Information-Theoretic Modeling

Lecture 8: Universal Source Coding

Teemu Roos

Department of Computer Science, University of Helsinki

Fall 2009

UNIVERSITY OF HELSINKI

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 — ooh-la-la
   - Asymptotics: $\frac{k}{2} \log n$

1. Universal Source Codes
   - Definitions
   - Universal Models

2. Two-Part Codes
   - Discrete Parameters
   - Continuous Parameters — ooh-la-la
   - Asymptotics: $\frac{k}{2} \log n$

3. Advanced Universal Codes
   - Mixture Codes
   - Normalized Maximum Likelihood
   - Universal Prediction

Definitions

We call a probability distribution $p : \mathcal{D} \rightarrow [0, 1]$ a **model**.

A **model class** $\mathcal{M} = \{p_\theta : \theta \in \Theta\}$ is a set of probability distributions (models).

Definitions

We call a probability distribution $p : \mathcal{D} \to [0,1]$ a **model**.

A **model class** $\mathcal{M} = \{p_\theta : \theta \in \Theta\}$ is a set of probability distributions (models).

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

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

Definitions

We call a probability distribution $p : \mathcal{D} \to [0,1]$ a **model**.

A **model class** $\mathcal{M} = \{p_\theta : \theta \in \Theta\}$ is a set of probability distributions (models).

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

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

Depends on D!

Definitions

We call a probability distribution $p : \mathcal{D} \rightarrow [0,1]$ a **model**.

A **model class** $\mathcal{M} = \{p_\theta : \theta \in \Theta\}$ is a set of probability distributions (models).

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

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

Depends on D!

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

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

Universal models

Universal model

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

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

Universal models

Universal model

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

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

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

Universal models

Universal model

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

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

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

Universal models

Universal model

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

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

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

Universal models

Universal model

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

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

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

Universal models

Universal model

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

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

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

Universal models

Universal model

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

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

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

Universal models

Universal model

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

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

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

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

Universal models

Universal model

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

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

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

Universal models

Universal model

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

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

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

Universal models

Universal model

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

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

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

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

Universal models

Universal model

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

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

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

Universal models

Universal model

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

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

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

Universal models

Universal model

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

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

$$\lim_{n\to\infty} \frac{1}{n} E_{D\sim p_\theta}\left[\log_2 \frac{1}{q(D)}\right] = H(p_\theta^{(1)}) \tag{2}$$

Universal models

Universal model

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

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

$$\lim_{n\to\infty} \frac{1}{n} E_{D\sim p_\theta} \left[\log_2 \frac{1}{q(D)}\right] = H(p_\theta^{(1)}) \qquad (2)$$

This is another (stochastic) definition of universality, equivalent to $\frac{1}{n} D(p_\theta \,\|\, q) \to 0$ for all $\theta \in \Theta$. It is weaker since (1) $\Rightarrow$ (2).

Universal models

The typical situation might be as follows:

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 $p \in [0, 1]$.

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 $p \in [0, 1]$.
2. However, we do not know p in advance.

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 $p \in [0, 1]$.
2. However, we do not know p in advance.
3. We'd like to encode data at rate $H(p)$.

Two-Part Codes

Let $\mathcal{M} = \{p_\theta \,:\, \theta \in \Theta\}$ be a parametric probabilistic model class, i.e., a set of distributions p_θ indexed by parameter θ.

Two-Part Codes

Let $\mathcal{M} = \{p_\theta \ : \ \theta \in \Theta\}$ be a parametric probabilistic model class, i.e., a set of distributions p_θ indexed by parameter θ.

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 $\ell_1(\theta)$.

Two-Part Codes

Let $\mathcal{M} = \{p_\theta \ : \ \theta \in \Theta\}$ be a parametric probabilistic model class, i.e., a set of distributions p_θ indexed by parameter θ.

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

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

Two-Part Codes

Let $\mathcal{M} = \{p_\theta : \theta \in \Theta\}$ be a parametric probabilistic model class, i.e., a set of distributions p_θ indexed by parameter θ.

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

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

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

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

Two-Part Codes

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

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

Two-Part Codes

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

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

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

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

Two-Part Codes

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

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

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

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

Two-Part Codes

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

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

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**.

Continuous Parameters

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

Continuous Parameters

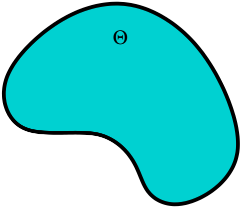
What if the parameters are continuous (like polynomial coefficients)? We can't encode all continuous values with finite code-lengths!

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

Continuous Parameters

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

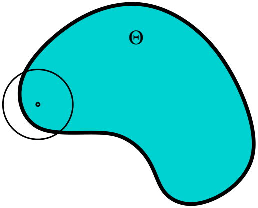
Solution: Quantization. Choose a discrete subset of points, $\theta^{(1)}, \theta^{(2)}, \ldots$, and use only them.



Continuous Parameters

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

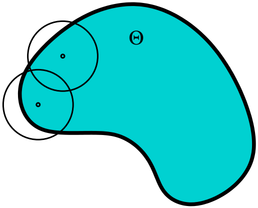
Solution: Quantization. Choose a discrete subset of points, $\theta^{(1)}, \theta^{(2)}, \ldots$, and use only them.



Continuous Parameters

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

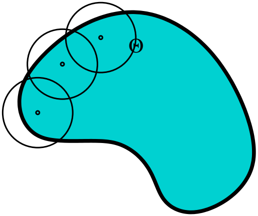
Solution: Quantization. Choose a discrete subset of points, $\theta^{(1)}, \theta^{(2)}, \ldots$, and use only them.



Continuous Parameters

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

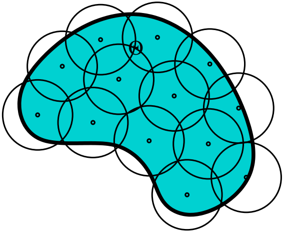
Solution: **Quantization.** Choose a discrete subset of points, $\theta^{(1)}, \theta^{(2)}, \ldots$, and use only them.



Continuous Parameters

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

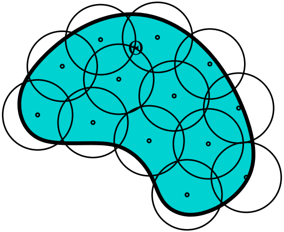
Solution: Quantization. Choose a discrete subset of points, $\theta^{(1)}, \theta^{(2)}, \ldots$, and use only them.



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.

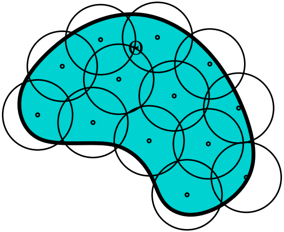


If the points are sufficiently *dense* (in a code-length sense) then the code-length for data is still almost as short as $\min_{\theta \in \Theta} \ell_\theta(D)$.

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 $\theta^{(1)}, \theta^{(2)}, \ldots$?

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.

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

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

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

About Quantization

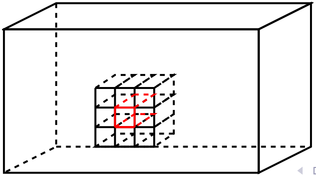
How many points should there be in the subset $\theta^{(1)}, \theta^{(2)}, \ldots$?

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

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

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



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

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

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

The code-length for the quantized parameters becomes

$$\ell(\theta^q) \approx \log_2 n^{k/2} = \frac{k}{2} \log_2 n \ .$$

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

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

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

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

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

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

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 $\frac{k}{2}\log_2 n$.

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

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

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

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.

1. Universal Source Codes
   - Definitions
   - Universal Models

2. Two-Part Codes
   - Discrete Parameters
   - Continuous Parameters — ooh-la-la
   - Asymptotics: $\frac{k}{2} \log n$

3. Advanced Universal Codes
   - Mixture Codes
   - Normalized Maximum Likelihood
   - Universal Prediction

Mixture Universal Model

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

Mixture Universal Model

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

For instance, given a code for the parameters, let w be a distribution over the parameter space Θ (quantized if necessary) defined as

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

Mixture Universal Model

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

For instance, given a code for the parameters, let w be a distribution over the parameter space Θ (quantized if necessary) defined as

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

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

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

[corrected on Oct 5, 2009]

Mixture Universal Model

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

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

[corrected on Oct 5, 2009]

The right-hand side is equal to

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

Mixture Universal Model

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

$$\log_2 \frac{1}{\sum_{\theta \in \Theta} p_\theta(D)\, w(\theta)} \leq \log_2 \frac{1}{p_{\hat{\theta}}(D)\, w(\hat{\theta})} \quad \text{[corrected on Oct 5, 2009]}$$

$$= \log_2 \frac{1}{p_{\hat{\theta}}(D)} + \log_2 \frac{c}{2^{-\ell(\hat{\theta})}} \quad .$$

The right-hand side is equal to

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

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

Normalized Maximum Likelihood

Consider again the maximum likelihood model

$$p_{\hat{\theta}}(D) = \max_{\theta \in \Theta} p_{\theta}(D) \quad .$$

It is the best probability assignment achievable under model $\mathcal{M}$.

Normalized Maximum Likelihood

Consider again the maximum likelihood model

$$p_{\hat{\theta}}(D) = \max_{\theta \in \Theta} p_{\theta}(D) \quad .$$

It is the best probability assignment achievable under model $\mathcal{M}$.

Unfortunately, it is not possible to use the ML model for coding because is not a probability distribution, i.e.,

$$C = \sum_{D \in \mathcal{D}} p_{\hat{\theta}}(D) > 1 \quad ,$$

unless $\hat{\theta}$ is constant wrt. D.

Normalized Maximum Likelihood

Normalized Maximum Likelihood

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

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

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.

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

$$q(D') < p_{\mathrm{nml}}(D')$$

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

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

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

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

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

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

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

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

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 slower than n, so **NML is also a universal model**.

Universal Models

For 'smooth' parametric models, the regret of NML, $\log_2 C$, grows slower than n, so **NML is also a universal model**.

We have seen three kinds of universal codes:

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

Universal Models

For ‘smooth’ parametric models, the regret of NML, $\log_2 C$, grows slower than n, so **NML is also a 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?

Uses of Universal Codes

So what do we do with them?

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

Uses of Universal Codes

So what do we do with them?

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

1. compression,

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,

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,

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.

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 most $i \in \{1, \ldots, n\}$.

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 most $i \in \{1, \ldots, n\}$.

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

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 most $i \in \{1, \ldots, n\}$.

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*

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

Modern:

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

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

Next Week

Next week:

Next Week

Next week:

- more about MDL principle,

Next Week

Next week:

- more about MDL principle,
- even more about MDL principle.