Machine Learning - Intro

Aarti Singh

Machine Learning 10-701
Jan 18, 2023

Teaching team

Instructors:

Aarti

TAs: Nari Johnson

Dhruv Malik

Yusha Liu

Education Associate:

Joshmin

Admin:

Mary

Logistics

Lectures:	Mon, Wed	9:30-10:50 am	POS 153
Recitations:	Fri	9:30-10:50 am	POS 153
Office hours:	Mon, Tue, Wed, Thurs	(check website)	

Lectures will be recorded. Strictly for your use only.

Office hours will NOT be recorded.

Videos: [Canvas]

Logistics

Webpage: https://www.cs.cmu.edu/~aarti/Class/10701_Spring23
Syllabus, policies, schedule of lectures, recitations, office hours, slides, reading material, homeworks, …

Piazza: http://piazza.com/cmu/spring2023/10701
announcements, questions for Teaching team, discussion forum for students

Homework submission: Gradescope

Grades: Canvas

Expectations

- In-person attendance, videos for review (or emergencies) only, zoom available for medical or other exceptional cases only
  - Please stay home if sick

- Interact!
  - Ask questions in class by raising hand
  - Respond to questions in class by raising hand
  - In-class polls

- In-person Office hours (starting next week)

Recitations

- Strongly recommended
  - Brush up pre-requisites
  - Hands-on exercises
  - Review material (difficult topics, clear misunderstandings, extra new topics, HW and exam solutions)
  - Ask questions

- 1st Probability Review - **FRIDAY**

  by Yusha

  Fri Jan 20 9:30-10:50 am POS 153

Grading

- Grading
  - 5 homework assignments (4 x 12% + 8% = 58%)
  - 1 depth exercise (12%)
  - 1 midterm, 1 final: (13+16 = 29%)
    - midterm - Mar 1 during class
  - Participation (3%)

- Late days
  - total 5 across homeworks, no more than 2 per HW
  - 50% credit for 24 hrs after late days
  - late days are for unforeseen situations (interviews, conference, etc.), do NOT include them in your plan

Homeworks ~~& QnAs~~

- Collaboration
  - You may **discuss** the questions
  - Each student writes their own answers, without copying from discussion notes or ongoing conversations
  - Each student must write their own code for the programming part
  - **Don't search for answers on the web, Google, previous years' homeworks, etc.**
    - please ask us if you are not sure if you can use a particular reference
    - list resources used (references, discussants) on top of submitted homework
- Homeworks are hard, start early ☺
- Due on gradescope

Waitlist + Audits + Pass/Fail

- Waitlist

  we'll let everyone in

  keep attending lectures, recitations and office hours and doing HW

- Audits and Pass/Fail

  Audits allowed (with some requirement)

  Pass/Fail allowed

About the course

- Machine Learning **Algorithms, Theory, Principles and Applications**
  - Classification: Naïve Bayes, Logistic Regression, Neural Networks, Support Vector Machines, k-NN, Decision Trees, …
  - Regression: Linear regression, Kernel regression, Nonparametric regression, …
  - Unsupervised methods: Kernel density estimation, mixture models, clustering, PCA, …
  - Graphical models, Hidden Markov Models, Reinforcement learning
  - Core concepts: Probability, Optimization, Theory, Model selection, overfitting, bias-variance tradeoffs, Fairness …

- See **tentative** lecture schedule on webpage – MAY CHANGE
- Material: Class slides/videos + Reading material

Recommended textbooks

- Textbooks (Recommended, not required):

  Pattern Recognition and Machine Learning, Christopher Bishop (available online)

  Machine Learning: A probabilistic perspective, Kevin Murphy

  (available online)

  Machine Learning, Tom Mitchell

  The elements of statistical learning: Data mining, inference and prediction, Trevor Hastie, Robert Tibshirani, Jerome Friedman

Pre-requisites

- **Assume mathematical maturity**
  - Basic Probability and Statistics

    Probability distributions – discrete and continuous, Mean, Variance, Conditional probabilities, Bayes rule, Central limit theorem…
  - Programming (python) and principles of computing
  - Multivariate Calculus

    Derivatives, integrals of multi-variate functions
  - Linear Algebra

    Matrix inversions, eigendecomposition, …

- **Tutorial videos**
  - Probability, Calculus, Functional Analysis, SVD

    https://www.youtube.com/channel/UC7gOYDYEgXG1yIH_rc2LgOw/playlists
  - Linear Algebra

    http://www.cs.cmu.edu/~zkolter/course/linalg/index.html

- **Self-assessment test** on webpage

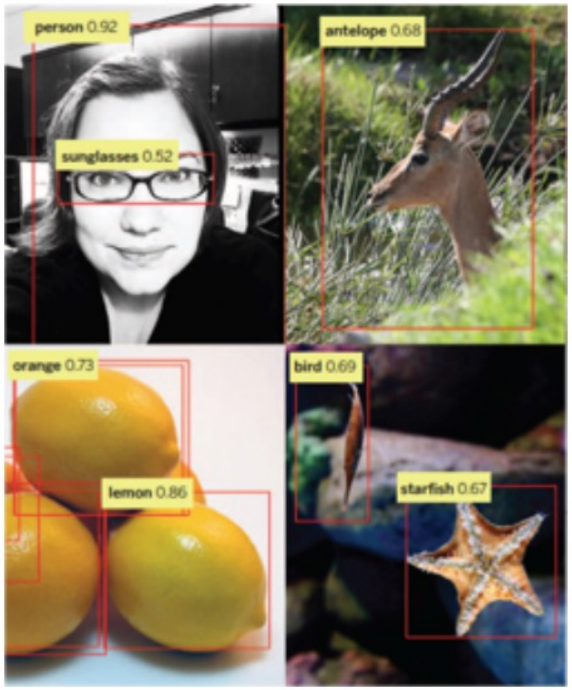
Related courses

- Related courses – Intro to ML algorithms and principles
  - 10-301 – Undergrad version for non-SCS majors 10-315
  - 10-601 – Masters version
  - 10-701 – PhD version
  - 10-715 – PhD students doing research in machine learning (hardest, most mathematical)

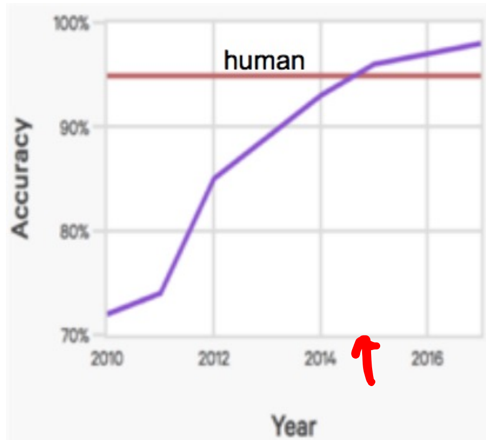
Other related courses:

- 10-606, 10-607 – Math background for ML
- 10-605, 10-805 – Machine Learning with Large Datasets
- 11-663 – Machine Learning in Practice (ML software)
- 10-702, 10-704, 10-707, 10-708, 10-709, 15-859(A/B) – related advanced topics

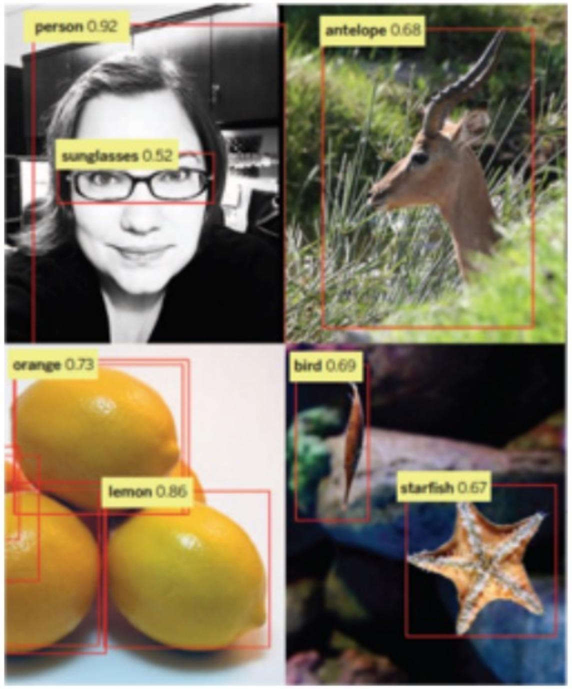
Machine Learning in Action



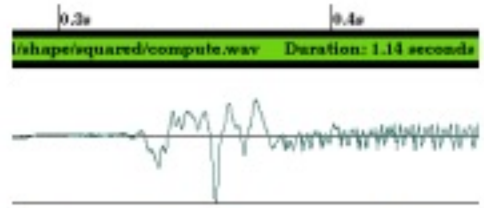
Computer vision



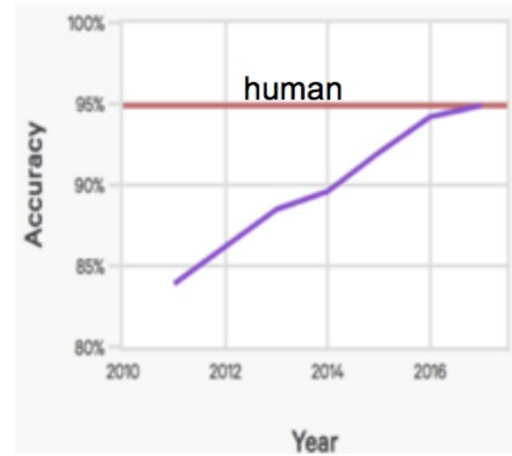
Machine Learning in Action



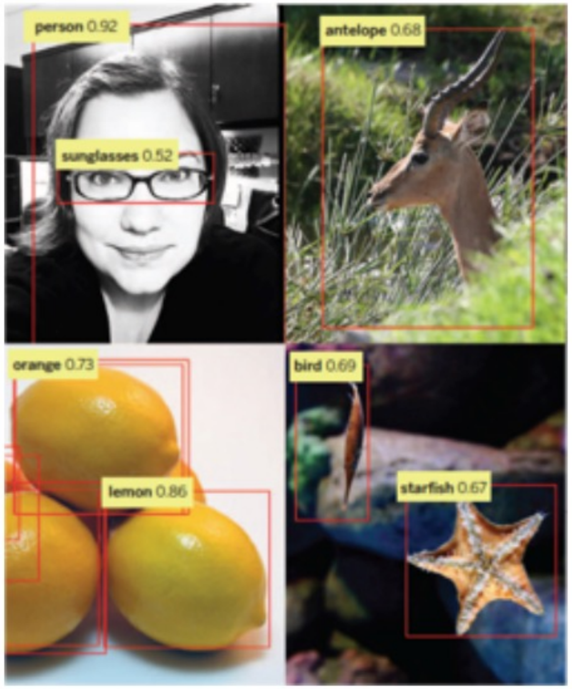
Computer vision



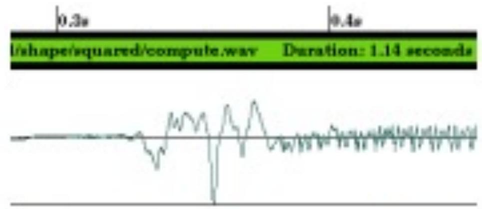
Speech Recognition



Machine Learning in Action



Computer vision

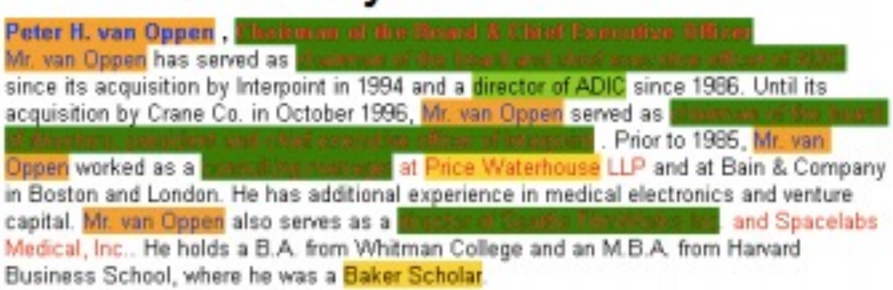


Speech Recognition

Robotic control

Games & Reasoning

Text analysis


Peter H. van Oppen , [illegible]
Mr. van Oppen has served as [illegible]
since its acquisition by Interpoint in 1994 and a director of ADIC since 1986. Until its acquisition by Crane Co. in October 1996, Mr. van Oppen served as [illegible] . Prior to 1985, Mr. van Oppen worked as a [illegible] at Price Waterhouse LLP and at Bain & Company in Boston and London. He has additional experience in medical electronics and venture capital. Mr. van Oppen also serves as a [illegible] and Spacelabs Medical, Inc.. He holds a B.A. from Whitman College and an M.B.A. from Harvard Business School, where he was a Baker Scholar.

Protein folding

Machine Learning in Action

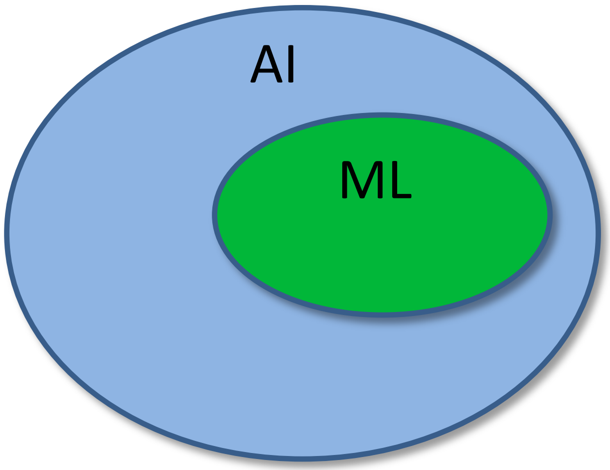
- How have you interacted with ML in your daily life so far?

ML is ubiquitous

- Wide applicability
- Software too complex to write by hand
- Improved machine learning algorithms
- Improved data capture, networking, faster computers
- Demand for self-customization to user, environment

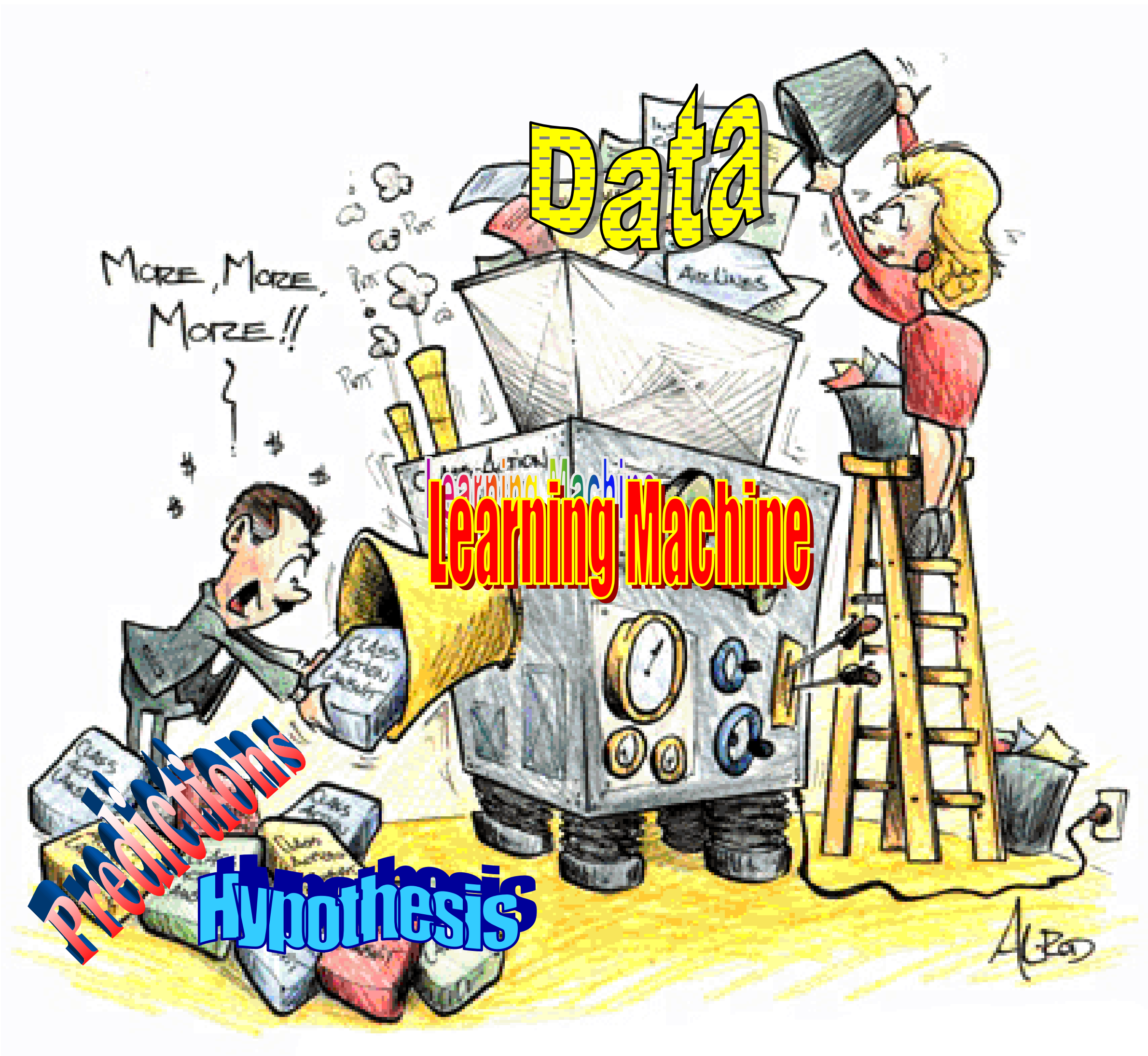
AI: develop intelligent agents

ML: learn to generalize using data



Fun begins …

What is Machine Learning?



Data

⇩

Learning algorithm

⇩

Knowledge

What is Machine Learning?

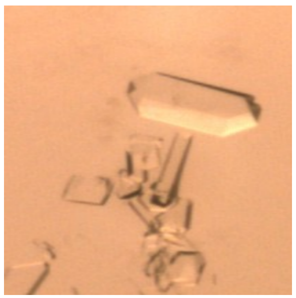
Design and Analysis of algorithms that

- improve their <u>performance</u>
- at some <u>task</u>
- with <u>experience</u>

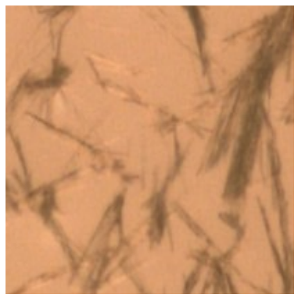
(experience) **(performance on task)**

Human learning

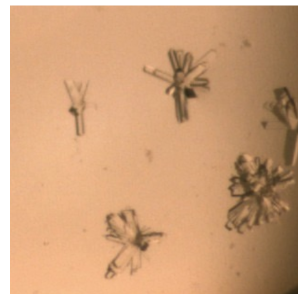
Task: Learning stage of protein crystallization



Crystal

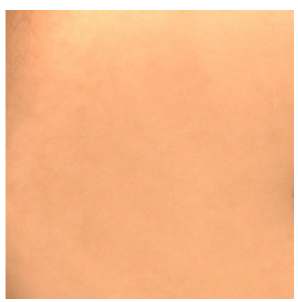


Needle

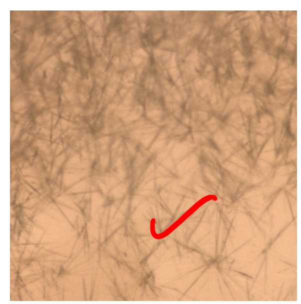


Tree

Tree



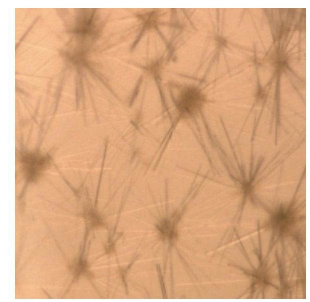
Empty



Needle

Experience

- Predict the label of the test image?



?

Performance

Tasks, Experience, Performance

Tasks, Experience, Performance

Machine Learning Tasks

Broad categories -

- **Supervised learning**

  Classification, Regression

- **Unsupervised learning**

  Density estimation, Clustering, Dimensionality reduction

- **Graphical models, Hidden Markov models**
- **Reinforcement learning**
- Semi-supervised learning
- Active learning
- Many more …

Supervised Learning

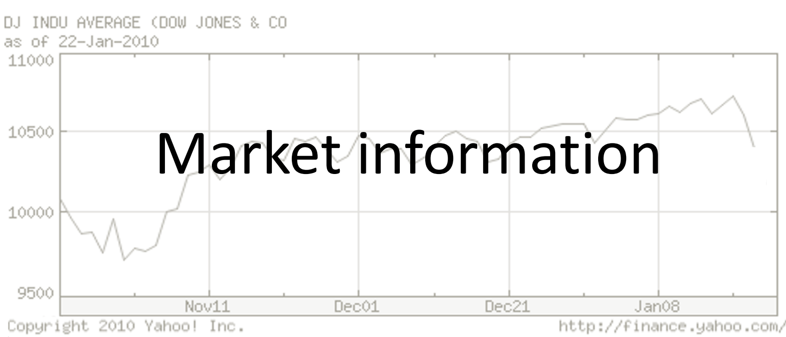
Input $X \in \mathcal{X}$

Label $Y \in \mathcal{Y}$

Document/Article

"Sports"
"News"
"Science"
…

Discrete Labels
Classification



Market information

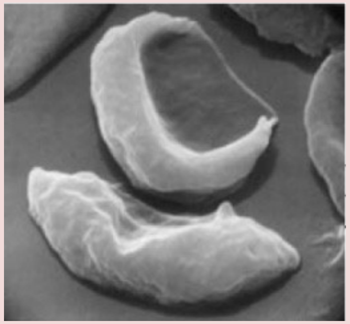
Share Price
"$ 24.50"

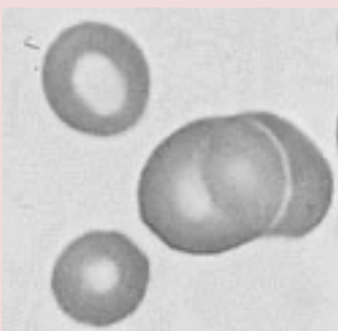
Continuous Labels
Regression

Task: Given $X \in \mathcal{X}$, predict $Y \in \mathcal{Y}$.

$\equiv$ Construct **prediction rule** $f : \mathcal{X} \rightarrow \mathcal{Y}$

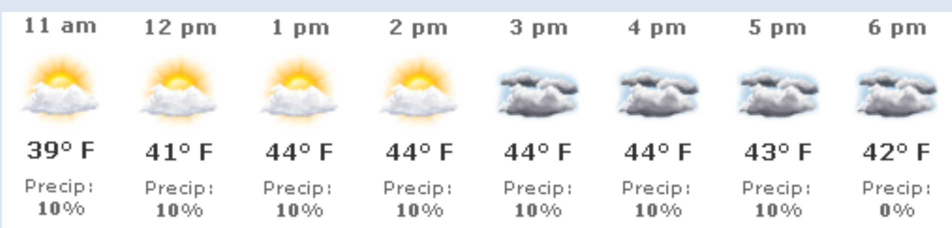
Classification or Regression?



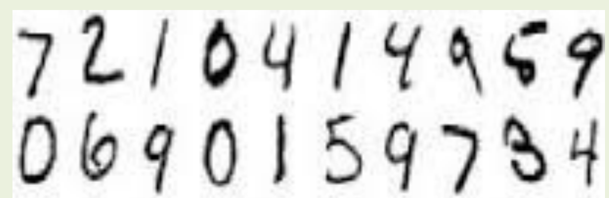


Medical Diagnosis

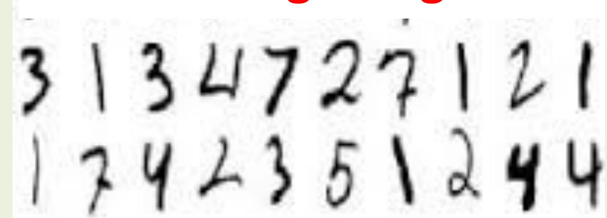
"Anemic"
"Healthy"



Weather prediction



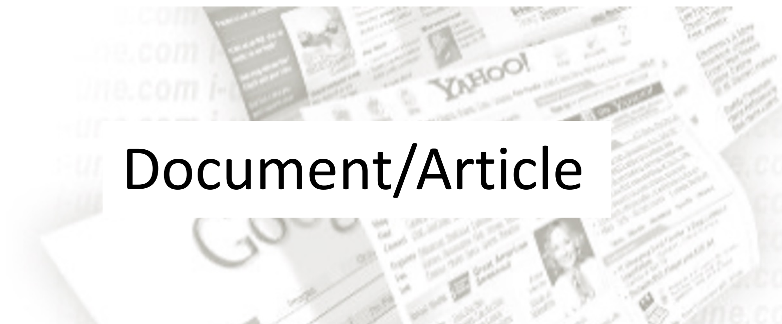
Handwriting recognition



Unsupervised Learning

Aka "learning without a teacher"

Input $X \in \mathcal{X}$



Word distribution
(Probability of a word)

$p(X) = p\left(X = \begin{bmatrix} \text{the} \\ \text{an} \\ \vdots \end{bmatrix}\right)$

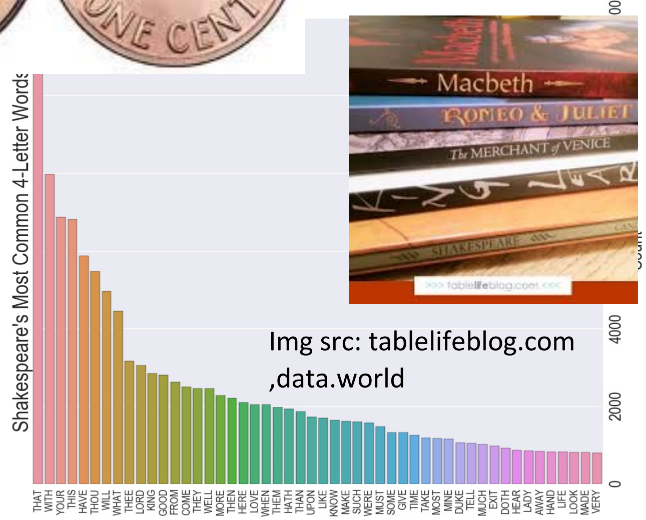
Task: Given $X \in \mathcal{X}$, learn $f(X)$.

Unsupervised Learning

Learning a Distribution

Bias of a coin

Distribution of words in text



Img src: tablelifeblog.com ,data.world

Clustering

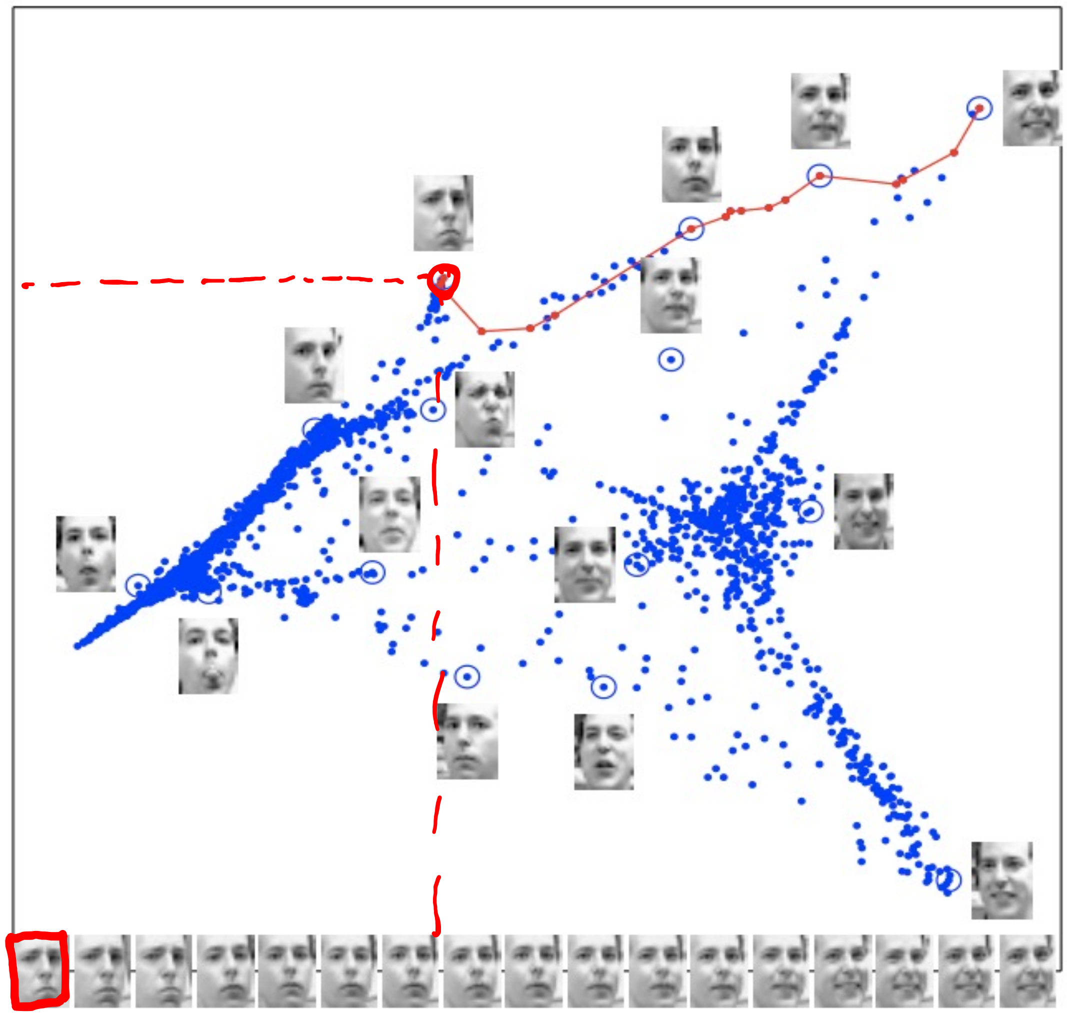
Unsupervised Learning

Dimensionality Reduction/Embedding

[Saul & Roweis '03]

Images have thousands or millions of pixels.

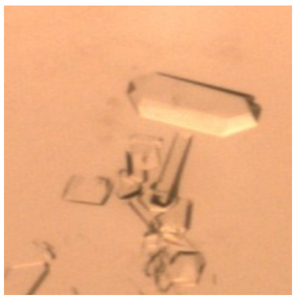
Can we give each image a small set of coordinates, such that similar images are near each other?



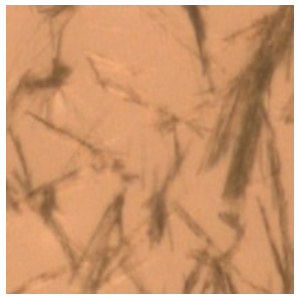
Tasks, **Experience**, Performance

Experience = Training Data

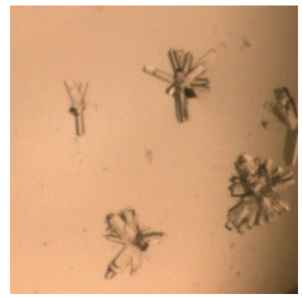
Task: Learning stage of protein crystallization



Crystal

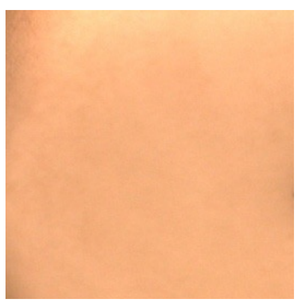


Needle

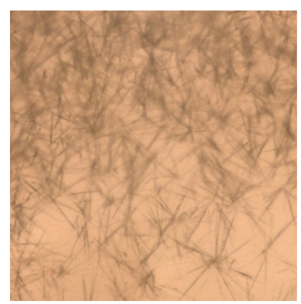


Tree

Tree

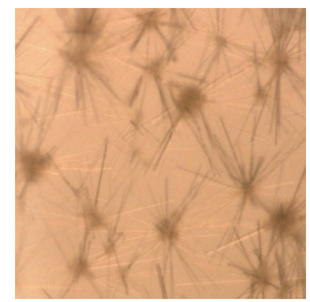


Empty



Needle

Experience

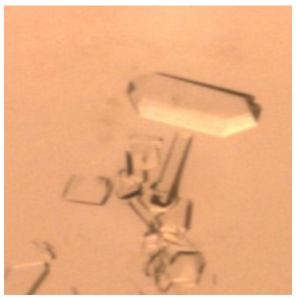


?

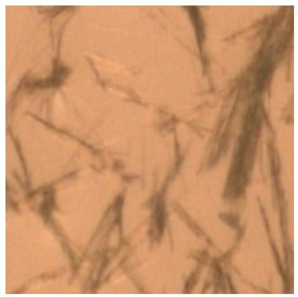
Performance

Training Data $\neq$ Test Data

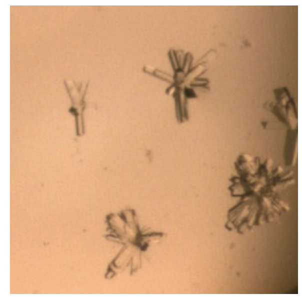
Task: Learning stage of protein crystallization



Crystal

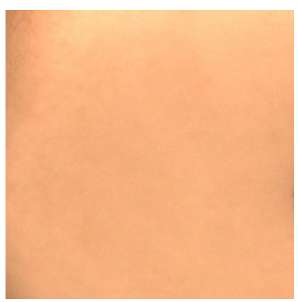


Needle

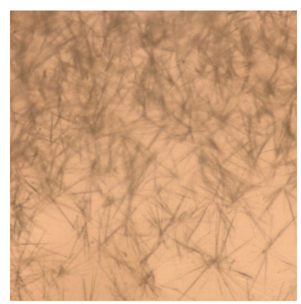


Tree

Tree

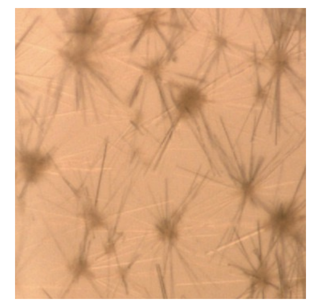


Empty



Needle

Experience



?

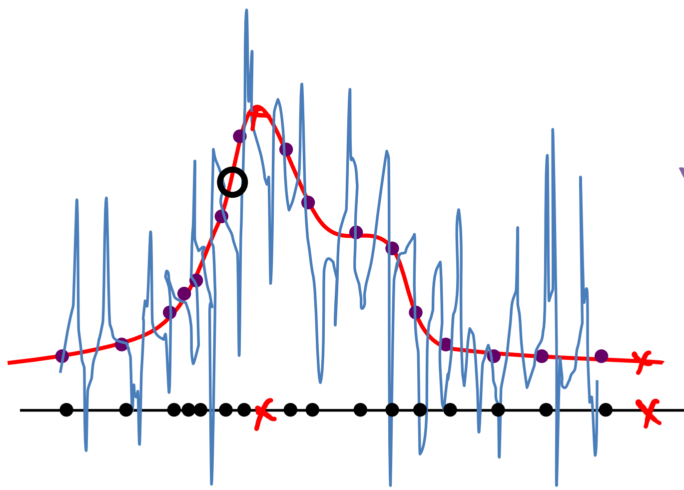
Performance

Generalization & Overfitting

A good ML algorithm

should: **generalize** aka perform well on test data

should not: **overfit** the training data (gap between training & test error)



Y = part density

X = Roller speed

Critical to report testing and NOT training accuracy

Regression example: Blood samples were collected for 100 subjects who were administered a covid-19 vaccine.

An ML algorithm was trained to predict the number of antibodies in the blood of these 100 subjects given their profiles.

The normalized mean square error of the trained model was 0.001 for predicting the antibodies in these 100 subjects.

- Is this a good model?

10 more subjects were then recruited and the normalized mean square error of the model's predictions of antibodies for these 10 subjects was 0.35.

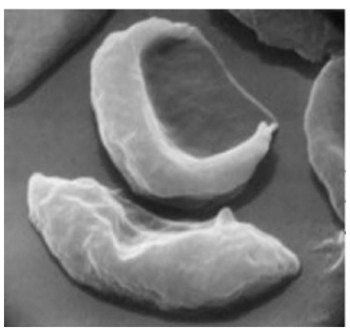
Tasks, Experience, **Performance**

Performance Measure

Performance:

true label prediction

$\text{loss}(Y, \underbrace{f(X)})$ - Measure of closeness between label *Y* and prediction *f*(*X*) for test data X

X	Diagnosis, *Y*	*f*(*X*)	$\text{loss}(Y, f(X))$
	"Anemic cell"	"Anemic cell"	0
		"Healthy cell"	1

$1_A = \begin{cases} 1 & A \\ 0 & A^c \end{cases}$

$$\boxed{\text{loss}(Y, f(X)) = 1_{\{f(X) \neq Y\}} \qquad \textbf{0/1 loss}}$$

Performance Measure

$\min_{\beta} \frac{1}{n} \sum_{i=1}^{n} loss(Y_i, f(X_i))$ — training loss

$f(X_i)$ " $X_i\beta$

n training points $\{X_i, Y_i\}_{i=1}^{n}$

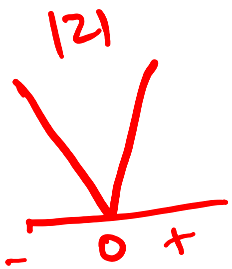
Performance:

$f(X)$ " $X\beta$

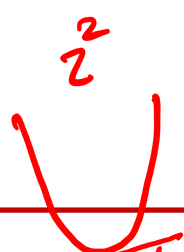
$\text{loss}(Y, f(X))$ - Measure of closeness between label *Y* and prediction *f*(*X*) for test data X

X	Share price, *Y*	*f*(*X*)	$\text{loss}(Y, f(X))$
Past performance, trade volume etc. as of Sept 8, 2010	"\$24.50"	"\$24.50"	0
		"\$26.00"	1?
		"\$26.10"	2?

$|z|$



z^2



$$\boxed{\text{loss}(Y, f(X)) = (f(X) - Y)^2 \quad \textbf{squared loss}}$$

z

Performance Measure

For test data X, measure of closeness between label *Y* and prediction *f*(*X*)

Binary Classification $\text{loss}(Y, f(X)) = 1_{\{f(X) \neq Y\}}$ **0/1 loss**

Regression $\text{loss}(Y, f(X)) = (f(X) - Y)^2$ **squared loss**

Lets think of unsupervised tasks next.

Performance Measure

For test data X, measure how good is the learnt distribution, clustering or embedding $f(X)$

Learning a distribution

- What performance measure would you use?

Poll

- A classifier with 100% accuracy on training data and 70% accuracy on test data is better than a classifier with 80% accuracy on training data and 80% accuracy on test data.

  A. True B. False

- Which classifier is better, given following statistics on test accuracy?

	Mean	Best run	Std
Classifier A	92%	97%	15%
Classifier B	87%	100%	5%

Glossary of Machine Learning

- Task
- Supervised learning
  - Classification
  - Regression
- Unsupervised learning
  - Learning distribution
  - Clustering
  - Dimensionality reduction/Embedding
- Input, X
- Label, Y
- Prediction, f(X)

- Experience = Training data
- Test data
- Overfitting
- Generalization
- Performance
- Likelihood
- Loss – 0/1, squared, negative log likelihood