Gaussian Processes for Natural Language Processing

http://goo.gl/18heUk

Trevor Cohn[1] Daniel Preoţiuc-Pietro[2] Neil Lawrence[2]

Computing and Information Systems[1] Department of Computer Science[2]

THE UNIVERSITY OF MELBOURNE

The University Of Sheffield.

ACL 2014 Tutorial, 22 June 2014

(Special thanks also to Daniel Beck)

Gaussian Processes

Brings together several key ideas in one framework

- Bayesian
- kernelised
- non-parametric
- non-linear
- modelling uncertainty

Elegant and powerful framework, with growing popularity in machine learning and application domains.

Gaussian Processes

State of the art for regression

- exact posterior inference
- supports very complex non-linear functions
- elegant model selection

Now mature enough for use in NLP

- support for classification, ranking, etc
- fancy kernels, e.g., text
- sparse approximations for large scale inference

Several great toolkits:

`https://github.com/SheffieldML/GPy`

`http://www.gaussianprocess.org/gpml`

Tutorial Scope

Covers

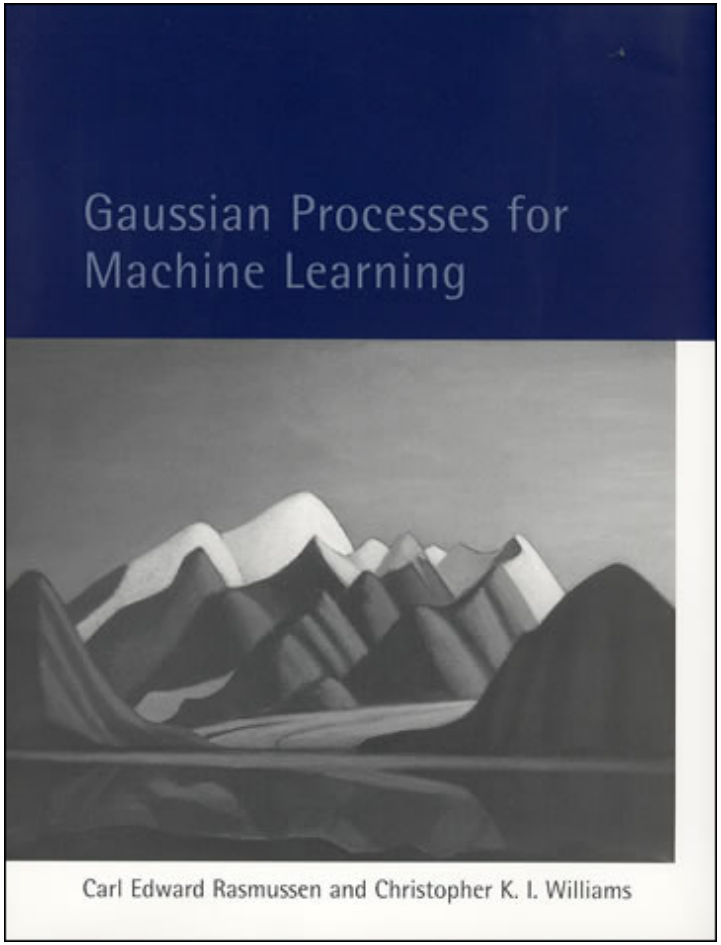
1. GP fundamentals (1 hour)
   - focus on regression
   - weight space vs. function space view
   - squared exponential kernel
2. NLP applications (1 hour 15)
   - sparse GPs
   - multi-output GPs
   - kernels
   - model selection
3. Further topics (30 mins)
   - classification and other likelihoods
   - unsupervised inference
   - scaling to big data

See also materials from the GP Summer/Winter Schools

`http://ml.dcs.shef.ac.uk/gpss/gpws14/`

Outline

Book



Rasmussen and Williams (2006)

Outline

Outline

The Gaussian Density

Covariance from Basis Functions

Basis Function Representations

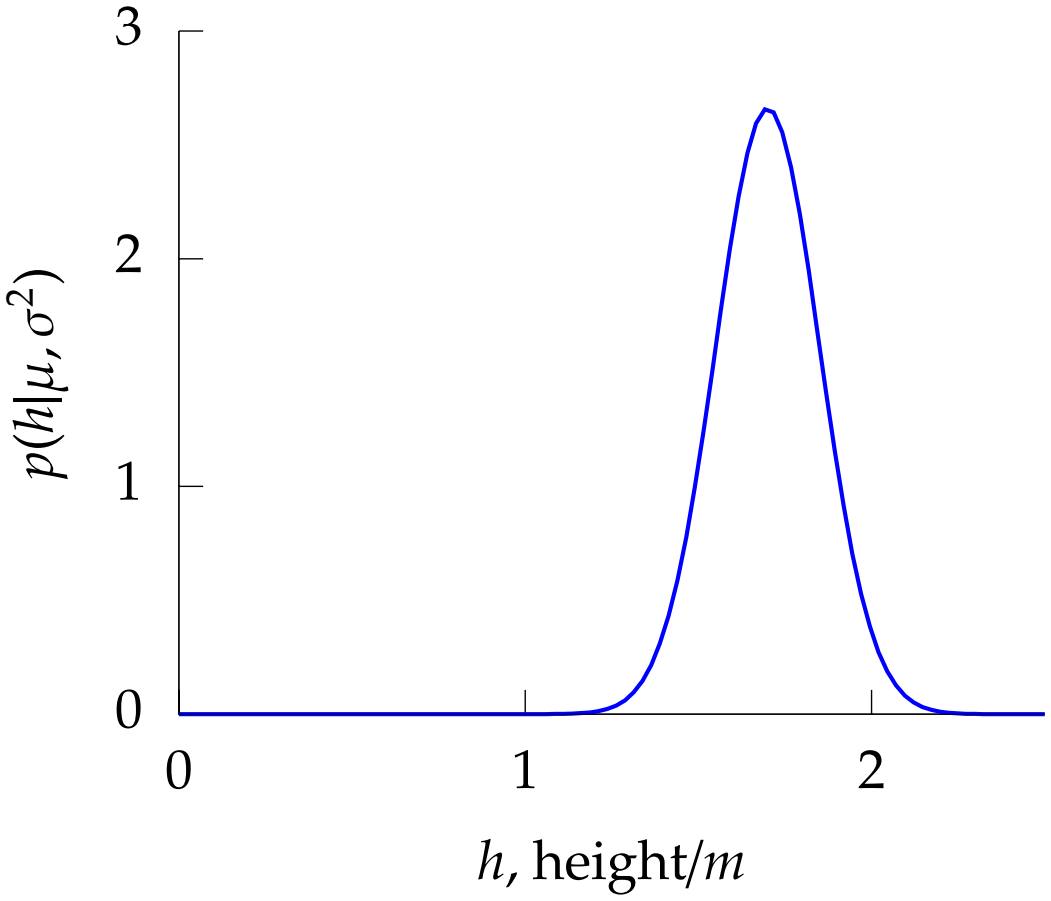
The Gaussian Density

- Perhaps the most common probability density.

$$
\begin{aligned}
p(y|\mu, \sigma^2) &= \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(y-\mu)^2}{2\sigma^2}\right) \\
&\triangleq \mathcal{N}\left(y|\mu, \sigma^2\right)
\end{aligned}
$$

- The Gaussian density.

Gaussian Density



The Gaussian PDF with $\mu = 1.7$ and variance $\sigma^2 = 0.0225$. Mean shown as red line. It could represent the heights of a population of students.

Gaussian Density

$$\mathcal{N}\left(y|\mu,\sigma^2\right) = \frac{1}{\sqrt{2\pi\sigma^2}}\exp\left(-\frac{(y-\mu)^2}{2\sigma^2}\right)$$

σ^2 is the variance of the density and μ is the mean.

Two Important Gaussian Properties

Sum of Gaussians

- Sum of Gaussian variables is also Gaussian.

$$y_i \sim \mathcal{N}\left(\mu_i, \sigma_i^2\right)$$

Two Important Gaussian Properties

Sum of Gaussians

- Sum of Gaussian variables is also Gaussian.

$$y_i \sim \mathcal{N}\left(\mu_i, \sigma_i^2\right)$$

And the sum is distributed as

$$\sum_{i=1}^{n} y_i \sim \mathcal{N}\left(\sum_{i=1}^{n} \mu_i, \sum_{i=1}^{n} \sigma_i^2\right)$$

Two Important Gaussian Properties

Sum of Gaussians

- Sum of Gaussian variables is also Gaussian.

$$y_i \sim \mathcal{N}\left(\mu_i, \sigma_i^2\right)$$

And the sum is distributed as

$$\sum_{i=1}^{n} y_i \sim \mathcal{N}\left(\sum_{i=1}^{n} \mu_i, \sum_{i=1}^{n} \sigma_i^2\right)$$

(*Aside*: As sum increases, sum of non-Gaussian, finite variance variables is also Gaussian [central limit theorem].)

Two Important Gaussian Properties

Sum of Gaussians

- Sum of Gaussian variables is also Gaussian.

$$y_i \sim \mathcal{N}\left(\mu_i, \sigma_i^2\right)$$

And the sum is distributed as

$$\sum_{i=1}^{n} y_i \sim \mathcal{N}\left(\sum_{i=1}^{n} \mu_i, \sum_{i=1}^{n} \sigma_i^2\right)$$

(*Aside*: As sum increases, sum of non-Gaussian, finite variance variables is also Gaussian [central limit theorem].)

Two Important Gaussian Properties

Scaling a Gaussian

- Scaling a Gaussian leads to a Gaussian.

Two Important Gaussian Properties

Scaling a Gaussian

- Scaling a Gaussian leads to a Gaussian.

$$y \sim \mathcal{N}\left(\mu, \sigma^2\right)$$

Two Important Gaussian Properties

Scaling a Gaussian

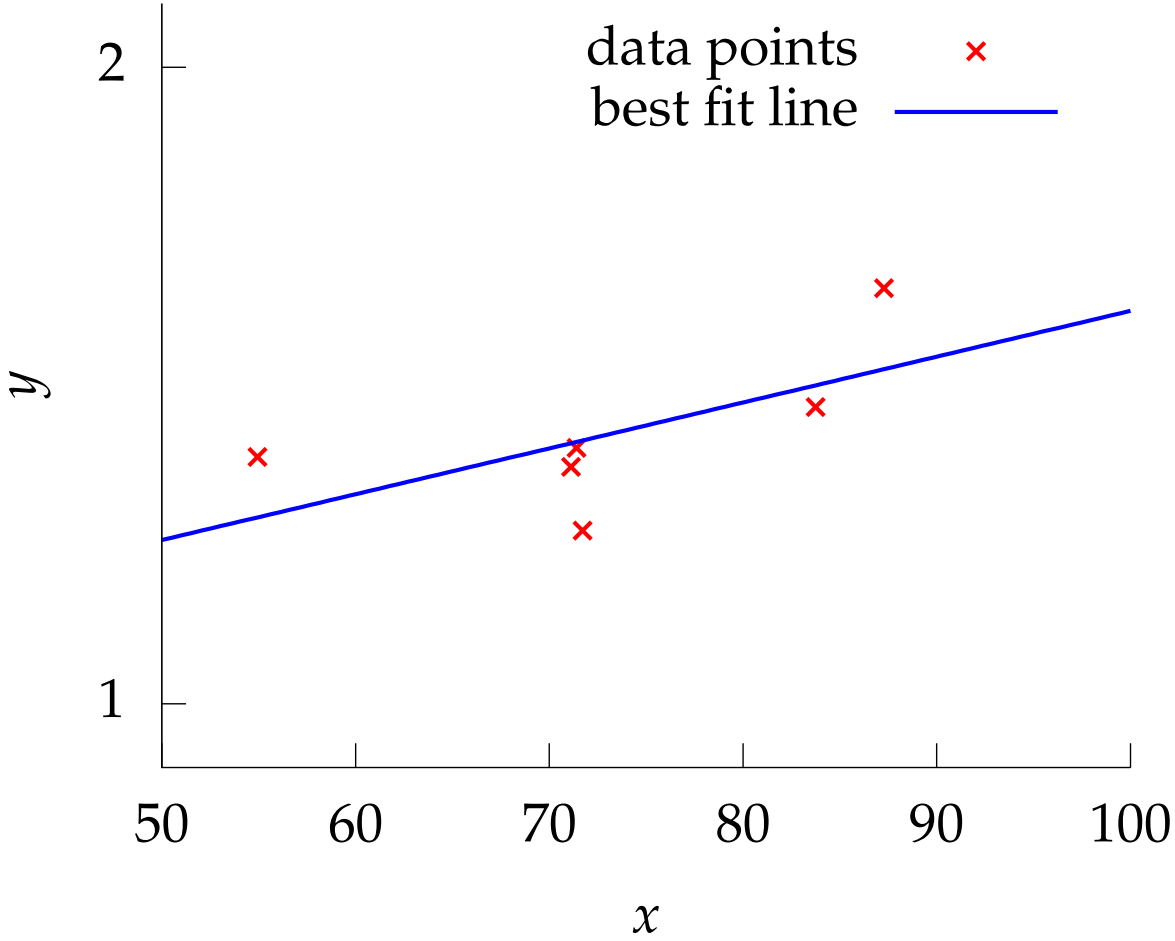
- Scaling a Gaussian leads to a Gaussian.

$$y \sim \mathcal{N}\left(\mu, \sigma^2\right)$$

And the scaled density is distributed as

$$wy \sim \mathcal{N}\left(w\mu, w^2\sigma^2\right)$$

Linear Function

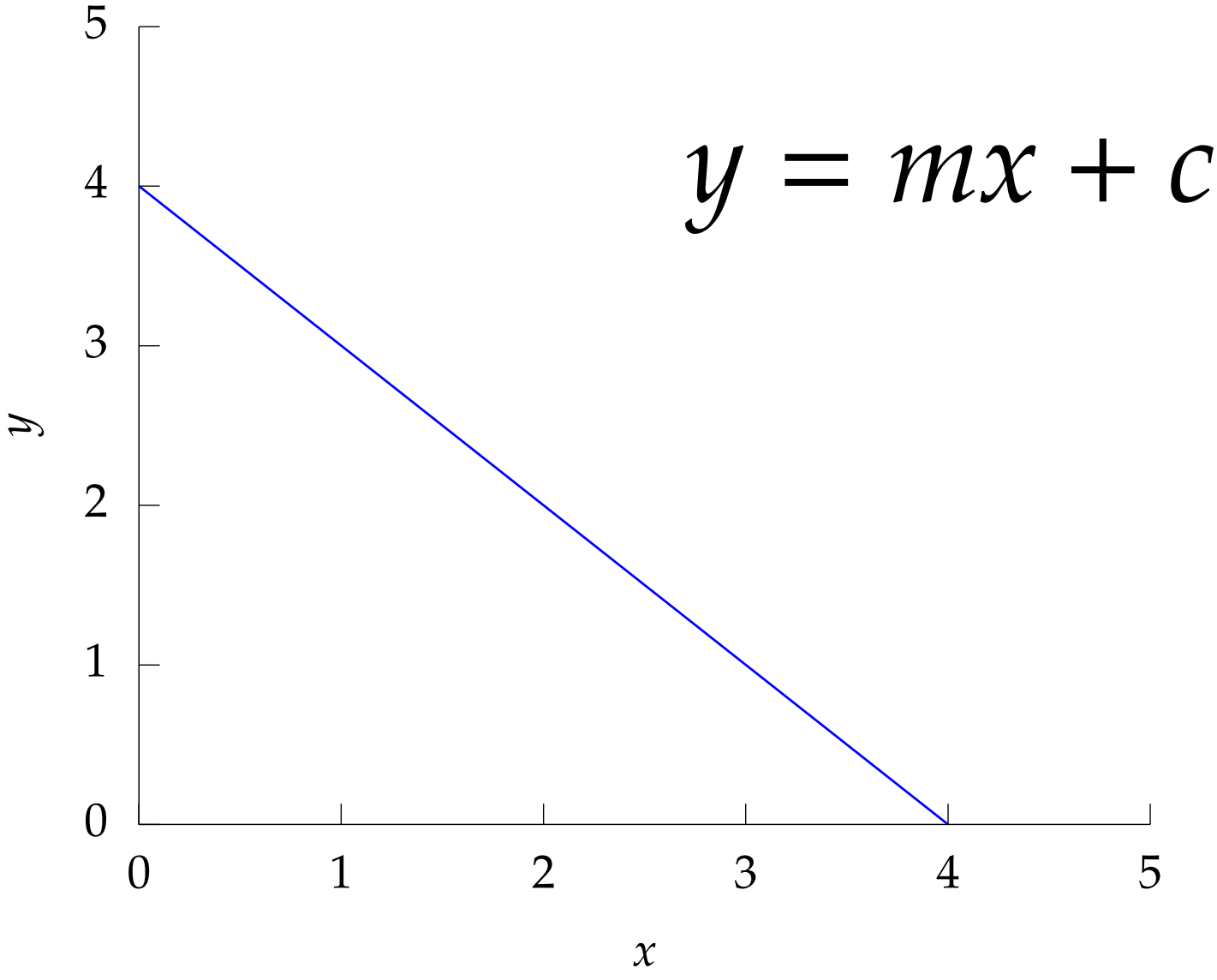


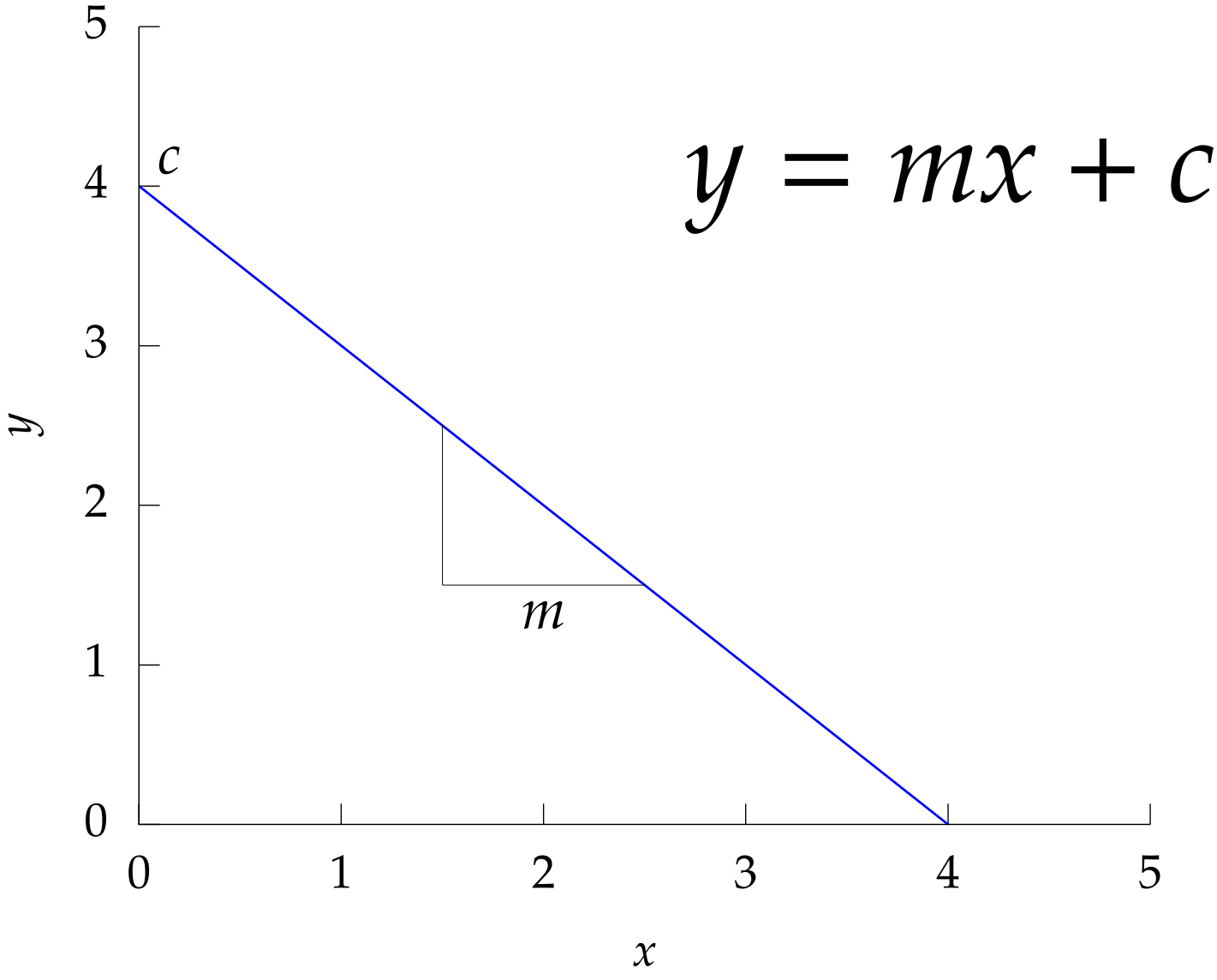
A linear regression between x and y.

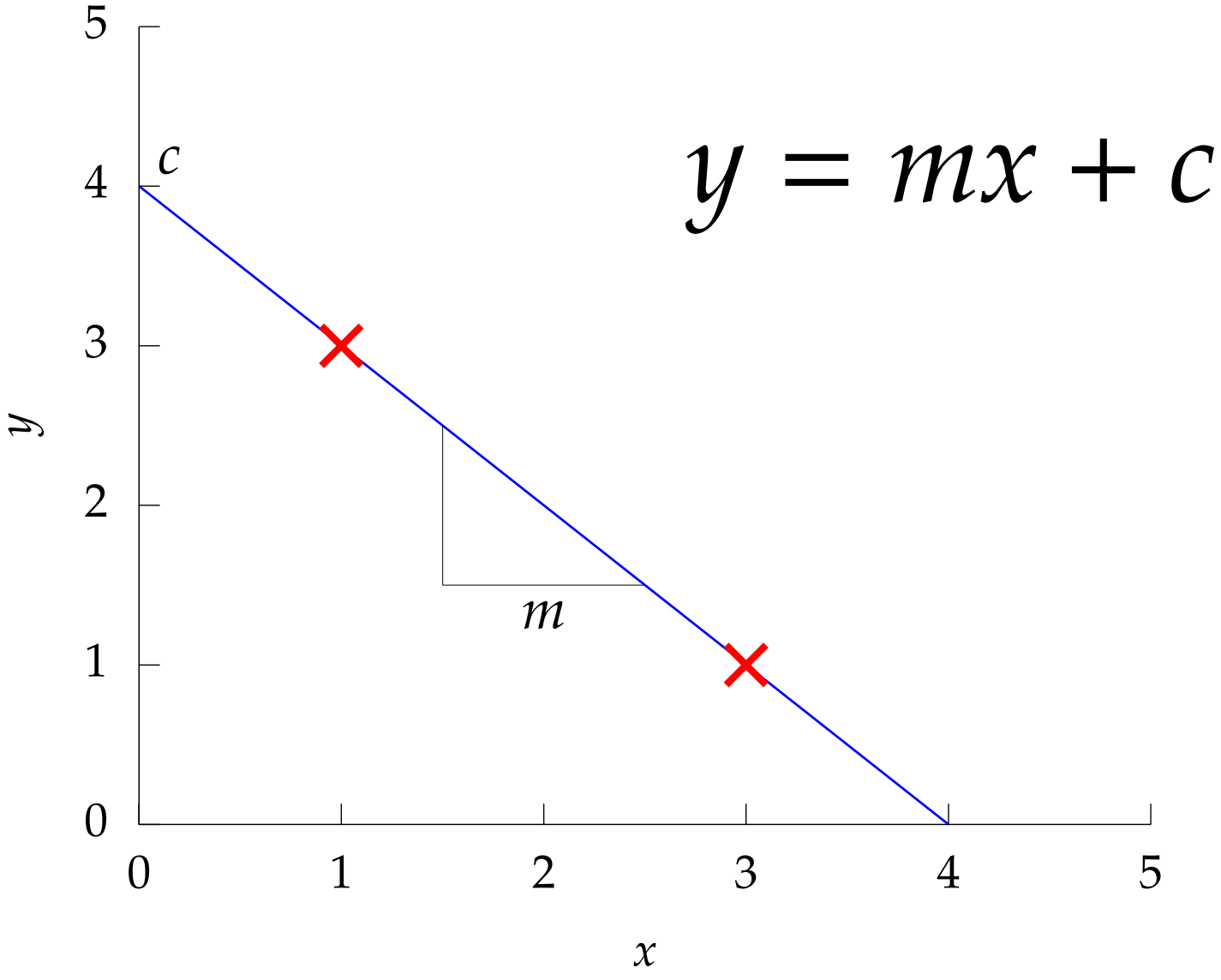
Regression Examples

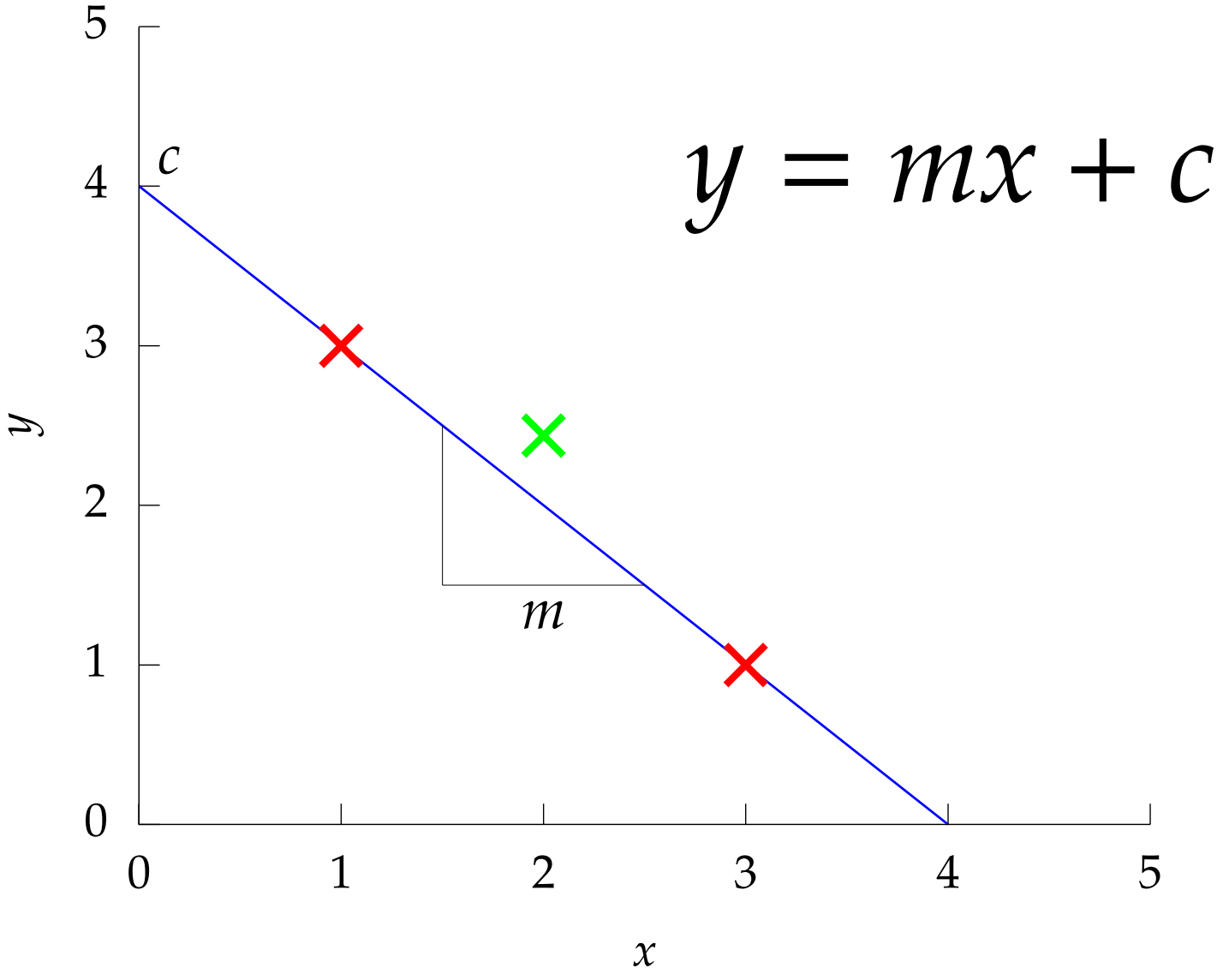
- Predict a real value, y_i given some inputs $\mathbf{x}_i$.
- Predict quality of meat given spectral measurements (Tecator data).
- Radiocarbon dating, the C14 calibration curve: predict age given quantity of C14 isotope.
- Predict quality of different Go or Backgammon moves given expert rated training data.

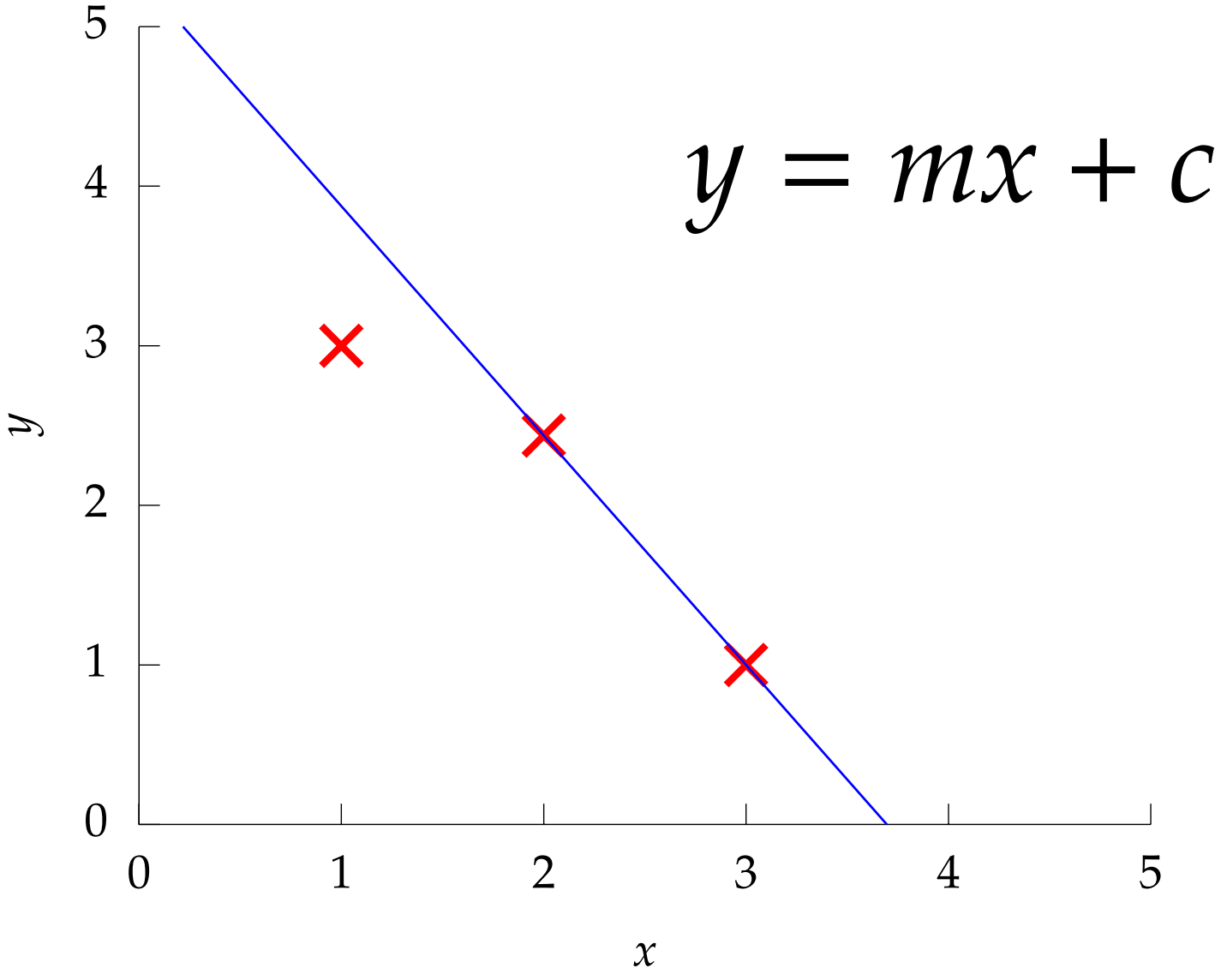
$$y = mx + c$$

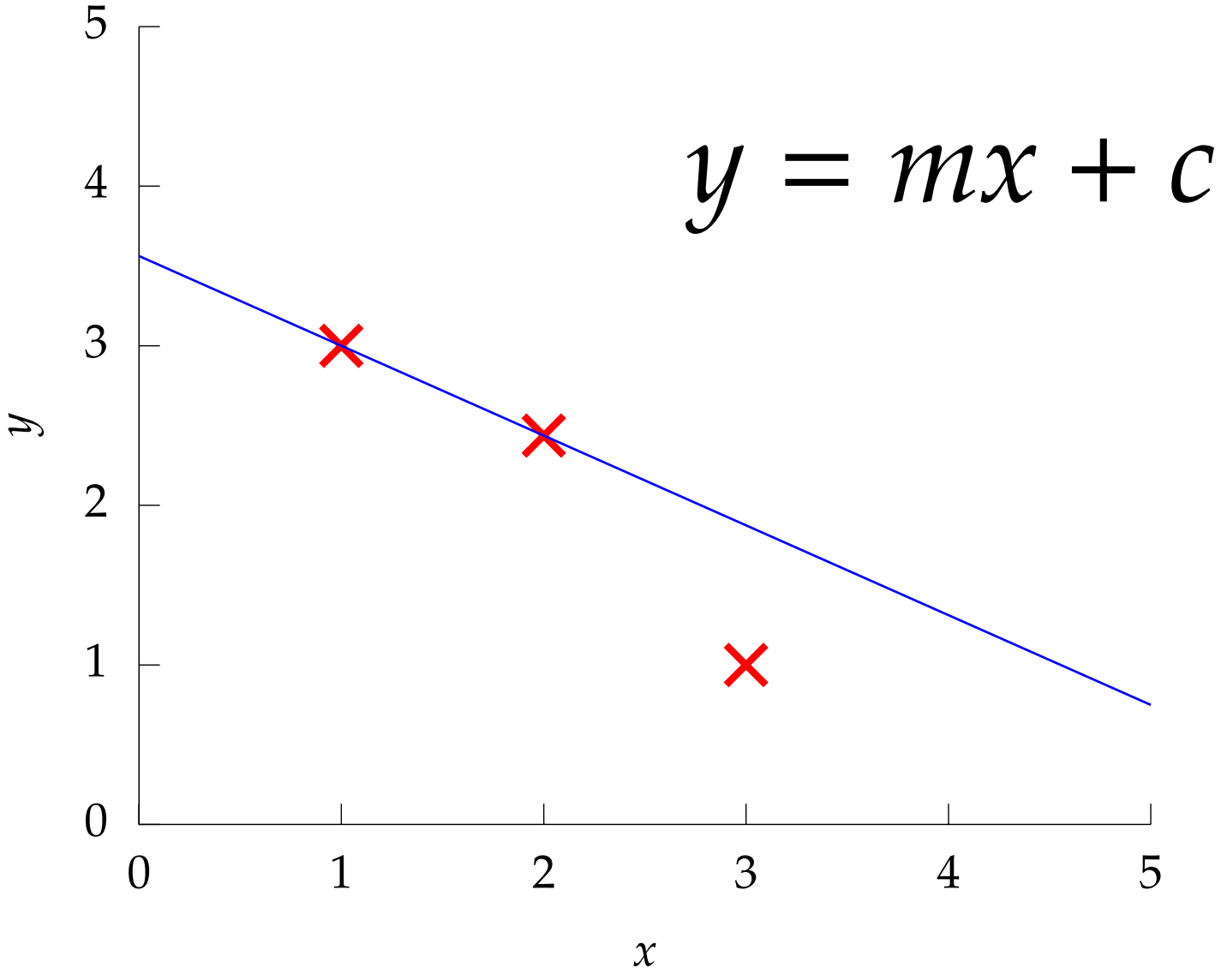

$y = mx + c$
5
4
3
2
1
0
0
1
2
3
4
5
x
y

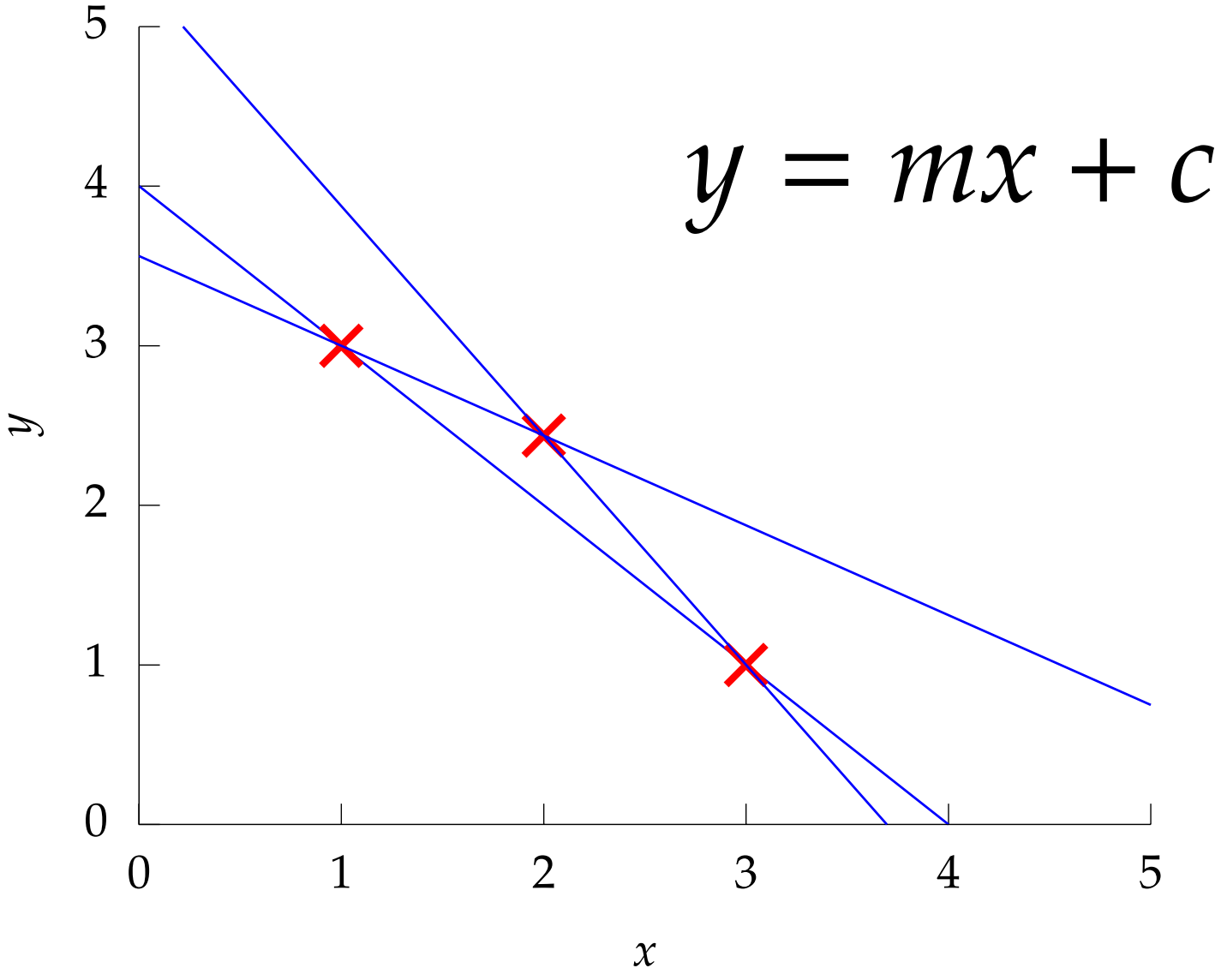

$y = mx + c$
c
m
y
x
0
1
2
3
4
5


$y = mx + c$
c
m
y
x
0
1
2
3
4
5


$y = mx + c$
c
m
y
x
0
1
2
3
4
5


$y = mx + c$
5
4
3
2
1
0
0
1
2
3
4
5
x
y


$y = mx + c$
5
4
3
2
1
0
0
1
2
3
4
5
x
y


$y = mx + c$
0
1
2
3
4
5
x
y

point 1: $x = 1, y = 3$

$$3 = m + c$$

point 2: $x = 3, y = 1$

$$1 = 3m + c$$

point 3: $x = 2, y = 2.5$

$$2.5 = 2m + c$$

riens. L'opinion contraire est une illusion de l'esprit qui, perdant de vue les raisons fugitives du choix de la volonté dans les choses indifférentes, se persuade qu'elle s'est déterminée d'elle-même et sans motifs.

Nous devons donc envisager l'état présent de l'univers, comme l'effet de son état antérieur, et comme la cause de celui qui va suivre. Une intelligence qui, pour un instant donné, connaîtrait toutes les forces dont la nature est animée, et la situation respective des êtres qui la composent, si d'ailleurs elle était assez vaste pour soumettre ces données à l'analyse, embrasserait dans la même formule les mouvemens des plus grands corps de l'univers et ceux du plus léger atome : rien ne serait incertain pour elle, et l'avenir comme le passé, serait présent à ses yeux. L'esprit humain offre, dans la perfection qu'il a su donner à l'Astronomie, une faible esquisse de cette intelligence. Ses découvertes en Mécanique et en Géométrie, jointes à celle de la pesanteur universelle, l'ont mis à portée de comprendre dans les mêmes expressions analytiques, les états passés et futurs du système du monde. En appliquant la même méthode à quelques autres objets de ses connaissances, il est parvenu à ramener à des lois générales, les phénomènes observés, et à prévoir ceux que des circonstances données doivent faire éclore. Tous ces efforts dans la recherche de la vérité, tendent à le rapprocher sans cesse de l'intelligence que nous venons de concevoir, mais dont il restera toujours infiniment éloigné. Cette tendance propre à l'espèce humaine, est ce qui la rend supérieure aux animaux ; et ses progrès en ce genre, distinguent les nations et les siècles, et font leur véritable gloire.

Rappelons-nous qu'autrefois, et à une époque qui

other, we say that its choice is an effect without a cause. It is then, says Leibnitz, the blind chance of the Epicureans. The contrary opinion is an illusion of the mind, which, losing sight of the evasive reasons of the choice of the will in indifferent things, believes that choice is determined of itself and without motives.

We ought then to regard the present state of the universe as the effect of its anterior state and as the cause of the one which is to follow. Given for one instant an intelligence which could comprehend all the forces by which nature is animated and the respective situation of the beings who compose it—an intelligence sufficiently vast to submit these data to analysis—it would embrace in the same formula the movements of the greatest bodies of the universe and those of the lightest atom; for it, nothing would be uncertain and the future, as the past, would be present to its eyes. The human mind offers, in the perfection which it has been able to give to astronomy, a feeble idea of this intelligence. Its discoveries in mechanics and geometry, added to that of universal gravity, have enabled it to comprehend in the same analytical expressions the past and future states of the system of the world. Applying the same method to some other objects of its knowledge, it has succeeded in referring to general laws observed phenomena and in foreseeing those which given circumstances ought to produce. All these efforts in the search for truth tend to lead it back continually to the vast intelligence which we have just mentioned, but from which it will always remain infinitely removed. This tendency, peculiar to the human race, is that which renders it superior to animals; and their progress

height: "The day will come when, by study pursued through several ages, the things now concealed will appear with evidence; and posterity will be astonished that truths so clear had escaped us." Clairaut then undertook to submit to analysis the perturbations which the comet had experienced by the action of the two great planets, Jupiter and Saturn; after immense calculations he fixed its next passage at the perihelion toward the beginning of April, 1759, which was actually verified by observation. The regularity which astronomy shows us in the movements of the comets doubtless exists also in all phenomena.

The curve described by a simple molecule of air or vapor is regulated in a manner just as certain as the planetary orbits; the only difference between them is that which comes from our ignorance.

Probability is relative, in part to this ignorance, in part to our knowledge. We know that of three or a greater number of events a single one ought to occur; but nothing induces us to believe that one of them will occur rather than the others. In this state of indecision it is impossible for us to announce their occurrence with certainty. It is, however, probable that one of these events, chosen at will, will not occur because we see several cases equally possible which exclude its occurrence, while only a single one favors it.

The theory of chance consists in reducing all the events of the same kind to a certain number of cases equally possible, that is to say, to such as we may be equally undecided about in regard to their existence, and in determining the number of cases favorable to the event whose probability is sought. The ratio of

point 1: $x = 1, y = 3$

$$3 = m + c + \epsilon_1$$

point 2: $x = 3, y = 1$

$$1 = 3m + c + \epsilon_2$$

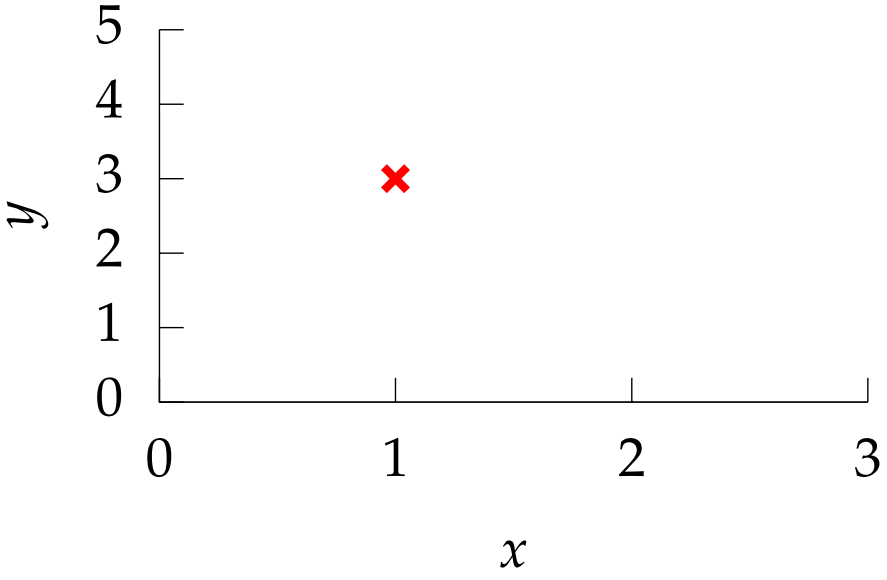
point 3: $x = 2, y = 2.5$

$$2.5 = 2m + c + \epsilon_3$$

Underdetermined System

What about two unknowns and *one* observation?

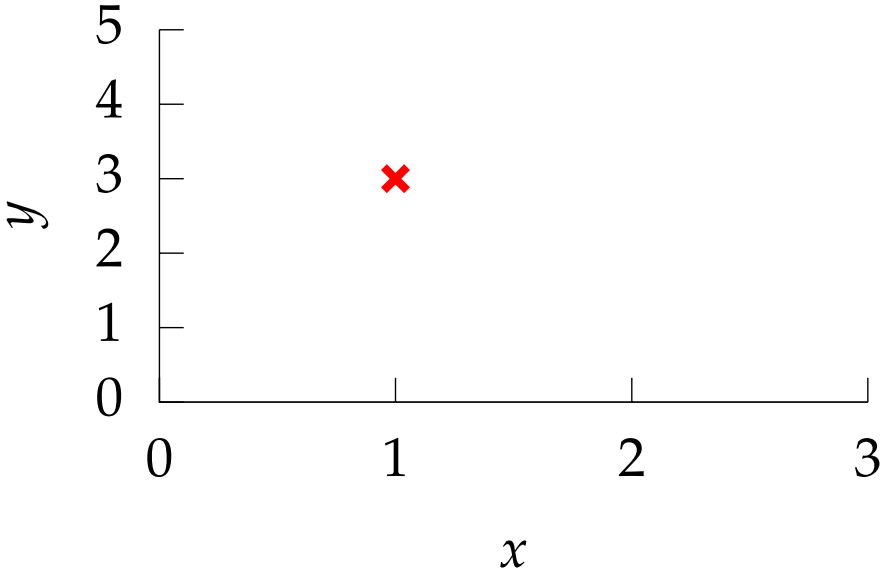
$$y_1 = mx_1 + c$$



Underdetermined System

Can compute m given c.

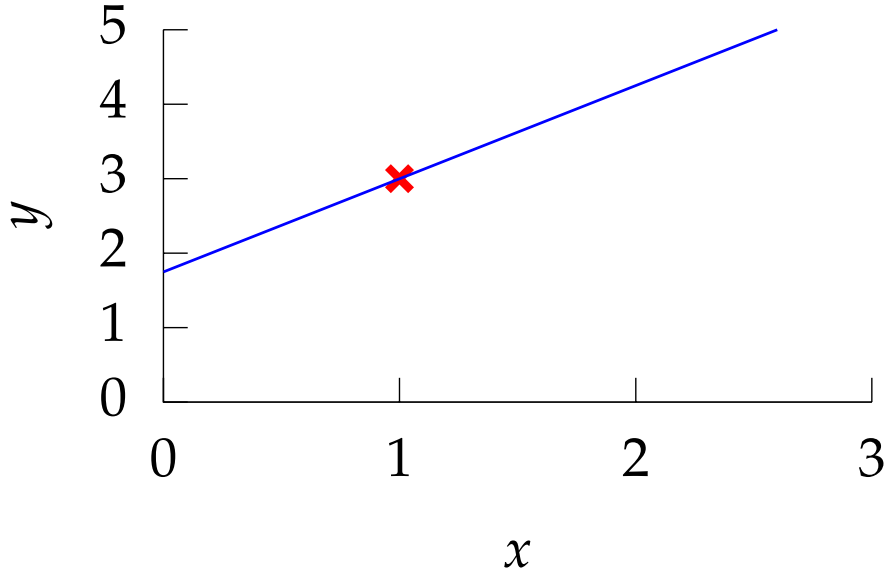
$$m = \frac{y_1 - c}{x}$$



Underdetermined System

Can compute m given c.

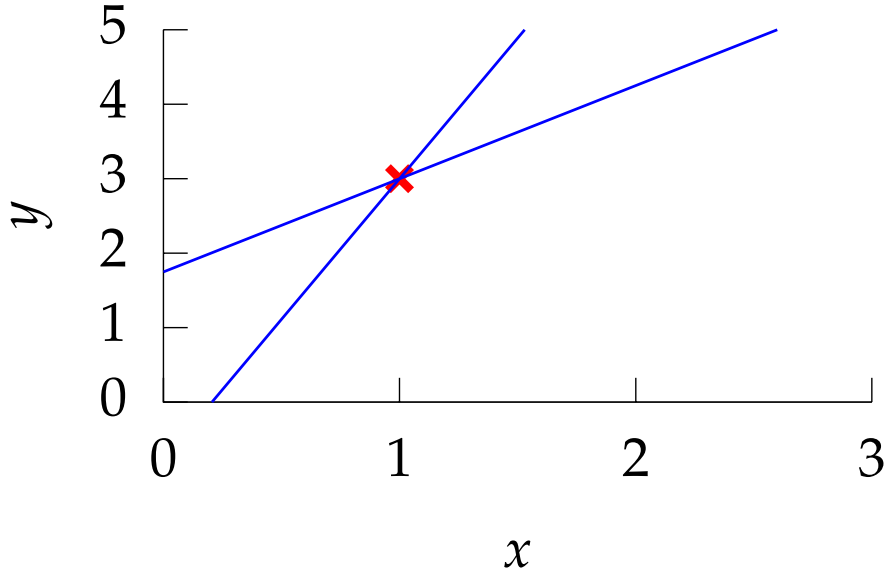
$$c = 1.75 \implies m = 1.25$$



Underdetermined System

Can compute m given c.

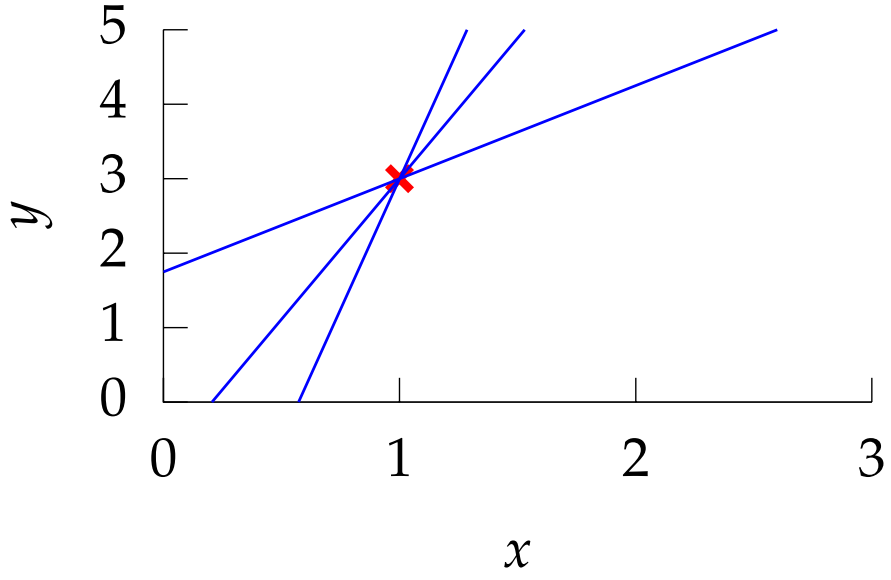
$$c = -0.777 \implies m = 3.78$$



Underdetermined System

Can compute m given c.

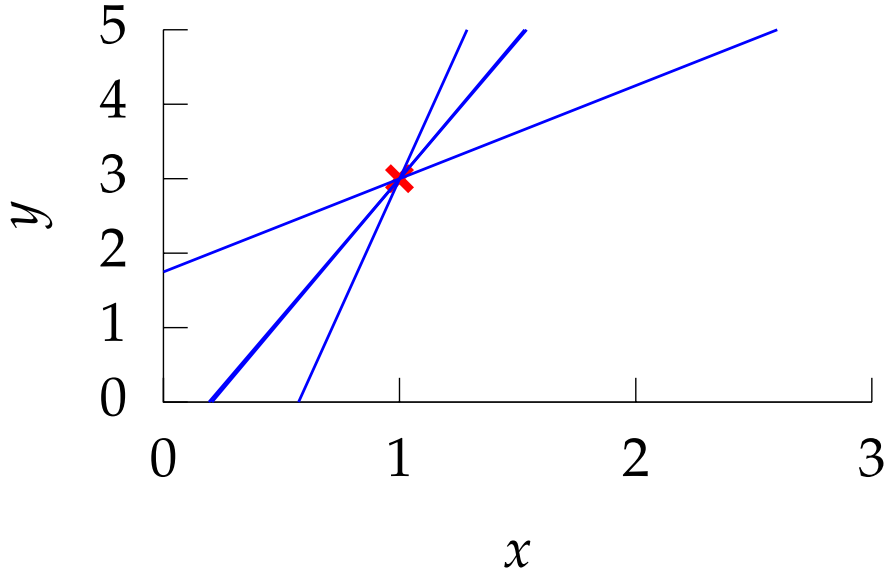
$$c = -4.01 \implies m = 7.01$$



Underdetermined System

Can compute m given c.

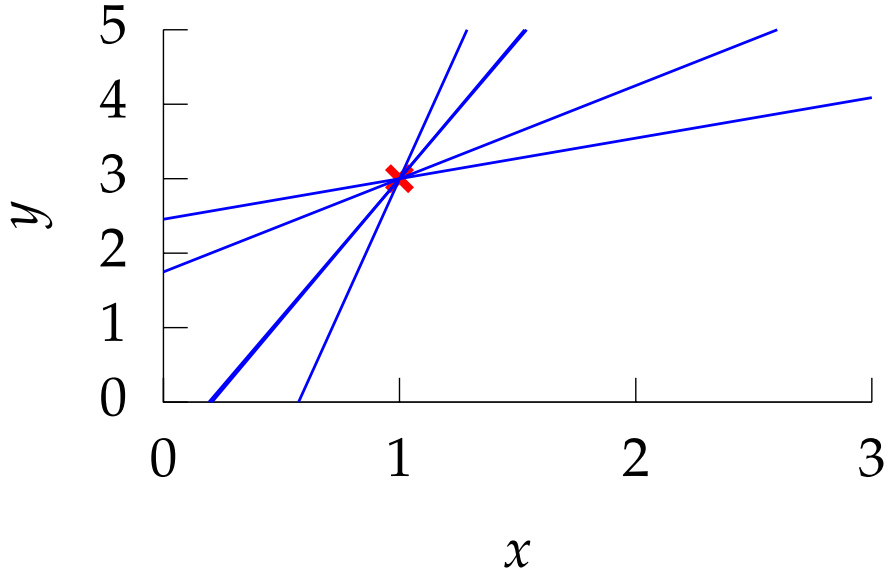
$$c = -0.718 \implies m = 3.72$$



Underdetermined System

Can compute m given c.

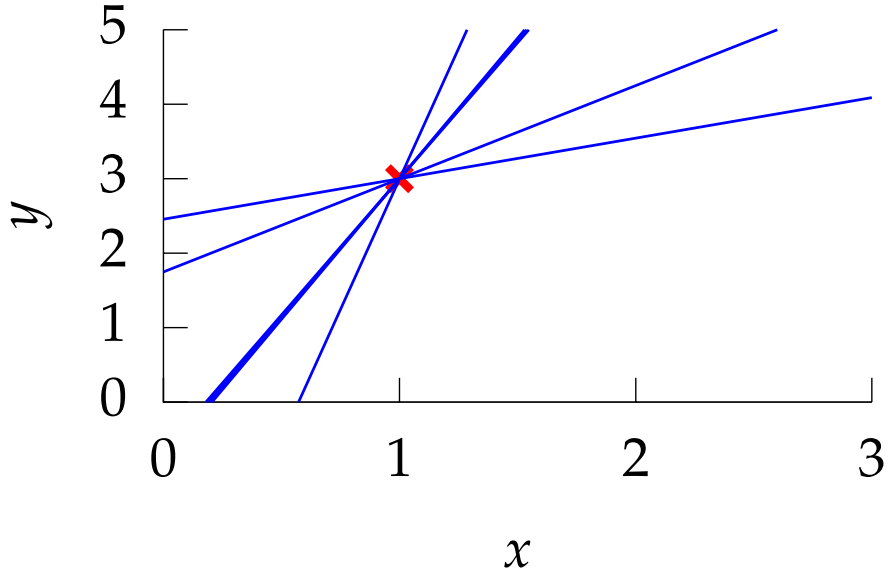
$$c = 2.45 \implies m = 0.545$$



Underdetermined System

Can compute m given c.

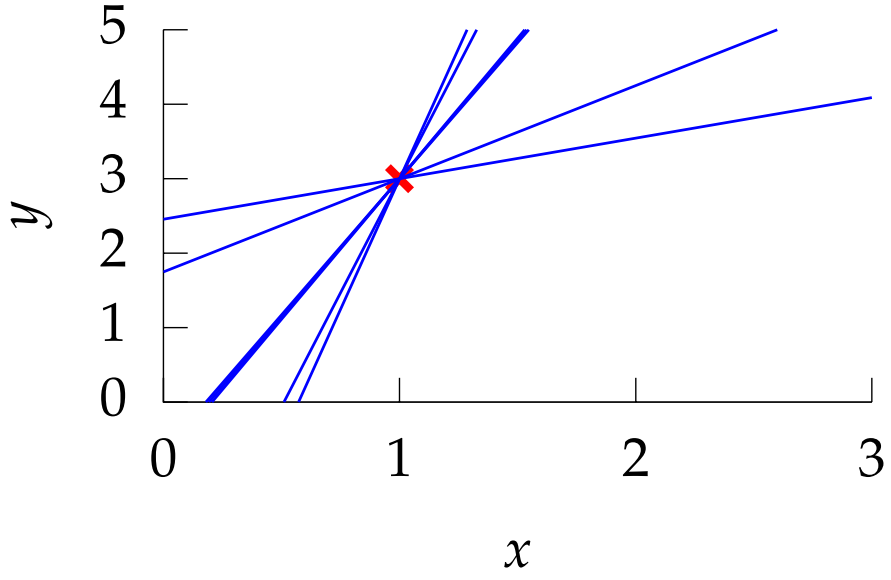
$$c = -0.657 \implies m = 3.66$$



Underdetermined System

Can compute m given c.

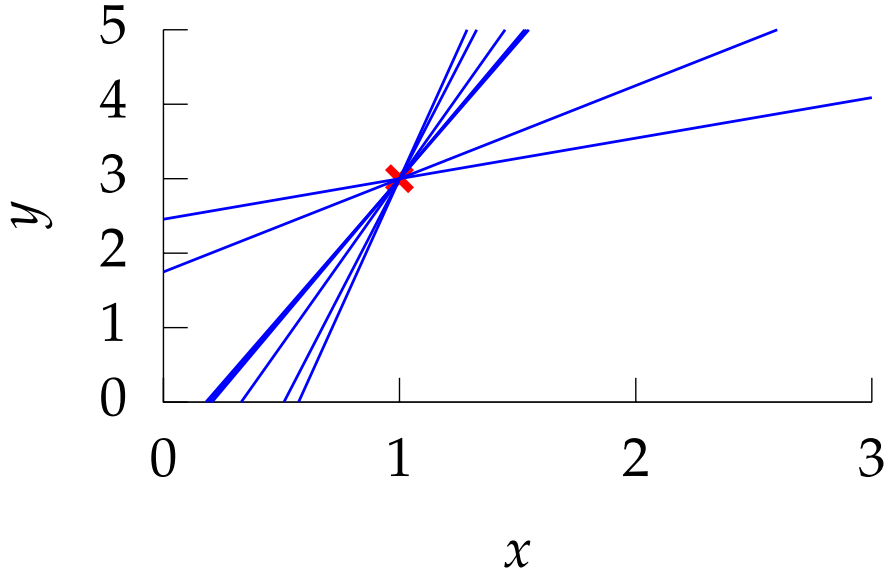
$$c = -3.13 \implies m = 6.13$$



Underdetermined System

Can compute m given c.

$$c = -1.47 \implies m = 4.47$$

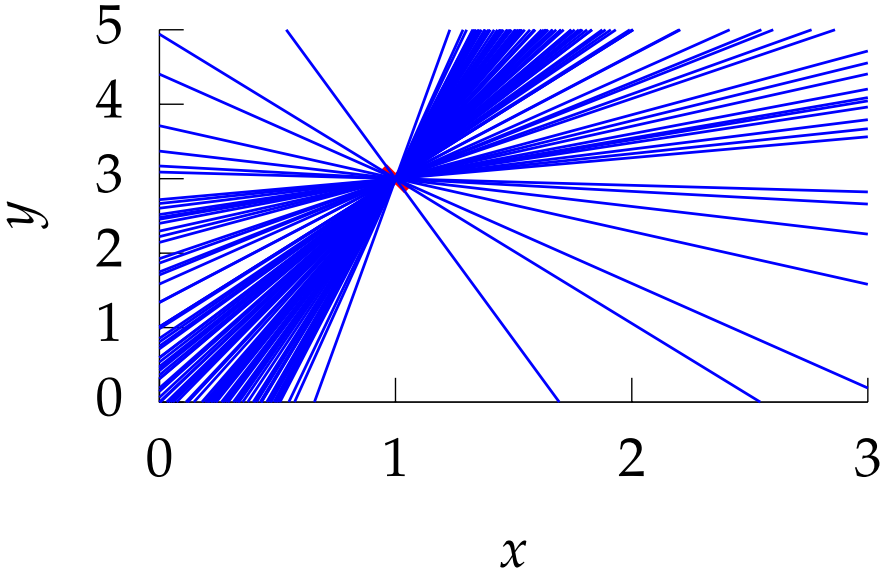


Underdetermined System

Can compute m given c. Assume

$$c \sim \mathcal{N}(0, 4),$$

we find a distribution of solutions.



Probability for Under- and Overdetermined

- To deal with overdetermined introduced probability distribution for 'variable', ϵ_i.
- For underdetermined system introduced probability distribution for 'parameter', c.
- This is known as a Bayesian treatment.

Multivariate Prior Distributions

- For general Bayesian inference need multivariate priors.
- E.g. for multivariate linear regression:

$$y_i = \sum_i w_j x_{i,j} + \epsilon_i$$

  (where we've dropped c for convenience), we need a prior over $\mathbf{w}$.
- This motivates a *multivariate* Gaussian density.
- We will use the multivariate Gaussian to put a prior *directly* on the function (a Gaussian process).

Multivariate Prior Distributions

- For general Bayesian inference need multivariate priors.
- E.g. for multivariate linear regression:

$$y_i = \mathbf{w}^\top \mathbf{x}_{i,:} + \epsilon_i$$

  (where we've dropped c for convenience), we need a prior over $\mathbf{w}$.
- This motivates a *multivariate* Gaussian density.
- We will use the multivariate Gaussian to put a prior *directly* on the function (a Gaussian process).

Prior Distribution

- Bayesian inference requires a prior on the parameters.
- The prior represents your belief *before* you see the data of the likely value of the parameters.
- For linear regression, consider a Gaussian prior on the intercept:

$$c \sim \mathcal{N}(0, \alpha_1)$$

Posterior Distribution

- Posterior distribution is found by combining the prior with the likelihood.
- Posterior distribution is your belief *after* you see the data of the likely value of the parameters.
- The posterior is found through **Bayes' Rule**

$$p(c|y) = \frac{p(y|c)p(c)}{p(y)}$$

Bayes Update

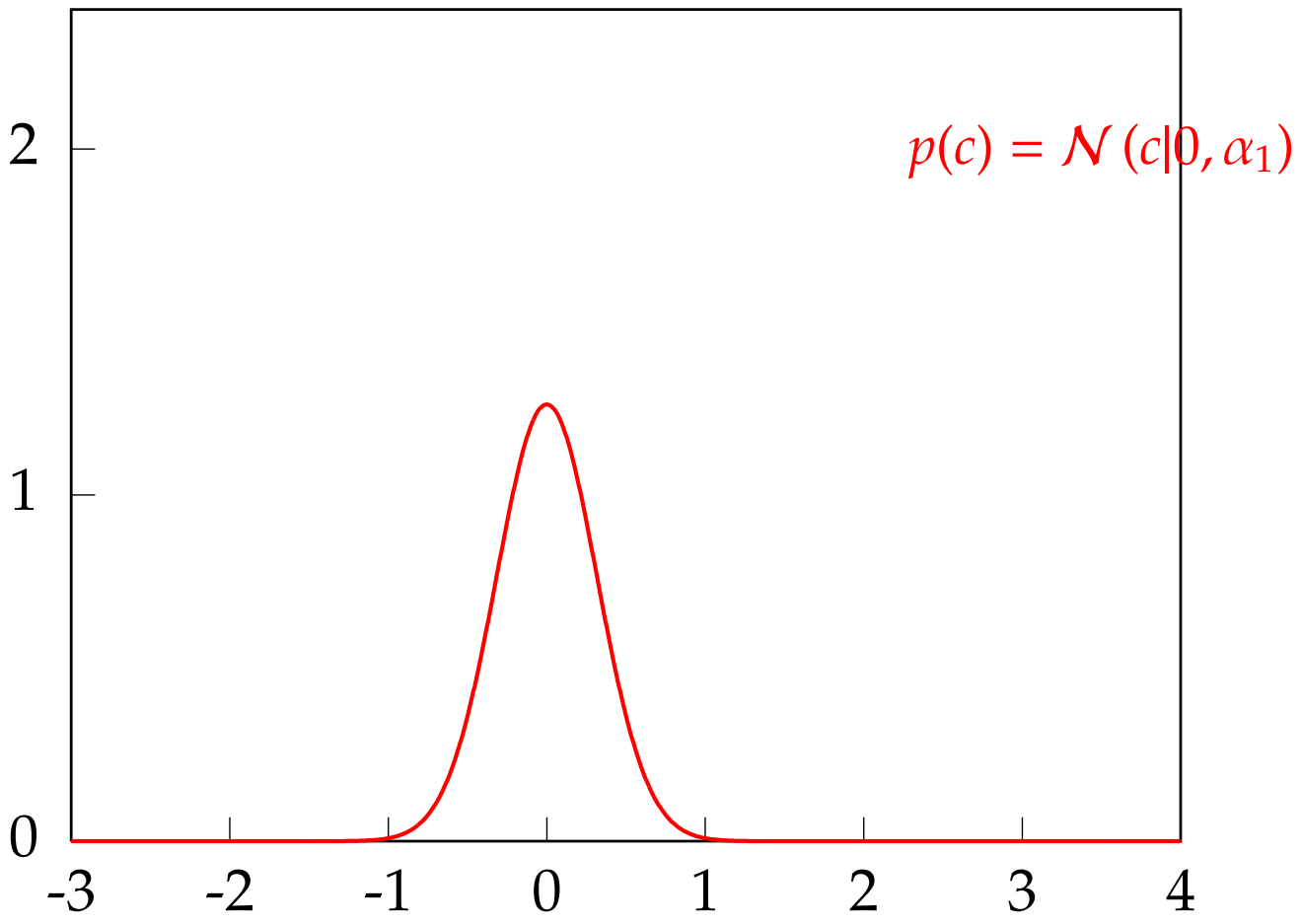


Figure : A Gaussian prior combines with a Gaussian likelihood for a Gaussian posterior.

Bayes Update

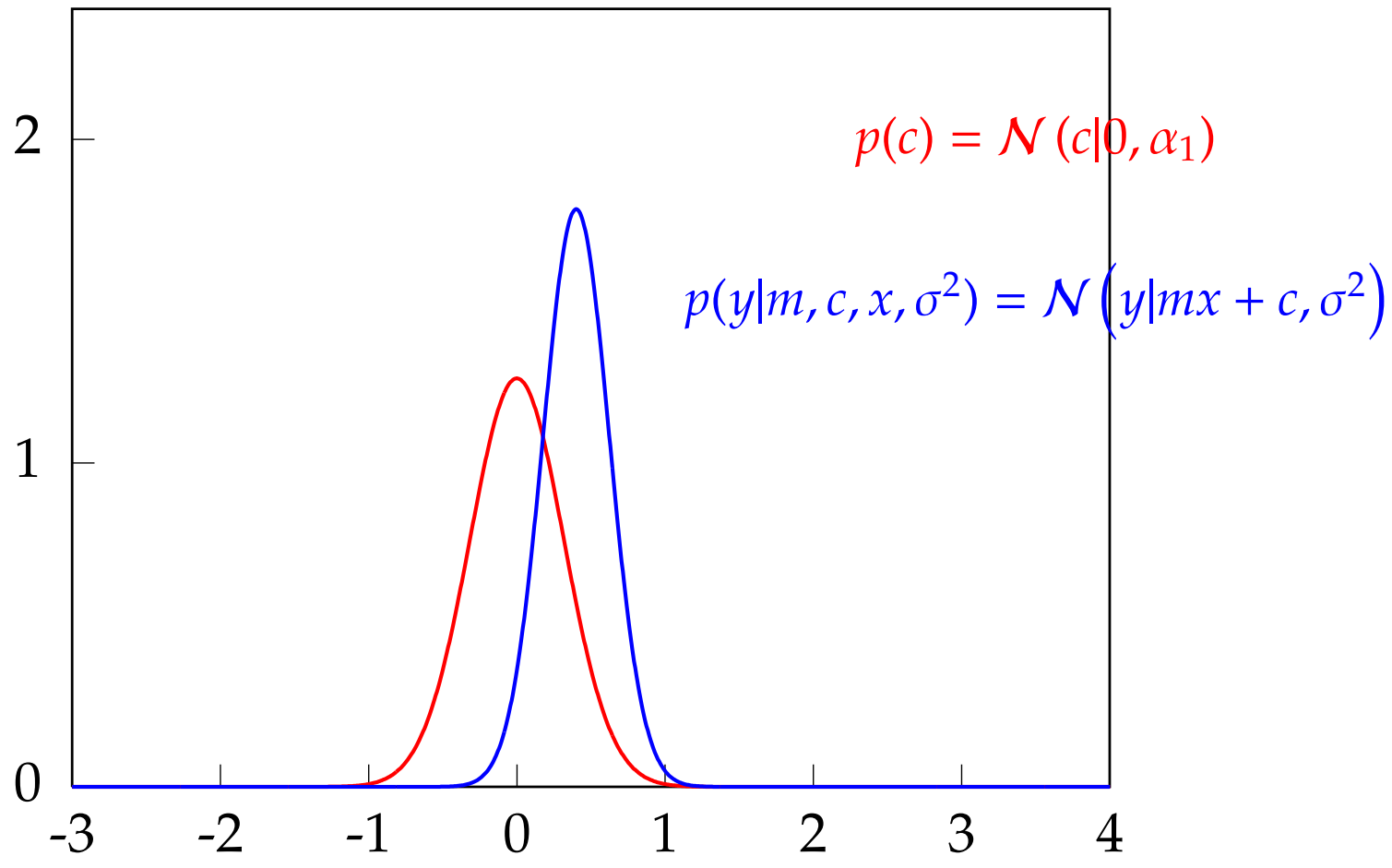


Figure : A Gaussian prior combines with a Gaussian likelihood for a Gaussian posterior.

Bayes Update

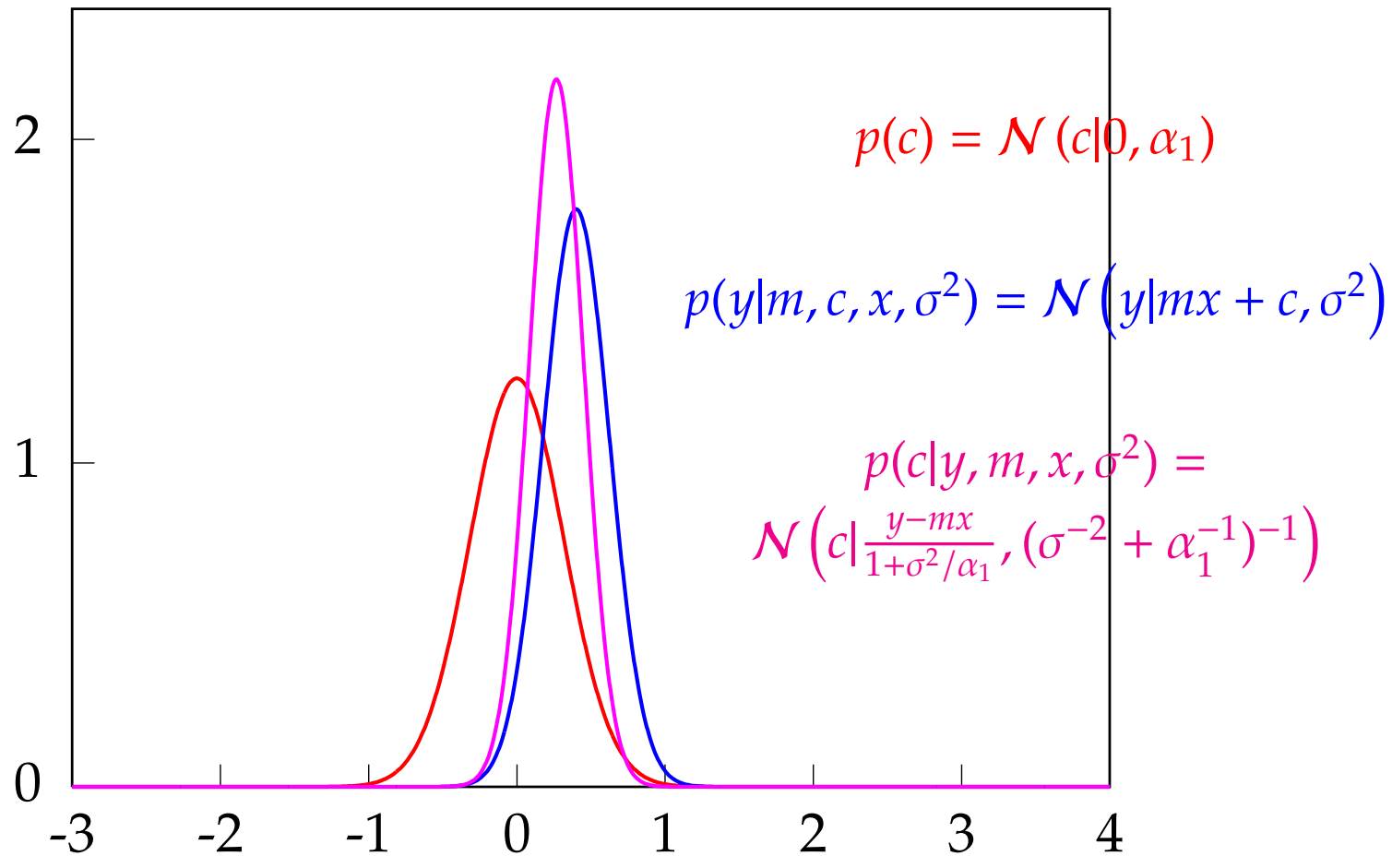


Figure : A Gaussian prior combines with a Gaussian likelihood for a Gaussian posterior.

Stages to Derivation of the Posterior

- Multiply likelihood by prior
  - they are "exponentiated quadratics", the answer is always also an exponentiated quadratic because $\exp(a^2)\exp(b^2) = \exp(a^2 + b^2)$.
- Complete the square to get the resulting density in the form of a Gaussian.
- Recognise the mean and (co)variance of the Gaussian. This is the estimate of the posterior.

Multivariate Regression Likelihood

- Noise corrupted data point

$$y_i = \mathbf{w}^\top \mathbf{x}_{i,:} + \epsilon_i$$

Multivariate Regression Likelihood

- Noise corrupted data point

$$y_i = \mathbf{w}^\top \mathbf{x}_{i,:} + \epsilon_i$$

- Multivariate regression likelihood:

$$p(\mathbf{y}|\mathbf{X}, \mathbf{w}) = \frac{1}{(2\pi\sigma^2)^{n/2}} \exp\left(-\frac{1}{2\sigma^2} \sum_{i=1}^{n} \left(y_i - \mathbf{w}^\top \mathbf{x}_{i,:}\right)^2\right)$$

Multivariate Regression Likelihood

- Noise corrupted data point

$$y_i = \mathbf{w}^\top \mathbf{x}_{i,:} + \epsilon_i$$

- Multivariate regression likelihood:

$$p(\mathbf{y}|\mathbf{X}, \mathbf{w}) = \frac{1}{(2\pi\sigma^2)^{n/2}} \exp\left(-\frac{1}{2\sigma^2} \sum_{i=1}^{n} \left(y_i - \mathbf{w}^\top \mathbf{x}_{i,:}\right)^2\right)$$

- Now use a multivariate Gaussian prior:

$$p(\mathbf{w}) = \frac{1}{(2\pi\alpha)^{\frac{p}{2}}} \exp\left(-\frac{1}{2\alpha}\mathbf{w}^\top \mathbf{w}\right)$$

Two Dimensional Gaussian

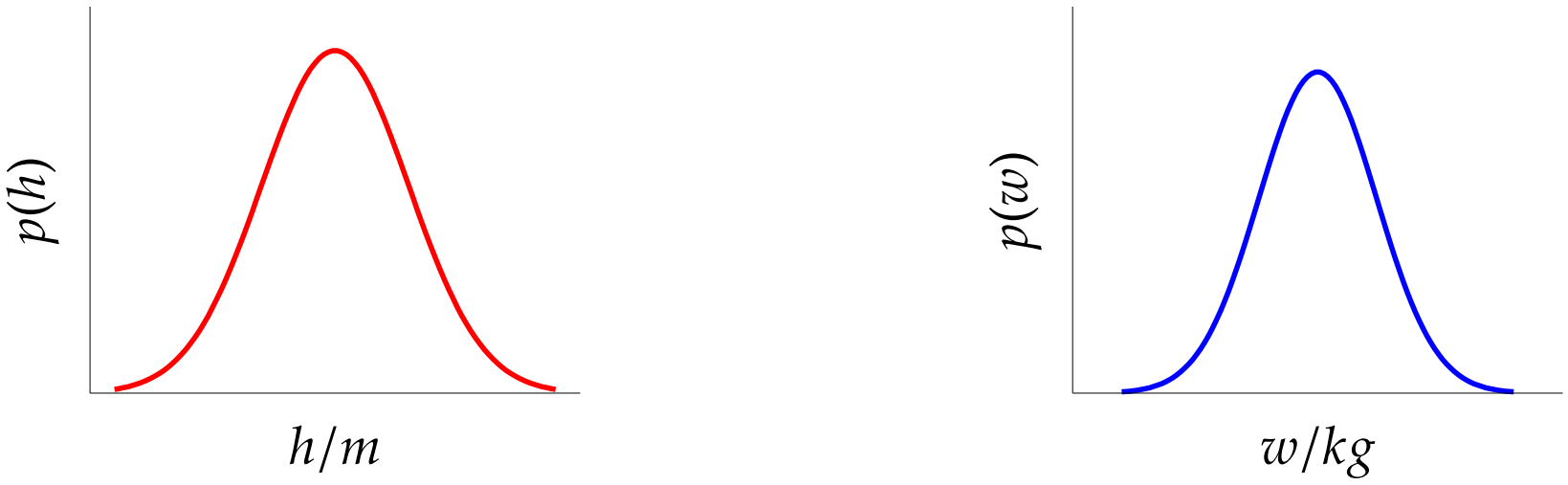
- Consider height, h/m and weight, w/kg.
- Could sample height from a distribution:

$$p(h) \sim \mathcal{N}(1.7, 0.0225)$$

- And similarly weight:

$$p(w) \sim \mathcal{N}(75, 36)$$

Height and Weight Models

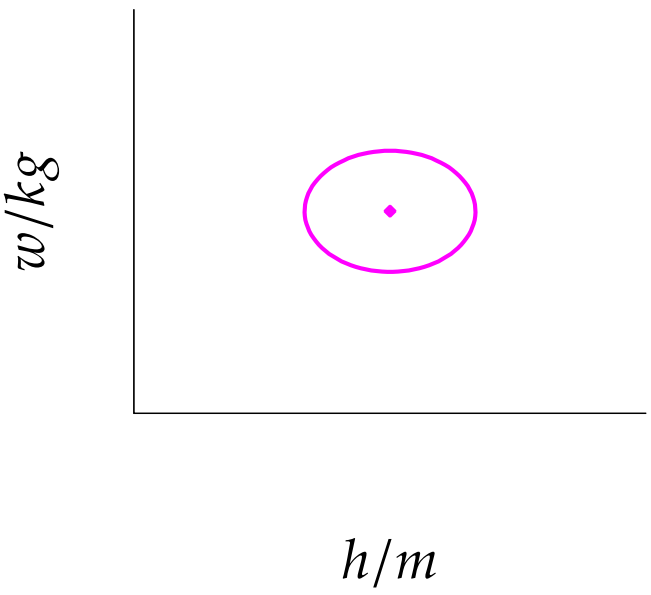


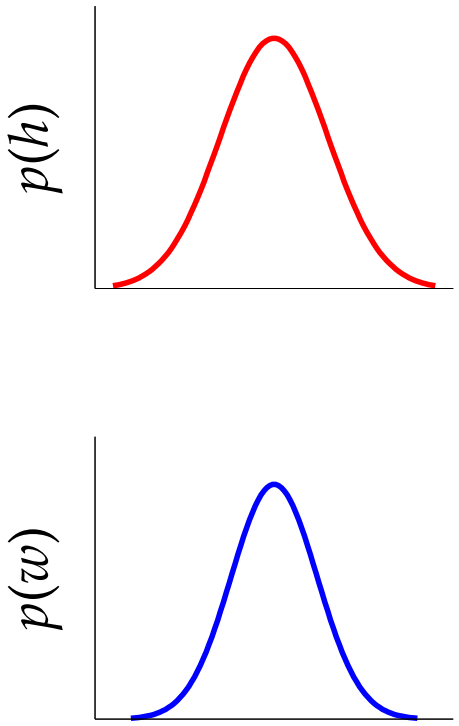
Gaussian distributions for height and weight.

Sampling Two Dimensional Variables

Marginal Distributions

Joint Distribution



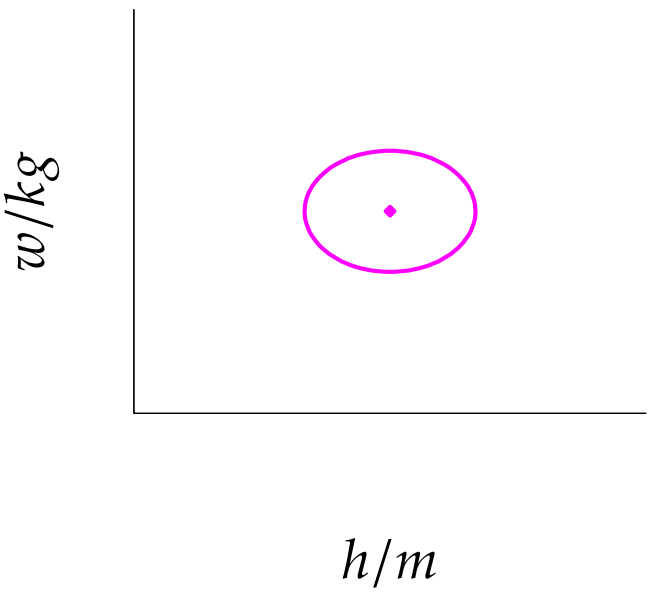


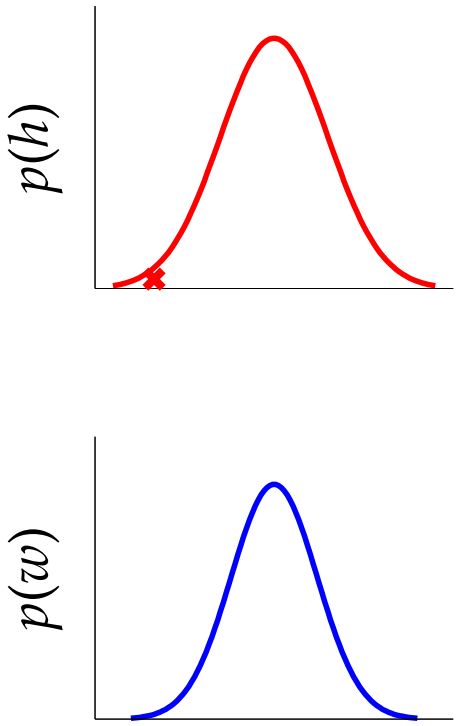
Samples of height and weight

Sampling Two Dimensional Variables

Marginal Distributions

Joint Distribution



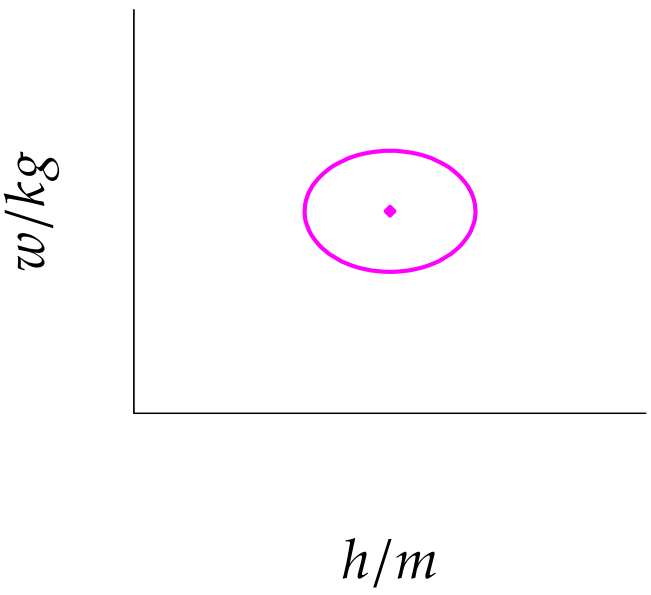


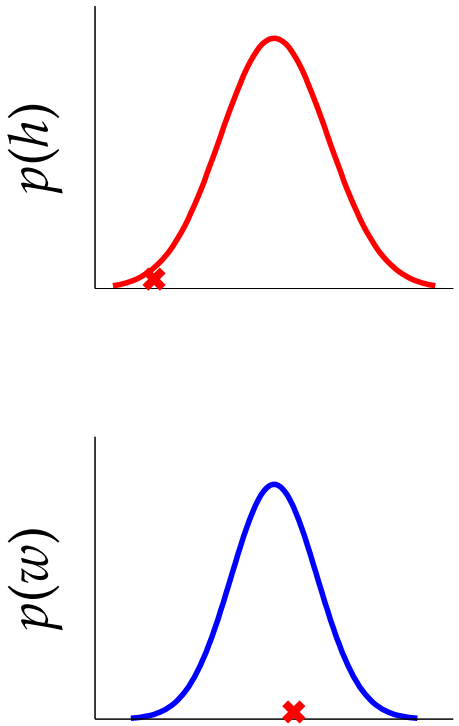
Samples of height and weight

Sampling Two Dimensional Variables

Marginal Distributions

Joint Distribution



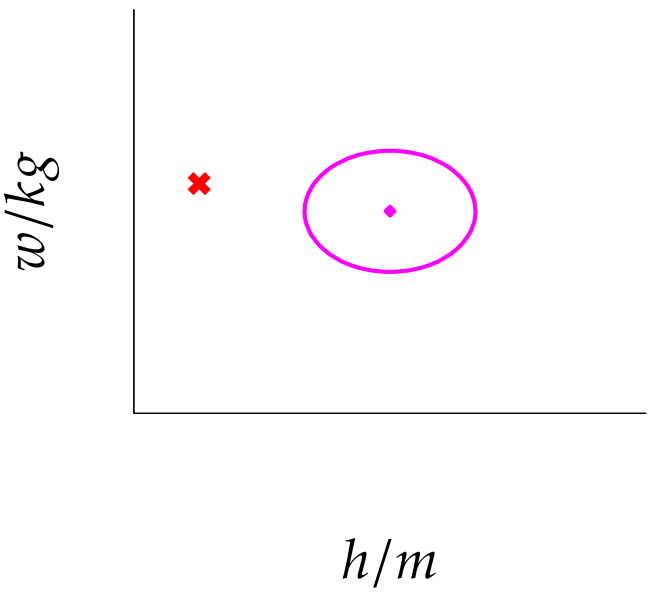


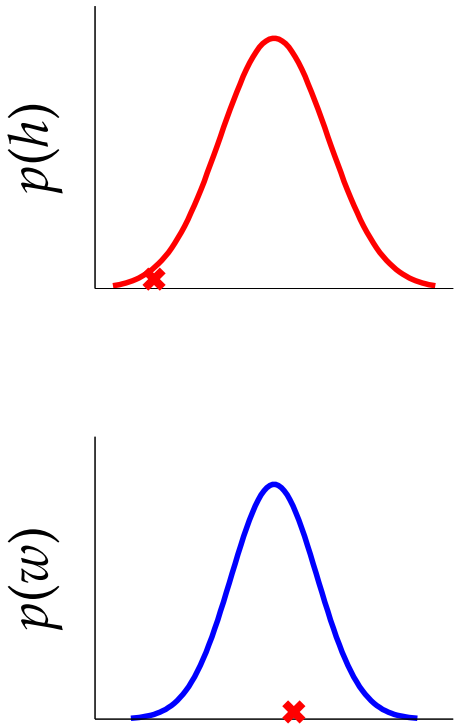
Samples of height and weight

Sampling Two Dimensional Variables

Marginal Distributions

Joint Distribution



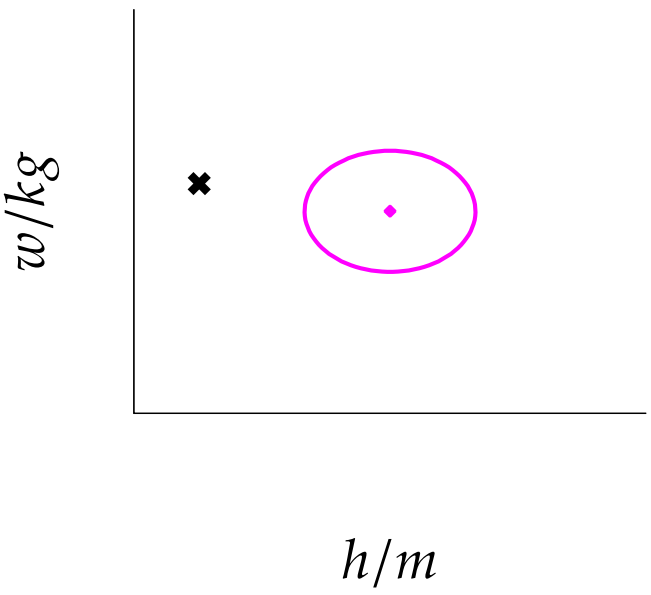


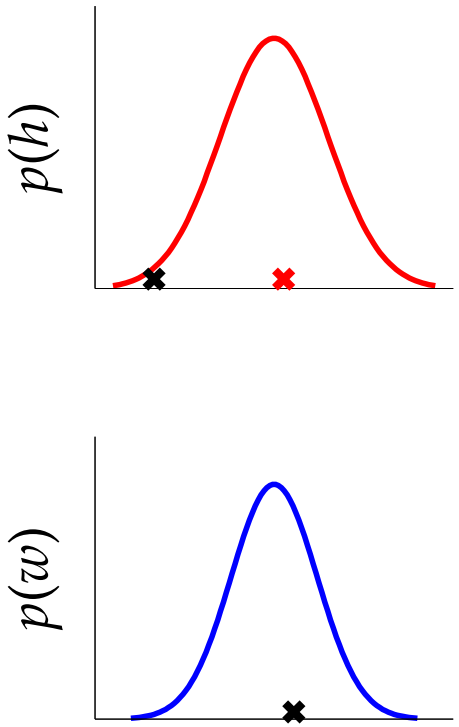
Samples of height and weight

Sampling Two Dimensional Variables

Marginal Distributions

Joint Distribution



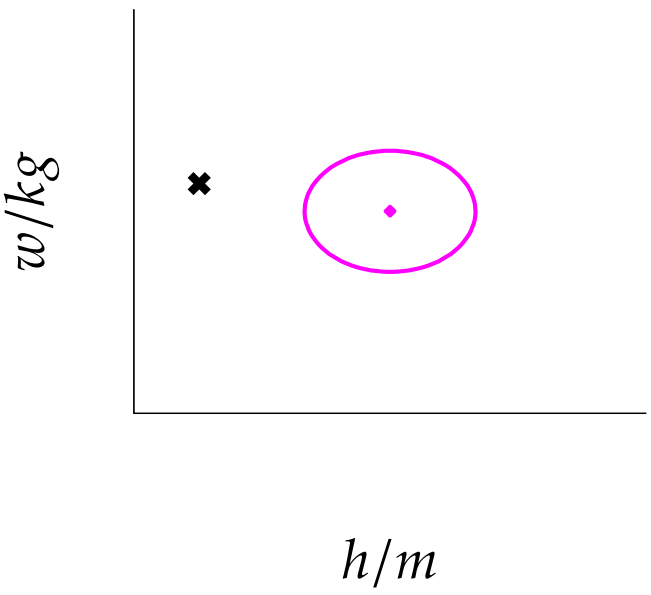


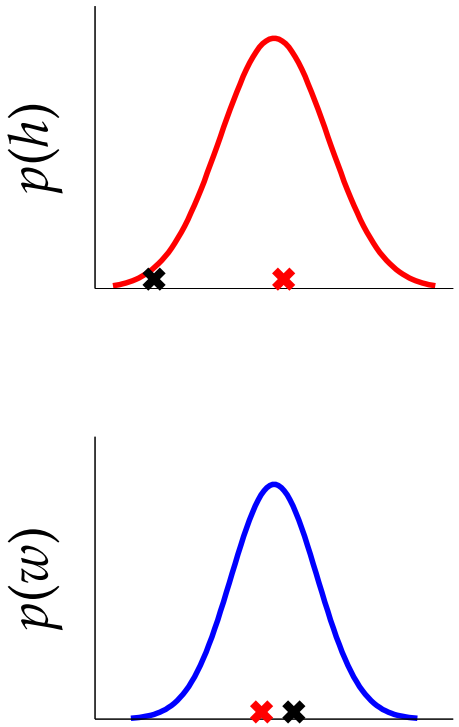
Samples of height and weight

Sampling Two Dimensional Variables

Marginal Distributions

Joint Distribution



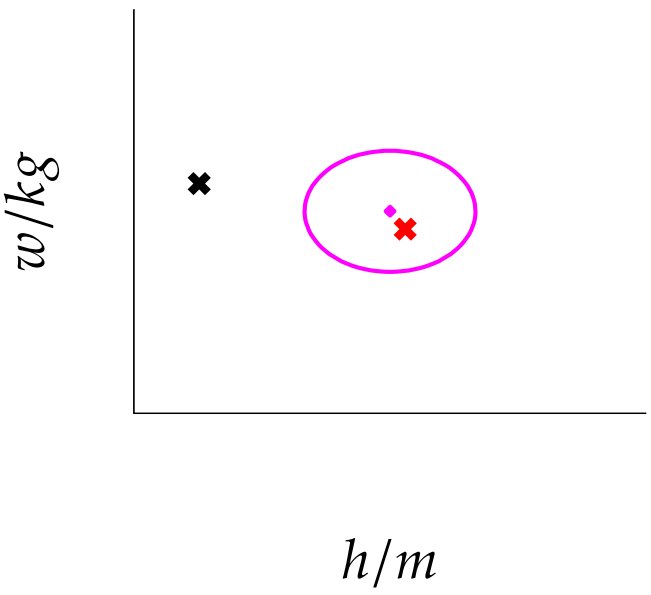


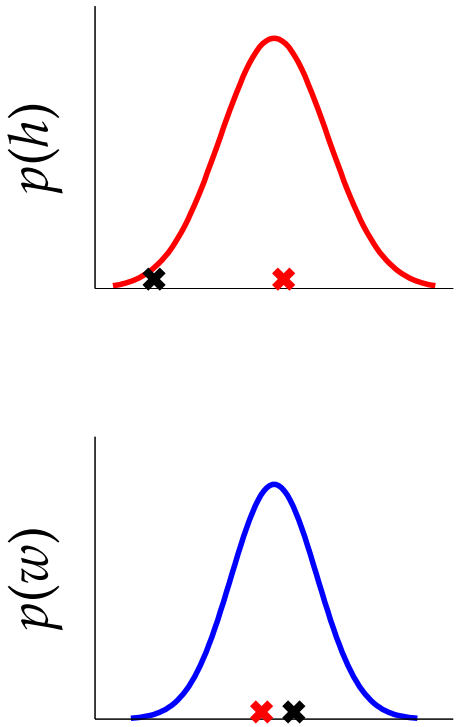
Samples of height and weight

Sampling Two Dimensional Variables

Marginal Distributions

Joint Distribution



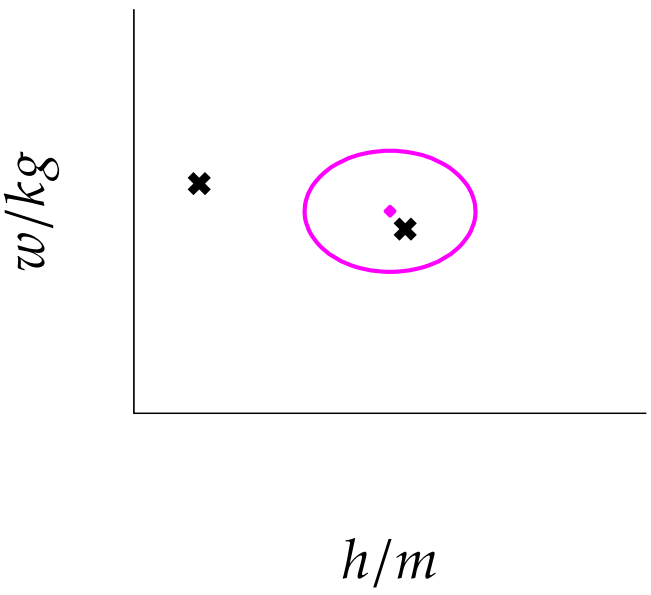


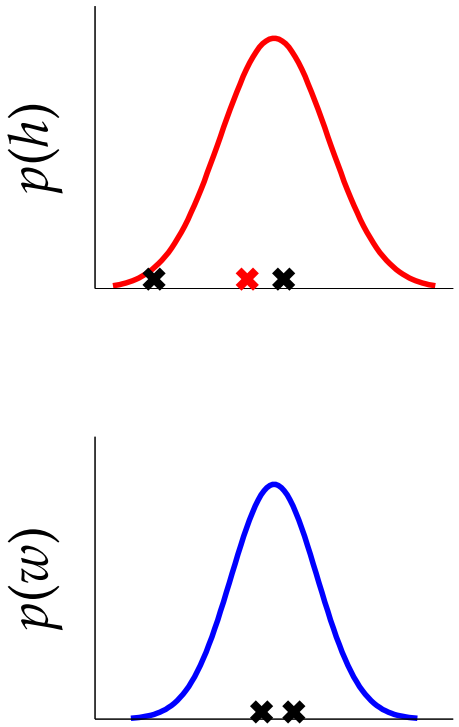
Samples of height and weight

Sampling Two Dimensional Variables

Marginal Distributions

Joint Distribution



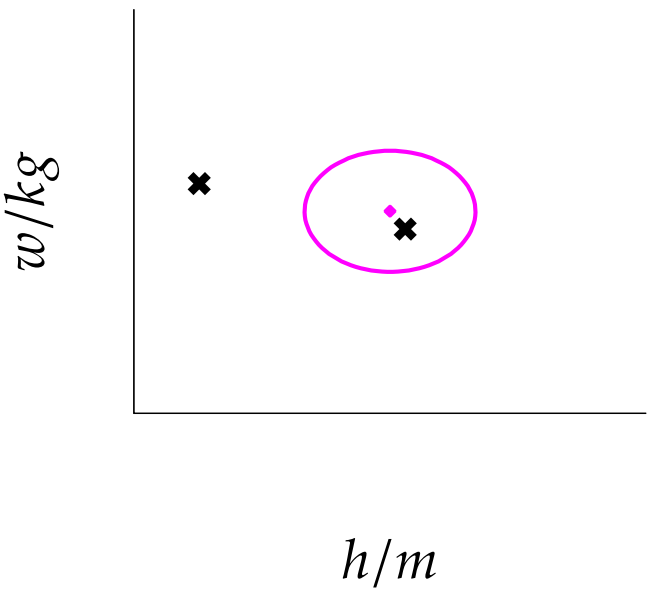


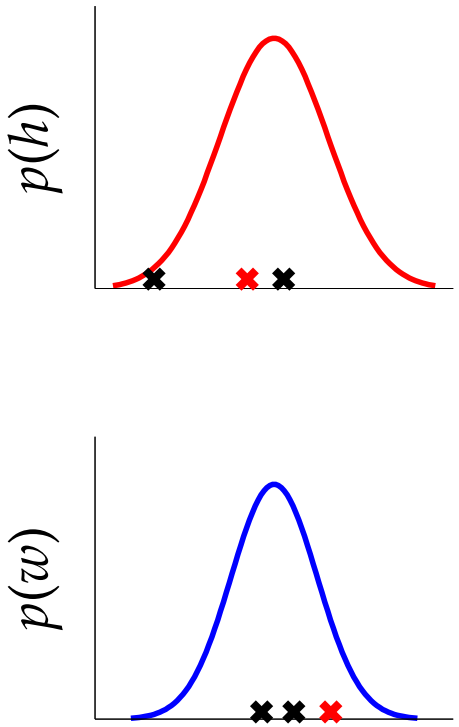
Samples of height and weight

Sampling Two Dimensional Variables

Marginal Distributions

Joint Distribution



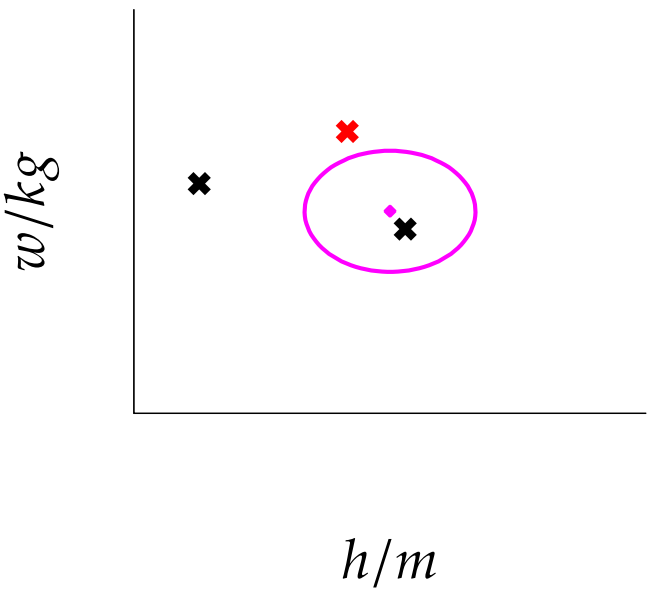


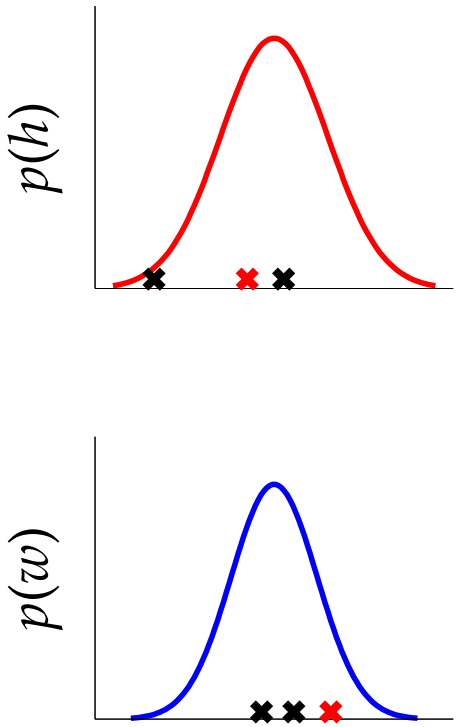
Samples of height and weight

Sampling Two Dimensional Variables

Marginal Distributions

Joint Distribution



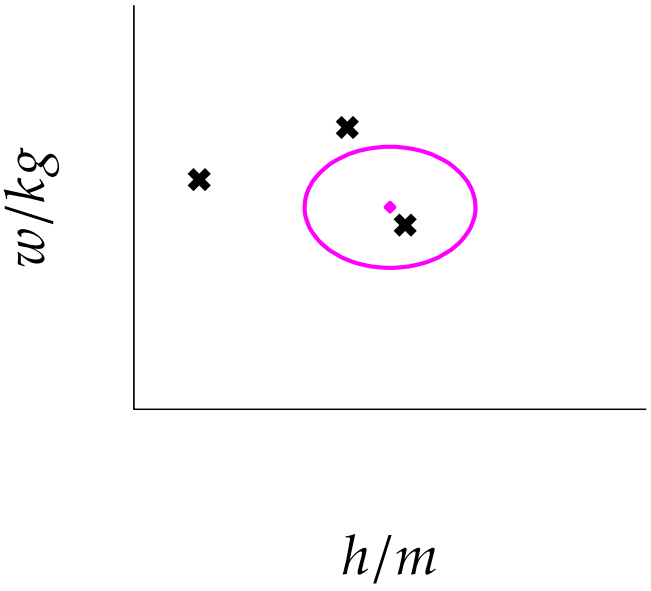


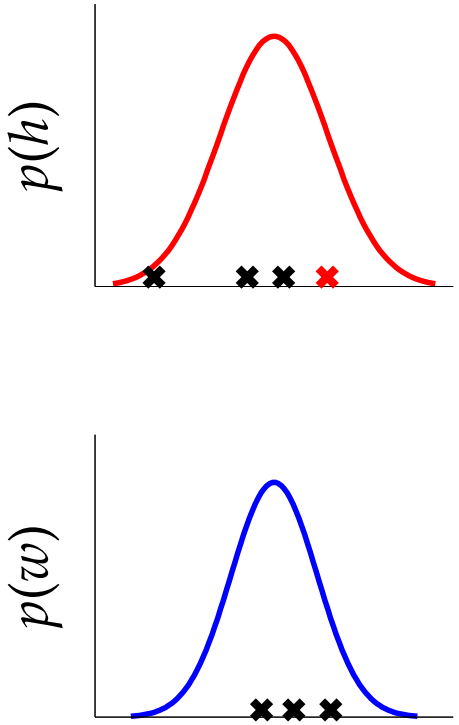
Samples of height and weight

Sampling Two Dimensional Variables

Marginal Distributions

Joint Distribution



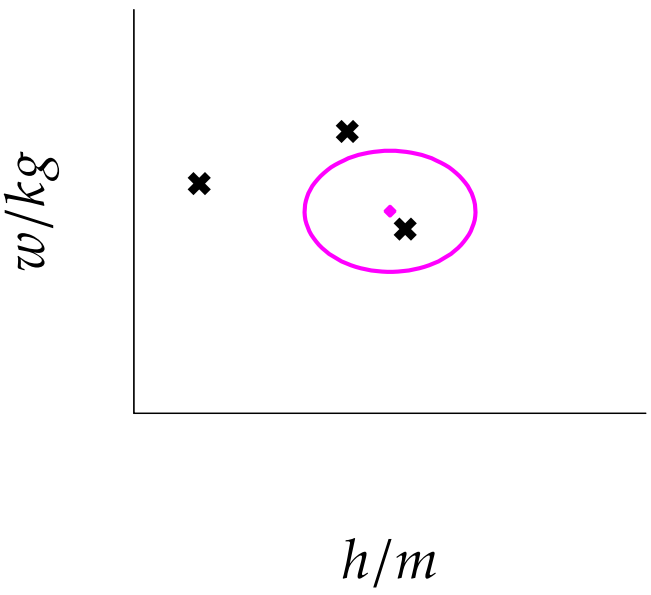


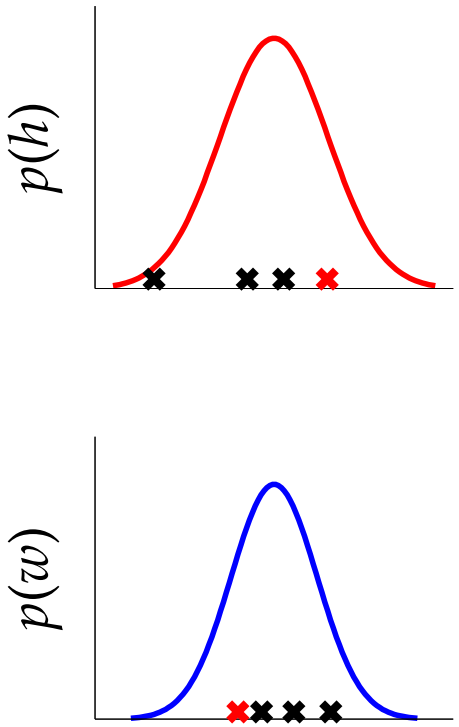
Samples of height and weight

Sampling Two Dimensional Variables

Marginal Distributions

Joint Distribution



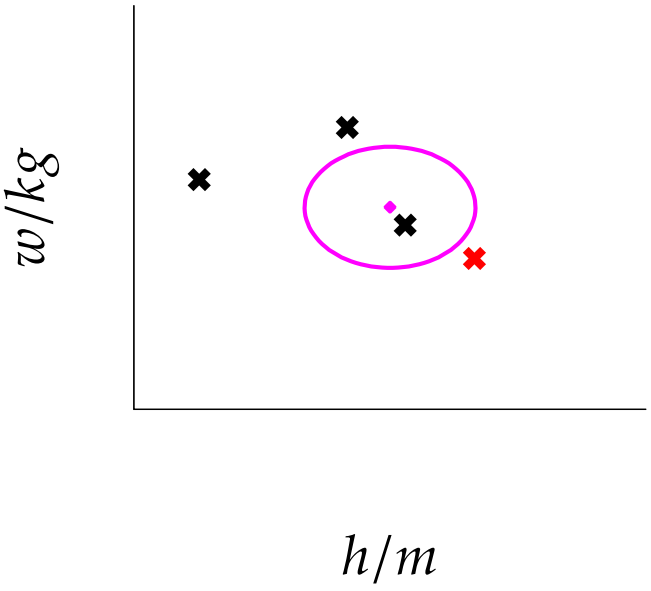


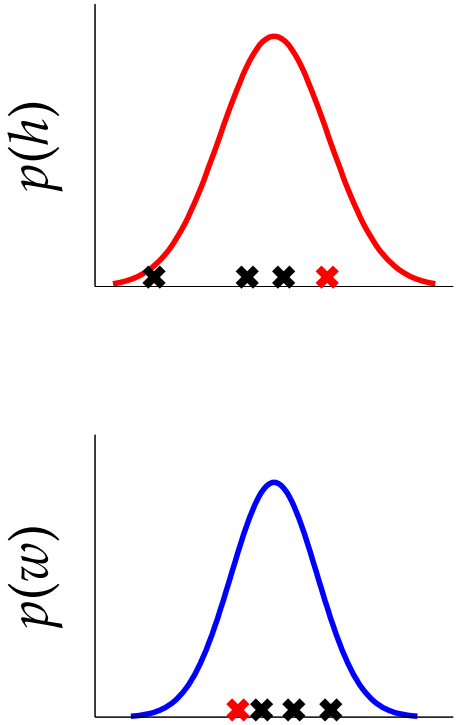
Samples of height and weight

Sampling Two Dimensional Variables

Marginal Distributions

Joint Distribution



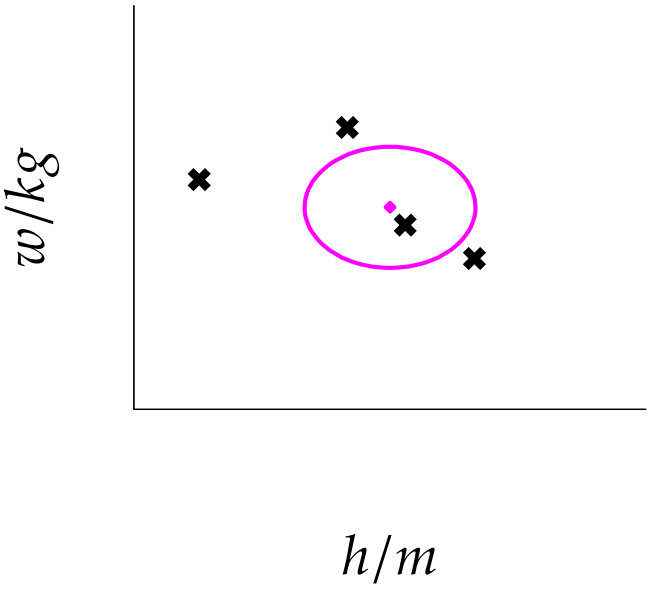


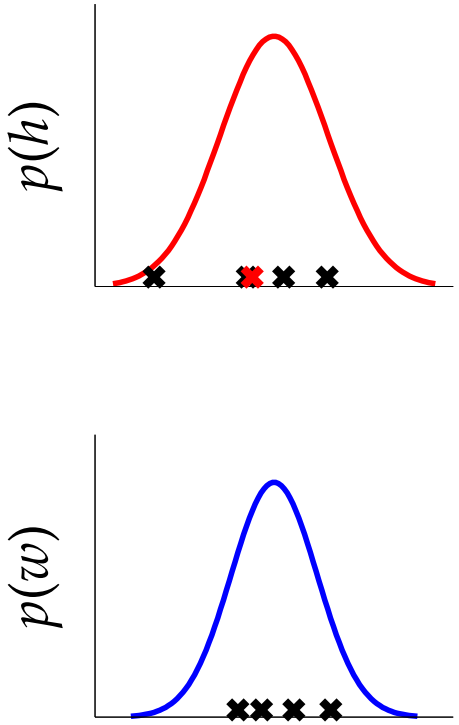
Samples of height and weight

Sampling Two Dimensional Variables

Marginal Distributions

Joint Distribution



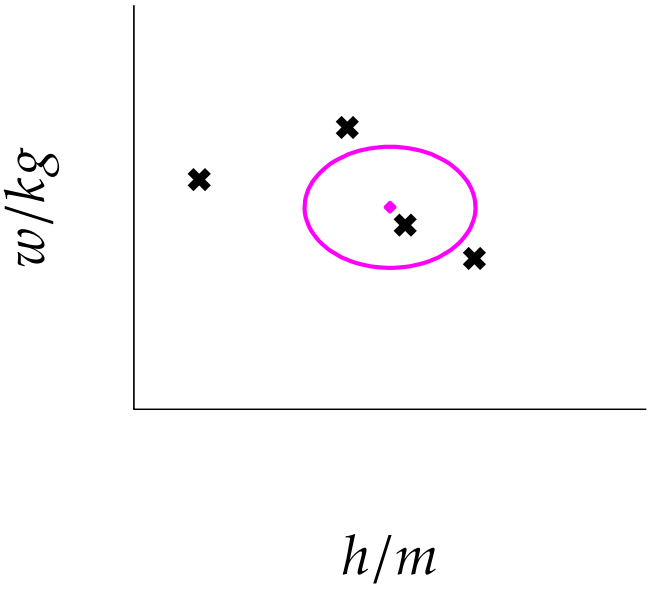


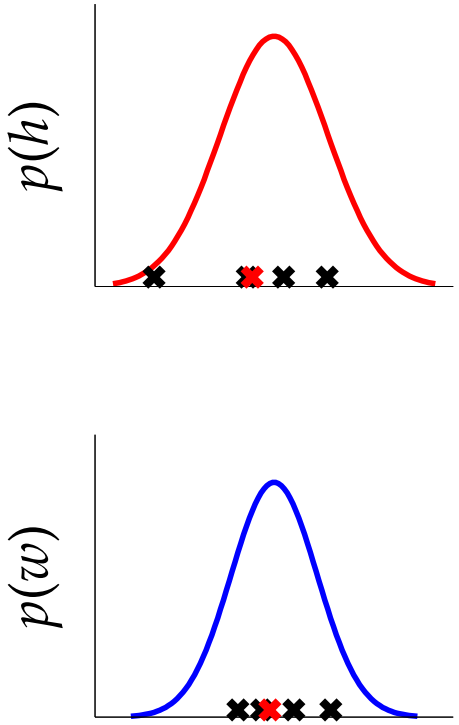
Samples of height and weight

Sampling Two Dimensional Variables

Marginal Distributions

Joint Distribution



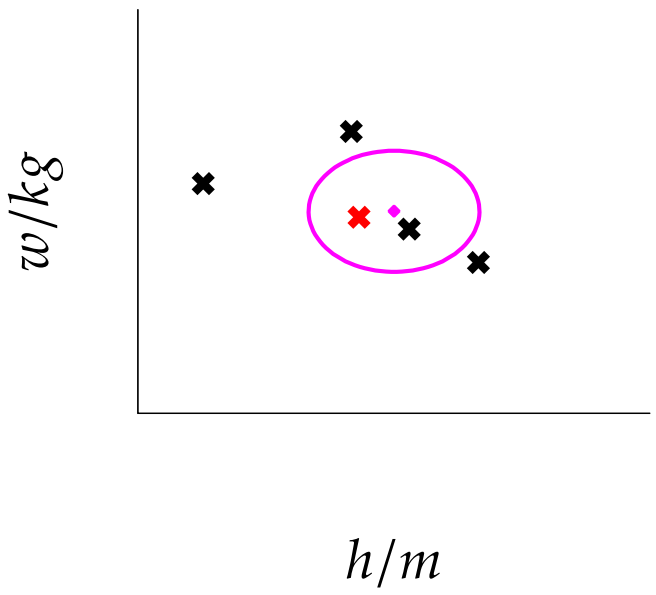


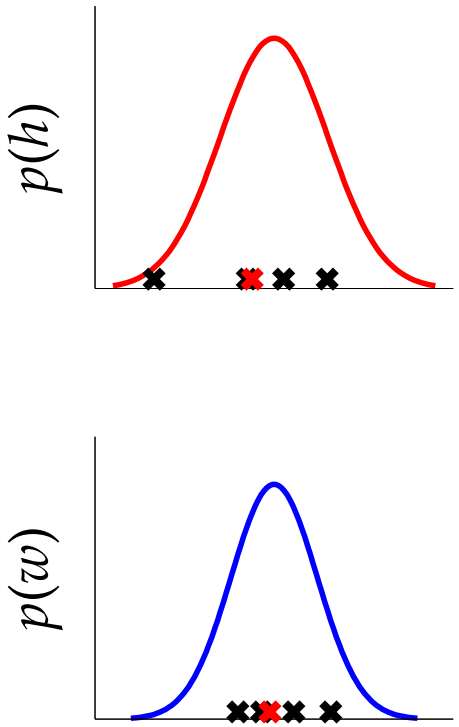
Samples of height and weight

Sampling Two Dimensional Variables

Marginal Distributions

Joint Distribution



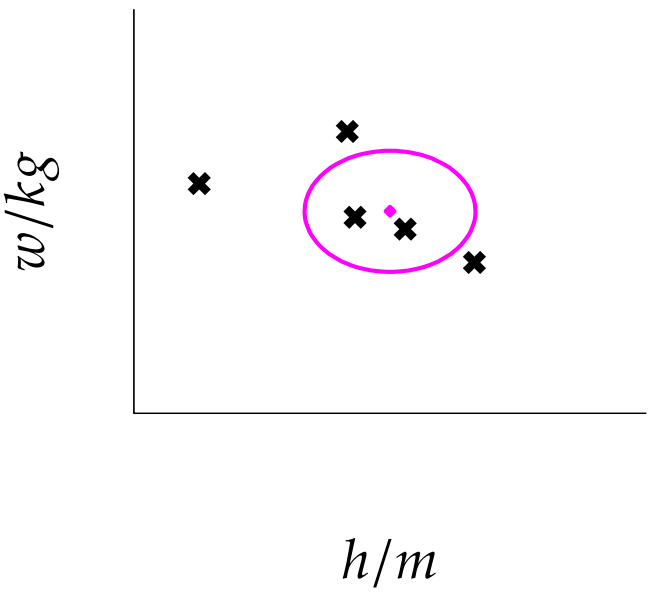


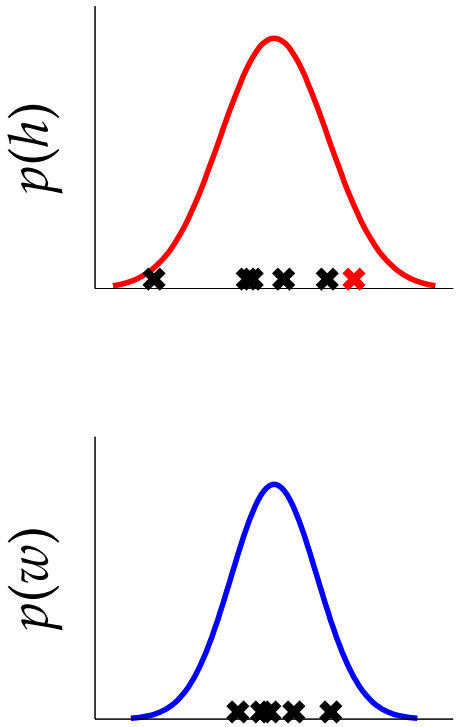
Samples of height and weight

Sampling Two Dimensional Variables

Marginal Distributions

Joint Distribution



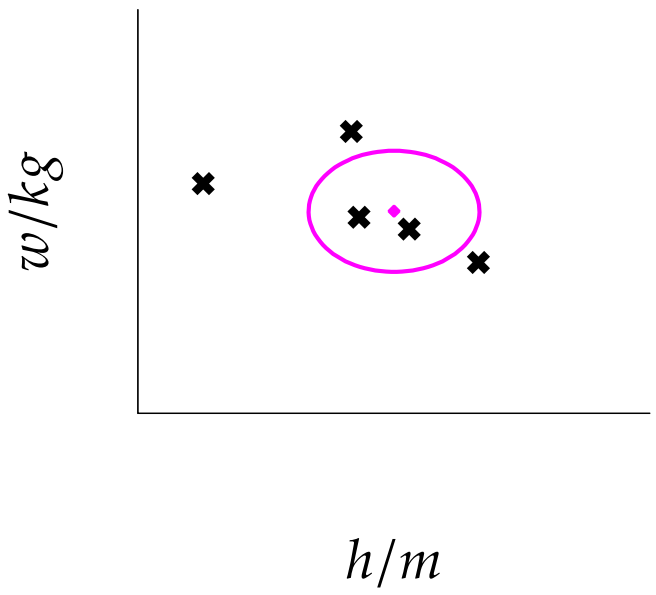


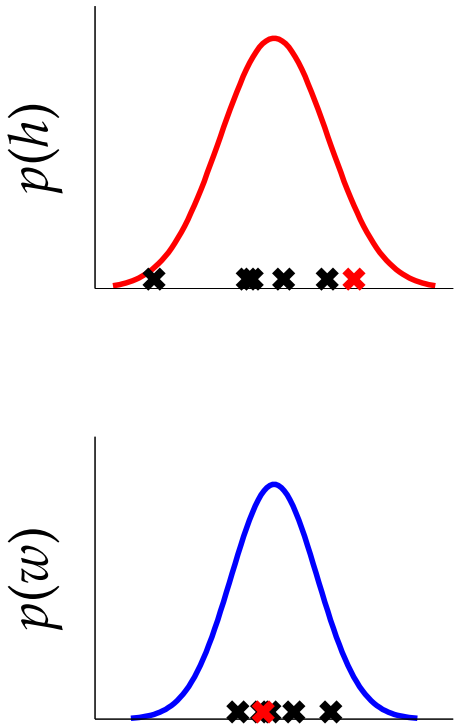
Samples of height and weight

Sampling Two Dimensional Variables

Marginal Distributions

Joint Distribution



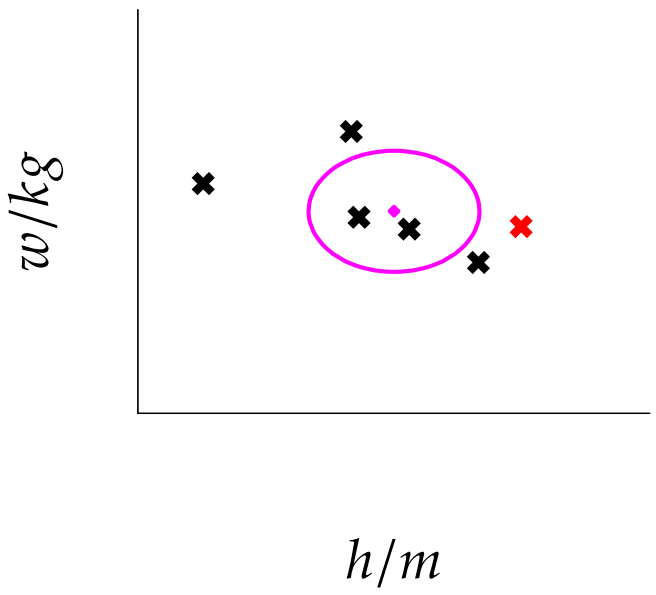


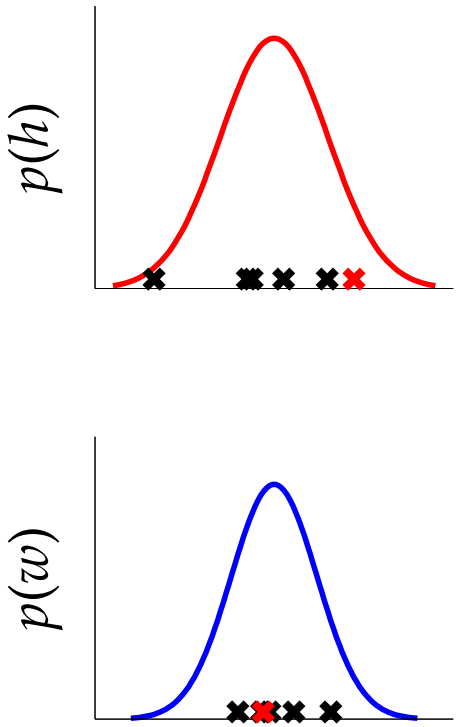
Samples of height and weight

Sampling Two Dimensional Variables

Marginal Distributions

Joint Distribution



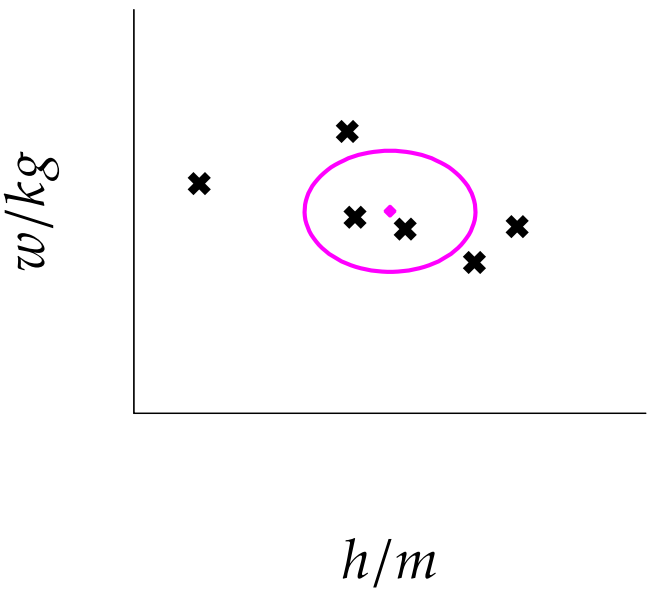


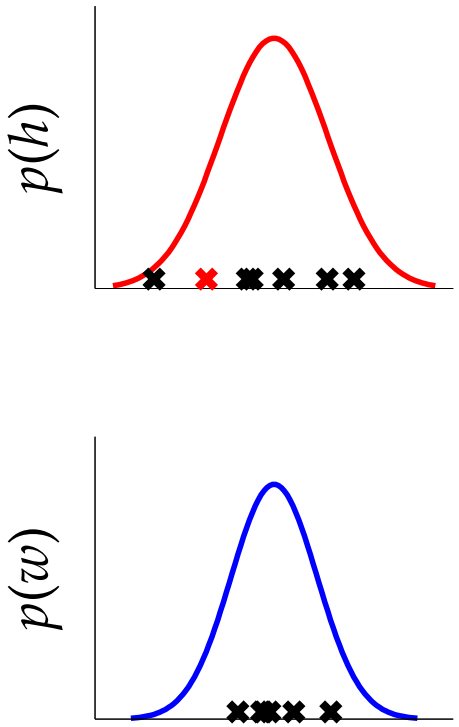
Samples of height and weight

Sampling Two Dimensional Variables

Marginal Distributions

Joint Distribution



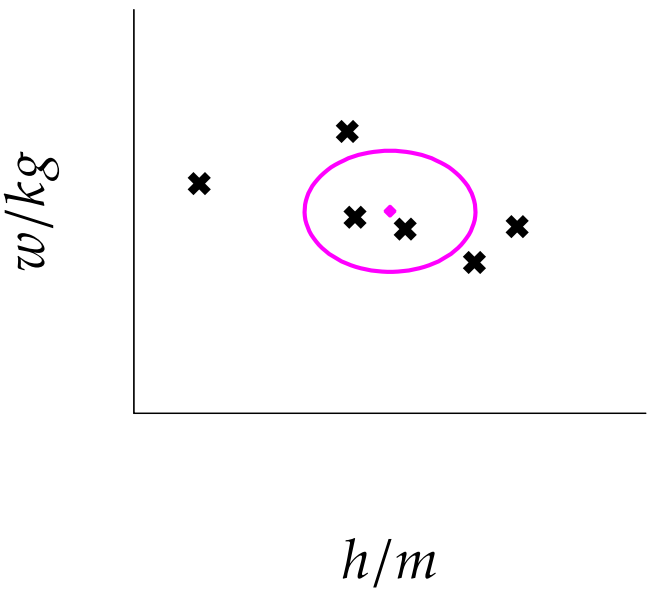


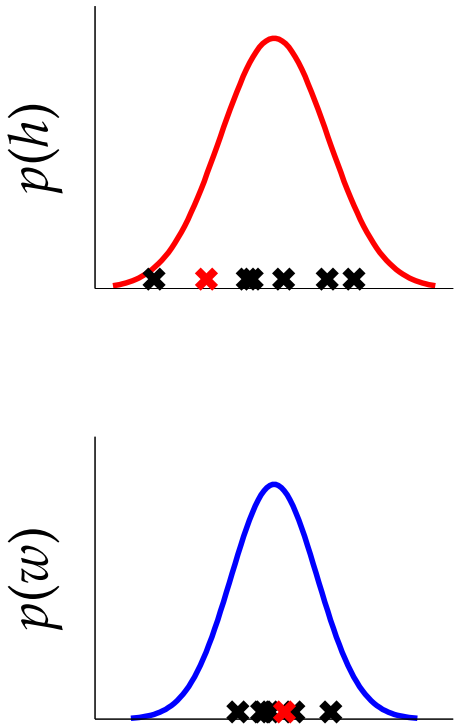
Samples of height and weight

Sampling Two Dimensional Variables

Marginal Distributions

Joint Distribution



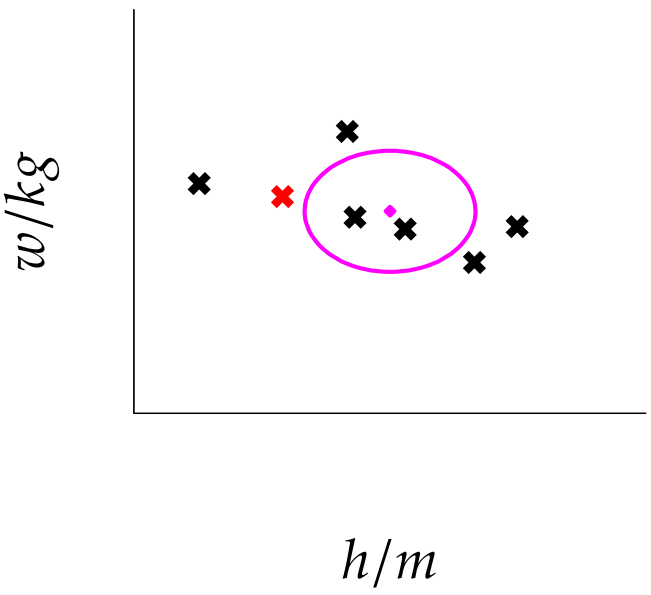


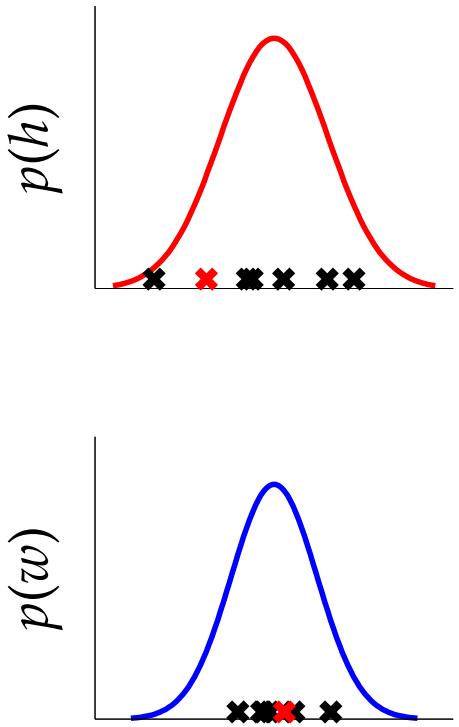
Samples of height and weight

Sampling Two Dimensional Variables

Marginal Distributions

Joint Distribution



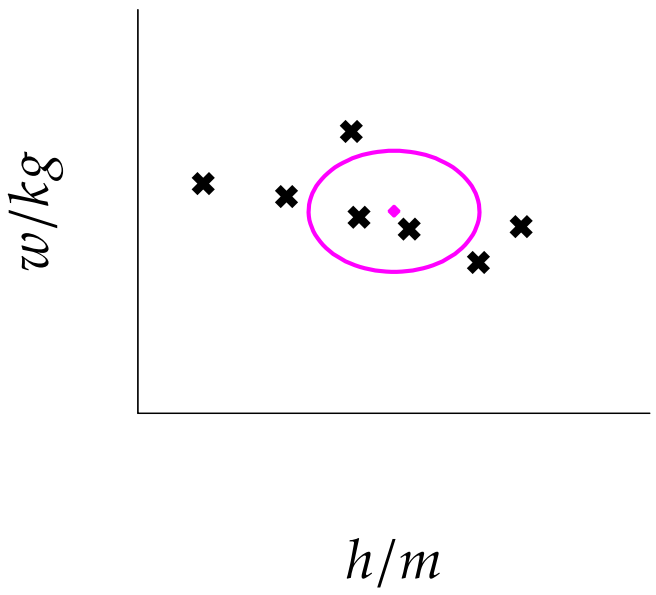


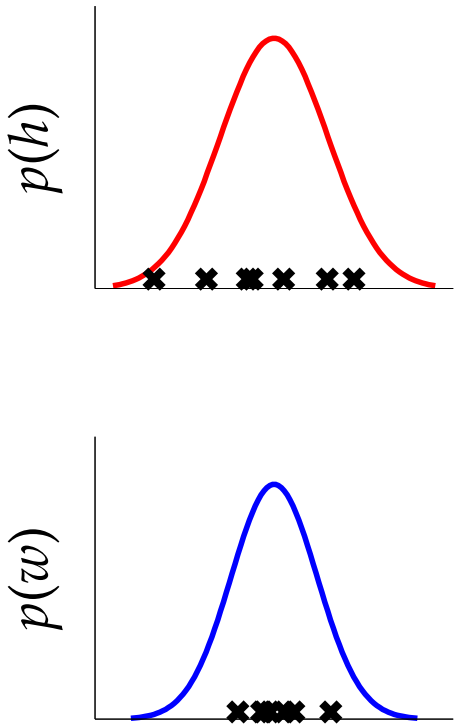
Samples of height and weight

Sampling Two Dimensional Variables

Marginal Distributions

Joint Distribution





Samples of height and weight

Independence Assumption

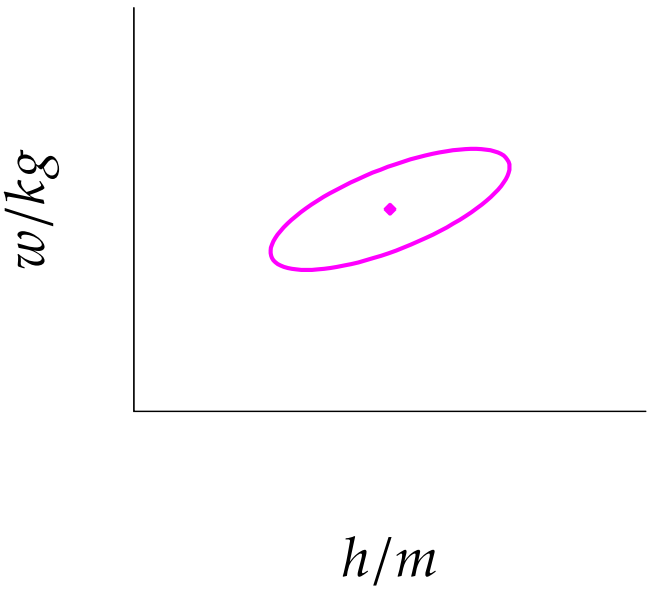
- This assumes height and weight are independent.

$$p(h, w) = p(h)p(w)$$

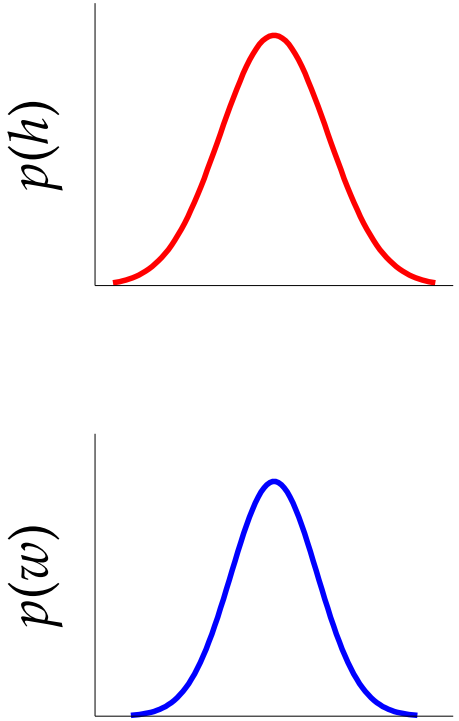
- In reality they are dependent (body mass index) = $\frac{w}{h^2}$.

Sampling Two Dimensional Variables

Joint Distribution

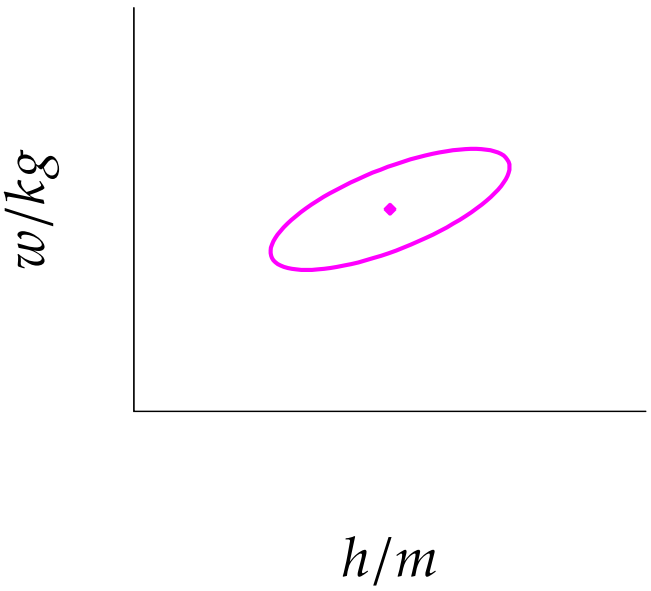


Marginal Distributions

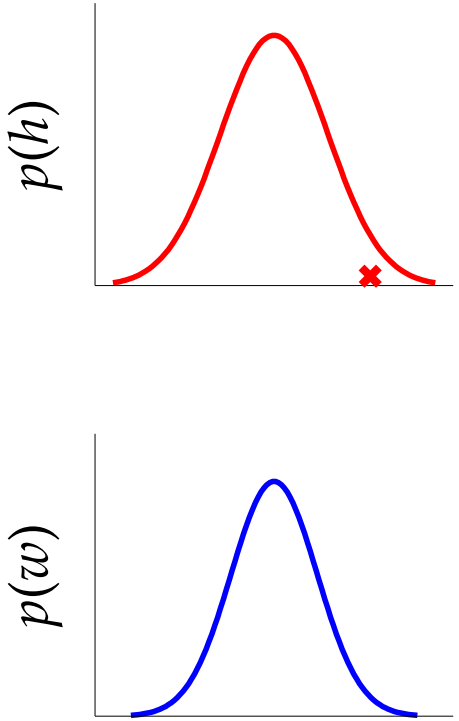


Sampling Two Dimensional Variables

Joint Distribution

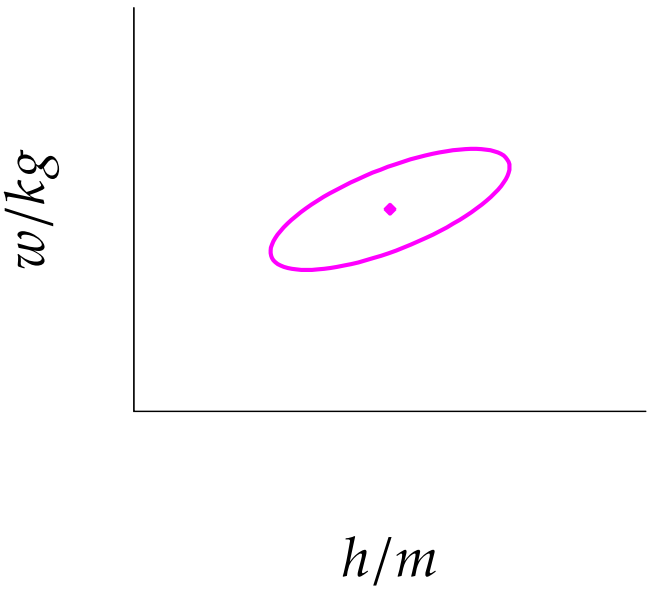


Marginal Distributions

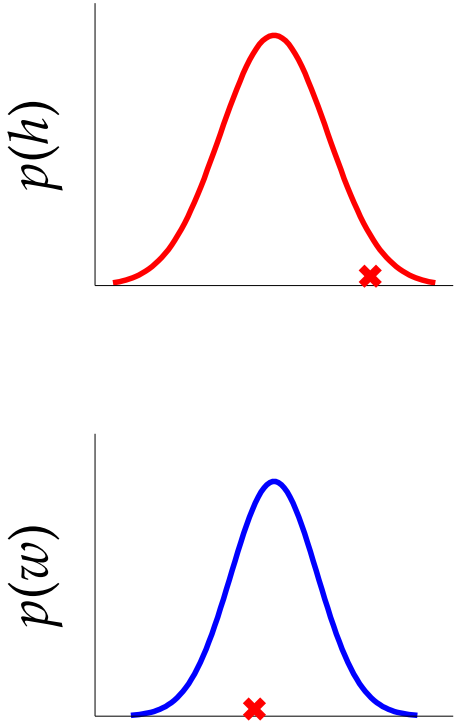


Sampling Two Dimensional Variables

Joint Distribution

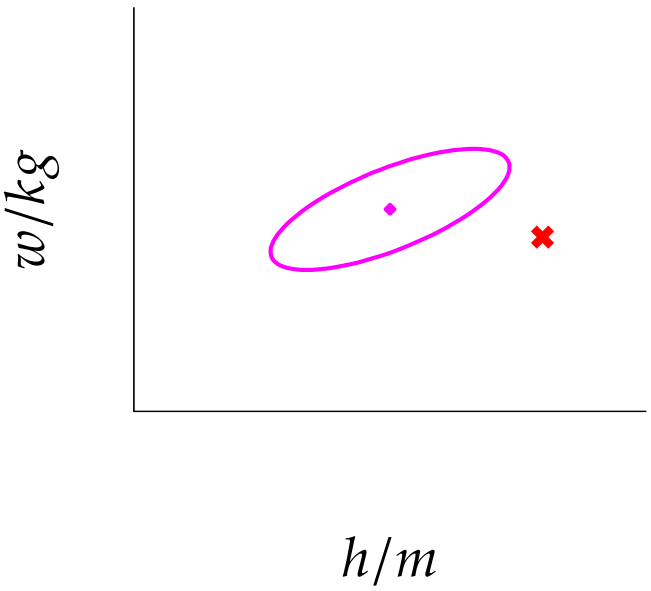


Marginal Distributions

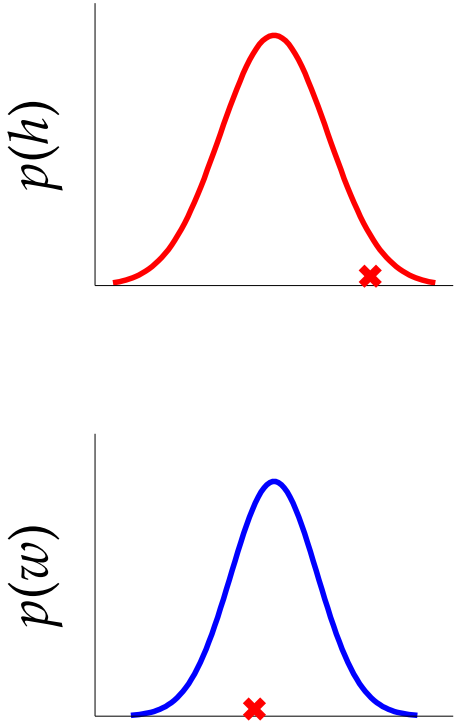


Sampling Two Dimensional Variables

Joint Distribution

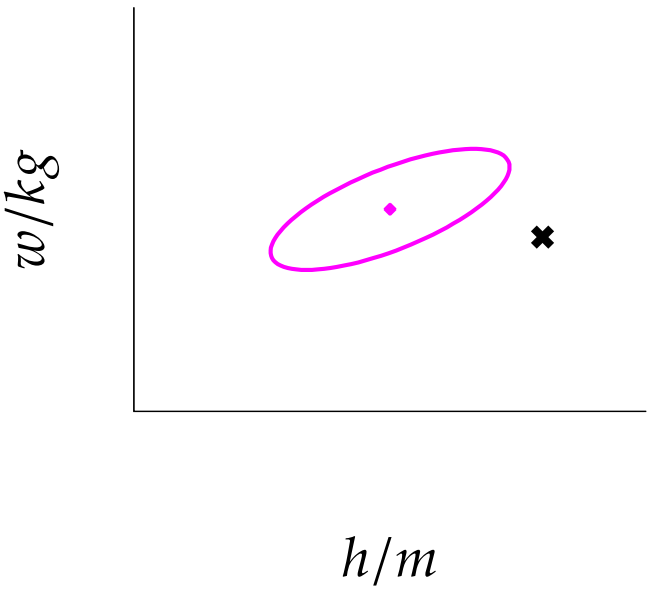


Marginal Distributions

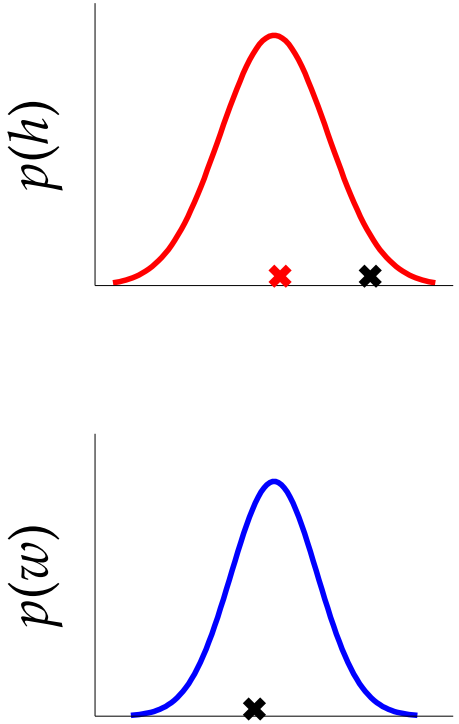


Sampling Two Dimensional Variables

Joint Distribution

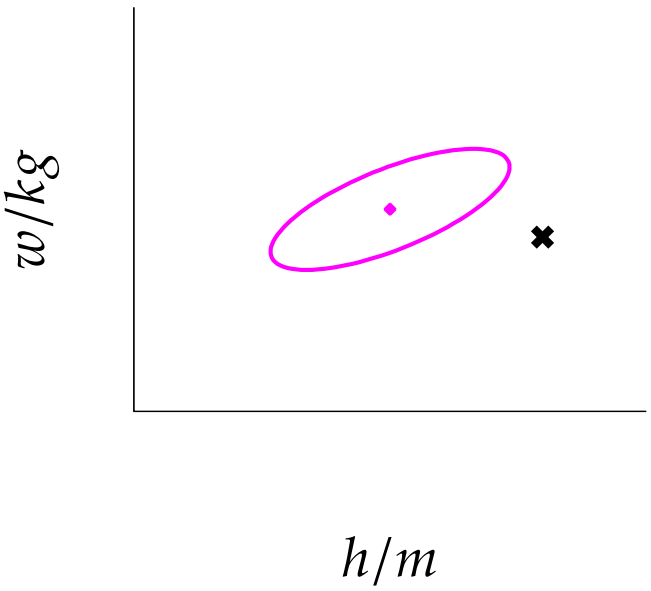


Marginal Distributions

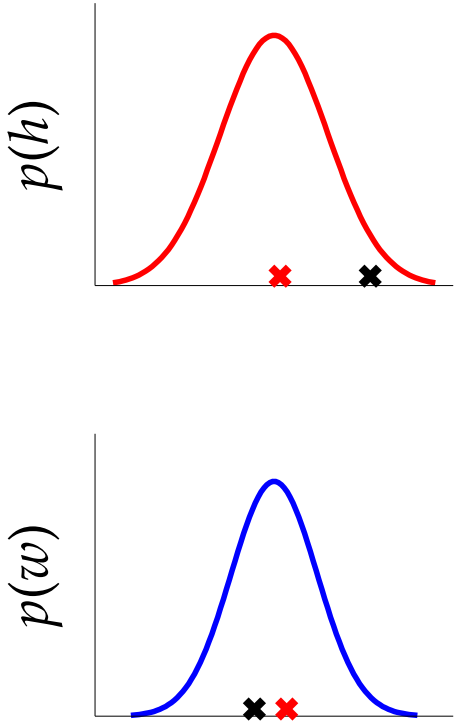


Sampling Two Dimensional Variables

Joint Distribution

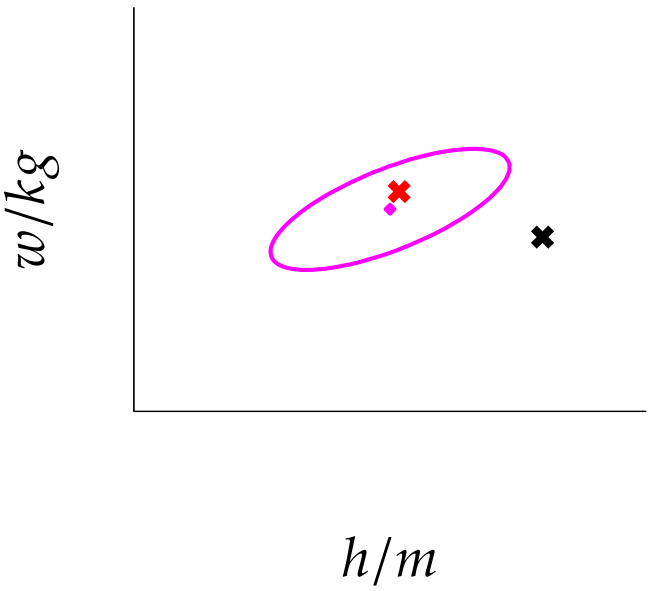


Marginal Distributions

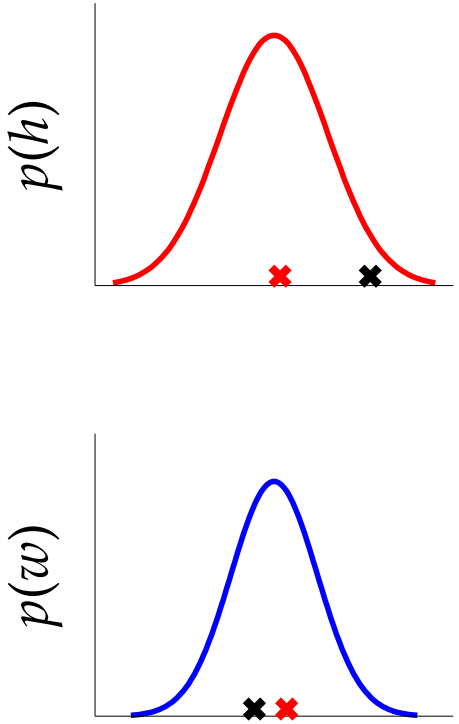


Sampling Two Dimensional Variables

Joint Distribution

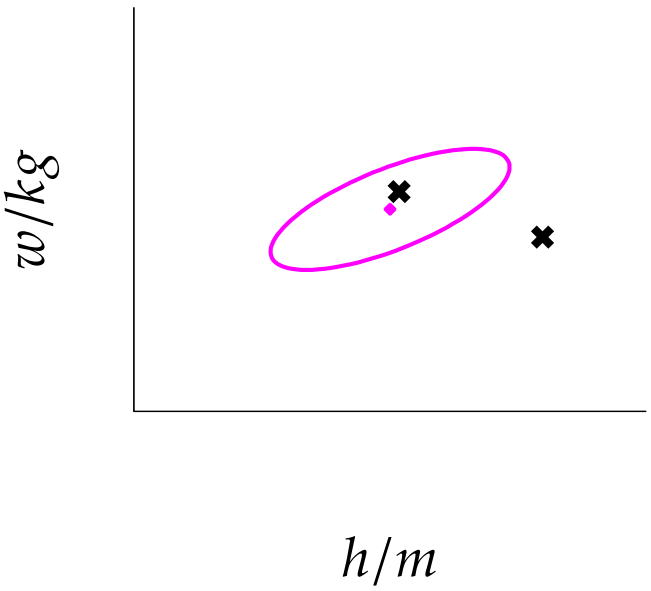


Marginal Distributions

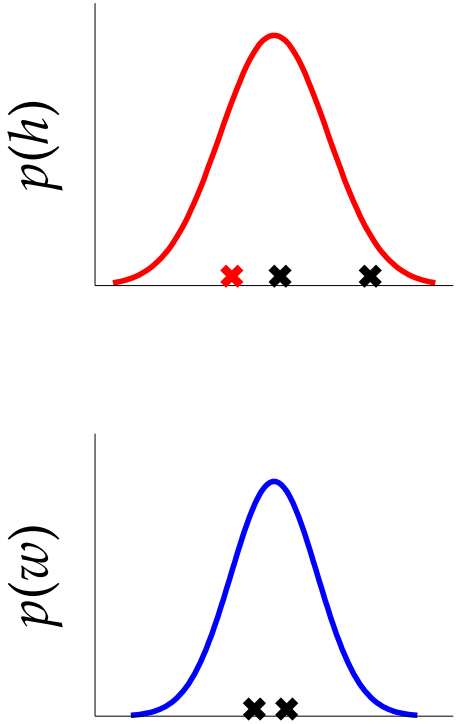


Sampling Two Dimensional Variables

Joint Distribution

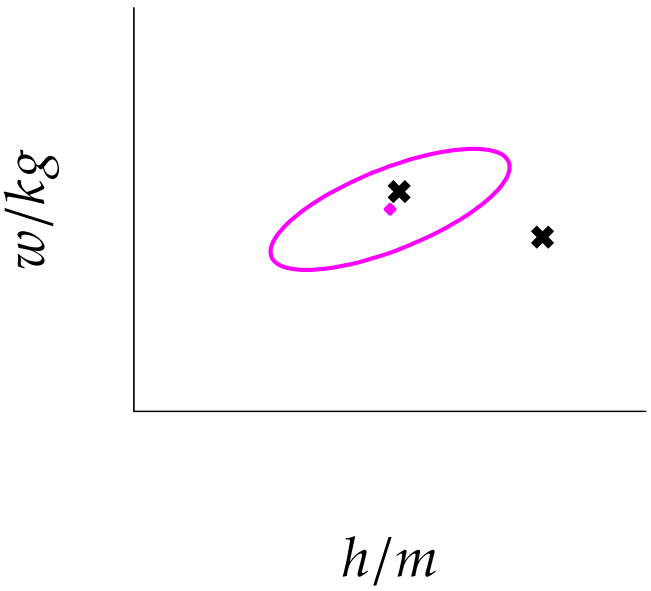


Marginal Distributions

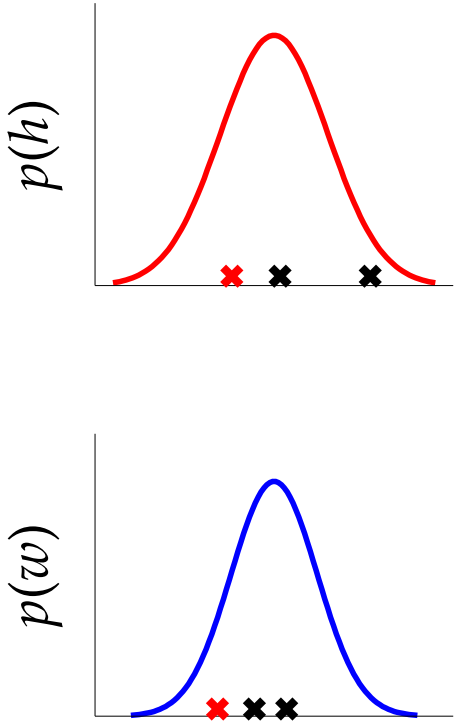


Sampling Two Dimensional Variables

Joint Distribution

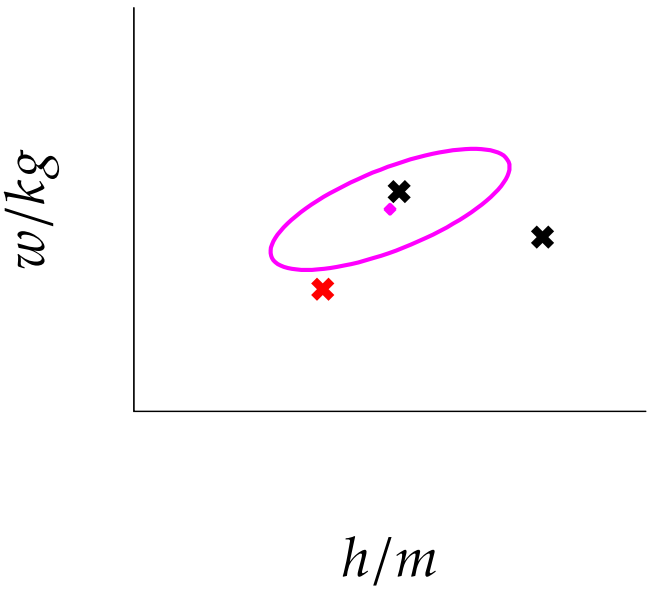


Marginal Distributions

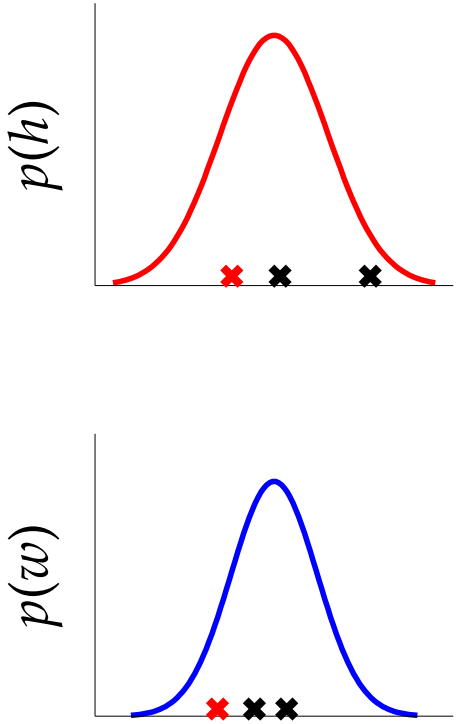


Sampling Two Dimensional Variables

Joint Distribution



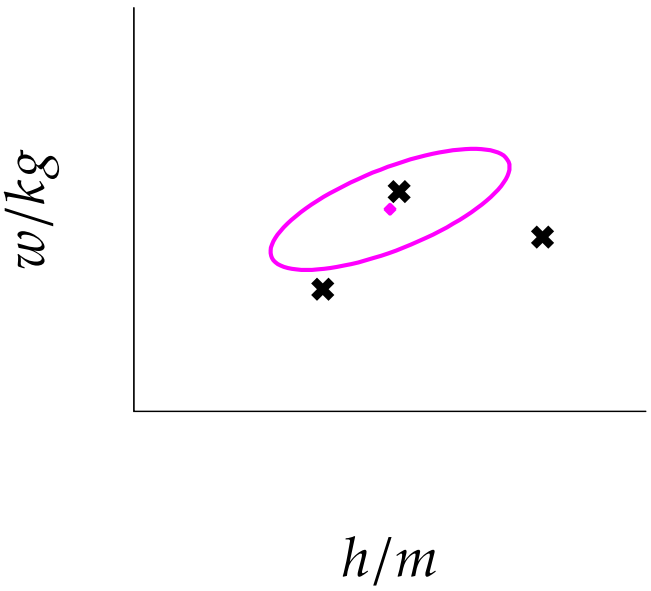
Marginal Distributions

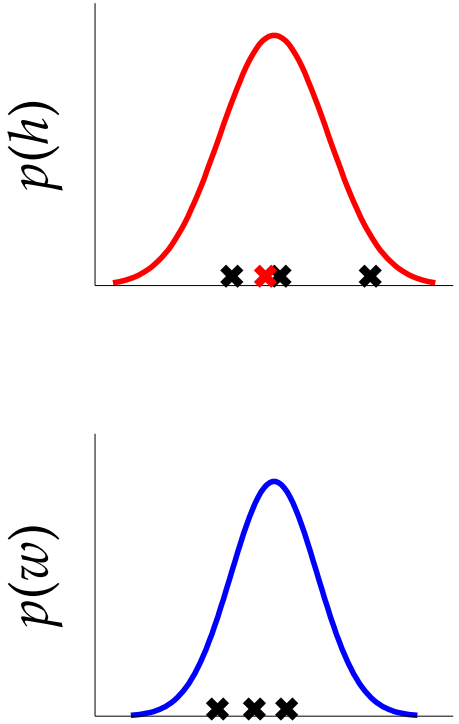


Sampling Two Dimensional Variables

Marginal Distributions

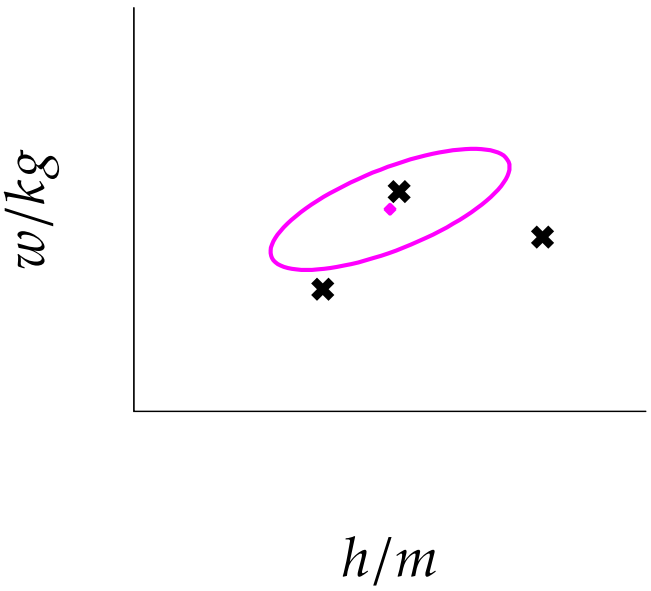
Joint Distribution



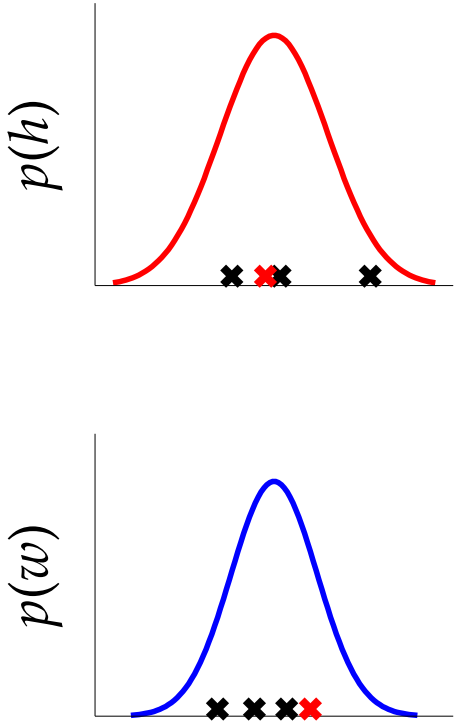


Sampling Two Dimensional Variables

Joint Distribution

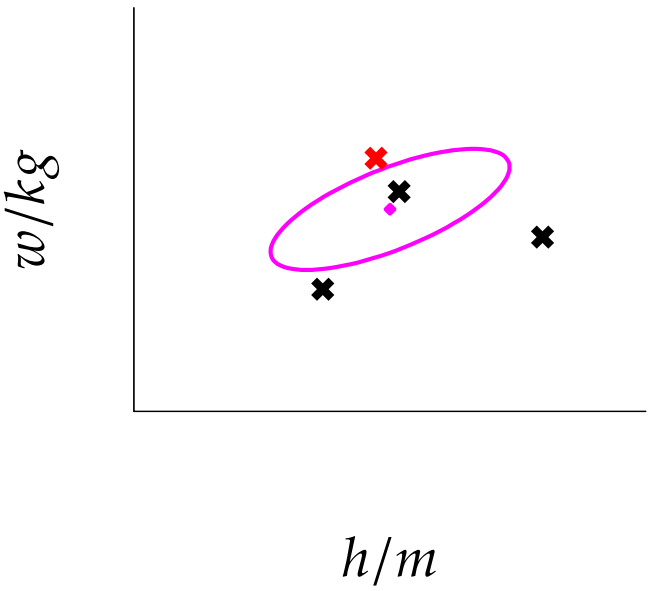


Marginal Distributions

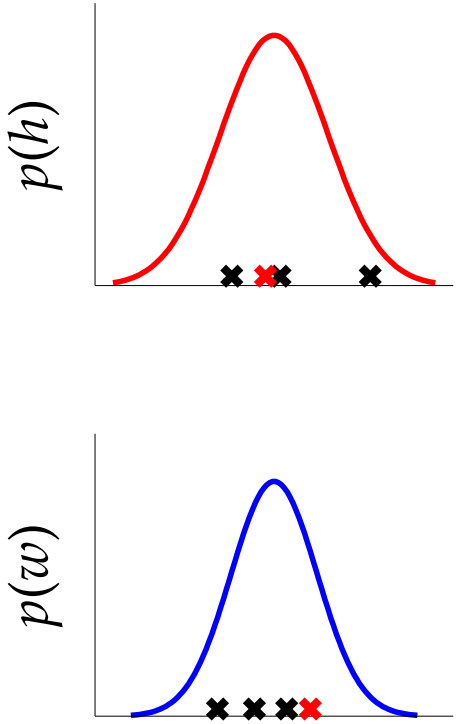


Sampling Two Dimensional Variables

Joint Distribution



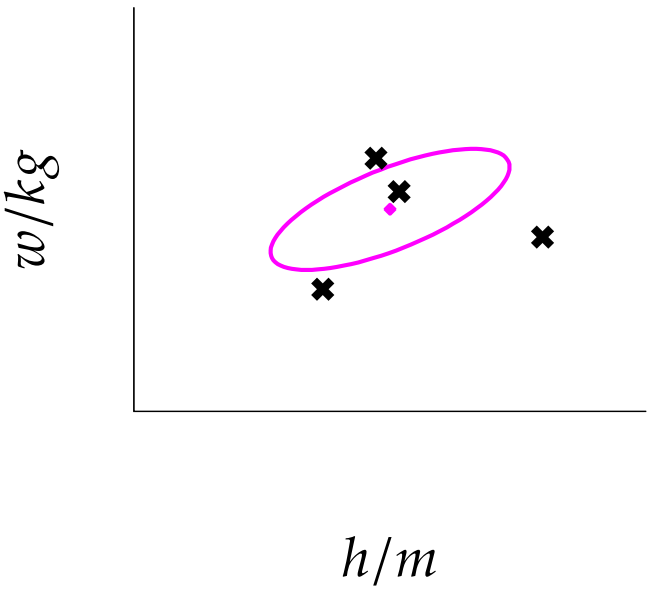
Marginal Distributions

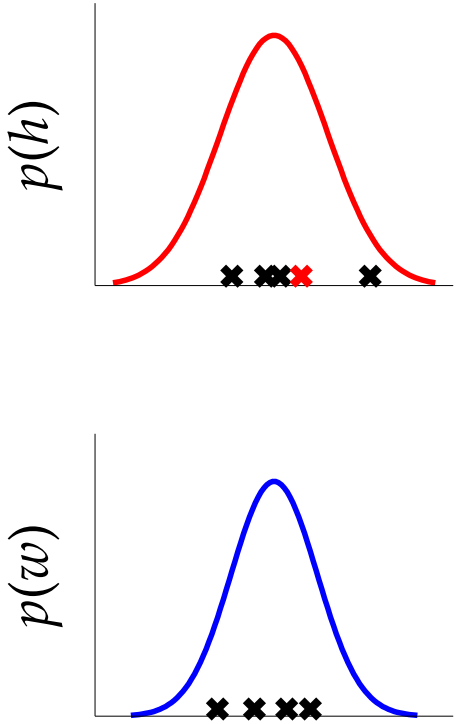


Sampling Two Dimensional Variables

Marginal Distributions

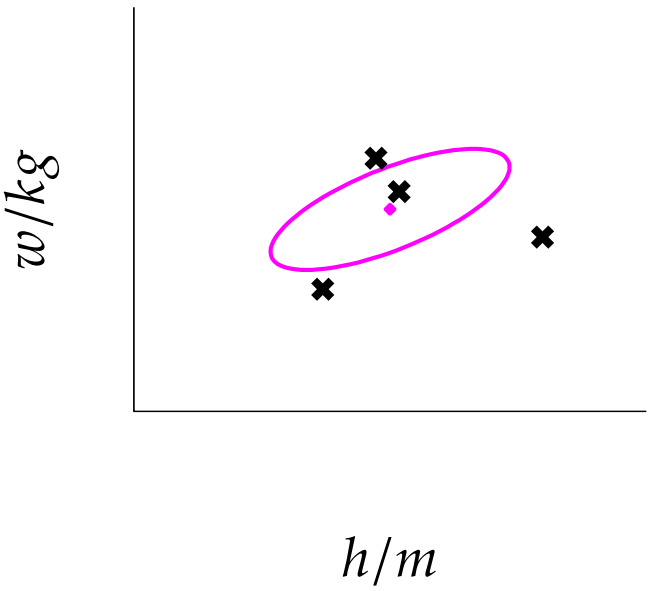
Joint Distribution



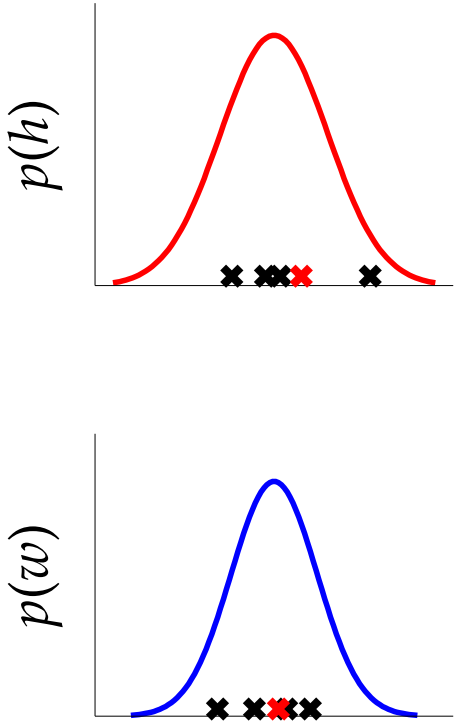


Sampling Two Dimensional Variables

Joint Distribution

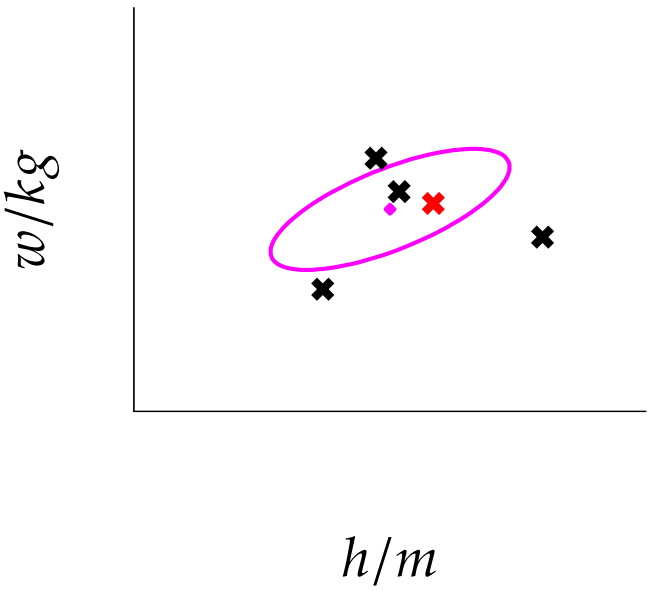


Marginal Distributions

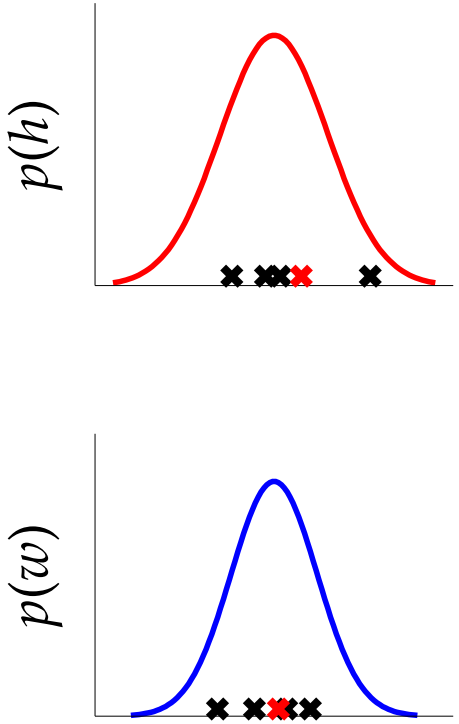


Sampling Two Dimensional Variables

Joint Distribution

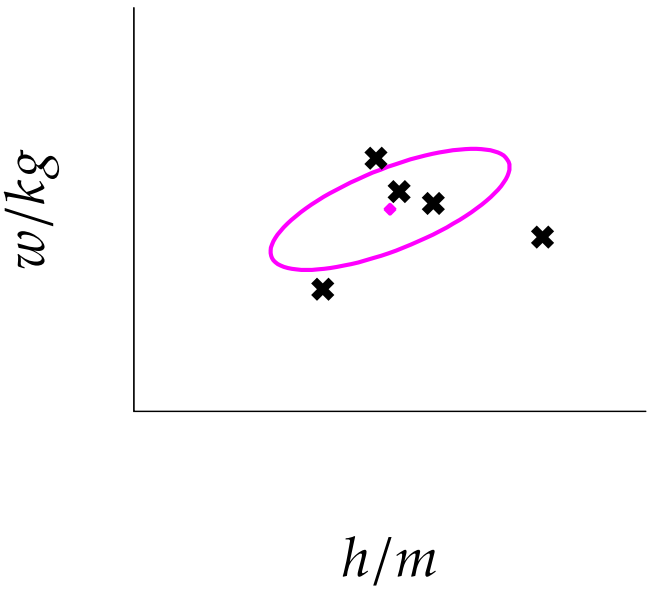


Marginal Distributions

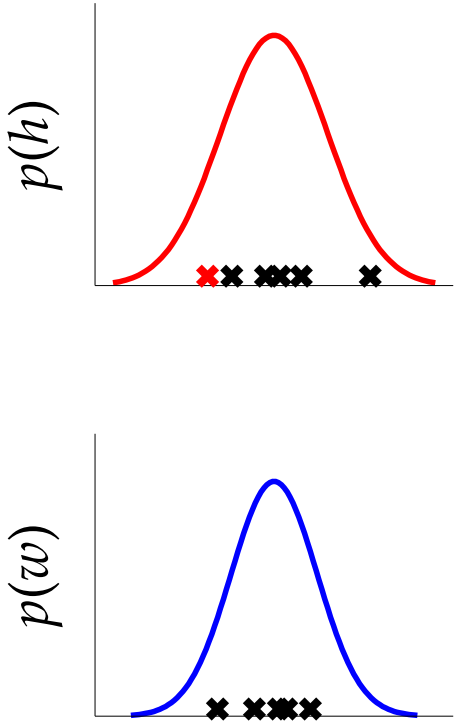


Sampling Two Dimensional Variables

Joint Distribution

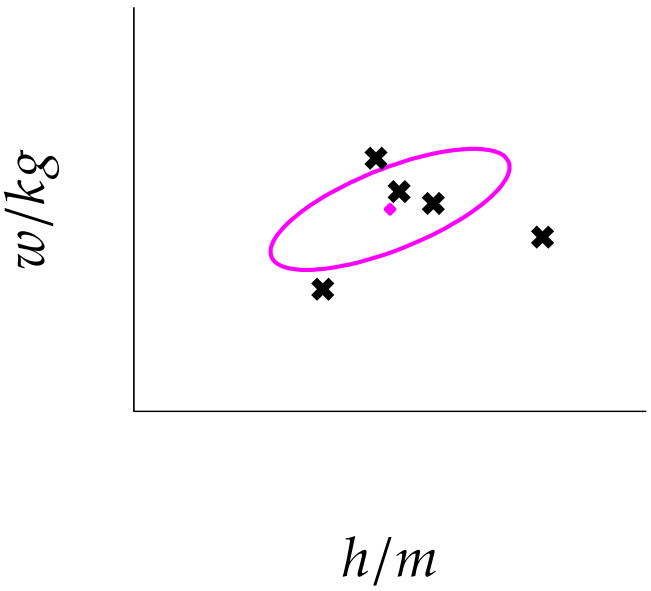


Marginal Distributions

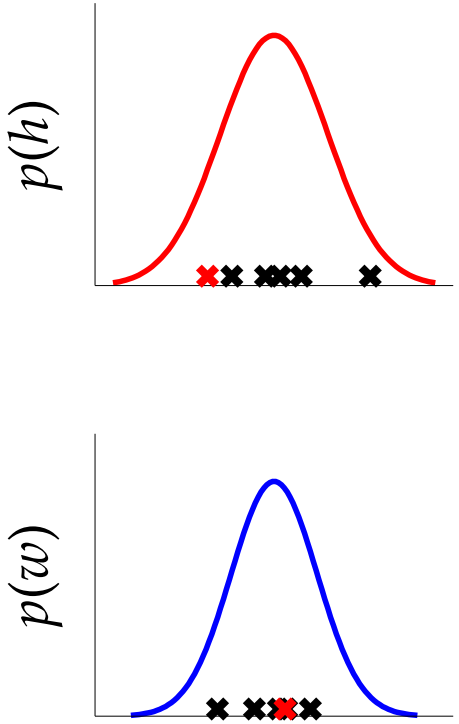


Sampling Two Dimensional Variables

Joint Distribution

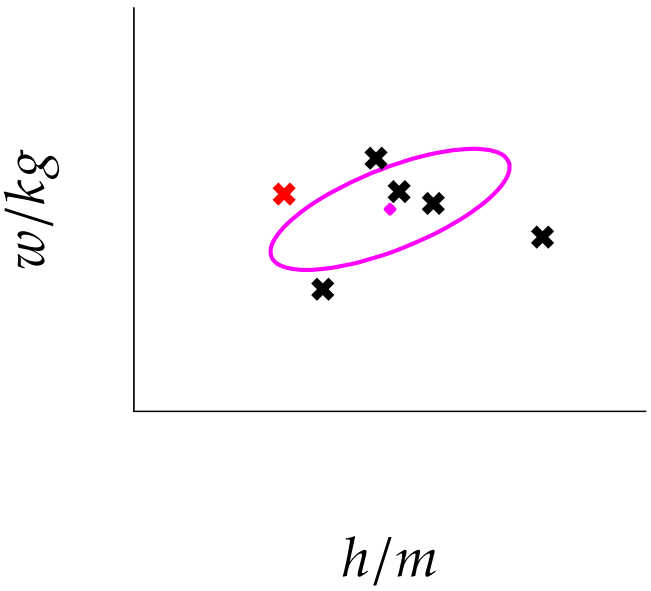


Marginal Distributions

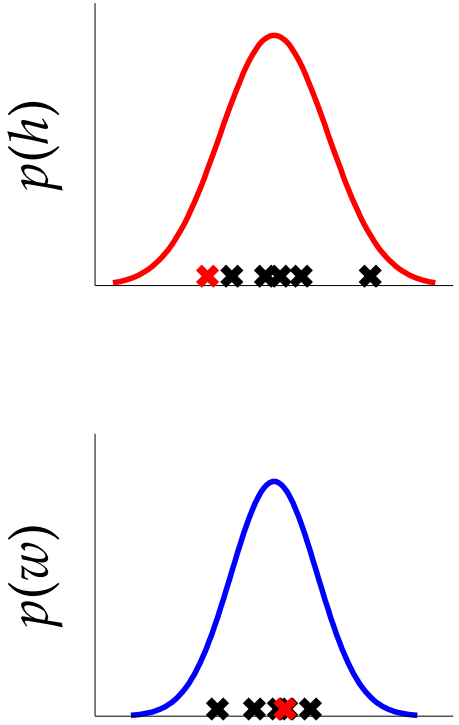


Sampling Two Dimensional Variables

Joint Distribution

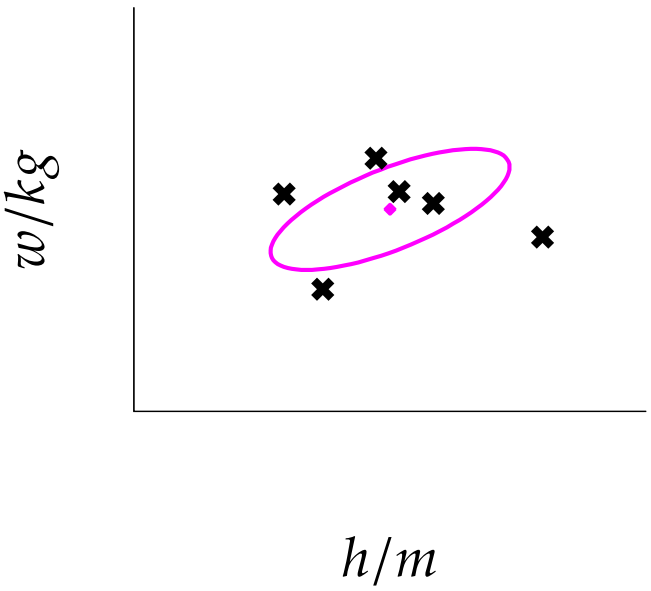


Marginal Distributions

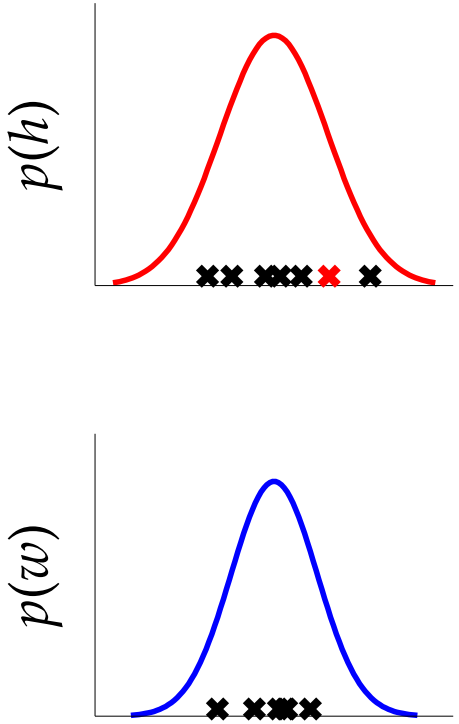


Sampling Two Dimensional Variables

Joint Distribution

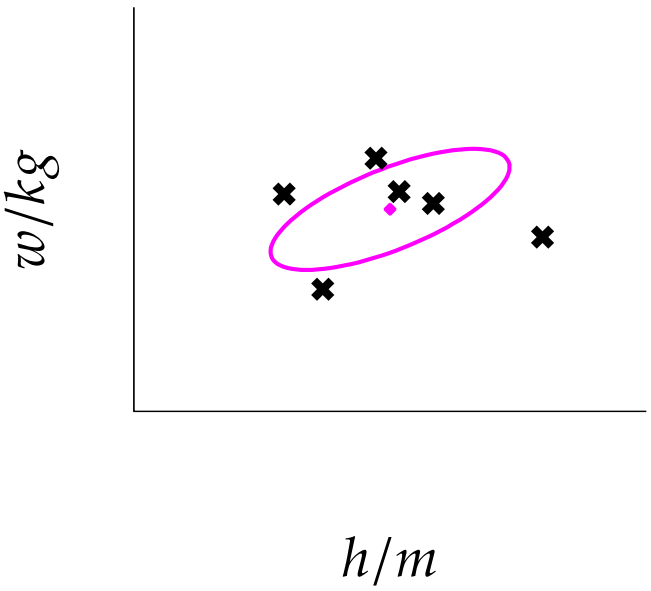


Marginal Distributions

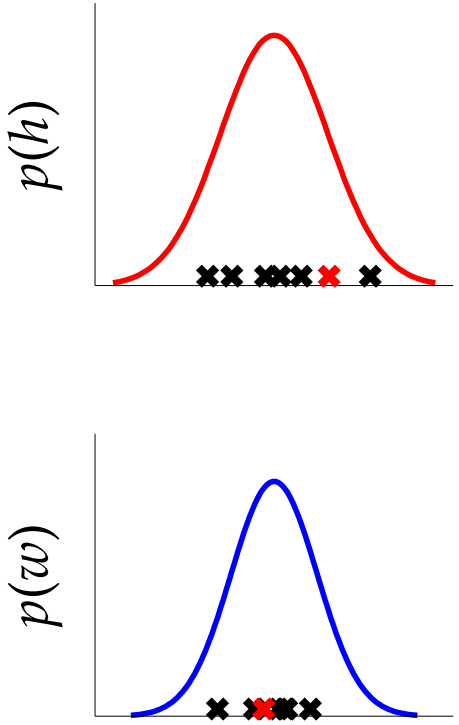


Sampling Two Dimensional Variables

Joint Distribution

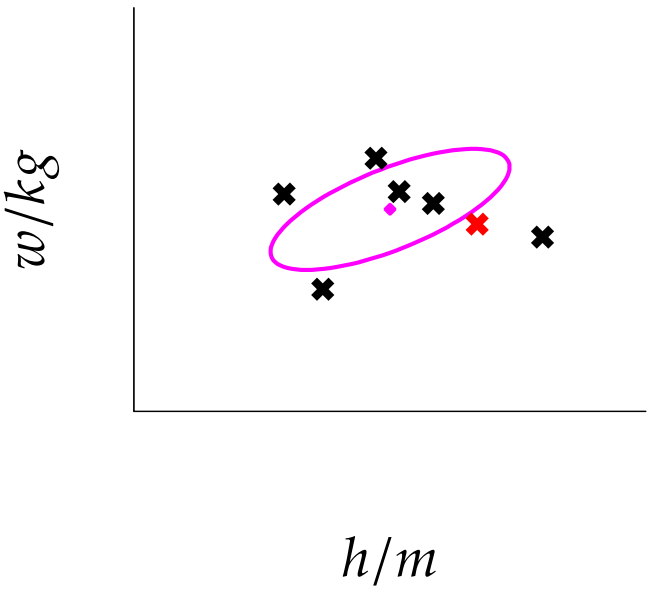


Marginal Distributions

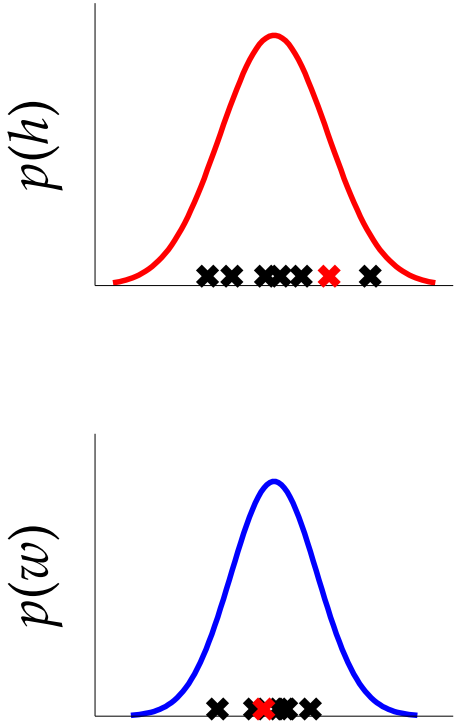


Sampling Two Dimensional Variables

Joint Distribution

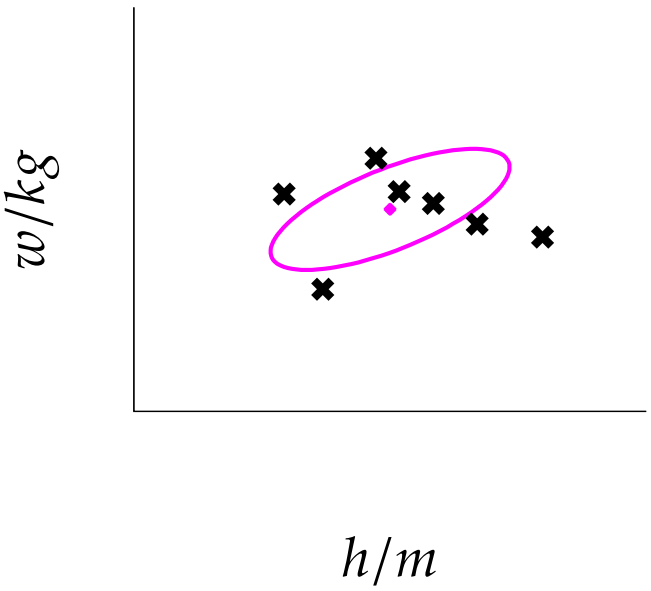


Marginal Distributions

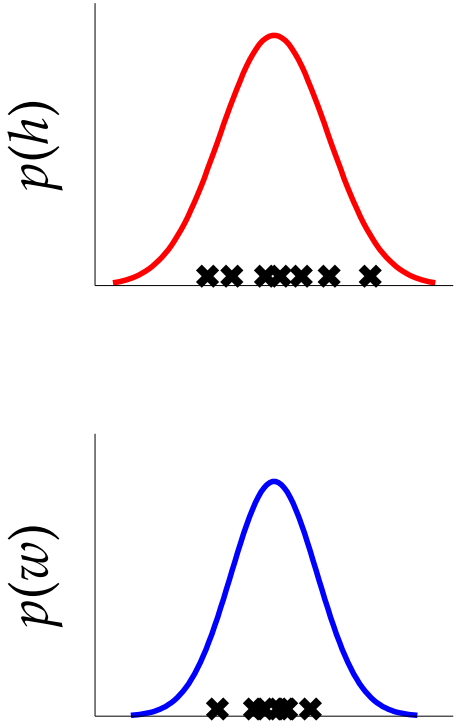


Sampling Two Dimensional Variables

Joint Distribution



Marginal Distributions



Independent Gaussians

$$p(w, h) = p(w)p(h)$$

Independent Gaussians

$$p(w,h) = \frac{1}{\sqrt{2\pi\sigma_1^2}\sqrt{2\pi\sigma_2^2}} \exp\left(-\frac{1}{2}\left(\frac{(w-\mu_1)^2}{\sigma_1^2} + \frac{(h-\mu_2)^2}{\sigma_2^2}\right)\right)$$

Independent Gaussians

$$p(w,h) = \frac{1}{2\pi\sqrt{\sigma_1^2\sigma_2^2}} \exp\left(-\frac{1}{2}\left(\begin{bmatrix} w \\ h \end{bmatrix} - \begin{bmatrix} \mu_1 \\ \mu_2 \end{bmatrix}\right)^\top \begin{bmatrix} \sigma_1^2 & 0 \\ 0 & \sigma_2^2 \end{bmatrix}^{-1} \left(\begin{bmatrix} w \\ h \end{bmatrix} - \begin{bmatrix} \mu_1 \\ \mu_2 \end{bmatrix}\right)\right)$$

Independent Gaussians

$$p(\mathbf{y}) = \frac{1}{2\pi\,|\mathbf{D}|^{\frac{1}{2}}} \exp\left(-\frac{1}{2}(\mathbf{y}-\boldsymbol{\mu})^\top \mathbf{D}^{-1}(\mathbf{y}-\boldsymbol{\mu})\right)$$

Correlated Gaussian

Form correlated from original by rotating the data space using matrix $\mathbf{R}$.

$$p(\mathbf{y}) = \frac{1}{2\pi\,|\mathbf{D}|^{\frac{1}{2}}} \exp\left(-\frac{1}{2}(\mathbf{y}-\boldsymbol{\mu})^\top \mathbf{D}^{-1}(\mathbf{y}-\boldsymbol{\mu})\right)$$

Correlated Gaussian

Form correlated from original by rotating the data space using matrix $\mathbf{R}$.

$$p(\mathbf{y}) = \frac{1}{2\pi\,|\mathbf{D}|^{\frac{1}{2}}} \exp\left(-\frac{1}{2}(\mathbf{R}^\top\mathbf{y} - \mathbf{R}^\top\boldsymbol{\mu})^\top\mathbf{D}^{-1}(\mathbf{R}^\top\mathbf{y} - \mathbf{R}^\top\boldsymbol{\mu})\right)$$

Correlated Gaussian

Form correlated from original by rotating the data space using matrix $\mathbf{R}$.

$$p(\mathbf{y}) = \frac{1}{2\pi\,|\mathbf{D}|^{\frac{1}{2}}} \exp\left(-\frac{1}{2}(\mathbf{y}-\boldsymbol{\mu})^\top \mathbf{R}\mathbf{D}^{-1}\mathbf{R}^\top(\mathbf{y}-\boldsymbol{\mu})\right)$$

this gives a covariance matrix:

$$\mathbf{C}^{-1} = \mathbf{R}\mathbf{D}^{-1}\mathbf{R}^\top$$

Correlated Gaussian

Form correlated from original by rotating the data space using matrix $\mathbf{R}$.

$$p(\mathbf{y}) = \frac{1}{2\pi\,|\mathbf{C}|^{\frac{1}{2}}} \exp\left(-\frac{1}{2}(\mathbf{y}-\boldsymbol{\mu})^\top \mathbf{C}^{-1}(\mathbf{y}-\boldsymbol{\mu})\right)$$

this gives a covariance matrix:

$$\mathbf{C} = \mathbf{R}\mathbf{D}\mathbf{R}^\top$$

Recall Univariate Gaussian Properties

1. Sum of Gaussian variables is also Gaussian.

$$y_i \sim \mathcal{N}\left(\mu_i, \sigma_i^2\right)$$

Recall Univariate Gaussian Properties

1. Sum of Gaussian variables is also Gaussian.

$$y_i \sim \mathcal{N}\left(\mu_i, \sigma_i^2\right)$$

$$\sum_{i=1}^{n} y_i \sim \mathcal{N}\left(\sum_{i=1}^{n} \mu_i, \sum_{i=1}^{n} \sigma_i^2\right)$$

Recall Univariate Gaussian Properties

1. Sum of Gaussian variables is also Gaussian.

$$y_i \sim \mathcal{N}\left(\mu_i, \sigma_i^2\right)$$

$$\sum_{i=1}^{n} y_i \sim \mathcal{N}\left(\sum_{i=1}^{n} \mu_i, \sum_{i=1}^{n} \sigma_i^2\right)$$

2. Scaling a Gaussian leads to a Gaussian.

Recall Univariate Gaussian Properties

1. Sum of Gaussian variables is also Gaussian.

$$y_i \sim \mathcal{N}\left(\mu_i, \sigma_i^2\right)$$

$$\sum_{i=1}^{n} y_i \sim \mathcal{N}\left(\sum_{i=1}^{n} \mu_i, \sum_{i=1}^{n} \sigma_i^2\right)$$

2. Scaling a Gaussian leads to a Gaussian.

$$y \sim \mathcal{N}\left(\mu, \sigma^2\right)$$

Recall Univariate Gaussian Properties

1. Sum of Gaussian variables is also Gaussian.

$$y_i \sim \mathcal{N}\left(\mu_i, \sigma_i^2\right)$$

$$\sum_{i=1}^{n} y_i \sim \mathcal{N}\left(\sum_{i=1}^{n} \mu_i, \sum_{i=1}^{n} \sigma_i^2\right)$$

2. Scaling a Gaussian leads to a Gaussian.

$$y \sim \mathcal{N}\left(\mu, \sigma^2\right)$$

$$wy \sim \mathcal{N}\left(w\mu, w^2\sigma^2\right)$$

Multivariate Consequence

- If

$$\mathbf{x} \sim \mathcal{N}\left(\mu, \mathbf{\Sigma}\right)$$

Multivariate Consequence

- If

$$\mathbf{x} \sim \mathcal{N}\left(\mu, \mathbf{\Sigma}\right)$$

- And

$$\mathbf{y} = \mathbf{W}\mathbf{x}$$

Multivariate Consequence

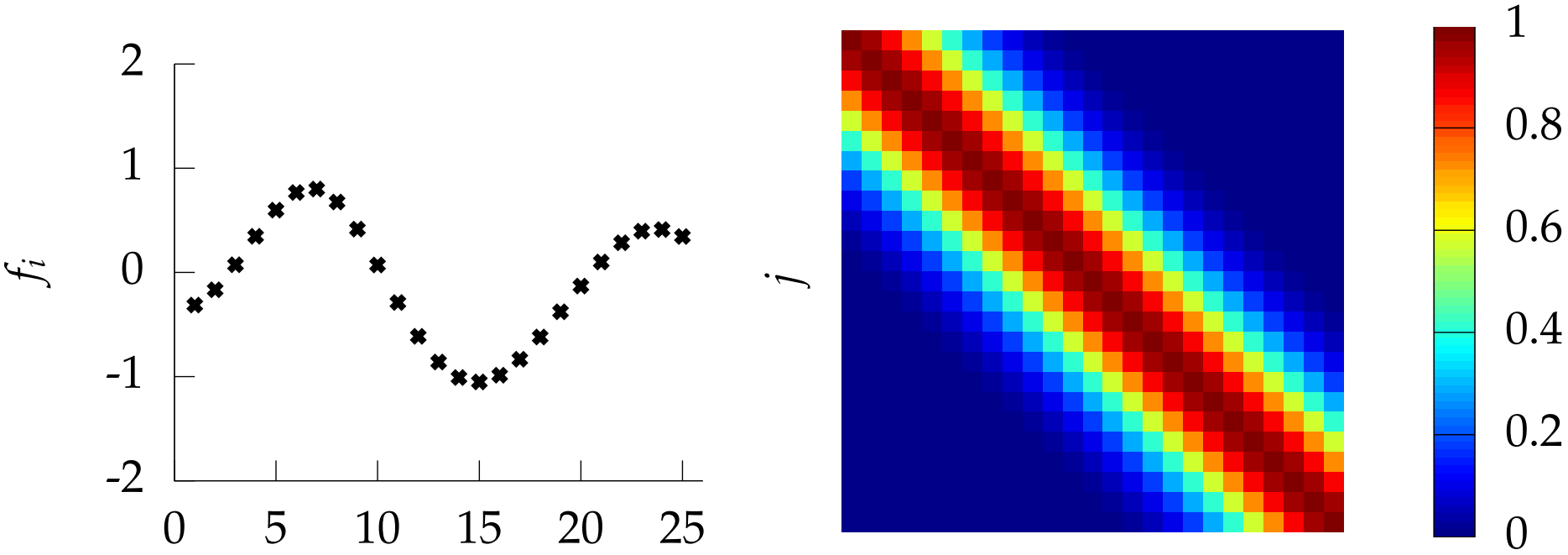
- If
$$\mathbf{x} \sim \mathcal{N}\left(\mu, \mathbf{\Sigma}\right)$$
- And
$$\mathbf{y} = \mathbf{W}\mathbf{x}$$
- Then
$$\mathbf{y} \sim \mathcal{N}\left(\mathbf{W}\mu, \mathbf{W}\mathbf{\Sigma}\mathbf{W}^\top\right)$$

Sampling a Function

Multi-variate Gaussians

- We will consider a Gaussian with a particular structure of covariance matrix.
- Generate a single sample from this 25 dimensional Gaussian distribution, $\mathbf{f} = [f_1, f_2 \dots f_{25}]$.
- We will plot these points against their index.

Gaussian Distribution Sample

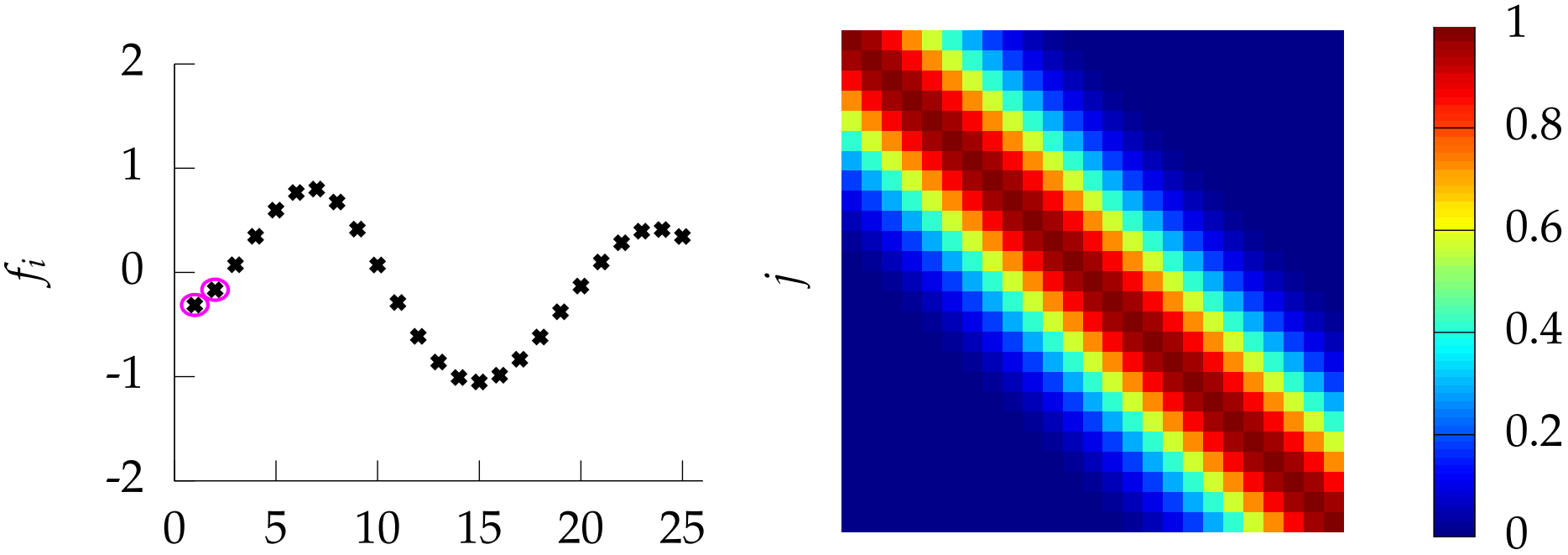


(a) A 25 dimensional correlated random variable (values ploted against index)

(b) colormap showing correlations between dimensions.

Figure : A sample from a 25 dimensional Gaussian distribution.

Gaussian Distribution Sample

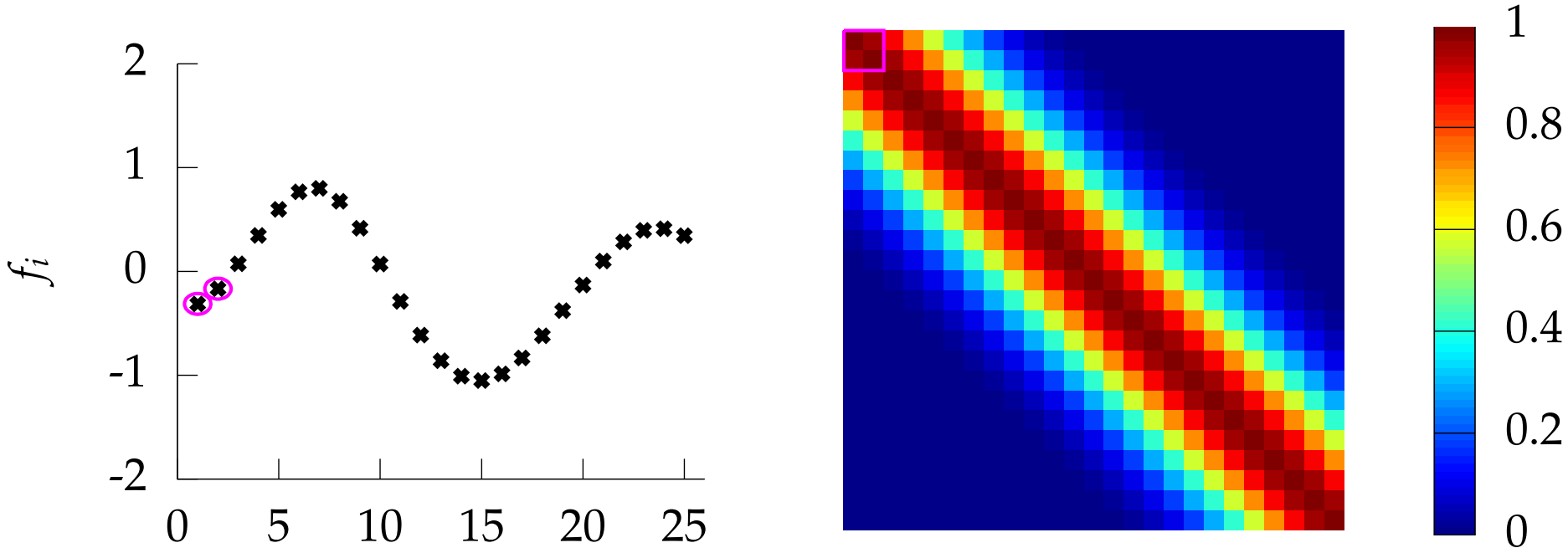


(a) A 25 dimensional correlated random variable (values ploted against index)

(b) colormap showing correlations between dimensions.

Figure : A sample from a 25 dimensional Gaussian distribution.

Gaussian Distribution Sample

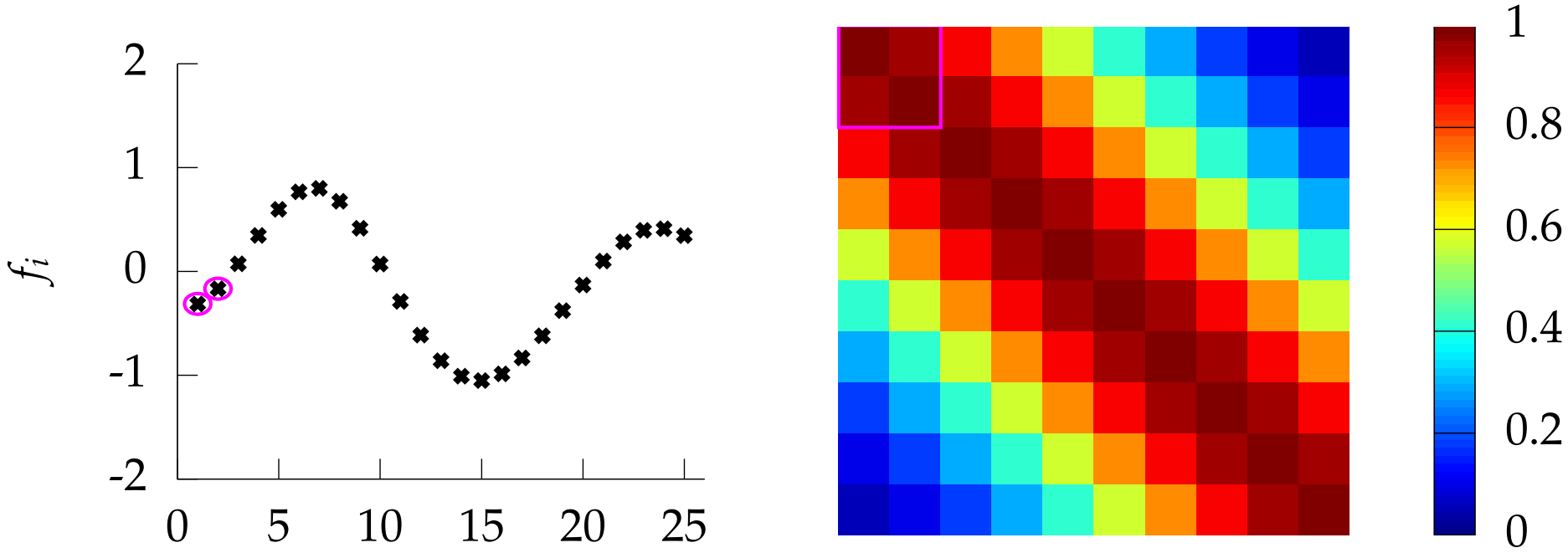


(a) A 25 dimensional correlated random variable (values ploted against index)

(b) colormap showing correlations between dimensions.

Figure : A sample from a 25 dimensional Gaussian distribution.

Gaussian Distribution Sample

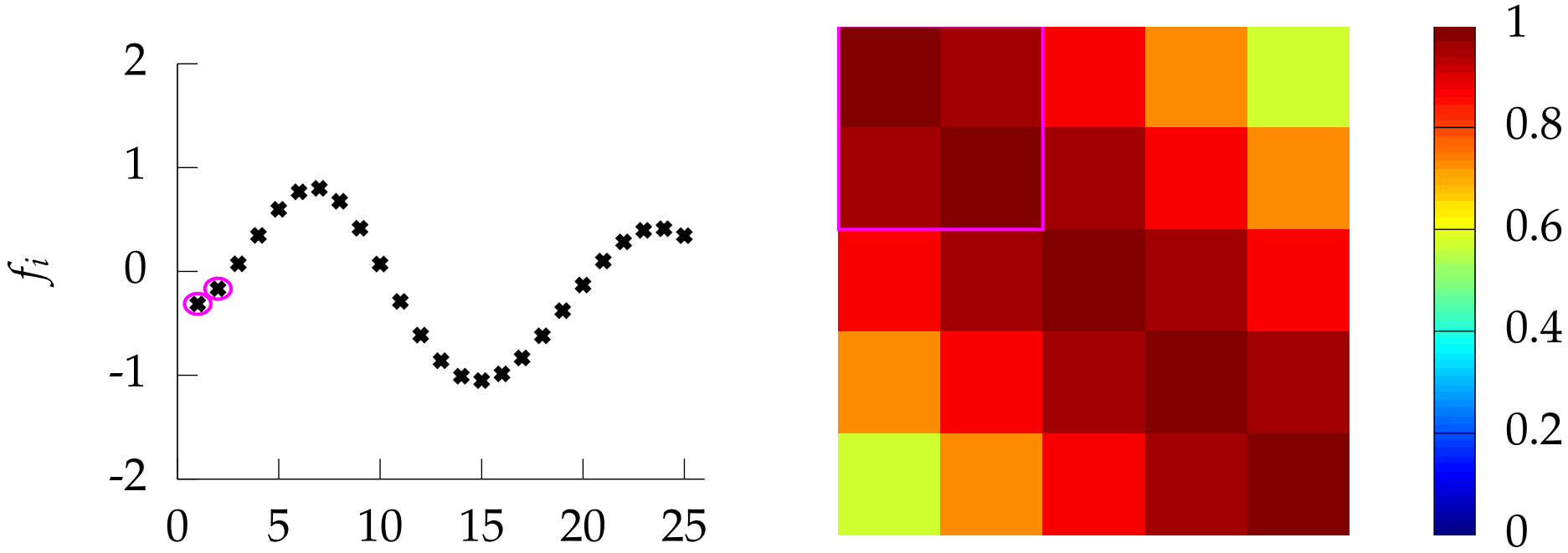


(a) A 25 dimensional correlated random variable (values ploted against index)

(b) colormap showing correlations between dimensions.

Figure : A sample from a 25 dimensional Gaussian distribution.

Gaussian Distribution Sample

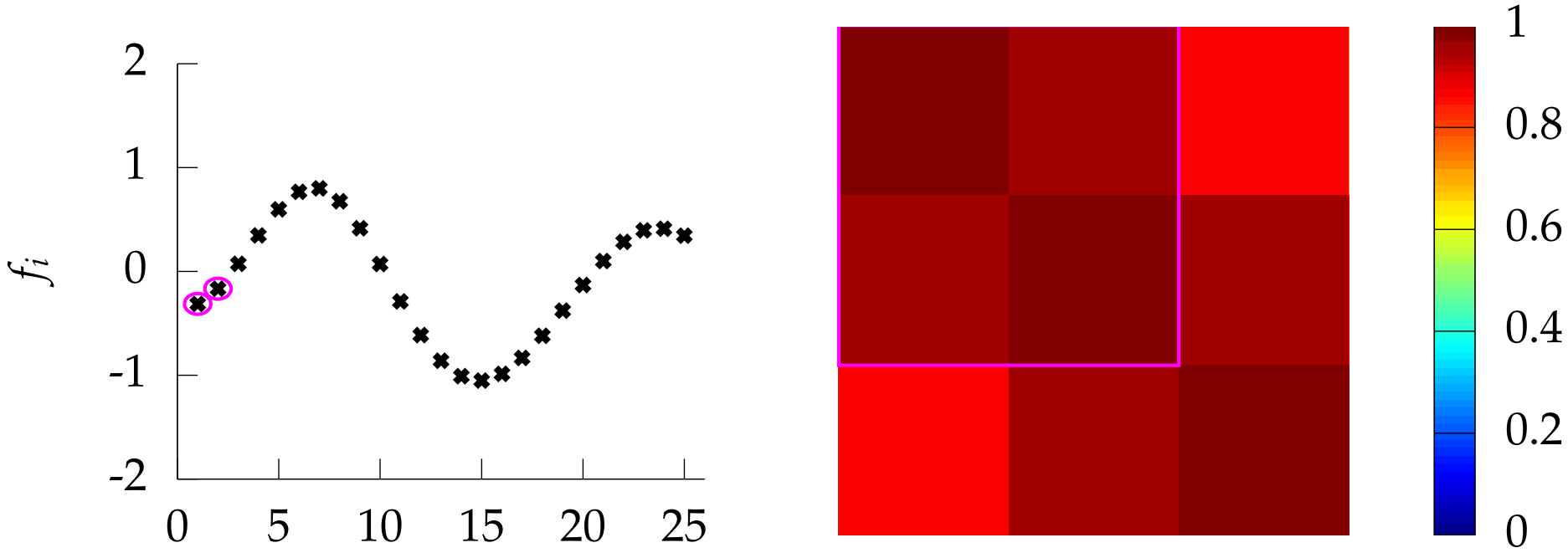


(a) A 25 dimensional correlated random variable (values ploted against index)

(b) colormap showing correlations between dimensions.

Figure : A sample from a 25 dimensional Gaussian distribution.

Gaussian Distribution Sample

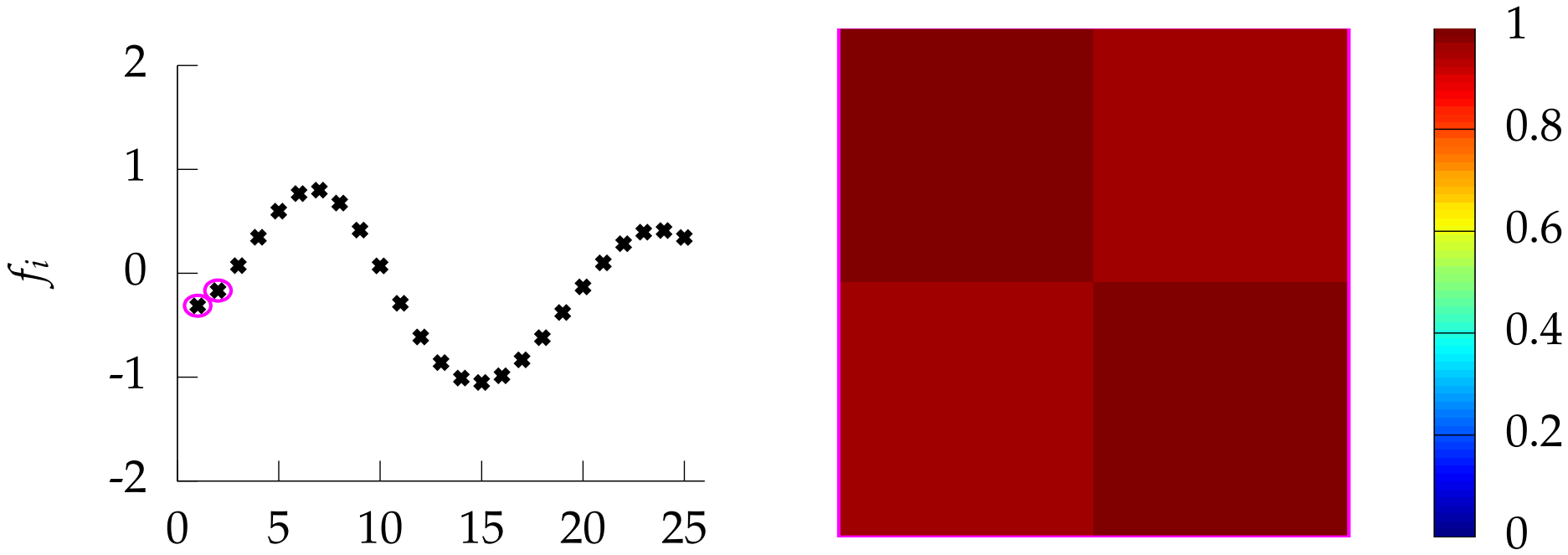


(a) A 25 dimensional correlated random variable (values ploted against index)

(b) colormap showing correlations between dimensions.

Figure : A sample from a 25 dimensional Gaussian distribution.

Gaussian Distribution Sample

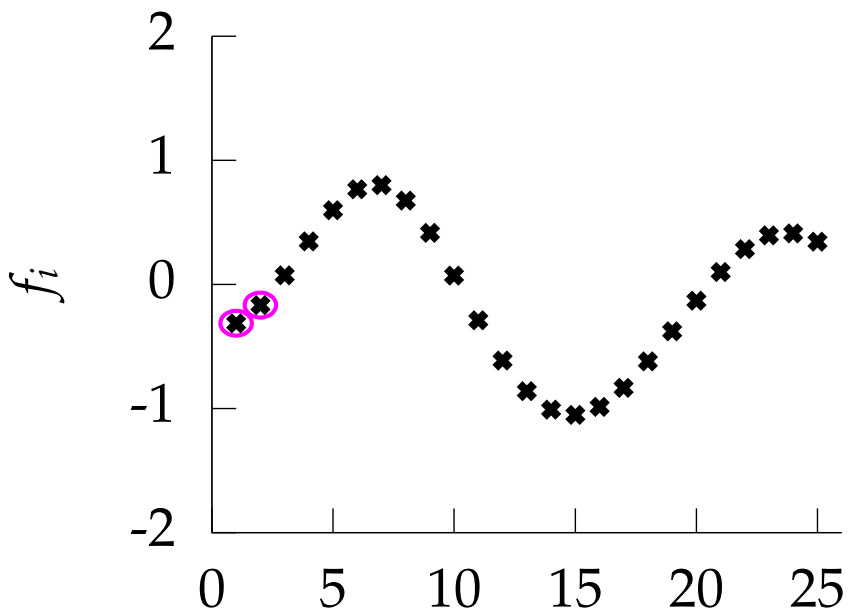


(a) A 25 dimensional correlated random variable (values ploted against index)

(b) colormap showing correlations between dimensions.

Figure : A sample from a 25 dimensional Gaussian distribution.

Gaussian Distribution Sample



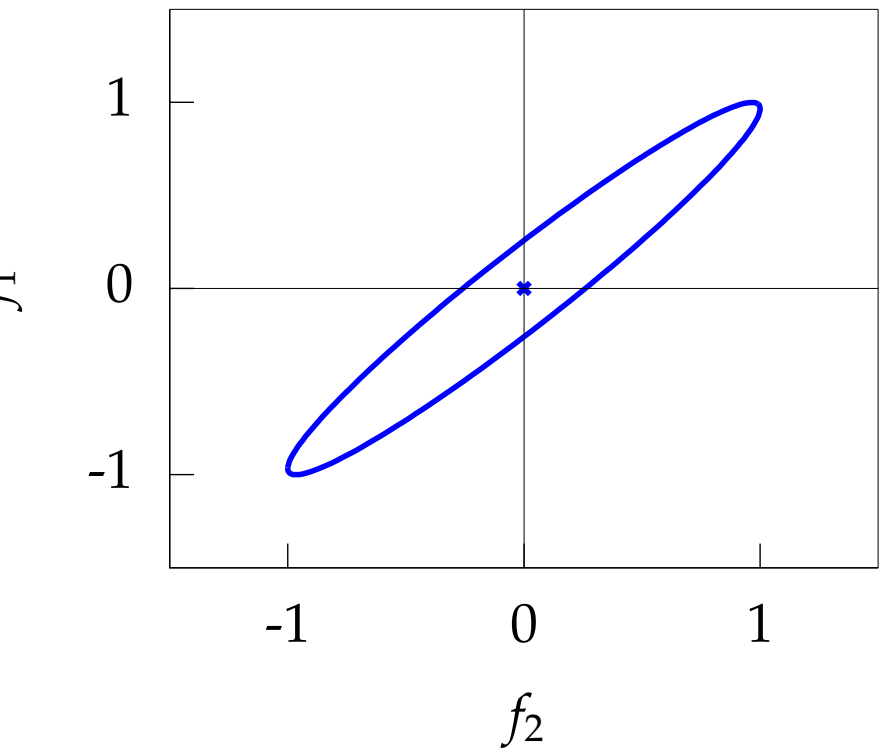
(a) A 25 dimensional correlated random variable (values ploted against index)

$$\begin{bmatrix} 1 & 0.96587 \\ 0.96587 & 1 \end{bmatrix}$$

(b) correlation between f_1 and f_2.

Figure : A sample from a 25 dimensional Gaussian distribution.

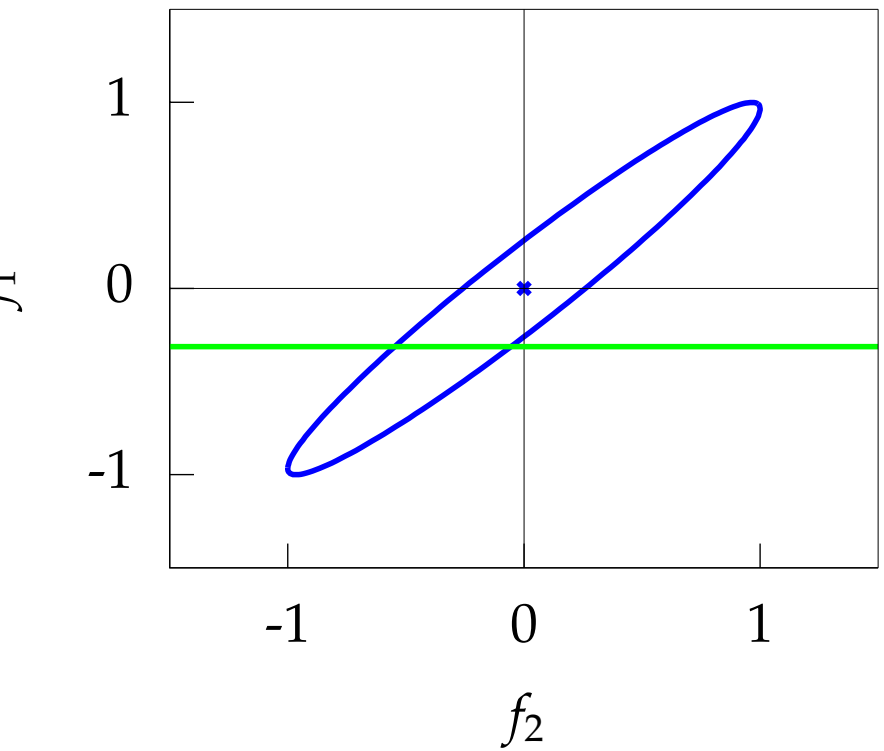
Prediction of f_2 from f_1



$$\begin{bmatrix} 1 & 0.96587 \\ 0.96587 & 1 \end{bmatrix}$$

- The single contour of the Gaussian density represents the joint distribution, $p(f_1, f_2)$.

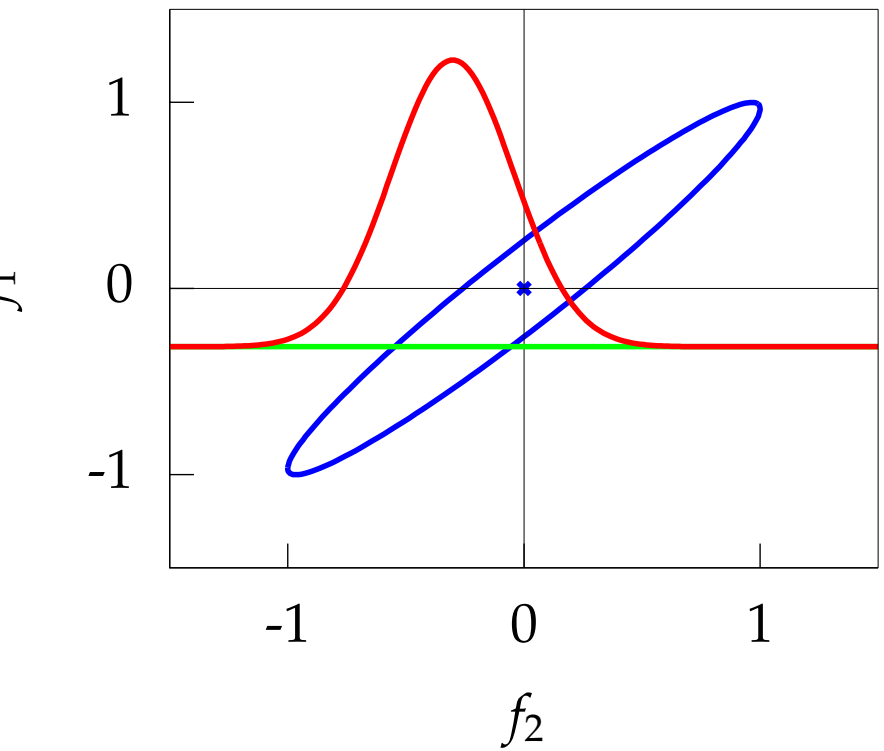
Prediction of f_2 from f_1



$$\begin{bmatrix} 1 & 0.96587 \\ 0.96587 & 1 \end{bmatrix}$$

- The single contour of the Gaussian density represents the joint distribution, $p(f_1, f_2)$.
- We observe that $f_1 = -0.313$.

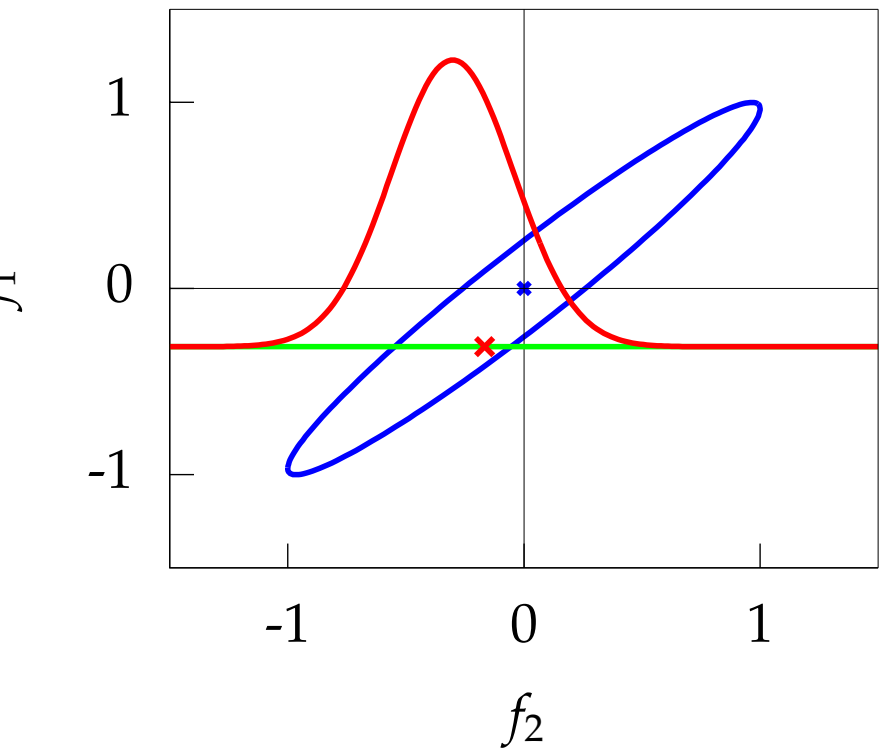
Prediction of f_2 from f_1



$$\begin{bmatrix} 1 & 0.96587 \\ 0.96587 & 1 \end{bmatrix}$$

- The single contour of the Gaussian density represents the joint distribution, $p(f_1, f_2)$.
- We observe that $f_1 = -0.313$.
- Conditional density: $p(f_2|f_1 = -0.313)$.

Prediction of f_2 from f_1



$$\begin{bmatrix} 1 & 0.96587 \\ 0.96587 & 1 \end{bmatrix}$$

- The single contour of the Gaussian density represents the joint distribution, $p(f_1, f_2)$.
- We observe that $f_1 = -0.313$.
- Conditional density: $p(f_2|f_1 = -0.313)$.

Prediction with Correlated Gaussians

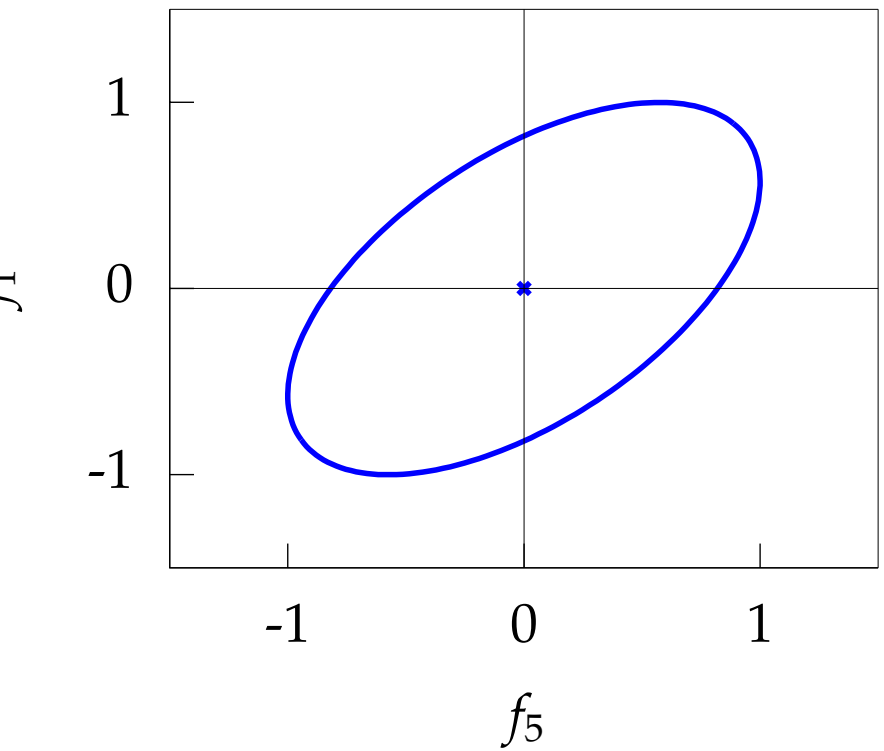
- Prediction of f_2 from f_1 requires *conditional density*.
- Conditional density is *also* Gaussian.

$$p(f_2|f_1) = \mathcal{N}\left(f_2|\frac{k_{1,2}}{k_{1,1}}f_1, k_{2,2} - \frac{k_{1,2}^2}{k_{1,1}}\right)$$

where covariance of joint density is given by

$$\mathbf{K} = \begin{bmatrix} k_{1,1} & k_{1,2} \\ k_{2,1} & k_{2,2} \end{bmatrix}$$

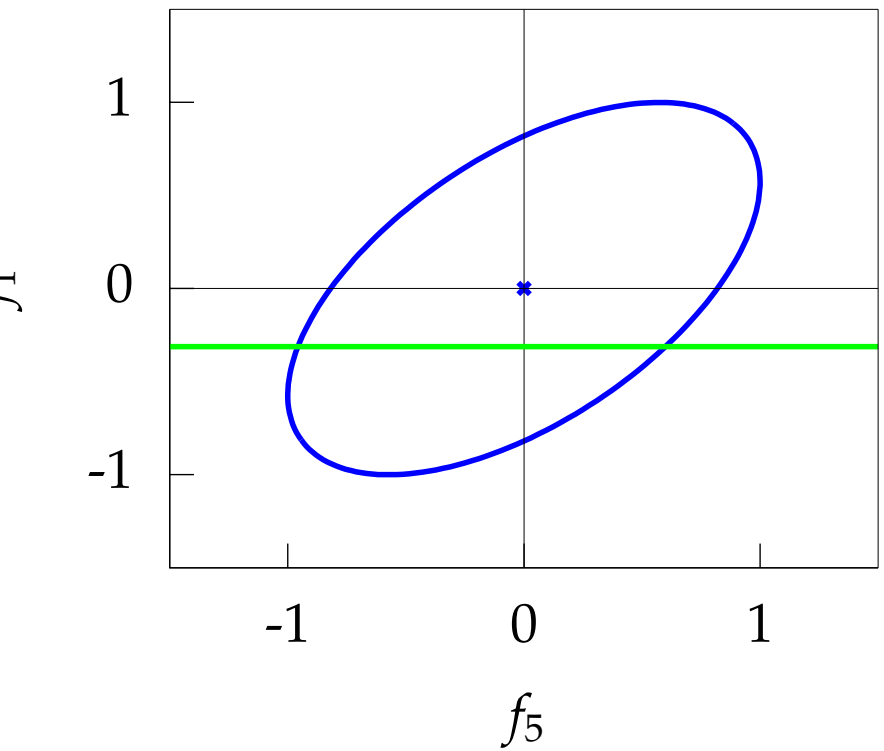
Prediction of f_5 from f_1



$$\begin{bmatrix} 1 & 0.57375 \\ 0.57375 & 1 \end{bmatrix}$$

- The single contour of the Gaussian density represents the joint distribution, $p(f_1, f_5)$.

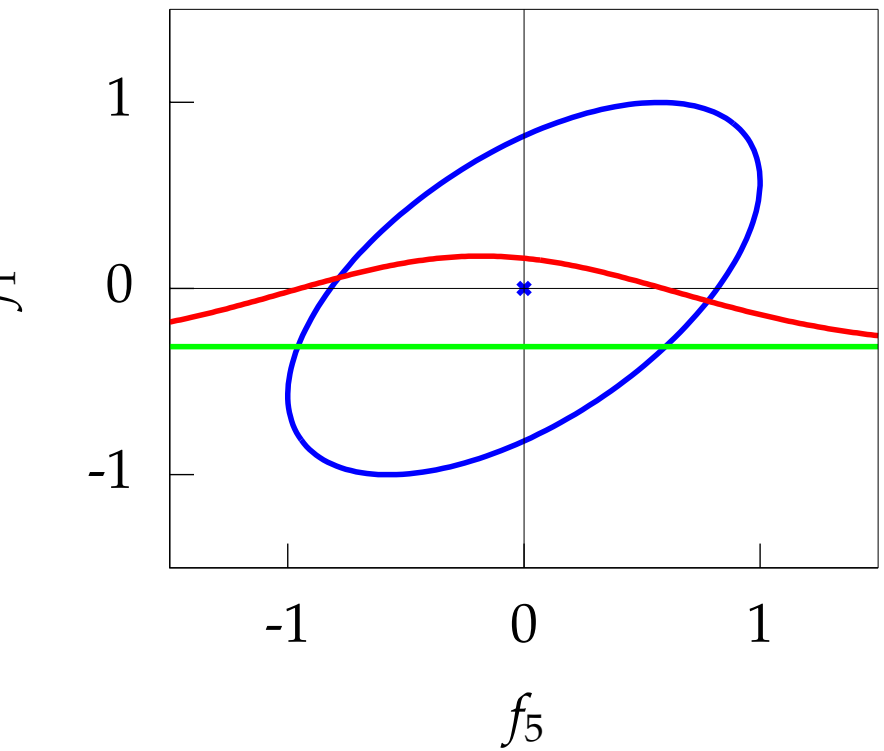
Prediction of f_5 from f_1



$$\begin{bmatrix} 1 & 0.57375 \\ 0.57375 & 1 \end{bmatrix}$$

- The single contour of the Gaussian density represents the joint distribution, $p(f_1, f_5)$.
- We observe that $f_1 = -0.313$.

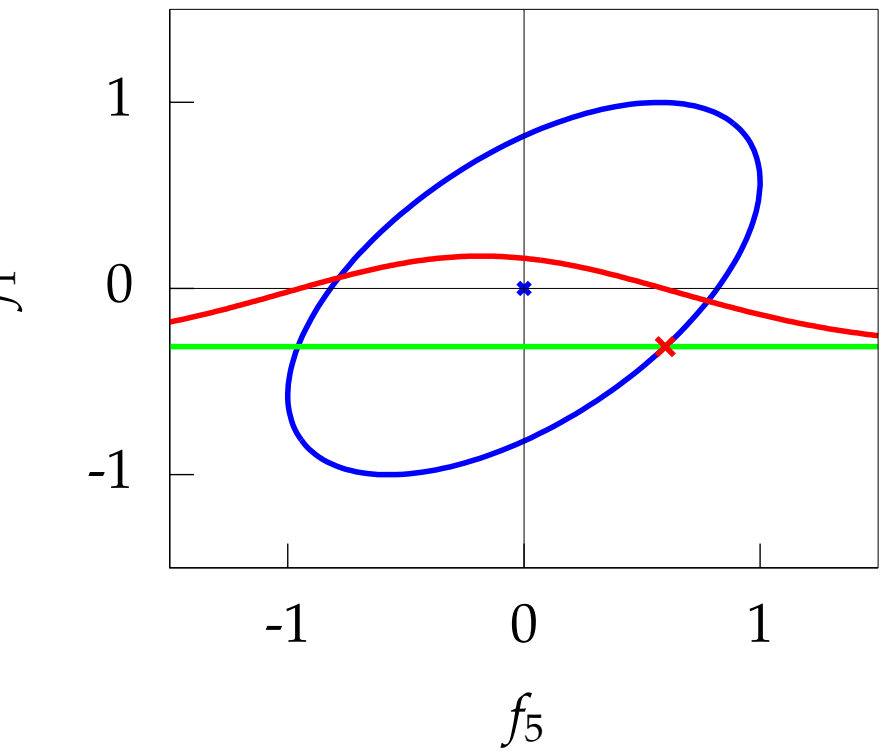
Prediction of f_5 from f_1



$$\begin{bmatrix} 1 & 0.57375 \\ 0.57375 & 1 \end{bmatrix}$$

- The single contour of the Gaussian density represents the joint distribution, $p(f_1, f_5)$.
- We observe that $f_1 = -0.313$.
- Conditional density: $p(f_5|f_1 = -0.313)$.

Prediction of f_5 from f_1



$$\begin{bmatrix} 1 & 0.57375 \\ 0.57375 & 1 \end{bmatrix}$$

- The single contour of the Gaussian density represents the joint distribution, $p(f_1, f_5)$.
- We observe that $f_1 = -0.313$.
- Conditional density: $p(f_5|f_1 = -0.313)$.

Prediction with Correlated Gaussians

- Prediction of $\mathbf{f}_*$ from $\mathbf{f}$ requires multivariate *conditional density*.
- Multivariate conditional density is *also* Gaussian.

$$p(\mathbf{f}_*|\mathbf{f}) = \mathcal{N}\left(\mathbf{f}_*|\mathbf{K}_{*,\mathbf{f}}\mathbf{K}_{\mathbf{f},\mathbf{f}}^{-1}\mathbf{f}, \mathbf{K}_{*,*} - \mathbf{K}_{*,\mathbf{f}}\mathbf{K}_{\mathbf{f},\mathbf{f}}^{-1}\mathbf{K}_{\mathbf{f},*}\right)$$

- Here covariance of joint density is given by

$$\mathbf{K} = \begin{bmatrix} \mathbf{K}_{\mathbf{f},\mathbf{f}} & \mathbf{K}_{*,\mathbf{f}} \\ \mathbf{K}_{\mathbf{f},*} & \mathbf{K}_{*,*} \end{bmatrix}$$

Prediction with Correlated Gaussians

- Prediction of $\mathbf{f}_*$ from $\mathbf{f}$ requires multivariate *conditional density*.
- Multivariate conditional density is *also* Gaussian.

$$p(\mathbf{f}_*|\mathbf{f}) = \mathcal{N}\left(\mathbf{f}_*|\boldsymbol{\mu}, \boldsymbol{\Sigma}\right)$$

$$\boldsymbol{\mu} = \mathbf{K}_{*,\mathbf{f}}\mathbf{K}_{\mathbf{f},\mathbf{f}}^{-1}\mathbf{f}$$

$$\boldsymbol{\Sigma} = \mathbf{K}_{*,*} - \mathbf{K}_{*,\mathbf{f}}\mathbf{K}_{\mathbf{f},\mathbf{f}}^{-1}\mathbf{K}_{\mathbf{f},*}$$

- Here covariance of joint density is given by

$$\mathbf{K} = \begin{bmatrix} \mathbf{K}_{\mathbf{f},\mathbf{f}} & \mathbf{K}_{*,\mathbf{f}} \\ \mathbf{K}_{\mathbf{f},*} & \mathbf{K}_{*,*} \end{bmatrix}$$

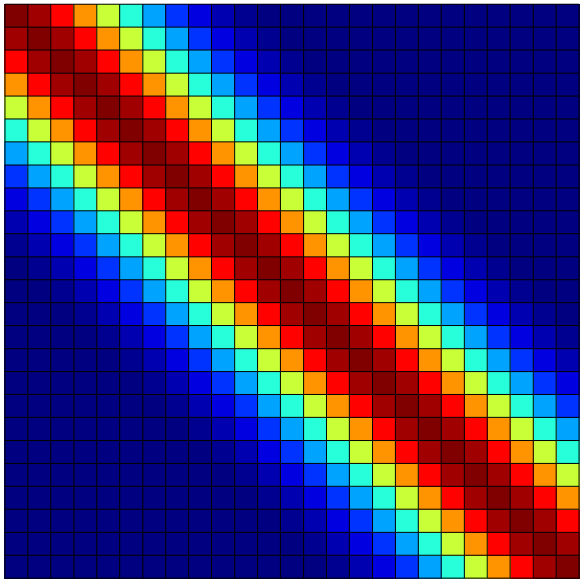
Covariance Functions

Where did this covariance matrix come from?

Exponentiated Quadratic Kernel Function (RBF, Squared Exponential, Gaussian)

$$k(\mathbf{x}, \mathbf{x}') = \alpha \exp\left(-\frac{\|\mathbf{x} - \mathbf{x}'\|_2^2}{2\ell^2}\right)$$

- Covariance matrix is built using the *inputs* to the function $\mathbf{x}$.
- For the example above it was based on Euclidean distance.
- The covariance function is also know as a kernel.



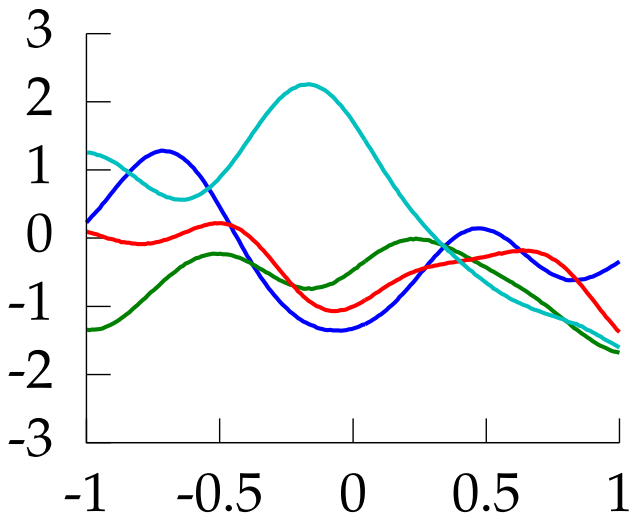
Covariance Functions

Where did this covariance matrix come from?

Exponentiated Quadratic Kernel Function (RBF, Squared Exponential, Gaussian)

$$k(\mathbf{x}, \mathbf{x}') = \alpha \exp\left(-\frac{\|\mathbf{x} - \mathbf{x}'\|_2^2}{2\ell^2}\right)$$

- Covariance matrix is built using the *inputs* to the function $\mathbf{x}$.
- For the example above it was based on Euclidean distance.
- The covariance function is also know as a kernel.



Covariance Functions

Where did this covariance matrix come from?

$$k\left(x_i, x_j\right) = \alpha \exp\left(-\frac{\left\|x_i - x_j\right\|^2}{2\ell^2}\right)$$

$$x_1 = -3.0, x_1 = -3.0$$

$$k_{1,1} = 1.00 \times \exp\left(-\frac{(-3.0 - -3.0)^2}{2 \times 2.00^2}\right)$$

$x_1 = -3.0$, $x_2 = 1.20$, and $x_3 = 1.40$ with $\ell = 2.00$ and $\alpha = 1.00$.

Covariance Functions

Where did this covariance matrix come from?

$$k\left(x_i, x_j\right) = \alpha \exp\left(-\frac{\left\|x_i - x_j\right\|^2}{2\ell^2}\right)$$

$$x_1 = -3.0, x_1 = -3.0$$

$$k_{1,1} = 1.00 \times \exp\left(-\frac{(-3.0 - -3.0)^2}{2 \times 2.00^2}\right)$$

$$\begin{bmatrix} \boxed{1.00} & & \\ & & \\ & & \end{bmatrix}$$

$x_1 = -3.0$, $x_2 = 1.20$, and $x_3 = 1.40$ with $\ell = 2.00$ and $\alpha = 1.00$.

Covariance Functions

Where did this covariance matrix come from?

$$k\left(x_i, x_j\right) = \alpha \exp\left(-\frac{\left\|x_i - x_j\right\|^2}{2\ell^2}\right)$$

$$x_2 = 1.20,\ x_1 = -3.0$$

$$k_{2,1} = 1.00 \times \exp\left(-\frac{(1.20 - -3.0)^2}{2 \times 2.00^2}\right)$$

$$\begin{bmatrix} 1.00 & \\ & \\ & \end{bmatrix}$$

$x_1 = -3.0$, $x_2 = 1.20$, and $x_3 = 1.40$ with $\ell = 2.00$ and $\alpha = 1.00$.

Covariance Functions

Where did this covariance matrix come from?

$$k\left(x_i, x_j\right) = \alpha \exp\left(-\frac{\left\|x_i - x_j\right\|^2}{2\ell^2}\right)$$

$$x_2 = 1.20,\ x_1 = -3.0$$

$$k_{2,1} = 1.00 \times \exp\left(-\frac{(1.20 - -3.0)^2}{2 \times 2.00^2}\right)$$

$$\begin{bmatrix} 1.00 & & \\ \boxed{0.110} & & \\ & & \end{bmatrix}$$

$x_1 = -3.0$, $x_2 = 1.20$, and $x_3 = 1.40$ with $\ell = 2.00$ and $\alpha = 1.00$.

Covariance Functions

Where did this covariance matrix come from?

$$k\left(x_i, x_j\right) = \alpha \exp\left(-\frac{\left\|x_i - x_j\right\|^2}{2\ell^2}\right)$$

$$x_2 = 1.20,\ x_1 = -3.0$$

$$k_{2,1} = 1.00 \times \exp\left(-\frac{(1.20 - -3.0)^2}{2 \times 2.00^2}\right)$$

$$\begin{bmatrix} 1.00 & 0.110 \\ 0.110 & \\ & \end{bmatrix}$$

$x_1 = -3.0$, $x_2 = 1.20$, and $x_3 = 1.40$ with $\ell = 2.00$ and $\alpha = 1.00$.

Covariance Functions

Where did this covariance matrix come from?

$$k\left(x_i, x_j\right) = \alpha \exp\left(-\frac{\left\|x_i - x_j\right\|^2}{2\ell^2}\right)$$

$$x_2 = 1.20,\ x_2 = 1.20$$

$$k_{2,2} = 1.00 \times \exp\left(-\frac{(1.20-1.20)^2}{2\times 2.00^2}\right)$$

$$\begin{bmatrix} 1.00 & 0.110 \\ 0.110 & \\ & \end{bmatrix}$$

$x_1 = -3.0$, $x_2 = 1.20$, and $x_3 = 1.40$ with $\ell = 2.00$ and $\alpha = 1.00$.

Covariance Functions

Where did this covariance matrix come from?

$$k\left(x_i, x_j\right) = \alpha \exp\left(-\frac{\left\|x_i - x_j\right\|^2}{2\ell^2}\right)$$

$$x_2 = 1.20,\ x_2 = 1.20$$

$$k_{2,2} = 1.00 \times \exp\left(-\frac{(1.20-1.20)^2}{2\times 2.00^2}\right)$$

$$\begin{bmatrix} 1.00 & 0.110 & \\ 0.110 & \boxed{1.00} & \\ & & \end{bmatrix}$$

$x_1 = -3.0$, $x_2 = 1.20$, and $x_3 = 1.40$ with $\ell = 2.00$ and $\alpha = 1.00$.

Covariance Functions

Where did this covariance matrix come from?

$$k\left(x_i, x_j\right) = \alpha \exp\left(-\frac{\left\|x_i - x_j\right\|^2}{2\ell^2}\right)$$

$$x_3 = 1.40,\ x_1 = -3.0$$

$$k_{3,1} = 1.00 \times \exp\left(-\frac{(1.40 - -3.0)^2}{2 \times 2.00^2}\right)$$

$$\begin{bmatrix} 1.00 & 0.110 \\ 0.110 & 1.00 \end{bmatrix}$$

$x_1 = -3.0$, $x_2 = 1.20$, and $x_3 = 1.40$ with $\ell = 2.00$ and $\alpha = 1.00$.

Covariance Functions

Where did this covariance matrix come from?

$$k\left(x_i, x_j\right) = \alpha \exp\left(-\frac{\left\|x_i - x_j\right\|^2}{2\ell^2}\right)$$

$$x_3 = 1.40,\ x_1 = -3.0$$

$$k_{3,1} = 1.00 \times \exp\left(-\frac{(1.40 - -3.0)^2}{2 \times 2.00^2}\right)$$

$$\begin{bmatrix} 1.00 & 0.110 & \\ 0.110 & 1.00 & \\ \boxed{0.0889} & & \end{bmatrix}$$

$x_1 = -3.0$, $x_2 = 1.20$, and $x_3 = 1.40$ with $\ell = 2.00$ and $\alpha = 1.00$.

Covariance Functions

Where did this covariance matrix come from?

$$k\left(x_i, x_j\right) = \alpha \exp\left(-\frac{\left\|x_i - x_j\right\|^2}{2\ell^2}\right)$$

$$x_3 = 1.40,\ x_1 = -3.0$$

$$k_{3,1} = 1.00 \times \exp\left(-\frac{(1.40 - -3.0)^2}{2 \times 2.00^2}\right)$$

$$\begin{bmatrix} 1.00 & 0.110 & 0.0889 \\ 0.110 & 1.00 & \\ 0.0889 & & \end{bmatrix}$$

$x_1 = -3.0$, $x_2 = 1.20$, and $x_3 = 1.40$ with $\ell = 2.00$ and $\alpha = 1.00$.

Covariance Functions

Where did this covariance matrix come from?

$$k\left(x_i, x_j\right) = \alpha \exp\left(-\frac{\left\|x_i - x_j\right\|^2}{2\ell^2}\right)$$

$$x_3 = 1.40,\ x_2 = 1.20$$

$$k_{3,2} = 1.00 \times \exp\left(-\frac{(1.40-1.20)^2}{2\times 2.00^2}\right)$$

$$\begin{bmatrix} 1.00 & 0.110 & 0.0889 \\ 0.110 & 1.00 & \\ 0.0889 & & \end{bmatrix}$$

$x_1 = -3.0$, $x_2 = 1.20$, and $x_3 = 1.40$ with $\ell = 2.00$ and $\alpha = 1.00$.

Covariance Functions

Where did this covariance matrix come from?

$$k\left(x_i, x_j\right) = \alpha \exp\left(-\frac{\left\|x_i - x_j\right\|^2}{2\ell^2}\right)$$

$$x_3 = 1.40,\ x_2 = 1.20$$

$$k_{3,2} = 1.00 \times \exp\left(-\frac{(1.40-1.20)^2}{2\times 2.00^2}\right)$$

$$\begin{bmatrix} 1.00 & 0.110 & 0.0889 \\ 0.110 & 1.00 & \\ 0.0889 & \boxed{0.995} & \end{bmatrix}$$

$x_1 = -3.0$, $x_2 = 1.20$, and $x_3 = 1.40$ with $\ell = 2.00$ and $\alpha = 1.00$.

Covariance Functions

Where did this covariance matrix come from?

$$k\left(x_i, x_j\right) = \alpha \exp\left(-\frac{\left\|x_i - x_j\right\|^2}{2\ell^2}\right)$$

$$x_3 = 1.40,\ x_2 = 1.20$$

$$k_{3,2} = 1.00 \times \exp\left(-\frac{(1.40-1.20)^2}{2\times 2.00^2}\right)$$

$$\begin{bmatrix} 1.00 & 0.110 & 0.0889 \\ 0.110 & 1.00 & 0.995 \\ 0.0889 & 0.995 & \end{bmatrix}$$

$x_1 = -3.0$, $x_2 = 1.20$, and $x_3 = 1.40$ with $\ell = 2.00$ and $\alpha = 1.00$.

Covariance Functions

Where did this covariance matrix come from?

$$k\left(x_i, x_j\right) = \alpha \exp\left(-\frac{\left\|x_i - x_j\right\|^2}{2\ell^2}\right)$$

$$x_3 = 1.40,\ x_3 = 1.40$$

$$k_{3,3} = 1.00 \times \exp\left(-\frac{(1.40-1.40)^2}{2\times 2.00^2}\right)$$

$$\begin{bmatrix} 1.00 & 0.110 & 0.0889 \\ 0.110 & 1.00 & 0.995 \\ 0.0889 & 0.995 & \end{bmatrix}$$

$x_1 = -3.0$, $x_2 = 1.20$, and $x_3 = 1.40$ with $\ell = 2.00$ and $\alpha = 1.00$.

Covariance Functions

Where did this covariance matrix come from?

$$k\left(x_i, x_j\right) = \alpha \exp\left(-\frac{\left\|x_i - x_j\right\|^2}{2\ell^2}\right)$$

$$x_3 = 1.40,\ x_3 = 1.40$$

$$k_{3,3} = 1.00 \times \exp\left(-\frac{(1.40-1.40)^2}{2\times 2.00^2}\right)$$

$$\begin{bmatrix} 1.00 & 0.110 & 0.0889 \\ 0.110 & 1.00 & 0.995 \\ 0.0889 & 0.995 & \boxed{1.00} \end{bmatrix}$$

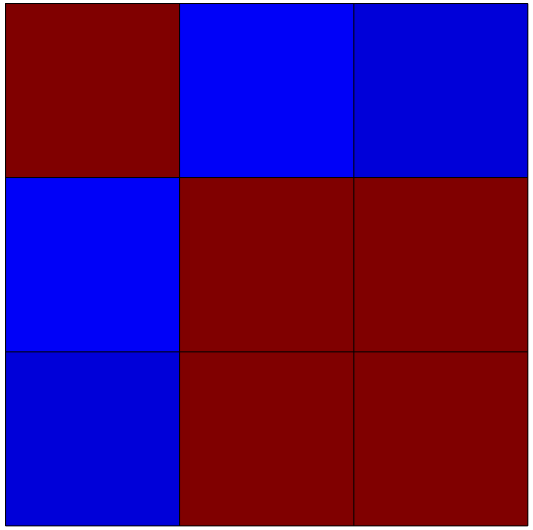
$x_1 = -3.0$, $x_2 = 1.20$, and $x_3 = 1.40$ with $\ell = 2.00$ and $\alpha = 1.00$.

Covariance Functions

Where did this covariance matrix come from?

$$k\left(x_i, x_j\right) = \alpha \exp\left(-\frac{\left\|x_i - x_j\right\|^2}{2\ell^2}\right)$$

$$x_3 = 1.40,\ x_3 = 1.40$$



$$k_{3,3} = 1.00 \times \exp\left(-\frac{(1.40-1.40)^2}{2\times 2.00^2}\right)$$

$x_1 = -3.0$, $x_2 = 1.20$, and $x_3 = 1.40$ with $\ell = 2.00$ and $\alpha = 1.00$.

Covariance Functions

Where did this covariance matrix come from?

$$k\left(x_i, x_j\right) = \alpha \exp\left(-\frac{\left\|x_i - x_j\right\|^2}{2\ell^2}\right)$$

$$x_1 = -3, x_1 = -3$$

$$k_{1,1} = 1.0 \times \exp\left(-\frac{(-3--3)^2}{2\times 2.0^2}\right)$$

$x_1 = -3$, $x_2 = 1.2$, $x_3 = 1.4$, and $x_4 = 2.0$ with $\ell = 2.0$ and $\alpha = 1.0$.

Covariance Functions

Where did this covariance matrix come from?

$$k\left(x_i, x_j\right) = \alpha \exp\left(-\frac{\left\|x_i - x_j\right\|^2}{2\ell^2}\right)$$

$$x_1 = -3, x_1 = -3$$

$$k_{1,1} = 1.0 \times \exp\left(-\frac{(-3--3)^2}{2\times 2.0^2}\right)$$

$$\begin{bmatrix} \boxed{1.0} & & \\ & & \\ & & \end{bmatrix}$$

$x_1 = -3$, $x_2 = 1.2$, $x_3 = 1.4$, and $x_4 = 2.0$ with $\ell = 2.0$ and $\alpha = 1.0$.

Covariance Functions

Where did this covariance matrix come from?

$$k\left(x_i, x_j\right) = \alpha \exp\left(-\frac{\left\|x_i - x_j\right\|^2}{2\ell^2}\right)$$

$$x_2 = 1.2,\ x_1 = -3$$

$$k_{2,1} = 1.0 \times \exp\left(-\frac{(1.2 - -3)^2}{2 \times 2.0^2}\right)$$

$$\begin{bmatrix} 1.0 & & \\ & & \\ & & \end{bmatrix}$$

$x_1 = -3$, $x_2 = 1.2$, $x_3 = 1.4$, and $x_4 = 2.0$ with $\ell = 2.0$ and $\alpha = 1.0$.

Covariance Functions

Where did this covariance matrix come from?

$$k\left(x_i, x_j\right) = \alpha \exp\left(-\frac{\left\|x_i - x_j\right\|^2}{2\ell^2}\right)$$

$$x_2 = 1.2,\ x_1 = -3$$

$$k_{2,1} = 1.0 \times \exp\left(-\frac{(1.2 - -3)^2}{2 \times 2.0^2}\right)$$

$$\begin{bmatrix} 1.0 & & \\ \boxed{0.11} & & \\ & & \\ & & \end{bmatrix}$$

$x_1 = -3$, $x_2 = 1.2$, $x_3 = 1.4$, and $x_4 = 2.0$ with $\ell = 2.0$ and $\alpha = 1.0$.

Covariance Functions

Where did this covariance matrix come from?

$$k\left(x_i, x_j\right) = \alpha \exp\left(-\frac{\left\|x_i - x_j\right\|^2}{2\ell^2}\right)$$

$$x_2 = 1.2,\ x_1 = -3$$

$$k_{2,1} = 1.0 \times \exp\left(-\frac{(1.2 - -3)^2}{2 \times 2.0^2}\right)$$

$$\begin{bmatrix} 1.0 & 0.11 & \\ 0.11 & & \\ & & \\ & & \end{bmatrix}$$

$x_1 = -3$, $x_2 = 1.2$, $x_3 = 1.4$, and $x_4 = 2.0$ with $\ell = 2.0$ and $\alpha = 1.0$.

Covariance Functions

Where did this covariance matrix come from?

$$k\left(x_i, x_j\right) = \alpha \exp\left(-\frac{\left\|x_i - x_j\right\|^2}{2\ell^2}\right)$$

$$x_2 = 1.2,\ x_2 = 1.2$$

$$k_{2,2} = 1.0 \times \exp\left(-\frac{(1.2-1.2)^2}{2\times 2.0^2}\right)$$

$$\begin{bmatrix} 1.0 & 0.11 & \\ 0.11 & & \\ & & \\ & & \end{bmatrix}$$

$x_1 = -3$, $x_2 = 1.2$, $x_3 = 1.4$, and $x_4 = 2.0$ with $\ell = 2.0$ and $\alpha = 1.0$.

Covariance Functions

Where did this covariance matrix come from?

$$k\left(x_i, x_j\right) = \alpha \exp\left(-\frac{\left\|x_i - x_j\right\|^2}{2\ell^2}\right)$$

$$x_2 = 1.2,\ x_2 = 1.2$$

$$k_{2,2} = 1.0 \times \exp\left(-\frac{(1.2-1.2)^2}{2\times 2.0^2}\right)$$

$$\begin{bmatrix} 1.0 & 0.11 & \\ 0.11 & \boxed{1.0} & \\ & & \\ & & \end{bmatrix}$$

$x_1 = -3$, $x_2 = 1.2$, $x_3 = 1.4$, and $x_4 = 2.0$ with $\ell = 2.0$ and $\alpha = 1.0$.

Covariance Functions

Where did this covariance matrix come from?

$$k\left(x_i, x_j\right) = \alpha \exp\left(-\frac{\left\|x_i - x_j\right\|^2}{2\ell^2}\right)$$

$$x_3 = 1.4,\ x_1 = -3$$

$$k_{3,1} = 1.0 \times \exp\left(-\frac{(1.4 - -3)^2}{2 \times 2.0^2}\right)$$

$$\begin{bmatrix} 1.0 & 0.11 \\ 0.11 & 1.0 \end{bmatrix}$$

$x_1 = -3$, $x_2 = 1.2$, $x_3 = 1.4$, and $x_4 = 2.0$ with $\ell = 2.0$ and $\alpha = 1.0$.

Covariance Functions

Where did this covariance matrix come from?

$$k\left(x_i, x_j\right) = \alpha \exp\left(-\frac{\left\|x_i - x_j\right\|^2}{2\ell^2}\right)$$

$$x_3 = 1.4,\ x_1 = -3$$

$$k_{3,1} = 1.0 \times \exp\left(-\frac{(1.4 - -3)^2}{2 \times 2.0^2}\right)$$

$$\begin{bmatrix} 1.0 & 0.11 & \\ 0.11 & 1.0 & \\ \boxed{0.089} & & \\ & & \end{bmatrix}$$

$x_1 = -3$, $x_2 = 1.2$, $x_3 = 1.4$, and $x_4 = 2.0$ with $\ell = 2.0$ and $\alpha = 1.0$.

Covariance Functions

Where did this covariance matrix come from?

$$k\left(x_i, x_j\right) = \alpha \exp\left(-\frac{\left\|x_i - x_j\right\|^2}{2\ell^2}\right)$$

$$x_3 = 1.4,\ x_1 = -3$$

$$k_{3,1} = 1.0 \times \exp\left(-\frac{(1.4 - -3)^2}{2 \times 2.0^2}\right)$$

$$\begin{bmatrix} 1.0 & 0.11 & 0.089 \\ 0.11 & 1.0 & \\ 0.089 & & \\ & & \end{bmatrix}$$

$x_1 = -3$, $x_2 = 1.2$, $x_3 = 1.4$, and $x_4 = 2.0$ with $\ell = 2.0$ and $\alpha = 1.0$.

Covariance Functions

Where did this covariance matrix come from?

$$k\left(x_i, x_j\right) = \alpha \exp\left(-\frac{\left\|x_i - x_j\right\|^2}{2\ell^2}\right)$$

$$x_3 = 1.4,\ x_2 = 1.2$$

$$k_{3,2} = 1.0 \times \exp\left(-\frac{(1.4-1.2)^2}{2\times 2.0^2}\right)$$

$$\begin{bmatrix} 1.0 & 0.11 & 0.089 \\ 0.11 & 1.0 & \\ 0.089 & & \end{bmatrix}$$

$x_1 = -3$, $x_2 = 1.2$, $x_3 = 1.4$, and $x_4 = 2.0$ with $\ell = 2.0$ and $\alpha = 1.0$.

Covariance Functions

Where did this covariance matrix come from?

$$k\left(x_i, x_j\right) = \alpha \exp\left(-\frac{\left\|x_i - x_j\right\|^2}{2\ell^2}\right)$$

$$x_3 = 1.4,\ x_2 = 1.2$$

$$k_{3,2} = 1.0 \times \exp\left(-\frac{(1.4-1.2)^2}{2\times 2.0^2}\right)$$

$$\begin{bmatrix} 1.0 & 0.11 & 0.089 \\ 0.11 & 1.0 & \\ 0.089 & \boxed{1.0} & \\ & & \end{bmatrix}$$

$x_1 = -3$, $x_2 = 1.2$, $x_3 = 1.4$, and $x_4 = 2.0$ with $\ell = 2.0$ and $\alpha = 1.0$.

Covariance Functions

Where did this covariance matrix come from?

$$k\left(x_i, x_j\right) = \alpha \exp\left(-\frac{\left\|x_i - x_j\right\|^2}{2\ell^2}\right)$$

$$x_3 = 1.4,\ x_2 = 1.2$$

$$k_{3,2} = 1.0 \times \exp\left(-\frac{(1.4-1.2)^2}{2\times 2.0^2}\right)$$

$$\begin{bmatrix} 1.0 & 0.11 & 0.089 \\ 0.11 & 1.0 & 1.0 \\ 0.089 & 1.0 & \\ & & \end{bmatrix}$$

$x_1 = -3$, $x_2 = 1.2$, $x_3 = 1.4$, and $x_4 = 2.0$ with $\ell = 2.0$ and $\alpha = 1.0$.

Covariance Functions

Where did this covariance matrix come from?

$$k\left(x_i, x_j\right) = \alpha \exp\left(-\frac{\left\|x_i - x_j\right\|^2}{2\ell^2}\right)$$

$$x_3 = 1.4,\ x_3 = 1.4$$

$$k_{3,3} = 1.0 \times \exp\left(-\frac{(1.4-1.4)^2}{2\times 2.0^2}\right)$$

$$\begin{bmatrix} 1.0 & 0.11 & 0.089 \\ 0.11 & 1.0 & 1.0 \\ 0.089 & 1.0 & \\ & & \end{bmatrix}$$

$x_1 = -3$, $x_2 = 1.2$, $x_3 = 1.4$, and $x_4 = 2.0$ with $\ell = 2.0$ and $\alpha = 1.0$.

Covariance Functions

Where did this covariance matrix come from?

$$k\left(x_i, x_j\right) = \alpha \exp\left(-\frac{\left\|x_i - x_j\right\|^2}{2\ell^2}\right)$$

$$x_3 = 1.4,\ x_3 = 1.4$$

$$k_{3,3} = 1.0 \times \exp\left(-\frac{(1.4-1.4)^2}{2\times 2.0^2}\right)$$

$$\begin{bmatrix} 1.0 & 0.11 & 0.089 \\ 0.11 & 1.0 & 1.0 \\ 0.089 & 1.0 & \boxed{1.0} \end{bmatrix}$$

$x_1 = -3$, $x_2 = 1.2$, $x_3 = 1.4$, and $x_4 = 2.0$ with $\ell = 2.0$ and $\alpha = 1.0$.

Covariance Functions

Where did this covariance matrix come from?

$$k\left(x_i, x_j\right) = \alpha \exp\left(-\frac{\left\|x_i - x_j\right\|^2}{2\ell^2}\right)$$

$$x_4 = 2.0,\ x_1 = -3$$

$$k_{4,1} = 1.0 \times \exp\left(-\frac{(2.0 - -3)^2}{2 \times 2.0^2}\right)$$

$$\begin{bmatrix} 1.0 & 0.11 & 0.089 \\ 0.11 & 1.0 & 1.0 \\ 0.089 & 1.0 & 1.0 \\ & & \end{bmatrix}$$

$x_1 = -3$, $x_2 = 1.2$, $x_3 = 1.4$, and $x_4 = 2.0$ with $\ell = 2.0$ and $\alpha = 1.0$.

Covariance Functions

Where did this covariance matrix come from?

$$k\left(x_i, x_j\right) = \alpha \exp\left(-\frac{\left\|x_i - x_j\right\|^2}{2\ell^2}\right)$$

$$x_4 = 2.0,\ x_1 = -3$$

$$k_{4,1} = 1.0 \times \exp\left(-\frac{(2.0 - -3)^2}{2 \times 2.0^2}\right)$$

$$\begin{bmatrix} 1.0 & 0.11 & 0.089 \\ 0.11 & 1.0 & 1.0 \\ 0.089 & 1.0 & 1.0 \\ \boxed{0.044} & & \end{bmatrix}$$

$x_1 = -3$, $x_2 = 1.2$, $x_3 = 1.4$, and $x_4 = 2.0$ with $\ell = 2.0$ and $\alpha = 1.0$.

Covariance Functions

Where did this covariance matrix come from?

$$k\left(x_i, x_j\right) = \alpha \exp\left(-\frac{\left\|x_i - x_j\right\|^2}{2\ell^2}\right)$$

$$x_4 = 2.0,\ x_1 = -3$$

$$k_{4,1} = 1.0 \times \exp\left(-\frac{(2.0 - -3)^2}{2 \times 2.0^2}\right)$$

$$\begin{bmatrix} 1.0 & 0.11 & 0.089 & 0.044 \\ 0.11 & 1.0 & 1.0 & \\ 0.089 & 1.0 & 1.0 & \\ 0.044 & & & \end{bmatrix}$$

$x_1 = -3$, $x_2 = 1.2$, $x_3 = 1.4$, and $x_4 = 2.0$ with $\ell = 2.0$ and $\alpha = 1.0$.

Covariance Functions

Where did this covariance matrix come from?

$$k\left(x_i, x_j\right) = \alpha \exp\left(-\frac{\left\|x_i - x_j\right\|^2}{2\ell^2}\right)$$

$$x_4 = 2.0,\ x_2 = 1.2$$

$$k_{4,2} = 1.0 \times \exp\left(-\frac{(2.0-1.2)^2}{2\times 2.0^2}\right)$$

$$\begin{bmatrix} 1.0 & 0.11 & 0.089 & 0.044 \\ 0.11 & 1.0 & 1.0 & \\ 0.089 & 1.0 & 1.0 & \\ 0.044 & & & \end{bmatrix}$$

$x_1 = -3$, $x_2 = 1.2$, $x_3 = 1.4$, and $x_4 = 2.0$ with $\ell = 2.0$ and $\alpha = 1.0$.

Covariance Functions

Where did this covariance matrix come from?

$$k\left(x_i, x_j\right) = \alpha \exp\left(-\frac{\left\|x_i - x_j\right\|^2}{2\ell^2}\right)$$

$$x_4 = 2.0,\ x_2 = 1.2$$

$$k_{4,2} = 1.0 \times \exp\left(-\frac{(2.0-1.2)^2}{2\times 2.0^2}\right)$$

$$\begin{bmatrix} 1.0 & 0.11 & 0.089 & 0.044 \\ 0.11 & 1.0 & 1.0 & \\ 0.089 & 1.0 & 1.0 & \\ 0.044 & \boxed{0.92} & & \end{bmatrix}$$

$x_1 = -3$, $x_2 = 1.2$, $x_3 = 1.4$, and $x_4 = 2.0$ with $\ell = 2.0$ and $\alpha = 1.0$.

Covariance Functions

Where did this covariance matrix come from?

$$k\left(x_i, x_j\right) = \alpha \exp\left(-\frac{\left\|x_i - x_j\right\|^2}{2\ell^2}\right)$$

$$x_4 = 2.0,\ x_2 = 1.2$$

$$k_{4,2} = 1.0 \times \exp\left(-\frac{(2.0-1.2)^2}{2\times 2.0^2}\right)$$

$$\begin{bmatrix} 1.0 & 0.11 & 0.089 & 0.044 \\ 0.11 & 1.0 & 1.0 & 0.92 \\ 0.089 & 1.0 & 1.0 & \\ 0.044 & 0.92 & & \end{bmatrix}$$

$x_1 = -3$, $x_2 = 1.2$, $x_3 = 1.4$, and $x_4 = 2.0$ with $\ell = 2.0$ and $\alpha = 1.0$.

Covariance Functions

Where did this covariance matrix come from?

$$k\left(x_i, x_j\right) = \alpha \exp\left(-\frac{\left\|x_i - x_j\right\|^2}{2\ell^2}\right)$$

$$x_4 = 2.0,\ x_3 = 1.4$$

$$k_{4,3} = 1.0 \times \exp\left(-\frac{(2.0-1.4)^2}{2\times 2.0^2}\right)$$

$$\begin{bmatrix} 1.0 & 0.11 & 0.089 & 0.044 \\ 0.11 & 1.0 & 1.0 & 0.92 \\ 0.089 & 1.0 & 1.0 & \\ 0.044 & 0.92 & & \end{bmatrix}$$

$x_1 = -3$, $x_2 = 1.2$, $x_3 = 1.4$, and $x_4 = 2.0$ with $\ell = 2.0$ and $\alpha = 1.0$.

Covariance Functions

Where did this covariance matrix come from?

$$k\left(x_i, x_j\right) = \alpha \exp\left(-\frac{\left\|x_i - x_j\right\|^2}{2\ell^2}\right)$$

$$x_4 = 2.0,\ x_3 = 1.4$$

$$k_{4,3} = 1.0 \times \exp\left(-\frac{(2.0-1.4)^2}{2\times 2.0^2}\right)$$

$$\begin{bmatrix} 1.0 & 0.11 & 0.089 & 0.044 \\ 0.11 & 1.0 & 1.0 & 0.92 \\ 0.089 & 1.0 & 1.0 & \\ 0.044 & 0.92 & \boxed{0.96} & \end{bmatrix}$$

$x_1 = -3$, $x_2 = 1.2$, $x_3 = 1.4$, and $x_4 = 2.0$ with $\ell = 2.0$ and $\alpha = 1.0$.

Covariance Functions

Where did this covariance matrix come from?

$$k\left(x_i, x_j\right) = \alpha \exp\left(-\frac{\left\|x_i - x_j\right\|^2}{2\ell^2}\right)$$

$$x_4 = 2.0,\ x_3 = 1.4$$

$$k_{4,3} = 1.0 \times \exp\left(-\frac{(2.0-1.4)^2}{2\times 2.0^2}\right)$$

$$\begin{bmatrix} 1.0 & 0.11 & 0.089 & 0.044 \\ 0.11 & 1.0 & 1.0 & 0.92 \\ 0.089 & 1.0 & 1.0 & 0.96 \\ 0.044 & 0.92 & 0.96 & \end{bmatrix}$$

$x_1 = -3$, $x_2 = 1.2$, $x_3 = 1.4$, and $x_4 = 2.0$ with $\ell = 2.0$ and $\alpha = 1.0$.

Covariance Functions

Where did this covariance matrix come from?

$$k\left(x_i, x_j\right) = \alpha \exp\left(-\frac{\left\|x_i - x_j\right\|^2}{2\ell^2}\right)$$

$$x_4 = 2.0,\ x_4 = 2.0$$

$$k_{4,4} = 1.0 \times \exp\left(-\frac{(2.0-2.0)^2}{2\times 2.0^2}\right)$$

$$\begin{bmatrix} 1.0 & 0.11 & 0.089 & 0.044 \\ 0.11 & 1.0 & 1.0 & 0.92 \\ 0.089 & 1.0 & 1.0 & 0.96 \\ 0.044 & 0.92 & 0.96 & \end{bmatrix}$$

$x_1 = -3$, $x_2 = 1.2$, $x_3 = 1.4$, and $x_4 = 2.0$ with $\ell = 2.0$ and $\alpha = 1.0$.

Covariance Functions

Where did this covariance matrix come from?

$$k\left(x_i, x_j\right) = \alpha \exp\left(-\frac{\left\|x_i - x_j\right\|^2}{2\ell^2}\right)$$

$$x_4 = 2.0,\ x_4 = 2.0$$

$$k_{4,4} = 1.0 \times \exp\left(-\frac{(2.0-2.0)^2}{2\times 2.0^2}\right)$$

$$\begin{bmatrix} 1.0 & 0.11 & 0.089 & 0.044 \\ 0.11 & 1.0 & 1.0 & 0.92 \\ 0.089 & 1.0 & 1.0 & 0.96 \\ 0.044 & 0.92 & 0.96 & \boxed{1.0} \end{bmatrix}$$

$x_1 = -3$, $x_2 = 1.2$, $x_3 = 1.4$, and $x_4 = 2.0$ with $\ell = 2.0$ and $\alpha = 1.0$.

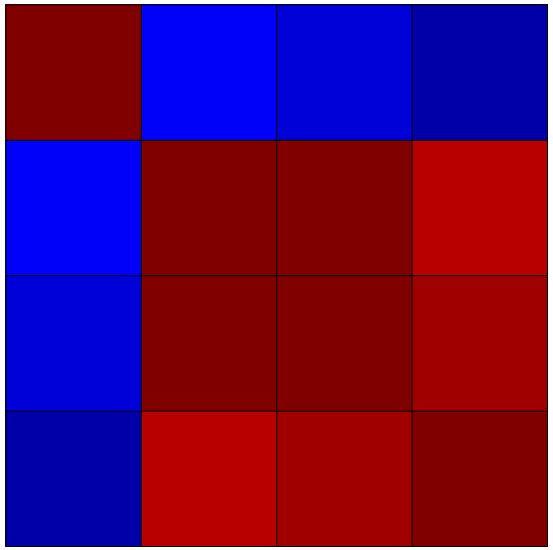
Covariance Functions

Where did this covariance matrix come from?

$$k\left(x_i, x_j\right) = \alpha \exp\left(-\frac{\left\|x_i - x_j\right\|^2}{2\ell^2}\right)$$

$$x_4 = 2.0,\ x_4 = 2.0$$

$$k_{4,4} = 1.0 \times \exp\left(-\frac{(2.0-2.0)^2}{2\times 2.0^2}\right)$$



$x_1 = -3$, $x_2 = 1.2$, $x_3 = 1.4$, and $x_4 = 2.0$ with $\ell = 2.0$ and $\alpha = 1.0$.

Covariance Functions

Where did this covariance matrix come from?

$$k\left(x_i, x_j\right) = \alpha \exp\left(-\frac{\left\|x_i - x_j\right\|^2}{2\ell^2}\right)$$

$$x_1 = -3.0, x_1 = -3.0$$

$$k_{1,1} = 4.00 \times \exp\left(-\frac{(-3.0 - -3.0)^2}{2 \times 5.00^2}\right)$$

$x_1 = -3.0$, $x_2 = 1.20$, and $x_3 = 1.40$ with $\ell = 5.00$ and $\alpha = 4.00$.

Covariance Functions

Where did this covariance matrix come from?

$$k\left(x_i, x_j\right) = \alpha \exp\left(-\frac{\left\|x_i - x_j\right\|^2}{2\ell^2}\right)$$

$$x_1 = -3.0, x_1 = -3.0$$

$$k_{1,1} = 4.00 \times \exp\left(-\frac{(-3.0 - -3.0)^2}{2 \times 5.00^2}\right)$$

$$\begin{bmatrix} \boxed{4.00} & & \\ & & \\ & & \end{bmatrix}$$

$x_1 = -3.0$, $x_2 = 1.20$, and $x_3 = 1.40$ with $\ell = 5.00$ and $\alpha = 4.00$.

Covariance Functions

Where did this covariance matrix come from?

$$k\left(x_i, x_j\right) = \alpha \exp\left(-\frac{\left\|x_i - x_j\right\|^2}{2\ell^2}\right)$$

$$x_2 = 1.20,\ x_1 = -3.0$$

$$k_{2,1} = 4.00 \times \exp\left(-\frac{(1.20 - -3.0)^2}{2 \times 5.00^2}\right)$$

$$\begin{bmatrix} 4.00 & \\ & \\ & \end{bmatrix}$$

$x_1 = -3.0$, $x_2 = 1.20$, and $x_3 = 1.40$ with $\ell = 5.00$ and $\alpha = 4.00$.

Covariance Functions

Where did this covariance matrix come from?

$$k\left(x_i, x_j\right) = \alpha \exp\left(-\frac{\left\|x_i - x_j\right\|^2}{2\ell^2}\right)$$

$$x_2 = 1.20,\ x_1 = -3.0$$

$$k_{2,1} = 4.00 \times \exp\left(-\frac{(1.20 - -3.0)^2}{2 \times 5.00^2}\right)$$

$$\begin{bmatrix} 4.00 & & \\ \boxed{2.81} & & \\ & & \end{bmatrix}$$

$x_1 = -3.0$, $x_2 = 1.20$, and $x_3 = 1.40$ with $\ell = 5.00$ and $\alpha = 4.00$.

Covariance Functions

Where did this covariance matrix come from?

$$k\left(x_i, x_j\right) = \alpha \exp\left(-\frac{\left\|x_i - x_j\right\|^2}{2\ell^2}\right)$$

$$x_2 = 1.20,\ x_1 = -3.0$$

$$k_{2,1} = 4.00 \times \exp\left(-\frac{(1.20 - -3.0)^2}{2 \times 5.00^2}\right)$$

$$\begin{bmatrix} 4.00 & 2.81 \\ 2.81 & \\ & \end{bmatrix}$$

$x_1 = -3.0$, $x_2 = 1.20$, and $x_3 = 1.40$ with $\ell = 5.00$ and $\alpha = 4.00$.

Covariance Functions

Where did this covariance matrix come from?

$$k\left(x_i, x_j\right) = \alpha \exp\left(-\frac{\left\|x_i - x_j\right\|^2}{2\ell^2}\right)$$

$$x_2 = 1.20,\ x_2 = 1.20$$

$$k_{2,2} = 4.00 \times \exp\left(-\frac{(1.20-1.20)^2}{2\times 5.00^2}\right)$$

$$\begin{bmatrix} 4.00 & 2.81 \\ 2.81 & \\ & \end{bmatrix}$$

$x_1 = -3.0$, $x_2 = 1.20$, and $x_3 = 1.40$ with $\ell = 5.00$ and $\alpha = 4.00$.

Covariance Functions

Where did this covariance matrix come from?

$$k\left(x_i, x_j\right) = \alpha \exp\left(-\frac{\left\|x_i - x_j\right\|^2}{2\ell^2}\right)$$

$$x_2 = 1.20,\ x_2 = 1.20$$

$$k_{2,2} = 4.00 \times \exp\left(-\frac{(1.20-1.20)^2}{2\times 5.00^2}\right)$$

$$\begin{bmatrix} 4.00 & 2.81 & \\ 2.81 & \boxed{4.00} & \\ & & \end{bmatrix}$$

$x_1 = -3.0$, $x_2 = 1.20$, and $x_3 = 1.40$ with $\ell = 5.00$ and $\alpha = 4.00$.

Covariance Functions

Where did this covariance matrix come from?

$$k\left(x_i, x_j\right) = \alpha \exp\left(-\frac{\left\|x_i - x_j\right\|^2}{2\ell^2}\right)$$

$$x_3 = 1.40,\ x_1 = -3.0$$

$$k_{3,1} = 4.00 \times \exp\left(-\frac{(1.40 - -3.0)^2}{2 \times 5.00^2}\right)$$

$$\begin{bmatrix} 4.00 & 2.81 \\ 2.81 & 4.00 \end{bmatrix}$$

$x_1 = -3.0$, $x_2 = 1.20$, and $x_3 = 1.40$ with $\ell = 5.00$ and $\alpha = 4.00$.

Covariance Functions

Where did this covariance matrix come from?

$$k\left(x_i, x_j\right) = \alpha \exp\left(-\frac{\left\|x_i - x_j\right\|^2}{2\ell^2}\right)$$

$$x_3 = 1.40,\ x_1 = -3.0$$

$$k_{3,1} = 4.00 \times \exp\left(-\frac{(1.40 - -3.0)^2}{2 \times 5.00^2}\right)$$

$$\begin{bmatrix} 4.00 & 2.81 \\ 2.81 & 4.00 \\ \boxed{2.72} & \end{bmatrix}$$

$x_1 = -3.0$, $x_2 = 1.20$, and $x_3 = 1.40$ with $\ell = 5.00$ and $\alpha = 4.00$.

Covariance Functions

Where did this covariance matrix come from?

$$k\left(x_i, x_j\right) = \alpha \exp\left(-\frac{\left\|x_i - x_j\right\|^2}{2\ell^2}\right)$$

$$x_3 = 1.40,\ x_1 = -3.0$$

$$k_{3,1} = 4.00 \times \exp\left(-\frac{(1.40 - -3.0)^2}{2 \times 5.00^2}\right)$$

$$\begin{bmatrix} 4.00 & 2.81 & 2.72 \\ 2.81 & 4.00 & \\ 2.72 & & \end{bmatrix}$$

$x_1 = -3.0$, $x_2 = 1.20$, and $x_3 = 1.40$ with $\ell = 5.00$ and $\alpha = 4.00$.

Covariance Functions

Where did this covariance matrix come from?

$$k\left(x_i, x_j\right) = \alpha \exp\left(-\frac{\left\|x_i - x_j\right\|^2}{2\ell^2}\right)$$

$$x_3 = 1.40,\ x_2 = 1.20$$

$$k_{3,2} = 4.00 \times \exp\left(-\frac{(1.40-1.20)^2}{2\times 5.00^2}\right)$$

$$\begin{bmatrix} 4.00 & 2.81 & 2.72 \\ 2.81 & 4.00 & \\ 2.72 & & \end{bmatrix}$$

$x_1 = -3.0$, $x_2 = 1.20$, and $x_3 = 1.40$ with $\ell = 5.00$ and $\alpha = 4.00$.

Covariance Functions

Where did this covariance matrix come from?

$$k\left(x_i, x_j\right) = \alpha \exp\left(-\frac{\left\|x_i - x_j\right\|^2}{2\ell^2}\right)$$

$$x_3 = 1.40,\ x_2 = 1.20$$

$$k_{3,2} = 4.00 \times \exp\left(-\frac{(1.40-1.20)^2}{2\times 5.00^2}\right)$$

$$\begin{bmatrix} 4.00 & 2.81 & 2.72 \\ 2.81 & 4.00 & \\ 2.72 & \boxed{4.00} & \end{bmatrix}$$

$x_1 = -3.0$, $x_2 = 1.20$, and $x_3 = 1.40$ with $\ell = 5.00$ and $\alpha = 4.00$.

Covariance Functions

Where did this covariance matrix come from?

$$k\left(x_i, x_j\right) = \alpha \exp\left(-\frac{\left\|x_i - x_j\right\|^2}{2\ell^2}\right)$$

$$x_3 = 1.40,\ x_2 = 1.20$$

$$k_{3,2} = 4.00 \times \exp\left(-\frac{(1.40-1.20)^2}{2\times 5.00^2}\right)$$

$$\begin{bmatrix} 4.00 & 2.81 & 2.72 \\ 2.81 & 4.00 & 4.00 \\ 2.72 & 4.00 & \end{bmatrix}$$

$x_1 = -3.0$, $x_2 = 1.20$, and $x_3 = 1.40$ with $\ell = 5.00$ and $\alpha = 4.00$.

Covariance Functions

Where did this covariance matrix come from?

$$k\left(x_i, x_j\right) = \alpha \exp\left(-\frac{\left\|x_i - x_j\right\|^2}{2\ell^2}\right)$$

$$x_3 = 1.40,\ x_3 = 1.40$$

$$k_{3,3} = 4.00 \times \exp\left(-\frac{(1.40-1.40)^2}{2\times 5.00^2}\right)$$

$$\begin{bmatrix} 4.00 & 2.81 & 2.72 \\ 2.81 & 4.00 & 4.00 \\ 2.72 & 4.00 & \end{bmatrix}$$

$x_1 = -3.0$, $x_2 = 1.20$, and $x_3 = 1.40$ with $\ell = 5.00$ and $\alpha = 4.00$.

Covariance Functions

Where did this covariance matrix come from?

$$k\left(x_i, x_j\right) = \alpha \exp\left(-\frac{\left\|x_i - x_j\right\|^2}{2\ell^2}\right)$$

$$x_3 = 1.40,\ x_3 = 1.40$$

$$k_{3,3} = 4.00 \times \exp\left(-\frac{(1.40-1.40)^2}{2\times 5.00^2}\right)$$

$$\begin{bmatrix} 4.00 & 2.81 & 2.72 \\ 2.81 & 4.00 & 4.00 \\ 2.72 & 4.00 & \boxed{4.00} \end{bmatrix}$$

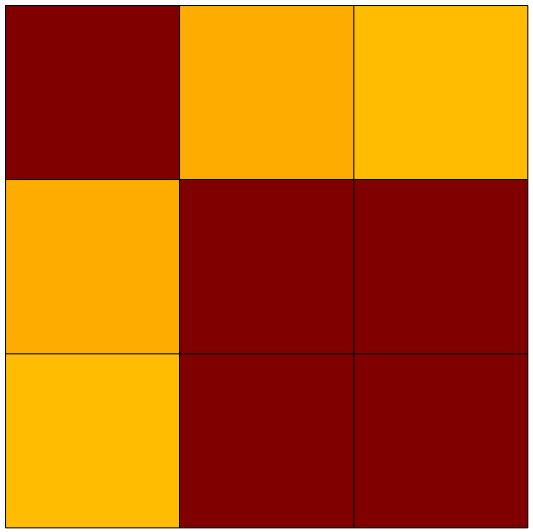
$x_1 = -3.0$, $x_2 = 1.20$, and $x_3 = 1.40$ with $\ell = 5.00$ and $\alpha = 4.00$.

Covariance Functions

Where did this covariance matrix come from?

$$k\left(x_i, x_j\right) = \alpha \exp\left(-\frac{\left\|x_i - x_j\right\|^2}{2\ell^2}\right)$$

$$x_3 = 1.40,\ x_3 = 1.40$$



$$k_{3,3} = 4.00 \times \exp\left(-\frac{(1.40-1.40)^2}{2\times 5.00^2}\right)$$

$x_1 = -3.0$, $x_2 = 1.20$, and $x_3 = 1.40$ with $\ell = 5.00$ and $\alpha = 4.00$.

Outline

Basis Function Form

Radial basis functions commonly have the form

$$\phi_k(\mathbf{x}_i) = \exp\left(-\frac{\left|\mathbf{x}_i - \boldsymbol{\mu}_k\right|^2}{2\ell^2}\right).$$

- Basis function maps data into a "feature space" in which a linear sum is a non linear function.

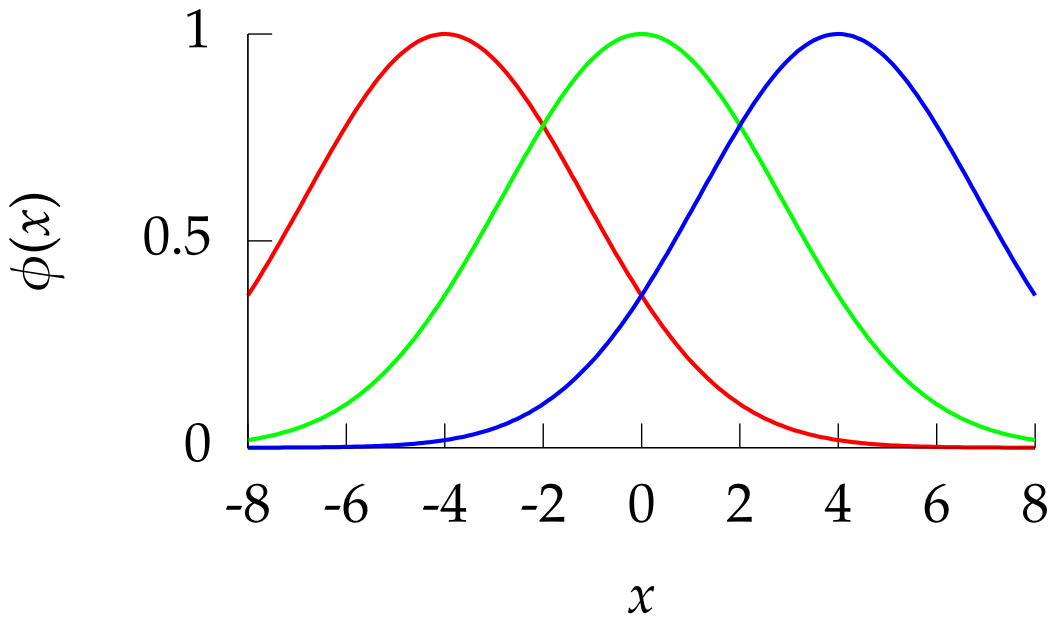


Figure : A set of radial basis functions with width $\ell = 2$ and location parameters $\boldsymbol{\mu} = [-4 \;\; 0 \;\; 4]^\top$.

Basis Function Representations

- Represent a function by a linear sum over a basis,

$$f(\mathbf{x}_{i,:}; \mathbf{w}) = \sum_{k=1}^{m} w_k \phi_k(\mathbf{x}_{i,:}), \tag{1}$$

- Here: m basis functions and $\phi_k(\cdot)$ is kth basis function and

$$\mathbf{w} = [w_1, \ldots, w_m]^\top .$$

- For standard linear model: $\phi_k(\mathbf{x}_{i,:}) = x_{i,k}$.

Random Functions

Functions derived using:

$$f(x) = \sum_{k=1}^{m} w_k \phi_k(x),$$

where $\mathbf{W}$ is sampled from a Gaussian density,

$$w_k \sim \mathcal{N}(0, \alpha).$$

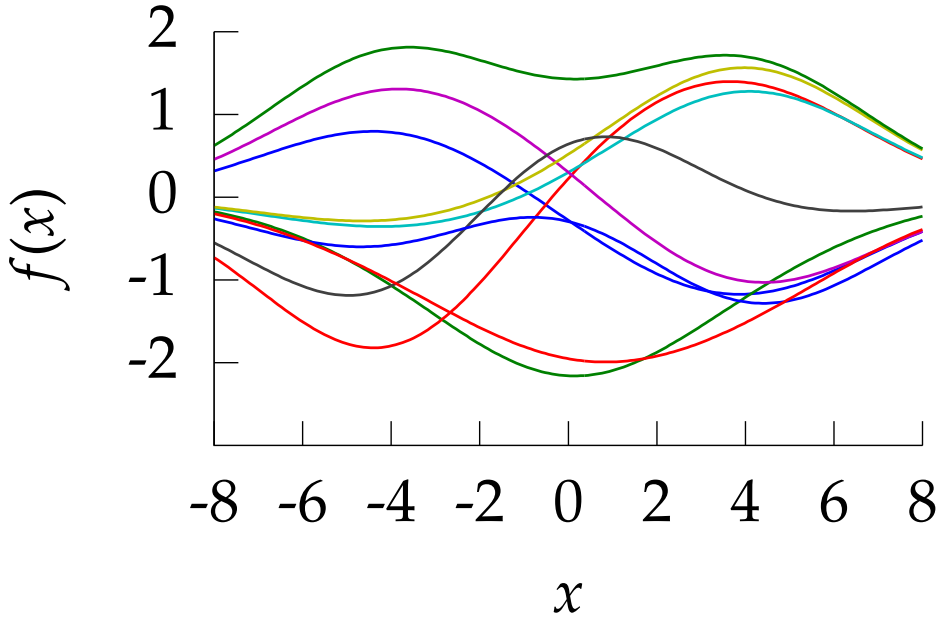


Figure : Functions sampled using the basis set from figure 3. Each line is a separate sample, generated by a weighted sum of the basis set. The weights, $\mathbf{w}$ are sampled from a Gaussian density with variance $\alpha = 1$.

Direct Construction of Covariance Matrix

Use matrix notation to write function,

$$f\left(\mathbf{x}_i; \mathbf{w}\right) = \sum_{k=1}^{m} w_k \phi_k\left(\mathbf{x}_i\right)$$

Direct Construction of Covariance Matrix

Use matrix notation to write function,

$$f(\mathbf{x}_i; \mathbf{w}) = \sum_{k=1}^{m} w_k \phi_k(\mathbf{x}_i)$$

computed at training data gives a vector

$$\mathbf{f} = \mathbf{\Phi}\mathbf{w}.$$

Direct Construction of Covariance Matrix

Use matrix notation to write function,

$$f(\mathbf{x}_i; \mathbf{w}) = \sum_{k=1}^{m} w_k \phi_k(\mathbf{x}_i)$$

computed at training data gives a vector

$$\mathbf{f} = \mathbf{\Phi}\mathbf{w}.$$

$$\mathbf{w} \sim \mathcal{N}(\mathbf{0}, \alpha\mathbf{I})$$

Direct Construction of Covariance Matrix

Use matrix notation to write function,

$$f(\mathbf{x}_i; \mathbf{w}) = \sum_{k=1}^{m} w_k \phi_k(\mathbf{x}_i)$$

computed at training data gives a vector

$$\mathbf{f} = \mathbf{\Phi}\mathbf{w}.$$

$$\mathbf{w} \sim \mathcal{N}(\mathbf{0}, \alpha\mathbf{I})$$

$\mathbf{w}$ and $\mathbf{f}$ are only related by an *inner product*.

Direct Construction of Covariance Matrix

Use matrix notation to write function,

$$f(\mathbf{x}_i; \mathbf{w}) = \sum_{k=1}^{m} w_k \phi_k(\mathbf{x}_i)$$

computed at training data gives a vector

$$\mathbf{f} = \mathbf{\Phi}\mathbf{w}.$$

$$\mathbf{w} \sim \mathcal{N}(\mathbf{0}, \alpha\mathbf{I})$$

$\mathbf{w}$ and $\mathbf{f}$ are only related by an *inner product*.

$\mathbf{\Phi} \in \Re^{n \times p}$ is a *design matrix*

Direct Construction of Covariance Matrix

Use matrix notation to write function,

$$f(\mathbf{x}_i; \mathbf{w}) = \sum_{k=1}^{m} w_k \phi_k(\mathbf{x}_i)$$

computed at training data gives a vector

$$\mathbf{f} = \mathbf{\Phi}\mathbf{w}.$$

$$\mathbf{w} \sim \mathcal{N}(\mathbf{0}, \alpha\mathbf{I})$$

$\mathbf{w}$ and $\mathbf{f}$ are only related by an *inner product*.

$\mathbf{\Phi} \in \Re^{n \times p}$ is a *design matrix*

$\mathbf{\Phi}$ is fixed and non-stochastic for a given training set.

Direct Construction of Covariance Matrix

Use matrix notation to write function,

$$f(\mathbf{x}_i; \mathbf{w}) = \sum_{k=1}^{m} w_k \phi_k(\mathbf{x}_i)$$

computed at training data gives a vector

$$\mathbf{f} = \mathbf{\Phi}\mathbf{w}.$$

$$\mathbf{w} \sim \mathcal{N}(\mathbf{0}, \alpha\mathbf{I})$$

$\mathbf{w}$ and $\mathbf{f}$ are only related by an *inner product*.

$\mathbf{\Phi} \in \Re^{n \times p}$ is a *design matrix*

$\mathbf{\Phi}$ is fixed and non-stochastic for a given training set.

$\mathbf{f}$ is Gaussian distributed.

Expectations

- We have

$$\langle \mathbf{f} \rangle = \mathbf{\Phi} \langle \mathbf{w} \rangle .$$

We use $\langle \cdot \rangle$ to denote expectations under prior distributions.

Expectations

- We have
$$\langle \mathbf{f} \rangle = \mathbf{\Phi} \langle \mathbf{w} \rangle .$$
- Prior mean of $\mathbf{w}$ was zero giving
$$\langle \mathbf{f} \rangle = \mathbf{0}.$$

We use $\langle \cdot \rangle$ to denote expectations under prior distributions.

Expectations

- We have
$$\langle \mathbf{f} \rangle = \mathbf{\Phi} \langle \mathbf{w} \rangle .$$
- Prior mean of $\mathbf{w}$ was zero giving
$$\langle \mathbf{f} \rangle = \mathbf{0}.$$
- Prior covariance of $\mathbf{f}$ is
$$\mathbf{K} = \left\langle \mathbf{f}\mathbf{f}^\top \right\rangle - \langle \mathbf{f} \rangle \langle \mathbf{f} \rangle^\top$$

We use $\langle \cdot \rangle$ to denote expectations under prior distributions.

Expectations

- We have
$$\langle \mathbf{f} \rangle = \mathbf{\Phi} \langle \mathbf{w} \rangle .$$
- Prior mean of $\mathbf{w}$ was zero giving
$$\langle \mathbf{f} \rangle = \mathbf{0}.$$
- Prior covariance of $\mathbf{f}$ is
$$\mathbf{K} = \left\langle \mathbf{f}\mathbf{f}^\top \right\rangle - \langle \mathbf{f} \rangle \langle \mathbf{f} \rangle^\top$$
$$\left\langle \mathbf{f}\mathbf{f}^\top \right\rangle = \mathbf{\Phi} \left\langle \mathbf{w}\mathbf{w}^\top \right\rangle \mathbf{\Phi}^\top,$$
giving
$$\mathbf{K} = \alpha \mathbf{\Phi}\mathbf{\Phi}^\top.$$

We use $\langle \cdot \rangle$ to denote expectations under prior distributions.

Covariance between Two Points

- The prior covariance between two points $\mathbf{x}_i$ and $\mathbf{x}_j$ is

$$k\left(\mathbf{x}_i, \mathbf{x}_j\right) = \alpha \phi_{:}\left(\mathbf{x}_i\right)^{\top} \phi_{:}\left(\mathbf{x}_j\right),$$

Covariance between Two Points

- The prior covariance between two points $\mathbf{x}_i$ and $\mathbf{x}_j$ is

$$k\left(\mathbf{x}_i, \mathbf{x}_j\right) = \alpha \phi_{:}\left(\mathbf{x}_i\right)^{\top} \phi_{:}\left(\mathbf{x}_j\right),$$

or in sum notation

$$k\left(\mathbf{x}_i, \mathbf{x}_j\right) = \alpha \sum_{k=1}^{m} \phi_k\left(\mathbf{x}_i\right) \phi_k\left(\mathbf{x}_j\right)$$

Covariance between Two Points

- The prior covariance between two points $\mathbf{x}_i$ and $\mathbf{x}_j$ is

$$k\left(\mathbf{x}_i, \mathbf{x}_j\right) = \alpha \phi_{:}\left(\mathbf{x}_i\right)^{\top} \phi_{:}\left(\mathbf{x}_j\right),$$

  or in sum notation

$$k\left(\mathbf{x}_i, \mathbf{x}_j\right) = \alpha \sum_{k=1}^{m} \phi_k\left(\mathbf{x}_i\right) \phi_k\left(\mathbf{x}_j\right)$$

- For the radial basis used this gives

Covariance between Two Points

- The prior covariance between two points $\mathbf{x}_i$ and $\mathbf{x}_j$ is

$$k\left(\mathbf{x}_i, \mathbf{x}_j\right) = \alpha \phi_{:}\left(\mathbf{x}_i\right)^{\top} \phi_{:}\left(\mathbf{x}_j\right),$$

  or in sum notation

$$k\left(\mathbf{x}_i, \mathbf{x}_j\right) = \alpha \sum_{k=1}^{m} \phi_k\left(\mathbf{x}_i\right) \phi_k\left(\mathbf{x}_j\right)$$

- For the radial basis used this gives

$$k\left(\mathbf{x}_i, \mathbf{x}_j\right) = \alpha \sum_{k=1}^{m} \exp\left(-\frac{\left|\mathbf{x}_i - \boldsymbol{\mu}_k\right|^2 + \left|\mathbf{x}_j - \boldsymbol{\mu}_k\right|^2}{2\ell^2}\right).$$

Constructing Covariance Functions

- Sum of two covariances is also a covariance function.

$$k(\mathbf{x}, \mathbf{x}') = k_1(\mathbf{x}, \mathbf{x}') + k_2(\mathbf{x}, \mathbf{x}')$$

Constructing Covariance Functions

- Product of two covariances is also a covariance function.

$$k(\mathbf{x}, \mathbf{x}') = k_1(\mathbf{x}, \mathbf{x}')k_2(\mathbf{x}, \mathbf{x}')$$

Multiply by Deterministic Function

- If $f(\mathbf{x})$ is a Gaussian process.
- $g(\mathbf{x})$ is a deterministic function.
- $h(\mathbf{x}) = f(\mathbf{x})g(\mathbf{x})$
- Then

$$k_h(\mathbf{x}, \mathbf{x}') = g(\mathbf{x})k_f(\mathbf{x}, \mathbf{x}')g(\mathbf{x}')$$

where k_h is covariance for $h(\cdot)$ and k_f is covariance for $f(\cdot)$.

Covariance Functions

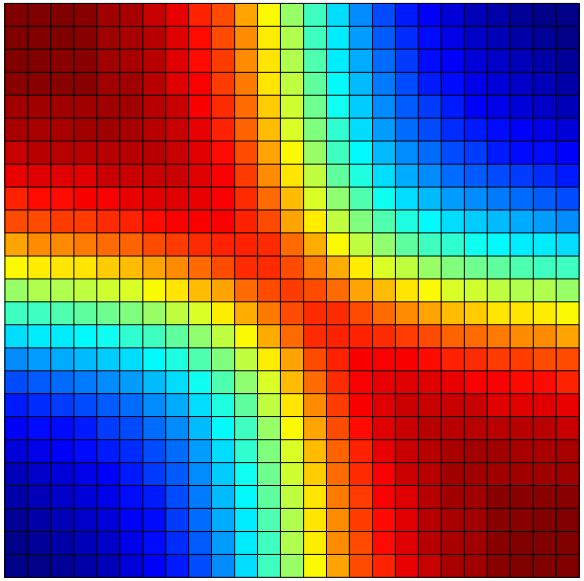
MLP Covariance Function

$$k(\mathbf{x}, \mathbf{x}') = \alpha \operatorname{asin}\left(\frac{w\mathbf{x}^\top\mathbf{x}' + b}{\sqrt{w\mathbf{x}^\top\mathbf{x} + b + 1}\sqrt{w\mathbf{x}'^\top\mathbf{x}' + b + 1}}\right)$$

- Based on infinite neural network model.

$$w = 40$$

$$b = 4$$



Covariance Functions

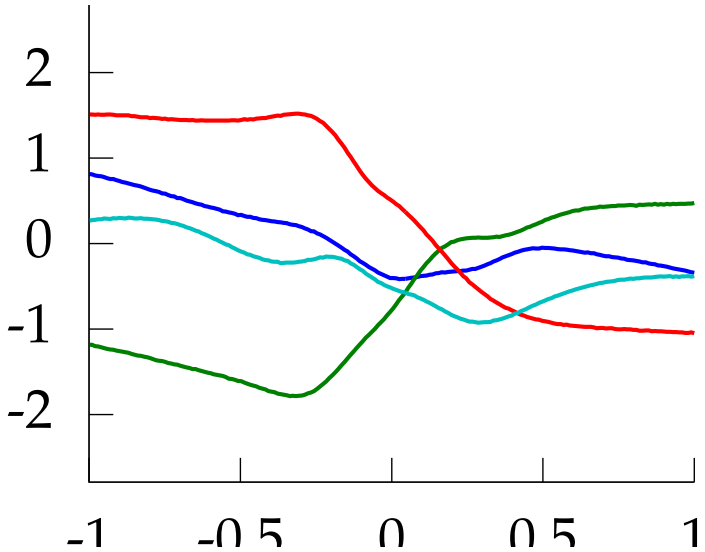
MLP Covariance Function

$$k(\mathbf{x}, \mathbf{x}') = \alpha \operatorname{asin}\left(\frac{w\mathbf{x}^\top \mathbf{x}' + b}{\sqrt{w\mathbf{x}^\top \mathbf{x} + b + 1}\sqrt{w\mathbf{x}'^\top \mathbf{x}' + b + 1}}\right)$$

- Based on infinite neural network model.

$$w = 40$$

$$b = 4$$



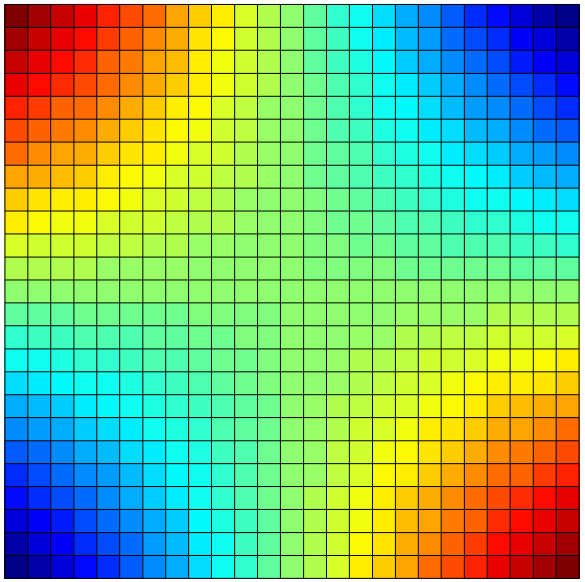
Covariance Functions

Linear Covariance Function

$$k(\mathbf{x}, \mathbf{x}') = \alpha \mathbf{x}^\top \mathbf{x}'$$

- Bayesian linear regression.

$$\alpha = 1$$



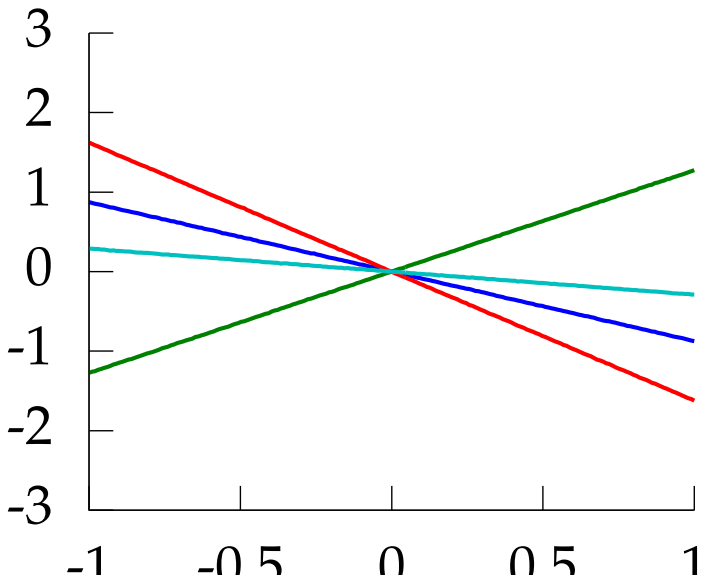
Covariance Functions

Linear Covariance Function

$$k(\mathbf{x}, \mathbf{x}') = \alpha \mathbf{x}^\top \mathbf{x}'$$

- Bayesian linear regression.

$$\alpha = 1$$



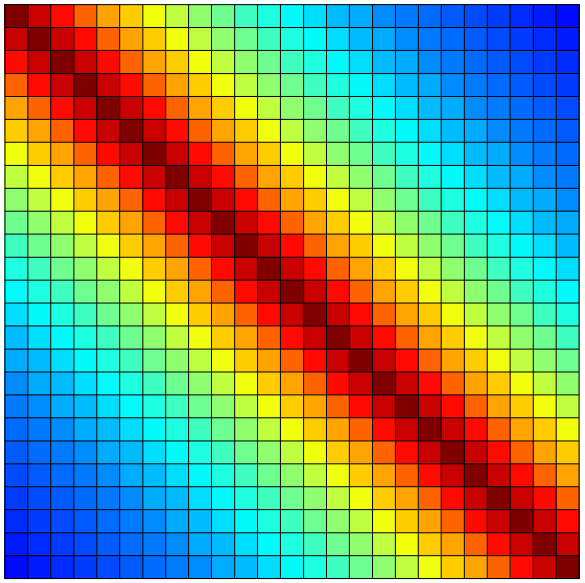
Covariance Functions

Where did this covariance matrix come from?

Ornstein-Uhlenbeck (stationary Gauss-Markov) covariance function

$$k(\mathbf{x}, \mathbf{x}') = \alpha \exp\left(-\frac{|\mathbf{x} - \mathbf{x}'|}{2\ell^2}\right)$$

- In one dimension arises from a stochastic differential equation. Brownian motion in a parabolic tube.
- In higher dimension a Fourier filter of the form $\frac{1}{\pi(1+x^2)}$.



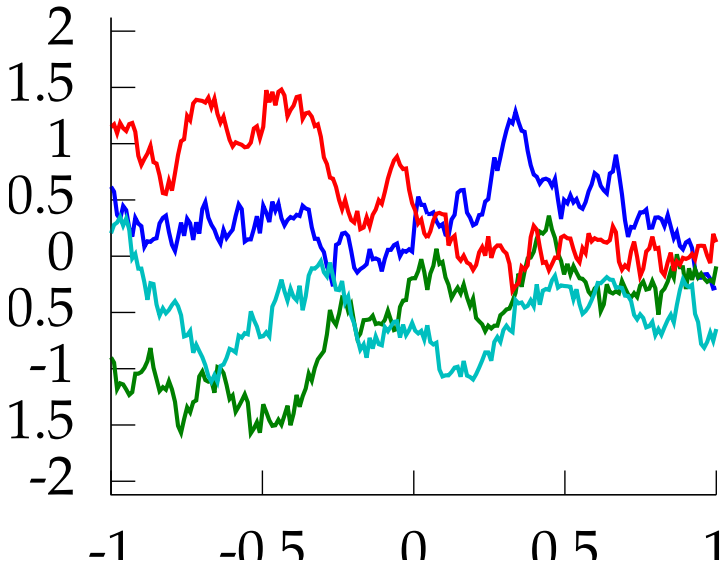
Covariance Functions

Where did this covariance matrix come from?

Ornstein-Uhlenbeck (stationary Gauss-Markov) covariance function

$$k(\mathbf{x}, \mathbf{x}') = \alpha \exp\left(-\frac{|\mathbf{x} - \mathbf{x}'|}{2\ell^2}\right)$$

- In one dimension arises from a stochastic differential equation. Brownian motion in a parabolic tube.
- In higher dimension a Fourier filter of the form $\frac{1}{\pi(1+x^2)}$.



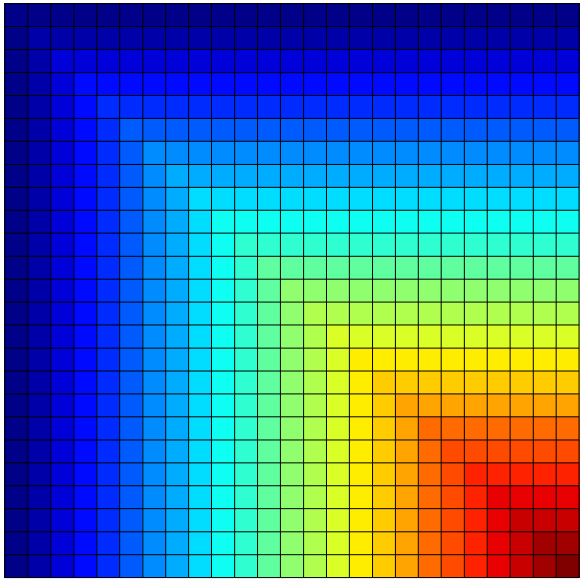
Covariance Functions

Where did this covariance matrix come from?

Markov Process

$$k(t, t') = \alpha \min(t, t')$$

- Covariance matrix is built using the *inputs* to the function t.



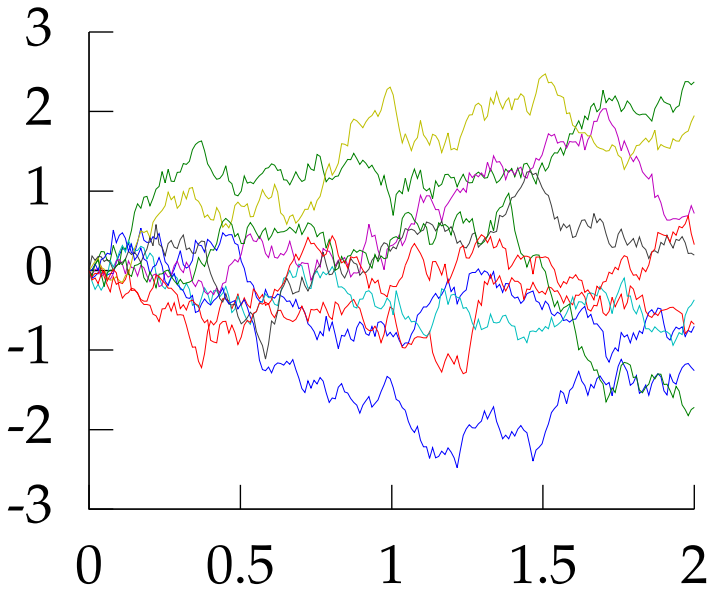
Covariance Functions

Where did this covariance matrix come from?

Markov Process

$$k(t, t') = \alpha \min(t, t')$$

- Covariance matrix is built using the *inputs* to the function t.



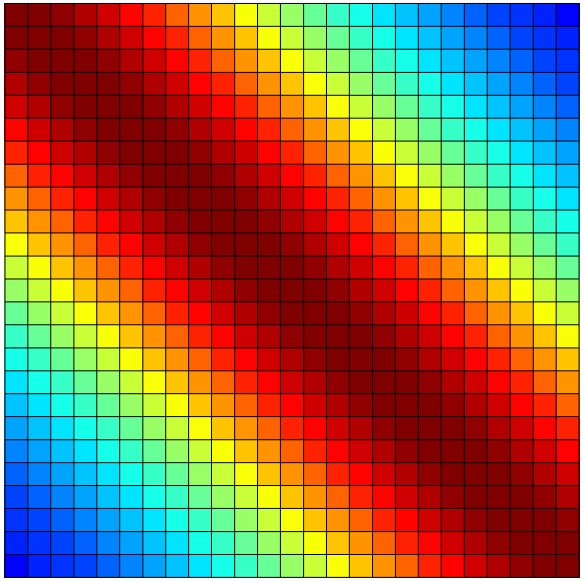
Covariance Functions

Where did this covariance matrix come from?

Matern 5/2 Covariance Function

$$k(\mathbf{x}, \mathbf{x}') = \alpha\left(1 + \sqrt{5}r + \frac{5}{3}r^2\right)\exp\left(-\sqrt{5}r\right) \quad \text{where} \quad r = \frac{\|\mathbf{x} - \mathbf{x}'\|_2}{\ell}$$

- Matern 5/2 is a twice differentiable covariance.
- Matern family constructed with Student-t filters in Fourier space.



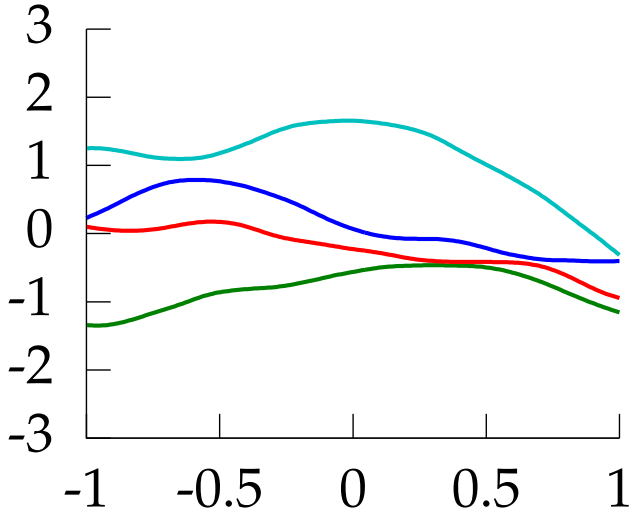
Covariance Functions

Where did this covariance matrix come from?

Matern 5/2 Covariance Function

$$k(\mathbf{x}, \mathbf{x}') = \alpha\left(1 + \sqrt{5}r + \frac{5}{3}r^2\right)\exp\left(-\sqrt{5}r\right) \quad \text{where} \quad r = \frac{\|\mathbf{x} - \mathbf{x}'\|_2}{\ell}$$

- Matern 5/2 is a twice differentiable covariance.
- Matern family constructed with Student-t filters in Fourier space.



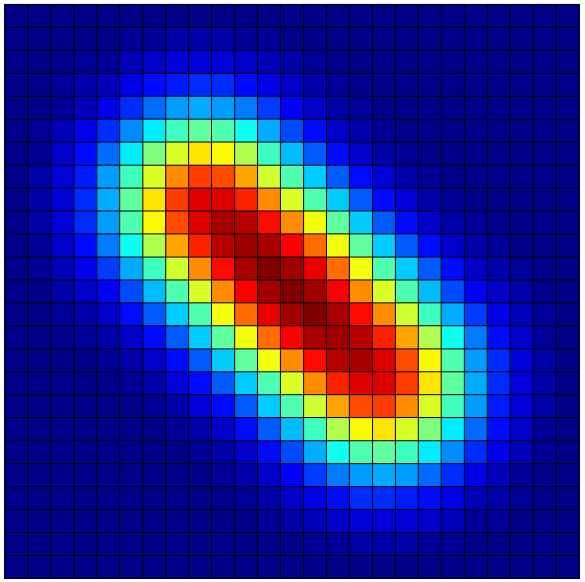
Covariance Functions

RBF Basis Functions

$$k(\mathbf{x}, \mathbf{x}') = \alpha \boldsymbol{\phi}(\mathbf{x})^\top \boldsymbol{\phi}(\mathbf{x}')$$

$$\phi_i(x) = \exp\left(-\frac{\left\|x - \mu_i\right\|_2^2}{\ell^2}\right)$$

$$\boldsymbol{\mu} = \begin{bmatrix} -1 \\ 0 \\ 1 \end{bmatrix}$$



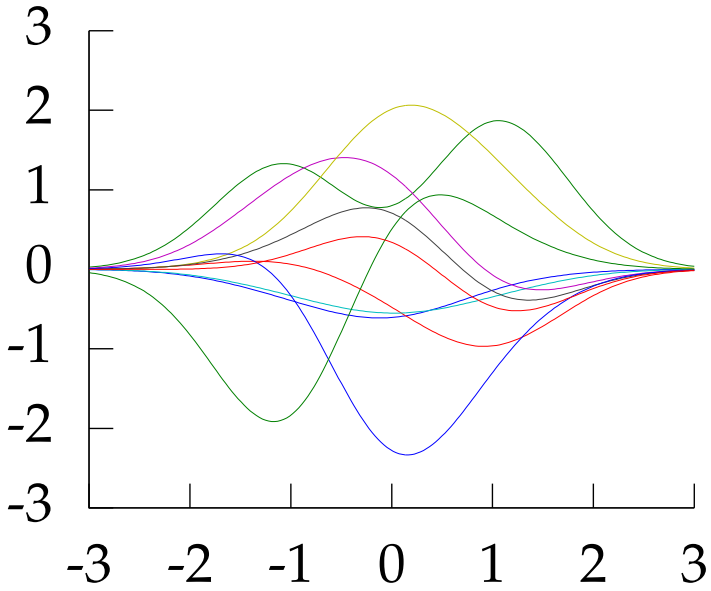
Covariance Functions

RBF Basis Functions

$$k(\mathbf{x}, \mathbf{x}') = \alpha \boldsymbol{\phi}(\mathbf{x})^\top \boldsymbol{\phi}(\mathbf{x}')$$

$$\phi_i(x) = \exp\left(-\frac{\left\|x - \mu_i\right\|_2^2}{\ell^2}\right)$$

$$\boldsymbol{\mu} = \begin{bmatrix} -1 \\ 0 \\ 1 \end{bmatrix}$$



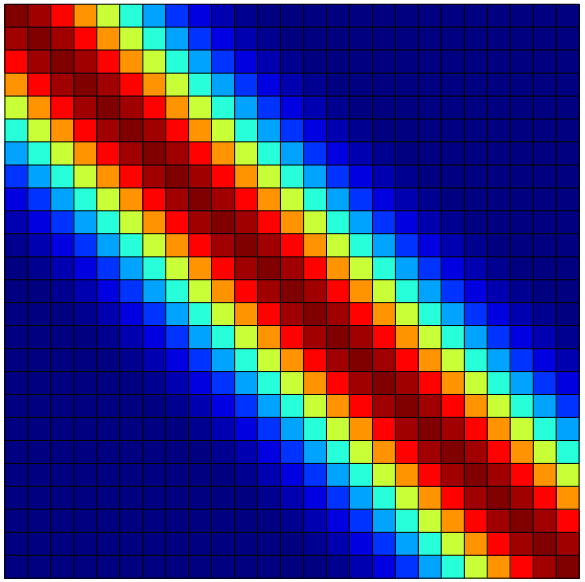
Covariance Functions

Where did this covariance matrix come from?

Exponentiated Quadratic Kernel Function (RBF, Squared Exponential, Gaussian)

$$k(\mathbf{x}, \mathbf{x}') = \alpha \exp\left(-\frac{\|\mathbf{x} - \mathbf{x}'\|_2^2}{2\ell^2}\right)$$

- Covariance matrix is built using the *inputs* to the function $\mathbf{x}$.
- For the example above it was based on Euclidean distance.
- The covariance function is also know as a kernel.



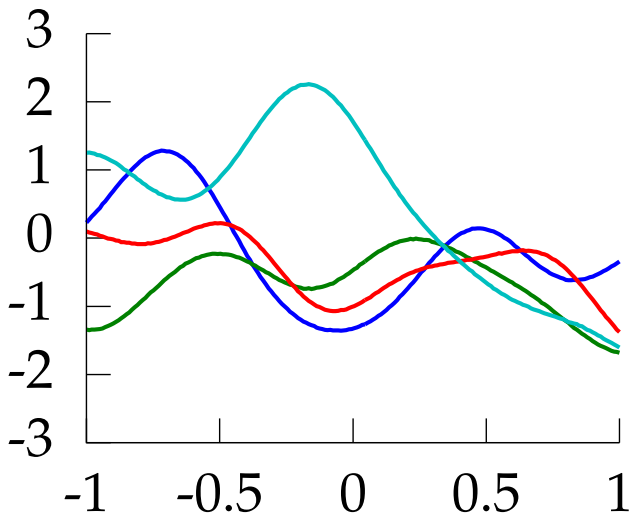
Covariance Functions

Where did this covariance matrix come from?

Exponentiated Quadratic Kernel Function (RBF, Squared Exponential, Gaussian)

$$k(\mathbf{x}, \mathbf{x}') = \alpha \exp\left(-\frac{\|\mathbf{x} - \mathbf{x}'\|_2^2}{2\ell^2}\right)$$

- Covariance matrix is built using the *inputs* to the function $\mathbf{x}$.
- For the example above it was based on Euclidean distance.
- The covariance function is also know as a kernel.



Gaussian Process Interpolation

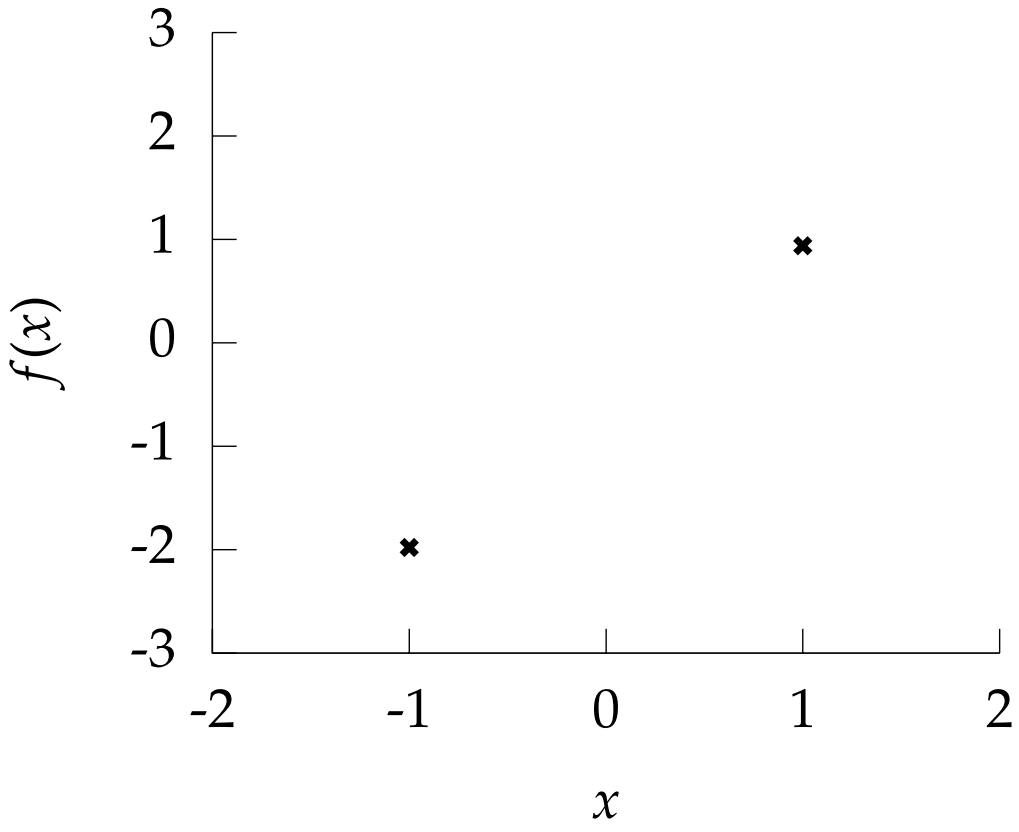


Figure : Real example: BACCO (see *e.g.* (Oakley and O'Hagan, 2002)). Interpolation through outputs from slow computer simulations (*e.g.* atmospheric carbon levels).

Gaussian Process Interpolation

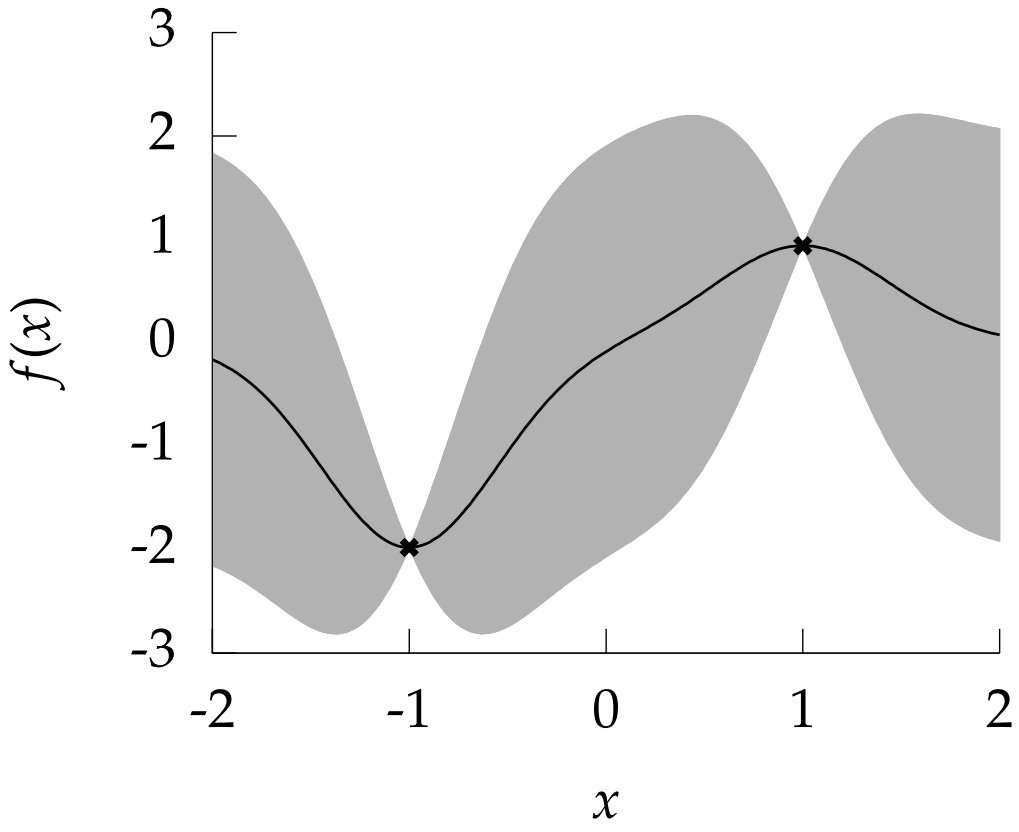


Figure : Real example: BACCO (see *e.g.* (Oakley and O'Hagan, 2002)). Interpolation through outputs from slow computer simulations (*e.g.* atmospheric carbon levels).

Gaussian Process Interpolation

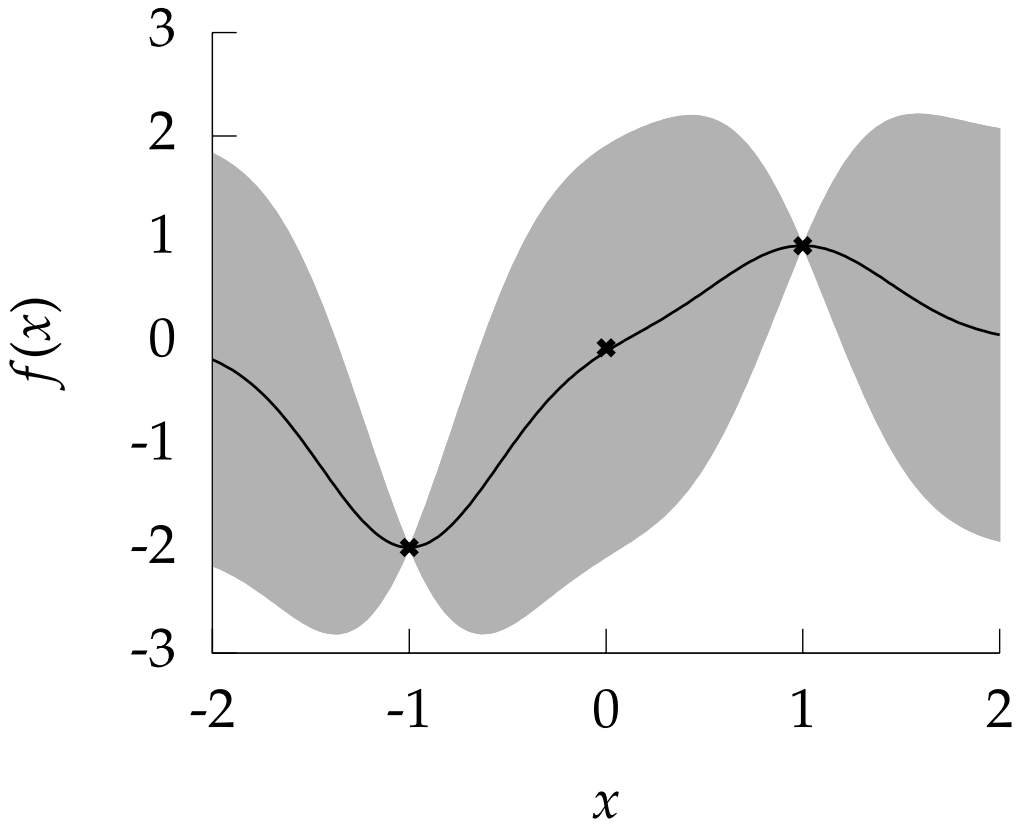


Figure : Real example: BACCO (see *e.g.* (Oakley and O'Hagan, 2002)). Interpolation through outputs from slow computer simulations (*e.g.* atmospheric carbon levels).

Gaussian Process Interpolation

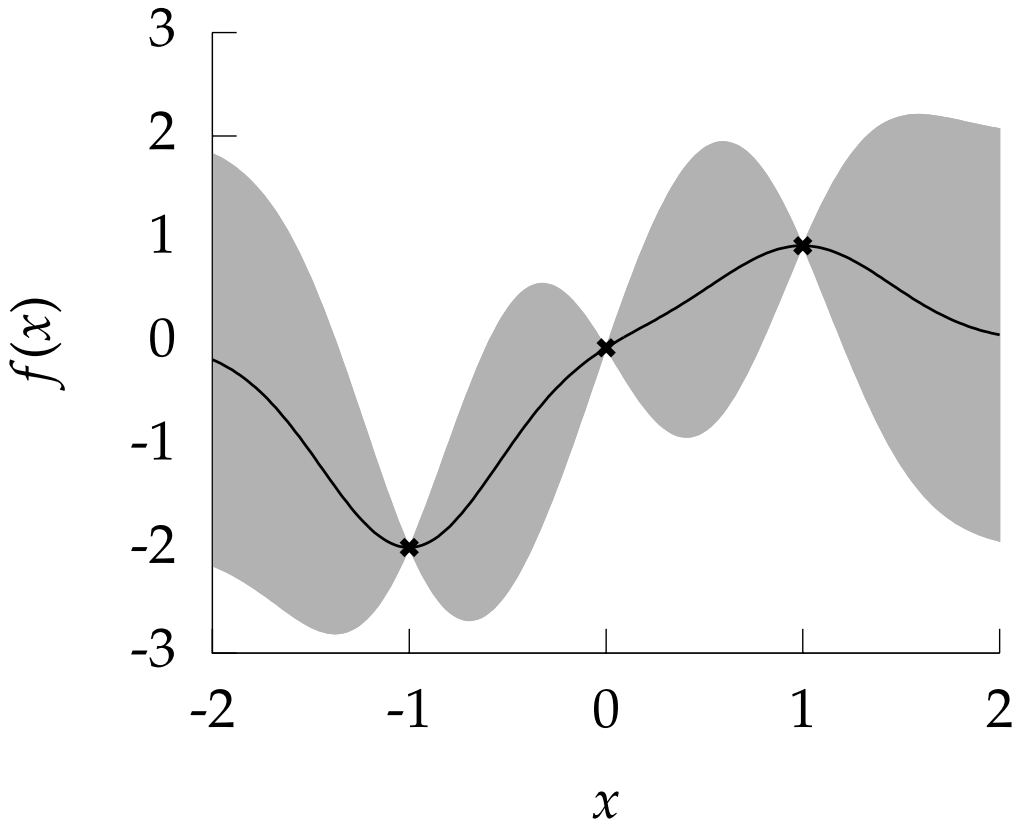


Figure : Real example: BACCO (see *e.g.* (Oakley and O'Hagan, 2002)). Interpolation through outputs from slow computer simulations (*e.g.* atmospheric carbon levels).

Gaussian Process Interpolation

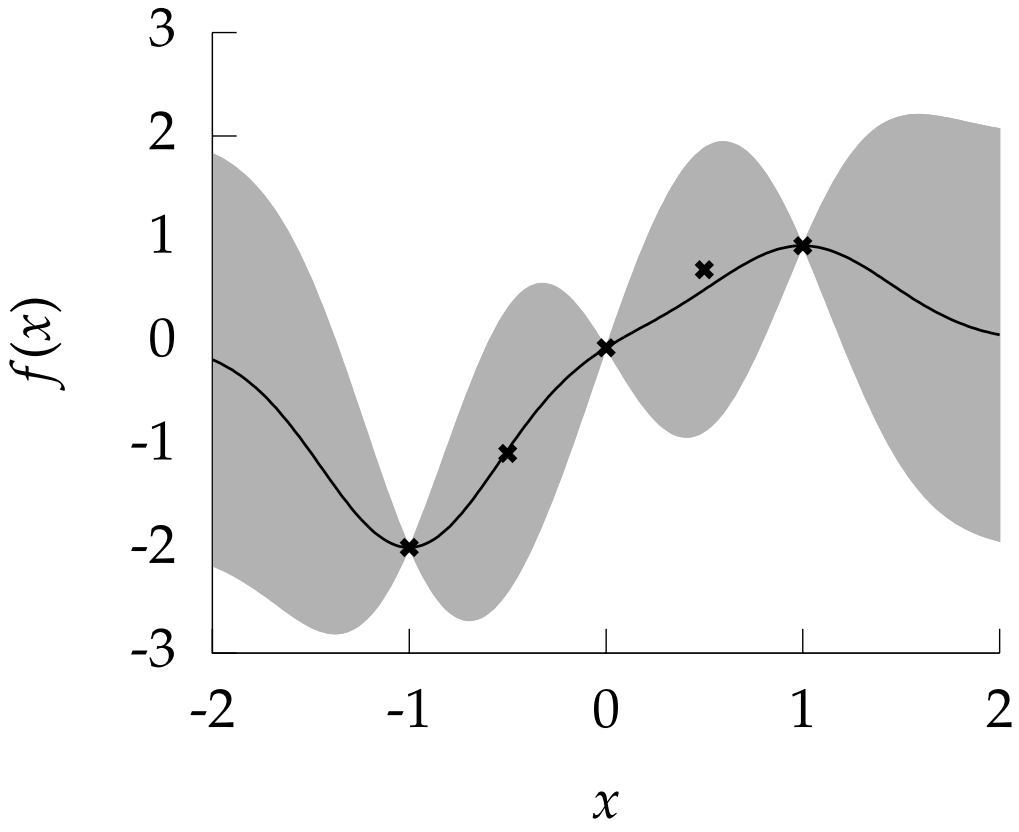


Figure : Real example: BACCO (see *e.g.* (Oakley and O'Hagan, 2002)). Interpolation through outputs from slow computer simulations (*e.g.* atmospheric carbon levels).

Gaussian Process Interpolation

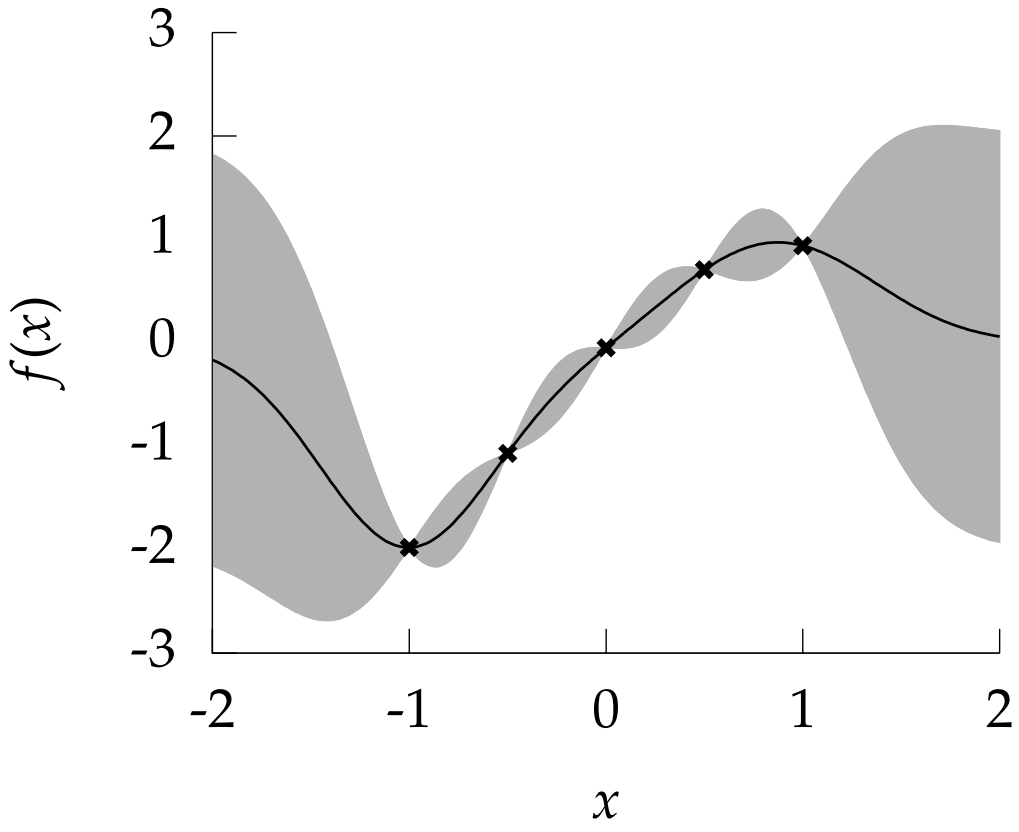


Figure : Real example: BACCO (see *e.g.* (Oakley and O'Hagan, 2002)). Interpolation through outputs from slow computer simulations (*e.g.* atmospheric carbon levels).

Gaussian Process Interpolation

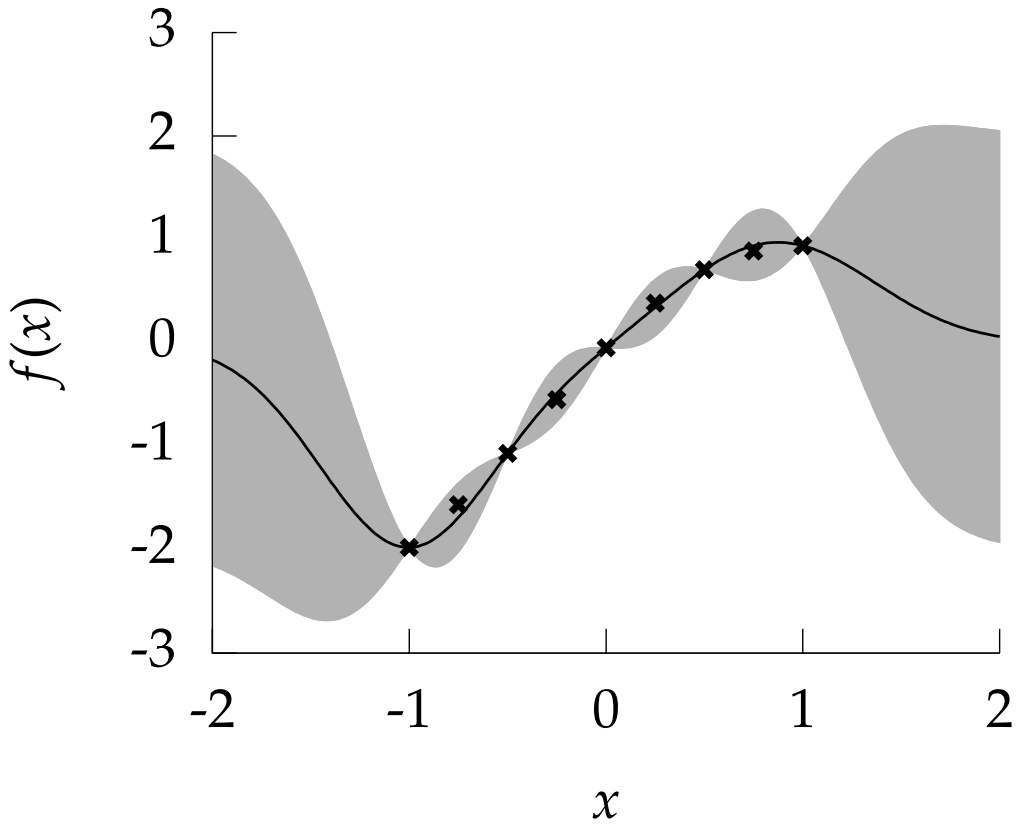


Figure : Real example: BACCO (see *e.g.* (Oakley and O'Hagan, 2002)). Interpolation through outputs from slow computer simulations (*e.g.* atmospheric carbon levels).

Gaussian Process Interpolation

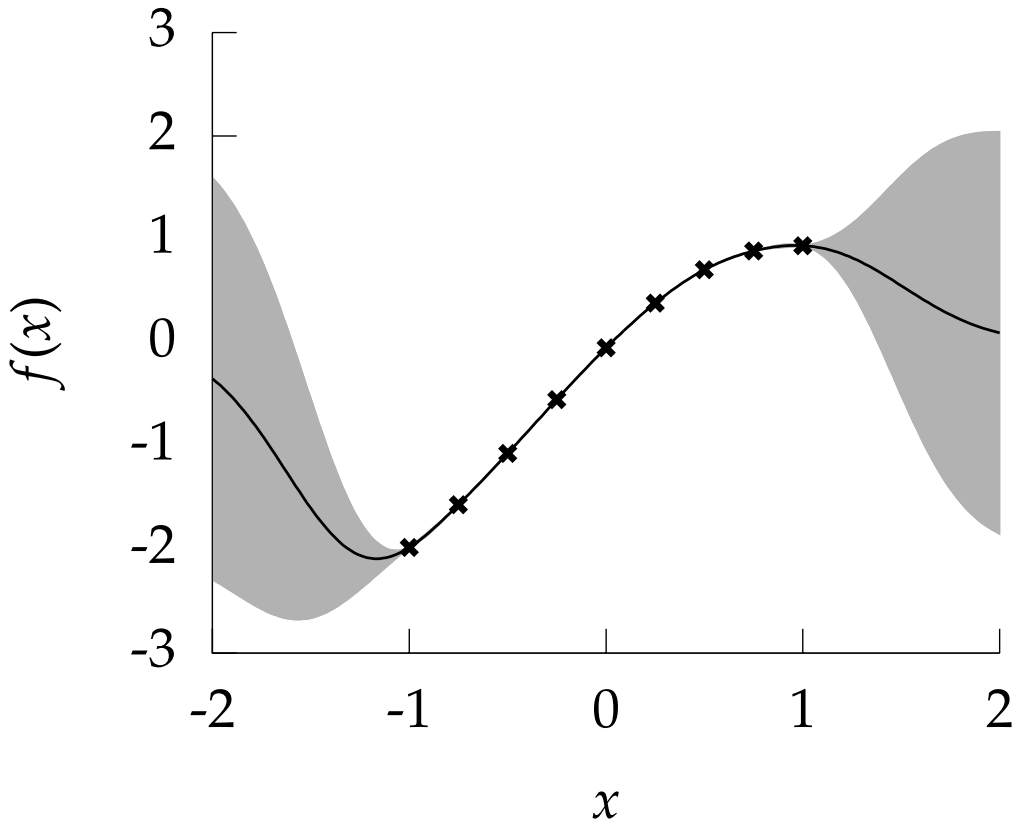


Figure : Real example: BACCO (see *e.g.* (Oakley and O'Hagan, 2002)). Interpolation through outputs from slow computer simulations (*e.g.* atmospheric carbon levels).

Gaussian Noise

- Gaussian noise model,

$$p\left(y_i|f_i\right) = \mathcal{N}\left(y_i|f_i, \sigma^2\right)$$

  where σ^2 is the variance of the noise.
- Equivalent to a covariance function of the form

$$k(\mathbf{x}_i, \mathbf{x}_j) = \delta_{i,j}\sigma^2$$

  where $\delta_{i,j}$ is the Kronecker delta function.
- Additive nature of Gaussians means we can simply add this term to existing covariance matrices.

Gaussian Process Regression

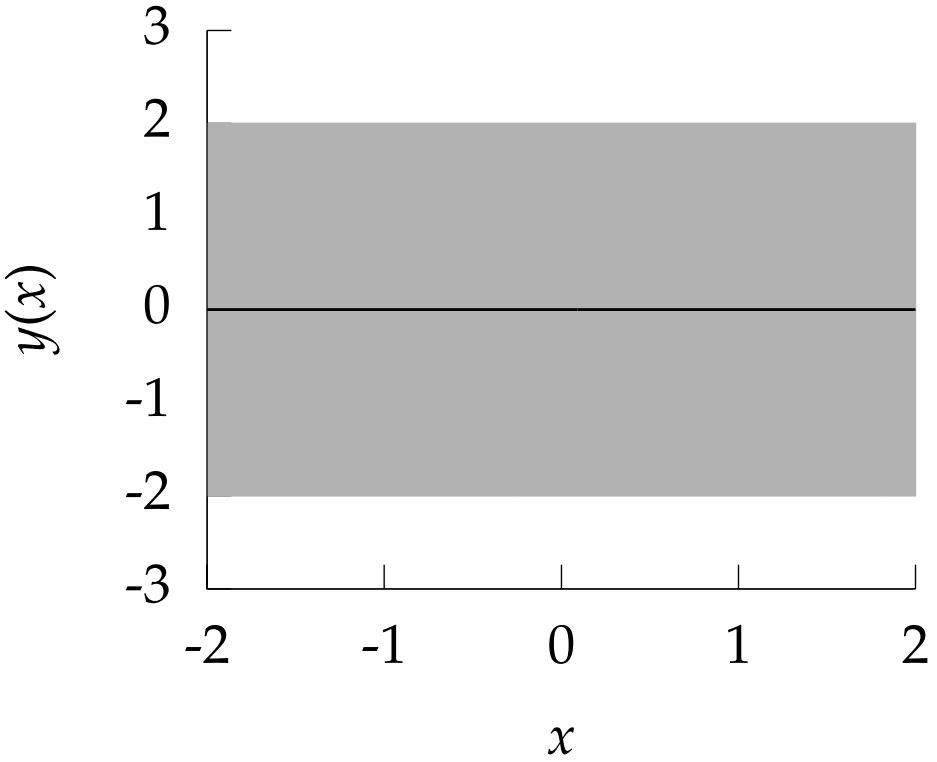


Figure : Examples include WiFi localization, C14 callibration curve.

Gaussian Process Regression

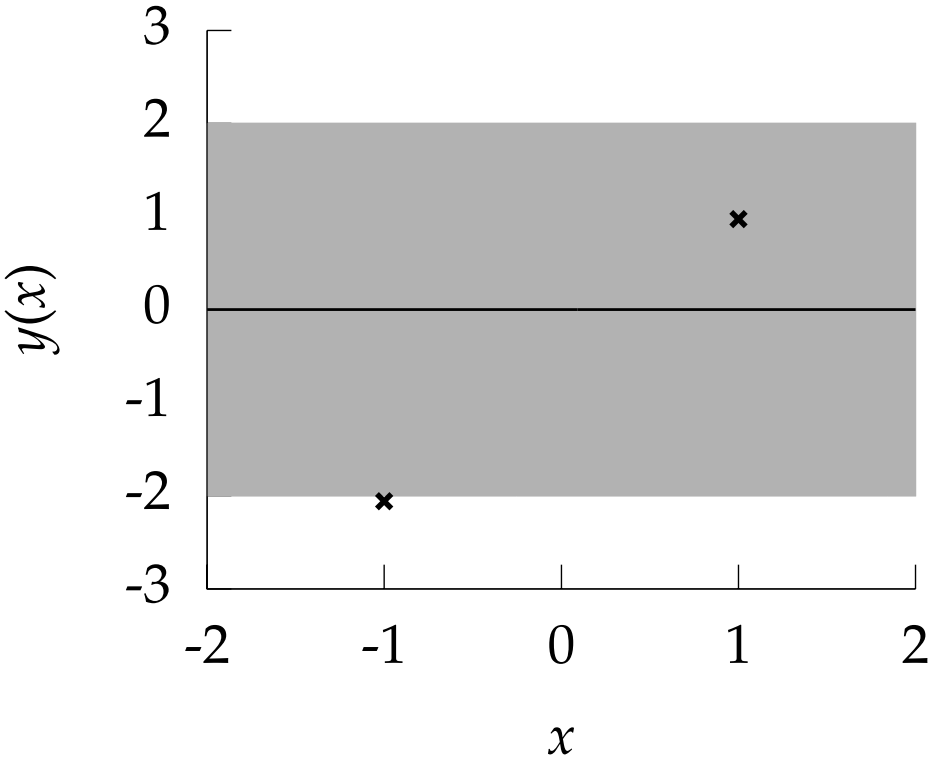


Figure : Examples include WiFi localization, C14 callibration curve.

Gaussian Process Regression

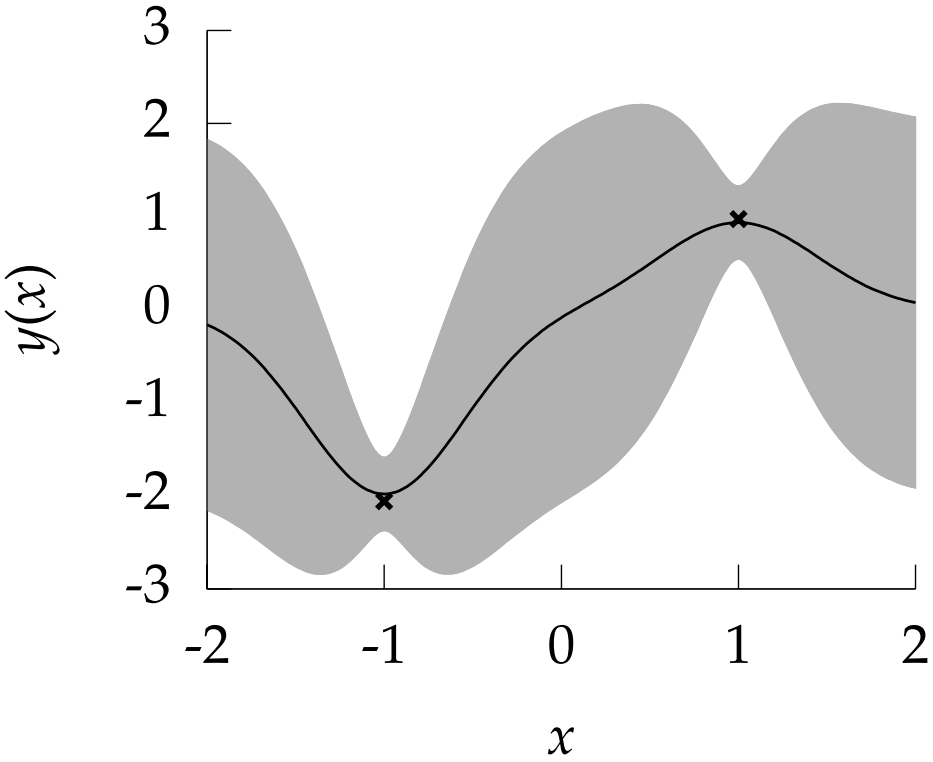


Figure : Examples include WiFi localization, C14 callibration curve.

Gaussian Process Regression

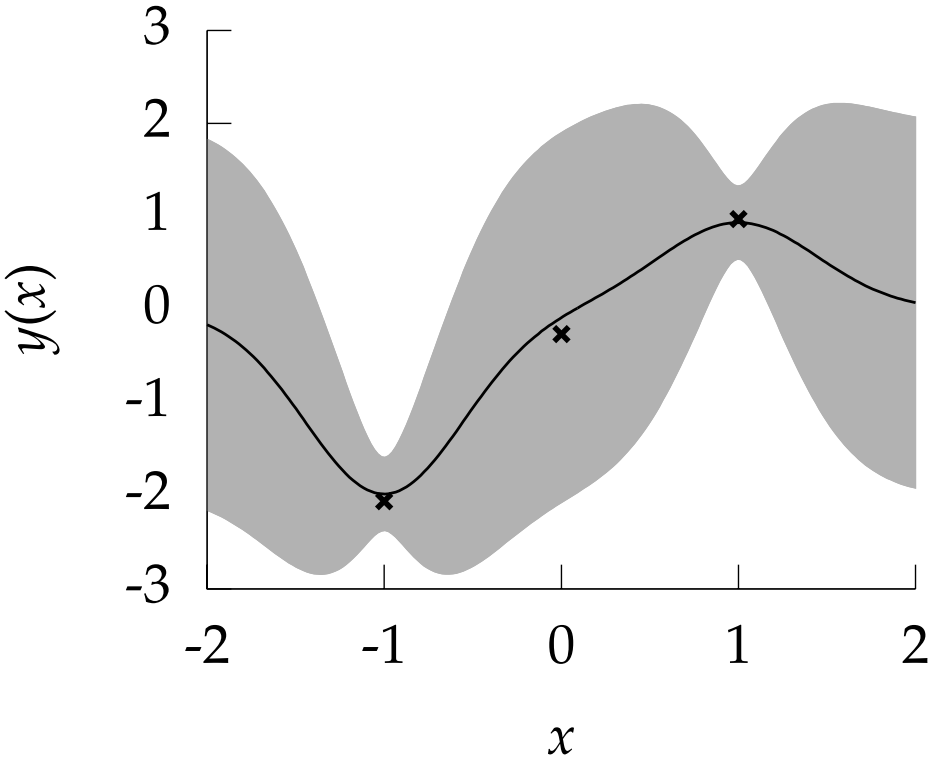


Figure : Examples include WiFi localization, C14 callibration curve.

Gaussian Process Regression

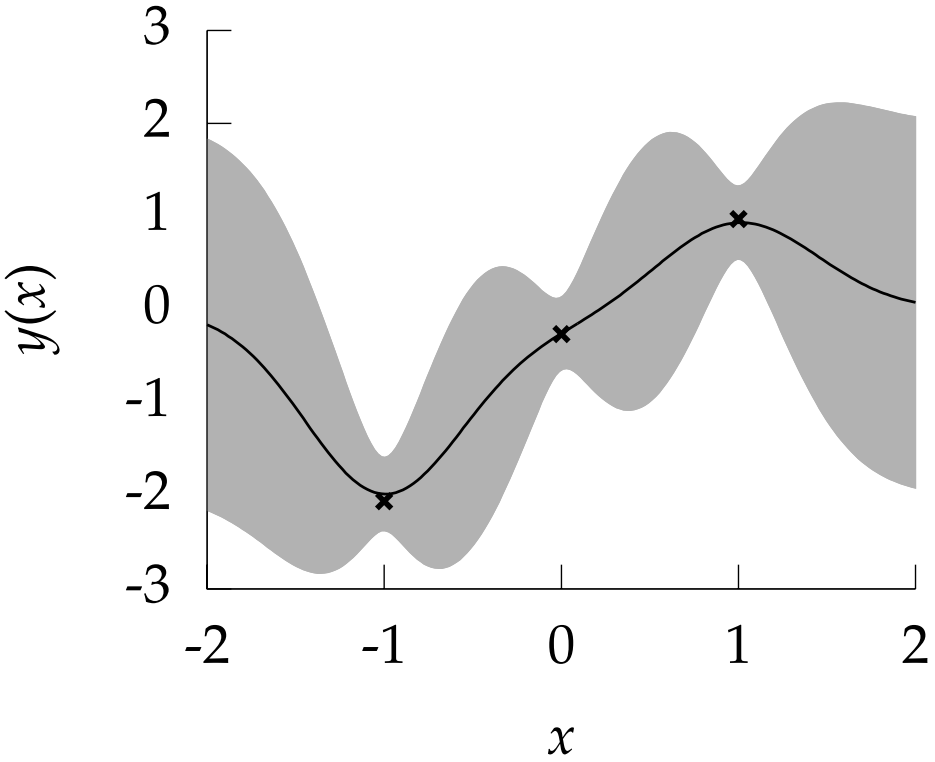


Figure : Examples include WiFi localization, C14 callibration curve.

Gaussian Process Regression

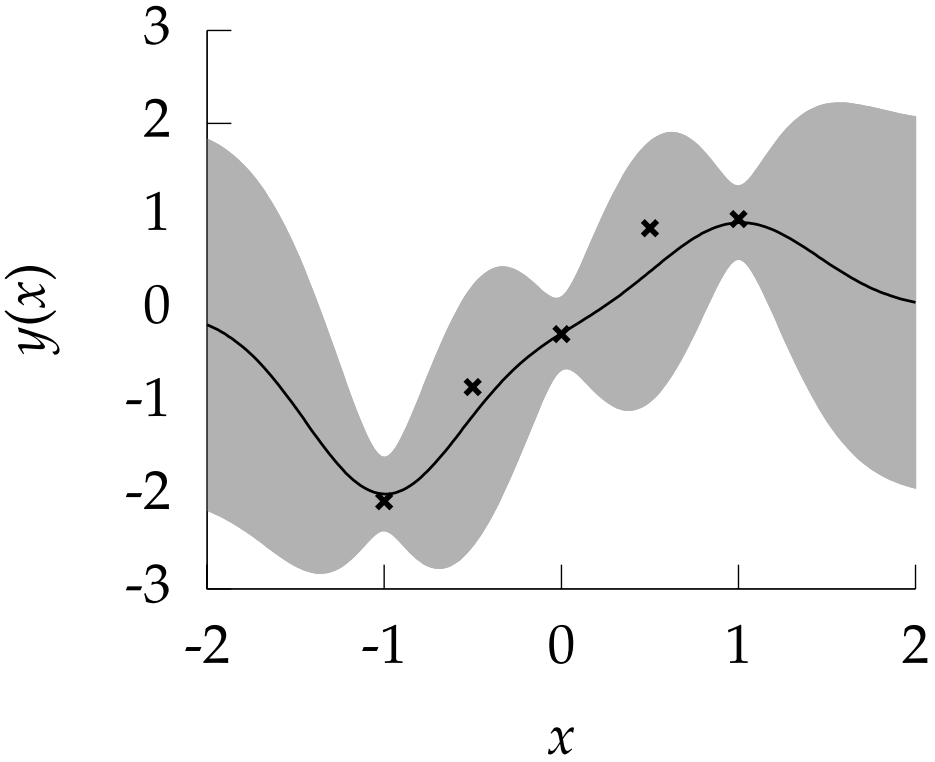


Figure : Examples include WiFi localization, C14 callibration curve.

Gaussian Process Regression

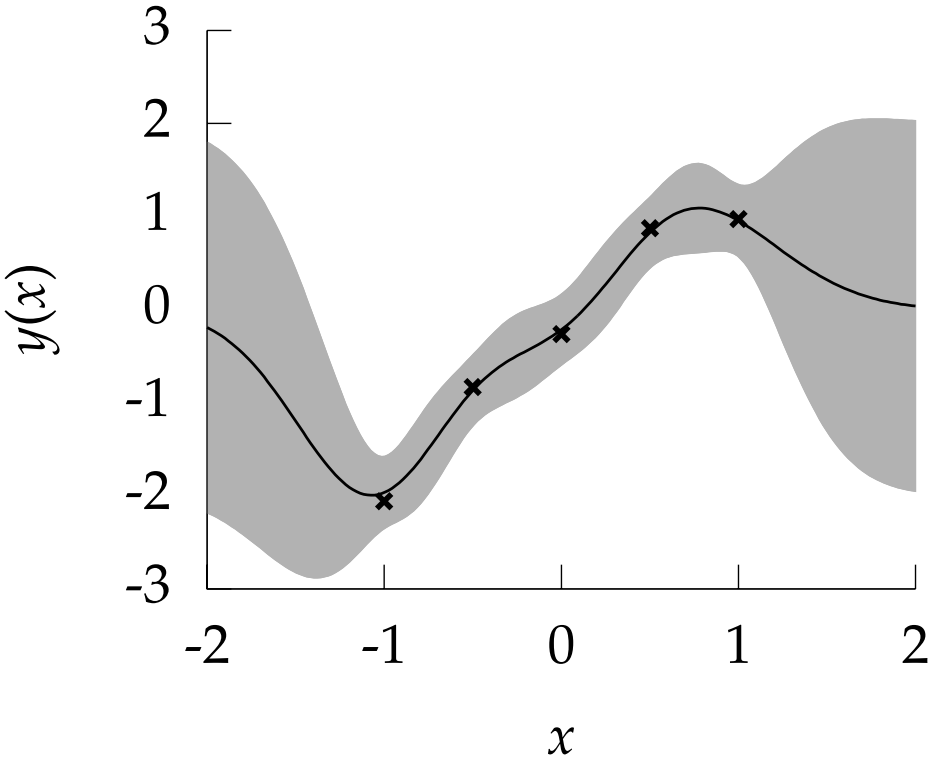


Figure : Examples include WiFi localization, C14 callibration curve.

Gaussian Process Regression

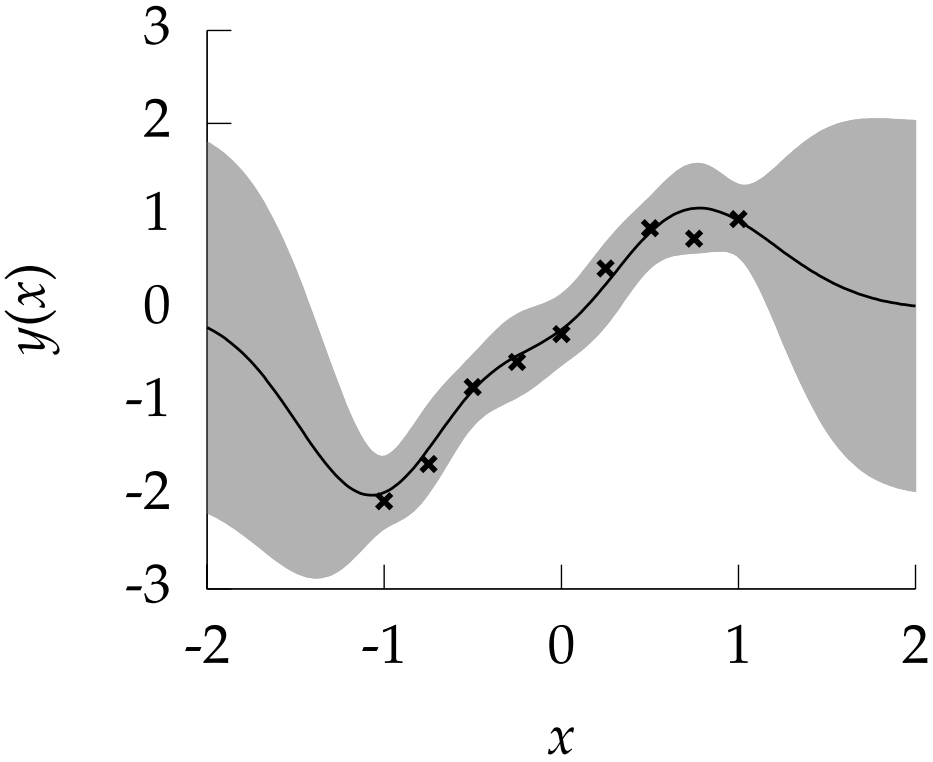


Figure : Examples include WiFi localization, C14 callibration curve.

Gaussian Process Regression

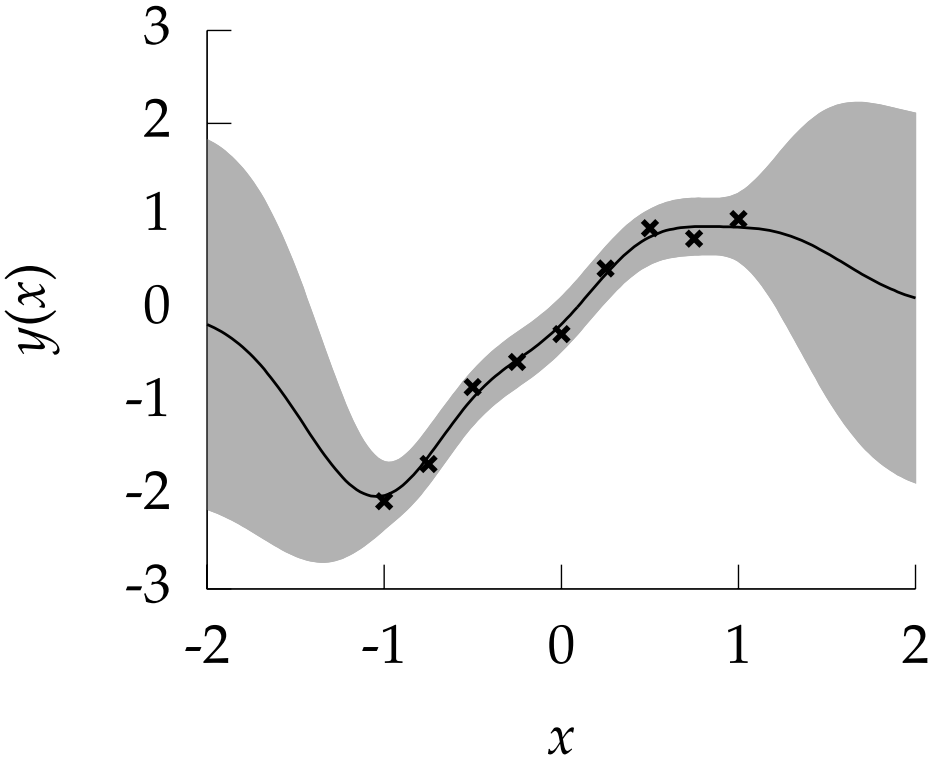


Figure : Examples include WiFi localization, C14 callibration curve.

Learning Covariance Parameters

Can we determine covariance parameters from the data?

$$\mathcal{N}\left(\mathbf{y}|\mathbf{0}, \mathbf{K}\right) = \frac{1}{(2\pi)^{\frac{n}{2}}|\mathbf{K}|^{\frac{1}{2}}}\exp\left(-\frac{\mathbf{y}^\top\mathbf{K}^{-1}\mathbf{y}}{2}\right)$$

The parameters are *inside* the covariance function (matrix).

$$k_{i,j} = k(\mathbf{x}_i, \mathbf{x}_j; \boldsymbol{\theta})$$

Learning Covariance Parameters

Can we determine covariance parameters from the data?

$$\mathcal{N}\left(\mathbf{y}|\mathbf{0},\mathbf{K}\right) = \frac{1}{(2\pi)^{\frac{n}{2}}|\mathbf{K}|^{\frac{1}{2}}}\exp\left(-\frac{\mathbf{y}^\top\mathbf{K}^{-1}\mathbf{y}}{2}\right)$$

The parameters are *inside* the covariance function (matrix).

$$k_{i,j} = k(\mathbf{x}_i, \mathbf{x}_j; \boldsymbol{\theta})$$

Learning Covariance Parameters

Can we determine covariance parameters from the data?

$$\log \mathcal{N}\left(\mathbf{y}|\mathbf{0}, \mathbf{K}\right) = -\frac{1}{2}\log|\mathbf{K}| - \frac{\mathbf{y}^\top\mathbf{K}^{-1}\mathbf{y}}{2} - \frac{n}{2}\log 2\pi$$

The parameters are *inside* the covariance function (matrix).

$$k_{i,j} = k(\mathbf{x}_i, \mathbf{x}_j; \boldsymbol{\theta})$$

Learning Covariance Parameters

Can we determine covariance parameters from the data?

$$E(\boldsymbol{\theta}) = \frac{1}{2}\log|\mathbf{K}| + \frac{\mathbf{y}^\top\mathbf{K}^{-1}\mathbf{y}}{2}$$

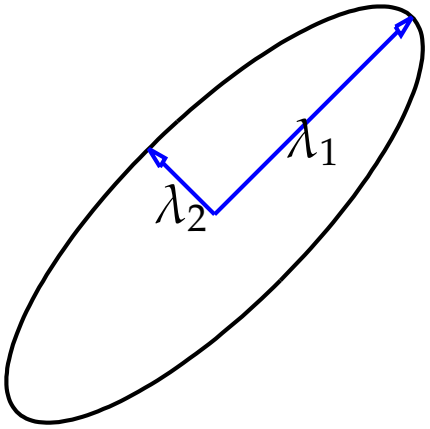
The parameters are *inside* the covariance function (matrix).

$$k_{i,j} = k(\mathbf{x}_i, \mathbf{x}_j; \boldsymbol{\theta})$$

Eigendecomposition of Covariance

A useful decomposition for understanding the objective function.

$$\mathbf{K} = \mathbf{R}\boldsymbol{\Lambda}^2\mathbf{R}^\top$$

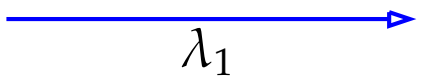


Diagonal of $\boldsymbol{\Lambda}$ represents distance along axes.
$\mathbf{R}$ gives a rotation of these axes.

where $\boldsymbol{\Lambda}$ is a *diagonal* matrix and $\mathbf{R}^\top\mathbf{R} = \mathbf{I}$.

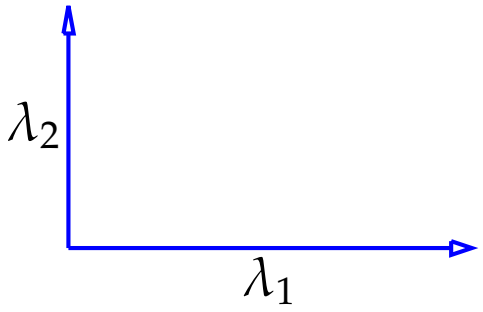
Capacity control: $\log |\mathbf{K}|$

$$\mathbf{\Lambda} = \begin{bmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{bmatrix}$$



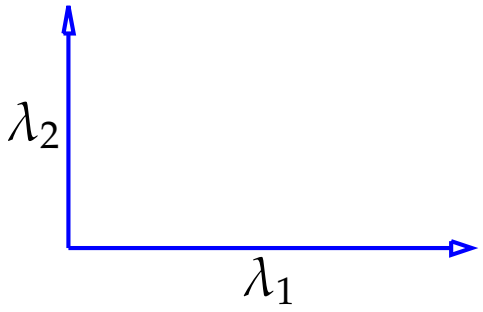
Capacity control: $\log |\mathbf{K}|$

$$\mathbf{\Lambda} = \begin{bmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{bmatrix}$$



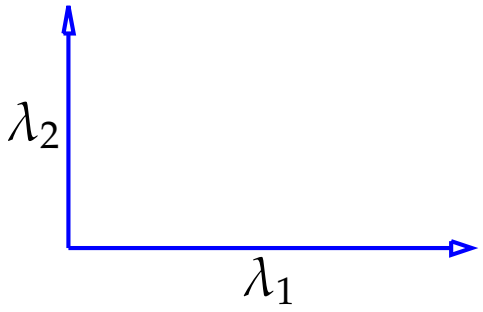
Capacity control: $\log |\mathbf{K}|$

$$\mathbf{\Lambda} = \begin{bmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{bmatrix}$$



Capacity control: $\log |\mathbf{K}|$

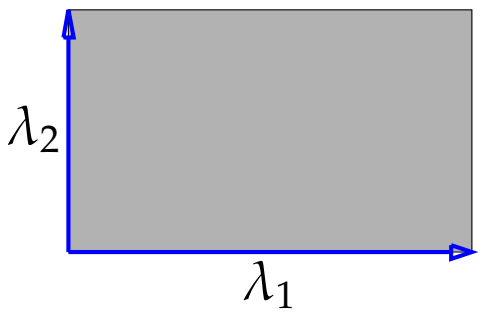
$$\mathbf{\Lambda} = \begin{bmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{bmatrix}$$



$$|\mathbf{\Lambda}| = \lambda_1 \lambda_2$$

Capacity control: $\log |\mathbf{K}|$

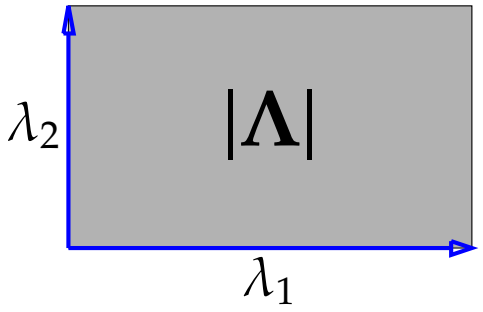
$$\mathbf{\Lambda} = \begin{bmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{bmatrix}$$



$$|\mathbf{\Lambda}| = \lambda_1 \lambda_2$$

Capacity control: $\log |\mathbf{K}|$

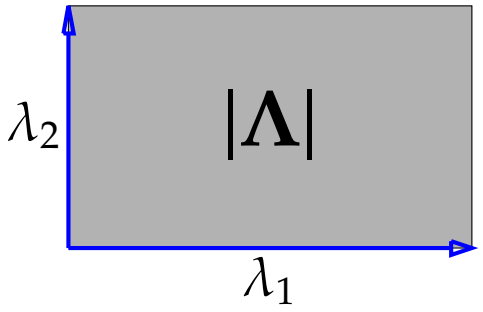
$$\mathbf{\Lambda} = \begin{bmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{bmatrix}$$



$$|\mathbf{\Lambda}| = \lambda_1 \lambda_2$$

Capacity control: $\log |\mathbf{K}|$

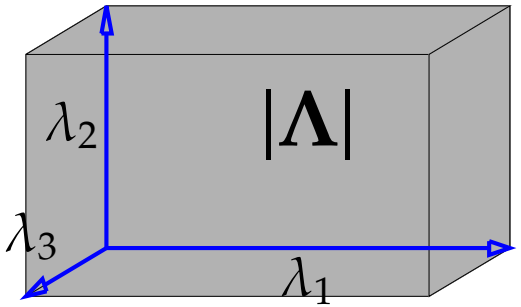
$$\mathbf{\Lambda} = \begin{bmatrix} \lambda_1 & 0 & 0 \\ 0 & \lambda_2 & 0 \\ 0 & 0 & \lambda_3 \end{bmatrix}$$



$$|\mathbf{\Lambda}| = \lambda_1 \lambda_2$$

Capacity control: $\log |\mathbf{K}|$

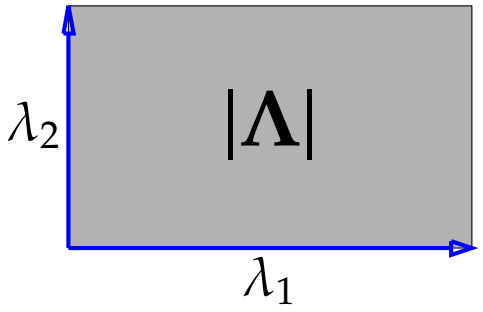
$$\mathbf{\Lambda} = \begin{bmatrix} \lambda_1 & 0 & 0 \\ 0 & \lambda_2 & 0 \\ 0 & 0 & \lambda_3 \end{bmatrix}$$



$$|\mathbf{\Lambda}| = \lambda_1 \lambda_2 \lambda_3$$

Capacity control: $\log |\mathbf{K}|$

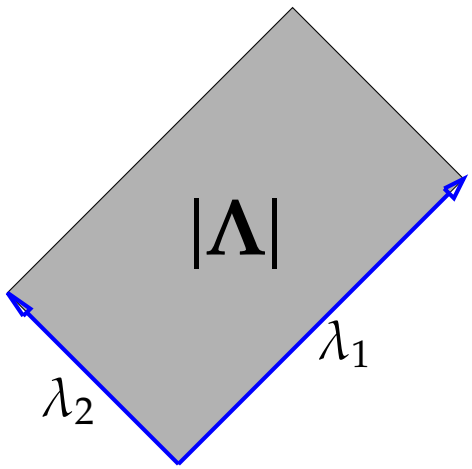
$$\mathbf{\Lambda} = \begin{bmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{bmatrix}$$



$$|\mathbf{\Lambda}| = \lambda_1 \lambda_2$$

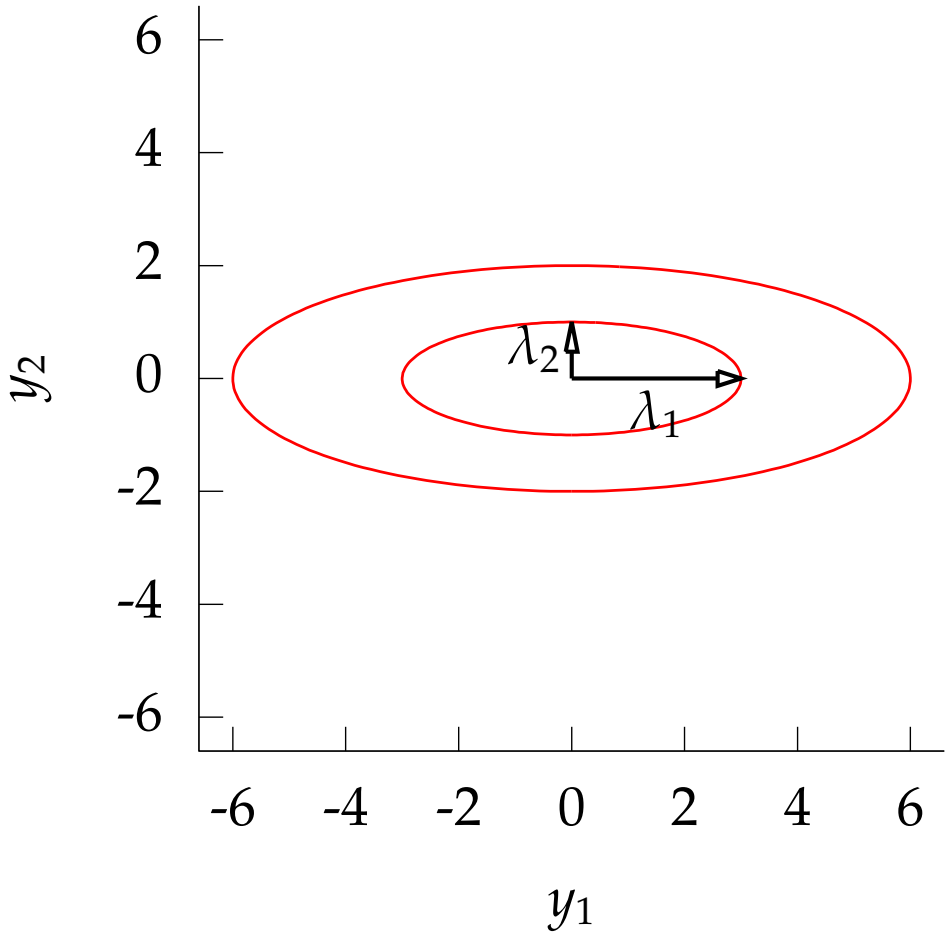
Capacity control: $\log |\mathbf{K}|$

$$\mathbf{R}\boldsymbol{\Lambda} = \begin{bmatrix} w_{1,1} & w_{1,2} \\ w_{2,1} & w_{2,2} \end{bmatrix}$$

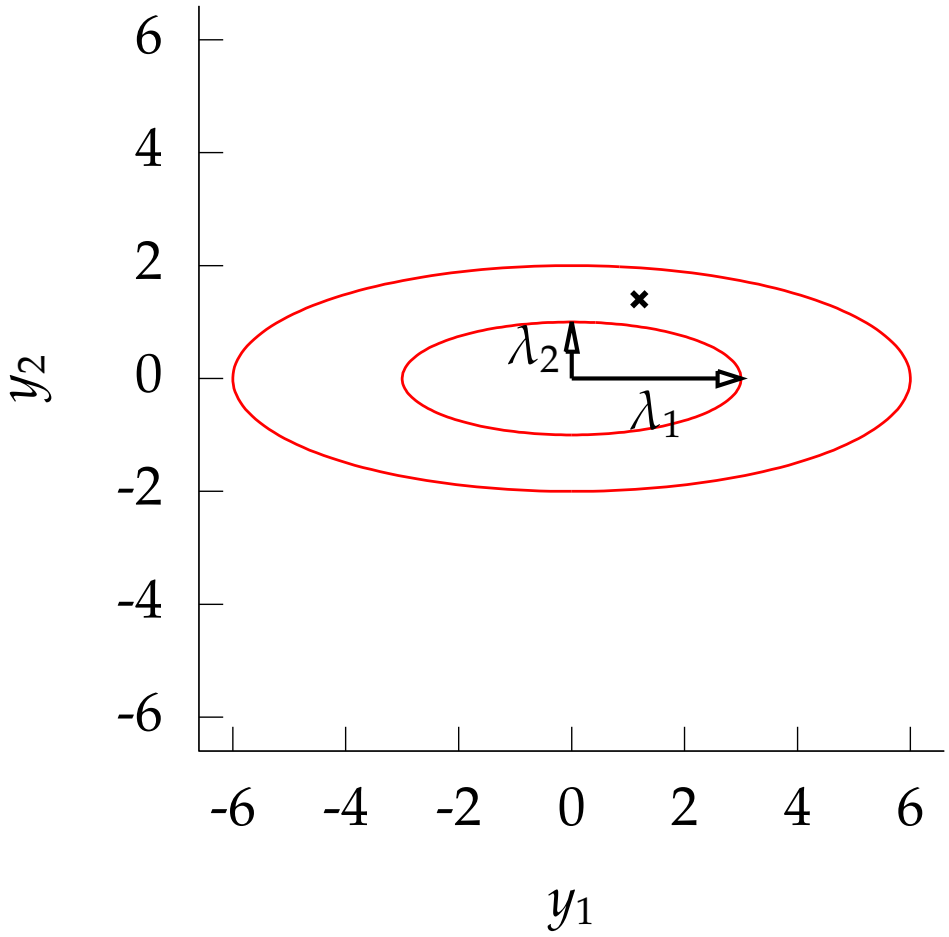


$$|\mathbf{R}\boldsymbol{\Lambda}| = \lambda_1 \lambda_2$$

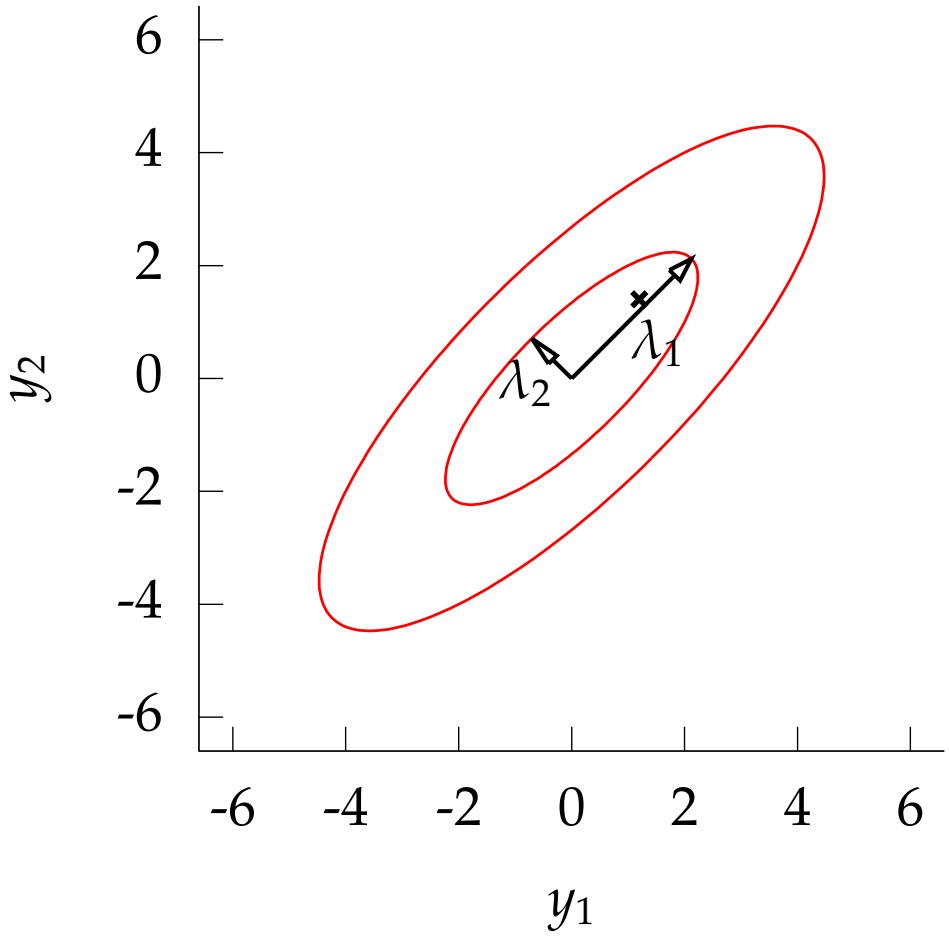
Data Fit: $\frac{\mathbf{y}^\top \mathbf{K}^{-1} \mathbf{y}}{2}$



Data Fit: $\frac{\mathbf{y}^\top \mathbf{K}^{-1} \mathbf{y}}{2}$

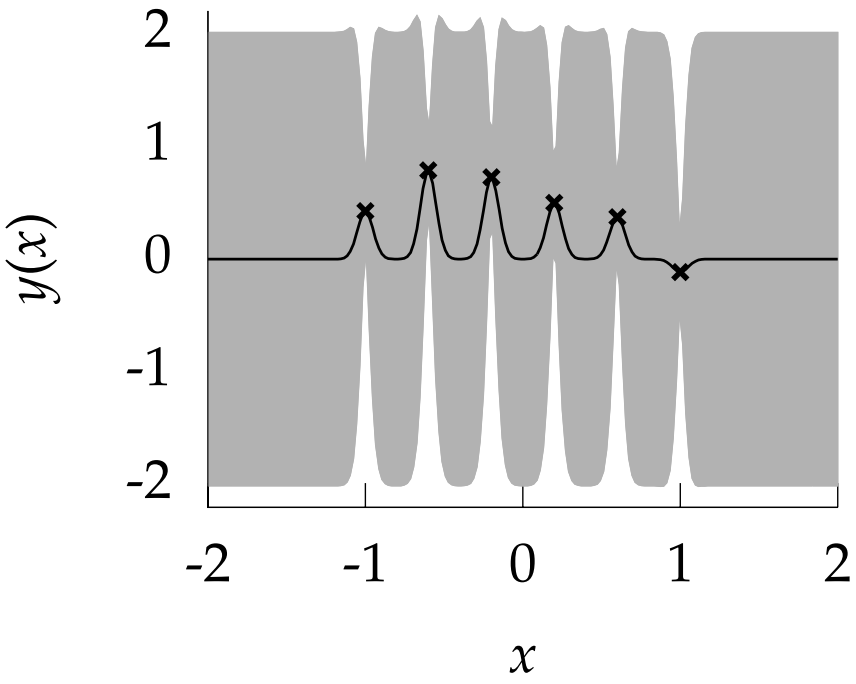


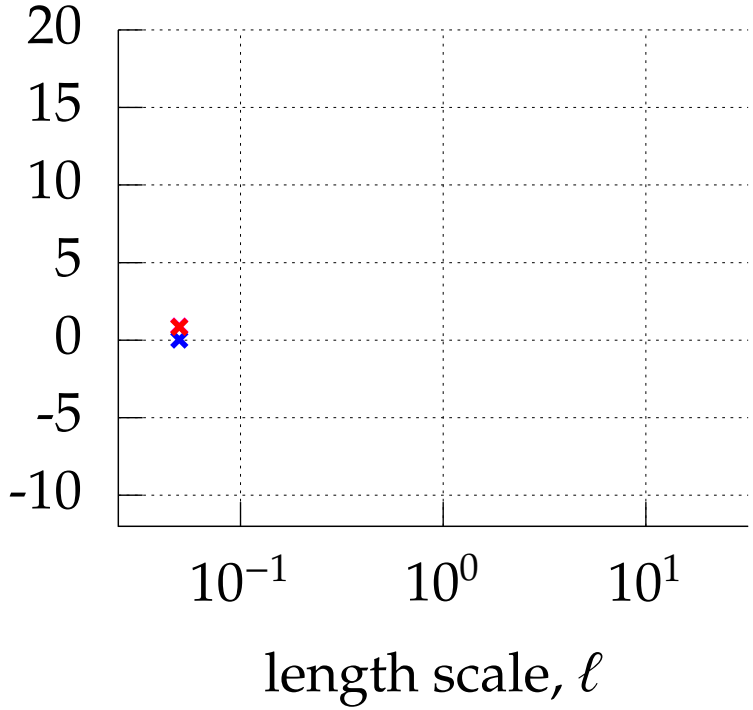
Data Fit: $\frac{\mathbf{y}^\top \mathbf{K}^{-1} \mathbf{y}}{2}$



Learning Covariance Parameters

Can we determine length scales and noise levels from the data?

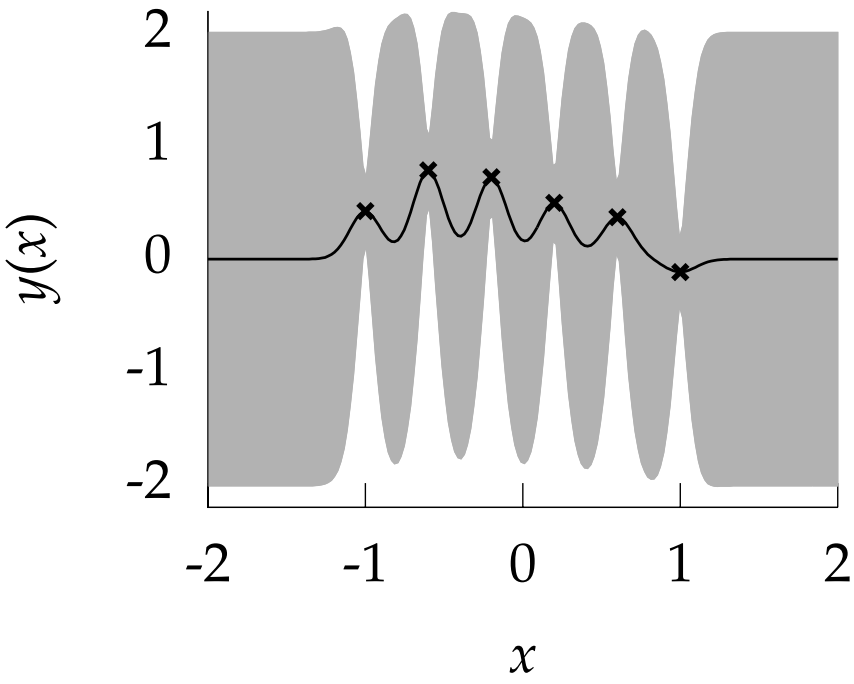


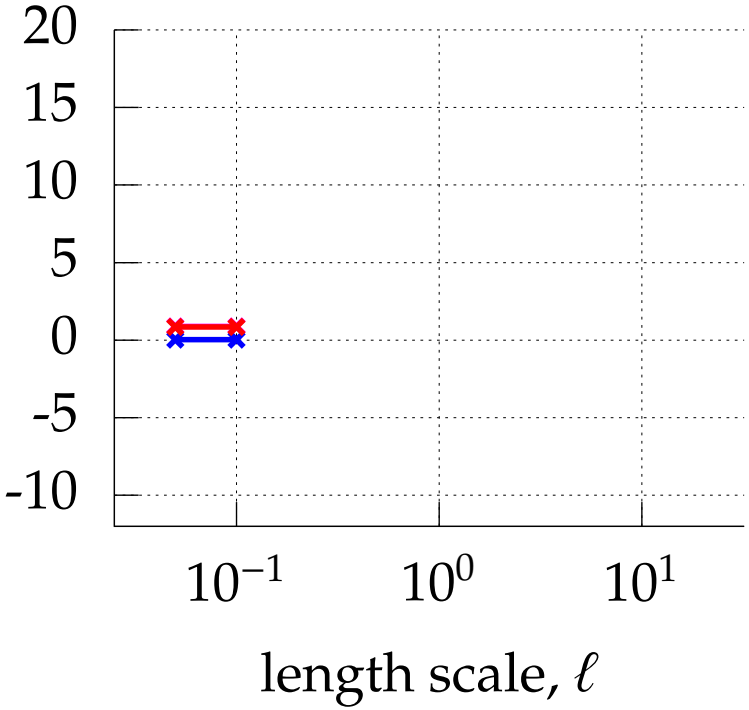


$$E(\boldsymbol{\theta}) = \frac{1}{2}\log|\mathbf{K}| + \frac{\mathbf{y}^\top \mathbf{K}^{-1}\mathbf{y}}{2}$$

Learning Covariance Parameters

Can we determine length scales and noise levels from the data?

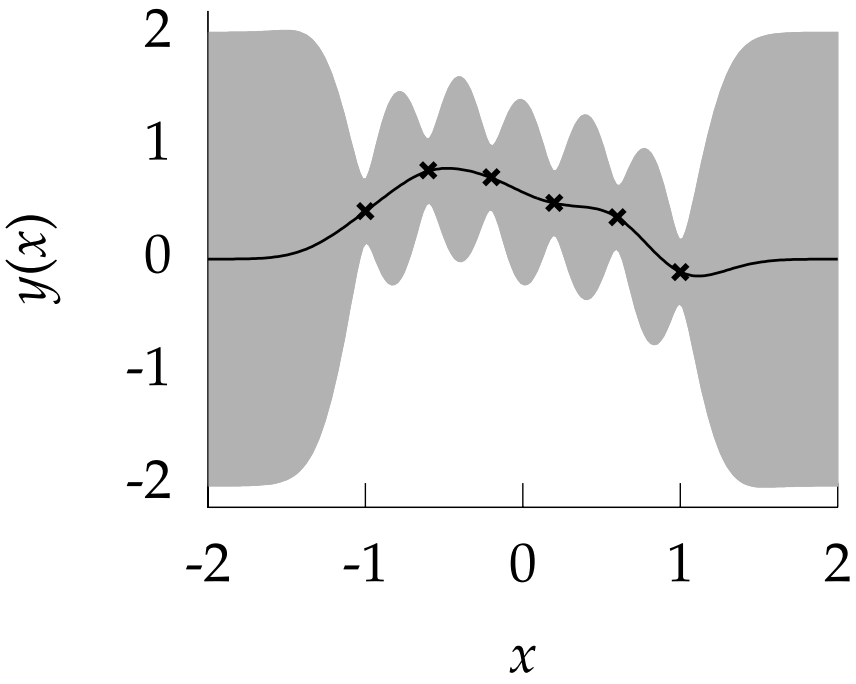


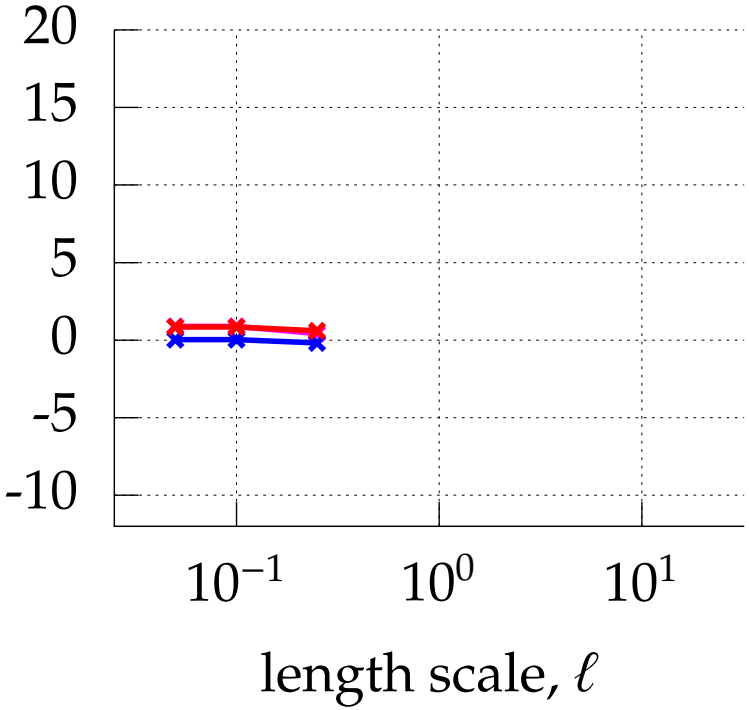


$$E(\boldsymbol{\theta}) = \frac{1}{2}\log|\mathbf{K}| + \frac{\mathbf{y}^\top\mathbf{K}^{-1}\mathbf{y}}{2}$$

Learning Covariance Parameters

Can we determine length scales and noise levels from the data?

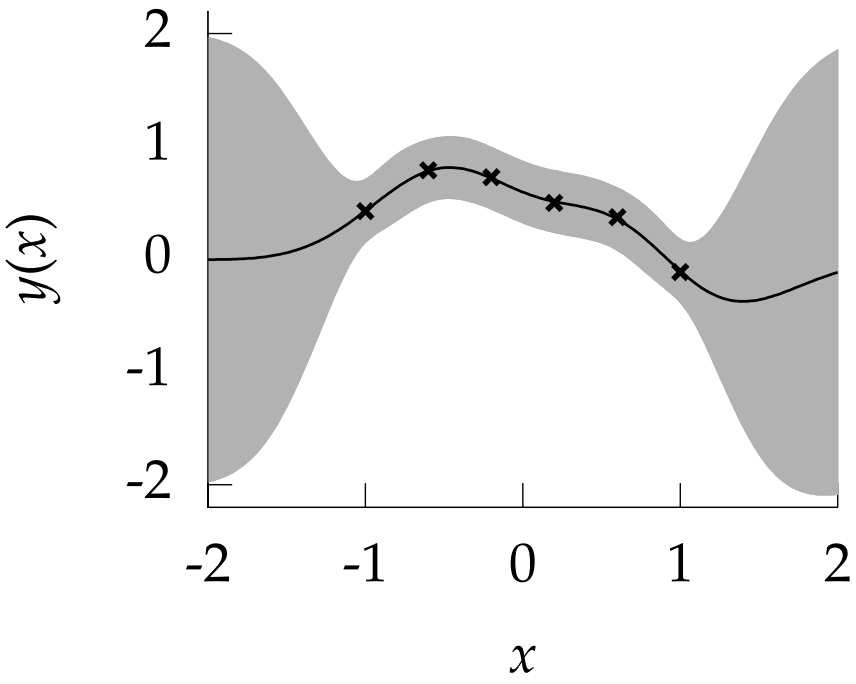


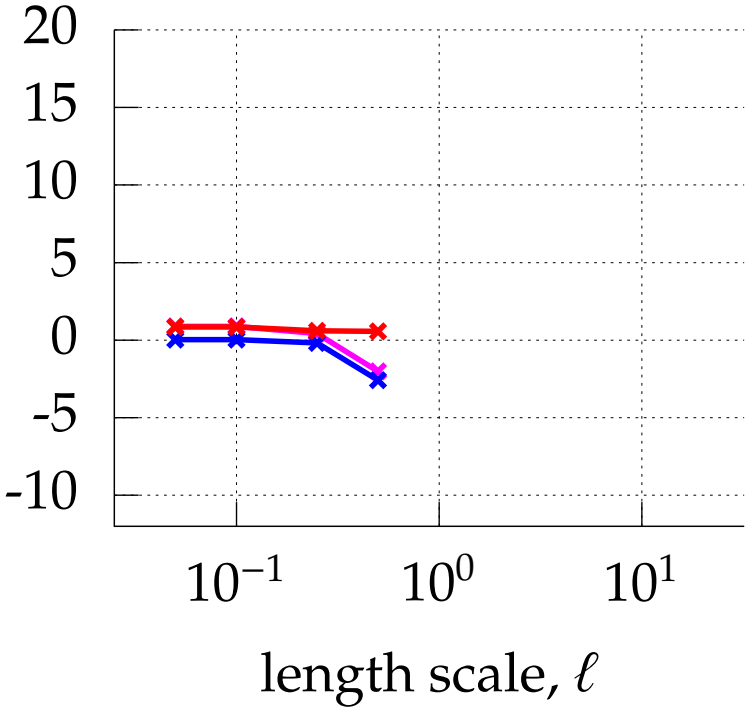


$$E(\boldsymbol{\theta}) = \frac{1}{2}\log|\mathbf{K}| + \frac{\mathbf{y}^\top\mathbf{K}^{-1}\mathbf{y}}{2}$$

Learning Covariance Parameters

Can we determine length scales and noise levels from the data?

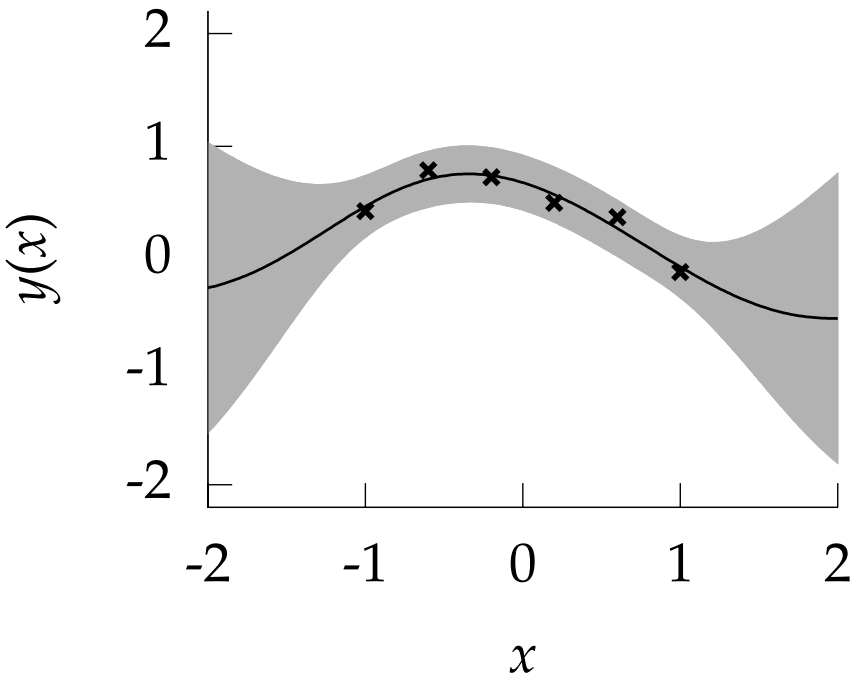


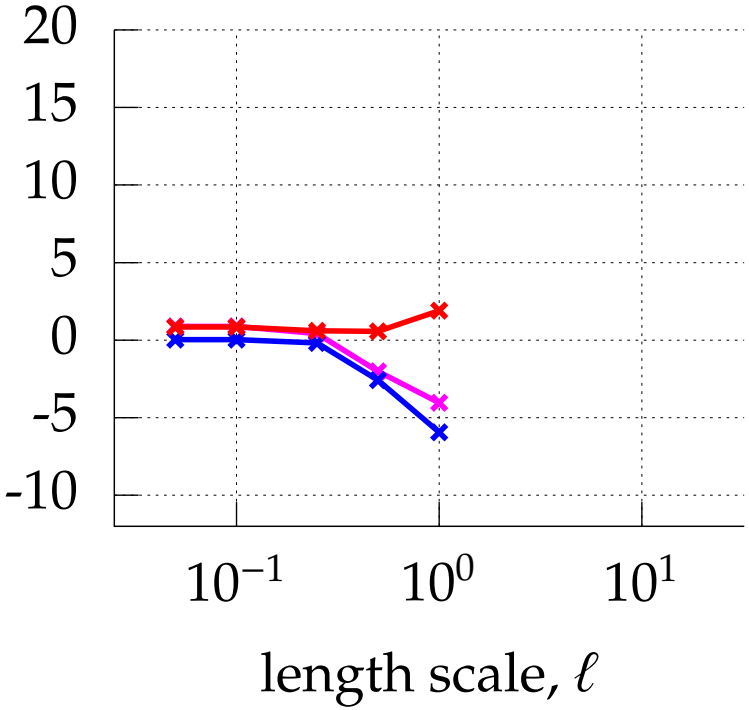


$$E(\boldsymbol{\theta}) = \frac{1}{2}\log|\mathbf{K}| + \frac{\mathbf{y}^\top \mathbf{K}^{-1}\mathbf{y}}{2}$$

Learning Covariance Parameters

Can we determine length scales and noise levels from the data?

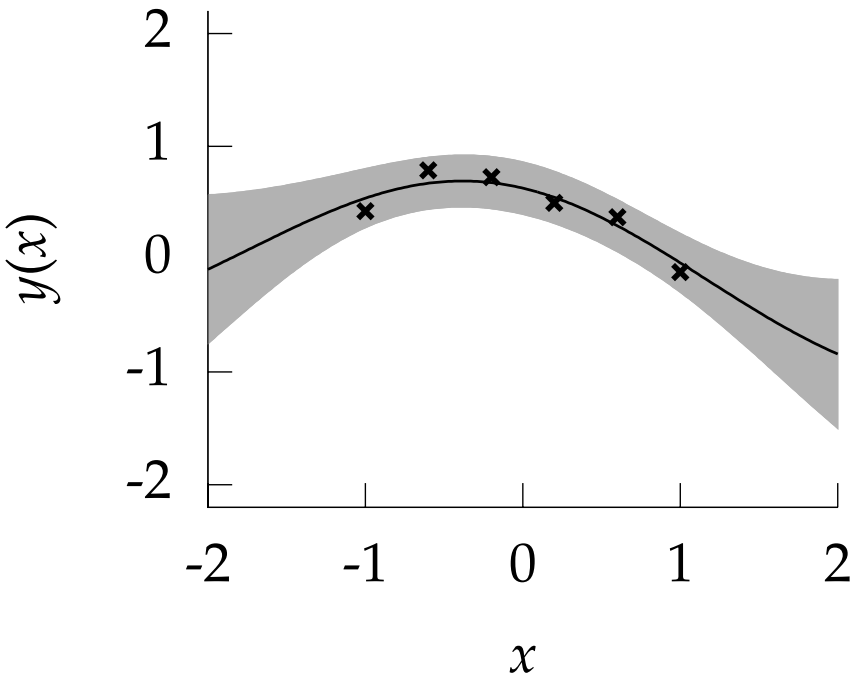


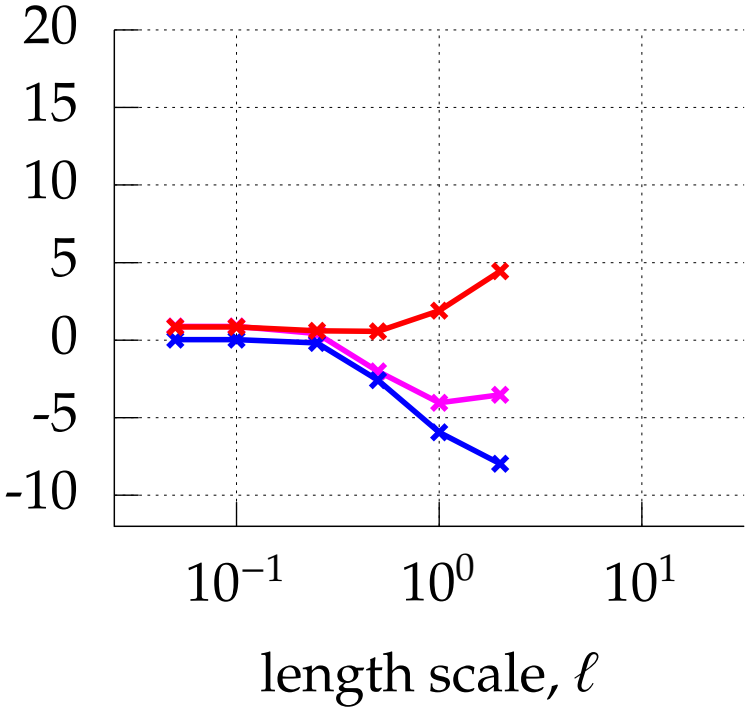


$$E(\boldsymbol{\theta}) = \frac{1}{2}\log|\mathbf{K}| + \frac{\mathbf{y}^\top \mathbf{K}^{-1}\mathbf{y}}{2}$$

Learning Covariance Parameters

Can we determine length scales and noise levels from the data?

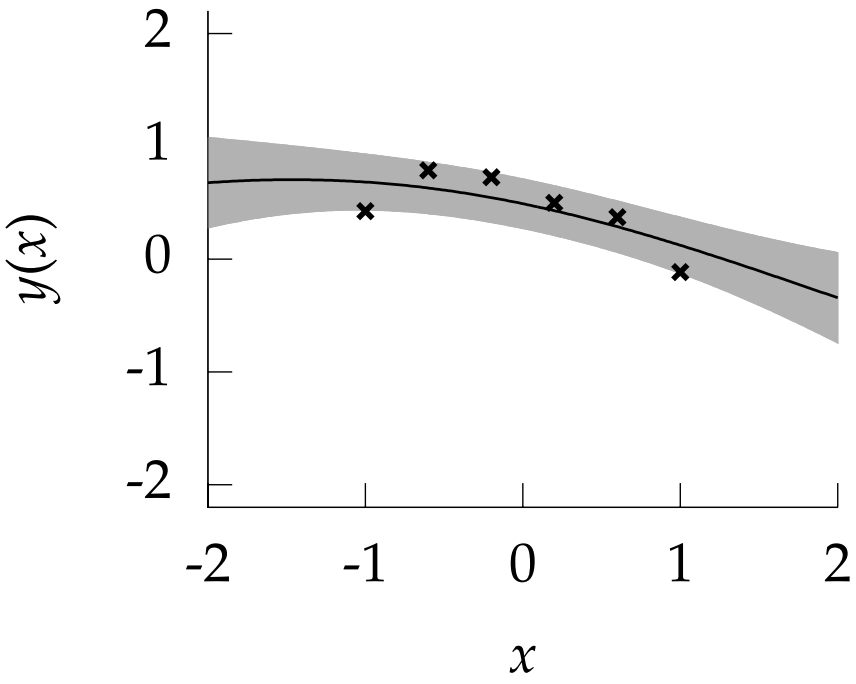


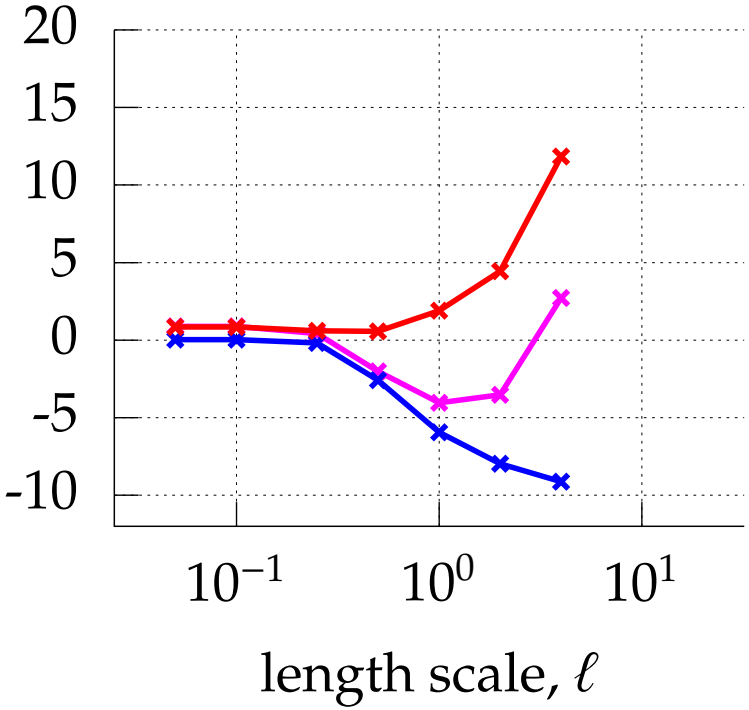


$$E(\boldsymbol{\theta}) = \frac{1}{2}\log|\mathbf{K}| + \frac{\mathbf{y}^\top \mathbf{K}^{-1}\mathbf{y}}{2}$$

Learning Covariance Parameters

Can we determine length scales and noise levels from the data?

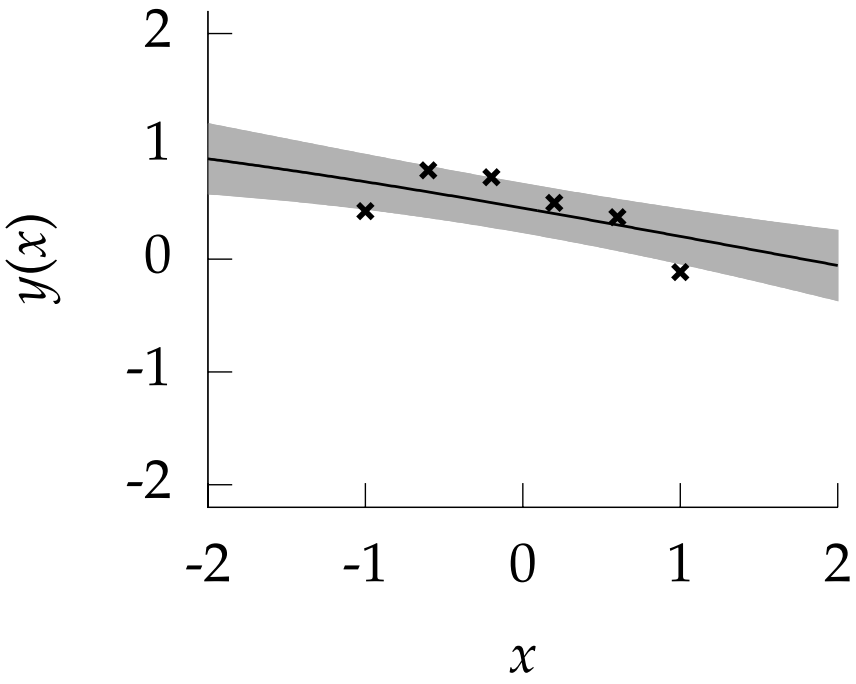


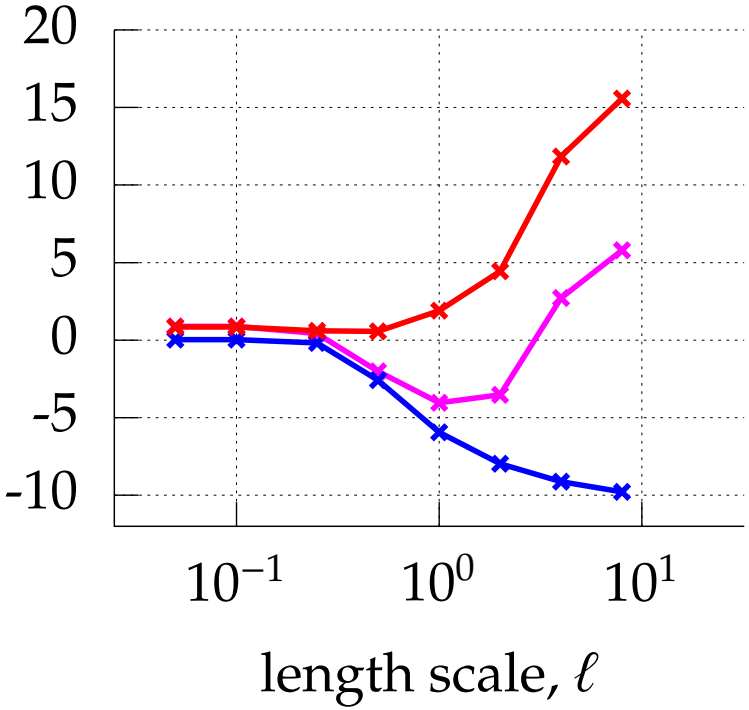


$$E(\boldsymbol{\theta}) = \frac{1}{2}\log|\mathbf{K}| + \frac{\mathbf{y}^\top \mathbf{K}^{-1}\mathbf{y}}{2}$$

Learning Covariance Parameters

Can we determine length scales and noise levels from the data?

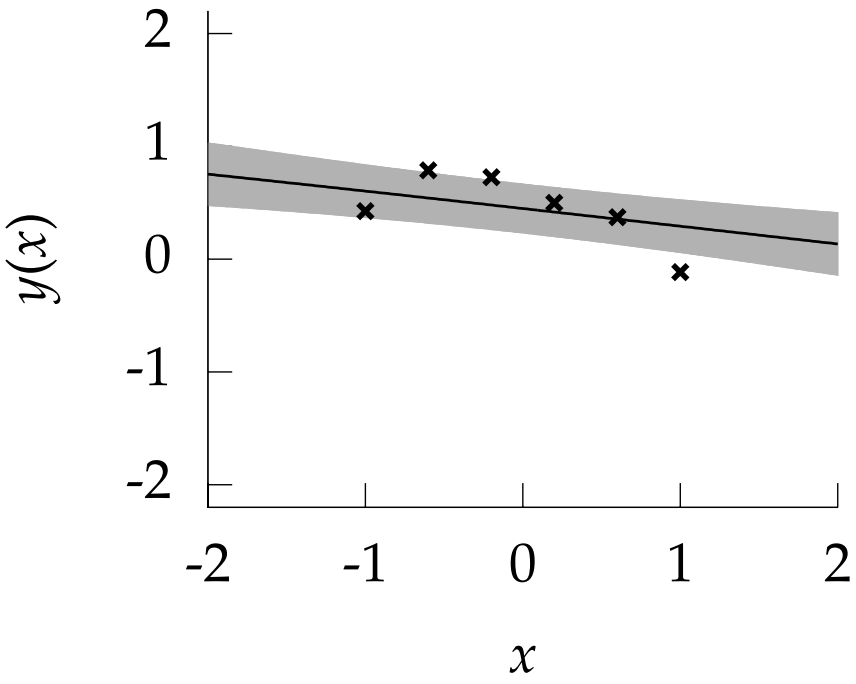


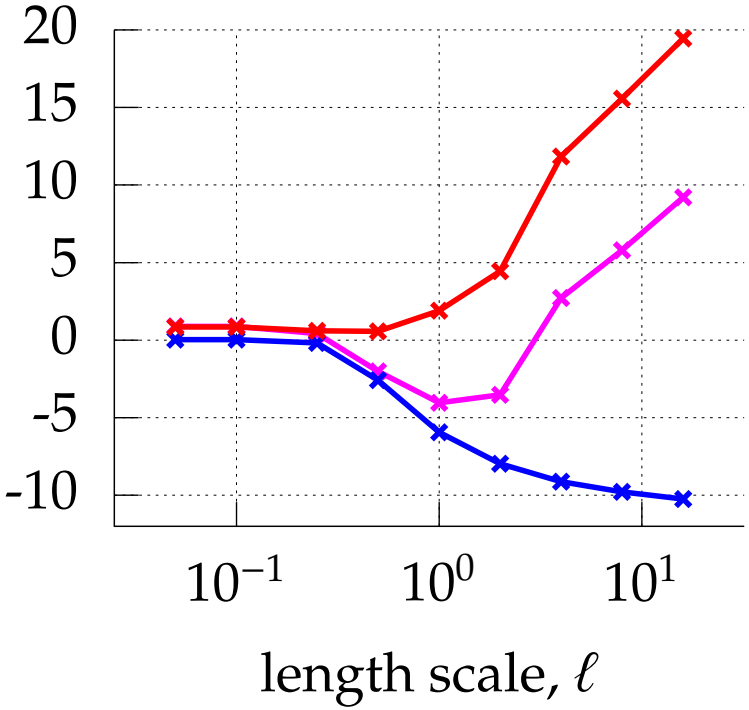


$$E(\boldsymbol{\theta}) = \frac{1}{2}\log|\mathbf{K}| + \frac{\mathbf{y}^\top\mathbf{K}^{-1}\mathbf{y}}{2}$$

Learning Covariance Parameters

Can we determine length scales and noise levels from the data?





$$E(\boldsymbol{\theta}) = \frac{1}{2}\log|\mathbf{K}| + \frac{\mathbf{y}^\top \mathbf{K}^{-1}\mathbf{y}}{2}$$

Gene Expression Example

- Given given expression levels in the form of a time series from Della Gatta et al. (2008).
- Want to detect if a gene is expressed or not, fit a GP to each gene (Kalaitzis and Lawrence, 2011).

Kalaitzis and Lawrence *BMC Bioinformatics* 2011, **12**:180
http://www.biomedcentral.com/1471-2105/12/180

BMC Bioinformatics

RESEARCH ARTICLE Open Access

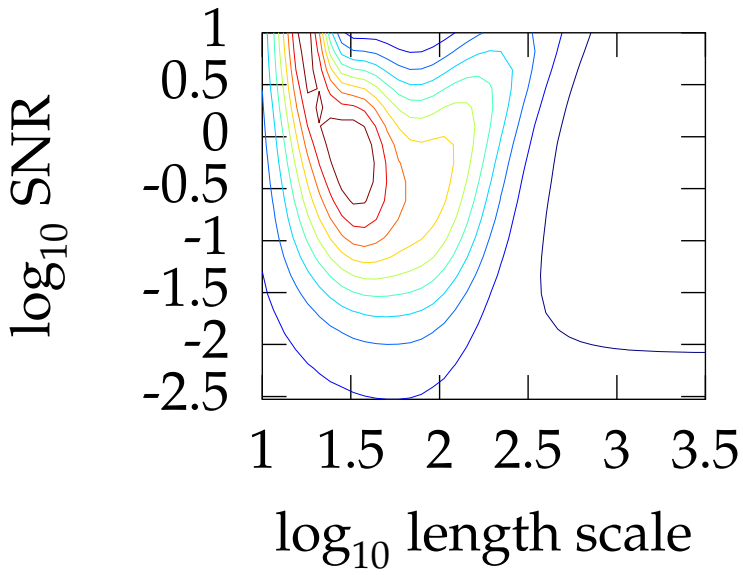
A Simple Approach to Ranking Differentially Expressed Gene Expression Time Courses through Gaussian Process Regression

Alfredo A Kalaitzis* and Neil D Lawrence*

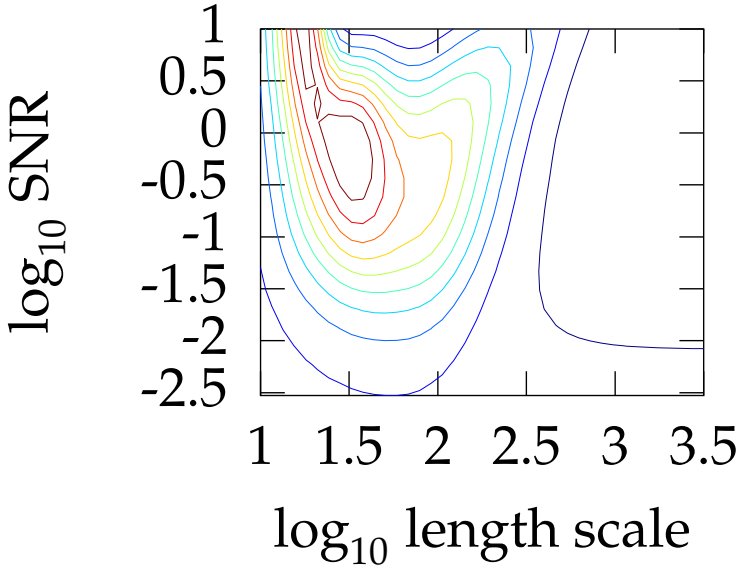
Abstract

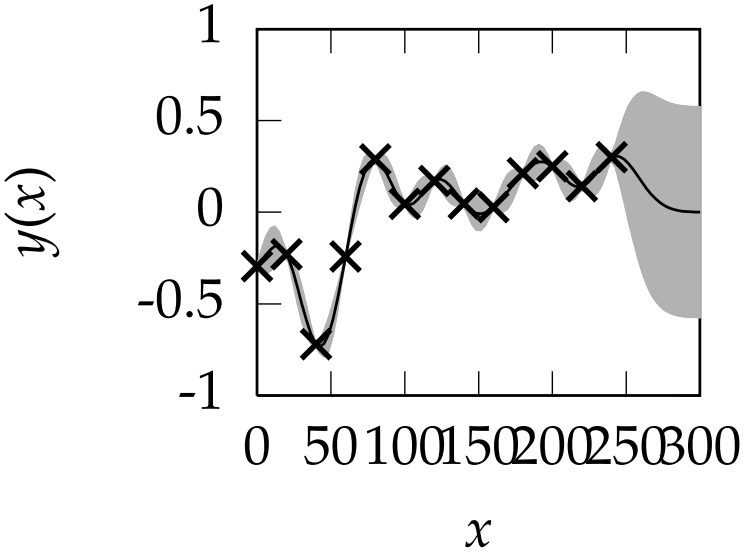
Background: The analysis of gene expression from time series underpins many biological studies. Two basic forms of analysis recur for data of this type: removing inactive (quiet) genes from the study and determining which genes are differentially expressed. Often these analysis stages are applied disregarding the fact that the data is drawn from a time series. In this paper we propose a simple model for accounting for the underlying temporal nature of the data based on a Gaussian process.

Results: We review Gaussian process (GP) regression for estimating the continuous trajectories underlying in gene expression time-series. We present a simple approach which can be used to filter quiet genes, or for the case of time series in the form of expression ratios, quantify differential expression. We assess via ROC curves the rankings produced by our regression framework and compare them to a recently proposed hierarchical Bayesian model for the analysis of gene expression time-series (BATS). We compare on both simulated and experimental data showing that the proposed approach considerably outperforms the current state of the art.

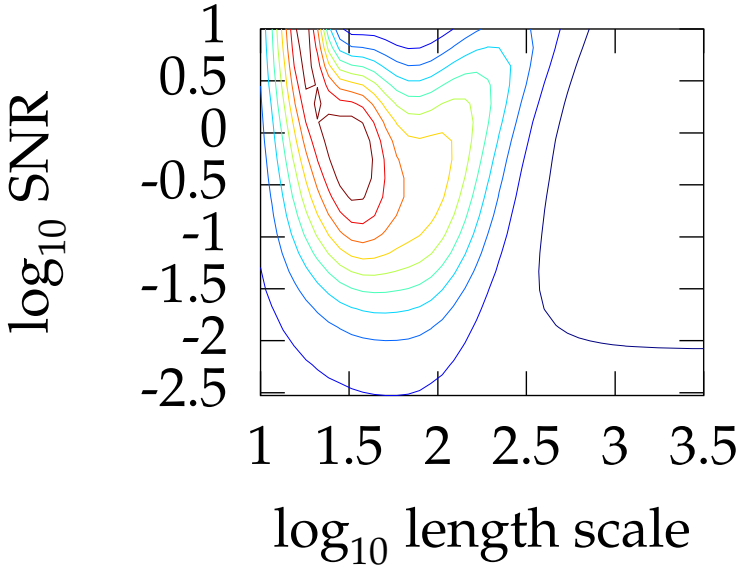


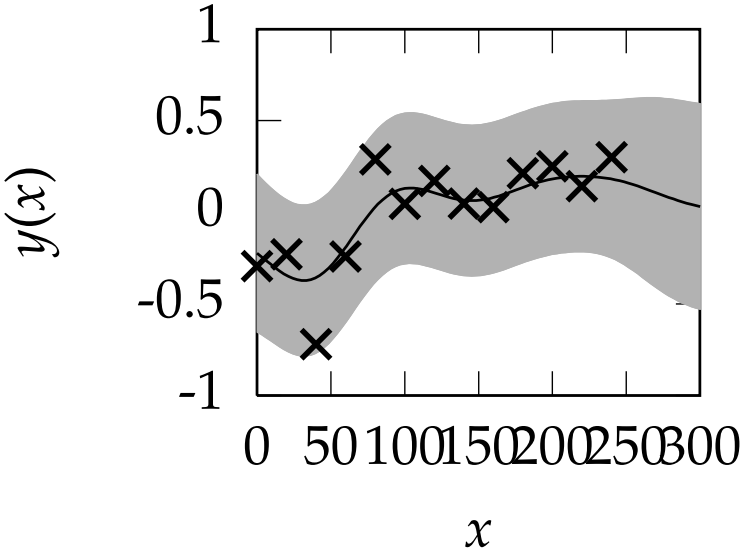
Contour plot of Gaussian process likelihood.



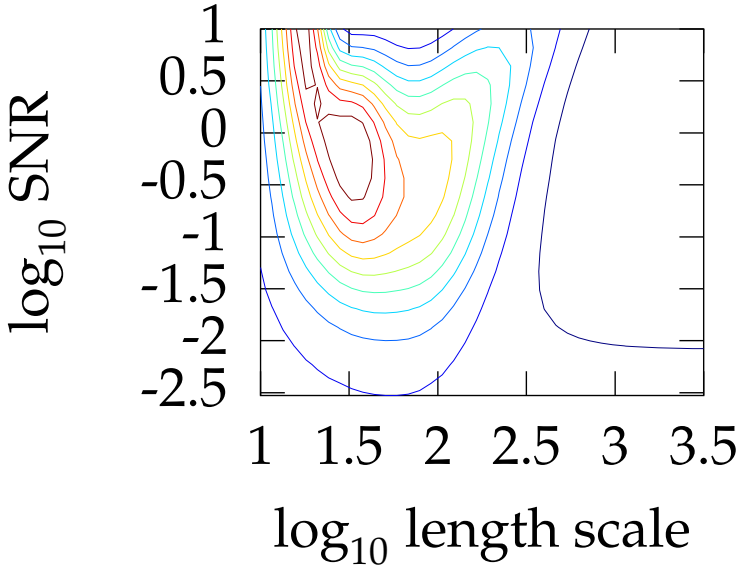


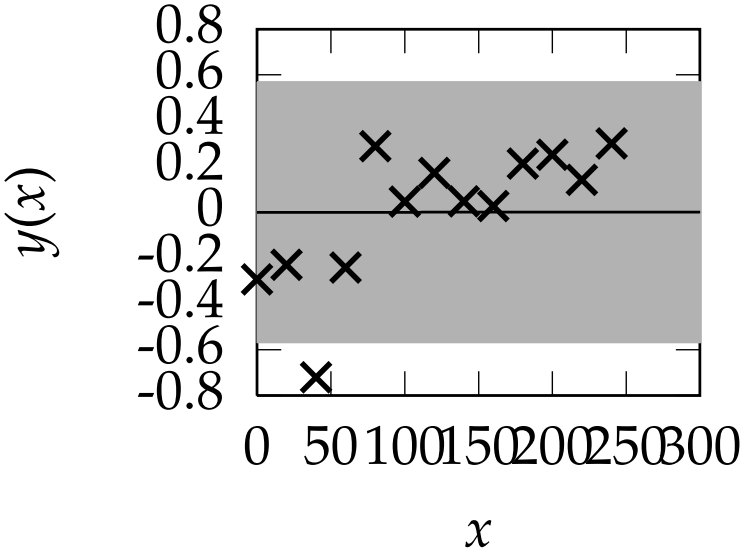
Optima: length scale of 1.2221 and $\log_{10}$ SNR of 1.9654 log likelihood is -0.22317.





Optima: length scale of 1.5162 and $\log_{10}$ SNR of 0.21306 log likelihood is -0.23604.





Optima: length scale of 2.9886 and $\log_{10}$ SNR of -4.506 log likelihood is -2.1056.

Limitations of Gaussian Processes

- Inference is $O(n^3)$ due to matrix inverse (in practice use Cholesky).
- Gaussian processes don't deal well with discontinuities (financial crises, phosphorylation, collisions, edges in images).
- Widely used exponentiated quadratic covariance (RBF) can be too smooth in practice (but there are many alternatives!!).

Gaussian Process Summer School

- Series of summer schools on GPs: `http://ml.dcs.shef.ac.uk/gpss/`
- Next edition 15th–17th September, followed by a workshop on 18th September.
- Limited to 50 students, combination of lectures and practical sessions.
- Facebook page: `https://www.facebook.com/gaussianprocesssummerschool`
- Videos from earlier editions here: `https://www.youtube.com/user/ProfNeilLawrence`

References I

G. Della Gatta, M. Bansal, A. Ambesi-Impiombato, D. Antonini, C. Missero, and D. di Bernardo. Direct targets of the trp63 transcription factor revealed by a combination of gene expression profiling and reverse engineering. *Genome Research*, 18(6):939–948, Jun 2008. [URL]. [DOI].

A. A. Kalaitzis and N. D. Lawrence. A simple approach to ranking differentially expressed gene expression time courses through Gaussian process regression. *BMC Bioinformatics*, 12(180), 2011. [DOI].

P. S. Laplace. *Essai philosophique sur les probabilités*. Courcier, Paris, 2nd edition, 1814. Sixth edition of 1840 translated and repreinted (1951) as *A Philosophical Essay on Probabilities*, New York: Dover; fifth edition of 1825 reprinted 1986 with notes by Bernard Bru, Paris: Christian Bourgois Éditeur, translated by Andrew Dale (1995) as *Philosophical Essay on Probabilities*, New York:Springer-Verlag.

J. Oakley and A. O'Hagan. Bayesian inference for the uncertainty distribution of computer model outputs. *Biometrika*, 89(4):769–784, 2002.

C. E. Rasmussen and C. K. I. Williams. *Gaussian Processes for Machine Learning*. MIT Press, Cambridge, MA, 2006. [Google Books] .

Outline

Outline

Case study: User impact on Twitter

Predicting and characterising user impact on Twitter

- define a user-level impact score
- use user's text and profile information as features to predict the score
- analyse the features which better predict the score
- provide users with 'guidelines' for improving their score

Instance of a text prediction problem

- emphasis on feature analysis and interpretability (specific to social science applications)
- non-linear variation

See our paper Lampos et al. (2014), EACL.

Sparse GPs

Exact inference in a GP

- Memory: $\mathcal{O}(n^2)$
- Time: $\mathcal{O}(n^3)$
  where n is the number of training points.

Sparse GP approximation

- Memory: $\mathcal{O}(n \cdot m)$
- Time: $\mathcal{O}(n \cdot m^2)$
  where m is selected at runtime, $m \ll n$.

Are usually needed when $n > 1000$.

Sparse GPs

Many options for sparse approximations

- Based on Inducing Variables
  - Subset of Data (SoD)
  - Subset of Regressors (SoR)
  - Deterministic Training Conditional (DTC)
  - Partially Independent Training Conditional Approximations (PITC)
  - Fully Independent Training Conditional Approximations (FITC)
- Fast Matrix Vector Multiplication (MVM)
- Variational Methods

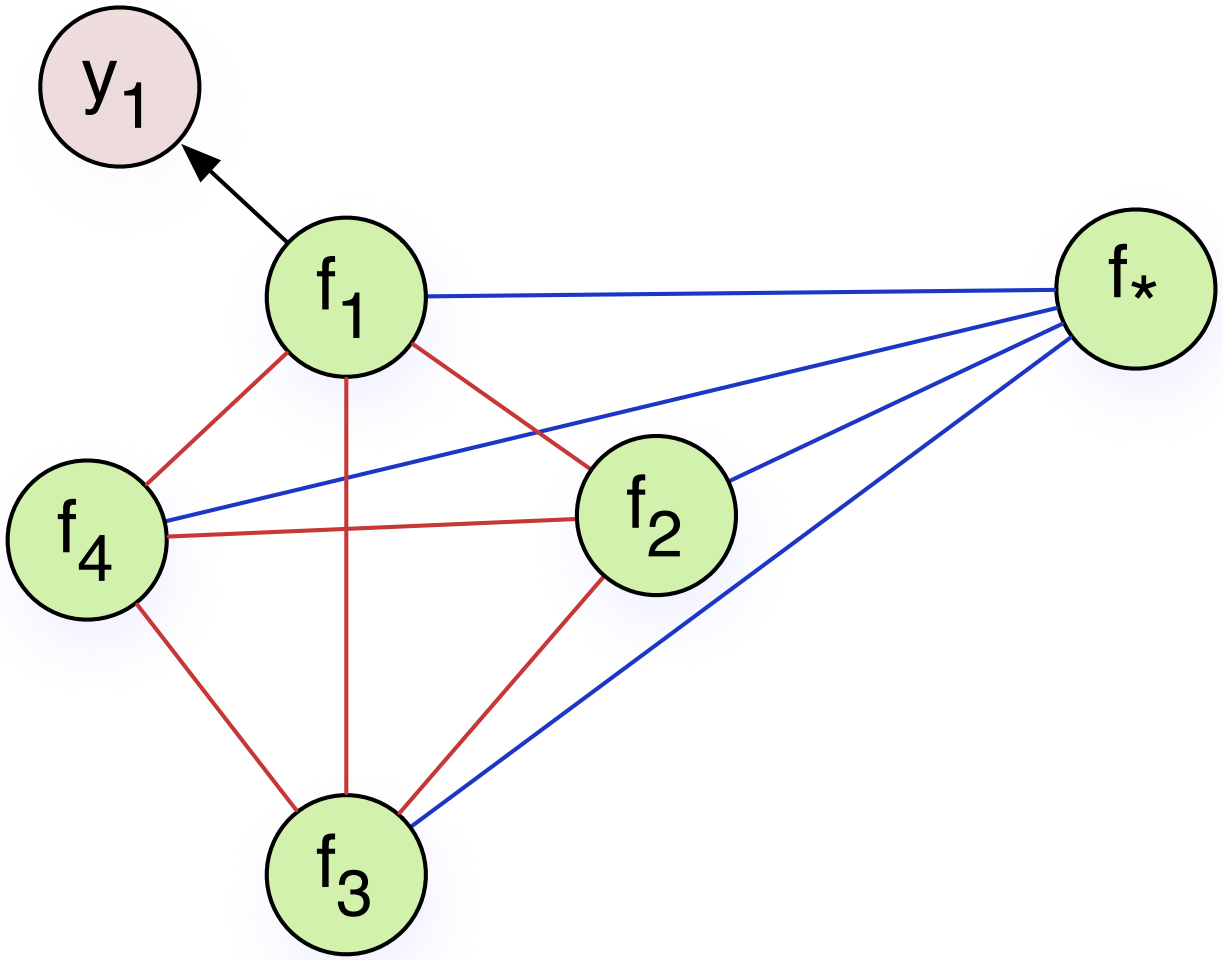
See Quiñonero Candela and Rasmussen (2005) for an overview.

Sparse GPs

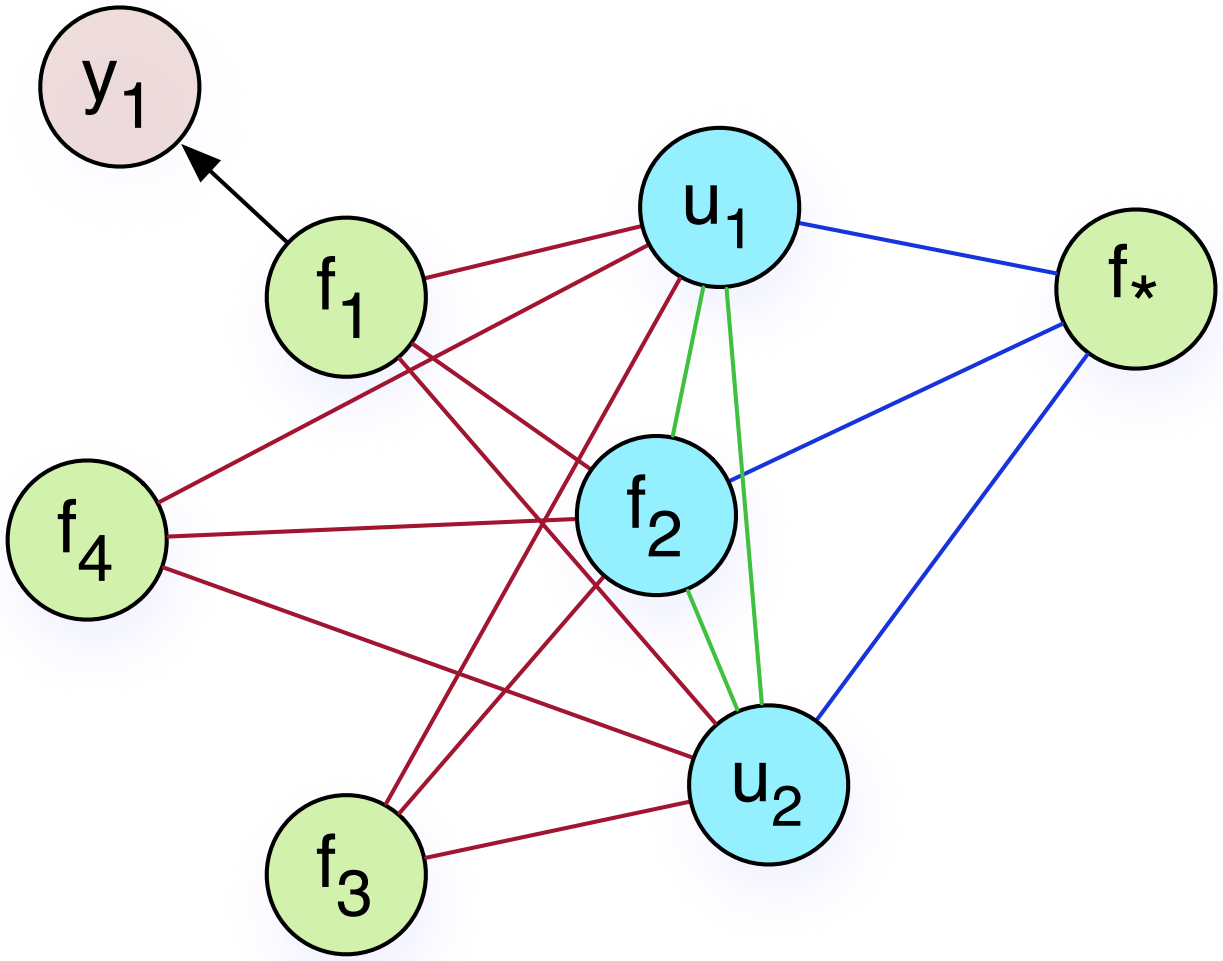
Sparse approximations

- $f(x_i)$ are treated as latent variables
  - a subset are treated exactly ($|\boldsymbol{u}| = m$)
  - the other are given a computationally cheaper treatment
- following Quiñonero Candela and Rasmussen (2005), we view Sparse GP approximation as 'exact inference with an approximate prior'
- modify the joint prior $p(f(x), f(x_*))$ to reduce the $\mathcal{O}(n^3)$ complexity
- different methods based on the effective prior used
- the GP model is concerned only with the conditional of the outputs given the inputs

Inducing points



Inducing points



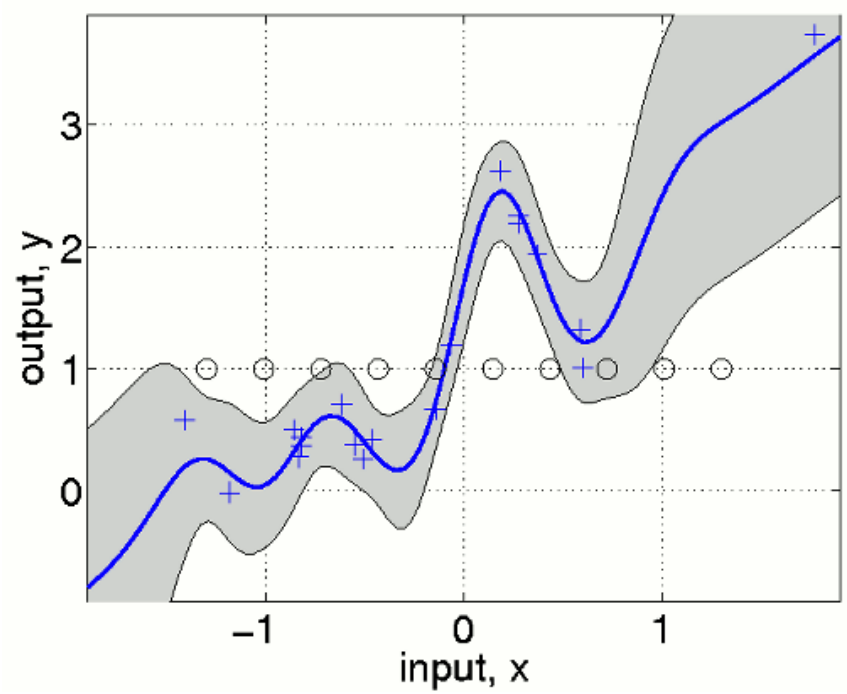
Inducing points

The inducing points $u = (u_1, ..., u_m)$ 'induce' the dependencies between train and test points. All computations are based on cross-covariances between training, test and inducing points only.

- assume that f and f_* are conditionally independent given the inducing points u:

$$p(f(x_*), f(x)) \simeq q(f(x_*), f(x)) = \int q(f(x_*)|u)p(f(x)|u)p(u)du$$

Inducing points



- choosing the inducing points: usually equispaced, not necessarily in the training set
- a random subset of the training points can be used
- note the predictive variance will be overestimated outside the support of the inducing points
- photo from GPML documentation

Fully Independent Training Conditional (FITC)

Exact inference in a GP:

$$p(f(x_*)|y) = \mathcal{N}(K_{x_*,x}(K_{x,x} + \sigma^2 I)^{-1} y, K_{x_*,x_*} - K_{x_*,x}(K_{x,x} + \sigma^2 I)^{-1} K_{x,x_*})$$

where $p(f(x), f(x_*)) = \mathcal{N}\left(0, \begin{smallmatrix} K_{x,x} & K_{x_*,x} \\ K_{x,x_*} & K_{x_*,x_*} \end{smallmatrix}\right)$

FITC predictive distribution:

$$q_{FITC}(f(x_*)|y) = \mathcal{N}(Q_{x_*,x}(Q_{x,x} + \mathrm{diag}(K_{x,x} - Q_{x,x} + \sigma^2 I))^{-1} y,$$
$$K_{x_*,x_*} - Q_{x_*,x}(Q_{x,x} + \mathrm{diag}(K_{x,x} - Q_{x,x} + \sigma^2 I))^{-1} Q_{x,x_*})$$

based on a low-rank plus diagonal approximation:

$q_{FITC}(f(x), f(x_*)) = \mathcal{N}\left(0, \begin{smallmatrix} Q_{x,x} - \mathrm{diag}(Q_{x,x} - K_{x,x}) & Q_{x,x_*} \\ Q_{x_*,x} & K_{x_*,x_*} \end{smallmatrix}\right)$ and
$Q_{i,j} \triangleq K_{i,u} K_{u,u}^{-1} K_{u,j}$

Predicting and characterising user impact

500 million Tweets a day in Twitter

- important and some not so important information
- breaking news from media
- friends
- celebrity self promotion
- marketing
- spam

Can we automatically predict the impact of a user?

Can we automatically identify factors which influence user impact?

Defining user impact

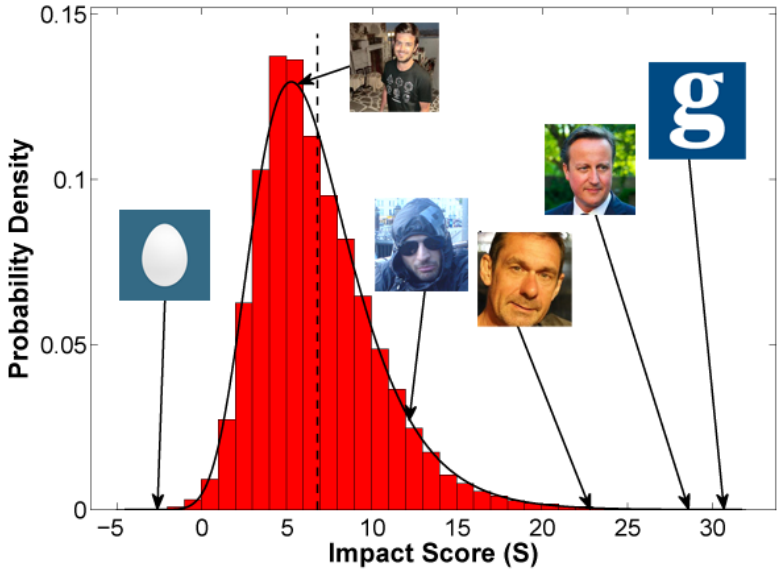
Define impact as a function of network connections

- no. of followers
- no. of followees
- no. of time the account is listed by others

$$\text{Impact} = ln\left(\frac{\text{listings}\cdot\text{followers}^2}{\text{followees}}\right)$$

Dataset

- 38.000 UK users
- all tweets from one year
- 48 million deduplicated messages



User controlled features

Only features under the user's control (e.g. not no. of retweets)

- User features (18)
  extracted from the account profile
  aggregated text features
- Text features (100)
  user's topic distribution
  topics computed using spectral clustering on the word co-occurrence (NPMI) matrix

Models

Regression task

- Gaussian Process regression model
- $n = 38000 \cdot 9/10$, use Sparse GPs with FITC
- Squared Exponential kernel (k-dimensional):

$$k(\boldsymbol{x}_p, \boldsymbol{x}_q) = \sigma_f^2 \exp\left(-\frac{1}{2}(\boldsymbol{x}_p - \boldsymbol{x}_q)^T D(\boldsymbol{x}_p - \boldsymbol{x}_q)\right) + \sigma_n^2 \delta_{pq}$$

  where $D \in \mathbb{R}^{k \times k}$ is a symmetric matrix.
- if $D_{ARD} = \mathrm{diag}(\boldsymbol{l})^{-2}$:

$$k(\boldsymbol{x}_p, \boldsymbol{x}_q) = \sigma_f^2 \exp\left(-\frac{1}{2}\sum_{d=1}^{k} \frac{(\boldsymbol{x}_{pd} - \boldsymbol{x}_{pd})^2}{l_d^2}\right) + \sigma_n^2 \delta_{pq}$$

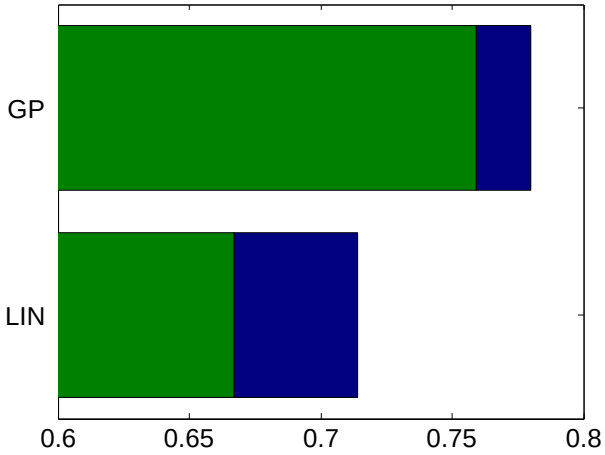
Automatic Relevance Determination (ARD)

- $$k_{ARD}(\boldsymbol{x}_p, \boldsymbol{x}_q) = \sigma_f^2 \exp\left(-\frac{1}{2}\sum_{d=1}^{k} \frac{(\boldsymbol{x}_{pd} - \boldsymbol{x}_{pd})^2}{l_d^2}\right) + \sigma_n^2 \delta_{pq}$$
- SE kernel with automatic relevance determination (ARD), with the vector $\boldsymbol{l}$ denoting the characteristic length-scales of each feature
- l_d measures the distance for being uncorrelated along x_d
- $1/l_d^2$ proportional to how relevant a feature is: large length-scales means the covariance becomes independent of that feature value
- sorting by length-scales indicates which features impact the prediction the most
- tuning these parameters in done via Bayesian model selection

Prediction results

Experiments

- 10-fold cross validation
- using predictive mean
- baseline model is ridge regression **(LIN)**
- **Profile features**
- **Text features**



Pearson correlation

Conclusions

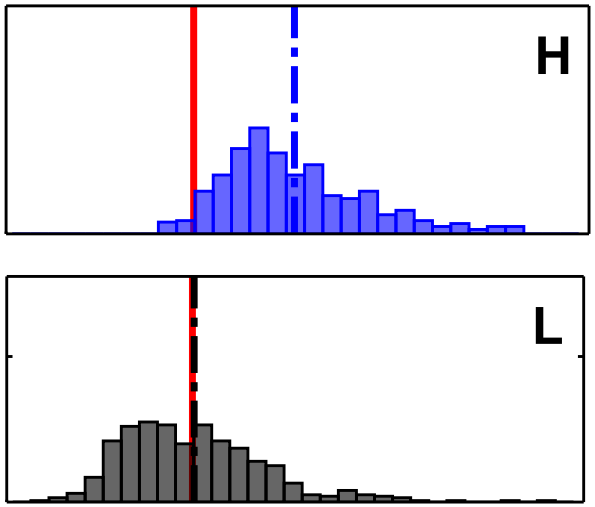
- GPs substantially better than ridge regression
- non-linear GPs with only profile features performs better than linear methods with all features
- GPs outperform SVR
- adding topic features improves all models

Selected features

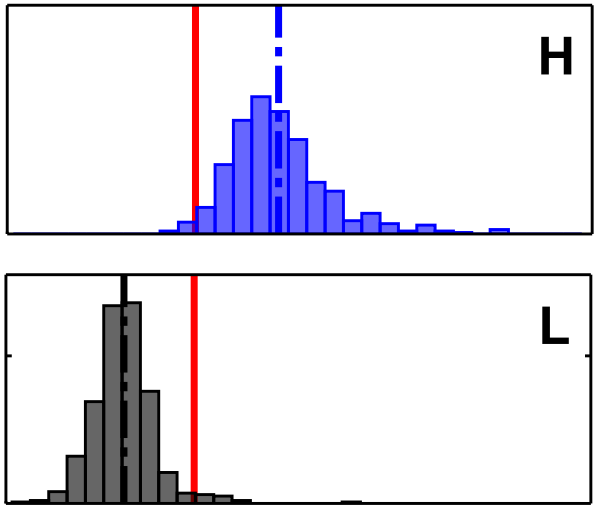
Feature	Importance
Using default profile image	0.73
Total number of tweets (entire history)	1.32
Number of unique @-mentions in tweets	2.31
Number of tweets (in dataset)	3.47
Links ratio in tweets	3.57
T1 (Weather): mph, humidity, barometer, gust, winds	3.73
T2 (Healthcare, Housing): nursing, nurse, rn, registered, bedroom, clinical, #news, estate, #hospital	5.44
T3 (Politics): senate, republican, gop, police, arrested, voters, robbery, democrats, presidential, elections	6.07
Proportion of days with non-zero tweets	6.96
Proportion of tweets with @-replies	7.10

Feature analysis

No. of unique @-mentions

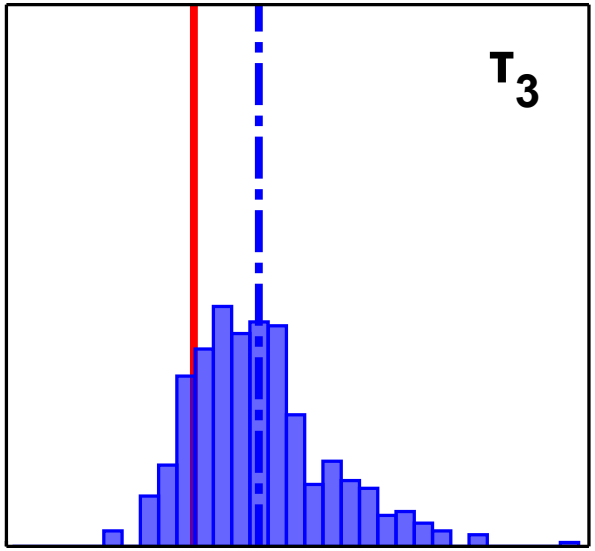


No. of tweets

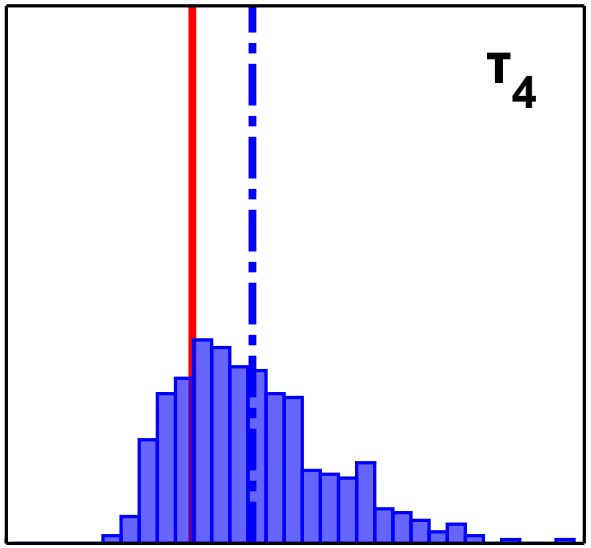


Impact histogram for users with **high (H)** values of this feature as opposed to **low (L)**. **Red line** is the mean impact score.

Feature analysis



damon, potter, #tvd, harry elena, kate, portman, pattinson, hermione, jennifer



senate, republican, gop, police, arrested, voters, robbery, democrats, presidential, elections

Impact histogram for users with **high (H)** values of this feature. **Red line** is the mean impact score.

Conclusions

User impact is highly predictable

- user behaviour very informative
- ‘tips’ for improving your impact

GP framework suitable

- non-linear modelling
- ARD feature selection
- sparse GPs allow large scale experiments
- empirical improvements over linear models & SVR

Outline

Case study: MT Evaluation

Evaluation of Machine Translation

- human assessment of translation quality
- many 'good' translations, no gold standard
- judgements highly subjective, biased, noisy

Instance of general NLP annotation problem

- multiply annotated data, mixing experts and novices
- slippery task definition, low agreement

See our paper Cohn and Specia (2013), ACL.

Multi-task learning

Multi-task learning

- form of transfer learning
- several related tasks sharing the same input data representation
- learn the types, extent of correlations

Compared to domain adaptation

- tasks need not be identical (even regression vs classification)
- no explicit 'target' domain
- several sources of variation besides domain
- no assumptions of data asymmetry

Multi-task learning for MT Evaluation

Modelling individual annotators

- each bring own biases
- but correlated decisions with others' annotations
- could even find clusters of common solutions

Here use multi-output GP regression

- joint inference over several translators
- learn degree of inter-task transfer
- learn per-translator noise
- incorporate task meta-data

Previous work on modelling MT Quality

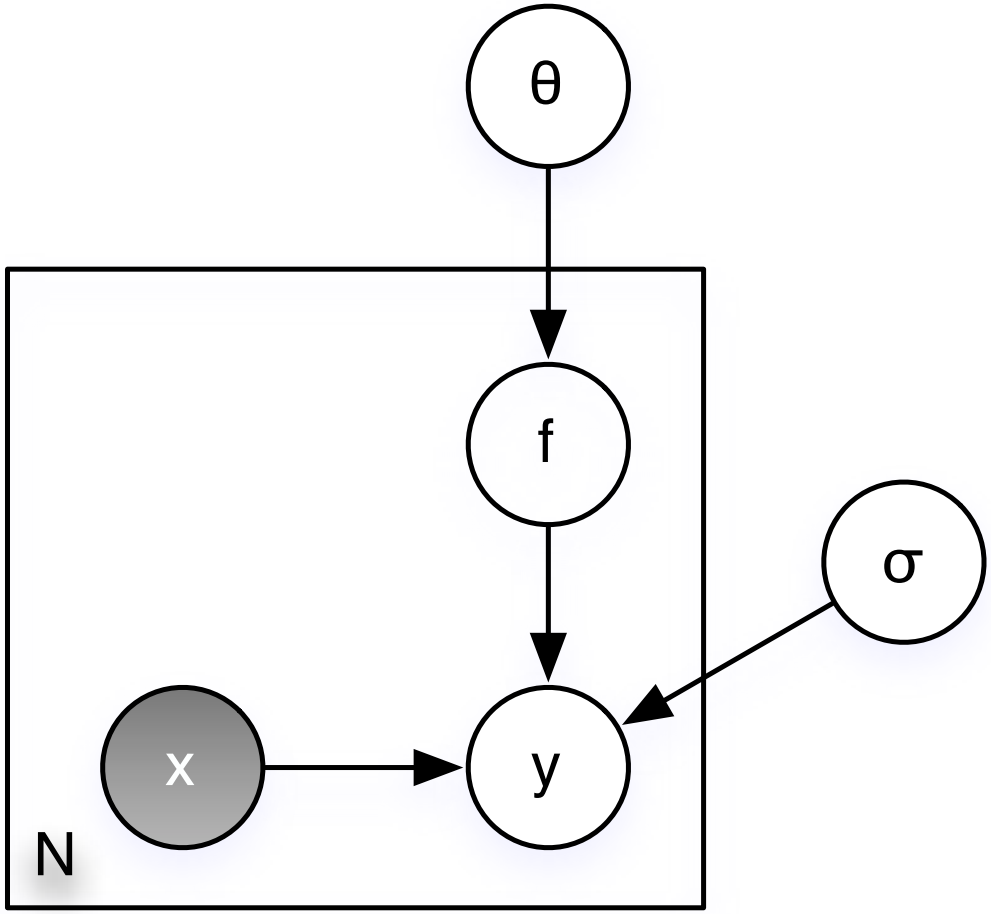
Typically simplified into a single-task modelling problem

- learn one model from one "good" annotator
- average several annotators
- ignore variation and simply pool data

Here framed as Transfer Learning

- each individual is a separate "task"
- joint modelling of individuals and the group

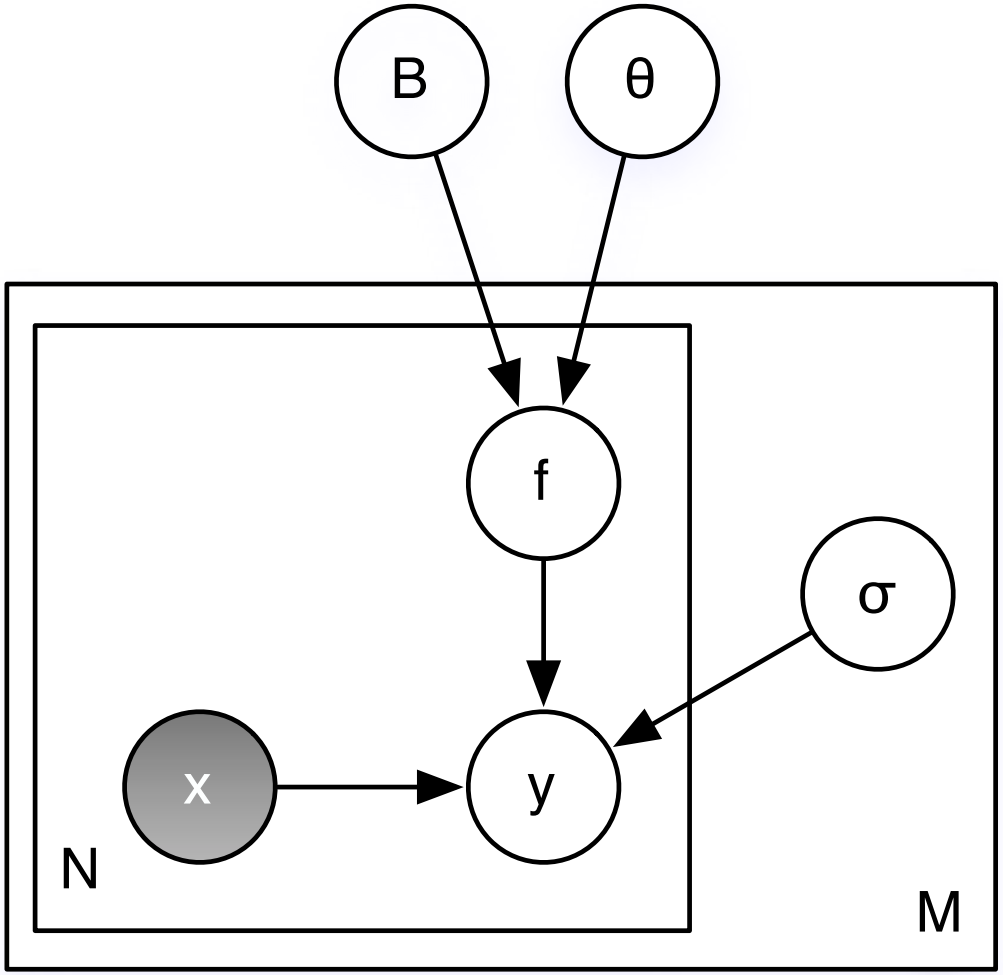
Review: GP Regression



$$\mathbf{f} \sim \mathrm{GP}(\mathbf{0}, \theta)$$

$$y_i \sim \mathrm{N}(f(\mathbf{x}_i), \sigma^2)$$

Multi-task GP Regression

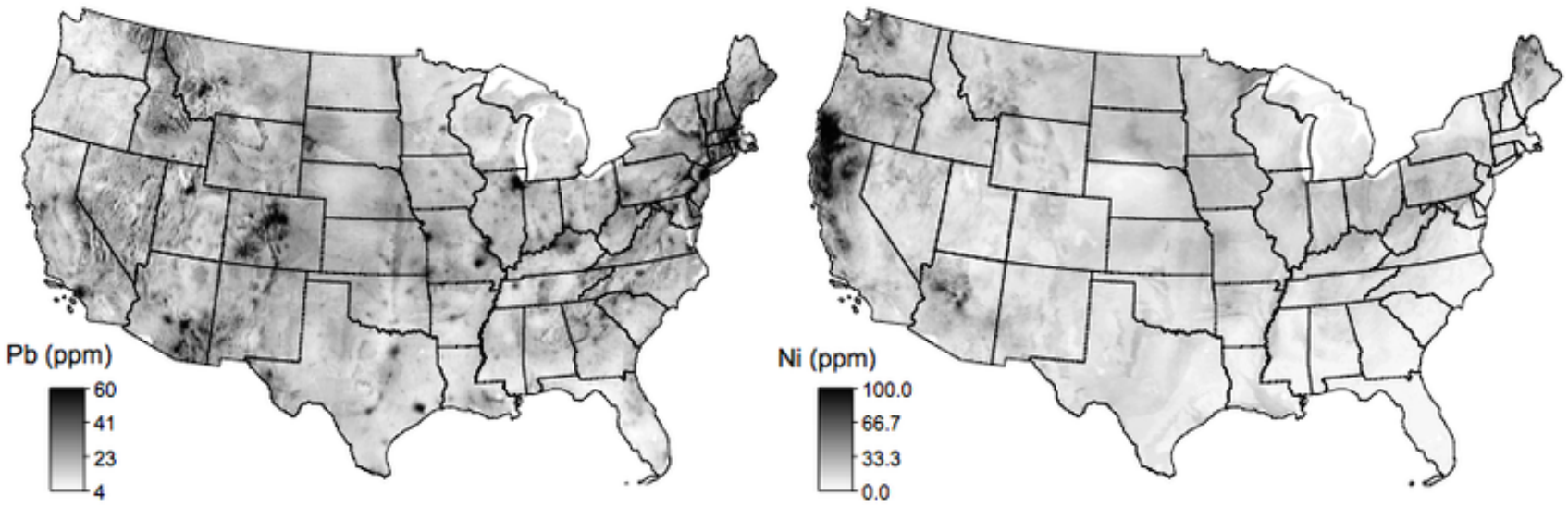


$$\mathbf{f} \sim \mathrm{GP}(\mathbf{0}, \left(B, \theta\right))$$

$$y_{im} \sim \mathrm{N}(f_m(\mathbf{x}_i), \sigma_m^2)$$

See Alvarez et al. (2011).

Geostatistical origins: ‘co-kriging’



Kriging is the application of GPs in geostatistics, to predict e.g., locations of minerals from several soil samples. Co-kriging jointly models several different outputs expected to have significant correlation, e.g., lead and nickel deposits.

See Alvarez, Rosasco and Lawrence, 2012.

Multi-task Covariance Kernels

Represent data as $(\mathbf{x}, t, y)$ tuples, where t is a task identifier. Define a *separable* covariance kernel,

$$K(\mathbf{x}, \mathbf{x}')_{t,t'} = B_{t,t'} k_\theta(\mathbf{x}, \mathbf{x}') + \text{noise}$$

- effectively each input augmented with t, indexing the task of interest
- the coregionalisation matrix, $\mathbf{B} \in \mathcal{R}^{M \times M}$ weights inter-task covariance
- the data kernel k_θ takes data points $\mathbf{x}$ as input e.g., exponentiated quadratic

Coregionalisation Kernels

Generally $\mathbf{B}$ can be any symmetric positive semi-definite matrix. Some interesting choices

- $\mathbf{I}$ encodes independent learning
- $\mathbf{1}$ encodes pooled learning
- interpolating the above
- full rank $\mathbf{B} = \mathbf{W}\mathbf{W}^\top$, or low rank variants

Known as the Intrinsic model of coregionalisation (IMC).

See Alvarez et al. (2011); Bonilla et al. (2008)

Stacking and Kronecker products

Response variables are a matrix

$$\mathbf{Y} = \mathcal{R}^{N \times M}$$

Represent data in 'stacked' form

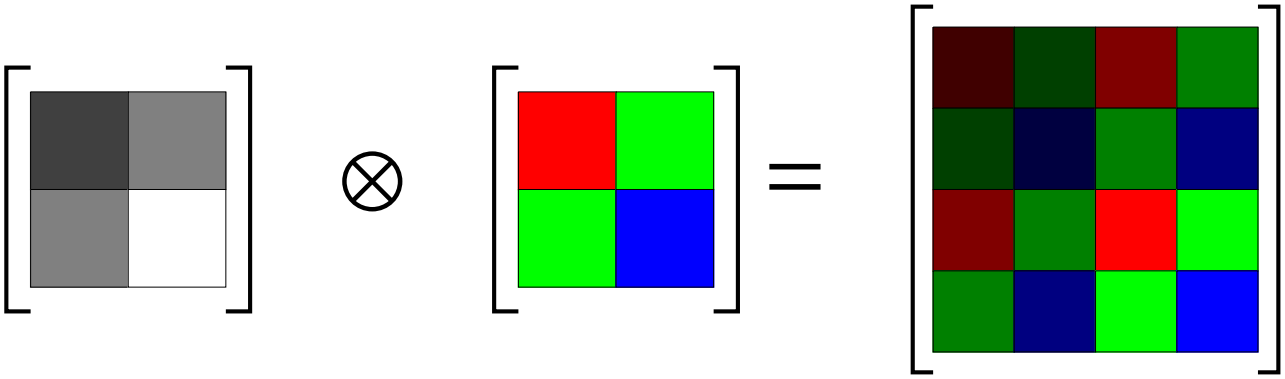
$$\mathbf{X} = \begin{bmatrix} \mathbf{x}_1 \\ \mathbf{x}_2 \\ \vdots \\ \mathbf{x}_N \\ \vdots \\ \mathbf{x}_1 \\ \mathbf{x}_2 \\ \vdots \\ \mathbf{x}_N \end{bmatrix} \qquad \mathbf{y} = \begin{bmatrix} y_{11} \\ y_{21} \\ \vdots \\ y_{N1} \\ \vdots \\ y_{1M} \\ y_{2M} \\ \vdots \\ y_{NM} \end{bmatrix}$$

Kernel a Kronecker product $\mathbf{K}(\mathbf{X}, \mathbf{X}) = \mathbf{B} \otimes k_{\text{data}}(\mathbf{X}_o, \mathbf{X}_o)$

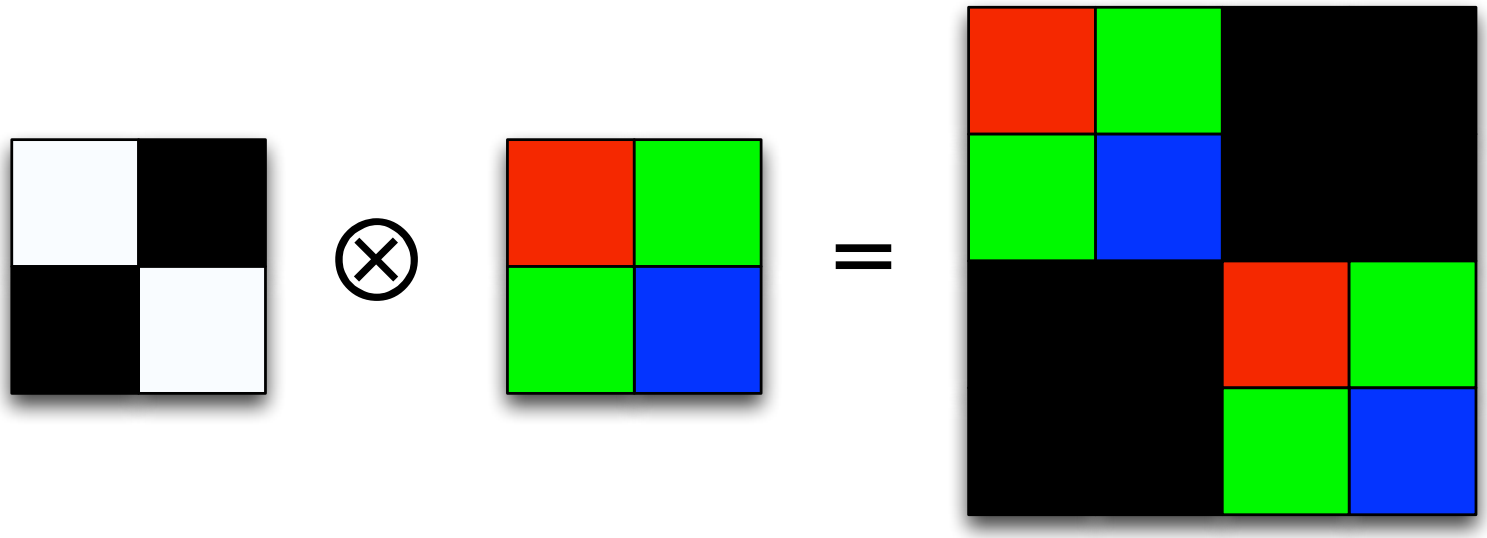
Kronecker product

$$\begin{bmatrix} a & b \\ c & d \end{bmatrix} \otimes \mathbf{K} = \begin{bmatrix} a\mathbf{K} & b\mathbf{K} \\ c\mathbf{K} & d\mathbf{K} \end{bmatrix}$$

Kronecker product

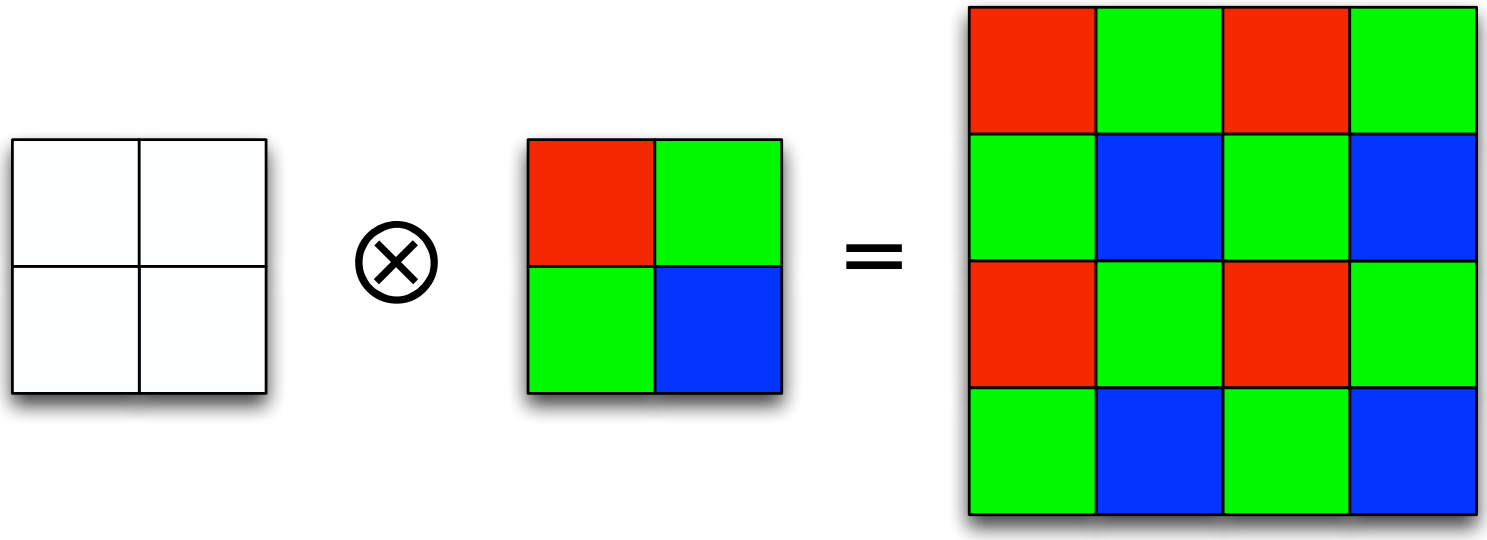


Choices for B: Independent learning



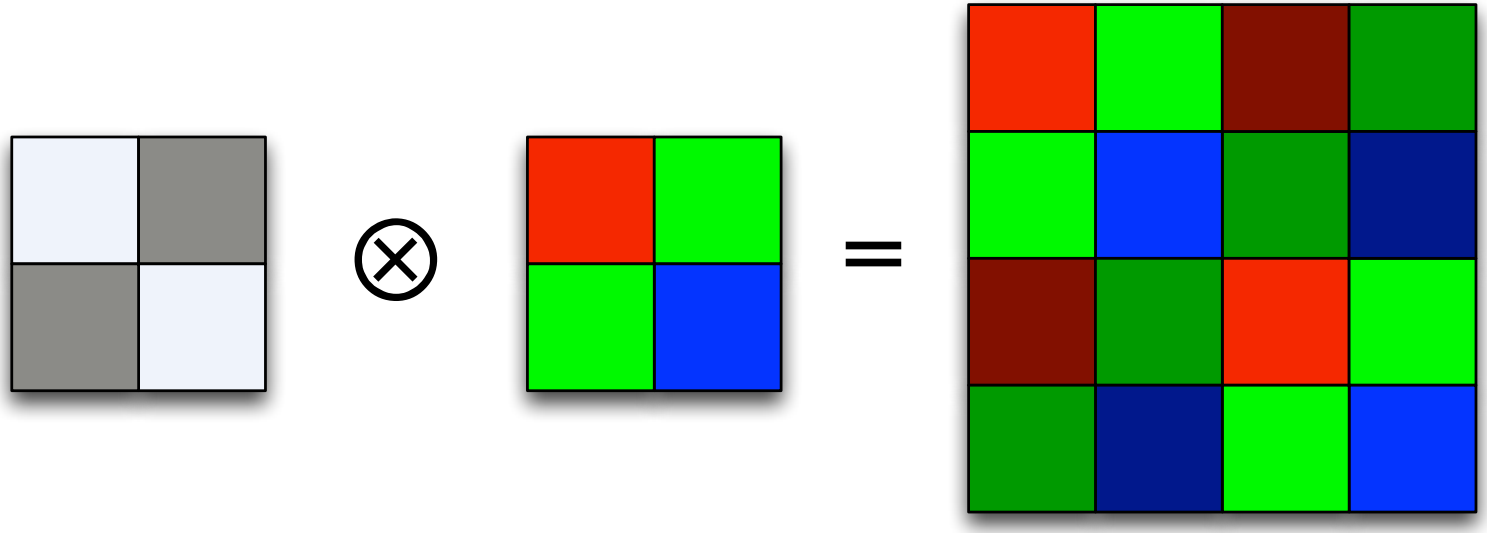
$$\mathbf{B} = \mathbf{I}$$

Choices for B: Pooled learning



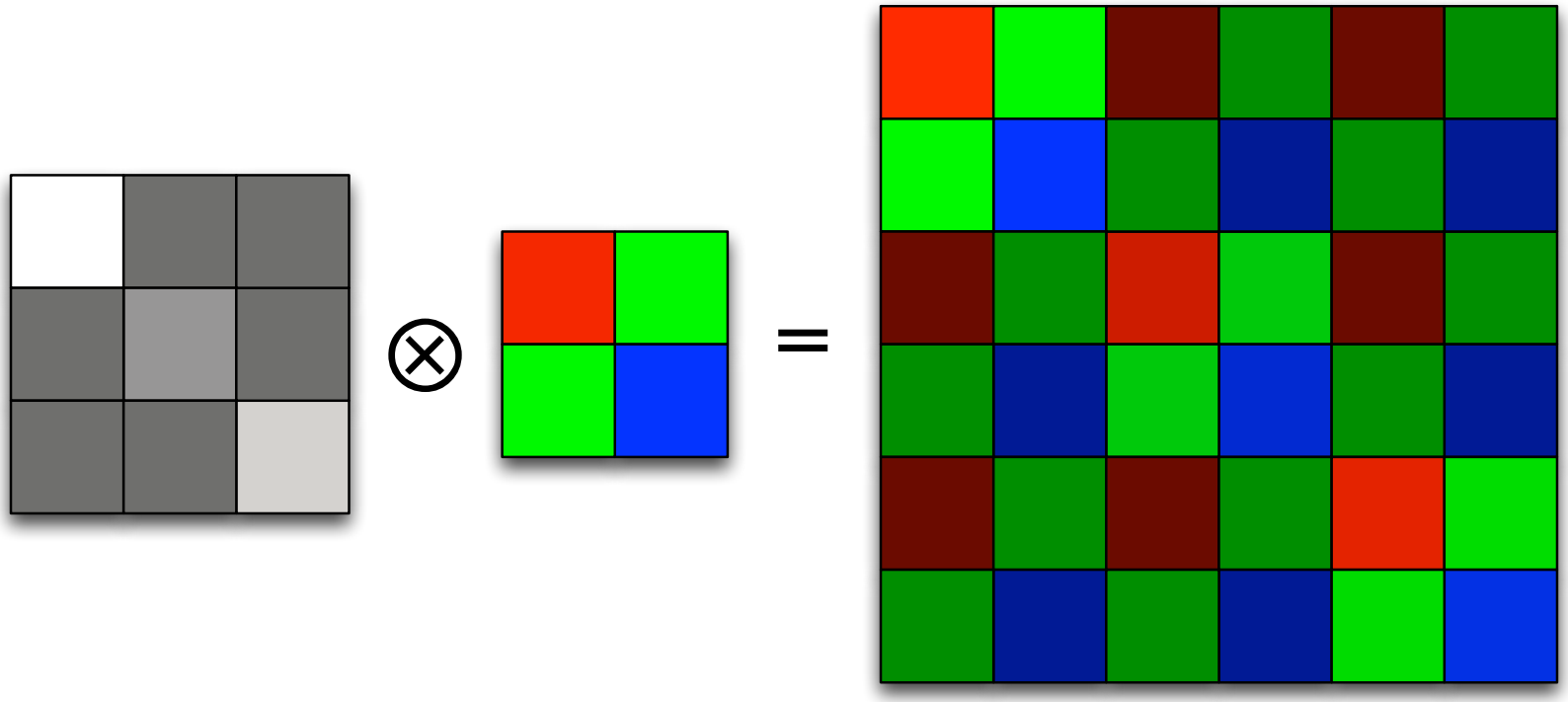
$$\mathbf{B} = \mathbf{1}$$

Choices for B: Interpolating independent and pooled learning



$$\mathbf{B} = \mathbf{1} + \alpha \mathbf{I}$$

Choices for B: Modulating independent and pooled learning II



$$\mathbf{B} = \mathbf{1} + \mathrm{diag}(\alpha)$$

Compared to Daumé III (2007)

Feature augmentation approach to multi-task learning. Uses horizontal data stacking:

$$\mathbf{X} = \begin{bmatrix} \mathbf{X}^{(1)} & \mathbf{X}^{(1)} & \mathbf{0} \\ \mathbf{X}^{(2)} & \mathbf{0} & \mathbf{X}^{(2)} \end{bmatrix} \qquad \mathbf{y} = \begin{bmatrix} \mathbf{y}^{(1)} \\ \mathbf{y}^{(2)} \end{bmatrix}$$

where $(\mathbf{X}^{(i)}, \mathbf{y}^{(i)})$ are the training data for task i. This expands the feature space by a factor of M.

Equivalent to a multitask kernel

$$k(\mathbf{x}, \mathbf{x}')_{t,t'} = (1 + \delta(t, t'))\, \mathbf{x}^\top \mathbf{x}'$$
$$\mathbf{K}(\mathbf{X}, \mathbf{X}) = (\mathbf{1} + \mathbf{I}) \otimes k_{\text{linear}}(\mathbf{X}, \mathbf{X})$$

$\Rightarrow$ A specific choice of $\mathbf{B}$ with a linear data kernel

Compared to Evgeniou et al. (2006)

In the regularisation setting, Evgeniou et al. (2006) show that the kernel

$$K(\mathbf{x}, \mathbf{x}')_{t,t'} = (1 - \lambda + \lambda M \delta(t, t'))\, \mathbf{x}^\top \mathbf{x}'$$

is equivalent to a linear model with regularisation term

$$J(\Theta) = \frac{1}{M}\left(\sum_t \|\theta_t\|^2 + \frac{1-\lambda}{\lambda}\|\theta_t - \frac{1}{M}\sum_{t'} \theta_{t'}\|^2\right)$$

This regularises each task's parameters θ_t towards the mean parameters over all tasks, $\frac{1}{M}\sum_{t'} \theta_{t'}$.

A form of *interpolation* method from before.

Linear model of coregionalisation

Consider a mixture of several components,

$$K(\mathbf{x}, \mathbf{x}')_{t,t'} = \sum_{q=1}^{Q} \mathbf{B}^{(q)}_{t,t'} k_{\theta_q}(\mathbf{x}, \mathbf{x}')$$

Includes per-component

- data kernel, parameterised by θ_q
- coregionalisation matrix, $\mathbf{B}^{(q)}$

More flexible than ICM, which corresponds to $Q = 1$. Can capture multi-output correlations, e.g., as different length scales.

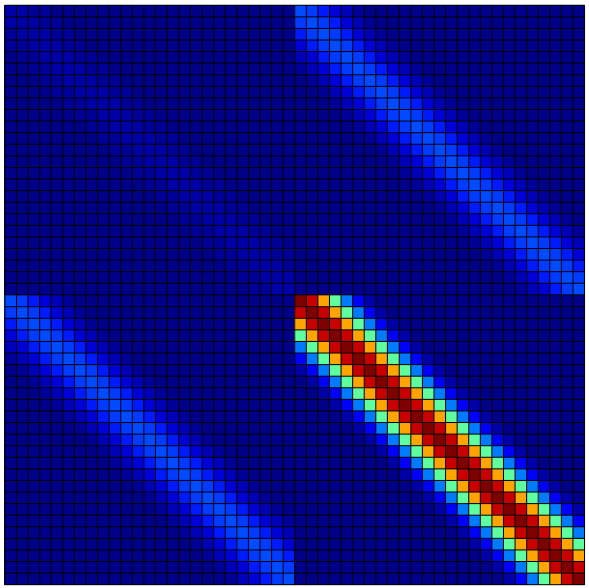
See Alvarez et al. (2011)

ICM samples

$$\mathbf{K}(\mathbf{X}, \mathbf{X}) = \mathbf{w}\mathbf{w}^\top \otimes k(\mathbf{X}, \mathbf{X}).$$

$$\mathbf{w} = \begin{bmatrix} 1 \\ 5 \end{bmatrix}$$

$$\mathbf{B} = \begin{bmatrix} 1 & 5 \\ 5 & 25 \end{bmatrix}$$

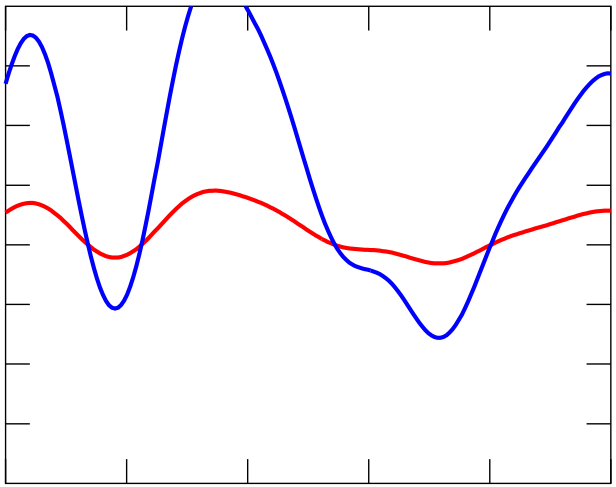


ICM samples

$$\mathbf{K}(\mathbf{X}, \mathbf{X}) = \mathbf{w}\mathbf{w}^\top \otimes k(\mathbf{X}, \mathbf{X}).$$

$$\mathbf{w} = \begin{bmatrix} 1 \\ 5 \end{bmatrix}$$

$$\mathbf{B} = \begin{bmatrix} 1 & 5 \\ 5 & 25 \end{bmatrix}$$

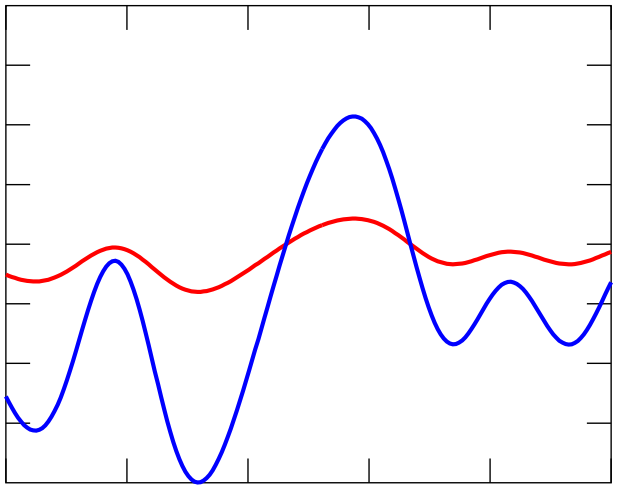


ICM samples

$$\mathbf{K}(\mathbf{X}, \mathbf{X}) = \mathbf{w}\mathbf{w}^\top \otimes k(\mathbf{X}, \mathbf{X}).$$

$$\mathbf{w} = \begin{bmatrix} 1 \\ 5 \end{bmatrix}$$

$$\mathbf{B} = \begin{bmatrix} 1 & 5 \\ 5 & 25 \end{bmatrix}$$

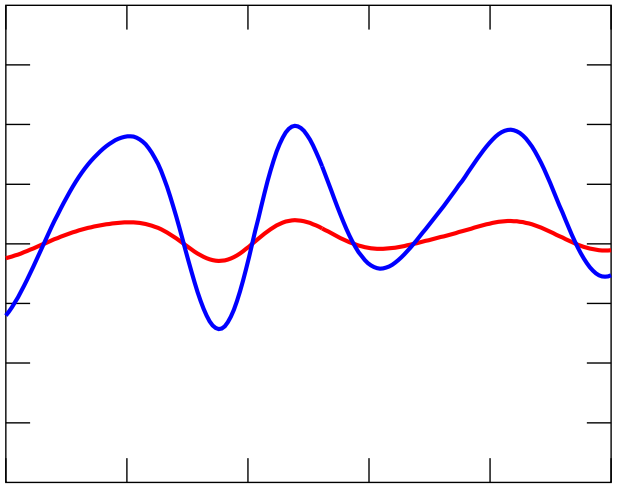


ICM samples

$$\mathbf{K}(\mathbf{X}, \mathbf{X}) = \mathbf{w}\mathbf{w}^\top \otimes k(\mathbf{X}, \mathbf{X}).$$

$$\mathbf{w} = \begin{bmatrix} 1 \\ 5 \end{bmatrix}$$

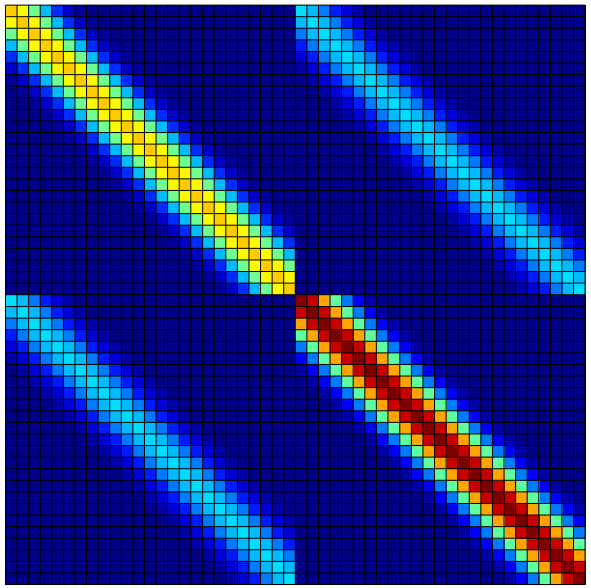
$$\mathbf{B} = \begin{bmatrix} 1 & 5 \\ 5 & 25 \end{bmatrix}$$



ICM samples

$$\mathbf{K}(\mathbf{X}, \mathbf{X}) = \mathbf{B} \otimes k(\mathbf{X}, \mathbf{X}).$$

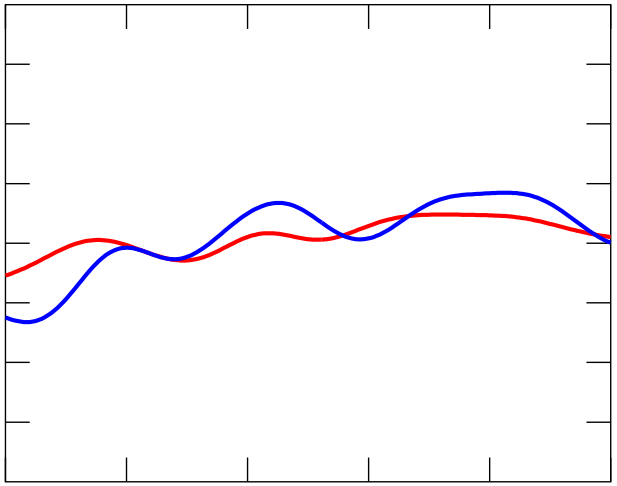
$$\mathbf{B} = \begin{bmatrix} 1 & 0.5 \\ 0.5 & 1.5 \end{bmatrix}$$



ICM samples

$$\mathbf{K}(\mathbf{X}, \mathbf{X}) = \mathbf{B} \otimes k(\mathbf{X}, \mathbf{X}).$$

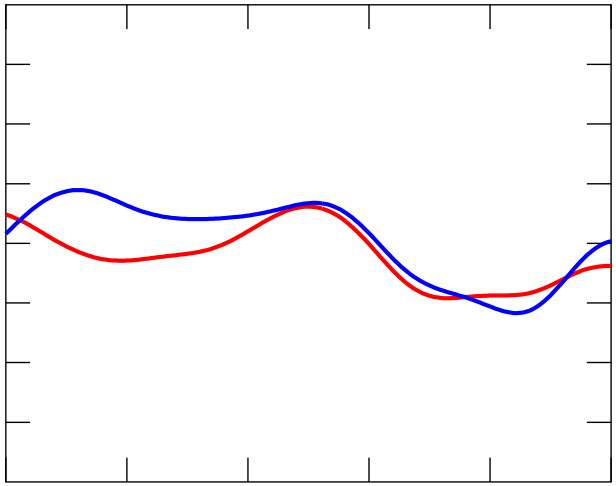
$$\mathbf{B} = \begin{bmatrix} 1 & 0.5 \\ 0.5 & 1.5 \end{bmatrix}$$



ICM samples

$$\mathbf{K}(\mathbf{X}, \mathbf{X}) = \mathbf{B} \otimes k(\mathbf{X}, \mathbf{X}).$$

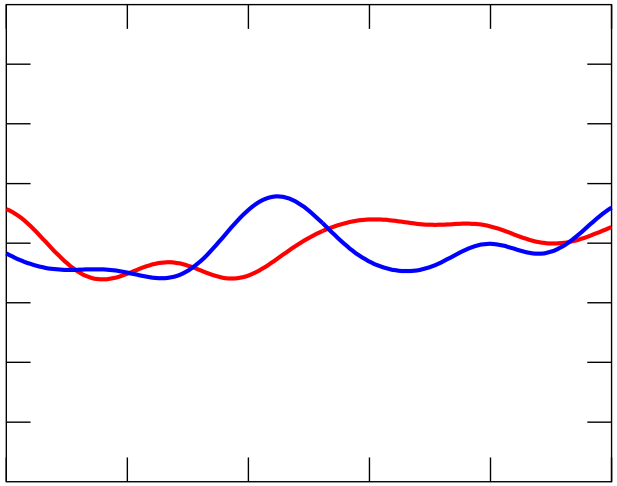
$$\mathbf{B} = \begin{bmatrix} 1 & 0.5 \\ 0.5 & 1.5 \end{bmatrix}$$



ICM samples

$$\mathbf{K}(\mathbf{X}, \mathbf{X}) = \mathbf{B} \otimes k(\mathbf{X}, \mathbf{X}).$$

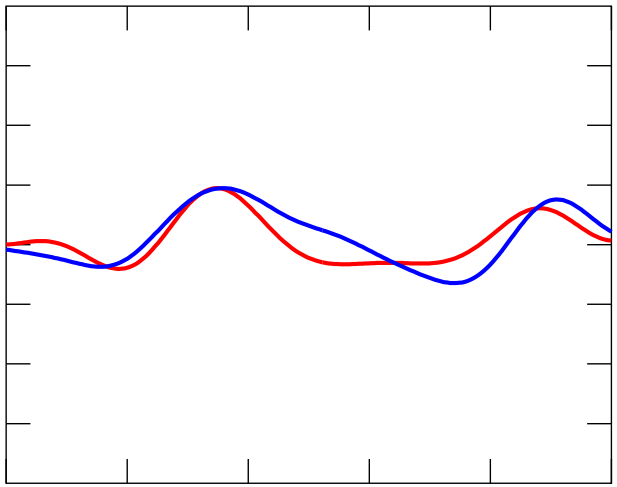
$$\mathbf{B} = \begin{bmatrix} 1 & 0.5 \\ 0.5 & 1.5 \end{bmatrix}$$



ICM samples

$$\mathbf{K}(\mathbf{X}, \mathbf{X}) = \mathbf{B} \otimes k(\mathbf{X}, \mathbf{X}).$$

$$\mathbf{B} = \begin{bmatrix} 1 & 0.5 \\ 0.5 & 1.5 \end{bmatrix}$$



Application to MT quality estimation

Case study of MT Quality Estimation, manual assessment of translation quality given source and translation texts

Human judgements are highly subjective, biased, noisy

- typing speed
- experience levels
- expectations from MT

'Quality' can be measured many ways

- subjective scoring (1-5) for fluency, adequacy, perceived effort to correct
- post-editing effort: HTER or time taken
- binary judgements, ranking, . . .

Experimental setup

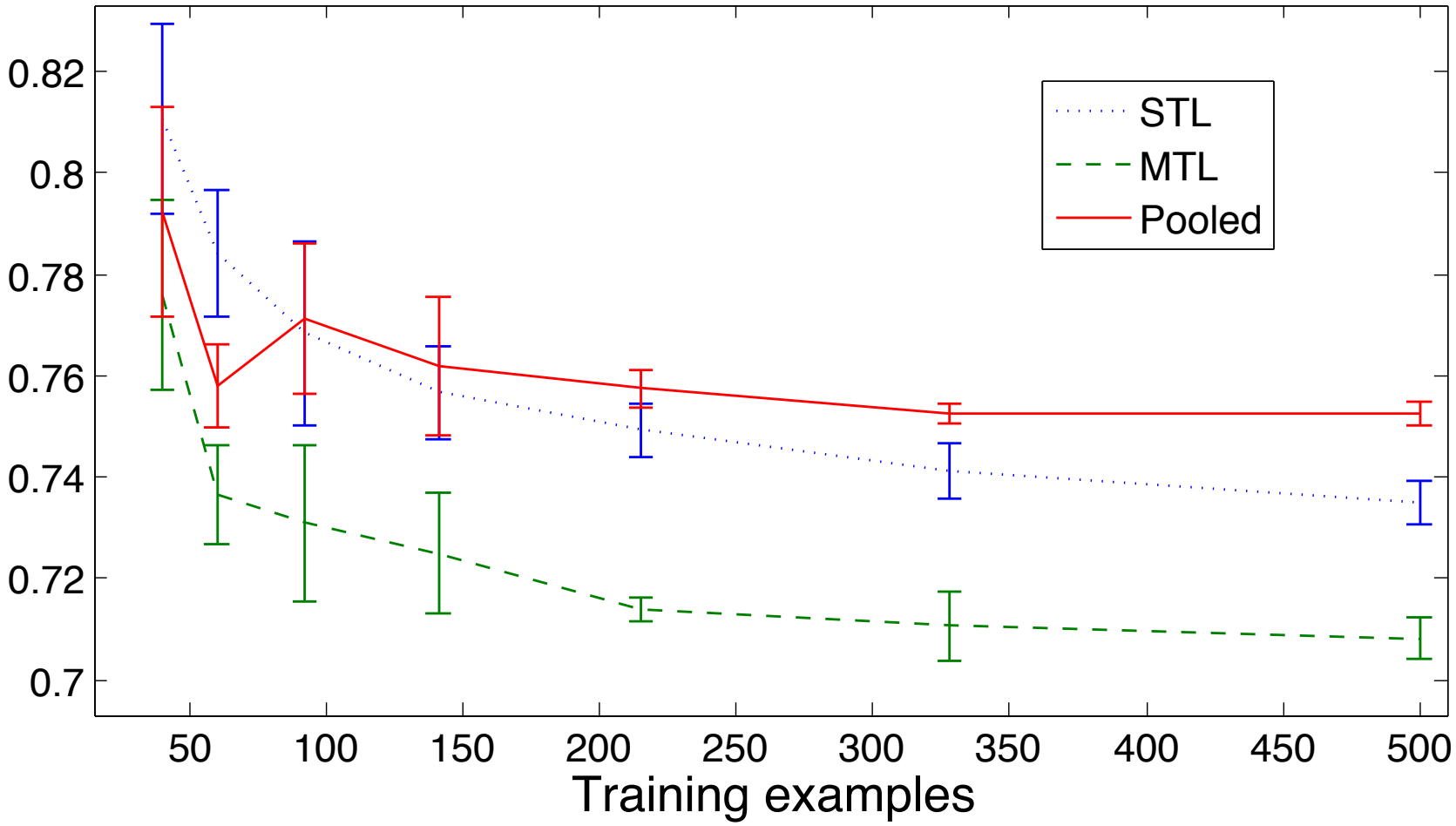
Quality estimation data

- 2k examples of source sentence and MT output
- measuring subjective post-editing (1-5) WMT12
- post-editing time per word, in log seconds WPTP12
- 17 dense features extracted using
  Quest toolkit (Specia et al., 2013)
- using official train/test split, or random assignment

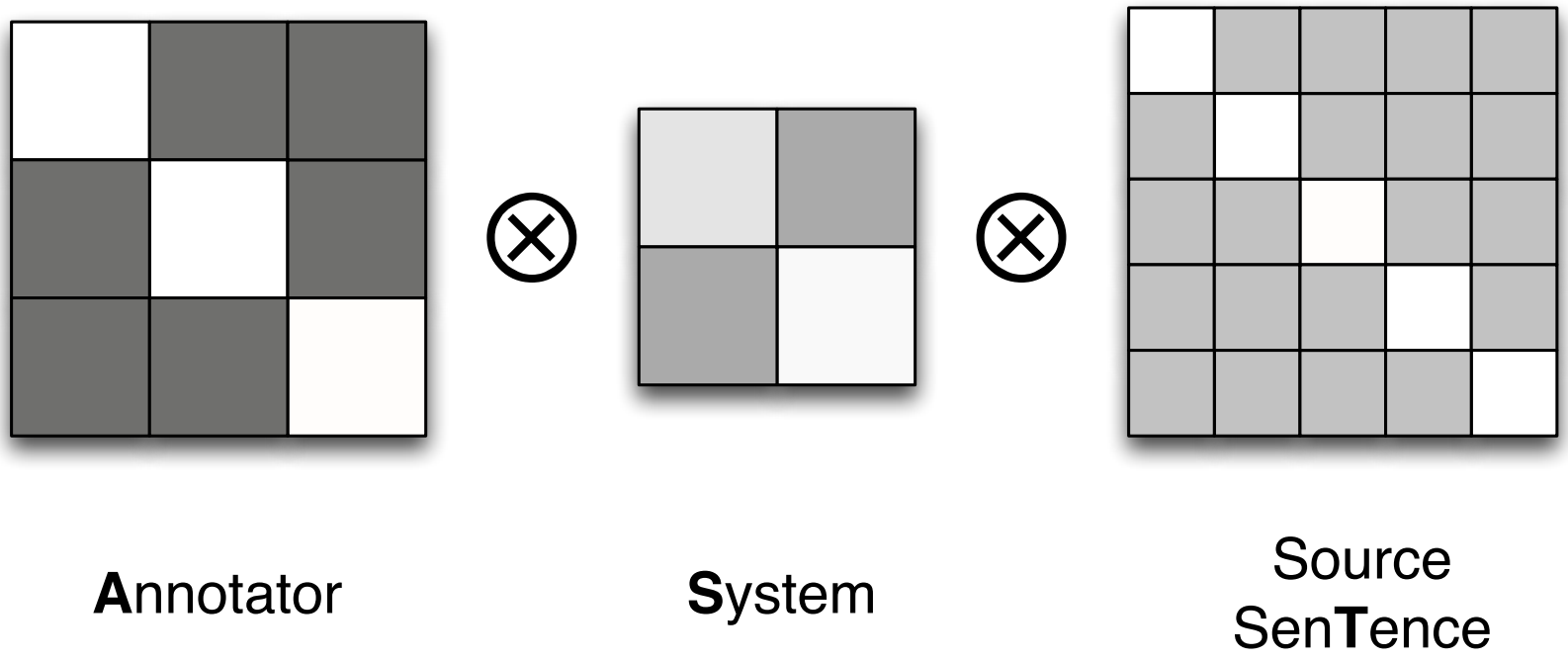
Gaussian Process models

- exponentiated quadratic data kernel (RBF)
- hyper-parameter values trained using type II MLE
- consider simple interpolation coregionalisation kernels
- include per-task noise or global tied noise

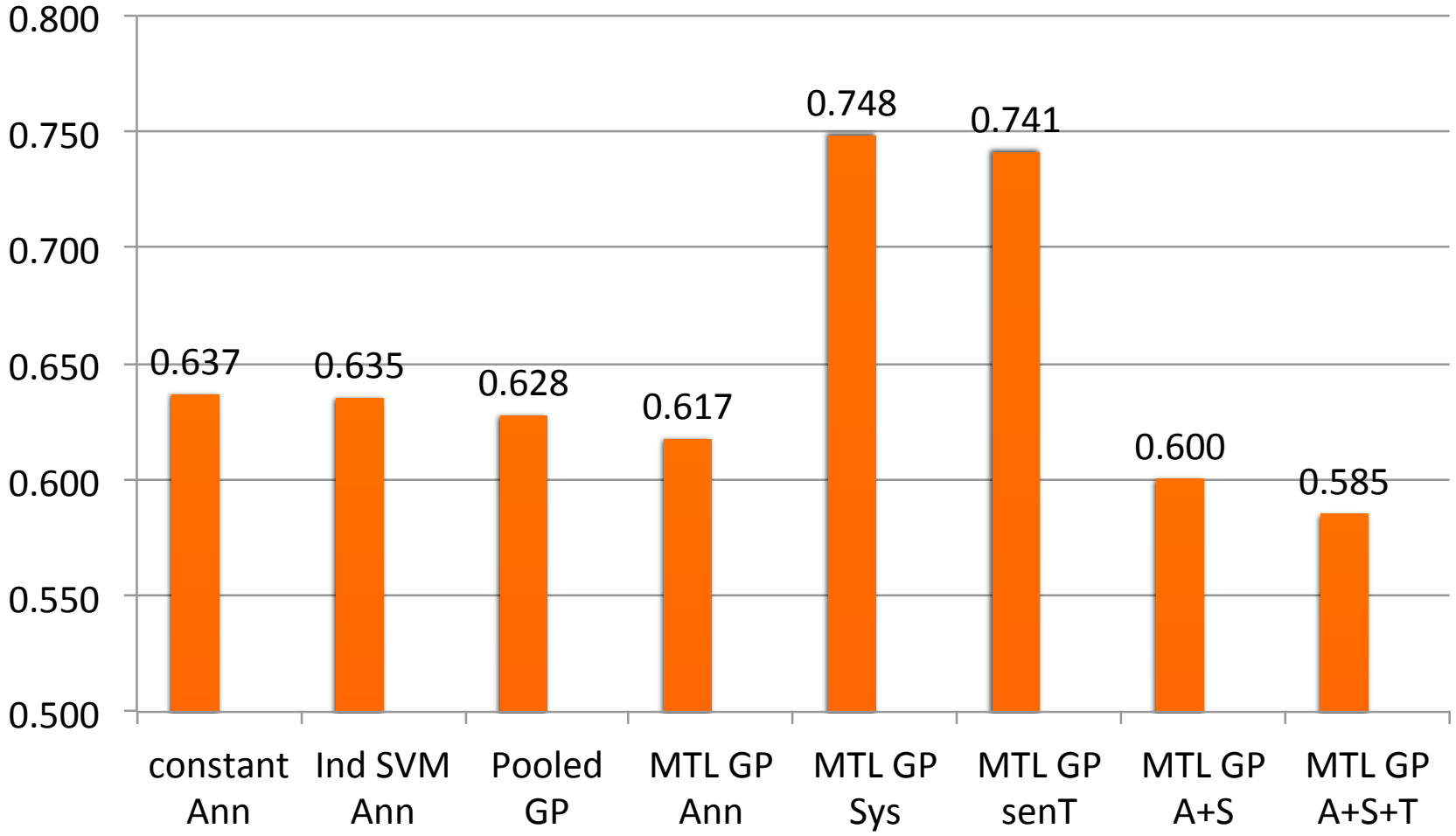
Results: WMT12 RMSE for 1-5 ratings



Incorporating layers of task metadata



Results: WPTP12 RMSE post-editing time



Extensions and wider applications

Application to data annotation

- crowd-sourcing: dealing with noise and biases
  Rogers et al. (2010); Groot et al. (2011)
- intelligent 'active' data acquisition

Joint learning of correlated phenomena in NLP

- domain adaptation
- same data annotated for many things (PTB etc)
- multi-lingual applications and language universals

Applicable with other likelihoods, e.g.,

- classification
- ordinal regression (ranking)
- structured prediction

Outline

Case study: Temporal patterns of words

Categorising temporal patterns of hashtags in Twitter

- collect hashtag normalised frequency time series for months
- use models learnt on past frequencies to forecast future frequencies
- identify and group similar temporal patterns
- emphasise periodicities in word frequencies

Instance of a forecasting problem

- emphasis on forecasting (extrapolation)
- different effects modelled by specific kernels

See our paper Preoţiuc-Pietro and Cohn (2013), EMNLP.

Model selection

Although parameter free, we still need to specify to a GP:

- the kernel parameters a.k.a. hyper-parameters θ
- the kernel definition $H_i \in \mathcal{H}$

Training a GP = selecting the kernel and its parameters

Can use only training data (and no validation)

Bayesian model selection

Marginal likelihood or Bayesian evidence:

$$p(y|x, \theta, H_i) = \int_f p(y|X, f, H_i) p(f|\theta, H_i)$$

The posterior over the hyperparameters is hard to compute due to the integral in the denominator:

$$p(\theta|y, x, H_i) = \frac{p(y|x, \theta, H_i) p(\theta|H_i)}{\int p(y|x, \theta, H_i) p(\theta|H_i) d\theta}$$

We approximate it by maximising over the Bayesian evidence (type II maximum likelihood - ML-II)

Bayesian model selection

For GP regression, the negative log of the evidence (a.k.a. NLML) can be computed analytically:

$$-\log(p(y|x,\theta)) = \frac{1}{2}y^T K_y^{-1} y + \frac{1}{2}\log|K_y| + \frac{n}{2}log2\pi$$

where $K_y = K_f + \sigma_n^2 I$ and K_f is the covariance matrix for the latent function f

Bayesian model selection

The posterior for a model given the data is:

$$p(H_i|y,x) = \frac{p(y|x,H_i)p(H_i)}{p(y|x)}$$

Assuming the prior over models is flat:

$$p(H_i|y,x) \propto p(y|x,H_i) = \int_\theta p(y|x,\theta,H_i)p(\theta|H_i)$$

Bayesian model selection

Occam's razor: 'the simplest solution is to be preferred over a more complex one'

The evidence must normalise:

- automatic trade-off between data-fit and model complexity
- complex models are penalised because they can describe many datasets
- simple models can describe a few datasets, thus the chance of a good data fit is low
- can be thought as the probability that a random draw of a function from the model can generate the training set

Identifying temporal patterns in word frequencies

Word/hashtag frequencies in Twitter

- very time dependent
- many 'live' only for hours reflecting timely events or memes
- some hashtags are constant over time
- some experience bursts at regular time intervals
- some follow human activity cycles

Can we automatically forecast future hashtag frequencies?

Can we automatically categorise temporal patterns?

Twitter hashtag temporal patterns

Regression task

- Extrapolation: forecast future frequencies
- using predictive mean

Dataset

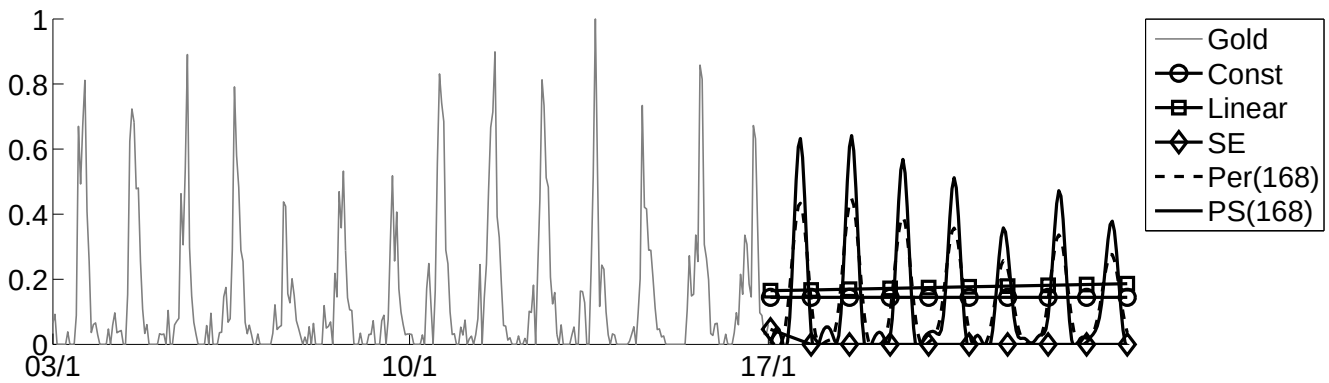
- two months of Twitter Gardenhose (10%)
- first month for training, second month for testing
- 1176 hashtags occurring in both splits
- $\sim$ 6.5 million tweets
- 5456 tweets/hashtag

Kernels

The kernel

- induces the covariance in the response between pairs of data points
- encodes the prior belief on the type of function we aim to learn
- for extrapolation, kernel choice is paramount
- different kernels are suitable for each specific category of temporal patterns: isotropic, smooth, periodic, non-stationary, etc.

Kernels



#goodmorning

	Const	Linear	SE	Per	PS
NLML	-41	-34	-176	-180	**-192**
NRMSE	0.213	0.214	0.262	0.119	**0.107**

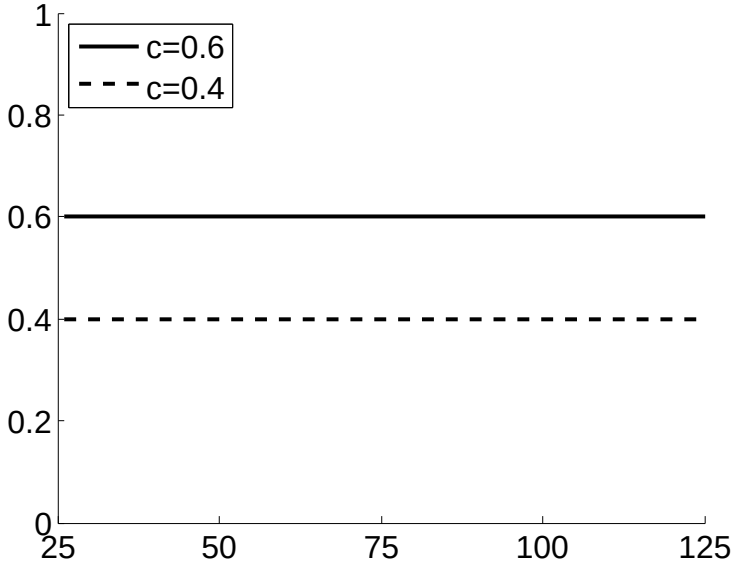
Lower is better

Use Bayesian model selection techniques to choose between kernels

Kernels: Constant

$$k_C(x, x') = c$$

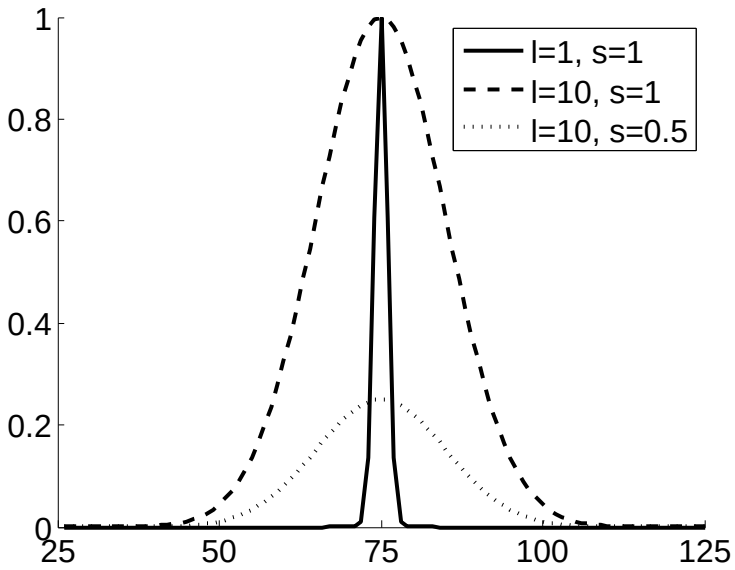
- constant relationship between outputs
- predictive mean is the value c
- assumes signal is modelled by Gaussian noise centred around the value c



Kernels: Squared exponential

$$k_{SE}(x, x') = s^2 \cdot \exp\left(-\frac{(x - x')^2}{2l^2}\right)$$

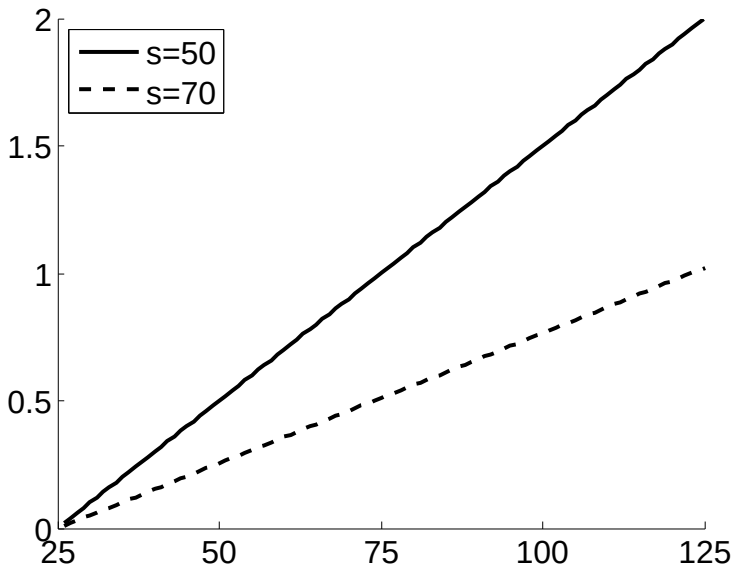
- smooth transition between neighbouring points
- best describes time series with a smooth shape e.g. uni-modal burst with a steady decrease
- predictive variance increases exponentially with distance



Kernels: Linear

$$k_{Lin}(x, x') = \frac{|x \cdot x'| + 1}{s^2}$$

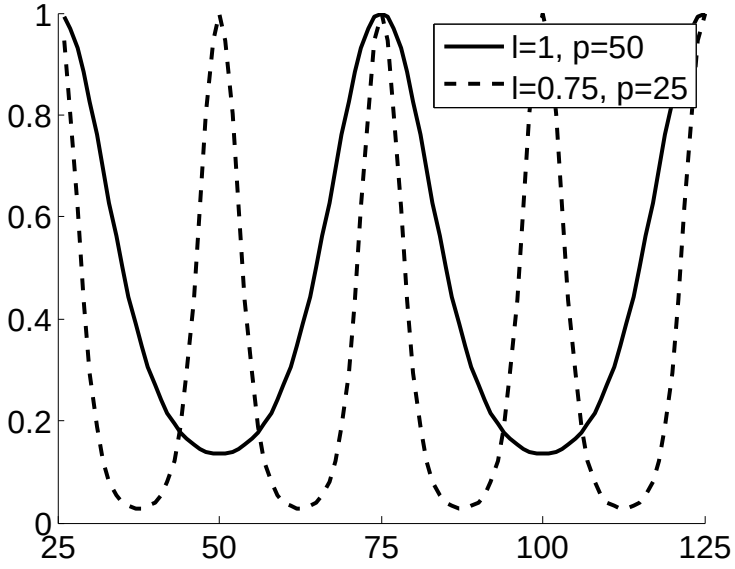
- non-stationary kernel: covariance depends on the data points values, not only on their difference $|t - t'|$
- equivalent to Bayesian linear regression with $\mathcal{N}(0, 1)$ priors on the regression weights and a prior of $\mathcal{N}(0, s^2)$ on the bias



Kernels: Periodic

$$k_{PER}(x, x') = s^2 \cdot \exp \cdot \left(-\frac{2\sin^2(2\pi(x - x')/p)}{l^2} \right)$$

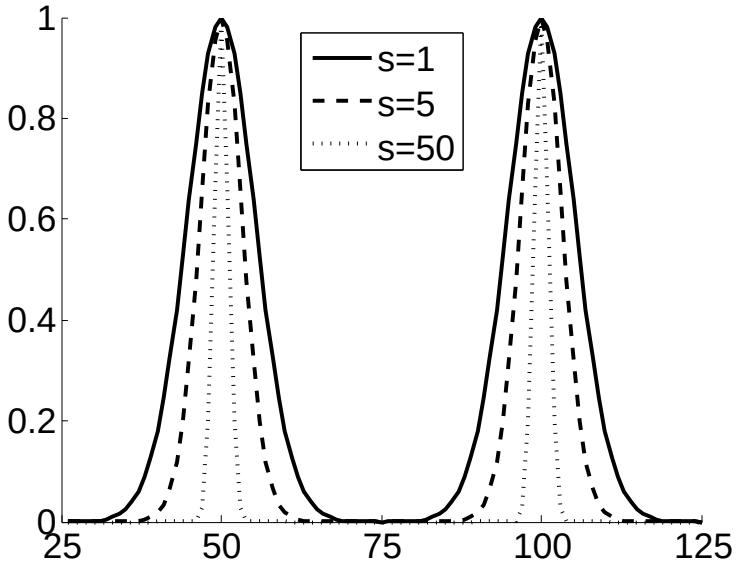
- s and l are characteristic length-scales
- p is the period (distance between consecutive peaks)
- best describes periodic patterns that oscillate smoothly between high and low values



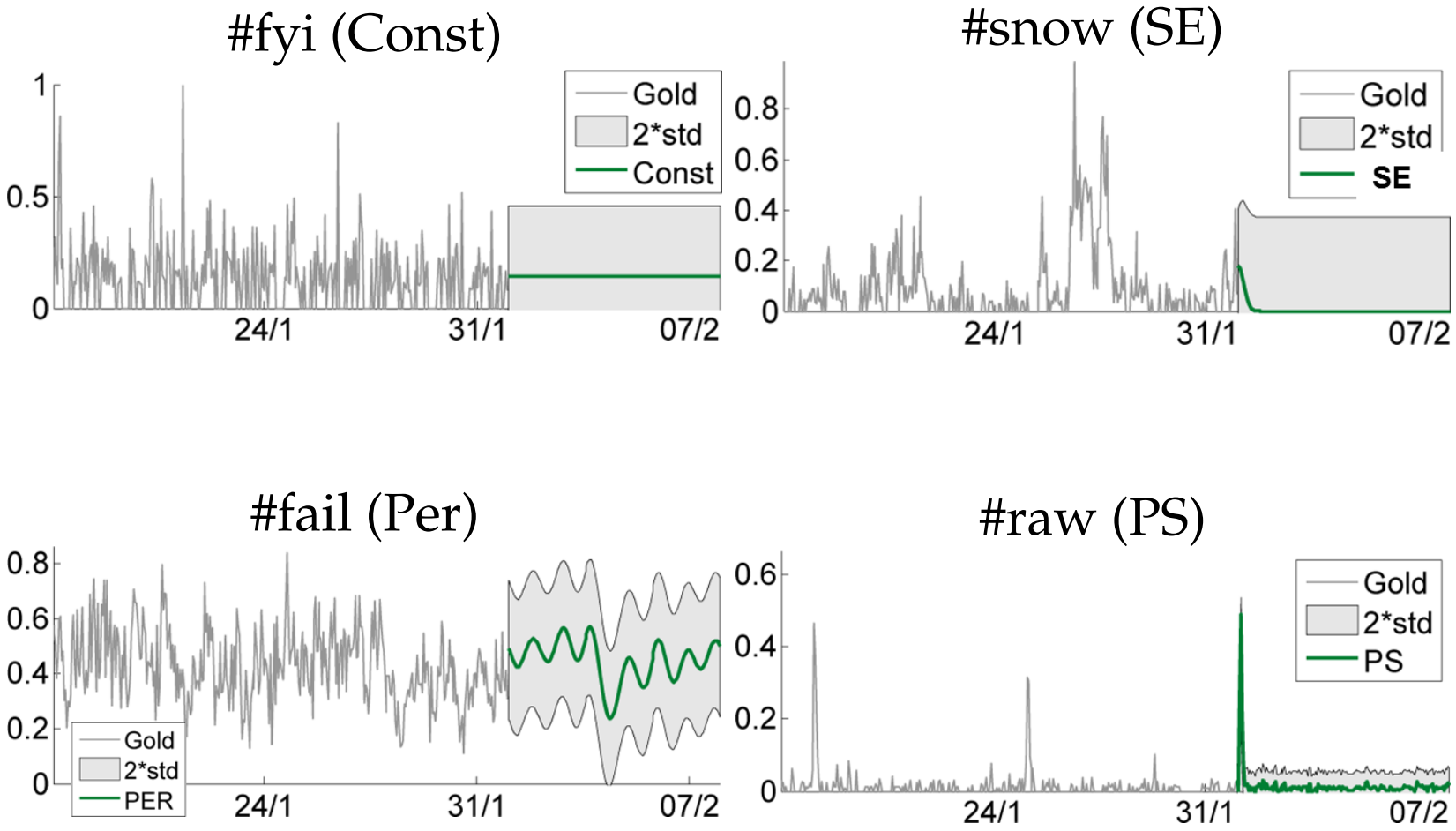
Kernels: Periodic Spikes

$$k_{PS}(x, x') = \cos\left(\sin\left(\frac{2\pi \cdot (x - x')}{p}\right)\right) \cdot \exp\left(\frac{s\cos(2\pi \cdot (x - x'))}{p} - s\right)$$

- p is the period
- s is a shape parameter controlling the width of the spike
- best describes time series with constant low values, followed by abrupt periodic rise



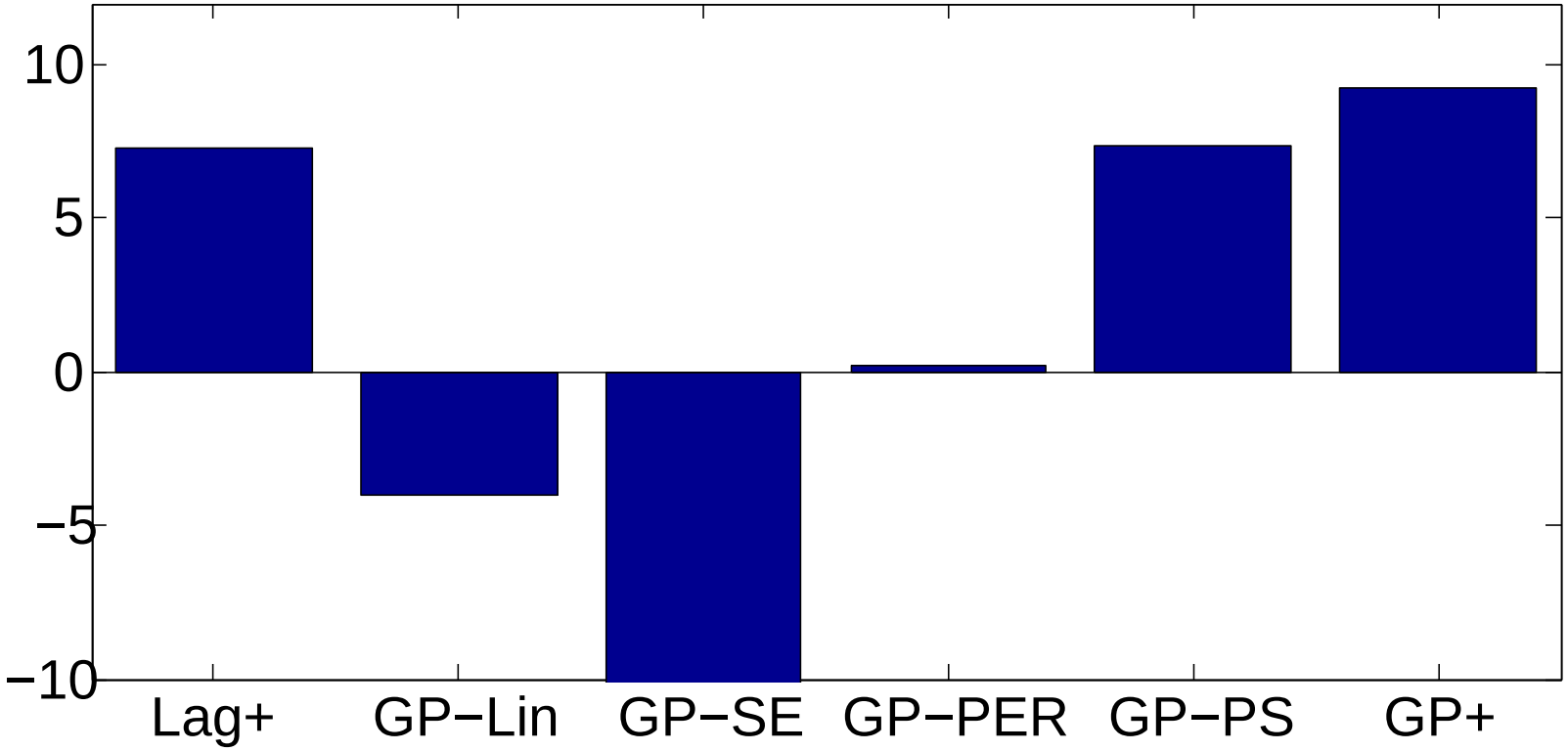
Results: Examples



Results: Categories

Const	SE	PER	PS
#funny	#2011	#brb	#ff
#lego	#backintheday	#coffee	#followfriday
#likeaboss	#confessionhour	#facebook	#goodnight
#money	#februarywish	#facepalm	#jobs
#nbd	#haiti	#fail	#news
#nf	#makeachange	#love	#nowplaying
#notetoself	#questionsidontlike	#rock	#tgif
#priorities	#savelibraries	#running	#twitterafterdark
#social	#snow	#xbox	#twitteroff
#true	#snowday	#youtube	#ww
49	**268**	**493**	**366**

Results: Forecasting



Application: Text classification

Task

- assign the hashtag of a given tweet based on its text

Methods

- Most frequent (MF)
- Naive Bayes model with empirical prior (NB-E)
- Naive Bayes with GP forecast as prior (NB-P)

	MF	**NB-E**	**NB-P**
Match@1	7.28%	16.04%	**17.39%**
Match@5	19.90%	29.51%	**31.91%**
Match@50	44.92%	59.17%	**60.85%**
MRR	0.144	0.237	**0.252**

Higher is better

References

Alvarez, M. A., Rosasco, L., and Lawrence, N. D. (2011). Kernels for vector-valued functions: A review. *Foundations and Trends in Machine Learning*, 4(3):195–266.

Bonilla, E., Chai, K. M., and Williams, C. (2008). Multi-task Gaussian process prediction. NIPS.

Cohn, T. and Specia, L. (2013). Modelling annotator bias with multi-task Gaussian processes: An application to machine translation quality estimation. ACL.

Daumé III, H. (2007). Frustratingly easy domain adaptation. ACL.

Evgeniou, T., Micchelli, C. A., and Pontil, M. (2006). Learning multiple tasks with kernel methods. *Journal of Machine Learning Research*, 6(1):615.

Groot, P., Birlutiu, A., and Heskes, T. (2011). Learning from multiple annotators with gaussian processes. ICANN.

Lampos, V., Aletras, N., Preoţiuc-Pietro, D., and Cohn, T. (2014). Predicting and Characterising User Impact on Twitter. EACL.

Preoţiuc-Pietro, D. and Cohn, T. (2013). A temporal model of text periodicities using Gaussian Processes. EMNLP.

Quiñonero Candela, J. and Rasmussen, C. E. (2005). A unifying view of sparse approximate gaussian process regression. *Journal of Machine Learning Research*, 6:1939–1959.

Rogers, S., Girolami, M., and Polajnar, T. (2010). Semi-parametric analysis of multi-rater data. *Statistics and Computing*, 20(3):317–334.

Specia, L., Shah, K., De Souza, J. G., and Cohn, T. (2013). Quest-a translation quality estimation framework. Citeseer.

Specia, L., Turchi, M., Cancedda, N., Dymetman, M., and Cristianini, N. (2009). Estimating the sentence-Level quality of Machine Translation systems. EAMT.

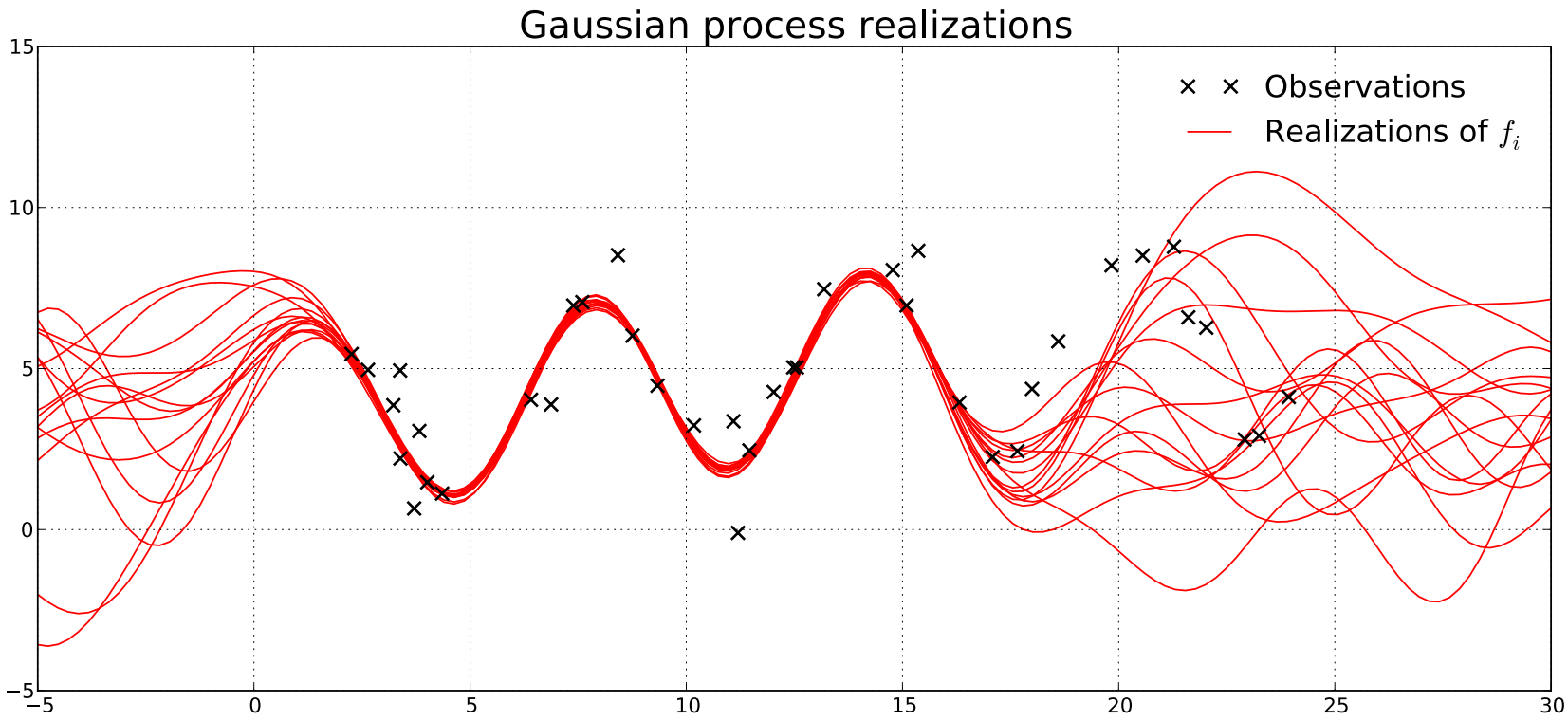
Outline

Outline

Recap: Regression

Observations, y_i, are a noisy version of latent process f_i,

$$y_i = f_i(\mathbf{x}_i) + \epsilon_i, \quad \text{with } \epsilon_i \sim \mathcal{N}(0, \sigma^2)$$



Likelihood models

Analytic solution for Gaussian likelihood *aka* noise

$$\begin{aligned} &\text{Gaussian (process) prior} \\ \times\ &\text{Gaussian likelihood} \\ =\ &\text{Gaussian posterior} \end{aligned}$$

But what about other likelihoods?

- Counts $\mathbf{y} \in \mathcal{N}$
- Classification $\mathbf{y} \in \{C_1, C_2, ..., C_k\}$
- Ordinal regression (ranking) $C_1 < C_2 < ... < C_k$
- ...

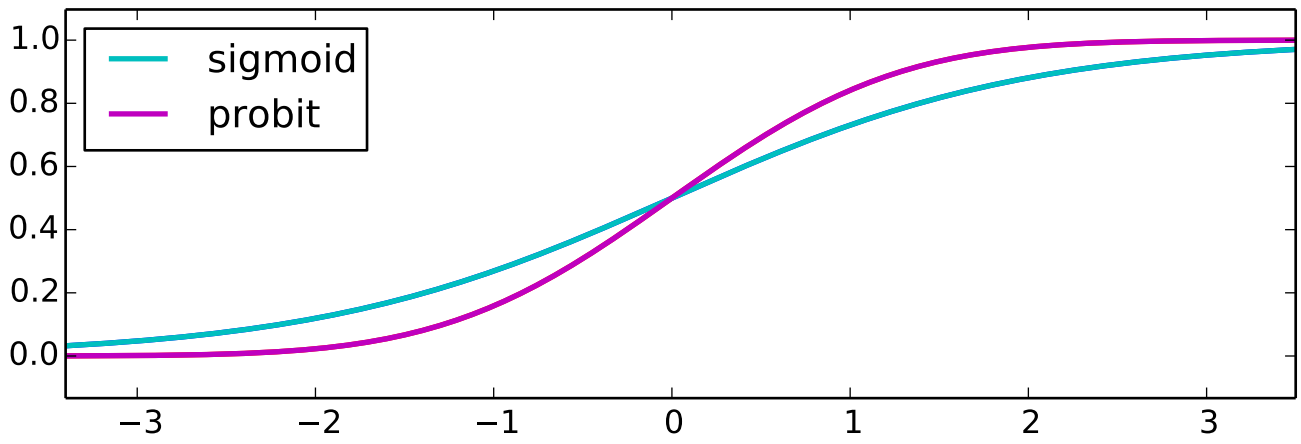
Classification

Binary classification, $y_i \in \{0, 1\}$.

Two popular choices for the likelihood

- Logistic sigmoid: $p(y_i = 1|f_i) = \sigma(f_i) = \frac{1}{1+\exp(-f_i)}$
- Probit function: $p(y_i = 1|f_i) = \Phi(f_i) = \int_{-\infty}^{f_i} \mathcal{N}(z|0, 1)dz$

"Squashing" input from $(-\infty, \infty)$ into range $[0, 1]$



Squashing function

Pass latent function through logistic function to obtain probability, $\pi(x) = p(y_i = 1|f_i)$

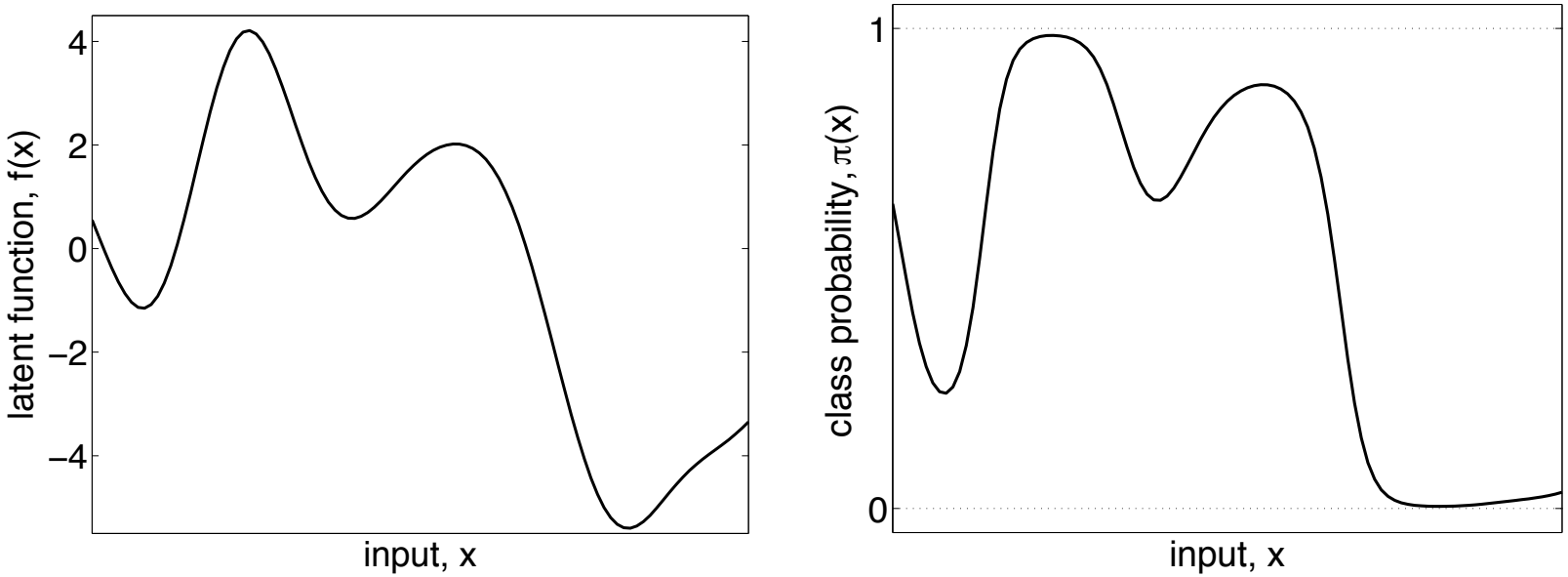


Figure from Rasmussen and Williams (2006)

Inference Challenges: for test case $\mathbf{x}_*$

Distribution over latent function

$$p(f^*|X, \mathbf{y}, \mathbf{x}_*) = \int p(f^*|X, \mathbf{x}_*, \mathbf{f}) \underbrace{p(\mathbf{f}|X, \mathbf{y})}_{\text{posterior}} d\mathbf{f}$$

Distribution over classification output

$$p(y_* = 1|X, \mathbf{y}, \mathbf{x}_*) = \int \sigma(f_*) p(f_*|X, \mathbf{y}, \mathbf{x}_*) df_*$$

Problem: likelihood no longer conjugate with prior, so no analytic solution.

Approximate inference

Several inference techniques proposed for non-conjugate likelihoods:

- Laplace approximation
  Williams and Barber (1998)
- Expectation propagation
  Minka (2001)
- Variational inference
  Gibbs and MacKay (2000)
- MCMC
  Neal (1999)

And more, including sparse approaches for large scale application.

Laplace approximation

Approximate non-Gaussian posterior by a Gaussian, centred at the mode

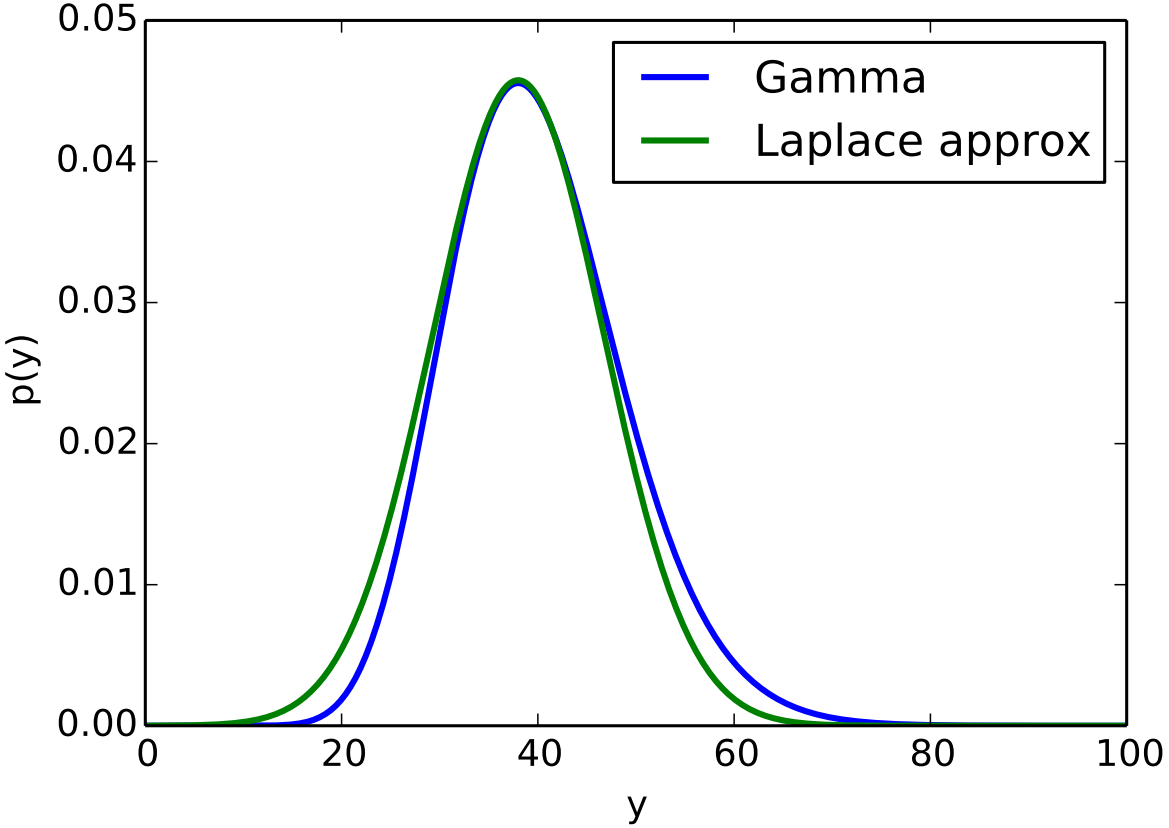


Figure from Rogers and Girolami (2012)

Laplace approximation

Log posterior

$$\Phi(\mathbf{f}) = \log p(\mathbf{y}|\mathbf{f}) + \log p(\mathbf{f}|X) + \text{const}$$

Find the posterior mode, $\hat{\mathbf{f}}$, i.e., MAP estimation, $O(n^3)$.

Then take a second order Taylor series expansion about mode, and fit with a Gaussian

- with mean, $\mu = \hat{\mathbf{f}}$
- and co-variance $\Sigma = (K^{-1} + \nabla\nabla \log p(\mathbf{y}|\mathbf{f}))^{-1}$

Allows for computation of posterior and marginal likelihood, but predictions may still be intractable.

Expectation propagation

Take intractable posterior:

$$p(\mathbf{f}|\mathbf{X}, \mathbf{y}) = \frac{1}{Z} p(\mathbf{f}|\mathbf{X}) \prod_{i=1}^{n} p(y_i|f_i)$$

$$Z = \int p(\mathbf{f}|\mathbf{X}) \prod_{i=1}^{n} p(y_i|f_i) d\mathbf{f}$$

Approximation with fully factorised distribution

$$q(\mathbf{f}|\mathbf{X}, \mathbf{y}) = \frac{1}{Z_{EP}} p(\mathbf{f}|\mathbf{X}) \prod_{i=1}^{n} t(f_i)$$

Expectation propagation

Approximate posterior defined as

$$q(\mathbf{f}|\mathbf{y}) = \frac{1}{Z_{EP}} p(\mathbf{f}) \prod_{i=1}^{n} t(f_i)$$

where each component assumed to be Gaussian

- $p(y_i|f_i) \approx t_i(f_i) = \tilde{Z}_i \mathcal{N}(f_i|\tilde{\mu}_i, \tilde{\sigma}_i^2)$
- $p(\mathbf{f}|\mathbf{X}) \sim \mathcal{N}(\mathbf{f}|\mathbf{0}, K_{nn})$

Results in Gaussian formulation for $q(\mathbf{f}|\mathbf{y})$

- allows for tractable multiplication, division with Gaussians
- and marginalisation, expectations etc

Expectation propagation

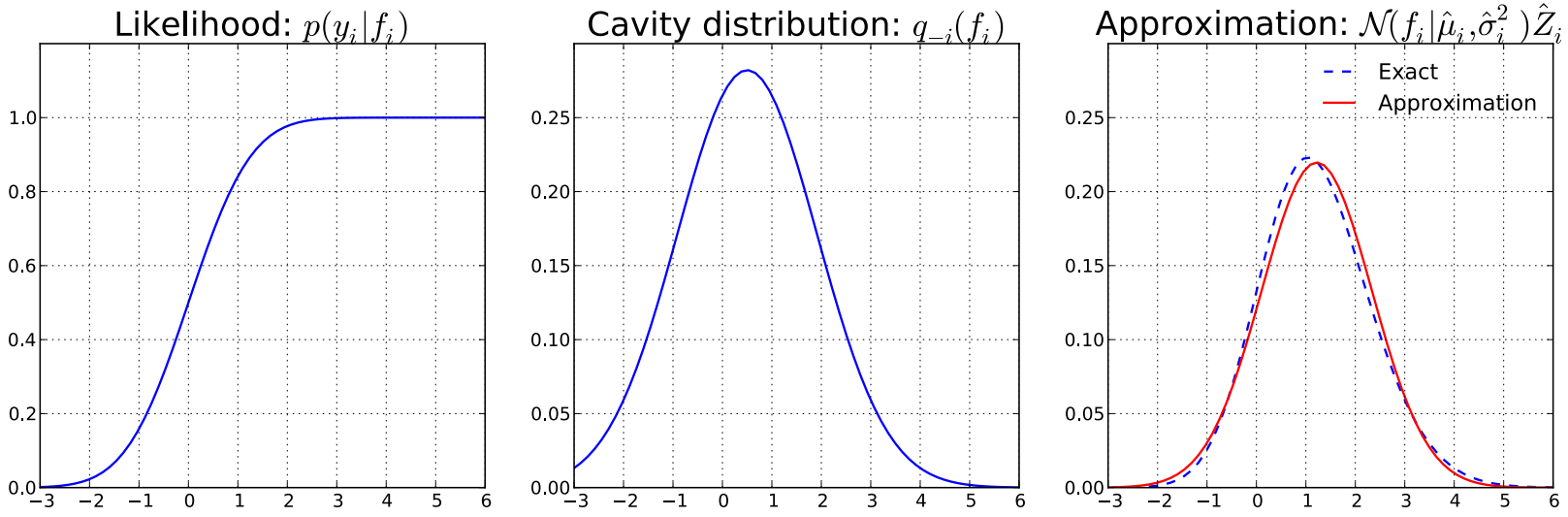
EP algorithm aims to fit $t_i(f_i)$ to the posterior, starting with a guess for q, then iteratively refining as follows

- minimise KL divergence between the true posterior for f_i and the approximation, t_i

$$\min_{t_i} \mathrm{KL}\left(p(y_i|f_i)q_{-i}(f_i) \,\|\, t_i(f_i)q_{-i}(f_i)\right)$$

  where $q_{-i}(f_i)$ is the cavity distribution formed by marginalising $q(\mathbf{f})$ over $f_j,\ j \neq i$ then dividing by $t_i(f_i)$.
- key idea: only need accurate approximation for globally feasible f_i
- match moments to update t_i, then update $q(\mathbf{f})$

Expectation propagation



Expectation propagation

No proof of convergence

- but empirically works well
- often more accurate than Laplace approximation

Formulated for many different likelihoods

- complexity $O(n^3)$, dominated by matrix inversion
- sparse EP approximations can reduce this to $O(nm^2)$

See Minka (2001) and Rasmussen and Williams (2006) for further details.

Multi-class classification

Consider multi-class classification, $y \in \{C_1, C_2, \ldots, C_k\}$.

Draw vector of k latent function values for each input

$$\mathbf{f} = \left(f_1^1, \ldots, f_n^1, f_1^2, \ldots, f_n^2, f_1^k, \ldots, f_n^k\right)$$

Formulate clasification probability using soft-max

$$p(y_i = c|\mathbf{f}_i) = \frac{\exp(f_i^c)}{\sum_{c'} \exp(f_i^{c'})}$$

Multi-class classification

Assume k latent processes are uncorrelated, leading to prior covariance $\mathbf{f} \sim \mathcal{N}(\mathbf{0}, K)$ where

$$K = \begin{pmatrix} K_1 & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{0} & K_2 & \cdots & \mathbf{0} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \cdots & K_k \end{pmatrix}$$

is block diagonal $kn \times kn$ with each K_j of size $n \times n$.

Various approximation methods for inference, e.g., Laplace (Williams and Barber, 1998), EP (Kim and Ghahramani, 2006), MCMC (Neal, 1999).

Outline

GPs for Structured Prediction

- GPSC (Altun et al., 2004):
  - Defines a likelihood over label sequences: $p(\mathbf{y}|\mathbf{x})$, with latent variable over full sequences $\mathbf{y}$
  - HMM-inspired kernel, combining features from each observed symbol x_i and label pairs
  - MAP inference for hidden function values, $\mathbf{f}$, and sparsification trick for tractable inference
- GPstruct (Bratières et al., 2013):
  - Base model is a CRF:

$$p(\mathbf{y}|\mathbf{x}, \mathbf{f}) = \frac{\exp \sum_c f(c, \mathbf{x}_c, \mathbf{y}_c)}{\sum_{\mathbf{y}' \in \mathcal{Y}} \exp \sum_c f(c, \mathbf{x}_c, \mathbf{y}'_c)}$$

  - Assumes that each potential $f(c, \mathbf{x}_c, \mathbf{y}_c)$ is drawn from a GP
  - Bayesian inference using MCMC (Murray et al., 2010)

Outline

String Kernels

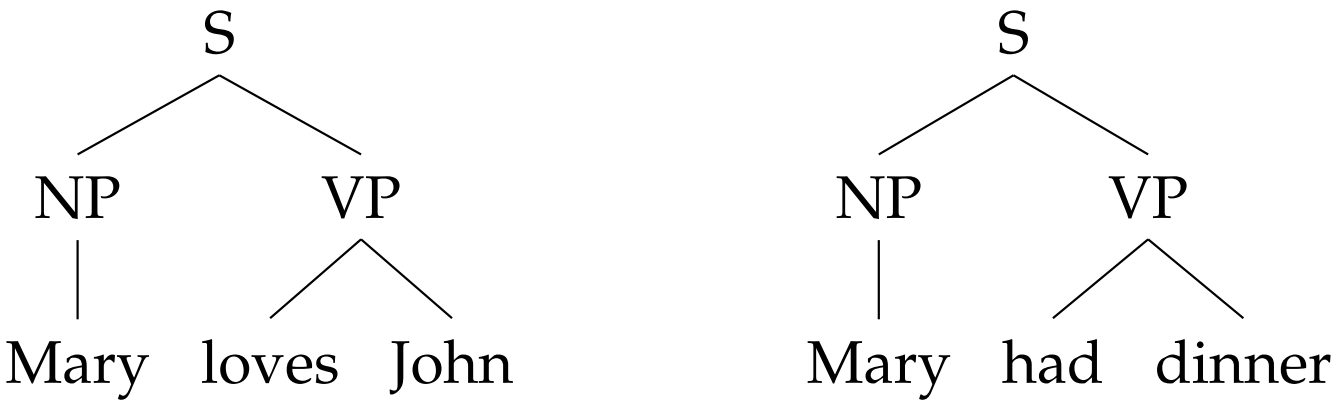
$$k(x, x') = \sum_{s \in \Sigma^*} w_s \phi_s(x) \phi_s(x'),$$

- $\phi_s(x)$: counts of substring s inside x;
- $0 \leq w_s \leq 1$: weight of substring s;
- s can also be a subsequence (containing gaps);

- s = char sequences → ngram kernels (Lodhi et al., 2002) (useful for stems);
  k(**ba**r, **ba**t) = 3 (**b**,**a**,**ba**)
- s = word sequences → Word Sequence kernels (Cancedda et al., 2003);
  k(**gas** only **injection**, **gas** assisted plastic **injection**) = 3
- Soft matching:
  k(battle, battles) ≠ 0
  k(battle, combat) ≠ 0

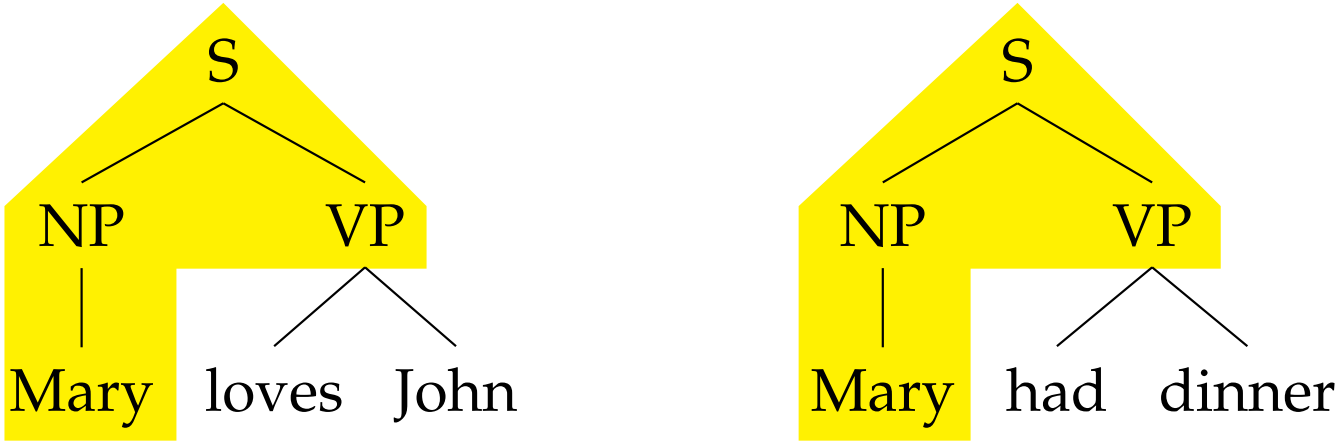
Tree Kernels

- Subset Tree Kernels (Collins and Duffy, 2001)



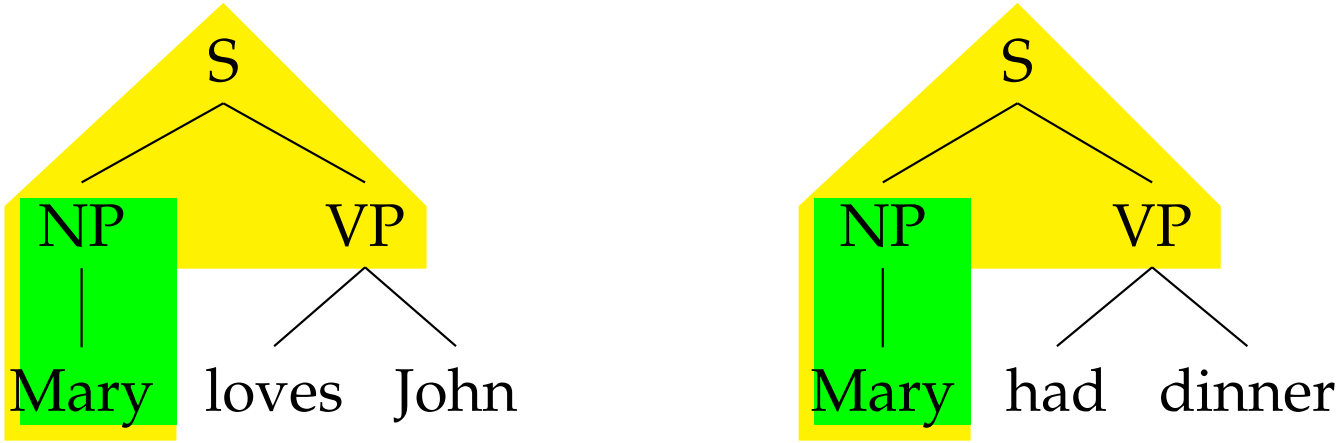
Tree Kernels

- Subset Tree Kernels (Collins and Duffy, 2001)



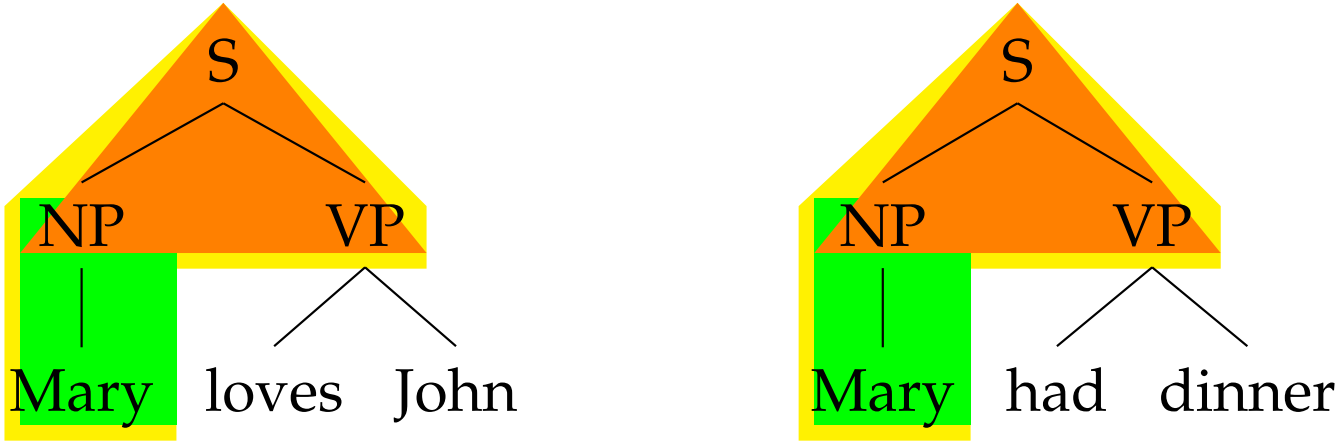
Tree Kernels

- Subset Tree Kernels (Collins and Duffy, 2001)



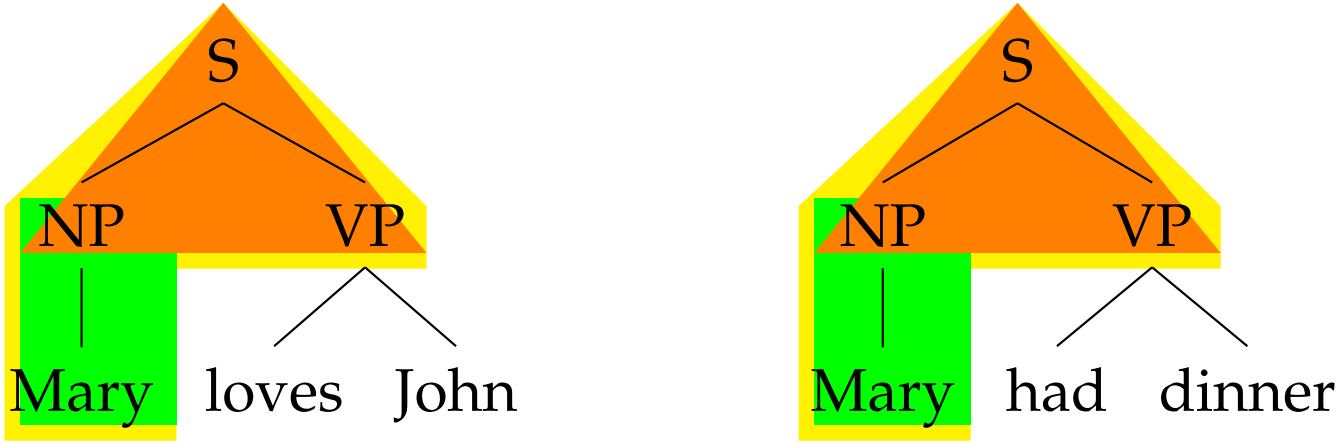
Tree Kernels

- Subset Tree Kernels (Collins and Duffy, 2001)



Tree Kernels

- Subset Tree Kernels (Collins and Duffy, 2001)



- Partial Tree Kernels (Moschitti, 2006): allows "broken" rules, useful for dependency trees;
- Soft matching can also be applied.

More on GPs + structured kernels in Daniel Beck's SRW presentation tomorrow

References I

Altun, Y., Hofmann, T., and Smola, A. J. (2004). Gaussian Process Classification for Segmenting and Annotating Sequences. In *Proceedings of ICML*, page 8, New York, New York, USA. ACM Press.

Bratières, S., Quadrianto, N., and Ghahramani, Z. (2013). Bayesian Structured Prediction using Gaussian Processes. *arXiv:1307.3846*, pages 1–17.

Cancedda, N., Gaussier, E., Goutte, C., and Renders, J.-M. (2003). Word-Sequence Kernels. *The Journal of Machine Learning Research*, 3:1059–1082.

Collins, M. and Duffy, N. (2001). Convolution Kernels for Natural Language. In *Advances in Neural Information Processing Systems*.

Gibbs, M. N. and MacKay, D. J. (2000). Variational gaussian process classifiers. *IEEE Transactions on Neural Networks*, 11(6):1458–1464.

Kim, H.-C. and Ghahramani, Z. (2006). Bayesian gaussian process classification with the em-ep algorithm. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 28(12):1948–1959.

References II

Lodhi, H., Saunders, C., Shawe-Taylor, J., Cristianini, N., and Watkins, C. (2002). Text Classification using String Kernels. *The Journal of Machine Learning Research*, 2:419–444.

Minka, T. (2001). Expectation propagation for approximate bayesian inference. In *Proceedings of the Seventeenth conference on Uncertainty in Artificial Intelligence*, pages 362–369. Morgan Kaufmann Publishers Inc.

Moschitti, A. (2006). Making Tree Kernels practical for Natural Language Learning. In *EACL*, pages 113–120.

Murray, I., Adams, R. P., and Mackay, D. (2010). Elliptical slice sampling. In *International Conference on Artificial Intelligence and Statistics*, pages 541–548.

Neal, R. (1999). Regression and classification using gaussian process priors. *Bayesian Statistics*, 6.

Rasmussen, C. E. and Williams, C. K. (2006). *Gaussian processes for machine learning*, volume 1. MIT press Cambridge, MA.

Rogers, S. and Girolami, M. (2012). *A First Course in Machine Learning*. Chapman & Hall/CRC.

References III

Williams, C. K. and Barber, D. (1998). Bayesian classification with gaussian processes. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 20(12):1342–1351.