Introduction to data mining and machine learning

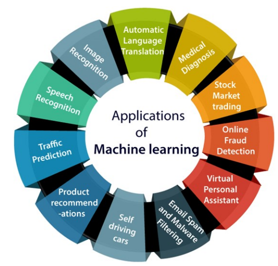
Data Mining—CS 530
Spring 2021
Chapman University

Instructor: **Uri Maoz**
TAs: **Dehua (Andy) Liang**
Caitlyn Chavez
Elnaz Lashgari

1

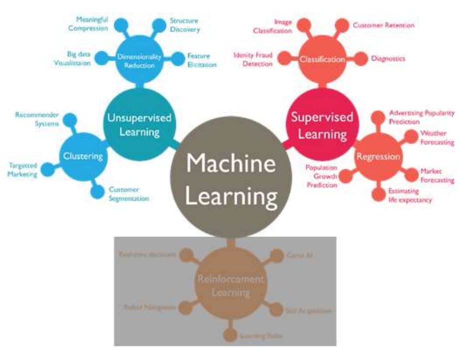
Machine learning

- Machine learning pervasive in academia, in industry, and in every-day life



2

Focus of course



3

Focus of course

- Data mining & machine-learning extremely strong tools. But if (as too often) used incorrectly, do more harm than good
- While application-centered, course will give intuition & insight, & often delve into math behind methods

4

What this course is & is not

Course is not
- Hand-wavy intro to deep-learning (& other cool, fashionable concepts)
- Theory-only introduction to machine-learning
- Easy, entertaining undergraduate course

5

What this course is & is not

Course is
- Broad, Practical intro to machine-learning
- Dive into math & theory as needed
- Discussion of intuition & insights into material
- Wide-ranging final project
  - Typically Work on real data
- Teach concepts crucial for advanced machine-learning courses & continuing on your own

6

What this course is & is not

- This is a graduate course
- Expect that everyone here is specifically interested in course material
- Everyone willing to work hard to master material
  - Material we do not finish in class will be left for home
- Backgrounds diverse
  - Some more math & science
  - Some more programming
  - Some with other backgrounds

7

What you need to know for this course

- Course programming language: Python (3.8)
- Math we will be using
  - Linear algebra (matrix & vector algebra)
  - Calculus
  - Probability

$$e^{i\pi} + 1 = 0$$

8

More about course

- Later classes build upon information disseminated in earlier classes
- Online classes will be recorded & slides distributed for students to revisit material as needed

9

More about course

- Later classes build upon information disseminated in earlier classes
- Online classes will be recorded & slides distributed for students to revisit material as needed
- Pace is fast, but feel free to ask questions if something is not clear

10

More about course

- Later classes build upon information disseminated in earlier classes
  - If you must miss a class—highly not recommended—make very sure you read though material & understand it completely. Talk to colleagues, TAs, or me
- Pace is fast, but feel free to ask questions if something is not clear
- Math involved in course not trivial. We will try to give intuition so that even if math not completely clear, its motivation & meaning will be
  - Again, feel free to ask questions

11

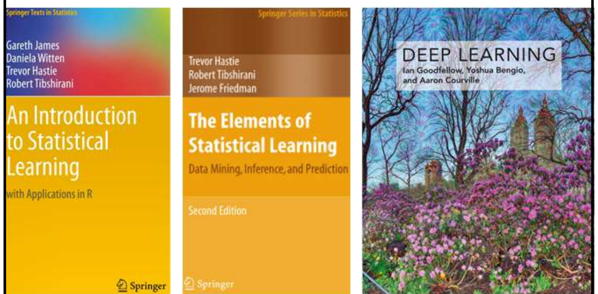
TA recitation classes

Recitation classes will

- Cover more technical material
  - Example: Python in a nutshell
- Work through step-by-step examples of material from class, including coding examples
- Go over solutions to homework exercises & quizzes

12

Optional textbooks



13

Bureaucracy

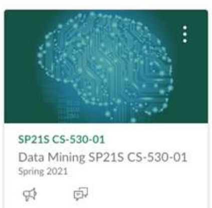
CS530

14

Course website

On Canvas

https://canvas.chapman.edu/



15

Topics

1. Regression, resampling, & regularization
   – Simple linear-regression
   – Multiple regression
   – Resampling methods
   – Regularization

16

Topics

2. Unsupervised learning
   – K-Means
   – PCA/LDA
   – ICA (if there is time)

17

Topics

3. Classification
   – Logistic regression
   – Naive Bayes
   – Tree-based methods
   – Support-vector machines

18

Topics

4. Neural networks & deep learning
   - Single neurons (biological & models)
   - Feed-forward neural networks (fixed weight & learning)
   - Perceptron
   - Gradient descent & Back Propagation in multi-layer feed-forward neural networks
   - Other architectures (convolutional layers, autoencoders, recurrent neural networks)
   - Regularization, data augmentation, & optimization in neural networks

19

Grading

Grade breakdown	Total points
Homework	25 points
– Typically, every week (worst 2 dropped)	
Quizzes	10 points
– Typically, take-home	
Final project	30 points
– In groups (individual grading)	
Final exam	20 points
Participation (not attendance)	15 points
Overall	100 points

20

Grading

- Homework & quizzes track comprehension continuously
  - What do you & don't you understand?
  - Homework typically given early in the week for submission by Sunday at 11:59pm
- Final project
  - Work on actual dataset using all material learned in class

21

Grading

- Participation (more than just attendance)
  - Class discussions, TA recitation class, office hours
  - Exit surveys
  - Group work (also judged by peers)
  - Camera on during entire class (while online)
- Quality of participation matters more than quantity

22

Exit surveys (part of participation)

Typically made up of two questions

1. Please list two things that you learned in class today
2. Please list two things that were unclear to you in class today

23

Introduction to Machine-Learning, Deep Learning & Artificial Intelligence (AI)

24

History of AI & vital force

Ancient Greece
Pygmalion & Galatea

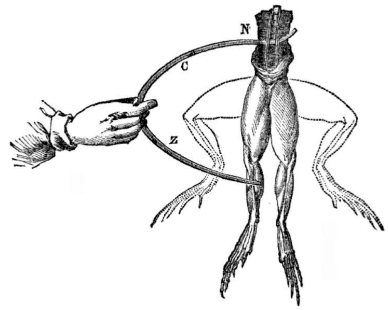
16th Century Europe
Jewish Folklore

19th Century Italy
Carlo Collodi

Neural networks

25

Is vital force mysterious?



Galvani, circa 1780

Neural networks

26

History of Mechanical AI

Mesopotamia,
Egypt, Greece, China

1642
Pascal
Pascaline, adder

1694
Leibniz step reckoner
4 basic arithmetic
operations

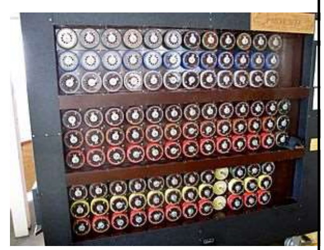
1801
Jacquard
Weaving loom w/
punch cards

Neural networks

27

Electronic AI in WWII

Enigma Machine



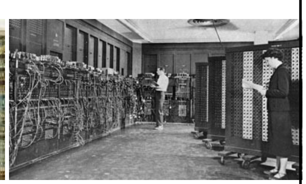
Bletchley Park “Bombe”

Neural networks

28

Electronic AI after WWII

1945, Mark I
First semi-electronic computer
Performed arithmetic faster than
most humans



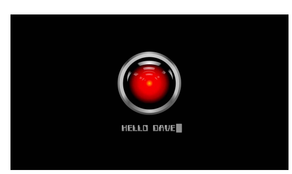
1946, ENIAC
1000x faster Mark 1
3x24 meters, 30,000 Kg

Neural networks

29

Modern Conceptions of AI

AI can be anthropomorphic
(C3PO, Star Wars)



Or not
(HAL 9000, Space Odyssey)

Neural networks

30

Software: Algorithmic advances

- 1952 – first checkers learning program (Samuel, IBM)
- 1957 – Perceptron, simplest neural network (Rosenblatt)
- 1967 – Nearest neighbor for pattern recognition
- 1981 – Explanation-based learning of rules from examples
- 1990's – ML formalized: applied to data mining, adaptive software, …
- 2000's – ML everywhere: used in technology, science, engineering, ...

31

Artificial intelligence (AI)

- Term AI coined in 1956 conference at Dartmouth
- Initial optimism: "within a generation… the problem of creating 'artificial intelligence" will substantially be solved" (Minsky, 1967)
- Attempts to make general artificial intelligence failed. Rule-based programs successful in specific fields: word problems in math, proving geometry theorems, ...

32

Introduction to Data Mining

33

Data-processing pipeline

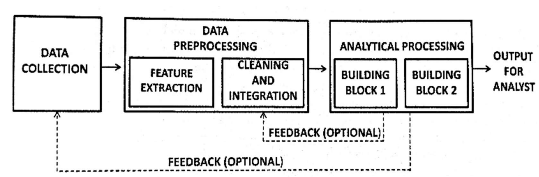


Image Source

34

Data-processing pipeline

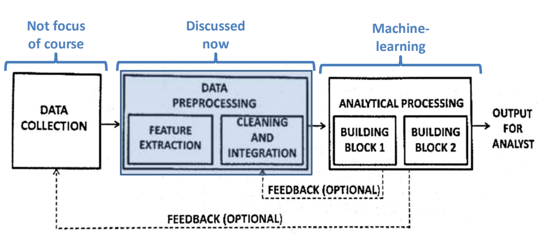


Image Source

35

Data processing: feature extraction

- Select features of dataset for further analysis

	Date	RegionName	InventorySeasonallyAdjusted	MedianListingPricePerSqft
2346125	10/31/10	acworthcobbga	1193	85.705263
2363168	11/30/10	acworthcobbga	1183	84.719280
2380867	12/31/10	acworthcobbga	1179	82.703214
2398571	1/31/11	acworthcobbga	1182	81.462677
2416287	2/28/11	acworthcobbga	1204	79.934225

36

Data processing: feature extraction

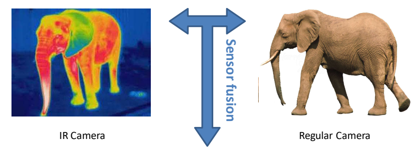
- Select features of dataset for further analysis
- Sometimes data needs combining into features

Date	RegionName	InventorySeasonallyAdjusted_AllHomes	InventoryRaw_AllHomes	MedianListingPricePerSqft_3Bedroom	MedianListingPricePerSqft_4Bedroom	MedianListingPricePerSqft_5BedroomOrMore
10/31/10	acworthcobbga	1193	115	83.368095	85.166538	102.055259
11/30/10	acworthcobbga	1183	116	81.132555	84.830942	100.167294
12/31/10	acworthcobbga	1179	117	78.709236	82.995195	101.900126
1/31/11	acworthcobbga	1182	111	78.723404	80.395795	095.128279
2/28/11	acworthcobbga	1204	117	76.523077	79.317641	094.220904

37

Data processing: feature extraction

- Select features of dataset for further analysis
- Sometimes data needs combining into features



38

Data processing: Data cleaning

- Almost all data types contain errors (or at least inaccuracies)

	Date	RegionName	InventorySeasonallyAdjusted	MedianListingPricePerSqft
2346125	10/31/10	acworthcobbga	1193	85.705263
2363168	11/30/10	acworthcobbga	1183	84.719280
2380867	12/31/10	acworthcobbga	1179	82.703214
2398571	1/31/11	acworthcobbga	1182	81.462677
2416287	2/28/11	acworthcobbga	1204	79.934225

39

Data processing: Data cleaning

- Almost all data types contain errors (or at least inaccuracies)

	Date	RegionName	InventorySeasonallyAdjusted	MedianListingPricePerSqft
2346125	10/31/10	acworthcobbga	1193	85.705263
2363168	11/30/10	acworthcobbga	1183	**847.19280**
2380867	12/31/10	acworthcobbga	1179	82.703214
2398571	1/31/11	acworthcobbga	1182	81.462677
2416287	2/28/11	acworthcobbga	1204	79.934225

40

Data processing: Data cleaning Handling missing entries

1. Eliminating record

	Date	RegionName	InventorySeasonallyAdjusted	MedianListingPricePerSqft
2346125	10/31/10	acworthcobbga	1193	85.705263
~~2363168~~	~~11/30/10~~	~~acworthcobbga~~	~~1183~~	~~NaN~~
2380867	12/31/10	acworthcobbga	1179	82.703214
2398571	1/31/11	acworthcobbga	1182	81.462677
2416287	2/28/11	acworthcobbga	1204	79.934225

41

Data processing: Data cleaning Handling missing entries

1. Eliminating record: problematic when many entries have NaNs

	Date	RegionName	InventorySeasonallyAdjusted	MedianListingPricePerSqft
2346125	10/31/10	acworthcobbga	1193	85.705263
~~2363168~~	~~11/30/10~~	~~acworthcobbga~~	~~1183~~	~~NaN~~
2380867	12/31/10	acworthcobbga	1179	82.703214
2398571	1/31/11	acworthcobbga	1182	81.462677
2416287	2/28/11	acworthcobbga	1204	79.934225

42

Data processing: Data cleaning
Handling missing entries

2. Estimating missing value (from other values)

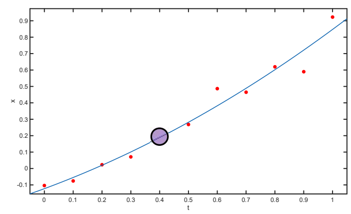
	Date	RegionName	InventorySeasonallyAdjusted	MedianListingPricePerSqft
2346125	10/31/10	acworthcobbga	1193	85.705263
2363168	11/30/10	acworthcobbga	1183	X
2380867	12/31/10	acworthcobbga	1179	82.703214
2398571	1/31/11	acworthcobbga	1182	81.462677
2416287	2/28/11	acworthcobbga	1204	79.934225

X = (85.705263+82.703214+81.462677+79.934225)/4= 82.451345

44

Data processing: Data cleaning
Handling missing entries

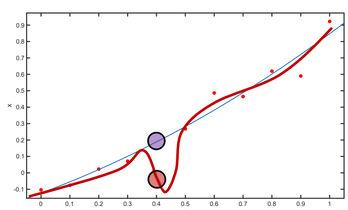
2. Estimating missing value (from other values)



45

Data processing: Data cleaning
Handling missing entries

2. Estimating missing value assumes: underlying model



46

Data processing: Data cleaning
Missing entries

3. Designing analytics to work with missing data: not always works, especially when using off-the-shelf code (increasingly common)

	Date	RegionName	InventorySeasonallyAdjusted	MedianListingPricePerSqft
2346125	10/31/10	acworthcobbga	1193	85.705263
2363168	11/30/10	acworthcobbga	1183	NaN
2380867	12/31/10	acworthcobbga	1179	82.703214
2398571	1/31/11	acworthcobbga	1182	81.462677
2416287	2/28/11	acworthcobbga	1204	79.934225

numpy.nanmean, numpy.std, …

47

Data processing: Data cleaning
Outlier detection

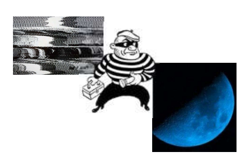
Definition of outlier: "An observation that deviates so much from the other observations as to arouse suspicions that it was generated by a different mechanism" (Hawkins, 1980)

	Date	RegionName	InventorySeasonallyAdjusted	MedianListingPricePerSqft
2346125	10/31/10	acworthcobbga	1193	85.705263
2363168	11/30/10	acworthcobbga	1183	**847.19280**
2380867	12/31/10	acworthcobbga	1179	82.703214
2398571	1/31/11	acworthcobbga	1182	81.462677
2416287	2/28/11	acworthcobbga	1204	79.934225

49

Data processing: Data cleaning
Outlier detection

- Outliers might arise from
  - noise
    - human error
    - sensor error
    - communication error
    - etc.
  - malicious activity,
  - rare event
- Detecting outliers necessitates model for normal data

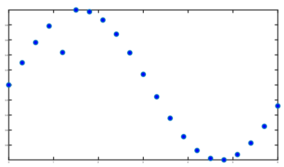


50

How to detect outliers?

- Easier when you plot your data
  - Always good idea to start by plotting your data
  - Can you spot the outlier?

0	0	3.3	-0.15775
0.3	0.29552	3.6	-0.44252
0.6	0.56464	3.9	-0.68777
0.9	0.78333	4.2	-0.87158
1.2	0.43204	4.5	-0.97753
1.5	0.99749	4.8	-0.99616
1.8	0.97385	5.1	-0.92581
2.1	0.86321	5.4	-0.77276
2.4	0.67546	5.7	-0.55069
2.7	0.42738	6	-0.27942
3	0.14112		

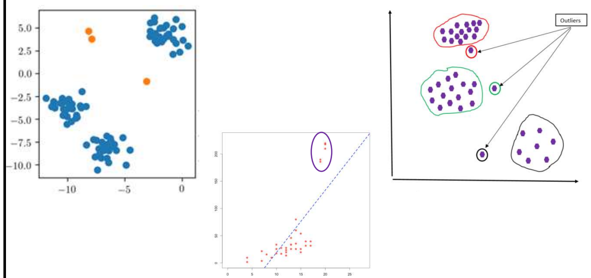


Neural networks

51

How to detect outliers?

- Easier when you plot your data
  - Always good idea to start by plotting your data

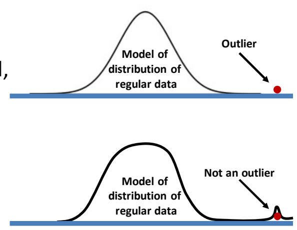


Neural networks

52

Data processing: Data cleaning Outlier detection

- Detecting outliers necessitates model for regular data
- Without reasonable model, outlier detection is ill-defined problem

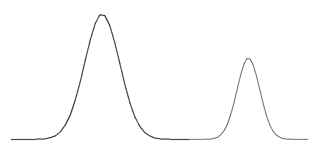


Neural networks

53

Dealing with bimodal distributions

- How to handle missing values with bi modal distribution? Cannot replace them with median
  - Can separate into 2 distributions and handle separately

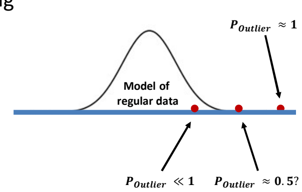


Neural networks

54

Data processing: Data cleaning Outlier detection

- Often better to calculate probability of sample being outlier—*outlier score*—than to make binary decision

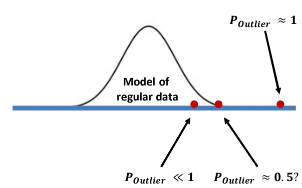


Neural networks

55

Data processing: Data cleaning Outlier detection

- Outlier detection & removal large, non-trivial topic
- Various methods exist
  - Extreme-value
  - Probabilistic
  - Clustering
  - Distance-based
  - Density-based
  - Information-theory
  - Component independence
  - …

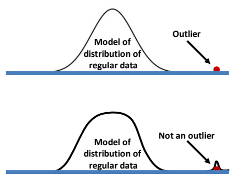


Neural networks

56

Recap of Data Mining

- No silver bullet to dealing with missing data
  - Removing it decreases available data
  - Interpolating requires some model
- Similarly, defining outliers requires some assumptions about underlying distribution



57

General Mathematical Framework of Machine-Learning

60

Goal of machine learning

Machine learning focuses on the problem of **prediction**.

Given a training set
$\{(\vec{x}_1, y_1), \ldots, (\vec{x}_n, y_n)\} \in \mathbb{R}^p \times \mathbb{R}$,
we learn a predictor function
$h_n : \mathbb{R}^p \to \mathbb{R}$
that predicts label y for some $\vec{x} \notin \{\vec{x}_1, \ldots, \vec{x}_n\}$—i.e., for $\vec{x}$ that was not in the training set.

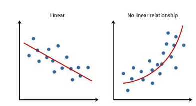
61

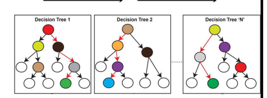
Goal of machine learning

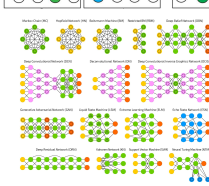
The predictor h_n is typically chosen from some function class $\mathcal{H}$.

$\mathcal{H}$ could be

- A set of linear regression functions
- A random forest with hyper-parameters in some subspace
- Neural networks of certain architecture, depth, and width
- …







62

Goal of machine learning

For some training set $\{(\vec{x}_1, y_1), \ldots, (\vec{x}_n, y_n)\}$, h_n could be chosen using empirical-risk minimization (ERM):

Here, an $h \in \mathcal{H}$ is chosen such that

$$h = \operatorname{argmin} \frac{1}{n} \sum_{i=1}^{n} \ell(h(\vec{x}_i), y_i)$$

where ℓ is a loss function. So, for ERM, h is chosen such that it minimizes the loss on the training set.

The loss function, ℓ, can be:

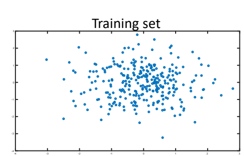
- Regression (squared loss): $\ell(y', y) = (y' - y)^2$
- Classification (0-1 loss): $\ell(y', y) = \begin{cases} 0, & y' = y \\ 1, & y' \neq y \end{cases}$

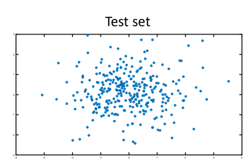
63

Goal of machine learning

The goal of machine learning is to find h_n with a small loss on data $(\vec{x}, y)$ not in the training set—i.e., to find h_n that generalizes well to unseen data.

We typically assume that the training set is drawn from some (unknown) probability distribution D over $\mathbb{R}^p \times \mathbb{R}$. And we evaluate h_n on test sample $(\vec{x}, y)$ independently drawn from the same probability distribution D.





64

Group exercise

In breakout rooms, we will now work on the following google doc:
https://docs.google.com/document/d/1KnPD10ETqpDFK18wBuJzb13U_IsDac8BE2fFkz-UHYM/edit?usp=sharing

Neural networks

65

General machine-learning problem

- Machine-learning is about function approximation
- The data $\overleftrightarrow{x} = \begin{bmatrix} x_{1,1} & \cdots & x_{p,1} \\ \vdots & \ddots & \vdots \\ x_{1,n} & \cdots & x_{p,n} \end{bmatrix}$ is assumed to be n samples of p dimensions each.

	name	mfr	type	calories	protein	fat	sodium	fiber	carbo	sugars	potass	vitamins	shelf	weight	cups
0	100%_Bran	N	C	70	4	1	130	10.0	5.0	6	280	25	3	1.00	0.33
1	100%_Natural_Bran	Q	C	120	3	5	15	2.0	8.0	8	135	0	3	1.00	1.00
2	All-Bran	K	C	70	4	1	260	9.0	7.0	5	320	25	3	1.00	0.33
3	All-Bran_with_Extra_Fiber	K	C	50	4	0	140	14.0	8.0	0	330	25	3	1.00	0.50
4	Almond_Delight	R	C	110	2	2	200	1.0	14.0	8	-1	25	3	1.00	0.75
5	Apple_Cinnamon_Cheerios	G	C	110	2	2	180	1.5	10.5	10	70	25	1	1.00	0.75
6	Apple_Jacks	K	C	110	2	0	125	1.0	11.0	14	30	25	2	1.00	1.00
7	Basic_4	G	C	130	3	2	210	2.0	18.0	8	100	25	3	1.33	0.75
8	Bran_Chex	R	C	90	2	1	200	4.0	15.0	6	125	25	1	1.00	0.67
9	Bran_Flakes	P	C	90	3	0	210	5.0	13.0	5	190	25	3	1.00	0.67

Neural networks

66

General machine-learning problem

- The input data ,$\overleftrightarrow{x}$, is typically associated with an output, $\vec{y}$, such that $\vec{y} = f(\overleftrightarrow{x})$ for some unknown f (or h) that we want to learn.

$$\vec{y} = f\left(\begin{bmatrix} x_{1,1} & \cdots & x_{p,1} \\ \vdots & \ddots & \vdots \\ x_{1,n} & \cdots & x_{p,n} \end{bmatrix}\right)$$

Neural networks

67

General machine-learning problem

- Machine-learning can be boiled down to function approximation
- The data $\overleftrightarrow{x} = \begin{bmatrix} x_{1,1} & \cdots & x_{p,1} \\ \vdots & \ddots & \vdots \\ x_{1,n} & \cdots & x_{p,n} \end{bmatrix}$ is typically n samples of p dimensions each
- The labels are typically one dimensional for each sample. So, $\vec{y}$ is (n x 1)
- So $\vec{y} = f(\overleftrightarrow{x})$ is the machine-learning problem, and the function to approximate is f

Neural networks

68

General machine-learning problem

- We want to learn

relationship between $\boldsymbol{\overleftrightarrow{x}} = \begin{bmatrix} \boldsymbol{x_{1,1}} & \cdots & \boldsymbol{x_{p,1}} \\ \vdots & \ddots & \vdots \\ \boldsymbol{x_{1,n}} & \cdots & \boldsymbol{x_{p,n}} \end{bmatrix}$

$\vec{y} = \boldsymbol{f}(\boldsymbol{\overleftrightarrow{x}})$ is the general machine-learning problem

- Hence, function we want approximate is $\boldsymbol{f}$ when given $\boldsymbol{\overleftrightarrow{x}}$ and $\vec{y}$.
- After computing $\hat{\boldsymbol{f}}$, we can find $\hat{\vec{y}'} = \hat{\boldsymbol{f}}\left(\overleftrightarrow{x'}\right)$

Neural networks

69

Example data: find relation between $\boldsymbol{\overleftrightarrow{x}}$ and $\vec{\boldsymbol{y}}$

Health rating

rating
68.402973
33.983679
59.425505
93.704912
34.384843
29.509541
33.174094
37.038562
49.120253
53.313813

	name	mfr	type	calories	protein	fat	sodium	fiber	carbo	sugars	potass	vitamins	shelf	weight	cups
0	100%_Bran	N	C	70	4	1	130	10.0	5.0	6	280	25	3	1.00	0.33
1	100%_Natural_Bran	Q	C	120	3	5	15	2.0	8.0	8	135	0	3	1.00	1.00
2	All-Bran	K	C	70	4	1	260	9.0	7.0	5	320	25	3	1.00	0.33
3	All-Bran_with_Extra_Fiber	K	C	50	4	0	140	14.0	8.0	0	330	25	3	1.00	0.50
4	Almond_Delight	R	C	110	2	2	200	1.0	14.0	8	-1	25	3	1.00	0.75
5	Apple_Cinnamon_Cheerios	G	C	110	2	2	180	1.5	10.5	10	70	25	1	1.00	0.75
6	Apple_Jacks	K	C	110	2	0	125	1.0	11.0	14	30	25	2	1.00	1.00
7	Basic_4	G	C	130	3	2	210	2.0	18.0	8	100	25	3	1.33	0.75
8	Bran_Chex	R	C	90	2	1	200	4.0	15.0	6	125	25	1	1.00	0.67
9	Bran_Flakes	P	C	90	3	0	210	5.0	13.0	5	190	25	3	1.00	0.67

$\vec{\boldsymbol{y}}$ $\boldsymbol{\overleftrightarrow{x}}$

Neural networks

70

Example data

rating
68.402973
33.983679
59.425505
93.704912
34.384843
29.509541
33.174094
37.038562
49.120253
53.313813

	name	mfr	type	calories	protein	fat	sodium	fiber	carbo	sugars	potass	vitamins	shelf	weight	cups
0	100%_Bran	N	C	70	4	1	130	10.0	5.0	6	280	25	3	1.00	0.33
1	100%_Natural_Bran	Q	C	120	3	5	15	2.0	8.0	8	135	0	3	1.00	1.00
2	All-Bran	K	C	70	4	1	260	9.0	7.0	5	320	25	3	1.00	0.33
3	All-Bran_with_Extra_Fiber	K	C	50	4	0	140	14.0	8.0	0	330	25	3	1.00	0.50
4	Almond_Delight	R	C	110	2	2	200	1.0	14.0	8	-1	25	3	1.00	0.75
5	Apple_Cinnamon_Cheerios	G	C	110	2	2	180	1.5	10.5	10	70	25	1	1.00	0.75
6	Apple_Jacks	K	C	110	2	0	125	1.0	11.0	14	30	25	2	1.00	1.00
7	Basic_4	G	C	130	3	2	210	2.0	18.0	8	100	25	3	1.33	0.75
8	Bran_Chex	R	C	90	2	1	200	4.0	15.0	6	125	25	1	1.00	0.67
9	Bran_Flakes	P	C	90	3	0	210	5.0	13.0	5	190	25	3	1.00	0.67

What we want to predict for unseen data, $\vec{y}$

What we know about the features of the data, $\overleftrightarrow{x}$

71

General machine-learning problem

- The input data ,$\overleftrightarrow{x}$, is typically associated with an output, $\vec{y}$, such that $\vec{y} = f(\overleftrightarrow{x})$ for some unknown f that we want to approximate (learn).

Health rating

rating
68.402973
33.983679
59.425505
93.704912
34.384843
29.509541
33.174094
37.038562
49.120253
53.313813

$= f$

calories	protein	fat	sodium	fiber	carbo	sugars	potass	vitamins	shelf	weight	cups
70	4	1	130	10.0	5.0	6	280	25	3	1.00	0.33
120	3	5	15	2.0	8.0	8	135	0	3	1.00	1.00
70	4	1	260	9.0	7.0	5	320	25	3	1.00	0.33
50	4	0	140	14.0	8.0	0	330	25	3	1.00	0.50
110	2	2	200	1.0	14.0	8	-1	25	3	1.00	0.75
110	2	2	180	1.5	10.5	10	70	25	1	1.00	0.75
110	2	0	125	1.0	11.0	14	30	25	2	1.00	1.00
130	3	2	210	2.0	18.0	8	100	25	3	1.33	0.75
90	2	1	200	4.0	15.0	6	125	25	1	1.00	0.67
90	3	0	210	5.0	13.0	5	190	25	3	1.00	0.67

$\underbrace{\quad}_{\vec{y}}$ $\qquad \overleftrightarrow{x}$

72

Supervised Machine-Learning

- Given a set of samples where $\vec{y} = f(\overleftrightarrow{x})$ for some unknown f, supervised machine-learning strives to learn a good approximation of f, $\hat{f}$. The goal is for the approximation $\hat{f}$ to be a good approximation of f such that it could well predict $\overrightarrow{y'} = \hat{f}\left(\overleftrightarrow{x'}\right)$ for some $\overleftrightarrow{x'}$ different than $\overleftrightarrow{x}$.
  - **Regression problem:** $\vec{y}$ is *continuous*
  - **Classification problem:** $\vec{y}$ is *categorical*
    - $\vec{y}$ are typically termed *labels* in this case

73

General supervised machine-learning problem

$$\vec{y} = f\left(\begin{bmatrix} x_{1,1} & \cdots & x_{p,1} \\ \vdots & \ddots & \vdots \\ x_{1,n} & \cdots & x_{p,n} \end{bmatrix}\right)$$

Health rating

rating
68.402973
33.983679
59.425505
93.704912
34.384843
29.509541
33.174094
37.038562
49.120253
53.313813

$= f$

calories	protein	fat	sodium	fiber	carbo	sugars	potass	vitamins	shelf	weight	cups
70	4	1	130	10.0	5.0	6	280	25	3	1.00	0.33
120	3	5	15	2.0	8.0	8	135	0	3	1.00	1.00
70	4	1	260	9.0	7.0	5	320	25	3	1.00	0.33
50	4	0	140	14.0	8.0	0	330	25	3	1.00	0.50
110	2	2	200	1.0	14.0	8	-1	25	3	1.00	0.75
110	2	2	180	1.5	10.5	10	70	25	1	1.00	0.75
110	2	0	125	1.0	11.0	14	30	25	2	1.00	1.00
130	3	2	210	2.0	18.0	8	100	25	3	1.33	0.75
90	2	1	200	4.0	15.0	6	125	25	1	1.00	0.67
90	3	0	210	5.0	13.0	5	190	25	3	1.00	0.67

$\vec{y}$ $\qquad \overleftrightarrow{x}$

What we want to predict for unseen data, $\vec{y}$

What we know about the features of the data, $\overleftrightarrow{x}$

74

Unsupervised Machine-Learning

- Only the data, $\overleftrightarrow{x}$, is given—without labels, $\vec{y}$.
- The goal is to model $\overleftrightarrow{x}$ differently—or find another representation for $\overleftrightarrow{x}$ or find structure in $\overleftrightarrow{x}$—under some assumptions
  - **Clustering**: discover the inherent groupings in the data samples (under some assumed metric)
  - **Dimensionality reduction**: find lower-dimensional representation of data (under some assumed metric)—change feature-space

75

General unsupervised machine-learning problem

$$\begin{bmatrix} x_{1,1} & \cdots & x_{p,1} \\ \vdots & \ddots & \vdots \\ x_{1,n} & \cdots & x_{p,n} \end{bmatrix}$$

calories	protein	fat	sodium	fiber	carbo	sugars	potass	vitamins	shelf	weight	cups
70	4	1	130	10.0	5.0	6	280	25	3	1.00	0.33
120	3	5	15	2.0	8.0	8	135	0	3	1.00	1.00
70	4	1	260	9.0	7.0	5	320	25	3	1.00	0.33
50	4	0	140	14.0	8.0	0	330	25	3	1.00	0.50
110	2	2	200	1.0	14.0	8	-1	25	3	1.00	0.75
110	2	2	180	1.5	10.5	10	70	25	1	1.00	0.75
110	2	0	125	1.0	11.0	14	30	25	2	1.00	1.00
130	3	2	210	2.0	18.0	8	100	25	3	1.33	0.75
90	2	1	200	4.0	15.0	6	125	25	1	1.00	0.67
90	3	0	210	5.0	13.0	5	190	25	3	1.00	0.67

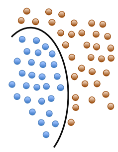
$\overleftrightarrow{x}$

Find latent relations within $\overleftrightarrow{x}$

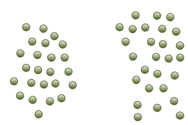
76

Supervised vs. unsupervised learning

- Supervised learning



- Unsupervised learning



77

Supervised vs. Unsupervised Learning

- Many problems in machine learning can be thought of as supervised or unsupervised
- But many other problems are in between
  - A lot of data, only some of it has labels
    - Under some assumption (e.g., smoothness) unlabeled samples can learn labeling from labeled ones
    - Example: Photo archive where only some data is labeled

78

Example: Linear Regression

In general (multiple) linear-regression, we have input data $\vec{x}$ and output $\vec{y}$ and we assume

$$\vec{y} = \vec{x}\,\vec{\beta} + \vec{\varepsilon}.$$

So, we approximate using $\hat{\vec{\beta}}$:

$$\hat{\vec{y}} = \vec{x}\,\hat{\vec{\beta}}.$$

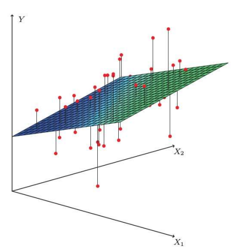
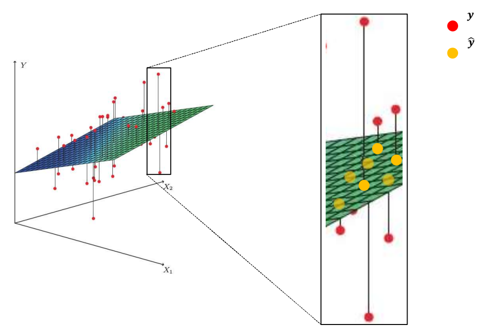


Image Source

79

Example: Linear Regression



80

Artificial Intelligence & machine learning

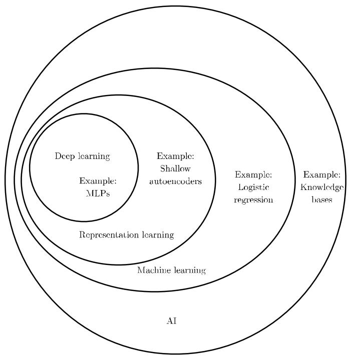
81

Rule-based vs. learning from examples

- Rule-based learning: try to understand logical foundation of problems & teach computers to follow those rules
  - Limited success in specific domains
- Learning from examples: expose computer to examples & let computer learn rule from examples
  - Much more successful generally
  - **This is what machine-learning is all about**

82

Deep learning & AI



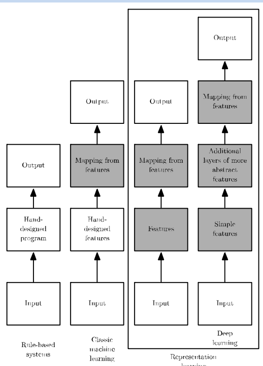
83

Deep learning & AI: rules & representations

Types of AI	Rule-Based	Example-Based
Given Representation	**Classic AI**	**Classic Machine Learning**
Learn Representation		**Deep Learning**

84

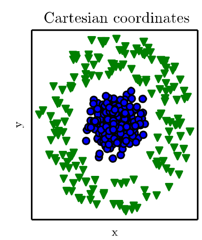
Deep learning & AI: rules & representations



85

Representation in learning

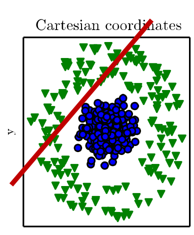
- Find a line separating **blue** from **green** samples



86

Representation in learning

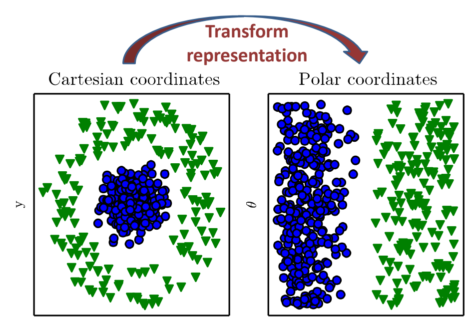
- Find a line separating **blue** from **green** samples



Impossible ☹

87

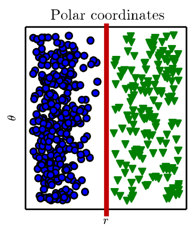
Representation in learning



88

Representation in learning

- Find a line separating **blue** from **green** samples



Easy ☺