

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



- **What do students/teachers expect?**
- **Introduction to machine learning**
 - What is machine learning
 - Applications of machine learning
 - The history of machine learning
 - Machine learning in this course
 - The pros and cons of using machine learning
- **Organization of GEO5017**
 - The teachers
 - Learning activities
 - Assessment
 - Communication

Learning objectives:

- explain the impact, limits, and dangers of machine learning;
- give use cases of machine learning for the built environment.



What do students expect?

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

Prerequisites - What do teachers expect?



- **Basic calculus and linear algebra**
 - Comfortable with matrix-vector operations
 - Familiar with taking derivatives and gradients.
- **Basic probability and statistics**
 - Know fundamental concepts such as probabilities, Gaussian distributions, mean, standard deviation, etc.
- **Proficiency in Python programming**
 - All assignments will be in Python (utilizing libraries like [Numpy](#))

Ready to follow GEO5017?

Self-assessment of math fundamentals

https://3d.bk.tudelft.nl/courses/geo5017/prerequisites/GEO5017_Math_self_assessment.pdf



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?

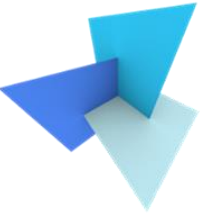


*A computer program is said to learn from experience E with respect to some task T and some performance measure P if its performance on T , as measured by P , improves with experience E . - **Tom Mitchell***

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

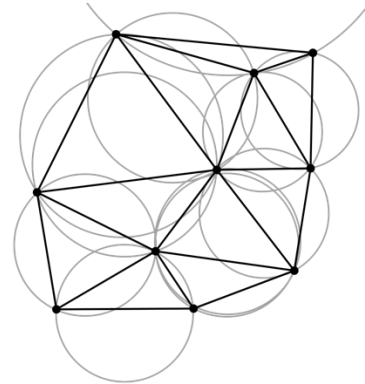
- A. The process of the algorithm examining a large amount of historical weather data.
- B. The weather prediction task.
- C. The probability of it correctly predicting a future date's weather.
- D. None of these.

What are machine learning algorithms?

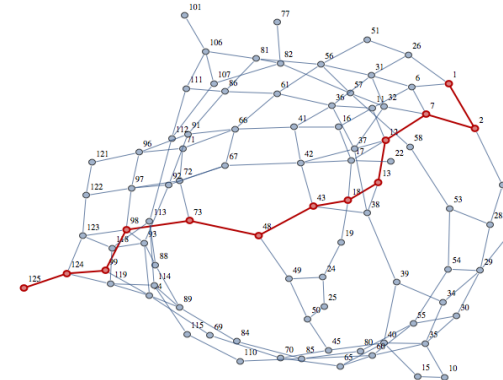


$$\begin{cases} 2x - 3y + z = -1 \\ x - y + 2z = -3 \\ 3x + y - z = 9 \end{cases}$$

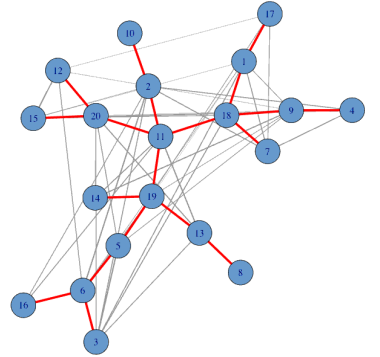
Equation solving



Delaunay triangulation



Shortest path



Minimum spanning tree



Face recognition



Autonomous driving



Spam filtering

Customers who bought this item also bought

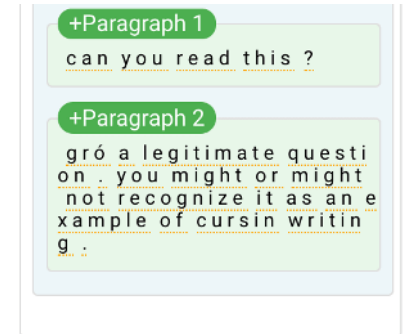
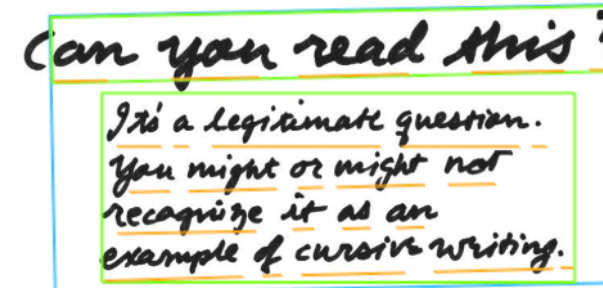


Recommender systems

Applications of machine learning



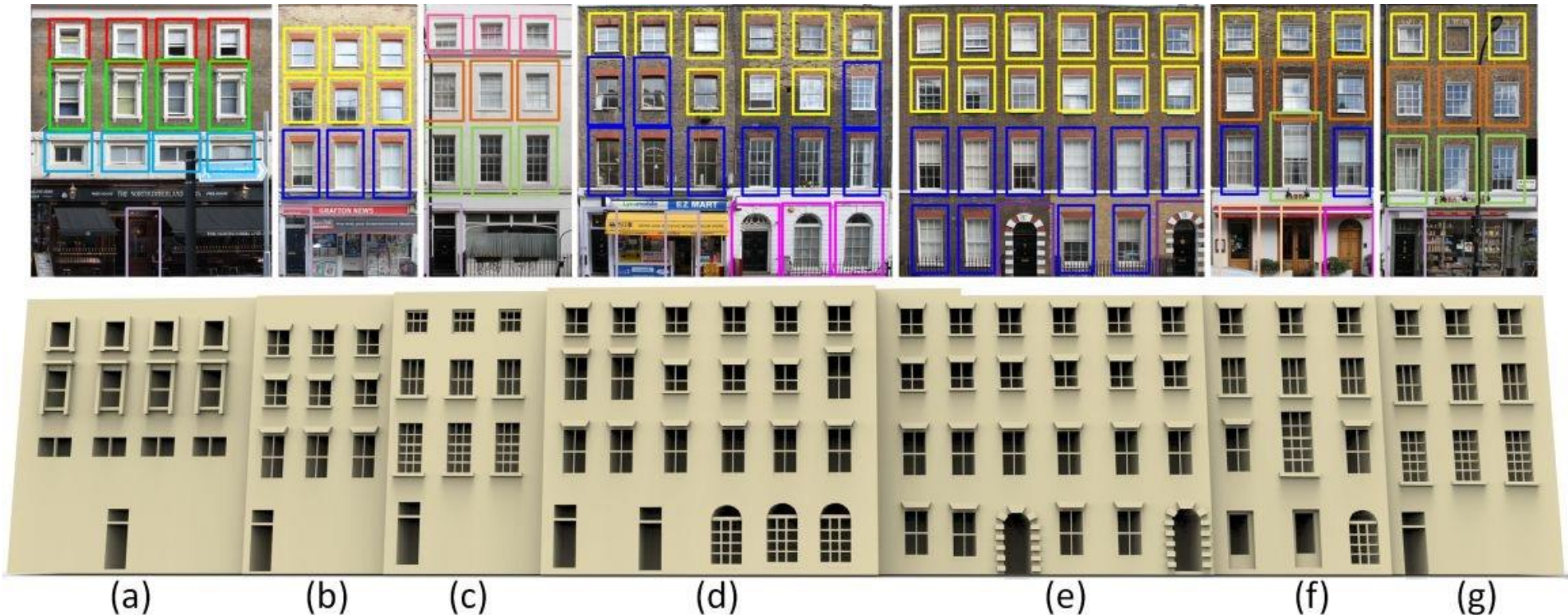
- Self-driving cars
- Face recognition
- Handwriting recognition
- Amazon product recommendation
- Spam filtering
- Automatic translation
- Speech recognition
- ...



Applications of machine learning



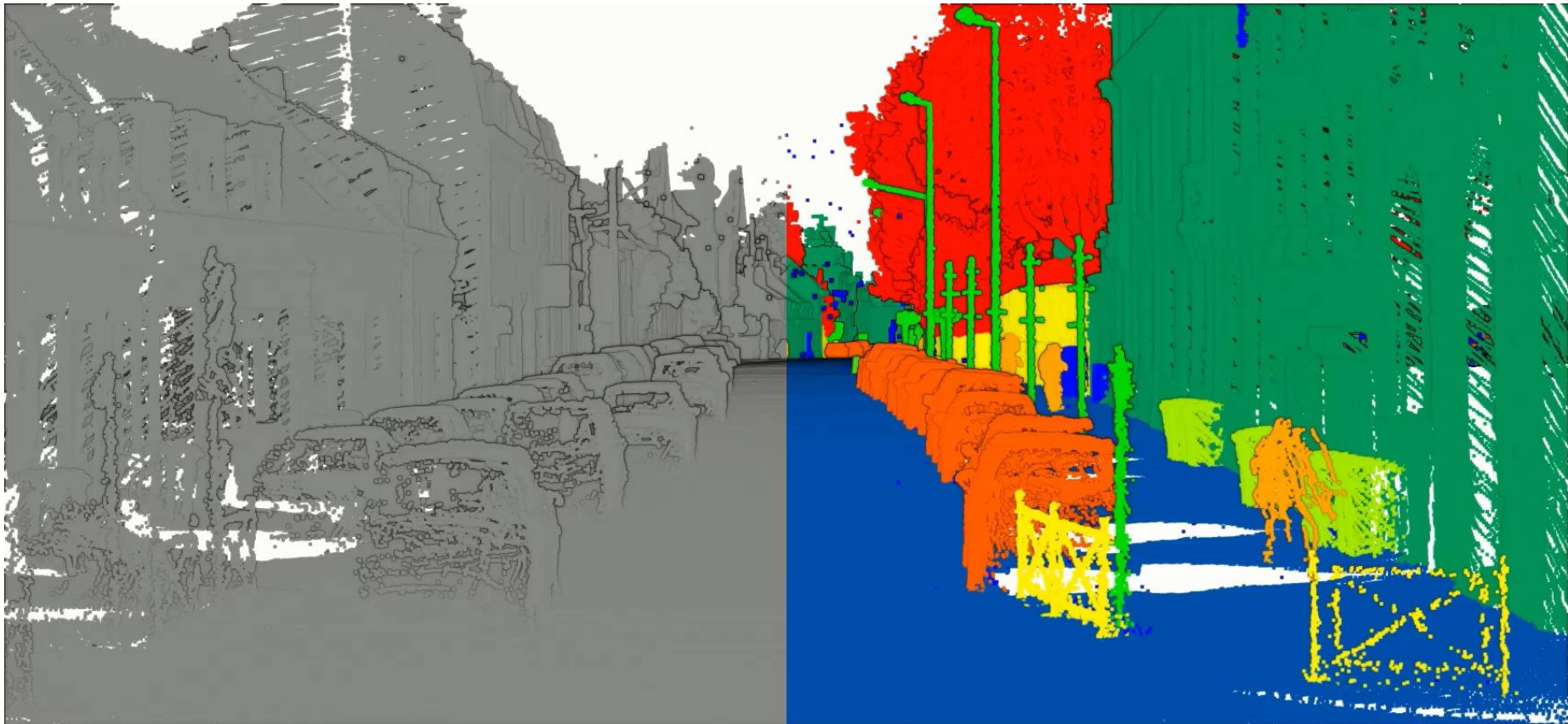
- Façade parsing and its applications in 3D modeling



Applications of machine learning



- Semantic segmentation





History of machine learning

(We only review a small subset of events and influential methods/techniques)

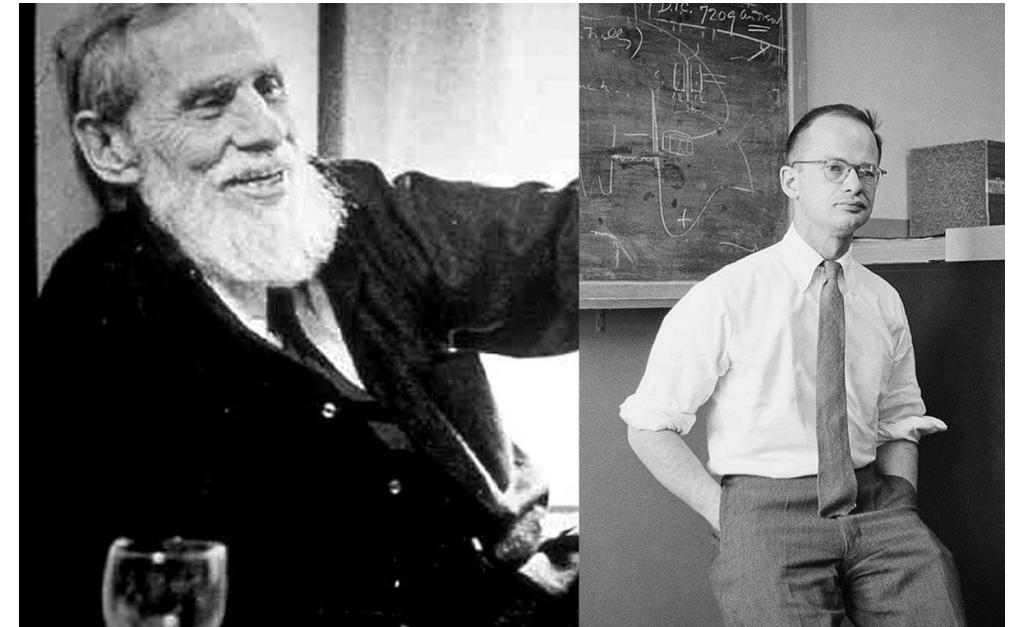
- 1943: First mathematical model of neural networks
 - Warren McCulloch (left) and Walter Pitts (right)

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 every possible path
 - Find optimal move by measuring chances of winning
 - Mechanisms to continuously improve
 - Remember previous moves
 - Compare with chances of winning

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

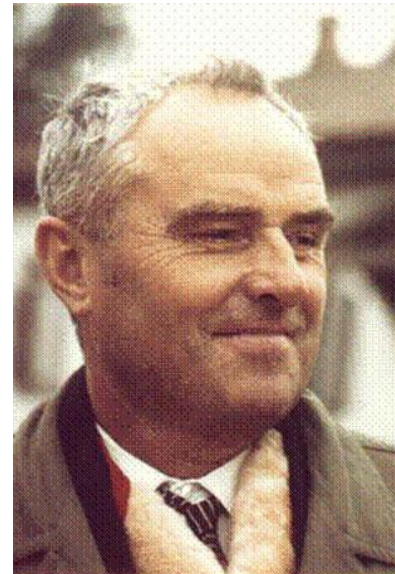


Arthur Samuel and IBM 700

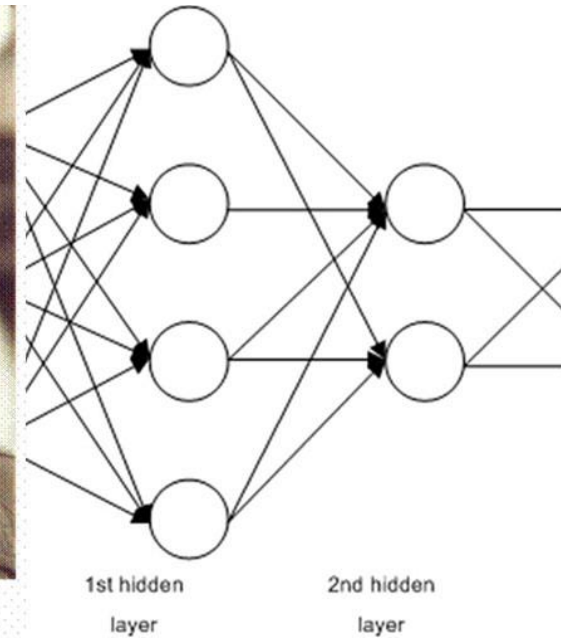
History of machine learning



- 1965: First Deep Neural Network
 - Foundations for nowadays' most powerful algorithms
 - First multi-layer perceptron
 - Alexey Ivakhnenko is considered the father of deep learning
- Not popular until around 2010
 - Limited computing power
 - Lack of annotated data



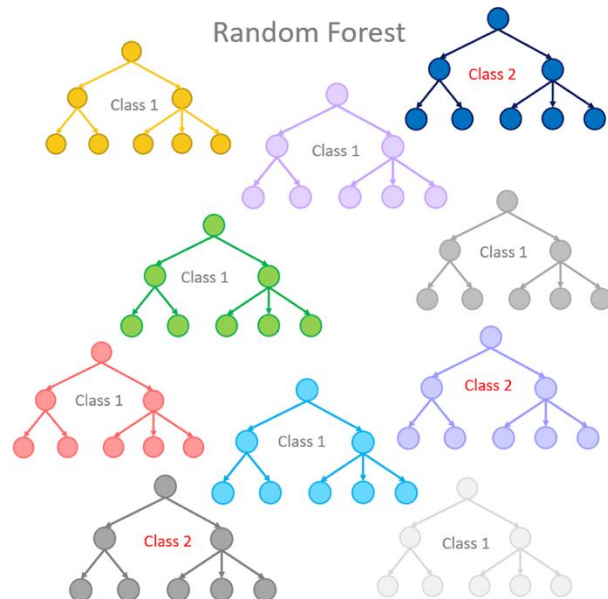
О.Г. Ивахненко (1967 p.)



History of machine learning



- 1995: Random decision forest
 - Creates and merges decisions from individual tree structures into a "forest"
 - The final prediction is based on **majority voting**
 - 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 sub-optimal 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 straightforward, 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 EDH methods [6] 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 forward through multiple heuristics 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.

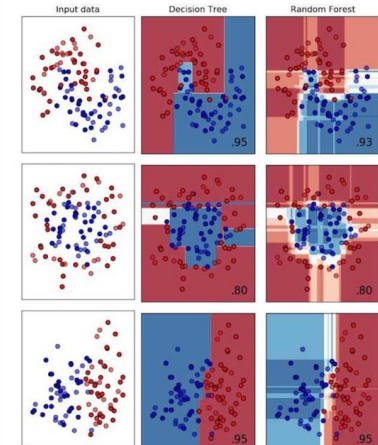
Our study shows that this difficulty is not intrinsic to tree classifiers. In this paper we describe a method to overcome this apparent limitation. We will illustrate the idea using oblique decision trees which are convenient for optimizing training set accuracy. We begin by describing oblique decision 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. Some of the trees may differ drastically depending on how the hyperplanes are selected.

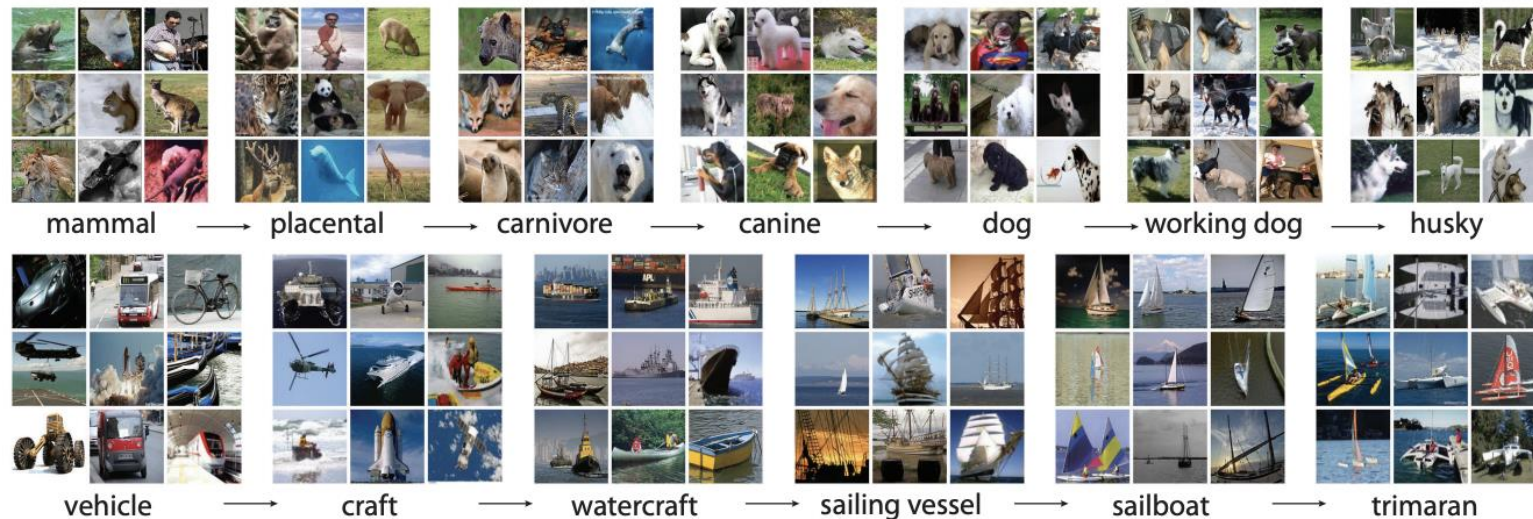
Most of the sophistication in tree growing algorithms is in the attempt to minimize the tree size, but there is little promise on the generalization accuracy. Instead of investigating these algorithms, we focus our attention on general methods for improving generalization accuracy. We therefore start 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 (the axes 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.



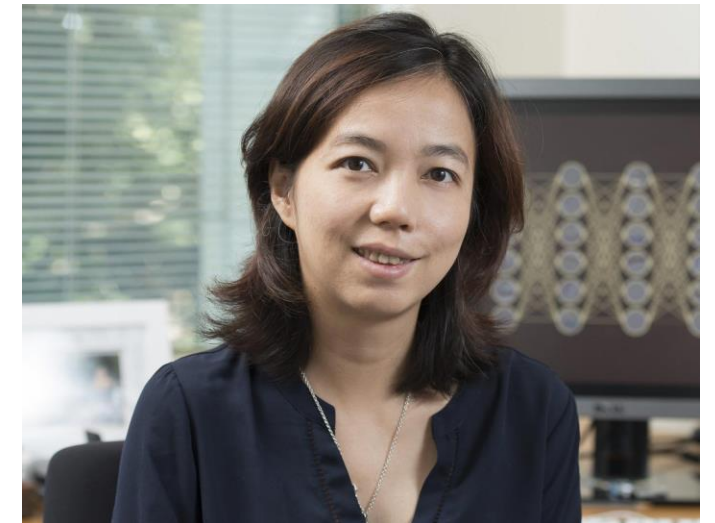


History of machine learning

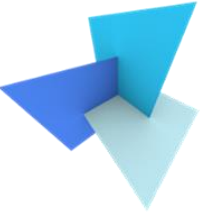
- 2009: ImageNet: dataset
 - 1K object categories
 - > 14 million manually annotated images
 - Crowdsourced annotation (otherwise 19 years by Li's teams)



[Imagenet: A large-scale hierarchical image database](#)



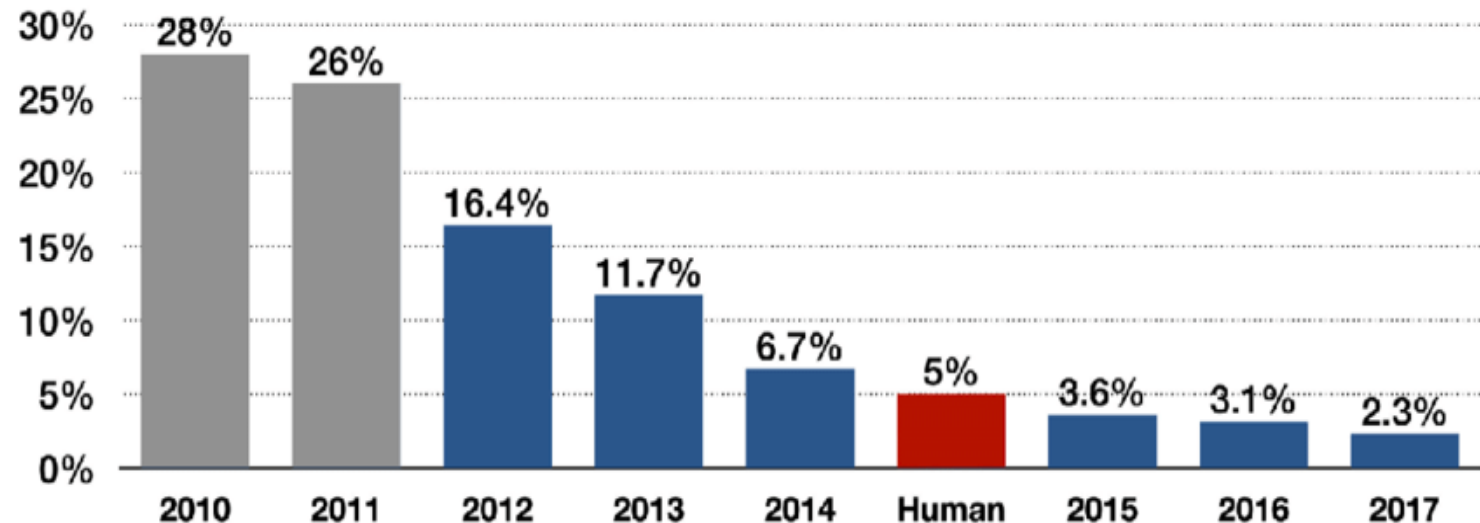
Fei-Fei Li



History of machine learning

- 2009: ImageNet: Large Scale Visual Recognition Challenge

- 1K object categories
 - > 14 million manually annotated images
 - Crowdsourced annotation
 - Error rate: 28% (2010), 16% (2012, AlexNet) ...
 - The start of a "deep learning revolution"
- Deep learning revolution
 - Transformed the AI industry





History of machine learning

- 2014: Generative adversarial networks (GAN)
 - Teaches AI how to generate new data based on training set
 - Two network opposing each other
 - Generator vs Discriminator

Generative adversarial networks



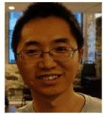
Ian
Goodfellow



Jean
Pouget-Abadie



Mehdi
Mirza



Bing
Xu



David
Warde-Farley



Sherjil
Ozair



Aaron
Courville



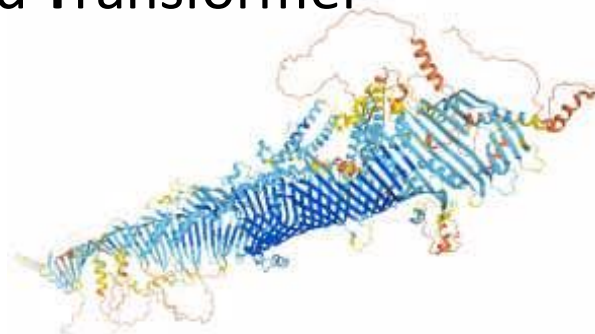
Yoshua
Bengio



History of machine learning



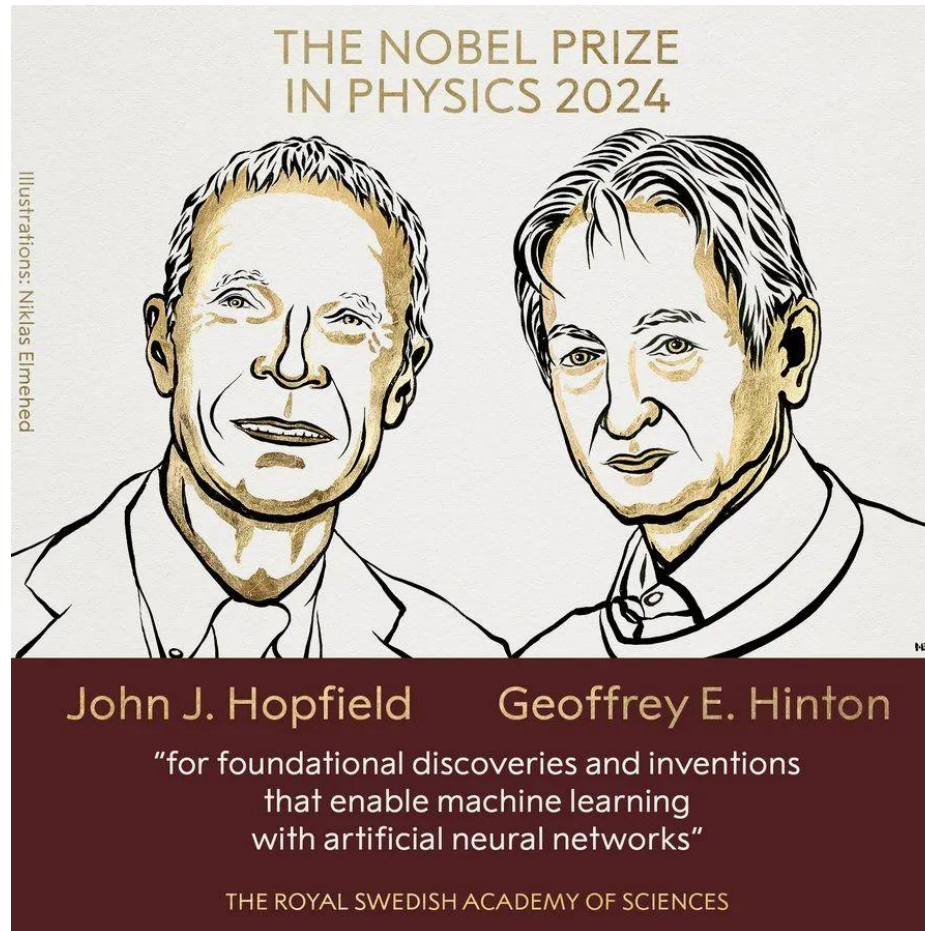
- 2015: DeepMind's AlphaGo
 - The first AI to beat a professional Go player
- 2017: Waymo launches autonomous taxis
- 2021: DeepMind's AlphaFold
 - Reveals human protein structures
- 2022: ChatGPT
 - **Generative Pre-trained Transformer**



History of machine learning



- 2024



Learning representations by back-propagating errors

David E. Rumelhart*, Geoffrey E. Hinton†
& Ronald J. Williams*

* Institute for Cognitive Science, C-015, University of California,
San Diego, La Jolla, California 92093, USA

† Department of Computer Science, Carnegie-Mellon University,
Pittsburgh, Philadelphia 15213, USA

We describe a new learning procedure, back-propagation, for
networks of neurone-like units. The procedure repeatedly adjusts

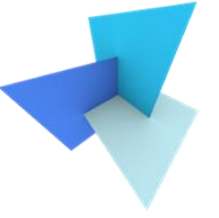


1986



Machine learning in this course

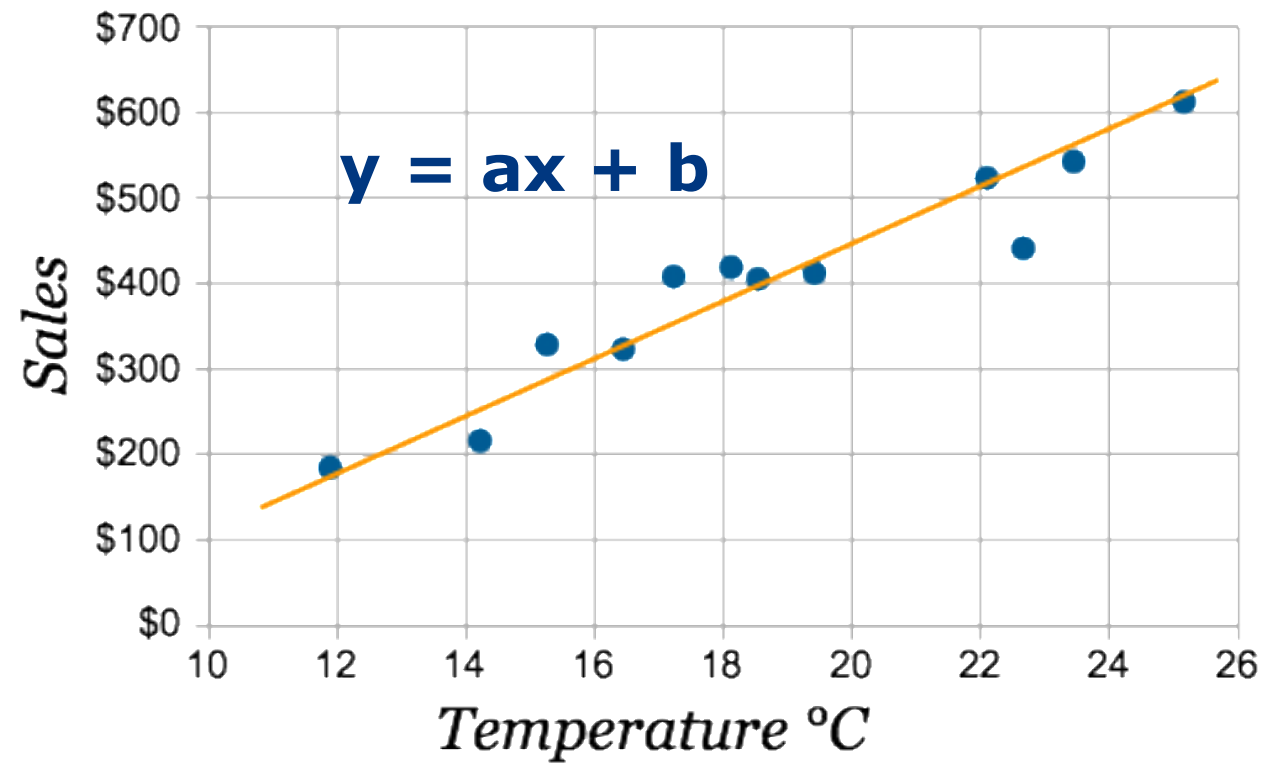
- Different types of machine learning
 - Supervised learning
 - Unsupervised learning
 - ~~○ Semi-supervised learning~~
 - ~~○ Reinforcement learning~~



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
- 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
 - Spam mail detection

Example of regression



Example of classification



Supervised learning



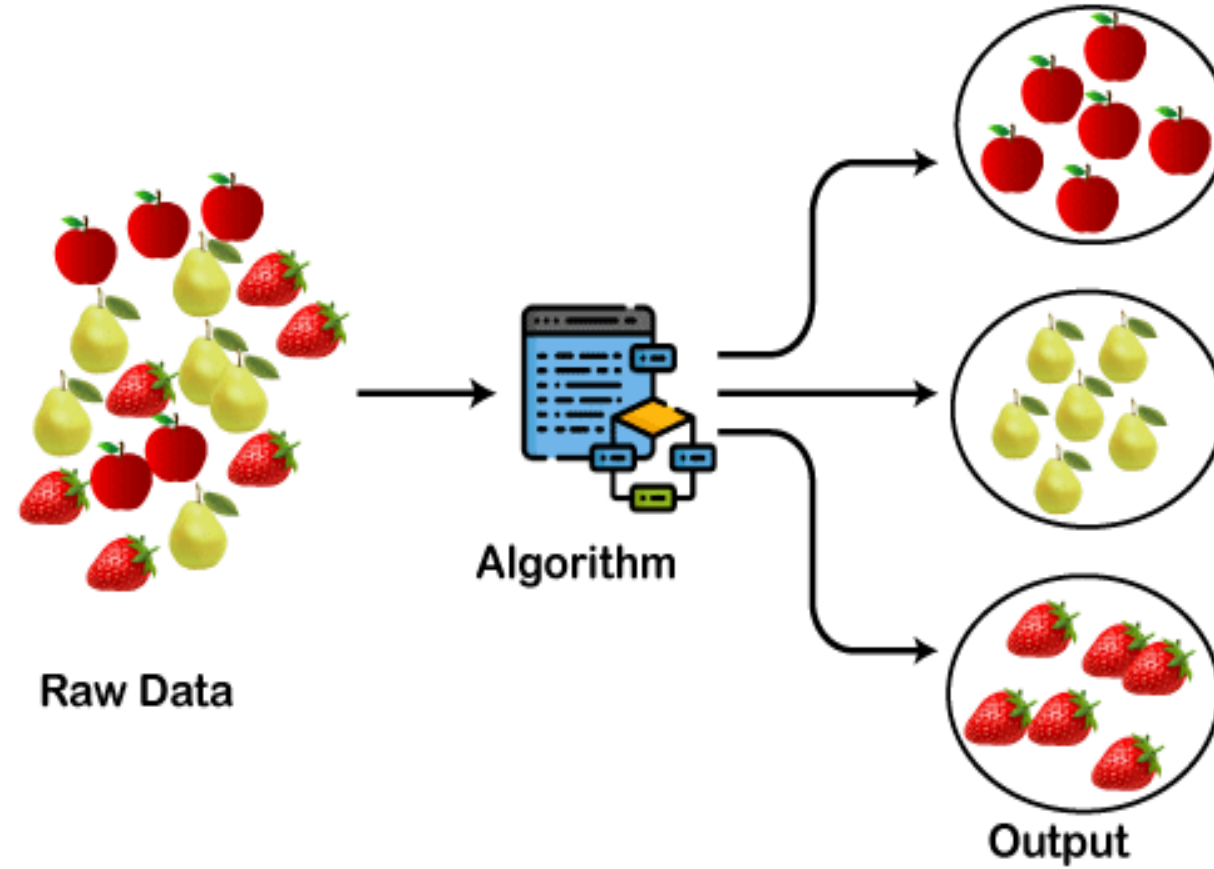
- Exercise 1: Which is regression, and which is classification?
 - **Problem 1:** Use a learning algorithm to predict tomorrow's temperature (in degrees Centigrade/Fahrenheit)
 - **Problem 2:** Given historical data of two football teams' wins/losses, examine the statistics of the two teams and predict which team will win tomorrow's match.
- Exercise 2: Turn the following regression problem into a classification problem
 - Given sizes, energy labels, ages, distance to city center of a house, predict its 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
- 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

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 its 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 AlphaGo
 - 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
 - Has potential in large range of real-world applications: self-driving cars, robotics, resource management, education



Limitation and danger of using ML

- Generalization issue/Data biases
 - A model trained on one dataset may not work well on other datasets
 - Dataset does not reflect the realities of the environment
 - E.g., facial recognition systems trained primarily on images of white men and women
 - 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
 - Require large amounts of data to give useful results
 - fewer data -> poor results
 - poor quality annotation -> poor results



Caltech 101 dataset

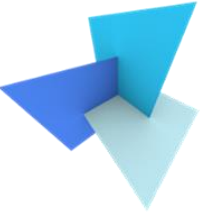


Limitation and danger of using ML

- Lack of data & lack of good data
 - Require large amounts of data to give useful results
 - fewer data -> poor results
 - poor quality annotation -> poor results



Biased against "rare events"



Limitation and danger of using ML

- Lack of data & lack of good data
 - Require large amounts of data 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



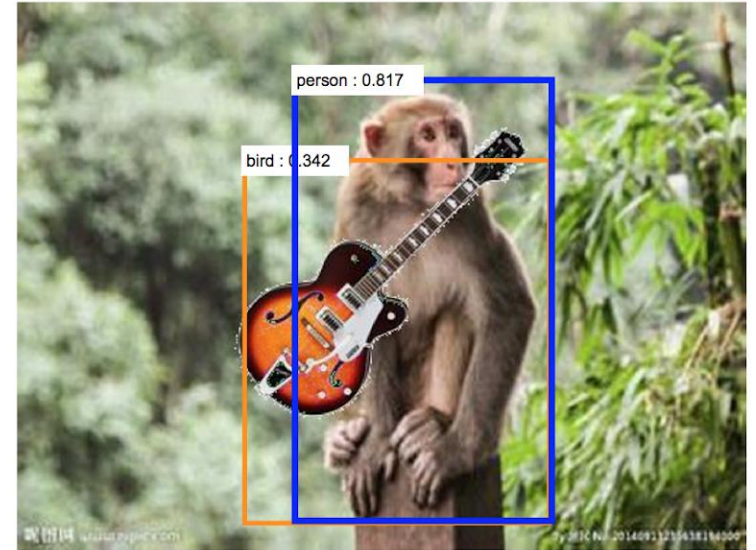
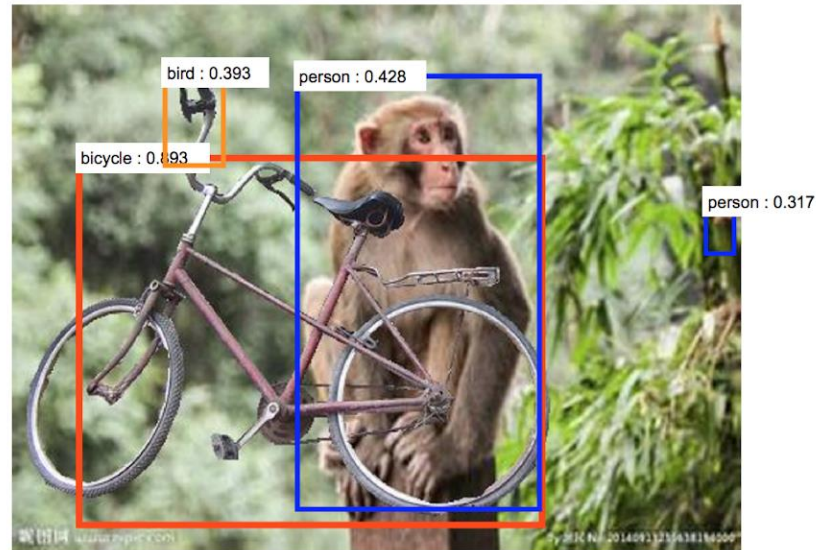
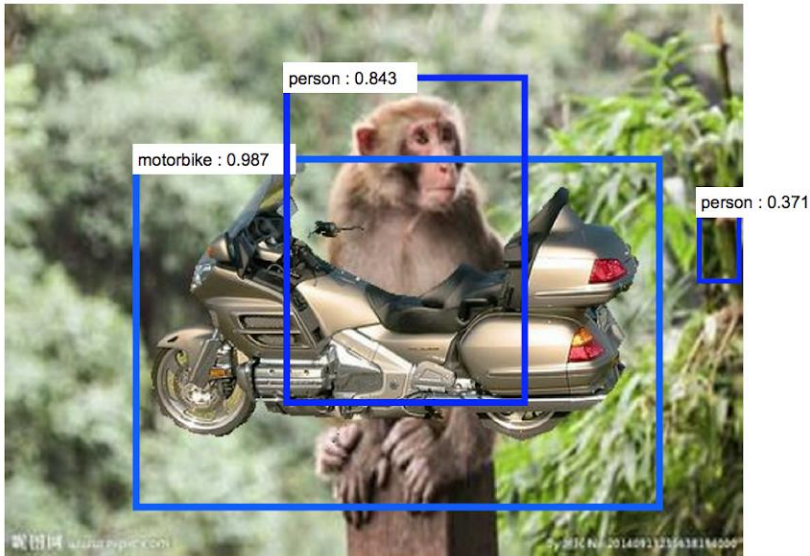
Limitation and danger of using ML

- 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

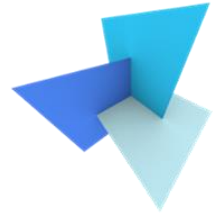


- Sensitive to changes in context



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

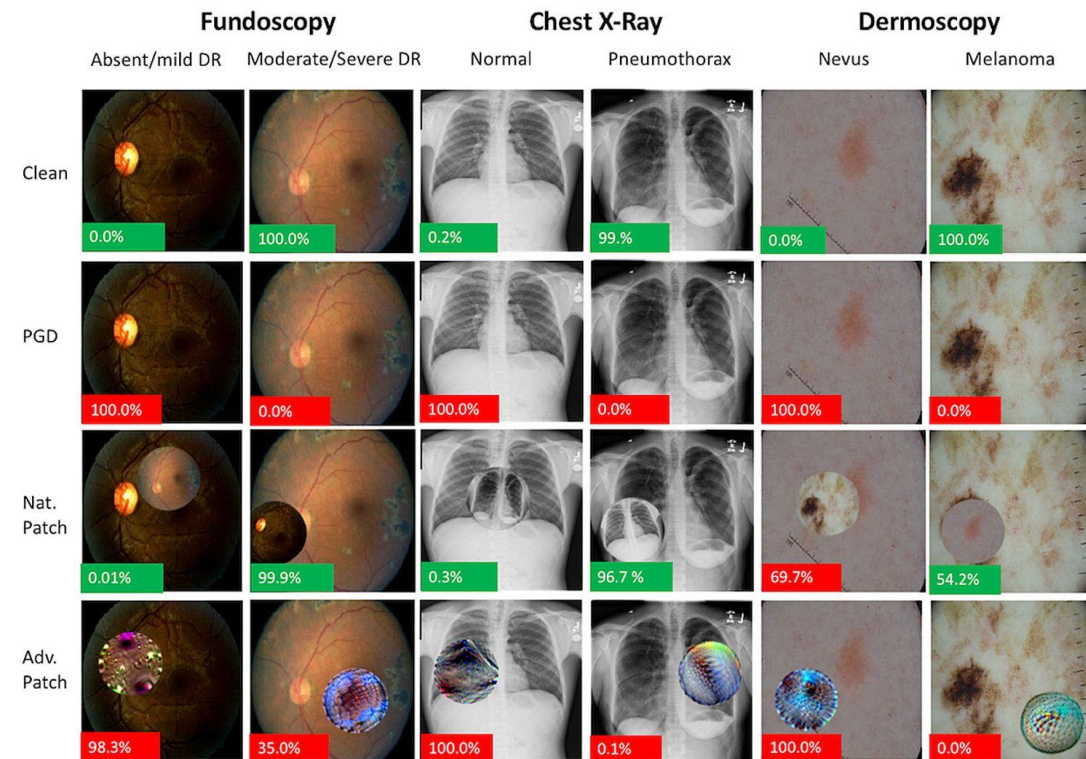
Limitation and danger of using ML



- 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
 - Linear regression, clustering, Bayesian classification, logistic regression, SVM, decision trees, random forest, neural networks, deep learning ...
 - Hands-on experiences
 - Data processing, feature crafting, feature selection, parameter tuning, etc.
 - Our focus: processing geo-spatial data

Learning objectives

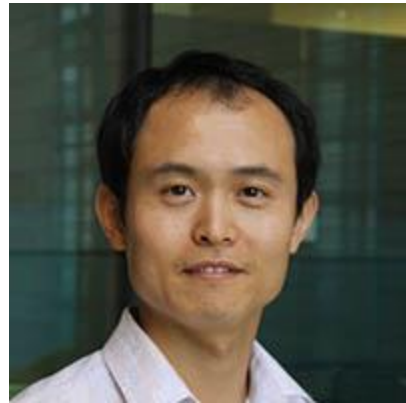


- **Explain the impact, limits, and dangers** of machine learning; **give use cases** of machine learning for the built environment;
- **Explain the main concepts in machine learning** (e.g., regression, classification, unsupervised learning, supervised learning, dimensionality reduction, overfitting, training, validation, cross-validation, learning curve, and regularization);
- **Explain the principles** of well-established unsupervised and supervised machine learning techniques (e.g., clustering, linear regression, Bayesian classification, logistic regression, SVM, random forest, and neural networks);
- **Preprocess data** (e.g., labeling, normalization, feature selection, augmentation, train-test split) for applying machine learning techniques;
- **Select and apply the appropriate machine learning method** for a specific geospatial data processing task (e.g., object classification or semantic segmentation);
- **Evaluate** the performance of machine learning models.

Organization of GEO5017



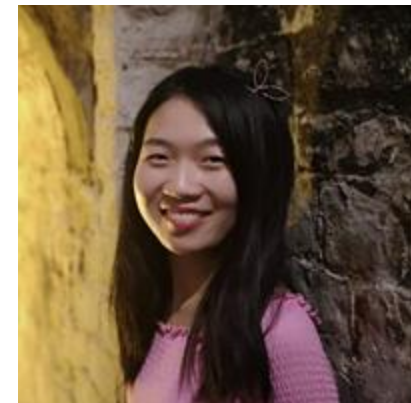
- The teachers



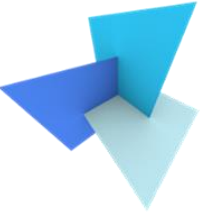
[Liangliang Nan](#)
LiangliangNan#0976



[Nail Ibrahimli](#)
nibrahimli#5857



[Shenglan Du](#)
Shenglan Du#2136



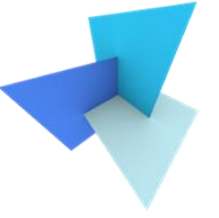
Learning activities

- Lectures
 - Mostly 2 x 45min per week
 - Lecture room
- Lab exercises (and work on assignments)
 - Mostly 2 sessions (2 x 45min each) per week
 - In the booked lecture rooms
 - Teachers available

Lectures



1 & 2. Introduction to ML & the course	[Liangliang]
3 & 4 Linear regression & Gradient descent	[Liangliang]
5 & 6 Clustering & Nearest neighbor classification	[Liangliang]
7 & 8 Bayesian classification & logistic regression	[Shenglan]
9 & 10 Support vector machine (SVM)	[Shenglan]
11 & 12 Decision trees and random forest	[Shenglan]
13 & 14 Neural networks	[Nail]
15 & 16 Deep learning (CNN)	[Nail]



Assessment

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



Assessment

- Pass (on a 100-point scale)?
 - Assignments $\geq 55\%$
 - Exam $\geq 55\%$
 - Total $\geq 57.5\%$
- Repair an assignment: if lower than 55%
 - Only one chance
 - Only be assessed with a 60%
- Retake the exam: if lower than 55%
 - Only one chance
 - An entirely new examination

Assignments



- Two mandatory assignments, each released after the lecture
- Programming: implementation and experiment with ML algorithm(s)
- Work in groups (3 students per group)
- What to submit
 - Report
 - Individual contribution

Isaac Newton (75 %)

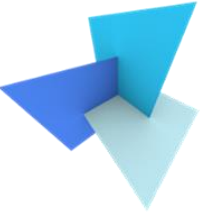
- Compared the reconstruction results from method [1] and method [2];
- Implemented the function `reorient_normals ()`;
- Came up with a novel reconstruction method and implemented it in function `reconstruct()`;
- Wrote the “Methodology” section of the report.

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.



Assignments

- Two mandatory assignments, each released after the lecture
- Programming: implementation and experiment with ML algorithm(s)
- Work in groups (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:** -40%

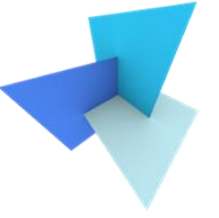


Assignments

- Two mandatory assignments, each released after the lecture
- Programming: implementation and experiment with ML algorithm(s)
- Work in groups (3 students per group)
- What to submit
- We allow multiple submissions
 - Incorporating comments from teachers/peers
 - Evaluation based on 1st submission + 5% maximum

Example:

First submission 60%, then final mark will be $\leq 65\%$



Assignments

- Two mandatory assignments, each released after the lecture
- Programming: implementation and experiment with ML algorithm(s)
- Work in groups (3 students per group)
- What to submit
- We allow multiple submissions
- Strict deadlines
 - Late submission
 - 10% deducted per day late
 - Not acceptable after 3 days late

Assignments

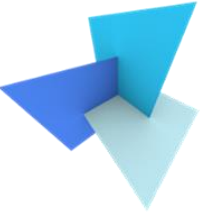


- Two mandatory assignments, each released after the lecture
- Programming: implementation and experiment with ML algorithm(s)
- Work in groups (3 students per group)
- What to submit
- We allow multiple submissions
- Strict deadline
- Teamwork: **Everyone active in coding/discussion/reporting**
 - **We 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



Assignments

- Copy from others/Internet, or use ChatGPT, DeepSeek...
 - Code
 - Sentences
 - Figures
 - ...
- Submit to BrightSpace [plagiarism check turned on]



Assignments

- Not designed to challenge or test you, but
 - To help you gain knowledge and experience
 - To help teacher to gain insights into your progress
 - Tune teaching methods and improve materials

Forget the mark

Ask questions

Enjoy the process!!!

Final exam



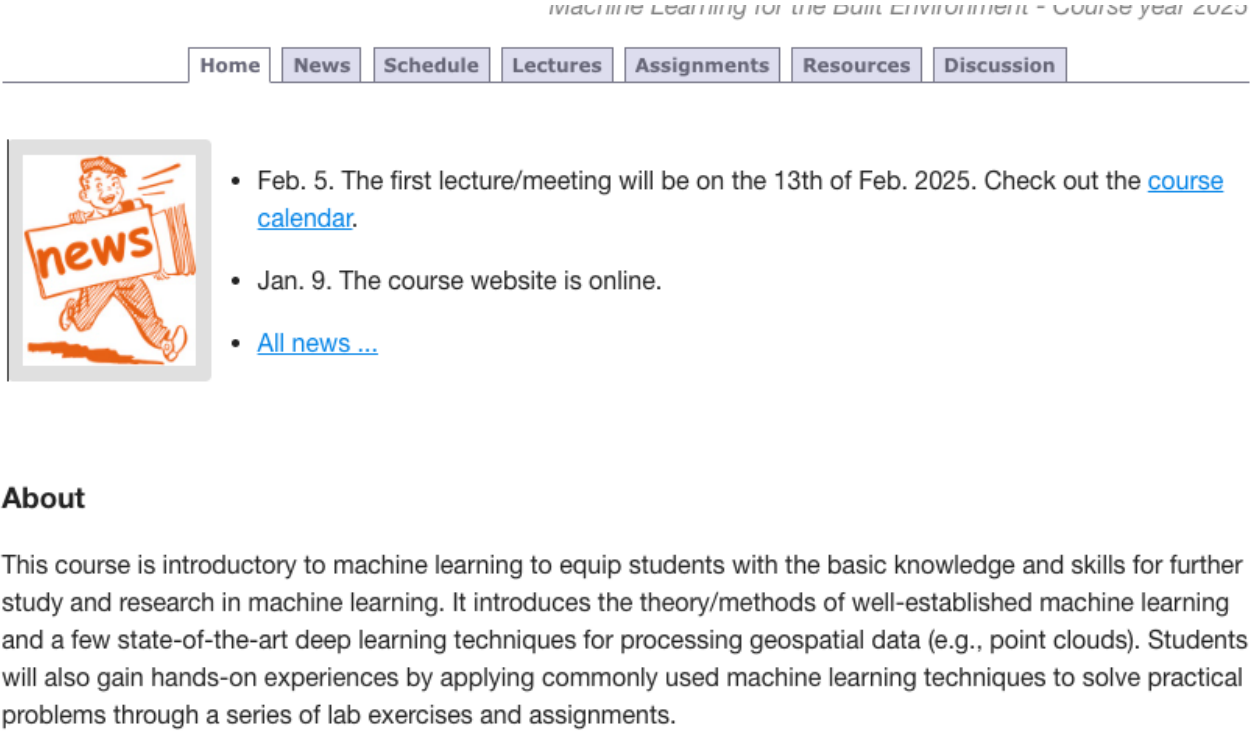
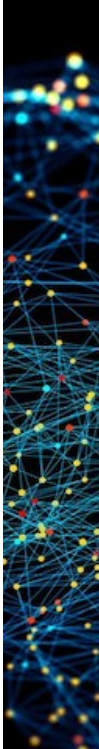
- Multiple-choice questions
- Open questions
 - Lectures, handouts, assignments
- Example questions available before the exam

Communication




- Course website

- <https://3d.bk.tudelft.nl/courses/geo5017/>



Machine Learning for the Built Environment - Course year 2025

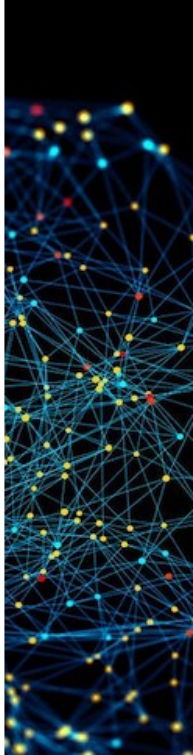
Home News Schedule Lectures Assignments Resources Discussion



- Feb. 5. The first lecture/meeting will be on the 13th of Feb. 2025. Check out the [course calendar](#).
- Jan. 9. The course website is online.
- [All news ...](#)

About

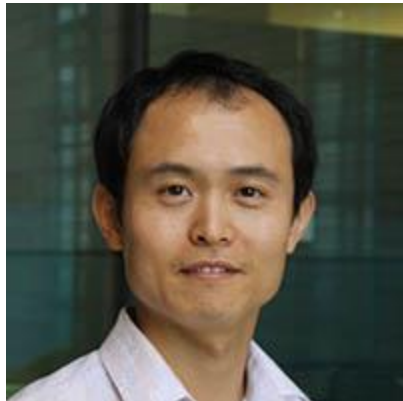
This course is introductory to machine learning to equip students with the basic knowledge and skills for further study and research in machine learning. It introduces the theory/methods of well-established machine learning and a few state-of-the-art deep learning techniques for processing geospatial data (e.g., point clouds). Students will also gain hands-on experiences by applying commonly used machine learning techniques to solve practical problems through a series of lab exercises and assignments.



Communication



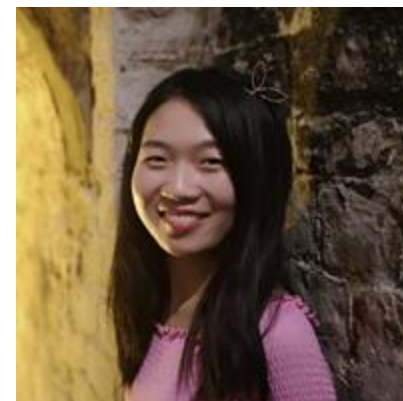
- Discussion
 - Lab/Lecture hours
 - Discord channel



[Liangliang Nan](#)
LiangliangNan#0976



[Nail Ibrahimli](#)
nibrahimli#5857



[Shenglan Du](#)
Shenglan Du#2136



Team up for assignments

- Group assignments
- 3 students per team
- Assignments only visible for those in teams

The screenshot shows the TU Delft course interface for GEO5017 Machine Learning for the Built Environment. The top navigation bar includes the TU Delft logo, the course title, and icons for a grid, email, chat, and notifications. The user profile 'Liangliang Nan as Student' is on the right. Below the navigation bar, a menu bar contains 'Course Home', 'Content', 'Collaboration', 'Assignments', 'Grades', and 'Help'. The 'Collaboration' menu is expanded, showing a list of options: 'Discussions', 'Virtual Classroom', 'Groups' (which is highlighted with a red arrow), and 'Group Self Enrollment'. The background of the page features a large image with the text 'GEO5017 (2024/25) Machine Learning for the Built Environment'. At the bottom, there are sections for 'Announcements' and a 'Calendar' dropdown.

Next Lecture



- Linear regression & gradient decent

