Learning Theory

Questions suitable for Theoretical Analysis

Questions suitable for Theoretical Analysis

- How hard is a given learning task?

Questions suitable for Theoretical Analysis

- How hard is a given learning task?
- How many training examples are needed?

Questions suitable for Theoretical Analysis

- How hard is a given learning task?
- How many training examples are needed?
- How many errors should we expect during and after training?

Questions suitable for Theoretical Analysis

- How hard is a given learning task?
- How many training examples are needed?
- How many errors should we expect during and after training?
- How large is the risk of failing to learn?

What might Fail?

Assumptions:

- Concept Learning
- Training and test data from same distribution $\mathcal{D}$

Assumptions:

- Concept Learning
- Training and test data from same distribution $\mathcal{D}$

What kind of errors can occur?

Assumptions:

- Concept Learning
- Training and test data from same distribution $\mathcal{D}$

What kind of errors can occur?

- The result of leaning can be bad
  The resulting hypothesis makes too many errors

Assumptions:

- Concept Learning
- Training and test data from same distribution $\mathcal{D}$

What kind of errors can occur?

- The result of leaning can be bad
  The resulting hypothesis makes too many errors
- Learning itself can fail
  The learning algorithm may not find any reasonable hypothesis

How bad hypotheses are we prepared to accept?

How bad hypotheses are we prepared to accept?

True Error — the probability that a given hypothesis gives the wrong answer

How bad hypotheses are we prepared to accept?

True Error — the probability that a given hypothesis gives the wrong answer

$$\mathrm{error}_{\mathcal{D}}(h) \equiv P_{x \in \mathcal{D}}\left[h(x) \neq c(x)\right]$$

How bad hypotheses are we prepared to accept?

True Error — the probability that a given hypothesis gives the wrong answer

$$\mathrm{error}_{\mathcal{D}}(h) \equiv P_{x \in \mathcal{D}}\left[h(x) \neq c(x)\right]$$

A hypothesis h is called approximately correct (ganska rätt) if

$$\mathrm{error}_{\mathcal{D}}(h) < \epsilon$$

How often may learning fail?

How often may learning fail?

Risk that learning does not find an approximately correct hypothesis

How often may learning fail?

Risk that learning does not find an approximately correct hypothesis

$$P_L\left[\mathrm{error}_{\mathcal{D}}(h) \geq \epsilon\right]$$

How often may learning fail?

Risk that learning does not find an approximately correct hypothesis

$$P_L\left[\mathrm{error}_{\mathcal{D}}(h) \geq \epsilon\right]$$

The algorithm L is said to probably (troligen) find a solution if

$$P_L\left[\mathrm{error}_{\mathcal{D}}(h) \geq \epsilon\right] < \delta$$

PAC-learning

PAC-learning

Probably Approximately Correct

PAC-learning

Probably Approximately Correct

Given

C	the concept to learn
ϵ	limit on the error
δ	limit on the risk
n	size of the examples

PAC-learning

Probably Approximately Correct

Given

C	the concept to learn
ϵ	limit on the error
δ	limit on the risk
n	size of the examples

PAC-learnable: Time to find a solution grows polynomially with respect to $\mathrm{size}(C)$, n, $\frac{1}{\epsilon}$ and $\frac{1}{\delta}$

Analysis of a Consistent Learner

Analysis of a Consistent Learner

- Assumtion: no errors in training examples

Analysis of a Consistent Learner

- Assumtion: no errors in training examples
- Examples are drawn from the distribution $\mathcal{D}$

Analysis of a Consistent Learner

- Assumtion: no errors in training examples
- Examples are drawn from the distribution $\mathcal{D}$
- The solution is consistent with all training examples

Analysis of a Consistent Learner

- Assumtion: no errors in training examples
- Examples are drawn from the distribution $\mathcal{D}$
- The solution is consistent with all training examples
- "Dangerous Hypotheses":

$$\mathrm{error}_{\mathcal{D}}(h) \geq \epsilon$$

Analysis of a Consistent Learner

- Assumtion: no errors in training examples
- Examples are drawn from the distribution $\mathcal{D}$
- The solution is consistent with all training examples
- "Dangerous Hypotheses":

$$\mathrm{error}_{\mathcal{D}}(h) \geq \epsilon$$

We do not want learning to produce a dangerous hypothesis!

Analysis of a Consistent Learner

- Assumtion: no errors in training examples
- Examples are drawn from the distribution $\mathcal{D}$
- The solution is consistent with all training examples
- "Dangerous Hypotheses":

$$\mathrm{error}_{\mathcal{D}}(h) \geq \epsilon$$

We do not want learning to
produce a dangerous hypothesis!

How large is the risk that a dangerous hypothesis is consistent with all training examples?

- Probability that one hypothesis h is contradicted by one example

$$\mathrm{error}_{\mathcal{D}}(h)$$

- Probability that one hypothesis h is contradicted by one example

$$\mathrm{error}_{\mathcal{D}}(h)$$

- Probability that h is not contradicted

$$1 - \mathrm{error}_{\mathcal{D}}(h)$$

- Probability that one hypothesis h is contradicted by one example

$$\mathrm{error}_{\mathcal{D}}(h)$$

- Probability that h is not contradicted

$$1 - \mathrm{error}_{\mathcal{D}}(h)$$

- Risk that a *dangerous hypothesis* ($\mathrm{error}_{\mathcal{D}}(h) \geq \epsilon$) is not contradicted by a randomly drawn example

$$\leq (1 - \epsilon)$$

- Risk that a *dangerous hypothesis* is not contradicted by one randomly drawn example

$$\leq (1 - \epsilon)$$

- Risk that a *dangerous hypothesis* is not contradicted by one randomly drawn example

$$\leq (1 - \epsilon)$$

- Risk that a *dangerous hypothesis* is not contradicted my m randomly drawn examples

- Risk that a *dangerous hypothesis* is not contradicted by one randomly drawn example

$$\leq (1 - \epsilon)$$

- Risk that a *dangerous hypothesis* is not contradicted my m randomly drawn examples

$$\leq (1 - \epsilon)^m$$

- Risk that a *dangerous hypothesis* is not contradicted by one randomly drawn example

$$\leq (1 - \epsilon)$$

- Risk that a *dangerous hypothesis* is not contradicted my m randomly drawn examples

$$\leq (1 - \epsilon)^m$$

- How large is the risk that any dangerous hypothesis in H happens to be consistent with all examples:

- Risk that a *dangerous hypothesis* is not contradicted by one randomly drawn example

$$\leq (1 - \epsilon)$$

- Risk that a *dangerous hypothesis* is not contradicted my m randomly drawn examples

$$\leq (1 - \epsilon)^m$$

- How large is the risk that any dangerous hypothesis in H happens to be consistent with all examples:

$$\leq |H| \cdot (1 - \epsilon)^m$$

- Risk that a *dangerous hypothesis* is not contradicted by one randomly drawn example

$$\leq (1-\epsilon)$$

- Risk that a *dangerous hypothesis* is not contradicted my *m* randomly drawn examples

$$\leq (1-\epsilon)^m$$

- How large is the risk that any dangerous hypothesis in H happens to be consistent with all examples:

$$\leq |H| \cdot (1-\epsilon)^m$$

$$\leq |H| \cdot e^{-\epsilon m}$$

How many training examples are needed?

How many training examples are needed?

How many examples m are needed to make the risk of ending up with a dangerous hypothesis less than δ?

How many training examples are needed?

How many examples m are needed to make the risk of ending up with a dangerous hypothesis less than δ?

$$\delta \geq |H| \cdot e^{-\epsilon m}$$

How many training examples are needed?

How many examples m are needed to make the risk of ending up with a dangerous hypothesis less than δ?

$$\delta \geq |H| \cdot e^{-\epsilon m}$$

$$e^{\epsilon m} \geq \frac{|H|}{\delta}$$

How many training examples are needed?

How many examples m are needed to make the risk of ending up with a dangerous hypothesis less than δ?

$$\delta \geq |H| \cdot e^{-\epsilon m}$$

$$e^{\epsilon m} \geq \frac{|H|}{\delta}$$

$$m \geq \frac{1}{\epsilon} \left[\ln |H| + \ln \frac{1}{\delta} \right]$$

Important relation:

$$m \geq \frac{1}{\epsilon}\left[\ln|H| + \ln\frac{1}{\delta}\right]$$

Important relation:

$$m \geq \frac{1}{\epsilon}\left[\ln|H| + \ln\frac{1}{\delta}\right]$$

Is this PAC-learnable?

Important relation:

$$m \geq \frac{1}{\epsilon}\left[\ln|H| + \ln\frac{1}{\delta}\right]$$

Is this PAC-learnable?

Potential problem: $|H|$ might be too large

Learning Conjunctions

Learning Conjunctions

Example: $\text{Sunny} \wedge \neg\text{Windy} \wedge \text{Humid}$

Learning Conjunctions

Example: $\text{Sunny} \wedge \neg\text{Windy} \wedge \text{Humid}$

n attributes $\Rightarrow 3^n$ possible concepts $\Rightarrow |H| = 3^n$

Learning Conjunctions

Example: $\text{Sunny} \wedge \neg\text{Windy} \wedge \text{Humid}$

n attributes $\Rightarrow 3^n$ possible concepts $\Rightarrow |H| = 3^n$

$$m \geq \frac{1}{\epsilon}\left[n \ln 3 + \ln \frac{1}{\delta}\right]$$

Learning Conjunctions

Example: $\text{Sunny} \land \neg\text{Windy} \land \text{Humid}$

n attributes $\Rightarrow 3^n$ possible concepts $\Rightarrow |H| = 3^n$

$$m \geq \frac{1}{\epsilon}\left[n \ln 3 + \ln \frac{1}{\delta}\right]$$

- Linear w.r.t. $\frac{1}{\epsilon}$

Learning Conjunctions

Example: $\text{Sunny} \wedge \neg\text{Windy} \wedge \text{Humid}$

n attributes $\Rightarrow$ 3^n possible concepts $\Rightarrow$ $|H| = 3^n$

$$m \geq \frac{1}{\epsilon}\left[n \ln 3 + \ln \frac{1}{\delta}\right]$$

- Linear w.r.t. $\frac{1}{\epsilon}$
- Linear w.r.t. n

Learning Conjunctions

Example: $\mathrm{Sunny} \wedge \neg\mathrm{Windy} \wedge \mathrm{Humid}$

n attributes $\Rightarrow 3^n$ possible concepts $\Rightarrow |H| = 3^n$

$$m \geq \frac{1}{\epsilon}\left[n \ln 3 + \ln \frac{1}{\delta}\right]$$

- Linear w.r.t. $\frac{1}{\epsilon}$
- Linear w.r.t. n
- Logarithmic w.r.t. $\frac{1}{\delta}$

Learning Conjunctions

Example: $\text{Sunny} \wedge \neg\text{Windy} \wedge \text{Humid}$

n attributes $\Rightarrow 3^n$ possible concepts $\Rightarrow |H| = 3^n$

$$m \geq \frac{1}{\epsilon}\left[n \ln 3 + \ln \frac{1}{\delta}\right]$$

- Linear w.r.t. $\frac{1}{\epsilon}$
- Linear w.r.t. n
- Logarithmic w.r.t. $\frac{1}{\delta}$

Seems PAC-learnable!

Further, *time for each example* must be polynomial.

Find-S: Ok

Unbiased Learning

Unbiased Learning

All subsets of X are hypotheses

Unbiased Learning

All subsets of X are hypotheses

$$|X| = 2^n$$

Unbiased Learning

All subsets of X are hypotheses

$$|X| = 2^n$$

$$|H| = 2^{2^n}$$

Unbiased Learning

All subsets of X are hypotheses

$$|X| = 2^n$$

$$|H| = 2^{2^n}$$

$$m \geq \frac{1}{\epsilon}\left[2^n \ln 2 + \ln \frac{1}{\delta}\right]$$

Unbiased Learning

All subsets of X are hypotheses

$$|X| = 2^n$$

$$|H| = 2^{2^n}$$

$$m \geq \frac{1}{\epsilon}\left[2^n \ln 2 + \ln \frac{1}{\delta}\right]$$

Not PAC-learnable!

Unbiased Learning

All subsets of X are hypotheses

$$|X| = 2^n$$

$$|H| = 2^{2^n}$$

$$m \geq \frac{1}{\epsilon}\left[2^n \ln 2 + \ln\frac{1}{\delta}\right]$$

Not PAC-learnable!

However, this estimate is an upper bound

We have not proven that m actually grows exponentially w.r.t. n

Unbiased Learning

All subsets of X are hypotheses

$$|X| = 2^n$$

$$|H| = 2^{2^n}$$

$$m \geq \frac{1}{\epsilon}\left[2^n \ln 2 + \ln \frac{1}{\delta}\right]$$

Not PAC-learnable!
However, this estimate is an upper bound
We have not proven that m actually grows exponentially w.r.t. n
However, in this case it *is* true

Problem with $|H|$

Problem with $|H|$

- Gives too pessimistic estimates

Problem with $|H|$

- Gives too pessimistic estimates
- Can't be used when $|H| = \infty$

Problem with $|H|$

- Gives too pessimistic estimates
- Can't be used when $|H| = \infty$

Vapnik — Chervonenkis observation:

The important thing is not the *number of* hypotheses, but how they can form subsets in X

Scattering

A finite set S is scattered (splittras) by the hypotheses H if every subset of S is described by a $h \in H$

Scattering

A finite set S is scattered (splittras) by the hypotheses H if every subset of S is described by a $h \in H$

The size of S is a measure of the expressive power of H

Scattering

A finite set S is scattered (splittras) by the hypotheses H if every subset of S is described by a $h \in H$

The size of S is a measure of the expressive power of H

VC Dimension

$$\mathrm{VC}(H)$$

Size of the largest subset
of X which can be scattered by H

Example:

H	Intervals on the real axis
X	Real numbers

Example:

H Intervals on the real axis
X Real numbers

- Can 2 points be scattered?

Example:

H	Intervals on the real axis
X	Real numbers

- Can 2 points be scattered?
- Can 3 points be scattered?

Example:

H	Intervals on the real axis
X	Real numbers

- Can 2 points be scattered?
- Can 3 points be scattered?

Conclusion: $\mathrm{VC}(H) = 2$

Example:

H	Separating hyperplane
X	Points in $\Re^r$

Example:

H Separating hyperplane
X Points in $\Re^r$

- When $r = 1$

Example:

H	Separating hyperplane
X	Points in $\Re^r$

- When $r = 1$

$$\mathrm{VC}(H) = 2$$

Example:

H	Separating hyperplane
X	Points in $\Re^r$

- When $r = 1$

$$\mathrm{VC}(H) = 2$$

- When $r = 2$

Example:

H	Separating hyperplane
X	Points in $\Re^r$

- When $r = 1$

$$\mathrm{VC}(H) = 2$$

- When $r = 2$

$$\mathrm{VC}(H) = 3$$

Example:

H	Separating hyperplane
X	Points in $\Re^r$

- When $r = 1$

$$\mathrm{VC}(H) = 2$$

- When $r = 2$

$$\mathrm{VC}(H) = 3$$

- Generally

Example:

H Separating hyperplane
X Points in $\Re^r$

- When $r = 1$
$$\mathrm{VC}(H) = 2$$
- When $r = 2$
$$\mathrm{VC}(H) = 3$$
- Generally
$$\mathrm{VC}(H) = r + 1$$

Number of Training Examples

Number of Training Examples

Previous estimate

$$m \geq \frac{1}{\epsilon}\left[\ln|H| + \ln\frac{1}{\delta}\right]$$

Number of Training Examples

Previous estimate

$$m \geq \frac{1}{\epsilon}\left[\ln |H| + \ln \frac{1}{\delta}\right]$$

New estimate

$$m \geq \frac{1}{\epsilon}\left[4 \log_2 \frac{2}{\delta} + 8\mathrm{VC}(H) \cdot \log_2 \frac{13}{\epsilon}\right]$$

Number of Training Examples

Previous estimate

$$m \geq \frac{1}{\epsilon}\left[\ln|H| + \ln\frac{1}{\delta}\right]$$

New estimate

$$m \geq \frac{1}{\epsilon}\left[4\log_2\frac{2}{\delta} + 8\mathrm{VC}(H)\cdot\log_2\frac{13}{\epsilon}\right]$$

Much better (smaller)

Alternative Performance Measure for Learning Algorithms:

How many errors does the algorithm make during learning

Find-S

Find-S

- Only learns when making errors

Find-S

- Only learns when making errors
- Worst case: generalises only one attribute each time

Find-S

- Only learns when making errors
- Worst case: generalises only one attribute each time
- First example only chooses one specific hypothesis

Find-S

- Only learns when making errors
- Worst case: generalises only one attribute each time
- First example only chooses one specific hypothesis
- Maximally $n + 1$ changes

Find-S

- Only learns when making errors
- Worst case: generalises only one attribute each time
- First example only chooses one specific hypothesis
- Maximally $n + 1$ changes

Will maximally make $n + 1$ errors

Candidate Elimination

Candidate Elimination

- We must force the algorithm to guess

Candidate Elimination

- We must force the algorithm to guess
- Suppose we use a majority vote among all hypotheses remaining in *Version Space*

Candidate Elimination

- We must force the algorithm to guess
- Suppose we use a majority vote among all hypotheses remaining in *Version Space*
- Wrong answer only when at least half of VS give the wrong answer

Candidate Elimination

- We must force the algorithm to guess
- Suppose we use a majority vote among all hypotheses remaining in *Version Space*
- Wrong answer only when at least half of VS give the wrong answer
- For each error made, at least half of VS disappears

Candidate Elimination

- We must force the algorithm to guess
- Suppose we use a majority vote among all hypotheses remaining in *Version Space*
- Wrong answer only when at least half of VS give the wrong answer
- For each error made, at least half of VS disappears

Maximally $\log_2 |H|$ errors

Optimal Learning

- Best algorithm
- Worst case
  Trickiest concept, worst order of examples

Optimal Learning

- Best algorithm
- Worst case
  Trickiest concept, worst order of examples

Number of errors while learning concept C

$$\mathrm{Opt}(C)$$

Optimal Learning

- Best algorithm
- Worst case
  Trickiest concept, worst order of examples

Number of errors while learning concept C

$$\mathrm{Opt}(C)$$

Theoretical Limit

$$\mathrm{VC}(C) \leq \mathrm{Opt}(C) \leq \log_2 |C|$$