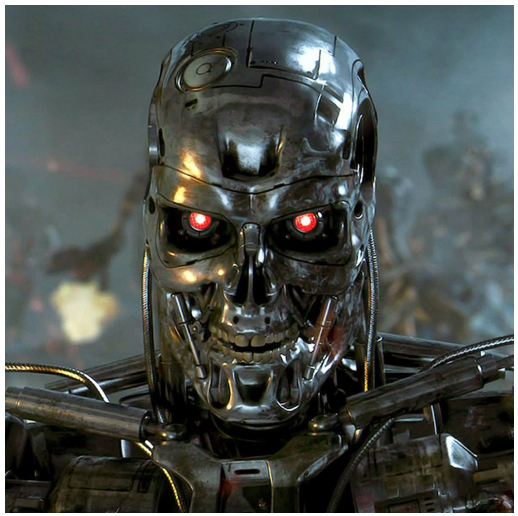
# Mathematics of Machine Learning

Rajen D. Shah
Email: r.shah@statslab.cam.ac.uk

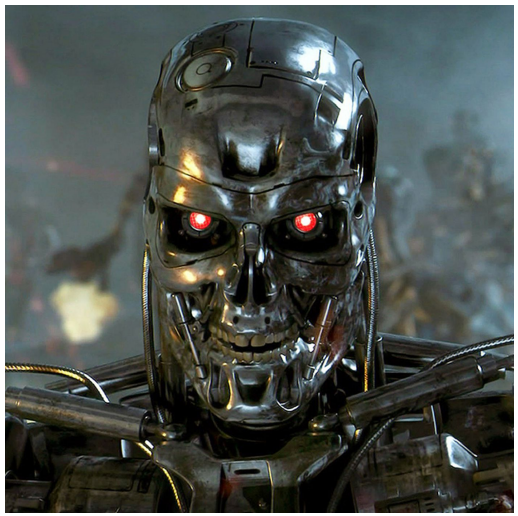
January 20, 2022

# Machine Learning



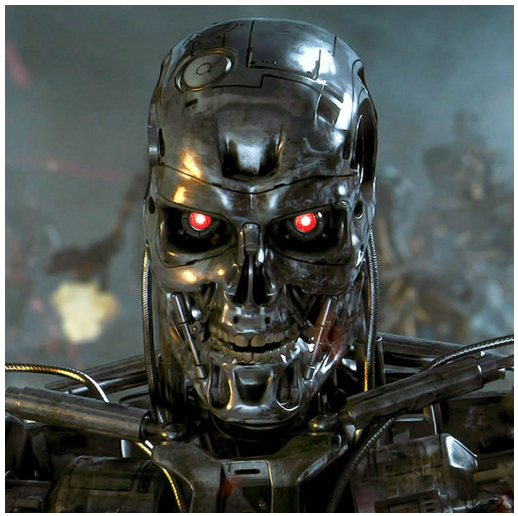
# Machine Learning

- Machine learning is the science of learning from data.



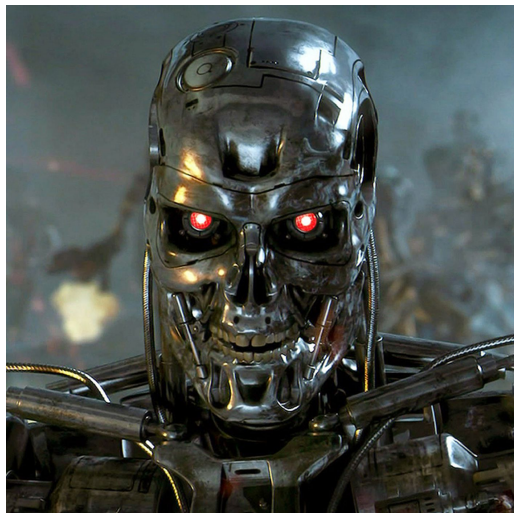
# Machine Learning

- Machine learning is the science of learning from data.
- Statistics is the science of learning from data.
- Overall aims of Statistics and Machine Learning are identical.

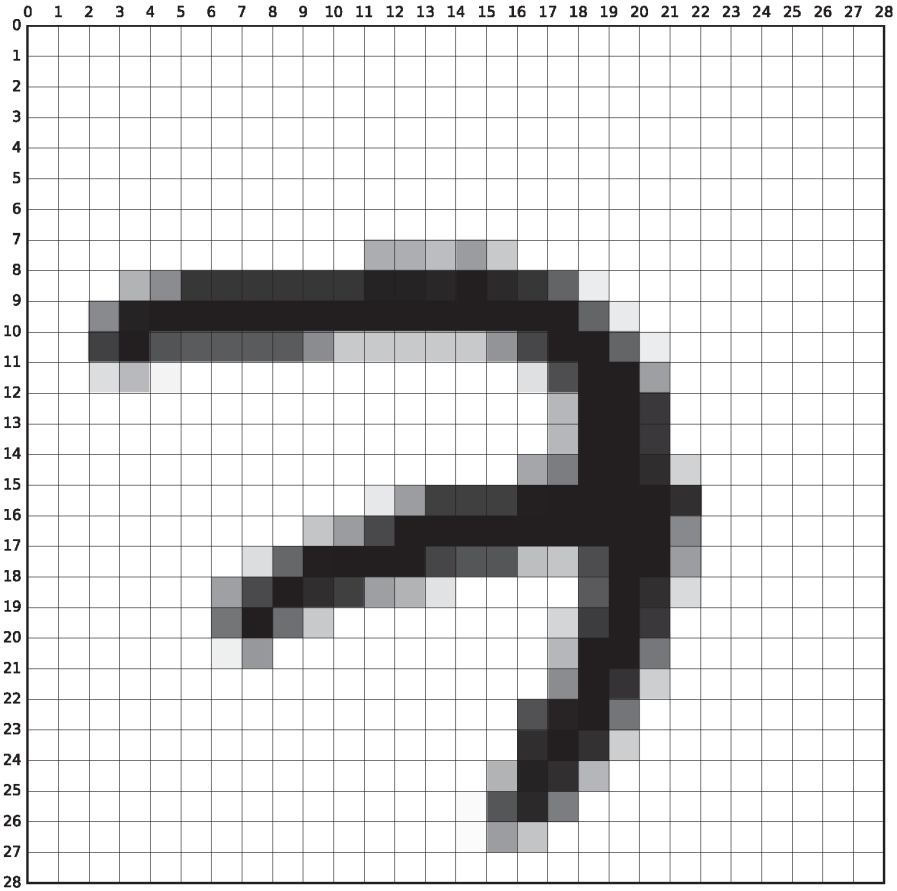


# Machine Learning

- Machine learning is the science of learning from data.
- Statistics is the science of learning from data.
- Overall aims of Statistics and Machine Learning are identical.
- Different histories:
  Mathematics $\rightarrow$ Statistics
  Computer Science $\rightarrow$ Machine Learning.
- (Traditionally) Emphasis on different sorts of problems.



# Handwritten digit recognition



(**a**) MNIST sample belonging to the digit '7'.

(**b**) 100 samples from the MNIST training set.

# Spam detection

To,

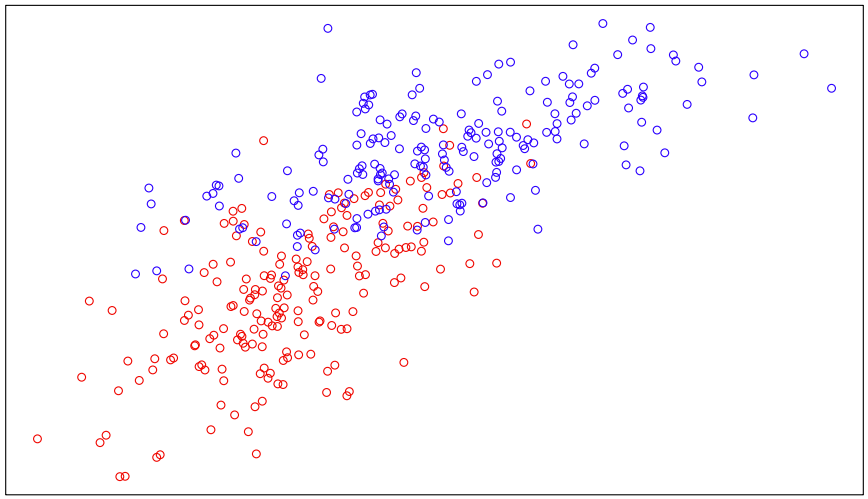
The Department of Pure Mathematics
and Mathematical Statistics (DPMMS)
University of Cambridge.
Subject – Application to inform that, I had invented a formula to divide any number by Zero.

Respected Sir / Madam,

I want to inform you that till now a date there is no any method in mathematics which is capable of dividing any number by zero. All the early attempts to find out the value of any number divided by zero is unsuccessful and as a result in modern mathematics it was assumed that 'Zero divided by Zero is indeterminate and any positive or negative number divided by zero is complex infinity.

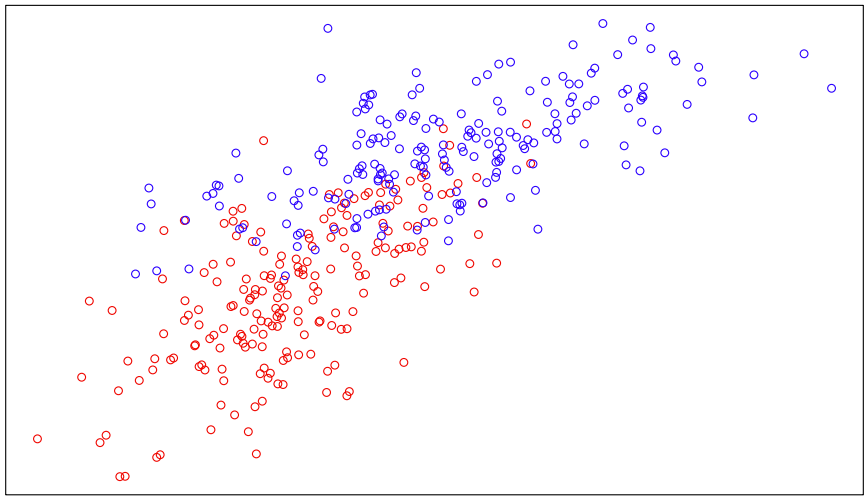
Sir / Madam, I need your help to use this formula on the benefit of humanity with safeguarding this formula from unsocial elements.

# Prediction problem



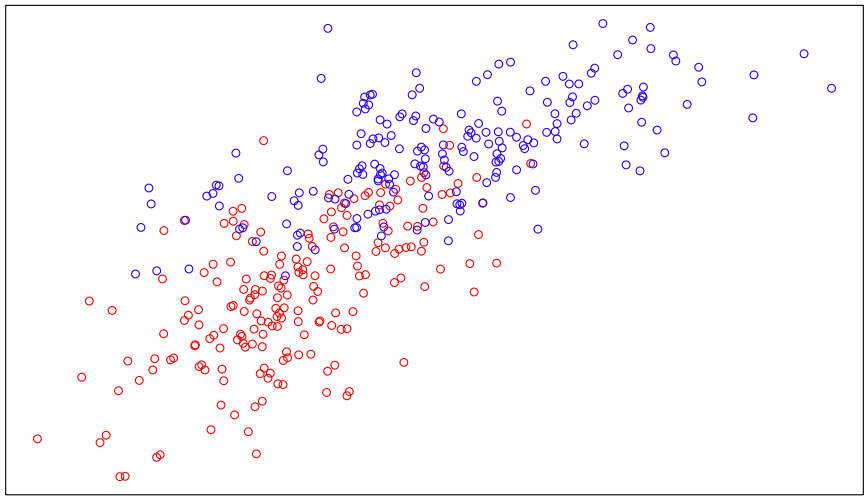
- Have *training data* consisting of input–output pairs $(X_1, Y_1), \ldots, (X_n, Y_n) \in \mathbb{R}^p \times \{-1, 1\}$.
- Construct a *hypothesis* or *classifier* $\hat{h} : \mathbb{R}^p \to \{-1, 1\}$.
- Aim to 'predict' the unknown $Y$ in new pairs $(X, Y)$ via $\hat{h}(X)$.

# Prediction problem



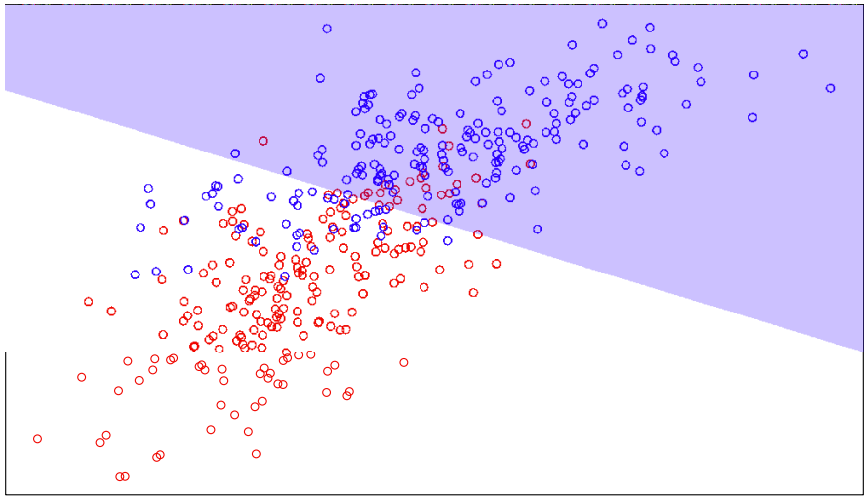
- Have *training data* consisting of input–output pairs $(X_1, Y_1), \ldots, (X_n, Y_n) \in \mathbb{R}^p \times \{-1, 1\}$.
- Construct a *hypothesis* or *classifier* $\hat{h} : \mathbb{R}^p \to \{-1, 1\}$.
- Aim to 'predict' the unknown $Y$ in new pairs $(X, Y)$ via $\hat{h}(X)$.
- Many classification algorithms: support vector classifiers, decision trees, neural networks, Adaboost, gradient boosting,...

# Prediction problem



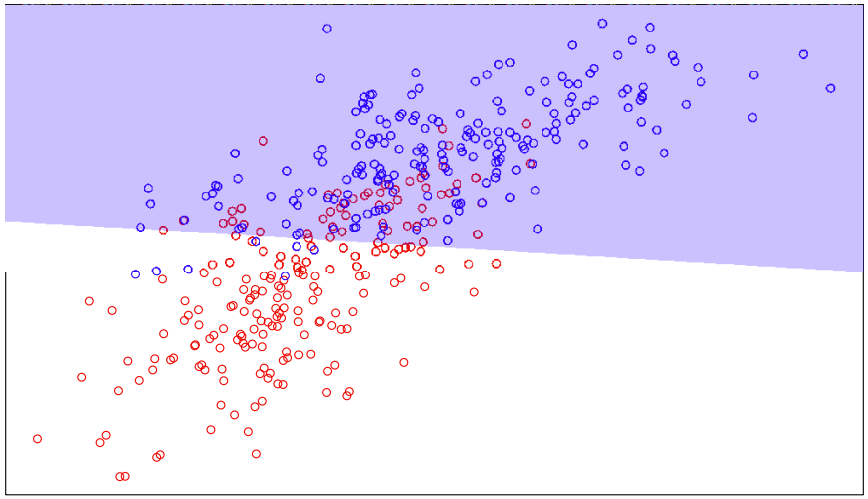
- Have *training data* consisting of input–output pairs $(X_1, Y_1), \ldots, (X_n, Y_n) \in \mathbb{R}^p \times \{-1, 1\}$.
- Construct a *hypothesis* or *classifier* $\hat{h} : \mathbb{R}^p \to \{-1, 1\}$.
- Aim to 'predict' the unknown $Y$ in new pairs $(X, Y)$ via $\hat{h}(X)$.
- Many classification algorithms: support vector classifiers, decision trees, neural networks, Adaboost, gradient boosting,...
- *Empirical risk minimisation*: given a class of hypotheses $\mathcal{H}$, choose $\hat{h}$ to be the best performer in $\mathcal{H}$ on the training data.

# Prediction problem



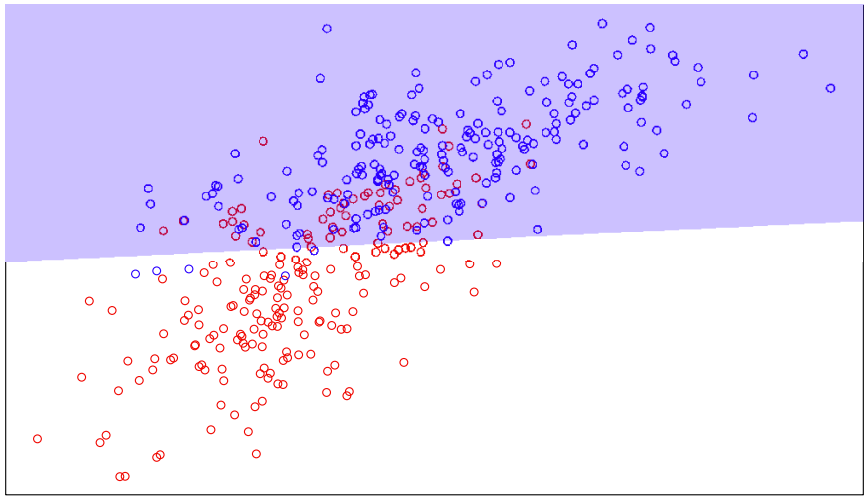
- Have training data consisting of input–output pairs $(X_1, Y_1), \ldots, (X_n, Y_n) \in \mathbb{R}^p \times \{-1, 1\}$.
- Construct a *hypothesis* or *classifier* $\hat{h} : \mathbb{R}^p \to \{-1, 1\}$.
- Aim to 'predict' the unknown $Y$ in new pairs $(X, Y)$ via $\hat{h}(X)$.
- Many classification algorithms: support vector classifiers, decision trees, neural networks, Adaboost, gradient boosting,...
- *Empirical risk minimisation*: given a class of hypotheses $\mathcal{H}$, choose $\hat{h}$ to be the best performer in $\mathcal{H}$ on the training data.

# Prediction problem



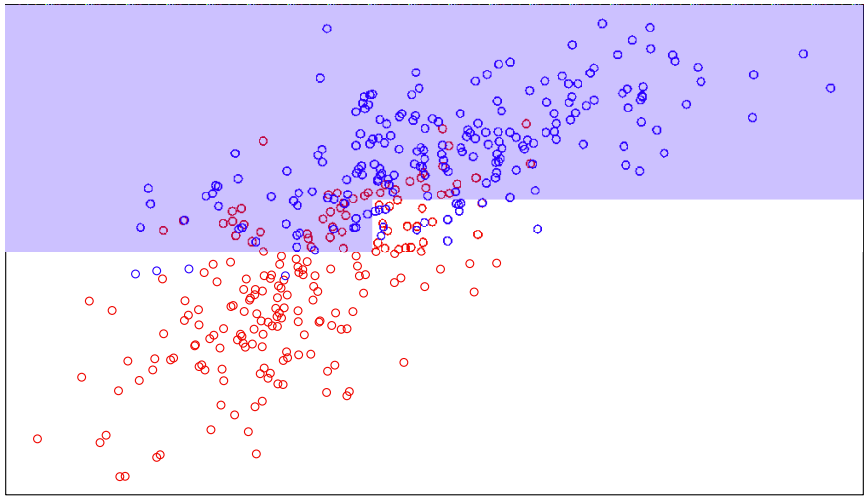
- Have training data consisting of input–output pairs $(X_1, Y_1), \ldots, (X_n, Y_n) \in \mathbb{R}^p \times \{-1, 1\}$.
- Construct a *hypothesis* or *classifier* $\hat{h} : \mathbb{R}^p \to \{-1, 1\}$.
- Aim to 'predict' the unknown $Y$ in new pairs $(X, Y)$ via $\hat{h}(X)$.
- Many classification algorithms: support vector classifiers, decision trees, neural networks, Adaboost, gradient boosting,...
- *Empirical risk minimisation*: given a class of hypotheses $\mathcal{H}$, choose $\hat{h}$ to be the best performer in $\mathcal{H}$ on the training data.

# Prediction problem



- Have training data consisting of input–output pairs $(X_1, Y_1), \ldots, (X_n, Y_n) \in \mathbb{R}^p \times \{-1, 1\}$.
- Construct a *hypothesis* or *classifier* $\hat{h} : \mathbb{R}^p \to \{-1, 1\}$.
- Aim to 'predict' the unknown $Y$ in new pairs $(X, Y)$ via $\hat{h}(X)$.
- Many classification algorithms: support vector classifiers, decision trees, neural networks, Adaboost, gradient boosting,...
- *Empirical risk minimisation*: given a class of hypotheses $\mathcal{H}$, choose $\hat{h}$ to be the best performer in $\mathcal{H}$ on the training data.

# Prediction problem



- Have training data consisting of input–output pairs $(X_1, Y_1), \ldots, (X_n, Y_n) \in \mathbb{R}^p \times \{-1, 1\}$.
- Construct a *hypothesis* or *classifier* $\hat{h} : \mathbb{R}^p \to \{-1, 1\}$.
- Aim to 'predict' the unknown $Y$ in new pairs $(X, Y)$ via $\hat{h}(X)$.
- Many classification algorithms: support vector classifiers, decision trees, neural networks, Adaboost, gradient boosting,...
- *Empirical risk minimisation*: given a class of hypotheses $\mathcal{H}$, choose $\hat{h}$ to be the best performer in $\mathcal{H}$ on the training data.

# Course Structure

1. **Statistical learning theory**
   - Theory for ERM. How does the complexity of $\mathcal{H}$ relate to the generalisation error?
   - Why study this?
     - Important area of probability theory: Concentration inequalities.
     - Should change the way you think about machine learning methods.

# Course Structure

1. **Statistical learning theory**
   - Theory for ERM. How does the complexity of $\mathcal{H}$ relate to the generalisation error?
   - Why study this?
     - Important area of probability theory: Concentration inequalities.
     - Should change the way you think about machine learning methods.
2. **Computation**
   - Modifying the basic ERM problem to make it computationally feasible.
   - $\ell_2$ and $\ell_1$ regularisation.
   - (Stochastic) gradient descent.

# Course Structure

1. **Statistical learning theory**
   - Theory for ERM. How does the complexity of $\mathcal{H}$ relate to the generalisation error?
   - Why study this?
     - Important area of probability theory: Concentration inequalities.
     - Should change the way you think about machine learning methods.
2. **Computation**
   - Modifying the basic ERM problem to make it computationally feasible.
   - $\ell_2$ and $\ell_1$ regularisation.
   - (Stochastic) gradient descent.
3. **Popular machine learning methods**
   - Cross-validation, Adaboost, gradient boosting, decision trees, random forests, neural networks.

# Prerequisites

- **Probability**
  - Comfortable with arguments involving conditioning e.g. manipulating conditional expectations
- **Optimisation**
  - Properties of convex functions and convex sets
- **Statistics**
  - Familiarity with linear regression may be helpful for basic intuition
- **Analysis and Topology**
  - Closed sets in $\mathbb{R}^d$. A continuous function on a closed bounded set attains its bounds.

# Other resources

- **Understanding Machine Learning: From Theory to Algorithms** (S. Shalev-Shwartz and S. Ben-David).
- **Stanford lecture notes** (P. Liang). Chapter 3 is great for the first part of our course.
- **MIT lecture notes** (P. Rigollet) - Part II is particularly good for the second part of our course.
- **High-Dimensional Statistics: A Non-Asymptotic Viewpoint** (M. Wainwright). At a higher mathematical level than our course.
- **The Elements of Statistical Learning** (T. Hastie, R. Tibshirani and J. Friedman). Less mathematical, useful for background and intuition.

# Admin

- Course notes available on the course website.
- Feedback form.
- R demos (no need to learn R for exam purposes).
- Exams: Revision sheet and pointers to example sheet questions.