TRANSITIONS IN BAYESIAN MODEL SELECTION PROBLEMS: NETWORK-BASED RECOMMENDER SYSTEM AND SYMBOLIC REGRESSION

Oscar Fajardo Fontiveros

Department of Chemical Engineering

UNIVERSITAT ROVIRA i VIRGILI

Transitions in Bayesian model selection problems: Network-based recommender system and symbolic regression

Oscar Fajardo Fontiveros

Advisors:
Roger Guimerà Manrique
Marta Sales Pardo

Doctoral Thesis
2021

UNIVERSITAT
ROVIRA i VIRGILI

Transitions in Bayesian model selection problems: Network-based recommender system and symbolic regression

Oscar Fajardo Fontiveros

Doctoral Thesis
Supervised by:
Dr. Roger Guimerà Manrique
Dr. Marta Sales Pardo

September 1, 2021

Summary

Model selection problems consists in looking for the best model given a set of proposed models and the data. This is used by scientists every day when we try to find the explanations of the phenomena that happens around us. In the scientific method there are two steps that are critical, the observation and the hypothesis. It's natural to think that when a scientist tries to solve a model selection problem, he/she have to think in the data that he have collected and if he have a good idea, then take advantage of it to perform the model selection problem. But this world is not perfect, and our data has some error and may be our hypothesis is wrong, so our model that we get is wrong too.

In this thesis we want to study the interplay of the likelihood and the prior in the Bayesian inference process in the case of model selection problem. The Bayes theorem has two important terms: the likelihood and the prior. The likelihood tell us how likely is our data given our model, and the prior is the information that we think a priori that is true. The prior is a probability distribution of models, that we choose given an hypothesis that we have, putting a high prior probability to these models that we think that are the correct one. If our prior is wrong, then we are going to fail in our predictions, and if it's right we are going to make better predictions.

To study this interplay between the likelihood and the prior we are going to solve a couple of problems: the recommender system and the symbolic regression. The recommender system problem consists in from known user preferences we try to predict unobserved ones. Here our data are ratings that user give to items. In this problem we want to analyze how extra information of the users and items (gender, type of item, nationality...) can affect to the inference procedure. Here, the hypothesis that we use was that similar users would rate similar ratings to similar items and vice-versa. So, the prior in this case will contain the information of the metadata. To make this study we

used a generative model, the Mixed-Membership Stochastic Block Model, a Bayesian framework, and synthetic data to control correlation of the ratings with the metadata. We studied all the possible scenarios, where the data can be correlated to the metadata and see how it can affect to the accuracy. In fact, when metadata is full correlated with the data, the best option is to use the metadata. If there is no correlation, the metadata would made the prediction worse. But if there is a high correlation, using both, metadata and data, would get the best performance.

The last problem that we studied was the symbolic regression. This problem consist in to find the best model through the space of mathematical closed-form expressions. This model has to fit the data and also not be very complex. Here we want to study when, given a dataset with noise, we can detect the true model or not. We use the Bayesian machine scientist that uses a Bayesian formulation. This procedure use as a prior the corpus of the Wikipedia and looks for models with similar attributes than known models to avoid choose complex expressions. We used this procedure in synthetic data where we control the noise of our data and we already know the true models, so we can know if we are wrong or not. What we get is that for low noise levels, the algorithm can identify models with similar complexity, but for higher noises levels the algorithm proposes simpler models because fits better with the noise.

Declaration

WE STATE that the present study, entitled "Transitions in Bayesian model selection problems: Network-based recommender system and symbolic regression", presented by Oscar Fajardo Fontiveros for the award of the degree of Doctor, has been carried out under our supervision at the Department of Chemical Engineering of this university.

Tarragona, September 1st, 2021

Doctoral Thesis Supervisor/s

Dr. Roger Guimerà Manrique

Dra. Marta Sales Pardo

Agradecimientos

En este viaje de casi cuatro años, yo he pasado por muchos altibajos, y debido a la situación epidimielógica en la que nos encontramos en la fecha de escitura de esta tesis, pues han habido más bajos al final. Por eso, me parece perfecto agradecer principalmente a Sergio Cobo. Él ha estado detrás mía desde el primer día hasta el último, ¡muchísimas gracias! Una pena que a pesar de que se me acabe este viaje no haya podido enseñarle la diferencia entre una escupiña y un berberecho pero todo no se puede conseguir en esta vida.

Luego me gustaría también agradecer a mis dos directores de tesis Marta Sales y Roger Guimerà, por su estupendísimo trabajo, increíble dedicación, y gran tutoría en este viaje. Siento los inconvenientes que pueda haber causado al final de este trayecto.

Como en todo viaje, también conoces a gente nueva con la que compartir la experiencia, por eso también quería agradecer a Ignasi Reichardt, a Marc Tarrés, a Claudia, a Pun, a Lluc Fonts y Lluís Danús. A los dos últimos ya los conocía de antes, pero los azares del destino me han hecho compartir despacho con ellos, pero una pena que tampoco entiendan la diferencia entre una escupiña y un berberecho. También he conocido a los Salineros Anónimos con los que he podido compartir buenos momentos y hablar de nuestras batallitas.

También quería agradecer a mic compañeros de viaje y amigos de la facultad, Aitor Martín, Javier Cristín, Alejandro Romero y compañía por también estar ahí y ser una buena vía de escape en los momentos más difíciles. También agradecer a los Pastanagues y a los ETDLN (Oscajalluse) por haber sido muy buenos amigos desde la facultad y a otras amistades, como Cristina, Alesia y Meijie. Y hablando de amistades perennes, quiero agradecer también a Lídia, la más durarera de mis amistades y la cual siempre estaré agradecido por lo mucho que ha hecho por mí.

Por último, y no por ello menos importante, quiero dedicarles este trabajo a mis padres y a mi hermano, que son los que han estodo siempre ahí, desde antes de que me fuera de Menorca, hasta ahora.

Contents

Chapter 1

Introduction

1.1 Likelihood and prior in a diagnose problem

In 2006 and 2007 Gerd Gigerenzer proposed the following problem to 1000 gynecologist attending to a course in risk communication [26]:

Assume you conduct breast cancer screening using mammography in a certain region. You know the following information about the women in this region:

- *The probability $P(cancer)$ that a woman has breast cancer is 1%.*
- *If a woman has breast cancer, the probability $P(positive|cancer)$ that she tests positive is 90%.*
- *If a woman does not have breast cancer, the probability $P(positive|no\ cancer)$ that she nevertheless tests positive is 9%.*

A woman tests positive. She wants to know from you whether that means that she has breast cancer for sure, or what the chances are. What is the best answer?

1. *The probability that she has breast cancer is about 81%.*
2. *Out of 10 women with a positive mammogram, about 9 have breast cancer.*

3. *Out of 10 women with a positive mammogram, about 1 has breast cancer.*

4. *The probability that she has breast cancer is about 1%.*

To solve this problem let us note that the pieces of information that we have about the test are probabilities and we want to compute which is the probability that a woman who tests positive has actually breast cancer $P(\text{cancer}|\text{positive})$. The test seems very accurate with a high probability of making the right diagnosis and a low probability of making you the wrong one. However, if the woman from the problem tests positive, that means that she seems unlucky, because a priori the probability that a woman has the disease is low. To compute $P(\text{cancer}|\text{positive})$ we have to use the Bayes Theorem:

$$P(H|E) = \frac{P(E|H)P(H)}{P(E)} \tag{1.1}$$

where H is our hypothesis in our case is that she has breast cancer, E is the evidence in our case, the results of the test. Using Bayesian nomenclature, $P(E|H)$ is called the likelihood and tells us how probable our observation is given the hypothesis. $P(H)$ is the prior, that is the probability that our hypothesis is true a priori, that is without any observations. $P(H|E)$ is the posterior, that is the plausibility that our hypothesis is the true given the observed evidence. The prior needs to be established with general information about the problem, in our case, the prevalence. The likelihood is established through the observations that we made. If we rewrite Eq. 1.1 in terms of our data from the problem and introduce the numerical values we get:

$$\begin{aligned} P(\text{cancer}|\text{positive}) &= \frac{P(\text{positive}|\text{cancer})P(\text{cancer})}{P(\text{positive})} \\ &= \frac{P(\text{positive}|\text{cancer})P(\text{cancer})}{P(\text{positive}|\text{cancer})P(\text{cancer}) + P(\text{positive}|\text{no cancer})P(\text{no cancer})} \\ &= 10\% \end{aligned}$$

So that the correct answer is option number 3. That means that 9 women over 10 who test positive, are wrongly diagnosed. As a curiosity, only 21% of the gynecologists gave the correct answer to the question, and 60% of them would have diagnosed the patient with breast cancer.

This example illustrates that accurate does not imply predictive and we can ask: Why is this happening?. To answer this question let's check the numbers. As we said before, the test is "accurate" in that it has a 90% probability to give a positive result if somebody has cancer. However, the problem is that breast cancer is not common. Due to the low prevalence of the disease, it makes it less plausible to have cancer even with a positive test result.

Now, let's focus on this question from a Bayesian point of view. Before our patient has been tested, we know that she had a 1% probability of having cancer. After she got the results of the test, we updated our posterior to a 10%, in other words, the probability of having cancer increased by one order of magnitude. And that is how the likelihood works, when we have new observed information, our beliefs of the hypothesis are updated.

Now imagine that a priori we do not know anything, so we do not have the information about the prevalence, or any genetic information about our patient. In this case the prior is equally distributed so $P(\text{ cancer}) = P(\text{no cancer}) = 50\%$. In this case our posterior after the test is done would be 91%. In this case the posterior is almost equal to the likelihood; in fact, it only depends on the likelihood distribution and because the test is accurate, a positives results leads to posterior that is also high.

As a conclusion of the problem we considered, we have introduced the Bayes theorem, a theorem that helps us to verify hypothesis given our prior information of the problem (represented in the prior) and the observations (likelihood). If the prior is low, eventually the posterior of our hypothesis will be low, and the same will happen if the likelihood is also low. Now, we are in a position to make a couple of reflections. With a more accurate test, we could increase the posterior; but with a worse test the posterior would decrease. The same would happen with the prior, if we had some extra information like genetic information, our prior would also change the posterior. If our patient has a gene that makes her more probable to have cancer, the posterior would increase; otherwise the posterior would decrease.

1.2 Bayes theorem applied to model selection

Up to here we have introduced the basic components for Bayesian Inference. Now let us consider a more general case. Imagine that we observe some data D. This dataset can be anything: a set of points $\{(x_i, y_i)|i \in \mathbb{N}\}$, the ratings that people give to a set of products or movies and so on. We want to find from all the possible models $M \in \mathcal{M}$ that can generate D, the model M^* that best explains D. From a Bayesian point of view, this model M^* should be the most probable one given the data D, in other words, the model that maximizes the posterior $P(M|D)$, with:

$$P(M|D) = \frac{P(D|M)P(M)}{P(D)} \tag{1.2}$$

Note that the likelihood has the information of how the model M fits the data D, in other words, how likely it is that my data have been generated by model M. The prior encapsulates my a priori explanation that model M is the correct model. Having no additional information about the model $P(M)$ = constant translates into an uniform distribution, so that, all the models are equiprobable. If our information is wrong, the prior would assign a low probability to the real model, but if it is right, it will assign a high probability.

In most cases, the models M that we want to verify depend on parameters θ. For example the intercept and the slope of a linear model with form $y = mx + n$ where $\theta = \{m, n\}$. In this case we have to marginalize the posterior integrating $P(M, \theta|D)$ to get $P(M|D)$, so:

$$P(M|D) = \int_\Theta P(M, \theta|D) d\theta = \frac{1}{P(D)} \int_\Theta P(D|M, \theta)P(M|\theta)P(M) d\theta \tag{1.3}$$

In both expressions, 1.2 and 1.3, the estimator to get the most plausible model is the posterior. There are several ways to find it, using Markov-Chain Montecarlo (MCMC), or using the Maximum a Posterior (MAP) algorithm (Ref. [38]).

Let's make an analysis from the information-theoretic point of view of Eq. 1.3. To do that, we rewrite Eq. 1.3 as:

$$P(M|D) = \frac{1}{P(D)} \int_\Theta P(D|M, \theta)P(M|\theta)P(M) d\theta = \frac{1}{P(D)} e^{-\mathcal{D}(M)} \tag{1.4}$$

Here $\mathcal{D}(M)$ is the description length:

$$\mathcal{D}(M) \equiv -\log P(D, M) = -\log \int_\Theta P(D|M, \theta)P(M|\theta)P(M) d\theta \tag{1.5}$$

This is the length in nats (natural digits) that we need to jointly describe our model and our data. As we said before, the most plausible model is that one that maximizes the posterior $P(M|D)$. From Eq. 1.4 it follows that the model that maximizes $P(M|D)$, must minimize the description length. In the literature this is the so called *minimum description length principle* [27, 58].

If we take $P(M)$ outside the integral in Eq. 1.5, we can split the description length in two terms:

$$\mathcal{D}(M) = \mathcal{D}_L(D|M) + \mathcal{D}_P(M), \tag{1.6}$$

where $\mathcal{D}_L(D)$ is the contribution of the description length of the likelihood of the model given the data that is

$$\mathcal{D}_L(D|M) \equiv -\log \int_\Theta P(D|M,\theta)P(M|\theta)d\theta \tag{1.7}$$

and $\mathcal{D}_P(M)$ is the log-prior that is:

$$\mathcal{D}_P(M) \equiv -\log P(M) \tag{1.8}$$

Observing Eq. 1.6 we can see that the most plausible model will be that one that minimizes the sum of $\mathcal{D}_L(D|M)$ and $\mathcal{D}_P(M)$. Here we can recover the discussion from the last section but applied to the model selection problem. Observe that, as before, the posterior depends on the likelihood and the prior, and the plausibility of each model will be a combination of both.

1.3 Likelihood vs. prior

The likelihood and the prior, tell us about different aspects of the model, one tells us how well the model fits the data and the other is the information about the a priori plausibility of a model. To illustrate the consequence of having these two terms, we are going to make an analogy with a physical model, the Ising model, and the ferromagnetic to paramagnetic transition.

The Ising model [8, 9, 11] is a model of a ferromagnet. Suppose that we have N magnetic particles (spins), distributed in a square lattice in a thermal bath. These spins can point up or down and they interact with their nearest neighbours with a negative energy coupling. The orientation of the spins depends on the orientation of their nearest neighbours and the thermal fluctuations. We can characterize the overall state of the system measuring the magnetization, that is the sum of the spins in the system. If all the spins point in the same direction, the magnetization in absolute value is not zero (ferromagnetic state), but if are not aligned the magnetization is zero (paramagnetic state). Because the system is in a thermal bath, the system is in a state (orientation of spins) that minimizes the Helmholtz free energy F defined as:

$$F = U - TS \tag{1.9}$$

Where U is the internal energy of the system, T the temperature of the thermal bath and S the entropy of the system. U depends of the configuration of the system: if two spins point in the same direction, U will decrease but if they point in different directions the energy will increase. The term TS is the entropic term of F and tells us how strong are the fluctuations in the spin configuration due to the thermal bath.

If the temperature is close to zero, $F \approx U$ and, the energetic term dominates over the entropic term. That means that the system will reach the equilibrium when all the spins point in the same direction, being our system a ferromagnet. As we increase the temperature, the fluctuations increase as TS starts to contribute to minimize F. Because of the fluctuations, some spins start to point in a different direction, in other words, we start to have a disordered system. This disorder will increase as we increase the temperature. That means also that the magnetization of the system will decrease. At a certain temperature, all the spins will be fluctuating from up to down, loosing all the magnetization. In this case the entropy dominates over the energy, the system is in a disordered state that we called paramagnetic. Therefore this system has two different behaviours depending on which term of the Helmholtz free energy prevails: the energy or the entropy.

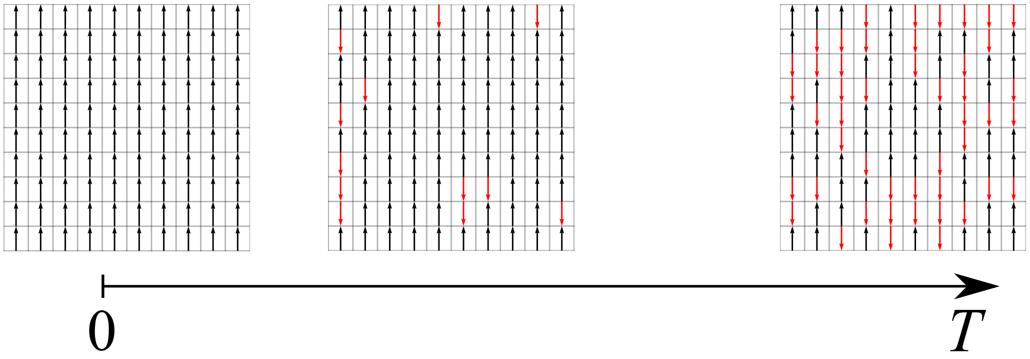

Figure 1.1: Representation of a system of 100 spins at different temperatures Three examples of a system of 100 spins at different temperature. On the left we have a system at $T = 0$ where the internal energy dominates and all spins points at the same direction. In the middle we have a system with $T \neq 0$ (below the transition), and as a consequence some spins point in the opposite direction, the entropic term starts to have a presence. On the right we have a system at high temperature and its spin point randomly to one direction or the other, the entropic term dominates completely over the Helmholtz free energy.

This example can be used to make an analogy with our model selection problem. In the model selection problem we need to minimize the description length $\mathcal{D}(M, D)$ (analogous to F) that is the sum of two terms of different nature, the first one is the data and the second one is the prior knowledge about our problem.

If we do not have any data, $\mathcal{D} = \mathcal{D}_P$, in other words, my predicted model will be one that minimizes the description length of the model. In fact that model is the most plausible according to prior distribution. As we add some data, the contribution of the likelihood to the description length is also increased and starts to play a role. For $N \to \infty$, the prior contribution to the description length is negligible respect the prior one, so $\mathcal{D} \approx \mathcal{D}_L$ and our selected models will be data driven.

Imagine that suddenly we have data with a lot of noise and we start to lower the noise. In this scenario, because the models do not fit well the data, the selected models will be also prior driven. As the noise decreases the log-likelihood will start to play a role in the model selection problem and our selected models will be both, data driven and prior driven models. Note that we are not talking about how good is our selected model. In the model driven regime, our prior distribution can assign to the real model a low probability, being our prior a bad prior.

As in the Ising model, here we have different situations where depending in how good our data are, our models have different nature:data driven, prior driven and a mixture of both.

1.4 Objectives

We have just seen how Bayes theorem works in model selection. Also, we have explained an interplay between the prior and the likelihood that, depending in the data and how good is our prior, we can get different types of models based on the prior distribution or the data. To study this interplay between prior and likelihood we propose two problems that we solve using Bayesian inference. The first problem is the recommender system problem where we have metadata of the linked items. We propose a mixed membership stochastic block model to make the predictions but, we put as a prior the metadata. To play with the prior quality we generate different datasets with different metadata correlation, letting us to study well how the transition of how metadata becomes more important as we increase the correlation. The second problem that we studied is the symbolic regression problem using the Machine Scientist method. Here we start from five formulas and then we generate different datasets with different noise levels and use the Machine Scientist to try to recover the formulas. Unlike the first problem where we change the goodness of the prior, here we change the goodness of the data to see the transition of the performance of the algorithm.

Chapter 2

Transitions in the accuracy of recommender systems when we consider node metadata

2.1 Introduction

In this chapter we are going to study our first case study, a recommender system. The recommender system is a common problem in computer science where from known user preferences we try to predict unobserved ones (Ref. [71]). Here, our observations are ratings $r_{i,j} \in R^O$ that an user i gives to an item j. Ratings can be a number, a binary value (for example like or dislike) and items can be movies, songs, books, etc (Ref. [72]). Besides our observed ratings, we have more information about the users and items: the user's gender, the age, the type of the item... This extra information can (or not) be related with our observed data, so it can be usefull (or not) to make predictions. According to the vice president of product of Netflix, Todd Yelling said that most of the data that they have about their users is ”garbage” when you want to predict new preferences [73]. In this chapter we are going to discuss how this extra information, in which from now now we call metadata, can help us make predictions by introducing this into the inference process. From a Bayesian point of view, we are going to consider the known preferences as the likelihood, and the metadata as the prior of our problem and see the interplay between these two terms when we want to make new predictions.

Recommender system techniques can be classified in three different groups: content-based filtering, collaborative filtering and hybrid filtering (Ref. [17]). Content-based filtering approaches (Ref. [13, 40, 59, 72]) consist in recommending similar items to those we know that our target user prefers. This approach uses the user's history but does not take into account other user's preferences. For example, if we know that our user likes action movies, it is very reasonable to recommend him/her action movies. This approach assumes that we can represent an item using a vector $X = (x_1, x_2, ..., x_n)$ where x_i is a feature that can be represented by a number, a string or a binary value. This can term obtained from either item's keyword, metadata or text description. This term can be for example a term frequency-inverse document frequency (TF-idf) (Ref. [5, 43]), where each term is weighted by the importance of it in the considered item with respect to the rest of items. Then, you can assign a profile vector to each user with a vector $X(u) = (x_1, x_2, ..., x_n)$ constructed aggregating the terms of items who the user liked or bought. Once we profile our user, we can use a similarity measure (cosine similarity for example) to find the items that are most similar to our user's profile.

$$\mathrm{sim}(X(u), X(j)) = \frac{X(u) \cdot X(j)}{||X(u)||\,||X(j)||} \tag{2.1}$$

Alternatively to content-based some can use collaborative filtering (CF) approaches (Ref. [21, 23, 30]) These methods consist in finding similarities between users and items to make predictions instead of only focusing in the history of an individual like the content-based filtering. These techniques exploit similarities between preferences of users to make recommendations. There are two types of CF methods: memory-based and model based.

Memory-based CF (also known as neighbour-based CF) techniques (Ref. [15, 23, 24]) use similarity measures calculated from explicit user-item ratings to find neighbours of users (user-user approach) or items (item-item approach), and then generate predictions from the similarities. One example of similarity measure is the cosine similarity. Let's consider that for item i we have a vector $\vec{i} \in R^N$ where N is the number of users so that $i_n = 1$ if user n has rated i and $i_n = 0$ otherwise. The similarity between items i and j is:

$$\mathrm{sim}(i, j) = \frac{\vec{i} \cdot \vec{j}}{\left|\left|\vec{i}\right|\right|\,\left|\left|\vec{j}\right|\right|} \tag{2.2}$$

So, the predicted link $r_{u,i}$ is computed doing the average over the neighbours of item i, ∂i:

$$r_{u,i} = \frac{\sum_{j \in \partial i} (\mathrm{sim}(i,j) r_{uj})}{\sum_{j \in \partial i} (|\mathrm{sim}(i,j)|)} \tag{2.3}$$

Model-based CF (also known as latent factor models) (Ref. [23, 29, 32, 41]) are techniques that try to model user's ratings assuming that a latent factor space exists for both items and users. One of the most famous methods is Matrix Factorization (MF), that consists in factorizing the ranking matrix R^O as the product of two matrices:

$$R^O = PQ \tag{2.4}$$

Where P is a $N \times K$ matrix where N is the number of users, K is the dimension of the latent space, and Q is a $M \times K$ matrix. We can interpret the vector that each row of P is a vector p_i that is a feature vector of user i, similar the content-based techniques' vectors. As we show later in 2.4, MF assumes that the rating of user u gives to item i is proportional to the closeness between the two in this space.

2.2 Mixed-Membership Stochastic Block Model for recommendation using metadata as priors

2.2.1 Mixed-membership Stochastic Block Models

Another model-based collaborative filtering approach, that originates from the problem of link prediction in complex networks (Ref. [59]), Mixed-Membership Stochastic Block Model (MMSBM) [47, 56, 62, 69]. This approach is amenable to Bayesian inference and has been shown to perform better than MF in recommendation tasks (Ref. [47]). To use this method as a recommender system we map the recommender system problem to a link prediction one. To do that we have to reconsider our data as a bipartite network (Ref. [16, 45]), a network with two types of nodes: users connected to items. The links between users and items are labeled with the ratings (fig. 2.1).

The MMSBM assumes that:

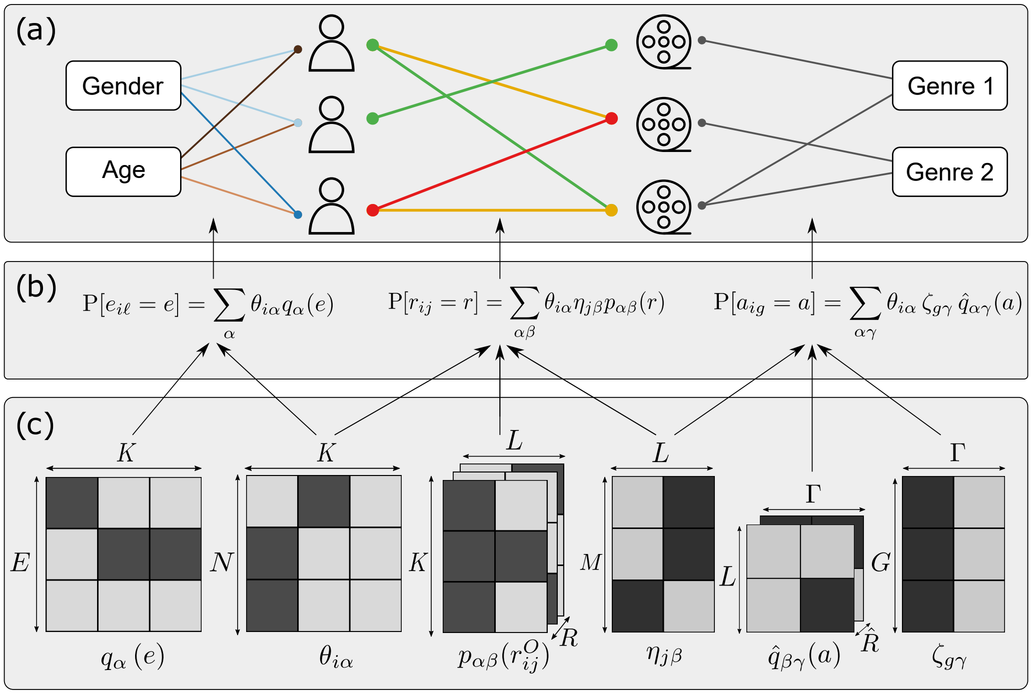
(a)
Gender
Age
Genre 1
Genre 2
(b)
$\mathrm{P}[e_{i\ell} = e] = \sum_{\alpha} \theta_{i\alpha} q_{\alpha}(e)$
$\mathrm{P}[r_{ij} = r] = \sum_{\alpha\beta} \theta_{i\alpha} \eta_{j\beta} p_{\alpha\beta}(r)$
$\mathrm{P}[a_{ig} = a] = \sum_{\alpha\gamma} \theta_{i\alpha}\, \zeta_{g\gamma}\, \hat{q}_{\alpha\gamma}(a)$
(c)
K
E
$q_{\alpha}\,(e)$
K
N
$\theta_{i\alpha}$
L
K
R
$p_{\alpha\beta}(r^{O}_{ij})$
L
M
$\eta_{j\beta}$
Γ
L
$\hat{R}$
$\hat{q}_{\beta\gamma}(a)$
Γ
G
$\zeta_{g\gamma}$

Figure 2.1: Multipartite mixed-membership stochastic block model with labeled links. (a), We cast the recommendation problem (in which one aims to predict how users will rate certain items) into a network inference problem. Here, users rate movies with three possible ratings (green, orange or red). Additionally, we have excluding attributes for users (two excluding genders and three excluding age groups, represented by different shades of the same color) and non-excluding attributes for movies (two movie genres; the connection to these attributes is binary, yes/no, but in general it does not need to be). Similar to ratings, we represent these attributes as bipartite networks. Although we frame our description of the model in terms of recommendations or link prediction in a bipartite network, the problem of link prediction in regular unipartite networks is just a particular case in which user nodes and item nodes are the same. (b) Each bipartite network in the multipartite network is modeled using a mixed-membership stochastic block model (see text). The individual block models are coupled by the user and item membership vectors ($\boldsymbol{\theta}$ and $\boldsymbol{\eta}$, respectively), shown in (c) along with all other model parameters and their dimensions (see text).

1. Nodes are distributed into groups. In other words, users are organized in groups that can be interpreted as a set of users that rate similarly

the same items. Items are organized in groups that are rated similarly by the same users.

2. Each node (user or item) can belong to each groups (of users or items) at the same time with a a finite probability. The vector of probabilities is called membership vector.

3. The probability that user u gives rating r_{ui} to item i depends only on the group membership of u and i.

Using the above assumptions we can model the bipartite network of users who rate movies and predict new preferences as the labels of the links.

Our system has N users and M items and our observed ratings R^O are labels $r_{i,j} \in [0, R]$. We define $\boldsymbol{\theta}_i$ as the normalized membership vector of K groups of user i, and each element $\theta_{i\alpha}$ represents the probability that user i belongs to group α (with $\sum_\alpha \theta_{i\alpha} = 1$). Similarly, $\boldsymbol{\eta}_j$ is the normalized membership vector of item j; $\eta_{j\beta}$ represents the probability that item j belongs to group β. Finally, $p_{\alpha\beta}(r)$ is the probability that a user in group α and an item in group β are connected with a rating r. The normalization condition here is $\sum_r p_{\alpha\beta}(r) = 1$. Finally, the probability that a user i rates an item j with a rating r_{ij} is:

$$P[r_{ij} = r] = \sum_{\alpha\beta} \theta_{i\alpha}\eta_{j\beta}p_{\alpha\beta}(r) \ . \tag{2.5}$$

2.2.2 Multipartite Mixed-Membership Stochastic Block Models to incorporate node metadata

The parameters $\boldsymbol{\theta}$, $\boldsymbol{\eta}$ and $p_{\alpha\beta}(r)$ that we are looking for are that ones that maximize the probability $P(\boldsymbol{\theta}, \boldsymbol{\eta}, \boldsymbol{p}|R^O)$. Applying the Bayes theorem we get:

$$\begin{aligned} P(\boldsymbol{\theta}, \boldsymbol{\eta}, \boldsymbol{p}|R^O) &\propto P(R^O|\boldsymbol{\theta}, \boldsymbol{\eta}, \boldsymbol{p})\, P(\boldsymbol{\theta}, \boldsymbol{\eta}, \boldsymbol{p}) \\ &\equiv L^R(\boldsymbol{\theta}, \boldsymbol{\eta}, \boldsymbol{p})\, P(\boldsymbol{\theta}, \boldsymbol{\eta}, \boldsymbol{p}) \ , \end{aligned} \tag{2.6}$$

where $L^R(\boldsymbol{\theta}, \boldsymbol{\eta}, \boldsymbol{p}) = P(R^O|\boldsymbol{\theta}, \boldsymbol{\eta}, \boldsymbol{p})$ is the likelihood of the model and $P(\boldsymbol{\theta}, \boldsymbol{\eta}, \boldsymbol{p})$ is the prior over model parameters. According to Eq. (2.5), the likelihood is

$$L^R(\boldsymbol{\theta},\boldsymbol{\eta},\boldsymbol{p}) = \prod_{(i,j)\in R^O} \left[\sum_{\alpha\beta} \theta_{i\alpha}\eta_{j\beta}p_{\alpha\beta}(r^O_{ij}) \right] \ . \tag{2.7}$$

The prior can be modeled in different ways, in case of a uniform prior MMSBM works fine when we perform predictions in recommender systems. Here we want to consider the use of metadata as a prior and then study how this metadata affects the inference process. The use of metadata as a prior was previously studied [22, 42, 48, 49, 50, 53, 56, 61, 63], but here we want to consider a new model that is to add metadata. First of all, let's consider the relationship between users (or items) and their metadata as another bipartite network. Merging the users/items-metadata bipartite networks to the users-item bipartite network, we get a multipartite structure (Fig. 2.1 a). With this model, we can consider that each bipartite network can be modeled as a MMSBM. If our metadata is non-excluding, meaning that each user (or item) can have assigned different attributes elements of this metadata (movie genre for example where a movie has more than one genre), the probability that any node i is linked with attribute element g with label a is:

$$P[a_{ig} = a] = \sum_{\alpha\gamma} \theta_{i\alpha}\, \zeta_{g\gamma}\, \hat{q}_{\alpha\gamma}(a) \tag{2.8}$$

where $\zeta_{g\gamma}$ is the membership vector of attribute g and $\hat{q}_{\alpha\gamma}(a)$ is the probability that a user in group α has an attribute of type a for an attribute in attribute group γ.

In case that the metadata is an excluding element, that is that each node can be linked to only one element (user's age for example, a person can only has one age), the probability that user i has an excluding attribute e (that is, the probability that the link $e_{i\ell}$ between user i and attribute node ℓ is of type e) is

$$P[e_{i\ell} = e] = \sum_{\alpha} \theta_{i\alpha} q_\alpha(e) \ , \tag{2.9}$$

where $q_\alpha(e)$ is the probability that a user of group α has an attribute of type e, and $\sum_e q_\alpha(e) = 1$.Note that equations. 2.8 and 2.9 can be applied also to items' metadata changing $\theta_{i\alpha}$ by $\eta_{j\beta}$.

This extension of the MMSBM is inspired by the works of Hric, Peixoto and Fortunato (Ref. [48]) and Newman and Clauset (Ref. [49]). The advantages of our approximation are that it includes also non-exclusive metadata (unlike Ref. [49]) and it can consider excluding and non-excluding metadata together. Also our model can introduce as many attributes as you want, giving to the model more flexibility. Our model also allows no having all attributes from nodes.

From here, we are going to assume that the most plausible membership parameters are that ones that fit better not only the rating bipartite network, but also the metadata bipartite networks. That means that when we are inferring ratings, we have to consider the overall multipartite network. In other words, we use the "likelihoods" of the metadata bipartite networks as our prior for the group memberships. For excluding metadata, its contribution to the prior Eq. 2.6 will be the likelihood of the bipartite network of the k-th non-excluding metadata $L^{A_k}(\boldsymbol{\theta}, \boldsymbol{\eta}, \boldsymbol{q})$, that is:

$$L^{A_k}(\boldsymbol{\theta}, \boldsymbol{\eta}, \boldsymbol{q}) = \prod_{(i,\ell_k)\in A_k^O} \left[\sum_{\alpha} \theta_{i\alpha} q_{\alpha}^{k}((e_k^O)_{i\ell_k}) \right] , \tag{2.10}$$

where ℓ_k is the k-th non-excluding attribute and the product is over all nodes i for which we observe attribute ℓ_k.

For the k-th non excluding attribute we have

$$L^{A_k}(\boldsymbol{\theta}, \boldsymbol{\eta}, \boldsymbol{\zeta}, \hat{\boldsymbol{q}}) = \prod_{(i,g)\in A_k^O} \left[\sum_{\alpha\gamma} \theta_{i\alpha} \zeta_{g\gamma}^{k} \hat{q}_{\alpha\gamma}^{k}((a_k^O)_{ig}) \right] . \tag{2.11}$$

where the product is over all observed associations between nodes i and attributes g within the k-th class of non-excluding attributes.

Applying Bayes theorem to Eq. 2.6 we get:

$$\begin{aligned} P(\boldsymbol{\theta}, \boldsymbol{\eta}, \boldsymbol{\zeta}, \boldsymbol{p}, \boldsymbol{q}, \hat{\boldsymbol{q}} | R^O, A^O) &\propto L^R(\boldsymbol{\theta}, \boldsymbol{\eta}, \boldsymbol{p}) \times \\ &\times \prod_k L^{A_k}(\boldsymbol{\theta}, \boldsymbol{\eta}, \boldsymbol{\zeta}, \boldsymbol{q}, \hat{\boldsymbol{q}}) \times \\ &\times P(\boldsymbol{\theta}, \boldsymbol{\eta}, \boldsymbol{\zeta}, \boldsymbol{p}, \boldsymbol{q}, \hat{\boldsymbol{q}}) \end{aligned} \tag{2.12}$$

Where we take into account both types of metadata. If we take the logarithms of Eq. 2.12 to better decompose the influence of our observed ratings (represented by the likelihood) and the metadata (represented by the prior):

$$\log P(\boldsymbol{\theta}, \boldsymbol{\eta}, \boldsymbol{\zeta}, \boldsymbol{p}, \boldsymbol{q}, \hat{\boldsymbol{q}}|R^O, A^O) = \mathcal{L}^R(\boldsymbol{\theta}, \boldsymbol{\eta}, \boldsymbol{p}) + \sum_k \mathcal{L}^{A_k}(\boldsymbol{\theta}, \boldsymbol{\eta}, \boldsymbol{\zeta}, \boldsymbol{q}, \hat{\boldsymbol{q}}) \ + C; , \tag{2.13}$$

where $\mathcal{L}^R(\boldsymbol{\theta}, \boldsymbol{\eta}, \boldsymbol{p})$, $\mathcal{L}^{A_k}(\boldsymbol{\theta}, \boldsymbol{\eta}, \boldsymbol{\zeta}, \boldsymbol{q}, \hat{\boldsymbol{q}})$ are the log-likelihoods of ratings and metadata, respectively and $C \equiv -\log P(R^O, A^O)$, that is constant because does not depend on the parameters. Here we can observe better that when we try to find the parameters that maximize $P(\boldsymbol{\theta}, \boldsymbol{\eta}, \boldsymbol{\zeta}, \boldsymbol{p}, \boldsymbol{q}, \hat{\boldsymbol{q}}|R^O, A^O)$ we have to take into account the whole multipartite network with the same weight, in other words, we are assuming that they have the same importance when we want to perform predictions. However, we can find situations in which metadata is not helpful. We can have ratings generated by mechanisms that doesn't have any relationship with metadata, like homophilia, where we assume that all users with the same gender have the same interests. To control of this we add an hyperparameter, similar to Refs. [42, 63], for each metadata that multiplies its log-likelihood:

$$\pi(\boldsymbol{\theta}, \boldsymbol{\eta}, \boldsymbol{\zeta}, \boldsymbol{p}, \boldsymbol{q}, \hat{\boldsymbol{q}}|R^O, A^O) = \mathcal{L}^R(\boldsymbol{\theta}, \boldsymbol{\eta}, \boldsymbol{p}) + \sum_k \lambda_k \mathcal{L}^{A_k}(\boldsymbol{\theta}, \boldsymbol{\eta}, \boldsymbol{\zeta}, \boldsymbol{q}, \hat{\boldsymbol{q}}) \ , \tag{2.14}$$

where λ_k is the hyperparameter that we add, $\pi(\boldsymbol{\theta}, \boldsymbol{\eta}, \boldsymbol{\zeta}, \boldsymbol{p}, \boldsymbol{q}, \hat{\boldsymbol{q}}|R^O, A^O)$ is the hyperparametrized log-posterior. We define λ_k as a non negative hyperparamter. λ_k contains the importance of the metadata in the inference process (Fig 2.2). If $\lambda_k = 0$ we have that the posterior is equal to the likelihood and we are not going to take into account the metadata when we make predictions. For low values of $\lambda_k = 0$, we start to see the metadata and it will have the a few impact when we try to predict. For $\lambda_k = 1$, we observe the whole multipartite network and we recover Eq. 2.13. For $\lambda_k \to \infty$, the metadata visibility is bigger than our observed ratings, and metadata overshadows the observed ratings when we want to make predictions. That means that the groups that we are going to find will take only into account the metadata, that is assuming that nodes with similar attributes have similar group memberships and therefore similar ratings (see Fig. 2.2). The study of the predictive power of our model in these different scenarios will be the main focus of this chapter.

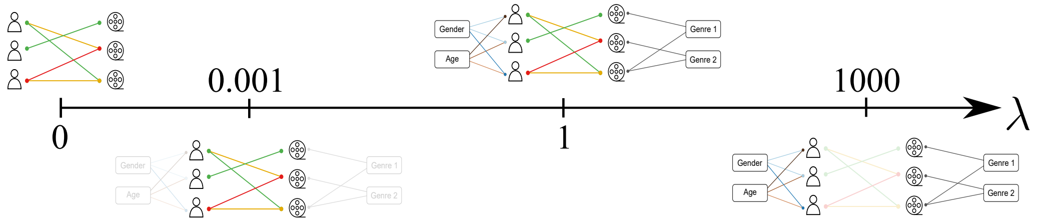

Figure 2.2: Schematic representation of how the hyperparameter λ controls the contribution of metadata to the posterior. In this figure we suppose, for simplicity of representation, that users and movies metadata have the same hyperparameter $\lambda_{\text{user}} = \lambda_{\text{item}} = \lambda$. As we said before, for $\lambda = 0$ we can only see the contribution of the observed ratings. As we increase the value of λ we start to take into account the influence of metadata in the inference process. When $\lambda = 1$, metadata and observed ratings has the same importance. When λ is too large the effect of metadata is too strong that basically we are only taking into account metadata to perform predictions.

Each sum of Eqs. 2.13 and 2.14 will contribute different only depending of the size of the observed ratings (for the likelihood) and the size of the observed metadata (for the prior). If for example we have infinite number of observed ratings, $\mathcal{L}^R(\boldsymbol{\theta}, \boldsymbol{\eta}, \boldsymbol{p})$ and $\mathcal{L}^{A_k}(\boldsymbol{\theta}, \boldsymbol{\eta}, \boldsymbol{\zeta}, \boldsymbol{q}, \hat{\boldsymbol{q}})$ will be infinite and then the metadata contribution will be useless because with only our observed ratings we can make predictions perfectly. But that is something that sometimes we can not control, we can not guarantee to collect a huge amount of ratings and we can not neither have all the metadata of a certain attribute.

2.3 Expectation maximization equations

We aim to maximize the parametric log-posterior in Eq. (2.13) as a function of the model parameters $\boldsymbol{\theta}, \boldsymbol{\eta}, \boldsymbol{p}, \boldsymbol{\zeta}, \boldsymbol{q}$ and $\hat{\boldsymbol{q}}$. Because logarithms of sums are hard to deal with, we use a variational trick that first introduces an auxiliary distribution $p(x)$ with $\sum_x p(x) = 1$ into a sum of terms as $\sum_x x = \sum_x p(x)\,(x/p(x))$. Then because $\sum_x p(x)\,(x/p(x)) = \langle x/p(x)\rangle$ we can use Jensens' inequality (Ref. [1, 10, 39]) $\log\langle y\rangle \geq \langle \log y\rangle$ to write $\log\left[\sum_x p(x)\,(x/p(x))\right] \geq \sum_x p(x) \log\left[x/p(x)\right]$.

Because both rating and attribute terms in Eq. (2.13) contain logarithms of sums, we introduce an auxiliary distribution for each of the terms as follows. For the ratings, we have

$$\begin{aligned}
\mathcal{L}^R &= \sum_{(i,j)\in R^O} \log \sum_{\alpha\beta} \theta_{i\alpha}\eta_{j\beta}p_{\alpha\beta}(r^O_{ij}) \\
&= \sum_{(i,j)\in R^O} \log \sum_{\alpha\beta} \omega_{ij}(\alpha,\beta) \frac{\theta_{i\alpha}\eta_{j\beta}p_{\alpha\beta}(r^O_{ij})}{\omega_{ij}(\alpha,\beta)} \\
&\geq \sum_{(i,j)\in R^O} \sum_{\alpha\beta} \omega_{ij}(\alpha,\beta) \log \frac{\theta_{i\alpha}\eta_{j\beta}p_{\alpha\beta}(r^O_{ij})}{\omega_{ij}(\alpha,\beta)} \qquad (2.15)
\end{aligned}$$

where $\omega_{ij}(\alpha,\beta)$ is the auxiliary distribution.

For the term corresponding to excluding node attributes we have

$$\begin{aligned}
\mathcal{L}^{A_k} &= \sum_{(i,\ell_k)\in A^O_k} \log \sum_{\alpha} \theta_{i\alpha} q^k_\alpha(i\ell_k) \\
&= \sum_{(i,\ell_k)\in A^O_k} \log \sum_{\alpha} \sigma^k_{i\ell_k}(\alpha) \frac{\theta_{i\alpha} q^k_\alpha(i\ell_k)}{\sigma_{i\ell_k}(\alpha)} \\
&\geq \sum_{(i,\ell_k)\in A^O_k} \sum_{\alpha} \sigma^k_{i\ell_k}(\alpha) \log \frac{\theta_{i\alpha} q^k_\alpha(i\ell_k)}{\sigma^k_{i\ell_k}(\alpha)} \qquad (2.16)
\end{aligned}$$

where $\sigma^k_{i\ell_k}(\alpha)$ is the auxiliary distribution, and to simplify the notation we have defined $q^k_\alpha(i\ell_k) \equiv q^k_\alpha(\left(e^O_k\right)_{i\ell_k})$.

Finally, for the term corresponding to non-excluding node attributes we have

$$\begin{aligned}
\mathcal{L}^{A_k} &= \sum_{(i,g)\in A^O_k} \log \sum_{\alpha\gamma} \theta_{i\alpha}\zeta^k_{g\gamma}\hat{q}_{\alpha\gamma}(ig) \\
&= \sum_{(i,g)\in A^O_k} \log \sum_{\alpha\gamma} \hat{\sigma}^k_{ig}(\alpha,\gamma) \frac{\theta_{i\alpha}\zeta^k_{g\gamma}\hat{q}_{\alpha\gamma}(ig)}{\hat{\sigma}^k_{ig}(\alpha,\gamma)} \\
&\geq \sum_{(i,g)\in A^O_k} \sum_{\alpha\gamma} \hat{\sigma}^k_{ig}(\alpha,\gamma) \log \frac{\theta_{i\alpha}\zeta^k_{g\gamma}\hat{q}_{\alpha\gamma}(ig)}{\hat{\sigma}^k_{ig}(\alpha,\gamma)} \qquad (2.17)
\end{aligned}$$

where $\hat{\sigma}^k_{ig}(\alpha,\gamma)$ is the auxiliary distribution, and to simplify the notation we have defined $\hat{q}^k_\alpha(ig) \equiv \hat{q}^k_{\alpha\gamma}(\left(a^O_k\right)_{ig})$.

Note that, in Eqs. (2.15)-(2.17) above, the equality is satisfied when maximizing with respect to the auxiliary distributions. By solving these optimization problems we obtain:

$$
\begin{aligned}
\omega_{ij}(\alpha,\beta) &= \frac{\theta_{i\alpha}\eta_{j\beta}p_{\alpha\beta}(r_{ij}^{O})}{\sum_{\alpha'\beta'}\theta_{i\alpha'}\eta_{j\beta'}p_{\alpha'\beta'}(r_{ij}^{O})}\,, && (2.18)\\
\sigma_{i\ell_k}^{k}(\alpha) &= \frac{\theta_{i\alpha}q_{\alpha}^{k}(i\ell_k)}{\sum_{\alpha'}\theta_{i\alpha'}q_{\alpha'}^{k}(i\ell_k)}\,, && (2.19)\\
\hat{\sigma}_{ig}^{k}(\alpha,\gamma) &= \frac{\theta_{i\alpha}\zeta_{g\gamma}^{k}\hat{q}_{\alpha\gamma}(ig)}{\sum_{\alpha'\gamma'}\theta_{i\alpha'}\zeta_{g\gamma'}\hat{q}_{\alpha'\gamma'}(ig)}\,. && (2.20)
\end{aligned}
$$

Therefore, the auxiliary distributions have the following interpretations: $\omega_{ij}(\alpha,\beta)$ is the contribution of user group α and item group β to the probability that user i gives item j a rating r_{ij}^{O}; $\sigma_{i\ell_k}^{k}(\alpha)$ is the contribution of user group (or item group) α to the probability that user (item) i has attribute type $(e_k^O)_{i\ell_k}$ in the k-th excluding attribute; and, finally, $\hat{\sigma}_{ig}^{k}(\alpha,\gamma)$ is the contribution of groups α and γ to the probability that, for the k-th non-excluding attribute, the association between node i and attribute g is of type $(a_k^O)_{ig}$.

To maximize the parametric log-postirior, we are going to compute the Lagrangian $\mathfrak{L}$ of the sum of all the likelihoods and priors:

$$
\begin{aligned}
\mathfrak{L} &= \sum_{(i,j)\in R^O}\sum_{\alpha,\beta}\omega_{ij}(\alpha,\beta)\log\frac{\theta_{i\alpha}\eta_{j\beta}p_{\alpha\beta}(r_{ij}^{O})}{\omega_{ij}(\alpha,\beta)} + \\
&+ \sum_{k}\lambda_k\sum_{(i,a)\in A_k^O}\sigma_{i\ell_k}^{k}(\alpha)\log\frac{\theta_{i\alpha}q_{\alpha}^{k}(i\ell_k)}{\sigma_{i\ell_k}^{k}(\alpha)} + \\
&+ \sum_{k}\lambda_k\sum_{(i,g)\in A_k^O}\sum_{\alpha\gamma}\hat{\sigma}_{ig}^{k}(\alpha,\gamma)\log\frac{\theta_{i\beta}\zeta_{g\gamma}^{k}\hat{q}_{\alpha\gamma}(ig)}{\hat{\sigma}_{ig}^{k}(\alpha,\gamma)} - \\
&- \sum_{i}\phi_i\sum_{\alpha}(\theta_{i\alpha}-1) - \sum_{j}\rho_j\sum_{\beta}(\eta_{j,\beta}-1) - \sum_{\alpha,\beta}\epsilon_{\alpha\beta}\sum_{r}(p_{\alpha\beta}(r_{ij}^{O})-1) - \\
&- \sum_{\alpha}\tau_\alpha\sum_{a}(q_{\alpha}^{k}(i\ell_k)-1) - \sum_{g}\alpha_g\sum_{\gamma}(\zeta_{g\gamma}^{k}-1) - \sum_{\beta,\gamma}\kappa_{\beta\gamma}\sum_{r}(\hat{q}_{\alpha\gamma}(ig)-1)
\end{aligned}
$$

Where $\phi_i, \rho_j, \epsilon_{\alpha\beta}, \tau_\alpha, \alpha_g$ and $\kappa_{\beta,\gamma}$ are the Lagrange multipliers that we have to find to compute the parameters.

2.3.1 Calculation of the parameters

Calculation of $\theta_{i\alpha}$

Computing the partial derivative respect $\theta_{i\alpha}$ and put it equal to 0, we get:

$$\frac{\partial \mathfrak{L}}{\partial \theta_{i\alpha}} = \sum_{j\in\partial i} \frac{\omega_{ij}(\alpha,\beta)}{\theta_{i\alpha}} + \sum_k \lambda_k \sum_{a\in\partial i} \frac{\sigma^k_{i\ell_k}(\alpha)}{\theta_{i\alpha}} + \sum_l \lambda_l \sum_{g\in\partial j} \frac{\hat{\sigma}^l_{ig}(\alpha,\gamma)}{\theta_{i\alpha}} - \phi_i = 0 \tag{2.21}$$

Solving 2.21 for $\theta_{i\alpha}$ we have:

$$\theta_{i\alpha} = \frac{\sum_{j\in\partial i}\sum_\beta \omega_{ij}(\alpha,\beta)}{\phi_i} + \sum_k \lambda_k \frac{\sum_{a\in\partial i}\sigma^k_{i\ell_k}(\alpha)}{\phi_i} + \sum_l \lambda_l \frac{\sum_{g\in\partial i}\hat{\sigma}^l_{ig}(\alpha,\gamma)}{\phi_i} \tag{2.22}$$

Now sum over α in both sides:

$$1 = \frac{\sum_{j\in\partial i} 1}{\phi_i} + \sum_k \lambda_k \frac{\sum_{a\in\partial i} 1}{\phi_i} + \sum_l \lambda_l \frac{\sum_{g\in\partial i} 1}{\phi_i} \tag{2.23}$$

Where we have take into account the normalization of $\theta_{i\alpha}, \omega_{ij}(\alpha,\beta)$, $\sigma^k_{i\ell_k}(\alpha)$ and $\hat{\sigma}^l_{ig}(\alpha,\gamma)$.

Solving 2.23 for ϕ_i and replacing it in 2.22, we obtain:

$$\theta_{i\alpha} = \frac{\sum_{j\in\partial i}\sum_\beta \omega_{ij}(\alpha,\beta) + \sum_k \lambda_k \sum_{a\in\partial i}\sigma^k_{i\ell_k}(\alpha) + \sum_l \lambda_l \sum_{g\in\partial_i{}^k}\sum_\gamma \hat{\sigma}^l_{ig}(\alpha,\gamma)}{d_i + \sum_k \lambda_k \delta^k_i + \sum_l \lambda_l \Delta^l_i} \tag{2.24}$$

Where d_i is the degree of user i in the network of ratings, $\delta^k_i = 1$ if user i has exclusive attribute ℓ_k and zero otherwise , and $\Delta^l_i \equiv |\partial_i{}^l|$.

Calculation of $\eta_{j\beta}$

Computing the partial derivative respect $\eta_{j\beta}$ equal to 0 we get:

$$\frac{\partial \mathfrak{L}}{\partial \eta_{j\beta}} = \sum_{i\in\partial j}\sum_\alpha \frac{\omega_{ij}(\alpha,\beta)}{\eta_{j\beta}} + \sum_k \lambda_k \sum_{a\in\partial j} \frac{\sigma^k_{j\ell_k}(\alpha)}{\theta_{j\alpha}} + \sum_l \lambda_l \sum_{g\in\partial j} \frac{\hat{\sigma}^l_{ig}(\alpha,\gamma)}{\eta_{j\beta}} - \rho_j = 0 \tag{2.25}$$

where ∂^k_j is the set of k-th attributes associated with item j. Note that for the attributes we change $\theta_{i\alpha}$ for $\eta_{j\beta}$. Solving 2.25 for $\eta_{j,\beta}$ we have:

$$\eta_{j\beta} = \frac{\sum_{i\in\partial j}\sum_\alpha \omega_{ij}(\alpha,\beta)}{\rho_j} + \sum_k \lambda_k \frac{\sum_{a\in\partial j}\sigma^k_{j\ell_k}(\alpha)}{\rho_j} + \sum_l \lambda_l \frac{\sum_{g\in\partial j}\hat{\sigma}^l_{ig}(\alpha,\gamma)}{\rho_j} \tag{2.26}$$

Now sum over β in both sides:

$$1 = \frac{\sum_{i\in\partial j} 1}{\rho_j} + \sum_k \lambda_k \frac{\sum_{a\in\partial^k_j} 1}{\rho_j} + \sum_l \lambda_l \frac{\sum_{g\in\partial^l_j} 1}{\rho_j} \tag{2.27}$$

Where we have take into account the normalization of $\eta_{j\beta}, \omega_{ij}(\alpha,\beta)$, $\sigma^k_{i\ell_k}(\alpha)$ and $\hat{\sigma}^l_{ig}(\alpha,\gamma)$.

Solving 2.27 for ρ_j and replacing it in 2.22, we obtain:

$$\eta_{j\beta} = \frac{\sum_{i\in\partial j}\sum_\alpha \omega_{ij}(\alpha,\beta) + \sum_k \lambda_k \sigma^k_{j\ell_k}(\beta) + \sum_l \lambda_l \sum_{i\in\partial^k_j}\sum_\gamma \hat{\sigma}^l_{ij}(\beta,\gamma)}{d_j + \sum_k \lambda_k \delta^l_j + \sum_l \lambda_l \Delta^l_j} \tag{2.28}$$

where d_j is the degree of item j in the network of ratings, and $\Delta^l_j = |{\partial_j}^l|$. As before, the term $\sigma^k_{j\ell_k}(\beta)$ is equal to zero if item j does not have attribute ℓ_k, so that $\delta^k_j = 1$ if item j has exclusive attribute ℓ_k and zero otherwise.

Calculation of $p_{\alpha\beta}(r^O_{ij})$

Computing the partial derivative respect $p_{\alpha\beta}(r^O_{ij})$ equal to 0, we get:

$$\frac{\partial \mathfrak{L}}{\partial p_{\alpha\beta}(r^O_{ij})} = \sum_{(i,j)\in R^O|r_{i,j}=r} \frac{\omega_{ij}(\alpha,\beta)}{p_{\alpha\beta}(r^O_{ij})} - \epsilon_{\alpha\beta} = 0 \tag{2.29}$$

Solving 2.29 for $p_{\alpha\beta}(r^O_{ij})$ we have:

$$p_{\alpha\beta}(r^O_{ij}) = \frac{\sum_{(i,j)\in R^O|r_{i,j}=r} \omega_{ij}(\alpha,\beta)}{\epsilon_{\alpha\beta}} \tag{2.30}$$

Now sum over r in both sides:

$$1 = \frac{\sum_{(i,j)\in R^O}\sum_r \omega_{ij}(\alpha,\beta)}{\epsilon_{\alpha\beta}} \tag{2.31}$$

Where we have take into account the normalization of $p_{\alpha\beta}(r^O_{ij})$. Solving 2.31 for $\epsilon_{\alpha\beta}$ and replacing it in 2.30, we obtain:

$$p_{\alpha\beta}(r^O_{ij}) = \frac{\sum_{(i,j)\in R^O|r_{i,j}=r} \omega_{ij}(\alpha,\beta)}{\sum_{(i,j)\in R^O} \omega_{ij}(\alpha,\beta)} \tag{2.32}$$

Calculation of $q_\alpha^k(e)$

Computing the partial derivative respect $q_\alpha^k(e)$ equal to 0, we get:

$$\frac{\partial \mathfrak{L}}{\partial q_\alpha^k(e)} = \sum_{(i,\ell_k)\in A_k^O|(e_k^O)_{i\ell_k}=e} \frac{\sigma_{i\ell_k}^k(\alpha)}{q_\alpha^k(e)} - \tau_\alpha = 0 \tag{2.33}$$

Solving 2.33 for $q_\alpha^k(e)$ we have:

$$q_\alpha^k(e) = \frac{\sum_{(i,\ell_k)\in A_k^O|(e_k^O)_{i\ell_k}=e} \sigma_{i\ell_k}^k(\alpha)}{\tau_\alpha} \tag{2.34}$$

Now sum over e in both sides:

$$1 = \frac{\sum_{(i,\ell_k)\in A_k^O} \sigma_{i\ell_k}^k(\alpha)}{\tau_\alpha} \tag{2.35}$$

Where we have take into account the normalization of $q_\alpha^k(i\ell_k)$. Solving 2.35 for τ_α and replacing it in 2.34, we obtain:

$$q_\alpha^k(e) = \frac{\sum_{(i,\ell_k)\in A_k^O|(e_k^O)_{i\ell_k}=e} \sigma_{i\ell_k}^k(\alpha)}{\sum_{(i,\ell_k)\in A_k^O} \sigma_{i\ell_k}^k(\alpha)} \tag{2.36}$$

Calculation of $\zeta_{g\gamma}^k$

Computing the partial derivative respect $\zeta_{g\gamma}^k$ equal to 0, we get:

$$\frac{\partial \mathfrak{L}}{\partial \zeta_{g\gamma}^k} = \sum_{g\in\partial_g^k} \sum_{\alpha} \frac{\hat{\sigma}_{ig}^k(\alpha,\gamma)}{\zeta_{g\gamma}^k} - \alpha_g = 0 \tag{2.37}$$

Solving 2.37 for $\zeta_{g\gamma}^k$ we have:

$$\zeta_{g\gamma}^k = \frac{\sum_{g\in\partial_g^k} \sum_\alpha \hat{\sigma}_{ig}^k(\alpha,\gamma)}{\alpha_g} \tag{2.38}$$

Now sum over γ in both sides:

$$1 = \frac{\sum_{g\in\partial_g^k} 1}{\alpha_g} \tag{2.39}$$

where we have taken into account the normalization of $\zeta_{g\gamma}^k$ and $\hat{\sigma}_{ig}^k(\alpha,\gamma)$.

Solving 2.39 for α_g and replacing it in 2.22, we obtain:

$$\zeta_{g\gamma}^k = \frac{\sum_{g\in\partial_g^k}\sum_\alpha \hat{\sigma}_{ig}^k(\alpha,\gamma)}{\Delta_g^k}, \tag{2.40}$$

where ∂_g^k is the set of nodes associated with attribute g, and $\Delta_g^k = |\partial_g{}^k|$.

Calculation of $\hat{q}_{\alpha\gamma}(a)$

Computing the partial derivative respect $\hat{q}_{\alpha\gamma}(a)$ equal to 0, we get:

$$\frac{\partial\mathfrak{L}}{\partial\hat{q}_{\alpha\gamma}(a)} = \sum_{(i,g)\in A_k^O|(a_k^O)_{ig}=a} \frac{\hat{\sigma}_{ig}^k(\alpha,\gamma)}{\hat{q}_{\alpha\gamma}(a)} - \kappa_{\alpha,\gamma} = 0 \tag{2.41}$$

Solving 2.41 for $\hat{q}_{\alpha\gamma}(a)$ we have:

$$\hat{q}_{\alpha\gamma}(a) = \frac{\sum_{(i,g)\in R^O}\hat{\sigma}_{ig}^k(\alpha,\gamma)}{\kappa_{\alpha,\gamma}} \tag{2.42}$$

Now sum over a in both sides:

$$1 = \frac{\sum_{(i,g)\in A_k^O|(a_k^O)_{ig}=a}\hat{\sigma}_{ig}^k(\alpha,\gamma)}{\kappa_{\alpha,\gamma}} \tag{2.43}$$

Where we have take into account the normalization of $\hat{q}_{\alpha\gamma}(a)$. Solving 2.43 for $\kappa_{\alpha,\gamma}$ and replacing it in 2.42, we obtain:

$$\hat{q}_{\alpha\gamma}(a) = \frac{\sum_{(i,g)\in A_k^O|(a_k^O)_{ig}=a}\hat{\sigma}_{ig}^k(\alpha,\gamma)}{\sum_{(i,g)\in A_k^O}\hat{\sigma}_{ig}^k(\alpha,\gamma)} \tag{2.44}$$

2.3.2 Summary of the parameters:

As a summary, we can classify the parameters that we have to compute as auxiliary functions, membership matrices and probability matrices. The auxiliary functions are:

$$\begin{aligned}
\omega_{ij}(\alpha,\beta) &= \frac{\theta_{i\alpha}\eta_{j\beta}p_{\alpha\beta}(r_{ij}^{O})}{\sum_{\alpha'\beta'}\theta_{i\alpha'}\eta_{j\beta'}p_{\alpha'\beta'}(r_{ij}^{O})}\,, \\
\sigma_{i\ell_k}^{k}(\alpha) &= \frac{\theta_{i\alpha}q_{\alpha}^{k}(i\ell_k)}{\sum_{\alpha'}\theta_{i\alpha'}q_{\alpha'}^{k}(i\ell_k)}\,, \\
\hat{\sigma}_{ig}^{k}(\alpha,\gamma) &= \frac{\theta_{i\alpha}\zeta_{g\gamma}^{k}\hat{q}_{\alpha\gamma}(ig)}{\sum_{\alpha'\gamma'}\theta_{i\alpha'}\zeta_{g\gamma'}\hat{q}_{\alpha'\gamma'}(ig)}\,.
\end{aligned}$$

The membership matrices are:

$$\theta_{i\alpha} = \frac{\sum_{j\in\partial i}\sum_{\beta}\omega_{ij}(\alpha,\beta)+\sum_{k}\lambda_k\sigma_{i\ell_k}^{k}(\alpha)+\sum_{l}\lambda_l\sum_{g\in\partial_i{}^{k}}\sum_{\gamma}\hat{\sigma}_{ig}^{l}(\alpha,\gamma)}{d_i+\sum_k\lambda_k\delta_i^k+\sum_l\lambda_l\Delta_i^l}\,,$$

$$\eta_{j\beta} = \frac{\sum_{i\in\partial j}\sum_{\alpha}\omega_{ij}(\alpha,\beta)+\sum_{k}\lambda_k\sigma_{j\ell_k}^{k}(\beta)+\sum_{l}\lambda_l\sum_{i\in\partial_j^{k}}\sum_{\gamma}\hat{\sigma}_{ij}^{l}(\beta,\gamma)}{d_j+\sum_k\lambda_k\delta_j^k+\sum_l\lambda_l\Delta_j^l}\,,$$

$$\zeta_{g\gamma}^{k} = \frac{\sum_{i\in\partial_g^k}\sum_{\alpha}\hat{\sigma}_{ig}^{k}(\alpha,\gamma)}{\Delta_g^k}\,.$$

And the probability matrices are:

$$p_{\alpha\beta}(r) = \frac{\sum_{(i,j)\in R^{O}|r_{ij}^{0}=r}\omega_{ij}(\alpha,\beta)}{\sum_{(i,j)\in R^{O}}\omega_{ij}(\alpha,\beta)}$$

$$q_{\alpha}^{k}(e) = \frac{\sum_{(i,\ell_k)\in A_k^{O}|(e_k^{O})_{i\ell_k}=e}\sigma_{i\ell_k}^{k}(\alpha)}{\sum_{(i,\ell_k)\in A_k^{O}}\sigma_{i\ell_k}^{k}(\alpha)}$$

$$\hat{q}_{\alpha\gamma}^{k}(a) = \frac{\sum_{(i,g)\in A_k^{O}|(a_k^{O})_{ig}=a}\hat{\sigma}_{ig}^{k}(\alpha,\gamma)}{\sum_{(i,g)\in A_k^{O}}\hat{\sigma}_{ig}^{k}(\alpha,\gamma)}$$

To find the parameters, we use an expectation-maximization algorithm (Ref. [4, 6, 7]), where the details can be find it in the Appendix B. An implementation of this algorithm, that we developed, can be found in a GitHub repository (Ref. [64]).

2.4 Synthetic data

We first use synthetic data to validate the expectation-maximization inference approach and to investigate the role of introducing node attributes. We generate synthetic data as shown in Fig. 2.1. Here and throughout the validations in the coming sections, we quantify link prediction performance by measuring rating prediction accuracy, that is, the fraction of correctly predicted ratings in cross-validation experiments, using k-fold cross-validation for $k = 5$ (see Appendix A for more information). Then, for each accuracy, we divided it by the accuracy with $\lambda_{\text{user}} = \lambda_{\text{item}} = 0$ and make the logarithm:

$$\alpha\left(\lambda_{\text{user}}, \lambda_{\text{item}}\right) \equiv \log \frac{\text{acc}\left(\lambda_{\text{user}}, \lambda_{\text{item}}\right)}{\text{acc}(0,0)} \tag{2.45}$$

With that we can see how good are our predictions respect no having metadata. If it is negative, that means that our predictions get worst. If it is equal to 0, there is no improvements, and if it is positive, using metadata improves the predictions.

Our synthetic rating networks consist of 200 users and 200 items, partitioned into $K = 2$ groups of users and $L = 4$ groups of items. Users have an excluding attribute labeled "male" or "female", and items have an excluding attribute labeled from 0 to 3, which may represent four different genres.

In the simplest case, in which ratings and attributes are completely correlated, all female users have membership vectors $\boldsymbol{\theta}_{\text{f}} = (0.8, 0.2)$; conversely, all male users have $\boldsymbol{\theta}_{\text{m}} = (0.2, 0.8)$. Similarly, an item with attribute a has a membership of 0.8 to group a and 0.067 to all other groups. Finally, for the probabilities that a user from group α is connected to an item of group β, we selected the following matrices:

$$\begin{aligned}
\boldsymbol{p}(r=0) &= \begin{pmatrix} 0.1 & 0.1 & 0.1 & 0.8 \\ 0.8 & 0.1 & 0.1 & 0.1 \end{pmatrix}, \\
\boldsymbol{p}(r=1) &= \begin{pmatrix} 0.1 & 0.1 & 0.8 & 0.1 \\ 0.1 & 0.8 & 0.1 & 0.1 \end{pmatrix}, \\
\boldsymbol{p}(r=2) &= \begin{pmatrix} 0.8 & 0.8 & 0.1 & 0.1 \\ 0.1 & 0.1 & 0.8 & 0.8 \end{pmatrix}.
\end{aligned}$$

To simulate partial correlation c or even no correlation ($c = 0$) between membership vectors and attributes, we reassign, with probability $1 - c$, the node attribute to a value selected uniformly at random among all possibilities (2 for users and 4 for items).

For the experiments reported in Fig. 2.3, we consider a number $|R^O| = 400$ of observed ratings (that is, 1% of all generated ratings), and all attribute links. Although the synthetic data are created with item genre as an excluding attribute, we carry out the inference process assuming that genre is a non-excluding attribute, which is what one would likely assume in real settings where the generating model is unknown.

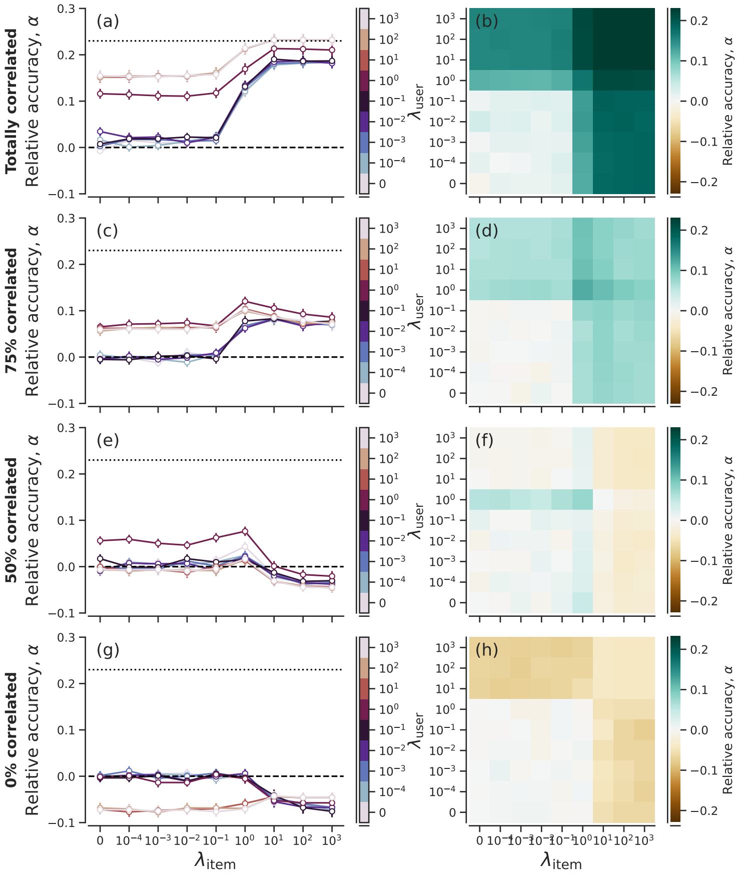

Figure 2.3: Predictive performance and effect of metadata on synthetic ratings. We create synthetic ratings from 200 users on 200 items, with different levels of correlation c between ratings and node attributes (see text). We then use 5-fold cross-validation to calculate the performance of the expectation-maximization equations at predicting unobserved ratings. In particular, we take as a reference the predictive accuracy π_0 of the algorithm when all attributes are ignored ($\lambda_{\text{user}} = \lambda_{\text{item}} = 0$), and measure relative accuracy α for a given pair $(\lambda_{\text{user}}, \lambda_{\text{item}})$ as the log-ratio $\alpha(\lambda_{\text{user}}, \lambda_{\text{item}}) = \log\left[\pi(\lambda_{\text{user}}, \lambda_{\text{item}})/\pi_0\right]$. The value $\alpha(\lambda_{\text{user}}, \lambda_{\text{item}}) = 0$ (dashed line) thus indicates no change with respect to the reference π_0, and $\alpha(\lambda_{\text{user}}, \lambda_{\text{item}}) > 0$ (respectively, $\alpha(\lambda_{\text{user}}, \lambda_{\text{item}}) < 0$) indicates predictions that are more (less) accurate than those obtained by ignoring node attributes. The maximum possible relative performance (dotted line) is obtained when each rating is assigned the exact probability that was used to generate it. For each value of the correlation ((a)-(b), full correlation, $c = 1$; (c)-(d), $c = 0.75$; (e)-(f), $c = 0.50$; (g)-(h), no correlation, $c = 0$) we show the variation of $\alpha(\lambda_{\text{user}}, \lambda_{\text{item}})$ with λ_{item} for different values of λ_{user} (left), and the whole dependence of $\alpha(\lambda_{\text{user}}, \lambda_{\text{item}})$ on both λ_{user} and λ_{item} (right).

We infer the values of the model parameters using the expectation-maximization equations, and use the inferred parameters to predict unobserved ratings in the bipartite ratings network. We do this for different levels of correlation c between the ratings and the attribute networks (Fig. 2.3), from a situation $c = 1$ in which the attributes are perfectly correlated with user and item membership vectors (all male users belong to one group and have identical parameters, and all females belong to another group with different parameters; items with each genre belong to the exact same mixture of groups) to a situation $c = 0$ in which user and item memberships and attributes are completely uncorrelated (Fig. 2.3).

Since we focus on sparse observations in which the number of observed ratings is low (only 1% of all ratings), model parameters cannot be inferred accurately from the ratings alone. Therefore, when we only consider the observed ratings R^O and ignore all attributes A^O by setting $\lambda_{\text{user}} = \lambda_{\text{item}} = 0$ in Eq. (2.13) (λ_{user} and λ_{item} correspond to the user and item attribute networks, respectively), the prediction of unobserved links is suboptimal, that is, the inferred probabilities of unobserved links differ significantly from the actual probabilities used to build the network.

When there is perfect correlation between node attributes and group memberships, considering the attributes A^O by setting $\lambda_{\text{user}} > 0$ and $\lambda_{\text{item}} > 0$ should in principle help in the inference process. In fact, since attributes are perfectly correlated to group memberships, in the limit $\lambda_{\text{user}} \to \infty$ and $\lambda_{\text{item}} \to \infty$ nodes will be forced into the correct groups and predictions should be near optimal. This is what we observe in our numerical experiments (Fig. 2.3a). Interestingly, as we increase the weight of the attributes in the log-posterior from $\lambda_{\text{user}} = \lambda_{\text{item}} = 0$, the effect on prediction accuracy is not smooth. Rather, below certain threshold values of λ_{user} and λ_{item}, using the attributes does not have any significant effect on prediction accuracy. Then, at those threshold values, a transition occurs and prediction accuracy increases abruptly until it reaches its theoretical maximum, as expected.

When attributes and ratings are completely uncorrelated (Fig. 2.3d), the role of attributes is reversed. Predictions are equally suboptimal at $\lambda_{\text{user}} = \lambda_{\text{item}} = 0$, but then, as λ_{user} and λ_{item} cross certain threshold values, predictions suddenly worsen as user and item nodes are forced into groups that are uncorrelated with their real membership vectors and, thus, with the observed ratings.

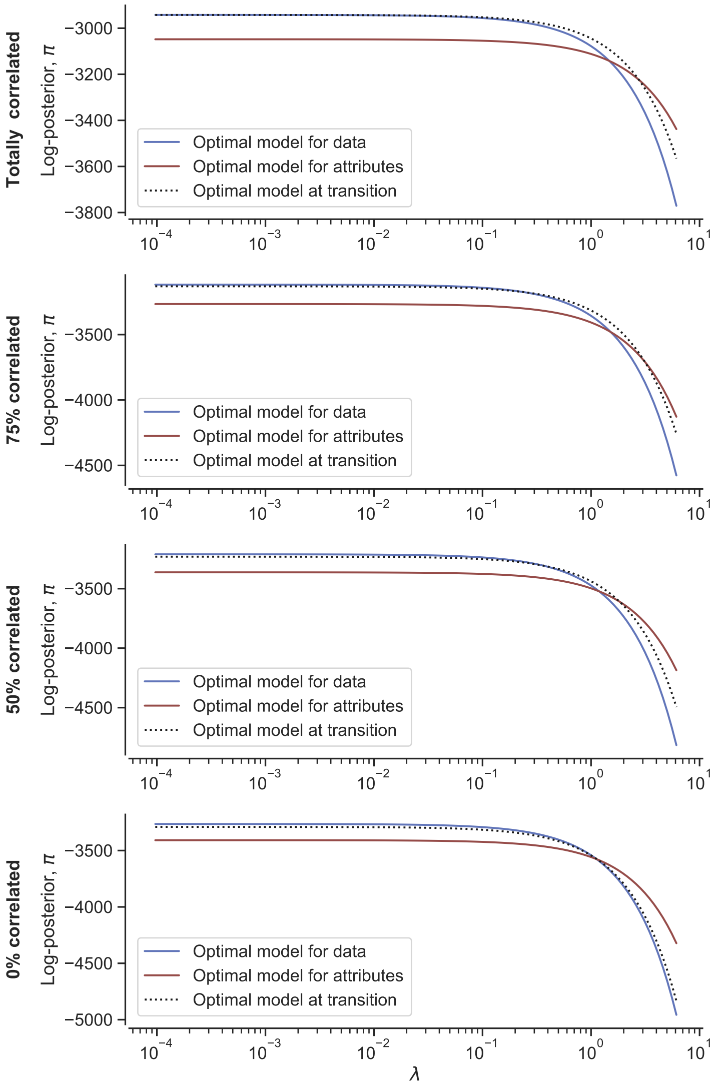

Figure 2.4: Transition between data-dominated and metadata-dominated inference regimes. For the synthetic data in Fig. 2.3, we plot the log-posterior $\pi(\boldsymbol{\theta}, \boldsymbol{\eta}, \boldsymbol{\zeta}, \boldsymbol{p}, \boldsymbol{q}, \hat{\boldsymbol{q}}|R^O, A^O)$ as a function of the hyperparameter $\lambda = \lambda_{\text{item}} = \lambda_{\text{user}}$ for three models: the model that maximizes the data likelihood L^R, the model that maximizes the metadata likelihood L^A, and the model that maximizes the posterior when two previous cases cross (that is, have equal posteriors). The position of the crossing coincides with the transitions and the maxima observed in Fig. 2.3

Unlike the extreme cases of total correlation or zero correlation, when attributes are partly correlated with the true group memberships of the nodes, the change in performance is not monotonic as we increase the importance of the attributes. As before, when λ_{user} and λ_{item} are small enough, we observe no difference with the situation in which the attributes are ignored entirely. In the other extreme, when $\lambda_{\text{user}} \to \infty$ and $\lambda_{\text{item}} \to \infty$ user and item nodes are forced into groups that match partly, but not perfectly, the true group memberships of the nodes, so the performance may increase or decrease with respect to the situation with no attributes, depending on whether the correlation is higher (Fig. 2.3b) or lower (Fig. 2.3c). However, we find that the most predictive models in this case are those at intermediate values of λ_{user} and λ_{item}, precisely at the transition region where both the observed ratings and the observed attributes play a role in determining the most plausible group memberships. In this case, the inferred node memberships do not coincide with either those that maximize L^R of those that maximize L^{A_k}.

To understand the transition from the rating-dominated to the attribute-dominated regime, we study the posterior of the two extreme models corresponding to the maximum a posterior estimates obtained by expectation-maximization for $\lambda_{\text{user}} = \lambda_{\text{item}} = 0$ and for $\lambda_{\text{user}} = \lambda_{\text{item}} \to \infty$ (Fig. 2.4). These are the most plausible models when only data (ratings) and only metadata (attributes) are taken into consideration, respectively. Regardless of the correlation between ratings and attributes, we find that the transition in predictability in Fig. 2.3 coincides with the region where the data-dominated and metadata-dominated posteriors cross. By considering Eq. (2.13) we see that this must be the case. Indeed, for each attribute network we find three regimes—one dominated by the L^R term, one dominated by the L^A term, and one in which both terms are comparable. Unless there is perfect or almost perfect correlation between attributes and node memberships, any improvement in predictive power must come from considering both the observed ratings and the observed attributes, and therefore in the transition region.

2.5 Theoretical interpretation of the transition

To better understand this transition, we look at the posterior of the two models corresponding to the maximum a posterior estimates for $\lambda_{\text{user}} = \lambda_{\text{item}} = 0$ and for $\lambda_{\text{user}} = \lambda_{\text{item}} \to \infty$ (Fig. 2.4). These are the most plausible models when only data (ratings) and only metadata (attributes), respectively, are taken into consideration.

If we draw upon the analogy between Bayesian statistics and statistical mechanics [2, 3, 19], we can equate the log-posterior to the (minus) free energy of a physical system, and interpret the crossover in terms of a transition in which λ plays the role of the tuning (temperature-like) parameter. Within this framework, these extreme models are the dominating maxima in the posterior landscape (or, by analogy, the states of the system at the two sides of the transition) and, therefore, the predictability crossover occurs when the data-dominated and metadata-dominated log-posteriors cross, that is, for a value λ^* such that

$$\mathcal{L}_0^R + \lambda^* \mathcal{L}_0^A = \mathcal{L}_\infty^R + \lambda^* \mathcal{L}_\infty^A \tag{2.46}$$

or

$$\lambda^* = \frac{\mathcal{L}_0^R - \mathcal{L}_\infty^R}{\mathcal{L}_\infty^A - \mathcal{L}_0^A} \, . \tag{2.47}$$

Here, the subindex 0 (or ∞) indicates the quantities corresponding to the model that maximizes the posterior for $\lambda_{\text{user}} = \lambda_{\text{item}} = 0$ (respectively, $\lambda_{\text{user}} = \lambda_{\text{item}} \to \infty$), and we group all attributes in a single term $\mathcal{L}^A$. As we show in Fig. 2.4, we find that, indeed, the transition in predictability in Fig. 2.3 coincides (at least in order of magnitude) with the point where the data-dominated and metadata-dominated log-posteriors cross.

Importantly, all log-likelihoods are extensive quantities. Therefore, the dependency on the number of observed ratings N_R and attributes N_A can be made explicit by defining intensive (that is, *per-link*) log-likelihoods $\ell^R = \mathcal{L}^R / N_R$ and $\ell^A = \mathcal{L}^A / N_A$. Then

$$\lambda^* \sim \frac{N_R}{N_A} \, , \tag{2.48}$$

and at the transition point λ^* we have that both $\mathcal{L}^R \sim N_R$ and $\lambda^* \mathcal{L}^A \sim N_R$ are of the same order. By considering Eq. (2.13) we see that this must be the case. Indeed, for each attribute network we find three regimes—one dominated by the $\mathcal{L}^R$ term, one dominated by the $\mathcal{L}^A$ term, and one in which both terms are comparable. Unless there is perfect or almost perfect correlation between attributes and node memberships, any improvement in predictive power must come from considering both the observed ratings and the observed attributes, and therefore in the transition region that we have identified.

2.6 Real data

Finally, we analyze two empirical data sets and study whether we observe the same behaviors as in synthetic networks. First, we consider the 100K MovieLens data set Ref. [44], which contains 100,000 ratings of movies by users. Age and gender attributes are available for users, which we model as excluding attributes (Fig. 2.5). Movies have genre attributes, which we model as non-excluding attributes. The relative weights of user and movie attributes are given by the parameters λ_{users} and λ_{items}.

Just as in the synthetic networks with small but finite correlation, we observe an intermediate value of λ_{user} and λ_{item} that provides more accurate rating predictions than either considering the observed ratings alone or considering the node attributes alone. This behavior is similar when we consider age only, gender only, or age and gender simultaneously. As in synthetic networks, the optimal combination of rating data and node metadata occurs for values of λ such that the ratings network and the attributes networks have comparable contributions to the log-posterior.

Second, we consider a data set on the votes of 441 members of the U.S. House of Representatives in the 108th U.S. Congress [33] (Fig. 2.6). Between Jannuary 2003 and Jannuary 2005, these representatives voted on 1,217 bills, casting one of 9 different types of vote, which, following previous analyses, we simplify to Yes, No, and Other [33]. In this data set, “users” are the representatives and “items” are the bills. The ratings represent the votes of the representatives on the bills. For representatives, we have attribute data indicating their party and state, which we model as excluding attributes. Although all votes of all members are recorded in the data set (in total, 536698 votes), for the purpose of our analysis we infer the parameters of the multipartite mixed-membership stochastic block model using 1% of the data, and predict the remaining 99% (and repeat this using each 1% of the data

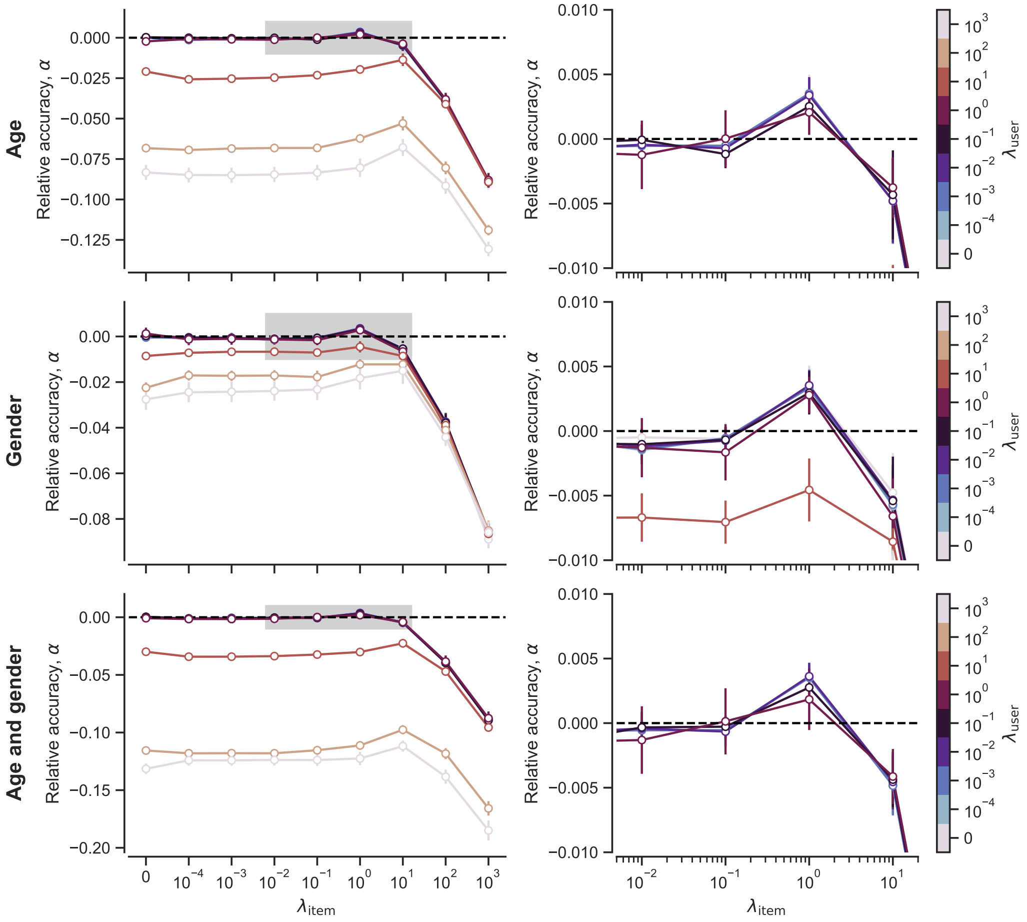

Figure 2.5: Predictive performance and effect of metadata on the MovieLens data set. As in Fig. 2.3, we take as a reference the predictive accuracy π_0 of the algorithm when all attributes are ignored ($\lambda_{\text{user}} = \lambda_{\text{item}} = 0$), and measure relative accuracy α for a given pair $(\lambda_{\text{user}}, \lambda_{\text{item}})$ as the log-ratio $\alpha(\lambda_{\text{user}}, \lambda_{\text{item}}) = \log[\pi(\lambda_{\text{user}}, \lambda_{\text{item}})/\pi_0]$. We consider three different attributes user nodes: (a)-(b), age; (c)-(d), gender; (e)-(f), age and gender combined. We plot the whole range of λ_{user} (left), and zoom into the intermediate (shaded) region of λ_{user} in which predictions are more accurate than the reference (right).

as training set).

Again, the effects of introducing the attributes in the inference process are very similar to those we encounter in synthetic data (Fig. 2.6). When using only the state of the representatives, we observe a behavior that is compat-

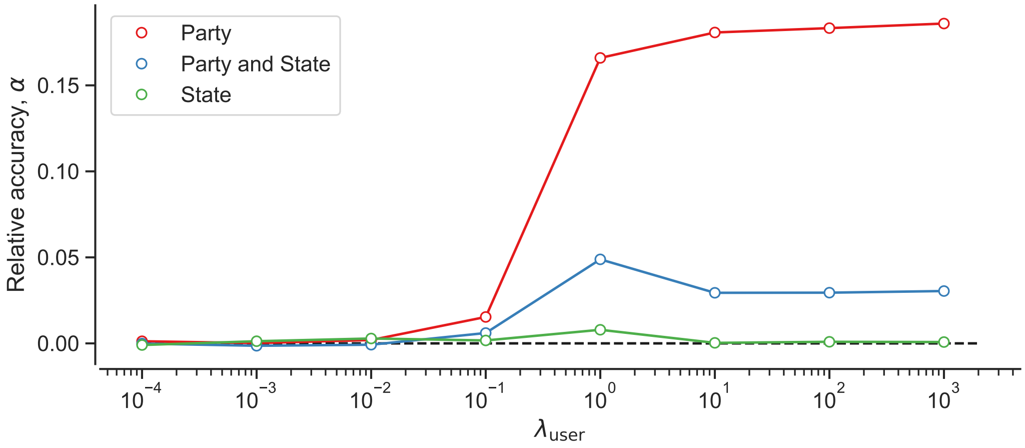

Figure 2.6: Predictive performance and effect of metadata on the U.S. Congress data set. As in Fig. 2.3, we take as a reference the predictive accuracy π_0 of the algorithm when all attributes are ignored ($\lambda_{\text{user}} = 0$), and measure relative accuracy α for a given λ_{user} as the log-ratio $\alpha(\lambda_{\text{user}}) = \log[\pi(\lambda_{\text{user}})/\pi_0]$. We consider three different attributes for user nodes: Party, State, and party and State simultaneously.

ible with small but finite correlation between attribute and voting patterns, since the optimal predictive performance is observed at intermediate values of λ_{user}. Rather, when we consider party affiliation we observe a behavior that is compatible with almost perfect correlation between attribute and voting behavior. Indeed, in this case the predictive performance of the model increases monotonically with λ_{user}, with an abrupt transition at $\lambda_{\text{user}} \approx 1$, just as for perfectly correlated attributes in synthetic data. When state and party are combined into a single excluding attribute (for example, "Democrat from Texas" is a group), we observe a behavior compatible with strong (but imperfect) correlation between attributes and voting behavior. In this case, predictive accuracy does not improve monotonically with λ_{user} because, for very large values, representatives are forced into small groups that are more prone to fluctuations, that is, the model overfits the data thus worsening the predictive power with respect to considering large groups associated to party affiliation alone.

2.7 Discussion

In this chapter we show that there is an evidence that metadata can help us in the inference problem. Also, thanks to the mechanism that we have used to add metadata to our inference problem we could study how this metadata effects the inference process and see the interplay between data and metadata.

We observed that metadata can help us depending if satisfies a couple of conditions at the same time. The first one is that metadata is correlated with our data, else our predictions will get worse when metadata is included. The second condition is that amount of metadata and data have to be such that their likelihoods (L^R and L^A) are of the same order. If this condition is not fulfilled, you will be able to make predictions with just the information with the highest likelihood.

Chapter 3

Transition in the detectability of closed-form mathematical models from noisy data

3.1 Introduction

In this chapter we analyze our second and final case study, a symbolic regression problem. In particular we study the interplay between the noise the observed data, the size of the dataset and the detectability of the true model that generated the data, i.e. the ability of an algorithm to find the true model. To this end, we will use a symbolic regression approach to obtain models from data.

Finally, symbolic regression is the task of identifying a closed-form model $y = F(\boldsymbol{x})$ from a certain dataset $D = \{(y^1, \boldsymbol{x}^1), (y^2, \boldsymbol{x}^2), ..., (y^N, \boldsymbol{x}^N)\}$. This is in contrast to *traditional* regression, in which one starts by proposing a model (linear regression, logistic regression or any other closed-form expression) and then only adjusts the parameters that better fit the data to the proposed model. Some computational approaches have been developed to that purpose (Ref. [25, 31, 34, 46, 67]) We start by defining precisely the problem that we want to solve. Consider that the true model that our data comes from is $y = F(\boldsymbol{x}, \theta)$ of K variables $\boldsymbol{x} = \{x_1, x_2, ...x_K\}$ and L parameters $\theta \in R^L$. As note earlier, our dataset D is a set of points $D = \{(y^1, \boldsymbol{x}^1), (y^2, \boldsymbol{x}^2), ..., (y^N, \boldsymbol{x}^N)\}$ and we are going to assume that each point has an error ϵ^k so $y^k = F(\boldsymbol{x}^k, \theta) + \epsilon^k$. Symbolic regression looks for the best closed-form expression $f(\boldsymbol{x})$ in a space of mathematical closed-form expressions. In symbolic regression we want the simplest models that satisfy the Occam Razor principle and fit the data well.

There are a several methods of symbolic regression in the literature, which can be classified in three categories: genetic programming methods, sparse regression methods, and a mixture of both methods. Genetic programming methods [20, 31, 57] start from a random generated closed-form expression that is represented as a network tree, where each internal node is a mathematical operation (sum, multiplication, sine...) and each leave is a variable or a parameter (Fig. 3.1). These trees are generated randomly and then new ones are generated using Darwinian natural selection and genetic operations. An example of Darwinian procedure is Darwinian reproduction. It is "asexual" and consists in selecting a random expression from our population according to some selection method based on fitness. After selecting our expressions, they are copied to another population that will be the new generation. An example of a genetic procedure is the genetic crossover or recombination. It consists in selecting two parents with the same criteria as before. Then for each selected parent, a piece of the tree is randomly selected and then these pieces of trees swapped. Genetic programming methods let us explore a large space of closed-form models, with no restrictions in the number of parameters, variables and mathematical functions. However, it cannot guarantee that the best models are explored with higher frequency.

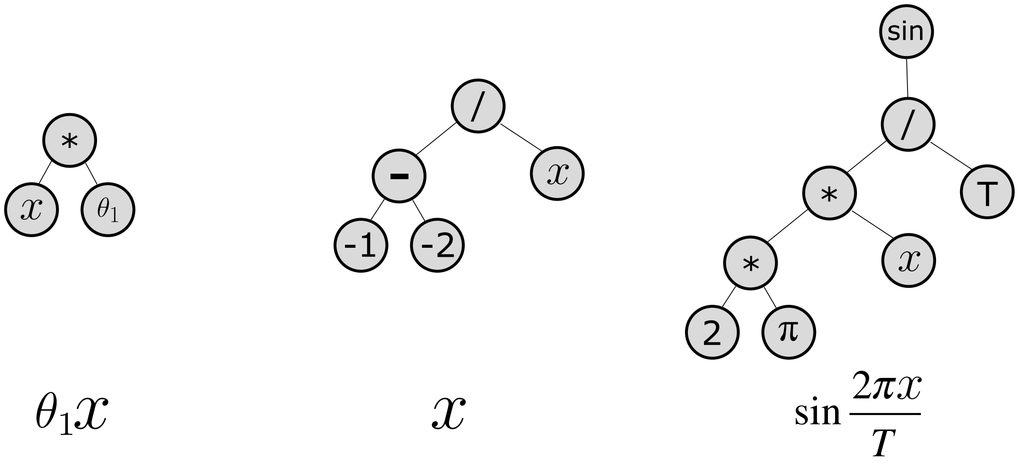

Figure 3.1: Examples of expressions represented by networks trees. Three examples of closed-form expressions and their network trees representations.

The other major class of methods to perform symbolic regression are sparse regression methods ([36, 46, 52, 54, 55]). These methods assume that the closed-form function $f\left(\boldsymbol{x}(t)\right)$ consists of a linear combination of a few "basis functions" $\{B_i\}$, so that

$$f\left(\boldsymbol{x}(t)\right) = a_0 + \sum_{i=1}^{N_R} a_i B_i(\boldsymbol{x}(t)), \tag{3.1}$$

where a_i are the sparse coefficients and B_i are the library of functions that are candidates. Then, we force the elimination of the functions that are not good performing a sparse regression.

Often, differential equations are well described using this approach because their forms often follows the assumption of linearity on relatively simple basis functions.

In general, these methods consists in propose that $f\left(\boldsymbol{x}(t)\right)$ is a linear combination of basis functions that are candidates. The restriction of the model being a linear combination of simple functions limits the space of functions that these methods can explore.

In this chapter we are going to use a Bayesian approach to symbolic regression, the Bayesian machine scientist (Ref. [65, 66, 68, 70]). This method intrinsically tries to fit the data and includes a measure of the structural complexity of the model. The goodness of fit is encoded in the likelihood and the complexity of the model in the prior. The Bayesian machine scientist can successfully recover the true generating model for synthetic data (Ref. [65]). With this in mind, we study how the quality of the data affects this problem and give an explanation from a Bayesian point of view.

3.2 The Bayesian machine scientist

Let us start by defining precisely the problem that we want to solve. Consider that the true model that our data comes from is $y = F(\boldsymbol{x}, \theta)$ of K variables $\boldsymbol{x} = \{x_1, x_2, ...x_K\}$ and L parameters $\theta \in R^L$. Our dataset D is a set of points $D = \{(y^1, x^1), (y^2, \boldsymbol{x}^2), ..., (y^N, \boldsymbol{x}^N)\}$ and we assume that each point has an error ϵ^k so $y^k = F(\boldsymbol{x}^k, \theta) + \epsilon^k$. This error is assumed to be an unbiased Gaussian error.

The Bayesian machine scientist assigns to each closed-form expression f_i a plausibility $p(f_i|D)$ given by:

$$p(f_i|D) = \frac{p(D|f_i)p(f_i)}{p(D)} \tag{3.2}$$

Because f_i has parameters associated that we do not know, we need to integrate the log-likelihood over all the parameters, so we have that:

$$p(f_i|D) = \frac{1}{p(D)} \int_{\Theta_i} p(D|f_i,\theta_i)p(f_i|\theta_i)p(f_i)d\theta_i = \frac{1}{p(D)}e^{-\mathcal{D}(f_i)}, \tag{3.3}$$

where the integral is over all the space Θ_i of possible values of parameters, and $p(f_i)$ is the prior over the expressions. $\mathcal{D}(f_i)$ is the description length that we defined in chapter 1.

$$\mathcal{D}(f_i) = \mathcal{D}_L(D|f_i) + \mathcal{D}_P(f_i), \tag{3.4}$$

where $\mathcal{D}_P(M)$ is (minus) the log-prior that is:

$$\mathcal{D}_P(f_i) \equiv -\log p(f_i) \tag{3.5}$$

and $\mathcal{D}_L(D)$ is the contribution to the description length of the integrated likelihood of the model:

$$\mathcal{D}_L(D|f_i) \equiv -\log \int_\Theta p(D|f_i,\theta)p(f_i|\theta)d\theta \tag{3.6}$$

In this case, $\mathcal{D}_L(D,f_i)$ is hard to compute because of the integral. For that reason, we make an approximation. In particular we assume that the likelihood $p(D|f_i,\theta_i)$ is peaked around the maximum θ_i^*. Using the Laplace approximation, we get that the log-likelihood can then be approximated as:

$$\mathcal{D}_L(D,f_i) \approx -\frac{B(D,f_i)}{2}, \tag{3.7}$$

where $B(D,f_i)$ is the Bayesian Information Criterion (BIC) that is defined as:

$$B(f_i) \equiv -2\log p(D|f_i,\theta^*) + L\log N \tag{3.8}$$

3.2.1 The Bayesian machine scientist naturally performs the two requirements of symbolic regression

We note that Eq. 3.3 is the posterior that we show in chapter 1 applied to the symbolic regression problem. That means that the best model will be that one that minimizes the description length $\mathcal{D}$. Because $\mathcal{D}$ is (minus) the sum of the logarithms of the prior and the integrated likelihood, this formalism allows us to fulfill the two requirements of symbolic regression to find a closed-form expression. On the one hand we have the integrated likelihood, that by definition tells us how likely is our model to generate our data, giving us a measure of fitness. On the other hand we have the prior, that encapsulates previous knowledge of the problem, including the fact that our model has to be simple.

To compute the likelihood, we assume that our observations are independent and that the errors ϵ_i are Gaussian around f_i. That means:

$$p(D|f_i,\theta_i) = \prod_{(y^k,\boldsymbol{x}^k)\in D} \frac{1}{s_y\sqrt{2\pi}} \int_{\Theta_i} e^{-\frac{\left(y^k - f_i^k(\boldsymbol{x};\theta)\right)^2}{2s_y^2}} d\theta_i, \tag{3.9}$$

where the best estimator of s_y^2 is the mean square error of $f_i(\boldsymbol{x})$. This error is different to the noise amplitude s_ϵ^2. To see the relationship between s_y^2 and s_ϵ^2 we need to take into account the discrepancy between f_i and the true model F, that we call it $\delta \equiv F - f_i$, and that $\epsilon^k = y^k - F(\boldsymbol{x}^k;\theta)$ we have that:

$$s_y^2 = \left\langle \delta_i^2 \right\rangle + \left\langle \epsilon_i^2 \right\rangle - 2\left\langle \delta_i \epsilon_i \right\rangle \tag{3.10}$$

For large N, we have that $\langle \delta_i \epsilon_i \rangle$ decreases as $\sqrt{N}$, so it is small compared to $\langle \delta_i^2 \rangle$ and $\langle \epsilon_i^2 \rangle$. Also, $\langle \epsilon_i^2 \rangle \approx s_\epsilon$, so:

$$s_y^2 = \left\langle \delta_i^2 \right\rangle + s_\epsilon^2 \tag{3.11}$$

Therefore we see that the observational error is the combination of two sources of errors (Ref. [12]): a random error and a systematic error. The random error is the one that you will always get when you repeat a measure. It is an error that is intrinsic to the measure and cannot be reduced. In our case is the noise of our samples s_ϵ. The systematic error is due to the imprecision of the model, and can be reduced by improving the model.

With regards of the prior, and in the spirit of Occam's razor, the prior has to catch the complexity of the models and assign a high plausibility to simple models. The way that the Bayesian machine scientist fulfills this condition is using a maximum entropy prior such that the statistical properties of models sampled from it are similar, to those in an empirical corpus of 4080 mathematical expressions extracted from Wikipedia.

To define the prior distribution over mathematical expressions, we took into account that mathematical expressions can be represented as network trees (Fig. 3.2). These trees are similar to the network trees from genetic programming, whose internal nodes are functions (operations) and whose leaves are variables and parameters. To model the distribution of trees we use an approach based on exponential random graph models (Ref. [28, 37, 35]), where we want to generate trees that preserves the statistical properties from the corpus. These statistical properties are the average number and the square of the number of mathematical operations in the expressions. The prior that fulfils these attributes is

$$p(f_i) = \sum_{o \in \mathcal{O}} \left[\alpha_o n_o(f_i) + \beta_o n_o^2(f_i) \right] \tag{3.12}$$

where the sum is over all the operations $o \in \mathcal{O}$, where $\mathcal{O} = \{\sin, \cos, +, *, ...\}$. α_o and β_o are hyperparameters that are fitted using least squares. $n_o(f_i)$ and $n_o^2(f_i)$ are the average number and the square of the number of mathematical operation o in the expression.

3.2.2 Sampling from the posterior distribution using MCMC

We have seen how the Bayesian machine scientist naturally looks for a closed-form expression that fits the data thanks to the likelihood and is as simple as possible thanks to the prior. Now we need to find the most plausible expression given our data, that is, the expression that maximizes $p(f_i|D)$. To do that we use a Markov chain Monte Carlo (MCMC) (Ref. [18]) procedure that allows us to explore the space of models and sample from $p(f_i|D)$.

We represent closed-form expressions as network trees where each internal node are functions or operations, and the leaves are parameters and variables. To explore the space of closed-form expressions we propose three different moves. These moves are accepted or rejected with different frequency, using

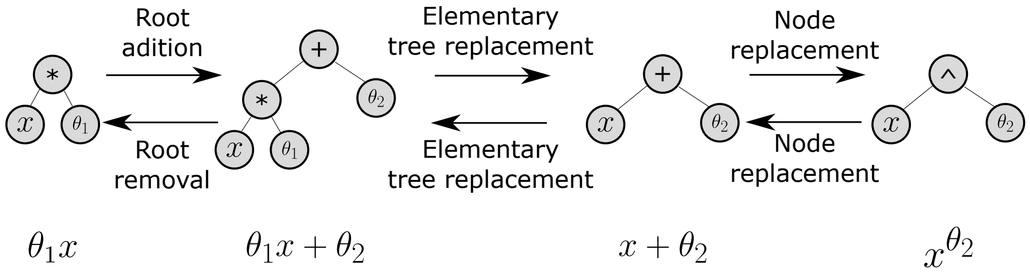

Figure 3.2: Network tree representation of closed-form expressions and MCMC moves. Here we represent one closed form expression and the movements that the Bayesian machine scientist uses to generate new expressions. The three movements are root addition/removal where a root is added/removed from the closed-form expression. In elementary tree replacement, an elementary tree (subtree with at least one operation) is substituted by another elementary tree. And in node replacement, a node that can contain a variable, a function or a parameter is replaced by another one.

the Metropolis-Hashting rule: root addition/removal, subtree replacement and node replacement (Fig. 3.2). The code used to perform can be found in a repository (see [60]).

3.3 Transition on the detectability of the models

Imagine that we want to find the true model that generated a dataset $D = \{(y^1, \boldsymbol{x}^1), (y^2, \boldsymbol{x}^2), ..., (y^N, \boldsymbol{x}^N)\}$. The question that we want to address is when is it possible to identify $F(\boldsymbol{x}^k, \theta)$ as a function of the noise amplitude and the number of observed points. On the one hand, with a finite number of points, it seems reasonable that when ϵ^k is low, the Bayesian machine scientist can identify $F(\boldsymbol{x}^k, \theta)$. On the other hand, when the noise is too large, it is not possible to find the true model and the Bayesian machine scientist will propose a simple model, which consists in a constant or a linear model, that are the most plausible a priori closed-form expression. Given these considerations, we want to study what happens in the middle, that is, when data can give us some clue about the true model despite the noise.

Specifically, we want to characterize the transition from the detectable region (characterized by low noise and informative data) to the undetectable region (that is, when data has too much noise the true model to be detected).

3.3.1 Synthetic data

To study this transition we make experiments with synthetic data that come from known closed-form expressions. The closed-form expressions that we use are selected from sampling the prior distribution explained in the last section and they are two-dimensional. Then we generate $N_D = 40$ datasets $D = \{(y^1, \boldsymbol{x}^1), (y^2, \boldsymbol{x}^2), ..., (y^N, \boldsymbol{x}^N)\}$ where $\boldsymbol{x} = (x_1, x_2)$ for each closed-form expression $y^k = F(\boldsymbol{x}^k, \theta) + \epsilon^k$, N is the number of points in the dataset and ϵ^k is a random number sampled from a normal distribution $\mathcal{N}(0, s_\epsilon)$. The closed-form expressions that we have chosen are:

$$F_1(x_1, x_2) = a_1(x_2 + b_1)x_1 \cos(x_1), \qquad x_1, x_2 \in [-2, 2] \tag{3.13}$$

$$F_2(x_1, x_2) = \frac{x_1^2 a_2^{x_1 x_2}}{x_1 + b_2}, \qquad x_1, x_2 \in [-2, 2] \tag{3.14}$$

Here a_i and b_i are the parameters from the expression F_i. Their values are: $a_1 = -1.1935$, $b_1 = -2.7828$, $a_2 = 1.2214$ and $b_2 = 3.1$. We can see that F_1 and F_2 are continuous functions in the domains that we have defined.

For each dataset we use the Bayesian machine scientist to sample from the posterior $p(f_i|D)$, and select the most plausible closed-form expression f_i. Then we compare the description length of the sampled closed-form expression $\mathcal{D}(f_i)$ with the description length of the real model $\mathcal{D}(F)$. If $\mathcal{D}(f_i) < \mathcal{D}(F)$, it means that the information that you need to encode the data and the model f_i is shorter than with F, so f_i is a better description of the data than F and that implies that the true model cannot be detected. For each level of noise amplitude s_ϵ and each number of points N, we can compute the fraction of datasets D for which the true model is detectable. We repeat this for different noise amplitudes and sizes and plot them in figures 3.3 a and c for expressions F_1 and F_2 respectively.

When the noise amplitude is low, the true model is always detected. When the noise amplitude is increased, the detectability remains almost at 100% until it reaches a certain noise amplitude, where it decreases suddenly to zero. As we can see in figures 3.3 a and c, the transition occurs at different noises amplitudes levels depending on the size of the dataset and the model itself. As the size of the dataset increases, the transition occurs at higher levels of noise amplitude.

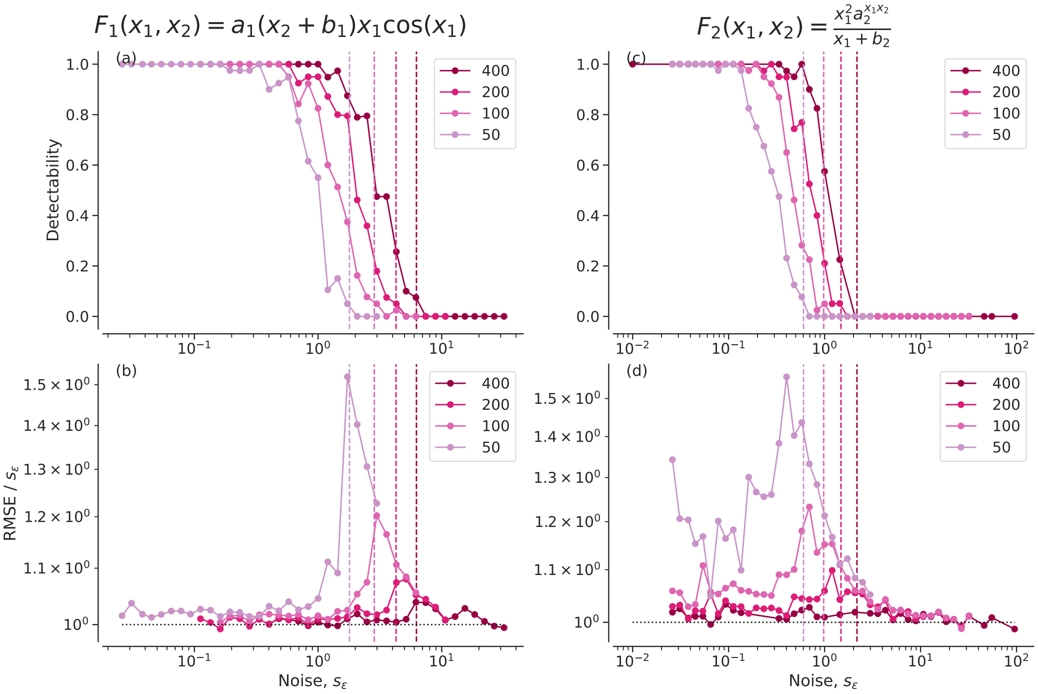

Figure 3.3: Detectability and relative mean square error of the predicted expressions. In panels a and c we show the detectability for different number of points of the expressions that we use to make our study. We perform one MCMC procedure to each of our 40 datasets. To each sample, we get the best model (that one that has the minimum description length), getting a total of 40 models. We compare their description length with the description length of the true model and compute the fraction of sampled models that have equal or greater description length than the true model (detectability). In panels b and d we show the root mean square error (RMSE) divided by the standard deviation of the noise amplitude s_ϵ. These means are computed over the 40 sampled models from the Bayesian machine scientist. The vertical line shows us where analytically the transition occurs (see Eq. 3.18).

3.3.2 The Bayesian machine scientist proposes simple models to reduce the observed error

The explanation of the transition, from a Bayesian point of view, is similar to the transition of the recommender system studied in chapter 2. As we said in the last section, the Bayesian machine scientist looks for the model that minimizes the description length $\mathcal{D}$, which in turn is the sum of the likelihood and prior's contribution to the description length ($\mathcal{D}_L$ and $\mathcal{D}_P$ respectively). $\mathcal{D}_L$ can be computed as:

$$\begin{aligned}
\mathcal{D}_L &= N \log\left(s_y\sqrt{2\pi}\right) - \frac{\sum_{(y^k,x^k)\in D}\left(y^k - f_i^k(\boldsymbol{x};\theta^*)\right)^2}{2s_y^2} - \frac{L+1}{2}\log N \\
&= \frac{N}{2}\log s_y^2 + \frac{\sum_{(y^k,\boldsymbol{x}^k)\in D}\left(y^k - f_i^k(x;\theta^*)\right)^2}{2s_y^2} + \frac{L+1}{2}\log N + \frac{N}{2}\log 2\pi \\
&= \frac{N}{2}\log s_y^2 + \frac{N}{2} - \frac{L+1}{2}\log N + \frac{N}{2}\log 2\pi, \qquad (3.15)
\end{aligned}$$

where s_y^2 is the standard deviation of a Gaussian, and its maximum likelihood estimator is the mean square error of f_i. Note that we add $+1$ next to the number of parameters. That is because when we fit the parameters of our model θ, we also fit s_y.

For low s_ϵ, the Bayesian machine scientist proposes expressions with longer description length, that is, more complex models. When the noise is increased to the transition region, the observed error starts to increase too (Fig. 3.3b, d). That is because the models proposed by the Bayesian machine scientist are similar in complexity, but implies an increase in δ. To reduce the description length the Bayesian machine scientist reduces the description length through the prior proposing simpler models with low contribution to the description length. When that happens, the error of the complex models is similar to the simpler models.

We can see this in more detail in figure 3.4 for expressions F_1 and F_2 where we show the $\mathcal{D}_P$ distributions. We can observe that for low noise amplitudes, the Bayesian machine scientist does not typically find an expression with a lower prior than that of the true model. In fact, they have a complexity similar to the true model. When we reach the transition, the Bayesian machine scientist starts to sample models that are less complex than the true model and some of them are constants or just one variable ($f = x_1$ or $f = x_2$).

For large noise amplitudes, the Bayesian machine scientist finds the simplest models for all the datasets.

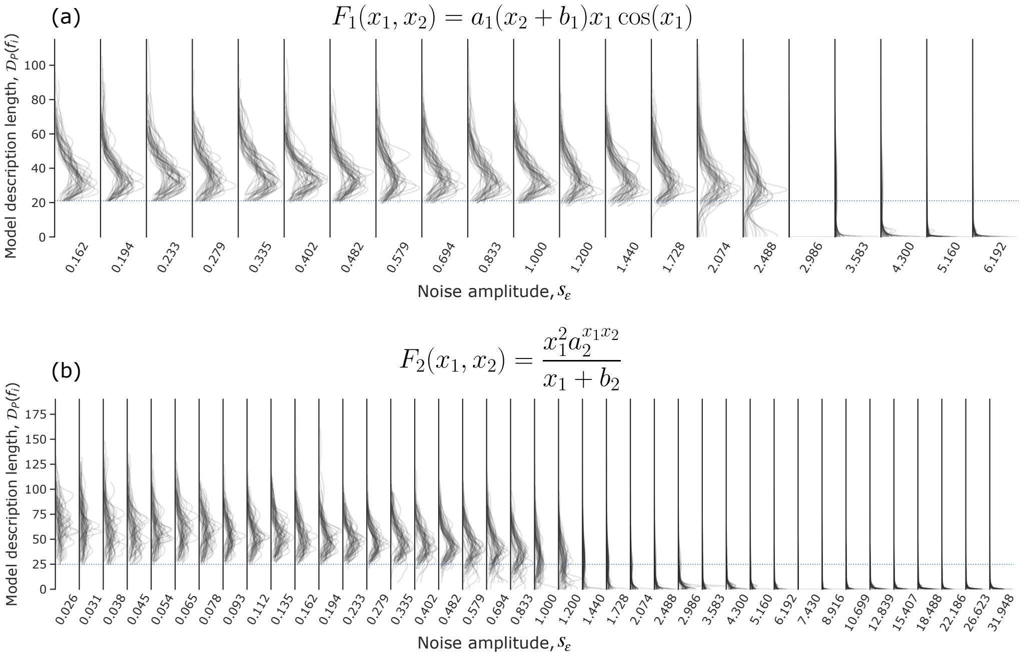

Figure 3.4: Distribution of the prior contributions of the description length for F_1 and F_2. Distribution of the prior contributions of the description length for F_1 (panel a) and F_2 (panel b). Each curve represents one MCMC sampling of the dataset of the Bayesian machine scientist. We plot, from the stationary state, the distribution of $\mathcal{D}_P$ of each repetition and for each noise level s_ϵ. The blue doted line represents the prior contribution to the description length of the true model $\mathcal{D}_P(F_i)$. The origin of $\mathcal{D}_P$ is set to the simplest model.

For low noise amplitudes, the Bayesian machine scientist tries to predict the true model with the contribution of the data and the prior, so our predicted models are both prior and data based. But for high noises, the data are dominated by the noise, so the data contribution is negligible and our predicted models are only prior based.

3.3.3 The description length gives us evidence about the transition

Until now we have been analyzed the transition as a function of the observation error s_ϵ. Now we characterize the transition in terms of the description length. From now we concentrate in the case of $N = 100$. In this study, we compute, the mean minimum description length and plot it in figure 3.5. We also compute, for each dataset, the description length of the true model and the description length of the simplest model, the constant model $f = cte$.

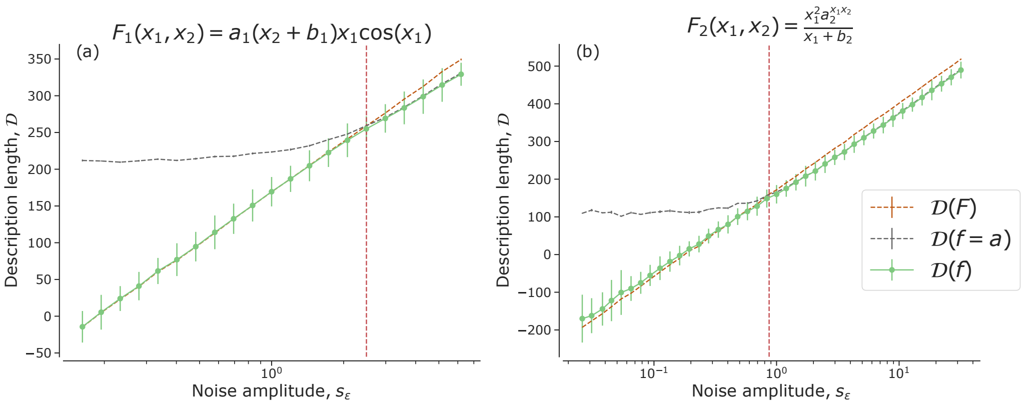

Figure 3.5: Description lengths as a function of the noise intensity. Description lengths for the simplest model $f = cte$ (grey curve), the true model F (green line) and the models with the lowest description lengths f_i founded by the Bayesian machine scientist for the case of $N = 100$. The constant that we put as the simplest model is the mean of y^k. Each point is the mean of the description length obtained from the 40 datasets with a confidence interval of 99.7%. The vertical line shows us where analytically the transition occurs (see Eq. 3.18).

We see that the description length of the minimum description length models that has been found, has two different behaviours: For low noises, the description length follows the same behaviour as the description length of the true model. In the transition, it changes its behaviour and it converges to the description length of the simplest model. That suggests that we can compute analytically where the transition happens, in this case, when the description lengths of the true model and the simplest model are the same. To do that, we recover the expression of $\mathcal{D}_L$ and put it in the description length, so:

$$\mathcal{D} = \frac{N}{2}\log s_y^2 + \frac{N}{2} - \frac{L+1}{2}\log N + \frac{N}{2}\log 2\pi - \log p \tag{3.16}$$

If we impose that $\mathcal{D}(f = cte) = \mathcal{D}(F)$, we get that:

$$\frac{s_y^2(F)}{s_y^2(f = cte)} = \left(N^{L-1}p^2(F)\right)^{\frac{1}{N}} \tag{3.17}$$

where for convenience, and without loss of generality, we impose that $\mathcal{D}_P$ of any model without operation is equal to zero.

Taking into account, the dependence of s_y^2 with the systematic error (Eq. 3.11), we have that we can rewrite equation 3.17 as:

$$s^2_{\epsilon,\text{transition}} = \frac{\langle\delta_i^2\rangle}{\left(N^{L-1}p^2(F)\right)^{-\frac{1}{N}} - 1} \tag{3.18}$$

In figures 3.3 and 3.5 we show a vertical line in the position where theoretically the transition occurs. We can see, that our predicted value of the transition is similar to the observed ones.

3.3.4 The study applied to a function with discontinuities of first kind with infinite jump

Until now, we have performed our study of the detectability to continuous functions that are continuous and well-behaved in the considered regions. Now we repeat the same study on a function with a discontinuity of first kind with an infinite jump. This discontinuities are characterized by the divergence to infinity of the limit of the function when the dependent variables tends to where the discontinuity is. The function that we are going to study is:

$$F_3(x_1, x_2) = \frac{a_3 x_2}{a_3 x_1 + b_3^{x_2} + x_2}, \quad x_1, x_2 \in [-2, 2] \tag{3.19}$$

F_3 has a discontinuity of first kind with infinite jump in the curve $\mathcal{C} : a_3x_1 + b_3^{x_2} + x_2 = 0$. In figure 3.6a we can see the detectability transition, and also with the same behaviour respect the size of the dataset. The difference between the continuous functions and the discontinuous cases is apparent from in figure 3.6b, where the observed error increase exponentially as the noise intensity decreases. The reason for that is because of the systematic error $\langle \delta^2 \rangle$ is too high because of the divergence in $\mathcal{C}$. That implies that with low noise, the error is too high, but at the same time, models that predict the rest of the function well are sampled. For higher noise, simplest models are optimal because we are only seeing noise. Also, we can see that the predicted value of $s_{\epsilon,\text{transition}}$ does not match with the observed one. That means that the mechanism of the transition is different in this case.

$$F_3(x_1, x_2) = \frac{a_3x_2}{a_3x_1 + b_3^{x_2} + x_2}$$

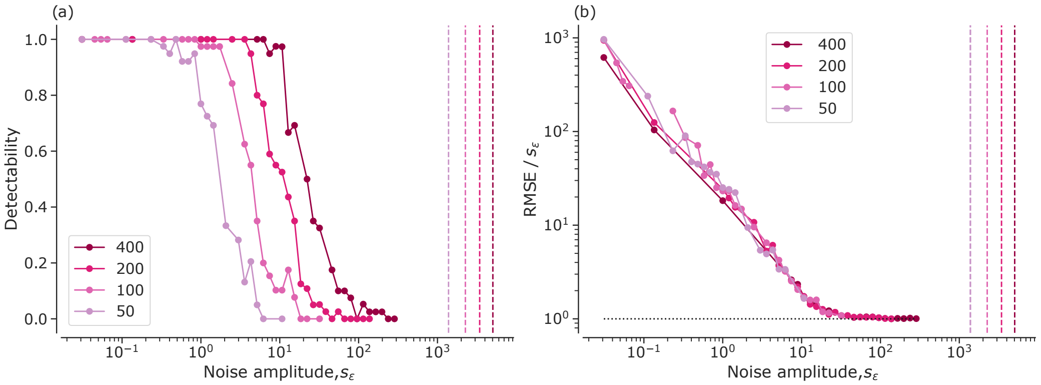

Figure 3.6: Detectability and relative mean square error of the predicted expressions. In panels a,c and e we show the detectability for different number of points of the expressions that we use to make our study. We perform 40 different MCMC procedures to sample 40 proposed models and compare their description length with the description length of the true model. We compute the fraction of sampled models that have equal or greater description length than the true model (detectability). In panels b,d,f we show the root mean square error (RMSE) divided by the standard deviation of the noise amplitude s_ϵ. These means are computed over the 40 sampled models from the Bayesian machine scientist. The vertical line shows us where analytically the transition occurs (see Eq. 3.18).

To see what happens with the prior, we studied the $\mathcal{D}_P$ distribution of F_3 (Fig. 3.7). In this case, for low noises we have some sampled models with lower complexity than F_3, similar than the transition regime for F_1 and F_2. When the relative RMSE is equal to one, almost all the distributions are picked at 0, meaning that we are in the detectability regime and all the proposed models are the simplest models.

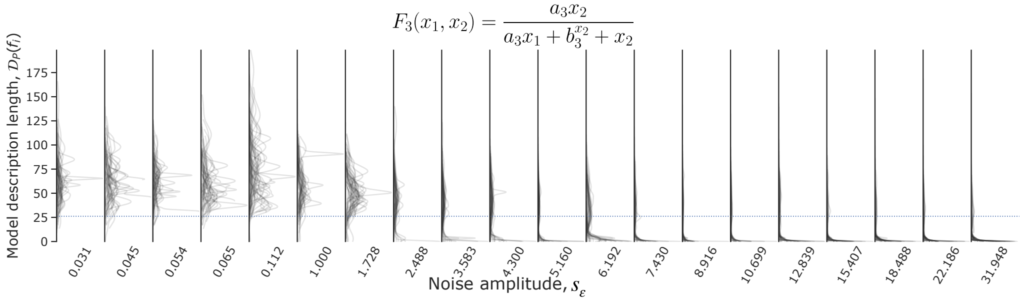

Figure 3.7: Distribution of the prior contributions of the description length for F_3. Distribution of the prior contributions of the description length for F_3. Each curve represents one MCMC procedure of the Bayesian machine scientist. We plot, from the stationary state, the distribution of $\mathcal{D}_P$ of each repetition and for each noise level s_ϵ. The blue doted line represents the prior contribution to the description length of the true model $\mathcal{D}_P(F_i)$. The origin of $\mathcal{D}_P$ is set to the simplest model.

And finally, we analyze the description length evolution as before (Fig. 3.8). We can see that before the transition, models better than the simplest model are found, but they are worse than the true model. The most interesting thing, but at the same time mysterious, happens after the transition. In this case better models than the true and simplest models are found. At the same time, the simplest model and the true model have the same amount of description length in the detectability regime. For this phenomena, we still do not have any explanation of why it happens.

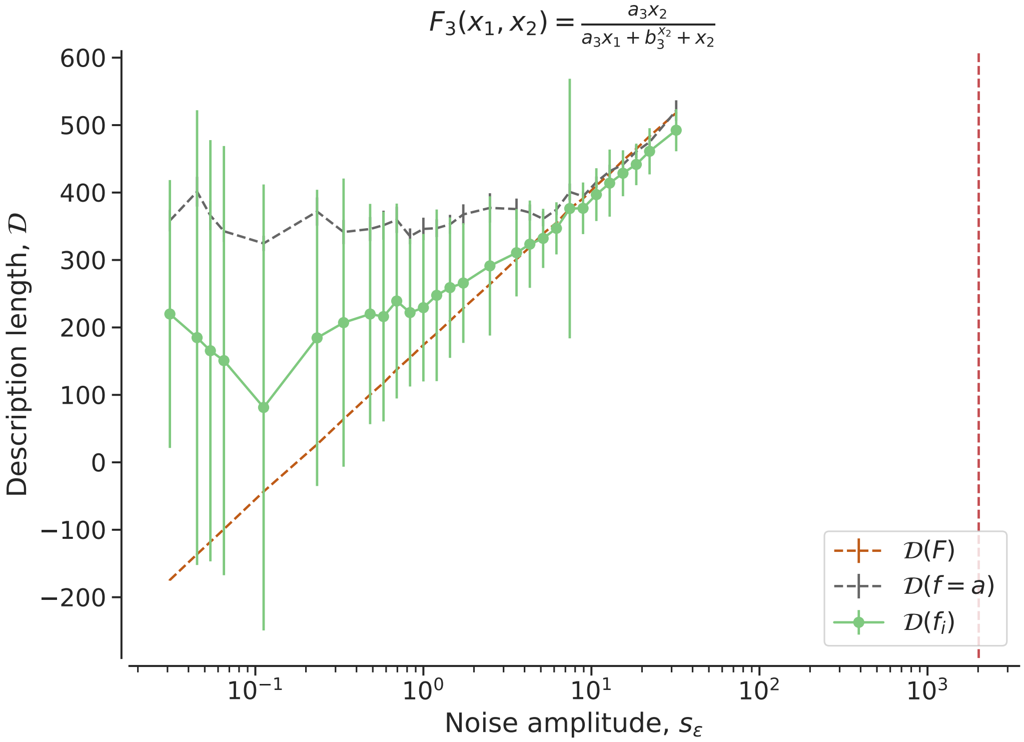

Figure 3.8: Description lengths as a function of the noise intensity for F_3. Description lengths for the simplest model $f = cte$ (grey curve), the true model F (green line) and the models with the lowest description lengths f_i founded by the Bayesian machine scientist for the case of $N = 100$. The constant that we put as the simplest model is the mean of y^k. Each point is the mean of the description length obtained from the 40 datasets with a confidence interval of 99.7%. The vertical line shows us where analytically the transition occurs (see Eq. 3.18).

3.4 Discussion

In this chapter we have discussed the role of the likelihood and the prior in a symbolic regression problem using the Bayesian machine scientist. We found a transition from a regime we can detect models similar to the true model, to another where simple solutions driven by the prior are found. This transition is controlled by the noise and the volume of the data that we have.

We also discovered that the continuity plays a roll in this study. When the true model is continuous in its domain, we can see a peak in the observed error. This is because complex functions start to not fit well and simple functions fit better after the transition. But if the function has a discontinuity of first kind with infinite jump, the error will be larger than the noise amplitude that generated the data. As the noise increase, the error decrease until because more simpler models are proposed until the undetectable regime is achieved. For this case, we have more questions to answer. We still don't have any answer to why the error decrease equally independently of the number of points. Also, despite we can find the transition point when the function is continuous only looking to the error, we cannot find it when the function is not continuous.

Chapter 4

Conclusions and perspectives

The principal objective of this thesis was to study the interplay between the likelihood and the prior in model selection problems. The importance of that relies in the fact that the likelihood has the information of how probable our data is given the proposed model, and the prior is a bias that we impose to favour models that we believe that are correct a priori. To show light of this interplay in complex situations we select a couple of problems: the recommender system using metadata and the symbolic regression. To make a better study, we use synthetic data because its versatility to control the experiments.

The recommender problem consists in predicting whether a user is going to like an item or not given information about the user preferences. What we want to study here is how extra information about the users and the items (metadata) can affect to the accuracy of our predictions. We used a probabilistic model, the Mixed-Membership Stochastic Block Model (MMSBM), and a Bayesian framework to add the metadata. We create datasets of ratings, with 200 users and 200 items and 400 ratings. We compute the accuracy our datasets for different levels of rating-metadata correlations and different values of the importance of metadata controlled by an hyperparameter. For low values of the hyperparameter, the MMSBM only sees the data, if it is equal to one, see all the data and metadata together and if it's large, it only see the metadata. So with this hyperparameter we can control the dominance of the prior (large values of the hyperparameter) over the likelihood (low values of the hyperparameter).

What we conclude was that data effects to the accuracy depending of the correlation and the amount of observed rating and observed metadata.

If metadata is highly correlated with metadata, we get that the accuracy improves when we are using metadata. If ratings are not correlated with metadata, the accuracy gets worse than not using metadata. Importantly, for metadata to have an effect, the amount of data and metadata needs to have similar contributions to the prior.

The second problem that we use to interplay of prior and likelihood is the symbolic regression problem. This problem consist in to find the best model through the space of mathematical closed-form expressions that that fits the data and is simple. Here what we want to study is how the noise of data can affect to the detectability of the true model that generated the data. We used the Bayesian machine scientist that looks for the models with the shortest description lengths. The role of measure the fitness of the dataset is given by the likelihood and the role of measure the simplicity of the model is given by the prior. To perform our study we generate 40 datasets for different noises levels and three known expressions. What we observed was an abrupt transition from a detectable regime where the Bayesian machine scientist could identify expressions with similar complexities than the true model, and a regime where the Bayesian machine scientist can not identify the true model and simpler expressions than the true model were proposed instead.

This transition is characterized by the noise amplitude and the number of points of the datasets. This transition takes place because, as we increase the noise amplitude, the observed error of the proposed model also is increased. The observed error is increased until the Bayesian machine scientist can not longer see the model and it only sees the noise. In this point when both, the true model and the simplest model, have the same observed error, the description length of the true model and the simplest model are equal. After that, the models that fit better the data are simple models that has a large prior. That is translated into a reduction of the prior contribution of the description length. This argument is true except for non continuous functions with a discontinuity of first kind with infinite jump. In that case, the observed error is high for low level noises but the algorithm still proposes models with similar complexity. But in the indetectability regime, the Bayesian machine scientist proposes simple models. The explanations of this phenomena is because the increase of the discrepancy of the sampled models with the true model in the divergence. One possible solution might be the increment of number of points in the divergence.

To conclude this book, we have seen how the interplay of the data and the prior determines the type of the predicting models and effects to the quality of our predictions. In the case of the recommender system we have a data regime versus a metadata (represented by the prior) regime and both interact through the accuracy. In the case of symbolic regression, in the detectability regime we have that both, likelihood (the data) and the prior have a role in the inference process. But for high noise amplitudes, the Bayesian machine scientist can only predicts the noise proposing simple models with the largest prior.

We think that this work can help to better understand the Bayesian inference process from the point of view of the balance of the data and the prior.

Appendix A

k-fold cross-validation

In science, is very important to make good predictions. When we train a dataset of size N, it is important to the model that we are training has the lowest error possible, but also, it is important that we can make predictions to unseen. If our predictions have a low error given our dataset but do not predict unseen data, we say that our model is overfitted. In figure A.1 we can see that using a 9th degree polynomial we reduce the error to 0 because is the best curve that fits nine points, but if we make predictions beyond the observed range, we are going to make huge errors.

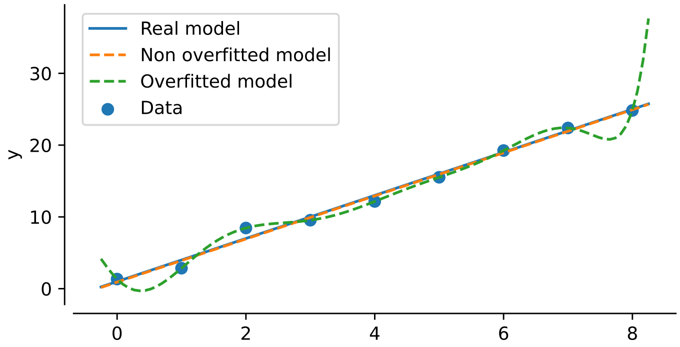

Figure A.1: Representation of overfitting using polynomial regression. In blue we can see the real model, the dots represent the nine points data with noise and the dashed lines are adjusted models. We use polynomial regression to find the dashed models, in orange we use a 1st degree polynomial and in green a 9th degree polynomial.

Overfitting usually happens because our model is to complex that can only predict our dataset as we can see in figure A.1. One way to avoid overfitting, is splitting our dataset in two: a training set that usually have at least a 60% of the dataset, and a test set that contains the rest. The training set is the subset of the dataset that we use to train our model, and the test set is used to validate the trained model.

A common method used to split the data is k-fold cross validation (Ref. [14, 51]). Here we split the dataset in k equally and random distributed small datasets. Then we choose one of these splits as a test set and the rest as a training set (Fig. A.2).

Later we train our model and get the estimator given this selection. After that, we change the training set and repeat until we get k estimators of what we want to measure (Fig. A.2). The final measured estimator will be the mean of every measure of the estimator given the different datasets. One advantage of the k-fold cross validation is the reduction of the variability in the estimator, gaining more generality in our measures.

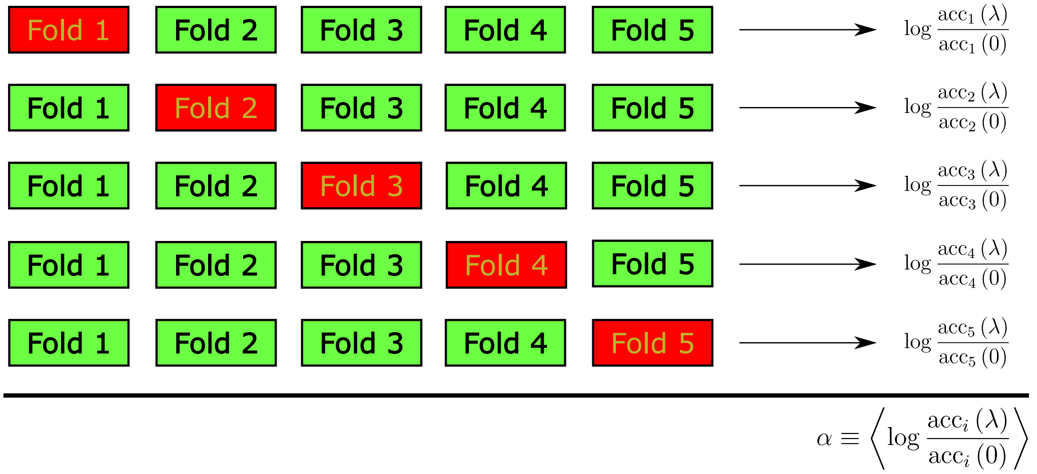
Fold 1
Fold 2
Fold 3
Fold 4
Fold 5
$\log \frac{\mathrm{acc}_1(\lambda)}{\mathrm{acc}_1(0)}$
$\log \frac{\mathrm{acc}_2(\lambda)}{\mathrm{acc}_2(0)}$
$\log \frac{\mathrm{acc}_3(\lambda)}{\mathrm{acc}_3(0)}$
$\log \frac{\mathrm{acc}_4(\lambda)}{\mathrm{acc}_4(0)}$
$\log \frac{\mathrm{acc}_5(\lambda)}{\mathrm{acc}_5(0)}$
$\alpha \equiv \left\langle \log \frac{\mathrm{acc}_i(\lambda)}{\mathrm{acc}_i(0)} \right\rangle$

Figure A.2: Representation of how k-fold cross validation works. Representation of how k-fold cross validation works for $k = 5$. In green the training set used for training the algorithm. In red we have the test set to verify the trained model with unseen data. We can see that for each test set we compute an estimator, in this case the log-fold change, to later compute the relative improvement α.

Appendix B

Expectation-maximization algorithm

To obtain a maximum of the posterior we start by berating random initial conditions for each model parameter $\boldsymbol{\theta}, \boldsymbol{\eta}, \boldsymbol{p}, \boldsymbol{\zeta}, \boldsymbol{q}, \hat{\boldsymbol{q}}$.

Then we perform iteratively two steps until the model parameters convergence:

1. Expectation step: compute the auxiliary functions $\omega_{ij}(\alpha, \beta)$, $\sigma^{k}_{i\ell_k}(\alpha)$, and $\hat{\sigma}^{k}_{ig}(\alpha, \gamma)$ using current values for $\boldsymbol{\theta}, \boldsymbol{\eta}, \boldsymbol{p}, \boldsymbol{\zeta}, \boldsymbol{q}, \hat{\boldsymbol{q}}$ using Eqs. 2.18, 2.19 and 2.20.

2. Maximization step: Compute the new values for the model parameters using the values for the auxiliary functions and Eqs. 2.24, 2.28, 2.32, 2.36, 2.40 and 2.44.

Because the posterior landscape is very rugged, to make predictions we perform the EM algorithm 10 times and consider all of the models to estimate the average probability that user i rates item j with rating r (see [47]) as follows:

$$\langle p(r_{ij} = r | R^O, A^O_k) \rangle \approx \frac{1}{N} \sum_{n=1}^{N} p_n(r_{ij} = r | R^O, A^O_k, (\dots)) \tag{B.1}$$

where $(\dots) = \{\boldsymbol{\theta}, \boldsymbol{\eta}, \boldsymbol{p}, \boldsymbol{\zeta}, \boldsymbol{q}, \hat{\boldsymbol{q}}\}$, and $p_n(r_{ij} = r | R^O, A^O_k, (\dots))$ is the probability that user i rates item j with rating r in run n of the EM algorithm.

Bibliography

[1] J. L. W. V. Jensen. "Sur les fonctions convexes et les inégalités entre les valeurs moyennes". In: *Acta Mathematica 1906 30:1* 30.1 (Dec. 1906), pp. 175–193. ISSN: 1871-2509. DOI: 10.1007/BF02418571. URL: https://link.springer.com/article/10.1007/BF02418571.

[2] E T Jaynes. "Information theory and statistical mechanics. {II}". In: *Phys. Rev.* 108.2 (Oct. 1957), pp. 171–190. DOI: 10.1103/PhysRev.108.171.

[3] E T Jaynes. "Information theory and statistical mechanics". In: *Phys. Rev.* 106.4 (May 1957), pp. 620–630. DOI: 10.1103/PhysRev.106.620.

[4] Rolf Sundberg. "Maximum Likelihood Theory for Incomplete Data from an Exponential Family". In: *Scandinavian Journal of Statistics* 1.2 (1974), pp. 49–58.

[5] G. Salton, A. Wong, and C. S. Yang. "A Vector Space Model for Automatic Indexing". In: *Communications of the ACM* 18.11 (Nov. 1975), pp. 613–620. DOI: 10.1145/361219.361220.

[6] A P Dempster, N M Laird, and D B Rubin. "Maximum Likelihood from Incomplete Data via the EM Algorithm". In: *Source: Journal of the Royal Statistical Society. Series B (Methodological)* 39.1 (1977), pp. 1–38. URL: https://www.jstor.org/stable/2984875.

[7] Richard A REDNERf and Homer F Walker. "MIXTURE DENSITIES, MAXIMUM LIKELIHOOD AND THE EM ALGORITHM*". In: *SIAM REVIEW* 26.2 (1984).

[8] Kerson Huang. *Statistical Mechanics*. 2nd ed. Hoboken, NJ: John Wiley \& Sons, 1987.

[9] H Eugene Stanley. *Introduction to Phase Transitions and Critical Phenomena*. Oxford, UK: Oxford University Press, 1987.

[10] G H Hardy, J E Littlewood, and G Pólya. "Inequalities (Cambridge Mathematical Library)". In: (1988). URL: https://books.google.com/books/about/Inequalities.html?hl=es&id=t1RCSP8YKt8C.

[11] Nigel D Goldenfeld. *Lectures on \protect{P}hase \protect{T}ransitions and the \protect{R}enormalisation \protect{G}roup*. Reading, MA: Addison-Wesley, 1992.

[12] John R Taylor. *An Introduction to Error Analysis*. University Science Books, 1997.

[13] Robin Van Meteren and Maarten Van Someren. "Using Content-Based Filtering for Recommendation". In: *ECML/MLNET Workshop on Machine Learning and the New Information Age* (2000), pp. 47–56. ISSN: 15506606. URL: http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.25.5743&amp;rep=rep1&amp;type=pdf.

[14] Trevor. Hastie, Robert. Tibshirani, and J. H. (Jerome H.) Friedman. *The elements of statistical learning : data mining, inference, and prediction : with 200 full-color illustrations*. Springer, 2001, p. 533. ISBN: 9780387952840.

[15] Badrul Sarwar et al. "Item-based collaborative filtering recommendation algorithms". In: *Proceedings of the 10th international conference on World Wide Web*. WWW '01. New York, NY, USA: ACM, 2001, pp. 285–295. DOI: 10.1145/371920.372071.

[16] R Albert and A.-L. Barabási. "Statistical mechanics of complex networks". In: *Rev. Mod. Phys.* 74 (2002), pp. 47–97.

[17] Robin Burke. "Hybrid Recommender Systems: Survey and Experiments". In: *User Modeling and User-Adapted Interaction 2002 12:4* 12.4 (2002), pp. 331–370. ISSN: 1573-1391. DOI: 10.1023/A:1021240730564. URL: https://link.springer.com/article/10.1023/A:1021240730564.

[18] Olle Häggström. "Finite Markov Chains and Algorithmic Applications (London Mathematical Society Student Texts)". In: (2002), p. 126. URL: https://books.google.com/books/about/Finite_Markov_Chains_and_Algorithmic_App.html?hl=es&id=hpLxIJ9LwRgC.

[19] E T Jaynes. *Probability Theory: The Logic of Science*. Cambridge University Press, 2003.

[20] John R. Koza. *Genetic Programming : On the Programming of Computers By Means of Natural Selection Complex Adaptive Systems.* 2003, pp. 1–609. ISBN: 0262111705. URL: `papers2://publication/uuid/5DADD85F-EE2F-42E1-8BF8-2CC6959C4FA0`.

[21] Jonathan L. Herlocker et al. "Evaluating collaborative filtering recommender systems". In: *ACM Transactions on Information Systems* 22.1 (2004), pp. 5–53. ISSN: 10468188. DOI: `10.1145/963770.963772`.

[22] C Tallberg. "A {B}ayesian approach to modeling stochastic blockstructures with covariates". In: *J. Math. Sociol.* 29.1 (2004), pp. 1–23.

[23] G Adomavicius and A Tuzhilin. "Towards the next generation of recommender systems: {A} survey of the state-of-the-art and possible extensions". In: *IEEE T. Knowl. Data En.* 17 (2005), pp. 734–749.

[24] Laurent Candillier, Frank Meyer, and Marc Boullé. "Comparing State-of-the-Art Collaborative Filtering Systems". In: *Machine Learning and Data Mining in Pattern Recognition.* Ed. by Petra Perner. Berlin, Heidelberg: Springer Berlin Heidelberg, 2007, pp. 548–562. ISBN: 978-3-540-73499-4.

[25] S Džeroski and L Todorovski, eds. *Computational Discovery of Scientific Knowledge.* Lecture Notes in Artificial Intelligence. Springer, 2007.

[26] Gerd Gigerenzer et al. "Helping doctors and patients make sense of health statistics". In: *Psychological Science in the Public Interest, Supplement* 8.2 (2007), pp. 53–96. ISSN: 15291006. DOI: `10.1111/j.1539-6053.2008.00033.x`.

[27] P D Grünwald. *The Minimum Description Length Principle.* Cambridge, Massachusetts: The MIT Press, 2007.

[28] Garry Robins et al. "An introduction to exponential random graph (p*) models for social networks". In: *Soc. Netw.* 29.2 (2007), pp. 173–191. DOI: `10.1016/j.socnet.2006.08.002`.

[29] J. Ben Schafer et al. "Collaborative Filtering Recommender Systems". In: *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)* 4321 LNCS (2007), pp. 291–324. DOI: `10.1007/978-3-540-72079-9{\_}9`. URL: `https://link.springer.com/chapter/10.1007/978-3-540-72079-9_9`.

[30] Tian Chen and Liang He. "Collaborative Filtering Based on Demographic Attribute Vector". In: *Future Computer and Communication, International Conference on* (June 2009), pp. 225–229. DOI: 10.1109/FCC.2009.68.

[31] M Schmidt and H Lipson. "Distilling free-form natural laws from experimental data". In: *Science* 324.5923 (2009), pp. 81–85. ISSN: 1095-9203. DOI: 10.1126/science.1165893.

[32] Xiaoyuan Su and Taghi M Khoshgoftaar. "A survey of collaborative filtering techniques". In: *Adv. in Artif. Intell.* 2009 (2009), 4:2–4:2. ISSN: 1687-7470. DOI: 10.1155/2009/421425.

[33] A S Waugh et al. "Party polarization in {C}ongress: {A} network science approach". In: *arXiv: Physics and Society* (2009). URL: http://arxiv.org/abs/0907.3509.

[34] James Evans and Andrey Rzhetsky. "Machine science". In: *Science* 329.5990 (2010), pp. 399–400.

[35] A Caimo and N Friel. "Bayesian inference for exponential random graphs". In: *Soc. Netw.* 33.1 (2011), pp. 41–55.

[36] Trent Mcconaghy. *Genetic Programming Theory and Practice IX*. Ed. by Rick Riolo, Ekaterina Vladislavleva, and Jason H. Moore. Genetic and Evolutionary Computation. New York, NY: Springer New York, 2011. ISBN: 978-1-4614-1769-9. DOI: 10.1007/978-1-4614-1770-5. URL: http://link.springer.com/10.1007/978-1-4614-1770-5.

[37] Tom A B Snijders. "Statistical models for social networks". In: *Ann. Rev. Sociol.* 37.1 (2011), pp. 131–153.

[38] D Barber. *Bayesian reasoning and machine learning.* Cambridge University Press, 2012.

[39] Thomas M. Cover and Joy A. Thomas. "Elements of Information Theory (Google eBook)". In: 2012 (2012), p. 776. URL: https://books.google.com/books/about/Elements_of_Information_Theory.html?hl=es&id=VWq5GG6ycxMC.

[40] J Bobadilla et al. "Recommender systems survey". In: *Knowledge-Based Syst.* 46 (2013), pp. 109–132.

[41] J. Bobadilla et al. “Recommender systems survey”. In: *Knowledge-Based Systems* 46 (2013), pp. 109–132. ISSN: 09507051. DOI: 10.1016/j.knosys.2013.03.012. URL: http://dx.doi.org/10.1016/j.knosys.2013.03.012.

[42] J Yang, J McAuley, and J Leskovec. “Community Detection in Networks with Node Attributes”. In: *2013 IEEE 13th International Conference on Data Mining*. 2013, pp. 1151–1156. DOI: 10.1109/ICDM.2013.167.

[43] Marco de Gemmis et al. “Semantics-Aware Content-Based Recommender Systems”. In: *Recommender Systems Handbook, Second Edition* (Jan. 2015), pp. 119–159. DOI: 10.1007/978-1-4899-7637-6{_}4. URL: https://link.springer.com/chapter/10.1007/978-1-4899-7637-6_4.

[44] F M Harper and J A Konstan. “The {M}ovieLens Datasets: History and Context”. In: *ACM Trans. Interact. Intell. Syst.* 5.4 (2015). ISSN: 2160-6455.

[45] Albert-László Barabási. “Network Science”. In: *Network Science* (2016), 474 pages. URL: https://books.google.com/books/about/Network_Science.html?hl=es&id=iLtGDQAAQBAJ.

[46] Steven L Brunton, Joshua L Proctor, and J Nathan Kutz. “Discovering governing equations from data by sparse identification of nonlinear dynamical systems”. In: *Proc. Natl. Acad. Sci. U.S.A.* 113.15 (2016), pp. 3932–3937.

[47] Antonia Godoy-Lorite et al. “Accurate and scalable social recommendation using mixed-membership stochastic block models”. In: *Proceedings of the National Academy of Sciences* 113.50 (2016), pp. 14207–14212. ISSN: 0027-8424. DOI: 10.1073/pnas.1606316113. URL: http://www.pnas.org/lookup/doi/10.1073/pnas.1606316113.

[48] D Hric, T P Peixoto, and S Fortunato. “Network Structure, Metadata, and the Prediction of Missing Nodes and Annotations”. In: *Phys. Rev. X* 6.3 (Sept. 2016), p. 31038. DOI: 10.1103/PhysRevX.6.031038.

[49] M E J Newman and A Clauset. “Structure and inference in annotated networks”. In: *Nat. Comm.* 7 (2016), p. 11863.

[50] A White and T B Murphy. “Mixed-membership of experts stochastic blockmodel”. In: *Netw. Sci.* 4.1 (2016), pp. 48–80.

[51] Gareth James et al. *An Introduction to Statistical Learning*. Springer Texts in Statistics. New York, NY: Springer US, 2017. DOI: 10.1007/978-1-0716-1418-1. URL: https://link.springer.com/10.1007/978-1-0716-1418-1.

[52] N. M. Mangan et al. "Model selection for dynamical systems via sparse regression and information criteria". In: *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences* 473.2204 (Aug. 2017), p. 20170009. ISSN: 1364-5021. DOI: 10.1098/rspa.2017.0009. URL: https://royalsocietypublishing.org/doi/10.1098/rspa.2017.0009.

[53] Leto Peel, Daniel B Larremore, and Aaron Clauset. "The ground truth about metadata and community detection in networks". In: *Sci. Adv.* 3.5 (2017). DOI: 10.1126/sciadv.1602548.

[54] Samuel H. Rudy et al. "Data-driven discovery of partial differential equations". In: *Science Advances* 3.4 (Apr. 2017), e1602614. ISSN: 2375-2548. DOI: 10.1126/sciadv.1602614. URL: https://advances.sciencemag.org/lookup/doi/10.1126/sciadv.1602614.

[55] Z T Wilson and N V Sahinidis. "The {ALAMO} approach to machine learning". In: *Comput. Chem. Eng.* 106 (2017), pp. 785–795. DOI: 10.1016/j.compchemeng.2017.02.010.

[56] S. Cobo-López et al. "Optimal prediction of decisions and model selection in social dilemmas using block models". In: *EPJ Data Science* 7.1 (2018). DOI: 10.1140/epjds/s13688-018-0175-3.

[57] Patryk Orzechowski, William La Cava, and Jason H Moore. "Where Are We Now? A Large Benchmark Study of Recent Symbolic Regression Methods". In: *Proceedings of the Genetic and Evolutionary Computation Conference*. GECCO '18. New York, NY, USA: Association for Computing Machinery, 2018, pp. 1183–1190. ISBN: 9781450356183. DOI: 10.1145/3205455.3205539. URL: https://doi-org.sabidi.urv.cat/10.1145/3205455.3205539.

[58] T Vallès-Català et al. "Consistencies and inconsistencies between model selection and link prediction in networks". In: *Phys. Rev. E* 97 (2018), p. 62316. DOI: 10.1103/PhysRevE.97.062316.

[59] Antonia Godoy-Lorite, Roger Guimerà, and Marta Sales-Pardo. “Network-Based Models for Social Recommender Systems”. In: *Business and Consumer Analytics: New Ideas* (2019), pp. 491–512. DOI: 10.1007/978-3-030-06222-4{_}11.

[60] Roger Guimerà. *Bayesian Machine Scientist*. 2019. URL: https://bitbucket.org/rguimera/machine-scientist/src/no_degeneracy/.

[61] N Stanley, T Bonacci, and R Kwitt. “Stochastic block models with multiple continuous attributes”. In: *Appl. Netw. Sci.* 4 (2019), p. 54.

[62] M Tarrés-Deulofeu et al. “Tensorial and bipartite block models for link prediction in layered networks and temporal networks”. In: *Phys. Rev. E* 99.3 (2019), p. 32307.

[63] M Contisciani, E A Power, and C De Bacco. “Community detection with node attributes in multilayer networks”. In: *Sci. Rep.* 10.1 (2020), pp. 1–16.

[64] Oscar Fajardo-Fontiveros. *Multipartite MMSBM*. 2020. URL: https://github.com/oscarcapote/Multipartite_MMSBM.

[65] Roger Guimerà et al. “A Bayesian machine scientist to aid in the solution of challenging scientific problems”. In: *Science Advances* 6.5 (2020). ISSN: 23752548. DOI: 10.1126/sciadv.aav6971.

[66] Ignasi Reichardt et al. “Bayesian Machine Scientist to Compare Data Collapses for the Nikuradse Dataset”. In: *Phys. Rev. Lett.* 124.8 (Feb. 2020), p. 84503. DOI: 10.1103/PhysRevLett.124.084503. URL: https://link.aps.org/doi/10.1103/PhysRevLett.124.084503.

[67] Silviu-Marian Udrescu and Max Tegmark. “AI Feynman: A physics-inspired method for symbolic regression”. In: *Science Advances* 6.16 (Apr. 2020). DOI: 10.1126/SCIADV.AAY2631. URL: https://www.science.org/doi/abs/10.1126/sciadv.aay2631.

[68] Oriol Artime and Manlio De Domenico. “Percolation on feature-enriched interconnected systems.” eng. In: *Nature communications* 12.1 (Apr. 2021), p. 2478. ISSN: 2041-1723 (Electronic). DOI: 10.1038/s41467-021-22721-z.

[69] Gaël Poux-Médard et al. “Complex decision-making strategies in a stock market experiment explained as the combination of few simple strategies”. In: (2021). DOI: 10.1140/epjds/s13688-021-00280-z. URL: https://doi.org/10.1140/epjds/s13688-021-00280-z.

[70] Xabier Rodríguez-Martínez, Enrique Pascual-San-José, and Mariano CampoQuiles. “Accelerating organic solar cell material’s discovery: high-throughput screening and big dat”. In: *Energy \& Environmental Science* 14 (2021), pp. 3301–3322.

[71] Charu C. Aggarwal. “Recommender systems : the textbook”. In: (), p. 498. URL: https://books.google.com/books/about/Recommender_Systems.html?hl=es&id=GKjWCwAAQBAJ.

[72] Hee Deuk et al. “A literature review and classification of recommender systems research”. In: (). DOI: 10.1016/j.eswa.2012.02.038.

[73] David Z. Morris. *Netflix says Geography, Age, and Gender Are ‘Garbage’ for Predicting Taste.* URL: https://fortune.com/2016/03/27/netflix-predicts-taste/.

UNIVERSITAT ROVIRA i VIRGILI