Computational Learning Theory: An Analysis of a Conjunction Learner

Machine Learning

THE UNIVERSITY OF UTAH

Slides based on material from Dan Roth, Avrim Blum, Tom Mitchell and others

This lecture: Computational Learning Theory

- The Theory of Generalization
- Probably Approximately Correct (PAC) learning
- Positive and negative learnability results
- Agnostic Learning
- Shattering and the VC dimension

Where are we?

- The Theory of Generalization
  - When can be trust the learning algorithm?
  - What functions can be learned?
  - Batch Learning
- Probably Approximately Correct (PAC) learning
- Positive and negative learnability results
- Agnostic Learning
- Shattering and the VC dimension

This section

1. Analyze a simple algorithm for learning conjunctions

2. Define the PAC model of learning

3. Make formal connections to the principle of Occam's razor

This section

1. Analyze a simple algorithm for learning conjunctions

2. Define the PAC model of learning

3. Make formal connections to the principle of Occam's razor

Learning Conjunctions

The true function $f = x_2 \wedge x_3 \wedge x_4 \wedge x_5 \wedge x_{100}$

Training data

- <(1,1,1,1,1,1,…,1,1), 1>
- <(1,1,1,0,0,0,…,0,0), 0>
- <(1,1,1,1,1,0,...0,1,1), 1>
- <(1,0,1,1,1,0,...0,1,1), 0>
- <(1,1,1,1,1,0,...0,0,1), 1>
- <(1,0,1,0,0,0,...0,1,1), 0>
- <(1,1,1,1,1,1,…,0,1), 1>
- <(0,1,0,1,0,0,...0,1,1), 0>

Learning Conjunctions

$$f = x_2 \wedge x_3 \wedge x_4 \wedge x_5 \wedge x_{100}$$

Training data

– <(1,1,1,1,1,1,…,1,1), 1>

~~– <(1,1,1,0,0,0,…,0,0), 0>~~

– <(*1,1,1,1,1*,0,...0,1,*1*), 1>

~~– <(1,0,1,1,1,0,...0,1,1), 0>~~

– <(*1,1,1,1,1*,0,...0,0,*1*), 1>

~~– <(1,0,1,0,0,0,...0,1,1), 0>~~

– <(*1,1,1,1,1*,1,…,0,*1*), 1>

~~– <(0,1,0,1,0,0,...0,1,1), 0>~~

A simple learning algorithm (*Elimination*)

- Discard all negative examples

Learning Conjunctions

$$f = x_2 \wedge x_3 \wedge x_4 \wedge x_5 \wedge x_{100}$$

Training data

- <(1,1,1,1,1,1,…,1,1), 1>
- ~~<(1,1,1,0,0,0,…,0,0), 0>~~
- <(1,1,1,1,1,0,...0,1,1), 1>
- ~~<(1,0,1,1,1,0,...0,1,1), 0>~~
- <(1,1,1,1,1,0,...0,0,1), 1>
- ~~<(1,0,1,0,0,0,...0,1,1), 0>~~
- <(1,1,1,1,1,1,…,0,1), 1>
- ~~<(0,1,0,1,0,0,...0,1,1), 0>~~

A simple learning algorithm (*Elimination*)

- Discard all negative examples
- Build a conjunction using the features that are common to all positive conjunctions

$$h = x_1 \wedge x_2 \wedge x_3 \wedge x_4 \wedge x_5 \wedge x_{100}$$

Learning Conjunctions

$$f = x_2 \wedge x_3 \wedge x_4 \wedge x_5 \wedge x_{100}$$

Training data

- <(*1*,*1*,*1*,*1*,*1*,1,...,1,*1*), 1>
- ~~<(1,1,1,0,0,0,...,0,0), 0>~~
- <(*1*,*1*,*1*,*1*,*1*,0,...0,1,*1*), 1>
- ~~<(1,0,1,1,1,0,...0,1,1), 0>~~
- <(*1*,*1*,*1*,*1*,*1*,$\boxed{0}$,...0,$\boxed{0}$,*1*), 1>
- ~~<(1,0,1,0,0,0,...0,1,1), 0>~~
- <(*1*,*1*,*1*,*1*,*1*,1,...,$\boxed{0}$,*1*), 1>
- ~~<(0,1,0,1,0,0,...0,1,1), 0>~~

A simple learning algorithm (*Elimination*)

- Discard all negative examples
- Build a conjunction using the features that are common to all positive conjunctions

$$h = x_1 \wedge x_2 \wedge x_3 \wedge x_4 \wedge x_5 \wedge x_{100}$$

Positive examples ***eliminate*** irrelevant features

Learning Conjunctions

$$f = x_2 \wedge x_3 \wedge x_4 \wedge x_5 \wedge x_{100}$$

Training data

- <(1,1,1,1,1,1,…,1,1), 1>
- <(1,1,1,0,0,0,…,0,0), 0>
- <(1,1,1,1,1,0,...0,1,1), 1>
- <(1,0,1,1,1,0,...0,1,1), 0>
- <(1,1,1,1,1,0,...0,0,1), 1>
- <(1,0,1,0,0,0,...0,1,1), 0>
- <(1,1,1,1,1,1,…,0,1), 1>
- <(0,1,0,1,0,0,...0,1,1), 0>

A simple learning algorithm:

- Discard all negative examples
- Build a conjunction using the features that are common to all positive conjunctions

$$h = x_1 \wedge x_2 \wedge x_3 \wedge x_4 \wedge x_5 \wedge x_{100}$$

Clearly this algorithm produces a conjunction that is consistent with the data, that is $err_S(h) = 0$, if the target function is a monotone conjunction
Exercise: Why?

Learning Conjunctions: Analysis

$f = x_2 \wedge x_3 \wedge x_4 \wedge x_5 \wedge x_{100}$ $h = \boxed{x_1 \wedge} x_2 \wedge x_3 \wedge x_4 \wedge x_5 \wedge x_{100}$

Claim 1: Any hypothesis consistent with the training data will only make mistakes on positive future examples

Why?

Learning Conjunctions: Analysis

$f = x_2 \wedge x_3 \wedge x_4 \wedge x_5 \wedge x_{100}$

$h = \boxed{x_1 \wedge} x_2 \wedge x_3 \wedge x_4 \wedge x_5 \wedge x_{100}$

Claim 1: Any hypothesis consistent with the training data will only make mistakes on positive future examples

Why?

A mistake will occur only if some literal z (in our example x_1) is present in h but not in f

This mistake can cause a positive example to be predicted as negative by h

Specifically: $x_1 = 0$, $x_2 = 1$, $x_3 = 1$, $x_4 = 1$, $x_5 = 1$, $x_{100} = 1$

Learning Conjunctions: Analysis

$f = x_2 \wedge x_3 \wedge x_4 \wedge x_5 \wedge x_{100}$

$h = \boxed{x_1 \wedge} x_2 \wedge x_3 \wedge x_4 \wedge x_5 \wedge x_{100}$

Claim 1: Any hypothesis consistent with the training data will only make mistakes on positive future examples

Why?

A mistake will occur only if some literal z (in our example x_1) is present in h but not in f

- This mistake can cause a positive example to be predicted as negative by h   Specifically: $x_1 = 0$, $x_2 = 1$, $x_3 = 1$, $x_4 = 1$, $x_5 = 1$, $x_{100} = 1$

The reverse situation can never happen

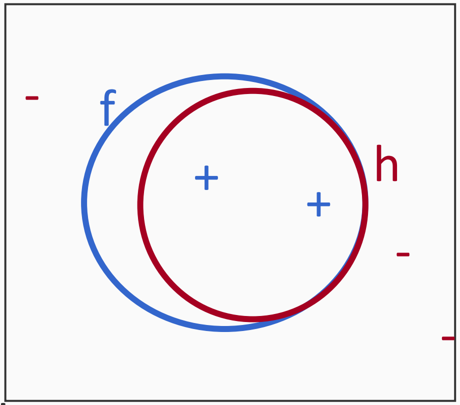
- For an example to be predicted as positive in the training set, every relevant literal must have been present

Learning Conjunctions: Analysis

$$f = x_2 \wedge x_3 \wedge x_4 \wedge x_5 \wedge x_{100} \qquad h = \boxed{x_1 \wedge} x_2 \wedge x_3 \wedge x_4 \wedge x_5 \wedge x_{100}$$

Claim 1: Any hypothesis consistent with the training data will only make mistakes on positive future examples

Why?



A mistake will occur only if some literal z (in our example x_1) is present in h but not in f

This mistake can cause a positive example to be predicted as negative by h   Specifically: $x_1 = 0$, $x_2 =1$, $x_3=1$, $x_4=1$, $x_5=1$, $x_{100}=1$

The reverse situation can never happen

For an example to be predicted as positive in the training set, every relevant literal must have been present

Learning Conjunctions: Analysis

Theorem: Suppose we are learning a conjunctive concept with n dimensional Boolean features using m training examples. If

$$m > \frac{n}{\epsilon}\left(\log(n) + \log\left(\frac{1}{\delta}\right)\right)$$

then, with probability > 1 - δ, the error of the learned hypothesis $err_D(h)$ will be less than ϵ.

Learning Conjunctions: Analysis

Theorem: Suppose we are learning a conjunctive concept with n dimensional Boolean features using m training examples. If

$$m > \frac{n}{\epsilon}\left(\log(n) + \log\left(\frac{1}{\delta}\right)\right)$$

Poly in n, $1/\delta$, $1/\epsilon$

then, with probability > 1 - δ, the error of the learned hypothesis $err_D(h)$ will be less than ϵ.

If we see these many training examples, then the algorithm will produce a conjunction that, with high probability, will make few errors

Learning Conjunctions: Analysis

Theorem: Suppose we are learning a conjunctive concept with n dimensional Boolean features using m training examples. If

$$m > \frac{n}{\epsilon}\left(\log(n) + \log\left(\frac{1}{\delta}\right)\right)$$

then, with probability > 1 - δ, the error of the learned hypothesis $err_D(h)$ will be less than ϵ.

Let's prove this assertion

Proof Intuition

$f = x_2 \wedge x_3 \wedge x_4 \wedge x_5 \wedge x_{100}$

$h = \boxed{x_1 \wedge} x_2 \wedge x_3 \wedge x_4 \wedge x_5 \wedge x_{100}$

What kinds of examples would drive a hypothesis to make a mistake?

Proof Intuition

$f = x_2 \wedge x_3 \wedge x_4 \wedge x_5 \wedge x_{100}$

$h = \boxed{x_1 \wedge} x_2 \wedge x_3 \wedge x_4 \wedge x_5 \wedge x_{100}$

What kinds of examples would drive a hypothesis to make a mistake?

Positive examples, where x_1 is absent

f would say `true` and h would say `false`

Proof Intuition

$f = x_2 \wedge x_3 \wedge x_4 \wedge x_5 \wedge x_{100}$

$h = \boxed{x_1 \wedge} x_2 \wedge x_3 \wedge x_4 \wedge x_5 \wedge x_{100}$

What kinds of examples would drive a hypothesis to make a mistake?

Positive examples, where x_1 is absent

f would say `true` and h would say `false`

None of these examples appeared during training

Otherwise x_1 would have been eliminated

Proof Intuition

$f = x_2 \wedge x_3 \wedge x_4 \wedge x_5 \wedge x_{100}$

$h = x_1 \wedge x_2 \wedge x_3 \wedge x_4 \wedge x_5 \wedge x_{100}$

What kinds of examples would drive a hypothesis to make a mistake?

Positive examples, where x_1 is absent

- f would say `true` and h would say `false`

None of these examples appeared during training

- Otherwise x_1 would have been eliminated

If they never appeared during training, maybe their appearance in the future would also be rare!

Proof Intuition

$f = x_2 \wedge x_3 \wedge x_4 \wedge x_5 \wedge x_{100}$ $h = \boxed{x_1 \wedge} x_2 \wedge x_3 \wedge x_4 \wedge x_5 \wedge x_{100}$

What kinds of examples would drive a hypothesis to make a mistake?

Positive examples, where x_1 is absent

- f would say `true` and h would say `false`

None of these examples appeared during training

- Otherwise x_1 would have been eliminated

If they never appeared during training, maybe their appearance in the future would also be rare!

Let's quantify our surprise at seeing such examples

Learning Conjunctions: Analysis

Let p(z) be the probability that, in an example drawn from D, the feature z is absent but the example has a positive label

- That is, after training is done, p(z) is the probability that in a randomly drawn example, the literal z causes a mistake

- For any z in the target function, p(z) = 0

Learning Conjunctions: Analysis

Let p(z) be the probability that, in an example drawn from D, the feature z is absent but the example has a positive label

- That is, after training is done, p(z) is the probability that in a randomly drawn example, the literal z causes a mistake

Remember that there will only be mistakes on positive examples for this toy problem

- For any z in the target function, p(z) = 0

Learning Conjunctions: Analysis

Let p(z) be the probability that, in an example drawn from D, the feature z is absent but the example has a positive label

- That is, after training is done, p(z) is the probability that in a randomly drawn example, the literal z causes a mistake

- For any z in the target function, p(z) = 0

Remember that there will only be mistakes on positive examples for this toy problem

$$f = x_2 \wedge x_3 \wedge x_4 \wedge x_5 \wedge x_{100}$$

$$h = \boxed{x_1 \wedge} x_2 \wedge x_3 \wedge x_4 \wedge x_5 \wedge x_{100}$$

Learning Conjunctions: Analysis

Let p(z) be the probability that, in an example drawn from D, the feature z is absent but the example has a positive label

- That is, after training is done, p(z) is the probability that in a randomly drawn example, the literal z causes a mistake

- For any z in the target function, p(z) = 0

Remember that there will only be mistakes on positive examples for this toy problem

$$f = x_2 \land x_3 \land x_4 \land x_5 \land x_{100}$$

<(0,*1*,*1*,*1*,*1*,0,...0,1,*1*), 1>

$$h = x_1 \land x_2 \land x_3 \land x_4 \land x_5 \land x_{100}$$

Learning Conjunctions: Analysis

Let p(z) be the probability that, in an example drawn from D, the feature z is absent but the example has a positive label

- That is, after training is done, p(z) is the probability that in a randomly drawn example, the literal z causes a mistake
- For any z in the target function, p(z) = 0

Remember that there will only be mistakes on positive examples for this toy problem

$$f = x_2 \wedge x_3 \wedge x_4 \wedge x_5 \wedge x_{100}$$

$$h = \boxed{x_1 \wedge} x_2 \wedge x_3 \wedge x_4 \wedge x_5 \wedge x_{100}$$

<($\boxed{0}$,*1*,*1*,*1*,*1*,0,...0,1,*1*), $\boxed{1}$>

p(x_1): Probability that this situation occurs

Learning Conjunctions: Analysis

Let p(z) be the probability that, in an example drawn from D, the feature z is absent but the example has a positive label

- That is, after training is done, p(z) is the probability that in a randomly drawn example, the literal z causes a mistake

- For any z in the target function, p(z) = 0

Learning Conjunctions: Analysis

Let p(z) be the probability that, in an example drawn from D, the feature z is absent but the example has a positive label

- That is, after training is done, p(z) is the probability that in a randomly drawn example, the literal z causes a mistake

- For any z in the target function, p(z) = 0

We know that $err_D(h) \leq \sum_{z \in h} p(z)$

Via direct application of the union bound

Learning Conjunctions: Analysis

Let p(z) be the probability that, in an example drawn from D, the feature z is absent but the example has a positive label

- That is, after training is done, p(z) is the probability that in a randomly drawn example, the literal z causes a mistake

- For any z in the target function, p(z) = 0

We know that $err_D(h) \leq \sum_{z \in h} p(z)$

Via direct application of the union bound

Union bound
For a set of events, probability that at least one of them happens < the sum of the probabilities of the individual events

Learning Conjunctions: Analysis

n = dimensionality

- Call a literal z *bad* if $p(z) > \frac{\epsilon}{n}$
- Intuitively, a bad literal is one that has a significant probability of not appearing with a positive example
  - (And, if it appears in all positive training examples, it can cause errors)

If there are no bad literals, then $\mathrm{err}_D(h) < \epsilon$

Learning Conjunctions: Analysis

n = dimensionality

- Call a literal z *bad* if $p(z) > \frac{\epsilon}{n}$
- Intuitively, a bad literal is one that has a significant probability of not appearing with a positive example
  - (And, if it appears in all positive training examples, it can cause errors)

If there are no bad literals, then $err_D(h) < \epsilon$

- Why? Because $err_D(h) \le \sum_{z \in h} p(z)$

Learning Conjunctions: Analysis

n = dimensionality

- Call a literal z *bad* if $p(z) > \frac{\epsilon}{n}$
- Intuitively, a bad literal is one that has a significant probability of not appearing with a positive example
  – (And, if it appears in all positive training examples, it can cause errors)

If there are no bad literals, then $\text{err}_D(h) < \epsilon$

– Why? Because $err_D(h) \leq \sum_{z \in h} p(z)$

Let us try to see when this will not happen

Learning Conjunctions: Analysis

n = dimensionality

- Call a literal z *bad* if $p(z) > \frac{\epsilon}{n}$
- Intuitively, a bad literal is one that has a significant probability of not appearing with a positive example
  - (And, if it appears in all positive training examples, it can cause errors)

What if there are bad literals?

Learning Conjunctions: Analysis

n = dimensionality

- Call a literal z *bad* if $p(z) > \frac{\epsilon}{n}$
- Intuitively, a bad literal is one that has a significant probability of not appearing with a positive example
  - (And, if it appears in all positive training examples, it can cause errors)

What if there are bad literals?

Let z be a bad literal

What is the probability that it will not be eliminated by one training example?

Learning Conjunctions: Analysis

n = dimensionality

- Call a literal z *bad* if $p(z) > \frac{\epsilon}{n}$
- Intuitively, a bad literal is one that has a significant probability of not appearing with a positive example
  - (And, if it appears in all positive training examples, it can cause errors)

What if there are bad literals?

Let z be a bad literal

What is the probability that it will not be eliminated by one training example?

$$\begin{aligned} Pr(z \text{ survives one example}) &= 1 - Pr(z \text{ is eliminated by one example}) \\ &\leq 1 - p(z) \\ &< 1 - \frac{\epsilon}{n} \end{aligned}$$

Learning Conjunctions: Analysis

n = dimensionality

- Call a literal z *bad* if $p(z) > \frac{\epsilon}{n}$
- Intuitively, a bad literal is one that has a significant probability of not appearing with a positive example
  - (And, if it appears in all positive training examples, it can cause errors)

There was one example of this kind

<($\boxed{1}$,1,1,1,1,0,...0,1,1), 1>

What if there are bad literals?

Let z be a bad literal

What is the probability that it will not be eliminated by one training example?

$$\begin{aligned} Pr(z \text{ survives one example}) &= 1 - Pr(z \text{ is eliminated by one example}) \\ &\leq 1 - p(z) \\ &< 1 - \frac{\epsilon}{n} \end{aligned}$$

Learning Conjunctions: Analysis

n = dimensionality

What we know so far:

$$Pr(\text{A bad literal is not eliminated by one example}) < 1 - \frac{\epsilon}{n}$$

Learning Conjunctions: Analysis

n = dimensionality

What we know so far:

$$Pr(\text{A bad literal is not eliminated by one example}) < 1 - \frac{\epsilon}{n}$$

But say we have m training examples. Then

$$Pr(\text{A bad literal survives } m \text{ examples}) < \left(1 - \frac{\epsilon}{n}\right)^m$$

Learning Conjunctions: Analysis

n = dimensionality

What we know so far:

$$Pr(\text{A bad literal is not eliminated by one example}) < 1 - \frac{\epsilon}{n}$$

But say we have m training examples. Then

$$Pr(\text{A bad literal survives } m \text{ examples}) < \left(1 - \frac{\epsilon}{n}\right)^m$$

There are at most n bad literals. So

$$Pr(\text{Any bad literal survives } m \text{ examples}) < n\left(1 - \frac{\epsilon}{n}\right)^m$$

Learning Conjunctions: Analysis

$$Pr(\text{Any bad literal survives } m \text{ examples}) < n\left(1 - \frac{\epsilon}{n}\right)^m$$

We want this probability to be small

Why? So that we can choose enough training examples so that the probability that any z survives all of them is less than some δ

Learning Conjunctions: Analysis

$$Pr(\text{Any bad literal survives } m \text{ examples}) < n\left(1 - \frac{\epsilon}{n}\right)^m$$

We want this probability to be small

Why? So that we can choose enough training examples so that the probability that any z survives all of them is less than some δ

That is, we want $n\left(1 - \frac{\epsilon}{n}\right)^m < \delta$

Learning Conjunctions: Analysis

$$Pr(\text{Any bad literal survives } m \text{ examples}) < n\left(1 - \frac{\epsilon}{n}\right)^m$$

We want this probability to be small

Why? So that we can choose enough training examples so that the probability that any z survives all of them is less than some δ

That is, we want $n\left(1 - \frac{\epsilon}{n}\right)^m < \delta$

We know that $1 - x < e^{-x}$. So it is sufficient to require $ne^{-\frac{m\epsilon}{n}} < \delta$

Learning Conjunctions: Analysis

$$Pr(\text{Any bad literal survives } m \text{ examples}) < n\left(1 - \frac{\epsilon}{n}\right)^m$$

We want this probability to be small

Why? So that we can choose enough training examples so that the probability that any z survives all of them is less than some δ

That is, we want $n\left(1 - \frac{\epsilon}{n}\right)^m < \delta$

We know that $1 - x < e^{-x}$. So it is sufficient to require $ne^{-\frac{m\epsilon}{n}} < \delta$

Or equivalently, $m > \frac{n}{\epsilon}\left(\log(n) + \log\left(\frac{1}{\delta}\right)\right)$

Learning Conjunctions: Analysis

To guarantee a probability of failure (i.e, error > ϵ) that is less than δ, the number of examples we need is

$$m > \frac{n}{\epsilon}\left(\log(n) + \log\left(\frac{1}{\delta}\right)\right)$$

Poly in n, $1/\delta$, $1/\epsilon$

That is, if m has this property, then

- With probability 1 - δ, there will be no bad literals,
- Or equivalently, with probability 1 - δ, we will have $err_D(h) < \epsilon$

Learning Conjunctions: Analysis

To guarantee a probability of failure (i.e, error > ϵ) that is less than δ, the number of examples we need is

$$m > \frac{n}{\epsilon}\left(\log(n) + \log\left(\frac{1}{\delta}\right)\right)$$

$\boxed{\text{Poly in n, } 1/\delta,\ 1/\epsilon}$

That is, if m has this property, then

- With probability 1 - δ, there will be no bad literals,
- Or equivalently, with probability 1 - δ, we will have err_D(h) < ϵ

What does this mean:

- If ϵ = 0.1 and δ = 0.1, then for n = 100, we need 6908 training examples

Learning Conjunctions: Analysis

To guarantee a probability of failure (i.e, error > ϵ) that is less than δ, the number of examples we need is

$$m > \frac{n}{\epsilon}\left(\log(n) + \log\left(\frac{1}{\delta}\right)\right)$$

Poly in n, $1/\delta$, $1/\epsilon$

That is, if m has this property, then

- With probability 1 - δ, there will be no bad literals,
- Or equivalently, with probability 1 - δ, we will have $\text{err}_D(h) < \epsilon$

What does this mean:

- If ϵ = 0.1 and δ = 0.1, then for n = 100, we need 6908 training examples
- If ϵ = 0.1 and δ = 0.1, then for n = 10, we need only 461 examples

Learning Conjunctions: Analysis

To guarantee a probability of failure (i.e, error > ϵ) that is less than δ, the number of examples we need is

$$m > \frac{n}{\epsilon}\left(\log(n) + \log\left(\frac{1}{\delta}\right)\right)$$

Poly in n, $1/\delta$, $1/\epsilon$

That is, if m has this property, then

- With probability 1 - δ, there will be no bad literals,
- Or equivalently, with probability 1 - δ, we will have $\text{err}_D(h) < \epsilon$

What does this mean:

- If ϵ = 0.1 and δ = 0.1, then for n = 100, we need 6908 training examples
- If ϵ = 0.1 and δ = 0.1, then for n = 10, we need only 461 examples
- If ϵ = 0.1 and δ = 0.01, then for n = 10, we need 691 examples

Learning Conjunctions: Analysis

To guarantee a probability of failure (i.e, error > ϵ) that is less than δ, the number of examples we need is

$$m > \frac{n}{\epsilon}\left(\log(n) + \log\left(\frac{1}{\delta}\right)\right)$$

That is, if m has this property, then

- With probability 1 - δ, there will be no bad literals,
- Or equivalently, with probability 1 - δ, we will have $\mathrm{err}_D(h) < \epsilon$

What we have here is a PAC guarantee

Our algorithm is Probably Approximately Correct