We offer the alternative view that users of the SPP model should consider the process of normalizing the claims to be a transformation of the data rather than the application of a parameter. An analogous transformation occurs when we take (natural) logarithms. When we do that, we do not consider the base of the logarithm (e) to be a parameter. Similarly, we should not consider the lower bound to be a parameter.

To improve the clarity of this concept, we present Table 1 comparing the more traditional two parameter Pareto Type I and the SPP. We denote the raw claim amounts as observations of the random variable $\mathbf{C}$ and the normalized claim amounts as $\mathbf{Z}$[1].

Distribution	Pareto Type I	SPP
Random Variable	Observed Claim Amount $\mathbf{C}$	Normalized Claims Amount $\mathbf{Z}$
Transformation	Not Applicable	$\mathbf{Z} = g(\mathbf{C})$ $g(\mathbf{C}) = \mathbf{C}$/lower bound
Parameters	$k > 0$ (scale); $a > 0$ (shape)	$q > 0$ (shape)
Domain	$[k, \infty]$	$[1, \infty]$
Density	$a\frac{k^a}{C^{a+1}}$	$qz^{-(q+1)}$

Table 1: The Pareto Type I and the SPP

The support of both the Pareto Type I and the SPP distribution are claims in excess of a threshold. The support for the former is all claim amounts greater than k. The support of the latter is all claim amounts greater than the lower bound which results in the normalized claim amounts greater than 1.

We can now work with model forms in the space of $\mathbf{Z}$ and then use g^{-1} to transform back into the space of $\mathbf{C}$. We can also now present the density and distribution functions.

$$f(z) = qz^{-(q+1)} \tag{1.3.2}$$

$$F(z) = 1 - z^{-q} \tag{1.3.3}$$

In Appendix B.1, we provide the derivation of the distribution function (Equation (1.3.3)).

[1]Later in this paper, we advocate an approach that requires plotting data on an x, y coordinate system. We use $\mathbf{C}$ and $\mathbf{Z}$ to avoid confusion with that coordinate system.

2 When is it appropriate to use the SPP?

Philbrick introduced the SPP as a distribution to model excess claims. As such, the most common actuarial use of the SPP is in the modeling of claims in the tail of a distribution which is of interest when the tail is said to be "thick" or "heavy." Of course, the terms 'thick" or "heavy" have no formal definition.

The simplicity/elegance of the SPP has had the unintended consequence that the SPP is widely-used without an assessment to determine if and where the data follow a Pareto distribution. (We note that Philbrick did not include such an assessment.) We propose an "assessment approach" (as compared to Philbrick's "selection approach") in this paper. We begin by applying our proposed approach to normalized data. We then extend the concept to data that is not normalized. We recommend that latter approach for actuaries to use in fitting the Pareto model.

2.1 The Zipf Plot

Specifically, we note that Pareto-distributed data plot as a straight line on a **Zipf plot**[Cirillo 2013].

To construct a Zipf plot, we plot the (empirical) survival function on the y-axis and the data points on the (transformed) x-axis. Both axes are on a $\log^2$ scale.

y-values From Equation (1.3.3), we recognize that the survival function is $1-(1-z^{-q})=z^{-q}$ and the natural logarithm of the survival function is $-q\ln z$.

x-values We note that the x values of the Zipf plot are $\ln z$

We represent the linear relationship [3] of the y and x values as:

$$-q\log z \sim b_1 \log z \tag{2.1.4}$$

Simplifying Equation 2.1.4, we have the straightforward observation that the coefficient of $\log z$ on the right hand side of the relationship (i.e., b_1) represents an estimator for the negative of the Pareto parameter (i.e., $-q$).

2.2 Zipf Plot Example Normalized Pseudo Data

In this section, we demonstrate the use of the Zipf plot using *normalized* pseudo data. We refer to this data as *normalized* consistent with the *Philbrick* definition raw values have been divided by the selected lower bound.

[2] All reference to logarithms that I include throughout this paper are natural logarithms.

[3] This notation indicates that the left side of the $\sim$ is a function of the right side of the $\sim$ without specifying any possible other terms of the relationship.

To generate that pseudo data, we assume that each of n observed points is located at the midpoint of evenly-spaced probability intervals[4] . That is, the empirical distribution and survival functions for the ith **ordered** point, $(z_{(i)}, i \in [1, n])$ are:

$$F(z_i) = \frac{i - 0.5}{n} \tag{2.2.5}$$

$$S(z_i) = \frac{n - i + 0.5}{n}. \tag{2.2.6}$$

From Equation (1.3.3), we recognize that associated normalized data points, z have values $S(z_i)^{(-1/q)}$. Also, importantly, we use $z_{(1)}$ to represent the first observed value. We later discuss the significance of $z_{(0)}$.

Then pseudo data points on the Zipf plot are:

$$(x_i, y_i) = (\ln z_{(i)}, \ln(\frac{n - i + 0.5}{n}))$$

where we now use $z_{(i)}$ to indicate the i-th order statistic of the data sample.

Then, we can use linear modeling tools to facilitate calculation of slope of line through the data points using ordinary least squares.

For the normalized data, using the following logic, we understand that the constant in the relationship (that is, the y-intercept) is, by definition, 0:

- Since the x-axis represents values of the $\log(z)$, we denote the minimum z-value as $z_{(0)}$.
- $F(z_{(0)}) = 0; S(z_{(0)}) = 1$
- The y-value at $z_{(0)}$ is $\ln(S(z_{(0)})$ which is equal to $\ln 1 = 0$.

Similarly we understand that $z_{(0)} = 1$ results in an x-value $= \ln(z_{(0)}) = 0$. We now recognize that the line fit to the point on the Zipf-plot passes through the origin.

We present Zipf plots using 100 pseudo-data points at q values of 0.5, 1.0, 1.5 and 2.0 in Figure 1 and include the least squares fitted line and the associated regression coefficient. In Appendix C.1, we present the `R` code used to generate Figure 1 which includes a function that can be used to generate Zipf-plots (adapted from [Cirillo 2013]).

2.3 The Lower Bound

Suppose however that the data were *not* normalized by a *selected* lower bound (we will denote that lower bound as **B** here). Then, we multiply each of the

[4] We emphasize that this is pseudo data meant only to support the reader's replication of the example. We acknowledge that this is not the only means through which one could generate pseudo data. In practice, we assume that the reader would be applying the recipe to observed data.

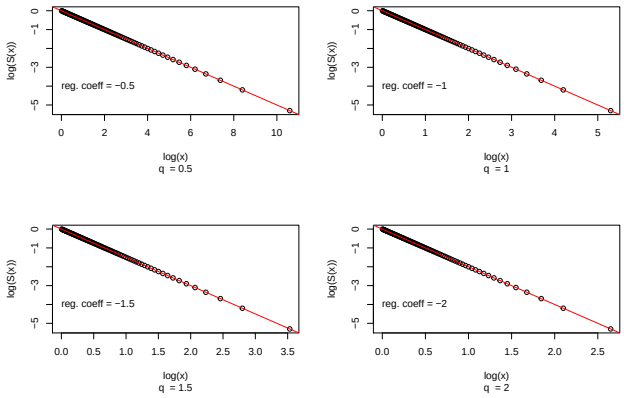

Figure 1: Zipf Plots

x-values in Figure 1 by $\mathbf{B}$, and the lns of the x-values would move rightward by $\ln B$. We can now see that we have returned to our original claim amounts $\mathbf{C}$. More importantly however, the constant in the linear relationship between $\ln \mathbf{c}$ and $\ln \mathbf{S}(\mathbf{z}) = \ln \mathbf{S}(\mathbf{c})$ is no longer 0 since the $c_{(0)}$ is now B rather than 0.

We can now use the linear relationship to solve for the x-value at which the y-value of the fitted line is equal to 0 (i.e., the x-intercept).

That is, we evaluate $\ln S(c) = \beta_0 + \beta_1 \log(c)$ at $c_{(0)}$.

$$\begin{aligned} \ln S(c_{(0)}) &= \beta_0 + \beta_1 \ln(x_{(0)}) \\ \ln 1 &= \beta_0 + \beta_1 \ln(c_{(0)}) \\ 0 &= \beta_0 + \beta_1 \ln(c_{(0)}) \\ c_{(0)} &= \exp(-\beta_0/\beta_1) \end{aligned}$$

Readers should recognize that we do not observe $c_{(0)}$; $c_{(1)}$ is our first observed value. The linear model provides statistical support for the threshold which in *Philbrick* was *selected*

2.4 Analysis Recipe

The elegance and stated purpose (modeling of excess layers) of the SPP invite its use without an evaluation of the appropriateness of the model. Key findings of this research paper are as follows:

Identify if and where data are Pareto-distributed Actuaries should use Zipf plots to understand first which data regions are indeed Pareto dis-

tributed. That is, the actuary should identify the range of data that exhibits a linear pattern on a Zipf plot. Statistical evaluation of linearity is outside the scope of this paper. However, we would suggest the following initial tests. (Note that below, we recommend a reevaluation of this assumption)

- a plot of the residuals of linear model as a function of fitted values
- a visual analysis

If the data do not appear to be linear on a Zipf plot, a Pareto model should not be selected.

If the data are (reasonably) linear in a certain region, the actuary, should then discard all data points outside the region of linearity.

Determine threshold and Pareto-parameter The actuary should then use the results of a linear model fit through the the remaining points (and only those points) of the Zipf-plot to parameterize that model. (This Zipf plot will differ from the initial plot as the empirical survival function will be calculated using only the retained data.)

- The negative of the covariate of $\ln c$ should be used as the estimator for the q parameter. (We explore advantages of this estimator to the maximum likelihood estimator in the next section.)
- the ratio of the negative of the intercept of the linear model and the covariate of $\ln c$ represents the value at which the data begin to be Pareto distributed.
- We can (and should) use tools (e.g., autocorrelation of residuals) that we use to statistically evaluate linear models to determine whether the data are indeed Pareto distributed (i.e., linear).
- The linear model output includes the standard error of the covariate (i.e., parameter uncertainty).

Application The actuary can then return to the presentation in Philbrick for formulæ relevant for modeling.

Actuarial judgment Actuaries should continue to apply judgment where appropriate throughout the process including but not limited to, assigning predictive value to underlying data points and in interpreting and using the modeling results.

Application of actuarial judgment is particularly important in addressing the practical issues that we discuss in Section 2.4.1.

2.4.1 Practical Issues

In executing this recipe in practice, there is one primary issue that we have encountered. That is, data are never perfectly Pareto-distributed as are the data in Figure 1. Specifically data will generally display some level of non-linearity. That is, data will typically display a concave down or concave up pattern.

Concave Downward Data that is concave downward will yield an indicated threshold that is greater than the initially selected threshold.

Concave Upward Data that is concave upward will produce an indicated threshold that is less that the initially selected threshold.

We have offer the following options for responding to this issue.

Re-select and Refit We could reexamine the data and identify a new value above which data are Pareto-distributed. We could then refit the linear model. This would be an iterative process.

Exclude individual data We could also exclude (inappropriately) influential points in fitting the model.

Accept the model In our experience, in many cases that data exhibit the patterns described, the indicated Pareto-parameter only changes slightly as we apply one of the two prior approached. So, we could retain the Pareto-parameter from the model but use judgment that considers both the model and the data in selecting the threshold value.

Actuaries should use professional judgment in the selection of which of these options to use.

3 The Pareto Parameter

Pseudo data plotted on Zipf plot also provides an important tool to help us understand the relationship between the data and the Pareto model. In this section, we use those tools to understand the following:

- Why the linear model produces a more robust estimator for the Pareto parameter.
- The effect of trend on the parameter and the lower bound.

3.1 Parameter Estimation

We now compare the approach that we present to Philbrick's approach to estimating the Pareto parameter.

- The maximum likelihood estimator (MLE) presented in *Philbrick*is:

$$q = \frac{n}{\sum \ln x}. \tag{3.1.7}$$

  We should recognize that the MLE is simply the reciprocal of the mean of $\ln x$.

- Option 2 is to use the coefficient of the linear model described in Section 2. For convenience, we will refer to this estimator as the CLM (coefficient of linear model).

We evaluate these alternatives considering the practical issues of missing data related to the estimation of the parameter. Specifically, we should understand that our observations may not be a representative sample of the claims that would be generated by a phenomenon that produces Pareto-distributed data.

To understand the effect on the parameter, we first consider situations where we only observed certain data points but that all possible data points are perfectly Pareto distributed. (That is, there is no process variance in the underlying data generation.) In Figure 2, we present an example where there are potentially 20 observed points (our population) and between 5 and 15 points are *not* observed in the samples (our samples). We generate all possible combinations under these conditions. In the upper panel, we plot the mean indicated parameter using MLE and CLM; in the lower panel, we plot the standard deviation of the indicated parameters.

We note that neither approach perfectly reproduces the underlying parameter. However, we did note that for many different values of the true parameter, the CLM resulted in an estimate closer to the actual value. We performed this analysis through simulation and present the underlying `R` code in Appendix C. Intuitively however we can recognize that a model (as is the basis for the CLM)

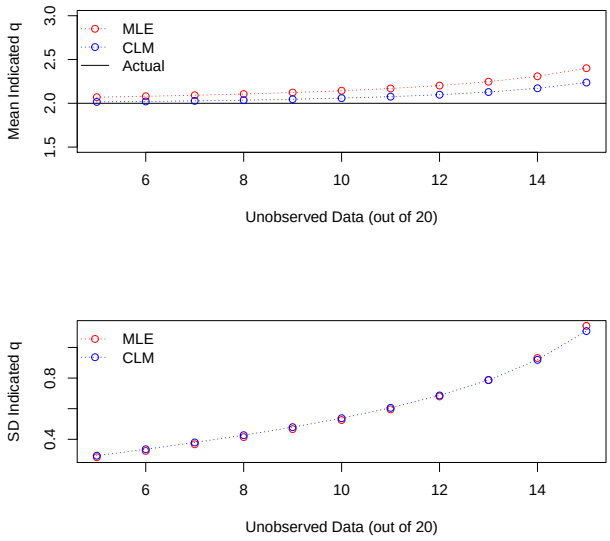

Figure 2: Comparison of Parameter Estimation Models

will help to extract signal from the data whereas an average (as this the basis for the MLE) effectively does not distinguish between noise and signal.

3.2 Trend

Philbrick espouses that the Pareto parameter should not be adjusted for claims inflation. That is, he argues that claims inflation results in frequency trend as more claim enter the "Pareto layer" but that there is no change to the SPP severity model.

With our recommended *assessment approach*, this is no longer intuitive or desirable. That is, Philbrick is implying that claims "become" Pareto-distributed once they trend to values above the threshold but that data is not Pareto-distributed below the threshold. In our approach, the use of the Zipf-plots maximizes the data used in our modeling. We identify the threshold above which all claims (which we presume have been appropriately adjusted to a common cost level) are Pareto-distributed. The Philbrick approach implies different levels of trend on either side of the threshold and that the trends acts in a way as to "push" claims over the threshold so as to preserve the Pareto parameter. While there is certainly a possibility that all these conditions are met, we view the simultaneous existence of all conditions as unlikely.

More specifically, we view "cost-leveling" as a separate modeling choice for the

actuary. That modeling is outside the scope of this paper. Using the linear model-based approach to determining the Pareto parameter *and* the threshold, and assuming that each claim is subject to the same rate of trend, each claim in the Pareto-distributed portion of the data would move right (inflationary trend) or left (deflationary trend) by an amount equal to the ln of the trend adjustment. The x-intercept (that is the threshold value) would similarly shift and the y-coordinates would not change. Similarly the covariate would not change.

4 Actuarial Application

4.1 Parameter Values

To provide context to the Pareto parameter, we first review the application of the SPP.

We can use Equation (4.1.8), to calculate the limited expected value through b as presented in Appendix B.3 as:[5]

$$\mathbb{E}[X;b] = \frac{q - b^{1-q}}{q-1} \tag{4.1.8}$$

In that derivation, there is no restriction that $q > 1$ as exists in the determination of the unlimited mean presented in Equation (B.2.14). That is, we recognize that, although the limited expected value is undefined, expected values are defined when we have an upper limit (such as a policy limit). Moreover, In Section IV., Philbrick indicates that:

> ... but most actual data suggests that the tail of the Pareto is still somewhat too thick at extremely high loss amounts. In other words, the theoretical density at high loss amounts is larger than empirical experience tends to indicate. Rather than discard the Pareto, it is easier to postulate that the distribution is censored or truncated at some high, but finite, value. As we have seen earlier, any upper limitation (either censorship point or truncation point) will produce formulæ for the mean claim size that are finite for all possible values of q.

As such, users of the SPP need not "fear" q values less than 1 for most insurance applications.

4.2 Claim Costs by Layer

Estimating claims for an excess policy is, of course, likely the most common use of the SPP. This was also a focus of Section III of Philbrick. For the expected claim amount for the layer between AP and L, we have:

[5] Philbrick used b to refer to both the "lower bound" and the policy limit. We will not do that in this paper primarily for clarity as using a variable to represent the lower bound implied at least the possibility that the lower bound was a parameter. Conveniently, it also allows us to use the traditional policy notation as attaching at AP through limit L with the resulting layer width equal to $L - AP$.

$$\begin{aligned}\mathbb{E}[X;AP,L] &= \frac{q-L^{1-q}}{q-1} - \frac{q-AP^{1-q}}{q-1} \\ &= \frac{AP^{1-q}-L^{1-q}}{q-1} \end{aligned} \tag{4.2.9}$$

4.3 Policy Claims Estimate

The purpose of the Philbrick calculation was likely to demonstrate that the average claim size in the layer between AP and L was equal to the expected value of claims limited to L/AP net of the lower bound but multiplied by AP. The latter is calculated as Equation (4.1.8) -1 which simplifies to:

$$\frac{1-b^{1-q}}{q-1} \times AP \tag{4.3.10}$$

We can demonstrate that using Equation (4.2.9) and the survival function as follows:

$$\begin{aligned}\frac{\frac{AP^{1-q}-L^{1-q}}{q-1}}{S(AP)} &= \frac{\frac{AP^{1-q}-L^{1-q}}{q-1}}{AP^{-q}} \\ &= \frac{1}{q-1} \times \frac{AP}{AP} \times \frac{AP^{1-q}-L^{1-q}}{AP^{-q}} \\ &= \frac{AP}{q-1} \times \frac{AP^{1-q}-L^{1-q}}{AP^{1-q}} \\ &= \frac{AP}{q-1} \times \left(1-\left(\frac{L}{AP}\right)^{1-q}\right) \\ &= \frac{1-(L/AP)^{1-q}}{q-1} \times AP \end{aligned} \tag{4.3.11}$$

As mentioned, the most common actuarial application of the SPP is to estimate the number of claims, their average value and the resulting aggregate claim amount to a policy. We summarize those formulæ for N ground-up claims in Table 2.

Number of Claims	$S(AP) = N \times AP^{-q}$
Average Value of Individual Claims	$\frac{1-(L/AP)^{1-q}}{q-1} \times AP$
Aggregate Claim Amount	$N \times \frac{AP^{1-q} - L^{1-q}}{q-1}$

Table 2: Policy Analysis

5 Concluding Remarks

Our goal with this paper was to provide additional guidance in deploying Philbrick's elegant solution to a complex problem. Our guidance supplements Philbrick with data visualization and model fitting that we expect would produce more robust solutions to the application of the Single Parameter Pareto in modeling excess claim layers.

Appendices

A Errata

In reviewing Philbrick, we noted two typographical errors and one calculation error. These are discussed below.

A.1 Philbrick Errata #1

In the application of formula (4.1.8), we should understand that there is a minor typographical error in Philbrick. The second paragraph following Equation (6) appears on Page 56 and includes the following:

$$\begin{aligned} b &= 20 \times (500,000/25,000) \\ &\text{which should be} \\ b &= 20 = 500,000/25,000 \end{aligned}$$

A.2 Philbrick Errata #2

Starting at the bottom of Page 58 and extending to Page 59, Philbrick presents an example with a q parameter of 1.5 and expected claim count of 7 that results in the following (where $S(x)$ represents the survival function):

$$\begin{aligned} F(4) &= 1 - 4^{-1.5} \\ F(4) &= 7/8 \\ S(4) &= 1 - F(4) = 1/8 \end{aligned}$$

$$\begin{aligned} \mathbb{E}[n] &= 7 \\ \mathbb{E}[n; x > 4] &= 7 \times S(4) = 7/8 \end{aligned} \tag{A.2.12}$$

(It is unfortunate that, in this example both $\mathbb{E}[n; x > 4]$ and $S(4)$ both equal 7/8.)

$$\begin{aligned} \mathbb{E}[X] &= \frac{1.5}{1.5 - 1} \\ \mathbb{E}[X] &= 3 \end{aligned}$$

$$\begin{aligned}
\mathbb{E}[X;4] &= \frac{1.5 - 4^{1-1.5}}{1.5 - 1} \\
\mathbb{E}[X;4] &= \frac{1.5 - 4^{-0.5}}{0.5} \\
\mathbb{E}[X;4] &= \frac{1.5 - .5}{0.5} \\
\mathbb{E}[X;4] &= 2
\end{aligned}$$

The average severity of claims in the layer is $(\mathbb{E}[X] - \mathbb{E}[X;4])/S(4) = 8$. Using the frequency calculated in Equation (A.2.12), we estimate claims in the layer to be $8 \times 7/8 = 7$ which agrees with Philbrick's calculation.

The error occurs when the example is extended to calculate claims in the layer from $AP = 3$ to $L = 7.5$. Using the approach above, we have the following:

$$\begin{aligned}
\mathbb{E}[X;3] &= 1.845299 \\
\mathbb{E}[X;7.5] &= 2.269703 \\
F(3) &= 0.8075499 \\
S(3) &= 0.1924501.
\end{aligned}$$

We have average claim amounts in the layer at

$$\frac{\mathbb{E}[X;7.5] - \mathbb{E}[X;3]}{1 - F(3)} = 2.205267$$

which agrees with Philbrick's calculation of "net average claim size" on Page 59. However, the corresponding frequency should be $7 \times S(3) = 1.347151$ and resulting expected claims in the layer of 2.970827. The purpose of the $F(87,500/75,000) = F(2.5)$ term in the frequency calculation is not entirely clear to this author.

A.3 Philbrick Errata #3

Equation (11) indicates that "nth moment of the Pareto distribution with no upper limit is" $\frac{q}{q+n}$. Then, in Equation (12) the second moment is represented in the calculation of variance by $\frac{q}{q-n}$ and of course we have the calculation of mean (first moment, $n = 1$) as $\frac{q}{q-1}$. We can see the error in Equation (11).

B Derivation of Forumulæ

B.1 SPP Cumulative Distribution Function

$$\begin{aligned}
F(x) &= \int_1^x f(x)\,\mathrm{d}x \\
&= \int_1^x qx^{-(q+1)}\,\mathrm{d}x \\
&= q\int_1^x x^{-(q+1)}\,\mathrm{d}x \\
&= q\frac{1}{-(q+1)+1}x^{-q}\Big|_1^x \\
&= q\frac{1}{-q}x^{-q}\Big|_1^x \\
&= -x^{-q}\Big|_1^x \\
&= -x^{-q} - (-1^{-q}) \\
F(x) &= 1 - x^{-q} \qquad \text{(B.1.13)}
\end{aligned}$$

B.2 Expected Values

$$\begin{aligned}
\mathbb{E}[X] &= \int_1^\infty xf(x)\,\mathrm{d}x \\
&= \int_1^\infty xqx^{-(q+1)}\,\mathrm{d}x \\
&= q\int_1^\infty x^{-q}\,\mathrm{d}x \\
&= q\frac{1}{-q+1}x^{-q+1}\Big|_1^\infty \\
&= \frac{q}{1-q}x^{-q+1}\Big|_1^\infty \\
\mathbb{E}[X] &= \frac{q}{1-q}\frac{1}{x^{q-1}}\Big|_1^\infty \qquad \text{(B.2.14)}
\end{aligned}$$

We can see that for $x = 1$ (the lower limit of integration) equation (B.2.14) evaluates to $\frac{q}{1-q}$. However for $x = \infty$ (the upper limit of integration), we have the following[6]:

[6] In the limit as $x \to \infty$, the expression evaluates to $-\frac{q}{q+1}$. However evaluated at ∞, the expression is undefined.

$$\frac{q}{1-q}\frac{1}{x^{q-1}} = \begin{cases} 0, & \text{if } q > 1 \\ \text{undefined}, & \text{if } q = 1 \\ \infty, & \text{if } q < 1 \end{cases}$$

and therefore we have:

$$\mathbb{E}[X] = \begin{cases} 0 - \dfrac{q}{1-q}, & \text{if } q > 1 \\ \text{undefined}, & \text{if } q = 1 \\ \infty, & \text{if } q < 1 \end{cases}$$

or more simply:

$$\mathbb{E}[X] = \begin{cases} \dfrac{q}{q-1}, & \text{if } q > 1 \\ \text{undefined}, & \text{if } q \leq 1 \end{cases} \tag{B.2.15}$$

B.3 SPP Limited Expected Value

The limited expected value is calculated as:

$$\begin{aligned}
\mathbb{E}[X;b] &= \int_1^b x f(x)\,\mathrm{d}x + b(1-F(b)) \\
&= \int_1^b xqx^{-(q+1)}\,\mathrm{d}x + b(1-F(b)) \\
&= q\int_1^b x^{-q}\,\mathrm{d}x + b(1-F(b)) \\
&= q\frac{1}{-q+1}x^{-q+1}\Big|_1^b + b(1-F(b)) \\
&= \frac{q}{1-q}\frac{1}{x^{q-1}}\Big|_1^b + b(1-F(b)) \\
&= \frac{q}{1-q}\left[\frac{1}{b^{q-1}} - \frac{1}{1^{q-1}}\right] + b\left[1-(1-b^{-q})\right] \\
&= \frac{q}{1-q}\left[\frac{1}{b^{q-1}} - 1\right] + b\left[b^{-q}\right] \\
&= \frac{q}{q-1}\left[1 - \frac{1}{b^{q-1}}\right] + b^{1-q} \\
&= \frac{q}{q-1}\left[1 - b^{1-q}\right] + b^{1-q} \\
&= \frac{1}{q-1}\left[q - qb^{1-q} + (q-1)b^{1-q}\right] \\
&= \frac{1}{q-1}\left[q - b^{1-q}\right] \\
&= \frac{q-b^{1-q}}{q-1}
\end{aligned} \tag{B.3.16}$$

B.4 Maximum Likelihood Estimator for Parameter

The negative log-likelihood (NLL) function given data $D = x_1 \dots x_n$ is defined as:

$$
\begin{aligned}
L(q) &= \prod_{i=1}^{n} f x_i \\
\text{NLL} &= -\sum_{i=1}^{n} \ln(f x_i) \\
\text{NLL} &= -\sum_{i=1}^{n} \ln(q x_i^{-(q+1)}) \\
\text{NLL} &= -\sum_{i=1}^{n} \left[\ln q + \ln x_i^{-(q+1)}\right] \\
\text{NLL} &= -\sum_{i=1}^{n} \left[\ln q - (q+1)\ln x_i\right] \\
\text{NLL} &= -n \ln q + \sum_{i=1}^{n} (q+1) \ln x_i \\
\text{NLL} &= -n \ln q + (q+1) \sum_{i=1}^{n} \ln x_i
\end{aligned}
$$

We can calculate the MLE of q by taking partial derivatives and setting equal to 0.

$$
\begin{aligned}
0 &= \frac{\partial}{\partial q} \left[-n \ln q + (q+1) \sum_{i=1}^{n} \ln x_i \right] \\
0 &= -n \frac{1}{q} + \sum_{i=1}^{n} \ln x \\
\sum_{i=1}^{n} \ln x &= \frac{n}{q} \\
q &= \frac{n}{\sum_{i=1}^{n} \ln x}
\end{aligned}
$$

C R Code

We present the R code used to generate Figure 1 and Figure 2 below.

C.1 R Code for Figure 1

```
zipfplot <- function(data) {
  data <- x_values
  data <- sort(as.numeric(data)) #sorting data
  y <- 1 - ppoints(data) # computing 1-F(x)

  plot(x = data, y = y, log = 'xy', xlab = 'x on log scale',
    ylab = '1-F(x) on log scale')
}

n_points <- 100

y_vals <- ( n_points - (n_points:1) + 2 / n_points ) /
  n_points

par(mfrow = c(2,2))

x_vals <- y_vals

lapply(X = c(0.5, 1, 1.5, 2), FUN = function(q){

  #q <- 2

  plot(x = log(x_vals ^ (-1/q)), y = log(y_vals),
    xlab = 'log(x)',
    ylab = 'log(S(x))', sub = paste0('q  = ', q))

  fit <- lm(log(y_vals) ~ log(x_vals ^ (-1/q)) + 0)

  abline(fit, col = 'red')

  text(x = 0, y = -4, labels = paste0('reg. coeff = ',
    round(fit$coefficient, 1)),
    adj = 0)
})
```

C.2 R Code for Figure 2

```
# This may take a reasonably long time to run!
missing_pts <- 5:15

n_points <- 20

scale <- 200000

q <- 2

y_vals <- (n_points - (n_points:0)) / n_points
y_vals <- (y_vals[1:(length(y_vals) - 1)] +
  y_vals[2:length(y_vals)]) / 2

x_values <- (1 - y_vals) ^ (-1 / q)
rm(y_vals)

mle <- lapply(X = missing_pts, FUN = function(missing) {
  no_sampled_points <- n_points - missing

  combn(x = x_values, m = no_sampled_points,
    FUN = function(sampled_points) {
      no_sampled_points / sum(log(sampled_points))
    }
  )
})


clm <- lapply(X = missing_pts,   FUN = function(missing){

  no_sampled_points <- n_points - missing

  combn(x = x_values, m = no_sampled_points,
    FUN = function(sampled_points) {
      sampled_points <- sampled_points[order(sampled_points)]

      y_vals <- (no_sampled_points - (no_sampled_points:0) ) /
        no_sampled_points
      y_vals <- (y_vals[1:(length(y_vals) - 1)] +
        y_vals[2:length(y_vals)]) / 2
      y_vals <- 1 - y_vals

      -lm(log(y_vals) ~  log(sampled_points) + 0)$coefficient
    }
```

```
  )
})


clm_mean <- sapply(clm, mean, simplify = TRUE)
mle_mean <- sapply(mle, mean, simplify = TRUE)
clm_sd <- sapply(clm, sd, simplify = TRUE)
mle_sd <- sapply(mle, sd, simplify = TRUE)
save(clm, mle, clm_mean, clm_sd, mle_mean, mle_sd,
  file = './param_test.RData')

par(mfrow = c(2, 1))
plot(x = missing_pts, y = sapply(mle, mean, simplify = TRUE),
  ylab = 'Mean Indicated q',
  xlab = 'Number of Unobserved Data Points (out of 20)',
  type = 'n', ylim = c(1.5, 3))
abline(h = 2, lty = 'solid')
points(x = missing_pts, y = mle_mean, col = 'red')
lines(x = missing_pts, y = mle_mean, col = 'red',
  lty = 'dotted')
points(x = missing_pts, y = clm_mean, col = 'blue')
lines(x = missing_pts, y = clm_mean, col = 'blue',
  lty = 'dotted')
legend('topleft', legend = c('MLE', 'CLM', 'Actual'),
  lty = c('dotted', 'dotted', 'solid'), pch = c(1, 1, NA),
  col = c('red','blue', 'black'), bty = 'n')

plot(x = missing_pts, y = sapply(mle, sd, simplify = TRUE),
  ylab = 'SD Indicated q',
  xlab = 'Number of Unobserved Data Points (out of 20)',
  type = 'n')
points(x = missing_pts, y = sapply(mle, sd, simplify = TRUE),
  col = 'red')
lines(x = missing_pts, y = sapply(mle, mean, simplify = TRUE),
  col = 'red', lty = 'dotted')
points(x = missing_pts, y = sapply(clm, sd, simplify = TRUE),
  col = 'blue')
lines(x = missing_pts, y = sapply(clm, sd, simplify = TRUE),
  col = 'blue', lty = 'dotted')
legend('topleft', legend = c('MLE', 'CLM'),
  lty = c('dotted', 'dotted'),
  pch = c(1, 1), col = c('red','blue'), bty = 'n')
```

References

[1] Pasquale Cirillo. "Are Your Data Really Pareto Distributed?" In: *Physica A: Statistical Mechanics and its Applications* 392 (23 2013), pp. 5947–5962. URL: https://arxiv.org/abs/1306.0100.

[2] Stephen W. Philbrick. "A Practical Guide to the Single Parameter Pareto Distribution". In: *Proceedings of the Casualty Actuarial Society* LXXII (1985), pp. 44–84.