1 Exercise 1: Generalized entropy and diversity

Strategy: Equate the corresponding information measure of text T to that of text T' and solve for the diversity, D. The probabilities will be given by $p_i = \frac{1}{D}$.

1.1 a) Simpson Concentration

$$S[T] = S[T'] \tag{1}$$

$$= \sum_{j=1}^{D} p_j^2 \tag{2}$$

$$= \sum_{j=1}^{D} (\frac{1}{D})^2 \tag{3}$$

$$= (D)(\frac{1}{D^2}) \tag{4}$$

$$S[T] = D^{-1} = \sum_{i=1}^{n} p_i^2 \tag{5}$$

$$\implies D = \frac{1}{\sum_{i=1}^{n} p_i^2} \tag{6}$$

1.2 b) Gini Index

$$G[T] = G[T'] \tag{7}$$

$$= 1 - \sum_{j=1}^{D} p_j^2 \tag{8}$$

$$G[T] = 1 - D^{-1} = 1 - \sum_{i=1}^{n} p_i^2 \tag{9}$$

$$\implies D = \frac{1}{\sum_{i=1}^{n} p_i^2} \tag{10}$$

The Simpson Concentration and Gini Concentration have the same diversity!

1.3 c) Shannon Entropy

$$
\begin{aligned}
H[T] &= H[T'] \qquad (11)\\
&= -\sum_{j=1}^{D} p_j \ln p_j \qquad (12)\\
&= -(D)(\frac{1}{D})\ln(\frac{1}{D}) \qquad (13)\\
&= -\ln(\frac{1}{D}) \qquad (14)\\
H[T] &= \ln D = -\sum_{i=1}^{n} p_i \ln p_i \qquad (15)\\
\implies D &= e^{-\sum p_i \ln p_i} \qquad (16)
\end{aligned}
$$

Note: In the line above, the lower and upper limits of the summation have been left out for esthetic reasons.

1.4 d) Renyi Entropi

$$
\begin{aligned}
H_q^{(R)}[T] &= H_q^{(R)}[T'] \qquad (17)\\
&= \frac{1}{q-1}[-\ln\sum_{j=1}^{D} p_j^q] \qquad (18)\\
&= \frac{1}{1-q}\ln[(D)(\frac{1}{D})^q] \qquad (19)\\
&= \frac{1}{1-q}\ln[D^{1-q}] \qquad (20)\\
H_q^{(R)} &= \ln D = \frac{1}{1-q}\ln\sum_{i=1}^{n} p_i^q \qquad (21)\\
\implies D &= [\sum_{i=1}^{n} p_i^q]^{\frac{1}{1-q}} \qquad (22)
\end{aligned}
$$

1.5 e) Generalized Tsallis Entropy

$$H_q^{(Ts)}[T] = H_q^{(Ts)}[T'] \tag{23}$$

$$= \frac{1}{q-1} - \frac{1}{q-1}[\ln \sum_{j=1}^{D} p_j^q] \tag{24}$$

$$H_q^{(Ts)}[T] = \frac{1}{q-1} + \ln D = \frac{1}{q-1} - \frac{1}{q-1}[\ln \sum_{i=1}^{D} p_i^q] \tag{25}$$

$$\implies \ln D = -\frac{1}{q-1}[\ln \sum_{i=1}^{D} p_i^q] \tag{26}$$

$$\implies D = [\sum_{i=1}^{n} p_i^q]^{\frac{1}{1-q}} \tag{27}$$

The Renyi Entropy and Generalized Tsallis Entropy have the same diversity!

1.6 f) The limiting case of $q \to 1$

Here it will be shown that as $q \to 1$, the diversity of the Renyi and Generalized Tsallis Entropy becomes the diversity of the Shannon Entropy. For simplicity (hopefully), the limit will be evaluated for $\ln D$ instead, then exponentiated at the end.

$$\lim_{q \to 1} \ln D = \lim_{q \to 1} \frac{\ln[\sum_{i=1}^{n} p_i^q]}{1-q} = \frac{0}{0} \tag{28}$$

Recall that the probabilities are normalized. In other words, $\sum_i = p_i = 1$ and $\ln 1 = 0$. This is the reason why the indeterminate form $\frac{0}{0}$ is obtained. Applying L'Hopital's Rule once will get rid of this issue:

$$\lim_{q \to 1} \ln D = -\lim_{q \to 1} \frac{d}{dq}[\sum_i p_i^q]; \text{ via L'Hopital} \tag{29}$$

$$\tag{30}$$

Now, there's three rules that will be convenient to recall:

1. Differential operators can be interchanged with summations
2. $\frac{d}{dx} \ln f(x) = \frac{f'(x)}{f(x)}$
3. $\frac{d}{dx} a^x = a^x \ln a$, where $a > 0$ and $a \neq 1$

Thus, applying these to the right hand side (RHS) of the above equation, the limit becomes:

$$\lim_{q\to 1} \ln D = -\lim_{q\to 1}[\frac{\sum_i p_i^q \ln p_i}{\sum_i p_i^q}] \tag{31}$$

Applying the normalization condition to the denominator in the RHS:

$$\lim_{q\to 1} \ln D = -\lim_{q\to 1}[\sum_i p_i^q \ln p_i] = -[\sum_i p_i \ln p_i] \tag{32}$$

$$\implies \lim_{q\to 1} D = e^{-\sum_i p_i \ln p_i} \tag{33}$$

$\therefore$ In the limit $q \to 1$, the Diversity of the Renyi and Generalized Tsallis Entropies becomes the Shannon Entropy Diversity!

2 Exercise 2: Mandelbrotian derivation of Zipf's Law

Want: The set of probabilities p_i that minimize Ψ. Thus, need to solve the following equation for p_i:

$$\frac{d\Psi}{dp_i} = \frac{dF}{dp_i} + \lambda\frac{dG}{dp_i} = 0 \tag{34}$$

NOTE: The derivatives are rather extensive to type in LaTeX in excruciating detail. As such, only the key ideas behind it will be discussed here. To see my full derivation, please follow this link: https://github.com/ecasiano/PrinciplesOfComplexSystems/blob/master/HW05/Q02/CSYS300_%20HW05.pdf

The function $F(\{p_i\})$ is the "Cost over Information" function and is defined as:

$$F(\{p_i\}) = \frac{C}{H} \tag{35}$$

where $C = \sum_{i=1}^{n} p_i \ln(i+a)$ and $H = -g\sum_{i=1}^{n} p_i \ln p_i$. The derivative of F respect to p_i becomes:

$$\frac{dF}{dp_i} = -\frac{1}{g}[\frac{\sum_i \ln(i+a)}{\sum_i p_i \ln p_i} - \frac{(\sum_i p_i \ln(i+a))(\sum_i(\ln p_i + 1))}{(\sum_i p_i \ln p_i)^2}] \tag{36}$$

Which, using the definition for the cost and information functions, can be rewritten as:

$$\frac{dF}{dp_i} = \frac{\sum_i \ln(i+a)}{H} + \frac{gC\sum_i(\ln p_i + 1)}{H^2} \tag{37}$$

The function $G(\{p_i\})$ is known as the "constraint equation" and, for this case, is defined as:

$$G(\{p_i\}) = -1 + \sum_i p_i = 0 \tag{38}$$

i.e, G is the normalization condition for the probabilities. Taking the derivative of G with respect to p_i, it is seen that:

$$\frac{dG}{dp_i} = \sum_i \tag{39}$$

Don't worry, the summand is not missing, it is just 1 :)

Substituting the derivatives of F and G into $\frac{d\Psi}{dp_i}$:

$$\frac{d\Psi}{dp_i} = \frac{\sum_i \ln(i+a)}{H} + \frac{gC\sum_i(\ln p_i + 1)}{H^2} + \lambda\sum_i = 0 \tag{40}$$

The above expression can be all written as a single sum, instead of four:

$$\sum_i [\frac{1}{H}\ln(i+a) + \frac{gC}{H^2}\ln p_i + \frac{gC}{H^2} + \lambda] = 0 \tag{41}$$

The summand must then be zero. Setting the summand to zero and solving for $\ln p_i$:

$$\ln p_i = -1 - \frac{\lambda H^2}{gC} + \ln(i+a)^{-H/gC} \tag{42}$$

Exponentiating on both sides of the equation:

$$p_i = e^{-1-\frac{\lambda H^2}{gC}}(i+a)^{-H/gC} \tag{43}$$

Thus, the probabilities follow a power-law scaling $p_i \propto (i+a)^{-H/gC}$. Nevertheless, an exact solution can be obtained for p_i by determining the value of the pre-factor. Substituting the expression for $\ln p_i$ into the definition of H (but not p_i), it can be seen that:

$$H = g + \frac{\lambda H^2}{C} + H \tag{44}$$

Solving for λ:

$$\lambda = -\frac{gC}{H^2} \tag{45}$$

Substituting this expression for λ into the exponent of the pre-factor of p_i, the terms in the exponent cancel out to zero and, thus, the pre-factor becomes unity. Therefore, the exact solution for p_i is:

$$p_i = (i+a)^{-H/gC} \tag{46}$$

$\therefore$ The probabilities follow an Inverse Power-Law Scaling of $p_i = (i+a)^{-\alpha}$, where $\alpha = \frac{H}{gC}$.

3 Exercise 3: Testing out the previous result

The code developed to solve this exercise can be found here: https://github.com/ecasiano/PrinciplesOfComplexSystems/blob/master/HW05/Q03/zipfarama.py

3.1 a)

In the previous problem, it was obtained that:

$$p_i = (i+a)^{-\alpha} \tag{47}$$

In the case that $a = 1$ and $n \to \infty$, an estimate of the scaling exponent α can be obtained computationally. First of all, recall, the normalization condition on the probabilities:

$$\sum_{i=1}^{n} (i+a)^{-\alpha} = 1 \tag{48}$$

Below, the algorithm developed is illustrated.

Algorithm:

```
1. Initialize random value of $\alpha$
2. Set desired tolerance
3. Calculate the error: $|\sum_i^n (i+a)^\alpha - 1|$
4. While error $>$ tolerance:
   -If $\sum_i^n (i+a)^\alpha < 1$, decrease $\alpha$
   -else, increase $\alpha$
   -Recalculate the error
```

Running the above routine for $n = 1,000,000$ words, the scaling exponent was estimated as $\alpha \approx 1.73$. The terminal output in Figure 1 shows all the iterations of the loop, with the corresponding value of α and the absolute error of the normalization.

3.2 b)

Want: Estimate the value of a that gives $\alpha = 1$ in terms of n by approximating the summation of the normalization condition as an integral.

The normalization condition can be approximated as an integral as:

$$\sum_{i=1}^{n} (i+a)^{-\alpha} = 1 \to \int_1^n (x+a)^{-\alpha} dx = 1 \tag{49}$$

Setting $\alpha = 1$:

$$\int_1^n \frac{1}{x+a} dx = 1 \tag{50}$$

Using the substitution: $u = x + a$ and $du = dx$, the integral becomes:

$$\int_{1+a}^{n+a} \frac{1}{u} du = 1 \tag{51}$$

Note that the limits of integration have been replaced by $u(x = 1)$ (lower limit) and $u(x = n)$ (upper limit). The definite integral becomes:

$$\int_{1+a}^{n+a} \frac{1}{u} du = \ln u|_{u=1+a}^{u=n+a} = \ln(\frac{n+a}{1+a}) = 1 \tag{52}$$

Exponentiating the last part of the line above:

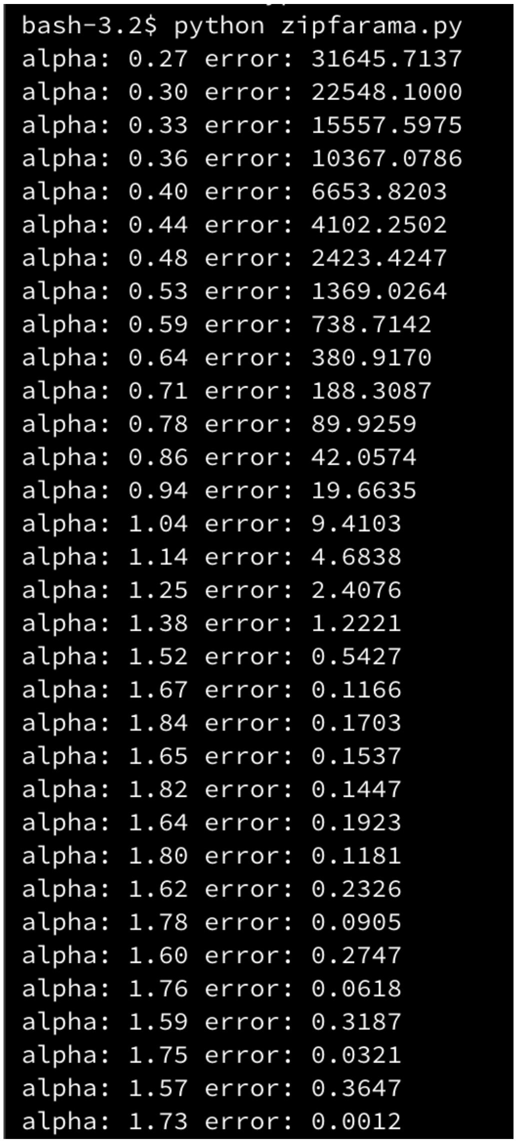
$$\frac{n+a}{1+a} = e \tag{53}$$

Solving for a:

$$a = \frac{e-n}{1-e} \tag{54}$$

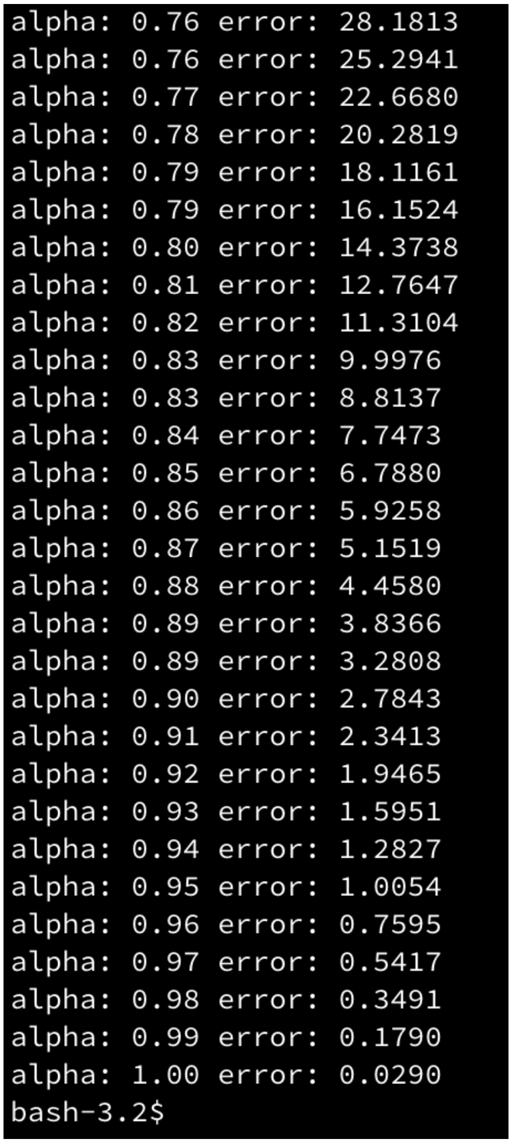
In the limit of $n \to \infty$, $a \to \infty \implies p_i \to 0$. In other words, the probabilities, which follow an inverse-power law will vanish. This makes sense, since it is expected that the probability associated with element i will become increasingly small the larger the sample becomes.

The referenced code confirms the above result. Just substitute the parameter a in the code by the derived expression and alpha will end up being $\alpha = 1$ after a few iterations. See Figure 2.



```
bash-3.2$ python zipfarama.py
alpha: 0.27 error: 31645.7137
alpha: 0.30 error: 22548.1000
alpha: 0.33 error: 15557.5975
alpha: 0.36 error: 10367.0786
alpha: 0.40 error: 6653.8203
alpha: 0.44 error: 4102.2502
alpha: 0.48 error: 2423.4247
alpha: 0.53 error: 1369.0264
alpha: 0.59 error: 738.7142
alpha: 0.64 error: 380.9170
alpha: 0.71 error: 188.3087
alpha: 0.78 error: 89.9259
alpha: 0.86 error: 42.0574
alpha: 0.94 error: 19.6635
alpha: 1.04 error: 9.4103
alpha: 1.14 error: 4.6838
alpha: 1.25 error: 2.4076
alpha: 1.38 error: 1.2221
alpha: 1.52 error: 0.5427
alpha: 1.67 error: 0.1166
alpha: 1.84 error: 0.1703
alpha: 1.65 error: 0.1537
alpha: 1.82 error: 0.1447
alpha: 1.64 error: 0.1923
alpha: 1.80 error: 0.1181
alpha: 1.62 error: 0.2326
alpha: 1.78 error: 0.0905
alpha: 1.60 error: 0.2747
alpha: 1.76 error: 0.0618
alpha: 1.59 error: 0.3187
alpha: 1.75 error: 0.0321
alpha: 1.57 error: 0.3647
alpha: 1.73 error: 0.0012
```

Figure 1: Terminal output for the algorithm described to estimate the scaling exponent α. For this run the number of words or elements in the summation has been set to $n = 1,000,000$ and $a = 1$.



```
alpha: 0.76 error: 28.1813
alpha: 0.76 error: 25.2941
alpha: 0.77 error: 22.6680
alpha: 0.78 error: 20.2819
alpha: 0.79 error: 18.1161
alpha: 0.79 error: 16.1524
alpha: 0.80 error: 14.3738
alpha: 0.81 error: 12.7647
alpha: 0.82 error: 11.3104
alpha: 0.83 error: 9.9976
alpha: 0.83 error: 8.8137
alpha: 0.84 error: 7.7473
alpha: 0.85 error: 6.7880
alpha: 0.86 error: 5.9258
alpha: 0.87 error: 5.1519
alpha: 0.88 error: 4.4580
alpha: 0.89 error: 3.8366
alpha: 0.89 error: 3.2808
alpha: 0.90 error: 2.7843
alpha: 0.91 error: 2.3413
alpha: 0.92 error: 1.9465
alpha: 0.93 error: 1.5951
alpha: 0.94 error: 1.2827
alpha: 0.95 error: 1.0054
alpha: 0.96 error: 0.7595
alpha: 0.97 error: 0.5417
alpha: 0.98 error: 0.3491
alpha: 0.99 error: 0.1790
alpha: 1.00 error: 0.0290
bash-3.2$
```

Figure 2: Terminal output for the algorithm described to estimate the scaling exponent α. For this run the number of words or elements in the summation has been set to $n = 1,000,000$ and $a = \frac{e-n}{1-e}$.

4 Exercise 4: Extrapolating a CCDF given it's fit

The code used to generate the plot and estimates in this exercise is found here: https://github.com/ecasiano/PrinciplesOfComplexSystems/blob/master/HW05/Q04/completeDist.py

4.1 a) and b)

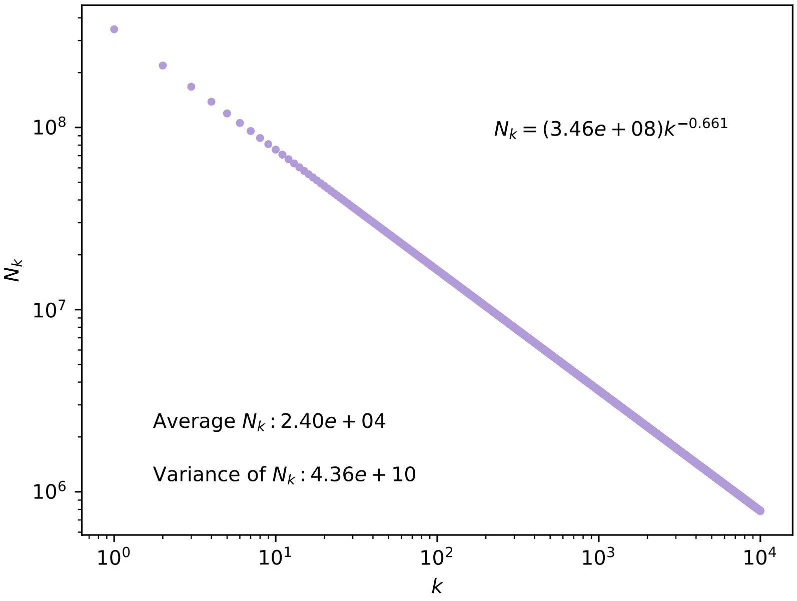


Figure 3: Reconstructed distribution of group sizes N_k as a function of k. N_k represents the number of distinct words that show up with frequency k. This distribution was generated by extrapolating the given CCDF, which spans from $k = 200$ to $k = 10^7$, to the range $k : [1, 199]$. NOTE: For ease of loading the image, only 5 orders of magnitude in k have been plotted. Nevertheless, the distribution has been calculated for the full of range of frequencies, $k \in [10^0, 10^7]$.

4.2 c) Fraction of words that show up only once

Define the total number of words, t, as:

$$t = \sum_{\text{All } k} kN_k \tag{55}$$

The fraction of words that show up only once out of all the words, $N_1^{(g)}$ can be obtained by:

$$N_1^{(g)} = \frac{N_1}{t} \tag{56}$$

where N_1 is the number of unique (and, in this case, also the total) words that show up only once.

The result was: $N_1^{(g)} \approx 5.67x10^{-10}$

4.3 d) Total unique words and fraction of unique words

The total number of unique words,t_u, can be obtained by summing all the group sizes, Nk:

$$t_u = \sum_{\text{All } k} N_k \tag{57}$$

The fraction of unique words out of all the total words is then:

$$N_u^{(g)} = \frac{t_u}{t} \tag{58}$$

The results obtained for the distribution were $t_u \approx 2.40x10^{11}$ for the total of unique words and $N_u^{(g)} \approx 3.94x10^{-7}$. This small fraction suggests that the body of text is mostly made up of words that show up at least twice.

4.4 e) What fraction of words are left out when only words with $200 \leq k \leq 10^7$ are considered?

The fraction of left out words, $N_{LeftOut}^{(g)}$, is given by:

$$N_{LeftOut}^{(g)} = \frac{\sum_{k=1}^{199} k * Nk}{t} \tag{59}$$

The fraction of left out words is: $5.12x10^{-7}$.

This extremely small value suggests that an insignificant fraction of the total words lied in the originally neglected interval of $k : [1, 199]$. On the other hand, the fraction of words that have $200 \leq k \leq 10^7$ is: 0.999999488. Thus, it is reasonable to neglect the interval not considered originally.