UNIVERSITY OF ICELAND
SCHOOL OF ENGINEERING AND NATURAL SCIENCES
FACULTY OF INDUSTRIAL ENGINEERING, MECHANICAL ENGINEERING AND COMPUTER SCIENCE

JÜLICH
FORSCHUNGSZENTRUM

JLESC Summer School

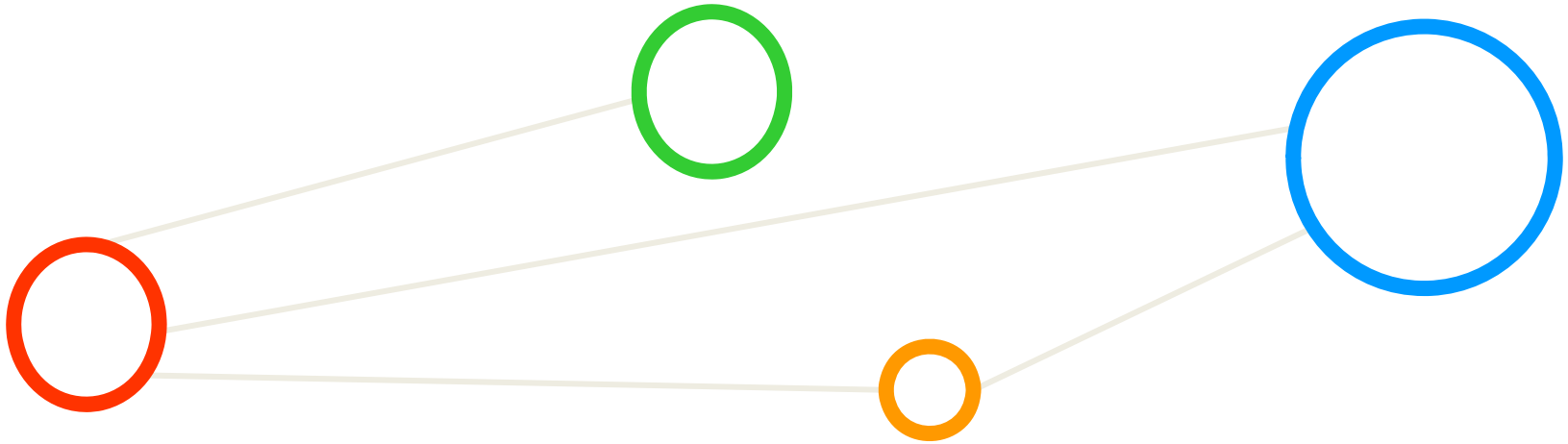
Data Analytics
Machine Learning – Welcome & Introduction

Morris Riedel
Jülich Supercomputing Center (JSC) // University of Iceland

Markus Götz
Jülich Supercomputing Center (JSC) // University of Iceland

Christian Bodenstein
Jülich Supercomputing Center (JSC) // University of Iceland

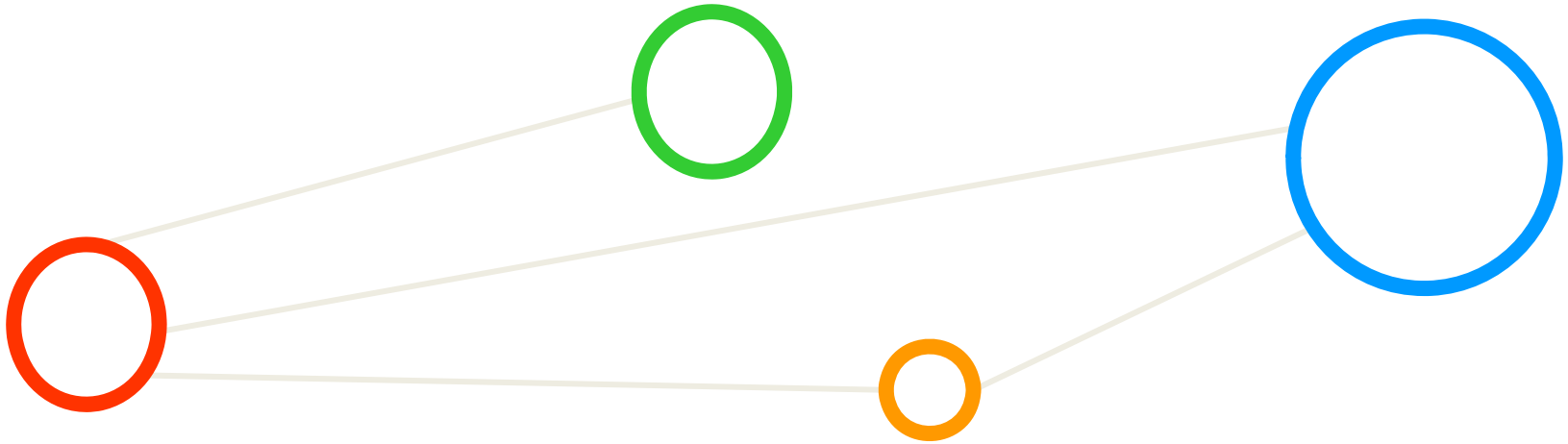
Data Analytics – Welcome & Introduction



Data Analytics – Welcome & Introduction – Agenda

1. Welcome & Introduction (Morris Riedel) ~ 0:45 min
   - Scientific Big Data Analytics
   - Statistical Learning Theory Basics
2. Classification (Morris Riedel) ~ 1:00
   - Highly Parallel piSVM tool; validation, importance of feature extraction
   - Exercise (hands-on), remote sensing application
3. Clustering (Markuz Goetz) ~ 0:45
   - Highly Parallel DBSCAN tool; unsupervised learning
   - Exercise (hands-on), earth quake application
4. Deep Learning (Christian Bodenstein) ~ 0:45
   - Recent approaches in learning from data
   - Exercise (hands-on), various applications

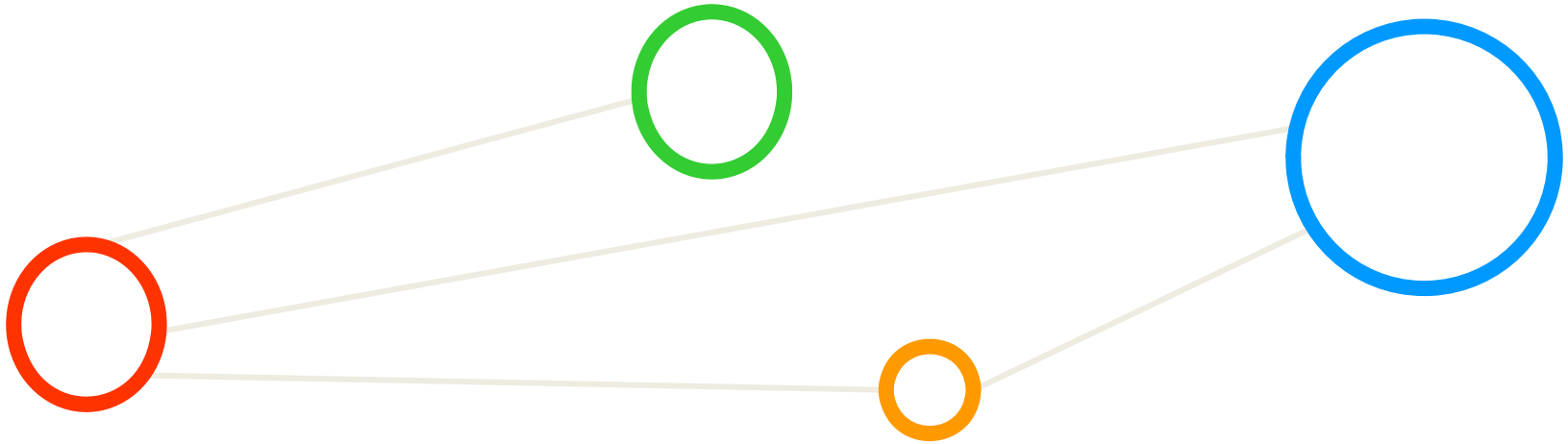
Data Analytics – Welcome & Introduction – Outline



Data Analytics – Welcome & Introduction – Outline

- Introduction
  - Statistical Data Mining & Machine Learning
  - Scientific Big Data Analytics
  - Unsupervised vs. Supervised Learning Approaches
- Statistical Learning Theory Basics
  - *(Any proper tutorial needs to include these elements at least shortly)*
  - Feasibility of Learning
  - Mathematical Building Blocks (work against the ‚danger zone', cf. talk yesterday)
  - In-Sample vs. Out-of-Sample Performance
  - Error Measures & Noisy Targets
  - Theory of Generalization (in short)
  - Problem of Overfitting
  - Theory of Regularization (in short)
- Backup Slides

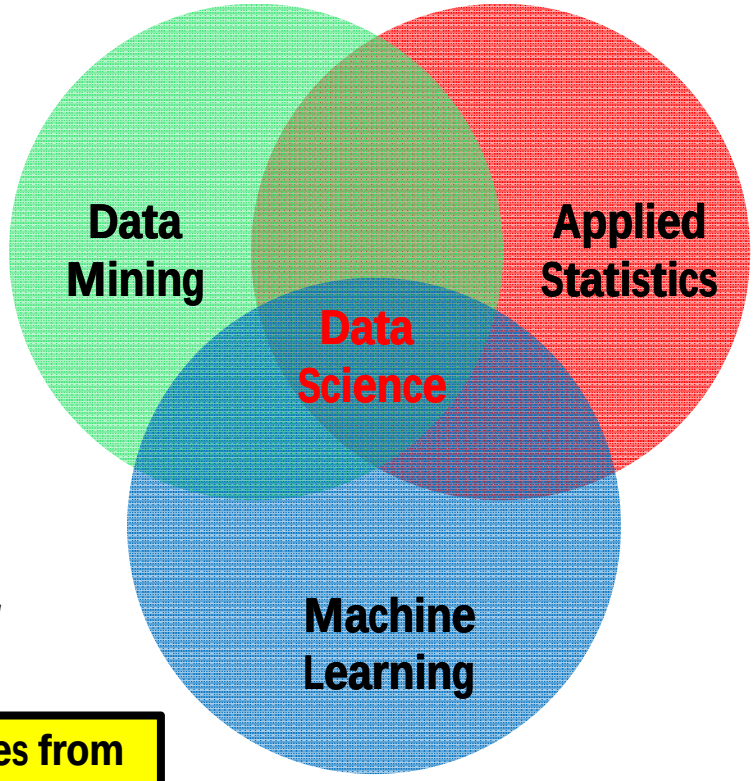
Scientific Big Data Analytics



Statistical Data Mining & Data Analytics

1. Some pattern exists
2. No exact mathematical formula
3. Data exists

- Idea ‘Learning from Data‘ shared with a wide variety of other disciplines
  - E.g. signal processing, etc.



‘People with statistical learning skills are in high demand.’

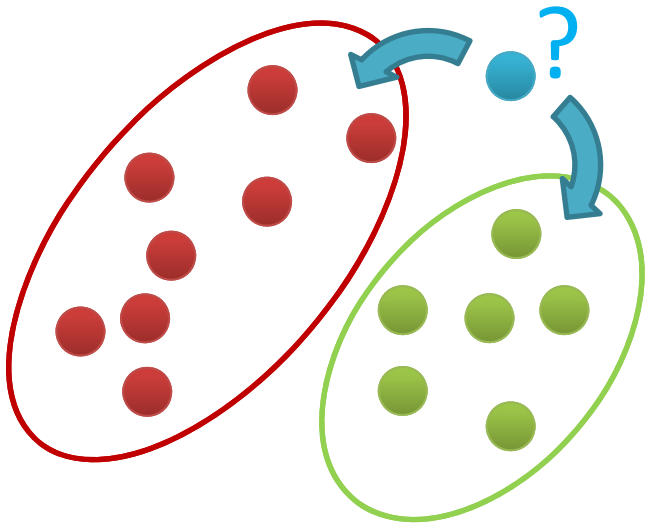
[1] An Introduction to Statistical Learning

- **Statistical Data Mining is a very broad subject and goes from very abstract theory to extreme practice (‘rules of thumb’)**

Selected Machine Learning Methods

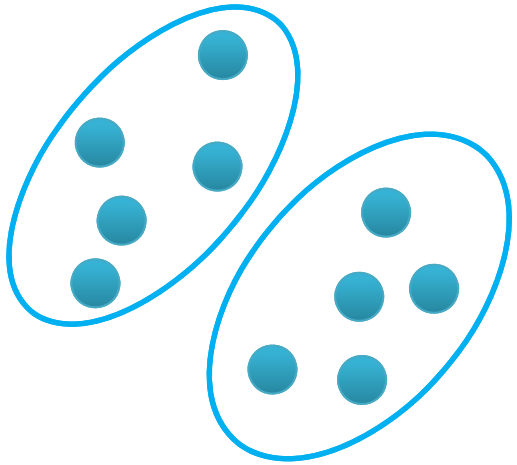
- **Statistical data mining methods can be roughly categorized in classification, clustering, or regression augmented with various (statistical) techniques for feature extraction or reduction**

Classification



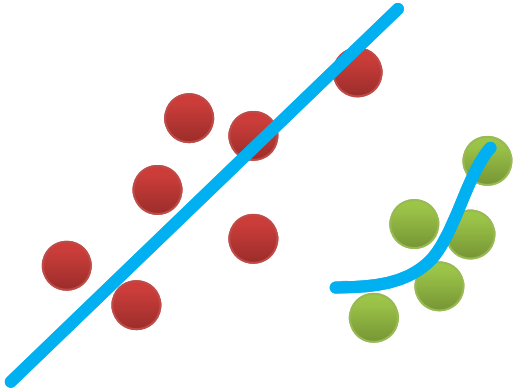
- Groups of data exist
- New data classified to existing groups

Clustering



- No groups of data exist
- Create groups from data close to each other

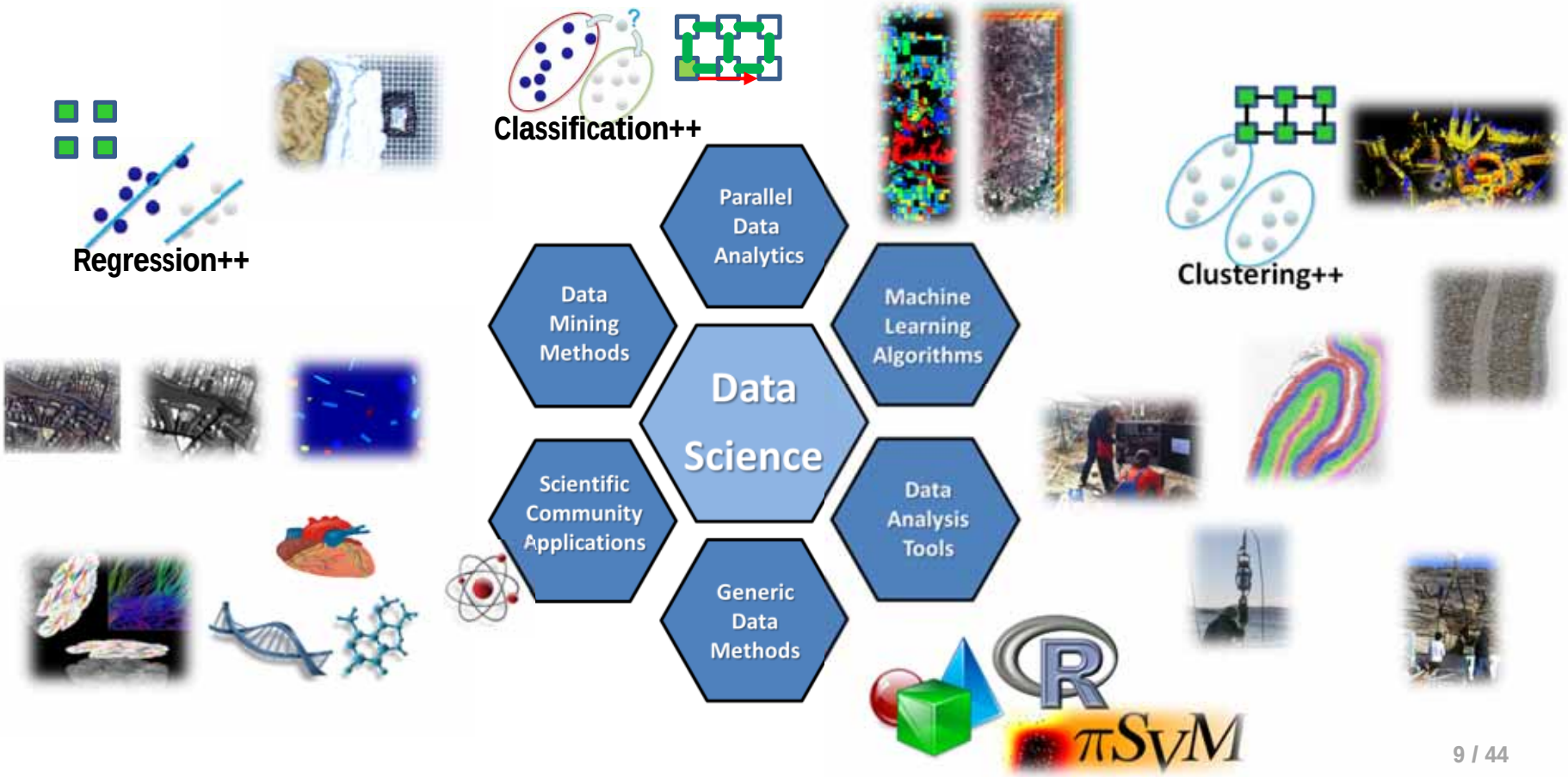
Regression



- Identify a line with a certain slope describing the data

Scientific Big Data Analytics

- Consists of ‘interesting mix’ of different approaches
  - Parallel computing techniques with machine learning/data mining methods
  - Goal: Advance science & engineering with data-intensive applications



Learning Approaches – Unsupervised Learning

- Each observation of the predictor measurement(s) has no associated response measurement:
  - Input $\mathbf{x} = x_1, \ldots, x_d$
  - No output
  - Data $(\mathbf{x}_1), \ldots, (\mathbf{x}_N)$

➢ **Lecture on Clustering by Markus Goetz will give more details & insights**

- Goal: Seek to understand relationships between the observations
  - Clustering analysis: check whether the observations fall into distinct groups
- Challenges
  - No response/output that could supervise our data analysis
  - Clustering groups that overlap might be hardly recognized as distinct group

- **Unsupervised learning approaches seek to understand relationships between the observations**
- **Unsupervised learning approaches are used in clustering algorithms such as k-means, etc.**
- **Unupervised learning works with data = [input, ---]**

[1] An Introduction to Statistical Learning

Learning Approaches – Supervised Learning

- Each observation of the predictor measurement(s) has an associated response measurement:
  - Input $\mathbf{x} = x_1, ..., x_d$
  - Output $y_i, i = 1, .., n$
  - Data $(\mathbf{x}_1, y_1), ..., (\mathbf{x}_N, y_N)$

Training Examples

$(\mathbf{x}_1, y_1), ..., (\mathbf{x}_N, y_N)$

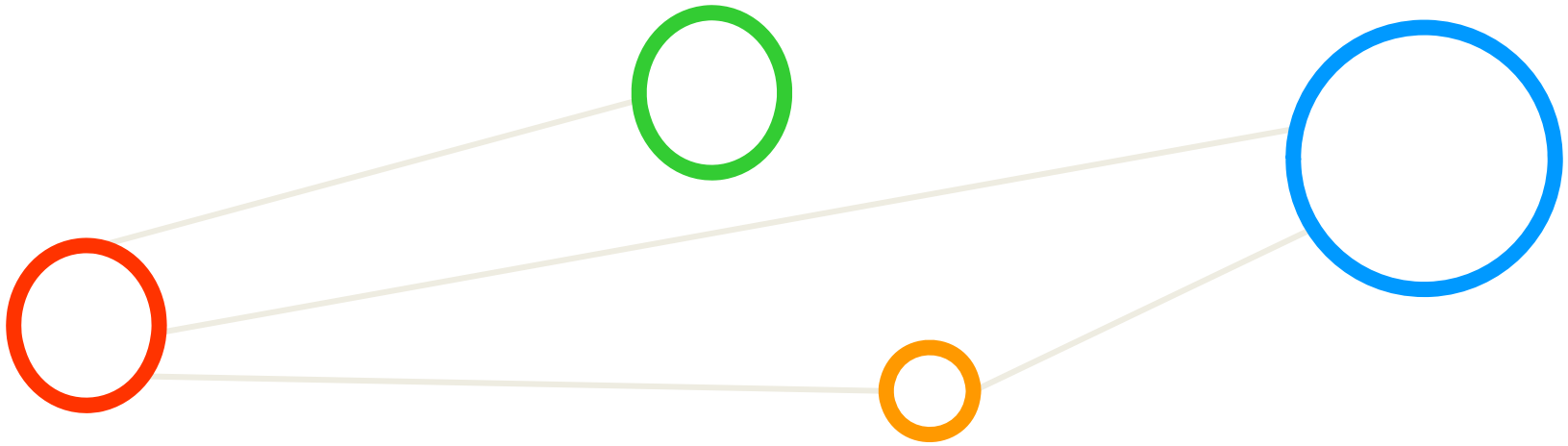
(historical records, groundtruth data, examples)

- Goal: Fit a model that relates the response to the predictors
  - Prediction: Aims of accurately predicting the response for future observations
  - Inference: Aims to better understanding the relationship between the response and the predictors

- **Supervised learning approaches fits a model that related the response to the predictors**
- **Supervised learning approaches are used in linear/logistic regression or support vector machines**
- **Supervised learning works with data = [input, correct output]**

[1] An Introduction to Statistical Learning

Statistical Learning Theory Basics



Feasibility of Learning

- **Statistical Learning Theory deals with the problem of finding a predictive function based on data**

[2] Wikipedia on 'statistical learning theory'

- Theoretical framework underlying practical learning algorithms
  - E.g. Support Vector Machines (SVMs)
  - Best understood for 'Supervised Learning'

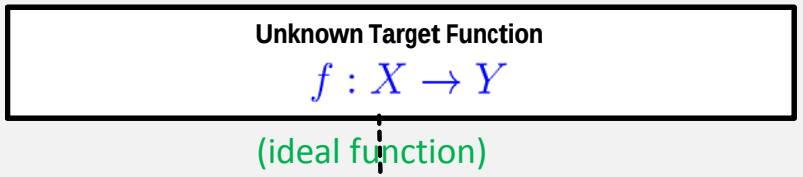
- Theoretical background used to solve 'A learning problem'
  - Inferring one 'target function' that maps between input and output
  - Learned function can be used to predict output from future input (fitting existing data is not enough)

Unknown Target Function

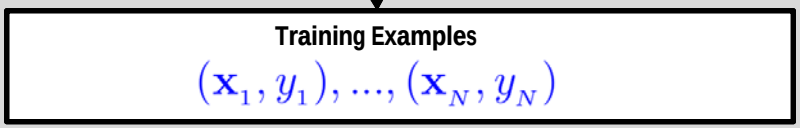
$$f : X \rightarrow Y$$

(ideal function – we will never know)

Mathematical Building Blocks (1)



(ideal function)



(historical records, groundtruth data, examples)

Elements we not exactly (need to) know

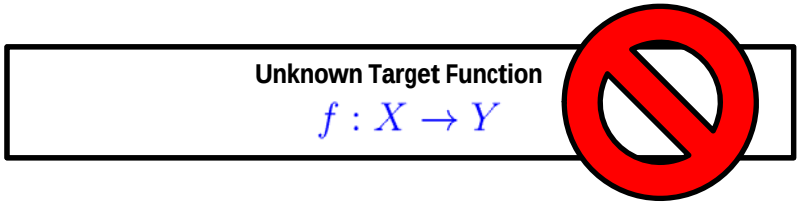
Elements we must and/or should have and that might raise huge demands for storage

Elements that we derive from our skillset and that can be computationally intensive

Elements that we derive from our skillset

Feasibility of Learning – Hypothesis Set & Final Hypothesis

- The ‘ideal function‘ will remain unknown in learning

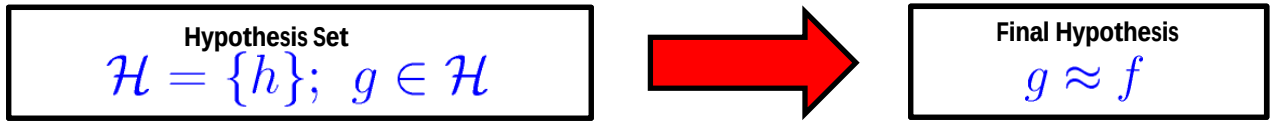


  - Impossible to know and learn from data
  - If known a straightforward implementation would be better than learning
  - E.g. hidden features/attributes of data not known or not part of data

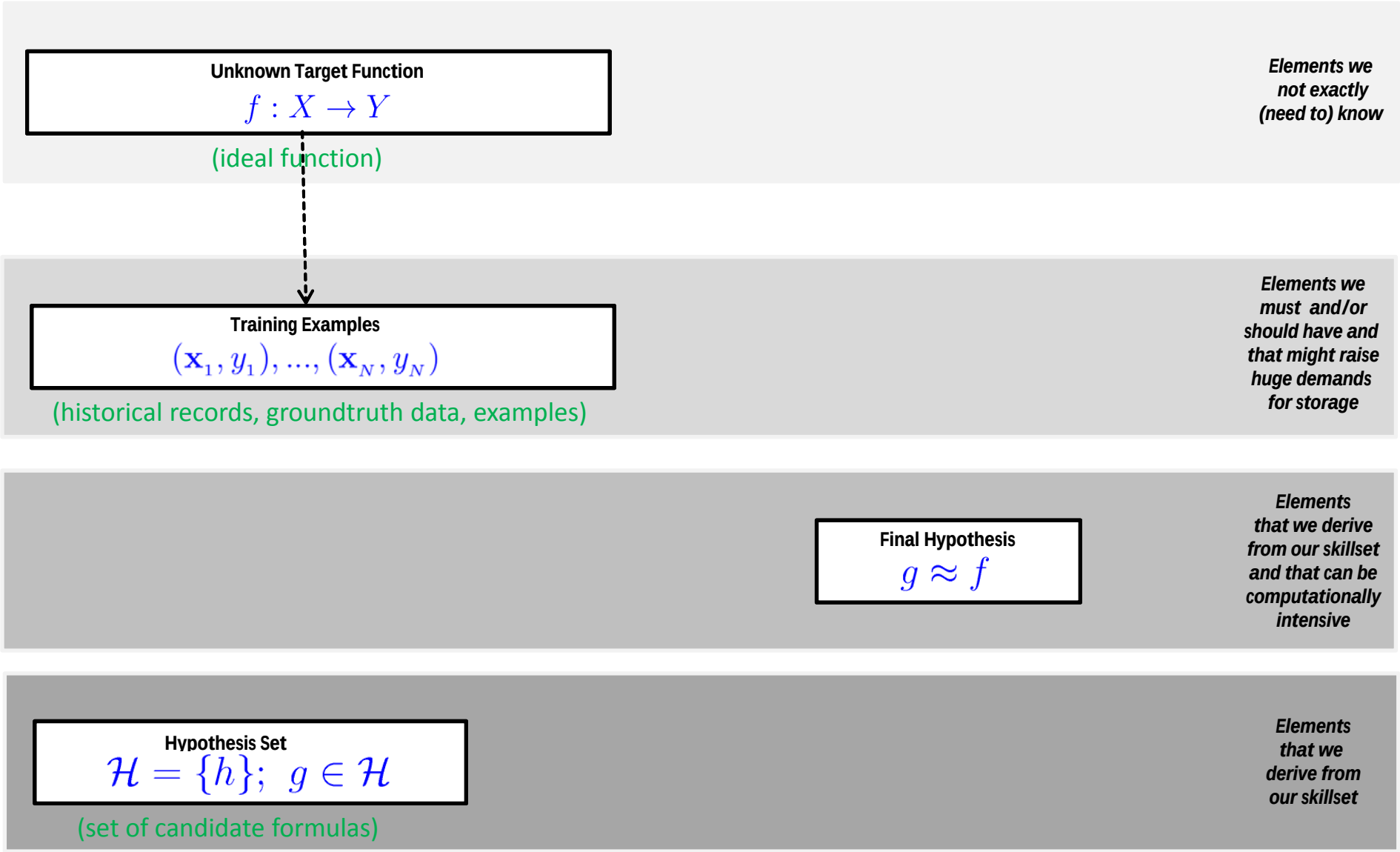
- But ‘(function) approximation‘ of the target function is possible
  - Use training examples to learn and approximate it
  - Hypothesis set $\mathcal{H}$ consists of m different hypothesis (candidate functions)

$$\mathcal{H} = \{h_1, \ldots, h_m\};$$

‘select one function‘ that best approximates $g : X \rightarrow Y$

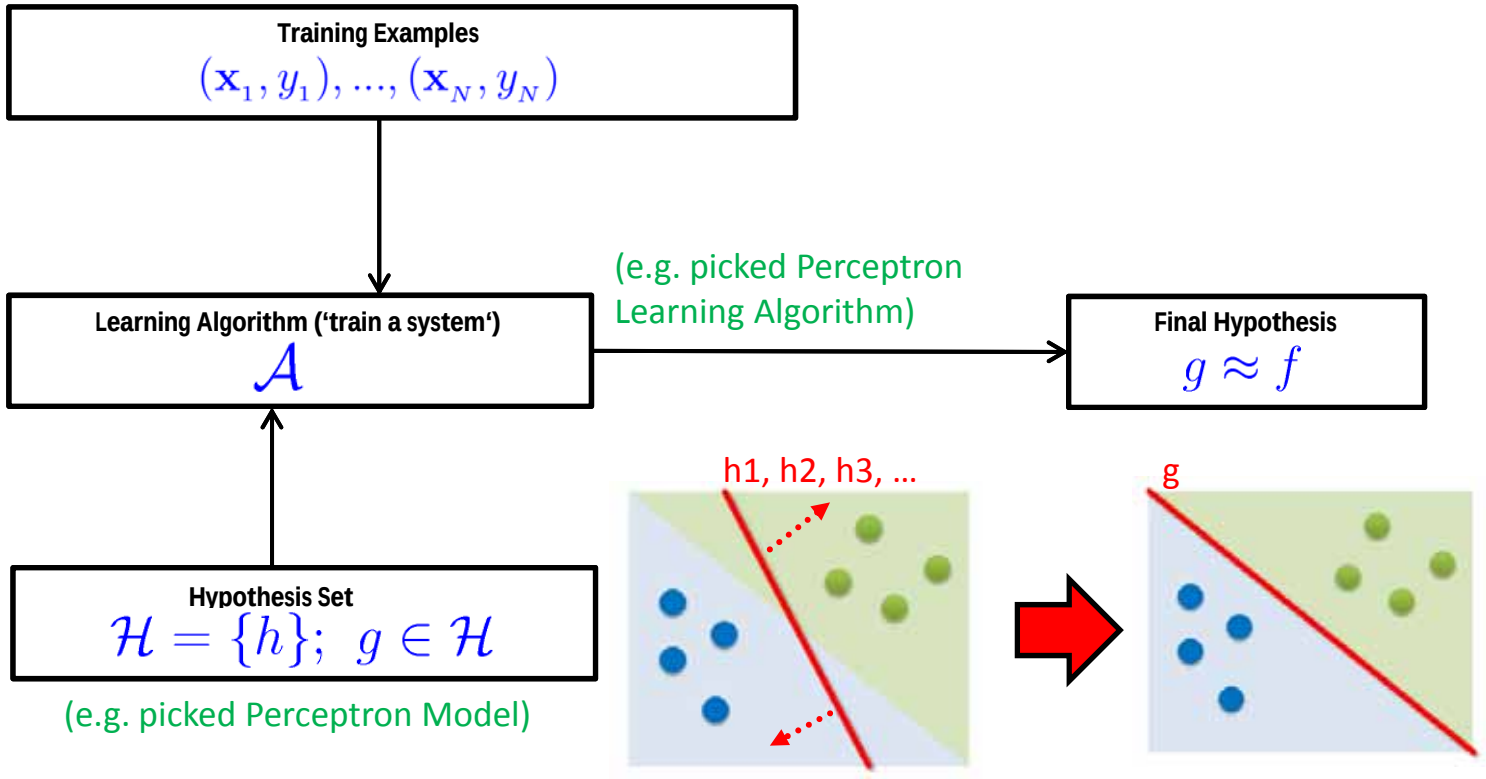


Mathematical Building Blocks (2)

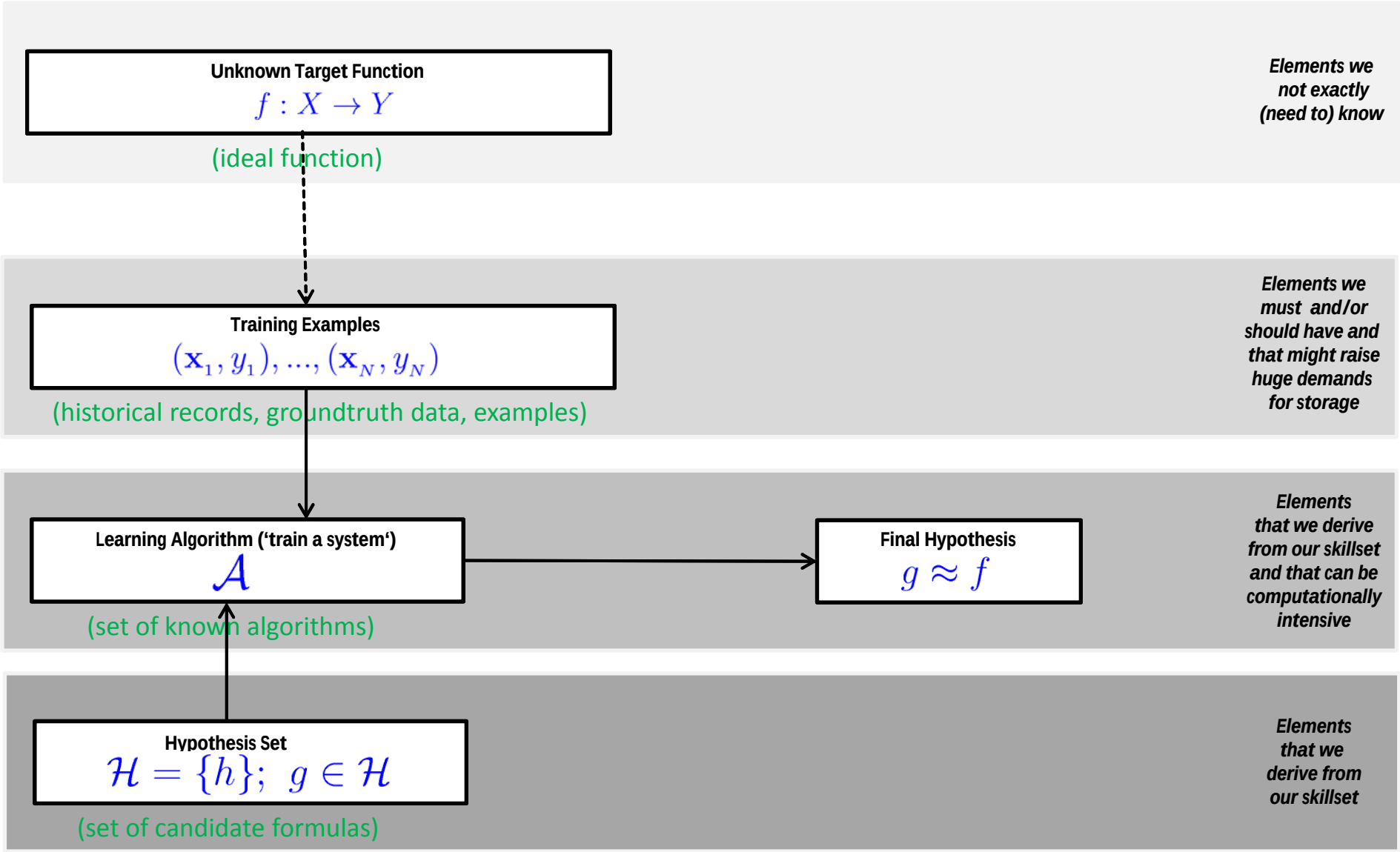


The Learning Model: Hypothesis Set & Learning Algorithm

- The solution tools – the learning model:
  1. Hypothesis set $\mathcal{H}$ - a set of candidate formulas /models
  2. Learning Algorithm $\mathcal{A}$ - 'train a system' with known algorithms

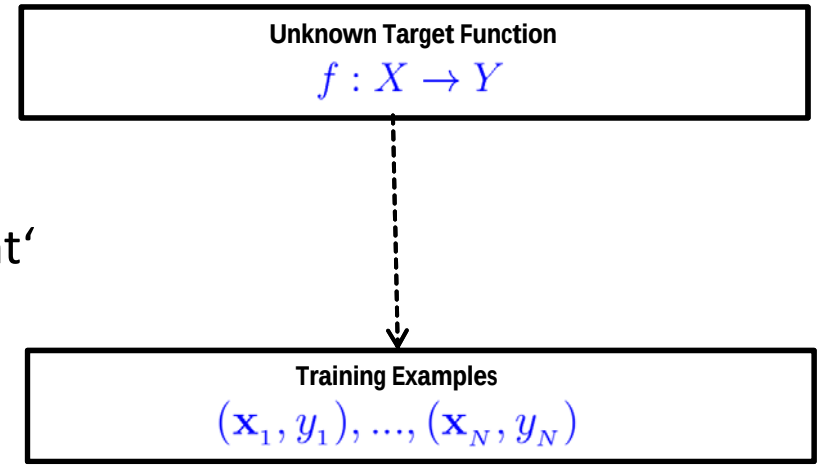


Mathematical Building Blocks (3)

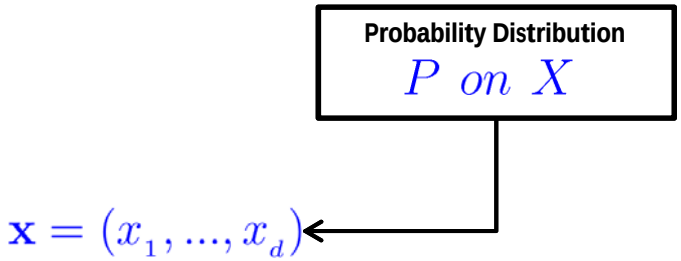


Feasibility of Learning – Probability Distribution

- Predict output from future input
  (fitting existing data is not enough)
  - In-sample ‘1000 points‘ fit well
  - Possible: Out-of-sample >= ‘1001 point‘ doesn‘t fit very well
  - Learning ‘any target function‘ is not feasible (can be anything)



- Assumptions about ‘future input‘
  - Statement is possible to define about the data outside the in-sample data $(\mathbf{x}_1, y_1), ..., (\mathbf{x}_N, y_N)$
  - All samples (also future ones) are derived from same unknown probability distribution $P \ on \ X$

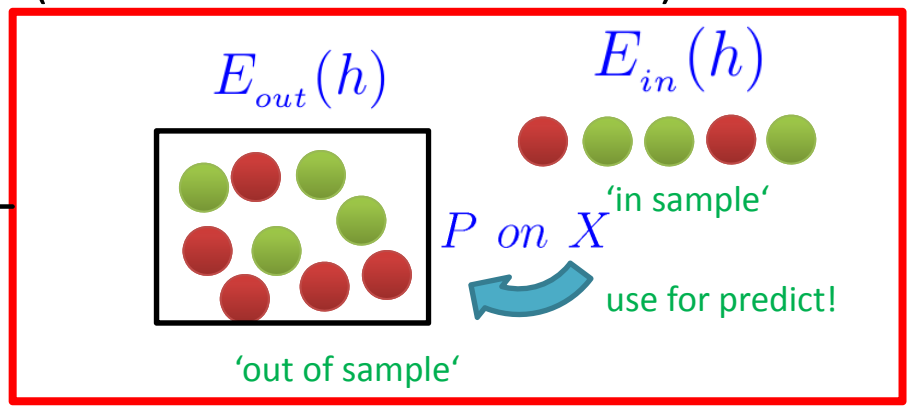


- **Statistical Learning Theory assumes an unknown probability distribution over the input space X**

Feasibility of Learning – In Sample vs. Out of Sample

- Given ‘unknown‘ probability $P \ on \ X$
  - Given large sample N for $(\mathbf{x}_1, y_1), ..., (\mathbf{x}_N, y_N)$
  - There is a probability of ‘picking one point or another‘
  - ‘Error on in sample‘ is known quantity (using labelled data): $E_{in}(h)$
  - ‘Error on out of sample‘ is unknown quantity: $E_{out}(h)$
  - In-sample frequency is likely close to out-of-sample frequency (both are close to each other)

depend on which hypothesis h out of M different ones



$\mathcal{H} = \{h_1, ..., h_m\};$

$$E_{in}(h) \approx E_{out}(h)$$

use $E_{in}(h)$ as a proxy thus the other way around in learning

$$E_{out}(h) \approx E_{in}(h)$$

- **Statistical Learning Theory part that enables that learning is feasible in a probabilistic sense (P on X)**

Feasibility of Learning – Union Bound & Factor M

- **The union bound means that (for any countable set of m 'events') the probability that at least one of the events happens is not greater that the sum of the probabilities of the m individual 'events'**

- Assuming no overlaps in hypothesis set
  - Apply mathematical rule 'union bound'
  - Note the usage of g instead of h

Final Hypothesis

$$g \approx f$$

Think if E_{in} deviates from E_{out} with more than tolerance Є it is a 'bad event' in order to apply union bound

$$\Pr\ [\ |\ E_{in}(g) - E_{out}(g)\ |\ > \epsilon\ ]\ <= \Pr\ [\ |\ E_{in}(h_1) - E_{out}(h_1)\ |\ > \epsilon$$

$$\text{or}\ \ |\ E_{in}(h_2) - E_{out}(h_2)\ |\ > \epsilon\ \ \ldots$$

$$\text{or}\ \ |\ E_{in}(h_M) - E_{out}(h_M)\ |\ > \epsilon\ \ ]$$

'visiting M different hypothesis'

$$\Pr\ [\ |\ E_{in}(g) - E_{out}(g)\ |\ > \epsilon\ ]\ <= \sum_{m=1}^{M} \Pr\ [\ |\ E_{in}(h_m) - E_{out}(h_m)\ |\ > \epsilon\ ]$$

$$\Pr\ [\ |\ E_{in}(g) - E_{out}(g)\ |\ > \epsilon\ ]\ <= \sum_{m=1}^{M} 2e^{-2\epsilon^2 N}$$

fixed quantity for each hypothesis obtained from Hoeffdings Inequality

$$\Pr\ [\ |\ E_{in}(g) - E_{out}(g)\ |\ > \epsilon\ ]\ <= 2Me^{-2\epsilon^2 N}$$

problematic: if M is too big we loose the link between the in-sample and out-of-sample

Feasibility of Learning – Modified Hoeffding's Inequality

- Errors in-sample $E_{in}(g)$ track errors out-of-sample $E_{out}(g)$
  - Statement is made being 'Probably Approximately Correct (PAC)'
  - Given M as number of hypothesis of hypothesis set $\mathcal{H}$
  - 'Tolerance parameter' in learning ϵ
  - Mathematically established via 'modified Hoeffdings Inequality':
    (original Hoeffdings Inequality doesn't apply to multiple hypothesis)

[3] Valiant, 'A Theory of the Learnable', 1984

'Approximately' 'Probably'

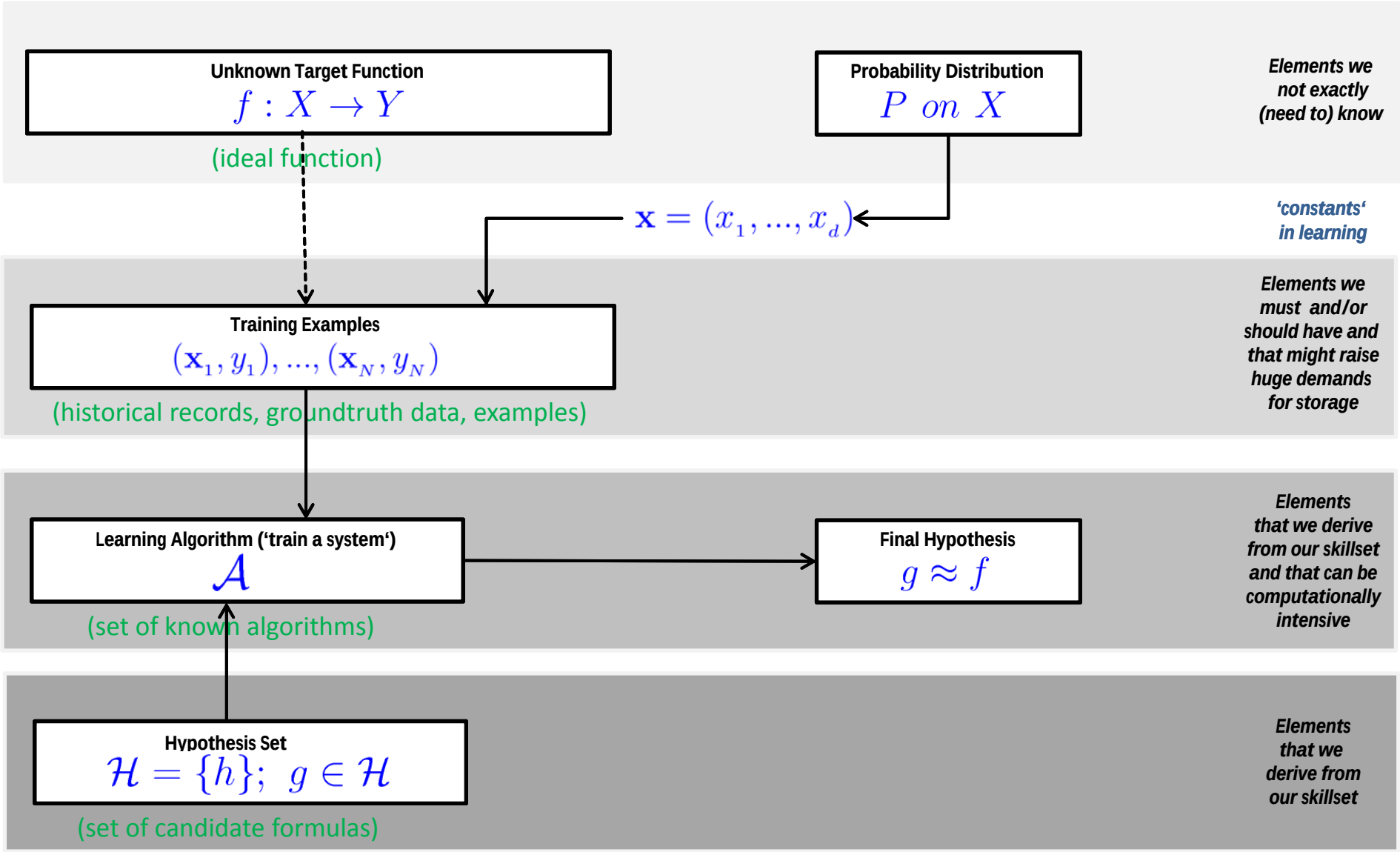
$$\Pr\ [\ |\ E_{in}(g) - E_{out}(g)\ |\ > \epsilon\ ]\ <=\ 2Me^{-2\epsilon^2 N}$$

'Probability that E_{in} deviates from E_{out} by more than the tolerance Є is a small quantity depending on M and N'

- Theoretical 'Big Data' Impact
  - The more samples N the more reliable will track $E_{in}(g)\ E_{out}(g)$ well
  - (But: the 'quality of samples' also matter, not only the number of samples)

- **Statistical Learning Theory part describing the Probably Approximately Correct (PAC) learning**

Mathematical Building Blocks (4)



Statistical Learning Theory – Error Measure & Noisy Targets

- Question: How can we learn a function from (noisy) data?
- ‘Error measures‘ to quantify our progress, the goal is: $h \approx f$
  - Often user-defined, if not often ‘squared error‘:

$$e(h(\mathbf{x}), f(\mathbf{x})) = (h(\mathbf{x}) - f(\mathbf{x}))^2$$

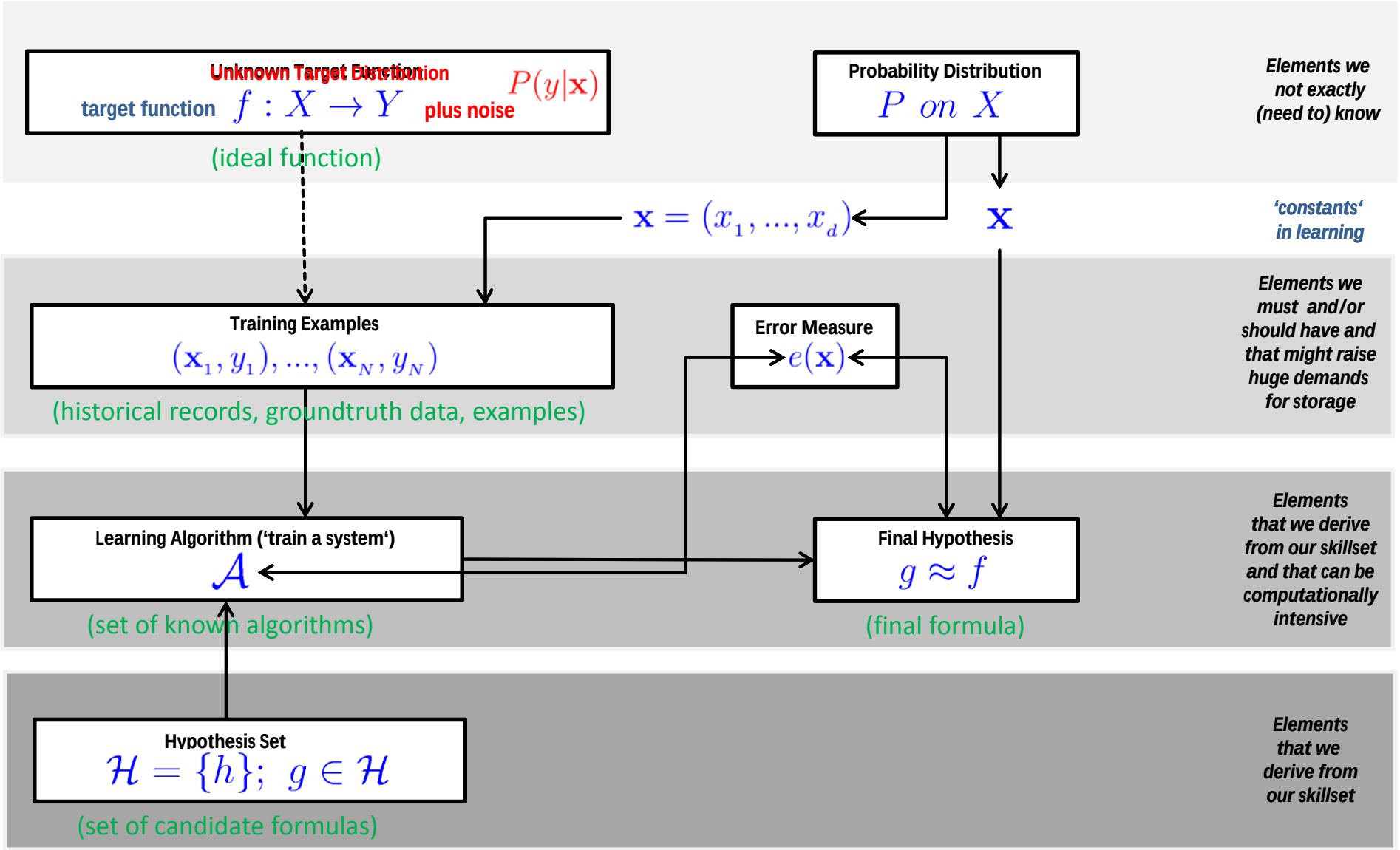
Error Measure
α

  - Aka ‘point-wise error measure‘
- ‘(Noisy) Target function‘ is not a (deterministic) function
  - Getting with ‘same x in‘ the ‘same y out‘ is not always given in practice
  - Problem: ‘Noise‘ in the data that hinders us from learning
  - Idea: Use a ‘target distribution‘ instead of ‘target function‘

Unknown Target Distribution
target function $f : X \to Y$ plus noise $P(y|\mathbf{x})$
(ideal function)

- **Statistical Learning Theory refines the learning problem of learning an unknown target distribution**

Mathematical Building Blocks (5)



Theory of Generalization – Learning Process Summary

- ‘Learning Well‘
  - Two core building blocks that achieve $E_{out}(g)$ approximates 0
- First core building block
  - Theoretical result using Hoeffdings Inequality $E_{out}(g) \approx E_{in}(g)$
  - Using $E_{out}(g)$ directly is not possible – it is an unknown quantity
- Second core building block
  - Practical result using tools & techniques to get $E_{in}(g) \approx 0$
  - (e.g. linear models like Perceptron using perceptron learning algorithm)
  - Using $E_{in}(g)$ is possible – it is a known quantity
  - Lessons learned from practice: in many situations ‘close to 0‘ impossible
  - But M = infinity number of potential Hypothesis → How can we learn?

- **Full learning means that we can make sure that $E_{out}(g)$ is close enough to $E_{in}(g)$ [from theory]**
- **Full learning means that we can make sure that $E_{in}(g)$ is small enough [from practical techniques]**

Factor M from the Union Bound & Hypothesis Overlaps

$$\Pr\ [\ |\ E_{in}(g) - E_{out}(g)\ |\ > \epsilon\ ]\ <= \Pr\ [\ |\ E_{in}(h_1) - E_{out}(h_1)\ |\ > \epsilon$$
$$\text{or}\ \ |\ E_{in}(h_2) - E_{out}(h_2)\ |\ > \epsilon\ \ \ldots$$
$$\text{or}\ \ |\ E_{in}(h_M) - E_{out}(h_M)\ |\ > \epsilon\ \ ]$$

assumes no overlaps, all probabilities happen disjointly

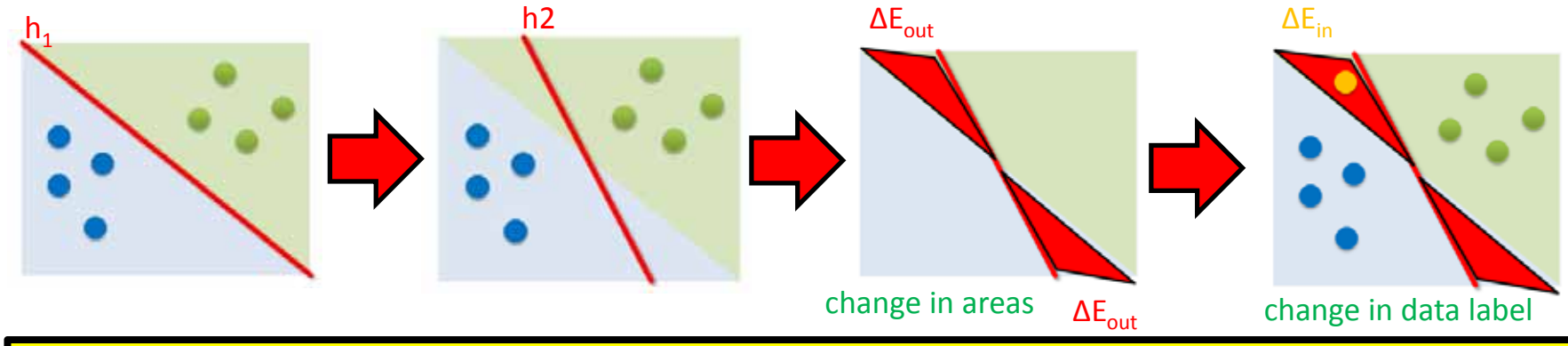
$$\Pr\ [\ |\ E_{in}(g) - E_{out}(g)\ |\ > \epsilon\ ]\ <=\ 2Me^{-2\epsilon^2 N}$$

takes no overlaps of M hypothesis into account

- Union bound is a 'poor bound', ignores correlation between h
  - Overlaps are common: the interest is shifted to data points changing label

$$|\ E_{in}(h_1) - E_{out}(h_1)\ | \approx\ \ |\ E_{in}(h_2) - E_{out}(h_2)\ |$$

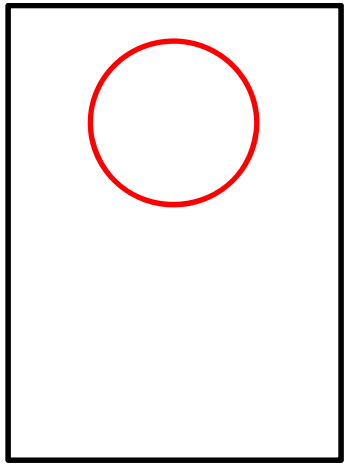
(at least very often, indicator to reduce M)



- **Statistical Learning Theory provides a quantity able to characterize the overlaps for a better bound**

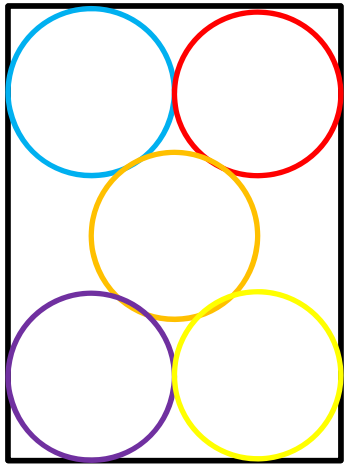
Replacing M & Large Overlaps Summary

(Hoeffding Inequality)



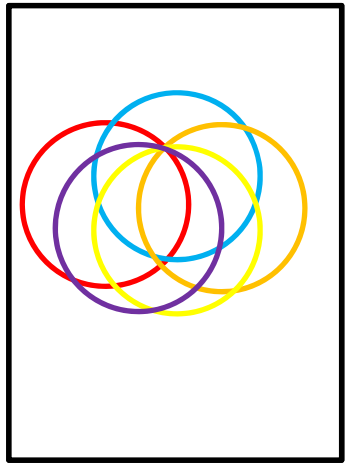
(valid for 1 hypothesis)

(Union Bound)



(valid for M hypothesis, worst case)

(towards Vapnik Chervonenkis Bound)



(valid for m (N) as growth function)

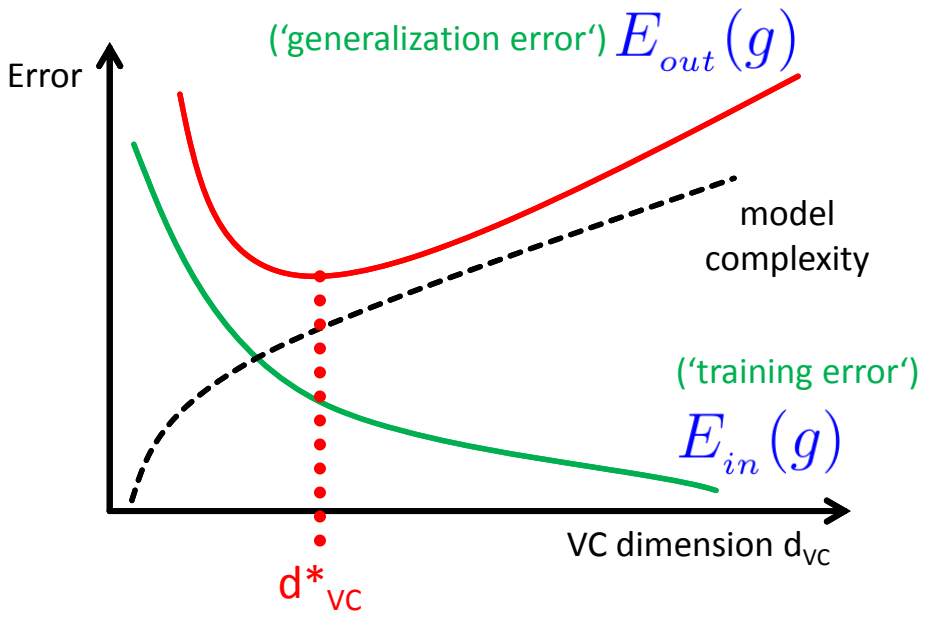
- Characterizing the overlaps was the idea of the growth function
  - Number of dichotomies: $\mathbf{m}_{\mathcal{H}}(N) = \max_{\mathbf{x}_1,\mathbf{x}_2,\ldots,\mathbf{x}_N} |\mathcal{H}(\mathbf{x}_1,\mathbf{x}_2,\ldots,\mathbf{x}_N)|$
    Number of hypothesis but
    on finite number N of points
  - Much redundancy: Many hypothesis will report the same dichotomies

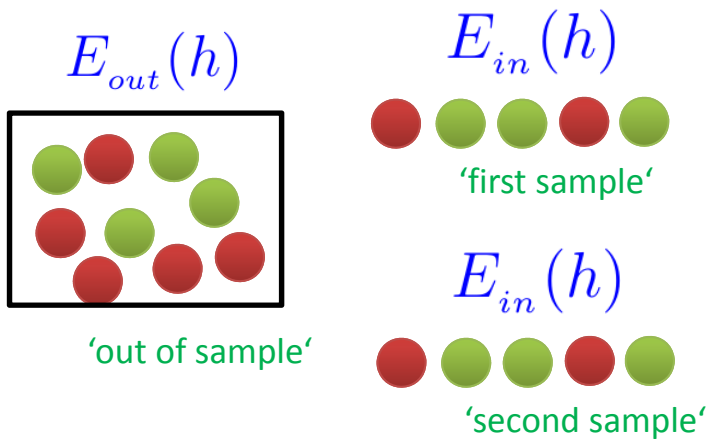
- **The mathematical proofs that $m_H(N)$ can replace M is a key part of the theory of generalization**

Complexity of the Hypothesis Set – VC Dimension (1)

- Vapnik-Chervonenkis (VC) Dimension over instance space X
  - VC dimension gets a 'generalization bound' on all possible target functions

Issue: unknown to 'compute' – VC solved this using the growth function on different samples





idea: 'first sample' frequency close to 'second sample' frequency

- **Complexity of Hypothesis set H can be measured by the Vapnik-Chervonenkis (VC) Dimension d_{VC}**
- **Ignoring the model complexity d_{VC} leads to situations where $E_{in}(g)$ gets down and $E_{out}(g)$ gets up**

Complexity of the Hypothesis Set – VC Dimension (2)

$$\Pr\ [\ |\ E_{in}(g) - E_{out}(g)\ |\ >\ \epsilon\ ]\ <=\ 2Me^{-2\epsilon^2 N}$$

$$\mathbf{m}_{\mathcal{H}}(N) = \max_{\mathbf{x}_1,\mathbf{x}_2,\ldots,\mathbf{x}_N} |\mathcal{H}(\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N)|$$

- Vapnik-Chervonenkis (VC) Inequality
  - Result of mathematical proof when replacing M with growth function m
  - 2N of growth function to have another sample (2 x $E_{in}(h)$, no $E_{out}(h)$)

$$\Pr\ [\ |\ E_{in}(g) - E_{out}(g)\ |\ >\ \epsilon\ ]\ <=\ 4m_{\mathcal{H}}(2N)e^{-1/8\epsilon^2 N}$$

(characterization of generalization)

- But 'VC Dimension' is the main notion in learning
  - Related to work about breakpoints, already computed for many models
  - Used in practical situations (outside the theory): VC dimension 3 vs. 500

- **The Vapnik-Chernonenkis Inequality is the most important result in machine learning theory**
- **The mathematial proof brings us that M can be replaced by growth function**

Complexity of the Hypothesis Set – VC Dimension (3)

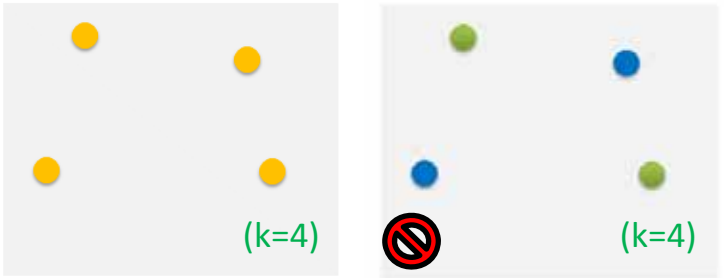
- **Important from all the theory: VC dimension of H is the largest value of N for which $m_H(N) = 2^N$**

- VC(H) is the size of largest finite subset of space X shattered by H
  - The most points H is able to shatter (cf. breakpoint – 1)
  - Strong statement possible: If VC(H) is finite → g will generalize
  - Independent of learning algorithm, input distribution or target function
- Example: 2D Perceptron model
  - VC dimension = 3 regardless of which exact hypothesis h

Final Hypothesis

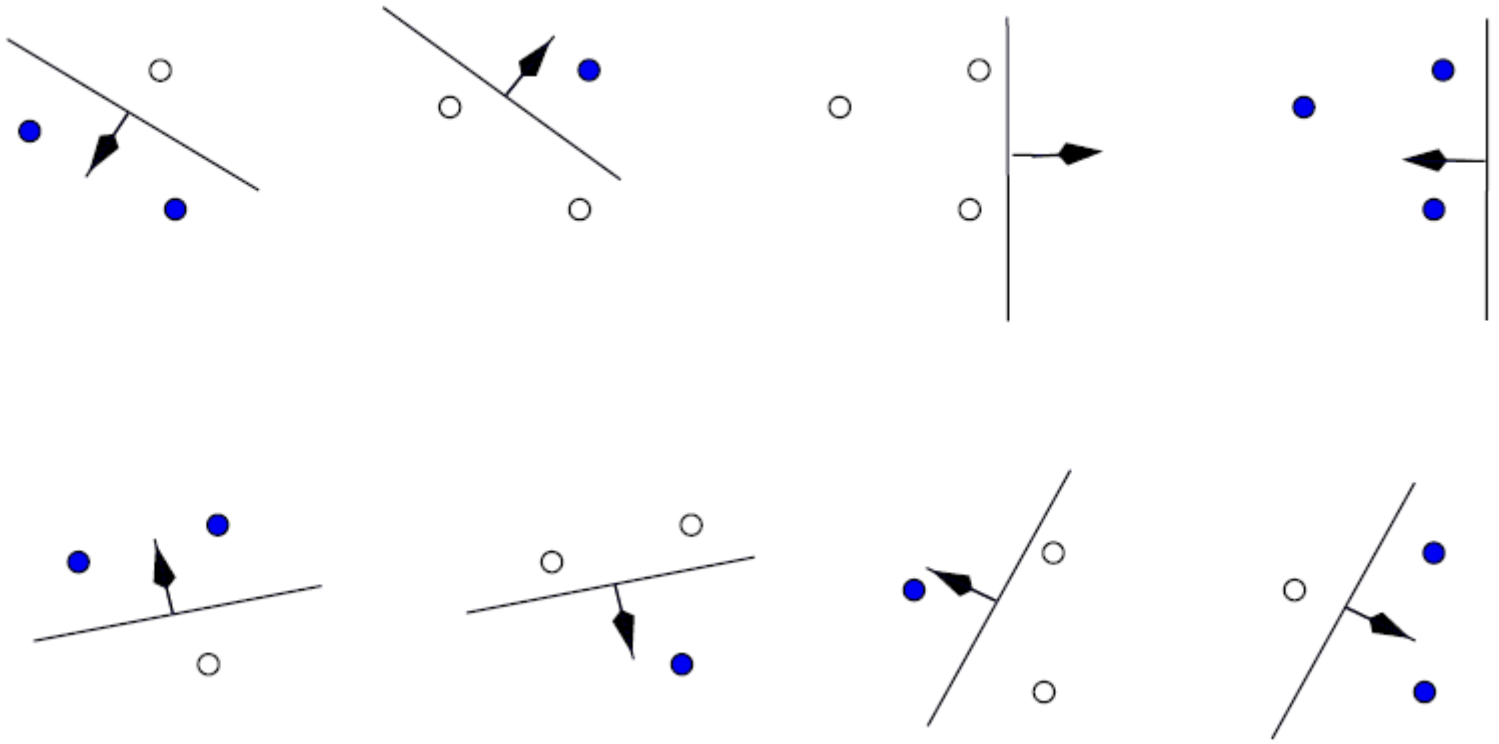
$$g \approx f$$

(characterization of generalization: VC dimension = 3)



(breakpoint k = 4)

Complexity of the Hypothesis Set – VC Dimension Example

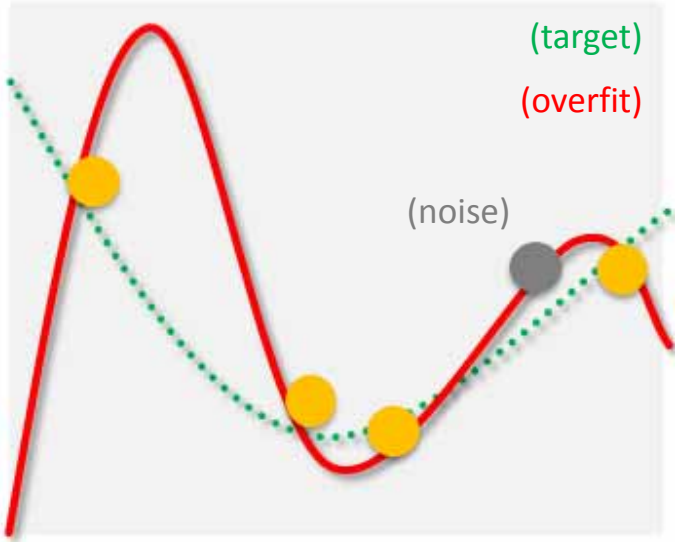


[4] C. Burges, 1998

Problem of Overfitting – Motivational Example

- **Overfitting refers to fit the data too well – more than is warranted – thus may misguide the learning**
- **Overfitting is not just 'bad generalization' - e.g. the VC dimension covers noiseless & noise targets**
- **Theory of Regularization are approaches against overfitting and prevent it using different methods**

- Key problem: noise in the target function leads to overfitting
  - Effect: 'noisy target function' and its noise misguides the fit in learning
  - There is always 'some noise' in the data
  - Consequence: poor target function ('distribution') approximation
- Example: Target functions is second order polynomial (i.e. parabola)
  - Using a higher-order polynomial fit
  - Perfect fit: low $E_{in}(g)$, but large $E_{out}(g)$



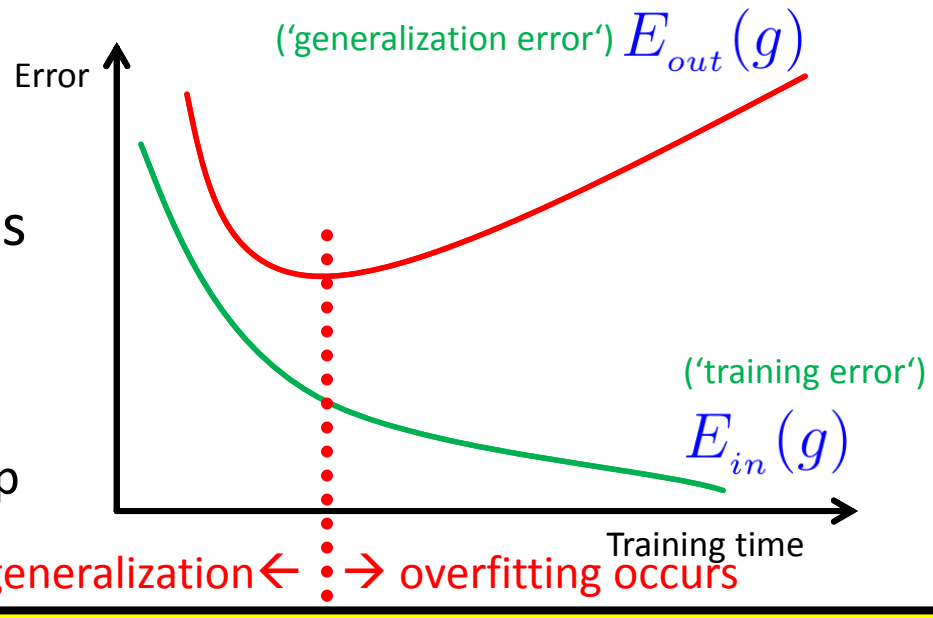
(but simple polynomial works good enough)
('over': here meant as 4th order, a 3rd order would be better, 2nd best)

Problem of Overfitting – Clarifying Terms

- **A good model must have low training error (E_{in}) and low generalization error (E_{out})**
- **Model overfitting is if a model fits the data too well (E_{in}) with a poorer generalization error (E_{out}) than another model with a higher training error (E_{in})**

[4] Introduction to Data Mining

- Overfitting & Errors
  - $E_{in}(g)$ goes down
  - $E_{out}(g)$ goes up
- 'Bad generalization area' ends
  - Good to reduce $E_{in}(g)$
- 'Overfitting area' starts
  - Reducing $E_{in}(g)$ does not help
  - Reason 'fitting the noise'



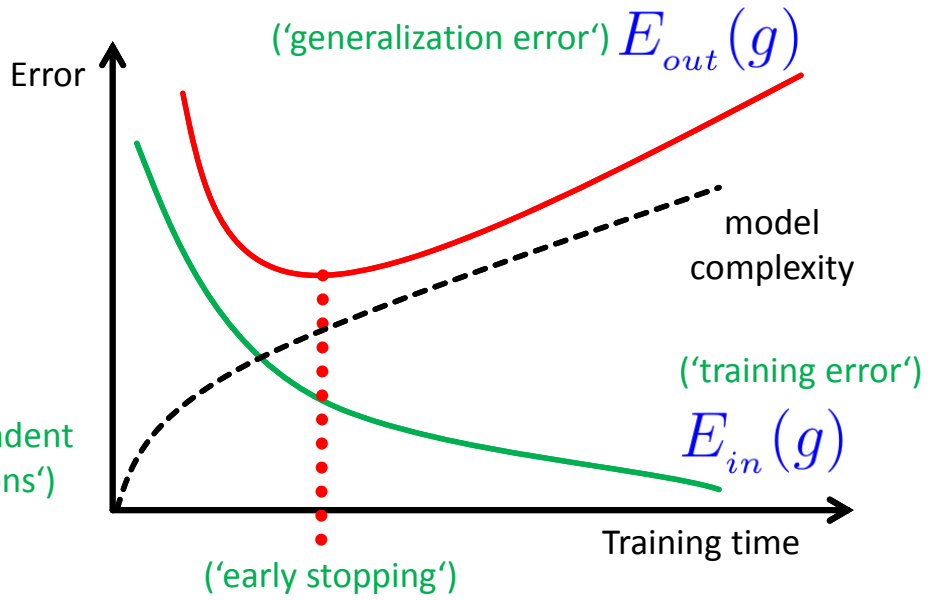
- **The two general approaches to prevent overfitting are (1) regularization and (2) validation**

➢ **Lecture on classification includes aspects of validation as another method against overfitting**

Problem of Overfitting – Model Relationships

- Review ‘overfitting situations‘
  - When comparing ‘various models‘ and related to ‘model complexity‘
  - Different models are used, e.g. 2nd and 4th order polynomial
  - Same model is used with e.g. two different instances (e.g. two neural networks but with different parameters)
- Intuitive solution
  - Detect when it happens
  - ‘Early stopping regularization term‘ to stop the training
  - Early stopping method (later)

(‘model complexity measure: the VC analysis was independent of a specific target function – bound for all target functions‘)



- **‘Early stopping‘ approach is part of the theory of regularization, but based on validation methods**

Theory of Regularization – Key Principle Summary

- Initial setup
  - Considered as ‘free fit‘ – ‘fit as far as the model can do‘
  - E.g. use 4th order polynomial model and fit the 5 data points (cf. Figure A)
- ‘Putting the brakes‘ regularization to avoid overfitting
  - Apply a ‘restrained fit‘ – ‘preventing to fit the data perfectly‘
  - E.g. use 4th order polynomial model but use ‘minimal brakes‘ (cf. Figure B)

(equivalent meaning of explicitly forbidding some of the hypothesis to be considered in learning)

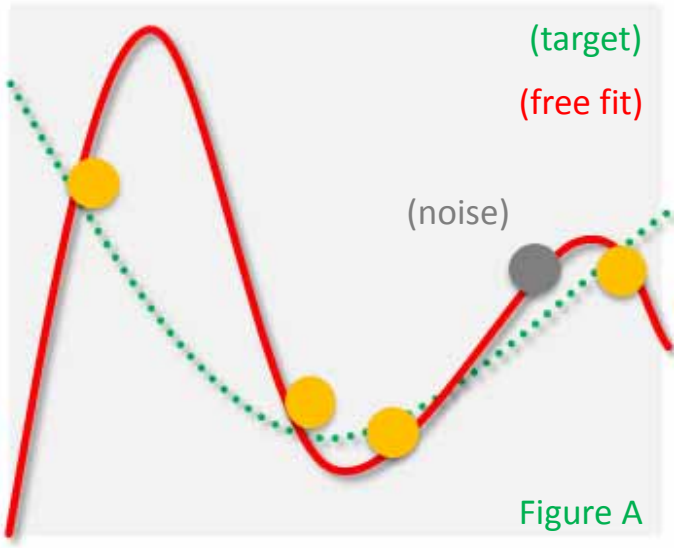


Figure A

(not getting all the points right, but much better fit)

‘apply regularization’

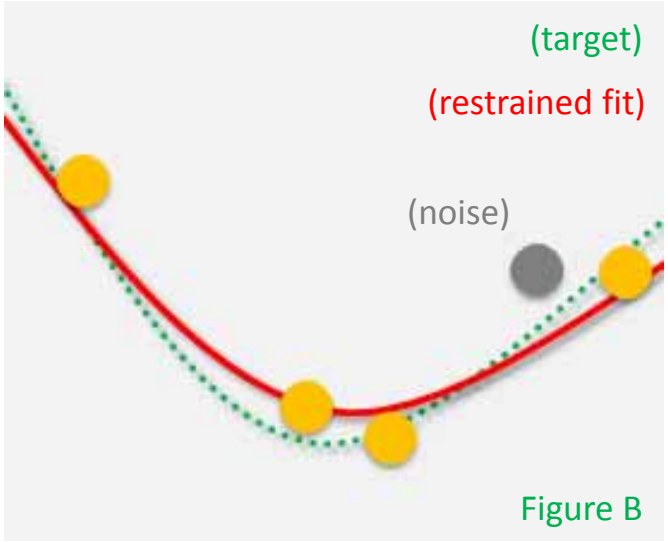
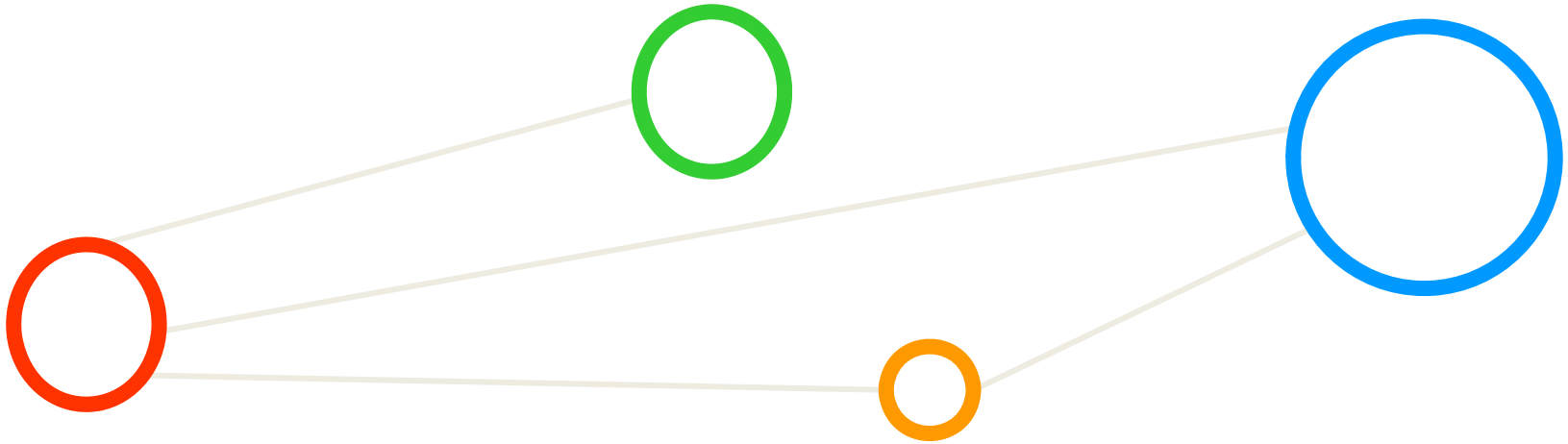


Figure B

Bibliography



Bibliography

- [1] An Introduction to Statistical Learning with Applications in R, Online: http://www-bcf.usc.edu/~gareth/ISL/index.html
- [2] Wikipedia on 'Statistical Learning Theory', Online: http://en.wikipedia.org/wiki/Statistical_learning_theory
- [3] Leslie G. Valiant, *'A Theory of the Learnable'*, Communications of the ACM 27(11):1134–1142, 1984, Online: https://people.mpi-inf.mpg.de/~mehlhorn/SeminarEvolvability/ValiantLearnable.pdf
- [4] C. Burges, *'A Tutorial on Support Vector Machines for Pattern Recognition'*, Data Mining and Knowledge Discovery 2, 121-167, 1998

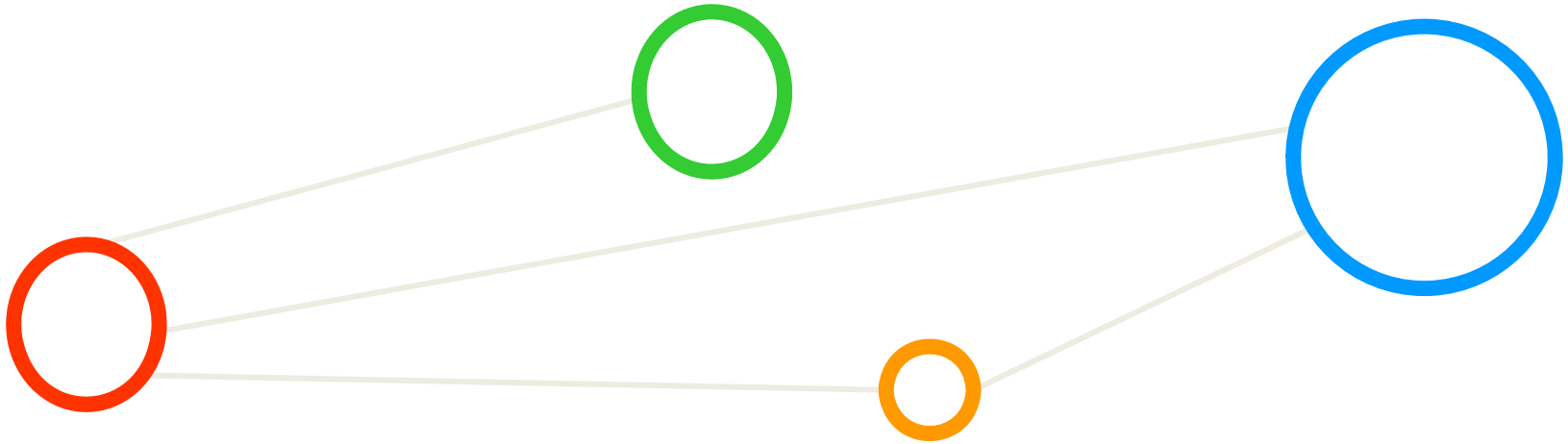
Acknowledgements and more Information

- Yaser Abu-Mostafa, Caltech Lecture series, YouTube
- Andrew Ng, Standford Lecture series, YouTube

Thanks

Slides available at http://www.morrisriedel.de/talks

Backup Slides



Problem of Overfitting – Noise Term Revisited

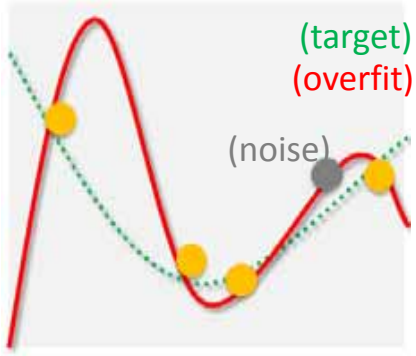
- ‘(Noisy) Target function‘ is not a (deterministic) function
  - Getting with ‘same x in‘ the ‘same y out‘ is not always given in practice
  - Idea: Use a ‘target distribution‘ instead of ‘target function‘

Unknown Target Distribution

target function $f : X \rightarrow Y$ plus noise $P(y|\mathbf{x})$

(ideal function)

- **Fitting some noise in the data is the basic reason for overfitting and harms the learning process**
- **Big datasets tend to have more noise in the data so the overfitting problem might occur even more intense**



(‘function view‘)

‘shift the view’

(‘# data view‘)

‘# samples‘

- ‘Different types of some noise‘ in data
  - Key to understand overfitting & preventing it
  - ‘Shift of view‘: refinement of noise term
  - Learning from data: ‘matching properties of # data‘

Problem of Overfitting – Stochastic Noise

- Stoachastic noise is a part ‘on top of‘ each learnable function
  - Noise in the data that can not be captured and thus not modelled by f
  - Random noise : aka ‘non-deterministic noise‘
  - Conventional understanding established early in this course
  - Finding a ‘non-existing pattern in noise not feasible in learning‘

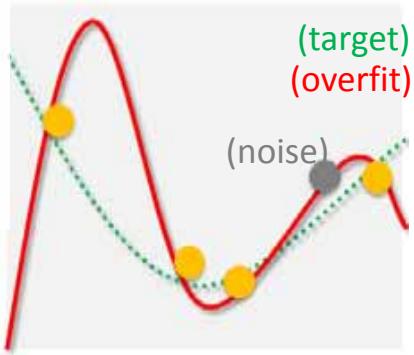
Unknown Target Distribution

target function $f : X \rightarrow Y$ plus noise $P(y|\mathbf{x})$

(ideal function)

- Practice Example
  - Random fluctuations and/or measurement errors in data
  - Fitting a pattern that not exists ‘out-of-sample‘

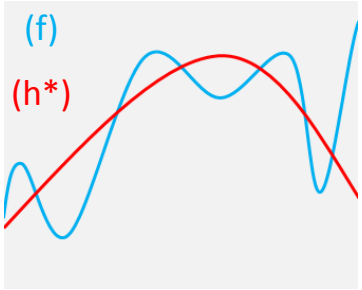
    (puts learning progress off-track and away from f)



- **Stochastic noise here means noise that can‘t be captured, because it‘s just pure ‘noise as is‘ (nothing to look for) – aka no pattern in the data to understand or to learn from**

Problem of Overfitting – Deterministic Noise

- Part of target function f that H can not capture: $f(\mathbf{x}) - h^*(\mathbf{x})$
  - Hypothesis set H is limited so best h* can not fully approximate f
  - h* approximates f, but fails to pick certain parts of the target f
  - 'Behaves like noise', existing even if data is 'stochastic noiseless'
- Different 'type of noise' than stochastic noise
  - Deterministic noise depends on $\mathcal{H}$ (determines how much more can be captured by h*)
  - E.g. same f, and more sophisticated $\mathcal{H}$: noise is smaller (stochastic noise remains the same, nothing can capture it)
  - Fixed for a given $\mathbf{x}$, clearly measurable (stochastic noise may vary for values of $\mathbf{x}$)



(learning deterministic noise is outside the ability to learn for a given h*)

- **Deterministic noise here means noise that can't be captured, because it is a limited model (out of the league) – aka 'learning with a toddler statistical learning'**

Problem of Overfitting – Impacts on Learning

- **The higher the degree of the polynomial (cf. model complexity), the more degrees of freedom are existing and thus the more capacity exists to overfit the training data**

- Understanding deterministic noise & target complexity
  - Increasing target complexity increases deterministic noise (at some level)
  - Increasing the number of data N decreases the deterministic noise
- Finite N case: $\mathcal{H}$ tries to fit the noise
  - Fitting the noise straightforward (e.g. linear regression, cf. Bodenstein)
  - Stochastic (in data) and deterministic (simple model) noise will be part of it
- Two 'solution methods' for avoiding overfitting
  - Regularization: 'Putting the brakes in learning', e.g. early stopping (more theoretical, hence 'theory of regularization')
  - Validation: 'Checking the bottom line', e.g. other hints for out-of-sample (more practical, methods on data that provides 'hints')

➢ **Lecture on classification includes aspects of validation as another method against overfitting**