

The program takes one input file and produces three text files—a sequence file, a test file, and an error file. The three output files are placed in the same folder as the input file.

### *Input File*

An input file for simple probabilities or simple frequencies generation consists of a header specifying the number of distinct events, followed by a listing of the probabilities or frequencies for each event (for details, see Table 3 below). An input file for contextual probabilities, contextual frequencies—exact, or contextual frequencies—block generation consists of a header specifying the number of contexts, the context size, and the number of distinct events, followed by a listing of the contexts and the probabilities or frequencies (for details, see Table 4 below).

Note that an incorrect header value will cause the program to generate an error message. For example, specifying 7 as the number of contexts when only six contexts are listed in the input file may cause the program to generate the message "ERROR – Encountered the end of file while attempting to read a nonnegative integer". Thus some error messages could be the result of an incorrect header value.

### *Sequence File*

The sequence file contains the generated sequences. The sequence file's name is the input file's name plus the suffix `_###seq001`, where `###` is `sp`, `sf`, `cp`, `cfe`, and `cfb` for simple probabilities, simple frequencies, contextual probabilities, contextual frequencies—exact, and contextual frequencies—block generation, respectively. If a file with that name already exists, then the suffix number is incremented by 1 to `_###seq002`, and so on until a nonexistent file name is found. If the suffix `_###seq999` is reached and a file with that name already exists, the file is overwritten.

### *Test File*

The test file's main purpose is to reassure the user that the program's implementation of the selected generation method using the selected input file was as expected. The program randomly chooses a sequence from the sequence file, calculates various quantities using the selected sequence, and reports the quantities in a test file. For contextual probabilities and contextual frequencies—block generation, the program reports (1) the number of occurrences of each context in the selected sequence, (2) the observed conditional relative frequencies (CRFs), and (3) the minimum, maximum, and mean absolute discrepancies between the observed CRFs and the nonzero expected CRFs from the input file. The discrepancies can range from 0 to 1. The mean discrepancy should, generally, be smaller for longer sequences than shorter sequences, and for contextual frequencies—block generation than contextual probabilities generation. The program also reports, for contextual frequencies—block generation, whether or not the observed block frequencies match the expected block frequencies from the input file. For example, with respect to the first input file in Table 4, the program would determine whether or not every 16 occurrences of context 3 was followed by events 1—4 8, 4, 0, and 4 times, respectively.

For contextual frequencies—exact generation, the program reports (1) the number of occurrences of each context in the selected sequence, (2) the observed frequencies with which each event follows each context, and (3) whether or not the observed frequencies match the

expected frequencies from the input file. For simple probabilities generation, the program reports (1) the length of the selected sequence, (2) the observed relative frequencies (RFs), and (3) the minimum, maximum, and mean absolute discrepancies between the observed RFs and the nonzero expected RFs from the input file. Finally, for simple frequencies generation, the program reports (1) the length of the selected sequence, (2) the observed frequencies of each event, and (3) whether or not the observed frequencies match the expected frequencies from the input file.

The test file's name is the input file's name plus the suffix `_##test###`, where # are characters identical to those in the sequence file's name. This makes it easy to determine which test file belongs with which sequence file when there are multiple sequence and test files.

### *Error File*

The program copies every character it reads from the input file to an error file. If there is a problem with the input file (e.g., an invalid character or value), an error message directs the user to the error file and the location of the error. If there is no problem reading the input file, the error file is deleted and the user does not see it. The error file's name is the input file's name plus the suffix `_error`.

Table 1  
*Simple Probabilities and Simple Frequencies*

|             |    |    |    |    |   |   |   |   |      |
|-------------|----|----|----|----|---|---|---|---|------|
| Event       | 1  | 2  | 3  | 4  | 5 | 6 | 7 | 8 |      |
| Probability | 1  | 1  | 1  | 1  | 1 | 1 | 1 | 1 | /8   |
| Event       | 1  | 2  | 3  |    |   |   |   |   |      |
| Probability | 8  | 1  | 1  |    |   |   |   |   | /10  |
| Event       | 1  | 2  | 3  | 4  | 5 |   |   |   |      |
| Frequency   | 3  | 3  | 3  | 3  | 3 |   |   |   | /15  |
| Event       | 1  | 2  | 3  | 4  |   |   |   |   |      |
| Frequency   | 60 | 60 | 30 | 30 |   |   |   |   | /180 |

Table 2  
*Transition Matrices with Conditional Probabilities or  
 Conditional Frequencies—Exact/Block*

| Context | Event |    |   |   |     |
|---------|-------|----|---|---|-----|
|         | 1     | 2  | 3 | 4 |     |
| 1       | 0     | 8  | 4 | 4 | /16 |
| 2       | 4     | 0  | 4 | 8 | /16 |
| 3       | 8     | 4  | 0 | 4 | /16 |
| 4       | 4     | 4  | 8 | 0 | /16 |
| 1-1     | 3     | 5  | 2 |   | /10 |
| 1-2     | 5     | 2  | 3 |   | /10 |
| 1-3     | 5     | 3  | 2 |   | /10 |
| 2-1     | 2     | 5  | 3 |   | /10 |
| 2-2     | 5     | 3  | 2 |   | /10 |
| 2-3     | 5     | 2  | 3 |   | /10 |
| 3-1     | 3     | 5  | 2 |   | /10 |
| 3-2     | 5     | 2  | 3 |   | /10 |
| 3-3     | 5     | 3  | 2 |   | /10 |
| 1-1-2   | 25    | 25 |   |   | /50 |
| 1-2-1   | 25    | 25 |   |   | /50 |
| 1-2-2   | 50    | 0  |   |   | /50 |
| 2-1-1   | 0     | 50 |   |   | /50 |
| 2-1-2   | 25    | 25 |   |   | /50 |
| 2-2-1   | 25    | 25 |   |   | /50 |

Table 3  
*Input Files for the Simple Probabilities and Simple Frequencies  
 in Table 1*

---

8 /  
 1 1 1 1 1 1 1 1 /8

---

3 /  
 8 1 1 /10

---

5 /  
 3 3 3 3 3 /15

---

4 /  
 60 60 30 30 /180

---

*Note.* The header of an input file specifies the number of distinct events and must end with a forward slash. Numbers in an input file must be separated by at least one whitespace (i.e., space, tab, or hard return). If a number precedes a forward slash, at least one whitespace must intervene. An input file must be a text (\*.txt) file.

Table 4  
*Input Files for the Transition Matrices in Table 2*

---

```
4 1 4 /
1 0 8 4 4 /16
2 4 0 4 8 /16
3 8 4 0 4 /16
4 4 4 8 0 /16
```

---

```
9 2 3 /
1 1 3 5 2 /10
1 2 5 2 3 /10
1 3 5 3 2 /10
2 1 2 5 3 /10
2 2 5 3 2 /10
2 3 5 2 3 /10
3 1 3 5 2 /10
3 2 5 2 3 /10
3 3 5 3 2 /10
```

---

```
6 3 2 /
1 1 2 25 25 /50
1 2 1 25 25 /50
1 2 2 50 0 /50
2 1 1 0 50 /50
2 1 2 25 25 /50
2 2 1 25 25 /50
```

---

*Note.* The header of an input file specifies the number of contexts, the context size, and the number of distinct events, and must end with a forward slash. Numbers in an input file must be separated by at least one whitespace (i.e., space, tab, or hard return). If a number precedes a forward slash, at least one whitespace must intervene. An input file must be a text (\*.txt) file.