Empirical signatures of universality, hierarchy and clustering in culture

Babeanu, A.I.

Chapter 2

Evidence for mixed rationalities in preference formation

Understanding the mechanisms underlying the formation of cultural traits is an open challenge. This is intimately connected to cultural dynamics, which has been the focus of a variety of quantitative models. Recent studies have emphasized the importance of connecting those models to empirically accessible snapshots of cultural dynamics. In particular, it has been suggested that empirical cultural states, which differ systematically from randomized counterparts, exhibit properties that are universally present. Hence, a question about the mechanism responsible for the observed patterns naturally arises. This study proposes a stochastic structural model for generating cultural states that retain those robust empirical properties. One ingredient of the model assumes that every individual's set of traits is partly dictated by one of several universal "rationalities," informally postulated by several social science theories. The second, new ingredient assumes that, apart from a dominant rationality, each individual also has a certain exposure to the other rationalities. It is shown that both ingredients are required for reproducing the empirical regularities. This suggests that the effects of cultural dynamics in the real world can be described as an interplay of multiple, mixing rationalities, providing indirect evidence for the class of social science theories postulating such a mixing. The model should be seen as a static, effective description of culture, while a dynamical, more fundamental description is left for future research.

This chapter is based on the following scientific article:
A. I. Băbeanu and D. Garlaschelli, *Complexity*, Article ID 3615474 (2018).

2.1 Introduction

A solid theoretical understanding of how preferences form is currently lacking. There is little doubt that preferences, opinions, values and beliefs, which are generically referred to as "cultural traits", are dynamical entities, and that interpersonal social influence plays an important role in driving their dynamics, among other factors. Moreover, a complete theoretical understanding should account for the fact that the dynamics of traits takes place in parallel along multiple dimensions, namely that opinions and preferences can develop in relation to multiple topics or aspects of life. Along these lines, various dynamical models been developed and studied [1], such as the Axelrod model [2], which is very representative for studies of multidimensional dynamics, commonly referred to as "cultural dynamics", in contrast to studies of unidimensional dynamics, commonly referred to as "opinion dynamics". Various studies of cultural dynamics extending the Axelrod model can be found in the literature [3, 4, 5, 6, 7, 8, 9, 10, 11]. Recent studies [12, 13, 14] (Chap. 1) have shown that models of cultural dynamics are sensitive to the initial conditions, namely to how the initial vectors of agents' traits are chosen: initial cultural states constructed from empirical data show systematic deviations from their shuffled and random counterparts. In fact, Ref. [14] (Chap. 1) argues that such deviations point towards universal structural properties inherent in empirical cultural states. More insights about the formation of cultural traits should be achievable by studying these states, since they can be regarded as partial snapshots of cultural dynamics in the real world.

The universal properties mentioned above are expressed in terms of the effects the empirical cultural state has on social influence models whose initial conditions are specified by this state – here, a "cultural state" is a set of cultural vectors (SCV), where each cultural vector encodes the sequence of cultural traits associated to one agent in the model. On one hand, an Axelrod-type model [2] of (multi-dimensional) cultural dynamics is used to evaluate the propensity of the cultural state to long-term cultural diversity (LTCD). On the other hand, a Count-Bouchaud-type model [15] of (one-dimensional) opinion dynamics is used to evaluate the propensity of the cultural state to short-term collective behavior (STCB). Both measures are functions of a common parameter ω, controlling for the range of social influence in cultural space, which allows for an LTCD-STCB correspondence to be drawn for a given cultural state. It turns out that an empirical cultural state generally induces an LTCD-STCB curve that is close to the second diagonal ($\mathrm{LTCD}(\omega) \approx 1 - \mathrm{STCB}(\omega), \forall\omega$), while exhibiting, for a given STCB value, higher LTCD values than a trait-shuffled cultural state, which in turn exhibits higher LTCD values than a randomly generated counterpart [12, 14] (also see Chap 1). These results seem universal [14] (Chap. 1), namely independent of the data set used for constructing the cultural vectors composing the empirical cultural state, suggesting that real-world cultural dynamics is governed by universal laws. Moreover, as argued in Ref. [14] (Chap. 1), this type of analysis suggests that inter-agent social influence, the essential ingredient of cultural

dynamics models, is insufficient for explaining the observed structure. Although it is meaningful to incorporate additional ingredients into social influence models, while attempting to give rise to empirical-like structure in a dynamical setting, this study does not aim for that. Instead, it aims at providing an effective, phenomenological, static description of the observed structure, which should provide additional insights before developing a more fundamental, dynamical description.

The purpose of this study is to develop a structural stochastic model that would generate realistic cultural states, while incorporating plausible ingredients from social science. Specifically, these states should retain the universal properties inherent to empirical cultural states that are observed in Ref. [14] (Chap. 1). In fact, Ref. [13] has already investigated various ways of generating sets of cultural vectors in random, but non-uniform ways. A method that appeared particularly promising relied on the notion of "cultural prototypes": a few underlying, abstract sequences of logically compatible, self-enforcing cultural traits, which govern the way the generated vectors are distributed in cultural space. According to the method, each cultural vector is partly a copy of one of the prototypes and partly random. The implicit claim is that each cultural prototype is induced by one of a few (3 to 5) fundamental and universal "principles of social life", or "rationalities", that would strongly affect any process of trait formation in any social system. Such entities are postulated, under different names and in slightly different numbers, by several theoretical frameworks in social science [16, 17, 18, 19, 20]. The exact number of such entities depends on the exact theory that is considered, as different theories are built on somewhat different arguments and pieces of evidence. It is important that the number is larger than 1 but not too large, while independent of system size. From a natural science perspective, such ideas are attractive, since they exhibit a certain reductionist tendency of trying to understand the observed socio-cultural variability in terms of combinations of a few, elementary and universal building blocks. Various parallels and similarities between these theories are discussed in the literature [21, 22, 23]. For the purpose of the current study, all these theories are equivalent. Still, for creating an instructive and compact context, the discussion is restricted to one of them, namely to Plural Rationality Theory, chosen for reasons discussed in Sec. 2.5.

Plural Rationality Theory (PRT), also referred to as "(Grid-Group) Cultural Theory" [16], is a qualitative description of socio-cultural structure and dynamics as an interplay between a small number of irreducible "ways of life", or "rationalities". These ways of life are understood as abstract, "elementary building blocks" of societies and are supposedly recognizable regardless of the geographical context, of the historical context or of the scale of the system that is studied. It is believed that the ways of life go along with different perceptions of risk [24, 25] and, interestingly, that they always coexist, although either of them is often dominant for a given period of time, for a given (part of) the system that one studies[1]. Such

[1]It may be useful to think of the ways of life as being the elements of a complete, orthogonal basis of some abstract vector space. One may then associate a vector in this space to a certain

ideas appear compatible with recent empirical findings concerning the existence of a small number of behavioural phenotypes in dyadic games [26]. In PRT, each way of life is understood as a self-enforcing combination of a "pattern of (social) relations" and a "cultural bias". On one hand, a pattern of relations is often understood as a tendency of organizing the social ties between people in a certain way, thus a connectivity pattern in the social graph. On the other hand, a cultural bias is a combination of preferences, opinions, values and beliefs that are compatible with each other and with the associated pattern of relations. By comparison to the definitions in Ref. [14] (Chap. 1), one can easily realize that a cultural bias can be thought of as a point or a region in "cultural space" that is representative for the respective "way of life". A cultural bias is formally represented here by the notion of "cultural prototype", previously used in Ref. [13].

This notion is at the core of two stochastic, structural models of culture that are defined and studied here. The first model, called "Prototype Generation" (PG), postulates that each cultural vector is partly a copy of one of the k prototypes and partly random. This generation method is similar to the "Prototype Evolution" method of Ref. [13], although with small technical differences. The second model, called "Mixed Prototype Generation" (MPG), postulates that each cultural vector is an asymmetric mixture (or combination) of all the prototypes. From the perspective of PRT, this "mixing" is a formal realization of the idea that every person combines the ways of life in a unique way, such that preferences and opinions related to different aspects of life – cultural traits of different cultural features (or variables) – are due to the "influence" of different cultural biases, though at any given moment in time one cultural bias is usually dominating. In the literature concerned with PRT and the other, similar, theories, this mixing aspect often goes under the name of "the multiple self", and was not implemented in Ref. [13]. The importance of mixing for correctly interpreting (and testing) PRT has been already stressed on [25], while the general importance of multiple selves for social science has also been extensively discussed [27]. Moreover, research on preferences in economic contexts also suggests that the multiple self is important [28, 29, 30]. On the other hand, research in cross-cultural psychology appears to be divided: some studies seem to ignore the multiple self [31], while others seem to acknowledge it [32, 33]. This study provides further insights on this matter, by directly comparing the PG and MPG models with each other and with empirical data,

Sec. 2.2 explains the models in detail, while Sec. 2.3 describes how the free parameters are tuned, as to reproduce some lower-order properties of one empirical cultural state. Cultural states generated with the two models are then evaluated, in Sec. 2.4, by means of the LTCD-STCB analysis of Refs. [12, 14] (Chap. 1). It is shown that cultural states generated by PG are structurally dissimilar to

part of a certain socio-cultural system, at a given moment in time. It is not clear to what extent such vectors would be related to the cultural vectors used in this study. This is only a semi-formal analogy that is not exploited further here, nor in any other study so far, to the extent that the authors are aware of.

the empirical ones, as they do not exhibit the universal LTCD-STCB behavior, after tuning the free parameters to empirical data in terms of simpler, but meaningful quantities. On the other hand, cultural states generated with MPG are structurally similar to the empirical ones, as they reproduce the universal LTCD-STCB behavior, after applying an analogous tuning procedure. This suggests that the mixing, multiple self ingredient is crucial for describing the effects of preference formation in terms of cultural prototypes, and that MPG should be regarded as the successful model. Sec. 2.5 further discusses the results, their limitations, as well as extensions of this work and questions that are worth investigating in the future. The manuscript is concluded in Sec. 2.6.

2.2 Model description

This section describes the two stochastic models of culture: the Prototype Generation (PG) model and the Mixed Prototype Generation (MPG) model, which are used below for generating sets of cultural vectors (SCVs) that can be quantitatively studied with the LTCD-STCB tool, previously applied to empirical SCVs in Refs. [12, 14] (Chap. 1). Both models rely on the concept of cultural prototype introduced above.

An SCV can be visualized as a table of cultural traits, where the columns correspond to cultural vectors (or sequences) and the rows correspond to cultural features (or variables). If the SCV is constructed from empirical data, the columns correspond to real people that are sampled by a social survey, while the rows correspond to questions that are asked in the social survey. This is illustrated by Fig. 2.1, which is explained in detail below. Consistently with Ref. [14] (Chap. 1), a "cultural space" is the set of all possible cultural vectors (or combinations of traits) allowed by the given set of cultural features: one combination of traits is one point in this discrete space. For the purpose of this work, the general set-up is restricted to cultural spaces defined in terms of features that are exclusively nominal. In this setting, distances between points in the cultural space are given by Eq. (2.5) of Sec. 2.3. Disregarding ordinal features makes the modeling paradigm compatible with the (arguably strong) assumption that one prototype corresponds to one point in cultural space, meaning that a prototype picks up one and only one trait of any given feature. Other limitations of this assumptions are extensively discussed in Sec. 2.5, together with possible ways of relaxing it, for the purpose of generalizing the current modeling paradigm in future work.

The two models are schematically illustrated in Fig. 2.1. The figure first shows a sketch of an empirical SCV, where the rows correspond to cultural features, the columns correspond to cultural vectors and the letters correspond to cultural traits – the n'th row shows the traits of the N agents that are expressed (or formulated) with respect to the n'th feature. Then, it shows a set of 3 cultural prototypes (their number could have been different), in 3 different colors, all of them spanning over all features (or questions) relevant for the empirical set of vectors. Finally,

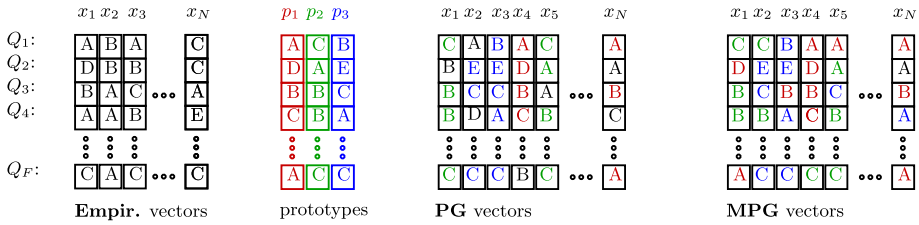

Figure 2.1: Schematic illustration of the two stochastic models, showing (from left to right): an empirical SCV with N vectors (x_1 to x_N) and F nominal variables (Q_1 to Q_F); a set of $k = 3$ cultural prototypes for the same F variables; a SCV with N vectors generated, from the prototypes, using the PG model; a SCV with N vectors generated, from the same prototypes, using the MPG model. For the PG and MPG sketches, red, green and blue denote the copies of cultural traits from one of the first, second and third prototype respectively, while black denotes the explicitly random generation of traits.

it illustrates a typical set of vectors generated using the PG method, followed by one generated using the MPG method. The colors distinguish between the prototypes, while indicating how the traits are copied from the prototypes to the cultural vectors, while black denotes traits that generated in an explicitly random way (uniform distribution, independently of the prototypes).

There are several things worth noting in relation to Fig. 2.1. First, the possibility that two or more prototypes pick the same trait for a certain feature is allowed by the current modeling paradigm (note that any of the traits that can be copied from one of the prototypes can also be generated via explicit randomness). This is essential for controlling the average prototype-prototype distance, as will become apparent below. Second, a PG vector is partly copied from one prototype and partly generated in an explicitly random way, while a MPG vector is a mixture of copies from all the prototypes, with one dominating prototype and with few traits generated in an explicitly random way. Third, both models make use of another type of randomness, in addition to the explicitly random trait generation and to the randomness involved in generating the prototypes. This randomness has to do with assigning every trait of every vector to a "prototype of origin", once the random generation fraction and the influence fractions of the prototypes are specified. In the case of MPG, it is mainly this trait-assignment randomness that allows for the generation of a multitude of distinct cultural vectors from a small set of fixed prototypes, in the presence of little explicitly random trait generation.

The procedure for generating the cultural prototypes is the same for both the PG and the MPG models. One needs to specify the number of prototypes k, as well as the value of another parameter $\alpha \in (0, 1)$, which controls for the expected cultural distance between the prototypes. This parameter governs the expected

number of overlaps (or coincidences) between prototypes in terms of how they are distributed over the traits of a specific feature. In the extreme case of $\alpha \to 1$, all prototypes pick the same trait for every feature, yielding the smallest possible separation between the prototypes in cultural space (which coincides with the minimum of 0 allowed by the cultural distance definition in Eq. (2.5)). In the other extreme case of $\alpha \to 0$, the prototypes are distributed as uniformly as possible over the traits of every feature, yielding the largest possible separation between the prototypes in cultural space (which only coincides with the maximum of 1 allowed by Eq. (2.5) if the number of traits q is larger or equal to the number of prototypes k for every feature). This is achieved by a formulation in terms of the set of integer partitions I_k^q describing the possible ways of distributing the k prototypes over the q traits of a certain feature. The α parameter actually controls the probability distribution over the set I_k^q, via the "compactness" of the integer partitions in this set. Sec. 2.A.2 precisely describes how these probabilities are assigned and how the set I_k^q is computationally generated in the first place, for any combination of k and q. Once the prototypes are chosen, everything else is conditional on them, for both models.

According to the **Prototype Generation (PG)** model, each cultural vector is a partial realization of one of the prototypes. Each of the N cultural vectors is generated by copying a random sequence of traits from one of the k prototypes, while generating the other traits in a uniformly random way – choosing the prototype is done randomly for every vector. Then, a subset of the F features of length round$(\beta \cdot F)$ is randomly and independently selected for each vector and the traits of these features are copied from the prototype to the vector. Here, "round" returns the integer that is closest to its argument, while $\beta \in [0, 1]$ is a third model parameter, in addition to k and α (which are already needed for the purpose of specifying the prototypes, in the manner described above). The β parameter specifies the fraction of traits that are directly copied from the prototype, thus controlling for the expected distance between a vector and its prototype. The traits for the remaining features are generated randomly and independently, according to uniform feature-level probability distributions – the explicit random generation mentioned above. Thus, β also controls for the amount of explicitly random generation of traits. The PG method effectively specifies that there are k "classes" of cultural vectors and those of a certain class are located at a certain, β-controlled average distance from the associated cultural prototype. This is similar to the "Prototype Evolution" method of Ref. [13], although there are small differences in how exactly the vectors are generated in the two cases. Moreover, the method of Ref. [13] did not allow for controlling the expected cultural distance between the prototypes.

According to the **Mixed Prototype Generation (MPG)** model, each cultural vector is a combination of all prototypes, although an unbalanced combination, meaning that the numbers of traits copied from the different prototypes are deliberately unequal. The extent of this discrepancy is explicitly controlled via the third model parameter, which, like for PG, is called β. Although the exact

definition and usage of the $\beta \in (0,1)$ parameter is different in MPG than in PG, its role is quite similar. Specifically, also in the context of MPG, β (indirectly) controls for the fraction of traits copied from the dominating prototype to the vector: more traits are copied from the dominating prototype if the discrepancy between the prototypes is higher. In addition to traits copied from the prototypes, there are traits that are generated in an explicitly random way, but in a small number. For each generated vector, this number is by construction not higher than the number of traits copied from the lowest-contributing prototype. Consequently, if there are k prototypes, the number of traits generated via explicit randomness does not exceed $F/(k+1)$. Thus, $1/(k+1)$ is an upper bound for the fraction of explicit randomness in an entire set of cultural vectors generated with MPG. It is also important to note that, like for PG, this fraction is controlled by β and that the upper bound is reached when β is in the limit of minimal imbalance. The limited usage of explicitly random trait generation by MPG means that cultural vectors are more strongly constrained by the prototypes, compared to PG. Still, MPG allows for generating a large variety of possible cultural vectors, since the k prototypes can mix in many different ways.

The MPG model needs a procedure of specifying, for each generated vector, the k values of the numbers of traits that are to be copied from the k prototypes, along with the number associated to explicitly random generation. These $k+1$ positive, integer numbers should add up to F and have their discrepancy controlled by the β parameter. Moreover, there is no reason to believe that the sequence of numbers associated to one β value should be the same across all generated vectors, so randomness should be involved in choosing these numbers. Therefore, the model needs a probabilistic way of drawing $k+1$ random, positive integers $\{t_1(\beta), .., t_{k+1}(\beta)\}$ satisfying $\sum_{l=1}^{k+1} t_l(\beta) = F$, such that their expected discrepancy is controlled via a single parameter β. The procedure chosen for this purpose is described below.

This procedure heavily relies on isometrically mapping the discrete $\{0, 1, .., F\}$ set of integers to the $[0, 1]$ interval of the real axis. For each generated vector, the latter interval is split into $k+1$ parts, by performing "cuts" in k randomly chosen points. In this manner, a sequence of $k+1$ preliminary weights $\{W_1, ..., W_{k+1}\}$, subject to $\sum_{l=1}^{k+1} W_l = 1$ is numerically obtained. These weights are obviously independent of β and have a fixed expected discrepancy. A β-dependent transformation (explained below) is applied on the preliminary weights $\{W_1, ..., W_{k+1}\}$, thus providing a sequence of β-dependent weights $\{w_1(\beta), ..., w_{k+1}(\beta)\}$ satisfying $\sum_{l=1}^{k+1} w_l(\beta) = 1$, with expected discrepancy controlled by β. Finally, the sequence of β-dependent weights is converted back to the desired sequence $\{t_1(\beta), .., t_{k+1}(\beta)\}$. This final operation is non-trivial, requiring a self-consistent, joint rounding procedure, which is generally difficult to choose, since one cannot generally ensure that $w_l = \text{round}(t_l/F), \forall l$ – a non-trivial problem of weight discretization. Here, a simple, pragmatic choice is made: converting the lowest k weights to the closest, lower integer, while converting the highest weight to the

integer needed for satisfying the summation constraint – this ensures that the highest weight, which should correspond to the dominating prototype, is converted to the highest integer.

The only aspect of MPG remaining to be explained is how the β-dependent weights $\{w_1(\beta), ..., w_{k+1}(\beta)\}$ are obtained from the preliminary weights. This is done by raising the latter to a common power $p(\beta)$ and then normalizing:

$$w_l(\beta) = \frac{(W_l)^{p(\beta)}}{\sum_{l'=1}^{k+1}(W_{l'})^{p(\beta)}}, \tag{2.1}$$

where the common power $p(\beta) \in (0, +\infty)$ controls for the average discrepancy between these weights and maps to $\beta \in (0, 1)$ via:

$$p(\beta) = \tan\left(\beta\frac{\pi}{2}\right), \tag{2.2}$$

where the tangent is a convenient choice of a smooth, continuous function, with the appropriate domain and range. Thus, a value $\beta > 0.5$ implies a value $p > 1$ and a higher discrepancy of $\{w_1^p, ..., w_{k+1}^p\}$ than that of $\{W_1, ..., W_{k+1}\}$, while a value $\beta < 0.5$ implies a value $p < 1$ and a lower discrepancy of $\{w_1^p, ..., w_{k+1}^p\}$ than that of $\{W_1, ..., W_{k+1}\}$.

Before describing the fitting and the outcomes of the PG and MPG models, it is worth summarizing a few important aspects. Both models rely on the notion of cultural prototypes, which is currently formalized in a simplistic manner, which is only sensible for cultural spaces defined exclusively in terms of nominal features. The procedure for generating the prototypes is the same for both models and relies on two parameters, k and α, which specify, respectively, the number of prototypes and the expected distance between them. The differences between PG and MPG consist in how the cultural vectors are generated conditionally on the prototypes: for PG, every vector is in part a copy from one of the prototypes and in part explicitly random; for MPG, every vector is an imbalanced mixture of all prototypes and explicitly random to a much lower extent, which is how the "multiple-self" ingredient is implemented. Nonetheless, in both cases, there is a third model parameter, β, which governs, in different ways, the lengths of the randomly selected subsets of features whose traits that are copied from the prototypes. In both cases, β effectively controls for the expected distance between a vector and its (dominating) prototype, as well as for the fraction of explicit randomness.

2.3 Model fitting

Before applying the LTCD-STCB analysis on SCVs generated with either the PG or MPG models, it is useful to somehow constrain some of the free model parameters. This is done in terms of statistical quantities simpler than the LTCD

and the STCB measures, that can be evaluated on both empirical SCVs and on the model SCVs. On the empirical side, the quantities are averaged over several, empirical SCVs constructed by randomly selecting $N = 500$ cultural vectors from the 13000 available ones in Eurobarometer data set [34], while restricting to the nominal features – let "(EBM_n)" stand for the nominal part of the Eurobarometer data set. The empirical data is formatted according to the procedure explained in Ref. [14] (Chap. 1). On the model side, these quantities are averaged over many SCVs, of the same size N, that are realizable in the cultural space of ($\mathrm{EBM_n}$), for the given combination of parameters – the prototypes are independently generated upon creating every model SCV.

The two simple quantities in terms of which the models are tuned to empirical data are the average and the standard deviation of the inter-vector distances in the SCV, which are here denoted by "AIVD" and "SIVD" respectively:

$$\mathrm{AIVD} = \frac{2}{N(N-1)} \sum_{i<j} d_{ij}, \tag{2.3}$$

$$\mathrm{SIVD} = \sqrt{\frac{2}{N(N-1)} \sum_{i<j} (d_{ij} - \mathrm{AIVD})^2}, \tag{2.4}$$

where N is the number of cultural vectors and d_{ij} is the cultural distance, as defined and used in Refs. [14, 13, 12] (and Chap. 1). The notation $i < j$ denotes that the respective summation is carried out over all distinct pairs (i, j). In the case of a fully-nominal cultural space, with which this study is dealing, d_{ij} reduces to the Hamming distance between the two sequences of symbols encoding cultural vectors i and j:

$$d_{ij} = 1 - \frac{1}{F} \sum_{l=1}^{F} \delta(x_i^l, x_j^l) = \frac{1}{F} \sum_{l=1}^{F} d_{ij}^l, \tag{2.5}$$

with, d_{ij} taking values within the $[0, 1]$ interval. Here, l iterates over the F nominal features, x_i^l, x_j^l are the traits of vectors i and j with respect to feature l and δ stands for the Kroneker-Delta function. The second equality shows that the cultural distance can be expressed as an average over feature-level contributions, which becomes useful below. Previous work has shown that an empirical SCV is characterized by a lower AIVD than its random counterpart and a higher SIVD than both its random and shuffled counterparts [12, 13]. The AIVD and SIVD quantities, which incorporate pairwise distance information, are conceptually different than what is often used in the context of cultural dynamics and of the Axelrod model, namely the size of the largest connected component, which can be regarded as an overall measure of similarity. Instead, the latter is somewhat similar to the STCB quantity explained and used in Sec. 2.4.

It is instructive to see that the expressions of AIVD and SIVD can be rewritten

in the following way:

$$\text{AIVD} = \frac{1}{F}\sum_{l=1}^{F}\frac{2}{N(N-1)}\sum_{i<j}d_{ij}^{l}, \tag{2.6}$$

$$\text{SIVD} = \sqrt{\frac{1}{F^2}\sum_{l,l'=1}^{F}\left[\frac{2}{N(N-1)}\sum_{i<j}d_{ij}^{l}d_{ij}^{l'} - \frac{4}{N^2(N-1)^2}\sum_{i<j}d_{ij}^{l}\sum_{i'<j'}d_{i'j'}^{l'}\right]}, \tag{2.7}$$

by using a feature-level cultural distance d_{ij}^{l} introduced via Eq. (2.5) – the transition from (2.4) to (2.7) was suggested by the SI of Ref [12].

Note that the AIVD can be understood as an average over feature-level AIVD contributions, which are represented by the expression within the l-summation of Eq. (2.6). It can be checked that the (nominal) feature-level AIVD contribution is a measure of how uniformly the N vectors are distributed over the possible traits of that feature. This is more obvious when expressing the expected value of the AIVD contribution in terms of probabilities associated to the traits, which is shown in Eq. (2.8) below. Thus, for an empirical SCV containing only nominal features, the AIVD is a measure of average uniformity of the empirical frequency distributions associated to the features. Consequently, the AIVD is also a measure of how subjective the questions/topics associated to the features are on average – when the frequencies of possible answers are more similar to each other, there is less justification to talk about a "better", a "more correct" or a "more agreed upon" answer, so the question is inherently more subjective.

Also note that, in Eq. (2.7), the quantity inside the average over pairs of features (k,l) is the covariance between features k and l, defined in terms of the feature-level cultural distances. Given that this quantity is averaged over all possible pairs of features and that the square-root is a monotonous function, the SIVD encodes information about the pairwise correlations between features, although in a somewhat indirect way.

For both models, the choice made here is that of:

- tuning the α parameter in terms of the AIVD quantity (Eqs. (2.3), (2.6)), for any combination of values of the β and k parameters;
- tuning the β parameter in terms of the SIVD quantity (Eqs. (2.4), (2.7)), for any value of the k parameter, based on the previous fitting of α in terms of AIVD;
- simply repeating the tuning (and the LTCD-STCB analysis in Sec. 2.4) for several values of k.

This implies that, for every value of k, the tuning (or fitting) is done at two levels: the α-AIVD level and the β-SIVD level, the former being nested into the

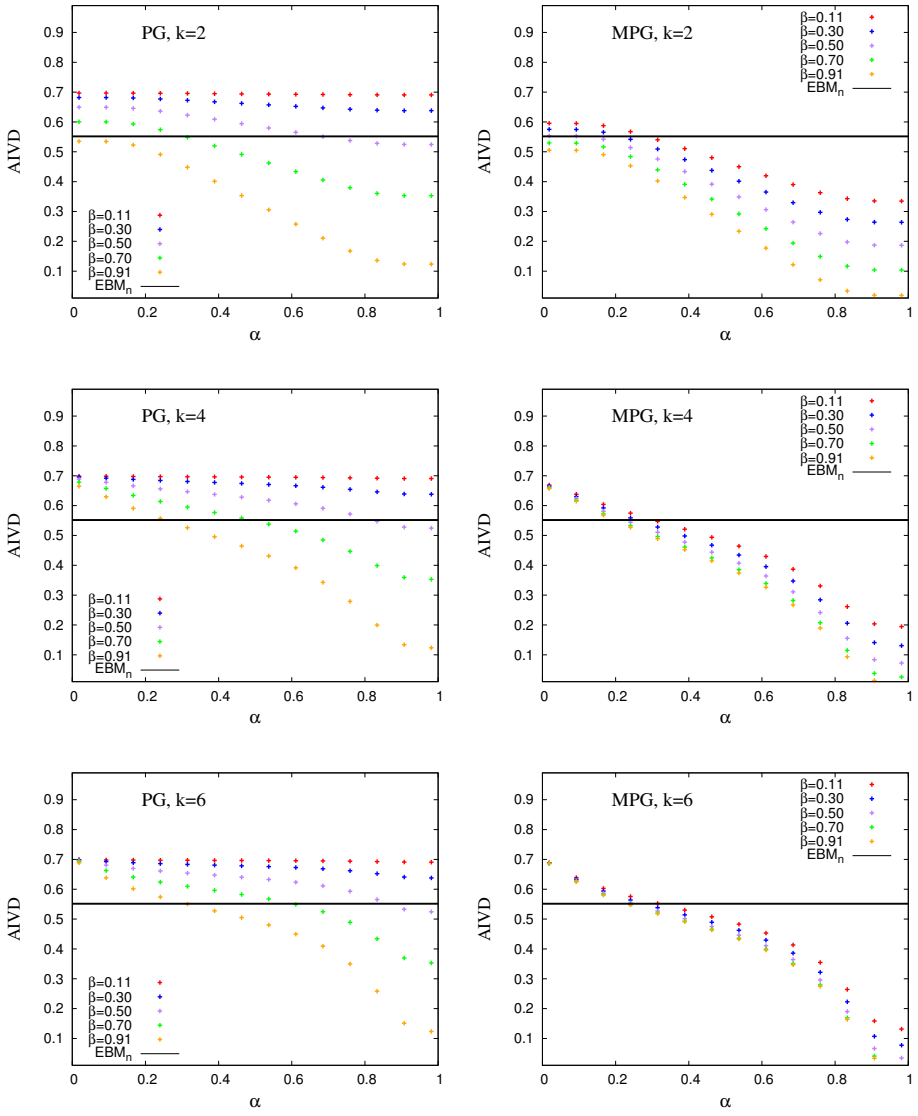

Figure 2.2: Dependence on model AIVD on the α parameter, for several values of the β parameter (legend), for $k = 2$ (top), $k = 4$ (center) and $k = 6$ (bottom) prototypes, for the PG (left) and MPG (right) models. The horizontal lines show the empirical AIVD uncertainty range (one standard error on each side of the mean).

latter. In practice, the fitting is carried out automatically, using a nested, 2-levels algorithm that relies on a modified bisection-type method for each level. The algorithm is precisely described in Sec. 2.C. In order to work, this approach relies on the assumption that there is one, unique solution for the fitting problem, for every value of k. This uniqueness is demonstrated via Figs. 2.2 and 2.3, which are also used for providing a general intuition of how the fitting works and of how the AIVD and SIVD quantities depend on α, β and k, for the two models.

Before entering this description, it is worth mentioning that the computer time for the fitting algorithm is greatly reduced by being able to evaluate the average (model) AIVD quantity analytically, in a manner that properly accounts for all SCVs that can be generated for any combination of k, α and β. While the calculation is described in detail in Sec. 2.B, a schematic understanding can already be provided here. The essential ingredient of the calculation is a simple, exact formula for the expected AIVD contribution of one feature of range q:

$$\langle \mathrm{AIVD}\left(\{p_1, ..., p_q\}\right)\rangle = 1 - \sum_{i=1}^{q} p_i^2, \tag{2.8}$$

which assumes that the probabilities of its traits $\{p_1, ..., p_q\}$ are all known – see Sec. 2.B for the proof. For a discrete probability distribution, Eq. (2.8) is a measure of uniformity very similar to the Shannon entropy. Conditional on a specific choice of the prototypes, this set of probabilities (thus the feature-level probability distribution) is fully determined by the integer partition describing how the prototypes are distributed over the traits and by the fraction of traits that are randomly generated, the latter being controlled by β. In this context, Eq. (2.8) already assumes that an averaging is performed over SCVs generated from the same set of prototypes. One still needs to perform an average of this expression over integer partitions (Eq. (2.20) of Appendix Sec. 2.B), according to the probability distribution controlled by α (Eqs. (2.12) and (2.13) of Appendix Sec. 2.A.1), followed by another average over all features (Eq. (2.19) of Appendix Sec. 2.B), since different features will in general have different ranges q. At a superficial inspection, using a similar approach for analytically computing the SIVD quantity appears very complicated, if at all possible. Numerical calculations are instead employed for computing the (model) SIVD.

Fig. 2.2 deals with the first-level fitting. It shows the dependence of the analytically computed AIVD quantity (see above) on the α parameter, for several β values, for several k values and for both the PG and MPG models. Moreover, it shows the empirical AIVD uncertainty range[2] via the horizontal bands in the six panels. Thus, a solution of the first-level fitting is indicated by an intersection between a model curve of a given combination of k and β and the horizontal band. Note that, for either of the two models and for any combination of k and β, if a solution exists, this solution is actually unique. In order to understand

[2] An uncertainty range, as defined in Sec. 2.C, is the interval spanned by one standard mean error on each side of the mean.

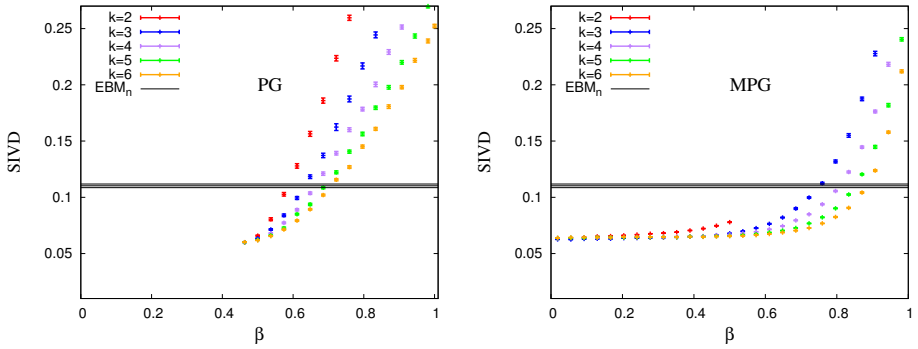

Figure 2.3: Dependence of model SIVD on the β parameter, for several values of the number of prototypes k (legend), for the PG (left) and MPG (right) models, where the α parameter is tuned such that the empirical AIVD is reproduced. The error bars of the points show the numerical uncertainty ranges, while the horizontal lines show the empirical SIVD uncertainty range (one standard error on each side of the mean).

the behavior implicit in Fig. 2.2, which is explained below, one should keep in mind that AIVD measures the average uniformity of the feature-level probability distributions.

First, it is worth focusing on the AIVD dependence on the α and β parameters. Note, on one hand, that for a given combination of k and β, the AIVD generally decreases with α, or at least remains constant. This is due to the fact that the AIVD decreases with decreasing distance between prototypes, thus with increasing α. For PG, this decrease is stronger for higher β values, since for low β value the uniformity is anyway high, because of the large fraction of randomly generated traits. For MPG, this β-dependence of the decrease is not that strong, since the fraction of randomly generated traits cannot exceed $1/(k+1)$. On the other hand, for a given combination of k and α, the AIVD generally decreases with increasing β. This is due to the fact that the AIVD decreases with decreasing fraction of randomly generated traits, thus with increasing β.

Second, it is worth focusing on the AIVD dependence on the number of prototypes k. For PG, for a given α, a larger number of prototypes k implies a higher AIVD, since traits copied from prototypes are more uniformly distributed, but this has a significant effect only for large β values, again due to the uniformity being anyway in place for small β values. For MPG, the corresponding behavior is more subtle. While for large, $\beta \to 1$ values, the AIVD still increases with increasing k at a given α (for the same reason as for PG), the AIVD(α) curves corresponding to small β approach the AIVD(α) curve corresponding to large $\beta \to 1$ with increasing k, rather than remaining in place (which is the case for

PG). This is related to the fact that the upper bound on the fraction of randomly generated traits $1/(k+1)$ decreases with increasing k, thus decreasing the role of β in controlling the AIVD via the uniform component of the feature-level probability distributions.

Fig. 2.3 deals with the second-level fitting. Everything shown in this figure relies on α already being tuned (at the first level) such that the empirical AIVD is matched – as apparent from Fig. 2.2, the tuned α value depends on β and on k. Fig. 2.3 shows the dependence of the numerically computed SIVD quantity (with uncertainty ranges) on the β parameter, for several k values and for both the PG and MPG models. Moreover, it shows the empirical SIVD uncertainty range via the horizontal bands in the two panels. Thus, a solution of the second-level fitting is indicated by an intersection between a model curve of a given k and the horizontal band. Note, again, that for either of the models and either of the k values, if a solution exists, this solution is actually unique. The exact technical procedure employed for producing any of the model points in Fig. 2.3 is described at the end of Sec. 2.C, followed by the explanation of the final choice of values for the α and β parameters, for use in the analysis of Sec. 2.4.

Note that the SIVD increases with β for both models and for all k values, suggesting that the extent of feature-feature correlation increases with decreasing distance between vectors dominated by the same prototype. For PG, all SIVD(β) curves meet for some $\beta \approx 0.45$, at which point they also end. No points are plotted for lower β because α cannot be tuned in terms of AIVD, which can be understood from Fig. 2.2 when noticing the AIVD(α) curves of low β that do not cross the empirical line. For MPG, the SIVD(β) curve of $k = 2$ ends at a value of $\beta \approx 0.5$, before crossing the empirical line, meaning that the MPG model cannot be entirely tuned when only 2 prototypes are used. No points are plotted for higher β because α cannot be tuned in terms of AIVD, which can be understood from Fig. 2.2, by noticing the AIVD(α) curves of $k = 2$ and high β that do not cross the empirical line. This is due to certain limitations of the current modeling paradigm, which are further discussed in Sec. 2.5.

2.4 Model Outcomes

Here, the most important results of this work are presented. The focus is on the LTCD-STCB analysis, applied to sets of cultural vectors generated with the PG and MPG models. The aim is to assess how well the two models reproduce the universal empirical patterns described in Ref. [14] (Chap. 1). Fig. 2.4 illustrates the results obtained with the two models, whereas Fig. 2.5 summarizes, for comparison purposes, the empirical results, focusing on the nominal part of the Eurobarometer dataset (EBM_n) – formatted according to the procedure explained in Ref. [14] (Chap. 1).

Before describing the results, it is worth recalling the main ingredients of the LTCD-STCB analysis. This is essentially a two-dimensional plot showing

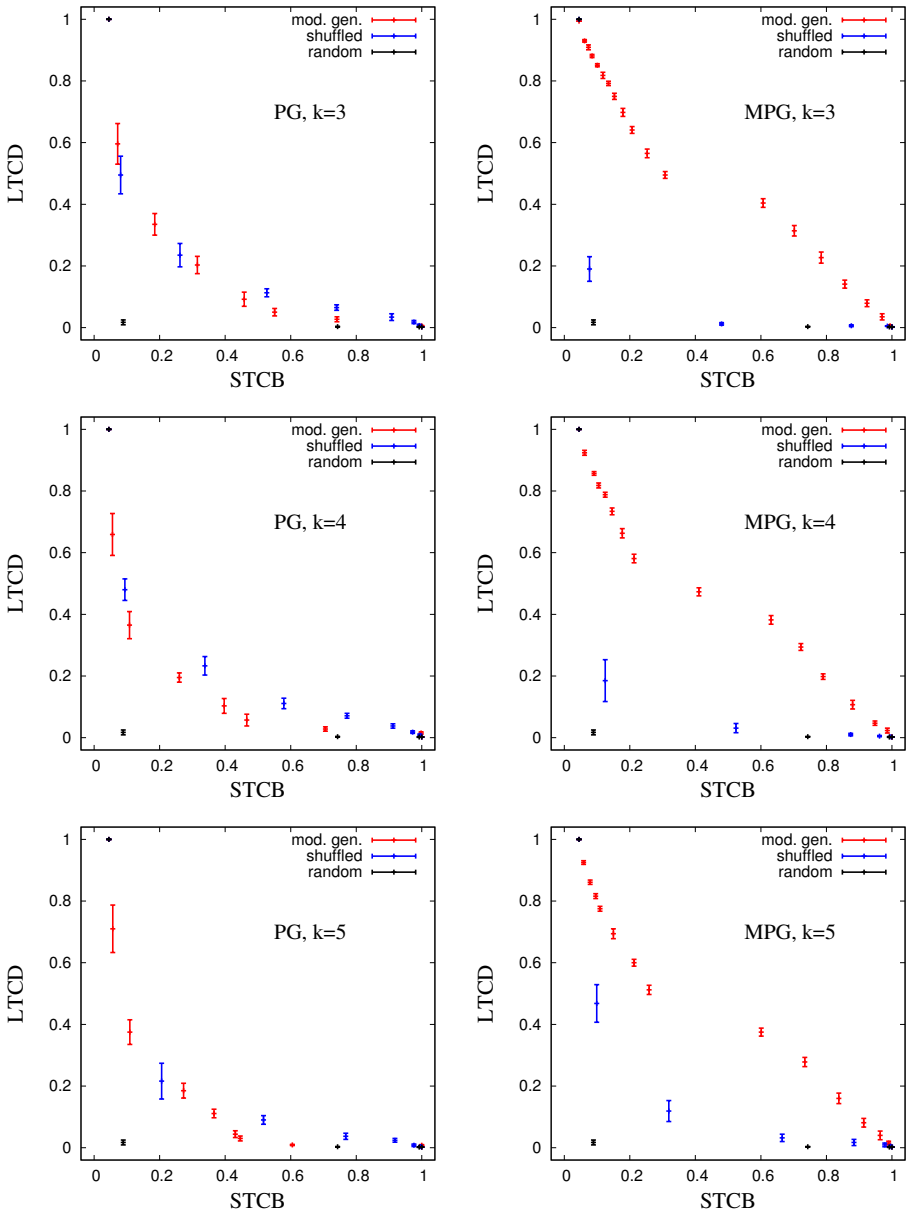

Figure 2.4: The correspondence between Long-term cultural diversity (LTCD) and short-term collective behavior (STCB) for a model-generated (red), a shuffled (blue) and a random (black) SCV obtained via the PG model (left) and MPG model (right), for $k = 3$ (top), $k = 4$ (centre) and $k = 5$ (bottom) prototypes. Error bars denote standard deviations over multiple trait dynamics runs. There are $N = 500$ elements in each set of cultural vectors.

the correspondence between the LTCD quantity vs the STCB quantity, both of them being evaluated on empirical, on shuffled and on random SCVs. Drawing the LTCD-STCB correspondence is made possible by the fact that, for each of the three scenarios, both quantities depend on the bounded-confidence threshold ω, which controls the maximal cultural distance over which social influence can act. On one hand, the LTCD quantity is a measure of cultural diversity after a long-term process of cultural dynamics driven by ω-bounded social influence, starting from an initial cultural state specified by the respective SCV. Essentially, it counts the number of distinct points in cultural space (commonly referred to as "cultural domains") towards which the agents converge in the final state of a minimalisitic, bounded-confidence Axelrod model. The STCB quantity is a measure of collective behavior (or social coordination) after a short-term process of opinion dynamics driven by ω-bounded social influence. Essentially, it is the standard deviation of the aggregate opinion distribution of the agent population, resulting from a minimalistic Cont-Bouchaud-type model applied to the (cultural) graph obtained by drawing a link for each pair of agents separated by a cultural distance smaller than ω. Mathematically, the two quantities, as functions of the bounded-confidence threshold ω, are captured by the following two expressions:

$$\mathrm{LTCD}(\omega) = \frac{\langle N_D \rangle_\omega}{N}, \quad \mathrm{STCB}(\omega) = \sqrt{\sum_A \left(\frac{S_A}{N}\right)^2_\omega}, \tag{2.9}$$

where N_D is the number of cultural domains in the final state of the Axelrod-type model, N is the number of agents (and cultural vectors) and S_A is the size of the A'th of connected components in the ω-determined cultural graph. The average in the LTCD formula is taken over multiple simulations of the Axelrod-type model. The STCB quantity is calculated analytically, once the cultural connected components are found, based on the assumption of independent opinion-agreement within each connected component. An essential difference between the two quantities, reflected in the long-term/short-term distinction, consists of an idealized separation between two time-scales, in terms of the role that the SCV specified as input plays: cultural vectors, together with the distances between them, are assumed to be dynamical by the LTCD definition and static by the STCB definition, such that one deals with dynamics of vectors and with dynamics on vectors in the two cases respectively. The interested reader is referred to Refs. [14, 12] (and Chap. 1) for more details and remarks about the LTCD-STCB analysis.

For both the PG and the MPG models, the α and β parameters are tuned in the manner described in Sec. 2.3 for every value of the number of prototypes k, while the latter is simply iterated over. In Fig. 2.4, the LTCD-STCB plot is shown for the values $k = 3$, $k = 4$ and $k = 5$, for the PG (left) and the MPG (right) models. The value $k = 2$ is omitted since the α and β parameters could not be both tuned for MPG with two prototypes. All SCVs are generated using the cultural space of $\mathrm{EBM_n}$, whose empirical SCVs also served for providing the AIVD and SIVD values in terms of which the tuning was conducted (Sec. 2.3).

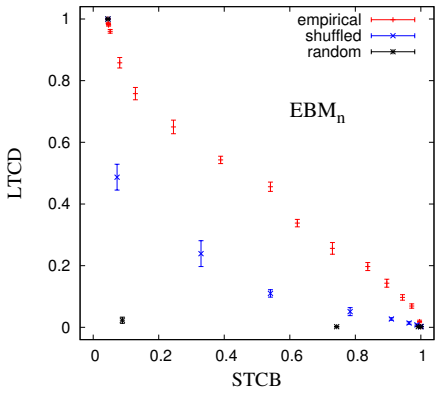

Figure 2.5: The correspondence between long-term cultural diversity (LTCD) and short-term collective behavior (STCB) for the empirical (red), shuffled (blue) and random (black) sets of cultural vectors, for the nominal part of the Eurobarometer data set (EBM$_n$). Error bars denote standard deviations over multiple cultural dynamics runs. There are $N = 500$ elements in each set of cultural vectors

When looking at Fig. 2.4, one should ask whether the universal, empirical patterns are reproduced by any of the six illustrated model scenarios. Qualitatively, the patterns are defined first in terms of a higher compatibility between LTCD and STCB in the model-generated SCV than in the shuffled SCV and a higher compatibility in the shuffled SCV than in the random one, second in terms of the model-generated LTCD-STCB curve being close to the second diagonal. These empirical features are visible in Fig. 2.5. It is clear that PG does not satisfy these criteria for any value of k. Indeed, the model-generated curve is far below the second diagonal for most of the relevant interval and often below the shuffled curve. MPG, however, appears to satisfy all these criteria for all k values, although for $k = 3$ it is not obvious that the shuffled curve is indeed above the random one, due to the lack of points in the lower-left corner. This has to do with the effective discreteness of the bounded-confidence threshold ω spectrum, due to the finite number of nominal features available – in other words, it is meaningless to split the ω axis into intervals that are smaller than the nearest-neighbor spacing of the cultural space lattice. For a direct comparison with analogous empirical curves, one should use Fig. 2.5, which shows the results of the LTCD-STCB analysis applied to EBM$_n$ data. However, it is only meaningful to compare the qualitative nature of the empirical and the model curves, rather than the exact values, since, as discussed in Sec. 2.5, neither model has a maximum-likelihood nature, due to a certain simplicity in the way prototypes are formalized and chosen here. Still, MPG apparently does generate SCVs that are structurally similar to the empirical ones. Thus, the notion of cultural prototypes, even if implemented in a simplistic

way, can be used to reproduce the important, universal properties of empirical cultural states, as long as mixing of prototypes is in place.

2.5 Discussion

The purpose of this study was to develop a way of generating cultural states that reproduce the apparently universal properties of the empirical ones, namely those described by Ref. [14] (Chap. 1). This naturally calls for input from social science, in particular from social science theories that are intended to describe universal aspects of culture and society. There is an entire "class" of social science theories that appear relevant for this purpose, originating from either psychology or cultural anthropology [16, 17, 18, 19, 20], some of them being explicit attempts at unifying social science. All of them make use of cultural prototypes, although in somewhat different ways, under different names and numbers. Moreover, they had all been overlooked by previous studies of cultural dynamics, on which Ref. [14] (Chap. 1) largely builds: Ref. [13] was the first study that connected quantitative studies of cultural dynamics with these theories, via the generic, formal notion of cultural prototypes. For creating an instructive and compact context, this work focused on one of these theories, namely on Plural Rationality Theory (PRT).

There are several aspects justifiying the focus on Plural Rationality Theory. First, its informal notion of cultural bias matches very well the more formal notion of cultural prototype, in the manner used in Ref. [13] and here. Second, it is more appealing from a natural science perspective than the others, in particular from a physics and complex systems perspective. This is largely due to various concepts that are qualitatively (and sometimes just implicitly) invoked by PRT, such as: energy landscapes, symmetry breaking, graph/network theory, dynamical systems, crossovers (possibly phase transitions), self-organization and fractals. Third, it explicitly claims to provide some insight into how preferences form: preferences are formed in the process of building social relations, while different patterns of relations (and types of institutional settings) go along with different conglomerates of preferences (the cultural biases). Finally, this dualism between patterns of relations on one hand and cultural biases on the other hand comes along with distinguishing between a "social plane" and a "cultural plane" of interacting human systems, while acknowledging the dynamical nature of both, as well as the strong coupling and interdependency between the two. Thus, PRT seems to resonate well, on one hand to research on social network structure and dynamics, on the other hand to research on cultural structure and dynamics.

Up to now, little work has been done to explore either of these two connections. While Ref. [13] and the present work are the first steps in exploring the latter connection, some steps have also been taken in exploring the former connection [35, 36]. Note, however, that Ref. [13] refers to several theories similar to PRT, without explicitly mentioning PRT, that Ref. [36] focuses on a social theory similar to PRT, while still discussing a connection with PRT and that Ref. [35]

works with an earlier, more rudimentary version of PRT, which gave less importance to the notions of "way of life", "rationality" and "cultural bias". Although the coupling between social dynamics and cultural dynamics is recognized and studied by quantitative complex systems research (for instance Refs. [9, 37]), this has been carried out in isolation from PRT.

In loose terms, each rationality of PRT has, as a "projection" on the cultural plane, one distinct cultural bias. These cultural biases correspond to the cultural prototypes used in this study. In agreement with Ref. [13], a cultural prototype is a combination of cultural traits, thus one point in cultural space – the limitations of this assumptions are extensively discussed below. Relying on these notions, two stochastic, structural models of culture are developed and studied here: Prototype Generation (PG) and Mixed Prototype Generation (MPG). It is important that, regardless of which model is used, once the prototypes and the remaining free parameters (parameter β, for either PG or MPG) are specified, one implicitly defines a cultural space distribution (CSD): a probability mass function taking the cultural space as a support, as defined in Ref. [14] (Chap. 1). Generating a set of N cultural vectors is then equivalent to selecting N points at random according to this distribution. Thus, the resulting cultural states are generated in a non-uniformly random way, with non-uniformities depending on the prototypes and on other model specifications.

For this study, the usage of both stochastic models is restricted to cultural spaces constructed only from sets of nominal features. This is due to the assumption that every prototype picks one and only one trait in any feature, which from a PRT perspective means that, upon answering a question under the influence of one cultural bias, a respondent can only provide one specific answer. In reality, even a specific cultural bias would generally point towards several answers, although with different probabilities, so it would be more realistic to say that every prototype corresponds to one probability distribution defined over that feature. Not allowing for this freedom makes this modeling paradigm incompatible to ordinal features, whose associated traits are by construction sorted along an axis, in which case it is not reasonable to assume that a prototype points to one trait of a feature with full probability and to its nearest-neighbors with zero probability. Nonetheless, the paradigm is reasonably compatible with nominal features, in which case the distance between any two traits of one feature is anyway assumed to be the same.

The current study belongs to a preliminary, simplistic paradigm which makes use of what one may call "sharp prototypes". A more realistic paradigm, which would account for the probabilistic nature of the cultural biases, would make use of what one may call "diffuse prototypes". Using sharp prototypes comes at the cost of not having enough flexibility to reproduce the empirical, feature-level frequency distributions, with either of the two models, since every prototype corresponds to a probability distribution entirely peaked on one trait. Instead, using diffuse prototypes would allow this by enforcing, for every feature, that the empirical distribution is a linear combination of the prototype distributions. Nonetheless,

as shown in Sec. 2.3, both models are still able to reproduce the empirical average uniformity of the feature-level frequency distributions, namely the AIVD quantity. This is partly due to both models making some use of uniformly-random trait generation, independently of the prototypes. This translates to a flat noise component in the probability distribution of every feature, which in a sense compensates for the rigid peaks of the sharp prototypes. When also considering the results of Sec. 2.4, the usage of sharp prototypes restricted to nominal variables appears to be enough as a proof of concept. This justifies further research towards the more sophisticated paradigm relying on diffuse prototypes. Although this is left for future studies, it is worth contemplating upon, in order to better understand the purpose, greater context and limitations of the current paradigm.

Working with diffuse prototypes should go hand in hand with a method of inferring them from data. One can imagine doing this by applying a sensible clustering method on the empirical set of cultural vectors, followed by a sensible method of constructing one diffuse cultural prototype from every cluster, as a probabilistic entity that is representative of that cluster. The main advantage of this approach is that once the prototypes are constructed and provided as input to a sensible stochastic model, the artificial SCVs generated with this model would be close-to-representative of the same distribution in cultural space as the empirical SCV on which the method is applied in the first place. This means that the model would have a maximum-likelihood flavor, and could be used for generating synthetic data, which would also reproduce the feature-level frequency distributions.

By contrast, the approximation of sharp prototypes used here is too strong to be employed together with a method of inferring them from data. Instead, sharp prototypes are being assigned to randomly chosen positions in the given cultural space. On one hand, the fact that the prototypes are randomly chosen makes any model symmetric up to any permutation of the traits of any feature, as long as all features are nominal, which is the case here, a symmetry which is broken by an empirical SCV and also by an artificial SCV generated from a specific choice of the prototypes. On the other hand, the fact the prototypes are sharp does not allow for the exact frequency distribution of a specific feature to be reproduced, not even up to a permutation of the traits. Still, after parameter tuning, one should expect from a good model to provide a cultural space distribution whose rough "shape" is compatible with the empirical data, though the "orientation" and the structural details implied, for instance, by the feature-level distributions would not be compatible. This should reflect in roughly reproducing the universal LTCD-STCB patterns emphasized in Ref. [14] (Chap. 1): one one hand, the formulation of the LTCD and STCB observables is also symmetric up to permuting the traits of any feature, and thus independent of the "orientation"; on the other hand, the empirical, feature-level frequency distributions should heavily depend on the specific data set, thus being of little relevance for the universal patterns.

There are various aspects that make the random generation of prototypes sensible for the purpose of the present work. First, results are evaluated for

various values of the number of prototypes k, which is considered a free parameter for both the PG and MPG model. Second, the expected prototype-prototype distance is controlled for via parameter α. Third, for every choice of parameters, the prototypes are independently drawn for each realized cultural state in the set used for computing the model AIVD and SIVD quantities for fitting purposes. These compensate somewhat for not inferring the prototypes from empirical data.

In order to give an example of how the sharp prototypes approximation can be pushed beyond its limits, it is worth recalling that fitting the MPG model is not possible for $k = 2$ prototypes, as pointed out at the end of Sec. 2.3: the α parameter can be successfully tuned in terms of the AIVD only for small β values, which do not allow for the subsequent fitting of the β parameter in terms of the SIVD. This is related to there being at least $q = 3$ traits associated to every nominal feature selected from the Eurobarometer data set, while there are only two, prototype-induced peaks in the model probability distribution of every feature, on top of the uniform component. Since the integrated probability of the uniform component cannot exceed $1/k$ by construction, all the distributions are bound to be relatively non-uniform, such that the empirical average uniformity is only attained for small-α (few coincidences between the prototype-induced peaks) and small-β (large uniform component) combinations. This does not hold for the PG model, as in this case the integrated probability of the uniform component can attain any value between 0 and 1. Nonetheless, if $k > 2$, the fitting of the MPG leads to generated cultural states that reproduce much better the universal empirical patterns than PG. This justifies considering MPG the successful model, while emphasizing the importance of the mixing ingredient, which validates the multiple self assumption.

When thinking in terms of the feature-level probability distributions, it might seem that the MPG and PG models are not that different from each other. As mentioned above, for both models, if there are k prototypes, the probability distribution of a certain feature would consist of k peaks of equal probability contents and of a uniform component associated to the explicitly random trait generation. Although the probability content of the uniform component of MPG is bounded from above, that of PG is not bounded in any way, so one might think that MPG is just a particular realization of PG. However, this reasoning is misleading, as it focuses on partial information encoded in the feature-level probability distributions, disregarding the rest of the information encoded in the complete cultural space distribution. With PG, a cultural vector whose trait, with respect to a certain feature, is generated under the probability peak of a certain prototype will have its trait generated, with respect to another feature, under the well-determined probability peak of the same prototype or under the uniform component. By contrast, with MPG, a cultural vector whose trait, with respect to a certain feature, is generated under the probability peak of a certain prototype, will have its trait generated, with respect to another feature, under the probability peak of any prototype – though with a higher likelihood under the peak of the dominating prototype – or under the uniform component. Thus, for the same choice

of the prototypes and the same extent of explicitly random generation of traits (and consequently the same AIVD), PG implies a different level of cross-feature correlation and a different shape of the cultural space distribution than MPG. This conceptually explains the impact of the mixing ingredient.

Although this study does not attempt at providing a complete mathematical theory of trait dynamics and formation, one can argue that the MPG model qualifies as a good effective, [3] static description of (generic snapshots of) trait dynamics. This static description is inspired by Plural Rationality Theory which, although originating in cultural anthropology, does seem to integrate notions of both psychology and of a (complex) systems based understanding of society. Although it is formulated in an a qualitative, informal way, Plural Rationality Theory and related research should be of use for developing a complete formal theory of trait dynamics, at least as a source of guidance and inspiration.

2.6 Summary and conclusions

This study was dedicated to developing and testing a stochastic model for generating cultural states that would be structurally similar to the empirical ones. The aim was to reproduce the universal, empirical properties pointed out in Ref. [14] (Chap. 1), while relying on some social science hypothesis. Following up on previous work, the idea of cultural prototypes was used for this purpose. The study first tested the hypothesis that each cultural vector is a partial realization of one prototype and random for the rest, which is what was previously assumed. This turned out to be insufficient for reproducing the empirical patterns. Instead, one has to assume that each cultural vector is a combination, or mixture of all prototypes, although still dominated by either of them, which is what the MPG model encodes. This additional, mixing ingredient is actually suggested by the same social science theories that inspired the prototypes idea in the first place. In this specific, social science context, this aspect is often referred to as "the multiple-self". These results provide indirect evidence for social science theories like PRT, that postulate, in one way or another, some notion of cultural prototypes, along with some associated notion of mixing.

Still, there is a certain rigidity in the way prototypes are currently formalized (Sec. 2.5), related to the assumption that every prototype corresponds to one and only one value of every cultural variable, instead of corresponding to a probability distribution over the variable. This makes the cultural space distribution induced by the successful, MPG model generally incompatible with the cultural space frequency distribution with respect to which it is fitted. As it stands, MPG is is far from being a maximum-likelihood type of model and thus cannot be used to generate synthetic data. Nonetheless, this is arguably achievable once diffuse

[3] Note that "effective description of" stands for "description of the effects of", for "approximate description" or for "phenomenological description", as used in the physics literature, rather than for "successful or "efficacious".

prototypes are used instead of sharp ones, while being inferred from the data rather than randomly chosen. In this sense, this work can be seen as an important step towards a realistic, maximum-likelihood model of empirical cultural states, and towards generating synthetic sets of cultural vectors. Moreover, MPG can be considered an effective description of the outcome of trait dynamics, since the generated cultural states seem to reproduce the generic structure of the empirical ones. The LTCD-STCB analysis, used for validating this effective theory, could also be used for validating a more fundamental, dynamical theory of culture. It appears likely that Plural Rationality Theory has more to say for aiding the development of such a theory.

Appendices

2.A Controlling the generation of prototypes

This section describes the calculation of probabilities attached to sets of cultural prototypes employed by the PG and MPG models defined in Sec. 2.2. These probabilities are collectively controlled via a parameter (α), which effectively dictates the expectation value of the average prototype-prototype cultural distance for one set of prototypes. The assignment of traits to prototypes is conducted independently for every feature, so the discussion is reduced to assigning probabilities to prototype-to-trait mappings at the level of a single feature. Furthermore, since prototype generation neglects empirical occurrence frequencies of specific traits, the problem is symmetric with respect to permutations of the traits, so the discussion is further reduced to assigning probabilities to "topologies" of prototype-to-trait mappings at the level of a single feature. Mathematically, such a topology is an "integer partition". Integer partitions turn out to be the mathematical objects to which elementary probabilities are to be assigned. Sec. 2.A.1 explains the procedure for assigning the probabilities to integer partitions, while Sec. 2.A.2 explains the procedure for generating the integer partitions.

2.A.1 Integer partition probabilities

Let I_k be the set of all integer partitions of k elements, where an integer partition of k elements is an ordered sequence of integers that add up to k, also called "parts". Let the ordered sequence $(k_1, ..., k_s) \in I_k$ be one generic element of this set, where s counts the number of non-zero parts. This notation implies that the parts are sorted for descending values $k_i \geq k_{i+1} \forall i \in \{1, .., s-1\}$ and that they add up to $k = \sum_{i=1}^{s} k_i$. For instance, $(3, 2, 2, 1)$ is an integer partition of 8 elements with 4 parts. For the purpose of this work, an element of the integer partition corresponds to one prototype. For a specific choice of the prototypes

and a specific feature, an integer partition is a representation of how the prototypes are distributed over the traits of this feature, up to a permutation of these traits. Thus, when the fraction of traits that are randomly generated vanishes, the probabilities of the traits are just the normalized part sizes – in the example above, the ordered sequence of probabilities associated to the traits would be $(\frac{3}{8}, \frac{2}{8}, \frac{2}{8}, \frac{1}{8})$. Random trait generation then simply introduces a uniform, noise component to the feature probability distribution, whose contribution increases with the fraction of traits that are randomly generated. Thus, the integer partition is in any case a proxy for the feature probability distribution, regardless of which stochastic model is used.

Let $c(k_1, ..., k_s)$ be the "compactness" of integer partition $(k_1, ..., k_s)$, defined by:

$$c(k_1, ..., k_s) = \sum_{i=1}^{s} \frac{k_i(k_i - 1)}{2}, \tag{2.10}$$

which counts the number of pairs of elements belonging to the same part. For instance, the compactness of integer partition $(3, 2, 2, 1)$ is $c(3, 2, 2, 1) = 3^2 + 1^2 + 1^2 + 0^2 = 11$. The compactness thus counts the prototype-prototype coincidences for one feature. In light of the above paragraph, a small compactness implies a high uniformity for the feature probability distribution and thus a high value of the associated (feature-level) AIVD contribution.

Let I_k^q be the set of integer partitions of k elements of at most q parts (which implies that $I_k^q \subseteq I_k$). This definition is needed for working with features with range $q < k$. Furthermore, **let** $c_{k,q}^{\min}$ and $c_{k,q}^{\max}$ be the minimal and maximal compactness values attainable by the elements of I_k^q. These notions are needed for normalizing generic compactness values. They formally read:

$$\begin{aligned} c_{k,q}^{\min} &= c(\lambda').(\lambda' \in I_k^q \wedge \nexists \lambda \in I_k^q.(c(\lambda) < c(\lambda'))), \\ c_{k,q}^{\max} &= c(\lambda').(\lambda' \in I_k^q \wedge \nexists \lambda \in I_k^q.(c(\lambda) > c(\lambda'))), \end{aligned} \tag{2.11}$$

where the "." (dot) notation stands for "with the property that".

At this point, it is possible to define an non-normalized probability mass function parametrized by α over the discrete set of integer partitions I_k^q, function whose shape would depend on α. High α values correspond to integer partitions of high compactness values being favored over those of low compactness values, while low α values correspond to integer partitions of low compactness values being favored over those of high compactness values. For simplicity, the function is chosen to be monotonous when re-expressed in terms of compactness. A simple choice for such a function, denoted here by $\rho_{k,q}^{\alpha}$, is given by:

$$\rho_{k,q}^{\alpha}(\lambda) = \exp\left\{ \tan\left[(2\alpha - 1)\frac{\pi}{2}\right] \frac{2c(\lambda) - c_{k,q}^{\max} - c_{k,q}^{\min}}{c_{k,q}^{\max} - c_{k,q}^{\min}} \right\}, \tag{2.12}$$

where the inner fraction linearly maps the compactness $c(\lambda)$ from interval $[c_{k,q}^{\min}, c_{k,q}^{\max}]$ to interval $[-1, 1]$, while the argument of the tan function linearly maps α from interval $(0, 1)$ to interval $(-1, 1)$, from where it is further mapped to $(-\infty, \infty)$ by the tan function. In this manner, the function is increasing with $c(\lambda)$ for $\alpha > 0.5$ (implying a relatively low expectation value of average prototype-prototype separation), the function is decreasing with $c(\lambda)$ for $\alpha < 0.5$ (implying a relatively high expectation value of average prototype-prototype separation) and the function is a constant of $c(\lambda)$ for $\alpha = 0.5$. The actual probability $P_{k,q}^{\alpha}(\lambda)$ associated to integer partition λ can then be obtained via the normalization:

$$P_{k,q}^{\alpha}(\lambda) = \frac{\rho_{k,q}^{\alpha}(\lambda)}{\sum_{\lambda \in I_k^q} \rho_{k,q}^{\alpha}(\lambda)}, \tag{2.13}$$

with the sum in the denominator being taken over all integer partitions in I_k^q.

2.A.2 Integer partition generation

Let $I \stackrel{\text{d}}{=} \{0^I, 1^I\} \cup I_1 \cup I_2 \cup ...$ be the set of all integer partitions of any size, together with a "null" element 0^I and a "unity" element 1^I, which are meaningful in relation to the $\oplus$ operation defined below and are needed for keeping some of the following definitions compact and self-consistent.

Let the integer partition "merging" $\oplus : I \times I \to I$, acting on two integer partitions of k_a and k_b elements, with s_a and s_b parts respectively, be defined in the following way:

$$(k_1^a, ..., k_{s_a}^a) \oplus (k_1^b, ..., k_{s_b}^b) = (k_1, ..., k_s), \tag{2.14}$$

producing another integer partition of $k = k_a + k_b$ elements and $s = s_a + s_b$ parts, such that the sequence of parts in the resulting partition is a sorted merging of the two original sequences of parts. For instance: $(3, 2, 2, 1) \oplus (4, 2) = (4, 3, 2, 2, 2, 1)$. Moreover, any integer partition $\lambda \in I$ satisfies $\lambda \oplus 0^I = 0^I$ and $\lambda \oplus 1^I = \lambda$.

Let the integer partition "multi-merging" $\otimes : I \times \mathcal{P}(I) \to \mathcal{P}(I)$, where $\mathcal{P}(I)$ is the set of all subsets of I, be defined by:

$$\alpha \otimes \{\alpha_1, ..., \alpha_\sigma\} = \{\alpha \oplus \alpha_1, ..., \alpha \oplus \alpha_\sigma\}, \tag{2.15}$$

where $\alpha, \alpha_1, ..., \alpha_\sigma \in I$ are all integer partitions. The $\otimes$ operation produces a set of integer partitions of σ elements from an initial set of integer partitions of the same size and another integer partition α, by merging α with each element α_i in the initial set via the $\oplus$ operation.

Relying on the notions above, the following recursive definition of function $\text{sip}(k, m_L, m_V) : \mathbb{N} \times \mathbb{N}^* \times \mathbb{N}^* \to \mathcal{P}(I)$ encodes the procedure for generating the set of integer partitions of k elements, of maximally m_L parts, with maximal part

value m_V:

$$\mathrm{sip}(k, m_L, m_V) = \\ = \begin{cases} \{1^I\} & k = 0, \\ \{0^I\} & k > m_L \cdot m_V, \\ \{(m_V, ..., m_V)_{m_L \text{ entries}}\} & k = m_L \cdot m_V, \\ \bigcup_{x \in \overline{1, \min(k, m_V)}} [(x) \otimes \mathrm{sip}(k - x, m_L - 1, x)] & \text{else.} \end{cases} \tag{2.16}$$

definition inspired by Ref. [38], where the order of the four cases matters, in the sense that one case is considered only if none of the conditions of the above cases is valid. The last line returns the set resulted from the reunion "∪" of all sets of integer partitions of type $(x) \otimes \mathrm{sip}(k - x, m_L - 1, x)$, where x spans the indicated interval. This general formulation, which also takes the maximal part value m_V as argument, is required for a compact recursive definition. But of actual interest for this work is the set of integer partitions of k elements and maximal part value q, I_k^q, given by:

$$I_k^q = \mathrm{sip}(k, q, k) - \{0^I, 1^I\}, \tag{2.17}$$

where the last part of the expression takes out the null and/or the unity element, which might be present in the set of integer partitions as leftovers from the computation. Here we explicitly show how the sip function works when calculating the set of integer partitions of 4 elements of maximally 3 parts, given by $I_4^3 = \mathrm{sip}(4, 3, 4) - \{0^I, 1^I\}$, where:

$$\begin{aligned} &\mathrm{sip}(4,3,4) = \bigcup_{x \in \overline{1,4}} (x) \otimes \mathrm{sip}(4 - x, 2, x) = \\ &= [(1) \otimes \mathrm{sip}(3,2,1)] \cup [(2) \otimes \mathrm{sip}(2,2,2)] \cup [(3) \otimes \mathrm{sip}(1,2,3)] \cup [(4) \otimes \mathrm{sip}(0,2,4)] = \\ &= [(1) \otimes \{0^I\}] \cup [(2) \otimes \bigcup_{x \in \overline{1,2}} (x) \otimes \mathrm{sip}(2 - x, 1, x)] \cup \\ &\cup [(3) \otimes (1) \otimes \mathrm{sip}(0,1,1)] \cup [(4) \otimes \{1^I\}] = \\ &= \{0^I\} \cup [(2) \otimes [[(1) \otimes \mathrm{sip}(1,1,1)] \cup [(2) \otimes \mathrm{sip}(0,1,2)]]] \cup [(3) \otimes (1) \otimes \{1^I\}] \cup \{(4)\} = \\ &= \{0^I\} \cup [(2) \otimes [[(1) \otimes \{(1)\}] \cup [(2) \otimes \{1^I\}]]] \cup [(3) \otimes \{(1)\}] \cup \{(4)\} = \\ &= \{0^I\} \cup [(2) \otimes [\{(1,1)\} \cup \{(2)\}]] \cup \{(3,1)\} \cup \{(4)\} = \\ &= \{0^I\} \cup [(2) \otimes \{(1,1), (2)\}] \cup \{(3,1), (4)\} = \\ &= \{0^I\} \cup \{(2,1,1), (2,2)\} \cup \{(3,1), (4)\} = \\ &= \{0^I, (2,1,1), (2,2), (3,1), (4)\}, \end{aligned} \tag{2.18}$$

yealding $I_4^3 = \{(2,1,1), (2,2), (3,1), (4)\}$, which is the expected result.

2.B Analytic calculations of model average inter-vector distance

This section explains the analytic calculation for the expectation value of the average inter-vector distance (AIVD) for sets of cultural vectors generated using either the PG or MPG model. The first part of this section just gives the essential formulas – Eqs. (2.19) and (2.20)) are common for the two models; the difference between the model becomes apparent when comparing Eq. 2.21 with Eq. 2.22. The second part gives the proof Eq. (2.8), which is the basis for Eq. (2.20).

The expectation value of the AIVD, as a function of the three model parameters k, α, β is given by the average over the feature-level expectation values:

$$\langle \mathrm{AIVD} \rangle^{k}_{\alpha,\beta} = \frac{1}{F} \sum_{q} n_q \, \langle \mathrm{AIVD} \rangle^{k,q}_{\alpha,\beta} \,, \tag{2.19}$$

where the sum goes over all possible values ranges q and n_q is the number of features with range q, with $\sum_q n_q = F$ being implicitly satisfied, where F is the number of features. Note that the feature-level contribution also depends on q. In turn, this contribution is given by:

$$\begin{aligned} \langle \mathrm{AIVD} \rangle^{k,q}_{\alpha,\beta} = 1 - \sum_{(k_1,\ldots,k_s)}^{I^q_k} & P^{\alpha}_{k,q}(k_1,\ldots,k_s) \cdot \\ & \cdot \left\{ \sum_{i=1}^{s} \left[\pi^{k}_{\beta,F} \frac{k_i}{k} + \left(1 - \pi^{k}_{\beta,F}\right) \frac{1}{q} \right]^2 + (q-s) \left(\frac{1 - \pi^{k}_{\beta,F}}{q} \right)^2 \right\}, \end{aligned} \tag{2.20}$$

which is essentially a weighted averaging of Eq. (2.8) over the set of integer partitions $(k_1, \ldots, k_s) \in I^q_k$, where the weights are the integer partition probabilities $P^{\alpha}_{k,q}(k_1, \ldots, k_s)$. These are calculated in the manner described in Sec. 2.A.1, while the integer partitions themselves are generated in the manner described in Sec. 2.A.2. The set of p_i's of Eq. (2.8) depends on the integer partition in the manner illustrated between the braces of Eq. (2.20), where the first term accounts for the s traits that are covered by the (non-zero) elements of the integer partition, namely those under the peak(s) of one (or more) prototype and under the flat noise component, while the second term accounts for the remaining $q - s$ traits, namely those that are only under the flat noise component. The dependence on whether the PG or the MPG model is used is captured by $\pi_{\beta,F}$, which is the average fraction of traits directly copied from prototypes, given by:

$$\pi^{k}_{\beta,F} = \frac{\mathrm{round}(\beta F)}{F}, \tag{2.21}$$

for PG, where the "round" function accounts for the fact that only integer numbers

of traits can be copied, and:

$$\pi^k_{\beta,F} = \frac{1}{|W^\beta_{k+1}|} \sum_{w}^{W^\beta_{k+1}} w, \tag{2.22}$$

for MPG, where w iterates over all values of W^β_{k+1}, which is a large sequence of lowest MPG discrete weights (see Sec. 2.2), which are numerically generated during a previous step, for each used combination of (k, β) values. $|W^\beta_{k+1}|$ is the number of elements in this sequence of discrete weights. For this study, $|W^\beta_{k+1}| = 10^5$ elements were generated for every (k, β) combination, which allows for a very precise numerical calculation of $\pi^k_{\beta,F}$ in the case of MPG.

The consistency between the analytical AIVD calculation explained above and the numerical calculation is illustrated here via Fig. 2.6. The expected AIVD value is shown as a function of the β parameter, for 5 values of the α parameter and 3 values of the k parameter, for both the PG and MPG models. The analytical values are shown by the lines, while the numerical ones are shown by the dots, which have small, almost indiscernible error bars attached. For the numerical case, 50 sets of $N = 500$ cultural vectors are generated for each combination of parameters. Note that the numerical profiles follow closely the analytical ones, with small deviations that are consistent with the expected fluctuations of the mean.

It is now worth presenting a proof of Eq. (2.8), on which Eq. (2.20) is based. Consider a feature with q traits and a set of a-priori probabilities $\{p_1, ..., p_q\}$ attached to them. Then, the entry of each cultural vector generated with respect to this feature is an independent, random choice from the q traits, according to the probability mass function $(p_1, ..., p_q)$. Thus, the expected AIVD contribution from N cultural vectors is given by:

$$\begin{aligned}
&\langle \mathrm{AIVD}\left(\{p_1, ..., p_q\}\right)\rangle = 1 - \frac{2}{N(N-1)} \cdot \\
&\cdot \sum_{x_1,...,x_q}^{x_1+...+x_q=N} \sum_{i=1}^{q} \frac{x_i(x_i-1)}{2} f\left(N, {}^{x_1,...,x_q}_{p_1,...,p_q}\right) = 1 - \frac{2}{N(N-1)} \cdot \\
&\cdot \sum_{i=1}^{q} \sum_{x_i}^{x_i \le N} \frac{x_i(x_i-1)}{2} \sum_{x_1,...,x_{i-1},x_{i+1},...,x_q}^{x_1+...+x_{i-1}+x_{i+1}+...+x_q=N-x_i} f\left(N, {}^{x_1,...,x_q}_{p_1,...,p_q}\right) = \\
&\qquad = 1 - \frac{2}{N(N-1)} \sum_{i=1}^{q} \sum_{x_i}^{x_i \le N} \frac{x_i(x_i-1)}{2} S_i,
\end{aligned} \tag{2.23}$$

where $f\left(N, {}^{x_1,...,x_q}_{p_1,...,p_q}\right)$ denotes the probability that the N independent, random variables fill the q traits with the frequency distribution $(x_1, ..., x_q)$, given the associated probability distribution $(p_1, ..., p_q)$, where $\sum_{i-1}^{q} x_i = N$. This is conventionally called the multinomial distribution. In the above derivation, S_i stands

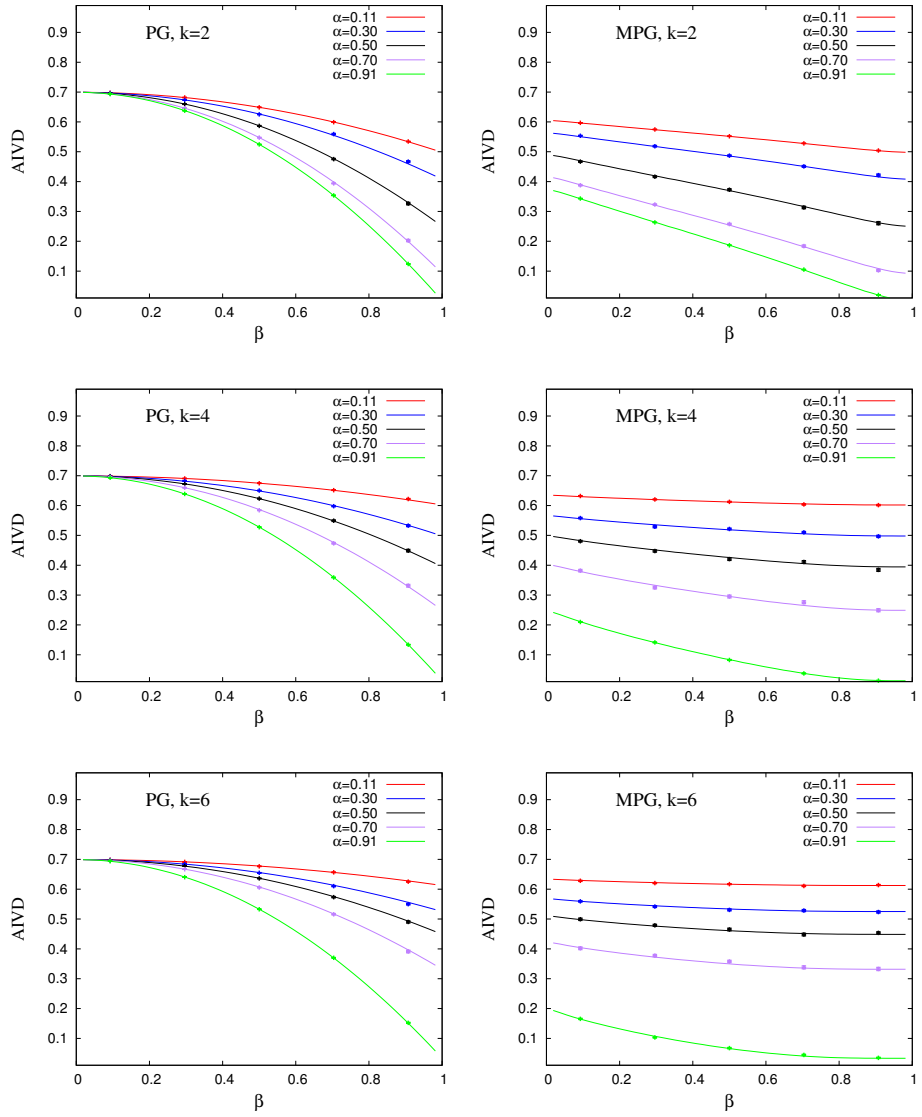

Figure 2.6: Comparison between numerical (dots) and analytical (line) expected AIVD as a function of β, for the PG (left) and MPG (right) models, with $k = 2$ (top), $k = 4$ (center) and $k = 6$ (bottom) prototypes, for several values of α (legend).

for the summation over all elements of the multinominal except that which has a certain, x_i number of entries for the ith trait, which can be further manipulated:

$$S_i = \sum_{x_1,\ldots,x_{i-1},x_{i+1},\ldots,x_q}^{x_1+\ldots+x_{i-1}+x_{i+1}+\ldots+x_q=N-x_i} f\left(N, {}^{x_1,\ldots,x_q}_{p_1,\ldots,p_q}\right) = \sum_{x_1,\ldots,x_{i-1},x_{i+1},\ldots,x_q}^{x_1+\ldots+x_{i-1}+x_{i+1}+\ldots+x_q=N-x_i} \frac{N!}{x_1!\ldots x_{i-1}!x_i!x_{i+1}!\ldots x_q!} p_1^{x_1}\ldots p_{i-1}^{x_{i-1}} p_i^{x_i} p_{i+1}^{x_{i+1}}\ldots p_q^{x_q} = p_i^{x_i}\frac{N!}{x_i!(N-x_i)!}\cdot$$
$$\cdot \sum_{x_1,\ldots,x_{i-1},x_{i+1},\ldots,x_q}^{x_1+\ldots+x_{i-1}+x_{i+1}+\ldots+x_q=N-x_i} \frac{(N-x_i)!}{x_1!\ldots x_{i-1}!x_{i+1}!\ldots x_q!} p_1^{x_1}\ldots p_{i-1}^{x_{i-1}} p_{i+1}^{x_{i+1}}\ldots p_q^{x_q} =$$
$$= \binom{N}{x_i} p_i^{x_i}(1-p_i)^{N-x_i}. \quad (2.24)$$

This shows that S_i is just a term of the binomial distribution. By inserting the final expression of Eq. (2.24) in the final expression of (2.23), one gets:

$$\langle \mathrm{AIVD}\left(\{p_1,\ldots,p_q\}\right)\rangle = 1 - \frac{1}{N(N-1)} \sum_{i=1}^{q} \sum_{x_i}^{x_i \leq N} (x_i^2 - x_i) \binom{N}{x_i} p_i^{x_i}(1-p_i)^{N-x_i} =$$
$$= 1 - \frac{1}{N(N-1)} \sum_{i=1}^{q} \left[Np_i(Np_i - p_i + 1) - Np_i\right] = 1 - \sum_{i=1}^{q} p_i^2, \quad (2.25)$$

which concludes the proof of Eq. (2.8), after using the well known expressions for the first and second moments $\langle x_i\rangle$ and $\langle x_i^2\rangle$ of the binomial distribution. Note that the dependence on N cancels out during the derivation.

Another, arguably shorter proof can be formulated with the aid of indicator functions of the type $\mathbb{I}_i(x)$, which gives 1 if cultural vector x is an entry of trait i and gives 0 otherwise. One can express the feature-level AIVD of one, generic set of cultural vectors in terms of indicator functions and write the expected, feature-level AIVD as an average of this expression. The p_i^2 part of Eq. (2.8) then appears from an averaging of the $\mathbb{I}_i(x)\mathbb{I}_i(y)$ product, where x and y are two arbitrary cultural vectors.

2.C Fitting algorithm

This section explains the procedure used for simultaneously tuning the α and β parameters of either of the two stochastic models of culture, such that a match is obtained between the model and the empirical data, in terms of the averages of the AIVD and SIVD observables:

$$\begin{aligned} \langle \mathrm{AIVD}(\alpha,\beta)\rangle &= \mathrm{AIVD}_{\mathrm{emp}}, \\ \langle \mathrm{SIVD}(\alpha,\beta)\rangle &= \mathrm{SIVD}_{\mathrm{emp}}, \end{aligned} \quad (2.26)$$

for a fixed number of prototypes k, assuming that either of the two equalities above is satisfied when there is an overlap between the uncertainty range associated to the quantity on the left side and that associated to the quantity on the right side.

There are multiple reasons why this problem is challenging:

- an analytical formula for the $\langle \mathrm{SIVD}(\alpha, \beta) \rangle$ quantity could not be found
- although an analytical formula for the $\langle \mathrm{AIVD}(\alpha, \beta) \rangle$ quantity was found (Eqs. (2.19) to (2.22)) [4], it does not allow for inverting the function and for analytically solving the system
- the $\langle \mathrm{SIVD}(\alpha, \beta) \rangle$, $\mathrm{AIVD}_{\mathrm{emp}}$ and $\mathrm{SIVD}_{\mathrm{emp}}$ quantities have non-vanishing uncertainty ranges attached to them

Assuming that there exists a unique solution to the above system, a numerical approach for solving it is in order. The method used here relies on a nested, 2-level, adapted bisection method. The first (inner) level of the method takes care of fitting, via bisection, the first quantity for a fixed β – it finds the α value for which $\langle \mathrm{AIVD}(\alpha, \beta) \rangle = \mathrm{AIVD}_{\mathrm{emp}}$ is satisfied for a given β. The second (outer) level of the method takes care of fitting, via bisection, the second quantity – it finds the β for which $\langle \mathrm{SIVD}(\alpha(\beta), \beta) \rangle = \mathrm{SIVD}_{\mathrm{emp}}$ is satisfied, where $\alpha(\beta)$ is provided by the first level. This choice of assigning the AIVD and SIVD observables and the α and β parameters to the two levels in this manner is numerically convenient for several reasons. First, the AIVD can be much more easily computed via the analytical formula, such that assigning it to the first level, which is repeated multiple times (once for each value of β that the second level samples) is more effective. Second, the model AIVD turns out to be relatively insensitive to β for relatively many combinations of values for the k and α parameters, such that fitting AIVD in terms of α within the first level makes more sense.

In addition to adaptations required by the 2-level scheme, other adaptations with respect to the traditional bisection method are needed for allowing it to work with model and empirical uncertainties, as well as to enhance the numerical precision for the $\langle \mathrm{SIVD}(\alpha, \beta) \rangle$ quantity when needed, to the extent needed. Moreover, in addition to statistical errors originating directly in the empirical uncertainties of the $\mathrm{AIVD}_{\mathrm{emp}}$ and $\mathrm{SIVD}_{\mathrm{emp}}$ quantities and in the numerical uncertainty of the model SIVD quantity, the second level of the method is also affected by "systematic errors" on $\langle \mathrm{SIVD}(\alpha(\beta), \beta) \rangle$, originating in the fitting procedure at the first level, and indirectly in the empirical uncertainty of $\mathrm{AIVD}_{\mathrm{emp}}$ – which for all practical purposes can be assumed fixed, thus motivating using the term "systematic" for its propagation to the model SIVD at the second level.

In order to address all these challenges in a self consistent way, the method developed here turns out to be quite sophisticated, which is why it is explained in detail in the following four sections. Specifically, Sec. 2.C.1 focuses on the first fitting level, Sec. 2.C.2 focuses on the second fitting level, Sec. 2.C.3 describes how

[4] Which implies that the specific uncertainty range of $\langle \mathrm{AIVD}(\alpha, \beta) \rangle$ has a null width.

various sub-problems invoked by the previous two sections are addressed, while Sec. 2.C.4 describes how the tools presented in Sections 2.C.1, 2.C.2 and 2.C.3 are used for producing some of the results presented in Sections 2.3 and 2.4. The method is potentially of use for addressing other problems that are formally similar to the problem presented here, although certain adaptations might be needed.

Since the method has mostly an algorithmic nature, much of it is explained via pseudocode, such that a few conventions that will be extensively used below and that are not necessarily standard are worth mentioning. First, the "=" symbol is used with double meaning: in a normal statement (such as "$a = b$") it is to be interpreted as an assignment (of the value of variable b to variable a); in the header of an **if** or **while** statement (such as "**if** $a = b$") it is to be interpreted as a check (of whether the values of a and b are equal). A variable is implicitly declared when it first appears, either on the left side of an assignment or in the header of a function definition (in which case it is also called an argument or function parameter); the scope of the variable is the part of the function below and to the right of the place where it first appears. Functions are distinguished from each other through their names, their numbers of arguments and the types of those arguments [5] On the other hand, the arguments of a function are distinguished from each other via their order. Some variables are actually ordered sequences of other variables, which in turn are denoted by $(x_1, .., x_n)$ notation. In the same spirit, an assignments of the type $X = (x_1, .., x_n)$ is referred to as a "variable compression", while one of the type $(x_1, .., x_n) = X$ is referred to as a "variable decompression". These allow for keeping the pseudocode compact, while still rigorous. An uncertainty range refers to an interval $[x - \delta x, x + \delta x]$, where x is a mean and δx is an error relying (directly, or indirectly) on a standard mean error calculation, the uncertainty range being formally encoded by the sorted $(x, \delta x)$ sequence. Note that the square brackets "[,]" are consistently used to denote an interval of real numbers, while the round brackets "(,)" are used to denote an ordered sequence of two or more elements. Finally, it is worth noting that the pseudocode relies heavily on function calls and on recursive definitions, and that there is a certain parallelism between the functions defined in Sec. 2.C.1 and those defined in Sec. 2.C.2.

2.C.1 First level fitting

This section presents the algorithm part concerned with the first fitting level. The algorithm is split in three main functions: Fit-1, Bisect-1, Displace-1, all of them returning the same type of information. Fit-1 always calls Bisect-1, while the latter may or may not call Displace-1 at any stage, which in turn may

[5] Sometimes this can be confusing, since the types of the arguments are only mentioned in the text before the definition of the function. In these cases however, the reader is guided by the names of the arguments, which in the function definition are kept as close as possible to those in the function call(s).

or may not call BISECT-1. The pseudocode also invokes two constants, which are assumed to be known a-priori and available for use anywhere in these three functions. The first constant is $\delta\alpha$, which controls the desired resolution ($\delta\alpha$ is essentially a grid-spacing) in the α parameter, which is here set to the inverse of the number of features: $\delta\alpha = \frac{1}{F}$ [6]. The second constant is $\mathrm{AIVD}_{\mathrm{emp}}$, which stands for the AIVD uncertainty range for the empirical data.

Function FIT-1 acts as an interface for the first-level fitting, which consists of tuning the α parameter, for given values of β and k, such that the AIVD quantity matches the empirical value. Here, β is a real number belonging to $[0, 1]$ while k is a strictly positive integer number. The method returns the left (α_L) and right (α_R) margins of the tightest α interval found, together with the estimated α match within this interval assuming linearity (α_{fit}) and an associated error (α_{err}). It assumes that the empirical AIVD can actually be uniquely matched by varying α, for the given values of β and k. The method essentially carries out some initializations (Lines 2,3), before passing the task to BISECT-1.

1: **function** FIT-1(β, k)
2: $(\alpha_L, \alpha_R) = \text{Init-1}(\delta\alpha)$ ▷ initializing the α-interval
3: $\mathrm{AIVD}_L = \langle\mathrm{AIVD}\rangle^k_{\alpha_L,\beta}$; $\mathrm{AIVD}_R = \langle\mathrm{AIVD}\rangle^k_{\alpha_R,\beta}$ ▷ analytics (Eq. (2.19))
4: **return** BISECT-1($\alpha_L, \alpha_R, \mathrm{AIVD}_L, \mathrm{AIVD}_R, \beta, k$)
5: **end function**

Function BISECT-1 is mostly a typical, recursive implementation of the bisection method. This sequentially narrows down the $[\alpha_L, \alpha_R]$ interval, such that at each stage the empirical AIVD is contained, namely that $\min(\mathrm{AIVD}_L, \mathrm{AIVD}_R) < \mathrm{AIVD}_{\mathrm{emp}} < \max(\mathrm{AIVD}_L, \mathrm{AIVD}_R)$ is satisfied, where the AIVD_L and AIVD_R values correspond to the left and right margins of the α interval. Here, α_L, α_R, AIVD_L and AIVD_R are real numbers belonging to $[0, 1]$ while β and k are of the same type as in FIT-1. It returns the same type of information as FIT-1. The method converges, the fitting being considered complete, when the interval has reached the $\delta\alpha$ resolution limit, in which case estimations for an "ideal" α inside this interval α_{fit} and its error α_{err} are made and returned together with the boundaries of the interval (lines 3-6). Moreover, the method may also call DISPLACE-1 in case the AIVD_M value corresponding to the computed midpoint α_M happens to fall within the $\mathrm{AIVD}_{\mathrm{emp}}$ uncertainty range (lines 8-10) – this is needed in order to keep the output format consistent and the final α interval relatively narrow. Otherwise, the method decides to zoom in (by calling itself) on either the left or right halves of the interval, depending on the position of $\mathrm{AIVD}_{\mathrm{emp}}$ with respect to AIVD_L, AIVD_M and AIVD_R (lines 11-16).

1: **function** BISECT-1($\alpha_L, \alpha_R, \mathrm{AIVD}_L, \mathrm{AIVD}_R, \beta, k$)
2: $\alpha_M = \text{MIDDLE}(\alpha_L, \alpha_R, \delta\alpha)$ ▷ computing midpoint on the α grid

[6] There is no clear lower bound on $\delta\alpha$, regardless of which stochastic model is used, but $\frac{1}{F}$ is a lower bound on $\delta\beta$ when PG is used, so for simplicity the choice $\delta\alpha = \delta\beta = \frac{1}{F}$ is made.

3: **if** $\neg$DISTINCT$(\alpha_M, \alpha_L, \alpha_R)$ **then**
4: $(\alpha_{\text{fit}}, \alpha_{\text{err}}) =$ INTERNFITLIN-1$(\alpha_L, \alpha_R, \text{AIVD}_L, \text{AIVD}_R, \text{AIVD}_{\text{emp}})$
5: **return** $(\alpha_L, \alpha_R, \alpha_{\text{fit}}, \alpha_{\text{err}})$ ▷ fitting complete
6: **end if**
7: $\text{AIVD}_M = \langle \text{AIVD} \rangle^k_{\alpha_M, \beta}$ ▷ analytics (Eq. (2.19))
8: **if** MATCH-1$(\text{AIVD}_M, \text{AIVD}_{\text{emp}})$ **then**
9: **return** DISPLACE-1$(\alpha_L, \alpha_R, \alpha_M, \text{AIVD}_L, \text{AIVD}_R, \beta, k)$
10: **end if**
11: **if** ORD-1$(\text{AIVD}_L, \text{AIVD}_R) =$ ORD-1$(\text{AIVD}_M, \text{AIVD}_{\text{emp}})$ **then**
12: $\alpha_L = \alpha_M$; $\text{AIVD}_L = \text{AIVD}_M$ ▷ selecting right interval
13: **else**
14: $\alpha_R = \alpha_M$; $\text{AIVD}_R = \text{AIVD}_M$ ▷ selecting left interval
15: **end if**
16: **return** BISECT-1$(\alpha_L, \alpha_R, \text{AIVD}_L, \text{AIVD}_R, \beta, k)$ ▷ zooming in
17: **end function**

Function DISPLACE-1 attempts to displace the midpoint α_M previously calculated at some stage in BISECT-1, in a way that its associated AIVD would fall outside the empirical uncertainty range. This function has all the arguments of BISECT-1 and α_M as an additional one, which is a real number belonging to $[0, 1]$. It returns the same type of information as FIT-1. The method first computes a "secondary" midpoint α'_M to the left of α_M and its corresponding AIVD'_M value. If the resolution limit $\delta\alpha$ is not reached and AIVD'_M falls outside the AIVD_{emp} range, BISECT-1 is applied further to the $[\alpha'_M, \alpha_R]$ interval (lines 2-11). Otherwise, the analogous procedure is applied on the right side (12-21). If the procedure fails to provide a convenient, secondary midpoint on either side, the fitting is considered complete with the current $[\alpha_L, \alpha_R]$ interval and the $\alpha_{\text{fit}}, \alpha_{\text{err}}$ estimates made like in BISECT-1 (lines 22-23).

1: **function** DISPLACE-1$(\alpha_L, \alpha_R, \alpha_M, \text{AIVD}_L, \text{AIVD}_R, \beta, k)$
2: $\alpha'_M =$ MIDDLE$(\alpha_L, \alpha_M, \delta\alpha)$ ▷ trying displacement to the left
3: **if** DISTINCT$(\alpha'_M, \alpha_L, \alpha_M)$ **then**
4: $\text{AIVD}'_M = \langle \text{AIVD} \rangle^k_{\alpha'_M, \beta}$ ▷ analytics (Eq. (2.19))
5: **if** $\neg$MATCH-1$(\text{AIVD}'_M, \text{AIVD}_{\text{emp}})$ **then**
6: **if** ORD-1$(\text{AIVD}_L, \text{AIVD}_R) =$ ORD-1$(\text{AIVD}'_M, \text{AIVD}_{\text{emp}})$ **then**
7: $\alpha_L = \alpha'_M$; $\text{AIVD}_L = \text{AIVD}'_M$
8: **return** BISECT-1$(\alpha_L, \alpha_R, \text{AIVD}_L, \text{AIVD}_R, \beta, k)$ ▷ zooming in
9: **end if**
10: **end if**
11: **end if**
12: $\alpha'_M =$ MIDDLE$(\alpha_M, \alpha_R, \delta\alpha)$ ▷ trying displacement to the right
13: **if** DISTINCT$(\alpha'_M, \alpha_M, \alpha_R)$ **then**
14: $\text{AIVD}'_M = \langle \text{AIVD} \rangle^k_{\alpha'_M, \beta}$ ▷ analytics (Eq. (2.19))
15: **if** $\neg$MATCH-1$(\text{AIVD}'_M, \text{AIVD}_{\text{emp}})$ **then**

16: **if** ORD-1($\text{AIVD}_L, \text{AIVD}_R$) $\neq$ ORD-1($\text{AIVD}'_M, \text{AIVD}_{\text{emp}}$) **then**
17: $\alpha_R = \alpha'_M$; $\text{AIVD}_R = \text{AIVD}'_M$
18: **return** BISECT-1($\alpha_L, \alpha_R, \text{AIVD}_L, \text{AIVD}_R, \beta, k$) ▷ zooming in
19: **end if**
20: **end if**
21: **end if**
22: $(\alpha_{\text{fit}}, \alpha_{\text{err}})$ = INTERNFITLIN-1($\alpha_L, \alpha_R, \text{AIVD}_L, \text{AIVD}_R, \text{AIVD}_{\text{emp}}$)
23: **return** $(\alpha_L, \alpha_R, \alpha_{\text{fit}}, \alpha_{\text{err}})$ ▷ fitting complete
24: **end function**

2.C.2 Second level fitting

This section presents the algorithm part concerned with the second fitting level. Each of the three functions of the first fitting level (Sec. 2.C.1) has a correspondent here: FIT-2, BISECT-2, DISPLACE-2, all of them returning the same type of information [7], each of them having a similar, structure, purpose and role to the correspondent within the first fitting level. Additionally, this section presents the pseudocode for a fourth function, NUMSIVD, which carries out the numerical SIVD calculations. In addition to the two constants introduced at the first level, the second level pseudocode invokes two other constants, which are also assumed to be known a-priori and available for use anywhere in these four functions. First, $\delta\beta$ is the desired resolution in the β parameter, which is here set to the inverse of the number of features: $\delta\beta = \frac{1}{F}$. Second, SIVD_{emp} is the SIVD uncertainty range for the empirical data.

In relation to the first three functions, the descriptions below attempt to mostly emphasize the elements that come in addition with respect to their first-level correspondents. Some of these elements have a repetitive nature and are worth explaining before moving to the specific description of each function. First, the (generic) $\bar{\beta}_{\text{X}}$ notation (where "X" can stand for "L", "R" or "M") denotes the (generic) "composite fitting information" $\bar{\beta}_{\text{X}} = (\beta, \alpha_L, \alpha_R, \alpha_{\text{fit}}, \alpha_{\text{err}})_{\text{X}}$, which is a 5-tuple consisting of a β value together with the associated four values returned by a (generic) call FIT-1(β, k) for that specific β and some arbitrary k. Second, whenever an "SIVD_{X}" variable appears in the first three functions (where "X" is again a generic label), except for SIVD_{emp}, it actually denotes the (generic) "composite SIVD information" $\text{SIVD}_{\text{X}} = ((\text{SIVD}_L^{\text{fit}}, \text{SIVD}_L^{\text{err}}), (\text{SIVD}_R^{\text{fit}}, \text{SIVD}_R^{\text{err}}))_{\text{X}}$, which is a pair of pairs of real numbers, each inner pair corresponding to a model SIVD uncertainty range associated to one margin of an α interval returned by a call to FIT-1, while both inner pairs have the same β. This schematically reads:

$$(\beta, \alpha_L) \rightarrow (\text{SIVD}_L^{\text{fit}}, \text{SIVD}_L^{\text{err}}),$$
$$(\beta, \alpha_R) \rightarrow (\text{SIVD}_R^{\text{fit}}, \text{SIVD}_R^{\text{err}}),$$

[7] The type of information returned by the three functions at a second-level fitting is different than that of the three functions at the first-level fitting, and actually more complex.

Third, any (generic) call NumSIVD$(\bar{\beta}, k)$ is necessarily preceded by an associated (generic) call Fit-1(β, k) and by an associated (generic) variable compression $\bar{\beta} = (\beta, \alpha_L, \alpha_R, \alpha_{\text{fit}}, \alpha_{\text{err}})$, the last two being needed for producing the composite fitting information $\bar{\beta}$. Fourth, whenever a piece of composite SIVD information appears in a call to Ord-2 or Match-2, it is accompanied by an associated piece of composite fitting information, which allows for the mean, statistical error and systematic error of in the model SIVD to be all reconstructed within, for a given combination of β and k.

Function Fit-2 acts as an interface for the second-level fitting, which consists of tuning the β parameter, for a given value of k, such that the SIVD quantity matches the empirical value, relying on an underlying tuning of the α parameter in terms of the AIVD quantity (using Fit-1). Here, k is a strictly positive, integer number. The method returns the composite fitting information associated to the left ($\bar{\beta}_L$) and right ($\bar{\beta}_R$) margins of the tightest β interval found, together with the estimated β match within this interval (β_{fit}) and its associated error (β_{err}). It assumes that the empirical SIVD can actually be uniquely matched by varying β and α, for the given value of k. After checking that there exists a meaningful $[\beta_L, \beta_R]$ interval for which the first-level fitting is possible (lines 2,3), the method conducts the numeric SIVD calculations on both sides of the interval (line 6), preceded, on each side, by the first level fitting and the decompression (lines 4,5, as explained above), in order to finally pass the task to Bisect-2.

1: **function** Fit-2(k)
2: $(\beta_L, \beta_R) =$ Init-2$(\delta\beta, k, \text{AIVD}_{\text{emp}})$ ▷ initializing the β-interval
3: **if** $\beta_L < \beta_R$ **then**
4: $(\alpha_L^L, \alpha_L^R, \alpha_L^{\text{fit}}, \alpha_L^{\text{err}}) =$ Fit-1(β_L, k); $(\alpha_R^L, \alpha_R^R, \alpha_R^{\text{fit}}, \alpha_R^{\text{err}}) =$ Fit-1(β_R, k)
5: $\bar{\beta}_L = (\beta_L, \alpha_L^L, \alpha_L^R, \alpha_L^{\text{fit}}, \alpha_L^{\text{err}})$; $\bar{\beta}_R = (\beta_R, \alpha_R^L, \alpha_R^R, \alpha_R^{\text{fit}}, \alpha_R^{\text{err}})$
6: $\text{SIVD}_L =$ NumSIVD$(\bar{\beta}_L, k)$; $\text{SIVD}_R =$ NumSIVD$(\bar{\beta}_R, k)$ ▷ numerics
7: **return** Bisect-2$(\bar{\beta}_L, \bar{\beta}_R, \text{SIVD}_L, \text{SIVD}_R, k)$
8: **end if**
9: **return FittingImpossibleError**
10: **end function**

Function Bisect-2 is another recursive implementation of the bisection method, which sequentially narrows down the $[\beta_L, \beta_R]$ interval, such that at each stage the empirical SIVD is contained. Here, $\bar{\beta}_L, \bar{\beta}_R$ are 5-tuples of real numbers encoding the left and right pieces of composite fitting information, $\text{SIVD}_L, \text{SIVD}_R$ are the pairs of pairs of real numbers encoding the left-β and right-β pieces of composite SIVD information, while k is of the same type as in Fit-2. It returns the same type of information as Fit-2. Like Bisect-1, the function consists of a part concerned with convergence (lines 4-7), a part concerned with the jump to Displace-2 (lines 11-13) and a part concerned with choosing between the left and right β subintervals and with zooming in on the chosen one (lines 14-19). Note the additional statements concerned with decompressing the composite fit-

ting information (line 2) and with preparing the numeric SIVD calculations at the midpoint (lines 8-9).

1: **function** BISECT-2($\bar{\beta}_L, \bar{\beta}_R, \mathrm{SIVD}_L, \mathrm{SIVD}_R, k$)
2: $(\beta_L, \alpha_L^L, \alpha_L^R, \alpha_L^{\mathrm{fit}}, \alpha_L^{\mathrm{err}}) = \bar{\beta}_L$; $(\beta_R, \alpha_R^L, \alpha_R^R, \alpha_R^{\mathrm{fit}}, \alpha_R^{\mathrm{err}}) = \bar{\beta}_R$
3: $\beta_M = \mathrm{MIDDLE}(\beta_L, \beta_R, \delta\beta)$
4: **if** $\neg$DISTINCT($\beta_M, \beta_L, \beta_R$) **then**
5: $(\beta_{\mathrm{fit}}, \beta_{\mathrm{err}}) = \mathrm{INTERNFITLIN\text{-}2}(\bar{\beta}_L, \bar{\beta}_R, \mathrm{SIVD}_L, \mathrm{SIVD}_R, \mathrm{SIVD}_{\mathrm{emp}})$
6: **return** $(\bar{\beta}_L, \bar{\beta}_R, \beta_{\mathrm{fit}}, \beta_{\mathrm{err}})$
7: **end if**
8: $(\alpha_M^L, \alpha_M^R, \alpha_M^{\mathrm{fit}}, \alpha_M^{\mathrm{err}}) = \mathrm{FIT\text{-}1}(\beta_M, k)$
9: $\bar{\beta}_M = (\beta_M, \alpha_M^L, \alpha_M^R, \alpha_M^{\mathrm{fit}}, \alpha_M^{\mathrm{err}})$
10: $\mathrm{SIVD}_M = \mathrm{NUMSIVD}(\bar{\beta}_M, k)$ ▷ numerics
11: **if** MATCH-2($\bar{\beta}_M, \mathrm{SIVD}_M, \mathrm{SIVD}_{\mathrm{emp}}$) **then**
12: **return** DISPLACE-2($\bar{\beta}_L, \bar{\beta}_R, \bar{\beta}_M, \mathrm{SIVD}_L, \mathrm{SIVD}_R, k$)
13: **end if**
14: **if** ORD-2($\bar{\beta}_L, \bar{\beta}_R, \mathrm{SIVD}_L, \mathrm{SIVD}_R$) = ORD-2($\bar{\beta}_M, \mathrm{SIVD}_M, \mathrm{SIVD}_{\mathrm{emp}}$) **then**
15: $\bar{\beta}_L = \bar{\beta}_M$; $\mathrm{SIVD}_L = \mathrm{SIVD}_M$ ▷ selecting right interval
16: **else**
17: $\bar{\beta}_R = \bar{\beta}_M$; $\mathrm{SIVD}_R = \mathrm{SIVD}_M$ ▷ selecting left interval
18: **end if**
19: **return** BISECT-2($\bar{\beta}_L, \bar{\beta}_R, \mathrm{SIVD}_L, \mathrm{SIVD}_R, k$) ▷ zooming in
20: **end function**

Function DISPLACE-2 attempts to displace the midpoint β_M previously calculated at some stage in BISECT-2, in a way that its associated SIVD uncertainty range does not overlap with the empirical one. This function has all the arguments of BISECT-1 and $\bar{\beta}_M$ as an additional one, which is a 5-tuple of real numbers encoding the midpoint composite fitting information. It returns the same type of information as FIT-2. Like DISPLACE-1, the function consists of a part that attempts a displacement to the left (lines 4-15), one that attempts a displacement to the right (lines 16-27) and one that takes care of the convergence (lines 28-29). Note the additional statements concerned with decompressing the composite fitting information (lines 2-3) and with preparing the numeric SIVD calculations for the left/right secondary midpoint (lines 6-7/18-19).

1: **function** DISPLACE-2($\bar{\beta}_L, \bar{\beta}_R, \bar{\beta}_M, \mathrm{SIVD}_L, \mathrm{SIVD}_R, k$)
2: $(\beta_L, \alpha_L^L, \alpha_L^R, \alpha_L^{\mathrm{fit}}, \alpha_L^{\mathrm{err}}) = \bar{\beta}_L$; $(\beta_R, \alpha_R^L, \alpha_R^R, \alpha_R^{\mathrm{fit}}, \alpha_R^{\mathrm{err}}) = \bar{\beta}_R$;
3: $(\beta_M, \alpha_M^L, \alpha_M^R, \alpha_M^{\mathrm{fit}}, \alpha_M^{\mathrm{err}}) = \bar{\beta}_M$
4: $\beta'_M = \mathrm{MIDDLE}(\beta_L, \beta_M, \delta\beta)$ ▷ trying displacement to the left
5: **if** DISTINCT($\beta'_M, \beta_L, \beta_M$) **then**
6: $(\dot{\alpha}_M^L, \dot{\alpha}_M^R, \dot{\alpha}_M^{\mathrm{fit}}, \dot{\alpha}_M^{\mathrm{err}}) = \mathrm{FIT\text{-}1}(\beta'_M, k)$
7: $\bar{\beta}'_M = (\beta'_M, \dot{\alpha}_M^L, \dot{\alpha}_M^R, \dot{\alpha}_M^{\mathrm{fit}}, \dot{\alpha}_M^{\mathrm{err}})$
8: $\mathrm{SIVD}'_M = \mathrm{NUMSIVD}(\bar{\beta}'_M, k)$ ▷ numerics

```
 9:        if ¬Match-2(β̄′_M, SIVD′_M, SIVD_emp) then
10:            if Ord-2(β̄_L, β̄_R, SIVD_L, SIVD_R) = Ord-2(β̄′_M, SIVD′_M, SIVD_emp)
    then
11:                β̄_L = β̄′_M; SIVD_L = SIVD′_M
12:                return Bisect-2(β̄_L, β̄_R, SIVD_L, SIVD_R, k)        ▷ zooming in
13:            end if
14:        end if
15:    end if
16:    β′_M = Middle(β_M, β_R, δβ)                  ▷ trying displacement to the right
17:    if Distinct(β′_M, β_M, β_R) then
18:        (α̇^L_M, α̇^R_M, α̇^fit_M, α̇^err_M) = Fit-1(β′_M, k)
19:        β̄′_M = (β′_M, α̇^L_M, α̇^R_M, α̇^fit_M, α̇^err_M)
20:        SIVD′_M = NumSIVD(β̄′_M, k)                                    ▷ numerics
21:        if ¬Match-2(β̄′_M, SIVD′_M, SIVD_emp) then
22:            if Ord-2(β̄_L, β̄_R, SIVD_L, SIVD_R) ≠ Ord-2(β̄′_M, SIVD′_M, SIVD_emp)
    then
23:                β̄_R = β̄′_M; SIVD_R = SIVD′_M
24:                return Bisect-2(β̄_L, β̄_R, SIVD_L, SIVD_R, k)        ▷ zooming in
25:            end if
26:        end if
27:    end if
28:    (β_fit, β_err) = InternFitLin-2(β̄_L, β̄_R, SIVD_L, SIVD_R, SIVD_emp)
29:    return (β̄_L, β̄_R, β_fit, β_err)
30: end function
```

Function NumSIVD numerically generates a piece of composite SIVD information with a precision that is as high as possible. Here, $\bar{\beta}$ is a 5-tuple of real numbers encoding a composite fitting information, while k is a positive integer number. One sequence of SIVD values is numerically generated (lines 4-5 and 12-13) for each of the two margins of the α interval (contained in $\bar{\beta}$), for the given β (also contained in $\bar{\beta}$) and the given k. An uncertainty range is obtained from each of the two sequences (lines 6 and 16). These two uncertainty ranges are used together with the information in $\bar{\beta}$ to produce estimates for an average, a statistical error and a systematic error that are $\bar{\beta}$-specific rather than (α, β)-specific (lines 7,8 and 17,18). The number of SIVD values in the two sequences is increased and the calculations are repeated as long as the condition in line 10 remains true, namely as long as: the statistical error is higher than the systematic error, the desired separation between the model and empirical (statistical) uncertainty ranges is not reached and the maximal SIVD sequence length is not reached. The desired separation and the SIVD sequence length are controlled via variables s and n, initialized in line 2 – the initial values of these variables, as well as the upper bound on the latter are hard-coded, as visible in the pseudocode, and have been decided after some experimentation with NumSIVD, but they are not essential for the actual outcome. Also note the decompression of the composite

fitting information (line 3) and the decompression of SIVD uncertainty ranges (lines 9 and 19).

1: **function** NumSIVD($\bar{\beta}, k$)
2: $n = 20$; $s = 5$ ▷ initial number of realizations and desired separation
3: $(\beta, \alpha_L, \alpha_R, \alpha_{\text{fit}}, \alpha_{\text{err}}) = \bar{\beta}$
4: $\text{SIVD}_L^{\text{seq}} = \text{GenSeqSIVD}(\alpha_L, \beta, k, n)$
5: $\text{SIVD}_R^{\text{seq}} = \text{GenSeqSIVD}(\alpha_R, \beta, k, n)$
6: $\text{SIVD}_L = \text{CompAvgErr}(\text{SIVD}_L^{\text{seq}})$; $\text{SIVD}_R = \text{CompAvgErr}(\text{SIVD}_R^{\text{seq}})$
7: $\text{SIVD} = \text{Interpol}(\alpha_L, \alpha_R, \alpha_{\text{fit}}, \text{SIVD}_L, \text{SIVD}_R)$
8: $\text{SIVD}_{\text{syst}} = \text{CompSystErr}(\alpha_L, \alpha_R, \alpha_{\text{err}}, \text{SIVD}_L, \text{SIVD}_R)$
9: $(\text{SIVD}_{\text{avg}}, \text{SIVD}_{\text{stat}}) = \text{SIVD}$; $(\text{SIVD}_{\text{emp}}^{\text{avg}}, \text{SIVD}_{\text{emp}}^{\text{stat}}) = \text{SIVD}_{\text{emp}}$
10: **while** $\text{SIVD}_{\text{stat}} > \text{SIVD}_{\text{syst}} \wedge (\text{SIVD}_{\text{stat}} + \text{SIVD}_{\text{emp}}^{\text{stat}} > |\text{SIVD}_{\text{emp}}^{\text{avg}} - \text{SIVD}_{\text{avg}}|/s) \wedge n < 350$ **do**
11: $n = 2 \cdot n$
12: $\text{SIVD}_L^{\text{tmpSeq}} = \text{GenSeqSIVD}(\alpha_L, \beta, k, n)$
13: $\text{SIVD}_R^{\text{tmpSeq}} = \text{GenSeqSIVD}(\alpha_R, \beta, k, n)$
14: $\text{SIVD}_L^{\text{seq}} = \text{Merge}(\text{SIVD}_L^{\text{seq}}, \text{SIVD}_L^{\text{tmpSeq}})$
15: $\text{SIVD}_R^{\text{seq}} = \text{Merge}(\text{SIVD}_R^{\text{seq}}, \text{SIVD}_R^{\text{tmpSeq}})$
16: $\text{SIVD}_L = \text{CompAvgErr}(\text{SIVD}_L^{\text{seq}})$; $\text{SIVD}_R = \text{CompAvgErr}(\text{SIVD}_R^{\text{seq}})$
17: $\text{SIVD} = \text{Interpol}(\alpha_L, \alpha_R, \alpha_{\text{fit}}, \text{SIVD}_L, \text{SIVD}_R)$
18: $\text{SIVD}_{\text{syst}} = \text{CompSystErr}(\alpha_L, \alpha_R, \alpha_{\text{err}}, \text{SIVD}_L, \text{SIVD}_R)$
19: $(\text{SIVD}_{\text{avg}}, \text{SIVD}_{\text{stat}}) = \text{SIVD}$
20: **end while**
21: **return** $(\text{SIVD}_L, \text{SIVD}_R)$
22: **end function**

2.C.3 Used functions

This section describes functions that are used by the pseudocode in sections 2.C.1 or 2.C.2 but are not described there. The following is a list of functions for which the pseudocode is also provided, following each text description.

Function InterfitLin-1 fine-tunes the α parameter such that AIVD_{emp} is matched, relying on a linear approximation of the model AIVD as a function of α within the (α_L, α_R) interval, using the boundary values AIVD_L and AIVD_R. Its arguments are of the same type as those of InterFitLin (described below), except that AIVD_L and AIVD_R are real numbers rather than uncertainty ranges. The output structure is entirely the same as that of InterFitLin. It is essentially a first-level fitting interface for InternFitLin, which is called after specifying that the errors associated to AIVD_L and AIVD_R are zero.

1: **function** InternFitLin-1($\alpha_L, \alpha_R, \text{AIVD}_L, \text{AIVD}_R, \text{AIVD}_{\text{emp}}$)
2: $\text{AIVD}'_L = (\text{AIVD}_L, 0)$; $\text{AIVD}'_R = (\text{AIVD}_R, 0)$

3: **return** InternFitLin($\alpha_L, \alpha_R, \text{AIVD}'_L, \text{AIVD}'_R, \text{AIVD}_{\text{emp}}$)
4: **end function**

Function InterfitLin-2 fine-tunes the β parameter such that SIVD_{emp} is matched, relying on a linear approximation of the model SIVD as a function of β within the $[\beta_L, \beta_R]$ interval, using the boundary information stored in SIVD_L and SIVD_R. Its arguments are of the same type as those of InterFitLin (described below), except that $\bar{\beta}_L$ and $\bar{\beta}_R$ are 5-tuples or real numbers rather than real numbers and SIVD_L and SIVD_R are pieces composite SIVD information rather than uncertainty ranges. The output structure is entirely the same as that of InterFitLin. It is essentially a second-level fitting interface for InternFitLin, which is called after carrying out the following two operations: computing the mean, statistical error and systematic error on each of the two margins of the β interval, using the right combination of composite fitting information and composite SIVD information (lines 2,3); compressing information into an SIVD uncertainty range for each of the two margins, after choosing the highest among the two errors for each margin.

1: **function** InternFitLin-2($\bar{\beta}_L, \bar{\beta}_R, \text{SIVD}_L, \text{SIVD}_R, \text{SIVD}_{\text{emp}}$)
2: $(\text{SIVD}_L^{\text{avg}}, \text{SIVD}_L^{\text{stat}}, \text{SIVD}_L^{\text{syst}}) = \text{MeanStatSyst}(\bar{\beta}_L, \text{SIVD}_L)$
3: $(\text{SIVD}_R^{\text{avg}}, \text{SIVD}_R^{\text{stat}}, \text{SIVD}_R^{\text{syst}}) = \text{MeanStatSyst}(\bar{\beta}_R, \text{SIVD}_R)$
4: $\text{SIVD}'_L = (\text{SIVD}_L^{\text{avg}}, \text{max}(\text{SIVD}_L^{\text{stat}}, \text{SIVD}_L^{\text{syst}}))$
5: $\text{SIVD}'_R = (\text{SIVD}_R^{\text{avg}}, \text{max}(\text{SIVD}_R^{\text{stat}}, \text{SIVD}_R^{\text{syst}}))$
6: **return** InternFitLin($\beta_L, \beta_R, \text{SIVD}'_L, \text{SIVD}'_R, \text{SIVD}_{\text{emp}}$)
7: **end function**

Function Match-1 checks whether AIVD (real value) falls within the uncertainty range specified by AIVD_{emp}. It acts as an interface for Match (described below) within the first-level fitting scheme.

1: **function** Match-1($\text{AIVD}, \text{AIVD}_{\text{emp}}$)
2: $\text{AIVD}' = (\text{AIVD}, 0)$
3: **return** Match($\text{AIVD}', \text{AIVD}_{\text{emp}}$)
4: **end function**

Function Match-2 checks whether there is an overlap between the model SIVD uncertainty range obtained from $\bar{\beta}$ (composite fitting information) and SIVD (composite SIVD information) and the empirical one encoded by SIVD_{emp}. It acts as an interface for Match (described below) within the second-level fitting scheme.

1: **function** Match-2($\bar{\beta}, \text{SIVD}, \text{SIVD}_{\text{emp}}$)
2: $(\text{SIVD}_{\text{avg}}, \text{SIVD}_{\text{stat}}, \text{SIVD}_{\text{syst}}) = \text{MeanStatSyst}(\bar{\beta}, \text{SIVD})$
3: $\text{SIVD}' = (\text{SIVD}_{\text{avg}}, \text{max}(\text{SIVD}_{\text{stat}}, \text{SIVD}_{\text{syst}}))$
4: **return** Match($\text{SIVD}', \text{SIVD}_{\text{emp}}$)

5: **end function**

Function ORD-1 (first version) checks whether AIVD_L (real value) is smaller than AIVD_R (real value), acting as an interface for ORD within the first-level fitting scheme.

1: **function** ORD-1($\mathrm{AIVD}_L, \mathrm{AIVD}_R$)
2: **return** ORD($\mathrm{AIVD}_L, \mathrm{AIVD}_R$)
3: **end function**

Function ORD-1 (second version) checks whether AIVD (real value) is smaller than the average stored in $\mathrm{AIVD}_{\mathrm{emp}}$ (uncertainty range), acting as an interface for ORD within the first-level fitting scheme.

1: **function** ORD-1($\mathrm{AIVD}, \mathrm{AIVD}_{\mathrm{emp}}$)
2: $(\mathrm{AIVD}^{\mathrm{avg}}_{\mathrm{emp}}, \mathrm{AIVD}^{\mathrm{err}}_{\mathrm{emp}}) = \mathrm{AIVD}_{\mathrm{emp}}$
3: **return** ORD($\mathrm{AIVD}, \mathrm{AIVD}^{\mathrm{avg}}_{\mathrm{emp}}$)
4: **end function**

Function ORD-2 (first version) checks whether the average stored in the SIVD uncertainty range obtained from $\bar{\beta}_L$ (composite fitting information) and SIVD_L (composite SIVD information) is smaller than the average stored in that obtained from $\bar{\beta}_R$ (composite fitting information) and SIVD_R (composite SIVD information), acting as an interface for ORD within the second-level fitting scheme.

1: **function** ORD-2($\bar{\beta}_L, \bar{\beta}_R, \mathrm{SIVD}_L, \mathrm{SIVD}_R$)
2: $(\mathrm{SIVD}^{\mathrm{avg}}_L, \mathrm{SIVD}^{\mathrm{stat}}_L, \mathrm{SIVD}^{\mathrm{syst}}_L) = \textsc{MeanStatSyst}(\bar{\beta}_L, \mathrm{SIVD}_L)$
3: $(\mathrm{SIVD}^{\mathrm{avg}}_R, \mathrm{SIVD}^{\mathrm{stat}}_R, \mathrm{SIVD}^{\mathrm{syst}}_R) = \textsc{MeanStatSyst}(\bar{\beta}_R, \mathrm{SIVD}_R)$
4: **return** ORD($\mathrm{SIVD}^{\mathrm{avg}}_L, \mathrm{SIVD}^{\mathrm{avg}}_R$)
5: **end function**

Function ORD-2 (second version) checks whether the average stored in the SIVD uncertainty range obtained from $\bar{\beta}$ (composite fitting information) and SIVD (composite SIVD information) is smaller than the average stored $\mathrm{SIVD}_{\mathrm{emp}}$, acting as an interface for ORD within the second-level fitting scheme.

1: **function** ORD-2($\bar{\beta}, \mathrm{SIVD}, \mathrm{SIVD}_{\mathrm{emp}}$)
2: $(\mathrm{SIVD}_{\mathrm{avg}}, \mathrm{SIVD}_{\mathrm{stat}}, \mathrm{SIVD}_{\mathrm{syst}}) = \textsc{MeanStatSyst}(\bar{\beta}, \mathrm{SIVD})$
3: $(\mathrm{AIVD}^{\mathrm{avg}}_{\mathrm{emp}}, \mathrm{AIVD}^{\mathrm{err}}_{\mathrm{emp}}) = \mathrm{AIVD}_{\mathrm{emp}}$
4: **return** ORD($\mathrm{SIVD}_{\mathrm{avg}}, \mathrm{AIVD}^{\mathrm{avg}}_{\mathrm{emp}}$)
5: **end function**

Function MEANSTATSYST estimates a mean, a statistical error and a systematic error from a piece of composite fitting information and an associated piece of composite SIVD information, which are the two arguments of the function. It returns the 3-tuple comprising of the three computed real numbers. Note the decompression of composite fitting information (line 2) and the decompression of

composite SIVD information (line 3).

1: **function** MeanStatSyst($\bar{\beta}$, SIVD)
2: $(\beta, \alpha_L, \alpha_R, \alpha_{\text{fit}}, \alpha_{\text{err}}) = \bar{\beta}$
3: $(\text{SIVD}_L, \text{SIVD}_R) = \text{SIVD}$
4: $\text{SIVD}' = \text{Interpol}(\alpha_L, \alpha_R, \alpha_{\text{fit}}, \text{SIVD}_L, \text{SIVD}_R)$
5: $\text{SIVD}_{\text{syst}} = \text{CompSystErr}(\alpha_L, \alpha_R, \alpha_{\text{err}}, \text{SIVD}_L, \text{SIVD}_R)$
6: $(\text{SIVD}_{\text{avg}}, \text{SIVD}_{\text{stat}}) = \text{SIVD}'$
7: **return** $(\text{SIVD}_{\text{avg}}, \text{SIVD}_{\text{stat}}, \text{SIVD}_{\text{syst}})$
8: **end function**

The following is a list of functions for which only text explanations are provided in schematic way, sometimes accompanied by figures.

- Init-1($\delta\alpha$):
  - gives the left and right boundaries of the largest possible interval for which the α parameter is compatible with the stochastic model in use, given the grid spacing $\delta\alpha$
  - input: δ is a real number
  - in practice it returns $(\delta\alpha, 1 - \delta\alpha)$ regardless of whether PG or MPG is used
- Init-2($\delta\beta, k, \text{AIVD}_{\text{emp}}$):
  - gives the left and right boundaries of the largest possible interval, if any, for which the β parameter allows for the (first level) fitting of AIVD(α) to successfully take place, given the grid spacing $\delta\beta$
  - input: $\delta\beta$ is a real number, k is a positive integer and AIVD_{emp} is an uncertainty range
  - assumes that there exists at most one β interval $[\beta_L, \beta_R]$ for which there exists an α such that $\langle \text{AIVD} \rangle^k_{\alpha,\beta} = \text{AIVD}_{\text{emp}}$ is satisfied
  - starts from the largest interval allowed by the model and independently adjusts each of the two boundaries via a branching algorithm, until the desired interval is reached
  - returns two (incompatible) boundaries $\beta_L > \beta_R$ if such an interval does not exist
- Middle(l, r, δ):
  - computes the value closest to the average between l and r, on a grid of spacing δ
  - input: l, r, δ are all real numbers
  - assumes that the interval length $l - r$ is equal to an integer times δ

- DISTINCT(m, l, r):
  - checks whether m is different than both l and r
  - input: m, l, r are all real numbers constrained constrained to a grid of constant spacing
- INTERNFITLIN$(p_L, p_R, O_L, O_R, O_{\text{emp}})$:
  - adjusts a parameter p such that an observable O attains a value compatible with the empirical in O_{emp} interval, assuming that O is a linear function of p within the $[p_L, p_R]$ interval
  - input: p_L, p_R are real numbers, encoding the left and right boundaries of the interval; O_L, O_R are mean-error pairs of real numbers encoding the theoretical uncertainty ranges of the observable for the left and for the right boundaries; O_{emp} is a mean-error pair of real numbers encoding the empirical uncertainty range
  - returns the value and associated error of the p parameter resulting from this fitting process $(p_{\text{fit}}, p_{\text{err}})$, computed based on geometrical considerations, in the manner illustrated in Fig. 2.7(a)
  - p_{fit} is calculated first by intersecting the theoretical line with the empirical one, disregarding all errors; then, p_{err} is calculated by assuming that the theoretical error is constant within the $[p_L, p_R]$ interval, with value given by interpolating the errors contained by O_L and O_R at p_{fit}
  - p_{err} takes its origin both in the the empirical error as well as in the theoretical error, but also depends on the slope resulting from the linear approximation
- MATCH(r_1, r_2):
  - checks whether there is an overlap between the uncertainty ranges encoded by r_1 and r_2
  - input: r_1, r_2 are mean-error pairs of real numbers
- ORD(v_L, v_R):
  - checks whether the condition $v_L < v_R$ is satisfied
  - input: v_L, v_R are real numbers
  - assumes that $v_L \neq v_R$
- GENSEQSIVD(α, β, k, n)
  - numerically generates a sequence of n SIVD values according to the respective stochastic model, subject to parameter values indicated by k, α, β

  - input: α, β are real numbers, while k, n are positive integers

- Merge($\text{SIVD}_1^{\text{seq}}, \text{SIVD}_2^{\text{seq}}$)
  - merges two sequences of (real) SIVD values
  - input: $\text{SIVD}_1^{\text{seq}}, \text{SIVD}_2^{\text{seq}}$ are both sequences of (real) SIVD values
- CompAvgErr(SIVD^{seq})
  - computes the mean and standard error of the mean from SIVD^{seq}
  - input: SIVD^{seq} is a sequence of real SIVD values
- Interpol($\alpha_L, \alpha_R, \alpha_{\text{fit}}, \text{SIVD}_L, \text{SIVD}_R$)
  - estimates the mean and error in SIVD corresponding to α_{fit} based on the values attained for α_L
  - input: $\alpha_L, \alpha_R, \alpha_{\text{fit}}$ are real numbers, while $\text{SIVD}_L, \text{SIVD}_R$ are mean-error pairs of real numbers
  - uses on a linear interpolation within the $[\alpha_L, \alpha_R]$ interval, separately for the mean and for the error
- CompSystErr($\alpha_L, \alpha_R, \alpha_{\text{err}}, \text{SIVD}_L, \text{SIVD}_R$)
  - estimates the systematic error $\text{SIVD}^{\text{syst}}$ of the SIVD quantity induced by the error α_{err} (associated to fitting the α parameter in terms of the AIVD quantity), assuming that SIVD is a linear function of α within the $[\alpha_L, \alpha_R]$ interval
  - input: $\alpha_L, \alpha_R, \alpha_{\text{err}}$ are real numbers while while $\text{SIVD}_L, \text{SIVD}_R$ are mean-error pairs of real numbers encoding the theoretical uncertainty ranges on the left and right boundaries
  - $\text{SIVD}^{\text{syst}}$ is computed based on geometrical considerations, in the manner illustrated in Fig. 2.7(b)

2.C.4 Algorithm usage

This section explain how the formalism presented throughout this document is effectively used for producing the results shown in Sections 2.3 and 2.4.

First, the formalism is used for producing the plots showing the SIVD(β) dependence ("Model fitting" section). For either PG or MPG, for a specific k value and a specific β on-grid value, the drawn model SIVD uncertainty range (Fig. 2.3) is obtained after the following computational steps:

1: $(\alpha_L, \alpha_R, \alpha_{\text{fit}}, \alpha_{\text{err}}) = \text{Fit-1}(\beta, k)$ ▷ executing 1st-level fitting
2: $\bar{\beta} = (\beta, \alpha_L, \alpha_R, \alpha_{\text{fit}}, \alpha_{\text{err}})$ ▷ creating composite fitting information

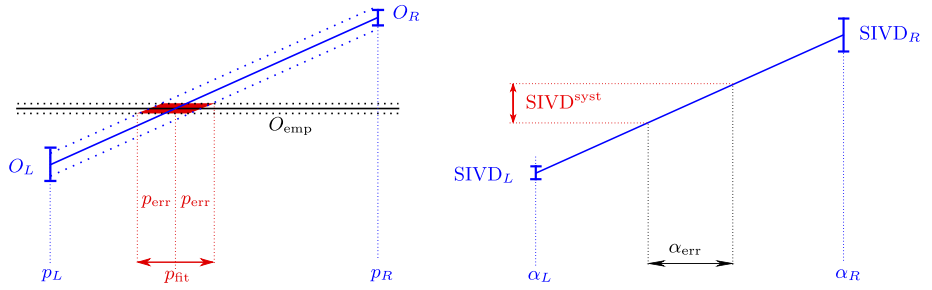

Figure 2.7: Illustration of computation carried out by InterFitLin (a) and by CompSystErr (b), with the output quantities highlighted in red.

3: $\text{SIVD} = \text{NumSIVD}(\bar{\beta}, k)$ ▷ numeric SIVD calculations
4: $(\text{SIVD}_{\text{avg}}, \text{SIVD}_{\text{stat}}, \text{SIVD}_{\text{syst}}) = \text{MeanStatSyst}(\bar{\beta}, \text{SIVD})$

which provides the values of the SIVD average SIVD_{avg}, the SIVD statistical error $\text{SIVD}_{\text{stat}}$ and the SIVD systematic error $\text{SIVD}_{\text{syst}}$. One can then place a point at coordinates $(\beta, \text{SIVD}_{\text{avg}})$, within the respective k curve, with an error bar given by the maximum between $\text{SIVD}_{\text{stat}}$ and $\text{SIVD}_{\text{syst}}$.

Second, the formalism is used for providing the best-fitting, on-grid values for the α and β model parameters, which are used for generating sets of cultural vectors on which the LTCD-STCB analysis is applied (Sec. 2.4). For either PG or MPG and for a specific k value, the following procedure is followed:

1: $(\bar{\beta}_L, \bar{\beta}_R, \beta_{\text{fit}}, \beta_{\text{err}}) = \text{Fit-2}(k)$ ▷ Executing 2nd-level fitting
2: $(\beta_L, \alpha_L^L, \alpha_L^R, \alpha_L^{\text{fit}}, \alpha_L^{\text{err}}) = \bar{\beta}_L$ ▷ Decompressing left-β composite fitting information
3: $(\beta_R, \alpha_R^L, \alpha_R^R, \alpha_R^{\text{fit}}, \alpha_R^{\text{err}}) = \bar{\beta}_R$ ▷ Decompressing right-β composite fitting information
4: **if** $\beta_{\text{fit}} - \beta_L < \beta_R - \beta_{\text{fit}}$ **then**
5: $\beta = \beta_L$ ▷ choosing β_L, since it is closer to β
6: **if** $\alpha_L^{\text{fit}} - \alpha_L^L < \alpha_L^R - \alpha_L^{\text{fit}}$ **then**
7: $\alpha = \alpha_L^L$ ▷ choosing α_L^L, since it is closer to α_L^{fit}
8: **else**
9: $\alpha = \alpha_L^R$ ▷ choosing α_L^R, since it is closer to α_L^{fit}
10: **end if**
11: **else**
12: $\beta = \beta_R$ ▷ choosing β_R, since it is closer to β
13: **if** $\alpha_R^{\text{fit}} - \alpha_R^L < \alpha_R^R - \alpha_R^{\text{fit}}$ **then**
14: $\alpha = \alpha_R^L$ ▷ choosing α_R^L, since it is closer to α_R^{fit}
15: **else**

16: $\alpha = \alpha_R^R$ ▷ choosing α_R^R, since it is closer to α_R^{fit}
17: **end if**
18: **end if**

which provides the best on-grid values for the (α, β) pair.

Bibliography

[1] Claudio Castellano, Santo Fortunato, and Vittorio Loreto. Statistical physics of social dynamics. *Rev. Mod. Phys.*, 81:591–646, May 2009.

[2] Robert Axelrod. The dissemination of culture. *Journal of Conflict Resolution*, 41(2):203–226, 1997.

[3] Konstantin Klemm, Víctor M. Eguíluz, Raúl Toral, and Maxi San Miguel. Global culture: A noise-induced transition in finite systems. *Phys. Rev. E*, 67:045101, Apr 2003.

[4] Konstantin Klemm, Víctor M. Eguíluz, Raúl Toral, and Maxi San Miguel. Nonequilibrium transitions in complex networks: A model of social interaction. *Phys. Rev. E*, 67:026120, Feb 2003.

[5] Marcelo N. Kuperman. Cultural propagation on social networks. *Phys. Rev. E*, 73:046139, Apr 2006.

[6] Andreas Flache and Michael W. Macy. Local convergence and global diversity: The robustness of cultural homophily. *arXiv:physics/0701333*, 2007.

[7] J. C. González-Avella, M. G. Cosenza, and K. Tucci. Nonequilibrium transition induced by mass media in a model for social influence. *Phys. Rev. E*, 72:065102, Dec 2005.

[8] Damon Centola, Juan Carlos González-Avella, Víctor M. Eguíluz, and Maxi San Miguel. Homophily, cultural drift, and the co-evolution of cultural groups. *Journal of Conflict Resolution*, 51(6):905–929, 2007.

[9] Jens Pfau, Michael Kirley, and Yoshihisa Kashima. The co-evolution of cultures, social network communities, and agent locations in an extension of Axelrod's model of cultural dissemination . *Physica A: Statistical Mechanics and its Applications*, 392(2):381–391, 2013.

[10] Federico Battiston, Vincenzo Nicosia, Vito Latora, and Maxi San Miguel. Robust multiculturality emerges from layered social influence. *Sci. Rep.*, 7(1809), 2017.

[11] Alex Stivala, Yoshihisa Kashima, and Michael Kirley. Culture and cooperation in a spatial public goods game. *Phys. Rev. E*, 94:032303, Sep 2016.

[12] Luca Valori, Francesco Picciolo, Agnes Allansdottir, and Diego Garlaschelli. Reconciling long-term cultural diversity and short-term collective social behavior. *Proc. Natl. Acad. Sci.*, 109(4):1068–1073, 2012.

[13] Alex Stivala, Garry Robins, Yoshihisa Kashima, and Michael Kirley. Ultrametric distribution of culture vectors in an extended Axelrod model of cultural dissemination. *Sci. Rep.*, 4(4870), 2014.

[14] Alexandru-Ionuţ Băbeanu, Leandros Talman, and Diego Garlaschelli. Signs of universality in the structure of culture. *The European Physical Journal B*, 90(12):237, Dec 2017.

[15] Rama Cont and Jean-Philipe Bouchaud. Herd behavior and aggregate fluctuations in financial markets. *Macroeconomic Dynamics*, 4(02):170–196, 2000.

[16] Michael Thompson, Richard J. Ellis, and Aaron Wildavsky. *Cultural Theory*. Westview Press, 1990.

[17] Alan Page Fiske. The four elementary forms of sociality: framework for a unified theory of social relations. *Psychol Rev*, 99(4):689–723, 1992.

[18] Harry C. Triandis. The psychological measurement of cultural syndromes. *Am Psychol*, 51(4):407–415, 1996.

[19] Richard A. Shweder, Nancy C. Much, Manamohan Mahapatra, and Lawrence Park. The "big three" of morality (autonomy, community, divinity) and the "big three" explanations of suffering. In Allan M. Brandt and Paul Rozin, editors, *Morality and Health*, pages 119–169. Routledge, New York, 1997.

[20] Jesse Graham, Brian A. Nosek, Jonathan Haidt, Ravi Iyer, Spassena Koleva, and Peter H. Ditto. Mapping the moral domain. *J Pers Soc Psychol*, 101(2):366–385, 2011.

[21] Marco Verweij. Towards a theory of constrained relativism: Comparing and combining the work of pierre bourdieu, mary douglas and michael thompson, and alan fiske. *Sociological Research Online*, 12:1–15, 2007.

[22] Joshua R. Bruce. Uniting theories of morality, religion, and social interaction: Grid-group cultural theory, the "big three" ethics, and moral foundations theory. *Psychology and Society*, 5:37–50, 2013.

[23] Marco Verweij, Marieke van Egmond, Ulrich Kühnen, Shenghua Luan, Steven Ney, and M. Aenne Schoop. I disagree, therefore i am: how to test and strengthen cultural versatility. *Innovation: The European Journal of Social Science Research*, 27:1–12, 2014.

[24] Steve Rayner. Cultural theory and risk analysis. In S. Krimsky and D. Golding, editors, *Social Theories of Risk*, pages 83–115. Praeger, 1992.

[25] James Tansey and Steve Rayner. Cultural theory and risk. In Robert L. Heath and H. Dan O'Hair, editors, *Handbook of Risk and Crisis Communication*, pages 53–79. Routledge, New York, 2009.

[26] Julia Poncela-Casasnovas, Mario Gutiérrez-Roig, Carlos Gracia-Lázaro, Julian Vicens, Jesús Gómez-Gardeñes, Josep Perelló, Yamir Moreno, Jordi Duch, and Angel Sánchez. Humans display a reduced set of consistent behavioral phenotypes in dyadic games. *Science Advances*, 2(8), 2016.

[27] Jon Elster. *The Multiple Self*. Cambridge University Press, 1985.

[28] Ernst Fehr and Karla Hoff. Introduction: Tastes, castes and culture: the influence of society on preferences. *The Economic Journal*, 121(556):F396–F412, 2011.

[29] Alain Cohn, Ernst Fehr, and M. A. Marechal. Business culture and dishonesty in the banking industry. *Nature*, 516(7529):86–89, 2014.

[30] A. Cohn, J Engelmann, E. Fehr, and Michael André. Maréchal. Evidence for countercyclical risk aversion: An experiment with financial professionals. *American Economic Journal*, 105(2):860–85, 2015.

[31] Shalom H. Schwartz. Universals in the content and structure of values: Theoretical advances and empirical tests in 20 countries. volume 25 of *Advances in Experimental Social Psychology*, pages 1 – 65. Academic Press, 1992.

[32] Richard Nisbett. *The Multiple Self*. The Free Press, 2003.

[33] Ulrich Kühnen and Daphna Oyserman. Thinking about the self influences thinking in general: cognitive consequences of salient self-concept. *Journal of Experimental Social Psychology*, 38(5):492 – 499, 2002.

[34] Karlheinz Reif and Anna Melich. Euro-barometer 38.1: Consumer protection and perceptions of science and technology, november 1992. `http://www.icpsr.umich.edu/icpsrweb/ICPSR/studies/06045`, 1995.

[35] Jonathan L. Gross and Steve Rayner. *Measuring Culture: A Paradigm for the Analysis of Social Organization*. Columbia Univ Press, 1985.

[36] Maroussia Favre and Didier Sornette. A generic model of dyadic social relationships. *PLoS ONE*, 10(3):1–16, 2015.

[37] Vudtiwat Ngampruetikorn and Greg J. Stephens. Bias, belief, and consensus: Collective opinion formation on fluctuating networks. *Phys. Rev. E*, 94:052312, Nov 2016.

[38] Peter Taylor. Algorithm for generating integer partitions. `http://math.stackexchange.com/questions/18659/algorithm-for-generating-integer-partitions`, 01 2011.