Minimizing errors on entropy health tests

The joy of oversampling

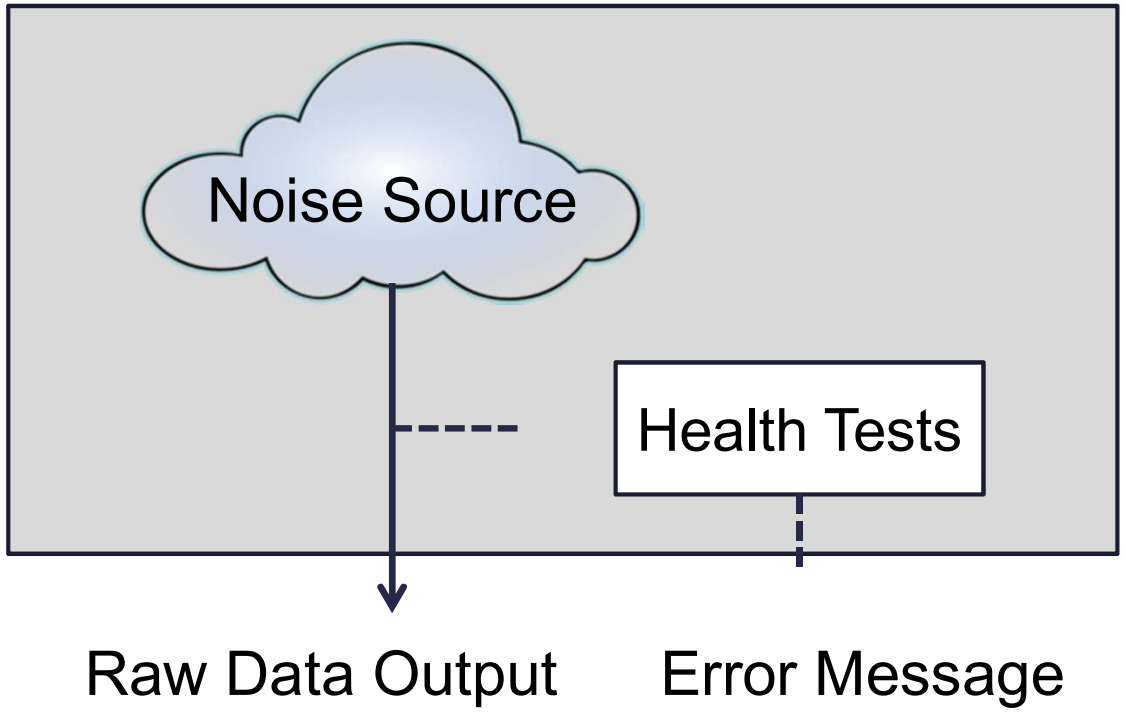
Scott Fluhrer

May 3, 2016

Agenda

- NIST Health Test Model
- Positive and Negative Failures
- A Better Way
- Recommendations

NIST Entropy Health Test Model (simplified)



Reasons we run Health Tests

Unlike the rest of the system, Known Answer Tests don't work on noise sources.

We run Health Tests to verify that the noise source is functioning properly:

- To catch degrading hardware
  - Infant failures or hardware that's past its 'best-by' date
- To catch environmental-based attacks
  - An attacker may be chilling the entire system to -40°

Potential Failures in this System

False Negatives

- Not detecting a problem when there is one
  - Obviously, it is important to minimize this possibility

False Positives

- Claiming there is a problem on a working system
  - We'd like to minimize this as well

False Positives

To keep the false negative probability low, the current 800-90B draft asks that the false positive rate be at least 2^{-50}

This may appear to be an acceptably low probability, except:

- There may be billions of IOT devices
- Each device may access its entropy source many times over its lifetime
- Slightly degraded devices may have a significantly higher rate of false positives

Problems with False Positives

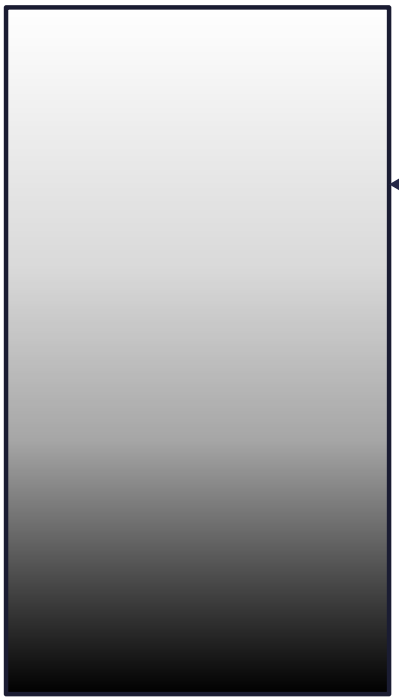
A high rate of false positives means that the manufacturers will try to just log an error and continue running

Error messages have a high likelihood of being ignored

A high false positive rate will mean that service personnel will ignore these errors

If these problems were required to have a low false negative rate, this would be an acceptable trade-off.

NIST Model vs Proposed New Model



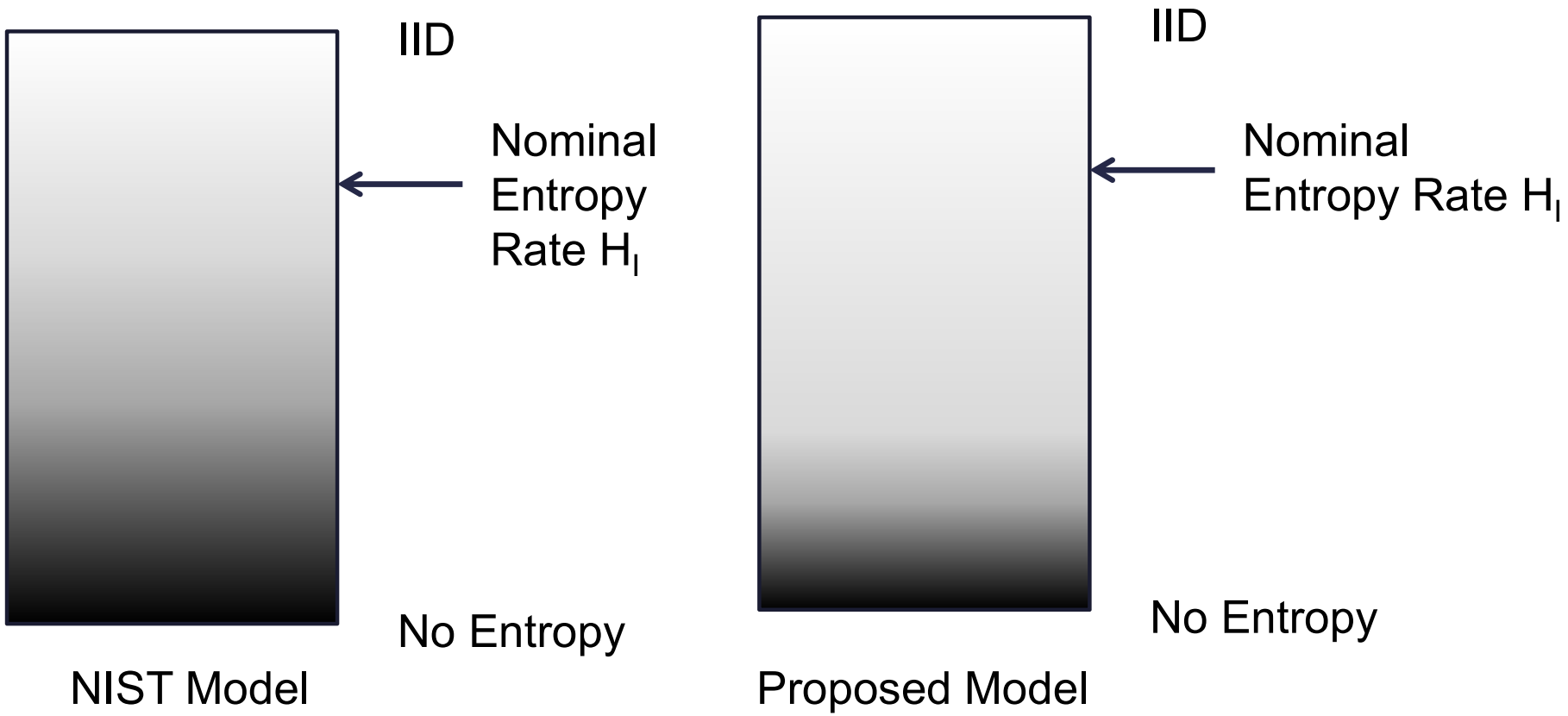
IID

Nominal
Entropy
Rate H_I

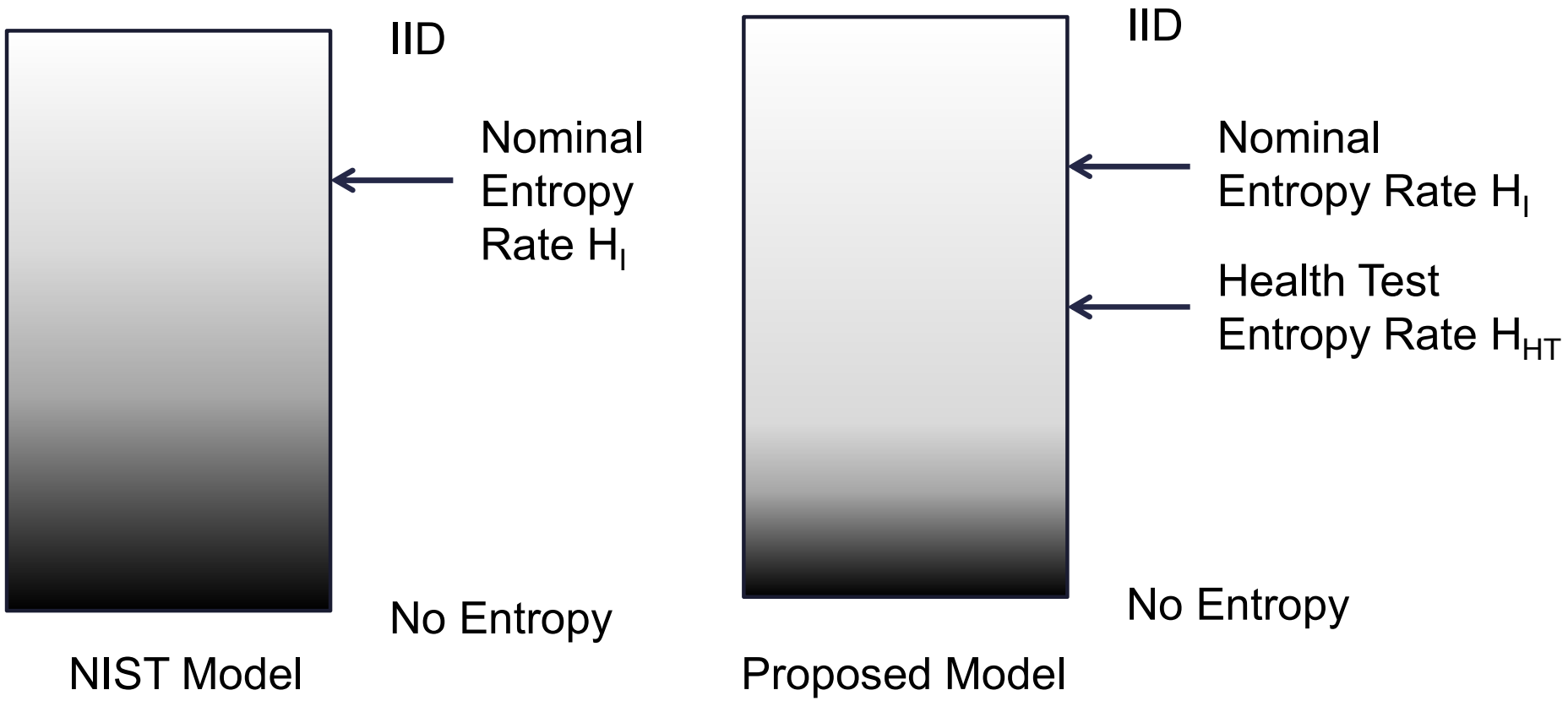
No Entropy

NIST Model

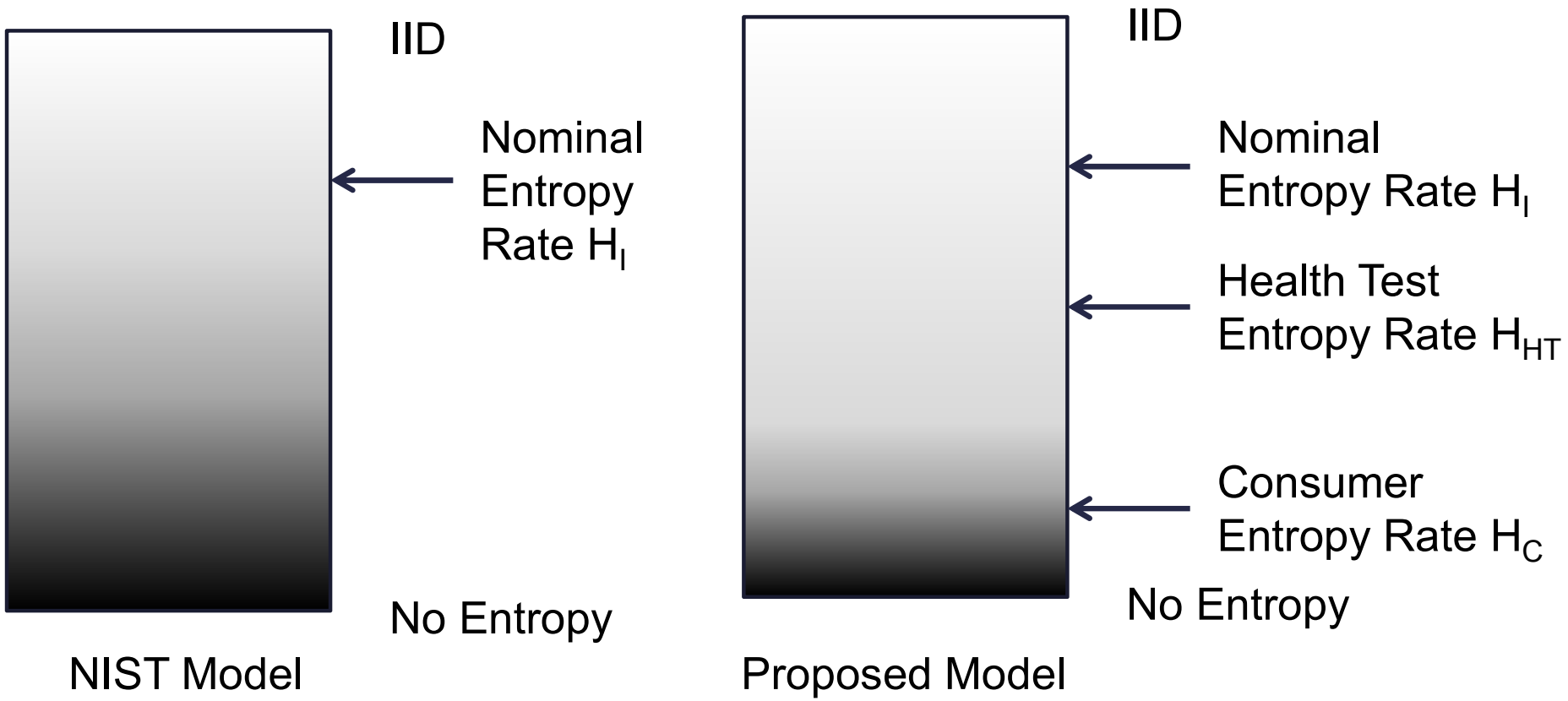
NIST Model vs Proposed New Model



NIST Model vs Proposed New Model



NIST Model vs Proposed New Model



Entropy Parameters

H_I Nominal Entropy Rate (as Current)

H_{HT} Entropy Rate Used for Health Tests

H_C Entropy Rate Given to Consumer

$H_I > H_{HT}$ improves false positive rate

$H_{HT} > H_C$ improves false negative rate

What are the costs?

Low false positive and false negative rate – what could be wrong?

- This uses more entropy samples than required

  Why isn't this a deal-breaker?

  Well, in some environments, sampling the entropy source is cheap

  It's not that much more

- This also assumes a health test that will actually catch problems

Recommendations

- NIST should explicitly allow developers to tune their health tests for an entropy rate lower than H_I
  - This is so that a reference lab will not decide to reject it
- NIST should allow developers the option to use H_I, H_{HT}, H_C (and document the values they declare).
  - This is so buyers of noise sources can make more informed decisions
- More research on better health tests

Thank you.