Lecture 0b: Tractable multivariate Bayesian inference

Now that we covered the topic of tractability and analytic tractability in univariate models. We can finally move on the models with many latent variables. This is the bread and butter of machine learning and almost everything we do in this field involves from hundreds to billions of variables and parameters. While tractability is common in practically useful univariate models, it is unfortunately very rare in multivariate models. The problem, which we have already discussed at the beginning of this chapter, is that the number of distinguishable states increases exponentially with the number of parameters. This is reflected in the number of total states in discrete models and the number of bins you need to approximate the evidence integral reliably for continuous models. While having a large number of distinguishable states is a course, tractability is still possible when there is special structure in the parameter space. For example, as we discussed at the beginning of the chapter, this structure can be concentration, meaning that only a exponentially small structure of states have non-vanishing probability. Tractability can also come from a set of symmetries that partition the state space into a large number of equivalent copies of a small tractable set. In a multivariate inference problem, we aim to estimate the posterior over several variables $x_1, \ldots, x_N$. In this case, the evidence integral is

$$p(D) = \int_{a_1}^{b_1} \int_{a_2}^{b_2} \cdots \int_{a_N}^{b_N} p(D \mid x_1, \ldots, x_N) p(x_1, \ldots, x_N) \mathrm{d}x_1 \ldots \mathrm{d}x_N \ . \qquad (1)$$

This multivariate integrals are again defined as limit of sum over quantization of the function:

$$p(D) = \lim_{\Delta x \to 0} \sum_{k_1} \cdots \sum_{k_N} p(D \mid x_{k_1}, \ldots, x_{k_N}) p(x_{k_1}, \ldots, x_{k_N}) \left(\Delta z\right)^N \ . \qquad (2)$$

In the univariate case, we saw that we could approximate the integral arbitrarily well by replacing the limit with a small value of Δx. However, this approach quickly become hopeless in the multivariate case as the total number of bins is an exponential function of the number of dimensions. Sometimes we can solve the integral in the good-old calculus way by solving the individual univariate integrals one by one in a recursive way. However, there are often smarter ways which exploit the symmetries of the underlying function. However, analytical tractability in any sort of multivariate integral is very rarely possible.

These expressions for multivariate integrals are very cumbersome to write! We therefore often use a single symbol of integration with omitted bounds and and collect the variables into a vector $\boldsymbol{x} = (x_1, \ldots, x_N)$. We can then express the integral as

$$p(D) = \int p(D \mid \boldsymbol{x}) p(\boldsymbol{x}) \mathrm{d}\boldsymbol{x} \ . \tag{3}$$

As noted before, we are to usually able to evaluate these integrals neither using the tools of calculus or the quantization method. Is it then useful to work with mathematical object that we cannot really evaluate? It turns out that our profession of "statisticians" also helps us evaluating these math expressions! In fact, the integral in Eq. 3 can be seen as the expected value of the function $p(D \mid \boldsymbol{x})$ with respect to the sampling distribution $p(\boldsymbol{x})$:

$$p(D) = \mathbb{E}_{\boldsymbol{x} \sim p(D)}[p(D \mid \boldsymbol{x})] \ . \tag{4}$$

We can now approximate the integral by replacing the expectation with an average over a finite number of samples:

$$p(D) \approx \frac{1}{N} \sum_n p(D \mid \boldsymbol{x}_n) \ , \qquad \boldsymbol{x}_n \sim p(\boldsymbol{x}). \tag{5}$$

This is a very amusing turn of events! We started using probabilities for dealing with the messiness of the real-wold and we ended up using the same probabilistic techniques for estimating precise mathematics quantities! This is actually very fortunate as we can re-cycle methods developed for one purpose (e.g. averages of real data) to this new purpose which is central for solving the original statistical problem. Unfortunately, this approach does not work in most inference situation since you would need an exponentially large number of samples to get a proper accuracy. You cannot cheat intractability with a simple reformulation. However, this sort of *Monte Carlo* techniques will be central in most of the advanced approximation schemes we will discuss in this course.

0.1 A more abstract notation

Very often we want to write expression that work without changes in both the univariate and the multivariate case. In these situations we drop the bold symbol for vectors and we express a potentially multivariate integral using the same notation of univariate integrals. For example, we usually write the KL divergence between two potentially distributions $p(x)$ and $q(x)$ as follows

$$D_{\mathrm{KL}}[p, q] = \int p(x) \log \left(\frac{p(x)}{q(x)} \right) \mathrm{d}x \ . \tag{6}$$

This is a step of abstraction. Some results are valid regardless to the dimensionality of the inference problem and we want to express them without the need to represent them into a specific N-dimensional integral. Abstracting details away allows you to reason at an higher level without being burdened by details you already understand in principle. However, this abstraction should be couple with hands on experience in the nitty-gritty details of specific inference problems.

0.2 Marginalization and conditioning

Two operations have fundamental importance in multivariate statistics: marginalization and conditioning. A large fraction of derivations we will perform in the rest of this book are just a sequence of these two operations. It is therefore crucially important for you to become deeply familiar with both their mathematical definition and their intuitive meaning. Consider a three-variate distribution defined by the density

$$p(x_1, x_2, x_3) \ .$$

The multivariate density can be use to assign probability to specific pattern of values. This is often referred to as *joint density*, *joint distribution* or simply *joint*. The information encoded in the joint density captures the statistical dependencies between the variables. For example, the joint density could tell us that x_1 is likely to be equal to 1 when x_2 is approximately equal to x_3 while being likely equal to -1 otherwise. However, we are often just interested in the probable values of a single variable such as x_1 regardless to the values of other variables. There are two main ways to obtain a univariate distribution over x_1 given the joint. If we do not know the values of x_2 and x_3, we can remove them from the joint by integrating over their range:

$$p(x_1) = \int p(x_1, x_2, x_3) \mathrm{d}x_2 \mathrm{d}x_3 \ . \tag{7}$$

This gives us the so called *marginal distribution* of x_1. This expression tells us that the probability density $p(x_1)$ can be obtained by summing (integrating) the density of all pattern of possible values in which the first variable is equal to x_1. Similarly, we can obtain the *marginal joint distribution* of x_1 and x_2 by integrating over the single variable x_3:

$$p(x_1, x_2) = \int p(x_1, x_2, x_3) \mathrm{d}x_3 \ . \tag{8}$$

In this case, we say that x_3 has been "integrated out" or "marginalized out" since the resulting distribution does no longer depend on x_3. All these formulae can be extended to higher dimensional multivariate distributions in the obvious way. In general, we can obtain the marginal joint distribution of a subset of variables by integrating out all the remaining variable from the joint distribution. Unfortunately, these marginalization integrals cannot usually be computed in closed form and are often difficult to approximate. However, even if we do not have a neat expression for a marginal density such as $p(x_1)$, it is always very easy to sample a value x_1 from it as far as we can sample x_1, x_2, x_3 from the joint density. In fact, this simply entails sampling $x_1, x_2, x_3 \sim p(x_1, x_2, x_3)$ and ignoring the values x_2 and x_3.

Marginal (joint) distributions are used when we are interested in the values of a set of variables while the value of the remained variables are unknown. On the other hand, returning for now to our three-variate example, if the values of x_2 and x_3 are already known then the relevant univariate distribution over x_1 is

the conditional distribution

$$p(x_1 \mid x_2, x_3) = \frac{p(x_1, x_2, x_3)}{p(x_2, x_3)} \ . \tag{9}$$

Note that the denominator in the formula is the joint marginal distribution

$$p(x_2, x_3) = \int p(x_1, x_2, x_3) \mathrm{d}x_1 \ . \tag{10}$$

We can also obtain joint conditional distribution over a set of variables such as x_1 and x_2 by conditioning on the remaining variables:

$$p(x_1, x_2 \mid x_2, x_3) = \frac{p(x_1, x_2, x_3)}{p(x_3)} \ . \tag{11}$$

Again, these expressions can be generalized to the N-variate case in the obvious way.

As you probably have already noticed, we have already used these formulae extensively under the pseudonym of Bayes' theorem. In fact, Bayesian inference is nothing more than conditioning of a set of variables (which we called latents) given another set of variables that we assume to be directly observable. In this contexts, we usually write the joint distribution $p(D, \boldsymbol{x})$ over the data D and the latent x as a product between a likelihood distribution $p(D \mid \boldsymbol{x})$ and a prior distribution $p(\boldsymbol{x})$. For example, in our Bayesian analysis of a thermometer, we assumed to know the height of the mercury bar and we obtained the probability of the body temperature given the observed value.

0.3 Change of variable formula

The primary aim of this book is to integrate the techniques of Bayesian statistics with the powerful tools of deep learning and differentiable programming. Deep learning networks are multivariate differentiable transformations that map input variables $\boldsymbol{x}$ into a transformed output $\boldsymbol{f}(\boldsymbol{x}) = \boldsymbol{z}$. For example, the function f may be a convolutional neural network that alternate convolutions with *tanh* activation functions. In probability theory, variables are associated to probability densities that determine their statistical behavior. It is therefore natural to ask if it is possible to obtain the probability density of the transformed variable $\boldsymbol{z} = \boldsymbol{f}(\boldsymbol{x})$ if we know that the input variable $\boldsymbol{x}$ follows a distribution $p_0(\boldsymbol{x})$. Every time we are trying to answer a problem, we should also start by what we intuitively know for sure. In this case, we know that we can sample the variable $\boldsymbol{x}$ simply by sampling $\boldsymbol{x}$ and then applying the transformation to the resulting sample. We can summarize this realization in the formula

$$f(\boldsymbol{x}) \sim p(z) \text{ with } \boldsymbol{x} \sim p_0(\boldsymbol{x}) \ . \tag{12}$$

Note that we do not know the form of the density $p(z)$ jet. Expectations are obtained by averaging samples. It is therefor clear that we can also express the expected value of the distribution $p(\boldsymbol{z})$ in terms of the distribution $p_0(\boldsymbol{x})$:

$$\mathbb{E}_{\boldsymbol{z} \sim p(\boldsymbol{z})}[\boldsymbol{z}] = \mathbb{E}_{\boldsymbol{x} \sim p_0(\boldsymbol{x})}[\boldsymbol{f}(\boldsymbol{x})] \ . \tag{13}$$

This does not fully characterize the distribution $p(\boldsymbol{z})$ since many different distributions can have the same expected value. However, we can extend this reason into a full characterization by introducing a test function $\phi(\boldsymbol{z})$. It is easy to see that the expectation of any test function can be obtained in terms of $p_0(\boldsymbol{x})$ since it is just an average of values of $\phi(\boldsymbol{z})$ for inputs sampled from $\boldsymbol{z}$, which we know how to sample. This results in the following formula:

$$\mathbb{E}_{\boldsymbol{z}\sim p(\boldsymbol{z})}[\phi(\boldsymbol{z})] = \mathbb{E}_{\boldsymbol{x}\sim p_0(\boldsymbol{x})}[\phi(\boldsymbol{f}(\boldsymbol{x}))]\,. \tag{14}$$

It makes intuitive sense that this formula fully characterizes the behavior of $p(\boldsymbol{z})$ since it holds for every possible test function, which we can "configure" to probe different parts of the distributions. For example, if we set $\phi(\boldsymbol{z})$ to be the delta function $\delta(\boldsymbol{z}-\boldsymbol{z}^*)$, we can directly probe the value of the density at the point $\boldsymbol{z}^*$. You can visualize the delta test function $\delta(\boldsymbol{z}-\boldsymbol{z}^*)$ as being zero everywhere except for a spike at $\boldsymbol{z}=\boldsymbol{z}*$ which singles out a single value within an integral. Applied to our expectation this leads to the formula

$$\mathbb{E}_{\boldsymbol{z}\sim p(\boldsymbol{z})}[\delta(\boldsymbol{z}-\boldsymbol{z}^*)] = \int \delta(\boldsymbol{z}-\boldsymbol{z}^*)p(\boldsymbol{z})\mathrm{d}\boldsymbol{z} = p(\boldsymbol{z}^*)\ . \tag{15}$$

In order to find a formula for the density $p(\boldsymbol{z})$, we now need to rewrite Eq. 14 as an integral with respect to the new variable $\boldsymbol{z}$. For now, let us assume that the function $\boldsymbol{f}$ is invertible, meaning that there exist another function $\boldsymbol{f}^{-1}(\boldsymbol{z})$ such that $\boldsymbol{f}(\boldsymbol{f}^{-1}(\boldsymbol{z})) = \boldsymbol{z}$ and $\boldsymbol{f}^{-1}(\boldsymbol{f}(\boldsymbol{x})) = \boldsymbol{x}$. Invertible differentiable functions are often called *diffeomorphisms* by mathematicians or by somewhat self-aggrandizing machine learning researchers. Using the inverse, we can express the expectation as follows

$$\mathbb{E}_{\boldsymbol{x}\sim p_0(\boldsymbol{x})}[\phi(\boldsymbol{x})] = \int \phi(\underbrace{\boldsymbol{f}(\boldsymbol{x})}_{\boldsymbol{z}})p_0(\underbrace{\boldsymbol{x}}_{f^{-1}(\boldsymbol{z})})\underbrace{\mathrm{d}\boldsymbol{x}}_{??}\ . \tag{16}$$

To properly transform the multivariate differential we need to go back to our books on multivariate calculus. After some studying, we will find the following pretty transformation rule for differentials:

$$\mathrm{d}\boldsymbol{x} = |\det D\boldsymbol{f}^{-1}(\boldsymbol{z})|\mathrm{d}\boldsymbol{z} \tag{17}$$

where

$$D\boldsymbol{f}^{-1} = \begin{bmatrix} \frac{\partial f_1^{-1}(\mathbf{z})}{\partial z_1} & \cdots & \frac{\partial f_1^{-1}(\mathbf{z})}{\partial z_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial f_m^{-1}(\mathbf{z})}{\partial z_1} & \cdots & \frac{\partial f_m^{-1}(\mathbf{z})}{\partial z_n} \end{bmatrix} \tag{18}$$

is the Jacobi matrix of the function inverse $\boldsymbol{f}^{-1}$ which describe its linear behavior around a point $\boldsymbol{z}$. using this fomrula, we obtain:

$$\mathbb{E}_{\boldsymbol{x}\sim p_0(\boldsymbol{x})}[\phi(f(\boldsymbol{x}))] = \int \phi(\boldsymbol{z})p_0(f^{-1}(\boldsymbol{z})|\det Df(\boldsymbol{z})|^{-1}\mathrm{d}\boldsymbol{z}\ . \tag{19}$$

We can now extract the value of the density $p(\boldsymbol{z})$ at an arbitrary point $\boldsymbol{z}^*$ by using a delta test function:

$$\int \delta(\boldsymbol{z} - \boldsymbol{z}^*) p_0(f^{-1}(\boldsymbol{z})| \det Df^{-1}(\boldsymbol{z})|^{-1} \mathrm{d}\boldsymbol{z} = p_0(f^{-1}(\boldsymbol{z}^*)| \det Df^{-1}(\boldsymbol{z}^*)| \ . \quad (20)$$

This leads to the very important formula:

$$p(\boldsymbol{z}) = p_0(f^{-1}(\boldsymbol{z}))| \det Df^{-1}(\boldsymbol{z})| \ , \quad (21)$$

which allows us to obtain the density of a transformed variable using the inverse transformation and the determinant of its Jacobi matrix. We can now analyze the formula in order to obtain more intuitive understanding. The term $p_0(f^{-1}(\boldsymbol{z}))$ tells us that we need to look at the density of the value used to "generate" the variable $\boldsymbol{z}$. This is intuitively clear, if a value of $\boldsymbol{x}$ is unlikely and the function is one-to-one, then its transformed value $\boldsymbol{f}(\boldsymbol{x})$ will be unlikely as well. Therefore, the inverse function $\boldsymbol{f}^{-1}$ is used to recover the probability density of the input in order to evaluate the probability of the density of the output. More mysterious is the appearance of the absolute determinant of the Jacobi matrix. Let us try to make sense of this term. First of all, it is relatively easy to show that the determinant of the inverse Jacobi matrix $D\boldsymbol{f}^{-1}$ is equal to the inverse of the determinant of the Jacobi matrix of $\boldsymbol{f}$:

$$\det D\boldsymbol{f}^{-1}(\boldsymbol{z}) = \frac{1}{\det D\boldsymbol{f}(\boldsymbol{x}^*)} \ , \quad (22)$$

where $\boldsymbol{x}^*$ is $\boldsymbol{f}^{-1}(\boldsymbol{z})$. As noted above, the Jacobi matrix $\det D\boldsymbol{f}(\boldsymbol{x}^*)$ gives the linear transformation that best approximate the function around the point $\boldsymbol{x}^*$. Therefore, in order to understand the appearance of its inverse absolute determinant in the formula, we need to understand what is the significance of the determinant of a linear transformation. It turns out that the determinant of a transformation capture how much volume is either expanded or contracted. For example, consider a square defined by the four vertices $(0,0), (0,1), (1,0)$ and $(1,1)$. The volume of this square is equal to one. Let us now consider a linear transformation encoded by the matrix

$$A = \begin{pmatrix} a & 0 \\ 0 & b \end{pmatrix} \ . \quad (23)$$

if we apply this matrix to the four vertices, we obtain the new set of vertices $(0,0), (0,b), (a,0)$ and (a,b). This is a rectangle with volume $a \times b$. This is visualized in Fig. 1a. This is also the value of the determinant of the matrix since, when a matrix is diagonal, its determinant is equal to the product of the diagonal elements. Note that, if either a or b is equal to zero, then the determinant becomes equal to zero as well and the square collapses into a line segment. In this case, the transformation is not invertible since the information encoded in the values of one of the axis is lost and multiple input points map to the same output point. The relationship between the value of the determinant and the volume

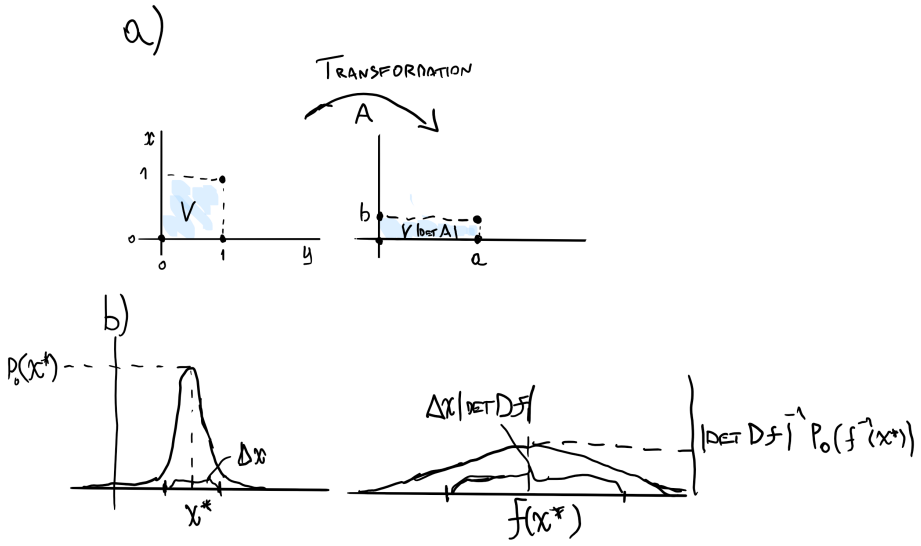

Figure 1: Visualization of volume distortions and the change of density formula.

distortion effect of the transformation remains true in arbitrary dimension and for non-diagonal matrices. Furthermore, the sign of the determinant tells us if a transformation turns left handed shapes into right handed shapes by flipping the axes.

Summarising, the absolute value of the determinant of the Jacobi matrix in Eq. 21 tells us how much the volume around a point $\boldsymbol{x}^*$ is either expanded or contracted. We can now try to make sense of the inverse determinant in the formula. Imagine that a point $\boldsymbol{x}^*$ with density $p_0(\boldsymbol{x}^*)$ is mapped by the function into the point $\boldsymbol{z} = f(\boldsymbol{x})$. Locally, we can analyze the transformation in terms of the linear mapping encoded by the Jacobi matrix $D\boldsymbol{f}(\boldsymbol{x}^*)$. If the determinant of this matrix is much larger than one, the volume around $\boldsymbol{x}^*$ is greatly inflated and ends up covering a much larger region around $\boldsymbol{z}$. Therefore, the density gets spread out more so that less density is assigned to the point $\boldsymbol{z}$. The formula takes into account of this spreading out effect by dividing by the absolute determinant, so to reduce the value of the density when the volume is expanded by $\boldsymbol{f}$. On the other hand, the same formula tells us that the value of the density is increased when the volume around $\boldsymbol{x}^*$ is compressed since more density is concentrated on the point $\boldsymbol{z}$. This behavior is visualized in Fig. 1b.

1 The multivariate Gaussian distribution

While plenty of univariate models are analytic tractable, analytic tractability is very rare in multivariate problems. The only widely used analytically tractable models are the multivariate Gaussian model for continuous multivariate measurements and the Dirichlet-multinomial model for categorical data. These models are simple to learn and should be in the toolbox of any machine learning researcher and engineers. In this section, we will discuss the theory of multivariate Gaussian inference and get some insights into the deeper reasons of its tractability. A N-variables multivariate Gaussian distribution can be obtained by applying a linear transformation to a vector of standard univariate random variables $\boldsymbol{\xi}$:

$$\boldsymbol{x} = f(\boldsymbol{\xi}) = A\boldsymbol{\xi} + \boldsymbol{\mu} \tag{24}$$

with

$$p(\boldsymbol{\xi}) = \prod_{n=1}^{N} \frac{1}{(2\pi)^{1/2}} e^{-\frac{1}{2}\xi_n^2} = \frac{1}{(2\pi\sigma^2)^{N/2}} e^{-\frac{1}{2}\sum_{n=1}^{N}\xi^2} = \frac{1}{(2\pi)^{N/2}} e^{-\frac{1}{2}\boldsymbol{\xi}^T\boldsymbol{\xi}} \ . \tag{25}$$

The transformation matrix A introduces correlations since each new variable is a combination of a set of independently variables which give a shared source of variability. We can now express the density of the transformed variable using the change of variables formula for the density:

$$p(\boldsymbol{x}) = \left|\det\left[\frac{\partial \boldsymbol{x}}{\partial \boldsymbol{\xi}}\right]\right|^{-1} p(f^{-1}(\boldsymbol{x})) \ , \tag{26}$$

where $\left[\frac{\partial \boldsymbol{x}}{\partial \boldsymbol{\xi}}\right]$ is the matrix of partial derivative of the transformation (the so called Jacobi matrix). This matrix tells how how much the density is distorted by the transformation around each point $\boldsymbol{\xi}$. In particular, the absolute determinant of the matrix is a measure of "change of volume" around each point. If the absolute determinant of the matrix is bigger then one these space is stretched around that point while, if it is smaller than one, the space is compressed.
In our case, the transformation is linear and therefore the matrix of partial derivatives is equal to the transformation matrix itself:

$$p(\boldsymbol{x}) = \left|\det\left[\frac{\partial \boldsymbol{x}}{\partial \boldsymbol{\xi}}\right]\right|^{-1} p(f^{-1}(\boldsymbol{x})) \ , \tag{27}$$

$$= |\det A|^{-1} \frac{1}{(2\pi)^{N/2}} e^{-\frac{1}{2}\left(A^{-1}(\boldsymbol{x}-\boldsymbol{\mu})\right)^T\left(A^{-1}(\boldsymbol{x}-\boldsymbol{\mu})\right)} \ , \tag{28}$$

$$= \frac{1}{|\det A|\,(2\pi)^{N/2}} e^{-\frac{1}{2}(\boldsymbol{x}-\boldsymbol{\mu})^T\left(AA^T\right)^{-1}(\boldsymbol{x}-\boldsymbol{\mu})} \ . \tag{29}$$

Interestingly, the exponent of this expression depends on the transformation matrix only through the combination AA^T. This new parameter is the covariance

matrix of the multivariate Gaussian, which as you can guess gives the covariance between all pairs of variables:

$$C = AA^T \tag{30}$$

Since $\det AA^T \det A^2$, we can write the multivariate Gaussian probability density solely in terms of the mean vector $\boldsymbol{\mu}$ and the covariance matrix C:

$$\mathcal{N}(\boldsymbol{x}; \boldsymbol{\mu}, C) = \frac{1}{(2\pi \det C)^{N/2}} e^{-\frac{1}{2}(\boldsymbol{x}-\boldsymbol{\mu})^T C^{-1}(\boldsymbol{x}-\boldsymbol{\mu})} \ . \tag{31}$$

The covariance matrix has two important properties that justify its interpretation in terms of covariance between variables. First of all, it is symmetric:

$$C^T = \left(AA^T\right)^T = \left(A^T\right)^T A^T = AA^T = C \ , \tag{32}$$

where we used the fact that the transposition of a matrix product is equal to the reversed product of the individual transposes. The send property is usually referred to as *positive definiteness*, which guarantees that the variance of any linear combination of variables is always positive. More formally, given any vector $\boldsymbol{v}$ we have that

$$\boldsymbol{v}^T C \boldsymbol{v} = \boldsymbol{v}^T AA^T \boldsymbol{v} = (A^T \boldsymbol{v})^T (A^T \boldsymbol{v}) = \left\| A^T \boldsymbol{v} \right\|^2 \geq 0 \ . \tag{33}$$

1.1 The canonical form

The usual parameterization of multivariate Gaussian distributions in terms of a mean vector and a covariance matrix is very intuitive. However, in many calculations it is more convenient to parameterize the distribution in a different form. This can be easily done by expanding the square in the exponent of a multivariate Gaussian density:

$$(\boldsymbol{x}-\boldsymbol{\mu})^T C^{-1} (\boldsymbol{x}-\boldsymbol{\mu}) = \boldsymbol{x}^T C^{-1} \boldsymbol{x} - 2(C^{-1}\boldsymbol{\mu})^T \boldsymbol{x} \ , \tag{34}$$

where we used the fact that C^{-1} is symmetric to re-write $\boldsymbol{\mu}^T C^{-1}$ as $(C^{-1}\boldsymbol{\mu})^T$. We can now define the two natural parameters

$$\boldsymbol{\eta} = C^{-1}\boldsymbol{\mu} \tag{35}$$

and

$$\Omega = C^{-1} \ . \tag{36}$$

This is the so called *canonical parameterization* of the multivariate Gaussian. The Ω parameter is simply the inverse of the covariance matrix and it is often referred to as *precision matrix*. We can now write the density with respect to the natural parameters as follows

$$p(\boldsymbol{x}; \boldsymbol{\eta},) = e^{-x^T \Omega x + \eta^T x - A(\eta, \Omega)} \ , \tag{37}$$

where the function $A(\boldsymbol{\eta}, \omega)$ collects all the terms that do not depend on $\boldsymbol{x}$:

$$A(\eta, \Omega) = -\frac{N}{2} \log (2\pi \det \Omega) - \frac{1}{2} \boldsymbol{\eta}\ \Omega^{-1} \eta\ . \tag{38}$$

The use of the canonical form makes the derivation of Bayesian posterior distributions and other conditioning formula considerably easier. It is therefore crucially important to be familiar with this parameterization. Once a formula has been found in the natural parameterization, the usual mean and covariance parameters can be obtain from it by inverting Eq. 35 and Eq. 36:

$$\boldsymbol{\mu} = \Omega^{-1} \boldsymbol{\eta} \tag{39}$$

and

$$C = \Omega^{-1}\ . \tag{40}$$

1.2 Marginalization and conditioning

Marginalization and conditioning are usually intractable operations that cannot be performed in closed form and often not even tractably approximated. However, in the case of the multivariate Gaussian distribution we have that all marginal and conditional distributions are still multivariate Gaussians, whose parameters can be computed in closed form! This extreme analytical tractability is part of the reason for the prevalence of Gaussian distributions both in multivariate statistics and in machine learning. In order to write down the formulae, we need to introduce some notation. Consider a multivariate Gaussian variable $\boldsymbol{x}$ with mean vector $\boldsymbol{\mu}$ and covariance matrix C. Let us now split this variable into two sub-variables $\boldsymbol{x} = (\boldsymbol{x}_1, \boldsymbol{x}_2)$. This result in a similar split of the mean vector $\boldsymbol{\mu} = (\boldsymbol{\mu}_1, \boldsymbol{\mu}_2)$ and in a partition of the covariance matrix into four blocks:

$$C = \begin{bmatrix} C_{11} & C_{12} \\ C_{12}^T & C_{22} \end{bmatrix}\ . \tag{41}$$

As you can probably guess, the matrices C_{11} and C_{22} are respectively the covariance matrices of $\boldsymbol{x}_1$ and $\boldsymbol{x}_2$. On the other hand C_{12} is the so called *cross-covariance matrix* between the two set of variables. The formula for the marginal distribution of $\boldsymbol{x}_1$ is simply

$$p(\boldsymbol{x}_1) = \mathcal{N}(\boldsymbol{x}_1; \boldsymbol{\mu}_1, C_{11})\ . \tag{42}$$

This is easy to understand since we know that the marginal distribution is Gaussian and we know both its mean and covariance matrix, which fully determine the distribution. However, proving this formula by solving the integral in the definition of marginalization is surprisingly tricky.
On the other hand, the conditional distribution of $\boldsymbol{x}_1$ given $\boldsymbol{x}_2$ is

$$p(\boldsymbol{x}_1 \mid \boldsymbol{x}_2) = \mathcal{N}(\boldsymbol{x}_1; \boldsymbol{\mu}_1 + C_{12} C_{22}^{-1} (\boldsymbol{x}_2 - \boldsymbol{\mu}_2), C_{11} - C_{12} C_{22}^{-1} C_{12}^T)\ . \tag{43}$$

As you can see, conditioning a value of $\boldsymbol{x}_2$ perturbs the statistics of $\boldsymbol{x}_1$ through the cross-correlation matrix C_{12}. This makes sense since we would not expect any effect of conditioning when the two blocks are uncorrelated.

1.3 Multivariate Gaussian inference

We can now discuss the multivariate Gaussian Bayesian inference problem. Consider D a data-set comprised of M vectors of observations $\boldsymbol{y}_m$. These observations can represent the multiple readings of a set of measuring devices. For example, the devices could be a array of thermometers spaced 1cm apart and placed on an arm. We can model these the probability density of this set of readings using a multivariate Gaussian likelihood:

$$p(D \mid \boldsymbol{x}) = \prod_{m=1}^{M} \mathcal{N}(\boldsymbol{y}_m; \boldsymbol{x}, C) \tag{44}$$

where we assumed that the mean is given by the latent variable $\boldsymbol{x}$ and that the error distribution has covariance matrix C. The diagonal entries of this matrix determine the variance of the error of each device while the off-diagonal components determine the correlations between the errors. For example, in the case of the array of thermometers, error correlations can arise from motion of the arm which disturbs the reading of many thermometers simultaneously.
We can now express the prior as another multivariate Normal distribution:

$$p(\boldsymbol{x}) = \mathcal{N}(\boldsymbol{x}; \boldsymbol{\mu}_0, S_0) \ . \tag{45}$$

In our example, this models the distribution of temperatures on nearby locations of arm skin. The mean vector $\boldsymbol{\mu_0}$ gives the mean temperature value of each location while the covariance matrix S_0 determines the correlations between the temperature values at the different locations. Note that, in this example, these correlations will generally be very high since body temperature tend to not vary substantially between different locations in the body.
To obtain the posterior, we use the usual technique to multiply prior and likelihood and drop all the terms of proportionality that do not depend on $\boldsymbol{x}$:

$$p(\boldsymbol{x} \mid D) \propto \mathcal{N}(\boldsymbol{x}; \boldsymbol{\mu}_0, S_0) \prod_{m=1}^{M} \mathcal{N}(\boldsymbol{y}_m; \boldsymbol{x}, C) \ . \tag{46}$$

$$\propto \exp\left(-\frac{1}{2}(\boldsymbol{x} - \boldsymbol{\mu})^T S_0^{-1} (\boldsymbol{x} - \boldsymbol{\mu}) - \frac{1}{2} \sum_{m=1}^{M} (\boldsymbol{y}_m - \boldsymbol{x})^T C^{-1} (\boldsymbol{y}_m - \boldsymbol{x}) \right) . \tag{47}$$

We can further simplify this expression by expanding the square

$$(\boldsymbol{y}_m - \boldsymbol{x})^T C^{-1} (\boldsymbol{y}_m - \boldsymbol{x}) = \boldsymbol{y}_m^T C^{-1} \boldsymbol{y}_m - 2\boldsymbol{x}^T C^{-1} \boldsymbol{y}_m + \boldsymbol{x}^T C^{-1} \boldsymbol{x} \ ,$$

and by noticing that only two of these three terms depend on $\boldsymbol{x}$. Analogously, we can perform the same simplification to the square in the prior term. If we do

so, we obtain

$$\mathcal{N}(x \mid D) \propto \exp\left(-\frac{1}{2}\boldsymbol{x}^T S_0^{-1}\boldsymbol{x} + \boldsymbol{x}^T S_0^{-1}\boldsymbol{\mu} + \boldsymbol{x}^T C^{-1}\sum_{m=1}^{M}\boldsymbol{y}_m - M\boldsymbol{x}^T C^{-1}\boldsymbol{x}\right), \tag{48}$$

$$= \exp\left(-\frac{1}{2}\boldsymbol{x}^T\left(S_0^{-1} + (C/M)^{-1}\right)\boldsymbol{x} + \boldsymbol{x}^T\left(S_0^{-1}\boldsymbol{\mu} + C^{-1}\sum_{m=1}^{M}\boldsymbol{y}_m\right)\right),$$

$$= \exp\left(-\frac{1}{2}\boldsymbol{x}^T\Omega\boldsymbol{x} + \boldsymbol{x}^T\boldsymbol{\eta}\right).$$

This is a multivariate normal distribution in canonical form with natural parameters

$$\boldsymbol{\eta} = S_0^{-1}\boldsymbol{\mu} + (C/M)^{-1}\frac{1}{M}\sum_{m=1}^{M}\boldsymbol{y}_m \tag{49}$$

and

$$\Omega = S_0^{-1} + (C/M)^{-1}. \tag{50}$$

The natural parameter can be used to obtain the mean and covariance matrix of the posterior distribution using the following formulas:

$$\boldsymbol{\mu}_{\text{new}} = \Omega^{-1}\boldsymbol{\eta} = \left(S_0^{-1} + (C/M)^{-1}\right)^{-1}\left(S_0^{-1}\boldsymbol{\mu} + (C/M)^{-1}\frac{1}{M}\sum_{m=1}^{M}\boldsymbol{y}_m\right), \tag{51}$$

and

$$S_{\text{new}} = \Omega^{-1} = \left(S_0^{-1} + (C/M)^{-1}\right)^{-1}. \tag{52}$$

Therefore, we arrived at the expression for the multivariate Gaussian posterior:

$$p(\boldsymbol{x} \mid D) = \mathcal{N}(\boldsymbol{x}; \boldsymbol{\mu}_{\text{new}}, S_{\text{new}}). \tag{53}$$

Now that we have the exact posterior. We can compute the model evidence by re-arranging Bayes formula using a few elementary algebraic manipulations:

$$p(D) = \frac{p(D \mid \boldsymbol{x})p(\boldsymbol{x})}{p(\boldsymbol{x} \mid D)}. \tag{54}$$

All terms in the right hand side of the equation are known and we can therefore use them to compute the model evidence in the left hand side. Note that, while all right hand side terms depend on $\boldsymbol{x}$, the model evidence as expressed by Eq. 54 does not. This means that we can perform the explicit calculation using any value of $\boldsymbol{x}$ we like while always getting the same answer. In practice, this freedom can be exploited by choosing a value of $\boldsymbol{x}$ that simplify the calculation. For example, we can set $\boldsymbol{x} = \boldsymbol{\mu}_{\text{new}}$ so to make the square term in the posterior

to vanish. With some computation (do them!), we can show that

$$p(D) = \prod_{m=1}^{M} \mathcal{N}(y_m \mid \boldsymbol{\mu}_0, S_{\text{new}}) \tag{55}$$
$$= \frac{1}{(2\pi|\det S_{\text{new}}|)^{MN/2}} \exp\left(-\frac{1}{2}\sum_{m=1}^{M}(\boldsymbol{y}_m - \boldsymbol{\mu}_0)^T S_{\text{new}}^{-1}(\boldsymbol{y}_m - \boldsymbol{\mu}_0)\right) .$$

This formula tells us that the marginal distribution of the data is again a multivariate Gaussian with the same mean vector as the prior and the covariance matrix of the posterior. This expression is also referred to as *marginal likelihood* since it is the likelihood of the model once we marginalize out the latent variable $\boldsymbol{x}$.
There is however a much easier method to obtain this formula. By definition of the Gaussian likelihood, we have that

$$\boldsymbol{y}_m = \boldsymbol{x} + \boldsymbol{\xi} \tag{56}$$

where $\boldsymbol{x}$ is the mean of the measurement given the latent and $\boldsymbol{\xi}$ is a centered (i.e. zero mean) Gaussian noise vector with covariance matrix C. Moreover, the prior tells us that

$$\boldsymbol{x} = \boldsymbol{\mu} + \boldsymbol{\zeta} \tag{57}$$

where $\boldsymbol{\mu}$ is the prior mean and $\boldsymbol{\zeta}$ is a centered Gaussian vector with covariance S_0. if we combine these two formulas, we obtain

$$\boldsymbol{y}_m = \boldsymbol{\mu}_0 + \boldsymbol{\zeta} + \boldsymbol{\xi} \ . \tag{58}$$

The distribution of y_m is a multivariate Gaussian since the sum of two Gaussian variables is always another Gaussian variables. Furthermore, its mean is the prior mean $\boldsymbol{\mu}$ since

$$\mathbb{E}[\boldsymbol{y}_m] = \mathbb{E}[\boldsymbol{\mu}_0 + \boldsymbol{\zeta} + \boldsymbol{\xi}] = \boldsymbol{\mu} + \mathbb{E}[\boldsymbol{\zeta}] + \mathbb{E}[\boldsymbol{\xi}] = \boldsymbol{\mu}_0 \ . \tag{59}$$

Finally, the covariance of y_m is given by

$$\mathbb{E}\left[(\boldsymbol{y}_m - \boldsymbol{\mu}_0)^T(\boldsymbol{y}_m - \boldsymbol{\mu}_0)\right] = \mathbb{E}\left[\boldsymbol{\xi}^T\boldsymbol{\xi}\right] + \mathbb{E}\left[\boldsymbol{\zeta}^T\boldsymbol{\zeta}\right] + 2\mathbb{E}\left[\boldsymbol{\xi}^T\boldsymbol{\zeta}\right] \tag{60}$$
$$= C + S_0 \ ,$$

since $\boldsymbol{\xi}$ and $\boldsymbol{\zeta}$ are uncorrelated sources of randomness and therefore $\mathbb{E}\left[\boldsymbol{\xi}^T\boldsymbol{\zeta}\right] = 0$. Therefore, the marginal distribution of the m-th data point is

$$p(\boldsymbol{y}_m) = \mathcal{N}(\boldsymbol{y}_m; \boldsymbol{\mu}_0, S_0 + C) \tag{61}$$
$$= \mathcal{N}(\boldsymbol{y}_m; \boldsymbol{\mu}_0, S_{\text{new}}) \ .$$

1.3.1 Multivariate inference as selective filtering

In this section, I will provide a deeper analysis of the multivariate Gaussian posterior distribution. Like in the univariate case, the posterior mean is a trade-off between the prior mean and the empirical average of the data:

$$\boldsymbol{\mu}_{\text{new}} = \Gamma\boldsymbol{\mu_0} + (\mathrm{I} - \Gamma)\frac{1}{M}\sum_{m=1}^{M}\boldsymbol{y}_m \ , \tag{62}$$

where the trade-off factor is now given by the matrix

$$\Gamma = \left(S_0^{-1} + (C/M)^{-1}\right)^{-1} S_0^{-1} \ . \tag{63}$$

This matrix summarizes the relative contribution of prior and likelihood for all possible "directions" in the $\boldsymbol{x}$ space. The trade-off factor is now a matrix instead of a scalar value since the trade-off coefficients depend on the overlap between the sub-space spanned by the measurement error and the sub-space spanned by the prior. We can understand this behavior intuitively using our array of thermometers example. in this case, a direction in the latent space is a pattern of temperatures at different locations. Now assume that the measurement errors are uncorrelated while the prior is highly correlated due to the fact that temperature in nearby skin patches tend to be similar. The update in the mean depends on the matrix product $(\mathrm{I} - \Gamma)\boldsymbol{y}_m$. Since the prior is highly correlated, this matrix acts as a filter by removing the high frequency spatial components of the readings while letting pass the low frequency spatial components. In fact, the high frequency components are likely to be due to measurement noise since we know that the underlying temperatures vary smoothly. This allows us to interpret the different directions in the latent space as different "components" of the spatial pattern of temperatures. More formally, we define a component as an eigenvector of Γ. An eigenvector of a matrix is a vector that is scaled by the matrix product without any change in direction:

$$\Gamma\boldsymbol{v}_k = \lambda_k\boldsymbol{v}_k \tag{64}$$

where the eigenvalue λ determines how much the matrix either shrinks or expands the vector. Every possible data vector $\bar{\boldsymbol{y}}$ can be decomposed into a linear combination of these eigenvectors:

$$\bar{\boldsymbol{y}} = \sum_k a_k\boldsymbol{v}_k \ , \tag{65}$$

where the coefficients a_k are the components of the expansion. This is analogous to the Fourier expansion of a signal into oscillatory components with different frequencies. An eigenvector of Γ with eigenvalue λ is also an eigenvector of $\mathrm{I} - \Gamma$ with eigenvalue $1 - \lambda$. Furthermore, it is possible to show that each eigenvalue λ is a positive number between 0 and 1. Now consider the projector matrix $\mathcal{P}_k$ which projects orthogonally each vector to the one-dimensional sub-space defined by $\boldsymbol{v}_k$. By applying this projector to equation to Eq. 62 and noticing

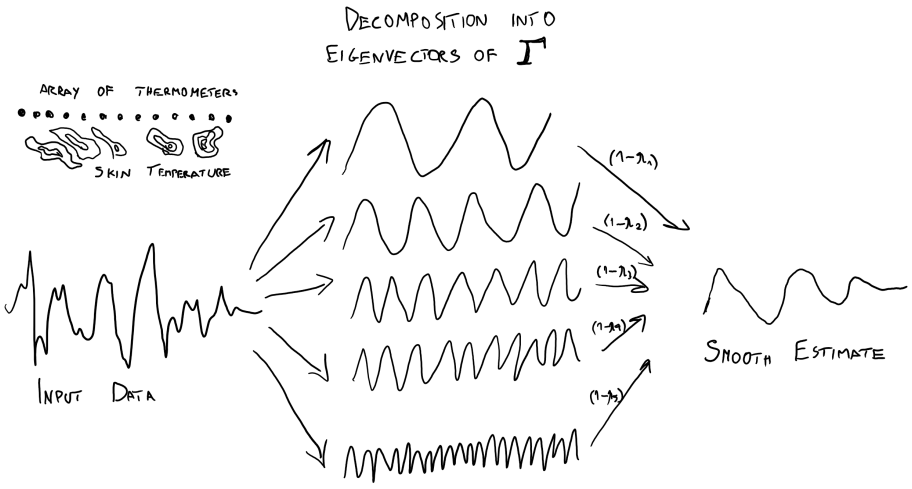

Figure 2: Multivariate Gaussian inference as a probabilistic filter.

that the matrix Γ commutes with the projector, we obtain a univariate formula for the update along the eigenvector direction:

$$\mathcal{P}_k \boldsymbol{\mu}_{\text{new}} = \lambda_k \mathcal{P}_k \boldsymbol{\mu}_{\mathbf{0}} + (1 - \lambda_k) \frac{1}{N} \sum_{m=1}^{M} \mathcal{P}_k \boldsymbol{y}_m \ . \tag{66}$$

where $0 \leq \lambda \leq 1$. This is identical to the mean update rule in univariate Bayesian inference. Now, going back to our temperature example, we can see that the eigenvectors of Γ represent spatial patterns with different spatial frequencies and that the relative eigenvalues λ are small for low frequencies and close to one for high frequencies so that the former type of pattern is filtered out from the data and filled by the prior mean while the second passes almost untouched. This filtering behavior is visualized in Fig. 2.

1.3.2 Example: Signal denoising

The filtering behavior of multivariate Gaussian inference becomes much clearer in the special case of signal denoising. Imagine a situation where the radio astronomers of the SETI project want to detect alien messages incoming from the Alpha Centauri star system. In this example, the latent vector $\boldsymbol{x}$ to collect the values of alien radio signal in a finite number of equally spaced sampled time points:

$$\boldsymbol{x} = (x_1, x_2, \ldots, x_t, \ldots x_T) \ .$$

Unfortunately, the radio-telescopes set up by the SETI team were littered by a large amount of pigeon dung (or perhaps cosmic microwave background), which resulted in a high white noise corruption in the measured signal $\boldsymbol{y}$:

$$\boldsymbol{y} = (y_1, \ldots, y_T) \ , \tag{67}$$

with

$$y_t \sim \mathcal{N}(y_t; x_t, \sigma^2) \ . \tag{68}$$

The SETI data signal processing team needs therefore to remove this white noise in order to recover the alien message and come out with the first prove of alien life.

I will now show that this denoising problem can be recasted as a multivariate Gaussian inference problem which can be solved in closed form. It is usual to express radio signals in terms of their sinusoidal components. In mathematical terms, this decomposition is given by the discrete Fourier transform:

$$x_t = \sum_{n=1}^{T/2} \left(a_n \sin(\omega_n t) + b_n \cos(\omega_n t)\right) \ , \tag{69}$$

where $\omega_n = 2\pi n/T$ is the frequency of the n-th oscillatory component of the signal and a_n and b_n determine the amplitude (and sign) of each sinusoidal component. In signal processing, it is common to model signals of this kinds as a random signals following a multivariate Gaussian distribution. This can be done by assigning an independent centered univariate Gaussian distribution to each Fourier coefficient:

$$p(a_n, b_n) = \mathcal{N}(a_n; 0, s(n))\mathcal{N}(b_n; 0, s(\omega_n)) \ . \tag{70}$$

The variance term $s(\omega_n)$ determines how much power is transmitted at the n-th frequency band. This function is referred to as the *spectral density* of the signal, which encapsulates all its statistical properties. Usually, the spectrum of the signal is assumed to be known a priori. The Fourier transform in Eq. 69 can be easily re-expressed in matrix notation. First, we note that the sine and cosine components can be expressed using vectors:

$$\boldsymbol{v}_n = (\sin(\omega_1 1), \ldots, \sin(\omega_n t), \ldots, \sin(\omega_n T)) \ , \tag{71}$$

$$\boldsymbol{w}_n = (\cos(\omega_1 1), \ldots, \cos(\omega_n t), \ldots, \cos(\omega_n T)) \ . \tag{72}$$

These components can then be rearranged into the columns of a matrix U. Similarly, we can concatenate the Fourier coefficients a_n and b_n into a vector $\boldsymbol{z}$ keeping in mind that we need to use the same order that we used for composing the matrix U. Using this notation, the Fourier transform becomes a simple matrix multiplication:

$$\boldsymbol{x} = U\boldsymbol{z} \ . \tag{73}$$

it is now easy to see that the vector $\boldsymbol{x}$ follows the multivariate Gaussian distribution

$$p(\boldsymbol{x}) = \mathcal{N}(\boldsymbol{x}; \boldsymbol{0}, USU^T) \ , \tag{74}$$

where the *cross-spectral density matrix* D has a diagonal form with the spectrum $s(\omega_n)$ in its diagonal. Importantly, since the sine and cosine components are mutually orthogonal, U is an orthogonal matrix, meaning that $U^T U = UU^T = I$. This implies that the vectors $\boldsymbol{v}_n$ and $\boldsymbol{w}_n$ are all eigenvectors of the covariance

matrix with eigenvalue $s(\omega_n)$. This reflects the fact that we assumed the Fourier coefficient to be independent random variables. While this can initially seem as a rather arbitrary choice, it is possible to show that all translationally invariant periodic signals can be expressed in this form.
We now expressed the denoising problem in a standard multivariate Gaussian inference form since both the prior over the signal and the likelihood are multivariate Gaussian distributions. We can noise obtain the distribution over the denoised signal by using the closed form formula (Eq. 62) that we obtained in the previous section. First of all, we need to evaluate the matrix Γ. In this case, the matrix is given by:

$$\begin{aligned}\Gamma &= \left((UDU^T)^{-1} + \sigma^{-2} I\right)^{-1} (UDU^T)^{-1} \\ &\quad - \left(UD^{-1}U^T + \sigma^{-1} I\right)^{-1} UD^{-1}U^T \\ &= U(D^{-1} + \sigma^{-2} I)^{-1} U^T U D^{-1} U^T \\ &= U(D^{-1} + \sigma^{-2} I)^{-1} D^{-1} U^T \ ,\end{aligned} \tag{75}$$

where we used the fact that $I = UU^T$ since U is an orthogonal matrix. Since $(D^{-1} + \sigma^{-2} I)^{-1} D^{-1}$ is diagonal, the matrix Γ has eigenvectors $\boldsymbol{v}_n$ and $\boldsymbol{w}_n$ with eigenvalues

$$\lambda_n = \frac{s(\omega_n)^{-1}}{s(\omega_n)^{-1} + \sigma^{-2}} \ . \tag{76}$$

We can now use Eq. 66 to analyze the posterior Fourier components as function of the Fourier components of the data. In fact, the projector in Eq. 66 is in this case just the Fourier transform. Consequently, each Fourier component of the data is processed independently by the Bayesian analysis. Consider the (sine) Fourier coefficient $\tilde{y}_n$ of the measured signal:

$$\tilde{y}_n = \sum_{t=1}^{T} \sin(\omega_n t) y_t \ . \tag{77}$$

Since the analysis treats the Fourier coefficients independently, the posterior over the signal Fourier coefficients is simply the product of the individual posterior components:

$$p(a_1, \ldots, a_n, \ldots, b_n, \cdots \mid \boldsymbol{y}) = \prod_{n=1}^{T/2} \mathcal{N}(a_n, \tilde{\mu}_{\text{new}}^{(a_n)}, {\sigma_{\text{new}}^{(a_n)}}^2) \mathcal{N}(b_n, \tilde{\mu}_{\text{new}}^{(b_n)}, {\sigma_{\text{new}}^{(b_n)}}^2) \ . \tag{78}$$

The parameters of the posterior over each of the Fourier coefficient can now be computed as univariate Gaussian updates. For example, the mean of the n-th sine component can be obtained by applying Eq. 66 as follows:

$$\begin{aligned}\tilde{\boldsymbol{\mu}}_{\text{new}}^{a_n} &= \lambda_k 0 + (1 - \lambda_k)\tilde{y}_n \\ &= \frac{\sigma^{-2}}{s(\omega_n)^{-1} + \sigma^{-2}} \tilde{y}_n \ .\end{aligned} \tag{79}$$

Now imagine a situation where the signal spectral density $s(\omega_k)$ is much larger than the than the noise variance in a small band $(\omega_{\min}, \omega_{\max})$ while being much smaller than the noise floor outside this band. In this case, $1 - \lambda_n \approx 1$ when $\omega_n \in (\omega_{\min}, \omega_{\max})$ while $1 - \lambda_n \approx 0$ otherwise. In this scenario, Eq. preserves the Fourier coefficients of the data inside the band while it forces the coefficients to zero outside it as they are probably dominated by the noise. In signal processing, this is known as a band pass filter.

1.4 Multivariate Gaussian inference with linear observation models

So far, we only considered inference problems where the latent $\boldsymbol{x}$ gives directly the mean of the measurements vector $\boldsymbol{y}$. However, very often the mean of the data is given by a linear transformation of the latent variables. The use of linear models allows us to use a latent space with different dimensionality than the number of measured quantities. This is very important in many machine learning problems as we often wish to encode the data into a small number of latent features. Consider a d_y dimensional measurement vector $\boldsymbol{y}$ and a d_x dimensional latent vector $\boldsymbol{x}$. The latent is mapped to the mean of the measurement through a linear operator represented by a $d_y \times d_x$ matrix A. This result in a multivariate Gaussian likelihood:

$$p(D \mid \boldsymbol{x}) = \prod_{m=1}^{M} \mathcal{N}(\boldsymbol{y}_m; A\boldsymbol{x}, C) \ . \tag{80}$$

Assuming the usual multivariate prior $\mathcal{N}(\boldsymbol{x}; \boldsymbol{\mu}_0, S_0)$, we can compute the posterior using the usual approach. First, we evaluate the joint density up to any multiplicative term that does not depend on $\boldsymbol{x}$:

$$\mathcal{N}(x \mid D) \propto \exp\left(-\frac{1}{2}\boldsymbol{x}^T S_0^{-1}\boldsymbol{x} + (A\boldsymbol{x})^T S_0^{-1}\boldsymbol{\mu} + \boldsymbol{x}^T C^{-1} \sum_{m=1}^{M} \boldsymbol{y}_m - (A\boldsymbol{x})^T C^{-1} (A\boldsymbol{x})\right) , \tag{81}$$

$$= \exp\left(-\frac{1}{2}\boldsymbol{x}^T \left(S_0^{-1} + A^T (C/M)^{-1} A\right)\boldsymbol{x} + \boldsymbol{x}^T \left(S_0^{-1}\boldsymbol{\mu} + A^T (C/M)^{-1} \frac{1}{M} \sum_{m=1}^{M} \boldsymbol{y}_m\right)\right) .$$

Again, we obtained an expression proportional to a multivariate Gaussian in canonical form with natural parameters

$$\boldsymbol{\eta}_A = S_0^{-1}\boldsymbol{\mu} + A^T (C/M)^{-1} \frac{1}{M} \sum_{m=1}^{M} \boldsymbol{y}_m \tag{82}$$

and

$$\Omega_A = S_0^{-1} + A^T (C/M)^{-1} A \ . \tag{83}$$

the posterior can the be re-expressed in terms of the mean vector and covariance matrix:

$$\boldsymbol{\mu}_{\text{new}} = (S_0^{-1} + A^T(C/M)^{-1}A)^{-1}\left(S_0^{-1}\boldsymbol{\mu} + A^T(C/M)^{-1}\frac{1}{M}\sum_{m=1}^{M}\boldsymbol{y}_m\right) \quad (84)$$

and

$$S_{\text{new}} = (S_0^{-1} + A^T(C/M)^{-1}A)^{-1} \ . \quad (85)$$

This formulae for the posterior are valid for all possible dimensionalities d_y and d_x. However, the nature of the solution and its interpretation changes dramatically depending on the number of latent and measured dimensions. The case where d_x is smaller than d_y is said to be over-constrained. In this scenario, as far as the matrix A is full rank, the data provides enough information to fully identify the latent vector $\boldsymbol{x}$. It is possible to re-write the posterior mean in Eq. 84 in a more compact form as a multivariate convex combination of the prior and the least squares fit of the data:

$$\boldsymbol{\mu}_{\text{new}} = \Gamma_A\boldsymbol{\mu_0} + (\mathrm{I} - \Gamma_A)\frac{1}{M}\sum_{m=1}^{M}A^+\boldsymbol{y}_m \ , \quad (86)$$

where the Moore–Penrose pseudo-inverse matrix $A^+ = A^T(AA^T)^{-1}$ projects the data into the least-squares fit in the lower dimensional latent space $\boldsymbol{x}$. In this expression, the trade-off factor is now given by the matrix

$$\Gamma_A = \left(S_0^{-1} + A^T(C/M)^{-1}A\right)^{-1}S_0^{-1} \ . \quad (87)$$

As you can see, the over-constrained case is very similar to the case without observation model. The only difference is that the both the data and the noise precision matrix (i.e. the inverse of the covariance matrix) is first projected into the latent space. Note that, if the dimensionality of the data is much higher than the dimensionality of the latent space, the measurement error will tent to be very low since many measurements provide information about each latent variable. The situation is very different for $d_x > d_y$. In this case, the data cannot provide full information about the latent since the matrix A necessarily has a non-empty null space comprised by vectors $\boldsymbol{x}_{\text{null}}$ that are annihilated by the transformation: $A\boldsymbol{x}_{\text{null}} = 0$. Clearly, the data does not provide any information concerning the component of the latent in the null space since any such component would not influence the measurements. We can also define the column space as the orthogonal complement of the null space. By definition, any change in the column space produces an observable shift in the likelihood that can be detected given a large enough number of data points. However, the measurements can still indirectly provide some level of information concerning the null space if there are correlation in the prior distribution. In fact, the under-constrained posterior can be expressed as

$$p(\boldsymbol{x} \mid D) = p(\boldsymbol{x}_{\text{col}}, \boldsymbol{x}_{\text{null}} \mid D) = p(\boldsymbol{x}_{\text{null}} \mid \boldsymbol{x}_{\text{coll}})p(\boldsymbol{x}_{\text{coll}} \mid D) \ , \quad (88)$$

where we use the fact that the likelihood does not depend on $\boldsymbol{x}_{\text{col}}$. The column space posterior distribution can be found by defining the new latent d_y dimensional vector $\boldsymbol{z} = A\boldsymbol{x}$, which follows the prior distribution $\mathcal{N}(A\boldsymbol{\mu}_0, AS_0A^T)$. In fact, in this new variable the problem reduces to the standard multivariate inference without linear observation models.
Note that for the number of datapoints M tending to infinity, this posterior converges to a deterministic distribution $\delta(\boldsymbol{x}_{\text{col}} - \boldsymbol{x}^*_{\text{col}})$. On the other hand, the (marginal) null space posterior converges to

$$p(\boldsymbol{x}_{\text{null}} \mid D) = \int p(\boldsymbol{x}_{\text{null}} \mid \boldsymbol{x}_{\text{coll}})\delta(\boldsymbol{x}_{\text{col}} - \boldsymbol{x}^*_{\text{col}})\mathrm{d}\boldsymbol{x}_{\text{col}} = p(\boldsymbol{x}_{\text{null}} \mid \boldsymbol{x}^*_{\text{col}}) \ . \tag{89}$$

Since the distribution $p(\boldsymbol{x}_{\text{null}} \mid \boldsymbol{x}^*_{\text{col}})$ is usually not deterministic, we arrive at the conclusion that it is not possible to fully reduce the uncertainly concerning the null space variables even at the limit of infinite data. This should not be surprising, you can not expect to learn the DNA composition of your food just by tasting it, even if you do it a very large number of times. Real world measurement devices, such as the taste buds in our tongue, cannot usually extract full information about a system. In the mathematical formalism of Bayesian inference, this results in a likelihood function that does not depend on certain latent features.

1.4.1 Example 1: Spatial extrapolation with Gaussian process regression

We can now re-frame our example with the spatial array of thermometers using the elegant framework of Gaussian process regression. Consider a set $(\boldsymbol{x}_1, \dots, \boldsymbol{x}_k, \dots, \boldsymbol{x}_J)$ of spatial locations dispersed throughout Europe. These are the coordinates of meteorological stations each of which measure the local temperature. We can assume that the k-th measurement have the true temperature $f(\boldsymbol{x}_k)$ as mean and the error variance σ^2. We are not really interested in the value of the temperature at the stations themselves. Instead, we would like to extrapolate these measurements to a set of interesting locations $(x^*_1, \dots, x^*_L)$ corresponding for example to city centers, stadiums, amusement parks and so on. We can frame this extrapolation as a Gaussian inference problem by modeling the temperature values at both station and interesting locations using a multivariate Gaussian distribution:

$$p(\boldsymbol{f}) = \mathcal{N}(\boldsymbol{f}_{\text{tot}}; \boldsymbol{\mu}_{\text{tot}}, K_{\text{tot}}) \ , \tag{90}$$

where we collected all the values in the vector $\boldsymbol{f}_t exttot = (\boldsymbol{f}(x_1), \dots, f(x^*_1), \dots)$. In this expression, the total mean vector is the concatenation of the measurement locations mean vector $\boldsymbol{f}$ at the "interesting" locations mean vector $\boldsymbol{f}_*$. Similarly, the covariance matrix K_{tot} can be written in block form as follows

$$K_{\text{tot}} = \begin{pmatrix} K & K_* \\ K_*^T & K_{**} \end{pmatrix} \tag{91}$$

where K is the covariance matrix restricted to the measurement locations, K_{**} is the covariance matrix of the temperatures at the "interesting" locations and K_* is the cross covariance matrix, whose entries provide the temperature covariances between each pair of measurement location and "interesting" location. Since the values of the temperature values vary spatially, we can now write the parameter of this prior as an explicit function of the location. For example, if the j-th value is associated to the location x_j, we obtain its mean as the output of a mean function $\mu(x_k)$ defined for all possible locations. Similarly, we express the covariance K_{ij} between any two points $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$ using a covariance function $k(\boldsymbol{x}_i, \boldsymbol{x}_j)$. Usually, the covariance function is a function of the distance $||\boldsymbol{x}_i - \boldsymbol{x}_j||$. A commonly used covariance function is the squared exponential:

$$k(x, x') = e^{-\frac{1}{2l^2}\sum(x-x')^2} \tag{92}$$

where l is the characteristic length scale of the latent function. Roughly speaking, this covariance functions says that the values of the temperature at two spatial locations are correlated when their distance is smaller than l while their correlation is negligible otherwise. Be careful, the covariance function should not be confused with a Gaussian probability density. It just gives the spatial structure of the correlations and it should not be interpreted as a probability distribution. The mean function usually plays a less important role in a Gaussian process regression. In this case, we set it to be equal to the vector of historical average temperature $\boldsymbol{t}_{\text{avg}}$. We can now introduce a set of temperature measurements $\boldsymbol{y} = (y_1, \ldots, y_J)$. By definition of the problem, the temperature is only measured at the locations $(x_1, \ldots, x_J)$. This can be modeled using the following observation model:

$$A = \begin{pmatrix} I & 0 \\ 0 & 0 \cdot \end{pmatrix} \tag{93}$$

Clearly, the null space of this model is given by the values of the temperature at the "interesting" locations while the column space is given by the values at the location of the meteorological stations. Temperature measurements tend to be very precise. We can therefore assume the posterior distribution at the measurement locations to be a deterministic distribution centered at the true temperature values $\boldsymbol{y}$. In the Gaussian process regression literature, this is referred to as the *noiseless case*. We can now express the posterior distribution over the "interesting" locations using Eq. 43:

$$\mu_{*\text{new}} = \boldsymbol{t}_{\text{avg}} + K_* K^{-1}(\boldsymbol{y} - \boldsymbol{t}_{\text{avg}}) \tag{94}$$

$$K_{*\text{new}} = K_{**} - K_*^T K^{-1} K_* \ . \tag{95}$$

Remember that the entry of both covariance and cross-covariance matrices depend on the covariance function $k(\boldsymbol{x}, \boldsymbol{x}')$, whose values decay to zero when the distance is larger than the length scale. Consequently, the posterior at locations distant from any meteorological station reduces to the prior historical values. Conversely, the posterior is highly influenced by the measurements in locations close to many stations.

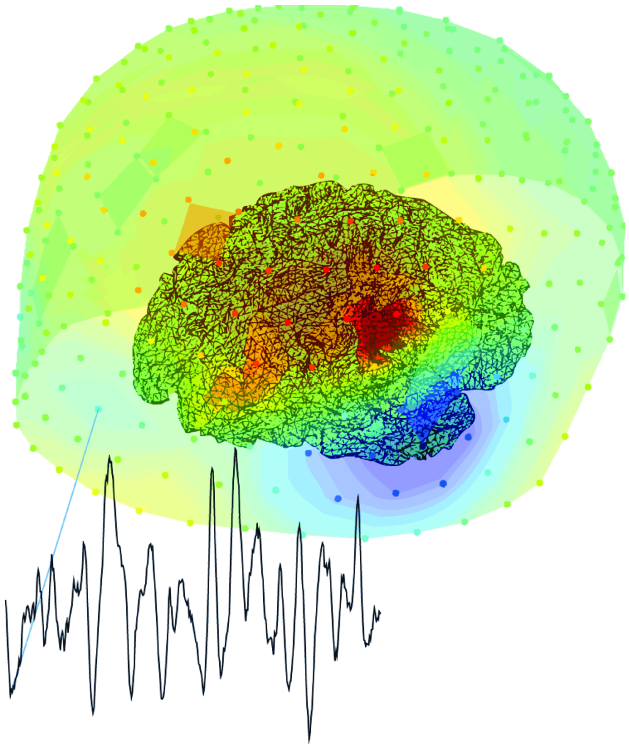

Figure 3: Electric activity in the brain generate MEG signals on the head. We can use these readings to infer the location of brain activity.

1.4.2 Example 2: MEG source reconstruction

Another interesting case of under-constrained multivariate Gaussian inference is involved in the analysis of neural data. Magnetoencephalography (MEG) devices record magnetic fields in the head originating from the human brain. A MEG measurement is comprised by several hundreds of channels, each recording the magnetic field in a given location of the head. We denote these measurements at a given time point as $\boldsymbol{y}$. The goal of neuroscientists is to use these magnetic fields to infer the electrical activity of the brain under different cognitive conditions. We can organize the (sub-threshold) electrical activity of the brain cortex into a vector $\boldsymbol{x}$. Each entry of this vector gives the electrical potential in a small area of the cortex. The brain activity $\boldsymbol{x}$ generates MEG measurement through a linear mapping G, which can be computed by analyzing the geometric and magnetic properties of brain, skull and skin. this is visualized in Fig. 3. Therefore, we can write

$$\boldsymbol{y} = G\boldsymbol{x} + \boldsymbol{\epsilon} \ , \tag{96}$$

where $\boldsymbol{\epsilon}$ is centered measurement noise with covariance C. The matrix G, also known as the *lead field*, is known in advance as it can be computed from structural

brain scans using magnetostatic calculations. our goal is now to compute the posterior distribution over the brain signal $\boldsymbol{x}$ by mapping the measurements on the brain cortex. As you can see, if we assign a multivariate Gaussian prior to $\boldsymbol{x}$, this is the kind of Gaussian inference problems that we solved in the previous sections. Note that usually the number of relevant brain areas is much higher than the number of MEG channels, so that the resulting inference is under-constrained. In this case, usually all brain location can generate some change in the MEG reading. However, several brain configuration can exactly cancel. These self canceling patterns, where the magnetic signal of a brain region is counteracted by the magnetic signal of another region, are part of the null space of G. Remember that in the under-constrained case the final result depends on the prior distribution even at the limit of infinitely many datapoints. It is therefore crucial to assign a proper prior that respect the underlying biology of the brain. For example, we can use a prior covariance matrix S_0 that assigns high correlations to nearby cortical locations since brain activity tend to vary continuously across the cortex.

1.5 Bayesian linear regression

We can now move to one of the most important case of tractable inference: Bayesian linear regression. In a linear regression model, the output variables y_n is assumed to follow a normal distribution whose mean is a linear combination of the vector of predictors $\boldsymbol{x}_n$:

$$y_n \sim \mathcal{N}\big(\boldsymbol{w}^T \boldsymbol{x}_n, \sigma^2\big) \ . \tag{97}$$

We assign an uncoupled (conjugate) normal prior to each of the weights vector w:

$$\boldsymbol{w} \sim \mathcal{N}\big(0, \nu^2 \mathrm{I}\big) \ , \tag{98}$$

where I is the identity matrix. We can compute the posterior over the weights by evaluating the joint density (product of likelihood and prior) up to terms the do not depend on w:

$$p(\boldsymbol{w} \mid D) \propto \exp\left(-\frac{1}{2\nu^2}\boldsymbol{w}^T\boldsymbol{w}\right) \prod_n^N \exp\left(-\frac{1}{2\sigma^2}\left(y_n - \boldsymbol{w}^T\boldsymbol{x}_n\right)^2\right) \tag{99}$$

$$= \exp\left(-\frac{1}{2\sigma^2}\sum_n^N \left(y_n - \boldsymbol{w}^T\boldsymbol{x}_n\right)^2 - \frac{1}{2\nu^2}\boldsymbol{w}^T\boldsymbol{w}\right) \tag{100}$$

$$\propto \exp\left(\boldsymbol{w}^T\left(\frac{1}{\sigma^2}\sum_n^N y_n\boldsymbol{x}_n\right) - \frac{1}{2}\boldsymbol{w}^T\left(\sigma^{-2}\sum_n^N \boldsymbol{x}_n\boldsymbol{x}_n^T + \nu^{-2}I\right)\boldsymbol{w}\right) . \tag{101}$$

The list line of the derivations shows has the explicit form of a (non-normalized) multivariate normal density with the following natural parameters:

$$\boldsymbol{\eta}_+ = -\frac{1}{\sigma^2}\sum_n^N y_n \boldsymbol{x}_n \tag{102}$$

$$\Lambda_+ = \sigma^{-2}\left(\sum_n^N \boldsymbol{x}_n \boldsymbol{x}_n^T + \frac{\sigma^2}{\nu^2}\right) \ . \tag{103}$$

Therefore, the posterior over the weights follows the multivariate normal distribution

$$\boldsymbol{w} \sim \mathcal{N}\big(0, \nu^2 \mathrm{I}\big) \ , \tag{104}$$

with

$$\boldsymbol{m}_+ = -\frac{1}{2}\Lambda^{-1}\boldsymbol{\eta} = \left(\sum_n^N \boldsymbol{x}_n \boldsymbol{x}_n^T + \frac{\sigma^2}{\nu^2}\right)^{-1}\left(\sum_n^N y_n \boldsymbol{x}_n\right) \ , \tag{105}$$

$$\Sigma_+ = \sigma^2\left(\sum_n^N \boldsymbol{x}_n \boldsymbol{x}_n^T + \frac{\sigma^2}{\nu^2}\right)^{-1} \ . \tag{106}$$

Given a target predictor $\boldsymbol{x}_*$, the optimal predictive distribution given the linear model and the dataset D is

$$y_* \sim \mathcal{N}\big(\boldsymbol{m}_+^T \boldsymbol{x}_n, \boldsymbol{x}_*^T \Lambda_+^{-1} \boldsymbol{x}_* + \sigma^2\big) \ . \tag{107}$$