Course Logistics and Introduction to Machine Learning

Piyush Rai

Introduction to Machine Learning (CS771A)

July 31, 2018

Course Logistics

- **Course name:** Introduction to Machine Learning or "ML" (CS771A)
- **Timing and Venue:** Tue/Thur 6:00-7:30pm, L-16

Course Logistics

- **Course name:** Introduction to Machine Learning or "ML" (CS771A)
- **Timing and Venue:** Tue/Thur 6:00-7:30pm, L-16
- **Course website:** `https://tinyurl.com/cs771-a18w` (slides/readings etc will be posted here)

Course Logistics

- **Course name:** Introduction to Machine Learning or "ML" (CS771A)
- **Timing and Venue:** Tue/Thur 6:00-7:30pm, L-16
- **Course website:** `https://tinyurl.com/cs771-a18w` (slides/readings etc will be posted here)
- **Piazza discussion site:** `https://tinyurl.com/cs771-a18p` (use it actively and responsibly)

Course Logistics

- **Course name:** Introduction to Machine Learning or “ML” (CS771A)
- **Timing and Venue:** Tue/Thur 6:00-7:30pm, L-16
- **Course website:** `https://tinyurl.com/cs771-a18w` (slides/readings etc will be posted here)
- **Piazza discussion site:** `https://tinyurl.com/cs771-a18p` (use it actively and responsibly)
- **Gradescope (for assignment submission):** `https://tinyurl.com/cs771-a18g`

Course Logistics

- **Course name:** Introduction to Machine Learning or "ML" (CS771A)
- **Timing and Venue:** Tue/Thur 6:00-7:30pm, L-16
- **Course website:** `https://tinyurl.com/cs771-a18w` (slides/readings etc will be posted here)
- **Piazza discussion site:** `https://tinyurl.com/cs771-a18p` (use it actively and responsibly)
- **Gradescope (for assignment submission):** `https://tinyurl.com/cs771-a18g`
- Course-related announcements will be sent on the **class mailing list** (and also on Piazza)

Course Logistics

- **Course name:** Introduction to Machine Learning or "ML" (CS771A)
- **Timing and Venue:** Tue/Thur 6:00-7:30pm, L-16
- **Course website:** `https://tinyurl.com/cs771-a18w` (slides/readings etc will be posted here)
- **Piazza discussion site:** `https://tinyurl.com/cs771-a18p` (use it actively and responsibly)
- **Gradescope (for assignment submission):** `https://tinyurl.com/cs771-a18g`
- Course-related announcements will be sent on the **class mailing list** (and also on Piazza)
- **Instructor:** Piyush Rai (e-mail: piyush@cse.iitk.ac.in, office: RM-502)

Course Logistics

- **Course name:** Introduction to Machine Learning or "ML" (CS771A)
- **Timing and Venue:** Tue/Thur 6:00-7:30pm, L-16
- **Course website:** `https://tinyurl.com/cs771-a18w` (slides/readings etc will be posted here)
- **Piazza discussion site:** `https://tinyurl.com/cs771-a18p` (use it actively and responsibly)
- **Gradescope (for assignment submission):** `https://tinyurl.com/cs771-a18g`
- Course-related announcements will be sent on the **class mailing list** (and also on Piazza)
- **Instructor:** Piyush Rai (e-mail: piyush@cse.iitk.ac.in, office: RM-502)
  - Prefix email subject by CS771A (better alternative: Piazza private message to instructor)

Course Logistics

- **Course name:** Introduction to Machine Learning or “ML” (CS771A)
- **Timing and Venue:** Tue/Thur 6:00-7:30pm, L-16
- **Course website:** `https://tinyurl.com/cs771-a18w` (slides/readings etc will be posted here)
- **Piazza discussion site:** `https://tinyurl.com/cs771-a18p` (use it actively and responsibly)
- **Gradescope (for assignment submission):** `https://tinyurl.com/cs771-a18g`
- Course-related announcements will be sent on the **class mailing list** (and also on Piazza)
- **Instructor:** Piyush Rai (e-mail: piyush@cse.iitk.ac.in, office: RM-502)
  - Prefix email subject by CS771A (better alternative: Piazza private message to instructor)
  - Office Hours: Wed 6:00-7:30pm (by appointment)

Course Logistics

- **Course name:** Introduction to Machine Learning or "ML" (CS771A)
- **Timing and Venue:** Tue/Thur 6:00-7:30pm, L-16
- **Course website:** `https://tinyurl.com/cs771-a18w` (slides/readings etc will be posted here)
- **Piazza discussion site:** `https://tinyurl.com/cs771-a18p` (use it actively and responsibly)
- **Gradescope (for assignment submission):** `https://tinyurl.com/cs771-a18g`
- Course-related announcements will be sent on the **class mailing list** (and also on Piazza)
- **Instructor:** Piyush Rai (e-mail: piyush@cse.iitk.ac.in, office: RM-502)
  - Prefix email subject by CS771A (better alternative: Piazza private message to instructor)
  - Office Hours: Wed 6:00-7:30pm (by appointment)
- **Auditing:** Don't need formal permission from me. Send me email to be added to the mailing list.

Course Logistics

- **Course name:** Introduction to Machine Learning or "ML" (CS771A)
- **Timing and Venue:** Tue/Thur 6:00-7:30pm, L-16
- **Course website:** `https://tinyurl.com/cs771-a18w` (slides/readings etc will be posted here)
- **Piazza discussion site:** `https://tinyurl.com/cs771-a18p` (use it actively and responsibly)
- **Gradescope (for assignment submission):** `https://tinyurl.com/cs771-a18g`
- Course-related announcements will be sent on the **class mailing list** (and also on Piazza)
- **Instructor:** Piyush Rai (e-mail: piyush@cse.iitk.ac.in, office: RM-502)
  - Prefix email subject by CS771A (better alternative: Piazza private message to instructor)
  - Office Hours: Wed 6:00-7:30pm (by appointment)
- **Auditing:** Don't need formal permission from me. Send me email to be added to the mailing list.
  - Will have access to all the course material; can participate in Piazza discussions

Course Logistics

- **Course name:** Introduction to Machine Learning or "ML" (CS771A)
- **Timing and Venue:** Tue/Thur 6:00-7:30pm, L-16
- **Course website:** `https://tinyurl.com/cs771-a18w` (slides/readings etc will be posted here)
- **Piazza discussion site:** `https://tinyurl.com/cs771-a18p` (use it actively and responsibly)
- **Gradescope (for assignment submission):** `https://tinyurl.com/cs771-a18g`
- Course-related announcements will be sent on the **class mailing list** (and also on Piazza)
- **Instructor:** Piyush Rai (e-mail: piyush@cse.iitk.ac.in, office: RM-502)
  - Prefix email subject by CS771A (better alternative: Piazza private message to instructor)
  - Office Hours: Wed 6:00-7:30pm (by appointment)
- **Auditing:** Don't need formal permission from me. Send me email to be added to the mailing list.
  - Will have access to all the course material; can participate in Piazza discussions
  - However, we are unable to grade your assignments/exams. Can't form project groups with creditors.

The TA Team


Shivam Bansal


Dhanajit Brahma


Sunabha Chatterjee


Prerit Garg


Gopichand Kotana


Neeraj Kumar


Pawan Kumar


Kranti Parida


Kawal Preet


Prem Raj

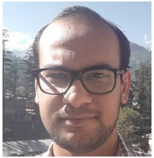

Utsav Singh


Samik Some


Vinay Verma

Project Mentors


Homanga Bhardwaj


Aadil Hayat


Ankit Jalan


Varun Khare


Sarthak Mittal


Gurpreet Singh

.. and some more..

Assignments, Exams, and Grading Policy

- Homework (4-5): 30%, Midsem Exam: 20%, Endsem Exam: 30%, Term Project: 20%

Assignments, Exams, and Grading Policy

- Homework (4-5): 30%, Midsem Exam: 20%, Endsem Exam: 30%, Term Project: 20%
- Homeworks will usually be a mix of
  - **Pen-paper based questions**:
  - **Programming questions:**

Assignments, Exams, and Grading Policy

- Homework (4-5): 30%, Midsem Exam: 20%, Endsem Exam: 30%, Term Project: 20%
- Homeworks will usually be a mix of
  - **Pen-paper based questions**: Derivations/analysis/improvements of ML algos studied in class, designing "new" ML algos for some problem scenarios (using techniques studied in class)
  - **Programming questions:**

Assignments, Exams, and Grading Policy

- Homework (4-5): 30%, Midsem Exam: 20%, Endsem Exam: 30%, Term Project: 20%
- Homeworks will usually be a mix of
  - **Pen-paper based questions**: Derivations/analysis/improvements of ML algos studied in class, designing "new" ML algos for some problem scenarios (using techniques studied in class)
  - **Programming questions:** Implement ML algos from scratch or using existing software tools, improve existing ML algos, apply ML algos to analyze data. Can use Python/MATLAB

Assignments, Exams, and Grading Policy

- Homework (4-5): 30%, Midsem Exam: 20%, Endsem Exam: 30%, Term Project: 20%
- Homeworks will usually be a mix of
  - **Pen-paper based questions**: Derivations/analysis/improvements of ML algos studied in class, designing "new" ML algos for some problem scenarios (using techniques studied in class)
  - **Programming questions:** Implement ML algos from scratch or using existing software tools, improve existing ML algos, apply ML algos to analyze data. Can use Python/MATLAB
- Homework solutions must be prepared using LaTeX. We will provide a LaTeX template.

Assignments, Exams, and Grading Policy

- Homework (4-5): 30%, Midsem Exam: 20%, Endsem Exam: 30%, Term Project: 20%
- Homeworks will usually be a mix of
  - **Pen-paper based questions**: Derivations/analysis/improvements of ML algos studied in class, designing "new" ML algos for some problem scenarios (using techniques studied in class)
  - **Programming questions:** Implement ML algos from scratch or using existing software tools, improve existing ML algos, apply ML algos to analyze data. Can use Python/MATLAB
- Homework solutions must be prepared using LaTeX. We will provide a LaTeX template.
  - Many resources to learn LaTeX available online. Must have skill. Learn it now!

Assignments, Exams, and Grading Policy

- Homework (4-5): 30%, Midsem Exam: 20%, Endsem Exam: 30%, Term Project: 20%
- Homeworks will usually be a mix of
  - **Pen-paper based questions**: Derivations/analysis/improvements of ML algos studied in class, designing "new" ML algos for some problem scenarios (using techniques studied in class)
  - **Programming questions:** Implement ML algos from scratch or using existing software tools, improve existing ML algos, apply ML algos to analyze data. Can use Python/MATLAB
- Homework solutions must be prepared using LaTeX. We will provide a LaTeX template.
  - Many resources to learn LaTeX available online. Must have skill. Learn it now!
- Homeworks must be submitted via Gradescope (no email submissions)

Assignments, Exams, and Grading Policy

- Homework (4-5): 30%, Midsem Exam: 20%, Endsem Exam: 30%, Term Project: 20%
- Homeworks will usually be a mix of
  - **Pen-paper based questions**: Derivations/analysis/improvements of ML algos studied in class, designing "new" ML algos for some problem scenarios (using techniques studied in class)
  - **Programming questions:** Implement ML algos from scratch or using existing software tools, improve existing ML algos, apply ML algos to analyze data. Can use Python/MATLAB
- Homework solutions must be prepared using LaTeX. We will provide a LaTeX template.
  - Many resources to learn LaTeX available online. Must have skill. Learn it now!
- Homeworks must be submitted via Gradescope (no email submissions)
- Late homeworks will receive 10%/20%/30% penalty for 24/48/72 hour late submission, resp.

Assignments, Exams, and Grading Policy

- Homework (4-5): 30%, Midsem Exam: 20%, Endsem Exam: 30%, Term Project: 20%
- Homeworks will usually be a mix of
  - **Pen-paper based questions**: Derivations/analysis/improvements of ML algos studied in class, designing "new" ML algos for some problem scenarios (using techniques studied in class)
  - **Programming questions:** Implement ML algos from scratch or using existing software tools, improve existing ML algos, apply ML algos to analyze data. Can use Python/MATLAB
- Homework solutions must be prepared using LaTeX. We will provide a LaTeX template.
  - Many resources to learn LaTeX available online. Must have skill. Learn it now!
- Homeworks must be submitted via Gradescope (no email submissions)
- Late homeworks will receive 10%/20%/30% penalty for 24/48/72 hour late submission, resp.
- Exams will be closed-book (an A4-sized cheat-sheet allowed)

Term Project (Worth 20%)

- To be done in groups of 5-6 students. Must form the groups by August 25 and inform us

Term Project (Worth 20%)

- To be done in groups of 5-6 students. Must form the groups by August 25 and inform us
- Avoid smaller/larger groups (unless there is a very strong reason for that)

Term Project (Worth 20%)

- To be done in groups of 5-6 students. Must form the groups by August 25 and inform us
- Avoid smaller/larger groups (unless there is a very strong reason for that)
- We will float a list containing several project ideas that you can choose from

Term Project (Worth 20%)

- To be done in groups of 5-6 students. Must form the groups by August 25 and inform us
- Avoid smaller/larger groups (unless there is a very strong reason for that)
- We will float a list containing several project ideas that you can choose from
- You may also propose your own project idea
  - .. but please discuss the scope/feasibility with me and/or project mentors

Term Project (Worth 20%)

- To be done in groups of 5-6 students. Must form the groups by August 25 and inform us
- Avoid smaller/larger groups (unless there is a very strong reason for that)
- We will float a list containing several project ideas that you can choose from
- You may also propose your own project idea
    - .. but please discuss the scope/feasibility with me and/or project mentors
- Many types of projects possible (theoretical/applied/mixed); even building a cool/useful app/portal for something using ML as its backend would be a nice project. Explore around for ideas.

Term Project (Worth 20%)

- To be done in groups of 5-6 students. Must form the groups by August 25 and inform us
- Avoid smaller/larger groups (unless there is a very strong reason for that)
- We will float a list containing several project ideas that you can choose from
- You may also propose your own project idea
  - .. but please discuss the scope/feasibility with me and/or project mentors
- Many types of projects possible (theoretical/applied/mixed); even building a cool/useful app/portal for something using ML as its backend would be a nice project. Explore around for ideas.
- Important: Don't (re)use a project from another course you've done before (or doing currently)

Term Project (Worth 20%)

- To be done in groups of 5-6 students. Must form the groups by August 25 and inform us
- Avoid smaller/larger groups (unless there is a very strong reason for that)
- We will float a list containing several project ideas that you can choose from
- You may also propose your own project idea
  - .. but please discuss the scope/feasibility with me and/or project mentors
- Many types of projects possible (theoretical/applied/mixed); even building a cool/useful app/portal for something using ML as its backend would be a nice project. Explore around for ideas.
- Important: Don't (re)use a project from another course you've done before (or doing currently)
- Need to submit a formal project proposal by Sept 7 (not graded but mandatory)

Term Project (Worth 20%)

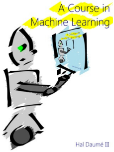
- To be done in groups of 5-6 students. Must form the groups by August 25 and inform us
- Avoid smaller/larger groups (unless there is a very strong reason for that)
- We will float a list containing several project ideas that you can choose from
- You may also propose your own project idea
  - .. but please discuss the scope/feasibility with me and/or project mentors
- Many types of projects possible (theoretical/applied/mixed); even building a cool/useful app/portal for something using ML as its backend would be a nice project. Explore around for ideas.
- Important: Don't (re)use a project from another course you've done before (or doing currently)
- Need to submit a formal project proposal by Sept 7 (not graded but mandatory)
  - Should contain a problem description, tentative plan of action, etc (I'll provide more guidelines later)

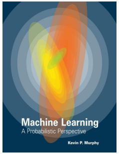
Term Project (Worth 20%)

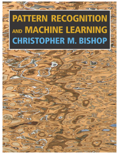
- To be done in groups of 5-6 students. Must form the groups by August 25 and inform us
- Avoid smaller/larger groups (unless there is a very strong reason for that)
- We will float a list containing several project ideas that you can choose from
- You may also propose your own project idea
  - .. but please discuss the scope/feasibility with me and/or project mentors
- Many types of projects possible (theoretical/applied/mixed); even building a cool/useful app/portal for something using ML as its backend would be a nice project. Explore around for ideas.
- Important: Don't (re)use a project from another course you've done before (or doing currently)
- Need to submit a formal project proposal by Sept 7 (not graded but mandatory)
  - Should contain a problem description, tentative plan of action, etc (I'll provide more guidelines later)
- Please don't wait until Sept 7 to finalize the project idea

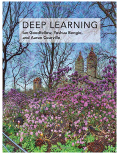
Textbook and References

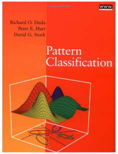
- Many excellent texts but none “required”. Some of them include (list not exhaustive)

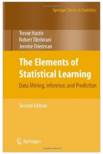


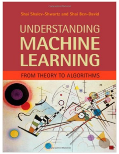


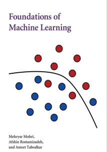






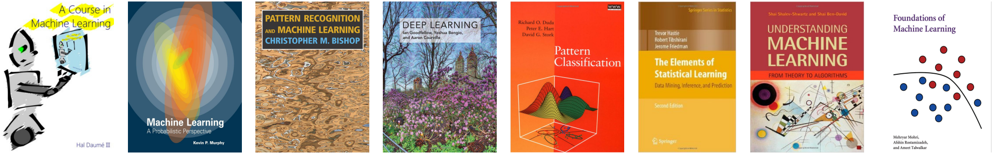






Textbook and References

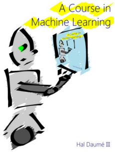
- Many excellent texts but none "required". Some of them include (list not exhaustive)

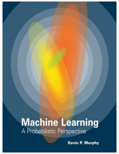


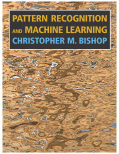
- Different books might vary in terms of
  - Set of topics covered
  - General approach taken e.g., classical statistics, deep learning, probabilistic/Bayesian, theory

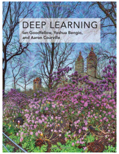
Textbook and References

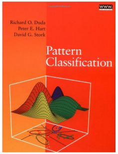
- Many excellent texts but none "required". Some of them include (list not exhaustive)

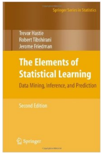


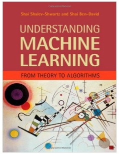


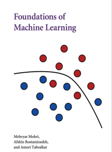








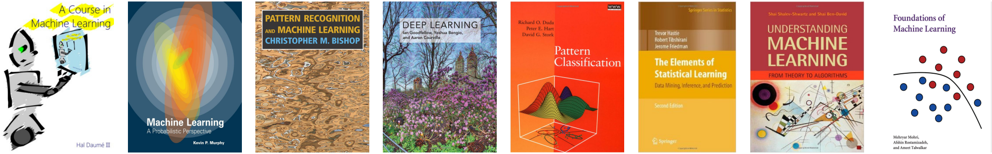




- Different books might vary in terms of
  - Set of topics covered
  - General approach taken e.g., classical statistics, deep learning, probabilistic/Bayesian, theory
  - Terminology and notation (beware of this especially)

Textbook and References

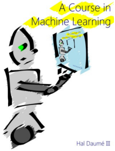
- Many excellent texts but none “required”. Some of them include (list not exhaustive)

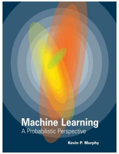


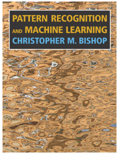
- Different books might vary in terms of
  - Set of topics covered
  - General approach taken e.g., classical statistics, deep learning, probabilistic/Bayesian, theory
  - Terminology and notation (beware of this especially)
- Avoid using too many sources until you have developed a reasonable understanding of a concept

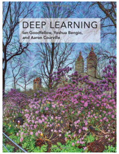
Textbook and References

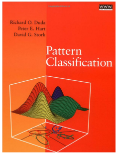
- Many excellent texts but none “required”. Some of them include (list not exhaustive)

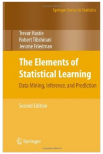


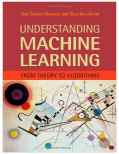


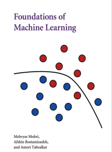












- Different books might vary in terms of
  - Set of topics covered
  - General approach taken e.g., classical statistics, deep learning, probabilistic/Bayesian, theory
  - Terminology and notation (beware of this especially)
- Avoid using too many sources until you have developed a reasonable understanding of a concept
- We will provide you the reading material from the relevant sources

Collaboration vs Cheating

- Collaboration is encouraged. Cheating/copying will lead to strict punishments.

Collaboration vs Cheating

- Collaboration is encouraged. Cheating/copying will lead to strict punishments.
- Feel free to discuss homework assignments with your classmates.

Collaboration vs Cheating

- Collaboration is encouraged. Cheating/copying will lead to strict punishments.
- Feel free to discuss homework assignments with your classmates.
- However, your own solution must be in your own words (same goes for coding assignments)

Collaboration vs Cheating

- Collaboration is encouraged. Cheating/copying will lead to strict punishments.
- Feel free to discuss homework assignments with your classmates.
- However, your own solution must be in your own words (same goes for coding assignments)
- Plagiarism from other sources (for assignments/project) will also lead to strict punishment unless you duly credit the original source(s)

Collaboration vs Cheating

- Collaboration is encouraged. Cheating/copying will lead to strict punishments.
- Feel free to discuss homework assignments with your classmates.
- However, your own solution must be in your own words (same goes for coding assignments)
- Plagiarism from other sources (for assignments/project) will also lead to strict punishment unless you duly credit the original source(s)
- Other things that will lead to punishment
  - Use of unfair means in the exams
  - Fabricating experimental results in assignments/project

Collaboration vs Cheating

- Collaboration is encouraged. Cheating/copying will lead to strict punishments.
- Feel free to discuss homework assignments with your classmates.
- However, your own solution must be in your own words (same goes for coding assignments)
- Plagiarism from other sources (for assignments/project) will also lead to strict punishment unless you duly credit the original source(s)
- Other things that will lead to punishment
  - Use of unfair means in the exams
  - Fabricating experimental results in assignments/project
- Important: Both copying as well as helping someone copy will be equally punishable

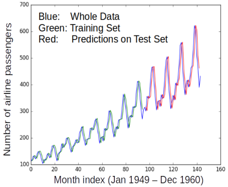
Intro to Machine Learning

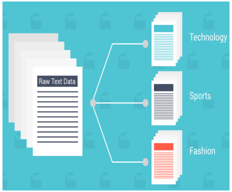
Machine Learning (ML)

- Designing algorithms that ingest data and learn a (hypothesized) model of the data

Machine Learning (ML)

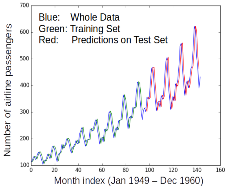
- Designing algorithms that ingest data and learn a (hypothesized) model of the data
- The learned model can be used to
  - Detect patterns/structures/themes/trends etc. in the data
  - Make predictions about future data and make decisions

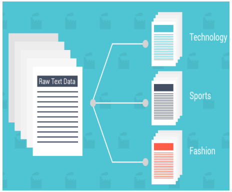




Machine Learning (ML)

- Designing algorithms that ingest data and learn a (hypothesized) model of the data
- The learned model can be used to
  - Detect patterns/structures/themes/trends etc. in the data
  - Make predictions about future data and make decisions

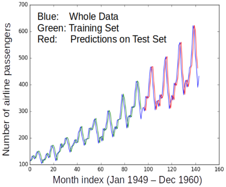


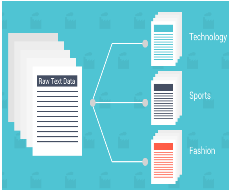


- Modern ML algorithms are heavily "data-driven"

Machine Learning (ML)

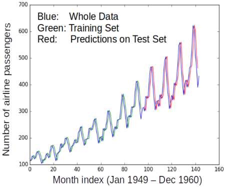
- Designing algorithms that ingest data and learn a (hypothesized) model of the data
- The learned model can be used to
  - Detect patterns/structures/themes/trends etc. in the data
  - Make predictions about future data and make decisions

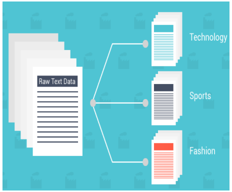




- Modern ML algorithms are heavily "data-driven"
  - No need to pre-define and hard-code all the rules (usually infeasible/impossible anyway)

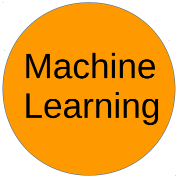
- Designing algorithms that ingest data and learn a (hypothesized) model of the data
- The learned model can be used to
  - Detect patterns/structures/themes/trends etc. in the data
  - Make predictions about future data and make decisions



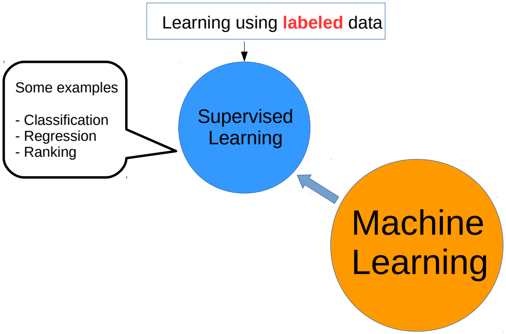


- Modern ML algorithms are heavily "data-driven"
  - No need to pre-define and hard-code all the rules (usually infeasible/impossible anyway)
  - The rules are not "static"; can adapt as the ML algo ingests more and more data

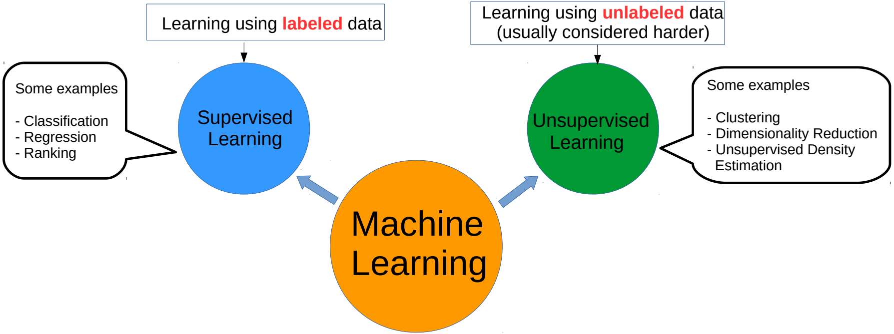
A Loose Taxonomy for ML



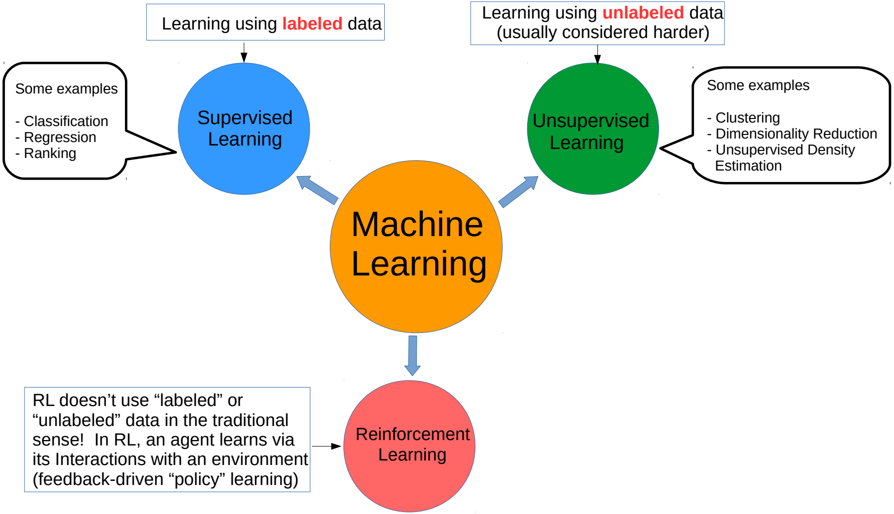
A Loose Taxonomy for ML



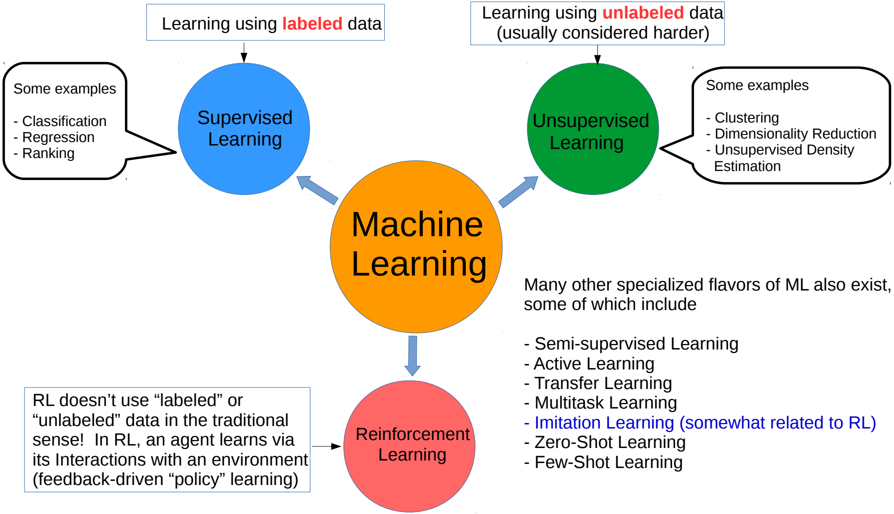
A Loose Taxonomy for ML



A Loose Taxonomy for ML



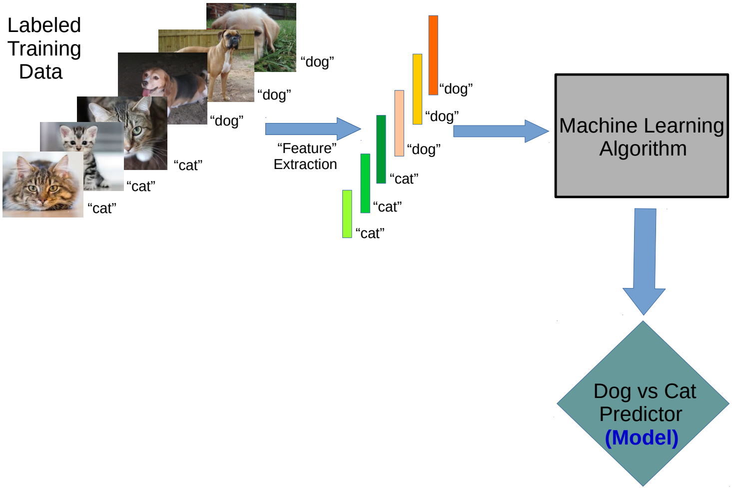
A Loose Taxonomy for ML



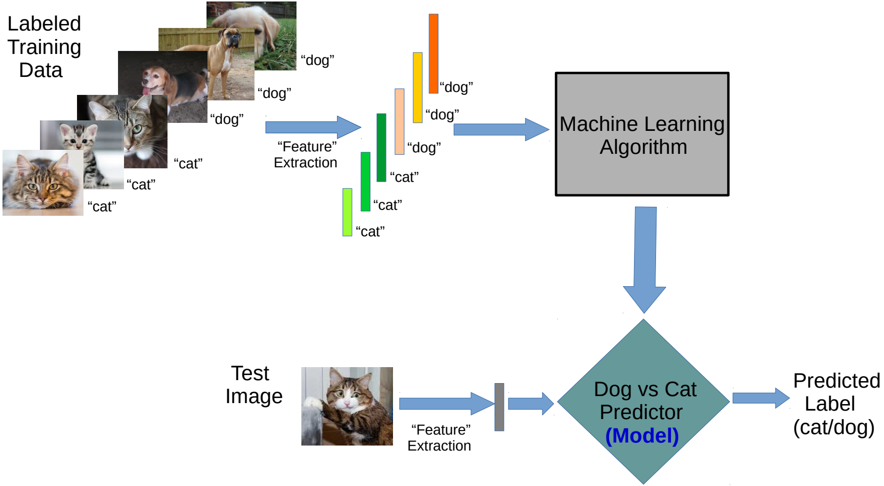
Many other specialized flavors of ML also exist, some of which include

- Semi-supervised Learning
- Active Learning
- Transfer Learning
- Multitask Learning
- Imitation Learning (somewhat related to RL)
- Zero-Shot Learning
- Few-Shot Learning

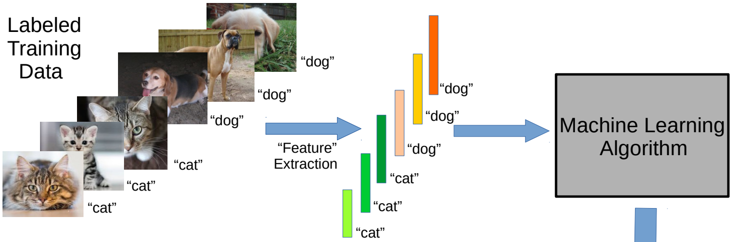
A Typical Supervised Learning Workflow (for Classification)



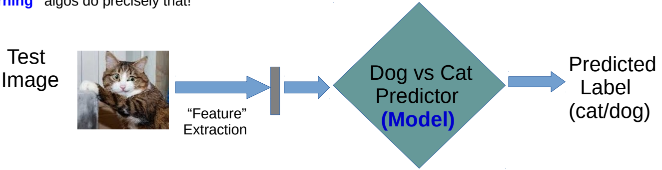
A Typical Supervised Learning Workflow (for Classification)



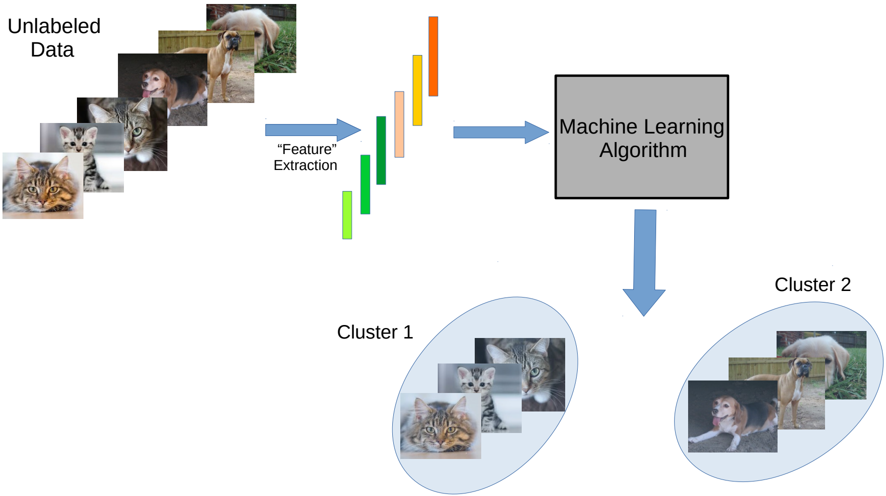
A Typical Supervised Learning Workflow (for Classification)



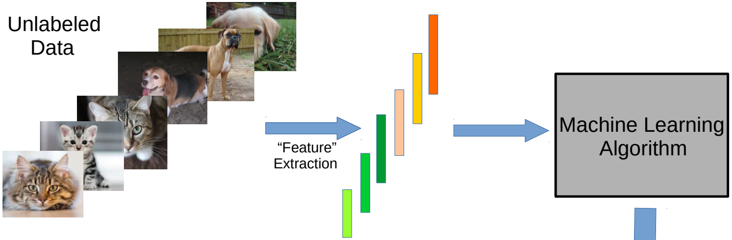
Note: The feature extraction phase may be part of the machine learning algorithm itself
(referred to "feature learning" or "representation learning")
Modern **"deep learning"** algos do precisely that!

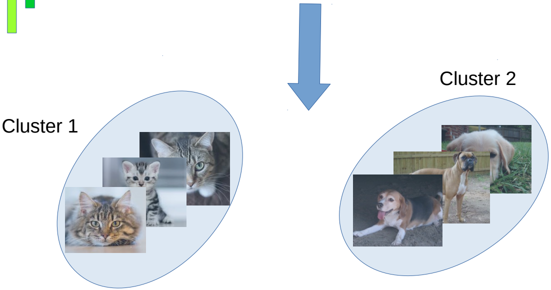


A Typical Unupervised Learning Workflow (for Clustering)



A Typical Unupervised Learning Workflow (for Clustering)

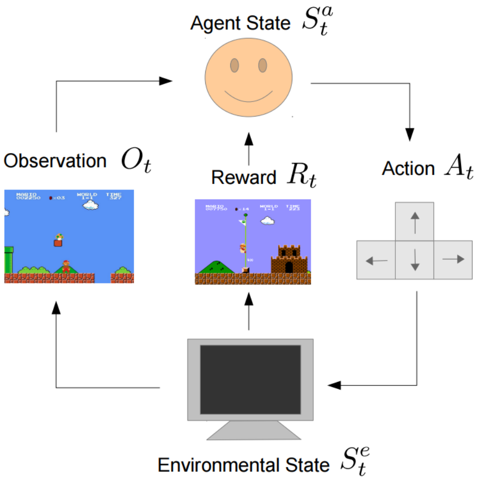




Note: Unsupervised Learning too can have (and often has) a "test" phase. E.g., in this case, given a new cat/dog image, predict which of the two clusters it belongs to.

Can do it by assigning the image to the cluster with closer centroid

A Typical Reinforcement Learning Workflow

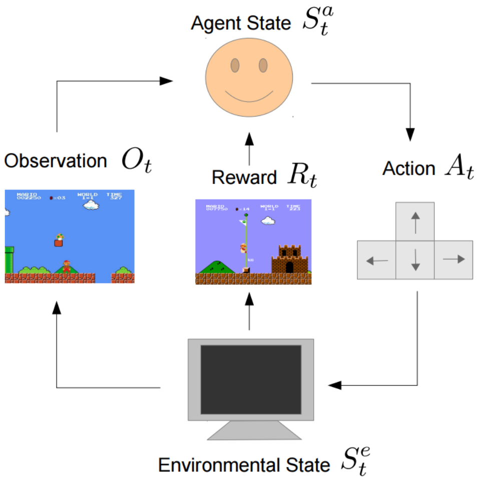


Agent's goal is to learn a policy for some task

Agent does the following repeatedly

- Senses/observes the environment
- Takes an action based on its current policy
- Receives a reward for that action
- Updates its policy

A Typical Reinforcement Learning Workflow



Agent's goal is to learn a policy for some task

Agent does the following repeatedly

- Senses/observes the environment
- Takes an action based on its current policy
- Receives a reward for that action
- Updates its policy

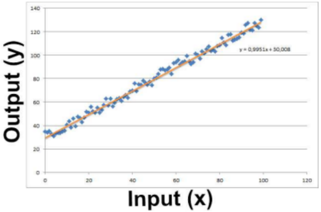
There IS supervision, not explicit (as in Supervised Learning) but rather implicit (feedback based)

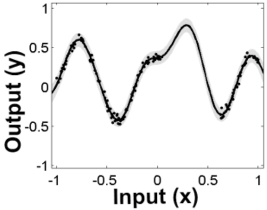
Geometric View of Some Basic ML Problems

Regression

Supervised Learning: Learn a line/curve (the "model") using training data consisting of Input-output pairs (each output is a real-valued number)

Use it to predict the outputs for new "test" inputs

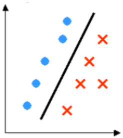




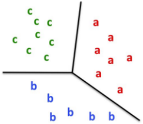
Classification

Supervised Learning: Learn a linear/nonlinear separator (the "model") using training data consisting of input-output pairs (each output is discrete-valued "label" of the corresponding input)

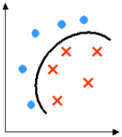
Use it to predict the labels for new "test" inputs



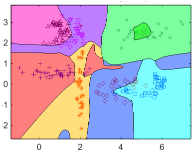
Two-Class (binary)
Linear Classification



Multi-Class
Linear Classification



Two-Class (binary)
Nonlinear Classification

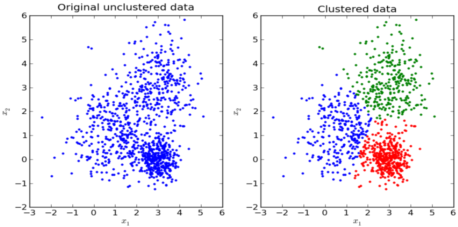


Multi-Class
Nonlinear Classification

Geometric View of Some Basic ML Problems

Clustering

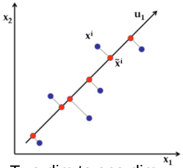
Unsupervised Learning: Learn the grouping structure for a given set of unlabeled inputs



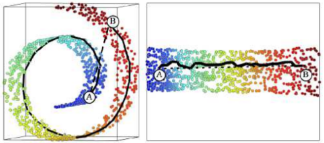
Dimensionality Reduction

Unsupervised Learning: Learn a Low-dimensional representation for a given set of high-dimensional inputs

Note: DR also comes in supervised flavors (supervised DR)

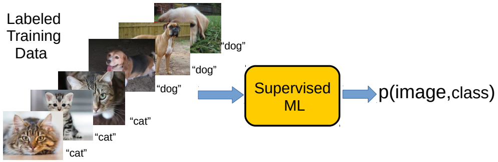


Two-dim to one-dim linear projection



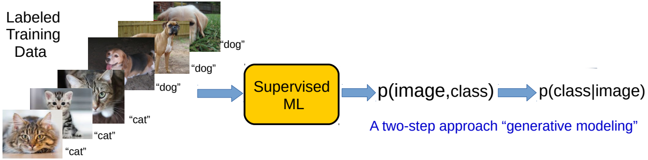
Three-dim to two-dim nonlinear projection (a.k.a. manifold learning)

- Supervised Learning ("predict y given $\boldsymbol{x}$") can be thought of as estimating $p(y|\boldsymbol{x})$



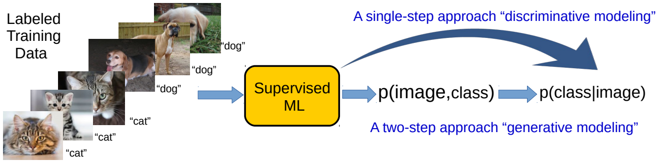
Machine Learning = Probability Density Estimation

- Supervised Learning ("predict y given $\boldsymbol{x}$") can be thought of as estimating $p(y|\boldsymbol{x})$



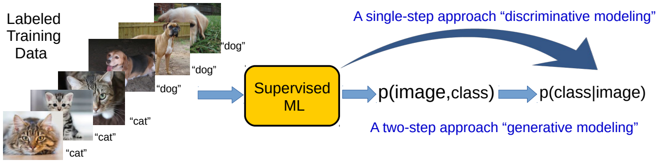
Machine Learning = Probability Density Estimation

- Supervised Learning ("predict y given $\boldsymbol{x}$") can be thought of as estimating $p(y|\boldsymbol{x})$

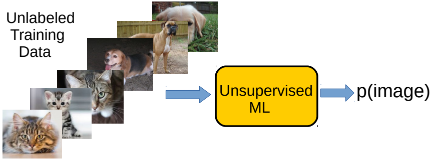


Machine Learning = Probability Density Estimation

- Supervised Learning ("predict y given $\boldsymbol{x}$") can be thought of as estimating $p(y|\boldsymbol{x})$

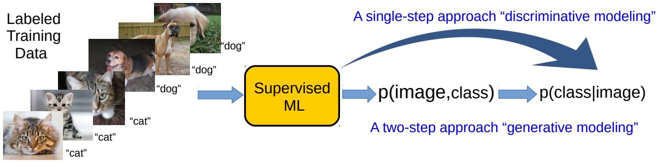


- Unsupervised Learning ("model $\boldsymbol{x}$") can also be thought of as estimating $p(\boldsymbol{x})$

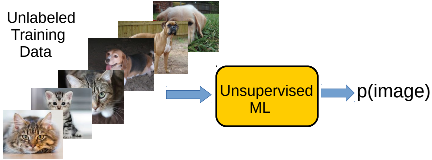


Machine Learning = Probability Density Estimation

- Supervised Learning ("predict y given $\boldsymbol{x}$") can be thought of as estimating $p(y|\boldsymbol{x})$



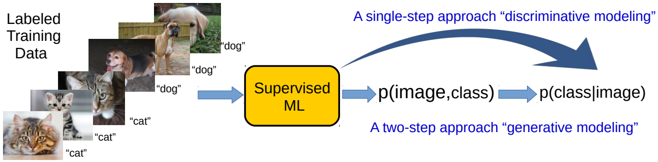
- Unsupervised Learning ("model $\boldsymbol{x}$") can also be thought of as estimating $p(\boldsymbol{x})$



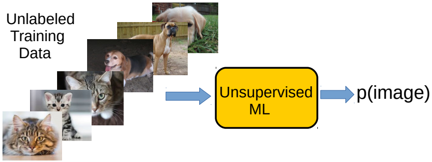
- Harder for Unsupervised Learning because there is no supervision y

Machine Learning = Probability Density Estimation

- Supervised Learning ("predict y given $\boldsymbol{x}$") can be thought of as estimating $p(y|\boldsymbol{x})$



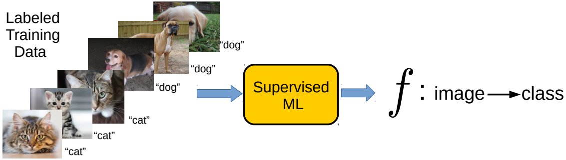
- Unsupervised Learning ("model $\boldsymbol{x}$") can also be thought of as estimating $p(\boldsymbol{x})$



- Harder for Unsupervised Learning because there is no supervision y
- Other ML paradigms (e.g., Reinforcement Learning) can be thought of as learning prob. density

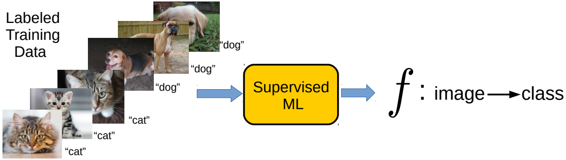
Machine Learning = Function Approximation

- Supervised Learning ("predict y given $\boldsymbol{x}$") can be thought learning a function that maps $\boldsymbol{x}$ to y

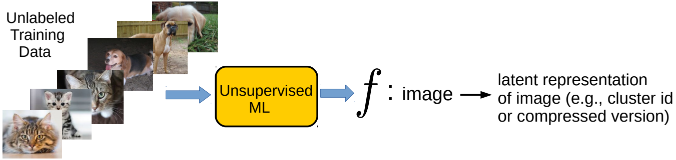


Machine Learning = Function Approximation

- Supervised Learning ("predict y given $\boldsymbol{x}$") can be thought learning a function that maps $\boldsymbol{x}$ to y

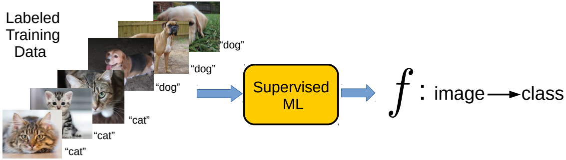


- Unsupervised Learning ("model $\boldsymbol{x}$") can also be thought of as learning a function that maps $\boldsymbol{x}$ to some useful latent representation of $\boldsymbol{x}$

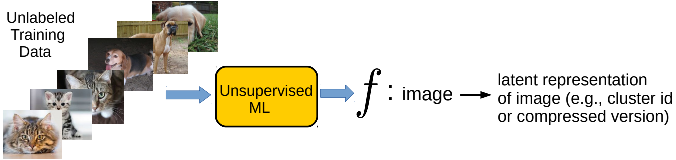


Machine Learning = Function Approximation

- Supervised Learning ("predict y given $\boldsymbol{x}$") can be thought learning a function that maps $\boldsymbol{x}$ to y



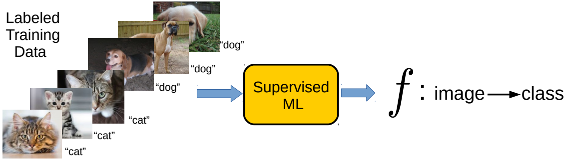
- Unsupervised Learning ("model $\boldsymbol{x}$") can also be thought of as learning a function that maps $\boldsymbol{x}$ to some useful latent representation of $\boldsymbol{x}$



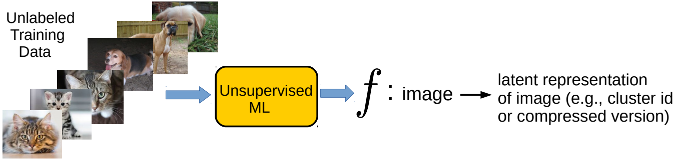
- Harder for Unsupervised Learning because there is no supervision y

Machine Learning = Function Approximation

- Supervised Learning ("predict y given $\boldsymbol{x}$") can be thought learning a function that maps $\boldsymbol{x}$ to y



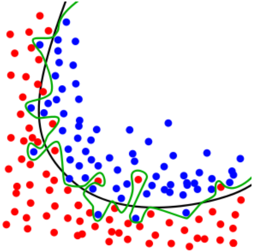
- Unsupervised Learning ("model $\boldsymbol{x}$") can also be thought of as learning a function that maps $\boldsymbol{x}$ to some useful latent representation of $\boldsymbol{x}$



- Harder for Unsupervised Learning because there is no supervision y
- Other ML paradigms (e.g., Reinforcement Learning) can be thought of as doing function approx.

Overfitting and Generalization

- Doing well on the training data is not enough for an ML algorithm

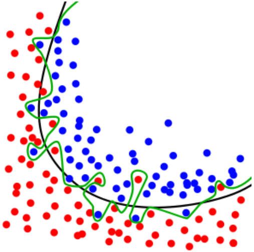


- Trying to do too well (or perfectly) on training data may lead to bad "generalization"

Picture courtesy: Wikipedia

Overfitting and Generalization

- Doing well on the training data is not enough for an ML algorithm

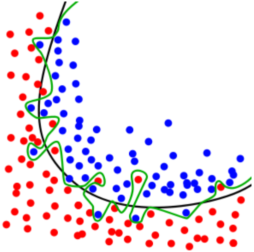


- Trying to do too well (or perfectly) on training data may lead to bad "generalization"
- Generalization: Ability of an ML algorithm to do well on future "test" data

Picture courtesy: Wikipedia

Overfitting and Generalization

- Doing well on the training data is not enough for an ML algorithm

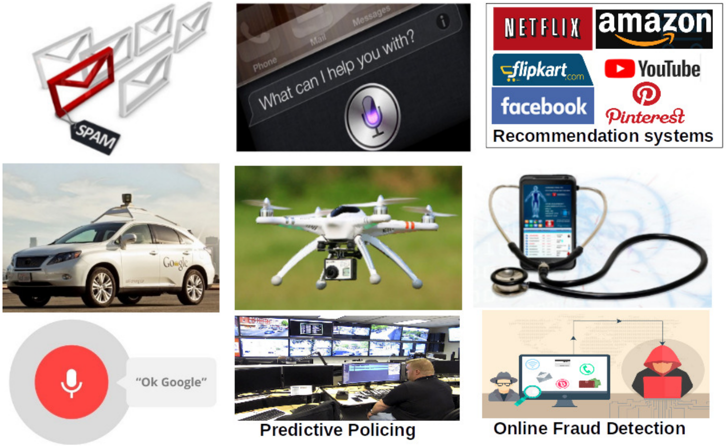


- Trying to do too well (or perfectly) on training data may lead to bad "generalization"
- Generalization: Ability of an ML algorithm to do well on future "test" data
- Simple models/functions tend to prevent overfiting and generalize well: A key principle in designing ML algorithms (called "regularization"; more on this later)

Picture courtesy: Wikipedia

Machine Learning in the real-world

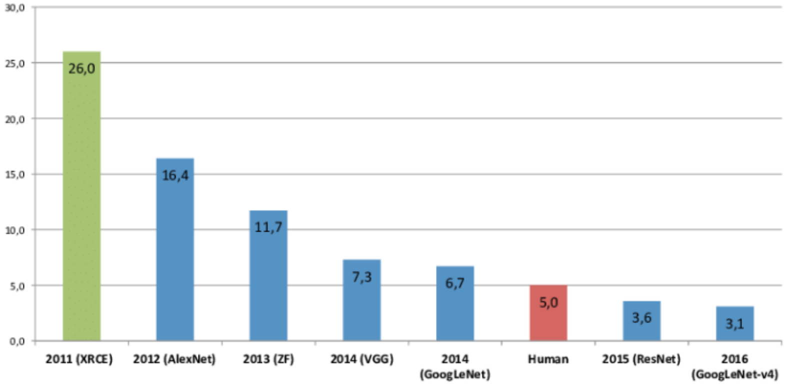
Broadly applicable in many domains (e.g., internet, robotics, healthcare and biology, computer vision, NLP, databases, computer systems, finance, etc.).



Picture courtesy: gizmodo.com,rcdronearena.com,www.wiseyak.com,www.charlesdong.com

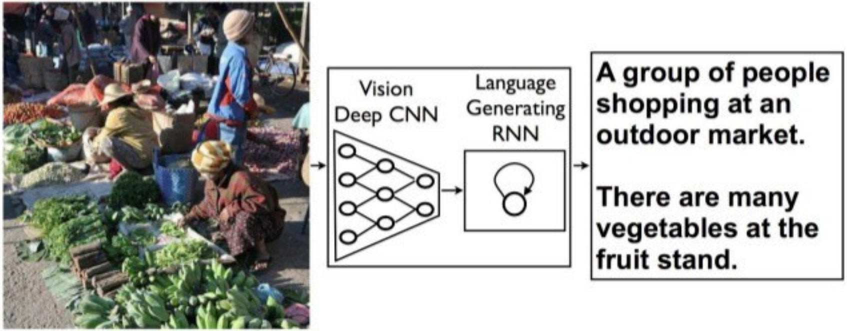
Machine Learning helps Computer Vision

ML algorithms can learn to recognize images better than humans!



Machine Learning helps Computer Vision

ML algorithms can learn to generate captions for images



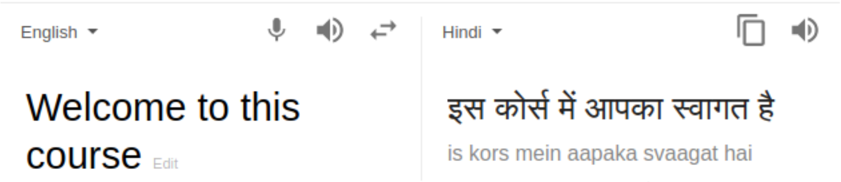
http://arxiv.org/abs/1411.4555 "Show and Tell: A Neural Image Caption Generator"

Machine Learning helps Computer Vision

ML algorithms can learn to answer questions about images (Visual QA)

Machine Learning helps NLP

ML algorithms can learn to translate text



(even “transliterate”)

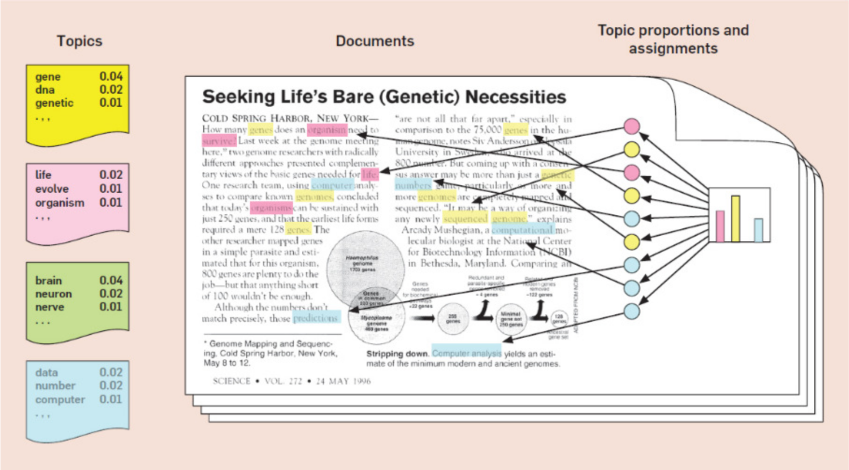
Machine Learning helps NLP

ML algorithms can learn to summarize text

Input: Article 1st sentence	Model-written headline
metro-goldwyn-mayer reported a third-quarter net loss of dlrs 16 million due mainly to the effect of accounting rules adopted this year	mgm reports 16 million net loss on higher revenue
starting from july 1, the island province of hainan in southern china will implement strict market access control on all incoming livestock and animal products to prevent the possible spread of epidemic diseases	hainan to curb spread of diseases
australian wine exports hit a record 52.1 million liters worth 260 million dollars (143 million us) in september, the government statistics office reported on monday	australian wine exports hit record high in september

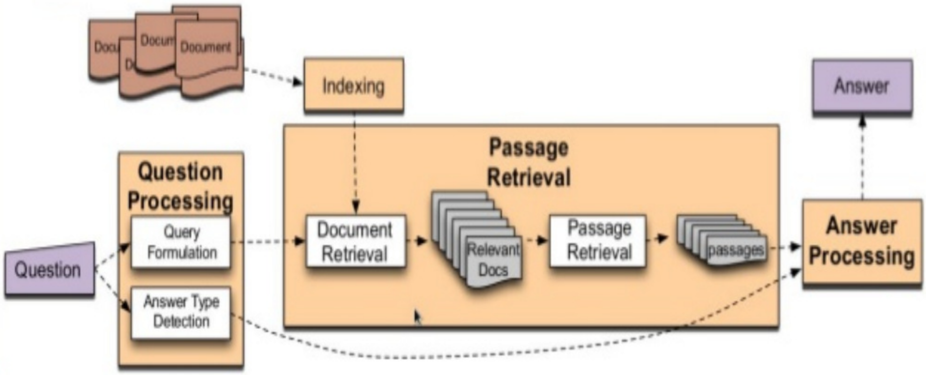
Machine Learning helps NLP

ML algorithms can learn the topics in a text corpus ("Topic Modeling")



Machine Learning helps Search and Info Retrieval

ML algorithms can learn to search for the answer to a given question from a large database of documents



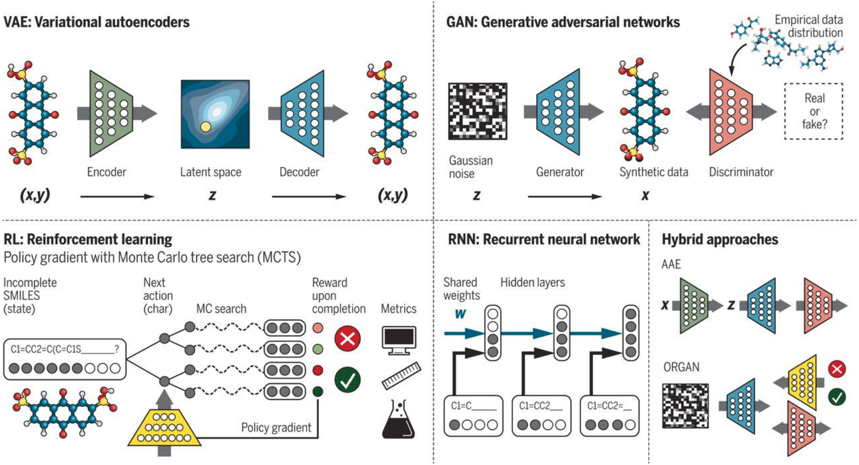
Machine Learning meets Speech Processing

ML algorithms can learn to translate speech in **real time**

Picture courtesy: https://news.microsoft.com/

Machine Learning helps Chemistry

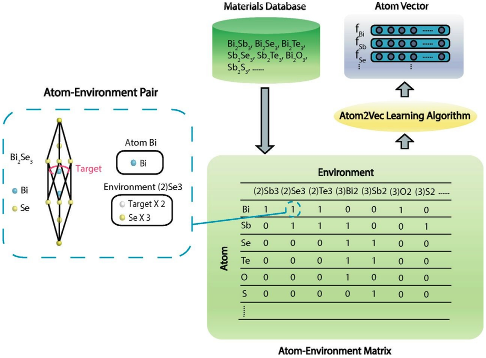
ML algorithms can understand properties of molecules and learn to synthesize new molecules



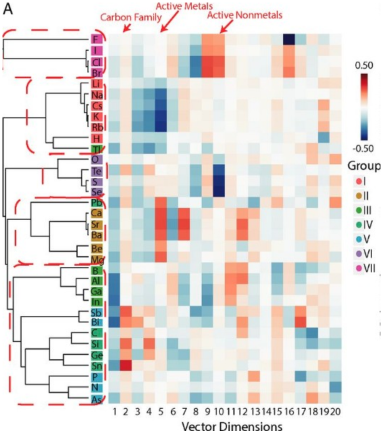
Picture courtesy: Inverse molecular design using machine learning: Generative models for matter engineering (Science, 2018)

Machine Learning helps Chemistry

ML algorithms can "read" databases of matetials and recreate the Periodic Table within hours



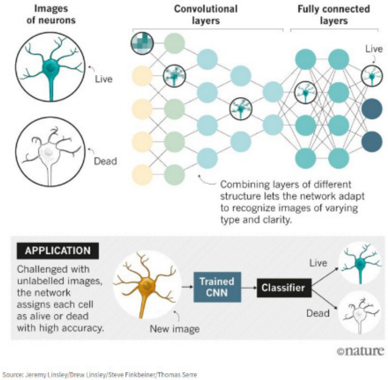
"Recreated" Periodic Table



Picture courtesy: Learning atoms for materials discovery (PNAS, 2018)

Machine Learning helps Many Other Areas..

Biology



Finance

Picture courtesy: (1) https://www.nature.com/articles/d41586-018-02174-z (2) https://responsiblefinanceforum.org

Machine Learning: A Brief Timeline and Some Milestones

1750-1900

- Origins of Bayes Theorem: Thomas Bayes and Pierre-Simon Laplace (~1800)
- Least Squares method: Legendre (1805)

1900-1950

- Early works on PCA (Peason), Factor Analysis (Spearman), CCA (by Hotelling), for exploratory data analysis
- Early works on Discriminant Analysis methods (Fisher) for classification
- McCulloch-Pitts model of the artificial neuron (1943)
- Turing predicted that Machine Learning is possible (1947)
- Design of the "Turing Test" (1950)

1950-1960

- Minsky & Edmonds' neural net machine (SNARC)
- Arthur Samuels' Checkers Player based on Machine Learning
- Rosenblatt's Perceptron algo
- Widrow-Hoff's ADALINE algo
- K-means clustering algo (Lloyd)
- Early origins of Reinforcement Learning (Bellman and others)

1960-1970

- Multi-layer ADALINE (called MADALINE)
- Nearest Neighbors algorithm
- Early works on Genetic Algorithms inspired by natural evolution (John Holland)

1970-1980

- Minsky-Papert showed that Perceptron can't learn XOR function (and nonlinear functions in general). Led to the so-called "AI Winter"
- Automatic Differentiation (later rediscovered as Backprop)

1980-1990

- Fukushima's Neurocognitron model for visual recognition
- Multi-layer Perceptrons (can learn nonlinear functions). Sunny days of AI are back!
- The Backpropagation algorithm to train deep neural nets
- Decision Trees, ID3 (Quinlan)
- NetTalk: Neural nets that can learn to pronounce English words
- Modern Reinforcement Learning

1990-2000

- Neural nets slumber due to lack of compute power
- Support Vector Machines (SVM), Kernel methods, Bayesian methods
- Random Forests, Boosting
- Continued work on neural nets for images, sequences (CNN, LSTM, etc)
- Deep Blue beats Kasparov

2000-2010

- Continued focus on classical statistical and probabilistic models, connections b/w learning & cognition
- Areas like vision, NLP, speech embrace data-driven ML methods as opposed to rule-based methods
- ML based algorithm wins the Netflix Challenge
- Neural nets re-emerged and rebranded as Deep Learning (Hinton, Bengio, LeCun, Ng, and others), thanks to improved training, GPUs

2010-2020

- Watson beats Humans in Jeopardy!
- Advances in ML/deep learning result in breakthroughs in areas like vision, NLP, speech, etc.
- Software frameworks (e.g., Tensorflow, PyTorch, ease implementing ML algos)
- Drones, self-driving cars, etc.
- A lot of industry/media focus, excitement, and hype
- Focus on Fairness, Accountability, and Transparency in ML algorithms

Road Ahead

Another AI Winter ?

(Tentative) List of topics

- Supervised Learning
  - nearest-neighbors methods, decision trees
  - linear/non-linear regression and classification
- Unsupervised Learning
  - Clustering and density estimation
  - Dimensionality reduction and manifold learning
  - Latent factor models and matrix factorization
- Probabilistic Modeling
- Deep Learning
- Ensemble Methods
- Learning from sequential data
- Recent advances in ML

Course Goals

By the end of the semester, you should be able to:

- Understand how various machine learning algorithms work
- Implement them (and, hopefully, their variants/improvements) on your own
- Look at a real-world problem and identify if ML is an appropriate solution
- If so, identify what types of algorithms might be applicable
- Feel inspired to work on and learn more about Machine Learning :-)

Course Goals

By the end of the semester, you should be able to:

- Understand how various machine learning algorithms work
- Implement them (and, hopefully, their variants/improvements) on your own
- Look at a real-world problem and identify if ML is an appropriate solution
- If so, identify what types of algorithms might be applicable
- Feel inspired to work on and learn more about Machine Learning :-)

Caution: There will be quite a bit of maths in this course (can't be avoided!). You are expected to be (or to make yourself) comfortable with multivariate calculus, linear algebra, probability and statistics. Please use the provided reference materials to brush up these concepts.