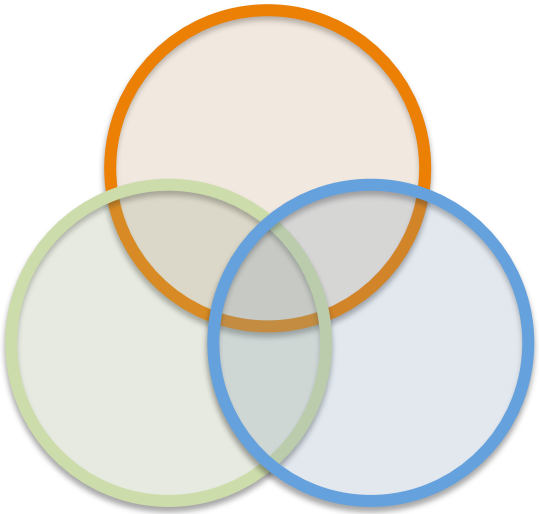
Machine Learning for the Quantified Self



Chapter 6

Mathematical Foundations of Supervised Learning

Overview

- Previously: we performed clustering
- Today: let us turn supervised
  - Fundamentals
  - Learning setup
  - Brief overview of learning algorithms
  - Our CrowdSignals case study

Supervised learning (1)

- What do we mean when we say ”machines can learn”?
  - Task
  - Historical data
  - Improvement on task performance
- Mitchell: “*A computer program is said to learn from experience E with respect to some class of tasks T and performance P, if its performance at tasks in T improves with E.*”

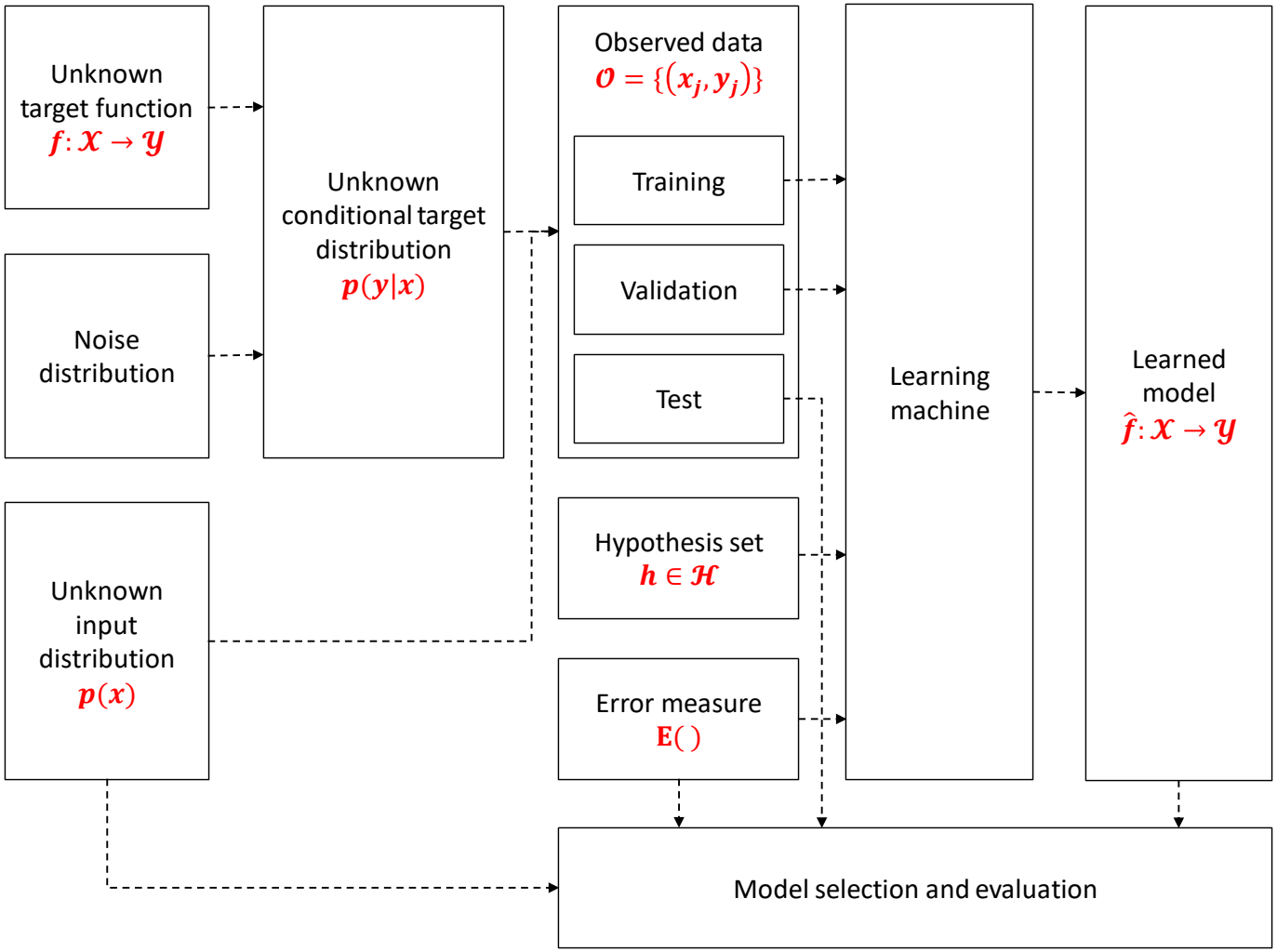
Supervised learning (2)

- In supervised learning we want to learn
  - a *functional relationship,* alternatively called an unknown target function $f : \mathcal{X} \rightarrow \mathcal{Y}$
- Here we map an observation $x \in \mathcal{X}$ to the target $y \in \mathcal{Y}$ (or *g)*
- Here, $\mathcal{X}$ represents the space of all possible inputs (and $\mathcal{Y}$ all possible targets)
- Not likely to be deterministic
  - Measurement errors
  - *Noisy target* (e.g. because we do not have all observations)

Supervised learning (3)

- Assume unknown conditional target distribution $p(y|x)$
- Gives probability of y given that x is fixed
- If we calculate $f(x)$ we add noise from a *noise distribution*
  - Bernoulli or categorical distribution for discrete target
  - Normal distribution for continuous
- More difficult to learn the target function with this noise
- In addition we have the distribution of the observations $p(x)$

Supervised learning (4)



Observed Data

- We separate our dataset into a training, validation, and test set
- We learn a function $\hat{f}(x)$ that fits our observed data in the training set
- Should stratify training set in case of unbalanced dataset
- Evaluate generalizability of $\hat{f}(x)$ upon test set
- Stop learning process based on validation set
- Small dataset: cross validation

Error measure (1)

- Assume we have a hypothesis for the target function *h*
- How far apart is *h* from *f* ? *E(f, h)*
  - *risk*
- We can compute it per point *e(f(x), h(x))*
  - loss
- From e to E (given *p(x)*):

$$E(f,h) = \int_{\mathcal{X}} e(f(x),h(x))p(x)dx$$

Error measure (2)

- Could we computer this?
  - Nope
- We approximate it using the data we have

$$E(f,h) \approx \frac{1}{N}\sum_{j=1}^{N} e(y_j, h(x_j))$$

- What definitions do we have for *e*?
  - Classification example: classification rate, F1, AUC, …..
  - Regression: mean squared error, mean absolute error, …..

Error measure (3)

- In sample error:

$$E_{in}(h) = \frac{1}{N} \sum_{(x,y) \in \mathcal{O}_{Train}} e(y, h(x))$$

- Out of sample error:

$$E_{out}(h) = \int_{\mathcal{X} \setminus \mathcal{O}_{Train}} e(f(x), h(x)) p(x) dx$$

Hypotheses

- We select the hypothesis that minimizes the in-sample error:

$$\hat{f} = \underset{h \in \mathcal{H}}{\operatorname{argmin}} E_{in}(h)$$

- What is the set of hypotheses?
  - For linear regression:

$$h_\theta(x_j) = \theta_0 + \sum_{k=1}^{p} \theta_k x_j^k = \theta^T x_j$$

Model selection

- We select the hypothesis that has the lowest in-sample error on the validation set
- We should be careful about overfitting and not use too many features

Learning theory

- Are we able to learn the perfect model (*f*)?
  - Even with infinite training examples the answer is often “no”
  - Why? Because we cannot guarantee that $f \in \mathcal{H}$
- Despite this, learning is possible if the in-sample error is a good estimator of the out-sample error $|E_{out}(\hat{f}) - E_{in}(\hat{f})|$
- Difference with higher *N* is *most likely* very small

PAC Learnable (1)

- We can make this a bit more formal:

Definition 6.2. A hypothesis set is said to be *PAC learnable* (probably approximately correct), if a learning algorithm exists that fulfills the following condition: For every $\varepsilon > 0$ and $\delta \in (0,1)$ there is an $m \in \mathbb{N}$, so that for a random training sample with length larger than m, the following inequality holds with probability $1 - \delta$: $|E_{out}(\hat{f}) - E_{in}(\hat{f})| < \varepsilon$

- Assume we have *M* hypotheses, then:

$$E_{out}(\hat{f}) \leq E_{in}(\hat{f}) + \sqrt{\frac{1}{2N} \log \frac{2M}{\delta}}$$

- Hence, every finite set of hypotheses is PAC learnable

PAC Learnable (2)

- What if we have an infinite set of hypotheses?
- Let us focus on a binary classification problem and assume we have a function *h* that is applied to all *N* instances in our set:

Definition 6.3. Let the hypothesis set $\mathcal{H}$ be a set of functions $h : \mathcal{X} \to \{0,1\}$. For a set of input vectors $X = \{x_1, x_2, \cdots, x_N\}$, we call $\mathcal{H}_X = \{(h(x_1), \cdots, h(x_N)) : h \in \mathcal{H}\}$ a *restriction of* $\mathcal{H}$ *on* X.

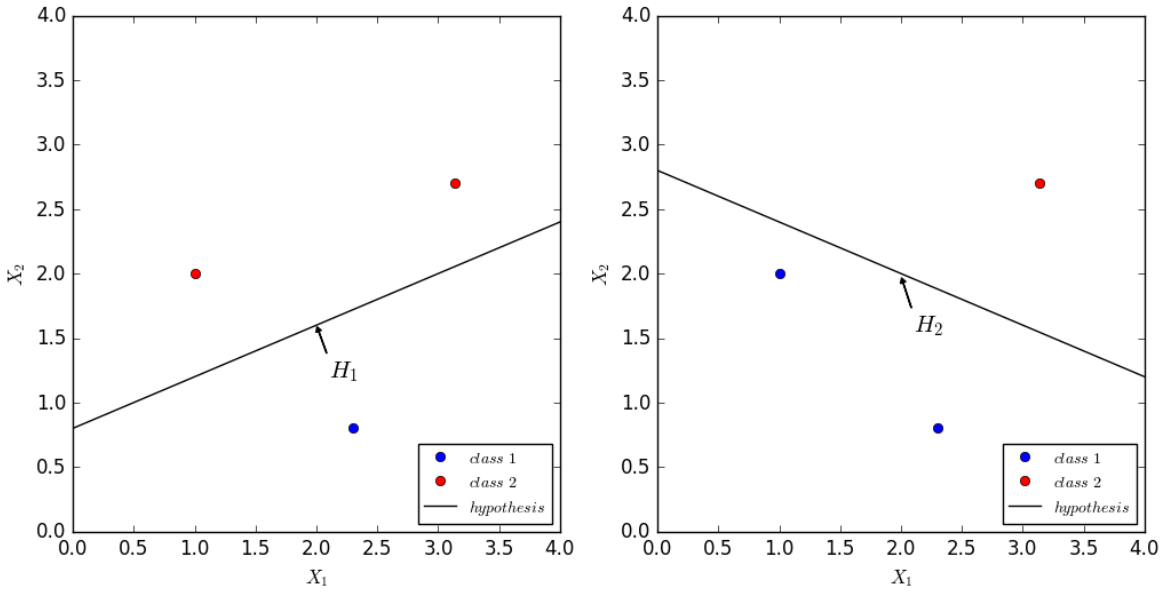
VC Dimension (1)

- How many hypotheses do we have?
  - 2^N
- A hypothesis set $\mathcal{H}$ shatters X when it can represent every possible labeling (i.e. the 2^N)
- We then define:

Definition 6.4. The Vapnik-Chervonenkis (VC) dimension d_{VC} of a hypothesis set $\mathcal{H}$ is the maximum number of input vectors that can be shattered. The VC dimension is infinite if there are arbitrarily large sets of input vectors that can be shattered.

VC Dimension (2)

- We need to show this for an example, e.g. two real-valued attributes and the set of hypotheses involves a line
  - Can we represent all combinations for three points?



VC Dimension (3)

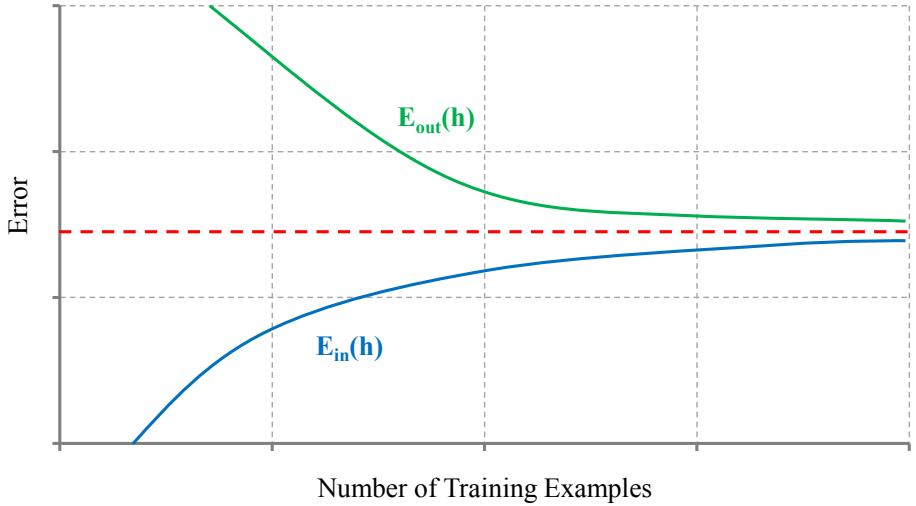
- Can we do it for 4?
  - Nope, VC-dimension is 3
- Now a nice result: all hypothesis sets with finite VC-dimension are PAC learnable
- In addition we can shown that:

$$E_{out}(\hat{f}) \leq E_{in}(\hat{f}) + \sqrt{\frac{8}{N}\log\frac{4m_{\mathcal{H}}(2N)}{N}}$$

- With $m_{\mathcal{H}}(N)$ being the growth function
  - Measures the maximum number of elements for all possible restrictions $\mathcal{H}_X$

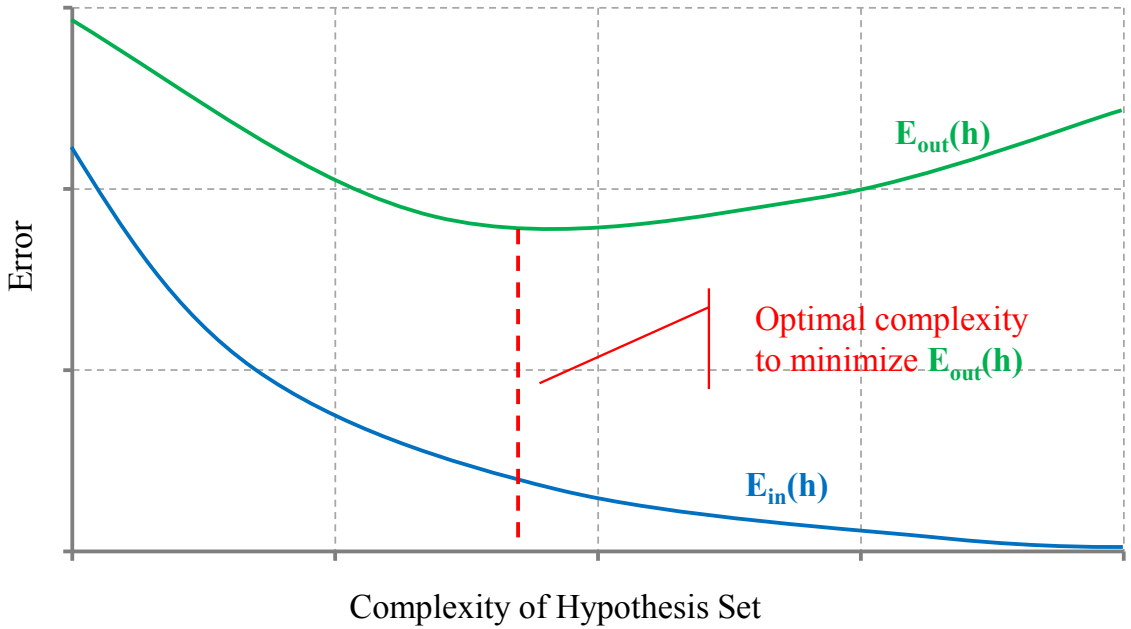
Implications (1)

- What are the implications?
  - With few training examples, it is easy to obtain a low in-sample error
  - As *N* increases we will converge to a maximum
  - The out of sample error will decrease



Implications (2)

- Given a fixed *N* we can study the influence of more complex hypotheses



Implications (3)

- Complexity versus training examples: