6.7900: Machine Learning
Welcome!

Lecture start: Tues/Thurs 2:35pm

Who's speaking today? Prof. Tamara Broderick

Course website: gradml.mit.edu (includes, e.g., links to Piazza https://piazza.com/mit/fall2024/67900/)

Materials: Slides, video, etc linked from gradml.mit.edu after the lecture (but there is no livestream)

Today's Plan

I. Logistics

II. Machine learning: why & what

III. Getting started: can we solve all of supervised learning in the first lecture or two?

6.7900: Machine Learning

Staff

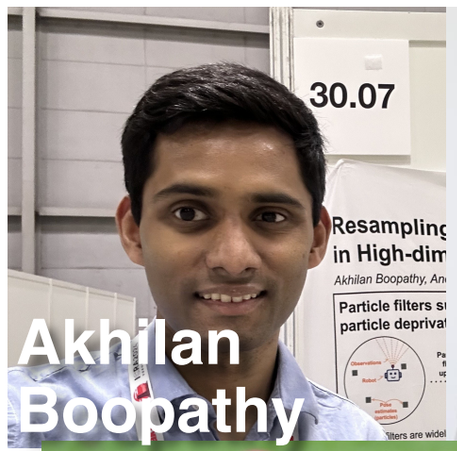
Instructors:

Leslie Kaelbling

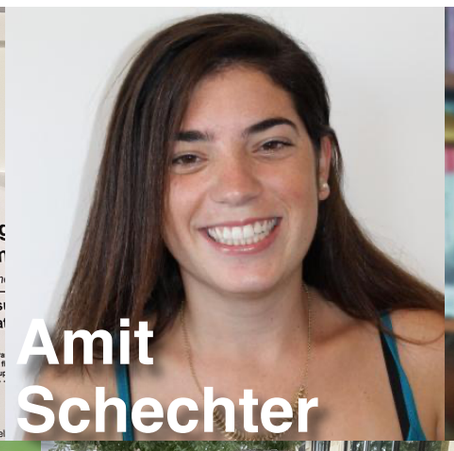
Tamara Broderick

Tommi Jaakkola

Teaching Assistants:

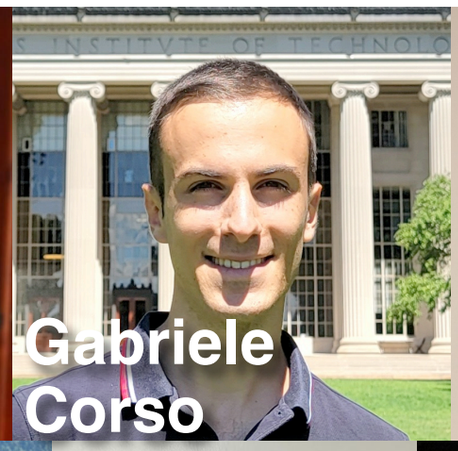


Akhilan Boopathy



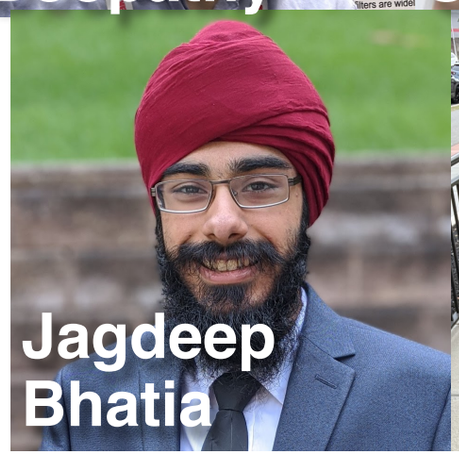
Amit Schechter

Feng Zhu

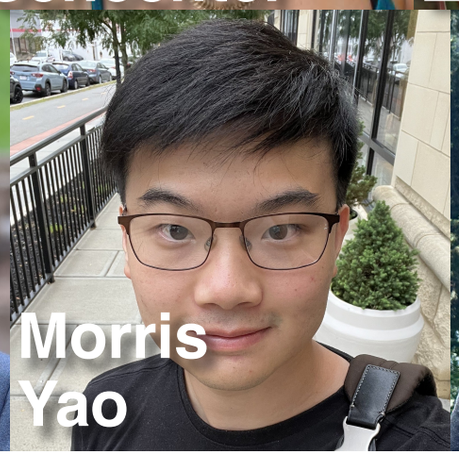


Gabriele Corso

Greg Pylypovych



Jagdeep Bhatia



Morris Yao

Ray Wang

Timmy Qian

Tom Wang

6.7900: Machine Learning
Finding the info you need

6.7900: Machine Learning
Finding the info you need

First, check the course website: gradml.mit.edu

6.7900: Machine Learning
Finding the info you need

First, check the course website: gradml.mit.edu

Questions during lecture:

- Can raise your hand and ask live
- Can post on Piazza. TA(s) will monitor live

6.7900: Machine Learning
Finding the info you need

First, check the course website: gradml.mit.edu

Questions during lecture:

- Can raise your hand and ask live
- Can post on Piazza. TA(s) will monitor live

Technical questions outside of lecture:

- Read/post publicly on Piazza. Office hours starting Sept 12

6.7900: Machine Learning
Finding the info you need

First, check the course website: gradml.mit.edu

Questions during lecture:

- Can raise your hand and ask live
- Can post on Piazza. TA(s) will monitor live

Technical questions outside of lecture:

- Read/post publicly on Piazza. Office hours starting Sept 12

Personal or administrative questions:

- Post privately on Piazza

6.7900: Machine Learning
Finding the info you need

First, check the course website: gradml.mit.edu

Questions during lecture:

- Can raise your hand and ask live
- Can post on Piazza. TA(s) will monitor live

Technical questions outside of lecture:

- Read/post publicly on Piazza. Office hours starting Sept 12

Personal or administrative questions:

- Post privately on Piazza

If you absolutely have to email (e.g. to CC an S^3 Dean):

- Email Prof. Kaelbling (LPK@mit)

6.7900: Machine Learning
Finding the info you need

First, check the course website: gradml.mit.edu

Questions during lecture:

- Can raise your hand and ask live
- Can post on Piazza. TA(s) will monitor live

Technical questions outside of lecture:

- Read/post publicly on Piazza. Office hours starting Sept 12

Personal or administrative questions:

- Post privately on Piazza

If you absolutely have to email (e.g. to CC an S³ Dean):

- Email Prof. Kaelbling (LPK@mit)

Submit homework: Gradescope

Check grades: Canvas

6.7900: Machine Learning
Is this course right for you?

6.7900: Machine Learning
Is this course right for you?

This class dives into the mechanism behind a wide range of modern machine learning tools.

6.7900: Machine Learning
Is this course right for you?

This class dives into the mechanism behind a wide range of modern machine learning tools.

Introductory machine learning

- At the level of 6.390 or 6.370 or 6.C01

Linear algebra

- At the level of 18.06 or 18.C06

Probability

- At the level of 6.3700, 6.3800, or 18.600

6.7900: Machine Learning
Is this course right for you?

This class dives into the mechanism behind a wide range of modern machine learning tools.

Introductory machine learning

- At the level of 6.390 or 6.370 or 6.C01

Linear algebra

- At the level of 18.06 or 18.C06

Probability

- At the level of 6.3700, 6.3800, or 18.600

Homework 0 is a readiness assessment

- Released today (Thursday September 5)
- Due Tuesday September 10

6.7900: Machine Learning
Course schedule

6.7900: Machine Learning
Course schedule

Weekly:

- Two lectures (Tues, Thurs)
- Lecture materials and scribe notes available after lectures
- (Optional) Friday problem sessions, *except* this week (9/6)

6.7900: Machine Learning
Course schedule

Weekly:

- Two lectures (Tues, Thurs)
- Lecture materials and scribe notes available after lectures
- (Optional) Friday problem sessions, *except* this week (9/6)

Six homeworks + one readiness assessment (“HW0”)

- We’re using as teaching tools

6.7900: Machine Learning
Course schedule

Weekly:

- Two lectures (Tues, Thurs)
- Lecture materials and scribe notes available after lectures
- (Optional) Friday problem sessions, *except* this week (9/6)

Six homeworks + one readiness assessment (“HW0”)

- We’re using as teaching tools

Two mini-projects: More of machine learning pipeline

6.7900: Machine Learning
Course schedule

Weekly:

- Two lectures (Tues, Thurs)
- Lecture materials and scribe notes available after lectures
- (Optional) Friday problem sessions, *except* this week (9/6)

Six homeworks + one readiness assessment (“HW0”)

- We’re using as teaching tools

Two mini-projects: More of machine learning pipeline

Exams: We’re using as evaluation tools

- **Midterm**: Th Oct 24, 7–9pm (conflict Fr Oct 25, 8–10am)
  - Contact us in advance if you plan to take the conflict exam
- **Final**: Scheduled by registrar. Make your plans now so that you can be physically present at MIT during finals period.

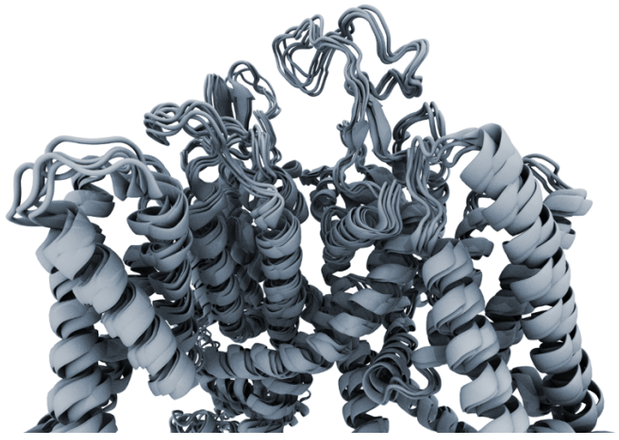
Machine learning (ML): why & what

Machine learning (ML): why & what

ML is increasingly a part of major discoveries and decisions:

Machine learning (ML): why & what

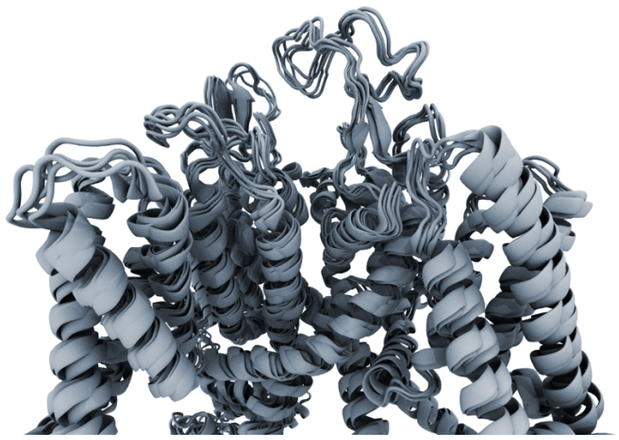
ML is increasingly a part of major discoveries and decisions:



- AlphaFold: big stride in predicting "a protein's 3D shape based solely on the 1D string of molecules that comprise it" [Jumper et al 2021; Toews 2021; Levine, Tu 2024]

Machine learning (ML): why & what

ML is increasingly a part of major discoveries and decisions:

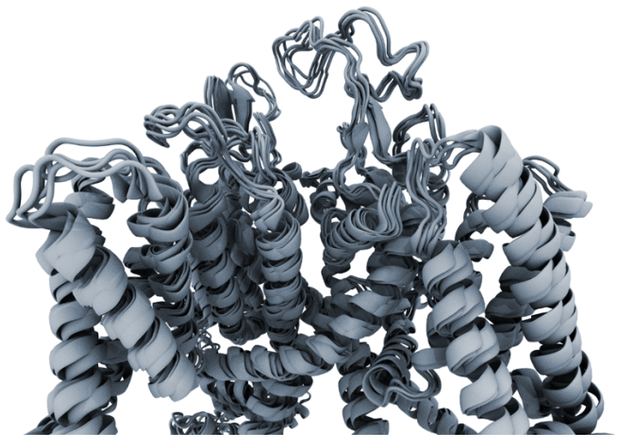


- AlphaFold: big stride in predicting "a protein's 3D shape based solely on the 1D string of molecules that comprise it" [Jumper et al 2021; Toews 2021; Levine, Tu 2024]

- First image of black hole
- First image of black hole at center of Milky Way [Gasparini 2023; Fletcher 2022]

Machine learning (ML): why & what

ML is increasingly a part of major discoveries and decisions:



- AlphaFold: big stride in predicting "a protein's 3D shape based solely on the 1D string of molecules that comprise it" [Jumper et al 2021; Toews 2021; Levine, Tu 2024]

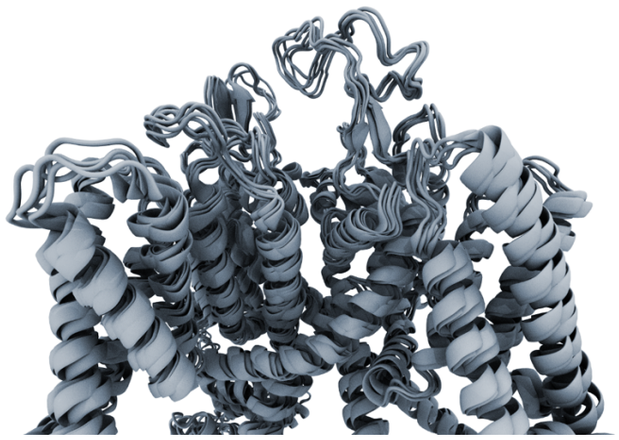
- First image of black hole
- First image of black hole at center of Milky Way [Gasparini 2023; Fletcher 2022]

Google Translate

- Language translation [Caswell, Liang 2020]

Machine learning (ML): why & what

ML is increasingly a part of major discoveries and decisions:



- AlphaFold: big stride in predicting "a protein's 3D shape based solely on the 1D string of molecules that comprise it" [Jumper et al 2021; Toews 2021; Levine, Tu 2024]

- First image of black hole
- First image of black hole at center of Milky Way [Gasparini 2023; Fletcher 2022]

Google Translate

- Language translation [Caswell, Liang 2020]

The Economist

- Election predictions [Heidemanns et al 2020]

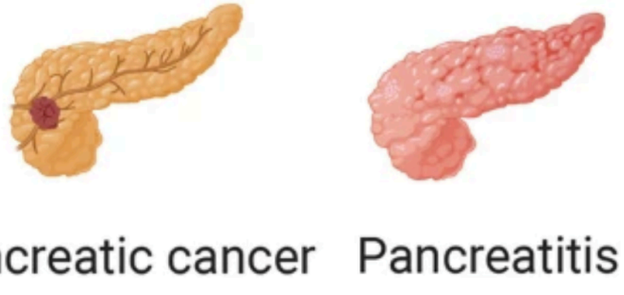
Machine learning (ML): why & what

Just this past week:

Machine learning (ML): why & what

Just this past week:

[Kawai et al 2024;
Shi et al 2024]



Pancreatic cancer Pancreatitis

- Predicting pancreatic cancer from serum samples

Machine learning (ML): why & what

Just this past week:

[Kawai et al 2024;
Shi et al 2024]

[Perre 2024; Goswami et al 2024]

- Predicting pancreatic cancer from serum samples
- Predicting battery fires in electric vehicles for prevention

Machine learning (ML): why & what

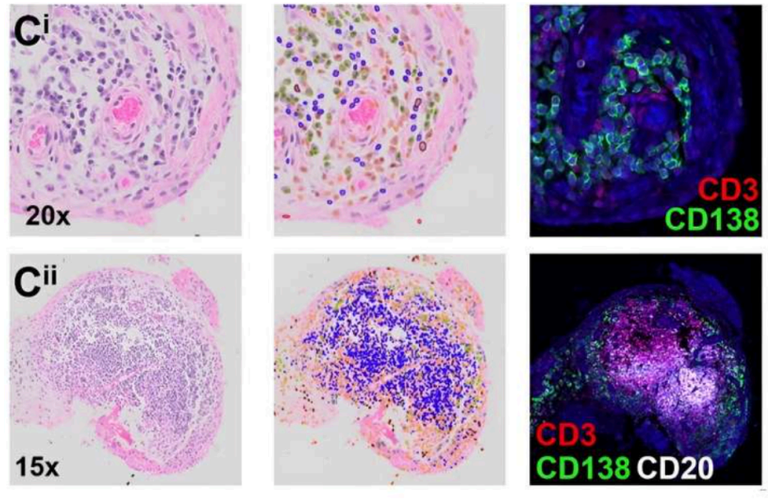
Just this past week:

[Kawai et al 2024;
Shi et al 2024]

- Predicting pancreatic cancer from serum samples

[Perre 2024; Goswami et al 2024]

- Predicting battery fires in electric vehicles for prevention



[Weill Cornell Medical College 2024;
Bell et al 2024]

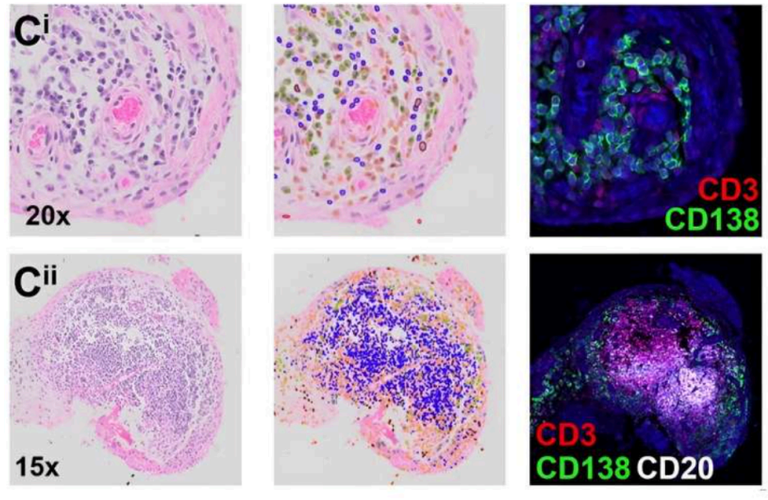
- Identifying subtypes of rheumatoid arthritis

Machine learning (ML): why & what

Just this past week:

[Kawai et al 2024;
Shi et al 2024]

[Perre 2024; Goswami et al 2024]



[Weill Cornell Medical College 2024;
Bell et al 2024]

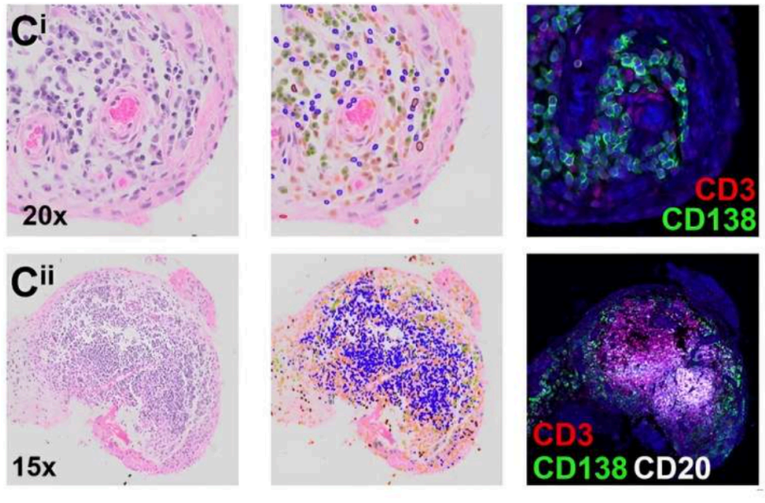
- Predicting pancreatic cancer from serum samples
- Predicting battery fires in electric vehicles for prevention
- Identifying subtypes of rheumatoid arthritis
- And many many more…

Machine learning (ML): why & what

Just this past week:

[Kawai et al 2024; Shi et al 2024]

[Perre 2024; Goswami et al 2024]



[Weill Cornell Medical College 2024; Bell et al 2024]

- Predicting pancreatic cancer from serum samples
- Predicting battery fires in electric vehicles for prevention
- Identifying subtypes of rheumatoid arthritis
- And many many more…
- **What is machine learning?** A set of methods for using data to become better at a task.
  - No quick summary will do full justice. See the rest of the course!

Machine learning (ML): why & what

- But also: "'[M]any scientists were never really trained properly to [apply ML] because the field is still relatively new,' says Casey Bennett at DePaul University […] 'I see a lot of common mistakes repeated over and over,' he says. For ML tools used in health research, he adds, 'it's like the Wild West right now.'" [Ball 2023]

Machine learning (ML): why & what

- But also: "'[M]any scientists were never really trained properly to [apply ML] because the field is still relatively new,' says Casey Bennett at DePaul University […] 'I see a lot of common mistakes repeated over and over,' he says. For ML tools used in health research, he adds, 'it's like the Wild West right now.'" [Ball 2023]
- **Why take this course?** To understand the mechanism behind machine learning methods
  - To responsibly apply them
  - To understand and evaluate them
  - To innovate
  - To ultimately advance our understanding of science, engineering, and social science

Getting started

- Example (ongoing) problem: spam detection [Mathur 2024; Google Support]

Getting started

- Example (ongoing) problem: spam detection [Mathur 2024; Google Support]
- Data (e.g. each data point is an email)

Getting started

- Example (ongoing) problem: spam detection [Mathur 2024; Google Support]
- Data (e.g. each data point is an email)
  - *n*th data point has **features** $x^{(n)} \in \mathcal{X}$ (often $\mathcal{X} = \mathbb{R}^D$)

D: feature dimension

Getting started

- Example (ongoing) problem: spam detection [Mathur 2024; Google Support]
- Data (e.g. each data point is an email)
  - *n*th data point has **features** $x^{(n)} \in \mathcal{X}$ (often $\mathcal{X} = \mathbb{R}^D$)
    - E.g. timestamp, # of string “free”, # of string “money”

D: feature dimension

Getting started

- Example (ongoing) problem: spam detection [Mathur 2024; Google Support]
- Data (e.g. each data point is an email)
  - *n*th data point has **features** $x^{(n)} \in \mathcal{X}$ (often $\mathcal{X} = \mathbb{R}^D$)
    - E.g. timestamp, # of string “free”, # of string “money”
  - *n*th data point has **label** $y^{(n)} \in \mathcal{Y}$

D: feature dimension

Getting started

- Example (ongoing) problem: spam detection [Mathur 2024; Google Support]
- Data (e.g. each data point is an email)
  - *n*th data point has **features** $x^{(n)} \in \mathcal{X}$ (often $\mathcal{X} = \mathbb{R}^D$)
    - E.g. timestamp, # of string “free”, # of string “money”
  - *n*th data point has **label** $y^{(n)} \in \mathcal{Y}$
    - E.g. $\mathcal{Y} = \{0, 1\}$, interpreted as not-spam and spam

D: feature dimension

Getting started

- Example (ongoing) problem: spam detection [Mathur 2024; Google Support]
- Data (e.g. each data point is an email)
  - *n*th data point has **features** $x^{(n)} \in \mathcal{X}$ (often $\mathcal{X} = \mathbb{R}^D$)
    - E.g. timestamp, # of string “free”, # of string “money”
  - *n*th data point has **label** $y^{(n)} \in \mathcal{Y}$
    - E.g. $\mathcal{Y} = \{0, 1\}$, interpreted as not-spam and spam
  - **Training set**: $\{(x^{(n)}, y^{(n)})\}_{n=1}^{N}$

D: feature dimension

N: number of training data points

Getting started

- Example (ongoing) problem: spam detection [Mathur 2024; Google Support]
- Data (e.g. each data point is an email)
  - *n*th data point has **features** $x^{(n)} \in \mathcal{X}$ (often $\mathcal{X} = \mathbb{R}^D$)
    - E.g. timestamp, # of string “free”, # of string “money”
  - *n*th data point has **label** $y^{(n)} \in \mathcal{Y}$
    - E.g. $\mathcal{Y} = \{0, 1\}$, interpreted as not-spam and spam
  - **Training set**: $\{(x^{(n)}, y^{(n)})\}_{n=1}^{N}$
- Accomplish a task

D: feature dimension

N: number of training data points

Getting started

- Example (ongoing) problem: spam detection [Mathur 2024; Google Support]
- Data (e.g. each data point is an email)
  - *n*th data point has **features** $x^{(n)} \in \mathcal{X}$ (often $\mathcal{X} = \mathbb{R}^D$)
    - E.g. timestamp, # of string “free”, # of string “money”
  - *n*th data point has **label** $y^{(n)} \in \mathcal{Y}$
    - E.g. $\mathcal{Y} = \{0, 1\}$, interpreted as not-spam and spam
  - **Training set**: $\{(x^{(n)}, y^{(n)})\}_{n=1}^{N}$
- Accomplish a task
  - E.g. label email as spam or not-spam

D: feature dimension

N: number of training data points

Getting started

- Example (ongoing) problem: spam detection [Mathur 2024; Google Support]
- Data (e.g. each data point is an email)
  - *n*th data point has **features** $x^{(n)} \in \mathcal{X}$ (often $\mathcal{X} = \mathbb{R}^D$)
    - E.g. timestamp, # of string “free”, # of string “money”
  - *n*th data point has **label** $y^{(n)} \in \mathcal{Y}$
    - E.g. $\mathcal{Y} = \{0, 1\}$, interpreted as not-spam and spam
  - **Training set**: $\{(x^{(n)}, y^{(n)})\}_{n=1}^{N}$
- Accomplish a task
  - E.g. label email as spam or not-spam
  - **Decision rule** $h : \mathcal{X} \to \mathcal{Y}$

D: feature dimension

N: number of training data points

Getting started

- Example (ongoing) problem: spam detection [Mathur 2024; Google Support]
- Data (e.g. each data point is an email)
  - *n*th data point has **features** $x^{(n)} \in \mathcal{X}$ (often $\mathcal{X} = \mathbb{R}^D$)
    - E.g. timestamp, # of string “free”, # of string “money”
  - *n*th data point has **label** $y^{(n)} \in \mathcal{Y}$
    - E.g. $\mathcal{Y} = \{0, 1\}$, interpreted as not-spam and spam
  - **Training set**: $\{(x^{(n)}, y^{(n)})\}_{n=1}^{N}$
- Accomplish a task
  - E.g. label email as spam or not-spam
  - **Decision rule** $h : \mathcal{X} \to \mathcal{Y}$
    - E.g. $h(x) = 1$

D: feature dimension

N: number of training data points

Getting started

- Example (ongoing) problem: spam detection [Mathur 2024; Google Support]
- Data (e.g. each data point is an email)
  - *n*th data point has **features** $x^{(n)} \in \mathcal{X}$ (often $\mathcal{X} = \mathbb{R}^D$)
    - E.g. timestamp, # of string “free”, # of string “money”
  - *n*th data point has **label** $y^{(n)} \in \mathcal{Y}$
    - E.g. $\mathcal{Y} = \{0, 1\}$, interpreted as not-spam and spam
  - **Training set**: $\{(x^{(n)}, y^{(n)})\}_{n=1}^{N}$
- Accomplish a task
  - E.g. label email as spam or not-spam
  - **Decision rule** $h : \mathcal{X} \to \mathcal{Y}$
    - E.g. $h(x) = 1$
    - E.g. $h(x) = 1$ if the string “free” occurs, else 0

D: feature dimension

N: number of training data points

Getting started

- Example (ongoing) problem: spam detection [Mathur 2024; Google Support]
- Data (e.g. each data point is an email)
  - *n*th data point has **features** $x^{(n)} \in \mathcal{X}$ (often $\mathcal{X} = \mathbb{R}^D$)
    - E.g. timestamp, # of string “free”, # of string “money”
  - *n*th data point has **label** $y^{(n)} \in \mathcal{Y}$
    - E.g. $\mathcal{Y} = \{0, 1\}$, interpreted as not-spam and spam
  - **Training set**: $\{(x^{(n)}, y^{(n)})\}_{n=1}^{N}$
- Accomplish a task
  - E.g. label email as spam or not-spam
  - **Decision rule** $h : \mathcal{X} \to \mathcal{Y}$ (can depend on the training set)
    - E.g. *h*(*x*) = 1
    - E.g. *h*(*x*) = 1 if the string “free” occurs, else 0

D: feature dimension

N: number of training data points

Getting started

- Example (ongoing) problem: spam detection [Mathur 2024; Google Support]
- Data (e.g. each data point is an email)
  - *n*th data point has **features** $x^{(n)} \in \mathcal{X}$ (often $\mathcal{X} = \mathbb{R}^D$)
    - E.g. timestamp, # of string “free”, # of string “money”
  - *n*th data point has **label** $y^{(n)} \in \mathcal{Y}$
    - E.g. $\mathcal{Y} = \{0, 1\}$, interpreted as not-spam and spam
  - **Training set**: $\{(x^{(n)}, y^{(n)})\}_{n=1}^{N}$
- Accomplish a task
  - E.g. label email as spam or not-spam
  - **Decision rule** $h : \mathcal{X} \to \mathcal{Y}$ (can depend on the training set)
    - E.g. $h(x) = 1$
    - E.g. $h(x) = 1$ if the string “free” occurs, else 0
    - E.g. $h(x) = 1$ if the timestamp of x matches the timestamp of any email in the training set exactly, else 0

D: feature dimension

N: number of training data points

Getting started

- Example (ongoing) problem: spam detection [Mathur 2024; Google Support]
- Data (e.g. each data point is an email)
  - *n*th data point has **features** $x^{(n)} \in \mathcal{X}$ (often $\mathcal{X} = \mathbb{R}^D$)
    - E.g. timestamp, # of string "free", # of string "money"
  - *n*th data point has **label** $y^{(n)} \in \mathcal{Y}$
    - E.g. $\mathcal{Y} = \{0, 1\}$, interpreted as not-spam and spam
  - **Training set**: $\{(x^{(n)}, y^{(n)})\}_{n=1}^{N}$

D: feature dimension

N: number of training data points

- Accomplish a task
  - E.g. label email as spam or not-spam
  - **Decision rule** $h : \mathcal{X} \to \mathcal{Y}$ (can depend on the training set)
    - E.g. $h(x) = 1$
    - E.g. $h(x) = 1$ if the string "free" occurs, else 0
    - E.g. $h(x) = 1$ if the timestamp of x matches the timestamp of any email in the training set exactly, else 0
    - Decision rule? with the example features above: $h(x) = 1$ if the string "free money" occurs, else 0

Getting started

- Example (ongoing) problem: spam detection [Mathur 2024; Google Support]
- Data (e.g. each data point is an email)
  - *n*th data point has **features** $x^{(n)} \in \mathcal{X}$ (often $\mathcal{X} = \mathbb{R}^D$)
    - E.g. timestamp, # of string “free”, # of string “money”
  - *n*th data point has **label** $y^{(n)} \in \mathcal{Y}$
    - E.g. $\mathcal{Y} = \{0, 1\}$, interpreted as not-spam and spam
  - **Training set**: $\{(x^{(n)}, y^{(n)})\}_{n=1}^{N}$
- Accomplish a task
  - E.g. label email as spam or not-spam
  - **Decision rule** $h : \mathcal{X} \to \mathcal{Y}$ (can depend on the training set)
    - E.g. $h(x) = 1$
    - E.g. $h(x) = 1$ if the string “free” occurs, else 0
    - E.g. $h(x) = 1$ if the timestamp of x matches the timestamp of any email in the training set exactly, else 0
    - Not a decision rule with the example features above: $h(x) = 1$ if the string “free money” occurs, else 0

D: feature dimension

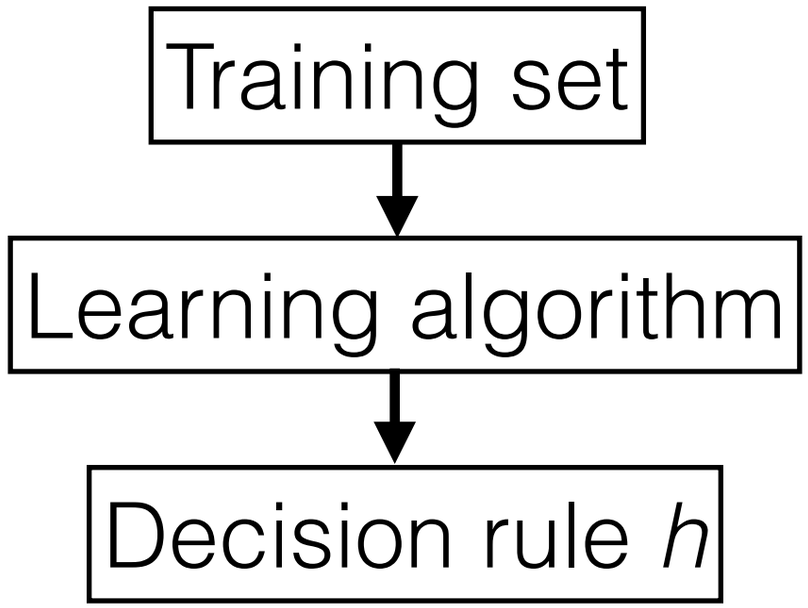
N: number of training data points

Getting started

- **Supervised learning**: learn a decision rule from features $x \in \mathcal{X}$ to labels $y \in \mathcal{Y}$ given training set $\{(x^{(n)}, y^{(n)})\}_{n=1}^{N}$

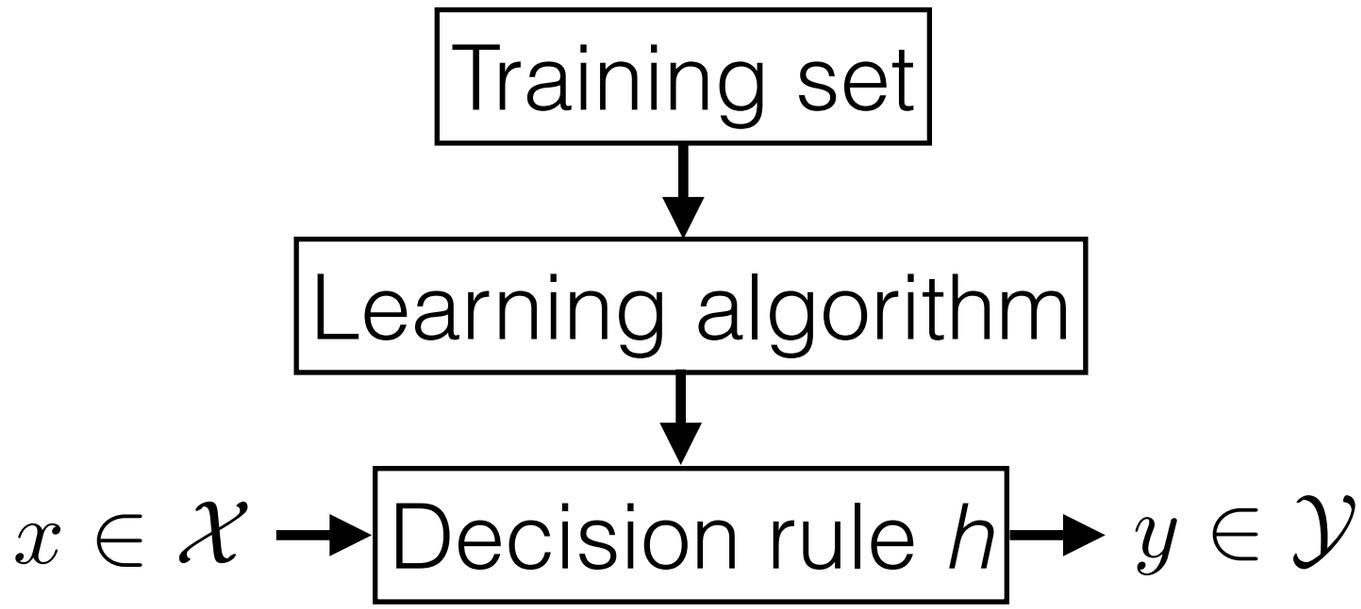
Getting started

- **Supervised learning**: learn a decision rule from features $x \in \mathcal{X}$ to labels $y \in \mathcal{Y}$ given training set $\{(x^{(n)}, y^{(n)})\}_{n=1}^{N}$



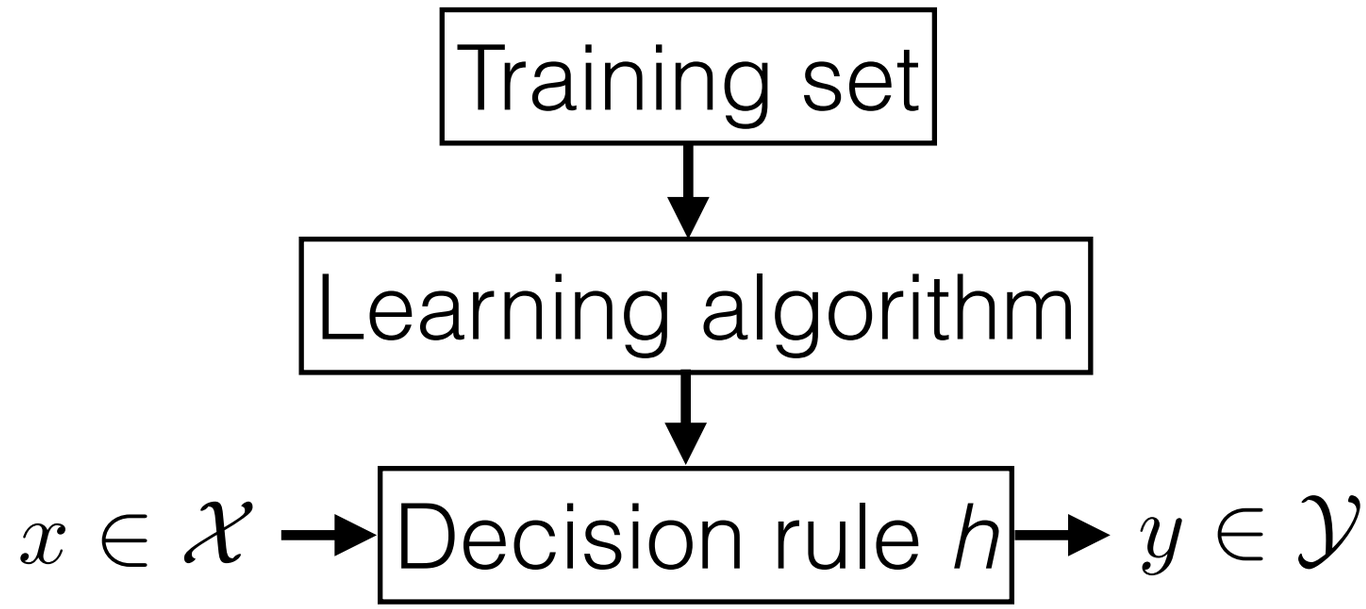
Getting started

- **Supervised learning**: learn a decision rule from features $x \in \mathcal{X}$ to labels $y \in \mathcal{Y}$ given training set $\{(x^{(n)}, y^{(n)})\}_{n=1}^{N}$



Getting started

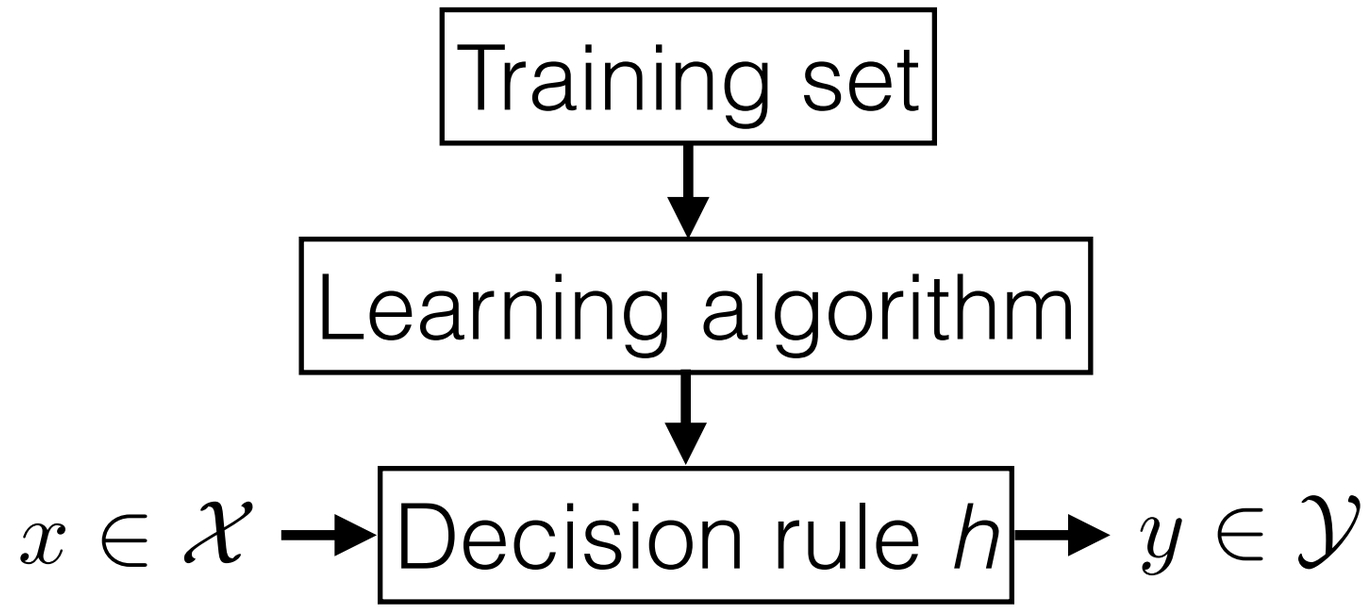
- **Supervised learning**: learn a decision rule from features $x \in \mathcal{X}$ to labels $y \in \mathcal{Y}$ given training set $\{(x^{(n)}, y^{(n)})\}_{n=1}^{N}$



- What does it mean to perform well?

Getting started

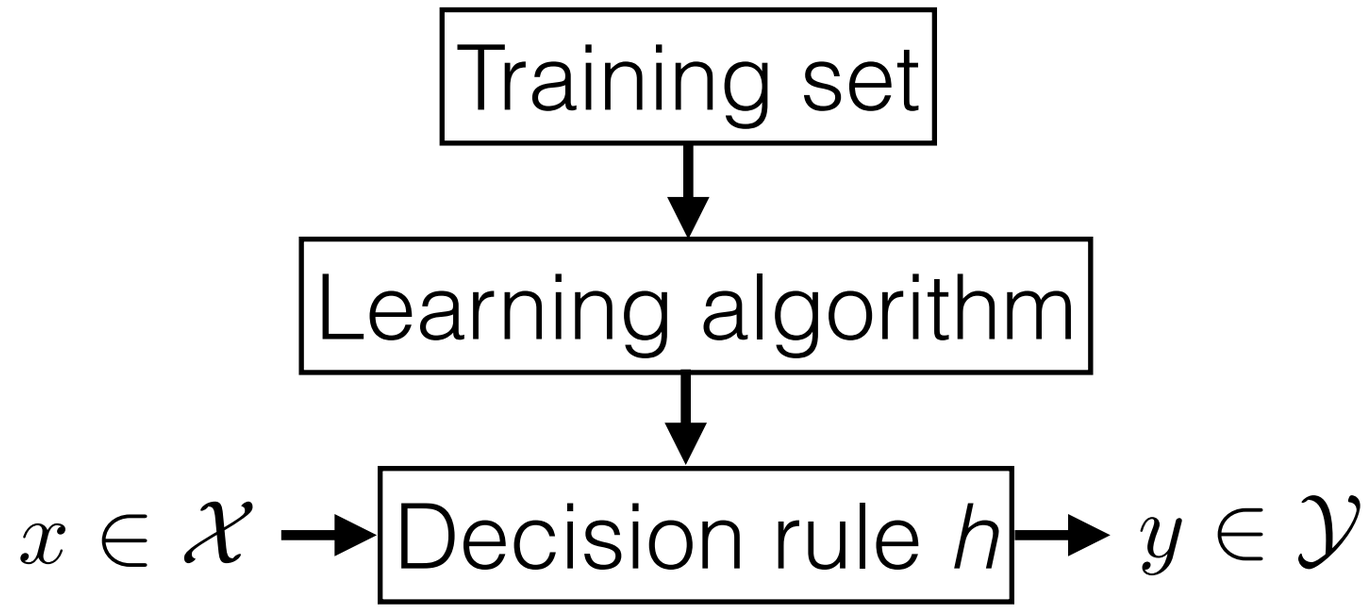
- **Supervised learning**: learn a decision rule from features $x \in \mathcal{X}$ to labels $y \in \mathcal{Y}$ given training set $\{(x^{(n)}, y^{(n)})\}_{n=1}^{N}$



- What does it mean to perform well?
  - Loss(actual, guess)

Getting started

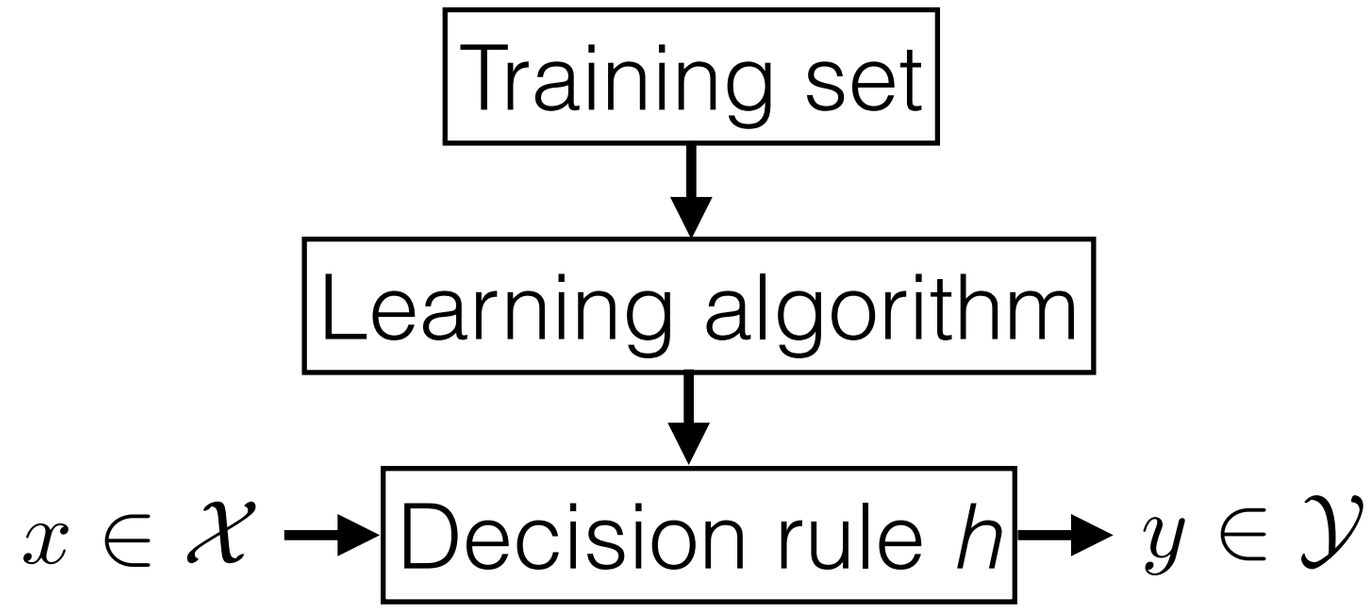
- **Supervised learning**: learn a decision rule from features $x \in \mathcal{X}$ to labels $y \in \mathcal{Y}$ given training set $\{(x^{(n)}, y^{(n)})\}_{n=1}^{N}$



- What does it mean to perform well?
  - Loss(actual, guess)
  - Loss function $L : \mathcal{Y} \times \mathcal{Y} \to \mathbb{R}$

Getting started

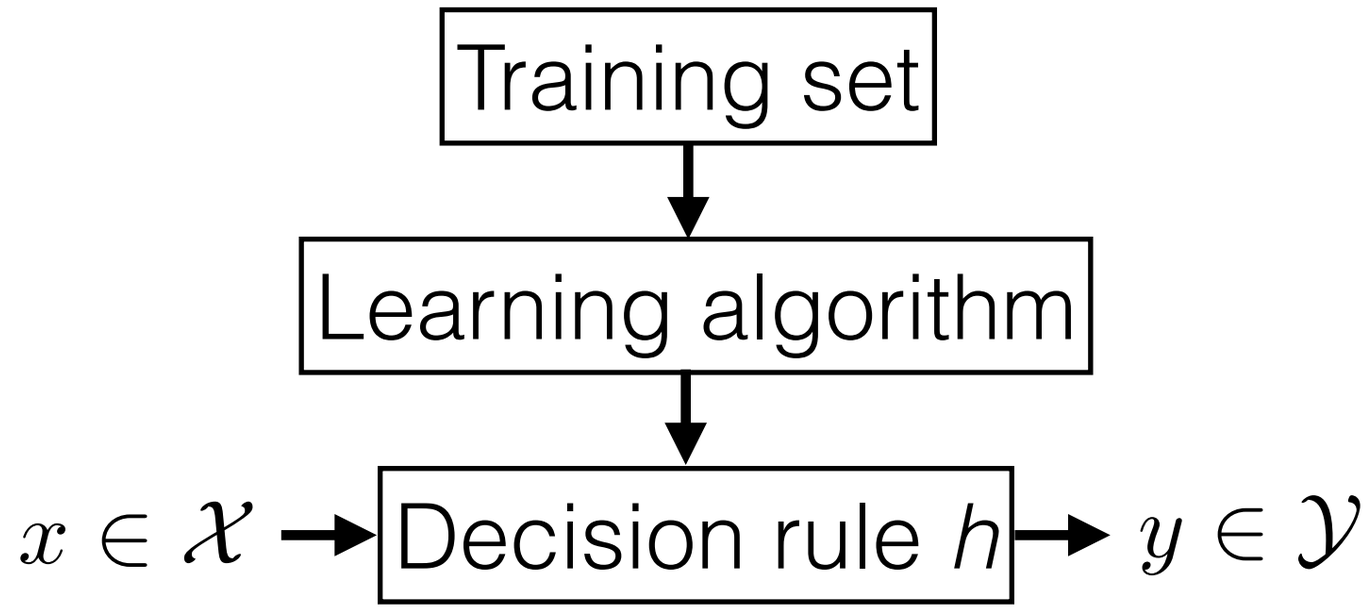
- **Supervised learning**: learn a decision rule from features $x \in \mathcal{X}$ to labels $y \in \mathcal{Y}$ given training set $\{(x^{(n)}, y^{(n)})\}_{n=1}^{N}$



- What does it mean to perform well?
  - Loss(actual, guess)
  - Loss function $L : \mathcal{Y} \times \mathcal{Y} \to \mathbb{R}$
  - Example: 0-1 loss: $L(a, g) = \mathbf{1}(a \neq g)$

Getting started

- **Supervised learning**: learn a decision rule from features $x \in \mathcal{X}$ to labels $y \in \mathcal{Y}$ given training set $\{(x^{(n)}, y^{(n)})\}_{n=1}^{N}$

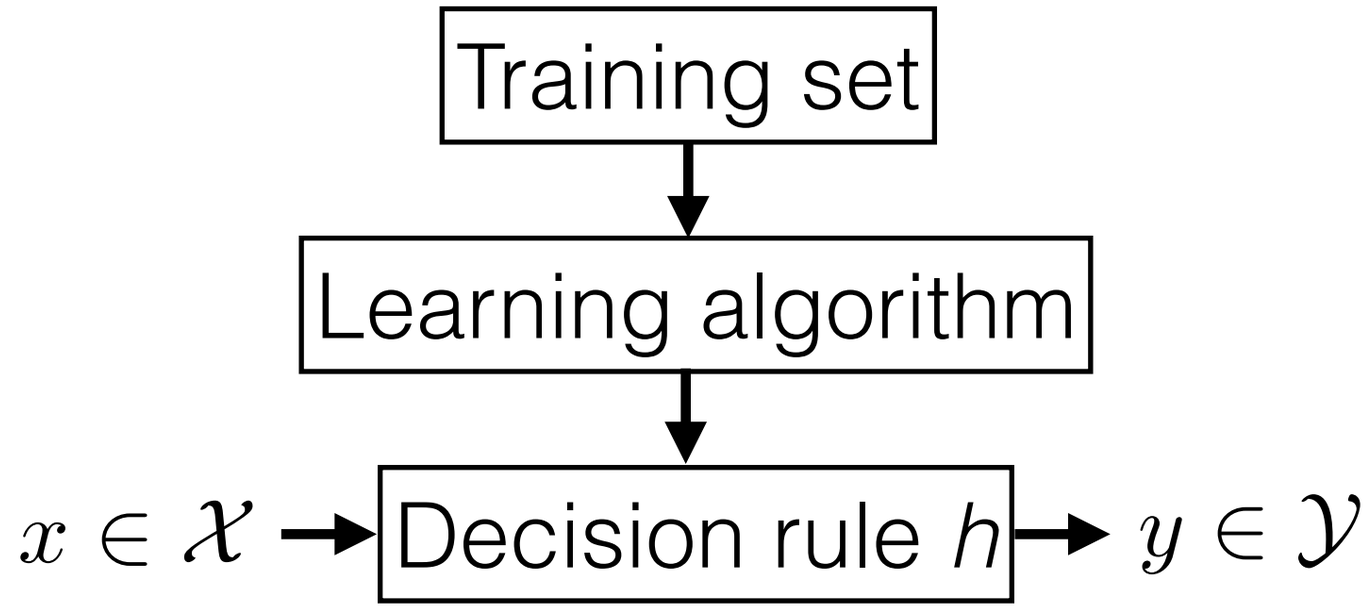


- What does it mean to perform well?
  - Loss(actual, guess)
  - Loss function $L : \mathcal{Y} \times \mathcal{Y} \to \mathbb{R}$
  - Example: 0-1 loss: $L(a, g) = \mathbf{1}(a \neq g)$

L	g = 0	g = 1
a = 0	0	1
a = 1	1	0

Getting started

- **Supervised learning**: learn a decision rule from features $x \in \mathcal{X}$ to labels $y \in \mathcal{Y}$ given training set $\{(x^{(n)}, y^{(n)})\}_{n=1}^{N}$

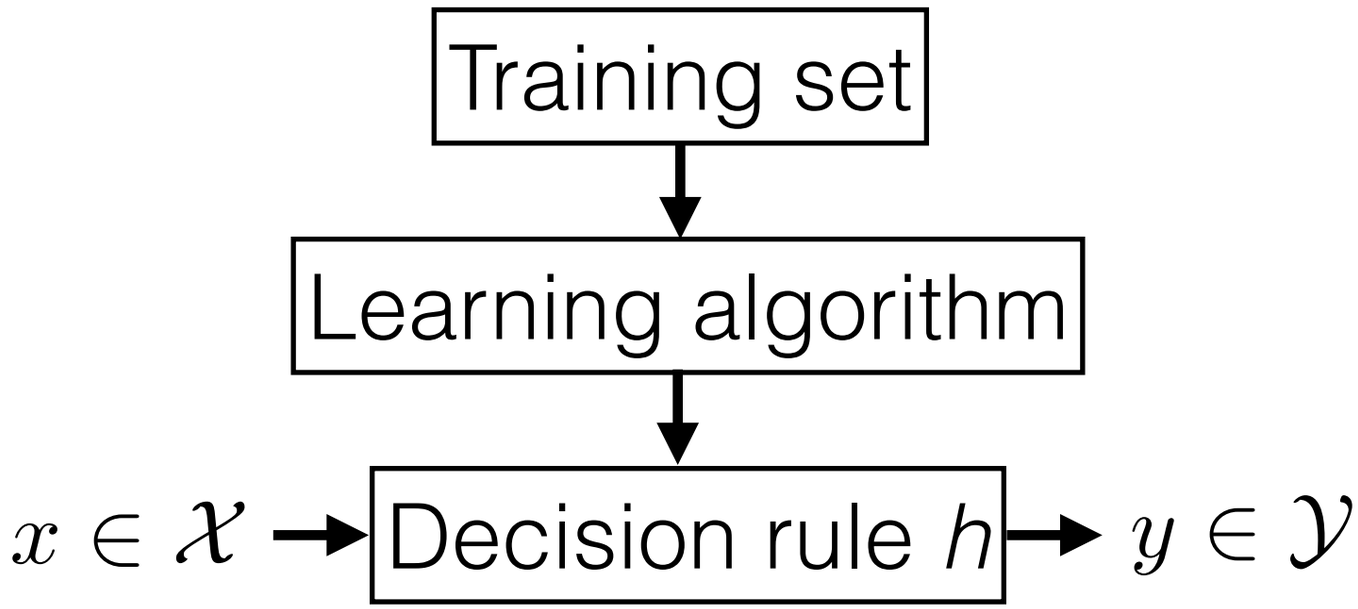


- What does it mean to perform well?
  - Loss(actual, guess)
  - Loss function $L : \mathcal{Y} \times \mathcal{Y} \to \mathbb{R}$
  - Example: 0-1 loss: $L(a, g) = \mathbf{1}(a \neq g)$
  - Example: asymmetric loss

L	g = 0	g = 1
a = 0	0	1
a = 1	1	0

Getting started

- **Supervised learning**: learn a decision rule from features $x \in \mathcal{X}$ to labels $y \in \mathcal{Y}$ given training set $\{(x^{(n)}, y^{(n)})\}_{n=1}^{N}$

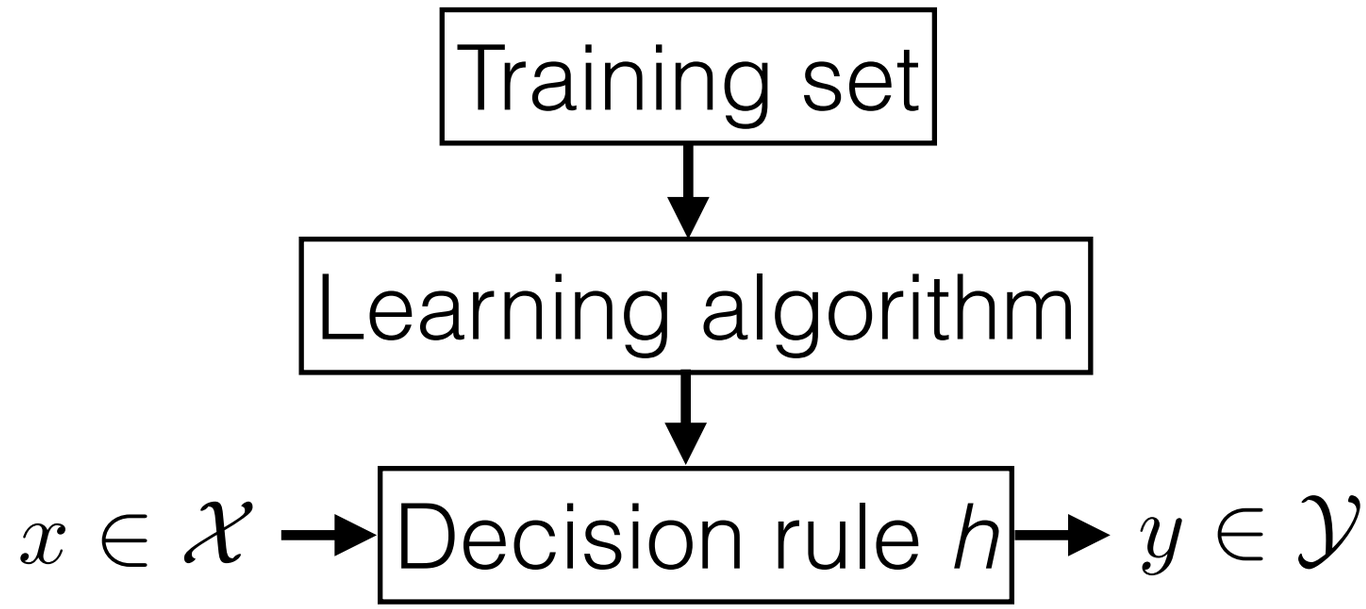


- What does it mean to perform well?
  - Loss(actual, guess)
  - Loss function $L : \mathcal{Y} \times \mathcal{Y} \to \mathbb{R}$
  - Example: 0-1 loss: $L(a, g) = \mathbf{1}(a \neq g)$
  - Example: asymmetric loss

L	g = 0	g = 1
a = 0	0	1
a = 1	1	0

Getting started

- **Supervised learning**: learn a decision rule from features $x \in \mathcal{X}$ to labels $y \in \mathcal{Y}$ given training set $\{(x^{(n)}, y^{(n)})\}_{n=1}^{N}$

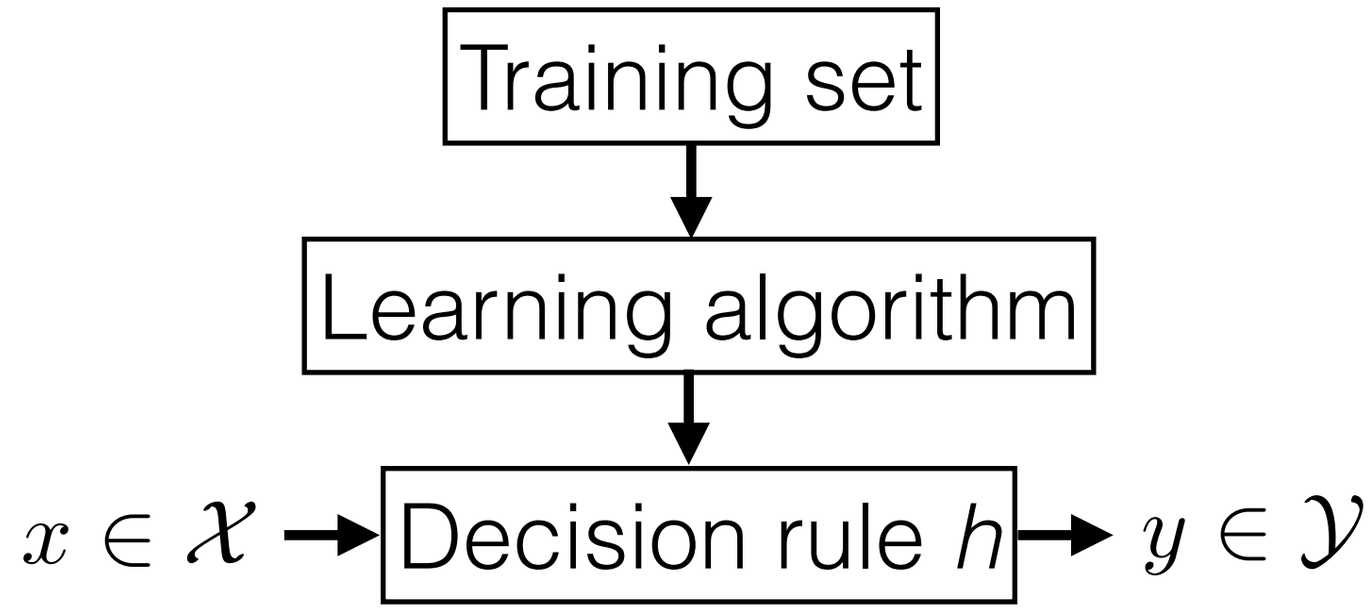


- What does it mean to perform well?
  - Loss(actual, guess)
  - Loss function $L : \mathcal{Y} \times \mathcal{Y} \to \mathbb{R}$
  - Example: 0-1 loss: $L(a, g) = \mathbf{1}(a \neq g)$
  - Example: asymmetric loss

L	g = 0	g = 1
a = 0	0	15
a = 1	1	0

Getting started

- **Supervised learning**: learn a decision rule from features $x \in \mathcal{X}$ to labels $y \in \mathcal{Y}$ given training set $\{(x^{(n)}, y^{(n)})\}_{n=1}^{N}$



- What does it mean to perform well?
  - Loss(actual, guess)
  - Loss function $L : \mathcal{Y} \times \mathcal{Y} \to \mathbb{R}$
  - Example: 0-1 loss: $L(a, g) = \mathbf{1}(a \neq g)$
  - Example: asymmetric loss
  - For a data point (*x*,*y*) and decision rule *h*: $L(y, h(x))$

L	g = 0	g = 1
a = 0	0	15
a = 1	1	0

Getting started

- What does it mean to perform well?

Getting started

- What does it mean to perform well?
  - Suppose we get a new data point $(x^{(N+1)}, y^{(N+1)})$

Getting started

- What does it mean to perform well?
  - Suppose we get a new data point $(x^{(N+1)}, y^{(N+1)})$
  - Want low $L(y^{(N+1)}, h(x^{(N+1)}))$

Getting started

- What does it mean to perform well?
  - Suppose we get a new data point $(x^{(N+1)}, y^{(N+1)})$
  - Want low $L(y^{(N+1)}, h(x^{(N+1)}))$
  - But we don't actually know the label, and before we get the data point (e.g. email), we don't know the features

Getting started

- What does it mean to perform well?
  - Suppose we get a new data point $(x^{(N+1)}, y^{(N+1)})$
  - Want low $L(y^{(N+1)}, h(x^{(N+1)}))$
  - But we don't actually know the label, and before we get the data point (e.g. email), we don't know the features
  - "Probability is the language of uncertainty"

Getting started

- What does it mean to perform well?
  - Suppose we get a new data point $(x^{(N+1)}, y^{(N+1)})$
  - Want low $L(y^{(N+1)}, h(x^{(N+1)}))$
  - But we don't actually know the label, and before we get the data point (e.g. email), we don't know the features
  - "Probability is the language of uncertainty"
    - Let *X*, *Y* be random variables expressing our uncertainty over this new data point

Getting started

- What does it mean to perform well?
  - Suppose we get a new data point $(x^{(N+1)}, y^{(N+1)})$
  - Want low $L(y^{(N+1)}, h(x^{(N+1)}))$
  - But we don't actually know the label, and before we get the data point (e.g. email), we don't know the features
  - "Probability is the language of uncertainty"
    - Let *X*, *Y* be random variables expressing our uncertainty over this new data point
  - We can measure performance by the **risk** (a.k.a. expected loss) $\mathbb{E}[L(Y, h(X))]$

Getting started

- What does it mean to perform well?
  - Suppose we get a new data point $(x^{(N+1)}, y^{(N+1)})$
  - Want low $L(y^{(N+1)}, h(x^{(N+1)}))$
  - But we don't actually know the label, and before we get the data point (e.g. email), we don't know the features
  - "Probability is the language of uncertainty"
    - Let *X*, *Y* be random variables expressing our uncertainty over this new data point
    - We can measure performance by the **risk** (a.k.a. expected loss) $\mathbb{E}[L(Y, h(X))]$
- What if I want to perform well on the next *M* points?

Getting started

- What does it mean to perform well?
  - Suppose we get a new data point $(x^{(N+1)}, y^{(N+1)})$
  - Want low $L(y^{(N+1)}, h(x^{(N+1)}))$
  - But we don't actually know the label, and before we get the data point (e.g. email), we don't know the features
  - "Probability is the language of uncertainty"
    - Let *X*, *Y* be random variables expressing our uncertainty over this new data point
    - We can measure performance by the **risk** (a.k.a. expected loss) $\mathbb{E}[L(Y, h(X))]$
- What if I want to perform well on the next *M* points?

$$\frac{1}{M} \sum_{m=1}^{M} \mathbb{E}[L(Y^{(N+m)}, h(X^{(N+m)}))]$$

Getting started

- What does it mean to perform well?
  - Suppose we get a new data point $(x^{(N+1)}, y^{(N+1)})$
  - Want low $L(y^{(N+1)}, h(x^{(N+1)}))$
  - But we don't actually know the label, and before we get the data point (e.g. email), we don't know the features
  - "Probability is the language of uncertainty"
    - Let *X*, *Y* be random variables expressing our uncertainty over this new data point
    - We can measure performance by the **risk** (a.k.a. expected loss) $\mathbb{E}[L(Y, h(X))]$
- What if I want to perform well on the next *M* points?

$$\frac{1}{M}\sum_{m=1}^{M}\mathbb{E}[L(Y^{(N+m)}, h(X^{(N+m)}))]$$

  - Exercise: if we assume the pairs $\{(X^{(N+m)}, Y^{(N+m)})\}_{m=1}^{M}$ are iid across m, show we get back the single-point risk

Getting started

- What does it mean to perform well?
  - Suppose we get a new data point $(x^{(N+1)}, y^{(N+1)})$
  - Want low $L(y^{(N+1)}, h(x^{(N+1)}))$
  - But we don't actually know the label, and before we get the data point (e.g. email), we don't know the features
  - "Probability is the language of uncertainty"
    - Let *X*, *Y* be random variables expressing our uncertainty over this new data point
    - We can measure performance by the **risk** (a.k.a. expected loss) $\mathbb{E}[L(Y, h(X))]$
- What if I want to perform well on the next *M* points?

$$\frac{1}{M}\sum_{m=1}^{M}\mathbb{E}[L(Y^{(N+m)}, h(X^{(N+m)}))]$$

iid = independent and identically distributed

  - Exercise: if we assume the pairs $\{(X^{(N+m)}, Y^{(N+m)})\}_{m=1}^{M}$ are iid across m, show we get back the single-point risk

Getting started

- What does it mean to perform well?
  - Suppose we get a new data point $(x^{(N+1)}, y^{(N+1)})$
  - Want low $L(y^{(N+1)}, h(x^{(N+1)}))$
  - But we don't actually know the label, and before we get the data point (e.g. email), we don't know the features
  - "Probability is the language of uncertainty"
    - Let *X*, *Y* be random variables expressing our uncertainty over this new data point
    - We can measure performance by the **risk** (a.k.a. expected loss) $\mathbb{E}[L(Y, h(X))]$
- What if I want to perform well on the next *M* points?

$$\frac{1}{M}\sum_{m=1}^{M}\mathbb{E}[L(Y^{(N+m)}, h(X^{(N+m)}))]$$

iid = independent and identically distributed

  - Exercise: if we assume the pairs $\{(X^{(N+m)}, Y^{(N+m)})\}_{m=1}^{M}$ are iid across m, show we get back the single-point risk

Getting started

- So we'd like to choose a decision rule $h : \mathcal{X} \to \mathcal{Y}$ that minimizes risk on a new (as-yet-unseen) point: $\mathbb{E}[L(Y, h(X))]$

Getting started

- So we'd like to choose a decision rule $h : \mathcal{X} \to \mathcal{Y}$ that minimizes risk on a new (as-yet-unseen) point: $\mathbb{E}[L(Y, h(X))]$
- If we know the distribution of (*X*, *Y*), choosing *h* is often straightforward

Getting started

- So we'd like to choose a decision rule $h : \mathcal{X} \to \mathcal{Y}$ that minimizes risk on a new (as-yet-unseen) point: $\mathbb{E}[L(Y, h(X))]$
- If we know the distribution of (*X*, *Y*), choosing *h* is often straightforward
- **Proposition**. Consider *K*-class **classification**: $\mathcal{Y} = \{1, \ldots, K\}$

labels take *K* unordered & mutually exclusive values

Getting started

- So we'd like to choose a decision rule $h : \mathcal{X} \to \mathcal{Y}$ that minimizes risk on a new (as-yet-unseen) point: $\mathbb{E}[L(Y, h(X))]$
- If we know the distribution of (*X*,*Y*), choosing *h* is often straightforward
- **Proposition**. Consider *K*-class **classification**: $\mathcal{Y} = \{1, \ldots, K\}$
  - Suppose *X* is a discrete random variable & we know *p*(*x*,*y*)

labels take *K* unordered & mutually exclusive values

Getting started

- So we'd like to choose a decision rule $h : \mathcal{X} \to \mathcal{Y}$ that minimizes risk on a new (as-yet-unseen) point: $\mathbb{E}[L(Y, h(X))]$
- If we know the distribution of (*X*,*Y*), choosing *h* is often straightforward

labels take *K* unordered & mutually exclusive values

- **Proposition**. Consider *K*-class **classification**: $\mathcal{Y} = \{1, \ldots, K\}$
  - Suppose *X* is a discrete random variable & we know *p*(*x*,*y*)
  - Then the following decision rule minimizes the risk of a new point: $h(x) = \arg\min_k \sum_{j=1}^{K} L(j,k) p(y = j|x)$

Getting started

- So we'd like to choose a decision rule $h : \mathcal{X} \to \mathcal{Y}$ that minimizes risk on a new (as-yet-unseen) point: $\mathbb{E}[L(Y, h(X))]$
- If we know the distribution of (*X*,*Y*), choosing *h* is often straightforward

labels take *K* unordered & mutually exclusive values

- **Proposition**. Consider *K*-class **classification**: $\mathcal{Y} = \{1, \ldots, K\}$
  - Suppose *X* is a discrete random variable & we know *p*(*x*,*y*)
  - Then the following decision rule minimizes the risk of a new point: $h(x) = \arg\min_k \sum_{j=1}^{K} L(j,k) p(y = j|x)$

Loss(actual,guess)

Getting started

- So we'd like to choose a decision rule $h : \mathcal{X} \to \mathcal{Y}$ that minimizes risk on a new (as-yet-unseen) point: $\mathbb{E}[L(Y, h(X))]$
- If we know the distribution of (*X*,*Y*), choosing *h* is often straightforward

labels take *K* unordered & mutually exclusive values

- **Proposition**. Consider *K*-class **classification**: $\mathcal{Y} = \{1, \ldots, K\}$
  - Suppose *X* is a discrete random variable & we know *p*(*x*,*y*)
  - Then the following decision rule minimizes the risk of a new point: $h(x) = \arg\min_k \sum_{j=1}^{K} L(j,k) p(y = j|x)$
- *Proof*.

Getting started

- So we'd like to choose a decision rule $h : \mathcal{X} \to \mathcal{Y}$ that minimizes risk on a new (as-yet-unseen) point: $\mathbb{E}[L(Y, h(X))]$
- If we know the distribution of (*X*,*Y*), choosing *h* is often straightforward

labels take *K* unordered & mutually exclusive values

- **Proposition**. Consider *K*-class **classification**: $\mathcal{Y} = \{1, \ldots, K\}$
  - Suppose *X* is a discrete random variable & we know *p*(*x*,*y*)
  - Then the following decision rule minimizes the risk of a new point: $h(x) = \arg\min_k \sum_{j=1}^{K} L(j,k) p(y = j|x)$
- *Proof.* $\mathbb{E}[L(Y, h(X))] = \sum_{j=1}^{K} \sum_{x \in \mathcal{X}} L(j, h(x)) p(x, y = j)$

Getting started

- So we'd like to choose a decision rule $h : \mathcal{X} \to \mathcal{Y}$ that minimizes risk on a new (as-yet-unseen) point: $\mathbb{E}[L(Y, h(X))]$
- If we know the distribution of (*X*,*Y*), choosing *h* is often straightforward

labels take *K* unordered & mutually exclusive values

- **Proposition**. Consider *K*-class **classification**: $\mathcal{Y} = \{1, \ldots, K\}$
  - Suppose *X* is a discrete random variable & we know *p*(*x*,*y*)
  - Then the following decision rule minimizes the risk of a new point: $h(x) = \arg\min_k \sum_{j=1}^{K} L(j,k) p(y = j|x)$
- *Proof.* $\mathbb{E}[L(Y, h(X))] = \sum_{j=1}^{K} \sum_{x \in \mathcal{X}} L(j, h(x)) p(x, y = j)$

$$= \sum_{j=1}^{K} \sum_{x \in \mathcal{X}} L(j, h(x)) p(y = j|x) p(x)$$

Getting started

- So we'd like to choose a decision rule $h : \mathcal{X} \to \mathcal{Y}$ that minimizes risk on a new (as-yet-unseen) point: $\mathbb{E}[L(Y, h(X))]$
- If we know the distribution of (*X*,*Y*), choosing *h* is often straightforward

labels take *K* unordered & mutually exclusive values

- **Proposition**. Consider *K*-class **classification**: $\mathcal{Y} = \{1, \ldots, K\}$
  - Suppose *X* is a discrete random variable & we know *p*(*x*,*y*)
  - Then the following decision rule minimizes the risk of a new point: $h(x) = \arg\min_k \sum_{j=1}^{K} L(j,k) p(y = j|x)$
- *Proof.* $\mathbb{E}[L(Y, h(X))] = \sum_{j=1}^{K} \sum_{x \in \mathcal{X}} L(j, h(x)) p(x, y = j)$

$$= \sum_{j=1}^{K} \sum_{x \in \mathcal{X}} L(j, h(x)) p(y = j|x) p(x)$$

Getting started

- So we'd like to choose a decision rule $h : \mathcal{X} \to \mathcal{Y}$ that minimizes risk on a new (as-yet-unseen) point: $\mathbb{E}[L(Y, h(X))]$
- If we know the distribution of (*X*,*Y*), choosing *h* is often straightforward

labels take *K* unordered & mutually exclusive values

- **Proposition**. Consider *K*-class **classification**: $\mathcal{Y} = \{1, \ldots, K\}$
  - Suppose *X* is a discrete random variable & we know *p*(*x*,*y*)
  - Then the following decision rule minimizes the risk of a new point: $h(x) = \arg\min_k \sum_{j=1}^{K} L(j,k) p(y = j|x)$
- *Proof.*

$$\begin{aligned}
\mathbb{E}[L(Y, h(X))] &= \sum_{j=1}^{K} \sum_{x \in \mathcal{X}} L(j, h(x)) p(x, y = j) \\
&= \sum_{j=1}^{K} \sum_{x \in \mathcal{X}} L(j, h(x)) p(y = j|x) p(x) \\
&= \sum_{x \in \mathcal{X}} p(x) \sum_{j=1}^{K} L(j, h(x)) p(y = j|x)
\end{aligned}$$

Getting started

- So we'd like to choose a decision rule $h : \mathcal{X} \to \mathcal{Y}$ that minimizes risk on a new (as-yet-unseen) point: $\mathbb{E}[L(Y, h(X))]$
- If we know the distribution of (*X*,*Y*), choosing *h* is often straightforward

labels take *K* unordered & mutually exclusive values

- **Proposition**. Consider *K*-class **classification**: $\mathcal{Y} = \{1, \ldots, K\}$
  - Suppose *X* is a discrete random variable & we know *p*(*x*,*y*)
  - Then the following decision rule minimizes the risk of a new point: $h(x) = \arg\min_k \sum_{j=1}^{K} L(j,k) p(y = j|x)$
- *Proof.*

$$\begin{aligned}
\mathbb{E}[L(Y, h(X))] &= \sum_{j=1}^{K} \sum_{x \in \mathcal{X}} L(j, h(x)) p(x, y = j) \\
&= \sum_{j=1}^{K} \sum_{x \in \mathcal{X}} L(j, h(x)) p(y = j|x) p(x) \\
&= \sum_{x \in \mathcal{X}} p(x) \sum_{j=1}^{K} L(j, h(x)) p(y = j|x)
\end{aligned}$$

Getting started

- So we'd like to choose a decision rule $h : \mathcal{X} \to \mathcal{Y}$ that minimizes risk on a new (as-yet-unseen) point: $\mathbb{E}[L(Y, h(X))]$
- If we know the distribution of (*X*,*Y*), choosing *h* is often straightforward

labels take *K* unordered & mutually exclusive values

- **Proposition**. Consider *K*-class **classification**: $\mathcal{Y} = \{1, \ldots, K\}$
  - Suppose *X* is a discrete random variable & we know *p*(*x*,*y*)
  - Then the following decision rule minimizes the risk of a new point: $h(x) = \arg\min_k \sum_{j=1}^{K} L(j,k) p(y = j|x)$
- *Proof.*

$$\begin{aligned}
\mathbb{E}[L(Y, h(X))] &= \sum_{j=1}^{K} \sum_{x \in \mathcal{X}} L(j, h(x)) p(x, y = j) \\
&= \sum_{j=1}^{K} \sum_{x \in \mathcal{X}} L(j, h(x)) p(y = j|x) p(x) \\
&= \sum_{x \in \mathcal{X}} p(x) \sum_{j=1}^{K} L(j, h(x)) p(y = j|x)
\end{aligned}$$

Getting started

- So we'd like to choose a decision rule $h : \mathcal{X} \to \mathcal{Y}$ that minimizes risk on a new (as-yet-unseen) point: $\mathbb{E}[L(Y, h(X))]$
- If we know the distribution of (*X*,*Y*), choosing *h* is often straightforward

labels take *K* unordered & mutually exclusive values

- **Proposition**. Consider *K*-class **classification**: $\mathcal{Y} = \{1, \ldots, K\}$
  - Suppose *X* is a discrete random variable & we know *p*(*x*,*y*)
  - Then the following decision rule minimizes the risk of a new point: $h(x) = \arg\min_k \sum_{j=1}^{K} L(j,k) p(y = j|x)$
- *Proof.*

$$\begin{aligned}
\mathbb{E}[L(Y, h(X))] &= \sum_{j=1}^{K} \sum_{x \in \mathcal{X}} L(j, h(x)) p(x, y = j) \\
&= \sum_{j=1}^{K} \sum_{x \in \mathcal{X}} L(j, h(x)) p(y = j|x) p(x) \\
&= \sum_{x \in \mathcal{X}} p(x) \sum_{j=1}^{K} L(j, h(x)) p(y = j|x)
\end{aligned}$$

Getting started

- So we'd like to choose a decision rule $h : \mathcal{X} \to \mathcal{Y}$ that minimizes risk on a new (as-yet-unseen) point: $\mathbb{E}[L(Y, h(X))]$
- If we know the distribution of (*X*,*Y*), choosing *h* is often straightforward

labels take *K* unordered & mutually exclusive values

- **Proposition**. Consider *K*-class **classification**: $\mathcal{Y} = \{1, \ldots, K\}$
  - Suppose *X* is a discrete random variable & we know *p*(*x*,*y*)
  - Then the following decision rule minimizes the risk of a new point: $h(x) = \arg\min_k \sum_{j=1}^{K} L(j,k) p(y = j|x)$
- *Proof.*

$$\begin{aligned}\mathbb{E}[L(Y, h(X))] &= \sum_{j=1}^{K} \sum_{x \in \mathcal{X}} L(j, h(x)) p(x, y = j) \\ &= \sum_{j=1}^{K} \sum_{x \in \mathcal{X}} L(j, h(x)) p(y = j|x) p(x) \\ &= \sum_{x \in \mathcal{X}} p(x) \sum_{j=1}^{K} L(j, h(x)) p(y = j|x)\end{aligned}$$

- Exercise: similar proposition and proof but continuous *x*

Getting started

- So we'd like to choose a decision rule $h : \mathcal{X} \to \mathcal{Y}$ that minimizes risk on a new (as-yet-unseen) point: $\mathbb{E}[L(Y, h(X))]$
- If we know the distribution of (*X*,*Y*), choosing *h* is often straightforward

labels take *K* unordered & mutually exclusive values

- **Proposition**. Consider *K*-class **classification**: $\mathcal{Y} = \{1, \ldots, K\}$
  - Suppose *X* is a discrete random variable & we know *p*(*x*,*y*)
  - Then the following decision rule minimizes the risk of a new point: $h(x) = \arg\min_k \sum_{j=1}^{K} L(j,k) p(y = j|x)$
- *Proof.*

$$\begin{aligned}\mathbb{E}[L(Y, h(X))] &= \sum_{j=1}^{K} \sum_{x \in \mathcal{X}} L(j, h(x)) p(x, y = j) \\ &= \sum_{j=1}^{K} \sum_{x \in \mathcal{X}} L(j, h(x)) p(y = j|x) p(x) \\ &= \sum_{x \in \mathcal{X}} p(x) \sum_{j=1}^{K} L(j, h(x)) p(y = j|x)\end{aligned}$$

- Exercise: similar proposition and proof but continuous *x*
- Example: *K*=2 and asymmetric classification loss

Getting started

- So we'd like to choose a decision rule $h : \mathcal{X} \to \mathcal{Y}$ that minimizes risk on a new (as-yet-unseen) point: $\mathbb{E}[L(Y, h(X))]$
- If we know the distribution of (*X*,*Y*), choosing *h* is often straightforward

labels take *K* unordered & mutually exclusive values

- **Proposition**. Consider *K*-class **classification**: $\mathcal{Y} = \{1, \ldots, K\}$
  - Suppose *X* is a discrete random variable & we know *p*(*x*,*y*)
  - Then the following decision rule minimizes the risk of a new point: $h(x) = \arg\min_k \sum_{j=1}^{K} L(j,k) p(y = j|x)$
- *Proof.*

$$\begin{aligned}
\mathbb{E}[L(Y, h(X))] &= \sum_{j=1}^{K} \sum_{x \in \mathcal{X}} L(j, h(x)) p(x, y = j) \\
&= \sum_{j=1}^{K} \sum_{x \in \mathcal{X}} L(j, h(x)) p(y = j|x) p(x) \\
&= \sum_{x \in \mathcal{X}} p(x) \sum_{j=1}^{K} L(j, h(x)) p(y = j|x)
\end{aligned}$$

- Exercise: similar proposition and proof but continuous *x*
- Example: *K*=2 and asymmetric classification loss

Getting started

- So we'd like to choose a decision rule $h : \mathcal{X} \to \mathcal{Y}$ that minimizes risk on a new (as-yet-unseen) point: $\mathbb{E}[L(Y, h(X))]$
- If we know the distribution of (*X*,*Y*), choosing *h* is often straightforward

labels take *K* unordered & mutually exclusive values

- **Proposition**. Consider *K*-class **classification**: $\mathcal{Y} = \{1, \ldots, K\}$
  - Suppose *X* is a discrete random variable & we know *p*(*x*,*y*)
  - Then the following decision rule minimizes the risk of a new point: $h(x) = \arg\min_k \sum_{j=1}^{K} L(j,k) p(y = j|x)$
- *Proof.*

$$\begin{aligned}\mathbb{E}[L(Y, h(X))] &= \sum_{j=1}^{K} \sum_{x \in \mathcal{X}} L(j, h(x)) p(x, y = j) \\ &= \sum_{j=1}^{K} \sum_{x \in \mathcal{X}} L(j, h(x)) p(y = j|x) p(x) \\ &= \sum_{x \in \mathcal{X}} p(x) \sum_{j=1}^{K} L(j, h(x)) p(y = j|x)\end{aligned}$$

- Exercise: similar proposition and proof but continuous *x*
- Example: *K*=2 and asymmetric classification loss
  - *k*=1 case:
  - *k*=2 case:

Getting started

- So we'd like to choose a decision rule $h : \mathcal{X} \to \mathcal{Y}$ that minimizes risk on a new (as-yet-unseen) point: $\mathbb{E}[L(Y, h(X))]$
- If we know the distribution of (*X*,*Y*), choosing *h* is often straightforward

labels take *K* unordered & mutually exclusive values

- **Proposition**. Consider *K*-class **classification**: $\mathcal{Y} = \{1, \ldots, K\}$
  - Suppose *X* is a discrete random variable & we know *p*(*x*,*y*)
  - Then the following decision rule minimizes the risk of a new point: $h(x) = \arg\min_k \sum_{j=1}^{K} L(j,k) p(y = j|x)$
- *Proof.*

$$\begin{aligned}\mathbb{E}[L(Y, h(X))] &= \sum_{j=1}^{K} \sum_{x \in \mathcal{X}} L(j, h(x)) p(x, y = j) \\ &= \sum_{j=1}^{K} \sum_{x \in \mathcal{X}} L(j, h(x)) p(y = j|x) p(x) \\ &= \sum_{x \in \mathcal{X}} p(x) \sum_{j=1}^{K} L(j, h(x)) p(y = j|x)\end{aligned}$$

- Exercise: similar proposition and proof but continuous *x*
- Example: *K*=2 and asymmetric classification loss
  - *k*=~~1~~ 0 case:
  - *k*=~~2~~ 1 case:

Getting started

- So we'd like to choose a decision rule $h : \mathcal{X} \to \mathcal{Y}$ that minimizes risk on a new (as-yet-unseen) point: $\mathbb{E}[L(Y, h(X))]$
- If we know the distribution of (*X*,*Y*), choosing *h* is often straightforward

labels take *K* unordered & mutually exclusive values

- **Proposition**. Consider *K*-class **classification**: $\mathcal{Y} = \{1, \ldots, K\}$
  - Suppose *X* is a discrete random variable & we know *p*(*x*,*y*)
  - Then the following decision rule minimizes the risk of a new point: $h(x) = \arg\min_k \sum_{j=1}^{K} L(j,k) p(y = j|x)$
- *Proof.*

$$\begin{aligned}\mathbb{E}[L(Y, h(X))] &= \sum_{j=1}^{K} \sum_{x \in \mathcal{X}} L(j, h(x)) p(x, y = j) \\ &= \sum_{j=1}^{K} \sum_{x \in \mathcal{X}} L(j, h(x)) p(y = j|x) p(x) \\ &= \sum_{x \in \mathcal{X}} p(x) \sum_{j=1}^{K} L(j, h(x)) p(y = j|x)\end{aligned}$$

- Exercise: similar proposition and proof but continuous *x*
- Example: *K*=2 and asymmetric classification loss
  - *k*=~~1~~0 case: $L(1,0)p(y = 1|x)$
  - *k*=~~2~~1 case:

Getting started

- So we'd like to choose a decision rule $h : \mathcal{X} \to \mathcal{Y}$ that minimizes risk on a new (as-yet-unseen) point: $\mathbb{E}[L(Y, h(X))]$
- If we know the distribution of (*X*,*Y*), choosing *h* is often straightforward

labels take *K* unordered & mutually exclusive values

- **Proposition**. Consider *K*-class **classification**: $\mathcal{Y} = \{1, \ldots, K\}$
  - Suppose *X* is a discrete random variable & we know *p*(*x*,*y*)
  - Then the following decision rule minimizes the risk of a new point: $h(x) = \arg\min_k \sum_{j=1}^{K} L(j,k) p(y = j|x)$
- *Proof.*

$$\begin{aligned}\mathbb{E}[L(Y, h(X))] &= \sum_{j=1}^{K} \sum_{x \in \mathcal{X}} L(j, h(x)) p(x, y = j) \\ &= \sum_{j=1}^{K} \sum_{x \in \mathcal{X}} L(j, h(x)) p(y = j|x) p(x) \\ &= \sum_{x \in \mathcal{X}} p(x) \sum_{j=1}^{K} L(j, h(x)) p(y = j|x)\end{aligned}$$

- Exercise: similar proposition and proof but continuous *x*
- Example: *K*=2 and asymmetric classification loss
  - ~~*k*=1~~ 0 case: $L(1,0)p(y = 1|x)$
  - ~~*k*=2~~ 1 case: $L(0,1)p(y = 0|x)$

Getting started

- So we'd like to choose a decision rule $h : \mathcal{X} \to \mathcal{Y}$ that minimizes risk on a new (as-yet-unseen) point: $\mathbb{E}[L(Y, h(X))]$
- If we know the distribution of (*X*,*Y*), choosing *h* is often straightforward

labels take *K* unordered & mutually exclusive values

- **Proposition**. Consider *K*-class **classification**: $\mathcal{Y} = \{1, \ldots, K\}$
  - Suppose *X* is a discrete random variable & we know *p*(*x*,*y*)
  - Then the following decision rule minimizes the risk of a new point: $h(x) = \arg\min_k \sum_{j=1}^{K} L(j,k) p(y = j|x)$
- *Proof.*

$$\begin{aligned}\mathbb{E}[L(Y, h(X))] &= \sum_{j=1}^{K} \sum_{x \in \mathcal{X}} L(j, h(x)) p(x, y = j) \\ &= \sum_{j=1}^{K} \sum_{x \in \mathcal{X}} L(j, h(x)) p(y = j|x) p(x) \\ &= \sum_{x \in \mathcal{X}} p(x) \sum_{j=1}^{K} L(j, h(x)) p(y = j|x)\end{aligned}$$

- Exercise: similar proposition and proof but continuous *x*
- Example: *K*=2 and asymmetric classification loss
  - ~~*k*=1~~ 0 case: $L(1,0)p(y = 1|x)$
  - ~~*k*=2~~ 1 case: $L(0,1)p(y = 0|x)$
- If 0-1 loss, choose *h*(*x*)=0 if *p*(*y*=1|*x*) < *p*(*y*=0|*x*)

Getting started

- So we'd like to choose a decision rule $h : \mathcal{X} \to \mathcal{Y}$ that minimizes risk on a new (as-yet-unseen) point: $\mathbb{E}[L(Y, h(X))]$
- If we know the distribution of (*X*, *Y*), choosing *h* is often straightforward

Getting started

- So we'd like to choose a decision rule $h : \mathcal{X} \to \mathcal{Y}$ that minimizes risk on a new (as-yet-unseen) point: $\mathbb{E}[L(Y, h(X))]$
- If we know the distribution of (*X*, *Y*), choosing *h* is often straightforward
- **Proposition**. Consider **regression** with $\mathcal{X} = \mathbb{R}^D, \mathcal{Y} = \mathbb{R}$

typically (but not always)
refers to a continuous label

Getting started

- So we'd like to choose a decision rule $h : \mathcal{X} \to \mathcal{Y}$ that minimizes risk on a new (as-yet-unseen) point: $\mathbb{E}[L(Y, h(X))]$
- If we know the distribution of (*X*, *Y*), choosing *h* is often straightforward

typically (but not always) refers to a continuous label

- **Proposition**. Consider **regression** with $\mathcal{X} = \mathbb{R}^D, \mathcal{Y} = \mathbb{R}$
  - Assume *X*, *Y* density *p*(*x*,*y*) & **square loss** $L(a, g) = (a - g)^2$

Getting started

- So we'd like to choose a decision rule $h : \mathcal{X} \to \mathcal{Y}$ that minimizes risk on a new (as-yet-unseen) point: $\mathbb{E}[L(Y, h(X))]$
- If we know the distribution of (*X*,*Y*), choosing *h* is often straightforward

typically (but not always) refers to a continuous label

- **Proposition**. Consider **regression** with $\mathcal{X} = \mathbb{R}^D, \mathcal{Y} = \mathbb{R}$
  - Assume *X*,*Y* density *p*(*x*,*y*) & **square loss** $L(a, g) = (a - g)^2$
  - (Assume all needed integrals exist.) Then this decision rule minimizes the risk of a new point: $h(x) = \mathbb{E}[Y|X = x]$

Getting started

- So we'd like to choose a decision rule $h : \mathcal{X} \to \mathcal{Y}$ that minimizes risk on a new (as-yet-unseen) point: $\mathbb{E}[L(Y, h(X))]$
- If we know the distribution of (*X*,*Y*), choosing *h* is often straightforward

typically (but not always) refers to a continuous label

- **Proposition**. Consider **regression** with $\mathcal{X} = \mathbb{R}^D, \mathcal{Y} = \mathbb{R}$
  - Assume *X*,*Y* density *p*(*x*,*y*) & **square loss** $L(a, g) = (a - g)^2$
  - (Assume all needed integrals exist.) Then this decision rule minimizes the risk of a new point: $h(x) = \mathbb{E}[Y|X = x]$
- Exercise: complete the proof.

Getting started

- So we'd like to choose a decision rule $h : \mathcal{X} \to \mathcal{Y}$ that minimizes risk on a new (as-yet-unseen) point: $\mathbb{E}[L(Y, h(X))]$
- If we know the distribution of (*X*,*Y*), choosing *h* is often straightforward

typically (but not always) refers to a continuous label

- **Proposition**. Consider **regression** with $\mathcal{X} = \mathbb{R}^D, \mathcal{Y} = \mathbb{R}$
  - Assume *X*,*Y* density *p*(*x*,*y*) & **square loss** $L(a, g) = (a - g)^2$
  - (Assume all needed integrals exist.) Then this decision rule minimizes the risk of a new point: $h(x) = \mathbb{E}[Y|X = x]$
- Exercise: complete the proof.
  - Hint: start by writing out the risk for this loss and the given density

Getting started

- So we'd like to choose a decision rule $h : \mathcal{X} \to \mathcal{Y}$ that minimizes risk on a new (as-yet-unseen) point: $\mathbb{E}[L(Y, h(X))]$
- If we know the distribution of (*X*,*Y*), choosing *h* is often straightforward

typically (but not always) refers to a continuous label

- **Proposition**. Consider **regression** with $\mathcal{X} = \mathbb{R}^D, \mathcal{Y} = \mathbb{R}$
  - Assume *X*,*Y* density *p*(*x*,*y*) & **square loss** $L(a, g) = (a - g)^2$
  - (Assume all needed integrals exist.) Then this decision rule minimizes the risk of a new point: $h(x) = \mathbb{E}[Y|X = x]$
- Exercise: complete the proof.
  - Hint: start by writing out the risk for this loss and the given density
  - Hint: it might be useful to observe that

$$y - h(x) =$$
$$y - E[Y|X = x] + E[Y|X = x] - h(x)$$

Getting started

- So we'd like to choose a decision rule $h : \mathcal{X} \to \mathcal{Y}$ that minimizes risk on a new (as-yet-unseen) point: $\mathbb{E}[L(Y, h(X))]$
- If we know the distribution of (*X*,*Y*), choosing *h* is often straightforward

typically (but not always) refers to a continuous label

- **Proposition**. Consider **regression** with $\mathcal{X} = \mathbb{R}^D, \mathcal{Y} = \mathbb{R}$
  - Assume *X*,*Y* density *p*(*x*,*y*) & **square loss** $L(a, g) = (a - g)^2$
  - (Assume all needed integrals exist.) Then this decision rule minimizes the risk of a new point: $h(x) = \mathbb{E}[Y|X = x]$

Getting started

- So we'd like to choose a decision rule $h : \mathcal{X} \to \mathcal{Y}$ that minimizes risk on a new (as-yet-unseen) point: $\mathbb{E}[L(Y, h(X))]$
- If we know the distribution of (*X*,*Y*), choosing *h* is often straightforward

typically (but not always) refers to a continuous label

- **Proposition**. Consider **regression** with $\mathcal{X} = \mathbb{R}^D, \mathcal{Y} = \mathbb{R}$
  - Assume *X*,*Y* density *p*(*x*,*y*) & **square loss** $L(a, g) = (a - g)^2$
  - (Assume all needed integrals exist.) Then this decision rule minimizes the risk of a new point: $h(x) = \mathbb{E}[Y|X = x]$
- Plot of actual survey data on wage and age for a subset of people in the mid-Atlantic region of the United States in the 2000s decade

[An Intro to Statistical Learning 2023, Fig 1.1]

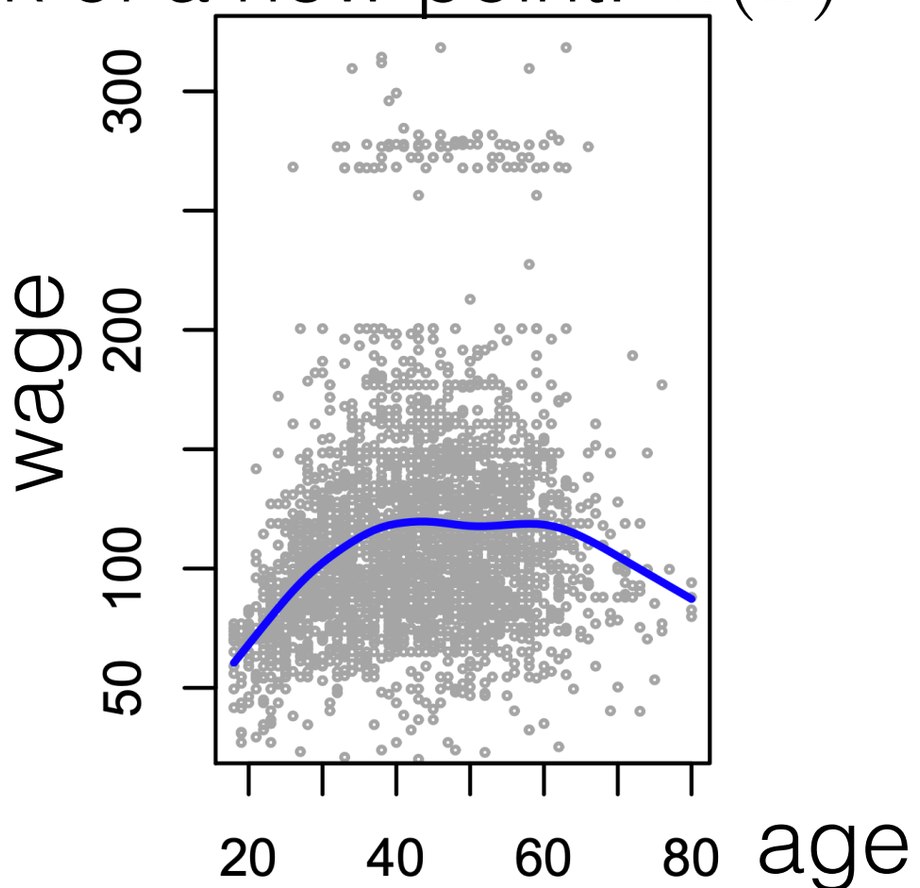
Getting started

- So we'd like to choose a decision rule $h : \mathcal{X} \to \mathcal{Y}$ that minimizes risk on a new (as-yet-unseen) point: $\mathbb{E}[L(Y, h(X))]$
- If we know the distribution of (*X*,*Y*), choosing *h* is often straightforward

typically (but not always) refers to a continuous label

- **Proposition**. Consider **regression** with $\mathcal{X} = \mathbb{R}^D, \mathcal{Y} = \mathbb{R}$
  - Assume *X*,*Y* density *p*(*x*,*y*) & **square loss** $L(a, g) = (a - g)^2$
  - (Assume all needed integrals exist.) Then this decision rule minimizes the risk of a new point: $h(x) = \mathbb{E}[Y|X = x]$

- Plot of actual survey data on wage and age for a subset of people in the mid-Atlantic region of the United States in the 2000s decade



[An Intro to Statistical Learning 2023, Fig 1.1]

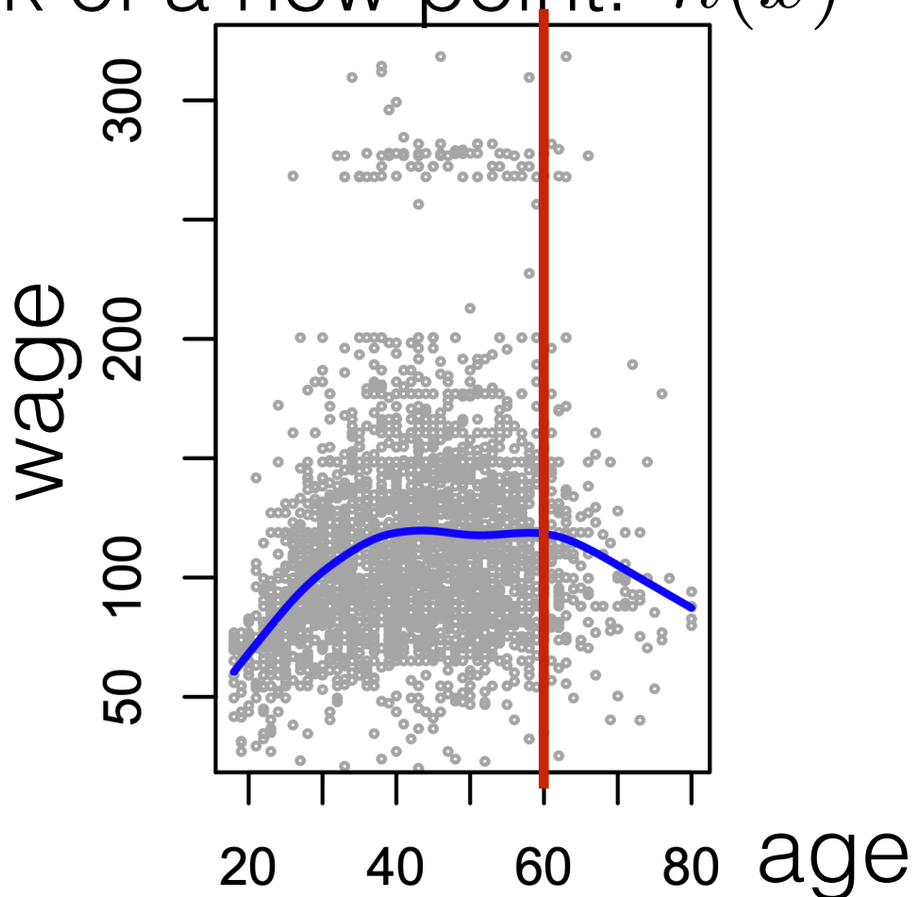
Getting started

- So we'd like to choose a decision rule $h : \mathcal{X} \to \mathcal{Y}$ that minimizes risk on a new (as-yet-unseen) point: $\mathbb{E}[L(Y, h(X))]$
- If we know the distribution of (*X*,*Y*), choosing *h* is often straightforward

typically (but not always) refers to a continuous label

- **Proposition**. Consider **regression** with $\mathcal{X} = \mathbb{R}^D, \mathcal{Y} = \mathbb{R}$
  - Assume *X*,*Y* density *p*(*x*,*y*) & **square loss** $L(a, g) = (a - g)^2$
  - (Assume all needed integrals exist.) Then this decision rule minimizes the risk of a new point: $h(x) = \mathbb{E}[Y|X = x]$

- Plot of actual survey data on wage and age for a subset of people in the mid-Atlantic region of the United States in the 2000s decade



[An Intro to Statistical Learning 2023, Fig 1.1]

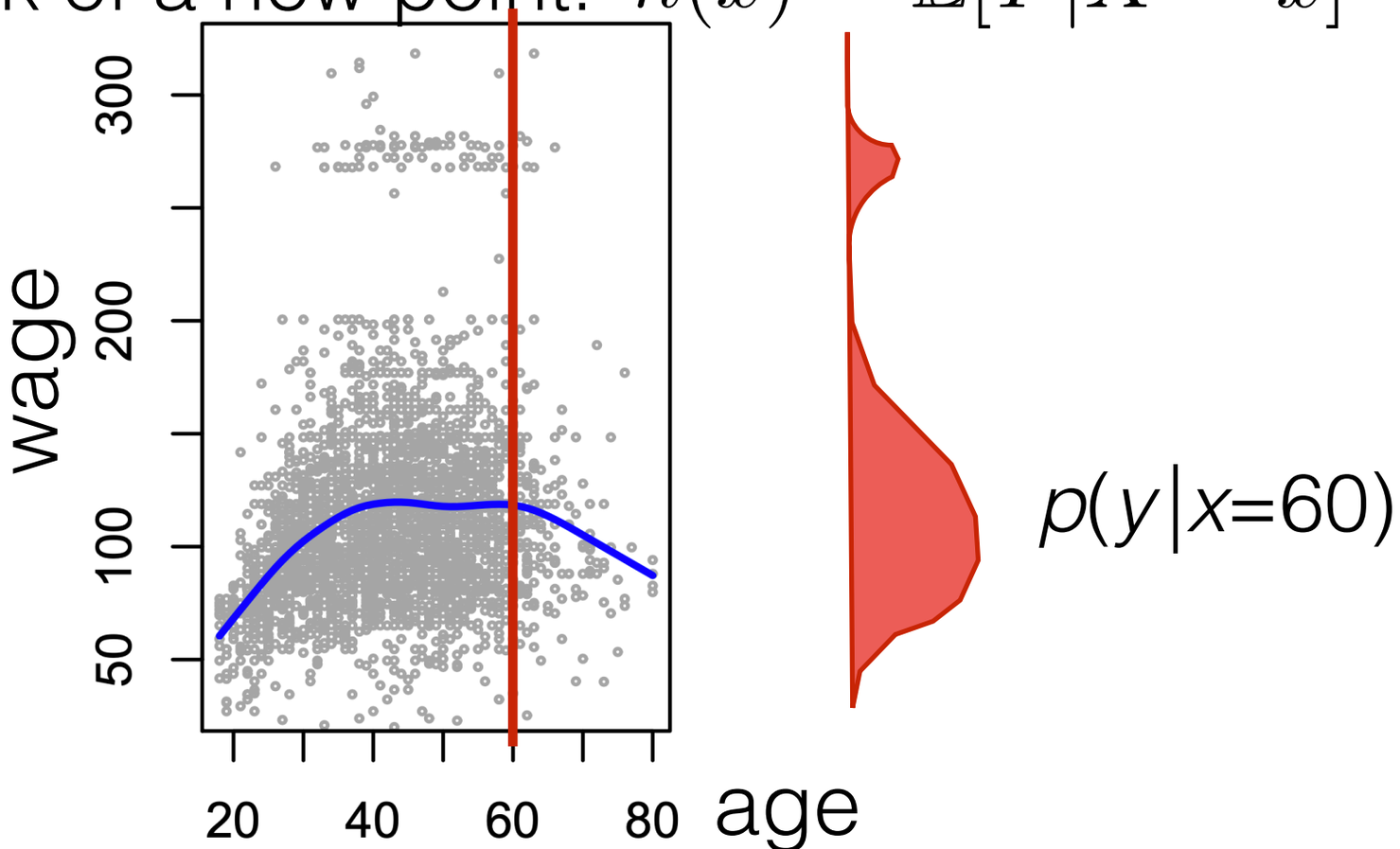
Getting started

- So we'd like to choose a decision rule $h : \mathcal{X} \to \mathcal{Y}$ that minimizes risk on a new (as-yet-unseen) point: $\mathbb{E}[L(Y, h(X))]$
- If we know the distribution of (*X*,*Y*), choosing *h* is often straightforward

typically (but not always) refers to a continuous label

- **Proposition**. Consider **regression** with $\mathcal{X} = \mathbb{R}^D, \mathcal{Y} = \mathbb{R}$
  - Assume *X*,*Y* density *p*(*x*,*y*) & **square loss** $L(a, g) = (a - g)^2$
  - (Assume all needed integrals exist.) Then this decision rule minimizes the risk of a new point: $h(x) = \mathbb{E}[Y|X = x]$

- Plot of actual survey data on wage and age for a subset of people in the mid-Atlantic region of the United States in the 2000s decade



[An Intro to Statistical Learning 2023, Fig 1.1]

Getting started

- So we'd like to choose a decision rule $h : \mathcal{X} \to \mathcal{Y}$ that minimizes risk on a new (as-yet-unseen) point: $\mathbb{E}[L(Y, h(X))]$
- If we know the distribution of (*X*, *Y*), choosing *h* is often straightforward

Getting started

- So we'd like to choose a decision rule $h : \mathcal{X} \to \mathcal{Y}$ that minimizes risk on a new (as-yet-unseen) point: $\mathbb{E}[L(Y, h(X))]$
- If we know the distribution of (X, Y), choosing h is often straightforward
- **Problem:** we don't usually know the distribution of (X, Y)

Getting started

- So we'd like to choose a decision rule $h : \mathcal{X} \to \mathcal{Y}$ that minimizes risk on a new (as-yet-unseen) point: $\mathbb{E}[L(Y, h(X))]$
- If we know the distribution of (*X*, *Y*), choosing *h* is often straightforward
- **Problem:** we don't usually know the distribution of (*X*, *Y*)
- Idea: Use our training data!

Getting started

- So we'd like to choose a decision rule $h : \mathcal{X} \to \mathcal{Y}$ that minimizes risk on a new (as-yet-unseen) point: $\mathbb{E}[L(Y, h(X))]$
- If we know the distribution of (*X*, *Y*), choosing *h* is often straightforward
- **Problem:** we don't usually know the distribution of (*X*, *Y*)
- Idea: Use our training data! We need to make an assumption about how our training data relates to our future data point(s)

Getting started

- So we'd like to choose a decision rule $h : \mathcal{X} \to \mathcal{Y}$ that minimizes risk on a new (as-yet-unseen) point: $\mathbb{E}[L(Y, h(X))]$
- If we know the distribution of (*X*, *Y*), choosing *h* is often straightforward
- **Problem:** we don't usually know the distribution of (*X*, *Y*)
- Idea: Use our training data! We need to make an assumption about how our training data relates to our future data point(s)
  - A common assumption: $(X^{(n)}, Y^{(n)})$ are iid for all *n*

Getting started

- So we'd like to choose a decision rule $h : \mathcal{X} \to \mathcal{Y}$ that minimizes risk on a new (as-yet-unseen) point: $\mathbb{E}[L(Y, h(X))]$
- If we know the distribution of (*X*, *Y*), choosing *h* is often straightforward
- **Problem:** we don't usually know the distribution of (*X*, *Y*)
- Idea: Use our training data! We need to make an assumption about how our training data relates to our future data point(s)
  - A common assumption: $(X^{(n)}, Y^{(n)})$ are iid for all *n*
- Additional idea: how about we replace the **risk** (of a new point) by the **empirical risk** over the training data?

$$\mathbb{E}[L(Y, h(X))] \approx \frac{1}{N} \sum_{n=1}^{N} L(y^{(n)}, h(x^{(n)}))$$

References (1/2)

Ball “Is AI leading to a reproducibility crisis in science?” *Nature News Feature*. December 5, 2023.

Caswell and Liang. “Recent Advances in Google Translate” *Google Research Blog*. June 8, 2020. https://research.google/blog/recent-advances-in-google-translate/

Fletcher “The First Milky Way Black Hole Image Lets Scientists Test Physics” *Scientific American*. September 1, 2022.

Gasparini “Machine Learning Delivers Sharper Black Hole Image” *Physics*. April 14, 2023.

Google Support. https://support.google.com/messages/answer/9327903?hl=en Accessed September 5, 2024.

Goswami, Basab Ranjan Das, et al. "Advancing battery safety: Integrating multiphysics and machine learning for thermal runaway prediction in lithium-ion battery module." *Journal of Power Sources* 614 (2024): 235015.

Heidemanns, Merlin, Andrew Gelman, and G. Elliott Morris. "An updated dynamic Bayesian forecasting model for the US presidential election." *Harvard Data Science Review* 2.4 (2020): 10-1162.

Jumper, et al. "Highly accurate protein structure prediction with AlphaFold." *Nature* 596.7873 (2021): 583-589.

Kawai, Munenori, et al. "Early detection of pancreatic cancer by comprehensive serum miRNA sequencing with automated machine learning." British Journal of Cancer (2024): 1-11.

References (2/2)

Levine, Herbert, and Yuhai Tu. "Machine learning meets physics: A two-way street." Proceedings of the National Academy of Sciences 121.27 (2024): e2403580121.

Mathur. “Google Messages now uses 'signals' from unencrypted chats to train AI spam detection” *Android Police*. September 2, 2024.

Perre. “Preventing car battery fires with help from machine learning” *The University of Arizona News*. August 29, 2024.

Shi, Wenjie, et al. "Integrating a microRNA signature as a liquid biopsy-based tool for the early diagnosis and prediction of potential therapeutic targets in pancreatic cancer." British Journal of Cancer 130.1 (2024): 125-134.

Toews “AlphaFold Is The Most Important Achievement In AI—Ever” *Forbes*, October 3, 2021.

Weill Cornell Medical College. “Machine learning helps identify rheumatoid arthritis subtypes” *Medical Xpress*. August 29, 2024.