

The lecture will start at 08:15

# TDT 4173: Machine Learning

## Introduction

Zhirong Yang  
Department of Computer Science NTNU

## Reference group members needed

- ▶ We want 3+ students to join the reference group.
- ▶ As a reference group member
  - ▶ You can help us improve the course
  - ▶ You get a certificate of the trusted position
  - ▶ You need to
    - ▶ actively communicate with other students
    - ▶ have two meetings with the teaching team (we provide food and drinks)
    - ▶ reflect student feedback to us
- ▶ Email me [zhirong.yang@ntnu.no](mailto:zhirong.yang@ntnu.no) if you want to join

# Outline

- ▶ Introduction to Machine Learning
- ▶ break
- ▶ Practical course information
- ▶ Introduction of the individual assignment task

# Machine Learning Introduction

# ML Application

## Face recognition

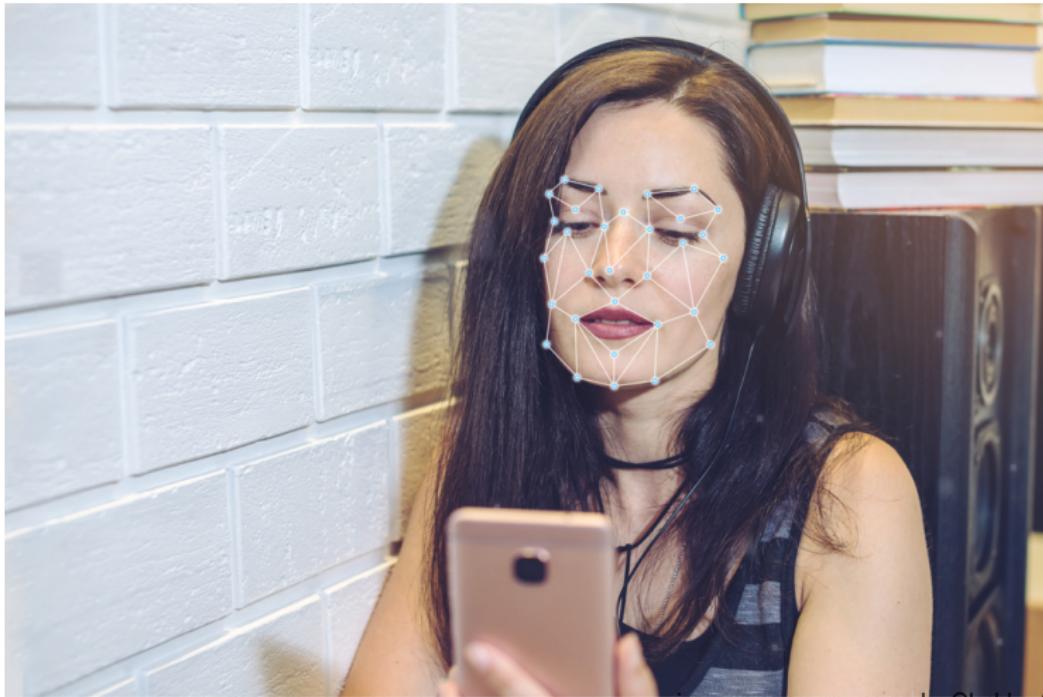


image source: rnz.co.nz by Oleshko Artem

# ML Application

## Spam e-mail filtering



image source: SVMA News

# ML Application

## Machine translation

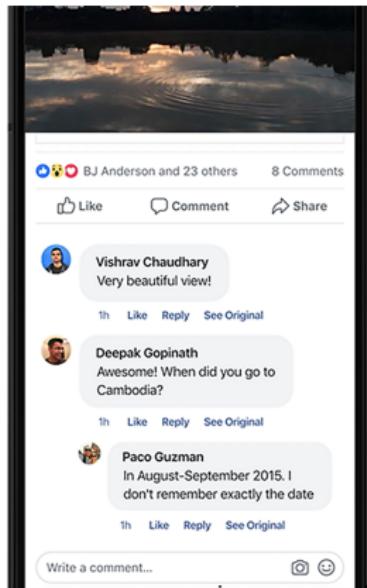
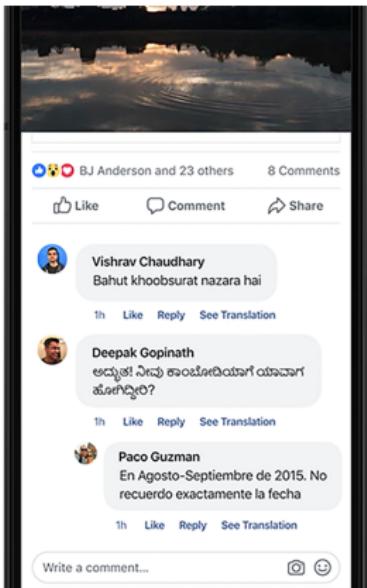


image source: Facebook Engineering

# ML Application

## Speech recognition (virtual personal assistants)



Image source: Shutterstock

## ML Application

## Recommendation

## Frequently Bought Together



**Price For Both: £28.39**

Show availability and delivery details

- This Item:** Rucanor Aluminium Baseball Bat, Silver - 60 cm £18.40
  - WILSON** Official League Individual Baseball £9.99

#### **Customers Who Bought This Item Also Bought**



**REYDON Softball Ball**  
★ ★ ★ ★ ★ (1)  
£3.99



New Midwest Slugger  
Baseball Glove Vinyl  
Catching Mitt Left Hand  
Junior / Senior



Rawlings 9" Indoor / Outdoor  
T-Ball Training Baseball -  
TVB  
 (3)  
\$12.99

Page 1 of 18

# ML Application

## Cancer Diagnosis

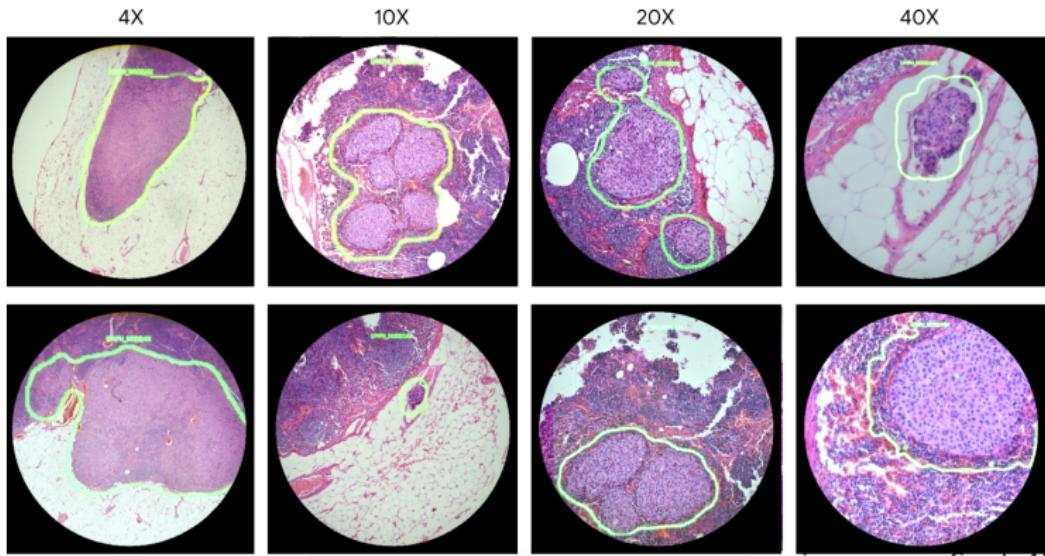


image source: Google AI Blog

# What is machine learning?

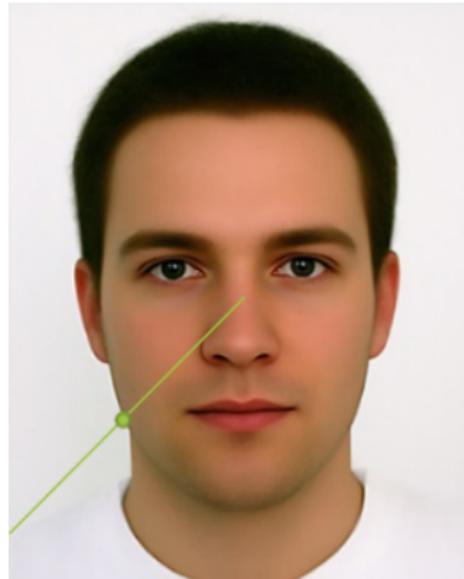
Machine learning gives computers the ability to learn **without being explicitly programmed.**

—Arthur Samuel (1959)

## Example: face extraction (detection)

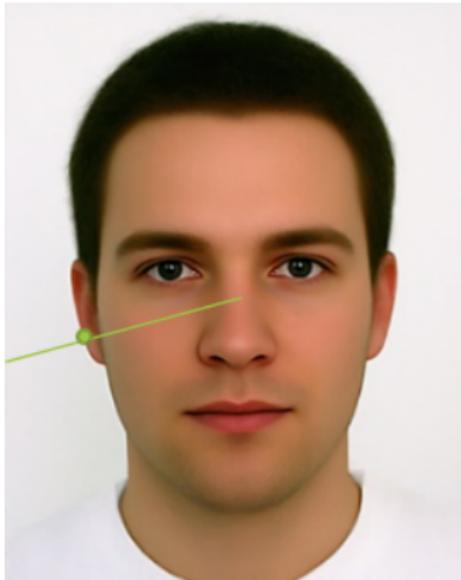


# An face detection algorithm (non-ML)



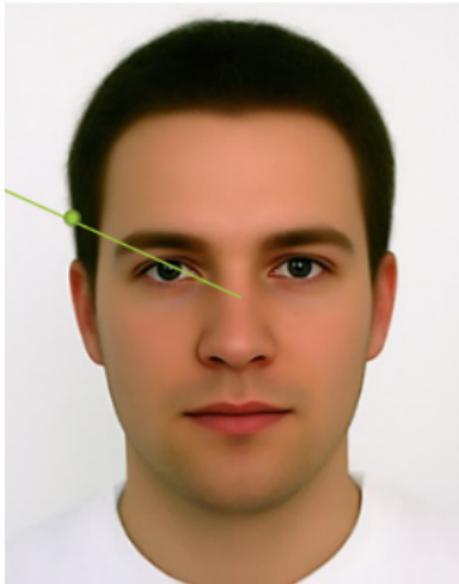
1. Calculate gradients along a line from the center and record the place with the maximum gradient magnitude

## An face detection algorithm (non-ML)



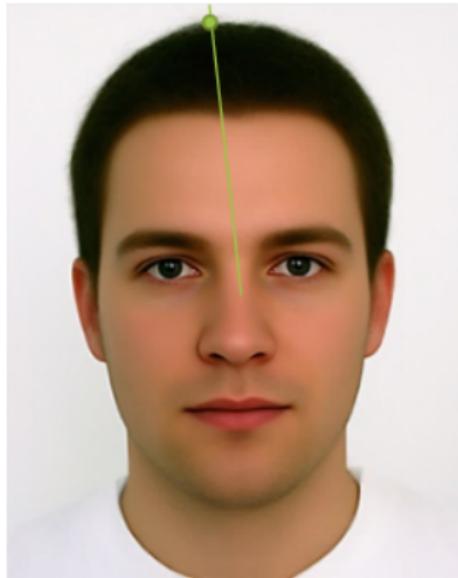
1. Calculate gradients along a line from the center and record the place with the maximum gradient magnitude

## An face detection algorithm (non-ML)



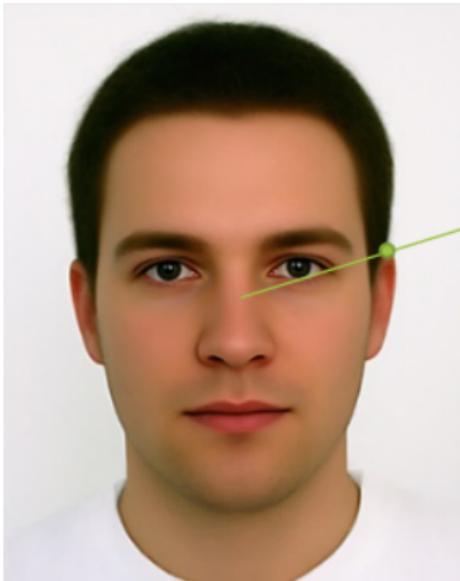
1. Calculate gradients along a line from the center and record the place with the maximum gradient magnitude

## An face detection algorithm (non-ML)



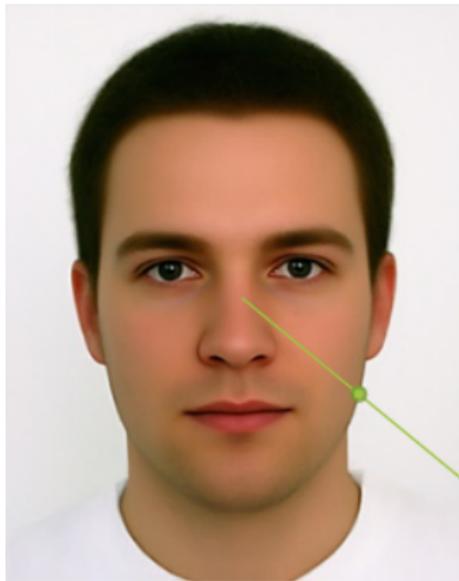
1. Calculate gradients along a line from the center and record the place with the maximum gradient magnitude

# An face detection algorithm (non-ML)



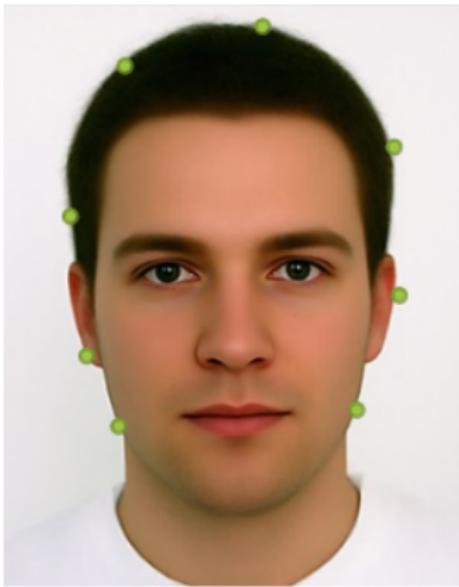
1. Calculate gradients along a line from the center and record the place with the maximum gradient magnitude

## An face detection algorithm (non-ML)



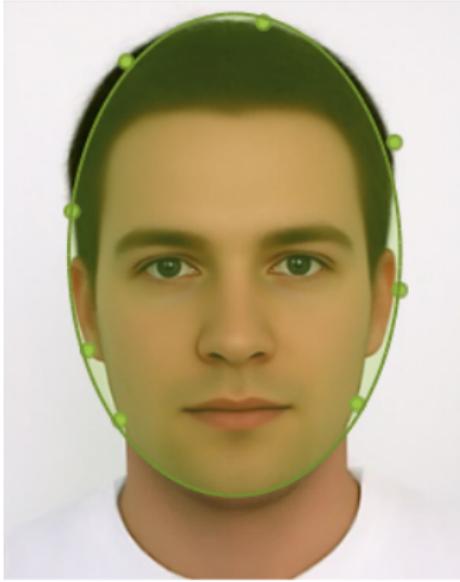
1. Calculate gradients along a line from the center and record the place with the maximum gradient magnitude

## An face detection algorithm (non-ML)



1. Calculate gradients along a line from the center and record the place with the maximum gradient magnitude
2. Repeat for many lines and get a set of “border points”

## An face detection algorithm (non-ML)



1. Calculate gradients along a line from the center and record the place with the maximum gradient magnitude
2. Repeat for many lines and get a set of “border points”
3. Fit the closest ellipse

## An face detection algorithm (non-ML)



1. Calculate gradients along a line from the center and record the place with the maximum gradient magnitude
2. Repeat for many lines and get a set of “border points”
3. Fit the closest ellipse

What are the drawbacks of the algorithm?

# Exceptions

## Hairstyles



## Occlusion



## Background



## Wearings



image source: Google

26 years ago



26 years ago



Human still defeated computers by 10 stones in handicap Go.

# AlphaGo (2015)



## ARTICLE

doi:10.1038/nature16961

### Mastering the game of Go with deep neural networks and tree search

David Silver<sup>1</sup>\*, Aja Huang<sup>1</sup>\*, Chris J. Maddison<sup>1</sup>, Arthur Guez<sup>2</sup>, Laurent Sifre<sup>3</sup>, George van den Driessche<sup>2</sup>, Julian Schrittwieser<sup>1</sup>, Ioannis Antonoglou<sup>2</sup>, Veda Panneershelvam<sup>2</sup>, Marc Lanctot<sup>1</sup>, Sander Dieleman<sup>2</sup>, Dominik Grewe<sup>1</sup>, John Nham<sup>3</sup>, Nal Kalchbrenner<sup>2</sup>, Ilya Sutskever<sup>2</sup>, Timothy Lillicrap<sup>2</sup>, Madeleine Leach<sup>2</sup>, Koray Kavukcuoglu<sup>2</sup>, Thore Graepel<sup>2</sup> & Demis Hassabis<sup>1</sup>

The game of Go has long been viewed as the most challenging of classic games for artificial intelligence owing to its enormous search space and the intrinsic difficulty of evaluating board positions and moves. Here we introduce a new approach to computer Go that uses ‘value networks’ to evaluate board positions and ‘policy networks’ to select moves. These deep neural networks are trained by a novel combination of supervised learning from human expert games, and reinforcement learning from games of self-play. Without any lookahead search, the neural networks play Go at the level of state-of-the-art Monte Carlo tree search programs that simulate thousands of random games of self-play. We also introduce a new search algorithm that combines Monte Carlo simulation with value and policy networks. Using this search algorithm, our program AlphaGo achieved a 99.8% win rate against other Go programs, and defeated the human European Go champion by 5 games to 0. This is the first time that a computer program has defeated a human professional player in the full-sized game of Go, a feat previously thought to be at least a decade away.

Machine Learning defeated Europe champion (2 dan professional)

# AlphaGo (2016)



Machine Learning defeated world ex-champion (9 dan professional)

# AlphaGo (2017)



Machine Learning defeated current world champion

# AlphaGo (2017)



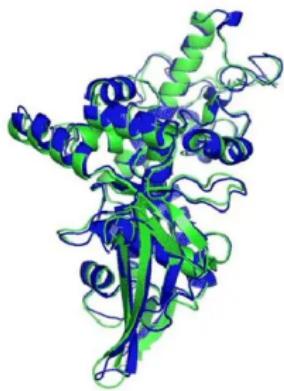
Machine Learning defeated the team of five top Go players (all are 9 dan professional)

# AlphaStar (2019 October)

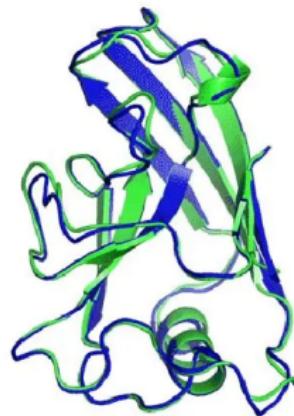


DeepMind's StarCraft 2 AI defeated 99.8% of human players

# AlphaFold 2: predicting protein structure (2021)



T1037 / 6vr4  
90.7 GDT  
(RNA polymerase domain)



T1049 / 6y4f  
93.3 GDT  
(adhesin tip)

- Experimental result
- Computational prediction

image source: A Silicon Valley Insider

# DALL-E-2: Text2Image (2022)



vibrant portrait painting of Salvador Dalí with a robotic half face



a shiba inu wearing a beret and black turtleneck



a close up of a handpalm with leaves growing from it



an espresso machine that makes coffee from human souls, artstation



panda mad scientist mixing sparkling chemicals, artstation



a corgi's head depicted as an explosion of a nebula

image source: Analytics India Magazine

## Midjourney: AI image editing (2023)

Midjourney Bot ✅BOT Today at 5:31 PM  
**close up photo two men middle aged white men  
facing off in a boxing ring --ar 16:9 - Image #3**

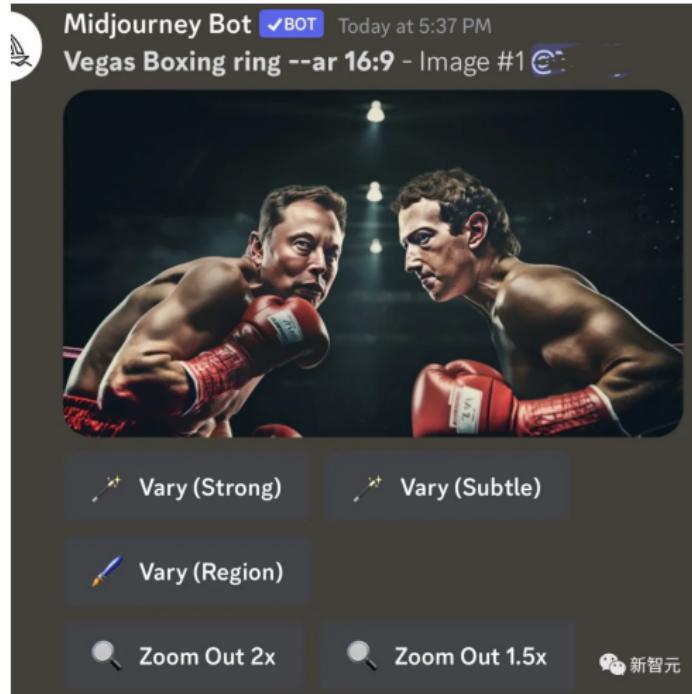


→ Vary (Strong)

### Vary (Subtle)

新智元

# Midjourney: AI image editing (2023)



# ChatGPT (2022-2023)

Me Can you write a function that takes a string input and capitalizes the first letter of each word?

ChatGPT Sure, here is a possible function that capitalizes the first letter of each word in a string input:

```
def capitalize_first_letters(string):
    words = string.split()
    capitalized_words = []
    for word in words:
        capitalized_word = word[0].upper() + word[1:]
        capitalized_words.append(capitalized_word)
    return " ".join(capitalized_words)
```

79 Give me the motivation to go and do my homework

ChatGPT Remember that completing your homework is an important step towards achieving your goals. Whether it's getting good grades, graduating, or pursuing a career, your education is a crucial part of your future success. Additionally, completing your homework can help you develop valuable skills such as time management, discipline, and responsibility. So, remind yourself of your long-term goals and the benefits of hard work, and get started on your homework!

source: [110 Best ChatGPT Examples To Look At In 2023](#)

# Some rising AI companies

- ▶ 40 Growing AI Companies & Startups In 2023
- ▶ Synthesia
- ▶ Grammarly
- ▶ <https://www.H2O.ai>

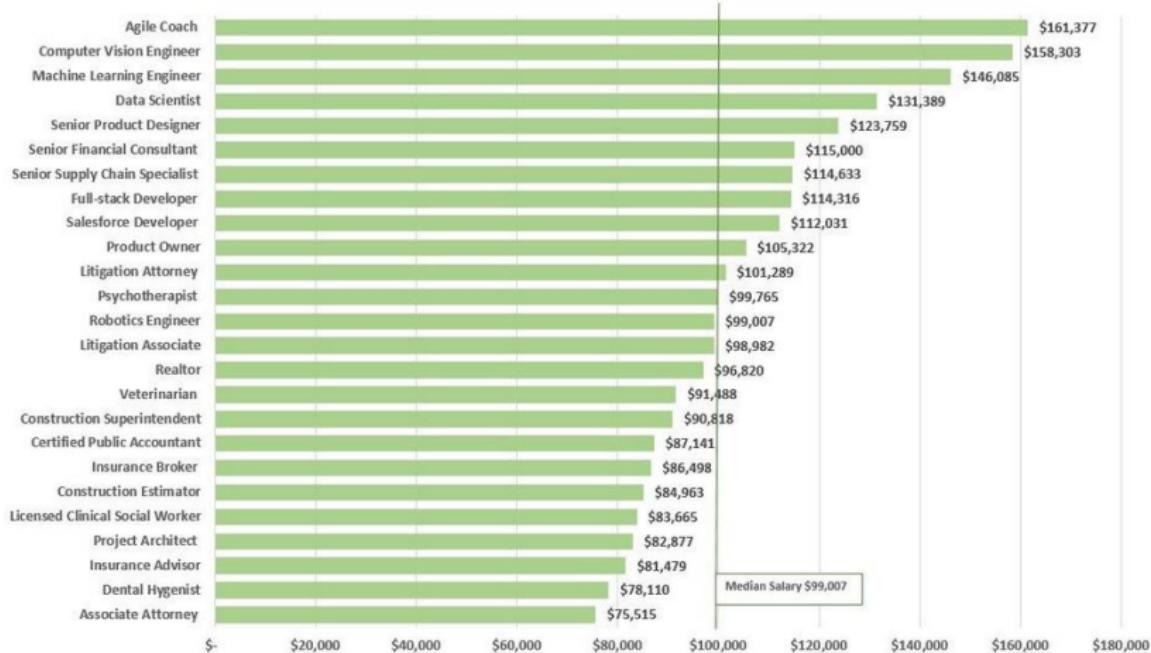
# AI/ML job demands

Job Trends for a Machine Learning Engineer from 2010 to 2020



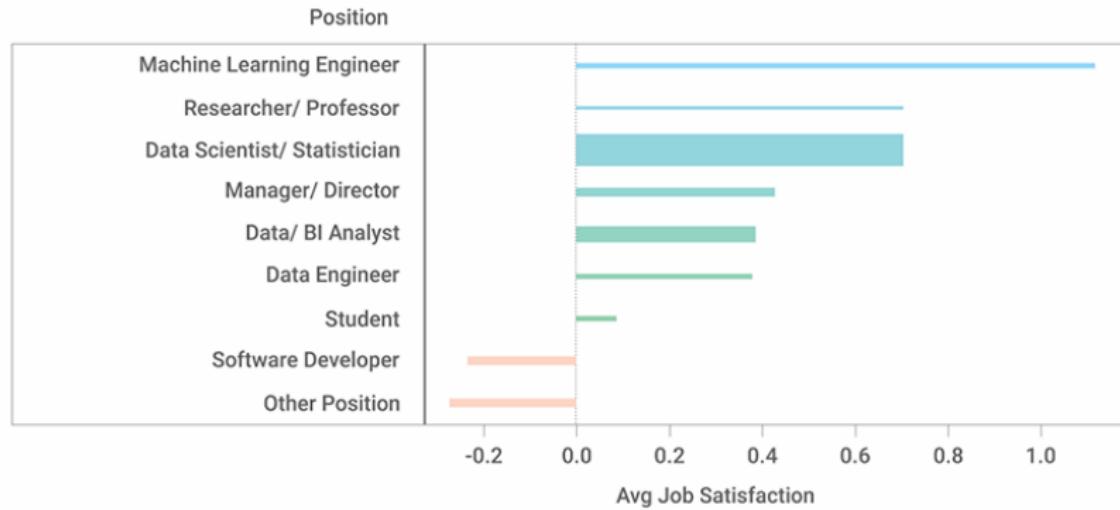
# AI/ML jobs: salary

Indeed's Best Jobs In the U.S.  
Average Base Salary, 2019



# Job satisfaction

KDnuggets Poll: Avg Job Satisfaction by Position



# Demand for AI skills across occupations

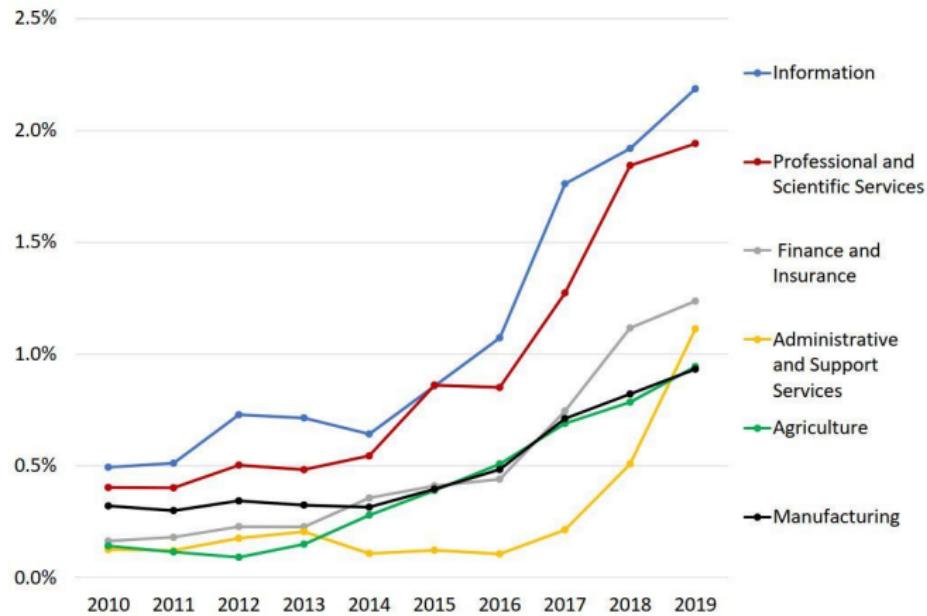


image source: Centre for Economic Policy Research

# Increasing AI Labor Demand

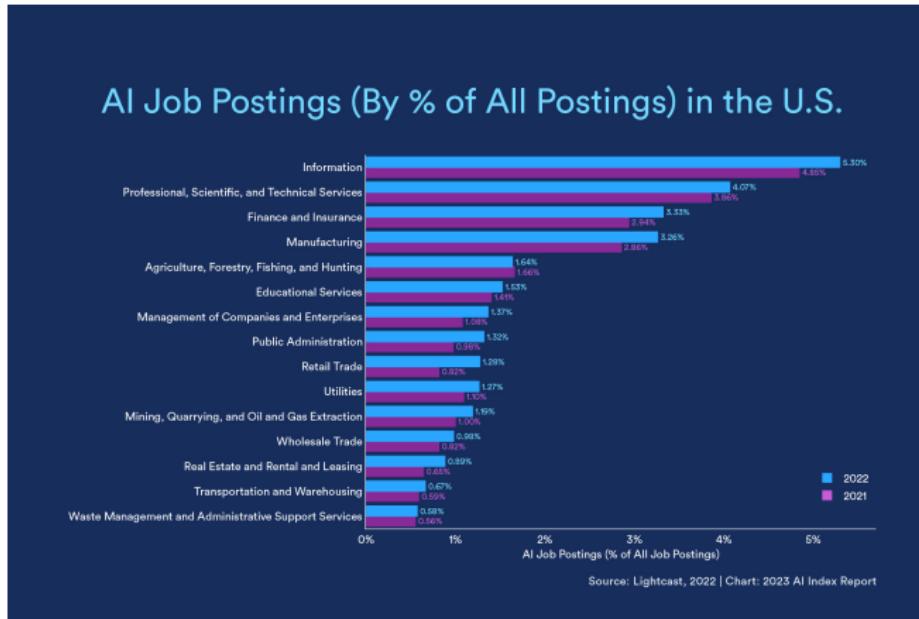


image source: Shana Lynch

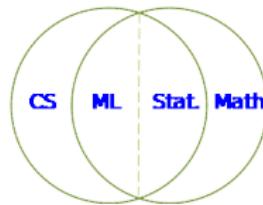
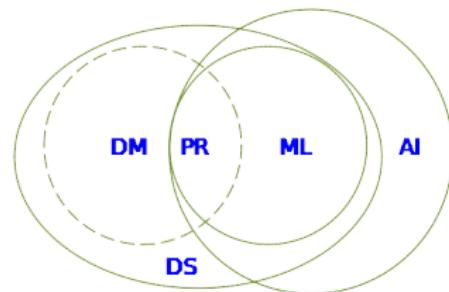
## A more formal definition of ML

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

—Tom Mitchell (1997)

# Related fields

- ▶ Data Science (DS)
- ▶ Data Mining (DM)
  - ▶ DM is not clearly defined
- ▶ Pattern Recognition (PR)
  - ▶ ML evolved from PR
  - ▶  $ML \supset PR$
- ▶ Artificial Intelligence (AI)
  - ▶ ML is a branch of AI
  - ▶ ML is a tool that realizes AI
- ▶ Statistics
  - ▶ ML is an application of statistics
  - ▶ ML is closer to computer science  
(≈ computational statistics)



# ML problem types

- ▶ Supervised learning
  - ▶ Provided with  $\{x_i, y_i\}_{i=1}^N$ , try to learn  $f$  to infer  $y = f(x)$
  - ▶ Examples: classification and regression

# ML problem types

- ▶ Supervised learning
  - ▶ Provided with  $\{x_i, y_i\}_{i=1}^N$ , try to learn  $f$  to infer  $y = f(x)$
  - ▶ Examples: classification and regression
- ▶ Unsupervised learning
  - ▶ Provided with  $\{x_i\}_{i=1}^N$ , try to find structure in the data
  - ▶ Examples: cluster analysis and generation

# ML problem types

- ▶ Supervised learning
  - ▶ Provided with  $\{x_i, y_i\}_{i=1}^N$ , try to learn  $f$  to infer  $y = f(x)$
  - ▶ Examples: classification and regression
- ▶ Unsupervised learning
  - ▶ Provided with  $\{x_i\}_{i=1}^N$ , try to find structure in the data
  - ▶ Examples: cluster analysis and generation
- ▶ Reinforcement learning
  - ▶ Provided with feedback, a computer program interacts with a dynamic environment to achieve a certain goal

## Supervised Learning: classification

$M(\theta) \leftarrow \text{Train} \left( \begin{matrix} \boxed{5} & \boxed{0} & \boxed{4} & \boxed{1} & \dots \\ 5 & 0 & 4 & 1 & \dots \end{matrix} \right)$

$\text{Predict} \left( \boxed{2}; M(\theta) \right) = ?$

# Unsupervised Learning: clustering

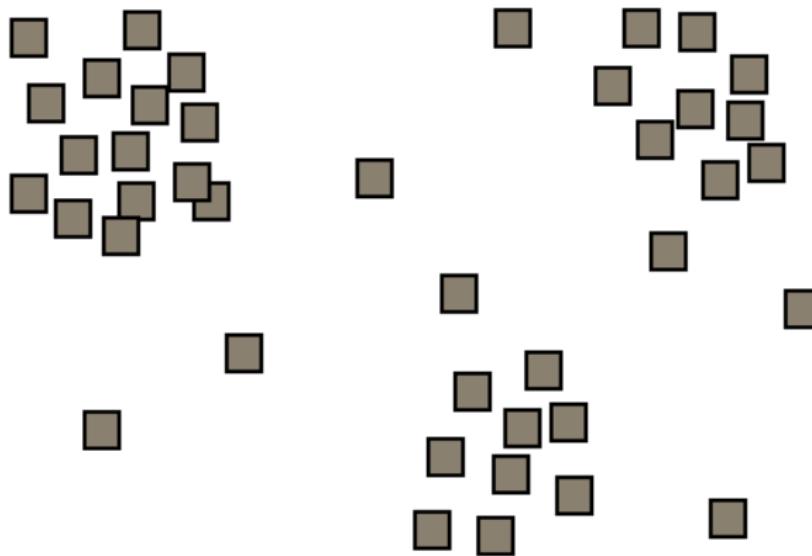


image source: Wikipedia

## Unsupervised Learning: clustering

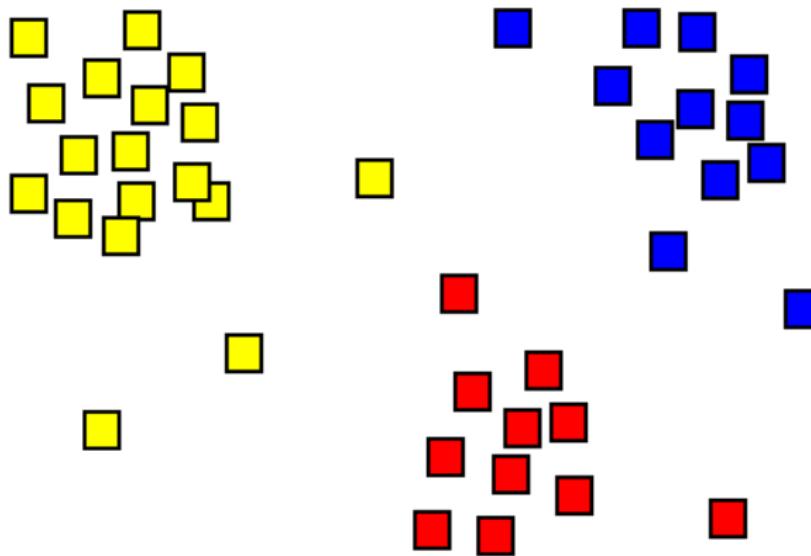


image source: Wikipedia

# Reinforcement learning example

Convolutional Agent

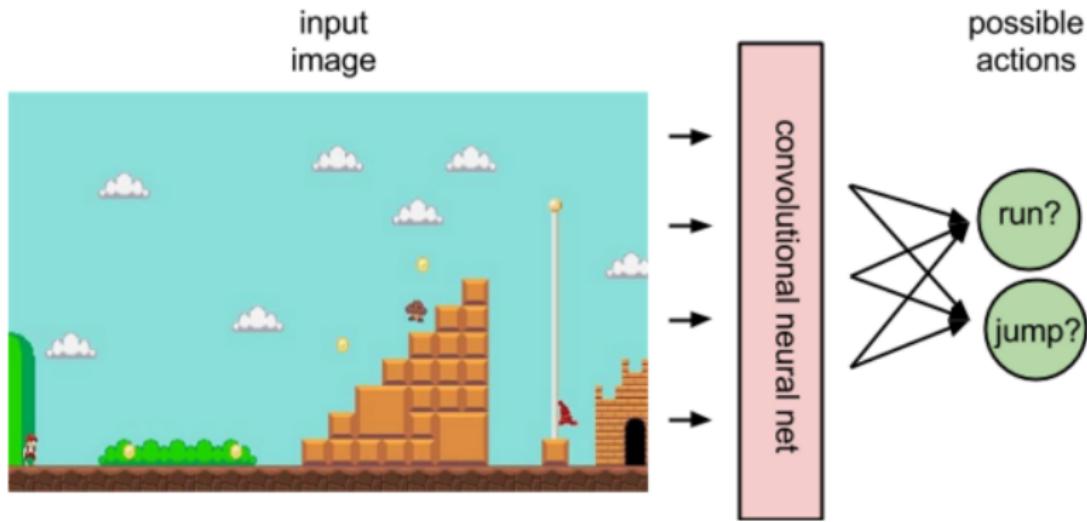
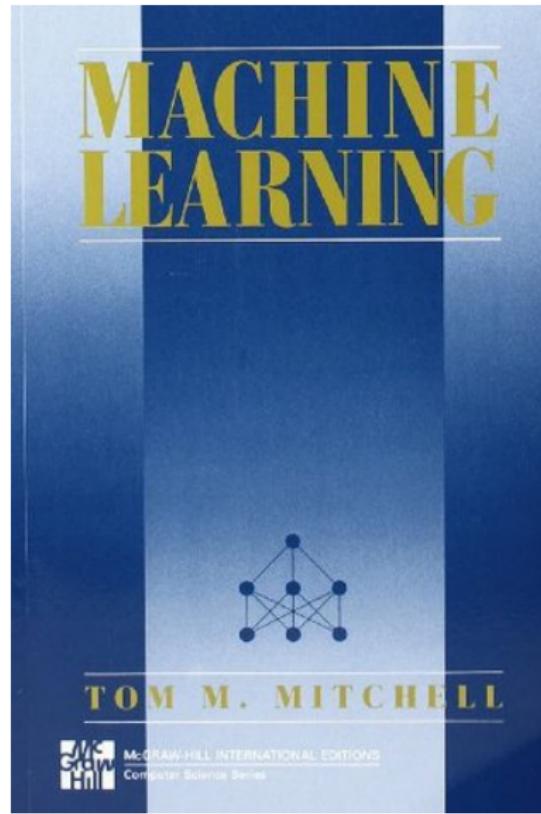
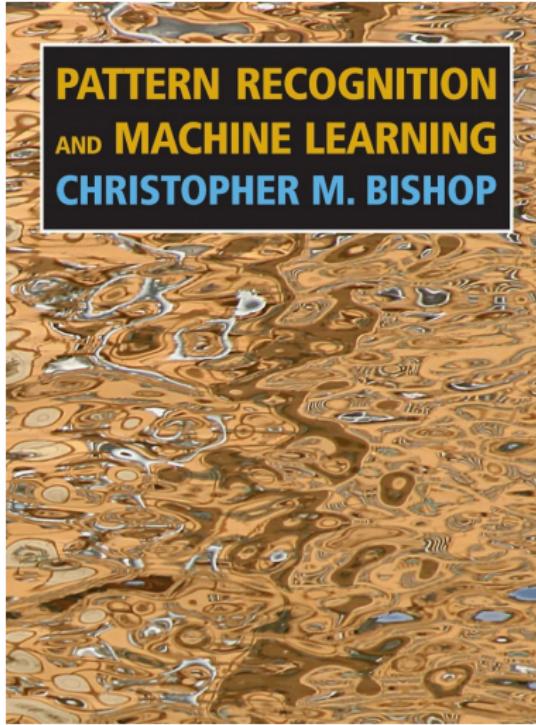


image source: Skymind.ai

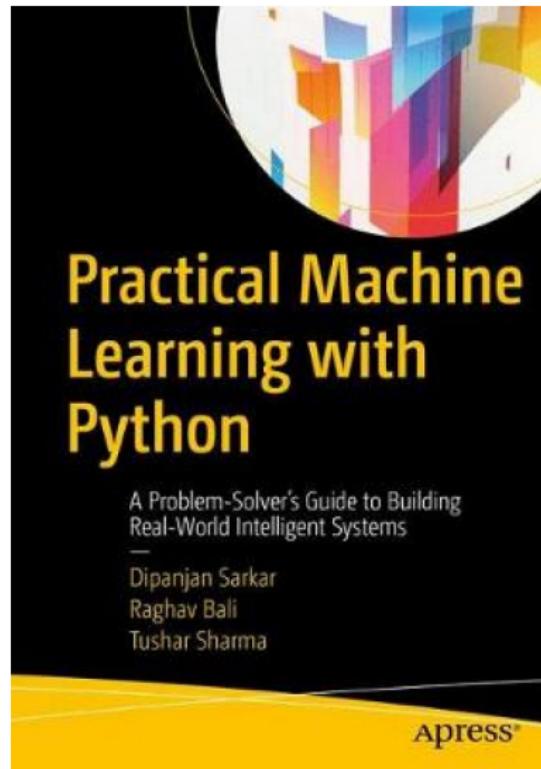
# Textbook 1



## Textbook 2



## Textbook 3



# Homework: software installation

- ▶ We use Python in this course
- ▶ We suggest [Anaconda](#)
- ▶ Install [Scikit-learn](#) and [Pandas](#) if they are not in your Python yet
  - ▶ Other needed software will be announced later
- ▶ Take a look at the tutorials for beginners
  - ▶ [Tutorial of Scikit-learn](#)
  - ▶ [Tutorial of Pandas](#)

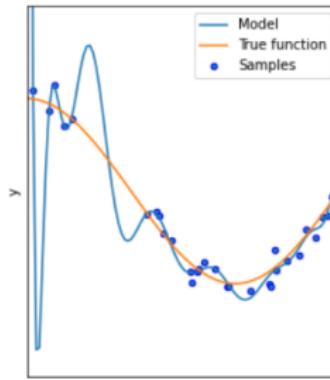
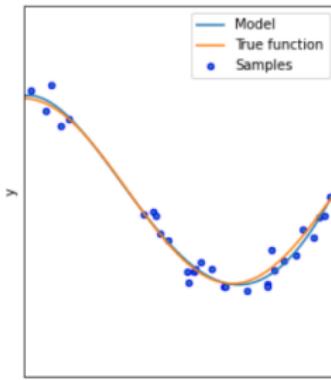
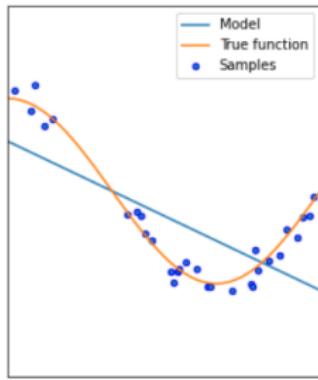
## Further reading: Design choices

- ▶ How to make design choices for ML approaches?
  - ▶ What experience can we learn from?
  - ▶ What exactly should be learned?
  - ▶ How shall it be represented?
  - ▶ Target function: collection of rules, neural network ?
  - ▶ What specific algorithm can we use to learn it?
- ▶ Pointers:
  - ▶ Tom Mitchell, Chapter 2

## Further reading: Inductive Bias

- ▶ Inductive Bias:
  - ▶ Fundamental assumption or set of assumptions that the learner makes about the target function that enables it to generalize beyond the training data.
- ▶ Pointers:
  - ▶ Tom Mitchell, Chapter 2
  - ▶ [From Pralhad Teggi](#)
  - ▶ [Video from Statistical Machine Learning, Univ. of Tübingen](#)

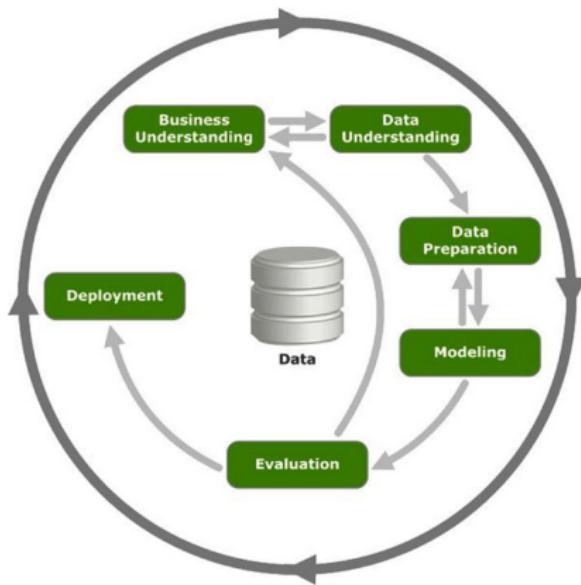
## Further reading: Underfitting and Overfitting



See [Video from Statistical Machine Learning, Univ. of Tübingen](#)

## Further reading: Machine Learning working process

The CRISP-DM model



Read book “Practical Machine Learning with Python” (pages 45-52)

Break

## Practical Course Information

# Teaching time and place

- ▶ Most lectures and exercise sessions are physical
  - ▶ except the guest lecture by BearingPoint will be online
- ▶ Lectures
  - ▶ Mainly on Fridays 8-10, with some exceptional times
  - ▶ The timetable in "NTNU studies" can be obsolete
  - ▶ Check Blackboard for updated information (Course information⇒Course schedule)
- ▶ Exercises (lab hours)
  - ▶ The exercise sessions are mainly for Q&A
  - ▶ Attend if needed
  - ▶ Check Blackboard for times and rooms.

# Plagiarism

- ▶ We take this seriously
- ▶ If a plagiarism case is found, both sides will fail the assignment/project and thereby the course.

# Teaching team

- ▶ Main contact: Zhirong Yang
- ▶ Other lecturers: Massimiliano Ruocco, Odd Erik Gundersen, Liyuan Xing
- ▶ Teaching Assistants: Ruslan Khalitov, Tong Yu, and Lei Cheng
- ▶ Student assistants: Erik Svendsmark, Erlend Heir, Erlend Hordvei, and Jonas Nordstrøm

# Communication

- ▶ Communication channels
  - ▶ Blackboard
  - ▶ Exercise sessions (personal face2face guidance)
  - ▶ If needed, contact a TA for remote guidance
  - ▶ Piazza
  - ▶ Contact us or the reference group for feedback
- ▶ Preferred language: English

# Course Structure

- ▶ Part I
  - ▶ 1<sup>st</sup> round content: introduction, supervised learning, unsupervised learning, deep learning, time series, ensemble learning
  - ▶ Submit: individual assignment (**deadline 17.09.2023!**)
- ▶ Part II
  - ▶ Teaming for project
  - ▶ 2<sup>nd</sup> round content: common ML practice, evaluation methods, reproducibility, probabilistic learning, case-based reasoning, and presentation(s) from industry
  - ▶ Submit: project
- ▶ Presentations by competition winners and runner-ups
- ▶ See Blackboard for a detailed lecture plan (and possible updates)

# Material

For each topic, we will provide

- ▶ Introductory presentation
  - ▶ Give an overview of the topic and provide pointers to relevant work
  - ▶ Slides published in Blackboard
  - ▶ We provide some lecture recordings
- ▶ Reading material
  - ▶ Considered as must-reads/watches to cover the topic
  - ▶ Will provide a starting point for group discussions
  - ▶ To be used in the assignment and the project
- ▶ Additional material
  - ▶ Provided to get a deeper understanding of the methods
  - ▶ Includes state-of-the-art in the field

## Group work: Setup

- ▶ You must pass the individual assignment before project work
- ▶ You work as groups for the project
  - ▶ The students in a group receive the same project points
- ▶ Each group consists of  $\leq 3$  students:
  - ▶ Option 1: you notify us of your group members
    - ▶ You can use Piazza to find other group members
  - ▶ Option 2: you notify us you want to join randomly
    - ▶ We will randomly group students of Option 2
  - ▶ Option 3: If you do not take Option 1 or 2, you have to solo the project
    - ▶ We will create groups of a single student
- ▶ Send your notifications to Lei Cheng ([lei.cheng@ntnu.no](mailto:lei.cheng@ntnu.no))
  - ▶ **Grouping deadline: October 1st, 2023, 23:59**
  - ▶ After the deadline, the remaining students will go to Option 3
- ▶ We are not responsible if some group member(s) leaves during the project

## Group project work: a prediction task

- ▶ In the project, your group competes with us
  - ▶ A higher ranking usually leads to a better course grade
- ▶ Each group submits
  - ▶ Predictions for the competition
  - ▶ Two short Jupyter notebooks
    - ▶ They correspond to your two selected predictions in Kaggle
    - ▶ They contain only the necessary steps to produce your predictions
    - ▶ We may re-run your notebook to check whether the results are reproducible.
  - ▶ A long Jupyter notebook
    - ▶ It contains all attempts in your group work
    - ▶ e.g., exploratory data analysis, all models/algorithms no matter they are helpful or not, model interpretation

# Course grade

- ▶ The course is letter-graded
  - ▶ We first calculate your points (0-100)
  - ▶ Then the points are converted to letters according to the ranges
    - ▶ A: 89-100
    - ▶ B: 77-88
    - ▶ C: 65-76
    - ▶ D: 53-64
    - ▶ E: 41-52
    - ▶ F: 0-40
- ▶ The points are calculated almost based on the project
- ▶ But you must pass the individual assignment first
  - ▶ Try to pass in the first chance (i.e. with the submission before the deadline)
  - ▶ If you fail the first time,
    - ▶ we give you a second chance (resubmit by another given date)
    - ▶ if your re-submission passes, you can still continue to project, **but with a deduction (-5) in your course points**
- ▶ If you fail both chances, you fail the course.

# Course points

- ▶ Your course points are the sum of
  - ▶ base points of your group project
  - ▶ possible deductions
- ▶ Base points
  - ▶  $\propto$  the number of Virtual Teams (VTs) you defeat
  - ▶ max 100 (defeat all VTs) and min 41 (defeat 1 VT)
  - ▶ if you defeat 0 VT, you fail the course
  - ▶ VTs are prepared by the teachers and assistants
  - ▶ there will be 5 or more VTs
- ▶ Possible deductions
  - ▶ pass of individual assignment in the second chance (-5)
  - ▶ late submission ( $\leq$  3 days after the deadline) of the project (-10)
  - ▶ no exploratory data analysis (-3)
  - ▶ only one predictor is used (-3)
  - ▶ no feature engineering (-3)
  - ▶ no model interpretation (-3)
- ▶ More details will be announced later

## Course points and letter grade (example)

- ▶ A student
  - ▶ passes the individual assignment in the first chance
  - ▶ submits the project in time
  - ▶ his/her team defeats 7 of 10 VTs
  - ▶ no model interpretation in the notebooks (-3)
- ▶ Then the student's course points
  - ▶ base points =  $41 + \frac{100 - 41}{10 - 1} \times (7 - 1) \approx 80$
  - ▶ course points =  $80 - 3 = 77$
  - ▶ rounded letter grade is B

Thank you

Questions?