

# impl SelectionMeasure for RangeDivergence

Michael Shoemate

March 18, 2025

## 1 Hoare Triple

### Precondition

#### Compiler-verified

- Associated Constant `ONE_SHOT`
  - `ONE_SHOT` is true if the measure supports one-shot top- $k$  composition.
- Associated Type `RV`
  - `RV` must implement trait `InverseCDF`.
- Method `random_variable` *Types consistent with pseudocode.*
- Method `aprivacy_map` *Types consistent with pseudocode.*

#### Caller-verified

- Method `random_variable`
  - `scale` is positive (*cannot be null due to `FBig` dtype*).
- Method `privacy_map`
  - `d_in` is non-null and non-negative.
  - `scale` is non-null and non-negative.

### Pseudocode

```
1 class RangeDivergence(SelectionMeasure):
2     ONE_SHOT = True
3     RV = GumbelRV
4
5     @staticmethod
6     def random_variable(shift: FBig, scale: FBig) -> GumbelRV:
7         return GumbelRV(shift=shift, scale=scale)
8
9     @staticmethod
10    def privacy_map(d_in: f64, scale: f64, k: usize) -> f64:
11        if d_in < 0:
12            raise ValueError("input distance must be non-negative")
13
14        if scale.is_zero():
15            return f64.INFINITY
16
17        return d_in.inf_div(scale).inf_mul(f64.inf_cast(k))
```

## Postcondition

**Theorem 1.1.** The implementation is consistent with all associated items in the **SelectionMeasure** trait.

1. Associated Constant **ONE\_SHOT**
2. Associated Type RV
3. Method **random\_variable**
4. Method **privacy\_map**

*Proof of valid associated constant: **ONE\_SHOT**.* Since the proof of **privacy\_map** only holds if  $k$  is less than two, **ONE\_SHOT** is defined to be false.  $\square$

**Definition 1.2.** A random variable follows the Exponential distribution if it has density

$$f(x) = \frac{1}{\beta} e^{-z} \tag{1}$$

where  $z = \frac{x-\mu}{\beta}$ ,  $\mu$  is the shift (location) parameter and  $\beta$  is the scale parameter.

*Proof of valid associated type: **RV**.* The associated type RV is defined as **ExponentialRV**, which represents a random variable following the Exponential distribution 1.2. The compiler verifies that **ExponentialRV** implements the **InverseCDF** trait.  $\square$

*Proof of valid method: **random\_variable**.* By the precondition on **scale** being positive, **random\_variable** returns a valid instance of **ExponentialRV**.  $\square$

*Proof of valid method: **privacy\_map**.* To be merged from: <https://github.com/opensp/opensp/pull/1678/files>  $\square$