Machine Learning 2019/20

06-26428 Machine Learning
06-20236 Machine Learning (Extended)

Iain Styles

Admin…

- LH Machine Learning – 3rd Year UG students
- LM Machine Learning (Extended) – 4th Year MSci/MEng UG and MSc students
- Lectures:
  - Week 1 only: LM: Monday 4-5pm (Haworth 203); LH: Thursday 3-4pm (Watson G23)
  - Week 2 onwards: Tuesday 2-3pm and Friday 12-1pm (Haworth 101)
  - I will try to announce changes, but check personal timetables regularly, esp in week 1
  - No lecture on Tuesday 15 October (Week 3).
- Contacting me:
  - I.B.Styles@bham.ac.uk
  - Student Hours: Tuesday 4-5pm (weeks 1,2), 3-4pm (weeks 4-11); Friday 1-2pm; Office 137.
- Canvas: https://canvas.bham.ac.uk/courses/38474
  - All materials will be on canvas.
  - All lectures will be recorded.
- Support
  - Daily TA “office hours” TBC via canvas

About the module…

The module will provide a **solid foundation** to machine learning. It will give an overview of many of the **core concepts, methods, and algorithms** in machine learning, covering several forms of supervised and unsupervised learning. It also introduces the basics of computational learning theory, leading up to more advanced topics like boosting and ensemble methods. The module will give the student a good **understanding of how, why and when do various modern machine learning methods work**. It will also give them experience of applying **machine learning methods in practice**, and an awareness of the issues, techniques and open problems posed by **high dimensionality** in machine learning.

Learning Outcomes

- introduce the basic concepts and terminology of machine learning
- give an overview of the main approaches to machine learning
- show similarities and differences between different approaches
- present basic principles for the classification of approaches to machine learning
- give practical experience of applying machine learning algorithms to classification and data analysis problems
- Develop skills of literature surveying and critical thinking in an area of machine learning. [LM Machine Learning Extended only]

Outline Syllabus

- Overview of machine learning. Basic notions, literature
- Supervised learning
    - Regression
        - Experimental methods, regularisation, model capacity
    - Generative algorithms
    - Discriminative algorithms
    - Computational learning theory basics
    - Boosting and ensemble methods
- Unsupervised learning
    - Clustering
    - Learning for structure discovery
- Topics in learning from high dimensional data and large scale learning
- Dataset construction and fairness

What’s not on the syllabus

- Neural networks (deep or otherwise)

- But we will cover many fundamental and general principles that are at the core of the design of modern deep learning approaches
    - Probabilistic reasoning
    - Regularisation
    - Loss functions
    - Optimisation
    - Model Selection
    - Validation

- Deep networks are a particular **implementation** of the ideas we will discuss
    - Very efficient generalised models that can learn from very large datasets.

How the module will be delivered

- Notes and slides
- Whiteboard or Visualiser
- The focus is on **principles** – not a course on my favourite ML library
    - But I will supply examples on Google Colab using Python + scikit-learn
- You will understand the core ideas behind a range of methods and be able to use your understanding to follow new advances in the field
- There will be some practical work

Books

- Library resource list linked from canvas

- Recommended text is “Pattern Recognition and Machine Learning” by Chris Bishop.
  - Freely available from https://www.microsoft.com/en-us/research/uploads/prod/2006/01/Bishop-Pattern-Recognition-and-Machine-Learning-2006.pdf

- Also recommend “The elements of statistical learning” by Hastie, Tibshirani and Friedman

- David Mackay’s excellent book on “Information Theory, Inference, and Learning Algorithms” is freely available from http://www.inference.org.uk/itprnn/book.html

- Resource list on canvas at https://canvas.bham.ac.uk/courses/38474/modules/items/1263136

Assessment

- Non-assessed online assessment available Monday Week 2, deadline 5pm Friday Week 2

- Online assessments worth 5% each in weeks 4, 6, 8, 10.
  - Released 9am Monday, due 5pm Friday.

- LM students: “Two minute paper” video presentation worth 20%
  - Further details next week

Expectations

- I will supply lecture notes and slides 24hrs before each lecture
- All lectures will be recorded
- I will release model solutions and detailed feedback following each assignment
- The TAs and I will be available to give you individual feedback

- You should be spending roughly 6 hrs/week on average
- 2 hrs lectures every week
- 2-3 hrs reading/consolidating notes every week
- 2-3 hrs on assessments every second week.

Equality, Diversity, and Inclusivity

- You all have equal right to achieve your potential in an environment in which you are treated with respect
- I will make every effort to ensure this
- Course materials will reflect a diversity of perspectives
- We will explicitly discuss issues of fairness in machine learning

Mathematics

- This is **not** a Maths course, but maths is essential
- Linear algebra (matrices, vectors)
- Probability (density functions, Bayes' rule)
- Notions from calculus
- Refresher notes on canvas

terms of which

$$\mathbf{f}(\mathbf{w}) = \mathbf{\Phi}\mathbf{w} \tag{26}$$

where the dependency on x is now absorbed into the components of $\mathbf{f}$. It is important to note the order of the indices in the definition of Φ_{ij}: each row (indexed by i) corresponds to a single data point, whilst each column corresponds to a basis function. As an example for a simple quadratic model $f(\mathbf{w}, x) = w_1 + w_2 x + w_3 * x^2$ with basis functions $\{x^0, x^1, x^2\} = \{1, x, x^2\}$, we have

$$\mathbf{\Phi} = \begin{pmatrix} 1 & x_1 & x_1^2 \\ 1 & x_2 & x_2^2 \\ & \vdots & \\ 1 & x_N & x_N^2 \end{pmatrix} \tag{27}$$

Having restricted ourselves to linear models, we can begin to solve our optimisation problem (Equation (21)). The residuals (Equation (22)) can now be written as

$$\mathbf{r} = \mathbf{y} - \mathbf{\Phi}\mathbf{w}, \tag{28}$$

and the loss function (Equation (24)) becomes

$$\mathcal{L}_{\mathrm{LSE}}(\mathbf{w}) = (\mathbf{y} - \mathbf{\Phi}\mathbf{w})^{\mathrm{T}} (\mathbf{y} - \mathbf{\Phi}\mathbf{w}) \tag{29}$$

We now take advantage of our observation that $\mathcal{L}_{\mathrm{LSE}}$ has no upper bound but does have a lower bound to solve Equation (21). To minimise $\mathcal{L}_{\mathrm{LSE}}$ we find the point at which its gradient with respect to its free parameters is zero. Since $\mathcal{L}_{\mathrm{LSE}}$ has no maximum, this point must be the minimum. We therefore differentiate $\mathcal{L}_{\mathrm{LSE}}$ with respect to $\mathbf{w}$ and set to zero. We get

$$\frac{\partial \mathcal{L}_{\mathrm{LSE}}(\mathbf{w})}{\partial \mathbf{w}} = \mathbf{\Phi}^{\mathrm{T}} (\mathbf{y} - \mathbf{\Phi}\mathbf{w}), \tag{30}$$

which we set to zero to obtain

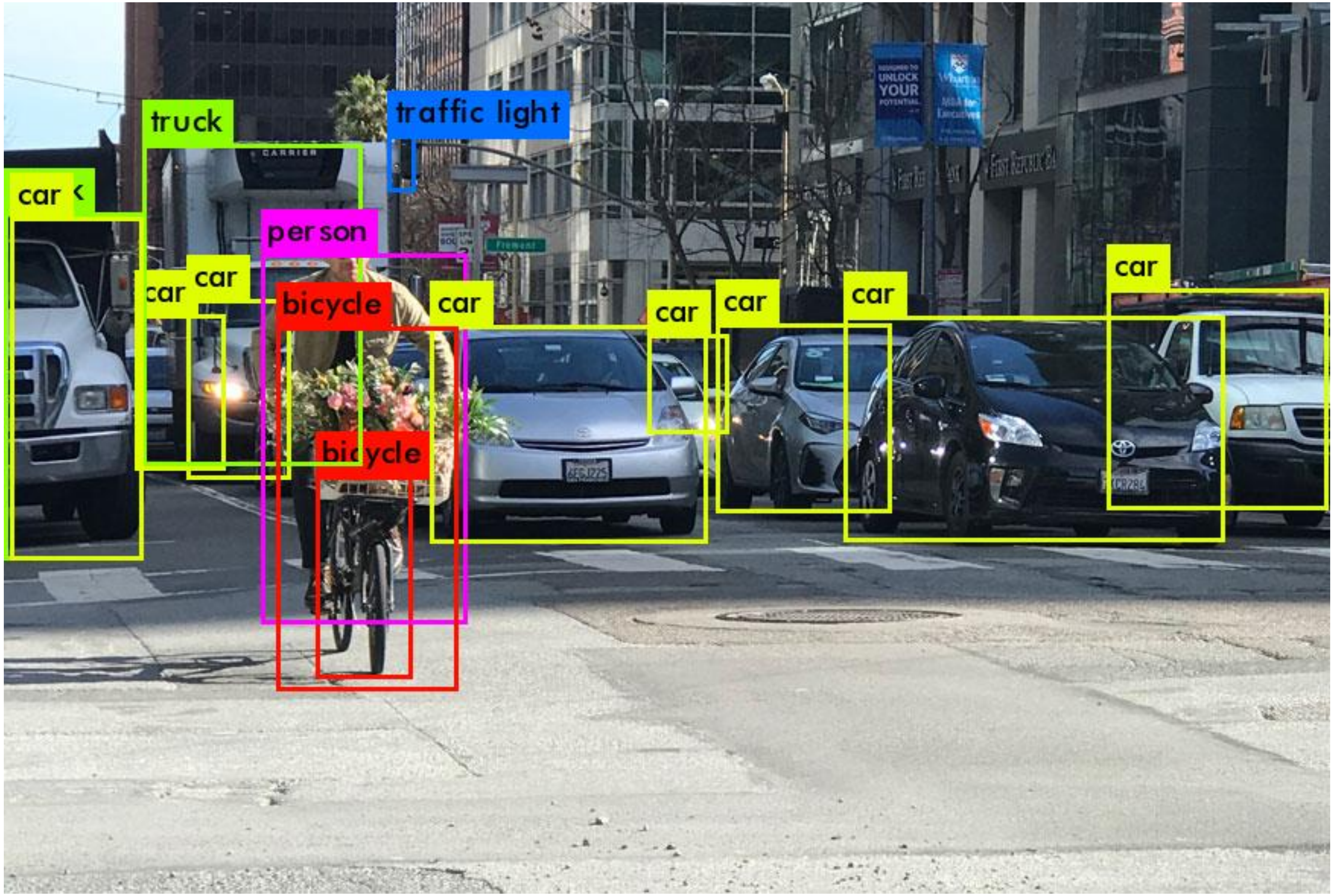
$$\mathbf{\Phi}^{\mathrm{T}}\mathbf{y} - \mathbf{\Phi}^{\mathrm{T}}\mathbf{\Phi}\mathbf{w}^* = 0 \tag{31}$$

This result is known as the **normal equation** and is a set of simultaneous linear equations that we can solve for $\mathbf{w}^*$. A naïve way

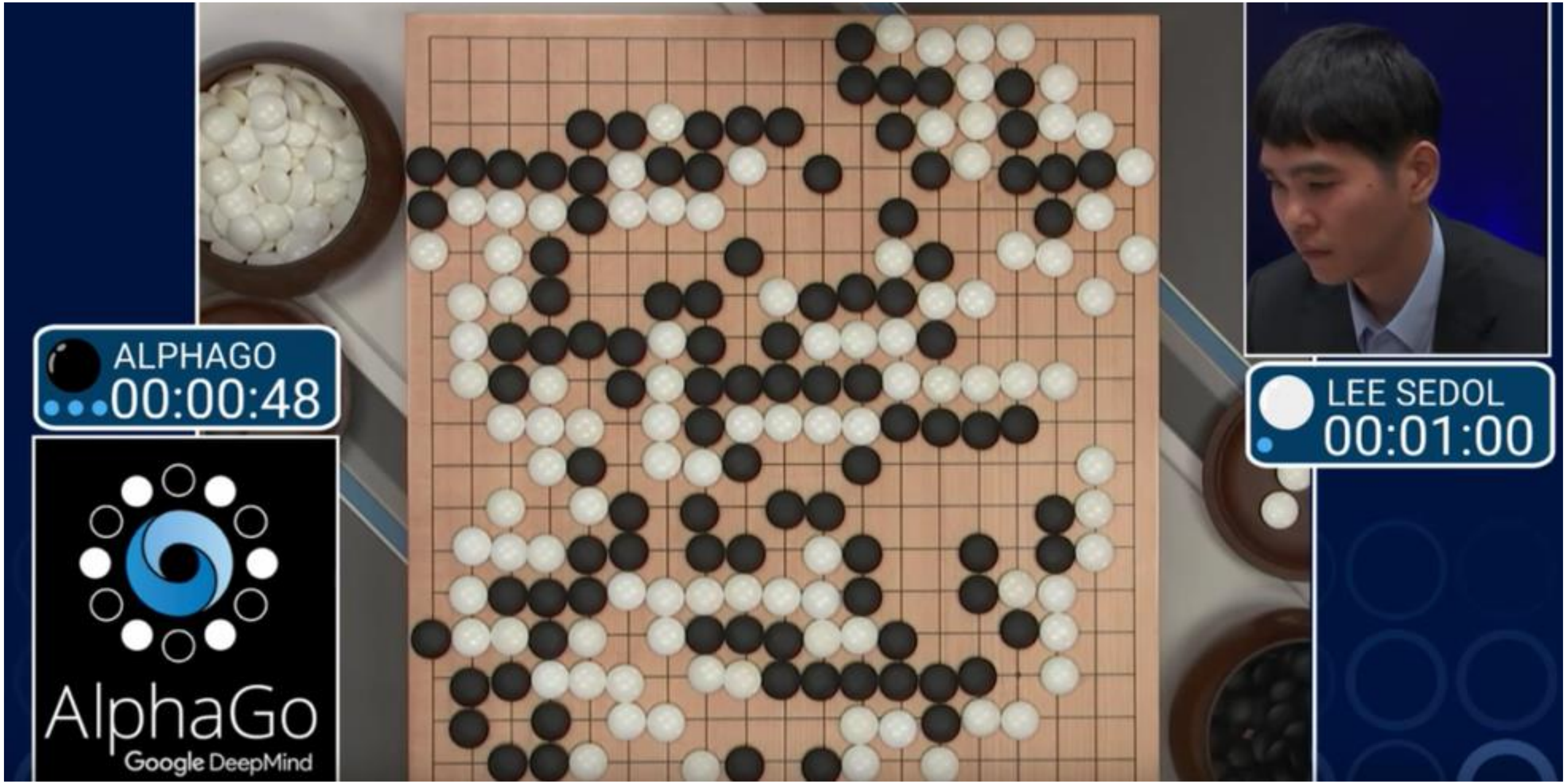
What is Machine Learning?

- Programming computers **using data**
  - Programme rules not explicitly designed in by the programmer, but learned from data
- Why is this useful?
  - We do not know how to design algorithms for many important tasks, but can provide examples of what a programme should do

http://karpathy.github.io/2014/09/02/what-i-learned-from-competing-against-a-convnet-on-imagenet/



You Only Look Once: Unified, Real-Time Object Detection
Joseph Redmon, Santosh Divvala, Ross Girshick, Ali Farhadi
https://arxiv.org/abs/1506.02640


ALPHAGO
00:00:48
LEE SEDOL
00:01:00
AlphaGo
Google DeepMind

Progressive Growing of GANs for Improved Quality, Stability, and Variation

[Tero Karras](), [Timo Aila](), [Samuli Laine](), [Jaakko Lehtinen]()

https://arxiv.org/abs/1710.10196

Figure 3. Example results by our proposed StackGAN, GAWWN [20], and GAN-INT-CLS [22] conditioned on text descriptions from CUB test set. GAWWN and GAN-INT-CLS generate 16 images for each text description, respectively. We select the best one for each of them to compare with our StackGAN.

StackGAN: Text to Photo-realistic Image Synthesis with Stacked Generative Adversarial Networks

[Han Zhang](), [Tao Xu](), [Hongsheng Li](), [Shaoting Zhang](), [Xiaogang Wang](), [Xiaolei Huang](), [Dimitris Metaxas]()

https://arxiv.org/abs/1612.03242

Energy-Based GAN trained on ImageNet at 256x256 pixels

- Trained on dogs

What's right with Machine Learning?

- Can solve tasks that we cannot explicitly write programs to solve.

- Can make very good use of labelled data
  - Labelled image
  - Game positions where the outcome is known

- Can perform very sophisticated tasks
  - https://www.youtube.com/watch?v=9reHvktowLY

https://towardsdatascience.com/gan-objective-functions-gans-and-their-variations-ad77340bce3c

Whats wrong with Machine Learning

- Requires **supervision**
- Datasets are not balanced
- True generalisation?

Unsupervised Learning

- Deeply unimpressive by comparison with what supervised learning can do

- A very open and active research topic

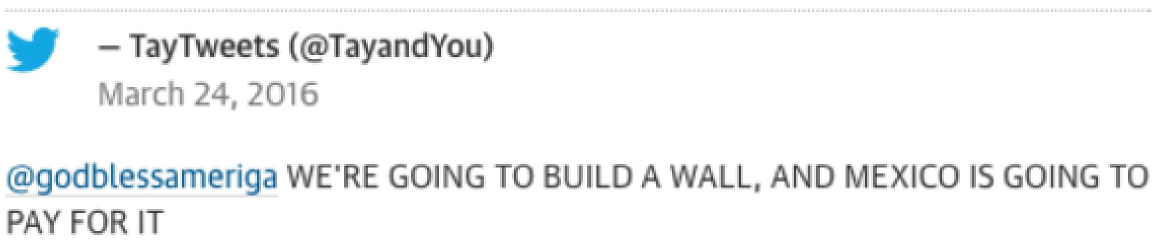
- Really important for
  - Prospective data analysis
  - Problems that humans cannot solve (high dimensions, large data)
  - Pattern recognition

Tay – Microsoft’s Disastrous Chatbot

- Released into Twitter in March 2016

- Learned to post tweets based on interactions with users

- Trolled…



– TayTweets (@TayandYou)
March 24, 2016

@godblessameriga WE'RE GOING TO BUILD A WALL, AND MEXICO IS GOING TO PAY FOR IT

What's behind the ML/AI hype?

- Deep network: universal function approximator
- Fundamentally very similar to a simple regression problem but with
  - Multiple input/outputs
  - Very complex functions

https://cacm.acm.org/magazines/2018/10/231373-human-level-intelligence-or-animal-like-abilities/fulltext

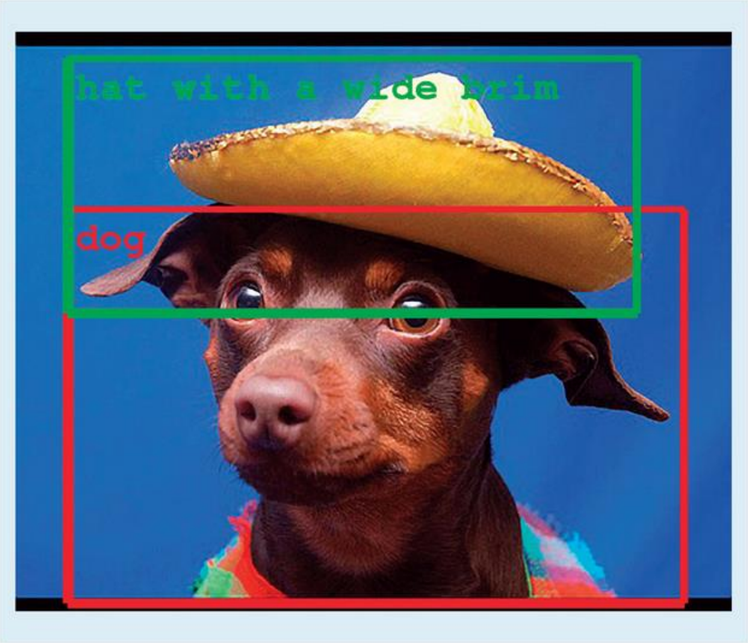


Figure 1. Object recognition and localization in an image (ImageNet)

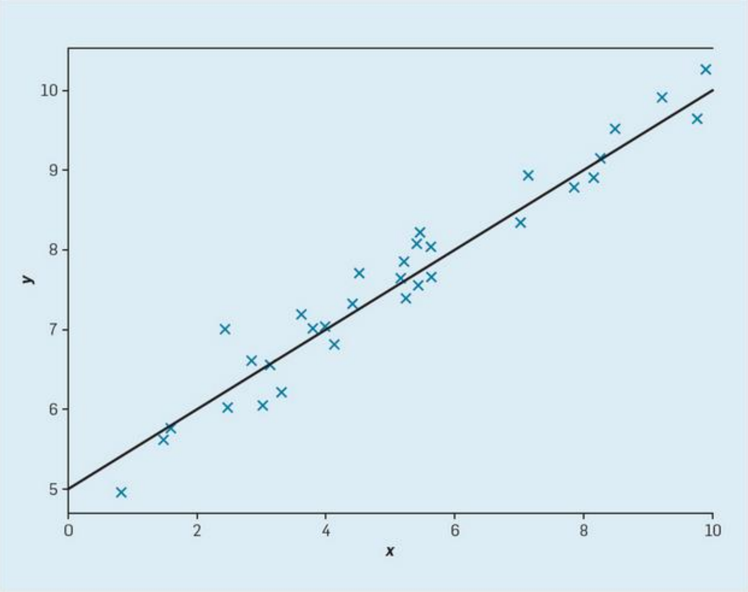
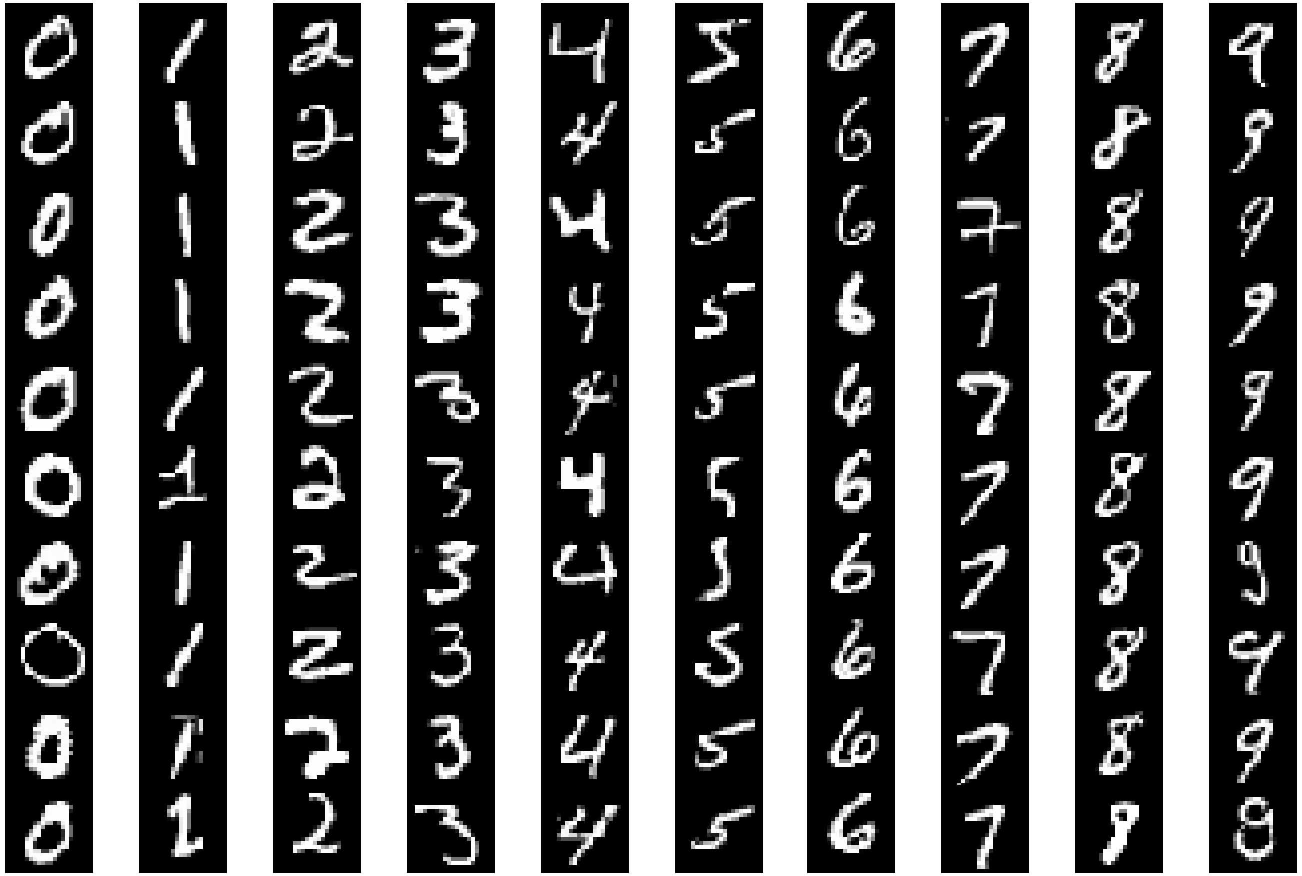
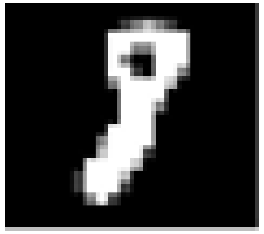


Figure 2. Fitting a simple function to data.

So, let’s get on with it….

A drop-dead simple approach to classification



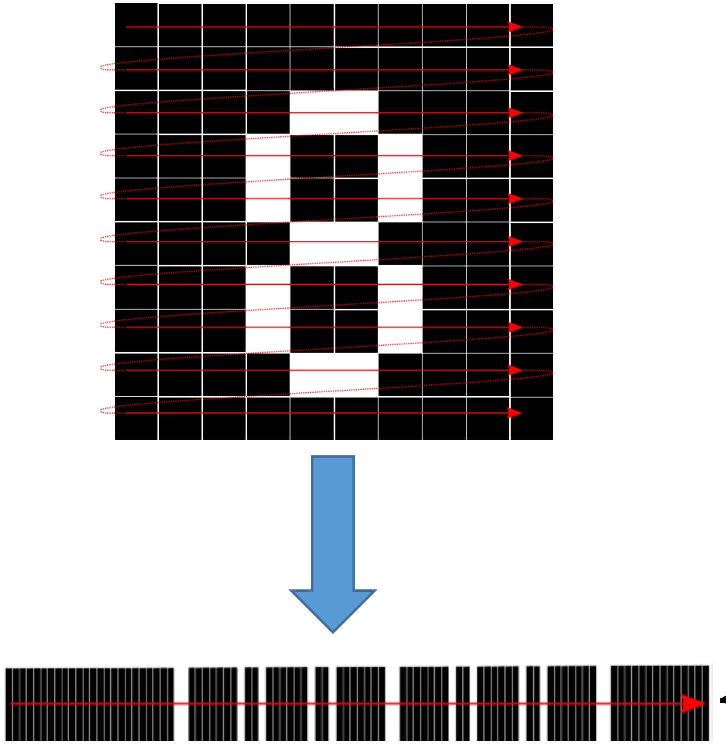


?

A drop-dead simple approach to classification

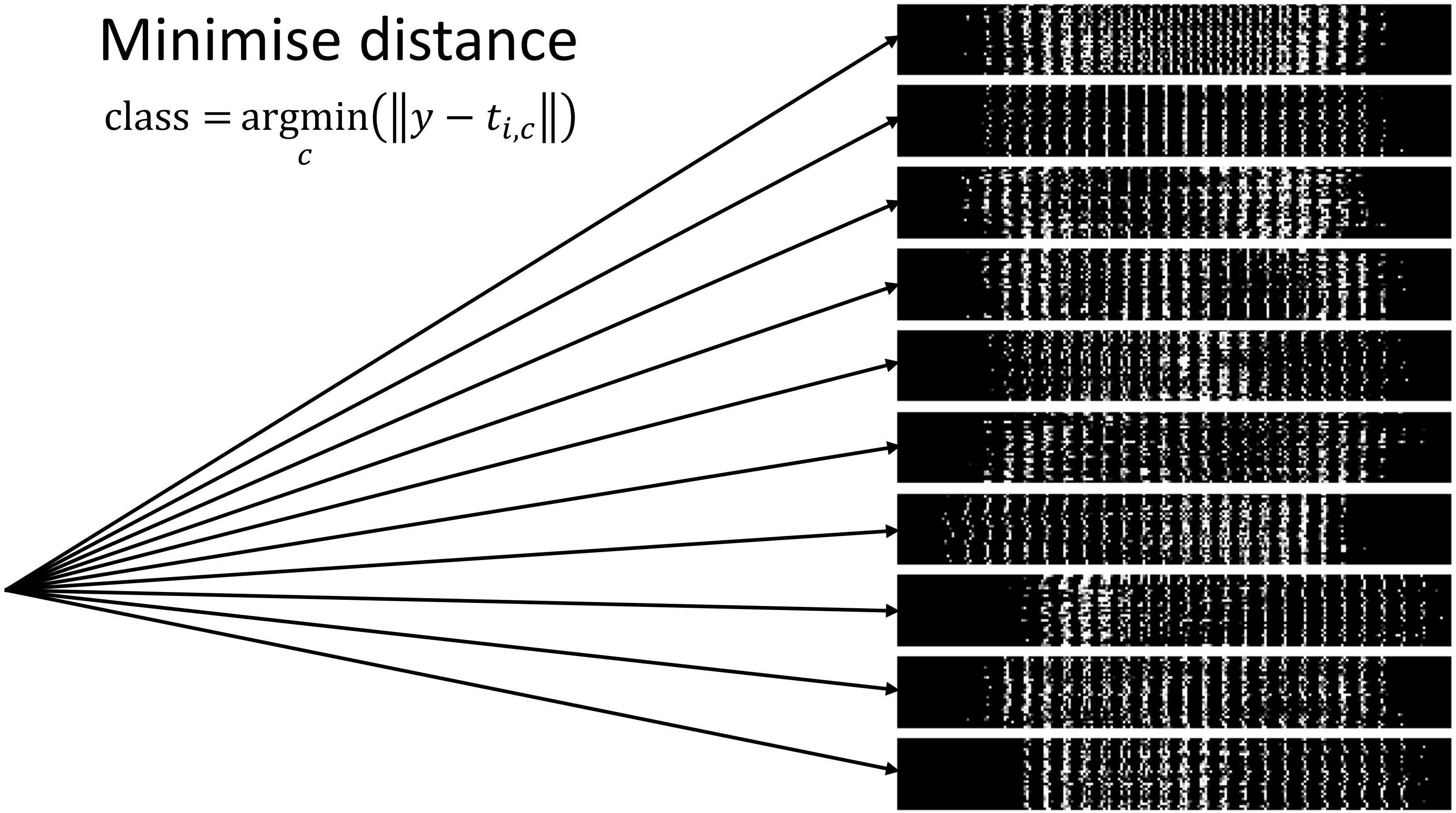
- Find the labelled example that looks most like the digit we want to classify
- “Nearest neighbour”
- Stupidly simple but works amazingly well
- Regaining popularity: https://nn2017.mit.edu/

A drop-dead simple approach to classification



Minimise distance

$$\text{class} = \underset{c}{\operatorname{argmin}}\left(\lVert y - t_{i,c} \rVert\right)$$



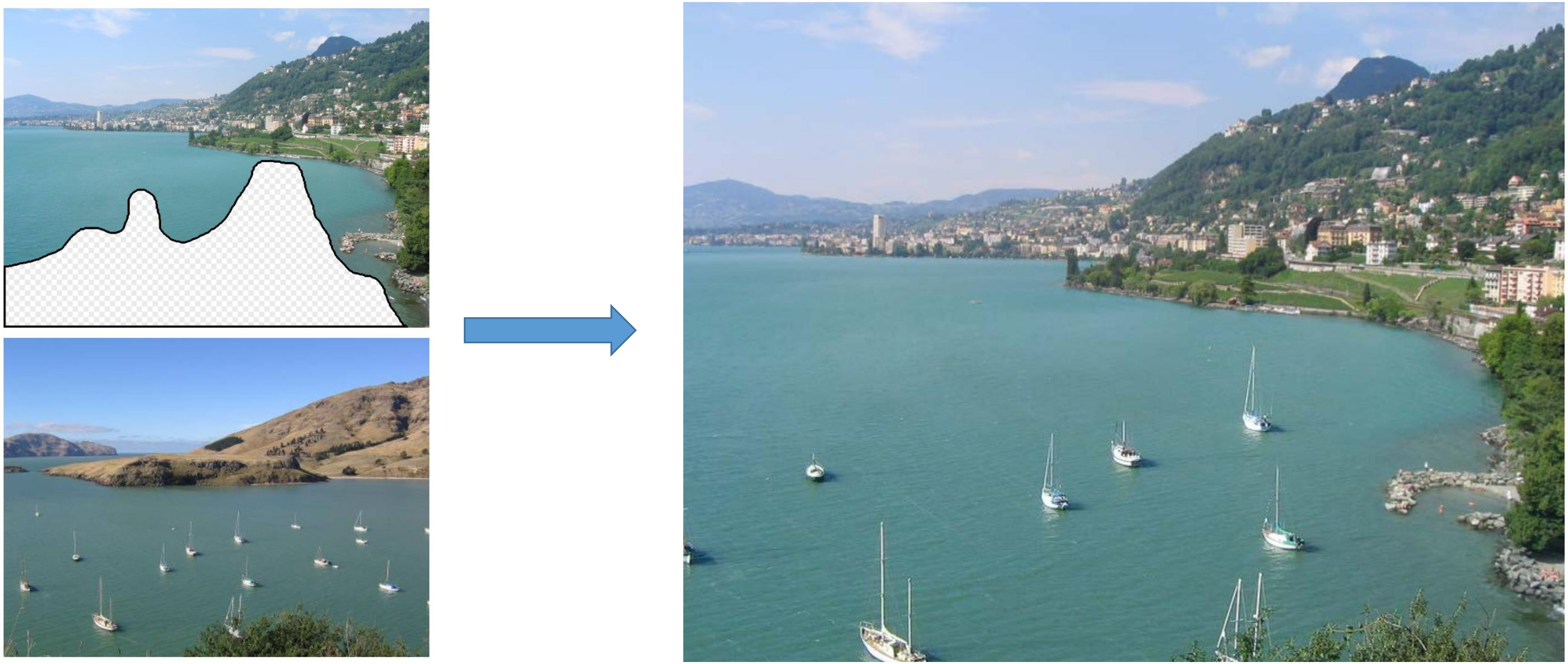
A drop-dead simple approach to classification

Prediction

True class

	0	1	2	3	4	5	6	7	8	9
0	83	1	1	0	0	0	5	0	10	0
1	0	100	0	0	0	0	0	0	0	0
2	1	11	53	2	1	0	3	4	25	0
3	0	11	2	48	0	1	4	3	28	3
4	2	9	0	0	42	0	2	3	16	26
5	2	7	0	4	0	36	2	0	43	6
6	3	6	0	0	0	1	80	0	10	0
7	0	11	0	1	0	0	1	75	4	8
8	2	13	0	6	1	3	3	4	65	3
9	0	5	1	1	4	0	0	4	2	83

Some more things you can do with nearest neighbours



https://nn2017.mit.edu/wp-content/uploads/sites/5/2017/12/Efros-NIPS-NN-17.pdf

Next week

- Regression
  - Continuous input/output
- Allows us to develop many key ideas in a rigorous way
  - Objective functions
  - Model capacity (under/over-fitting)
  - Regularisation
  - Cross-validation
  - Probabilistic methods