INTRODUCTION TO MACHINE LEARNING

Zhenyu Liao
Associated Professor at EIC HUST
https://zhenyu-liao.github.io/

About myself

- Education and experiences:
  - *2021 Associated Professor with EIC, HUST, Wuhan, China*
  - *2020 Postdoctoral Researcher at University of California, Berkeley, USA*
  - *2019 Ph.D. Statistics and Machine Learning, University of Paris-Saclay, France.*
  - *2016 M.Sc. Signal and Image Processing, University of Paris-Saclay, France.*
  - *2014 B.Sc. Optical & Electronic Information Huazhong university of Science and Technology, Wuhan, China.*
- Awards and prizes :
  - *2021 Recipient of East Lake Youth Talent Program Fellowship of Huazhong University of Science & Technology, Wuhan, China.*
  - *2019 ED STIC Ph.D. Student Award of University Paris-Saclay, France.*
  - *2016: Recipient of the Supélec Foundation Ph.D. Fellowship, France.*
- Academic services:
  - *Referee of European Research Council (ERC); external reviewer of Natural Sciences and Engineering Research Council of Canada (NSERC).*
  - *Reviewer of conferences: NeurIPS, ICML, ICLR, AISTATS, AAAI, etc.*
  - *Reviewer of journals: Journal of Machine Learning Research (JMLR), IEEE Trans. on Pattern Analysis and Machine Intelligence (IEEE-TPAMI), IEEE Trans. on Signal Processing (IEEE-TSP), IEEE Trans. on Neural Networks and Learning Systems (IEEE-TNNLS), SIAM Journal on Scientific Computing (SISC), Pattern Recognition (PR)*

Motivation

- Big Data era: **huge amount** of **large size** data
- Need of **automatic processing** of these big data
- Machine learning (**ML**) as an answer

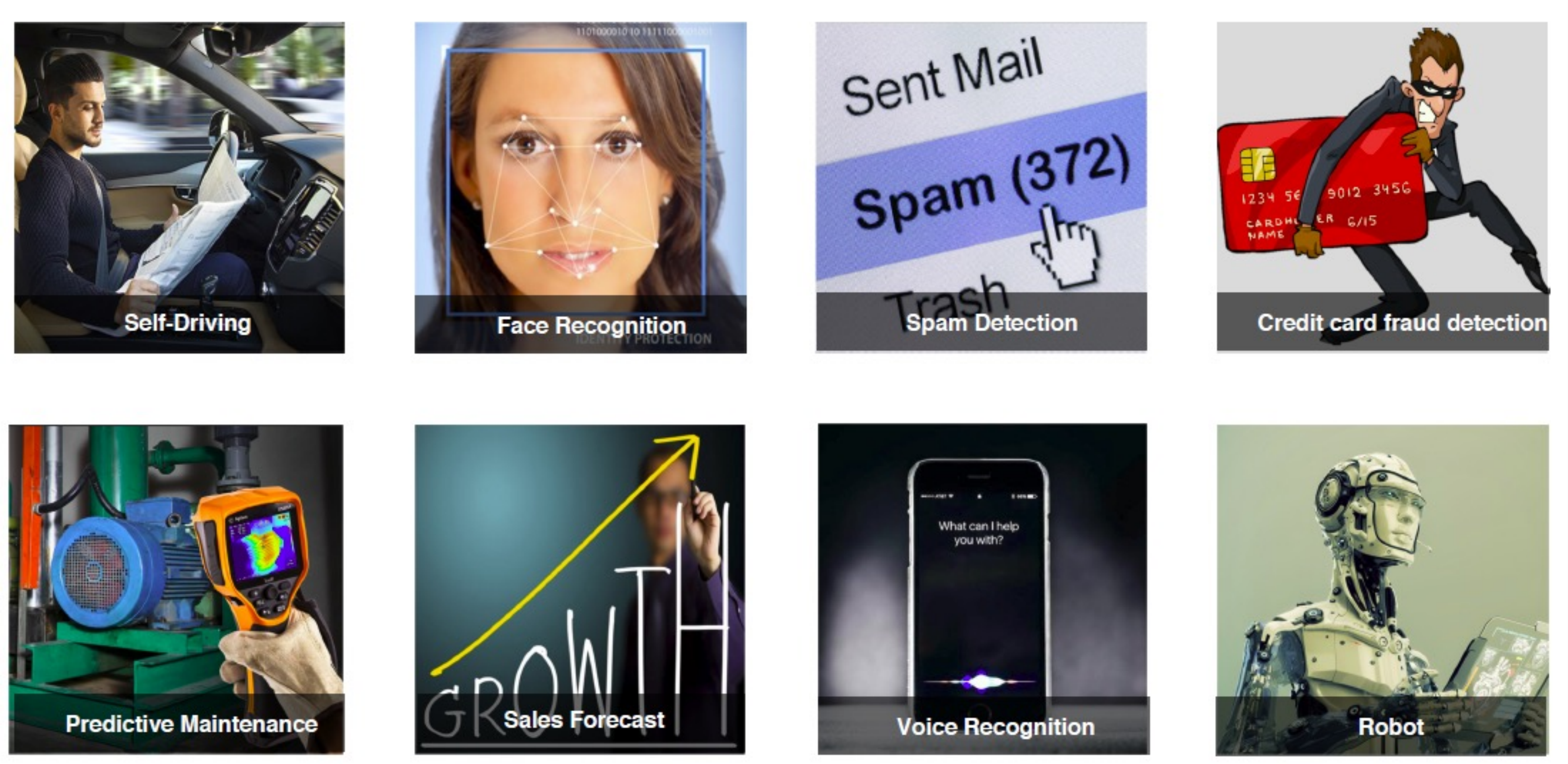
Outline

- Introduction and background of machine learning: rule-based and data-based
- Basics of machine learning
- Machine learning in practice: example of (least-squares) linear regression model
- More advanced considerations in machine learning
- Resources and references

Outline

- Introduction and background of machine learning: rule-based and data-based
- Basics of machine learning
- Machine learning in practice: example of least-squares linear regression model
- More advanced considerations in machine learning
- Resources and references

Machine Learning in everyday life



Applications of machine learning

- <u>Computer Vision</u> (CV):
  - *acquiring, processing, analyzing and understanding digital images and videos*
  - *(direct) examples:*
    - face recognition: Face ID (Apple), security services (Britain, USA, China), etc
    - optical character recognition (OCR): data entry, automatic number plate recognition, etc
    - image search (Google)
    - detecting events: visual surveillance or people counting, etc
  - *play a key role in*
    - autonomous vehicle
    - robotics
  - *to mimic **visual system** of human beings*
  - *rather well developed field, superhuman performance!*

Image: wikipedia

Applications of machine learning

- <u>Natural Language Processing</u> (NLP):
  - *interactions between computers and human natural languages*
  - *examples:*
    - automated online assistant
      - *smart speakers: Amazon Echo, Google Home and Apple HomePod*
      - *on smartphones: Siri (Apple) and Cortana (Microsoft Windows OS)*
      - *online chatbots customer service: Taobao, eBay, Burberry*
    - automatic machine translation
    - question answering system
    - natural language understanding
  - *to mimic **reading system** of human beings*
  - *less developed, but moving fast forward*

Image: wikipedia

Applications of machine learning

- Healthcare
  - *identifying diseases, developing new medicines*
- Robots
- Finance and economics
  - *automatic financial trading*
- Educations
- etc.

Image: wikipedia

Machine Learning (ML) and Artificiel Intelligence (AI)

A point of history:

- born in a workshop at Dartmouth College in 1956
- 1970-1980: AI winter
- 1980s: expert systems, *if–then* rules
- 1990s-21st century: rapid growth due to increasing computational power
  - *1997: Deep Blue (IBM), first* ***chess****-playing computer world champion*
- 2010s-now: age of Deep learning
  - *2017: AlphaGo (DeepMind), first computer* ***Go****-playing system world champion*

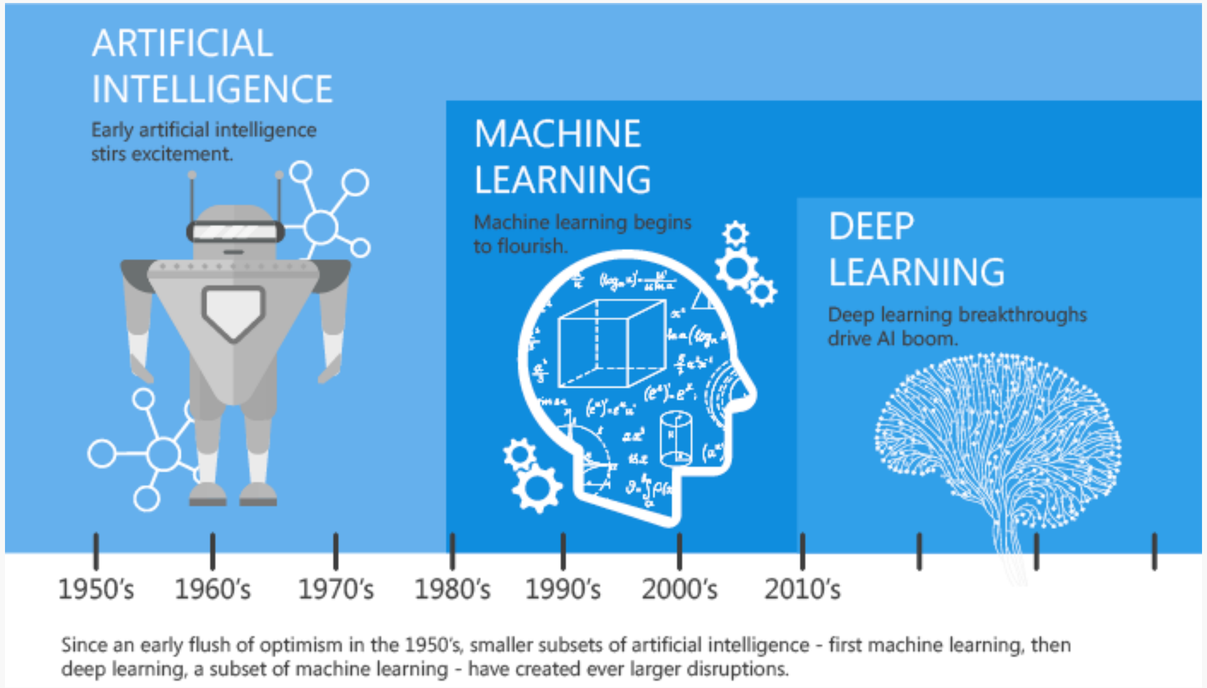


Image: LinkedIn | Machine Learning vs Deep learning

Example of rule-based expert system

- Example: housing prices prediction. House No.1:

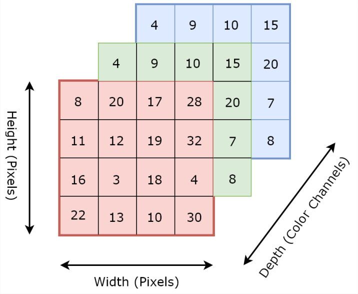
MSSub Class	MSZoning	LotFro ntage	LotArea	Street	Alley	LotShape	LandCo ntour	Utilities	HouseStyl e	Price
20	RH	80	11622	Pave	NA	Reg	Lv1	AllPub	1 Story	65k

- Rule-based system:
  - *price = 0*
  - ***If*** *HouseStyle=1 Story* ***and*** *LotArea<=8000,* ***then*** *price=LotArea*5*
  - ***If*** *HouseStyle=1 Story* ***and*** *LotArea>8000,* ***then*** *price=LotArea*6*
  - ***If*** *HouseStyle=2 Story* ***and*** *LotArea<=12000,* ***then*** *price=LotArea*6.5*
  - ***…***
  - ***if*** *LandContour=Lv1* ***and*** *Street=Pace,* ***then*** *price=price-6k*
  - *…*
- Data from https://www.kaggle.com/alphaepsilon/housing-prices-dataset.

Why not expert systems?

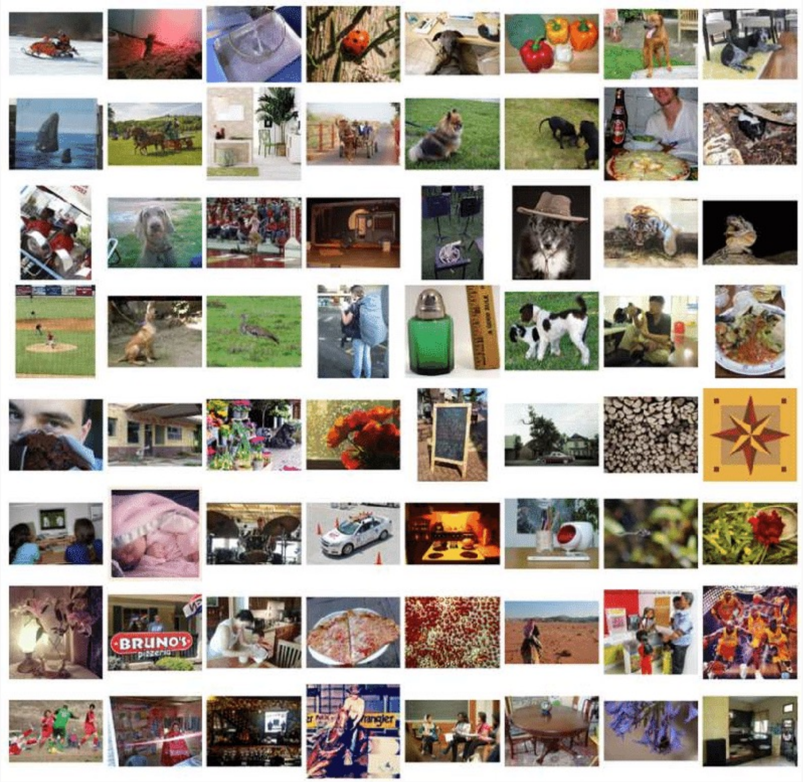
Image: The Verge

- Expert systems: human experts teach computers *rules*
  - *computationally* *impossible* *even for the simple* ***Go*** *game*
    - played across a 19 x 19 grid
    - **search space** = 19 x 19 = 361 for each move
    - 15 turns: 361^30 = 5.3^{e}76 possible configurations
    - approximately 1^{e}80 atoms in the universe
  - *even worse for continuous search space*
    - driving: turn the steering wheel at any time for any degree
- Finding efficient rules are **difficult**: dogs vs. cats



How modern machine learning works?

- learn from (a huge number of) **data**
- try to find "rules" (or the so-called features) from the **data**
- with large amount of **data**, may find better rules than human experts



ML

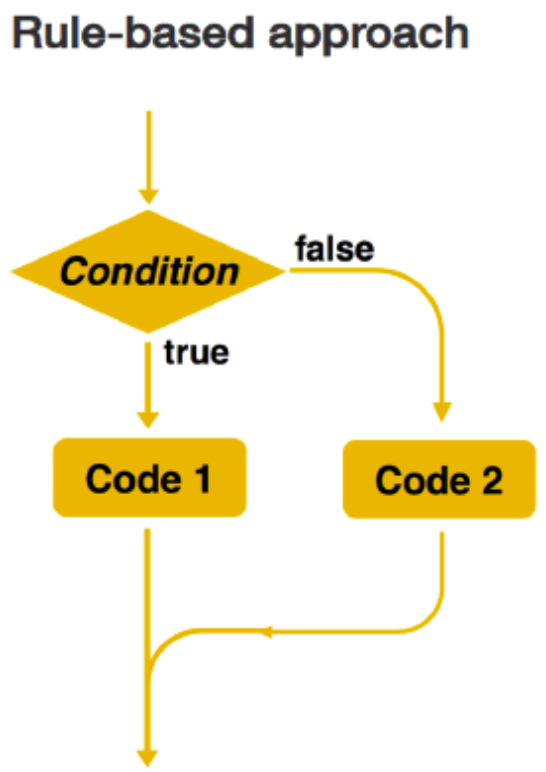
Labels:

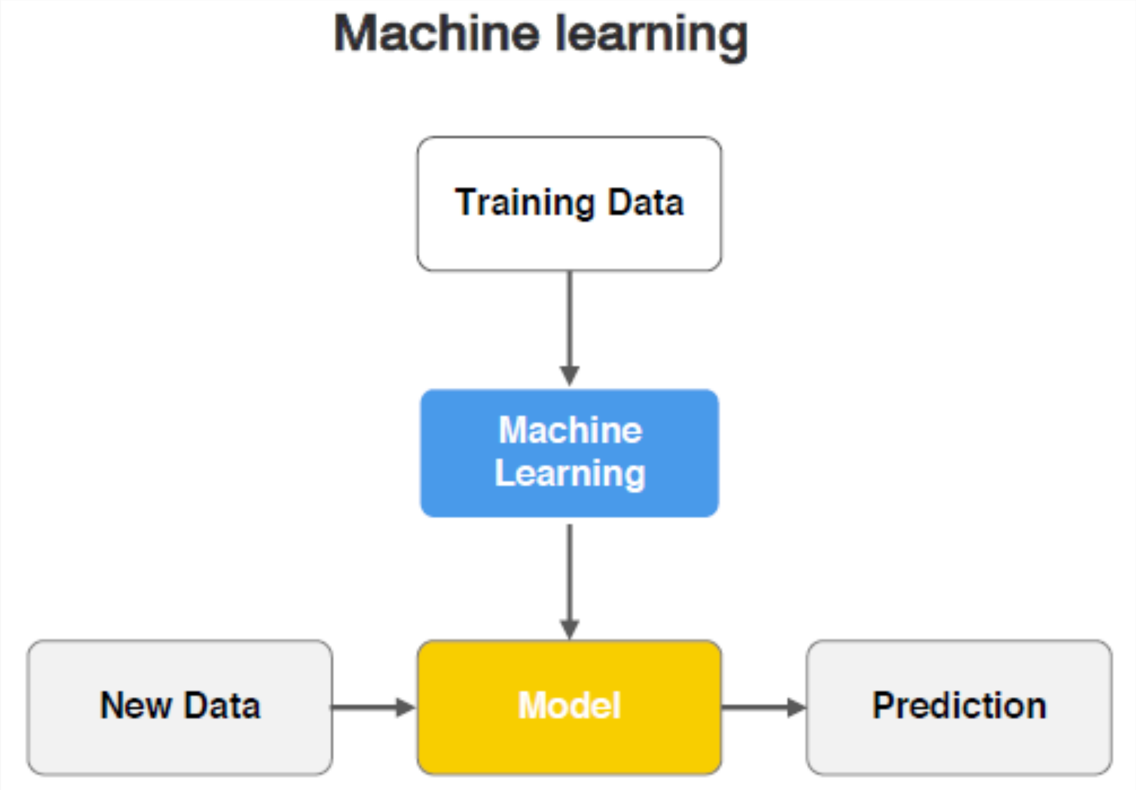
- dogs
- cats
- trains
- aircrafts
- etc.

Why things are working now?

1. Big data
   - *International Data Corporation (IDC) reports the global data volume from 4.4 ZB (1 ZB=10^9 TB = 10^12 GB) to 44 ZB between 2013 and 2020, and predicts 163 ZB by 2025*
   - *continue to grow with development of* ***Internet of things*** *devices*
2. Rapid growth of modern computational systems
   - *Moore's law: the number of transistors in a dense integrated circuit (IC) doubles about every two years, named after Gordon Moore.*
   - *Graphics processing unit (GPU): large-scale parallel computing*
3. **More advanced models/algorithms** (deep neural network models)

Expert system versus ML approach





- Explicitly programmed to solve problem
- Decision rules are clearly defined by humans

- Rules not defined by humans, learned from data/examples
- Highly complex, often not understandable or interpretable

When to use data-driven machine learning?

- For **complicated** tasks: when efficient rules (by human) are hardly accessible
- when it is easy to get a lot of data! (perhaps not the case in healthcare applications)
- When **automatization** is needed
  - *extremely **large-scale** problems, impossible for human: analyze reviews and provide recommendations for film websites (Netflix)*
  - ***high labor costs:** manufacturing in Europe (BMW)*
- Looking for **higher efficiency** and higher quality products
  - *face recognition in security systems*

What do we need to do ML?

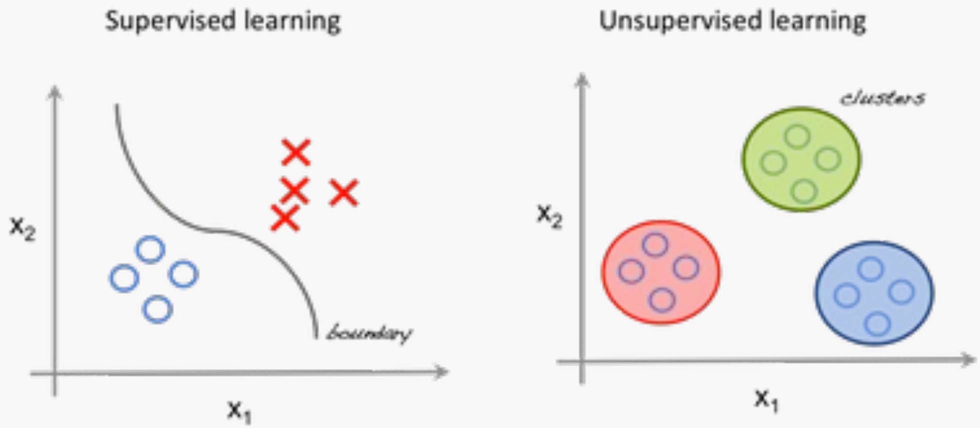
1. Clearly formulate your problem:
   - ***input**? **output**? how can we use (input data) to decide (output)?*
2. **Large amount of data!**
   - *in general, data are more important than algorithms/models*
   - ***the more data, the better!***
   - *not only the data, but also the **correct** labels!*
3. Find efficient **criterion** to evaluate your machine learning model
4. Find significant representation/features from data with algorithms

Outline

- Introduction and background of machine learning: rule-based and data-based
- Basics of machine learning
- Machine learning in practice: example of least-squares linear regression model
- More advanced considerations in machine learning
- Resources and references

Basics of machine learning:

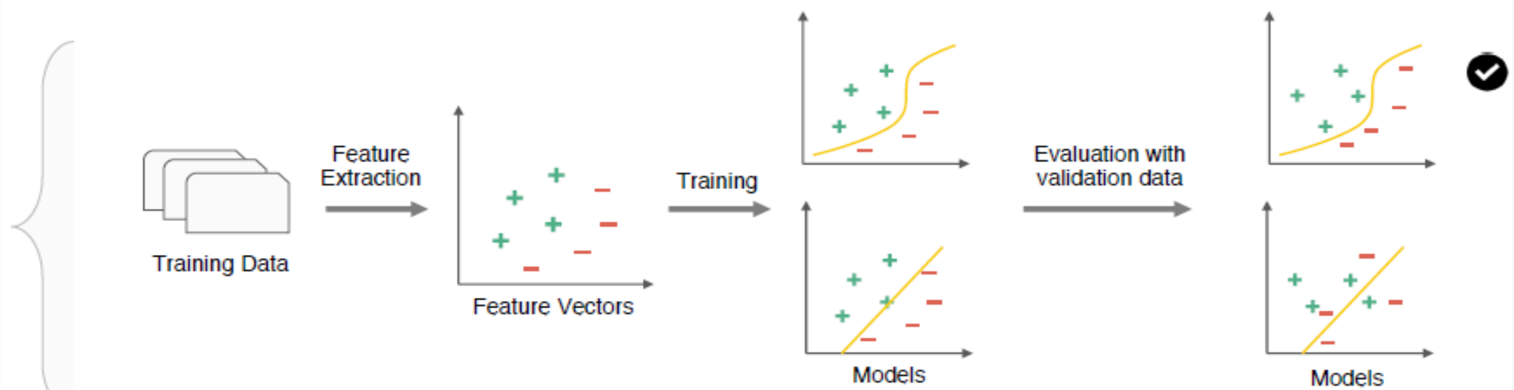
1. **Supervised learning** (<u>**need**</u> labels/targets): classification, regression
2. **Semi-supervised learning** (<u>**less**</u> labels/targets): useful in practice
3. **Unsupervised learning** (<u>**no**</u> labels/targets): find clusters/communities within data



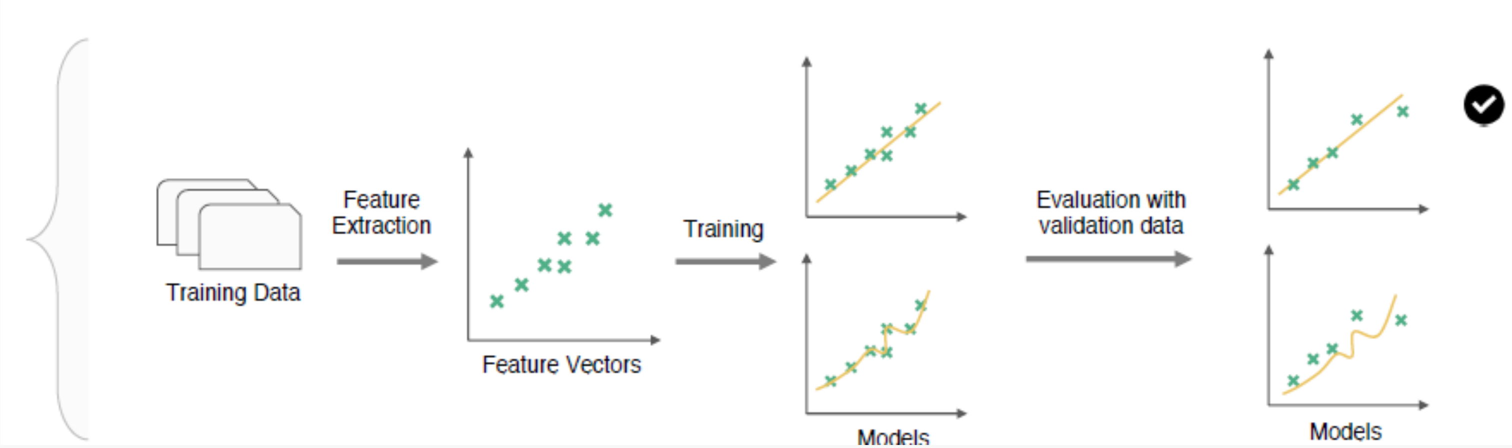
4. **Reinforcement learning** (RL): learn to <u>**make decisions**</u> (the chess and Go games, robotics)

Basics of machine learning: supervised

- Classification: e.g., cats versus dogs, face recognition, anomaly detection



- Regression: housing prices prediction, consumer prediction

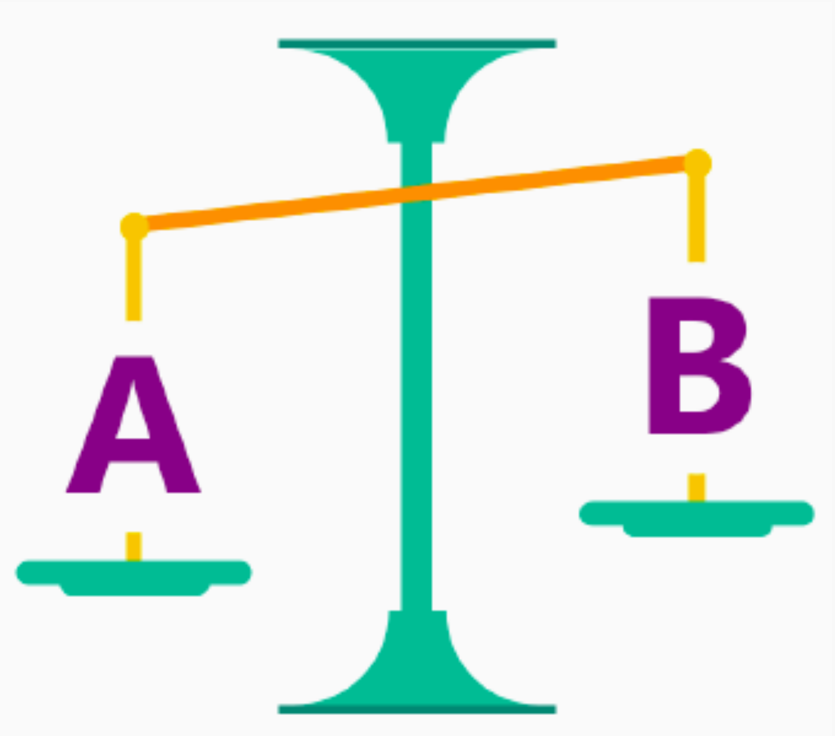


Five typical questions ML can answer

1. Is this A or B?
2. Is this normal?
3. How many? How much?
4. How is this organized? What is the structure of?
5. What should we do?

Based on slides of Prof. Ling Yu: lingyu@fhxy.net.cn

Five typical questions ML can answer

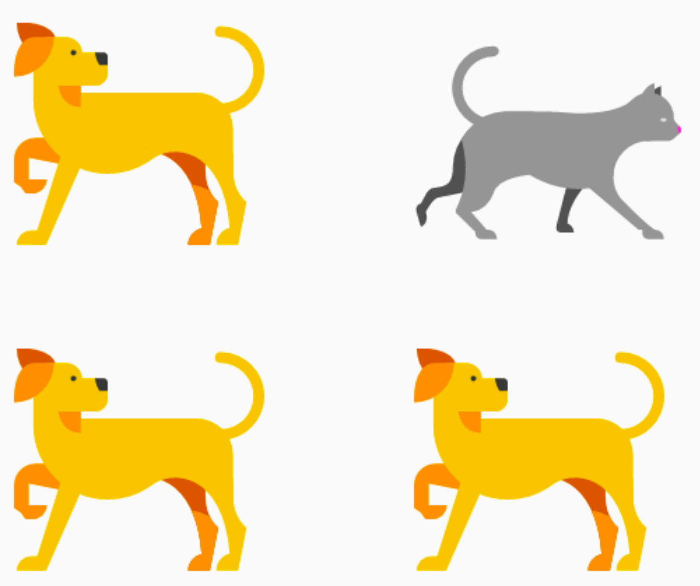


- **Question:** Is this A or B?
- **Answer: classification!**

- Example:
  - *Will this tire fail in the next 1,000 miles: Yes or no?*
  - *Which brings in more customers: a $5 coupon or a 25% discount?*

Five typical questions ML can answer

- **Question:** Is this normal?
- **Answer: anomaly detection** (one-versus-rest classification)!



- Example:
  - *Credit card fraud analysis*
  - *when purchase at a store that is geographically "weird"*

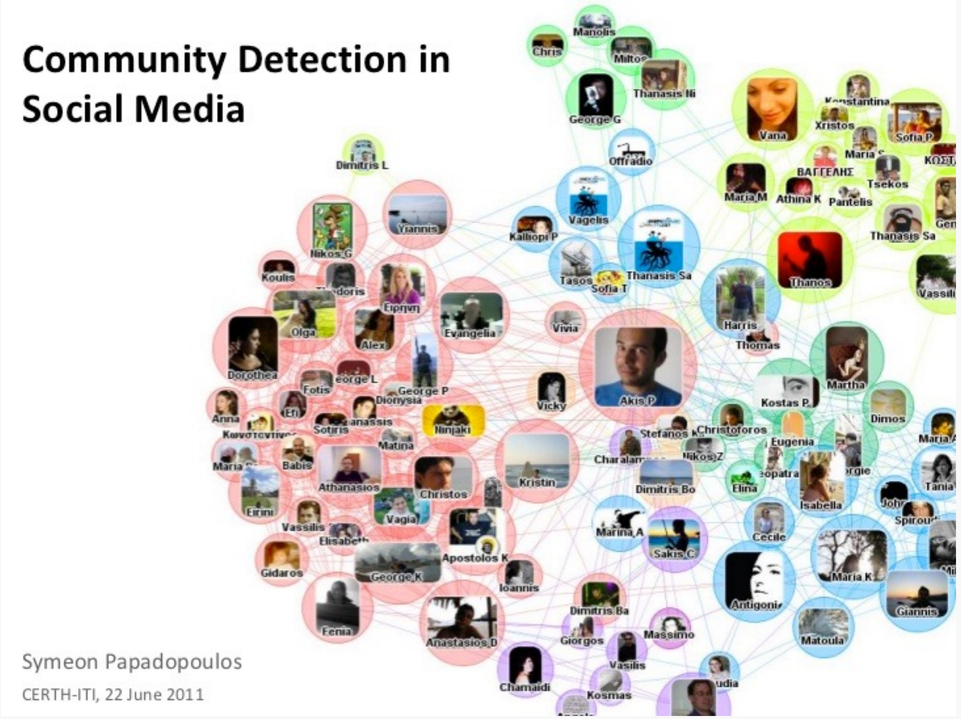
Five typical questions ML can answer

- **Question:** How many? How much?
- **Answer: regression analysis**!

- Example:
  - *How much should one offer for a 180m^2 area house?*

Five typical questions ML can answer

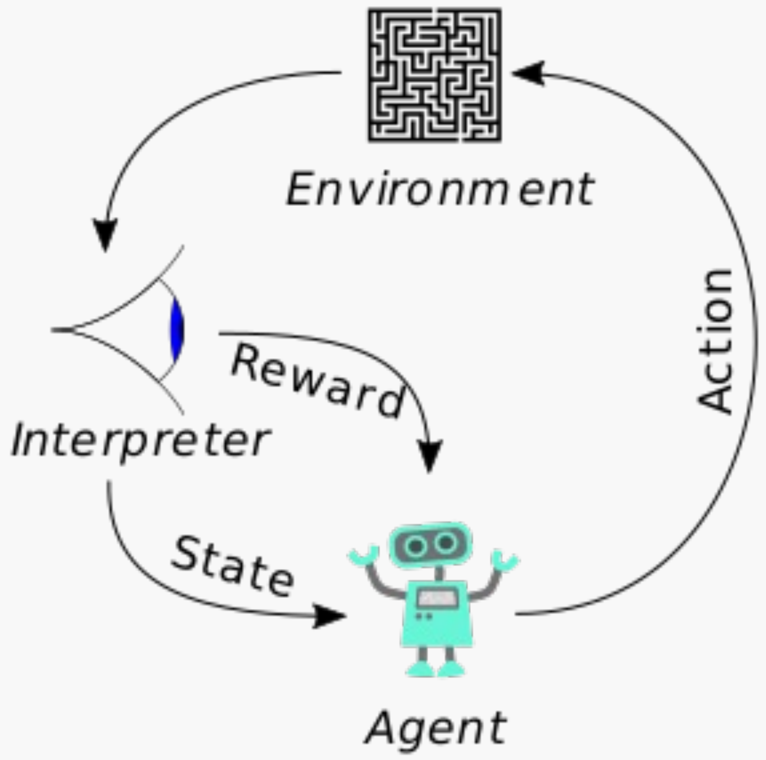
- **Question:** How is this organized? What is the structure of?
- **Answer: unsupervised methods, e.g., clustering, community detection and dimension reduction!**



- Example:
  - *What is the structure of your Facebook social network?*
  - *Which viewers like the same types of movies?*

Five typical questions ML can answer

- What should we do?
- Reinforcement learning!



- Example:
  - *for a self-driving car: at a yellow light, brake or accelerate?*

Outline

- Introduction and background of machine learning: rule-based and data-based
- Basics of machine learning
- Machine learning in practice: example of least-squares linear regression model
- More advanced considerations in machine learning
- Resources and references

Essence of machine learning

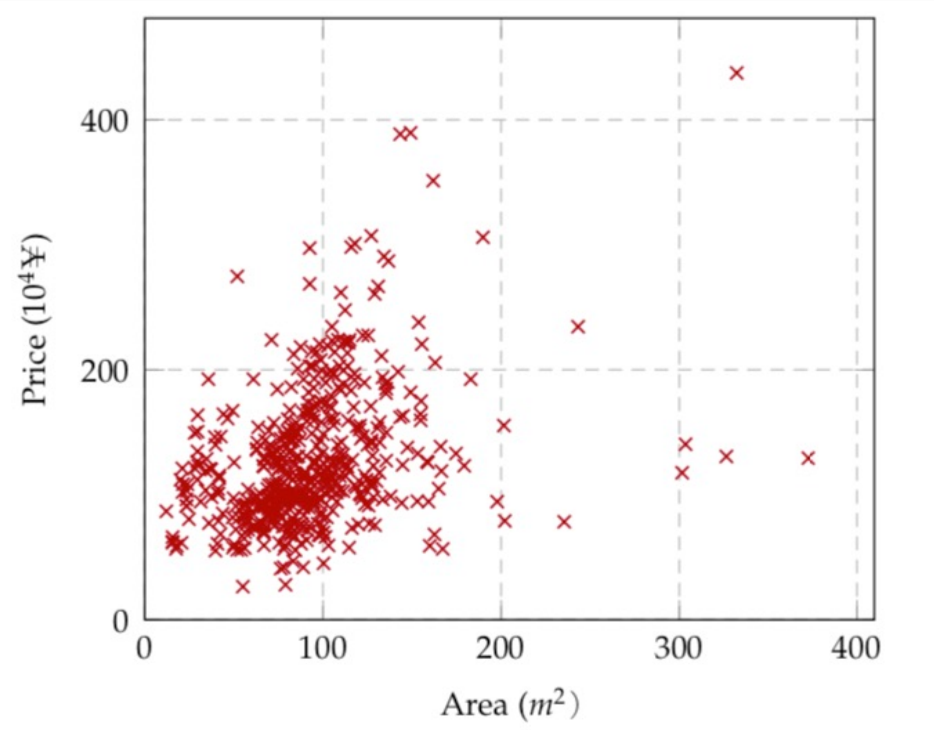
- **Build predictive model from historical data to make future decision**!
  1. *collect historical data*
  2. *choose appropriate model and train/fit it with data*
  3. *make prediction based on the well-trained model and make your decision!*

- **Three key steps** to successful machine learning:
  1. *collet good and massive amount of data*
  2. *which model to use: linear regression, logistic regression or deep learning?*
  3. *use (powerful) algorithms to train your model*

Example: housing prices prediction

- **Question:** for a $180\ m^2$ house, how much should we offer for?

1. collect historical data
2. choose your own model
3. fit the model with data (algorithm)
4. make prediction



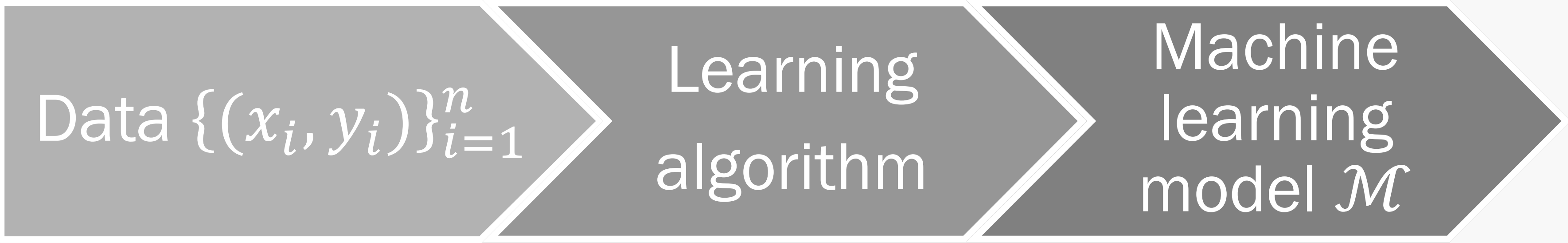
Data from: https://www.kaggle.com/alphaepsilon/housing-prices-dataset

Step 1+2: data and model

No.	Area (m^2): x	Price (10^4 RMB): y
1	78.50	145.95
2	88.72	98.00
3	93.68	214.90
…	…	…

- Get in total n (historical) data (area, price)
- x_i: data/feature (input)
- y_i: target (output)
- **Objective:**
  - for new input $\hat{x}$, predict its output $\hat{y}$

Flow diagram



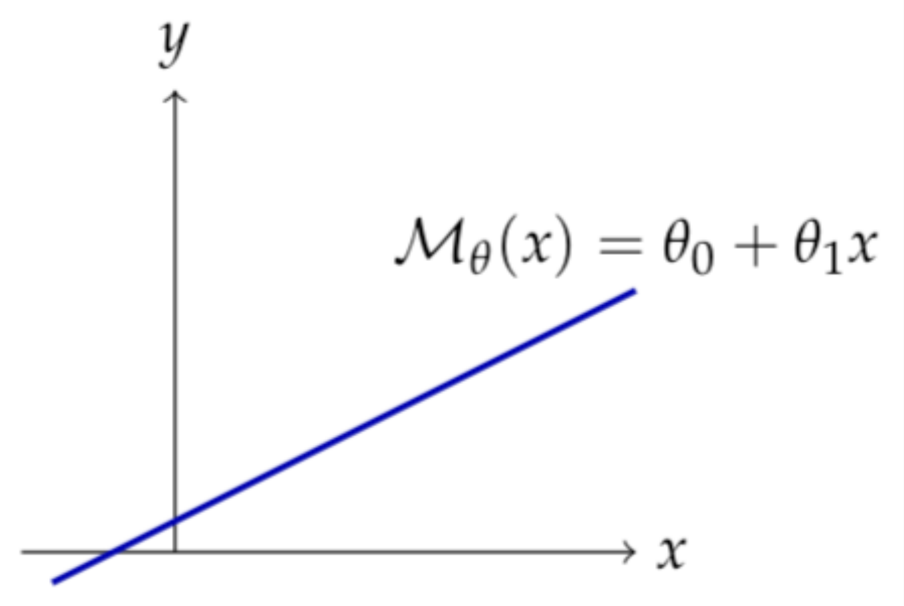
Making prediction: $\hat{x} \rightarrow \mathcal{M} \rightarrow \hat{y}$

Linear regression model

$$y: \quad \mathcal{M}_\theta(x) = \theta_0 + \theta_1 x$$

- describe a linear relation between input x and output y
- model $\mathcal{M}_\theta$ determined by (θ_0, θ_1)
- how to decide if a model $\mathcal{M}_\theta$ is good?

=><u>objective function</u>!



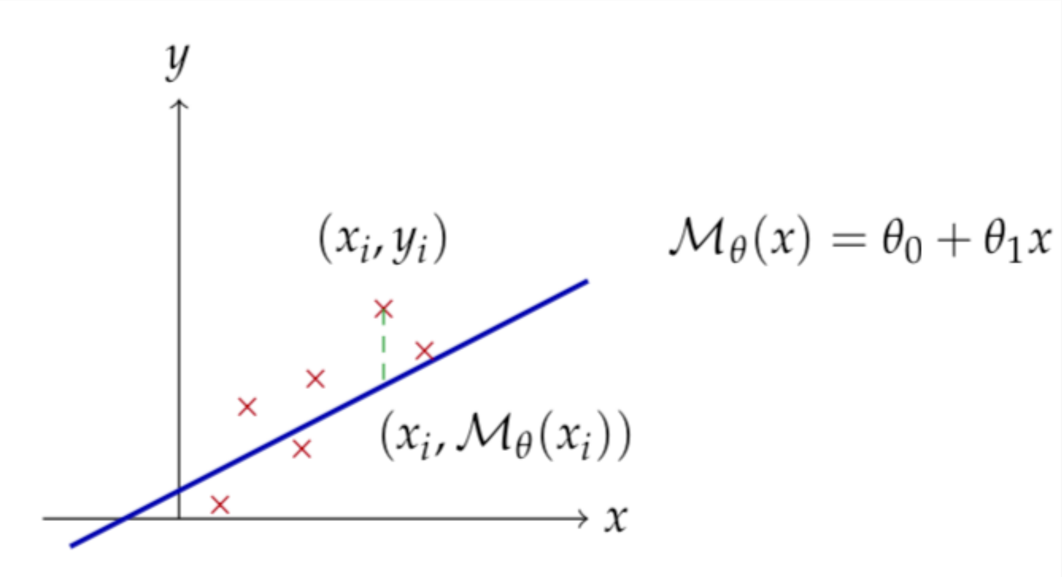
Mean squared error objective function

- linear regression $\mathcal{M}_\theta(x) = \theta_0 + \theta_1 x$, determined by <u>two parameters</u> (θ_0, θ_1)
- for data $\{(x_i, y_i)\}_{i=1}^n$, <u>**objective/loss function**</u> $L(\theta_0, \theta_1)$ tells us how model (parameter) fits the data

$$L(\theta_0, \theta_1) = \frac{1}{n}\sum_{i=1}^{n}(y_i - \mathcal{M}_\theta(x_i))^2$$

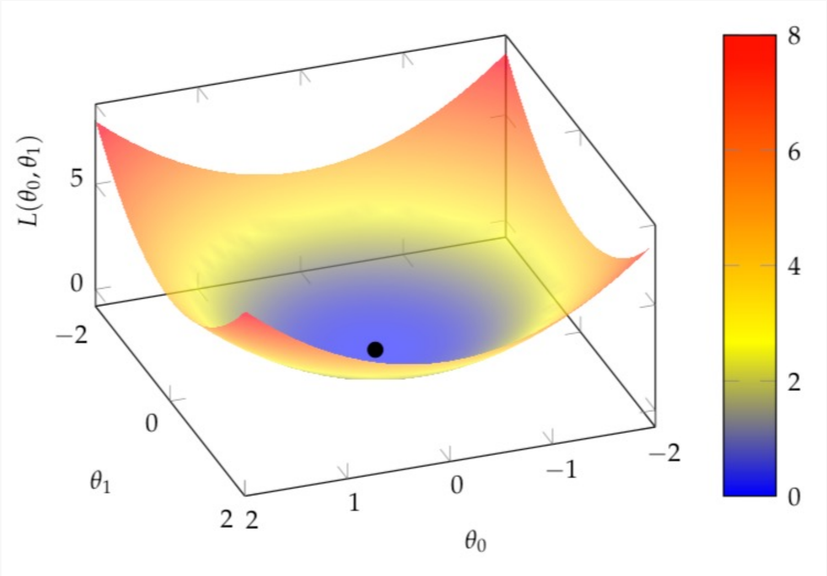
$$= \frac{1}{n}\sum_{i=1}^{n}(y_i - \theta_0 - \theta_1 x_i)^2$$

- $L(\theta_0, \theta_1) \uparrow$, **larger difference** between model output and historical data
- find the best $(\theta_0, \theta_1) \Longrightarrow \min_{\theta_0, \theta_1} L(\theta_0, \theta_1)$



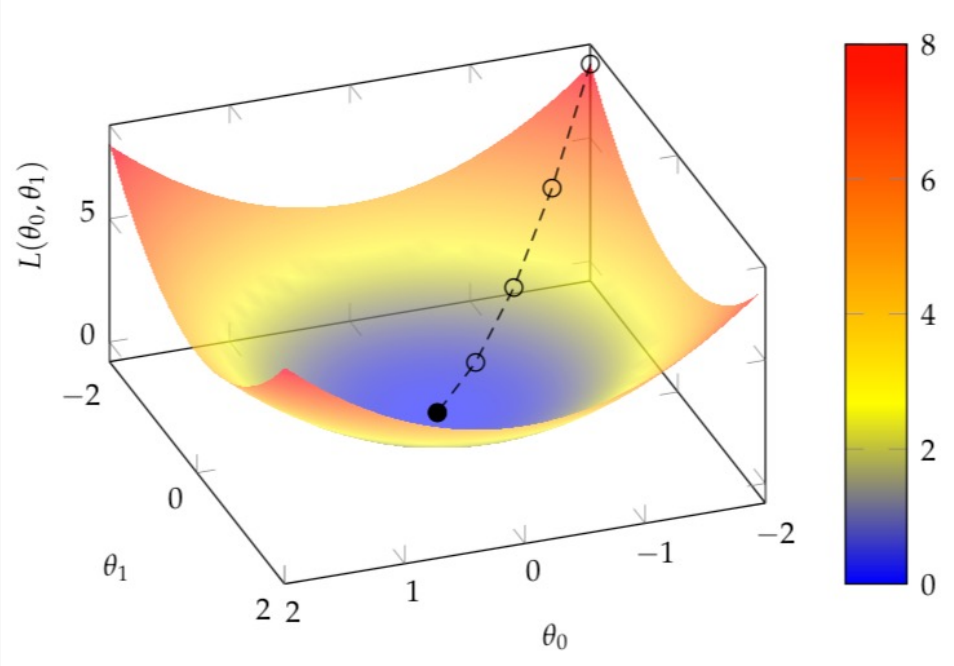
Learning model by minimization

- For a given dataset $\{(x_i, y_i)\}_{i=1}^{n}$, plot $L(\theta_0, \theta_1)$ as follows



- wish to reach the bottom (black) point of $L(\theta_0, \theta_1)$, get the corresponding (θ_0^*, θ_1^*) value

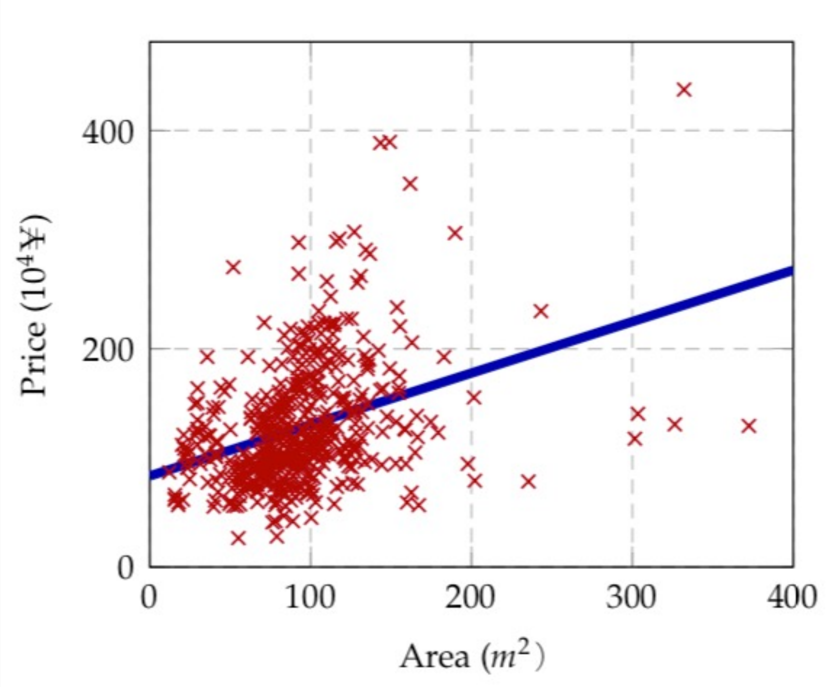
Learning model with gradient descent



Algorithm

- Start from any point (θ_0^0, θ_1^0), **update** (θ_0^k, θ_1^k) to minimize $L(\theta_0, \theta_1)$, to reach the minimum (θ_0^*, θ_1^*): $L(\theta_0^{k+1}, \theta_1^{k+1}) \leq L(\theta_0^k, \theta_1^k)$.
- iterations update according to $\theta_j^{k+1} = \theta_j^k - \alpha \nabla_{\theta_j} L(\theta_0, \theta_1), j = 0,1$, until convergence.

Example: housing prices prediction



$$\Longrightarrow \theta_0^* = 83.5384, \theta_1^* = 0.4734.$$

- For a 180 m^2 house, our model predicts $\mathcal{M}_{\theta^*}(180) = \theta_0^* + 180\ \theta_1^* =$ 168.75×10^4RMB.
- **Interpretation**: price= θ_0^*(for all)+ θ_1^*×area.

Sum-up and further discussions

- Three steps of machine learning:
  1. *define your **model***
  2. *choose your **metric** (objective function)*
  3. *train/**optimize** the model and make prediction!*
- Here we discuss the example of
  1. *linear regression model: input-output linear relation*
  2. *mean squared error: for regression problem*
  3. *gradient descent optimization method*
- Many many more choices:
  1. *for nonlinear input-output relation: polynomial regression, SVM, neural nets*
  2. *for classification problem: 0-1 error, cross-entropy error, hinge loss, etc.*
  3. *more and more advanced optimization methods*

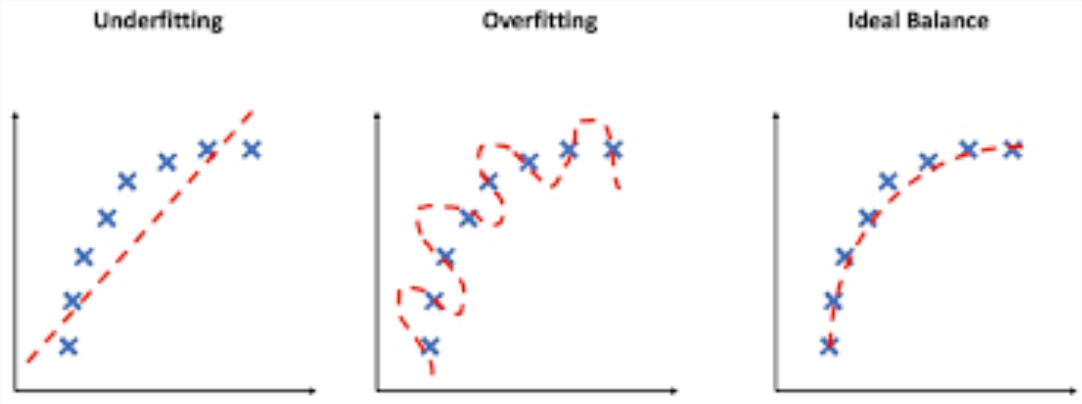
Outline

- Introduction and background of machine learning: rule-based and data-based
- Basics of machine learning
- Machine learning in practice: example of least-squares linear regression model
- More advanced considerations in machine learning
- Resources and references

About model fitting

Training is important, but what really matters is <u>test and future prediction</u>!

- **Model** versus **task** complexity: over- and under-fitting

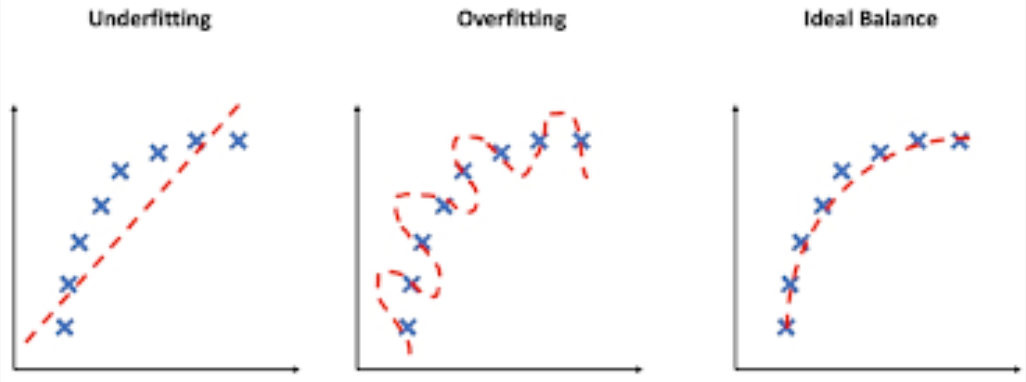


- Example: real data can be noisy
  - *a underlying <u>simple</u> model (quadratic function* $y = -x^2 + 10x + 1$*) + small noise*
  - *if fit with a too <u>complex</u> function (cubic function* $y = \theta_3 x^3 + \theta_2 x^2 + \theta_1 x^1 + \theta_0$*)*
- **bias** and **variance** trade-off

About model fitting

Training is important, but what really matters is <u>**test and future prediction**</u>!

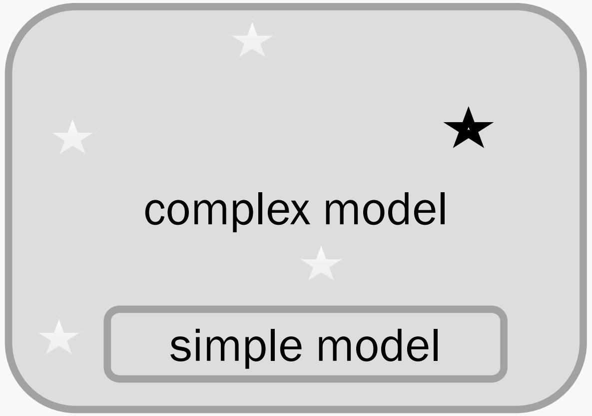
- **Model** versus **task** complexity: over- and under-fitting, the bias and variance trade-off
- Task: $-x^2 + 10x + 1$ + small noise
- Model:
  - $\theta_1 x^1 + \theta_0$ *too simple*
  - $\theta_3 x^3 + \theta_2 x^2 + \theta_1 x^1 + \theta_0$ *too complex*
  - *good fit* $\theta_2 x^2 + \theta_1 x^1 + \theta_0$

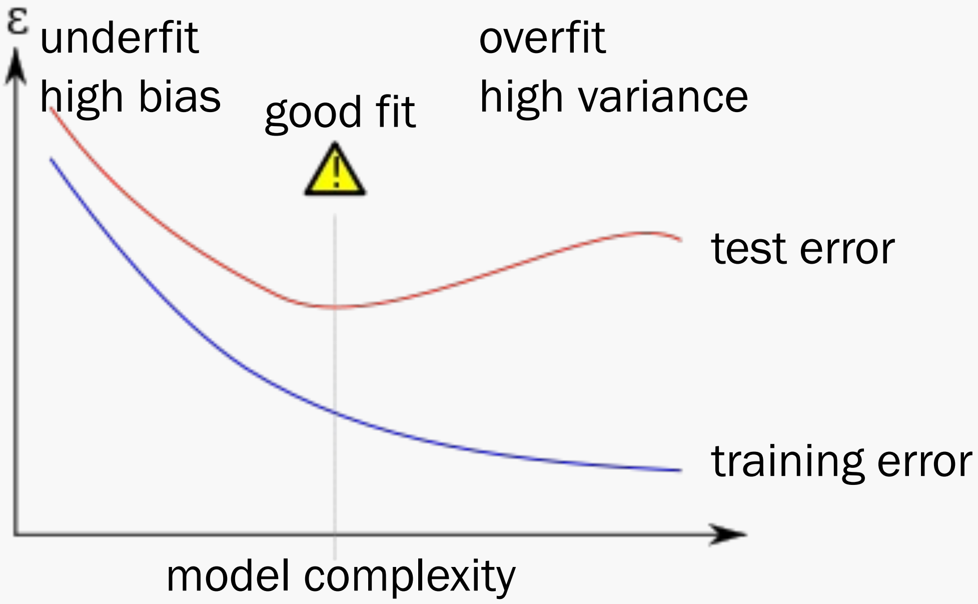


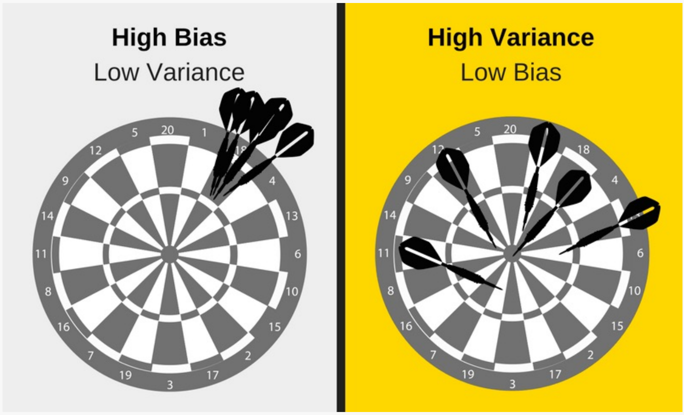
	Training error	Test error	Bias	Variance
Underfitting	large	large	large	small
Overfitting	small (≈ 0)	large	small	large
Good fit	small	small	small	small

About model fitting

- **Model** versus **task** complexity: bias and variance trade-off
  - *model too simple: **unable** to fit the training data*
    - high bias between model and target
  - *model too complex: contains **many sub-models** almost perfectly fit the data*
    - high variance between sub-models

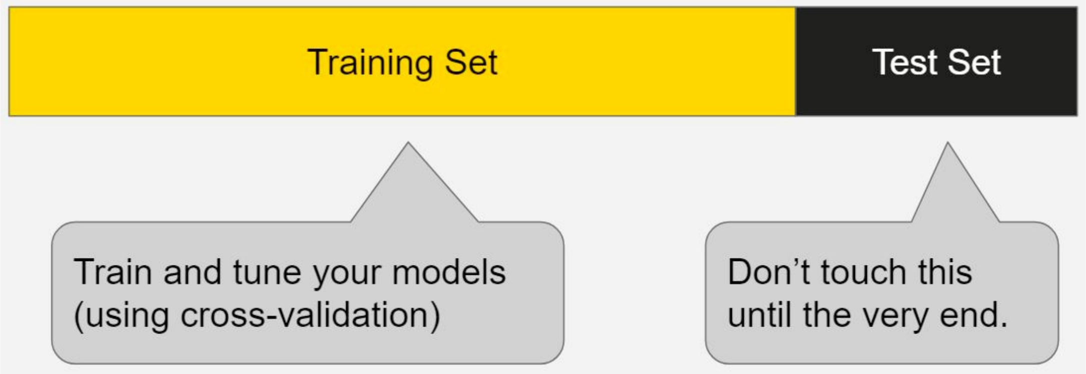




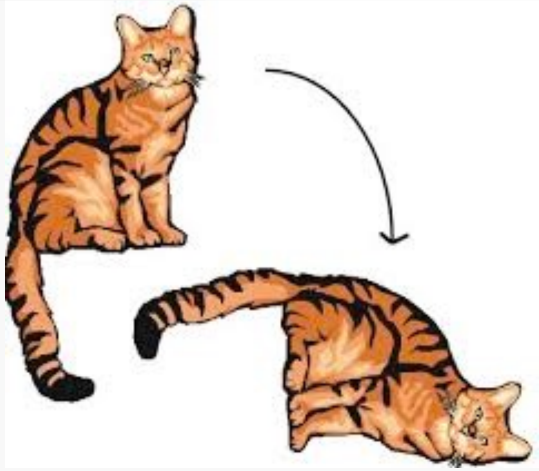


About model fitting

- To avoid overfitting: training, cross-validation and test sets



- regularization with a prior knowledge
- model ensembling

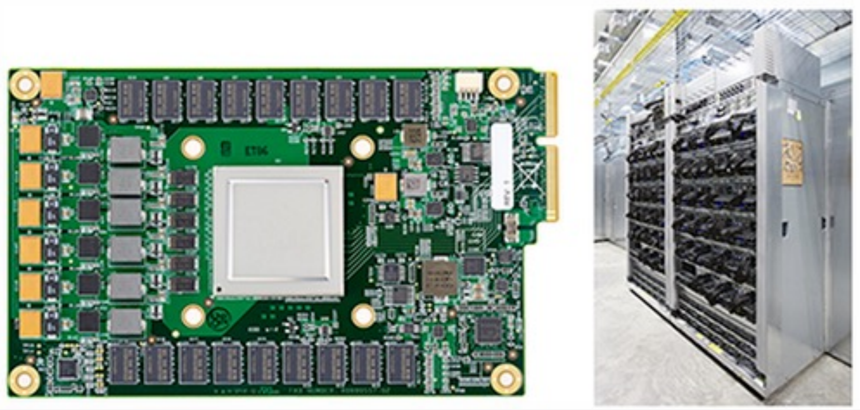


Doing SotA ML can be expensive

- State-of-the-Art (SotA) Machine Learning models for various applications:

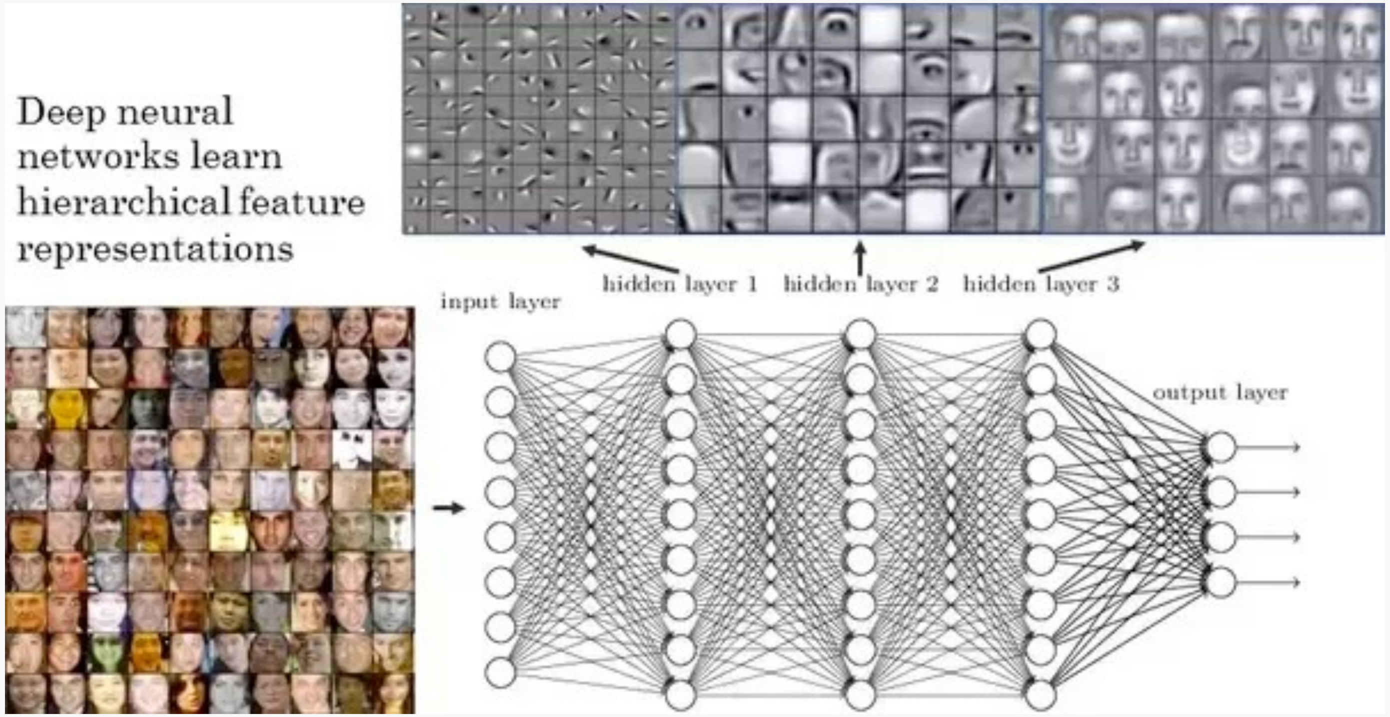
Applications	Algorithms	Accuracies	Size	Time
CV: multi-class (10,000+) image classification	PolyNet (2017)	91.3% (top 1) 95.75% (top 5) almost human	92 M	3700 hours in total
NLP: question answering	BERT	93.2 Test F1	340M	100 hours in Google
RL: game playing	AlphaGo Lee (2016)	superhuman		4-6 weeks in Google

- source: https://www.stateoftheart.ai/
- Tensor Processing Unit (TPU) of Google



Good news: transfer learning

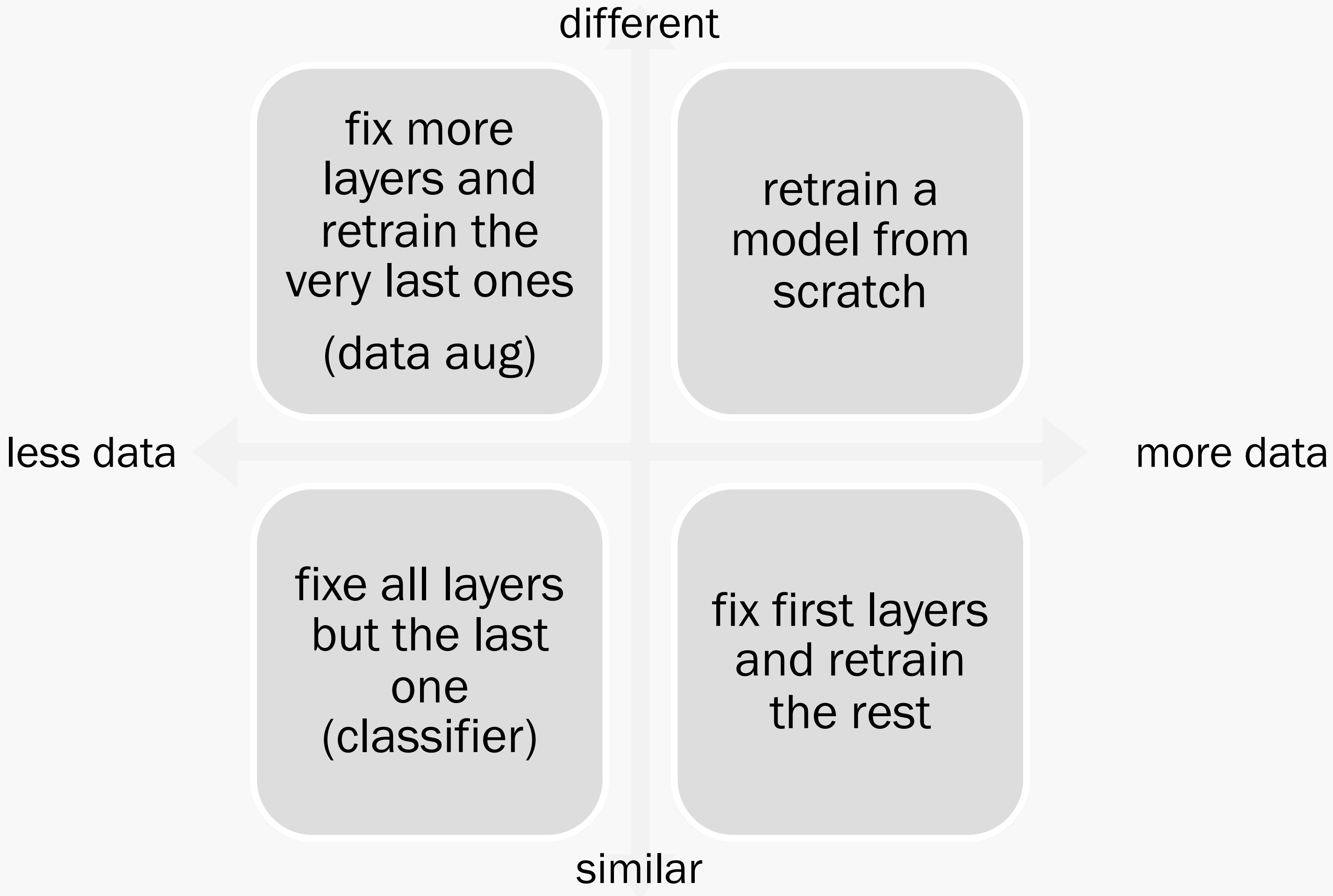
- Transfer existing knowledge/model/structure to solve my problem!
- Example: using pre-trained model for image classifcation, with Microsoft ResNet150 (trained on ImageNet dataset http://www.image-net.org/)



- Edges --- contour --- high-level features --- classifier

Good news: transfer learning

- Transfer existing knowledge/model/structure to solve my problem!



ML in practice: further needs

- **Confidence interval:** statistical estimates or Bayes methods
  - *security services, autonomous vehicle, how much can we trust the model?*
- **Model interpretability:** much harder for large-scale problems
  - *investment, to customer bussiness, why should we make this decision?*

Outline

- Introduction and background of machine learning: rule-based and data-based
- Basics of machine learning
- Machine learning in practice: example of least-squares linear regression model
- More advanced considerations in machine learning
- Resources and references

ML in practice: datasets

- Popular publics datasets for MLers:
  - *UCI machine learning repository:* *https://archive.ics.uci.edu/ml/datasets.php*
  - *Kaggle datasets and competitions:* *https://www.kaggle.com/datasets*
  - *https://en.wikipedia.org/wiki/List_of_datasets_for_machine-learning_research*
  - *Scikit-learn:* *https://scikit-learn.org/stable/datasets/index.html*

ML in practice: various frameworks

- Tensorflow https://www.tensorflow.org/
- Pytorch https://pytorch.org/
- Scikit-learn https://scikit-learn.org/stable/
- all based on Python
- many others…

ML in practice: online courses

- Udacity Introduction to machine learning: https://www.udacity.com/course/intro-to-machine-learning--ud120
- Coursera Machine Learning: https://www.coursera.org/learn/machine-learning
- edX Machine Learning: https://www.edx.org/course/machine-learning
- Stanford CS231n Convolutional Neural Networks for Visual Recognition: https://www.youtube.com/playlist?list=PL3FW7Lu3i5JvHM8ljYj-zLfQRF3EO8sYv

ML in practice: references

- More math/stats and theory textbooks:
  - *Pattern Recognition and Machine Learning*
  - *Deep Learning 2015: http://www.deeplearningbook.org/*

- More hands-on and application-oriented:
  - *Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems*
  - *Deep Learning with Python*