THE REAL DISTRIBUTION OF MINIMAL PUZZLES

Author	Message

denis_berthier

Joined: 19 Jun 2007
Posts: 697
Location: Paris, France

Posted: Sat Jul 11, 2009 1:48 pm Post subject: quote edit

> **Red Ed wrote:**
>
> > **denis_berthier wrote:**
> >
> > The correction factors define the relative density of the two measures on B - nothing more, nothing less.
>
> No, they define the "relative density" of n and n+1 clue objects in the set of all indexed subgrids. It's an unjustified leap of faith to assume that the same is true in the subset B. Are there only 65/17 times more (indexed) 17-clue puzzles than (indexed) 16-clue puzzles? No, of course not.

I think there's one thing you haven't understood.
At each floor, there is a uniform measure of probability on indexed puzzles. On B, define the measure of a puzzle to be the unique probability it gets from floor n.
These numbers on B form a probability measure (obviously - because B cuts all the branches at some point and that any probability at floor n is split into smaller equal probabilities at floor n-1).
Moreover, the probabilities thus defined on B are exactly the probabilities of pussles produced by a top-down generator (modulo the multiplying factor of the number of identical underlying puzzles).

This probability on B is not uniform.

What we want is a uniform probability on B. For this, we just have to consider the cf correction factors.

Back to top profile pm www

denis_berthier

Posted: Sat Jul 11, 2009 1:50 pm Post subject: quote edit

eleven wrote:

Joined: 19 Jun 2007
Posts: 697
Location: Paris, France

> What about bottom up generation without dropping clues ?
>
> If you have a table, which shows, how often a puzzle with n clues was tried, how many of those puzzles were minimal, unique, with multi solutions and with no solution, can you calculate then the bias from it ?

I think one could do an easy computation similar to mine for this type of generator. But the grids it produces don't have any intrinsic meaning.

Back to top profile pm www

eleven

Posted: Sat Jul 11, 2009 2:02 pm Post subject: quote

Joined: 10 Feb 2008
Posts: 339

Red Ed wrote:

> I don't trust bottom-up generators: they don't generate clue structures that mimic those in ordinary grids (as evidenced by the 0.5-clue difference in minimal puzzle size as compared to top-down generators).

One more remark, Red Ed. I am sure that the bias must be different for bottom up than top down generation, because bottom up has to deal with no solution puzzles. So again my question. Would it be possible to get a bias out of the data ? Isn't the reason for the bias that puzzles with n clues are not visited that often as they should have been according to the real dstribution ?

Back to top profile pm

denis_berthier

Posted: Sat Jul 11, 2009 2:15 pm Post subject: quote edit

Joined: 19 Jun 2007
Posts: 697
Location: Paris, France

Red Ed wrote:

> **denis_berthier wrote:**
>
> > The correction factors define the relative density of the two measures on B - nothing more, nothing less.
>
> No, they define the "relative density" of n and n+1 clue objects in the set of all indexed subgrids. It's an unjustified leap of faith to assume that the same is true in the subset B. Are there only 65/17 times more (indexed) 17-clue puzzles than (indexed) 16-clue puzzles? No, of course not.

I had missed the last sentence. It is a complete misinterpretation of the correction factors.

The correction factors don't give the real ratio P(n+1)/P(n) on B (which is the only thing we're interested in).
If you have some experimental P(n+1) and P(n) on B, obtained from a top-down generator, multiplying these P(n+1) and P(n) by the correction factors gives you (an estimation of) the corrected, unbiased ratio of probabilities on B.

Considering 17-clue and 16-clue puzzles is a very bad example, as we don't have any experimental data on P(17) or P(16), even when taking samples of 1,000,000 puzzles.

Just to answer in advance any new objection: in case the real P(16) would be 0, all the experimental P(0) you'll get will also be 0, P(17)/P(16) will not be defined. Then, you just have to limit the computations at P(17). In practice, evenwith 1,000,000 puzzle samples, we have to limit it downwards at 20.

denis_berthier

Joined: 19 Jun 2007
Posts: 697
Location: Paris, France

Posted: Sat Jul 11, 2009 3:54 pm Post subject:

THE REAL DISTRIBUTION OF MINIMAL PUZZLES

As the first version has led to some misinterpretations, here is a new version with a few more details and a lot of redundancies.
I've also added the unbiased distribution of clues of minimal puzzles, that I had forgotten in the first version.

1) Introduction
Yesterday evening, I read that camera lens makers have always been confronted to a series of problems: chromatic aberration, purple fringing, barrel or pincushion distortion, vignetting, ... The classical approach, and the only one feasible with classical cameras, was to improve the lenses - which led to large, complex and expensive lenses. But, with the advent of digital camears, slight problems of the above types (and others) can be accepted in the lens, provided that we can measure them precisely and correct them by the software. This is heresy for traditional lens makers such as Leica, but some camera makers have adopted this solution.

This morning, I woke up with the following idea: we are unable to build unbiased collections of minimal puzzles. What if we take the collections as they are produced by the current generators and apply a correction to the results? Indeed, I was very surprised that the analysis of bias in top-down generators is straightforward and I could program the corrections needed in less than one hour.

2) A forest of puzzles

Consider first the following 80-floor structure (a forest of trees with branches pointing downwards), the nodes being indexed puzzles (which may have one or more solutions).
floor 81 : all the N different complete solution grids (N is huge, but no one is asking you to cross the whole forest), each indexed by the empty sequence; notice that all the puzzles at this floor have 81 clues;
floor 80: each indexed puzzle P at floor 81 sprouts 81 branches pointing to floor 80, one for each clue C in P; the head of this C branch will be the indexed puzzle obtained from P by removing clue C and indexed by the 1-element sequence (C); notice that all the puzzles at floor 80 have 80 clues;

Now, recursively, given floor n (all indexed puzzles of which have n clues and are indexed by sequences of length 81-n), build floor n-1 as follows:
each indexed puzzle P at floor n sprouts n branches pointing to floor n-1, one for each clue C in P; the head of the C branch will be the indexed puzzle obtained from P by removing clue C and indexed by the (81-n+1)-element sequence obtained by appending C to the end of the sequence indexing P; notice that all the indexed puzzles at floor n-1 have n-1 clues and that the index of a puzzle is the (ordered) sequence of deletions that led to it.

It is easy to see that, at floor n, there are N * 81! / n! indexed puzzles, each of which has its underlying (non indexed) puzzle identical to that of (81 - n)! indexed puzzles at the same floor (including itself).

Along each branch of this forest, there is one and only one (indexed) minimal puzzle. Above it, all the indexed puzzles have redundant clues; below it, all the indexed puzzles have multiple solutions.
Consider now the border B (i.e. the set of indexed minimal puzzles). This border crosses all the branches of our forest at different floors.

If we consider that each indexed puzzle at each floor has the same probability of being reached from the top in (81 - n) steps and these probabilities sum up to 1 at each floor, then any indexed puzzle at floor n has probability 1/ {(N * 81! / n!) of being reached.
And any non-indexed puzzle at floor n has probability Pn = 1/ {(N * 81! / n!) / (81 - n)!} = 1/N * 1/81! * n! * (81 - n)! of being reached.
For small n, this is a inimaginably small, umanageable number.

But the ratio Pn+1 / Pn is very simple: **Pn+1 / Pn = (n + 1) / (81 - n).**

Now consider the following probability on B: for any indexed puzzle Q in B, there is a unique floor n containing Q; assign to Q its uniquely defined probability as an element of floor n. This defines a number for all the elements of B. But as B cuts all the branches of the forest and any indexed puzzle at a level n has its probability split into equal parts for its immediate descendants, it is easy to see that these numbers form a probability measure on B.
A probability measure on the underlying puzzles of B (i.e. on the set B~ of equivalence classes of B, which is also the set of non-indexed minimal puzzles) is obtained by giving to an equivalence class the sum of the probabilitiees of its members.
And, from the existence of the ratios Pn+1 / Pn, it is easy to see that the probability induced on B~ by these probabilities is non uniform.
But, as we know precisely this bias wrt uniformity, we can correct it easily by applying the correction factors cf(n) to the probabilities defined on B~. Only the relative values of the cf(n) is important: they satisfy **cf(n+1) / cf(n) = (81 - n) / (n+1)**.

3) Top down generators and the set of paths among the single solution puzzles

Consider now a top-down generator. The first phase is the generation of unbiased complete grids, which we know how to do.
For each initial complete grid, the second phase of the generator consists of following some path donwards in our forest until it reaches an indexed minimal puzzle on the border B, where it stops.
The generator works according to the probabilities on our forest of indexed puzzles: its output, non indexed puzzles are produced according to the probabilities on B~, instead of having a uniform distribution on minimal puzzles.
But, as we know its bias precisely, it is easy to apply the correction factors.

What does this mean in practice, when we want to compute statistics of minimal puzzles?
If you have a variable X (defined on minimal puzzles), let
on(n) be the observed number of puzzles with n clues in some random sample generated top-down,
X(n) be the observed mean value of X for puzzles with n clues in the same sample.
The raw (biased) mean of X (obtained from the random sample) is computed classically as sum(X(n) * on(n)) / sum(on(n))
The corrected mean of X (unbiased in the set of minimal puzzles) must be computed as: **unbiased-mean(X) = sum(X(n) * on(n) * cf(n)) / sum(on(n) * cf(n))**.

This formula shows that the cf(n) sequence needs be defined only modulo a multiplicative factor.
For convenience, I chose cf(20) =1 in my computations. This gives the following sequence of correction factors:

Code:

```
cf-sequence[19...31] =   0.333333333333333   1
 2.9047619047619   7.92207792207792
 20.3218520609825   49.1111424807077
 111.973404856014   241.173487382183
 491.279326148891   947.467271858576
 1731.57811753464   3001.40207039337
 4937.79050290523
```

It may be shocking to consider that some puzzles must be given a weight 3000 times greater than other puzzles (ratio between 30-clue and 20-clue puzzles), but that's how it is.
A consequence of this is that statistics on minimal puzzles must rely on very large samples.

4) Applications

Let's use the raw computations for 2 collections (sudogen0_1M and rabrnd1m) of 1 million puzzles each, generated by 2 top-down generators with very different first phases.
For details on these collections and computations, see the "rating rules / puzzles" thread or my web page:

http://www.carva.org/denis.berthier/HLS/Classification/index.html.
We get the following results.

4.1) The mean number of clues of minimal puzzles = 25.39

sudogen0_1M: raw-average = 24.380591 unbiased-average = 25.3910685435253
rabrnd1M: raw-average = 24.384134 unbiased-average = 25.3920009372608
The 2 collections lead to the same result.

This is 1 more than the raw-average.

4.2) The mean SER of minimal puzzles = 4.06

sudogen0_1M: raw-average = 3.7722223 unbiased-average = 4.06321801478134
rabrnd1M: raw-average = 3.76660230000001 unbiased-average = 4.06047314710775

4.3) The mean NRCZT-rating of minimal puzzles = 2.09

sudogen0_1M: raw-average = 1.94113060000003 unbiased-average = 2.09015177491554
rabrnd1M: data not available

4.4) More generally, one can compute the real distribution of clues of minimal puzzles

It is merely the product of the observed distribution and the correction factors, namely on(n) * cf(n) (normalised, of course, by sum(on(n) * cf(n)).

Code:

```
#clues  raw-occurrences   unbiased-occurrences(normalised to total 1,000,000)
19      0      0.0
20      44      0.445783397960804
21      2428      71.4546401571198
22      34548      2772.89378629794
23      172512      35518.4339857941
24      342335      170334.474247819
25      297838      337882.893778113
26      122116      298382.499310421
27      25315      126002.056742764
28      2686      25783.4743915203
29      168      2947.28805790044
30      10      304.085275815125
31      0      0.0
```

The large majority of puzzles is still in the range [23 - 27] clues, but the real distribution is notably different from the raw distribution obtained from a top-down generator.

5) Remarks
In the " how many minimal sudokus has an average grid" thread, another approach has been taken by Red Ed, who tries to estimate the number of minimal puzzles with n clues, a very hard problem. His estimation of the average number of clues is 26.4 (but he mentions that this is must be taken "with a pinch of salt"), much above the 25.39 value computed here.
In the present approach, no estimation of these numers is needed. Only very simple computations lead to an estimation of the real values from a knowledge of the observed values obtained from a large sample produced by a top-down generator.

6) A remark on bottom-up generators
A similar analysis for bottom-up generators is more difficult (but I din't have time to really think of it), because these generators are not purely bottom-up. Staring with 0 clues, they add clues until they reach a single solution puzzle, but after that they delete clues until they reach a minimal puzzle.

Back to top profile pm www

Red Ed
Joined: 06 Jun 2005
Posts: 570

Posted: Sat Jul 11, 2009 4:33 pm Post subject: quote

Posting twice doesn't make it right, Denis

I don't know why I bother, but let's keeping plugging away at this. Start by fixing a solution grid and a pair of minimal puzzles at levels n+1 and n. Let p[n+1] and p[n] be the respective probabilities that the top-down search process stops at those minimal puzzles.

First, let's just do a sanity-check: you're not claiming that **Pn** (your notation) equals p[n] (my notation) are you? That would be very, very, bad.

Now then, *if* it were the case that p[n+1]/p[n] = **Pn+1/Pn** *then* your formulae would hold true. Are you claiming that relation is true (even if the one above is not)? If so, how do you think you have proved that? Your text seems a bit woolly about that point.

Back to top profile pm

denis_berthier
Joined: 19 Jun 2007
Posts: 697
Location: Paris, France

Posted: Sat Jul 11, 2009 4:54 pm Post subject: quote edit

> **Red Ed wrote:**
> Posting twice doesn't make it right

And claiming it is wrong without saying what's wrong with the proof doesn't

make it wrong.
I improved the first version with lots of redundancies in order to eliminate any ambiguity.

> **Red Ed wrote:**
> Start by fixing a solution grid and a pair of minimal puzzles at levels n+1 and n. Let p[n+1] and p[n] be the respective probabilities that the top-down search process stops at those minimal puzzles.
> First, let's just do a sanity-check: you're not claiming that **Pn** (your notation) equals p[n] (my notation) are you? That would be very, very, bad.

It's neither good nor bad. It's just plainly false: the Pn are for all the initial grids, not for a single one. They are much smaller than your p[n].

> **Red Ed wrote:**
> Now then, *if* it were the case that p[n+1]/p[n] = **Pn+1/Pn** *then* your formulae would hold true. Are you claiming that relation is true (even if the one above is not)?

Are you speaking of indexed or non-indexed puzzles? The Pn+1/Pn are for non-indexed puzzles.

> **Red Ed wrote:**
> If so, how do you think you have proved that? Your text seems a bit woolly about that point.

Read the new version. How do you think you can disprove any part of it? :)

Back to top profile pm www

Red Ed

Joined: 06 Jun 2005
Posts: 570

Posted: Sat Jul 11, 2009 5:10 pm Post subject: quote

> **denis_berthier wrote:**
>> **Red Ed wrote:**
>> Now then, *if* it were the case that p[n+1]/p[n] = **Pn+1/Pn** *then* your formulae would hold true. Are you claiming that relation is true (even if the one above is not)?
>
> Are you speaking of indexed or non-indexed puzzles? The Pn+1/Pn are for non-indexed puzzles.

I defined p[n] for non-indexed minimal puzzles.

Now please answer the question about the ratio p[n+1]/p[n].

Back to top profile pm

denis_berthier

Posted: Sat Jul 11, 2009 5:21 pm Post subject: quote edit

Joined: 19 Jun 2007
Posts: 697
Location: Paris, France

> **Red Ed wrote:**
>
> > **denis_berthier wrote:**
> >
> > > **Red Ed wrote:**
> > >
> > > Start by fixing a solution grid and a pair of minimal puzzles at levels n+1 and n. Let p[n+1] and p[n] be the respective probabilities that the top-down search process stops at those minimal puzzles.
> > > First, let's just do a sanity-check: you're not claiming that **Pn** (your notation) equals p[n] (my notation) are you? That would be very, very, bad.
> >
> > It's neither good nor bad. It's just plainly false: the Pn are for all the initial grids, not for a single one. They are much smaller than your p[n].
>
> I defined p[n] for non-indexed minimal puzzles.

That was an essential information, because the ratios are different for indexed and non-indexed. My formula is valid only for non-indexed.

> **Red Ed wrote:**
>
> Now please answer the question about the ratio p[n+1]/p[n].

If it is for non-indexed puzzles, then your ratio is equal to mine. Merely because it depends only on the floors at which your two non-indexed puzzles are.
(But don't forget that the distribution on B~, when restricted to the part of it coming from some fixed initial grid, is not the same for all the initial grids).

Back to top profile pm www

Red Ed

Posted: Sat Jul 11, 2009 5:53 pm Post subject: quote

Joined: 06 Jun 2005
Posts: 570

Alright, fine, we should average over grids & search paths. No quibble there.

Now, I think that you will believe the following to be true:

- Define m(n) to be the number of minimal puzzles (from all 6.67e21 solution grids) with n clues.
- Define PROC to be the top-down search process that picks a solution grid at random, then repeatedly removes redundant clues at random until a minimal puzzle is found.
- Define s(n) to be the probability that PROC stops at an n-clue minimal puzzle.
- **s(n) is proportional to m(n) / choose(81,n)**

Do you think it's true or false? And why?

Back to top profile pm

denis_berthier

Joined: 19 Jun 2007
Posts: 697
Location: Paris, France

Posted: Sat Jul 11, 2009 6:02 pm Post subject: quote edit

> **Red Ed wrote:**
>
> Alright, fine, we should average over grids & search paths. No quibble there.
>
> Now, I think that you will believe the following to be true:
>
> - Define m(n) to be the number of minimal puzzles (from all 6.67e21 solution grids) with n clues.
> - Define PROC to be the top-down search process that picks a solution grid at random, then repeatedly removes redundant clues at random until a minimal puzzle is found.
> - Define s(n) to be the probability that PROC stops at an n-clue minimal puzzle.
> - **s(n) is proportional to m(n) / choose(81,n)**
>
> Do you think it's true or false? And why?

I have no idea. Probably false. What's the relationship with the previous discussion?

Back to top profile pm www

Red Ed

Joined: 06 Jun 2005
Posts: 570

Posted: Sat Jul 11, 2009 6:28 pm Post subject: quote

Oh dear, are your probability skills a little rusty Denis? Mixture of notation in what follows; yours in bold.

Although not phrased in this way, you've defined **cf(n)** = u1 * choose(81,n) for some undefined constant u1 independent of n. (It doesn't matter that u1 is undefined).

Now, when you say **unbiased-mean(X) = sum(X(n) * on(n) * cf(n)) / sum(on(n) * cf(n))**, I imagine that you would be happy to go a little further and say that u2 * **X(n) * on(n) * cf(n)** is a "corrected" version of **X(n)**, for some undefined u2 independent of n. (Again, the undefined constant of proportionality doesn't matter here.) -- *do you agree* ❓

Now we could define **X(n)** = 1 for all minimal puzzles. That would mean that u2 * **on(n)** * choose(81,n) is a "corrected" estimate for the number of minimal puzzles with n clues; i.e. for m(n). Now your **on(n)** has mean proportional to my s(n), so this boils down to my statement above, that m(n) is proportional to s(n) * choose(81,n).

So that's where the claim came from. But I can see that the argument may have been a little non-obvious, so let's cut to the chase: please just try to answer the bit with the big yellow question mark by it. This is directly relevant to my journey to disproving your claims, as requested, so I'd be grateful for an answer.

Back to top profile pm

denis_berthier

Posted: Sat Jul 11, 2009 7:33 pm Post subject: quote edit

Joined: 19 Jun 2007
Posts: 697
Location: Paris, France

Red Ed wrote:

> Oh dear, are your probability skills a little rusty Denis? Mixture of notation in what follows; yours in bold.

Red Ed, this is not an exam and you are not the examiner. Don't worry about my probability skills.

Red Ed wrote:

> I imagine that you would be happy to go a little further and say that u2 * X(n) * on(n) * cf(n) is a "corrected" version of X(n), for some undefined u2 independent of n

Oh, it is that basic!
u1 doesn't matter in unbiased-mean(X) because it appears in the numerator and denominator.
As for whether u2 would matter or not, I can't imagine you're serious. Is 3 a corrected version of 1?

I've given a precise estimate of the distribution of minimal puzzles wrt to the number of clues. From which an estimate of the number of minimal puzzles with n clues could be deduced in a very basic way, IF we knew the number of minimal puzzles. We need no u2 or u3 or whatever you call it.

Back to top profile pm www

Red Ed

Joined: 06 Jun 2005
Posts: 570

Posted: Sat Jul 11, 2009 7:43 pm Post subject: quote

So, stroppiness aside, the answer to the big yellow question mark is "yes". Good. Therefore you must also agree that

s(n) is proportional to m(n) / choose(81,n)

Do please speak now if you somehow disagree with the bit in red.

Back to top profile pm

denis_berthier

Joined: 19 Jun 2007
Posts: 697
Location: Paris, France

Posted: Sat Jul 11, 2009 8:00 pm Post subject: quote edit

Red Ed wrote:

> So, stroppiness aside, the answer to the big yellow question mark is "yes". Good. Therefore you must also agree that
>
> *s(n) is proportional to m(n) / choose(81,n)*
>
> Do please speak now if you somehow disagree with the bit in red.

I didn't answer "yes".

You're completely reverting everything.

What you call s(n) is the distribution of clues in minimal puzzles produced by a

top-down generator, the observational counterpart of which is on(n).
What you call choose(81, n) is merely a rewriting of cf(n), unmanageable in practice.
From the observational fact on(n) and this mathematical function of n, one can get an estimation of the number of minimal puzzles with n clues, in the obvious way:
m(n) = MP * on(n) * cf(n) / sum(cf(n)), where MP is the total number of minimal puzzles.

It may seem to be the same thing as your formula, but it isn't: yours suggests that m(n) is known. Mine estimates m(n) from empirical evidence.

Last edited by denis_berthier on Sat Jul 11, 2009 8:39 pm; edited 3 times in total