Adaptive Information Processing

Model complexity and the MDL principle

Tjalling Tjalkens and Bert de Vries

February 19, 2009

Signal Processing Group

Global overview

Part A: The Bayesian Information Criterion

Part B: Bayesian model estimation and the Context-tree model selection

Part C: Descriptive complexity

Prerequisites

Additional reading

Introduction

Bishop §1.2: Probability Theory
Bishop §1.3: Model Selection
Bishop §1.4: The Curse of Dimensionality

Probabilities

Bishop §2.1: Binary Variables
Bishop §2.2: Multinomial Variables

Part A

The Bayesian Information Criterion

Parameter and model estimation

Additional reading

Introduction

Bishop §3.3: Bayesian Linear Regression
Bishop §3.4: Bayesian Model Comparison

Parameter estimation

Define our variables!

Model	$\boldsymbol{\mathcal{M}_i}$	model prior	$\boldsymbol{p(\mathcal{M}_i)}$
Parameters	$\boldsymbol{\theta_i}$	parameter prior	$\boldsymbol{p(\theta_i\|\mathcal{M}_i)}$
Data	$\boldsymbol{x^N}$		

A-posteriori parameter distribution

$$\boldsymbol{p(\theta_i|\mathcal{M}_i, x^N) = \frac{p(\theta_i|\mathcal{M}_i)p(x^N|\mathcal{M}_i, \theta_i)}{p(x^N|\mathcal{M}_i)}}$$

$$\boldsymbol{p(x^N|\mathcal{M}_i) = \int_{\Theta_i} p(\theta_i|\mathcal{M}_i)p(x^N|\mathcal{M}_i, \theta_i)\, d\theta_i}$$

Parameter estimation

Maximum Likelihood

We want a point estimate for $\boldsymbol{\theta}_i$ (given $\boldsymbol{\mathcal{M}}_i$).

$$\hat{\boldsymbol{\theta}}_i = \arg\max_{\boldsymbol{\theta}_i} p(\boldsymbol{\theta}_i | \boldsymbol{\mathcal{M}}_i, \boldsymbol{x}^N) = \arg\max_{\boldsymbol{\theta}_i} p(\boldsymbol{x}^N | \boldsymbol{\mathcal{M}}_i, \boldsymbol{\theta}_i)$$

Where we assume a uniform prior or want to work without priors.

Model estimation

A-posteriori model distribution

$$\boldsymbol{p(\mathcal{M}_i|x^N) = \frac{p(\mathcal{M}_i)p(x^N|\mathcal{M}_i)}{p(x^N)}}$$

$$p(x^N) = \int_{\mathcal{M}_i} p(\mathcal{M}_i)p(x^N|\mathcal{M}_i)\, d\mathcal{M}_i$$

Model estimation

Maximum Likelihood

We want a point estimate for $\mathcal{M}$.

$$\hat{\mathcal{M}} = \arg\max_{\mathcal{M}_i} p(\mathcal{M}_i | x^N) = \arg\max_{\mathcal{M}_i} p(x^N | \mathcal{M}_i)$$

Where we assume a uniform prior or want to work without priors.

Model estimation

We need to compute

$$p(x^N|\mathcal{M}_i) = \int_{\Theta_i} p(\theta_i|\mathcal{M}_i)p(x^N|\mathcal{M}_i,\theta_i)\,d\theta_i$$

Often $p(\theta_i|\mathcal{M}_i,x^N)$ is sharply peaked and because

$$p(\theta_i|\mathcal{M}_i,x^N) \propto p(\theta_i|\mathcal{M}_i)p(x^N|\mathcal{M}_i,\theta_i),$$

we might be able to approximate the integrand given above.

Maximum Likelihood and Overfitting

Additional reading

Overfitting

Bishop §1.1: Example: Polynomial Curve Fitting

Attempt 1 (Maximum Likelihood)

We approximate the integrand by its peak ($\boldsymbol{\theta}_i^{\text{MAP}}$ or $\boldsymbol{\theta}_i^{\text{ML}}$)

$$\boldsymbol{p(\theta_i|\mathcal{M}_i)p(x^N|\mathcal{M}_i,\theta_i) \approx}$$
$$\boldsymbol{\delta(\theta_i - \theta_i^{\text{ML}})p(\theta_i|\mathcal{M}_i)p(x^N|\mathcal{M}_i,\theta_i)}$$

and find

$$\boldsymbol{p(x^N|\mathcal{M}_i) \propto p(\theta_i^{\text{ML}}|\mathcal{M}_i)p(x^N|\mathcal{M}_i,\theta_i^{\text{ML}})}$$

So we end up with

$$\boldsymbol{\mathcal{M}^{\text{MAP}} = \arg\max_{\mathcal{M}_i} p(\theta_i^{\text{ML}}|\mathcal{M}_i)p(x^N|\mathcal{M}_i,\theta_i^{\text{ML}})}$$
$$\boldsymbol{\mathcal{M}^{\text{ML}} = \arg\max_{\mathcal{M}_i} p(x^N|\mathcal{M}_i,\theta_i^{\text{ML}})}$$

Attempt 1: an example

Consider a linear regression model.

$$y_n = \theta^T \underline{x}_n + n_n;$$

$$y_n \in \mathbb{R}; \qquad \theta \in \mathbb{R}^k; \qquad \underline{x}_n \in \mathbb{R}^k; \qquad n_n \sim \mathcal{N}(0, \sigma^2)$$

Observe: $(y_1, \underline{x}_1), (y_2, \underline{x}_2), \ldots, (y_N, \underline{x}_N)$.

ML estimate: $\hat{\theta} = (X^T X)^{-1} X^T \underline{y}$.

Matrix: $X = [\underline{x}_1, \underline{x}_2, \ldots \underline{x}_N]^T$.

Models: $\mathcal{M} \subset \{1, 2, \ldots, k\}$. e.g.

$$\mathcal{M} = \{1, 3\}; \qquad y_n = \theta_1 x_{n1} + \theta_3 x_{n3} + n_n$$

Attempt 1: an example (continued)

$$N = 50; \quad \underline{x} \in [0,1]^3; \quad \theta = (0, 0.6, 0);$$

$\sigma^2 = 1$ actual $\sigma^2 = 0.799$

$\mathcal{M}$	$\hat{\theta}_1$	$\hat{\theta}_2$	$\hat{\theta}_3$	$\hat{\sigma}^2$	$\ln P_{ML}$	$\ln P_{ML}$
					$\sigma^2 = 1$	$\sigma^2 = \hat{\sigma}^2$
$\{\}$	0	0	0	0.949	−69.675	−69.642
$\{1\}$	0.690	0	0	0.804	−66.040	−65.485
$\{2\}$	0	0.604	0	0.799	−65.934	−65.352
$\{3\}$	0	0	0.307	0.912	−68.738	−68.635
$\{12\}$	0.379	0.361	0	0.780	−65.441	−64.728
$\{13\}$	1.171	0	−0.522	0.766	−65.099	−64.286
$\{23\}$	0	0.970	−0.472	0.766	−65.101	−64.287
$\{123\}$	0.908	0.752	−0.940	0.686	−63.097	−61.525

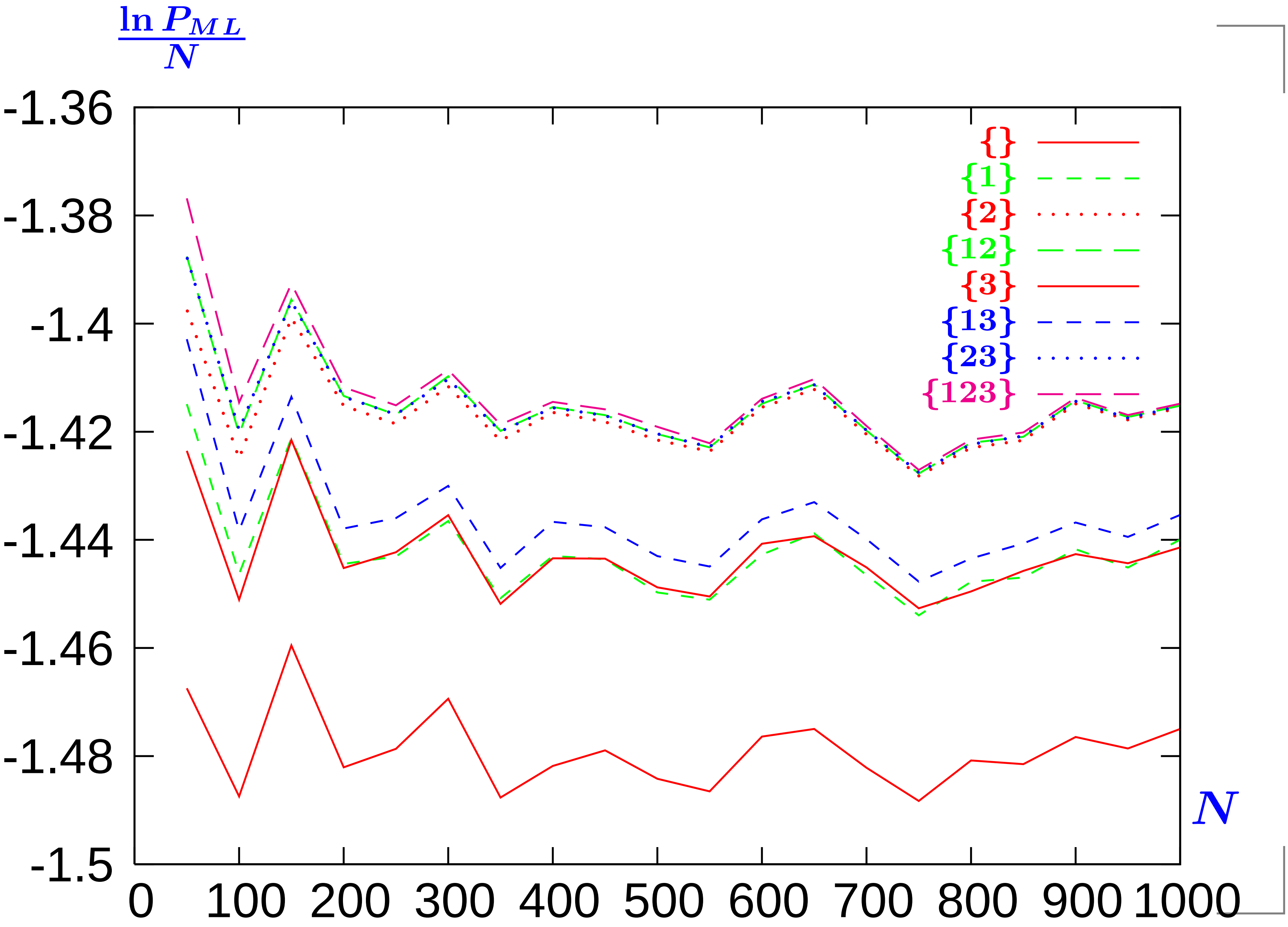
Attempt 1: an example (continued)

$$N = 1000; \quad \underline{x} \in [0,1]^3; \quad \theta = (0, 0.6, 0);$$

$\sigma^2 = 1$ actual $\sigma^2 = 1.015$

$\mathcal{M}$	$\hat{\theta}_1$	$\hat{\theta}_2$	$\hat{\theta}_3$	$\hat{\sigma}^2$	$\ln P_{ML}$	$\ln P_{ML}$
					$\sigma^2 = 1$	$\sigma^2 = \hat{\sigma}^2$
$\{\}$	0	0	0	1.144	-1491	-1486
$\{1\}$	0.435	0	0	1.083	-1460	-1459
$\{2\}$	0	0.619	0	1.015	-1426	-1426
$\{3\}$	0	0	0.507	1.058	-1448	-1447
$\{12\}$	-0.099	0.693	0	1.013	-1425	-1425
$\{13\}$	0.105	0	0.430	1.056	-1447	-1446
$\{23\}$	0	0.549	0.095	1.013	-1426	-1426
$\{123\}$	-0.173	0.622	0.167	1.010	-1424	-1424

Attempt 1: an example (continued)



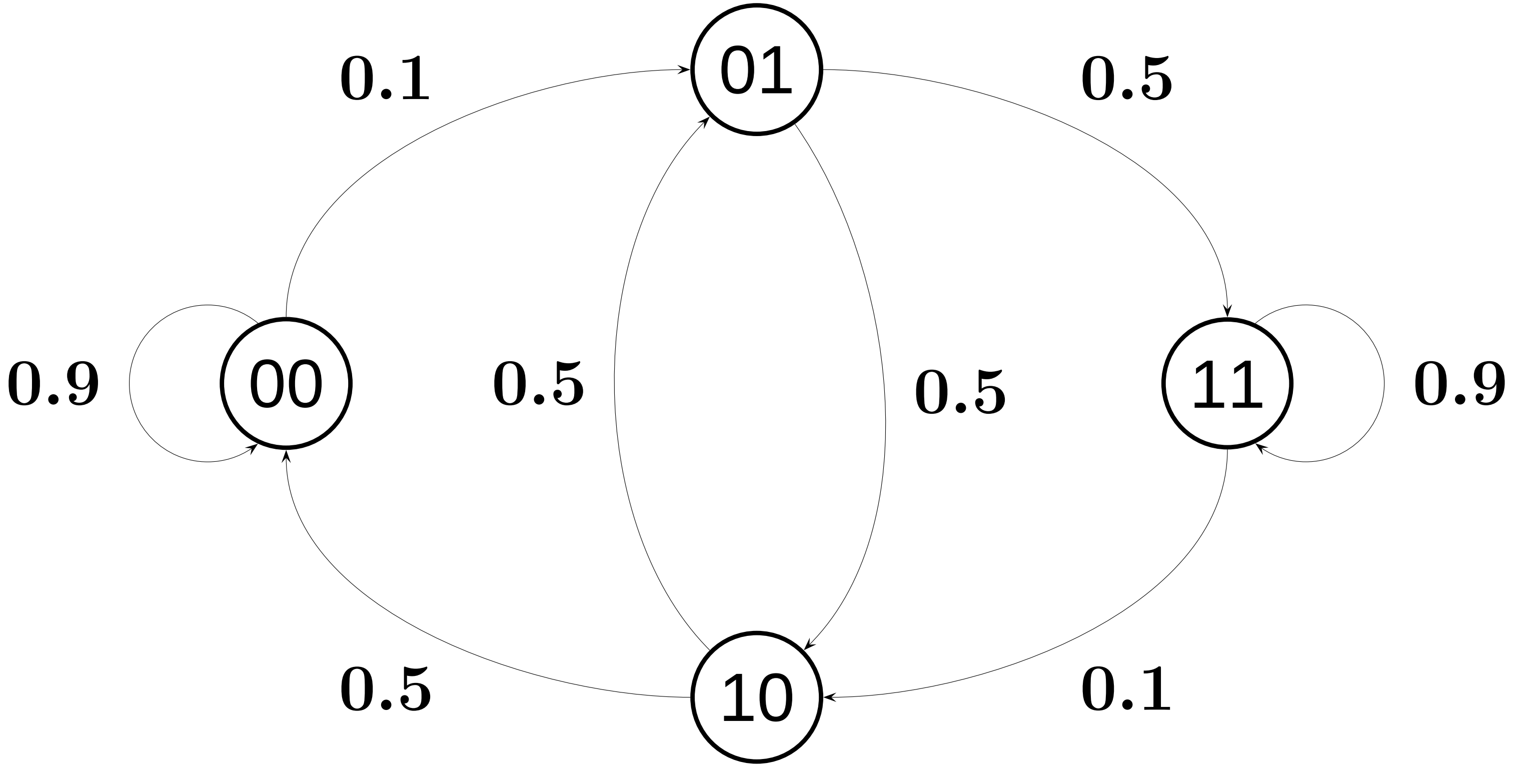
Attempt 1: another example

A discrete data example.

Consider a binary second order Markov process:
$\Pr\{X_i = 1|x^{i-1}\} = \Pr\{X_i = 1|x_{i-2}x_{i-1}\}$.
So, it is actually a set of four i.i.d. sub-sources.
ML estimate of an i.i.d. binary source:

$$
\begin{aligned}
n(s|x) &= \text{the number of times } s \in \mathcal{X}^* \text{ occurs in } x \\
p(x^N|\theta) &= (1-\theta)^{n(0|x^N)}\theta^{n(1|x^N)} \\
\frac{\partial}{\partial\theta}\ln p(x^N|\theta) &= \frac{n(1|x^N) - N\theta}{\theta(1-\theta)} = 0 \\
\hat{\theta} &= \frac{n(1|x^N)}{N}
\end{aligned}
$$

Attempt 1: another example (ctd.)



Attempt 1: another example (ctd.)

Let S be the state variable of an m-th order Markov source, so $S_i = X_{i-m} \dots X_{i-1}$ and $l(S_i) = m$ bits, then

$$\theta_s = \Pr\{X_i = 1 | S_i = s\}$$

The Maximum Likelihood estimator is

$$\hat{\theta}_s = \frac{n(s1|x^N)}{n(s0|x^N) + n(s1|x^N)}$$

With this we find the ML probability for x^N

$$p(x^N|m, \underline{\hat{\theta}}) = p(x_1, \dots x^m) \prod_{s \in \{0,1\}^m} \left\{ \hat{\theta}_s^{n(s1|x^N)} (1 - \hat{\theta}_s)^{n(s0|x^N)} \right\}$$

Attempt 1: another example (ctd.)

```
octave:1> mytest(50,[0.1,0.5,0.5,0.9],4)
ML models sequence logprobs:
Order 0:  logpr = -34.657359
Order 1:  logpr = -14.546445
Order 2:  logpr = -14.883390
Order 3:  logpr = -15.185437
Order 4:  logpr = -15.444986

octave:2> mytest(200,[0.1,0.5,0.5,0.9],4)
ML models sequence logprobs:
Order 0:  logpr = -137.416984
Order 1:  logpr = -102.949521
Order 2:  logpr = -87.992931
Order 3:  logpr = -84.732718
Order 4:  logpr = -80.546002
```

Attempt 1: conclusion

Obviously, this method does not work.

- Any model that includes the actual model assigns essentially the same probability to the data.

Attempt 1: conclusion

Obviously, this method does not work.

- Any model that includes the actual model assigns essentially the same probability to the data.
- We observe that (usually) the higher order models give higher probabilities to the sequence.

Attempt 1: conclusion

Obviously, this method does not work.

- Any model that includes the actual model assigns essentially the same probability to the data.
- We observe that (usually) the higher order models give higher probabilities to the sequence.
- But high order models cannot predict well (too restricted).

Attempt 1: conclusion

Obviously, this method does not work.

- Any model that includes the actual model assigns essentially the same probability to the data.
- We observe that (usually) the higher order models give higher probabilities to the sequence.
- But high order models cannot predict well (too restricted).
- The higher order models are too well tuned.

Attempt 1: conclusion

Obviously, this method does not work.

- Any model that includes the actual model assigns essentially the same probability to the data.
- We observe that (usually) the higher order models give higher probabilities to the sequence.
- But high order models cannot predict well (too restricted).
- The higher order models are too well tuned.

This is undesirable, the estimated model adapts itself to the noise and the resulting model is an over estimation of the actual model.

Preventing Overfitting

Additional reading

Laplace Approximation

Bishop §4.4: The Laplace Approximation

Attempt 2 (Laplace approximation)

We approximate the integrand by a Gaussian around the peak. The mean and variance of the Gaussian are determined by the integrand.

This approximation turns out to give more interesting results.

Laplace approximation

Suppose we have an arbitrary non-negative real function $f(z)$, where z is a k-dimensional vector. We need an estimate of the normalizing constant Z_f.

$$Z_f = \int f(z)\, dz$$

Let z_0 be a maximum of $f(z)$. Use the Taylor expansion.

$$\ln f(z) \approx \ln f(z_0) - \frac{1}{2}(z - z_0)A(z - z_0)$$

$$A_{ij} = -\frac{\partial^2}{\partial z_i \partial z_j} \ln f(z)\bigg|_{z=z_0}$$

Laplace approximation

Approximate $f(z)$ by the unnormalized Gaussian

$$g(z) = f(z_0) \exp\left(-\frac{1}{2}(z - z_0)A(z - z_0)\right)$$

A, not necessarily good, approximation of Z_f is

$$Z_f \approx Z_g = \int g(z)\, dz = f(z_0)\sqrt{\frac{(2\pi)^k}{\det A}}$$

Laplace approximation

Example 1:

$$f(z) = \frac{1}{z^2+1} \quad \text{Has maximum at } z_0 = 0.$$

$$Z_f = \pi$$

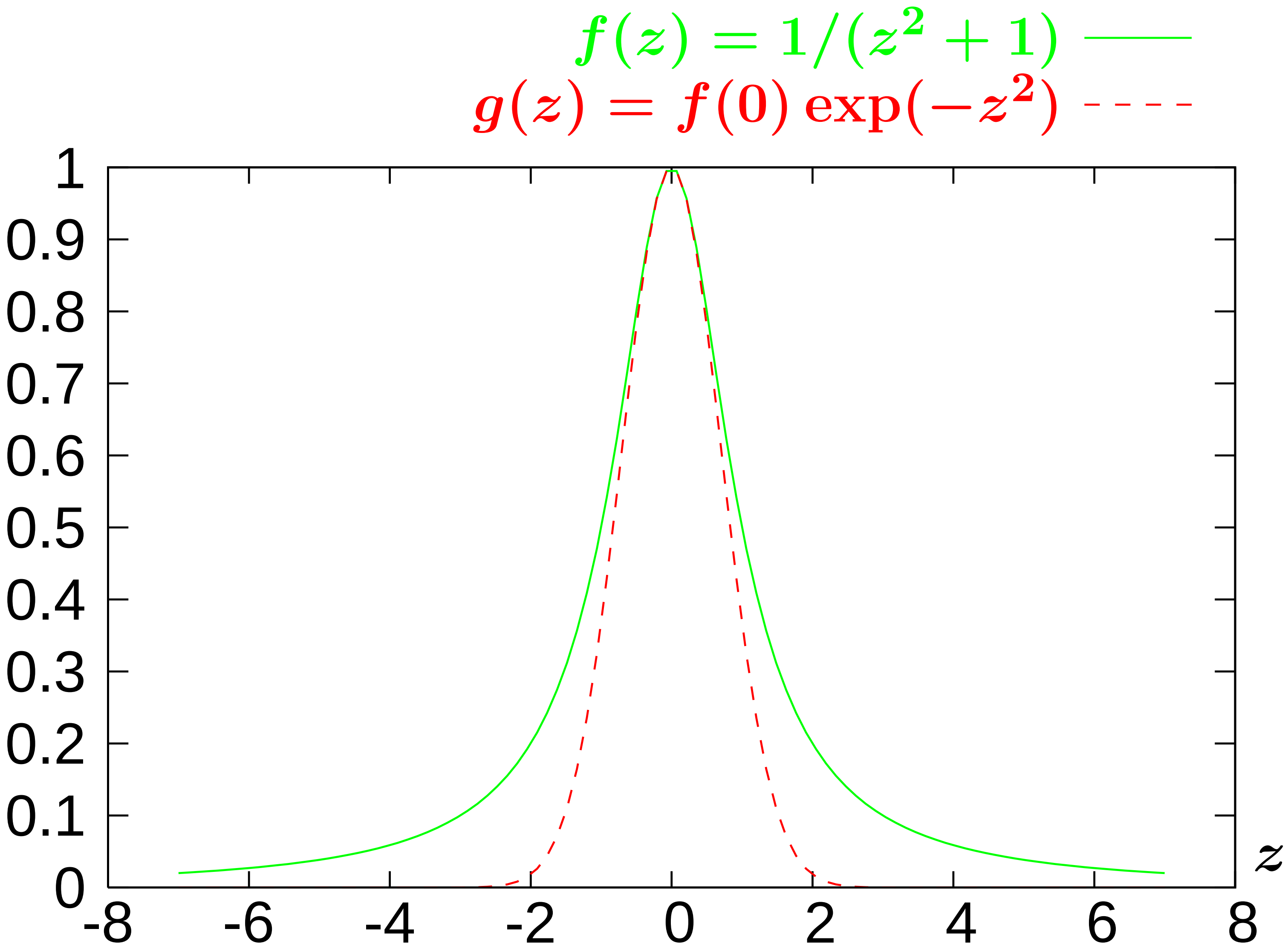
$$A = -\frac{\partial^2}{\partial z^2} \ln f = -\frac{f'' f - f'^2}{f^2}$$

$$f(0) = 1; \quad f'(0) = 0; \quad f''(0) = -2; \text{ so } A = 2$$

$$g(z) = f(0) \exp\left(-\frac{1}{2} z A z\right) = e^{-z^2}$$

$$Z_g = \sqrt{\pi}$$

Laplace approximation



Attempt 2 (Laplace approximation)

Consider again $p(x^N|\mathcal{M}_i)$.

$$p(x^N|\mathcal{M}_i) = \int_{\Theta} p(\theta_i|\mathcal{M}_i)p(x^N|\mathcal{M}_i,\theta_i)\,d\theta_i$$

We again use the fact that

$$p(\theta_i|\mathcal{M}_i,x^N) \propto p(\theta_i|\mathcal{M}_i)p(x^N|\mathcal{M}_i,\theta_i)$$

is often sharply peaked, say at $\hat{\theta}_i$. Using the Laplace approximation we may write

$$p(x^N|\mathcal{M}_i) \approx \sqrt{\frac{(2\pi)^k}{\det A}}p(\hat{\theta}_i|\mathcal{M}_i)p(x^N|\mathcal{M}_i,\hat{\theta}_i)$$

Attempt 2 (Laplace approximation)

Comparing two models give

$$\frac{p(\mathcal{M}_i|x^N)}{p(\mathcal{M}_j|x^N)} \approx \frac{p(\mathcal{M}_i)}{p(\mathcal{M}_j)} \frac{\sqrt{\frac{(2\pi)^{k_i}}{\det A_i}} p(\hat{\theta}_i|\mathcal{M}_i)}{\sqrt{\frac{(2\pi)^{k_j}}{\det A_j}} p(\hat{\theta}_j|\mathcal{M}_j)} \frac{p(x^N|\mathcal{M}_i, \hat{\theta}_i)}{p(x^N|\mathcal{M}_j, \hat{\theta}_j)}$$

Attempt 2 (Laplace approximation)

Comparing two models give

$$\frac{p(\mathcal{M}_i|x^N)}{p(\mathcal{M}_j|x^N)} \approx \frac{p(\mathcal{M}_i)}{p(\mathcal{M}_j)} \frac{\sqrt{\frac{(2\pi)^{k_i}}{\det A_i}}\, p(\hat{\theta}_i|\mathcal{M}_i)}{\sqrt{\frac{(2\pi)^{k_j}}{\det A_j}}\, p(\hat{\theta}_j|\mathcal{M}_j)} \frac{p(x^n|\mathcal{M}_i,\hat{\theta}_i)}{p(x^N|\mathcal{M}_j,\hat{\theta}_j)}$$

initial model preference

Attempt 2 (Laplace approximation)

Comparing two models give

$$\frac{p(\mathcal{M}_i|x^N)}{p(\mathcal{M}_j|x^N)} \approx \frac{p(\mathcal{M}_i)}{p(\mathcal{M}_j)} \frac{\sqrt{\frac{(2\pi)^{k_i}}{\det A_i}}\,p(\hat{\theta}_i|\mathcal{M}_i)}{\sqrt{\frac{(2\pi)^{k_j}}{\det A_j}}\,p(\hat{\theta}_j|\mathcal{M}_j)} \frac{p(x^N|\mathcal{M}_i, \hat{\theta}_i)}{p(x^N|\mathcal{M}_j, \hat{\theta}_j)}$$

cost of (number of) parameters

Attempt 2 (Laplace approximation)

Comparing two models give

$$\frac{p(\mathcal{M}_i|x^N)}{p(\mathcal{M}_j|x^N)} \approx \frac{p(\mathcal{M}_i)}{p(\mathcal{M}_j)} \frac{\sqrt{\frac{(2\pi)^{k_i}}{\det A_i}} p(\hat{\theta}_i|\mathcal{M}_i)}{\sqrt{\frac{(2\pi)^{k_j}}{\det A_j}} p(\hat{\theta}_j|\mathcal{M}_j)} \frac{p(x^N|\mathcal{M}_i, \hat{\theta}_i)}{p(x^N|\mathcal{M}_j, \hat{\theta}_j)}$$

likelihood ratio

Attempt 2 (Laplace approximation)

Comparing two models give

$$\frac{p(\mathcal{M}_i|x^N)}{p(\mathcal{M}_j|x^N)} \approx \frac{p(\mathcal{M}_i)}{p(\mathcal{M}_j)} \frac{\sqrt{\frac{(2\pi)^{k_i}}{\det A_i}}p(\hat{\theta}_i|\mathcal{M}_i)}{\sqrt{\frac{(2\pi)^{k_j}}{\det A_j}}p(\hat{\theta}_j|\mathcal{M}_j)} \frac{p(x^N|\mathcal{M}_i,\hat{\theta}_i)}{p(x^N|\mathcal{M}_j,\hat{\theta}_j)}$$

Cost factors are: initial model preference, cost of (number of) parameters, likelihood ratio.

This is ML model estimation, it works because we consider the model complexity also!

BIC: Bayesian Information Criterion

A more refined approximation (Schwartz criterion or Bayesian Information Criterion) gives

$$\log \frac{p(\mathcal{M}_i|x^N)}{p(\mathcal{M}_j|x^N)} \approx \log \frac{p(\mathcal{M}_i)}{p(\mathcal{M}_j)} + \log \frac{p(x^N|\mathcal{M}_i,\hat{\theta}_i)}{p(x^N|\mathcal{M}_j,\hat{\theta}_j)} + \frac{1}{2}(k_i - k_j)\log N,$$

where k_i resp. k_j gives the number of free parameters in model $\mathcal{M}_i$ or $\mathcal{M}_j$ respectively.

This BIC is widely applied and turned out to be very usefull.

What happens when we apply the correction term $\frac{k}{2}\log N$? We shall revisit the two examples.

Example 1 with BIC correction

$$N = 50; \quad \underline{x} \in [0,1]^3; \quad \theta = (0, 0.6, 0);$$

$\sigma^2 = 1$ actual $\sigma^2 = 0.852$

$\mathcal{M}$	$\hat\theta_1$	$\hat\theta_2$	$\hat\theta_3$	$\hat\sigma^2$	$\ln P_{BIC}$
$\{\}$	0	0	0	1.068	−72.653
$\{1\}$	0.699	0	0	0.909	−70.632
$\{2\}$	0	0.773	0	0.841	−68.923
$\{3\}$	0	0	0.572	0.944	−71.491
$\{12\}$	0.159	0.662	0	0.837	−70.790
$\{13\}$	0.553	0	0.172	0.905	−72.478
$\{23\}$	0	0.811	−0.050	0.840	−70.869
$\{123\}$	0.240	0.728	−0.159	0.834	−72.670

Example 1 with BIC correction

$$N = 1000; \quad \underline{x} \in [0,1]^3; \quad \theta = (0, 0.6, 0);$$

$$\sigma^2 = 1 \quad \text{actual } \sigma^2 = 0.977$$

$\mathcal{M}$	$\hat{\theta}_1$	$\hat{\theta}_2$	$\hat{\theta}_3$	$\hat{\sigma}^2$	$\ln P_{BIC}$
$\{\}$	0	0	0	1.077	−1457.2
$\{1\}$	0.411	0	0	1.022	−1433.4
$\{2\}$	0	0.551	0	0.976	−1410.3
$\{3\}$	0	0	0.362	1.034	−1439.3
$\{12\}$	−0.017	0.564	0	0.976	−1413.7
$\{13\}$	0.315	0	0.128	1.020	−1435.6
$\{23\}$	0	0.637	−0.117	0.974	−1412.7
$\{123\}$	0.040	0.620	−0.133	0.974	−1416.1

Example 2 with BIC correction

```
octave:1> mytest(50,[0.1,0.5,0.5,0.9],4)
Parameter scaled ML log probabilities:
Order 0:  logpr = -36.613371
Order 1:  logpr = -18.458468
Order 2:  logpr = -22.707436
Order 3:  logpr = -30.833529
Order 4:  logpr = -46.741170

octave:2> mytest(200,[0.1,0.5,0.5,0.9],4)
Parameter scaled ML log probabilities:
Order 0:  logpr = -140.066143
Order 1:  logpr = -108.247838
Order 2:  logpr = -98.589565
Order 3:  logpr = -105.925987
Order 4:  logpr = -122.932541
```

BIC correction

The examples indicate that the correct model (order) is recovered, basically by using an ML selection criterion with an additional penalty term for the model complexity.

However, this BIC is derived as an approximation to the true Bayesian a-posteriori probability.

A better justification for the BIC should exist!