MLE of a Pareto Distribution

You are creating artwork with different sized circles which follow a Pareto distribution:

$X \sim \text{Pareto}(\alpha)$

A Pareto distribution is defined by a single parameter α and has PDF

$$f(x) = \frac{\alpha}{x^{\alpha+1}}$$

You would like the alpha in your artwork to match that of sand in your local beach. You go to the beachand collect 100 particles of sand and measure their size. Call the measured radii $x_1, \ldots, x_{100}$:

```
observations = [1.677, 3.812, 1.463, 2.641, 1.256, 1.678, 1.157,
1.146, 1.323, 1.029, 1.238, 1.018, 1.171, 1.123, 1.074, 1.652,
1.873, 1.314, 1.309, 3.325, 1.045, 2.271, 1.305, 1.277, 1.114,
1.391, 3.728, 1.405, 1.054, 2.789, 1.019, 1.218, 1.033, 1.362,
1.058, 2.037, 1.171, 1.457, 1.518, 1.117, 1.153, 2.257, 1.022,
1.839, 1.706, 1.139, 1.501, 1.238, 2.53 , 1.414, 1.064, 1.097,
1.261, 1.784, 1.196, 1.169, 2.101, 1.132, 1.193, 1.239, 1.518,
2.764, 1.053, 1.267, 1.015, 1.789, 1.099, 1.25 , 1.253, 1.418,
1.494, 1.015, 1.459, 2.175, 2.044, 1.551, 4.095, 1.396, 1.262,
1.351, 1.121, 1.196, 1.391, 1.305, 1.141, 1.157, 1.155, 1.103,
1.048, 1.918, 1.889, 1.068, 1.811, 1.198, 1.361, 1.261, 4.093,
2.925, 1.133, 1.573]
```

Derive a formula for the MLE estimate of α based on the data you have collected.

Writing the Log Likelihood Function

The first major objective in MLE is to come up with a log likelihood expression for our data. To do so we start by writing how likely our dataset looks, if we are told the value of α:

$$L(\alpha) = f(x_1 \ldots x_n) = \prod_{i=1}^{n} \frac{\alpha}{x_i^{\alpha+1}}$$

Optimization will be much easier if we instead try to optimize the log likelihood:

$$\begin{aligned} LL(\alpha) = \log L(\alpha) &= \log \prod_{i=1}^{n} \frac{\alpha}{x_i^{\alpha+1}} \\ &= \sum_{i=1}^{n} \log \frac{\alpha}{x_i^{\alpha+1}} \\ &= \sum_{i=1}^{n} \log \alpha - (\alpha+1) \log x_i \\ &= n \log \alpha - (\alpha+1) \sum_{i=1}^{n} \log x_i \end{aligned}$$

Selecting α

We are going to select α to be the value which maximizes the log likelihood. To do so we are going to need the derivative of LL w.r.t. α

$$\begin{aligned} \frac{\partial LL(\alpha)}{\partial \alpha} &= \frac{\partial LL(\alpha)}{\partial \alpha} \left(n \log \alpha - (\alpha+1) \sum_{i=1}^{n} \log x_i \right) \\ &= \frac{n}{\alpha} - \sum_{i=1}^{n} \log x_i \end{aligned}$$

One way to optimize is to take the derivative and set it equal to zero:

$$\begin{aligned} 0 &= \frac{n}{\alpha} - \sum_{i=1}^{n} \log x_i \\ \sum_{i=1}^{n} \log x_i &= \frac{n}{\alpha} \\ \alpha &= \frac{n}{\sum\limits_{i=1}^{n} \log x_i} \end{aligned}$$

At this point we have a formula that we can use to calculate α! Wahoo

Putting it into code

```python
import math

def estimate_alpha(observations):
    # This code computes the MLE estimate of alpha
    log_sum = 0
    for x_i in observations:
        log_sum += math.log(x_i)
    n = len(observations)
    return n / log_sum

def main():
    observations = [1.677, 3.812, 1.463, 2.641, 1.256, 1.678, 1.157, 1.146,
    1.323, 1.029, 1.238, 1.018, 1.171, 1.123, 1.074, 1.652, 1.873, 1.314,
    1.309, 3.325, 1.045, 2.271, 1.305, 1.277, 1.114, 1.391, 3.728, 1.405,
    1.054, 2.789, 1.019, 1.218, 1.033, 1.362, 1.058, 2.037, 1.171, 1.457,
    1.518, 1.117, 1.153, 2.257, 1.022, 1.839, 1.706, 1.139, 1.501, 1.238,
    2.53 , 1.414, 1.064, 1.097, 1.261, 1.784, 1.196, 1.169, 2.101, 1.132,
    1.193, 1.239, 1.518, 2.764, 1.053, 1.267, 1.015, 1.789, 1.099, 1.25 ,
    1.253, 1.418, 1.494, 1.015, 1.459, 2.175, 2.044, 1.551, 4.095, 1.396,
    1.262, 1.351, 1.121, 1.196, 1.391, 1.305, 1.141, 1.157, 1.155, 1.103,
    1.048, 1.918, 1.889, 1.068, 1.811, 1.198, 1.361, 1.261, 4.093, 2.925,
    1.133, 1.573]
    alpha = estimate_alpha(observations)
    print(alpha)

if __name__ == '__main__':
    main()
```