CS135
Introduction to Machine Learning

Lecture 0: Course Outline & Intro to ML

Instructor

- Instructor: Joseph (Joe) Robinson
- E-mail: jrobs.vision@gmail.com
- Personal Webpage: http://www.jrobs-vision.com/

Tufts

About Me

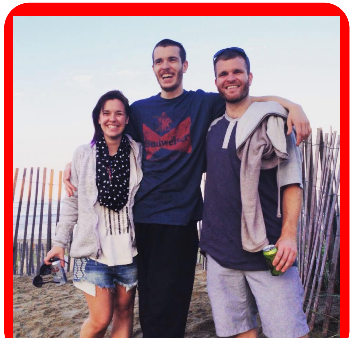
Joseph (Joe) Robinson

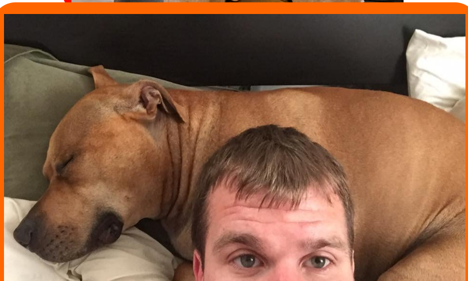
Northeastern University

Electrical & Computer Engineering (B.S.)
PhD, SMILE Lab (2020)

Academics and Research

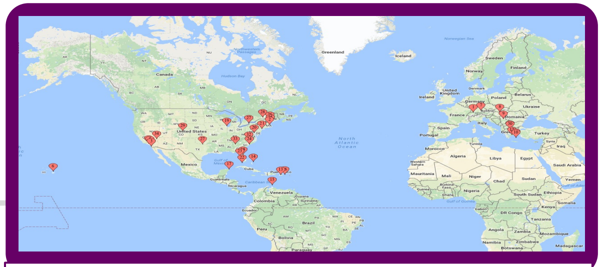
TRECVID





Family

Day in a life



Travelled lots

My Career Path: In a Nutshell

Northern Essex Community College

Summer 2010

REU
RESEARCH EXPERIENCES FOR UNDERGRADUATES

Undergrad

Northeastern University, Boston, Massachusetts

Summer 2011

REU
RESEARCH EXPERIENCES FOR UNDERGRADUATES

1st Co-op

analogic

2nd Co-op

Raytheon
BBN Technologies

Grad

Northeastern University, Boston, Massachusetts

SMILE lab
Synergetic Media Learning Lab

Grad Internships

MIT Lincoln Laboratory

2x

STR SYSTEMS & TECHNOLOGY RESEARCH

Snap Inc.

ISM CONNECT

Grad Support

ALERT
AWARENESS AND LOCALIZATION OF EXPLOSIVES-RELATED THREATS

Paper Pile (https://scholar.google.com/citations?user=o-XLJIgAAAAJ&hl=en)

- **Joseph P. Robinson**, Zaid Khan, Yu Yin, Ming Shao, and Yun Fu. "Families In Wild Multimedia (FIW-MM): A Multi-Modal Database for Recognizing Kinship." *arXiv preprint arXiv*:2007.14509 (2020). [paper]
- **Joseph P. Robinson**, Ming Shao, and Yun Fu. "Visual Kinship Recognition: A Decade in the Making." *arXiv preprint arXiv:2006.16033* (2020). [paper]
- **Joseph P. Robinson**, Gennady Livitz, Yann Henon, Can Qin, Yun Fu, Samson Timoner. "Face Recognition: Too Bias, or Not Too Bias?." in *CVPR Workshop* (2020). [paper]
- Yu Yin, Songyao Jiang, **Joseph P. Robinson**, Yun Fu. "Dual-Attention GAN for Large-Pose Face Frontalization." in *IEEE Automatic Face and Gesture (FG) Recognition* (2020). [paper]
- **Joseph P. Robinson**, Yu Yin, Zaid Khan, Ming Shao, Siyu Xia, Michael Stopa, Samson Timoner, Matthew A. Turk, Rama Chellappa, Yun Fu. "Recognizing Families In the Wild (RFIW): The 4th Edition." in *IEEE Automatic Face and Gesture Recognition* (2020). [paper]
- Lichen Wang, Bin Sun, **Joseph P Robinson**, T Jing, and Yun Fu. "EV-Action: Electromyography-Vision Multi-Modal Action Dataset." in *IEEE FG* (2020). [paper]
- Yu Yin, **Joseph P Robinson**, Yulun Zhang, and Yun Fu. "Joint Super-Resolution and Alignment of Tiny Faces." in Conference on Artificial Intelligence (AAAI) (2020). [paper]
- W. Zhuang, Y. Wang, **Joseph P. Robinson**, C. Wang, M. Shao, Y. Fu, S. Xia, "Towards 3D Dance Motion Synthesis and Control." *CoRR arXiv:2006.05743* (2020). [paper]
- Pengyu Gao, Siyu Xia, **Joseph P Robinson**, Junkang Zhang, Chao Xia, Ming Shao, and Yun Fu. "What Will Your Child Look Like? DNA-Net: Age and Gender Aware Kin Face Synthesizer."CoRR *arXiv:1911.07014* (2020). [paper]
- **Joseph P Robinson**, Yuncheng Li, Ning Zhang, Yun Fu, and Sergey Tulyakov. "Laplace Landmark Localization." *in ICCV* (2019). [paper, poster]
- **Joseph P Robinson**, Ming Shao, Hongfu Liu, Yue Wu, Timothy Gillis, and Yun Fu. "Visual Kinship Recognition of Families In the Wild" IEEE TPAMI Special Edition: Computational Face (2018). [paper]
- Yue Wu, Zhengming Ding, Hongfu Liu, **Joseph P Robinson**, and Yun Fu. "Kinship Classification through Latent Adaptive Subspace," in *IEEE FG* (2018). [paper]
- **Joseph P Robinson**, Ming Shao, Handong Zhao, Yue Wu, Timothy Gillis, Yun Fu. "Recognizing Families In the Wild: Data Challenge Workshop," *ACM MM Workshop on RFIW* (2017). [paper, proceedings, contents]
- Shuyang Wang, **Joseph P Robinson**, and Yun Fu. "Kinship Verification on Families in the Wild with Marginalized Denoising Metric Learning," in *IEEE FG* (2017). [paper]
- **Joseph P Robinson**, Ming Shao, Yue Wu, and Yun Fu. "Families In the Wild (FIW): large-scale kinship image database and benchmarks." *In Proceedings of the 2016 ACM MM Conference* (2016). [paper]
- **Joseph P Robinson** and Yun Fu. "Pre-trained D-CNN models for detecting complex events in unconstrained videos." *SPIE Commercial + Scientific Sensing and Imaging* (2016). [paper]

Post Graduation

VICARIOUS SURGICAL

2020-2022

Northeastern University

Jan 2023

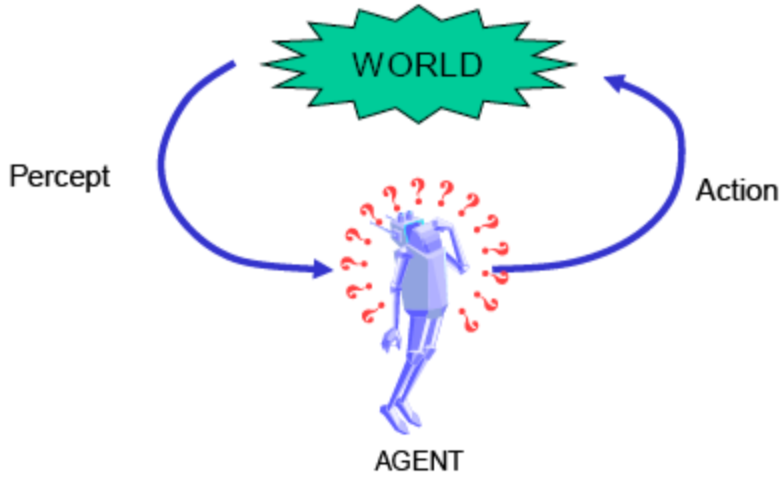
AEYEZONE
ON-SITE EXPLAINABLE AI

Co-Founder
[link]

Tufts

Intelligence

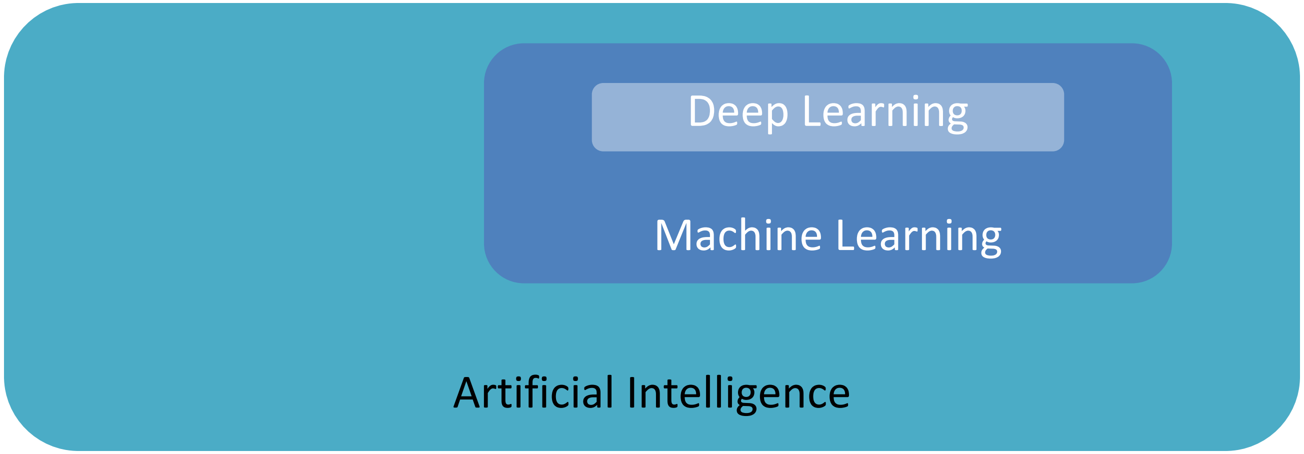
- Ability for abstract thought, understanding, communication, reasoning, planning, emotional intelligence, problem solving, **learning**
- The ability to learn and/or adapt is generally considered a hallmark of intelligence



What is Artificial Intelligence?

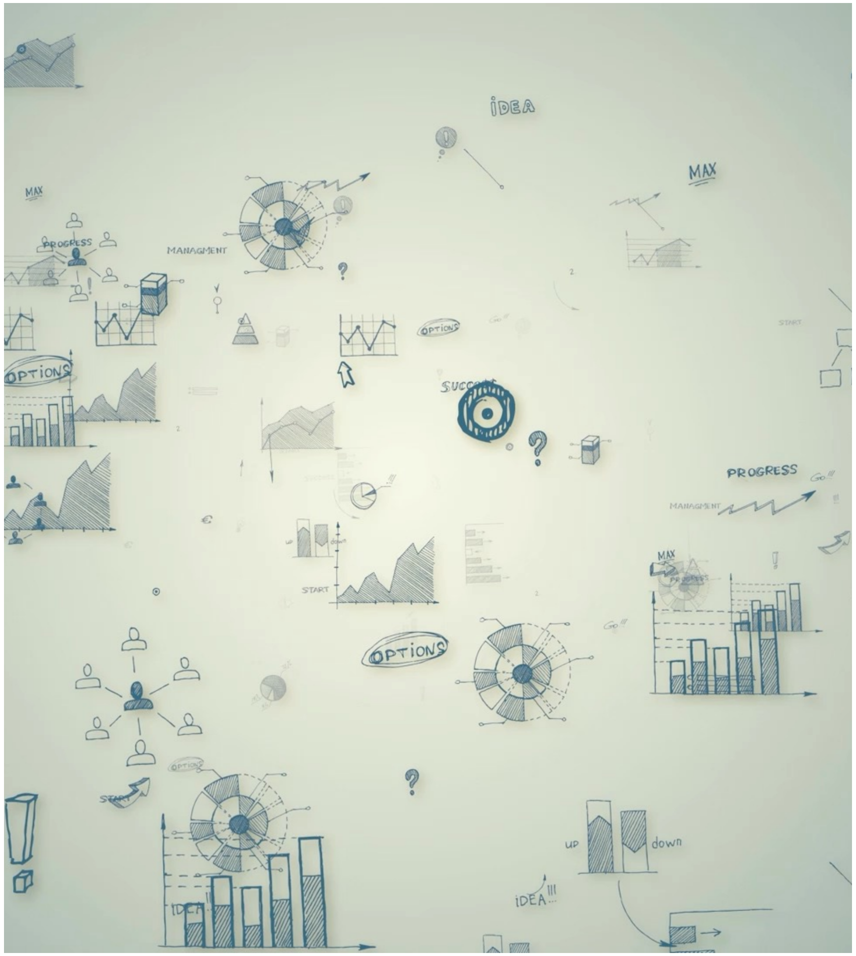
- Historical definition (Dartmouth Workshop on AI, 1956):

> "The study of the conjecture that every aspect of learning or any other feature of intelligence can in principle be so precisely described that a machine can be made to simulate it."



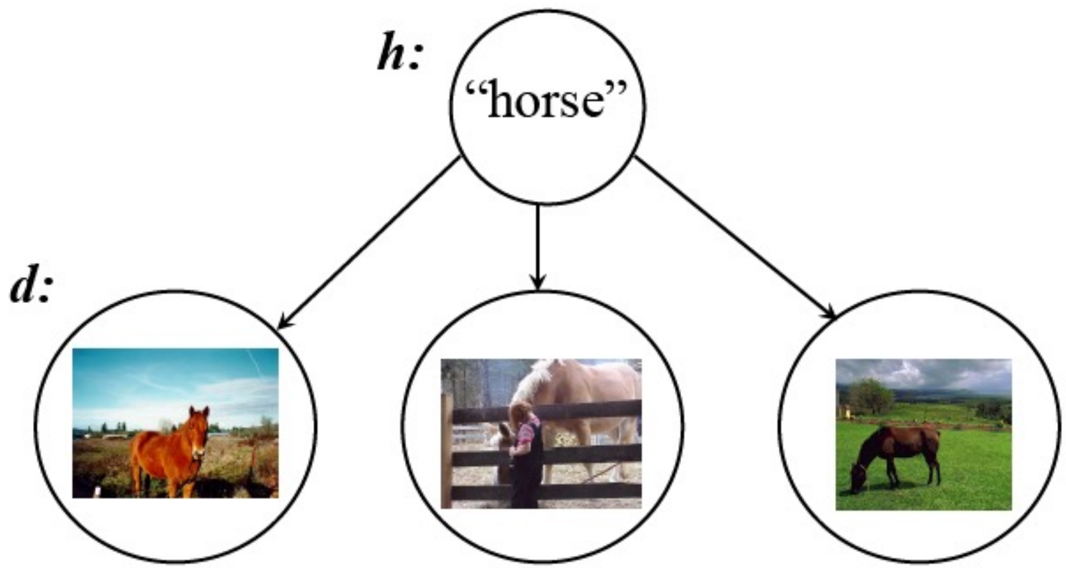
What is Machine Learning?

- **Machine learning** is a set of methods that automatically detect data patterns.
- These uncovered patterns are then used to predict future data or to perform other kinds of decision-making under uncertainty.
- The fundamental premise is *learning* from data!!



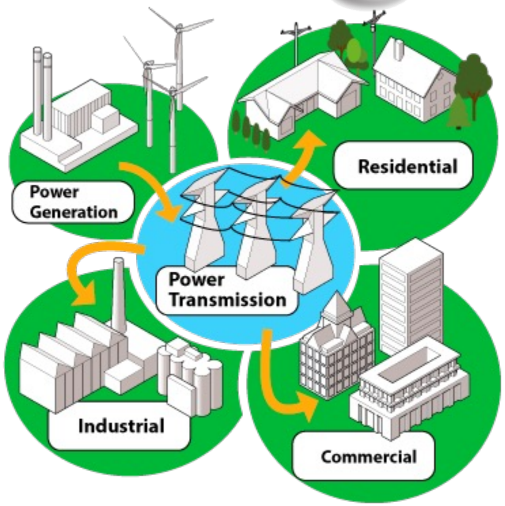
What is Machine Learning?

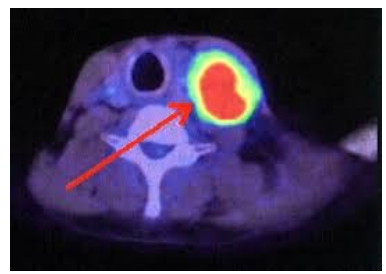
- "Machine learning studies the process of constructing abstractions (features, concepts, functions, relations and ways of acting) automatically from data."

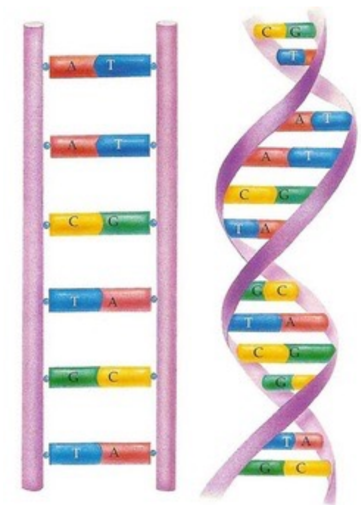


Tufts

The data all around us

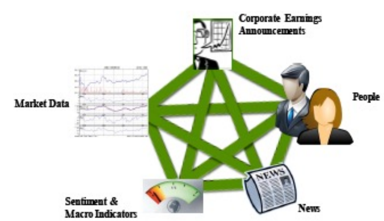




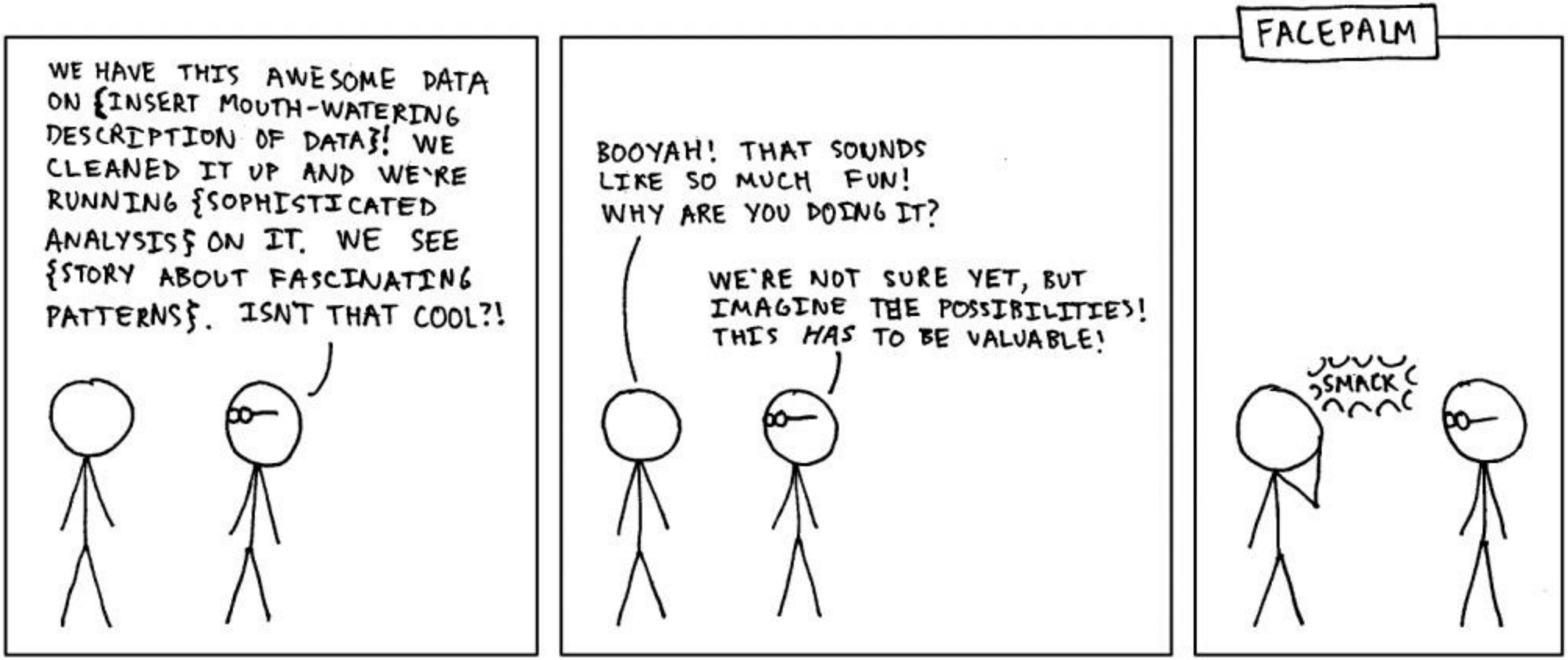


Machine Learning to the rescue

- Machine Learning is one of the front-line technologies to handle Information Overload
- **Business**
  - Mining correlations, trends, and spatio-temporal predictions.
  - Efficient supply chain management.
  - Opinion mining and sentiment analysis.
  - Recommender systems.

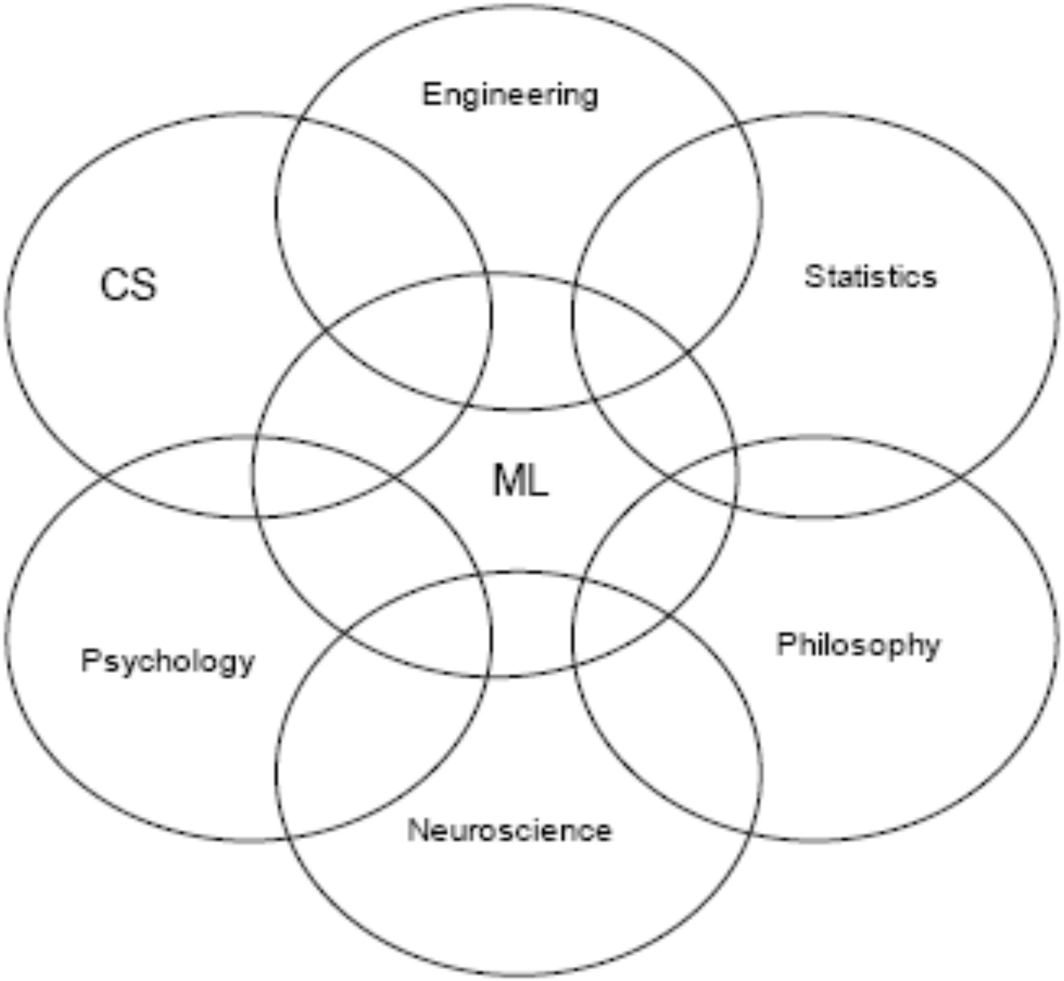


Keep the problem(s) in mind

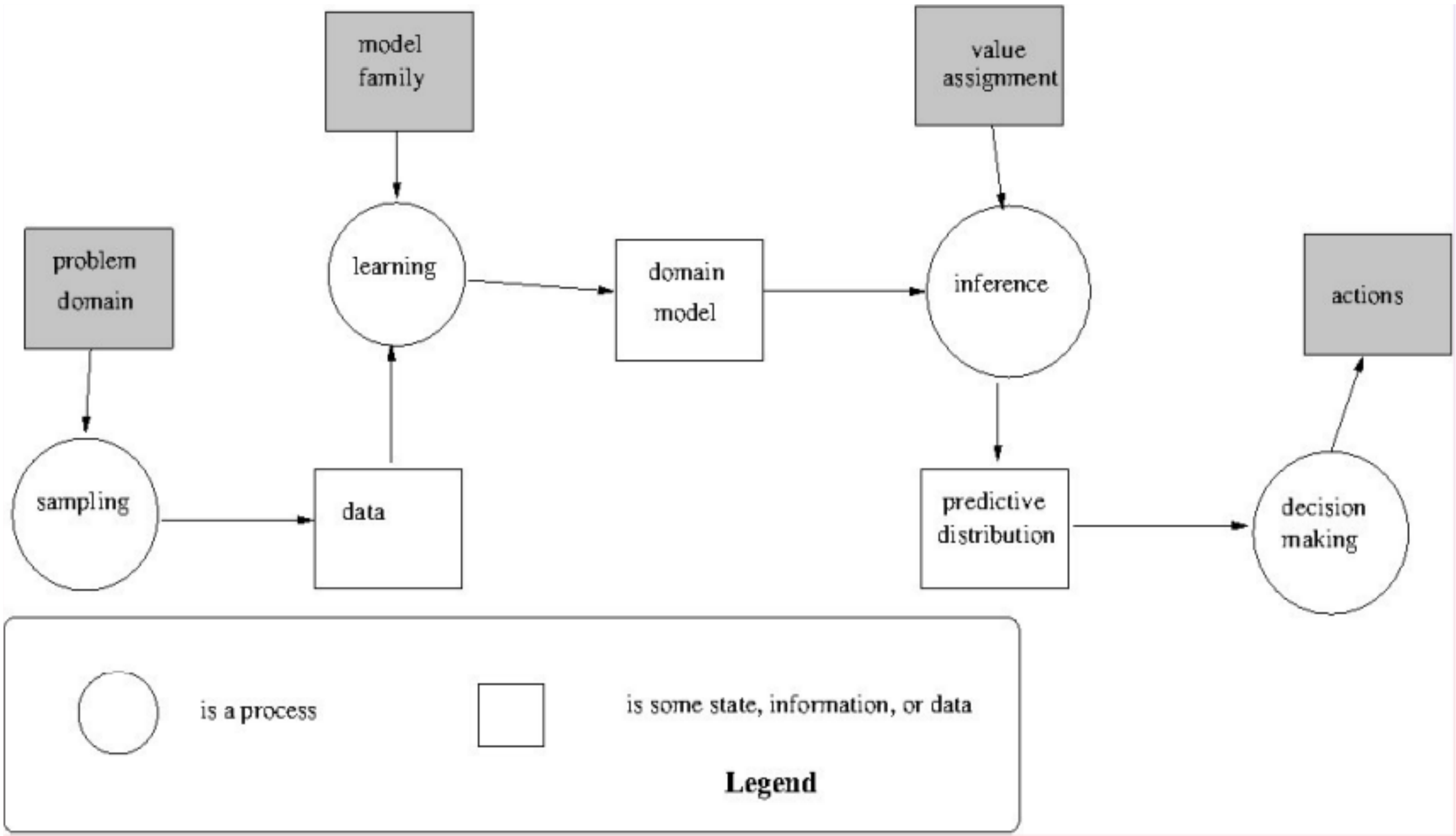


Tufts

Fields related to Machine Learning



Machine Learning Process



Tufts

Learning (Formally)

- **Task**
  - To apply some machine learning method to the data obtained from a given domain (*Training Data*)
  - The domain has some characteristics, which we are trying to learn (*Model*)
- **Objective**
  - To minimise the error in prediction
- **Types of Learning**
  - Supervised Learning
  - Unsupervised Learning
  - Semi-Supervised Learning
  - Active Learning

Supervised Learning

- Classification / Regression problem
- Where some samples of data (Training data) with the correct class labels are provided.
  - i.e. Some correspondence between input (**X**) & output (**Y**) given
- Using knowledge from training data, the classifier/regressor model is learnt
  - i.e. Learn some function **f : f(X) = Y**
- **f** may be probabilistic/deterministic
- Learning the model ≡ Fitting the parameters of model to minimise prediction error
- Model can then be tested on test-data

Regression

Regression

- Linear regression

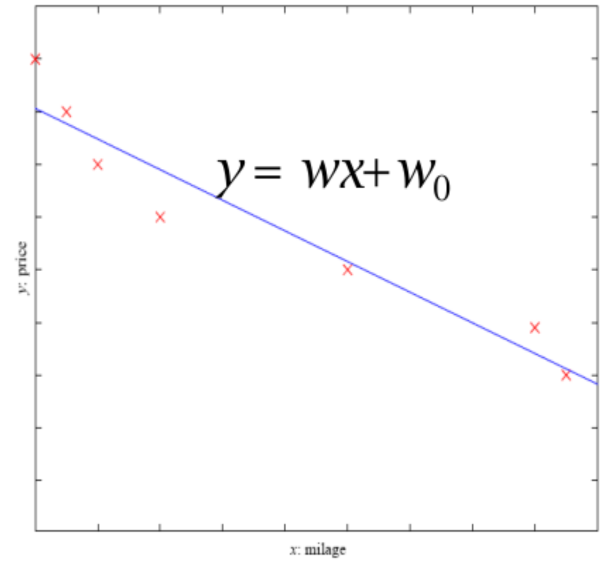
  - Example: Price of a used car
  - x : car attributes

    y : price

    $$y = g\ (x,\theta)$$

    g ( ) model,

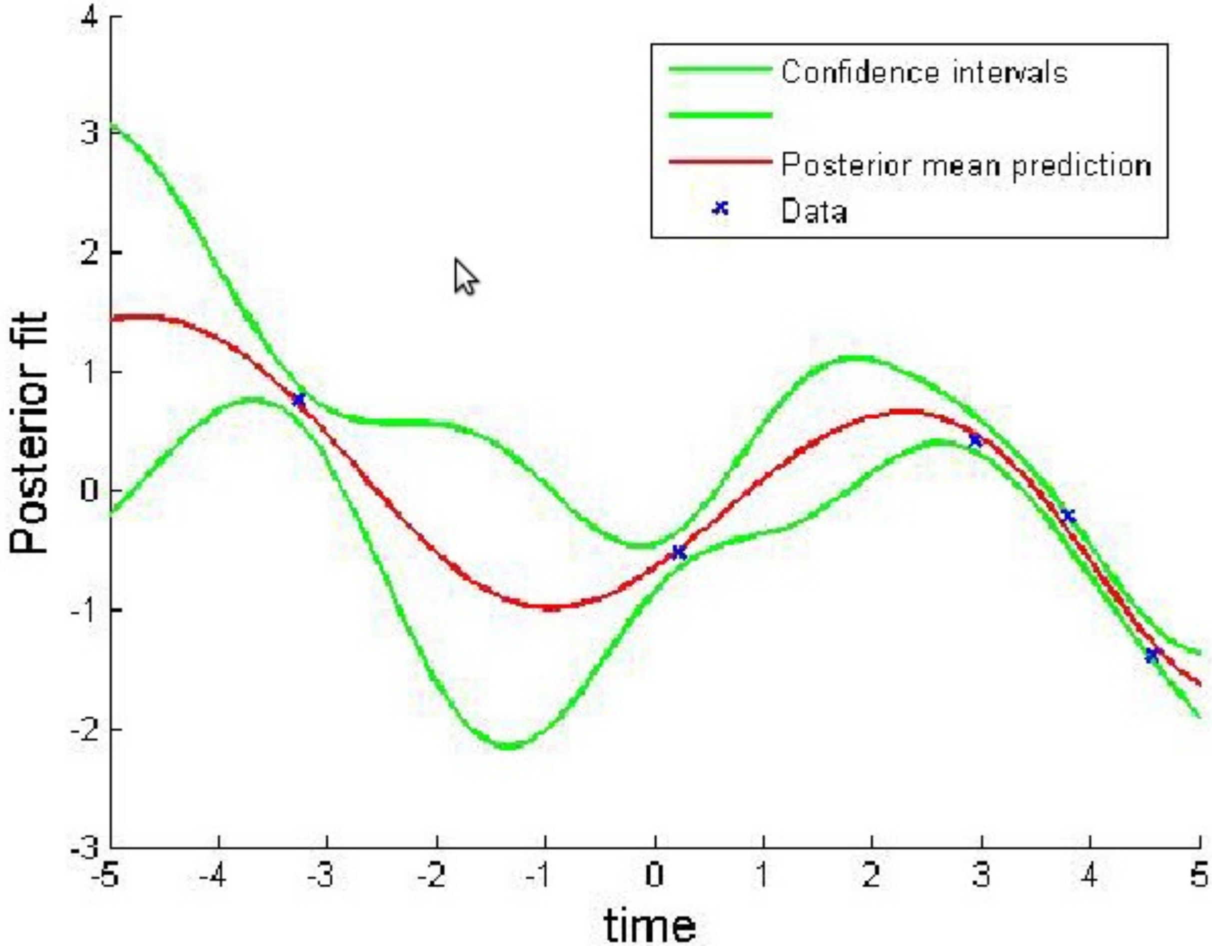
    $\theta = (w,w_0)$ parameters (slope and intercept)



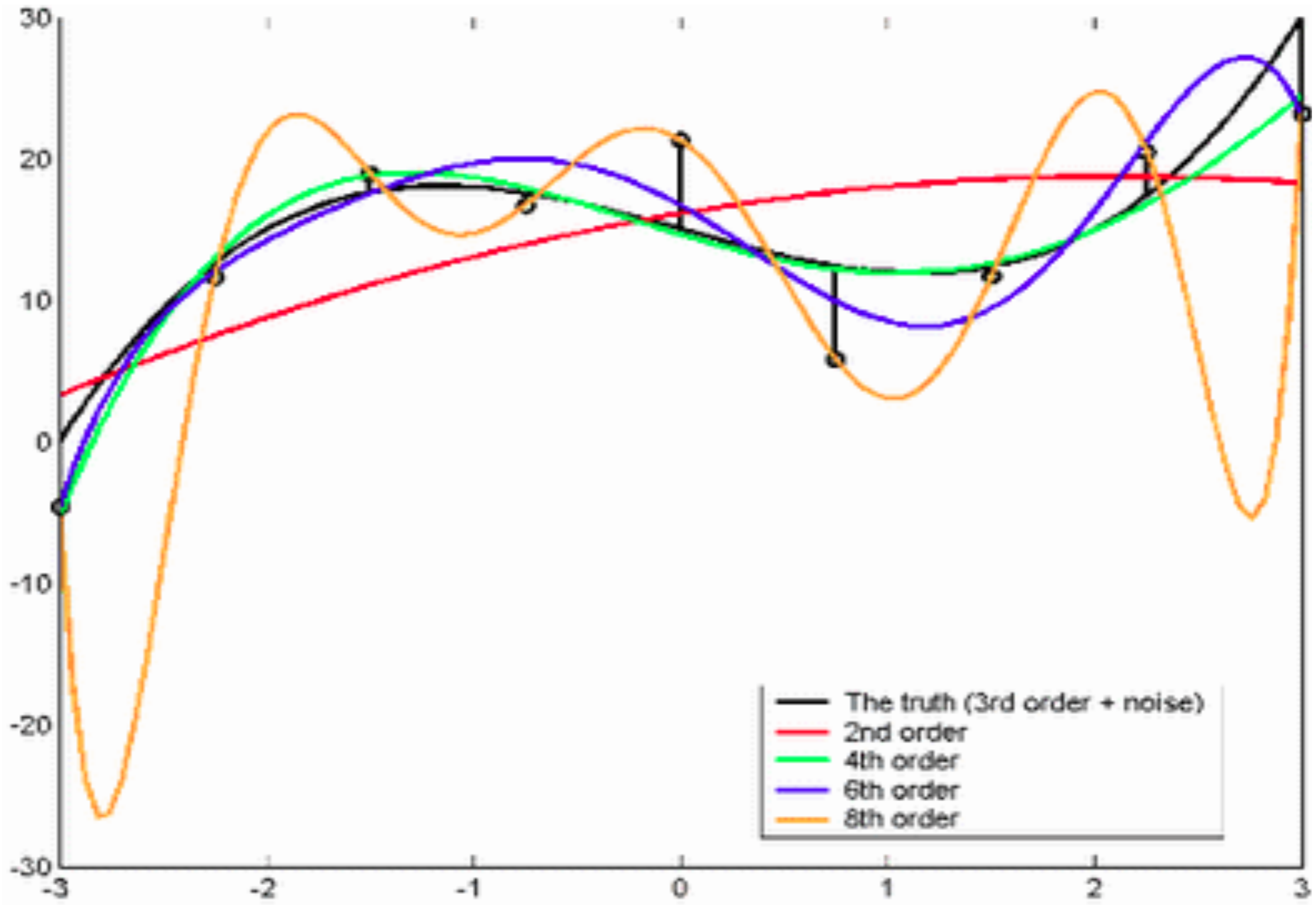
- Uses
  - Stock Prediction
  - Outlier detection

Regression

- Non Linear regression



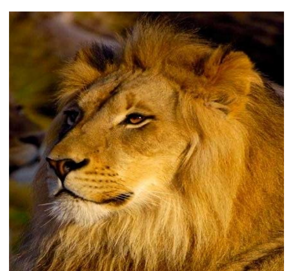
All models are not good



- Constrain the parameters

Classification

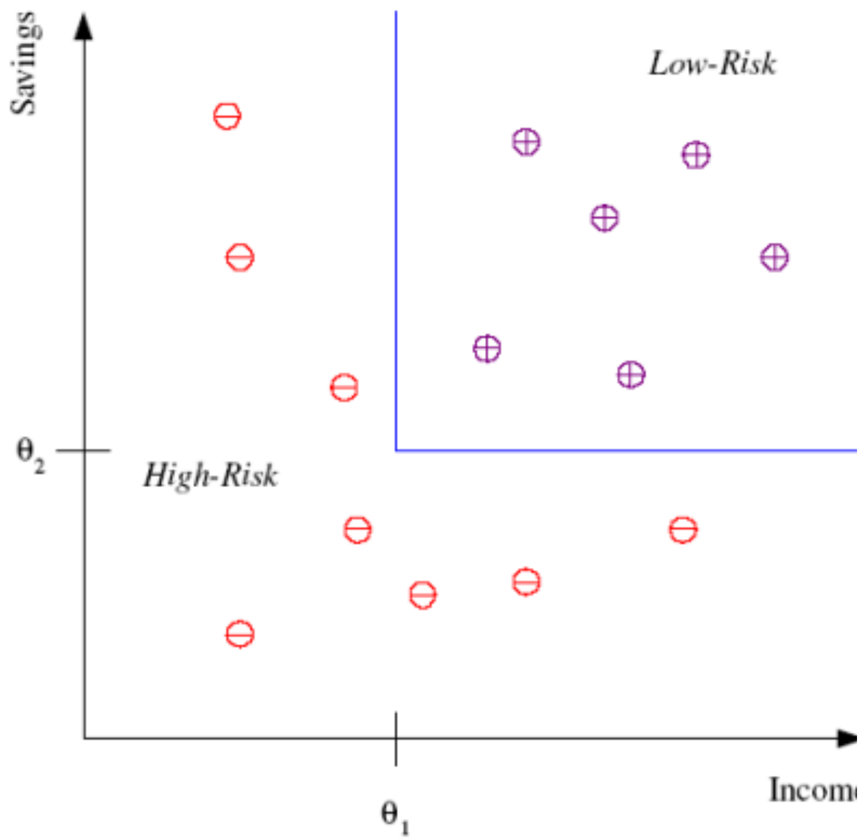
Supervised Classification example

	f1	f2	f3	f4	Class label
d1					BearHead
d2					???
d3					LionHead

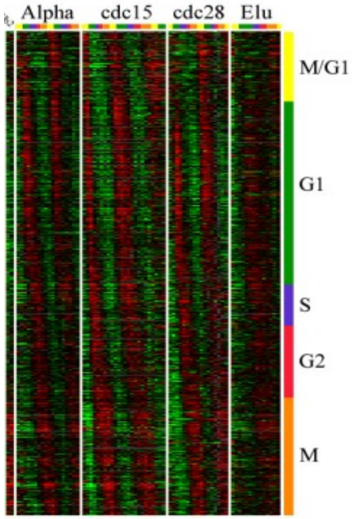
Source: LHI Animal Faces Dataset

Classification

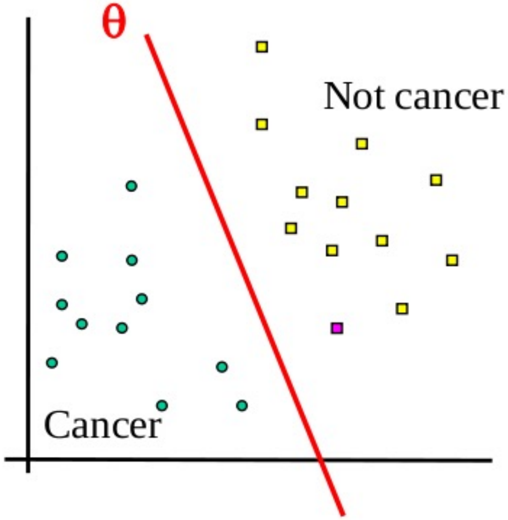
- Example:
  - Credit Scoring
- Goal:
  - Differentiating between high-risk and low-risk customers based on their income and savings
- Discriminant:
  - IF **income > θ_1** AND **savings > θ_2** THEN **low-risk** ELSE **high-risk**
- Discriminant is called '**hypothesis**'
- Input attribute space is called '**Feature Space**'
- Here Input data is 2-dimensional and the output is binary

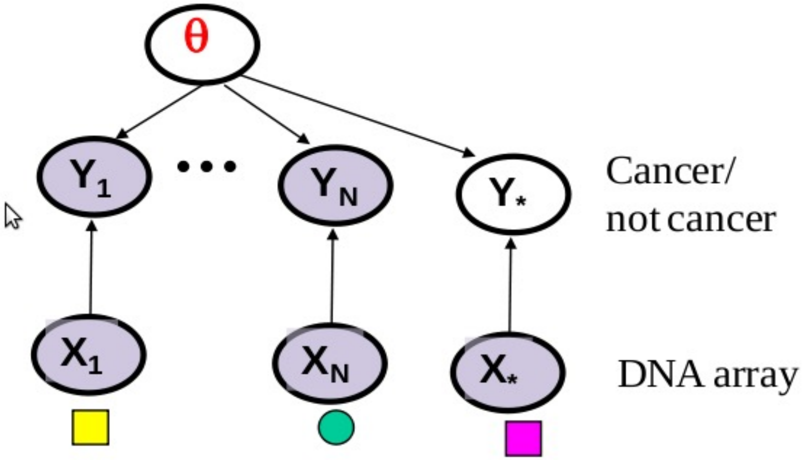


Other applications



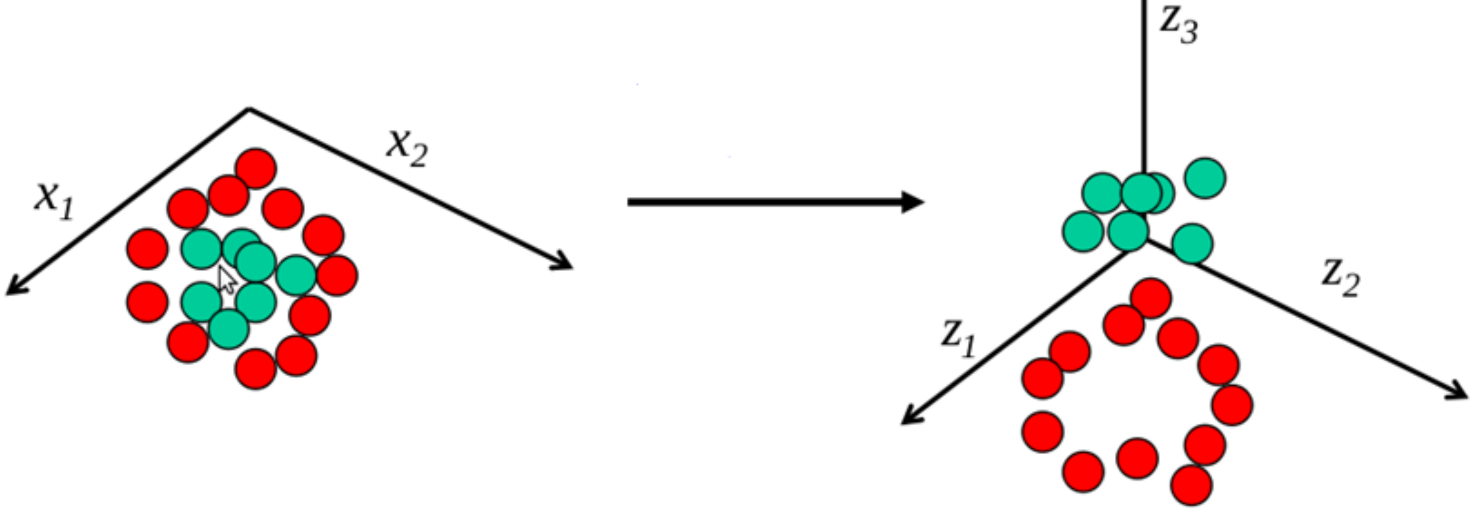
Classification of DNA arrays





Building non-linear classifiers

$$\begin{pmatrix} x \\ y \end{pmatrix} \Rightarrow \begin{pmatrix} x^2 \\ \sqrt{2}xy \\ y^2 \end{pmatrix}$$

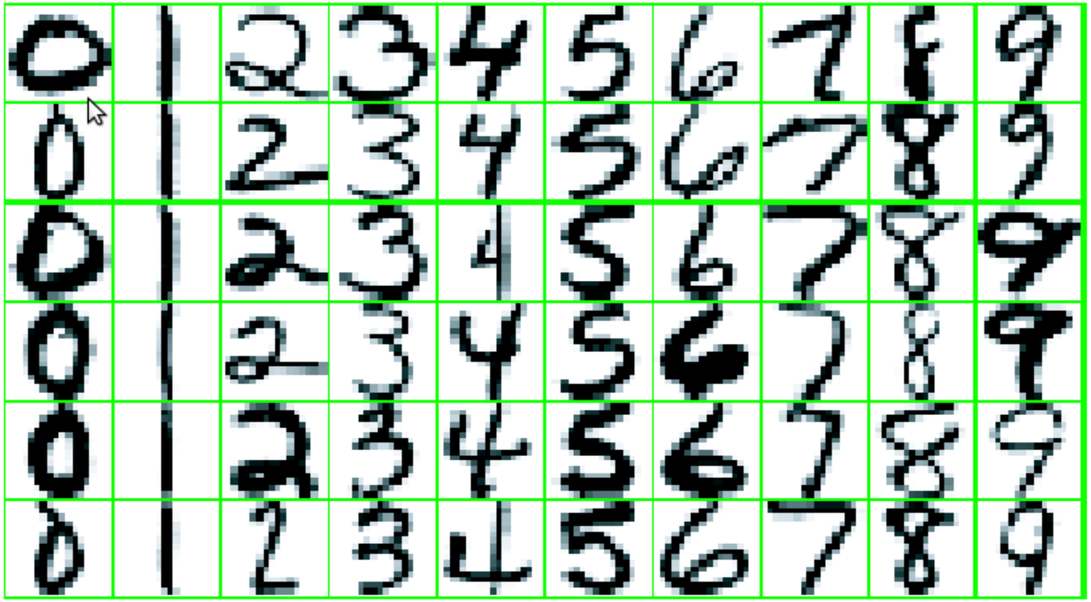


function implicitly maps from 2D to 3D, making problem linearly separable

- Curse of dimensionality

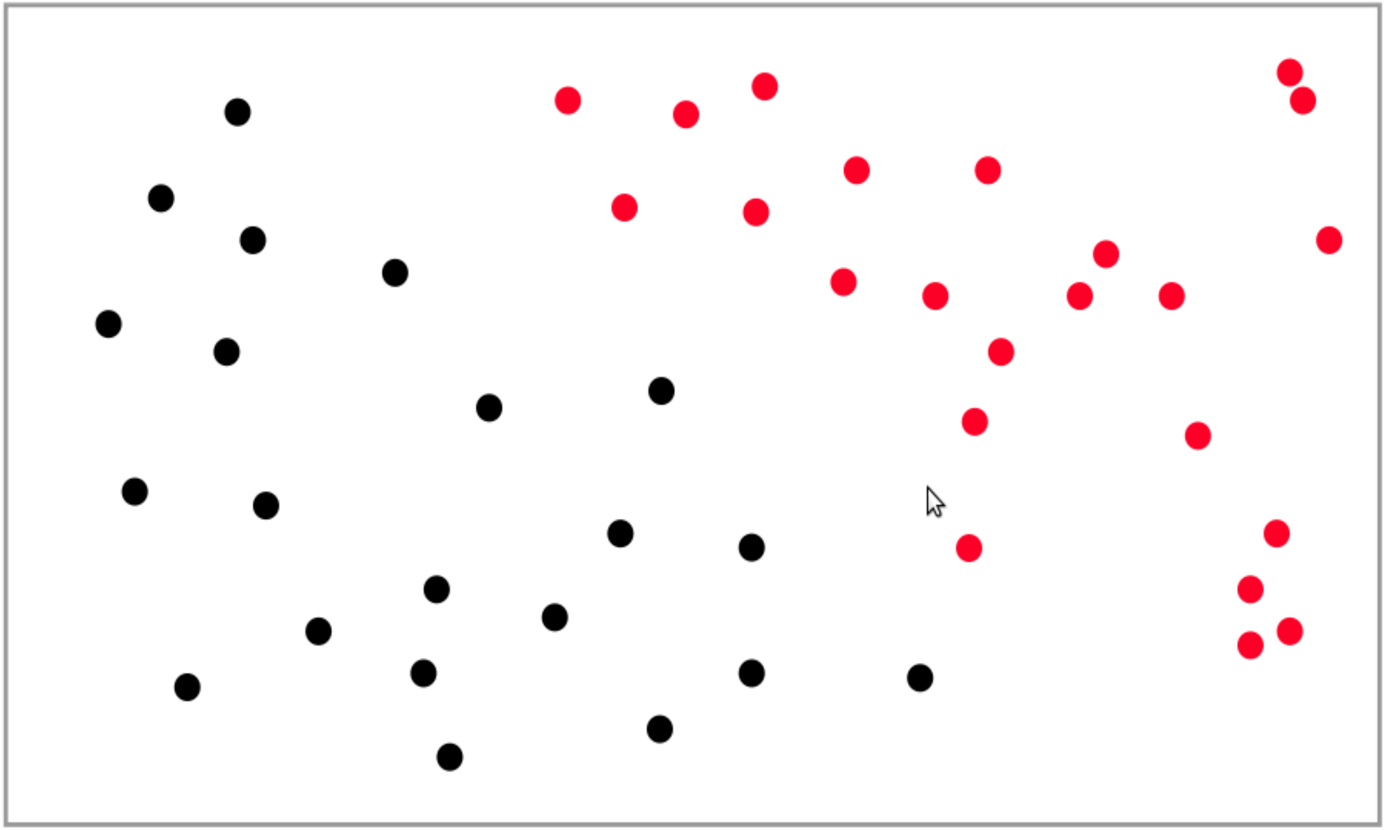
Application

Example: Handwritten digit recognition for postal codes

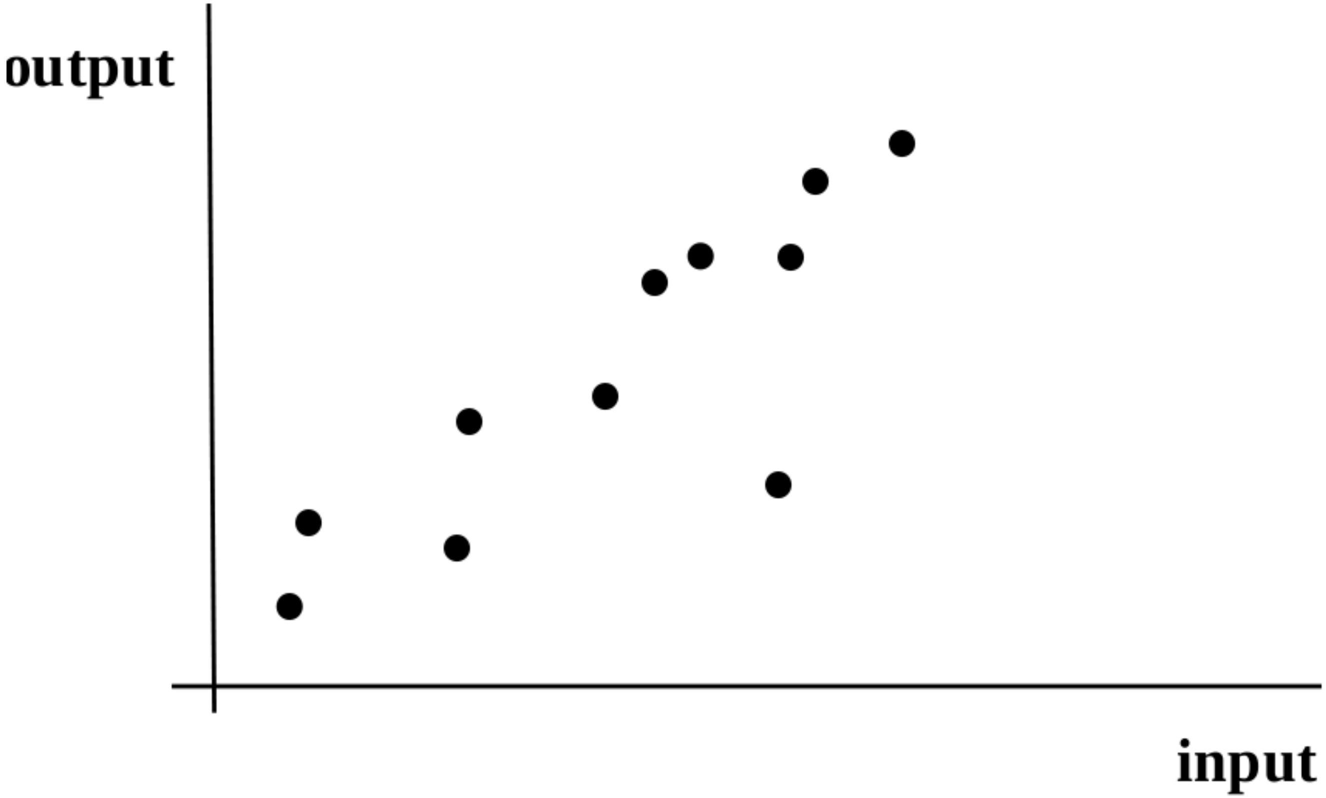


What is the right hypothesis?

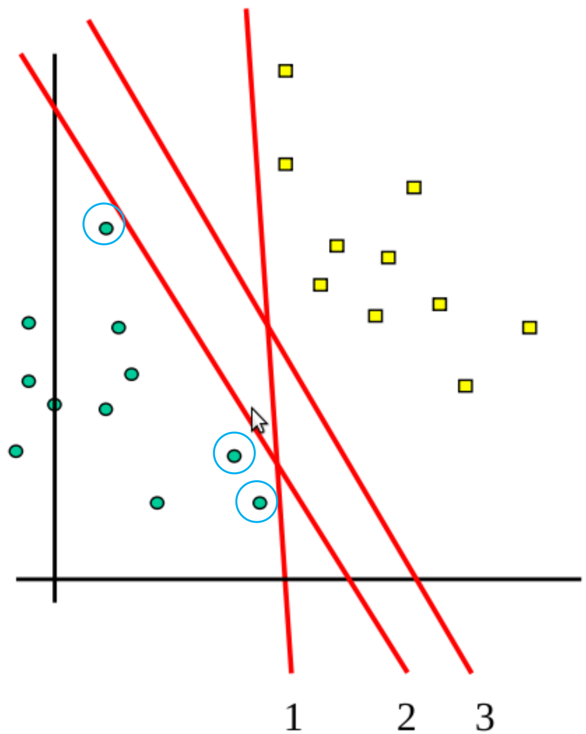
What is the correct hypothesis for this classification problem?

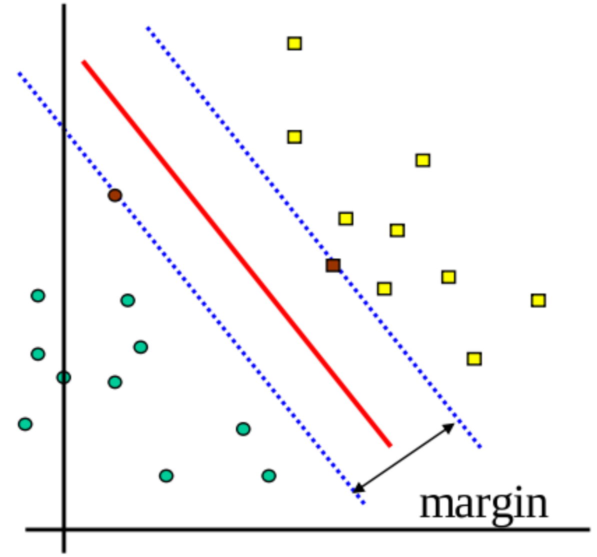


What is the correct hypothesis for this classification problem?



Which linear hypothesis is better?





- Max – Margin Classifier

Other considerations

- **Feature extraction**: which are the good features that characterize the data
- **Model selection**: picking the right model using some scoring/fitting function:
  - It is important not only to provide a good predictor but also to assess accurately how "good" the model is on unseen test data
  - So a good performance estimator is needed to rank the model
- **Model averaging**: Instead of picking a single model, it might be better to do a weighted average over the best-fit models

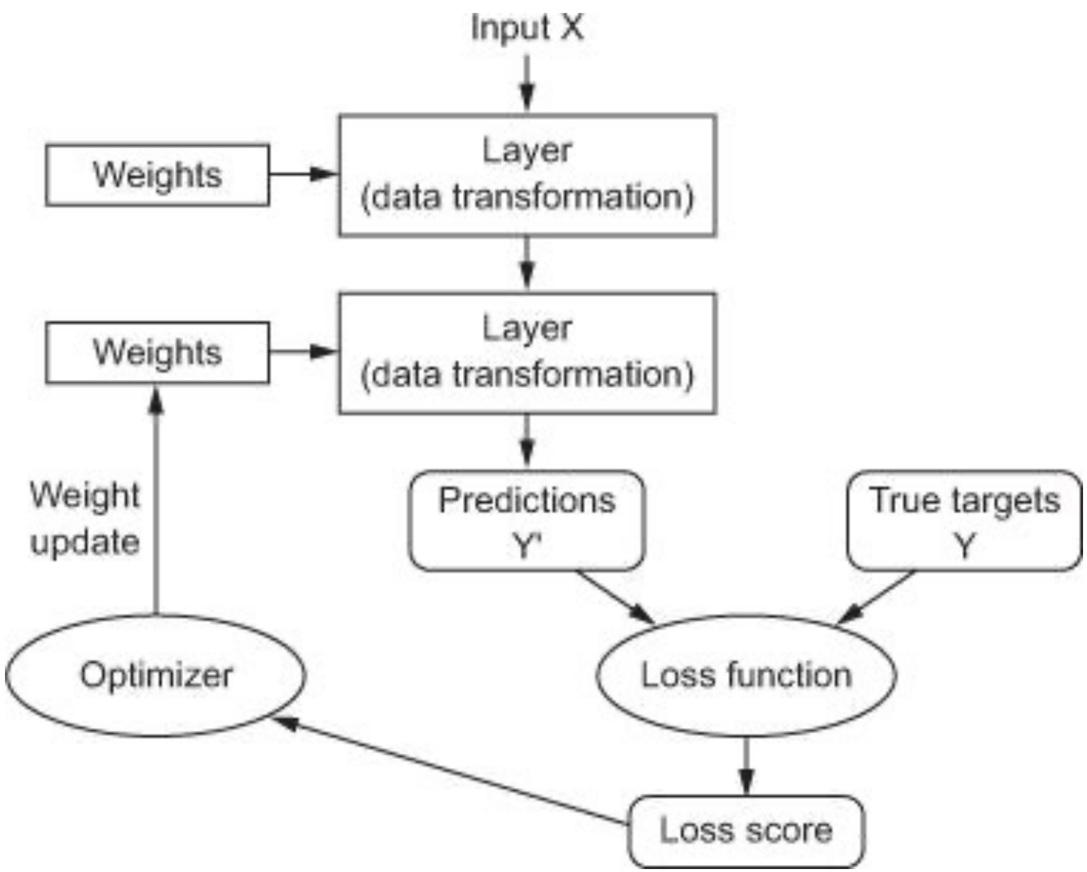
Which hypothesis is better?

- Unless you know something about the distribution of problems your learning algorithm will encounter, ***any hypothesis that agrees with all your data is as good as any other.***
- You have to make assumptions about the underlying features.
- Hence learning is inductive, not deductive.

Deep Learning

Anatomy of a deep neural network

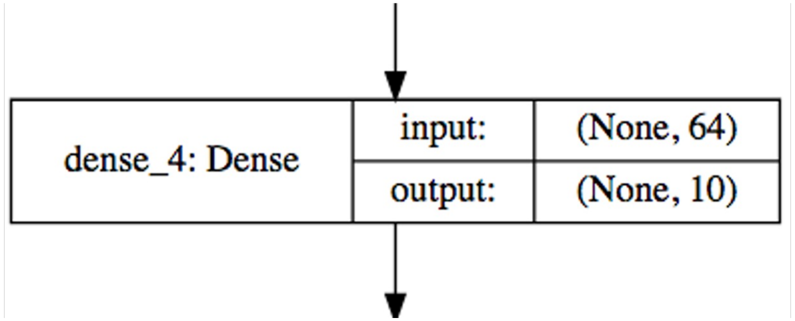
- Layers
- Input data and targets
- Loss function
- Optimizer

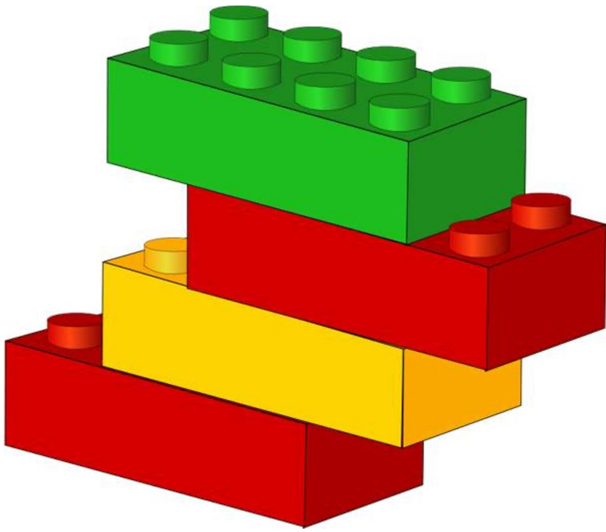


Tufts

Layers

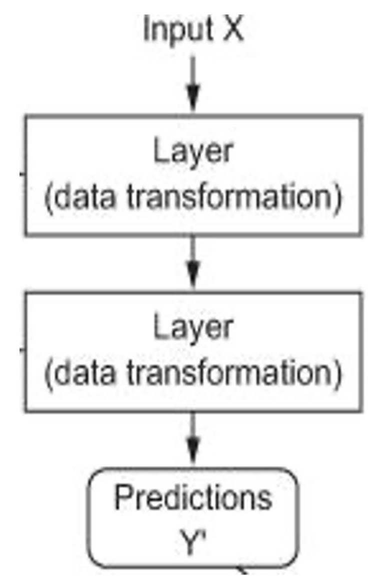
- Data processing modules
- Many different kinds exist
  - densely connected
  - convolutional
  - recurrent
  - pooling, flattening, merging, normalization, etc.
- Input: one or more tensors
  output: one or more tensors
- Usually have a state, encoded as **weights**
  - learned, initially random
- When combined, form a **network** or a **model**

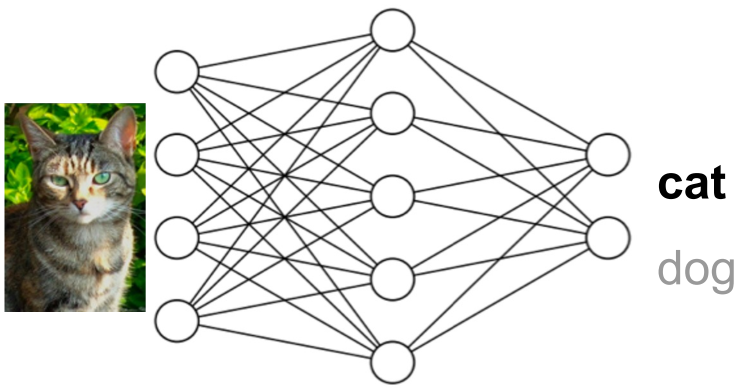




Input data and targets

- The network maps the input data X to predictions Y′
- During training, the predictions Y′ are compared to true targets Y using the loss function



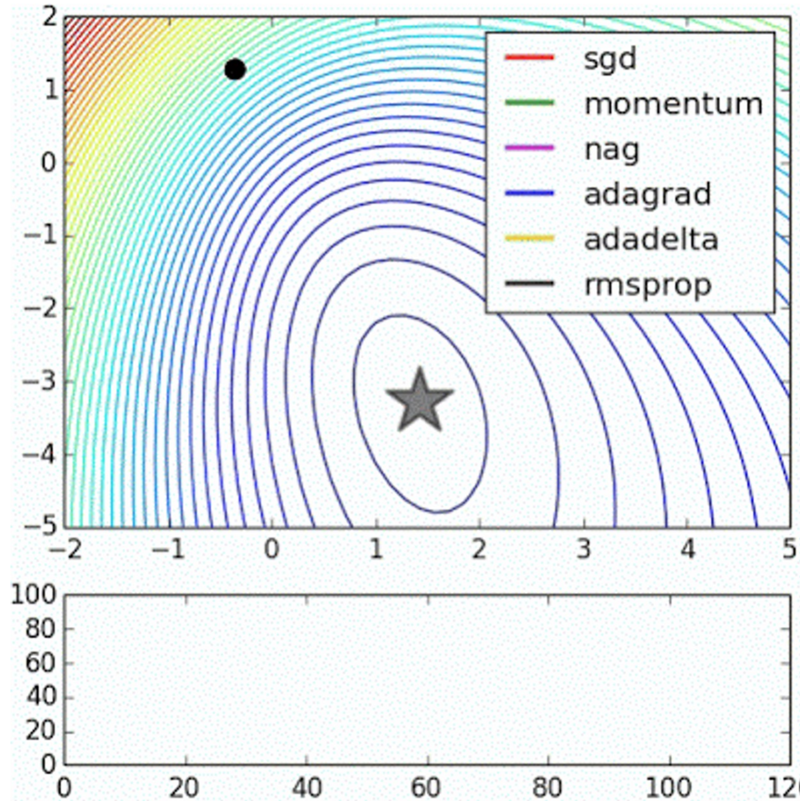


Loss function

- The quantity to be minimized (optimized) during training
  - the only thing **the network** cares about
  - there might also be other metrics **you** care about
- Common tasks have "standard" loss functions:
  - *mean squared error* for regression
  - *binary cross-entropy* for two-class classification
  - *categorical cross-entropy* for multi-class classification
  - etc.
- https://lossfunctions.tumblr.com/

Optimizer

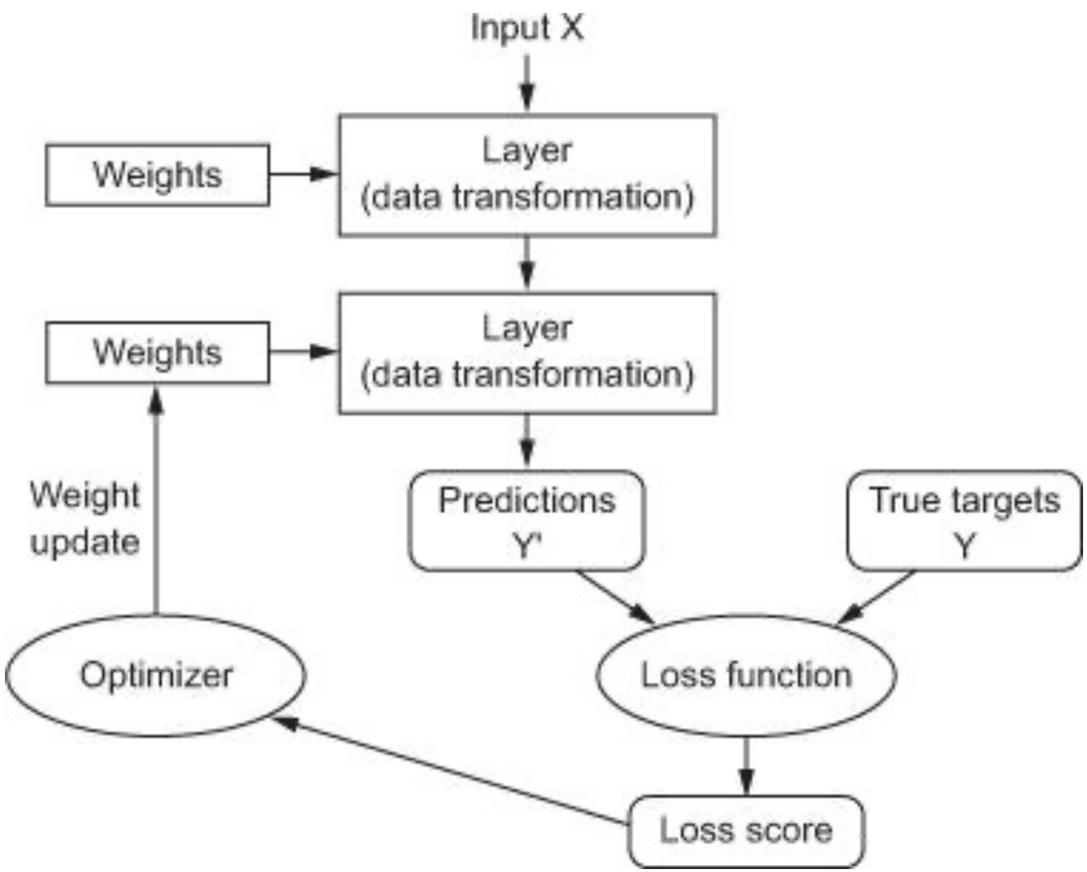
- How to update the weights based on the loss function
- *Learning rate (+scheduling)*
- Stochastic gradient descent, momentum, and their variants
    - RMSProp is usually a good first choice
    - more info: http://ruder.io/optimizing-gradient-descent/



Animation from: https://imgur.com/s25RsOr

Tufts

Anatomy of a deep neural network

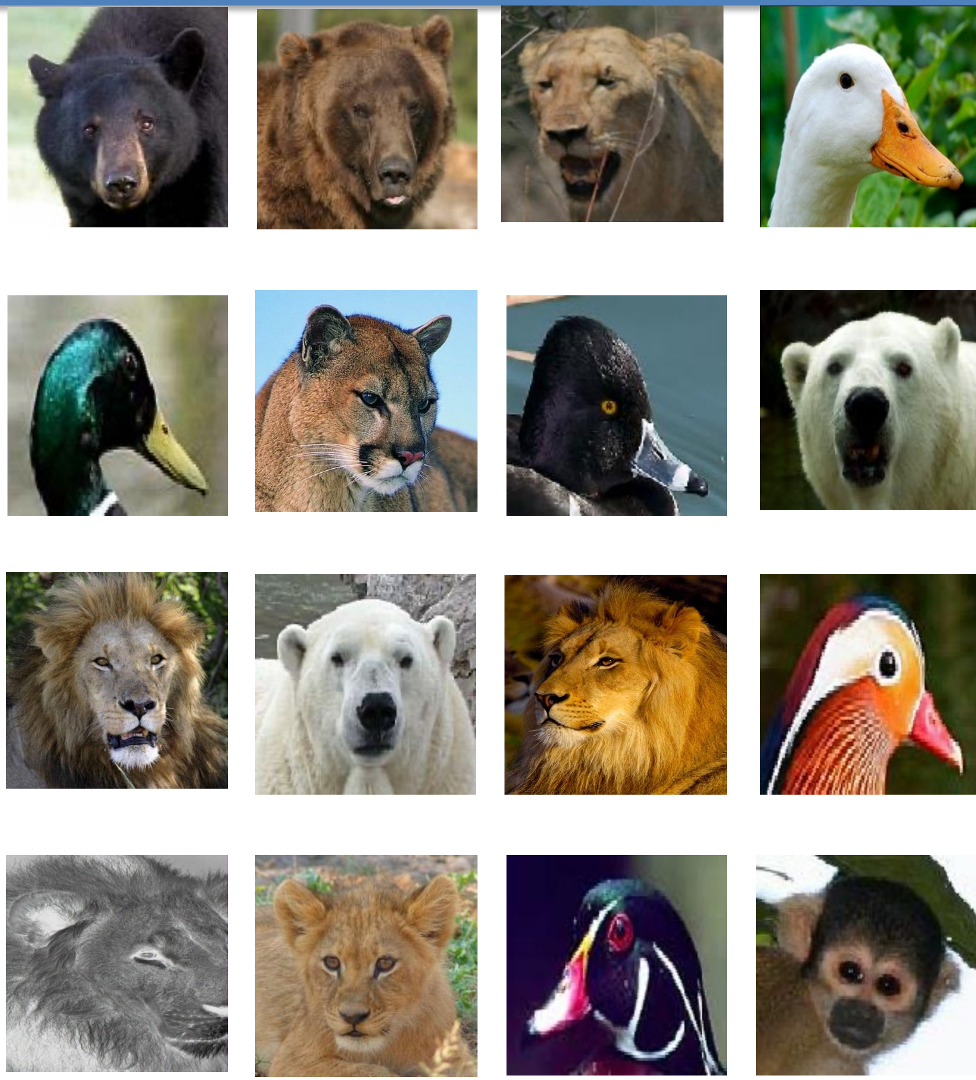


Tufts

Unsupervised Learning

Unsupervised Learning

- Labels may be too expensive to generate or may be completely unknown
- There are lots of training data but with no class labels assigned to it

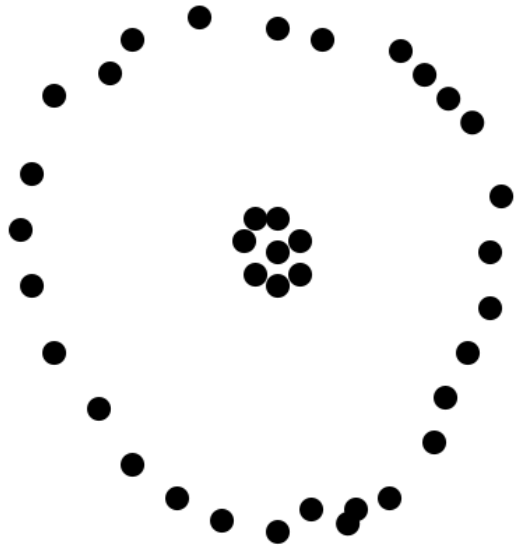


???

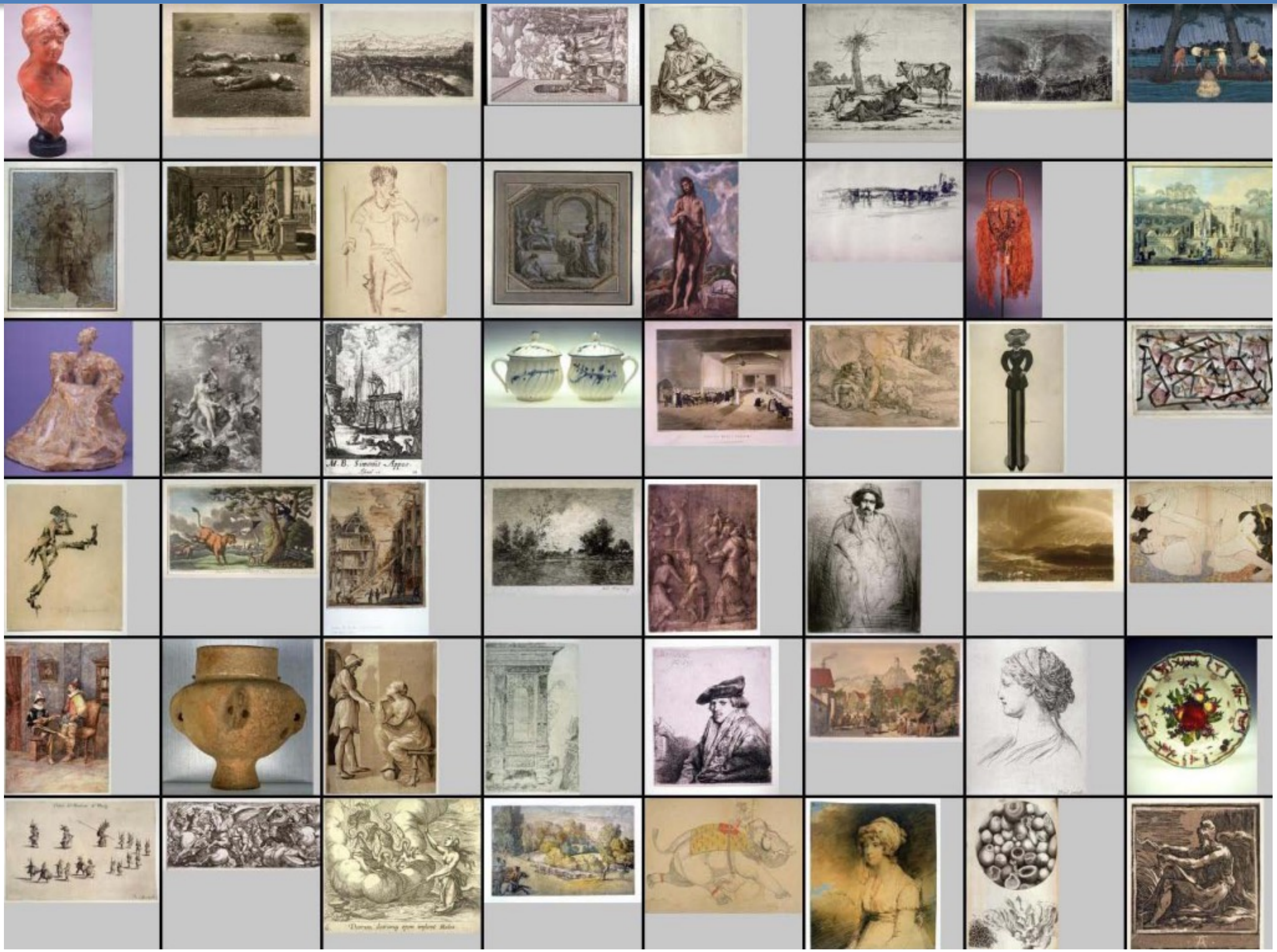
Source: LHI Animal Faces Dataset

Unsupervised Learning

- For example clustering
- Clustering –
  - grouping similar objects
- Similar in which way?

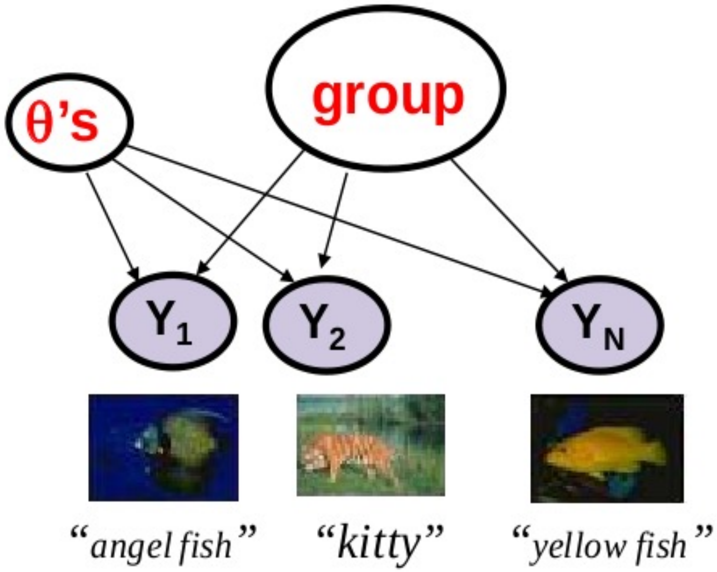


Clustering



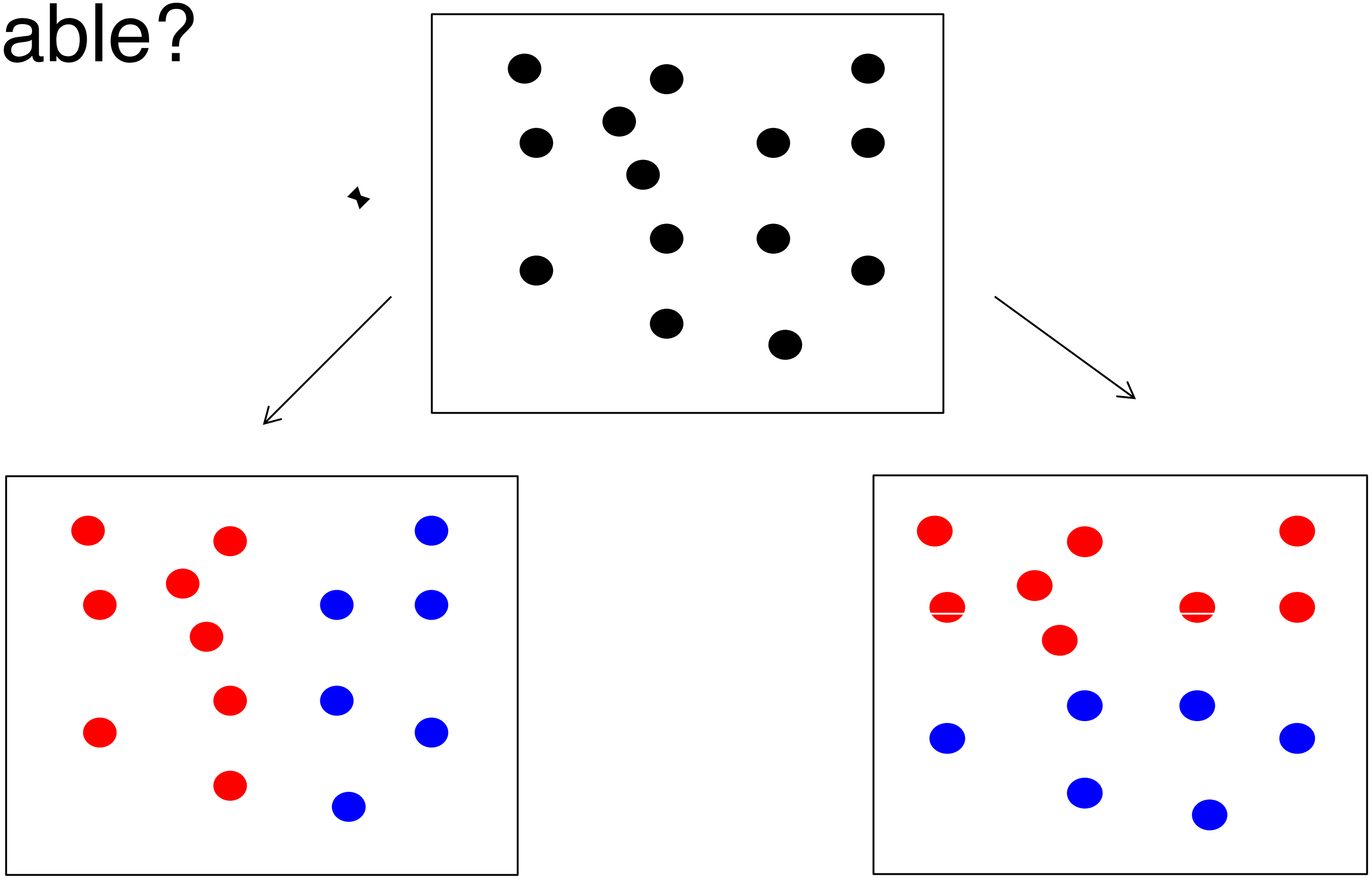
Tufts

Clustering



Clustering Problems

- How to tell which type of clustering is desirable?



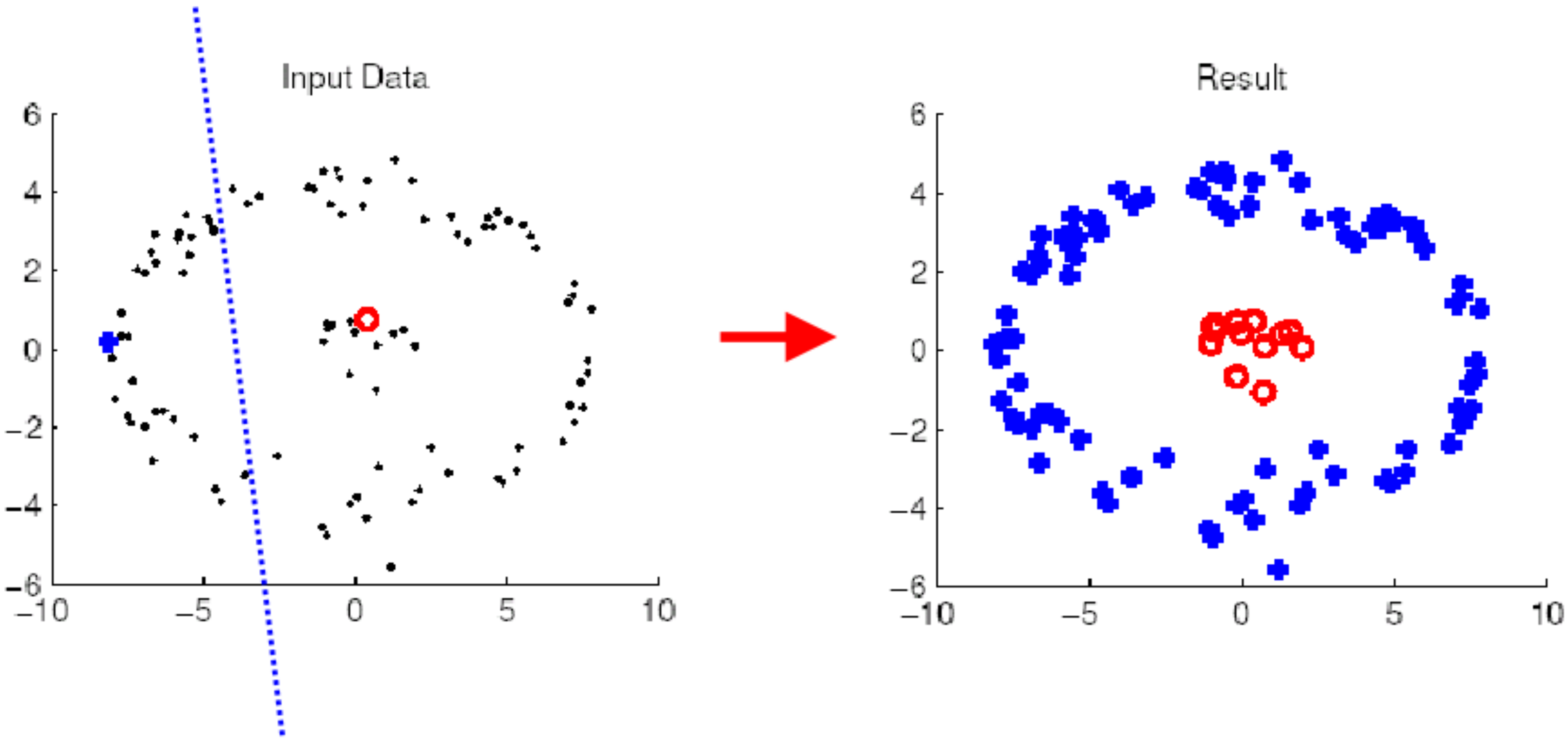
Semi-Supervised Learning

Semi-Supervised Learning

- Supervised learning + Additional unlabeled data
- Unsupervised learning + Additional labeled data
- **Learning Algorithm:**
  - Start from the labeled data to build an initial classifier
  - Use the unlabeled data to enhance the model
- **Some Techniques:**
  - Co-Training: two or more learners can be trained using an independent set of different features
  - Or to model the joint probability distribution of the features and labels.

Example

- ideally...



Tufts

Semi-Supervised Learning

Recommended Books and Software

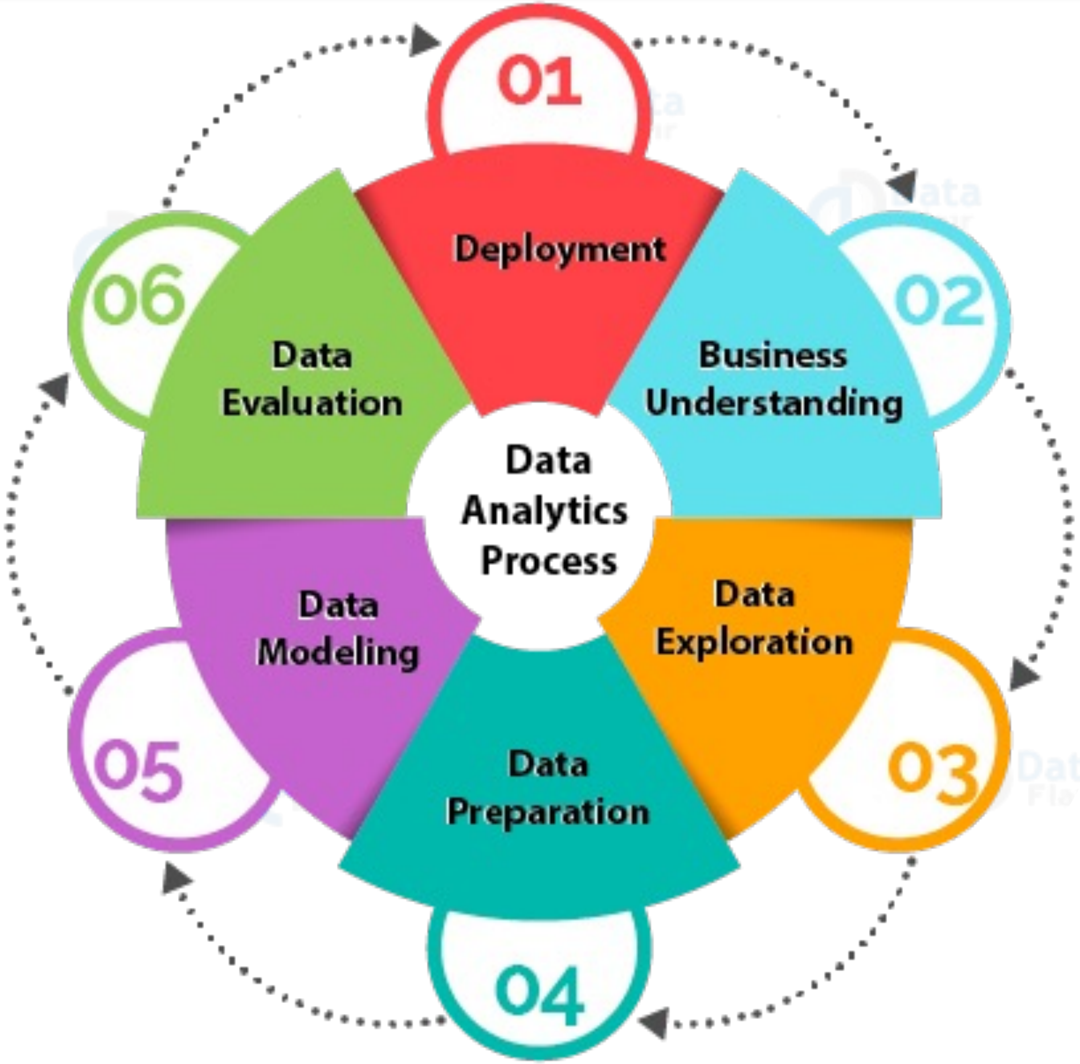
- Python for Data Analytics, W. McKinley, O'Reilly Media, https://www.dropbox.com/s/3il8999q7dssohe/Python4DataAnalysis.pdf?dl=0
- Mastering Python for Data Science, S. Madhavan, Packt Publishing Limited, https://www.dropbox.com/s/s58e0vy6y7qcb0k/Mastering%20Python%20for%20Data%20Science.pdf?dl=0
- Pep8 Coding Style Guidelines, https://www.python.org/dev/peps/pep-0008/
- Pycharm, https://www.jetbrains.com/student/
- An Introduction to Statistical Learning, Gareth J., et al, https://www.dropbox.com/s/6a7z2zu4xvq6y2h/An%20Introduction%20to%20Statistical%20Learning.pdf?dl=0

Python and Installation

- ❑ Open source
- ❑ High-level scripting language
- ❑ A bunch of toolboxes and libraries

- ❑ Python 3.7
- ❑ Recommendation: Anaconda

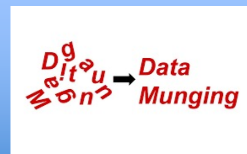
Machine Learning: Workflow

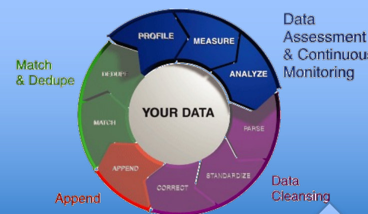


Tufts

Topics To Cover

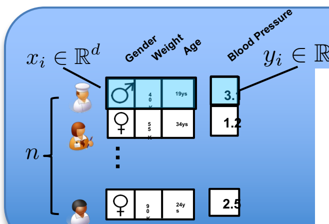
Data Handling

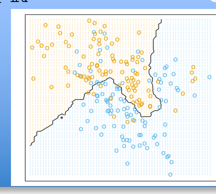




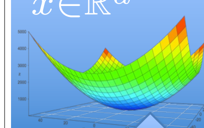
Data Munging
Data Preprocess
Data Types
...

Statistics & Machine Learning



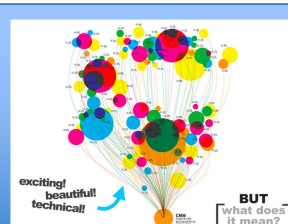


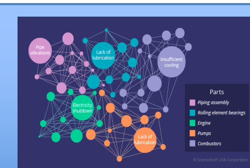
$$\arg\min_{x \in \mathbb{R}^d} f(x)$$



Regression, Classification
Regularization, Cross Validation
KNN, Optimization

Data Analysis and Presentation



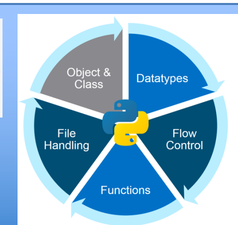



Story Telling
Flow Charts
Presentation/Discourse
Effective Techniques
...

Computational Methods

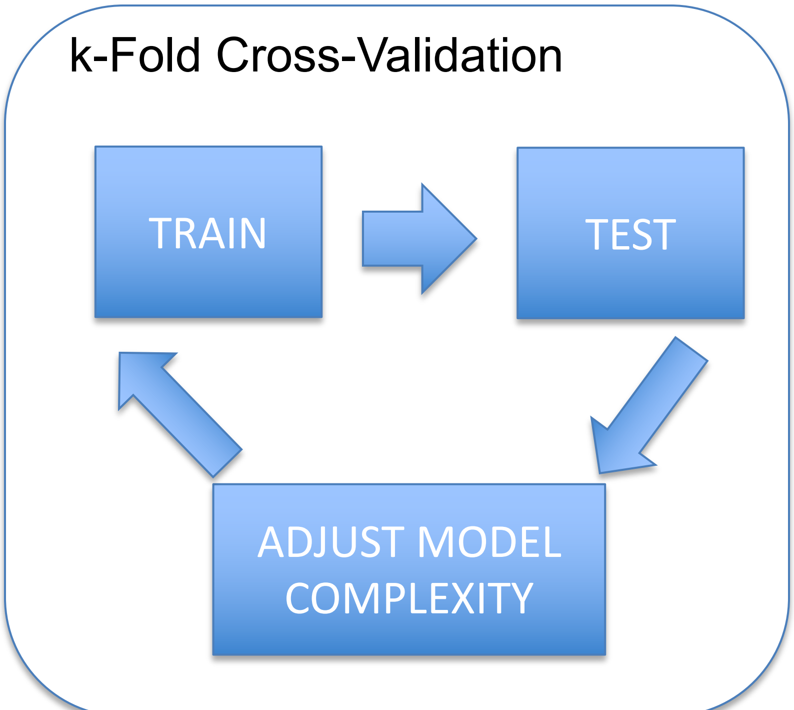
Pandas + scikit learn NumPy

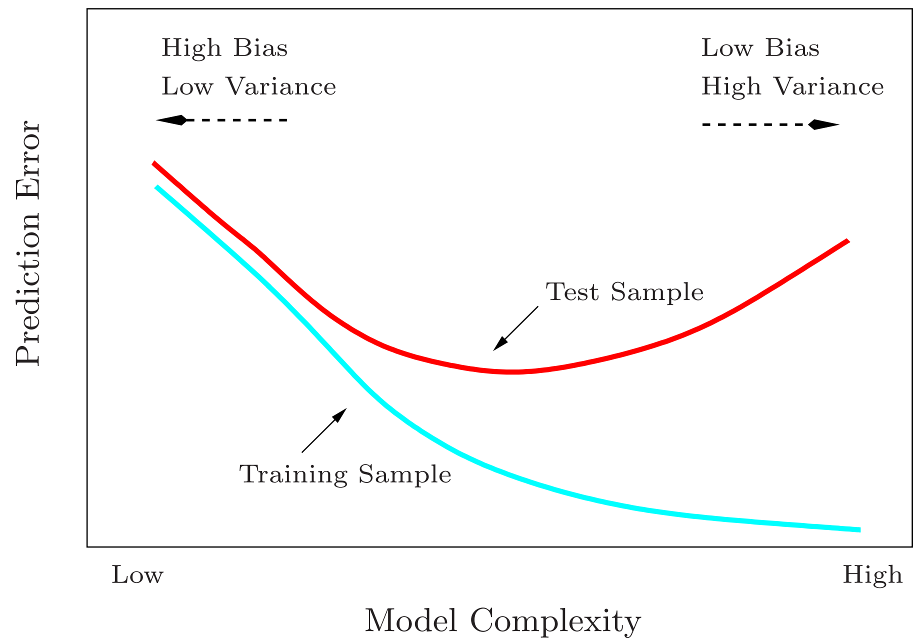
python matplotlib



Data Types
Function/Flow Control
Related Packages
...

- The Machine Learning Pipeline:



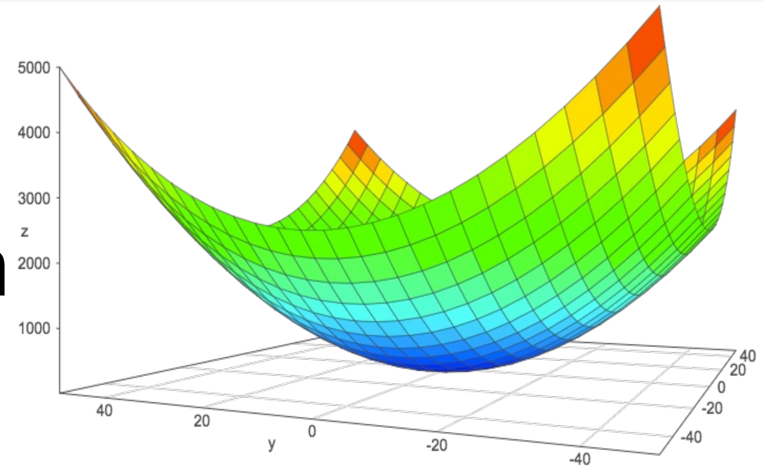


- Algorithms ⟷ Assumptions

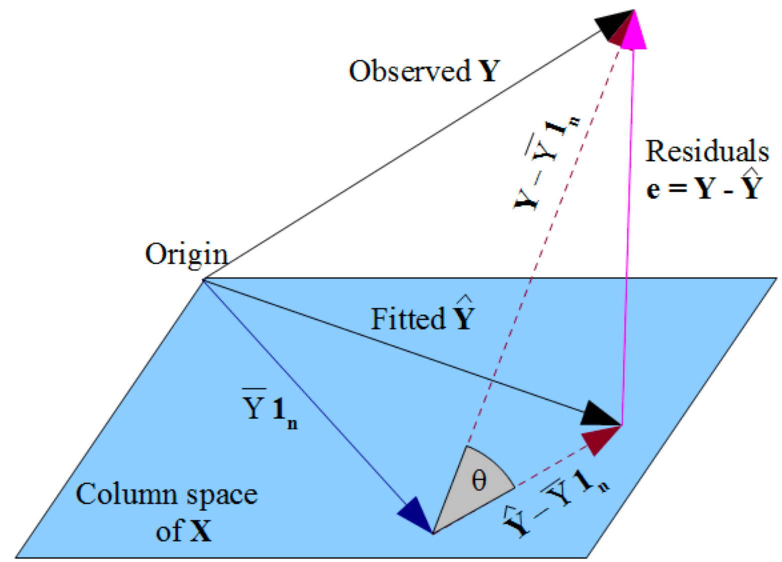
Tufts

Learning Methods

- ❑ Gradient Descent, Distance Metrics, Newton Method, and Bayesian



- ❑ Sparsity and Data Representations



- ❑ Regression and Classification

Tufts

Topic Breakdown

- **Topics:**
  - Python + related packages, tools, and Jupyter
  - Linear Regression
  - Data munching and crunching
  - KNN
  - Feature Selection
  - Lasso & Ridge Penalties
  - Curse of Dimensionality
  - Matrix Factorization
  - Principle Component Analysis (PCA)
  - Accuracy and AUC
  - Data Visualization
  - Representing a Corpus of Text and comparing Exemplars
  - Content-based recommendation systems
  - K-Means
  - Deep Learning
  - More…

By the end of the class, you will…

- ❑ …be confident in **Python programming**.
- ❑ …know how **represent** and **interpret** different data types.
- ❑ …know how to use **linear algebra** and **probability** to solve data analysis problems.
- ❑ …be able to clean, analyze and visualize large datasets:
  - ❑ **Preprocess** data.
  - ❑ **Train** a model.
  - ❑ **Validate** your model's performance.
  - ❑ **Analyze** solution analytically and visually.
- ❑ …understand concepts of **regression** and **classification** as a means to determine relationships and trends in data.
- ❑ …understand fundamentals of **data visualization**.
- ❑ …be able to use **data analysis tools** in real-world problems.

This Course is Not…

- *…a Mathematical class:*
  - There will be a probability theory and linear algebra, but we will focus on scripting Python and **simple methods**

- *…a programming-only class:*
  - There will be math + stats: we should understand what we are doing, and why

Logistics

Grading Scheme
- ❑ Homework: 35%
- ❑ Course projects: 30%
- ❑ Midterm: 25%
- ❑ Class Participation 10%

Homework
- ❑ ~4 homework assignments
- ❑ ~3 project-based assignments (2 required + 1 optional/ extra credit)
- ❑ Submit **code** and **typewritten** report

- ❑ Class participation will be in the form of regular poll questions.

- ❑ Use Piazza to ask questions!!