University of California
Santa Barbara

Reverse first principles: Weber's law and optimality in different senses

A thesis submitted in partial satisfaction
of the requirements for the degree

Master of Science
in
Psychology

by

Jason T. Wilkes

Committee in charge:

Professor Leda Cosmides, Chair
Professor John Tooby
Professor Tamsin German

June 2015

The Thesis of Jason T. Wilkes is approved.

Professor John Tooby

Professor Tamsin German

Professor Leda Cosmides, Committee Chair

June 2015

Reverse first principles:

Weber's law and optimality in different senses

by

Jason T. Wilkes

To Leda Cosmides, John Tooby, and Randy Gallistel,
for taking a risk.

Acknowledgements

The author would like to thank Leda Cosmides, Randy Gallistel, Tamsin German, Erin Horowitz, Jim Roney, and John Tooby.

Abstract

Reverse first principles:

Weber's law and optimality in different senses

by

Jason T. Wilkes

The relationship between optimality and evolvability is analyzed through a case study of Weber's law, a common property of many sensory systems across a wide array of species. After demonstrating a variety of senses in which Weber's law is mathematically optimal, we ask whether principled methods exist for evaluating such optimality analyses. We argue that at least one such method exists: examining the evolvability of a trait with respect to each of the different metrics that it happens to optimize. Through evolvability analyses of Weber's law, it is demonstrated that optimality-equivalent measures of phenotypic quality need not be selectively equivalent: a trait that is optimal by two measures may have very different behavior under selection for each. This non-equivalence allows different optimality analyses of the same phenomenon to be assessed by a standard other than intuition, and in a manner requiring fewer degrees of freedom than are needed to model selection from scratch. Two qualitatively different models of selection are explored: phenotypic selection, a basic form in which mutations directly affect the model phenotype, and embryological selection, a more exotic form in which mutations affect the algorithm by which the phenotype is built.

Contents

Chapter 1

Introduction

Not long after the founding of experimental psychology came one of its strangest discoveries: though sensory systems are not "perfect," their imperfections often take a predictable form. The simplest such form was also the first to be discovered, and its precise description has come to be known as Weber's law. Though it is not a "law" in any strict sense, it is phrased in a manner that potentially applies to any sensory system, in any species, at any life stage, and it has been observed in a wide variety of each (see (Akre & Johnsen, 2014) for an extensive review). However, the mathematical statement of Weber's law is so concise that its rather bizarre content is easy to miss. The law is simply written:

$$\Delta X = \epsilon X$$

What does this mean? By necessity, the above symbols refer to rather abstract quantities – quantities not tied to any single sensory system or zoological niche. The symbol X represents the magnitude of a sensory stimulus, measured in whatever units are appropriate to the sensory system in question. The symbol ΔX represents imperfections, measured by the so-called "just noticeable difference" (jnd) between two stimuli. Having presented the sensory system with a stimulus of magnitude X (say, a weight of X grams) sup-

pose we then present it with stimuli extremely close to X in magnitude. For sufficiently small differences, we can trick the system into believing the magnitudes are the same. The smallest difference that does not trick the system in this way is known as the "just noticeable difference," and is represented by the symbol ΔX.

Weber's law is often described as the proportionality of the jnd to the magnitude of the stimulus. Pause for a moment and reconsider the content of this result. One could be forgiven for expecting the "errors" or "imperfections" in a sensory system to be in some sense random. In practice, not only are they non-random, but they often appear to take a specific functional form. Stranger still, that functional form is shared between a large number of different sensory systems in a wide variety of species. Weber's law is simply the statement that the imperfections form a line: it states that ΔX is a linear function of the magnitude of the stimulus X, and ϵ is the slope of that line.

Without exception, the systems we study in psychology are the downstream effect of cumulative mutation and natural selection. This process "prefers" the most effective solution to an adaptive problem that it finds in the pool of available genetic variation. But selection is more engineer than physicist, and has no obvious concern with ensuring that the organisms thus constructed will obey a set of elegant, general laws that are easily describable in mathematical form. As such, whenever such a general law appears to be present, there is good reason to divert our attention and exert every effort to understand its origins, since the opportunity does not often arise in biological fields of inquiry.

However, we have just seen that one of the earliest discoveries of experimental psychology was a mathematical description of the shape of imperfections in an arbitrary sensory system, and this abstract description has somehow managed – in a surprisingly large number of cases over the last century and a half – to describe the real world. As fruitless as the search for general principles in biology often appears to be, it seems that selection has thrown us a crumb. To say the least, this is a crumb worth exploring.

1.1 A formidable literature

Historically, Weber's law has been studied primarily as a candidate property of sensory systems. In recent decades, however, Weber's law has been observed in systems more usefully described as "cognitive," rather than strictly "sensory." Examples include the representation of temporal duration in humans (Wearden & McShane, 1988; Grondin, Ouellet, & Roussel, 2001) rats (Gibbon & Church, 1981), and pigeons (Dews, 1970; Gibbon & Church, 1981), and the cognitive representation of number in humans (Dehaene, Dehaene-Lambertz, & Cohen, 1998) and rhesus macaques (Jordan & Brannon, 2006).

The literature on this topic has grown sufficiently large over the past century and a half that no brief literature review can do it justice. The following list is adapted from a table presented by Akre and Johnsen (2014) in an extensive review of this literature. The original (rather large) table summarizes a subset of the domains and species in which Weber's law has been observed, supplemented with known deviations from it when necessary.[1] These domains include:

- Auditory amplitude: in humans, birds, insects, anurans, misc mammals.
- Auditory frequency: in humans, parakeets (∼300-2000 Hz).
- Visual area: in humans, coyotes, seals.
- Visual intensity: in humans, misc species.
- Visual wavelength: in humans, goldfish, honeybees, butterflies.
- Numerical cognition: in humans, monkeys, fish, chimps, crows.
- Temporal duration in parasites, humans, pigeons, starlings.

[1] In such a large literature, the claim that Weber's law is "observed" in a given domain can only mean: observed to some reasonable degree of approximation, for some significant range of stimuli, though this range of stimuli is not always the entire dynamic range of a sensory system.

- Chemoreception: in bats, bees, ants, misc bacteria.
- Tactile pressure: in humans.
- Vibrational frequency: in humans, misc primates.
- Edge sharpness: in humans.
- Electroreception: in weakly electric fish.

There are many contexts in which Weber's law is not observed, including certain tasks within the same domains and species listed above (Bizo, Chu, Sanabria, & Killeen, 2006; Masin, 2009; Akre & Johnsen, 2014). For instance, Weber's law is often not observed in audition. One such example is discussed by Rabinowitz, Lim, Braida, and Durlach (1976): for pulsed tones of constant frequency (1000 Hz) and varying amplitude, Weber's law holds only for amplitudes in the range ∼10-40 dB, while sensitivity increases with amplitude outside that range. This is a genuine deviation from Weber's law, but how large a deviation is it? The Weber-range of ∼10-40 dB seems quite small. However, the decibel is a logarithmic unit, so this range is considerably larger than it appears at first glance. As such, while many genuine deviations from Weber's law are known, the units in which a physical quantity is expressed can have a significant effect on how large (or small) such deviations appear to our intuition. However far it may be from a universal principle, the zoological and sensory/cognitive breadth of Weber's law is striking, given the rarity of such general phenomena in biology.

1.2 Questions about "why" questions

Why is Weber's law such a widespread property of sensory systems? In biology, one way of addressing "why" questions is the optimality analysis. If a trait observed in

organisms appears to be optimal by some measure independent of biology – principles of engineering, probabilistic inference, algorithm design, etc. – then our confidence that the trait is a naturally selected adaptation is increased, and increased to the extent that the optimized measure is potentially relevant to fitness, and thus a potential target of selection.[2] However, suppose we are confronted with two or more optimality analyses of the same trait. This is a common occurrence, given the expanse of the modern scientific literature, but there is little discussion of precisely how different optimality analyses might be evaluated, so the task is generally left to our intuition, and to the elegance with which the result is expressed. Can we do better? Are there principled methods of comparing their relative strength, and assessing their individual weaknesses? Or are we simply forced to arrange them on a shelf and stare at them, like a collection of novelty nutcrackers?

In what follows, we begin with the question of how to model an arbitrary sensory system, a seemingly impossible task that any mathematical analysis or computer simulation of Weber's law appears to require. We then discuss various standards by which Weber's law is (and is not) optimal. Having done so, we find ourselves with a disorganized bag of different optimality analyses. This raises the question of how to evaluate such results. We argue that at least one such method exists: examining the evolvability of the trait in question with respect to each of the different metrics that it happens to optimize. The logic is as follows: A trait may be optimal with respect to some measure M, but if it is not evolvable under selection for M, then its optimality under M is of little relevance to our understanding of evolved organisms. Through evolvability analyses of Weber's law, we find that optimality-equivalent metrics of phenotypic quality need not be selectively

[2]The trait need not be strictly optimal, of course, and exact optimality is unlikely to be found in real organisms. Our use of the term "optimality" is analogous to the concept of a circle in Euclidean geometry. Both are nonexistent limiting abstractions, but each is useful in conversation – verbal or mathematical – whatever degree of approximation one happens to encounter in reality.

equivalent: a phenotype that is optimal with respect to two metrics may be evolvable with respect to one, neither, or both at different rates and with different tolerances for random drift. Further, this requires fewer degrees of freedom than a typical model of selection: many difficult decisions (such as how to model the phenotype, or what is an appropriate choice of "first principles") need not be made, since such decisions have already been made by the optimality analyses themselves, and those decisions should be taken at face value in assessing them relative to one another. We end by examining an existing debate in the literature on Weber's law, and arguing that the disputants may not in fact be in disagreement after all.

1.3 How to model an arbitrary sensory system?

Given the rarity of phenomena as general as Weber's law, we want to explore it in some depth. Unfortunately, this very generality presents a significant problem: how on earth are we supposed to model an arbitrary sensory system? Any model that is sufficiently general to represent an arbitrary sensory system will surely be impoverished to the point of uselessness. Or will it? Faced with a problem for which no good solution is possible, the best thing to do is: build a bad solution. We don't need to model every detail of a sensory system; what we want is to explore Weber's law. From this point of view, we only need a minimal, skeletal model of exactly those features about which Weber's law speaks: the *magnitude* of an input stimulus, and the *just-noticeable difference* between two input stimuli. Indeed, any mathematical analysis or computer simulation of Weber's law *as such* – as opposed to its appearance in a particular sensory system – must be of this skeletal nature, given its appearance across such a wide variety of species and sensory domains.

Now we have two problems. How should we represent these two features of a sensory

system? At this point, a researcher hoping to explore Weber's law (whether by mathematical analyses or computer simulations) is faced with a choice between a "discrete" model on the one hand, and a "continuous" model on the other. Their apparent (and actual) differences are a common source of confusion, so it is worth a brief examination of each.

1.4 Models that feel discrete

The simplest way of capturing both the magnitude of an input stimulus and the just noticeable differences between two stimuli is in terms of a finite number of points within a continuous interval. Each point has both a magnitude and a set of neighbors, so we have a toy language in which both absolute magnitudes and just-noticeable differences can be discussed. For example, consider an imaginary robot visual system that represents brightness within the range $[0, 12]$ by allocating one point per unit intensity, except within the range from 5 to 8, where it allocates points twice as densely, giving something like [0, 1, 2, 3, 4, 5, 5.5, 6, 6.5, 7, 7.5, 8, 9, 10, 11, 12]. There are 16 total "symbols" in this set, so each member could be encoded as a 4-bit binary number. The important point for our purposes is not the encoding, but the sense in which this toy model of a sensory system allows one to discuss both magnitude (each point has a size), and just-noticeable differences (each point has neighbors).

Despite the extreme minimalism of this representation – a set of N points in some interval $[a, b]$ – there are already N dimensions along which the representation can vary, since each point can vary independently of all the others along the continuous interval between its left and right neighbor. Further, the size of each of those N dimensions is not fixed from the start, but varies as a point's left and right neighbor changes position: your left neighbor moves to the right, and suddenly you have less wiggle room. Avoiding

this complication by allowing each point to move completely independently of the others within the interval $[a, b]$ would seem to simplify the overall geometry of the space, but this apparent simplification comes at a cost: any given set of N points would then correspond to $N!$ different locations in this larger rectangular space, and we no longer have the benefit of a one-to-one relationship between toy sensory systems and locations in the simpler space of possibilities. Despite the almost childish simplicity of the model above, we have already got a non-trivial space of possibilities (essentially an N dimensional tetrahedron). We mention this to illustrate the first issues that a modeler faces when attempting to model Weber's law, and the reason for the existence of such models in the literature (Portugal & Svaiter, 2010). Choosing such a model is both imperfect and entirely understandable.

Further, it is not as limited as it seems as first glance. We can think of such models in two ways, which can be informally described as "dots" and "slots." In the "dots" model, our toy robot might map the stimulus 4.6 to the "dot" 5, its nearest neighbor. Or perhaps, since our robot can't tell the difference between 4.4 and 4.6 (by definition), it might just have to always map downward (mapping 4.6 to 4), or upward (mapping 4.4 to 5), or choose randomly between the left and right neighbors of a stimulus. These are different variants of what we can call "the dots model." In the "slots" model however, we can instead think of our robot as mapping the stimulus 4.6 to the "slot" $[4, 5]$, and essentially saying "the stimulus was somewhere in here." The slots model appears more "continuous," though its toy sensory systems are specified in the same way as the "dots" model: by a discrete set of points. This terminology offers a concise way of describing certain models in the existing literature. For instance, Portugal and Svaiter (2010) examine a discrete model with the "dots" interpretation.

To compare different toy sensory systems, we first need to decide which subsets can be fairly compared with one another. Fortunately, there is a natural choice: only com-

pare toy sensory systems with equal numbers of "dots." A model of how to best spend one's income (for any meaning of "best") should hold constant the total income when comparing strategies, so as not to come to the unhelpful conclusion that it is better to spend frivolously as a billionaire than it is to save for retirement on minimum wage. Similarly, the only way to meaningfully ask how Weber's law relates to various alternatives is to hold constant a toy sensory system's overall "wealth," and examine how different methods of allocating that wealth vary with respect to a given measure of fitness.

1.5 Models that feel continuous

Can we improve on the above by using a genuinely continuous model (i.e., one not specified by a discrete set of values)? Perhaps. For example, we could model a minimal sensory system not in terms of N points within some interval $[a, b]$, but in terms of a continuous function whose height in different locations represents the relative *density* of those (imaginary) points at different locations within $[a, b]$, without needing to "commit" to a particular set. Returning to our toy robot sensory system above, the "density" would be a step function whose height between 5 and 8 is twice as large as it is elsewhere in $[0, 12]$. This conveys the robot's emphasis on the middle of the represented range, without specifying any particular numbers. Further, it allows us to discuss jnds, at least in a relative sense: since the density function's height between 5 and 8 is *twice* as large as it is elsewhere, the jnds in that region are *half* as large as those outside. This faithfully reflects the properties of the set of numbers above, without having to decide on a particular set. However, we can recover specific numbers by imposing a constraint on the density function's integral. For example, by adding the assumption "the total area beneath the function is 15," we can translate the "continuous" model into a "discrete" one by solving for the locations x_k where the function's integral from 0 to x_k is an integer.

1.6 Think of these as dialects

The conceptual background above should be enough to allow the reader to understand a large part of the literature on optimality analyses of Weber's law, and why it looks the way it does. Some authors use models that appear "discrete" (Portugal & Svaiter, 2010), while others use models that appear "continuous" (Piantadosi, in press).

These are not so much different models as they are different dialects in which the same concepts can be discussed. The "discrete" models are not inherently discrete, since we had no trouble translating our original example into a density function. Nor are the "continuous" models inherently continuous, since we can easily recover a discrete representation by partitioning the density function's integral into N equal chunks, for any N. This is a common source of confusion, but it cannot be avoided by simply choosing a different model. Any paper attempting to examine Weber's law mathematically needs *some* toy model of an arbitrary sensory system, in order to discuss those features of sensory systems about which Weber's law speaks: the magnitude of a stimulus, and the system's jnd at that magnitude. The "continuous" and "discrete" models one finds in the existing literature do not necessarily differ in their assumptions about the underlying sensory systems. They are simply accents in a domain-specific vocabulary for discussing the pieces of an arbitrary sensory system that are relevant to Weber's law: the magnitude of a sensory stimulus, and the "noise" in the sensory system at that magnitude.

The benefit of models in the continuous dialect is their elegance, while the benefit of models in the discrete dialect is that they make the space of toy sensory systems finite-dimensional. A finite-dimensional model space is not at all necessary, but it both (a) facilitates computer simulations, and (b) makes the mathematics more accessible, since it allows us to use standard multivariable calculus, rather than calculus of variations. Calculus of variations is perhaps a more elegant and powerful framework for discussing

optimality when the candidate optima are entire functions rather than discrete sets of points, but explaining the necessary background would at least double the length of this paper, as well as making the evolvability analyses that follow much more opaque than is necessary. For an attempt at an intuitive exposition of calculus of variations using a minimal amount of mathematical formalism, see the final chapter of Wilkes (in press).

1.7 Scale invariance

In what follows, the term "scale invariance" will be used synonymously with the term "Weber's law," while we will avoid the terms "Fechner's law" and "logarithmic mapping" in order to avoid implicitly taking sides in an ongoing debate, which we discuss explicitly in the final section. It is commonly recognized that Weber's law is scale invariant, but to justify using the terms synonymously, it should be the case that scale invariance implies Weber's law. This simple fact can be demonstrated quickly, while also illustrating how one can move back and forth between the continuous and discrete dialects. This is not so much a result as an attempt to convey to the reader both the conceptual origins of the model on which the following evolvability analyses are based, and a detailed understanding of the assumptions on which such models do and do not rest.

Toward that end, suppose f is a density function with the meaning discussed above, and assume that f is "scale invariant." What does this mean? Well, the area under f within any small interval represents the number of imaginary dots in that interval (or equivalently, the number of actual dots placed there by any of f's partner models in the discrete dialect, as in our robot example above). So what would it mean to assume that f is "scale invariant"? This only means that the area (number of imaginary dots) under f within any interval should not change when we change scale. What is "changing scale"? Changing units, like switching from feet to meters, while still describing the same thing.

Indeed, if there happened to be any general "laws" that applied to multiple sensory systems, then at very least they should be phrased in a manner that is independent of any particular system of units. Otherwise, it is not entirely clear what it would mean to say that an empirical observation about the visual system is "the same" as an empirical observation about the auditory system. Philosophy aside, if f is scale invariant, then changing units from x to $X = ax$ for any $a > 0$ should not change the area (number of imaginary dots) in any small sub-interval. That is, the quantity (height of f)$\cdot$(tiny width) in one system of units should be the same as in another. To say this in abbreviated form:

$$f(x)dx = f(X)dX = f(ax)d(ax)$$

for all x, or equivalently, $f(x) = af(ax)$. Our goal is to solve this equation, not for an unknown variable as in elementary algebra, but for the unknown function f. Following the standard method,[3] this functional equation can be reduced to a differential equation by differentiating both sides with respect to a and setting $a = 1$. Doing so gives $0 = xf'(x) + f(x)$, which can be rearranged to give $\frac{df}{f} = -\frac{dx}{x}$. Integrating both sides, we

[3]In mathematical fields, the closest thing to publishing one's raw data is the act of explaining the thought processes behind a derivation, rather than simply presenting the derivation itself. This footnote is an attempt to provide the "raw data" of the derivation that follows. We have just encountered a "functional equation" – an equation involving an unknown function. Functional equations are tricky to deal with in general, and their solutions often appear to involve a series of clever but unmotivated tricks. The mathematically inclined reader will recognize this phenomenon from basic calculus, where one finds that derivatives can generally be computed by applying a small set of methods in a relatively automatic fashion, while integrals seem to demand a bit more artful cleverness. In this respect, generic functional equations are more like integrals than derivatives. Naturally, a common inclination when faced with a functional equation is to simply differentiate it, in the hopes that it might be reduced to a differential equation, since such equations are more well studied, and generally more susceptible to simple algorithmic methods. This vague mathematical inclination turns out to work a surprising amount of the time, but even when it leads to a solution, it can leave the reader with the feeling that some magical sleight of hand occurred in the derivation they just read. In the derivation that follows, the trick of differentiating with respect to a and setting it equal to 1 is not some fancy advanced method; it is simply the tried and true mathematical practice of following the vague intuition above, in the hopes of finding any legal sequence of steps that reduces an unfamiliar problem to a more familiar one. Such methods can be found in the majority of academic papers involving any degree of mathematics, but there seems to be an unwritten rule that they should never be discussed explicitly, though failing to do so serves no function other than making one's results more difficult to understand for readers not formally trained in mathematics.

obtain $f(x) = c/x$, for some unknown constant c.

In the example of our imaginary robot above, we mentioned that the "continuous" and "discrete" models found in the existing literature do not necessarily imply different hypotheses about the underlying sensory systems, but are rather two equally valid mathematical dialects in which the same Weber-relevant concepts can be discussed. We can now demonstrate how translating from one dialect to another can often make non-obvious properties more intuitively apparent. Above, we started from a qualitative requirement of "scale invariance," and after a bit of mathematics we obtained a particular density function c/x. Weber's law is implicit in this density function, but it may not be obvious how. The unknown constant c is the mathematics telling us that we did not yet decide how many "dots" the toy sensory system should have. Let's make such a decision, but leave it as an unspecified number so as not to lose any generality. While remaining agnostic about the total number of dots, we can build a discrete model from our density function, analogous to a histogram whose bins all contain an equal number of dots. Let's do that. The symbols x_k for $1 \leq k \leq n$ will stand for the locations where one histogram bin stops and another begins. Each contains the same number of dots, say N. This gives:

$$\int_{x_k}^{x_{k+1}} \left(\frac{c}{x}\right) dx \;=\; N$$

for all $1 \leq k \leq n$. This means that $\ln\left(\frac{x_{k+1}}{x_k}\right) = \frac{N}{c}$, which is constant for all k. Equivalently, this tells us that $x_{k+1} \propto x_k$, with the same constant of proportionality for all k. Since x_{k+1} is larger than x_k by assumption, the constant of proportionality must be greater than 1, so we can write $x_{k+1} = (1+\epsilon)x_k$ with no loss of generality. This gives us a discrete (though arbitrarily precise) representation of our continuous density function, and more readily reveals to our intuition one of its particularly interesting properties. Since each point is just $(1+\epsilon)$ times the previous point, an arbitrary point can be written

$x_k = x_0(1+\epsilon)^k$. It is now trivial to compute the jnd between two such points in this toy sensory system, as follows:

$$\Delta x_k \equiv x_{k+1} - x_k = x_0(1+\epsilon)^{k+1} - x_0(1+\epsilon)^k = x_0(1+\epsilon)^k(1+\epsilon-1) = \epsilon x_0(1+\epsilon)^k = \epsilon x_k$$

So $\Delta x_k = \epsilon x_k$ for all k, and the resulting toy sensory system obeys Weber's law. The purpose of the above discussion is not to present a result, (though the reader is welcome to interpret it as one) but rather to lubricate the substantive results that follow by killing three conceptual birds with one stone. First, it is commonly said that Weber's law is scale invariant, and the above discussion shows the converse: scale invariance implies Weber's law. Second, the apparent differences between discrete and continuous models may cause apparent disagreement between individuals who may not in fact disagree. Any model in this literature faces the problem of representing an arbitrary sensory system, and any such model can only hope to represent the subset of features that are relevant to a discussion of Weber's law. Such models are necessarily skeletal. Whether the discussion takes place in a dialect that feels "discrete" or "continuous" is not particularly important. Both are simply quantitative ways of discussing the relationship between the magnitude of a stimulus, and the "noise" in a sensory system at that magnitude. Neither dialect makes stronger or weaker assumptions about the internal architecture of those sensory systems, since both styles of model are bleached of all details that a sensory system might possess, except the two relevant to Weber's law. Further, we can easily translate between the two dialects, and sometimes a result can be seen more clearly in one dialect than in the other, as we saw above.

Finally, the discussion above was an attempt to familiarize the reader with the conceptual "raw data" of Weber's law, by demonstrating some of the paths of reasoning by which authors in this literature might arrive at their different models in the first place, rather than simply presenting a particular model ex nihilo, and discussing the results

that follow from it. This has the unfortunate effect of adding a bit more verbosity (and a bit less formality) to the surrounding discussion, but it is the closest thing one has in mathematical work to the scientifically desirable practice of publishing one's raw data.

Chapter 2

Optimality

In the section that follows, we will examine various standards with respect to which Weber's law is "optimal." An earlier version of this manuscript consisted primarily of such optimality analyses. In what follows, we will examine existing analyses in this vein, and offer some new ones. However, we do not to present them as new results, but rather as pedagogical demonstrations to frame and clarify the more substantive results of the evolvability analyses that follow. The exploration of evolvability is both an attempt to further understand Weber's law, as well as a broader attempt to provide more substance to the ways in which optimality analyses can be evaluated. Before doing so, however, we need some optimality analyses to use as raw material. This is the task to which we now turn.

2.1 Optimizing worst-case relative error

Portugal and Svaiter (2010) demonstrated that Weber's law is optimal by the standard of minimax relative error. In the terminology discussed above, they began from a discrete model with the "dots" interpretation. They then demonstrated that model

sensory systems obeying Weber's law achieve a minimum value of the maximum relative error, compared to other sensory systems with the same number of "dots." Portugal and Svaiter's result offers a simple and intuitive way to understand Weber's law, but unfortunately, the result is demonstrated along the lines of a formal proof, with considerably more technical machinery than is needed to arrive at the conclusion. However, this result and its model will play an important role in the evolvability analyses that follow, so it is worth quickly demonstrating the same result with less technical machinery. This is not Portugal and Svaiter's derivation, but the result belongs to them. The maximum relative error of a model sensory system M over the interval $[a, b]$ is

$$E_M = \max_{a \leq x \leq b} RE(x), \quad \text{where} \quad RE(x) = \min_{x_k \in M} \frac{|x_k - x|}{x} \tag{2.1}$$

Since Portugal and Svaiter's model is what we have called a "discrete model in the dots interpretation," any stimulus x is represented by its nearest "dot" in the set M. The above minimum over all the x_k in M is simply a way of telling the mathematics that the model is making this assumption. How can we find the set of sensory systems that minimize the measure E_M? It helps to first express the measure in an equivalent but simpler way. Toward that end, note that any discrete dots model M divides $[a, b]$ into smaller subintervals $I_k = [x_k, x_{k+1}]$, the subintervals between successive dots. For any such subinterval:

$$\max_{x \in I_k} RE(x) = \max_{x \in I_k} \left(\min \left\{ \frac{x - x_k}{x}, \frac{x_{k+1} - x}{x} \right\} \right) \tag{2.2}$$

The meaning of this is simple: Any interval I_k between two dots will contain some "stimuli" that are mapped to its left dot, and others that are mapped to its right dot. The relative error in representing a given stimulus depends on which of these dots the stimulus is mapped to, and the above equation simply expresses this fact explicitly. More simply, it says: What is the worst-case stimulus in any subinterval between two dots? It

is whichever stimulus has the worst representation, where "worst representation" simply means "largest relative error between that stimulus and whichever dot is nearest to it." In each interval between two dots, this "worst-case" occurs at the arithmetic mean of those two dots, i.e., when $x = \bar{x}_k \equiv (1/2)(x_k + x_{k+1})$. Expressing this fact mathematically and simplifying the result, we obtain:

$$\max_{x \in I_k} RE(x) = \frac{\bar{x}_k - x_k}{\bar{x}_k} = \frac{x_{k+1} - x_k}{x_{k+1} + x_k} \tag{2.3}$$

Now, using the facts above, it is simple to determine which toy sensory systems minimize the *global* maximum relative error over all such subintervals I_k. For any fixed number of dots, any perturbation of the set of dots M that decreases the max relative error within one subinterval must increase the max relative error for another. So the toy sensory systems with the smallest overall value of this measure must be the ones for which the maximum relative error is constant in each subinterval I_k. That is:

$$\mathrm{Var}\left(\max_{x \in I_k} RE(x)\right) = 0 \tag{2.4}$$

where the variance is taken over the set of subintervals I_k. To put it more simply, each subinterval has a "peak" (the worst case relative error between its left and right dot), and we can minimize the highest peak by pushing all the peaks down to a constant height. With foresight, we will call this constant $\epsilon/2$. Summarizing the above paragraph, the toy sensory systems that minimize the max relative error satisfy:

$$\max_{x \in I_k} RE(x) = constant \equiv \frac{\epsilon}{2} \tag{2.5}$$

for all k, so by equation (2.3):

$$\frac{x_{k+1} - x_k}{x_{k+1} + x_k} = \frac{\epsilon}{2} \tag{2.6}$$

Finally, defining $\Delta x_k \equiv x_{k+1} - x_k$ as in the above discussion of scale invariance, we find that $\Delta x_k = \epsilon \bar{x}_k$. This is a different sense of Weber's law than we found earlier. The

arithmetic mean on the right side of this equation was the result of mapping model-stimuli to the *nearest* dot. Similarly, carrying out the same derivation as above, but instead choosing to always map stimuli to the next highest or next lowest dot, the results one finds are (respectively) $\Delta x_k = \epsilon x_k$ and $\Delta x_k = \epsilon x_{k+1}$. Each of these expresses a linear scaling of jnds with stimulus magnitudes, but in ways that each differ from one another by less than one jnd. We remain agnostic about which is the "best" representation of Weber's law, since skeletal models of this form should only be taken as a more precise form of verbal argument – a form in which we have the hope of revealing properties that may not have been obvious from verbal argument alone. However, any model should be aware of its limits, and the toy models above should not be taken literally in the fine-grained details of how they differ at sub-jnd scale, since the accidental details of their implementation become visible at the scale of a single jnd.

2.2 Optimizing variance of relative error

Toward the goal of developing principled methods by which optimality analyses can be evaluated, let's create an optimality analysis that is "bad," on purpose. It will still pick out Weber's law as the optimum of *something*, but by design, that something will not appear especially well-motivated. Above we found that minimizing the maximum relative error leads to Weber's law. In this model, each subinterval had a "peak," and Weber's law minimized the highest peak (max relative error) by pushing all the peaks down to the same height.

We want to construct an optimality analysis that is bad on purpose, and we can do so simply by jumping into the above derivation starting at equation (2.4). Generally speaking, optimality analyses begin from "first principles," or at least principles asserted to be more fundamental than the results one ends up demonstrating. We know that

equation (2.4) will get us to Weber's law, so we can construct our bad optimality analysis by simply asserting that the variance of relative error (as defined in equation 2.4) is a fundamentally important quantity that a well-designed sensory system should minimize, and arguing this point in eloquent language and elegant mathematics.

We have therefore picked-out the same sensory systems – the ones obeying Weber's law – by describing them in a different way: they minimize the variance of relative error in the sense defined above. In what follows, we will describe the (non-)result of this section as the minivar derivation, and the previous section's result as the minimax derivation. These two standards of phenotypic quality pick out the same set of sensory systems. As far as optimality analyses are concerned, the two metrics are analytically identical, and one is forced to rely on intuition and the eloquence with which each result is expressed in order to judge them. We explicitly constructed the minivar metric post-hoc, as an example of an unconvincing measure of quality, so intuition is considerably less kind to this new derivation than it is to the original. We now have two optimality analyses with differing intuitive appeal, to be used as raw materials in the evolvability analyses that follow.

2.3 Optimizing integrals of relative error

In the dots model, stimuli are represented as points, so arithmetic is straightforward to perform on the represented values. In the slots model, stimuli are represented as intervals. This model is specified by giving the boundaries *between* the slots. By varying the locations of those boundaries, we obtain different toy sensory systems. In this section, we will present an optimality analysis in the slots model.

Suppose we want to examine the total relative error of a toy sensory system under the "slots" interpretation. There are two general strategies, and we will give a brief

summary of each before proceeding to the mathematics. First, we could penalize each "node" (location where two slots meet) by the integral of relative error from itself to its left neighbor, plus the integral from itself to its right neighbor. This measure includes each subinterval twice, but not redundantly: given any subinterval S (i.e., any "slot"), it includes one term that is the integral (over S) of the relative error *relative to* the left boundary of S, while the other integral is the same quantity relative to the right boundary of S. These are different quantities, and neither is more important a priori. Further, this is a genuine slots-model, since neither the toy sensory systems nor the metric for evaluating them requires any mapping of stimuli to a particular dot. As we will demonstrate shortly, this measure of total relative error gives Weber's law as its optimum.

However, we should also discuss a metric for which the optimum is *not* Weber's law. We can obtain such a metric by modifying the one above as follows. Suppose that we penalize each "node" (location where two slots meet) by two integrals, as above, but this time we integrate only to the arithmetic means of the node and its left and right neighbors. In some ways this appears more motivated than the previous measure. However, the motivation for using the arithmetic mean came from the dots model, where it arose from the idea of mapping a stimulus to whichever dot was closest. That is, the point where a stimulus' closest dot stopped being x_k and started being x_{k+1} was the arithmetic mean of those two dots. In the slots model however, the arithmetic mean is a bit of a kludgey addition, since stimuli are represented by intervals, not numbers. Still, we are free to examine the resulting metric anyway, for purposes of illustration. The optimum of this measure is *not* Weber's law, but a kind of Weber-like hybrid. As we will see shortly, Weber's law is expressible as $x_k = \sqrt{x_{k-1}x_{k+1}}$, and this is the optimum of the first of the two metrics in this section: the one that did not involve the arithmetic mean. This hybrid metric has the optimum $x_k = \sqrt{\bar{x}_k^L \bar{x}_k^R}$, where $\bar{x}_k^L$ is the "left mean,"

(i.e., the mean of x_k and x_{k-1}) while $\bar{x}_k^R$ is the "right mean" (i.e., the mean of x_k and x_{k+1}).

Which of these is the "true" measure of total relative error? We leave that for the reader to decide. Each has its deficits. Leaving that aside, the former specifies exactly the toy sensory systems obeying Weber's law, while the latter gives a visually similar but structurally different set. Toward the goal of developing methods by which different optimality analyses of the *same phenomenon* might be compared, only the former qualifies as a contender in the evolvability competition that follows. We mention the latter only to emphasize that not every reasonable-sounding function of relative error gives Weber's law as its optimum. By any measure, the majority do not.

Having provided the necessary conceptual background, we can back-up the assertions above. The degrees of freedom of a sensory system in the slots model are the locations of the boundaries between its n slots. Given any (smooth) measure of phenotypic quality, the toy sensory systems optimizing that measure will occur when each of its partial derivatives are zero: one partial derivative for each degree of freedom.[1] Translating our verbose English description of the first metric above into abbreviated form, its dependence on the k^{th} boundary is:

$$E_k = \int_{x_{k-1}}^{x_{k+1}} \frac{|x_k - x|}{x} dx$$

This can be broken into two integrals and simplified as follows:

$$E_k = \int_{x_{k-1}}^{x_k} \frac{x_k - x}{x} dx + \int_{x_k}^{x_{k+1}} \frac{x - x_k}{x} dx$$

[1]For readers unfamiliar with the concept: the max and min height of a curve occur at locations where the curve is flat or horizontal (picture a sine wave or a bell curve). Similarly, the top and bottom of an n-dimensional mountain will occur at locations where the mountain is "horizontal" in each of the n directions one can walk on it. The mathematically-inclined reader who can see the exceptions to these general claims can verify that no such corner cases occur in the derivation that follows, so the explanation is sufficient for the limited purpose of this footnote.

$$= \int_{x_{k-1}}^{x_k} \Big(\frac{x_k}{x} - 1\Big)dx + \int_{x_k}^{x_{k+1}} \Big(1 - \frac{x_k}{x}\Big)dx$$

$$= [x_k \ln(x) - x]_{x_{k-1}}^{x_k} + [x - x_k \ln(x)]_{x_k}^{x_{k+1}}$$

$$= 2x_k \ln(x_k) - 2x_k + (x_{k-1} + x_{k+1}) - x_k \ln(x_{k-1}x_{k+1})$$

Now, the optimal sensory systems for this metric can be found by differentiating E_k with respect to x_k (for each k) and setting all the results equal to zero. This gives:

$$0 = \frac{\partial E_k}{\partial x_k} = 2\ln(x_k) - \ln(x_{k-1}x_{k+1})$$

Isolating x_k, we obtain:

$$x_k = \sqrt{x_{k-1}x_{k+1}}$$

This is Weber's law in disguise. To see this more clearly, square the above condition and rearrange it to obtain:

$$\frac{x_k}{x_{k-1}} = \frac{x_{k+1}}{x_k}$$

for all k. This tells us that the optimal sensory systems by this metric satisfy $x_k \propto x_{k+1}$, with the same constant of proportionality for all k. But this is exactly the condition we arrived at in the earlier discussion of scale invariance. Following the same steps outlined there, we find $\Delta x_k = \epsilon x_k$, so the optimal sensory systems by this metric are exactly the ones obeying Weber's law. Having already given nicknames to our other two metrics, minimax and minivar, it is only fair to give a similar 7 character nickname to this new contender as well. In the evolvability analyses that follow (and their supporting code in the supplementary materials), we will call this metric "totalre," for "total relative error." Three is a crowd, as the saying goes, so having collected a small crowd of optimality analyses, we now turn to the question of how they might be compared.

Chapter 3

Evolvability

3.1 Reverse First Principles

An earlier version of this manuscript was titled "Weber's law from first principles" and it consisted entirely of optimality analyses: mathematical demonstrations that Weber's law is optimal with respect to various measures of quality. Such results are simply a mathematical demonstration of the logical relationship between two statements: beginning from some assumption X (for a variety of X), one arrives at the conclusion of Weber's law (or more generally, the target trait of the optimality analysis). Since then, the author has had the opportunity to participate as both reviewer and reviewee in evaluating results of this form, and has experienced the difficulty of this task from both sides. Whatever one's opinion on the value of optimality analyses, the interdisciplinary nature of mathematical/cognitive psychology leads to a social problem in presenting them. The problem is illustrated by the following common pattern:

> Researcher A has an optimality analysis of some phenomenon P, showing that P arises from optimizing measure M. S/he says that previous work by researcher B optimized measure N, but by optimizing N, the conclusion P

> was implicit in the premises. Then researcher A proceeds to optimize M, and shows that P emerges as a result. Was the conclusion P implicit in A's premises, or wasn't it?

What is never discussed is exactly what it means to say that a conclusion is implicit in a set of premises. Any mathematical result has to be equivalent to or strictly weaker than the assumptions on which it is based; deductive reasoning can only move downhill or sideways, not up. This fact applies equally to both of our hypothetical researchers above. If either finds that their conclusion is not implicit in their premises, then the result is wrong. What we learn from the whole exercise is the fact that the conclusion *is* implicit in those premises – a fact that would not have been obvious to someone who had not studied the issue in detail. To the extent that Researcher C is interested in phenomenon P, and does not see the implication immediately, C has the potential to learn something of value from either result. Both A's conclusions and B's were implicit in the premises from which they began, and both results are potentially valuable to C, or to one another. The fact that all mathematical results are "trivial" in this fundamental sense does not mean that they appear trivial to our primate brains, or that nothing can be learned from them. Mathematics pervades the sciences, not because it has a magical ability to demonstrate conclusions from nothing, but because it has the ability to clarify the relationships between different claims, however similar or different those claims intuitively appear.[1]

Optimality analyses provide a valuable clarification of the relationships between ideas, but in interdisciplinary fields such as ours, they suffer from a brutal catch-22: (a) optimality analyses are inherently mathematical; (b) mathematical results are inherently

[1]The mathematician Jerry Bona is quoted as saying "The Axiom of Choice is obviously true, the well-ordering principle is obviously false, and who can tell about Zorn's lemma?" The quote is a joke; he knew the three statements had been proved equivalent, but their content appears so different that the equivalence is not obvious. Until one has studied the issue in detail.

"trivial," in the fundamental (but unimportant) sense above; (c) the perceived triviality of any mathematical result increases in proportion to the clarity with which it is expressed; (d) as the perceived triviality of a result increases, its likelihood of being published decreases; (e) the relationships described in (c) and (d) are unbounded, as a result of item (b).

In short, our lack of any principled methods of evaluating optimality analyses other than their perceived non-triviality has tipped the incentives toward obfuscation rather than clarification. Through the best of intentions, we have been penalizing our colleagues in proportion to the clarity with which their results are expressed. No one is to blame for this, but it is important that we recognize it openly and honestly as an incentive that has influenced mathematical work in our field. Only by doing so can we hope to align our field's incentives with the proper use of mathematics: as a tool of clarification, rather than obfuscation.

So, are there any principled standards by which different optimality analyses might be evaluated? Can we develop methods that usefully supplement intuition, without requiring us to discard intuition entirely? How on earth are we supposed to decide what constitutes an appropriate choice of "first principles"? This is an impossible problem, but it need not be solved completely. Any constraints would be better than nothing.

The field of logic known as "reverse mathematics" has faced a similar problem. What are the proper axioms for the foundations of mathematics? This too is an impossible problem, seemingly solvable only by vague intuition and verbal argument between university-dwelling primates. The field's solution is delightfully tricky. Rather than asking which first principles (axioms) are the most self-evident (a recipe for unending arguments) simply leave aside the issue of self-evidence, and study first principles in reverse. At a sufficiently low level of abstraction, the distinction between "axioms" and "theorems" disappears: they are all just well-formed sentences in a formal language.

First principles (axioms) can therefore be studied and categorized just like any other mathematical object: by studying which "second principles" (theorems) they imply, and much more strangely, which second principles imply them – the clever trick from which the field derives its name. Reverse mathematics evaluates first principles, not a priori, but by using second principles as fuel.[2]

In experimental psychology, our objects of study are evolved organisms. Optimality analyses are one way of addressing "why" questions about such organisms. An optimality analysis consists of:

1. Some feature of a phenotype.

2. A proposed metric of quality.

3. A demonstration that 1 is optimal with respect to 2.

The 1st item is a choice of research topic, the 2nd is a choice of "first principles," and the 3rd is the result. Science is large, and there's no point constraining item 1. The dictates of correct mathematics constrain item 3. Item 2, however, is large source of wiggle-room for the researcher, but optimality analyses are valuable nonetheless. How can different choices of this metric be judged? Rather than debate over the most self-evident choice of 2 (a recipe for unending arguments), suppose we simply take such choices at face value, and examine their strength "empirically," by some independent standard. Since our objects of study are evolved organisms, one natural standard suggests itself: a trait (e.g., Weber's law) may be optimal by some measure M, but if that trait does not or

[2]This is a painfully incomplete description of one of the most fascinating (and undeservedly obscure) areas of mathematical logic. I will resist the temptation to ride off on a digression unrelated to the topic of this paper, but one detail is too interesting to omit. Perhaps the field's most startling lesson is that, of the well-known theorems of mathematics thus far examined, the majority are provably equivalent to one of five statements, while most theorems one learns below the graduate level are equivalent to one of the weakest three. This is startling, to say the least. The interested reader can refer to Simpson (2009) for the gory details.

cannot evolve under selection for M, then its optimality with respect to M is of little relevance to our understanding of evolved organisms.

This is not a silver bullet, and it does not tell us what the "right" choice is. However, it allows optimality analyses and the "first principles" on which they are based to be evaluated in reverse: not by their a priori appeal, but by independent standards – standards treated as logically prior to the first principles themselves, not in some objective sense, but within the limited context of evaluating those principles relative to one another. However imperfect such standards may be, it is worth searching for them, given the central importance of optimality analyses (and engineering analyses more generally) as a means of addressing "why" questions in biology. Further, the goal is constructive, not destructive: it is an attempt to improve our methods of evaluating these results, not an argument that existing results are wrong.

So, we have multiple optimality analyses. Can we meaningfully assess their strength relative to one another? What, if anything, would we learn from such an exercise? Who knows? Let's see.

3.2 Selection for phenotypes

Modeling selection from scratch presents a researcher with many degrees of freedom. How should the phenotype be modeled? What is the space of possible phenotypes? For which metric should the simulation select? Fortunately, none of these degrees of freedom arise when evolvability is used as a method of assessing optimality analyses themselves. Each optimality analysis has made the majority of such decisions individually. This is a general property of assessing different candidate solutions by a common standard. In building a car, there are similarly many degrees of freedom. However, in assessing the speed of cars relative to one another, one only needs to provide a race track, and

assure that the same race track is used for the assessment of each. To be fair, a car manufacturer might argue that a particular choice of race track biased the results against his particular model, so a fair test should involve multiple competitions on as many different courses as possible, in rain or shine, under both quantitative and qualitative variations in unimportant details.

This is the strategy we have pursued in the evolvability analyses of this section. The simulations are written in Python, and their full source code is available in the supplementary materials, accompanied by instructions for running them oneself. An attempt was made to adhere to the following design principle: faced with an unclear decision of how to implement some detail of mutation or selection, do not choose either. Rather, implement both alternatives, and provide a configurable option to the user. A full list of these options can be produced by running `./selection --help` from the source directory.

This strategy leads to a rather large set of options. Varying each individually would lead to an unhelpfully large set of figures, so we will pursue the strategy of examining each of our three metrics under two sets of parameters, which we will call the "most flexible" and "least flexible" set. The meaning of these terms is as follows. Many configuration options arose from unclear choices of how to implement mutation. Faced with any such choice, at least two alternatives were implemented. For example: suppose each individual phenotype is specified by 20 dots, the "litter size" in each generation is 32, and the probability of a mutation (in a given dot, of a given offspring, in a given generation) is 1/10. Each of these numbers is a configurable parameter, but even having made these choices, there are a number of ways in which mutations could affect the phenotype. Four such choices are listed below, under the names given to their corresponding command-line options:

1. `allow-crossing`: Our phenotypes are specified by a set of "dots" in a continuous interval. Suppose some phenotype contains 20 dots, two of which are located at 41.6 and 42.0. A mutation then occurs in the latter, and the magnitude of that mutation is drawn from a normal distribution. Suppose the value we draw is -0.5. Should the dots be allowed to cross? That is, should the new locations of these two dots be 41.5 and 41.6, or should they be 41.6 and $41.6 + tiny$, for some fixed value of $tiny$ that represents the minimum distance between any two dots? The former possibility belongs to what we will call the "most flexible" option set, while the latter belongs to the "least flexible." If the user does not specify this option, it defaults to the most flexible.[3]

2. `allow-drift`: This parameter determines how to compare phenotypes with equal fitness. In selecting for (say) the minimax metric, mutations that effect the phenotype without affecting its worst case relative error will have the same fitness as measured by the minimax metric. In case of a tie, this parameter determines whether to prefer the status quo, or instead select randomly among the tied phenotypes. The latter option belongs to the "most flexible" option set, since it makes a larger number of unpredictable choices than the alternative.

3. `allow-jitter`: This parameter specifies whether small mutation-independent jitter should be injected into phenotypes. In a phenotype with 20 dots and a mutation probability of 1/10 per dot per generation, suppose 2 dots receive mutations in a given generation. If this option is set to "no," then the phenotype will differ from

[3]Though these option sets are varied in the results presented below, we face the additional goal of helping the user by providing sensible defaults. As such, if the user fails to specify an option, it will generally default to the "most flexible" value, if such a determination makes sense for the option in question. This choice was made so that however variable the results appear when running with all of the defaults, the results will generally be *more predictable* when specifying a larger number of options. The goal is that the simulations should appear in their most vulnerable state when a user runs our code without changing the defaults. The question of which parameter values are more "flexible" is ultimately a judgment call, but this principle should help the reader to understand why various choices were made.

its parent only in those 2 dots. If this option is set to "yes," then in addition to those 2 mutations, each dot in the child will receive a small amount of jitter, whose value is either $-tiny$, 0, or $+tiny$, for a value of $tiny$ that is small relative to the mean effect of a mutation. The value of "yes" for this option belongs to the "more flexible" option set. In a sense, this parameter can be thought of as determining the phenotypic similarity of identical twins.

4. `init`: Specifies whether the initial (1st generation) phenotype should be chosen by randomly allocating dots, or by placing them in a fixed, uniform spacing. Since each region within the continuous interval is treated the same by either rule, either rule leads to an initial phenotype that is very far from Weber's law. This can be seen most clearly in the time-lapse videos included in the supplementary materials. This option can be set to either `uniform` or `random`, and the latter belongs to the "most flexible" option set.

In each generation we choose one winner from which to breed. Doing otherwise would not change the fitness assigned to any given phenotype by any given metric, so it is not likely to change the performance of different metrics *relative to* one another. However, it would slow down the overall process of selection.[4] This is a helpful choice for our purposes, since the goal is not to simulate every detail of natural selection in the wild, but to examine the evolvability of Weber's law relative to different metrics, each of which it optimizes individually. The following graphs demonstrate the dynamics of selection for each of these metrics, under both option sets described above.

[4]Also on the topic of speed, an interesting note: The code has an option called `parallelism`, which can be set to either of two parallelism implementations, or `serial`, a third option that only uses one CPU. Generating and scoring multiple children in parallel dramatically sped-up selection in early versions of the code. However, after seeking out and optimizing various performance bottlenecks, the code now runs quickest *without* parallelism. This is not an uncommon phenomenon in programming, but it is rather surprising when one first encounters it. To say the least, the circumstances under which parallelism improves an algorithm are somewhat more limited than we often realize.

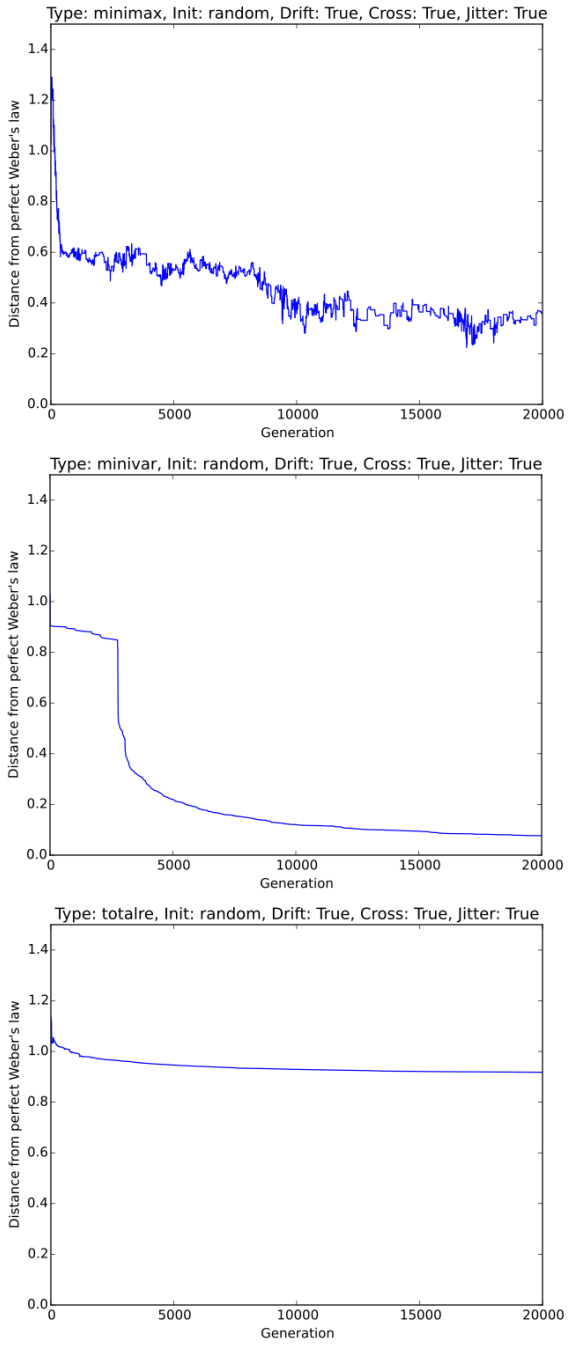

Figure 3.1: The behavior of our three metrics under selection, relative to the "most flexible" option set. Each plot shows the Euclidean distance from Weber's law over 20,000 generations. Weber's law is the optimum of each of these metrics, but it does not evolve equally with respect to each.

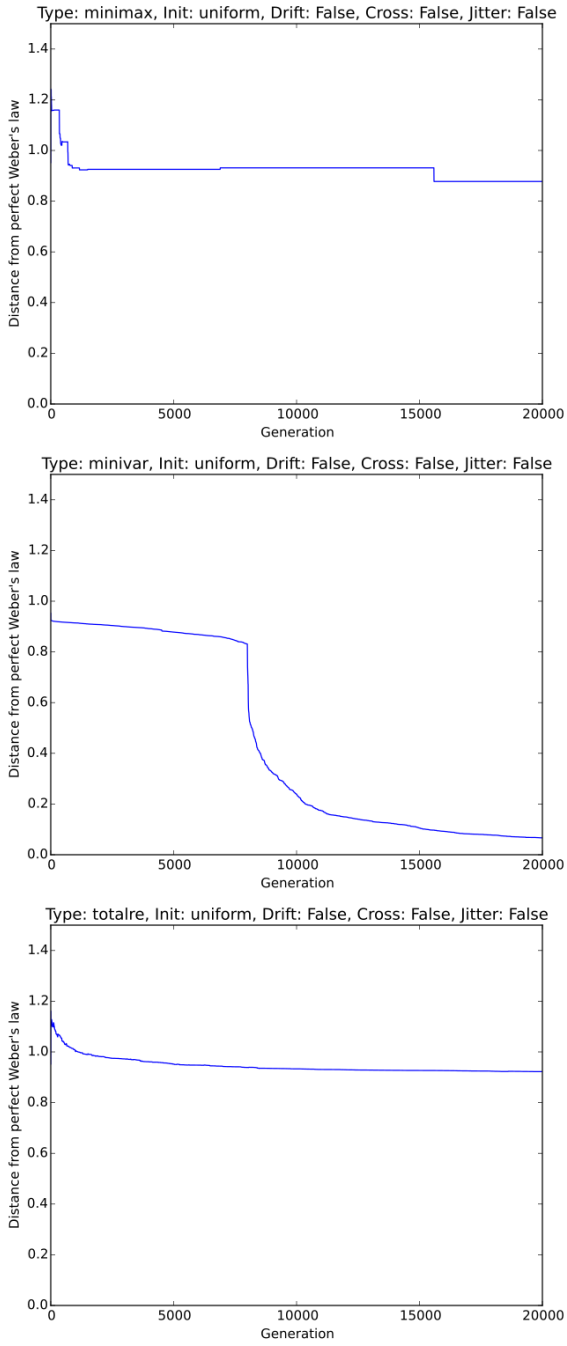

Figure 3.2: The behavior of our three Weber-metrics under selection, relative to the "least flexible" option set. As in the most flexible option set, the minivar metric is considerably more capable of approaching its own optimum. Notice that the sudden drop in minivar is not a macromutation; minivar shows the same pattern of behavior in figure 3.1.

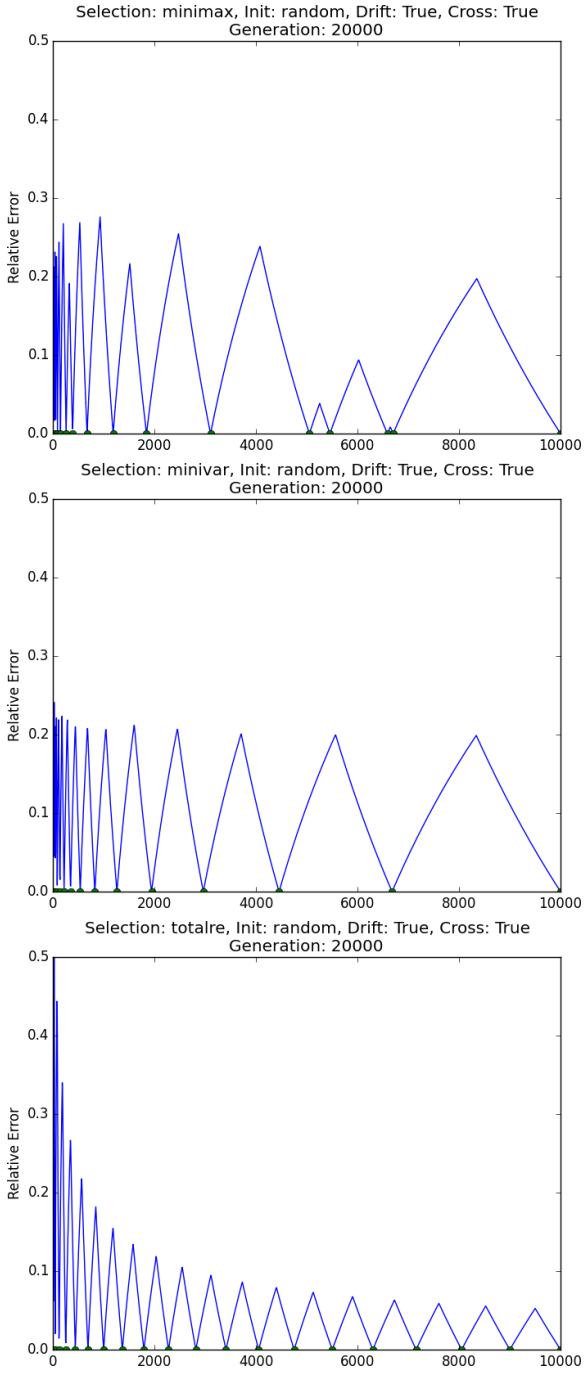

Figure 3.3: Final phenotypes for each metric, under the "most flexible" option set, for the runs pictured in figure 3.1. Each metric's deviations from exact Weber's law take a different form. (1) Minimax focuses only on the highest peak, so as it approaches Weber's law, one observes large deviations in unpredictable locations throughout the sensory system's dynamic range. (2) Minivar's deviations from Weber's law are minimal. (3) Totalre consistently shows extreme deviations from Weber's law for small stimuli, but remains approximately Weber-like for larger stimuli.

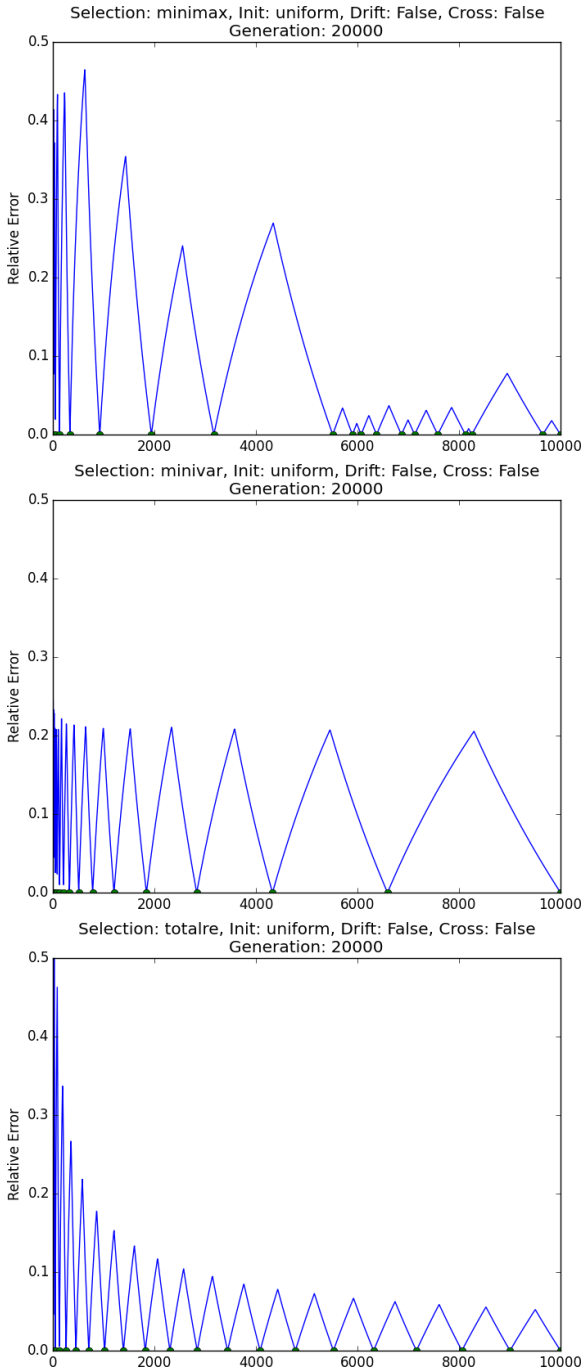

Figure 3.4: Final phenotypes for each metric, under the "least flexible" option set, for the runs pictured in figure 3.2. The final phenotypes for each metric are similar to figure 3.3, but the deviations observed in minimax are more pronounced. While the optimum of each metric is Weber's law, the direction in which each metric approaches "exact Weber" shows deviations from it that are qualitatively consistent between the most and least flexible option sets.

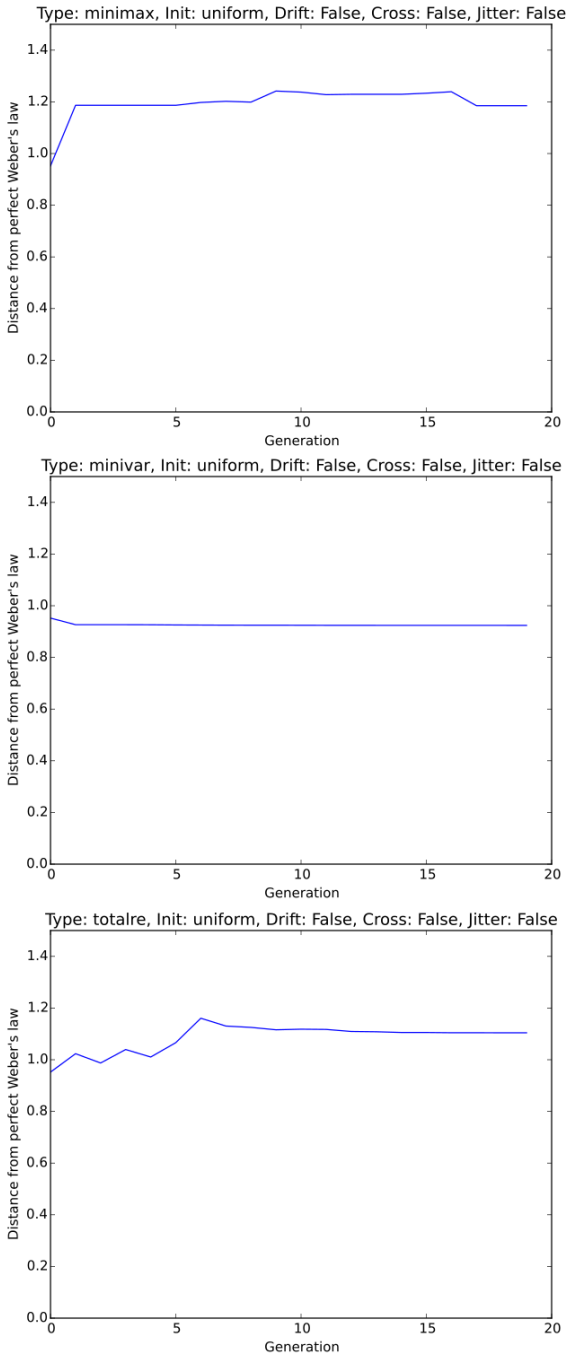

Figure 3.5: Even when comparing identical initial phenotypes, there is the appearance of an unequal (thus, unfair) starting point between the three metrics, (e.g., in figures 3.1 and 3.2). This is caused by the different behavior of each metric in the first few generations. These images show the first 20 generations of the runs from figure 3.2. Note (a) the equal starting points and (b) the different initial behavior. The phenotypic changes that cause this behavior for each metric can be seen more readily in the time-lapse videos available in the supplemental materials.

In each case, we observe behavior that may not have been obvious from the optimality analysis alone, but which makes perfect sense after the fact, and serves to illuminate each optimality analysis individually. First, consider the minimax metric. The minimax metric evolves erratically and slowly at one extreme, and fails to approach its own optimum at the other. In both cases, this results from its focus on a single dimension of phenotypic variation: the *worst-case* relative error. Any mutation that *would* push minimax closer to its own optimum is entirely invisible to it, *unless* that variation also happens to improve the worst-case performance of the sensory system.

In figure 3.2, minimax shows a kind of downward-staircase behavior. The reason is as follows: almost immediately, the minimax metric finds a series of mutations that split its tallest peak into two or more shorter peaks. This improves the phenotype by its own standard, while leading it further away from any phenotypes obeying Weber's law (shown in figure 3.5). Having gone slightly down hill, the phenotype then promptly gets stuck in a local minimum, like a clumsily-tossed bowling ball, and never reaches its own global minimum: Weber's law. It can only jump out of these local minima through a "macromutation." This is the downward staircase phenomenon shown by minimax in figure 3.2. The minimax metric can only improve by a long-jump from the local minimum in which it is stuck, to another far away location that happens to be lower. As figure 3.2 shows, this occurs rather infrequently, and with minimal effects. The minimax metric thus performs fairly poorly under selection, and its behavior is not particularly robust to changes in the parameters of selection.[5]

The totalre metric (our integrated relative error measure from earlier) performs poorly as well, but in a different way. Though the ability of totalre to reach its own optimum

[5]To be clear, this is not intended as a criticism of Portugal and Svaiter's (2010) result, or their strategy of model-building. In fact, our own derivation of the same result above was presented in the original version of this manuscript, before learning of Portugal and Svaiter's paper. As such, the clumsy performance of the minimax metric under selection is just as much a criticism of myself as it is of them.

is incredibly poor (by the standard of figures 3.1 and 3.2), it carries out its failure in a more smooth manner than minimax (visible in figures 3.3 and 3.4). This is because totalre focuses its attention on *all* dimensions of phenotypic variation, while minimax focuses only on a single dimension. Minimax therefore fails to capture many of the mutations that would have led it toward its own optimum. In contrast, totalre generally captures all such mutations, but it suffers from a deficit of its own. As a phenotype approaches Weber's law under selection for totalre, selection becomes *much* weaker in the lower part of the sensory system's dynamic range. Why? It is a simple property of the N-dimensional "mountain" defined by totalre, though an analysis of the mountain's peak in our discussion of optimality failed to reveal this simple property to our intuition. Totalre is based on *area*, and as one approaches Weber's law, the areas at the low end of the dynamic range become thinner and thinner. But as they become thinner, they contribute less to the totalre metric, so the selection pressure acting on those thin peaks becomes weaker, causing them to become *taller*, not by selection for taller peaks, but because selection increasingly begins to neglect them. Nonetheless, totalre arrives at a phenotype that is somewhat Weber-like in the higher parts of its dynamic range. Of our three metrics, totalre is least competent at evolving Weber-like behavior for lower stimulus magnitudes, while minimax is least competent at evolving Weber-like behavior over any consistent subset of its dynamic range. This can be observed most clearly in the time-lapse videos available in the supplemental materials.

So, having examined our three metrics at these two extremes, what have we learned? Each of our combatants behaved differently, but the clear winner at both extremes was the uncultured underdog: minivar – the one explicitly constructed as an example of a *bad* optimality analysis. The minivar derivation was far too short to deserve its own paper, and it derived Weber's law from a metric that appeared entirely ad-hoc and unmotivated to our intuition. However, as we have just seen, the unmotivated metric of

this unconvincing optimality analysis is miles ahead of the minimax and totalre metrics when it comes to selection, both in its ability to reach its own optimum (the sensory systems obeying Weber's law), and in its ability to smoothly take advantage of any variation in the phenotype that happens to push it closer to that optimum.

The point is not to suggest that the minivar optimality analysis is somehow "better" than the minimax optimality analysis, or any other. To the contrary, the author finds the minivar derivation extremely unsatisfying and inelegant. But "unsatisfying" and "inelegant" are the vocabulary of the primate brain; concepts natural selection does not understand. However, leaving aside the minivar *derivation* – the unsophisticated clothing in which the minivar metric was dressed – the metric itself has an elegance of its own. We may not have expected it to win, but hindsight is 20-20, and it is not difficult to see why it did. It is a detail we already mentioned above.

None of our metrics can predict which mutations will occur, but the minivar metric is superior to the competition in that it can smoothly and efficiently take advantage of *any* mutation that happens to push it closer to its own optimum. The other two metrics can be given a lengthier and more elegant mathematical spin, but unlike minivar – the less cultivated underdog – they are far less capable of reaching their own optima under selection. Why does this matter?

A "metric" is simply a concise description of a possible selection pressure. Natural selection does not select for relative error minimization, or inference under uncertainty, or even vision, but for a single metric: gene replication. Mathematically described metrics or verbally described selection pressures are simply helpful ways of summarizing engineering problems that sometimes overlap with selection's single overarching goal of gene replication. But because this process does not literally select for the specific metrics we use to describe it, the metrics that *do* best describe it (among those with a fixed optimum) are arguably the ones capable of capturing any existing variation that pushes

them closer to their own optimum. This is at least one intuition-independent standard by which an optimality analysis may be judged. Though its accompanying derivation is least impressive academically, the minivar metric is superior to the others in this respect. The lesson is simple and unprofound after the fact, but easy to forget in our everyday thinking about selection, whether mathematics or verbal argument is our preferred language for doing so.

3.3 Selection for embryology

In scientific explorations of any phenomenon – empirical or theoretical – one must face the constant fact that one's conclusions (and the premises from which they were derived) were at every moment subject to all the ugly laws of primatology. This cannot help but make one doubt one's own results. However convincing or unconvincing the reader finds the results and arguments above, s/he should worry that they were simply the result of one such primate – the author – attempting to make a convincing point to his colleagues. Were that the case, the author would be the least qualified individual to make this judgment. As such, having found ourselves in the previous section attempting to make a convincing argument, we feel obligated thereafter to undercut that argument as much as possible, exploring more exotic ways of thinking about selection until we find one that breaks every intuition we built-up from exploring the models above, and presenting the results of this exploratory process whether they appear to reinforce the above conclusions, or refute them, or teach us an entirely unrelated but equally interesting lesson.

Toward that end, we will briefly build and examine an entirely different and rather exotic model of selection – one unlike the above in as many ways as possible. As scientists, we cannot step outside the matrix of the primate brain, but we can do our best, by

purposefully breaking our own arguments and results as soon as this matrix generates them. The shattered pieces of such arguments are of greater scientific value than the original intact edifice, since those pieces come slightly closer, however imperfectly, to a manner of scientific exploration unencumbered by our own inevitable status-striving and self-deception. In the interest of honest exploration, let's do that.

The model of selection above was a model of selection for phenotypes, allowing each phenotype to vary along each of its degrees of freedom. Pause for a moment and reconsider the previous sentence. In calling something a "degree of freedom," we have (silently and implicitly) made a rather strong claim about embryology. Any such implicit claim is unavoidable in mathematical models or in verbal argument, since either discussion makes implicit assumptions about the set of possible phenotypes. Yet skeletal models of a phenomenon remain a valuable way of illuminating the relationships between ideas. Still, explicit is better than implicit, so rather than make bad assumptions about embryology implicitly, let us make such bad assumptions explicitly. As we saw above, a toy sensory system obeying Weber's law can be represented (in the discrete dialect) as a set of points of the form $x_k = x_0(1+\epsilon)^k$, where ϵ is the Weber fraction. In the discussion above, we have entirely failed to address what is perhaps the most striking feature of a representation of this form: its structural simplicity. All of our toy sensory systems are "simple" as models, but the subset obeying Weber's law is simple in a different way: they are simple to *build*. Such toy sensory systems can be generated by a single operation: repeated multiplication by a constant. In studying Weber's law, empirically or mathematically, one cannot help but feel that sensory systems with this property possess a kind of self-similarity, vaguely hinting at some simple regularity in the underlying embryological process that builds them. These intuitions are difficult to express formally, so (to the author's knowledge) no attempt has been made to formally examine Weber's law through this sideways lens: its structural simplicity.

As before, we need not solve the impossible problem of modeling the embryological development of an arbitrary sensory system in perfect detail. We only need a skeletal language for making the above intuitions precise. Whatever the details of its embryological development, Weber's law simply *is* structurally simple in this way: it is massively simpler to describe than we have any reason to expect a relationship between sensory magnitudes and jnds should be. So let us revisit the principle from the beginning of the paper: when a good model is impossible, a bad model is perfect. Here is a (carefully crafted) bad model.

By "an embryology," we will mean an ordered list of composable functions, serving as a simple representation of the algorithm by which a phenotype is built. This list of functions can be thought of as an assembly line. Our model phenotypes will be the same as in the discussion above: a finite set of dots in a continuous interval. This time, however, we will not examine the set of possible phenotypes directly, but rather the set of possible embryologies. We will build each candidate embryology – each list of functions – from a set of simple primitives. Our model phenotypes are sets of numbers, so a sensible choice of primitives would be functions that map numbers to numbers. We will choose the simplest such functions as our primitives: `add`, `mul`, and `pow`. We want to allow both quaLitative and quaNTitative mutations, and the simplest way of doing so is as follows.

Each primitive will be treated as a function of one argument, with a continuous "knob" attached. That is, in selection for embryologies, the primitives `add(x,c) = x+c`, `mul(x,c) = x*c`, and `pow(x,c) = x^c` are each thought of, not as two-parameter functions, but as continuous families of one-parameter functions, one for each c. An embryology will be an ordered list of such functions, and each primitive in that embryology will have its own binding of c to some value. Small changes to the value of c in a particular slot of a particular embryology will allow us to discuss "quaNTitative" mutations, while changes to an embryology's list of primitives will allow us to discuss "quaLitative" mu-

tations. Both types of mutations will modify the algorithm by which the phenotype is built. For example, a possible embryology would be:

```
[add(x, -0.35), mul(x, 2.14), add(x, 1.20)]
```

Given this embryology, the phenotype is constructed as follows: first, reduce the embryology to a single function by recursively composing all the functions in the list. It is useful to refer to both the list of functions and the single function it reduces to as "an embryology," and we will do so in what follows when the meaning is clear from the context. An embryology thus acts like a pipeline, starting at the right side of the list and moving left:

$$\texttt{embryology: add(}\bullet\texttt{,1.20)} \longrightarrow \texttt{mul(}\bullet\texttt{,2.14)} \longrightarrow \texttt{add(}\bullet\texttt{,-0.35)}$$

A phenotype is then constructed by recursively applying the embryology n times to a "zygote" value x_0, in order to generate the set of "dots" x_k that specify a particular phenotype. The value x_0 is fed to the pipeline to produce x_1, which is fed to the pipeline to produce x_2, and so on. Note that the resulting phenotypes need not be at all "simple" in the same sense as Weber's law, since the resulting pipelines can essentially be arbitrarily complex polynomials (decreasing in some places, increasing in others, with arbitrary slopes along the way). That said, this is not simply a model in which the embryologies are polynomials, and their parameters are mutated directly, since such a model involves only "quantitative" mutations. The list representation in terms of simpler primitives allows us to represent an embryology as a "computation," and to mutate that computation in both quaNTitative and quaLitative ways. The choice of primitives determines how mutations walk us through the larger space of functions – the space whose points are different algorithms for building a toy phenotype.

What types of mutations should we allow? Having decided on our primitives, the choice is fairly straightforward. A quaLitative mutation is simply one of the natural

operations on a list-like data structure: insertions, deletions, swaps, replacements, etc. A quaNTitative mutation is simply a mutation in the parameter c of some function in the list. To illustrate, reconsider our example embryology above.

A quaNTitative mutation might turn the item `mul(x,2.14)` into `mul(x,2.32)`, by adding the quantity 0.18 to the "knob" parameter c. Suppose the value of 0.18 is determined by drawing it from a normal distribution. Which normal distribution? A sensible choice for the mean is zero, since the smallest mutation is no mutation at all. The standard deviation is a less clear choice, but two reasonable alternatives suggest themselves: it could be either (a) a fixed number independent of c, such as $\frac{1}{10}$, or (b) a fixed proportion of c, such as $c \cdot \frac{1}{10}$. Since the best choice is unclear, both possibilities were implemented, following the same principle as before. In the code, these types of mutation are called `uniform` and `scaled`, respectively. They are both quaNTitative mutations. The value given as $\frac{1}{10}$ above is configurable using the option `sd-ish`.

A quaLitative mutation, however, might turn the list `[..., mul(x,2.14), ...]` into the list `[..., mul(x,2.14), mul(x,2.14), ...]`. More generally, instead of inserting a clone, a randomly selected primitive with a random parameter may be inserted. Further, we allow for the list items to be deleted by mutations, which might turn `[f,g,h]` into `[f,h]`. Mutations can also swap adjacent elements, turning (say) `[f,g,h]` into `[f,h,g]`. Finally, we allow for the possibility of changing a primitive without changing its parameter, so a list item `mul(x,2.14)` may mutate into `add(x,2.14)` or `pow(x,2.14)`, etc.

The toy language above is the result of a few simple ideas: (a) an "embryology" is a computation that builds a phenotype, (b) computation is the composition of functions, and (c) by item b, different computations can be represented as different compositions of different functions. The set of possible toy embryologies is large, but extremely simple to understand. Once we decided on a set of primitives, the set of possible mutations (both

quantitative and qualitative) turned out to be a fairly straightforward decision, requiring fewer choices along the way than we might have expected.

So, what now? There are two overlapping goals. The narrow goal is to examine whether Weber's law evolves under selection for any of our metrics from earlier. The more broad goal is to explore this oddball model of selection, and its competence in climbing hills and finding optima. What on earth happens under selection for different embryologies? Can we learn anything of value? Let's see.

The first thing one notices is that selection for embryologies is both much faster and much slower than selection for phenotypes. There are typically significant stretches of time in which nothing seems to happen. These periods of stasis are punctuated by what appear to be massive instantaneous changes – macromutations – at least when viewed from the "geological time scale" of the graphs in the previous section. At first, this seemed to suggest that the algorithm was deeply broken, or at least deeply uninteresting, and therefore unlikely to be worth even a brief discussion. However, a more fine-grained, generation-by-generation examination reveals more interesting dynamics. As the code runs, it prints out a log summarizing the winner in each generation, with respect to whichever metric we are examining. This log was written primarily for debugging, and for anyone else who might be running the code. However, the raw log output itself often turns out to be a more effective way to convey the dynamics of embryology selection than a graph showing how some scalar quantity varies with respect to another. The following is an excerpt from the middle of a run, examining embryology selection for our old friend the minivar metric. The numbers on the left are just the values of that metric over time; the specific values are not important, but lower is better.[6] In light of the discussion above, this excerpt from the raw log output gives one a surprisingly good understanding

[6]Note: each line represents the overall "winner" among all the offspring in a generation, which makes beneficial mutations appear rather common. However, this is simply selection in action. That is, each line represents a single round of mutation *and* selection, not simply mutation.

of the dynamics one sees under selection for embryologies.

```
0.138620661563 [pow(x, c = 3.68)] none:none
0.138620661563 [pow(x, c = 3.68)] none:none
0.138620661563 [pow(x, c = 3.68)] none:none
0.138620661563 [pow(x, c = 3.68)] none:none
0.138620661563 [pow(x, c = 3.68)] none:none
0.138620661563 [pow(x, c = 3.68)] none:none
0.138620661563 [pow(x, c = 3.68)] none:none
0.138620661563 [pow(x, c = 3.68)] none:none
0.138620661563 [pow(x, c = 3.68)] none:none
0.138620661563 [pow(x, c = 3.68)] none:none
0.138620661563 [pow(x, c = 3.68)] none:none
0.138620661563 [pow(x, c = 3.68)] none:none
0.138620661563 [pow(x, c = 3.68)] none:none
0.138620661563 [mul(x, c = 0.73), pow(x, c = 3.68)] qualitative:insert
0.138620661563 [mul(x, c = 0.73), pow(x, c = 3.68)] none:none
0.138620661563 [mul(x, c = 0.81), pow(x, c = 3.68)] quantitative:relative
0.136983156340 [mul(x, c = 1.02)] quantitative:uniform,qualitative:delete
0.136983156340 [mul(x, c = 1.02)] none:none
0.136983156340 [mul(x, c = 1.02)] none:none
0.094783182253 [mul(x, c = 1.44), mul(x, c = 1.02)] qualitative:insert
0.089452371115 [mul(x, c = 1.44)] qualitative:delete
0.089452371115 [mul(x, c = 1.44)] none:none
0.084289995949 [mul(x, c = 1.41)] quantitative:relative
0.084289995949 [mul(x, c = 1.41)] none:none
0.069470091540 [mul(x, c = 1.34)] quantitative:relative
0.069470091540 [mul(x, c = 1.34)] none:none
0.045299720100 [mul(x, c = 1.26)] quantitative:uniform
0.036167891853 [mul(x, c = 1.23)] quantitative:uniform
0.036167891853 [mul(x, c = 1.23)] none:none
0.036167891853 [mul(x, c = 1.23)] qualitative:swap
0.021305009638 [mul(x, c = 1.19)] quantitative:relative
0.021305009638 [mul(x, c = 1.19)] none:none
0.021305009638 [mul(x, c = 1.19)] none:none
```

This raw data-dump of a rather exotic selection model contains a surprising number of familiar lessons about selection. First, at the beginning of the run, we see the status quo winning every generation; the vast majority of mutations are not beneficial. Then, the status quo is interrupted, not by a macromutation or even by a minor improvement, but

by drift. The first insertion above happened to have an identical fitness by the minivar metric – at least within the precision of the log output above – and it happened to have been selected as the winner by pure chance. However, that small amount of variation gave selection more to work with, and things start to get interesting. The first visible improvement is small, but it resulted from two simultaneous mutations: a deletion and a minor quantitative change. We then see a succession of small changes. First, the phenotype transitions from `[mul(x, c = 1.02)]` to `[mul(x, c = 1.44)]`, but it does so in a funny way: neither by many small quantitative mutations, nor by a massively improbable "macromutation," but by two small qualitative mutations in different generations, each of which had fairly minor effects on the overall structure of the phenotype, but observable effects on fitness. We then see the gradual accumulation of small, quantitative changes in each generation, with an occasional `none:none` representing a generation in which all available mutations happened to be harmful. Those quantitative changes push the numbers on the left toward their theoretical optimum, without any changes to the phenotype's qualitative structure. In the end, we have a simple embryology: `[mul(x, c = 1.19)]` representing an algorithm that recursively multiplies its toy zygote x_0 by a fixed number $(1+\epsilon)$, for some ϵ slightly greater than 0. In the end, the toy embryology has built a toy phenotype obeying Weber's law, using a model of selection seemingly different in every way from our original model of selection for phenotypes.

Compared to selection in the real world, the excerpt above has clearly got the speed all wrong; selection in real organisms is surely much slower, by any measure. But despite its wildly unrealistic speed, we observed a number of interesting phenomena – more than we expected to observe from this simple toy model of embryology selection. The earlier model of phenotype selection was instructive, but it had a rather limited imagination: selection could only move dots left and right, by relatively small amounts. In this more exotic model, selection was equally constrained by the dictates of fitness. However, by examining

not the phenotypes themselves, but the algorithms that build them, our model gained an interesting ability: the ability of small quaNTitative changes to be supplemented, not by miraculously restructuring macromutations, but by equally small quaLitative changes: insertions and deletions of primitives in the embryological "computation."

Chapter 4

Discussion

So, do the above results teach us anything of value about real-world phenotypes, or real world embryologies? Maybe, but we would not argue the point with any confidence. However, they do teach us something about Weber's law. The optimality and evolvability analyses above offer an imperfect but unambiguous proof-of-concept that this widespread structure is likely to be an adaptation by natural selection: a feature, not a bug. We began by demonstrating that this structure is "optimal" by three different measures of quality: minimax, minivar, and totalre. These are three different mountains with a shared peak: the toy sensory systems obeying Weber's law. Such optimality analyses are a helpful illustration of the relationships between ideas, but the differences between them are difficult to interpret by any means other than intuition. We then compared these metrics under a simple model of selection, examining the ability of each to reach its own optimum. In doing so, we found that the clear winner under selection was the unsophisticated underdog, minivar, and argued that its success lied in its ability to efficiently take advantage of any mutations that happened to push it closer to its own optimum. Having developed a supplement to intuition by studying phenotype selection, we attempted to break those new intuitions by building a model of selection that was different in almost

every way. This final analysis focused on selection between embryologies – the algorithms by which phenotypes are built. This analysis offered a new way of understanding the structural simplicity of Weber's law, and additionally demonstrated that Weber's law is evolvable under this more exotic type of selection as well. In summary, the results of this paper provide an imperfect but unambiguous demonstration that this widespread feature of the natural world – observed in multiple sensory systems in a diverse array of species – is both optimal and evolvable, and achieves each in a variety of senses.

Chapter 5

Epilogue: The kernel of an old debate

Throughout this paper, we have thus far failed to address an important debate in the surrounding literature: the issue of logarithmic vs linear representation. When Weber's law is observed in a sensory or cognitive system, it is often argued that the system in question maps objective quantities to their logarithms. The issue dates back to Fechner (1860), who first argued that Weber's law implies a logarithmic representation of sensory magnitude. This view is common in the modern literature (Dehaene, 2003; Portugal & Svaiter, 2010; Sun, Goyal, & Varshney, 2012), while others have argued that a linear representation is more likely (Gallistel, 2011; Gibbon & Church, 1981), or that the implication need not hold (Laming, 2010). A question of this age is unlikely to be resolved any time soon. Faced with such a difficult question about the brain, it is often instructive to examine a real world analogue in more well-understood, human-made computing systems. However different the two may be, doing so can help to sharpen our thinking by teaching us what is possible in computing, and by giving us a simplified sandbox in which to determine whether there is, in fact, any genuine disagreement. Can we find

a real world example in which Weber's law – the linear scaling of jnds – is found, but Fechner's law – the mapping of quantities to their logarithms – is not?

The "kernel" of an operating system is a piece of low-level software that deals directly with the underlying hardware of the machine. It does so in order to provide an abstraction layer between the hardware and the rest of the code above it, so that everyday programs can simply issue high-level requests like "create a file," "give me a pointer to 1024 bytes of free memory," or "execute this code," without worrying about the low-level details of how this plumbing is implemented.

One of the many plumbing tasks a kernel has to perform is memory management – allocating different subsets of the system's available memory to different tasks, so that tasks running concurrently or in parallel do not step on each other's toes, writing to memory that another task is already using to store execution state or data. Allocating free memory is such a fundamental task in a multitasking computing system that the algorithms for carrying it out have been heavily optimized over time. Perhaps the most heavily optimized such system is found in the Linux kernel.[1]

The heart of the Linux kernel's memory management subsystem is a function called `kmalloc`. The kernel allocates memory in "pages": chunks of a fixed size, usually 4 kilobytes (Rusling, 1999). To simplify and speed-up the process of memory allocation, `kmalloc` will only allocate pages in blocks that are a power of two in size (see Rusling, 1999, chapter 3). When the kernel receives a "stimulus" in the form of request for (say) 60 pages of memory, that stimulus is mapped to an "internal representation" in the form

[1]In 2014, Linux ran 95.2% of the world's top 500 supercomputers (Prometeus GmbH, 2014), up from 87.8% in 2007 (Vaughan-Nichols, 2007). Linux powers 76.6% of smartphones worldwide as of 2014 (International Data Corporation, 2015); Android is a derivative of Linux, as is Chrome OS (Vaughan-Nichols, 2012). Linux runs 60% of web servers, as estimated by a competitor (Niccolai, 2008), and is found on a wide variety of routers, televisions, refrigerators, traffic lights, security cameras, humanoid robots, and lightbulbs, as well as many of the embedded hardware devices inside consumer PCs (Linux Devices Archive, 2012). We mention this only to emphasize that the following example was not cherry-picked from an obscure computing system. In spite of its lack of a visible presence in consumer markets, Linux is arguably the most widespread general purpose operating system in the history of computing.

of a request for 64 pages, since 64 is the next highest power of two. As a result, the Linux kernel's memory allocation subsystem exactly obeys Weber's law with respect to the quantity "number of memory pages requested." The jnd between two such quantities is not constant, but scales linearly, in proportion to the absolute magnitude of those requests.

This concrete example can help to clarify our thoughts surrounding the "linear vs log" debate in the psychophysical literature. At first, the example appears to align with only one side of this old debate: While the representation has linearly increasing jnds, it is not the case that the requested number of pages is transformed into its base 2 logarithm. On the contrary, the representation is as close to the identity as a map obeying Weber's law can be. So clearly, the mapping is linear, not logarithmic. Or is it? What does it mean to say that a quantity is "mapped to its logarithm"?

To illustrate: a request for 2000 pages of free memory is made, and the kernel maps this number to 2048, which in a 16-bit binary representation is 0000100000000000. Was this number "mapped to its logarithm," or wasn't it? In one sense, clearly not: $\log_2(2000)$ is approximately 11, which is nowhere near 2048. In another sense, yes. The logarithm of 0000100000000000 is available immediately. This number has a 1 in its 11^{th} slot, with zeros elsewhere, and $\log_2(2048)$ is exactly 11. Is the number "mapped to its logarithm" or is it not? In a precise sense: it is both.

In the same set of bits, we have a *linear* representation in a *binary* code, and a *logarithmic* representation in a *unary* code. The unary code is simply the number of binary digits one must cross to find the one and only 1. Both quantities are represented simultaneously, in different bases, in the same set of bits. In our field, proponents of the linear theory emphasize the need for arithmetic operations (Gallistel, 2011). We agree wholeheartedly. However, in this simple example from the Linux kernel, performing such operations is trivial in either interpretation: the successor operation in the

log-interpretation consists of shifting a single bit one position to the left: transforming 0000100000000000 into 0001000000000000, for instance. This is a single machine instruction in every modern CPU architecture: in x86 assembly it is written `sal`, for "**s**hift **a**rithmetic **l**eft." When used on a set of quantities obeying Weber's law, any such operation will have two equally valid interpretations: one involving the number itself, another involving its logarithm.

More detailed examination blurs the distinction even further. The kernel *uses* the ease with which the log can be extracted from the representation to optimize page allocation. Since 2048 is the 11^{th} power of two, it is also the 11^{th} item on the list of page-block sizes that the kernel is willing to hand out. Exploiting this fact, the kernel keeps an array called `free_area` whose k^{th} element contains a memory map describing all of the free and used blocks of 2^k contiguous pages of memory (see Rusling, 1999, section 3.4.1, Page Allocation). Both interpretations are used explicitly, and with zero overhead in switching between them: the computation of the log is not "computation," but reinterpretation.

These two elegantly superimposed representations – linear in binary, log in unary – arise simply from the fact that the quantities involved were powers of n, represented in base n, where (for the kernel) n happens to be 2. However, as we have seen repeatedly throughout this paper, Weber's law can be described in the discrete dialect as a representation built from successive powers of a different number: $(1 + \epsilon)$, where ϵ is the Weber fraction.

Against this background, it is worth questioning whether the disputants on either side of the "linear vs log" debate are, in fact, in a state of disagreement. Weber's law has somehow managed to give sufficient simultaneous evidence of both logarithmic and linear mappings as to fuel many years of debate over which form the representation *really* takes. Is one side crazy? Or are both sides describing the same elephant?

In summary, however Weber's law works in the nervous system, Weber's law in the

Linux kernel works in a way that may shed light on the linear vs log debate in our field. It is not known at present how something as simple as a phone number is physically represented in the nervous system, so the issue of lines and logs is unlikely to be resolved any time soon. However, in what is perhaps the most heavily optimized and performance-tuned code-base in the world, there exists a principled reason to allocate resources with an exponential spacing. The resources thus represented are not mapped to their logarithms in the usual sense. But they absolutely *are* mapped to their logarithms in an unusual sense: as a unary representation overlaid on a binary one. No debate over lines and logs has ever arisen over this detail of the Linux kernel's memory allocation. But were any such debate ever to arise, we can confidently say that the valuable time of disputants on either side might be better spent burying the hatchet, and perhaps sharing a drink over what is – after all – a shared research interest. Without any need to compromise, it is possible that both sides are right.

References

Akre, K. L., & Johnsen, S. (2014). Psychophysics and the evolution of behavior. *Trends in Ecology and Evolution*, *29*, 291-300.

Bizo, L., Chu, J., Sanabria, F., & Killeen, P. R. (2006). The failure of weber's law in time perception and production. *Behavioural Processes*, *71*, 201-210.

Dehaene, S. (2003). The neural basis of the weber-fechner law: a logarithmic mental number line. *Trends in Cognitive Sciences*, *7*, 145-147.

Dehaene, S., Dehaene-Lambertz, G., & Cohen, L. (1998). Abstract representations of numbers in the animal and human brain. *Trends in Neurosciences*, *21*, 355-361.

Dews, P. B. (1970). The theory of fixed-interval responding. In W. N. Schoenfeld (Ed.), *The theory of reinforcement schedules.* New York: Appleton-Century-Crofts.

Fechner, G. (1860). *Elemente der psychophysik*. Leipzig: Breitkipf and Härtel.

Gallistel, C. R. (2011). Mental magnitudes. In S. Dehaene & L. Brannon (Eds.), *Space, time and number in the brain: Searching for the foundations of mathematical thought* (p. 3-12). New York: Elsevier.

Gibbon, J., & Church, R. M. (1981). Time left: linear versus logarithmic subjective time. *Journal of Experimental Psychology: Animal Behavior Processes*, *7*, 87-107.

Grondin, S., Ouellet, B., & Roussel, M. (2001). About optimal timing and stability of weber fraction for duration discrimination. *Acoustical Science and Technology*, *22*, 370-372.

International Data Corporation. (2015, January). *Smartphone os market share, q4 2014.* Retrieved from `idc.com/prodserv/smartphone-os-market-share.jsp`

Jordan, K., & Brannon, E. (2006). Weber's law influences numerical representations in rhesus macaques (macaca mulatta). *Animal Cognition*, *9*, 159-172.

Laming, D. (2010). Fechner's law: where does the log transform come from? *Seeing Perceiving*, *23(2)*, 155-171.

Linux Devices Archive. (2012, February). *Linux devices archive index.* Retrieved from `archive.linuxgizmos.com`

Masin, S. (2009). The (weber's) law that never was. In *Proceedings of fechner day, vol 25* (p. 441-446).

Niccolai, J. (2008, September). *Ballmer still searching for an answer to google.* Retrieved from `pcworld.com/article/151568/article.html`

Piantadosi, S. T. (in press). A rational analysis of the approximate number system.

Portugal, R., & Svaiter, B. (2010). Weber-fechner law and the optimality of the logarithmic scale. *Minds and Machines*, *21*, 73-81.

Prometeus GmbH. (2014, November). *Top 500 list: June 2014.* Retrieved from `top500.org/list/2014/06`

Rabinowitz, W. M., Lim, J. S., Braida, L. D., & Durlach, N. I. (1976). Intensity preception. vi. summary of recent data on deviations from weber's law for 1000-hz tone pulses. *The Journal of the Acoustical Society of America*, *59(6)*.

Rusling, D. (1999). *The linux kernel*. Free book, distributed under the GNU General Public License.

Simpson, S. (2009). *Subsystems of second order arithmetic*. Cambridge University Press.

Sun, J., Goyal, V., & Varshney, L. (2012). A framework for bayesian optimality of psychophysical laws. *Journal of Mathematical Psychology*.

Vaughan-Nichols, S. (2007, November). *The fastest computers are linux computers.* Retrieved from `practical-tech.com/infrastructure/the-fastest-computers-are-linux-computers`

Vaughan-Nichols, S. (2012, March). *Android and linux re-merge into one operating system.* Retrieved from `zdnet.com/article/android-and-linux-re-merge-into-one-operating-system`

Wearden, J. H., & McShane, B. (1988). Interval production as an analogue of the peak procedure: Evidence for similarity of human and animal timing processes. *Quarterly Journal of Experimental Psychology*, *40B*, 363-375.

Wilkes, J. (in press). *Burn math class: A mathematical creation story*. New York: Basic Books.