SP 800-90B Refinements

Validation Process, Estimator Confidence Intervals, and Assessment Stability.

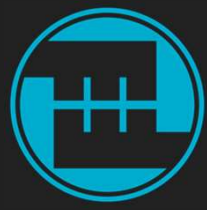
ICMC 2020

(Version 20200921)

This is the Greatest and Best Presentation in the World… Tribute.

This presentation covers three related topics:

- The creation of the recent IG 7.19, which helps to clarify CMVP's interpretation of SP 800-90B.
- Some problems in the statistical construction of some of the SP 800-90B estimators and a proposed fix (and why this isn't generally a big deal).
- A proposed statistical (meta-)assessment strategy that helps to reduce the variation in statistical assessment (limiting validation risk).



CMUF Entropy WG and IG 7.19

Process? What is it?

SP 800-90B becomes a requirement in November 2020, but it's been in development for a long time:

- Initial work started around 2002 within the ANSI X9 F1 working group. NIST requested comments on the draft ANSI X9.82 documents and held a workshop in July 2004.
- SP 800-90B Draft 1 was released in August 2012, and NIST held a workshop to discuss it in December 2012.
- SP 800-90B Draft 2 was released in January 2016, and NIST held a workshop to discuss it in May 2016.
- The final SP 800-90B was released in January 2018.
    - It had significant changes to address the received comments.
    - Some of these changes caused problems.

It's a Repeatable Way to Perform a Task (but that's not important right now…)

I've tried to understand this validation scheme as well as possible by:

- Doing entropy validation work using drafts of this standard since 2016.
- Implementing an SP 800-90B test tool in 2016, and updating this test tool as per the final document in January 2018.
- Providing substantial fixes and improvements to NIST's reference tool.

After years of use, I had some comments on the document:

- UL's SP 800-90B comments integrated feedback from myself (KeyPair, formerly UL), Ben Jackson (Sandia National Laboratories, formerly UL), Uyen Dinh (UL), and Jonathan Smith (Cygnacom).

Neverending SP 800-90B Comments

- Since the release of the final SP 800-90B I've sent in 10 versions of comments on the final document.
- (Sorry, 90B authors. ☹)
- The most recent comment set has about 55 comments in it.
- These informed many discussions with the SP 800-90B authors.
- I provided pull requests that implemented the suggested estimator fixes for the NIST tool.
- Unfortunately, there isn't a GitHub analog where we can resolve process and validation comments.

How to Proceed?

We [The CMUF Entropy WG] Have the Technology

- There is some history of IGs originating within the CMUF (e.g., IG G.19).
- There are some restrictions:
  - CMVP can interpret the technical requirements of SP 800-90B, resolve ambiguities, and provide additional (optional) flexibility, but it can't unilaterally impose additional technical requirements.
    - Note: The SP 800-90B document largely does not specify how a laboratory must verify compliance to the "shall" requirements of SP 800-90B, so this IG can also provide this sort of guidance.
  - The next versions of SP 800-90B and 90C are already being authored, and conflicts with these future documents would be unhelpful.
  - Not all labs have chosen to participate within the CMUF Entropy WG (so the review phase for this IG was more exciting than we anticipated).
  - We can author a proposal but, fundamentally, CMVP gets the final say.

Though the Road May Wind…
Yea Your Hearts Grow Weary…

- The first draft was done on August 26, 2019.
- There were about 40 different public drafts of this IG.
- Development of this IG was coordinated with CMVP, the NIST 90B authors, and the CMUF Entropy WG participants.
- During the review phase, CMVP shared about 60 comments from laboratories which included significant comments from various vendors.
- The CMUF Entropy WG provided suggested comment resolutions.
- CMVP made some additional changes based on internal NIST feedback.
- The final IG was published on August 28, 2020.

Highlights (1): You Will Find a Fortune…

- Resolution #10: “Combining the outputs of the noise source copies… shall be considered part of the digitization process, and so Resolution #1 shall apply.”
- Resolution #1: “…the vendor shall justify why all processing occurring within the digitization process does not conceal noise source failures from the health tests or obscure the statistical properties of the underlying raw noise output from this digitization process.”

Highlights (2):
... Though it Will Not Be the Fortune You Seek

- Resolution #5: "… if a non-vetted conditioning component retains state and the primary noise source is non-independent, then the vendor shall provide mathematical evidence that the conditioning component's entropy output is not below its assessed value."
- Resolution #13: "The technical argument supporting the expected $H_{\text{submitter}}$ value shall be based on the vendor's description of the source of unpredictability within the noise source and how the noise source outputs vary depending on this identified unpredictability. Statistical testing may be used to establish parameters referenced within this argument, but the $H_{\text{submitter}}$ value shall not be the result of some general statistical testing process that does not account for the design of the noise source."

Highlights (3): And, Oh, So many Startlements…

- Additional Comment #3: “The tester shall verify that each conditioning component’s implementation is fully consistent with the component’s design. This verification shall be performed by means of either running a computerized test developed for testing just the conditioning component (separate from the statistical testing of the noise source) or by the code review. The lab shall describe in the Entropy Test Report submitted to the CMVP the chosen method for verifying the correctness of each conditioning component’s implementation… The requirement for the tester to verify the correctness of a conditioning component’s implementation includes the testing or a code review of the functionality of the component that may not be addressed by the CAVP testing of the approved algorithms.”

Confidence Intervals in the SP 800-90B Estimators

A Motivating Comment

- A comment from the audience at an ICMC 2019 Entropy Testing Panel discussion: “Min Entropy does not satisfy the Central Limit Theorem.”

  Q: Gee, Josh, what do you want to do today?

  A: The same thing that we do every day, Pinky. Statistical analysis and millions of rounds of simulation!

Confidence Intervals

- Most estimators (all but the Markov Estimator) produce 99% two-sided confidence intervals for an observed statistic and then use one of the confidence bounds to produce a lower bound for the min entropy assessment for that noise source.
- A confidence interval construction procedure yields “correct” confidence intervals at a certain confidence level if, across many trials, the proportion of confidence intervals that contain the true value of the parameter is equal to the confidence level.
- We want a balanced confidence interval, that is the probability that the true value is less than or equal to the upper confidence interval bound is 99.5%, and similarly the probability that the true value is greater than or equal to the lower bound is 99.5%.

How Confident?

- In empirical settings, how can we establish the “true value”?
  - If we knew this, we wouldn’t be running the estimator in the first place!
- We do know that 0.5% of the generated lower confidence interval bounds are supposed to be too high, and 0.5% of the generated upper confidence interval bounds are supposed to be too low (because we are looking for the 99% CI).
- In the SP 800-90B estimators, the confidence interval bounds are produced using increasing functions of the observed statistic, so the medians of the produced confidence lower and upper bounds should correspond to the 0.5% percentile and the 99.5% percentile of the observed statistics (respectively).
- This is closely related to how Bootstrap Confidence Intervals are created using a bootstrap distribution.

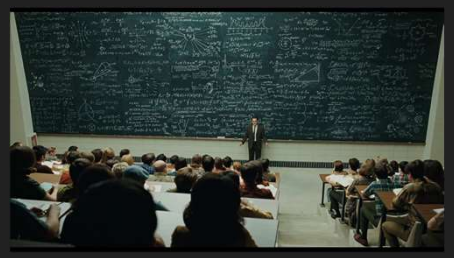
General Observations

- All estimators produce incorrect confidence intervals when operated on non-IID data, and some of the estimators produce incorrect confidence intervals with all data (both IID and non-IID).
- The problems can be broadly classified as follows:
  1. The reference distribution is presumed to be binomial, but it isn't.
     - The MCV, t-Tuple, and LRS estimators for all data, and all prediction estimators for non-IID data.
  2. The estimators use the Central Limit Theorem without satisfying its hypotheses.
     - The Collision and Compression Estimators (for non-IID data).
  3. The Markov Estimator's results are not directly comparable to the results of the other estimators, as it does not produce its estimate using confidence intervals.

The Non-IID Case

- For non-IID noise sources, the generated bounds are not expected to have the desired confidence.
  - The confidence associated with the generated bound is source-dependent.
  - Our empirical testing suggests that for non-IID sources the estimators generally produce estimates with a lower confidence than intended, thus the produced min entropy estimates are **higher** than intended.

Some Confidence Interval Calculations



The MCV Estimator (Step 2), t-Tuple Estimator (Step 4), and LRS Estimator (Step 4) all produce a confidence interval upper bound using something like this:

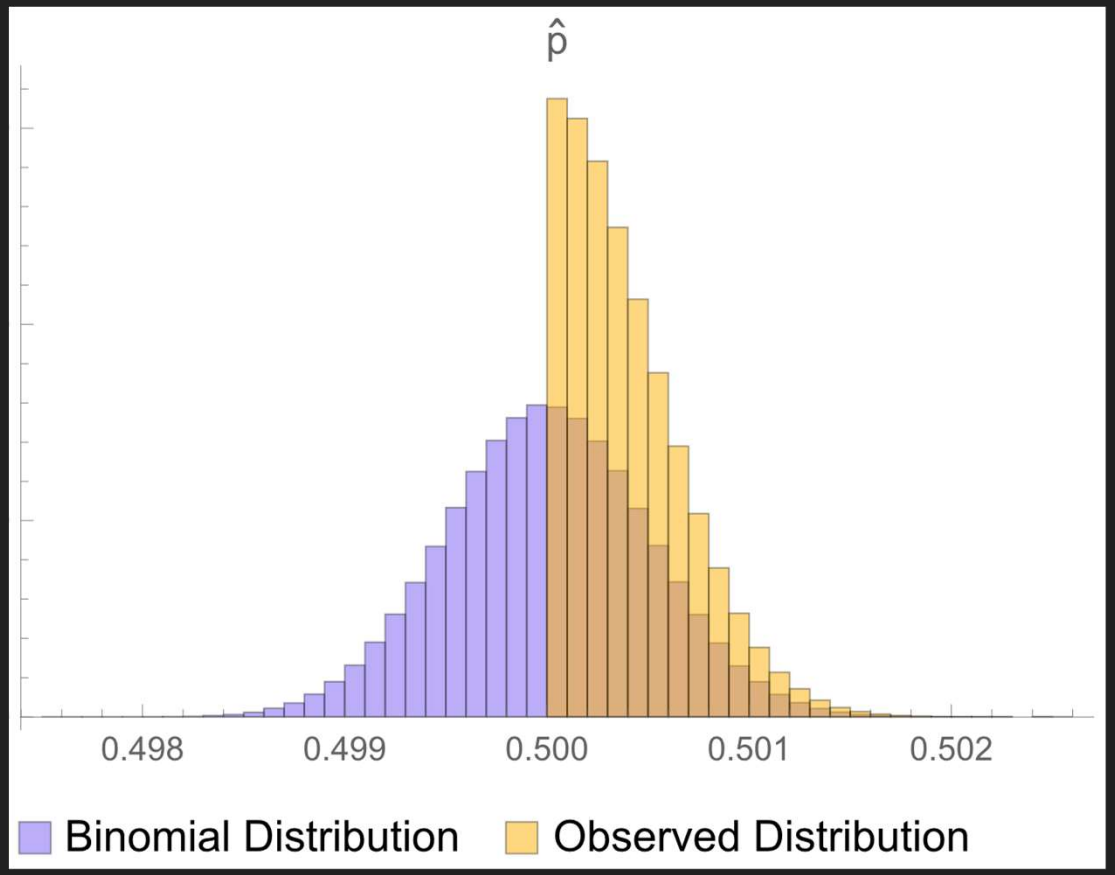
$$p_u = \min\left(1, \hat{p} + z_{0.995}\sqrt{\frac{\hat{p}(1-\hat{p})}{L-1}}\right)$$

- Assumptions:
  - The $\hat{p}$ values are binomially distributed.
  - This binomial distribution can be reasonably approximated using the Normal Distribution (this is an implicit application of the Central Limit Theorem).
- This **upper bound** for $\hat{p}$ translates to a **lower bound** for the assessed entropy.

MCV Estimator

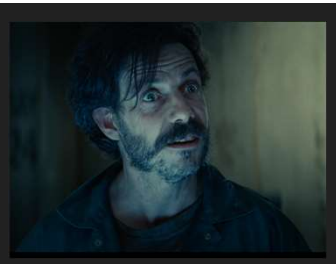
- There is a substantial difference between the presumed distribution and the actual distribution.
- In general, the maximum of several binomially distributed variables is not binomially distributed.
- These variables are not independent.
  - The result is actually the maximum of a multinomial distribution.
  - In particular, note that the most common value can never occur fewer than $\lceil L/k \rceil$ times.
  - This is most clear in the binary case, where the resulting distribution arises from a “fold-and-sum” of the reference binomial distribution.

MCV Estimator



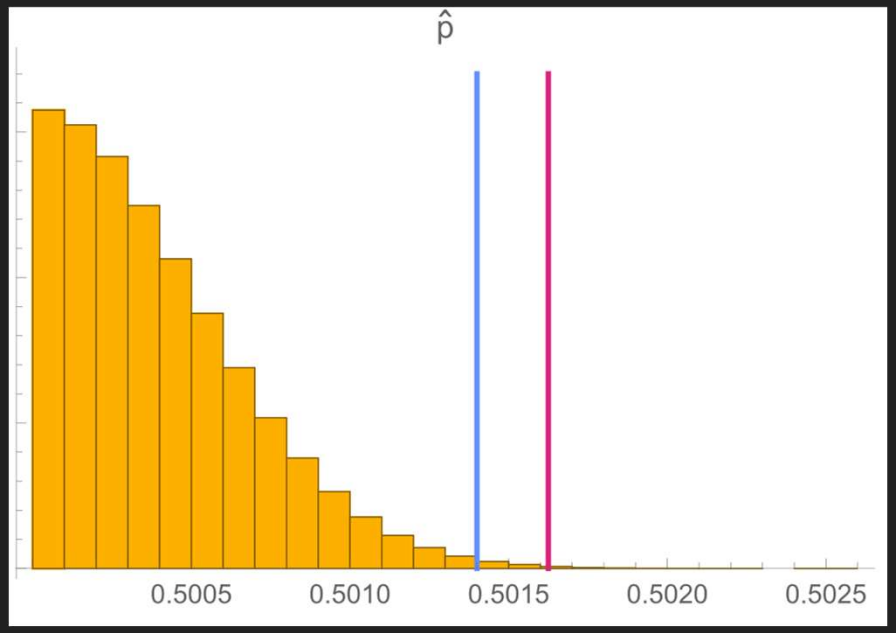
Ideal IID Binary Source
(n=1,001,000)

MCV Estimator



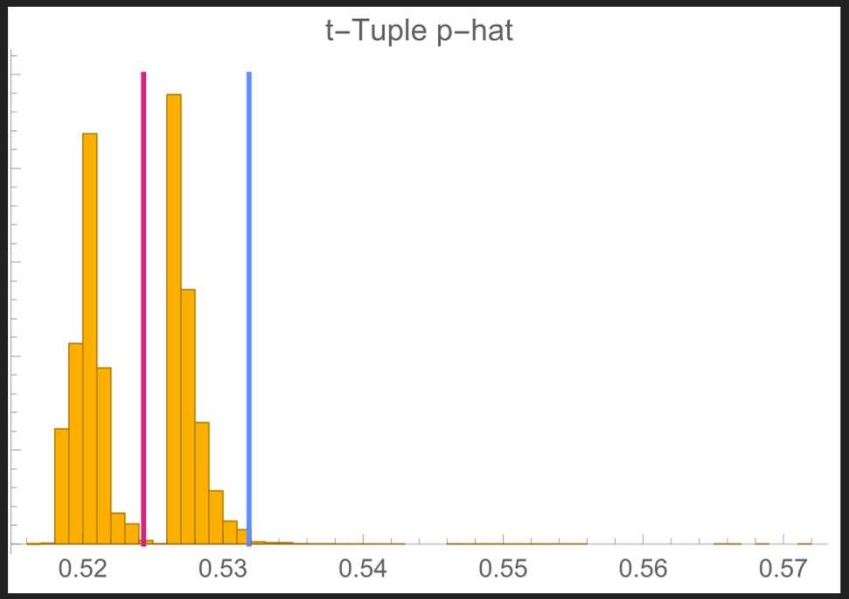
- The magenta line is the median of the calculated confidence interval bounds, and the blue line is the empirically observed confidence interval bound (the 99.5th percentile) for an IID ideal binary noise source.
- This results in a higher confidence than intended, so this estimator produces a **lower** min entropy assessment than a similar estimator that worked as intended.

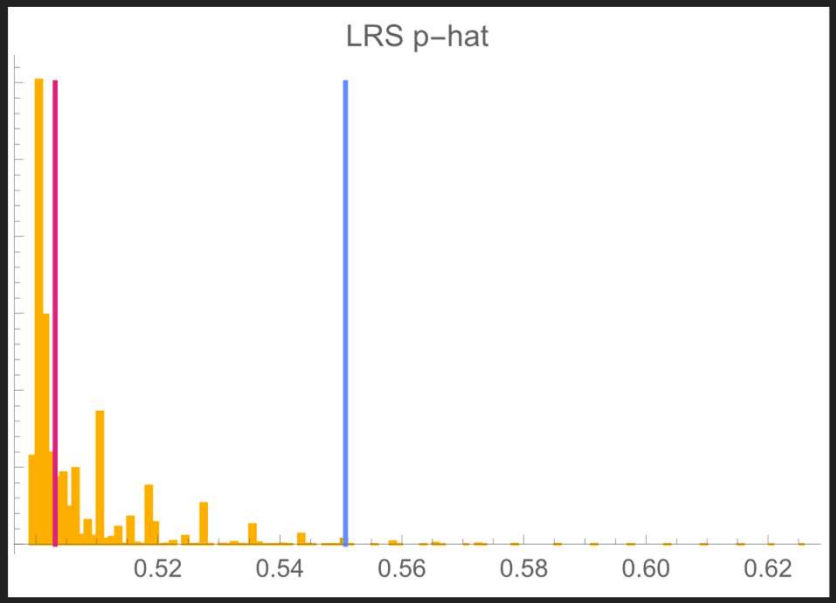
Ideal IID Binary Source
(n=1,001,000)



t-Tuple and LRS Estimators

Something else seems to be going on for these estimators:





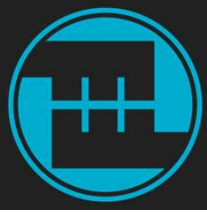
Ideal IID Binary
Source (n=1,001,000)

The t-Tuple and LRS Estimator Issues

- For the t-Tuple and LRS Estimators, we have larger-scale versions of the same problems that the MCV estimator had, and some additional issues.
  - For the t-Tuple Estimator, the most common i-tuple cannot have fewer than $\lceil (L - i + 1)/k^i \rceil$ occurrences.
  - The underlying distributions (before taking the maximum) aren't expected to be binomial, because the substrings overlap.
    - In the t-Tuple Estimator, with the count of (possibly overlapping) occurrences of a fixed i-tuple, each trial isn't independent.
    - In the LRS Estimator, the (nominally unbiased) calculation of the W-tuple collision probability estimate requires that each W-length substring be IID.
  - The maximum of these distributions isn't expected to have a binomial distribution.
- Empirically, these seem to result in a lower confidence than intended, so this estimator produces a **higher** min entropy assessment than a similar estimator that worked as intended.

How to Fix this?

- These problems arise because the estimators presume a particular underlying distribution that isn't correct.
- Proposed solution: Don't do that. Instead:
  1. Perform many rounds of testing.
  2. Empirically establish the relevant confidence interval bound for each estimator parameter through bootstrapping.
  3. Profit! (Also, very big files.)

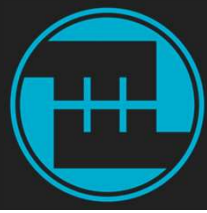


My Complications…

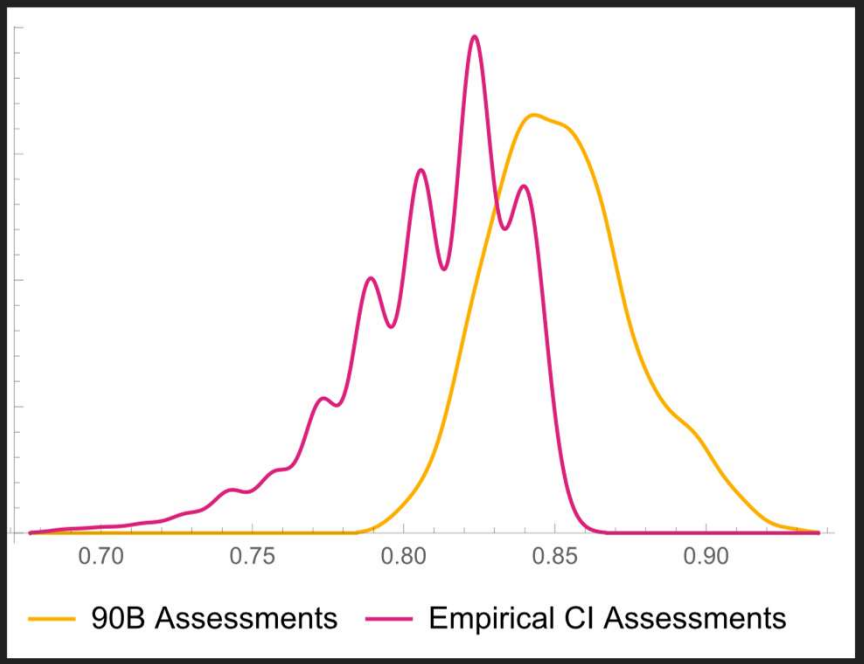
- The 2018 Markov test no longer uses a confidence interval.
  - The 2016 and 2012 drafts did, but it resulted in artificially low min entropy estimates unless a large amount of data was used.
  - We can fix this issue using empirically selected parameter values, making the results comparable to the other estimators.
- How many rounds?
  - For a 99.5th percentile, we would like **at least** 1,000 sets of 1 million samples.

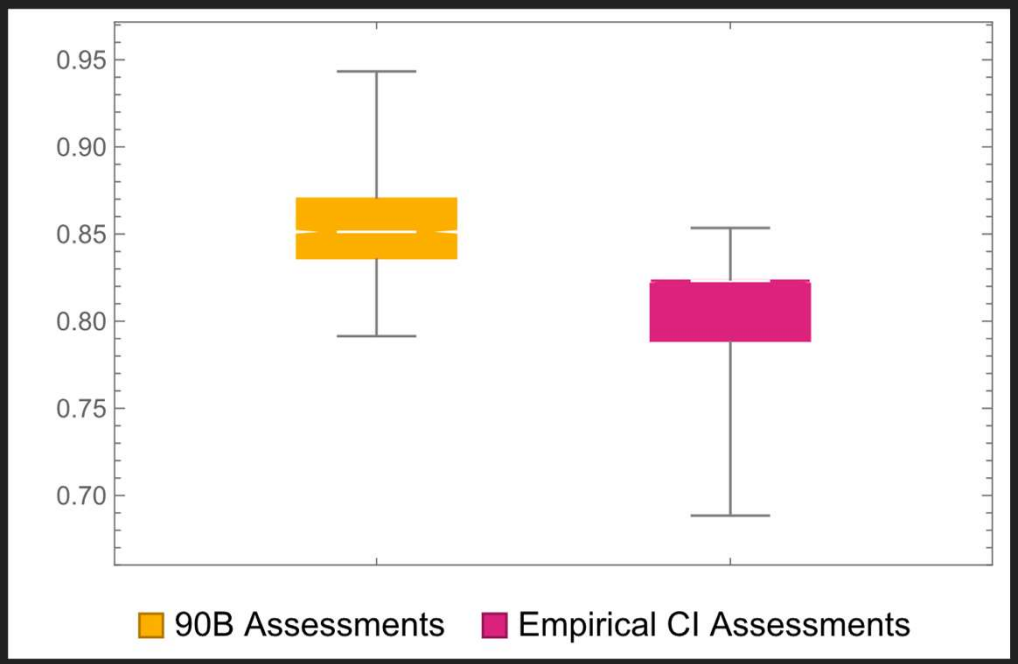
Had Complications…

- The various predictor estimators all find a P_{global} value (using a confidence interval bound) and a P_{local} value (not using a confidence interval bound). What to do here?
- We could build a confidence interval for the longest run, but that makes P_{local} likely to dominate the predictor.
- The original predictor only uses the P_{local} calculation in really aberrant conditions.
- We want a source with most likely symbol probability P_{local} to have a 99% chance that there is **no run** of the observed longest run length across all r of the test rounds.
  - Take the maximum run length observed across all testing rounds, and calculate P_{local} as the value such that there is a 99% chance that there is no run of the observed maximal length after r rounds (so, do a binary search for 0.99^r instead of 0.99).



Empirical CI Estimator Results





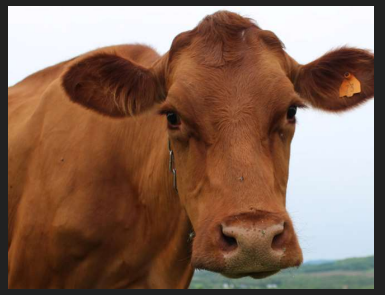
Simulated 1GHz Ideal RO Sampled at Approximately 1MHz with a Per-Cycle Period Jitter of $\sigma \approx 11.8$ps
(n= 1,000)

It's OK To Start Thinking…
But You Gotta Know When To Stop!

- The use of incorrect confidence interval bounds affects the assessment for all sources.
- This sounds bad, but…
  - Use of bounds associated with an unintended confidence level is still **more conservative** than not attempting to bound the result at all.
  - The resulting error isn't large for any source that we've tested (the observed error has been on the order of 5%).

Strategy for Assessment Stability

How Now?



- For a fixed noise source with fixed entropy-relevant parameters, a min entropy estimator's assessment conforms to some fixed distribution.
- It's hard to comment on this distribution given only a single result.
- Sometimes this distribution is unhelpfully wide, which risks intermittent "downstream" validation failures.
- How do we get a more meaningful and stable assessment?

What's That All About?

General idea: Do many assessments instead of just one.

- For a fixed entropy estimator, source, and entropy relevant parameters, the produced entropy estimate follows a distribution.
- Calculate some summary statistic for this distribution.
- Use this summary statistic as an overall assessment for the associated estimator.
- Take the minimum overall assessment across all the estimators.
- Result!

It's About This Long

How many assessments?

- The more assessments you do the more stable the result is, and the less bootstrap confidence intervals reduce the result.
- Doing more assessments requires getting more data.
- Anything over about 20 rounds will be helpful in reducing variation, but I suggest at least 100 rounds.

It's About This Wide

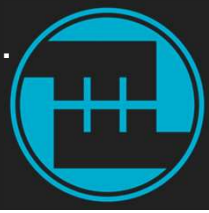
Which statistic?

- We want something that conveys important information about the distribution.
- How about the minimum result or some low quantile, like the 1st percentile?
  - This approach functionally punishes the vendor for supplying more data.
  - In most cases, this is effectively taking a low quantile of results that are themselves intended to be lower quantiles.
  - In most cases, this would push the confidence well above the desired 99% level.

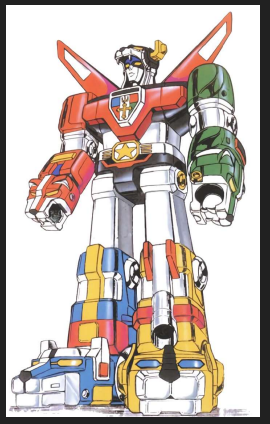
It's About This Country

Which statistic?

- The median
  - Is (in some sense) central.
  - Is fairly stable.
  - Doesn't take many samples to estimate.
- We Have a Winner!
- … Well, Mostly
  - The median of Markov Estimator results isn't directly comparable to the medians of all the other estimators' results (for which confidence intervals are used).
  - For the Markov Estimator, take the 0.5^{th} percentile.
  - This makes the result directly comparable with all the other estimator results.



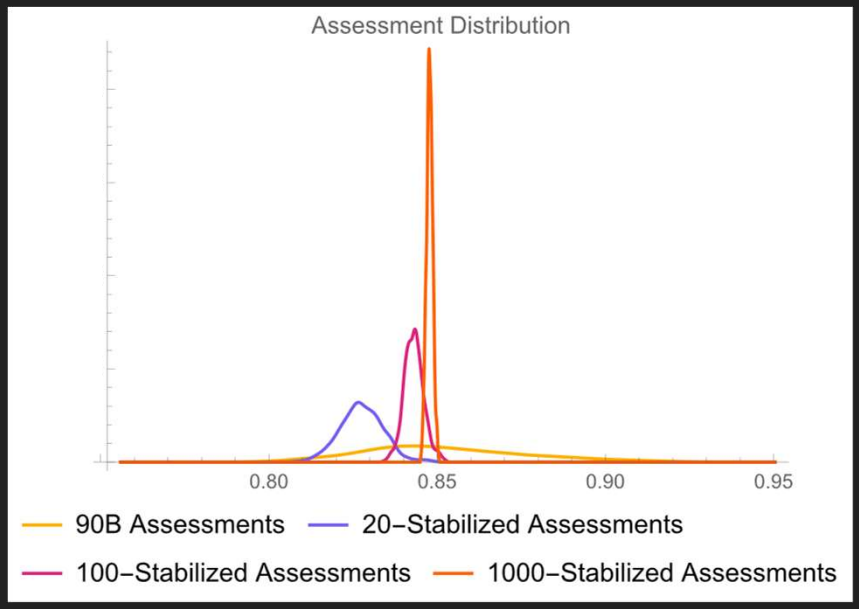
About Which We're Singing About

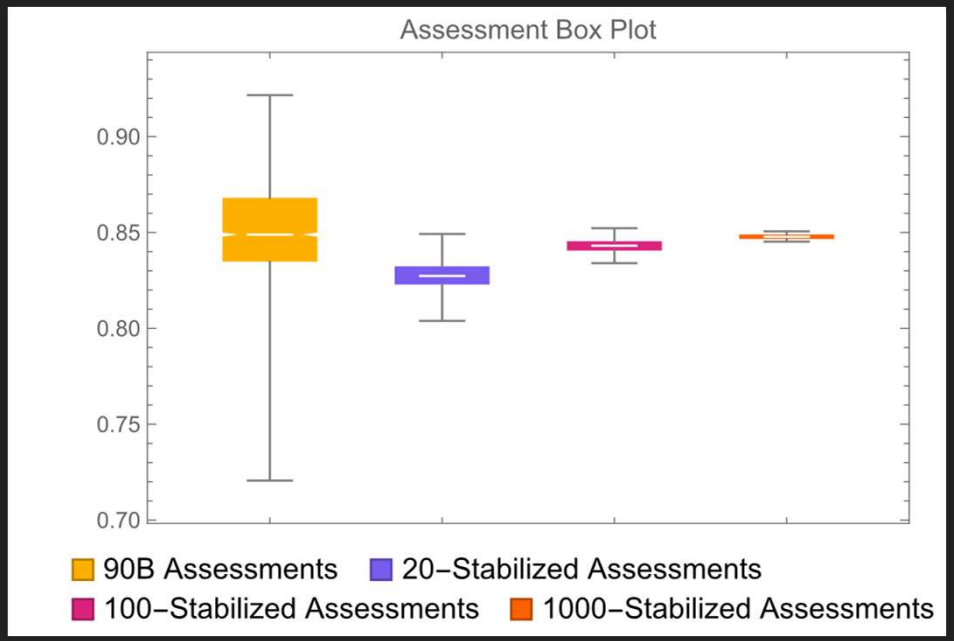


Summary:

- Perform r rounds of testing.
- Summarize the results of the r rounds of testing for each estimator.
  - For all estimators other than the Markov Estimator, bootstrap the median.
  - For the Markov Estimator, bootstrap the 0.5th percentile.
- Take the minimum of the per-estimator overall results.
- This is the r-stabilized assessment.

Example Results

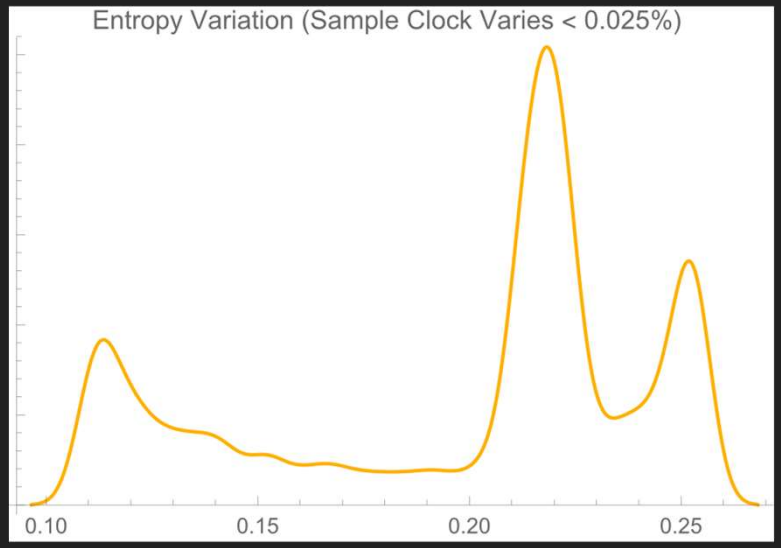




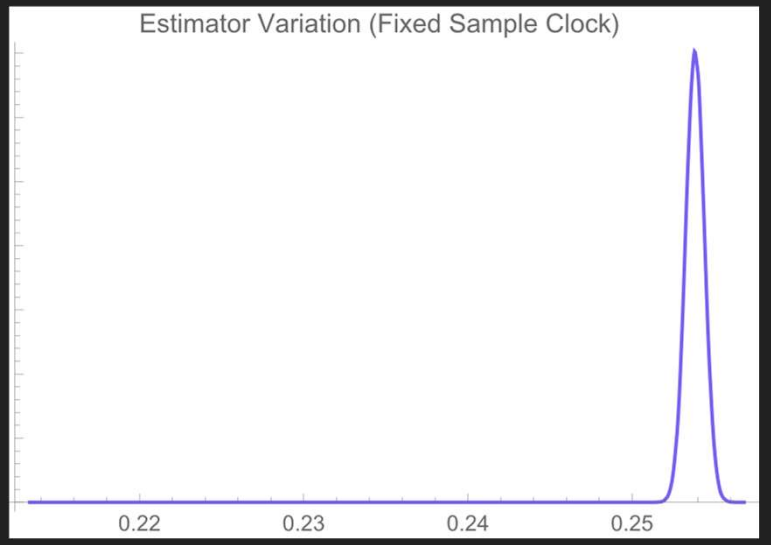
Simulated 1GHz Ideal RO Sampled at Approximately 1MHz with a Per-Cycle Period Jitter of $\sigma \approx 11.8$ps
(n=1,000)

Right, That's Bad. Okay, Important Safety Tip... Thanks Egon!

- Not all assessment variation is due to estimator variation.
- Many (most?) noise sources have entropy-relevant parameters that can affect the actual min entropy produced.



$\neq$



Simulated 1GHz Ideal RO Sampled at Approximately 1MHz with a Per-Cycle Period Jitter of $\sigma \approx 2.8$ps (n=250,000)

He's No Fun, He Fell Right Over

- How can this be compliantly integrated into the SP 800-90B assessment process?
- Calculate the r-stabilized assessment.
- Perform a single "Large Block Assessment" on the full data set.
- Take the minimum of the r-stabilized assessment and the "Large Block Assessment".
- This is mappable to the SP 800-90B process (the Large Block Assessment **is** the SP 800-90B assessment process, but with more data than required by SP 800-90B).
- If the r-stabilized assessment is lower than the Large Block Assessment, this can be thought of as reducing $H_{submitter}$.

BUT WAIT, THERE'S MORE!!!

Carnac… Will Give That a Shot

I can read your minds!

- An audience member who just ate some yogurt is thinking “All this testing requires access to a high quality statistical package and computer algebra system! That’s impractical!”
- An audience member with a birthmark the shape of North Carolina’s 2013-2017 12th congressional district is thinking “We would like to be able to perform such testing, but we don’t want to do a bunch of coding!”
- An audience member with soulful eyes and a distaste for sauerkraut is thinking “My heart weeps for want of stable min entropy estimates, but I fear that I am destined for lesser things.”
- … and many audience members are thinking “I regret not bringing more coffee to this presentation…”

Cue The Drumroll!
All Right, Open Your Boxes!

- OK, that isn't really practical. Try GitHub.
  - https://www.github.com/KeyPair-Consulting/Theseus
- This is the SP 800-90B assessment tool that I wrote while working at UL.
- This tool is now owned by KeyPair.
- KeyPair generously agreed to release this tool as open source.
- The code here is mostly licensed under the 3-Clause BSD license, with some small parts more liberally licensed.
- You can use this tool to perform all the testing approaches that I presented today.

Questions?