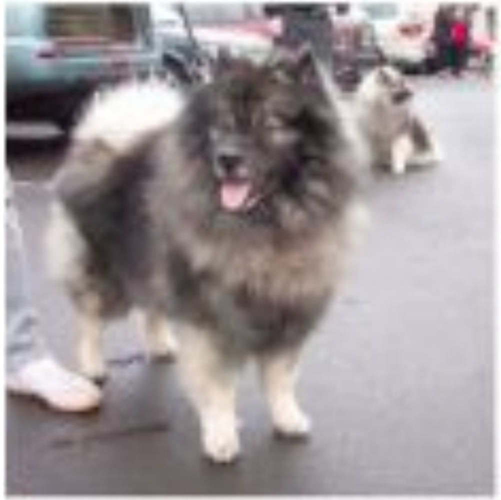
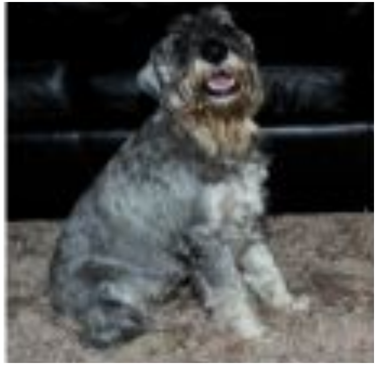
Deep Learning: Applications and Limitations

Cho-Jui Hsieh
Assistant Professor
UCLA Computer Science

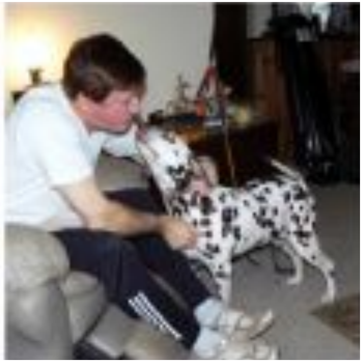
How to make the machine learn from data?



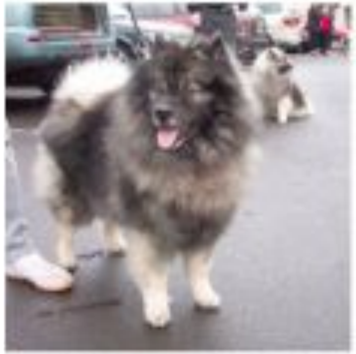
a) Keeshond
b) Mini Schnauzer
c) Giant Schnauzer
d) Dalmatian



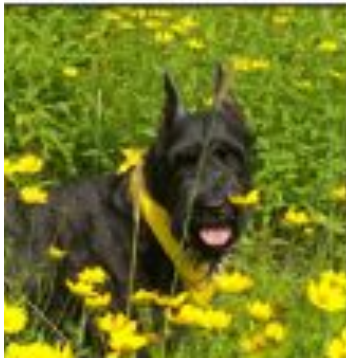
Schnauzer



Dalmatian



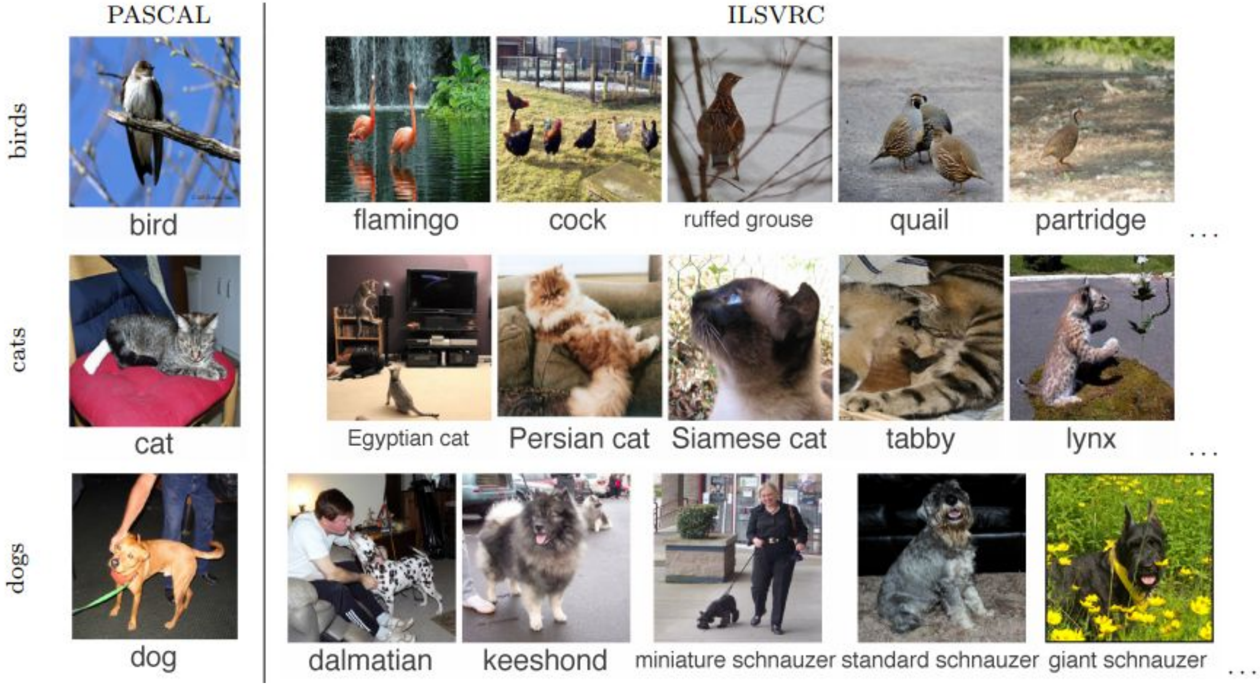
Keeshond



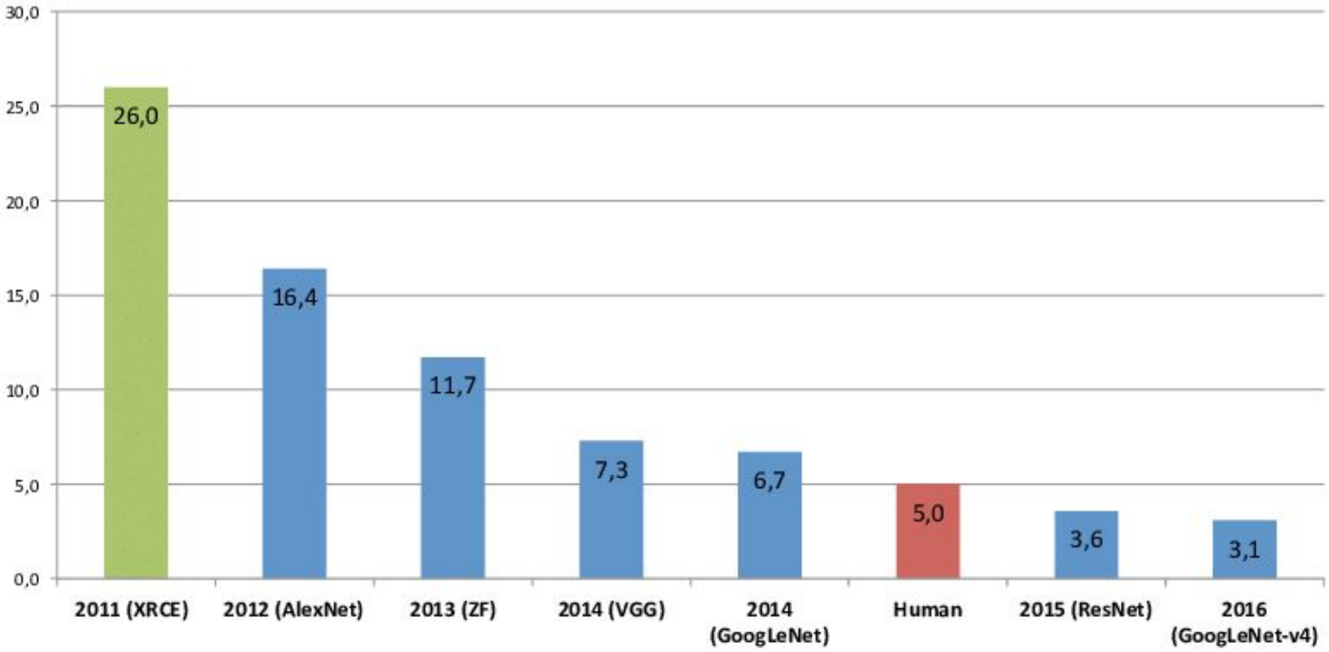
Giant Schnauzer

ImageNet Challenge

- 1.5 million images, 1000 classes

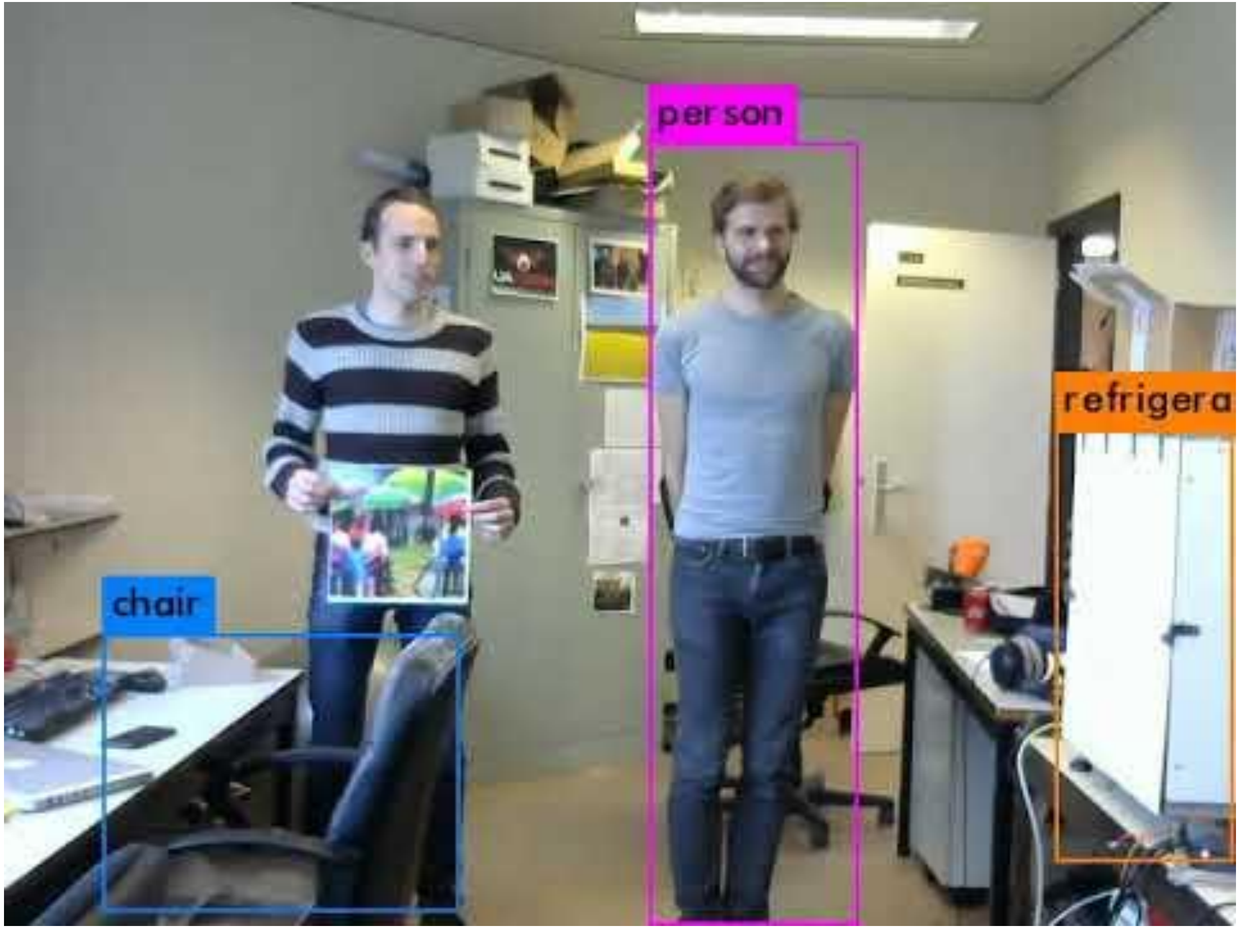


Machine Learning for ImageNet Classification



Prediction error: **Outperform human performance**

But...



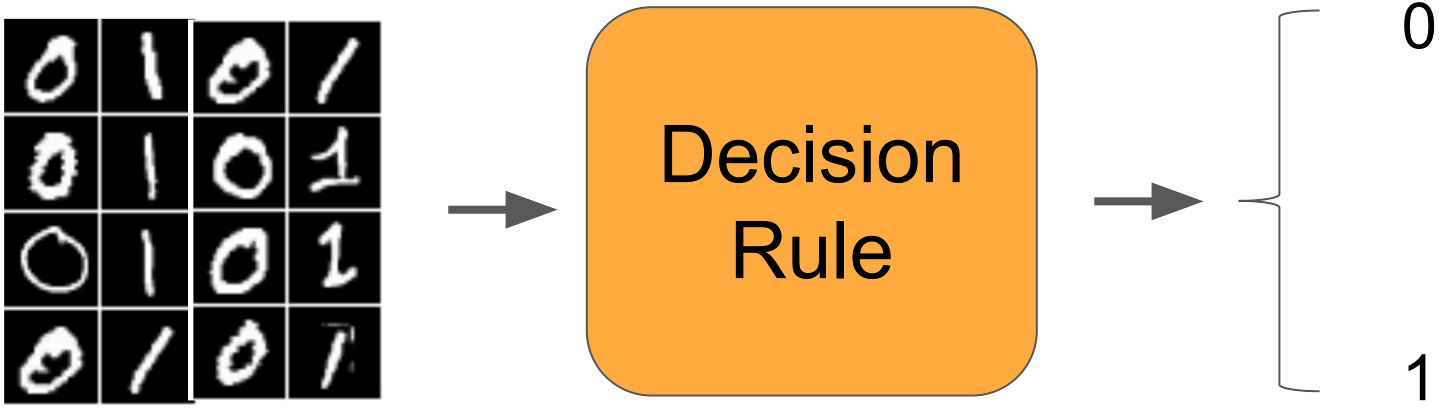
Roadmap

- What is machine learning?
- Deep Neural networks
- Challenges / Research topics

Roadmap

- **What is machine learning?**
- Deep Neural networks
- Challenges / Research topics

Example: Image Classification



Decision Rule

0

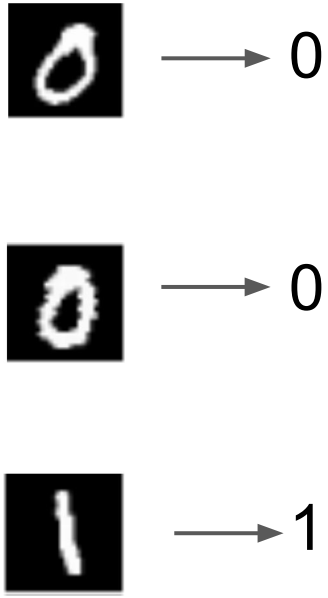
1

Hand written digits

How to learn the **decision rule**?

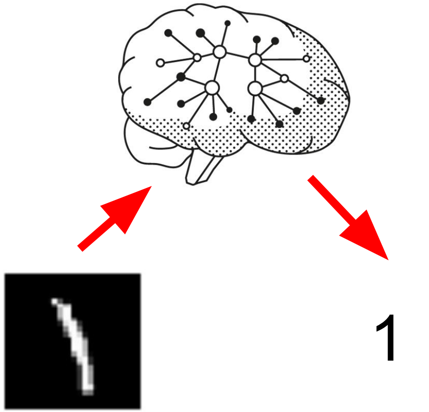
Human learning

Observation



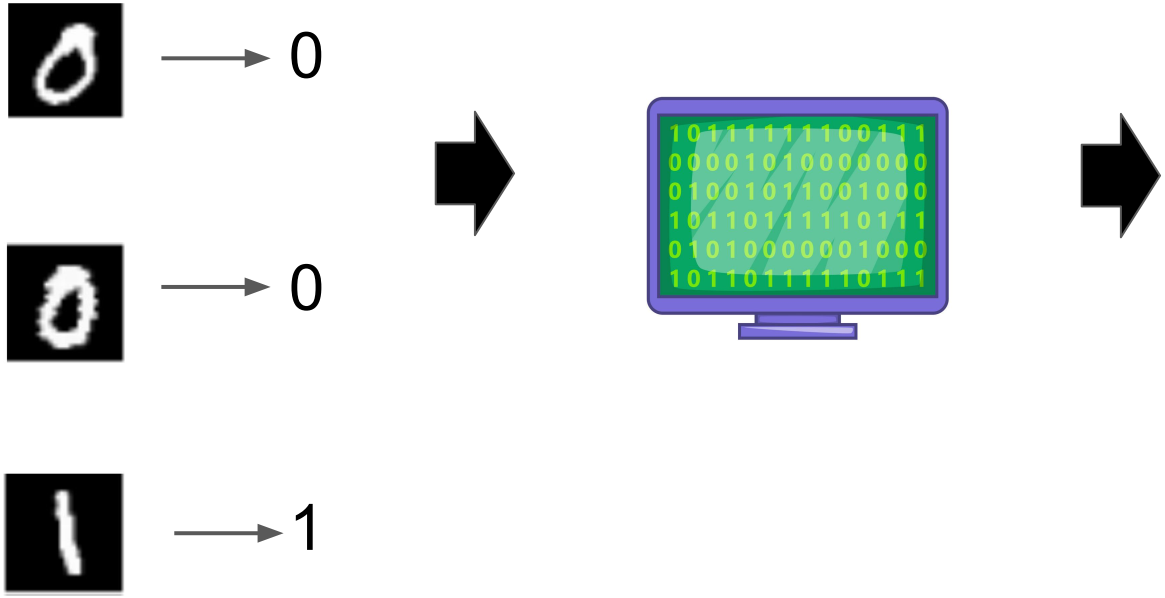
Learning

Decision rule



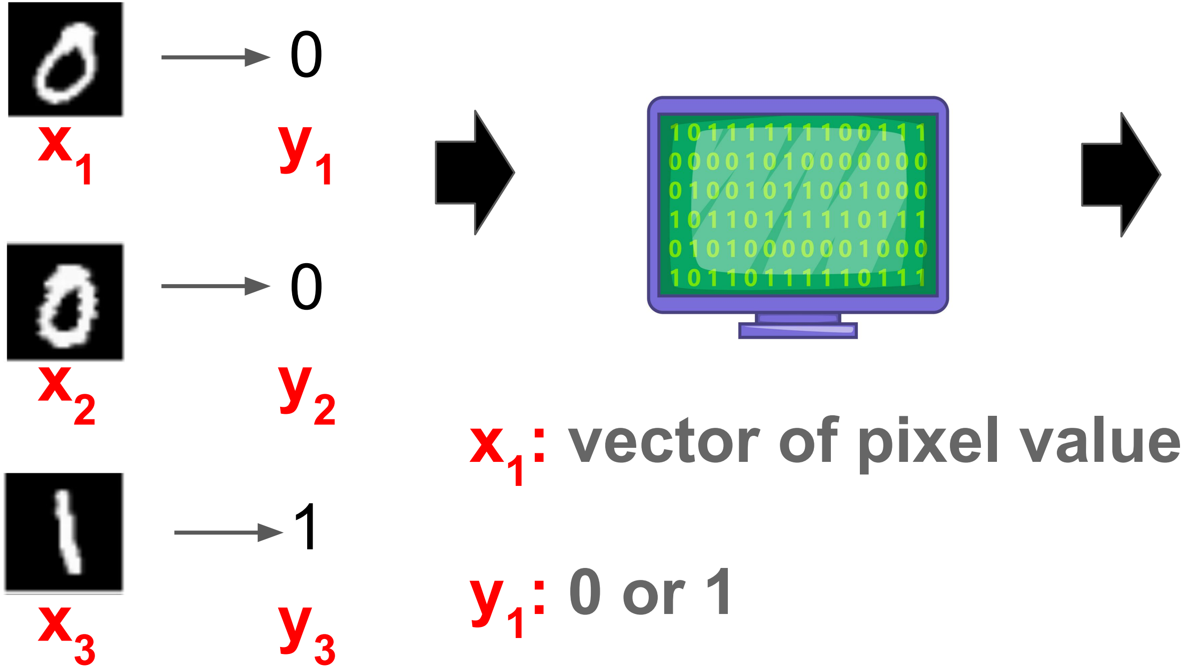
Machine learning

Training Data **Machine Learning** **Decision rule**



Machine learning

Training Data **Machine Learning** **Decision rule**



$x_1 \longrightarrow 0$ y_1

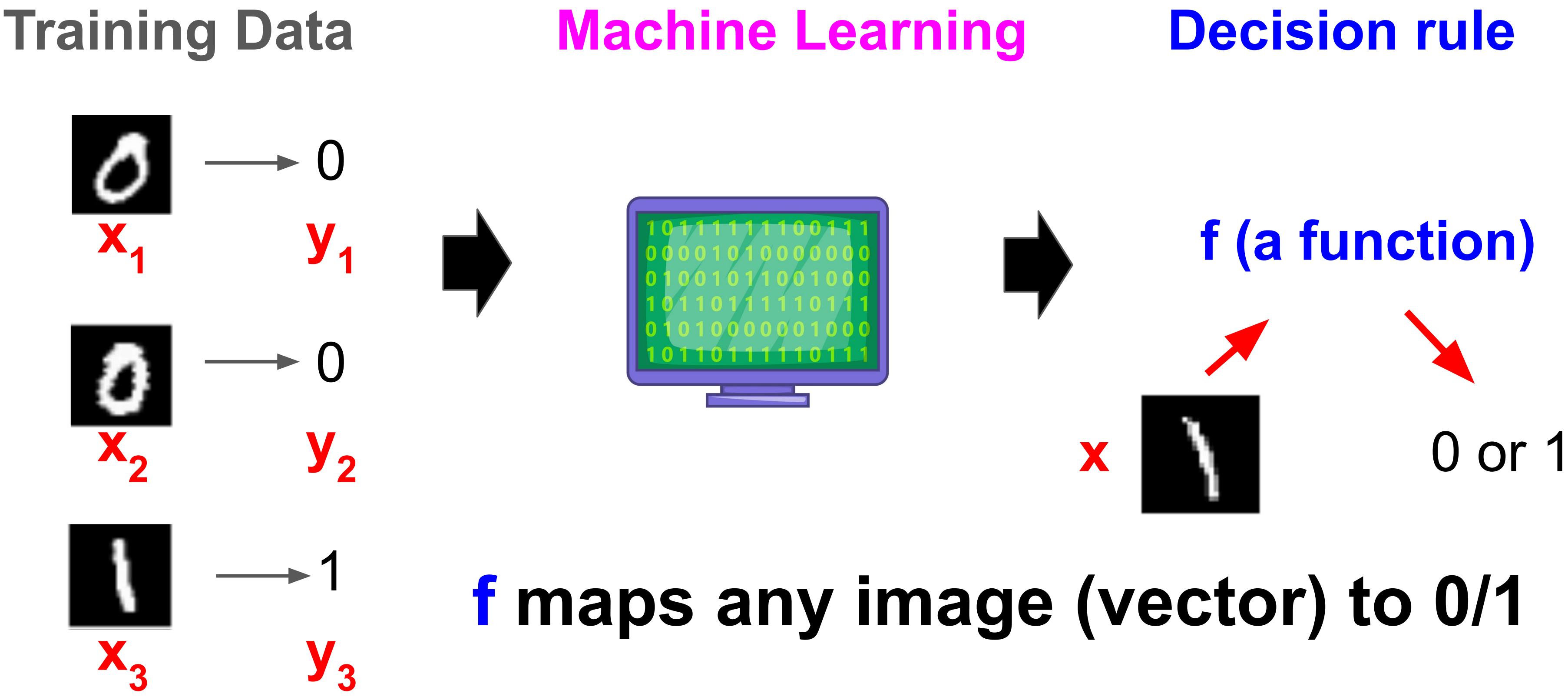
$x_2 \longrightarrow 0$ y_2

$x_3 \longrightarrow 1$ y_3

x_1: vector of pixel values [0, 24, 128, ...]

y_1: 0 or 1

Machine learning



Training Data

Machine Learning

Decision rule

x_1 → 0 y_1

x_2 → 0 y_2

x_3 → 1 y_3

f (a function)

x

0 or 1

f maps any image (vector) to 0/1

Decision function (model)

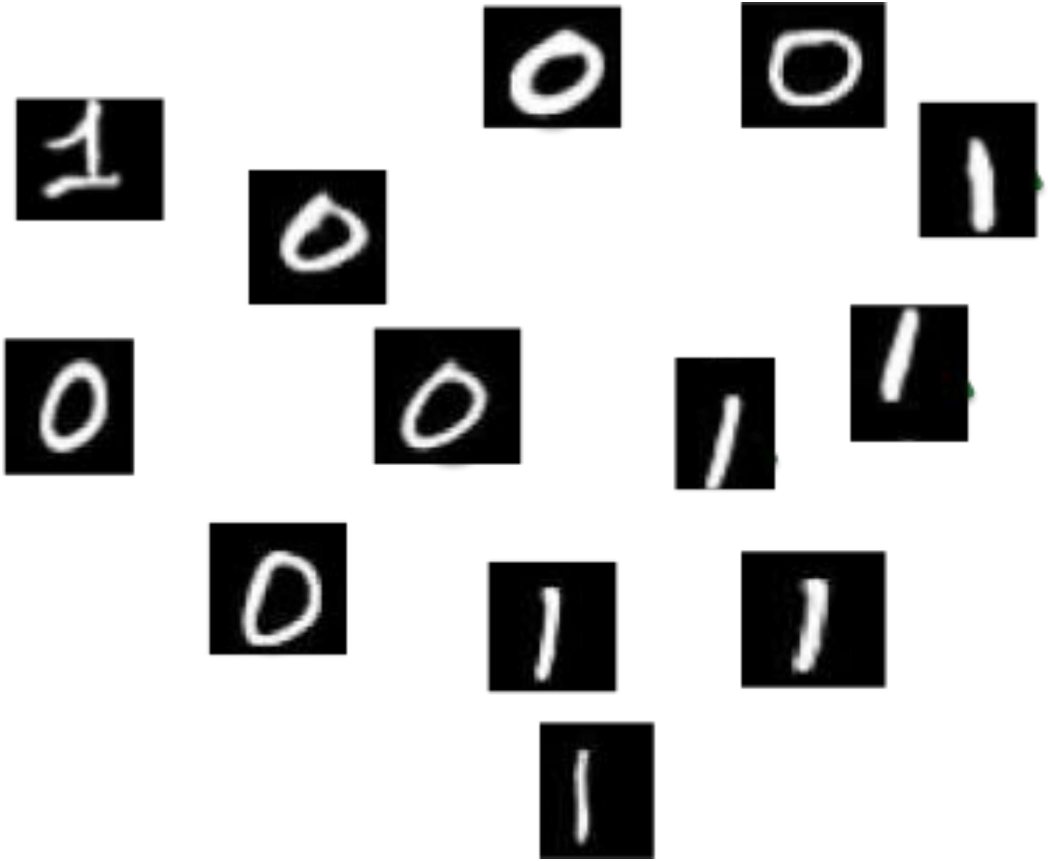
- Machine learning:

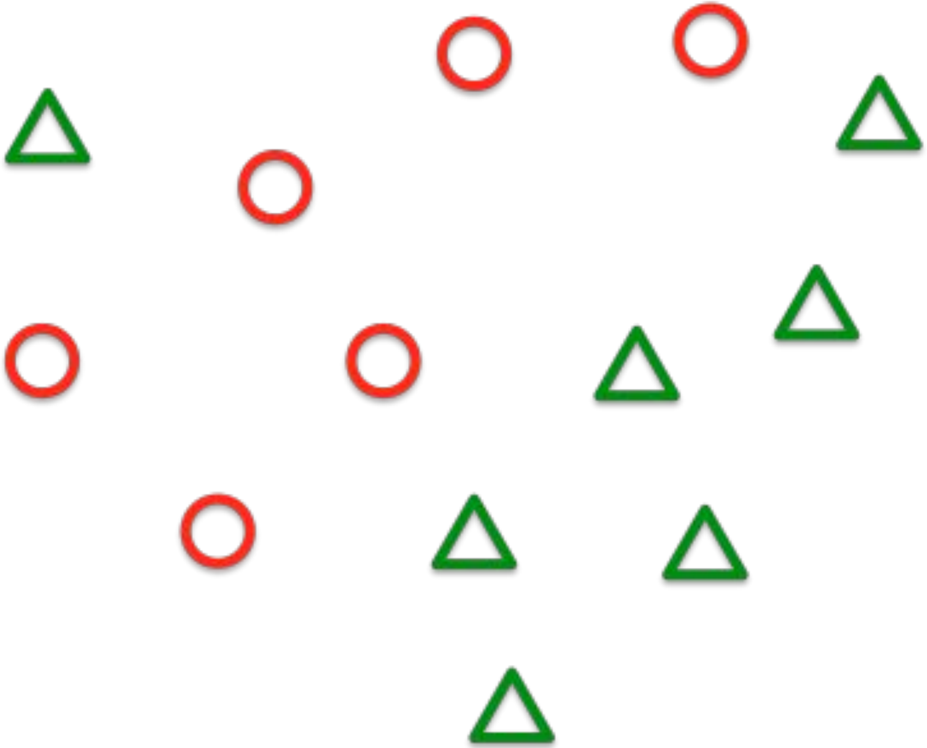
  Find the best function to map input to output (according to training data)

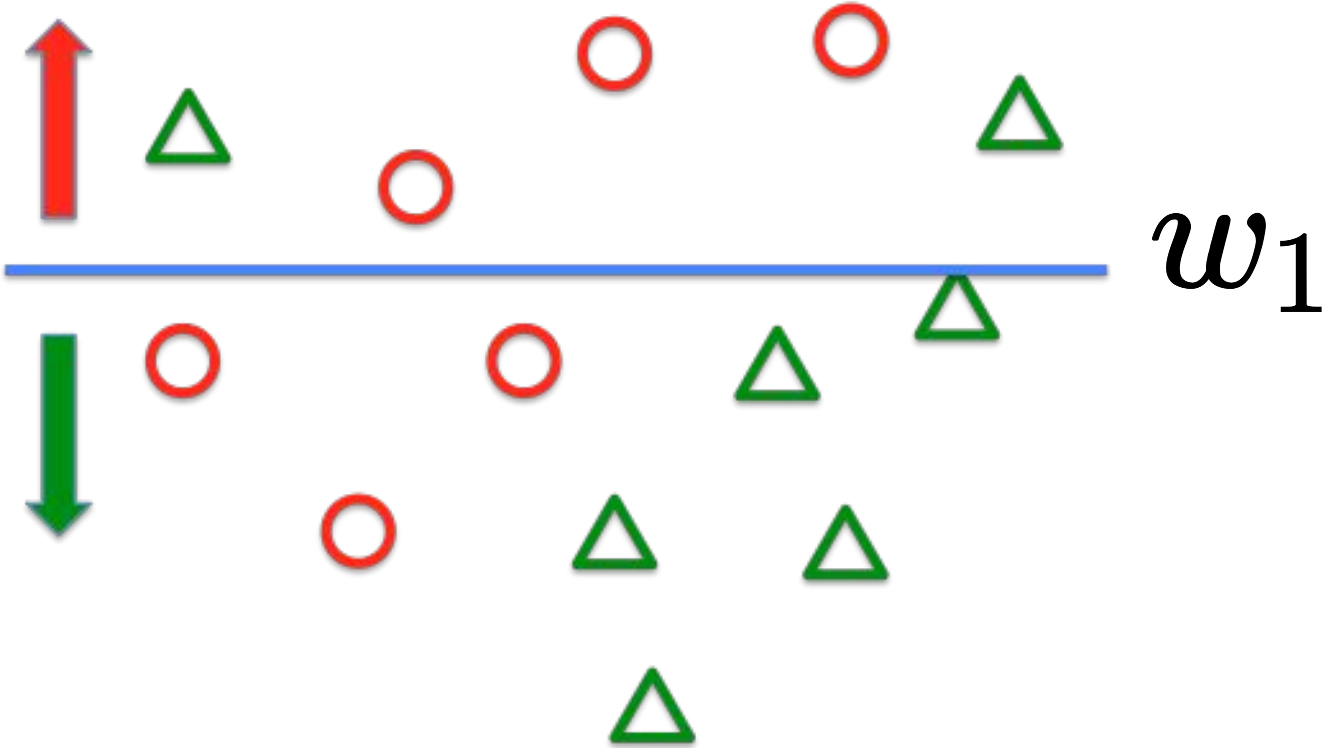
- Let’s assume we use **Linear function**:

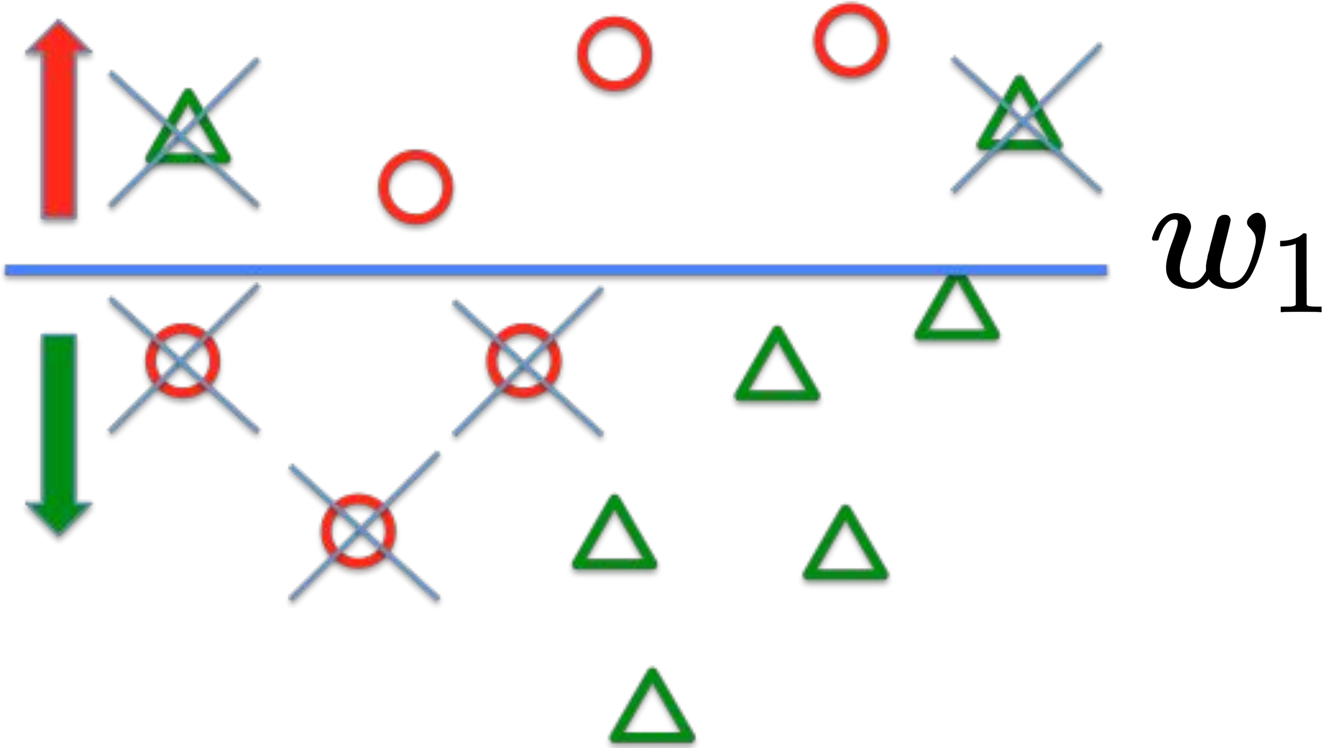
$$f_w(x) = w^T x = \sum_i w_i x_i$$

- Our goal is to find the best **w**

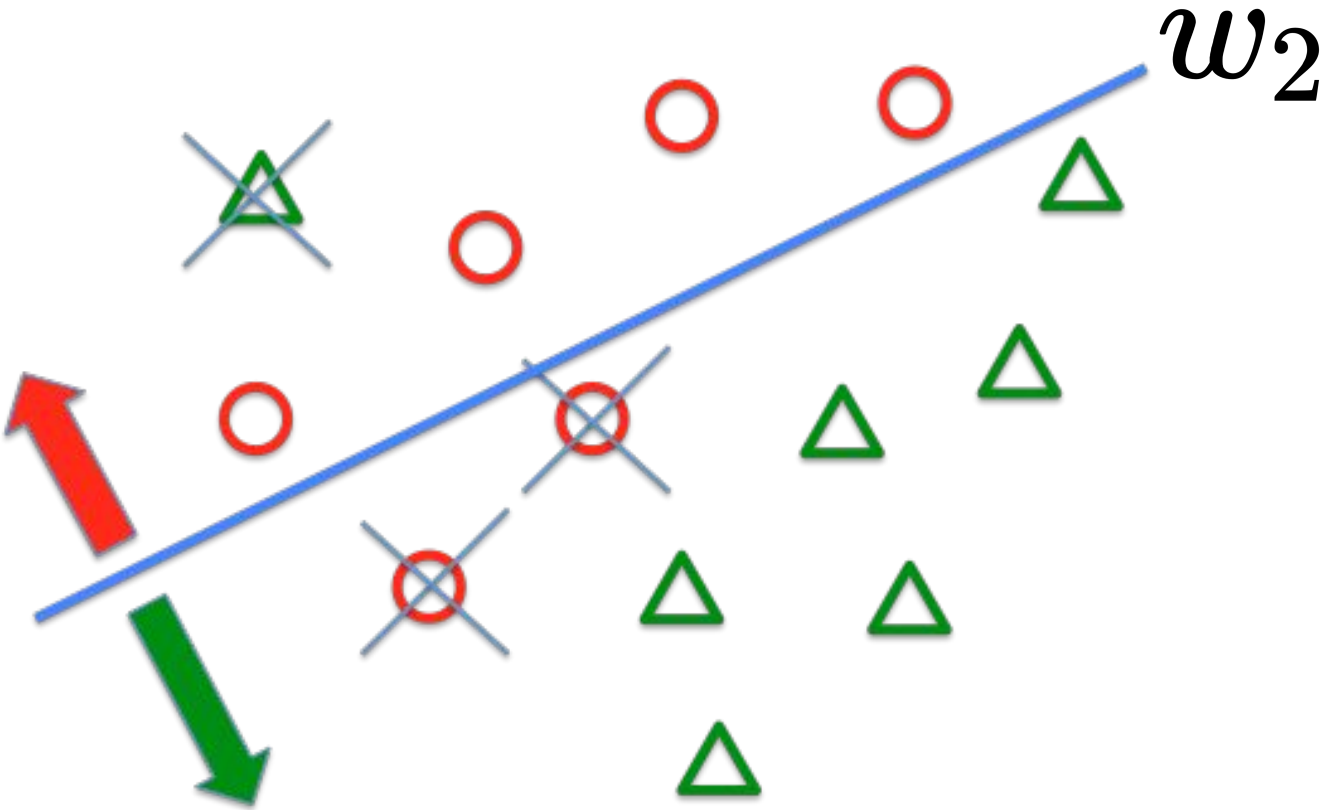




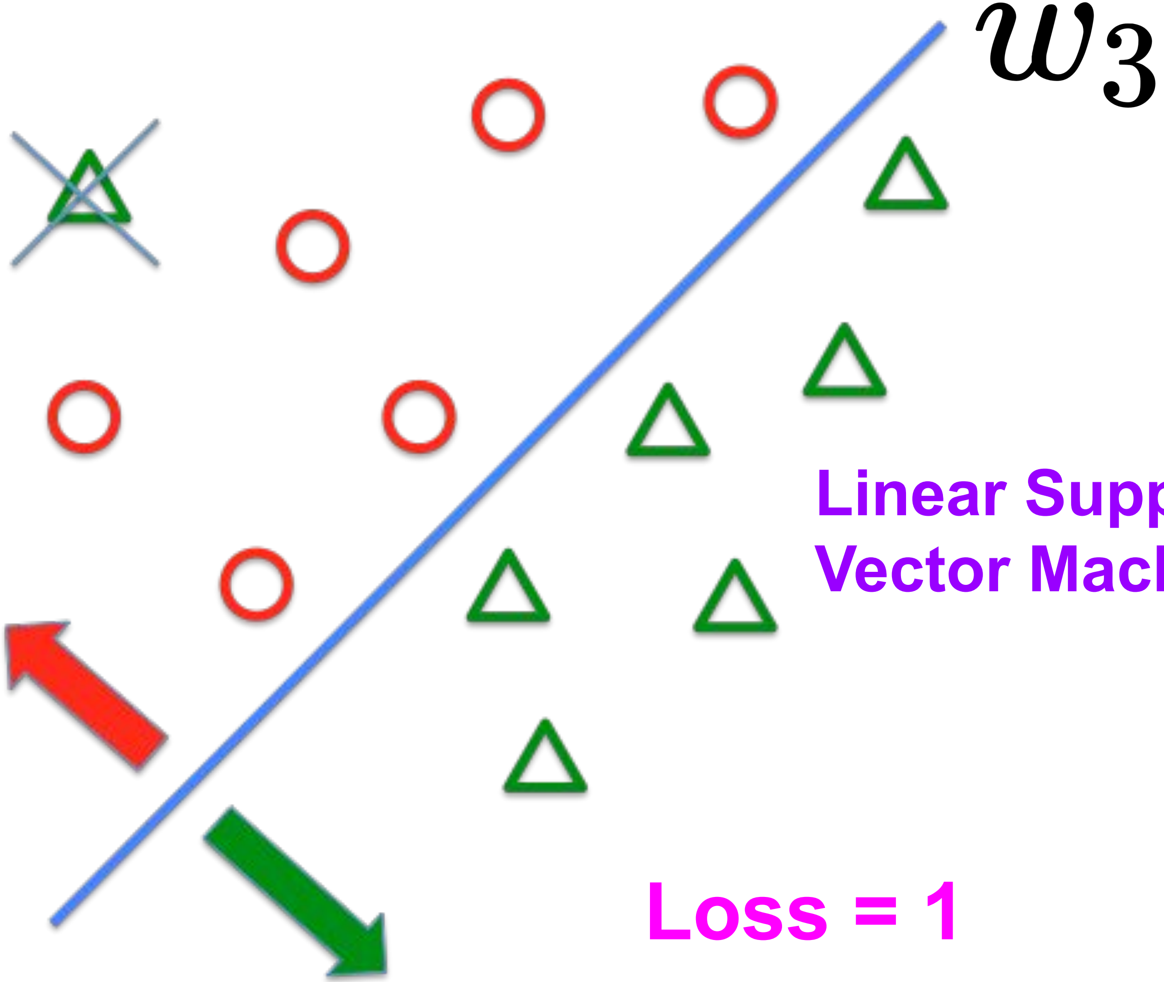

w_1



Loss = 5



Loss = 3



Linear Support Vector Machine (SVM)

Loss = 1

Classification

- Given training data D = $(x_1, y_1), \ldots, (x_n, y_n)$

Which function should we pick?

Pick the one with smallest loss on training data

How good is f on training data?

- Error on this data:

$$\operatorname{loss}(f(x), y) = \begin{cases} 0 & \text{if } f(x) = y \\ 1 & \text{if } f(x) \neq y \end{cases}$$

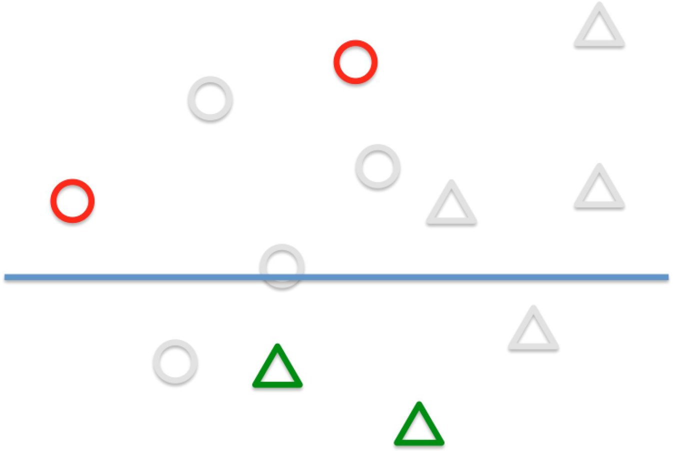
- Training error (on all training data):

$$\textstyle\sum_x \operatorname{loss}(f(x), y)$$

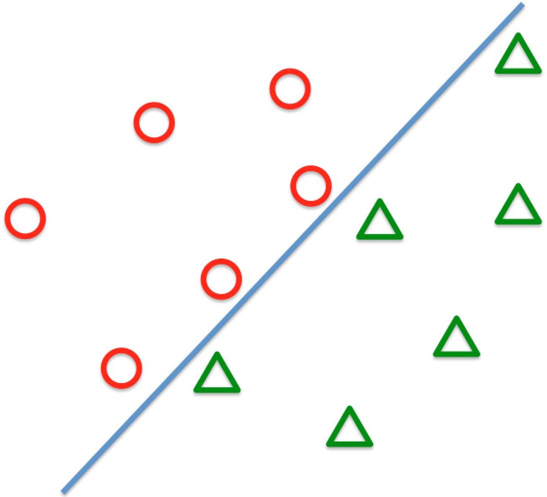
- **Machine learning = optimization**

$$\textstyle\min_f \sum_x \operatorname{loss}(f(x), y)$$

More data -> better performance



Not enough data =>
bad hyperplane



More data => better estimation

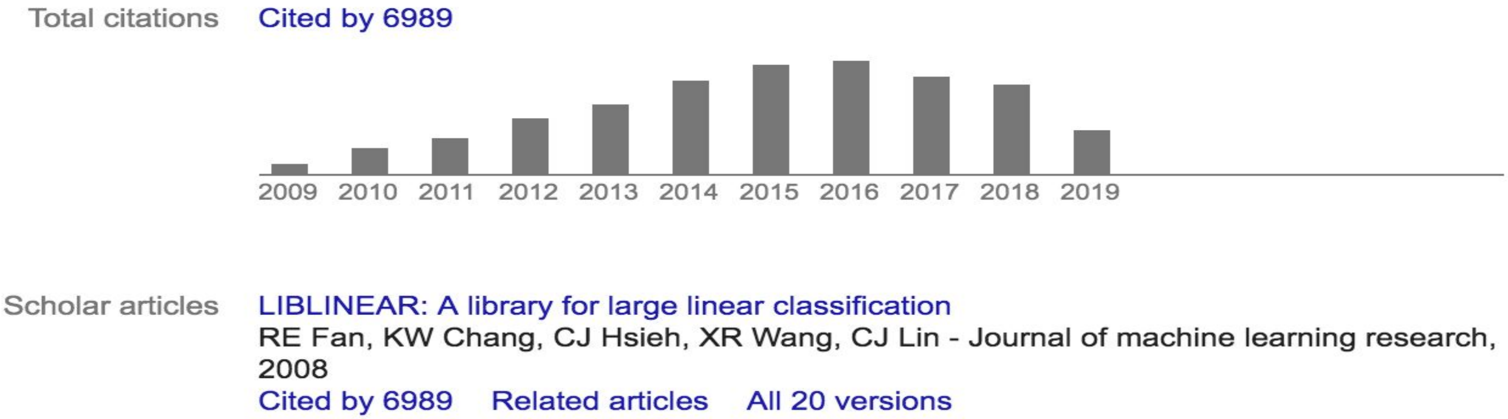
How to find the best linear function?

- Using **Optimization** (One of my research focus)
  - Design a way to find the minimizier of a function
  - Make it efficient on **large datasets**
    - ImageNet: millions of data
    - Google search: trillions of data

Software/toolbox

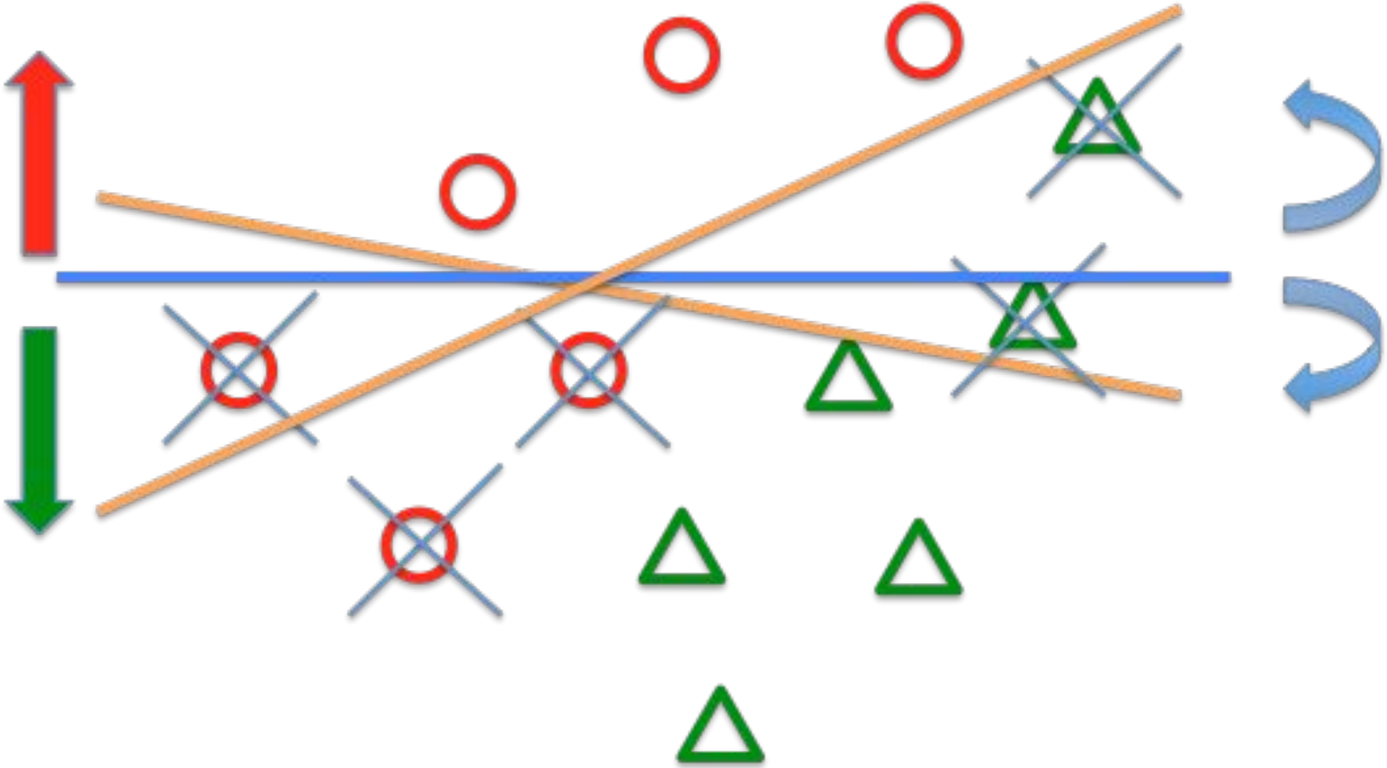
Easy-to-use software available

- LIBLINEAR (Fan, Chang, **Hsieh**, Wang, Lin, 2008)
  - Used in all linear classification tasks (e.g., scikit-learn)

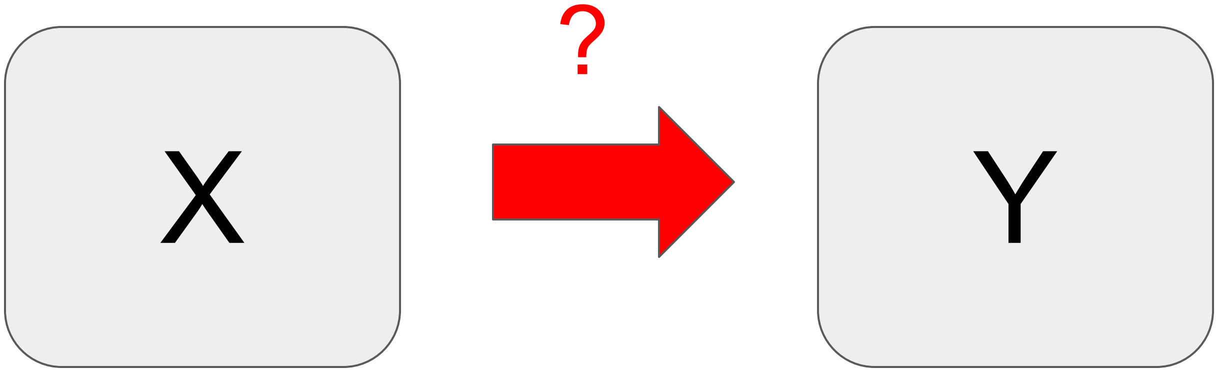


Total citations Cited by 6989

Scholar articles LIBLINEAR: A library for large linear classification
RE Fan, KW Chang, CJ Hsieh, XR Wang, CJ Lin - Journal of machine learning research, 2008
Cited by 6989 Related articles All 20 versions



Machine learning
= Find the function to map X to Y based on data



X

?

Y

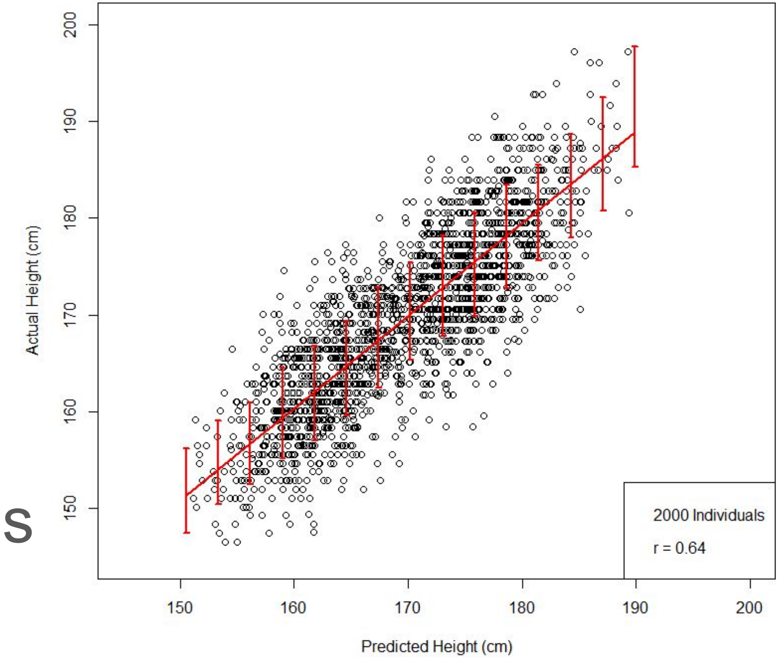
Images
Video
Sentence
Document

Label
Real number
Image
Sentence

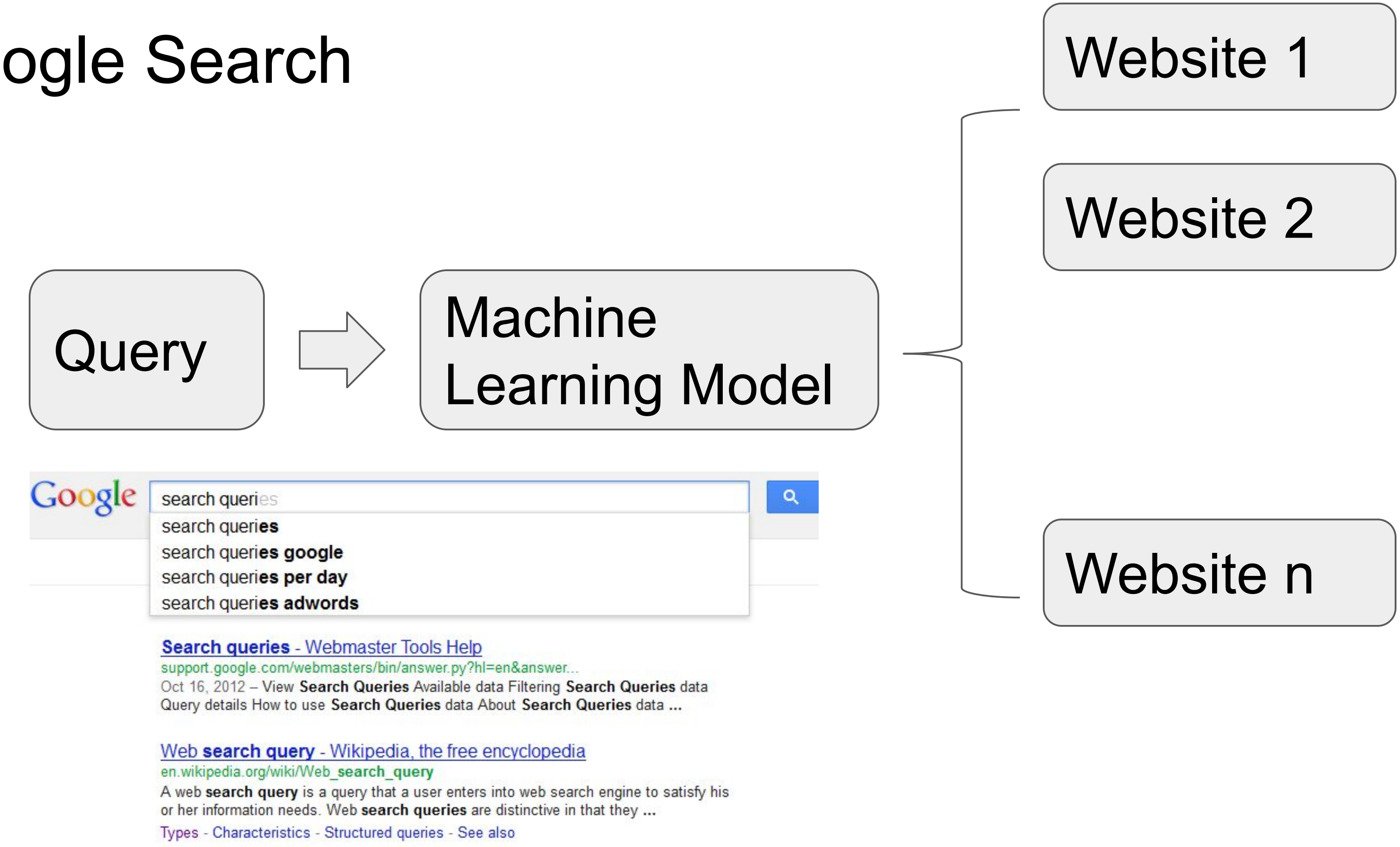
Regression

- y is real number instead of 0/1
- Example:
  - Stock price prediction
  - Predict height by genomic features
- Just need to switch to square loss

$$\text{loss}(f(x), y) = (y - f(x))^2$$



Google Search



search queries

search queries

search queries **google**

search queries **per day**

search queries **adwords**

Search queries - Webmaster Tools Help
support.google.com/webmasters/bin/answer.py?hl=en&answer...
Oct 16, 2012 – View **Search Queries** Available data Filtering **Search Queries** data Query details How to use **Search Queries** data About **Search Queries** data ...

Web **search query** - Wikipedia, the free encyclopedia
en.wikipedia.org/wiki/Web_**search_query**
A web **search query** is a query that a user enters into web search engine to satisfy his or her information needs. Web **search queries** are distinctive in that they ...
Types - Characteristics - Structured queries - See also

Roadmap

- What is machine learning?
- **Neural network**
- Applications beyond classification
- Limitations

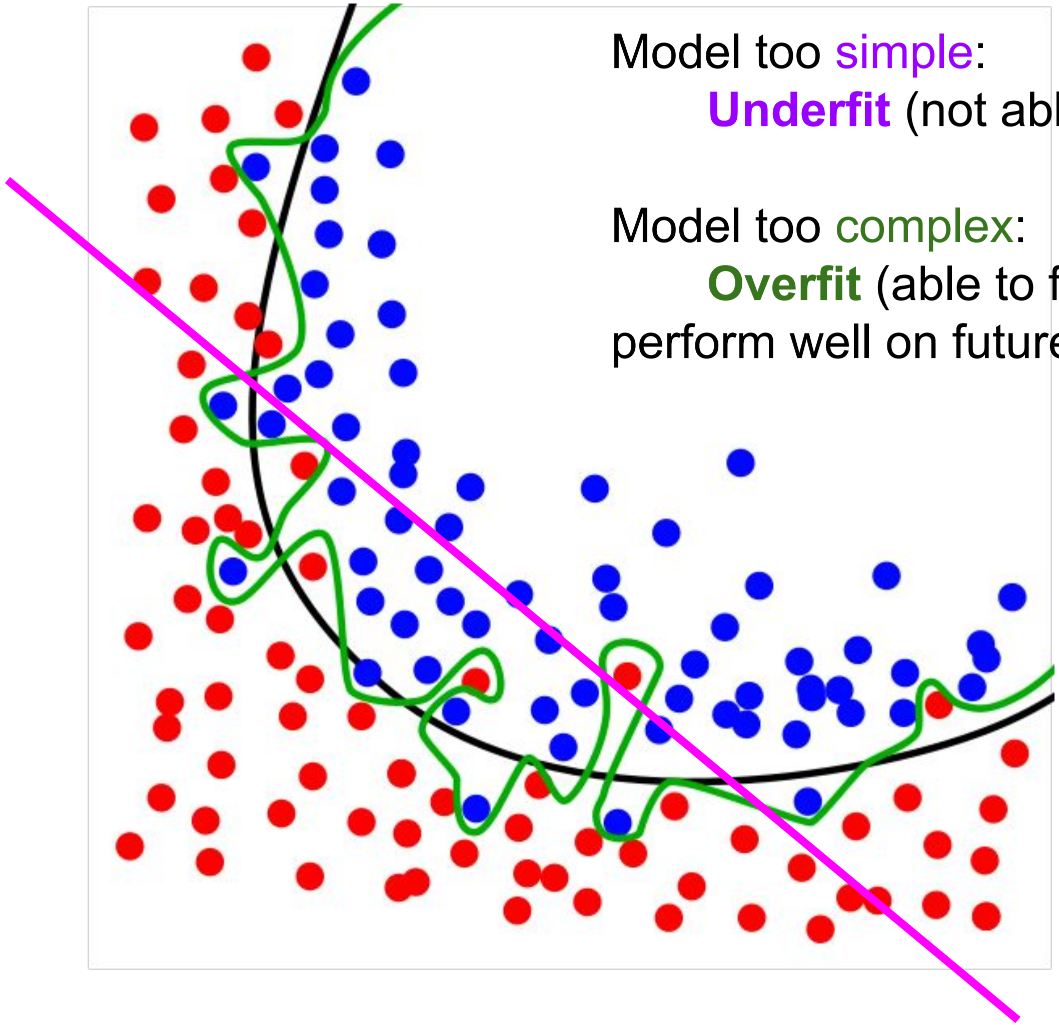
- Linear classification:

  X -> Linear function -> y

- Neural network:

  X -> Neural network -> y

- Neural network defines a nonlinear function
  - Can represent more complex mapping
  - Need more data



Model too simple:

Underfit (not able to improve when adding data)

Model too complex:

Overfit (able to fit all training data but may not perform well on future prediction)

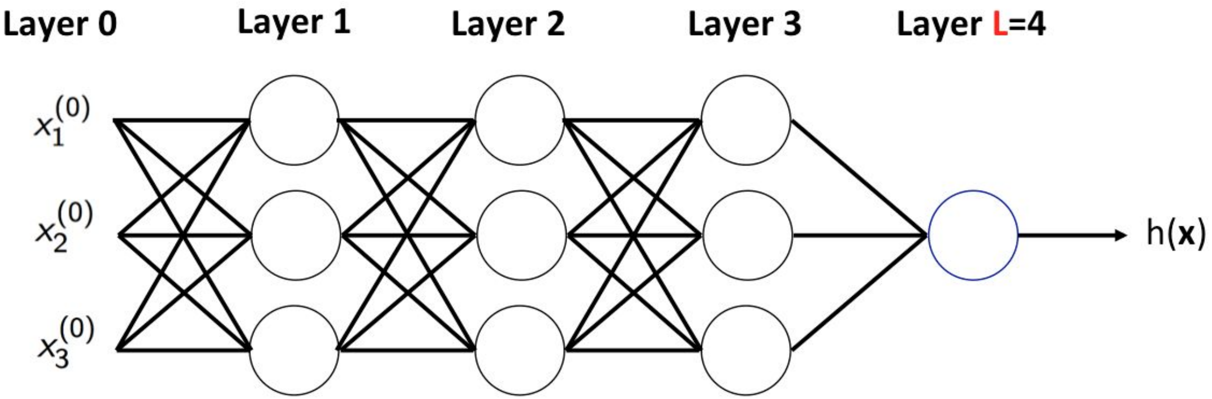
Neural networks need more training data than linear model

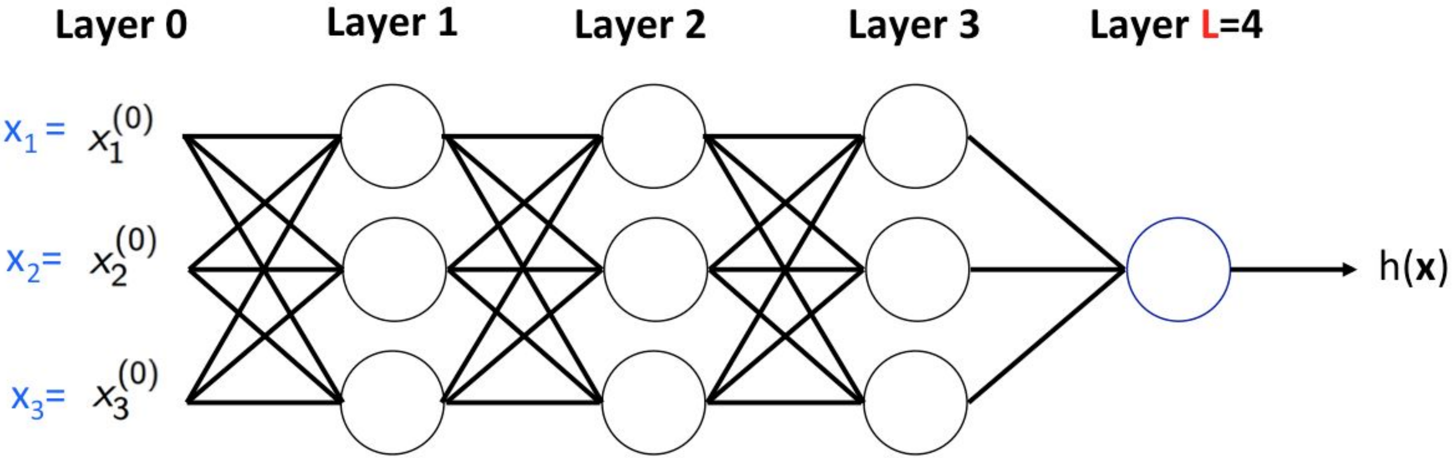
A Brief History

- Before 1998:
  - Main stream of machine learning research
- 1998--2012:
  - Dominated by linear/kernel SVM
- 2012--now:
  - Back due to **Big Data** and **Efficient Hardware**
  - Lead to exciting performance on many tasks

Neural network

- Replace linear function by neural network function





Layer 0 **Layer 1** **Layer 2** **Layer 3** **Layer L=4**

$x_1 = x_1^{(0)}$

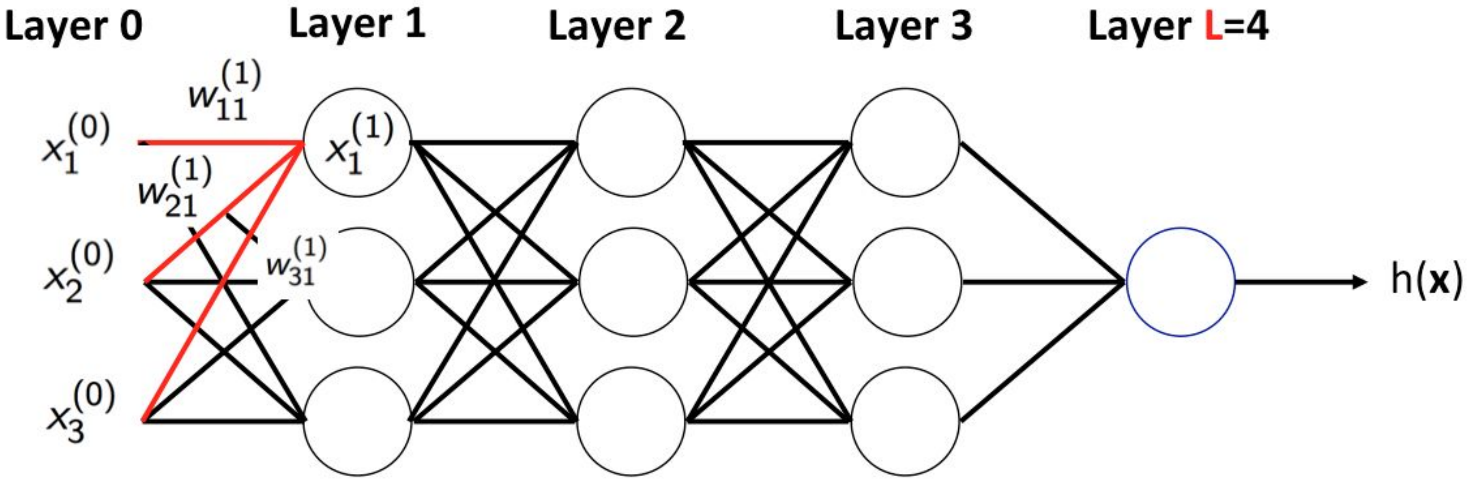
$x_2 = x_2^{(0)}$

$x_3 = x_3^{(0)}$

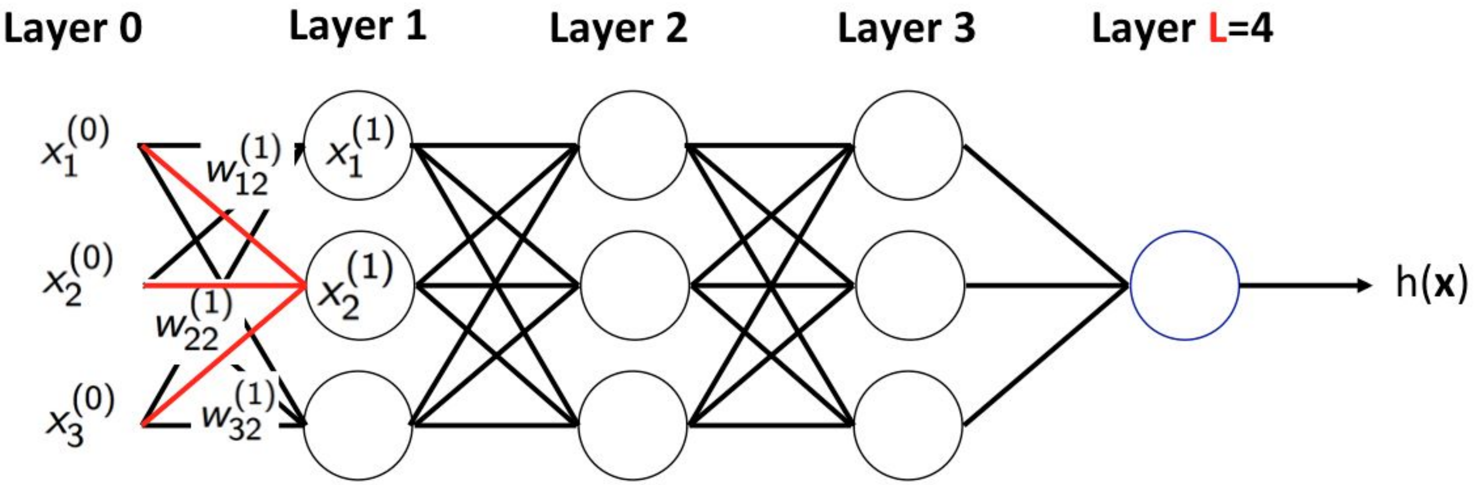
$h(\mathbf{x})$

features for one data point

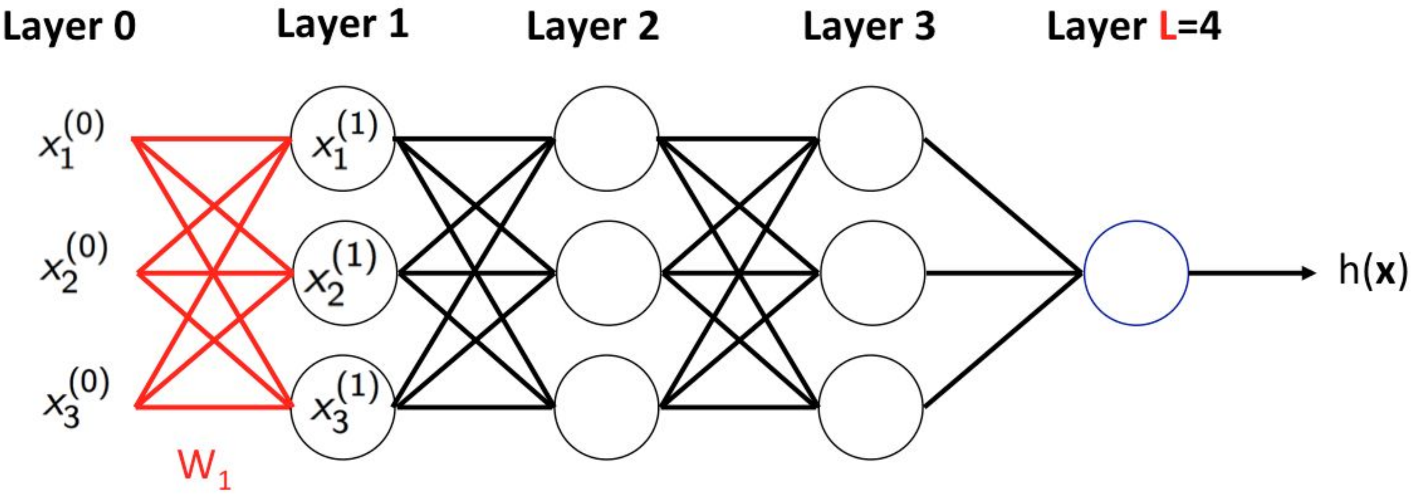
$\mathbf{x} = [x_1, x_2, x_3]$



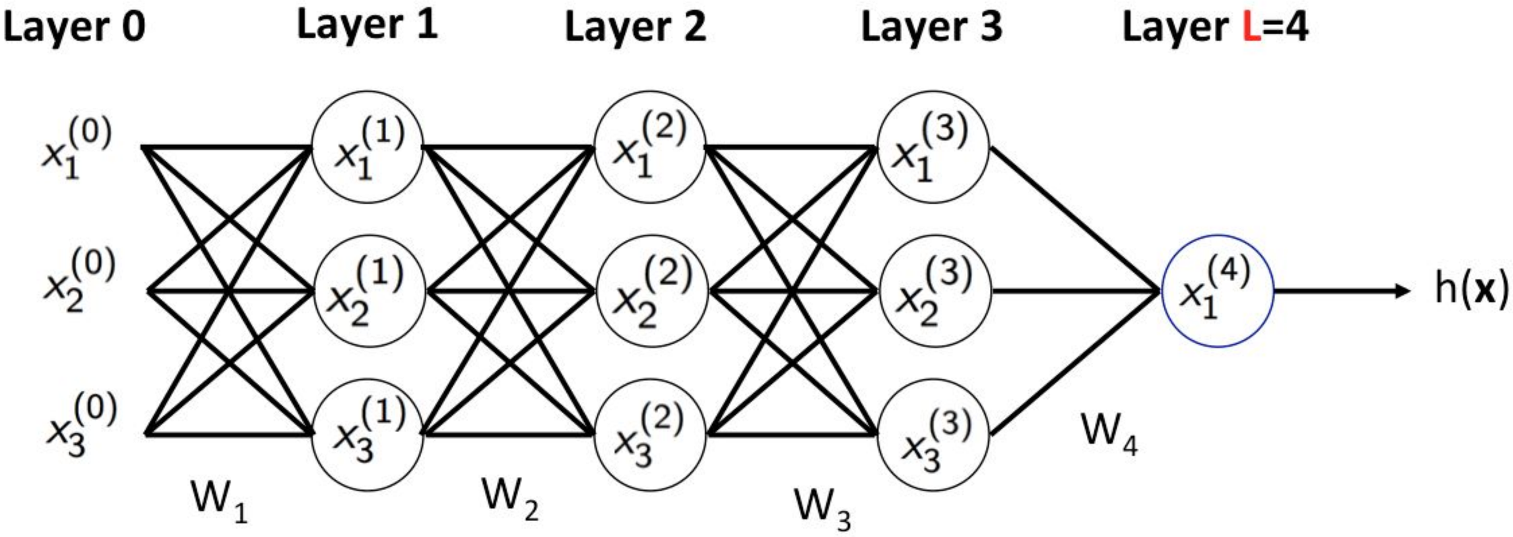
$$x_1^{(1)} = \theta(\textstyle\sum_{i=1}^{3} w_{i1}^{(1)} x_i^{(0)})$$



$$x_2^{(1)} = \theta(\sum_{i=1}^{3} w_{i2}^{(1)} x_i^{(0)})$$



$$\boldsymbol{x}^{(1)} = \begin{bmatrix} x_1^{(1)} \\ x_2^{(1)} \\ x_3^{(1)} \end{bmatrix} = \theta\left(\begin{bmatrix} w_{11}^{(1)} & w_{21}^{(1)} & w_{31}^{(1)} \\ w_{12}^{(1)} & w_{22}^{(1)} & w_{32}^{(1)} \\ w_{13}^{(1)} & w_{23}^{(1)} & w_{33}^{(1)} \end{bmatrix} \times \begin{bmatrix} x_1^{(0)} \\ x_2^{(0)} \\ x_3^{(0)} \end{bmatrix} \right) = \theta(W_1 \boldsymbol{x}^{(0)})$$



$$h(\boldsymbol{x}) = x_1^{(4)} = \theta(W_4\boldsymbol{x}^{(3)}) = \theta(W_4\theta(W_3\boldsymbol{x}^{(2)}))$$

$$= \cdots = \theta(W_4\theta(W_3\theta(W_2\theta(W_1\boldsymbol{x}))))$$

Learning neural network

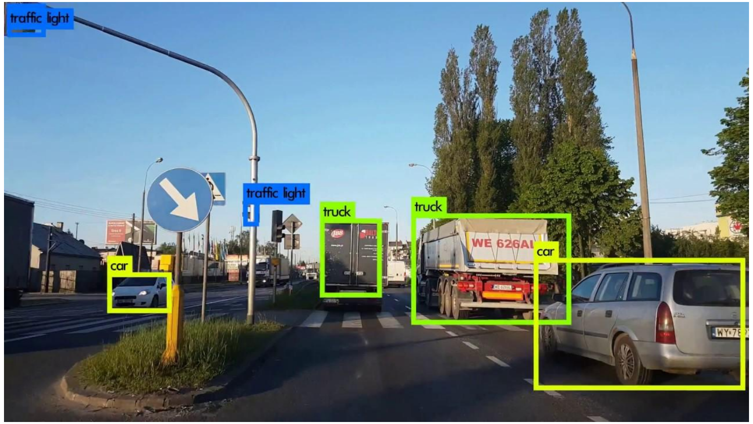
- Find the weights W=[W$_1$, W$_2$, … W$_L$] to minimize Loss
  - Solve the same optimization problem

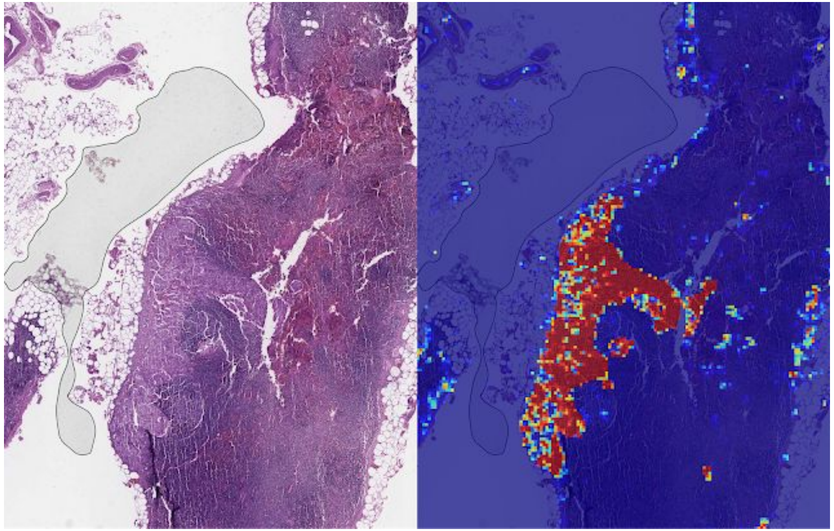
$$\min_W \sum_{x \in D} \text{loss}(f_W(x), y)$$

- However, multiple layers lead to high computational cost

More complex output space

- Object detection: output bounding box + label

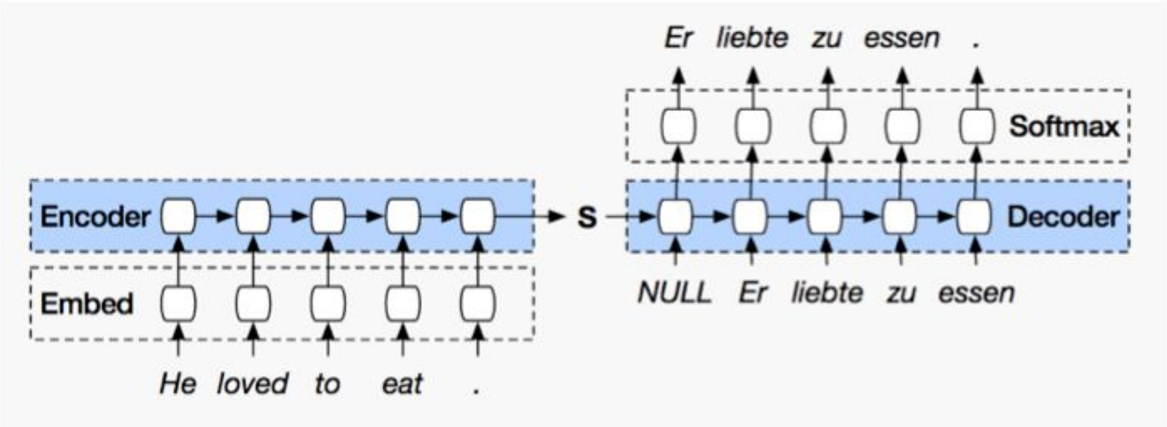




Cancer detection

More complex output space

- Machine translation (or any sentence generation)



Challenges / Research Topics

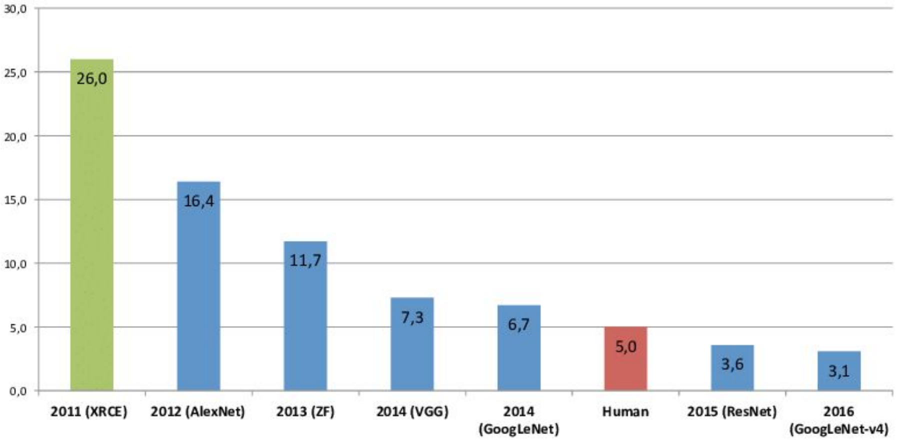
Challenges / Research Topics

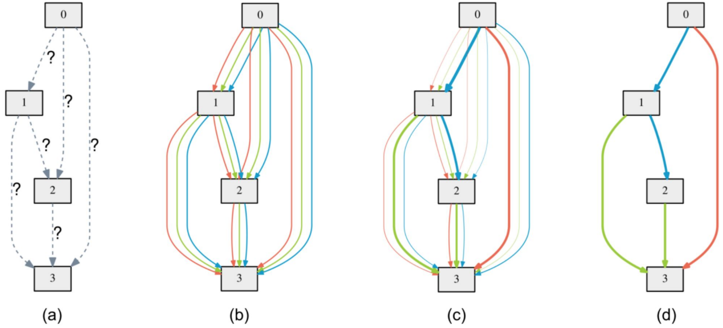
- Model design
- Efficiency (both training and testing)
- Robustness
- Explainability
- ....

Model Design

- Design network architecture For each task
  - NLP, computer vision, graph mining, etc

- Automatic model design
  - Neural architecture search

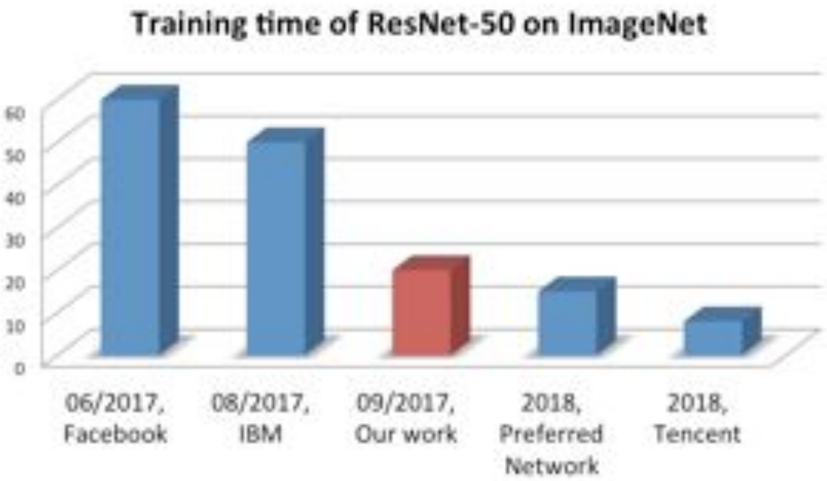


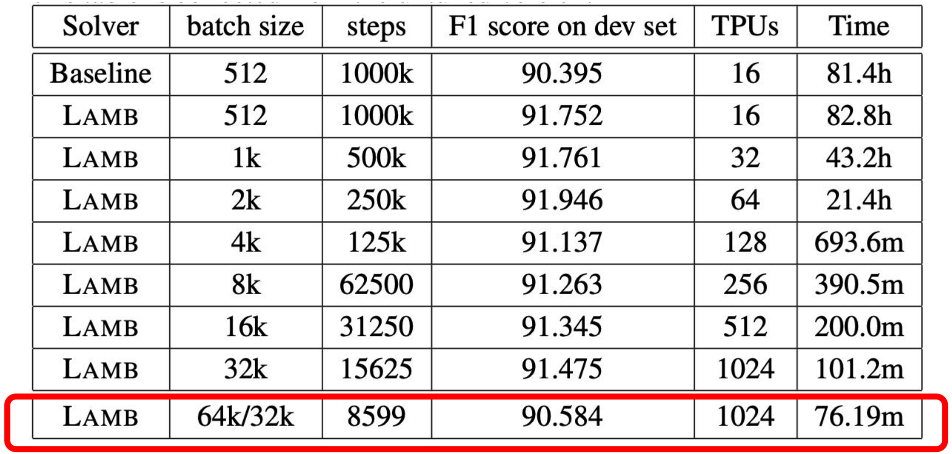


Training Efficiency

- Training by SGD
  - Sample data point => check current prediction => update
- 3 days for training ImageNet (~1.5 million images) on 1 GPU
- Speed up using multiple GPUs/TPUs
  - Efficient algorithm for large batch size
  - Learning rate scheduling

ImageNet (1.5 million images) can be solved in minutes





Solver	batch size	steps	F1 score on dev set	TPUs	Time
Baseline	512	1000k	90.395	16	81.4h
LAMB	512	1000k	91.752	16	82.8h
LAMB	1k	500k	91.761	32	43.2h
LAMB	2k	250k	91.946	64	21.4h
LAMB	4k	125k	91.137	128	693.6m
LAMB	8k	62500	91.263	256	390.5m
LAMB	16k	31250	91.345	512	200.0m
LAMB	32k	15625	91.475	1024	101.2m
LAMB	64k/32k	8599	90.584	1024	76.19m

BERT training in 76 minutes

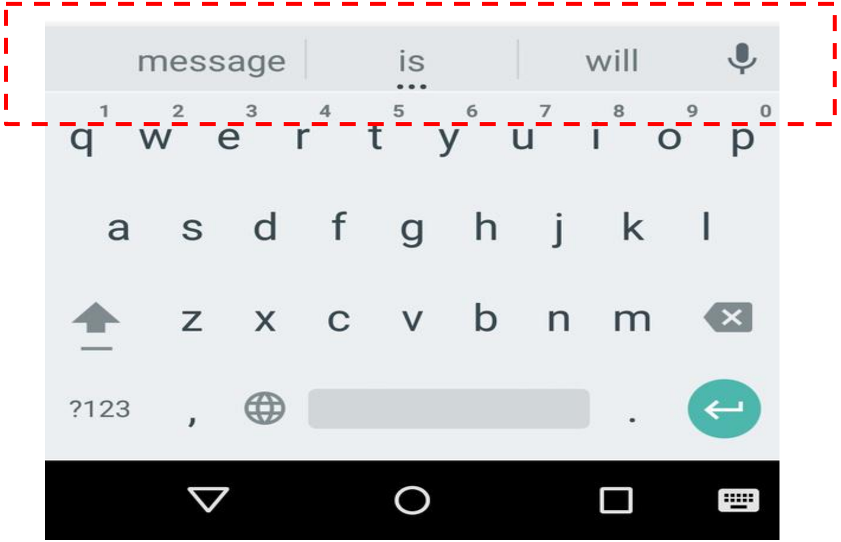
Model size and inference speed

Limited storage on mobile devices

Storage Almost Full

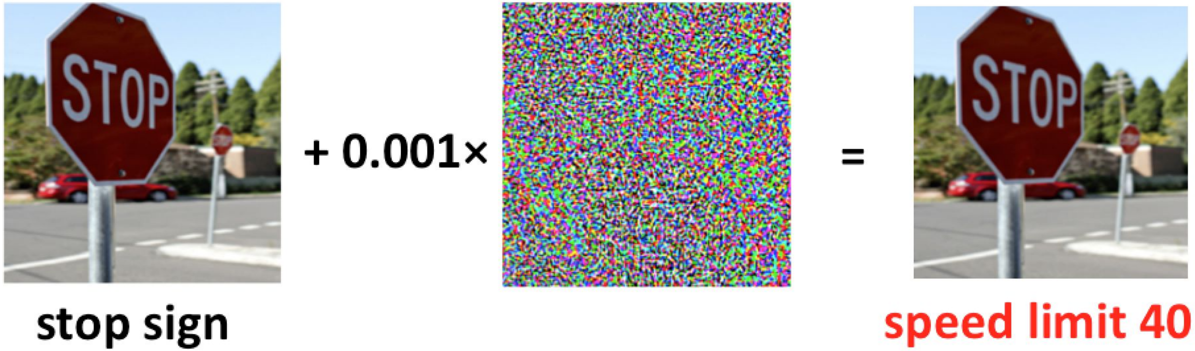
You can manage your storage in Settings.

Inference needs to be done in milliseconds

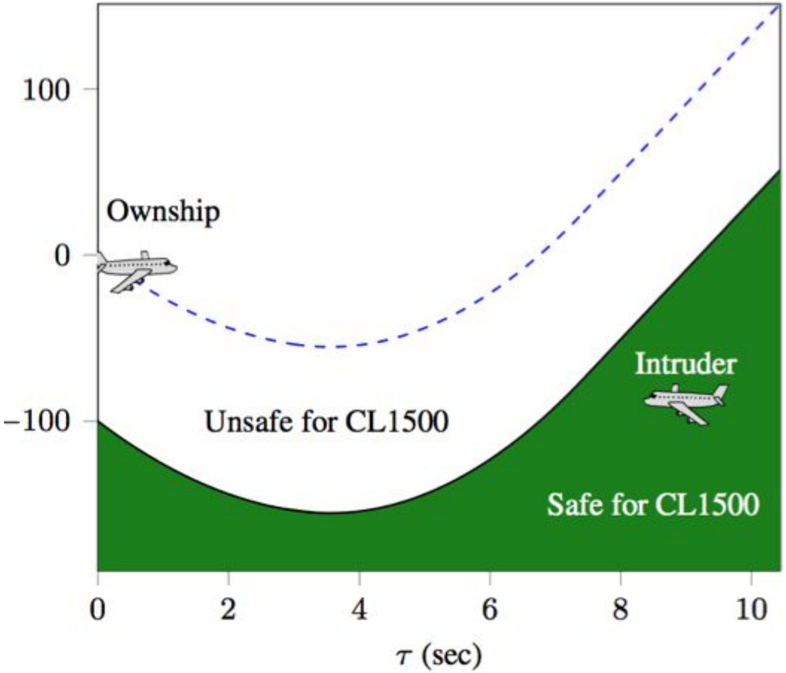


Models are sensitive to small perturbation

- Although models outperform human, a small perturbation to a natural image can make accuracy -> 0



Models are sensitive to small perturbation



Current solution

- Consider perturbation in the training phase

	MNIST 0.1 perturb	MNIST 0.3 perturb	CIFAR 0.03 perturb
(Wong 2018)	3.67	43.1	78.22
(Xiao 2018)	4.4	19.3	79.73
(Gowal 2018)	2.9	8.2	68.5
(Zhang 2019)	**2.4**	**7.6**	**67.2**

(error rate)

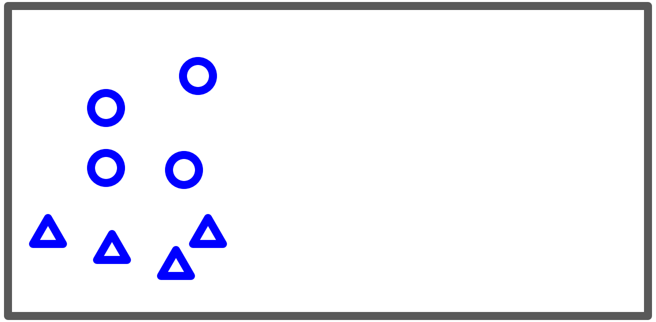
How to make it work for more complex problems?

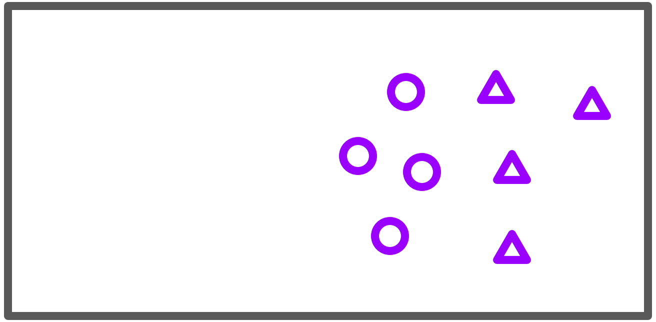
Robustness of other models (tree, nearest neighbor, …)

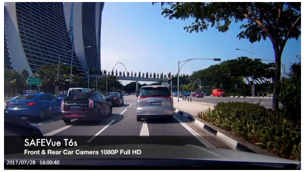
Similar problem: Domain shift

Similar problem: Domain shift

- Self-driving cars: weather, environment, …
- Medical data: different race, different sensors, ...





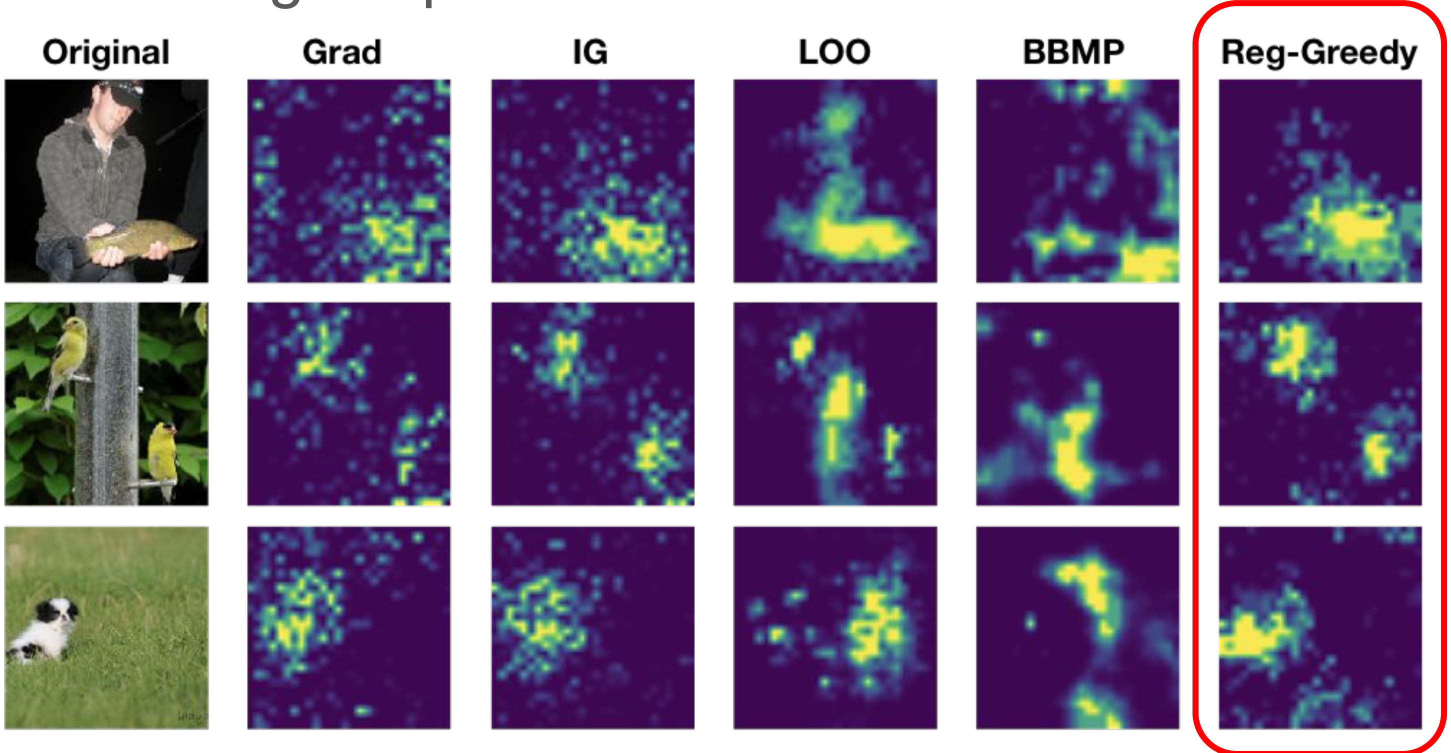


Training data

Test-time input

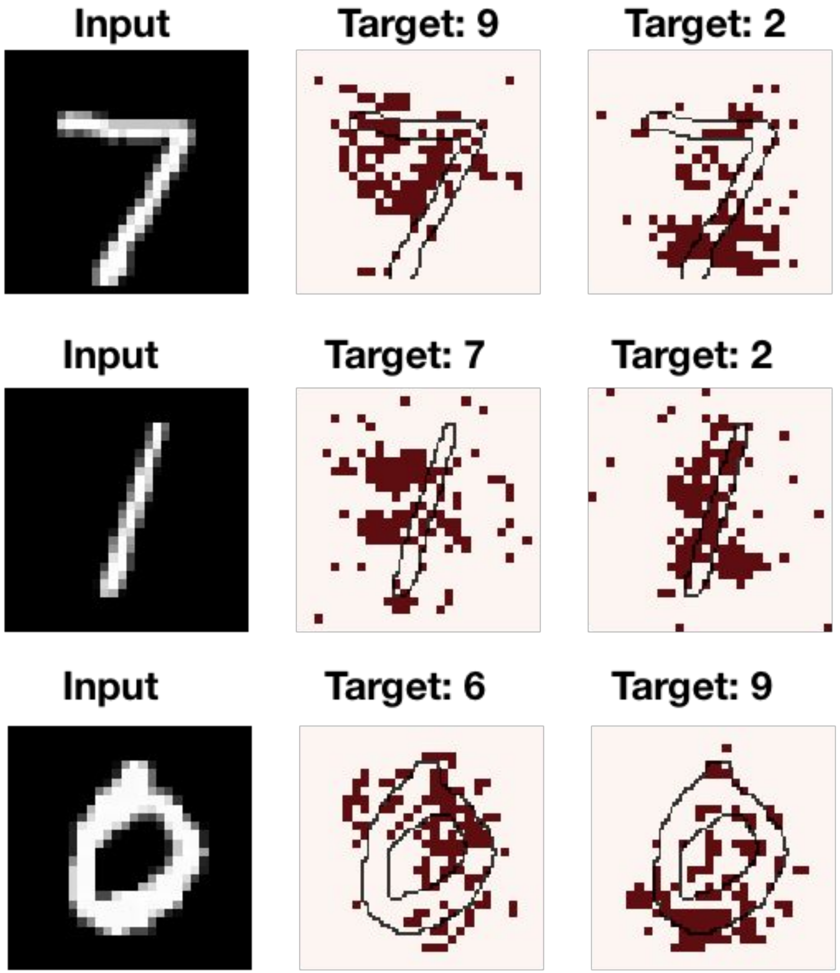
Model Interpretability

- Why is this image is predicted as class X?



Model Interpretability

- Why is this image is predicted as class X but not Y?



Conclusion

Machine learning
= Find the function to map X to Y based on data