Log, Stock and Two Simple Lotteries. Technical Supplement*

Sergiy Verstyuk†

This version: 22 January 2018

Abstract

These supplemental materials include technical details on the algorithm for decision-making under risk utilized in the main text, as well as a concise review of neural foundations underlying the main text's theoretical framework. Several peripheral extensions to the main work are also placed here.

*Acknowledgements: see the main text.

†Contact address: verstyuk@cmsa.fas.harvard.edu.

Contents

F Algorithm for decision-making under risk

F.1 Scalar random variable case

Consider the primitive construction block of the process of decision-making under uncertainty: evaluation of a simple lottery. We will present the detailed steps of such an evaluation using an illustrative example.

Let x be a discretely distributed scalar random variable with probability mass function $\ddot{g}(x)$:

$$x \sim \ddot{g}(x), \tag{F.1}$$

with $\ddot{g}(x)$ defined, say, as in Figure F.0.

$$\ddot{g}(x) = \begin{cases} 1/8 & \text{if } x = 7, \\ 1/8 & \text{if } x = 9, \\ 1/4 & \text{if } x = 11, \\ 1/4 & \text{if } x = 13, \\ 1/8 & \text{if } x = 19, \\ 1/8 & \text{if } x = 21. \end{cases}$$

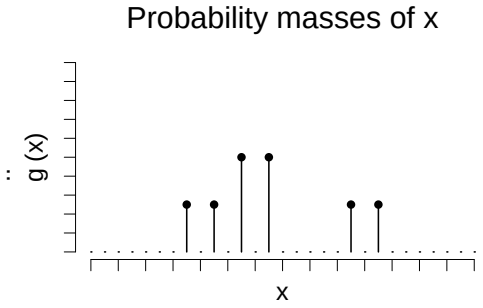


Figure F.0: Description of $\ddot{g}(x)$.

Define (Shannon) entropy of a random variable x (or rather of the corresponding probability distribution) as

$$\mathcal{E}(\ddot{g}(x)) := -\sum_{i=1}^{n_q} \ddot{g}(x_i) \log \ddot{g}(x_i), \tag{F.2}$$

where

$$n_q := |\text{supp}(\ddot{g})|. \tag{F.3}$$

Entropy is a measure of the average uncertainty in a random variable. Note the important feature of the entropy functional (and its relatives) in that it does not depend on the actual values taken by a random variable, $\{x_i\}_1^{n_q}$, and is only a function of the probability masses, $\{\ddot{g}(x_i)\}_1^{n_q}$. In our case,

$$\mathcal{E}(\ddot{g}(x)) = -\left(4 \times \frac{1}{8} \log \frac{1}{8} + 2 \times \frac{1}{4} \log \frac{1}{4}\right) = 2.5 \text{ bits}. \tag{F.4}$$

Step 1: Simplification of discrete distribution.

Let

$$\hat{x} \sim \ddot{h}(\hat{x}) \tag{F.5}$$

$$\ddot{h}(\hat{x}) = \begin{cases} 1/4 & \text{if } \hat{x} = 8, \\ 1/2 & \text{if } \hat{x} = 12, \\ 1/4 & \text{if } \hat{x} = 20. \end{cases}$$

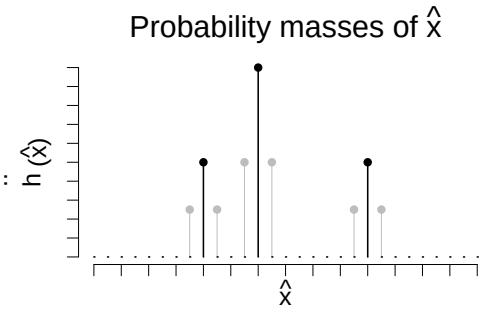


Figure F.1: Description of $\ddot{h}(\hat{x})$.

be a simplified version of x, described in Figure F.1.
(Specific form of $\ddot{h}(\cdot)$, in particular its relation to $\ddot{g}(\cdot)$, is a matter of choice that is discussed later.)

Probability mass function $\ddot{h}(\hat{x})$ is coarser than $\ddot{g}(x)$, i.e. is characterized by lower entropy:

$$\begin{aligned} \mathcal{E}(\ddot{h}(\hat{x})) &= -\sum_{j=1}^{\hat{n}_q} \ddot{h}(\hat{x}_j) \log \ddot{h}(\hat{x}_j) = \\ &= -\left(2 \times \frac{1}{4} \log \frac{1}{4} + \frac{1}{2} \log \frac{1}{2}\right) = 1.5 \text{ bits}, \end{aligned} \tag{F.6}$$

where

$$\hat{n}_q := |\text{supp}(\ddot{h})|. \tag{F.7}$$

Step 2: Generating codebook.

Assume uniform discretization of the range of $\ddot{G}(x)$ into n_d quantiles, where the number of discretization points is defined by

$$n_d := \frac{1}{\delta}$$

for cell size

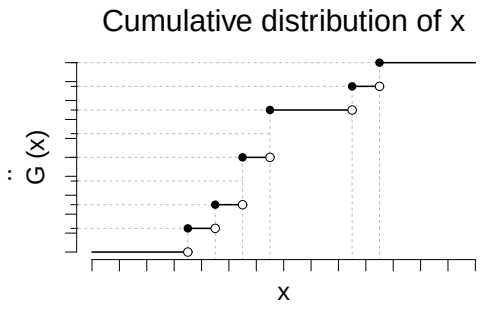
$$\delta := \gcd(\{\ddot{g}(x_i)\}_1^{n_q}),$$

with $\gcd(a_1, \ldots, a_n)$ returning the greatest common divisor of $a_1, \ldots, a_n \in \mathbb{R}$, so that

$$\ddot{G}(x_i) \geq \ddot{G}(x_{i-1}) + \delta.$$

Left panel of Figure F.2.1 visualizes the process of discretization.

Right panel of Figure F.2.1 exhibits a codebook, a map from source alphabet to output alphabet, which in our case summarizes optimal code for probability distribution $\ddot{G}(x)$. We will use it for exchanging information that sources from this distribution (like a binary Morse code for encoding English letters and Arabic numerals, or genetic code in the DNA



Codebook for $\ddot{G}(x)$

x	$\ddot{G}(x)$	$\ddot{g}(x)$	Native code	Optimal code
7	$1/8$	$1/8$	000111	000
9	$2/8$	$1/8$	001001	001
11	$4/8$	$1/4$	001011	10
13	$6/8$	$1/4$	001101	11
19	$7/8$	$1/8$	010011	010
21	$8/8$	$1/8$	010101	011
$\mathrm{E}[L(x)]$	-	-	6	2.5

Figure F.2.1: Generating codebook for $\ddot{G}(x)$.

with four nucleotides for encoding different amino acids). It is constructed using the most basic algorithm for optimal coding, Huffman procedure.

Given some alphabet, expected length of any instantaneous (this term is defined later) code for random variable x is bounded below by the entropy of x:

$$\mathrm{E}[L(x)] = \sum_{i=1}^{n_q} \ddot{g}(x_i) L(x_i) \geq \mathcal{E}(\ddot{g}(x)), \tag{F.8}$$

where $L(x_i)$ is the length of the codeword associated with x_i.

Optimal coding guarantees an expected codeword length within 1 bit of the lower bound, i.e. $\mathcal{E}(\ddot{g}(x)) \leq \mathrm{E}[L^*(x)] < \mathcal{E}(\ddot{g}(x)) + 1$, and expected length of Huffman code is at least as small as that of any other optimal code. In our simple example, Huffman code achieves the lower bound: $\mathrm{E}[L^*(x)] = \mathcal{E}(\ddot{g}(x)) = 2.5$ bits, c.f. equation (F.2). (Incidentally, Huffman coding here is equivalent to another optimal coding procedure, so called Shannon code, which assigns codeword lengths of $\lceil \log 1/\ddot{g}(x) \rceil$ and overshoots the lower bound by less than 1 bit.)

Additonaly, Figure F.2.1 provides what we call "native" code for x. It is constructed using alphabet $\mathcal{A} := \{\texttt{000000}, \ldots, \texttt{111111}\}$, which is assumed to be the default code that the decision-making agent is endowed with at the outset and that is used for all operations (such as communication, computation, etc.). It is not optimized for probability distribution of x, or to be more accurate, assumes uniform distribution of a random variable over the domain $\{0, \ldots, 63\}$, and uses fixed 6-bit-long codewords. (This distinguishment is without loss of generality, and native code may coincide with optimal code.)

Similarly, assume uniform discretization of the range of $\ddot{H}(\hat{x})$ into $\hat{n}_d$ quantiles, where

$$\hat{n}_d := \frac{1}{\hat{\delta}}$$

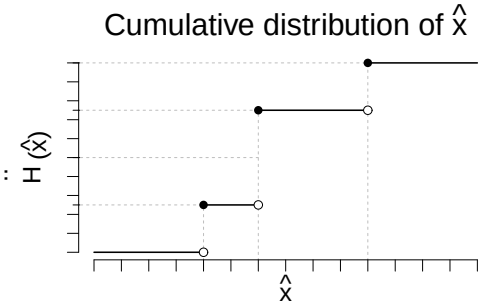
for cell size

$$\hat{\delta} := \gcd(\{\ddot{h}(\hat{x}_j)\}_1^{\hat{n}_q}),$$

so that

$$\ddot{H}(\hat{x}_j) \geq \ddot{H}(\hat{x}_{j-1}) + \hat{\delta}.$$

Figure F.2.2 presents discretization and coding results for probabiltiy distribution $\ddot{H}(\hat{x})$.



Codebook for $\ddot{H}(\hat{x})$

$\hat{x}$	$\ddot{H}(\hat{x})$	$\ddot{h}(\hat{x})$	Native code	Optimal code
8	$1/4$	$1/4$	001000	00
12	$3/4$	$1/2$	001100	1
20	$4/4$	$1/4$	010100	01
$\mathrm{E}[L(\hat{x})]$	-	-	6	1.5

Figure F.2.2: Generating codebook for $\ddot{H}(\hat{x})$.

Again, the optimal code reaches lower bound $\mathrm{E}[L^*(\hat{x})] = \mathcal{E}(\ddot{h}(\hat{x})) = 1.5$ bits, as in equation (F.6), using 1 to 2 bits per codeword; while the native code still uses 6 bits as before.

In general, native codebook differs from optimal codebook, which leads to certain code redundancy. Such redundancy, or what we call code overhead, can take one of two different forms (leading the discussion below in terms of $\ddot{h}(\hat{x})$, which in this connection is more relevant):

(i) D-overhead is an artefact of divergence of native codebook from optimal one and arises when using the former instead of the latter. This requires (applying "Wrong code" theorem in Cover and Thomas, 2006):

$$\begin{aligned}\text{D-overhead} &= \hat{n}_d \times \mathcal{D}\left(\ddot{h}(\hat{x}) \,\middle\|\, \frac{1}{|\mathcal{A}|}\right) = \hat{n}_d \times \sum_{j=1}^{\hat{n}_q} \ddot{h}(\hat{x}_j) \log \frac{\ddot{h}(\hat{x}_j)}{1/|\mathcal{A}|} = \\ &= \hat{n}_d \times \left(\log|\mathcal{A}| - \mathcal{E}(\ddot{h}(\hat{x}))\right), \end{aligned} \tag{F.9}$$

which is bounded by

$$\text{D-overhead} \in [0, \hat{n}_d \log|\mathcal{A}|] \tag{F.10}$$

(for $\mathrm{supp}(\ddot{h}(\hat{x})) \subseteq \mathcal{A}$, and since uniform distribution is a maximum entropy distribution for a given finite support set).

Here, relative entropy (or Kullback–Leibler divergence) $\mathcal{D}(\ddot{\pi}_1 \| \ddot{\pi}_2)$ for some probability mass functions $\ddot{\pi}_1$ and $\ddot{\pi}_2$ is defined as follows:

$$\mathcal{D}(\ddot{\pi}_1(\chi) \| \ddot{\pi}_2(\chi)) := \sum_{i=1}^{|\mathrm{supp}(\ddot{\pi}_1)|} \ddot{\pi}_1(\chi_i) \log \frac{\ddot{\pi}_1(\chi_i)}{\ddot{\pi}_2(\chi_i)}. \tag{F.11}$$

In our example D-overhead amounts to

$$\text{D-overhead} = \hat{n}_d \times \left(\log|\mathcal{A}| - \mathcal{E}(\ddot{h}(\hat{x}))\right) = 4 \times (6 - 1.5) = 18 \text{ bits}. \tag{F.12}$$

(ii) P-overhead comprises the preamble that provides the description of the optimal codebook. In our case it amounts to:

$$\begin{aligned}\text{P-overhead} = \text{“ } &\texttt{001000 000010 00 001100 000001 1}\\ &\underbrace{\texttt{010100}}_{\substack{\text{value}}}\ \underbrace{\texttt{000010}}_{\substack{\text{length of}\\ \text{codeword}}}\ \underbrace{\texttt{01}}_{\text{codeword}}\ \underbrace{\texttt{111111}}_{\substack{\text{EOM}\\ \text{symbol}}}\text{”} =\\ &= 3\times(6+6)+5+6 = 47 \text{ bits,}\end{aligned} \tag{F.13}$$

and in general (allowing for maximum length of a codeword to be 64 bits) this requires:

$$\begin{aligned}\text{P-overhead} &= |\text{supp}(\ddot{h})|\times(\log|\mathcal{A}|+\log|\mathcal{A}|)+\sum_{j=1}^{|\text{supp}(\ddot{h})|}L^*(\hat{x}_j)+\log|\mathcal{A}|\geq\\ &\geq |\text{supp}(\ddot{h})|\times 2\log|\mathcal{A}|-\sum_{j=1}^{|\text{supp}(\ddot{h})|}\log\ddot{h}(\hat{x}_j)+\log|\mathcal{A}|\geq\\ &\geq |\text{supp}(\ddot{h})|\times 2\log|\mathcal{A}|+|\text{supp}(\ddot{h})|\times\log|\text{supp}(\ddot{h})|+\log|\mathcal{A}|=\\ &= \hat{n}_q\times(2\log|\mathcal{A}|+\log\hat{n}_q)+\log|\mathcal{A}|,\end{aligned} \tag{F.14}$$

with equality signs for $\ddot{h}(\hat{x})$ that is both dyadic and uniform (here the first inequality is due to bound on the optimal code length). However, P-overhead is unbounded from above even for fixed $\hat{n}_q$ as optimal $L^*(\hat{x}_j)\to\infty$ for $\hat{x}_j$ such that $\ddot{h}(\hat{x}_j)\to 0$ (in which case D-overhead also blows up owing to $\hat{n}_d\to\infty$).

The choice of smallest overhead is straight-forward:

$$\begin{aligned}\text{overhead}^* &:= \min\{\text{D-overhead}, \text{P-overhead}\} =\\ &= \min\{18, 47\} = 18 \text{ bits.}\end{aligned} \tag{F.15}$$

It is clear that for finite alphabet $\mathcal{A}$, D-overhead $\in\mathcal{O}(\hat{n}_d)$, while P-overhead is at best an $\mathcal{O}(\hat{n}_q\log\hat{n}_q)$ expression. Given that $\hat{n}_q\leq\hat{n}_d$, asymptotically P-overhead may become more attractive, but only in cases when resolution of discretization is much higher than resolution of quantization (more on which later).

In information theory arguments it is often assumed for reasons of simplicity that the alphabet used for coding is the optimal one and achieves the lower bound of expected code length. Within our framework, this situation can be understood as asymptotics in terms of number of entire algorithm's uses while holding other parameters such as $|\text{supp}(\ddot{h})|$, $|\mathcal{A}|$, $\max_i L^*(\hat{x}_i)$ finite, i.e. as effectively changing the alphabet used by paying fixed but asymptotically negligible costs of the necessary P-overhead. However, asymptotic point of view is sensible only when dealing with stationary probability distributions, hence it would loose the generality with respect to potential applications.

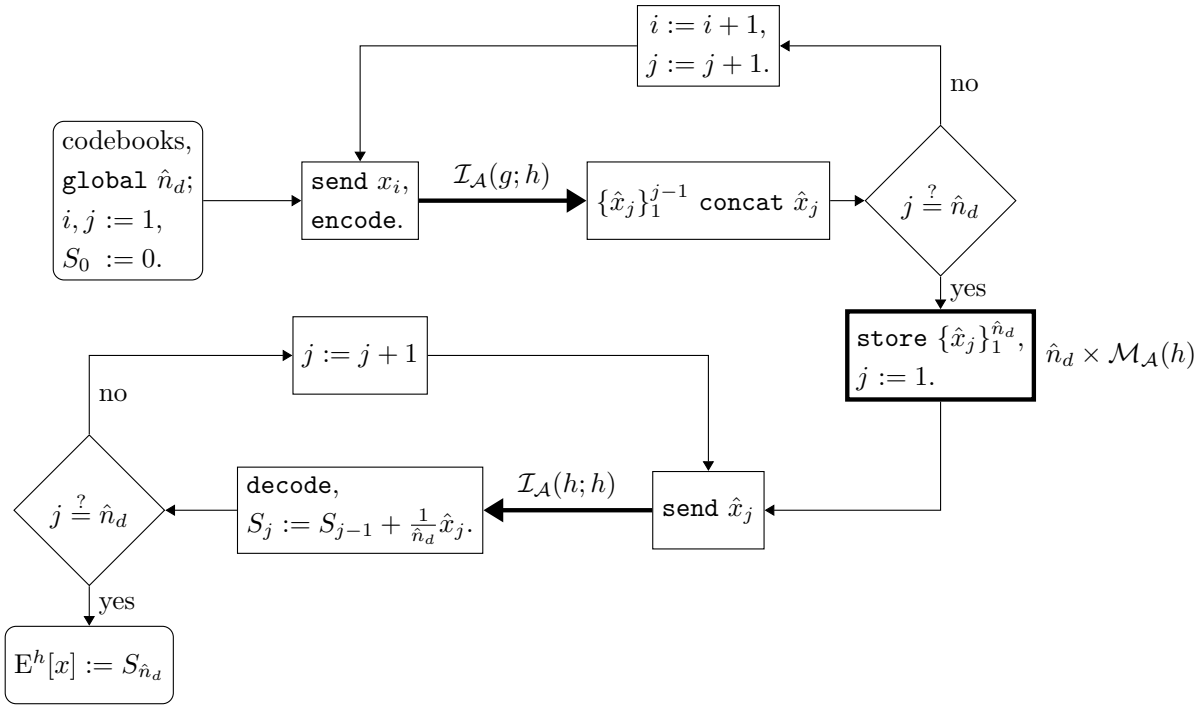


Figure F.3: Flowchart of description, storage and computation steps (scalar random variable case, computation of the mean).

Step 3: Description of simplified distribution.

After codebook generation step, the next three steps are best illustrated by the flowchart on Figure F.3.

Description of the simplified distribution is conducted by sending values of its domain at each discretization interval via communication channel. But first, consider the task of sending via communication channel just a single realization of random variable.

In information theory, communication channels are characterized by their channel capacity, which is formally defined as the maximum mutual information between input and output random variables:

$$\max_{\ddot{g}(x)} \left\{ \mathcal{I}(\ddot{g}(x); \ddot{h}(\hat{x})) \right\}. \tag{F.16}$$

In turn, mutual information between random variables x and $\hat{x}$ is defined as the relative entropy between their joint probability mass function $\ddot{f}(x, \hat{x})$ and the product of their marginal probability mass functions $\ddot{g}(x)\ddot{h}(\hat{x})$, which makes it

$$\mathcal{I}(\ddot{g}(x); \ddot{h}(\hat{x})) := \mathcal{D}\left(\ddot{f}(x, \hat{x}) \| \ddot{g}(x)\ddot{h}(\hat{x})\right) = \sum_{i=1}^{n_q} \sum_{j=1}^{\hat{n}_q} \ddot{f}(x_i, \hat{x}_j) \log\left(\frac{\ddot{f}(x_i, \hat{x}_j)}{\ddot{g}(x_i)\ddot{h}(\hat{x}_j)}\right). \tag{F.17}$$

It can be shown that

$$\begin{aligned} \mathcal{I}(\ddot{g}(x); \ddot{h}(\hat{x})) &= \mathcal{E}(\ddot{g}(x)) + \mathcal{E}(\ddot{h}(\hat{x})) - \mathcal{E}(\ddot{g}(x), \ddot{h}(\hat{x})) = \\ &= \mathcal{E}(\ddot{g}(x)) - \mathcal{E}(\ddot{f}(x|\hat{x})) = \mathcal{E}(\ddot{h}(\hat{x})) - \mathcal{E}(\ddot{f}(\hat{x}|x)), \end{aligned} \tag{F.18}$$

thus mutual information measures expected reduction in uncertainty about a random variable after observing realization of another random variable.

Operationally, channel capacity is defined as the highest rate (in bits per channel use, or per unit of time) at which information can be sent with arbitrarily low error probability. (To be pedantic, bandwidth is one of the factors determining channel capacity.) That is, channel capacity imposes a constraint on the complexity of messages that can be sent via it without error. Message is a codeword representing realization of a random variable. Complexity of a message is measured by its entropy, i.e. by entropy of the underlying random variable, because one of the crucial features of the entropy functional is that it does not depend on the actual values taken by a random variable, but only on the corresponding probability masses. Therefore, channel capacity limits the admissible entropy of the output, and whenever the entropy of the desired input is higher than available capacity, sending information without error requires an adjustment to the input. This leads us to using input from the simplified probability distribution $\ddot{h}(\hat{x})$ rather than from the original probability distribution $\ddot{g}(x)$. Note that for transparency reasons, we choose to deal here with bearing losses at the time of encoding but with a lossless communication itself, which implies $\mathcal{E}(\ddot{f}(\hat{x}|x)) = 0$ in (F.18).

Proceeding further, we now consider usage of the communication channel to transmit the description of the simplified distribution by sending values of its domain, $\hat{x}$, at each discretization interval. The construction that follows will be based on Proposition F.1.

Proposition F.1 (Description of Probability Distribution)**.** *Let χ be random variable distributed according to probability mass function $\ddot{\pi}(\chi)$. Assume discretization of the range of the corresponding cumulative distribution function $\ddot{\Pi}(\chi)$, assume uniform discretization, assume using instantaneous code and assume receiving unordered (i.e., randomly ordered) codewords. Then optimal description of the probability distribution $\ddot{\Pi}(\chi)$ takes*

$$\frac{1}{\delta} \times \mathcal{E}(\ddot{\pi}(\chi)) \text{ bits},$$

where

$$\delta := \gcd\left(\{\ddot{\pi}(\chi_i)\}_1^{|\mathrm{supp}(\ddot{\pi})|}\right).$$

Proof. See §F.3. □

Basically, this Proposition establishes a useful correspondence between the entropy of random variable χ (really, the entropy of probability mass function $\ddot{\pi}(\chi)$) and the length of the description of its probability distribution $\ddot{\Pi}(\chi)$.

The assumption about uniform discretization can be motivated by, first, economizing on the need to also describe the discretization rule (i.e., the method of allocating discretization cell widths) itself and thus confining to rules that are fixed or that can be

readily deduced from the very structure of a description sequence, and, second, by uniform prior belief on the locations of probability masses over the domain or more generally by principle of insufficient reason.

An instantaneous (or prefix) code is defined as a code system such that no codeword in it is a prefix for any other codeword in the system.

An alternative approach to description of probability distributions is taken in quantization literature (an encyclopedic overview of this literature is available in Gray and Neuhoff, 1998). State of the art procedure is to construct a finite set of reproducible probability distributions, find the element of this set that is closest in some metric to the desired distribution, and to pass on just the unique index of this closest set element. This modern type-based scheme was developed by Reznik (2010), who also proposed an asymptotically optimal algorithm of its implementation. Expanding the set of fully reproducible distributions—by increasing the density of reproduction lattice (grid)—allows for more accurate description of the distribution in focus. It is worth noting, however, that procedures implementing a classic tree-based scheme still dominate the practice and are used e.g. for data compression in ZIP and JPEG standards. In contrast to this quantization-motivated approach, which boils down to describing the range of the probability distribution in target, our approach reduces to description of its domain. This is done for the sake of simplicity, so that formalization of the description step conformed with that of the following two steps as much as possible.

Lastly, recall that our decision-making agent is endowed with a native code based on alphabet $\mathcal{A}$, which is used for all operations including transmission via the communication channel of the simplified probability distribution's desription sequence. A channel that transmits information from input random variable x into output random variable $\hat{x}$ using alphabet $\mathcal{A}$ will be denoted by $\mathcal{I}_{\mathcal{A}}(\ddot{g}(x);\ddot{h}(\hat{x}))$.

Therefore, in our example we have:

- Effective channel capacity demands per codeword [in information-theoretic terms; or equivalently in engineering terms, per one channel activation] equal

$$\begin{aligned}\mathcal{I}(\ddot{g}(x);\ddot{h}(\hat{x})) &= \mathcal{E}(\ddot{h}(\hat{x})) - \mathcal{E}(\ddot{f}(\hat{x}|x)) = \\ &= \mathcal{E}(\ddot{h}(\hat{x})) - 0 = 1.5 \text{ bits};\end{aligned} \tag{F.19}$$

- Physical channel capacity demands per codeword equal

$$\begin{aligned}\mathcal{I}_{\mathcal{A}}(\ddot{g}(x);\ddot{h}(\hat{x})) &= \mathcal{E}_{\mathcal{A}}(\ddot{h}(\hat{x})) - \mathcal{E}_{\mathcal{A}}(\ddot{f}(\hat{x}|x)) = \\ &= \mathcal{E}(\ddot{h}(\hat{x})) + \mathcal{D}\left(\ddot{h}(\hat{x}) \middle\| \frac{1}{|\mathcal{A}|}\right) - 0 = 1.5 + 4.5 = 6 \text{ bits}\end{aligned} \tag{F.20}$$

  (i.e., the length of native code's codeword).

Above, we again used the relative entropy property previously applied by the "Wrong code" theorem:

$$\mathcal{E}_{\mathcal{A}}(\ddot{h}(\hat{x})) = \mathcal{E}(\ddot{h}(\hat{x})) + \mathcal{D}\left(\ddot{h}(\hat{x}) \middle\| \frac{1}{|\mathcal{A}|}\right). \tag{F.21}$$

Sending every domain value corresponding to each discretization cell takes $\hat{n}_d$ codewords, or channel transmission operations. Aggregating, we get channel capacity demands per full code sequence (per full transmission), i.e. $\hat{n}_d \times$ (channel capacity per codeword) [equivalently, $\hat{n}_d \times$ (channel capacity per activation)]:

- Effective channel capacity demands per full code sequence [per full transmission] equal

$$\hat{n}_d \times \mathcal{I}(\ddot{g}(x); \ddot{h}(\hat{x})) = 4 \times 1.5 = 6 \text{ bits}; \tag{F.22}$$

- Physical channel capacity demands per full code sequence equal

$$\hat{n}_d \times \mathcal{I}_{\mathcal{A}}(\ddot{g}(x); \ddot{h}(\hat{x})) = 4 \times 6 = 24 \text{ bits}. \tag{F.23}$$

Thus, our desired input "`000 001 10 10 11 11 010 011`" would take

$$n_d \times \mathcal{E}(\ddot{g}(x)) = 8 \times 2.5 = 20 \text{ bits}, \tag{F.24}$$

our net input "`00 1 1 01`" takes only

$$\hat{n}_d \times \mathcal{E}(\ddot{h}(\hat{x})) = 4 \times 1.5 = 6 \text{ bits}, \tag{F.25}$$

the sent message "`001000 001100 001100 010100`" takes

$$\hat{n}_d \times \mathcal{I}(\ddot{g}(x); \ddot{h}(\hat{x})) + \text{overhead} = 4 \times 1.5 + 4 \times 4.5 = 6 + 18 = 24 \text{ bits}, \tag{F.26}$$

and net output is the same as net input in (F.25), also taking 6 bits, due to the fact that communication not exceeding channel capacity is lossless.

To sum up, it takes 24 bits to fully describe our simplified probability distribution $\ddot{h}(\hat{x})$. (Alternative state of the art procedure due to Reznik (2010) would use 20 bits instead.)

Step 4: Storage in working memory.

Next, the description of the simplified distribution is loaded into "working memory". Memory storage is a resource that will be used for performing computations in Step 5.

In our example, distinguishing between operations in optimal and native code, we have:

- Effective working memory capacity demands per codeword [per one memory writing operation]

$$\mathcal{M}(\ddot{h}(\hat{x})) = \mathrm{E}[L^*(\hat{x})] = \mathcal{E}(\ddot{h}(\hat{x})) = 1.5 \text{ bits}; \tag{F.27}$$

- Physical working memory capacity demands per codeword

$$\mathcal{M}_{\mathcal{A}}(\ddot{h}(\hat{x})) = L_{\mathcal{A}}(\hat{x}) = \mathcal{E}_{\mathcal{A}}(\ddot{h}(\hat{x})) = 6 \text{ bits}. \tag{F.28}$$

On aggregate, this makes:

- Effective working memory capacity demands per full code sequence [per full storage session]:
$$\hat{n}_d \times \mathcal{M}(\ddot{h}(\hat{x})) = 4 \times 1.5 = 6 \text{ bits;} \tag{F.29}$$
- Physical working memory capacity demands per full code sequence:
$$\hat{n}_d \times \mathcal{M}_{\mathcal{A}}(\ddot{h}(\hat{x})) = 4 \times 6 = 24 \text{ bits.} \tag{F.30}$$

Step 5: Computation of statistic.

Taking mean of $\hat{x}$ as a statistic of interest, it is computed as a partial sum of $\hat{x}_j$, each weighted by factor $1/\hat{n}_d$, for $j = 1, 2, \ldots, \hat{n}_d$. In order to conduct $\hat{n}_d$ of these summation operations, every time an $\hat{x}_j$ has to be retrieved from the working memory (costlessly, but this is WLOG) and sent via communication channel to the "arithmetic unit" (whose operation costs we abstract away from for simplicity reasons). Thus, the computation process utilizes the communication channel similarly to description process.

Effective channel capacity demands per codeword $\mathcal{I}(\ddot{h}(\hat{x}); \ddot{h}(\hat{x}))$ equal 1.5 bits, while physical channel capacity demands per codeword $\mathcal{I}_{\mathcal{A}}(\ddot{h}(\hat{x}); \ddot{h}(\hat{x}))$ equal 6 bits (note the replacement of $\ddot{g}(x)$ with $\ddot{h}(\hat{x})$ in the mutual information functionals' arguments). Effective channel capacity demands per full code sequence $\hat{n}_d \times \mathcal{I}(\ddot{h}(\hat{x}); \ddot{h}(\hat{x}))$ equal 6 bits, and physical channel capacity demands per full code sequence $\hat{n}_d \times \mathcal{I}_{\mathcal{A}}(\ddot{h}(\hat{x}); \ddot{h}(\hat{x}))$ equal 24 bits. Our net input are the same 6 bits as in (F.25), the sent message repeats 24 bits from (F.26), and net output equals net input.

Digression: In the interest of generality we should emphasize that the case when x is a continuously distributed random variable does not pose any practical difficulties (although some care must be taken nevertheless, as continuous entropy is not invariant to transformations such as change of variables). Previous formulas can be readily converted into their continuous counterpats, so in principle we can treat the presented algorithm as applicable to both cases. However, for our specific coding examples to remain valid (and/or basing on findings of neurophysiological nature), we may wish to convert a continous random variable into its discrete analog as a preliminary step.

Step 0: Quantization of continuous distribution.

Let x be a continuously distributed scalar random variable with probabiliy density function $\bar{g}(x)$,
$$x \sim \bar{g}(x), \tag{F.31}$$
as depicted in the left panel of Figure F.0*.

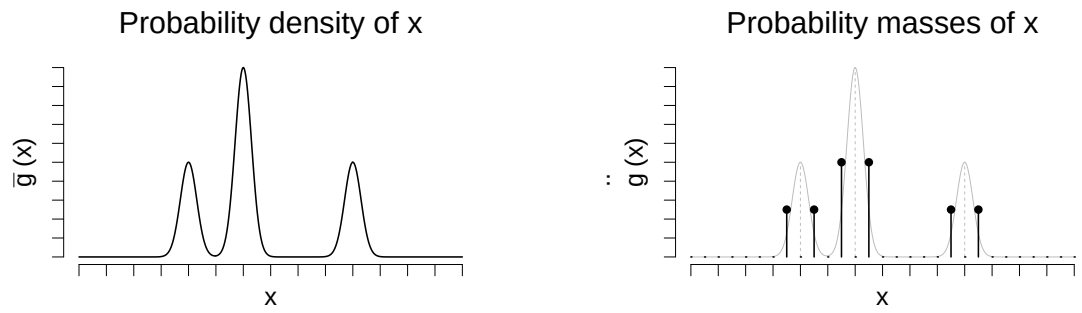


Figure F.0*: Depiction of $\bar{g}(x)$ and $\ddot{g}(x)$.

Define differential (or continuous) entropy of random variable x as

$$\mathcal{E}(\bar{g}(x)) := -\int_{\mathrm{supp}(\bar{g})} \bar{g}(x) \log \bar{g}(x)\, dx, \tag{F.32}$$

(in this case, a continuous alphabet is presumed).

Assume uniform quantization of $\mathrm{supp}(\bar{g})$ into identical cells of width Δ not optimized for particular distribution. This is optimal under fairly general conditions for high resolution quantization (i.e., "small" Δ and "large" n_q) case. As a result, we obtain quantized probability density function $\ddot{g}(x)$, defined by a sequence of equations

$$\ddot{g}(x_i) := \int_{\Delta i}^{\Delta(i+1)} \bar{g}(x)\, dx = \bar{g}(x_i)\Delta \tag{F.33}$$

for $i = 1, 2, \ldots, n_q$. Here, n_q is the number of quantization points, determined by covering (tiling) the set $\mathrm{supp}(\bar{g}(x))$ with Δ-cells:

$$n_q := \frac{m(\mathrm{supp}(\bar{g}))}{\Delta} \tag{F.34}$$

for some appropriate measure m on σ-algebra over $\mathrm{supp}(\bar{g})$. This also gives the relationship

$$|\mathrm{supp}(\ddot{g})| := n_q, \tag{F.35}$$

which we have already used earlier.

The process of quantization is visualized by the right panel of Figure F.0*. Using cell width $\Delta := 2$, we thus produced the probability mass function $\ddot{g}(x)$ from our previously discussed example.

For Riemann-integrable $\bar{g}(x)$, the following correspondence between continuous and discrete entropy holds as $\Delta \to 0$ (Cover and Thomas, 2006):

$$\mathcal{E}(\ddot{g}(x)) \simeq \mathcal{E}(\bar{g}(x)) + |\mathrm{supp}(\ddot{g})| := \mathcal{E}(\bar{g}(x)) + n_q, \tag{F.36}$$

which in our example boils down to

$$2.5 \text{ bits} \simeq \mathcal{E}(\bar{g}(x)) + 6 \text{ bits}. \tag{F.37}$$

Wrapping-up remarks: To recap, the information processing algorithm consists of the following steps:

[**0.** Quantization of continuous distribution.]

1. Simplification of discrete distribution.

2. Generating codebook.

3. Description of simplified distribution.

4. Storage in working memory.

5. Computation of statistic.

The last three steps contain potential "bottlenecks" on the way of information flow (shown in bold on Figure F.3), formally represented by the following, potentially binding, physical constraints:

- for the Description step, communication channel capacity demands $\hat{n}_d \times \mathcal{I}_{\mathcal{A}}(\ddot{g}(x); \ddot{h}(\hat{x}))$ are bounded by the available full description channel capacity $\mathcal{K}_D$,

$$\hat{n}_d \times \mathcal{I}_{\mathcal{A}}(\ddot{g}(x); \ddot{h}(\hat{x})) \leq n_{\mathcal{I}D} \times \widehat{\mathcal{I}}_{\mathcal{A}D} =: \mathcal{K}_D, \tag{F.38}$$

  where available channel capacity is formed by $n_{\mathcal{I}D}$ number of $\widehat{\mathcal{I}}_{\mathcal{A}D}$-bit wide physical communication channels;

- for the Storage step, working memory capacity demands $\hat{n}_d \times \mathcal{M}_{\mathcal{A}}(\ddot{h}(\hat{x}))$ are bounded by the available full storage memory capacity $\mathcal{K}_S$,

$$\hat{n}_d \times \mathcal{M}_{\mathcal{A}}(\ddot{h}(\hat{x})) \leq n_{\mathcal{M}} \times \widehat{\mathcal{M}}_{\mathcal{A}} =: \mathcal{K}_S, \tag{F.39}$$

  where available memory capacity is formed by $n_{\mathcal{M}}$ number of $\widehat{\mathcal{M}}_{\mathcal{A}}$-bit large physical working memory cells;

- for the Computation step, communication channel capacity demands $\hat{n}_d \times \mathcal{I}_{\mathcal{A}}(\ddot{h}(\hat{x}); \ddot{h}(\hat{x}))$ are bounded by the available full computation channel capacity $\mathcal{K}_C$,

$$\hat{n}_d \times \mathcal{I}_{\mathcal{A}}(\ddot{h}(\hat{x}); \ddot{h}(\hat{x})) \leq n_{\mathcal{I}C} \times \widehat{\mathcal{I}}_{\mathcal{A}C} =: \mathcal{K}_C, \tag{F.40}$$

  where available channel capacity is formed by $n_{\mathcal{I}C}$ number of $\widehat{\mathcal{I}}_{\mathcal{A}C}$-bit wide physical communication channels.

Taking a wider, computational complexity perspective, the above three steps can be viewed as particular examples of, respectively, communication, space and time complexity concepts (see Arora and Barak, 2009).

Utilized in the manner of the described algorithm, binding constraints on description channel capacity, on working memory capacity, or on computation channel capacity demands are operationally equivalent to each other.

Define $\mathcal{K}^*$ as the full physical capacity bound implied by the tightest constraint:

$$\mathcal{K}^* := \min\{\mathcal{K}_D,\ \mathcal{K}_S,\ \mathcal{K}_C\}. \tag{F.41}$$

(In our coding example, $\mathcal{K}^* = 24$ bits.)

Conditional on the value of $\mathcal{K}^*$, we can assume without loss of generality any one of the three steps is the physical "bottleneck" at play (essentially, this is a matter of taste).

Lastly, the same results can be obtained by replacing lossless description of the simplified distribution (Step 3) with Monte Carlo sampling from it [or with its numerical quadrature approximation]. Specifically, instead of computing the exact statistic for the simplified distribution described fully, we can compute its approximation using the simplified distribution's empirical counterpart [or using weighted qudrature nodes]. Rather than taking $\{\hat{x}_j\}_1^{\hat{n}_d}$, we can take instead a sample of $\hat{x}_j \sim \ddot{h}(\hat{x}_j)$, for $j = 1, 2, \ldots, \hat{n}_{mc}$ [or nodes $\{\hat{x}_j\}_1^{\hat{n}_{nq}}$]. The only material difference is that $\hat{n}_{mc}$ (in general, $\hat{n}_{mc} \neq \hat{n}_d$) [$\hat{n}_{nq}$, in general $\hat{n}_{nq} \neq \hat{n}_d$] would now reflect the size of Monte Carlo sample [number of quadrature nodes].

Overall, additionally taking Monte Carlo sampling [numerical quadrature approximation] as a possible replacement for description of the simplified distribution step, along with storage and computation steps, there are 5 alternative mechanisms that generate the same communication-based calculus, modulo change of interpretation and possibly redefinition of $\hat{n}_d$ parameter.

As a remark on formal notation, from a computational standpoint the presented algorithm constitutes an ancillary procedure $\mathcal{P}_{\int}$ for evaluation of the expectation operator $\mathrm{E}^h[x]$.

F.2 Vector random variable case

We are still not ready to apply the approach presented above even to the most basic task of choice between two simple lotteries, as in Figure 1. For instance, recall full physical capacity bound $\mathcal{K}^*$ defined in equation (F.41). It is important to realize that even allocating $2\mathcal{K}^*$ for processing 2 lotteries does not mean just repeating the previous algorithm 2 times.

To begin with, we need to account for the well-known result that the problem of encoding several—even independent—random variables simultaneously is not equivalent to a combination of the corresponding stand-alone problems (this is evident from the so called sphere-covering argument used in geometry, see Cover and Thomas, 2006).

Also, handling functions of two or more random variables requires some care. For $K \times K$ matrix $\boldsymbol{A}$, $\mathcal{E}(\pi(\boldsymbol{A}\boldsymbol{\chi}))$ works smoothly, but there are difficulties with non-square $\boldsymbol{A}$, i.e. with entropy of a convolution of random variables: even $\mathcal{E}(\ddot{\pi}(\sum_k \chi_k))$ is cumbersome (cumbersome enough to warrant works by Tao, 2010; Tao and Vu, 2006). Going further, for bijective function $\varphi : \mathbb{R}^K \mapsto \mathbb{R}^K$, $\mathcal{E}(\pi(\varphi(\boldsymbol{\chi})))$ works relatively straight-forwardly, but

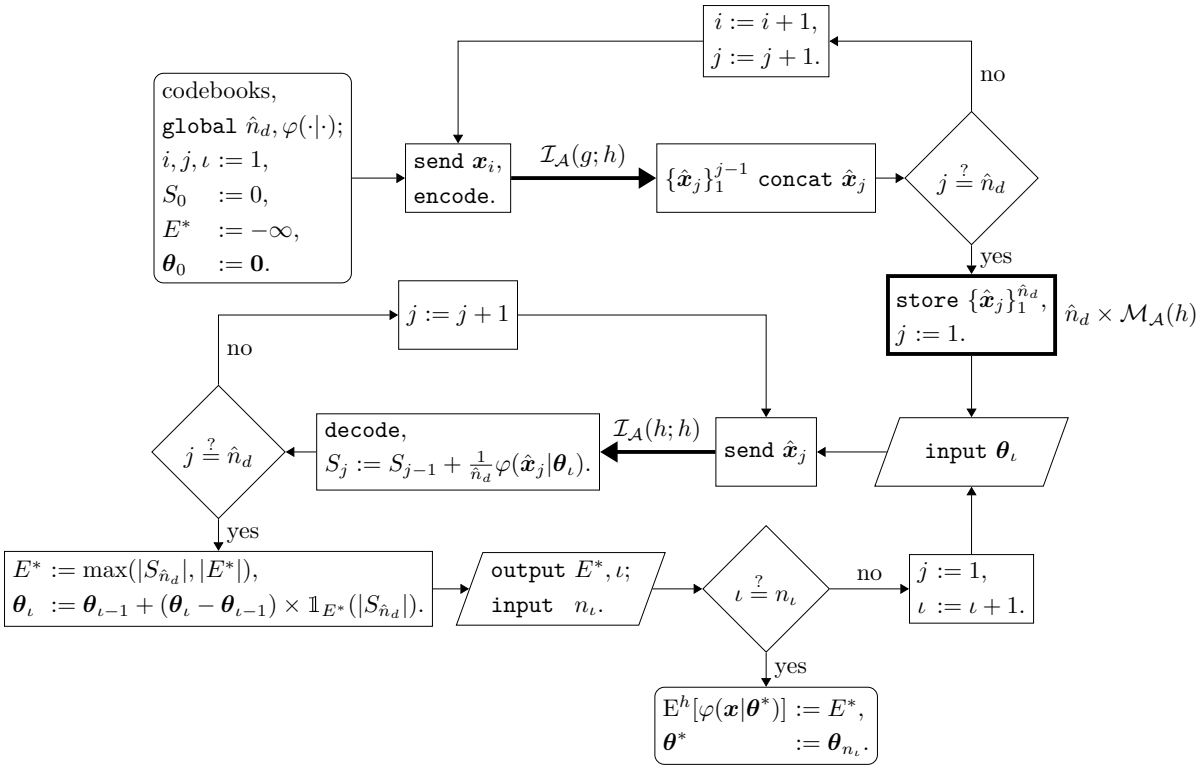


Figure F.4: Flowchart of description, storage and computation steps (vector random variable case, computation of some predetermined statistic, allowing for iterations on the latter).

result is not available in general for $\varphi : \mathbb{R}^K \mapsto \mathbb{R}^{K'}, K > K'$, i.e. for entropy of "non-linear convolution".

Therefore, we will proceed as follows. Firstly, we use the fact that $\mathcal{E}(\cdot)$ and $\mathcal{I}(\cdot\,;\cdot)$ functionals conveniently generalize to K-dimensional random vectors $\boldsymbol{\chi}$ in the desired manner.

Also, we apply transformations defined by $\mathbb{R}^K \mapsto \mathbb{R}^{K'}$ functions (including those defined by $K' \times K$ matrices) after rather than before any communication and storage processes, i.e. in the Computation rather than Description step.

Lastly, optimization (i.e., maximizing $\varphi(\boldsymbol{\chi})$ or equating it to 0) is executed as several repetitions of the Computation step iterating on function $\varphi(\cdot|\boldsymbol{\theta})$ for different choices of parameter vector $\boldsymbol{\theta}$ (e.g., number of iterations $n_\iota = 2$ for binary choice from our motivating example, $n_\iota = n$ for weighted sums on an n-points grid).

The generalized algorithm is presented in the flowchart on Figure F.4. It still deals with a simple lottery, but now the lottery is not scalar- but vector-valued, and the computed statistic is not just the average but any parameterized function, whose parameters are allowed to vary in the course of an iterative optimization procedure.

Notice, the optimization process here is constrained by available capacity, but such a constraint is by construction invariant to specific choices of parameter $\boldsymbol{\theta}$ or specification of $\varphi(\cdot|\cdot)$. This will allow us to segregate the actual decision-making problem from background

problem of the optimal utilization of available capacity, and to solve them separately.

Finally, note that from a computational standpoint, now the algorithm solves a given optimization problem using the presented above (ancillary) expectation operator evaluation procedure $\mathcal{P}_{\int}$ at each iteration of the optimization procedure.

Capacity accounting: Next, let us carefully consider the issue of communication channel (or storage memory) capacity accounting.

Redefine the full physical capacity bound implied by the tightest (per iteration, in case of computation step) constraint $\mathcal{K}^*$ more generally as:

$$\mathcal{K}^* := \min\{\mathcal{K}_D,\ \mathcal{K}_S,\ \mathcal{K}_C \,/\, n_\iota\}. \tag{F.42}$$

(For instance, if $n_\iota = 2$, to be able to use in the full procedure 6 bits per codeword for each of 4 discretization cells as in our coding example requires: $\mathcal{K}_D \geq 24$, $\mathcal{K}_S \geq 24$, and $\mathcal{K}_C \geq 48$.)

Without loss of generality, assume the binding constraint is the computation step:

$$\mathcal{K}^* = \mathcal{K}_C \,/\, n_\iota, \tag{F.43}$$

while demands for full per iteration computation channel capacity yield (in the string of equalities below, we are moving backwards along the algorithm's path):

$$\begin{aligned}
\mathcal{K}_C \,/\, n_\iota = \mathcal{K}^* &= \\
&= \hat{n}_d \times \mathcal{I}_{\mathcal{A}}(\ddot{h}(\hat{\boldsymbol{x}}); \ddot{h}(\hat{\boldsymbol{x}})) = \\
&= \hat{n}_d \times \mathcal{M}_{\mathcal{A}}(\ddot{h}(\hat{\boldsymbol{x}})) = \\
&= \hat{n}_d \times \mathcal{I}_{\mathcal{A}}(\ddot{g}(\boldsymbol{x}); \ddot{h}(\hat{\boldsymbol{x}})) = \\
&= \hat{n}_d \times \mathcal{I}(\ddot{g}(\boldsymbol{x}); \ddot{h}(\hat{\boldsymbol{x}})) + \text{overhead}.
\end{aligned} \tag{F.44}$$

Rearranging,

$$\mathcal{I}(\ddot{g}(\boldsymbol{x}); \ddot{h}(\hat{\boldsymbol{x}})) = \frac{\mathcal{K}_C}{\hat{n}_d \times n_\iota} - \frac{\text{overhead}}{\hat{n}_d} =: \kappa, \tag{F.45}$$

where κ denotes effective capacity bound per codeword (binding, as we assumed above).

Clearly, effective capacity κ is not equivalent to available full computation channel capacity $\mathcal{K}_C$, or more generally to available full physical capacity $\mathcal{K}^*$. It might, say, fall not just because of reduction in computation channel capacity, but also when using more discretization points and more iterations, as well as due to larger per-codeword overhead. This demonstrates why effective capacity κ, measured as implied information processing capacity constraint's bound, differs substantially from full physical capacity $\mathcal{K}^*$ (see the main text for further elaboration).

In general, effective capacity κ bounds the tightest constraint from above (also expanding the mutual information functional below):

$$\kappa \geq \mathcal{I}(\ddot{g}(\boldsymbol{x}); \ddot{h}(\hat{\boldsymbol{x}})) = \mathcal{E}(\ddot{g}(\boldsymbol{x})) + \mathcal{E}(\ddot{h}(\hat{\boldsymbol{x}})) - \mathcal{E}(\ddot{g}(\boldsymbol{x}), \ddot{h}(\hat{\boldsymbol{x}})) =: \mathcal{E}(\ddot{g}(\boldsymbol{x})) + \mathcal{E}(\ddot{h}(\hat{\boldsymbol{x}})) - \mathcal{E}(\ddot{f}(\boldsymbol{x}, \hat{\boldsymbol{x}})) =$$
$$= -\sum_{i=1}^{n_q} \ddot{g}(\boldsymbol{x}_i) \log \ddot{g}(\boldsymbol{x}_i) - \sum_{j=1}^{\hat{n}_q} \ddot{h}(\hat{\boldsymbol{x}}_j) \log \ddot{h}(\hat{\boldsymbol{x}}_j) + \sum_{i=1}^{n_q} \sum_{j=1}^{\hat{n}_q} \ddot{f}(\boldsymbol{x}_i, \hat{\boldsymbol{x}}_j) \log \ddot{f}(\boldsymbol{x}_i, \hat{\boldsymbol{x}}_j), \tag{F.46}$$

with its continuous counterpart being

$$\kappa \geq \mathcal{I}(\bar{g}(\boldsymbol{x}); \bar{h}(\hat{\boldsymbol{x}})) = \mathcal{E}(\bar{g}(\boldsymbol{x})) + \mathcal{E}(\bar{h}(\hat{\boldsymbol{x}})) - \mathcal{E}(\bar{g}(\boldsymbol{x}), \bar{h}(\hat{\boldsymbol{x}})) =: \mathcal{E}(\bar{g}(\boldsymbol{x})) + \mathcal{E}(\bar{h}(\hat{\boldsymbol{x}})) - \mathcal{E}(\bar{f}(\boldsymbol{x}, \hat{\boldsymbol{x}})) =$$
$$= -\int_{\mathrm{supp}(\bar{g})} \bar{g}(\boldsymbol{x}) \log \bar{g}(\boldsymbol{x})\, d\boldsymbol{x} - \int_{\mathrm{supp}(\bar{h})} \bar{h}(\hat{\boldsymbol{x}}) \log \bar{h}(\hat{\boldsymbol{x}})\, d\hat{\boldsymbol{x}} +$$
$$+ \int_{\mathrm{supp}(\bar{g})} \int_{\mathrm{supp}(\bar{h})} \bar{f}(\boldsymbol{x}, \hat{\boldsymbol{x}}) \log \bar{f}(\boldsymbol{x}, \hat{\boldsymbol{x}})\, d\hat{\boldsymbol{x}}\, d\boldsymbol{x}. \tag{F.47}$$

Note that in the main text, we focus on continuously distributed random variables exclusively (unless stated otherwise), so differentiation between the "double-dot" and "bar" probability distributions is irrelevant and we refrain from using these accents.

F.3 Proof of Proposition F.1, with additional comments

Proof. The range of CDF $\ddot{\Pi}(\chi)$ is exhaustively and efficiently (i.e. without intersections) tiled by δ-cells. Discretization of the range (once $n_d := 1/\delta$ is known) leaves only the domain of $\ddot{\Pi}(\chi)$ to describe.

Description of the CDF domain is done by specifying a sequence of domain values corresponding to each discretization cell, that is $\{\chi_i\}_1^{n_d}$ for $\chi_i = \ddot{\Pi}^{-1}(i\delta)$. Expected length of respective optimal codewords using instantaneous code equals $\mathcal{E}(\ddot{\pi}(\chi))$, which pins down the lower bound for average length of codewords in the description sequence. This can be shown by adjusting the argument about optimal instantaneous code and the bound on its expected length (e.g., see Cover and Thomas, 2006).

Specifically, denoting by n_i the number of discretization points per each quantization point i, we wish to minimize total description length:

$$\min_{\{L(\chi_i)\}_{i=1}^{n}} \sum_{i=1}^{n_d} L(\chi_i) = \sum_{i=1}^{n_q} n_i L(\chi_i) = n_d \sum_{i=1}^{n_q} \frac{n_i}{n_d} L(\chi_i) \tag{F.48}$$

subject to Kraft's inequality (that necessarily holds for any instantaneous code and guarantees the existence of such code)

$$\sum_{i=1}^{n_q} 2^{-L(\chi_i)} \leq 1$$

which yields

$$L^*(\chi_i) := -\log \frac{n_i}{n_d}.$$

At the same time, in the process of discretization we allocated n_i-s according to the rule

$$n_i := \frac{\ddot{\Pi}(\chi_i) - \ddot{\Pi}(\chi_{i-1})}{\delta} =: \frac{\ddot{\pi}(\chi_i)}{\delta},$$

hence

$$\frac{n_i}{n_d} = \ddot{\pi}(\chi_i).$$

Substituting into objective function (F.48) produces

$$\sum_{i=1}^{n_d} L^*(\chi_i) = -n_d \sum_{i=1}^{n_q} \frac{n_i}{n_d} \log \frac{n_i}{n_d} = -n_d \sum_{i=1}^{n_q} \ddot{\pi}(\chi) \log \ddot{\pi}(\chi) =: n_d \, \mathcal{E}(\ddot{\pi}(\chi)),$$

which is the claimed result.

Lastly, receiving χ_i as a sequence ordered, say, from lower to higher cumulative probability mass $\ddot{\Pi}(\chi_i)$ would have allowed shorter codeword lengths by ruling out the subset of the support corresponding to $\chi_{i'} < \chi_i$ with each new χ_i received. This possibility is assumed away.

□

Some additional comments are warranted. Note that minimizing $\sum_{i=1}^{n_d} L(\chi_i) = \sum_{i=1}^{n_q} n_i L(\chi_i)$ (i.e., number-of-occurences-weighted sum of codeword lengths) is equivalent to minimizing $\mathrm{E}[L(\chi)] = \sum_{i=1}^{n_q} \ddot{\pi}(\chi_i) L(\chi_i)$ (probability-weighted sum of lengths), hence the code assignment we obtain can also be written as

$$L^*(\chi_i) = -\log \ddot{\pi}(\chi_i),$$

which is a classic result for optimal instantaneous codes, and in expectation achieves the theoretical lower bound:

$$\mathrm{E}[L^*(\chi_i)] = -\sum_{i=1}^{n_q} \ddot{\pi}(\chi_i) \log \ddot{\pi}(\chi_i) =: \mathcal{E}(\ddot{\pi}(\chi)).$$

In some cases suggested theoretically optimal codeword lengths may be achievable only asymptotically, but using Shannon code assignment $L(\chi_i) := \lceil -\log \ddot{\pi}(\chi_i) \rceil$, the following bound is always achievable in practice:

$$\frac{1}{\delta} \times (\mathcal{E}(\ddot{\pi}(\chi)) + 1) \text{ bits.}$$

References for Algorithm

[1] Arora, Sanjeev and Boaz Barak. (2009) *Computational Complexity: A Modern Approach*, New York, NY: Cambridge University Press.

[2] Cover, Thomas M. and Joy A. Thomas. (2006) *Elements of Information Theory*, 2nd ed., New York: Wiley-Interscience.

[3] Gray, Robert M. and David L. Neuhoff. (1998) "Quantization." *IEEE Transactions on Information Theory*, 44(6): 2325–2383.

[4] MacKay, David J. C. (2003) *Information Theory, Inference, and Learning Algorithms*, Cambridge University Press.

[5] Reznik, Yuriy A. (2010) "Quantization of Discrete Probability Distributions." Manuscript.

[6] Tao, Terence. (2010) "Sumset and Inverse Sumset Theory for Shannon Entropy." *Combinatorics, Probability and Computing*, 19: 603–639.

[7] Tao, Terence and Van Vu. (2006) "Entropy Methods." Manuscript.

G Algorithm for decision-making under risk: A toy primer

Consider a fruit-tree bearing some random number of fruits every year, with known probability distribution guiding possible fruit harvests: say, the harvest may be 7 or 9 fruits each with probability $1/8$, 11 or 13 fruits with respective probabilities $1/4$, and 19 or 21 fruits with probabilities $1/8$. (The corresponding tables and plots are provided above in §F.)

There is also a potential investor. Essentially, he/she would first wish to assess the value of tree as a capital good. For instance, how productive the tree is on average: say, its mean harvest is $7\times 1/8+9\times 1/8+11\times 1/4+\ldots+21\times 1/8 = 13$ fruits; perhaps investor also cares about variance of the harvest: the latter can be calculated to be 20 here. Secondly, he would then make the decision about buying or selling the tree on the market: say, deciding to buy 3 trees.

Operationally, this involves sending a message, i.e., a sequence of signals, describing the above probability disribution from one, perceptive part of investor's brain to another, calculating part, and computing the relevant statistic of interest. However, if the probability distribution is "too complex" given investor's information processing limitations (we can think of them as individual "bandwidth"), the above procedure becomes infeasible and has to be modified.

In such a case the distribution is "simplified": investor chooses another probability distribution that roughly approximates the original one but possesses lower "complexity", and uses this subjective simplified distribution in computations of the value of the tree. Say, calculating the average harvest as $8\times 1/4+12\times 1/2+20\times 1/4 = 13$ fruits, also relying on this new distribution in calculating the variance of just 19. Then, he decides how many trees to buy or sell: the decision is likely to be affected and differ from preceding 3 trees.

Moreover, investor may wish to bias the subjective distribution downward and calculate the average as $7\times 1/4+11\times 1/2+19\times 1/4 = 12$ fruits, with the corresponding variance calculation of 19 still; and then choosing to buy, say, 3 trees again, as lower productivity offsets lower risk.[1]

Formally, the investor here solves the following 2-period consumption-investment problem (the notation below is the same as in the main text, so specific details are omitted):

$$\max_{\{C_t,q_t\}}\{u(C_t)+\beta \mathrm{E}_t^h\left[u(C_{t+1})\right]\} = \{u(C_t)+\beta\int_{\mathbb{R}_+} u(C_{t+1})\,h(\hat{D}_{t+1})\,d\hat{D}_{t+1}\}$$

[1] Taking this toy example even further, investor's identical (i.e., having the same information processing limits) twin employed outside of investment business—perhaps as an econometrician in academia—would likely stick to the unbiased average of 13 fruits and think that buying, say, 4 trees would have been a more sensible decision, being rather puzzled by his brother's cautious investment choice.

subject to budget constraints

$$C_t + P_t q_t = (P_t + D_t) q_{t-1} =: W_t,$$
$$C_{t+1} = \hat{D}_{t+1} q_t =: \hat{W}_{t+1}$$

with q_{t-1} and D_t given; and also where

$$h(\hat{D}_{t+1}) := \int_{\text{supp}(g)} f(D_{t+1}, \hat{D}_{t+1})\, dD_{t+1},$$
$$f(D_{t+1}, \hat{D}_{t+1}) := \arg \left\{ \min_{f(\cdot,\cdot)} \mathrm{E}^f \left[|u(C(D_{t+1})) - u(C(\hat{D}_{t+1}))|^2 \right] \text{ s.t. } \mathcal{I}(g(D_{t+1}); h(\hat{D}_{t+1})) \leq \kappa \right\},$$
$$g(D_{t+1}) \text{ is given;}$$

plus the necessary technical restrictions. Basically, he chooses the controls as if facing a proxy variable $\hat{D}_{t+1}$ (i.e., D_{t+1} with some noise).

In the context of our example above, canonical rational inattention theory (Sims, 2003, 2006) would essentially envisage a consumer receving a nature's signal that informs about the fruit harvest which has just ripened (e.g., that this turned out to be a bad year, yielding only 7 or 9 fruits with probability $1/2$ each). The consumer will then decide how many fruits to eat (say, 5), and how many to commit selling on the market for reinvestment into trees (say, the rest). In anticipation, before the harvest ripens, the consumer chooses the probability distribution of such signal in every contingency taking into account the complexity of the fruit harvest's distribution (i.e., an optimal information acquisition strategy). If the latter is too high, the consumer needs to restrict the accuracy of the signal he is able to receive, thus behaving rationally inattentively. Clearly, this is a different problem from the one presented above.

Formally, the consumer there solves the following 2-period consumption-investment problem:

$$\max_{f(\cdot,\cdot)} \mathrm{E}_t^f \left[u(C_{t+1}) + \beta u(C_{t+2}) \right] =$$
$$= \int_{\mathbb{R}^3_+} \left[u(C_{t+1}) + \beta u(C_{t+2}) \right] f(\{C_{t+1}, q_{t+1}\}, D_{t+1})\, d\{C_{t+1}, q_{t+1}\}\, dD_{t+1}$$

subject to budget constraints

$$C_{t+1} + P_{t+1} q_{t+1} = (P_{t+1} + D_{t+1}) q_t =: W_{t+1},$$
$$C_{t+2} = D_{t+2} q_{t+1} =: W_{t+2}$$

with q_t and D_{t+2} given; and also subject to

$$\mathcal{I}(g(D_{t+1}); h(\{C_{t+1}, q_{t+1}\})) \leq \kappa,$$
$$h(\{C_{t+1}, q_{t+1}\}) := \int_{\text{supp}(g)} f(\{C_{t+1}, q_{t+1}\}, D_{t+1})\, dD_{t+1},$$
$$g(D_{t+1}) \text{ is given;}$$

plus the necessary technical restrictions. Basically, he ends up choosing the controls while observing D_{t+1} with some noise (i.e., a proxy variable $\hat{D}_{t+1}$).

References for Algorithm primer

[1] Sims, Christopher A. (2003) "Implications of Rational Inattention." *Journal of Monetary Economics*, 50: 665–690.

[2] Sims, Christopher A. (2006) "Rational Inattention: A Research Agenda." Manuscript.

H Neurofoundations

This work is explicitly grounded on existing knowledge about the human brain and cognition, however patchy that knowledge is. Unfortunately, a disproportionate amount of current understanding is based on non-human evidence that can be gathered using not just relatively crude functional magnetic resonance imaging, but also high-resolution invasive methods. Naturally, vision takes a key spot by virtue of being one of the most complex neural systems observed in non-humans. To put things into perspective, primary visual cortex is one of the most intensively studied region of the brain, yet according to the estimate of Olshausen and Field (2005) we understand only 15% of its function.

Our framework is built not at the level of individual neurons, but rather functional subsystems (really, populations of neurons); and it is most directly related to computational neuroscience, less so to cognitive or behavioral neuroscience/psychology, and only indirectly to cellular neuroscience or neurophysiology (omitting altogether the discussion of specific anatomical regions of the central nervous system, chemical neuromodulators, etc.). However, Eliasmith and Anderson (2003), Eliasmith (2013) as well as Rao (2010) show how such computational model can be implemented with neurologically plausible apparatus.

Functional subsystems included in our framework are linked in a centralized serial (hierarchical, sequential) order, but each such module allows for distributed and/or parallel processing within the subsystem. Indeed, it is a well established fact that human brain utilizes diverse forms of computation, e.g. a hierarchical organization has been found in the visual cortex and distributed architecture seems to be utilized in the memory system (but even these example systems are not fully specialized and exhibit different types of processing); see Squire et al. (2008) for details.

We incorporate some key neuroscientific elements and concepts:

- Limited capacity communication channel, e.g. the optic nerve may be thought of as a limited-capacity channel (Burton, 2000).

- Working memory, defined as a limited capacity system that temporarily maintains and stores information to support human thought processes by providing an interface between perception, long-term memory and action (Baddeley, 2003); a good primer on the operations with working memory necessary for on-line computations or solving more abstract Tower of Hanoi problem and Raven's Progressive Matrices test are provided by Smith and Jonides (2003).

- Bottleneck, arising when a downstream subsystem (output) has tighter capacity limits that an upstream one (input), for instance in communication channels serving auditory and visual processing (Baddeley et al., 1997; Zhaoping, 2006) as well as due to working memory restrictions (Reynolds et al., 2008); ultimately, such

physical capacity limits reflect energy costs and a pursuit for metabolic efficiency (see Laughlin et al., 2000).

- "Native" (default, fixed, internal) code, which particular brain subsystem uses for its operations and which is a given (e.g., see Campbell and Epp, 2005); it is optimal with respect to *both* physiological characteristics of the brain subsystem and the "average" stimulus it encodes (i.e., in environmental sense), but in general it is not efficient (in the sense of Shannon) for the "average" stimulus, and channel capacity related arguments should account for this fact (as Laming, 2010, puts it, "the human subject can be viewed as a communication system with fixed 'coding'," see his paper for more details); taking a different perspective, as long as "native" codebook reflects prior beliefs about the stimulus to be encoded, then this prior should arguably be understood in the sense of empirical Bayes methodology.

- Efficient code, which is optimal with respect to the "average" stimulus the corresponding brain subsystem encodes (i.e., efficient in Shannon sense);[2] though it may not be optimal for some subset—say, most current—stimuli, an ad hoc improvement and additional reduction in capacity requirements of the brain subsystem involved can still be achieved, albeit at a cost to auxiliary subsystems (Bor et al., 2003); ensembles and summary statistics, as parts of scene processing, are also the concepts relevant here (e.g., see Cohen et al., 2016).

- Interface for encoding and decoding, which in information-theoretic sense are abstract notions defined as a map ("codebook") from input, source alphabet to output, target alphabet and, respectively, its reverse map (Cover and Thomas, 2006; Campbell and Epp, 2005; also see Hertz et al., 1991), while in neurosience they take concrete forms of mapping an input (say, sensory) stimulus to a neural response in terms of spike sequence and, respectively, its inverse (Squire et al., 2008; Dayan and Abbott, 2001; Doya et al., 2007); with Doya et al. (2007), Dayan and Abbott (2001), as well as Borst and Theunissen (1999) providing a bridge between abstract information-theoretic and applied neural treatments of these concepts.

- Value (of some reward) and value function (recognizing its dynamic recursive forward-looking aspects) are established notions in neuroscience, e.g. see Rangel et al. (2008), as well as Schultz et al. (1997), McClure et al. (2004) and Doya (2008) for laboratory evidence related to rewards; value fiunction is understood in terms similar to classical dynamic programming, e.g. see Lee et al. (2012) and Doya (2007) for short reviews, as well as Dayan and Abbott (2001) or Glimcher et al. (2008)

[2]Unfortunately, under the influence of the so called "efficient code hypothesis" (Barlow, 1961), the convention in neuroscience and psychology does not discriminate between the concepts of "native" (fixed) and efficient codes, usually assuming that the latter applies. Laming (2010) offers a comprehensive critique.

for textbook treatment, with Bertsekas and Tsitsiklis (1996) as well as Sutton and Barto (1998) offering theoretical underpinnings.

- Probabilistic objects, such as probability distributions (including prior or posterior), risk (variance or entropy) and ambiguity, or likelihoods and mathematical expectations (e.g., expected value of the reward) are accepted components of brain activity, see Schultz et al. (2008) for an excellent review, as well as Ma and Jazayeri (2014), Doya (2008), Platt and Huettel (2008), Rushworth and Behrens (2008), Knill and Pouget (2004), Pouget et al. (2013), Yang and Shadlen (2007), Vilares et al. (2012), Barber et al. (2003).

- Prediction errors, which is the discrepancy between new information and some "reference frame" (some predicted value or prior distribution), is another important concept, for instance used explicitly or implicitly in information coding (in the form of "predictive coding", which amounts to encoding and transmitting only residual errors in prediction, see Huang and Rao, 2011, Spratling, 2012, Summerfield and Egner, 2009, Zhaoping, 2006, Dayan and Abbott, 2001) or learning (in the form of "reward prediction errors" in reinforcement learning, see Lee et al., 2012, Doya, 2007, Dayan and Abbott, 2001, Schultz et al., 1997, Daw et al., 2011); also see below with regard to the process of learning.[3]

- Numerical/quantity information is treated here in correspondence with that in mathematics, and although by no means taken for granted, there actually is neuroscientific evidence that in important issues this may be a valid approach; see Dehaene(2009), Nieder and Dehaene (2009) and Nieder (2005) for reviews, specific instances are (i) discrete (numerocity, e.g. number of items) and continuous (extent, e.g. length) quantities (they are supported by functionally overlapping populations of neurons bolstering the idea of abstract quantity, a generalized magnitude system in the brain, as shown by Tudusciuc and Nieder, 2007, 2009), (ii) symbolic signs (e.g., Arabic digit "3" or written word "three") and analog iconic signs (e.g., sets of dots) representing quantities (they are supported by the same neural populations expressing some shared abstract code, see Piazza et al., 2007) as well as (iii) correspondence between concrete quantities and abstract formal symbols ("variables") used in computations (this is a natural feature of neural networks, e.g. see Dehaene and Changeux, 2003, for a simple neural model primer), hence we freely interchange within each pair as we often implicitly do in applied mathematics (and in line with common practice in computational neuroscience, see Dayan and Abbott, 2001).

[3]Predictive coding may be playing an important role in implementing Bayesian inference (i.e., only differences from prior are encoded in the process of updating some relevant posterior distribution), for example see Huang and Rao (2011), also see Kwisthout and van Rooij (2013). Reward prediction errors as a concept are consistent with the mechanism of reference-dependent valuation, e.g. see Kahneman and Tversky (1979, 1992).

The framework also utilizes commonly accepted neural mechamisms and processes:

- Non-linear computation and function approximation potentially performed by neural networks as well as machinery for training neural network parameters that provide great flexibility in terms of plausible neural mechanisms available to execute—and explain—human behavior (Hertz et al., 1991; Dayan and Abbott, 2001; Cybenko, 1989; also see Bullinaria, 2000, regarding the benefits and pitfalls of such flexibility).[4]

- Learning, which in neuroscience boils down to adjusting neural network synapses/connection weights, is usually differentiated into unsupervised learning and supervised (most commonly, reinforcement learning based on minimizing (reward) prediction errors), and may apply to learning entropy-reducing (redundancy-reducing in the neuroscientific literature) approximations, auxiliary transformations, but most importantly the model itself, including the value function; Hertz et al. (1991), Dayan and Abbott (2001) focus on the relevant mechanisms in artificial, while Lee et al. (2012), Rangel et al. (2008), Doya (2007) discuss such mechanisms from the perpective of biological neural networks; by way of clarification, (i) we focus on networks that have converged to the solution (i.e., solution to our dynamic programming and informational problems are already known—this may alternatively be interpreted as if fixed time and mental costs have been incurred at the outset— in abstract symbolic form, and subsequently any new parameter values such as variance-covariance matrix are just "plugged into" the optimal solution), and (ii) so called active learning and evidence accumulation is beyond the scope of our work and is assumed away.

- Tranformations are applied to input information in order to reduce redundancy (decompositions, decorrelation) and/or dimensionality/entropy (filters); e.g. in visual processing the neuroscientifically accepted procedures are principal component analysis for Gaussian and independent component analysis or wavelet-like basis function representation for non-Gaussian heavy-tailed inputs (Dayan and Abbott, 2001; Simoncelli and Olshausen, 2001; Zhaoping, 2006; Doi et al., 2012), with a number of studies demonstrating emergence of receptive/projective fields similar to those observed in animal cortical areas and at the same time consistent with the result of implementing the above transformation procedures when neural models are trained on natural images (for example, see Olshausen and Field, 1996, or Spratling, 2012); such redundancy/dimension-reducing transformations are a form of efficient coding implemented in terms of structural ("causal" in neuroscience) higher-level objects (in case of wavelet-like transforms the conventional term in neuroscience is "sparse coding"), and constitute intermediate stages of predictive coding proce-

[4]Hertz et al. (1991) shows how even the simplest one-layer feed-forward artificial neural network containing a finite number of neurons can perform PCA.

dures, for more details see Huang and Rao (2011), Zhaoping (2006), Dayan and Abbott (2001).

- An important instance of such transformations is coordinate transformation and subsequent adjustment of the mean (as a form of "compensation") stated in Propositions 4–5 as well as C.1, which effectively reduces to linear rotation and translation operations; it could conceivably be implemented by a mechanism similar to the one responsible for maintaining organism's internal image of the world in spatial cognition, i.e. determination of the head-centered coordinates of some target given retinal coordinates that must account for any eye movement, which is evidently performed (once initial non-linear decomposition has been computed) online in real time via simple linear operations (see Pouget and Sejnowski, 1997, or Dayan and Abbott, 2001); also, this mechanism may be related to neuroscientific findings that in primates' brains expected (mean) reward is clearly discriminated from reward uncertainty, i.e. measures of its risk (variance/entropy) and ambiguity (see Schultz et al., 2008, for a survey).

- Probabilistic computations, required to operate with probabilistic objects, seem to be one of the supported animal brain processes; literature includes examples of manipulations with probabilities/probability densities, priors and posteriors (Barber et al., 2003; Vilares et al., 2012; Kepecs and Mainen, 2012; Ma, 2012) as well as with likelihood functions and their ratios (Yang and Shadlen, 2007), performing integration/marginalization of joint probability densities (Beck et al., 2011), optimally combining several models (O'Reilly et al., 2013; Knill and Pouget, 2004), model inversion (Dayan and Abbott, 2001; Botvinick and Toussaint, 2012), implementing Monte Carlo sampling (Griffiths et al., 2012; Buesing et al., 2011; Hoyer and Hyvärinen, 2003) and stochastic mental simulations (Battaglia et al., 2013), etc. (for a broader overview, see Doya et al., 2007; Pouget et al., 2013; Dayan and Daw, 2008; Ma and Jazayeri, 2014).

- Recursive processing of information, which is necessary for iterative optimization (as well as for evaluating a menu of available options in serial rather than parallel manner within each optimizing iteration), is a feasible neural mechanism as a number of closed loops passing across a sequence of different brain areas has already been identified, according to Miller and Wallis (2008), Doya and Kimura (2008), Doya (2007), Daw et al. (2005) and Tanaka et al. (2004); this should not be confused with the standard feedforward, recurrent and feedback connections (though the role of feedback connections being less well understood) within the same area/between functionally similar areas of the brain, e.g. see Dayan and Abbott (2001) as well as Zhaoping (2006).

The machinery comprising the above neuroscientific elements and processes resonates

strongly with the proposals of Chris Eliasmith's group that underlie the functional architecture and implementation principles of the large-scale computational prototype of human brain constructed and reported in Eliasmith et al. (2012), with (some aspects of) the general approach of the neural model for decision making under uncertainty by Rao (2010), with the "levels of understanding" framework discussed and updated in Poggio (2012), with the theoretical treatment of "early vision" in Zhaoping (2006), among others.

Note that neuroscience often relies on Bayesian formalism for dealing with probabilistic material (e.g., see Doya et al., 2007; Pouget et al., 2013; Ma and Jazayeri, 2014; Dayan and Daw, 2008); while information theory uses both classical and Bayesian approaches (Cover and Thomas, 2006, is inclined to the former, but MacKay, 2003, emphasizes the latter); for the sake of simpler exposition, we follow the steps of Cover and Thomas (2006) and do not adopt Bayesian formalism explicitly, but our approach is reconciled with the Bayesian one by restricting ourselves to uninformative flat priors and ever-recurrent informational dynamics that precludes learning/updating. Additionally, note that we abstract away from neuron noise, which may play an important role in neural systems (Simoncelli and Olshausen, 2001; Eliasmith and Anderson, 2003; Cordes et al., 2007; Pouget et al., 2013; Ma and Jazayeri, 2014; though also see the sampling argument of Hoyer and Hyvarinen, 2003).

Further supporting details organized in a more systematic book format can be found in Squire et al. (2008), Dayan and Abbott (2001), Doya et al. (2007) that particularly emphasizes Bayesian approach, Baddeley et al. (2000) that stresses information-theoretic foundations, and Glimcher et al. (2008) that focuses on economic decision-making.

References for Neurofoundations

[1] Baddeley, Roland, L. F. Abbott, Michael C.A. Booth, Frank Sengpiel, Tobe Freeman, Edward A. Wakeman and Edmund T. Rolls. (1997) "Responses of neurons in primary and inferior temporal visual cortices to natural scenes." *Proceedings of the Royal society B*, 264(1389): 1775–1783.

[2] Baddeley, R., P. Hancock and P. Földiák (eds.). (2000) *Information Theory and the Brain*, New York, NY: Cambridge University Press.

[3] Barber, M. J., J.W. Clark and C. H. Anderson. (2003) "Neural Representation of Probabilistic Information." *Neural Computation*, 15: 1843–1864.

[4] Barlow, Horace B. (1961) "Possible Principles Underlying the Transformation of Sensory Messages." In: WA Rosenblith (ed.), *Sensory Communication*, pages 217–234, Cambridge, MA: MIT Press.

[5] Battaglia, Peter W., Jessica B. Hamrick and Joshua B. Tenenbaum. (2013) "Simulation as an Engine of Physical Scene Understanding." *Proceedings of the National Academy of Sciences*, 110(45): 18327–18332.

[6] Beck, Jeffrey M., Peter E. Latham and Alexandre Pouget. (2011) "Marginalization in Neural Circuits with Divisive Normalization." *Journal of Neuroscience*, 31(43):15310–15319.

[7] Bertsekas, Dimitri P. and John N. Tsitsiklis. (1996) *Neuro-Dynamic Programming*, Belmont, MA: Athena Scientific.

[8] Bor, Daniel, John Duncan, Richard J. Wiseman and Adrian M. Owen. (2003) "Encoding Strategies Dissociate Prefrontal Activity from Working Memory Demand." *Neuron*, 37: 361–367.

[9] Borst, Alexander and Frederic E. Theunissen. (1999) "Information Theory and Neural Coding." *Nature Neuroscience*, 2(11): 947–957.

[10] Botvinick, Matthew and Marc Toussaint. (2012) "Planning as Inference." *Trends in Cognitive Sciences*, 16(10): 485–488.

[11] Buesing, Lars, Johannes Bill, Bernhard Nessler and Wolfgang Maass. (2011) "Neural Dynamics as Sampling: A Model for Stochastic Computation in Recurrent Networks of Spiking Neurons." *PLoS Computational Biology*, 7(11): 1–22.

[12] Bullinaria, John A. (2000) "Free Gifts from Connectionist Modelling." In: R. Baddeley, P. Hancock and P. Foldiak (eds.), *Information Theory and the Brain*, pages 221–240, New York, NY: Cambridge University Press.

[13] Burton, Brian G. (2000) "Problems and Solutions in Early Visual Processing." In: R. Baddeley, P. Hancock and P. Foldiak (eds.), *Information Theory and the Brain*, pages 25–40, New York, NY: Cambridge University Press.

[14] Campbell, Jamie I. D. and Lynette J. Epp. (2005) "Architectures for Arithmetic." In: Jamie I.D. Campbell (ed.), *Handbook of Mathematical Cognition*, pages 347–360, Psychology Press.

[15] Cohen, Michael A., Daniel C. Dennett and Nancy Kanwisher. (2016) "What is the Bandwidth of Perceptual Experience?" *Trends in Cognitive Sciences*, 20(5): 324–335.

[16] Cordes, Sara, C. R. Gallistel, Rochel Gelman and Peter Latham. (2007) "Nonverbal Arithmetic in Humans: Light from Noise." *Perception & Psychophysics*, 69(7): 1185–1203.

[17] Cybenko, George. (1989) "Approximations by Superpositions of Sigmoidal Functions." *Mathematics of Control, Signals, and Systems*, 2(4): 303–314.

[18] Daw, Nathaniel D., Samuel J. Gershman, Ben Seymour, Peter Dayan, and Raymond J. Dolan. (2011) "Model-Based Influences on Humans' Choices and Striatal Prediction Errors." *Neuron*, 69: 1204–1215.

[19] Daw, Nathaniel D., Yael Niv and Peter Dayan. (2005) "Uncertainty-Based Competition Between Prefrontal and Dorsolateral Striatal Systems for Behavioral Control." *Nature Neuroscience*, 8(12): 1704–1711.

[20] Dayan, Peter and Laurence F. Abbott. (2001) *Theoretical Neuroscience: Computational and Mathematical Modeling of Neural Systems*, The MIT Press.

[21] Dayan, Peter and Nathaniel D. Daw. (2008) "Decision Theory, Reinforcement Learning, and the Brain." *Cognitive, Affective, & Behavioral Neuroscience*, 8(4): 429–453.

[22] Dehaene, Stanislas. (2009) "Origins of Mathematical Intuitions: The Case of Arithmetic." *Annals of the New York Academy of Sciences*, 1156: 232–259.

[23] Dehaene, Stanislas and Jean-Pierre Changeux. (2003) "Development of Elementary Numerical Abilities: A Neuronal Model." *Journal of Cognitive Neuroscience*, 5(4): 390–407.

[24] Doi, Eizaburo, Jeffrey L. Gauthier, Greg D. Field, Jonathon Shlens, Alexander Sher, Martin Greschner, Timothy A. Machado, Lauren H. Jepson, Keith Mathieson, Deborah E. Gunning, Alan M. Litke, Liam Paninski, E. J. Chichilnisky and Eero P. Simoncelli. (2012) "Efficient Coding of Spatial Information in the Primate Retina." *Journal of Neuroscience*, 32(46):16256–16264.

[25] Doya, Kenji. (2007) "Reinforcement Learning: Computational Theory and Biological Mechanisms." *HFSP Journal*, 1(1): 30–40.

[26] Doya, Kenji. (2008) "Modulators of Decision Making." *Nature Neuroscience*, 11(4): 410–416.

[27] Doya, Kenji, Shin Ishii, Alexandre Pouget and Rajesh P.N. Rao (eds.). (2007) *Bayesian Brain: Probabilistic Approaches to Neural Coding*, The MIT Press.

[28] Doya, Kenji and Minoru Kimura. (2008) "The Basal Ganglia and the Encoding of Value." In: Paul W. Glimcher, Colin F. Camerer, Ernst Fehr, Russell A. Poldrack (eds.), *Neuroeconomics: Decision Making and the Brain*, pages 407–416, Academic Press.

[29] Eliasmith, Chris. (2013) *How to Build a Brain: A Neural Architecture for Biological Cognition*, New York: Oxford University Press.

[30] Eliasmith, Chris and Charles H. Anderson. (2003) *Neural Engineering: Computation, Representation, and Dynamics in Neurobiological Systems*, The MIT Press.

[31] Eliasmith,Chris, Terrence C. Stewart, Xuan Choo, Trevor Bekolay, Travis DeWolf, Yichuan Tang, Daniel Rasmussen. (2012) "A Large-Scale Model of the Functioning Brain." *Science*, 338(6111): 1202–1205.

[32] Paul W. Glimcher, Colin F. Camerer, Ernst Fehr, Russell A. Poldrack (eds.). (2008) *Neuroeconomics: Decision Making and the Brain*, Academic Press.

[33] Griffiths, Thomas L., Edward Vul and Adam N. Sanborn. (2012) "Bridging Levels of Analysis for Probabilistic Models of Cognition." *Current Directions in Psychological Science*, 21(4): 263–268.

[34] Hertz, John, Anders Krogh and Richard G. Palmer. (1991) *Introduction to the Theory of Neural Computation*, Reading, MA: Perseus.

[35] Hoyer, Patrik O. and Aapo Hyvärinen. (2003) "Interpreting Neural Response Variability as Monte Carlo Sampling of the Posterior." Manuscript.

[36] Huang, Yanping and Rajesh P. N. Rao. (2011) "Predictive Coding." *WIREs Cognitive Science*, 2: 580–593.

[37] Kahneman, Daniel and Amos Tversky. (1979) "Prospect Theory: An Analysis of Decision under Risk", *Econometrica*, 47(2): 263–291.

[38] Kepecs, Adam and Zachary F. Mainen. (2012) "A Computational Framework for the Study of Confidence in Humans and Animals." *Philosophical Transactions of the Royal Society B*, 367: 1322–1337.

[39] Knill, David C. and Alexandre Pouget. (2004) "The Bayesian Brain: The Role of Uncertainty in Neural Coding and Computation." *Trends in Neurosciences*, 27(12): 712–719.

[40] Kwisthout, Johan and Iris van Rooij. (2013) "Predictive Coding and the Bayesian Brain: Intractability Hurdles That Are Yet To Be Overcome." Manuscript.

[41] Laming, Donald. (2010) "Statistical Information and Uncertainty: A Critique of Applications in Experimental Psychology." *Entropy*, 12: 720–771.

[42] Laughlin, Simon B., John C. Anderson, David O'Carroll and Rob De Ruyter Van Steveninck. (2000) "Coding Efficiency and the Metabolic Cost of Sensory and Neural Information." In: R. Baddeley, P. Hancock and P. Foldiak (eds.), *Information Theory and the Brain*, pages 41–61, New York, NY: Cambridge University Press.

[43] Lee, Daeyeol, Hyojung Seo and Min Whan Jung. (2012) "Neural Basis of Reinforcement Learning and Decision Making." *Annual Review of Neuroscience*, 35: 287–308.

[44] Ma, Wei Ji. (2012) "Organizing Probabilistic Models of Perception." *Trends in Cognitive Sciences*, 16(10): 511–518.

[45] Ma, Wei Ji and Mehrdad Jazayeri. (2014) "Neural Coding of Uncertainty and Probability." *Annual Review of Neuroscience*, 37: 205–220.

[46] McClure, Samuel M., David I. Laibson, George Loewenstein, Jonathan D. Cohen. (2004) "Separate Neural Systems Value Immediate and Delayed Monetary Rewards." *Science*, 306: 503–507.

[47] Miller, Earl and Jonathan Wallis. (2008) "The Prefrontal Cortex and Executive Brain Functions." In: Larry R. Squire et al. (eds.), *Fundamental Neuroscience*, pages 1199–1222, 3rd ed., Academic Press.

[48] Nieder, Andreas. (2005) "Counting on Neurons: The Neurobiology of Numerical Competence." *Nature Reviews Neuroscience*, 6: 177–190.

[49] Nieder, Andreas and Stanislas Dehaene. (2009) "Representation of Number in the Brain." *Annual Review of Neuroscience*, 32: 185–208.

[50] Olshausen, Bruno A. and Field, David J. (1996) "Emergence of Simple-CellReceptive Field Properties by Learning a Sparse Code for Natural Images." *Nature*, 381: 607–609.

[51] Olshausen, Bruno A. and Field, David J. (2005) "How Close AreWe to Understanding V1?" *Neural Computation*, 17: 1665–1699.

[52] O'Reilly, Jill X., Saad Jbabdi, Matthew F. S. Rushworth, Timothy E. J. Behrens. (2013) "Brain Systems for Probabilistic and Dynamic Prediction: Computational Specificity and Integration." *PLOS Biology*, 11(9): 1–14.

[53] Piazza, Manuela, Philippe Pinel, Denis Le Bihan and Stanislas Dehaene. (2007) "A Magnitude Code Common to Numerosities and Number Symbols in Human Intraparietal Cortex." *Neuron*, 53: 293–305.

[54] Platt, Michael L. and Scott A. Huettel. (2008) "Risky Business: The Neuroeconomics of Decision Making Under Uncertainty." *Nature Neuroscience*, 11(4): 398–403.

[55] Poggio, Tomaso. (2012) "The Levels of Understanding framework, revised." *Perception*, 41(9): 1017–1023.

[56] Pouget, Alexandre, Jeffrey M. Beck, Wei Ji Ma and Peter E. Latham. (2013) "Probabilistic Brains: Knowns and Unknowns." *Nature Neuroscience*, 16(9): 1170–1178.

[57] Pouget, Alexandre and Terrence J. Sejnowski. (1997) "Spatial Transformations in the Parietal Cortex Using Basis Functions." *Journal of Cognitive Neuroscience*, 9(2): 222–237.

[58] Rangel, Antonio, Colin Camerer and P. Read Montague. (2008) "A Framework for Studying the Neurobiology of Value-Based Decision Making." *Nature Reviews Neuroscience*, 9: 545–556.

[59] Rao, Rajesh P. N. (2010) "Decision Making Under Uncertainty: a Neural Model Based on Partially Observable Markov Decision Processes." *Frontiers in Computational Neuroscience*, 4(146): 1–18.

[60] Reynolds, John H., Jacqueline P. Gottlieb, and Sabine Kastner. (2008) "Attention." In: Larry R. Squire et al. (eds.), *Fundamental Neuroscience*, pages 1113–1132, 3rd ed., Academic Press.

[61] Rushworth, Matthew F. S. and Timothy E. J. Behrens. (2008) "Choice, Uncertainty and Value in Prefrontal and Cingulate Cortex." *Nature Neuroscience*, 11(4): 389–397.

[62] Schultz, Wolfram, Peter Dayan and P. Read Montague. (1997) "A Neural Substrate of Prediction and Reward." *Science*, 275: 1593–1599.

[63] Schultz, Wolfram, Kerstin Preuschoff, Colin Camerer, Ming Hsu, Christopher D. Fiorillo, Philippe N. Tobler and Peter Bossaerts. (2008) "Explicit Neural Signals Reflecting Reward Uncertainty." *Philosophical Transactions of the Royal Society B*, 363: 3801–3811.

[64] Simoncelli, Eero P. and Bruno A. Olshausen. (2001) "Natural Image Statistics and Neural Representation." *Annual Review of Neuroscience*, 24: 1193–1216.

[65] Smith, Edward E. and John Jonides. (2003) "Executive Control and Thought." In: Larry R. Squire, James L. Roberts, Nicholas C. Spitzer, Michael J. Zigmond, Susan K. McConnell and Floyd E. Bloom (eds.), *Fundamental Neuroscience*, pages 1377–1394, 2nd ed., Academic Press.

[66] Spratling, Michael W. (2012) "Unsupervised Learning of Generative and Discriminative Weights Encoding Elementary Image Components in a Predictive Coding Model of Cortical Function." *Neural Computation*, 24: 60–103.

[67] Squire, Larry R., Floyd Bloom, Nicholas C. Spitzer, Sascha du Lac, Anirvan Ghosh, Darwin Berg (eds.). (2008) *Fundamental Neuroscience*, 3rd ed., Academic Press.

[68] Summerfield, Christopher and Tobias Egner. (2009) "Expectation (and Attention) in Visual Cognition." *Trends in Cognitive Sciences*, 13(9): 403–409.

[69] Sutton, Richard S. and Andrew G. Barto. (1998) *Reinforcement Learning: An Introduction*, Cambridge, MA: MIT Press.

[70] Tanaka, Saori C., Kenji Doya, Go Okada, Kazutaka Ueda,Yasumasa Okamoto and Shigeto Yamawaki. (2004) "Prediction of Immediate and Future Rewards Differentially Recruits Cortico-Basal Ganglia Loops." *Nature Neuroscience*, 7(8): 887–893.

[71] Tudusciuc, Oana and Andreas Nieder. (2007) "Neuronal Population Coding of Continuous and Discrete Quantity in the Primate Posterior Parietal Cortex." *Proceedings of the National Academy of Sciences*, 104(36): 14513–14518.

[72] Tudusciuc, Oana and Andreas Nieder. (2009) "Contributions of Primate Prefrontal and Posterior Parietal Cortices to Length and Numerosity Representation." *Journal of Neurophysiology*, 101: 2984–2994.

[73] Tversky, Amos and Daniel Kahneman. (1992) "Advances in Prospect Theory: Cumulative Representation of Uncertainty", *Journal of Risk and Uncertainty*, 5: 297–323.

[74] Vilares, Iris, James D. Howard, Hugo L. Fernandes, Jay A. Gottfried and Konrad P. Kording. (2012) "Differential Representations of Prior and Likelihood Uncertainty in the Human Brain." *Current Biology*, 22(18): 1641–1648.

[75] Yang, Tianming and Michael N. Shadlen. (2007) "Probabilistic Reasoning by Neurons." *Nature*, 447: 1075–1082.

[76] Zhaoping, Li. (2006) "Theoretical Understanding of the Early Visual Processes by Data Compression and Data Selection." *Network: Computation in Neural Systems*, 17: 301–334.

I Generalized optimization problem

More explicitly, in place of optimization problem

$$\max_{\boldsymbol{\theta}} \mathrm{E}_t^g \left[\varphi^\sharp(\boldsymbol{x}|\boldsymbol{\theta})\right] = \int_{\mathrm{supp}(g)} \varphi^\sharp(\boldsymbol{x}|\boldsymbol{\theta})\, g(\boldsymbol{x})\, d\boldsymbol{x}, \qquad \{\mathcal{P}_\theta\}$$

where

$$g(\boldsymbol{x}) \text{ is given,}$$

we consider generalized optimization problem

$$\max_{\boldsymbol{\theta}} \mathrm{E}^h \left[\varphi(\hat{\boldsymbol{x}}|\boldsymbol{\theta})\right] = \int_{\mathrm{supp}(h)} \varphi(\hat{\boldsymbol{x}}|\boldsymbol{\theta})\, h(\hat{\boldsymbol{x}})\, d\hat{\boldsymbol{x}}, \qquad \{\mathcal{P}_{\theta\mathcal{I}}\}$$

where

$$\begin{aligned} h(\hat{\boldsymbol{x}}) &:= \int_{\mathrm{supp}(g)} f(\boldsymbol{x}, \hat{\boldsymbol{x}})\, d\boldsymbol{x}, \\ f(\boldsymbol{x}, \hat{\boldsymbol{x}}) &:= \arg \left\{ \min_{f(\boldsymbol{x},\hat{\boldsymbol{x}})} \mathrm{E}^f \left[d(\varphi^\sharp(\boldsymbol{x}|\boldsymbol{\theta}), \varphi(\hat{\boldsymbol{x}}|\boldsymbol{\theta})) \right] \text{ subject to } \mathcal{I}(g(\boldsymbol{x}); h(\hat{\boldsymbol{x}})) \leq \kappa \right\}, \\ & g(\boldsymbol{x}) \text{ is given.} \end{aligned}$$

For a given $\boldsymbol{\theta}$, objective function $\varphi^\sharp(\cdot)$ by construction weakly envelops $\varphi(\cdot)$ from above. Distortion function may be chosen with the purpose of producing objective function approximation in the sense of L^p norm (to power p):

$$d(\varphi^\sharp(\boldsymbol{x}|\boldsymbol{\theta}), \varphi(\hat{\boldsymbol{x}}|\boldsymbol{\theta})) = ||\varphi^\sharp(\boldsymbol{x}|\boldsymbol{\theta}) - \varphi(\hat{\boldsymbol{x}}|\boldsymbol{\theta})||_p^p = |\varphi^\sharp(\boldsymbol{x}|\boldsymbol{\theta}) - \varphi(\hat{\boldsymbol{x}}|\boldsymbol{\theta})|^p =: d(\boldsymbol{x}, \hat{\boldsymbol{x}}). \tag{I.49}$$

It is zero at partitions of $\mathrm{supp}(g(\boldsymbol{x})) \times \mathrm{supp}(h(\hat{\boldsymbol{x}}))$ such that $\varphi^\sharp(\boldsymbol{x}|\boldsymbol{\theta}) = \varphi(\hat{\boldsymbol{x}}|\boldsymbol{\theta})$, and positive otherwise. Pointwise, non-integrated approximation error for $p = 2$ equals

$$\begin{aligned} d(\boldsymbol{x}, \hat{\boldsymbol{x}}) &= |\varphi^\sharp(\boldsymbol{x}|\boldsymbol{\theta}) - \varphi(\hat{\boldsymbol{x}}|\boldsymbol{\theta})|^2 = \\ &= \left| \varphi^\sharp(\boldsymbol{x}|\boldsymbol{\theta}) + \frac{\partial \varphi^\sharp(\boldsymbol{x}|\boldsymbol{\theta})}{\partial \boldsymbol{x}^\intercal} (\hat{\boldsymbol{x}} - \boldsymbol{x}) + o(|\boldsymbol{x} - \hat{\boldsymbol{x}}|) - \varphi^\sharp(\boldsymbol{x}|\boldsymbol{\theta}) \right|^2 = \\ &= \left(\frac{\partial \varphi^\sharp(\boldsymbol{x}|\boldsymbol{\theta})}{\partial \boldsymbol{x}^\intercal} (\hat{\boldsymbol{x}} - \boldsymbol{x}) + o(|\boldsymbol{x} - \hat{\boldsymbol{x}}|) \right)^2. \end{aligned} \tag{I.50}$$

Thus, in regions with "large" $\partial \varphi^\sharp(\boldsymbol{x}|\boldsymbol{\theta}) / \partial \boldsymbol{x}^\intercal$, values of $\hat{\boldsymbol{x}}$ "far" from $\boldsymbol{x}$ will be penalized and ensured as "low" probability mass $f(\boldsymbol{x}, \hat{\boldsymbol{x}})$ as possible.

The generalized optimization problem can be expressed more condensely by defining a non-linear (information-processing capacity) "constrained expectations" operator

$$\begin{aligned} \mathrm{E}^{\kappa,p} \left[\varphi^\sharp(\boldsymbol{x}|\boldsymbol{\theta})\right] := \mathrm{E}^h \left[\varphi(\hat{\boldsymbol{x}}|\boldsymbol{\theta})\right] \text{ s.t.} \qquad & h(\hat{\boldsymbol{x}}) := \int_{\mathrm{supp}(g)} f(\boldsymbol{x}, \hat{\boldsymbol{x}})\, d\boldsymbol{x}, \\ & f(\boldsymbol{x}, \hat{\boldsymbol{x}}) := \arg \left\{ \min_{f(\boldsymbol{x},\hat{\boldsymbol{x}})} \mathrm{E}^f \left[d(\varphi^\sharp(\boldsymbol{x}|\boldsymbol{\theta}), \varphi(\hat{\boldsymbol{x}}|\boldsymbol{\theta})) \right] \right. \\ & \qquad \left. \text{s.t. } \mathcal{I}(g(\boldsymbol{x}); h(\hat{\boldsymbol{x}})) \leq \kappa \right\}, \\ & g(\boldsymbol{x}) \text{ is given.} \end{aligned} \tag{I.51}$$

Then, in place of solving

$$\max_{\boldsymbol{\theta}} \mathrm{E}^{g} \left[\varphi^{\sharp}(\boldsymbol{x}|\boldsymbol{\theta}) \right],$$

we say that we are solving

$$\max_{\boldsymbol{\theta}} \mathrm{E}^{\kappa,p} \left[\varphi^{\sharp}(\boldsymbol{x}|\boldsymbol{\theta}) \right].$$

Original optimization problem is a special case of the generalized formulation. For any $g(\boldsymbol{x})$ such that $\mathcal{E}(g(\boldsymbol{x})) < \infty$, there exists $\kappa^{\sharp} < \infty$ such that for any $\kappa \geq \kappa^{\sharp}$ and any chosen norm L^p (to power p), we have $\mathcal{P}_{\theta\mathcal{I}} \iff \mathcal{P}_{\theta}$, i.e.

$$\max_{\boldsymbol{\theta}} \mathrm{E}^{\kappa,p} \left[\varphi^{\sharp}(\boldsymbol{x}|\boldsymbol{\theta}) \right] \iff \max_{\boldsymbol{\theta}} \mathrm{E}^{g} \left[\varphi^{\sharp}(\boldsymbol{x}|\boldsymbol{\theta}) \right]. \tag{I.52}$$

J Choice of wealth shares invested: Resolving the dilemma of circularity

J.1 Alternative approaches to resolution

In addition to rather agnostic approach based on the robustness argument, which is actually taken in the paper, there are alternative ways of dealing with the dilemma.

One obvious method to resolve it would be to appeal to a fixed-point argument and use optimal allocations, in equilibrium leading to the use of net supplies of each tree as corresponding shares. But this would require (i) iterative updating of shares and of the distortion function, imposing unrealistic information processing demands (or continuous updating with some analogue of abstract formal symbolic-like calculations); or else (ii) making "schizophrenic" assumptions about the agent's information set: that he knows (nominal) net supplies before making allocation choices that will affect market prices and thus net supplies, but does not use that knowledge about net supplies to back out optimal allocations directly, without the need for any optimization (dismissing this last criticism by assuming our representative agent is formed by a continuum of identical agents is possible only if the latter are not aware of their identicality). Another alternative would be to use agent's previous-period holdings as shares; but in a stationary problem like ours this suffers from the same criticism of "schizophrenia" as the previous approach.

In the interest of completeness, §J.2 shows that the iterative/continuous updating approach yields results qualitatively similar to (in some sense actually a degenerate version of) those stemming from a more agnostic approach taken in the main body of the paper.

J.2 Iterative/continuous updating approach

Here we sketch the core features of the solution to feasible consumption and portfolio choice problem $\mathcal{P}_{\mathcal{QI}}$ with the correspoding informational sub-problem based on the distortion function from Proposition 3.1 in its unaltered form.

Right away, Propositions 3.2, 3.3 and 4 become redundant. As a result, so does Proposition C.1.

Next, Proposition 5 is modified in the following way. For random vector $\boldsymbol{r}_{t+1} \sim \mathcal{N}(\boldsymbol{\mu}_r, \boldsymbol{\Sigma}_r)$, specific solution to the informational problem takes the form (note the absence of the boundary solution case)

$$f(\boldsymbol{\omega}_t^\intercal \boldsymbol{r}_{t+1} | \boldsymbol{\omega}_t^\intercal \hat{\boldsymbol{r}}_{t+1}) = (2\pi)^{-\frac{1}{2}} \left|\frac{\lambda}{2}\right|^{-\frac{1}{2}} \exp\left(-\frac{\left(\boldsymbol{\omega}_t^\intercal(\boldsymbol{r}_{t+1} - \hat{\boldsymbol{r}}_{t+1} + \check{\boldsymbol{\mu}}_r)\right)^2}{2(\lambda/2)}\right) \quad \forall \hat{\boldsymbol{r}}_{t+1} \in \mathbb{R}^K,$$

resulting in decomposition

$$\boldsymbol{\omega}_t^\intercal \boldsymbol{r}_{t+1} = \boldsymbol{\omega}_t^\intercal \hat{\boldsymbol{r}}_{t+1} - \boldsymbol{\omega}_t^\intercal \check{\boldsymbol{\mu}}_r + \epsilon_{r,t+1},$$

also producing

$$\boldsymbol{\omega}_t^\intercal \boldsymbol{\Sigma}_r \boldsymbol{\omega}_t = \boldsymbol{\omega}_t^\intercal \hat{\boldsymbol{\Sigma}}_r \boldsymbol{\omega}_t + \Psi_r,$$

where

$$\epsilon_{r,t+1} \sim \mathcal{N}(0, \Psi_r), \qquad \Psi_r = \frac{\lambda}{2},$$
$$\boldsymbol{\omega}_t^\intercal \hat{\boldsymbol{r}}_{t+1} \sim \mathcal{N}(\boldsymbol{\omega}_t^\intercal \hat{\boldsymbol{\mu}}_r, \boldsymbol{\omega}_t^\intercal \hat{\boldsymbol{\Sigma}}_r \boldsymbol{\omega}_t), \qquad \boldsymbol{\omega}_t^\intercal \hat{\boldsymbol{\Sigma}}_r \boldsymbol{\omega}_t = \boldsymbol{\omega}_t^\intercal \boldsymbol{\Sigma}_r \boldsymbol{\omega}_t - \Psi_r;$$

$$\lambda = 2e^{-2\kappa} |\boldsymbol{\omega}_t^\intercal \boldsymbol{\Sigma}_r \boldsymbol{\omega}_t|.$$

Or more conveniently (signifying Moore–Penrose pseudo-inverse and pseudo-determinant in the sense of the product of all non-zero eigenvalues with + superscript and subscript, respectively),

$$f(\boldsymbol{r}_{t+1}|\hat{\boldsymbol{r}}_{t+1}) = (2\pi)^{-\frac{K}{2}} |\boldsymbol{\Psi}_r|_+^{\frac{1}{2}} \exp\left(-\frac{1}{2}(\boldsymbol{r}_{t+1} - \hat{\boldsymbol{r}}_{t+1} + \check{\boldsymbol{\mu}}_r)^\intercal \boldsymbol{\Psi}_r^+ (\boldsymbol{r}_{t+1} - \hat{\boldsymbol{r}}_{t+1} + \check{\boldsymbol{\mu}}_r)\right) \quad \forall \hat{\boldsymbol{r}}_{t+1} \in \mathbb{R}^K,$$

resulting in decomposition

$$\boldsymbol{r}_{t+1} = \hat{\boldsymbol{r}}_{t+1} - \check{\boldsymbol{\mu}}_r + \boldsymbol{\epsilon}_{r,t+1},$$

also producing

$$\boldsymbol{\Sigma}_r = \hat{\boldsymbol{\Sigma}}_r + \boldsymbol{\Psi}_r,$$

where

$$\boldsymbol{\epsilon}_{r,t+1} \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{\Psi}_r), \qquad \boldsymbol{\Psi}_r = \frac{\lambda}{2}(\boldsymbol{\omega}_t^\intercal \boldsymbol{\omega}_t)^{-2} \boldsymbol{\omega}_t \boldsymbol{\omega}_t^\intercal,$$
$$\hat{\boldsymbol{r}}_{t+1} \sim \mathcal{N}(\hat{\boldsymbol{\mu}}_r, \hat{\boldsymbol{\Sigma}}_r), \qquad \hat{\boldsymbol{\Sigma}}_r = \boldsymbol{\Sigma}_r - \boldsymbol{\Psi}_r;$$

$$\lambda = 2e^{-2\kappa} \boldsymbol{\omega}_t^\intercal \boldsymbol{\omega}_t |\boldsymbol{\Sigma}_r|.$$

(Note that

$$|\boldsymbol{\omega}_t^\intercal \boldsymbol{\Sigma}_r \boldsymbol{\omega}_t| \lesseqgtr \boldsymbol{\omega}_t^\intercal \boldsymbol{\omega}_t |\boldsymbol{\Sigma}_r|$$

in general, as can be easily shown using Rayleigh–Ritz theorem, and hence λ multipliers above are different for the same values of κ. In short, this stems from the fact that differential entropy is not invariant to transformations such as rescaling of random variables involved.) This latter, more convenient formulation implies a rank-one matrix $\boldsymbol{\Psi}_r$, so that elements of vector $\boldsymbol{\epsilon}_{r,t+1}$ are perfectly correlated (random vector $\boldsymbol{\epsilon}_{r,t+1}$ is then said to have singular Normal distribution). It is also noteworthy that, counter to intuition, elements of approximation error variance-covariance matrix $\boldsymbol{\Psi}_r$ that correspond to bigger components of $\boldsymbol{\omega}_t$ are relatively larger and, hence, respective approximation precisions are smaller.

Finally, Proposition 2 in its part applicable to interior soluton case holds without change, i.e. subjective correlations between elements of $\hat{\boldsymbol{r}}_{t+1}$ are inflated versions of their objective counterparts (as can be seen from the direction of change of generic correlation coefficient $\hat{\rho}_{r,kl}$ for $k, l \in \{1, \ldots, K\}$ when λ increases).

Solution to consumption and investment sub-problem is adjusted inasmuch as to account for changes in $\hat{\boldsymbol{\Sigma}}_r$.

To summarize, the main features of the solution under iterative/continuous updating approach are similar to those based on the robustness argument. Solutions to informational sub-problem have the same broad form, which in turn determines the extent of differences to the solutions to consumption and investment sub-problems, and key results such as categorization hold under both approaches.

K Machine-aided information processing

Our treatment is general enough to also account for information processing performed with the aid of machines.

When presenting in part §2.2 (together with §F) the process of decision-making under risk that underlies our framework, we illustrated the reduction of probability distribution's entropy by amalgamation of several lower-probability events into one. (Moreover, extreme tail events—e.g., those missing from the observed sample—might have been completely omitted with their probabilities equated to zero.) The algorithm discussed there also focused on mental computations as the practical implementation of information processing we are concerned about.

However, we can push this logic further and argue that it also holds for machine-aided computations. For example, this is trivially true when the costs—in terms of information processing capacity demands—of formalization and coding machine instructions are proportional to entropy of the distribution concerned (as was the case with mental computations). (Such a proportional relationship can be established by way of renowned Solomonoff-Kolmogorov-Chaitin complexity notion (see Rissanen, 2007), e.g. employing the arguments of Leung-Yan-Cheong and Cover (1978).) Therefore, the same algorithm of §F applies to real-world situations in which machine-computing technologies are used heavily. The only technical difference pertains to implementation and interpretation: while in the case of mental computations the most natural information processing "bottleneck" is working memory, or perhaps the computation of optimal decision, in the case of machine computations it is the summarization of given information, i.e. expressing everything in unambiguous formal language (which, as a matter of fact, admits "back-of-the-envelope" approximations, but precludes "hand-waving"). The operational outcome of extending our logic to account also for machine computations again boils down to reduction in the entropy of the probability distributions used, because the high-entropy true distribution has to be simplified to become amenable for being coded up.

In this sense of machine-aided information processing, a computationally cheap way of summarizing a complex true probability distribution that saves on formalization and coding costs is using observed sample data, which thus serve as a simplified distribution. (Supposing subsequent mental computations above and beyond that, "low" information processing capacity κ would require further simplification of the sample distribution, as well as further compensating mean-adjustment.)

If (representative) investor is the leading example of an agent using mental computations, an econometrician processing sample data is the most intuitive way of thinking about an agent that uses machine computations. The corresponding effective information processing capacity κ is then easiest understood as effective capacity of an econometrician, rather than as effective capacity of investor relying exclusively on sample data. Indeed, econometricians usually utilize sample distribution as is without imposing any (pessimistic) adjustments to the mean, in contrast to (optimally behaving) investors.

This fits perfectly into our formal framework (although not contemplating it ex ante, econometrician's behavior nevertheless falls under its logic ex post): according to Proposition 3.1, optimal magnitude of mean adjusment is measured from the origin point of zero wealth share invested in risky assets $\boldsymbol{\omega}_t$, with said point implying matching expected values of returns and at the same time presenting a sensible way to think about an impartial non-market-participating econometrician.

References for Machine-aided processing

[1] Leung-Yan-Cheong, Sik K. and Thomas M. Cover. (1978) "Some Equivalences Between Shannon Entropy and Kolmogorov Complexity", *IEEE Transactions on Information Theory*, IT-24 (3): 331–338.

[2] Rissanen, Jorma. (2007) *Information and Complexity in Statistical Modeling*, Springer.