CS5489
MACHINE LEARNING: ALGORITHMS & APPLICATIONS

LECTURE 1 - INTRO

2023/24 Semester A

Course General Info

- Teaching Team
  - Prof. Antoni B. CHAN
    - abchan@cityu.edu.hk, Office: AC1-G7311
  - TAs:

Name	email	Name	email
Feiyu CHEN	feiyuchen3-c@my.cityu.edu.hk	Weibo SHU	weiboshu2-c@my.cityu.edu.hk
Wei LIN	wlin38-c@my.cityu.edu.hk	Chengtai CAO	chengtcao2-c@my.cityu.edu.hk
Qiangqiang WU	qiangqwu2-c@my.cityu.edu.hk	Hongzong LI	hongzli2-c@my.cityu.edu.hk

  - Office hours: *see Canvas, under Syllabus.*
- Canvas-based course site
  - It is your own responsibility to check Canvas and University e-mail account regularly for announcements and updates.

Class Times

- Lecture
  - **Tuesdays, 13:00-14:50** (LT-18)
- Tutorials
  - Tuesdays, 15:00-15:50 (MMW2450)
  - Tuesdays, 16:00-16:50 (MMW2450)
  - Wednesdays, 11:00-11:50 (MMW2450)
  - Office hours before/after tutorials:
    - Tues 17:00-18:00 in MMW2478
    - Wedn 12:00-13:00 in MMW2450

Teaching Activities

- **Lectures** (2 hours per week)
  - present machine learning algorithms: intuition and idea, and algorithm. Illustrate algorithms on both toy and real-world examples.
  - practice questions each week (*not graded*)
- **Tutorials** (1 hour per week)
  - Use machine learning algorithms on small examples to gain better understanding. Implement algorithms.
- **Assignments** (2)
  - Apply machine learning algorithms to larger datasets, compare and interpret the results of different algorithms.
- **Course Project**
  - Apply machine learning to solve a real-world problem.
    - Kaggle competition OR your own research project.
  - 2 students per group.

Assessment

- **Coursework** (70%)
  - Tutorial exercises (10%) – due 2 weeks after lecture.
  - Assignments (20%) – due Weeks 6 and 10.
  - Midterm (10%) – Week 8
  - Course Project (30%) – due Week 14
    - Project proposal, report, and presentation.
- **Final Exam** (30%)
- **Note:** Must get at least 30% on final exam and 30% on course project to pass the course.

Programming

- **Python**
  - high-level scripting language
  - Jupyter, aka iPython Notebooks (ipynb)
    - interactive computational environment in web browser.
    - can combine text (Markdown) with Python code and output.
  - Libraries
    - numpy – arrays, linear algebra
    - scikit, scikit-learn – scientific computing, machine learning
    - matplotlib, pylab – plotting
    - keras – deep learning
  - Introduction later today

Assignments

- **Assignment/Projects**
  - kaggle.com – a website for data science competitions.
    - Assignments will use Kaggle for evaluation
      - http://inclass.kaggle.com
    - Course projects based on current Kaggle competitions.
      - select among a list of candidates.
  - Code/Report submission
    - Jupyter notebooks (ipynb)
    - Python scripts (for project, if necessary)

Sitting in …

- *Can I just attend the lectures without registering?*
- Reasons to take the course:
  1. The assignments are also teaching activities that give you hands-on experience (both in theory and programming).
     - If you are going to do these, you might as well do it for credit.
  2. Free to do whatever you want for the course project – apply ML to your research problem / ideas.
     - A good course project could lead to a publication.
  3. Machine Learning is an important topic in CS and increasingly important in other fields.
     - It looks good on your transcript/CV.
- Space in lecture hall is limited…
  - Seats are reserved for students who have registered for the course (or are planning to register).

Course Abstract

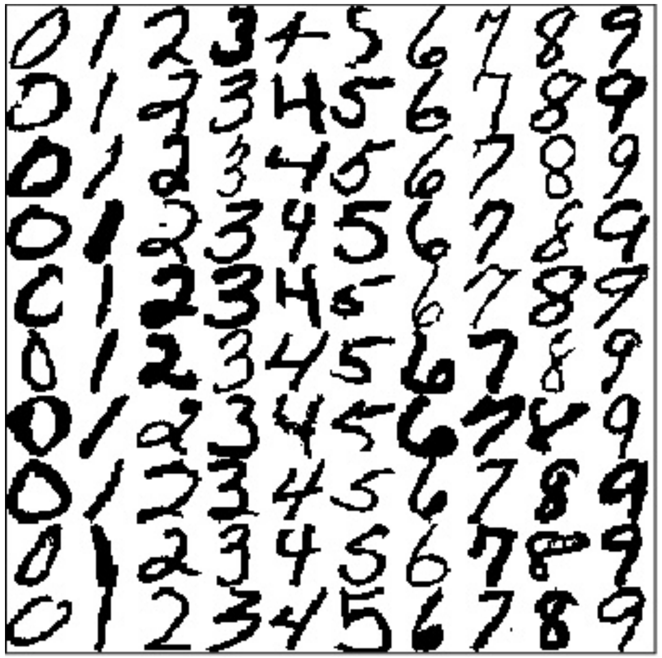
- The goal of this course is to introduce students to the field of machine learning, its algorithms and applications.
  - Machine learning algorithms allow computers to automatically learn to recognize complex patterns from empirical data, such as text and web documents, images, videos, sound, sensor-data, and databases.
  - This course is intended to give a broad overview of machine learning from the practical standpoint, with a focus on implementing and applying machine learning algorithms to real-world problems.
  - At the end of the course, students will have both working knowledge of and practical experience implementing and applying machine learning algorithms on different domains.

CILOs

1. Identify and explain common machine learning algorithms.
2. implement machine learning algorithms
3. Apply machine learning algorithms to solve real-world problems.
4. Evaluate the effectiveness of different machine learning algorithms and discuss their advantages and disadvantages.

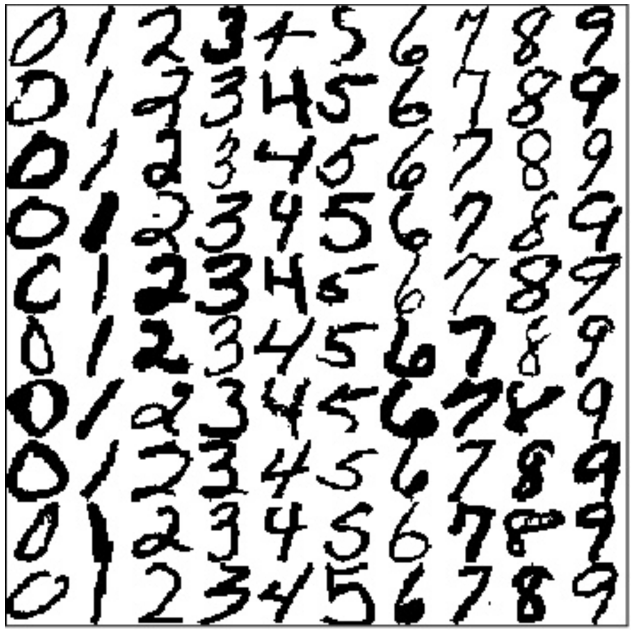
What is Machine Learning?

- Arthur Samuel, 1959
  - *Machine Learning*: field of study that gives computers the ability to learn without being explicitly programmed.
    - e.g. computer learns to play checkers by playing against itself.
- There are many applications that are difficult to program by hand.
  - Example: Recognizing handwritten digits in an image.



What is Machine Learning?

- Example: Recognizing handwritten digits in an image
  - 28x28 image → 784-dim vector
  - a lot of variations & permutations
  - difficult to identify rules & code by hand
- ML solution:
  - gather some example data.
  - train computer to discover differences automatically



What is Machine Learning?

- Tom Mitchell, 1997
  - *Well-posed Learning Problem*: "A computer program is said to learn from experience *E* with respect to some class of tasks *T* and performance measure *P*, if its performance at tasks in *T*, as measured by *P*, improves with experience *E*."
    - e.g., the computer gets better at recognizing digits as it sees more examples, as measured by the error rate.
- A closer look…

Well-posed Learning Problem

- "A computer program is said to learn from experience *E* with respect to some class of tasks *T* and performance measure *P*, if its performance at tasks in *T*, as measured by *P*, improves with experience *E*."

"class of tasks T"

learning is task-specific (recognition, clustering, etc.)

"performance measure P"

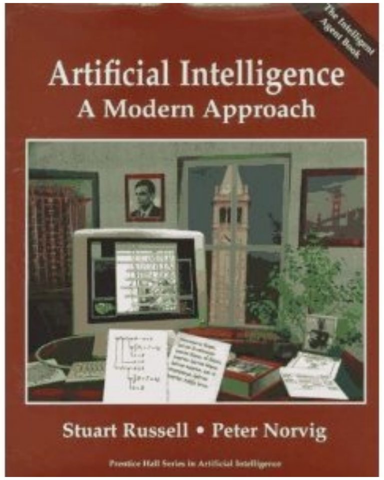
optimize a loss function (e.g., error rate), but also prevent overfitting (regularization). "generalization"

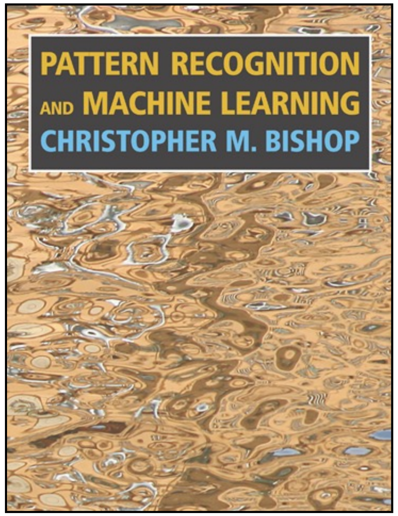
"experience E"

data-driven! More data is better!

Machine Learning vs. Artificial Intelligence

- Machine learning grew out of early work in AI
  - and other fields: statistics, physics, neuroscience, …
  - fueled by more powerful computers and more data.
- "Traditional" Artificial Intelligence (Russell-Norvig)
  - Turing test (is it a computer or a human?)
  - solving by searching (A*, α-β pruning, game playing)
  - knowledge-based (representation, reasoning, logic)
  - planning, scheduling, natural language processing
- Machine Learning (Bishop)
  - probability, statistics, Bayesian formulation
  - statistical learning theory
  - regression, classification, clustering





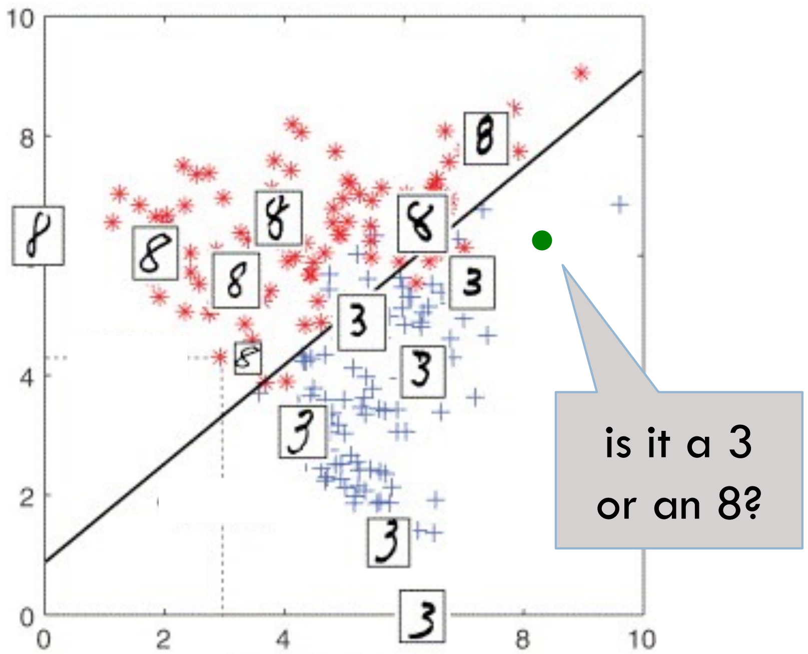
Topics in Machine Learning

- Supervised Learning
- Unsupervised Learning
- Reinforcement Learning
- Learning Theory
- Deep Learning

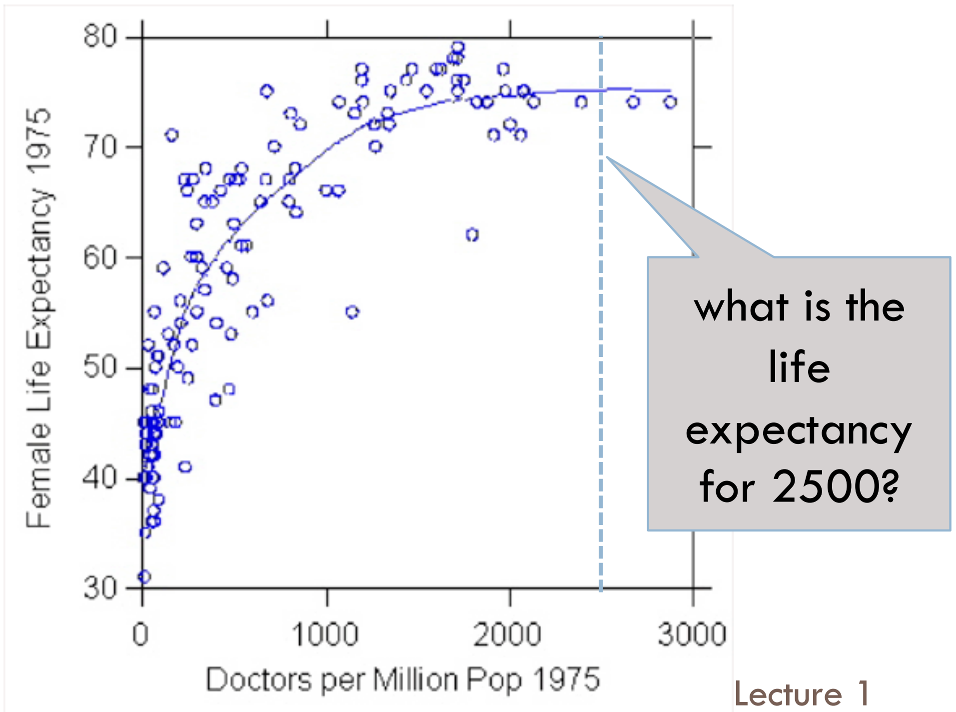
Supervised Learning

- Training data has inputs and outputs
  - e.g., digit recognition (input=image, output=digit)
  - learn a function mapping inputs to outputs

Classification

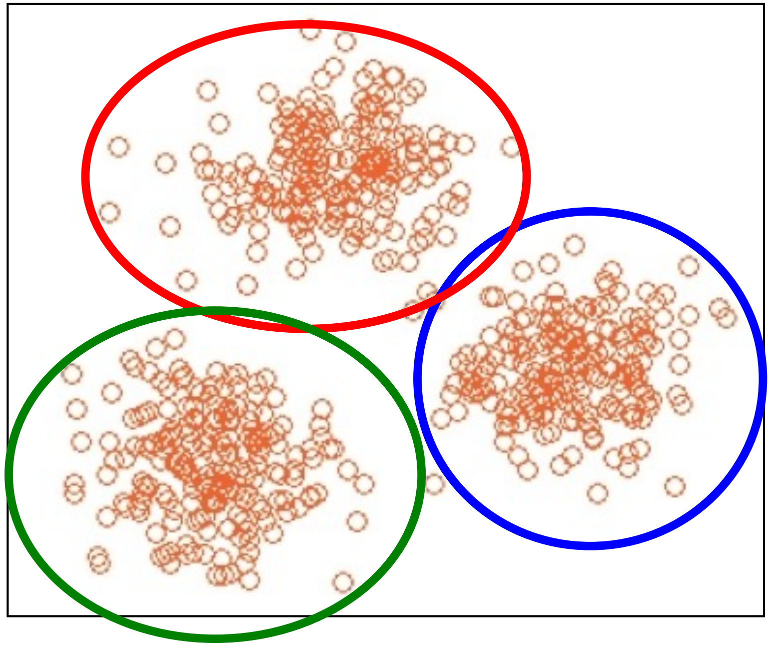


Regression

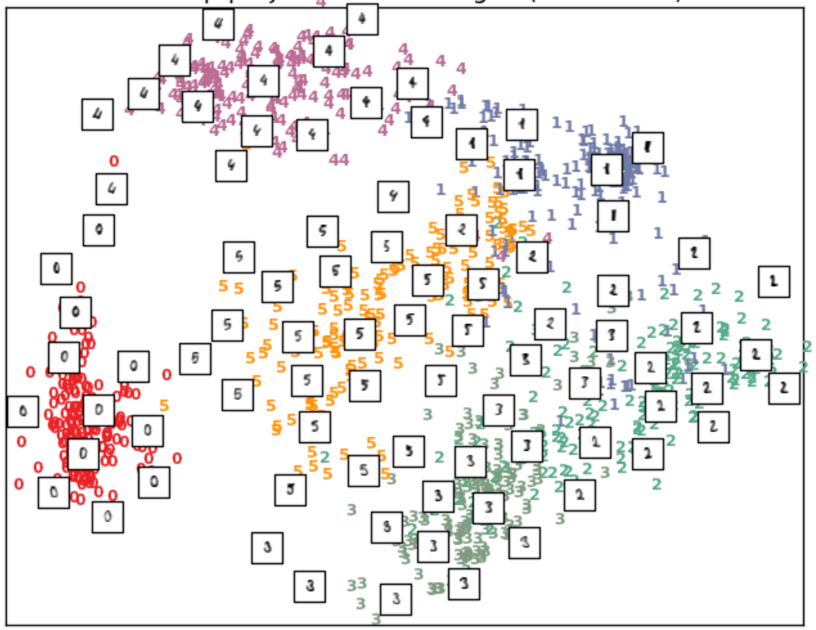


Unsupervised Learning

- Training data only has inputs (no outputs)
  - e.g., collection of web documents
  - clustering - discover groups of similar examples.
  - visualization - project high-dim data to 2 or 3-dimensions.

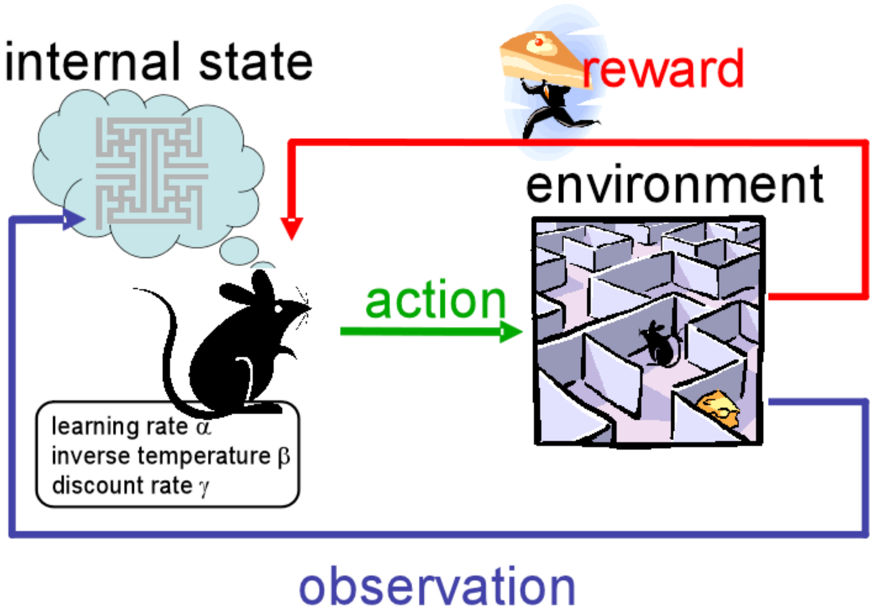


Isomap projection of the digits (time 1.51s)



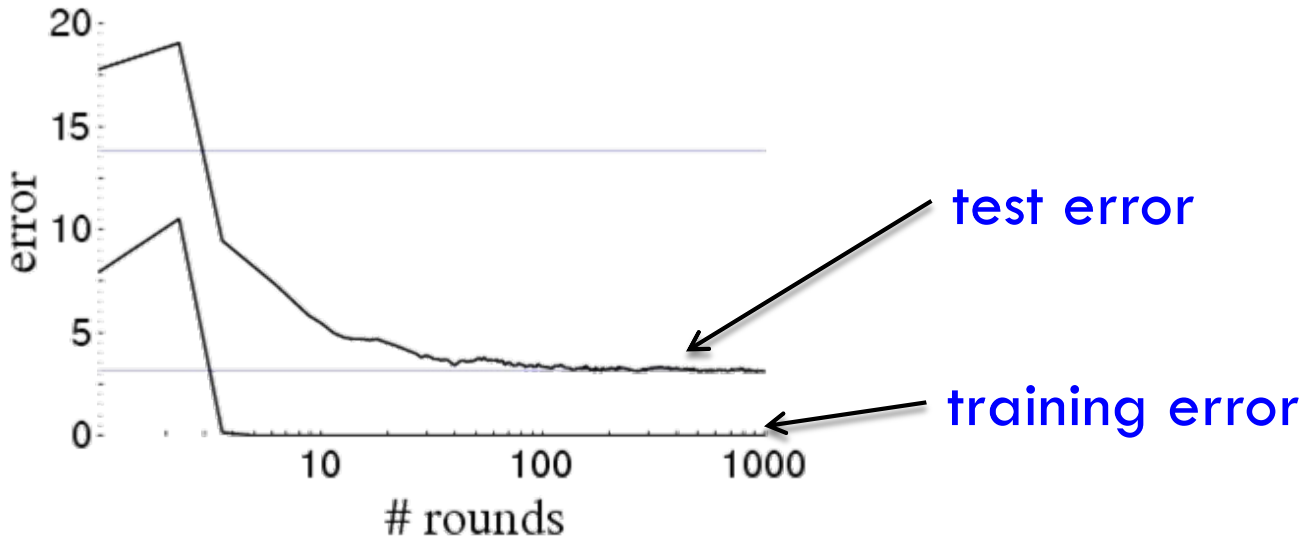
Reinforcement Learning

- Make a sequence of actions, given current states
  - e.g. a robot interacting with its environment
  - Maximize the reward
    - at some point, receive a reward or a punishment.
    - actions may also affect future reward.



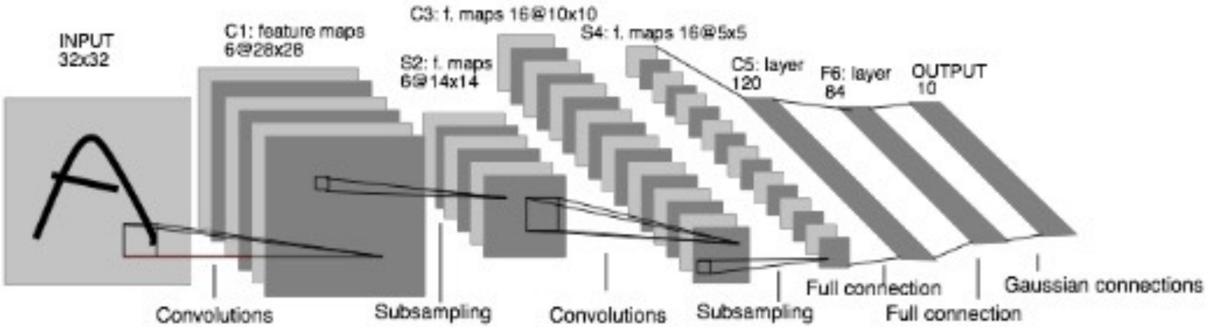
Learning Theory

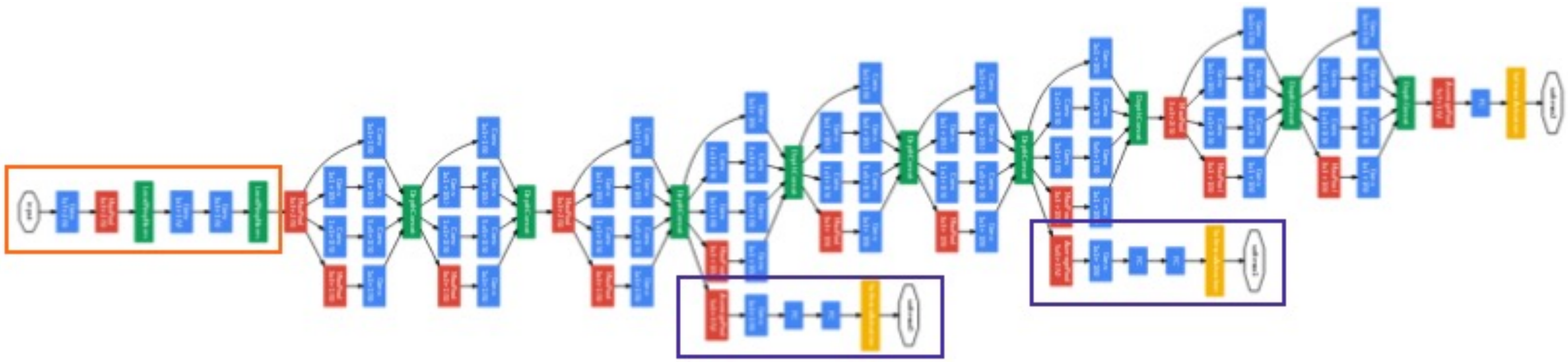
- Why does machine learning work?
  - performance guarantees – bounds on the expected test error.
  - What types of functions can be represent by an algorithm, and how much data do we need?



Deep Learning

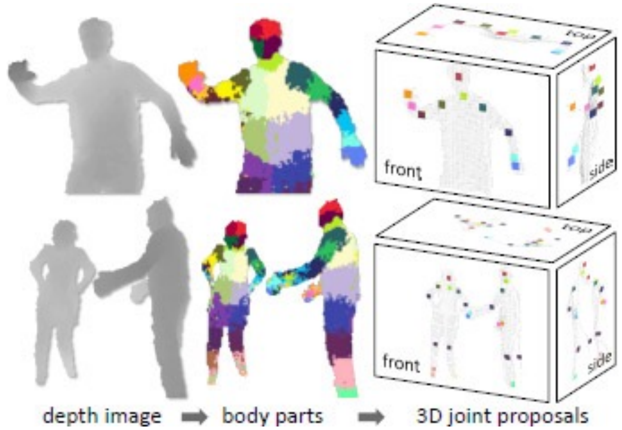
- Deep learning is supervised/unsupervised learning using multi-layer neural networks.
  - Improved training algorithms to prevent overfitting
  - Faster parallelization (GPUs)
  - More data (millions of examples)



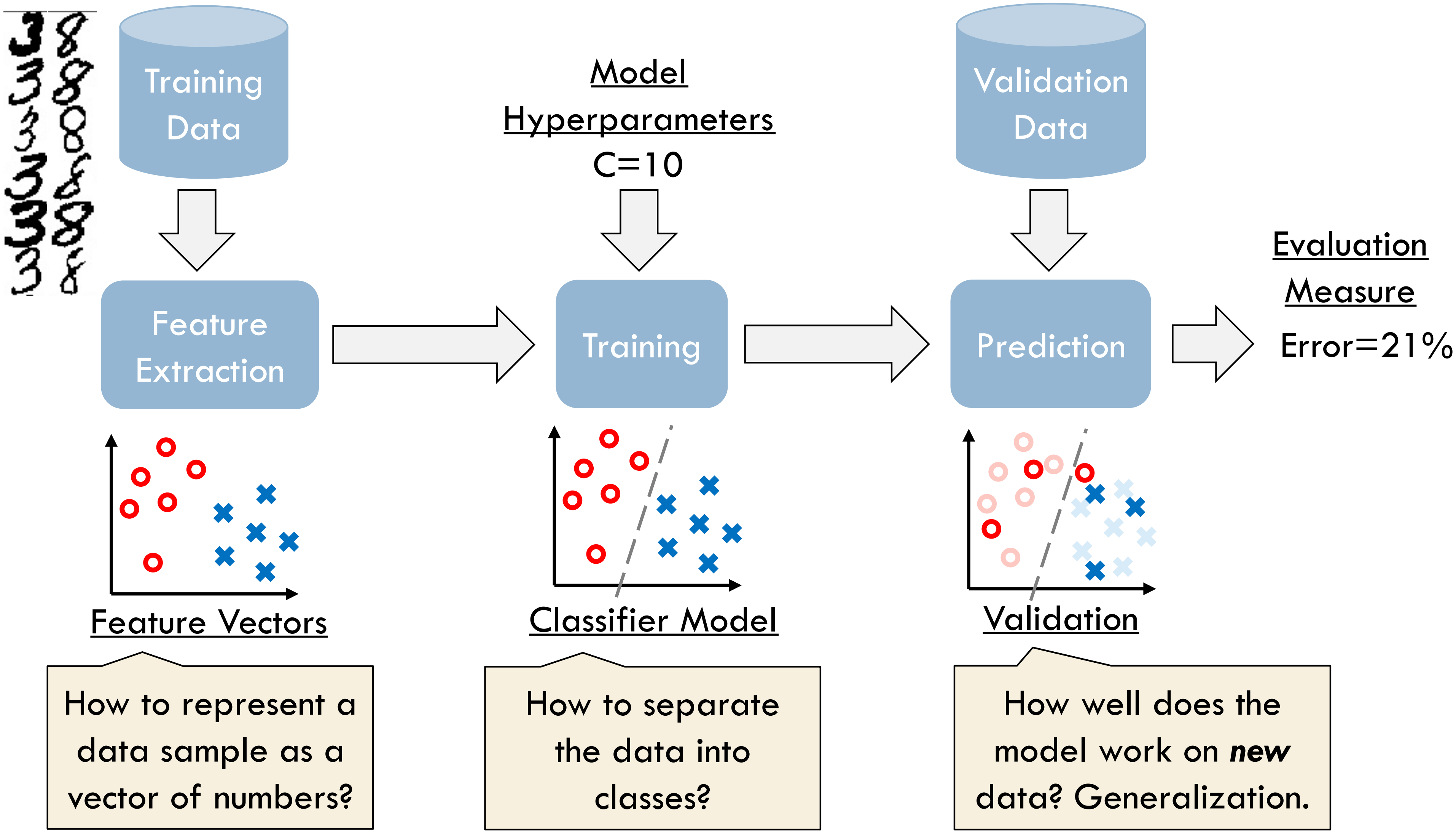


ML in the Real World

- Google
  - spam email classifier, speech recognition, AlphaGo, machine translation, image search, quick links,
- Face detection & recognition
  - digital cameras, Google street view, Facebook
- Business
  - credit card fraud detection
  - stock trading (portfolio optimization)
- Recommendation systems
  - Netflix, Amazon
- Human pose recognition (Kinect)
- Controllers (reinforcement learning)
- Self-driving Cars
- Generative AI (ChatGPT, Dall-E)

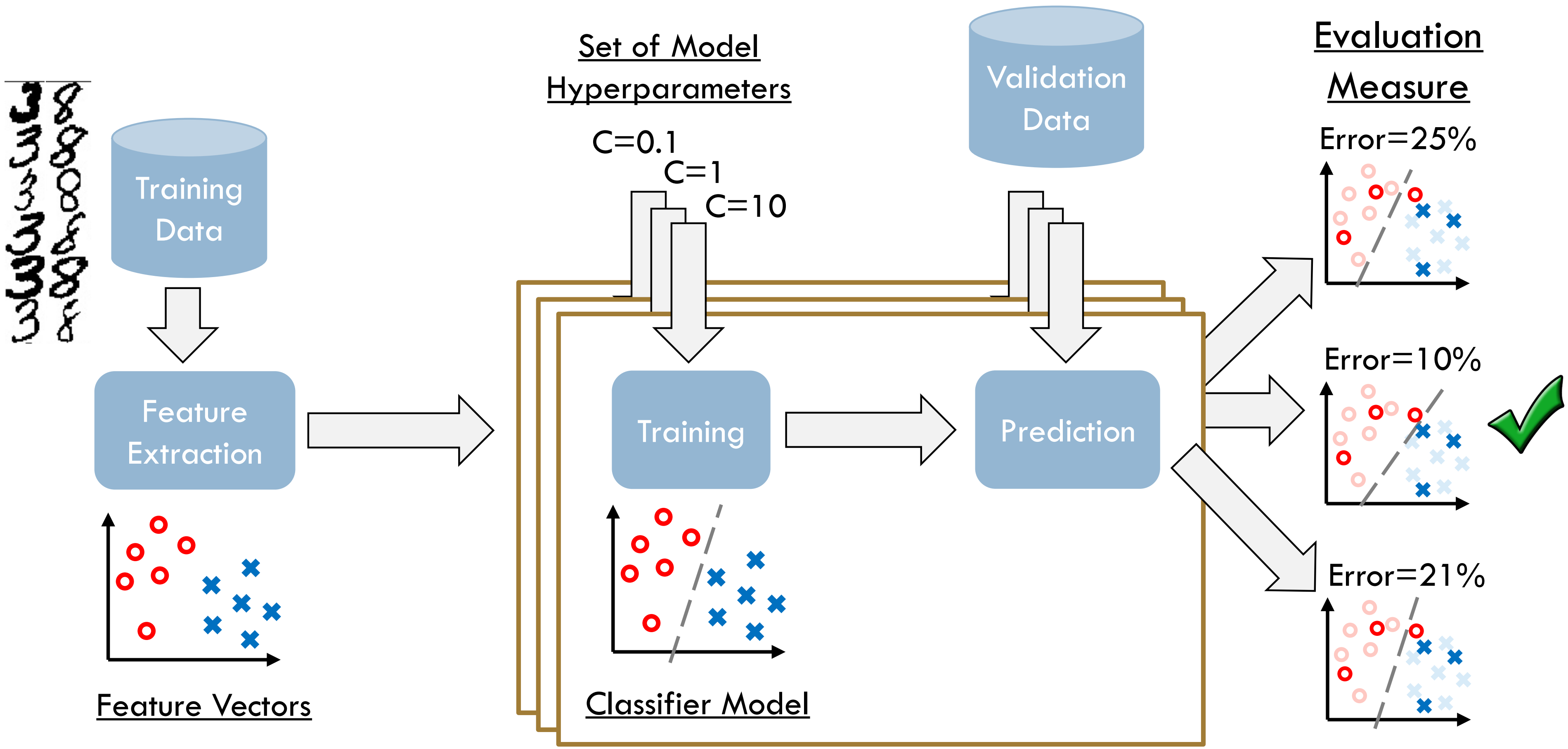


ML Training Pipeline

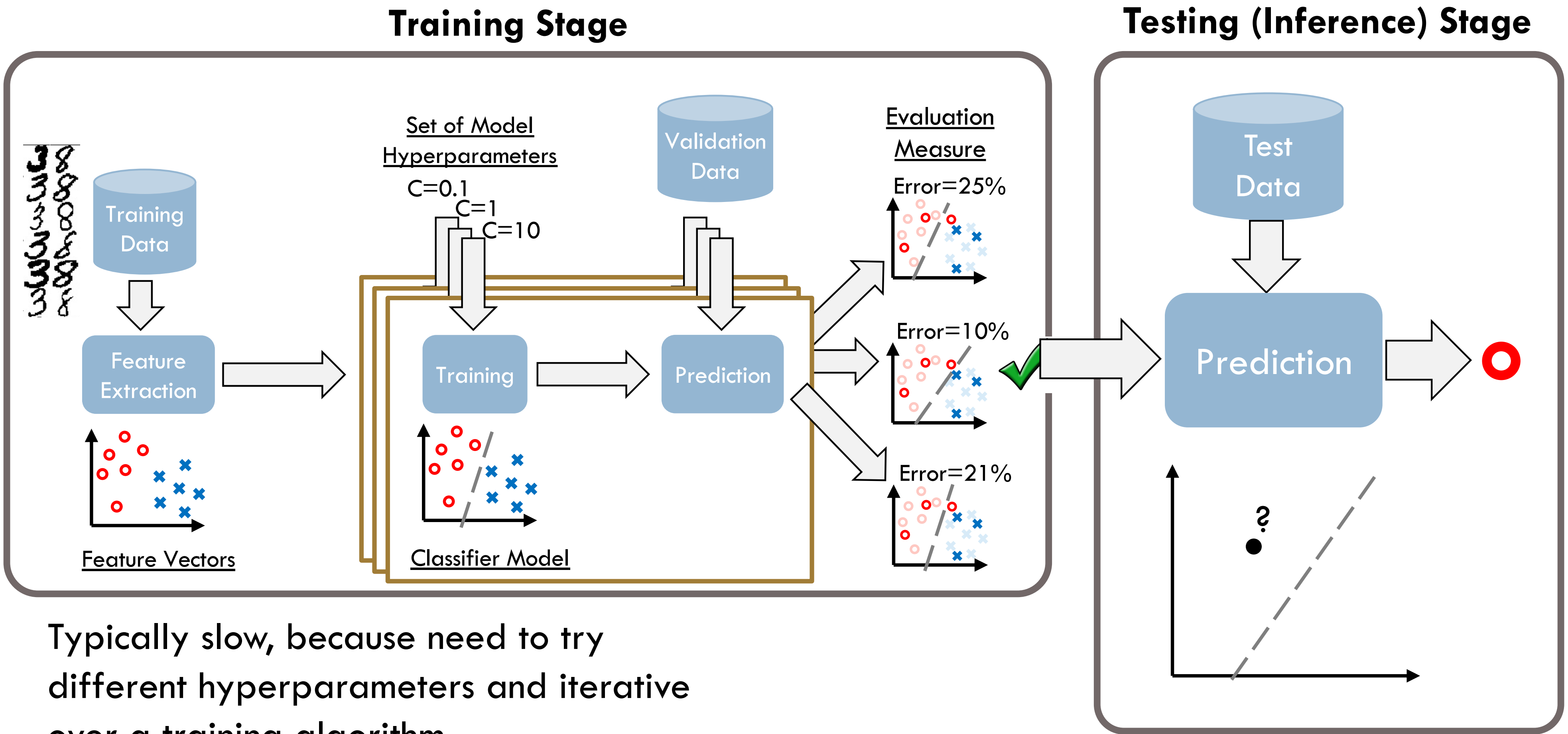


Model Selection Pipeline

- Select hyperparameters that yield low generalization error.



Training & Inference Stages

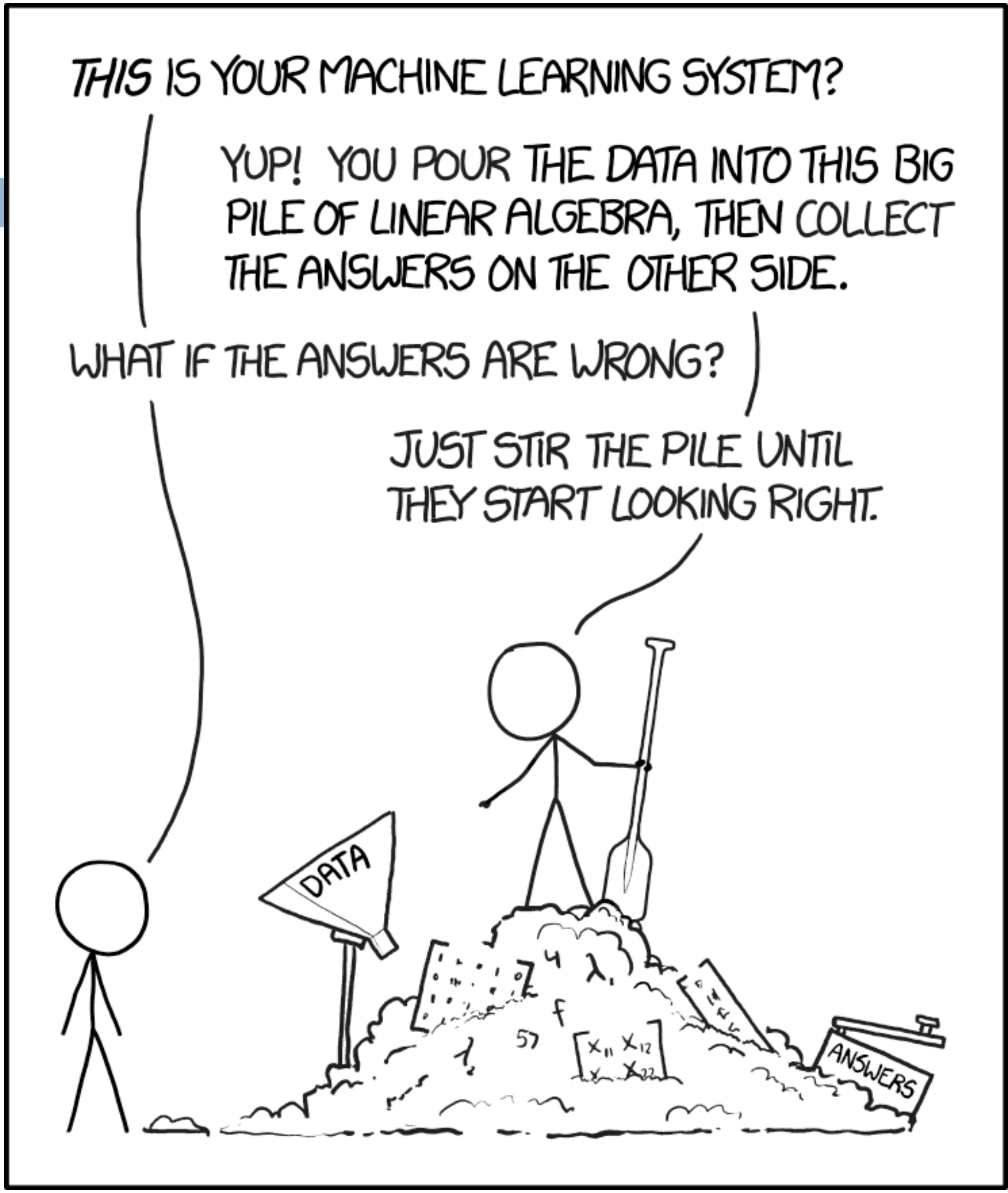


Typically slow, because need to try different hyperparameters and iterative over a training algorithm.

Typically fast, because the model is already estimated.

Building Blocks for ML

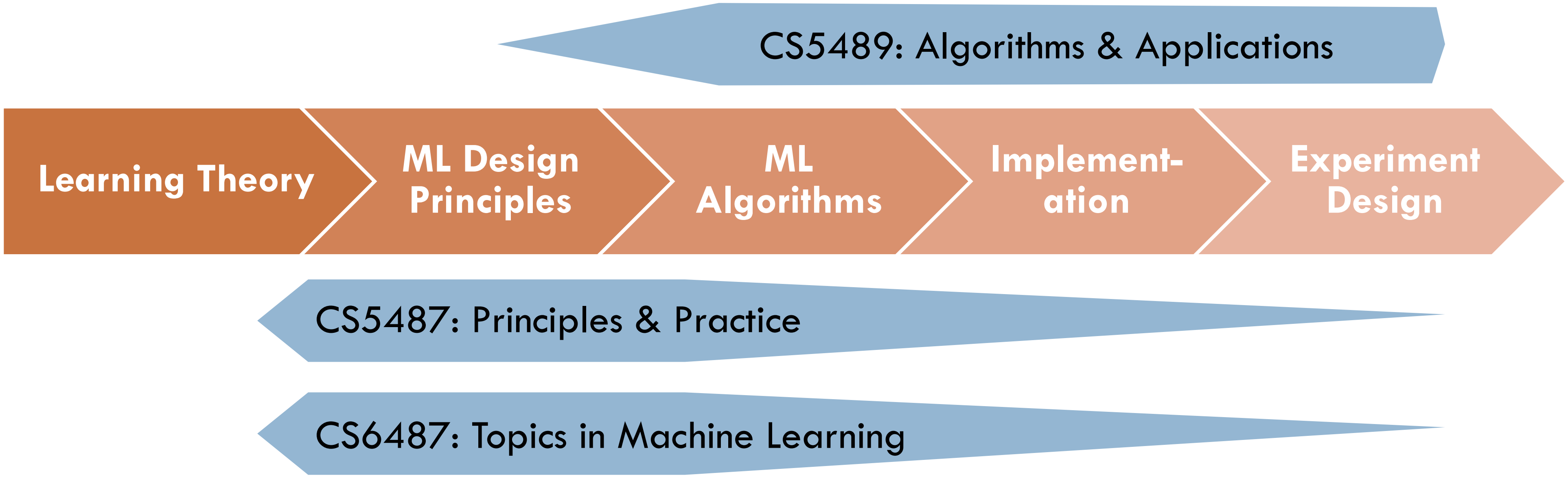
- What are the tools needed for Machine Learning?
  - *linear algebra*
    - matrices, inverse, eigenvector, SVD, …
  - *probability & statistics*
    - random variables, expectation, Bayes' theorem, …
  - *optimization algorithms*
- Don't worry, we will review these as necessary.
  - we need these tools to understand *how* ML works, and implement ML algorithms.
  - many algorithms are already implemented in the ML libraries.



https://xkcd.com/1838/

Relationship with other ML courses

- CS5489 mainly focuses on the intuition of how machine learning algorithms work, implementation of algorithms, applying machine learning and analyzing the results.



- CS5487/CS6487 mainly focus on ML design principles and derivation of algorithms.
- CS5491 (AI) covers knowledge representation, uncertainty reasoning, reinforcement learning, search etc.

CS5489 Schedule (2023/24 A)

Wk	Lecture Topic	References MG (pg)	References H (Ch)	References B (Ch)	References GBC (Ch)	Tutorial	Assessment
1 (5/9)	Lec 1: Introduction / Python					Tut 1	
	Supervised Learning						
2 (12/9)	Lec 2: Probabilistic Models & Bayes Classifiers	70-71	4	4.2		Tut 2	
3 (19/9)	Lec 3: Discriminative Classifiers (LR & SVM)	58-70	5, 6	4.3, 7.1		Tut 3	A1 out
4 (26/9)	Lec 4: Nonlinear Classifiers (KSVM, AdaBoost & RF)	85-106	6, 7	14.2, 14.3		Tut 4	
5 (3/10)	Lec 5: Regression	47-58	8, 9	3.1, 7.1, 6.4		Tut 5	
	Unsupervised Learning						
6 (10/10)	Lec 6: Dimensionality Reduction	142-170	13, 14	12		Tut 6	A1 due, A2 out
7 (17/10)	Lec 7: Clustering	170-183	10.1-10.4	9.1-9.3		Tut 7	
8 (24/10)	**Midterm (Lec 2-5)**					-	
	Deep learning						
9 (31/10)	Lec 8: Neural networks	106-121		5.1-5.5	6	Tut 8	
10 (7/11)	Lec 9: CNNs			5.5	7, 9	Tut 9	A2 due, Pr. out
11 (14/11)	Lec 10: Deep Learning				8	Tut 10	
12 (21/11)	Lec 11: Deep generative models				14, 20.9, 20.19	Tut 11	
13 (28/11)	Lec 12: Graphical models (if time)			8		-	
14 (8/12)	Project Presentations					Pr. Pres.	Pr. Due

References

- Textbooks
  - **MG**: Muller & Guido, "Introduction to Machine Learning with Python", O'Reilly, 2017.
  - **H**: Harrington, "*Machine Learning in Action*", Manning Publications Co., 2012.
  - **B**: C.M. Bishop, "*Pattern Recognition and Machine Learning*", Springer, 2006.
  - **GBC**: Goodfellow, Bengio, Courville, "*Deep Learning*", MIT Press 2016.
    - http://www.deeplearningbook.org
- Papers:
  - Batch Norm: https://arxiv.org/abs/1502.03167
  - SGD: https://arxiv.org/abs/1802.06175
  - ResNet Ensembles: https://arxiv.org/abs/1605.06431

Other References

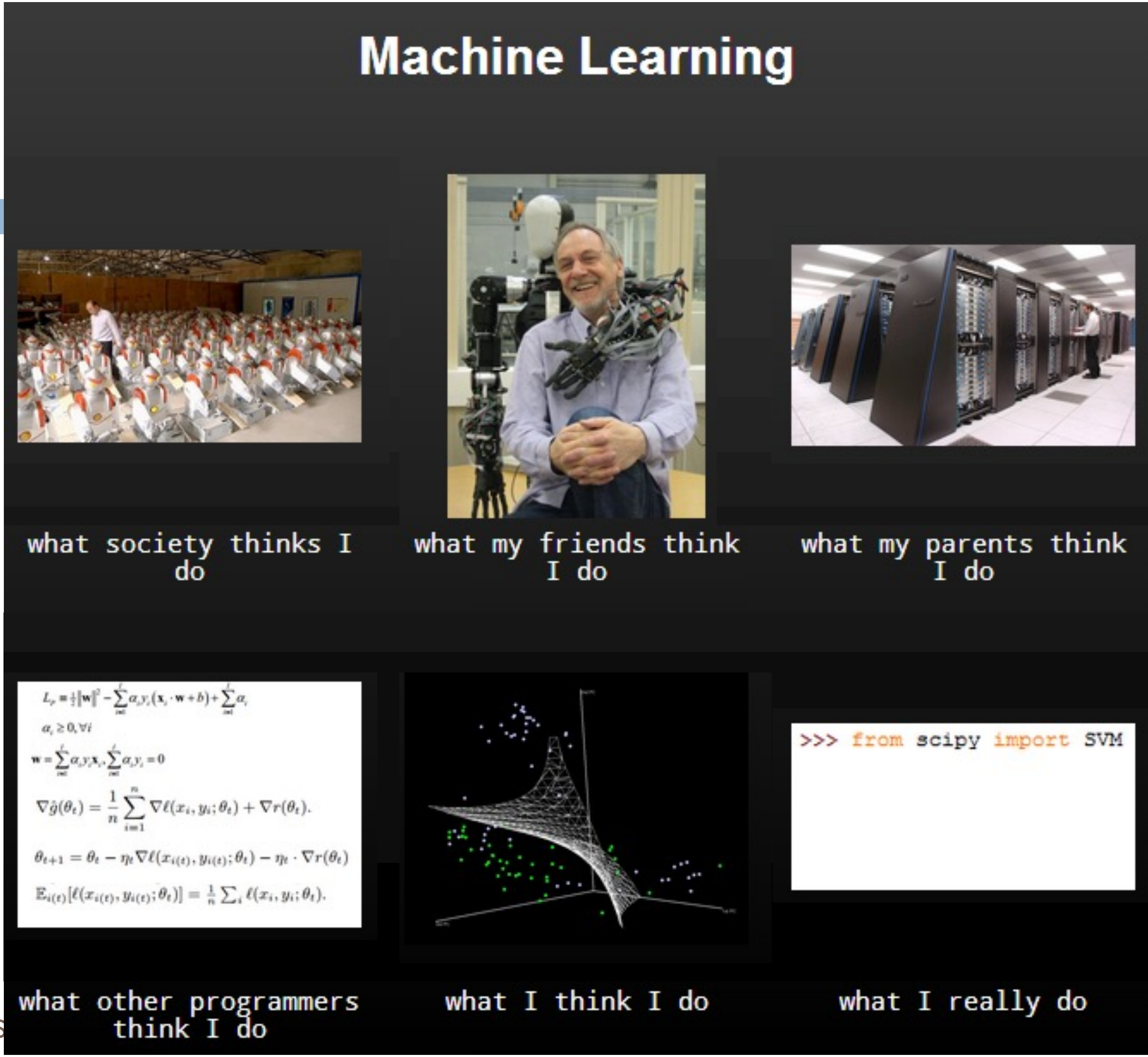
- **Online Reference Books**
  - A. Rajaraman, and J. Ullman, "*Mining of Massive Datasets*", Cambridge University Press, 2011. (http://infolab.stanford.edu/~ullman/mmds.html)
  - H. Daume III, "*A course in Machine Learning*", (http://ciml.info/)
- **Other Books**
  - R.O. Duda, P.E. Hart, & D.G. Stork, "*Pattern Classification (2nd Ed.)*", Wiley-Interscience, 2001.
  - T. Hastie, R. Tibshirani, and J. Friedman, "*The Elements of Statistical Learning: Data Mining, Inference, and Prediction (2nd Ed.)*", Springer-Verlag, 2009.

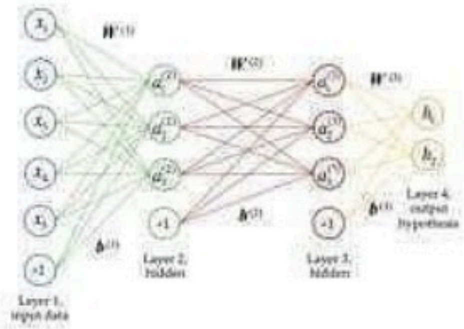
Computing Resources

- **CS Lab JupyterHub**
  - Provides Jupyter notebooks on a central server for multiple users.
  - Select between CPU and GPU resource
  - *dedicated for this course.*
- **CS Lab clusters**
  - High Throughput GPU Clusters (HTGCx)
  - *shared among all CS students / staff.*
- Check Canvas page for more details.

Academic Honesty

- CityU has *Rules of Academic Honesty* and has required all students to complete an online tutorial on subject and declare your understanding
- Plagiarism…
  - It is serious fraud to plagiarize others' work.
  - Punishment ranges from warning to course failure.
- How to prevent plagiarism...
  - Finish the assignments by yourself! You have to write the program/solution yourself.
    - okay to talk about *how* to do the problem with your classmates.
    - **Protect your code**; don't give it away as a "reference" copy.
  - In plagiarism cases, we treat the giver and the copier as both guilty.
  - You hurt your own grades by not reporting cheating.
- As instructors...
  - We have responsibility to report academic dishonesty cases so as not to compromise the quality of education.
  - We take suspected plagiarism cases very seriously.


Machine Learning
what society thinks I do
what my friends think I do
what my parents think I do
what other programmers think I do
what I think I do
>>> from scipy import SVM
what I really do

	A parrot	Machine learning algorithm
		
Learns random phrases	✅	✅
Doesn't understand anything about what it learns	✅	✅
Occasionally speaks nonsense	✅	✅
Is a cute birdie parrot	✅	❌