Supplementary Information

Table S1

	Column 1	Column 2	Column 3	Column 4	Column 5
Fig.4a $\boldsymbol{s = 0.1, \tau = 0.1,}$ $\boldsymbol{c_i = 0, c_j = 0}$	$w = 0.4$	$w = 0.45$	$w = 0.5$	$w = 0.55$	$w = 0.6$
Fig.4b $\boldsymbol{w = 0.5, \tau = 0.1, c_i = 0, c_j = 0}$	$s = 0.01$	$s = 0.0325$	$s = 0.055$	$s = 0.0775$	$s = 0.1$
Fig.4c $\boldsymbol{w = 0.5, \tau = 0.1,}$ $\boldsymbol{s = 0.1}$	$c_i = -0.04$ $c_j = 0.04$	$c_i = -0.02$ $c_j = 0.02$	$c_i = 0$ $c_j = 0$	$c_i = 0.02$ $c_j = -0.02$	$c_i = 0.04$ $c_j = -0.04$
Fig.4d $\boldsymbol{s = 0.1, \tau = 0.1,}$ $\boldsymbol{s = 0.1}$	$c_i = -0.04$ $c_j = -0.04$	$c_i = -0.02$ $c_j = -0.02$	$c_i = 0$ $c_j = 0$	$c_i = 0.02$ $c_j = 0.02$	$c_i = 0.04$ $c_j = 0.04$

Table S1. Parameters used to generate the decoy influence maps in Fig. 4a-d.

Table S2

	$\boldsymbol{c_i, c_j}$	$\boldsymbol{s}$	$\boldsymbol{\beta_1}$	$\boldsymbol{\beta_2}$	$\boldsymbol{k}$	$\boldsymbol{\tau}$
Vanilla divisive normalisation	$free$	1	0	1	0.001	$free$
Recurrent divisive normalisation	$free$	1	1	1	0.001	$free$
Adaptive gain	$free$	$free$	1	1	1	$free$
Range normalisation	0	1	0	0	0.001	$free$

Table S2. Parametrisations of grandmother model corresponding to computational schemes proposed in the literature.

Model derivations

For the grandmother model, we facilitated comparison between the adaptive gain model (which incorporates a logistic transformation) with other normalisation models (which do not), by leveraging the following limit:

$$\lim_{k \to 0} \frac{x^k - 1}{k} = \log_e x \, , \forall \, x > 0$$

[S1]

Thus, substituting x above for our variables of interest, $v(A_i)$ and μ, we obtain:

$$\log v(A_i) - \log \mu_i = \lim_{k \to 0} \frac{v(A_i)^k - 1}{k} - \frac{{\mu_i}^k - 1}{k} = \lim_{k \to 0} \frac{v(A_i)^k - {\mu_i}^k}{k}$$

[S2]

Parameter k thus corresponds to the extent to which inputs are compressed before being transduced, with $k{\sim}0$ signifying a log compression, and $k{\sim}1$ signifying linear input. Our results revealed that variations in parameter k, which allow us to interpolate between those two schemes, do not result in impactful systematic changes of the predicted pattern of decoy influence. This is consistent with simulations of the transfer function under different values of parameters k and s, whereby the transfer function retains a consistent functional form across values of k, when $s < 1$ (**Fig. S4**).

Simulations revealed that implementing a value of k as high as 0.001 satisfactorily approximates the natural logarithm (**Fig. S5**). Consequently, when we plug in the parameter values for vanilla divisive normalisation specified in **Table S2** $(c_i, c_j = free, s = 1, \beta_1 = 0, \beta_2 = 1, k = 0.001)$ into the general grandmother model eq. 7, we may substitute in the relationship outlined above in eq. S2, arriving at the formulation of recurrent divisive normalisation as per eq. 1:

$$u(A_i)^{DN} = \frac{1}{e^{-(v(A_i)^{0.001} - {\mu_i}^{0.001})(0.001)^{-1}}} = \frac{1}{e^{-\,(\log v(A_i) - \log \mu_i)}} = \frac{1}{\frac{\mu_i}{v(A_i)}} = \frac{v(A_i)}{\mu}$$

$$= \frac{v(A_i)}{v({avg^{ABD}}_i) + c_i}$$

[S3]

Similarly, we may simplify the grandmother model (eq. 7) into recurrent divisive normalisation (eq. 2) by plugging in the relevant parameter values $(c_i, c_j = free, s = 1, \beta_1 = 1, \beta_2 = 1, k = 0.001)$ specified in **Table S2:**

$$u(A_i)^{RDN} = \frac{1}{1 + e^{-(v(A_i)^{0.001} - {\mu_i}^{0.001})(0.001)^{-1}}} = \frac{1}{1 + e^{-(\log v(A_i) - \log \mu_i)}} = \frac{1}{1 + \frac{\mu_i}{v(A_i)}}$$
$$= \frac{v(A_i)}{v(A_i) + \mu_i} = \frac{v(A_i)}{v(A_i) + v({avg^{ABD}}_i) + c_i}$$

[S4]

Note that if we allow parameter s to vary freely, it assumes the role of a power transform for inputs:

$$u(A_i)^{RDN+s} = \frac{1}{1 + e^{-(v(A_i)^{0.001} - {\mu_i}^{0.001})(s \cdot 0.001)^{-1}}} = \frac{1}{1 + e^{-(\log v(A_i) - \log \mu_i) s^{-1}}}$$
$$= \frac{1}{1 + \frac{{\mu_i}^{s^{-1}}}{v(A_i)^{s^{-1}}}} = \frac{v(A_i)^{s^{-1}}}{v(A_i)^{s^{-1}} + {\mu_i}^{s^{-1}}} = \frac{v(A_i)^{s^{-1}}}{v(A_i)^{s^{-1}} + (v({avg^{ABD}}_i) + c_i)^{s^{-1}}}$$

[S5]

Similarly, we may obtain the formulation of range normalisation eq. 6 by plugging in the relevant parameter values $(c_i, c_j = 0, s = 1, \beta_1 = 0, \beta_2 = 0, k = 0.001)$ from **Table S2** into the general grandmother model (eq. 7):

$$u(A_i)^{RN} = \frac{1}{e^{-(v(A_i)^{0.001} - {\mu_i}^{0.001})(0.001)^{-1}}} = \frac{1}{e^{-(\log v(A_i) - \log \mu_i)}} = \frac{1}{\frac{\mu_i}{v(A_i)}} = \frac{v(A_i)}{\mu_i}$$
$$= \frac{v(A_i)}{v({rng^{ABD}}_i)}$$

[S6]

Note that while the expression above in eq. S6 does not contain the scaling term β_1 from eq. 6, in the context of the larger grandmother model, its function is absorbed by the weighting parameter w, as in eq. 4.

Finally, we may obtain the formulation of the adaptive gain model (eq. 3) by plugging in the relevant parameter values $(c_i, c_j = free, s = free, \beta_1 = 1, \beta_2 = 1, k = 1)$ from **Table S2** into the general grandmother model (eq. 7):

$$u(A_i)^{AG} = \frac{1}{1 + e^{-(v(A_i) - \mu_i)\, s^{-1}}} = \frac{1}{1 + e^{-(v(A_i) - (c_i + v({avg^{ABD}}_i))\, s^{-1}}}$$
$$= \frac{1}{1 + e^{-(v(A_i) - v({avg^{ABD}}_i) - c_i) s^{-1}}}$$

[S7]

Fig. S1

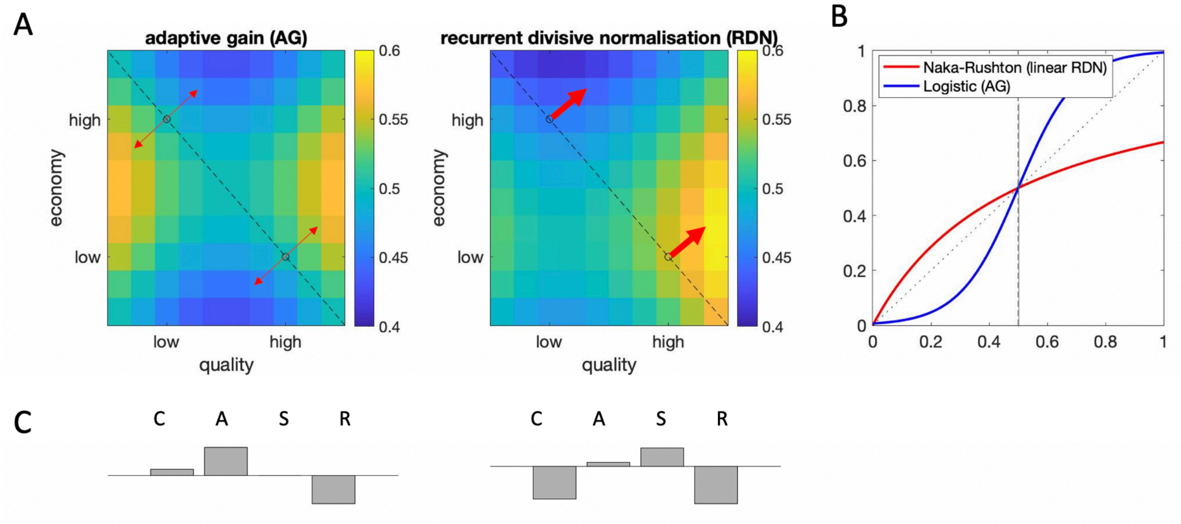

Figure S1. The function of the exponential nonlinearity in the adaptive gain model. **A**. Simulated decoy maps for the adaptive gain (eq. 3) and recurrent divisive normalisation (eq. 2) models. It is noteworthy that unlike recurrent divisive normalisation, the adaptive gain model (with bias terms $c_i = c_j = 0$) produces symmetrical regions of repulsion and attraction around the line of isopreference, whereas recurrent divisive normalization produces stronger repulsion and attraction for superior decoys (i.e. decoys above rather than below the isopreference line). **B**. To understand this difference, it is helpful to consider the form of the gain function (or transducer) implied by either account. The asymmetrical pattern of decoy influence seen for linear RDN occurs because this transducer is a (decelerating) Naka-Rushton function:

$$f^{lRDN}(x) = \frac{x}{x + \mu}$$

This function is concave, meaning that the derivative is always higher (i.e. curve steeper) for low than high attribute values irrespective of the value of x. By contrast, the gain function for AG is sigmoidal:

$$f^{AG}(x) = \frac{1}{1 + e^{-\frac{(x-\mu)}{s}}}$$

and it is thus symmetric around the midpoint (which is itself adjusted to the context; this is the “adaptive” part). In the rightmost panel above we have plotted the transductions applied to inputs by the Naka-Rushton function in red and by the Sigmoidal function in blue, with $x \in [0,1]$, μ = 0.5, s = 0.1. Notably, the Naka-Rushton function applies a non-symmetric mapping of inputs onto outputs for values lower than the additive constant (μ = 0.5, dashed line) compared to values higher than it, whereby the gain is higher for lower values. By contrast, the sigmoidal function applies a symmetric compression around the inflection point (μ = 0.5, dashed line). The recurrent divisive normalisation framework (without additional nonlinearity) implies that those attribute values which are always relatively smaller (closer to zero) are processed with higher gain. By contrast, the adaptive gain model implies that resources are allocated preferentially to the mean of a context, exaggerating binary distinctions which potentially straddle that midpoint (e.g. “low” vs. “high” value). One might term the former “prothetic” normalisation and the latter “metathetic” normalisation, because they differ in their assumptions about whether feature spaces have a “way up” or intrinsic magnitude-like representation, or whether they are fundamentally two ends of a symmetric continuum. **C**. The relative strengths of the compromise, attraction, similarity and repulsion effects per the two decoy maps.

Fig. S2

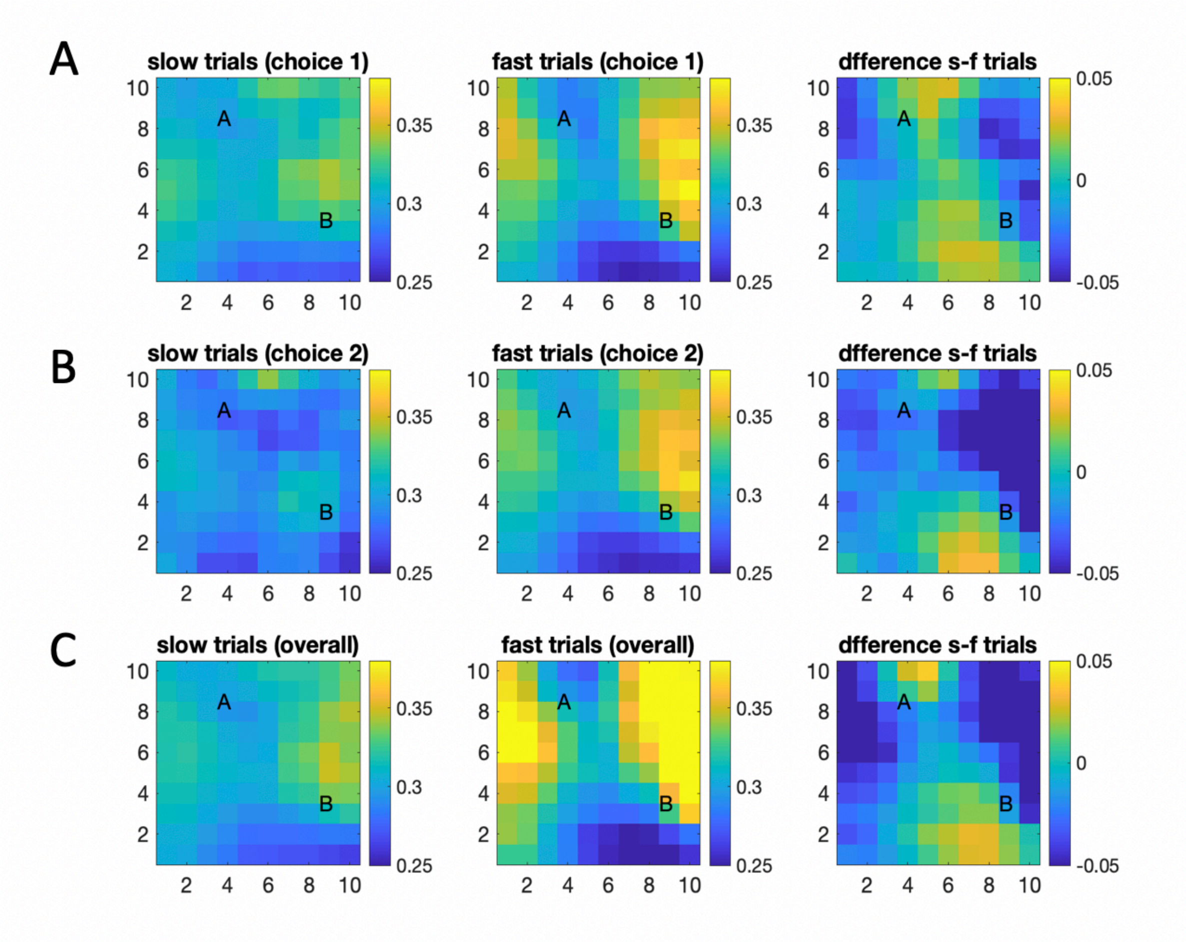

Figure S2. Decoy maps according to response latencies. **A**. Trials were split into fast and slow according to the median reaction time of each participant for the first choice. **B**. Trials were split into fast and slow according to the median reaction time of each participant for the second choice. **C**. Trials were split into fast and slow according to the overall median reaction time of each participant.

It appears that response time did not affect the overall form of the decoy effects at all, with qualitatively equivalent decoy influence maps observed for fast and slow trials on both the first and second of the sequential ranking judgments made on each trial. Despite their equivalence in form of the influence maps, we observe that decoy effects are more pronounced in fast relative to slow trials. This pattern emerges consistently regardless of whether we split the data according to time at first choice, second choice, or overall time. We note in passing that this reaction time effect can be captured by varying parameter s in the adaptive gain model, i.e. the overall slope of the transducer or gain of processing between fast and slow trials. At first glance, these results may appear to contradict the established finding whereby decoy effects are diminished under time pressure (e.g. for faster trials). However, it is important to note that our task involved a free response paradigm rather than a fixed response paradigm; it is thus likely that on average, those trials with overall low gain (e.g. where participants paid less attention) were those where slower responses were made, for instance because they were temporarily distracted. We believe that these findings constitute an interesting empirical contribution.

Fig. S3

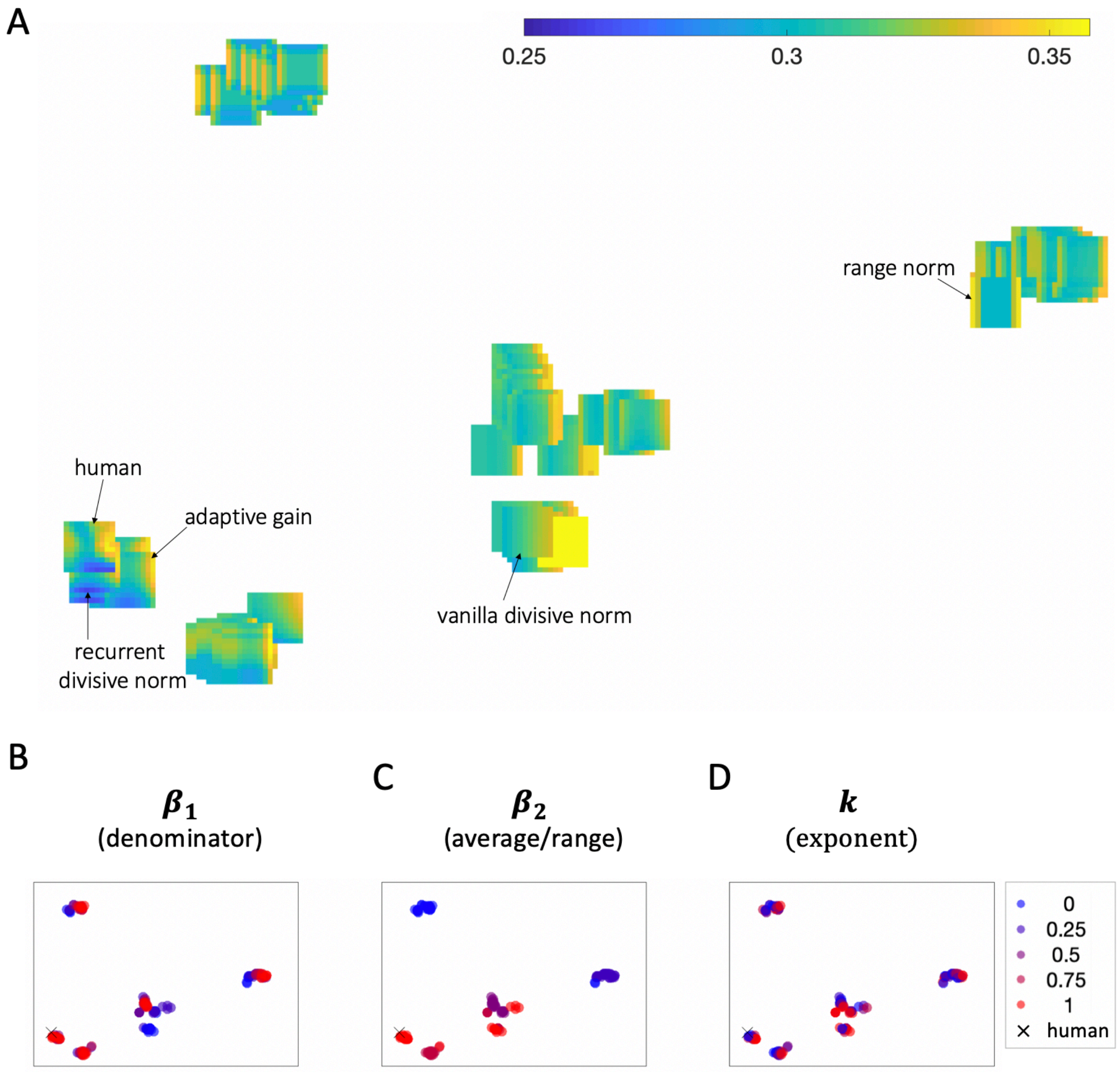

Figure S3. Embedding space for normalisation models of decoy effects, as in **Fig. 5**. **A**. *t*-distributed stochastic neighbour visualisation of the maps of decoy influence produced by different variants of the grandmother model. Each map represents a variant of the grandmother model positioned in 2D space such that models with similar decoy influence patterns are nearby, while models with more different decoy patterns are further apart. Heat maps illustrate decoy influence. **B-D**. Each model-produced decoy map is denoted as a dot and colour coded to indicate parameter value: β_1 (panel B), β_2 (panel C), or k (panel D). Human data is represented with a cross.

Fig. S4

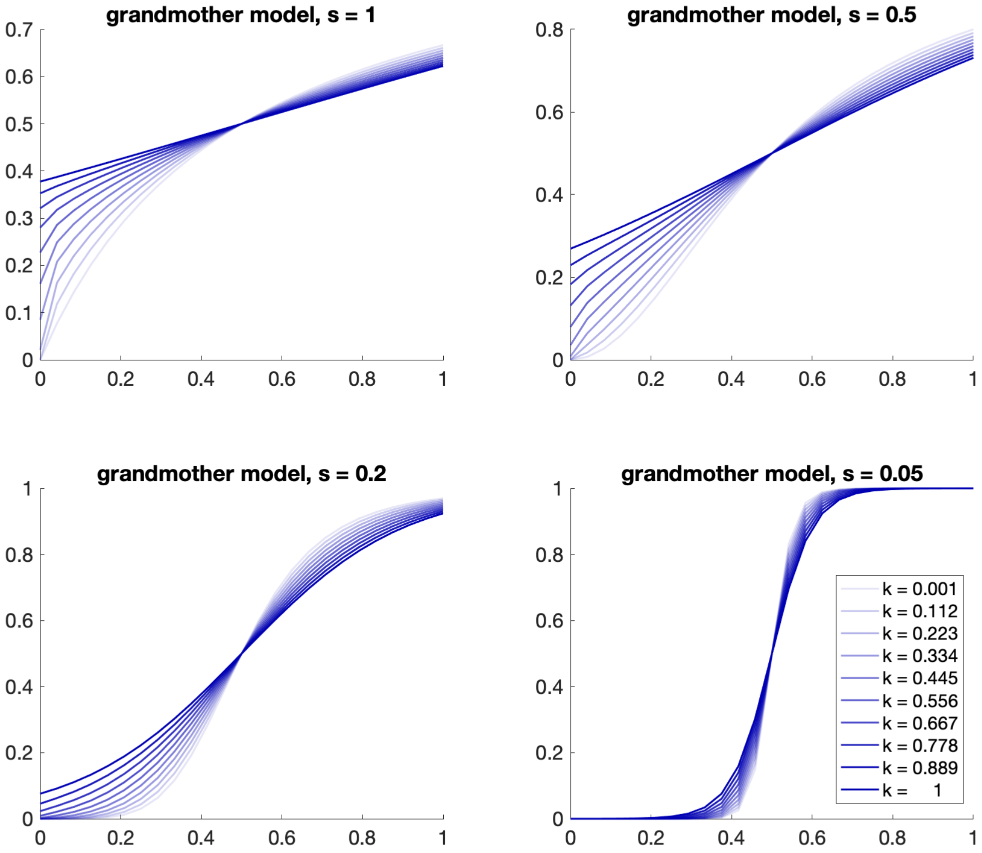

Figure S4. Illustration of the transfer function of the grandmother model, with $c_i = c_j = 0$, $\beta_1 = \beta_2 = 1$, and $v(avg_i) = .5$, across different values of k and s. Note that with this parametrisation, the transfer function is equivalent to a logistic function when $k = 1$, and to a Naka-Rushton function when $k \sim 0$ and $s = 1$, as in **Fig. S1**.

Fig. S5

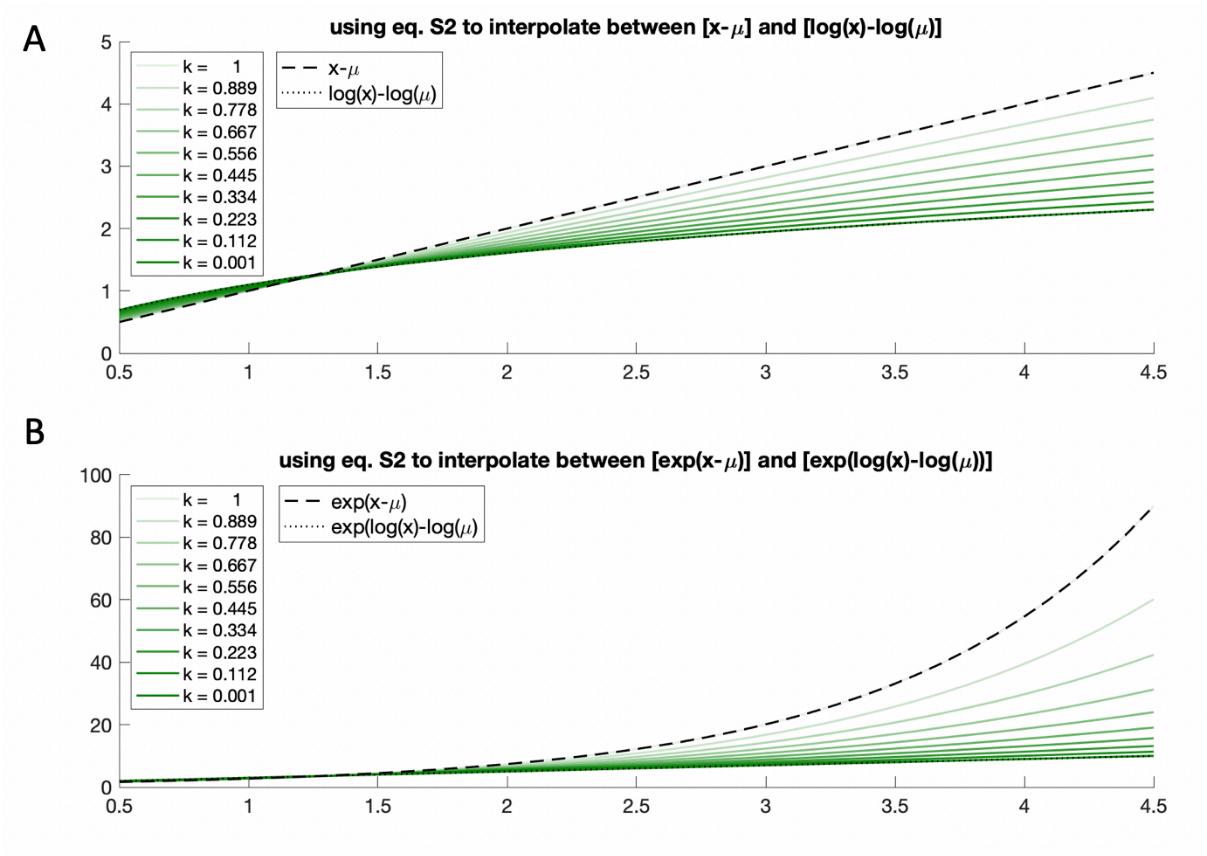

Figure S5. Illustration of the impact of varying parameter k between 0 and 1 in $f(x) = \frac{(x^k - \mu^k)}{k}$, as per eq. S2, with $\mu = .5$. **A**: When k approaches zero (darkest green curve), $f(x) \approx \log x - \log \mu$ (dotted black curve), and when $k = 1$ (lightest green curve), $f(x) = x - \mu$ (dashed black curve). Variations in parameter k (green curves) interpolate between those two functions. **B**: Substituting the functions above into an exponential function, as in the logistic formulation of the grandmother model (eq. S7). When k approaches zero (darkest green curve), $f(x)$ approaches $e^{(\log x - \log \mu)} = \frac{x}{\mu}$ (dotted line), allowing us to recover linear normalisation models, as per eq. S3-S5.