

fn sample_geometric_exp_slow

Michael Shoemate

May 26, 2025

This proof resides in “contrib” because it has not completed the vetting process.

Proves soundness of `fn sample_geometric_exp_slow` in `mod.rs` at commit `0be3ab3e6` (outdated¹). This proof is an adaptation of subsection 5.2 of [CKS20].

Vetting history

- Pull Request #519

1 Hoare Triple

Precondition

$x \in \mathbb{Q} \wedge x > 0$

Pseudocode

```
1 def sample_geometric_exp_slow(x) -> int:
2     k = 0
3     while True:
4         if sample_bernoulli_exp(x): #
5             k += 1
6         else:
7             return k
```

Postcondition

For any setting of the input parameter x such that the given preconditions hold, `sample_geometric_exp_slow` either returns `Err(e)` due to a lack of system entropy, or `Ok(out)`, where `out` is distributed as $Geometric(1 - \exp(-x))$.

2 Proof

Assume the preconditions are met.

Lemma 2.1. `sample_geometric_exp_slow` only returns `Err(e)` when there is a lack of system entropy.

Proof. The preconditions on x satisfy the preconditions on `sample_bernoulli_exp`, so by its definition, it only returns an error if there is a lack of system entropy. The only source of errors is from this function, therefore `sample_geometric_exp_slow` only returns `Err(e)` when there is a lack of system entropy. \square

¹See new changes with `git diff 0be3ab3e6..8a5d93c rust/src/traits/samplers/cks20/mod.rs`

Theorem 2.2. [CKS20] If the outcome of `sample_geometric_exp_slow` is `Ok(out)`, then `out` is distributed as $Geometric(1 - \exp(-x))$. That is, $P[\text{out} = k] = \exp(-x)(1 - \exp(-x))^k$

Proof. The distribution of the i^{th} boolean returned on line 4 is $B_i \sim Bernoulli(\exp(x))$, because the preconditions on `x` satisfy the preconditions for `sample_bernoulli_exp`.

$$\begin{aligned}
P[\text{out} = k] &= P[B_1 = B_2 = \dots = B_k = \perp \wedge B_{k+1} = \top] \\
&= P[B_{k+1} = \top] \prod_{i=1}^k P[B_i = \perp] && \text{All } B_i \text{ are independent.} \\
&= \exp(-x)(1 - \exp(-x))^k
\end{aligned}$$

□

Proof. 1 holds by 2.1 and 2.2.

□

References

[CKS20] Clément L. Canonne, Gautam Kamath, and Thomas Steinke. The discrete gaussian for differential privacy. *CoRR*, abs/2004.00010, 2020.