Course Logistics and Introduction to Machine Learning

Piyush Rai

Machine Learning (CS771A)

July 28, 2016

Course Logistics

- **Timing and Venue:** WF 6:00-7:30pm, RM 101

Course Logistics

- **Timing and Venue:** WF 6:00-7:30pm, RM 101
- **Course website:** `http://goo.gl/IrN4N1`. Please bookmark it.
- **Instructor:** Piyush Rai (Email: piyush@cse.iitk.ac.in)
- **Discussion site:** Use Piazza (`https://goo.gl/Kkb0vX`). Please register.

Course Logistics

- **Timing and Venue:** WF 6:00-7:30pm, RM 101
- **Course website:** `http://goo.gl/IrN4N1`. Please bookmark it.
- **Instructor:** Piyush Rai (Email: piyush@cse.iitk.ac.in)
- **Discussion site:** Use Piazza (`https://goo.gl/Kkb0vX`). Please register.
- **Background assumed:** basics of linear algebra, multivariate calculus, probability and statistics, optimization, programming (MATLAB).

Course Logistics

- **Timing and Venue:** WF 6:00-7:30pm, RM 101
- **Course website:** `http://goo.gl/IrN4N1`. Please bookmark it.
- **Instructor:** Piyush Rai (Email: piyush@cse.iitk.ac.in)
- **Discussion site:** Use Piazza (`https://goo.gl/Kkb0vX`). Please register.
- **Background assumed:** basics of linear algebra, multivariate calculus, probability and statistics, optimization, programming (MATLAB).
- **Grading:**
  - 4 homework assignments: 40%, Midterm exam: 20%, Final exam: 20%
  - Project: 20% (to be done in groups of 4-5; more details forthcoming)

Course Logistics

- **Timing and Venue:** WF 6:00-7:30pm, RM 101
- **Course website:** `http://goo.gl/IrN4N1`. Please bookmark it.
- **Instructor:** Piyush Rai (Email: piyush@cse.iitk.ac.in)
- **Discussion site:** Use Piazza (`https://goo.gl/Kkb0vX`). Please register.
- **Background assumed:** basics of linear algebra, multivariate calculus, probability and statistics, optimization, programming (MATLAB).
- **Grading:**
  - 4 homework assignments: 40%, Midterm exam: 20%, Final exam: 20%
  - Project: 20% (to be done in groups of 4-5; more details forthcoming)
  - Note: Exams will be closed-book (an A4 size cheat-sheet allowed)

Course Logistics

- **Timing and Venue:** WF 6:00-7:30pm, RM 101
- **Course website:** `http://goo.gl/IrN4N1`. Please bookmark it.
- **Instructor:** Piyush Rai (Email: piyush@cse.iitk.ac.in)
- **Discussion site:** Use Piazza (`https://goo.gl/Kkb0vX`). Please register.
- **Background assumed:** basics of linear algebra, multivariate calculus, probability and statistics, optimization, programming (MATLAB).
- **Grading:**
  - 4 homework assignments: 40%, Midterm exam: 20%, Final exam: 20%
  - Project: 20% (to be done in groups of 4-5; more details forthcoming)
  - Note: Exams will be closed-book (an A4 size cheat-sheet allowed)
- **Textbook:** No official textbook required
  - Required reading material will be provided on the class webpage

Course Logistics

- **Timing and Venue:** WF 6:00-7:30pm, RM 101
- **Course website:** `http://goo.gl/IrN4N1`. Please bookmark it.
- **Instructor:** Piyush Rai (Email: piyush@cse.iitk.ac.in)
- **Discussion site:** Use Piazza (`https://goo.gl/Kkb0vX`). Please register.
- **Background assumed:** basics of linear algebra, multivariate calculus, probability and statistics, optimization, programming (MATLAB).
- **Grading:**
  - 4 homework assignments: 40%, Midterm exam: 20%, Final exam: 20%
  - Project: 20% (to be done in groups of 4-5; more details forthcoming)
  - Note: Exams will be closed-book (an A4 size cheat-sheet allowed)
- **Textbook:** No official textbook required
  - Required reading material will be provided on the class webpage
- **Auditing?** Please let me know your email id to be added to the mailing list.

Intro to Machine Learning

Machine Learning

- Creating programs that can automatically learn rules from data

  "Field of study that gives computers the ability to learn without being explicitly programmed" (Arthur Samuel, 1959)

Machine Learning

- Creating programs that can automatically learn rules from data

  "Field of study that gives computers the ability to learn without being explicitly programmed" (Arthur Samuel, 1959)

- Traditional algorithms vs Machine Learning algorithms:
  - Traditional: Write programs using hard-coded (fixed) rules

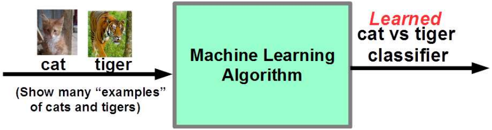
Machine Learning

- Creating programs that can automatically learn rules from data

  "Field of study that gives computers the ability to learn without being explicitly programmed" (Arthur Samuel, 1959)

- Traditional algorithms vs Machine Learning algorithms:
  - Traditional: Write programs using hard-coded (fixed) rules

  - Machine Learning (ML): Learn rules by looking at some training data



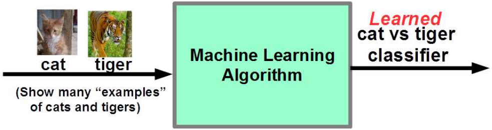
Machine Learning

- Creating programs that can automatically learn rules from data

  "Field of study that gives computers the ability to learn without being explicitly programmed" (Arthur Samuel, 1959)

- Traditional algorithms vs Machine Learning algorithms:
  - Traditional: Write programs using hard-coded (fixed) rules

  - Machine Learning (ML): Learn rules by looking at some training data



- Learned rules must generalize (do well) on future "test" data (idea of generalization; more later)

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 in the real-world

Some real-world applications

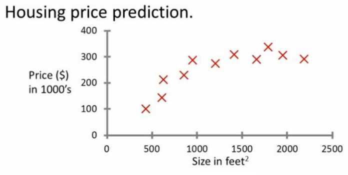
- Information retrieval (text, visual, and multimedia searches)
- Machine Translation
- Question Answering
- Social networks
- Recommender systems (Amazon, Netflix, etc.)
- Speech/handwriting/object recognition
- Ad placement on websites
- Credit-card fraud detection
- Weather prediction
- Autonomous vehicles (self-driving cars)
- Healthcare and life-sciences
- .. and many more applications in sciences and engineering

Supervised Learning

- Given: Training data as labeled instances $\{(x_1, y_1), \ldots, (x_N, y_N)\}$
- Goal: Learn a rule $(f : x \rightarrow y)$ to predict outputs y for new inputs x

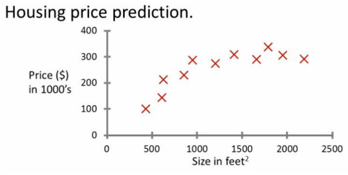
Supervised Learning

- Given: Training data as labeled instances $\{(x_1, y_1), \ldots, (x_N, y_N)\}$
- Goal: Learn a rule ($f : x \to y$) to predict outputs y for new inputs x
- Real-valued outputs (e.g., price of a house): **Regression**



Supervised Learning

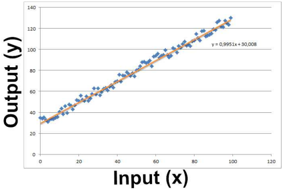
- Given: Training data as labeled instances $\{(x_1, y_1), \ldots, (x_N, y_N)\}$
- Goal: Learn a rule ($f : x \to y$) to predict outputs y for new inputs x
- Real-valued outputs (e.g., price of a house): **Regression**

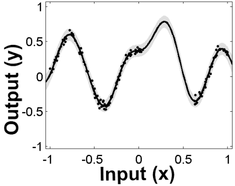


- Discrete-valued outputs (e.g., label of a hand-written digit): **Classification**

Supervised Learning: Pictorially

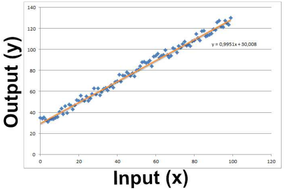
- Regression: fitting a line/non-linear curve

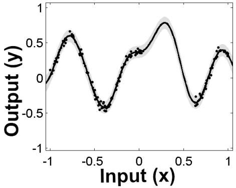




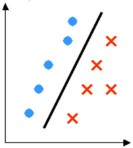
Supervised Learning: Pictorially

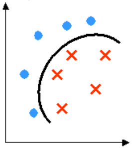
- Regression: fitting a line/non-linear curve





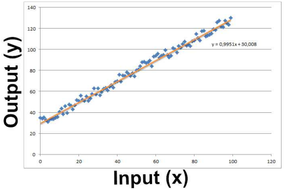
- Classification: finding a linear/nonlinear separator

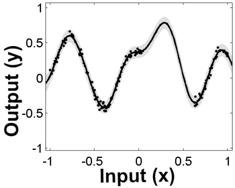




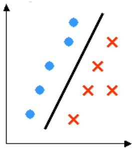
Supervised Learning: Pictorially

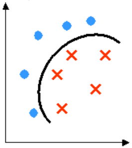
- Regression: fitting a line/non-linear curve





- Classification: finding a linear/nonlinear separator

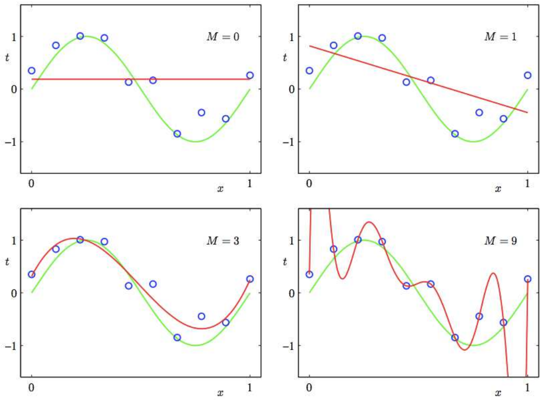




- **Generalization** is crucial (must do well on test data)

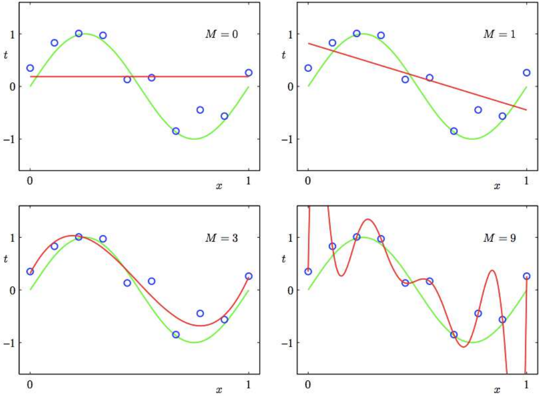
Generalization

- The right model complexity?



Generalization

- The right model complexity?



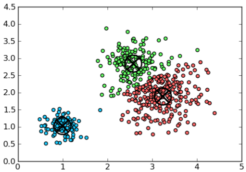
- Desired: hypotheses that are not too simple, not too complex (to avoid overfitting on training data)

Unsupervised Learning

- Given: Training data in form of unlabeled instances $\{x_1, \ldots, x_N\}$
- Goal: Learn the intrinsic latent *structure* that summarizes/explains data

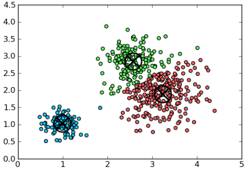
Unsupervised Learning

- Given: Training data in form of unlabeled instances $\{x_1, \ldots, x_N\}$
- Goal: Learn the intrinsic latent *structure* that summarizes/explains data
- Homogeneous groups as latent structure: **Clustering**

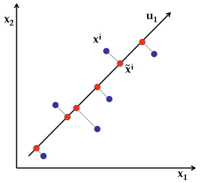


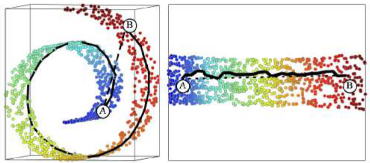
Unsupervised Learning

- Given: Training data in form of unlabeled instances $\{x_1, \ldots, x_N\}$
- Goal: Learn the intrinsic latent *structure* that summarizes/explains data
- Homogeneous groups as latent structure: **Clustering**



- Low-dimensional latent structure: **Dimensionality Reduction**





Unsupervised Learning: Some examples

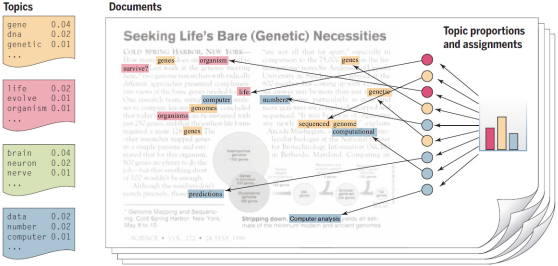
- Clustering large collections of images

Topic model picture courtesy: David Blei

Unsupervised Learning: Some examples

- Clustering large collections of images

- Topic discovery in large collections of text data

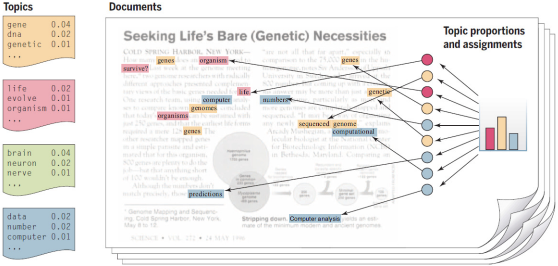


Topic model picture courtesy: David Blei

Unsupervised Learning: Some examples

- Clustering large collections of images

- Topic discovery in large collections of text data

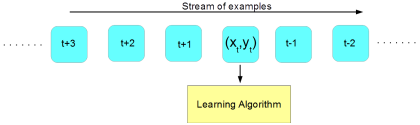


- Also used as a preprocessing step for many supervised learning algorithms (e.g., to learn/extract good features, to speed up the algorithms, etc.)

Topic model picture courtesy: David Blei

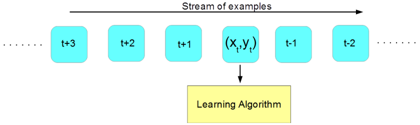
Some Other Learning Paradigms

- **Online Learning**
  - Learning with one example (or a small minibatch of examples) at a time

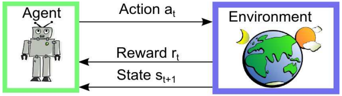


Some Other Learning Paradigms

- **Online Learning**
  - Learning with one example (or a small minibatch of examples) at a time



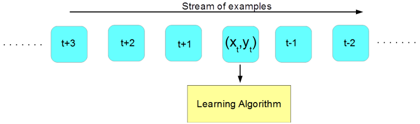
- **Reinforcement Learning**
  - Learning a "policy" by performing actions and getting rewards



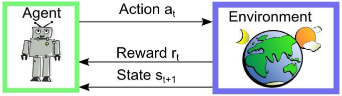
Reinforcement Learning Setup

Some Other Learning Paradigms

- **Online Learning**
  - Learning with one example (or a small minibatch of examples) at a time



- **Reinforcement Learning**
  - Learning a "policy" by performing actions and getting rewards



Reinforcement Learning Setup

- **Transfer/Multitask Learning**
  - Leveraging knowledge of solving one problem to solve a new problem

(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
- Online Learning
- Learning Theory
- Ensemble Methods
- Deep Learning
- Learning from time-series data

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 :-)

This class is **not** about:

- Introduction to machine learning tools/softwares