Mathematics Preliminaries

Introduction

University of Twente

Why these preliminaries?

- These preliminaries do not intend to teach you mathematics! They are here to refresh your knowledge, and help you discover your background shortcomings
- We try to provide you with the intuitions behind the algorithms and equations, but first we need to understand each other
- You can probably pass the course without explicit knowledge of these mathematics. However, if you want to understand the course, being comfortable with the terms presented here is necessary.

Linear Algebra
- Vectors
- Matrices

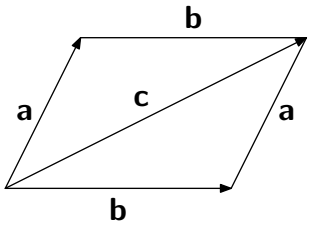
Probability Theory
- Basic Terms
- Rules of Probability
- Probability Densities
- Bayes Rule

Function optimisation

Linear Algebra
- Vectors
- Matrices

Probability Theory
- Basic Terms
- Rules of Probability
- Probability Densities
- Bayes Rule

Function optimisation

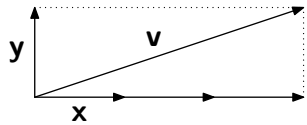


Example

On the left:
$\mathbf{a} + \mathbf{b} = \mathbf{c}$
Thus, if:
$\mathbf{a} + \mathbf{a} = 2\mathbf{a} = \mathbf{d}$
Then $\mathbf{d}$ is a vector of the same direction with $\mathbf{a}$ and double the length.

We will follow the convention of **bold** lower case letters for vectors and bold upper case letters for matrices in this course.

Vectors



- An n-dimensional space has n vectors as basis. These vectors are called *basis vectors* and they must be *linearly independent*
- We can then express any vector in this space using the basis vectors.

Example

Here, $\mathbf{v} = \mathbf{x} + \mathbf{x} + \mathbf{x} + \mathbf{y} = 3\mathbf{x} + \mathbf{y}$.
This is expressed as a vector, most commonly a column vector in pattern recognition,

$$\mathbf{v} = \begin{bmatrix} 3 \\ 1 \end{bmatrix}$$

Some vector properties

- The ith element of a vector $\mathbf{x}$ is denoted as x_i
- The length or 2-norm of a vector is denoted $|\mathbf{x}|$:

$$\begin{aligned} |\mathbf{x}| &= \sqrt{\sum_{i=1}^{n} x_i^2} \\ &= \sqrt{x_1^2 + x_2^2 + \cdots + x_n^2} \end{aligned}$$

- More generally, the p-norm of a vector is given by

$$|\mathbf{x}|_p = \sqrt[p]{\sum_{i=1}^{n} x_i^p}$$

Why do we need vectors for MLPR

- The "points" that represent objects are the head (tip, endpoint) of a feature vector
- Each dimension represents an extracted feature
- Machine learning algorithms apply operations on vectors.

Inner Product

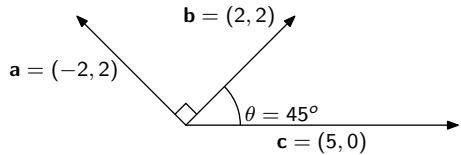
- We defined the multiplication of a scalar with a vector (recall $2\mathbf{a}$).
- The inner product (scalar product) of two n-dimensional vectors is defined as

$$\mathbf{a} \cdot \mathbf{b} = \sum_{i=1}^{n} a_i \, b_i \tag{1}$$

  which is equivalent with $\mathbf{a} \cdot \mathbf{b} = |\mathbf{a}||\mathbf{b}| \cos\theta$
- Two (non-zero) vectors are *linearly independent* if and only if their inner product is zero (thus $\cos\theta = 0, \theta = 90^o$)
- The inner product can be used to *project a vector* on another.

Inner product

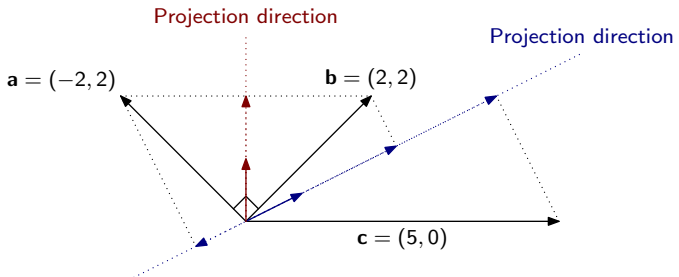
Example



In the vectors shown here:

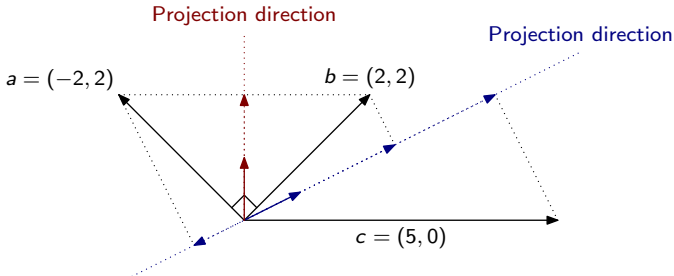
$$\begin{aligned}\mathbf{a} \cdot \mathbf{b} &= |\mathbf{a}||\mathbf{b}| \cos\theta = \sqrt{8}\sqrt{8}\cos 90^o = 0, \text{ or} \\ &= 2 \cdot 2 + 2 \cdot (-2) = 4 - 4 = 0 \\ \mathbf{b} \cdot \mathbf{c} &= |\mathbf{b}||\mathbf{c}| \cos\theta = \sqrt{8}\sqrt{25}\cos 45^o = 2\sqrt{2} \cdot 5 \cdot \sqrt{2}/2 = 10, \text{ or} \\ &= 2 \cdot 5 + 2 \cdot 0 = 10\end{aligned}$$

Projection Operation



In this figure we can see a graphical representation of the projection operation.

Projection Operation



- Notice that the result of the projection of a vector on a vector is a scalar
- The length of the projection is proportional to the inner product of the projected vector (scaled by the length of the projection vector).

Matrices

The matrix is a rectangular array of numbers, denoted by bold, uppercase letters

- The element or entry of row i and column j of a matrix $\mathbf{A}$ is denoted as a_{ij}
- The size or order is specified as $m \times n$ where m is the number of rows and n the number of columns
- Addition is defined for matrices of the same size. Let $\mathbf{A}$ and $\mathbf{B}$ be of the same size. If $\mathbf{C} = \mathbf{A} + \mathbf{B}$, then $c_{ij} = a_{ij} + b_{ij}$
- Example:

$$\mathbf{A} = \begin{bmatrix} 1 & 2 & 3 \\ 6 & 5 & 2 \end{bmatrix}, \qquad \mathbf{B} = \begin{bmatrix} 4 & 2 & 1 \\ 4 & -1 & 7 \end{bmatrix}$$

$$\mathbf{A} + \mathbf{B} = \begin{bmatrix} 5 & 4 & 4 \\ 10 & 4 & 9 \end{bmatrix}$$

Matrix multiplication

- $\mathbf{A} \cdot \mathbf{B}$ is defined if and only if the matrices are of order $m \times n$ and $n \times o$

$$\mathbf{A} = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix}, \mathbf{B} = \begin{bmatrix} b_{11} & b_{12} \\ b_{21} & b_{22} \end{bmatrix}$$

$$\mathbf{A} \cdot \mathbf{B} = \begin{bmatrix} a_{11}b_{11} + a_{12}b_{21} & a_{11}b_{12} + a_{12}b_{22} \\ a_{21}b_{11} + a_{22}b_{21} & a_{21}b_{12} + a_{22}b_{22} \end{bmatrix}$$

- Vectors can be regarded single column matrices

Some special matrices

Matrices with a special function are:

- The zero matrix. All elements are zero, so $\mathbf{A} + \mathbf{Z} = \mathbf{A}$
- The identity matrix, $\mathbf{I} = \begin{bmatrix} 1 & 0 & \cdots & 0 \\ 0 & 1 & & 0 \\ \vdots & & \ddots & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$, so that $\mathbf{A} \cdot \mathbf{I} = \mathbf{I} \cdot \mathbf{A} = \mathbf{A}$
- The inverse matrix of $\mathbf{A}$ is denoted $\mathbf{A}^{-1}$, and

$$\mathbf{A}^{-1} \cdot \mathbf{A} = \mathbf{A} \cdot \mathbf{A}^{-1} = \mathbf{I}$$

- The transpose matrix of $\mathbf{A}$ is denoted as $\mathbf{A}^{\top}$, where $a_{ij}^{\top} = a_{ji}$

Why do we need matrices in MLPR?

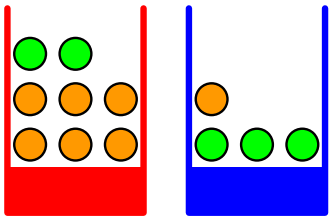
- Matrices can represent a system of equations
- Many concatenated data points (feature vectors) become a matrix
- Matrices can incorporate complex projections and translations of data points

Eigenvectors & Eigenvalues

- We call $\mathbf{e}$ an eigenvector and λ the corresponding eigenvalue of matrix $\mathbf{A}$ if:

$$\mathbf{Ae} = \lambda \mathbf{e}$$

- These will be important when we look at dimensionality reduction



Example

- Probability - Uncertainty
- What is the probability that I will get a green ball?

- Formally, a Random Variable is a function from a sample space to the measurable space of possible values of the variable

Properties of probabilities

The axioms of probabilities are

- if $\models \phi$, then $p(\phi) = 1$
- if $\neg(\phi \wedge \psi)$, then $p(\phi \vee \psi) = p(\phi) + p(\psi)$

Some consequences of these axioms:

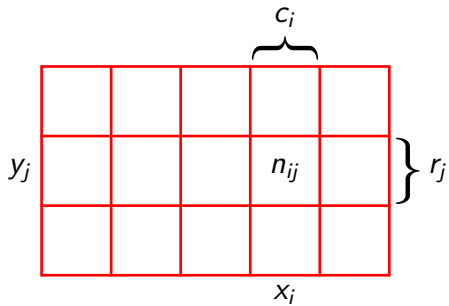
- $0 \leqslant p(X) \leqslant 1$
- If our events are mutually exclusive and include all possible outcomes, they sum up to 1
- In our case:

$$p(B = r) = 0.4 \text{ and } p(B = b) = 0.6$$

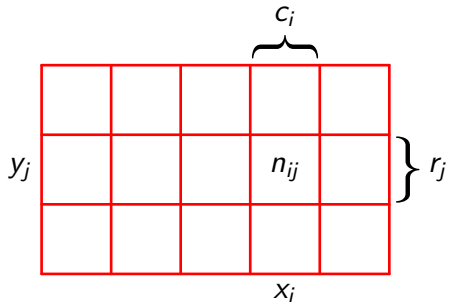
$$\sum_i p(B = i) = 1$$

Consider a more general example. We have two variables, X and Y, with values x_i for $i = 1 \ldots n$ and y_j for $j = 1 \ldots m$ respectively.

- We can create a lookup table, where we count how often each combination of values occurs:



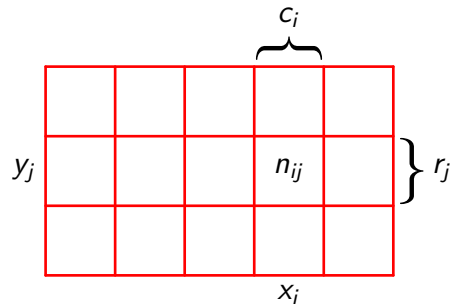
The Sum Rule



$$p(X = x_i, Y = y_j) = \frac{n_{ij}}{N} \ ,$$

$$p(X = x_i) = \frac{c_i}{N} = \sum_j p(X = x_i, Y = y_j)$$

The Product Rule

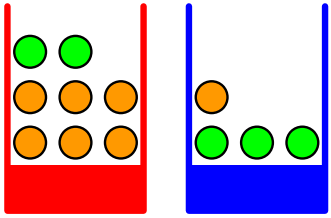


$$p(X = x_i, Y = y_j) = \frac{n_{ij}}{N} = \frac{n_{ij}}{c_i} \cdot \frac{c_i}{N}$$

$$\frac{n_{ij}}{c_i} = p(Y = y_j | X = x_i), \frac{c_i}{N} = p(X = x_i) \ ,$$

$$p(X = x_i, Y = y_j) = p(Y = y_j | X = x_i) \cdot p(X = x_i)$$

Using the rules

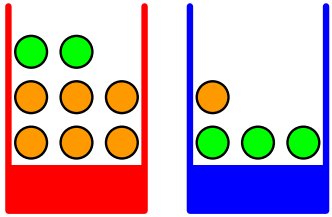


- We now want to estimate $p(F = g)$
- Given: $p(B = r) = 0.4$ and $p(B = b) = 0.6$

Example

$$
\begin{aligned}
p(F = g) &= \sum_B p(F = g, B) \\
&= p(F = g, B = b) + p(F = g, B = r) \\
&= p(F = g|B = r)p(B = r) + p(F = g|B = b)p(B = b) \\
&= 0.25 \cdot 0.4 + 0.75 \cdot 0.6 = 0.55
\end{aligned}
$$

Using the rules



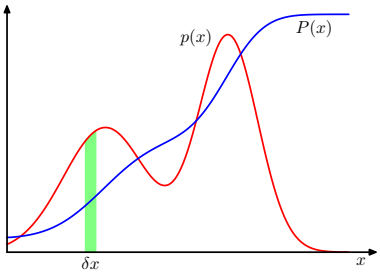
- We now want to estimate $p(F = g)$
- Given: $p(B = r) = 0.4$ and $p(B = b) = 0.6$

Example

$$
\begin{aligned}
p(F = g) &= \sum_B p(F = g, B) \\
&= p(F = g, B = b) + p(F = g, B = r) \\
&= p(F = g|B = r)p(B = r) + p(F = g|B = b)p(B = b) \\
&= 0.25 \cdot 0.4 + 0.75 \cdot 0.6 = 0.55
\end{aligned}
$$

Probability Density

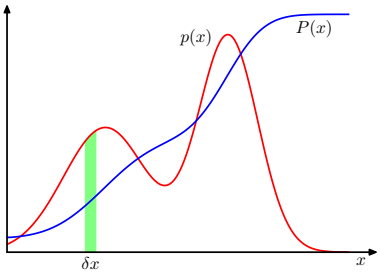
We have discussed the probability of a random variable to take a specific value, for instance p(B=r)



If the probability of a real valued variable x falling in the interval $(x, x + \delta x)$ is given by $p(x)\delta x$ for $\delta x \to 0$, then $p(x)$ is called the *probability density* over x.

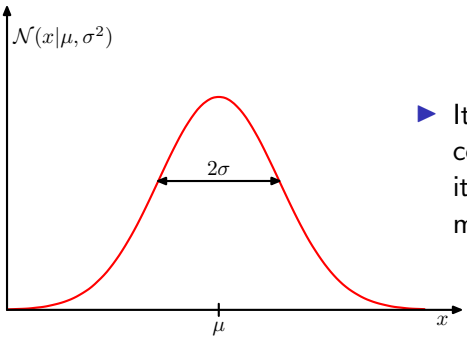
Probability Density

We have discussed the probability of a random variable to take a specific value, for instance p(B=r)



If the probability of a real valued variable x falling in the interval $(x, x + \delta x)$ is given by $p(x)\delta x$ for $\delta x \to 0$, then $p(x)$ is called the *probability density* over x.

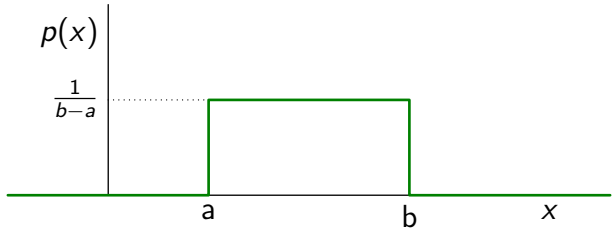
The Gaussian PDF



- It is one of the most commonly used pdf, since it is very easy to manipulate

- It is explicitly defined by two parameters. The mean μ and the variance σ
- It can also be defined in multiple dimensions. Then $\boldsymbol{\mu}$ is an n-dimensional vector and $\boldsymbol{\Sigma}$ a $n \times n$ dimensional matrix.
- $\boldsymbol{\Sigma}$ is now called the covariance matrix

The Uniform PDF



- It is a very useful function
- It represents the information that we have no information about the random variable
- It is explicitly defined by the limits of the random variable.

Expectation: Computing weigthed averages

Getting Pelted with Pebbles

Imagine you toss a fair coin, and for every head I give you a pebble, for every tail I give you two. After 10 throws, how many pebbles do you expect to have?

Expectation: Computing weigthed averages

Getting Pelted with Pebbles

Imagine you toss a fair coin, and for every head I give you a pebble, for every tail I give you two. After 10 throws, how many pebbles do you expect to have?

Intuitive answer

- We have a fair coin, so we expect pretty much equal numbers of heads and tails
- So, in this case, we expect 5 heads + 5 tails
- That is $5 \times 1 + 5 \times 2$ pebbles, or 15 pebbles
- As the number of tosses grows, we'll get closer to our expected outcome
- The tosses do not affect each other, so we can consider individual tosses
- We "expect" 1.5 pebbles per toss

Expectation: Computing weigthed averages

Getting Pelted with Pebbles

Imagine you toss a fair coin, and for every head I give you a pebble, for every tail I give you two. After 10 throws, how many pebbles do you expect to have?

Expectation

- We have a function, $f(C)$, which encodes how many pebbles we get for every toss:

$$f(C = h) = 1, \quad f(C = t) = 2$$

- Our particular coin is a fair coin:

$$p(C = h) = \tfrac{1}{2}, \quad p(C = t) = \tfrac{1}{2}$$

- By definition, the expectation is

$$\mathbb{E}[f(C)] = \sum_{C \in \{h,t\}} p(C)\, h(C) = \tfrac{1}{2}\, 1 + \tfrac{1}{2}\, 2 = 1.5$$

Expectation

We can use probability densities to estimate weighted averages of functions.

- The average value of function $f(x)$ under the distribution $p(x)$ is called the expectation of $f(x)$, it is denoted by $\mathbb{E}[f(x)]$ and is given by

$$\mathbb{E}[f(x)] = \int p(x)\, f(x)\, \mathrm{d}x$$

Data mean

When the function $f(x) = x$, the expectation $\mathbb{E}[x]$ is called the mean of the data

Variance

The variance of a x is defined as the expectation of $f(x) = (x - \mathbb{E}[x])^2$:

$$\text{var}[x] = \mathbb{E}[(x - \mathbb{E}[x])^2]$$

and it is an indication of how much variability there is in x around its mean value

- Notice that this definition is equivalent with

$$\text{var}[x] = \mathbb{E}[x^2] - \mathbb{E}[x]^2$$

$$p(X, Y) = p(X|Y) \cdot p(Y) = p(Y|X) \cdot p(X)$$

Thus,

$$p(X|Y) = \frac{p(Y|X) \cdot p(X)}{p(Y)}$$

Specifically, if X are our parameters (denoted w) and Y are our data ($\mathcal{D}$), this becomes:

$$p(w|\mathcal{D}) = \frac{p(\mathcal{D}|w) \cdot p(w)}{p(\mathcal{D})}$$

We call $p(w)$ the prior (probability distribution) of the parameters, $p(\mathcal{D}|w)$ the likelihood of the data and $p(w|\mathcal{D})$ the posterior (probability distribution) on the parameters.

$$p(w|\mathcal{D}) = \frac{p(\mathcal{D}|w) \cdot p(w)}{p(\mathcal{D})}$$

- What does this really mean?
- The denominator can be expressed using the sum rule as: $p(\mathcal{D}) = \sum_w p(\mathcal{D}, w) = \sum_w p(\mathcal{D}|w) \cdot p(w)$
- We are now allowed to incorporate prior knowledge in a prinicipled way
- Instead of considering different datasets (frequentinst), we consider different parameters settings (Bayesian)

Linear Algebra
- Vectors
- Matrices

Probability Theory
- Basic Terms
- Rules of Probability
- Probability Densities
- Bayes Rule

Function optimisation

Why function optimisation?

$$
\begin{aligned}
f(\text{  }) &= f(\text{  }) = f(\text{  }) = \cdots = \mathcal{C}_0 \\
f(\text{  }) &= f(\text{  }) = f(\text{  }) = \cdots = \mathcal{C}_1 \\
f(\text{  }) &= f(\text{  }) = f(\text{  }) = \cdots = \mathcal{C}_2 \\
f(\text{  }) &= f(\text{  }) = f(\text{  }) = \cdots = \mathcal{C}_3 \\
f(\text{  }) &= f(\text{  }) = f(\text{  }) = \cdots = \mathcal{C}_4 \\
f(\text{  }) &= f(\text{  }) = f(\text{  }) = \cdots = \mathcal{C}_5 \\
f(\text{  }) &= f(\text{  }) = f(\text{  }) = \cdots = \mathcal{C}_6 \\
f(\text{  }) &= f(\text{  }) = f(\text{  }) = \cdots = \mathcal{C}_7 \\
f(\text{  }) &= f(\text{  }) = f(\text{  }) = \cdots = \mathcal{C}_8 \\
f(\text{  }) &= f(\text{  }) = f(\text{  }) = \cdots = \mathcal{C}_9
\end{aligned}
$$

Why function optimisation?

$$
\begin{aligned}
f(\,\mathbf{w}) = f(\,\mathbf{w}) = f(\,\mathbf{w}) &= \cdots = \mathcal{C}_0 \\
f(\,\mathbf{w}) = f(\,\mathbf{w}) = f(\,\mathbf{w}) &= \cdots = \mathcal{C}_1 \\
f(\,\mathbf{w}) = f(\,\mathbf{w}) = f(\,\mathbf{w}) &= \cdots = \mathcal{C}_2 \\
f(\,\mathbf{w}) = f(\,\mathbf{w}) = f(\,\mathbf{w}) &= \cdots = \mathcal{C}_3 \\
f(\,\mathbf{w}) = f(\,\mathbf{w}) = f(\,\mathbf{w}) &= \cdots = \mathcal{C}_4 \\
f(\,\mathbf{w}) = f(\,\mathbf{w}) = f(\,\mathbf{w}) &= \cdots = \mathcal{C}_5 \\
f(\,\mathbf{w}) = f(\,\mathbf{w}) = f(\,\mathbf{w}) &= \cdots = \mathcal{C}_6 \\
f(\,\mathbf{w}) = f(\,\mathbf{w}) = f(\,\mathbf{w}) &= \cdots = \mathcal{C}_7 \\
f(\,\mathbf{w}) = f(\,\mathbf{w}) = f(\,\mathbf{w}) &= \cdots = \mathcal{C}_8 \\
f(\,\mathbf{w}) = f(\,\mathbf{w}) = f(\,\mathbf{w}) &= \cdots = \mathcal{C}_9
\end{aligned}
$$

Why function optimisation?

Training can often be cast as an optimisation problem:

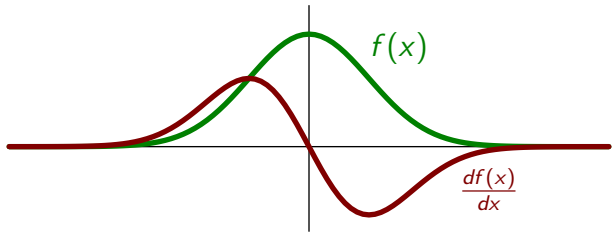
- Minimise the number of misclassifications
- Minimise the sum-squared error
- Maximise the probability of the parameters
- ...

How to optimise functions

If the function's nice:

- Take the partial derivative of the function with respect to the desired parameter(s)
- Set equal to zero
- Solve

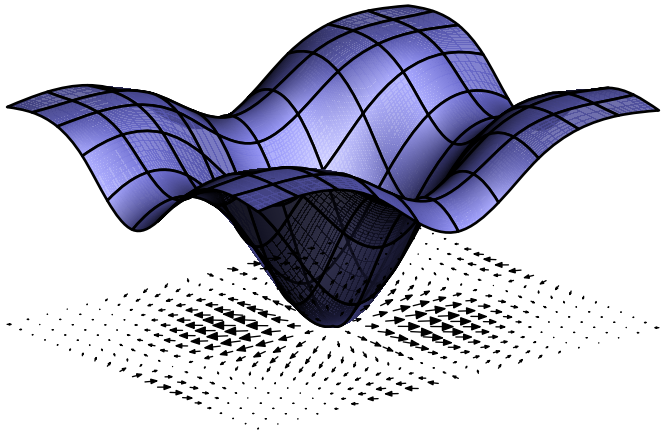
Function and derivative



How to optimise functions

If the function's not nice:

Gradient descent



How to optimise functions

If the function's not nice:

Gradient descent