CompositionMeasure for Approximate<ZeroConcentratedDivergence>

Michael Shoemate

This proof resides in **"contrib"** because it has not completed the vetting process.

Proves soundness of the implementation of `CompositionMeasure` for `Approximate<ZeroConcentratedDivergence>` in `mod.rs` at commit f5bb719 (outdated[1]).

1 Hoare Triple

Precondition

Compiler-Verified

Types matching pseudocode.

Caller-Verified

None

Pseudocode

```
class CompositionMeasure(ApproximateZeroConcentratedDivergence):
    def composability(  #
        self, adaptivity: Adaptivity
    ) -> Composability:
        return Composability.Sequential

    def compose(self, d_mids: Vec[Self_Distance]) -> Self_Distance:
        rho_g, del_g = 0.0, 0.0
        for rho_i, del_i in d_mids:
            rho_g = rho_g.inf_add(rho_i)
            del_g = del_g.inf_add(del_i)
        return rho_g, del_g
```

Postcondition

Theorem 1.1. `composability` returns `Ok(out)` if the composition of a vector of privacy parameters `d_mids` is bounded above by `self.compose(d_mids)` under `adaptivity` adaptivity and `out`-composability. Otherwise returns an error.

Proof. By the postcondition of `InfAdd` we have that $\sum_i \texttt{d\_mids}_i \leq \texttt{compose(d\_mids)}$, where the summation is applied independently to rhos and deltas, and the comparison applies to both the global rho and global delta.

[1] See new changes with `git diff f5bb719..89a18ad rust/src/combinators/sequential_composition/mod.rs`

Adaptivity	Sequential	Concurrent
Non-Adaptive	Lemma 8.2[BS16]	None
Adaptive	Lemma 8.2[BS16]	None
Fully-Adaptive	Theorem 1[WRRW23]	None

This table is reflected in the implementation of `composability` on line 2.

□

References

[BS16] Mark Bun and Thomas Steinke. Concentrated differential privacy: Simplifications, extensions, and lower bounds, 2016.

[WRRW23] Justin Whitehouse, Aaditya Ramdas, Ryan Rogers, and Zhiwei Steven Wu. Fully adaptive composition in differential privacy, 2023.