3D geoinformation
Department of Urbanism
Faculty of Architecture and the Built Environment
Delft University of Technology

GEO5017
Machine Learning for the Built Environment
https://3d.bk.tudelft.nl/courses/geo5017/

Lecture
Introduction

Liangliang Nan
https://3d.bk.tudelft.nl/liangliang/

Agenda

- **Lecture 1: Introduction to machine learning**
  - What do students expect?
  - What is machine learning
  - Applications of machine learning
  - The history of machine learning
  - Machine learning in this course
  - The advantages and disadvantages of machine learning
- **Organization of GEO5017**
  - The teachers
  - The course
  - Course logistics

What do students expect?

- Why do you choose this course?
- What do you want to learn from this course?
- What problems do you want to solve?

What is machine learning?

- Ways people have tried to define machine learning
  - *A field of study that gives computers the ability to learn without being explicitly programmed* - **Arthur Samuel**

Known for

- Pioneer in Machine Learning
- Development of TeX project (with Donald Knuth)
- Checkers-playing program

What is machine learning?

- Ways people have tried to define machine learning
    - *A field of study that gives computers the ability to learn without being explicitly programmed* - **Arthur Samuel**
    - *A computer program is said to learn from experience E with respect to some task T and some performance measure P, if its performance on T, as measured by P, improves with experience E.* - **Tom Mitchell**

Known for

- contributions to ML and AI
- Author of textbook ”Machine Learning”

What is machine learning?

- Ways people have tried to define machine learning
  - *A field of study that gives computers the ability to learn without being explicitly programmed* - **Arthur Samuel**
  - *A computer program is said to learn from experience E with respect to some task T and some performance measure P, if its performance on T, as measured by P, improves with experience E.* - **Tom Mitchell**
  - *Machine learning is the study of computer algorithms that can improve automatically through experience and by the use of data. Machine learning algorithms build a model based on sample data, known as training data, in order to make predictions or decisions without being explicitly programmed to do so.* - ***Wikipedia***

What is machine learning?

A computer program is said to learn from experience E with respect to some task T and some performance measure P if its performance on T, as measured by P, improves with experience E. - ***Tom Mitchell***

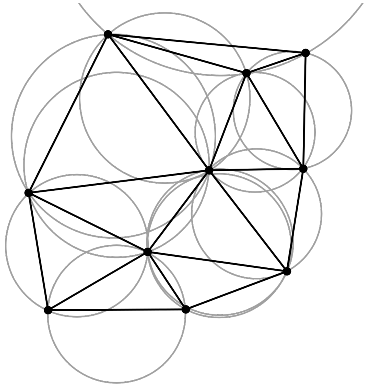
Suppose we feed a learning algorithm a lot of historical weather data, and have it learned to predict weather. What would be a reasonable choice for P?

A. The process of the algorithm examining a large amount of historical weather data.
B. The weather prediction task.
C. The probability of it correctly predicting a future date's weather.
D. None of these.

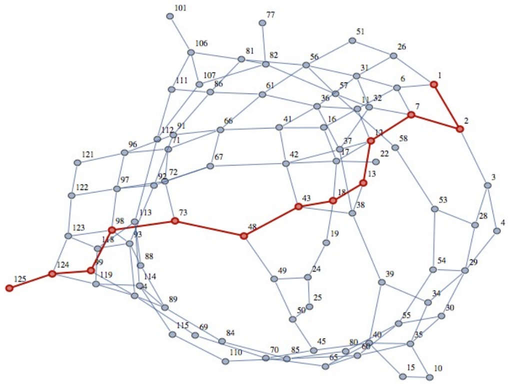
What are machine learning algorithms?

$$\begin{cases} 2x - 3y + z = -1 \\ x - y + 2z = -3 \\ 3x + y - z = 9 \end{cases}$$

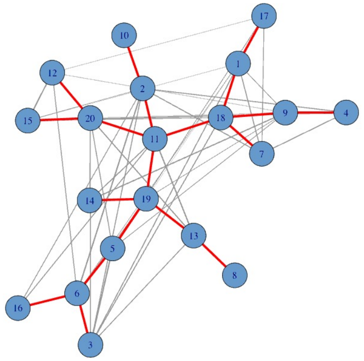
Equation solving



Delaunay triangulation



Shortest path



Minimum spanning tree

Face recognition

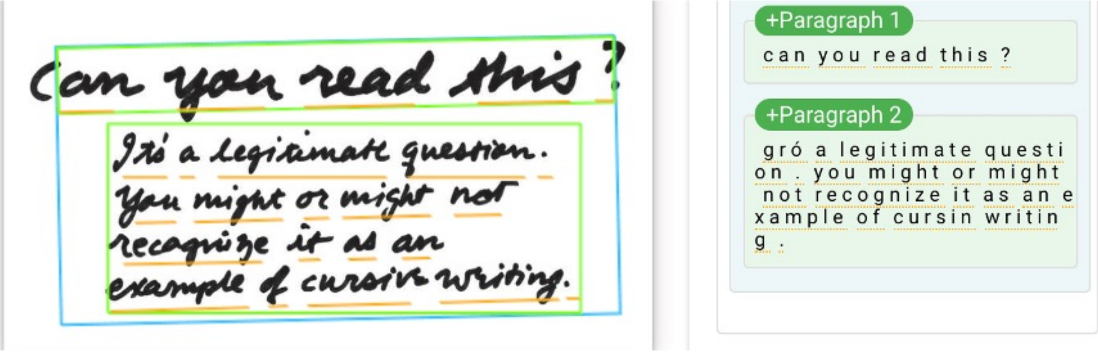
Autonomous driving

Spam filtering

Recommender systems

Applications of machine learning

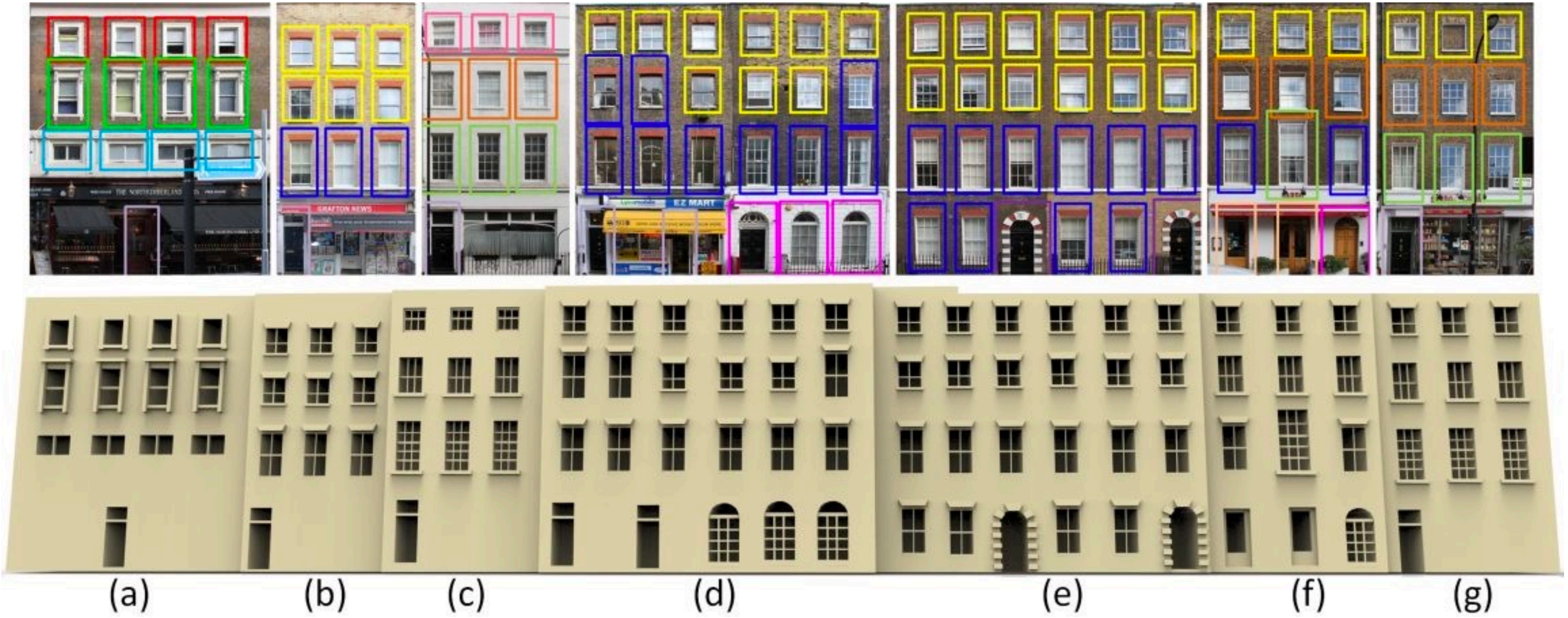
- Self-driving cars
- Face recognition
- Handwriting recognition
- Amazon product recommendation
- Spam filtering
- Automatic translation
- Speech recognition
- …



Applications of machine learning

Applications of machine learning

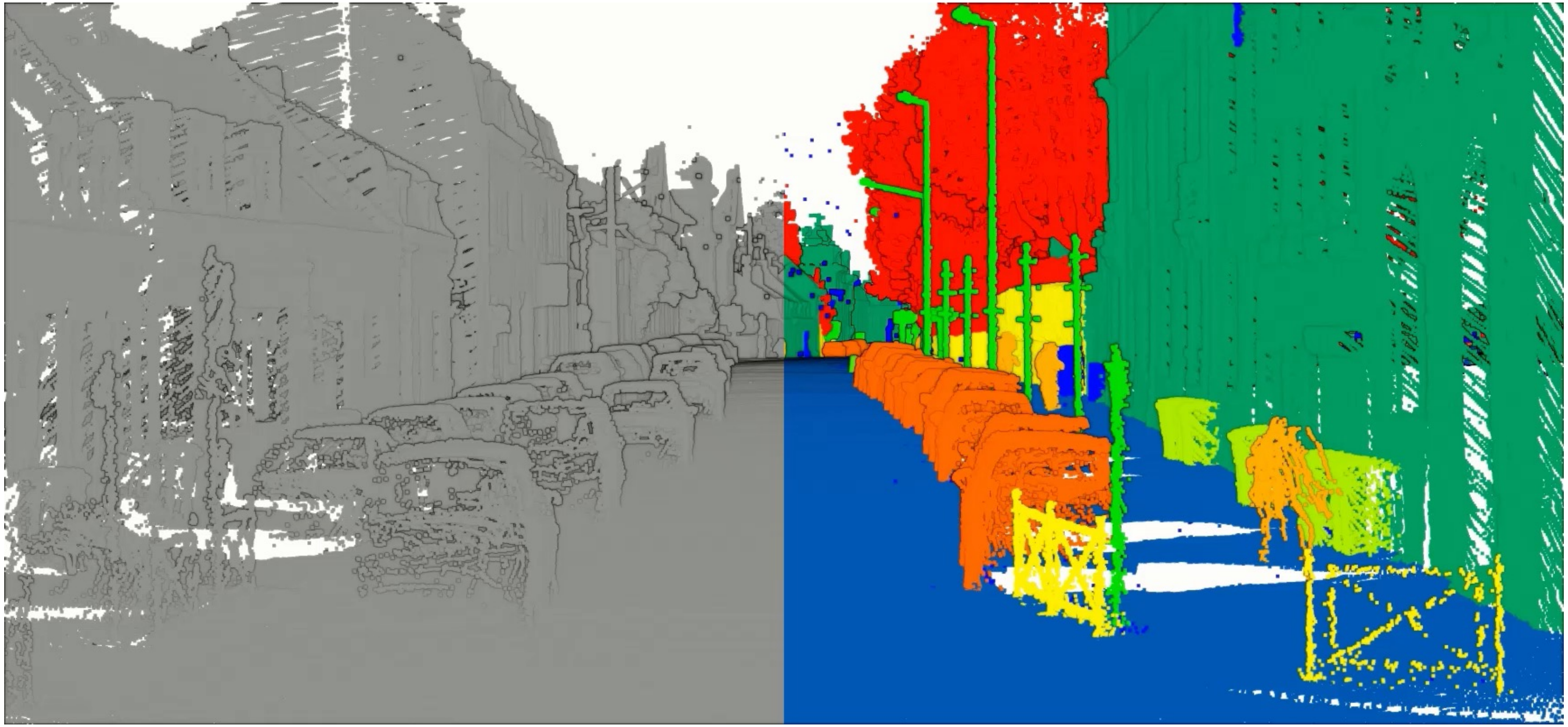
- Façade parsing and its applications in 3D modeling



Nan et al. Template Assembly for Detailed Urban Reconstruction. *Computer Graphics Forum, Vol. 34, No. 2, 2015*

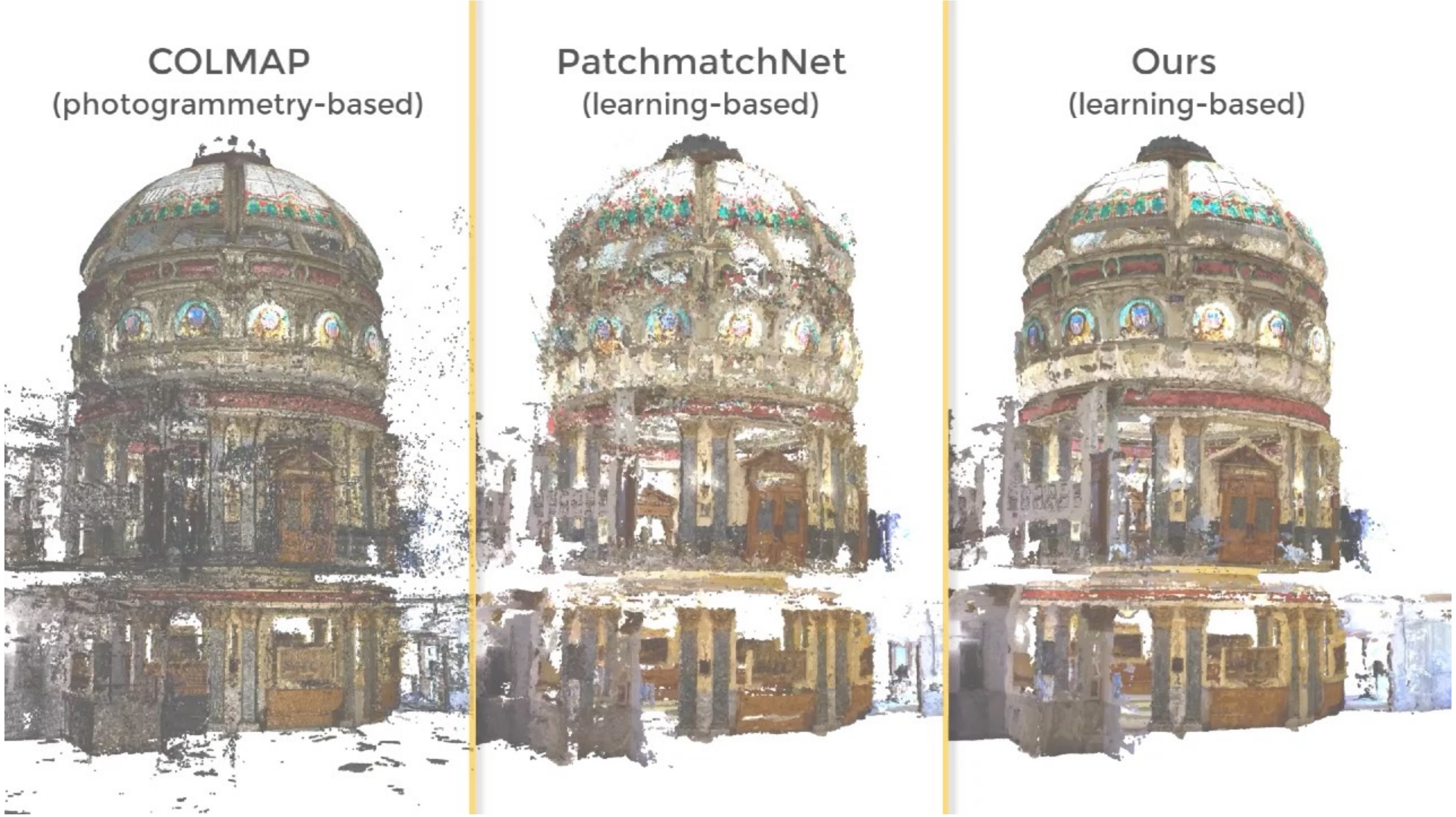
Applications of machine learning

- Semantic segmentation



Applications of machine learning

- 3D reconstruction from images



History of machine learning

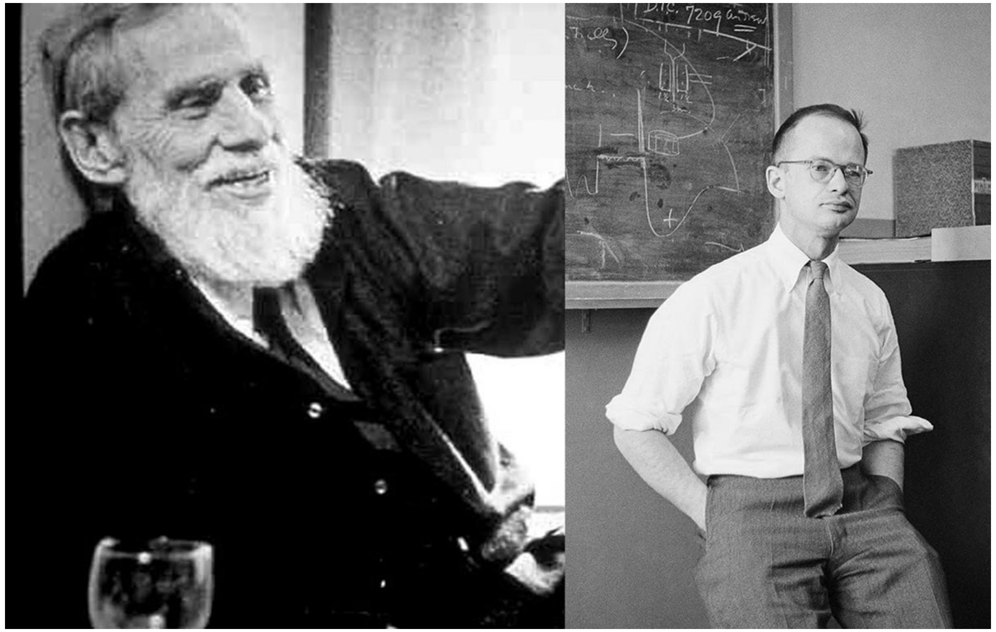
- **1943: first mathematical model of neural networks**
  - Warren McCulloch (left) and Walter Pitts (right)

BULLETIN OF
MATHEMATICAL BIOPHYSICS
VOLUME 5, 1943

A LOGICAL CALCULUS OF THE
IDEAS IMMANENT IN NERVOUS ACTIVITY

WARREN S. MCCULLOCH AND WALTER PITTS

FROM THE UNIVERSITY OF ILLINOIS, COLLEGE OF MEDICINE,
DEPARTMENT OF PSYCHIATRY AT THE ILLINOIS NEUROPSYCHIATRIC INSTITUTE,
AND THE UNIVERSITY OF CHICAGO



History of machine learning

- **1956: championship-level computer checkers game**
    - Not explore each and every possible path
    - But measure chances of winning
    - Mechanisms to continuously improve

Arthur Samuel is the first person to come up with and popularize the term "machine learning".

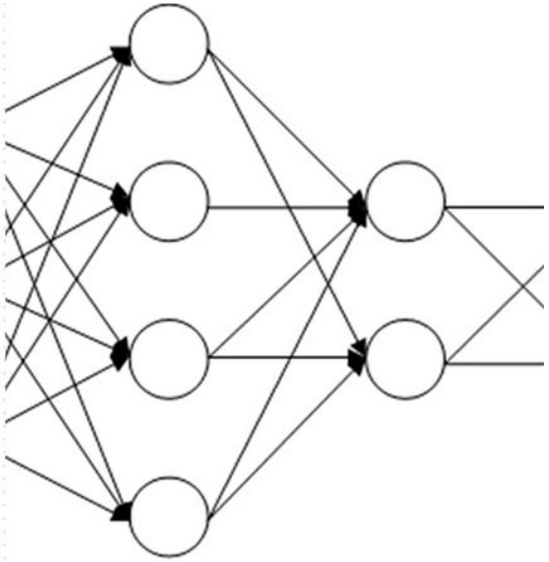
Arthur Samuel and IBM 700

History of machine learning

- **1965:** first Deep Neural Network by Alexey Ivakhnenko
  - Hierarchical representation of neural network
  - First multi-layer perceptron
  - Alexey Ivakhnenko considered the father of deep learning

  - Not popular until around 2010
    - Limited computing power
    - Lack of annotated data

О.Г. Івахненко (1967 р.)



History of machine learning

- **1967: Nearest Neighbor Pattern Classification**
  - Basic idea: It assigns to an unclassified sample point the classification of the nearest of a set of previously classified points.

IEEE TRANSACTIONS ON INFORMATION THEORY, VOL. IT-13, NO. 1, JANUARY 1967 21

ACKNOWLEDGMENT

The author is grateful to Prof. S. J. Mason of M.I.T. for his interest in this work, and for his many helpful suggestions. The author also wishes to thank Prof. K. N. Stevens and Prof. M. Eden for some very helpful discussions, and Prof. D. E. Troxel for his help in designing the sensory display.

REFERENCES

[1] I. Pollack, "The information of elementary auditory displays," *J. Acoust. Soc. Am.*, vol. 24, pp. 745–749, November 1952.
[2] ——, "The information of elementary auditory displays II," *J. Acoust. Soc. Am.*, vol. 25, pp. 765–769, July 1953.
[3] I. Pollack and L. Ficks, "Information of multidimensional auditory displays," *J. Acoust. Soc. Am.*, vol. 26, pp. 155–158, March 1954.
[4] W. R. Garner, "An informational analysis of absolute judgements of loudness," *J. Exper. Psychol.*, vol. 46, pp. 373–380, November 1953.
[5] C. W. Ericsen, "Multidimensional stimulus differences and the accuracy of discrimination," Wright Air Development Center, Wright-Patterson AFB, Dayton, Ohio, Tech. Rept., 1954.
[6] N. S. Andersen, and P. S. Fitts, "Amount of information gained during brief exposures to numerals and colors," *J. Exper. Psychol.*, vol. 56, pp. 362–369, October 1958.
[7] G. A. Miller, "The magical number seven, plus or minus two: some limits on our capacity for processing information," *Psychol. Rev.*, vol. 63, pp. 81–97, March 1956.
[8] F. Attneave, *Applications of Information Theory to Psychology.* New York: Holt, 1959.
[9] D. E. Troxel, "Comparison of tactile and visual reading rates," M.I.T. Electronics Research Lab., Cambridge, Mass. Quart. Progress Rept. 67, pp. 267–272, October 1962.
[10] R. W. Donaldson, "Approximate formulas for the information transmitted by a discrete communication channel," M.I.T. Electronics Research Lab., Cambridge, Mass. Quart. Progress Rept. 77, pp. 335–342, April 1965.
[11] A. de Saint-Exupery, *The Little Prince.* New York: Harcourt, Brace, and World, 1943.
[12] F. A. Geldard, *Cutaneous Channels of Communication*, W. A. Rosenblith, Ed. New York: Wiley 1961.

Nearest Neighbor Pattern Classification

T. M. COVER, MEMBER, IEEE, AND P. E. HART, MEMBER, IEEE

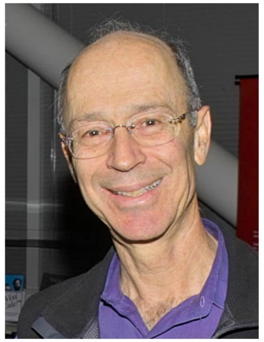
Abstract—The nearest neighbor decision rule assigns to an unclassified sample point the classification of the nearest of a set of previously classified points. This rule is independent of the underlying joint distribution on the sample points and their classifications, and hence the probability of error R of such a rule must be at least as great as the Bayes probability of error R^*—the minimum probability of error over all decision rules taking underlying probability structure into account. However, in a large sample analysis, we will show in the M-category case that $R^* \leq R \leq R^*(2 - MR^*/(M-1))$, where these bounds are the tightest possible, for all suitably smooth underlying distributions. Thus for any number of categories, the probability of error of the nearest neighbor rule is bounded above by twice the Bayes probability of error. In this sense, it may be said that half the classification information in an infinite sample set is contained in the nearest neighbor.

I. INTRODUCTION

IN THE CLASSIFICATION problem there are two extremes of knowledge which the statistician may possess. Either he may have complete statistical knowledge of the underlying joint distribution of the observation x and the true category θ, or he may have no knowledge of the underlying distribution except that which can be inferred from samples. In the first extreme, a standard Bayes analysis will yield an optimal decision procedure and the corresponding minimum (Bayes) probability of error of classification R^*. In the other extreme, a decision to classify x into category θ is allowed to depend only on a collection of n correctly classified samples $(x_1, \theta_1), (x_2, \theta_2), \cdots, (x_n, \theta_n)$, and the decision procedure is by no means clear. This problem is in the domain of nonparametric statistics and no optimal classification procedure exists with respect to all underlying statistics.

If it is assumed that the classified samples (x_i, θ_i) are independently identically distributed according to the distribution of (x, θ), certain heuristic arguments may be made about good decision procedures. For example, it is reasonable to assume that observations which are close together (in some appropriate metric) will have the same classification, or at least will have almost the same posterior probability distributions on their respective classifications. Thus to classify the unknown sample x we may wish to weight the evidence of the nearby x_i's most heavily. Perhaps the simplest nonparametric decision procedure of this form is the *nearest neighbor* (NN) rule, which classifies x in the category of its nearest neighbor. Surprisingly, it will be shown that, in the large sample case, this simple rule has a probability of error which

Manuscript received February 23, 1966; revised April 29, 1966. This work has been supported at Stanford University by U. S. Army Electronics Command under Contract DA28-043-AMC-01764(E) and by USAF under Contract AF49(638)1517; and at the Stanford Research Institute, Menlo Park, Calif., by RADC under Contract AF30(602)-3945.
T. M. Cover is with the Department of Electrical Engineering, Stanford University, Stanford, Calif.
P. E. Hart is with the Stanford Research Institute, Menlo Park, Calif.



Thomas Cover (bottom) and Peter Hart (top)

History of machine learning

- **1990:** Boosting algorithm
  - No single strong model is proposed
  - Aims to enhance predicting power
    - Combine the predictions of many weak models
      - Using averages or voting

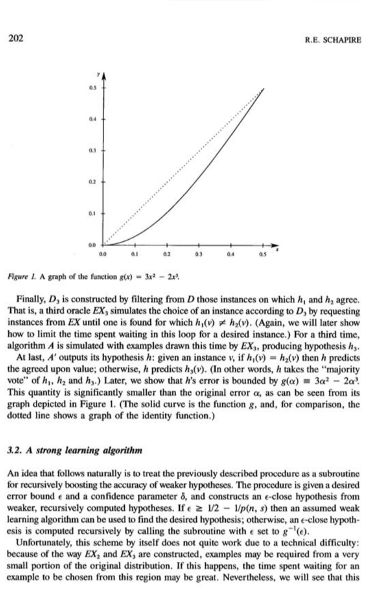


Figure 1. A graph of the function $g(x) = 3x^2 - 2x^3$.

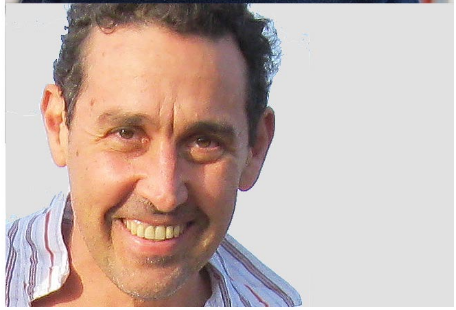
Finally, D_3 is constructed by filtering from D those instances on which h_1 and h_2 agree. That is, a third oracle EX_3 simulates the choice of an instance according to D_3 by requesting instances from EX until one is found for which $h_1(v) \neq h_2(v)$. (Again, we will later show how to limit the time spent waiting in this loop for a desired instance.) For a third time, algorithm A is simulated with examples drawn this time by EX_3, producing hypothesis h_3.

At last, A' outputs its hypothesis h: given an instance v, if $h_1(v) = h_2(v)$ then h predicts the agreed upon value; otherwise, h predicts $h_3(v)$. (In other words, h takes the "majority vote" of h_1, h_2 and h_3.) Later, we show that h's error is bounded by $g(\alpha) \equiv 3\alpha^2 - 2\alpha^3$. This quantity is significantly smaller than the original error α, as can be seen from its graph depicted in Figure 1. (The solid curve is the function g, and, for comparison, the dotted line shows a graph of the identity function.)

3.2. A strong learning algorithm

An idea that follows naturally is to treat the previously described procedure as a subroutine for recursively boosting the accuracy of weaker hypotheses. The procedure is given a desired error bound ϵ and a confidence parameter δ, and constructs an ϵ-close hypothesis from weaker, recursively computed hypotheses. If $\epsilon \geq 1/2 - 1/p(n, s)$ then an assumed weak learning algorithm can be used to find the desired hypothesis; otherwise, an ϵ-close hypothesis is computed recursively by calling the subroutine with ϵ set to $g^{-1}(\epsilon)$.

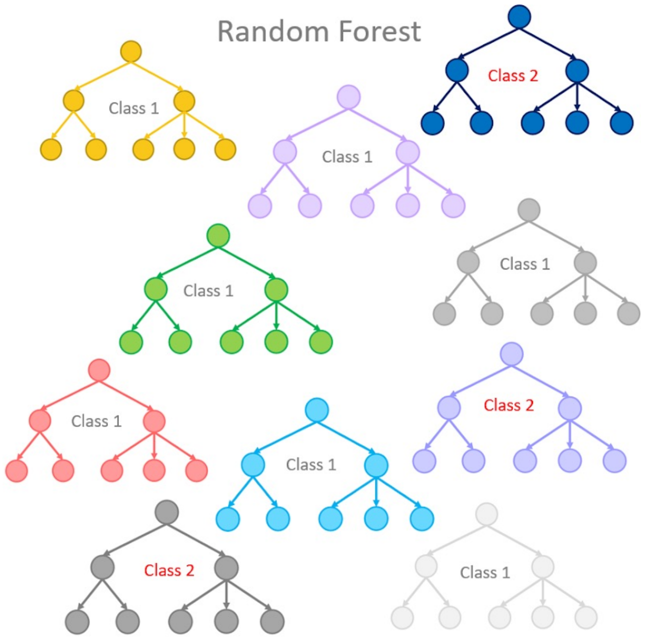
Unfortunately, this scheme by itself does not quite work due to a technical difficulty: because of the way EX_2 and EX_3 are constructed, examples may be required from a very small portion of the original distribution. If this happens, the time spent waiting for an example to be chosen from this region may be great. Nevertheless, we will see that this

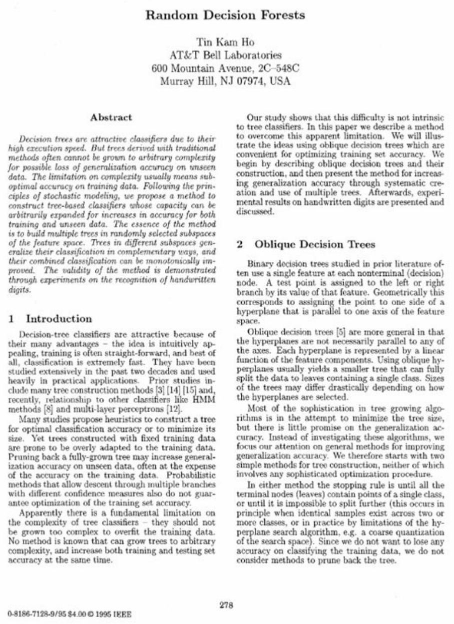


The Strength of Weak Learnability.
Robert Schapire (top) and Yoav Freund (bottom)

History of machine learning

- **1995: Random decision forests**
  - Creates and merges decisions from individual tree structures into a "forest”
  - Significantly improves its accuracy and decision-making





Random Decision Forests

Tin Kam Ho
AT&T Bell Laboratories
600 Mountain Avenue, 2C–548C
Murray Hill, NJ 07974, USA

Abstract

Decision trees are attractive classifiers due to their high execution speed. But trees derived with traditional methods often cannot be grown to arbitrary complexity for possible loss of generalization accuracy on unseen data. The limitation on complexity usually means suboptimal accuracy on training data. Following the principles of stochastic modeling, we propose a method to construct tree-based classifiers whose capacity can be arbitrarily expanded for increases in accuracy for both training and unseen data. The essence of the method is to build multiple trees in randomly selected subspaces of the feature space. Trees in different subspaces generalize their classification in complementary ways, and their combined classification can be monotonically improved. The validity of the method is demonstrated through experiments on the recognition of handwritten digits.

1 Introduction

Decision-tree classifiers are attractive because of their many advantages – the idea is intuitively appealing, training is often straight-forward, and best of all, classification is extremely fast. They have been studied extensively in the past two decades and used heavily in practical applications. Prior studies include many tree construction methods [3] [14] [15] and, recently, relationship to other classifiers like HMM methods [8] and multi-layer perceptrons [12].

Many studies propose heuristics to construct a tree for optimal classification accuracy or to minimize its size. Yet trees constructed with fixed training data are prone to be overly adapted to the training data. Pruning back a fully-grown tree may increase generalization accuracy on unseen data, often at the expense of the accuracy on the training data. Probabilistic methods that allow descent through multiple branches with different confidence measures also do not guarantee optimization of the training set accuracy.

Apparently there is a fundamental limitation on the complexity of tree classifiers – they should not be grown too complex to overfit the training data. No method is known that can grow trees to arbitrary complexity, and increase both training and testing set accuracy at the same time.

Our study shows that this difficulty is not intrinsic to tree classifiers. In this paper we describe a method to overcome this apparent limitation. We will illustrate the ideas using oblique decision trees which are convenient for optimizing training set accuracy. We begin by describing oblique decision trees and their construction, and then present the method for increasing generalization accuracy through systematic creation and use of multiple trees. Afterwards, experimental results on handwritten digits are presented and discussed.

2 Oblique Decision Trees

Binary decision trees studied in prior literature often use a single feature at each nonterminal (decision) node. A test point is assigned to the left or right branch by its value of that feature. Geometrically this corresponds to assigning the point to one side of a hyperplane that is parallel to one axis of the feature space.

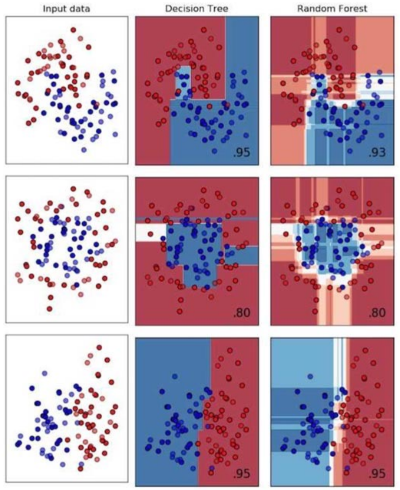
Oblique decision trees [5] are more general in that the hyperplanes are not necessarily parallel to any of the axes. Each hyperplane is represented by a linear function of the feature components. Using oblique hyperplanes usually yields a smaller tree that can fully split the data to leaves containing a single class. Sizes of the trees may differ drastically depending on how the hyperplanes are selected.

Most of the sophistication in tree growing algorithms is in the attempt to minimize the tree size, but there is little promise on the generalization accuracy. Instead of investigating these algorithms, we focus our attention on general methods for improving generalization accuracy. We therefore starts with two simple methods for tree construction, neither of which involves any sophisticated optimization procedure.

In either method the stopping rule is until all the terminal nodes (leaves) contain points of a single class, or until it is impossible to split further (this occurs in principle when identical samples exist across two or more classes, or in practice by limitations of the hyperplane search algorithm, e.g. a coarse quantization of the search space). Since we do not want to lose any accuracy on classifying the training data, we do not consider methods to prune back the tree.

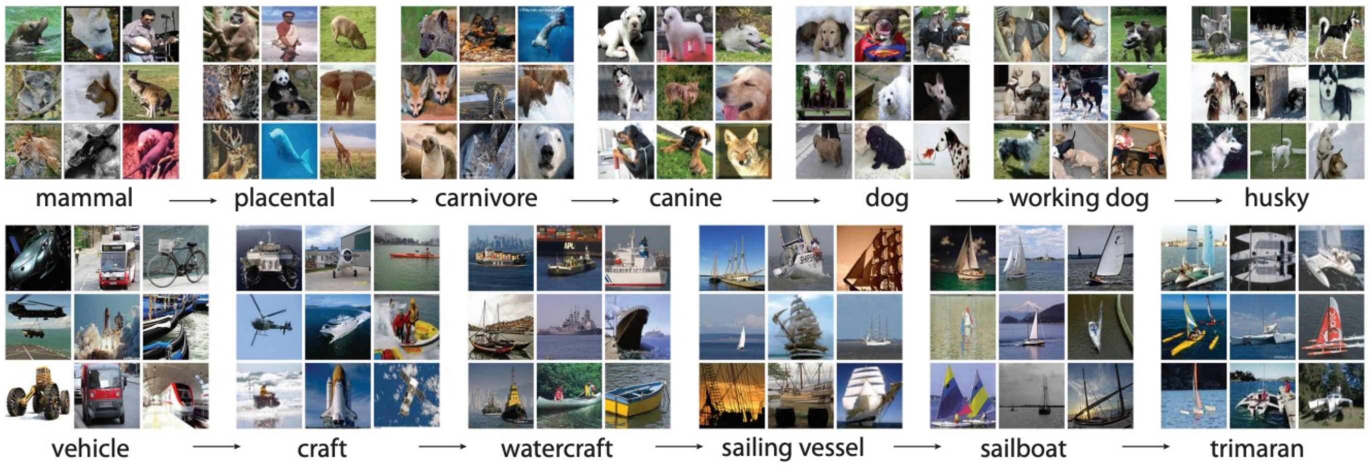
278

0-8186-7128-9/95 $4.00 © 1995 IEEE

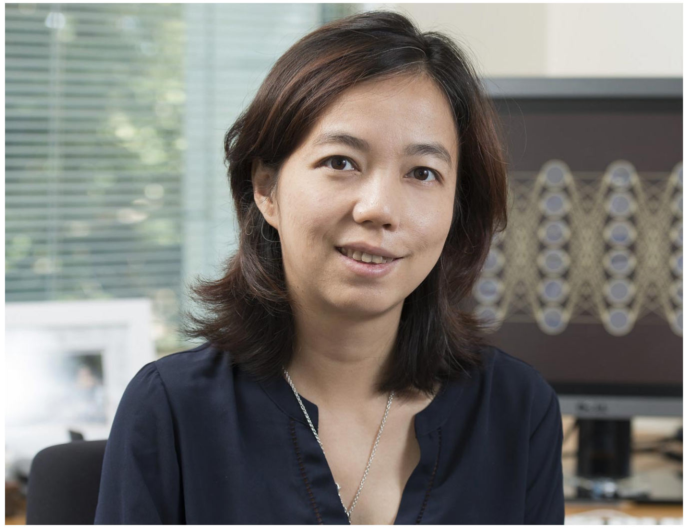


History of machine learning

- 2009: ImageNet Large Scale Visual Recognition Challenge
  - > 14 millions of manually annotated images
  - 1000 object categories



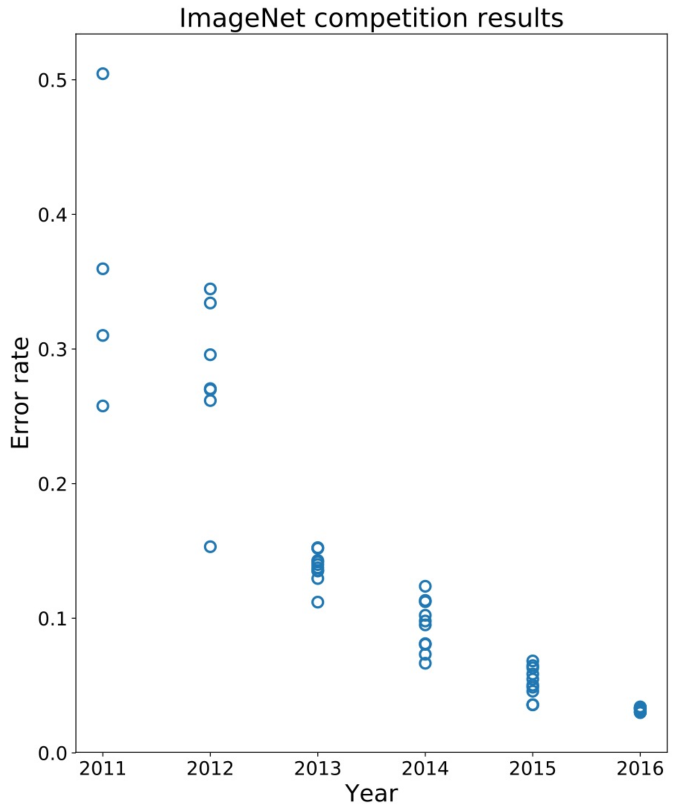
Imagenet: A large-scale hierarchical image database



Fei-Fei Li

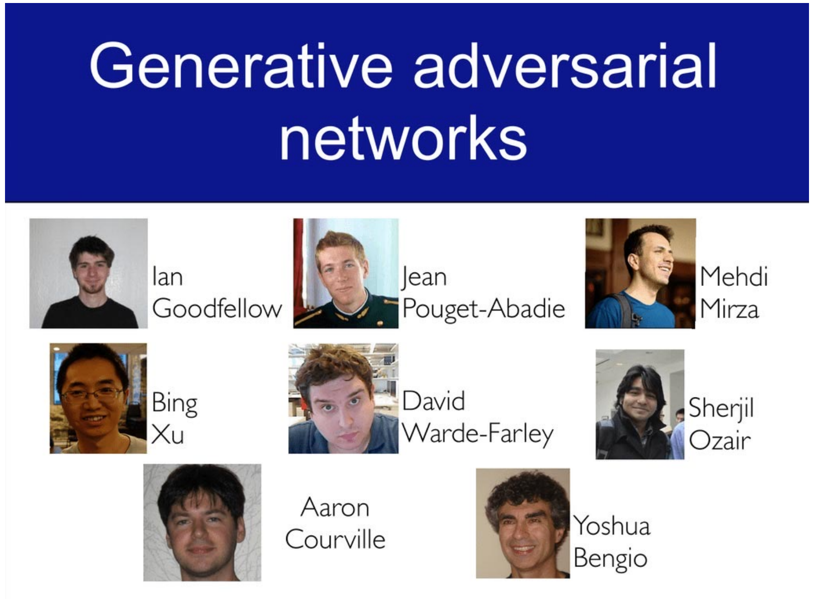
History of machine learning

- 2009: ImageNet Large Scale Visual Recognition Challenge
  - > 14 millions of manually annotated images
  - 1000 object categories
  - Error rate: 25% (2011), 16% (2012, AlexNet) …
  - The start of a "deep learning revolution"



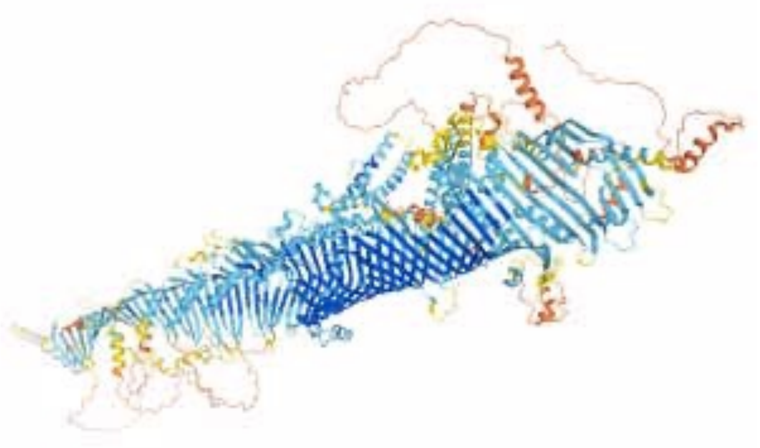
History of machine learning

- Generative adversarial networks (GAN)
  - Teaches AI how to generate new data based on training set
  - Two network opposing each other
    - Generator vs Discriminator



History of machine learning

- 2015: DeepMind's AlphaGo
  - The first AI to beat a professional Go player
- 2017: Waymo launches autonomous taxis
- 2021: DeepMind's AlphaFold
  - Reveals human protein structures



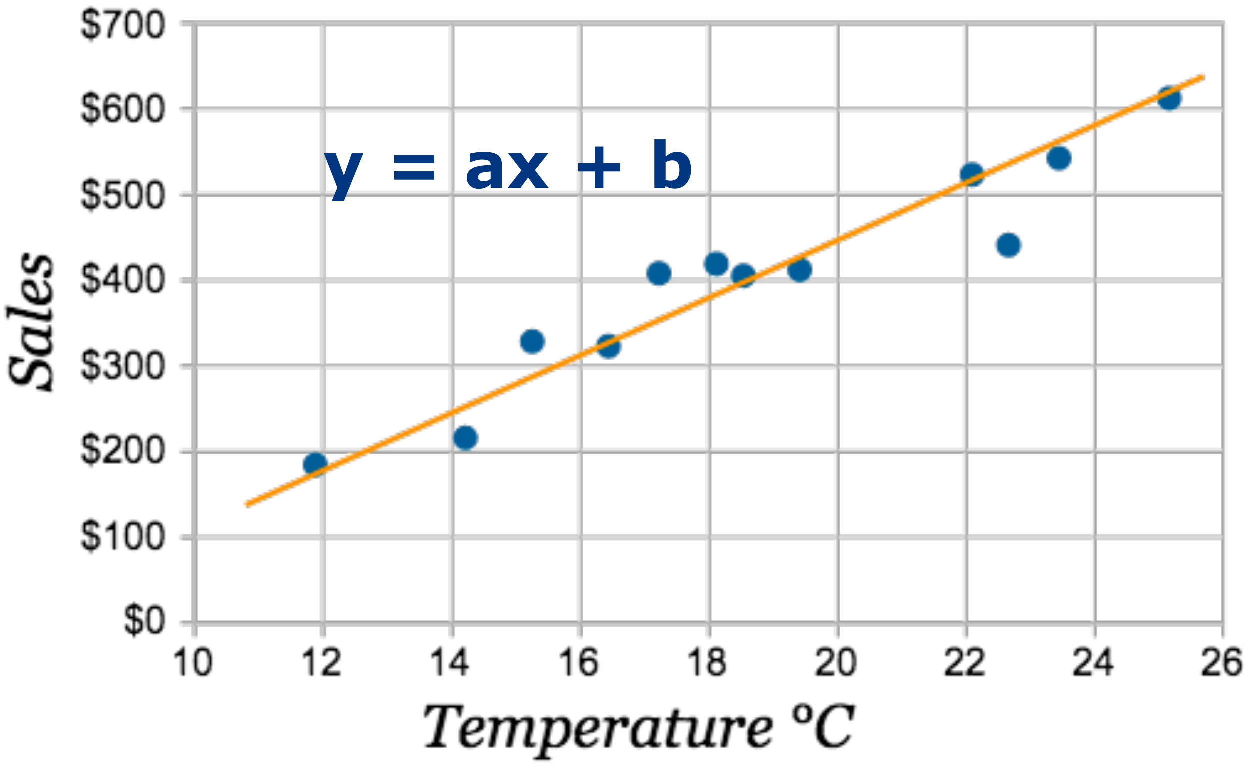
Machine learning in this course

- Different types of machine learning
  - Supervised learning
  - Unsupervised learning
  - ~~Semi-supervised learning~~
  - ~~Reinforcement learning~~

Supervised learning

- Learn from both labeled inputs and desired outputs
  - Almost all applications of deep learning that are in the spotlight these days belong in this category: optical character recognition, speech recognition, image classification/segmentation, object detection, and language translation
- Task driven: requires desired input and output data
- Good at
  - **Regression:** map input variables to a continuous function and predict values
    - Given sizes (and energy labels, ages, distance to city center) of houses, predict their price
    - Given a picture of a person, predict his/her age
  - **Classification**: map input variables into discrete categories
    - Given a patient with a tumor, predict whether the tumor is malignant or benign

Example of regression



Example of classification

Supervised learning

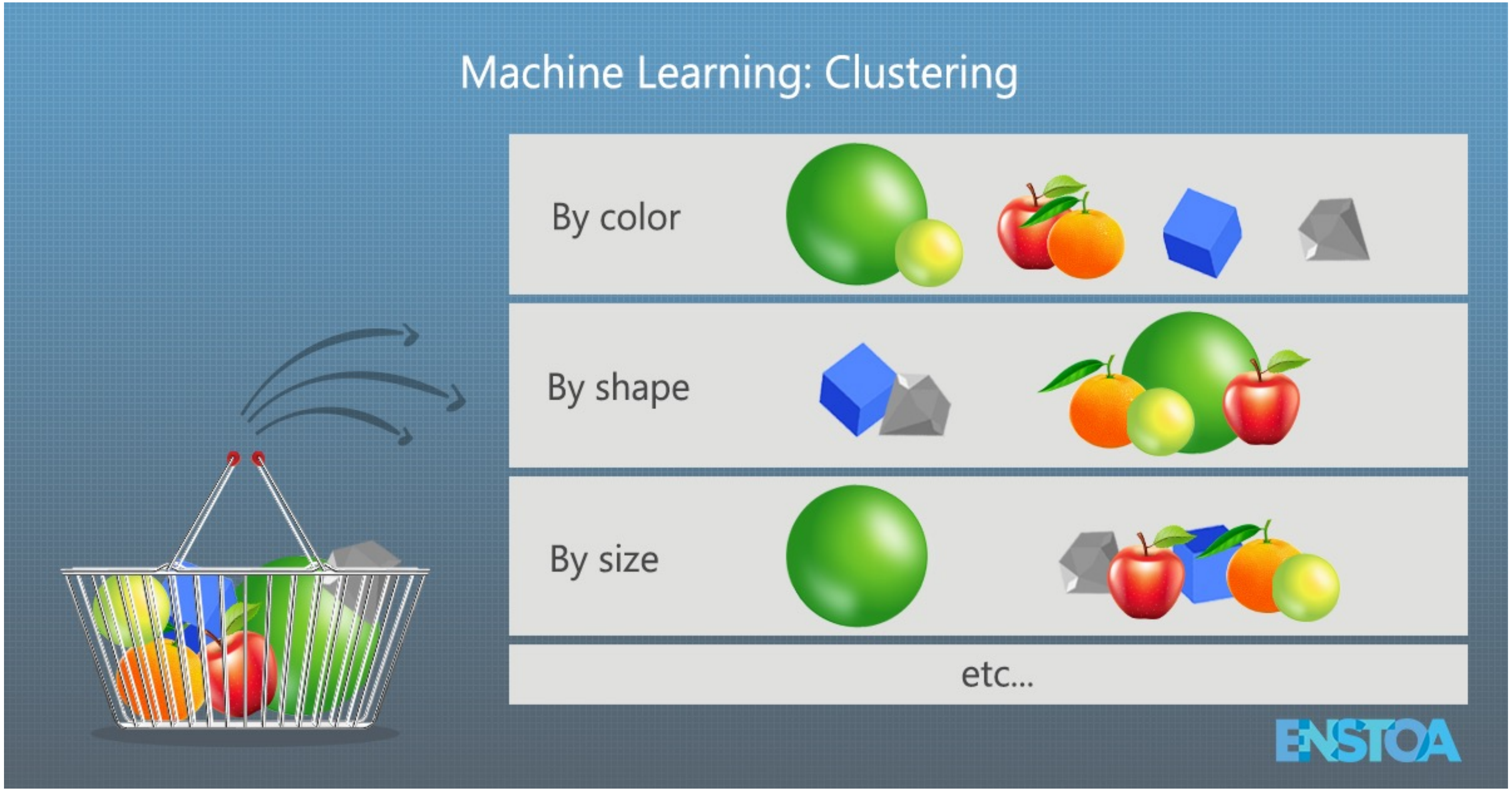
- Exercise 1: Which is regression, and which is classification?
  - Problem 1: Use a learning algorithm to predict tomorrow's temperature (in degrees Centigrade/Fahrenheit)
  - Problem 2: Examine the statistics of two football teams and predict which team will win tomorrow's match (given historical data of teams' wins/losses)

- Exercise 2: Turn the following regression problem into a classification problem
  - Given sizes (and energy labels, ages, distance to city center) of houses, predict their price

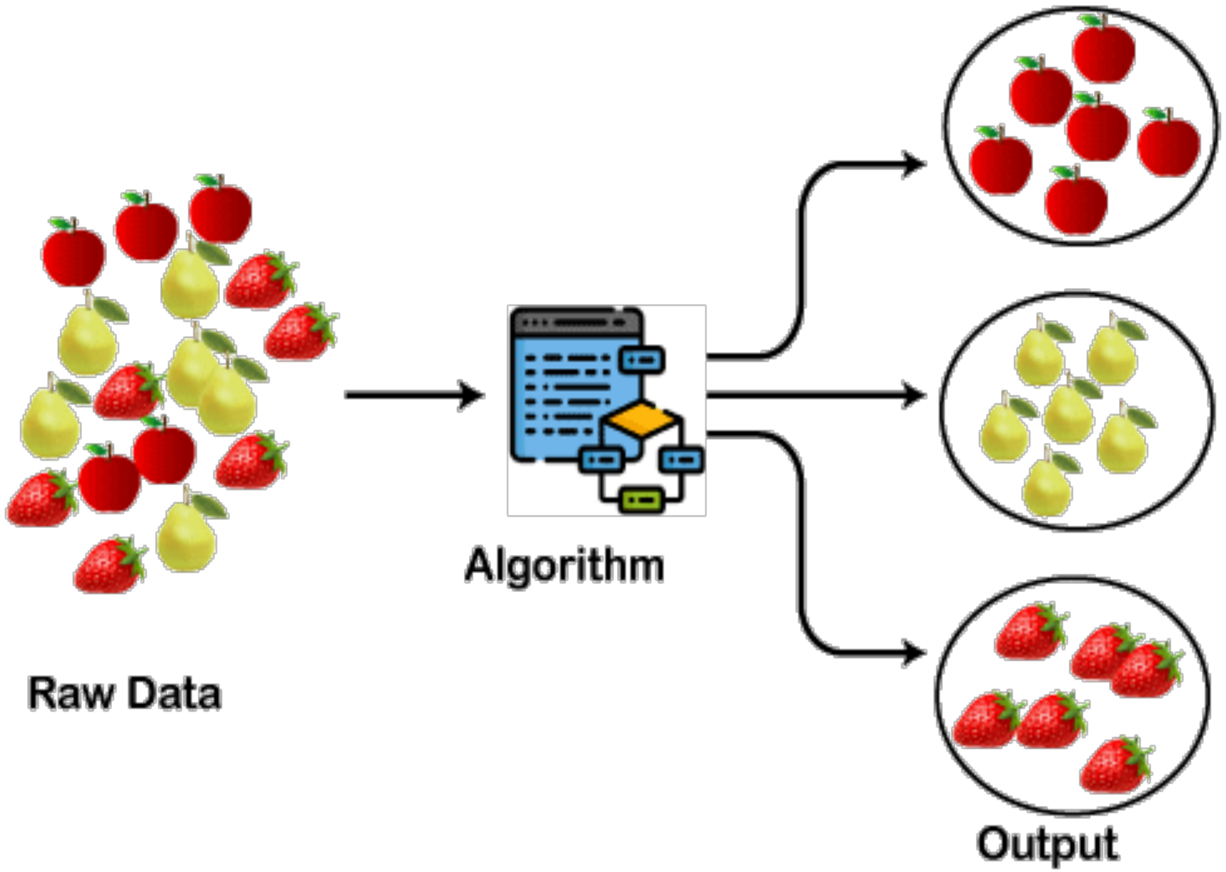
Unsupervised learning

- Train on unlabeled data to look for meaningful connection
  - Approach problems with little or no idea what our results should look like
  - Often a necessary step in better understanding a dataset before attempting to solve a supervised-learning problem
- Data driven: not trained with desired outcomes in mind
- Good at
  - Clustering: Splitting the dataset into groups based on similarity, without knowing what each group represents
    - Take a collection of 1M different genes and group these genes into groups that are somehow similar or related by different variables, such as lifespan, location, roles.
  - Anomaly detection: identifying rare items, events or observations
    - Automatic video surveillance for theft detection in ATM machines

Different features for clustering



Clustering vs classification



Clustering or classification?

Semi-supervised learning

- Mix of supervised and unsupervised learning
  - Training data might be provided, but the model is free to explore the data on it own and develop its own understanding of the dataset
  - Why: performance usually improves when trained on labeled datasets, but labeling data can be time consuming and expensive
  - Strikes a middle ground between the performance of supervised learning and the efficiency of unsupervised learning
- Good at
  - Machine translation: teaching algorithms to translate language based on less than a full dictionary of words
  - Fraud detection: identifying cases of fraud when you only have a few examples
  - Labelling data: algorithms trained on small data sets can learn to apply data labels to larger sets automatically

Reinforcement learning

- Teach a machine to complete a multi-step process with defined rules
  - Positive or negative cues are given
  - The algorithm decides on its own what steps to take to maximize reward
- Good at
  - Robotics: robots can learn to perform tasks
  - Video gameplay: to teach bots to play a number of video games
    - Example: DeepMind’s AlphoGo
  - Resource management: Given finite resources and a defined goal, help enterprises plan out how to allocate resources
- Mostly a research area and no significant successes beyond games

The advantages of using machine learning

- Near-human-level image classification
- Near-human-level speech recognition
- Near-human-level handwriting transcription
- Near-human-level autonomous driving
- Improvement in many tasks
  - Machine translation,
  - Text-to-speech conversion
  - Ad targeting
  - Search on the web
  - …

Limitation and danger of using ML

- Machine learning lacks common sense
    - AI is far from the cognitive level of cats

With only 800 million neurons, the cat's brain is far ahead of any giant artificial neural network.

Limitation and danger of using ML

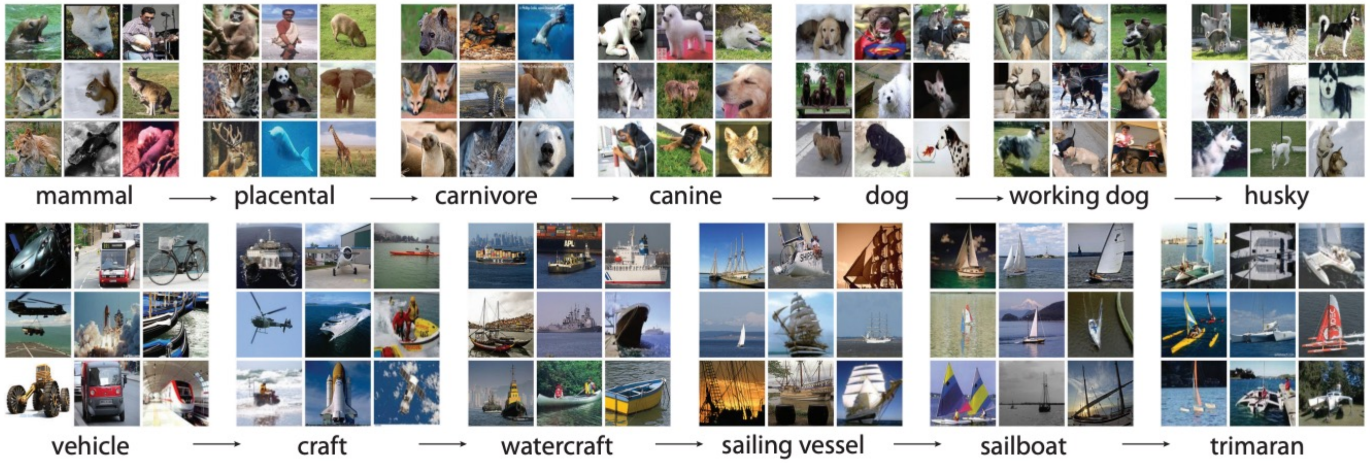
- Generalization issue/Data biases
  - Applying a model trained on one dataset may not work well one other datasets
  - Perform well on benchmarked datasets, but can fail badly on real world images outside the dataset
  - Dataset does not reflect the realities of the environment
    - E.g., facial recognition systems trained primarily on images of white men
    - E.g., breast cancer prediction algorithms primarily trained on X-rays of white women

Fact: almost all big datasets, generated by systems powered by ML/AI based models, are known to be biased.

Limitation and danger of using ML

- Lack of data & lack of good data
  - Many machine learning algorithms require large amounts of data before they begin to give useful results
    - fewer data -> poor results
    - poor quality annotation -> poor results

Caltech 101 dataset



Imagenet: A large-scale hierarchical image database

Limitation and danger of using ML

- Lack of data & lack of good data
  - Many machine learning algorithms require large amounts of data before they begin to give useful results
    - fewer data -> poor results
    - poor quality annotation -> poor results

Sensitive to view changes
(Faster-RCNN [0.1, 1.0])

Limitation and danger of using ML

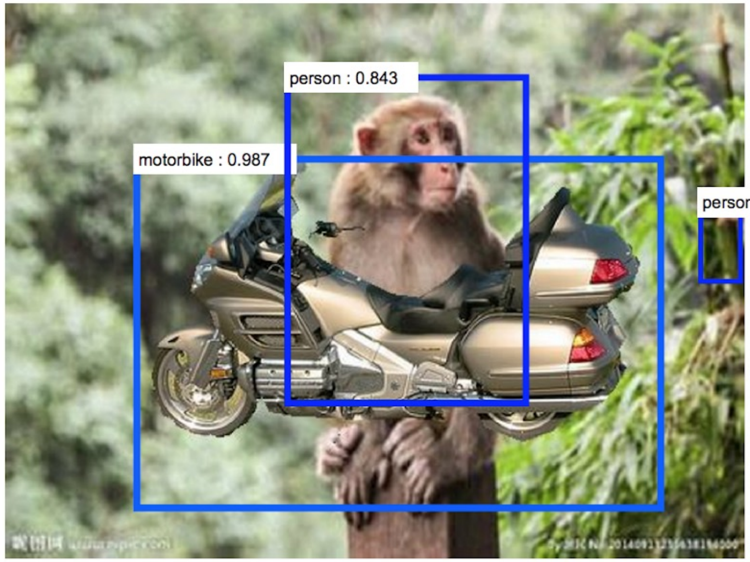
- Lack of data & lack of good data
    - Many machine learning algorithms require large amounts of data before they begin to give useful results
        - fewer data -> poor results
        - poor quality annotation -> poor results
- Reusing data is bad
- Data augmentation is useful to some extent
- Having more good data is almost always the preferred solution

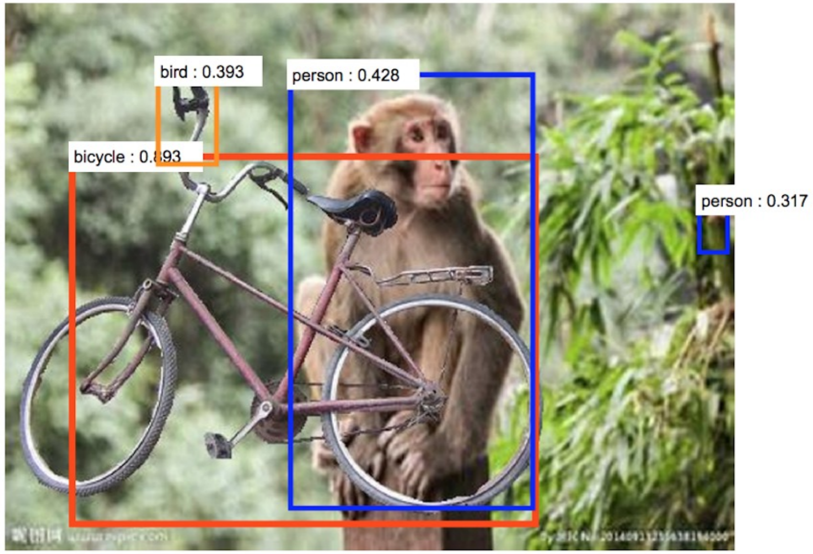
Limitation and danger of using ML

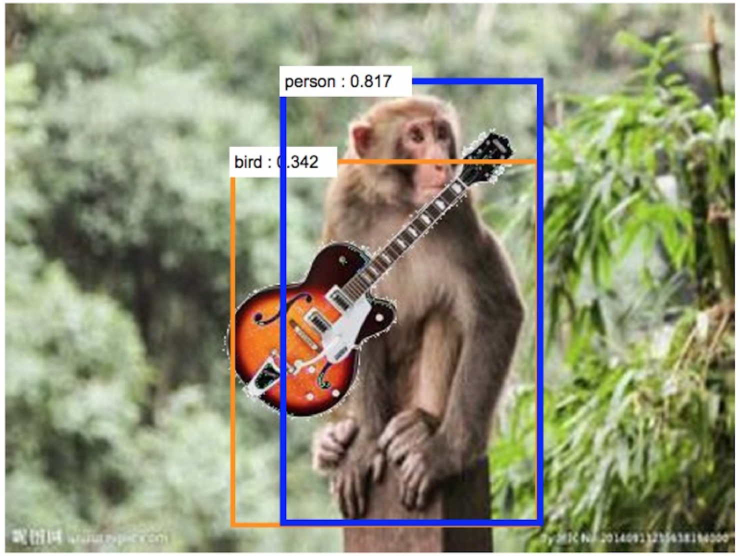
- Machine learning is stochastic, not deterministic
  - You can never assert that a result is 100% correct.
  - Example 1: weather forecast
    - Computationally expensive, may take weeks or longer
    - Replace simulation by machine learning?
  - Example 2: medical care
    - Error or inaccuracy may cause patient injury
      - recommend wrong drug
      - fail to notice a tumor

Limitation and danger of using ML

- Sensitive to changes in context





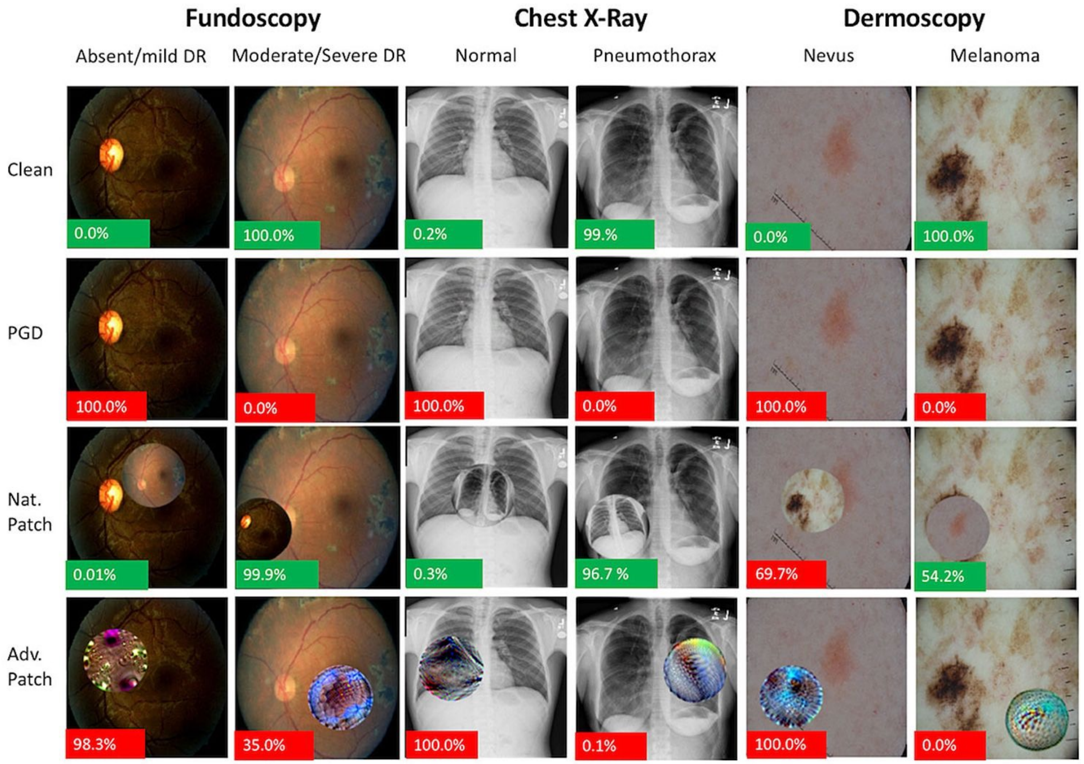


Photoshopping a guitar into a picture of a monkey in the jungle confuses deep nets

Limitation and danger of using ML

- Susceptibility to adversarial attacks
  - To find limitations: test ML learning systems with "adversarial examples"
  - Models susceptible to manipulation by inputs explicitly designed to fool them

Example:
- Introducing small amounts of noise (imperceptible to human) fools an ML system classifying medical images
- The noise could also be incorporated directly into the image-capture process
- Someone who has access to the data could commit different kinds of fraud, not just using adversarial attacks
- Very difficult to detect if the attack has occurred



Charles Choi. Medical Imaging AI Software Is Vulnerable to Covert Attacks. IEEE Spectrum. 04 Jun 2018

Limitation and danger of using ML

- Ethics
  - Trust algorithms and data more than our own judgment and logic
  - Who do we blame if an algorithm is wrong?
    - Example: failures in medical care
    - Example: accidents by autonomous driving cars

This course

- Machine learning
  - Introductory level
    - Basic theories & commonly used algorithms
      - Clustering, linear regression, Bayesian classification, logistic regression, SVM, decision trees, random forest, neural networks, deep learning …
    - Practical techniques
      - Data collection, data processing, features, parameter tuning, etc.
  - Hands-on experiences
  - (Focused on) processing geo-spatial data

Learning objectives

- Understand and explain the impact, limits, and dangers of machine learning; give use cases of machine learning for the built environment;
- Explain the main concepts in machine learning (e.g., regression, classification, unsupervised learning, supervised learning, overfitting, training, validation, cross-validation, and regularization);
- Explain the principles of commonly used unsupervised and supervised machine learning techniques (e.g., clustering, linear regression, logistic regression, SVM, random forest, and neural networks);
- Collect and pre-process data (e.g., labelling, normalization, feature selection, augmentation, train-test splitting) for applying machine learning techniques;
- Select and apply the appropriate machine learning method for a specific geospatial data processing task (e.g., object classification and semantic segmentation);
- Analyze and evaluate the performance of machine learning models.

Next Lecture

- Clustering
- k-nearest neighbor classification

GEO5017

- The teachers
- Learning activities
- Assessment
- Communication

Learning activities

- Lectures
  - 2 x 45min per week (Friday mornings)
  - Lecture room
- Lab exercises (and work on assignments)
  - 2 sessions (2 x 45min each) per week (Tuesday and Friday afternoons)
  - Geolab or online
  - Teachers available

Learning activities

- Lectures
  - 1. Introduction to machine learning [Liangliang]
  - 2 & 3 Unsupervised learning [Liangliang]
  - 4 & 5 Nearest neighbor classification & Linear regression [Liangliang]
  - 6 & 7 Bayesian classification & logistic regression [Shenglan]
  - 8 & 9 Support vector machine (SVM) [Shenglan]
  - 10 & 11 Decision trees and random forest [Shenglan]
  - 12 & 13 Neural networks [Nail]
  - 14 & 15 Deep learning [Nail]

Assessment

- 2 (or 3) group assignments (40 %)
    - Group performance
    - Personal contribution/Peer reviews
- Final exam (60%):
    - Lectures, handouts, assignments
        - Multiple-choice questions
        - Open questions

Assessment

- 2 (or 3) group assignments (40 %)
  - Group performance
  - Personal contribution/Peer reviews
- Final exam (60%):
  - Lectures, handouts, assignments
    - Multiple-choice questions
    - Open questions
- Pass?
  - Assignments >= 5.5
  - Exam >= 5.5
  - Total of 6.0 or above

Assessment

- Assignments
  - Each assignment released after the lecture
  - Programming: implementation and experiment with ML algorithm(s)
  - Work in groups (ideally 3 students per group)

Assessment

- Assignments
  - Each assignment released after the lecture
  - Programming: implementation and experiment with ML algorithm(s)
  - Work in groups (ideally 3 students per group)
  - What to submit
    - Report
      - <= 3 pages (excluding figures, tables, references)
      - Individual contribution

Isaac Newton (75 %)

- Compared the reconstruction results from method [1] and method [2];
- Implemented the function reorient_normals ();
- Came up with a novel reconstruction method and implemented it in function reconstruct();
- Wrote the "Methodology" section of the report.

Albert Einstein (20 %)

- Preparing and pre-processing of the point clouds, i.e., taking photos, run SfM and MVS, cropping the buildings from the messy point clouds, and normal estimation;
- Wrote the "Implementation Details" section of the report.

Thomas Edison (5 %)

- Wrote the "Abstract" section of the report.

Assessment

- Assignments
  - Each assignment released after the lecture
  - Programming: implementation and experiment with ML algorithm(s)
  - Work in groups (ideally 3 students per group)
  - What to submit
    - Report
    - Code
      - Collaboration using GitHub
      - [optional] Include the link to the GitHub repository in the report
      - Reproduce the results
        - **Doesn’t compile:** -10%
        - **Doesn’t reproduce the result:** -10%

Assessment

- Assignments
  - Each assignment released after the lecture
  - Programming: implementation and experiment with ML algorithm(s)
  - Work in groups (ideally 3 students per group)
  - What to submit
  - We allow multiple submissions
    - Incorporating comments from teachers/peers
    - Evaluation based on 1st submission + 0.5 maximum

Example:

First submission 6, then final mark will be <= 6.5

Assessment

- Assignments
  - Each assignment released after the lecture
  - Programming: implementation and experiment with ML algorithm(s)
  - Work in groups (ideally 3 students per group)
  - What to submit
  - We allow multiple submissions
  - Strict deadline
    - Late submission
      - 10% deducted per day late
      - Not acceptable after 3 days late

Assessment

- Assignments
  - Each assignment released after the lecture
  - Programming: implementation and experiment with ML algorithm(s)
  - Work in groups (ideally 3 students per group)
  - What to submit
  - We allow multiple submissions
  - Strict deadline
  - Teamwork: **Everyone active in coding/discussion/reporting**
    - **We strongly discourage**
      - report writing to one person and code writing to another
      - one person working on course A and another on course B
      - Perfectly equal individual contributions

Assessment

- Assignments
  - Copy from others/internet
    - Code
    - Sentences
    - Figures
    - …
  - Submit to BrightSpace [plagiarism check turned on]

Assessment

- Assignments
  - Not designed to challenge or test you, but
    - to help students gain knowledge
    - To help teacher to gain insights into students' progress and help you

Forget the mark

Ask questions

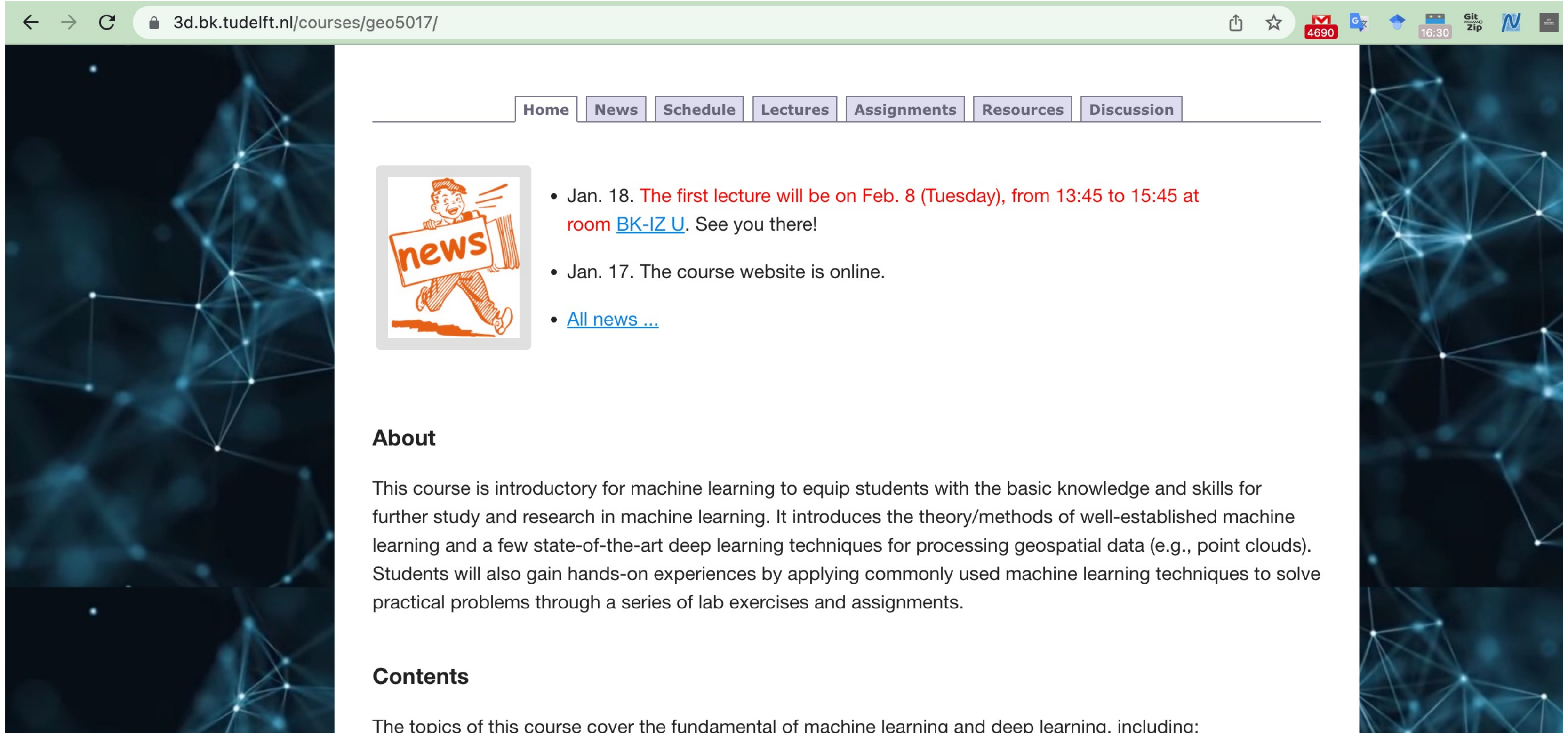
Enjoy the process!!!

Assessment

- Assignments
- Final exam
    - Lectures, handouts, assignments
        - Multiple-choice questions
        - Open questions
    - Example questions available before the exam

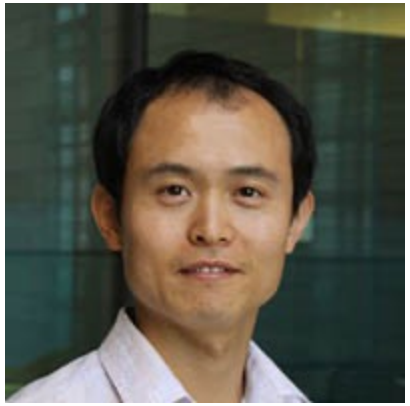
Communication

- Course website
  - https://3d.bk.tudelft.nl/courses/geo5017/



Communication

- Discussion
    - Lab/Lecture hours
    - Discord channel
    - Contact the teachers



Liangliang Nan
LiangliangNan#0976

Nail Ibrahimli
nibrahimli#5857

Shenglan Du
Shenglan Du#2136

Communication

- **Find your teammates for the assignments**
  - 3 students per team
  - Click on following link and put your name and student ID

    https://docs.google.com/document/d/1WMPXgWD0_2F9oDSub1K-g6NdRKqIRyWj3sUFDCpfFSk/edit

- **Lectures**: online or offline?
- **Lab exercises**: online or offline?