Data Science

Introduction to Machine Learning

July 6, 2021

Where were left off last time:

Preliminaries:

Where were left off last time:

Preliminaries:

1. Different distributions

Where were left off last time:

Preliminaries:

1. Different distributions
2. Different ways of reasoning about distributions (PDF, CDF)

Where were left off last time:

Preliminaries:

1. Different distributions
2. Different ways of reasoning about distributions (PDF, CDF)
3. Beginnings of Hypothesis Testing

Bounds

Bounds

1. We discussed ways to use the CDF of a distribution to get *bounds* on some value

Bounds

1. We discussed ways to use the CDF of a distribution to get *bounds* on some value
2. Without running more trials (or gathering more data), we can *increase* certainty by *widening* our bounds

Bounds

1. We discussed ways to use the CDF of a distribution to get *bounds* on some value
2. Without running more trials (or gathering more data), we can *increase* certainty by *widening* our bounds
3. But we weren't very concrete about how this relates to H_0 and H_1

Significance and Power

We need to talk about two aspects of interpreting experimental results:

Significance and Power

We need to talk about two aspects of interpreting experimental results:

1. *Significance*: How willing are we to reject H_0, even if it's true

Significance and Power

We need to talk about two aspects of interpreting experimental results:

1. *Significance*: How willing are we to reject H_0, even if it's true
2. *Power*: How willing are we to *fail* to reject H_0, even if it's false.

Errors

Significance and Power relate to errors.

Errors

Significance and Power relate to errors.

1. Type 1 error: “false positive” (Significance)

Errors

Significance and Power relate to errors.

1. Type 1 error: “false positive” (Significance)
2. Type 2 error: “false negative” (Power)

Errors in the Judicial System

Errors in the Judicial System

	Innocent	Guilty
Guilty Verdict	???????	Correct
Not Guilty Verdict	Correct	??????

Errors in the Judicial System

	Innocent	Guilty
Guilty Verdict	Type 1	Correct
Not Guilty Verdict	Correct	??????

Errors in the Judicial System

	Innocent	Guilty
Guilty Verdict	Type 1	Correct
Not Guilty Verdict	Correct	Type 2

Back to our experiment (flipping a coin)

Our hypotheses:

Back to our experiment (flipping a coin)

Our hypotheses:

1. H_0 the coin is fair ($p = 0.5$ that it lands Heads)

Back to our experiment (flipping a coin)

Our hypotheses:

1. H_0 the coin is fair ($p = 0.5$ that it lands Heads)
2. H_1 the coin is not fair ($p \neq 0.5$)

Back to our experiment (flipping a coin)

```
mu, sigma = normal_approx(1000, 0.5)
err = 0.05 # Our comfort with a type 1 error: 5%
lower, upper = norm_two_sided_bounds((1 - err), mu, sigma)
```

Back to our experiment (flipping a coin)

The result, with 95% probability:

Back to our experiment (flipping a coin)

The result, with 95% probability:

1. Lower ≈ 469 result in heads

Back to our experiment (flipping a coin)

The result, with 95% probability:

1. Lower ≈ 469 result in heads
2. Upper ≈ 531 result in heads

Back to our experiment (flipping a coin)

The result, with 95% probability:

1. Lower ≈ 469 result in heads
2. Upper ≈ 531 result in heads
3. What would we expect if the coin was fair?

Interpreting the results

Assuming the coin is fair

Interpreting the results

Assuming the coin is fair

1. Just a 5% chance that the number of heads we'd see lies outside this range

Interpreting the results

Assuming the coin is fair

1. Just a 5% chance that the number of heads we'd see lies outside this range
2. Have we *proven* anything?

Interpreting the results

Assuming the coin is fair

1. Just a 5% chance that the number of heads we'd see lies outside this range
2. Have we *proven* anything?
3. Are you convinced?

Interpreting the results

Assuming the coin is fair

1. Just a 5% chance that the number of heads we'd see lies outside this range
2. Have we *proven* anything?
3. Are you convinced?
4. If you're wrong you lose a limb, are you convinced now?

Interpreting the results

But *we* got to choose the significance! How seriously should we take these results?

Interpreting the results

But *we* got to choose the significance! How seriously should we take these results?

1. It is important that you communicate *why* you feel these results are valid.

Interpreting the results

But *we* got to choose the significance! How seriously should we take these results?

1. It is important that you communicate *why* you feel these results are valid.
2. It is *very easy* to lie with statistics:

Interpreting the results

But *we* got to choose the significance! How seriously should we take these results?

1. It is important that you communicate *why* you feel these results are valid.
2. It is *very easy* to lie with statistics:
   2.1 Imagine if H_0 was not in the 95% range, but in the 96% range

Interpreting the results

But *we* got to choose the significance! How seriously should we take these results?

1. It is important that you communicate *why* you feel these results are valid.
2. It is *very easy* to lie with statistics:
   2.1 Imagine if H_0 was not in the 95% range, but in the 96% range
   2.2 Why is 5% special?

p-Values

We computed *bounds* based on some chosen probability,
p-values flips this around:

p-Values

We computed *bounds* based on some chosen probability, *p-values* flips this around:

1. We assume H_0 is true.

p-Values

We computed *bounds* based on some chosen probability, *p-values* flips this around:

1. We assume H_0 is true.
2. We compute the probability that we would see a value *at least* as extreme as our actually observed value.

p-Values

Let's say we flipped a coin 1000 times (instead of having a distribution of such experiments)

p-Values

Let's say we flipped a coin 1000 times (instead of having a distribution of such experiments)

1. We observe 530 heads, this would give us a p-value of 6.2%

p-Values

Let's say we flipped a coin 1000 times (instead of having a distribution of such experiments)

1. We observe 530 heads, this would give us a p-value of 6.2%
2. We observe 532 heads, this would give us a p-value of 4.6%

p-Values

Let's say we flipped a coin 1000 times (instead of having a distribution of such experiments)

1. We observe 530 heads, this would give us a p-value of 6.2%
2. We observe 532 heads, this would give us a p-value of 4.6%
3. (The function for computing the p-values is in the notebook file)

Recap on the general problem

Many Machine Learning problems take the following form:

$$\text{minimize}_\theta \ \sum_{i=1}^{m} l(h_\theta(x^{(i)}), y^{(i)})$$

Recap on the general problem

Many Machine Learning problems take the following form:

$$\text{minimize}_\theta \sum_{i=1}^{m} l(h_\theta(x^{(i)}), y^{(i)})$$

We've now looked at some ls and an h.

Previously, on...

Hypothesis function

Previously, on...

Hypothesis function

1. We looked at a linear regression

Previously, on...

Hypothesis function

1. We looked at a linear regression
2. We 'fit' this linear regression to our dataset

Previously, on...

Hypothesis function

1. We looked at a linear regression
2. We 'fit' this linear regression to our dataset
3. If our data is actually linear, we also get *predictive* power

A wild h appears

Linear Regressions aren't the only possible hypothesis function!
We've also got:

A wild h appears

Linear Regressions aren't the only possible hypothesis function!
We've also got:

1. *Decision Trees* : 20-questions, the ML technique

A wild h appears

Linear Regressions aren't the only possible hypothesis function! We've also got:

1. *Decision Trees* : 20-questions, the ML technique
2. *Polynomials* : For when a straight line isn't cutting it

A wild h appears

Linear Regressions aren't the only possible hypothesis function! We've also got:

1. *Decision Trees* : 20-questions, the ML technique
2. *Polynomials* : For when a straight line isn't cutting it
3. *Neural networks* : What if we misunderstood neurons and made it a program?

A wild h appears

Linear Regressions aren't the only possible hypothesis function! We've also got:

1. *Decision Trees* : 20-questions, the ML technique
2. *Polynomials* : For when a straight line isn't cutting it
3. *Neural networks* : What if we misunderstood neurons and made it a program?
4. *Arbitrary Programs*: What is computers wrote the programs?

Do you realize?

A learning problem is said to be *realizable* if the true function exists within the learning problem's *hypothesis space*

1

Do you realize?

A learning problem is said to be *realizable* if the true function exists within the learning problem's *hypothesis space*

1. This means that the more expressive the hypothesis space (polynomials vs straight lines) the more likely that the problem is realizable.

1

Do you realize?

A learning problem is said to be *realizable* if the true function exists within the learning problem's *hypothesis space*

1. This means that the more expressive the hypothesis space (polynomials vs straight lines) the more likely that the problem is realizable.
2. What's the downside?

1

Do you realize?

A learning problem is said to be *realizable* if the true function exists within the learning problem's *hypothesis space*

1. This means that the more expressive the hypothesis space (polynomials vs straight lines) the more likely that the problem is realizable.
2. What's the downside?
3. Occam's[1] Razor is a data-scientist's best friend

[1] Also written as 'Ockham' or 'Ocham'

Decision Trees

We can view our tagged dataset (values of (x, tag)), as standing in for values of $(x, f(x))$.

Decision Trees

We can view our tagged dataset (values of (x, tag)), as standing in for values of $(x, f(x))$.

1. As with the linear regression the goal is to find an h that approximates f.

Decision Trees

We can view our tagged dataset (values of (x, tag)), as standing in for values of $(x, f(x))$.

1. As with the linear regression the goal is to find an h that approximates f.
2. But instead of a regression, we want a tree of *decisions*.

Decision Trees

We can view our tagged dataset (values of (x, tag)), as standing in for values of $(x, f(x))$.

1. As with the linear regression the goal is to find an h that approximates f.
2. But instead of a regression, we want a tree of *decisions*.
3. What's a decision?

Decisions! Decisions!

Each decision has two parts:

[2] not in the OO sense

Decisions! Decisions!

Each decision has two parts:

1. *Input* : An object[2] event/situation, that is described by a set of attributes (or *features*)

[2] not in the OO sense

Decisions! Decisions!

Each decision has two parts:

1. *Input* : An object[2] event/situation, that is described by a set of attributes (or *features*)
2. *Output*: A prediction of the 'value' based on the input

[2] not in the OO sense

Decisions! Decisions!

Each decision has two parts:

1. *Input* : An object[2] event/situation, that is described by a set of attributes (or *features*)
2. *Output*: A prediction of the 'value' based on the input
3. The boolean case (yes/no) is easy to visualize, but the values do not have to be discrete.

[2] not in the OO sense

Consider

You are asked to identify an animal based on a set of features (number of legs, weight, number of eyes, etc.)

Consider

You are asked to identify an animal based on a set of features (number of legs, weight, number of eyes, etc.)

1. The challenge is that the *order* of questions can matter!

Consider

You are asked to identify an animal based on a set of features (number of legs, weight, number of eyes, etc.)

1. The challenge is that the *order* of questions can matter!
2. You'll want the 'most significant' question first.

Consider

You are asked to identify an animal based on a set of features (number of legs, weight, number of eyes, etc.)

1. The challenge is that the *order* of questions can matter!
2. You'll want the 'most significant' question first.
3. Unfortunately, it can be very expensive(!!) to find the most significant question.

A tiny bit more formally:

A decision tree has two types of nodes:

A tiny bit more formally:

A decision tree has two types of nodes:

1. Decision nodes: Specifies a test on some attribute

A tiny bit more formally:

A decision tree has two types of nodes:

1. Decision nodes: Specifies a test on some attribute
2. Leaf node: A final classification/prediction

Small example:

We want to determine whether someone has ever seen an episode of Sponge Bob:

Small example:

We want to determine whether someone has ever seen an episode of Sponge Bob:

1. Are they older than 70: no.

Small example:

We want to determine whether someone has ever seen an episode of Sponge Bob:

1. Are they older than 70: no.
2. Are they older then 40: if yes...

Small example:

We want to determine whether someone has ever seen an episode of Sponge Bob:

1. Are they older than 70: no.
2. Are they older then 40: if yes...
   - 2.1 Do they have kids: if yes, yes.

Small example:

We want to determine whether someone has ever seen an episode of Sponge Bob:

1. Are they older than 70: no.
2. Are they older then 40: if yes...
   2.1 Do they have kids: if yes, yes.
   2.2 no.

Small example:

We want to determine whether someone has ever seen an episode of Sponge Bob:

1. Are they older than 70: no.
2. Are they older then 40: if yes...
   2.1 Do they have kids: if yes, yes.
   2.2 no.
3. Are they older than 4: yes.

Small example:

We want to determine whether someone has ever seen an episode of Sponge Bob:

1. Are they older than 70: no.
2. Are they older then 40: if yes...
   2.1 Do they have kids: if yes, yes.
   2.2 no.
3. Are they older than 4: yes.
4. Do they have older siblings: yes.

Small example:

We want to determine whether someone has ever seen an episode of Sponge Bob:

1. Are they older than 70: no.
2. Are they older then 40: if yes...
   2.1 Do they have kids: if yes, yes.
   2.2 no.
3. Are they older than 4: yes.
4. Do they have older siblings: yes.
5. no.

Oof

Even for such a small example, it starts getting unwieldy.

Oof

Even for such a small example, it starts getting unwieldy.

1. Luckily, libraries will be able to display trees nicely

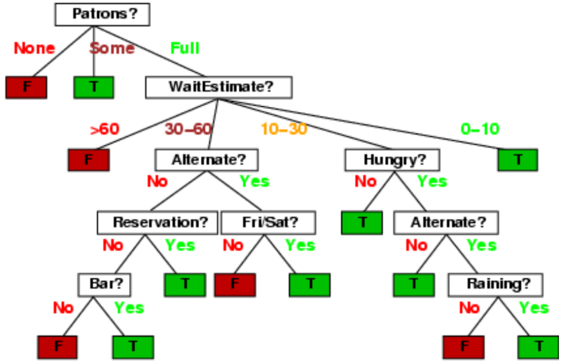
Oof

Even for such a small example, it starts getting unwieldy.

1. Luckily, libraries will be able to display trees nicely
2. For many trees it's not necessarily true that each 'decision', will have a meaningful-in-English question associated with it.

A prettier example

Should we wait for a table?



How many are there?

Decision Trees can encode arbitrary boolean functions.

How many are there?

Decision Trees can encode arbitrary boolean functions.

1. Each attribute can be 0/1

How many are there?

Decision Trees can encode arbitrary boolean functions.

1. Each attribute can be 0/1
    1.1 So our *input* space is 2^N

How many are there?

Decision Trees can encode arbitrary boolean functions.

1. Each attribute can be 0/1
   1.1 So our *input* space is 2^N
2. Each decision value can be 0/1, *for each possible combination of features!*

How many are there?

Decision Trees can encode arbitrary boolean functions.

1. Each attribute can be 0/1
   1.1 So our *input* space is 2^N
2. Each decision value can be 0/1, *for each possible combination of features!*
   2.1 So our *hypothesis space* is 2^{2^N}

Basic Algorithm

The goal is to find a *small* tree that correctly predicts the training samples

Basic Algorithm

The goal is to find a *small* tree that correctly predicts the training samples

1. Choose the “most significant” attribute

Basic Algorithm

The goal is to find a *small* tree that correctly predicts the training samples

1. Choose the "most significant" attribute
2. Once you make a choice for "most significant", you don't backtrack (greedy)

Basic Algorithm

The goal is to find a *small* tree that correctly predicts the training samples

1. Choose the "most significant" attribute
2. Once you make a choice for "most significant", you don't backtrack (greedy)
3. Now you've split your dataset, repeat the process for each subset.

Significant?

How do we pick the "most significant"?

Significant?

How do we pick the "most significant"?

1. We can't always :(

Significant?

How do we pick the "most significant"?

1. We can't always :(
2. We want to try and maximize *information gain*

Significant?

How do we pick the “most significant”?

1. We can’t always :(
2. We want to try and maximize *information gain*
3. For this class: let the libraries do the work for you.

Thanks for your time!

:)