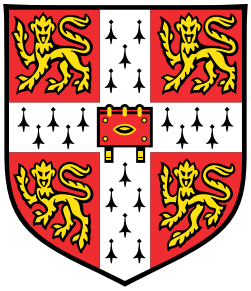
Statistical methods for the integrative analysis of single-cell multi-omics data



Ricard Argelaguet

European Bioinformatics Institute
University of Cambridge

This dissertation is submitted for the degree of
Doctor of Philosophy

Robinson College

April 2020

0.1 Theoretical foundations

0.2 Probabilistic modelling

A scientific model is a simple theoretical representation of a complex natural phenomenon to allow the systematic study of its behaviour. The general idea is that if a model is able to explain some observations, it might be capturing its true underlying laws and can therefore be used to make future predictions.

In particular, statistical models are a powerful abstraction of nature. They consist on a set of observed variables and a set of (hidden) parameters. The procedure of fitting the parameters using a set of observations is called inference or learning.

One of the major challenges of inference when dealing with real data sets is the distinction between signal and noise. An ideal model should learn only the information relevant to gain explanatory power while disregarding the noise. However, this is a non-trivial task in most practical situations. Very complex models will tend to overfit the training data, capturing large amounts of noise and consequently leading to a bad generalisation performance to independent data sets. On the other hand, simplistic models will fit the data poorly, leading to poor explanatory power.

The ideas above can be formalised using the framework of probability and statistics.

0.2.1 Maximum likelihood inference

A common approach is to define a statistical model of the data $\mathbf{Y}$ with a set of parameters $\boldsymbol{\theta}$ that define a probability distribution $p(\mathbf{Y}|\boldsymbol{\theta})$, called the likelihood function. A simple approach to fit a model is to estimate the parameters $\hat{\boldsymbol{\theta}}$ that maximise the likelihood:

$$\hat{\boldsymbol{\theta}} = \arg\max_{\boldsymbol{\theta}} p(\mathbf{Y}|\boldsymbol{\theta})$$

This process is called maximum likelihood learning[219, 20, 165]. However, in this setting there is no penalisation for model complexity, making maximum likelihood solutions prone to overfit in cases where the data is relatively sparse. Generalisations that account for model complexity have been proposed and include regularising terms that shrink parameters to small values. However, these are often particular cases of the more general framework of Bayesian statistics [85, 20, 165].

0.2.2 Bayesian inference

In the Bayesian framework, the parameters themselves are treated as random unobserved variables and we aim to obtain probability distributions for $\boldsymbol{\theta}$, rather than a single point estimate. To do so, prior beliefs are introduced into the model by specifying a prior probability distribution $p(\boldsymbol{\theta})$. Then, using Bayes' theorem [13], the prior hypothesis is updated based on the observed data $\mathbf{Y}$ by means of the likelihood $p(\mathbf{Y}|\boldsymbol{\theta})$ function, which yields a posterior distribution over the parameters:

$$p(\boldsymbol{\theta}|\mathbf{Y}) = \frac{p(\mathbf{Y}|\boldsymbol{\theta})p(\boldsymbol{\theta})}{p(\mathbf{Y})}$$

where $p(\mathbf{Y})$ is a constant term called the marginal likelihood, or model evidence [20, 165].
The choice of the prior distribution is a key part of Bayesian inference and captures beliefs about the distribution of a variable before the data is taken into account. With asymptotically large sample sizes, the choice of prior has negligible effects on the posterior estimates, but it becomes critical with sparse data [20, 165, 72].

There are two common considerations when defining the prior distributions. The first relates to the incorporation of subjective information, or predefined assumptions, into the model. For example, one could adapt the prior distribution to match the results from previous experiments (i.e. an informative prior). Alternatively, if no information is available one could set set uninformative priors by following maximum entropy principles [100].

The second strategy is based on convenient mathematical properties to make inference tractable. If the likelihood and the prior distributions do not belong to the same family of probability distributions (they are not conjugate) then inference becomes more problematic [185, 20, 165, 72]. The existence of conjugate priors is one of the major reasons that justify the widespread use of exponential family distributions in Bayesian models [72]. An example is the Automatic Relevance Determination prior discussed in **??**.

Again, the milestone of Bayesian inference is that an entire posterior probability distribution is obtained for each unobserved variable. This has the clear advantage of naturally handling uncertainity in the estimation of parameters. For instance, when making predictions, a fully Bayesian approach attempts to integrate over all the possible values of all unobserved varaibles, effectively propagating uncertainity across multiple layers of the model. Nevertheless, this calculation is sometimes intractable and one has to resort to point estimates [20, 165, 72]. The simplest approximation to the posterior distribution is to use its mode, which leads to the maximum a posteriori (MAP) estimate:

$$\hat{\boldsymbol{\theta}} = \arg\max_{\boldsymbol{\theta}} p(\boldsymbol{\theta}) p(\mathbf{Y}|\boldsymbol{\theta})$$

This is similar to the maximum likelihood objective function, but with the addition of a term $p(\boldsymbol{\theta})$. When the prior dsitribution is chosen smartly, this term penalises for model complexity. Therefore, in contrast to standard (non-penalised) maximum likelihood inference, Bayesian approaches naturally handle the problem of model complexity and overfitting[20, 165, 72]. At the limit of infinite observations, the influence of the prior to the posterior is negligible and the MAP estimate converges towards the Maximum likelihood estimate, hence providing a rational link between the two inference frameworks.

0.2.3 Deterministic approaches for Bayesian inference

The central task in Bayesian inference is the direct evaluation of the posterior distributions and/or the computation of expectations with respect to the posterior distributions. In sufficiently complex models, closed-form solutions are not available and one has to resort to approximation schemes, which broadly fall into two classes: stochastic or deterministic [72, 22].

Stochastic approaches hinge on the generation of samples from the posterior distribution via a Markov Chain Monte Carlo (MCMC) framework. Such techniques have the appealing property of generating exact results at the asymptotic limit of infinite computational resources. However, in practice, sampling approaches are computationally demanding and suffer from limited scalability to large data sets [22].
In contrast, deterministic approaches are based on analytical approximations to the posterior distribution, which often lead to biased results. Yet, given the appropriate settings, these approaches are potentially much faster and scalable to large applications [20, 165, 22].

0.2.3.1 Laplace approximation

The Laplace approximation is probably the simplest of the deterministic tecniques, where the aim is to construct a Gaussian approximation around the mode of the true posterior distribution using a second-order Taylor expansion [20, 165].
Suppose $\mathbf{X}$ contains all unobserved variables. The true posterior distribution can be written as:

$$p(\mathbf{X}) = \frac{f(\mathbf{X})}{Z}$$

where $f(\mathbf{X})$ is a function that depends on the unobserved variables and Z is an unknown normalisation constant to ensure that $\int p(\mathbf{X}) d\mathbf{X} = 1$.

The second-order Taylor expansion of $\log f(\mathbf{X})$ centered around its (known) mode $\hat{\mathbf{X}}$ is:

$$\log f(\mathbf{X}) \approx \log f(\hat{\mathbf{X}}) - \frac{1}{2}(\mathbf{X} - \hat{\mathbf{X}})^T \mathbf{A} (\mathbf{X} - \hat{\mathbf{X}})$$

where $\mathbf{A} = \nabla^2 \log f(\hat{\mathbf{X}})$ is the Hessian matrix of $\log f(\mathbf{X})$ evaluated at $\hat{\mathbf{X}}$.
Notice three things. First, the first-order term of the Taylor expansion is zero because $\hat{\mathbf{X}}$ is a stationary point. Second, the log function is monotonically increasing and therefore a maximum of $\log f(\mathbf{X})$ is also a maximum of $f(\mathbf{X})$. Third, the mode of the posterior $p(\mathbf{X})$ must be known, which requires the use of (complex) optimisation algorithms.
Taking the exponential in both sides:

$$f(\mathbf{X}) \approx f(\hat{\mathbf{X}}) \exp\{-\frac{1}{2}(\mathbf{X} - \hat{\mathbf{X}})^T \mathbf{A} (\mathbf{X} - \hat{\mathbf{X}})\}$$

which leads to the following multivariate Gaussian distribution approximation $q(\mathbf{X}) = \mathcal{N}\left(\mathbf{X} \,|\, \hat{\mathbf{X}}, \mathbf{A}\right)$:

$$q(\mathbf{X}) = \frac{|\, A \,|^{1/2}}{(2\pi^{d/2})} \exp\{-\frac{1}{2}(\mathbf{X} - \hat{\mathbf{X}})^T \mathbf{A} (\mathbf{X} - \hat{\mathbf{X}})\}$$

where d is the number of unobserved variables.
Despite its simplicity, the Laplace approximation is a useful strategy that has been successfully applied in practice. Nonetheless, this approximation has notable caveats: first, is limited by its own local definition, ignoring all the density beyond the mode of the posterior. Second, it does not

apply to discrete variables. Third, the inversion of the Hessian is very expensive in high-dimensional settings.

0.2.4 Variational inference

Variational inference is a deterministic family of methods that have been receiving widespread attention due to a positive balance between accuracy, speed, and ease of use [22, 248]. The core framework is derived below.

In variational inference the true (but intractable) posterior distribution $p(\mathbf{X}|\mathbf{Y})$ is approximated by a simpler (variational) distribution $q(\mathbf{X}|\mathbf{\Theta})$ where $\mathbf{\Theta}$ are the corresponding parameters. The parameters, which we will omit from the notation, need to be tuned to obtain the closest approximation to the true posterior.
The distance between the true distribution and the variational distribution is calculated using the KL divergence:

$$\mathrm{KL}(q(\mathbf{X})||p(\mathbf{X}|\mathbf{Y})) = -\int_z q(\mathbf{X}) \log \frac{p(\mathbf{X}|\mathbf{Y})}{q(\mathbf{X})}$$

Note that the KL divergence is not a proper distance metric, as it is not symmetric. In fact, using the reverse KL divergence $\mathrm{KL}(q(\mathbf{X})||p(\mathbf{X}|\mathbf{Y}))$ defines a different inference framework called expectation propagation [156].

If we allow any possible choice of $q(\mathbf{X})$, then the minimum of this function occurs when $q(\mathbf{X})$ equals the true posterior distribution $p(\mathbf{X}|\mathbf{Y})$. Nevertheless, since the true posterior is intractable to compute, this does not lead to any simplification of the problem. Instead, it is necessary to consider a restricted family of distributions $q(\mathbf{X})$ that are tractable to compute and subsequently seek the member of this family for which the KL divergence is minimised.

Doing some calculus it can be shown (see [20, 165]) that the KL divergence $\mathrm{KL}(q(\mathbf{X})||p(\mathbf{X}|\mathbf{Y}))$ is the difference between the log of the marginal probability of the observations $\log(\mathbf{Y})$ and a term $\mathcal{L}(\mathbf{X})$ that is typically called the Evidence Lower Bound (ELBO):

$$\mathrm{KL}(q(\mathbf{X})||p(\mathbf{X}|\mathbf{Y})) = \log(\mathbf{X}) - \mathcal{L}(\mathbf{X})$$

Hence, minimising the KL divergence is equivalent to maximising $\mathcal{L}(\mathbf{X})$ Figure 0.1:

$$\begin{aligned}\mathcal{L}(\mathbf{X}) &= \int q(\mathbf{X}) \Big(\log \frac{p(\mathbf{X}|\mathbf{Y})}{q(\mathbf{X})} + \log p(\mathbf{Y}) \Big) d\mathbf{X} \\ &= \mathbb{E}_q[\log p(\mathbf{X}, \mathbf{Y})] - \mathbb{E}_q[\log q(\mathbf{X})]\end{aligned} \tag{1}$$

The first term is the expectation of the log joint probability distribution with respect to the variational distribution. The second term is the entropy of the variational distribution. Importantly, given a simple parametric form of $q(\mathbf{X})$, each of the terms in Equation (1) can be computed in closed form.
In some occasions (see section X), we will use the following form for the ELBO:

$$\mathcal{L}(\mathbf{X}) = \mathbb{E}_q[\log p(\mathbf{Y}|\mathbf{X})] + (\mathbb{E}_q[\log p(\mathbf{X})] - \mathbb{E}_q[\log q(\mathbf{X})]) \tag{2}$$

where the first term is the expectation of the log likelihood and the second term is the difference in the expectations of the p and q distributions of each hidden variable.

In conclusion, variational learning involves minimising the KL divergence between $q(\mathbf{X})$ and $p(\mathbf{X}|\mathbf{Y})$ by instead maximising $\mathcal{L}(\mathbf{X})$ with respect to the distribution $q(\mathbf{X})$. The following image summarises the general picture of variational learning:

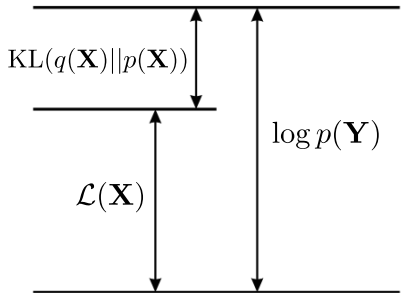


Figure 0.1: The quantity $\mathcal{L}(\mathbf{X})$ provides a lower bound on the true log marginal likelihood $\log p(\mathbf{Y})$, with the difference being given by the Kullback-Leibler divergence $\mathrm{KL}(q||p)$ between the variational distribution $q(\mathbf{X})$ and the true posterior $p(\mathbf{X}|\mathbf{Y})$

Yet, there are several approaches to define $q(\mathbf{X})$. The two most commonly used are called (unparametric) mean-field and (parametric) fixed-form [248, 22].

0.2.4.1 Mean-field variational inference

The most common type of variational Bayes, known as the mean-field approach, assumes that the variational distribution factorises over M disjoint groups of unobserved variables[205]:

$$q(\mathbf{X}) = \prod_{i=1}^{M} q(\mathbf{x}_i) \tag{3}$$

where typically all unobserved variables are assumed to be independent. Importantly, notice that no parametric assumptions were placed regarding the nature of $q(\mathbf{x}_i)$.

Evidently, in sufficiently complex models where the unobserved variables have dependencies this family of distributions do not contain the true posterior (Figure 0.2). Yet, this is a key assumption to obtain an analytical inference scheme that yields surprisingly accurate results [21, 63, 25].

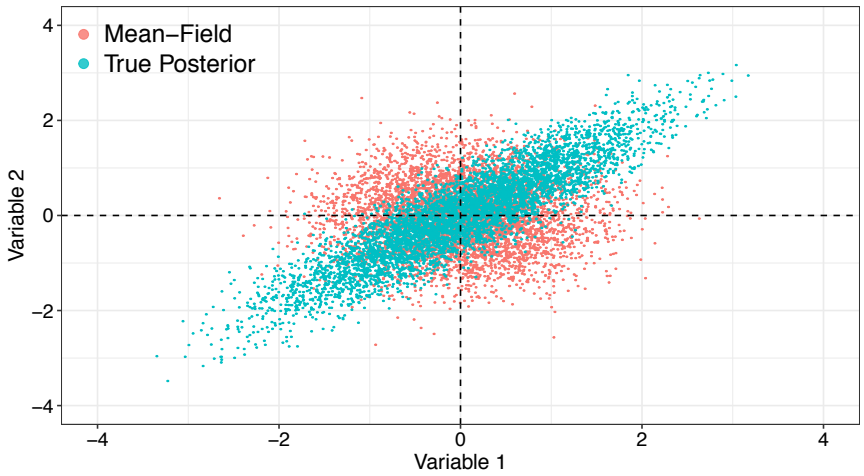


Figure 0.2: Illustrative example of sampling from a true posterior distribution (blue) versus a fitted mean-field varaitional distribution (red) in a model with two (correlated) unobserved variables. The mean-field approximation wrongly assumes that the unobserved variables are independent.

Using calculus of variations (derivations can be found in [20, 165]), it follows that the optimal distribution $q(\mathbf{X})$ that maximises the lower bound $\mathcal{L}(\mathbf{X})$ is

$$\log \hat{q}_i(\mathbf{x}_i) = \mathbb{E}_{-i}[\log p(\mathbf{Y}, \mathbf{X})] + \text{const} \tag{4}$$

where $\mathbb{E}_{-i}$ denotes an expectation with respect to the q distributions over all variables $\mathbf{x}_j$ except for $\mathbf{x}_i$.
The additive constant is set by normalising the distribution $\hat{q}_i(\mathbf{z}_i)$:

$$\hat{q}(\mathbf{x}_i) = \frac{\exp(\mathbb{E}_{-i}[\log p(\mathbf{Y}, \mathbf{X})])}{\int \exp(\mathbb{E}_{-i}[\log p(\mathbf{Y}, \mathbf{X})]) d\mathbf{X}}$$

While the form of $\hat{q}(\mathbf{x}_i)$ is not restricted to a specific parametric form, it can be shown that when using conjugate priors, the distributions $\hat{q}_i(\mathbf{x}_i)$ have the same functional form as the priors $\hat{p}(\mathbf{x}_i)$. An example is shown in Appendix X, but a detailed mathematical treatment with derivations of multiple examples can be found in [20, 165, 252].

0.2.4.2 Fixed-form variational inference

An alternative and straightforward choice is to directly define a parametric form for the distribution $q(\mathbf{X})$ with some parameters $\boldsymbol{\Theta}$. Once the choice of $q(\mathbf{X})$ is made, the parameters $\boldsymbol{\Theta}$ are optimised to minimise $\text{KL}(q(\mathbf{X})||p(\mathbf{X}|\mathbf{Y}))$ (the variational problem):

$$\hat{\boldsymbol{\Theta}} = \arg\min_{\boldsymbol{\Theta}} \text{KL}(q(\mathbf{X})||p(\mathbf{X}|\mathbf{Y})) \tag{5}$$

$$= \mathbb{E}[\log(q(\mathbf{X})) - \log(p(\mathbf{X}, \mathbf{Y}))] \tag{6}$$

Numerically optimising this function requires the evaluation of expectations with respect to $q(\mathbf{X})$. In closed form, this is only feasable for a limited group of variational distributions. Alternatively,

one can attempt Monte Carlo approximations, but in practice this turns to be slow and leads to high-variance estimates [25, 189, 25].

Typically, one would choose this distribution to factorise over parameters and to be of the same (exponential) family as the prior $p(\mathbf{X})$. In such case there is a closed form coordinate-ascent scheme available, and it turns out that the fixed-form formulation is equivalent to the (non-parametric) mean-field derivation when using conjugate priors.
Unfortunately, for generic models with arbitrary families of distributions, no closed-form variational distributions exist [248, 22].

However, while the parametric assumption certainly limits the flexibility of variational distributions, the advantage of this formulation is that it unveils the possibility to use fast gradient-based methods for the inference procedure [92, 189].

0.2.5 Expectation Propagation

Expectation Propagation (EP) is another deterministic strategy with a similar philosophy as the Variational approach. It is also based on minimising the KL divergence between a variational distribution $q(\mathbf{X})$ and the true posterior $p(\mathbf{X}|\mathbf{Y})$, but while variational inference minimises $KL(p||q)$, EP maximises the reverse KL-divergence $KL(q||p)$.

Interestingly, this simple difference leads to an inference scheme with stringkly different properties. This can be understood by inspecting the differences between the two KL divergence formulas:

Variational inference:

$$\mathrm{KL}(q(\mathbf{X})||p(\mathbf{X}|\mathbf{Y})) = -\int_z q(\mathbf{X}) \log \frac{p(\mathbf{X}|\mathbf{Y})}{q(\mathbf{X})} \tag{7}$$

Expectation propagation:

$$\mathrm{KL}(p(\mathbf{X}|\mathbf{Y})||q(\mathbf{X})) = -\int_z p(\mathbf{X}|\mathbf{Y}) \log \frac{q(\mathbf{X})}{p(\mathbf{X}|\mathbf{Y})} \tag{8}$$

In regions of $\mathbf{X}$ where the true posterior density $p(\mathbf{X}|\mathbf{Y})$ is small, setting a large density for $q(\mathbf{X})$ has a much stronger penalisation in Equation (8) than in Equation (7), because of the true posterior density being on the denominator. Hence, EP tends to avoid areas where the density $p(\mathbf{X}|\mathbf{Y})$ is very low, even if it does not correspond to areas of very high-density in $p(\mathbf{X}|\mathbf{Y})$. In contrast, in Equation (7) there is a strong penalty for having low-density $q(\mathbf{X})$ values.
As discussed in [20], the practical consequences of this duality can be observed when the posterior is multi-modal, as in any sufficiently complex model. In VI, $q(\mathbf{X})$ converges towards areas of high-density in $p(\mathbf{X}|\mathbf{Y})$, namely local optima. In contrast, EP tends to capture as much non-zero density regions from $p(\mathbf{X}|\mathbf{Y})$ as possible, thereby averaging across all optima. In the context of doing predictions, the VI solution is much more desirable than the EP solution, as the average of two good parameter values is not necessarily a good parameter itself.
A detailed mathematical treatment of EP, including derivations for specific examples, can be found in [20, 165, 156]

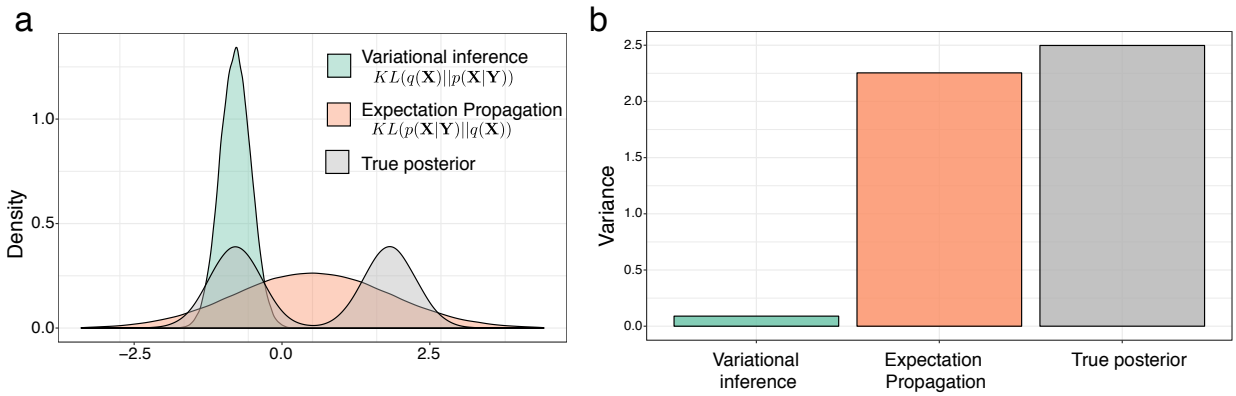


Figure 0.3: Illustrative comparison of Variational inference and Expectation Propagation. Shown is the (a) Density and (b) Variance of the true posterior distribution $p(\mathbf{X}|\mathbf{Y})$ (grey), the variational distribution (orange) and the expectation propagation distribution (green).

Following the rationale above, it is easy to predict that variational inference tends to be underestimate the variance of the posterior density. Yet, empirical research have shown that this is acceptable, provided that a good model selection is performed [21].

0.2.5.1 Conclusions

In this section we have introduced Bayesian modelling and variational inference methods, which will be used later in this chapter.
More generally, variational inference is growing in popularity for the analysis of big data sets and it has been applied to a myriad of different problems, including genome-wide association studies [37], population genetics, [186], network analysis [203] and natural language processing [23].

Yet, despite its increasing success, there is significant room for improvement. First and foremost, the theoretical guarantees of variational inference are not as developed as in sampling-based MCMC schemes[22, 248, 167]. As an example, the mean-field setting makes strong independence assumptions about the parameters. Although it tends to besurprisingly effective, it is not clear in which applications the dependencies between the parameters are important enough than the mean-field approximation could potentially break.
More generally, an open research problem is understanding what are the statistical properties of the variational posterior with respect to the exact posterior [22, 248].

As we shall demonstrate later, alternative strategies have been considered to allow some dependencies between the variables, resulting in *structured* mean-field approximations[91, 230]. However, they often lead to very complex (if not intractable) inference frameworks.

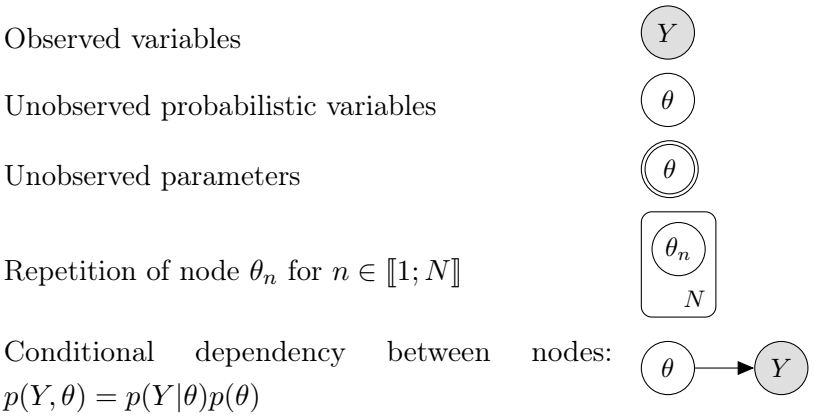
Finally, another area of extensive research is how to extend the applicability of VI to non-conjugate models. As discussed in Section 0.2.3, the ELBO of non-conjugate models contains intractable integrals, and setting up an inference scheme requires the use of either stochastic Monte Carlo approximations or deterministic approximations that introduce additional lower bounds [248, 207,

59]. In this thesis we follow this rationale to derive an inference framework for a model with non-gaussian likelihoods.

0.3 Graphical notations for probabilistic models

Probabilistic models can be represented in a diagrammatic format (i.e. a graph or a network) that offers a compact visual representation of complicated systems of probability distributions [20]. In a graphical model the relationship between the nodes becomes more explicit, namely their conditional independence properties which allow the joint distribution over all variables to be factorised into a series of simpler products involving subsets of variables [20]. The basic unit of a network is the node, which represents the different types of variables, including observed variables, unobserved probabilistic variables and unobserved parameters. The nodes are connected by unidirectional edges (arrows) which capture the conditional independence relationship between the variables.

For this thesis we adapted the graphical notations from [56].



0.4 Latent variable models for genomics

With the exponential growth in the use of high-throughput genomics, biological data sets are increasingly high dimensional, both in terms of samples and features. A key principle of biological data sets is that variation between the features results from differences in underlying, often unobserved, processes. Such processes, whether driven by biological or technical effects, are manifested by coordinated changes in multiple features. This key assumption sets off an entire statistical framework of exploiting the redundancy encoded in the data set to learn the (latent) sources of variation in an unsupervised fashion. This is the aim of dimensionality reduction techniques, or latent variable models (LVMs) [122, 217, 133, 182, 129, 218, 155].

0.4.1 General mathematical formulation

Given a dataset $\mathbf{Y}$ of N samples and D features, LVMs attempt to exploit the dependencies between the features by reducing the dimensionality of the data to a potentially small set of K latent variables, also called factors. The mapping between the low-dimensional space and the high-dimensional space is performed via a function $f(\mathbf{X}|\mathbf{\Theta})$ that depends on some parameters $\mathbf{\Theta}$.
The choice of $f(\mathbf{X}|\mathbf{\Theta})$ is essentially the field of dimensionality reduction. A trade-off exists between complexity and interpretation: while non-linear functions such as deep neural networks provide

more explanatory power, this leads to a considerable challenges in interpretation [251]. Hence, for most applications where interpretability is important, $f(\mathbf{X}|\mathbf{\Theta})$ is assumed to be linear [**XX**]:

$$\mathbf{Y} = \mathbf{Z}\mathbf{W}^T \tag{9}$$

where $\mathbf{Z} \in \mathbb{R}^{N\times K}$ is a matrix that contains the low-dimensional representation for each sample (i.e. the factors). The matrix $\mathbf{W} \in \mathbb{R}^{D\times K}$ contains the weights or loadings, which provide the linear mapping between the features and the factors.
Note that the aim in dimensionality reduction is to exploit the coordinated heterogeneity between features, and hence features are assumed to be centered without loss of generality.

The inference procedure consists in learning the values of all unobserved variables, including factors and weights. As we shall demonstrate, different inference schemes and assumptions on the prior distributions lead to significantly different model outputs [192].

0.4.2 Principal component Analysis

Principal Component Analysis (PCA) is the most popular technique for dimensionality reduction [93, 195]. Starting from Equation (9), two formulations of PCA exist [20]. In the maximum variance formulation, the aim is to infer an orthogonal projection of the data onto a low-dimensional space such that variance explained by the projected data is maximised:

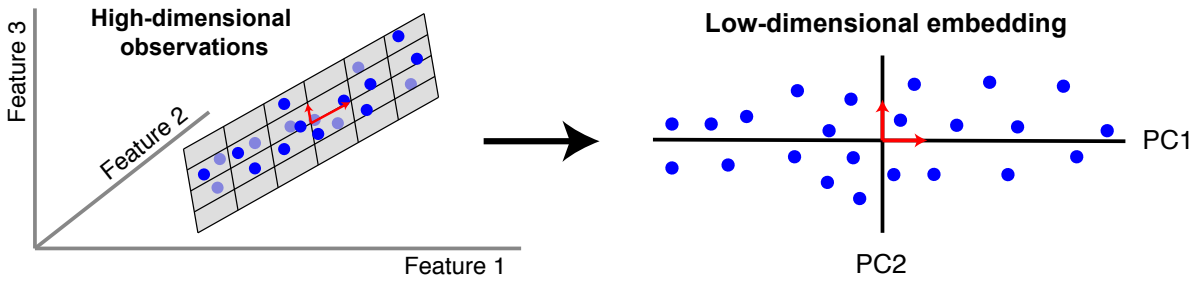


Figure 0.4

For a single principal component, the optimisation problem is:

$$\underset{\|\mathbf{w}\|=1}{\arg\max} = \mathbf{w}_1^T\mathbf{Y}^T\mathbf{Y}\mathbf{w} \tag{10}$$

where $\mathbf{Y}^T\mathbf{Y} = \mathbf{S} \in \mathbb{R}^{D\times D}$ is the data covariance matrix and $\mathbf{w}_1^T$ is the vector of loadings.
The k-th principal component can be found by subtracting from $\mathbf{Y}$ the reconstructed data by the previous $k-1$ principal components. If we define $\mathbf{z}_k = \mathbf{w}_k^T\mathbf{Y}$ to be the k-th principal component:

$$\hat{\mathbf{Y}} = \mathbf{Y} - \sum_{k=1}^{K}(\mathbf{z}_k\mathbf{w}_k^T)$$

Re-applying Equation (10) defines the new optimisation problem.

In its minimum error formulation, the aim is to find an equivalent projection that minimises the mean squared error between the observations and the data reconstructed using all principal components:

$$\arg\max_{\|\mathbf{w}\|=1} \|\mathbf{Y} - \sum_{k=1}^{K} (\mathbf{z}_k \mathbf{w}_k^T)\|^2$$

where $\|\cdot\|^2$ is the Frobenius norm.

In both cases, solving the optimisation problems via Lagrange multipliers leads, remarkably, to the same solution:

$$\mathbf{S}\mathbf{w}_k = \lambda_k \mathbf{w}_k \tag{11}$$

Hence, the loading vectors $\mathbf{w}_k$ are the eigenvectors of $\mathbf{S}$, which can be computed via singular value decomposition [20].

The reason why the maximum variance solution and the minimum reconstruction error solution are the same can be understood by applying Pythagoras theorem to the right triangle defined by the projection of a sample $\mathbf{y}_n$ to a loading vector $\mathbf{w}$ (Figure 0.5). Assuming again centered data, the variance of $\mathbf{y}_n$ is $\|\mathbf{y}_n\| = \mathbf{y}_n^T \mathbf{y}_n$. This variance decomposes as the sum of the variance in the latent space $\|\mathbf{z}_n\| = \mathbf{z}_n^T \mathbf{z}_n$ and the residual variance after reconstruction $\|\mathbf{y}_n - \mathbf{z}_n \mathbf{w}^T\|$:

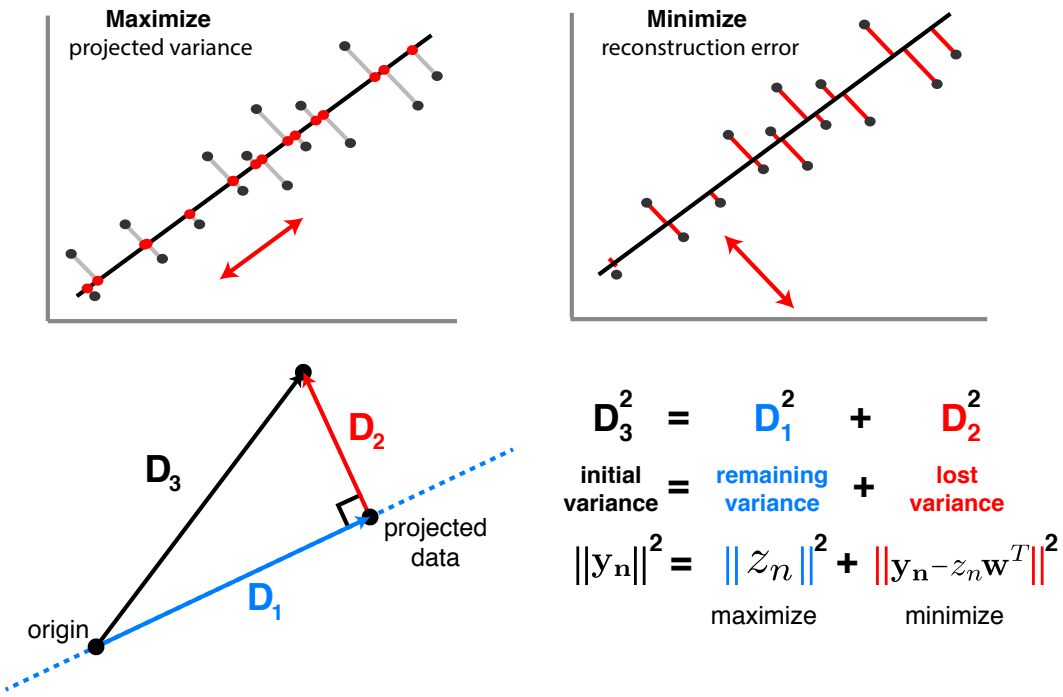


Figure 0.5: In the maximum variance formulation we aim at maximising the variance of the projected data (blue line), whereas in the minimum error formulation we are aimed at minimising the residual variance (red line). Given a fixed total variance (black line), both strategies are equivalent

The main strength of PCA relies on its simplicity and closed form solution. Additionally, the linear mapping has the advantage of yielding interpretable loadings, so that inspection of $\mathbf{w}_k$ reveals which features are jointly affected by the k-th principal component.
However, PCA suffers from serious drawbacks when applying it to real data sets [136]. First, biological measurements are inherently noisy, and there is no explicit account of noise in PCA. In practice, high variance components are often asociated with signal whereas low-variance components are assumed to be noise, but an ideal model should explicitly disentangle the uncoordinated variability that is attributed to noise from the coordinated variability that is characterised as signal. Second, in its original formulation, no missing data is allowed [98]. Third, there is no rationality on how to evaluate the fit and perform model selection. Finally, it does not offer a principled way of modelling prior information about the data.

0.4.3 Probabilistic Principal Component Analysis and Factor Analysis

A probabilistic version of PCA was initially proposed in [229]. It can be formulated by converting some (or all) fixed parameters into random variables and adding an explicit noise term to Equation (9):

$$\mathbf{Y} = \mathbf{W}\mathbf{Z} + \boldsymbol{\epsilon} \tag{12}$$

where the weights $\mathbf{W}$ are assumed to be non-probabilistic parameters, but the noise $\boldsymbol{\epsilon}$ and the latent variables $\mathbf{Z}$ (the principal components) are assumed to follow an isotropic normal distribution:

$$p(\mathbf{Z}) = \prod_{n=1}^{N}\prod_{k=1}^{K} \mathcal{N}\left(z_{nk} \,|\, 0, 1\right)$$
$$p(\epsilon) = \mathcal{N}\left(\epsilon \,|\, 0, \sigma^2\right)$$

All together, this leads to a normally-distributed likelihood:

$$p(\mathbf{Y}|\mathbf{W}, \mathbf{Z}, \sigma) = \prod_{n=1}^{N}\prod_{d=1}^{D} \mathcal{N}\left(y_{n,d} \,|\, \mathbf{w}_{:,k}^T \mathbf{z}_{n,:}, \sigma^2\mathbf{I}\right) \tag{13}$$

The corresponding graphical model is:

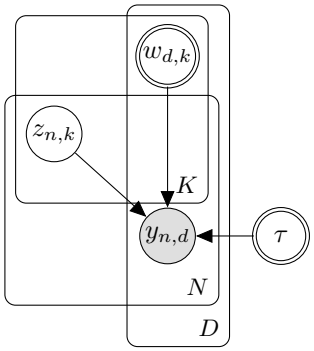


Figure 0.6: Graphical model for probabilistic PCA. The latent variables are modelled as random variables, whereas the loadings and the noise are modelled as deterministic parameters.

Importantly, the choice of the distribution for ϵ implies that the noise of each feature is independent but restricted to have the same variance σ. In practice this is a limiting assumption, as different features are expected to show different degrees of noise, albeit this constraint can be relaxed and forms the basis of Factor Analysis [200, 20].

The inference procedures involves learning the parameters $\mathbf{W}$, and σ^2 and a posterior probability distribution for $\mathbf{Z}$. As the model depends on latent variables, inference can be performed using the iterative Expectation-Maximisation (EM) algorithm [200, 20]. In the expectation step, the posterior distribution for $\mathbf{Z}$ is computed in closed form (due to conjugacy between the likelihood and the prior), given current estimates for the parameters $\mathbf{W}$, and σ^2. In the maximisation step, the parameters are calculated by maximising the expectation of the joint log likelihood under the posterior distribution of $\mathbf{Z}$ found in the E step [229].
Interestingly, the EM solution of probabilistic PCA lies in the same subspace than the traditional PCA solution [229], but the use of a probabilistic framework brings several benefits. First, model selection can be performed by comparing likelihoods across different settings of parameters. Second, missing data can naturally be accounted for by ignoring the missing observations from the likelihood. Finally, the probabilistic formulation sets the core framework for a Bayesian treatment of PCA, enabling a broad range of principled extensions tailored different types of data sets.

0.4.4 Bayesian Principal Component Analysis and Bayesian Factor Analysis

The full Bayesian treatment of PCA requires the specification of prior probability distributions for all unobserved variables:

$$
\begin{aligned}
p(\mathbf{Z}) &= \prod_{n=1}^{N}\prod_{k=1}^{K}\mathcal{N}\left(z_{nk}\,|\,0,1\right)\\
p(\mathbf{W}) &= \prod_{d=1}^{D}\prod_{k=1}^{K}\mathcal{N}\left(w_{dk}\,|\,0,1\right)\\
p(\epsilon) &= \mathcal{N}\left(\epsilon\,|\,0,\tau^{-1}\right)\\
p(\tau) &= \mathcal{G}\left(\tau\,|\,a_0,b_0\right)
\end{aligned}
$$

where τ is the precision (inverse of the variance) of the noise term. A generalisation to Bayesian Factor Analysis follows by allowing a separate noise term per feature:

$$
\begin{aligned}
p(\boldsymbol{\epsilon}) &= \prod_{d=1}^{D}\mathcal{N}\left(\epsilon_d\,|\,0,\tau_d^{-1}\right)\\
p(\boldsymbol{\tau}) &= \prod_{d=1}^{D}\mathcal{G}\left(\tau_d\,|\,a_0,b_0\right)
\end{aligned}
$$

where a_0 and b_0 are fixed hyperparameters. As in Equation (13), this results in a Normal likelihood:

$$
p(\mathbf{Y}|\mathbf{W},\mathbf{Z},\boldsymbol{\tau}) = \prod_{n=1}^{N}\prod_{d=1}^{D}\mathcal{N}\left(y_{nd}\,|\,\mathbf{w}_d^T\mathbf{z}_n,\tau_d\right)
$$

The corresponding graphical model is:

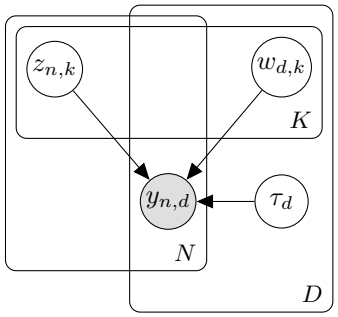


Figure 0.7: Graphical model for Bayesian Factor Analysis. All unobserved variables are modelled as random variables.

0.4.4.1 Hierarchical priors: Automatic relevance determination

A key advantage of the full Bayesian treatment is that it explicitly captures uncertainity on the estimation of all unobserved variables, as opposed to the probabilistic PCA model [19, 18]. Yet, more importantly, the use of (hierarchical) prior distributions allow different modelling assumptions to be encoded, providing a flexible and principled approach to extend PCA to a myriad of modelling scenarios, including multi-view generalisations [116, 240, 118, 32, 112, 253].

As an example, a major challenge in PCA is how to determine the dimensionality of the latent space (i.e. the number of principle components). As we will show, the use of hierarchical prior distributions allows the model to introduce sparsity assumptions on the loadings in such a way that the model automatically learns the number of factors.
In the context of Factor Analysis, one the first sparsity priors to be proposed was the Automatic Relevance determination (ARD) prior [168, 148, 19, 18].

$$p(\mathbf{W}|\boldsymbol{\alpha}) = \prod_{k=1}^{K} \mathcal{N}\left(\mathbf{w}_{:,k} \,|\, 0, \frac{1}{\alpha_k}\mathbf{I}_D\right) \qquad p(\boldsymbol{\alpha}) = \prod_{k=1}^{K} \mathcal{G}\left(\alpha_k \,|\, a_0^{\alpha}, b_0^{\alpha}\right)$$

The aim of this prior is two-fold. First, the zero-mean normal distribution specifies that, *a priori*, no information is available and all features are *inactive*. When exposed to some data, the posterior distribution for $\mathbf{W}$ will be estimated by weighting the contribution from the likelihood, potentially allowing features to escape from the zero-centered prior (Figure 0.8).
Second, performing inference on the variable $\boldsymbol{\alpha} = \{\alpha_1, \cdots, \alpha_k\}$ enables the model to discard inactive factors. To understand this, let us assume that only $K = 5$ true factors exist, but the model is initialised with $K = 20$ factors. In such case, inactive factors can be prunned out by driving the corresponding α_k to infinity. In turn, this causes the posterior $p(\mathbf{w}_{:,k}|\mathbf{Y})$ to be sharply peaked at zero, resulting in the inactivation of all its weights Figure 0.9.

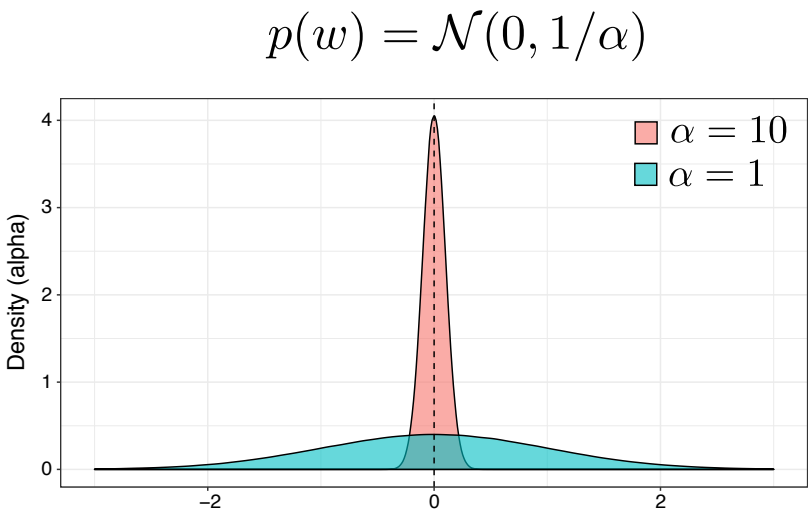


Figure 0.8: Visualisation of the sparsity-inducing Automatic Relevance Determination prior

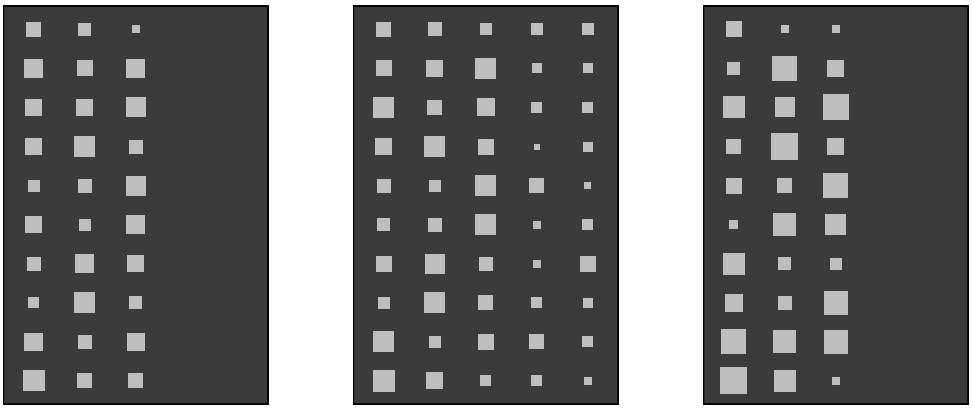


Figure 0.9: Hinton plots display the values of the loading matrix, similar to a heatmap, where bigger squares depict larger loadings. Shown are the Hinton plots for (a) the true weights, (b) the infered weights by a Factor Analysis model with no ARD prior (middle), and (c) the infered weights by a Factor Analysis model with ARD prior per factor. This figure was generated using simulated data with $N = 100$ samples, $D = 10$ features and $K = 3$ factors.

0.4.4.2 Hierarchical priors: Spike-and-slab prior

Sparse extensions of the Bayesian factor analysis model have been proposed as a regularisation mechanism but also to model inherent assumptions regarding the sparse nature of biological data [217, 71].

The variability observed in biological data is driven both by technical factors and biological factors. Thechnical factors (i.e. batch effects) tend to be relatively strong and alter the expression of a large proportion of genes, whereas the biological factors are potentially weak effects driven by changes in small gene regulatory networks [71]. Hence, a practical factor analysis model should be able to learn factors with different degrees of sparsity.

The ARD prior proposed in Section 0.4.4.1 allows entire factors to be dropped out from the model, but it provides a weak degree of regularisation when it comes to inactivating individual loadings within the active factors.

A sparse generalisation of the Factor Analysis model proposed above can be achieved by combining the ARD prior with a spike-and-slab prior [157, 230]:

$$p(w_{d,k} \mid \alpha_k, \theta_k) = (1 - \theta_k)\mathbb{1}_0(w_{d,k}) + \theta_k \mathcal{N}\left(w_{d,k} \mid 0, \alpha_k^{-1}\right) \tag{14}$$

$$p(\theta_k) = \text{Beta}\left(\theta_k \mid a_0^\theta, b_0^\theta\right) \tag{15}$$

$$p(\alpha_k) = \mathcal{G}\left(\alpha_k \mid a_0^\alpha, b_0^\alpha\right) \tag{16}$$

The corresponding graphical model is:

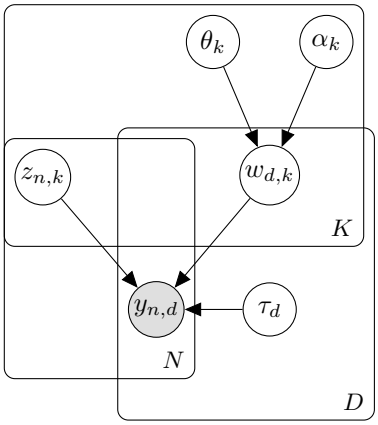


Figure 0.10: Graphical model for Bayesian sparse Factor Analysis. A double sparsity-inducing prior is used on the loadings: an ARD prior to prune inactive factors and a spike-and-slab prior to inactive individual features within the active factors.

The spike-and-slab prior is effectively a mixture model where features are sampled from a zero-inflated gaussian distribution, where $\theta_k \in (0, 1)$ dictates the level of sparsity per factor (i.e. how many active features). A value of θ_k close to 0 implies that most of the weights of factor k are shrinked to 0 (i.e. a sparse factor), whereas a value of θ_k close to 1 implies that most of the weights are non-zero (i.e. dense factors). By learning θ_k from the data, the model naturally accounts for combinations of sparse and dense factors.

0.5 Multi-view factor analysis models

Probabilistic PCA and Factor Analysis perform dimensionality reduction from a single input matrix. In some occasions data is collected from multiple data sources that exibit heterogeneous statistical properties, resulting in a structured data set where features are naturally partitioned into views [245, 135, 247]. A clear biological example is multi-omics data, where, for the same set of samples, multiple molecular layers are profiled. Each of the data modalities can be analysed separately using conventional (single-view) methods, but in the ideal strategy a single model should be used to

leverage information across all molecular layers using a flexible and principled approach. This is refered to as the multi-view learning problem [245, 135].
A tempting approach to circumvent the multi-view learning problem is to simply concatenate all different data sets before applying conventional (single-view) latent variable models [197]. However, this is prone to fail for several reasons. First, heterogeneous data modalities cannot always be modelled using the same likelihood function. For example, continuous measurements are often modelled using a normal distribution, but binary and count-based traits are not appropriately modelled by this distribution [179]. Second, even if all views are modelled with the same likelihood, differences in the scale and the magnitude of the variance can lead to some views being overrepresented in the latent space. Finally, in a multi-view data set we expect multiple sources of variation, some of which driven by a single view, whereas others could capture shared variability across multiple views. In other words, from a structured input space, one can also expect a structured latent representation. Not taking this behaviour into account can lead to challenges in the interpretability of the latent space.

A comprehensive review of multi-view machine learning methods can be found in [245] and a more genomics-oriented perspective can be found in [197]. For the purpose of this thesis, we will describe only the use of latent variable models for multi-view data integration.

0.5.1 Canonical Correlation Analysis

Canonical Correlation Analysis (CCA) is a simple extension of PCA to find linear components that capture correlations between two datasets [94, 82].
Given two data matrices $\mathbf{Y}_1 \in \mathbb{R}^{N \times D_1}$ and $\mathbf{Y}_2 \in \mathbb{R}^{N \times D_2}$ CCA finds a set of linear combinations $\mathbf{U} \in \mathbb{R}^{D_1 \times K}$ and $\mathbf{V} \in \mathbb{R}^{D_2 \times K}$ with maximal cross-correlation. For the first pair of canonical variables, the optimisation problem is:

$$(\hat{\mathbf{u}_1}, \hat{\mathbf{v}_1}) = \arg\max_{\mathbf{u}_1, \mathbf{v}_1} corr(\mathbf{u}_1^T \mathbf{Y}_1, \mathbf{v}_1^T \mathbf{Y}_2)$$

As in conventional PCA, the linear components are constraint to be orthogonal. Hence, the first pair of canonical variables $\mathbf{u}_1$ and $\mathbf{v}_1$ contain the linear combination of variables that have maximal correlation. Subsequently, Therefore, the second pair of canonical variables $\mathbf{u}_2$ and $\mathbf{v}_2$ is found out of the residuals of the first canonical variables.

Given the similarity with PCA, both methods share statistical properties, including the linear mapping between the low-dimensional space and the high-dimensional space, and the closed-form solution using singular value decomposition [94, 82].
Because of its simplicity and efficient computation, CCA has widespread use as a dimensionality reduction technique [82]. Yet, as expected, CCA suffers from the same pitfalls as PCA: difficulties in selecting the number of components, lack of sparsity in the solutions and absence of probabilistic formulation. In addition, CCA have been shown to overfit for datasets where $D >> N$ [152, 78]. Hence, probabilistic versions with sparsity assumptions that reduce overfitting and improve interpretability followed.

0.5.2 Probabilistic Canonical Correlation Analysis

Following the derivation of probabilistic PCA [229], a similar effort enabled a probabilistic formulation of CCA as a generative model [11].
In this model, the two matrix of observations $\mathbf{Y}^1$ and $\mathbf{Y}^2$ are decomposed in terms of two loading matrices $\mathbf{W}^1$ and $\mathbf{W}^2$ but a joint latent matrix $\mathbf{Z}$:

$$\mathbf{Y}^1 = \mathbf{W}^2\mathbf{Z} + \epsilon^1$$
$$\mathbf{Y}^2 = \mathbf{W}^2\mathbf{Z} + \epsilon^2$$

With the following prior probability distributions:

$$p(z_{nk}) = \mathcal{N}\left(z_{nk} \,|\, 0, 1\right)$$
$$p(\epsilon^1) = \mathcal{N}\left(\epsilon^1 \,|\, \tau_1^{-1}\right)$$
$$p(\epsilon^2) = \mathcal{N}\left(\epsilon^2 \,|\, \tau_2^{-1}\right)$$

As in [229], the loadings and the variance of the noise are assumed to be non-probabilistic parameters, whereas the factors are probabilistic unobserved variables. This yields the following likelihood functions:

$$p(\mathbf{Y}^1|\mathbf{W}^1, \mathbf{Z}, \tau_1) = \prod_{n=1}^{N}\prod_{d=1}^{D_1} \mathcal{N}\left(y_{n,d}^1 \,|\, (\mathbf{w}_{:,k}^1)^T\mathbf{z}_n, \tau_1^{-1}\right) \tag{17}$$
$$p(\mathbf{Y}^2|\mathbf{W}^2, \mathbf{Z}, \tau_2) = \prod_{n=1}^{N}\prod_{d=1}^{D_2} \mathcal{N}\left(y_{n,d}^2 \,|\, (\mathbf{w}_{:,k}^2)^T\mathbf{z}_n, \tau_2^{-1}\right)$$

The corresponding graphical model is:

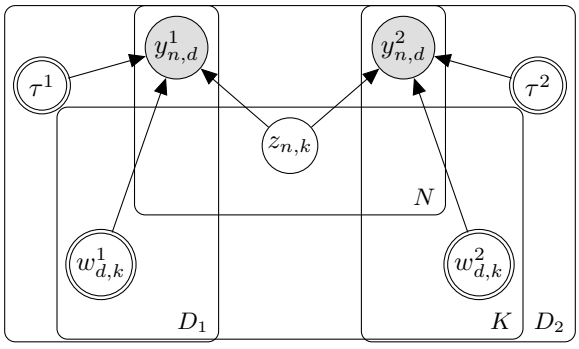


Figure 0.11: Graphical model for probabilistic Canonical Correlation Analysis

Notice that the observations for both data sets are generated from the same set of latent variables $\mathbf{Z}$. This ensures that the model is focused on capturing the variation associated with cross-correlated groups of features.
Analogously to probabilistic PCA, the expected value of the posterior distribution $p(\mathbf{Z}|\mathbf{Y}^1, \mathbf{Y}^2)$ span

the same subspace as standard CCA [11]. Nonetheless, one of the many advantage of a probabilistic formulation is that it enables a broad range of principled extensions into larger graphical models.

0.5.3 Bayesian Canonical Correlation Analysis

A fully Bayesian treatment of CCA followed based on exactly the same principle presented in Section 0.4.4 by introducing prior distributions to all unobserved variables [241, 117]:

$$
\begin{aligned}
p(\mathbf{Z}) &= \prod_{n=1}^{N}\prod_{k=1}^{K}\mathcal{N}\left(z_{nk}\,|\,0,1\right) \\
p(\epsilon^1) &= \mathcal{N}\left(\epsilon^1\,|\,\sigma_1^2\right) \\
p(\epsilon^2) &= \mathcal{N}\left(\epsilon^2\,|\,\sigma_2^2\right) \\
p(\mathbf{W}^1|\boldsymbol{\alpha}) &= \prod_{k=1}^{K}\mathcal{N}\left(\mathbf{w}^1_{:,k}\,|\,0,\frac{1}{\alpha_k}\mathbf{I}_{D_1}\right) \\
p(\mathbf{W}^2|\boldsymbol{\alpha}) &= \prod_{k=1}^{K}\mathcal{N}\left(\mathbf{w}^2_{:,k}\,|\,0,\frac{1}{\alpha_k}\mathbf{I}_{D_2}\right) \\
p(\boldsymbol{\alpha}) &= \prod_{k=1}^{K}\mathcal{G}\left(\alpha_k\,|\,a_0^\alpha,b_0^\alpha\right)
\end{aligned}
$$

Resulting in the same likelihood model as in Equation (17). Yet, notice that an ARD is introduced per factor, allowing an automatic inference of the dimensionality in the latent subspace. Also, there is some flexibility in the definition of noise. An independent noise term can be defined per view or per per feature. One could also model correlated noise by generalising the distribution to a multivariate gaussian with full-rank covariance. [241, 117].

The corresponding graphical model is:

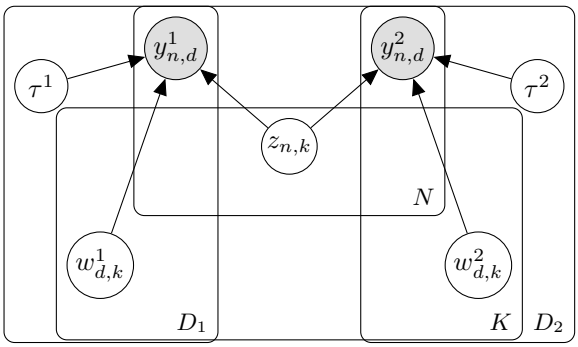


Figure 0.12: Graphical model for Bayesian Canonical Correlation Analysis

As expected, in practice this yields a more sparse solution than traditional CCA (Figure 0.13):

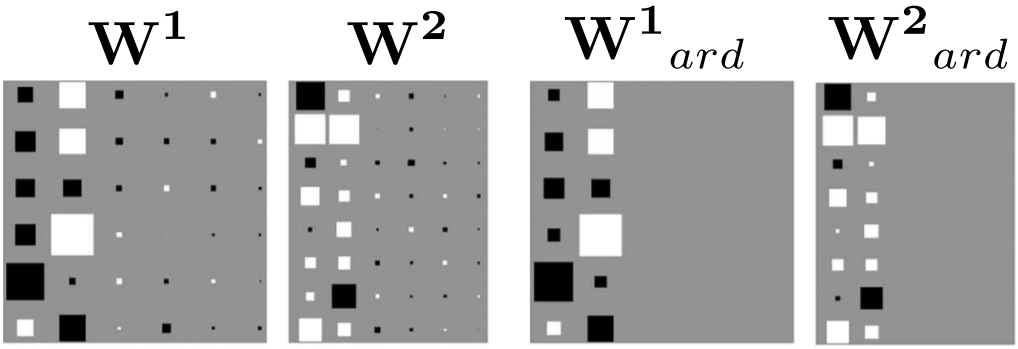


Figure 0.13: Comparison of the Hinton's diagram of $\mathbf{W}^1$ and $\mathbf{W}^2$ for the maximum likelihood CCA model (two left plots) and the variational bayes CCA model (two right plots). Reprinted from [241] with modifications.

0.5.4 Group Factor Analysis

Group Factor Analysis (GFA) is the natural generalisation of Bayesian Canonical Correlation Analysis to an arbitrary number of views. The original idea was originally presented in [240] and a series of generalisations followed, tailored with specific assumptions for different applications [118, 134, 32, 112, 253, 194]. In this section we will outline the core principle of GFA.

Given a data set of M views $\mathbf{Y}_1, \cdots, \mathbf{Y}_M$, the task of GFA is to find K factors that capture the variability *within* as well as the variability *between* views. In other words, we want to capture factors that not only explain variance that is shared across all views but we also want to capture factors that explain variance within a single view or between different subsets of views.
The starting point is to generalise the Bayesian CCA model (Section 0.5.3) to M views:

$$\begin{aligned}\mathbf{Y}^1 &= \mathbf{W}^2\mathbf{Z} + \epsilon^1 \\ \mathbf{Y}^2 &= \mathbf{W}^2\mathbf{Z} + \epsilon^2 \\ &\cdots \\ \mathbf{Y}^M &= \mathbf{W}^M\mathbf{Z} + \epsilon^M\end{aligned}$$

Notice that there is a common factor space for all views, but there is a view-specific weight matrix. The key to disentangle the activity of each factor in each view lies on the sparsity structure imposed in the weights. Intuitively, if a facto k is not driving any variation in a specific view m we want all the individual weights to be pushed to zero. As shown before, this behaviour can be achieved using Automatic Relevance Determination (ARD) priors. However, if we were to use the same approach as in Bayesian CCA, where the ARD prior for factor k is shared across all views, then factors would be restricted to have the same activity across all views.

In GFA this is generalised as follows:

$$p(\mathbf{W}) = \prod_{m=1}^{M} \prod_{k=1}^{K} \mathcal{N}\left(\mathbf{w}_{:,k}^{m} \,|\, 0, \frac{1}{\alpha_k^m}\right) \tag{18}$$

$$p(\boldsymbol{\alpha}) = \prod_{m=1}^{M} \prod_{k=1}^{K} \mathcal{G}\left(\alpha_k^m \,|\, a_0^\alpha, b_0^\alpha\right) \tag{19}$$

This is effectively setting an ARD prior per factor k and view m. The matrix $\boldsymbol{\alpha} \in \mathbb{R}^{M \times K}$ defines four types of factors: (1) Inactive factors that do not explain variance in any view, which corresponds to all values $\boldsymbol{\alpha}_k$ being large. (2) Fully shared factors that explain variance across all views, which corresponds to all values $\boldsymbol{\alpha}_k$ being small. (3) Unique factors that explain variance in a single view, which corresponds to all values $\boldsymbol{\alpha}_k$ being large, except for one entry. (4) Partially shared factors that explain variance in a subsets views, which corresponds to a mixture of small and large values for $\boldsymbol{\alpha}_k$.

The corresponding graphical model is:

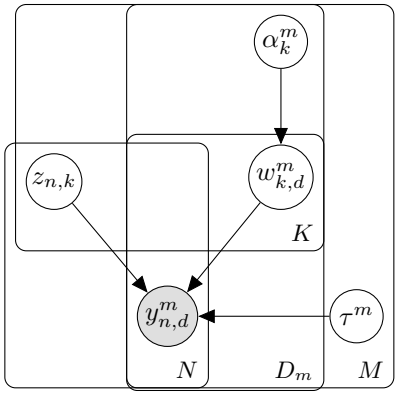


Figure 0.14: Graphical model for Bayesian Group Factor Analysis

Finally, notice that if $M = 1$ the model reduces to Bayesian PCA (Section 0.4.4), but when $M = 2$ the model does *not* reduce to Bayesian CCA because in the GFA setting factors are also allowed to capture both inter-specific variability (i.e. across views) intra-specific variability (within a view). In Bayesian CCA, the views share a common ARD prior per factor to enforce the factors to explain variation in both views, at the expense of ignoring sources of variability that are specific to a single view.

0.6 Multi-Omics Factor Analysis

The work described in this chapter results from a collaboration with Wolfgang Huber's group at the EMBL (Heidelberg, Germany). It has been peer-reviewed and published in [8].

The method was conceived by Florian Buettner, Oliver Stegle and me. I performed most of the mathematical derivations and implementation, but with significant contributions from Damien Arnol and Britta Velten. The CLL data application was led by Britta Velten whereas the single-cell application was lead by me, but with joint contributions in either cases. Florian Buettner, Wolfgang Huber and Oliver Stegle supervised the project.
The article was jointly written by Britta Velten and me, with contributions from all authors.

0.6.1 Model description

MOFA is a multi-view generalisation of traditional Factor Analysis to M input matrices (or views) based on the framework of Group Factor Analysis (discussed in **??**).
The input data consists on M views $\mathbf{Y}^m \in \mathbb{R}^{N \times D_m}$ with non-overlapping features that often represent different assays. However, there is flexibility in the definition of views.
Formally, the input data is factorised as:

$$\mathbf{Y}^m = \mathbf{Z}\mathbf{W}^{mT} + \boldsymbol{\epsilon}^m \tag{20}$$

where $\mathbf{Z} \in \mathbb{R}^{N \times K}$ is a matrix that contains the factor values and $\mathbf{W}^m \in \mathbb{R}^{D_m \times K}$ are a set of M matrices (one per view) that contain the weights that relate the high-dimensional space to the low-dimensional latent representation. Finally, $\boldsymbol{\epsilon}^m \in \mathbb{R}^{D_m}$ captures the residuals, or the noise, which is assumed to be normally distributed and heteroskedastic:

$$p(\epsilon_d^m) = \mathcal{N}\left(\epsilon_d^m \,|\, 0, 1/\tau_d^m\right) \tag{21}$$

Non-gaussian noise models can also be defined (see Section 0.6.6), but unless otherwise stated, we will always assume Gaussian residuals.
Altogether, this results in the following likelihood:

$$p(\mathbf{Y}|\mathbf{W}, \mathbf{Z}, \mathbf{T}) = \prod_{m=1}^{M} \prod_{d=1}^{D_m} \prod_{n=1}^{N} \mathcal{N}\left(y_{nd}^m \,|\, \mathbf{z}_n^T \mathbf{w}_d^m, 1/\tau_d^m\right) \tag{22}$$

Notice that the mathematical formulation so far is equivalent to the Group Factor Analysis described in **??**.

0.6.1.1 Prior distributions for the factors

For the factors, we define an isotropic Gaussian prior, as commonly done in most factor analysis models:

$$p(z_{nk}) = \mathcal{N}(z_{nk} \,|\, 0, 1) \tag{23}$$

This effectively assumes (1) a continuous latent space and (2) independence between samples and factors.

0.6.1.2 Prior distributions for the weights

The key determinant of the model is the regularization used on the prior distributions for the weights. Here we encode two levels of sparsity, a (1) view- and factor-wise sparsity and (2) an individual feature-wise sparsity. The aim of the factor- and view-wise sparsity is to disentangle the activity of factors to the different views, such that the weight vector $\mathbf{w}^m_{:,k}$ is shrunk to zero if the factor k does not explain any variation in view m.
In addition, we place a second layer of sparsity which encourages inactive weights on each individual feature. Mathematically, we express this as a combination of an Automatic Relevance Determination (ARD) prior [148] for the view- and factor-wise sparsity and a spike-and-slab prior [157] for the feature-wise sparsity: However, this formulation of the spike-and-slab prior contains a Dirac delta function, which makes the inference procedure troublesome. To solve this we introduce a re-parametrization of the weights w as a product of a Gaussian random variable $\hat{w}$ and a Bernoulli random variable s, [230] resulting in the following prior:

$$p(\hat{w}^m_{dk}, s^m_{dk}) = \mathcal{N}\left(\hat{w}^m_{dk} \,|\, 0, \frac{1}{\alpha^m_k}\right) \mathrm{Ber}(s^m_{dk} \,|\, \theta^m_k) \tag{24}$$

In this formulation α^m_k controls the activity of factor k in view m and θ^m_k controls the corresponding fraction of active weights (i.e. the sparsity levels).

Finally, we define conjugate priors for θ and α:

$$p(\theta^m_k) = \mathrm{Beta}\left(\theta^m_k \,|\, a^\theta_0, b^\theta_0\right) \tag{25}$$

$$p(\alpha^m_k) = \mathcal{G}(\alpha^m_k \,|\, a^\alpha_0, b^\alpha_0) \tag{26}$$

with hyper-parameters $a^\theta_0, b^\theta_0 = 1$ and $a^\alpha_0, b^\alpha_0 = 1e^{-5}$ to get uninformative priors. Posterior values of θ^m_k close to 0 implies that most of the weights of factor k in view m are shrinked to 0 (sparse factor). In contrast, a value of θ^m_k close to 1 implies that most of the weights are non-zero (non-sparse factor). A small value of α^m_k implies that factor k is active in view m. In contrast, a large value of α^m_k implies that factor k is inactive in view m.

All together, the joint probability density function of the model is given by

$$
\begin{aligned}
p(\mathbf{Y}, \hat{\mathbf{W}}, \mathbf{S}, \mathbf{Z}, \boldsymbol{\theta}, \boldsymbol{\alpha}, \boldsymbol{\tau}) = & \prod_{m=1}^{M} \prod_{n=1}^{N} \prod_{d=1}^{D_m} \mathcal{N}\left(y_{nd}^m \,|\, \sum_{k=1}^{K} s_{dk}^m \hat{w}_{dk}^m z_{nk}, 1/\tau_d\right) \\
& \prod_{m=1}^{M} \prod_{d=1}^{D_m} \prod_{k=1}^{K} \mathcal{N}\left(\hat{w}_{dk}^m \,|\, 0, 1/\alpha_k^m\right) \mathrm{Ber}(s_{d,k}^m | \theta_k^m) \\
& \prod_{n=1}^{N} \prod_{k=1}^{K} \mathcal{N}\left(z_{nk} \,|\, 0, 1\right) \\
& \prod_{m=1}^{M} \prod_{k=1}^{K} \mathrm{Beta}\left(\theta_k^m \,|\, a_0^\theta, b_0^\theta\right) \\
& \prod_{m=1}^{M} \prod_{k=1}^{K} \mathcal{G}\left(\alpha_k^m \,|\, a_0^\alpha, b_0^\alpha\right) \\
& \prod_{m=1}^{M} \prod_{d=1}^{D_m} \mathcal{G}\left(\tau_d^m \,|\, a_0^\tau, b_0^\tau\right).
\end{aligned}
\tag{27}
$$

and the corresponding graphical model is shown in Figure 0.16. This completes the definition of the MOFA model.

0.6.1.3 Interpretation of the factors

Each factor ordinates cells along a one-dimensional axis centered at zero. Samples with different signs indicate opposite phenotypes, with higher absolute value indicating a stronger effect. Intuitively, their interpretation is similar to that of principal components in PCA.
For example, if the k-th factor captures the variability associated with cell cycle, we could expect cells in the Mitosis state to be at one end of the factor (irrespective of the sign, only the relative positioning being of importance). In contrast, cells in G1 phase are expected to be at the other end of the factor. Cells with intermediate phenotype, or with no clear phenotype (for example if no cell cycle genes are profiled), are expected to be located around zero.

0.6.1.4 Interpretation of the weights

The weights provide a score for each gene on each factor, and are interpreted in a similar way as the factors. Genes with no association with the factor are expected to have values close to zero, as specified by the prior. In contrast, genes with strong association with the factor are expected to have large absolute values. The sign of the loading indicates the direction of the effect: a positive loading indicates that the feature is more active in the cells with positive factor values, and viceversa.
Following the cell cycle example from above, genes that are upregulated in the M phase are expected to have large positive weights, whereas genes that are downregulated in the M phase (or, equivalently, upregulated in the G1 phase) are expected to have large negative weights.

0.6.1.5 Missing values

The probabilistic formulation naturally accounts for incomplete data matrices, as missing observations do not intervene in the likelihood. In practice, we implement this using memory-efficient binary masks $\mathcal{O}^m \in \mathbb{R}^{N \times D_m}$ for each view m, such that $\mathcal{O}_{n,d} = 1$ when feature d is observed for sample n, 0 otherwise.

0.6.2 Downstream analysis

Once trained, the MOFA model can be queried for a set of downstream analysis (Figure 0.15):

- **Variance decomposition**: calculate the variance explained (R^2) by each factor in each view. This is the first and arguably the most important plot to be inspected once the model is trained, as it summarises the variation (i.e. the signal) in a complex multi-view data set using a simple heatmap. With a quick visual inspection, this plot can be used to determine which factors are shared between multiple data modalities and which ones are exclusive to a single data modality.
- **Ordination of the samples in the latent space**: as in any latent variable model, the samples can be visualised in the latent space using scatterplots or beeswarm plots. As we will demonstrate, simply by colouring or shaping the samples in the factor space using external covariates one can easily characterise the etiology of some of the factors.
- **Inspection of weights**: the feature weights can be interpreted as an importance score for each feature on each factor. Inspecting the top weights for a given factor can help to reveals the molecular signatures that underlie each factor.
- **Association analysis between factors and external covariates**: multi-omic data sets typically consist on a large set of molecular readouts that are used for model training, and a small set of additionals covariates or response variables such as clinical outcome measurements. The external covariates that are not used for model training can be linked to the factors *a posteriori* using a simple association analysis.
- **Imputation**: the latent factors capture a condensed low-dimensional representation of the data that can be used to generate (denoised) reconstructions of the input data. This can be valuable for the inspection of very sparse data sets.
- **Feature set enrichment analysis**: when a factor is difficult to characterise based only on the inspection of the top weights, one can compute a statistical test for enrichment of biological pathways using predefined gene-set annotations.

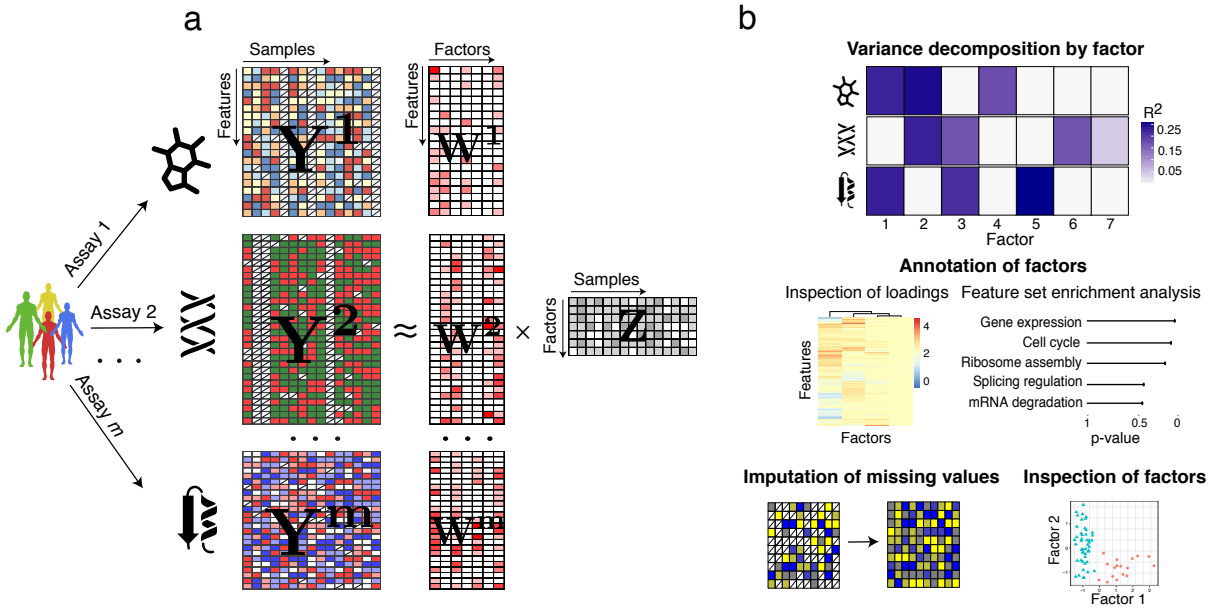


Figure 0.15: MOFA overview. The model takes M data matrices as input $(\mathbf{Y}^1, \cdots, \mathbf{Y}^M)$, one or more from each data modality, with co-occurrent samples but features that are not necessarily related and can differ in numbers. MOFA decomposes these matrices into a matrix of factors $(\mathbf{Z})$ and M weight matrices, one for each data modality $(\mathbf{W}^1, \cdots, \mathbf{W}^M)$. White cells in the weight matrices correspond to zeros, i.e. inactive features, whereas the cross symbol in the data matrices denotes missing values. The fitted MOFA model can be queried for different downstream analyses, including a variance decomposition to assess the proportion of variance explained by each factor in each data modality.

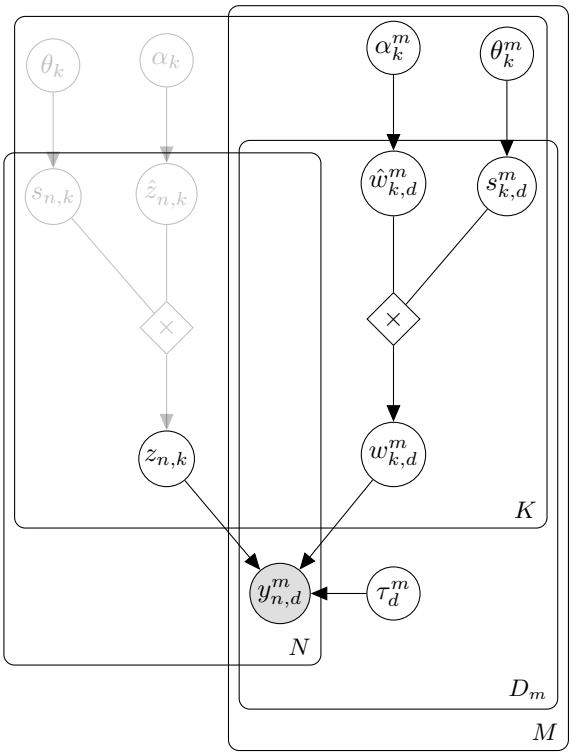


Figure 0.16: Graphical model for MOFA. The white circles represent hidden variables that are infered by the model, whereas the grey circles represent the observed variables. There are a total of four plates, each one representing a dimension of the model: M for the number of views, N for the number of samples, K for the number of factors and D_m for the number of features in the m-th view. The use of transparency in the top left nodes is intentional and becomes clear in Chapter 4 where we implement a spike-and-slab prior on the factors.

0.6.2.1 Inference

To make the model scalable to large data sets we adopt a Variational inference framework with a structured mean field approximation. A detailed overview is given in Section 0.2.4, and details on the variational updates for the MOFA model are given in **??**. To enable efficient inference for non-Gaussian likelihoods we employ local bounds [99, 207]. This is described in detail in Section 0.6.6

0.6.3 Model selection and consistency across random initilizations

The optimisation problem in MOFA is not convex and the resulting posterior distributions depend on the initialisation of the model. Thus, when doing random initialisation of the parameters and/or expectations it becomes mandatory to perform model selection and assess the consistency of the factors across different trials.

The strategy we adopted in this work is to train several instances MOFA models under different parameter initialisations, where the expectation of each node is randomly sampled from its underlying distribution. After fitting, we select the model with the highest ELBO for downstream analysis. In addition, we evaluate the robustness of the factors by plotting the Pearson correlations between factors across all trials:

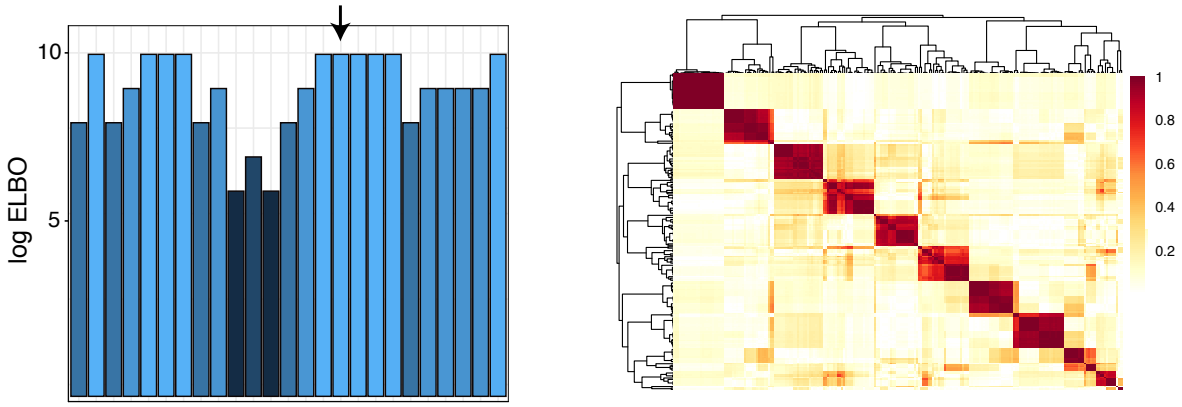


Figure 0.17: Model selection and robustness analysis in MOFA.
The left plot the log ELBO (y-axis) for 25 model instances (x-axis). The arrow indicates the model with the highest ELBO that would be selected for downstream analysis. The right plot displays the absolute value of the Pearson correlation coefficient between pairwise combinations of all factors across the 25 model instances. A block-diagonal matrix indicates that factors are robustly estimated regardless of the initialisation.

0.6.4 Learning the number of factors

As described in **??**, the use of an ARD prior allows factors to be actively prunned by the model if their variance explained is negligible. In the implementation we control the prunning of factors by a hyperparameter that defines a threshold on the minimum fraction of variance explained by a factor (across all views).
Additionally, because of the non-convexity of the optimisation problem, different model instances can potentially yield solutions with different number of active factors (Figure 0.18). Thus, the optimal number of factors can be selected by the model selection strategy outlined in Section 0.6.3.

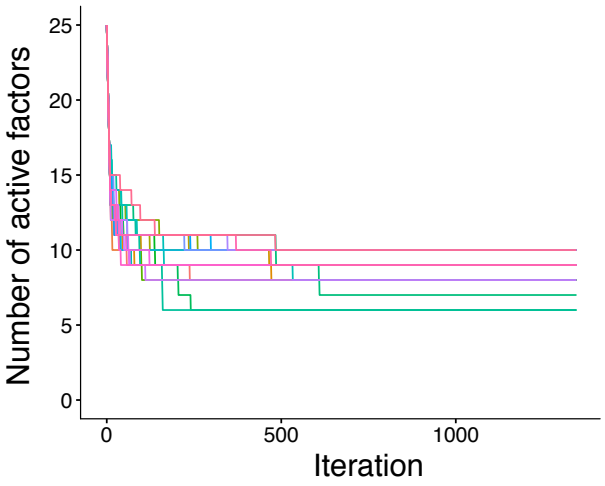


Figure 0.18: Training curve for the number of active factors across 25 different model instances.
The y-axis displays the number of active factors. The x-axis displays the iteration number. Different lines denote different model instances.

0.6.5 Monitoring convergence

An attractive property ofvVariational inference is that the objective function, the Evidence Lower Bound (ELBO), increases monotonically at every iteration. This provides a simple way of monitoring convergence:

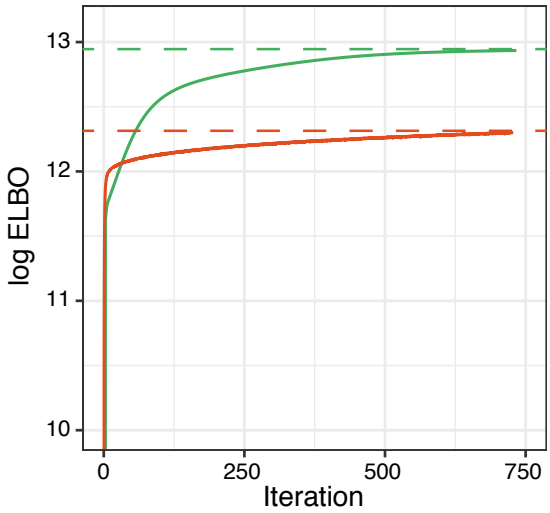


Figure 0.19: Training curve for two diffent initialisations of MOFA. The y-axis displays the log of the ELBO, with higher values indicating a better fit. The x-axis displays the iteration number. The horizontal dash lines mark the value of the ELBO upon convergence.

Training is stopped when the change in the lower bound becomes smaller than a predefined threshold. In MOFA we implemented

0.6.6 Modelling and inference with non-Gaussian data

To implement efficient variational inference in conjunction with a non-Gaussian likelihood we adapt prior work from [207] using local variational bounds. The key idea is to dynamically approximate non-Gaussian data by Gaussian pseudo-data based on a second-order Taylor expansion. To make the approximation justifiable we need to introduce variational parameters that are adjusted alongside the updates to improve the fit.

Denoting the parameters in the MOFA model as $\mathbf{X} = (\mathbf{Z}, \mathbf{W}, \boldsymbol{\alpha}, \boldsymbol{\tau}, \boldsymbol{\theta})$, recall that the variational framework approximates the posterior $p(\mathbf{X}|\mathbf{Y})$ with a distribution $q(\mathbf{X})$, which is indirectly optimised by optimising a lower bound of the log model evidence. The resulting optimization problem can be re-written as

$$\min_{q(\mathbf{X})} -\mathcal{L}(\mathbf{X}) = \min_{q(\mathbf{X})} \mathbb{E}_q\big[-\log p(\mathbf{Y}|\mathbf{X})\big] + \mathrm{KL}[q(\mathbf{X})||p(\mathbf{X})].$$

Expanding the MOFA model to non-Gaussian likelihoods we now assume a general likelihood of the form $p(\mathbf{Y}|\mathbf{X}) = p(\mathbf{Y}|\mathbf{C})$ with $\mathbf{C} = \mathbf{Z}\mathbf{W}^T$, that can write as

$$-\log p(\mathbf{Y}|\mathbf{X}) = \sum_{n=1}^{N}\sum_{d=1}^{D} f_{nd}(c_{nd})$$

with $f_{nd}(c_{nd}) = -\log p(y_{nd}|c_{nd})$. We dropped the view index m to keep notation uncluttered.
Extending [207] to our heteroscedastic noise model, we require $f_{nd}(c_{nd})$ to be twice differentiable and bounded by κ_d, such that $f''_{nd}(c_{nd}) \leq \kappa_d \, \forall n, d$. This holds true in many important models as for example the Bernoulli and Poisson case. Under this assumption a lower bound on the log likelihood can be constructed using Taylor expansion,

$$f_{nd}(c_{nd}) \leq \frac{\kappa_d}{2}(c_{nd} - \zeta_{nd})^2 + f'(\zeta_{nd})(c_{nd} - \zeta_{nd}) + f_{nd}(\zeta_{nd}) := q_{nd}(c_{nd}, \zeta_{nd}),$$

where $\boldsymbol{\zeta} = \zeta_{nd}$ are additional variational parameters that determine the location of the Taylor expansion and have to be optimised to make the lower bound as tight as possible. Plugging the bounds into above optimization problem, we obtain:

$$\min_{q(\mathbf{X}),\boldsymbol{\zeta}} \quad \sum_{d=1}^{D}\sum_{n=1}^{N} \mathbb{E}_q[q_{nd}(c_{nd}, \zeta_{nd})] + \mathrm{KL}[q(\mathbf{X})||p(\mathbf{X})]$$

The algorithm propsed in [207] then alternates between updates of $\boldsymbol{\zeta}$ and q($\boldsymbol{\Theta}$). The update for $\boldsymbol{\zeta}$ is given by

$$\zeta \leftarrow \mathbb{E}[\mathbf{W}]\mathbb{E}[\mathbf{Z}]^T$$

where the expctations are taken with respect to the corresponding q distributions.
On the other hand, the updates for $q(\mathbf{X})$ can be shown to be identical to the variational Bayesian updates with a conjugate Gaussian likelihood when replacing the observed data $\mathbf{Y}$ by a pseudo-data $\hat{\mathbf{Y}}$ and the precisions τ_{nd} (which were treated as random variables) by the constant terms κ_d introduced above.
The pseudodata is given by

$$\hat{y}_{nd} = \zeta_{nd} - f'(\zeta_{nd})/\kappa_d.$$

Depending on the log likelihoods $f(\cdot)$ different κ_d are used resulting in different pseudo-data updates. Two special cases implemented in MOFA are the Poisson and Bernoulli likelihood described in the following.

Bernoulli likelihood for binary data

When the observations are binary, $y \in \{0, 1\}$, they can be modelled using a Bernoulli likelihood:

$$\mathbf{Y}|\mathbf{Z}, \mathbf{W} \sim \mathrm{Ber}(\sigma(\mathbf{Z}\mathbf{W}^T)),$$

where $\sigma(a) = (1 + e^{-a})^{-1}$ is the logistic link function and $\mathbf{Z}$ and $\mathbf{W}$ are the latent factors and weights in our model, respectively.
In order to make the variational inference efficient and explicit as in the Gaussian case, we aim to approximate the Bernoulli data by a Gaussian pseudo-data as proposed in [207] and described above which allows to recycle all the updates from the model with Gaussian views. While [207]

assumes a homoscedastic approximation with a spherical Gaussian, we adopt an approach following [99], which allows for heteroscedaticity and provides a tighter bound on the Bernoulli likelihood. Denoting $c_{nd} = (\mathbf{Z}\mathbf{W}^T)_{nd}$ the Jaakkola upper bound [99] on the negative log-likelihood is given by

$$\begin{aligned} -\log\left(p(y_{nd}|c_{nd})\right) &= -\log\left(\sigma\left((2y_{nd}-1)c_{nd}\right)\right) \\ &\leq -\log(\zeta_{nd}) - \frac{(2y_{nd}-1)c_{nd} - \zeta_{nd})}{2} + \lambda(\zeta_{nd})\left(c_{nd}^2 - \zeta_{nd}^2\right) \\ &=: b_J(\zeta_{nd}, c_{nd}, y_{nd}) \end{aligned}$$

with λ given by $\lambda(\zeta) = \frac{1}{4\zeta}\tanh\left(\frac{\zeta}{2}\right)$.
This can easily be derived from a first-order Taylor expansion on the function $f(x) = -\log(e^{\frac{x}{2}} + e^{-\frac{x}{2}}) = \frac{x}{2} - \log(\sigma(x))$ in x^2 and by the convexity of f in x^2 this bound is global as discussed in [99]. In order to make use of this tighter bound but still be able to re-use the variational updates from the Gaussian case we re-formulate the bound as a Gaussian likelihood on pseudo-data $\hat{\mathbf{Y}}$.
As above we can plug this bound on the negative log-likelihood into the variational optimization problem to obtain

$$\min_{q(\mathbf{X}),\boldsymbol{\zeta}} \quad \sum_{d=1}^{D}\sum_{n=1}^{N} \mathbb{E}_q b_J(\zeta_{nd}, c_{nd}, y_{nd}) + \mathrm{KL}[q(\mathbf{X})||p(\mathbf{X})].$$

This is minimized iteratively in the variational parameter ζ_{nd} and the variational distribution of Z,W:
Minimizing in the variational parameter ζ this leads to the updates given by

$$\zeta_{nd}^2 = \mathbb{E}[c_{nd}^2]$$

as described in [99], [20].
For the variational distribution $q(\mathbf{Z}, \mathbf{W})$ we observe that the Jaakkola bound can be re-written as

$$b_J(\zeta_{nd}, c_{nd}, y_{nd}) = -\log\left(\varphi\left(\hat{y}_{nd}; c_{nd}, \frac{1}{2\lambda(\zeta_{nd})}\right)\right) + \gamma(\zeta_{nd}),$$

where $\varphi(\cdot; \mu, \sigma^2)$ denotes the density function of a normal distribution with mean μ and variance σ^2 and γ is a term only depending on $\boldsymbol{\zeta}$. This allows us to re-use the updates for $\mathbf{Z}$ and $\mathbf{W}$ from a setting with Gaussian likelihood by considering the Gaussian pseudo-data

$$\hat{y}_{nd} = \frac{2y_{nd} - 1}{4\lambda(\zeta_{nd})}$$

updating the data precision as $\tau_{nd} = 2\lambda(\zeta_{nd})$ using updates generalized for sample- and feature-wise precision parameters on the data.

Poisson likelihood for count data

When observations are a natural numbers, such as count data $y \in \mathbb{N} = \{0, 1, \cdots\}$, they can be modelled using a Poisson likelihood:

$$p(y|c) = \lambda(c)^y e^{-\lambda(c)}$$

where $\lambda(c) > 0$ is the rate function and has to be convex and log-concave in order to ensure that the likelihood is log-concave.
As done in [207], here we choose the following rate function: $\lambda(c) = \log(1 + e^c)$.

Then an upper bound of the second derivative of the log-likelihood is given by

$$f''_{nd}(c_{nd}) \leq \kappa_d = 1/4 + 0.17 * \max(\mathbf{y}_{:,d}).$$

The pseudodata updates are given by

$$\hat{y}_{nd} = \zeta_{nd} - \frac{\mathrm{S}(\zeta_{nd})(1 - y_{nd}/\lambda(\zeta_{nd}))}{\kappa_d}.$$

0.6.7 Model validation with simulated data

We used simulated data from the generative model to systematically test the technical capabilities of MOFA.

0.6.7.1 Recovery of simulated factors

First, we tested the ability of MOFA to recover simulated factors under varying number of views, features, factors and with different amounts of missing values.
For every simulation scenario we initialised a model with a high number of factors ($K = 100$), and inactive factors were automatically dropped during model training by the ARD prior. In addition, to test the robustness under different random initialisations, ten model instances were trained for every simulation scenario.
We observe that in most settings the model accurately recovers the correct number of factors (Figure 0.20). Exceptions occur when the dimensionality of the latent space is too large (more than 50 factors) or when an excessive amount of missing values (more than 80%) is present in the data.

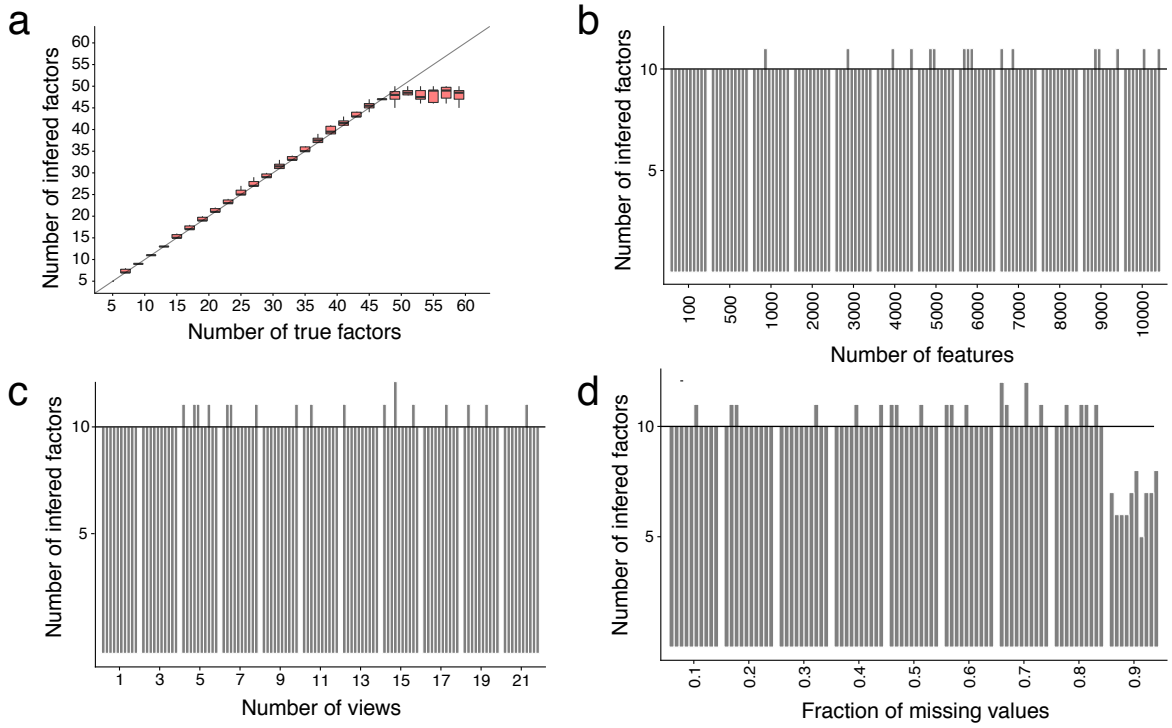


Figure 0.20: Assessing the ability to recover simulated factors.
In all plots the y-axis displays the number of infered factors. (a) x-axis displays the number of true factors, and boxplots summarise the distribution across 10 model instances. For (c-d) the true number of factors was set to $K = 10$ and each bar corresponds to a different model instance. (b) x-axis displays the number of features, (c) x-axis displays the number of views, (d) x-axis displays fraction of missing values.

0.6.7.2 View-wise sparsity on the weights

One of the most important statistical assumptions underlying MOFA is the ARD prior aimed at disentangling the activity of factors across views (see Section 0.4.4.1 and Section 0.6.1.
We simulated data from the generative model were the factors were set to be active or inactive in specific views by sampling α_k^m from a discrete distribution with values $\{1, 1e3\}$. We compared the performance with a popular integrative clustering method (iCluster) that is also formulated as a latent variable model [158]. In iCluster each factor shares the same sparsity constraint across all views, and hence the model is less accurate when it comes to the detection of factors that show differential activity across different views.:

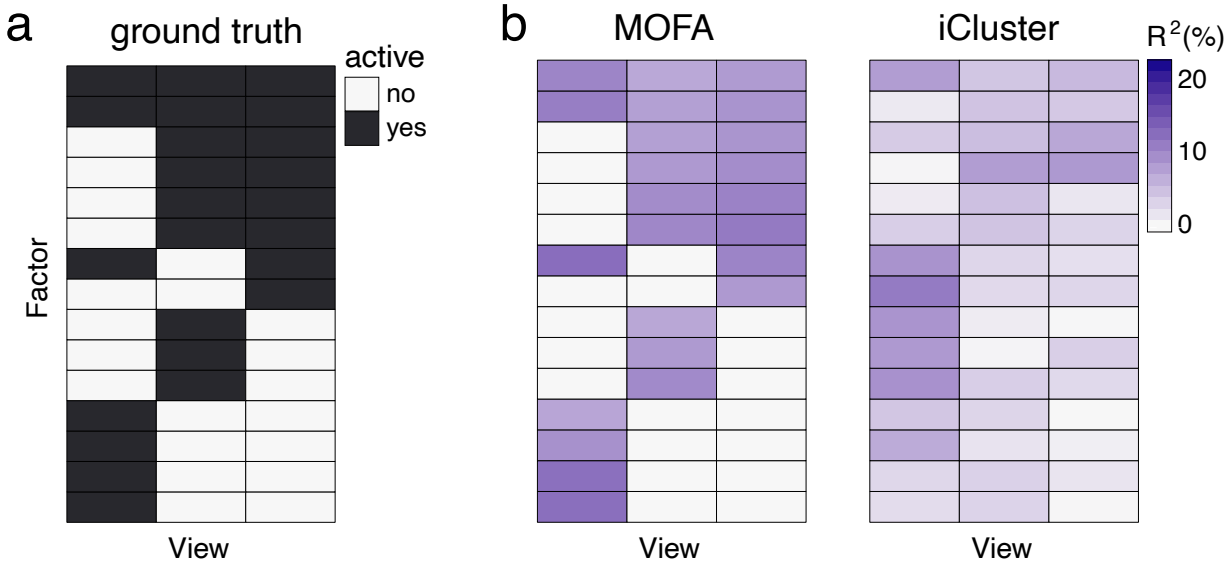


Figure 0.21: Evaluating the ability to recover differential factor activity across views. (a) The true activity pattern, with factors sampled to display differential activity across views. (b) Percentage of variance explained for each factor in each view, for MOFA and iCluster[158].

0.6.7.3 Feature-wise sparsity on the weights

In MOFA we implemented a spike-and-slab prior prior to enforce feature-wise sparsity on the weights with the aim of delivering a more interpretable solution (see Section 0.6.1.2).
To assess the effect of the spike-and-slab prior we trained a group of models with and without the spike-and-slab prior. Importantly, the model without spike-and-slab priors contains the ARD prior, which should provide some degree of regularisation. To compare both options to a non-spare method, we also fit a Principal Component Analysis on the concatenated data set.
As expected, we observe that the spike-and-slab prior induces more zero-inflated weights, although the ARD prior provided a moderate degree of regularisation. The PCA solution was notably more dense than both bayesian models (Figure 0.22).

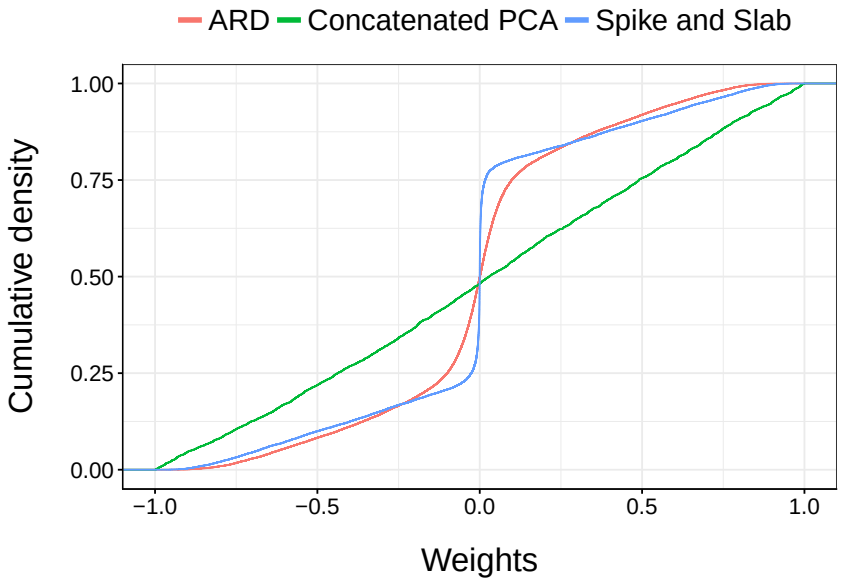


Figure 0.22: Assessing the sparsity priors on the weights.
The plot shows the empirical cumulative density function of the weights for an arbitrary factor in a single view. The weights were simulated with a sparsity level of $\theta_k^m = 0.5$ (50% of active features.)

0.6.7.4 Non-gaussian likelihoods

A key improvement of MOFA with respect to previous methods is the use of non-Gaussian likelihoods to integrate data modalities with different types of readouts. In particular, as described in Section 0.6.6, we implemented a Bernoulli likelihood to model binary data and a Poisson likelihood to model count data.

To validate both likelihood models, we simulated binary and count data using the generative model and we fit two sets of models for each data type: a group of models with a Gaussian likelihood and a group of models with a Bernoulli or Poisson likelihood, respectively.

Reassuringly, we observe that although the Gaussian likelihood is also able to recover the true number of factors, the models with the non-Gaussian likelihoods result in a better fit to the data:

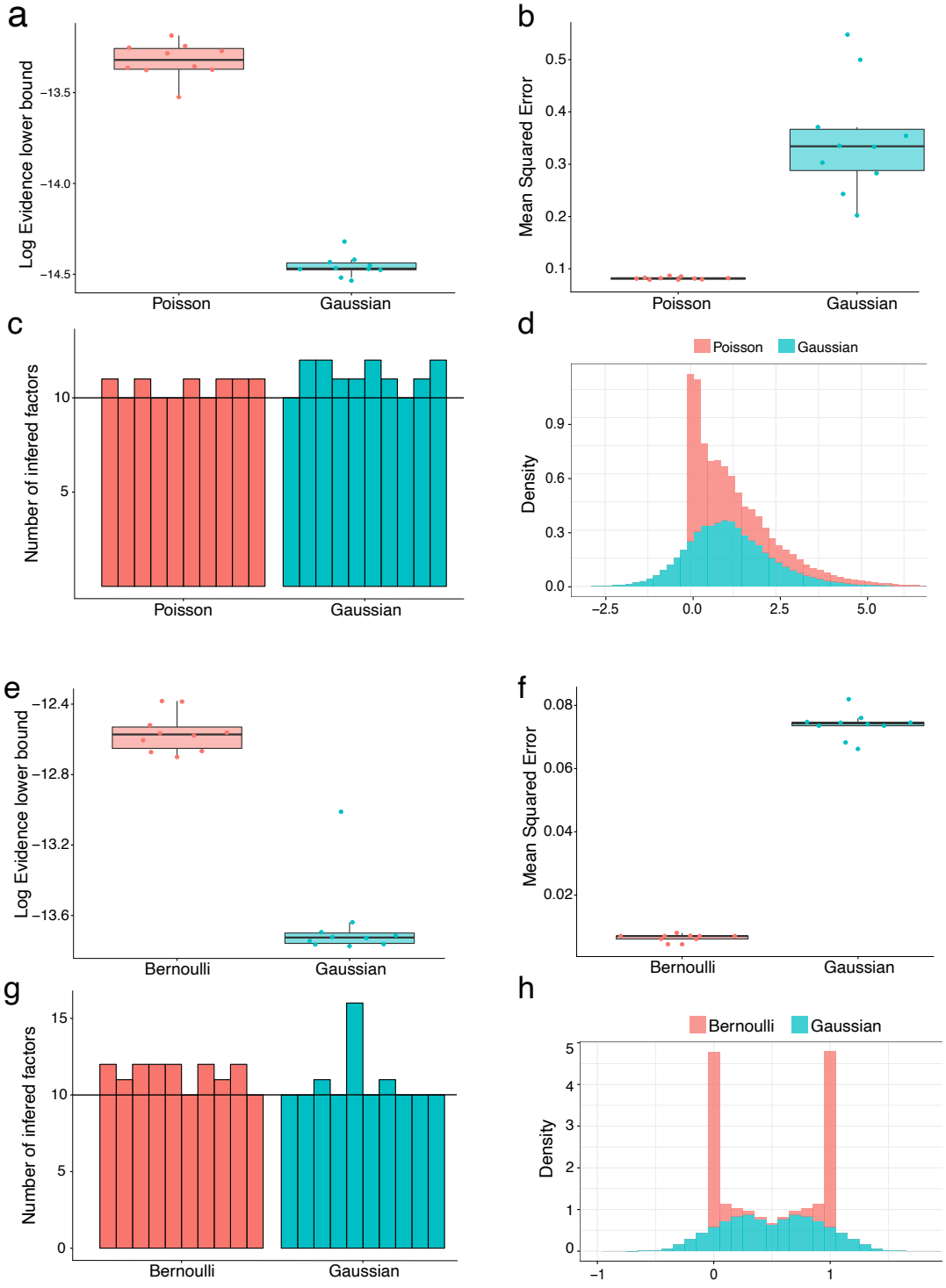


Figure 0.23: Validation of the non-gaussian likelihood models using simulated data.
(a-d) Comparison of Poisson and Gaussian likelihood models applied to count data.
(e-h) Comparison of Bernoulli and Gaussian likelihood models applied to binary data.
(a,e) The y-axis displays the ELBO for each model instance (x-axis). (b,f) The y-axis displays the mean reconstruction error for each model instance (x-axis). (c,g) The y-axis displays the number of estimated factrors for each model instance (x-axis). The horizontal dashed line marks the true number of factors $K = 10$. (d,h) Distribution of reconstructed data. Plotted are the expected values of the inferred posterior distributions, not samples from the corresponding posteriors. This is why reconstructed measurements are continuous and not discrete.

0.6.8 Application to chronic lymphocytic leukaemia

Personalised medicine is an attractive field for the use of multi-omics, as dissecting heterogeneity across patients is a major challenge in complex diseases, and requires data integration from multiple biological layers [40, 47, 2].

To demonstrate the potential of the method, we applied MOFA to a publicly available study of 200 patient samples of chronic lymphocytic leukaemia (CLL) profiled for somatic mutations, RNA expression, DNA methylation and *ex vivo* drug responses[55]. We selected this data set for three main reasons: (1) The complex missing data structure, with nearly 40% samples having incomplete assays (see **??**). As described in Section 0.6.1.5, the inference framework implemented in MOFA should cope with large amounts of missing values, including missing assays. (2) The different data modalities: after data processing, three assays had continuous observations whereas for the somatic mutations the observations were binary. As described in Section 0.6.6, MOFA can combine different likelihood models. (3) The existence of clinical covariates: this provides an excellent test to evaluate whether the MOFA factors can capture the variation underlying clinically-relevant phenotypes.

0.6.8.1 Data overview and processing

Here we proceed to briefly describe the different data modalities and outline the basic data processing steps that we performed before applying MOFA.

- **RNA expression** was profiled using bulk RNA-seq. Genes with low counts were filtered out and the data was subsequently normalized using DESeq2 [**Love2014**]. Feature selection was performed by considering the top 5,000 most variable genes.
- **DNA methylation** was profiled using Illumina 450K arrays [**XXX**]. We converted the beta-values to M-values, as it has better statistical properties when modelled with a Gaussian distribution [**XXX**]. Feature selection was performed by considering the top 1% most variable CpG sites.
- ***Ex vivo* Drug response** was screened using the ATP-based CellTiter-Glo assay. Briefly, the asay includes a panel of 62 drugs at 5 different concentrations each, for a total of 310 measurements.
- **TO FINISHSomatic mutations** were profiled using a combination of targeted and whole exome sequencing. Feature selection was performed by considering only mutations that were present in at least three samples, which resulted in a total of 69 mutations.

For more details on the data generation steps we refer the reader to [55].

0.6.8.2 Model overview

In this data set, MOFA recovered $K = 10$ factors, each one explaining a minimum of 3% of variance in at least one assay. Interestingly, MOFA detects Factors which are shared across several

data modalities (Factors 1 and 2, sorted by variance explained). Some factors capture sources of covariation between two data modalities (Factor 3 and 5, active in the RNA expression and drug response). In addition, some factors capture variation that is unique to a single data modality (Factor 4, active in the RNA expression data).
All together, the 10 MOFA factors explained 41% of variance in the drug response data, 38% in the mRNA expression, 24% in the DNA methylation and 24% in somatic mutations.

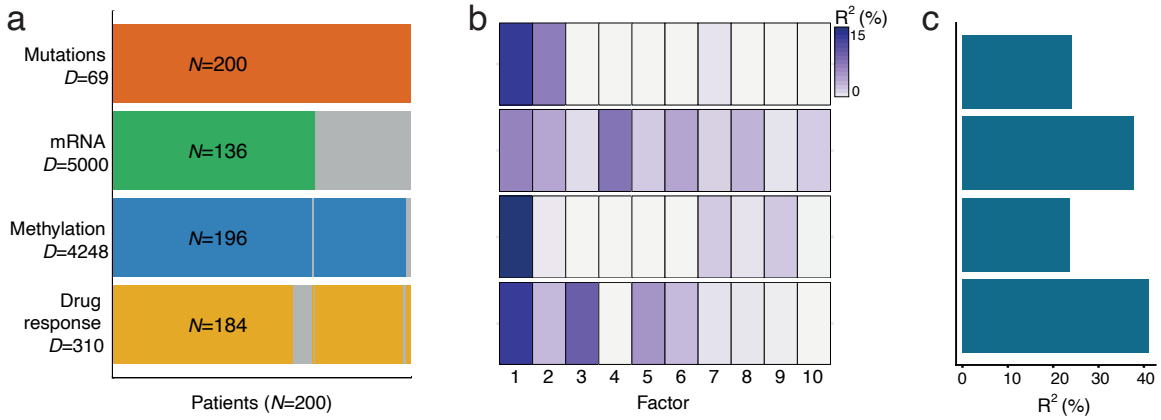


Figure 0.24: Application of MOFA to a study of chronic lymphocytic leukaemia. Model overview.
(a) Data overview. Assays are shown in different rows (D = number of features) and samples (N) in columns, with missing samples shown using grey bars. Notice that some samples are missing entire assays.
(b) Variance explained (%) by each Factor in each assay.
(c) Total variance explained (%) for each assay by all Factors.

The first two Factors are the most interesting from a molecular perspective, as they capture a phenotypic effect that is manifested across all molecular layers, from the genome to the transcriptome and ultimately in the drug response assay.
To annotate Factors 1 and 2 we proceeded to visualise the feature weights, starting by the (binary) somatic mutation data, as it is the simplest data modality to interpret. Inspection of the top weights revealed that Factor 1 was associated with the mutation status of the immunoglobulin heavy-chain variable (IGHV) region, while Factor 2 was aligned with trisomy of chromosome 12 (Figure 0.25). Remarkably, in a completely unsupervised fashion, MOFA recovered the two most important clinical markers in CLL as the two major axes of molecular disease heterogeneity [62, 31, 50].

Next, we visualised the samples in the latent space spanned by Factors 1 and 2. A scatterplot based on these factors shows a clear separation of patients by their IGHV status on the first Factor and presence or absence of trisomy 12 on the second Factor (Figure 0.25). Note that this latent representation enables simple patient stratification into molecular subgroups (see dashed lines), a first step towards personalised medicine.
Interestingly, 24 patients lacked IGHV status measurements (see grey crosses) due to quality control filtering in the DNA sequencing assay. Nonetheless, MOFA is able to pool information from the other molecular layers to map those samples to the latent space, and could be classified to the corresponding molecular subgroup.

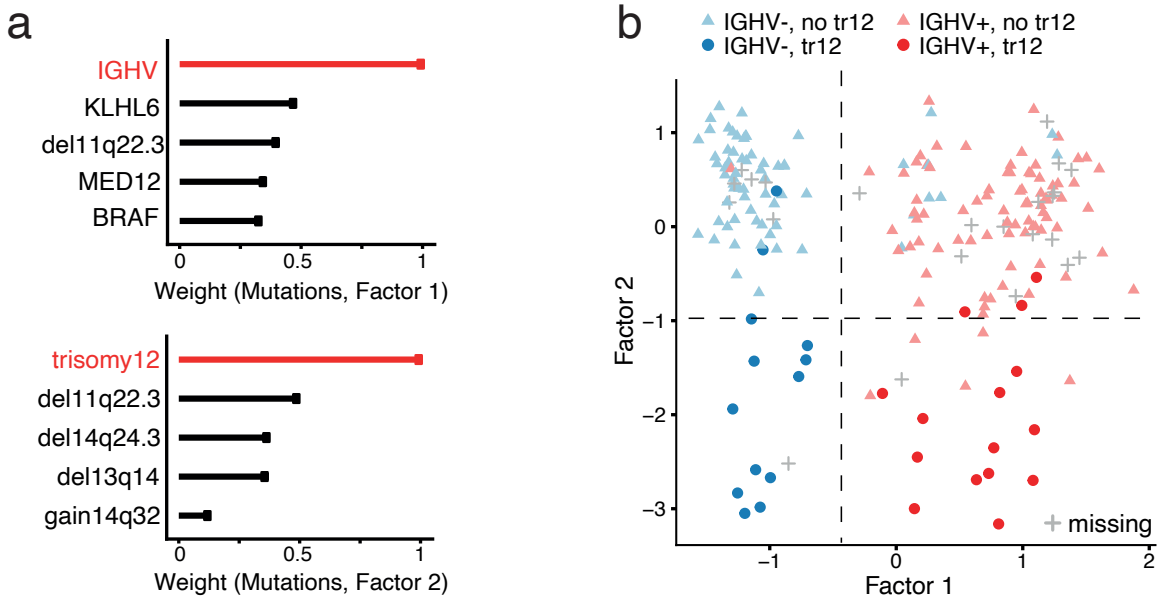


Figure 0.25: Visualisation of the genetic signature underlying Factor 1 and 2
(a) Absolute loadings of the top features of Factors 1 and 2 in the Mutations data. (b) Visualization of samples using Factors 1 and 2. The colours denote the IGHV status of the tumours; symbol shape and colour tone indicate chromosome 12 trisomy status.

IGHV status is currently the most important prognostic marker in CLL and has routinely been used to distinguish between two distinct subtypes of the disease[62]. Molecularly, it is a surrogate of the level of activation of the B-cell receptor, which is in turn related to the differentiation state of the tumoral cells. Multiple studies have associated mutated IGHV with a better resposne to chemoimmunotherapy, whereas unmutated IGHV patients have a worse prognosis [62, 31, 50].
In clinical practice, the IGHV status has been considered binary. Our results suggest that this is a fairly good approximation, but a more complex structure with at least three groups or a potential underlying continuum (Figures 0.25 and 0.26), as also suggested in [183].

0.6.8.3 Molecular characterisation of Factor 1

An important step in the MOFA pipeline is the characterisation of the molecular signatures underlying each Factor. I will demonstrate this for Factor 1, although a similar strategy can be applied to Factor 2.

On the RNA expression, inspection of the top weights pinpoint genes that have been previously associated to IGHV status, some of which have been proposed as clinical markers[237, 160]. Heatmaps of the RNA expression levels for these genes reveals clear differences between samples when ordinated according to the Factor 1 values.

On the drug response data the weights highlight kinase inhibitors targeting the B-cell receptor pathway. Splitting the patients into three groups based on k-means clustering shows clear separation in the drug response curves.

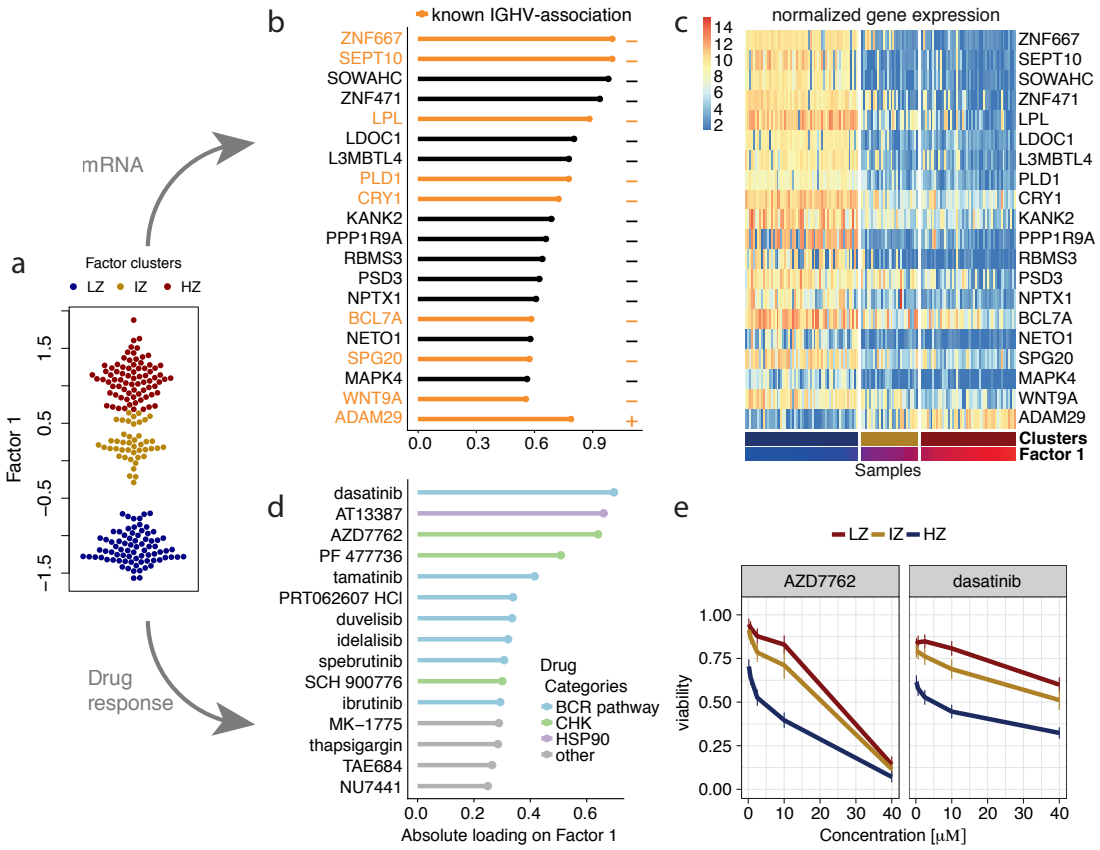


Figure 0.26: Characterization of MOFA Factor 1 as IGHV status.
(a) Beeswarm plot with Factor 1 values for each sample with colours corresponding to three groups found by 3-means clustering with low factor values (LZ), intermediate factor values (IZ) and high factor values (HZ).
(b) Absolute weights for the genes with the largest absolute weights in the mRNA data. Plus or minus symbols on the right indicate the sign of the loading. Genes highlighted in orange were previously described as prognostic markers in CLL and associated with IGHV status (Vasconcelos et al, 2005; Maloum et al, 2009; Trojani et al, 2012; Morabito et al, 2015; Plesingerova et al, 2017).
(c) Heatmap of gene expression values for genes with the largest weights as in (b).
(d) Absolute weights of the drugs with the largest weights, annotated by target category.
(e) Drug response curves for two of the drugs with top weights, stratified by the clusters as in (a).

0.6.8.4 Characterisation of other Factors

Despite their clinical importance, Factor 1 (IGHV status) and Factor 2 (chr12 trisomy) they explain less than 20% variability in each data modality, suggesting the existence of more subtle sources of variation. As an example, we will also characterise Factor 5, which explains 2% of the variance in the mRNA and 6% of variance in the drug response.

As mentioned in Section 0.6.2, instead of exploring the feature weights individually, factors can be annotated using gene set annotations. This procedure is particularly appealing for RNA expression data, as a rich amount of resources exist that have categorised genes into ontologies in terms of biological pathways, molecular function and cellular components [**Fabregat2015**, **Ashburner2000**].

Briefly, the idea is to aggregate the weights using prior information to obtain a single statistic for each gene set, which can be tested against a competitive null hypothesis. Inspired from [69],

in MOFA we implemented several scoring schemes and a variety of parametric and unparametric statistical tests. We refer the reader to [69] for details.

Appling Gene Set Enrichment Analysis on the RNA weights using the Reactome annotations [**Fabregat2015**] reveals that Factor 2 is strongly enriched for oxidative stress and senescence pathways. Inspection of the top features highlights the importance of heat shock proteins (HSPs), a group of proteins that are essential for protein stability which are up-regulated upon stress conditions like high temperatures, pH shift or oxidative stress. Importantly, HSPs can be elevated in tumour cells and potentially contribute to prolonged tumour cell survival[53].
In agreement with the findings from the mRNA view, the drugs with largest weights on Factor 5 belong to clinical categories associated with stress response, such as target reactive oxygen species (SD07, MIS-43, SD51) and DNA damage response (fludarabine, nutlin-3, doxorubicine) (Figure S12c-d).

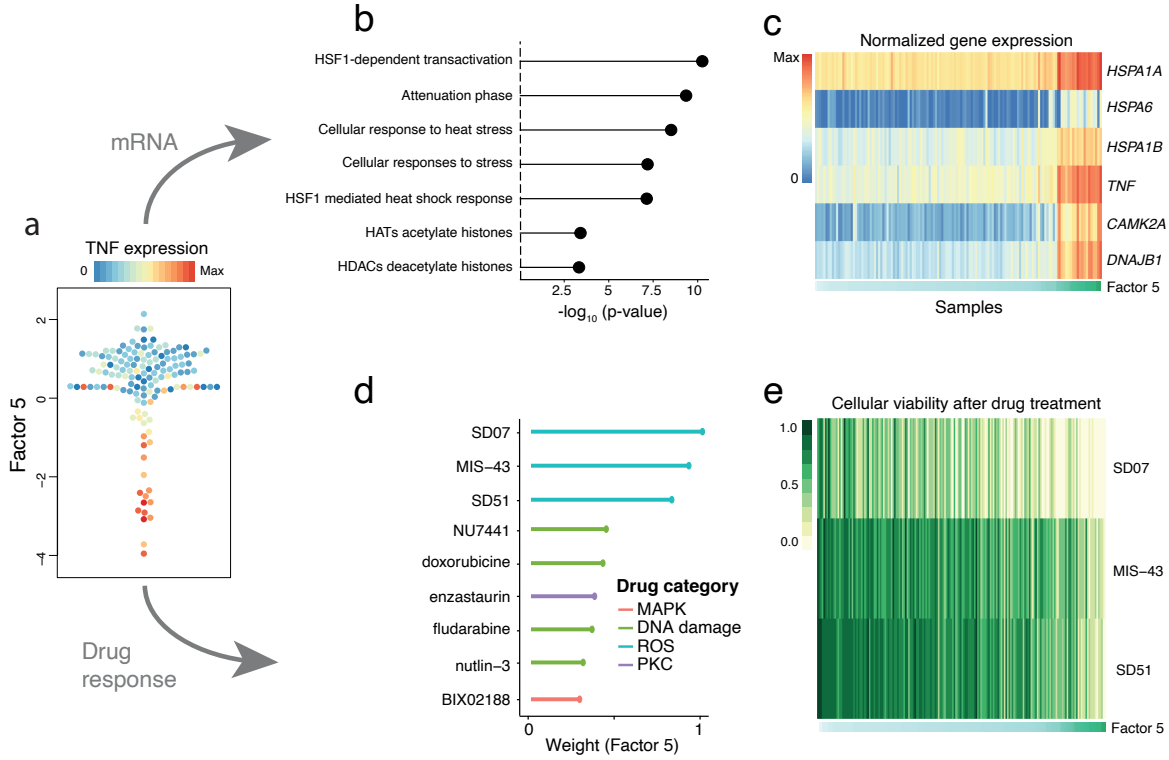


Figure 0.27: Characterization of Factor 5 in the CLL data as oxidative stress response.
(a) Beeswarm plot of Factor 5. Colours denote the expression of TNF, an inflammatory stress marker that is present among the top RNA weights.
(b) Gene set enrichment analysis for the top Reactome pathways. Displayed are the top pathways with the strongest enrichment in the RNA weights. P-values were adjusted for multiple testing using the Benjamini-Hochberg procedure.
(c) Heatmap of mRNA expression values for representative genes with the largest weights. Samples are ordered by their factor values.
(d) Scaled weights for the top drugs with the largest loading, annotated by target category.
(e) Heatmap of drug response values for the top three drugs with largest weight.

0.6.8.5 Prediction of clinical outcomes

We conjectured that the integration of multiple molecular layers could allow an improved prediction of the patients' clinical outcome.
To evaluate the utility of the MOFA factors as predictors of clinical outcomes we fit Cox regression models [48] using the patients' time to next treatment (TTT) as a response variable. Two types of analysis were performed: a univariate analysis where each Factor was independently associated with TTT, and a multivariate analysis where the combination of all Factors were used to predict TTT (Figure 0.28).
In the univariate Cox models, we observe that Factor 1 (IGHV status), Factor 7 (associated with chemo-immunotherapy treatment prior to sample collection) and Factor 8 (enriched for Wnt signalling) were significant predictors of TTT. Accordingly, when splitting patients into binary groups based on the corresponding Factor values, we observe clear differences in the survival curves. In the multivariate Cox model, MOFA (Harrell's C-Index C=0.78) outperformed all other input settings, including PCA on single-omic data (C=0.68-0.72), individual genetic markers (C=0.66) as well PCA applied to the concatenated data matrix (C=0.74).

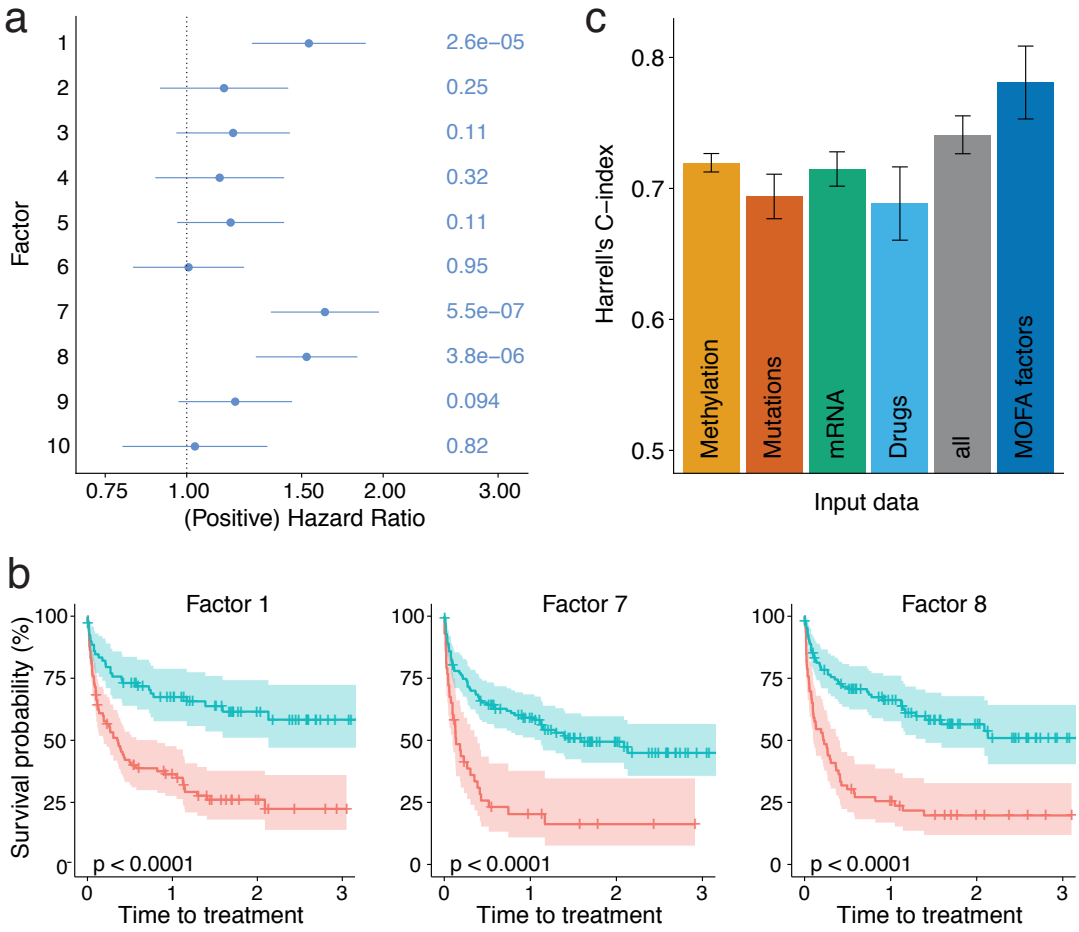


Figure 0.28: Association analysis between MOFA Factors and clinical putcome.
(a) Association of MOFA factors to time to next treatment using a univariate Cox regression with N = 174 samples (96 of which are uncensored cases) and p-values based on the Wald statistic. Error bars denote 95% confidence intervals. Numbers on the right denote p-values for each predictor.
(b) Kaplan-Meier plots measuring time to next treatment for the individual MOFA factors. The cut-points on each factor were chosen using maximally selected rank statistics, and p-values were calculated using a log-rank test on the resulting groups.
(c) Prediction accuracy of time to treatment for N = 174 patients using multivariate Cox regression trained with the 10 MOFA factors, as well using the first 10 principal components applied to single data modalities and the full data set. Shown are average values of Harrell's C-index from fivefold cross-validation. Error bars denote standard error of the mean.

0.6.8.6 Imputation of missing values

A promising application of MOFA is the imputation of missing values, including the potential to impute of entire assays.

The principle of imputation in MOFA follows the same logic as simulating from the generative model: if the factors and weights are known, the input data can be reconstructed by a simple matrix multiplication:

$$\hat{\mathbf{Y}} = \mathbb{E}[\mathbf{Z}]\mathbb{E}[\mathbf{W}]^T$$

where $\mathbb{E}[\mathbf{Z}]$ and $\mathbb{E}[\mathbf{W}]$ denote the expected values of the variational distributions for the factors and the weights, respectively. Notice that, when using the expectations of the posterior distributions,

the noise ϵ (Equation (20)) has a mean of zero and does not contribute to the predictions.
The equation above computes a point estimate for every sample n and feature d, but it ignores the uncertainity on $\mathbf{Z}$ and $\mathbf{W}$. Instead of relying in point estimates, one could adopt a more Bayesian approach and calculate the posterior predictive distribution by propagating the uncertainity [72]. Nonetheless, due to the nature of the optimisation problem in variational inference, the variance of the posterior distributions can be underestimated (see Section 0.2.5). In addition, this would more complex to implement and would result in a significant increase in computational complexity. Hence, and also because of the additional computational complexity, we did not attempt this approach.

To assess the imputation performance, we trained MOFA models using a data set of complete measurements (a total of N=121 samples) after masking parts of the drug response measurements. In a first experiment, we masked values at random, and in a second experiment we masked the entire drug response measurements. We compared the imputation accuracy of MOFA to some established imputation strategies, including imputation by feature-wise mean, SoftImpute [151], a k-nearest neighbour method [234].
For both imputation tasks, MOFA consistently yielded more accurate predictions, albeit the differences are less pronounced in the imputation of full assays, a significantly more challenging task.

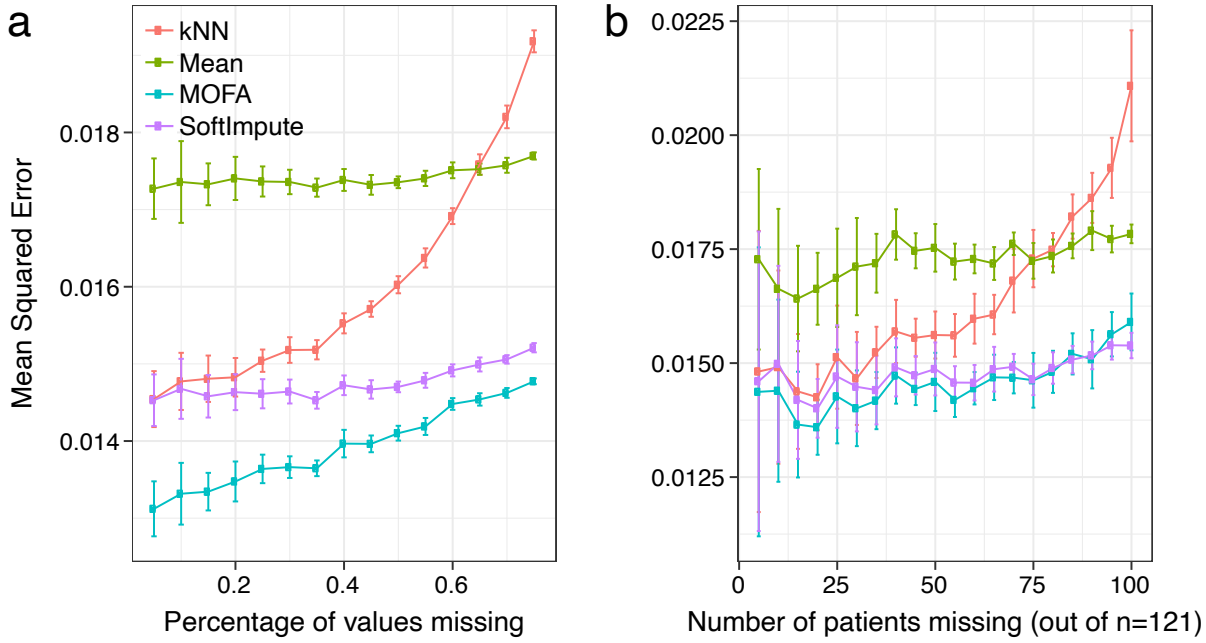


Figure 0.29: Evaluation of imputation performance in the drug response assay.
The y-axis shows the mean-squared error (MSE) across 15 trials for increasing fractions of missing data (x-axis). Two experiments were considered: (a) values missing at random and (b) entire assays missing at random. Each point displays the mean across all trials and the error bars depict the corresponding standard deviations.

0.6.9 Application to single-cell multi-omics

The emergence of single-cell multi-modal techniques has created open opportunities for the development of novel computational strategies [222, 45, 38].
To show case how MOFA can be used to integrate single-cell multi-omics data, we considered a simple data set that consists on 87 ESCs where RNA expression and DNA methylation were simultaneously measured using single-cell Methylation and Transcriptome sequencing (scM&T-seq)[6]. Two populations of ESCs were profiled: the first one contains 16 cells grown in 2i media, which is known to induce a native pluripotency state associated with genome-wide DNA hypomethylation [66]. The second population contains 71 cells grown in serum media, which contain stimuli that trigger a primed pluripotency state poised for differentiation [232].

0.6.10 Data processing

The RNA expression data was processed using *scran*[145] to obtain log normalised counts adjusted by library size. Feature selection was performed by selecting the top 5,000 most overdispersed genes[126]. A Gaussian likelihod was used for this data modality.
The DNA methylation data was processed as described in Chapter 1. Briefly, for each CpG site, we calculated a binary methylation rate from the ratio of methylated read counts to total read counts. Next, CpG sites were classified by overlapping with genomic contexts, namely promoters, CpG islands and enhancers (distal H3K27ac peaks). Finally, for each annotation we selected the top 5,000 most variable CpG sites with a minimum coverage of 10% across cells. Each of the resulting matrices was defined as a separate view for MOFA. A Bernoulli likelihod was used for this data modality.

0.6.10.1 Model overview

In this data set, MOFA inferred 3 factors with a minimum explained variance of 1% (Figure 0.30). Factor 1 captured the transition from naive to primed pluripotent states, which MOFA links to widespread coordinated changes between DNA methylation and RNA expression. Inspection of the gene weights for Factor 1 pinpoints important pluripotency markers including *Rex1/Zpf42* or *Essrb* [159]. As previously described both *in vitro* [6] and *in vivo* [10], the transition from naive to primed pluripotency state is concomitant with a genome-wide increase in DNA methylation levels. Factor 2 captured a second dimension of heterogeneity driven by the transition from a primed pluripotency state to a differentiated state, with RNA weights enriched with canonical differentiation markers including keratins and annexins [70].
Jointly, the combination of Factors 1 and 2 reconstruct the coordinated changes between the transcriptome and the epigenome along the differentiation trajectory from naive pluripotent cells to differentiated cells.

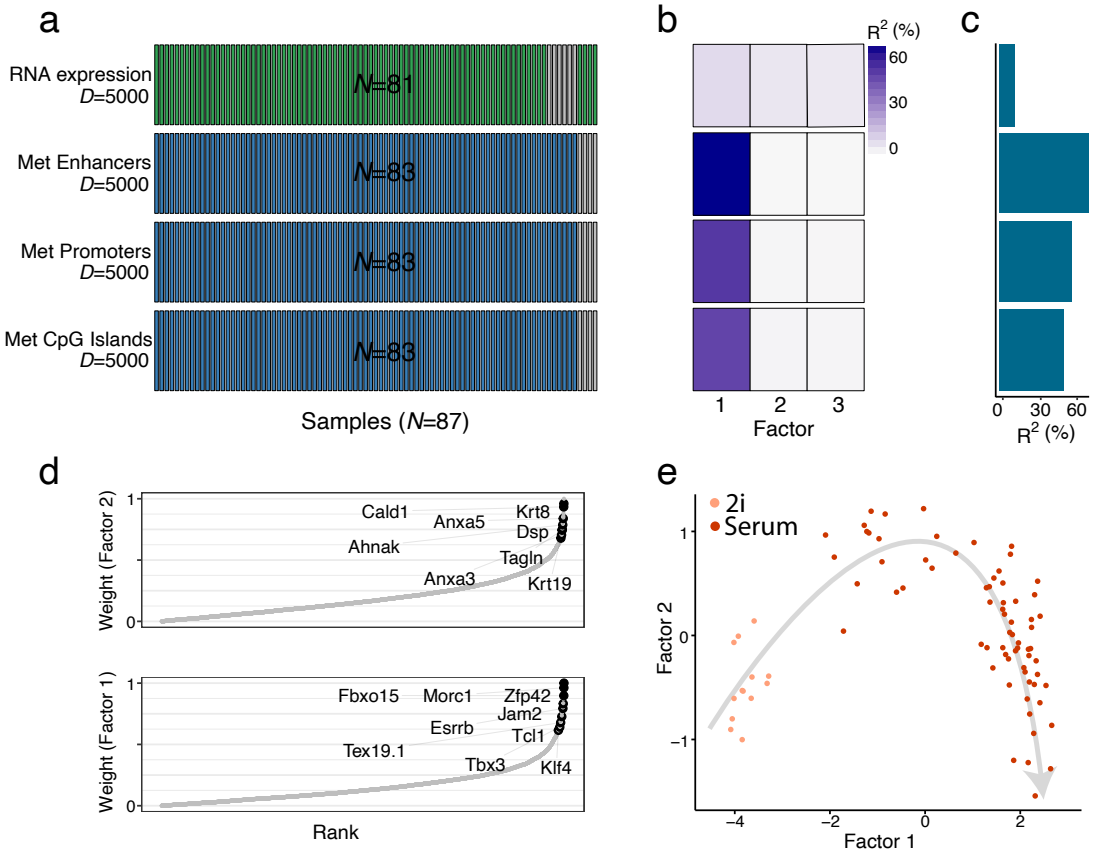


Figure 0.30: MOFA recovers a differentiation process from a single-cell multi-omics data set.
(a) Overview of the data modalities. Rows indicate number of features (D) and columns indicate number of samples (N). Grey bars denote missing samples.
(b) Fraction of variance explained per factor (column) and view (row).
(c) Cumulative fraction of variance explained per view (across all factors).
(d) mRNA weights of Factor 1 (bottom) and Factor 2 (top). The genes that are labelled are known markers of pluripotency (for Factor 1) or differentiation (for Factor 2).
(e) Scatter plot of Factor 1 (x-axis) against Factor 2 (y-axis). Cells are colored based on the culture condition. Grey arrow illustrates the differentiation trajectory from a naive pluripotency state to a differentiated state.

0.6.11 Limitations and open perspectives

MOFA solves important challenges for the integrative analysis of (single-cell) multi-omics data sets. Yet, the model is not free of limitations and there are open possibilities for future research:

- **Linearity**: this is an assumption that is critical for obtaining interpretable feature weights. Nonetheless, there is a trade-off between explanatory power and interpretability[125]. Non-linear approaches, including deep neural networks or variational autoencoders have shown promising results when it comes to dimensionality reduction [138, 57, 144], batch correction[144], denoising [61] or imputation [139]. Interestngly, very few multi-view factor analysis models exist that incorporate flexible non-linear assumptions, making it an interesting line of research to explore.

- **Scalability**: the size of biological datasets is rapidly increasing, particularly in the field of single cell sequencing [223, 36].
  When comparing the inference framework to previous methods that make use of sampling-based MCMC approaches, the variatonal framework implemented in MOFA yields a vast improvement in scalability. Yet, in its vanilla form, variational inference also becomes prohibitively slow with very large datasets [90, 22, 91]. This has been recently addressed by a reformulation of the variational inference problem in terms of a gradient descent optimisation problem, which enables the full machinery of stochastic inference to be applied in the context of Bayesian inference.

- **Generalisations to multi-group structures**: the sparsity assumptions in MOFA are based on the principle that features are structured into non-overlapping views. As such, the activity of the latent factors is also expected to be structured, so that different factors explain variability in different subsets of views (Figure 0.15). Following the same logic, many studies contain structured samples, as either multiple experiments or conditions. A simple generalisation of MOFA would be to intuitively break the assumption of independent samples and introduce an additional prior that captures the group structure at the sample level.

- **Tailored likelihoods for single-cell analysis**: MOFA enables the modular extension to arbitrary non-gaussian likelihoods, provided that they can be locally bounded and integrated into the variational framework (see Section 0.6.6). New likelihood models such as zero-inflated negative binomial distributions [196] could make MOFA more suited to the analysis of single-cell data.

- **Bayesian treatment of predictions**: in the current implementation of MOFA, only the point estimates for the posterior distributions are used in the downstream analysis. While convienient for most operations, this ignores the uncertainity associated with the point estimates, which is a major strength of Bayesian modelling. Future extensions could attempt a more comprehensive Bayesian treatment that propagates uncertainity in the downstream analyses, mainly when it comes to making predictions and imputation [72].

- **Incorporation of prior information**: an unsupevised approach is appealing for discovering the principal axes of variation, but sometimes this can yield challenges in the interpretation of

factors. Future extensions could exploit the rich information encoded in gene set ontologies, similar to the methodology proposed in [30].