Word Frequency Distributions: The *zipfR* Package

Marco Baroni$^1$ and Stefan Evert$^2$

$^1$Center for Mind/Brain Sciences
University of Trento

$^2$Cognitive Science Institute
University of Onsabrück

Potsdam, 3-14 September 2007
Outline

Lexical statistics & word frequency distributions
  Basic notions of lexical statistics
  Typical frequency distribution patterns
  Zipf’s law
  Some applications

Statistical LNRE Models
  ZM & fZM
  Sampling from a LNRE model
  Great expectations
  Parameter estimation for LNRE models

zipfR
Lexical statistics

- Statistical study of the distribution of **types** (words or other linguistic units) in texts
  - remember the distinction between **types** and **tokens**?

- Different from other categorical data because of the extreme richness of types
  - people often speak of **Zipf’s law** in this context
Basic terminology

- **N**: sample / corpus size, number of **tokens** in the sample
- **V**: **vocabulary** size, number of distinct **types** in the sample
- **V_m**: **spectrum element** *m*, number of types in the sample with frequency *m* (i.e. exactly *m* occurrences)
- **V_1**: number of **hapax legomena**, types that occur only once in the sample (for hapaxes, #types = #tokens)

A sample: a b b c a a b a

N = 8, V = 3, V_1 = 1
Rank / frequency profile

- The sample: c a a b c c a c d
- Frequency list ordered by decreasing frequency

<table>
<thead>
<tr>
<th>t</th>
<th>f</th>
</tr>
</thead>
<tbody>
<tr>
<td>c</td>
<td>4</td>
</tr>
<tr>
<td>a</td>
<td>3</td>
</tr>
<tr>
<td>b</td>
<td>1</td>
</tr>
<tr>
<td>d</td>
<td>1</td>
</tr>
</tbody>
</table>
Rank / frequency profile

- The sample: \texttt{c a a b c c a c d}

- Frequency list ordered by decreasing frequency:

\[
\begin{array}{|c|c|}
\hline
 t & f \\
\hline
 c & 4 \\
 a & 3 \\
b & 1 \\
d & 1 \\
\hline
\end{array}
\]

- Rank / frequency profile: type labels instead of ranks:

\[
\begin{array}{|c|c|}
\hline
 r & f \\
\hline
 1 & 4 \\
 2 & 3 \\
 3 & 1 \\
 4 & 1 \\
\hline
\end{array}
\]

- Expresses type frequency as function of rank of a type
Rank/frequency profile of Brown corpus
### Top and bottom ranks in the Brown corpus

<table>
<thead>
<tr>
<th>Rank</th>
<th>Frequency</th>
<th>Word</th>
<th>Rank Range</th>
<th>Frequency</th>
<th>Randomly selected examples</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>62642</td>
<td>the</td>
<td>7967–8522</td>
<td>10</td>
<td>recordings, undergone, privileges</td>
</tr>
<tr>
<td>2</td>
<td>35971</td>
<td>of</td>
<td>8523–9236</td>
<td>9</td>
<td>Leonard, indulge, creativity</td>
</tr>
<tr>
<td>3</td>
<td>27831</td>
<td>and</td>
<td>9237–10042</td>
<td>8</td>
<td>unnatural, Lolotte, authenticity</td>
</tr>
<tr>
<td>4</td>
<td>25608</td>
<td>to</td>
<td>10043–11185</td>
<td>7</td>
<td>diffraction, Augusta, postpone</td>
</tr>
<tr>
<td>5</td>
<td>21883</td>
<td>a</td>
<td>11186–12510</td>
<td>6</td>
<td>uniformly, throttle, agglutinin</td>
</tr>
<tr>
<td>6</td>
<td>19474</td>
<td>in</td>
<td>12511–14369</td>
<td>5</td>
<td>Bud, Councilman, immoral</td>
</tr>
<tr>
<td>7</td>
<td>10292</td>
<td>that</td>
<td>14370–16938</td>
<td>4</td>
<td>verification, gleamed, groin</td>
</tr>
<tr>
<td>8</td>
<td>10026</td>
<td>is</td>
<td>16939–21076</td>
<td>3</td>
<td>Princes, nonspecifically, Arger</td>
</tr>
<tr>
<td>9</td>
<td>9887</td>
<td>was</td>
<td>21077–28701</td>
<td>2</td>
<td>blitz, pertinence, arson</td>
</tr>
<tr>
<td>10</td>
<td>8811</td>
<td>for</td>
<td>28702–53076</td>
<td>1</td>
<td>Salaries, Evensen, parentheses</td>
</tr>
</tbody>
</table>
Frequency spectrum

- The sample: c a a b c c a c d
- Frequency classes: 1 (b, d), 3 (a), 4 (c)
- Frequency spectrum:

<table>
<thead>
<tr>
<th>$m$</th>
<th>$V_m$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>3</td>
<td>1</td>
</tr>
<tr>
<td>4</td>
<td>1</td>
</tr>
</tbody>
</table>
Frequency spectrum of Brown corpus
Vocabulary growth curve

- The sample: abbcabcaba
Vocabulary growth curve

- The sample: a b b c a a b a
- \( N = 1, \ V = 1, \ V_1 = 1 \ (V_2 = 0, \ldots) \)
Vocabulary growth curve

- The sample: a b b c a a b a
- $N = 1, V = 1, V_1 = 1$ ($V_2 = 0, \ldots$)
- $N = 3, V = 2, V_1 = 1$ ($V_2 = 1, V_3 = 0, \ldots$)
Vocabulary growth curve

- The sample: a b b c a a b a
- $N = 1$, $V = 1$, $V_1 = 1$ ($V_2 = 0$, ...)
- $N = 3$, $V = 2$, $V_1 = 1$ ($V_2 = 1$, $V_3 = 0$, ...)
- $N = 5$, $V = 3$, $V_1 = 1$ ($V_2 = 2$, $V_3 = 0$, ...)
Vocabulary growth curve

- The sample: a b b c a a b a
- \( N = 1, \ V = 1, \ V_1 = 1 \) (\( V_2 = 0, \ldots \))
- \( N = 3, \ V = 2, \ V_1 = 1 \) (\( V_2 = 1, \ V_3 = 0, \ldots \))
- \( N = 5, \ V = 3, \ V_1 = 1 \) (\( V_2 = 2, \ V_3 = 0, \ldots \))
- \( N = 8, \ V = 3, \ V_1 = 1 \) (\( V_2 = 0, \ V_3 = 1, \ V_4 = 1, \ldots \))
Vocabulary growth curve of Brown corpus

With $V_1$ growth in red (curve smoothed with binomial interpolation)
Outline

Lexical statistics & word frequency distributions
   Basic notions of lexical statistics
   Typical frequency distribution patterns
   Zipf’s law
   Some applications

Statistical LNRE Models
   ZM & fZM
   Sampling from a LNRE model
   Great expectations
   Parameter estimation for LNRE models

zipfR
Typical frequency patterns
Across text types & languages
Typical frequency patterns

The Italian prefix *ri*- in the *la Repubblica* corpus
Is there a general law?

- Language after language, corpus after corpus, linguistic type after linguistic type, ... we observe the same “few giants, many dwarves” pattern

- Similarity of plots suggests that relation between rank and frequency could be captured by a general law
Is there a general law?

- Language after language, corpus after corpus, linguistic type after linguistic type, ... we observe the same “few giants, many dwarves” pattern
- Similarity of plots suggests that relation between rank and frequency could be captured by a general law
- Nature of this relation becomes clearer if we plot $\log f$ as a function of $\log r$
Outline

Lexical statistics & word frequency distributions
  Basic notions of lexical statistics
  Typical frequency distribution patterns
  Zipf’s law
  Some applications

Statistical LNRE Models
  ZM & fZM
  Sampling from a LNRE model
  Great expectations
  Parameter estimation for LNRE models

zipfR
Zipf’s law

- Straight line in double-logarithmic space corresponds to **power law** for original variables
- This leads to Zipf’s (1949, 1965) famous law:

\[ f(w) = \frac{C}{r(w)^a} \]

- With \( a = 1 \) and \( C = 60,000 \), Zipf’s law predicts that:
  - most frequent word occurs 60,000 times
  - second most frequent word occurs 30,000 times
  - third most frequent word occurs 20,000 times
  - and there is a long tail of 80,000 words with frequencies between 1.5 and 0.5 occurrences (!)
Zipf’s law

- Straight line in double-logarithmic space corresponds to **power law** for original variables
- This leads to Zipf’s (1949, 1965) famous law:

  \[ f(w) = \frac{C}{r(w)^a} \]

- With \( a = 1 \) and \( C = 60,000 \), Zipf’s law predicts that:
  - most frequent word occurs 60,000 times
  - second most frequent word occurs 30,000 times
  - third most frequent word occurs 20,000 times
  - and there is a long tail of 80,000 words with frequencies between 1.5 and 0.5 occurrences(!)
Zipf’s law
Logarithmic version

- Zipf’s power law:

\[ f(w) = \frac{C}{r(w)^a} \]

- If we take logarithm of both sides, we obtain:

\[ \log f(w) = \log C - a \log r(w) \]

- Zipf’s law predicts that rank / frequency profiles are straight lines in double logarithmic space

- Best fit \( a \) and \( C \) can be found with least-squares method
Zipf’s law
Logarithmic version

- Zipf’s power law:
  \[ f(w) = \frac{C}{r(w)^a} \]

- If we take logarithm of both sides, we obtain:
  \[ \log f(w) = \log C - a \log r(w) \]

- Zipf’s law predicts that rank / frequency profiles are straight lines in double logarithmic space
- Best fit \(a\) and \(C\) can be found with least-squares method
- Provides intuitive interpretation of \(a\) and \(C\):
  - \(a\) is **slope** determining how fast log frequency decreases
  - \(\log C\) is **intercept**, i.e., predicted log frequency of word with rank 1 (log rank 0) = most frequent word
Zipf’s law
Fitting the Brown rank/frequency profile
Mandelbrot’s extra parameter:

$$f(w) = \frac{C}{(r(w) + b)^a}$$

Zipf’s law is special case with $b = 0$

Assuming $a = 1$, $C = 60,000$, $b = 1$:

- For word with rank 1, Zipf’s law predicts frequency of 60,000; Mandelbrot’s variation predicts frequency of 30,000
- For word with rank 1,000, Zipf’s law predicts frequency of 60; Mandelbrot’s variation predicts frequency of 59.94

Zipf-Mandelbrot law forms basis of statistical LNRE models

ZM law derived mathematically as limiting distribution of vocabulary generated by a character-level Markov process
Zipf-Mandelbrot vs. Zipf’s law
Fitting the Brown rank/frequency profile
Lexical statistics & word frequency distributions
  Basic notions of lexical statistics
  Typical frequency distribution patterns
  Zipf’s law
  Some applications

Statistical LNRE Models
  ZM & fZM
  Sampling from a LNRE model
  Great expectations
  Parameter estimation for LNRE models

zipfR
Applications of word frequency distributions

► Most important application: **extrapolation** of vocabulary size and frequency spectrum to larger sample sizes
  ▶ productivity (in morphology, syntax, . . .)
  ▶ lexical richness
    (in stylometry, language acquisition, clinical linguistics, . . .)
  ▶ practical NLP (est. proportion of OOV words, typos, . . .)

need method for predicting vocab. growth on unseen data
Applications of word frequency distributions

- Most important application: **extrapolation** of vocabulary size and frequency spectrum to larger sample sizes
  - productivity (in morphology, syntax, . . . )
  - lexical richness
    - (in stylometry, language acquisition, clinical linguistics, . . . )
  - practical NLP (est. proportion of OOV words, typos, . . . )

Ë need method for predicting vocab. growth on unseen data

- Direct applications of Zipf’s law
  - population model for Good-Turing smoothing
  - realistic prior for Bayesian language modelling

Ë need model of type probability distribution in the population
Vocabulary growth: Pronouns vs. *ri*- in Italian

<table>
<thead>
<tr>
<th>N</th>
<th>V (pron.)</th>
<th>V (ri-)</th>
</tr>
</thead>
<tbody>
<tr>
<td>5000</td>
<td>67</td>
<td>224</td>
</tr>
<tr>
<td>10000</td>
<td>69</td>
<td>271</td>
</tr>
<tr>
<td>15000</td>
<td>69</td>
<td>288</td>
</tr>
<tr>
<td>20000</td>
<td>70</td>
<td>300</td>
</tr>
<tr>
<td>25000</td>
<td>70</td>
<td>322</td>
</tr>
<tr>
<td>30000</td>
<td>71</td>
<td>347</td>
</tr>
<tr>
<td>35000</td>
<td>71</td>
<td>364</td>
</tr>
<tr>
<td>40000</td>
<td>71</td>
<td>377</td>
</tr>
<tr>
<td>45000</td>
<td>71</td>
<td>386</td>
</tr>
<tr>
<td>50000</td>
<td>71</td>
<td>400</td>
</tr>
<tr>
<td>…</td>
<td>…</td>
<td>…</td>
</tr>
</tbody>
</table>
Vocabulary growth: Pronouns vs. *ri*- in Italian

Vocabulary growth curves

![Graph 1](image1)

![Graph 2](image2)
Outline

Lexical statistics & word frequency distributions
  Basic notions of lexical statistics
  Typical frequency distribution patterns
  Zipf’s law
  Some applications

Statistical LNRE Models
  ZM & fZM
  Sampling from a LNRE model
  Great expectations
  Parameter estimation for LNRE models

zipfR
LNRE models for word frequency distributions

- LNRE = large number of rare events (cf. Baayen 2001)
- Statistics: corpus = random sample from population
  - population characterised by vocabulary of types $w_k$ with occurrence probabilities $\pi_k$
  - not interested in specific types $\Rightarrow$ arrange by decreasing probability: $\pi_1 \geq \pi_2 \geq \pi_3 \geq \cdots$
  - NB: not necessarily identical to Zipf ranking in sample!
LNRE models for word frequency distributions

- LNRE = large number of rare events (cf. Baayen 2001)
- Statistics: corpus = random sample from population
  - population characterised by vocabulary of types $w_k$ with occurrence probabilities $\pi_k$
  - not interested in specific types ↔ arrange by decreasing probability: $\pi_1 \geq \pi_2 \geq \pi_3 \geq \cdots$
  - NB: not necessarily identical to Zipf ranking in sample!

- LNRE model = population model for type probabilities, i.e. a function $k \mapsto \pi_k$ (with small number of parameters)
  - type probabilities $\pi_k$ cannot be estimated reliably from a corpus, but parameters of LNRE model can
Examples of population models

\[ \pi_k \]

\[ k \]

\[ \pi \]

\[ k \]
The Zipf-Mandelbrot law as a population model

What is the right family of models for lexical frequency distributions?

- We have already seen that the Zipf-Mandelbrot law captures the distribution of observed frequencies very well.
The Zipf-Mandelbrot law as a population model

What is the right family of models for lexical frequency distributions?

- We have already seen that the Zipf-Mandelbrot law captures the distribution of observed frequencies very well
- Re-phrase the law for type probabilities:

\[ \pi_k := \frac{C}{(k + b)^a} \]

- Two free parameters: \( a > 1 \) and \( b \geq 0 \)
- \( C \) is not a parameter but a normalization constant, needed to ensure that \( \sum_k \pi_k = 1 \)
- this is the **Zipf-Mandelbrot** population model
Outline

Lexical statistics & word frequency distributions
  Basic notions of lexical statistics
  Typical frequency distribution patterns
  Zipf’s law
  Some applications

Statistical LNRE Models
  ZM & fZM
  Sampling from a LNRE model
  Great expectations
  Parameter estimation for LNRE models

zipfR
The parameters of the Zipf-Mandelbrot model

- $a = 1.2$, $b = 1.5$
- $a = 2$, $b = 10$
- $a = 2$, $b = 15$
- $a = 5$, $b = 40$
The parameters of the Zipf-Mandelbrot model

- $a = 1.2$, $b = 1.5$
- $a = 2$, $b = 10$
- $a = 5$, $b = 40$
The finite Zipf-Mandelbrot model

- Zipf-Mandelbrot population model characterizes an *infinite* type population: there is no upper bound on $k$, and the type probabilities $\pi_k$ can become arbitrarily small

- $\pi = 10^{-6}$ (once every million words), $\pi = 10^{-9}$ (once every billion words), $\pi = 10^{-12}$ (once on the entire Internet), $\pi = 10^{-100}$ (once in the universe?)
The finite Zipf-Mandelbrot model

- Zipf-Mandelbrot population model characterizes an *infinite* type population: there is no upper bound on \( k \), and the type probabilities \( \pi_k \) can become arbitrarily small
  - \( \pi = 10^{-6} \) (once every million words), \( \pi = 10^{-9} \) (once every billion words), \( \pi = 10^{-12} \) (once on the entire Internet), \( \pi = 10^{-100} \) (once in the universe?)
- Alternative: finite (but often very large) number of types in the population
- We call this the *population vocabulary size* \( S \) (and write \( S = \infty \) for an infinite type population)
The finite Zipf-Mandelbrot model

- The **finite Zipf-Mandelbrot** model simply stops after the first $S$ types ($w_1, \ldots, w_S$)
- $S$ becomes a new parameter of the model
  - → the finite Zipf-Mandelbrot model has 3 parameters

Abbreviations:
- **ZM** for Zipf-Mandelbrot model
- **fZM** for finite Zipf-Mandelbrot model
Outline

Lexical statistics & word frequency distributions
  Basic notions of lexical statistics
  Typical frequency distribution patterns
  Zipf’s law
  Some applications

Statistical LNRE Models
  ZM & fZM
  Sampling from a LNRE model
  Great expectations
  Parameter estimation for LNRE models

zipfR
Sampling from a population model

Assume we believe that the population we are interested in can be described by a Zipf-Mandelbrot model:

Use computer simulation to sample from this model:

- Draw $N$ tokens from the population such that in each step, type $w_k$ has probability $\pi_k$ to be picked
- This allows us to make predictions for samples (= corpora) of arbitrary size $N \Leftrightarrow$ extrapolation
Sampling from a population model

#1:  1  42  34  23  108  18  48  18  1  ...
Sampling from a population model

#1: 1 42 34 23 108 18 48 18 1 ...
    time order room school town course area course time ...

#2: ...
#3: ...
#4: ...
#5: ...
#6: ...
#7: ...
#8: ...
...

Sampling from a population model

#1:  1  42  34  23  108  18  48  18  1  ...
    time order room school town course area course time ...

#2:  286  28  23  36  3  4  7  4  8  ...

...
Sampling from a population model

#1: 1 42 34 23 108 18 48 18 1 ...
time order room school town course area course time ...

#2: 286 28 23 36 3 4 7 4 8 ...

#3: 2 11 105 21 11 17 17 1 16 ...

...
# Sampling from a population model

<table>
<thead>
<tr>
<th>#1:</th>
<th>1</th>
<th>42</th>
<th>34</th>
<th>23</th>
<th>108</th>
<th>18</th>
<th>48</th>
<th>18</th>
<th>1</th>
<th>...</th>
</tr>
</thead>
<tbody>
<tr>
<td>time</td>
<td>order</td>
<td>room</td>
<td>school</td>
<td>town</td>
<td>course</td>
<td>area</td>
<td>course</td>
<td>time</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

| #2: | 286 | 28  | 23  | 36  | 3   | 4   | 7   | 4   | 8   | ... |

| #3: | 2   | 11  | 105 | 21  | 11  | 17  | 17  | 1   | 16  | ... |

| #4: | 44  | 3   | 110 | 34  | 223 | 2   | 25  | 20  | 28  | ... |

| #5: | 24  | 81  | 54  | 11  | 8   | 61  | 1   | 31  | 35  | ... |

| #6: | 3   | 65  | 9   | 165 | 5   | 42  | 16  | 20  | 7   | ... |

| #7: | 10  | 21  | 11  | 60  | 164 | 54  | 18  | 16  | 203 | ... |

| #8: | 11  | 7   | 147 | 5   | 24  | 19  | 15  | 85  | 37  | ... |

...
## Samples: type frequency list & spectrum

<table>
<thead>
<tr>
<th>rank $r$</th>
<th>$f_r$</th>
<th>type $k$</th>
<th>$m$</th>
<th>$V_m$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>37</td>
<td>6</td>
<td>1</td>
<td>83</td>
</tr>
<tr>
<td>2</td>
<td>36</td>
<td>1</td>
<td>2</td>
<td>22</td>
</tr>
<tr>
<td>3</td>
<td>33</td>
<td>3</td>
<td>3</td>
<td>20</td>
</tr>
<tr>
<td>4</td>
<td>31</td>
<td>7</td>
<td>4</td>
<td>12</td>
</tr>
<tr>
<td>5</td>
<td>31</td>
<td>10</td>
<td>5</td>
<td>10</td>
</tr>
<tr>
<td>6</td>
<td>30</td>
<td>5</td>
<td>6</td>
<td>5</td>
</tr>
<tr>
<td>7</td>
<td>28</td>
<td>12</td>
<td>7</td>
<td>5</td>
</tr>
<tr>
<td>8</td>
<td>27</td>
<td>2</td>
<td>8</td>
<td>3</td>
</tr>
<tr>
<td>9</td>
<td>24</td>
<td>4</td>
<td>9</td>
<td>3</td>
</tr>
<tr>
<td>10</td>
<td>24</td>
<td>16</td>
<td>10</td>
<td>3</td>
</tr>
<tr>
<td>11</td>
<td>23</td>
<td>8</td>
<td></td>
<td></td>
</tr>
<tr>
<td>12</td>
<td>22</td>
<td>14</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**sample #1**
Samples: type frequency list & spectrum

<table>
<thead>
<tr>
<th>rank $r$</th>
<th>$f_r$</th>
<th>type $k$</th>
<th>$m$</th>
<th>$V_m$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>39</td>
<td>2</td>
<td>1</td>
<td>76</td>
</tr>
<tr>
<td>2</td>
<td>34</td>
<td>3</td>
<td>2</td>
<td>27</td>
</tr>
<tr>
<td>3</td>
<td>30</td>
<td>5</td>
<td>3</td>
<td>17</td>
</tr>
<tr>
<td>4</td>
<td>29</td>
<td>10</td>
<td>4</td>
<td>10</td>
</tr>
<tr>
<td>5</td>
<td>28</td>
<td>8</td>
<td>5</td>
<td>6</td>
</tr>
<tr>
<td>6</td>
<td>26</td>
<td>1</td>
<td>6</td>
<td>5</td>
</tr>
<tr>
<td>7</td>
<td>25</td>
<td>13</td>
<td>7</td>
<td>7</td>
</tr>
<tr>
<td>8</td>
<td>24</td>
<td>7</td>
<td>8</td>
<td>3</td>
</tr>
<tr>
<td>9</td>
<td>23</td>
<td>6</td>
<td>10</td>
<td>4</td>
</tr>
<tr>
<td>10</td>
<td>23</td>
<td>11</td>
<td>11</td>
<td>2</td>
</tr>
<tr>
<td>11</td>
<td>20</td>
<td>4</td>
<td>:</td>
<td>:</td>
</tr>
<tr>
<td>12</td>
<td>19</td>
<td>17</td>
<td>:</td>
<td>:</td>
</tr>
</tbody>
</table>

sample #2
Random variation in type-frequency lists

Sample #1

Sample #2

\( r \leftrightarrow f_r \)

\( k \leftrightarrow f_k \)
Random variation: frequency spectrum
Random variation: vocabulary growth curve

Sample #1

Sample #2

Sample #3

Sample #4
Outline

Lexical statistics & word frequency distributions
   Basic notions of lexical statistics
   Typical frequency distribution patterns
   Zipf’s law
   Some applications

Statistical LNRE Models
   ZM & fZM
   Sampling from a LNRE model
   Great expectations
   Parameter estimation for LNRE models

zipfR
Expected values

- There is no reason why we should choose a particular sample to make a prediction for the real data – each one is equally likely or unlikely.

- Take the average over a large number of samples, called **expected value** or **expectation** in statistics.

- Notation: $E[V(N)]$ and $E[V_m(N)]$
  - Indicates that we are referring to expected values for a sample of size $N$
  - Rather than to the specific values $V$ and $V_m$ observed in a particular sample or a real-world data set.

- Expected values can be calculated efficiently **without** generating thousands of random samples.
The expected frequency spectrum

Sample #1

Sample #2

Sample #3

Sample #4
The expected vocabulary growth curve
Confidence intervals for the expected VGC

Sample #1

$E[V(N)]$

$V(N)$  $E[V(N)]$

Sample #1

$E[V_1(N)]$

$V_1(N)$  $E[V_1(N)]$
Outline

Lexical statistics & word frequency distributions
  Basic notions of lexical statistics
  Typical frequency distribution patterns
  Zipf’s law
  Some applications

Statistical LNRE Models
  ZM & fZM
  Sampling from a LNRE model
  Great expectations
  Parameter estimation for LNRE models

zipfR
Parameter estimation by trial & error

\[ a = 1.5, \quad b = 7.5 \]

\[ V_{m/E_m} \]

\[ V(N)/E(V(N)) \]

- **Observed**
- **ZM model**
Parameter estimation by trial & error

$ \text{observed ZM model} \ a = 1.3, \ b = 7.5$

$V_m/E[V_m]$

$m$

$V(N)/E[V(N)]$

$N$

$\text{observed}$

$\text{ZM model}$

$a = 1.3, \ b = 7.5$
Parameter estimation by trial & error

- Observed ZM model parameters: $a = 1.3$, $b = 0.2$

Graphs showing:
- $V_m / E[V_m]$ vs. $m$
- $V(N)/E[V(N)]$ vs. $N$

- Observed data represented by black bars.
- ZM model represented by red line.

Legend:
- Black: observed
- Red: ZM model
Parameter estimation by trial & error

**Observed ZM model**

\[ a = 1.5, \quad b = 7.5 \]

**Graphs**

- **Graph 1:**
  - **Y-axis:** \( V_{m(E)} \)
  - **X-axis:** \( m \)
  - **Legend:**
    - Observed
    - ZM model

- **Graph 2:**
  - **Y-axis:** \( \frac{V(N)}{E[V(N)]} \)
  - **X-axis:** \( N \)
  - **Legend:**
    - Observed
    - ZM model
Parameter estimation by trial & error

\[ a = 1.7, \quad b = 7.5 \]

- Observed data
- ZM model

Graphs showing the comparison between observed data and the ZM model with parameters \( a = 1.7 \) and \( b = 7.5 \).
Parameter estimation by trial & error

\[ \text{observed \ ZM model} \]
\[ a = 1.7, \ b = 80 \]

\[ \text{observed \ ZM model} \]
\[ a = 1.7, \ b = 80 \]

\[ \begin{array}{c}
0 \quad 5000 \quad 10000 \quad 15000 \quad 20000 \quad 25000 \\
0 \quad 2 \times 10^5 \quad 4 \times 10^5 \quad 6 \times 10^5 \quad 8 \times 10^5 \quad 1 \times 10^6
\end{array} \]
Parameter estimation by trial & error

\[ a = 2, \ b = 550 \]

![Graphs showing data points and model predictions](image-url)
Automatic parameter estimation
Minimisation of suitable cost function for frequency spectrum

- By trial & error we found \( a = 2.0 \) and \( b = 550 \)
- Automatic estimation procedure: \( a = 2.39 \) and \( b = 1968 \)
- Goodness-of-fit: \( p \approx 0 \) (multivariate chi-squared test)
Summary

LNRE modelling in a nutshell:

1. Compile observed frequency spectrum (and vocabulary growth curves) for a given corpus or data set.
2. Estimate parameters of LNRE model by matching observed and expected frequency spectrum.
3. Evaluate goodness-of-fit on spectrum (Baayen 2001) or by testing extrapolation accuracy (Baroni & Evert 2007).
   ▶ In principle, you should only go on if model gives a plausible explanation of the observed data!
4. Use LNRE model to compute expected frequency spectrum for arbitrary sample sizes, or use population model directly as Bayesian prior etc.
Summary

LNRE modelling in a nutshell:

1. compile **observed** frequency spectrum (and vocabulary growth curves) for a given corpus or data set
Summary

LNRE modelling in a nutshell:

1. compile **observed** frequency spectrum (and vocabulary growth curves) for a given corpus or data set
2. estimate parameters of **LNRE model** by matching observed and expected frequency spectrum
LNRE modelling in a nutshell:
1. compile **observed** frequency spectrum (and vocabulary growth curves) for a given corpus or data set
2. estimate parameters of **LNRE model** by matching observed and expected frequency spectrum
3. evaluate **goodness-of-fit** on spectrum (Baayen 2001) or by testing extrapolation accuracy (Baroni & Evert 2007)
   ▶ in principle, you should only go on if model gives a plausible explanation of the observed data!
Summary

LNRE modelling in a nutshell:

1. compile **observed** frequency spectrum (and vocabulary growth curves) for a given corpus or data set
2. estimate parameters of **LNRE model** by matching observed and expected frequency spectrum
3. evaluate **goodness-of-fit** on spectrum (Baayen 2001) or by testing extrapolation accuracy (Baroni & Evert 2007)
   ▶ in principle, you should only go on if model gives a plausible explanation of the observed data!
4. use LNRE model to compute **expected** frequency spectrum for arbitrary sample sizes
   ▶ **extrapolation** of vocabulary growth curve
   ▶ or use population model directly as Bayesian prior etc.
Outline

Lexical statistics & word frequency distributions
Basic notions of lexical statistics
Typical frequency distribution patterns
Zipf’s law
Some applications

Statistical LNRE Models
ZM & fZM
Sampling from a LNRE model
Great expectations
Parameter estimation for LNRE models

zipfR
http://purl.org/stefan.evert/zipfR

- Already installed on the Potsdam machines
- Explore your GUI for general package installation and managing options
library(zipfR)

?zipfR

data(package="zipfR")
Importing data

data(itaRi.spc)
data(itaRi.emp.vgc)

my.spc <- read.spc("my.spc.txt")
my.vgc <- read.vgc("my.vgc.txt")

my.tfl <- read.tfl("my.tfl.txt")
my.spc <- tfl2spc(my.tfl)
Looking at spectra

summary(ItaRi.spc)
ItaRi.spc

N(ItaRi.spc)
V(ItaRi.spc)
Vm(ItaRi.spc,1)
Vm(ItaRi.spc,1:5)

# Baayen's P
Vm(ItaRi.spc,1) / N(ItaRi.spc)

plot(ItaRi.spc)
plot(ItaRi.spc, log="x")
Looking at vgcs

summary(ItaRi.emp.vgc)
ItaRi.emp.vgc

N(ItaRi.emp.vgc)

plot(ItaRi.emp.vgc, add.m=1)
Creating vgcs with binomial interpolation

# interpolated vgc

ItaRi.bin.vgc <- vgc.interp(ItaRi.spc, N(ItaRi.emp.vgc), m.max=1)

summary(ItaRi.bin.vgc)

# comparison

plot(ItaRi.emp.vgc, ItaRi.bin.vgc, legend=c("observed","interpolated"))
Load the spectrum and empirical vgc of the rarer prefix *ultra-*
- Compute binomially interpolated vgc for *ultra-*
- Plot the binomially interpolated *ri-* and *ultra-* vgcs together
Estimating LNRE models

# fZM model; you can also try ZM and GIGP, and compare

ItaUltra.fzm <- lnre("fzm", ItaUltra.spc)

summary(ItaUltra.fzm)
# expected spectrum

ItaUltra.fzm.spc <- lnre.spc(ItaUltra.fzm, N(ItaUltra.fzm))

# compare

plot(ItaUltra.spc, ItaUltra.fzm.spc, legend=c("observed","fzm"))

# plot first 10 elements only

plot(ItaUltra.spc, ItaUltra.fzm.spc, legend=c("observed","fzm"), m.max=10)
Compare growth of two categories

# extrapolation of ultra- V to ri- sample size

ItaUltra.ext.vgc <- lnre.vgc(ItaUltra.fzm, N(ItaRi.emp.vgc))

# compare

plot(ItaUltra.ext.vgc, ItaRi.bin.vgc, N0=N(ItaUltra.fzm), legend=c("ultra-","ri-"))

# zooming in

plot(ItaUltra.ext.vgc, ItaRi.bin.vgc, N0=N(ItaUltra.fzm), legend=c("ultra-","ri-"), xlim=c(0,1e+5))