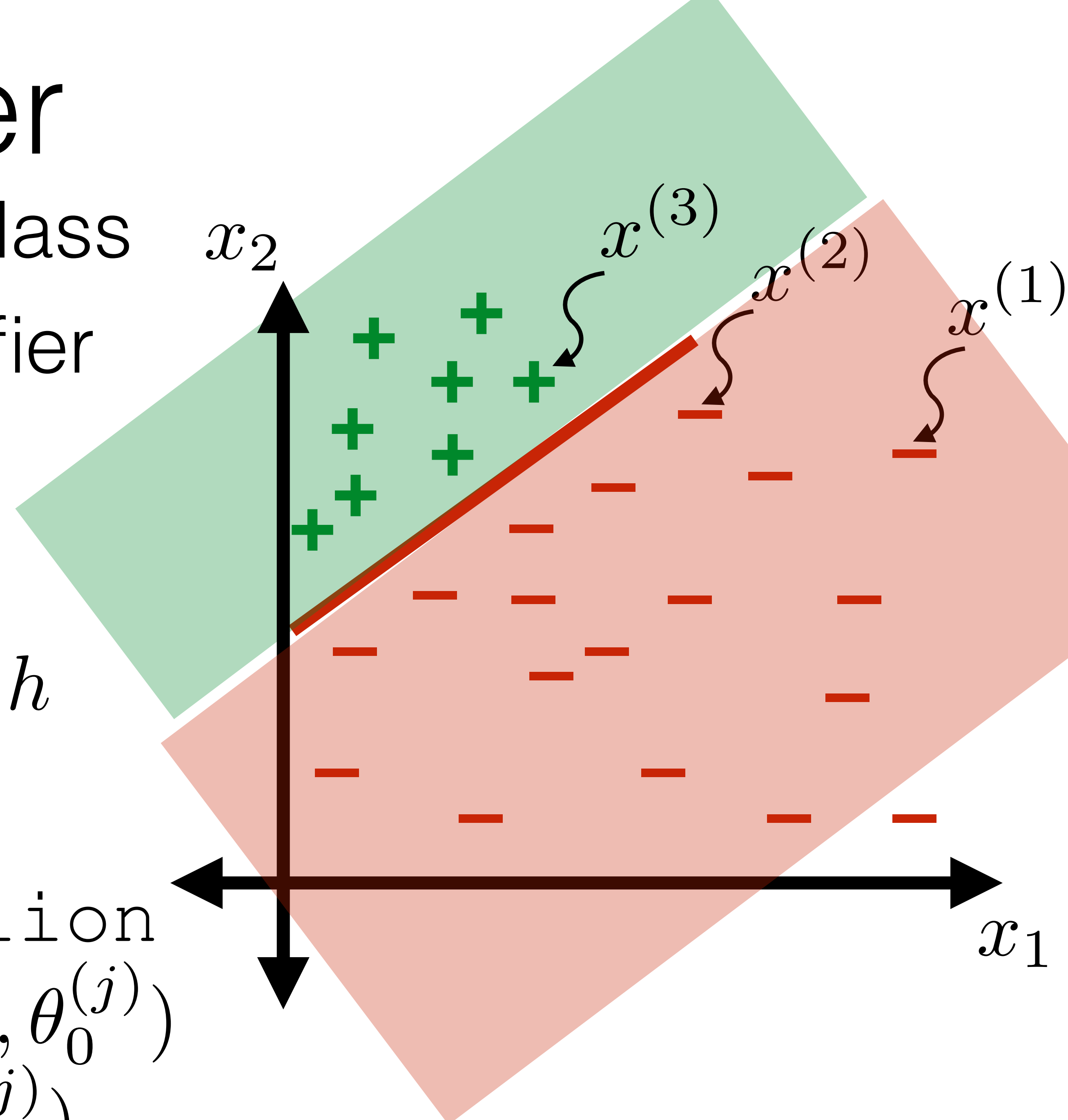


- Example:

for j = 1, …, 1 trillion
  Randomly sample $(\theta^{(j)}, \theta_0^{(j)})$
  Set $h^{(j)}(x) = h(x; \theta^{(j)}, \theta_0^{(j)})$
Ex_learning_alg($\mathcal{D}_n$; k < 1 trillion)

Learning a classifier

- Have data; have hypothesis class
- Want to choose a good classifier
  - Recall: $x \rightarrow \boxed{h} \rightarrow y$
  - New: $\mathcal{D}_n \rightarrow$ learning algorithm $\rightarrow h$



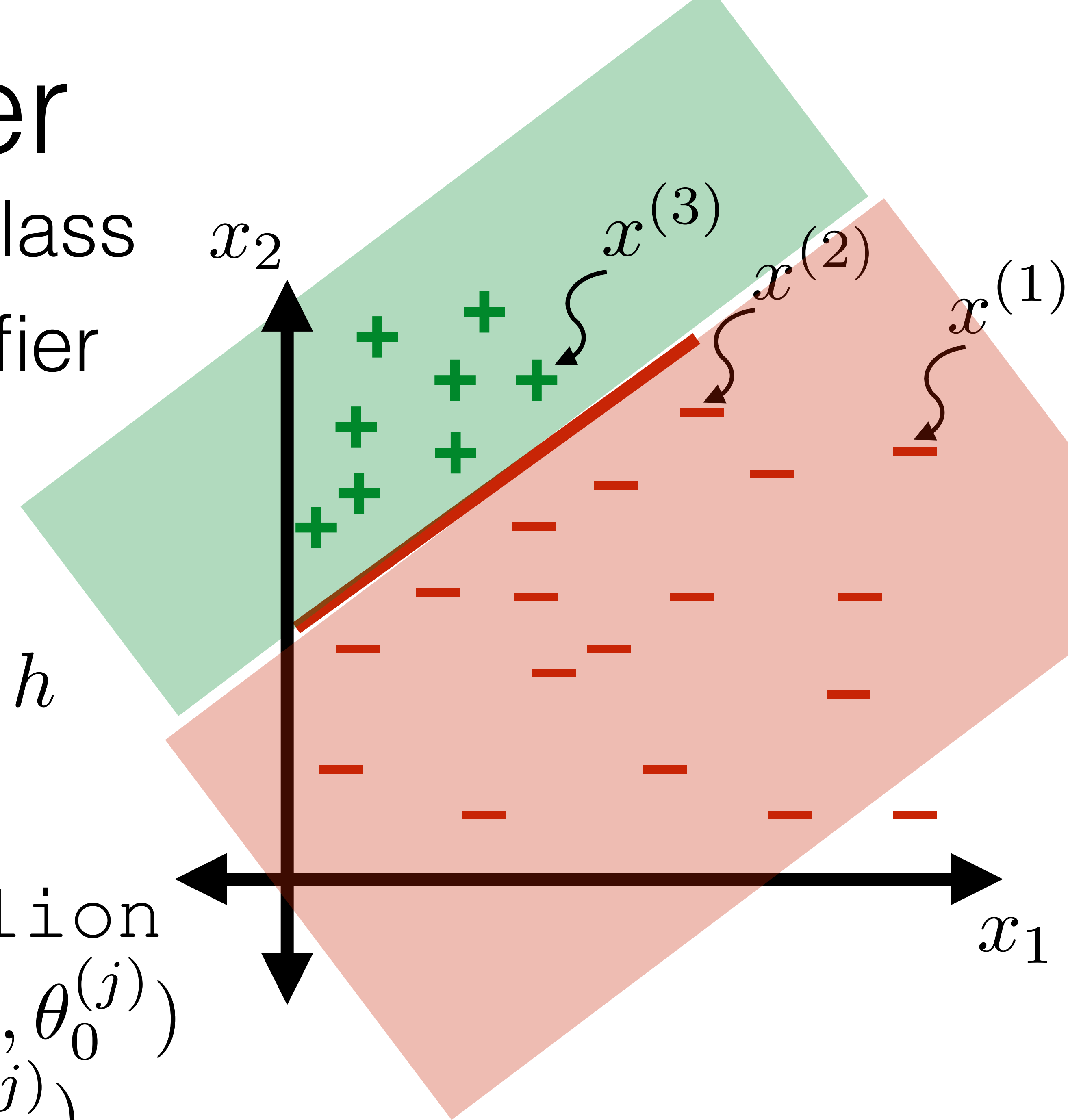
- Example:

for j = 1, …, 1 trillion
  Randomly sample $(\theta^{(j)}, \theta_0^{(j)})$
  Set $h^{(j)}(x) = h(x; \theta^{(j)}, \theta_0^{(j)})$
Ex_learning_alg($\mathcal{D}_n$; k < 1 trillion)

hyperparameter

Learning a classifier

- Have data; have hypothesis class
- Want to choose a good classifier
  - Recall: $x \rightarrow \boxed{h} \rightarrow y$
  - New: $\mathcal{D}_n \rightarrow$ learning algorithm $\rightarrow h$



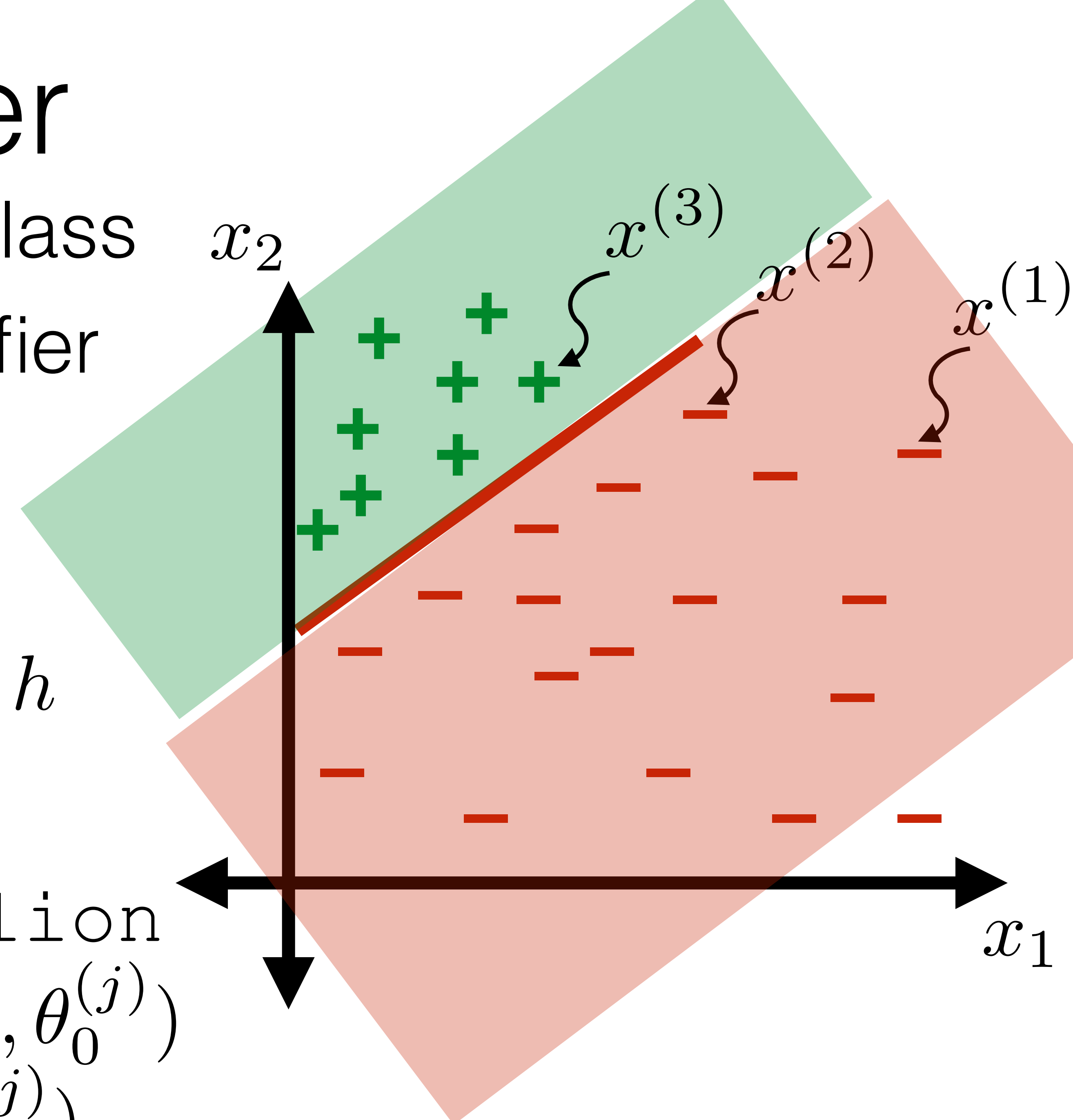
- Example:

```
for j = 1, …, 1 trillion
  Randomly sample (θ^(j), θ_0^(j))
  Set h^(j)(x) = h(x; θ^(j), θ_0^(j))
Ex_learning_alg( D_n ; k < 1 trillion)
  Set j* = argmin_{j∈{1,...,k}} E_n(h^(j))
```

hyperparameter

Learning a classifier

- Have data; have hypothesis class
- Want to choose a good classifier
  - Recall: $x \rightarrow \boxed{h} \rightarrow y$
  - New:
    $\mathcal{D}_n \rightarrow$ learning algorithm $\rightarrow h$



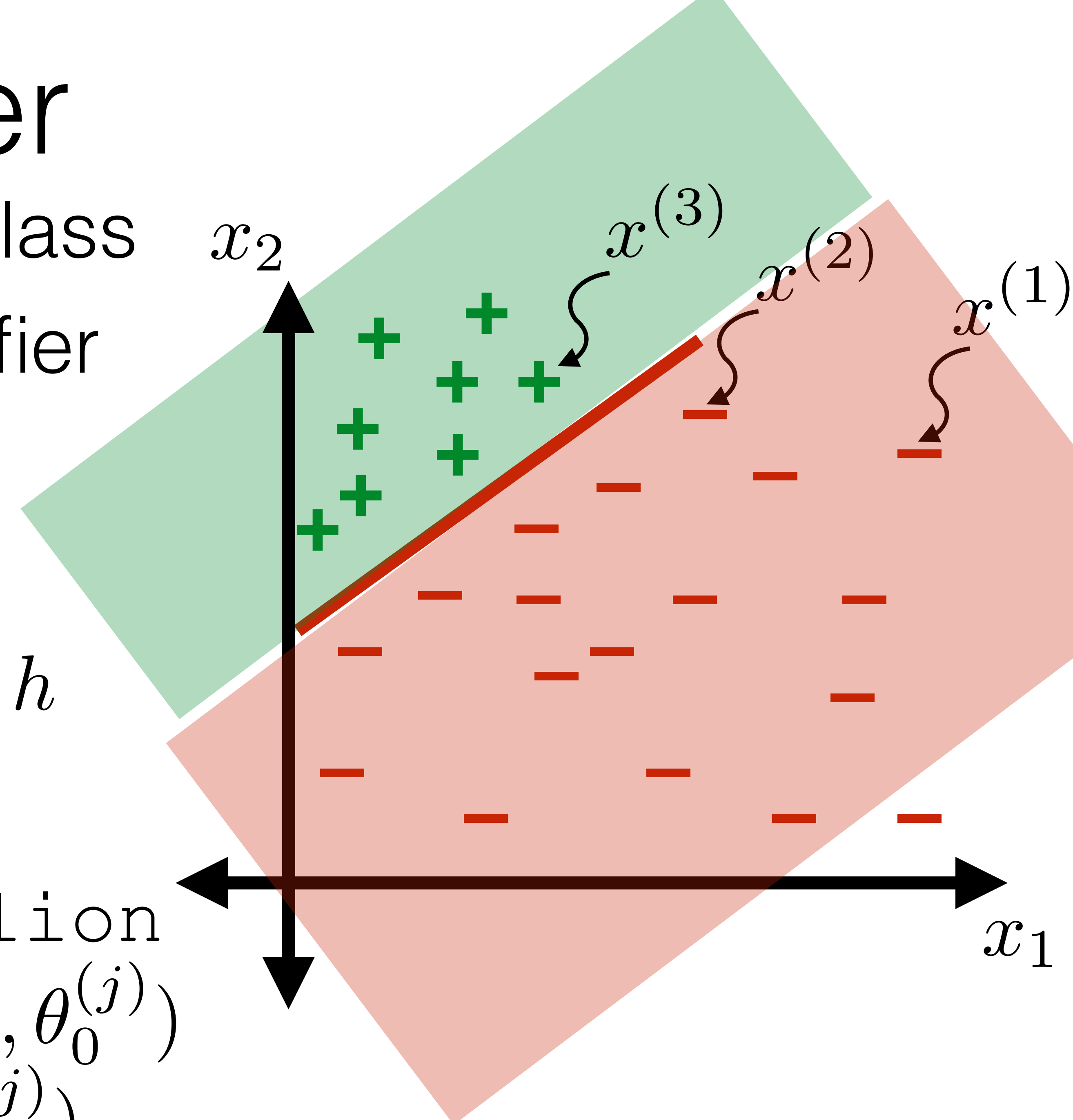
- Example:

```
for j = 1, …, 1 trillion
   Randomly sample (θ^(j), θ_0^(j))
   Set h^(j)(x) = h(x; θ^(j), θ_0^(j))
Ex_learning_alg( D_n ; k < 1 trillion)
   Set j* = argmin_{j∈{1,…,k}} E_n(h^(j))
   Return h^(j*)
```

hyperparameter

Learning a classifier

- Have data; have hypothesis class
- Want to choose a good classifier
  - Recall: $x \rightarrow \boxed{h} \rightarrow y$
  - New: $\mathcal{D}_n \rightarrow$ learning algorithm $\rightarrow h$



- Example:

    `for j = 1, …, 1 trillion`
        `Randomly sample` $(\theta^{(j)}, \theta_0^{(j)})$
        `Set` $h^{(j)}(x) = h(x; \theta^{(j)}, \theta_0^{(j)})$
    `Ex_learning_alg(` $\mathcal{D}_n$ `;` k `< 1 trillion)`
        `Set` $j^* = \text{argmin}_{j \in \{1,\ldots,k\}} \mathcal{E}_n(h^{(j)})$
        `Return` $h^{(j^*)}$

hyperparameter

- How does training error of `Ex_learning_alg(`$\mathcal{D}_n$;1) compare to the training error of `Ex_learning_alg(`$\mathcal{D}_n$;2)?