Understanding the complexity of computational models through optimization and sloppy parameter analyses: The case of the Connectionist Dual-Process Model

Conrad Perry, Rick Evertz, Marco Zorzi, Johannes C. Ziegler

Contents lists available at ScienceDirect

Journal of Memory and Language

journal homepage: www.elsevier.com/locate/jml

Understanding the complexity of computational models through optimization and sloppy parameter analyses: The case of the Connectionist Dual-Process Model

Conrad Perry [a,*], Rick Evertz [b], Marco Zorzi [c,d], Johannes C. Ziegler [e]

[a] Faculty of Health and Medical Sciences, The University of Adelaide, Adelaide, Australia
[b] Swinburne University of Technology, Department of Physics and Astronomy, Hawthorn, Australia
[c] Department of General Psychology and Padova Neuroscience Center, University of Padova, Padova, Italy
[d] IRRCS San Camillo Hospital, Venice-Lido, Italy
[e] Aix-Marseille University and Centre National de la Recherche Scientifique, Laboratoire de Psychologie Cognitive, Marseille, France

ARTICLE INFO

Keywords:
Reading
Optimization
Sloppy parameters
Computational modelling

ABSTRACT

A major strength of computational cognitive models is their capacity to accurately predict empirical data. However, challenges in understanding how complex models work and the risk of overfitting have often been addressed by trading off predictive accuracy with model simplification. Here, we introduce state-of-the-art model analysis techniques to show how a large number of parameters in a cognitive model can be reduced into a smaller set that is simpler to understand and can be used to make more constrained predictions with. As a test case, we created different versions of the Connectionist Dual-Process model (CDP) of reading aloud whose parameters were optimized on seven different databases. The results showed that CDP was not overfit and could predict a large amount of variance across those databases. Indeed, the quantitative performance of CDP was higher than that of previous models in this area. Moreover, sloppy parameter analysis, a mathematical technique used to quantify the effects of different parameters on model performance, revealed that many of the parameters in CDP have very little effect on its performance. This shows that the dynamics of CDP are much simpler than its relatively large number of parameters might suggest. Overall, our study shows that cognitive models with large numbers of parameters do not necessarily overfit the empirical data and that understanding the behavior of complex models is more tractable using appropriate mathematical tools. The same techniques could be applied to many different complex cognitive models whenever appropriate datasets for model optimization exist.

Computational modelling has progressively taken central stage in most areas of cognitive psychology. Rather than simply offering a mathematical description of the data, computational cognitive models make commitments about the mental processes and representations underlying a psychological phenomenon (e.g., Farrell & Lewandowsky, 2018). Therefore, computational models explain and predict cognition and behavior, and their elements can be interpreted in psychological terms.

In this article, we focus on reading aloud because it is an advanced area of computational cognitive modelling. Indeed, since the nineteen-eighties, a large number of computational models have been proposed that simulate various aspects of reading, including reading aloud (Coltheart et al., 2001; Perry et al., 2007; 2010b; 2013; Plaut et al., 1996; Seidenberg & McClelland, 1989; Smith et al., 2021; Zorzi et al., 1998), lexical decision (Grainger & Jacobs, 1996; Ratcliff et al., 2004), learning-to-read and dyslexia (Chang et al., 2019; Harm & Seidenberg, 1999; Perry et al., 2019; Ziegler et al., 2020), and eye movements (Engberg et al., 2005; Reichle et al., 2003). Reading is a highly complex task and models need to specify the transformations, functions, and representations that explain such complex behaviour. Even reading isolated words involves the complex interplay between visual, phonological, and semantic information.

The emphasis on the explanatory value of modelling and its role in exploring theoretical ideas (e.g., McClelland, 2009) can be a source of tension between model simplicity and descriptive adequacy (e.g., Jacobs & Grainger, 1994). Indeed, working out the best trade-off between complexity and goodness of fit can be difficult even if only considering statistical arguments (Pitt et al., 2002; Pitt & Myung, 2002) let alone

* Corresponding author.
E-mail address: ConradPerry@gmail.com (C. Perry).

https://doi.org/10.1016/j.jml.2023.104468
Received 27 February 2023; Received in revised form 10 September 2023; Accepted 11 September 2023

broader ones (e.g., Roberts & Pashler, 2000; Rogers & Rowe, 2002). In the context of computational cognitive modelling, this also brings to the table the issue of model simplification (see McClelland, 2009, for a thorough discussion).

Simplification refers to the formulation of the model in terms of its processes and representations and is thought to aid understanding. Although complex models may be better at predicting behavior than simpler models, they might also be much harder to understand. The corollary of this argument is that a simple model might be preferred over a complex one in spite of limitations in the phenomena addressed and potential inaccuracies in its predictions (e.g., Lee & Webb, 2005). In relation to computational models of reading aloud, Seidenberg (2012) described the emphasis on accurately simulating a broad range of empirical phenomena as a "data fitting orientation….that (has) misread the history of cognitive science and cognitive modeling" (p. 200). The idea advocated by Seidenberg (2012) is that computational models are useful as tools for exploring theoretical principles, whereas aiming at predictive accuracy may lead to heavily engineered and potentially overfit models.

What does over-fitting mean in the context of a computational model? In contrast to mathematical models, which simply offer a formal description of the data and thus allow modellers to use as many or as few model parameters as they need to fit the data, each parameter in a computational model corresponds to some commitment about there being a specific processing component within the overall system represented by the model.[1] Thus, the number of parameters in a computational model is completely constrained by the underlying theory that is being modelled. Yet, some computational models, like those of the connectionist dual process family (see below), have been criticized for over-fitting the data (Seidenberg, 2012; Seidenberg & Plaut, 1998).

The argument of over-fitting has been presented in various ways in the literature. First, since the publication of CDP+ (2007), there has been a strong emphasis on a model's ability to account for item-level variance in large-scale databases. Indeed, CDP+ accounted for 17% of the item-level variance in the Spieler and Balota (1997) database, while its competitors (DRC and PDP) remained below 4% of the variance. However, it is easy to argue that CDP+'s larger number of parameters allowed its authors to "tweak" some of them to obtain better fits. If modellers are allowed to change their parameters for each data set they simulate, there is a real danger of over-fitting in the sense that parameters could take any value as long as this helps to fit the empirical data. In this scenario, the commitment to stable processing components (stable parameters) is traded off against optimizing performance independent of the underlying theory in a purely post-hoc fashion.

A second argument is on the choice of benchmark effects and whether they might not necessarily reflect the core processes of visual word recognition and naming. That is, there are findings from some small-scale experiments in the literature that are potentially outliers. For example, the letter length effect with nonwords reported by Weekes (1997) is very strong compared to some studies. Alternatively, the spelling-sound irregularity effect where words with an irregular spelling-sound correspondence early in their letter seqeuence (e.g., *Th*ai*)* are slower to read than words with such correspondences later in their letter sequence (e.g., p*i*nt) found by Rastle and Coltheart (1998) relies on a tailored set of items and produces results that are not always significant by items (see Zorzi, 2000). Nonword stimuli that are not very word-like have also been used, and these are arguably not likely to be read using the normal cognitive processes (see Perry, 2018). By including such effects in the benchmarks and fitting models to the results of such experiments, it could be argued that the modelers are over-fitting their models to outlier experiments that do not reflect the core processes of reading.

[1] Note the notion of model parameters used in the present article does not apply to the connection weights of artificial neural networks.

A third argument is about accounting for variance caused by processes that are beyond the scope of the model (Seidenberg & Plaut, 1998). For example, in reading aloud, a large portion of the variance is accounted for by the acoustic characteristics of the initial phoneme, a rather uninteresting phenomenon (at least for understanding the cognitive processes used in reading) due to the fact that voice keys fail to accurately measure the onset of voiced versus unvoiced consonants. Given that current models do not implement a speech production mechanism that would allow them to capture such results, tweaking other parameters of the models to capture some initial phoneme variance could be considered as over-fitting because that part of the variance should not be explained (Kawamoto et al., 2008).

Finally, there has been an argument about how much variance there is to be explained in the empirical data (Adelman et al., 2013; Perry et al., 2010a; Rey et al., 2013; Seidenberg & Plaut, 1998). That is, large-scale databases might be contaminated by undesirable sources of error variance and to understand how well a model is performing, one would need to estimate the amount of experimental noise that models cannot or should not try to account for (Rey et al., 2013; Seidenberg & Plaut, 1998). In this respect, Perry et al. (2010a) looked at the variance that is common across the same items in different databases. They found that performance to identical items across four different databases correlated moderately, with r values ranging between .42 and .68 (i.e., 17.6% and 46.2% of the variance). Thus, if models account for more variance than there is to be explained, they are likely fitting undesirable sources of error variance.

In this article, we use state-of-the-art optimization and model analysis techniques to show how the effect of parameters in complex computational models can be investigated. We will show that the number of parameters that meaningfully affect the performance of a model may be far less than the total number of parameters. This makes understanding such models far more tractable because parameters that have little effect on performance can be largely ignored in many situations. Having fewer parameters that meaningfully affect model performance may also limit the possibility of overfitting data when optimisation techniques are used.

The goal of the article is twofold: First, we use model optimization techniques to examine the extent to which optimization really causes overfitting and affects model behavior. Second, we introduce the sloppy parameter analysis (SPA) technique initially proposed by Gutenkunst et al. (2007a) to quantify the extent to which different parameters affect the model's behavior when changed. To the best of our knowledge, SPA has never been applied to psychological or cognitive modeling in the way described here despite its popularity in several other areas of science (e.g., Gutenkunst et al., 2007a; Hartoyo et al., 2019; Transtrum et al., 2015). Specifically, SPA will allow us to examine which parameters are "stiff" (i.e., have a strong effect on model performance) and which parameters are "sloppy" (i.e., have little effect on model performance). The combination of model optimization with SPA provides insights into how complex computational models really are and which parameters are important for the predictions they generate. Before introducing these techniques, we provide a brief description of the computational model that will be used as a test-case.

Modeling reading aloud and the CDP model

The connectionist dual process (CDP) model is a state-of-the-art computational model of reading aloud that has been progressively refined and extended over the past 20 years in order to account for a large number of benchmark empirical phenomena encompassing skilled performance (Perry et al., 2007, 2010b, 2013; Zorzi et al., 1998), acquired dyslexia (Perry et al., 2007), reading acquisition and developmental dyslexia (Perry et al., 2019; Ziegler et al., 2014), reading in non-English orthographies such as Italian, German, and French (Perry et al., 2010a, 2014a, 2014b), and bilingual reading (Paulesu et al., 2021). CDP models have been the most successful in terms of quantitative fit to

behavioral data when compared to their competitors, such as those from the connectionist triangle model family (Plaut et al., 1996; Seidenberg & McClelland, 1989). Yet, CDP models are also more complex with respect to the number of representational levels or layers, input and output units, and type and number of inter- and intra-level connections.

As can be seen in Fig. 1, the representational levels in CDP can be broken down into two main groups based on whether they act as a memory-retrieval system for familiar words (Lexical Route) or are used for computing the likely pronunciation of letter strings (including novel words and nonwords) based on learned spelling-sound associations (Sublexical Route). There are also two sets of shared representations. One consists of the letter feature and letter nodes and the other consists of the phoneme and stress output buffers (see Fig. S1 for a more complete picture).

The input into CDP begins at the letter feature level where early visual representations of letters are activated. These then activate abstract letter representations. Once this has happened, the lexical route can be activated. The first representation in the lexical route is an orthographic lexicon. This holds written word representations. Next is the phonological lexicon, which holds spoken words, and the semantic system, which contains semantic information that the words can access. All of these are considered representationally separate and each representation has parameters that control the speed at which they can activate other representations. Once the phonological lexicon is activated, it can then activate phonemes and stress in the phoneme and stress output buffers. The letter level is also the input for the sublexical route. In this route, graphemes are first extracted from letters (graphemic parsing) and then put into the two-layer associative (TLA) network. The TLA network then computes the likely phoneme and stress pattern from the graphemes. The predicted phoneme and stress information is then pooled with memory-based information retrieved by the lexical route (for familiar letter strings) at the level of the phoneme and stress output buffers. Speech output then occurs when there are phonemes and stress nodes that have activation levels above minimum naming and minimum stress criteria and there are no other phonemes or stress nodes which are increasing in activation below those criteria. All 27 parameters examined later in this paper can be found in Table 1.

Simple is better

In many scientific domains, it is often assumed that the fewer parameters a model has the better (e.g., Lever, Krzywinski, & Altman, 2016). Whilst this is a reasonable assumption in many cases, CDP is a psychologically plausible model that aims at capturing all of the relevant processes involved in reading. Parameters are thus needed for many of these processes even if they do not allow the model to pick up large amounts of variance. Indeed, as Zorzi (2010) has already shown, it is possible to replace the 15 parameters that were in the lexical route of the original CDP+ model (Perry et al., 2007) with just two parameters that were used in the first CDP model (Zorzi et al., 1998) and very similar results are found.

The idea that models might be useful even with a large number of parameters has been investigated in many fields including biology, neuroscience, chemistry, physics, and psychology (e.g., Gutenkunst et al., 2007a; Hartoyo et al., 2019; Holmes et al., 2016; Transtrum et al., 2015). Large numbers of parameters are not necessarily considered bad as long as it can be understood how they affect performance and the limits of model behaviour, or indeed from a practical point of view, whether the model works well. This is different to many mathematical/statistical models where the number of parameters can be increased just to obtain better fits. We will show using optimization and model analysis techniques how this can be achieved with models in psychology with large numbers of parameters, notably CDP, where we will investigate 27 parameters. This will allow us to show how the individual parameters of CDP affect reading behaviour.

There is also a large literature in mathematical psychology on the complexity of cognitive models and how it might be evaluated (e.g., Pitt & Myung, 2002; Pitt et al.,2002; Veksler et al., 2015). Pitt and Myung (2002), for example, discuss the usefulness of a number of different measures to quantify model complexity. They suggest that simple measures that only use the number of parameters to measure complexity are not likely to be especially useful for evaluating cognitive models (which would include CDP). They suggest that the functional form of a model needs to be taken into account, where one has to investigate the number of distinct probability distributions a model can generate based on movements of its parameters. Pitt and Myung (2002) call this geometric complexity and suggest ways it could be measured. Other

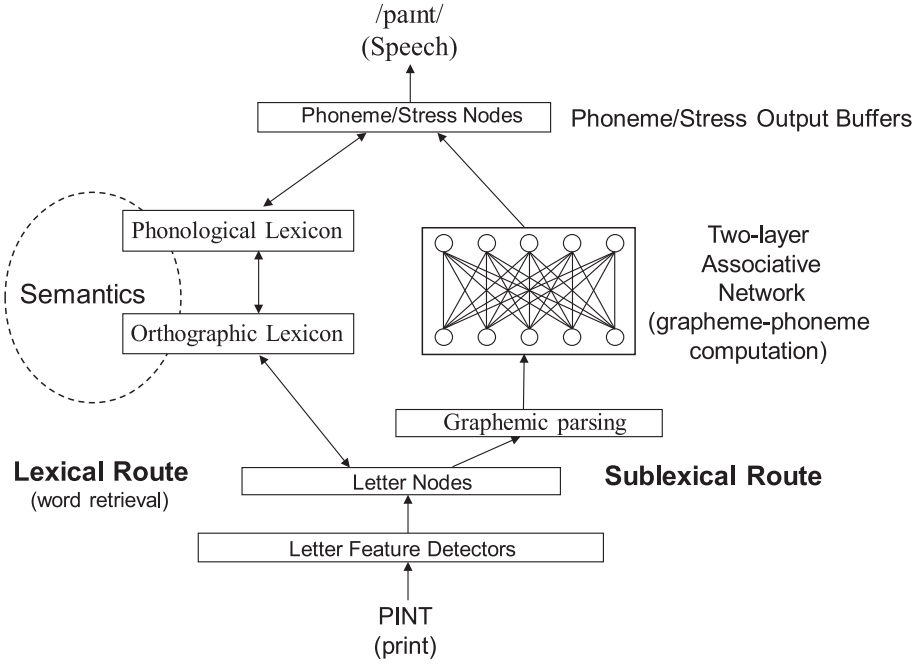

Fig. 1. The Connectionist Dual-Process Model of Reading Aloud (CDP++.parser).

Table 1
Parameters that the models were optimized on and the highest and lowest bounds which they could take.

	Lowest Level	Highest Level	Notes
Feature to letter excitation	0.002	0.01	Lexical route parameter
Feature to letter inhibition	−2	−0.5	Lexical route parameter
Letters lateral inhibition	−1	0	Lexical route parameter
Letter to orthographic lexicon excitation	0.01	0.1	Lexical route parameter
Letter to orthographic lexicon inhibition	−2.5	−0.1	Lexical route parameter
Orthographic lexicon lateral inhibition	−0.2	0	Lexical route parameter
Orthographic lexicon to letter excitation	0	0.1	Lexical route parameter
Orthographic lexicon to letter inhibition	−0.1	0	Lexical route parameter
Orthographic lexicon to phonological lexicon excitation	0.5	2.5	Lexical route parameter
Phonological lexicon lateral inhibition	−0.2	0	Lexical route parameter
Phonological lexicon to phoneme excitation	0.05	0.2	Lexical route parameter
Phonological lexicon to phoneme inhibition	−0.2	−0.02	Lexical route parameter
Phonological lexicon to orthographic lexicon excitation	0.1	4	Lexical route parameter
Phoneme to phoneme lateral inhibition	−0.2	0	Lexical route parameter
Phoneme to phonological lexicon excitation	0	0.2	Lexical route parameter
Phoneme to phonological lexicon inhibition	−0.25	−0.05	Lexical route parameter
TLA excitation parameter	0.03	0.12	This parameter represents how strong phonology computed by the TLA network activates the phoneme output buffer
Letter to letter scantime	1	20	This parameter represents how long it takes for the graphemic buffer to process each letter
Global activation rate	0.05	0.3	This parameter is used to change the slope of the sigmoid function (activation build-up) from input into a node.
Frequency modifier (both lexicons)	0.05	0.3	These could potentially differ across lexicons, although we treated them as one parameter
Minimum naming criterion	0.15	0.7	This parameter represents the minimum level of activation a phoneme node needs to reach so that it can be included in the phonemes that are output by the model
Phonological lexicon to stress excitation	0.01	0.15	Lexical route parameter
Phonological lexicon to stress inhibition	−0.2	0	Lexical route parameter
Stress to stress inhibition	−0.2	0	Stress output buffer parameter
Minimum stress naming criterion	0.01	0.6	This parameter represents the minimum level of activation stress nodes need to reach at the stress output buffer needs so that a word can be output
Grapheme parsing letter threshold	0.05	0.3	The level of activation which a letter must be over before graphemic parsing begins
Dead node level	0	20	This parameter is used to signal further searching needs to be done with a grapheme that the model thinks should be used but only has a weak connection to any phoneme

methods, such as Parameter Space Partitioning (Pitt et al., 2006), can also be used to get a more qualitative view of the performance of a model and what it can predict, but such techniques do not necessarily give information about which aspects of the model are responsible for its behavior.

In the next section, we introduce SPA, which is a method that allows the quantification of the extent to which different parameters used by a model affect its behavior. SPA allows us to gain within-model insight into complexity rather than insights into how the overall complexity of different models can be measured and used for the purpose of model comparison.

Sloppy parameters

The SPA technique was initially proposed by Gutenkunst et al. (2007a). The basic idea is that it allows one to quantify the extent to which different parameters affect a model's behavior. The key question is which parameters have strong effects on behavior and which have only small effects. This question will be addressed by optimizing CDP on major published data sets of reading aloud. Parameters in the model can then be varied to examine the effect they have on model performance. This will allow us to examine which parameters affect model fit across different datasets and which have a restricted or no effect at all. This will give insights into how complex CDP really is and which parameters are important for the predictions it generates.

It is worthwhile noting that prior model optimization is essential for conducting SPA. In principle, the calculations used for SPA could be done at any point within the cost landscape (i.e., how much error the model produces when tested with any possible parameter values). However, the strength of SPA is that it provides insight into the model output around an optimum region performance-wise (i.e., where the parameters of the model produce the lowest amount of error). Thus, the model must first be optimized. In general, the model is not optimized once, but is optimized many times. This helps ensure that the optimum set of parameters can be found. It also gives a distribution of values for each parameter that can give insight into its behavior (see below).

Another way to think about this is that optimization attempts to find the best parameter values of a model to reproduce experimental data. This process effectively places the model in an ideal situation, where its output coincides most strongly with that seen in an experiment. Thus, SPA at this point provides insight into how changes in model parameters induce deviation in the model output away from that which best fits the actual experiment (i.e., when the model cost is at a minimum). Applying SPA on an arbitrary (non-optimized) set of model parameters means that the model would be situated in a random region where the output is likely to be far away from that measured in an experiment. Therefore,

minor deviations in model parameters around such a random location would be likely to give little information about which parameters are important.

In terms of what we might expect to find from SPA, generally, even with highly complex models, only a few parameters tend to be stiff (i.e., affect model performance a lot). That is, parameter sensitivity often follows an exponential hierarchy of sensitivity where a small number of parameters affect model performance orders of magnitude more than the other parameters. A thorough discussion of why this commonly occurs is beyond the scope of this work (but see Francis & Transtrum, 2019; Mannakee et al., 2016; Transtrum et al., 2011).

The fact that the sloppiness of parameters in models often follows a hierarchical distribution is important because it suggests that there is no clear cut-off between sloppy and stiff parameters. Thus, researchers must rely on heuristic choices that best match the objectives of their experiments when using information from the parameter analysis to run studies (e.g., Gutenkunst et al., 2007a; Apgar et al., 2010). Alternatively, a counter-intuitive feature of many systems examined using this technique is that despite uncertainty in the specific value of many model parameters, the models are still capable of making accurate and precise predictions, thus providing good fits to experimental data (Transtrum et al., 2011). In this case, where a small number of stiff parameters account for most of the performance of a model, the number of factors that might be meaningfully investigated is vastly reduced.

As an alternative to a sloppy distribution of parameters, one might potentially find that different combinations of parameters in a model provide similar or identical predictions. That is, cases where the parameter values cannot not be predicted from the output of the model. When this happens, the model is said to suffer from structural unidentifiability (e.g., Villaverde et al., 2016). This means, in practical terms, that it is inherently difficult to constrain the specific values of model parameters with experimental data due to the compensatory nature of different parameter combinations. Such an insight would suggest that the causes of behavior in model performance would be difficult to predict. In this case, there would be informational redundancy in the model – that is, the model could predict the same behavior in multiple different ways – hence the same behavior could come from different underlying mechanisms. As an analogy, if someone was asked to think of two positive numbers (the parameters) that add up to 4, either $3 + 1$ or $2 + 2$ are equally possible and hence the problem is structurally unidentifiable because the parameter values cannot be determined from the answer.

Finding systems where there is structural unidentifiability has, interestingly, been relatively rare. Notably, in their seminal work, Gutenkunst et al. (2007a) investigated complex growth-factor-signalling networks in 17 different models that had between 10 and more than 200 parameters. They found that all had similar sensitivity distributions, where a small number of parameters accounted for most of the models' behavior, suggesting a universality across these systems. Since then, the breadth of applications has broadened significantly (e.g., Hartoyo et al. 2019; Machta et al., 2013; Panas et al., 2015; Ponce-Alvarez et al., 2020; Russell-Buckland et al., 2019; Transtrum et al., 2015, for examples in other domains) and a similar pattern where small numbers of parameters affect model performance has been found across many domains. Due to such data, Quinn (2019) has argued that sloppiness is perhaps a fundamental property of the parameters of high dimensional models and may even be responsible for why many complicated systems can be described by a small combination of important parameters.

Combining the results from optimization and SPA can give useful insights into the parameters and why some may be unidentifiable. Notably, a parameter may be unidentifiable but affect the model or it may simply have no effect on the model. This can be examined by comparing the distribution of parameters from the optimization together with results from the SPA. If the results from optimization show that values for a parameter are randomly distributed but the results from the SPA suggest that the parameter affects model performance, then this suggest the parameter is unidentifiable because it trades off with one or more other parameters. Alternatively, if SPA shows the parameter does not affect model performance, then it suggest that the optimisation causes a random distribution because the parameter simply has no effect on the model.

Given the ubiquity of sloppiness across such a broad range of phenomena, it is not unreasonable to assume that sloppiness may also be present within complex cognitive models. Indeed, if it were not present, one might wonder what would make the results of cognitive models so different from results in other areas of science. If sloppiness is present, it would show that even though CDP has many parameters, only a small number meaningfully change its behavior and thus its complexity is far less than the number of parameters might suggest.

SPA steps

The general method from which the sloppiness of parameters can be quantified requires a number of steps in setting up a model. As noted previously, it first requires parameter optimization. This is different from published versions of CDP, which all had parameter sets that were chosen by hand. Thus, we will first use a parameter optimization algorithm to find the best parameters to explain the largest amount of variance in seven published databases of 'reading aloud' (the term '*naming*' is also often used) before the SPA is done. Six of these databases contain words and one contains nonwords.

Once optimization has been done, it is possible to examine how systematic changes in parameters affect model performance. By examining how parameters and combinations of them change the performance of the model as they are moved away from the optimum, it is possible to get an idea of how much one parameter (or combinations of them) affects model performance compared to another. For example, if a small change in a parameter is made and the model is still close to its optimum, the parameter has little effect on performance. In contrast, if a small change in another parameter is made and it causes model performance to move away from its optimum, it suggests that the parameter has a big effect on performance. Therefore, the two parameters would have different levels of sensitivity. When this approach is extended to many parameters simultaneously, it can provide very useful information.

As an analogy, one can imagine sitting at the bottom of a crater where the physical shape of the crater is the state-space, with the amount of error being how far up from the bottom a point is. On a map, this could be represented by simple X and Y axes pointing north/south and east/west. The axes can be thought of as the parameters of the model. Such a map would be the equivalent to a 2-parameter topographic model where the change in the X and Y axis from the bottom of the crater could predict how far up the crater one had moved. One may then be interested in the fastest way up and hence out of the crater and the extent to which this corresponds to the X or Y axis. If moving along the X axis caused a fast ascent up the crater but moving along the Y axes did not, it would suggest that the X axis is stiff and the Y axis is sloppy. Thus, if we needed to predict the steepest route out of the crater, all we would need to do is move along the X axis. Hence the number of parameters that meaningfully affect performance is only one, despite the model having two parameters.

Optimization

Historically, the parameters in computational models were not typically set via optimization algorithms, even after they began to be used to predict quantitative data. This suggests that it is likely these models would be, if anything, under fit, because if a model has many parameters and their sensitivity is unknown, it is typically not possible to perform optimization well by hand. At least for psychological models, optimization should not be thought of as a simple step and then forgotten because examining the optimized versions of models (in our case CDP) is interesting in its own right and also points to potential data

interpretation problems.

Optimization of complex models may also be conceptually different compared to what one might believe from a naïve perspective, where optimization would be expected to find a single set of parameter values that minimize a cost function and the values would be the same if optimization was run independently twice. With complex models, this is not true because the optimization algorithms used are often stochastic, so by definition different parameter sets will be found across separate optimization runs. It also means that if a parameter has no effect on the model, it can take any value because it can never be optimized. As discussed above, parameters may also be unidentifiable. Practically, if the effect of a parameter is very weak (i.e., sloppy), optimization may also not be able to be run long enough to find the most optimum value due to computational limitations. A similar problem exists if there are many points in the state-space that produce values close to the minimum but use quite different parameters (i.e., similar minima but in different topographic locations). In this case, finding the minimum may be difficult and thus parameter values may be optimized towards different minima.

As discussed in the introduction, one common problem with optimisation is that it may cause over-fitting depending on the extent to which the model may fit different patterns and the richness of the data set. A model is overfit when it fits the data used for parameter optimization well but fails to reliably predict additional (i.e., held-out) data. If this is the case, SPA may potentially have poor validity because it may be biased by the particular parameters which caused the over-fitting and thus the sampling of the state-space may be misleading.

The type of data the models are tested on is important for considering how likely a model would be to be overfit. In particular, if overfitting cannot be ruled out easily it is at least less likely to occur with models that are designed to simulate many effects in many different data sets, all else being equal. This is because it is easier and less constraining to fit a model to one data point (i.e., a single empirical phenomenon) compared to many data points (e.g., Adelman et al., 2014). In this case, the parameter space usable to explain just one effect can only be larger than the parameter space that a model can use when it needs to explain many effects. For example, with models designed to explain many effects, it may be possible to show a good fit on one effect but be completely off on another, hence reducing the possible parameter space compared to a model that does not need to capture more than one effect. This is a clear advantage of using heterogeneous data for model optimisation and testing, and not just a single effect.

To get around the problem of testing computational models and their ability to generalize, one common practice has been to examine the performance of a model using a single set of parameters across multiple databases and comparing that to simple regression with multiple predictors thought to be important in the task as a quantitative baseline, as discussed in the introduction. Perhaps surprisingly, the same procedure where a model with one set of parameters is used to predict results from multiple datasets has not typically been used with regression modelling in this area. Below we will do this as it can potentially help distinguish between a drop in model performance due to overfitting and a drop in model performance due to actual differences in data sets. In this case, if model performance dropped considerably compared to a regression equation with relatively few terms, it may suggest overfitting had occurred.

Optimization specifics

To find the optimum parameter values for CDP on any given dataset, we used the particle swarm algorithm of Kennedy and Eberhart (1995). The code was adapted from the Standard POS 2006 version (Clerc, 2012). The default parameters were used. There is no inherent reason why this algorithm is preferred over others. However, it is well known to be a powerful search algorithm and can search across non-continuous state-spaces as well state-spaces of high dimensionality (Esmin, Coelho, & Matwin, 2015). Practically, it was very efficient in optimizing the parameter values of CDP and thus made the problem tractable. For each data set examined, all of which contained the mean reaction times (RTs) of individual items, optimization was done on 27 different parameters (see Table 1). Twenty-two particles were used and searching of the parameter space was allowed to continue for 26 cycles. This meant that there were 572 runs of different parameter sets each time a model was optimized on a dataset before searching was terminated. The search termination criterion was empirically defined based on initial optimization test-runs showing an asymptote in the cost function minimization. Moreover, the entire procedure was repeated 50 times for each data set and the model with the lowest error rate from the 50 models was chosen for further analyses to increase the chance of finding a model as close as possible to the optimum that CDP could achieve.

The model parameter values were found using two different cost functions. As is typical in the reading domain, one was a simple correlation between results produced by the model and human reading aloud latencies (i.e., RT data). We also added a small constant for each error the model makes (see below). The typical argument for using correlations is that these models do not describe processes that pertain to speech production (e.g., Levelt et al., 1999). This means that model latency measurement units (i.e., 'cycles') cannot be expected to predict the absolute RT values. We also used a more typical mean Sum-Squared-Error (MSE) cost function. To get the actual RTs within a reasonable range for the MSE optimized models, the actual RTs were simply divided by 6.5, which brings them into the range of values (i.e., cycles) that the model typically needs to produce a word. Using two different error functions is of interest when looking at the SPA because it allows us to examine the extent to which different and somewhat arbitrarily chosen cost functions can affect the behavior of CDP. Note that we could have added a third type of model where we used CDP scores optimized with a MSE cost function plus an overall constant. In this case, the constant would represent processes beyond the scope of the model (as typically done with regression when examining RTs). However, given that CDP has a parameter affecting overall speed (global activation), for the sake of simplicity we only used the simplest model without a constant.

Since the parameters of CDP cannot take any value, boundaries were added so the search space was finite. These were chosen by hand based on previous simulations from different sets of parameters that we have used and from absolute boundaries where the parameter could not move further (e.g., a parameter for an excitatory connection in CDP cannot take a value less than 0). The boundaries were then set much further away than these parameters so the models could potentially find solutions away from those parameters. As can be seen from simulations below, the main parameters that affected model performance did not cluster around the boundaries. This suggests that the range was not so tightly specified that having even wider boundaries would have been likely to change the results meaningfully.

To clean the data, we used a procedure that is standardly used on human behavioral data: words that produced RTs that were 3SDs above or below the mean value were removed. We also excluded from optimization and analyses the words that were not in the model's lexicon as well as words that were incorrectly named by the models.

Initial analyses showed that optimization occasionally led to high error rates on some data sets. This occurred because CDP would sometimes optimize itself on a small number of words that would then produce a very high correlation. This occurred because words that CDP produced an error on were removed from the analyses, thus sometimes leaving very small sets of words that had a different distribution compared to the whole set. Because of this, we added a small error penalty (.01 for the models optimized on correlations and 50 for models optimized on MSE scores) for each error CDP made when reading a word based on either the phonemes it produced or whether it produced the correct stress. The stress penalty was only applied when disyllabic words were used, since it is not possible to articulate monosyllables in isolation without stress. This had the effect of driving model errors to a low level

and the parameter values from the simulations with the lowest error score also tended to have the smallest number of errors or was very close. The equation used to evaluate the models using the correlation cost function was:

$$C_{\theta_i} = (1 - r_{\theta_i} + 0.01 \times NE_{\theta_i} + 0.01 \times NSE_{\theta_i})^2 \quad (1)$$

where C = is the overall cost (score), r = is the correlation, NE = is the number of errors, NSE = number of stress errors, and θ_i represents the model parameters $\theta_{i\ldots N}$, where N in our case was 27 (i.e., all of the parameters investigated).

The equation used to evaluate the models using the MSE optimization was:

$$C_{\theta_i} = \frac{1}{\text{Correct}_{\theta_i}} \sum_{1..j} \left(o_{\theta,j}/6.5 - e_{\theta,j}\right)^2 + 50 \times \text{NE}_{\theta_i} + 50 \times \text{NSE}_{\theta_i} \quad (1.1)$$

where C = is the overall cost (score), Correct is the number of words correctly produced by the model, o is the observed RT, e is the value the model produced, NE = is the number of errors, NSE = number of stress errors, θ_i represents the model parameters $\theta_{i\ldots N}$, where N in our case was 27 (i.e., all of the parameters investigated), and j represents the number of words the model was optimized on (note that words the model produced an error on were not used in the calculation).

The models were independently optimized on the individual items (word RTs) from each of seven databases. Four contained only monosyllables (Spieler & Balota, 1997, Balota & Spieler, 1998; Treiman et al., 1995; Seidenberg & Waters, 1989 (called the Waters database below to avoid confusion with the Seidenberg Nonwords)), two contained disyllabic words (Chateau & Jared, 2003; Yap & Balota, 2009), and one contained only nonwords (Seidenberg et al., 1994).

Finally, one minor difficulty in the area of reading aloud (and an issue that arises here) is determining what the 'correct' responses are for nonwords. This is for a number of reasons. These include people occasionally giving answers that appear to be caused by the misperception of letter order (e.g., saying 'plit' for 'pilt'), nonwords producing a distribution of answers but the models producing only one, the dialect of the language that the model uses differing (e.g., Received Pronunciation vs. Hoosier English) meaning the model can predict the wrong pattern due to being given incorrect statistical information, and whether certain sequences of letters forming nonwords can even meaningfully test a model (e.g., Perry, 2018). Thus, there is no perfect way of determining what should be an error in a model such as CDP when reading nonwords. Given this, for the sake of simplicity, we used a hand-coded list of 'correct' answers for nonwords (see Pritchard et al, 2012, for further discussion). The results produced are likely to be very similar to other possible lists of 'correct' answers. This is because all of the methods produce similar 'correct' lists on nonwords not deliberately designed to test these differences, and a small number of words being correct based on one method or incorrect based on another is very unlikely to meaningfully affect parameter optimization given the way the error scores are calculated.

Results and discussion

Optimization performance

The results of the 14 models (7 datasets × 2 cost functions) and hence 14 parameter sets that were created via optimization are presented in Table 2.

As can be seen, the correlations for the 50 runs of each dataset with the actual data where the model used a correlation cost function were surprisingly similar, although more variable with the MSE cost function. In addition, the models with the strongest correlations in the data with the MSE cost function models had *r* values that were only slightly less than the models using the correlation cost function. Alternatively, the models with the lowest *r* values were much lower with the MSE cost function models. This suggests that it was harder to fit models using the MSE cost function. This is unsurprising given searching for correlations is likely to be easier because exact values do not need to be found and also because our model testing used *r* scores. These results suggest that care needs to be taken interpreting the distributions of the parameters from large numbers of model optimization runs, because some of the variability with the MSE models may be due to difficulty in optimizing the parameters.

Quite unlike the correlations found testing the models, the mean overall RTs of the best fitting models varied a great deal with the models using a correlation cost function. Alternatively, the overall RTs were far more homogenous for the models optimized using the MSE cost function compared to the models optimized using the correlation cost function. This suggests that different optimization runs using the correlation cost function are likely to result in different minima while different runs with the MSE cost function are more likely to reflect optimization towards the same minimum.".

Next, we examined the generalization performance of the model with the lowest error score in each dataset by testing it on the other datasets. That is, for each dataset, the model with the lowest error score

Table 2
Results of the optimization runs on the different data sets.

Dataset	N. Words	Cost Function Score				Correlation (r)				Mean RTs			
		Avg	SD	Min	Max	Avg	SD	Min	Max	Avg	SD	Min	Max
Correlation Cost Function Models													
Seidenberg NWs	588	0.65	0.015	0.60	0.67	0.43	0.009	0.41	0.45	117.2	19.8	86.8	167.9
Chateau	901	0.27	0.006	0.26	0.28	0.55	0.007	0.53	0.57	78.7	23.6	38.7	129.9
Spieler Young	2998	0.35	0.008	0.33	0.37	0.47	0.006	0.45	0.48	85.7	28.0	38.5	143.2
Spieler Old	2998	0.31	0.008	0.30	0.34	0.5	0.006	0.48	0.51	80.7	26.4	36.3	165.5
Treiman	1327	0.39	0.009	0.36	0.4	0.43	0.01	0.42	0.46	78.0	14.5	40.6	107.1
Waters	1329	0.42	0.011	0.39	0.44	0.37	0.009	0.35	0.39	70.5	19.4	30.3	118.2
Yap	6714	0.39	0.007	0.37	0.4	0.62	0.006	0.6	0.63	68.8	23.4	29	133.6
MSE Cost Function Models													
Seidenberg NWs	588	1254.25	9.25	1244.9	1290.9	0.39	0.02	0.33	0.42	106.1	1.37	102.6	108.5
Chateau	901	353.31	30.02	306.8	437.8	0.51	0.04	0.41	0.56	98.3	1.96	93.09	103.2
Spieler Young	2998	293.66	23.77	261.8	334.8	0.41	0.04	0.28	0.46	69.3	1.62	64.5	73.8
Spieler Old	2998	323.47	28.85	247.8	377.5	0.45	0.04	0.35	0.49	100.2	2.12	95.91	104.7
Treiman	1327	400.15	10.17	386.8	445.5	0.38	0.04	0.26	0.43	94.0	1.25	89.78	95.7
Waters	1329	98.23	7.77	88	130.3	0.3	0.03	0.23	0.35	85.7	0.96	82.53	87.9
Yap	6714	782.71	25.61	752.2	875.7	0.58	0.04	0.46	0.62	96.0	2.39	89.55	100.7

Note: Correlation = Pearson correlation coefficient between the predictions of CDP on all word/nonwords and the actual data. Mean RTs = mean number of cycles the model was run for on each word in the data set. Seidenberg NWs = Seideberg et al. (1994), Chateau = Chateau & Jared (2003), Spieler Young/Spieler Old = Spieler and Balota (1997/1998), Treiman = Treiman et al. (1995), Waters = Seidenberg & Waters (1989), Yap = Yap & Balota (2009).

was first chosen (i.e., seven parameter sets for the models using the correlation cost function and seven parameter sets for the models using the MSE cost function). Each of these models was then used to predict the results of all seven datasets, hence leading to 49 correlations (i.e., 7 × 7) in each set. This included the correlations where the optimized parameters were tested on the items from the experiments they were optimized on.

One difficulty there is when selecting models is that whilst choosing the optimum models with the correlation cost function is obvious, it is less obvious which models should be chosen when using the MSE cost function. This is because the models are evaluated with correlations, not MSE scores (i.e., it is typical to report how strongly the models correlate with actual data rather than report MSE scores). One might therefore consider the optimum model to be the one with the strongest correlation, even if it had a MSE higher than another model. Alternatively, given the cost function uses a MSE function, the model with the lowest MSE might be considered the most model with the lowest error score.

Given these two possibilities, with the MSE cost function models, we first examined performance of the models based on both the lowest MSE scores and the strongest correlations. The results from the models using the strongest correlations in each experiment made slightly better predictions than the models using the lowest MSE scores in terms of correlations, and so these were used since the optimum model needs to be used with SPA. Interestingly, the performance of the optimized models on the data sets they were not optimized on (i.e., generalization performance) also tended to be better with the models chosen based on the highest correlation rather than the lowest MSE score. The results from the models selected based on having the strongest correlation are reported in Table 3, the results from the models selected based on having the lowest MSE scores are reported in Table S1 of the Supplementary material, and SDs on each database are reported in Table S2. As can be seen, differences between both types of model are small. When referring to the models that use a MSE cost function below, we will always refer to those where the optimal ones were selected based on having the strongest correlation with the data. Note that all calculations (i.e., calculations for the SPA below) apart from examining model RTs and their fits with actual data are still done with MSE scores with the models using a MSE cost function. An alternative method of final model selection would have been to test all possible models and choose the ones with the best generalization performance, thus avoiding a possible local minimum from models that potentially over-fit some aspects of the data and more poorly fitting models.

We also used simple regression to predict the RTs of the word only data sets with log word frequency, letter length, orthographic neighbors, and spelling-sound consistency as predictors (see Spieler & Balota, 1997; measurements were taken from the English Lexicon Project [Balota et al., 2007] except for spelling-sound consistency which was taken from Perry et al. [2010b]). The HAL database was used for word frequency. The regression analyses were done because it has been argued that a computational model should be better or at least as good as the amount of variance accounted for by a linear regression using these four predictors (Spieler & Balota, 1997). However, historically, most models stayed well below this critical benchmark. Even the latest version of CDP (Perry et al., 2013), which performed better than all previous models, stayed just below this critical benchmark.

As can be seen, correlations between model behavior and the different experiments were generally quite high. The optimized models also correlated more strongly than the regression equations (words) apart from on one data set (Spieler Young) where the results were of a similar strength. The results are also slightly higher than the best hand-picked parameter sets we have reported elsewhere (e.g., Perry et al., 2013). In particular, using exactly the same model as here, Perry et al. (2013) report *r* values of .43, .49, .42, and .29 for the Spieler and Balota (1997/1998), Treiman et al. (1995), and Seidenberg and Waters (1989), databases, respectively.

Rather than use the optimum models as a comparison, it is also possible to examine the generalization performance of each parameter set, which, across those 4 data sets was .41 in Perry et al. (2013). With the models here, the results ranged from .40 to .42 (excluding the models optimized on the Seidenberg nonwords). This suggests the results are quite comparable, despite no hand tuning of any parameters.

In terms of the Chateau and Jared (2003) and Yap and Balota (2009) databases, these are not so easy to compare because Perry et al. (2013) only reported regression results where 8 factors were used to predict RTs. They then compared the regression results to those from CDP plus an additional factor. The results they found showed that CDP picked up a similar amount of variance to the regression equation. Whilst we did not use the same factors to compare CDP as Perry et al. (2013) did, as can be seen from Table S3, the optimized CDP models do better than a regression that included 4 factors.

In terms of more specific details, the mean correlation from each data set the models were optimized on was .49/.48 (Correlation/MSE cost function) compared to .37/.41 (Correlation/MSE cost function) for sets they were not optimized on. The model optimized on the Seidenberg nonwords had the worst generalization performance. The Seidenberg nonwords were also comparatively difficult to predict for models optimized on the other databases. This suggests that, perhaps unsurprisingly, words are better at predicting words than nonwords.

The error rates of the different models are also informative (see Table 4). The only comparison where high error rates occurred was when the model using the correlation cost function was optimized on the Chateau and Jared words and then used to predict the Seidenberg nonwords. This suggests that models only optimized on words may but will not necessarily have trouble generalizing to nonword pronunciations.

It is possible to evaluate CDP further with respect to the regression equations in terms of how well those equations generalize to data sets they were not fit to. To do this, we ran regressions on all of the data sets with only words in them. In a similar way to what we did with CDP, we first took the parameters from the regression equations that were derived from each dataset they were calculated from. We then examined how well those parameters predicted the other data sets with only words in them (see Table S3). Overall, CDP showed a higher mean correlation than the regression equations across all of the comparisons the models were not optimized on (.42/.43 [Correlation cost/MSE cost] vs. .36 [regression]). This suggests that CDP is not only able to better fit the data when it is optimized on a particular data set, but that it also generalizes more accurately to novel data sets. Thus, despite CDP having a large number of parameters, it does not cause overfitting when compared to a regression equation with only a small number of parameters.

Distribution of parameters

The distribution of parameters found in the 50 runs of each experiment (plus another 50 runs of each experiment that were added for this graph but were otherwise not used in this study) can be seen in Fig. 2, which has the SDs of the data sets from each parameter of each experiment (the same data displayed as histograms can be found in Fig. S2). Small SDs mean similar parameter values were found whereas large SDs mean the parameters differed significantly.

The data for each parameter was initially normalized so the SDs were comparable. This was done by dividing each value of each parameter by the difference between the low and high boundaries found in Table 1 of that parameter. As can be seen from Fig. S2, most of the results show an essentially random distribution. With the normalization we used, this meant SDs of around .25. Therefore, the gradient in the colour change was stopped at .25 (which is black in the figure). For the sake of simplicity, we did not use more complicated measures for the spread of parameters. We also performed parameter recovery simulations on the output of the top model from each dataset. These were used to examine the extent to which similar models give similar results, in the easiest of

Table 3
Pearson correlation coefficients (*r*) calculated between (a) the results of CDP optimized on items from different databases and the items of different databases; and (b) *r* values from using a regression equation with four different factors (log Hal frequency, log word frequency, consistency, and orthographic neighbourhood) on the same databases.

Dataset model optimized on/ Regression	Overall Mean	Seidenberg NWs	Chateau	Spieler Young	Spieler Old	Treiman	Waters	Yap
Correlation Cost Function Models								
Seidenberg NWs	.31	**.44**	.39	.24	.30	.22	.10	.46
Chateau	.41	.19	**.57**	.37	.45	.43	.29	.59
Spieler Young	.42	.29	.39	**.48**	.51	.39	.32	.56
Spieler Old	.42	.29	.40	.46	**.50**	.40	.31	.57
Treiman	.37	.13	.26	.45	.51	**.46**	.35	.40
Waters	.35	.24	.19	.42	.43	.38	**.38**	.43
Yap	.45	.30	.54	.44	.49	.41	.32	**.63**
Max	.45	.44	.57	.48	.51	.46	.38	.63
Min	.31	.13	.19	.24	.30	.22	.10	.40
Median	.41	.29	.39	.44	.49	.40	.32	.56
Mean	.39	.27	.44	.41	.46	.38	.30	.52
MSE Cost Function Models								
Seidenberg NWs	.41	**.42**	.38	.44	.47	.38	.25	.56
Chateau	.44	.27	**.56**	.45	.48	.41	.33	.61
Spieler Young	.39	.27	.36	**.46**	.44	.36	.32	.53
Spieler Old	.44	.25	.55	.45	**.49**	.41	.32	.62
Treiman	.43	.19	.51	.45	.50	**.43**	.30	.62
Waters	.38	.30	.25	.44	.43	.36	**.35**	.51
Yap	.43	.29	.52	.43	.49	.40	.29	**.62**
Max	.44	.42	.56	.46	.50	.43	.35	.62
Min	.38	.19	.25	.43	.43	.36	.25	.51
Median	.43	.27	.51	.45	.48	.40	.32	.61
Mean	.42	.28	.45	.45	.47	.39	.31	.58
Regression (r)		.42	.50	.49	.36	.41	.32	.48
N. Words		588	901	2998	2998	1327	1329	6714

Note: Seidenberg NWs = Seidenberg et al. (1994), Chateau = Chateau & Jared (2003), Spieler Young/Old = Spieler and Balota (1997/1998), Treiman = Treiman et al. (1995), Waters = Seidenberg & Waters (1989), Yap = Yap & Balota (2009), Regression = *r* values from a four factor model for words (log frequency, neighborhood, letter length, consistency) or a single factor model for nonwords (letter length).

Table 4
Error rates (%) CDP produced when optimized on the items from different databases when predicting word pronunciations from different databases (i.e., the number of times the model produced the phonology of a word/nonword that people would be very unlikely to produce).

Dataset model optimized on	Overall Mean	Seidenberg NWs	Chateau	Spieler Young	Spieler Old	Treiman	Waters	Yap
Correlation Cost Function Models								
Seidenberg NWs	1.35	**3.74**	1.44	0.70	0.70	1.36	0.90	0.60
Chateau	3.26	19.39	**0.55**	0.53	0.53	0.98	0.53	0.31
Spieler Young	1.59	6.29	2.89	**0.17**	0.17	0.53	0.08	0.98
Spieler Old	1.23	6.46	0.89	0.17	**0.17**	0.45	0.08	0.37
Treiman	1.42	6.63	1.66	0.27	0.27	**0.45**	0.15	0.51
Waters	1.19	4.76	1.89	0.27	0.27	0.53	**0.08**	0.55
Yap	1.62	9.52	0.55	0.20	0.20	0.53	0.08	**0.22**
Max	3.26	19.39	2.89	0.70	0.70	1.36	0.90	0.98
Min	1.19	3.74	0.55	0.17	0.17	0.45	0.08	0.22
Median	1.42	6.46	1.44	0.27	0.27	0.53	0.08	0.51
Mean	1.66	8.11	1.41	0.33	0.33	0.69	0.27	0.51
MSE Cost Function Models								
Seidenberg NWs	2.15	**3.91**	2.77	1.27	1.27	2.19	1.13	2.52
Chateau	1.42	7.82	**0.67**	0.30	0.30	0.45	0.15	0.27
Spieler Young	1.09	5.61	0.67	**0.20**	0.20	0.53	0.08	0.37
Spieler Old	1.38	6.80	1.66	0.17	**0.17**	0.38	0.08	0.42
Treiman	1.81	10.20	1.00	0.30	0.30	**0.45**	0.15	0.25
Waters	1.51	6.80	2.22	0.23	0.23	0.38	**0.08**	0.63
Yap	0.98	5.10	0.67	0.20	0.20	0.45	0.08	**0.19**
Max	2.15	10.20	2.77	1.27	1.27	2.19	1.13	2.52
Min	0.98	3.91	0.67	0.17	0.17	0.38	0.08	0.19
Median	1.42	6.80	1.00	0.23	0.23	0.45	0.08	0.37
Mean	1.48	6.61	1.38	0.38	0.38	0.69	0.25	0.66
N. Words		588	901	2998	2998	1327	1329	6714

Note: Seidenberg NWs = Seidenberg et al. (1994), Chateau = Chateau & Jared (2003), Spieler Young/Old = Spieler and Balota (1997/1998), Treiman = Treiman et al. (1995), Waters = Seidenberg & Waters (1989), Yap = Yap & Balota (2009).

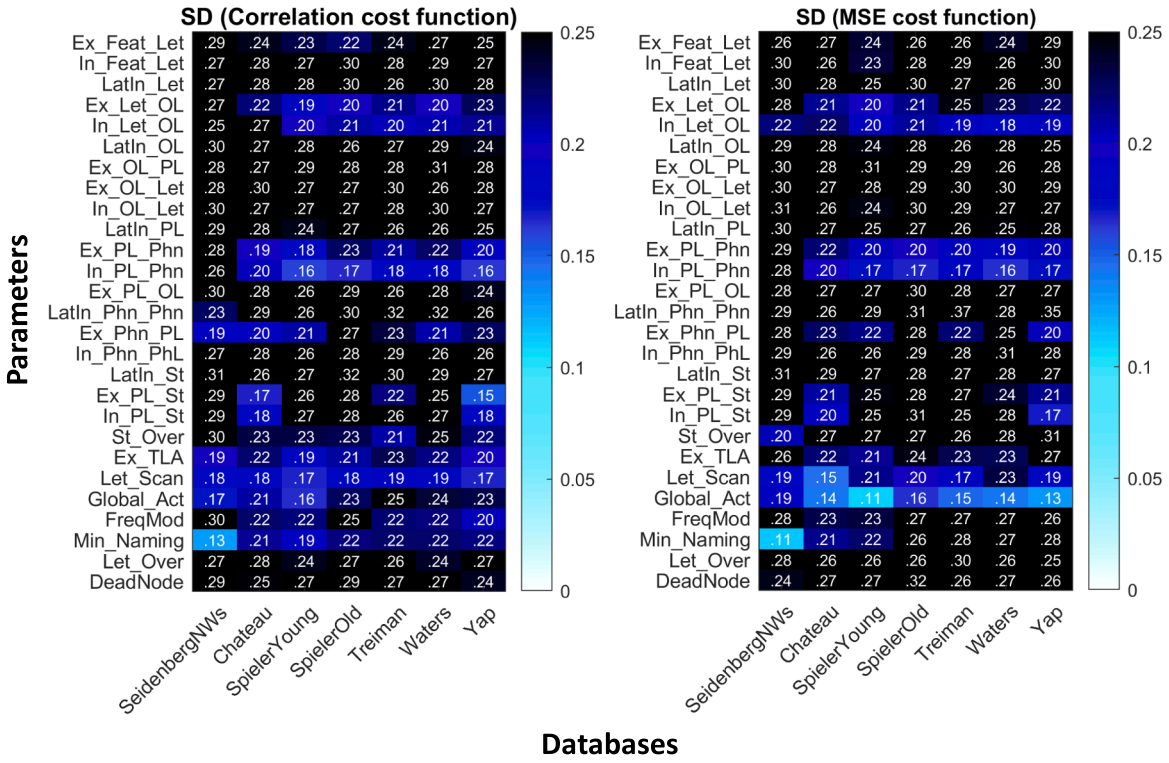

Fig. 2. Normalized standard deviations for each parameter from 100 optimization runs. The left panel displays data from models using a correlation cost function and the right panel displays data from models using a MSE cost function. Note: Normalization was done by dividing parameter values by the difference between the low and high values for each parameter taken from Table 1. SeidenbergNWs = Seidenberg et al. (1994), Chateau = Chateau & Jared (2003), Spieler Young/Old = Spieler & Balota (1997/1998), Waters = Seidenberg & Waters (1989), Yap = Yap & Balota (2009), Ex = Excitatory, In = Inhibitory, LatIn = Lateral inhbition, Feat = Feature, Let = Letter, OL = Orthographic lexicon, PL = Phonological lexicon, Phn = Phoneme, TLA = Two-later associative network, Let_Scan = Letter scanning time, FreqMod = Frequency modifier, Min_Naming = Minimum naming criterion, St = Stress, St_Over = Minimum stress naming criterion, Let_Over = Grapheme parsing letter threshold, DeadNode = Dead node level.

possible conditions for this to happen (see e.g., Wilson & Collins, 2019). If similar results were not found, it would suggest that the model was poor. As can be seen in Fig. S3 in the Supplementary material, the values were similar to those that were initially found, including the distributions.

The results show there is in fact a very broad parameter space that allows the models to perform reasonably well. This is because most parameters could take a wide range of values, as can be seen in Fig. S3 where histograms of the values are plotted. Those histograms show that values from both the correlation cost function and MSE cost function models are both negatively skewed, with greater skew occurring with the MSE cost function models. Since the right part of the distribution represents parameters with relatively random values, this shows most parameters take relatively random values. This should not be seen as surprising, because if most parameters have little effect on the performance of the model (as we will confirm later), then they should also be able to take a wide range of values without making much difference to model performance. Thus, most parameters will not be optimized meaningfully because there is simply no way to optimize them that would increase the performance of the model meaningfully. They are therefore either structurally unidentifiable or simply have little effect on the model. The model thus behaves like most models of real-world systems in this respect (e.g., Gutenkunst et al., 2007a).

There were some differences caused by the type of database. As can be seen, with the Seidenberg nonwords, the parameter range was broad with all orthographic lexical parameters, apart from the letter to orthographic lexicon inhibition. This was unlike the words. This should not be seen as surprising, because words are stored in the lexicon of CDP but nonwords are not, and so parameters relating to lexical (word) retrieval should have less effect with nonwords. The main exception to this is the letter to orthography inhibition parameter. This parameter stops spurious forms (i.e., words with similar but not the same letters as the word/nonword that is presented) from being incorrectly activated.

There were also differences caused by the cost function. As can be seen, models using the MSE cost function had tighter SDs than models using the correlation cost function with some parameters not directly related to the lexical route. Notably, with the models using the MSE cost function, the global activation parameter (slope of the activation function) had a much tighter distribution. There were also some parameters that were essentially randomly distributed in the MSE models that were weakly constrained in the correlation cost function models. This is interesting because it shows that the extent to which different parameters generate relatively weakly constrained distributions is less when the models are optimized on MSEs. This should not be seen as surprising given that correlations are not constrained in the same way as predicting the actual data. Thus, correlations allow more parameters to have weak contributions to the final scores.

A final important difference was that the Seidenberg nonwords used a reasonably specific minimum naming criterion with the MSE cost function models (i.e., how much activation at the phonemic buffer is needed before a word can be read aloud), unlike the word databases. This is likely to be because nonwords create less activation than words in the model because they are not supported by lexical information. Thus, it is harder for them to generate enough activation to reach the minimum naming criterion at the phoneme level. These results suggest that differences between the naming of words and nonwords can be compensated for by using a lower naming criterion before naming occurs with nonwords. Thus, with the MSE cost function model, CDP predicts people read words and nonwords aloud based on different amounts of information being available when the words are presented in

pure lists (i.e., only words or only nonwords) or mixed lists.

Sloppy parameter analysis

Once a model is optimized, determining whether it has sloppy characteristics is relatively straightforward. This can be achieved by constructing an appropriate cost function and performing a detailed analysis on how model performance changes when each parameter is varied. The cost function measures how closely the theoretical model fits the experimental data. The most commonly used function is a residual sum of squares, which is not exactly what we used (see equations (1) and (1.1) for more details).

$$C_\theta = \sum_{i=1}^{N} (y_i - f(\theta_i))^2 \tag{2}$$

where C_θ is the cost, y_i is the data to be modelled, and $f(\theta_i)$ is the system characterised by model parameters, $\theta_{i\ldots N}$. Minimal values in the cost function are representative of better fits to experimental data. Derivatives provide a natural tool to investigate how the model output changes with alterations in the model parameters. Specifically, we use the Hessian matrix, which is a square matrix that contains the second order partial derivatives of the cost function with respect to the model parameters. For a cost function, C_θ, that depends on a set of parameters $\theta_{i\ldots N}$, the Hessian matrix is:

$$H_C = \frac{\partial^2 C_\theta}{\partial\theta_i \partial\theta_j} \tag{3}$$

The Hessian matrix is useful as it maps out the geometry of the cost landscape. Decomposing the Hessian matrix of the cost function into a set of eigenvalues and eigenvectors enables us to analyse the sloppy structure of the system and infer which parameters may be sloppy or stiff. The eigenvector and eigenvalue pairs (eigenpairs) of the Hessian can be determined by solving a set of linear equations (See the Method section for more details).

The relevant information that is central to this research is the geometric interpretation of the eigenpairs. The local curvature of the cost function, which we can imagine as peaks, troughs, and valleys in high-dimensional space, is illuminated through the Hessian matrix where the signs of the computed eigenvalues describe whether the local landscape slopes upwards or downwards. Within the context of cost functions, the eigenvalue with the greatest magnitude is associated with the direction (eigenvector) in which the cost changes most significantly (eigenvalue). There is also a simple interpretation of eigenvalues when compared with other values. Notably if we have an eigenvalue with a value of 100 and compare it to an eigenvalue with a value of 1, a step in the direction related to the eigenvector which is associated with the large eigenvalue would have an influence on the cost 100 times greater than the smaller eigenvalue. As an analogy, if one was at the bottom of a crater with only two ways up, of which one side was very steep and the other less so, one eigenvector would point in the direction of the steepest way up and another the less steep way up. These eigenvectors would not necessarily and indeed be unlikely to be perfectly aligned with the axes of a map where, for example, simple X and Y axes pointing north/south and east/west are often used. Such a map would be the equivalent of a 2-parameter topographic model where the change in the X and Y axis from the bottom could predict how far up the crater one had moved. The eigenvalue associated with the steepest up would also be larger than the less steep side. Thus, if two people ran up from the bottom of the crater with equal effort, one on the steep and the other on the less steep side, the difference between the two runs in terms of how far they would ascend would be in the same proportion as the difference between the eigenvalues.

Generally, the local landscape of the cost function in the proximity of the region of the best fit can be represented by structures that are ellipsoidal in nature (Gutenkunst et al., 2007a), with different directions of the high-dimensional ellipsoids pinched and skewed along stiff and sloppy directions. The principal axes of the ellipsoids are provided by the eigenvectors and the respective widths along each axis are proportional to one over the square root of the paired eigenvalue. Each principal axis is a mixture of the bare parameter axes of the model (i.e., the parameters that the model uses) and how long a principal axis is relates to how stiff or sloppy it is (how steeply the cost changes in that associated direction). If a given principal axis is narrow, then it is said to point in a stiff direction.

Fig. 3 provides an illustration of ellipsoidal cost contours that have a sloppy and stiff direction. We can imagine the inner most ellipse as a basin where we are surrounded by steep inclines. We can see that parameter combinations that move along the stiff direction have a far greater influence on the model output than moving along a sloppy direction i.e., the hills become much steeper for a given step in a stiff direction compared to a sloppy direction. In example 3a, both bare parameters axes are misaligned with the sloppy and stiff directions. Thus, taking a step in the direction of Parameter 1 has an equivalent response as a step in the direction of Parameter 2, which means that these example parameters have compensatory influences. Alternatively, in example 3b, the bare parameter axes are aligned with a stiff (Parameter 1) and a sloppy (Parameter 2) axis. Thus, taking a step in the direction of Parameter 1 would cause a much larger change in model performance than the same sized step in the direction of Parameter 2.

The example in Fig. 3 gets to the heart issue of structural identifiability: specific values of the parameters are not necessarily well constrained by fits to the data because multiple combinations of parameters may give rise to the same model outcome (i.e., when different parameter movements cause a similar change to model performance). The bare parameter axes also rarely align perfectly with the ellipsoidal principal axes generated by the eigenvectors and eigenvalues, as found in most of the work on sloppy parameters, as discussed above. The degree of alignment between a bare parameter and principal axis thus provides a measure of how stiff or sloppy it is. If a parameter contributes greatly to the dominant eigenvector (that which is associated with the eigenvalue with the greatest magnitude), we can think of it as being maximally 'stiff'. Categorising the contributions of the model parameters to the dominant eigenvectors of the Hessian matrix thus allows us to explore the sloppy nature of the model (see Methods for a detailed explanation).

Methods

In this section we describe the computational methods employed throughout the research. We begin with a description of the Hessian calculation using finite differences and then explore the various sloppy model measures (stiff/sloppy parameter combinations) and parameter ranking[2] tests that we use to elucidate potentially significant and identifiable model parameters.

Hessian computation - finite difference methods

The Hessian matrix, H, of the cost function (model fit to experimental data) is the foundation of SPA. The Hessian matrix provides all the key information of the cost landscape. The Hessian matrix describes the deviation in model behaviour with respect to variations in model parameters. The cost functions used within this research were not a traditional sum of squared residuals (χ^2). In one case, the function maximizes the correlation between the model output and actual data with a small penalty for errors on individual items. The correlation is subtracted from 1 and the errors are added to the results with the final term squared. When the correlation is high and the errors are low, this cost function approaches zero – see equation (1). In the second case (see

[2] This is typically called *ranking* despite the fact that the ranking of each parameter is derived as a numerical number without respect to the other parameters and the numerical strength of this number is important.

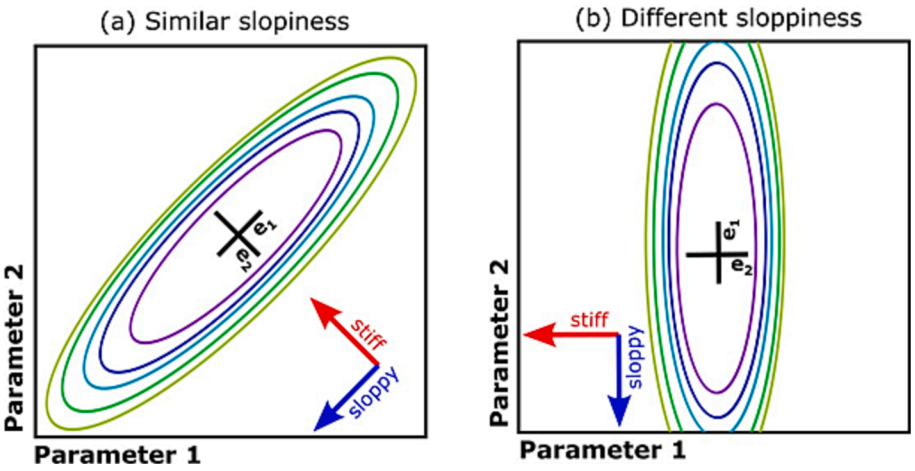

Fig. 3. Two-dimensional hessian ellipsoids which are aligned in two directions (e_1 and e_2). Cost landscapes occur around region of best fit where the contours form ellipsoids. Both examples have a sloppy and stiff principal axis where a step in the stiff direction results in a far greater change in the cost than a proportionate change in the sloppy direction. In example (a), the bare parameter axes (Parameter 1, Parameter 2) are misaligned with the stiff direction and hence have similar sloppiness, whereas in example (b) Parameter 1 is stiff and Parameter 2 is sloppy.

equation (1.1), a more traditional MSE was used, but a small penalty for errors on individual items was also added.

The Hessian matrix can be derived analytically for well-behaved models (i.e., sum of exponentials) and sufficiently simple cost functions. However, in most cases this is not possible and methods that obtain an approximate Hessian must be employed. For our model, an analytic form of the Hessian is not known, thus we employ finite difference methods to approximate the Hessian around the best fit. Finite differences methods are a set of numerical techniques that approximate derivatives using finite differences. A forward finite difference routine is used here and the approximate Hessian can be calculated as follows:

$$H_{\theta^*} = \frac{\partial^2 C}{\partial \theta^*_i \partial \theta^*_j} = \frac{C(\theta^* + \Delta\theta_i + \Delta\theta_j) - C(\theta^* + \Delta\theta_i) - C(\theta^* + \Delta\theta_j) + C(\theta^*)}{\Delta\theta_i \Delta\theta_j} \tag{4}$$

where the cost, C, is evaluated using the best fit model parameters, θ_i*, and from minute deviations for each parameter pair ($\Delta\theta_i$, $\Delta\theta_j$) about the region of best fit. Note that the * means the optimum (best fit) parameters.

Different parameters can take on values that range over many orders of magnitudes. To account for that we employed parameter scaling in the denominator of the Hessian matrix elements. This is necessary because the Hessian matrix is sensitive to the size of the values entered into it. As a point of comparison and consistency check, we used two different scalings; a logarithmic scaling

$$H_{\theta^*} = \frac{\partial^2 C}{\partial \log\theta^*_i \partial \log\theta^*_j} = \frac{C(\theta^* + \Delta\theta_i + \Delta\theta_j) - C(\theta^* + \Delta\theta_i) - C(\theta^* + \Delta\theta_j) + C(\theta^*)}{\Delta\log\theta_i \Delta\log\theta_j} \tag{5}$$

and a parameter range scaling where each parameter was scaled by its maximum range

$$H_{\theta^*} = \frac{\partial^2 C}{\partial \Theta_i \partial \Theta_j} = \frac{C(\theta^* + \Delta\theta_i + \Delta\theta_j) - C(\theta^* + \Delta\theta_i) - C(\theta^* + \Delta\theta_j) + C(\theta^*)}{\Delta\Theta_i \Delta\Theta_j} \tag{6}$$

where $\Theta_i = \Delta\theta_i / (\theta_{max} - \theta_{min})$.

To construct our finite-difference scaled Hessian matrices, a set of best fit parameters that are optimized to minimise the cost (maximise model performance) are found (see optimization section - Methods). We then work to map out the surrounding area by deviating each parameter value and recalculating the cost function. Every parameter deviation and subsequent cost calculation allows us to effectively place a flag in our imaginary landscape, enabling us to survey the local topography. This then tells us more specifically how quickly the cost function changes in each parameter direction. These flagged values ultimately go towards filling out Equations (5) and (6).

Each model parameter was iteratively adjusted with incremental changes up to ± 20% of the difference between the maximum and minimum value that the parameter can take. Positive and negative deviations were used to understand how the model behaves when parameters were changed in each direction. The maximum deviation was subdivided into incremental changes which were chosen to be percentages of the maximum parameter deviation and structured in a way such that for each deviation d_i, the next largest deviation d_{i+1} was equal to two times the initial amount ($d_{i+1} = 2d_i$). For all model parameters, the specific deviations were

$$\left[\theta^*_i \pm \ : \ 1/16 \times 0.2G\theta_i,\ 2/16 \times 0.2G\theta_i,\ 4/16 \times 0.2G\theta_i,\ 8/16 \times 0.2G\theta_i,\ 16/16 \times 0.2G\theta_i\right], \tag{7}$$

where G θ_i is the parameter range.

A minimum of two steps in a specific direction were required for each parameter due to the fact that the diagonal elements of the Hessian are computed using the equation

$$C(\theta_i^* + \Delta\theta_i + \Delta\theta_j) = C(\theta_i^* + 2\Delta\theta_i) \tag{8}$$

which is valid when i = j.

The maximum deviation value of 20% was chosen to make sure it covered a sufficiently large interval so alterations in the cost function could be observed. However, only the first and second smallest deviations values were used to fill out the finite difference elements and the direction (positive or negative) that was associated with the largest change in the cost was selected. It is worthwhile noting that the values used for equation (8) need to be close enough together so the approximation can be done reasonably but not so close together that the approximation may be likely to be affected by small fluctuations that are not representative of the immediate state-space they are embedded in. To examine this, we looked at how the cost score changed each time this function was used across all possible intervals in all of the databases (43327 data points). As can be seen in the Supplementary material (Fig. S5), the two smallest steps only ever cause the cost function to deviate tiny amounts, and there were no obvious outliers. The third step also causes relatively small differences, although a small number of outliers begin to be produced. Given the size of the changes, our choice of using the first two steps thus seems reasonable.

Hessian eigenvalues and eigenvectors

The specific importance and influence of each parameter in the sloppy model can be further quantified via ranking methods designed to make use of the information provided by the eigenvalues and eigenvectors of the Hessian matrix. Eigenvalues and eigenvectors of the Hessian matrix can be determined by solving the following set of linear equations

$$H_\theta E = \lambda E \tag{9}$$

where E is an eigenvector that satisfies Eq. (9) and λ is its respective eigenvalue. In practice we achieve this by using the Matlab$_{TM}$ eig() command which returns a diagonal NxM matrix of eigenvalues and a NxM matrix of eigenvectors, where each eigenvector is a column of the returned matrix. The leading eigenvalue of the returned quantities, λ_i (largest magnitude) points in the stiffest direction – that which corresponds to the thinnest principal axis of the cost ellipsoids (see Fig. 3), defined as

$$w_i = \frac{1}{\sqrt{\lambda_i}} \tag{10}$$

where w_i is the width related to the i-th eigenvalue. The eigenvectors associated with the eigenvalues describe the stiffest/sloppiest combination of parameters. A total of 27 parameters in each of seven data sets are explored within this study. For each data set, the Hessian matrix of the cost function is computed using forward finite difference measures, giving a total of seven 27 × 27 Hessian matrices for each of the correlation and MSE cost function model types. For each matrix the eigenvalues and eigenvectors are calculated and are investigated using the relevant measures of interest which are explored below. We note that we only used positive eigenvalues and applied a lower limit of 10^{-10} on the acceptable magnitude. Any values that fell below this were ignored.

Sloppy parameter analyses results

We first examined the seven databases that we optimized CDP on. As can be seen in Fig. 4, the results of the SPA showed that the eigenvalues

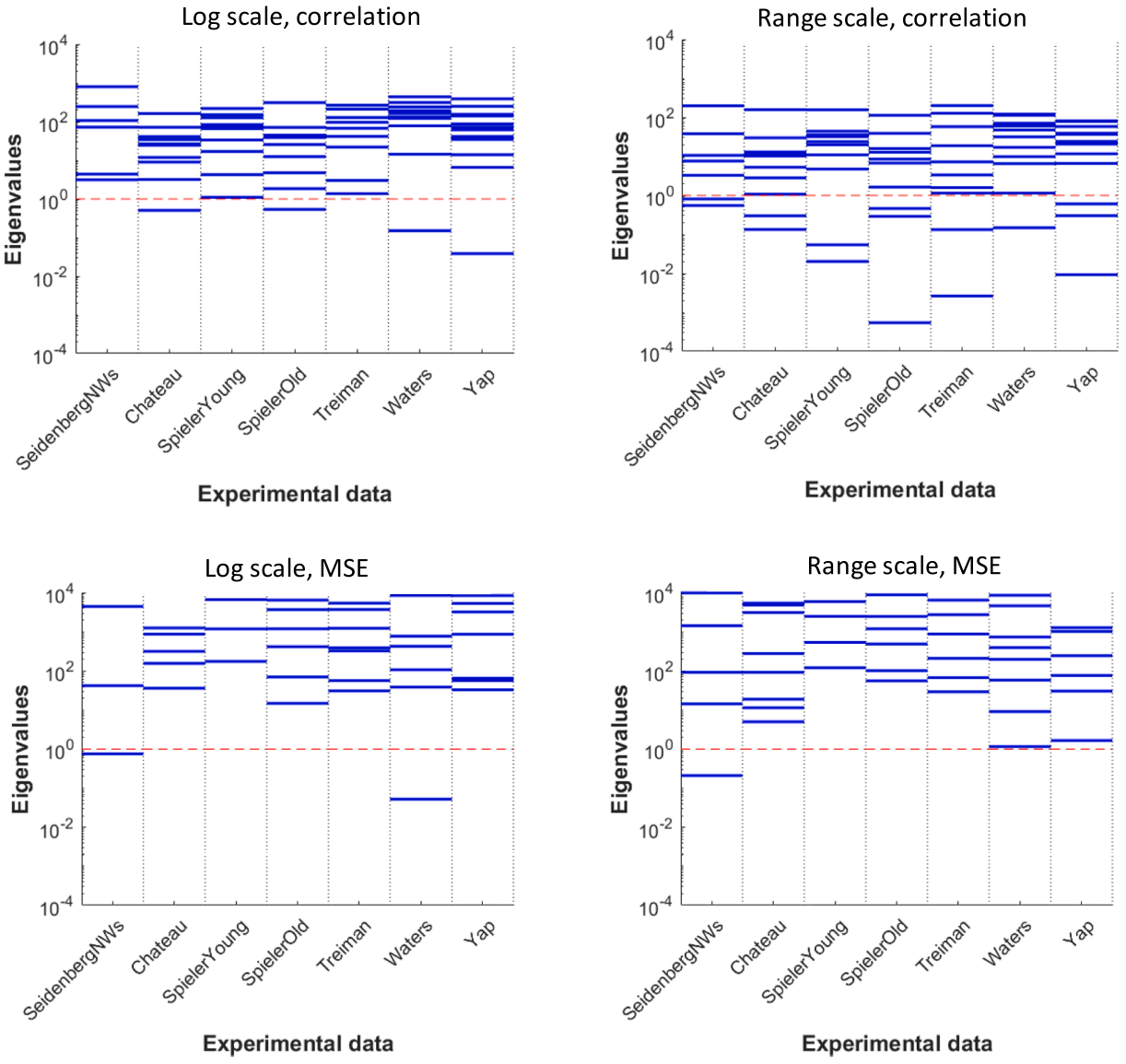

Fig. 4. Eigenvalue spectra generated from seven different datasets as a function of the type of scaling (log, range) and the type of cost function (correlation, MSE). Note: Y axis is log scaled and negative values are not shown. SeidenbergNWs = Seidenberg et al. (1994), Chateau = Chateau & Jared (2003), Spieler Young/Old = Spieler and Balota (1997/1998), Treiman = Treiman et al. (1995), Waters = Seidenberg & Waters (1989), Yap = Yap & Balota (2009).

calculated from each database showed a typical sloppy spectrum, with most of the variability always predicted by a small number of eigenvalues and many that have essentially no effect on the behavior of the model.

In particular, in the graph, each blue line represents an eigenvalue, and the Y axis represents how large it is. Because the Y-axis uses a log scale, the fairly linear spacing of the eigenvalues in each database means that there is an exponential increase in how large the value of each eigenvalue is. In addition, because the value of an eigenvalue is directly related to the extent to which the eigenvalue affects the performance of the model (e.g., an eigenvalue of 10 means that it accounts for 10 times more of the differences in model performance than a value of 1), it means that a small number of eigenvalues account for most of the variation in the performance of the model in each dataset. Thus, across all of the simulations, there are a small number of dimensions in the data that are responsible for most of the model performance (note that these should not be confused with dimensions of the raw parameter axes – i.e., the parameters that our model uses – for example the 'X' and 'Y' dimensions of a map).

This distribution of eigenvalues found is very important because it means that the model can potentially be tractably analysed. If every dimension had a similar effect on performance (i.e., the eigenvalues were all similar) the results would be much more difficult to interpret. This idea is very similar to principal component analyses (PCA) where a small number of components are typically found that explain most of the variance but only the top ones are used for further analyses. In our case, the large eigenvalues represent the strength of the eigenvectors that are associated with the most change in model performance. Since each eigenvector can be thought of as pointing in a direction (as it can be in PCA), then the raw parameters that are pointing in a similar direction as the eigenvectors with the large eigenvalues will be those that change model performance the most.

The differences between the simulations are also interesting and appear largely consistent with our observations in terms of how the correlation and MSE cost functions differ. As can be seen, the MSE cost function appeared to cause a greater range of eigenvalues than the correlation cost function, typically spanning across almost 4 orders of magnitude. Alternatively, the correlation cost function eigenvalues spanned over 2–3 orders of magnitude, which is less than the results from other models that have been reported in different areas. This suggests that the state-space of the MSE cost function models have stiffer dimensions than the correlation cost function models. Alternatively, the type of scaling had much less effect on the eigenvalues.

Next, we examined which parameters were important in CDP. We did this by calculating parameter rankings for all of the parameters based on the eigenvalues and eigenvectors found for each dataset (see Supplementary material for how this calculation was performed). This makes it possible to analyse how strongly each parameter was aligned with the strongest eigenvectors. High values represent parameters that are stiff parameters and align well with eigenvectors with high values. The results are shown in Fig. 5.

The results show an overall pattern that can be summarized as follows: First, the models optimized with the MSE cost function had stiffer parameters compared to models optimized with the correlation cost function. This suggests that optimizing on actual values (e.g., raw RTs) rather than correlations provides stronger constraints with respect to which parameters can affect model performance. Second, the number of parameters that could affect the models is much smaller than the total number of parameters. Third, the parameters that affect model performance tend to be similar across different data sets. As noted above, whilst there is no clear cut distinction between what is sloppy and what is not, even removing values less than 0.05 as was done in the figure removes the majority of values. Whilst this number is arbitrary given there is a distribution of values, these are parameters that have essentially no meaningful effect on the model – Thus most parameters have no meaningful effect on the model. In addition, whilst this number is not as intuitive as a SD or other well known statistic, we have offered an additional method of parameter ranking in the Supplementary material that produces similar results.

The results from the Seidenberg nonwords support the validity of the results in a more specific way. Notably, no orthographic parameters associated with the lexical route caused much change in model performance. This is not especially surprising because the lexical orthographic parameters are less useful with nonwords than words because only words can be retrieved from the lexicons. This result is consistent with the optimization data where nonwords were also less affected than words from lexical parameters.

Parameters that did not affect CDP's performance

One of the most illuminating aspects of the results are not the parameters that affected CDP, but those that never or very rarely did. This either means that they do not affect the results much or that they must work in complex combinations with other parameters.

In terms of our modelling, the most important observation is that all of the inhibitory and lateral inhibition parameters up until the phonological lexicon showed only very weak effects. This suggests these parameters can take a wide range of values and as long as they are set within very wide bounds, it makes little difference to model performance. Spatially, one might think of this as a flat-topped hill, where it is possible to climb to the top of a hill, but then there is a large flat space where it is possible to walk around without changing the altitude. This result could thus occur because once the parameters are set to minimize errors, there may be a very large bound in which they elicit essentially the same behavior. If there is enough inhibition from letters to the orthographic lexicon, for example, it does not make much difference if there is more. This is not because the search space was too constrained to never produce errors – it is possible to change these parameters by hand to cause CDP to create errors, so the bounds were not simply so tight as to exclude this possibility.

The fact that most of the inhibitory parameters do very little to the performance of CDP within a large range is an important insight for a number of reasons. First, it suggests that experiments that try and manipulate inhibitory context, such as when the effect of words that look like others words is examined, otherwise known as neighhorhood effects (e.g., Andrews, 1997), may be difficult to interpret. This is because if the parameter associated with this has very little effect on behavior, then studies examining it may suffer from power problems in that the effect of inhibition would be hard to find. In addition, even if differences were found, a tiny modulation of a stiff parameter could cause bigger differences than a large modulation of a sloppy inhibitory parameter. Thus, any effect found might potentially come from small confounds that produce unexpected changes of a stiff parameter.

A second set of parameters that had only a weak effect on any of the models were the excitatory feed*back* parameters (i.e., phonemes to phonological lexicon, phonological lexicon to orthographic lexicon, orthographic lexicon to letters). These only very occasionally affected model performance, and when they did they were generally much weaker than their feedforward counterparts. Thus, letter to word excitation affected the model more than word to letter excitation. This suggests it should be quite difficult, though not impossible, to find feedback effects because they are likely to be relatively small compared to feedforward effects. This is what has typically been found experimentally (e.g., Balota et al., 2004; Perry, 2003; Ziegler et al., 2008). Interestingly, as can be seen in Fig. 5, the parameters responsible for feedback effects had greater values with the correlation cost function compared to the MSE models. Thus, the way the models are optimized causes them to make different predictions.

Whilst knowing if a parameter is stiff or sloppy is useful, it is worthwhile noting that even if parameters do not have a large effect on model performance, this does not mean they are unimportant and could simply be dropped. One reason for this is that some of the parameters are of cognitive interest and thus they are hypotheses about possible

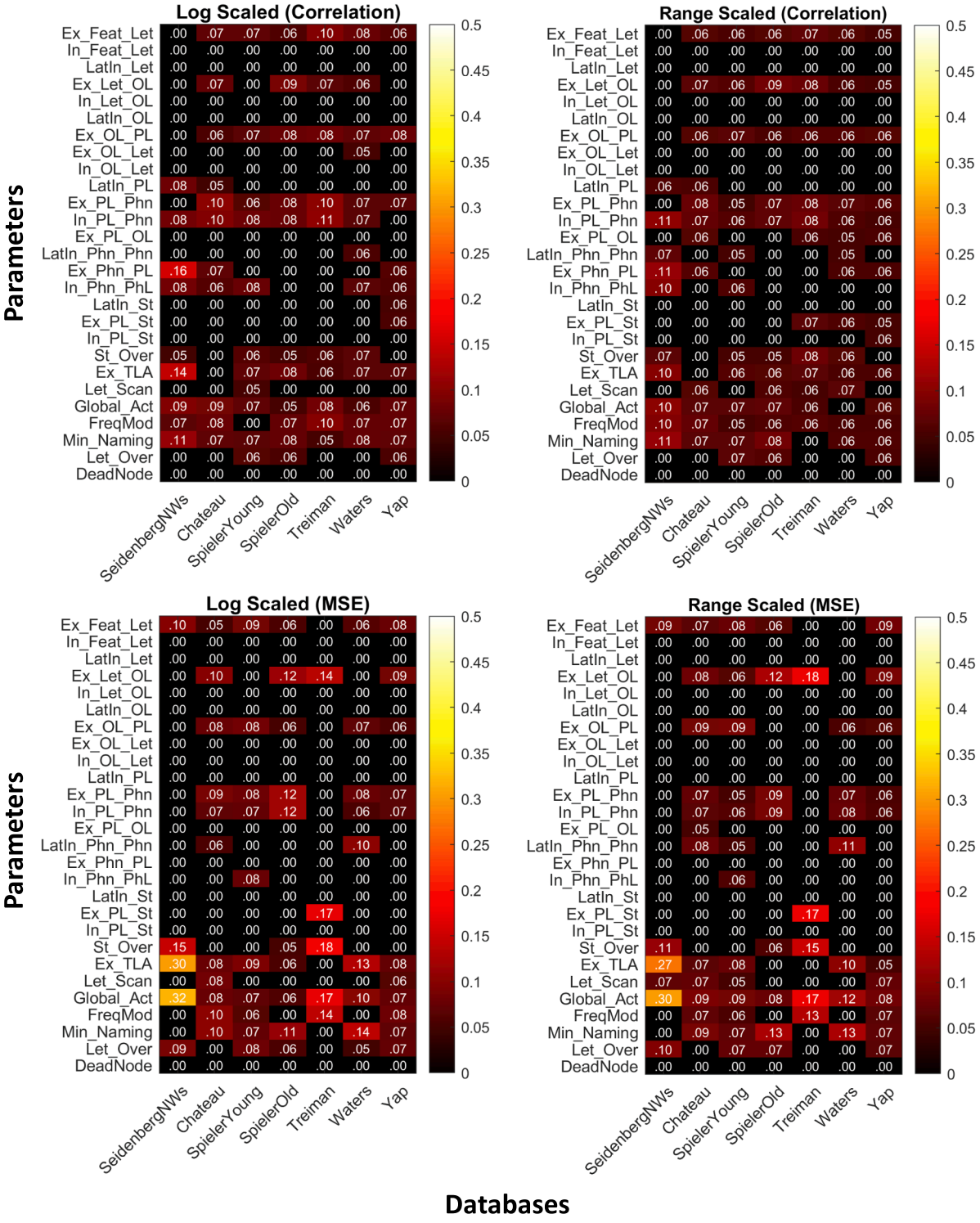

Fig. 5. Values of parameter rankings in different experiments for models as a function of cost function type (correlation, MSE) and scaling (log, range). Note: Values <.05 are displayed as.00. SeidenbergNWs = Seidenberg et al. (1994), Chateau = Chateau & Jared (2003), Spieler Young/Old = Balota (1997/1998), Waters = Seidenberg & Waters (1989), Yap = Yap & Balota (2009), Ex = Excitatory, In = Inhibitory, LatIn = Lateral inhbition, Feat = Feature, Let = Letter, OL = Orthographic lexicon, PL = Phonological lexicon, Phn = Phoneme, TLA = Two-later associative network, Let_Scan = Letter scanning time, FreqMod = Frequency modifier, Min_Naming = Minimum naming criterion, St = Stress, St_Over = Minimum stress naming criterion, Let_Over = Grapheme parsing letter threshold, DeadNode = Dead node level.

processing dynamics, not just there to somehow make the model perform better. For example, the feedback parameters are there because the model predicts that feedback always occurs and that feedback has a stronger influence on performance in some circumstances, such as in lexical decision (e.g., Stone and Van Orden, 1993) or word superiority experiments (e.g., McClelland & Johnston, 1977) where we assume the same cognitive architecture is used. A good example of this is by Stone and Van Orden (1993). They examined the extent to which consistency of the spelling-sound mapping of words affected behaviour in a lexical decision task. Previous to their study, only null effects of this variable had typically been found. They showed that an effect could be found by making the task very difficult. Assuming that this effect comes from feedback between phonology and orthography, removing feedback parameters to simplify the model would also remove the model's ability to predict these results. Thus, more parameters are used even though removing these parameters accounts for little quantitative performance in CDP (Perry et al., 2007; Zorzi, 2010).

Other parameters

There were a number of other parameters that had some effect on the model. These include the TLA excitatory parameter, letter-letter scan time, global activation, the frequency sensitivity of the lexicons, and the minimum naming criterion. There are also the stress parameters, although these are not so important in the models optimized on only monosyllables and so are more difficult to evaluate. As can be seen, the TLA excitation (strength of sublexical phonology generated) and global activation (slope of activation function) parameters were generally important in most of the experiments with the models. The fact that more than one of these parameters is associated with high eigenvalues across a number of experiments is interesting because it predicts that they should be quite difficult to deconfound in experiments unless an experimental manipulation could be found that had an effect on one but little effect on the other. The other parameters tended to be important in only a subset of the databases.

In summary, the results of the analyses of the parameters across seven large databases provide a number of useful insights. Notably, the number of parameters that affect CDP is much more limited than the number of parameters it has. This suggests CDP's predictions are much more constrained than what might be assumed from the number of parameters it has. The results also show that quite similar experiments can produce different patterns of sensitivity with the model's parameters. A consequence of this is that having accurate intuitions about the key factors in certain experiments according to the model is likely to be difficult, and only testing the differences with the stiff parameters can provide the correct insights. More generally, there are also effects that are commonly referred to in the literature on reading that have direct parameter equivalents in CDP. These include the minimum naming criterion and parameters to do with route emphasis such as the TLA excitation parameter (i.e., the dominance of phonology produced by the sublexical vs. lexical route), and these were often important in the model. This is exciting and provides some degree of validation that the parameters that are important in the model and that people often assume are the cause of underlying effects examined may be essentially referring to the same thing.

Effect of model fit and types of scaling

An important aspect of the validity of the results is whether somewhat arbitrary choices make differences to the outcomes. Notably, we can ask how much difference the actual cost function used and type of scaling made to the results. We used two different cost functions and two different types of scaling for the parameters. In terms of the type of cost function (see Fig. 5) the results were relatively similar, although the models using the correlation cost function produced weak effects across a greater number of parameters. This is unsurprising, because, as noted above, using a MSE cost function forces the model to optimize to a single set of values, unlike correlations which may allow a greater range of solutions.

The biggest difference in our data in terms of stiff parameters was seen with the Global Activation parameter (which determines the slope of the activation function). It was stiffer with models using a MSE compared to correlation cost function. Given its strength and function, this suggests that it is the probably the most important parameter for allowing the model to produce overall RTs similar to that of the actual data. With nonwords, it appears to work in conjunction with the TLA excitation parameter (strength of phonology from the sublexical route). The TLA parameter is likely to have less effect with words because phonology is also generated lexically from word recall, making it less difficult to produce phonemes that go above the minimum naming criterion (i.e., threshold at which phonemes can be read aloud).

Apart from the cost function, we also examined whether the scaling of the parameters affected the results (log vs. range bound). The results were extremely similar with both the correlation and MSE cost function. This is promising because it suggests that different types of scaling produce largely equivalent results. The results are also important because they suggest that the parameter bounds were likely to have been set within reasonable limits. Finally, it also shows that even though the eigenvalues were somewhat affected by the different scaling, such differences did not appear to affect predictions about the sloppiness of the parameters much. Thus, the final results of the SPA appear more stable than just the eigenvalues.

Differences between the sloppy parameter analyses and optimization

The results of the SPA are not only interesting in their own right, but they give complementary information with results from the optimization. As can be seen in Figs. 2 and 4, there are some similarities and differences. In terms of similarities, most of the parameters that had relatively tight ranges when optimized (i.e., those with small SDs), also tended to be stiff in the sloppy parameter analyses (e.g., Global activation). However, the letter scan time was an exception to this, where its effect on model performance was relatively weak. This parameter controls the speed at which individual letters are processed by the sublexical route of the model. This suggests that the effect that this parameter has on performance is likely to be intercorrelated with other parameters that produce a similar but stronger effect. There were also many parameters with values that were essentially randomly assigned that also did not appear to cause any meaningful effects on model performance.

Another type of dissociation of interest are the parameters that affected model performance but could take a wide range of values (e.g., the feature-to-letter excitation parameter). This is interesting because related methods have documented a similar pattern. For example, Holmes (2015; see also Holmes & Trueblood, 2018), describe a method by which posterior distributions of parameters can be estimated. A phenomenon they note with this method is that when intercorrelated variables make large changes on the same dimension, it causes the posterior distributions of the parameters to become broad. They take this as indicating sloppy behavior because more than one variable can account for the same behavior – that is the parameters are structurally unidentifiable. Thus, a wide range of parameter values may predict the same behavior because a movement in one parameter can be compensated by another. The analyses used here makes it simple to distinguish between this phenomena and parameters that can take a wide distribution of values because they simply do not affect the model much. As noted in the introduction, in the first case, the parameters can be identified because they still change model performance. In the second case, they do not change model performance much in the SPA and have a relatively random distribution in the optimized models.

Conclusion

Challenges in understanding how complex models work and the risk of overfitting has been often addressed in cognitive modelling by trading off predictive accuracy with model simplification. Our results show that

a cognitive model with a complex architecture and a large number of parameters like CDP does not overfit the empirical data even when using a parameter optimization algorithm. Notably, optimizing the model on one dataset yielded parameters that generalized appropriately to many other datasets. Most importantly, parameter optimization allowed us to introduce SPA, a state-of-the-art mathematical technique, to quantify the effects of different parameters on model performance. This analysis showed that the parameters that do and do not meaningfully affect the performance of a complex cognitive model like CDP can be identified. The parameters also showed an exponential sensitivity distribution. That is, a small number of parameters produced the majority of quantitative changes in the performance of the model and many parameters had almost no effect.

This type of distribution of parameters is found across many areas of science (e.g., Gutenkunst et al., 2007a) and is important because it means that a model can be tested by designing experiments that are linked to aspects of the model that are predicted to meaningfully affect performance and are thus testable (Gutenkunst et al., 2007b). It also means that some of the model's explanations of experiments could be a-priori ruled out. For example, it is common to assume that people's response threshold can change in different reading experiments (e.g., Besner & Reynolds, 2005; Perry et al., 2010a). People have tested this by manipulating stimulus difficulty, with the idea being that making some types of stimuli more difficult to process will cause people to set a higher response threshold for all stimuli. If the response threshold parameter was sloppy (unlike as found in the simulations), the model could not predict this. This means any effects found would need to be predicted due to movements of the stiff parameters. This is also a problem for trying to predict relatively weak effects. Even if effects could be found in experiments, they could occur in the model not because of a direct link between what is manipulated and a single parameter in the model, but what is being manipulated causing covariation with different stiff parameters that do affect performance. There is now a reasonable amount of literature on how experiments can be designed to test models based on an analyses of the sloppiness of their parameters (e.g., Casey et al., 2007). However, further discussion of it is beyond the scope of this work.

More generally, whilst here we only examined CDP, similar analyses could be applied to different types of cognitive models, such as other language models (e.g., WEAVER++, Levelt et al., 1999) or models of different phenomena, such as models of child development (e.g., van Geert, 1998). These analyses would allow the dynamics of different models to be further investigated and understood.

Apart from using sloppy analyses to examine single models, it would be interesting to perform these analyses across many cognitive models. Notably, one could investigate the extent to which the main computational mechanisms in different cognitive models that have been proposed, such as interactive activation (e.g., McClelland & Rumelhart, 1981), selection (e.g., Levelt et al., 1999), and decision making mechanisms (e.g., Brown & Heathcote, 2008; Ratcliff et al., 2004; Usher & McClelland, 2001), affect parameter sensitivity. This would give great insights into what experimental paradigms are useful for testing these core mechanisms. For example, if the parameters underlying a model's decision making mechanism (which many models have but CDP does not) tend to be the stiff ones and the other parameters are sloppy, it would mean the computations that occur before a decision is made may be difficult to elucidate. Alternatively, if parameters associated with decision making tend to be sloppy, it would suggest that effects of processing that come before them may be meaningfully examined. In this case, even though the final response given in the task is a simple decision, predictable differences in the decision making times would be largely driven by processes occurring before that decision. If there were such general patterns in areas of scientific inquiry, as observed in the current case study (i.e., reading), SPA would not only allow individual models to be better understood but could potentially give a-priori information on the best types of experiments that could be used to test their assumptions.

CRediT authorship contribution statement

Conrad Perry: Conceptualization, Data curation, Formal analysis, Funding acquisition, Methodology, Project administration, Resources, Software, Supervision, Validation, Visualization, Writing – review & editing. **Rick Evertz:** Conceptualization, Formal analysis, Methodology, Software, Supervision, Validation, Visualization, Writing – review & editing. **Marco Zorzi:** Conceptualization, Funding acquisition, Supervision, Writing – review & editing. **Johannes C. Ziegler:** Conceptualization, Funding acquisition, Supervision, Writing – review & editing.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Any data not already on the osf site for this project (https://osf.io/bkn6j/) is available from CP.

Acknowledgements

Correspondence with respect to this article can be sent to ConradPerry@gmail.com. The main files to run the sloppy analyses and generate the files and graphs are available on OSF at https://osf.io/bkn6j/. Further data from this project is available from the first author on request.

The authors declare that there is no conflict of interest. All authors approved the final version of the manuscript. This research has been supported by the Australian Research Council (Grant DP210100936). Johannes Ziegler is supported by the Institute of Convergence ILCB (ANR-16-CONV-0002) and the Excellence Initiative of Aix-Marseille University A*MIDEX (ANR-11-IDEX-0001-02).

Appendix A. Supplementary material

Supplementary data to this article can be found online at https://doi.org/10.1016/j.jml.2023.104468.

References

Adelman, J. S., Marquis, S. J., Sabatos-DeVito, M. G., & Estes, Z. (2013). The unexplained nature of reading. *Journal of Experimental Psychology: Learning, Memory, and Cognition, 39*(4), 1037–1053. https://doi.org/10.1037/a0031829

Adelman, J. S., Sabatos-DeVito, M. G., Marquis, S. J., & Estes, Z. (2014). Individual differences in reading aloud: A mega-study, item effects and some models. *Cognitive Psychology, 68*, 113–160. https://doi.org/10.1016/j.cogpsych.2013.11.001

Andrews, S. (1997). The effect of orthographic similarity on lexical retrieval: Resolving neighborhood conflicts. *Psychonomic Bulletin & Review, 4*, 439–461. https://doi.org/10.3758/BF03214334

Apgar, J. F., Witmer, D. K., White, F. M., & Tidor, B. (2010). Sloppy models, parameter uncertainty, and the role of experimental design. *Molecular Biosystems, 6*(10), 1890–1900. https://doi.org/10.1039/B918098B

Balota, D. A., Cortese, M. J., Sergent-Marshall, S. D., Spieler, D. H., & Yap, M. (2004). Visual word recognition of single-syllable words. *Journal of Experimental Psychology General, 133*(2), 283–316. https://doi.org/10.1037/0096-3445.133.2.283

Balota, D. A., & Spieler, D. (1998). The utility of item-level analysis in model evaluation: A reply to Seidenberg and Plaut. *Psychological Science, 9*, 238–240. https://doi.org/10.1111/1467-9280.00047

Balota, D. A., Yap, M. J., Cortese, M. J., Hutchison, K. A., Kessler, B., Loftis, B., … Treiman, R. (2007). The English lexicon project. *Behavior Research Methods, 39*(3), 445–459. https://doi.org/10.3758/BF03193014

Besner, D., & Reynolds, M. (2005). Contextual control over lexical and sublexical routines when reading English aloud. *Psychonomic Bulletin & Review, 12*, 113–118. https://doi.org/10.3758/BF03196355

Brown, S. D., & Heathcote, A. (2008). The simplest complete model of choice response time: Linear ballistic accumulation. *Cognitive Psychology, 57*(3), 153–178. https://doi.org/10.1016/j.cogpsych.2007.12.002

Casey, F. P., Baird, D., Feng, Q., Gutenkunst, R. N., Waterfall, J. J., Myers, C. R., … Sethna, J. P. (2007). Optimal experimental design in an epidermal growth factor

receptor signalling and down-regulation model. *IET Systems Biology, 1*(3), 190–202. https://doi.org/10.1049/iet-syb:20060065
Chang, Y.-N., Monaghan, P., & Welbourne, S. (2019). A computational model of reading across development: Effects of literacy onset on language processing. *Journal of Memory and Language, 108*. https://doi.org/10.1016/j.jml.2019.05.003
Chateau, D., & Jared, D. (2003). Spelling-sound consistency effects in disyllabic word naming. *Journal of Memory and Language, 48*(2), 255–280. https://doi.org/10.1016/S0749-596x(02)00521-1
Clerc, M. (2012). Standard Particle Swarm Optimisation. hal-00764996.
Coltheart, M., Rastle, K., Perry, C., Langdon, R., & Ziegler, J. C. (2001). DRC: A computational model of visual word recognition and reading aloud. *Psychological Review, 108*, 204–256. https://doi.org/10.1037/0033-295X.108.1.204
Engberg, R., Nuthmann, E., Richter, E., & Kliegl, R. (2005). SWIFT: A dynamical model of saccade generation during reading. *Psychological Review, 112*(4), 777–813. https://doi.org/10.1037/0033-295X.112.4.777
Esmin, A. A. A., Coelho, R. A., & Matwin, S. (2015). A review on particle swarm optimization algorithm and its variants to clustering high-dimensional data. *Artificial Intelligence Review, 44*, 23–45. https://doi.org/10.1007/s10462-013-9400-4
Farrell, S., & Lewandowsky, S. (2018). *Computational modelling of cognition and behavior*. Cambridge University Press.
Francis, B. L., & Transtrum, M. K. (2019). Unwinding the model manifold: Choosing similarity measures to remove local minima in sloppy dynamical systems. *Physical Review E, 100*, Article 012206. https://doi.org/10.1103/PhysRevE.100.012206
Grainger, J., & Jacobs, A. M. (1996). Orthographic processing in visual word recognition: A multiple read-out model. *Psychological Review, 103*, 518–565. https://doi.org/10.1037/0033-295x.103.3.518
Gutenkunst, R. N., Waterfall, J. J., Casey, F. P., Brown, K. S., Myers, C., & P., S. J. (2007a). Universally sloppy parameter sensitivities in systems biology models. *PLoS Computational Biology*, *15*(5), e1006694. https://doi.org/10.1371/journal.pcbi.0030189.
Gutenkunst, R. N., Casey, F. P., Waterfall, J. J., Myers, C. R., & Sethna, J. P. (2007). Extracting falsifiable predictions from sloppy models. *Annals of the New York Academy of Sciences, 1115*, 203–211. https://doi.org/10.1196/annals.1407.003
Harm, M. W., & Seidenberg, M. S. (1999). Phonology, reading acquisition, and dyslexia: Insights from connectionist models. *Psychological Review, 106*(3), 491–528. https://doi.org/10.1037/0033-295X.106.3.491
Hartoyo, A., Cadusch, P. J., Liley, D. T. J., & Hicks, D. G. (2019). Parameter estimation and identifiability in a neural population model for electro-cortical activity. *PLoS Computational Biology, 15*(5), e1006694.
Holmes, W. R. (2015). A practical guide to the Probability Density Approximation (PDA) with improved implementation and error characterization. *Journal of Mathematical Psychology, 68–69*, 13–24. https://doi.org/10.1016/j.jmp.2015.08.006
Holmes, W. R., & Trueblood, J. S. (2018). Bayesian analysis of the piecewise diffusion decision model. *Behavioral Research, 50*, 730–743. https://doi.org/10.3758/s13428-017-0901-y
Holmes, W. R., Trueblood, J. S., & Heathcote, A. (2016). A new framework for modeling decisions about changing information: The Piecewise Linear Ballistic Accumulator model. *Cognitive Psychology*, *85*, 1-29. https://doi.org/j.cogpsych.2015.11.002.
Jacobs, A. M., & Grainger, J. (1994). Models of visual word recognition: Sampling the state of the art. *Journal of Experimental Psychology: Human Perception and Performance, 20*, 1311–1334.
Kawamoto, A. H., Liu, Q., Mura, K., & Sanchez, A. (2008). Articulatory preparation in the delayed naming task. *Journal of Memory and Language, 58*(2), 347–365. https://doi.org/10.1016/j.jml.2007.06.002
Kennedy, J., & Eberhart, R. (1995). Particle Swarm Optimization. *Proceedings of ICNN'95 - International Conference on Neural Networks*. https://doi.org/10.1109/ICNN.1995.488968.
Lee, M. D., & Webb, M. R. (2005). Modeling individual differences in cognition. *Psychonomic Bulletin & Review, 12*(4), 605–621. https://doi.org/10.3758/BF03196751
Levelt, W. J. M., Roelofs, A., & Meyer, A. S. (1999). A theory of lexical access in speech production. *Behavioral and Brain Sciences, 22*, 1–75. https://doi.org/10.1017/S0140525X99001776
Lever, J., Krzywinski, M., & Altman, N. (2016). Model selection and overfitting. *Nature Methods, 13*, 703–704. https://doi.org/10.1038/nmeth.3968
Machta, B. B., Chachra, R., Transtrum, M. K., & Sethna, J. P. (2013). Parameter space compression underlies emergent theories and predictive models. *Science, 342*(6158), 604–607. https://doi.org/10.1126/science.1238723
Mannakee, B. K., Ragsdale, A. P., Transtrum, M. K., & Gutenkunst, R. N. (2016). Sloppiness and the geometry of parameter space. In *Uncertainty in Biology* (pp. 271–299). Switzerland: Springer. https://doi.org/10.1007/978-3-319-21296-8.
McClelland, J., & Rumelhart, D. (1981). An interactive activation model of context effects in letter perception: Part 1. An account of basic findings. *Psychological Review, 88*(5), 375–407. https://doi.org/10.1037/0033-295X.88.5.375
McClelland, J. L. (2009). The place of modeling in cognitive science. *Topic in Cognitive Science, 1*(1), 11–38. https://doi.org/10.1111/j.1756-8765.2008.01003.x
McClelland, J. L., & Johnston, J. C. (1977). The role of familiar units in perception of words and nonwords. *Perception & Psychophysics, 22*(3), 249–261.
Panas, D., Amin, H., Maccione, A., Muthmann, O., van Rossum, M., Berdondini, L., & Hennig, M. H. (2015). Sloppiness in spontaneously active neuronal networks. *Journal of Neuroscience, 35*(22), 8480–8492. https://doi.org/10.1523/JNEUROSCI.4421-14.2015
Paulesu, E., Bonandrini, R., Zapparoli, L., Rupani, C., Mapelli, C., Tassini, F., … Zorzi, M. (2021). Effects of Orthographic Consistency on Bilingual Reading. *Human and Computer Simulation Data. Brain Sciences, 11*(7). https://doi.org/10.3390/brainsci11070878
Perry, C. (2003). A phoneme– grapheme feedback consistency effect. *Psychonomic Bulletin & Review, 10*(2), 392–397. https://doi.org/10.3758/BF03196497
Perry, C. (2018). Reading orthographically strange nonwords: Modelling backup strategies in reading. *Scientific Studies of Reading, 22*(3), 264–272. https://doi.org/10.1080/10888438.2018.1433673
Perry, C., Ziegler, J. C., Braun, M., & Zorzi, M. (2010a). Rules versus statistics in reading aloud: New evidence on an old debate. *The European Journal of Cognitive Psychology, 5*, 798–812. https://doi.org/10.1080/09541440902978365
Perry, C., Ziegler, J. C., & Zorzi, M. (2007). Nested incremental modeling in the development of computational theories: The CDP+ model of reading aloud. *Psychological Review, 114*(2), 273–315. https://doi.org/10.1037/0033-295X.114.2.273
Perry, C., Ziegler, J. C., & Zorzi, M. (2010b). Beyond single syllables: Large-scale modeling of reading aloud with the Connectionist Dual Process (CDP++) model. *Cognitive Psychology, 61*(2), 106–151. https://doi.org/10.1016/j.cogpsych.2010.04.001
Perry, C., Ziegler, J. C., & Zorzi, M. (2013). A computational and empirical investigation of graphemes in reading. *Cognitive Science, 37*(5), 800–828. https://doi.org/10.1111/cogs.12030
Perry, C., Ziegler, J. C., & Zorzi, M. (2014a). When silent letters say more than a thousand words: An implementation and evaluation of CDP++ in French. *Journal of Memory and Language, 72*, 98–115. https://doi.org/10.1016/j.jml.2014.01.003
Perry, C., Ziegler, J. C., & Zorzi. (2014b). CDP++. Italian: Modelling sublexical and supralexical inconsistency in a shallow orthography. *Plos one, 9*(4), Article e94291. https://doi.org/10.1371/journal.pone.0094291
Perry, C., Ziegler, J. C., & Zorzi, M. (2019). Understanding dyslexia through personalized large-scale computational models. *Psychological Science, 30*(3), 386–395. https://doi.org/10.1177/0956797618823540
Pitt, M. A., Kim, W., Navarro, D. J., & Myung, I. J. (2006). Global Model Analyses by Paramater Space Partitioning. *Psychological Review*, *13*(1), 57-83. https://doi.org/0.1037/0033-295X.113.1.57.
Pitt, M. A., & Myung, I. J. (2002). When a good fit can be bad. *Trends in Cognitive Sciences, 6*(10), 421–425. https://doi.org/10.1016/s1364-6613(02)01964-2
Pitt, M. A., Myung, I. J., & Zhang, S. (2002). Toward a method of selecting among computational models of cognition. *Psychological Review, 109*(3), 472–491. https://doi.org/10.1037//0033-295X.109.3.472
Plaut, D. C., McClelland, J. L., Seidenberg, M. S., & Patterson, K. (1996). Understanding normal and impaired word reading: Computational principles in quasi-regular domains. *Psychological Review, 103*(1), 56–115. https://doi.org/10.1037/0033-295X.103.1.56
Ponce-Alvarez, A., Mochol, G., Hermoso-Mendizabal, A., De la Rocha, J., & Deco, G. (2020). Cortical state transitions and stimulus response evolve along stiff and sloppy parameter dimensions, respectively. *Elife, 9*, Article 353268. https://doi.org/10.7554/eLife.53268
Pritchard, S. C., Coltheart, M., Palethorpe, S., & Castles, A. (2012). Nonword reading: Comparing dual-route cascaded and connectionist dual-process models with human data. *Journal of Experimental Psychology: Human Perception and Performance, 38*(5), 1268–1288. https://doi.org/10.1037/a0026703
Quinn, K. N. (2019). *Patterns of Structural Hierarchies in Complex Systems* [PhD Thesis, Cornell University].
Rastle, K., & Coltheart, M. (1998). Whammies and double whammies: The effect of length on nonword reading. *Psychonomic Bulletin and Review, 5*, 277–283. https://doi.org/10.3758/BF03212951
Ratcliff, R., Gomez, P., & McKoon, G. (2004). A diffusion model account of the lexical decision task. *Psychological Review*, *111*(1), 159-118. https://doi.org/210.1037/0033-295X.111.1.159.
Reichle, E. D., Rayner, K., & Pollatsek, A. (2003). The E-Z Reader model of eye-movement control in reading: Comparisons to other models. *Behavioral and Brain Sciences, 26*, 445–476. https://doi.org/10.1017/s0140525x03000104
Rey, A., Courrieu, P., Madec, S., & Grainger, J. (2013). The unbearable articulatory nature of naming: On the reliability of word naming responses at the item level. *Psychonomic Bulletin & Review, 20*, 87–94. https://doi.org/10.3758/s13423-012-0336-5
Roberts, S., & Pashler, H. (2000). How persuasive is a good fit? A comment on theory testing. *Psychological Review, 107*(2), 358–367. https://doi.org/10.1037/0033-295X.107.2.358
Rogers, J. L., & Rowe, D. C. (2002). Theory development should begin (but not end) with good empirical fits: A comment on Roberts and Pashler (2000). *Psychological Review, 109*(3), 599–603. https://doi.org/10.1037/0033-295x.109.3.599
Russell-Buckland, J., Barnes, C. P., & Tachtsidis, I. (2019). A Bayesian framework for the analysis of systems biology models of the brain. *PLoS Computational Biology, 15*(4), e1006631.
Seidenberg, M. S. (2012). Computational models of reading. In M. Spivey, K. McRae, & M. Joanisse (Eds.), *The Cambridge Handbook of Psycholoinguistics* (pp. 186–203). Cambridge University Press. https://doi.org/10.1017/CBO9781139029377.010.
Seidenberg, M. S., & McClelland, J. L. (1989). A distributed, developmental model of word recognition and naming. *Psychological Review, 96*, 523–568. https://doi.org/10.1037/0033-295X.96.4.523
Seidenberg, M. S., & Plaut, D. C. (1998). Evaluating word reading models at the item level: Matching the grain of theory and data. *Psychological Science, 9*, 234–237.
Seidenberg, M. S., Plaut, D. C., Petersen, A. S., McClelland, J., & McCrae, K. (1994). Nonword pronunciation and models of word recognition. *Journal of Experimental Psychology: Human Perception and Performance, 20*, 1177–1196. https://doi.org/10.1037/0096-1523.20.6.1177
Seidenberg, M. S., & Waters, G. S. (1989). Word recognition and naming: A mega study [Abstract 30]. *Bulletin of the Psychonomic Society, 27*, 489.

Smith, A. C., Monaghan, P., & Huettig, F. (2021). The effect of orthographic systems on the developing reading system: Typological and computational analyses. *Psychological Review, 128*(1), 125–159. https://doi.org/10.1037/rev0000257

Spieler, D. H., & Balota, D. A. (1997). Bringing computational models of word naming down to the item level. *Psychological Science, 8*(6), 411–416. https://doi.org/10.1111/j.1467-9280.1997.tb00453.x

Stone, G. O., & Van Orden, G. C. (1993). Strategic control of processing in word recognition. *Journal of Experimental Psychology: Human Perception and Performance, 19*, 744–774. https://doi.org/10.1037/0096-1523.19.4.744

Transtrum, M. K., Machta, B. B., Brown, K. S., Daniels, B. C., Myers, C. R., & Sethna, J. P. (2015). Perspective: Sloppiness and emergent theories in physics, biology, and beyond. *The Journal of Chemical Physics, 143*(1), Article 010901. https://doi.org/10.1063/1.4923066

Transtrum, M. K., Machta, B. B., & Sethna, J. P. (2011). Geometry of nonlinear least squares with applications to sloppy models and optimization. *Physical Review E, 83*(3), Article 036701. https://doi.org/10.1103/PhysRevE.83.036701

Treiman, R., Mullennix, J., Bijeljac-Babic, R., & Richmond-Welty, E. D. (1995). The special role of rimes in the description, use, and acquisition of English orthography. *Journal of Experimental Psychology: General, 124*, 107–136. https://doi.org/10.1037/0096-3445.124.2.107

Usher, M., & McClelland, J. L. (2001). The time course of perceptual choice: The leaky accumulator model. *Psychological Review, 108*(3), 550–592. https://doi.org/10.1037/0033-295x.108.3.550

van Geert, P. (1998). A dynamic systems model of basic developmental mechanisms: Piaget, Vygotsky, and beyond. *Psychological Review, 4*(105), 634–677. https://doi.org/10.1037/0033-295X.105.4.634-677

Veksler, V. D., Myers, C. W., & Gluck, K. A. (2015). Model flexibility analysis. *Psychological Review, 122*(4), 755–769. https://doi.org/10.1037/a0039657

Villaverde, A. F., Barreiro, A., & Papchristodoulou, A. (2016). Structural identifiability of dynamic systems biology models. *PLoS Computational Biology, 12*(10), e10051.

Weekes, B. S. (1997). Differential effects of number of letters on words and nonword naming latency. *The Quarterly Journal of Experimental Psychology, 50*(A), 439–456. https://doi.org/10.1080/713755710

Wilson, R. C., & Collins, A. G. E. (2019). Ten simple rules for the computational modeling of behavioral data. *Elife, 8*, e49547.

Yap, M. J., & Balota, D. A. (2009). Visual word recognition of multisyllabic words. *Journal of Memory and Language, 60*, 502–529. https://doi.org/10.1016/j.jml.2009.02.001

Ziegler, J. C., Perry, C., & Zorzi, M. (2014). Modelling reading development through phonological decoding and self-teaching: Implications for dyslexia. *Philosophical Transactions of the Royal Society B: Biological Sciences, 369*, 20120397. https://doi.org/10.1098/rstb.2012.0397

Ziegler, J. C., Perry, C., & Zorzi, M. (2020). Learning to read and Dyslexia: From theory to intervention through personlized computational models. *Current Directions in Psychological Science, 29*(3), 293–300. https://doi.org/10.1177/0963721420915873

Ziegler, J. C., Petrova, A., & Ferrand, L. (2008). Feedback consistency effects in visual and auditory word Recognition: Where do we stand after more than a decade? *Journal of Experimental Psychology: Learning Memory and Cognition, 34*(3), 643–661. https://doi.org/10.1037/0278-7393.34.3.643

Zorzi, M. (2000). Serial processing in reading aloud: No challenge for a parallel model. *Journal of Experimental Psychology: Human Perception and Performance, 26*, 847–856. https://doi.org/10.1037/0096-1523.26.2.847

Zorzi, M. (2010). The connectionist dual process (CDP) approach to modeling reading aloud. *The European Journal of Cognitive Psychology, 22*(5), 836–860. https://doi.org/10.1080/09541440903435621

Zorzi, M., Houghton, G., & Butterworth, B. (1998). Two routes or one in reading aloud? A connectionist dual-process model. *Journal of Experimental Psychology: Human Perception and Performance, 24*(4), 1131–1161. https://doi.org/10.1037//0096-1523.24.4.1131