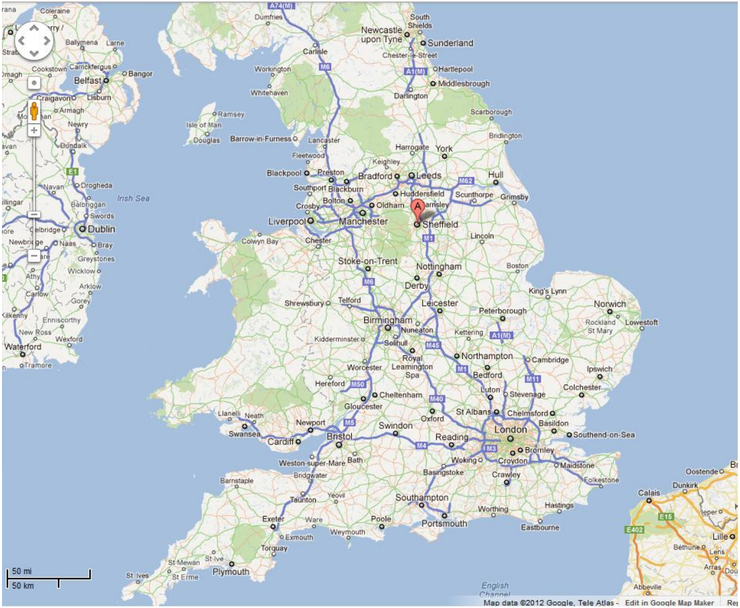
Probabilistic Models for Learning Data Representations

Andreas Damianou

Department of Computer Science, University of Sheffield, UK

IBM Research, Nairobi, Kenya, 23/06/2015

Sheffield


Coleraine
Ballymena
Larne
Cookstown
Carrickfergus
Belfast
Bangor
Craigavon
Lisburn
Armagh
Newry
Dundalk
Drogheda
Navan
Balbriggan
Swords
Dublin
Naas
Bray
Greystones
Wicklow
Arklow
Carlow
Gorey
Enniscorthy
New Ross
Wexford
Waterford
Tramore
Irish Sea
Dumfries
Carlisle
Workington
Whitehaven
Ramsey
Isle of Man
Douglas
Barrow-in-Furness
Lancaster
Fleetwood
Blackpool
Preston
Southport
Blackburn
Bolton
Crosby
Liverpool
Manchester
Oldham
Colwyn Bay
Chester
Newcastle upon Tyne
South Shields
Sunderland
Chester-le-Street
Hartlepool
Middlesbrough
Darlington
Scarborough
Bridlington
Harrogate
York
Keighley
Bradford
Leeds
Hull
Huddersfield
Scunthorpe
Grimsby
Sheffield
Lincoln
Stoke-on-Trent
Nottingham
Derby
Boston
King's Lynn
Norwich
Lowestoft
Shrewsbury
Telford
Leicester
Peterborough
Birmingham
Nuneaton
Kettering
Kidderminster
Solihull
Northampton
Royal Leamington Spa
Worcester
Cambridge
Ipswich
Bedford
Hereford
Cheltenham
Gloucester
Luton
Stevenage
Colchester
St Albans
Chelmsford
Basildon
Oxford
Swindon
London
Southend-on-Sea
Llanelli
Neath
Swansea
Newport
Cardiff
Bristol
Reading
Bromley
Croydon
Woking
Maidstone
Weston-super-Mare
Bath
Basingstoke
Crawley
Folkestone
Barnstaple
Bridgwater
Taunton
Yeovil
Southampton
Hastings
Exeter
Poole
Worthing
Portsmouth
Eastbourne
Weymouth
Exmouth
Torquay
Plymouth
St Ives
English Channel
Oostende
Dunkirk
Calais
Lille
Lens
Arras
Abbeville
50 mi
50 km
Map data ©2012 Google, Tele Atlas - Edit in Google Map Maker

SITraN

Outline

- Part 1: Probabilistic Models
  - Defining, fitting and using probabilistic models
  - Latent Variables
- Part 2: Gaussian processes
  - GPs as infinite dimensional Gaussian distributions
  - Unsupervised GPs: GP-LVM
- Part 3: Multiple views: MRD
- Summary

Outline

Part 1: Probabilistic Models
- Defining, fitting and using probabilistic models
- Latent Variables

Part 2: Gaussian processes
- GPs as infinite dimensional Gaussian distributions
- Unsupervised GPs: GP-LVM

Part 3: Multiple views: MRD

Summary

Probabilistic Models

"Probabilistic modelling involves the determination of a statistical model, a method for fitting that model to observed data, and a method for using the fitted model to solve the task at hand."

D. Blei, D. Mimno

Treating Data as Random Variables

	1	2	…	10000
1	0.81472	0.27603	…	0.58225
2	0.90579	0.6797		0.54074
3	0.12699	0.6551		0.86994
4	0.91338	0.16261		0.26478
5	0.63236	0.119		0.31807
⋮	⋮			⋮
50	0.75469	0.33712	…	0.64555

Treating Data as Random Variables

	1	2	⋯	10000
1	0.81472	0.27603	⋯	0.58225
2	0.90579	0.6797		0.54074
3	0.12699	0.6551		0.86994
4	0.91338	0.16261		0.26478
5	0.63236	0.119		0.31807
⋮	⋮			⋮
50	0.75469	0.33712	⋯	0.64555

$$\Downarrow$$

$$\mathbf{Y}$$

Treating Data as Random Variables

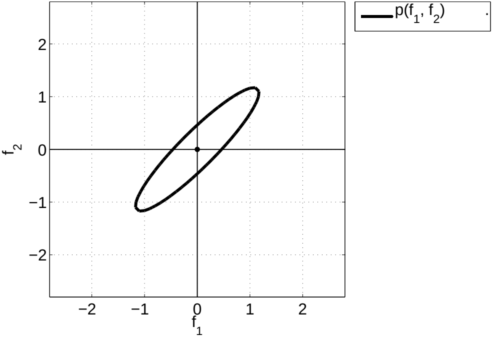
	1	2	$\cdots$	10000
1	0.81472	0.27603	$\cdots$	0.58225
2	0.90579	0.6797		0.54074
3	0.12699	0.6551		0.86994
4	0.91338	0.16261		0.26478
5	0.63236	0.119		0.31807
	$\vdots$			$\vdots$
50	0.75469	0.33712	$\cdots$	0.64555

$$\Downarrow$$
$$\mathbf{Y}$$

$$p(\mathbf{Y}) = ?$$

Gaussian distribution

Probability model: $p(f_1, f_2) \sim \mathcal{N}(\mathbf{0}, \mathbf{K})$

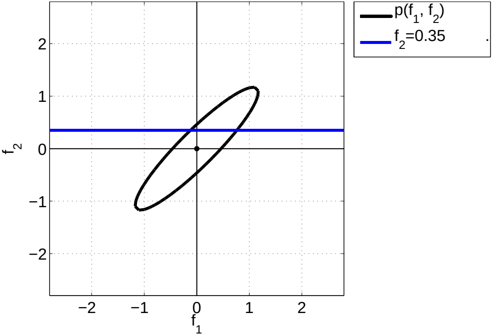


Covariance between f_1 and f_2:

$$\mathbf{K} = \begin{bmatrix} 1 & 0.92 \\ 0.92 & 1 \end{bmatrix}$$

Gaussian distribution

Probability model: $p(f_1, f_2) \sim \mathcal{N}(\mathbf{0}, \mathbf{K})$

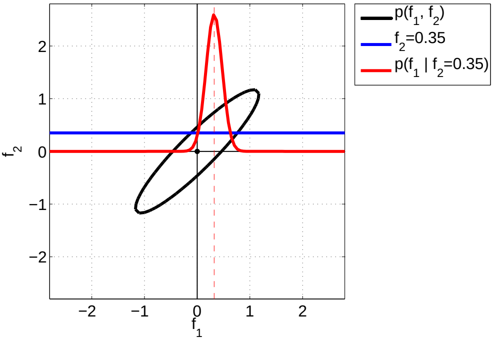


Covariance between f_1 and f_2:

$$\mathbf{K} = \begin{bmatrix} 1 & 0.92 \\ 0.92 & 1 \end{bmatrix}$$

Gaussian distribution

Probability model: $p(f_1, f_2) \sim \mathcal{N}(\mathbf{0}, \mathbf{K})$

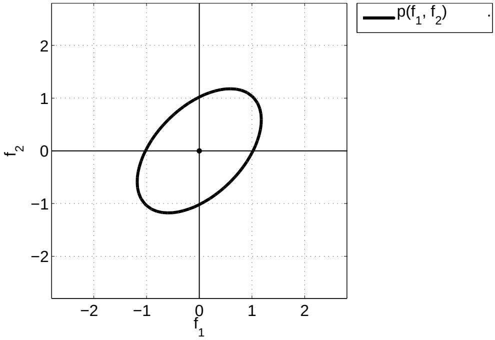


Covariance between f_1 and f_2:

$$\mathbf{K} = \begin{bmatrix} 1 & 0.92 \\ 0.92 & 1 \end{bmatrix}$$

Gaussian distribution

Probability model: $p(f_1, f_2) \sim \mathcal{N}(\mathbf{0}, \mathbf{K})$

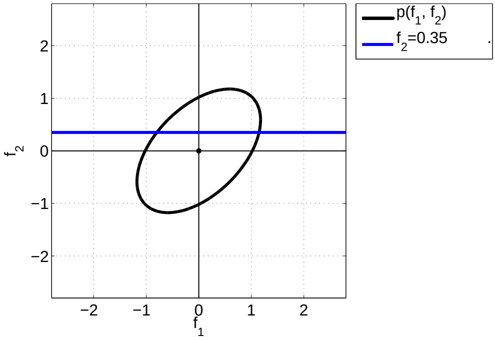


Covariance between f_1 and f_2:

$$\mathbf{K} = \begin{bmatrix} 1 & 0.5 \\ 0.5 & 1 \end{bmatrix}$$

Gaussian distribution

Probability model: $p(f_1, f_2) \sim \mathcal{N}(\mathbf{0}, \mathbf{K})$

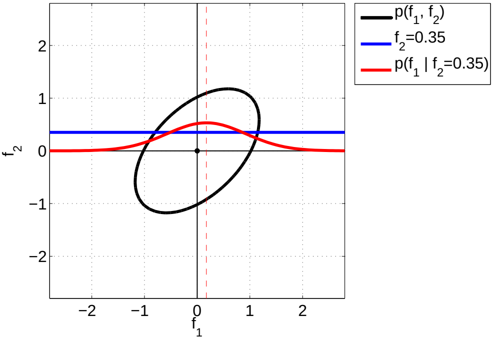


Covariance between f_1 and f_2:

$$\mathbf{K} = \begin{bmatrix} 1 & 0.5 \\ 0.5 & 1 \end{bmatrix}$$

Gaussian distribution

Probability model: $p(f_1, f_2) \sim \mathcal{N}(\mathbf{0}, \mathbf{K})$

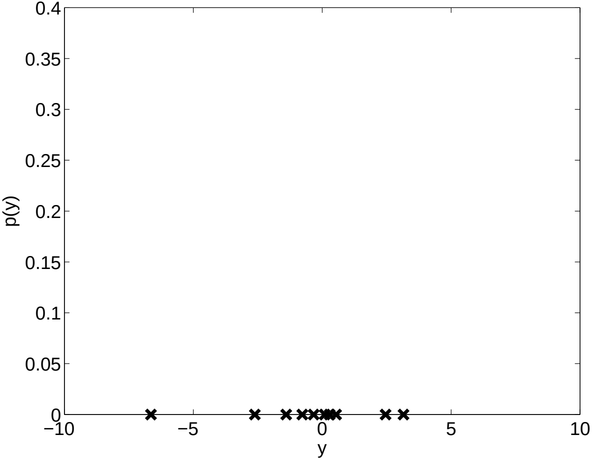


Covariance between f_1 and f_2:

$$\mathbf{K} = \begin{bmatrix} 1 & 0.5 \\ 0.5 & 1 \end{bmatrix}$$

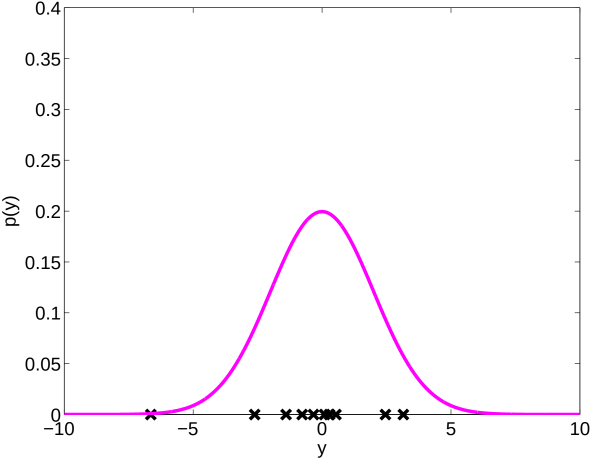
Model fitting

Which distribution (Hypothesis, $\mathcal{H}$) best *explains/fits* the data?



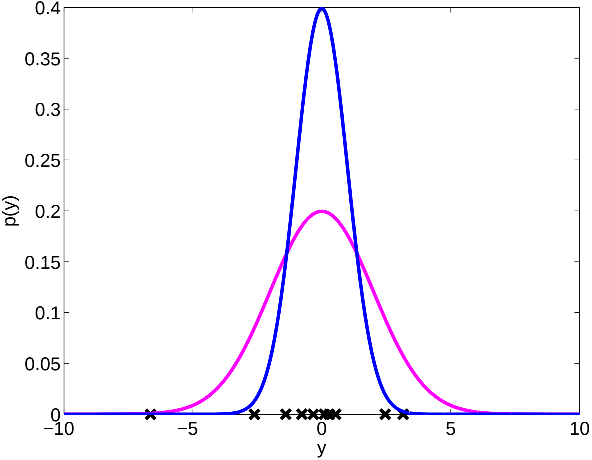
Model fitting

Which distribution (Hypothesis, $\mathcal{H}$) best *explains/fits* the data?



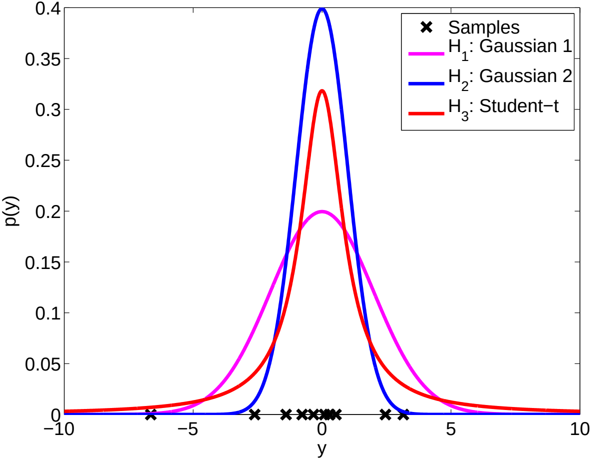
Model fitting

Which distribution (Hypothesis, $\mathcal{H}$) best *explains/fits* the data?



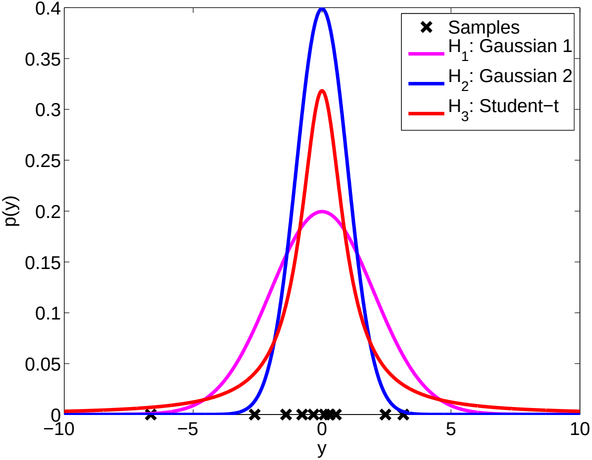
Model fitting

Which distribution (Hypothesis, $\mathcal{H}$) best *explains/fits* the data?



Model fitting

Which distribution (Hypothesis, $\mathcal{H}$) best *explains/fits* the data?



Model fitting can be done with *maximum likelihood*.

Bayes' rule

Taking things one step further: assume a model (hypothesis) $\mathcal{H}$ and a distribution for its parameters, θ.

- Assume a prior distribution for our parameters, θ.
- Assume a likelihood for the observed data, y, *given* the parameters.
- Find the posterior of the parameters, given the data.
- The normaliser of the posterior is the model evidence.
- All linked through *Bayes' rule*:

$$p(\theta|y,\mathcal{H}) = \frac{p(y|\theta,\mathcal{H})p(\theta|\mathcal{H})}{p(y|\mathcal{H}) = \int_\theta p(y|\theta,\mathcal{H})}$$

Bayes' rule

Taking things one step further: assume a model (hypothesis) $\mathcal{H}$ and a distribution for its parameters, θ.

- Assume a prior distribution for our parameters, θ.
- Assume a likelihood for the observed data, y, *given* the parameters.
- Find the posterior of the parameters, given the data.
- The normaliser of the posterior is the model evidence.
- All linked through *Bayes' rule*:

$$p(\theta|y,\mathcal{H}) = \frac{p(y|\theta,\mathcal{H})p(\theta|\mathcal{H})}{p(y|\mathcal{H}) = \int_\theta p(y|\theta,\mathcal{H})}$$

Bayes' rule

Taking things one step further: assume a model (hypothesis) $\mathcal{H}$ and a distribution for its parameters, θ.

- Assume a prior distribution for our parameters, θ.
- Assume a likelihood for the observed data, y, *given* the parameters.
- Find the posterior of the parameters, given the data.
- The normaliser of the posterior is the model evidence.
- All linked through *Bayes' rule*:

$$p(\theta|y,\mathcal{H}) = \frac{p(y|\theta,\mathcal{H})p(\theta|\mathcal{H})}{p(y|\mathcal{H}) = \int_\theta p(y|\theta,\mathcal{H})}$$

Bayes' rule

Taking things one step further: assume a model (hypothesis) $\mathcal{H}$ and a distribution for its parameters, θ.

- Assume a prior distribution for our parameters, θ.
- Assume a likelihood for the observed data, y, *given* the parameters.
- Find the posterior of the parameters, given the data.
- The normaliser of the posterior is the model evidence.
- All linked through *Bayes' rule*:

$$p(\theta|y,\mathcal{H}) = \frac{p(y|\theta,\mathcal{H})p(\theta|\mathcal{H})}{p(y|\mathcal{H}) = \int_\theta p(y|\theta,\mathcal{H})}$$

Bayes' rule

Taking things one step further: assume a model (hypothesis) $\mathcal{H}$ and a distribution for its parameters, θ.

- Assume a prior distribution for our parameters, θ.
- Assume a likelihood for the observed data, y, *given* the parameters.
- Find the posterior of the parameters, given the data.
- The normaliser of the posterior is the model evidence.
- All linked through *Bayes' rule*:

$$p(\theta|y, \mathcal{H}) = \frac{p(y|\theta, \mathcal{H})p(\theta|\mathcal{H})}{p(y|\mathcal{H}) = \int_\theta p(y|\theta, \mathcal{H})}$$

Occam's razor

"Everything should be made as simple as possible, but not simpler". *A. Einstein*

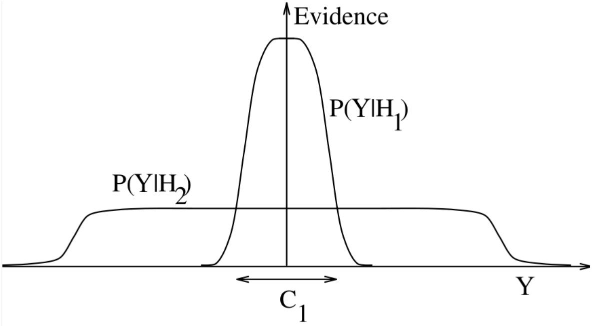


Fig. 1. This figure is reproduced with permission from MacKay (1991). It has also appeared in MacKay (1992) and MacKay (2003, chapter 28). The Y-axis indexes all possible data sets (under some idealized ordering). Each curve gives a probability distribution over data sets, so must enclose an area of 1. H_1 is a simple model focusing on data in region C_1. Given data is this region, H_1 has more evidence than a more powerful model H_2, which would be favored given more complex data (outside C_1). [Murray and Ghahramani, 2001]

Latent Variables

- What are the *latent* features of “cuteness”?

Another example: latent *process*

Is Beckham an expert in Newtonian & trajectory mechanics?

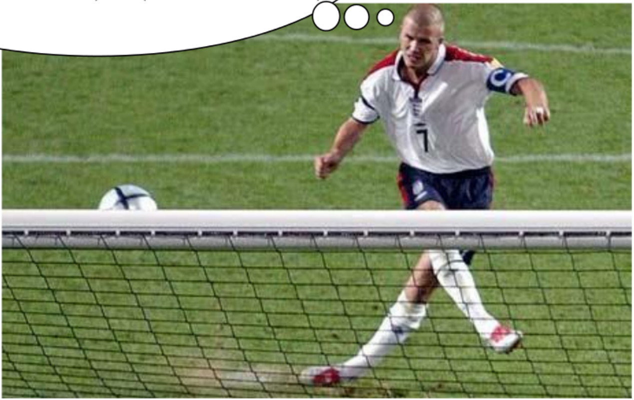
Another example: latent *process*

Is Beckham an expert in Newtonian & trajectory mechanics?

$$m\frac{\mathrm{d}^2\vec{x}(t)}{\mathrm{d}t^2} = -\nabla V(\vec{x}(t)), \quad \vec{x} = (x, y, z)$$

$$R_s = \sqrt{x^2 + y^2}$$

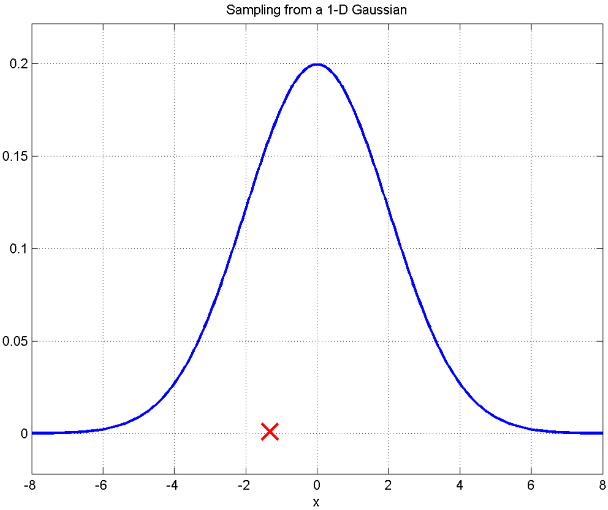
$$= \sqrt{\left(\frac{2v^2\cos^2\theta}{g}\left(\frac{\sin\theta}{\cos\theta} - m\right)\right)^2 + \left(m\frac{2v^2\cos^2\theta}{g}\left(\frac{\sin\theta}{\cos\theta} - m\right)\right)^2}$$



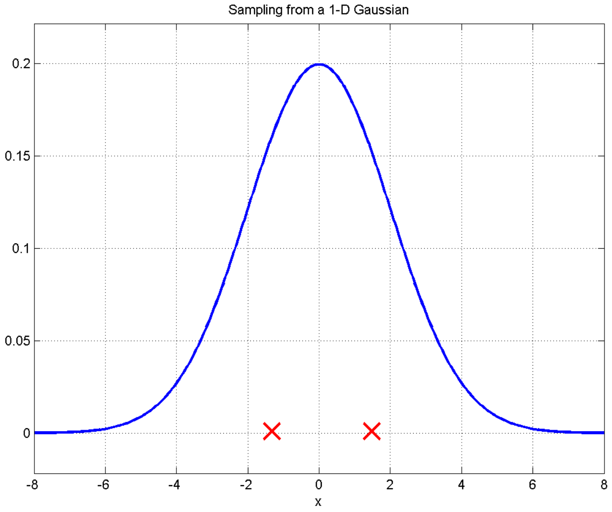
Outline

Introducing Gaussian Processes:

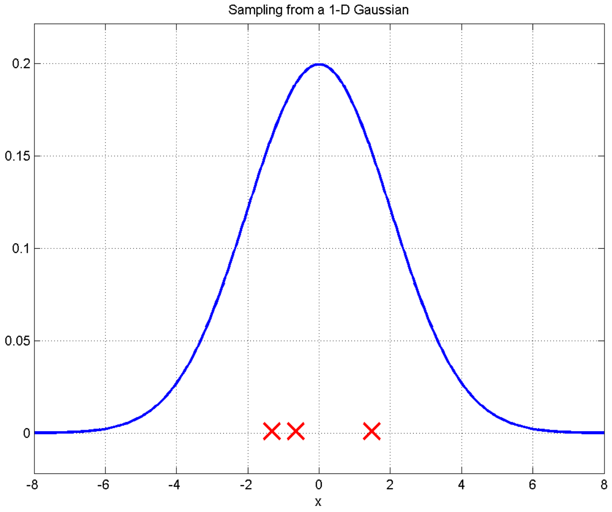
- A Gaussian distribution depends on a mean and a covariance matrix.
- A Gaussian process depends on a mean and a covariance function.



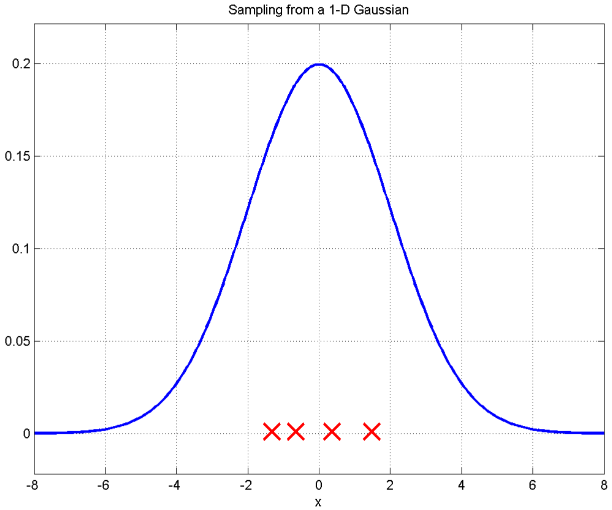
Sampling from a 1-D Gaussian
0.2
0.15
0.1
0.05
0
-8
-6
-4
-2
0
2
4
6
8
x

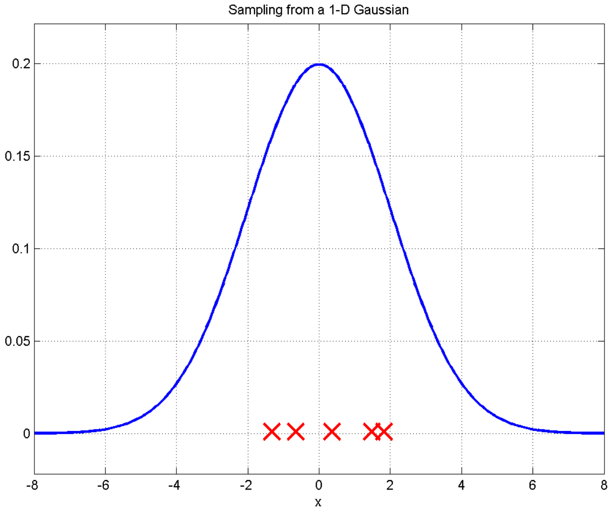


Sampling from a 1-D Gaussian
0.2
0.15
0.1
0.05
0
-8
-6
-4
-2
0
2
4
6
8
x

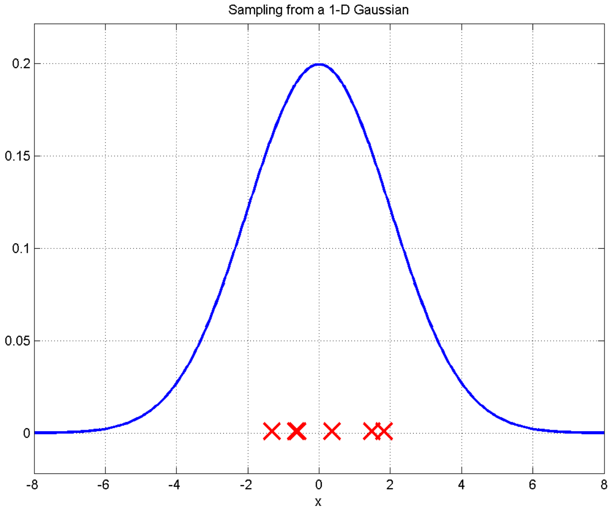


Sampling from a 1-D Gaussian
0.2
0.15
0.1
0.05
0
-8
-6
-4
-2
0
2
4
6
8
x

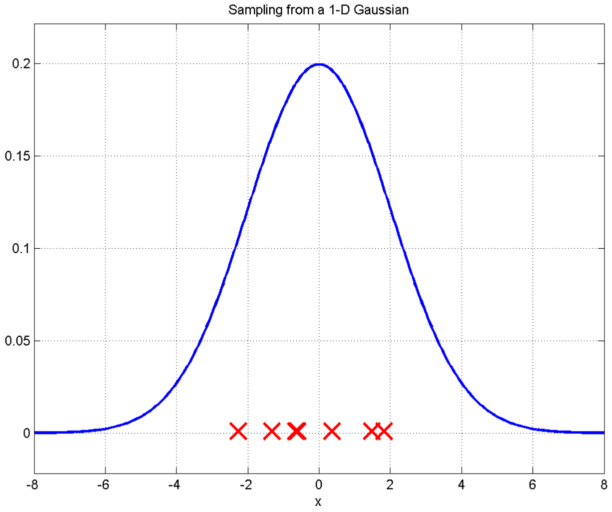

Sampling from a 1-D Gaussian
0.2
0.15
0.1
0.05
0
-8
-6
-4
-2
0
2
4
6
8
x



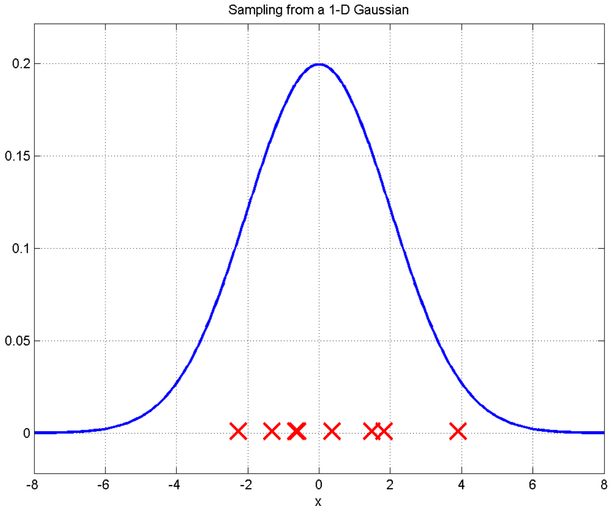
Sampling from a 1-D Gaussian
0.2
0.15
0.1
0.05
0
-8
-6
-4
-2
0
2
4
6
8
x



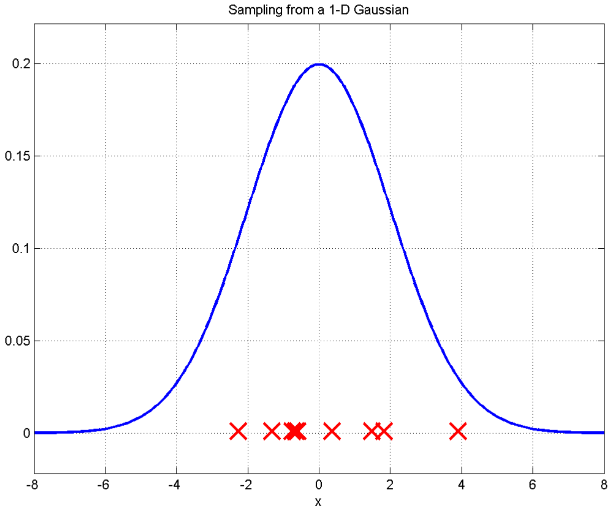
Sampling from a 1-D Gaussian
0.2
0.15
0.1
0.05
0
-8
-6
-4
-2
0
2
4
6
8
x



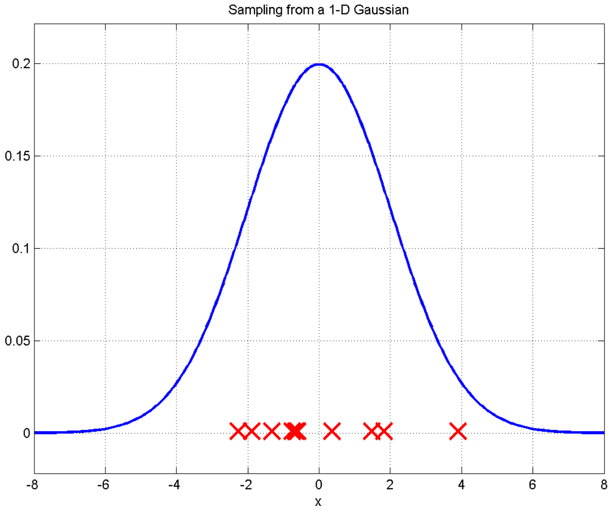
Sampling from a 1-D Gaussian
0.2
0.15
0.1
0.05
0
-8
-6
-4
-2
0
2
4
6
8
x

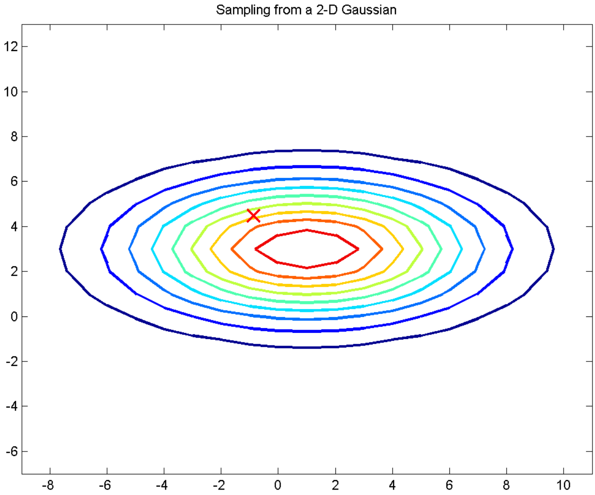


Sampling from a 1-D Gaussian
0.2
0.15
0.1
0.05
0
-8
-6
-4
-2
0
2
4
6
8
x

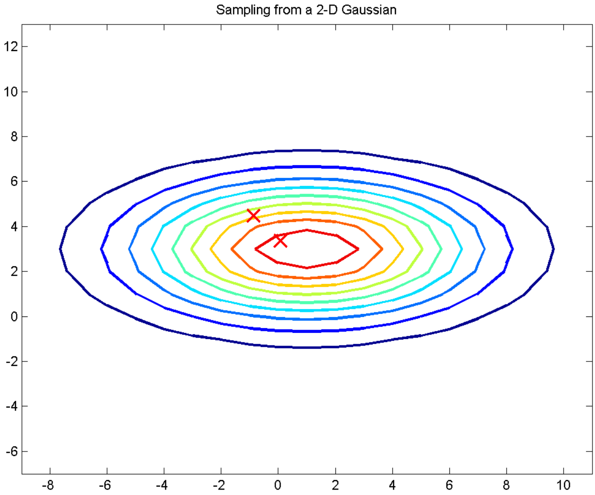


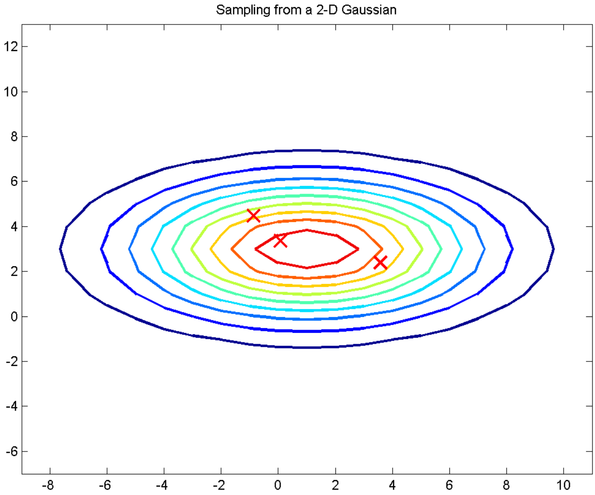
Sampling from a 1-D Gaussian
0.2
0.15
0.1
0.05
0
-8
-6
-4
-2
0
2
4
6
8
x

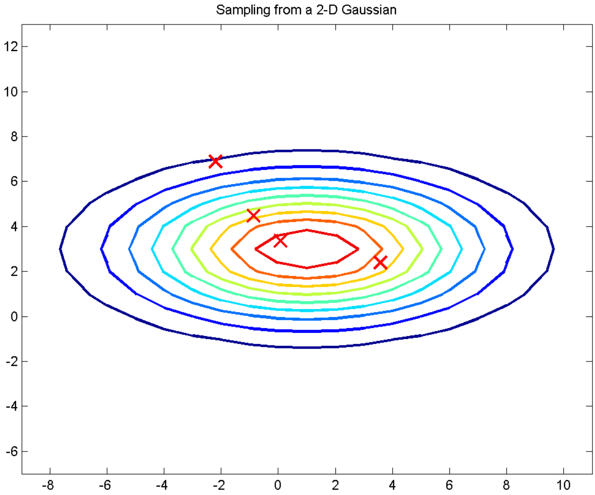

Sampling from a 1-D Gaussian
0.2
0.15
0.1
0.05
0
-8
-6
-4
-2
0
2
4
6
8
x

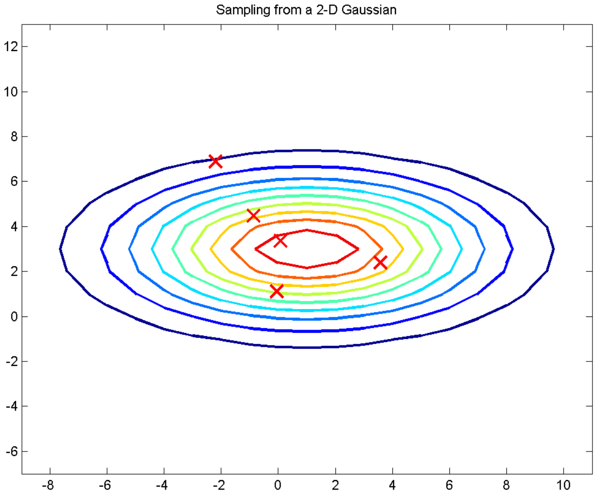


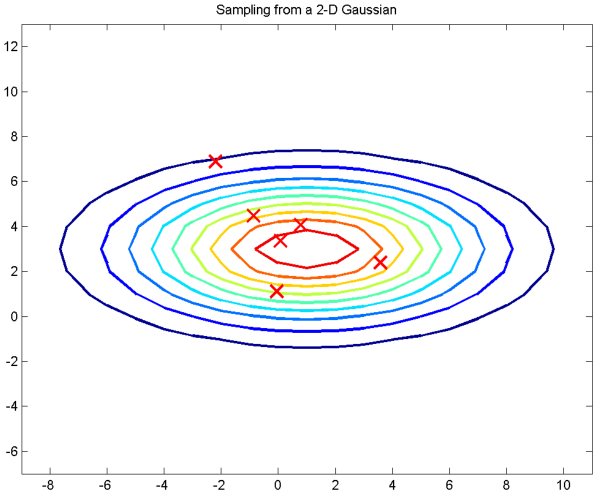
Sampling from a 2-D Gaussian
12
10
8
6
4
2
0
-2
-4
-6
-8
-6
-4
-2
0
2
4
6
8
10

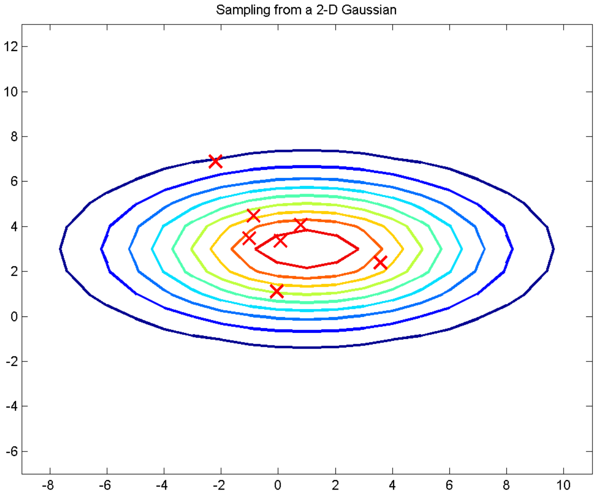

Sampling from a 2-D Gaussian
12
10
8
6
4
2
0
-2
-4
-6
-8
-6
-4
-2
0
2
4
6
8
10


Sampling from a 2-D Gaussian
12
10
8
6
4
2
0
-2
-4
-6
-8
-6
-4
-2
0
2
4
6
8
10

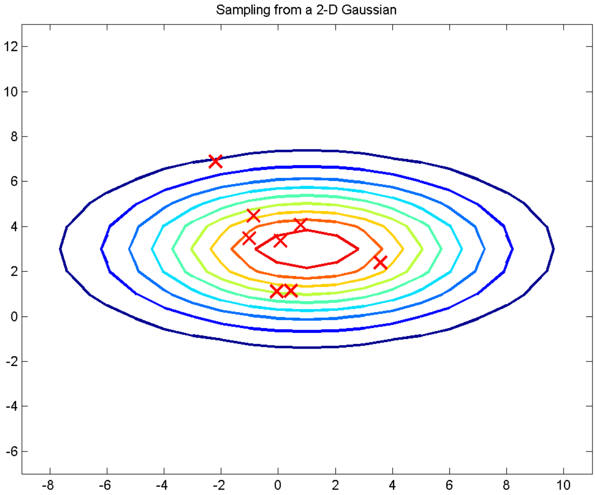

Sampling from a 2-D Gaussian
12
10
8
6
4
2
0
-2
-4
-6
-8
-6
-4
-2
0
2
4
6
8
10


Sampling from a 2-D Gaussian
12
10
8
6
4
2
0
-2
-4
-6
-8
-6
-4
-2
0
2
4
6
8
10

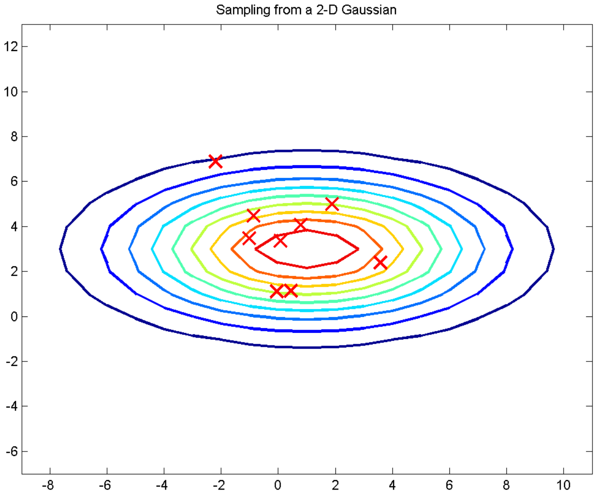

Sampling from a 2-D Gaussian
12
10
8
6
4
2
0
-2
-4
-6
-8
-6
-4
-2
0
2
4
6
8
10

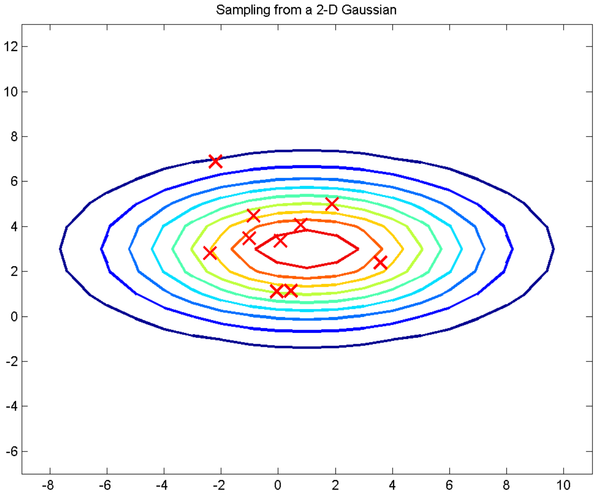


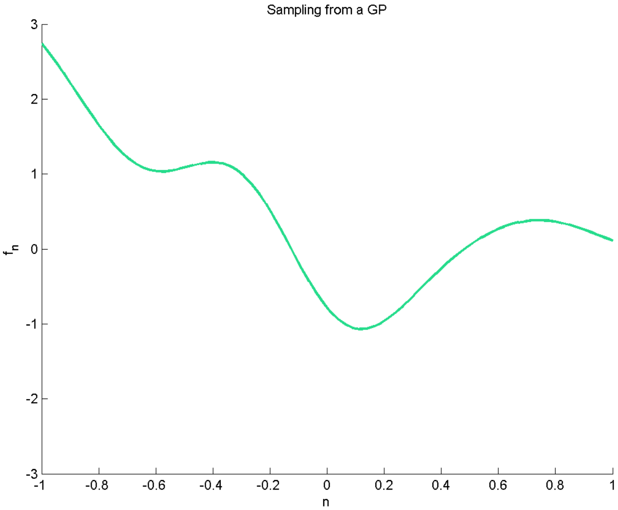
Sampling from a 2-D Gaussian
12
10
8
6
4
2
0
-2
-4
-6
-8
-6
-4
-2
0
2
4
6
8
10



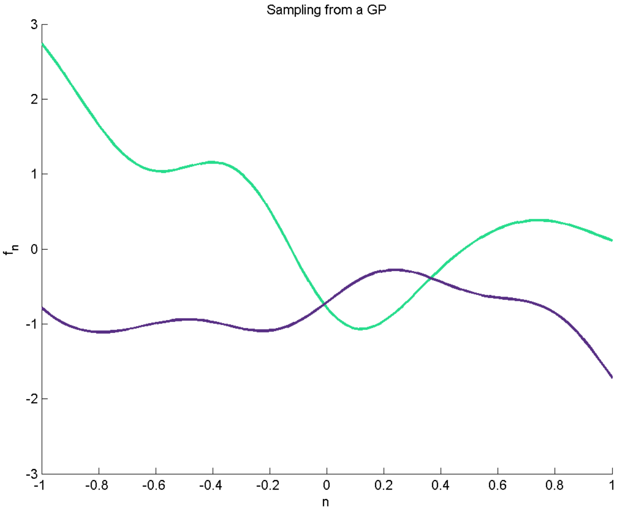
Sampling from a 2-D Gaussian
12
10
8
6
4
2
0
-2
-4
-6
-8
-6
-4
-2
0
2
4
6
8
10

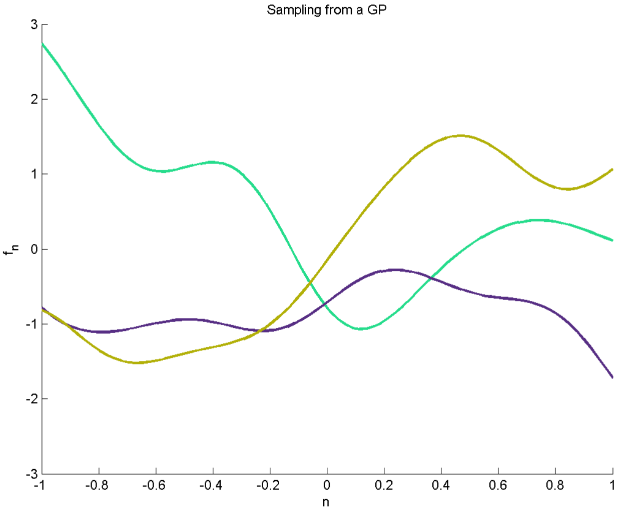

Sampling from a 2-D Gaussian
12
10
8
6
4
2
0
-2
-4
-6
-8
-6
-4
-2
0
2
4
6
8
10


Sampling from a 2-D Gaussian
12
10
8
6
4
2
0
-2
-4
-6
-8
-6
-4
-2
0
2
4
6
8
10

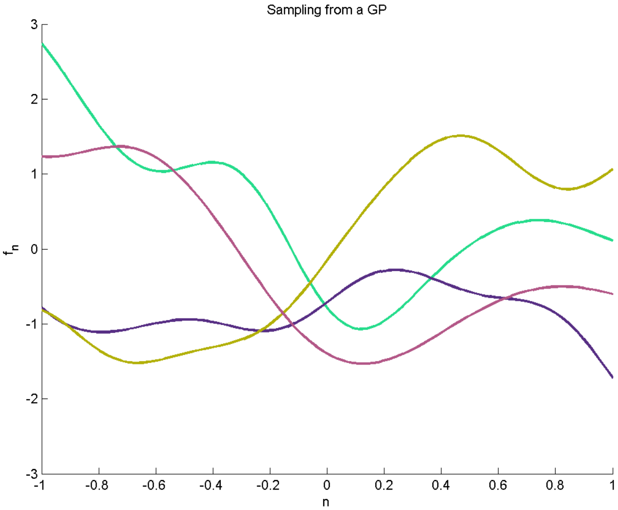


Sampling from a GP
3
2
1
0
-1
-2
-3
f_n
-1
-0.8
-0.6
-0.4
-0.2
0
0.2
0.4
0.6
0.8
1
n

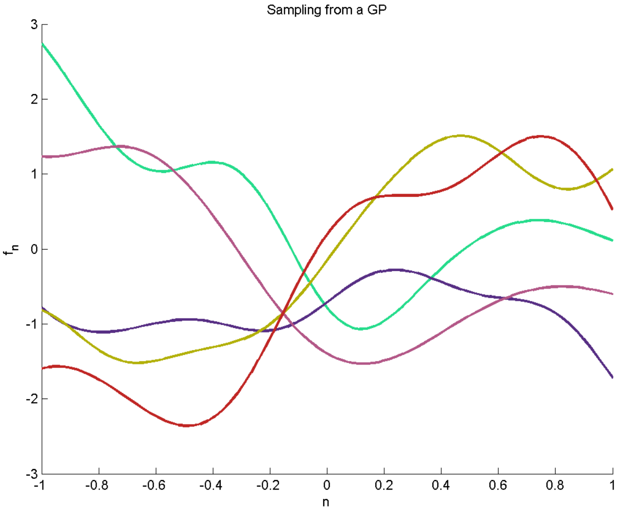

Sampling from a GP
3
2
1
0
-1
-2
-3
f_n
-1
-0.8
-0.6
-0.4
-0.2
0
0.2
0.4
0.6
0.8
1
n



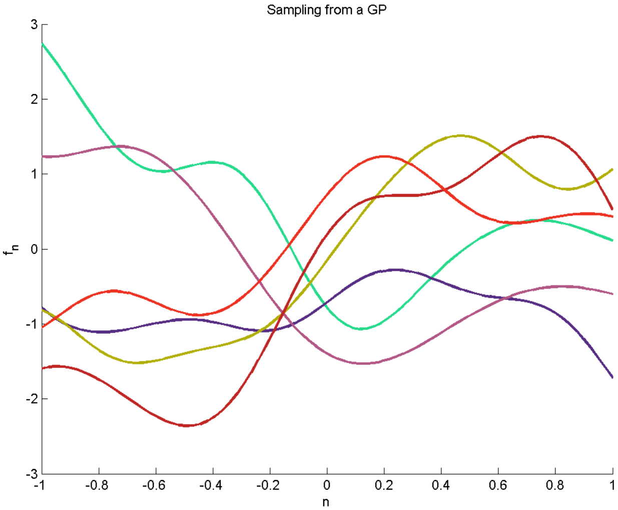
Sampling from a GP
3
2
1
0
-1
-2
-3
f_n
-1
-0.8
-0.6
-0.4
-0.2
0
0.2
0.4
0.6
0.8
1
n



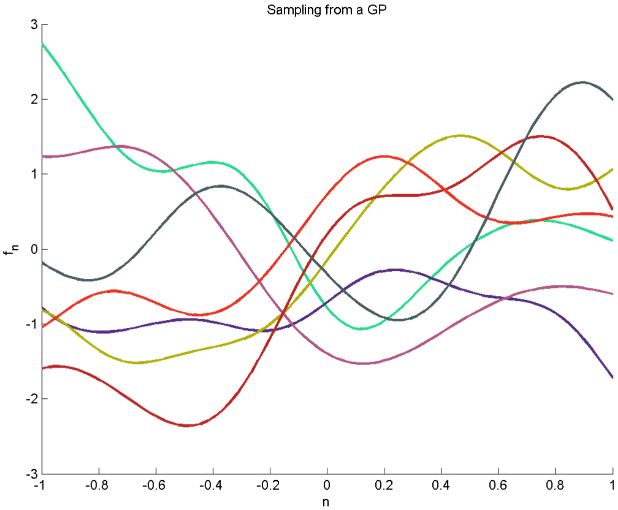
Sampling from a GP
3
2
1
0
-1
-2
-3
f_n
-1
-0.8
-0.6
-0.4
-0.2
0
0.2
0.4
0.6
0.8
1
n



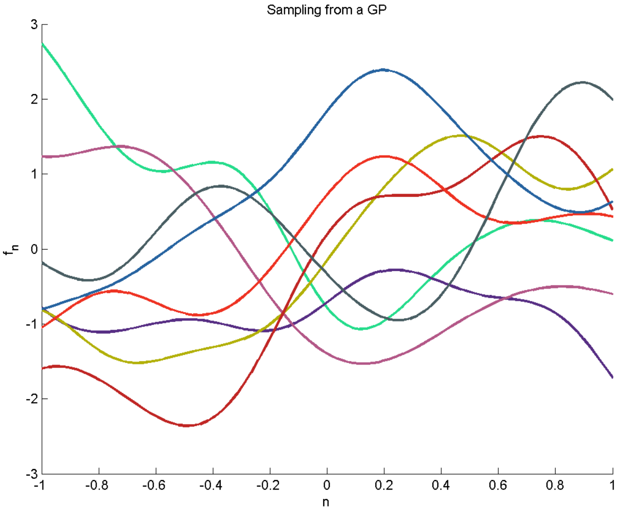
Sampling from a GP
3
2
1
0
-1
-2
-3
f_n
-1
-0.8
-0.6
-0.4
-0.2
0
0.2
0.4
0.6
0.8
1
n



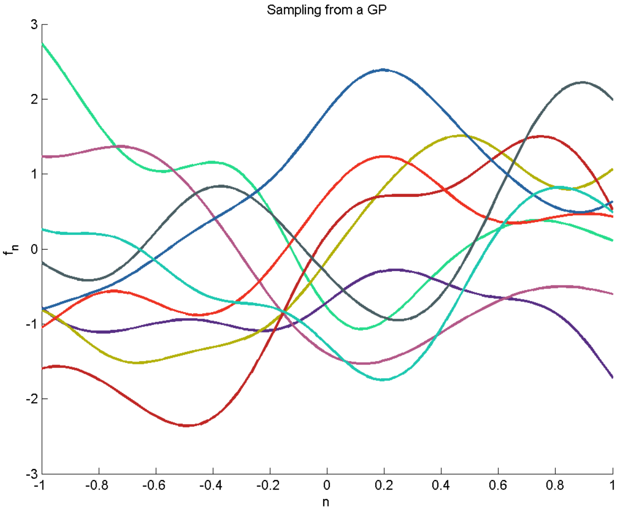
Sampling from a GP
3
2
1
0
-1
-2
-3
f_n
-1
-0.8
-0.6
-0.4
-0.2
0
0.2
0.4
0.6
0.8
1
n



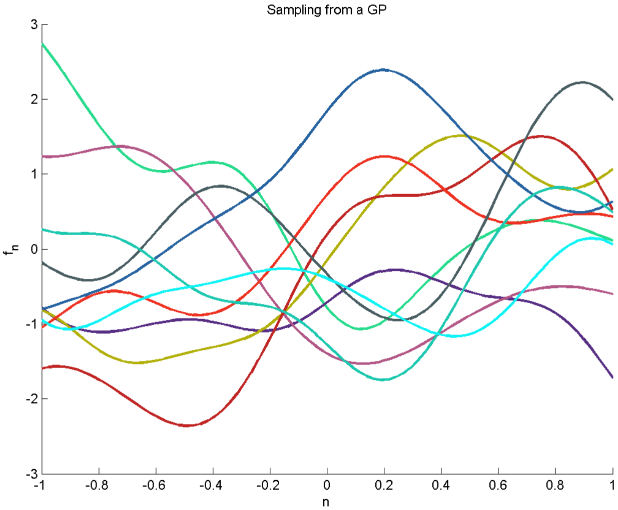
Sampling from a GP
3
2
1
0
-1
-2
-3
f_n
-1
-0.8
-0.6
-0.4
-0.2
0
0.2
0.4
0.6
0.8
1
n



Sampling from a GP
3
2
1
0
-1
-2
-3
f_n
-1
-0.8
-0.6
-0.4
-0.2
0
0.2
0.4
0.6
0.8
1
n



Sampling from a GP
3
2
1
0
-1
-2
-3
f_n
-1
-0.8
-0.6
-0.4
-0.2
0
0.2
0.4
0.6
0.8
1
n



Sampling from a GP
3
2
1
0
-1
-2
-3
f_n
-1
-0.8
-0.6
-0.4
-0.2
0
0.2
0.4
0.6
0.8
1
n

Infinite model... but we *always* work with finite sets!

Let's start with a multivariate Gaussian:

$$p(\underbrace{f_1, f_2, \cdots, f_s}_{\mathbf{f}_A}, \underbrace{f_{s+1}, f_{s+2}, \cdots, f_N}_{\mathbf{f}_B}) \sim \mathcal{N}(\boldsymbol{\mu}, \mathbf{K}).$$

with:

$$\boldsymbol{\mu} = \begin{bmatrix} \boldsymbol{\mu}_A \\ \boldsymbol{\mu}_B \end{bmatrix} \text{ and } \mathbf{K} = \begin{bmatrix} \mathbf{K}_{AA} & \mathbf{K}_{AB} \\ \mathbf{K}_{BA} & \mathbf{K}_{BB} \end{bmatrix}$$

Marginalisation property:

$$p(\mathbf{f}_A, \mathbf{f}_B) \sim \mathcal{N}(\boldsymbol{\mu}, \mathbf{K}). \quad \text{Then:}$$

$$p(\mathbf{f}_A) = \int_{\mathbf{f}_B} p(\mathbf{f}_A, \mathbf{f}_B) \mathrm{d}\mathbf{f}_B = \mathcal{N}(\boldsymbol{\mu}_A, \mathbf{K}_{AA})$$

Infinite model... but we *always* work with finite sets!

Let's start with a multivariate Gaussian:

$$p(\underbrace{f_1, f_2, \cdots, f_s}_{\mathbf{f}_A}, \underbrace{f_{s+1}, f_{s+2}, \cdots, f_N}_{\mathbf{f}_B}) \sim \mathcal{N}(\boldsymbol{\mu}, \mathbf{K}).$$

with:

$$\boldsymbol{\mu} = \begin{bmatrix} \boldsymbol{\mu}_A \\ \boldsymbol{\mu}_B \end{bmatrix} \text{ and } \mathbf{K} = \begin{bmatrix} \mathbf{K}_{AA} & \mathbf{K}_{AB} \\ \mathbf{K}_{BA} & \mathbf{K}_{BB} \end{bmatrix}$$

Marginalisation property:

$$p(\mathbf{f}_A, \mathbf{f}_B) \sim \mathcal{N}(\boldsymbol{\mu}, \mathbf{K}). \quad \text{Then:}$$

$$p(\mathbf{f}_A) = \int_{\mathbf{f}_B} p(\mathbf{f}_A, \mathbf{f}_B) d\mathbf{f}_B = \mathcal{N}(\boldsymbol{\mu}_A, \mathbf{K}_{AA})$$

Infinite model... but we *always* work with finite sets!

In the GP context:

$$\boldsymbol{\mu}_\infty = \begin{bmatrix} \boldsymbol{\mu}_{\mathbf{x}} \\ \cdots \\ \cdots \end{bmatrix} \quad \text{and} \quad \mathbf{K}_\infty = \begin{bmatrix} \mathbf{K}_{\mathbf{x}\mathbf{x}} & \cdots \\ \cdots & \cdots \end{bmatrix}$$

Posterior is also Gaussian!

$$p(\mathbf{f}_A, \mathbf{f}_B) \sim \mathcal{N}(\boldsymbol{\mu}, \mathbf{K}). \quad \text{Then:}$$
$$p(\mathbf{f}_A | \mathbf{f}_B) = \mathcal{N}(\cdots, \cdots)$$

In the GP context this can be used for inter/extrapolation:

$$p(f_* | f_1, \cdots, f_N) = p(f(x_*) | f(x_1), \cdots, f(x_N)) \sim \mathcal{N}$$

But where is $\mathbf{K}_{\cdot,\cdot}$ coming from in GPs?

Posterior is also Gaussian!

$$p(\mathbf{f}_A, \mathbf{f}_B) \sim \mathcal{N}(\boldsymbol{\mu}, \mathbf{K}). \quad \text{Then:}$$
$$p(\mathbf{f}_A | \mathbf{f}_B) = \mathcal{N}(\cdots, \cdots)$$

In the GP context this can be used for inter/extrapolation:

$$p(f_* | f_1, \cdots, f_N) = p(f(x_*) | f(x_1), \cdots, f(x_N)) \sim \mathcal{N}$$

But where is $\mathbf{K}_{\cdot,\cdot}$ coming from in GPs?

Posterior is also Gaussian!

$$p(\mathbf{f}_A, \mathbf{f}_B) \sim \mathcal{N}(\boldsymbol{\mu}, \mathbf{K}). \quad \text{Then:}$$
$$p(\mathbf{f}_A | \mathbf{f}_B) = \mathcal{N}(\cdots, \cdots)$$

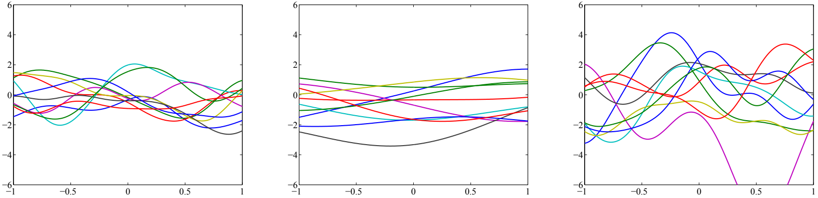
In the GP context this can be used for inter/extrapolation:

$$p(f_* | f_1, \cdots, f_N) = p(f(x_*) | f(x_1), \cdots, f(x_N)) \sim \mathcal{N}$$

But where is $\mathbf{K}_{\cdot,\cdot}$ coming from in GPs?

Covariance samples and hyperparameters

- $k(x, x') = \alpha \exp\left(-\frac{\gamma}{2}(x - x')^\top (x - x')\right)$
- The hyperparameters of the cov. function define the properties (and NOT an explicit form) of the sampled functions

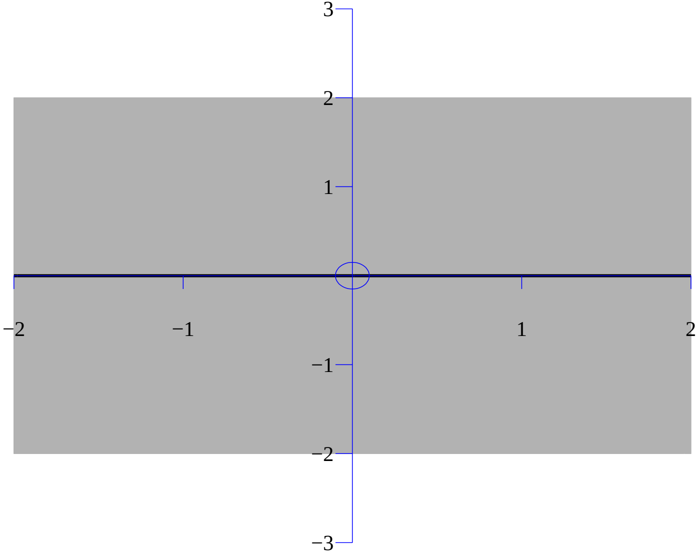


Incorporating Gaussian noise is tractable

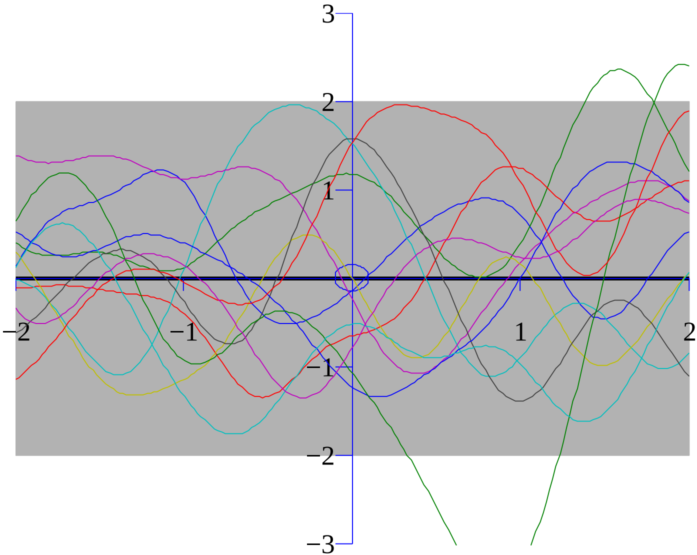
- So far we assumed: $\mathbf{f} = f(\mathbf{X})$
- Assuming that we only observe noisy versions $\mathbf{y}$ of the true outputs $\mathbf{f}$:

$$\mathbf{y} = f(\mathbf{X}) + \epsilon, \;\; \epsilon \sim \mathcal{N}(0, \sigma^2)$$

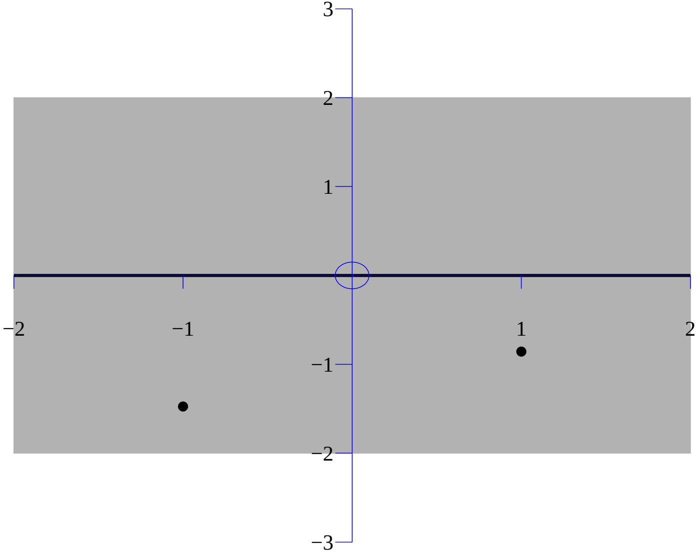
Fitting the data *(shaded area is **uncertainty**)*



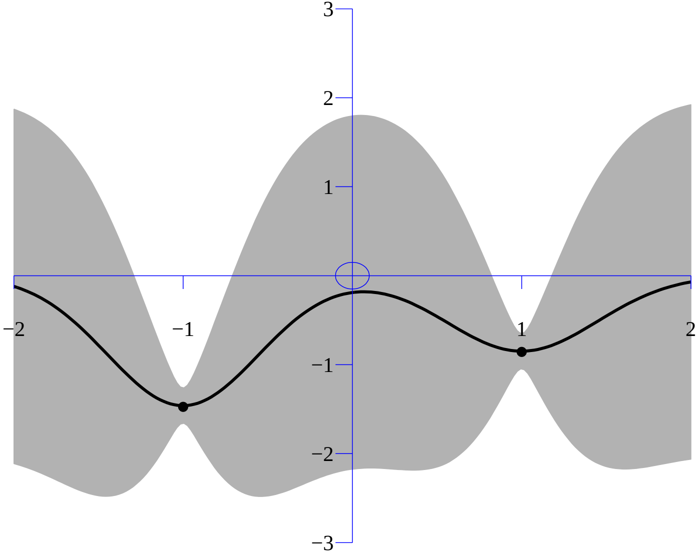
Fitting the data - Prior Samples



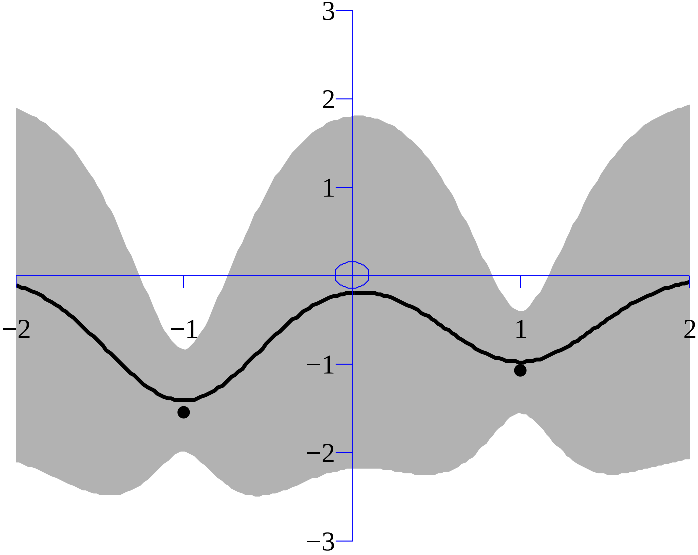
Fitting the data



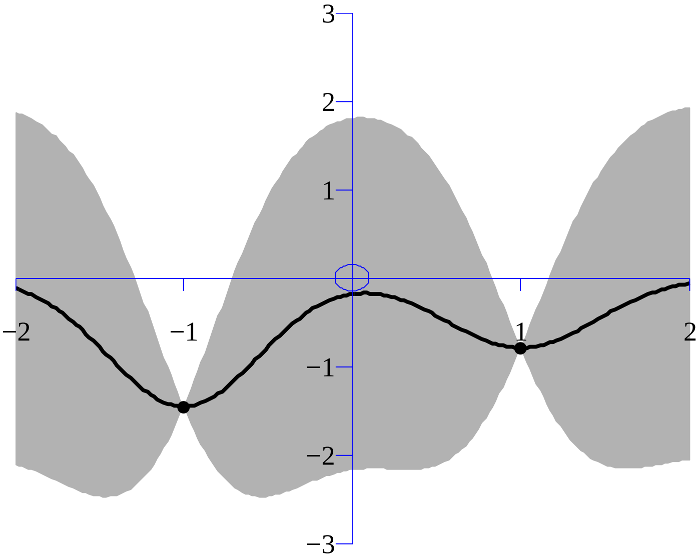
Fitting the data



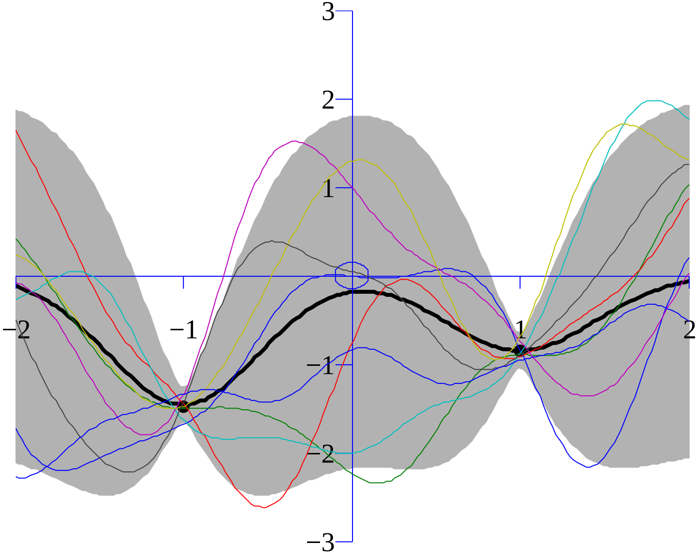
Fitting the data - more noise



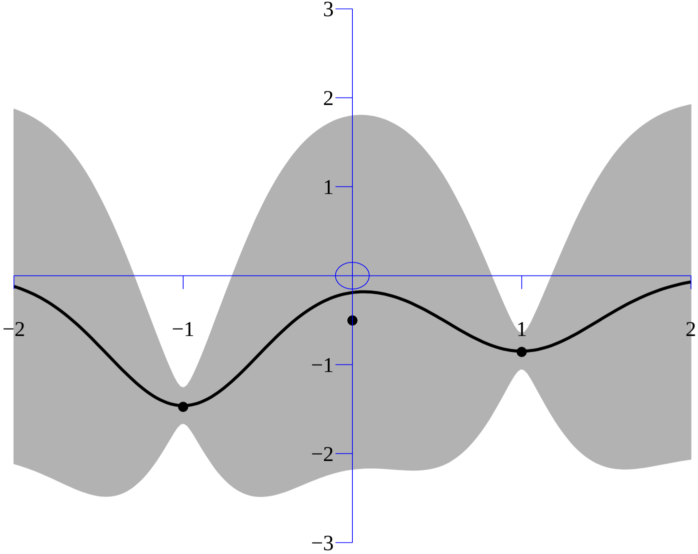
Fitting the data - no noise



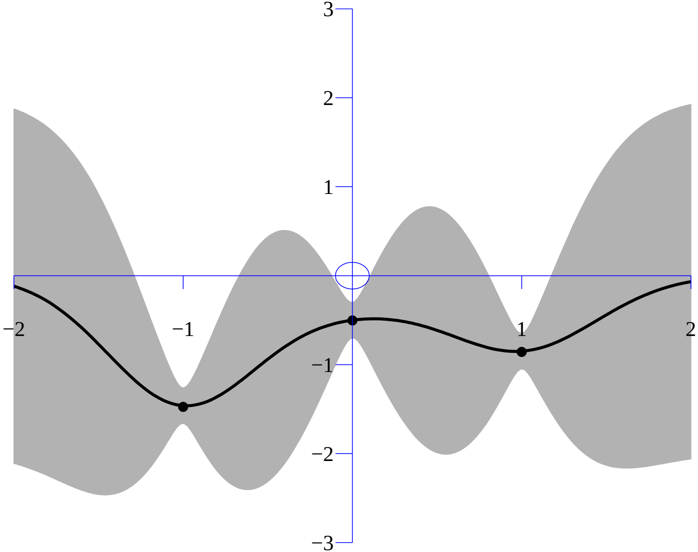
Fitting the data - Posterior samples



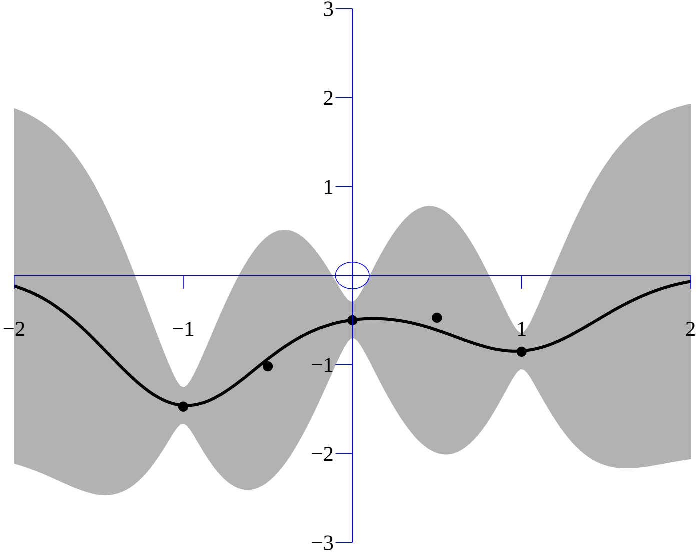
Fitting the data



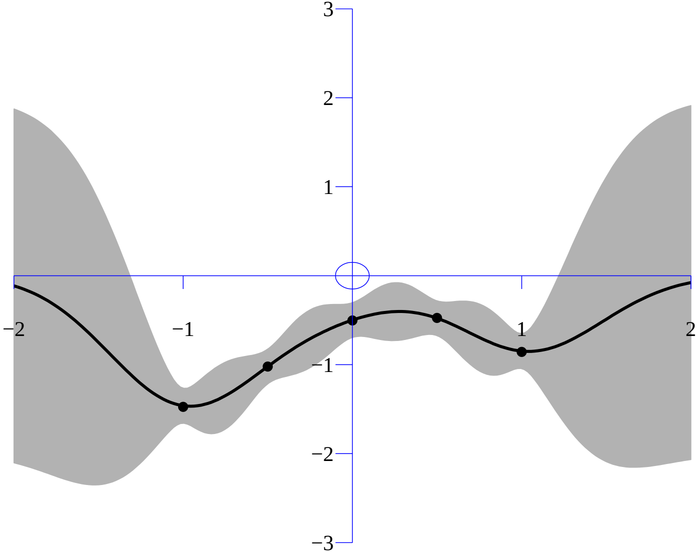
Fitting the data



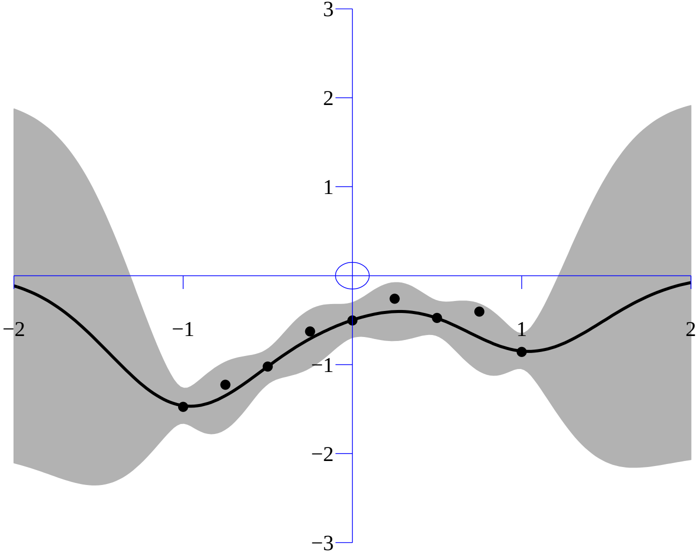
Fitting the data



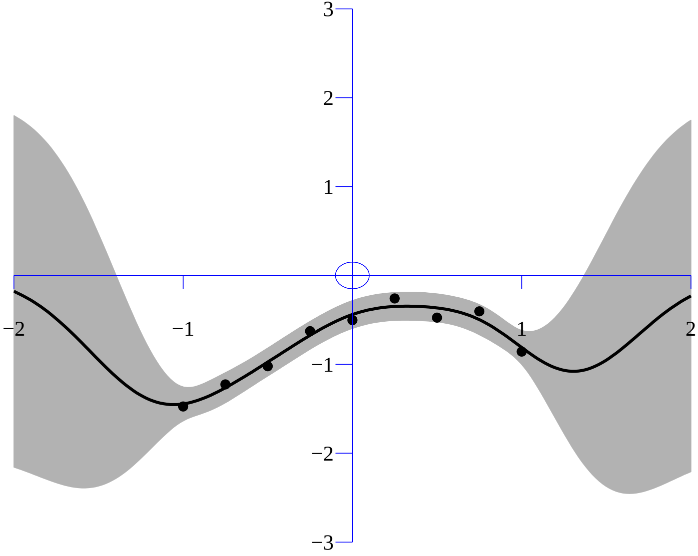
Fitting the data



Fitting the data

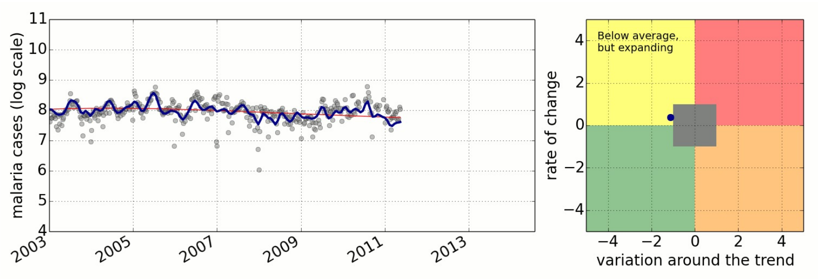


Fitting the data

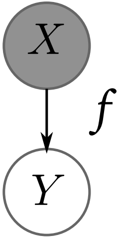


Application to Disease modelling

Ricardo Andrade Pacheco.
`http://ric70x7.github.io/research.html`

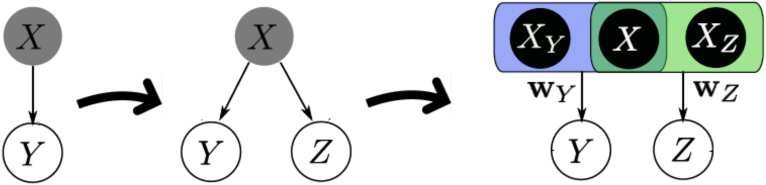


Unsupervised learning: GP-LVM

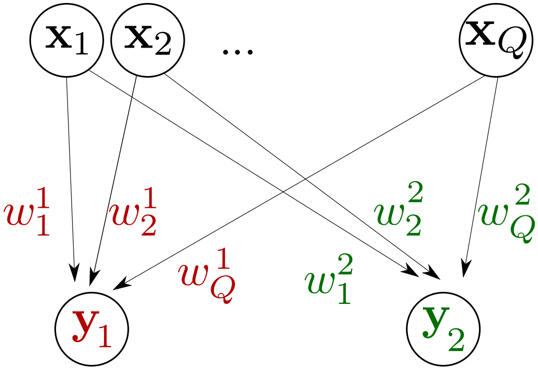


- If $\mathbf{X}$ is unobserved, treat it as a parameter and optimize over it.

Manifold Relevance Determination

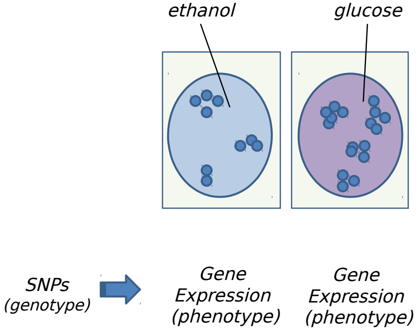


- Observations come into two different *views*: Y and Z.
- The latent space is segmented into parts private to Y, private to Z and shared between Y and Z.
- Used for data consolidation and discovering commonalities.



$\mathbf{x}_1$
$\mathbf{x}_2$
...
$\mathbf{x}_Q$
w_1^1
w_2^1
w_Q^1
w_1^2
w_2^2
w_Q^2
$\mathbf{y}_1$
$\mathbf{y}_2$

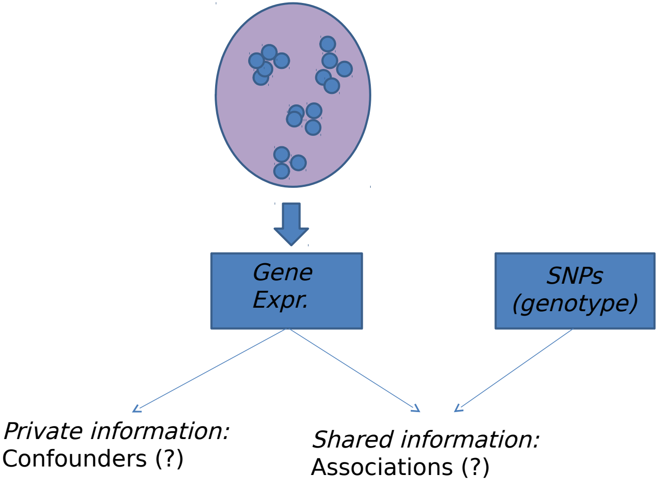
Consolidating complementary experimental data



Shared information: biological signal / confounders
Private information: environmental confounders

Confounders: Statistical relationships that do not reflect the true causality in the data

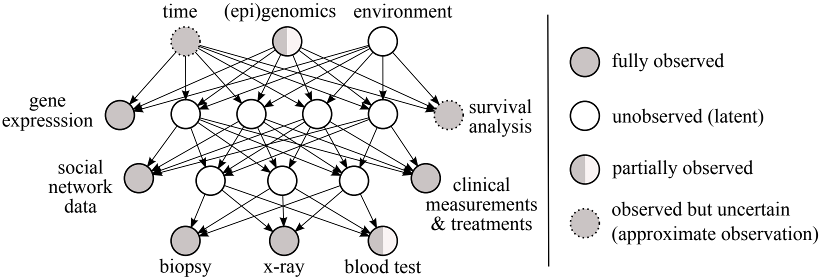
Discovering commonalities in heterogeneous data



Application to Health Modelling

Research agenda of Prof. Neil Lawrence's group:

http://sheffieldml.github.io/



Example: faces

https://youtu.be/rIPX3ClOhKY

Example: robotics

Summary

- ..

Thanks

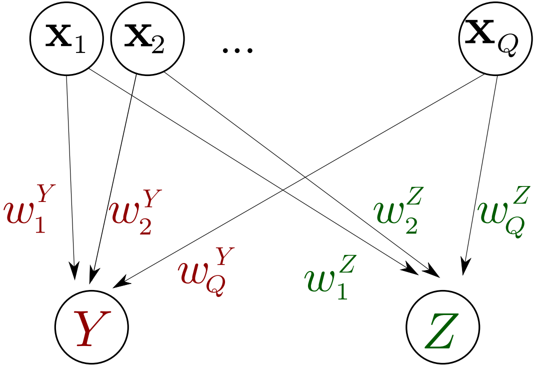
Thanks to Neil Lawrence, James Hensman, Michalis Titsias, Carl Henrik Ek.

References:

- N. D. Lawrence (2006) "The Gaussian process latent variable model" Technical Report no CS-06-03, The University of Sheffield, Department of Computer Science
- N. D. Lawrence (2006) "Probabilistic dimensional reduction with the Gaussian process latent variable model" (talk)
- C. E. Rasmussen (2008), "Learning with Gaussian Processes", Max Planck Institute for Biological Cybernetics, Published: Feb. 5, 2008 (Videolectures.net)
- Carl Edward Rasmussen and Christopher K. I. Williams. Gaussian Processes for Machine Learning. MIT Press, Cambridge, MA, 2006. ISBN 026218253X.
- M. K. Titsias (2009), "Variational learning of inducing variables in sparse Gaussian processes", AISTATS 2009
- A. C. Damianou, M. K. Titsias and N. D. Lawrence (2011), "Variational Gaussian process dynamical systems", NIPS 2011
- A. C. Damianou, C. H. Ek, M. K. Titsias and N. D. Lawrence (2012), "Manifold Relevance Determination", ICML 2012
- A. C. Damianou and N. D. Lawrence (2013), "Deep Gaussian processes", AISTATS 2013
- J. Hensman (2013), "Gaussian processes for Big Data", UAI 2013

BACKUP SLIDES

MRD weights



Dimensionality reduction: Linear vs non-linear

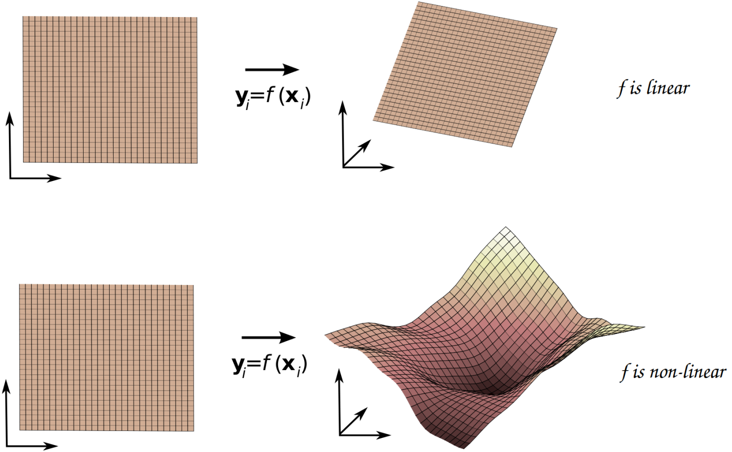


Image from: "Dimensionality Reduction the Probabilistic Way", N. Lawrence, ICML tutorial 2008