*** Applied Machine Learning Fundamentals *** Mathematical Foundations

Daniel Wehner

SAP SE

November 14, 2019

Duale Hochschule Baden-Württemberg

Find all slides on GitHub

Lecture Overview

Unit I	Machine Learning Introduction
Unit II	**Mathematical Foundations**
Unit III	Bayesian Decision Theory
Unit IV	Probability Density Estimation
Unit V	Regression
Unit VI	Classification I
Unit VII	Evaluation
Unit VIII	Classification II
Unit IX	Clustering
Unit X	Dimensionality Reduction

Agenda November 14, 2019

Section:

Introduction

Introduction

Section:

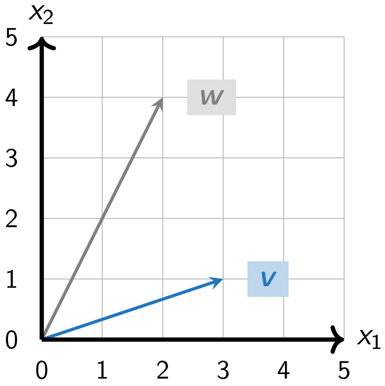
Linear Algebra

What is a Vector?

$$\boldsymbol{x} = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$$

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

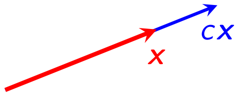
$$\boldsymbol{w} = \begin{bmatrix} 2 \\ 4 \end{bmatrix}$$



Multiplication by a Scalar

$$c\boldsymbol{x} = c\begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} cx_1 \\ cx_2 \end{bmatrix}$$

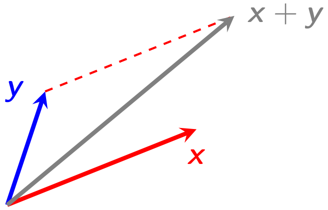
$$2\boldsymbol{v} = 2\begin{bmatrix} 3 \\ 1 \end{bmatrix} = \begin{bmatrix} 6 \\ 2 \end{bmatrix}$$



Addition of Vectors

$$\boldsymbol{x} + \boldsymbol{y} = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = \begin{bmatrix} x_1 + y_1 \\ x_2 + y_2 \end{bmatrix}$$

$$\boldsymbol{v} + \boldsymbol{w} = \begin{bmatrix} 3 \\ 1 \end{bmatrix} + \begin{bmatrix} 2 \\ 4 \end{bmatrix} = \begin{bmatrix} 5 \\ 5 \end{bmatrix}$$



Linear Combination of Vectors

$$\boldsymbol{u} = c_1 \boldsymbol{v}^{(1)} + c_2 \boldsymbol{v}^{(2)} + \cdots + c_n \boldsymbol{v}^{(n)}$$

Vector Transpose and inner and outer Product

- Vector transpose:

$$\boldsymbol{v} = \begin{bmatrix} 3 \\ 1 \end{bmatrix} \qquad \boldsymbol{v}^\intercal = \begin{bmatrix} 3 & 1 \end{bmatrix}$$

- Inner product / dot product / scalar product:

$$\boldsymbol{v} \cdot \boldsymbol{w} \equiv \boldsymbol{v}^\intercal \boldsymbol{w} \equiv \langle \boldsymbol{v}, \boldsymbol{w} \rangle$$

$$= \begin{bmatrix} 3 & 1 \end{bmatrix} \begin{bmatrix} 2 \\ 4 \end{bmatrix} = (3 \cdot 2) + (1 \cdot 4) = 10$$

Vector Transpose and inner and outer Product (Ctd.)

- Outer product:

$$\boldsymbol{v}\boldsymbol{w}^{\mathsf{T}} = \begin{bmatrix} 3 \\ 1 \end{bmatrix} \begin{bmatrix} 2 & 4 \end{bmatrix} = \begin{bmatrix} 6 & 12 \\ 2 & 4 \end{bmatrix}$$

The inner product yields a scalar value, the results of an outer product is a matrix!

Length of a Vector

- Length of a vector (Frobenius norm):

$$\|\boldsymbol{x}\| = \sqrt{\boldsymbol{x}^\intercal \boldsymbol{x}} \tag{1}$$

$$\|c\boldsymbol{x}\| = |c| \cdot \|\boldsymbol{x}\| \tag{2}$$

$$\|\boldsymbol{x} + \boldsymbol{y}\| \leqslant \|\boldsymbol{x}\| + \|\boldsymbol{y}\| \tag{3}$$

- Example:

$$\|\boldsymbol{v}\| = \sqrt{3^2 + 1^2} = 10$$

Angle between Vectors

- The angle between two vectors is given by:

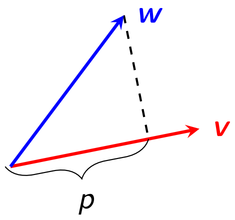
$$\cos \measuredangle(\boldsymbol{x}, \boldsymbol{y}) = \frac{\boldsymbol{x} \cdot \boldsymbol{y}}{\|\boldsymbol{x}\| \cdot \|\boldsymbol{y}\|} = \frac{\sum_{j=1}^{m} x_j \cdot y_j}{\sqrt{\sum_{j=1}^{m} (x_j)^2} \cdot \sqrt{\sum_{j=1}^{m} (y_j)^2}} \tag{4}$$

$$\cos \measuredangle(\boldsymbol{v}, \boldsymbol{w}) = \frac{\boldsymbol{v} \cdot \boldsymbol{w}}{\|\boldsymbol{v}\| \cdot \|\boldsymbol{w}\|} = \frac{10}{\sqrt{10} \cdot \sqrt{20}} \approx 0.71$$

- Inner product: $\boldsymbol{x} \cdot \boldsymbol{y} = \|\boldsymbol{x}\| \cdot \|\boldsymbol{y}\| \cdot \cos \measuredangle(\boldsymbol{x}, \boldsymbol{y})$

Projection of Vectors

- How is the projection of $\boldsymbol{x}$ onto $\boldsymbol{y}$ defined?
- Formally, we have:



$$\begin{aligned} p &= \|\boldsymbol{v}\| \cos \measuredangle(\boldsymbol{v}, \boldsymbol{w}) \\ &= \|\boldsymbol{v}\| \frac{\boldsymbol{v} \cdot \boldsymbol{w}}{\|\boldsymbol{v}\| \cdot \|\boldsymbol{w}\|} \\ &= \frac{\boldsymbol{v} \cdot \boldsymbol{w}}{\|\boldsymbol{w}\|} \end{aligned} \tag{5}$$

- Note that p is **not a vector!**

What is a Matrix?

General case ($\mathbb{R}^{n\times m}$):

$$\boldsymbol{X} = \begin{bmatrix} X_{11} & X_{12} & \cdots & X_{1m} \\ X_{21} & X_{22} & \cdots & X_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ X_{n1} & X_{n2} & \cdots & X_{nm} \end{bmatrix}$$

$$\boldsymbol{M} = \begin{bmatrix} 3 & 4 & 5 \\ 1 & 0 & 1 \end{bmatrix} \qquad \mathbb{R}^{2\times 3}$$

$$\boldsymbol{N} = \begin{bmatrix} 3 & 0 & 0 \\ 0 & 7 & 0 \\ 0 & 0 & 1 \end{bmatrix} \qquad \mathbb{R}^{3\times 3}$$

$$\boldsymbol{P} = \begin{bmatrix} 10 & 1 \\ 11 & 2 \end{bmatrix} \qquad \mathbb{R}^{2\times 2}$$

Matrix Transpose and Addition

- Transpose of a matrix:

$$\boldsymbol{M}^\intercal = \begin{bmatrix} 3 & 4 & 5 \\ 1 & 0 & 1 \end{bmatrix}^\intercal = \begin{bmatrix} 3 & 1 \\ 4 & 0 \\ 5 & 1 \end{bmatrix} \tag{6}$$

- Addition of matrices:

$$\boldsymbol{X} + \boldsymbol{Y} = \begin{bmatrix} X_{11} & X_{12} \\ X_{21} & X_{22} \end{bmatrix} + \begin{bmatrix} Y_{11} & Y_{12} \\ Y_{21} & Y_{22} \end{bmatrix} = \begin{bmatrix} X_{11} + Y_{11} & X_{12} + Y_{12} \\ X_{21} + Y_{21} & X_{22} + Y_{22} \end{bmatrix} \tag{7}$$

Matrix Multiplication

- Multiplication by scalars:

$$c\boldsymbol{X} = c\begin{bmatrix} X_{11} & X_{12} & X_{13} \\ X_{21} & X_{22} & X_{23} \end{bmatrix} = \begin{bmatrix} c \cdot X_{11} & c \cdot X_{12} & c \cdot X_{13} \\ c \cdot X_{21} & c \cdot X_{22} & c \cdot X_{23} \end{bmatrix} \tag{8}$$

- Matrix-vector multiplication:

$$\boldsymbol{z} = \boldsymbol{X}\boldsymbol{y} = \begin{bmatrix} X_{11} & X_{12} \\ X_{21} & X_{22} \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = \begin{bmatrix} X_{11} \cdot y_1 + X_{12} \cdot y_2 \\ X_{21} \cdot y_1 + X_{22} \cdot y_2 \end{bmatrix} \tag{9}$$

Matrix Multiplication (Ctd.)

- Matrix-matrix multiplication:

$$
\begin{aligned}
\boldsymbol{Z} &= \boldsymbol{X}\boldsymbol{Y} \\
&= \begin{bmatrix} X_{11} & X_{12} & X_{13} \\ X_{21} & X_{22} & X_{23} \end{bmatrix} \begin{bmatrix} Y_{11} & Y_{12} \\ Y_{21} & Y_{22} \\ Y_{31} & Y_{32} \end{bmatrix} \\
&= \begin{bmatrix} X_{11}Y_{11} + X_{12}Y_{21} + X_{13}Y_{31} & X_{11}Y_{12} + X_{12}Y_{22} + X_{13}Y_{32} \\ X_{21}Y_{11} + X_{22}Y_{21} + X_{23}Y_{31} & X_{21}Y_{12} + X_{22}Y_{22} + X_{23}Y_{32} \end{bmatrix} \quad (10)
\end{aligned}
$$

Matrix Inversion

- Matrix inversion is defined for **square matrices** $\boldsymbol{X} \in \mathbb{R}^{n \times n}$
- A matrix $\boldsymbol{X}$ multiplied by its inverse $\boldsymbol{X}^{-1}$ gives the **identity matrix**:

$$\boldsymbol{X}^{-1}\boldsymbol{X} = \boldsymbol{X}\boldsymbol{X}^{-1} = \boldsymbol{I} \tag{11}$$

$$\boldsymbol{I} = \begin{bmatrix} 1 & 0 & \dots & 0 \\ 0 & 1 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & 1 \end{bmatrix} \tag{12}$$

- If $\boldsymbol{X}^{-1}$ exists, we say that $\boldsymbol{X}$ is **non-singular**

Matrix Inversion (Ctd.)

- It holds that ($\boldsymbol{C}$ is the cofactor matrix):

$$\boldsymbol{X}^{-1} = \frac{1}{\det(\boldsymbol{X})} \boldsymbol{C}^{\intercal} \tag{13}$$

- A condition for invertability is that **the determinant has to be different than zero**
- **Example:**

$$\boldsymbol{X} = \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix} \qquad \det(\boldsymbol{X}) = 0 \qquad \boldsymbol{X}^{-1} = ?$$

Matrix Inversion Example

$$\boldsymbol{X} = \begin{bmatrix} 1 & 1/2 \\ -1 & 1 \end{bmatrix} \qquad \boldsymbol{X}^{-1} = \begin{bmatrix} 2/3 & -1/3 \\ 2/3 & 2/3 \end{bmatrix}$$

Please verify!

$$\boldsymbol{X}\boldsymbol{X}^{-1} = \boldsymbol{I} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} = \boldsymbol{X}^{-1}\boldsymbol{X}$$

Use for example the Gauss-Jordan algorithm to find the inverse!

Matrix Pseudoinverse

- **Question:** How can we invert a matrix $\boldsymbol{X} \in \mathbb{R}^{n \times m}$ which is not squared?
- **Left pseudoinverse** $\boldsymbol{X}^{\#}\boldsymbol{X}$:

$$\boldsymbol{X}^{\#}\boldsymbol{X} = \underbrace{(\boldsymbol{X}^\intercal \boldsymbol{X})^{-1}\boldsymbol{X}^\intercal}_{\text{left-multiplied}}\boldsymbol{X} = \boldsymbol{I}_m \tag{14}$$

- **Right pseudoinverse** $\boldsymbol{X}\boldsymbol{X}^{\#}$:

$$\boldsymbol{X}\boldsymbol{X}^{\#} = \boldsymbol{X}\underbrace{\boldsymbol{X}^\intercal(\boldsymbol{X}\boldsymbol{X}^\intercal)^{-1}}_{\text{right-multiplied}} = \boldsymbol{I}_n \tag{15}$$

Eigenvectors and Eigenvalues

- Some vectors $\boldsymbol{v}$ only change their length when multiplied by a matrix $\boldsymbol{X}$

Symmetric Matrices

- A squared $n \times n$ matrix $\boldsymbol{X}$ is symmetric, iff

$$\forall i, j: \qquad X_{ij} = X_{ji} \tag{16}$$

$$\boldsymbol{X} = \boldsymbol{X}^\intercal \tag{17}$$

- Some properties:
  - The inverse $\boldsymbol{X}^{-1}$ is also symmetric
  - Eigen-decomposition: $\boldsymbol{X}$ can be decomposed into $\boldsymbol{X} = \boldsymbol{Q}\boldsymbol{D}\boldsymbol{Q}^\intercal$, where the columns of $\boldsymbol{Q}$ are the eigenvectors of $\boldsymbol{X}$, and $\boldsymbol{D}$ is a diagonal matrix whose entries are the corresponding eigenvalues

Positive (semi-)definite Matrices

- A **squared symmetric** matrix $\boldsymbol{X}^{n \times n}$ is positive definite, iff for any vector $\boldsymbol{y} \in \mathbb{R}^n$:

$$\boldsymbol{y}^\intercal \boldsymbol{X} \boldsymbol{y} > 0 \tag{18}$$

- Or positive semi-definite, iff $\boldsymbol{y}^\intercal \boldsymbol{X} \boldsymbol{y} \geqslant 0$

Such matrices are important in machine learning. For instance, the covariance matrix is always positive semi-definite.

Section:

Statistics

Duale Hochschule
Baden-Württemberg

Random Variables

- What is a random variable?

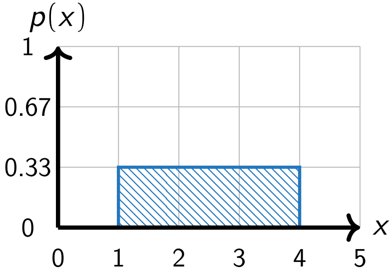
Random Variables

- What is a **random variable**?
  - It's a random number determined by chance (according to a distribution)
  - Random variables in machine learning: input data, output data, noise
- What is a **probability distribution**?

Random Variables

- What is a **random variable**?
  - It's a random number determined by chance (according to a distribution)
  - Random variables in machine learning: input data, output data, noise
- What is a **probability distribution**?
  - Describes the probability that a random variable is equal to a certain value
  - It can be given by the physics of an experiment (e. g. throwing dice)
  - **Discrete** vs. **continuous** distributions

Uniform Distribution



Every outcome is equally probable within a bounded region $\mathcal{R}$

$$p(x) = {}^{1}\!/_{\mathcal{R}} \tag{19}$$

Discrete Distributions

The random variables take on **discrete values**

Examples:

- When throwing a die, the possible values are given by a countably finite set:

$$x_i \in \{1, 2, 3, 4, 5, 6\}$$

- The number of sand grains at the beach (countably infinite set):

$$x_i \in \mathbb{N}$$

Discrete Distributions (Ctd.)

- All probabilities sum up to 1:

$$\sum_i p(x_i) = 1$$

- Discrete distributions are particularly important in classification
- A discrete distribution is described by a **probability mass function** (also called frequency function)

Bernoulli Distribution

- A **Bernoulli random variable** only takes on two values (e. g. 0 and 1):

$$x \in \{0, 1\} \tag{20}$$

$$p(x = 1|\mu) = \mu \tag{21}$$

$$\mathrm{Bern}(x|\mu) = \mu^x (1 - \mu)^{1-x} \tag{22}$$

$$\mathbb{E}\{x\} = \mu \tag{23}$$

$$\mathrm{var}\{x\} = \mu(1 - \mu) \tag{24}$$

- The only parameter is μ, i. e. the distribution is completely defined by this parameter

Binomial Distribution

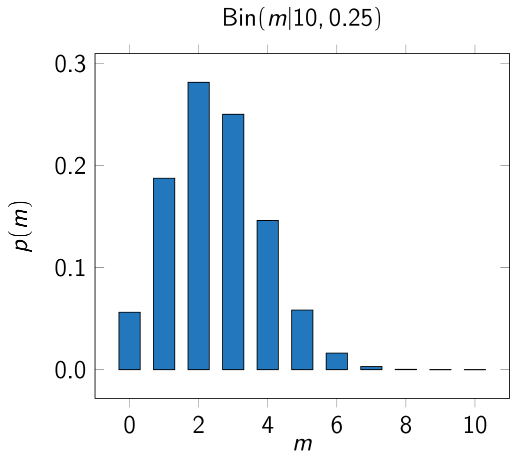
- Binomial variables are a sequence of n repeated Bernoulli variables
- **Example:** What is the probability of getting $m \in \mathbb{N}$ heads in N trials?

$$\text{Bin}(m|N, \mu) = \binom{N}{m} \mu^m (1 - \mu)^{N-m} \tag{25}$$

$$\mathbb{E}\{m\} = N\mu \tag{26}$$

$$\text{var}\{m\} = N\mu(1 - \mu) \tag{27}$$

Binomial Distribution (Ctd.)



Continuous Distributions

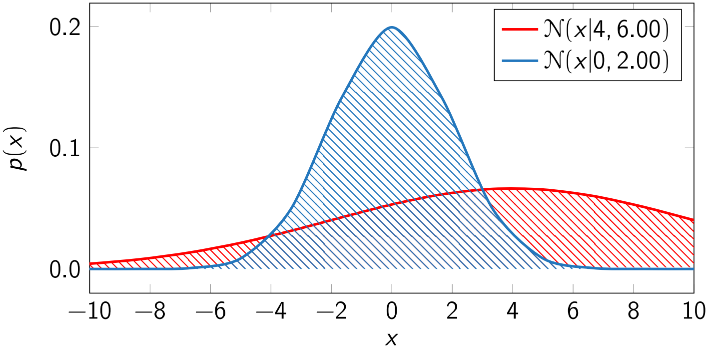
The random variables take on **continuous values**

- Continuous distributions are discrete distributions where the **number of discrete values goes to infinity** while the **probability of each value goes to zero**
- It's described by a probability density function which integrates to 1:

$$\int_{-\infty}^{+\infty} p(x)\,\mathrm{d}x = 1$$

Important

Gaussian Distribution



$$p(x) = \mathcal{N}(x|\mu, \sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left\{-\frac{(x-\mu)^2}{2\sigma^2}\right\} \tag{28}$$

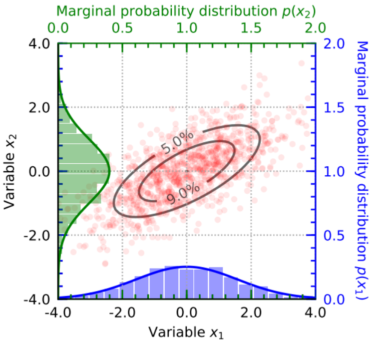
Central Limit Theorem

Central Limit Theorem:
The distribution of the sum of N i. i. d. (independent and identically distributed) random variables becomes increasingly Gaussian as N increases.

- The Gaussian distribution is one among the most important distributions
- Gaussians are often a good model
- Working with Gaussians leads to **analytical solutions for complex operations**

Multivariate Gaussian Distribution

$$p_D(\boldsymbol{x}) = \mathcal{N}(\boldsymbol{x}|\boldsymbol{\mu}, \boldsymbol{\Sigma}) = \frac{1}{\sqrt{(2\pi)^D|\boldsymbol{\Sigma}|}} \exp\left\{-\frac{1}{2}(\boldsymbol{x}-\boldsymbol{\mu})^\intercal \boldsymbol{\Sigma}^{-1}(\boldsymbol{x}-\boldsymbol{\mu})\right\} \tag{29}$$



For clarification: $\boldsymbol{x}$ and $\boldsymbol{\mu}$ are vectors while $\boldsymbol{\Sigma}$ is a matrix. The probability given by $\mathcal{N}(\boldsymbol{x}|\boldsymbol{\mu}, \boldsymbol{\Sigma}) \in [0; 1]$ is still a scalar value!

Basic Rules of Probability

- **Joint distribution:**

$$p(x, y) \tag{30}$$

- **Marginal distribution:**

$$p(y) = \int_x p(x, y)\,\mathrm{d}x \tag{31}$$

- **Conditional distribution:**

$$p(y|x) = \frac{p(x, y)}{p(x)} \tag{32}$$

Basic Rules of Probability (Ctd.)

- **Probabilistic independence:**

$$p(x, y) = p(x)p(y) \tag{33}$$

- **Chain rule of probabilities:**

$$\begin{aligned} p(x_1, \ldots, x_n) &= p(x_1|x_2, \ldots, x_n)p(x_2, \ldots, x_n) \\ &= p(x_1|x_2, \ldots, x_n)p(x_2|x_3, \ldots, x_n) \ldots p(x_{n-1}|x_n)p(x_n) \end{aligned} \tag{34}$$

- **Bayes rule:**

$$p(y|x) = \frac{p(x|y)p(y)}{p(x)} \tag{35}$$

Expectation

$$\mathbb{E}_{x\sim p(x)}\{f(x)\} = \mathbb{E}_x\{f\} = \mathbb{E}\{f\} = \sum_x p(x)f(x) \quad \text{discrete case} \tag{36}$$

$$= \int_x p(x)f(x)\,\mathrm{d}x \quad \text{continuous case} \tag{37}$$

Approximate expectation:

$$\mathbb{E}\{f\} = \int_x p(x)f(x)\,\mathrm{d}x \approx \frac{1}{N}\sum_{i=1}^{N} f(x_i) \tag{38}$$

Expectation (Ctd.)

- Some rules of expectations:
  - $\mathbb{E}\{a\boldsymbol{x}\} = a\mathbb{E}\{\boldsymbol{x}\}$
  - $\mathbb{E}\{\boldsymbol{x} + \boldsymbol{y}\} = \mathbb{E}\{\boldsymbol{x}\} + \mathbb{E}\{\boldsymbol{y}\}$
  - $\mathbb{E}\{\boldsymbol{x}\boldsymbol{y}\} = \mathbb{E}\{\boldsymbol{x}\}\mathbb{E}\{\boldsymbol{y}\}$ (if $\boldsymbol{x}$ and $\boldsymbol{y}$ are independent)
  - $\mathbb{E}\{\sum_i a_i x_i\} = \sum_i a_i \mathbb{E}\{x_i\}$
- Expectations of functions:
  - $\mathbb{E}\{g(\boldsymbol{x})\} = \int_{\boldsymbol{x}} p(\boldsymbol{x}) g(\boldsymbol{x}) \,\mathrm{d}\boldsymbol{x}$
  - In general: $\mathbb{E}\{g(\boldsymbol{x})\} \neq g(\mathbb{E}\{\boldsymbol{x}\})$

Section:

Optimization

Motivation

Every machine learning problem is an optimization problem!

- In every machine learning problem, you will have:
  - an objective function you want to optimize
  - data you want to learn from
  - parameters which need to be learned
  - assumptions on your problem, your data and how the world works
- Thus, we would like to have general solutions to the problem of learning
- Machine learning provides suitable objective functions for optimization based on the data, different models embody different objective functions and assumptions

Constrained Optimization

How to formalize an optimization problem

$$
\begin{aligned}
\min_{\theta} \; & J(\theta, D) = \dots && \leftarrow \text{cost function / objective} \\
\text{s. t.} \; & f(\theta, D) = 0 && \leftarrow \text{equality constraints} \\
& g(\theta, D) \geqslant 0 && \leftarrow \text{inequality constraints}
\end{aligned}
$$

What should an ideal optimization problem, i. e. the cost function and constraints look like?

Constrained Optimization

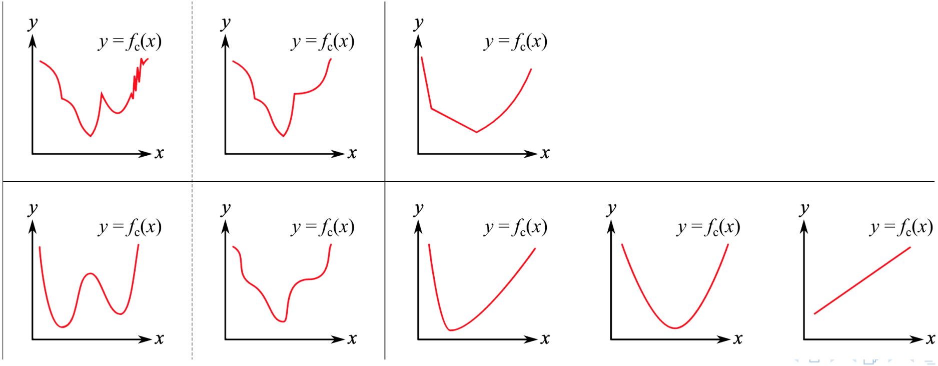
How to formalize an optimization problem

$$\begin{aligned} \min_{\theta} \; & J(\theta, D) = \dots && \leftarrow \text{convex function} \\ \text{s.t.} \; & f(\theta, D) = 0 && \leftarrow \text{linear function} \\ & g(\theta, D) \geqslant 0 && \leftarrow \text{convex set} \end{aligned}$$

Cost Functions

Which cost functions are there? Ideally, the cost function is convex

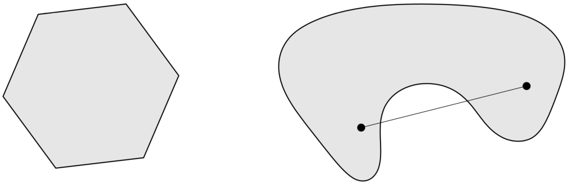
non-convex | convex



Convexity

Convex Sets

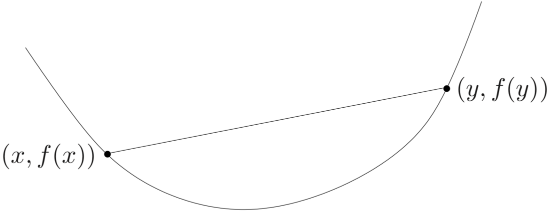
- A set $C \subseteq \mathbb{R}^n$ is convex if for each $x, y \in C$ and any $\alpha \in [0, 1]$, $\alpha x + (1 - \alpha) y \in C$. Examples are $\mathbb{R}^n$ and norm balls.



Convexity

Convex Functions

- A function $f : \mathbb{R}^n \to \mathbb{R}$ is convex if for $x, y \in dom(f)$ and any $\alpha \in [0, 1]$, $f(\alpha x + (1-\alpha)y) \leqslant \alpha f(x) + (1-\alpha)f(y)$. Examples are linear functions $f(x) = w^T x + b$ and quadratic functions $f(x) = x^T A x + b^T x + c$.



Convexity

Why are convex cost functions so nice?

- Local solutions are global optima
- Efficient implementations of optimizers are available

Convexity

How to recognize a convex function? Convexity conditions

- First-order convexity condition:
  Suppose $f : \mathbb{R}^n \rightarrow \mathbb{R}$ is differentiable. The function f is convex iff $f(y) \geqslant f(x) + \nabla f(x)^T (y - x) \;\; \forall x, y \in dom(f)$.
- Second-order convexity condition:
  Suppose $f : \mathbb{R}^n \rightarrow \mathbb{R}$ is twice differentiable. The function f is convex iff $\nabla^2 f(x) \geqslant 0 \forall x \in dom(f)$.

Constrained Optimization

How to solve an optimization problem with constraints

$$\min\ f(x,y) = 2 \cdot y + x$$
$$\text{s.t. } 0 = g(x,y) = y^2 + xy - 1$$

- Convert the problem to an unconstrained one
- Introduce Lagrange multipliers

Constrained Optimization

Lagrange multipliers

$$\min \; f(x, y) = 2 \cdot y + x$$
$$\text{s. t. } \; 0 = g(x, y) = y^2 + xy - 1$$

$$L(x, y, \lambda) = f(x, y) + \lambda g(x, y)$$
$$\frac{\partial L}{\partial x} = 1 + \lambda y$$
$$\frac{\partial L}{\partial y} = 2 + 2\lambda y + \lambda x$$
$$\frac{\partial L}{\partial \lambda} = y^2 + xy - 1$$

Constrained Optimization

Lagrange multipliers

$$\text{I. } 0 = 1 + \lambda y$$
$$\text{II. } 0 = 2 + 2\lambda y + \lambda x$$
$$\text{III. } 0 = y^2 + xy - 1$$

$$\text{I. } \lambda = -\frac{1}{y}$$
$$\text{I.} \rightarrow \text{II. } x = 0$$
$$\text{III. } y = \pm 1$$

Numerical Optimization

What to do if we cannot solve it analytically?

- Different numerical optimization algorithms exist for optimizing a function numerically on a computer if we can't solve it analytically
- Many approaches incrementally update an estimate $\theta_{new} := \theta_{old} + \alpha\delta\theta$ of the optimal parameters, so that after each update $J(\theta_{new}) < J(\theta_{old})$
- The challenge is to find the right step size α and direction $\delta\theta$
- Different algorithms differ in the number of required iterations, the computational cost per iteration, the convergence guarantees, the robustness with noisy cost functions and their memory usage

Numerical Optimization

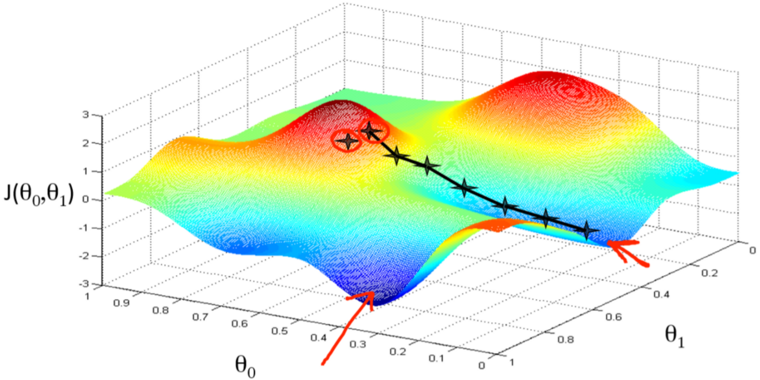
Optimization algorithms

- There are various approaches to numerical optimization
- Gradient-based methods require differentiable functions and not too many iterations, but only guarantee to find a local optimum, examples are:
  - Gradient Descent (with constant, variable or line-search-optimized step size)
  - (L-)BFGS
  - Conjugate Gradient Descent
- Non-gradient based methods may find a global optimum, but require a large number of steps, examples are:
  - Genetic Algorithms
  - Non-Linear Simplex
  - Nelder-Mead

Numerical Optimization

There are many other things you have to consider

Initialization also matters...



Want to learn more about optimization?

Every machine learning problem is an optimization problem!

- Deep Learning book chapters 4.3, 4.4 and 8 (Link chapters 4.3, 4.4, Link chapter 8) are highly recommended
- Boyd & Vandenberghe, Convex Optimization (Link)
- Stanford convex optimization course (Link)
- MOOC on constrained optimization (Link)

Section:

Wrap-Up

Summary

-

Self-Test Questions

1.

What's next...?

Unit I	Machine Learning Introduction
Unit II	Mathematical Foundations
Unit III	**Bayesian Decision Theory**
Unit IV	Probability Density Estimation
Unit V	Regression
Unit VI	Classification I
Unit VII	Evaluation
Unit VIII	Classification II
Unit IX	Clustering
Unit X	Dimensionality Reduction

Recommended Literature and further Reading

Thank you very much for the attention!

Topic: *** Applied Machine Learning Fundamentals *** Mathematical Foundations
Date: November 14, 2019

Contact:
Daniel Wehner
SAP SE
daniel.wehner@sap.com

Do you have any questions?