mml-books

September 24, 2025

[1]:
```
from IPython.core.display import HTML
HTML("""
<style>
div.text_cell_render {
    line-height: 1.5 !important;
}
</style>
""")
```

[1]:
```
<IPython.core.display.HTML object>
```

1 Chapter 5 : Vector Calculus

(see copy book for the missing sections) ## 5.6 Backpropagation and Automatic Differentiation (see notebook for missing subsections) ### 5.6.2 Automatic Differentiation - Automatic differentiation applies a series of elementary arithmetic operations, e.g., addition and multiplication and elementary functions, e.g., sin, cos, exp, log. - by applying the chain rule to these operations, the gradient of quite complicated functions can be computed automatically. - from dataflow between x and y, and by applying intermediate variables a,b, we can obtain this:

$$\frac{dy}{dx} = \frac{dy}{db} \cdot \frac{db}{da} \cdot \frac{da}{dx}$$

- Intuitively, the forward and reverse mode differ in the order of multiplication. Due to the associativity of matrix multiplication, we can choose between :

$$\frac{dy}{dx} = \left(\frac{dy}{db} \cdot \frac{db}{da}\right) \cdot \frac{da}{dx}$$
$$\frac{dy}{dx} = \frac{dy}{db} \cdot \left(\frac{db}{da} \cdot \frac{da}{dx}\right)$$

- reverse mode (5.120) : gradients are propagated backward through the graph, i.e., reverse to the data flow. (backpropagation) the reverse mode is computationally significantly cheaper than the forward mode in Neural Networks. - forward mode (5.121): where the gradients flow with the data from left to right through the graph. - exemple : see P162 which explains the concept of intermediate variables Computation graph with inputs x, function values f , and intermediate variables a, b, c, d, e.The set of equations that include intermediate variables can be thought of as a computation graph. In this examples, it shows how we can obtain $ $\frac{\partial f}{\partial x}$$.we observe that the computation required for calculating the derivative is of similar complexity as the computation of the function itself.– **Forma**

be the input variables to the function, $x_d + 1, ..., x_D - 1$ be the intermediate variables, and x_D the output variable. Then the computation graph can be expressed as follows:

$$\text{For } i = d+1, ..., D : x_i = g_i\left(x_{\text{Pa}(x_i)}\right)$$

Where the $ g_i(\cdot) $are elementary functions and $x_{\text{Pa}(x_i)}$ are the parent nodes of the variable xi in the graph. F $\frac{\partial f}{\partial x_i} = \sum_{x_j : x_i \in Pa(x_j)} \frac{\partial f}{\partial x_j} \frac{\partial x_j}{\partial x_i} = \sum_{x_j : x_i \in Pa(x_j)} \frac{\partial f}{\partial g_j} \frac{\partial g_j}{\partial x}$ Where $ Pa (x_j) is the set of parent nodes of x_j in the computation graph. (5.145) is the backpropagation of the gradient through the computation graph. - For neural network training, we backpropagate the error of the prediction with respect to the label.

1.1 5.7 Higher-Order Derivatives

- Sometimes, we are interested in derivatives of higher order, e.g., when we want to use Newton's Method for optimization, which requires second-order derivatives (Nocedal and Wright, 2006).
- Consider a function $f : \mathbb{R}^2 \to \mathbb{R}$ of two variables $x, y $. We use the following notation for higher-order partial derivatives (and for gradients):
  - $\partial^2 f$ / $\partial^2 x$ is the second partial derivative of f with respect to x. $\frac{\partial^n f}{\partial^n x}$ is the nth partial derivative of f with respect to x.
  - $\partial^2 f$ / $\partial y \cdot \partial x = \frac{\partial}{\partial y}\left(\frac{\partial f}{\partial x}\right)$ is the partial derivative obtained by first partial differentiating with respect to x and then with respect to y. $\frac{\partial^2 f}{\partial x \cdot \partial y}$ s the partial derivative obtained by first partial differentiating by y and then x.
- *Hessian* : the collection of all second-order partial derivatives
- If f (x, y) is a twice (continuously) differentiable function, then $\frac{\partial^2 f}{\partial y \cdot \partial x} = \frac{\partial^2 f}{\partial x \cdot \partial y}$; i.e, the order of differentiation does not matter.
- We obtain then the *Hessian matrix* :

$$\mathbf{H} = \begin{bmatrix} \frac{\partial^2 f}{\partial x^2} & \frac{\partial^2 f}{\partial x \partial y} \\ \frac{\partial^2 f}{\partial x \partial y} & \frac{\partial^2 f}{\partial y^2} \end{bmatrix}$$

- This matrix is symmetric. noted $ \nabla^2_{x,y} f(x, y)$, which is generaly an nxn matrix for $x \in \mathbb{R}^n$
- The Hessian measures the curvature of the function locally around (x, y).

1.2 5.8 Linearization and Multivariate Taylor Serie

- linear approximation of f around x_0:

$$f(x) \approx f(x_0) + (\nabla_x f)(x_0)(x - x_0) \tag{5.148}$$

  $(\nabla_x f)(x_0)$: the gradient of f with respect to x, evaluated at x_0 (an input)
- The original function is approximated by a straight line. This approximation is locally accurate, but the farther we move away from x_0 the worse the approximation gets.
- a special case of a multivariate Taylor series expansion of f at x_0, where we consider only the first two terms
- **Definition of Multivariate Taylor Series** :
  - Consider the function :

$$f : \mathbb{R}^D \to (R), \quad x \mapsto f(x), \quad x \in \mathbb{R}^D$$

    => is smoot at x_0

– When we define the difference vector $\delta := x - x_0$, the *Multivariate Tailor series* of f at (x_0) is defined as

$$f(x) = \sum_{k=0}^{\infty} \frac{D_x^k f(x_0)}{k!} \delta^k$$

Where $D_x^k f(x_0)$ is the k -th (total) derivative of f with respect to x, evaluated at x_0 .

- **Taylor Polynomial** : The Taylor polynomial of degree n of f at x0 contains the first n + 1 components of the series in (5.151) and is defined as :

$$T_n(x) = \sum_{k=0}^{n} \frac{D_x^k f(x_0)}{k!} \delta^k$$

δ^k is not defined for vectors $x \in (R)^D, D > 1$ and $k > 1$ Both $D_x^k f(x_0)$ and δ^k are k-th order tensors, i.e, k-dimensional arrays. The kth-order tensor $\delta^k \in \mathbb{R}^{DxDx...xD}$ is obtained as a k-fold outer product, denoted $\otimes, of the vector \delta \in \mathbb{R}^D$. For example $$ \delta^2 := \delta \otimes \delta = \delta, \delta T, \quad \delta^2![i,j] = \delta_i, \delta_j Exemple : Taylor Series Expansion of a Function with Two Vari - ables(.

1.3 5.9 Further Reading (p170)

- **Matrix differentials**: Matrix Differential Calculus with Applications in Statistics and Econometrics, Magnus, Jan R., and Neudecker, Heinz. 2007.
- **Automatic differentiation**: Elliott, Conal. 2009. Beautiful Differentiation. In: International Conference on Functional Programming. Griewank, Andreas, and Walther, Andrea. 2008. Evaluating Derivatives, Principles and Techniques of Algorithmic Differentiation. SIAM.
- **Extended Kalman filter**: Maybeck, Peter S. 1979. Stochastic Models, Estimation, and Control. Academic Press.
- **Other deterministic ways** to approximate the integral. unscented transform : Julier, Simon J., and Uhlmann, Jeffrey K. 1997. A New Extension of the Kalman Filter to nonlinear Systems. In: Proceedings of AeroSense Symposium on Aerospace/Defense Sensing, Simulation and Controls. or the **Laplace approximation** of Murphy, Kevin P. 2012. Machine Learning: A Probabilistic Perspective. MIT Press.

2 Chapter 6: Probability and Distributions

- study of uncertainty
- use this probability to measure the chance of something occurring in an experiment
- *random variable* : Quantifying uncertainty. a function that maps outcomes of random experiments to a set of properties that we are interested in.
- *probability distribution* : a function that measures the probability that a particular outcome (or set of outcomes) will occur. used in probabilistic modeling (Section 8.4), graphical models (Section 8.5), and model selection (Section 8.6).
- *probability space* :the sample space, the events, and the probability of an event

2.1 6.1 Construction of a Probability Space (p172)

The theory of probability aims at defining a mathematical structure to describe random outcomes of experiments. (see Jaynes, Edwin T. 2003. Probability Theory: The Logic of Science. Cambridge

University Press.) ### 6.1.1 Philosophical Issues - For plausible reasoning it is necessary to extend the discrete true and false values of truth to continuous plausibilities - probability theory can be considered a generalization of Boolean logic. - In ML, it is used to formalize the design of automated reasoning systems. (Pearl, Judea. 1988. Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference. Morgan Kaufmann.) - plausibility by E. T. Jaynes (1922 - 1998) : 1. The degrees of plausibility are represented by real numbers. 2. These numbers must be based on the rules of common sense. 3. . The resulting reasoning must be consistent, with the three following meanings of the word "consistent": consistency or non-contradiction, honesty, reproducibility. - **Cox–Jaynes theorem** (p174) : universal mathematical rules that apply to plausibility p - **Remark** : two interpretation of the probability in ML: the Bayesian (*"subjective probability" or "degree of belief"*) and frequentist interpretations (see Bishop, Christopher M. 2006. Pattern Recognition and Machine Learning. Springer.) - in ML, consider wether we are trying to model something categorical (a discrete random variable) or something continuous (a continuous random variable).

2.1.1 6.1.2 Probability and Random Variables

- three idea to not to be confused
  - **probability space**: allows us to quantify the idea of a probability
  - **random variables** :transfers the probability to a more convenient (often numerical) space.
  - **distribution or law associated with a random variable**
- from *Grinstead, Charles M., and Snell, J. Laurie. 1997. Introduction to Probability. American Mathematical Society.*, modern probability is based on:
  - *The sample space* Ω ("state space" "sample description space", "possibility space," and "event space") : he set of all possible outcomes of the experiment, usually denoted by Ω.
  - *The event space* $\mathcal{A}$ (collection of subsets of Ω) : the space of potential results of the experiment. A subset A of the sample space Ω is in the event space $\mathcal{A}$ if at the end of the experiment we can observe whether a particular outcome $\omega \in \Omega$ is in A.
  - *The probabilty* P ($P(A)$) : With each event $A \in \mathcal{A}$, we associate a number $P(A)$ that measures the probability or degree of belief that the event will occur. in $[0, 1]$ and $P(\Omega) = 1$
- The *probability space* $(\Omega, \mathcal{A}, P)$ models a real-world process or phenomenon. (referred to as an experiment) with random outcomes. In ML, instead we refer to it as *probabilities on quantities of interest* denoted by $\mathcal{T}$.
- $\mathcal{T}$: *Taget space*, **element of** $\mathcal{T}$:states
- ***Random Variable*** : It is a function $X : \Omega \to \mathcal{T}$ that takes an element of Ω (an outcome) and returns a particular quantity of interest x, a value in $\mathcal{T}$. For any subset $S \subseteq \mathcal{T}$, we associate $P_X(S) \in [0, 1]$ (the probability) to a particular event occurring corresponding to the random variable X .
- Exemple (P176) : Consider a statistical experiment where we model a funfair game consisting of drawing two coins from a bag (with replacement) . (see the rest on the book)
- *the probability of the output of* $X \neq$ *the probability of the samples in* Ω
- $X^{-1}(S)$: Pre-image of S by X: The set of elements of Ω that map to S under X: $\{\omega \subset \Omega : X(\omega) \subset S\}$
- the random variable X is to associate it with the probability of the pre-image of S:

$$P_X(S) = P(X \subset S) = P(X^{-1}(S)) = P(\{\omega \subset \Omega : X(\omega) \subset S\}) \tag{6.8}$$

- *Law Distribution* of random variable X: the function P_X or equivalently $P \circ X^{-1}$

- **Remark** : The target space, that is, the range $\mathcal{T}$ of the random variable X , is used to indicate the kind of probability space, i.e., a $\mathcal{T}$ random variable. When $\mathcal{T}$ is finite or countably infinite, this is called a discrete random variable (Section 6.2.1). For continuous random variables (Section 6.2.2), we only consider $\mathcal{T} = \mathbb{R}$ or $\mathcal{T} = \mathbb{R}^D$.

2.1.2 6.1.3 Statistics

- Using probability, we can consider a model of some process, where the underlying uncertainty is captured by random variables, and we use the rules of probability to derive what happens.
- In statistics, we observe that something has happened and try to figure out the underlying process that explains the observations.
- Thus ML is very close to statistics but We can use the rules of probability to obtain a "best-fitting" model for some data.
- in ML, we are intereste in generalization error (perforamance analysis). This analysis of future performance relies on probability and statistics (see *Concentration Inequalities: A Nonasymptotic Theory of Independence. Oxford University Press* or *Understanding Machine Learning: From Theory to Algorithms. Cambridge University Press.*.

2.2 6.2 Discrete and Continuous Probabilities

- focus our attention on ways to describe the probability of an event
- Depending on whether the target space is discrete or continuous, the natural way to refer to distributions is different
- *probability mass function* for discrete target space $\mathcal{T}$: $P(X = x)$ the probability that a random variable X takes a particular value $x \in \mathcal{T}$.
- *cumulative distribution function* $P(X \leq x)$: by convention to pecify the probability that a random variable X is less than a particular value x. Generaly, we specify the probability that a random variable X is in an interval, denoted by $P(a \leq X \leq b)$ for $a \leq b$.
- **Remark** :
  - *univariate distribution* to refer to distributions of a single random variable.
  - *multivariate distributions*: a vector of random variables ### 6.2.1 Discrete Probabilities
- Target space is discrete => The probabilty distribution of multiple random variables is same as filling out a (multidimensional) array of numbers.
- *Joint probability*: The Cartesian product of the target spaces of each of the random variables.

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

Where n_{ij} is the number of events with state x_i and y_j and N the total number of events.

or

$$P(X = x_i, Y = y_j) = P(X = x_i \cap Y = y_j)$$

- For two random variables X and Y , the probability that $X = x$ and $Y = y$ is (lazily) written as $p(x, y)$ and is called the joint probability. - The *marginal probability* that X takes the value x irrespective of the value of random variable Y is (lazily) written as $p(x)$. We write $X \sim p(x)$ to denote that the random variable X is distributed according to $p(x)$. - *Conditional probabilty* : if we consider only $X = x$, the probabilty for $Y = y$ is written as $p(y|x)$. - Exemple : see p179 - in ML, we use discrete probability distributions to model *categorical variables.* - Discrete distributions are also often used to construct probabilistic models that combine a finite number of continuous distributions ### 6.2.2 Continuous Probabilities - real-valued random variables.target spaces are

intervals of the real line R. - In this book, we pretend that we can perform operations on real random variables as if we have discrete probability spaces with finite states. - **Remark**: Continuous spaces have two additional technicalities (see p180) - *measure*: The size of a set (ex:cardinality of discrete sets, length of an interval in $\mathbb{R}$, colume of a region in $\mathbb{R}^d$ - *Borel* $\sigma - algebra$: Sets that behave well under set operations and additionally have a topology (see Jacod, Jean, and Protter, Philip. 2004. Probability Essentials. Springer.) - In this book, we pick random variables with their corresponding *Borel* $\sigma - algebra$, thus random variables are real-valued vector in $\mathbb{R}^D$ - **Probability Density Function** (pdf) : a function $f : \mathbb{R}^D \to \mathbb{R}$ where : 1. $\forall x \in \mathbb{R}^D : f(x) \geq 0$ 2. Its integral exists and

$$\int_{R^D} f(x)\, dx = 1 \tag{6.15}$$

For probability mass functions (pmf) of discrete random variables, the integral in (6.15) is replaced with a sum. - **the law or distribution of the random variable** X : We associate a random variable X with this function f by

$$P(a \leq X \leq b) : \int_a^b f(x)\, dx, \tag{6.16}$$

Where $a, b \in \mathbb{R}$ and $x \in \mathbb{R}$ are outcomes of the continuous random variable X. **Remark**: $P(X = x)$ is a set of measure zero. This is like trying to specify an interval in (6.16) where a = b. - **Cumulative Distribution Function** (cdf) of a multivariate real-valued random variable X with states $x \in \mathbb{R}^D$ is given by:

$$F_X(x) = P(X_1 \leq x_1, \cdots, X_D \leq x_D), \tag{6.17}$$

where $X = [X_1, \cdots, X_D]^T$, $x = [x_1, \cdots, x_D]^T$ and the right-hand side represents the probability that random variable Xi takes the value smaller than or equal to xi . - The cdf can be expressed also as the integral of the robability density function (pdf) f (x) so that

$$F_X(x) = \int_{-\infty}^{x_1} \cdots \int_{-\infty}^{x_D} f(z_1, \cdots, z_D)\, dz_1 \cdots dz_D \tag{6.18}$$

- **Remark** $f(x)$(pdf): is a nonnegative function that sums to one. There are cdfs, which do not have corresponding pdfs. *law of a random variable* X: the association of a random variable X with the pdf $f(x)$.

2.2.1 6.2.3 Contrasting Discrete and Continuous Distributions

- probabilities are positive and the total probability sums up to one. thus , for discrete random variables, $ p $\in$ [0, 1] $but for continuous random variables, the normalization does not imply that the value of the density is less th

- **Remark**: The states $z_1, ..., z_d$ do not in principle have any structure, i.e., there is usually no way to compare them, for example $z_1 =$ red, $z_2 =$ green, $z_3 =$ blue. However, in many machine learning applications discrete states take numerical values, e.g., $z_1 = -1.1, z_2 = 0.3, z_3 = 1.5$, where we could say $z1 < z2 < z3$
- In ML, there is no distinction of sample space Ω, the target space $\mathcal{T}$, and the random variable X.
- $\forall x \in \mathcal{T}$, $p(x)$ denotes the probability that random variable X has the outcome x. for discrete Random Variable $p(x) = P(X = x)$ (the probabilty mass function); pmf: Distribution, for continuous variables, $p(x)$ is the pdf (density). cdf $P(X \leq x)$ is also refered as Distributions
- **Remark** : "probability distribution => for discrete probability mass functions & for continuous probability density functions, although this is technically incorrect.

2.3 6.3 Sum Rule, Product Rule, and Bayes' Theorem

Recall: joint distribution = $p(x, y)$, marginal distributions = $p(x)$ or $p(y)$; conditional distribution of y given x = $p(y|x)$. (see *Jaynes, Edwin T. 2003. Probability Theory: The Logic of Science. Cambridge University Press.*) 1. **Sum rule** or **marginalization property**

$$p(x) = \begin{cases} \sum_{y \in \mathcal{Y}} p(x, y), & \text{if } y \text{ is discrete} \\ \int_{\mathcal{Y}} p(x, y)\, dy, & \text{if } y \text{ is continuous} \end{cases} \tag{6.20}$$

where $\mathcal{Y}$ are the states of the target space of random variable Y. - The sum rule relates the joint distribution to a marginal distribution. - the sum rule can be applied to any subset of the random variables:

$$p(x_i) = \int p(x_1, \cdots, x_D)\, dx_{\setminus i} \tag{6.21}$$

we integrate/sum out all random variables except x_i

- **Remark**: Sum rule is generally computationally hard because it performs high-dimensional sums or integral for my variables.

2. **Product Rule** : relates the joint distribution to the conditional distribution via:

$$p(x, y) = p(y|x)p(x) \tag{6.22}$$

- The product rule can be interpreted as the fact that every joint distribution of two random variables can be factorized (written as a product) of two other distributions.

$$p(y, x) = p(x|y)p(y)$$

3. **Bayes Theorem** or *probabilistic inverse* : Bayes' theorem is used to draw some conclusions about x given the observed values of y . allows us to invert the relationship between x and y given by the likelihood.

$$\underbrace{p(x|y)}_{\text{posterior}} = \frac{\overbrace{p(y|x)}^{\text{likelihood}}\ \overbrace{p(x)}^{\text{prior}}}{\underbrace{p(y)}_{\text{evidence}}} \tag{6.23}$$

- it is a direct consequence of the product rule in (6.22) since $p(x, y) = p(y, x)$
- *prior*: encapsulates our subjective prior knowledge of the unobserved (latent) variable x before observing any data.
- *likelihood* : describes how x and y are related, and in the case of discrete probability distributions, it is the probability of the data y if we were to know the latent variable x. The likelihood is sometimes also called the "measurement model". "probability of y given x"
- *posterior* : the quantity of interest in Bayesian statistics because it expresses exactly what we are interested in, i.e., what we know about x after having observed y .
- **marginal likelihood/evidence** :

$$p(y) = \int p(y|x)p(x)\, dx = \mathbb{E}_X\left[p(y|x)\right] \tag{6.27}$$

- the marginal likelihood is independent of x, and it ensures that the posterior $p(x|y)$ is normalized. The marginal likelihood can also be interpreted as the expected likelihood where we take the expectation with respect to the prior $p(x)$.

- **Remark** : In Bayesian statistics, the posterior distribution is the quantity of interest as it encapsulates all available information from the prior and the data. having the full posterior can be very useful for a downstream task.

2.4 6.4 Summary Statistics and Independence

summarizing sets of random variables and comparing pairs of random variables. useful view of how a random variable behaves ### 6.4.1 Means and Covariances - useful to describe properties of probability distributions (expected values and spread) - The concept of the expected value is central to machine learning - the **Expected Value** (or *the law of the unconscious statistician*) of a function $g : \mathbb{R} \to \mathbb{R}$ of a univariate continuous random variable $X \sim p(x)$ is given by

$$\mathbb{E}\left[g(x)\right] = \int_{\mathcal{X}} g(x)p(x)\,dx \tag{6.28}$$

- Correspondingly, the *expected value* of a function g of a discrete random variable $X \sim p(x)$ is given by

$$\mathbb{E}\left[g(x)\right] = \sum_{x\in\mathcal{X}} g(x)p(x), \tag{6.2}$$

where $\mathcal{X}$ is the set of possible outcomes (the target space) of the random variable X. In this section, we consider discrete random variables to have numerical outcomes. - **Remark** : We consider multivariate random variables X as a finite vector of univariate random variables $[X_1, \cdots, X_D]_\top$. For multivariate random variables, we define the expected value element wise :

$$\mathbb{E}_X\left[g(x)\right] = \begin{bmatrix} \mathbb{E}_{X_i}\left[g(x_i)\right] \\ \vdots \\ \mathbb{E}_{X_D}\left[g(x_D)\right] \end{bmatrix} \in \mathbb{R}^D \tag{6.30}$$

where the subscript E_{X_d} indicates that we are taking the expected value with respect to the dth element of the vector x.

The definition of the mean , is a special case of the expected value, obtained by choosing g to be the identity function.

Mean : The mean of a random variable X with states $x \in \mathbb{R}^D$ is an average and is defined as

$$\mathbb{E}_X\left[x\right] = \begin{bmatrix} \mathbb{E}_{X_i}\left[x_i\right] \\ \vdots \\ \mathbb{E}_{X_D}\left[x_D\right] \end{bmatrix} \in \mathbb{R}^D \tag{6.30}$$

where

$$E_{X_d}[x_d] := \begin{cases} \sum_{x_i\in\mathcal{X}} x_i p(x_d = x_i), & \text{if } X \text{ is a discrete random variable} \\ \int_{\mathcal{X}} x_d p(x_d)\,dx_d, & \text{if } X \text{ is a continuous random variable} \end{cases} \tag{6.32}$$

for $d = 1, \cdots, D$, where the subscript d indicates the corresponding dimension of x. The integral and sum are over the states $\mathcal{X}$of the target space of the random variable X .

Median he median is the "middle" value if we sort the values, i.e., 50% of the values are greater than the median and 50% are smaller than the median. For continuous Random Variable $cdf = 0.5$. the median is more robust to outliers than the mean. #### Mode the value of x having the highest frequency of occurrence. For a continuous random variable, the mode is defined as a peak in the density p(x). A particular density p(x) may have more than one mode, and furthermore there may be a very large number of modes in high-dimensional distributions. #### Exemple (see p188) - **Remark**: The *expected value* is a linear operator. #### Covariance (Univariate) : he covariance between two univariate random variables $X, Y \in \mathbb{R}$ is given by the expected product of their deviations from their respective means, i.e.

$$Cov_{X,Y}[x, y] := \mathbb{E}_{X,Y}[(x - \mathbb{E}_X[x])(y - \mathbb{E}_Y[y]] \tag{6.35}$$

The covariance intuitively represents the notion of how dependent random variables are to one another. **Remark** : - For multivariate random variables, $Cov[x, y]$ is called *cross-covariance* with covariance referring to $Cov[x, x]$. - When the random variable associated with the expectation or covariance is clear by its arguments, the subscript is often suppressed (for example, $E_X[x]$ is often written as $E[x]$). - By using the linearity of expectations, we get

$$Cov_{X,Y}[x, y] := \mathbb{E}[xy] - \mathbb{E}[x]\mathbb{E}[y] \tag{6.36}$$

- **variance**: The covariance of a variable with itself $\$\mathrm{Cov}[x, x]\$ - **Standard deviation** : $The square root of the variance, \sigma(x)$ #### Covariance (Multivariate) The notion of covariance generalized to multivariate random variables. If we consider two multivariate random variables X and Y with states $x \in \mathbb{R}^D$ and $yy \in \mathbb{R}^E$ respectively, the covariance between X and Y is defined as

$$Cov[x, y] := \mathbb{E}\left[xy^T\right] - \mathbb{E}[x]\mathbb{E}[y]^T \in \mathbb{R}^{D \mathrm{x} E} \tag{6.37}$$

For a multivariate random variable, the variance describes the relation between individual dimensions of the random variable.

Variance The variance of a random variable X with states $x \in \mathbb{R}^D$ and a mean vector $\mu \in \mathbb{R}^D$ is defined as

$$\begin{aligned} V_X[x] &= \mathrm{Cov}_X[x, x] \\ &= \mathbb{E}_X\left[(x - \mu)(x - \mu)^\top\right] \\ &= \mathbb{E}_X[xx^\top] - \mathbb{E}_X[x]\mathbb{E}_X[x]^\top \\ &= \begin{bmatrix} \mathrm{Cov}[x_1, x_1] & \mathrm{Cov}[x_1, x_2] & \cdots & \mathrm{Cov}[x_1, x_D] \\ \mathrm{Cov}[x_2, x_1] & \mathrm{Cov}[x_2, x_2] & \cdots & \mathrm{Cov}[x_2, x_D] \\ \vdots & \vdots & \ddots & \vdots \\ \mathrm{Cov}[x_D, x_1] & \mathrm{Cov}[x_D, x_2] & \cdots & \mathrm{Cov}[x_D, x_D] \end{bmatrix} \end{aligned}$$

The $D \times D$ matrix in (6.38c) is called the covariance matrix of the multivariate random variable X . it's a symmetric and positive semidefinite matrix and tells us something about the spread of the data. The variances of the marginals is in its diagonals :

$$p(x_i) = \int p(x_i, \cdots, x_D)\, dx_{\setminus i},$$

cross-covariance : The off-diagonal entries $Cov[x_i, x_j]$ for $i, j = 1, \cdots, D, i \neq j$

Correlation The correlation between two random variables X, Y is given by

$$corr\,[x,y] = \frac{Cov\,[x,y]}{\sqrt{\mathbb{V}\,[x]\,\mathbb{V}\,[x]}} \in [-1,1]$$

The correlation matrix is the covariance matrix of standardized random variables, $x/\sigma(x)$.

The covariance (and correlation) indicate how two random variables are related

2.4.1 6.4.2 Empirical Means and Covariances

In machine learning, we need to learn from empirical observations of data. X: a random variable Two steps: - with finite dataset (of size N), we can construct an empirical statistic that is a function of a finite number of identical random variables, $X_1, \cdots, X_N$. - we observe the data, that is, we look at the realization $x_1, \cdots, x_N$ of each of the random variables and apply the empirical statistic.

empirical mean or *sample mean*: an estimate of the mean (based on the mean of a particular dataset)

Empirical Mean and Covariance The empirical mean vector is the arithmetic average of the observations for each variable, and it is defined as

$$\bar{x} := \frac{1}{N}\sum_{n=1}^{N} x_n$$

where $x_n \in \mathbb{R}^D$

Similar to the empirical mean, the empirical covariance matrix is a $D \times D$ matrix

$$\Sigma := \frac{1}{N}\sum_{n=1}^{N} (x_n - \bar{x})\,(x_n - \bar{x})^\top, \quad x_n \in \mathbb{R}^D \tag{6.42}$$

Empirical covariance matrices are symmetric, positive semidefinite. Throughout the book, we use the empirical covariance, which is a biased estimate. The unbiased (sometimes called corrected) covariance has the factor N − 1 in the denominator instead of N . ### 6.4.3 Three Expressions for the Variance We now focus on a single random variable X The standard definition of variance, corresponding to the definition of covariance, is the expectation of the squared deviation of a random variable X from its expected value μ, i.e.,

$$V_X[x] := \mathbb{E}_X\left[(x-\mu)^2\right] \tag{6.43}$$

The expectation in (6.43) and the mean $\mu = \mathbb{E}_X\,[(x)]$ are computed using (6.32). The variance as expressed in (6.43) is the mean of a new random variable $Z := (X-\mu)^2$. Two-pass algorithm for estimating the variance in (6.43). - one pass through the data to calculate the mean μ using (6.41) - a second pass using this estimate $\hat{\mu}$ calculate the variance.

Raw-score formula for variance : "the mean of the square minus the square of the mean"

$$V_X[x] = \mathbb{E}_X[x^2] - \left(\mathbb{E}_X[x]\right)^2 \tag{6.44}$$

If the two terms in (6.44) are huge and approximately equal, we may suffer from an unnecessary loss of numerical precision in floating-point arithmetic.

The raw-score version of the variance can be useful in machine learning, e.g., when deriving the bias–variance decomposition (Bishop, Christopher M. 2006. Pattern Recognition and Machine Learning. Springer.).

A third way to understand the variance is that it is a sum of pairwise differences between all pairs of observations. Consider a sample $x_1, \cdot, x_N$ of realizations of random variable X and we compute the squared difference between pairs of xi and xj .

$$\frac{1}{N^2}\sum_{i,j=1}^{N}(x_i - x_j)^2 = 2\left[\frac{1}{N}\sum_{i=1}^{N}x_i^2 - \left(\frac{1}{N}\sum_{i=1}^{N}x_i\right)^2\right]. \tag{6.45}$$

We see that (6.45) is twice the raw-score expression (6.44). This means that we can express the sum of pairwise distances (of which there are N^2 of them) as a sum of deviations from the mean (of which there are N).

2.4.2 6.4.4 Sums and Transformations of Random Variables

Consider two random variables X, Y with states $x, y \in \mathbb{R}^D$. Then:

$$\mathbb{E}[x+y] = \mathbb{E}[x] + \mathbb{E}[y] \tag{6.46}$$

$$\mathbb{E}[x-y] = \mathbb{E}[x] - \mathbb{E}[y] \tag{6.47}$$

$$\mathbb{V}[x+y] = \mathbb{V}[x] + \mathbb{V}[y] + Cov[x,y] + Cov[y,x] \tag{6.48}$$

$$\mathbb{V}[x-y] = \mathbb{V}[x] + \mathbb{V}[y] - +Cov[x,y] - Cov[y,x] \tag{6.49}$$

Mean and (co)variance exhibit some useful properties when it comes to affine transformation of random variables.Consider a random variable X with mean μ and covariance matrix Σ and a (deterministic) affine transformation $y = Ax + b$ of x. Then y is a also a random variable whose mean vector and covariance matrix are given by

$$\mathbb{E}_{\mathcal{Y}}[y] = \mathbb{E}_{\mathcal{X}}[Ax+b] = A\mathbb{E}_{\mathcal{X}}[x] + b = A\mu + b \tag{6.50}$$

$$\mathbb{V}_{\mathcal{Y}}[y] = \mathbb{V}_{\mathcal{X}}[Ax+b] = \mathbb{V}_{\mathcal{X}}[Ax] = A\mathbb{V}_{\mathcal{X}}[x]A^T = A\Sigma A^T$$

respectiveley. Futhermore

$$\begin{aligned}
\mathrm{Cov}[x,\, Ax+b] &= \mathbb{E}\Big[x(Ax+b)^\top\Big] - \mathbb{E}[x]\,\mathbb{E}[Ax+b]^\top \\
&= \mathbb{E}[x]b^\top + \mathbb{E}[xx^\top]A^\top - \mu b^\top - \mu\mu^\top A^\top \\
&= \mu b^\top - \mu b^\top + \mathbb{E}[xx^\top]A^\top - \mu\mu^\top A^\top \\
&= \big(\mathbb{E}[xx^\top] - \mu\mu^\top\big)A^\top \\
&= \Sigma A^\top
\end{aligned}$$

Where $\Sigma = \mathbb{E}[xx^\top] - \mu\mu^\top$ is the covariance of X. ### 6.4.5 Statistical Independence #### (Independence) Two random variables X, Y are statistically independent if and only if

$$p(x, y) = p(x)\, p(y). \tag{6.53}$$

1. $p(y \mid x) = p(y)$
2. $p(x \mid y) = p(x)$
3. $V_{X,Y}[x + y] = V_X[x] + V_Y[y]$
4. $\mathrm{Cov}_{X,Y}[x, y] = 0$

The last point may not hold in converse, i.e., two random variables can have covariance zero but are not statistically independent. #### Exemple 6.5 (see p194)

Independent and identically distributed (i.i.d.) random variables $X_1, \cdots, X_N$ modeling problem in ML like this. For more than two random variables, the word "**independent**" usually refers to mutually independent random variables, where all subsets are independent The phrase "**identically distributed**" means that all the random variables are from the same distribution.

Conditional Independence Two random variables X and Y are conditionally independent given Z (noted $X \perp\!\!\!\perp Y \mid Z$) if and only if

$$p(x, y \mid z) = p(x \mid z)\, p(y \mid z), \quad \forall z \in \mathcal{Z} \tag{6.55}$$

where $\mathcal{Z}$ is the set of states of the random variable Z. This is his expansion:

$$p(x, y \mid z) = p(x \mid y, z)\, p(y \mid z) \tag{6.56}$$

From (6.55) with (6.56), we get :

$$p(x \mid y, z) = p(x \mid z) \tag{6.57}$$

This means : "given that we know z , knowledge about y does not change our knowledge of x"

2.4.3 6.4.6 Inner Products of Random Variables

If we have two uncorrelated random variables X, Y , then

$$\mathbb{V}[x + y] = \mathbb{V}[x] + \mathbb{V}[y] \tag{6.58}$$

Random variables can be considered vectors in a vector space, and we can define inner products to obtain geometric properties of random variables. Be the inner product where zero mean random variables X and Y:

$$\langle X, Y \rangle := \mathrm{Cov}[x, y] \tag{6.59}$$

We see that the covariance is symmetric, positive definite, and linear in either argument. The length of a random variable is

$$\|X\| = \sqrt{\mathrm{Cov}[x, x]} = \sqrt{V[x]} = \sigma[x] \tag{6.60}$$

(The standard deviation) The "longer" the random variable, the more uncertain it is; and a random variable with length 0 is deterministic.

Angle θ between two random variables X, Y:

$$\cos\theta = \frac{\langle X, Y\rangle}{\|X\|\,\|Y\|} = \frac{\mathrm{Cov}[x, y]}{\sqrt{V[x]}\,\sqrt{V[y]}}$$

(the correlation between the two random varibles)

$$X \perp Y \iff \langle X, Y\rangle = 0 \iff \mathrm{Cov}[x, y] = 0 \iff \text{they are uncorrelated}$$

Remark: the euclidean distance is not the best way to obtain distances between distributions (see P 197 for a statistical manifold or information geometry). It is done using Kullback-Leibler divergence, hich is a generalization of distances that account for properties of the statistical manifold. To see more about that, look up for : *Amari, Shun-ichi. 2016. Information Geometry and Its Applications. Springer.*

2.5 6.5. Gaussian Distribution

the most well-studied probability distribution for continuous-valued random variables. *Normal Distribution.*

The Gaussian distribution arises naturally when we consider sums of independent and identically distributed random variables. This is known as the **central limit theorem** (see Grinstead, Charles M., and Snell, J.Laurie. 1997. Introduction to Probability. American Mathematical Society.)

widely used in statistical estimation and machine learning as they have closed-form expressions for marginal and conditional distributions. ML areas that profit frmo Gaussian distribution: Gaussian processes, variational inference, and reinforcement learning. also in signal processing, contrl, and statistics

- For a univariate random variable, the Gaussian distribution has a density that is given by

$$p(x \mid \mu, \sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(x-\mu)^2}{2\sigma^2} \right) \tag{6.62}$$

  The *multivariate Gaussian distribution* is fully characterized by a *mean vector* μ and a *covariance matrix* σ and defined as

$$p(x \mid \mu, \Sigma) = \frac{1}{(2\pi)^{-D/2}|\Sigma|^{-1/2}} \exp\left(-\frac{1}{2}(x-\mu)^\top \Sigma^{-1}(x-\mu) \right)$$

  where $x \in \mathbb{R}^D$. We write $p(x) = \mathcal{N}(x \mid \mu, \Sigma$ or $X \sim \mathcal{N}(\mu, \Sigma)$
- *the standard normal distribution* : $\mu = 0$ and $\Sigma = I$
- When modeling with Gaussian random variables, variable transformations (Section 6.7) are often not needed. Since the Gaussian distribution is fully specified by its mean and covariance, we often can obtain the transformed distribution by applying the transformation to the mean and covariance of the random variable.

2.5.1 6.5.1 Marginals and Conditionals of Gaussians are Gaussians

(in the general case of multivariate random variables) Let X and Y be two multivariate random variables, that may have different dimensions. The Gaussian distribution in terms of the concatenated states $[x^\top\ y^\top]^\top$ so that

$$p(x,y) = \mathcal{N}\left(\begin{bmatrix}\mu_x \\ \mu_y\end{bmatrix}, \begin{bmatrix}\Sigma_{xx} & \Sigma_{xy} \\ \Sigma_{yx} & \Sigma_{yy}\end{bmatrix} \right) \tag{6.64}$$

Where $\Sigma_{xx} = \mathrm{Cov}[x,x]$ and $\Sigma_{yy} = \mathrm{Cov}[y,y]$ are the marginal covariance matrices of x and y , respectively, and $\Sigma_{xy} = \mathrm{Cov}[x,y]$ is the cross-covariance matrix between x and y . The conditional distribution p(x | y) is also Gaussian and given by

$$p(x \mid y) = \mathcal{N}(\mu_{x|y}, \Sigma_{x|y}) \tag{6.65}$$

$$\mu_{x|y} = \mu_x + \Sigma_{xy}\,\Sigma_{yy}^{-1}(y - \mu_y) \tag{6.66}$$

$$\Sigma_{x|y} = \Sigma_{xx} - \Sigma_{xy}\,\Sigma_{yy}^{-1}\Sigma_{yx} \tag{6.67}$$

in (6.66), the y-value is an observation and no longer random. **Remark**: il find it in : - The Kalman filter - Gaussian processes (Rasmussen and Williams, 2006) - Latent linear Gaussian models (Roweis and Ghahramani, 1999; Murphy, 2012)

The marginal distribution $p(x)$ of a joint Gaussian distribution $p(x,y)$ is itself Gaussian

$$p(x) = \int p(x,y)\,dy = \mathcal{N}(x \mid \mu_x, \Sigma_{xx}) \tag{6.68}$$

The corresponding result holds for p(y) **Exemple (on p200)**

2.5.2 6.5.2 Product of Gaussian Densities

The product of two Gaussians $\mathcal{N}(x \mid a, A)\mathcal{N}(x \mid b, B)$ is a Gaussian distribution scaled by a $c \in \mathbb{R}$, given by $c\mathcal{N}(x \mid c, C)$ with

$$C = (A^{-1} + B^{-1})^{-}1 \tag{6.74}$$

$$c = C(A^{-1}a + B^{-1}b) \tag{6.75}$$

$$c = (2\pi)^{-\frac{D}{2}}|A+B|^{-1/2}\exp\left[-\frac{1}{2}(a-b)^\top(A+B)^{-1}(a-b)\right] \tag{6.76}$$

The scaling constant c itself can be written in the form of a Gaussian density either in a or in b with an "inflated"covariance matrix $A+B$, i.e., $c = \mathcal{N}(a|b, A+B) = \mathcal{N}(b|a, A+B)$

Remark. For notation convenience, we will sometimes use $\mathcal{N}(x|m,S)$ to describe the functional form of a Gaussian density even if x is not a random variable.

2.5.3 6.5.3 Sums and Linear Transformations

$p(x) = \mathcal{N}(x|\mu_x, \Sigma_x)$ and $p(y) = \mathcal{N}(y|\mu_y, \Sigma_y)$

X, Y are independent Gaussian random variables $\iff p(x, y) = p(x)p(y) \iff x+y$ is also Gaussian distributed and given by

$$p(x + y) = \mathcal{N}(\mu_x + \mu_y, \Sigma_x + \Sigma_y)$$

Knowing that p(x + y) is Gaussian, the mean and covariance matrix can be determined immediately using the results from (6.46) through (6.49). This notion is imoprt for *Gaussian noise* acting on random variables.

Exemple 6.7 (p201) Remark: **Weighted sum of Gaussian densities** is very import for Chapter 11. **Theorem 6.12**. *Consider a mixture of two univariate Gaussian densities*

$$p(x) = \alpha p_1(x) + (1 - \alpha)p_2(x) \tag{6.80}$$

where the scalar \$ $0 < \alpha < 1$\$*is the mixture weight, and*$p_1(x)$ and $p_2(x)$ *are univariate Gaussian densities (Equation (6.62)) with different parameters, i.e.,* $(\mu_1, \sigma_1^2) \neq (\mu_2, \sigma_2^2)$.

the random variable x is from a density that is a mixture of two densities $p_1(x)$ and $p_2(x)$, weighted by α. *Then the mean of the mixture density* $p(x)$ *is given by the weighted sum of the means of each random variable:*

$$\mathbb{E}[x] = \alpha\mu_1 + (1 - \alpha)\mu_2 \tag{6.81}$$

The variance of the mixture density $p(x)$ *is given by*

$$\mathbb{V}[x] = \alpha\sigma_1^2 + (1 - \alpha)\sigma_2^2 + \alpha\mu_1^2 + (1 - \alpha)\mu_2^2 - \left[\alpha\mu_1 + (1 - \alpha)\mu_2\right]^2 \tag{6.82}$$

Proof : See **p202**

Remark: The preceding derivation holds for any density, but since the Gaussian is fully determined by the mean and variance, the mixture density can be determined in closed form.

Law of total variance (conditional variance formula) : generally states that for two random variables X and Y it holds that $\mathbb{V}[X] = \mathbb{E}_Y\big[\mathbb{V}[X \mid Y]\big] + \mathbb{V}_Y\big[\mathbb{E}[X \mid Y]\big]$ i.e., the (total) variance of X is the expected conditional variance plus the variance of a conditional mean. #### For linear transformation Any linear/affine transformation A of a Gaussian random variable x is also Gaussian distributed. The outcome is a Gaussian random variable with mean zero and covariance $AA^\top$.

$\implies$ the random variable x + μ is Gaussian with mean μ and identity covariance.

For $X \sim \mathcal{N}(\mu, \Sigma)$, a given matrix A and Y a random variable such that $y = Ax$, we got : - *The mean* of y: $\mathbb{E}[y] = \mathbb{E}[Ax] = A\mathbb{E}[x] = A\mu$

- *The variance* of y : $\mathbb{V}[y] = \mathbb{V}[Ax] = A\mathbb{V}[x]A^\top = A\Sigma A^\top$
- This means that the random variable y is distributed according to : $p(y) = \mathcal{N}\big(y \mid A\mu,\ A\Sigma A^\top\big)$

Reverse transformation: when we know that a random variable has a mean that is a linear transformation of another random variable. $A \in \mathbb{R}^{MxN}$: A full rank matrix where $M \geq N$ $y \in \mathbb{R}^M$: a *Gaussian random variable* with mean \$AX), i.e,

$$p(y) = \mathcal{N}(y \mid Ax, \Sigma) \tag{6.89}$$

$\Longrightarrow$ he corresponding probability distribution $p(x)$:

$$y = AX \Longleftrightarrow (A^\top A)^{-1} A^\top y = x \tag{6.90}$$

Hence, x is a linear transformation of y , and we obtain :

$$p(x) = \mathcal{N}\Big(x \;\Big|\; (A^\top A)^{-1} A^\top y,\ (A^\top A)^{-1} A^\top \Sigma A (A^\top A)^{-1}\Big) \tag{6.91}$$

2.5.4 6.5.4 Sampling from Multivariate Gaussian Distributions (p204)

In the case of a multivariate Gaussian, this process consists of three stages: 1. we need a source of pseudo-random numbers that provide a uniform sample in the interval $[0, 1]$ 2. we use a non-linear transformation such as the Box-Müller transform (Devroye, 1986) to obtain a sample from a univariate Gaussian 3. we collate a vector of thesesamples to obtain a sample from a multivariate standard normal $\mathcal{N}(0, I)$. general multivariate Gaussian $\Longrightarrow$ mean is non zero and the covariance is not the identity matrix.

To obtain samples from a multivariate normal $\mathcal{N}(\mu, \Sigma)$, we can use the properties ofa linear transformation of a Gaussian random variable: If $x \sim \mathcal{N}(0, I)$, then $y = Ax + \mu$, where $AA^\top = \Sigma$ is Gaussian distributed with mean μ and covariance matrix Σ. One convenient choice of A is to use the Cholesky decomposition of the covariance matrix $\Sigma = AA^\top$.

2.6 6.6. Conjugacy and the Exponential Family

- "named" probability distributions: Probability distributions that are used to model particular types of phenoma
- Each model aer related to each other in complex ways (Leemis, Lawrence M., and McQueston, Jacquelyn T. 2008. Univariate Distribution Relationships. American Statistician, 62(1), 45–53.
- Efron, Bradley, and Hastie, Trevor. 2016. Computer Age Statistical Inference: Algorithms, Evidence and Data Science. Cambridge University Press.
- desiderata for manipulating probability distributions in ML:

1. "closure property" when applying the rules of probability
2. As we collect more data, we do not need more parameters to describe the distribution.
3. Since we are interested in learning from data, we want parameter estimation to behave nicely.

- **exponential family** : provides the right balance of generality while retaining favorable computation and inference properties.
- **Exemple 6.8** (See p205) where it explains the Bernouilli and Binomial and Beta distributions

The Bernoulli distribution $\text{Ber}(\mu)$ is defined as

$$p(x \mid \mu) = \mu^x (1 - \mu)^{1-x}, \quad x \in \{0, 1\}$$

with expectation and variance

$$\mathbb{E}[x] = \mu, \qquad \mathbb{V}[x] = \mu(1 - \mu).$$

where $\mathbb{E}[x]$ and $\mathbb{V}[x]$ are the mean and variance of the binary random variable X. **Remark** : the Bernouilli distribution is sometimes express in the exponents in machine learning textbooks (a trick)

The Binomial distribution $\mathrm{Bin}(N, \mu)$ is defined as

$$p(m \mid N, \mu) = \binom{N}{m} \mu^m (1-\mu)^{N-m}, \quad m \in \{0, 1, \ldots, N\}$$

with expectation and variance of m

$$\mathbb{E}[m] = N\mu, \qquad \mathrm{Var}[m] = N\mu(1-\mu).$$

The **Beta distribution** $\mathrm{Beta}(\alpha, \beta)$ is a distribution over a continuous random variable $\mu \in [0, 1]$. It is defined as

$$p(\mu \mid \alpha, \beta) = \frac{\Gamma(\alpha+\beta)}{\Gamma(\alpha)\,\Gamma(\beta)} \mu^{\alpha-1}(1-\mu)^{\beta-1}, \quad \alpha > 0,\, \beta > 0$$

where $\Gamma(\cdot)$ is the Gamma function:

$$\Gamma(t) = \int_0^\infty x^{t-1} e^{-x}\, dx, \qquad t > 0$$

with the recurrence property

$$\Gamma(t+1) = t\,\Gamma(t).$$

The expectation and variance of μ are

$$\mathbb{E}[\mu] = \frac{\alpha}{\alpha+\beta}, \qquad \mathbb{V}[\mu] = \frac{\alpha\beta}{(\alpha+\beta)^2(\alpha+\beta+1)}.$$

Intuitively:
- α déplace la masse de probabilité vers **1**
- β déplace la masse de probabilité vers **0**

Cas particuliers :
- $\alpha = 1,\ \beta = 1 \ \Rightarrow\ U[0,1]$ (uniforme)
- $\alpha, \beta < 1 \ \Rightarrow$ bimodale (pics en 0 et 1)
- $\alpha, \beta > 1 \ \Rightarrow$ unimodale
- $\alpha = \beta > 1 \ \Rightarrow$ unimodale, symétrique, centrée en $[0, 1]$

see Leemis, Lawrence M., and McQueston, Jacquelyn T. 2008. Univariate Distribution Relationships. American Statistician, 62(1), 45–53.

Each distribution is created for particular reason, that reason may be considered when choosing distribution

2.6.1 6.6.1 Conjugacy (p208)

Conjugate Prior A prior is conjugate for the likelihood function if the posterior is of the same form/type as the prior.

In **Conjugacy**, we can algebraically calculate our posterior distribution by updating the parameters of the prior distribution.

Remark : When considering the geometry of probability distributions, conjugate priors retain the same distance structure as the likelihood Agarwal, Arvind, and Daumé III, Hal. 2010. A Geometric View of Conjugate Priors. Machine Learning, 81(1), 99–113.

Example 6.11 (Beta-Binomial Conjugacy) : See p208 **Example 6.12** (Beta-Bernoulli Conjugacy)

The Gamma prior is conjugate for the precision (inverse variance) in the univariate Gaussian likelihood, and the Wishart prior is conjugate for the precision matrix (inverse covariance matrix) in the multivariate Gaussian likelihood.

2.6.2 6.6.2 Sufficient Statistics

sufficient statistics:the idea that there are statistics that will contain all available information that can be inferred from data corresponding to the distribution under consideration. (carry all the information needed to make inference about the population as a representation of the distribution)

Set of distributions parametrized by θ. X: Random variable with distribution $p(x \mid \theta_0)$ given an unknown θ_0.

A vector $\phi(x)$ of statistics is called **sufficient statistics** for θ_0 if it contains all possible information about θ_0.

"contain all possible information" $\Longleftrightarrow$ the probability of x given θ can be factored into a part that does not depend on θ, and a part that depends on θ only via ϕ(x).

Fisher-Neyman (Theorem 6.5 in Lehmann, Erich Leo, and Casella, George. 1998. Theory of Point Estimation. Springer.) Let X have probability density function $p(x \mid \theta)$. The statistics $\phi(x)$ are **sufficient** for θ if and only if $p(x \mid \theta)$ can be written in the form

$$p(x \mid \theta) = h(x)\, g_\theta(\phi(x)), \tag{6.106}$$

where $h(x)$ is a distribution independent of θ and g_θ captures all dependence on θ via the sufficient statistics $\phi(x)$.

If $p(x \mid \theta)$ does not depend on θ, then $\phi(x)$ is trivially a sufficient statistic for any function ϕ. The more interesting case is that $p(x \mid \theta)$ is dependent only on $\phi(x)$ and not x itself. In this case, $\phi(x)$ is a sufficient statistic for θ.

In machine learning, we consider a finite number of samples from a distribution. (see Wasserman, Larry. 2007. All of Nonparametric Statistics. Springer.)

2.6.3 6.6.3 Exponential Family

Possible level when considering distributions (of continuous or discrete random variables) 1. we have a particular named distribution with fixed parameters, for example a univariate Gaussian N

0, 1 with zero mean and unit variance. 2. we fix the parametric form (the univariate Gaussian) and infer the parameters from data. For example, we assume a univariate Gaussian $\mathcal{N}(\mu, \sigma^2)$ with unknown mean μ and unknown variance σ^2 , and use a maximum likelihood fit to determine the best parameters (μ, σ^2) 3. to consider families of distributions, and in this book, we consider the exponential family. The univariate Gaussian is an example of a member of the exponential family.

Definition An **exponential family** is a family of probability distributions, parameterized by $\theta \in \mathbb{R}^D$, of the form

$$p(x \mid \theta) = h(x) \exp\Big(\langle \theta, \phi(x) \rangle - A(\theta)\Big), \tag{6.107}$$

where $\phi(x)$ is the vector of **sufficient statistics**. In general, any inner product (Section 3.2) can be used in the exponential family formula.
For concreteness, we use the standard dot product here: $\langle \theta, \phi(x) \rangle = \theta^\top \phi(x)$.

essentially a particular expression of $g_\theta(\phi(x))$in the Fisher-Neyman theorem (Theorem 6.14).

The log-partition function : result of the sum of the distributions when it is normalized with constant the terme $A(\theta)$

Exponential families can be considered as distributions of the form

$$p(x \mid \theta) \propto \exp\big(\theta^\top \phi(x)\big). \tag{6.108}$$

θ : The *natural parameters* We can transform (6.108) for convenient modeling and efficient computation based on the fact that we can capture information about data in ϕ(x). #### Example 6.13 (Gaussian as Exponential Family) p212 The **univariate Gaussian distribution** is a member of the exponential family with sufficient statistic

$$\phi(x) = \begin{bmatrix} x \\ x^2 \end{bmatrix},$$

and natural parameters

$$\theta = \begin{bmatrix} \frac{\mu}{\sigma^2} \\ -\frac{1}{2\sigma^2} \end{bmatrix}.$$

Example 6.14 (Bernoulli as Exponential Family) The **Bernoulli distribution** can be written in **exponential family** form:

- Sufficient statistic: $\phi(x) = x$
- Natural parameter: $\theta = \log \frac{\mu}{1-\mu}$
- Probability: $p(x \mid \theta) = \exp(\theta x - A(\theta))$
  where $A(\theta) = \log(1 + e^\theta)$.

- The relationship between θ and μ is invertible so that

$$\mu = \frac{1}{1 + exp(-\theta)}$$

- **Remark**: The relationship between the original Bernoulli parameter μ and the natural parameter θ is known as the *sigmoid* or logistic function. The sigmoid function squeezes a real value into the range $(0, 1)$.

Exponential families provide a convenient way to find conjugate pairs of distributions. Consider a random variable X in the **exponential family**:

$$p(x \mid \theta) = h(x) \exp\big(\langle \theta, \phi(x) \rangle - A(\theta)\big).$$

Every member of the exponential family has a **conjugate prior** of the form:

$$p(\theta \mid \gamma) = h_c(\theta) \exp\Big(\langle \begin{bmatrix} \gamma_1 \\ \gamma_2 \end{bmatrix}, \begin{bmatrix} \theta \\ -A(\theta) \end{bmatrix} \rangle - A_c(\gamma)\Big),$$

where $\gamma = \begin{bmatrix} \gamma_1 \\ \gamma_2 \end{bmatrix}$ has dimension $\dim(\theta) + 1$, and the sufficient statistics of the conjugate prior are \$ $\begin{bmatrix} \theta \\ -A(\theta) \end{bmatrix}$ \$.

As mentioned in the previous section, the main motivation for exponential families is that they have finite-dimensional sufficient statistics. Additionally, conjugate distributions are easy to write down, and the conjugate distributions also come from an exponential family.

2.7 6.7 Change of Variables/Inverse Transform

We have a few set of distributions even though we have a lot of type of distributions $\Longrightarrow$ how transformed random variables are distributed. **ex** : X, a random variable distributed according to the univariate normal distribution $\mathcal{N}(0, , 1)$, what is the distribution of X^2? or $\frac{1}{2}(X_1 + X_2)$ for X_1 and X_2.

One option to work out the distribution of $\frac{1}{2}(X_1 + X_2)$ is to calculate the mean and variance of X1 and X2 and then combine them.

Notation remark capital lette X, Y : random variables, small letter x, y : the values in the target space $\mathcal{T}$ that the random variables take. pmfs of discrete random variables X as $P(X = x)$. pdf is written as $f(x)$ for continous random variables X and $F_X(x)$ is the cdf.

Two approaches for obtaining distributions of transformations of random variables : - direct approach using the definition of a cumulative distribution function - change-of-variable(very widely used) approach that uses the chain rule of calculus (Section 5.2.2)

X with pmf $P(X = x)$, $U(x)$ an invertible function. The transformed random variable \$ Y := U(X)\$, with pmf $P(Y = y)$, then - *Transformation of interest (6.125a)*: $P(Y = y) = P(U(X) = y)$ - *Inverse (6.125b* : \$P(Y = y) = P(X = U^-1(y)) \$

Where we can observe that $x = U^{-}1(y)$. Therefore, for discrete random variables, transformations directly change the individual events (with the probabilities appropriately transformed).

Moment generating functions can also be used to study transformations of random variable see Casella, George, and Berger, Roger L. 2002. Statistical Inference. Duxbury.

2.7.1 6.7.1 Distribution Function Technique

uses cdf $F_X(x) = P(X \leq x)$. Its differential is the pdf $f(x)$. X Random variable, U a function, the pdf of the random variable $Y := U(X)$ is found by: 1. The **cumulative distribution function (CDF)**:

$$F_Y(y) = P(Y \leq y) \tag{6.126}$$

2. The **probability density function (PDF)** is obtained by differentiating the CDF:

$$f(y) = \frac{d}{dy} F_Y(y) \tag{6.127}$$

We also need to keep in mind that the domain of the random variable may have changed due to the transformation by U #### Example 6.16 : (p216) we considered a strictly monotonically increasing function $f(x) = 3x^2$ #### Theorem 6.15 *Let X be a continuous random variable with a strictly monotonic cumulative distribution function $F_X(x)$. Then the random variable Y defined as*

$$Y := F_X(X) \tag{6.132}$$

has a uniform distribution.

Theorem 6.15 is known as the *probability integral transform*, and it is used to derive algorithms for sampling from distributions by transforming the result of sampling from a uniform random variable (Bishop, 2006).

2.7.2 6.7.2 Change of Variables

The distribution function technique in Section 6.7.1 is derived from first principles, based on the definitions of cdfs and using properties of inverses, differentiation, and integration. This argument from first principles relies on two facts: 1. We can transform the cdf of Y into an expression that is a cdf of X 2. We can differentiate the cdf to obtain the pdf.

For univariate functions, we use the **substitution rule of integration**:

$$\int f(g(x))\, g'(x)\, dx = \int f(u)\, du, \qquad \text{where } u = g(x). \tag{6.133}$$

(based on the chain rule of calculus) #### Substitution Rule of Integration

The rule is derived from the **chain rule** of calculus and the **fundamental theorem of calculus**.

- **Fundamental theorem of calculus**:
  Differentiation and integration are (formal) inverses of each other.

- **Differential intuition**:
  If $u = g(x)$, then

$$\Delta u = g'(x)\,\Delta x \qquad \Rightarrow \qquad du \approx g'(x)\,dx.$$

- **Substitution formula**:

$$\int f(g(x))\,g'(x)\,dx = \int f(u)\,du.$$

X : random variable, U : an *invertible* function, $ Y = U(X)$, X has states $x \in [a, b]$

$$P(Y \leq y) = P(U(X) \leq); \tag{6.135}$$

$$P\big(U(X) \leq y\big) = P\big(U^{-1}(U(X)) \leq U^{-1}(y)\big) = P\big(X \leq U^{-1}(y)\big).$$

$$P\big(X \leq U^{-1}(y)\big) = \int_a^{U^{-1}(y)} f(x)\,dx. \tag{6.137}$$

$$F_Y(y) = \int_a^{U^{-1}(y)} f(x)\,dx. \tag{6.138}$$

We obtain the pdf

$$f(y) = \frac{d}{dy}F_Y(y) = \frac{d}{dy}\int_a^{U^{-1}(y)} f(x)\,dx \tag{6.139}$$

We are going to represent the formules with respect to dy

$$\int f\big(U^{-1}(y)\big)\,U^{-1'}(y)\,dy = \int f(x)\,dx, \qquad x = U^{-1}(y). \tag{6.140}$$

$$f(y) = f_x\mu(U^{-1}(y)) \cdot \frac{d}{dy}U^{-1}(y). \tag{6.142}$$

Recall that we assumed that U is a strictly increasing function. For decreasing functions:

$$f_Y(y) = f_X(U^{-1}(y)) \cdot \left|\frac{d}{dy}U^{-1}(y)\right| \tag{6.143}$$

Change-of-variable technique : The term $f_Y(y) = f_X(U^{-1}(y)) \cdot \left|\frac{d}{dy}U^{-1}(y)\right|$ in (6.143) measures how much a unit volume changes when applying U.

Remark :Compared to the discrete case, the continuous case requires the additional factor $\frac{d}{dy}U^{-1}(y)$ because $P(Y = y) = 0$ for all y. The **probability density function** is then $f_Y(y) = f_X(U^{-1}(y)) \cdot \left|\frac{d}{dy}U^{-1}(y)\right|$ and cannot be interpreted as the probability of an event involving y.

The determinant of the Jacobian matrix is used for multivariate random variablels.

Theorem 6.16 (Multivariate change of variable) Let $f_X(x)$ be the probability density of a multivariate continuous random variable X. If the vector-valued function $y = U(x)$ is differentiable and invertible for all x in its domain, then the density of $Y = U(X)$ is

$$f(y) = f_x(U^{-1}(y)) \cdot \left| \det \left(\frac{\partial}{\partial y} U^{-1}(y) \right) \right|. \tag{6.144}$$

Exemple 6.17 : While Example 6.17 is based on a bivariate random variable, which allows us to easily compute the matrix inverse, the preceding relation holds for higher dimensions.

2.7.3 6.8 Further Reading

- **Introductory and self-study texts**: Grinstead and Snell (1997), Walpole et al. (2011).
- **Philosophical aspects of probability**: Hacking (2001).
- **Software-oriented approaches**: Downey (2014).
- **Exponential families**: Barndorff-Nielsen (2014).
- **Machine learning applications**:
  - Probabilistic modeling in ML tasks (Chapter 8).
  - Normalizing flows for transforming random variables (Jimenez Rezende and Mohamed, 2015).
  - Variational inference in neural networks (Goodfellow et al., 2016, Chapters 16–20).

Notes on continuous random variables:
- Measure-theoretic issues are avoided for simplicity (Billingsley, 1995; Pollard, 2002).
- Conditional probabilities for continuous variables require care: $p(y \mid x)$ where $X = x$ is a set of measure zero.
- More precise notation involves $\mathbb{E}_y[f(y) \mid \sigma(x)]$.

Advanced probability theory references: Jaynes (2003), MacKay (2003), Jacod and Protter (2004), Grimmett and Welsh (2014), Shiryayev (1984), Lehmann and Casella (1998), Dudley (2002), Bickel and Doksum (2006), Çinlar (2011).
- Alternative approach: start from expectation and derive probability space properties (Whittle, 2000).

Machine learning with probabilistic modeling: MacKay (2003); Bishop (2006); Rasmussen and Williams (2006); Barber (2012); Murphy (2012).

3 Chapter 7 : Continuous Optimization

Mathematical formulations are expressed as numerical optimization methods. The notion of "good" is determined by the objective function or the probabilistic model. Given an objective function, finding the best value is done using optimization algorithms.

Two branches of optimization here: unconstrained and constrained optimization. We assume that our objective function is differentiable Since we consider data and models in RD , the optimization problems we face are continuous optimization problems, as opposed to combinatorial optimization problems for discrete variables.

Global minimum

Local minimum

Stationary points are the real roots of the derivative, that is, points that have zero gradient. For

$$\ell(x) = x^4 + 7x^3 + 5x^2 - 17x + 3$$

we obtain the corresponding gradient as

$$\frac{d\ell(x)}{dx} = 4x^3 + 21x^2 + 10x - 17$$

To check whether a stationary point is a minimum or maximum, we need to take the derivative a second time and check whether the second derivative is positive or negative at the stationary point.

We start from a point and follow the negative gradient since negative gradient indicates that we should go right. According to the Abel–Ruffini theorem, there is ingeneral no algebraic solution for polynomials of degree 5 or more.

For convex functions all local minima are global minimum.

3.1 7.1 Optimization Using Gradient Descent

On cherche à résoudre $ \min _{x}f(x)$.$Where f : \mathbb{R}^d \to \mathbb{R}$. is an objective function that captures the machine learning problem at hand. We assume that , f is differentiable and no solution can be found analytically.

Gradient descent: a first-order optimization algorithm to find local minimum of a function. one takes steps proportional to the negative of the gradient of the function at the current point. the gradient points in the direction of the steepest ascent.

Contour lines: set of lines where the function is at a certain value ($f(x) = c$ for some value $c \in \mathbb{R}$).

The gradient points in a direction that is orthogonal to the contour lines of the function we wish to optimize.

Gradient descent exploits the fact that $f(x_0)$ decreases fastest if one moves from x_0 in the direction of the negative gradient $-(\nabla f)(x_0)^\top$ of f at x_0.

If

$$x_1 = x_0 - \gamma((\nabla f)(x_0))^\top \tag{7.5}$$

for a small step-size $\gamma \geq 0$, then $f(x_1) \leq f(x_0)$.

The algorithm: If we want to find a local optimum $f(x^*)$ of a function

$$f : \mathbb{R}^n \to \mathbb{R}, \quad x \mapsto f(x),$$

we start with an initial guess x_0 of the parameters we wish to optimize and then iterate according to

$$x_{i+1} = x_i - \gamma_i (\nabla f)(x_i)^\top \tag{7.6}$$

For suitable step-size γ_i, the sequence

$$f(x_0) \geq f(x_1) \geq \ldots \text{ converges to a local minimum.}$$

Example 7.1 shows an example with a quadratic function in two dimensions

Remark Gradient descent can be relatively slow close to the minimum: Its asymptotic rate of convergence is inferior to many other methods. ### 7.1.1 Step-size The step-size is also called the learning rate. It is important because it can determine how accurate we are, too small $\Longrightarrow$ slow, too large $\Longrightarrow$ overshoot, fail to converge or diverge.

Adaptive gradient methods : rescale the step-size at each iteration, depending on local properties of the function.

Two simple heuristics : - When the function value increases after a gradient step, the step-size was too large. Undo the step and decrease the step-size. - When the function value decreases the step could have been larger. Try to increase the step-size.

Example 7.2 (Solving a Linear Equation System) p230

Remark The **condition number** is defined as

$$\kappa = \frac{\sigma_{\max}(A)}{\sigma_{\min}(A)}$$

The ratio of the maximum to the minimum singular value of A. The condition number essentially measures the ratio of the most curved direction versus the least curved direction.

Instead of directly solving $Ax = b$, one could instead solve

$$P^{-1}(Ax - b) = 0,$$

where P is called the **preconditioner**.

The goal is to design P^{-1} such that $P^{-1}A$ has a better condition number,
but at the same time P^{-1} is easy to compute.

3.1.1 7.1.2 Gradient Descent With Momentum

(to give gradient descent some memory| a "batch" optimization method)

a method that introduces an additional term to remember what happened in the previous iteration.This memory dampens oscillations and smoothes out the gradient updates.

The momentum-based method remembers the update Δx_i at each iteration i and determines the next update as a linear combination of the current and previous gradients

$$x_{i+1} = x_i - \gamma_i((\nabla f)(x_i))^\top + \alpha \Delta x_i \tag{7.11}$$

$$\Delta x_i = x_i - x_{i-1} = \alpha \Delta x_{i-1} - \gamma_{i-1}(\nabla f(x_{i-1}))^\top \tag{7.12}$$

where $\alpha \in [0, 1]$.

The momentum term is useful since it averages out different noisy estimates of the gradient.

3.1.2 7.1.3 Stochastic Gradient Descent

(a "cheap" approximation of the gradient) a stochastic approximation of the gradient descent method for minimizing an objective function that is written as a sum of differentiable functions

Stochastic: we do not know the gradient precisely, but instead only know a noisy approximation to it.

In machine learning, given $n = 1, \ldots, N$ data points, we often consider objective functions that are the sum of the losses L_n incurred by each example n.
In mathematical notation, we have the form

$$L(\theta) = \sum_{n=1}^{N} L_n(\theta), \tag{7.13}$$

where θ is the vector of parameters of interest, i.e., we want to find θ that minimizes L.

An example at chapter 9 :

$$L(\theta) = -\sum_{n=1}^{N} \log p(y_n \mid x_n, \theta), \tag{7.14}$$

where $x_n \in \mathbb{R}^D$ are the training inputs, y_n are the training targets, and θ are the parameters of the regression model.

Optimization is performed using the full training set by updating the vector of parameters according to :

$$\theta_{i+1} = \theta_i - \gamma_i(\nabla L(\theta_i)) = \theta_i - \gamma_i \sum_{n=1}^{N} (\nabla L_n(\theta_i))^\top, \tag{7.15}$$

for a suitable step-size parameter γ_i.

- **Batch Gradient Descent**: uses all L_n $(n = 1, \ldots, N)$ to compute the gradient.
- **Mini-Batch Gradient Descent**: randomly selects a subset of L_n to estimate the gradient.

- **Stochastic Gradient Descent (SGD)**: extreme case, uses only one randomly chosen L_n.
- **Key Insight**: convergence only requires the gradient estimate to be **unbiased**.
- The sum $\sum_{n=1}^{N} \nabla L_n(\theta_i)$ is an **empirical estimate of the expected gradient**.
- Any **unbiased subsample** of the data can be used instead, reducing computation.
- **Reason for approximate gradients**: practical limits on CPU/GPU memory and computation time.
- **Large mini-batch**:
  - More accurate gradient estimates (low variance).
  - Stable convergence.
  - Efficient with optimized matrix operations.
  - But more expensive per update.
- **Small mini-batch**:
  - Faster updates.
  - Higher variance in gradient (adds noise).
  - Noise can help escape bad local minima.
- **Key trade-off**: accuracy vs. efficiency vs. ability to generalize.
- **Machine learning goal**: not exact minimization of training objective, but good **generalization performance**.
- **Widely used**: mini-batch SGD scales well to large problems (deep learning, topic models, reinforcement learning, Gaussian processes, etc.).

3.1.3 7.2 Constrained Optimization and Lagrange Multipliers

We have additional contraints, that is, (primal problem) - **Problem type**: Constrained optimization.
- **Objective**: minimize a function $ f(x) $.
- **Constraints**:
- Given by real-valued functions $ g_i: R^D \to \mathbb{R}$.
– $Condition : \$g\_i(x) \leq 0\$, for \$i = 1, \ldots, m\$$.
– **Interpretation :**
– $Feasible set = all \$x\$ that satisfy the constraints.$
– $Optimization is restricted to this feasible set.$
– **General form :**
$\min_x f(x) \quad \text{subject to} \quad g_i(x) \leq 0 \quad \forall i = 1, \ldots, m$

indicator function : Converting the contrained problem into an unconstrained one with

$$J(x) = f(x) + \sum_{i=1}^{m} 1\big(g_i(x)\big), \tag{7.18}$$

where the indicator function $1(z)$ is defined as

$$1(z) = \begin{cases} 0 & \text{if } z \leq 0 \\ \infty & \text{otherwise} \end{cases} \tag{7.19}$$

This infinite step function is equally difficult to optimize. We can overcome this difficulty by introducing *Lagrange multipliers*. The idea of Lagrange multipliers is to replace the step function with a linear function.

We associate to problem (7.17) the **Lagrangian** by introducing the Lagrange multipliers $\lambda_i \geq 0$ corresponding to each inequality constraint, so that

$$\mathcal{L}(x, \lambda) = f(x) + \sum_{i=1}^{m} \lambda_i g_i(x), \tag{7.20a}$$

or equivalently

$$\mathcal{L}(x, \lambda) = f(x) + \lambda^\top g(x), \tag{7.20b}$$

where in the last line we have concatenated all constraints $g_i(x)$ into a vector $g(x)$, and all the Lagrange multipliers into a vector $\lambda \in \mathbb{R}^m$.

Lagrangian duality: The idea of converting an optimization problem in one set of variables x (called the primal variables), into another optimization problem in a different set of variables λ (called the dual variables). The problem in (7.17) is known as the ***primal problem*** corresponding to the primal variables x. The associated **Lagrangian dual problem** to the primal problem is given by

$$\max_{\lambda \in \mathbb{R}^m} \mathcal{D}(\lambda) \quad \text{subject to} \quad \lambda \geq 0, \tag{7.22}$$

where λ are the dual variables and

$$\mathcal{D}(\lambda) = \min_{x \in \mathbb{R}^d} L(x, \lambda).$$

Remark: Two concept of interest: *minimax inequality* and *weak duality* p(234-235) - *minmax inequality* : For any function with two arguments $\varphi(x, y)$, the maximin is less than the minimax, i.e.,

$$\max_x \min_y \varphi(x, y) \quad \leq \quad \min_y \max_x \varphi(x, y) \tag{7.23}$$

- *weak duality*: uses (7.23) to show that primal values are always greater than or equal to dual values.

$$\min_{x\in\mathbb{R}^d} \max_{\lambda\geq 0} \mathcal{L}(x,\lambda) \quad \geq \quad \max_{\lambda\geq 0} \min_{x\in\mathbb{R}^d} \mathcal{L}(x,\lambda) \tag{7.27}$$

(see page 195 for more logic and reasoning)

Equality Contraints: We consider the following constrained optimization problem:

$$\begin{aligned} \min_{x} \quad & f(x) \\ \text{subject to} \quad & g_i(x) \leq 0, \quad i = 1,\ldots,m, \\ & h_j(x) = 0, \quad j = 1,\ldots,n. \end{aligned} \tag{7.28}$$

3.2 7.3. Convex Optimization

When f $(\cdot)$ is a convex function, and when the constraints involving g$(\cdot)$ and h$(\cdot)$ are convex sets.

strong duality : The optimal solution of the dual problem is the same as the optimal solution of the primal problem.

Definition 7.2. : A set $\mathcal{C}$ is a **convex set** if for any $x, y \in \mathcal{C}$ and for any scalar θ with $0 \leq \theta \leq 1$, we have

$$\theta x + (1-\theta)y \in \mathcal{C}. \tag{7.29}$$

Definition 7.3. : Let function $f : \mathbb{R}^D \to \mathbb{R}$ be a function whose domain is a convex set. The function f is a **convex function** if for all x, y in the domain of f, and for any scalar θ with $0 \leq \theta \leq 1$, we have

$$f\big(\theta x + (1-\theta)y\big) \leq \theta f(x) + (1-\theta) f(y). \tag{7.30}$$

Remark. A concave function is the negative of a convex function.

Epigraph of the convex function: The resulting filled-in set after *"filling in"* a convex function.

Convexity in terms of its gradient $\nabla_x f(x)$: A function $f(x)$ is **convex** if and only if for any two points x, y it holds that

$$f(y) \;\geq\; f(x) + \nabla_x f(x)^\top (y - x). \tag{7.31}$$

If $f(x)$ is **twice differentiable**, then

$$(x) \text{ is convex} \iff \nabla_x^2 f(x) \succeq 0,$$

where $\nabla_x^2 f(x)$ is the Hessian matrix of $f(x)$ and $\succeq 0$ denotes **positive semidefiniteness**.

Remark: The inequality in (7.30) is sometimes called *Jensen's inequality*.

In summary, a **constrained optimization problem** is called a **convex optimization problem** if

$$\begin{aligned} \min_{x} \quad & f(x) \\ \text{subject to} \quad & g_i(x) \leq 0, \quad i = 1, \ldots, m, \\ & h_j(x) = 0, \quad j = 1, \ldots, n, \end{aligned} \tag{7.38}$$

where all functions $f(x)$ and $g_i(x)$ are convex functions, and all equality constraints $h_j(x) = 0$ define affine sets.

3.2.1 7.3.1. Linear Programming

Linear Program : All the preceding functions are linear, it has d variables and m linear constraints.

$$\begin{aligned} \min_{x \in \mathbb{R}^d} \quad & c^\top x \\ \text{subject to} \quad & Ax \leq b, \text{where } A \in \mathbb{R}^{m \times d},\ b \in \mathbb{R}^m. \end{aligned} \tag{7.39}$$

The **Lagrangian** is given by

$$\mathcal{L}(x, \lambda) \ = \ c^\top x \ + \ \lambda^\top (Ax - b), \tag{7.40}$$

where $\lambda \in \mathbb{R}^m$ is the vector of non-negative Lagrange multipliers.

Rearranging the terms correponding to x yields:

$$\mathcal{L}(x, \lambda) \ = \ (c + A^\top \lambda)^\top x \ - \ \lambda^\top b. \tag{7.41}$$

Taking the derivative of $\mathcal{L}(x, \lambda)$ with respect to x and setting it to zero gives

$$\frac{\partial \mathcal{L}(x, \lambda)}{\partial x} = c + A^\top \lambda = 0. \tag{7.42}$$

Dual Lagrangian and Dual Problem

From the Lagrangian, the **dual function** is

$$\mathcal{D}(\lambda) = -\lambda^\top b.$$

We want to **maximize** $\mathcal{D}(\lambda)$ subject to the stationarity condition and non-negativity of the multipliers:

$$\begin{aligned} \max_{\lambda \in \mathbb{R}^m} \quad & -b^\top \lambda \\ \text{subject to} \quad & c + A^\top \lambda = 0, \\ & \lambda \geq 0. \end{aligned} \tag{7.43}$$

Notes:
- d = number of primal variables, m = number of primal constraints.
- The dual LP has m variables, so depending on whether m or d is larger, we may choose to solve the **primal** (7.39) or **dual** (7.43) problem.

3.2.2 7.3.2 Quadratic Programming

quadratic program: the case of a convex quadratic objective function, where the constraints are affine. It has d variables and m linear constraints.

$$\begin{aligned} \min_{x \in \mathbb{R}^d} \quad & \frac{1}{2} x^\top Q x + c^\top x \\ \text{subject to} \quad & Ax \leq b, \\ & A \in \mathbb{R}^{m \times d},\ b \in \mathbb{R}^m,\ c \in \mathbb{R}^d. \end{aligned} \tag{7.45}$$

where the square symmetric matrix $Q \in \mathbb{R}^{d \times d}$ is positive definite, and therefore the objective function is *convex*.

The Lagrangian for the convex quadratic program is

$$\mathcal{L}(x, \lambda) = \frac{1}{2} x^\top Q x + c^\top x + \lambda^\top (Ax - b) \tag{7.48a}$$

which can also be rearranged as

$$\mathcal{L}(x, \lambda) = \frac{1}{2} x^\top Q x + (c + A^\top \lambda)^\top x - \lambda^\top b. \tag{7.48b}$$

Taking the derivative with respect to x and setting it to zero gives the stationarity condition:

$$\frac{\partial \mathcal{L}(x, \lambda)}{\partial x} = Qx + (c + A^\top \lambda) = 0. \tag{7.49}$$

Since Q is positive definite (and therefore invertible), we solve for x:

$$x = -Q^{-1}(c + A^\top \lambda). \tag{7.50}$$

Substituting this back into the Lagrangian gives the **dual function**:

$$D(\lambda) = -\frac{1}{2}(c + A^\top \lambda)^\top Q^{-1}(c + A^\top \lambda) - \lambda^\top b. \tag{7.51}$$

Finally, the **dual optimization problem** is

$$\begin{aligned} \max_{\lambda \in \mathbb{R}^m} \quad & -\frac{1}{2}(c + A^\top \lambda)^\top Q^{-1}(c + A^\top \lambda) - \lambda^\top b \\ \text{subject to} \quad & \lambda \geq 0. \end{aligned} \tag{7.52}$$

3.2.3 7.3.3 Legendre–Fenchel Transform and Convex Conjugate

Convex set can be described by its *supporting hyperplanes*. A hyperplane is called a *supporting hyperplane* of a convex set if it intersects the convex set, and the convex set is contained on just one side of it.

The Legendre transform : A concept that states : a convex functions can be equivalently described by a function of their gradient.

The Legendre-Fenchel transform (or the convex conjugate) : is a transformation (in the sense of a Fourier transform) from a convex differentiable function $f(x)$ to a function that depends on the tangents $s(x) = \nabla_x f(x)$. (the transformation of $f(\dot{})$ and not the variable x or the function evaluated at x.

Defintion 7.4. : The **convex conjugate** of a function $f : \mathbb{R}^D \to \mathbb{R}$ is a function f^* defined by

$$f^*(s) = \sup_{x \in \mathbb{R}^D} \big(\langle s, x \rangle - f(x)\big). \tag{7.53}$$

See p(243) in the book for more geometrical illutrations.

The Legendre-Fenchel conjugate turns out to be quite useful for machine learning problems that can be expressed as convex optimization problems.

[ ]: