Information Theory

http://www.inf.ed.ac.uk/teaching/courses/it/

Week 5
Models for stream codes

Iain Murray, 2013
School of Informatics, University of Edinburgh

Card prediction

3 cards with coloured faces:

1. one white and one black face
2. two black faces
3. two white faces

I shuffle cards and turn them over randomly. I select a card and way-up uniformly at random and place it on a table.

Question: You see a black face. What is the probability that the other side of the same card is white?

$$P(x_2{=}\mathtt{W}\,|\,x_1{=}\mathtt{B}) = \;{}^1\!/\!_3,\quad {}^1\!/\!_2,\quad {}^2\!/\!_3,\quad \text{other?}$$

Notes on the card prediction problem:

This card problem is Ex. 8.10a), MacKay, p142.

It is *not* the same as the famous 'Monty Hall' puzzle: Ex. 3.8–9 and `http://en.wikipedia.org/wiki/Monty_Hall_problem`

The Monty Hall problem is also worth understanding. Although the card problem is (hopefully) less controversial and more straightforward. The process by which a card is selected should be clear: $P(c) = 1/3$ for $c = 1, 2, 3$, and the face you see first is chosen at random: e.g., $P(x_1{=}\mathtt{B}|c{=}1) = 0.5$.

Many people get this puzzle wrong on first viewing (it's easy to mess up). We'll check understanding again with another prediction problem in a tutorial exercise. If you do get the answer right immediately (are you sure?), this is will be a simple example on which to demonstrate some formalism.

How do we solve it formally?

Use Bayes rule?

$$P(x_2{=}\mathtt{W}\,|\,x_1{=}\mathtt{B}) \;=\; \frac{\boxed{P(x_1{=}\mathtt{B}\,|\,x_2{=}\mathtt{W})}\;P(x_2{=}\mathtt{W})}{P(x_1{=}\mathtt{B})}$$

The $\boxed{\text{boxed}}$ term is no more obvious than the answer!

Bayes rule is used to 'invert' forward generative processes that we understand.

The first step to solve inference problems is to write down a model of your data.

The card game model

Cards: 1) $\mathtt{B}|\mathtt{W}$, 2) $\mathtt{B}|\mathtt{B}$, 3) $\mathtt{W}|\mathtt{W}$

$$P(c) = \begin{cases} {}^1\!/\!_3 & c = 1,2,3 \\ 0 & \text{otherwise.} \end{cases}$$

$$P(x_1{=}\mathtt{B}\,|\,c) = \begin{cases} {}^1\!/\!_2 & c = 1 \\ 1 & c = 2 \\ 0 & c = 3 \end{cases}$$

Bayes rule can 'invert' this to tell us $P(c\,|\,x_1{=}\mathtt{B})$;
infer the generative process for the data we have.

Inferring the card

Cards: 1) $\mathtt{B}|\mathtt{W}$, 2) $\mathtt{B}|\mathtt{B}$, 3) $\mathtt{W}|\mathtt{W}$

$$P(c\,|\,x_1{=}\mathtt{B}) = \frac{P(x_1{=}\mathtt{B}\,|\,c)\,P(c)}{P(x_1{=}\mathtt{B})}$$

$$\propto \begin{cases} {}^1\!/\!_2 \cdot {}^1\!/\!_3 = {}^1\!/\!_6 & c = 1 \\ 1 \cdot {}^1\!/\!_3 = {}^1\!/\!_3 & c = 2 \\ 0 & c = 3 \end{cases}$$

$$= \begin{cases} {}^1\!/\!_3 & c = 1 \\ {}^2\!/\!_3 & c = 2 \end{cases}$$

Q *"But aren't there two options given a black face, so it's 50–50?"*
A There are two options, but the likelihood for one of them is 2× bigger

Predicting the next outcome

For this problem we can spot the answer, for more complex problems we want a formal means to proceed.

$P(x_2\,|\,x_1{=}\mathtt{B})$?

Need to introduce c to use expressions we know:

$$\begin{aligned} P(x_2\,|\,x_1{=}\mathtt{B}) &= \sum_{c\in 1,2,3} P(x_2, c\,|\,x_1{=}\mathtt{B}) \\ &= \sum_{c\in 1,2,3} P(x_2\,|\,x_1{=}\mathtt{B}, c)\, P(c\,|\,x_1{=}\mathtt{B}) \end{aligned}$$

Predictions we would make if we knew the card, weighted by the posterior probability of that card.

$P(x_2{=}\mathtt{W}\,|\,x_1{=}\mathtt{B}) = {}^1\!/\!_3$

Strategy for solving inference and prediction problems:

When interested in something y, we often find we can't immediately write down mathematical expressions for $P(y\,|\,\text{data})$.

So we introduce stuff, z, that helps us define the problem:

$$P(y\,|\,\text{data}) = \textstyle\sum_z P(y, z\,|\,\text{data})$$

by using the sum rule. And then split it up:

$$P(y\,|\,\text{data}) = \textstyle\sum_z P(y\,|\,z, \text{data})\, P(z\,|\,\text{data})$$

using the product rule. If knowing extra stuff z we can predict y, we are set: weight all such predictions by the posterior probability of the stuff ($P(z\,|\,\text{data})$, found with Bayes rule).

Sometimes the extra stuff summarizes everything we need to know to make a prediction:

$$P(y\,|\,z, \text{data}) = P(y\,|\,z)$$

although not in the card game above.

Not convinced?

Not everyone believes the answer to the card game question.

Sometimes probabilities are counter-intuitive. I'd encourage you to write simulations of these games if you are at all uncertain. Here is an Octave/Matlab simulator I wrote for the card game question:

```
cards = [1 1;
         0 0;
         1 0];
num_cards = size(cards, 1);

N = 0; % Number of times first face is black
kk = 0; % Out of those, how many times the other side is white

for trial = 1:1e6
    card = ceil(num_cards * rand());
    face = 1 + (rand < 0.5);
    other_face = (face==1) + 1;
    x1 = cards(card, face);
    x2 = cards(card, other_face);

    if x1 == 0
        N = N + 1;
        kk = kk + (x2 == 1);
    end
end

approx_probability = kk / N
```

Sparse files

$\mathbf{x} = \texttt{000010000100000001000...000}$

We are interested in predicting the $(N{+}1)$th bit.

Generative model:

$$\begin{aligned} P(\mathbf{x}\,|\,f) &= \prod_i P(x_i\,|\,f) = \prod_i f^{x_i}(1-f)^{1-x_i} \\ &= f^k(1-f)^{N-k}, \qquad k = \sum_i x_i = \text{“\# 1s”} \end{aligned}$$

Can ‘invert’, find $p(f\,|\,\mathbf{x})$ with Bayes rule

Inferring $f = P(x_i{=}1)$

Cannot do inference without using beliefs

A possible expression of uncertainty: $p(f) = 1, \quad f \in [0,1]$

Bayes rule:

$$\begin{aligned} p(f\,|\,\mathbf{x}) &\propto P(\mathbf{x}\,|\,f)\,p(f) \\ &\propto f^k(1-f)^{N-k} \\ &= \mathrm{Beta}(f;\ k{+}1,\ N{-}k{+}1) \end{aligned}$$

Beta distribution:

$\mathrm{Beta}(f;\alpha,\beta) = \frac{1}{B(\alpha,\beta)} f^{\alpha-1}(1-f)^{\beta-1} = \frac{\Gamma(\alpha+\beta)}{\Gamma(\alpha)\Gamma(\beta)} f^{\alpha-1}(1-f)^{\beta-1}$

Mean: $\alpha/(\alpha+\beta)$

The $\mathrm{Beta}(\alpha,\beta)$ **distribution** is a standard probability distribution over a variable $f \in [0,1]$ with parameters α and β.

The dependence of the probability density on f is summarized by: $\mathrm{Beta}(f;\alpha,\beta) \propto f^{(\alpha-1)}(1-f)^{(\beta-1)}$.
The $1/B(\alpha,\beta) = \Gamma(\alpha+\beta)/(\Gamma(\alpha)\Gamma(\beta))$ term, which is $(\alpha+\beta-1)!/((\alpha-1)!(\beta-1)!)$ for integer α and β, makes the distribution normalized (integrate to one). Here, it’s just some constant with respect to the parameter f.

For comparison, perhaps you are more familiar with a Gaussian (or Normal), $\mathcal{N}(\mu,\sigma^2)$, a probability distribution over a variable $x \in [-\infty,\infty]$, with parameters μ and σ^2.

The dependence of the probability density on x is summarized by $\mathcal{N}(x;\mu,\sigma^2) \propto \exp(-0.5(x-\mu)^2/\sigma^2)$.
We divide this expression by $\int \exp(-0.5(x-\mu)^2/\sigma^2)\,\mathrm{d}x = \sqrt{2\pi\sigma^2}$ to make the distribution normalized.

We found that our posterior beliefs, given observations, are proportional to $f^k(1-f)^{N-k}$ and we know $f \in [0,1]$. Given the form of the f dependence, the posterior distribution must be a Beta distribution. We obtain the parameters α and β by comparing the powers of f and $(1{-}f)$ in the posterior and in the definition of the Beta distribution. Comparing terms and reading off the answer is easier than doing integration to normalize the distribution from scratch, as in equations 3.11 and 3.12 of MacKay, p52.

Again for comparison: if you were trying to infer a real-valued parameter $y \in [-\infty,\infty]$, and wrote down a posterior: $p(y\,|\,D) \propto p(D\,|\,y)\,p(y)$ and found $p(y\,|\,D) \propto \exp(-ay^2+by)$ for some constants a and b, you could state that $p(y\,|\,D) = \exp(-ay^2+by)/Z$, and derive that the constant must be $Z = \int \exp(-ay^2+by)\mathrm{d}y = \ldots$. Alternatively, you could realize that a quadratic form inside an exp is a Gaussian distribution. Now you just have identify its parameters. As $\mathcal{N}(y;\mu,\sigma^2) \propto \exp(-0.5(y-\mu)^2/\sigma^2) \propto \exp(-0.5y^2/\sigma^2 + (\mu/\sigma^2)y)$, we can identify the parameters of the posterior: $\sigma^2 = 1/(2a)$ and $\mu = b\sigma^2 = b/(2a)$.

References on inferring a probability

The ‘bent coin’ is discussed in MacKay §3.2, p51

See also Ex. 3.15, p59, which has an extensive worked solution.

The MacKay section mentions that this problem is the one studied by Thomas Bayes, published in 1763. This is true, although the problem was described in terms of a game played on a Billiard table.

The Bayes paper has historical interest, but without modern mathematical notation takes some time to read. Several versions can be found around the web. The original version has old-style typesetting. The paper was retypeset, but with the original long arguments, for Biometrica in 1958:
`http://dx.doi.org/10.1093/biomet/45.3-4.296`

Prediction

Prediction rule from marginalization and product rules:

$$P(x_{N+1}\,|\,\mathbf{x}) = \int P(x_{N+1}\,|\,f, \boxed{\mathbf{x}}) \cdot p(f\,|\,\mathbf{x})\;\mathrm{d}f$$

The $\boxed{\text{boxed}}$ dependence can be omitted here.

$$P(x_{N+1}{=}1\,|\,\mathbf{x}) = \int f \cdot p(f\,|\,\mathbf{x})\;\mathrm{d}f = \mathbb{E}_{p(f\,|\,\mathbf{x})}[f] = \frac{k+1}{N+2}.$$

Laplace’s law of succession

$$P(x_{N+1}{=}1\,|\,\mathbf{x}) = \frac{k+1}{N+2}$$

Maximum Likelihood (ML): $\hat{f} = \mathrm{argmax}_f\, P(\mathbf{x}\,|\,f) = \frac{k}{N}$.
ML estimate is *unbiased*: $\mathbb{E}[\hat{f}] = f$.

Laplace’s rule is like using the ML estimate, but imagining we saw a 0 and a 1 before starting to read in $\mathbf{x}$.

Laplace’s rule biases probabilities towards $^1\!/_2$.

ML estimate assigns zero probability to unseen symbols.
Encoding zero-probability symbols needs ∞ bits.

New prior / prediction rule

Could use a Beta prior distribution:

$$p(f) \;=\; \mathrm{Beta}(f;\ n_1,\ n_0)$$

$$\begin{aligned} p(f\,|\,\mathbf{x}) &\propto f^{k+n_1-1}\,(1-f)^{N-k+n_0-1} \\ &= \mathrm{Beta}(f;\ k{+}n_1,\ N{-}k{+}n_0) \end{aligned}$$

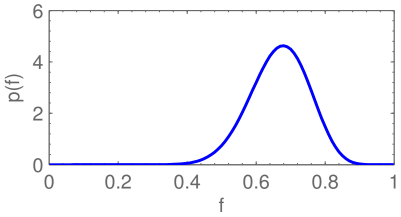
$$P(x_{N+1}{=}1\,|\,\mathbf{x}) \;=\; \mathbb{E}_{p(f\,|\,\mathbf{x})}[f] \;=\; \frac{k+n_1}{N+n_0+n_1}$$

Think of n_1 and n_0 as previously observed counts

($n_1{=}n_0{=}1$ gives uniform prior and Laplace’s rule)

Large pseudo-counts

Beta(20,10) distribution:



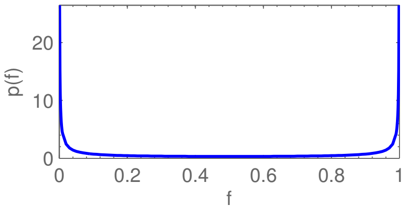
Mean: $^2\!/_3$

This prior says f close to 0 and 1 are very improbable

We’d need $\gg 30$ observations to change our mind
(to over-rule the prior, or pseudo-observations)

Fractional pseudo-counts

Beta(0.2,0.2) distribution:



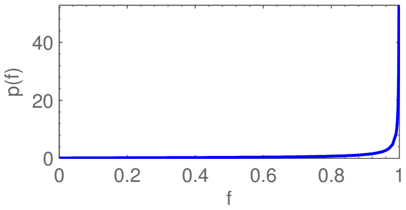
Mean: $1/2$ — notice prior says more than a guess of $f = 1/2$

f is probably close to 0 or 1 but we don't know which yet

One observation will rapidly change the posterior

Fractional pseudo-counts

Beta(1.2,0.2) distribution:



Posterior from previous prior and observing a single 1

Larger alphabets

i.i.d. symbol model:

$$P(\mathbf{x}\,|\,\mathbf{p}) = \prod_i p_i^{k_i}, \qquad \text{where } k_i = \sum_n \mathbb{I}(x_n = a_i)$$

The k_i are counts for each symbol.

Dirichlet prior, generalization of Beta:

$$p(\mathbf{p}\,|\,\boldsymbol{\alpha}) = \mathrm{Dirichlet}(\mathbf{p};\ \boldsymbol{\alpha}) = \tfrac{\delta(1-\sum_i p_i)}{B(\boldsymbol{\alpha})} \prod_i p_i^{\alpha_i - 1}$$

Dirichlet predictions (Lidstone's law):

$$P(x_{N+1}\!=\!a_i\,|\,\mathbf{x}) = \frac{k_i + \alpha_i}{N + \sum_j \alpha_j}$$

Counts k_i are added to pseudo-counts α_i. All $\alpha_i\!=\!1$ gives Laplace's rule.

More notes on the Dirichlet distribution

The thing to remember is that a Dirichlet is proportional to $\prod_i p_i^{\alpha_i - 1}$

The posterior $p(\mathbf{p}\,|\,\mathbf{x},\boldsymbol{\alpha}) \propto P(\mathbf{x}\,|\,\mathbf{p})\,p(\mathbf{p}\,|\,\boldsymbol{\alpha})$ will then be Dirichlet with the α_i's increased by the observed counts.

Details (for completeness): $B(\boldsymbol{\alpha})$ is the Beta function $\frac{\prod_i \Gamma(\alpha_i)}{\Gamma(\sum_i \alpha_i)}$.

I left the $0 \le p_i \le 1$ constraints implicit. The $\delta(1 - \sum_i p_i)$ term constrains the distribution to the 'simplex', the region of a hyper-plane where $\sum_i p_i = 1$. But I can't omit this Dirac-delta, because it is infinite when evaluated at a valid probability vector(!).

The density over just the first $(I\!-\!1)$ parameters is finite, obtained by integrating out the last parameter:

$$p(\mathbf{p}_{j<I-1}) = \int p(\mathbf{p}\,|\,\boldsymbol{\alpha})\;\mathrm{d}p_I = \frac{1}{B(\boldsymbol{\alpha})}\,\big(1 - \textstyle\sum_{i=1}^{I-1} p_i\big)^{\alpha_I - 1} \displaystyle\prod_{i=1}^{I-1} p_i^{\alpha_i - 1}$$

There are no infinities, and the relation to the Beta distribution is now clearer, but the expression isn't as symmetric.

Reflection on Compression

Take any complete compressor.

If "incomplete" imagine an improved "complete" version.

Complete codes: $\sum_{\mathbf{x}} 2^{-\ell(\mathbf{x})} = 1$, $\mathbf{x}$ is whole input file

Interpretation: implicit $Q(\mathbf{x}) = 2^{-\ell(bx)}$

If we believed files were drawn from $P(\mathbf{x}) \neq Q(\mathbf{x})$ we would expect to do $D(P||Q)\!>\!0$ bits better by using $P(\mathbf{x})$.

Compression is the modelling of probabilities of files.

If we think our compressor should 'adapt', we are making a statement about the structure of our beliefs, $P(\mathbf{x})$.

Structure

For any distribution:

$$P(\mathbf{x}) = P(x_1) \prod_{n=2}^{N} P(x_n\,|\,\mathbf{x}_{<n})$$

For i.i.d. symbols: $P(x_n\!=\!a_i\,|\,\mathbf{p}) = p_i$

$$P(x_n\,|\,\mathbf{x}_{<n}) = \int P(\mathbf{x}_n\,|\,\mathbf{p})\,p(\mathbf{p}\,|\,\mathbf{x}_{<n})\;\mathrm{d}\mathbf{p}$$

$$P(x_n\!=\!a_i\,|\,\mathbf{x}_{<n}) = \mathbb{E}_{p(\mathbf{p}\,|\,\mathbf{x}_{<n})}[p_i]$$

we saw: easy-to-compute from counts with a Dirichlet prior.

i.i.d. assumption is often terrible: want different structure.

Even then, do we need to specify priors (like the Dirichlet)?

Why not just fit p?

Run over file → counts $\mathbf{k}$

Set $p_i = \frac{k_i}{N}$, (Maximum Likelihood, and obvious, estimator)

Save $(\mathbf{p}, \mathbf{x})$, $\mathbf{p}$ in a header, $\mathbf{x}$ encoded using $\mathbf{p}$

Simple? Prior-assumption-free?

Fitting cannot be optimal

When fitting, we never save a file $(\mathbf{p}, \mathbf{x})$ where

$$p_i \neq \frac{k_i(\mathbf{x})}{N}$$

Informally: we are encoding $\mathbf{p}$ twice

More formally: the code is incomplete

However, `gzip` and arithmetic coders are incomplete too, but they are still useful!

In some situations the fitting approach is very close to optimal

Fitting isn't that easy!

Setting $p_i = \frac{k_i}{N}$ is easy. **How do we encode the header?**

Optimal scheme depends on $p(\mathbf{p})$; **need a prior!**

What precision to send parameters?
Trade-off between header and message size.

Interesting models will have many parameters.
Putting them in a header could dominate the message.

Having both ends learn the parameters while {en,de}coding the file avoids needing a header.

For more (non-examinable) detail on these issues see MacKay p352–353

Richer models

Images are not bags of i.i.d. pixels
Text is not a bag of i.i.d. characters/words

(although many "Topic Models" get away with it!)

Less restrictive assumption than:

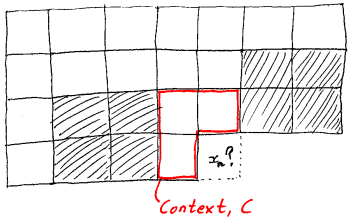
$$P(x_n \,|\, \mathbf{x}_{<n}) = \int P(\mathbf{x}_n \,|\, \mathbf{p})\, p(\mathbf{p} \,|\, \mathbf{x}_{<n})\; \mathrm{d}\mathbf{p}$$

is

$$P(x_n \,|\, \mathbf{x}_{<n}) = \int P(\mathbf{x}_n \,|\, \mathbf{p}_{C(\mathbf{x}_{<n})})\, p(\mathbf{p}_{C(\mathbf{x}_{<n})} \,|\, \mathbf{x}_{<n})\; \mathrm{d}\mathbf{p}_{C(\mathbf{x}_{<n})}$$

Probabilities depend on the local context, C:
— Surrounding pixels, already {en,de}coded
— Past few characters of text

Image contexts

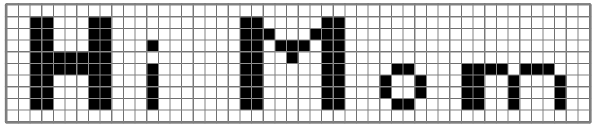


$$P(x_i\!=\!\texttt{Black} \,|\, C) = \frac{k_{\mathrm{B}|C} + \alpha}{N_C + \alpha|\mathcal{A}|} = \frac{2+\alpha}{7+2\alpha}$$

There are 2^p contexts of size p binary pixels
Many more counts/parameters than i.i.d. model

A good image model?

The context model isn't far off what several real image compression systems do for binary images.



With arithmetic coding we go from 500 to 220 bits

A better image model might do better

If we knew it was text and the font we'd need fewer bits!

Context size

How big to make the context?

`kids_make_nutr`$\boxed{?}$

Context length:

0: i.i.d. bag of characters
1: bigrams, give vowels higher probability
$>$**1:** predict using possible words
$\gg$**1:** use understanding of sentences?

Ideally we'd use really long contexts, as humans do.

Problem with large contexts

For simple counting methods, statistics are poor:

$$p(x_n = a_i \,|\, \mathbf{x}_{<n}) = \frac{k_{i|C} + \alpha}{N_C + \alpha|\mathcal{A}|}$$

$k_{\cdot|C}$ will be zero for most symbols in long contexts
Predictions become uniform ⇒ no compression.

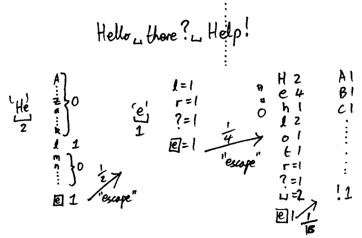
What broke? We believe some contexts are related:

`kids_make_nutr`$\boxed{?}$
`kids_like_nutr`$\boxed{?}$

while the Dirichlet prior says they're unrelated

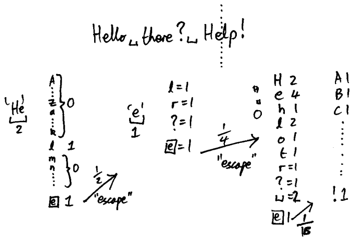
Prediction by Partial Match (PPM)

One way of smoothing predictions from several contexts:



Model: draw using fractions observed at context
Escape to shorter context with some probability (variant-dependent)

Prediction by Partial Match (PPM)



$$P(\texttt{l} \,|\, \texttt{Hello there? He}) = \tfrac{1}{2} + \tfrac{1}{2}\big(\tfrac{1}{4} + \tfrac{1}{4}\big(\tfrac{2}{16} + \tfrac{1}{16}\tfrac{1}{|\mathcal{A}|}\big)\big)$$

$$P(\texttt{!} \,|\, \texttt{Hello there? He}) = \tfrac{1}{2}\tfrac{1}{4}\tfrac{1}{16}\tfrac{1}{|\mathcal{A}|}$$

$$P(\texttt{\_} \,|\, \texttt{Hello there? He}) = \tfrac{1}{2}\big(\tfrac{1}{4}\big(\tfrac{2}{16} + \tfrac{1}{16}\tfrac{1}{|\mathcal{A}|}\big)\big)$$

Prediction by Partial Match comments

First PPM paper: Clearly and Witten (1984). Many variants since. The best PPM variant's text compression is now highly competitive. Although it is clearly possible to come up with better models of text.

The ideas are common to methods with several other names. PPM is a name used a lot in text compression for the combination of this type of model with arithmetic coding.

Other prediction methods and more advanced models

Better methods that smooth counts from different contexts:
`http://research.microsoft.com/pubs/64808/tr-10-98.pdf`

I covered Beta and Dirichlet priors to demonstrate that prediction rules can be derived from models. There isn't time in this course to take this idea further, but state-of-the-art predictions can result from Bayesian inference in more complicated *hierarchical* models:
`http://homepages.inf.ed.ac.uk/sgwater/papers/nips05.pdf`
`http://www.aclweb.org/anthology/P/P06/P06-1124.pdf`