Special course on Gaussian processes: Session #1

Michael Riis Andersen

Aalto University

michael.riis@gmail.com

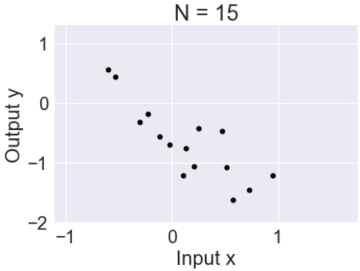
9/1-19

Agenda for today

1. Motivation for Gaussian processes
2. Course content, format, and evaluation
3. Warm up for Gaussian processes: Review of the multivariate Gaussian distribution
4. First assignment

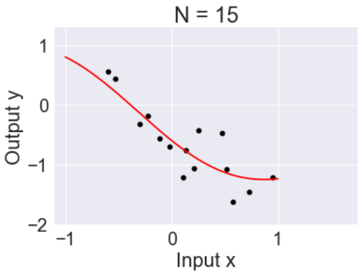
Gaussian processes in a nutshell

- It's all about learning functions from data
- Suppose we are given a data set $\mathcal{D} = \{\boldsymbol{x}_n, y_n\}_{n=1}^{N}$



Gaussian processes in a nutshell

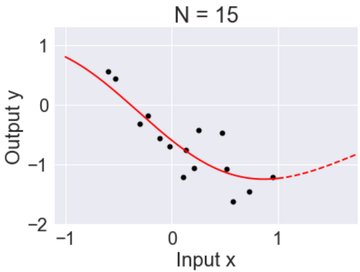
- It's all about learning functions from data
- Suppose we are given a data set $\mathcal{D} = \{\boldsymbol{x}_n, y_n\}_{n=1}^N$



- Gaussian processes (GPs) can
  - ... fit non-linear functions to data

Gaussian processes in a nutshell

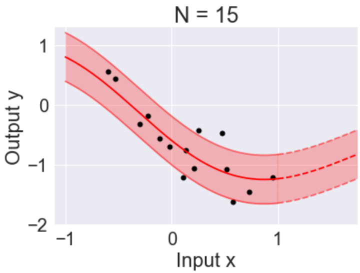
- It's all about learning functions from data
- Suppose we are given a data set $\mathcal{D} = \{\boldsymbol{x}_n, y_n\}_{n=1}^{N}$



- Gaussian processes (GPs) can
  - ... fit non-linear functions to data
  - ... make predictions for new inputs

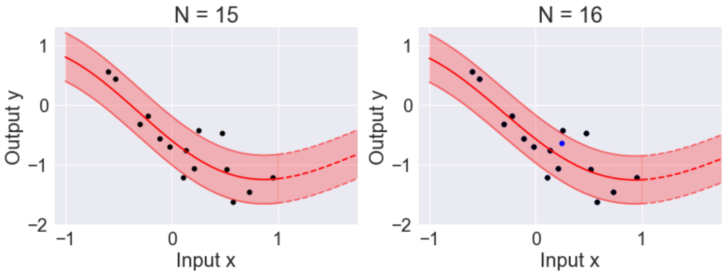
Gaussian processes in a nutshell

- It's all about learning functions from data
- Suppose we are given a data set $\mathcal{D} = \{\boldsymbol{x}_n, y_n\}_{n=1}^{N}$



- Gaussian processes (GPs) can
  - ... fit non-linear functions to data
  - ... make predictions for new inputs
  - ... provide sensible uncertainties

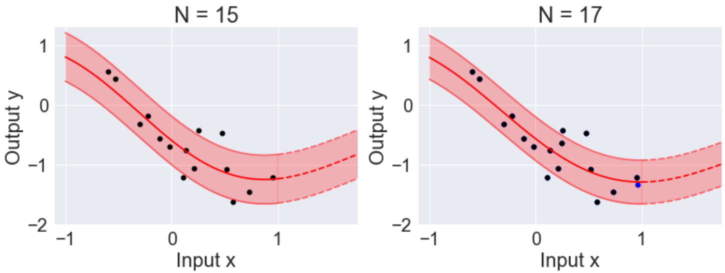
- It's all about learning functions from data
- Suppose we are given a data set $\mathcal{D} = \{\boldsymbol{x}_n, y_n\}_{n=1}^{N}$



- Gaussian processes (GPs) can
  - ... fit non-linear functions to data
  - ... make predictions for new inputs
  - ... provide sensible uncertainties
  - ... adjust model complexity to data (nonparametric)

Gaussian processes in a nutshell

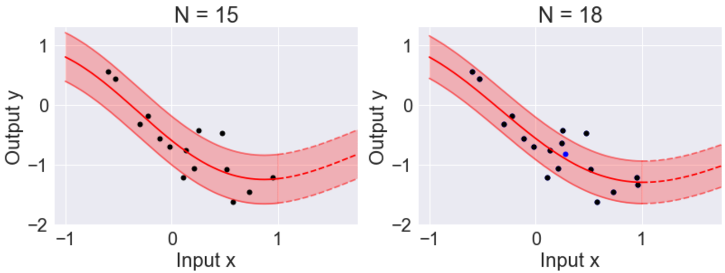
- It's all about learning functions from data
- Suppose we are given a data set $\mathcal{D} = \{\boldsymbol{x}_n, y_n\}_{n=1}^{N}$



- Gaussian processes (GPs) can
  - ... fit non-linear functions to data
  - ... make predictions for new inputs
  - ... provide sensible uncertainties
  - ... adjust model complexity to data (nonparametric)

Gaussian processes in a nutshell

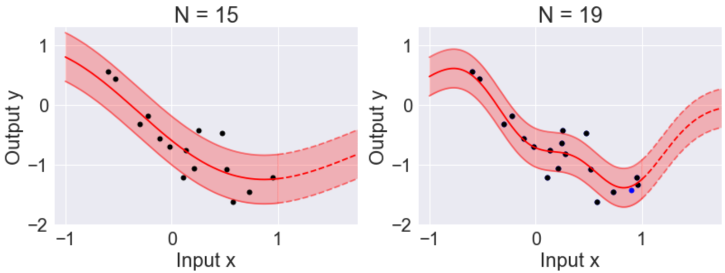
- It's all about learning functions from data
- Suppose we are given a data set $\mathcal{D} = \{\boldsymbol{x}_n, y_n\}_{n=1}^{N}$



- Gaussian processes (GPs) can
  - ... fit non-linear functions to data
  - ... make predictions for new inputs
  - ... provide sensible uncertainties
  - ... adjust model complexity to data (nonparametric)

Gaussian processes in a nutshell

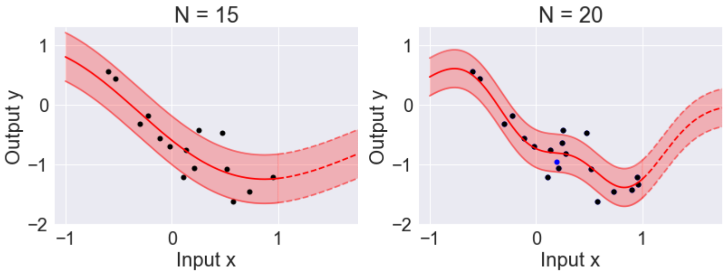
- It's all about learning functions from data
- Suppose we are given a data set $\mathcal{D} = \{\boldsymbol{x}_n, y_n\}_{n=1}^N$



- Gaussian processes (GPs) can
  - ... fit non-linear functions to data
  - ... make predictions for new inputs
  - ... provide sensible uncertainties
  - ... adjust model complexity to data (nonparametric)

Gaussian processes in a nutshell

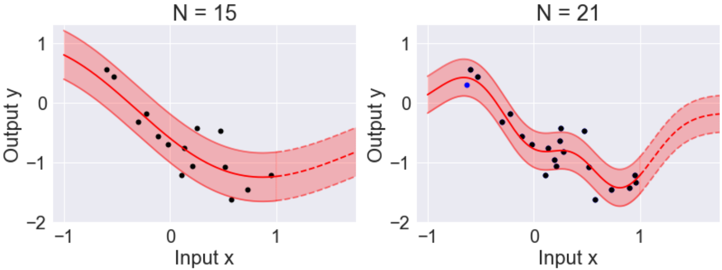
- It's all about learning functions from data
- Suppose we are given a data set $\mathcal{D} = \{\boldsymbol{x}_n, y_n\}_{n=1}^N$



- Gaussian processes (GPs) can
  - ... fit non-linear functions to data
  - ... make predictions for new inputs
  - ... provide sensible uncertainties
  - ... adjust model complexity to data (nonparametric)

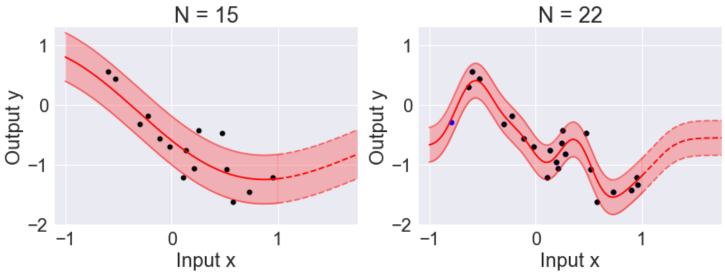
Gaussian processes in a nutshell

- It's all about learning functions from data
- Suppose we are given a data set $\mathcal{D} = \{\boldsymbol{x}_n, y_n\}_{n=1}^{N}$



- Gaussian processes (GPs) can
  - ... fit non-linear functions to data
  - ... make predictions for new inputs
  - ... provide sensible uncertainties
  - ... adjust model complexity to data (nonparametric)

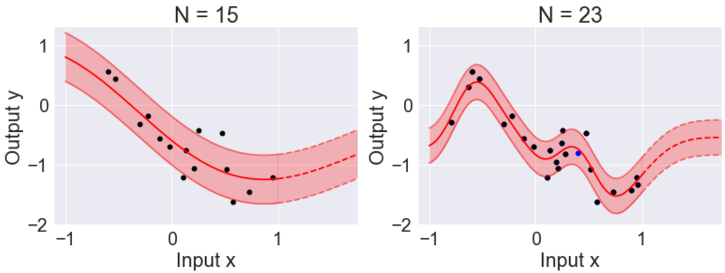
- It's all about learning functions from data
- Suppose we are given a data set $\mathcal{D} = \{\boldsymbol{x}_n, y_n\}_{n=1}^{N}$



- Gaussian processes (GPs) can
  - ... fit non-linear functions to data
  - ... make predictions for new inputs
  - ... provide sensible uncertainties
  - ... adjust model complexity to data (nonparametric)

Gaussian processes in a nutshell

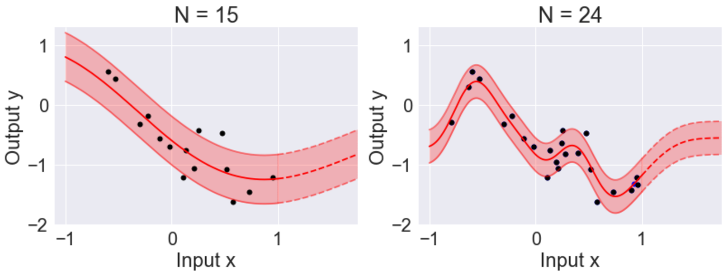
- It's all about learning functions from data
- Suppose we are given a data set $\mathcal{D} = \{\boldsymbol{x}_n, y_n\}_{n=1}^{N}$



- Gaussian processes (GPs) can
  - ... fit non-linear functions to data
  - ... make predictions for new inputs
  - ... provide sensible uncertainties
  - ... adjust model complexity to data (nonparametric)

Gaussian processes in a nutshell

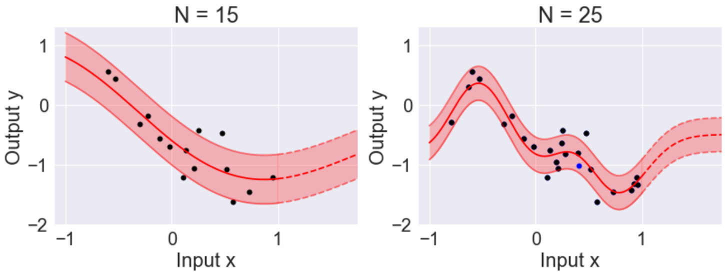
- It's all about learning functions from data
- Suppose we are given a data set $\mathcal{D} = \{\boldsymbol{x}_n, y_n\}_{n=1}^{N}$



- Gaussian processes (GPs) can
  - ... fit non-linear functions to data
  - ... make predictions for new inputs
  - ... provide sensible uncertainties
  - ... adjust model complexity to data (nonparametric)

Gaussian processes in a nutshell

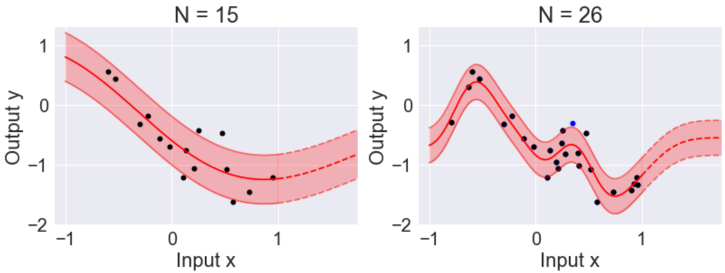
- It's all about learning functions from data
- Suppose we are given a data set $\mathcal{D} = \{\mathbf{x}_n, y_n\}_{n=1}^{N}$



- Gaussian processes (GPs) can
  - ... fit non-linear functions to data
  - ... make predictions for new inputs
  - ... provide sensible uncertainties
  - ... adjust model complexity to data (nonparametric)

Gaussian processes in a nutshell

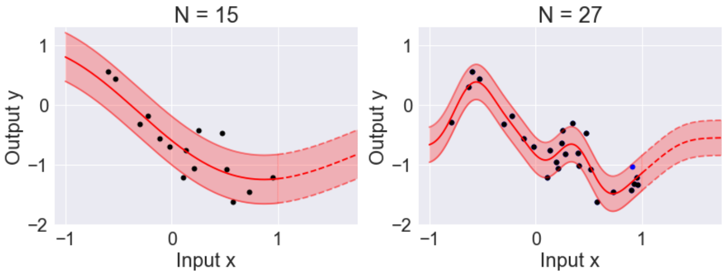
- It's all about learning functions from data
- Suppose we are given a data set $\mathcal{D} = \{\boldsymbol{x}_n, y_n\}_{n=1}^{N}$



- Gaussian processes (GPs) can
  - ... fit non-linear functions to data
  - ... make predictions for new inputs
  - ... provide sensible uncertainties
  - ... adjust model complexity to data (nonparametric)

Gaussian processes in a nutshell

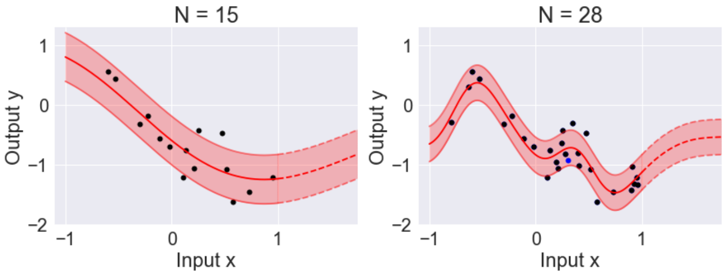
- It's all about learning functions from data
- Suppose we are given a data set $\mathcal{D} = \{\boldsymbol{x}_n, y_n\}_{n=1}^{N}$



- Gaussian processes (GPs) can
  - ... fit non-linear functions to data
  - ... make predictions for new inputs
  - ... provide sensible uncertainties
  - ... adjust model complexity to data (nonparametric)

Gaussian processes in a nutshell

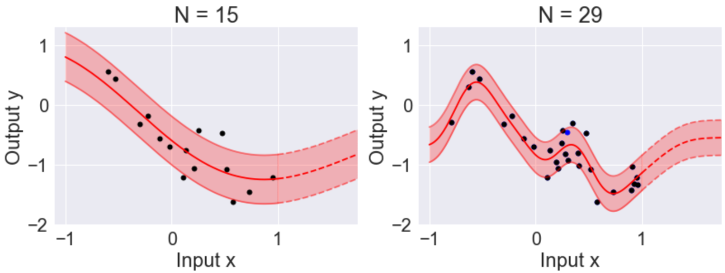
- It's all about learning functions from data
- Suppose we are given a data set $\mathcal{D} = \{\boldsymbol{x}_n, y_n\}_{n=1}^{N}$



- Gaussian processes (GPs) can
  - ... fit non-linear functions to data
  - ... make predictions for new inputs
  - ... provide sensible uncertainties
  - ... adjust model complexity to data (nonparametric)

Gaussian processes in a nutshell

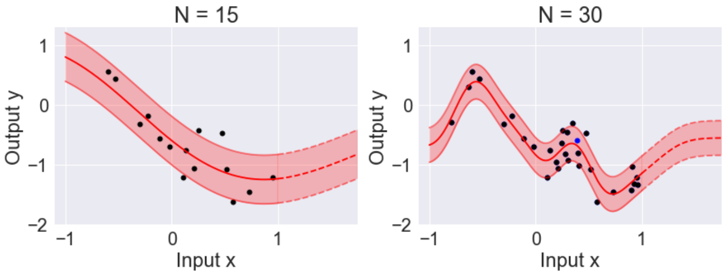
- It's all about learning functions from data
- Suppose we are given a data set $\mathcal{D} = \{\boldsymbol{x}_n, y_n\}_{n=1}^{N}$



- Gaussian processes (GPs) can
  - ... fit non-linear functions to data
  - ... make predictions for new inputs
  - ... provide sensible uncertainties
  - ... adjust model complexity to data (nonparametric)

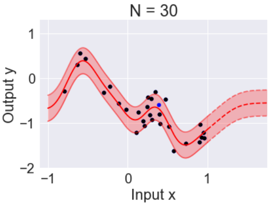
Gaussian processes in a nutshell

- It's all about learning functions from data
- Suppose we are given a data set $\mathcal{D} = \{\boldsymbol{x}_n, y_n\}_{n=1}^{N}$



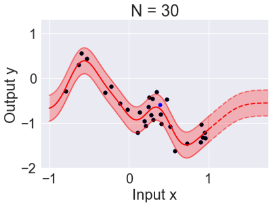
- Gaussian processes (GPs) can
  - ... fit non-linear functions to data
  - ... make predictions for new inputs
  - ... provide sensible uncertainties
  - ... adjust model complexity to data (nonparametric)

Functions with different domains

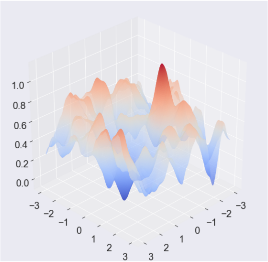


The real line

Functions with different domains

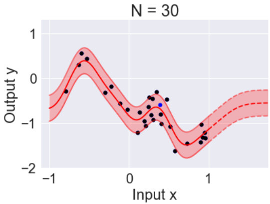


The real line

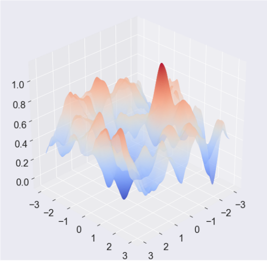


Higher dimensions

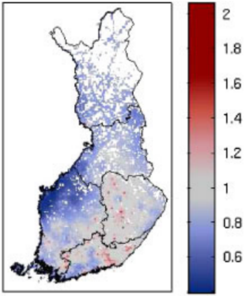
Functions with different domains



The real line

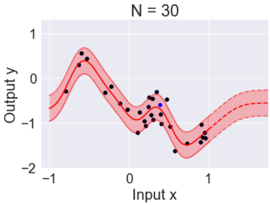


Higher dimensions

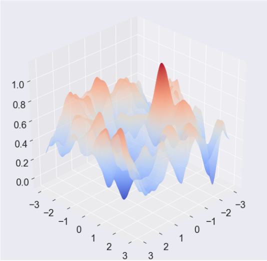


Finland

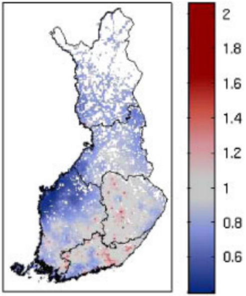
Functions with different domains



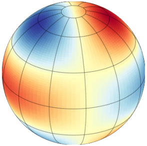
The real line



Higher dimensions

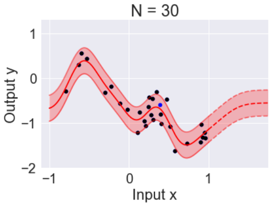


Finland

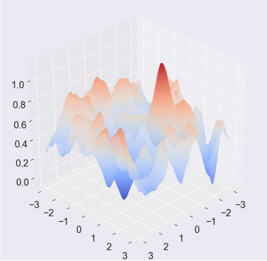


A sphere

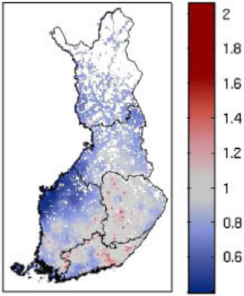
Functions with different domains



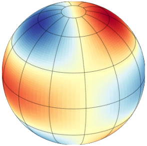
The real line



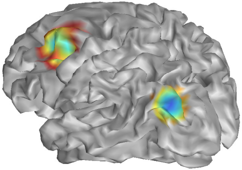
Higher dimensions



Finland



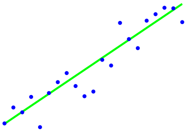
A sphere



A human brain

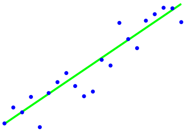
Multitude of Gaussian processes applications

- Regression (supervised learning)
  - Time series analysis
  - EEG brain imaging
  - Survival analysis for cancer data
  - Predicting rainfall
  - Robot dynamics
  - ...

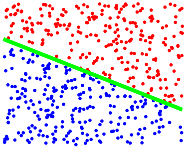


Multitude of Gaussian processes applications

- Regression (supervised learning)
    - Time series analysis
    - EEG brain imaging
    - Survival analysis for cancer data
    - Predicting rainfall
    - Robot dynamics
    - ...

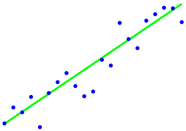


- Classification (supervised learning)
    - Recognizings handwritten digits
    - Brain decoding
    - ...

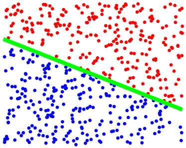


Multitude of Gaussian processes applications

- Regression (supervised learning)
  - Time series analysis
  - EEG brain imaging
  - Survival analysis for cancer data
  - Predicting rainfall
  - Robot dynamics
  - ...



- Classification (supervised learning)
  - Recognizings handwritten digits
  - Brain decoding
  - ...



- Dimensionality reduction (unsupervised learning)
- Optimization of black box functions (Bayesian optimization)
- Numerical integration (Bayesian quadrature)
- Solving differential equations (probabilistic numerics)

Course content

- The goal of the course is to introduce you to Gaussian processes, applications and some recent research directions
- We will cover
  1. ... Gaussian process regression & classification
  2. ... model section for Gaussian processes
  3. ... approximate inference & how to speed up GP inference
  4. ... spatio-temporal modelling
  5. ... some advanced topics based on your interests

Format of the course

- The course will be based on
  - shorts lectures
  - exercises (based on python notebooks)
  - project work + presentation in groups of 1-3 persons (optional)
- To pass the course, you need to
  - complete and hand in exercises
  - do project work (only for extra ECTS points)
  - 3 ECTS / 5 ECTS

Course plan

Lectures

- Lecture 1: Warm up: Properties of the multivariate normal distribution
- Lecture 2: Linear Gaussian models and intro to Gaussian processes
- Lecture 3: Kernels and model selection
- Lecture 4: Inducing points method (.. or how to make GPs faster)
- Lecture 5: Spectral kernels (.. or how to make GPs more flexible)
- Lecture 6: Spatio-temporal models

Assignments

- Assignment #1 due 23rd of January (midnight)
- Assignment #2 due 6th of February (midnight)
- Assignment #3 due 20th of February (midnight)

The properties of the multivariate Gaussian distribution

The multivariate Gaussian distribution

- **Definition** A random vector $\boldsymbol{x} = [x_1, x_2, \cdots, x_D]$ is said to have the multivariate Gaussian distribution if all linear combinations of $\boldsymbol{x}$ are Gaussian distributed:

$$y = \boldsymbol{a}^{\boldsymbol{T}} \boldsymbol{x} = a_1 x_1 + a_2 x_2 + \cdots + a_D x_D \sim \mathcal{N}(m, v)$$

  for all $\boldsymbol{a} \in \mathbb{R}^D$, where $\boldsymbol{a} \neq \mathbf{0}$

The multivariate Gaussian distribution

- **Definition** A random vector $\boldsymbol{x} = [x_1, x_2, \cdots, x_D]$ is said to have the multivariate Gaussian distribution if all linear combinations of $\boldsymbol{x}$ are Gaussian distributed:

$$y = \boldsymbol{a}^{\boldsymbol{T}} \boldsymbol{x} = a_1 x_1 + a_2 x_2 + \cdots + a_D x_D \sim \mathcal{N}(m, v)$$

  for all $\boldsymbol{a} \in \mathbb{R}^D$, where $\boldsymbol{a} \neq \boldsymbol{0}$

- The multivariate Gaussian density for a variable $\boldsymbol{x} \in \mathbb{R}^D$:

$$\mathcal{N}(\boldsymbol{x}|\boldsymbol{\mu}, \boldsymbol{\Sigma}) = (2\pi)^{-\frac{D}{2}} |\boldsymbol{\Sigma}|^{-\frac{1}{2}} \exp\left[-\frac{1}{2}(\boldsymbol{x} - \boldsymbol{\mu})^T \boldsymbol{\Sigma}^{-1} (\boldsymbol{x} - \boldsymbol{\mu})\right]$$

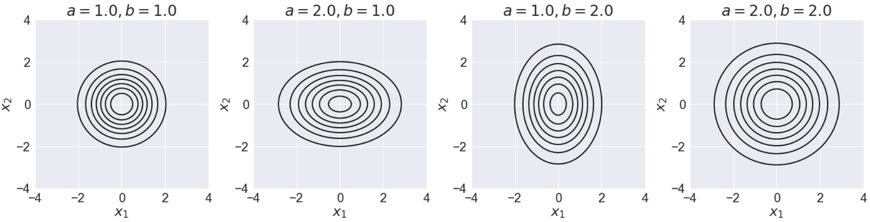
- Completely described by its parameters:
  - $\boldsymbol{\mu} \in \mathbb{R}^D$ is the mean vector
  - $\boldsymbol{\Sigma} \in \mathbb{R}^{D \times D}$ is the covariance matrix (positive definite)

- $(\boldsymbol{\Sigma})_{ij}$ is the covariance between the i'th and j'th elements in $\boldsymbol{x}$

Interpretation of the covariance matrix - 2D examples

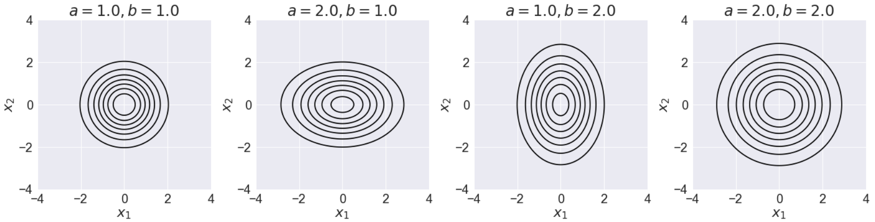
The diagonal of the covariance controls the scaling/marginal variances

$$\boldsymbol{\mu} = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \qquad \boldsymbol{\Sigma} = \begin{bmatrix} a & 0 \\ 0 & b \end{bmatrix} \tag{1}$$



The diagonal of the covariance controls the scaling/marginal variances

$$\boldsymbol{\mu} = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \qquad \boldsymbol{\Sigma} = \begin{bmatrix} a & 0 \\ 0 & b \end{bmatrix} \tag{1}$$



Questions to be discussed with your neighbor:

1. If $\boldsymbol{\Sigma}$ is diagonal, then x_1 and x_2 are uncorrelated? True or false?
2. If $\boldsymbol{\Sigma}$ is diagonal, then x_1 and x_2 are independent? True or false?
3. What is the volume (integral) of density?
4. Which of the four densities has the highest peak and why?

The density at the mode

- The density is given by

$$\mathcal{N}\left(\boldsymbol{x}\middle|\boldsymbol{\mu},\boldsymbol{\Sigma}\right) = (2\pi)^{-\frac{D}{2}}\left|\boldsymbol{\Sigma}\right|^{-\frac{1}{2}}\exp\left[-\frac{1}{2}\left(\boldsymbol{x}-\boldsymbol{\mu}\right)^T\boldsymbol{\Sigma}^{-1}\left(\boldsymbol{x}-\boldsymbol{\mu}\right)\right]$$

- The mode (highest density value) is achieve at $\boldsymbol{x} = \boldsymbol{\mu}$

$$\mathcal{N}\left(\boldsymbol{\mu}\middle|\boldsymbol{\mu},\boldsymbol{\Sigma}\right) = (2\pi)^{-\frac{D}{2}}\left|\boldsymbol{\Sigma}\right|^{-\frac{1}{2}}$$

- The determinant of the covariance is

$$\left|\boldsymbol{\Sigma}\right| = \left|\begin{bmatrix} a & \rho \\ \rho & b \end{bmatrix}\right| = ab - \rho^2 \tag{2}$$

- Therefore

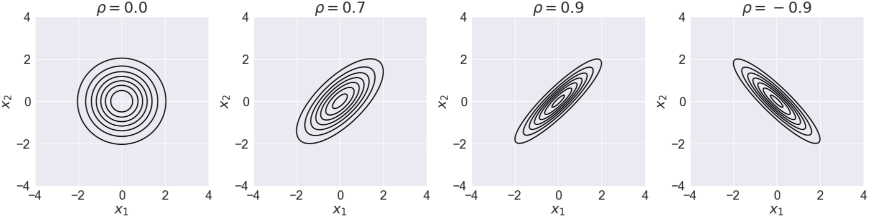
$$\mathcal{N}\left(\boldsymbol{\mu}\middle|\boldsymbol{\mu},\boldsymbol{\Sigma}\right) = (2\pi)^{-\frac{D}{2}}\left|\boldsymbol{\Sigma}\right|^{-\frac{1}{2}} = (2\pi)^{-\frac{D}{2}}\frac{1}{\sqrt{ab-\rho^2}}$$

Interpretation of the covariance matrix

The off-diagonals control the covariances:

$$(\mathbf{\Sigma})_{ij} = \text{cov}\left(x_i, x_j\right) = \mathbb{E}\left[x_i x_j\right] - \mu_i \mu_j \tag{3}$$

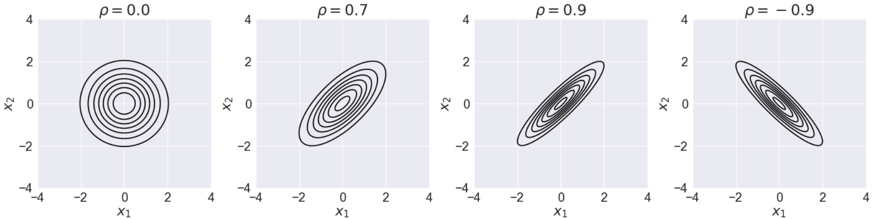
$$\boldsymbol{\mu} = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \qquad \mathbf{\Sigma} = \begin{bmatrix} 1 & \rho \\ \rho & 1 \end{bmatrix} \tag{4}$$



The off-diagonals control the covariances:

$$(\boldsymbol{\Sigma})_{ij} = \text{cov}\left(x_i, x_j\right) = \mathbb{E}\left[x_i x_j\right] - \mu_i \mu_j \tag{3}$$

$$\boldsymbol{\mu} = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \qquad \boldsymbol{\Sigma} = \begin{bmatrix} 1 & \rho \\ \rho & 1 \end{bmatrix} \tag{4}$$



Question:

- Which of the four densities has the highest peak and why?

Interpretation of the covariance matrix

Covariance matrices must be symmetric:

$$(\mathbf{\Sigma})_{ij} = \text{cov}\left(x_i, x_j\right) = \text{cov}\left(x_j, x_i\right) = (\mathbf{\Sigma})_{ji} \tag{5}$$

Consider the following set of covariance matrices:

$$\mathbf{\Sigma} = \begin{bmatrix} a & c \\ c & b \end{bmatrix} \tag{6}$$

c is the covariance between x_1 and x_2. Can c take any values?

Interpretation of the covariance matrix

Covariance matrices must be symmetric:

$$(\boldsymbol{\Sigma})_{ij} = \text{cov}\left(x_i, x_j\right) = \text{cov}\left(x_j, x_i\right) = (\boldsymbol{\Sigma})_{ji} \tag{5}$$

Consider the following set of covariance matrices:

$$\boldsymbol{\Sigma} = \begin{bmatrix} a & c \\ c & b \end{bmatrix} \tag{6}$$

c is the covariance between x_1 and x_2. Can c take any values?

$$\left|\rho\right| = \left|\frac{c}{\sqrt{a}\sqrt{b}}\right| \leq 1 \qquad \Rightarrow \qquad \left|c\right| \leq \sqrt{a}\sqrt{b} \tag{7}$$

Interpretation of the covariance matrix

Covariance matrices must be symmetric:

$$(\mathbf{\Sigma})_{ij} = \text{cov}\left(x_i, x_j\right) = \text{cov}\left(x_j, x_i\right) = (\mathbf{\Sigma})_{ji} \tag{5}$$

Consider the following set of covariance matrices:

$$\mathbf{\Sigma} = \begin{bmatrix} a & c \\ c & b \end{bmatrix} \tag{6}$$

c is the covariance between x_1 and x_2. Can c take any values?

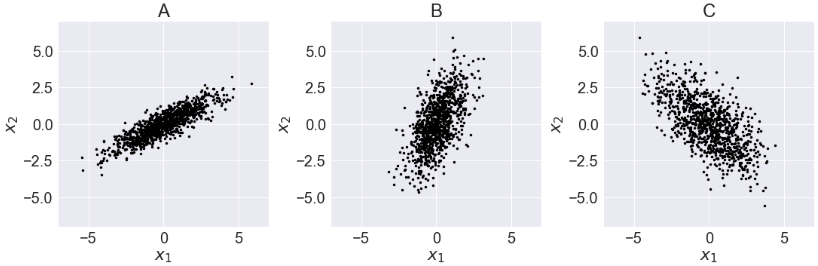
$$\left|\rho\right| = \left|\frac{c}{\sqrt{a}\sqrt{b}}\right| \leq 1 \qquad \Rightarrow \qquad \left|c\right| \leq \sqrt{a}\sqrt{b} \tag{7}$$

$\mathbf{\Sigma}$ must be positive definite

Determine which of the following 5 matrices are valid covariance matrices and match them to the set of samples below.

$$\mathbf{\Sigma}_1 = \begin{bmatrix} 3 & -2 \\ -2 & 3 \end{bmatrix} \qquad \mathbf{\Sigma}_2 = \begin{bmatrix} 3 & 2 \\ 1.5 & 3 \end{bmatrix} \qquad \mathbf{\Sigma}_3 = \begin{bmatrix} 1 & 1 \\ 1 & 3 \end{bmatrix}$$

$$\mathbf{\Sigma}_4 = \begin{bmatrix} 1 & -2 \\ -2 & 3 \end{bmatrix} \qquad \mathbf{\Sigma}_5 = \begin{bmatrix} 3 & 1.5 \\ 1.5 & 1 \end{bmatrix}$$



Discuss with your neighbor for 3 minutes

The multivariate Gaussian: Basic properties

- Gaussian distributions are closed under addition:

$$\boldsymbol{x}_1 \sim \mathcal{N}\left(\boldsymbol{m}_1, \boldsymbol{V}_1\right), \boldsymbol{x}_2 \sim \mathcal{N}\left(\boldsymbol{m}_2, \boldsymbol{V}_2\right) \quad \Rightarrow \quad \boldsymbol{x}_1 + \boldsymbol{x}_2 \sim \mathcal{N}\left(\boldsymbol{m}_1 + \boldsymbol{m}_2, \boldsymbol{V}_1 + \boldsymbol{V}_2\right)$$

The multivariate Gaussian: Basic properties

- Gaussian distributions are closed under addition:

$$\boldsymbol{x}_1 \sim \mathcal{N}\left(\boldsymbol{m}_1, \boldsymbol{V}_1\right), \boldsymbol{x}_2 \sim \mathcal{N}\left(\boldsymbol{m}_2, \boldsymbol{V}_2\right) \quad \Rightarrow \quad \boldsymbol{x}_1 + \boldsymbol{x}_2 \sim \mathcal{N}\left(\boldsymbol{m}_1 + \boldsymbol{m}_2, \boldsymbol{V}_1 + \boldsymbol{V}_2\right)$$

- For any finite number of independent variables:

$$\boldsymbol{x}_i \sim \mathcal{N}\left(\boldsymbol{m}_i, \boldsymbol{V}_i\right) \quad \Rightarrow \quad \sum_i \boldsymbol{x}_i \sim \mathcal{N}\left(\sum_i \boldsymbol{m}_i, \sum_i \boldsymbol{V}_i\right)$$

The multivariate Gaussian: Basic properties

- Gaussian distributions are closed under addition:

$$\boldsymbol{x}_1 \sim \mathcal{N}\left(\boldsymbol{m}_1, \boldsymbol{V}_1\right), \boldsymbol{x}_2 \sim \mathcal{N}\left(\boldsymbol{m}_2, \boldsymbol{V}_2\right) \quad \Rightarrow \quad \boldsymbol{x}_1 + \boldsymbol{x}_2 \sim \mathcal{N}\left(\boldsymbol{m}_1 + \boldsymbol{m}_2, \boldsymbol{V}_1 + \boldsymbol{V}_2\right)$$

- For any finite number of independent variables:

$$\boldsymbol{x}_i \sim \mathcal{N}\left(\boldsymbol{m}_i, \boldsymbol{V}_i\right) \quad \Rightarrow \quad \sum_i \boldsymbol{x}_i \sim \mathcal{N}\left(\sum_i \boldsymbol{m}_i, \sum_i \boldsymbol{V}_i\right)$$

- Gaussian distributions are closed under affine transformations:

$$\boldsymbol{x} \sim \mathcal{N}\left(\boldsymbol{m}, \boldsymbol{V}\right), \quad \Rightarrow \quad \boldsymbol{A}\boldsymbol{x} + \boldsymbol{b} \sim \mathcal{N}\left(\boldsymbol{A}\boldsymbol{m} + \boldsymbol{b}, \boldsymbol{A}\boldsymbol{V}\boldsymbol{A}^T\right)$$

The multivariate Gaussian: Basic properties

- Gaussian distributions are closed under addition:

$$\boldsymbol{x}_1 \sim \mathcal{N}(\boldsymbol{m}_1, \boldsymbol{V}_1), \boldsymbol{x}_2 \sim \mathcal{N}(\boldsymbol{m}_2, \boldsymbol{V}_2) \quad \Rightarrow \quad \boldsymbol{x}_1 + \boldsymbol{x}_2 \sim \mathcal{N}(\boldsymbol{m}_1 + \boldsymbol{m}_2, \boldsymbol{V}_1 + \boldsymbol{V}_2)$$

- For any finite number of independent variables:

$$\boldsymbol{x}_i \sim \mathcal{N}(\boldsymbol{m}_i, \boldsymbol{V}_i) \quad \Rightarrow \quad \sum_i \boldsymbol{x}_i \sim \mathcal{N}\left(\sum_i \boldsymbol{m}_i, \sum_i \boldsymbol{V}_i\right)$$

- Gaussian distributions are closed under affine transformations:

$$\boldsymbol{x} \sim \mathcal{N}(\boldsymbol{m}, \boldsymbol{V}), \quad \Rightarrow \quad \boldsymbol{A}\boldsymbol{x} + \boldsymbol{b} \sim \mathcal{N}\left(\boldsymbol{A}\boldsymbol{m} + \boldsymbol{b}, \boldsymbol{A}\boldsymbol{V}\boldsymbol{A}^T\right)$$

- Hence, manipulating Gaussian distributions often boils down to linear algebra

Discuss with your neighbor...

... how to use the following two results

$$\boldsymbol{x}_i \sim \mathcal{N}(\boldsymbol{m}_i, \boldsymbol{V}_i) \quad \Rightarrow \quad \sum_i \boldsymbol{x}_i \sim \mathcal{N}\left(\sum_i \boldsymbol{m}_i, \sum_i \boldsymbol{V}_i\right)$$

$$\boldsymbol{x} \sim \mathcal{N}(\boldsymbol{m}, \boldsymbol{V}) \quad \Rightarrow \quad \boldsymbol{A}\boldsymbol{x} + \boldsymbol{b} \sim \mathcal{N}\left(\boldsymbol{A}\boldsymbol{m} + \boldsymbol{b}, \boldsymbol{A}\boldsymbol{V}\boldsymbol{A}^T\right),$$

to calculate the distribution of $\boldsymbol{Y}$ in the following linear model?

$$\boldsymbol{Y} = \boldsymbol{\mu} + \boldsymbol{X}\boldsymbol{w} + \boldsymbol{\epsilon},$$

where

$$\boldsymbol{w} \sim \mathcal{N}(\boldsymbol{m}, \boldsymbol{V}) \qquad \boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \sigma^2 \boldsymbol{I})$$

Sampling from the multivariate Gaussian distribution

$$\boldsymbol{x} \sim \mathcal{N}\left(\boldsymbol{m}, \boldsymbol{V}\right) \quad \Rightarrow \quad \boldsymbol{A}\boldsymbol{x} + \boldsymbol{b} \sim \mathcal{N}\left(\boldsymbol{A}\boldsymbol{m} + \boldsymbol{b}, \boldsymbol{A}\boldsymbol{V}\boldsymbol{A}^T\right)$$

- Suppose we know how to generate samples from a standardized univariate Gaussian distribution
- How can we use the above result to generate samples from an arbitrary multivariate Gaussian distribution $\boldsymbol{y} \sim \mathcal{N}\left(\boldsymbol{m}, \boldsymbol{V}\right)$?

Sampling from the multivariate Gaussian distribution

$$\boldsymbol{x} \sim \mathcal{N}(\boldsymbol{m}, \boldsymbol{V}) \quad \Rightarrow \quad \boldsymbol{A}\boldsymbol{x} + \boldsymbol{b} \sim \mathcal{N}\left(\boldsymbol{A}\boldsymbol{m} + \boldsymbol{b}, \boldsymbol{A}\boldsymbol{V}\boldsymbol{A}^T\right)$$

- Suppose we know how to generate samples from a standardized univariate Gaussian distribution
- How can we use the above result to generate samples from an arbitrary multivariate Gaussian distribution $\boldsymbol{y} \sim \mathcal{N}(\boldsymbol{m}, \boldsymbol{V})$?
  1. Compute the matrix square root of $\boldsymbol{V} = \boldsymbol{L}\boldsymbol{L}^T$
  2. Generate a sample of $\boldsymbol{x}$ such that $x_i \sim \mathcal{N}(0, 1)$, i.e. $\boldsymbol{x} \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{I})$
  3. Compute $\boldsymbol{y} = \boldsymbol{L}\boldsymbol{x} + \boldsymbol{m}$

Sampling from the multivariate Gaussian distribution

$$\boldsymbol{x} \sim \mathcal{N}(\boldsymbol{m}, \boldsymbol{V}) \quad \Rightarrow \quad \boldsymbol{Ax} + \boldsymbol{b} \sim \mathcal{N}\left(\boldsymbol{Am} + \boldsymbol{b}, \boldsymbol{AVA}^T\right)$$

- Suppose we know how to generate samples from a standardized univariate Gaussian distribution
- How can we use the above result to generate samples from an arbitrary multivariate Gaussian distribution $\boldsymbol{y} \sim \mathcal{N}(\boldsymbol{m}, \boldsymbol{V})$?
  1. Compute the matrix square root of $\boldsymbol{V} = \boldsymbol{LL}^T$
  2. Generate a sample of $\boldsymbol{x}$ such that $x_i \sim \mathcal{N}(0, 1)$, i.e. $\boldsymbol{x} \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{I})$
  3. Compute $\boldsymbol{y} = \boldsymbol{Lx} + \boldsymbol{m}$
- Why does it work?

$$\boldsymbol{y} = \boldsymbol{Lx} + \boldsymbol{m} \sim \mathcal{N}\left(\boldsymbol{L0} + \boldsymbol{m}, \boldsymbol{LIL}^T\right) = \mathcal{N}(\boldsymbol{m}, \boldsymbol{V}) \tag{8}$$

The multivariate Gaussian: Marginalization

- Gaussian densities are closed on marginalization
- Let $\boldsymbol{x}_1$ and $\boldsymbol{x}_2$ be a partitioning of $\boldsymbol{x} = \boldsymbol{x}_1 \cup \boldsymbol{x}_2$, then

$$p(\boldsymbol{x}_1, \boldsymbol{x}_2) = \mathcal{N}\left(\begin{bmatrix}\boldsymbol{x}_1\\ \boldsymbol{x}_2\end{bmatrix} \,\middle|\, \begin{bmatrix}\boldsymbol{m}_1\\ \boldsymbol{m}_2\end{bmatrix}, \begin{bmatrix}\Sigma_{11} & \Sigma_{12}\\ \Sigma_{21} & \Sigma_{22}\end{bmatrix}\right) \tag{9}$$

The multivariate Gaussian: Marginalization

- Gaussian densities are closed on marginalization
- Let $\boldsymbol{x}_1$ and $\boldsymbol{x}_2$ be a partitioning of $\boldsymbol{x} = \boldsymbol{x}_1 \cup \boldsymbol{x}_2$, then

$$p(\boldsymbol{x}_1, \boldsymbol{x}_2) = \mathcal{N}\left(\begin{bmatrix}\boldsymbol{x}_1\\ \boldsymbol{x}_2\end{bmatrix} \middle| \begin{bmatrix}\boldsymbol{m}_1\\ \boldsymbol{m}_2\end{bmatrix}, \begin{bmatrix}\Sigma_{11} & \Sigma_{12}\\ \Sigma_{21} & \Sigma_{22}\end{bmatrix}\right) \tag{9}$$

then

$$p(\boldsymbol{x}_1) = \int p(\boldsymbol{x}_1, \boldsymbol{x}_2)\mathrm{d}\boldsymbol{x}_2 = \mathcal{N}\left(\boldsymbol{x}_1 | \boldsymbol{m}_1, \Sigma_{11}\right) \tag{10}$$

The multivariate Gaussian: Marginalization

- Gaussian densities are closed on marginalization
- Let $\boldsymbol{x}_1$ and $\boldsymbol{x}_2$ be a partitioning of $\boldsymbol{x} = \boldsymbol{x}_1 \cup \boldsymbol{x}_2$, then

$$p(\boldsymbol{x}_1, \boldsymbol{x}_2) = \mathcal{N}\left(\begin{bmatrix}\boldsymbol{x}_1 \\ \boldsymbol{x}_2\end{bmatrix} \middle| \begin{bmatrix}\boldsymbol{m}_1 \\ \boldsymbol{m}_2\end{bmatrix}, \begin{bmatrix}\boldsymbol{\Sigma}_{11} & \boldsymbol{\Sigma}_{12} \\ \boldsymbol{\Sigma}_{21} & \boldsymbol{\Sigma}_{22}\end{bmatrix}\right) \tag{9}$$

then

$$p(\boldsymbol{x}_1) = \int p(\boldsymbol{x}_1, \boldsymbol{x}_2) \mathrm{d}\boldsymbol{x}_2 = \mathcal{N}\left(\boldsymbol{x}_1 | \boldsymbol{m}_1, \boldsymbol{\Sigma}_{11}\right) \tag{10}$$

and

$$p(\boldsymbol{x}_2) = \int p(\boldsymbol{x}_1, \boldsymbol{x}_2) \mathrm{d}\boldsymbol{x}_1 = \mathcal{N}\left(\boldsymbol{x}_2 | \boldsymbol{m}_2, \boldsymbol{\Sigma}_{\mathbf{22}}\right) \tag{11}$$

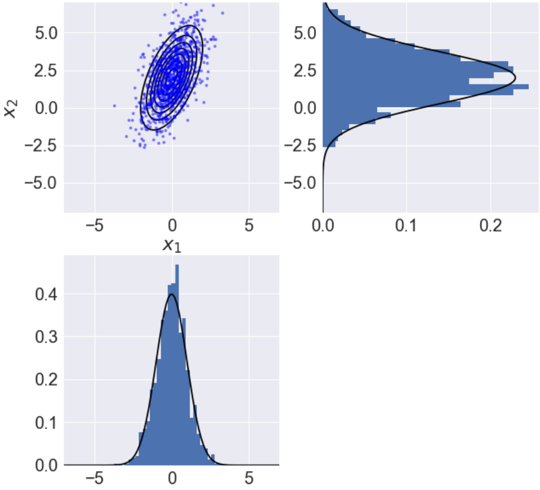
- The same is true for any partitioning

Marginalization example in 2D

$$\boldsymbol{x} \sim \mathcal{N}\left(\begin{bmatrix}0\\2\end{bmatrix}, \begin{bmatrix}1 & 1\\1 & 3\end{bmatrix}\right)$$

$$x_1 \sim \mathcal{N}(0, 1)$$

$$x_2 \sim \mathcal{N}(2, 3)$$



Conditioning

- Gaussian densities are closed under conditioning!
- Recall the definition of conditioning:

$$p(A|B) = \frac{p(A \cap B)}{p(B)}$$

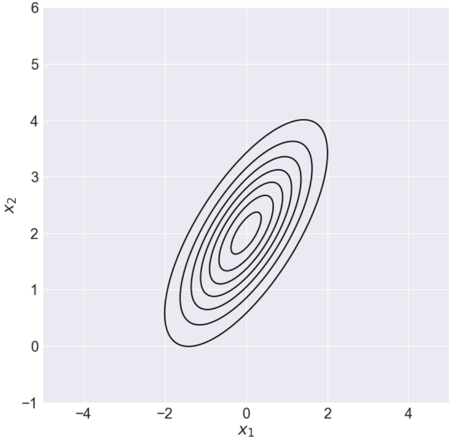
- Let $\boldsymbol{x}_1$ and $\boldsymbol{x}_2$ be a partitioning of $\boldsymbol{x} = \boldsymbol{x}_1 \cup \boldsymbol{x}_2$, then

$$p(\boldsymbol{x}_1, \boldsymbol{x}_2) = \mathcal{N}\left(\begin{bmatrix}\boldsymbol{x}_1 \\ \boldsymbol{x}_2\end{bmatrix} \Big| \begin{bmatrix}\boldsymbol{m}_1 \\ \boldsymbol{m}_2\end{bmatrix}, \begin{bmatrix}\boldsymbol{\Sigma}_{11} & \boldsymbol{\Sigma}_{12} \\ \boldsymbol{\Sigma}_{21} & \boldsymbol{\Sigma}_{22}\end{bmatrix}\right)$$

- The conditional of x_1 is given x_2 by:

$$p(\boldsymbol{x}_1|\boldsymbol{x}_2) = \mathcal{N}\left(\boldsymbol{x}_1|\boldsymbol{\Sigma}_{12}\boldsymbol{\Sigma}_{22}^{-1}\left[\boldsymbol{x}_2 - \boldsymbol{\mu}_2\right] + \boldsymbol{m}_1, \boldsymbol{\Sigma}_{11} - \boldsymbol{\Sigma}_{12}\boldsymbol{\Sigma}_{22}^{-1}\boldsymbol{\Sigma}_{21}\right)$$

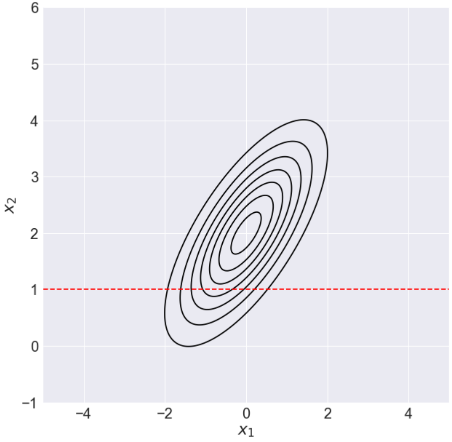
Conditioning example in 2D



- 2D example

$$\boldsymbol{\mu} = \begin{bmatrix} 0 \\ 2 \end{bmatrix} \quad \boldsymbol{\Sigma} = \begin{bmatrix} 1 & 0.8 \\ 0.8 & 1 \end{bmatrix}$$

Conditioning example in 2D

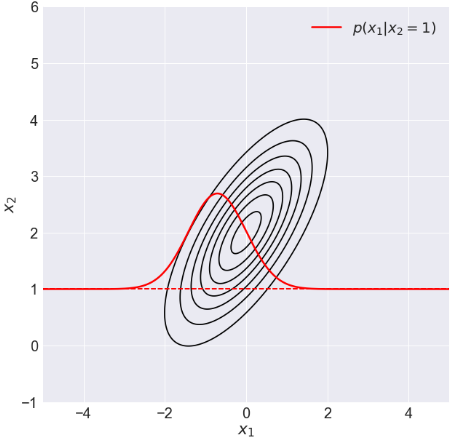


- 2D example

$$\boldsymbol{\mu} = \begin{bmatrix} 0 \\ 2 \end{bmatrix} \quad \boldsymbol{\Sigma} = \begin{bmatrix} 1 & 0.8 \\ 0.8 & 1 \end{bmatrix}$$

- Assume we observe $x_2 = 1$

Conditioning example in 2D



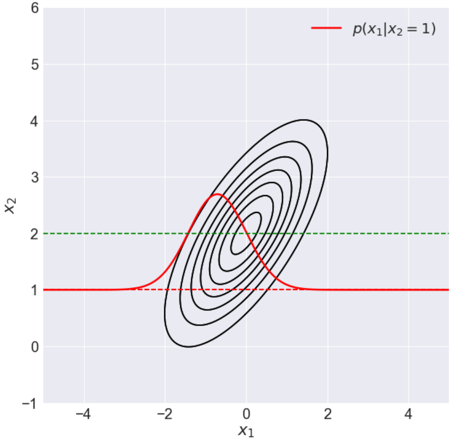
- 2D example

$$\boldsymbol{\mu} = \begin{bmatrix} 0 \\ 2 \end{bmatrix} \quad \boldsymbol{\Sigma} = \begin{bmatrix} 1 & 0.8 \\ 0.8 & 1 \end{bmatrix}$$

- Assume we observe $x_2 = 1$
- The conditional disitribution

$$p(x_1|x_2) = \mathcal{N}\left(x_1 | -\frac{\sqrt{2}}{2}, \frac{1}{2}\right)$$

Conditioning example in 2D

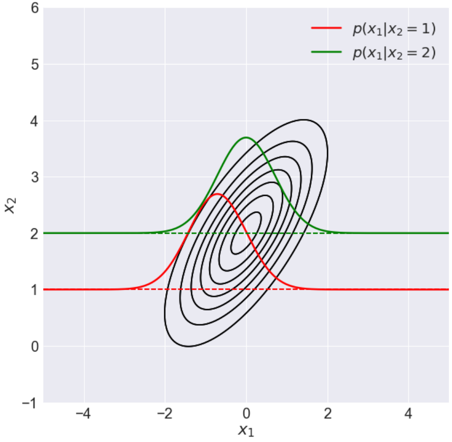


- 2D example

$$\boldsymbol{\mu} = \begin{bmatrix} 0 \\ 2 \end{bmatrix} \quad \boldsymbol{\Sigma} = \begin{bmatrix} 1 & 0.8 \\ 0.8 & 1 \end{bmatrix}$$

- Assume we observe $x_2 = 2$

Conditioning example in 2D



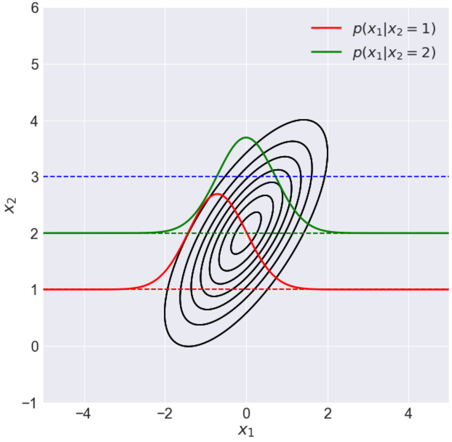
- 2D example

$$\boldsymbol{\mu} = \begin{bmatrix} 0 \\ 2 \end{bmatrix} \quad \boldsymbol{\Sigma} = \begin{bmatrix} 1 & 0.8 \\ 0.8 & 1 \end{bmatrix}$$

- Assume we observe $x_2 = 2$
- The conditional disitribution

$$p(x_1|x_2) = \mathcal{N}\left(x_1|0, \frac{1}{2}\right)$$

Conditioning example in 2D

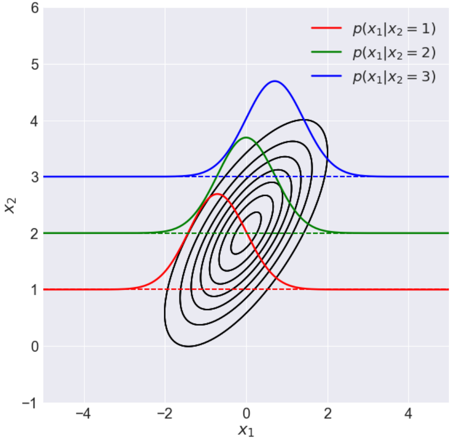


- 2D example

$$\boldsymbol{\mu} = \begin{bmatrix} 0 \\ 2 \end{bmatrix} \quad \boldsymbol{\Sigma} = \begin{bmatrix} 1 & 0.8 \\ 0.8 & 1 \end{bmatrix}$$

- Assume we observe $x_2 = 3$

Conditioning example in 2D



- 2D example

$$\boldsymbol{\mu} = \begin{bmatrix} 0 \\ 2 \end{bmatrix} \quad \boldsymbol{\Sigma} = \begin{bmatrix} 1 & 0.8 \\ 0.8 & 1 \end{bmatrix}$$

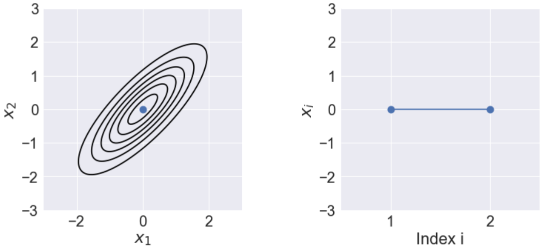
- Assume we observe $x_2 = 3$
- The conditional disitribution

$$p(x_1|x_2) = \mathcal{N}\left(x_1 \middle| \frac{\sqrt{2}}{2}, \frac{1}{2}\right)$$

Visualizing samples in higher dimensions

- Visualizations in 2D

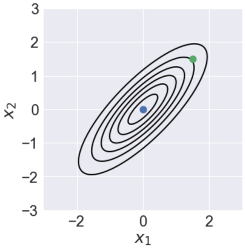
$$\mathbf{\Sigma} = \begin{bmatrix} 1 & 0.8 \\ 0.8 & 1 \end{bmatrix}$$

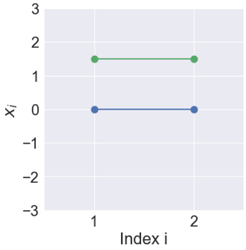


Visualizing samples in higher dimensions

- Visualizations in 2D

$$\Sigma = \begin{bmatrix} 1 & 0.8 \\ 0.8 & 1 \end{bmatrix}$$

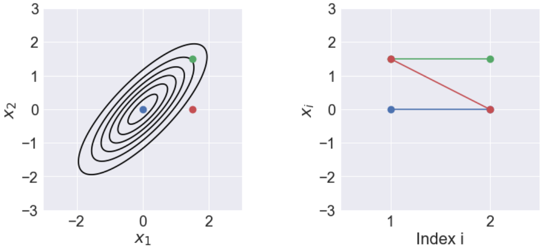




Visualizing samples in higher dimensions

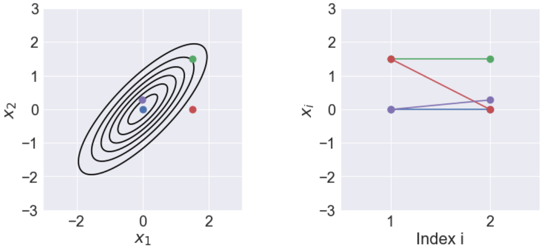
- Visualizations in 2D

$$\mathbf{\Sigma} = \begin{bmatrix} 1 & 0.8 \\ 0.8 & 1 \end{bmatrix}$$



- Visualizations in 2D

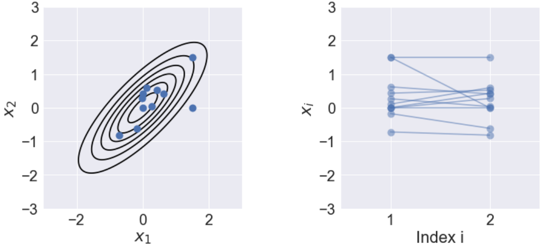
$$\mathbf{\Sigma} = \begin{bmatrix} 1 & 0.8 \\ 0.8 & 1 \end{bmatrix}$$



Visualizing samples in higher dimensions

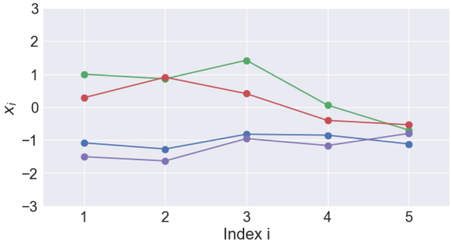
- Visualizations in 2D

$$\boldsymbol{\Sigma} = \begin{bmatrix} 1 & 0.8 \\ 0.8 & 1 \end{bmatrix}$$



Visualizing samples in higher dimensions

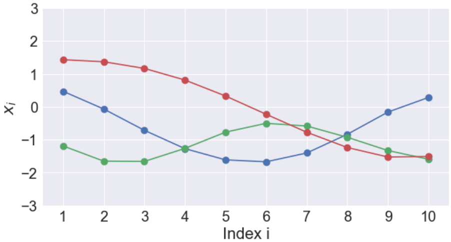
- Visualizations in 5D



$$\boldsymbol{\Sigma} = \begin{bmatrix} 1 & 0.8^1 & 0.8^2 & 0.8^3 & 0.8^4 \\ 0.8^1 & 1 & 0.8^1 & 0.8^2 & 0.8^3 \\ 0.8^2 & 0.8^1 & 1 & 0.8^1 & 0.8^2 \\ 0.8^3 & 0.8^2 & 0.8^1 & 1 & 0.8^1 \\ 0.8^4 & 0.8^3 & 0.8^2 & 0.8^1 & 1 \end{bmatrix}$$

Visualizing samples in higher dimensions

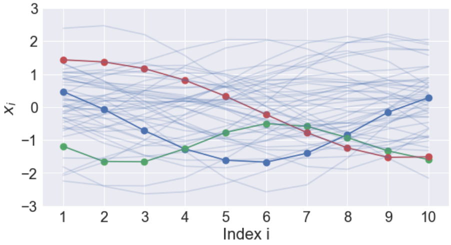
- Visualizations in 10D



$$\mathbf{\Sigma} = \begin{bmatrix} 1 & 0.8^1 & 0.8^2 & \dots & 0.8^9 \\ 0.8^1 & 1 & 0.8^1 & & \vdots \\ 0.8^2 & 0.8^1 & 1 & & \vdots \\ \vdots & & & \ddots & \vdots \\ 0.8^9 & \dots & \dots & \dots & 1 \end{bmatrix}$$

Visualizing samples in higher dimensions

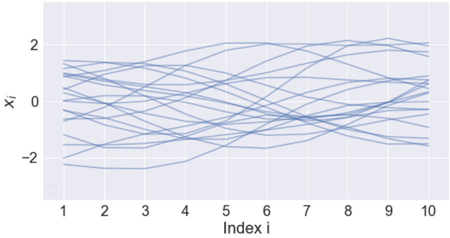
- Visualizations in 10D



$$\mathbf{\Sigma} = \begin{bmatrix} 1 & 0.8^1 & 0.8^2 & \dots & 0.8^9 \\ 0.8^1 & 1 & 0.8^1 & & \vdots \\ 0.8^2 & 0.8^1 & 1 & & \vdots \\ \vdots & & & \ddots & \vdots \\ 0.8^9 & \dots & \dots & \dots & 1 \end{bmatrix}$$

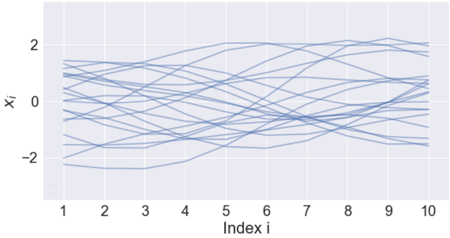
Back to conditioning

- So far, we have seen samples from the distribution $p(\boldsymbol{x}) = \mathcal{N}\left(\boldsymbol{x} \middle| \mathbf{0}, \boldsymbol{\Sigma}\right)$



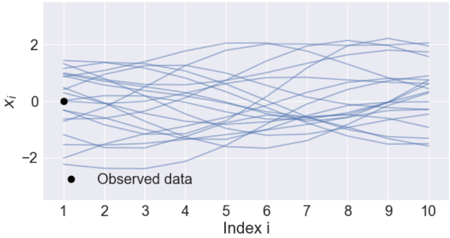
Back to conditioning

- So far, we have seen samples from the distribution $p(\boldsymbol{x}) = \mathcal{N}\left(\boldsymbol{x}\middle|\mathbf{0}, \boldsymbol{\Sigma}\right)$
- We can also write $p(\boldsymbol{x}) = p(x_1, \boldsymbol{x}_{2:10})$



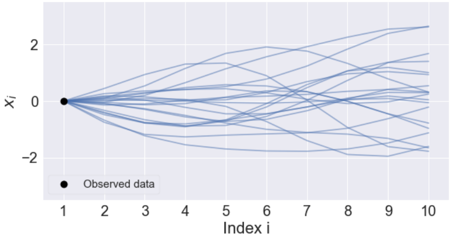
Back to conditioning

- So far, we have seen samples from the distribution $p\left(\boldsymbol{x}\right) = \mathcal{N}\left(\boldsymbol{x} \middle| \mathbf{0}, \boldsymbol{\Sigma}\right)$
- We can also write $p\left(\boldsymbol{x}\right) = p(x_1, \boldsymbol{x}_{2:10})$
- We now observe $x_1 = 0$



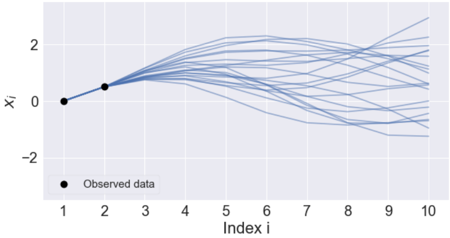
Back to conditioning

- So far, we have seen samples from the distribution $p(\boldsymbol{x}) = \mathcal{N}\left(\boldsymbol{x} \middle| \mathbf{0}, \boldsymbol{\Sigma}\right)$
- We can also write $p(\boldsymbol{x}) = p(x_1, \boldsymbol{x}_{2:10})$
- We now observe $x_1 = 0$
- Let's sample from the conditional distribution $p(\boldsymbol{x}_{2:10} \middle| x_1 = 0)$



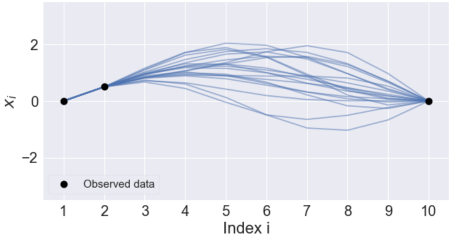
Back to conditioning

- So far, we have seen samples from the distribution $p\left(\boldsymbol{x}\right) = \mathcal{N}\left(\boldsymbol{x}\middle|\mathbf{0}, \boldsymbol{\Sigma}\right)$
- We can also write $p\left(\boldsymbol{x}\right) = p(x_1, \boldsymbol{x}_{2:10})$
- We now observe $x_1 = 0$
- Let's sample from the conditional distribution $p(\boldsymbol{x}_{2:10}\middle|x_1 = 0)$



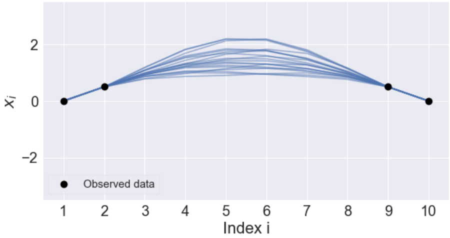
Back to conditioning

- So far, we have seen samples from the distribution $p(\boldsymbol{x}) = \mathcal{N}\left(\boldsymbol{x} \middle| \mathbf{0}, \boldsymbol{\Sigma}\right)$
- We can also write $p(\boldsymbol{x}) = p(x_1, \boldsymbol{x}_{2:10})$
- We now observe $x_1 = 0$
- Let's sample from the conditional distribution $p(\boldsymbol{x}_{2:10} \middle| x_1 = 0)$



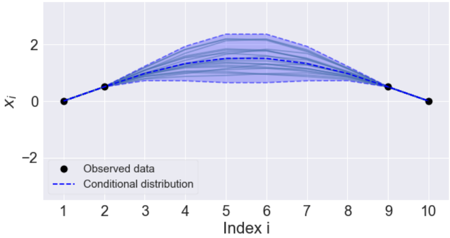
Back to conditioning

- So far, we have seen samples from the distribution $p(\boldsymbol{x}) = \mathcal{N}\left(\boldsymbol{x} \middle| \mathbf{0}, \boldsymbol{\Sigma}\right)$
- We can also write $p(\boldsymbol{x}) = p(x_1, \boldsymbol{x}_{2:10})$
- We now observe $x_1 = 0$
- Let's sample from the conditional distribution $p(\boldsymbol{x}_{2:10} \middle| x_1 = 0)$



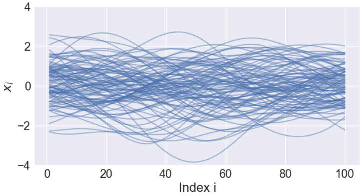
Back to conditioning

- So far, we have seen samples from the distribution $p(\boldsymbol{x}) = \mathcal{N}\left(\boldsymbol{x} \middle| \mathbf{0}, \boldsymbol{\Sigma}\right)$
- We can also write $p(\boldsymbol{x}) = p(x_1, \boldsymbol{x}_{2:10})$
- We now observe $x_1 = 0$
- Let's sample from the conditional distribution $p(\boldsymbol{x}_{2:10} \middle| x_1 = 0)$



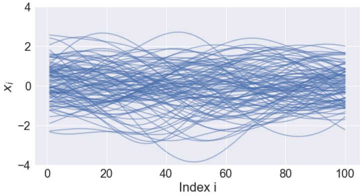
Back to conditioning II

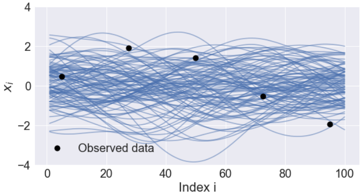
- Let's now consider a case with $\boldsymbol{x} \in \mathbb{R}^{100}$ dimensions with 5 observations



Back to conditioning II

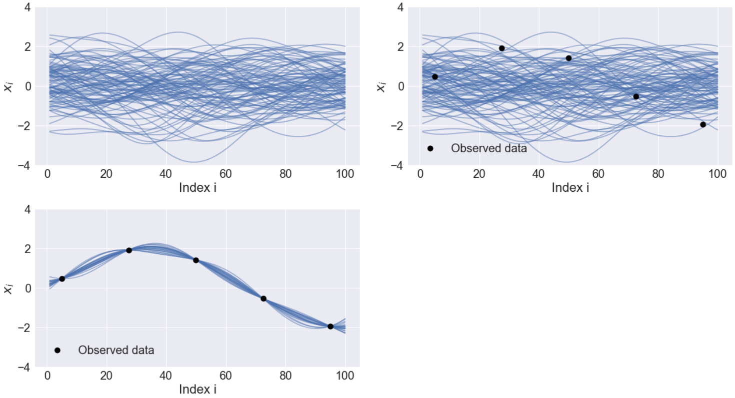
- Let's now consider a case with $\boldsymbol{x} \in \mathbb{R}^{100}$ dimensions with 5 observations





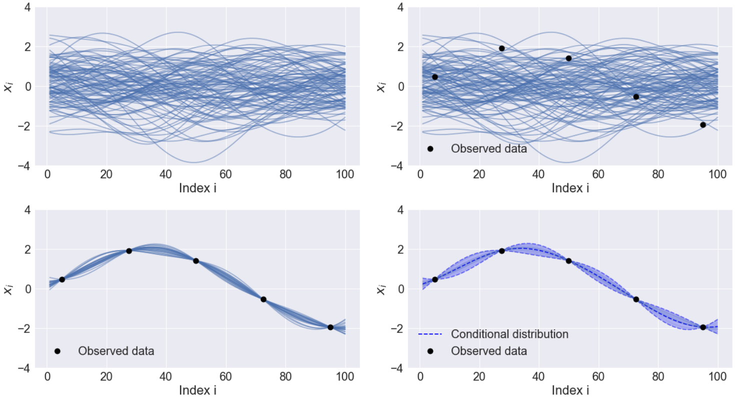
Back to conditioning II

- Let's now consider a case with $\boldsymbol{x} \in \mathbb{R}^{100}$ dimensions with 5 observations



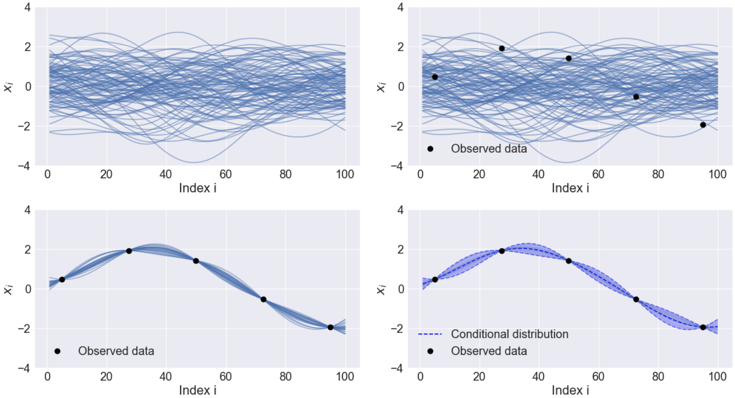
Back to conditioning II

- Let's now consider a case with $\boldsymbol{x} \in \mathbb{R}^{100}$ dimensions with 5 observations



Back to conditioning II

- Let's now consider a case with $\boldsymbol{x} \in \mathbb{R}^{100}$ dimensions with 5 observations



- Informally: We can think functions as vectors with infinite dimensions
- Using conditining in Gaussian distributions, we can do non-linear regression!

The end of todays lecture

- Next time
  - We will introduce Gaussian processes more formally
  - Read Chapter 1 & 2 in Gaussian processes for Machine Learning by Carl Rasmussen (http://www.gaussianprocess.org/gpml)
- First assignment
  - Warm up for Gaussian processes
  - Reviews the basics of Bayesian inference
  - Reviews the multivariate Gaussian density
  - Must be handed in through MyCourses