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
Today's Plan

I. (More) logistics
II. Machine learning setup
III. Linear classifiers
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.)
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:

Jehangir Amjad
Duane Boning
Tamara Broderick
Ike Chuang
Iddo Drori
Phillip Isola
David Sontag
6.036/6.862: Introduction to Machine Learning, Staff

Instructors:

Jehangir Amjad
Duane Boning
Tamara Broderick
Ike Chuang
Iddo Drori
Phillip Isola
David Sontag

Teaching Assistants:

Matt Beveridge
Satvat Jagwani
Dheekshita Kumar
Justin Lim
Caleb Noble
Hye young Shin
Peter Tran
Julie Vaughn
Audrey Wang
Crystal Wang
Quentin Wellens
Julia Wu
Emily Zhang
6.036/6.862: Introduction to Machine Learning, Staff

Instructors:
Jehangir Amjad
Duane Boning
Tamara Broderick
Ike Chuang
Iddo Drori
Phillip Isola
David Sontag

Teaching Assistants:
Matt Beveridge
Satvat Jagwani
Dheekshita Kumar
Justin Lim
Caleb Noble
Hye young Shin
Peter Tran
Julie Vaughn
Audrey Wang
Crystal Wang
Quentin Wellens
Julia Wu
Emily Zhang

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
Complete/update by noon today!
Welcome to 6.036

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

Complete/update by noon today!

- Lecture + course notes
6.036/6.862: Introduction to Machine Learning, Weekly Plan

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

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!)*

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

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

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**

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**

### 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**

- **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

---

### Week 1: Basics

<table>
<thead>
<tr>
<th>Task</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Week 1 Live Lecture</td>
<td></td>
</tr>
<tr>
<td>Introduction to ML</td>
<td></td>
</tr>
<tr>
<td>Linear classifiers</td>
<td></td>
</tr>
<tr>
<td><strong>Week 1 Nanoquiz</strong></td>
<td>NQ due Sep 4, 2020 16:00 EDT</td>
</tr>
<tr>
<td><strong>Week 1 Lab</strong></td>
<td>LAB due Sep 7, 2020 21:00 EDT</td>
</tr>
<tr>
<td><strong>Homework 1</strong></td>
<td>HW due Sep 9, 2020 23:00 EDT</td>
</tr>
</tbody>
</table>
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)

### Week 1: Basics

<table>
<thead>
<tr>
<th>Activity</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Week 1 Live Lecture</strong></td>
<td></td>
</tr>
<tr>
<td><strong>Introduction to ML</strong></td>
<td></td>
</tr>
<tr>
<td><strong>Linear classifiers</strong></td>
<td></td>
</tr>
<tr>
<td><strong>Week 1 Nanoquiz</strong></td>
<td>NQ due Sep 4, 2020 16:00 EDT</td>
</tr>
<tr>
<td><strong>Week 1 Lab</strong></td>
<td>LAB due Sep 7, 2020 21:00 EDT</td>
</tr>
<tr>
<td><strong>Homework 1</strong></td>
<td>HW due Sep 9, 2020 23:00 EDT</td>
</tr>
</tbody>
</table>
Machine learning (ML): why & what
Machine learning (ML): why & what

Machine learning algorithm confirms 50 new exoplanets in historic first

A new machine learning technique can be used to sift through massive datasets to discern exoplanets from false positives.
Machine learning algorithm confirms 50 new exoplanets in historic first

A new machine-learning algorithm based on a million observed stars was used to confirm 50 new exoplanets in a single year, the first time such an approach has been used by the community.

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
Machine learning (ML): why & what
Machine learning (ML): why & what
Machine learning (ML): why & what

5 Ways Machine Learning Can Thwart Phishing Attacks

Louis Columbus Senior Contributor
Enterprise & Cloud
Machine learning (ML): why & what
Machine learning (ML): why & what

ICICI Bank will use satellite images to assess the credit worthiness of farmers.

- ICICI Bank's new machine learning (ML) algorithms use satellite data and images to determine whether a farmer is creditworthy or not.
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.

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
    \[
    x^{(i)} = (x^{(i)}_1, \ldots, x^{(i)}_d)^\top \in \mathbb{R}^d
    \]
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
    $$x^{(i)} = (x^{(i)}_1, \ldots, x^{(i)}_d)^\top \in \mathbb{R}^d$$
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 \( i \in \{1, \ldots, n\} \)
  - Feature vector \( x^{(i)} = (x^{(i)}_1, \ldots, x^{(i)}_d)^\top \in \mathbb{R}^d \)
  - 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
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 $h : \mathbb{R}^d \rightarrow \{-1, +1\}$

$x \xrightarrow{h} y$
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 $h : \mathbb{R}^d \rightarrow \{-1, +1\}$

\[
 x \xrightarrow{h} y
\]
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 $h : \mathbb{R}^d \rightarrow \{-1, +1\}$
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 $h : \mathbb{R}^d \rightarrow \{-1, +1\}$

$$x \xrightarrow{h} y$$
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 \( h : \mathbb{R}^d \rightarrow \{-1, +1\} \)
  - 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 $h : \mathbb{R}^d \rightarrow \{-1, +1\}$
- 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 $x^{(i)} = (x^{(i)}_1, \ldots, x^{(i)}_d) \top \in \mathbb{R}^d$
  - 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 $h : \mathbb{R}^d \rightarrow \{-1, +1\}$
  
  
  \[
  x \rightarrow h \rightarrow y
  \]

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

\[ x_2 \]

\[ x^{(1)} \quad x^{(2)} \quad x^{(3)} \]

\[ x_1 \]
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

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

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

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

Math facts!

\[ \theta^\top x \]

$1 \times d$, $d \times 1$
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

Math facts!

$$\theta^\top x$$

1xd, dx1

$$x_1$$

$$x_2$$

$$x_3$$

$$x_4$$

$$x_5$$

$$x_6$$

$$x_7$$

$$x_8$$

$$x_9$$

$$x_{10}$$
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!

$\theta^\top x$

$1 \times d$, $d \times 1$
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!

$\theta^T x$

$1 \times d$, $d \times 1$
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

$\mathcal{H}$: All hypotheses that label +1 on one side of a line and -1 on the other side.

Math facts!

$x_1 = d, x_1 = dx_1$
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
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!

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

$1x_d, dx_1$
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!

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

$1 \times d$, $d \times 1$
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!

$x_1 + x_2 = \theta^T x / ||\theta||$
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!

$$\theta^T x / \| \theta \|$$

$1xd, dx1$
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!
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

Math facts!

$x : \theta^\top \frac{x}{\|\theta\|} = 0$
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!

$x : \theta^\top x / \|\theta\| = 0$
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!

$x \cdot \theta^T x / \|\theta\| = 0$
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!

$$x : \theta^T \frac{x}{||\theta||} = a$$

$$x : \theta^T \frac{x}{||\theta||} = 0$$
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!

$$x : \theta^T \frac{x}{\|\theta\|} = a$$

$$x : \theta^T \frac{x}{\|\theta\|} = 0$$
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!

$$x : \theta^T x/\|\theta\| = a$$
$$x : \theta^T x/\|\theta\| = 0$$
$$x : \theta^T x/\|\theta\| = -b$$
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!

$$\mathbf{x} \cdot \theta^\top \frac{x}{\|	heta\|} = -b$$
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 $\mathcal{H} : \text{set of } h \in \mathcal{H}$
- 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

Math facts!

$x \cdot \theta^\top x/\|\theta\| > -b$

$x \cdot \theta^\top x/\|\theta\| = -b$

$x \cdot \theta^\top x/\|\theta\| < -b$
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!

$x : \theta^T x / \|\theta\| > -b$

$x : \theta^T x / \|\theta\| = -b$

$x : \theta^T x + b \|\theta\| = 0$
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!

$x_1: \theta^\top \frac{x}{\|\theta\|} > -b$

$x_1: \theta^\top \frac{x}{\|\theta\|} = b$

$x_1: \theta^\top x + b \|\theta\| = 0$

$x_1: \theta^\top x + \theta_0 = 0$
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!

$x : \theta^T x / \|\theta\| > -b$

$x : \theta^T x + \theta_0 = 0$
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:
  
  $$x : \theta^T x / \|\theta\| > -b$$

  $$x : \theta^T x + \theta_0 = 0$$
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:
  $$h(x) = \text{sign}(\theta^T x + \theta_0)$$
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:
  $$h(x) = \text{sign}(\theta^T x + \theta_0)$$
  $$= \begin{cases} 
  +1 & \text{if } \theta^T x + \theta_0 > 0 \\
  -1 & \text{if } \theta^T x + \theta_0 < 0 
  \end{cases}$$
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

Math facts!

- Linear classifier:
  $$ h(x) = \text{sign}(\theta^T x + \theta_0) $$
  $$ = \begin{cases} 
  +1 & \text{if } \theta^T x + \theta_0 > 0 \\
  -1 & \text{if } \theta^T x + \theta_0 \leq 0 
  \end{cases} $$
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

Math facts!

- Linear classifier:
  $$ h(x) = \text{sign}(\theta^T x + \theta_0) $$
  $$ = \begin{cases} 
  +1 \text{ if } \theta^T x + \theta_0 > 0 \\
  -1 \text{ if } \theta^T x + \theta_0 \leq 0 
  \end{cases} $$
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.

Math facts!

- Linear classifier:
\[
\begin{align*}
  h(x) &= \text{sign}(\theta^\top x + \theta_0) \\
  &= \begin{cases} 
    +1 & \text{if } \theta^\top x + \theta_0 > 0 \\
    -1 & \text{if } \theta^\top x + \theta_0 \leq 0
  \end{cases}
\end{align*}
\]
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:
  \[
  h(x) = \text{sign}(\theta^T x + \theta_0)
  \]
  \[
  = \begin{cases} 
  +1 & \text{if } \theta^T x + \theta_0 > 0 \\
  -1 & \text{if } \theta^T x + \theta_0 \leq 0 
  \end{cases}
  \]
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

Math facts!

$\mathbf{x} : \theta^\top \mathbf{x} / \|\theta\| > -b$

$\mathbf{x} : \theta^\top \mathbf{x} + \theta_0 = 0$

$\mathbf{x} : \theta^\top \mathbf{x} / \|\theta\| = b$

- 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

Math facts!

- Linear classifier:
  $$h(x; \theta, \theta_0) = \operatorname{sign}(\theta^\top x + \theta_0)$$
  $$= \begin{cases} 
  +1 & \text{if } \theta^\top x + \theta_0 > 0 \\
  -1 & \text{if } \theta^\top x + \theta_0 \leq 0 
  \end{cases}$$
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

Math facts!

Linear classifier:

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

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

**Math facts!**

- Linear classifier:
  $$h(x; \theta, \theta_0) = \text{sign}(\theta^T x + \theta_0)$$
  $$= \begin{cases} 
  +1 & \text{if } \theta^T x + \theta_0 > 0 \\
  -1 & \text{if } \theta^T x + \theta_0 \leq 0 
  \end{cases}$$

$\mathcal{H} = \text{set of all such } h$
How good is a classifier?
How good is a classifier?

- Should predict well on future data

![Diagram showing classification of data points in a two-dimensional space with a decision boundary.](image)
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 $L(g, a)$
  - Example: 0-1 loss

\[ g: \text{guess, } a: \text{actual} \]
How good is a classifier?

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

  - 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 $L(g, a)$
  
  • 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 $L(g, a)$
  - 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 $L(g, a)$
  - 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 $L(g, a)$
  - 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 \( L(g, a) \)
  - 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 $L(g, a)$
  - 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 \( L(g, a) \)
  
  • 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: \( x \rightarrow h \rightarrow y \)
Learning a classifier
• Have data; have hypothesis class
• Want to choose a good classifier
• Recall: $x \rightarrow h \rightarrow y$
Learning a classifier

- Have data; have hypothesis class
- Want to choose a good classifier

- Recall: \( x \rightarrow h \rightarrow y \)
Learning a classifier

• Have data; have hypothesis class
• Want to choose a good classifier

• Recall: $x \rightarrow h \rightarrow y$
Learning a classifier

• Have data; have hypothesis class
• Want to choose a good classifier

• Recall: $x \rightarrow h \rightarrow y$

• New:
Learning a classifier

• Have data; have hypothesis class
• Want to choose a good classifier

• Recall: $x \rightarrow h \rightarrow y$
• New: $\mathcal{D}_n \rightarrow \text{learning algorithm} \rightarrow h$
Learning a classifier

• Have data; have hypothesis class
• Want to choose a good classifier

• Recall: $x \rightarrow h \rightarrow y$

• New: $\mathcal{D}_n \rightarrow \text{learning algorithm} \rightarrow h$

\[ x^{(1)} \quad x^{(2)} \quad x^{(3)} \]
Learning a classifier

- Have data; have hypothesis class
- Want to choose a good classifier

- Recall: $x \rightarrow h \rightarrow y$
- New: $D_n \rightarrow \text{learning algorithm} \rightarrow h$
Learning a classifier

- Have data; have hypothesis class
- Want to choose a good classifier

- Recall: $x \rightarrow h \rightarrow y$
- New: $D_n \rightarrow \text{learning algorithm} \rightarrow h$
Learning a classifier

- Have data; have hypothesis class
- Want to choose a good classifier

Recall: $x \rightarrow h \rightarrow y$

New: $\mathcal{D}_n \rightarrow \text{learning algorithm} \rightarrow h$
Learning a classifier

• Have data; have hypothesis class
• Want to choose a good classifier

• Recall: \( x \rightarrow h \rightarrow y \)
• New: \( D_n \rightarrow \text{learning algorithm} \rightarrow h \)
Learning a classifier

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

- Have data; have hypothesis class
- Want to choose a good classifier
- Recall: $x \rightarrow h \rightarrow y$
- New: $D_n \rightarrow$ learning algorithm $h$
- Example:
Learning a classifier

- Have data; have hypothesis class
- Want to choose a good classifier
  - Recall: $x \rightarrow h \rightarrow y$
  - New: $D_n \rightarrow$ learning algorithm $\rightarrow h$
  - Example:
    
    for $j = 1, \ldots, 1$ trillion
Learning a classifier

- Have data; have hypothesis class
- Want to choose a good classifier
  - Recall: \( x \rightarrow h \rightarrow y \)
  - New: \( \mathcal{D}_n \rightarrow \text{learning algorithm} \rightarrow h \)
  - Example:
    
    \[
    \text{for } j = 1, \ldots, 1 \text{ trillion} \\
    \text{Randomly sample } (\theta^{(j)}, \theta_0^{(j)})
    \]
Learning a classifier

- Have data; have hypothesis class
- Want to choose a good classifier

- Recall: \( x \rightarrow h \rightarrow y \)
- New: \( D_n \rightarrow \text{learning algorithm} \rightarrow h \)

- Example:

  for \( j = 1, \ldots, 1 \text{ trillion} \)
  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: $x \xrightarrow{h} y$
- New: $\mathcal{D}_n \xrightarrow{\text{learning algorithm}} h$
- Example:

\[
\text{for } j = 1, \ldots, 1 \text{ trillion} \\
\text{Randomly sample } (\theta^{(j)}, \theta_0^{(j)}) \\
\text{Set } h^{(j)}(x) = h(x; \theta^{(j)}, \theta_0^{(j)})
\]
Learning a classifier

- Have data; have hypothesis class
- Want to choose a good classifier

- Recall: $x \rightarrow h \rightarrow y$

- New: $D_n \rightarrow$ learning algorithm $\rightarrow h$

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

Ex_learning_alg
Learning a classifier

- Have data; have hypothesis class
- Want to choose a good classifier

- Recall: $x \rightarrow h \rightarrow y$
- New: $\mathcal{D}_n \rightarrow \text{learning algorithm} \rightarrow h$

- Example:
  
  for $j = 1, \ldots, 1$ 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: \( x \rightarrow h \rightarrow y \)

- New: \( D_n \rightarrow \text{learning algorithm} \rightarrow h \)

- Example:

\[
\begin{align*}
\text{for } j &= 1, \ldots, 1 \text{ trillion} \\
\text{Randomly sample } (\theta^{(j)}, \theta_0^{(j)}) \\
\text{Set } h^{(j)}(x) &= h(x; \theta^{(j)}, \theta_0^{(j)}) \\
\text{Ex_learning_alg}( D_n ; k < 1 \text{ trillion})
\end{align*}
\]
Learning a classifier

- Have data; have hypothesis class
- Want to choose a good classifier

- Recall: $x \rightarrow h \rightarrow y$
- New: $\mathcal{D}_n \rightarrow \text{learning algorithm} \rightarrow h$

- Example:
  
  for $j = 1, \ldots, 1$ trillion
  
  Randomly sample $(\theta^{(j)}, \theta_0^{(j)})$
  
  Set $h^{(j)}(x) = h(x; \theta^{(j)}, \theta_0^{(j)})$
  
  Ex_learning_alg($\mathcal{D}_n; k \leq 1$ trillion)
Learning a classifier
• Have data; have hypothesis class
• Want to choose a good classifier
• Recall: \( x \rightarrow h \rightarrow y \)
• New: \( \mathcal{D}_n \rightarrow \text{learning algorithm} \rightarrow h \)
• Example:
  
  for \( j = 1, \ldots, 1 \) trillion
  
  Randomly sample \( (\theta^{(j)}, \theta_0^{(j)}) \)
  
  Set \( h^{(j)}(x) = h(x; \theta^{(j)}, \theta_0^{(j)}) \)
  
  \text{Ex\_learning\_alg}( \mathcal{D}_n; k \ll 1 \) trillion
  
  Set \( j^* = \arg\min_{j \in \{1,\ldots,k\}} \mathcal{E}_n(h^{(j)}) \)
Learning a classifier

• Have data; have hypothesis class
• Want to choose a good classifier
  • Recall: $x \rightarrow h \rightarrow y$
  • New: $D_n \rightarrow \text{learning algorithm} \rightarrow h$
• Example:
  
  for $j = 1, \ldots, 1$ trillion
  
  Randomly sample $(\theta^{(j)}, \theta_0^{(j)})$
  
  Set $h^{(j)}(x) = h(x; \theta^{(j)}, \theta_0^{(j)})$
  
  Ex\_learning\_alg($D_n; k \ll 1$ trillion)

  Set $j^* = \arg\min_{j \in \{1, \ldots, k\}} \mathcal{E}_n(h^{(j)})$

  Return $h^{(j^*)}$
Learning a classifier

- Have data; have hypothesis class
- Want to choose a good classifier

Recall: \( x \rightarrow h \rightarrow y \)

New: \( \mathcal{D}_n \rightarrow \text{learning algorithm} \rightarrow h \)

Example:

\[
\text{for } j = 1, \ldots, 1 \text{ trillion} \\
\text{Randomly sample } (\theta^{(j)}, \theta_0^{(j)}) \\
\text{Set } h^{(j)}(x) = h(x; \theta^{(j)}, \theta_0^{(j)}) \\
\text{Ex\textunderscore learning\textunderscore alg}\left( \mathcal{D}_n; k < 1 \text{ trillion} \right) \\
\text{Set } j^* = \arg\min_{j \in \{1, \ldots, k\}} \mathcal{E}_n(h^{(j)}) \\
\text{Return } h^{(j^*)}
\]

How does training error of \text{Ex\textunderscore learning\textunderscore alg}(\mathcal{D}_n; 1) compare to the training error of \text{Ex\textunderscore learning\textunderscore alg}(\mathcal{D}_n; 2)?