Information-Theoretic Modeling

Lecture 7: Universal Source Coding

Teemu Roos

Department of Computer Science, University of Helsinki

Fall 2014

Lecture 8: Universal Source Coding

Moline Universal Model D, Little Casterton Working Weekend, 2006.

1. Universal Source Codes
   - Definitions
   - Universal Models
2. Two-Part Codes
   - Discrete Parameters
   - Continuous Parameters
   - Asymptotics: $\frac{k}{2} \log n$
3. Advanced Universal Codes
   - Mixture Codes
   - Normalized Maximum Likelihood
   - Universal Prediction

Definitions

Our basic setting is that we have some *data* $D = (x_1, \ldots, x_m)$ where the individual data points x_i come from some domain $\mathcal{X}$.

We write $\mathcal{D}$ for the set of all possible data. A typical situation is $\mathcal{D} = \mathcal{X}^n$ where n may or may not be known in advance.

A probability distribution p over $\mathcal{D}$ is called a *model*.

A set of models $\mathcal{M}$ is called a *model class*.

Model classes are often *parametric*: $\mathcal{M} = \{ p_\theta \mid \theta \in \Theta \}$ where p_θ is a model for each $\theta \in \Theta$, and $\Theta \subseteq \mathbb{R}^k$ for some k.

Definitions

Example: Gaussian model

Let p_{μ,σ^2} be the normal distribution over $\mathcal{X} = \mathbb{R}$ with mean μ and variance σ^2.

We have a parametric model class $\mathcal{M} = \{\, p_\theta \mid \theta \in \Theta \,\}$ where $\Theta = \{\, (\mu, \sigma^2) \in \mathbb{R}^2 \mid \sigma^2 > 0 \,\}$.

We can extend p_{μ,σ^2} into a distribution over $\mathcal{D} = \mathbb{R}^n$ by assuming independence: $p^{(n)}_{\mu,\sigma^2}(x_1, \ldots, x_n) = p_{\mu,\sigma^2}(x_1) \ldots p_{\mu,\sigma^2}(x_n)$.

We often abuse notation by just writing $p_\theta(x_1, \ldots, x_n)$ instead of $p^{(n)}_\theta(x_1, \ldots, x_n)$.

However, keep in mind that we may also have models that does not satisfy the independence assumption.

MDL Philosophy

It is good to keep in mind that we don't claim that we can find a "true" model p that really generates the data D, or even that such a "true" model exists.

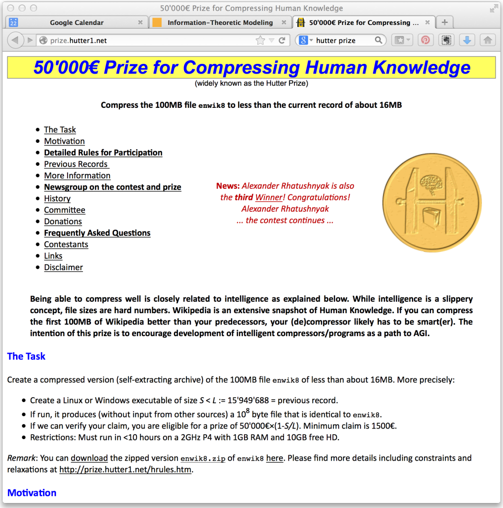
Instead, in the MDL philosophy is founded on the following informal claim.

Claim

The better a code based on model p can compress data D, the more regularities that pertain to D it exploits.

For example, think about the digits of $\pi = x_1.x_2x_3\ldots$.

Hutter Prize



50'000€ Prize for Compressing Human Knowledge

Google Calendar | Information-Theoretic Modeling | 50'000€ Prize for Compressing ...

prize.hutter1.net

hutter prize

50'000€ Prize for Compressing Human Knowledge

(widely known as the Hutter Prize)

Compress the 100MB file `enwik8` to less than the current record of about 16MB

- The Task
- Motivation
- **Detailed Rules for Participation**
- Previous Records
- More Information
- **Newsgroup on the contest and prize**
- History
- Committee
- Donations
- **Frequently Asked Questions**
- Contestants
- Links
- Disclaimer

News: *Alexander Rhatushnyak is also the* ***third*** *Winner! Congratulations!*
Alexander Rhatushnyak
... the contest continues ...

Being able to compress well is closely related to intelligence as explained below. While intelligence is a slippery concept, file sizes are hard numbers. Wikipedia is an extensive snapshot of Human Knowledge. If you can compress the first 100MB of Wikipedia better than your predecessors, your (de)compressor likely has to be smart(er). The intention of this prize is to encourage development of intelligent compressors/programs as a path to AGI.

The Task

Create a compressed version (self-extracting archive) of the 100MB file `enwik8` of less than about 16MB. More precisely:

- Create a Linux or Windows executable of size $S < L$:= 15'949'688 = previous record.
- If run, it produces (without input from other sources) a 10^8 byte file that is identical to `enwik8`.
- If we can verify your claim, you are eligible for a prize of 50'000€×(1-S/L). Minimum claim is 1500€.
- Restrictions: Must run in <10 hours on a 2GHz P4 with 1GB RAM and 10GB free HD.

Remark: You can download the zipped version `enwik8.zip` of `enwik8` here. Please find more details including constraints and relaxations at http://prize.hutter1.net/hrules.htm.

Motivation

Definitions

The model within $\mathcal{M}$ that achieves the shortest code-length for data D is the **maximum likelihood (ML) model**:

$$\min_{\theta \in \Theta} \log_2 \frac{1}{p_\theta(D)} = \log_2 \frac{1}{p_{\hat{\theta}}(D)} \ .$$

$p_{\hat{\theta}} = p_{\hat{\theta}(D)}$ depends on D!

For model q, the excess code-length or "**regret**" over the ML model in $\mathcal{M}$ is given by

$$\log_2 \frac{1}{q(D)} - \log_2 \frac{1}{p_{\hat{\theta}}(D)} \ .$$

Game-theoretic setting: Player chooses q first, then Nature chooses D. Player tries to keep regret small no matter what.

Universal models

Universal model

A model (code) whose regret grows slower than n, for all data sequences, is said to be a **universal model** (code) relative to model class $\mathcal{M}$:

$$\lim_{n\to\infty} \frac{1}{n} \max_{D\in\mathcal{D}} \left[\log_2 \frac{1}{q(D)} - \log_2 \frac{1}{p_{\hat{\theta}}(D)}\right] = 0 \ . \qquad (1)$$

Another (stochastic) definition of universality is

$$\frac{1}{n} D(p_\theta \,\|\, q) \to 0 \quad \text{for all } \theta \in \Theta. \qquad (2)$$

The second one is weaker since (1) $\Rightarrow$ (2). Proof.

Universal models

Universal model

A model (code) whose regret grows slower than n, for all data sequences, is said to be a **universal model** (code) relative to model class $\mathcal{M}$:

$$\lim_{n\to\infty} \frac{1}{n} \max_{D\in\mathcal{D}} \left[\log_2 \frac{1}{q(D)} - \log_2 \frac{1}{p_{\hat{\theta}}(D)}\right] = 0 \ . \qquad (1)$$

$$\log_2 \frac{1}{p_{\hat{\theta}}(D)} \leq \log_2 \frac{1}{p_{\theta}(D)}$$

Universal models

Universal model

A model (code) whose regret grows slower than n, for all data sequences, is said to be a **universal model** (code) relative to model class $\mathcal{M}$:

$$\lim_{n\to\infty} \frac{1}{n} \max_{D\in\mathcal{D}} \left[\log_2 \frac{1}{q(D)} - \log_2 \frac{1}{p_{\hat{\theta}}(D)}\right] = 0 \ . \qquad (1)$$

$$-\log_2 \frac{1}{p_{\hat{\theta}}(D)} \geq -\log_2 \frac{1}{p_\theta(D)}$$

Universal models

Universal model

A model (code) whose regret grows slower than n, for all data sequences, is said to be a **universal model** (code) relative to model class $\mathcal{M}$:

$$\lim_{n\to\infty} \frac{1}{n} \max_{D\in\mathcal{D}} \left[\log_2 \frac{1}{q(D)} - \log_2 \frac{1}{p_{\hat{\theta}}(D)}\right] = 0 \ . \tag{1}$$

$$\log_2 \frac{1}{q(D)} - \log_2 \frac{1}{p_{\hat{\theta}}(D)} \geq \log_2 \frac{1}{q(D)} - \log_2 \frac{1}{p_{\theta}(D)}$$

Universal models

Universal model

A model (code) whose regret grows slower than n, for all data sequences, is said to be a **universal model** (code) relative to model class $\mathcal{M}$:

$$\lim_{n\to\infty} \frac{1}{n} \max_{D\in\mathcal{D}} \left[\log_2 \frac{1}{q(D)} - \log_2 \frac{1}{p_{\hat{\theta}}(D)}\right] = 0 \ . \qquad (1)$$

$$E_{D\sim p_\theta}\left[\log_2 \frac{1}{q(D)} - \log_2 \frac{1}{p_{\hat{\theta}}(D)}\right]$$
$$\geq E_{D\sim p_\theta}\left[\log_2 \frac{1}{q(D)} - \log_2 \frac{1}{p_\theta(D)}\right]$$

Universal models

Universal model

A model (code) whose regret grows slower than n, for all data sequences, is said to be a **universal model** (code) relative to model class $\mathcal{M}$:

$$\lim_{n\to\infty} \frac{1}{n} \max_{D\in\mathcal{D}} \left[\log_2 \frac{1}{q(D)} - \log_2 \frac{1}{p_{\hat{\theta}}(D)}\right] = 0 \ . \qquad (1)$$

$$E_{D\sim p_\theta}\left[\log_2 \frac{1}{q(D)} - \log_2 \frac{1}{p_{\hat{\theta}}(D)}\right]$$
$$\geq E_{D\sim p_\theta}\left[\log_2 \frac{1}{q(D)}\right] - E_{D\sim p_\theta}\left[\log_2 \frac{1}{p_\theta(D)}\right]$$

Universal models

Universal model

A model (code) whose regret grows slower than n, for all data sequences, is said to be a **universal model** (code) relative to model class $\mathcal{M}$:

$$\lim_{n\to\infty} \frac{1}{n} \max_{D\in\mathcal{D}} \left[\log_2 \frac{1}{q(D)} - \log_2 \frac{1}{p_{\hat{\theta}}(D)}\right] = 0 \ . \qquad (1)$$

$$E_{D\sim p_\theta}\left[\log_2 \frac{1}{q(D)} - \log_2 \frac{1}{p_{\hat{\theta}}(D)}\right] \geq E_{D\sim p_\theta}\left[\log_2 \frac{1}{q(D)}\right] - \sum_D p_\theta(D) \log_2 \frac{1}{p_\theta(D)}$$

Universal models

Universal model

A model (code) whose regret grows slower than n, for all data sequences, is said to be a **universal model** (code) relative to model class $\mathcal{M}$:

$$\lim_{n\to\infty} \frac{1}{n} \max_{D\in\mathcal{D}} \left[\log_2 \frac{1}{q(D)} - \log_2 \frac{1}{p_{\hat{\theta}}(D)} \right] = 0 \ . \qquad (1)$$

$$E_{D\sim p_\theta} \left[\log_2 \frac{1}{q(D)} - \log_2 \frac{1}{p_{\hat{\theta}}(D)} \right]$$

$$\geq E_{D\sim p_\theta} \left[\log_2 \frac{1}{q(D)} \right] - H(p_\theta)$$

Universal models

Universal model

A model (code) whose regret grows slower than n, for all data sequences, is said to be a **universal model** (code) relative to model class $\mathcal{M}$:

$$\lim_{n\to\infty} \frac{1}{n} \max_{D\in\mathcal{D}} \left[\log_2 \frac{1}{q(D)} - \log_2 \frac{1}{p_{\hat{\theta}}(D)}\right] = 0 \ . \qquad (1)$$

$$E_{D\sim p_\theta}\left[\log_2 \frac{1}{q(D)} - \log_2 \frac{1}{p_{\hat{\theta}}(D)}\right] \geq E_{D\sim p_\theta}\left[\log_2 \frac{1}{q(D)}\right] - nH(p_\theta^{(1)})$$

Universal models

Universal model

A model (code) whose regret grows slower than n, for all data sequences, is said to be a **universal model** (code) relative to model class $\mathcal{M}$:

$$\lim_{n\to\infty} \frac{1}{n} \max_{D\in\mathcal{D}} \left[\log_2 \frac{1}{q(D)} - \log_2 \frac{1}{p_{\hat{\theta}}(D)}\right] = 0 \ . \tag{1}$$

$$\frac{1}{n} E_{D\sim p_\theta} \left[\log_2 \frac{1}{q(D)} - \log_2 \frac{1}{p_{\hat{\theta}}(D)}\right] \geq \frac{1}{n} E_{D\sim p_\theta} \left[\log_2 \frac{1}{q(D)}\right] - H(p_\theta^{(1)})$$

Universal models

Universal model

A model (code) whose regret grows slower than n, for all data sequences, is said to be a **universal model** (code) relative to model class $\mathcal{M}$:

$$\lim_{n\to\infty} \frac{1}{n} \max_{D\in\mathcal{D}} \left[\log_2 \frac{1}{q(D)} - \log_2 \frac{1}{p_{\hat{\theta}}(D)}\right] = 0 \ . \qquad (1)$$

$$\lim_{n\to\infty} \frac{1}{n} E_{D\sim p_\theta} \left[\log_2 \frac{1}{q(D)} - \log_2 \frac{1}{p_{\hat{\theta}}(D)}\right] \geq \lim_{n\to\infty} \frac{1}{n} E_{D\sim p_\theta} \left[\log_2 \frac{1}{q(D)}\right] - H(p_\theta^{(1)})$$

Universal models

Universal model

A model (code) whose regret grows slower than n, for all data sequences, is said to be a **universal model** (code) relative to model class $\mathcal{M}$:

$$\lim_{n\to\infty} \frac{1}{n} \max_{D\in\mathcal{D}} \left[\log_2 \frac{1}{q(D)} - \log_2 \frac{1}{p_{\hat{\theta}}(D)}\right] = 0 \ . \tag{1}$$

$$0 \geq \lim_{n\to\infty} \frac{1}{n} E_{D\sim p_\theta} \left[\log_2 \frac{1}{q(D)}\right] - H(p_\theta^{(1)})$$

Universal models

Universal model

A model (code) whose regret grows slower than n, for all data sequences, is said to be a **universal model** (code) relative to model class $\mathcal{M}$:

$$\lim_{n\to\infty} \frac{1}{n} \max_{D\in\mathcal{D}} \left[\log_2 \frac{1}{q(D)} - \log_2 \frac{1}{p_{\hat{\theta}}(D)}\right] = 0 \ . \qquad (1)$$

$$\lim_{n\to\infty} \frac{1}{n} E_{D\sim p_\theta} \left[\log_2 \frac{1}{q(D)}\right] \quad \leq \quad H(p_\theta^{(1)})$$

Universal models

Universal model

A model (code) whose regret grows slower than n, for all data sequences, is said to be a **universal model** (code) relative to model class $\mathcal{M}$:

$$\lim_{n\to\infty} \frac{1}{n} \max_{D\in\mathcal{D}} \left[\log_2 \frac{1}{q(D)} - \log_2 \frac{1}{p_{\hat{\theta}}(D)}\right] = 0 \ . \qquad (1)$$

$$\lim_{n\to\infty} \frac{1}{n} E_{D\sim p_\theta} \left[\log_2 \frac{1}{q(D)}\right] \overset{(Gibbs)}{=} H(p_\theta^{(1)})$$

Universal models

Universal model

A model (code) whose regret grows slower than n, for all data sequences, is said to be a **universal model** (code) relative to model class $\mathcal{M}$:

$$\lim_{n\to\infty} \frac{1}{n} \max_{D\in\mathcal{D}} \left[\log_2 \frac{1}{q(D)} - \log_2 \frac{1}{p_{\hat{\theta}}(D)}\right] = 0 \ . \qquad (1)$$

$$\lim_{n\to\infty} \frac{1}{n} E_{D\sim p_\theta} \left[\log_2 \frac{1}{q(D)}\right] \stackrel{(Gibbs)}{=} H(p_\theta^{(1)})$$

This is equivalent to $\frac{1}{n} D(p_\theta \parallel q) \to 0$ for all $\theta \in \Theta$.

Universal models

The typical situation might be as follows:

1. We know (think) that the source symbols are generated by a Bernoulli model with parameter $\theta \in [0, 1]$.
2. We'd like to encode data at rate $H(\theta)$.
3. However, we do not know θ in advance.

Again, we don't need to believe that data are *really* generated by a Bernoulli model.

Among i.i.d. models, the rate $H(\theta)$ is the best achievable.

1. Universal Source Codes
   - Definitions
   - Universal Models
2. Two-Part Codes
   - Discrete Parameters
   - Continuous Parameters
   - Asymptotics: $\frac{k}{2}\log n$
3. Advanced Universal Codes
   - Mixture Codes
   - Normalized Maximum Likelihood
   - Universal Prediction

Two-Part Codes

Let $\mathcal{M} = \{p_\theta \;:\; \theta \in \Theta\}$ be a parametric probabilistic model class.

If the parameter space Θ is discrete, we can construct a (prefix) code $C_1 \;:\; \Theta \to \{0,1\}^*$ which maps each parameter value to a codeword of length $\ell_1(\theta)$.

For any distribution p_θ, the Shannon code-lengths satisfy

$$\ell_\theta(D) = \left\lceil \log_2 \frac{1}{p_\theta(D)} \right\rceil \approx \log_2 \frac{1}{p_\theta(D)} \quad .$$

Using parameter value θ, the total code-length becomes ($\approx$)

$$\ell_1(\theta) + \log_2 \frac{1}{p_\theta(D)} \quad .$$

Two-Part Codes

Using the maximum likelihood parameter, the total code-length becomes

$$\ell_{\text{two-part}}(D) = \ell_1(\hat{\theta}) + \log_2 \frac{1}{p_{\hat{\theta}}(D)} \ .$$

Hence, the *regret* of the two-part code is

$$\ell_{\text{two-part}}(D) - \log_2 \frac{1}{p_{\hat{\theta}}(D)} = \ell_1(\hat{\theta}) < cn \quad \text{for all } c > 0 \text{ and large } n.$$

For discrete parameter models **the two-part code is universal**.

Universality of Two-Part Codes

Since the two-part code is universal, its regret goes to zero, but there may be other codes for which regret goes to zero *faster*.

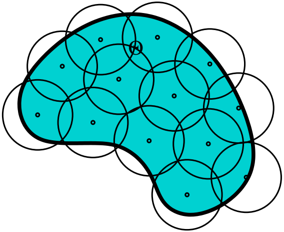
On the other hand, two-part codes have the advantage of being reasonably easy to understand.

Often they are also efficiently computable.

Continuous Parameters

What if the parameters are continuous (like polynomial coefficients)? We can't encode all continuous values with finite code-lengths!

Solution: Quantization. Choose a discrete subset of points, $\theta^{(1)}, \theta^{(2)}, \ldots$, and use only them.



Information Geometry!

If the points are sufficiently *dense* (in a code-length sense) then the code-length for data is still almost as short as $\min_{\theta \in \Theta} \ell_\theta(D)$.

About Quantization

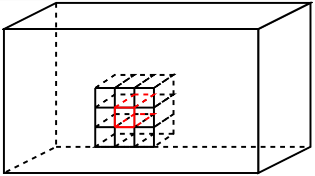
How many points should there be in the subset $\theta^{(1)}, \theta^{(2)}, \ldots$?

Intuition: Data does not allow us to tell apart θ_1 and θ_2 if $|\theta_1 - \theta_2| < c\frac{1}{\sqrt{n}}$. $\Rightarrow$ Don't care about higher precision.

Theorem (informally)

Optimal quantization accuracy is of order $\frac{1}{\sqrt{n}}$.

$\Rightarrow$ number of points $\approx \sqrt{n}^k = n^{k/2}$, where $k = \dim(\Theta)$.



Asymptotics: $\frac{k}{2} \log n$

With the precision $\frac{1}{\sqrt{n}}$ the code-length for data is almost optimal:

$$\min_{\theta^q \in \{\theta^{(1)}, \theta^{(2)}, \ldots\}} \ell_{\theta^q}(D) \approx \min_{\theta \in \Theta} \ell_\theta(D) = \log_2 \frac{1}{p_{\hat{\theta}}(D)} \quad (+O(1)) \ .$$

The total code-length becomes then ($\approx$)

$$\log_2 \frac{1}{p_{\hat{\theta}}(D)} + \frac{k}{2} \log_2 n \ ,$$

so that the regret is $\frac{k}{2} \log_2 n$.

Since $\log_2 n$ grows slower than n, the **two-part code is universal** also for continuous parameter models.

Mixture Universal Model

There are universal codes that are better than the two-part code.

For instance, given a uniquely decodable code for the parameters, let w be a p.m.f. over the parameter space Θ (quantized if continuous) defined as

$$w(\theta) = \frac{2^{-\ell(\theta)}}{c} \ , \quad \text{where } c = \sum_{\theta \in \Theta} 2^{-\ell(\theta)} \leq 1.$$

Let p^w be a **mixture distribution** over the data-sets $D \in \mathcal{D}$, defined as

$$p^w(D) = \sum_{\theta \in \Theta} p_\theta(D)\, w(\theta) \ ,$$

i.e., an "average" distribution, where each p_θ is weighted by $w(\theta)$.

Mixture Universal Model

The code-length of the **mixture model** p^w is given by

$$\log_2 \frac{1}{\sum_{\theta \in \Theta} p_\theta(D)\, w(\theta)} \leq \log_2 \frac{1}{p_{\hat{\theta}}(D)\, w(\hat{\theta})} = \log_2 \frac{1}{p_{\hat{\theta}}(D)} + \log_2 \frac{c}{2^{-\ell(\hat{\theta})}}$$

The right-hand side is equal to

$$\underbrace{\log_2 \frac{1}{p_{\hat{\theta}}(D)} + \ell(\hat{\theta})}_{\text{two-part code}} \underbrace{+ \log_2 c}_{\leq 0} \ ,$$

The mixture code is always at least as good as the two-part code.

Normalized Maximum Likelihood

Consider again the maximum likelihood model

$$p_{\hat{\theta}}(D) = \max_{\theta \in \Theta} p_\theta(D) \quad \Leftrightarrow \quad \ell_{\hat{\theta}}(D) = \log_2 \frac{1}{p_{\hat{\theta}}(D)} \ .$$

It is the best we can do under model $\mathcal{M}$.

Unfortunately, it is not possible to use the ML model for coding because is not a (fixed) probability distribution:

$$C = \sum_{D \in \mathcal{D}} p_{\hat{\theta}}(D) > 1 \quad \Leftrightarrow \quad \sum_{D \in \mathcal{D}} 2^{-\ell_{\hat{\theta}}(D)} > 1 \ ,$$

unless $\hat{\theta}$ is constant wrt. D. (Recall game-theoretic setting: Player chooses q before seeing data D.)

Normalized Maximum Likelihood

Normalized Maximum Likelihood

The **normalized maximum likelihood (NML) model** is obtained by normalizing the ML model:

$$p_{\mathrm{nml}}(D) = \frac{p_{\hat{\theta}}(D)}{C} \ , \quad \text{where } C = \sum_{D \in \mathcal{D}} p_{\hat{\theta}}(D) \ .$$

The regret of NML is given by

$$\log_2 \frac{1}{p_{\mathrm{nml}}(D)} - \log_2 \frac{1}{p_{\hat{\theta}}(D)} = \log_2 \frac{C}{p_{\hat{\theta}}(D)} - \log_2 \frac{1}{p_{\hat{\theta}}(D)} = \log_2 C \ ,$$

which is constant wrt. D.

Model Complexity

The quantity $\log_2 C$, which gives the (constant) regret of NML, is called the *parametric complexity* of model class $\mathcal{M}$.

Notice that if $\mathcal{D}$ and $\mathcal{M}$ are infinite, the sum defining C may diverge. In this case, we say that the parametric complexity of the model is infinite.

If the parametric complexity is infinite, then it's impossible to achieve constant regret. This is a real issue for some model classes used in practice.

Various work-arounds exist to extend NML to such model classes.

NML: Example

Consider the Bernoulli model: $p_\theta(D) = \theta^k(1-\theta)^{n-k}$, where k is the number of 1s.

It is easy to see that $\hat{\theta} = \frac{k}{n}$ and hence,

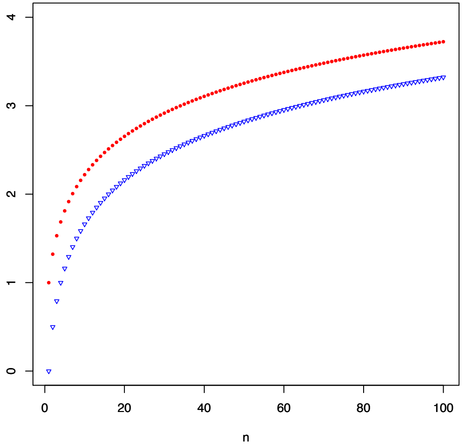
$$p_{\hat{\theta}}(D) = \left(\frac{k}{n}\right)^k \left(\frac{n-k}{n}\right)^{n-k} \quad .$$

We can compute C for fixed n as the sum

$$C = \sum_{k=0}^{n} \binom{n}{k} \left(\frac{k}{n}\right)^k \left(\frac{n-k}{n}\right)^{n-k} \quad .$$

For $n = 1, 2, \ldots, 100$: $C = 2, 2.5, 2.89, 3.22, 3.51, 3.78, \ldots, 13.21$.

NML: Example



●: $\log_2 C$ as a function of n

▽: $\frac{1}{2}\log_2 n$ (difference is const. $+\, o(1)$).

Normalized Maximum Likelihood

Let q be any distribution other than p_{nml}. Then

- there must a data-set $D' \in \mathcal{D}$ for which we have

$$q(D') < p_{\mathrm{nml}}(D')$$
$$\Leftrightarrow \underbrace{\log_2 \frac{1}{q(D')} - \log_2 \frac{1}{p_{\hat{\theta}}(D')}}_{\text{regret of } q} > \underbrace{\log_2 \frac{1}{p_{\mathrm{nml}}(D')} - \log_2 \frac{1}{p_{\hat{\theta}}(D')}}_{\text{regret of } p_{\mathrm{nml}} = \log_2 C} ,$$

For D', the regret of q is greater than $\log_2 C$.

Thus, the worst-case regret of q is greater than the (worst-case) regret of NML. $\Rightarrow$ NML has the least possible **worst-case regret**.

Universal Models

For 'smooth' parametric models, the regret of NML, $\log_2 C$, grows at rate $\frac{k}{2}\log_2 n$, so **NML is also a universal model**.

Since the regret of NML is the least possible, **NML is the optimal universal model**.

We have seen three kinds of universal codes:

1. two-part,
2. mixture,
3. NML.

There are also universal codes that are not based on any (explicit) model class: Lempel-Ziv (`gzip`)!

Uses of Universal Codes

So what do we do with them?

We can use universal codes for (at least) three purposes:

1. compression,
2. prediction,
3. model selection.

Universal Prediction

By the connection $p(D) = 2^{-\ell(D)}$, the following are equivalent:

- **good compression:** $\ell(D)$ is small,
- **good probability assignment:** $p(D) = \prod_{i=1}^{n} P(D_i \mid D_1, \ldots, D_{i-1})$ is high.
- **good predictions:** $p(D_i \mid D_1, \ldots, D_{i-1})$ is high for all $i \in \{1, \ldots, n\}$.

For instance, the mixture code gives a natural predictor which is equivalent to **Bayesian prediction**.

The NML model gives predictions that are good relative to the best model in the model class, **no matter what happens**.

Model (Class) Selection

Since a model class that enables good compression of the data must be based on exploiting the **regular features in the data**, the code-length can be used as a **yard-stick** for comparing model classes.

MDL Principle

MDL Principle

"Old-style":

- Choose the model $p_\theta \in \mathcal{M}$ that yields the shortest *two-part code-length*

$$\min_{\theta,\mathcal{M}} \ \ell_1(\theta) + \log_2 \frac{1}{p_\theta(D)}.$$

Modern:

- Choose the model class $\mathcal{M}$ that yields the shortest *universal code-length*

$$\min_{\mathcal{M}} \ell_{\mathcal{M}}(D).$$

Next Week

Next week: Minimum Description Length (MDL) principle