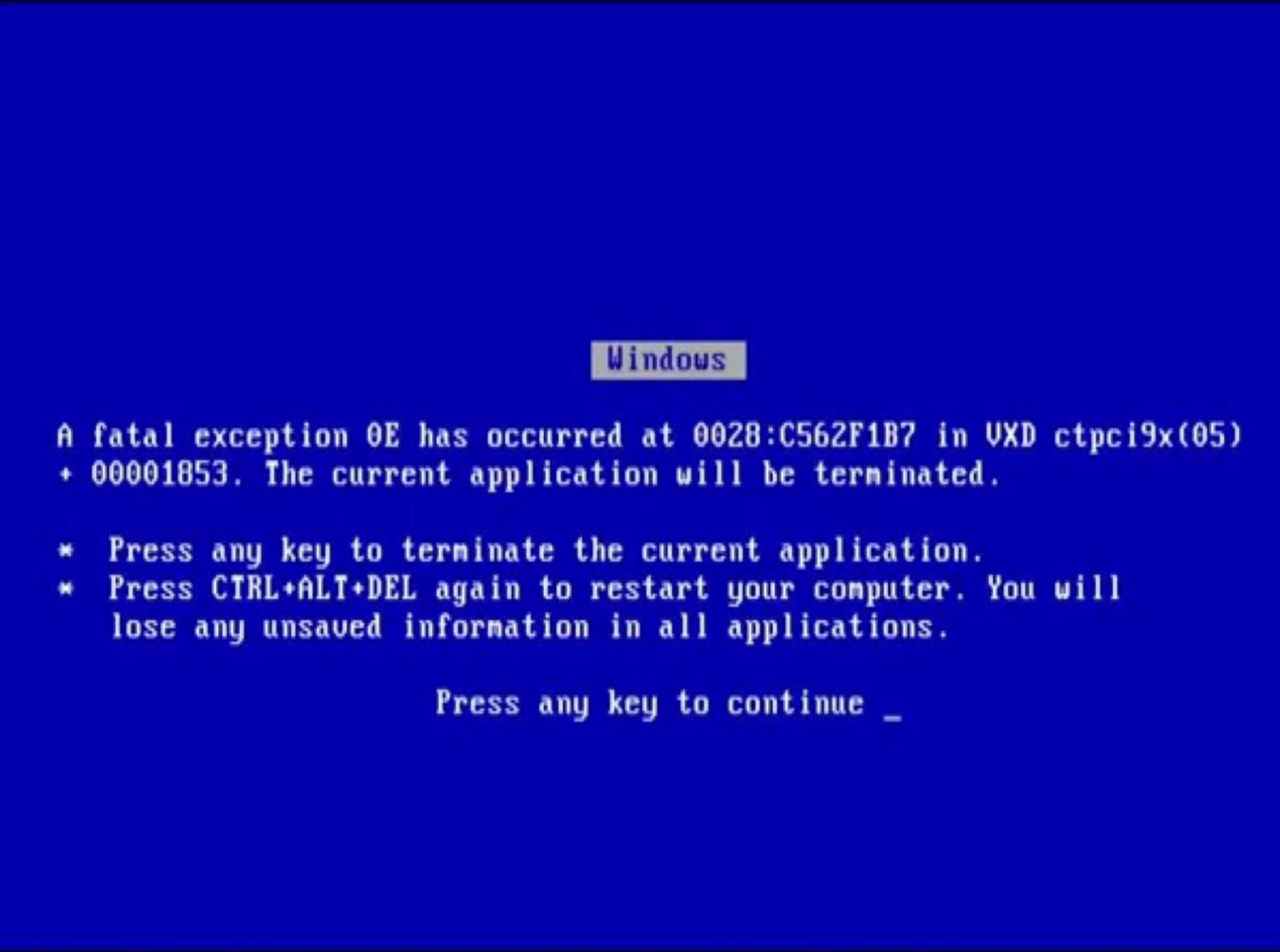
Machine Learning: Preliminaries & Overview

Winter 2018

Portland State University



```
                              Windows

A fatal exception 0E has occurred at 0028:C562F1B7 in VXD ctpci9x(05)
+ 00001853. The current application will be terminated.

*  Press any key to terminate the current application.
*  Press CTRL+ALT+DEL again to restart your computer. You will
   lose any unsaved information in all applications.

                     Press any key to continue _
```

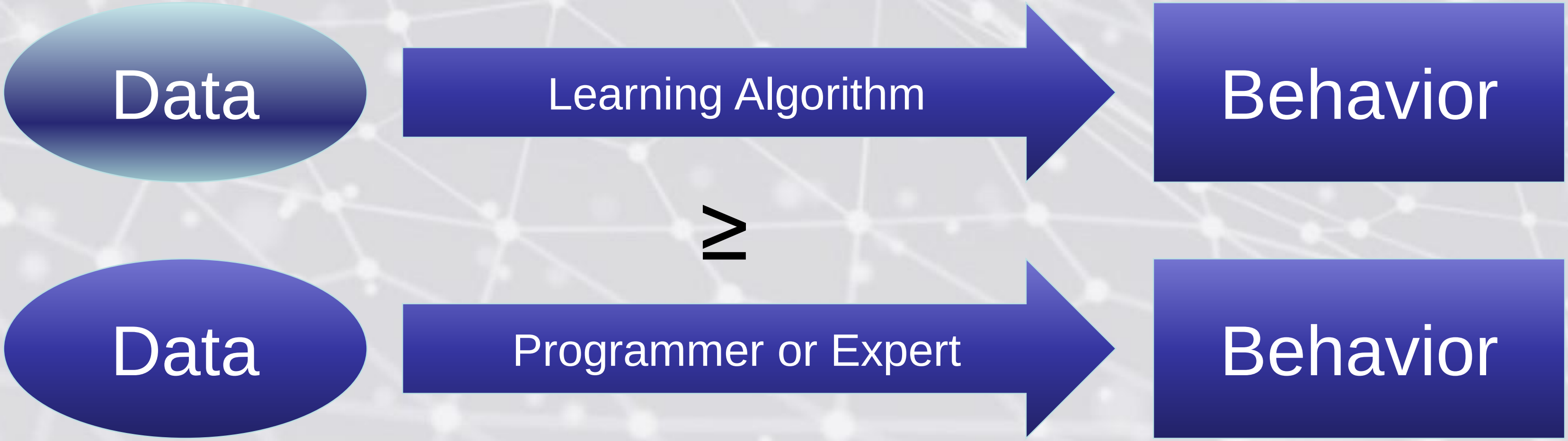
```
A fatal exc                                          ctpci9x(05)
+ 00001853.

*  Press an
*  Press CT                                          will
   lose any

                 Press any key to continue _
```

What is machine learning?

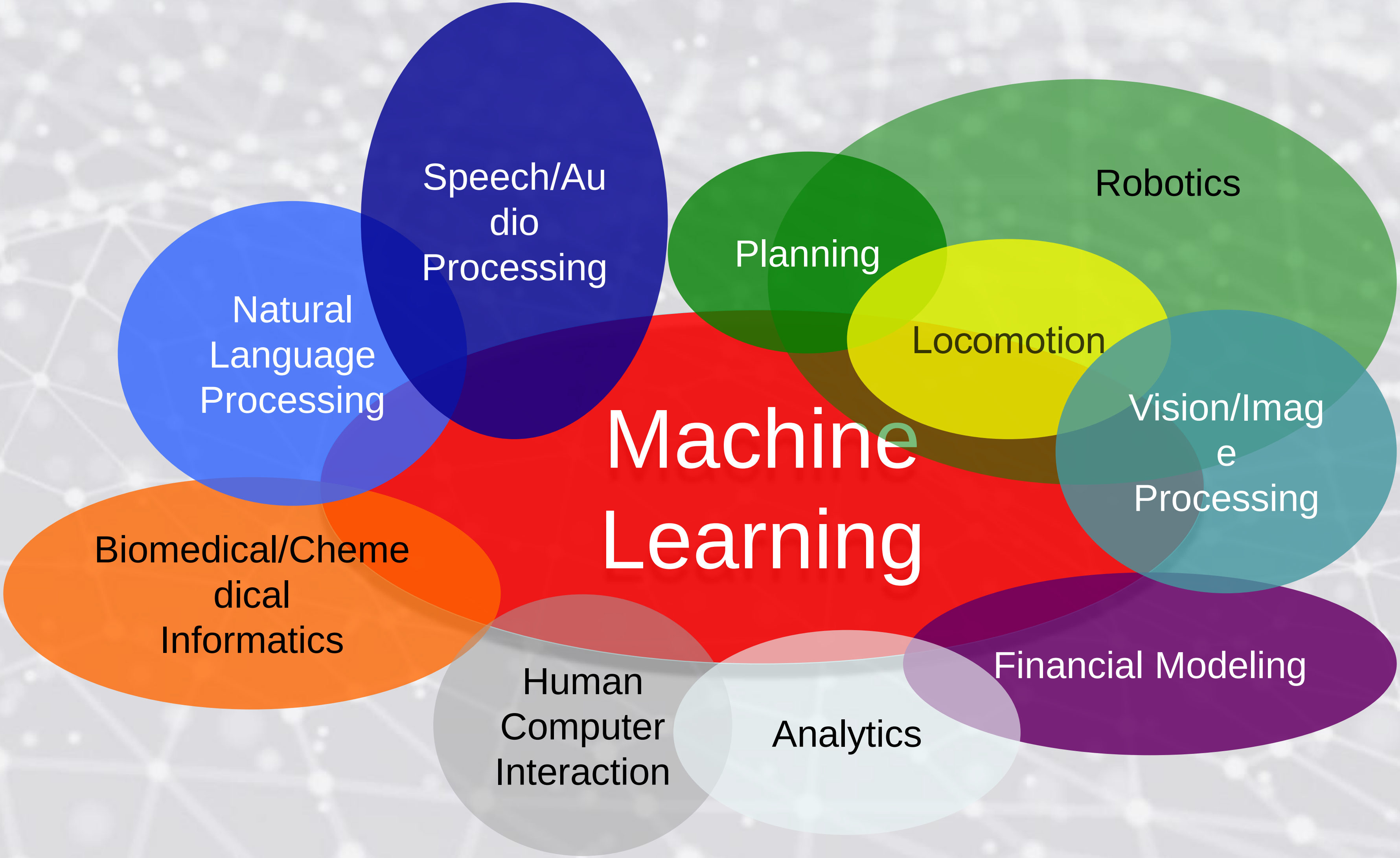
- Textbook definitions of “machine learning”:
  - Detecting patterns and regularities with a good and generalizable approximation (“model” or “hypothesis”)
  - Execution of a computer program to optimize the parameters of the model using training data or past experience.

Machine Learning

- Automatically identifying patterns in data
- Automatically making decisions based on data
- Hypothesis:



Machine Learning in Computer Science

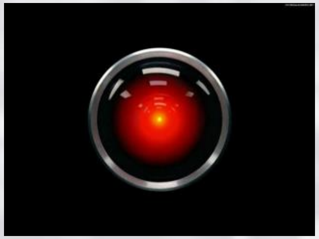


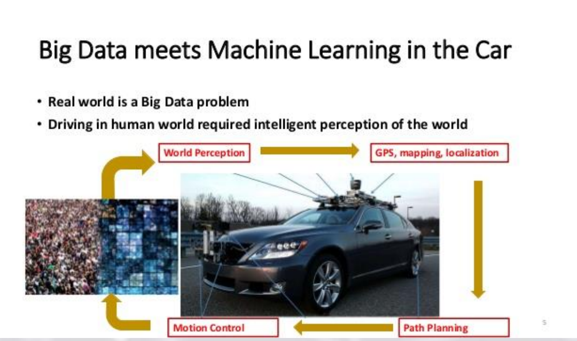
Major Tasks

- Regression
  - Predict a numerical value from “other information”; Output is a real value (e.g., ‘$35/share”)
- Classification
  - Predict a categorical value; Output is one of a number of classes (e.g., ‘A’)
- Clustering
  - Identify groups of similar entities
- Optimization

A Small Subset of Machine Learning Applications

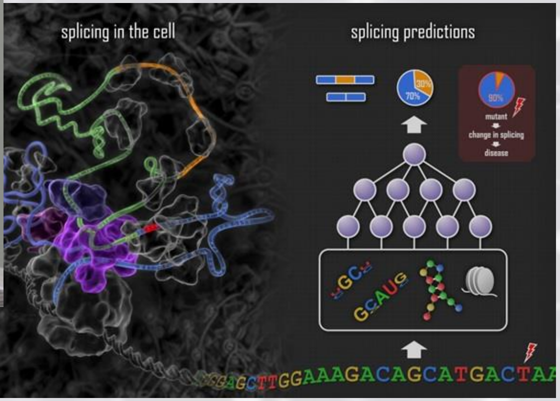
(*) Speech Recognition
(*) NLP (natural language processing); machine translation.
(*) Computer Vision
(*) Medical Diagnosis
(*) Autonomous Driving
(*) Statistical Arbitrage
(*) Signal Processing
(*) Recommender Systems
(*) ~~World Domination~~
(*) Fraud Detection
(*) Social Media
(*) Data Security
(*) Search
(*) A.I. & Robotics
(*) Genomics
(*) Computational Creativity
(*) Hi Scores



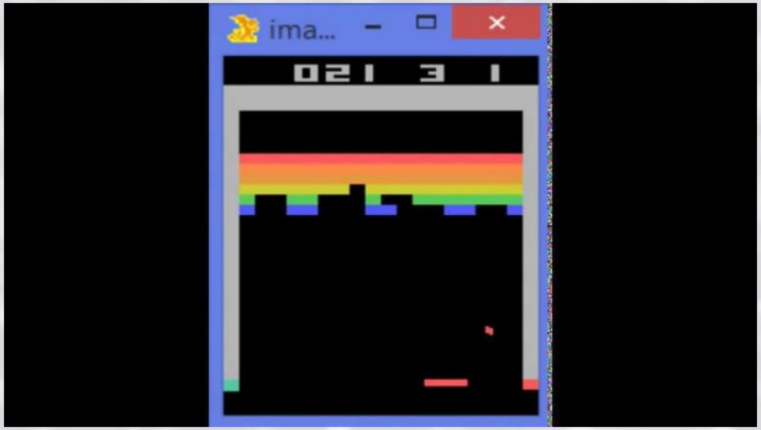

Big Data meets Machine Learning in the Car

- Real world is a Big Data problem
- Driving in human world required intelligent perception of the world

World Perception
GPS, mapping, localization
Path Planning
Motion Control



A Small Subset of Machine Learning Applications



- https://www.youtube.com/watch?v=V1eYniJ0Rnk

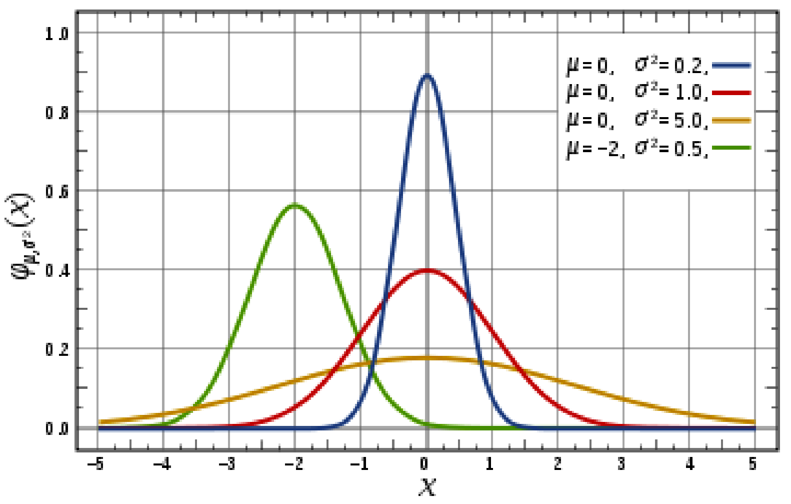
- https://www.youtube.com/watch?v=SCE-QeDfXtA

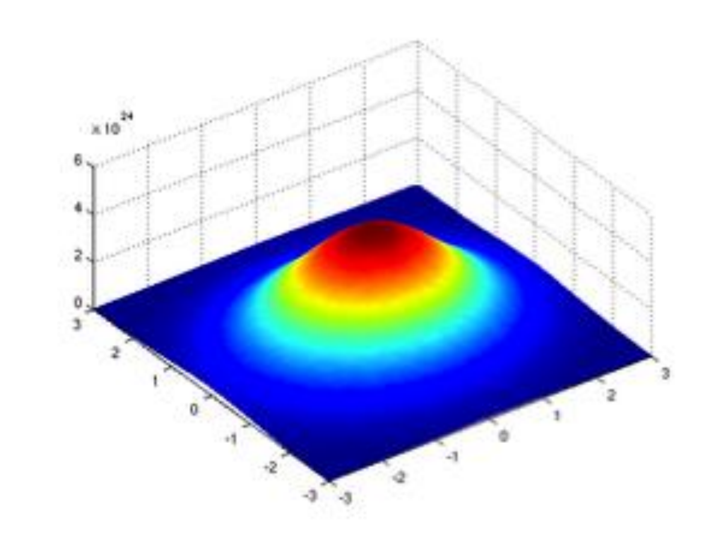
Mathematical Necessities

- Probability
- Statistics
- Calculus
  - Vector Calculus
- Linear Algebra
- Algorithms

Why do we need so much math?

- Probability Density Functions allow the evaluation of how likely a data point is under a model.
  - Want to identify good PDFs. (calculus)
  - Want to evaluate against a known PDF. (algebra)

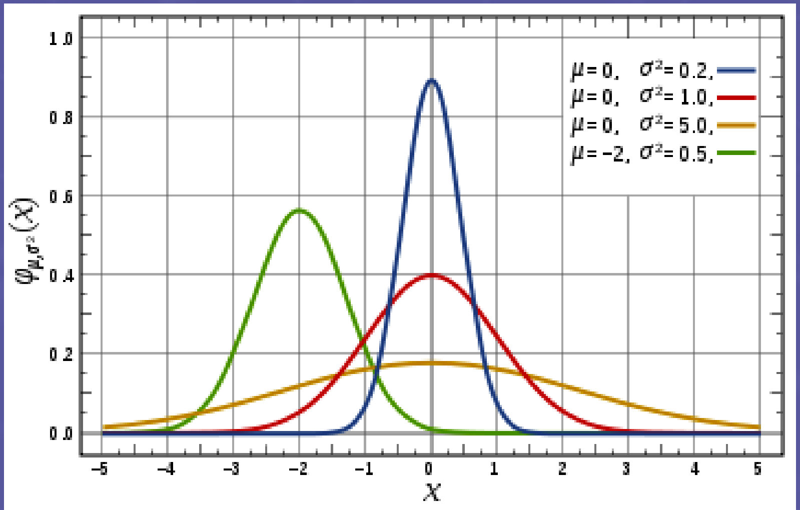


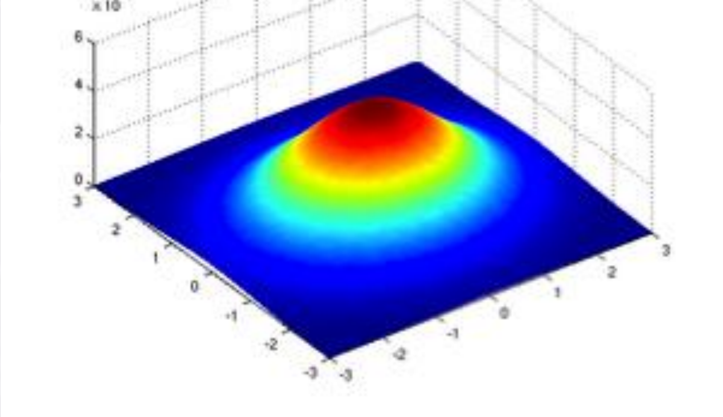


Gaussian Distributions

- We use Gaussian Distributions all over the place.

$$N(x;\mu,\sigma^2)=\frac{1}{\sqrt{2\pi\sigma^2}}exp\left\{-\frac{1}{2\sigma^2}(x-\mu)^2\right\}$$

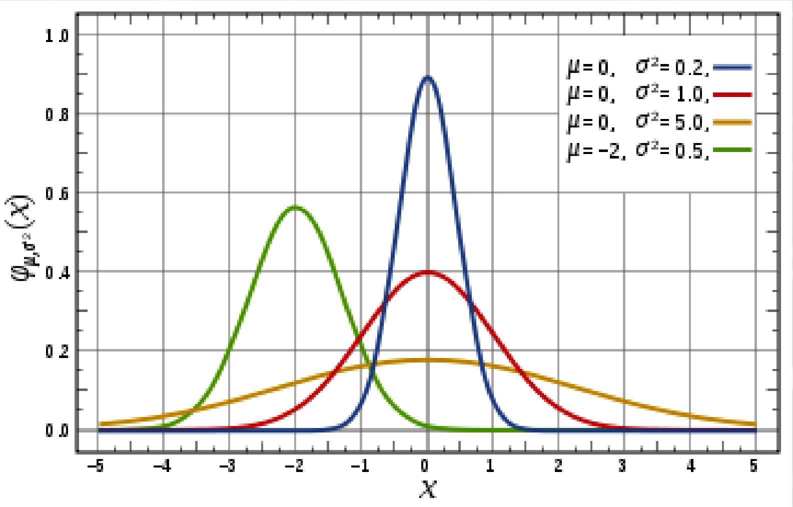


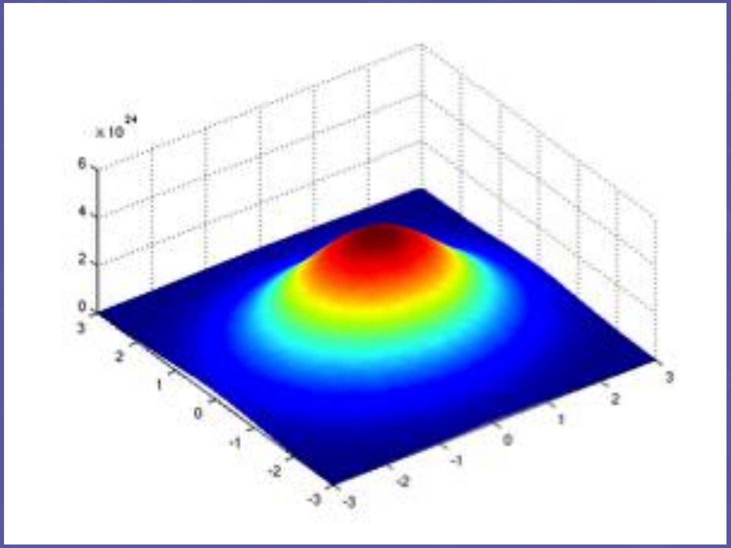


Gaussian Distributions

- We use Gaussian Distributions all over the place.

$$N(x;\mu,\Sigma)=\frac{1}{(2\pi)^{D/2}|\Sigma|^{1/2}}exp\left\{-\frac{1}{2}(x-\mu)^T\Sigma^{-1}(x-\mu)\right\}$$





Types of Machine Learning Methods

- **Supervised**
  - provide explicit training examples with correct answers
    - e.g. neural networks with back-propagation
- **Unsupervised**
  - no feedback information is provided
    - e.g., unsupervised clustering based on similarity
- “**Semi-supervised**”
  - some feedback information is provided but it is not detailed
    - e.g., only a fraction of examples are labeled
    - e.g., reinforcement learning: reinforcement single is single-valued assessment of current state

Data Data Data

- “There’s no data like more data”
- All machine learning techniques rely on the availability of data to learn from.
- There is an ever increasing amount of data being generated, but it’s not always easy to process.
- Is all data equal?
- (Good) Data (can) trump a choice of model!

Key Ingredients for Any Machine Learning Method

- **Features** (or “**attributes**”)
- Underlying **Representation** for “hypothesis”, “model”, or “target function”
- **Hypothesis space**
- **Learning method**
- **Data**:
  - **Training data**
    - Used to train the model
  - **Validation (or Development) data**
    - Used to select model hyperparameters, to determine when to stop training, or to alter training method
  - **Test data**
    - Used to evaluate trained model
- **Evaluation method**

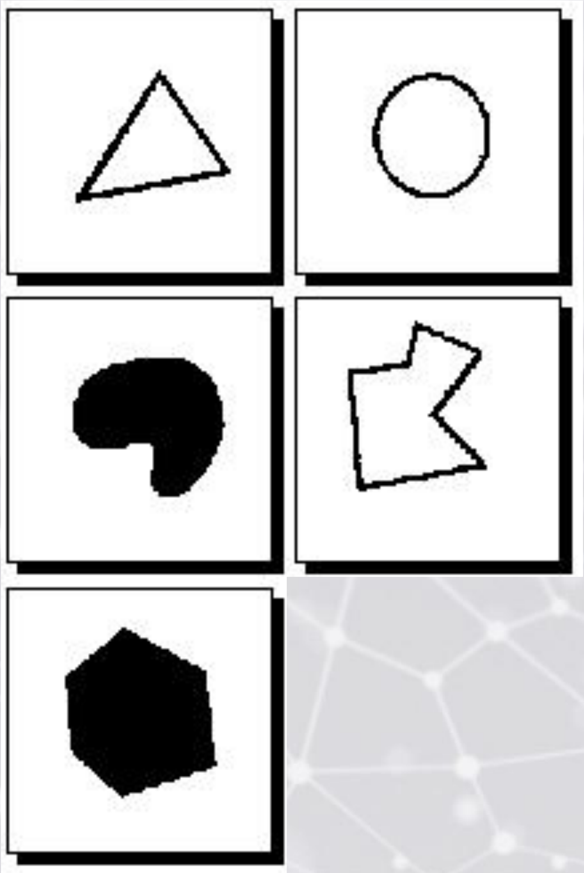
Assumption of all ML methods

Inductive learning hypothesis:

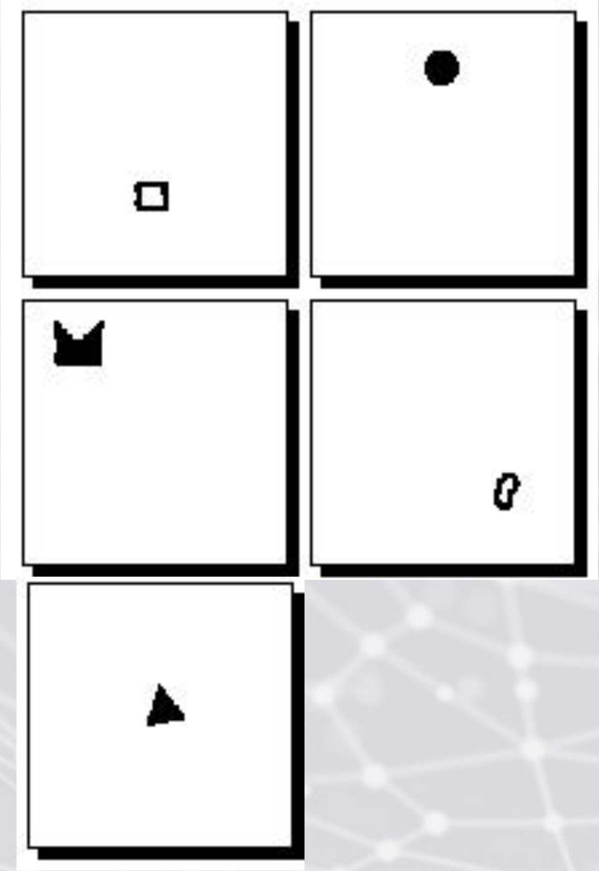
Any hypothesis that approximates target concept well over sufficiently large set of training examples will also approximate the concept well over other examples outside of the training set.

Q: What is the difference between "induction" and "deduction"?

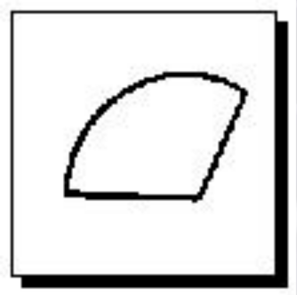
Training Examples: Class 1



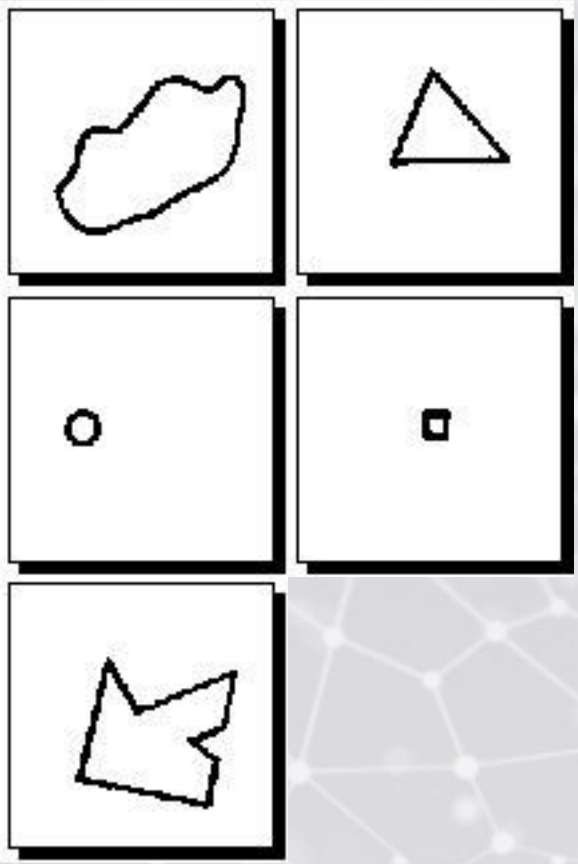
Training Examples: Class 2



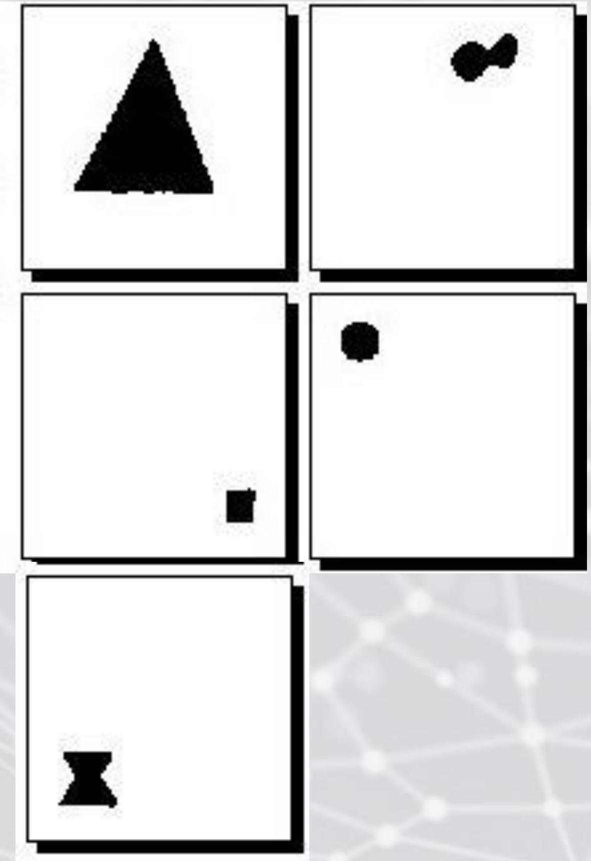
Test example: Class = ?



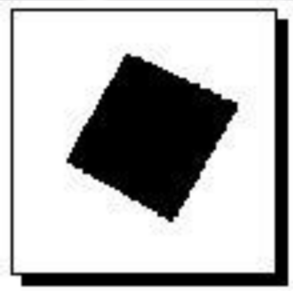
Training Examples: Class 1



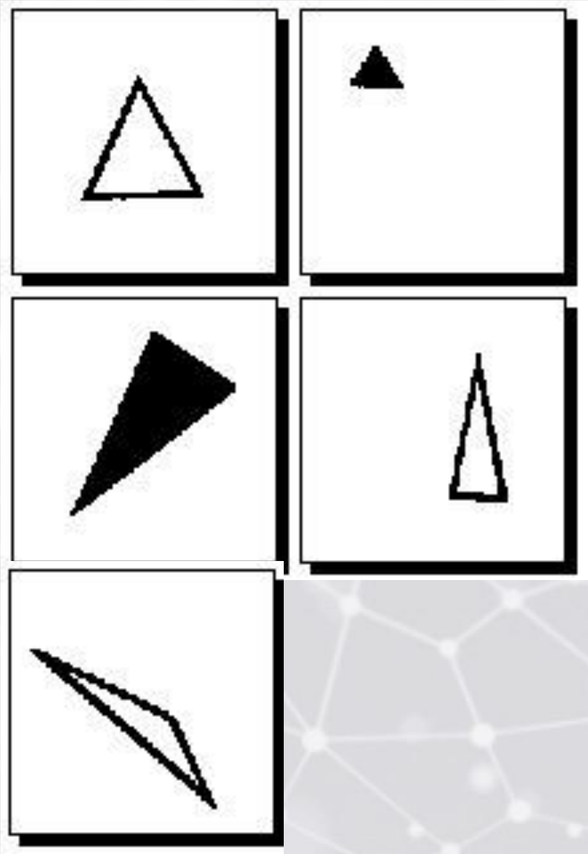
Training Examples: Class 2



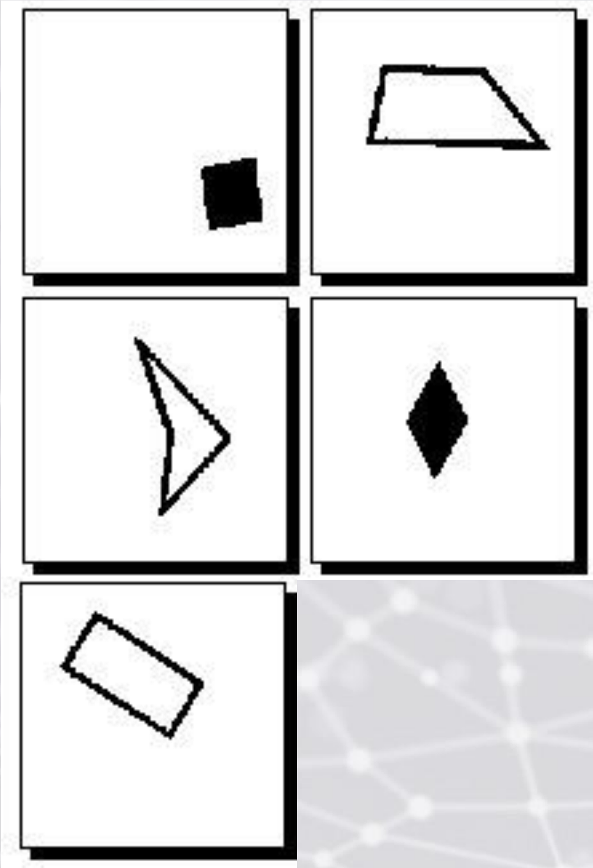
Test example: Class = ?



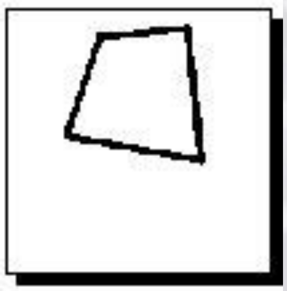
Training Examples: Class 1



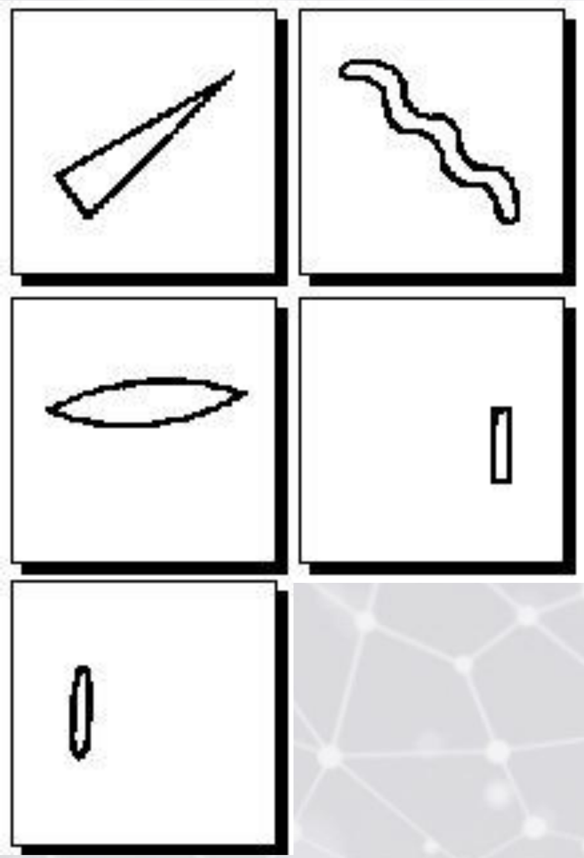
Training Examples: Class 2



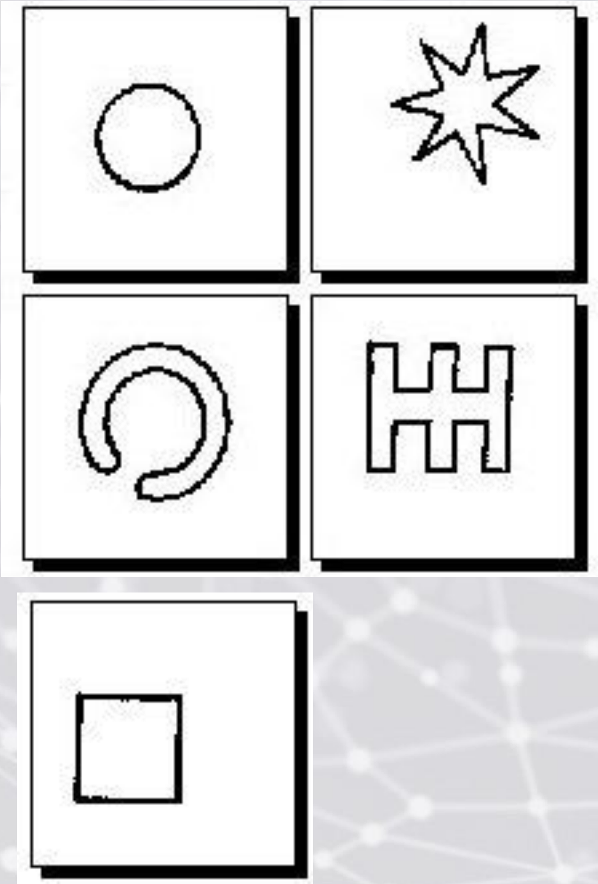
Test example: Class = ?



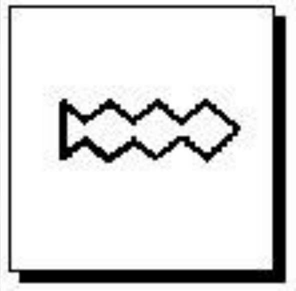
Training Examples: Class 1



Training Examples: Class 2

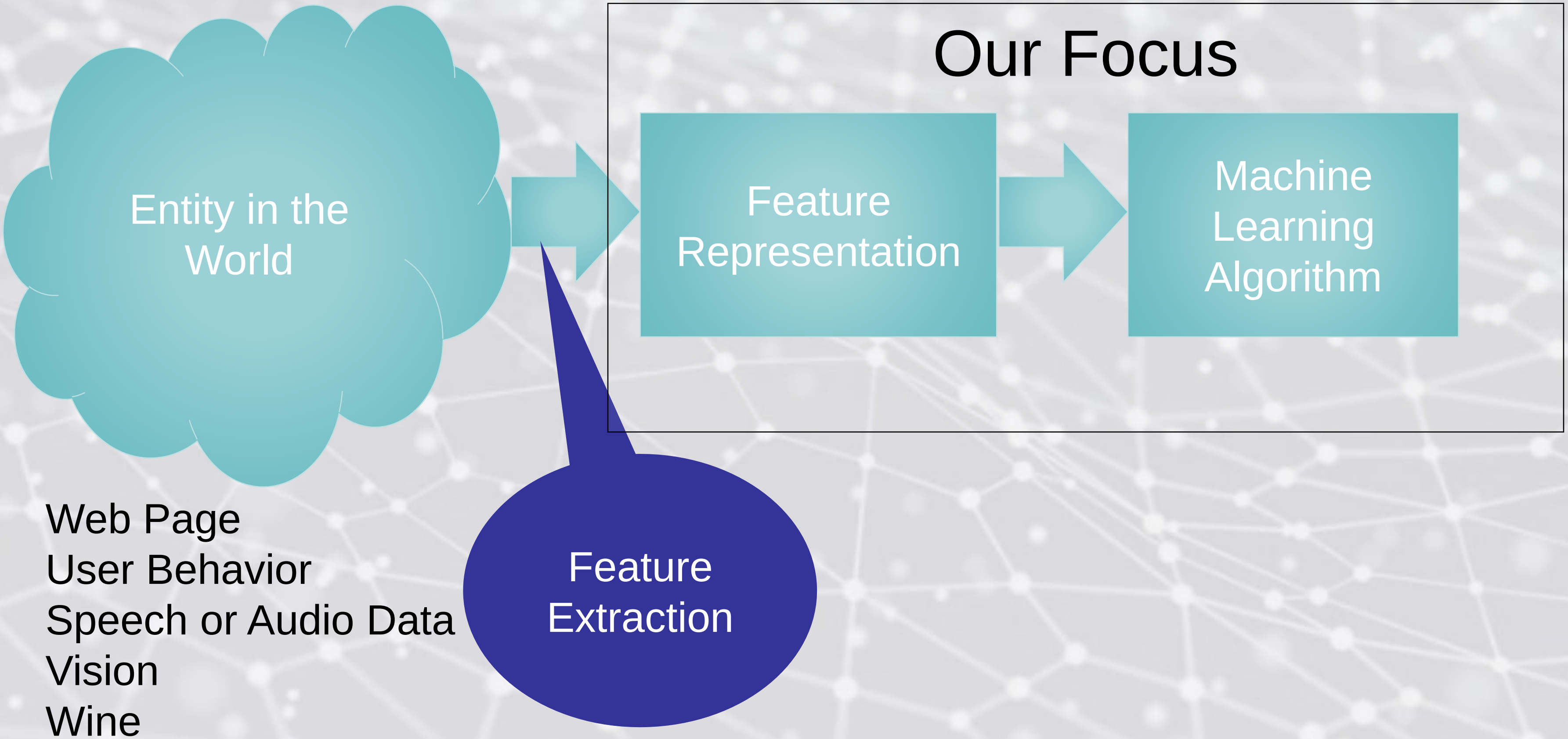


Test example: Class = ?



Feature Representations

- How do we view data?



Web Page
User Behavior
Speech or Audio Data
Vision
Wine
People
Etc.

Feature Representations

Height	Weight	Eye Color	Gender
66	170	Blue	Male
73	210	Brown	Male
72	165	Green	Male
70	180	Blue	Male
74	185	Brown	Male
68	155	Green	Male
65	150	Blue	Female
64	120	Brown	Female
63	125	Green	Female
67	140	Blue	Female
68	165	Brown	Female
66	130	Green	Female

Classification

- Identify which of *N* classes a data point, **x**, belongs to.
- **x** is a column vector of features.

$$\vec{x} = \begin{pmatrix} x_0 \\ x_1 \\ \dots \\ x_{n-1} \end{pmatrix} \quad \text{OR} \quad \vec{x} = \begin{pmatrix} f_0(x) \\ f_1(x) \\ \dots \\ f_{m-1}(x) \end{pmatrix}$$

Target Values

- In **supervised** approaches, in addition to a data point, **x,** we will also have access to a target value, **t.**

Goal of Classification

Identify a function *y*, such that *y*(**x**) = **t**

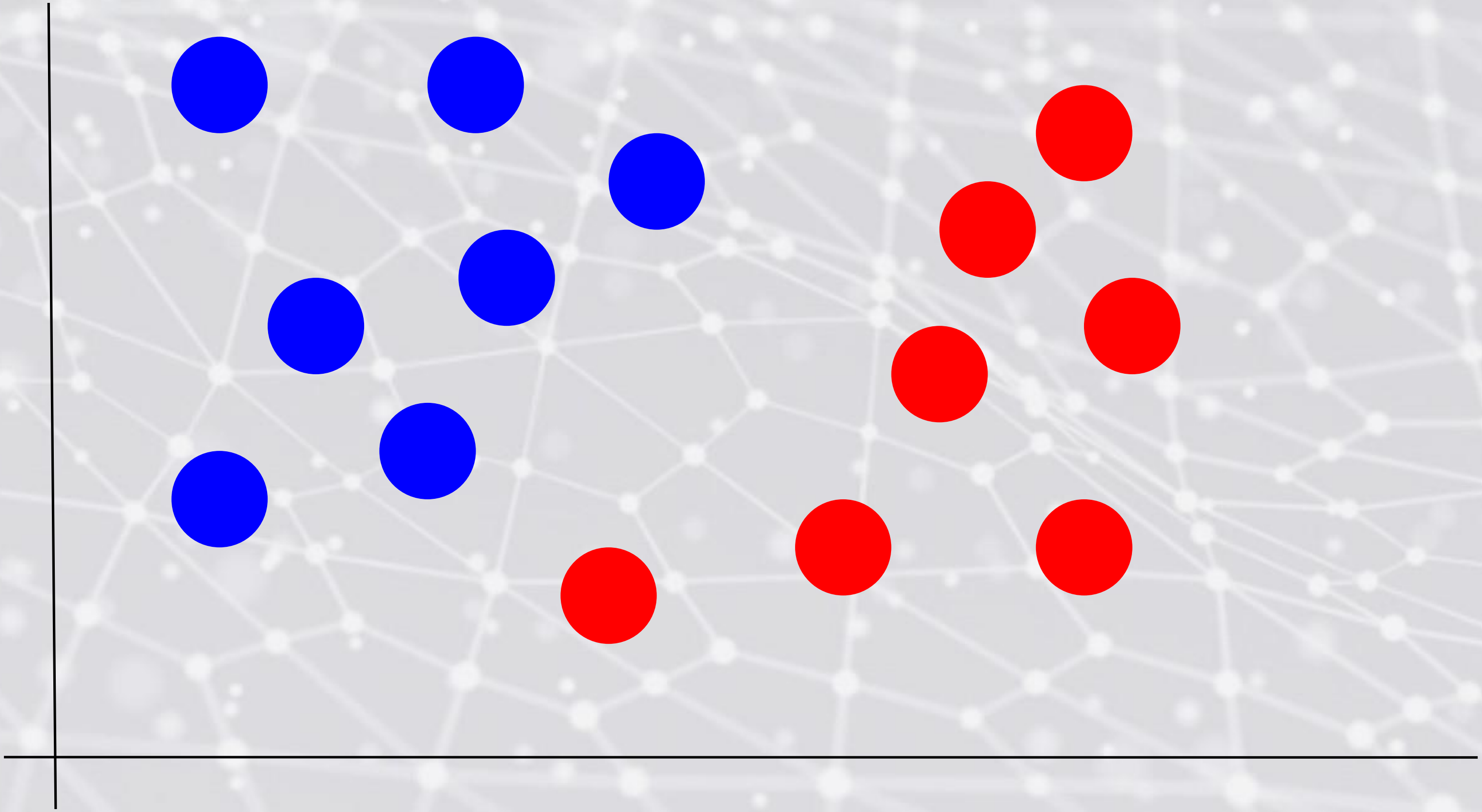
Feature Representations

Height	Weight	Eye Color	Gender
66	170	Blue	Male
73	210	Brown	Male
72	165	Green	Male
70	[illegible]	Blue	Male
74	[illegible]	Brown	Male
68	[illegible]	Green	[illegible]
65	[illegible]	Blue	Female
64	[illegible]	Brown	Female
63	125	Green	Female
67	140	Blue	Female
68	165	Brown	Female
66	130	Green	Female

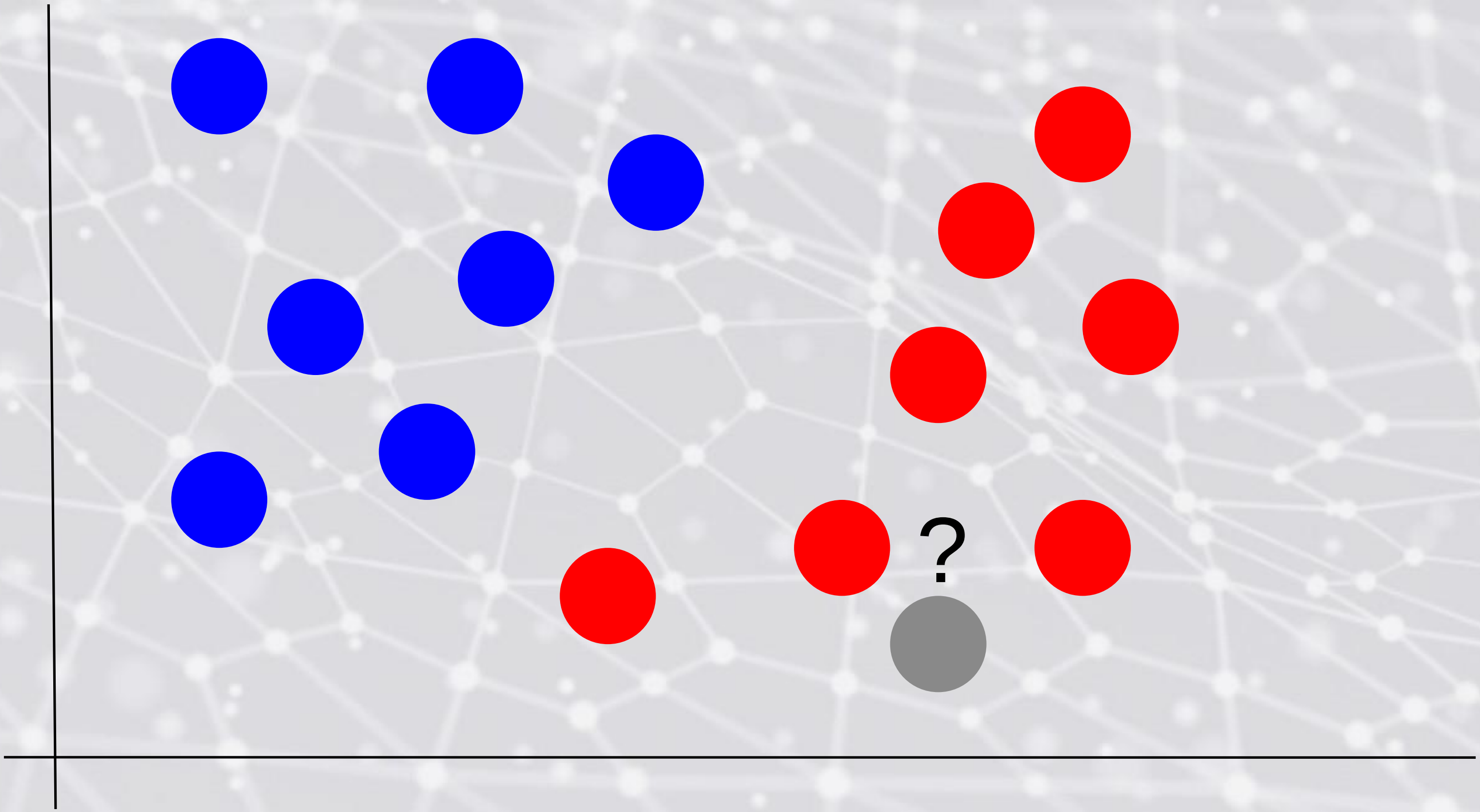
$$\vec{x_0} = \begin{pmatrix} 66 \\ 170 \\ Blue \end{pmatrix}$$

$$t_0 = Male$$

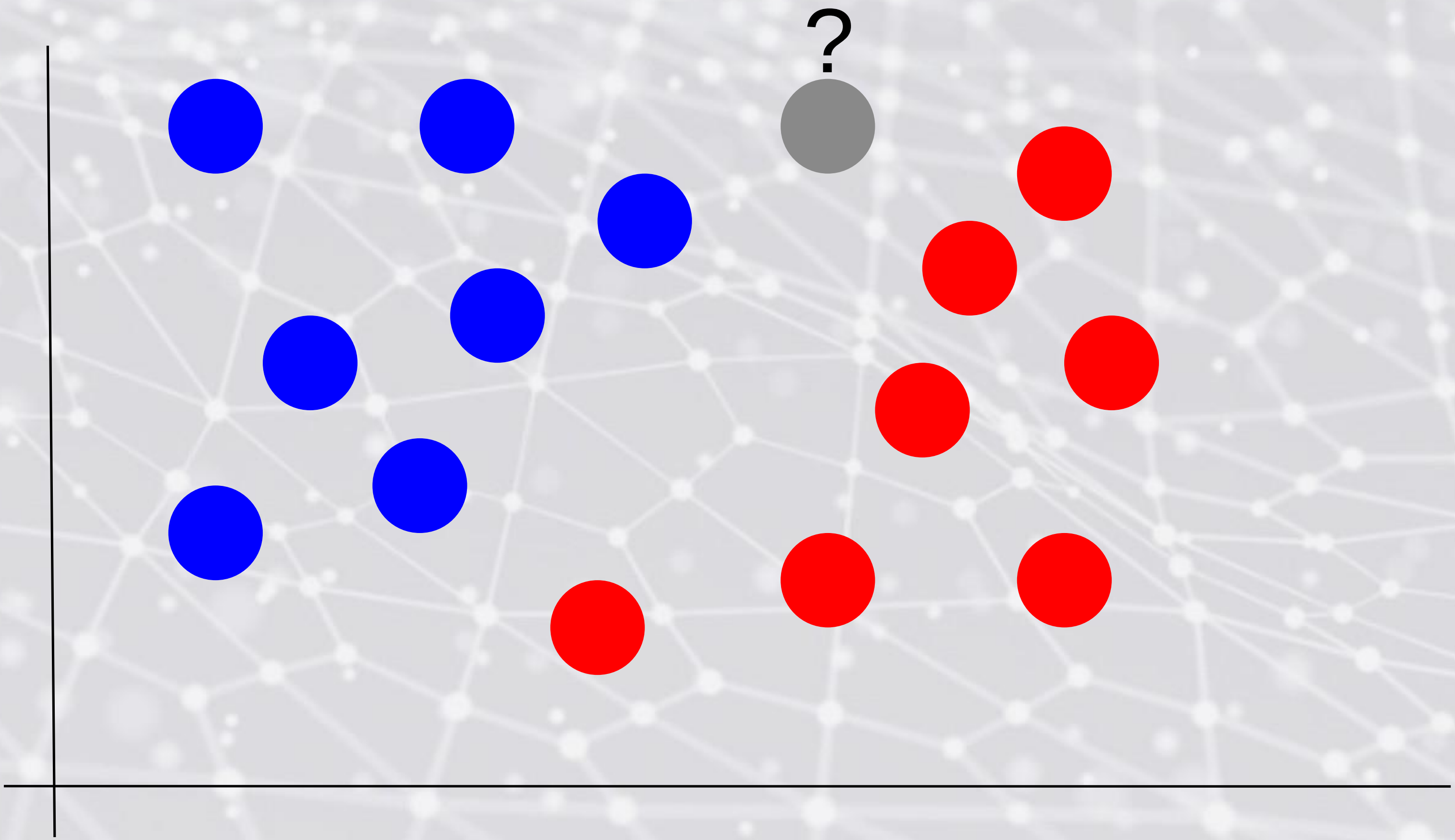
Graphical Example of Classification



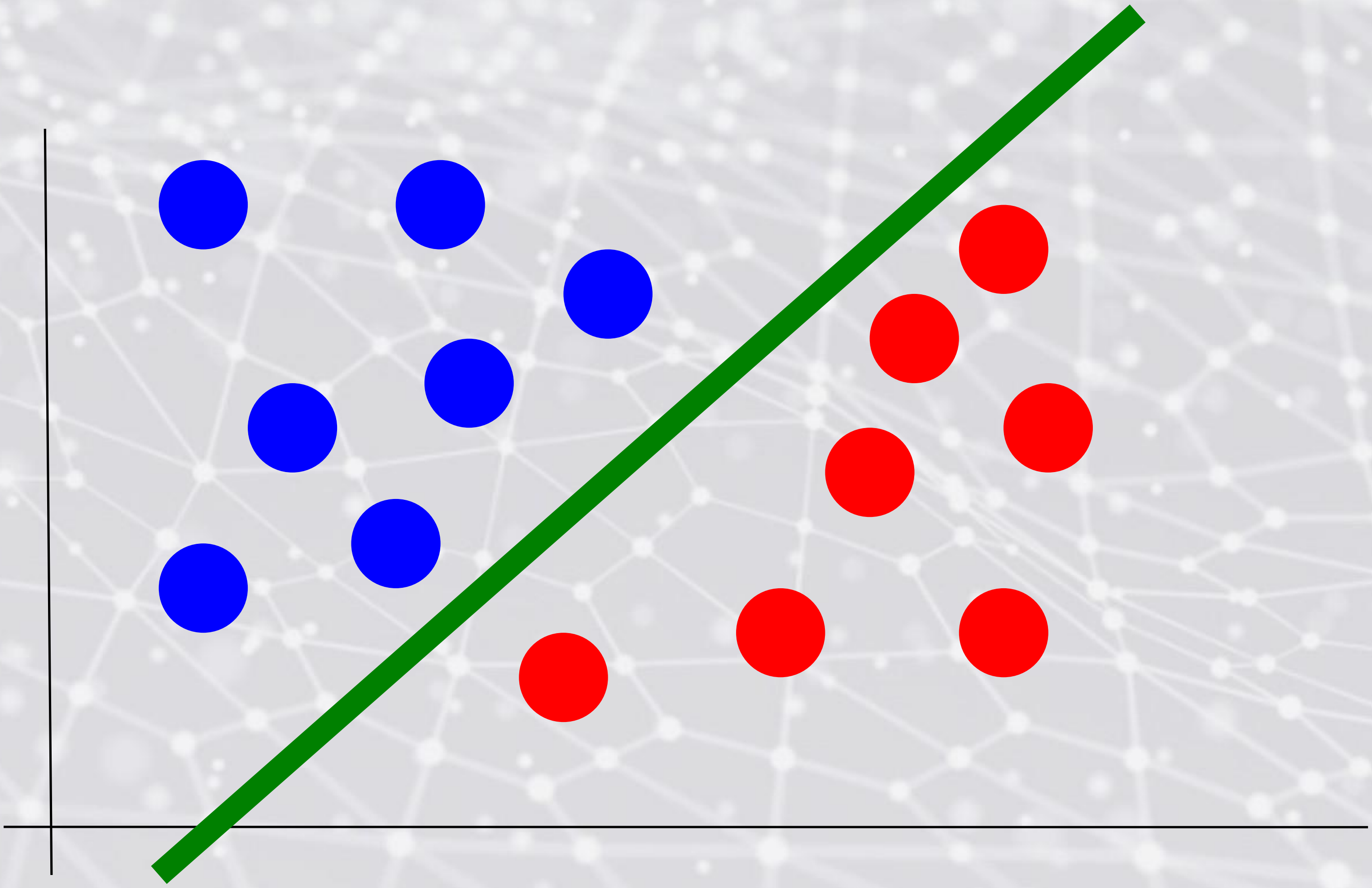
Graphical Example of Classification



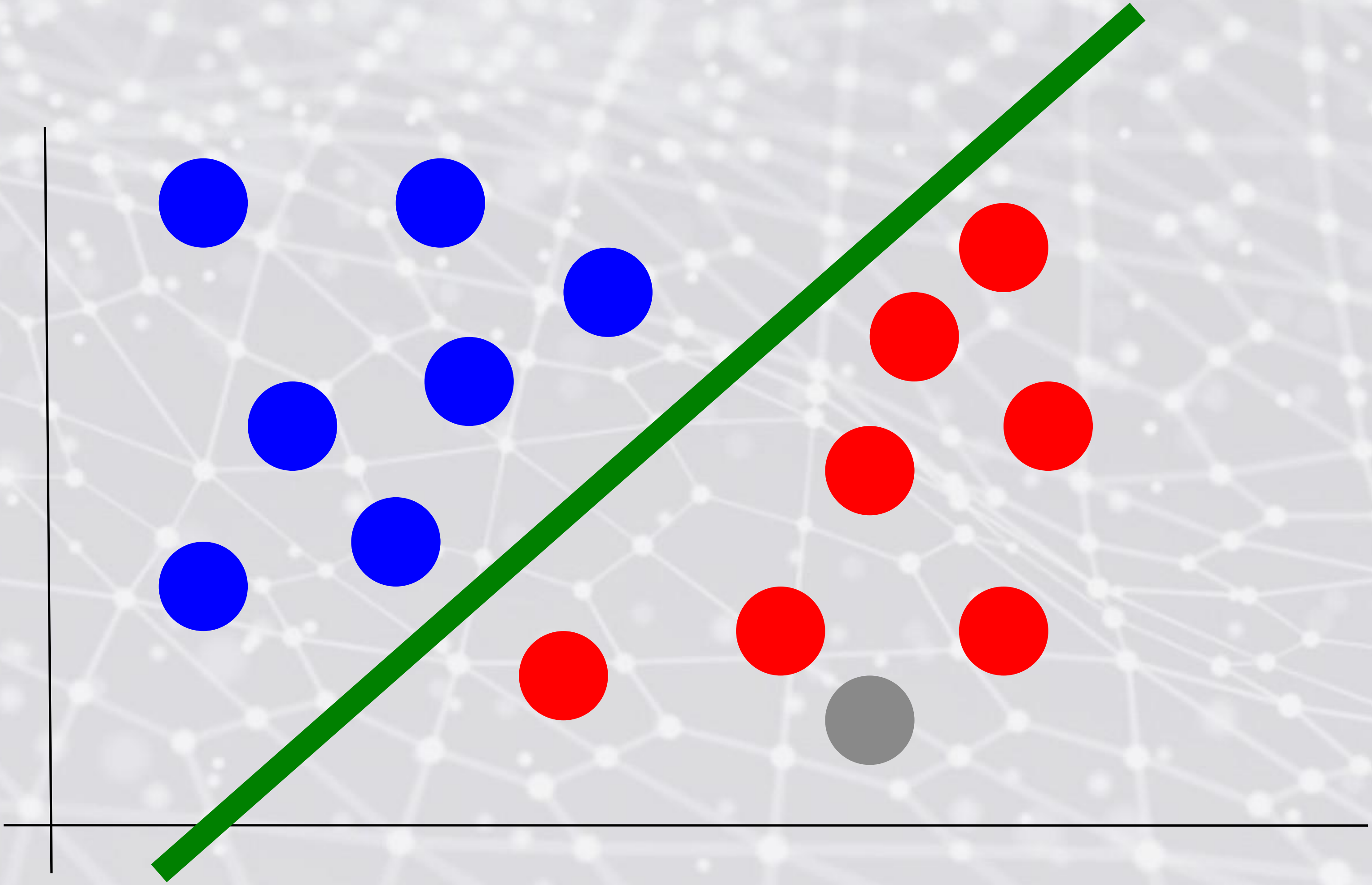
Graphical Example of Classification



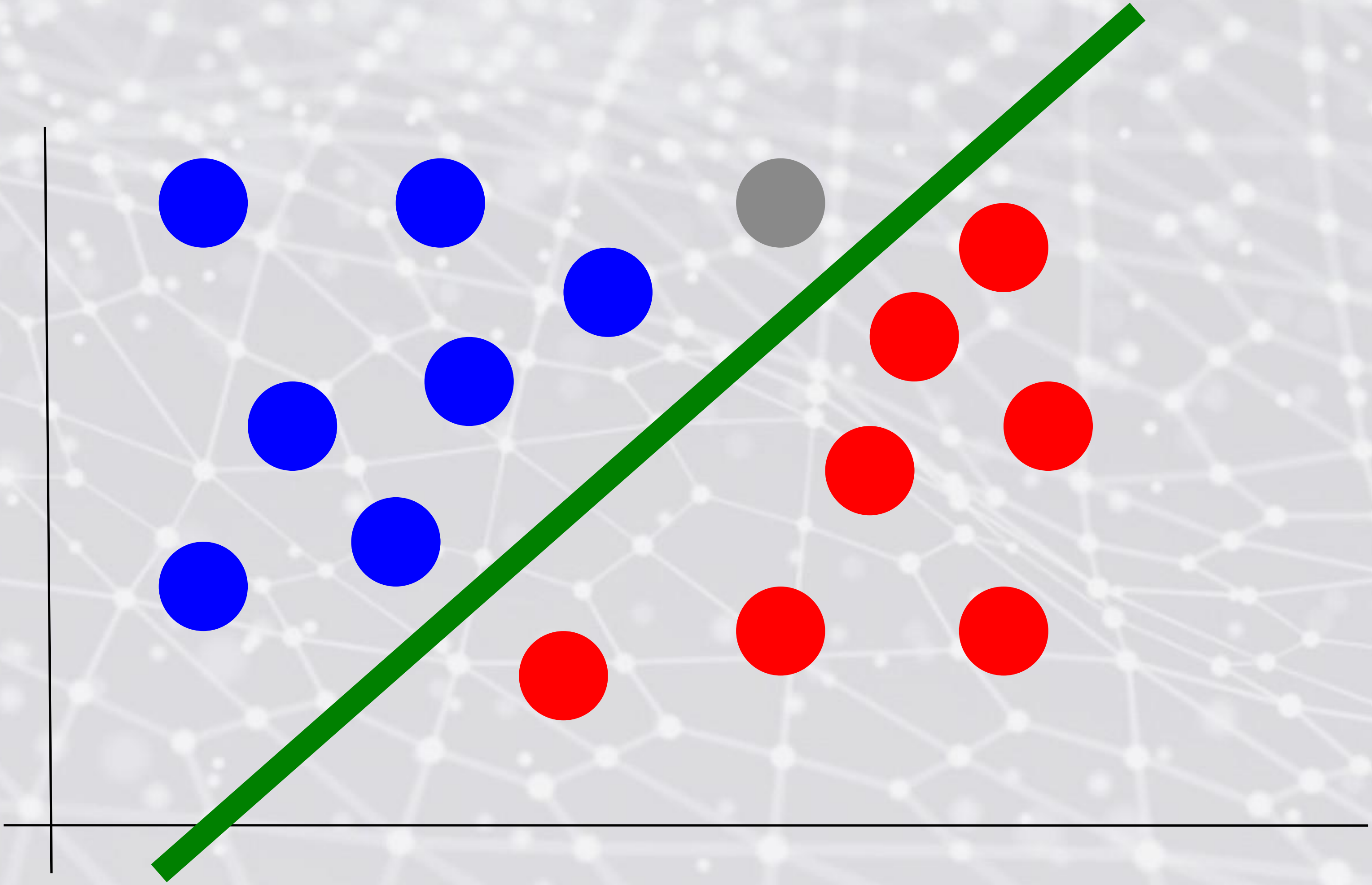
Graphical Example of Classification



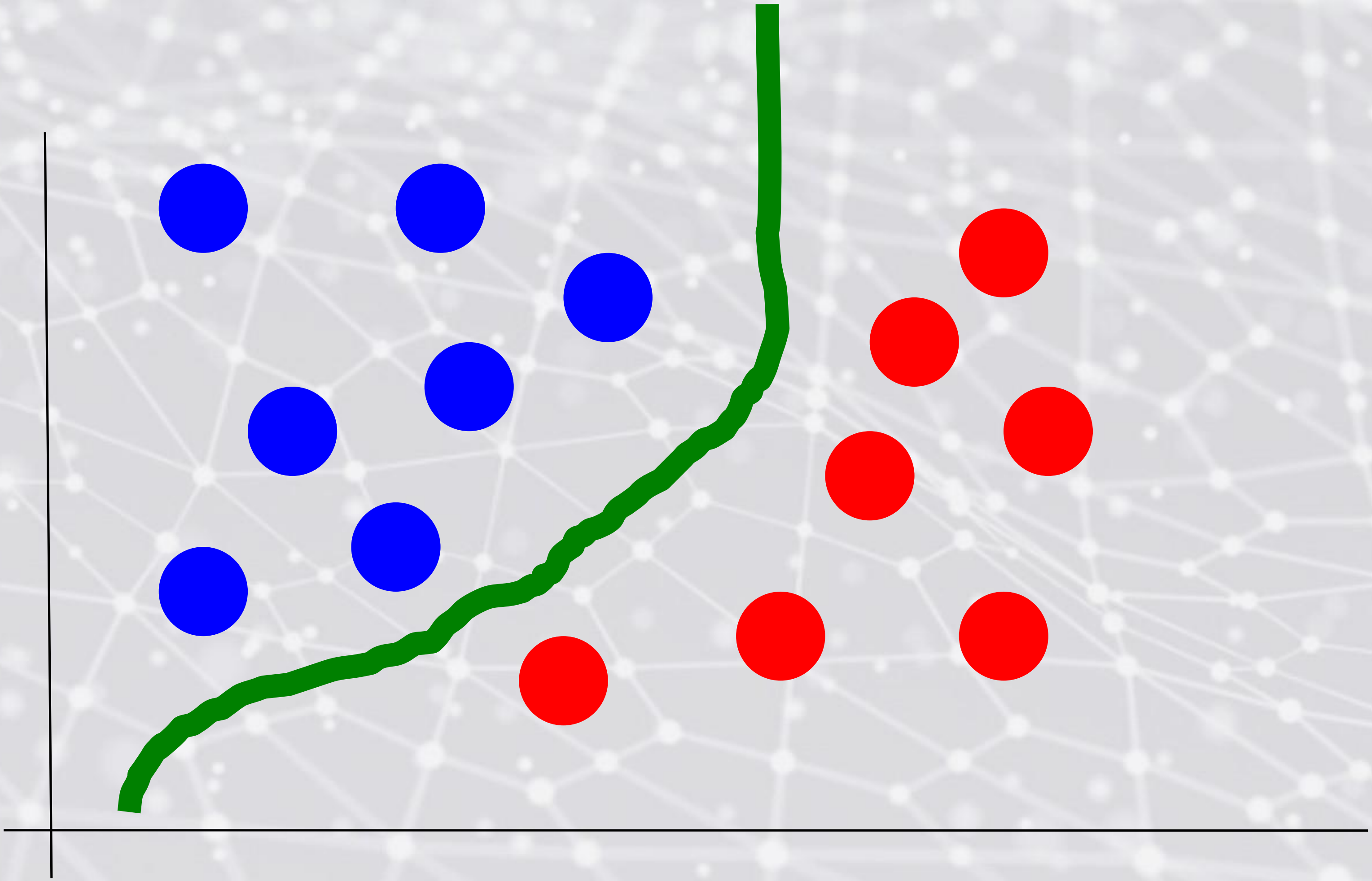
Graphical Example of Classification



Graphical Example of Classification



Decision Boundaries



Regression

- Regression is a **supervised** machine learning task.
  - So a target value, **t**, is given.
- Classification: nominal **t** $t \in \{c_0, \ldots, c_{N-1}\}$
- Regression: continuous **t** $t \in \mathbb{R}$

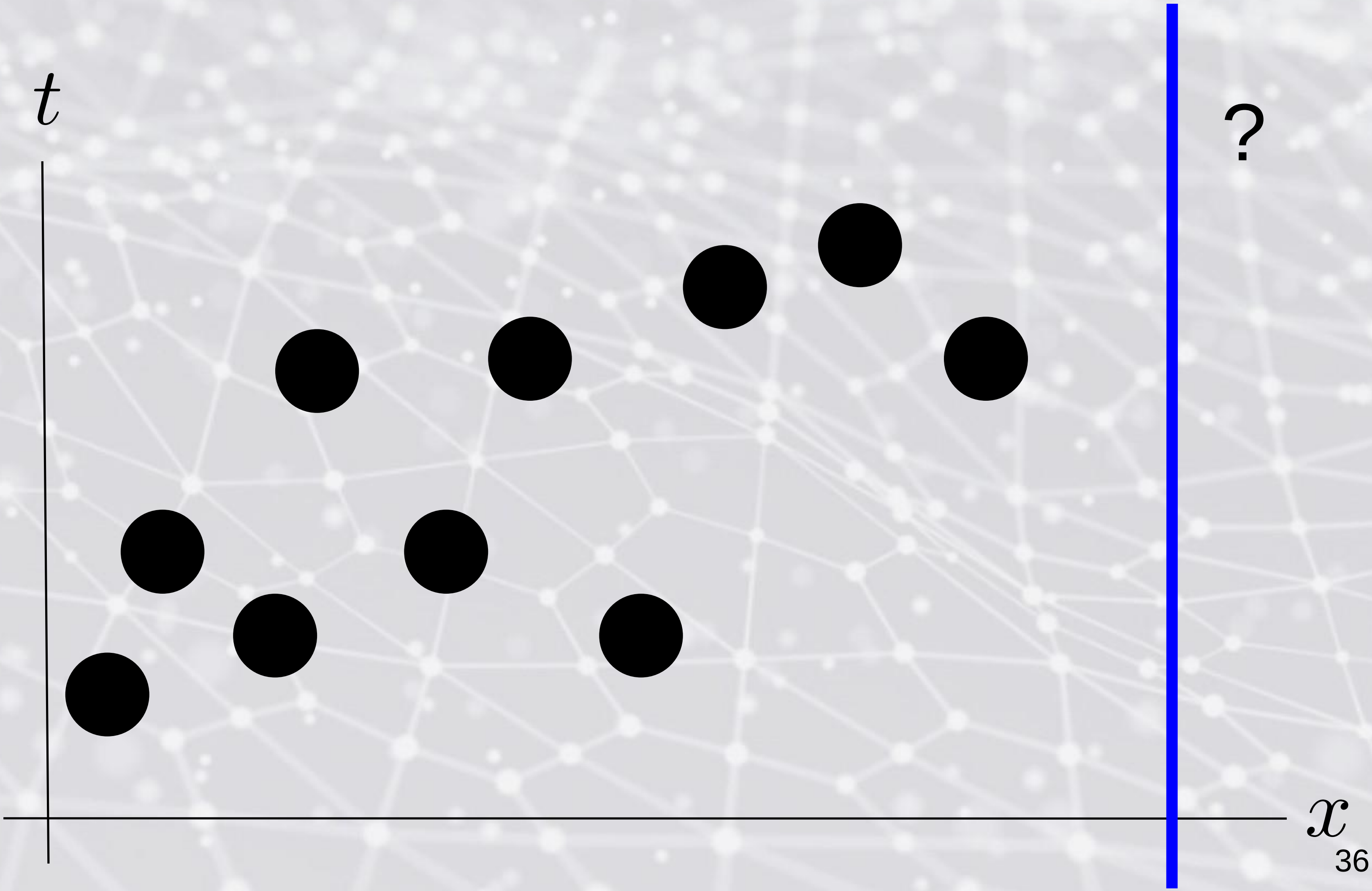
Goal of Classification

Identify a function *y*, such that *y*(**x**) = **t**

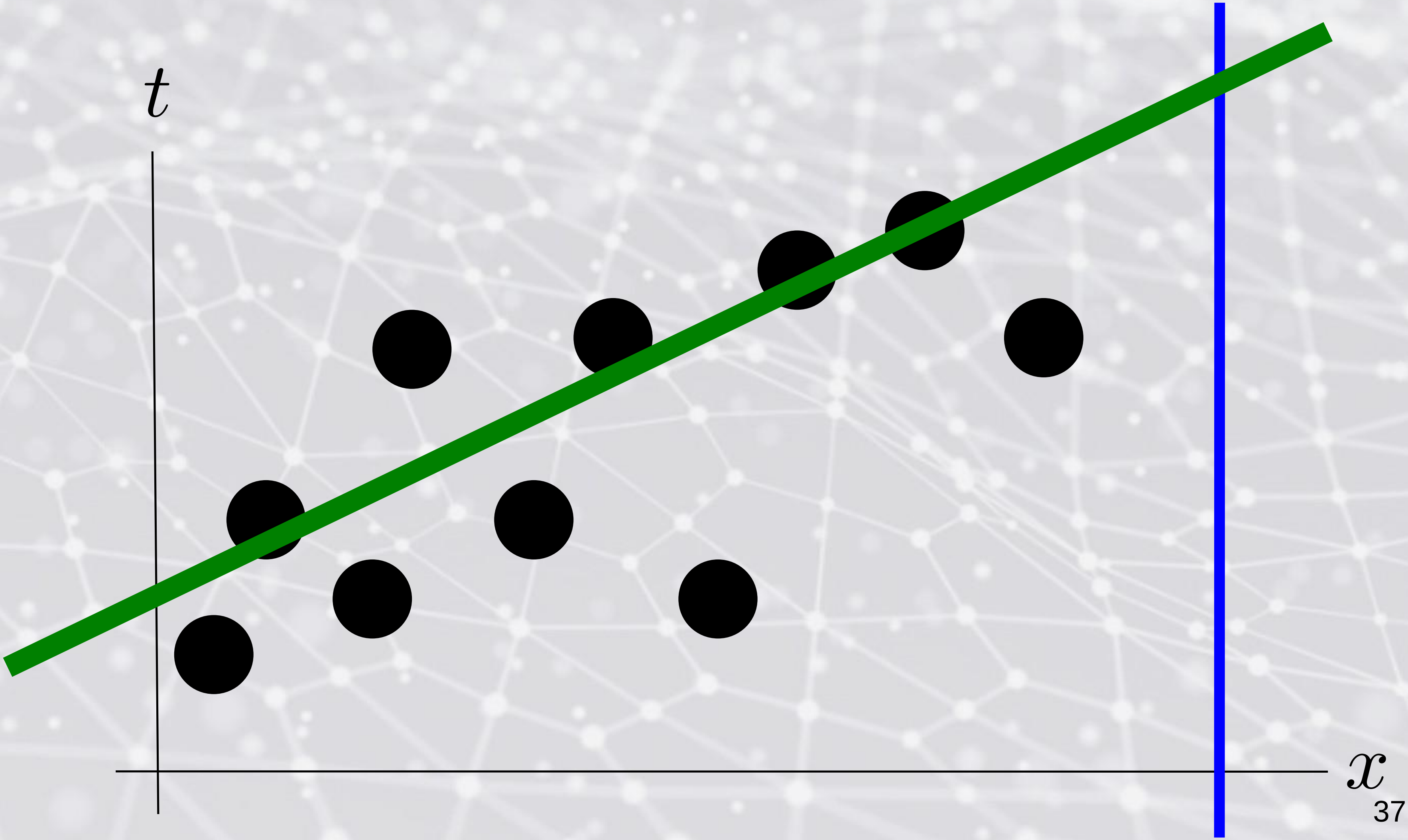
Differences between Classification and Regression

- Similar goals: Identify y(**x**) = **t**.
- What are the differences?
  - – The form of the function, y (naturally).
  - – Evaluation
    - Root Mean Squared Error
    - Absolute Value Error
    - Classification Error
    - Maximum Likelihood
  - – Evaluation drives the optimization operation that learns the function, y.

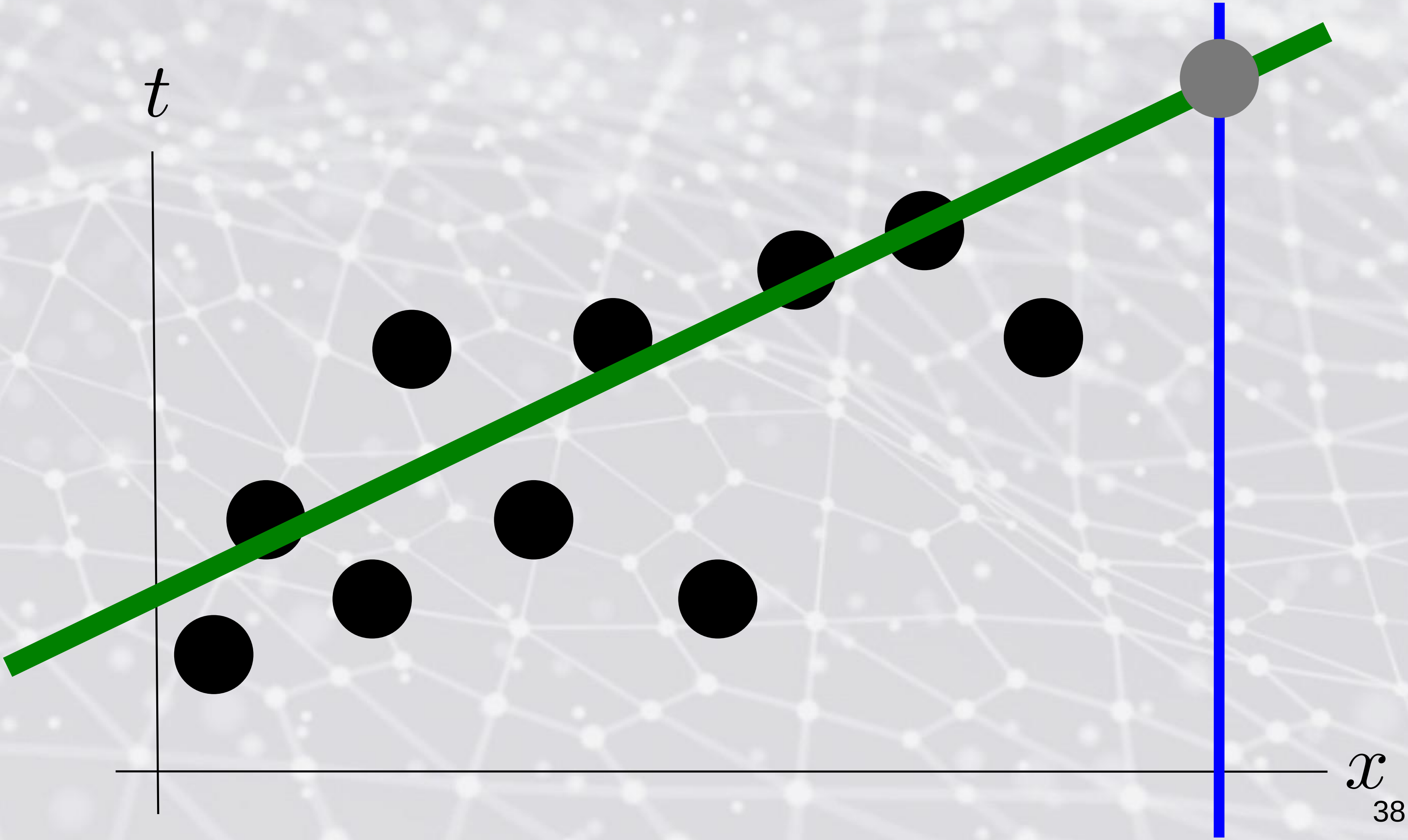
Graphical Example of Regression



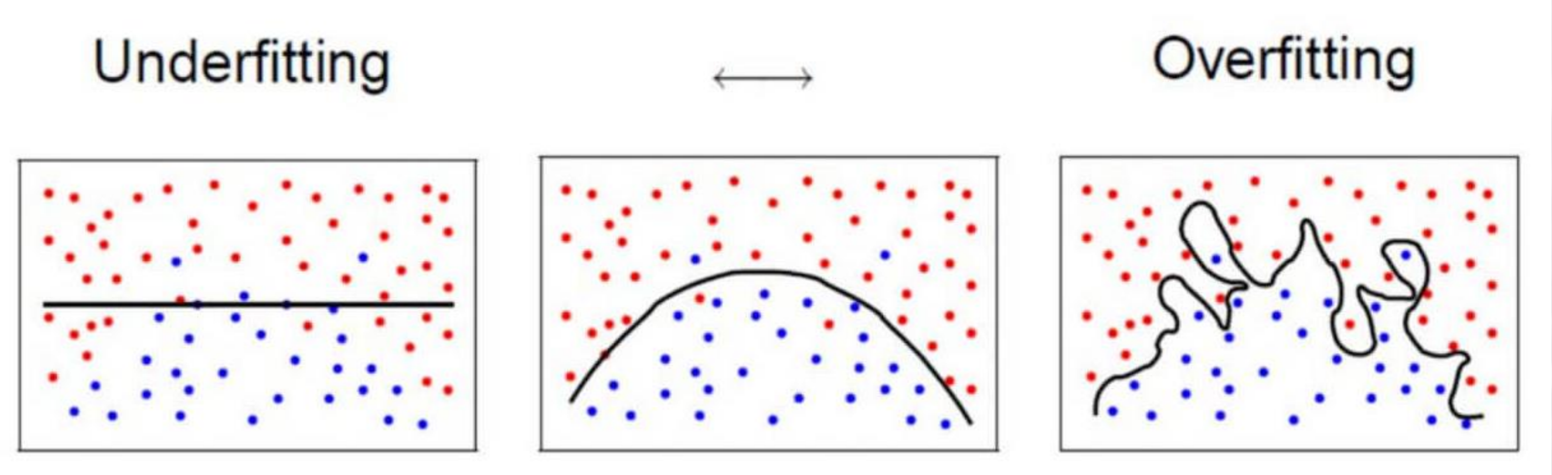
Graphical Example of Regression

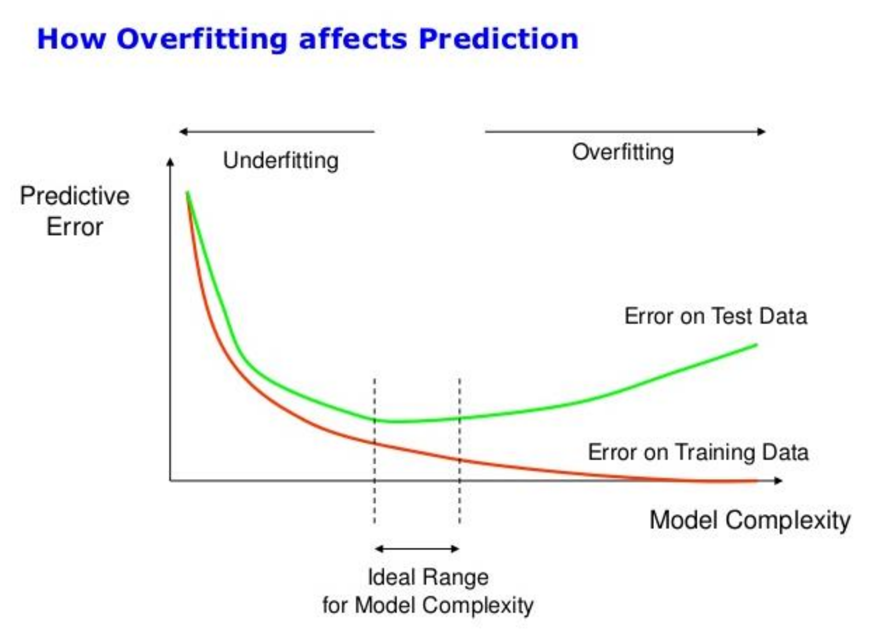


Graphical Example of Regression

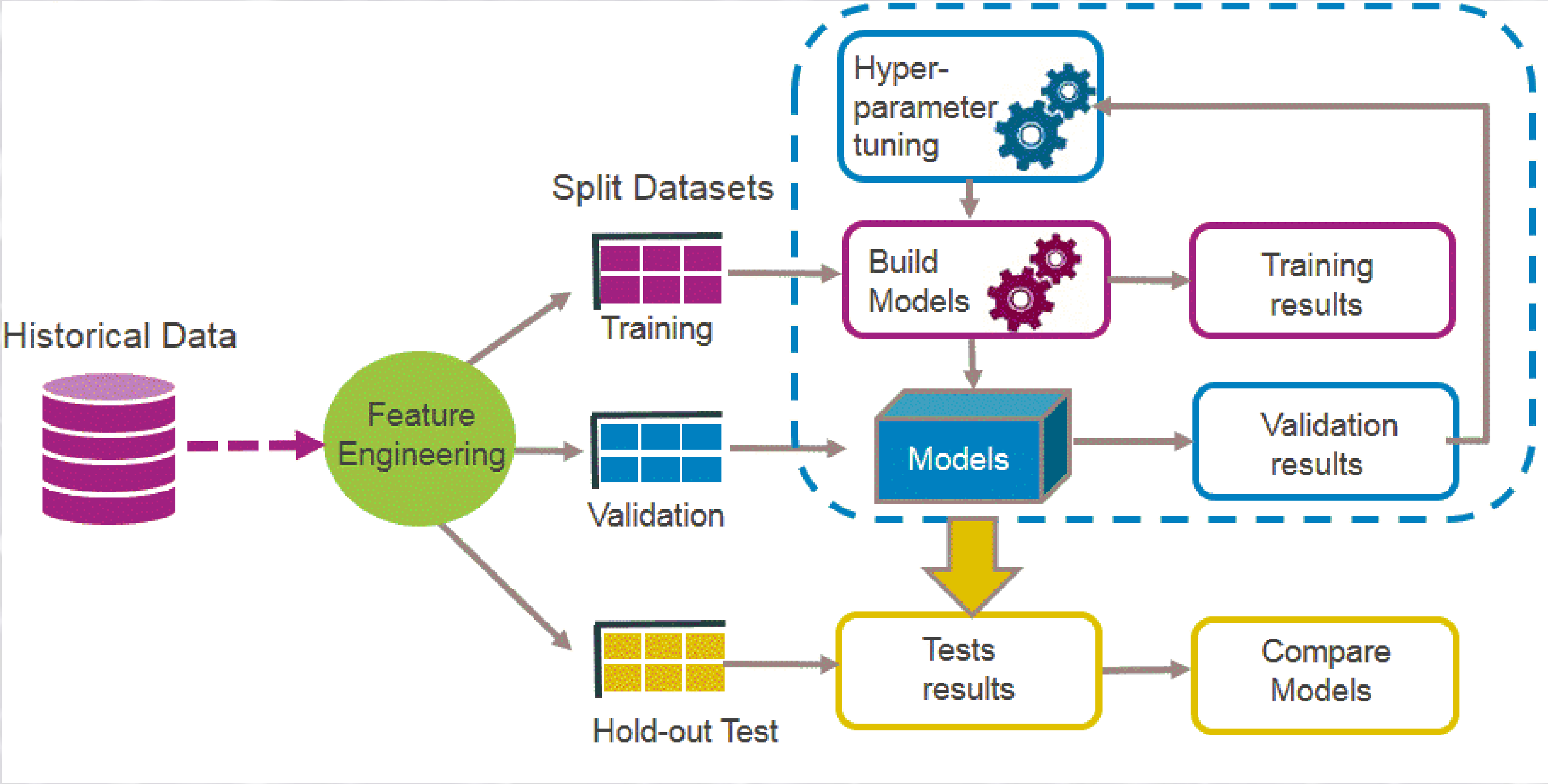


Generalization Problem in Prediction/Classification

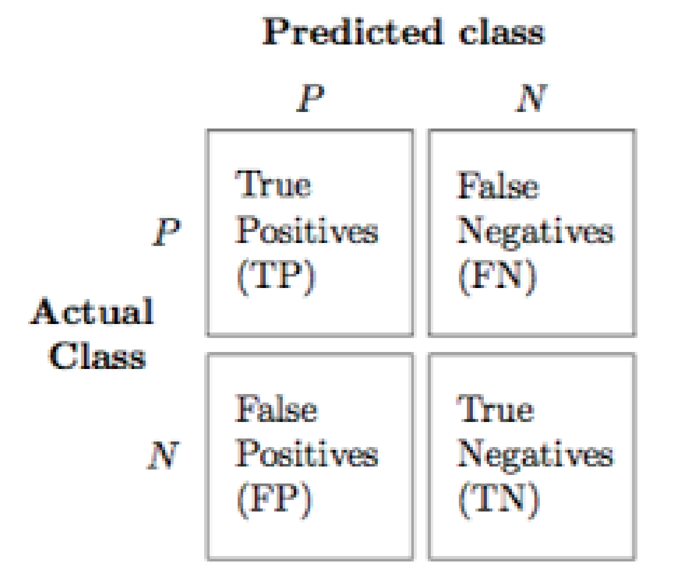




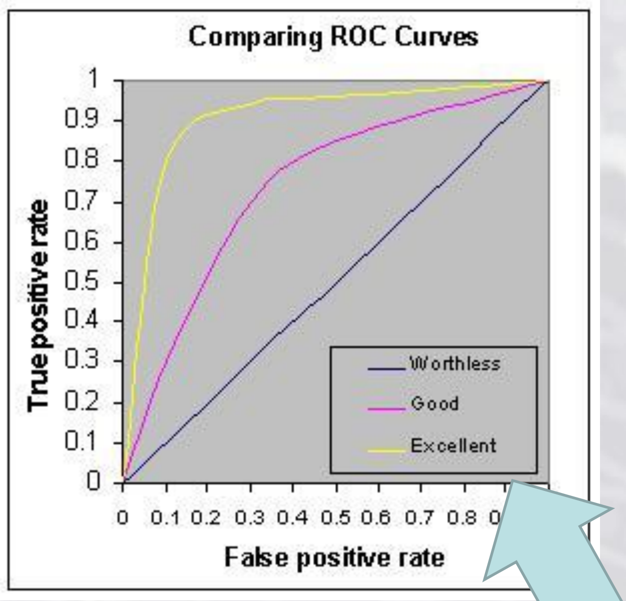
Common ML Pipeline

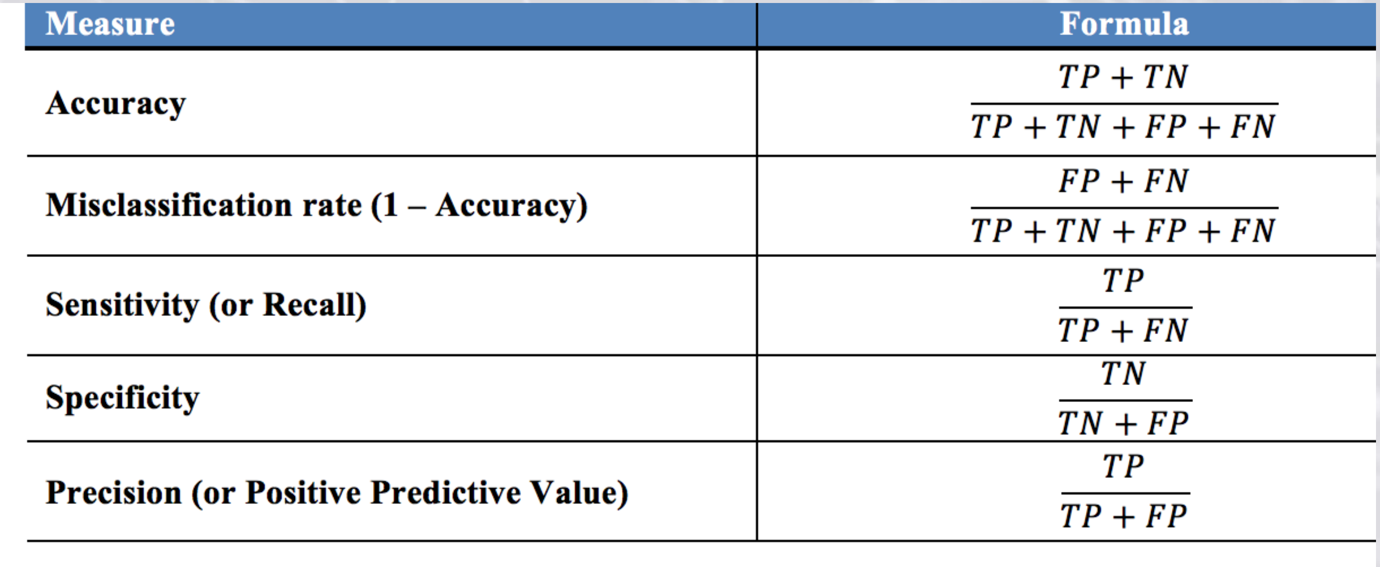


Confusion Matrix, ROC curves, etc.



		Predicted class	
		P	*N*
Actual Class	*P*	True Positives (TP)	False Negatives (FN)
	N	False Positives (FP)	True Negatives (TN)





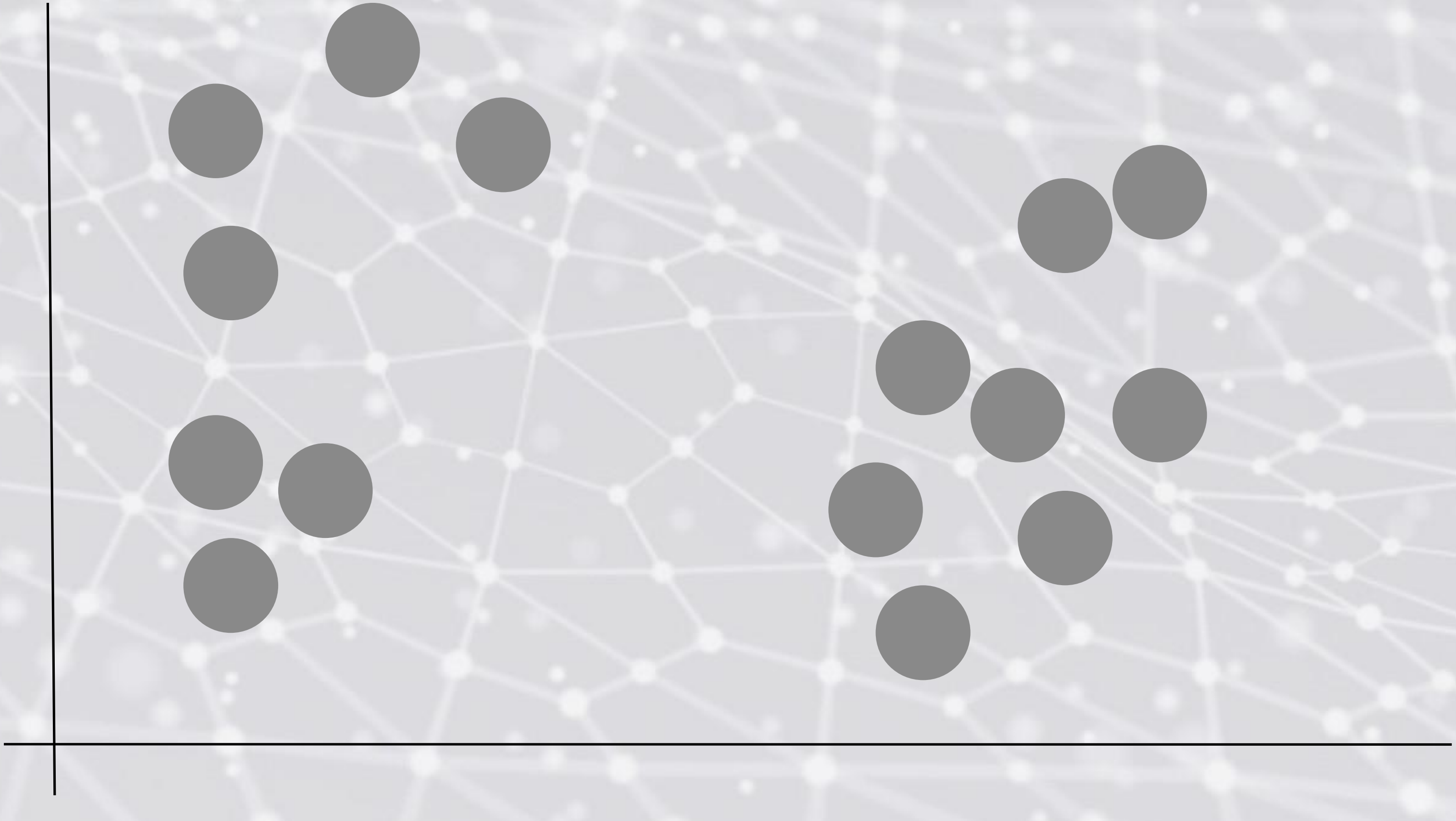
Measure	Formula
Accuracy	$\frac{TP+TN}{TP+TN+FP+FN}$
Misclassification rate (1 – Accuracy)	$\frac{FP+FN}{TP+TN+FP+FN}$
Sensitivity (or Recall)	$\frac{TP}{TP+FN}$
Specificity	$\frac{TN}{TN+FP}$
Precision (or Positive Predictive Value)	$\frac{TP}{TP+FP}$

Area under (the) curve (AUC) is a common metric used to assess/compare classifiers

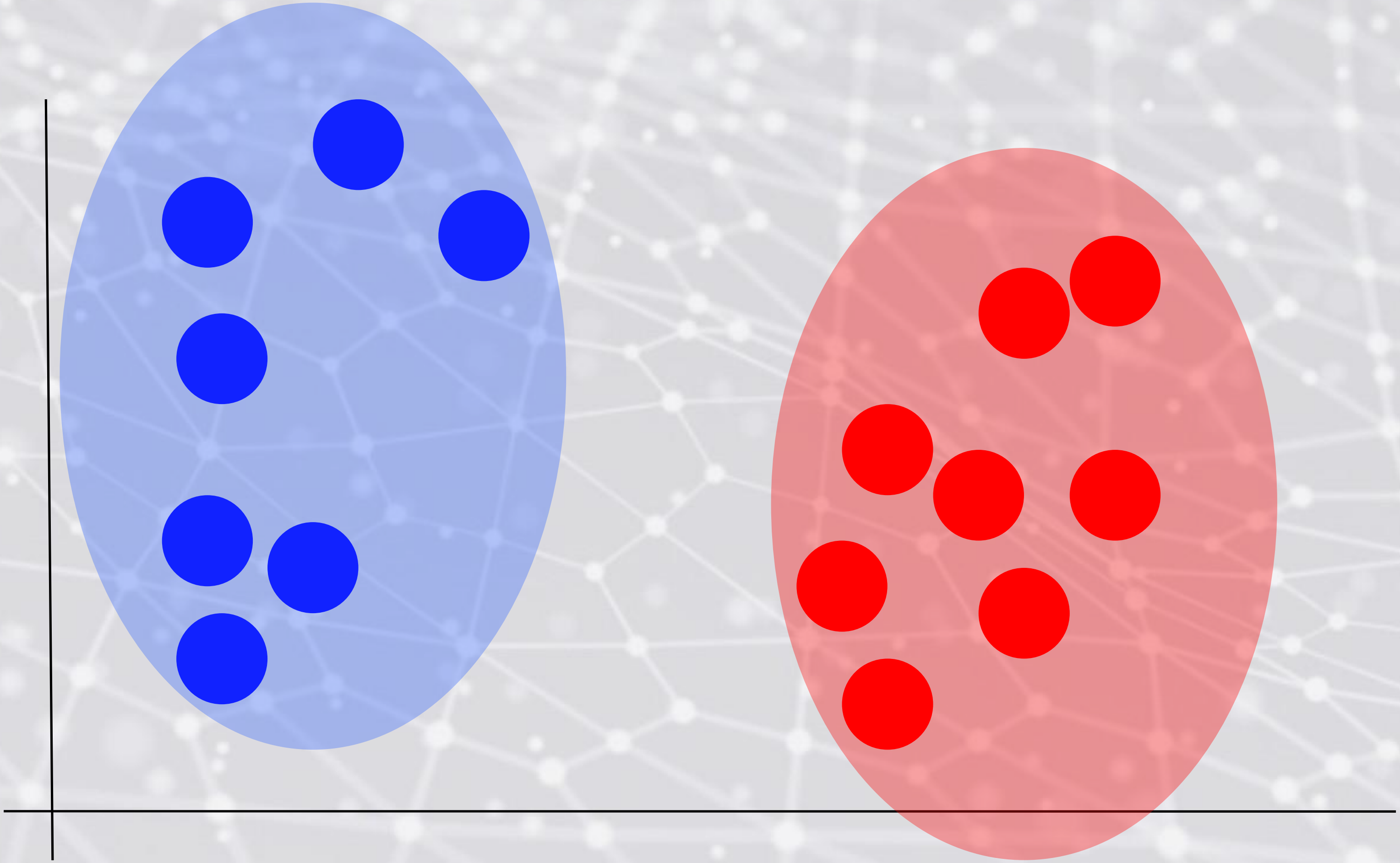
Clustering

- Clustering is an **unsupervised** learning task.
  - There is no target value to shoot for.
- Identify groups of “similar” data points, that are “dissimilar” from others.
- **Partition** the data into groups (clusters) that satisfy these constraints
  1. Points in the same cluster should be **similar.**
  2. Points in different clusters should be **dissimilar.**

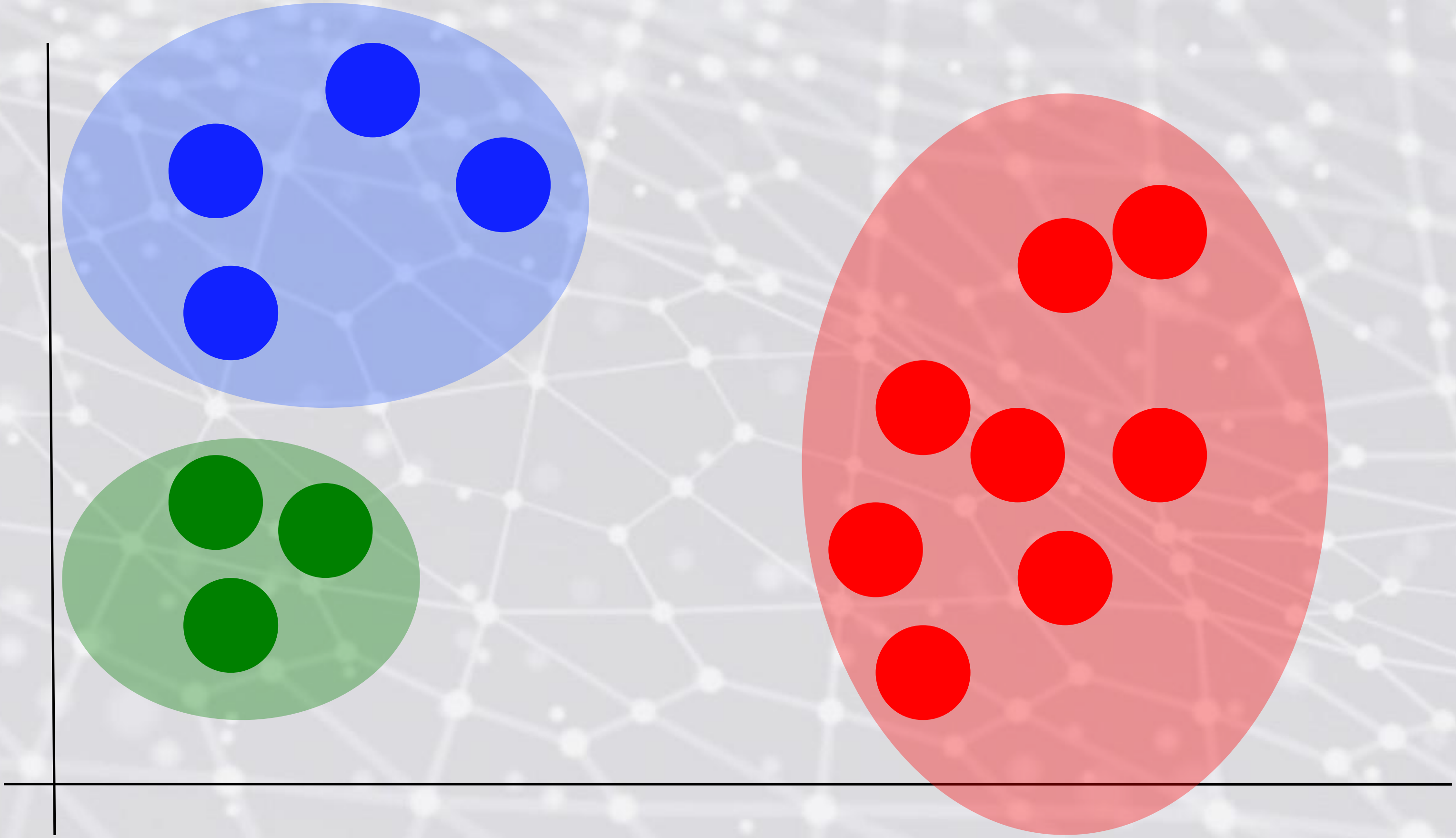
Graphical Example of Clustering



Graphical Example of Clustering

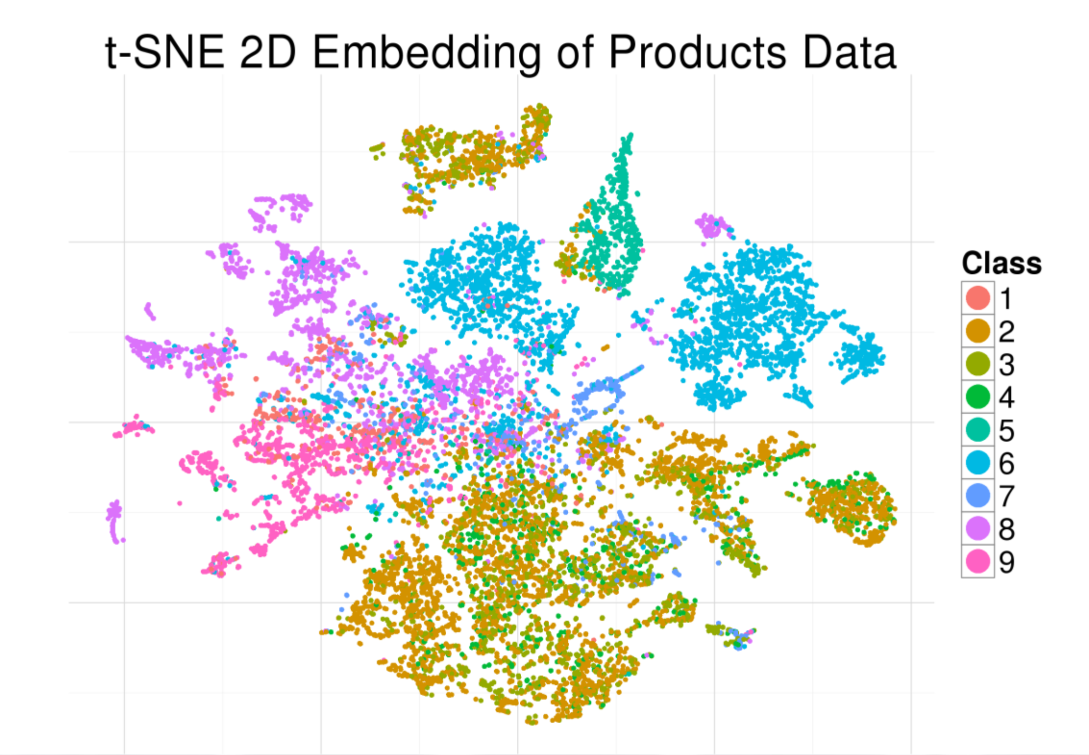


Graphical Example of Clustering



MNIST Classification

- 60k training/10k test images
- LeCun, Bengio, *et al.* (1998) used SVMs to get error rate of 0.8%.
- More recent research using CNNs (a type of neural network) yields 0.23% error.

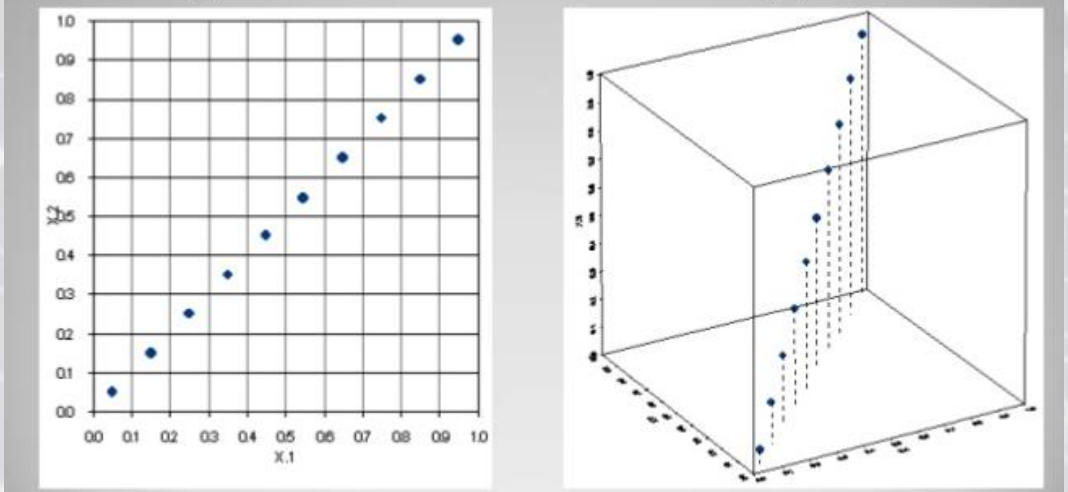


The Curse of Dimensionality

- In ML we are faced with a fundamental dilemma: to maintain a given model accuracy in higher dimensions we need a huge amount of data!
- An exponential increase in data required to densely populate space as the dimension increases.
- Points are equally far apart in high dimensional space (this is counter-intuitive).

Representation of 10% sample probability space

(i) 2-D (ii)3-D



The Number of Points Would Need to Increase Exponentially to Maintain a Given Accuracy.

10^n samples would be required for a *n*-dimension problem.

Dealing with High Dimensionality

What can we do?

- Use Domain Knowledge

  -- Feature engineering

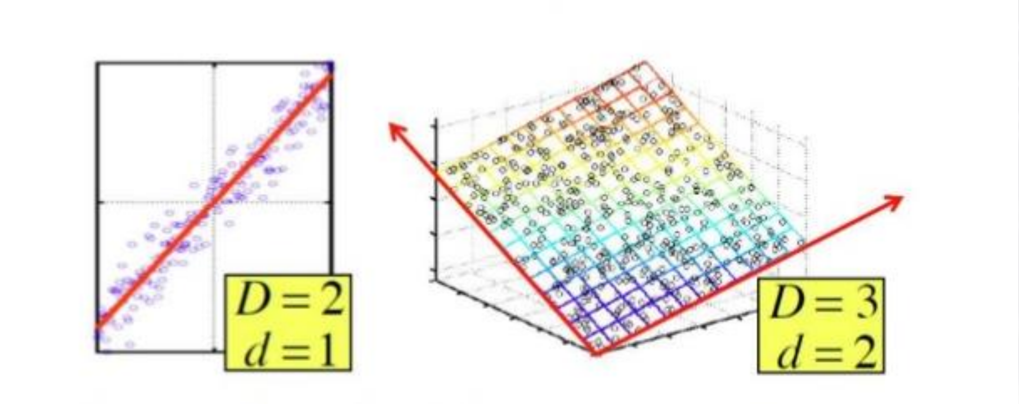
- Make assumptions about dimensions

  -- Independence: Count along each dimension separately

  -- Smoothness: Propagate class counts to neighboring regions

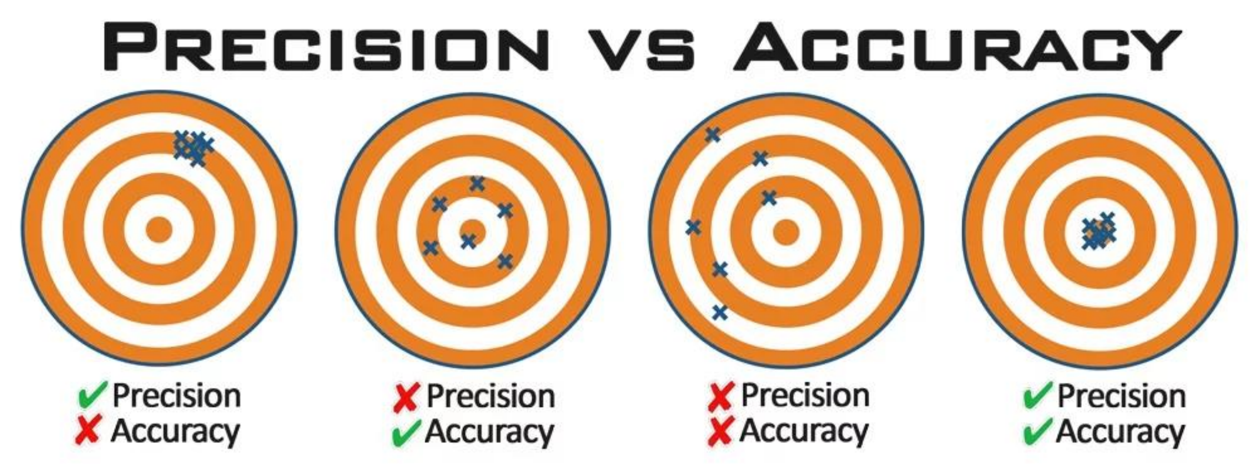
  -- Symmetry: e.g., invariance to order of dimensions

- Perform dimensionality reduction



Bias-Variance Tradeoff

- Whenever we train any type of ML algorithm/model we are making some <u>model choices</u>, and fitting the parameters of that model.
- The more degrees of freedom (dof) the algorithm has, the more complicated the model that can be fitted (recall: overfitting).
- Note that a model can be "bad" for (2) basic reasons: (1) it is inaccurate and doesn't match the data well; (2) it is not very precise, meaning that the there is a lot of variation in the results.
- (1) is known as bias; (2) is statistical variance.



Bias-Variance Tradeoff

- The MSE (mean-squared error) decouples to reflect what is known as the bias-variance tradeoff:

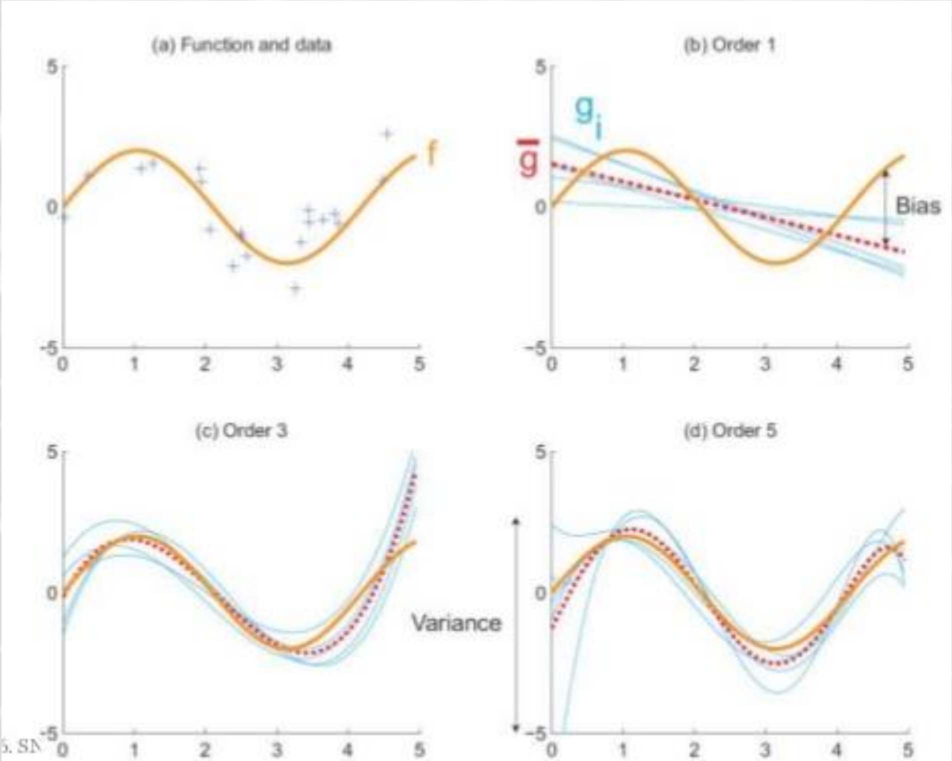
$$
\begin{aligned}
\mathrm{MSE}(\hat{\theta}) \equiv \mathbb{E}((\hat{\theta}-\theta)^2) &= \mathbb{E}\left[\left(\hat{\theta}-\mathbb{E}(\hat{\theta})+\mathbb{E}(\hat{\theta})-\theta\right)^2\right] \\
&= \mathbb{E}\left[\left(\hat{\theta}-\mathbb{E}(\hat{\theta})\right)^2+2\left((\hat{\theta}-\mathbb{E}(\hat{\theta}))(\mathbb{E}(\hat{\theta})-\theta)\right)+\left(\mathbb{E}(\hat{\theta})-\theta\right)^2\right] \\
&= \mathbb{E}\left[\left(\hat{\theta}-\mathbb{E}(\hat{\theta})\right)^2\right]+2\mathbb{E}\left[(\hat{\theta}-\mathbb{E}(\hat{\theta}))(\mathbb{E}(\hat{\theta})-\theta)\right]+\mathbb{E}\left[\left(\mathbb{E}(\hat{\theta})-\theta\right)^2\right] \\
&= \mathbb{E}\left[\left(\hat{\theta}-\mathbb{E}(\hat{\theta})\right)^2\right]+\boxed{2(\mathbb{E}(\hat{\theta})-\theta)\overbrace{\mathbb{E}(\hat{\theta}-\mathbb{E}(\hat{\theta}))}^{=\mathbb{E}(\hat{\theta})-\mathbb{E}(\hat{\theta})=0}}+\mathbb{E}\left[\left(\mathbb{E}(\hat{\theta})-\theta\right)^2\right] \\
&= \mathbb{E}\left[\left(\hat{\theta}-\mathbb{E}(\hat{\theta})\right)^2\right]+\mathbb{E}\left[\left(\mathbb{E}(\hat{\theta})-\theta\right)^2\right] \\
&= \mathrm{Var}(\hat{\theta})+\mathrm{Bias}(\hat{\theta},\theta)^2
\end{aligned}
$$

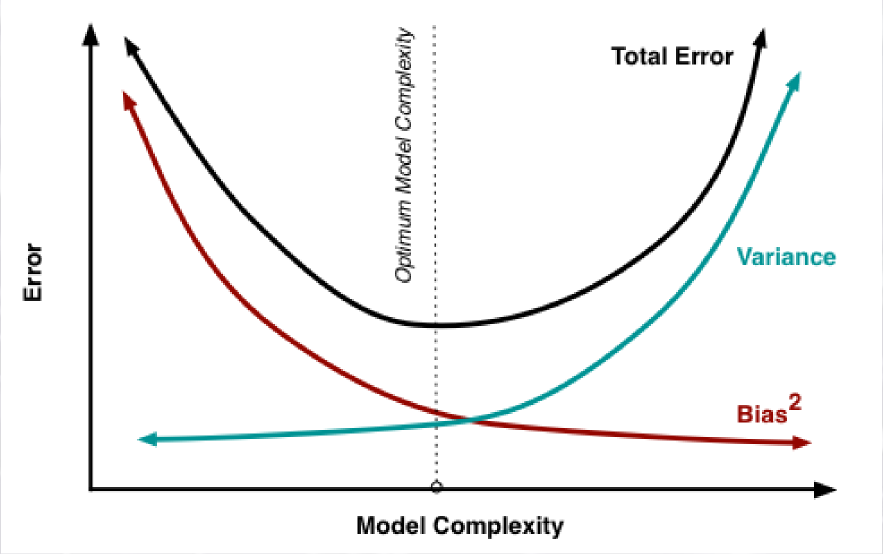
Where: $\theta := \textit{true parameter value}$

$\hat{\theta} := \textit{parameter estimate}$

Bias-Variance Tradeoff

- In pictures…


(a) Function and data
f
(b) Order 1
g_i
$\bar{g}$
Bias
(c) Order 3
Variance
(d) Order 5
5. SN


Error
Model Complexity
Optimum Model Complexity
Total Error
Variance
Bias^2