Poisson log-normal estimation with dependency between sites

Camille Mondon[1], Julien Chiquet[2], Mahendra Mariadassou[3], Stéphane Robin[4], Bertrand Servin[5]

[1]École normale supérieure
[1]Université Paris-Saclay
[2]MIA Paris-Saclay, AgroParisTech, INRAE
[3]MaIAGE, Université Paris-Saclay, INRAE
[4]CESCO, LPSM, Sorbonne Université
[5]GenPhySE, INRAE

September 10, 2023

The classical version of the PLN model assumes that n independent count vectors (each of dimension p) are observed. The aim of this course is to extend the PLN model to take into account a dependency between count vectors. Depending on the application, it may be assumed that the dependency structure between observations is free or takes a specific form. The associated inference algorithm will then need to be developed. This algorithm could eventually be integrated into the `R/C++` `PLNmodels` package. This subject is motivated in particular by an application to population genetics, where the p counts making up a vector are derived from p possibly related individuals, and where each vector of counts is associated with a locus along the genome. The aim here is to take into account the dependency between neighboring loci (known as linkage disequilibrium).

Keywords: Poisson, log-normal, EM, variational, Generalised Linear Mixed Models, Graphical Models, Genomics.

Acknowledgments: I would like to express my gratitude to my internship supervisors Julien Chiquet and Mahendra Mariadassou, for putting great effort into providing guidance whenever needed. I am also very grateful to Stéphane Robin and Bertrand Servin, with whom I have worked all along this internship, as well as to the whole team of the UMR Mathématique et Informatique Appliquées (Paris-Saclay) for their support and their kindness. Finally, I want to thank Emma, who has been there the whole time.

Table of contents

Introduction

This report summarises the advancements that have been made during my research internship at UMR Mathématique et Informatique Appliquées (Paris-Saclay), between April 2023 and July 2023, under the supervision of Julien Chiquet, Mahendra Mariadassou, Stéphane Robin and Bertrand Servin.

The goal was to design a multivariate count model adapted to a dataset on crossing-overs, which are breaks all along the DNA strand at the time of meiosis, and for multiples species of sheep. The MIA team has developed the Poisson log-normal model, which is accurate for counting multiple variables and measure their correlations.

But in this specific dataset, there is a clear spatial dependency that lacks in the classical PLN model where the sites are independent. So we wanted to design a PLN model where the latent variables have an auto-regressive structure, and implement in `R` into the `PLNmodels` package.

Notations

Throughout this report, it will be made use of the following notations:

- Let $\mathrm{KL}(P\|Q) = \int \log \frac{dP}{dQ} dP$.
- Let $\mathbf{Z} = (Z_1, \ldots, Z_n)$ and $\mathbf{Y} = (Y_1, \ldots, Y_n)$ be i.i.d. samples.
- Let $p(X;\theta)$ denote the Radon-Nikodym derivative $\frac{dP_\theta^X}{d\mu}$.
- Let $\ell(\theta; X) = \log p(X;\theta)$ the log-likelihood.
- $\mathbb{E}_\theta[f(Y,Z)] = \mathbb{E}_{(Y,Z)\sim P_\theta}[f(Y,Z)]$
- $\mathbb{E}_\psi[f(Y,Z)|Y] = \mathbb{E}_{Z|Y\sim Q_\psi}[f(Y,Z)|Y]$
- $\mathbb{E}_\psi[f(Z)] = \mathbb{E}_{Z\sim Q_\psi}[f(Z)]$
- In order to write more compact expressions, we will use as often as possible the convention that non-linear functions (mainly log, exp, ! and ^{2}) applied to vector are actually applied coefficient-wise.

1 Reminders

This section is dedicated to a comprehensive overview of the prerequisites to the inference of a Poisson log-normal model with auto-regressive Gaussian latent vectors. The theoretical context is that of **generalized linear models** (GLM) with **latent variables**, and **graphical models**.

1.1 Generalised linear models & latent variables

1.1.1 Exponential families

The exponential families form very a classical, but broad, class of statistical models. They are at the core of Generalised Linear Models, and their inference is usually carried out through Maximum Likelihood Estimation (MLE). The concept of exponential families is credited to E. J. G. Pitman, G. Darmois, and B. O. Koopman in 1935–1936.

Definition 1.1 (Exponential families (Murphy 2023), Section 2.4)**.** Let μ be a σ-finite measure on a measurable space $\mathscr{Y} \subseteq \mathbb{R}^p$. Consider a family $P_\theta, \theta \in \mathbb{R}^d$ of probability measures on $\mathscr{Y}$. This family $P_\theta, \theta \in \mathbb{R}^d$ is an **exponential family** if its densities with respect to μ can be written in the following way:

$$p_\theta(y) = \frac{1}{Z(\theta)} h(y) \exp\left[\langle \theta, T(y) \rangle\right] = h(y) \exp\left[\langle \theta, T(y) \rangle - A(\theta)\right]$$

where:

- the function $Z : \mathbb{R}^d \to \mathbb{R}_+^*$ (resp. $A : \theta \in \mathbb{R}^d \mapsto \log Z(\theta)$) is called the **partition function** (resp. **log-partition function**);
- the function $T : \mathscr{Y} \to \mathbb{R}^d$ is the **sufficient statistic**;
- the function $h : \mathscr{Y} \to \mathbb{R}$ is the **base measure**, often we will have $h = 1$.

Example 1.1. Many statistical models using usual distributions are exponential families. Among them:

- Bernoulli family
- Binomial $\mathscr{B}(n, p)$ family with known n
- Multivariate Gaussian family
- Exponential family
- Poisson family
- Geometric family

but there are also non-examples:

- Uniform distributions with unknown bounds
- Cauchy family (hence logistic)
- Hyper-geometric family
- Student family
- Most family of mixtures.

The intuition is that we must be able to separate the parameter and the variable by factorisation.

Proposition 1.1 (Log-partition function and cumulants)**.** *Let $Y : \Omega \to \mathbb{R}^p$ be a random variable in an exponential family. Then:*

$$\nabla A(\theta) = \mathbb{E}_\theta[T(Y)]$$
$$\nabla^2 A(\theta) = \mathrm{Cov}_\theta[T(Y)]$$

1.1.2 Generalised linear models

The exponential families can be used to generalise the classical linear regression model with normally distributed error, conditionally on the covariates.

Let $X : \Omega \to \mathscr{X} \subseteq \mathbb{R}^d$ and $Y : \Omega \to \mathscr{Y} \subseteq \mathbb{R}^p$ be random variables. A **generalised linear model** consists of three elements:

- An unknown **parameter matrix** $\mathbf{B} = (\beta_1|\ldots|\beta_p) \in \mathbb{R}^{d\times p}$;
- A **dispersion parameter** σ;
- An over-dispersed **natural exponential family** modelling the distribution of Y conditionally on X, whose parameter is the linear predictor $\theta = \mathbf{B}^\top X$ and sufficient statistic is the identity.

The notion of generalised linear model was first investigated by Wedderburn and Nelder (1972). For more details on the matter, see Dobson and Barnett (2018).

Definition 1.2 (GLM (Murphy 2023), Chapter 15)**.** A generalized linear model or GLM is the combination of two variables $X : \Omega \to \mathscr{X} \subseteq \mathbb{R}^d$ and $Y : \Omega \to \mathscr{Y} \subseteq \mathbb{R}^p$ with the following conditional density for $Y|X$ parametered by $\mathbf{B} \in \mathbb{R}^{d\times p}, \sigma > 0$:

$$p_{\mathbf{B},\sigma}(y \mid x) = h(y, \sigma^2) \exp\left[\frac{\langle \mathbf{B}^\top x, y\rangle - A(\mathbf{B}^\top x)}{\sigma^2}\right]$$

By Proposition 1.1, we have the following link between the log-partition function and the conditional expectation $\mathbb{E}[Y|X]$:

$$\mathbb{E}[Y|X] = \nabla A(\mathbf{B}^\top X)$$

The **link function** g such that $g(\mathbb{E}[Y|X]) = \mathbf{B}^\top X$ verifies:

$$g^{-1} = \nabla A$$

Example 1.2 (Poisson regression)**.** If we have $Y \in \mathbb{N}$ we can use:

$$p_\beta(y|x) = e^{-\mu}\frac{\mu^y}{y!} \text{ where } \mu = e^{\langle x,\beta\rangle}$$

Then $A(\langle x,\beta\rangle) = \mu = e^{\langle x,\beta\rangle}$ and $h(y) = -\log(y!)$.

1.1.3 Latent variables & Expectation-Maximisation (EM)

When the observed data is complete, the estimation of a GLM can be carried out through simple MLE (Murphy 2023, Section 15.1.3): the calculation are explicit and the optimisation often relies on the Iteratively Reweighted Least Squares algorithm or Stochastic Gradient Descent.

However, for the Poisson log-normal model we need **incomplete data models**, i.e. **hidden variables**. They have studied extensively in Robin (2018). In addition to the the observed response variable Y, we assume that there a hidden variable Z.

Then, the calculation of the MLE is not explicit anymore, but in some cases the classical **Expectation-Maximisation** algorithm (Dempster, Laird, and Rubin 1977) solves this issue.

The EM algorithm is an iterative procedure that can be described as follows:

- **(Initialisation)** Choose $\widehat{\theta}^{(0)} \in \Theta$.
- **(E-step)** For each $\theta \in \Theta$, compute the conditional **expectation** of the complete log-likelihood conditionally on the observed data Y with the current estimate $\widehat{\theta}^{(h)}$ of the parameters:

$$Q(\theta, \widehat{\theta}^{(h)}) := \mathbb{E}_{\widehat{\theta}^{(h)}}[\ell(\theta; Y, Z)|Y]$$

- **(M-step)** Find, if it exists, a new estimator $\widehat{\theta}^{(h+1)} \in \Theta$ of θ that **maximises** said expectation:

$$\widehat{\theta}^{(h+1)} \in \underset{\theta\in\Theta}{\operatorname{argmax}}\, Q(\theta, \widehat{\theta}^{(h)})$$

This way we construct a family $(\widehat{\theta}^{(h)})_h$ of estimators of θ.

1.2 Poisson log-normal model

In this section, we define the Poisson log-normal distribution and model. We will see that the Expectation-Maximisation algorithm is unavailable, since we are unable to produce a closed form of the moments of $p_\theta(Z|Y)$. This comes from the fact that PLN is not a GLM, but rather a GLMM (Generalised linear mixed model).

1.2.1 Definition

Definition 1.3 (PLN distribution (Aitchison and Ho 1989), Section 2)**.** For $\mu \in \mathbb{R}^p$ and a $p \times p$-positive-definite matrix $\mathbf{\Sigma}$, the **Poisson log-normal distribution**, denoted $\mathrm{PLN}(\mu, \mathbf{\Sigma})$ is a discrete distribution on $\mathbb{N}^p$ whose pmf is:

$$p(y) = \int_{\mathbb{R}^p} \prod_{j=1}^{j=p} f\left(y_j; \exp z_j\right) g\left(z; \mu, \mathbf{\Sigma}\right) dz$$

where $f(\cdot;\lambda)$ is the pmf of the Poisson $\mathscr{P}(\lambda)$ and $g\left(\cdot;\mu,\mathbf{\Sigma}\right)$ is the pdf of the Gaussian $\mathcal{N}_p\left(\mu,\mathbf{\Sigma}\right)$ distribution.

Remark.

- It is the distribution of Y when $Z \sim \mathcal{N}_p(0, \mathbf{\Sigma})$:

$$Y_j|Z \overset{\perp}{\sim} \mathscr{P}\left(\exp(\mu_j + Z_j)\right), 1 \leq j \leq p$$

- As a multivariate regression model, with $\mu = \mathbf{B}^\top X$, it is not a GLM but Generalised Linear Mixed Model because of the **latent variable** Z.

1.2.1.1 Comparing Poisson and PLN pmfs

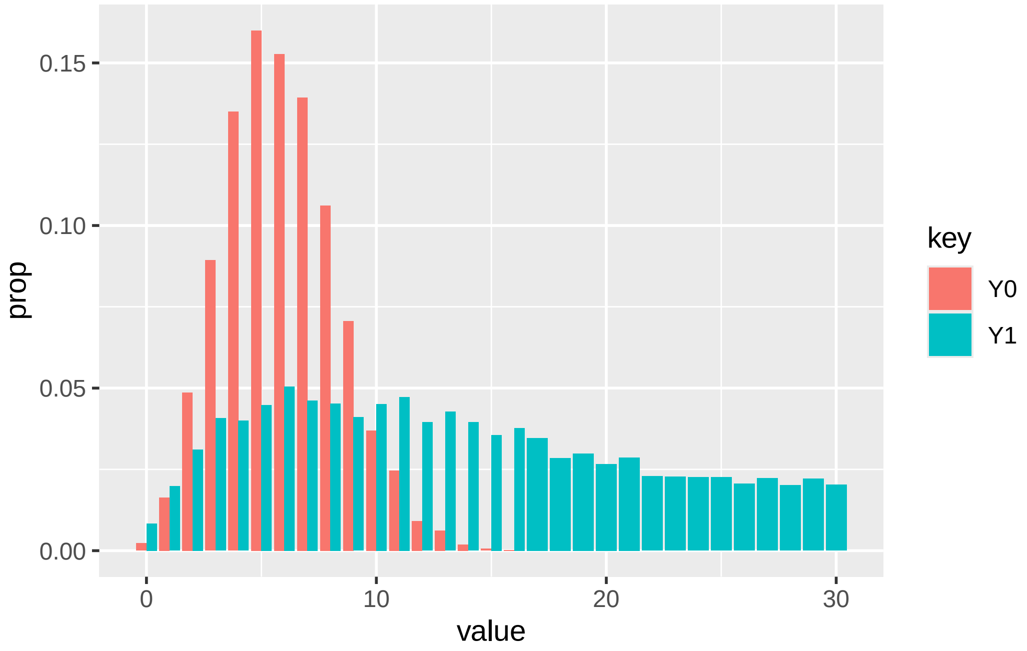


Figure 1.1: Comparing PLN and Poisson pmfs with $Y_1 \sim \mathrm{PLN}(\mu = \log(20), \sigma = 1)$, $Y_2 \sim \mathscr{P}(\lambda = \mathrm{mode}(\mathrm{PLN}(\mu, \sigma)))$.

Remark. There is clearly an over-dispersion phenomenon, compared to the Poisson distribution.

1.2.1.2 Example of multivariate PLN data

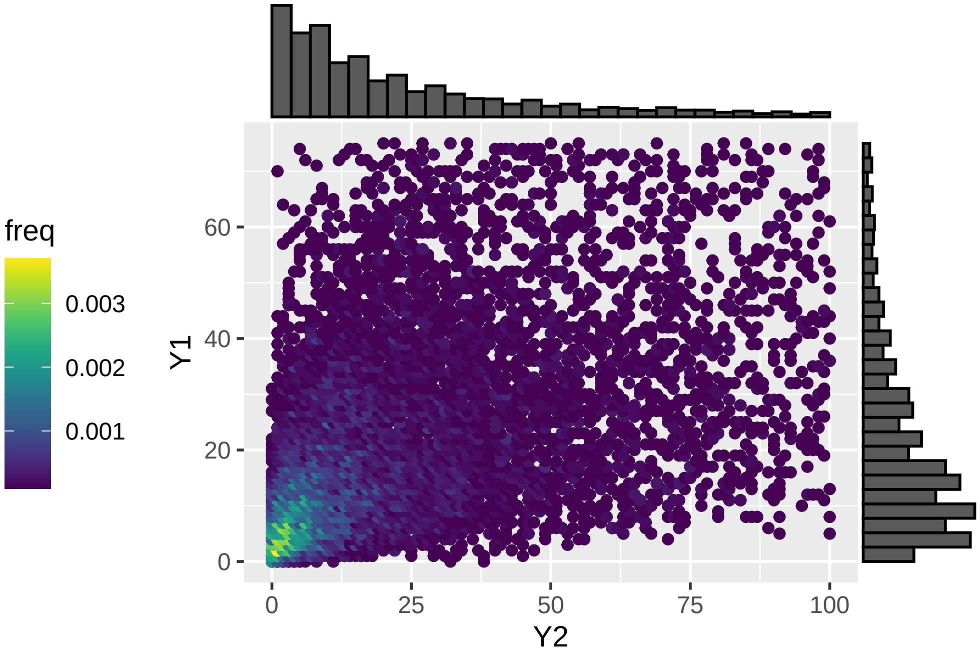


Figure 1.2: Multivariate PLN $\left(\mu = \begin{pmatrix} \log(20) \\ \log(20) \end{pmatrix}, \mathbf{\Sigma} = \begin{pmatrix} 1 & 1 \\ 1 & 2 \end{pmatrix}\right)$.

1.2.1.3 Remarks

Remark. When modelling real data, the regression layer is corrected by the **sampling effort** (offset): ratio of sample over whole population.

Why is the Poisson log-normal useful for **abundance tables** (e.g Joint Species Distribution Models)?

- **Poisson**: we count occurrences (**sites**), which happen **rarely** and **independently** (for instance number of animals appearing at a particular site):

  *The name 'law of rare event' may be misleading because the **total count of success** events in a Poisson process need not be small if the parameter np is not small. For example, the number of telephone calls to a busy switchboard in one hour follows a Poisson distribution with the events appearing frequent to the operator, but they are **rare** from the point of view of **one** average member of the population who is very unlikely to make a call to that switchboard in that hour.*

- **Multivariate**: At each site, we want to count **multiple species** simultaneously (abundance vector) while accounting for their correlation: we need a multivariate distribution whose marginal are Poisson distribution and whose **covariance matrix** is **arbitrary**, which is not easy to design (without forcing non-negative covariances).

- **Log-normal**: log is the link function, normal because we want to make some calculations.

1.2.2 Properties

There are simple expression for the first two cumulants of the PLN distribution:

- **Expectations:**
$$\mathbb{E}Y_{ij} = \mathbb{E}\mathbb{E}[Y_{ij}|Z_{ij}] = \mathrm{e}^{\mu_{ij}+\sigma_{jj}/2}$$
- **Variances:**
$$\operatorname{Var} Y_{ij} = \mathbb{E}\operatorname{Var}(Y_{ij}|Z_{ij}) + \operatorname{Var}(\mathbb{E}[Y_{ij}|Z_{ij}]) = \mathbb{E}Y_{ij} + \mathbb{E}[Y_{ij}]^2 \left(\mathrm{e}^{\sigma_{ij}/2} - 1\right)$$

Or they be written using a matrix notation:

$$\mathbb{E}[Y] = \exp\left(\mu + \frac{1}{2}\operatorname{diag}\mathbf{\Sigma}\right)$$

where $\operatorname{diag}\mathbf{\Sigma} \in \mathbb{R}^p$ denotes the vector of diagonal entries of $\mathbf{\Sigma}$ and exp is taken entry-wise.

$$\operatorname{Var}(Y) = \Delta_Y + \Delta_Y \left(\exp\mathbf{\Sigma} - J\right)\Delta_Y$$

where $\Delta_Y = \operatorname{diag}\mathbb{E}[Y]$ is the $p \times p$-diagonal matrix whose entries are the vector $\mathbb{E}[Y]$.

We notice that, if Z is i.i.d, Y is decorrelated.

Also, we prove the over-dispersion phenomenon noticed earlier: $\operatorname{Var} Y_{ij} > \mathbb{E}Y_{ij}$.

1.3 Generalised linear mixture models

1.3.1 Definition

A generalised linear mixture model is a type of hierarchical model. It generalises the GLM approach by adding a random effect Z to the fixed effect $\mathbf{B}^\top X$.

Definition 1.4 (GLMM)**.** A Generalised linear mixed model is the combination of three variables X, Y and $Z \sim \mathcal{N}_p(0, \mathbf{\Sigma})$ such that X and Z are independent and, conditionally on X and Z, Y has its distribution in an exponential family of the form:

$$p_\beta(y \mid x, z) = h(y)\exp\left[\frac{\langle \mathbf{B}^\top x + z, y\rangle - A(\mathbf{B}^\top x + z)}{\sigma^2}\right]$$

1.3.2 Variational Expectation-Maximisation (VEM)

The EM algorithm is actually a special case of the **Minimisation-Maximisation** or MM algorithm, which is a general iterative procedure where the E-step is replaced by a minimisation step over the set of all possible distributions for $Z|Y$.

If we restrict this set to some smaller parametric set $\mathcal{Q} = \left\{Q_\psi, \psi \in \Psi \subseteq \mathbb{R}^N\right\}$ of **variational distributions** to approximate the conditional distribution of $Z|Y$, we get the **variational EM** or VEM algorithm. This induces lower-bounding the log-likelihood by the **evidence lower-bound** (ELBO):

$$J(\theta, q; Y) = \mathbb{E}_q[\log p_\theta(Y, Z) - \log q(Z)]$$

The VEM algorithm is described as follows:

- **(Initialisation)** Choose $(\widehat{\theta}^{(0)}, \psi^{(0)})$.
- **(VE-step)** Compute the ELBO with current estimates $(\widehat{\theta}^{(h)}, \psi^{(h)})$ of the parameters:

$$\psi^{(h+1)} \in \underset{\psi \in \Psi}{\operatorname{argmin}} J(\widehat{\theta}^{(h)}, q_\psi, Y)$$

- **(M-step)** Find the new estimated parameters $\widehat{\theta}^{(h+1)}$ that **maximise** said ELBO.

$$\widehat{\theta}^{(h+1)} \in \underset{\theta \in \Theta}{\operatorname{argmax}} J(\theta, q_{\psi^{(h)}}, Y)$$

If we choose the parametric set of Gaussian distributions as set of variational distributions:

$$\mathcal{Q} = \left\{Q_\psi = Q_{\psi_1} \dots Q_{\psi_n} \middle| Q_{\psi_i} = \mathcal{N}_p(m_i, \mathbf{S}_i), \psi_i = (m_i, \mathbf{S}_i) \in \mathbb{R}^p \times \mathbb{R}^{p \times p}, 1 \leq i \leq n\right\}$$

then the parameter m (resp. $\mathbf{S}$) can be seen as an estimator of the conditional expectation $\mathbb{E}[Z|Y]$ (resp. conditional variance $\operatorname{Var}[Z|Y]$).

1.3.2.1 Example

Let us show, using the `PLNmodels` package, an implementation of the VEM algorithm to data generated with the `rPLN` function:

```
#>
#>  Initialization...
#>  Adjusting a full covariance PLN model with nlopt optimizer
#>  Post-treatments...
#>  DONE!
#> A multivariate Poisson Lognormal fit with full covariance model.
#> ==================================================================
#>  nb_param    loglik       BIC       ICL
#>        27 -270347.4 -270471.8 -262044.8
#> ==================================================================
#> * Useful fields
#>     $model_par, $latent, $latent_pos, $var_par, $optim_par
#>     $loglik, $BIC, $ICL, $loglik_vec, $nb_param, $criteria
#> * Useful S3 methods
#>     print(), coef(), sigma(), vcov(), fitted()
#>     predict(), predict_cond(), standard_error()
#> $B
#>                  Y1       Y2       Y3      Y4       Y5     Y6
#> (Intercept) 3.00499 3.025345 3.002611 3.08441 2.984871 2.9841
#> 
#> $Sigma
#>              Y1           Y2            Y3           Y4            Y5
#> Y1  1.035912834  1.003874981 -0.0198748300  0.014942778  0.0055759893
#> Y2  1.003874981  1.987567251 -0.0505681767  0.039993709  0.0064235659
#> Y3 -0.019874830 -0.050568177  1.2876012592 -0.003206137 -0.0008336345
#> Y4  0.014942778  0.039993709 -0.0032061369  3.030113571 -0.0020246097
#> Y5  0.005575989  0.006423566 -0.0008336345 -0.002024610  1.0350827029
#> Y6 -0.004381428  0.008718679 -0.0061865016  0.005713716 -0.0002457630
#>              Y6
#> Y1 -0.004381428
#> Y2  0.008718679
#> Y3 -0.006186502
#> Y4  0.005713716
#> Y5 -0.000245763
#> Y6  0.401051489
#> 
#> $Omega
#>              Y1            Y2            Y3            Y4            Y5
#> Y1  1.891616953 -0.9558513792 -0.0081376559  0.0031984026 -0.0042487685
#> Y2 -0.955851379  0.9868191819  0.0238293823 -0.0082272795 -0.0009792384
#> Y3 -0.008137656  0.0238293823  0.7775045218  0.0005271768  0.0005258753
#> Y4  0.003198403 -0.0082272795  0.0005271768  0.3301229112  0.0006789047
#> Y5 -0.004248769 -0.0009792384  0.0005258753  0.0006789047  0.9661372278
#> Y6  0.041271715 -0.0314113424  0.0113794236 -0.0044808605  0.0005653564
#>               Y6
#> Y1  0.0412717153
#> Y2 -0.0314113424
#> Y3  0.0113794236
#> Y4 -0.0044808605
#> Y5  0.0005653564
#> Y6  2.4948189004
#>
#> $Theta
#>    (Intercept)
#> Y1    3.004990
#> Y2    3.025345
#> Y3    3.002611
#> Y4    3.084410
#> Y5    2.984871
#> Y6    2.984100
```

1.3.3 Convergence of the VEM estimator

It was shown by the PLN team that, as a **M-estimator**, and under some classical assumptions:

- (A1) Assume that the parameter space Θ of θ is compact;
- (A2) Assume that the variational parameter space $\boldsymbol{\Psi}$ of the variational parameters ψ is bounded;

then the VEM estimator is:

- **consistent** around an an **unknown parameter** $\bar{\theta}$ which might differ from the original parameter θ
- but simulations suggest it is nearly **unbiased** according to the original parameter θ.

Theorem 1.1 (Consistency of $\hat{\theta}$)**.** *Under assumptions* $(A1)-(A2)$, *assume that the map* $\theta \to \mathbb{E}[\sup_{\psi\in\Psi} J(\theta, q_\psi, Y)(\theta; Y)]$ *attains a finite global maximum at* $\bar{\theta}$ *(which can be different from the true parameter* $\theta^\star$*). Then* $\hat{\theta} \to \bar{\theta}$ *under* $P_{\theta^\star}$*.*

Another general approach can be found in Gunawardana and Byrne (2005) where they define the notion of **Generalised Alternating Minimisation** procedures.

1.4 Auto-regressive (AR) multivariate Gaussian processes

In this subsection, we define the stationary multivariate auto-regressive distribution. The notion dates back to Udny Yule and Gilbert Walker, who designed the AR process of order q around 1930.

Definition 1.5 (AR(q))**.**

$$Z_i = \sum_{k=1}^{q} \varphi_k Z_{i-k} + \varepsilon_i$$

where the $\varepsilon_i, 1 \le i \le n$ are i.i.d. Gaussian variables.

We use the multivariate generalisation in dimension p.

Definition 1.6 (AR$_p(q)$)**.**

$$Z_i = \sum_{k=1}^{q} \mathbf{\Phi}_k Z_{i-k} + \varepsilon_i$$

where the $\varepsilon_i, 1 \leq i \leq n$ are i.i.d. multivariate Gaussian variables in $\mathbb{R}^p$.

We restrict to the particular case of order $q = 1$ where we can easily define a stationarity condition so that every vector has the same first two moments.

Definition 1.7 (SAR$(\mu, \mathbf{\Sigma}, \mathbf{\Phi})$)**.** If the parameters $\mathbf{\Sigma}$ and $\mathbf{\Phi}$ verify the stationarity condition:

$$\mathbf{\Sigma}_\varepsilon = \mathbf{\Sigma} - \mathbf{\Phi}\mathbf{\Sigma}\mathbf{\Phi}^\top \succ 0$$

then the centered random vectors $Z_1, \ldots, Z_n : \Omega \to \mathbb{R}^p$ are said to follow the SAR$(\mu, \mathbf{\Sigma}, \mathbf{\Phi})$ distribution if they follow an AR$_p(1)$ distribution:

$$Z_1 \sim \mathcal{N}_p(\mu, \mathbf{\Sigma}) \quad \text{and} \quad Z_i = \mathbf{\Phi} Z_{i-1} + \varepsilon_i$$

where $\mu_\varepsilon = \mu - \mathbf{\Phi}\mu$ and the $\varepsilon_i \sim \mathcal{N}_p(\mu_\varepsilon, \mathbf{\Sigma}_\varepsilon), 1 \leq i \leq n$ are i.i.d.

Remark.

- We have $Z_i \sim \mathcal{N}_p(\mu, \mathbf{\Sigma}), 1 \leq i \leq n$.
- If $\mathbf{\Phi} = \mathbf{0}$, $Z_i = \varepsilon_i, 1 \leq i \leq n$ are i.i.d.
- If $\mathbf{\Phi} = I_p$, $Z_1 = \cdots = Z_n$ are equal.

One can easily compute the full covariance matrix:

$$\begin{pmatrix} \mathbf{\Sigma} & \mathbf{\Sigma}\mathbf{\Phi}^\top & \cdots & \mathbf{\Sigma}\left(\mathbf{\Phi}^n\right)^\top \\ \mathbf{\Phi}\mathbf{\Sigma} & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & \mathbf{\Sigma}\mathbf{\Phi}^\top \\ \mathbf{\Phi}^n\mathbf{\Sigma} & \cdots & \mathbf{\Phi}\mathbf{\Sigma} & \mathbf{\Sigma} \end{pmatrix}$$

from which we deduce the existence unicity of the stationary auto-regressive distribution SAR$(\mu, \mathbf{\Sigma}, \mathbf{\Phi})$, since all marginal distributions are Gaussian.

1.5 KL-divergence between two Gaussian multivariate distributions

Proposition 1.2. *We denote the norm according to a positive-definite symmetric matrix* $\mathbf{\Sigma}$*:*

$$\|x\|_{\mathbf{\Sigma}}^2 = \langle \mathbf{\Sigma}^{-1} x, x \rangle$$

Then:

$$\mathrm{KL}\left(\mathcal{N}_p(\mu_0, \mathbf{\Sigma}_0) \| \mathcal{N}_p(\mu_1, \mathbf{\Sigma}_1)\right) = \frac{1}{2}\left[\|\mu_0 - \mu_1\|_{\mathbf{\Sigma}_1}^2 - \log\left|\mathbf{\Sigma}_1^{-1}\mathbf{\Sigma}_0\right| + \mathrm{Tr}\left(\mathbf{\Sigma}_1^{-1}\mathbf{\Sigma}_0\right) - p\right]$$

2 Contribution: Adding dependency between sites

2.1 Method of moments

To better understand the PLN distribution, and also to understand the impact of the VEM algorithm on the inference of the model, a naive idea is to compute the expectation and covariance matrix.

We can easily solve the system and find the unknown parameters:

$$\boldsymbol{\Sigma} = \log\left(J + \Delta_Y^{-1}\left(\boldsymbol{\Sigma}_Y - \Delta_Y\right)\Delta_Y^{-1}\right)$$

$$\mu = \log \mu_Y - \frac{1}{2}\operatorname{diag}\boldsymbol{\Sigma}$$

then estimate $\boldsymbol{\Sigma}_Y$ and μ_Y by their respective **empirical means**.

2.1.1 Theoretical performance analysis

Two problems arise:

- The data must verify the **over-dispersion** hypothesis empirically to be able to compute the logarithm.

- It seems hard to control the **bias** created by estimating Δ_Y^{-1}. By using a Taylor series expansion, one can prove that is **impossible to estimate the inverse of a Poisson parameter without bias**.

2.1.2 Numerical performance analysis

The moments estimators perform badly in comparison to the VEM estimators, especially as there is no reason that the empirical moments verify the over-dispersion condition.

In order to obtain better results, we could try to remove the zeroes in the counts, then estimate the Poisson parameters, before adding the correct number of zeroes.

However, they could be an alternative for the initialization of the VEM algorithm.

2.2 The new model

$$(Z_1, \dots, Z_n) \sim \mathrm{SAR}(\mu, \mathbf{\Sigma}, \mathbf{\Phi})$$

and

$$Y_{ij} | Z_i \overset{\perp}{\sim} \mathscr{P}(\exp Z_{ij}), 1 \leq j \leq p, 1 \leq i \leq n$$

If $\mathbf{\Phi} = \mathbf{0}$, then we retrieve the classical PLN model.

2.3 Computing the modified ELBO

Let us take the same approach as for non-AR PLN, and choose the set $\mathscr{Q}$ of Gaussian variational distributions. This means that we will try to have correct results with the simplifying hypothesis that the variational variables $Z_i \sim Q_{\psi_i}, 1 \leq i \leq n$ are mutually independent.

If this fails, there are other approaches, like choosing an AR structure for the variational variables.

Let us denote $\theta = (\mu, \mathbf{\Sigma})$ and $\psi = (m_1, \dots, m_n, \mathbf{S}_1, \dots, \mathbf{S}_n)$ such that $Q_\psi \in \mathscr{Q}$.

Also, to be more general we place ourselves in regression, with $\mathbf{B} = (\beta_1 | \dots | \beta_p) \in \mathbb{R}^{d \times p}$ and $\mu_i = \mathbf{B}^\top X_i, 1 \leq i \leq n$ (for the non-conditional version, it suffices to take $\mu_1 = \dots = \mu_n = \mu$). Then:

$$\begin{aligned}
J(\theta, \psi; \mathbf{Y}) &= \mathbb{E}_\psi [\ell_\theta(\mathbf{Y}|\mathbf{Z}) + \ell_\theta(\mathbf{Z}) - \ell_\psi(\mathbf{Z}) | \mathbf{Y}] \\
&= \sum_{i=1}^n \mathbb{E}_{\psi_i}[\ell(\psi; Y_i|Z_i)|Y_i] + \mathbb{E}_{\psi_{i-1}, \psi_i}[\ell(\theta, \psi_{i-1}, \psi_i; Z_{i-1})] - \mathbb{E}_\psi[\ell(\psi; \mathbf{Z})] \\
J(\theta, \psi; \mathbf{Y}) &= \sum_{i=1}^n J_i(\theta, \psi_{i-1}, \psi_i; Y_i)
\end{aligned}$$

where:

$$J_i(\theta, \psi_{i-1}, \psi_i; Y_i) = \sum_{j=1}^p \mathbb{E}_{\psi_i^j}\left[\ell(Y_i^j | Z_i^j) \middle| Y_i^j \right] - \mathbb{E}_{\psi_{i-1}}\left[\mathrm{KL}\left(\mathcal{N}_p(m_i, \mathbf{S}_i) \middle\| \mathcal{N}_p(\mathbf{\Phi}(Z_{i-1} - \mu_{i-1}) + \mu_i, \mathbf{\Sigma}_i) \right) \right]$$

Finally, we compute the ELBO:

Proposition 2.1. *For $1 \leq i \leq n$:*

$$\begin{aligned}
J_i(\theta, \psi_{i-1}, \psi_i; Y_i) = &- \left\| \exp\left(m_i + \tfrac{s_i^2}{2} \right) \right\|_1 + \langle m_i, Y_i \rangle - \| \log(Y_i!) \|_1 \\
&- \frac{1}{2} \| \mu_i - m_i - \mathbf{\Phi}(\mu_{i-1} - m_{i-1}) \|^2_{\mathbf{\Sigma}_i} + \frac{1}{2} \log |\mathbf{\Omega}_i \mathbf{S}_i| - \frac{1}{2} \langle \mathbf{\Omega}_i, \mathbf{S}_i + \mathbf{\Phi} \mathbf{S}_{i-1} \mathbf{\Phi}^\top \rangle + \frac{p}{2}
\end{aligned}$$

where we defined:

- $\mu_0 = m_0 = 0$, $\mathbf{S}_0 = 0$
- $\mathbf{\Sigma}_1 = \mathbf{\Sigma}$, $\mathbf{\Sigma}_i = \mathbf{\Sigma}_\varepsilon = \mathbf{\Sigma} - \mathbf{\Phi \Sigma \Phi}^\top$ *for* $2 \leq i \leq n$

- $\mathbf{\Omega}_i = \mathbf{\Sigma}_i^{-1}$
- $s_i = \operatorname{diag} \mathbf{S}_i$.

Remark. Fortunately, when we force $\mathbf{\Phi} = 0$, we retrieve the non-AR ELBO.

2.4 Implementation

The PLNmodels package is written in R, with two backends, one in torch, the other in C++ using nlopt.

2.4.1 Initialisation

There are multiple possibilities. Either we start with a VE-step and we initialise β, $\mathbf{\Sigma}$ and $\mathbf{\Phi}$, by:

- the method of moments
- their estimation through non-AR VEM
- a non-zero random value (in the case of $\mathbf{\Phi}$)

or we start with an M-step and we initialise $m_1, \dots, m_n, \mathbf{S}_1, \dots, \mathbf{S}_n$ and β using a Poisson regression, which is the current choice for the initialization of the non-AR VEM algorithm.

For $\mathbf{\Phi}$, we could also use AR estimation techniques, after estimating $\mathbf{Z}$ with non-AR VEM or Poisson regression.

2.4.2 M-step

Since there were explicit expressions for the zeroes of the objective function for classical PLN, we should hope the same goes for PLN-AR. But the AR parameter Φ complicates the differentiation. Explicit expressions are useful because they allow us to compute the updates for the C++ backend, and a profiled ELBO, useful to speed up the torch backend.

Let us denote $\widetilde{\mathbf{S}} = (s_1 | \dots | s_n)$ the matrix whose rows are the diagonals of the $\mathbf{S}_1, \dots, \mathbf{S}_n$.

- μ:
$$\nabla_{\mu_i} J(\theta, \psi) = (\mathbf{\Phi} \Omega_{i+1} \mathbf{\Phi}^\top - \Omega_i)(\mu_i - m_i) + \Omega_i \mathbf{\Phi}(\mu_{i-1} - m_{i-1}) + \mathbf{\Phi}^\top \Omega_{i+1}(\mu_{i+1} - m_{i+1})$$

- $\mathbf{\Sigma}$: Let us denote:
$$\mathbf{F}_i = \frac{1}{2} \mathbf{\Omega}_i \left[(\mu_i - m_i - \mathbf{\Phi}(\mu_{i-1} - m_{i-1}))(\mu_i - m_i - \mathbf{\Phi}(\mu_{i-1} - m_{i-1}))^\top + \mathbf{S}_i + \mathbf{\Phi} \mathbf{S}_{i-1} \mathbf{\Phi}^\top - \mathbf{\Sigma}_i \right] \mathbf{\Omega}_i$$

  Then:
$$\nabla_{\mathbf{\Sigma}} J(\theta, \psi) = \mathbf{F}_1 + \sum_{i=2}^{n} \mathbf{F}_i + \mathbf{\Phi}^\top \mathbf{F}_i \mathbf{\Phi}$$

- $\mathbf{\Phi}$:

$$\begin{aligned}\nabla_{\mathbf{\Phi}} J(\theta,\psi) = \mathbf{\Omega}_\varepsilon \sum_{i=2}^{n} \mathbf{\Phi\Sigma} - \mathbf{\Phi S}_{i-1} + (\mathbf{S}_i + \mathbf{\Phi S}_{i-1}\mathbf{\Phi}^\top)\mathbf{\Omega}_\varepsilon \mathbf{\Phi\Sigma} \\ - (\mu_i - m_i - \mathbf{\Phi}(\mu_{i-1} - m_{i-1}))(\mu_{i-1} - m_{i-1})^\top \\ - (\mu_i - m_i - \mathbf{\Phi}(\mu_{i-1} - m_{i-1}))(\mu_i - m_i - \mathbf{\Phi}(\mu_{i-1} - m_{i-1}))^\top \mathbf{\Omega}_\varepsilon \Phi^\top \mathbf{\Sigma}\end{aligned}$$

2.4.3 VE-step

We need to compute the gradients, but their solutions are non-explicit, so we will use the `nlopt` optimisation library in `C++`.

- $\mathbf{M}$:

$$\begin{aligned}\nabla_{\mathbf{M}} J(\theta,\psi) = -\exp(\mathbf{M} + \frac{1}{2}\widetilde{\mathbf{S}}) + \mathbf{Y} - e_1^\top \mathbf{\Omega}(\mu_1 - \mathbf{M}^\top e_1) \\ - \mathbf{\Omega}_\varepsilon(\boldsymbol{\mu} - \mathbf{M} - \mathbf{\Phi}(\boldsymbol{\mu} - \mathbf{M})\mathbf{N}) + \mathbf{\Phi}^\top \mathbf{\Omega}_\varepsilon(\boldsymbol{\mu} - \mathbf{M} - \mathbf{\Phi}(\boldsymbol{\mu} - \mathbf{M})\mathbf{N})\mathbf{N}^\top\end{aligned}$$

where $\mathbf{N} = \begin{pmatrix} 0 & 1 & \dots & 0 \\ \vdots & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & 1 \\ 0 & \dots & \dots & 0 \end{pmatrix}$.

- $\mathbf{S}_i$:

$$2\nabla_{\mathbf{S}_i} J(\theta,\psi) = \mathbf{S}_i^{-1} - \exp(m_i + \frac{1}{2}\mathrm{diag}\,\mathbf{S}_i) - (\mathbf{\Omega}_i + \mathbf{\Phi}^\top \mathbf{\Omega}_{i+1}\mathbf{\Phi})$$

2.5 Study of the conditional distribution $Z|Y$

- In dimension 1, the conditional distribution is almost Gaussian around $\log(Y)$. Using a Taylor development, we obtain expressions of its expectation and its variance with the *Lambert W* implicit functions:
  - $\mu(y) = y - W(e^y) = \log y - \frac{\log y}{y} + o(1)$
  - $\sigma^2(y) = (1 + W(e^y))^{-1}$
- In dimension $p > 1$, it is hard to compute the parameters of this Gaussian approximation.

3 Application: Variation in Recombination Rate and Its Genetic Determinism in Sheep Populations

This dataset on crossovers was suggested and studied in Petit et al. (2017) and Johnston, Huisman, and Pemberton (2018).

3.1 Context

Let us quickly describe the dataset:

- [`chrom`] are the **chromosome indices** of the sheep.
- [`wstart` – `wstop`] are the window spans (unit: megabases) which correpond to the **sites**.
- [`nco_[species]_[genre]`] are the **number of crossovers** for each species.
- [`coverage_[species]_[genre]`] is a measurement of the **offset**.

There are no covariates, so this will be PLN-AR estimation only.

Definition 3.1.

- A crossover is a reciprocal recombination (exchange of genetic information) between two homologous chromosomes during meiosis. It allows alleles to be exchanged between chromosomes, thereby contributing to genetic diversity.
- The recombination rate is the frequency of crossovers in a given window span.

The very low probability of a crossover (around 10^{-8}) justifies the use of a Poisson-based regression model. The parameter β will model the mean recombination rate along the genome and $\mathbf{\Sigma}$ the deviation around this mean.

Why do we need a modified PLN-AR model:

- Empirically observations of a **one-dimensional dependency along the sites** of the genome.
- Φ will model this one-dimensional spatial dependency between one site and the next.

3.2 Exploration

First, let us clean and discover this dataset.

Cleaning

The dataset needs to be pivoted from long to wide format. We also rescale the offset by window size.

Import

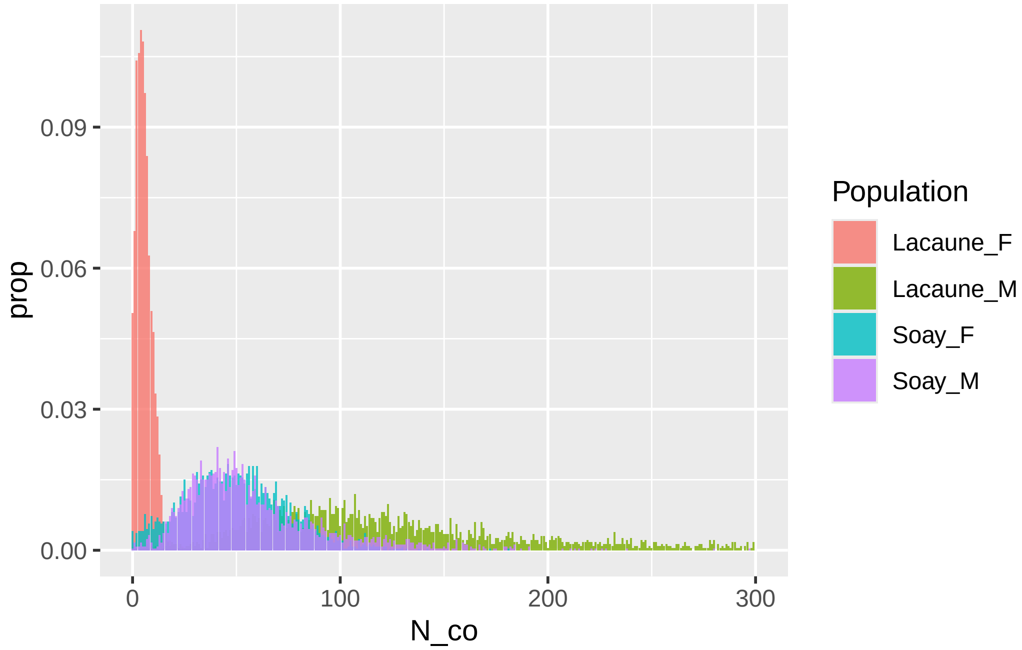
We perform a sanity check, that is to say that we check the number of crossovers in regions with no coverage information. It should hopefully be 0 and we confirm that it is.

```
#> # A tibble: 2 x 8
#>   chrom wstart wstop region_name          population sex     nco coverage
#>   <fct>  <dbl> <dbl> <glue>               <chr>      <chr> <int>    <dbl>
#> 1 23        62    63 23:62000000-63000000 Soay       F         0        0
#> 2 23        62    63 23:62000000-63000000 Soay       M         0        0
```

NAs removal

We remove them to avoid numerical problems during the inference.

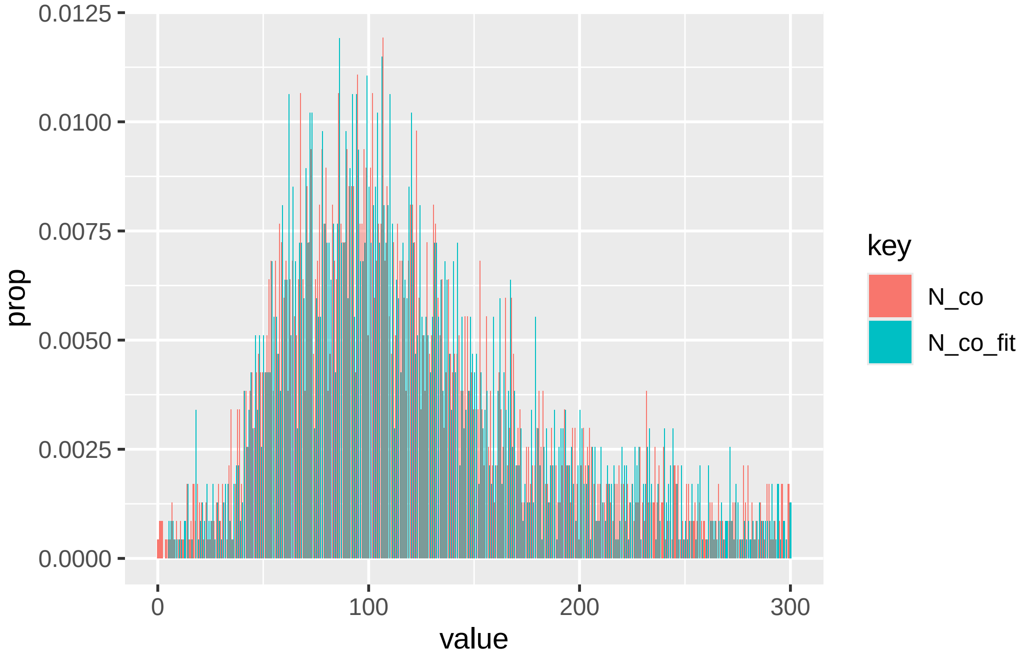
3.2.0.1 One-dimensional plot



3.2.0.2 Classical PLN fitting

First, we carry out a classical non-AR VEM estimation, and we analyze how it fits to the original data.

Comparing the fitted and original data



3.3 Goodness of fit

Once implemented, we shall perform various tests to check the goodness of fit of the predicted and original values, and compare them between the PLN and PLN-AR models:

- Coefficient of determination
- Prediction error and cross-validation (Manhattan distance or mean squared error in centered log-ratio coordinates)
- Q-Q plot
- Shapiro-Wicks

3.4 Analysis

Finally, we check that the auto-regressive structure explains the spatial shape of the crossover data.

References

Aitchison, J. and C. H. Ho (1989). “The multivariate Poisson-log normal distribution”. en. In: *Biometrika* 76.4, pp. 643–653. ISSN: 0006-3444, 1464-3510. DOI: 10.1093/biomet/76.4.643 (cit. on p. 9).

Dempster, A. P., N. M. Laird, and D. B. Rubin (Sept. 1977). “Maximum Likelihood from Incomplete Data Via the EM Algorithm”. In: *Journal of the Royal Statistical Society: Series B (Methodological)* 39.1, pp. 1–22. ISSN: 00359246. DOI: 10.1111/j.2517-6161.1977.tb01600.x (cit. on p. 8).

Dobson, Annette J. and Adrian G. Barnett (2018). *An Introduction to Generalized Linear Models*. Fourth edition. Chapman & Hall/CRC texts in statistical science series. Boca Raton: CRC Press, Taylor & Francis Group. ISBN: 978-1-138-74151-5. URL: https://www.taylorfrancis.com/books/9781315182780 (cit. on p. 7).

Gunawardana, Asela and William Byrne (2005). “Convergence Theorems for Generalized Alternating Minimization Procedures”. In: *Journal of Machine Learning Research* 6.69, pp. 2049–2073. ISSN: 1533-7928 (cit. on p. 14).

Johnston, Susan E, Jisca Huisman, and Josephine M Pemberton (July 2018). “A Genomic Region Containing REC8 and RNF212B Is Associated with Individual Recombination Rate Variation in a Wild Population of Red Deer (Cervus elaphus)”. In: *G3 Genes|Genomes|Genetics* 8.7, pp. 2265–2276. ISSN: 2160-1836. DOI: 10.1534/g3.118.200063 (cit. on p. 20).

Murphy, Kevin P. (2023). *Probabilistic Machine Learning: Advanced Topics*. Adaptive computation and machine learning series. Cambridge, Massachusetts: The MIT Press. ISBN: 978-0-262-04843-9. URL: https://probml.github.io/book2 (cit. on pp. 6–8).

Petit, Morgane et al. (Oct. 2017). “Variation in Recombination Rate and Its Genetic Determinism in Sheep Populations”. In: *Genetics* 207.2, pp. 767–784. ISSN: 1943-2631. DOI: 10.1534/genetics.117.300123 (cit. on p. 20).

Robin, Stéphane (Jan. 2018). *Models with Hidden Structure with Applications in Biology and Genomics*. URL: https://www6.inrae.fr/mia-paris/Equipes/Membres/Anciens/Stephane-Robin (cit. on p. 8).