Machine Learning - Lecture 17
Minimum Description Length (MDL NFL)

Chris Thornton

November 29, 2011

Introduction

Up to a point, machine learning is a ‘solved problem’.

To obtain a model that lets us predict elements of a dataset, we just have to make sure we’re using a method with an appropriate bias.

The method must be able to identify and build a model of salient patterns in the data.

If things are taking too long, we need to make the bias stronger.

Variable independence is all important

Most ML methods rely on input values having an independent relationship with output values.

With the Naïve Bayes Classifier, this independence is required.

But even with methods like k-means, it is vitally important.

If variables are *interdependent* in some way, we won't see particular classes associating with particular input values.

So, no reason to expect certain classes to occupy certain areas of the data space.

Methods which try to model shapes or areas cannot succeed.

Can we ignore the problem?

We can ignore the problem provided the dataset we're interested in guarantees variable independence.

Fortunately, the majority of datasets used in the field are of this type (cf. the UCI repository of machine learning databases).

But there are two ways the assumption can break down.

Can we ignore the problem?

We can ignore the problem provided the dataset we're interested in guarantees variable independence.

Fortunately, the majority of datasets used in the field are of this type (cf. the UCI repository of machine learning databases).

But there are two ways the assumption can break down.

- We may be dealing with an explicitly *relational* problem.

Can we ignore the problem?

We can ignore the problem provided the dataset we're interested in guarantees variable independence.

Fortunately, the majority of datasets used in the field are of this type (cf. the UCI repository of machine learning databases).

But there are two ways the assumption can break down.

- We may be dealing with an explicitly *relational* problem.
- We may be dealing with raw data (e.g., video feed.)

Can we ignore the problem?

We can ignore the problem provided the dataset we're interested in guarantees variable independence.

Fortunately, the majority of datasets used in the field are of this type (cf. the UCI repository of machine learning databases).

But there are two ways the assumption can break down.

- We may be dealing with an explicitly *relational* problem.
- We may be dealing with raw data (e.g., video feed.)

In both cases, there is no reason to expect independent association between input variables and class variables.

(This is why you don't see robots applying machine learning methods to sensor data.)

The checkerboard effect

With greater dependency between variable values, we expect less connection between specific values and specific classifications.

The worst-case scenario is where there is *no* connection at all.

In a 2d, two-class dataset, this scenario produces the effect of a **checkerboard**.

1	0	1	0
0	1	0	1
1	0	1	0
0	1	0	1

If we try to model this dataset in terms of contiguous areas, we end up with some form of lookup table and no useful generalization.

The XOR case

An extreme case of the effect occurs when we try to model the two classes in terms of two areas separated by a line.

This can't be done.

1	**0**
0	**1**

In Minsky and Papert's work, this example was used as a way of showing that Perceptrons cannot learn the boolean function exclusive-or (XOR).

Dealing with variable dependency

As variable dependency increases, classes cease to occupy specific areas in the data space.

Standard supervised methods start to fail.

How can we deal with this?

Dealing with variable dependency

As variable dependency increases, classes cease to occupy specific areas in the data space.

Standard supervised methods start to fail.

How can we deal with this?

- *Logical approach:* Assume the areas have disappeared due to salient patterns being inherently relational. Deploy some appropriately relational learning method.

Dealing with variable dependency

As variable dependency increases, classes cease to occupy specific areas in the data space.

Standard supervised methods start to fail.

How can we deal with this?

- *Logical approach:* Assume the areas have disappeared due to salient patterns being inherently relational. Deploy some appropriately relational learning method.
- *Optimistic approach:* Assume it's not that the areas have disappeared, it's just that they've become more complex. Come up with a method able to identify and model more complex areas.

Dealing with variable dependency

As variable dependency increases, classes cease to occupy specific areas in the data space.

Standard supervised methods start to fail.

How can we deal with this?

- *Logical approach:* Assume the areas have disappeared due to salient patterns being inherently relational. Deploy some appropriately relational learning method.
- *Optimistic approach:* Assume it's not that the areas have disappeared, it's just that they've become more complex. Come up with a method able to identify and model more complex areas.
- *Creative approach:* Just accept they're not there. Come up with a whole new approach...

Dealing with variable dependency

As variable dependency increases, classes cease to occupy specific areas in the data space.

Standard supervised methods start to fail.

How can we deal with this?

- *Logical approach:* Assume the areas have disappeared due to salient patterns being inherently relational. Deploy some appropriately relational learning method.
- *Optimistic approach:* Assume it's not that the areas have disappeared, it's just that they've become more complex. Come up with a method able to identify and model more complex areas.
- *Creative approach:* Just accept they're not there. Come up with a whole new approach...

Patterns that are not types of area

Methods that rely on variable independence have a built-in bias towards patterns of a *spatial* form.

Can we replace this with something more general?

Could there be a bias that is *universal*, i.e., correct in all cicrumstances?

The No Free Lunch (NFL) theorem

An interesting theoretical result suggests there cannot be a universally correct bias for supervised learning.

This is the so-called **No Free Lunch** theorem.

The No Free Lunch (NFL) theorem

An interesting theoretical result suggests there cannot be a universally correct bias for supervised learning.

This is the so-called **No Free Lunch** theorem.

- Assume there *is* a such a bias.

The No Free Lunch (NFL) theorem

An interesting theoretical result suggests there cannot be a universally correct bias for supervised learning.

This is the so-called **No Free Lunch** theorem.

- Assume there *is* a such a bias.
- It must yield effective generalization in all possible scenarios.

The No Free Lunch (NFL) theorem

An interesting theoretical result suggests there cannot be a universally correct bias for supervised learning.

This is the so-called **No Free Lunch** theorem.

- Assume there *is* a such a bias.
- It must yield effective generalization in all possible scenarios.
- But if we assemble the set of *all* supervised problems for a particular domain, we find that the performance of any single method averages-out to a chance-level of performance.

The No Free Lunch (NFL) theorem

An interesting theoretical result suggests there cannot be a universally correct bias for supervised learning.

This is the so-called **No Free Lunch** theorem.

- Assume there *is* a such a bias.
- It must yield effective generalization in all possible scenarios.
- But if we assemble the set of *all* supervised problems for a particular domain, we find that the performance of any single method averages-out to a chance-level of performance.

Proof

If the method scores above chance on a certain dataset, there must be a complementary dataset in which we see all the output values reversed. The method will score below the chance-level to exactly the same degree on this dataset.

Averaged over the set, the method produces chance-level generalization.

This seems to suggest there cannot can a universal bias for supervised learning.

What about unsupervised learning?

This leaves the possibility that there might be a universal bias for *unsupervised* learning.

One idea that has been popular over the years is that **simplicity** may be the vital factor.

This makes modeling more like explanation.

Instead of it being the process of looking for specific patterns, we see it as the task of finding the simplest way of representing the data.

Simpler models should generalize better

Under this approach, the focus on patterns is replaced with a stress on **parsimony**.

The aim of modeling is to uncover the simpler, implicit structures that underlie the more complex surface structures of the data.

This should always have a beneficial effect on prediction.

Because relatively simpler models make relatively fewer assumptions, there are relatively fewer ways in which their predictions can go wrong.

Machine learning as data compression

Various advantages in seeing (unsupervised) machine learning as the process of providing a dataset with a simpler representation.

It highlights the connection between machine learning and data compression.

In fact, data compression has the exact, same goal.

But the target is not better prediction. It is a smaller encoding of the data.

Data compression concepts

Two main forms of data compression:

Data compression concepts

Two main forms of data compression:

- **Lossless compression**. This is where it's possible to fully recover the original data from the model.

Data compression concepts

Two main forms of data compression:

- **Lossless compression**. This is where it's possible to fully recover the original data from the model.
- **Lossy compression**. Where it isn't.

Data compression concepts

Two main forms of data compression:

- **Lossless compression**. This is where it's possible to fully recover the original data from the model.
- **Lossy compression**. Where it isn't.

Compression ratio

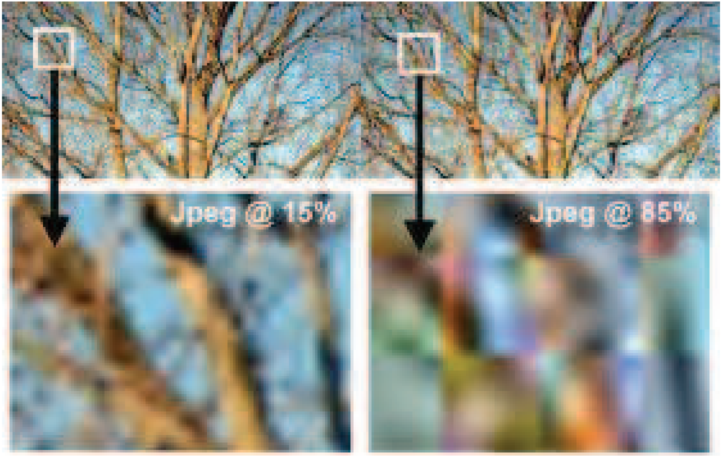
Performance is measured in terms of the compression ratio achieved.

$$\text{Compression ratio} = \frac{\text{Size of model}}{\text{Size of data}}$$

Normally sizes are measured in bits or bytes.

JPEG illustration

JPEG compression is a well-known compression-based image format.



Minimum description length

Simplification is also at the heart of the **Minimum Description Length** (MDL) approach to unsupervised learning.

Typically, MDL uses two-part codes, in which one part is the index of a model class, and the other is some data that configures the model.

The aim is to find the shortest two-part code that models the relevant dataset.

One difficulty is that the approach doesn't provide any general theory of how the codes are obtained.

Another is the fact that the task of finding the shortest code is uncomputable.

Kolmogorov Complexity

The Kolmogorov complexity of a dataset is defined to be the size of the *smallest* computer program that generates the dataset.

In this definition, computer programs are deemed to be the modeling language.

The idea leads to a formation definition of randomness:

Kolmogorov Complexity

The Kolmogorov complexity of a dataset is defined to be the size of the *smallest* computer program that generates the dataset.

In this definition, computer programs are deemed to be the modeling language.

The idea leads to a formation definition of randomness:

- A random dataset is a dataset whose Kolmogorov Complexity is equal to its size.

Kolmogorov Complexity

The Kolmogorov complexity of a dataset is defined to be the size of the *smallest* computer program that generates the dataset.

In this definition, computer programs are deemed to be the modeling language.

The idea leads to a formation definition of randomness:

- A random dataset is a dataset whose Kolmogorov Complexity is equal to its size.

This only happens if there is no underlying implicit structure. That only happens if the data are truly random.

Unfortunately, we still have no way of establishing the smallest generating program for any given dataset, so cannot measure complexity in practice.

Research continues...

Although there is considerable interest in the idea of simplicity-driven unsupervised learning, we still lack a robust, context-free implementation.

We don't know how well the approach would deal with the problem of variable dependencies.

But there is every reason to think that simplicity is a key issue for development of knowledge.

Occam’s Razor

After all, the idea has long been considered a fundamental objective for scientific research.

The principle of **Occam’s razor** (attributed to the 14th Century logician William of Ockham) states that ‘entities should not be multipled beyond necessity’.

This is essentially the the KISS principles (keep it simple stupid). It’s the same idea we see in MDL.

Simpler theories produce better predictions and thus better explanations.

Data compression, modeling, explanation, machine learning and discovery are all forms of simplification.

.

Summary

Summary

- Assumption of variable independence

Summary

- Assumption of variable independence
- Spatial bias

Summary

- Assumption of variable independence
- Spatial bias
- Checkerboard scenarios

Summary

- Assumption of variable independence
- Spatial bias
- Checkerboard scenarios
- No Free Lunch theorem

Summary

- Assumption of variable independence
- Spatial bias
- Checkerboard scenarios
- No Free Lunch theorem
- Lossy v. lossless data compression

Summary

- Assumption of variable independence
- Spatial bias
- Checkerboard scenarios
- No Free Lunch theorem
- Lossy v. lossless data compression
- Minimum Description Length

Summary

- Assumption of variable independence
- Spatial bias
- Checkerboard scenarios
- No Free Lunch theorem
- Lossy v. lossless data compression
- Minimum Description Length
- Kolmogorov Complexity

Summary

- Assumption of variable independence
- Spatial bias
- Checkerboard scenarios
- No Free Lunch theorem
- Lossy v. lossless data compression
- Minimum Description Length
- Kolmogorov Complexity
- Occam's Razor

Summary

- Assumption of variable independence
- Spatial bias
- Checkerboard scenarios
- No Free Lunch theorem
- Lossy v. lossless data compression
- Minimum Description Length
- Kolmogorov Complexity
- Occam's Razor

Questions

Questions

- According to Kolmogorov complexity theory, we can ignore choice of programming language in calculating the complexity value. Given a sufficiently large amount of data, use of a particular language simply imposes an ‘additive constant’. Describe the effect that is likely to occur with small amounts of data.

Questions

- According to Kolmogorov complexity theory, we can ignore choice of programming language in calculating the complexity value. Given a sufficiently large amount of data, use of a particular language simply imposes an ‘additive constant’. Describe the effect that is likely to occur with small amounts of data.
- Could there be any way around the NFL demonstration? In other words, could there be some kind of universal bias for supervised learning?

Questions

- According to Kolmogorov complexity theory, we can ignore choice of programming language in calculating the complexity value. Given a sufficiently large amount of data, use of a particular language simply imposes an ‘additive constant’. Describe the effect that is likely to occur with small amounts of data.
- Could there be any way around the NFL demonstration? In other words, could there be some kind of universal bias for supervised learning?
- JPEG encoding incorporates a scalable compression parameter for creating lossy encodings of images. How does this compression process work?

Questions

- According to Kolmogorov complexity theory, we can ignore choice of programming language in calculating the complexity value. Given a sufficiently large amount of data, use of a particular language simply imposes an ‘additive constant’. Describe the effect that is likely to occur with small amounts of data.
- Could there be any way around the NFL demonstration? In other words, could there be some kind of universal bias for supervised learning?
- JPEG encoding incorporates a scalable compression parameter for creating lossy encodings of images. How does this compression process work?
- Unix ‘compress’ implements a form of ‘Lempel-Ziv-Welch’ (LZW) compression. Is this lossy or lossless? How does the encoding work?

Questions

- According to Kolmogorov complexity theory, we can ignore choice of programming language in calculating the complexity value. Given a sufficiently large amount of data, use of a particular language simply imposes an ‘additive constant’. Describe the effect that is likely to occur with small amounts of data.
- Could there be any way around the NFL demonstration? In other words, could there be some kind of universal bias for supervised learning?
- JPEG encoding incorporates a scalable compression parameter for creating lossy encodings of images. How does this compression process work?
- Unix ‘compress’ implements a form of ‘Lempel-Ziv-Welch’ (LZW) compression. Is this lossy or lossless? How does the encoding work?

Further reading

Further reading

- Wolpert, D. H. (1996). The existence of a priori distinctions between learning algorithms. *Neural Computation*, *8*, No. 7.

Further reading

- Wolpert, D. H. (1996). The existence of a priori distinctions between learning algorithms. *Neural Computation*, *8*, No. 7.
- Wolpert, D. H. (1996). The lack of a priori distinctions between learning algorithms. *Neural Computation*, *8*, No. 7 (pp. 1341-1390).

Further reading

- Wolpert, D. H. (1996). The existence of a priori distinctions between learning algorithms. *Neural Computation*, *8*, No. 7.
- Wolpert, D. H. (1996). The lack of a priori distinctions between learning algorithms. *Neural Computation*, *8*, No. 7 (pp. 1341-1390).
- Wolpert, D. H. and Macready, W. G. (1997). No free lunch theorems for optimization. *IEEE Transactions on Evolutionary Computing*, *1* (pp. 67-82).

Further reading

- Wolpert, D. H. (1996). The existence of a priori distinctions between learning algorithms. *Neural Computation*, *8*, No. 7.
- Wolpert, D. H. (1996). The lack of a priori distinctions between learning algorithms. *Neural Computation*, *8*, No. 7 (pp. 1341-1390).
- Wolpert, D. H. and Macready, W. G. (1997). No free lunch theorems for optimization. *IEEE Transactions on Evolutionary Computing*, *1* (pp. 67-82).