The softmax function: Properties, motivation, and interpretation[*]

Michael Franke & Judith Degen

Abstract

The softmax function is a ubiquitous helper function, frequently used as a probabilistic link function for unordered categorical data, in different kinds of models, such as regression, artificial neural networks, or probabilistic cognitive models. To fully understand the models in which the softmax function occurs, different levels of understanding of the softmax function itself are necessary. For input-output oriented models, like regression or neural network models, mathematical properties are crucial. For models with interpretable and meaningful internal representations like probabilistic cognitive models, we also require a thorough conceptual understanding of the motivation for using the softmax function (instead of something else). This tutorial provides an in-depth exposition of the informal, mathematical and conceptual properties of the softmax function. It also provides two mathematical derivations (as a stochastic choice model, and as maximum entropy distribution), together with three conceptual interpretations that can serve as rationale for using the softmax function in models that require explainability of modeling choices.

1 Introduction & overview

The softmax function is a ubiquitous helper function, frequently used as a probabilistic link function for unordered categorical data. Within the Cognitive Sciences, it is commonly used in the context of neural networks, regression modeling, or probabilistic cognitive models. Despite its prevalence, there is often little concern about the rationale behind the softmax function or its mathematical properties. As a results, the models in which the softmax function occurs may remain partially opaque. This is perhaps less problematic, if unfortunate, for some areas of application (neural network or regression modeling), but insufficient understanding of the softmax function can be a genuine problem in other areas of application (such as cognitive modeling), as we will explain here.

The goal of this tutorial is to describe the softmax function in increasing level of conceptual and mathematical detail, so as to enable a better understanding of the models in which it occurs. The tutorial is structured so that readers with different demands can exit when their information needs are satisfied. We recommend that all readers start with Section 2, which gives definitions and formal notation for the rest of the paper. It is also highly recommended to at least skim Section 3, which introduces a distinction between *input-output (IO) models* and *internally-meaningful (IM) models*. Depending on the kind of model readers are interested in, they may want to focus on different parts of this tutorial. Readers mostly interested in IO models may want to solidify their understanding of the softmax function in applied contexts, like regression modeling or neural networks, and therefore find useful information in Section 4, which illustrates the workings of the softmax function based on input-output relations (using plots and intuitions), and Section 5, which adds mathematical detail by stating and proving key properties of the softmax function. For those readers working with (IO

[*] Thanks to Juliane Schwab for support in creating this manuscript, to Georg Schröter for pointing out the connection to thermodynamics and to Britta Grusdt, Xenia Ohmer, Polina Tsvilodub, and Fausto Carcassi for discussion and feedback.

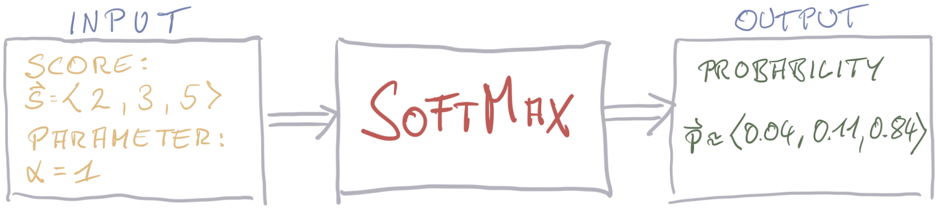
INPUT
SCORE:
$\vec{s}=\langle 2,3,5\rangle$
PARAMETER:
$\alpha=1$
SoftMax
OUTPUT
PROBABILITY
$\vec{p}\approx\langle 0.04, 0.11, 0.84\rangle$

Figure 1: The softmax function takes a score vector $\mathbf{s}$ and a value of the softmax-parameter α as input and returns a probability vector.

or IM) models which have an explicit and variable parameter, Section 6 provides some guidance on the interpretation of specific numerical values of the softmax parameter, which is also important to choosing priors when working with Bayesian models. Finally, readers who encounter the softmax function as a stochastic choice function in IM models, e.g., as a stochastic choice function in abstract models of decision-making in the context of cognitive models, psycholinguistics, computational sociology, or behavioral economics, may also care about the conceptual motivation for this modeling choice. To address this, this tutorial gives three different conceptual motivations (with accompanying mathematical derivations) for the softmax function. Concretely, Section 7 characterizes the softmax function as modeling sub-optimal, noise-perturbed decision making, where stochasticity enters due to one particular kind of stochastic error. Section 8 motivates the softmax function as an optimal choice of modellers who want to assume that choices are noise-perturbed, but want to remain maximally uncommitted about the nature of the noise. This interpretation is based on a formal result that shows that the softmax distribution is a maximum-entropy distribution. Finally, Section 9 interprets the same mathematical result in a different way, namely as an optimal tradeoff between exploration and exploitation from a decision-making agent's perspective.

2 Softmax basics: definition, notation & terminology

Formally, the softmax function is a mapping that takes a vector of *scores* $\mathbf{s} = \langle s_1, \dots, s_n \rangle$ and maps it to a vector of corresponding probabilities $\mathbf{p} = \langle p_1, \dots, p_n \rangle$, using the *softmax (optimality) parameter* α to modulate the shape of the output distribution $\mathbf{p}$ (see Figure 1). The common use case for the softmax function is to serve as a link function for unordered, categorical data. Given n discrete *outcome categories* $\mathbf{x} = \langle x_1, \dots, x_n \rangle$, we want a model that predicts how likely each category x_i is. In applications, outcome categories could be labels (Does this picture show a dog, a cat or a goldfish? Is this text fiction, law, or science?) or actions of an agent (Will Bubu order pizza, pasta or salad? How likely are participants to select option x_i in a forced-choice experiment?). A model would then be used to predict the outcome probabilities p_i for each category x_i.

A probability distribution $\mathbf{p} = \langle p_1, \dots, p_n \rangle$ over n (finite) categories must sum up to one: $\sum_i p_i = 1$ and requires that $0 \leq p_i \leq 1$ for all $1 \leq i \leq n$. Yet, for technical or conceptual reasons, it is often easier for a computational model to create a probabilistic prediction $\mathbf{p}$ by first, internally, constructing a less constrained vector of scores $\mathbf{s} = \langle s_1, \dots, s_n \rangle$ which do not necessarily sum up to one and where s_i can be any finite real number (positive or negative).[1] The main use case of the softmax function is then to map these non-normalized scores onto well-behaved probabilities. Section 3 will elaborate on the interpretation of scores, which depends on the kind of application or model at hand.

[1] In some context, such as in machine learning, the scores are sometimes referred to as *logits*.

The definition of the softmax function is:[2]

$$\mathrm{SM}(\mathbf{s}; \alpha) = \mathbf{p}, \quad \text{with: } p_i = \frac{\exp(\alpha\ s_i)}{\sum_j \exp(\alpha\ s_j)}$$

Authors often use a simpler notation, omitting the normalizing constant $Z = \sum_j \exp(\alpha\ s_j)$, to just write:

$$p_i \propto \exp(\alpha\ s_i)$$

The softmax function has a *softmax parameter* $\alpha \in \mathbb{R}$, which is sometimes omitted, i.e., implicitly set to 1.[3] The softmax parameter α modulates the output probability in systematic ways. Intuitively, the higher α, the more preferred (in terms of higher probability) will be options with higher scores. (Section 4 explores the effects of α on the output of the softmax function, Section 6 focuses on the interpretation of particular values of α.) But whether we need α in the first place, depends on the interpretation of the scores and the kind of model we have, which is the topic of the next Section 3.

3 Why care about the softmax function?

The softmax function is only one of infinitely many functions that map non-normalized scores onto probability vectors. Other functions may have other mathematical properties and different conceptual motivation. In order to fully understand the models in which the softmax function occurs, we must understand enough of the softmax function to know how to use it in practice and interpret results that we obtain with models that use it. However, different types of models and different researcher goals may require different levels of understanding of softmax. To capture two prevalent ways in which researchers apply models that use the softmax function, we distinguish between *input-output (IO) models* and *internally-meaningful (IM) models* and expand on each in turn.

IO models care only about making accurate predictions, e.g., for forecasting or decision making. IO models do not strive to accurately model the data-generating process through their internal parameters and computations. Their inner mechanics need not be interpretable or transparent and they do not aspire to be explanations of phenomena behind the data to be predicted, such as causal mechanisms of mental representations. Examples of IO models are regression models and (deep) neural networks used for engineering purposes (classification, forecasting, prediction etc.). Consequently, for IO models, it is usually *not* very important to ponder the conceptual interpretation of the softmax function. What matters most are its technical properties. IO models will often generate non-normalized scores **s** that are not intrinsically meaningful, but constructed, estimated or learned (from training data) to get the right probabilistic predictions via softmax. Since scores **s** are conceptually unconstrained, possibly freely estimated from the data, there is also no need to vary the softmax parameter α, which is often just clamped to a fixed, arbitrary value (usually: $\alpha = 1$), thereby dispensing with any need to understand what the softmax parameter does and how its values could be interpreted. In fact, if the scores can vary freely and are just estimated from the data, as in multinomial regression or neural network models for categorization, having α as an additional free parameter during training would

[2] We write $\exp s_i$ to mean e^{s_i}, using the same bracketing and association rules as for the log operator. The softmax function usually assumes the base e, but it can be expressed in terms of any base $b > 0$ if we compensate for this change with adjustments to the α parameter (see Corollary 9 in Section 5).

[3] In some contexts, this parameter is also sometimes referred to as "optimality parameter" or "rationality parameter." Some authors use the inverse of α, frequently denoted as τ, and refer to it as "temperature."

make the model overspecified (see Fact 8 in Appendix A). In sum, for IO models, what matters most are the formal properties of softmax covered in Sections 4 and 5.

IM models, on the other hand, are intended to be evaluated not only based on accurate prediction of (some aspect of) the data, but their inner mechanisms are supposed to be meaningfully interpretable and transparent, so that they may function as explanatory models of a phenomenon of interest. Examples of IM models are probabilistic cognitive models (Lewandowsky & Farrell, 2011; Lee & Wagenmakers, 2015) or agent-models in multi-agent simulations, game theory or population dynamics (Goeree et al., 2008; Sandholm, 2010). Requiring internal interpretability, IM models should care about the conceptual justification of the softmax function: after all, for a particular use case and the phenomena to be modelled or explained, a different mapping from non-normalized scores to probabilities may be more appropriate than softmax. Moreover, IM models may use internal parameters, including the non-normalized scores, which are independently meaningful (e.g., interpretable as expected utilities of a decision maker). If so, they are not unconstrained, so that estimation of the softmax parameter α becomes important, and so does the question of how to interpret values of the softmax parameter. Consequently, Section 6 builds on results from Section 5 to enlarge on the interpretation the softmax parameter. Subsequently, Sections 7, 8, and 9 cover three possible conceptual interpretations of the softmax, based on two mathematical derivations.

4 Softmax by I/O

The softmax function takes two inputs, the scores $\mathbf{s}$ and parameter α, and returns a probability vector $\mathbf{p}$ (see Figure 1). To better understand what softmax does, let us explore how different inputs change the output.

Simple I/O. Our first example (see Figure 2) assumes that there are ten outcomes $\mathbf{x} = \langle x_1, \dots, x_{10} \rangle$ (e.g., food items on a menu) associated with a vector of scores $\mathbf{s} = \langle s_1, \dots, s_{10} \rangle$ (e.g., the agent's preferences for each of these choices). Higher scores s_i are associated with a higher preference for the choice x_i. Figure 2 shows that the probability vector returned by the softmax function maps higher numerical scores to higher probabilities. In other words, the mapping from scores to probabilities is monotone increasing (for $\alpha > 0$). Consequently, the highest probability is assigned to the outcome(s) with the highest score. Moreover, we see that the mapping is *non-linear*. This is because the scores are input to an exponential function, which is a non-linear operation. We see non-linearity in the example by noticing that the score difference between the first and second highest scoring options is 1, while that between the second and third highest-scoring option is 2. Nevertheless, the probability difference between the first and second option is bigger than that between the second and the third. (We will see later that non-linearity is a result of a conceptual motivation behind softmax, namely that differences between scores should correspond to probability odds, not probability differences.)

Changing α. The rows in Figure 3 show the effect of changing the softmax parameter α for fixed scores. Generally speaking, the parameter α shapes the form of the probability distribution $\mathbf{p}$ by increasing or decreasing, respectively, the ratios between higher and lower scoring options. Increasing values for α makes higher-scoring options increasingly likely relative to lower-scoring options, ultimately approaching the output of the *argmax* function as α approaches ∞. In the penultimate row in Figure 3 we see that already increasing to $\alpha = 5$ results in a probability vector that places almost all probability mass on the single top-scoring outcome, at least for the score vectors in the left and middle column. At $\alpha = 0$, any information contained within the scores is lost due to their multiplication with

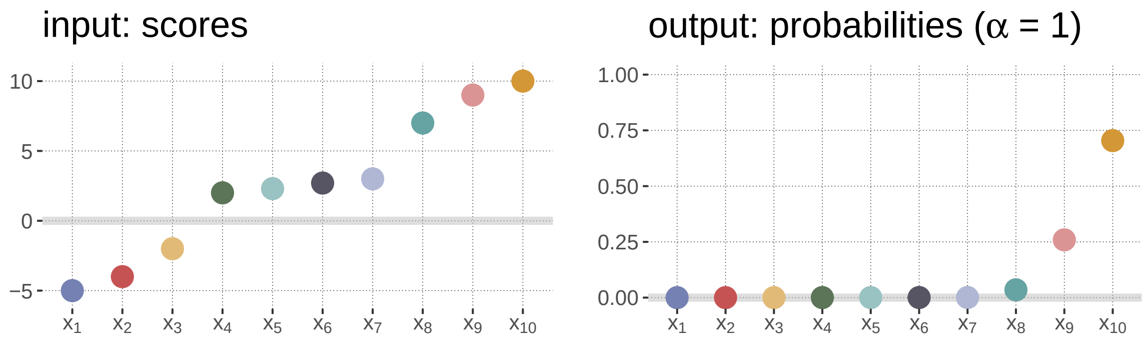

Figure 2: Example of the mapping between scores and probabilities under application of the softmax function with $\alpha = 1$ and a vector of scores $\mathbf{s} = \langle s_1, \dots, s_{10} \rangle = \langle -5,\ -4,\ -2,\ 2,\ 2.3,\ 2.7,\ 3,\ 7,\ 9,\ 10 \rangle$.

0, so that the result is a uniform probability distribution over outcome categories, as shown in the middle row in Figure 3. Finally, for values of $\alpha < 0$, the softmax function puts increased emphasis on the outcomes with the *lowest* score, resulting in a probability distribution $\mathbf{p}$ that favors outcomes which minimize the score. An example for $\alpha = -1$ is given in the last row of Figure 3. Negative values of α results in output that puts higher probability on outcomes the *lower* their scores are, i.e., as if we tried to *minimize* scores.. As α approaches $-\infty$, the output of the softmax function approaches the output of the *argmin* function. However, most often the range of α is restricted to non-negative or even positive values for technical or conceptual reasons.

Changing scores. Let us finally also explore how the output probabilities depend on properties of the input scores when keeping α fixed. The key to understanding how scores affect the softmax output is: **the only thing that matters are differences between scores** (see Fact 2 in Section 5). This observation has important consequences. First, if we add the same number to all scores, the output of the softmax function remains unchanged (see Fact 6 in Appendix A). This is shown in Figure 3 in the left and middle column. A consequence of this is that, even if all scores are negative, all probabilities p_i in the resulting probability vector $\mathbf{p}$ are positive and sum up to one (see Fact 1 in Section 5). Second, the softmax function is *not* invariant to multiplicative transformations of the scores. This is because multiplication with a positive constant change the relative differences between scores s_i and s_j (see Figure 3 left vs. middle column). The latter, in particular, means that the softmax function is not invariant to transformations like the standardization of scores. Multiplicative score transformations can, however, be recovered by dividing the optimality parameter α by the same constant (see Fact 8 in Appendix A).

5 Properties of the softmax function

This section covers important properties of the softmax function in some technical detail. As mentioned in the previous section, the most important observation that will help unlock a better understanding of the softmax operation is that what matters to output probabilities are the differences between input scores (Fact 2).

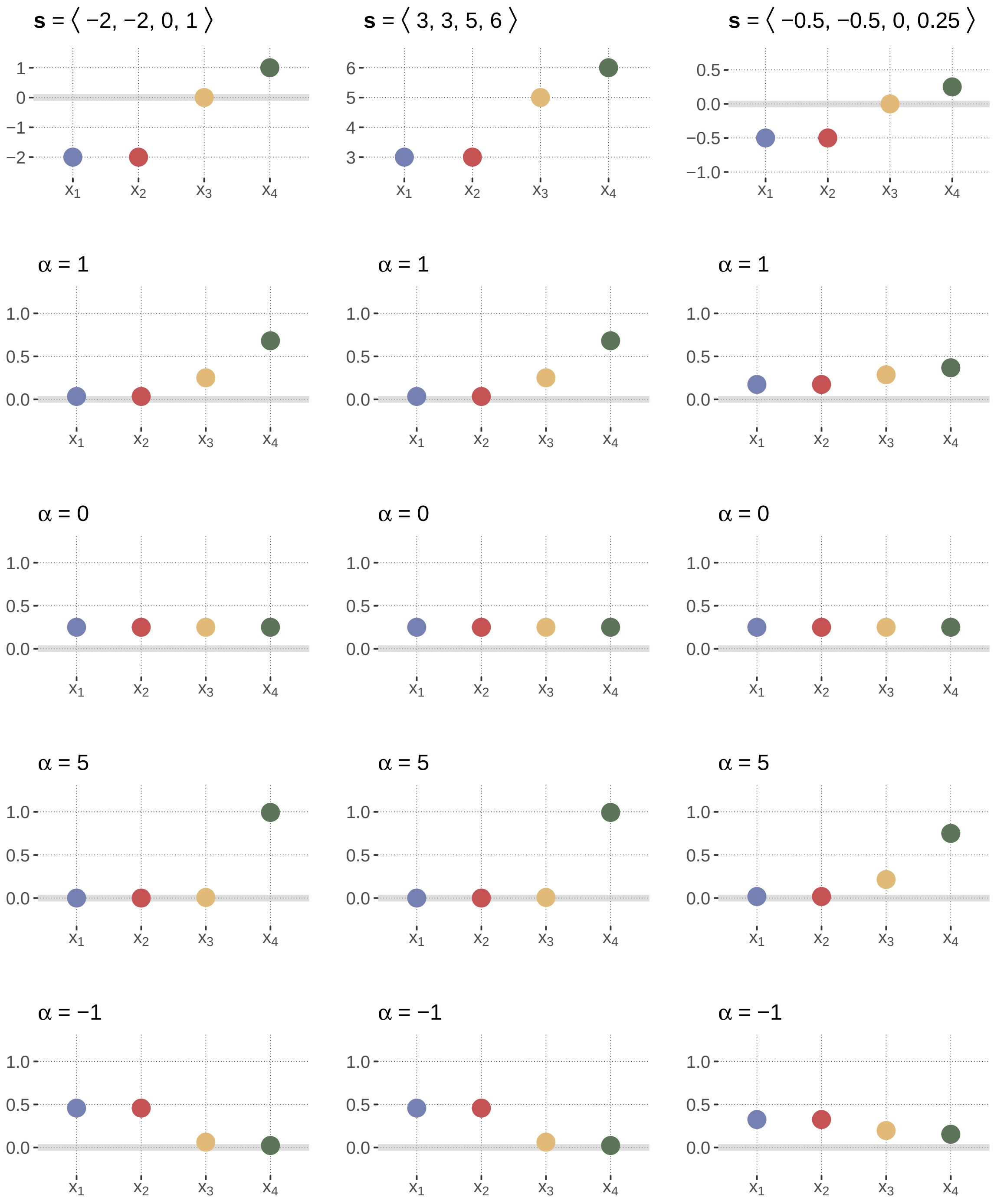

Figure 3: Examples of the output from the softmax function under different transformations of scores (columns) and different values of softmax parameter α. The three columns are different score vectors. The middle is obtained by adding 5 to each score in the vector on the left. The right score is obtained by multiplying the left scores with 0.25.

Well-behaved output for well-behaved input. To start with, let's reassure ourselves that the softmax operation is well-behaved in the sense that it yields positive probabilities $p_i > 0$ for all input scores (as long as there are finite outcome categories and all scores are finite).

Fact 1. Let $\mathbf{s} = \langle s_1, \ldots, s_n \rangle$ be a finite-length vector where all s_i are finite. Then $\mathbf{p} = \text{SoftMax}(\mathbf{s}; a)$ is a probability vector with positive probability for all outcome categories, i.e., i $p_i > 0$ for all and $\sum_{i=1}^{n} p_i = 1$.

A proof of Fact 1 is in Appendix B, as are the proofs for all other formal results to follow.

Defining probabilities through odds ratios. Any finite probability vector $\mathbf{p} = \langle p_1, \ldots, p_n \rangle$ over n categories can be fully determined in terms of just $n - 1$ real numbers. Since probabilities must sum up to one, it suffices to give, for example, just the probabilities $p_2, p_3, \ldots, p_n$, since the missing number p_1 can be retrieved as $p_1 = 1 - p_2 - p_3 - \ldots - p_n$. Indeed, an n-place probability vector can also be fully specified in terms of $n - 1$ odds. The *odds* in favor of option i over option j are the fraction of probabilities p_i/p_j. The probability vector $\mathbf{p} = \langle p_1, \ldots, p_n \rangle$ is fully determined, for instance, via the sequence of odds $p_1/p_2, p_1/p_3, \ldots, p_1/p_n$. For example, in multinomial regression the choice probabilities for n categories are usually determined in terms of $n - 1$ log-odds with respect to a single fixed reference category. One way to think of the softmax function is to fix how exactly a vector of scores translates into a sequence of odds, like $p_1/p_2, p_1/p_3, \ldots, p_1/p_n$.

Softmax defines odds in terms of score differences. Here is a simple mathematical result that helps understand many conceptual and technical aspects of the softmax function.

Fact 2. If $\mathbf{p} = \text{SoftMax}(\mathbf{s}; \alpha)$, then the odds p_i/p_j are a direct function of score differences $s_i - s_j$, namely: $p_i/p_j = \exp(\alpha\, (s_i - s_j))$.

This observation invites us to look at softmax as a function that determines, first and foremost, how differences between scores map onto probability ratios (odds), which in turn define the whole probability distribution (as explained above). This has implications for how to think of the scores themselves. If we use softmax, the absolute value of scores does not matter, only the differences do. So, adding or subtracting a constant to all scores does not change anything (see Fact 6 in Appendix A), but multiplying all scores with a positive number (other than one) will (Fact 7 in Appendix A). Conceptually, this means that scores are not meaningful in absolute terms, but only relatively, through their differences.

Moreover, Fact 2 highlights that the odds p_i/p_j only depend on the scores assigned to options i and j, but not the scores assigned to any other alternatives. This is the sense in which the softmax operation assures that odds of options x_i and x_j are independent of the presence, absence or score of alternatives x_k ($k \neq i$, $k \neq j$).[4] This independence property is conceptually and technically important in different ways for different areas of application of the softmax function. For a cognitive model, for example, if scores represent accumulated evidence for different categories, it is intuitive to assume that *relative* choice probabilities for two options x_i and x_j do not depend on the accumulated evidence in favor of any third alternative x_k. For a data-driven model, like a multinomial regression model or a neural network classifier, if we want to learn a target distribution over categories, then seeing training data that make us adjust the relative probability between x_i and x_j should ideally involve "local changes"

[4] Notice that other mappings from (positive) scores to probabilities, like a power function or a linear function, do not have this property.

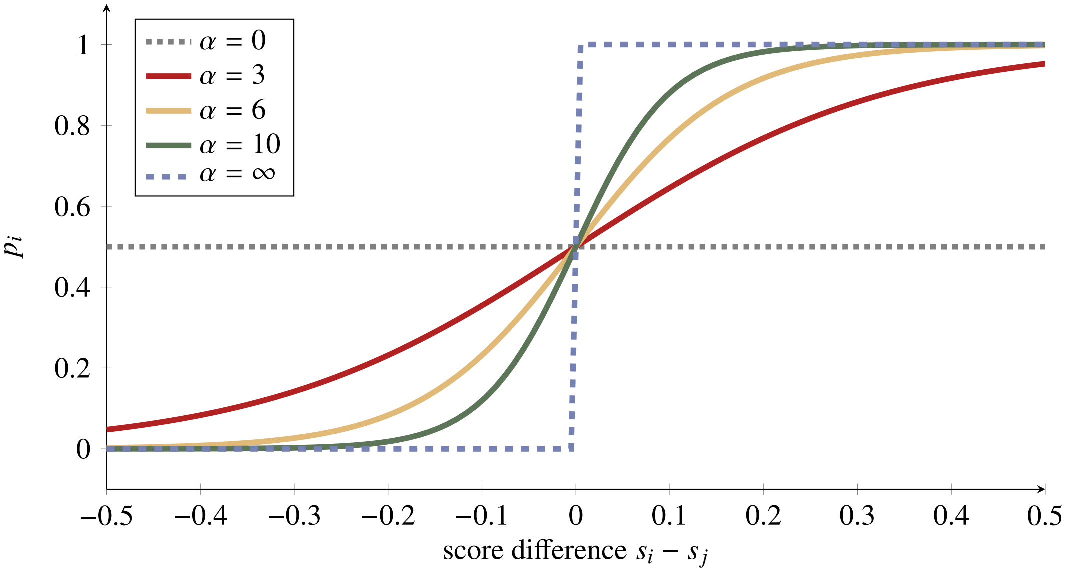

Figure 4: If there are only two outcome categories, the softmax probability of option x_i is a logistic function of the score difference $s_i - s_j$. The softmax parameter α scales the steepness of the curve, where higher α yields steeper curves. The limiting case of $\alpha = 0$ yields indifferent choice (ignoring the scores), and that of $\sigma \rightarrow \infty$ a sharp step function (choose the higher scoring option always).

to internal scores of only those two options, without thereby altering the odds of unrelated options x_k and x_l.

Example with two outcome options: Relation to logistic function. A simple example for thinking about softmax in terms of odds is the special case where we only have two outcome categories $n = 2$. In this case, we only need a single number, like the probability p_i of a reference category x_i, to determine the outcome probability vector. For $n = 2$, the softmax function "reduces to" the logistic function, so speak. Concretely, with $n = 2$ the probability p_i of option x_i is given by the logistic function of the difference in score between x_i and x_j ($i \neq j$).

Fact 3. When $n = 2$, the softmax probability p_i is given by a logistic function of the score difference $d = s_i - s_j$: $p_i = \frac{1}{1+\exp(-\alpha d)}$.

Figure 4 shows examples of the choice probability for the case of $n = 2$. The plot shows the S-shaped curve for choice probabilities as a function of score differences, and the way in which the softmax parameter modulates the steepness of this S-curve.

Further properties. We refer the more technically minded reader to Appendix A, where we list further properties of the softmax function that would take us too far into the weeds for the purpose of this tutorial.

6 Interpreting values of the optimality parameter

This section considers different ways of making sense of numerical values of the optimality parameter. This is usually only relevant when dealing with models in which the optimality parameter is free to vary and, ideally, the scores are interpretable (even if other parts of the IO model are not). When modeling goals make it relevant, interpretability of α can be important in at least two ways: (i) for interpretability *across* different models or data sets, e.g., for interpreting model fits (Bayesian posteriors or point-estimates, like MLE) for different models, or for the same model for data from different populations; and (ii) for interpretability *within* a single model and data set, e.g., for specification of reasonable constraints on the optimality parameter, such as priors in Bayesian modeling.

This section suggests two routes towards a better understanding of α. The first route builds on the observation expressed in Fact 2, that it is score differences that matter to softmax predictions. Based on this idea, we consider three questions one can ask about the values of α: (i) how to interpret a single, fixed value of α (e.g., as returned by a point-estimate after parameter estimation); (ii) what difference it makes to either use α_1 or α_2; and (iii) how to choose priors for optimality parameters in Bayesian data analysis. The second route to interpreting α is more holistic in that it is based on the average score entailed by or the entropy associated with the softmax distribution.

Optimality parameter as log-odds ratio under unit score difference. We saw in Section 5 (Fact 2) that what determines softmax probabilities are *differences* between scores: $\frac{p_i}{p_j} = \exp(\alpha\,(s_i - \; s_j))$. This observation provides an intuitive interpretation of the optimality parameter α in terms of the log-odds given a unit score difference. A *unit score difference* is a case where $s_i - s_j = 1$. For such a case we have:

$$\frac{p_i}{p_j} = \exp\left(\alpha\,(s_i - \; s_j)\right) = \exp \alpha$$

Consequently, α gives the log-odds under a unit score difference:

$$\alpha = \log \frac{p_i}{p_j}$$

For example, say that we have options x_i and x_j whose scores differ by 1, i.e., $s_i - s_j = 1$. Say further that we find that a maximum likelihood fit yields $\hat{\alpha} = 5$. With $\hat{\alpha} = 5$ o_i is predicted to be chosen with a probability that is $\exp(5) \approx 150$ times larger than the probability of o_j. Whether such an estimate of α, is to be judged large or small depends on whether a unit difference in scores itself is to be interpreted as large or small, which depends on the application at hand. If score differences are not interpretable, using the holistic strategy of interpretation introduced below might be more informative.

Interpreting differences between optimality parameters. To understand the difference between two optimality parameter values, we compare SoftMax($\mathbf{s}; \alpha$) against SoftMax($\mathbf{s}; \alpha'$) by looking at the factor $f = \frac{\alpha'}{\alpha}$ by which α' differs from α and note that:

$$\frac{p_i}{p_j} = \exp\left(\alpha\,(s_{i - s_j})\right)$$

$$\frac{p'_i}{p'_j} = \exp\left(\alpha'\,(s_{i - s_j})\right) = \exp\left(f\,\alpha\,(s_{i - s_j})\right) = \left[\exp\left(\alpha\,(s_{i - s_j})\right)\right]^f = \left[\frac{p_i}{p_j}\right]^f$$

In words, if the optimality operator increases by a factor f, the odds of choosing x_i over x_j ($s_i > s_j$) increase by the power of f.

For example, suppose you obtain a maximum-likelihood estimate of $\hat{\alpha}_1$ for the data from an online experiment, and $\hat{\alpha}_2$ for the data from the same experiment executed in the lab. Assume further that $\hat{\alpha}_2$ is twice as large as $\hat{\alpha}_1$. We can interpret this, following Fact 10, as follows. The probability vector

$\mathbf{p}_1$ predicted by $\hat{\alpha}_1$ has to be scaled by a power-law transformation function with power parameter $\frac{\hat{\alpha}_2}{\hat{\alpha}_1}$ to obtain the vector $\mathbf{p}_2$ which is predicted from softmax with $\mathbf{p}_2$. Whether this can be interpreted as large or small would again depend on which magnitude of score differences would count as large or small in the current application.

Prior distributions over α. If we interpret α, in line with Fact 2, as the log odds, $\log \frac{p_i}{p_j}$, for a unit difference in scores $s_i - s_j = 1$, a natural choice of family for a prior on the optimality parameter in Bayesian data analysis is a log-normal distribution (or any other log-something distribution). The mean of the log-normal prior distribution would correspond to the modeller's prior assumption about the expected log odds for a unit difference in scores. The variance of the log-normal prior distribution should be chosen based on the modeller's uncertainty about the log odds for a unit difference in scores. When choosing the variance, guidance is provided by the previous result that increasing α by a factor of f increases odds by the power f. The picture becomes more complicated when the scores are not fixed in advance but hinge on other model parameters. In that case, priors should always be chosen "holistically" and in light of the plausibility of the entailed prior predictive functions (Schad et al., 2021).

Holistic interpretations of α. Let us also consider a completely different approach to making sense of values of α. If we fix $\mathbf{s}$ and consider $\mathbf{p} = \text{SoftMax}(\mathbf{s}; \alpha)$, the functional role of α is to modulate the softmax probabilities $\mathbf{p}$. Consequently, we can also make sense of α-values by looking at suitable numerical values associated with $\mathbf{p}$, i.e., by some function $F(\alpha) \in \mathbb{R}$. This approach is holistic in the sense that it tries to make sense of α based on the whole distribution $\mathbf{p}$, rather than just in terms of comparing the odds of two options. Ideally, the function F has some practically useful properties, like being monotonic, and relates to intuitively meaningful concepts.

A useful choice is the expected score, $c = \mathbf{p} \cdot \mathbf{s}$, and the negative entropy, where entropy is defined as $\mathcal{H}(\mathbf{p}) = -\sum_i p_i \log p_i$. These metrics are especially useful when scaled so that the zero-value is associated with the probability vector obtained for the case of $\alpha = 0$, and the one-value associate with the limiting result for $\alpha \to \infty$. In this way, we obtain holistic metrics quantifying the effect of α on a closed scale that can be interpreted to range from, intuitively speaking, zero optimization (totally random choice) to maximum optimization. The top row of Figure 5 shows the functional relationship between α and the (scaled) expected scores, the bottom row shows the resulting (scaled) negative entropy of the softmax distribution, for two different vectors of scores (which reoccur in the example in Figure 8).

Based on a holistic interpretation, modellers can interpret a given value of α as a "degree of optimization." For instance, the examples in Figure 5 show that to approach the fully optimal (argmax) policy, larger values of α are required for Bo (right-hand side of Figure 5) than for Alex (left-hand side). Plots of this kind can be used to find which values of α would count as an expected degree of optimization, e.g., to inform the choice of the mean of a Bayesian prior, and we can inspect how changes in α lead to changes on this scale (such as to inform the variance of a Bayesian prior). Unfortunately, while the mapping from α to average score is straightforward to calculate, the inverse does not have a simple mathematical solution. Nevertheless, an arbitrarily close approximation can be obtained with numerical estimation (see example R code in Appendix C).

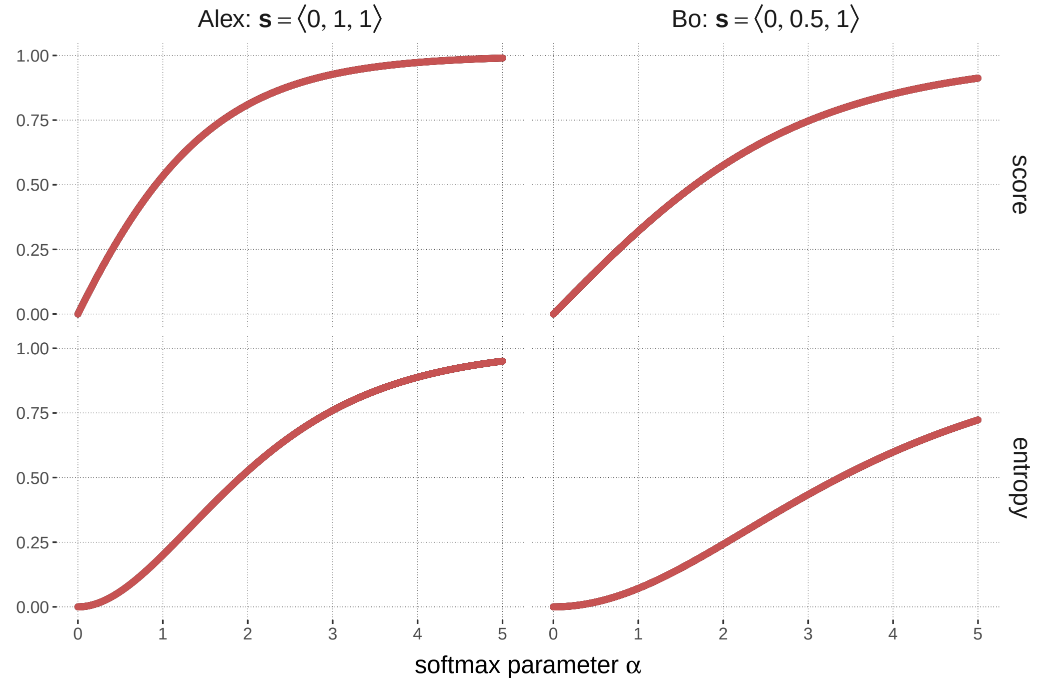

Figure 5: Examples of the relation between softmax parameter α and the (standardized) expected score achieved by the softmax policy for the given α (top row), as well as the (standardized) negative entropy of the softmax policy (bottom row). Values are scaled to be zero for the expected score under $\alpha = 0$, i.e., when choices are uniform at random. Values are scaled to be one for the softmax policy resulting from $\alpha \rightarrow \infty$, i.e., when decisions approximate the arg-max function. The plots show how the interpretation of α depends (holistically) on the scores: for some cases it can be easier to achieve high average scores of negative entropy with lower values of α.

7 Softmax as the outcome of noise-perturbed selection

When dealing with internally-meaningful (IM) models, the conceptual interpretation of all model ingredients matters, including that of the softmax function. modellers therefore might need to justify why the model contains softmax and not some other function mapping scores onto probabilities. In this and the two following sections, we provide three different conceptual interpretations of the softmax distribution, which yield three different rationales for its use. In this section, we give a mathematical derivation that supports a possible mechanistic interpretation of the softmax function, namely as the distribution resulting from errors in the computation or assessment of the scores. **Under this interpretation, the softmax function implements the assumption that a choice process is approximate, sub-optimal, or error-prone**.

Concretely, the distribution which is returned by the softmax function can be derived as the probability distribution entailed by a stochastic choice mechanism, in which the choice of outcome x_i is optimal (an x_i with the highest score is chosen), but in which the scores themselves are "wiggled" or "noise-perturbed" each time a decision has to be made (Luce, 1959; Train, 2009). Given specific choices about the probability of the "wiggles," we can derive the output of softmax function as the expected choice frequencies. Figure 6, to be unpacked below, illustrates this process.

Let's think of $x_1, \ldots, x_n$ as an agent's available choice options (actions), each with a score $s_1, \ldots, s_n$

which tells us how good each choice is. The agent in question could be a human decision maker (making perceptual or economic decisions), but it could also be an abstract system "choosing" a category based on the objective to maximize the score. An optimal (or rational) agent would always only choose x_i if $s_i = \max_j s_j$, so an optimal agent would maximize the score perfectly. But let us now assume that there is room for imperfection. Mistakes happen. But it's not the case that all mistakes are equally likely. An agent is more likely to choose a suboptimal option whose score is almost maximal than to choose a suboptimal option which is far worse. Essentially, this leads to the desire to formulate *probabilistic choice rules* such that the probability p_i of choosing x_i is higher the higher its relative score is (Luce, 1959; Train, 2009). The softmax choice rule can be derived as the probability p_i that an agent chooses x_i as the best choice from a noise-perturbed representation $s'_1, \dots, s'_n$ of the scores, where $s_j = s_j + \epsilon_j$ and all ϵ_j are independently and identically distributed random noise perturbations of the actual scores of the available options. In other words, each time the agent chooses from $x_1, \dots, x_n$, they choose $x \in \arg\max_j (s_j + \epsilon_j)$ for a vector of errors $\epsilon = \langle \epsilon_1, \dots, \epsilon_n \rangle$ which are sampled anew every time a choice is made (see Figure 6).

To derive the softmax function, we must make specific assumptions about the probability distribution from which each ϵ is sampled. Concretely, we assume that the error terms ϵ_j come from a **Gumbel distribution** with location $\mu = 0$ and scale parameter $\beta > 0$ (see proof of Fact 4 in Appendix B for a definition of the Gumbel distribution), in order to derive the following:

Fact 4. The softmax distribution $\mathbf{p} = \mathrm{SoftMax}(\mathbf{s}; \alpha)$ is the expected choice probability if outcome categories are selected based on maximization of noise-perturbed scores $\mathbf{s}'$, where each $s'_i = s_i + \epsilon_i$ is obtained by adding an iid sample ϵ_i from a Gumbel distribution with location $\mu = 0$ and scale $\beta = \frac{1}{\alpha}$.

Figure 7 gives examples of the probability density function of Gumbel distributions for different values of $\alpha = \frac{1}{\beta}$. The plots show that the error assumed in this derivation has a certain structure, namely that larger positive values for ϵ are more likely than very small negative ones. This means that a model that contains the softmax function and interprets it as the result of stochastic errors, as described here, commits itself to a rather specific sort of error. However, this assumption is necessary to obtain a result like Fact 4; a similar result is *not* derivable, for example, under the assumption that error terms are normally distributed (Train, 2009).

In sum, the derivation of softmax presented in this section tells us that *there exists* at least one way of justifying the softmax function in terms of a specific "noisy process model." This derivation must make rather specific assumptions about the nature of the assumed noise perturbation (additive trembles, *iid*-sampled from a specific distribution, namely a Gumbel distribution). This gives rise to two questions: First, what should we do if we have more specific domain knowledge about a likely noise process that adds "trembles" or stochasticity to the modelled choice process? — Here, the ideal situation would be to model the noise process directly because this makes for a more robust and empirically testable model. Second, what if have no (strong) prior conceptualization of how stochasticity in choice might arise? What if, specifically, we have no *a priori* reason to believe that the *iid*-additive Gumbell trembles are a reasonable or salient model of stochasticity in the choice process? — In this case, you might still want to use the softmax function, because it is not only compatible with the *iid*-additive Gumbell model, it is also, in a sense, the most neutral, or least specific model of stochastic trembles. This is the topic of the next section.

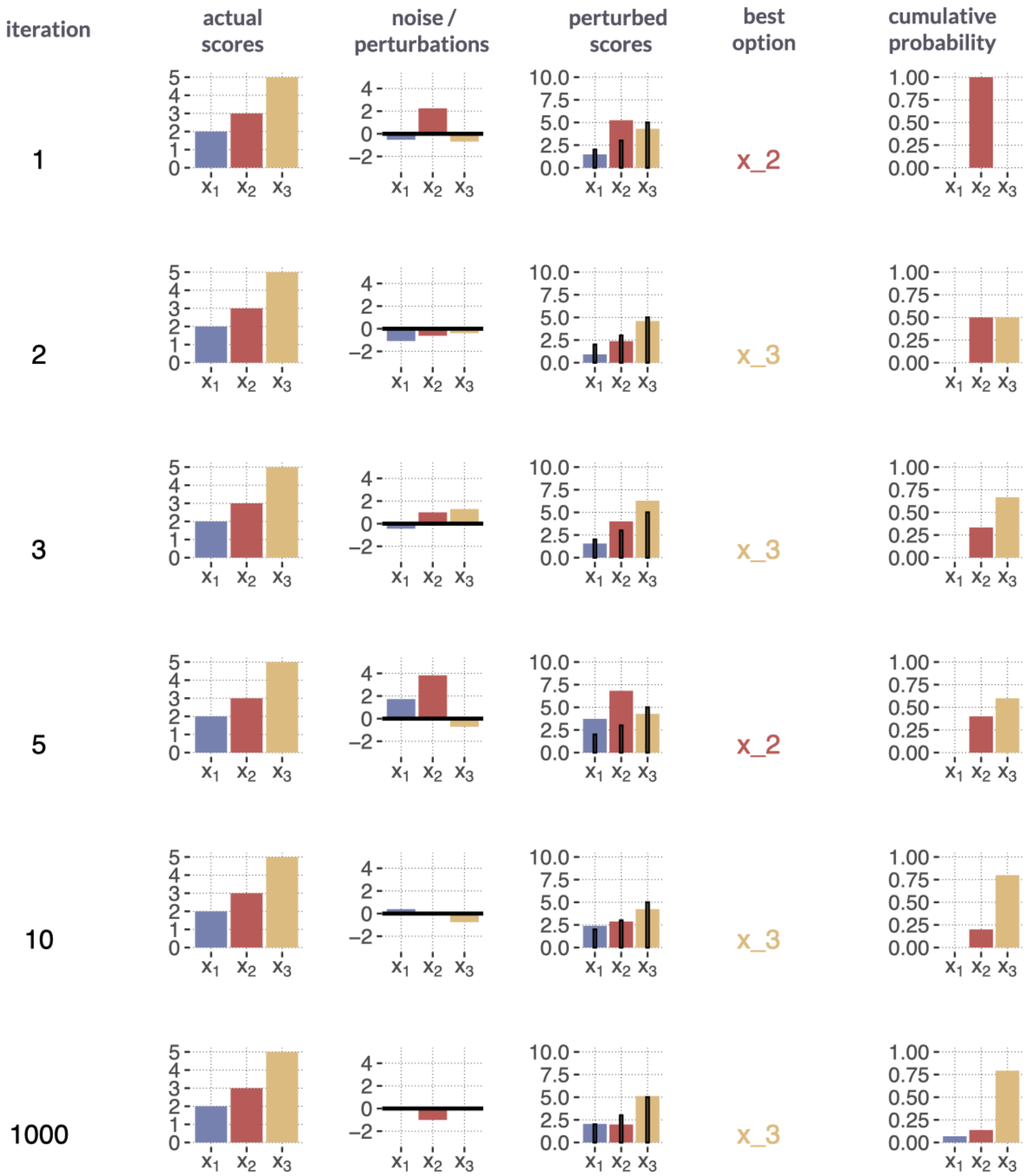

Figure 6: Illustration of repeatedly sampling the best option after noise-perturbations ($\alpha = 0.8$). The first column shows the number of choices the agent made. The second column shows the actual scores of three outcome categories (which are the same for each choice / row). For each choice (row) a new vector of Gumbel-distributed perturbations are sampled (column 3) and added to the original scores (as shown in column 4, where the actual scores are indicated with slimmer black bars). Based on the perturbed scores the best choice is selected (column 5). The final column shows the relative frequency choices so far. In the limit, as shown in approximation for 1000 trials in the last row, this frequency distribution approaches the softmax distribution.

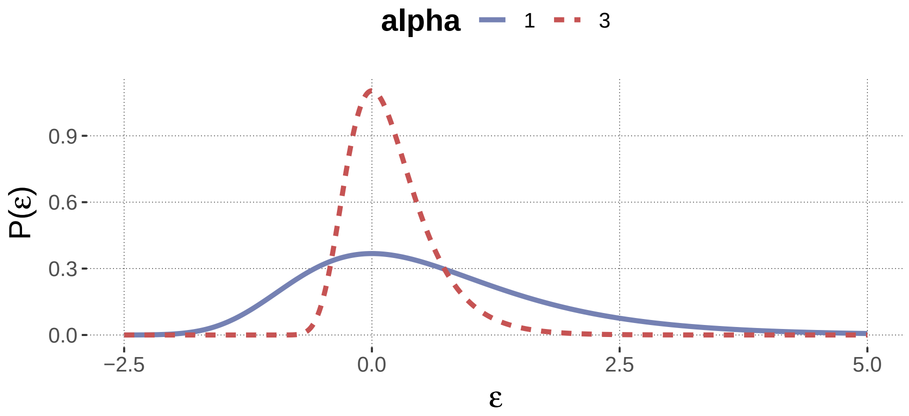

Figure 7: Examples of Gumbel probability density with location $\mu = 0$ and different values for $\beta = {}^{1}/_{\alpha}$.

8 Softmax as optimal choice if error source is unknown

This section shows that the softmax function is a maximum entropy distribution (for certain constraints), so that it is arguably a natural go-to option in the absence of strong prior knowledge of the noise processes that might affect the probabilities to be modelled. The previous section showed that softmax is a motivated choice if we want to model a noisy decision process, and if we are willing to accept, at least for practical purposes, a *specific* assumption about the origin of stochasticity. The result presented in this section can be interpreted as suggesting that **we can think of the softmax function as the most neutral, or least assuming, model of a stochastic choice process with unknown error source**.

Let us start with some motivating examples. Consider the following score vector for a three-way choice of Alex (see Figure 8):

$$\mathbf{s}^{[A]} = \langle 0, 1, 1 \rangle$$

Alex uses a stochastic choice policy $\mathbf{p}^{[A]}$, but you do not know which. Suppose that you knew or reasonably assumed for some purpose that the average score under Alex's choice policy, when repeatedly choosing one of the three options, is $\mathbf{p}^{[A]} \cdot \mathbf{s}^{[A]} = 0.75$. There are infinitely many stochastic policies that would achieve an expected score of 0.75, but they are not all equally "neutral" or "unassuming." For example, two choice policies that yield the average score of 0.75 for Alex are these:

$$\mathbf{p}^{[A,1]} = \langle 0.25, 0.75, 0 \rangle \qquad \mathbf{p}^{[A,2]} = \langle 0.25, 0.375, 0.375 \rangle$$

In $\mathbf{p}^{[A,1]}$ Alex never chooses option 3, but chooses option 1 with probability 0.25 and option 2 with probability 0.75. In $\mathbf{p}^{[A,2]}$ Alex chooses option 1 with probability 0.25 and options 2 and 3 with probability 0.375 each. A function which yields $\mathbf{p}^{[A,1]}$ for Alex's scores clearly makes rather specific assumptions about Alex choice processes which are *not* apparent just from the scores. In other words, if the scores are the only thing that we know about what influences Alex's (possibly noisy) choice process, stipulating $\mathbf{p}^{[A,1]}$ seems unwarranted: where should the difference between options 2 and 3 come from when it is not in the scores? On the other hand, the policy $\mathbf{p}^{[A,2]}$ does not seem to make any such additional assumptions about Alex's choice process. In fact, it is easy to see that it is the *only* three-place probability vector which (i) yields the fixed average score of 0.75 and (ii)

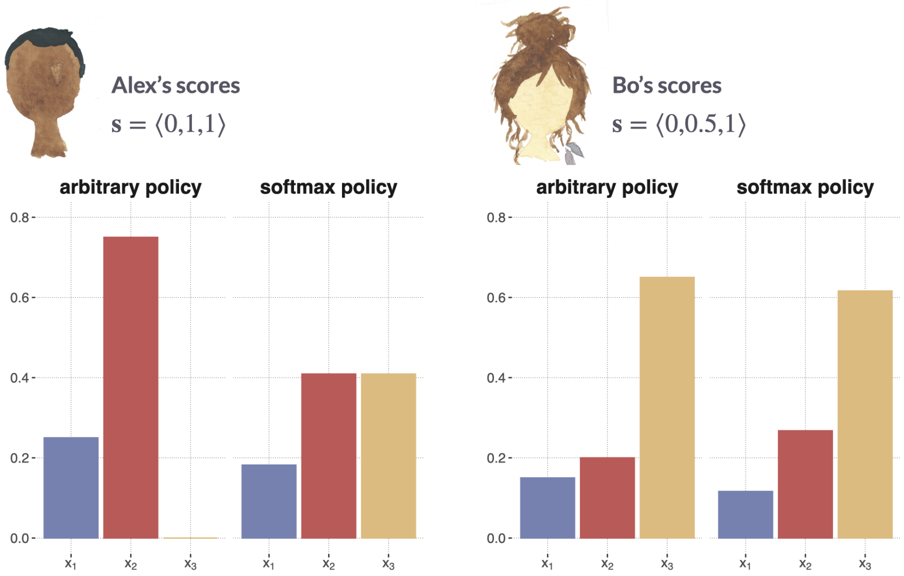

Figure 8: Two different stochastic choice policies for each of two score vectors. All pairs of scores and choice policies have an expected score of 0.75. Although there are infinitely many stochastic choice policies that yield a fixed expected score (see the left side on each panel), the softmax rule (on the right of each panel) always yields a "neutral" or "unassuming" choice policy corresponding to the maximum entropy distribution that generates the given average score. (Values for the softmax function that achieve the expected score of 0.75 are $\alpha \approx 0.41$ (left) and $\alpha \approx 1.67$ (right).)

assigns probabilities to choice options that respect the underlying scores (higher score yields higher probability; equal score yields equal probability). It is in this sense that $\mathbf{p}^{[A,2]}$ is the more "neutral" or "unassuming" model for Alex's stochastic choice policy. Indeed, as we will see below, $\mathbf{p}^{[A,2]}$ is the maximum entropy distribution satisfying the constraint that average scores equal 0.75 and $\mathbf{p}^{[A,2]}$ is the *only* distribution that results from the softmax rule that yields the average score of 0.75.

Let us also briefly consider a second example. Suppose that Bo's scores are these:

$$\mathbf{s}^{[B]} = \langle 0, 0.5, 1 \rangle$$

Bo uses a stochastic choice policy $\mathbf{p}^{[B]}$, but you, again, do not know which. Let's assume that Bo, too, has a known average score of $\mathbf{p}^{[B]} \cdot \mathbf{s}^{[B]} = 0.75$. In this case, there are infinitely many stochastic policies which yield this average score and also satisfy the constraint that they assign probabilities to choice options that respect the underlying scores (higher score yields higher probability; equal score yields equal probability). For example, the following two policies of Bo's meet both requirements; again, see Figure 8 for an illustration:

$$\mathbf{p}^{[B,1]} = \langle 0.15, 0.2, 0.65 \rangle \qquad \mathbf{p}^{[B,2]} \approx \langle 0.116, 0.269, 0.616 \rangle$$

The first vector $\mathbf{p}^{[B,1]}$ is an example that was chosen arbitrarily for illustration. The second vector $\mathbf{p}^{[B,2]}$ is the prediction of softmax for $\alpha = 1.66823$, the unique real-valued α that satisfies $\text{SoftMax}(\mathbf{s}^{[B]}; \alpha) \cdot \mathbf{s}^{[B]} = 0.75$. Unlike for the case of Alex, it may not be clear which one of these stochastic choice policies is more or less "neutral" or "unassuming." It depends on what we mean by these terms. One salient contender for filling in the blank is information theory. The *entropy* of a (categorical) distribution **p** is defined as:

$$\mathcal{H}(\mathbf{p}) = -\sum_j p_j \ \log p_j$$

In words, entropy measures the expected surprisal for an agent with beliefs **p** if outcomes indeed occur with probabilities described by **p**. Therefore, if we —as modellers— would like to minimize surprisal on average in the stochastic policies that we ascribe to a process of decision making, we best select a distribution that maximizes entropy (i.e., minimizes modeller's surprisal on average). The entropy of the two candidate policies in the running example are:

$$\mathcal{H}(\mathbf{p}^{[B,1]}) \approx 0.887 \qquad\qquad \mathcal{H}(\mathbf{p}^{[B,2]}) \approx 0.901$$

So, from the two candidate policies that both yield the same average score, the softmax policy has higher entropy and is therefore "more neutral" or "less assuming" in an information-theoretic sense.

Indeed, it is provable that, in general, softmax yields the maximum entropy distribution for a given value of the expected score. Assume that we know the scores $\mathbf{s} = \langle s_1, \ldots, s_n \rangle$ and we know the expected utility $c = \mathbf{p} \cdot \mathbf{s}$. We don't know anything else about $\mathbf{p} = \langle p_1, \ldots, p_n \rangle$, except that $\sum_i p_i = 1$. Then we can prove:

Fact 5. Softmax($\mathbf{s}; \alpha$) is the maximum likelihood solution for **p** under constraints $c = \mathbf{p} \cdot \mathbf{s}$ and $\sum_i p_i = 1$.

In sum, this result tells us that the softmax operation yields probability distributions which are, in a particular sense, neutral or unassuming. Consequently, we can motivate the choice of softmax in a model by saying that this is what researchers *should* optimally choose if they assume that choices are stochastic, but have no idea about, or do not want to make any commitment regarding, what the underlying error source could be, and want to otherwise remain as neutral as possible. As with the result presented in Section 7, this interpretation is itself not unassuming, because it requires that we accept that information-theoretic entropy is a good formalization of the relevant notion of "neutrality." This interpretation also puts particular emphasis on the constraint necessary to derive Fact 5, namely that the expected score is meaningful to the researcher. To address these issues, the following section looks at a completely different conceptual interpretation of the mathematical result presented in Fact 5.

9 Softmax as optimal balance between exploration and exploitation

The previous two interpretations of the softmax function were anchored in the idea that stochasticity, i.e., potential deviance from the $\alpha \rightarrow \times$ prediction, are sub-optimal or errors. This line of interpretation assumes that everything that matters to evaluating the goodness of a choice is captured in the scores, which are immutable. However, several converging lines of research plead for more nuanced pictures of "optimality," to also include representation and computation costs in the form of bounded rationality (Simon, 1959) or resource-bounded rationality (Lieder & Griffiths, 2019). Other work stresses long-term adaptive or ecological rationality, possibly also taking a changing environment into account (e.g., Anderson, 1990; Chater & Oaksford, 2000; Hagen et al., 2012; McNamara, 2013). In this general line of thought, **we may think of the softmax function as realizing an optimal tradeoff between exploitation and exploration**.

Consider decision-maker Alex in Figure 8, with the scores $\mathbf{s} = \{0, 1, 1\}$ for three choice options. The minimum score that Alex can attain is 0, the maximum is 1. If Alex wants to score minimally, there is only one thing to do: choose the first option. If Alex wants to score maximally, well, there are infinitely many stochastic choice protocols that serve. One of them is the maximum entropy distribution (for the aspired expected score of 1) where Alex chooses options 2 and 3 with equal probability. Is there anything special about this stochastic choice protocol from the point of view of ecological rationality? — Yes! Suppose that Alex makes repeated decisions in a variable environment, where scores can change over time. To detect changes in scores (and therefore exploit them quickly), it is —intuitively speaking— wise to play "maximally random" for a given desired expected outcome. Since similar considerations also apply to cases where Alex aspires to less than an average score of 1, this provides at least an informal, intuitive, rationale for the use of the softmax function.

In sum, rather than assuming that it is the *modellers'* expectation for a particular score, and the *modellers'* uncertainty about the error source that grounds the mathematical result reported in Fact 9, we can also adopt the agent perspective: if an agent wants to realize a fixed expected score, but also wants to hedge their bets maximally in the face of a dynamically changing world, then, if we assume that information-theoretic entropy is a good formalization of "playing as randomly as possible," we can motivate the softmax function as the optimal balance between exploitation (obtaining a fixed expected score) and exploration (using a choice policy that maximizes entropy).

10 Summary & conclusion

To sum up, the softmax function is frequently used in different kinds of models. To understand the models in which the softmax function occurs, at least an understanding of the input-output behavior of the softmax function is necessary. Knowledge of some mathematical properties helps better understand this input-output behavior, most notably the observation that softmax probabilities are really a function of *differences* between scores. Formal properties of the softmax function also help to interpret the α parameter, for which this tutorial suggested two perspectives: one in terms of (log) odds for unit differences in scores, and another, more holistic, in terms of either (scaled) expected scores or (scaled) negative entropy, both of which yield metrics that can be intuitively interpreted as "degree of optimization."

The tutorial introduced a distinction between what we called input-output models and internally-meaningful models, where the latter, but not necessarily the former, require explainability of all model ingredients, including the softmax function. Three different conceptual interpretations of the softmax function were offered, each of which may justify its use in a given situation. Concretely, we characterized the softmax distribution as either: (i) the expected choice distribution if choices are noise-perturbed in a particular way, (ii) the most neutral assumption about stochastic noise-perturbed choice if the specific error distribution is unknown, and (iii) the optimal tradeoff between exploitation and exploration. None of these interpretations are free of controversial assumptions. We must either assume particular error-distributions, or that information-theoretic entropy is the right formalization of intuitive concepts like "neutrality" or "exploration." Whether this is (approximately) correct for any particular modeling situation is something the modellers and the receiving community have to assess critically on a case-by-case basis.

A More formal properties of softmax

Invariance under addition. Since softmax probabilities are a function of the differences between input scores, adding or subtracting the same number from all scores should not change the softmax outcome:

Fact 6. Softmax is invariant under addition: for all $a \in \mathbb{R}$, $\text{SoftMax}(\mathbf{s}; \alpha) = \text{SoftMax}(\mathbf{s} + a; \alpha)$.

Non-invariance under multiplication. If we multiply all scores with the same factor $a \neq 1$, this *does* affect the softmax probabilities, expect in special cases.

Fact 7. Softmax is not invariant under multiplication: if $a \in \mathbb{R} > 0$ is a constant, then $\text{SoftMax}(\mathbf{s}; \alpha) = \text{SoftMax}(a\, \mathbf{s}; \alpha)$ only in trivial cases, namely if $\alpha = 0$, $a = 1$ or $s_i = s_j$ for all i and j.

Adjusting α. Even though, by Fact 7, softmax is not invariant under multiplication of scores, the effect of multiplication by positive factor $a > 0$ can be compensated with the choice of a different softmax parameter α. This is the content of the following fact, and important to understand that a model that leaves all scores and α as free parameters (to be inferred from or optimized based on some data) is overspecified (unless it adds additional constraints, such as Bayesian priors).

Fact 8. Multiplicative factors can be recovered by different optimality parameters: if $a \in \mathbb{R} > 0$ is a constant, then $\text{SoftMax}(\mathbf{s}; \alpha) = \text{SoftMax}(a\, \mathbf{s}; {}^{\alpha}\!/_{a})$.

Change of base. Although the softmax operation is frequently implemented in terms of the exponential function with base e, this is not a necessity (technically speaking). We can recover any other base by changing the α parameter accordingly.

Corollary 9. Fact 8 implies that the softmax function need not be expressed in base e, but can equivalently be expressed in terms of any basis $b > 0$ if we change the softmax parameter accordingly.

Power-law decomposition. For a given probability vector $\mathbf{p}$, a common transformation function (for scaling odds) is the *power law transformation*, defined as follows:

$$\text{Pow}(\mathbf{p}; \alpha) = \mathbf{q}, \quad \text{with: } q_i = \frac{p_i^{\alpha}}{\sum_j p_j^{\alpha}}$$

We can think of the softmax operation with parameter $\alpha = a$ as a composition of softmax with $\alpha = 1$, followed by a power law transformation with $\alpha = a$.

Fact 10. $\text{Pow}(\text{SoftMax}(\mathbf{s}, 1); \alpha) = \text{SoftMax}(\mathbf{s}, \alpha)$

B Proofs

Proof of Fact 1. The non-negativity of p_i follows directly from the reciprocity of the exponential function $\exp(x)$. For $x \geq 0$, the exponential function is defined as:

$$\exp(x) = 1 + x + \frac{x^2}{2} + \ldots \geq 1 > 0$$

For $x < 0$, we have:

$$\exp(x) = \frac{1}{\exp(-x)} > 0$$

Since $\exp(x) > 0$ for all $x \in \mathbb{R}$, $\frac{\exp(\alpha\ s_i)}{\sum_j \exp(\alpha\ s_j)} > 0$ for all $s_i, s_j, \alpha \in \mathbb{R}$. That $\sum_{i=1}^{n} p_i = 1$ is assured by normalization in the definition of softmax. □

Proof of Fact 2.

$$\frac{p_i}{p_j} = \frac{\exp(\alpha\ s_i)}{\sum_k \exp(\alpha s_k)}\ \frac{\sum_k \exp(\alpha s_k)}{\exp(\alpha\ s_j)} = \frac{\exp(\alpha\ s_i)}{\exp(\alpha\ s_j)} = \exp(\alpha\ s_i - \alpha\ s_j) = \exp(\alpha\ (s_i -\ \ s_j))$$

□

Proof of Fact 3.

$$\begin{aligned} p_i &= \frac{\exp(\alpha\ s_i)}{\exp(\alpha\ s_i) + \exp(\alpha\ s_j)} = \frac{\exp(\alpha\ s_i)}{\exp(\alpha\ s_i) + \exp(\alpha\ s_j + s_i - s_i)} \\ &= \frac{\exp(\alpha\ s_i)}{\exp(\alpha\ s_i) + \exp(\alpha\ s_j)\ \exp(\alpha\ - d)} = \frac{1}{1 + \exp(-\alpha d)} \end{aligned}$$

□

Proof of Fact 4. In general, the Gumbel distribution has the following cumulative density and probability density functions:

$$F(\epsilon; \mu, \beta) = \exp\left(-\exp\left(-\frac{\epsilon - \mu}{\beta}\right)\right) \qquad f(\epsilon; \mu, \beta) = \frac{1}{\beta}\exp\left(-\frac{\epsilon - \mu}{\beta}\right)\exp\left(-\exp\left(-\frac{\epsilon - \mu}{\beta}\right)\right) .$$

The variance of this distribution is a function of the scale parameter: $\frac{\pi^2}{6}\beta^2$. This is important for interpreting the optimality parameter α of the softmax function, because we will set $\alpha = \frac{1}{\beta}$ to obtain:

$$F(\epsilon; \mu = 0, \beta = {}^1\!/\!_\alpha) = \exp\left(-\exp\left(-\alpha\epsilon\right)\right) \quad f(\epsilon; \mu = 0, \beta = {}^1\!/\!_\alpha) = \alpha\exp\left(-\alpha\epsilon\right)\exp\left(-\exp\left(-\alpha\epsilon\right)\right) .$$

The stochastic “wiggles” ϵ can be thought of as random errors in the computation of the scores. (We could say that the agent makes rational choices given a momentarily and subjectively distorted representation of actual scores.) The probability that an agent who maximizes (noise-perturbed) utility chooses action a_i is therefore:

$$p_i \quad = \quad P(\forall j \neq i:\ s_i + \epsilon_i > s_j + \epsilon_j) \quad = \quad P(\forall j \neq i:\ \epsilon_j < \epsilon_i + s_i - s_j) \tag{1}$$

Given the assumed distribution of noise perturbations ϵ, we can spell out the probability p_i from (1) as a function $P(x_i; \alpha)$ of α. Let’s first assume, unrealistically, that we would know the value of ϵ_i. From the right-hand side of (1), p_i would then be determined by how likely it is to sample a set of ϵ_j-s all of which are below a given threshold $\epsilon_i + s_i - s_j$. Since all ϵ_j are sampled independently, this is the product of the cumulative densities for all ϵ_j being smaller than the threshold $\epsilon_i + s_i - s_j$:

$$P(x_i; \alpha)^{|\epsilon_i} \quad = \quad \prod_{j \neq i} F(\epsilon_i + s_i - s_j; \mu = 0, \beta = {}^1\!/\!_\alpha) \quad = \quad \prod_{j \neq i} \exp\left(-\exp\left(-\alpha(\epsilon_i + s_i - s_j)\right)\right)$$

But, of course, we do not know the value of ϵ_i. We only know its distribution, so that:

$$\begin{aligned}
P(x_i;\alpha) &= \int f(\epsilon_i; \mu = 0, \beta = {}^1\!/\!_\alpha) \prod_{j \neq i} \exp\Big(-\exp\Big(-\alpha(\epsilon_i + s_i - s_j)\Big)\Big)\ d\epsilon_i \\
&= \int \alpha \exp\left(-\alpha\epsilon_i\right) \exp\left(-\exp\left(-\alpha\epsilon_i\right)\right) \prod_{j \neq i} \exp\Big(-\exp\Big(-\alpha(\epsilon_i + s_i - s_j)\Big)\Big)\ d\epsilon_i \\
&= \alpha \int \exp\left(-\alpha\epsilon_i\right) \prod_{j} \exp\Big(-\exp\Big(-\alpha(\epsilon_i + s_i - s_j)\Big)\Big)\ d\epsilon_i \\
&= \alpha \int \exp\left(-\alpha\epsilon_i\right) \exp\left(-\sum_j \exp\Big(-\alpha(\epsilon_i + s_i - s_j)\Big)\right)\ d\epsilon_i \\
&= \alpha \int \exp\left(-\alpha\epsilon_i\right) \exp\left(-\exp\left(-\alpha\epsilon_i\right) \sum_j \exp\Big(-\alpha(s_i - s_j)\Big)\right)\ d\epsilon_i \\
&= \alpha \int \exp\left(-\alpha\epsilon_i\right) \exp\left(-c \exp\left(-\alpha\epsilon_i\right)\right)\ d\epsilon_i \qquad \left[\text{with } c = \sum_j \exp\Big(-\alpha(s_i - s_j)\Big)\right] \\
&= \frac{\exp(-c\left(-\alpha\epsilon_i\right))}{c}\Bigg|_{-\infty}^{\infty} \\
&= \lim_{\epsilon_i \to \infty} \frac{\exp(-c\exp\left(-\alpha\epsilon_i\right))}{c} - \lim_{\epsilon_i \to -\infty} \frac{\exp(-c\exp\left(-\alpha\epsilon_i\right))}{c} = \frac{1}{c} - 0 \\
&= \frac{1}{\sum_j \exp\Big(-\alpha(s_i - s_j)\Big)} = \frac{1}{\exp\left(-\alpha s_i\right) \sum_j \exp\Big(\alpha s_j\Big)} = \frac{\exp(\alpha\ s_i)}{\sum_j \exp(\alpha\ s_j)}\ .
\end{aligned}$$

□

Proof of Fact 5. Let $\mathcal{H}(\mathbf{p}) = -\sum_j p_j \log p_j$ be the entropy of $\mathbf{p}$. Consider further two auxiliary functions, namely $f(\mathbf{p}) = \mathbf{p} \cdot \mathbf{s}$, and $g(\mathbf{p}) = \sum_i p_i$. We want to find the critical points of the Lagrangian:

$$\mathcal{L}(\mathbf{p}, \alpha, \beta) = \mathcal{H}(\mathbf{p}) + \alpha\left(f(\mathbf{p}) - c\right) + \beta\left(g(\mathbf{p}) - 1\right)$$

As usual, the partial derivatives $\frac{\partial\mathcal{L}}{\partial\alpha}$ and $\frac{\partial\mathcal{L}}{\partial\beta}$ reduce to the auxiliary constraints. Suffice it to compute the partial derivatives $\frac{\partial\mathcal{L}}{\partial p_i}$ for an arbitrary $1 \leq i \leq n$:

$$\begin{aligned}
0 = \frac{\partial}{\partial p_i} &= -\sum_j p_j \log p_j + \alpha \sum_j p_j s_j + \beta \sum_j p_j \\
&= -\log p_i - 1 + \alpha s_i + \beta
\end{aligned}$$

Since the critical point implies $\frac{\partial\mathcal{L}}{\partial p_i} = 0$, we can solve for p_i:

$$p_i = \exp\left(\alpha s_i + \beta - 1\right) = \exp\left(\alpha s_i\right)\ \exp\left(\beta - 1\right) \tag{2}$$

With this, we can expand the second auxiliary constraint:

$$1 = \sum_j p_j = \sum_j \exp\left(\alpha s_i\right)\ \exp\left(\beta - 1\right)$$

This is equivalent to:

$$\exp(\beta - 1) = \frac{1}{\sum_j \exp(\alpha s_i)} \tag{3}$$

Combining Equations (2) and (3), we get:

$$p_i = \frac{\exp(\alpha s_i)}{\sum_j \exp(\alpha s_j)}$$

□

Proof of Fact 6.

$$p_i = \frac{\exp(\alpha\ (s_i + a))}{\sum_j \exp(\alpha\ (s_j + a))} = \frac{\exp(\alpha)^{s_i + a}}{\sum_j \exp(\alpha)^{s_j + a}} = \frac{\exp(\alpha)^{s_i} \exp(\alpha)^{a}}{\exp(\alpha)^{a} \sum_j \exp(\alpha)^{s_j}} = \frac{\exp(\alpha\ s_i)}{\sum_j \exp(\alpha\ s_j)}$$

□

Proof of Fact 7. To begin with, notice that if all scores in **s** are equal, softmax probabilities will be equal, no matter which α. Also, if $\alpha = 0$, all softmax probabilities will be equal, no matter the scores. Therefore, fix **s** with at least two scores s_i and s_j unequal ($s_i > s_j$) and let $\mathbf{p} = \mathrm{SoftMax}(\mathbf{s}; \alpha)$ for arbitrary $\alpha \neq 0$. Now consider $\mathbf{q} = \mathrm{SoftMax}(a\ \mathbf{s}; \alpha)$ for some a. If $a = 1$, obviously $\mathbf{p} = \mathbf{q}$. Otherwise, odds p_i/p_j are different from odds q_i/q_j, because from Fact 2, $p_i/p_j = \exp(\alpha(s_i - s_j))$ and $q_i/q_j = \exp(\alpha(a\ s_i - a\ s_j))$, the latter of which is the same as $\exp(\alpha(s_i - s_j))^a$ and further reduces to $(p_i/p_j)^a$. Since by assumption $s_i > s_j$, we know that $\exp(\alpha(s_i - s_j)) > 1$, and so $p_i/p_j \neq (p_i/p_j)^a$ (since $a \neq 1$). □

Proof of Fact 8.

$$p_i = \frac{\exp(\frac{\alpha}{a}\ a\ s_i)}{\sum_j \exp(\frac{\alpha}{a}\ a\ s_j)} = \frac{\exp(\frac{\alpha\ a\ s_i}{a})}{\sum_j \exp(\frac{\alpha\ a\ s_j}{a})} = \frac{\exp(\alpha\ s_i)}{\sum_j \exp(\alpha\ s_j)}$$

□

Proof of Corollary 9. Since by Fact 8 we have $\mathrm{SoftMax}(\mathbf{s}; \alpha) = \mathbf{p} = \mathrm{SoftMax}(a\ \mathbf{s}; \alpha/a)$, setting $a = \log b$ for some $b > 0$, we obtain:

$$p_i = \frac{\exp\left(\frac{\alpha}{\log b} s_i \log b\right)}{\sum_j \exp\left(\frac{\alpha}{\log b} s_j \log b\right)} = \frac{\exp(\log b)^{\frac{\alpha}{\log b} s_i}}{\sum_j \exp(\log b)^{\frac{\alpha}{\log b} s_j}} = \frac{b^{\frac{\alpha}{\log b} s_i}}{\sum_j b^{\frac{\alpha}{\log b} s_j}}$$

□

Proof of Fact 10. Let $C = \sum_k \exp s_k$ be the normalizing constant for SoftMax($\mathbf{s}$, 1). Moreover, assume that $\mathbf{q} = \mathrm{Pow}(\mathrm{SoftMax}(\mathbf{s}, 1); \alpha)$. Then:

$$q_i = \frac{\left(\frac{\exp s_i}{C}\right)^{\alpha}}{\sum_j \left(\frac{\exp s_j}{C}\right)^{\alpha}} = \frac{C^{-\alpha} (\exp s_i)^{\alpha}}{\sum_j C^{-\alpha} \left(\exp s_j\right)^{\alpha}} = \frac{\exp(\alpha s_i)}{\sum_j \exp(\alpha s_j)}$$

□

C R code for solving for α given expected scores

```r
softmax = function(scores, alpha) {
  # Calculate the softmax probabilities with a temperature parameter.
  #
  # Parameters:
  #   scores: Numeric vector of scores for each option.
  #   alpha:  Temperature parameter controlling the level of exploration.
  #
  # Returns:
  #   Numeric vector of probabilities obtained using the softmax function.
  #
  # Example:
  #   probabilities <- softmax(scores = c(0, 0.5, 1), alpha = 1.0)
  return (exp(alpha*scores) / sum(exp(alpha*scores)))
}

solve_alpha = function(scores, average_score, upper_bound = 100) {
  # Solve for the optimal alpha parameter using the softmax function.
  #
  # Parameters:
  #   scores: Numeric vector of scores for each option.
  #   average_score: Target average score for the softmax policy.
  #   upper_bound: Upper bound for the alpha parameter (default is 100).
  #
  # Returns:
  #   A list with the root field representing the optimal alpha parameter.
  #
  # Example:
  #   result <- solve_alpha(scores = c(0, 0.5, 1), average_score = 0.75)
  #   optimal_alpha <- result$root
  uniroot(interval = c(0,upper_bound),
          f = function(a) {
            policy = softmax(scores, a)
            return (policy %*% scores - average_score)
          })
}

# example (and sanity check)
scores <- c(0, 0.5, 1)
alpha_star <- solve_alpha(scores, 0.75)$root
(alpha_star)
(sm_policy <- softmax(scores, alpha_star))
dot(sm_policy, scores)
```

Listing 1: Example R code for estimating the value of α for a given vector of scores and a target value for the expected score.

References

Anderson, J. R. (1990). *The adaptive character of thought*. Lawrence Erlbaum.

Chater, N., & Oaksford, M. (2000). The rational analysis of mind and behavior. *Synthese*, *122*, 93–131.

Goeree, J. K., Holt, C. A., & Palfrey, T. R. (2008). Quantal response equilibrium. In S. N. Durlauf & L. E. Blume (Eds.), *The new palgrave dictionary of economics*. Palgrave Macmillan.

Hagen, E. H., Chater, N., Gallistel, C. R., Houston, A., Kacelnik, A., Kalenscher, T., Nettle, D., Oppenheimer, D., & Stephens, D. W. (2012). Decision making: What can evolution do for us? In P. Hammerstein & J. R. Stevens (Eds.), *Evolution and the mechanisms of decision making* (pp. 97–126). MIT Press.

Lee, M. D., & Wagenmakers, E.-J. (2015). *Bayesian cognitive modeling: A practical course*. Cambridge University Press.

Lewandowsky, S., & Farrell, S. (2011). *Computational modelling in cognition: Principles and practice*. Sage Publications.

Lieder, F., & Griffiths, T. L. (2019). Resource-rational analysis: Understanding human cognition as the optimal use of limited computational resources. *The Behavioral and Brain Sciences*, *43*, e1.

Luce, D. R. (1959). *Individual choice behavior: A theoretical analysis*. Wiley.

McNamara, J. M. (2013). Towards a richer evolutionary game theory. *Journal of The Royal Society Interface*, *10*(88), 1–9.

Sandholm, W. H. (2010). *Population games and evolutionary dynamics*. MIT Press.

Schad, D. J., Betancourt, M., & Vasishth, S. (2021). Toward a principled bayesian workflow in cognitive science. *Psychological Methods*, *26*(1), 103–126.

Simon, H. A. (1959). Theories of decision-making in economics and behavioral science. *American Economic Review*.

Train, K. E. (2009). *Discrete choice methods with simulation*. Cambridge University Press.