|  |
| --- |
| National Drought Mitigation Center - UNL |
| SPI Generator |
| NDMC  11/29/2018 |

# Overview

The SPI Generator application serves to generate SPI (Standard Precipitation Index) data and illustrate how to interact with the SPI DLL (Dynamically Linked Library). The application reads in precipitation data and supports different time scales and data types (weekly, monthly). It outputs SPI data and, optionally, frequency and drought period data. In addition to executing as a Windows GUI (Graphical User Interface), it can also execute from the command line.

Table of Contents

[Overview 2](#_Toc528928146)

[Standard Precipitation Index Summary 2](#_Toc528928147)

[Technical Details 3](#_Toc528928148)

[Command Line Application 3](#_Toc528928149)

[Usage Examples 4](#_Toc528928150)

[Command Line 4](#_Toc528928151)

[Windows GUI 5](#_Toc528928152)

[Input Files 9](#_Toc528928153)

[Formats 9](#_Toc528928154)

[Missing Data 10](#_Toc528928155)

[Output 11](#_Toc528928156)

[SPI 11](#_Toc528928157)

[Frequencies 13](#_Toc528928158)

[Drought Periods 13](#_Toc528928159)

[Error Logging 14](#_Toc528928160)

[Contact Information 15](#_Toc528928161)

# Standard Precipitation Index Summary

The SPI is calculated from the historical precipitation record at a weather station, where precipitation accumulation over a period of time is compared to that same period of time throughout the historical record at that location. The SPI for any precipitation accumulation value represents the probability that the location would have received at least the observed amount of precipitation over the time period. The SPI calculated in this program is based on representing the historical precipitation record with a gamma distribution. Positive SPI values represent wet conditions; the higher the SPI, the more unusually wet a period of time is. Negative SPI values represent dry conditions; the lower the SPI, the more unusually dry a period of time is.

The primary strength of the SPI is that its data can be compared between different climate regions. Since the SPI compares precipitation amounts at a weather station to that particular station’s own precipitation history, SPI data from different stations usually can be directly compared, because the SPI communicates how anomalous a precipitation amount is, not just the raw departure from normal precipitation. For regions with highly seasonal precipitation (particularly where very little to no precipitation usually falls for several months), users should be cautious when directly comparing data from these areas. The SPI generator can also be used to compute standardized indices for other variables (commonly, streamflow). Some caution should be exercised when analyzing this output, since the gamma distribution may not be the best distribution to use to model non-precipitation variables.

# Technical Details

The application requires Microsoft’s .NET 4.0 framework and can be compiled as a 32- or 64-bit executable. The application consists of two files: a vb.net 4.0 application and a c#.net 4.0 DLL.

* SPIGenerator.exe – the Windows console application.
* StandardPrecipitationIndex.dll – the SPI DLL.

Running the application as a Windows application or command line produces the same program flow:

1. Retrieve and verify input parameters
2. Parse input file
3. Aggregate data
4. Calculate SPI
5. Write results to file

# Command Line Application

If the application receives a series of parameters sent via the command line, the application will run without producing anything but the final output. Every exception encountered using the command line is logged to the Windows Application Event Log. The following parameters may be sent in no particular order.

|  |  |  |  |
| --- | --- | --- | --- |
| -i | input file | required if -d is empty | The full path to the input file (i.e., C:\temp\monthly\_input.txt). |
| -d | input directory | required if -i is empty | Directory path containing input files C:\inputdata\monthly.  A directory or file may be selected for input, not both. |
| -o | output directory | required | The full path to the output directory (i.e., D:\sample output). |
| -s | scale | required | List of time scales for aggregating data separated by a comma (i.e., 1,2,3). The list of time scales must not have any spaces. The difference is subtle: (1,24,48) not (1, 24, 48). |
| -t | type | required | The type of input data (i.e., daily, weekly, monthly). |
| -a | aggregate | required if -t is set to daily | Interval for aggregating data (i.e., weekly, monthly). If the -a argument is not set, the output format type will automatically be set to the input format type. Daily data must be aggregated to weekly or monthly. |
| -f | frequency |  | Flag indicating frequency data should be output. |
| -p | drought periods |  | Flag indicating drought periods should be output. |
| -l | data delimiter | required | Type of delimiter used to separate data (i.e., comma, space, tab). |
| -e | international date |  | Flag indicating if date should be output with an international format  (i.e., yyyy-mm-dd). |
| -b | international decimal |  | Flag indicating if decimal values are output with a comma instead of decimal point. |
| -r | output type |  | Flag indicating output type (i.e., comma, space, excel). |

# Usage Examples

Sample command line and Windows GUI application scenarios are below. Both application types use the same input files and export the same output files. Output file data is based on the example input file.

## Command Line

SPIGenerator.exe -i "C:\temp\monthly.txt" -o "D:\output" -s 1 -t monthly –f –l comma

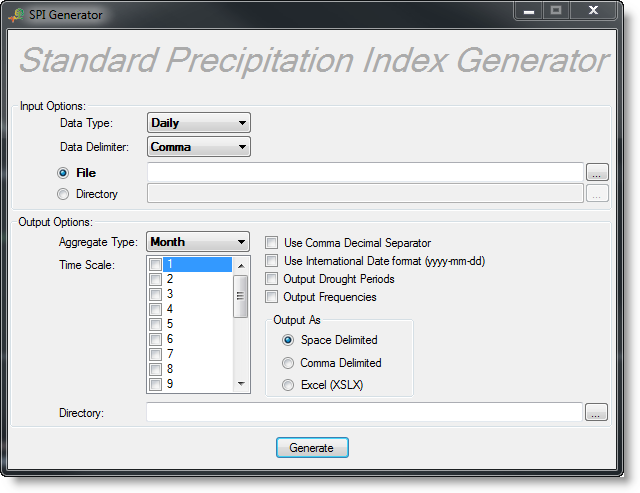
SPIGenerator.exe -i "C:\temp\monthly.txt" -o "D:\output" -s 12 -t monthly –l space

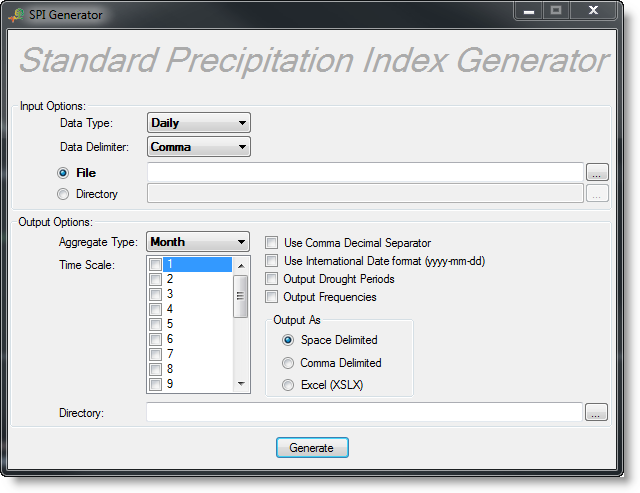
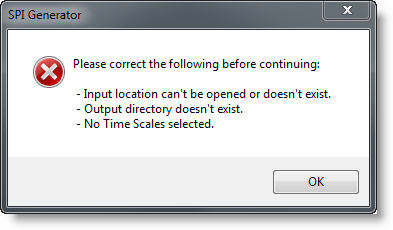
SPIGenerator.exe -i "C:\temp\daily.txt" -o "D:\output" -s 1,2 -t monthly –p –l comma

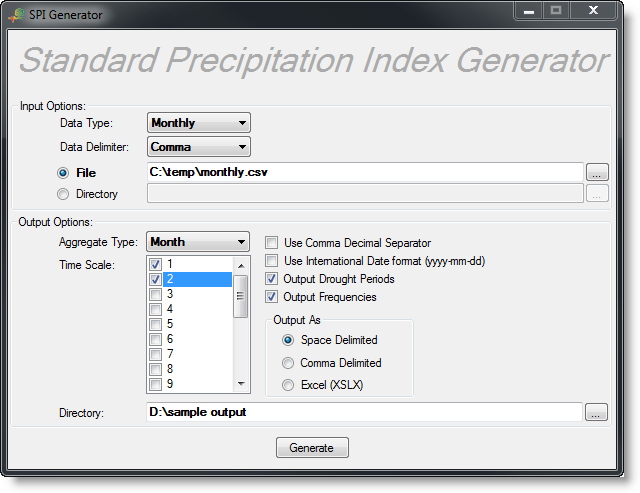
SPIGenerator.exe -d "C:\temp\monthly" -o "D:\output" -s 1,24,48,96 -t m –l c

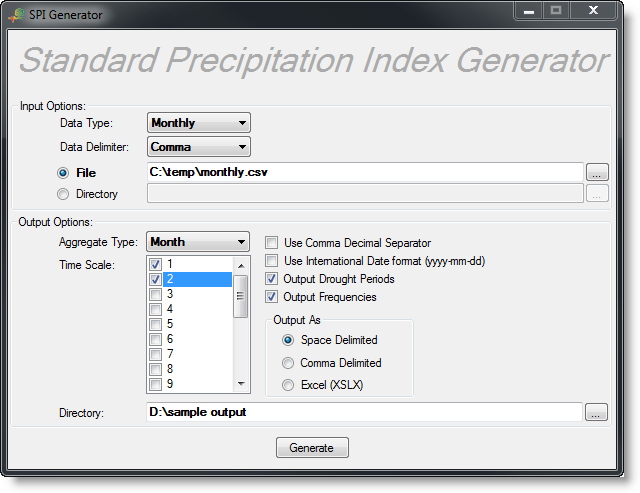
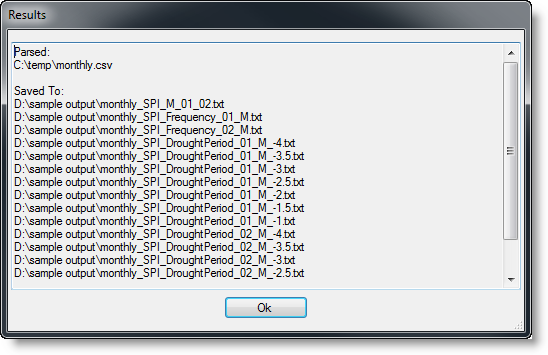
SPIGenerator.exe -d "C:\temp\daily" -o "D:\output" -s 1,12,24,48 -t d -a w –l t

## Windows GUI









# Input Files

The application will accept two types of file formats for input. Both types of input files should contain a complete set of data for analysis; SPI values will be calculated for all input data in the file.

The first line of both input files (generally the station name) is transferred to the first line of the output file. The second line of the first type of input file is the year and month for the first record in the time series. The remaining lines contain the monthly/weekly/daily precipitation data. The second type of input file lists dates in the first column of each record of precipitation.

The application expects data to be comma, space, or tab delimited, with individual lines separated by newlines. Any fields encountered after the expected fields will be ignored.

The sample input directory in the uncompressed application directory contains samples of all valid input data formats.

Other Notes:

* The aggregate type argument applies if the input type argument is set to daily or weekly; otherwise, the application automatically uses the input format type for the output format type.
* If parsing a directory, only files with a .txt, .cor, .csv, or .dat extension are parsed. Any extension may be used if parsing a single file.
* The .cor is a legacy file type from the SPI executable that was previously available from the NDMC and can be used only for monthly data.
* No fields may be omitted since the sequence of fields determines their meaning.
* Use -99.0 or -9999.0 to indicate no value.

## Formats

|  |  |
| --- | --- |
| header | a string to be passed as the header of the output file |
| yyyy | year |
| mm | month |
| pp.pp | precipitation |
| mm/dd/yyyy | date indicating the first day of the week (any valid date string is accepted) |

|  |  |  |
| --- | --- | --- |
| **comma delimited** | **space delimited** | **tab delimited** |
| header  yyyy,mm  pp.pp  pp.pp  pp.pp  … | header  yyyy mm  pp.pp  pp.pp  pp.pp  … | header  yyyy mm  pp.pp  pp.pp  pp.pp  … |
|  |  |  |

|  |  |  |
| --- | --- | --- |
| **comma delimited** | **space delimited** | **tab delimited** |
| header  mm/dd/yyyy,pp.pp  mm/dd/yyyy,pp.pp  mm/dd/yyyy,pp.pp  … | header  mm/dd/yyyy pp.pp  mm/dd/yyyy pp.pp  mm/dd/yyyy pp.pp  … | header  mm/dd/yyyy pp.pp  mm/dd/yyyy pp.pp  mm/dd/yyyy pp.pp  … |
|  |  |  |

# Missing Data

If the user chooses daily data as the input, it is very important that each day in the input record has a value, whether a real value or a missing data value (-99 or -9999). This means 1) that every day must have a value, and 2) that there cannot be any days missing from the time series between the start and end date. For example, the record cannot skip from 1 February to 3 February, or from 1 March to 1 April. If the user fails to supply a valid or missing data value for any of the dates in the record (i.e., if the record they supply is missing one or more dates), ***the output from the SPI generator will be incorrect***. The same rules apply to inputs of weekly or monthly data: if not all months or weeks in the record are supplied, ***the SPI output may be incorrect. It is the user’s responsibility to supply complete and correct data.***

Please remember the following when supplying daily data input:

* Daily data are summed by month (either calendar month or 4-week period, depending on which time step [monthly or weekly] is being used). Thus, missing daily data values, when being aggregated into monthly sums, are effectively treated as zeroes. For example, if 29 out of 31 input daily values for January 2003 were -99s, the value that would be converted to a 1-month SPI for January 2003 would be the sum of the two good values from that month.
* If an entire time step period (either a week or month) is filled with missing data values, any resulting output that has that particular time step in its aggregation period will be a missing data value. An example for monthly time steps: if a 3-month SPI were being calculated on a monthly time step for October 2000, the SPI-3 output for October would be a missing data value if one or more of August, September, and October 2000 is filled with missing data values. An example for weekly time steps: Each year in the record is made up of 7-day weeks, starting from January 1-7, 8-14, and so on. For weekly time steps, each “month” is a block of 4 consecutive weeks. If the user were calculating a 3-month SPI on a weekly time step, if one or more of the 12 weeks that went into that 3 “month” period were completely filled with missing data values, the resulting SPI output for that 3-month period would be a missing data value.

# Output

The application will output data to a space-delimited, comma-separated, or Excel file. The file format is the same using either the Windows GUI or command line.

## SPI

Each output file will begin with the header from the input file followed by column headers. SPI data can be created in weekly or monthly time steps; to create weekly SPI data, the user must start with either daily or weekly data. To create monthly SPI data, the user may start with daily, weekly or monthly data. The weekly SPI will always be calculated using four seven-day weeks to equal a month; the monthly calculations will use calendar months. Monthly output can only be derived from daily or monthly input data, not from weekly input data. If you have weekly input and want monthly SPI data, keep in mind that weekly aggregated SPI treats months as 4-week periods. Thus, although you cannot find a unique SPI value for each month when using weekly inputs, you can subsample the weekly output results. The column headers specify the time scale used for calculating data in the column. spi1 refers to an SPI with a time scale of 1 month while spi24 would indicate a time scale of 24 months. Column headers are followed by rows containing a date and the SPI:

header

date spi1

mm/dd/yyyy spi

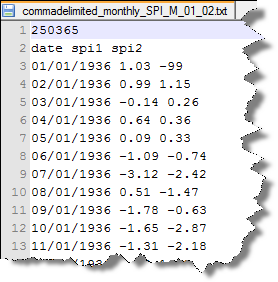
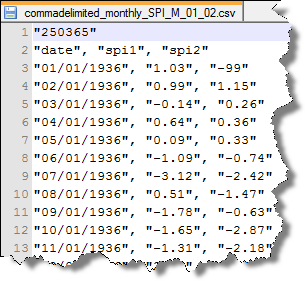
mm/dd/yyyy spi

mm/dd/yyyy spi

…

A -99.0 indicates there is no value.

The output file for raw SPI values is named using the name of the input file, the text SPI, the first letter of the aggregate type, time scales, and a .txt, .csv, or .xlsx extension (i.e., 451968\_SPI\_M\_01\_02.txt or 50136\_SPI\_W\_01.txt). After the header line and column labels are printed, the remaining lines list the resulting date and SPI values.

## Frequencies

The frequency of each SPI value can be optionally calculated and output to a file. The frequency output file is named using the name of the input file, the text SPI\_Frequency, time scale, aggregate type, and a .txt, .csv, or .xlsx extension (i.e., 451968\_SPI\_Frequency\_01\_M.txt or 50136\_SPI\_Frequency\_24\_W.txt). After the header line and column labels are printed, the remaining lines list the SPI value, number of occurrences, percentage, and ranking percentile.

|  |  |
| --- | --- |
|  |  |

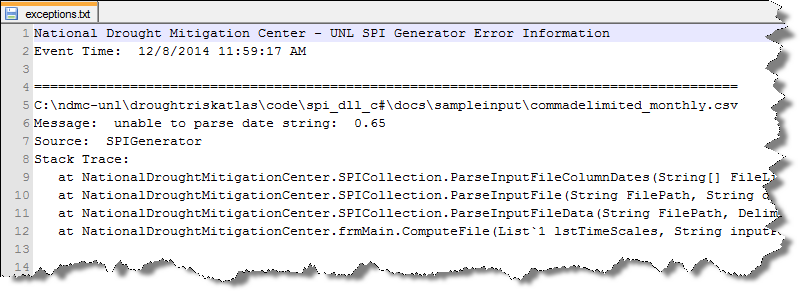
## Drought Periods

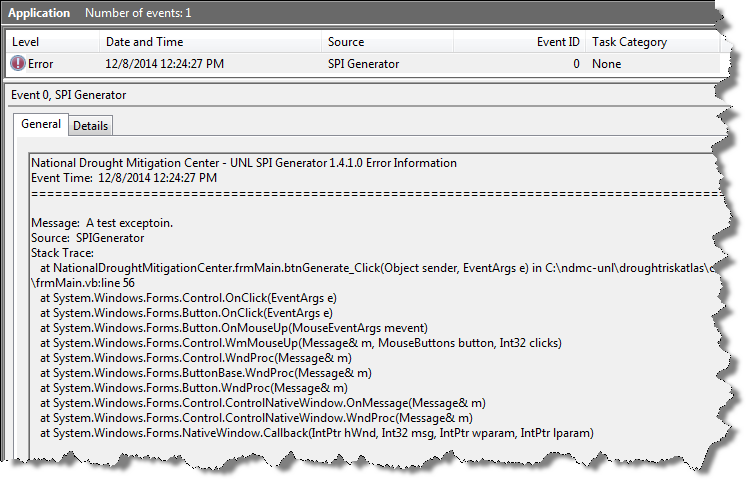
Drought periods can be optionally calculated and output to a file. The following levels are used to compute drought periods: -1, -1.5, -2, -2.5, -3, -3.5, -4. If no drought periods are calculated, a file is still generated with a message indicating no drought periods were found. The output file is named using the name of the input file, the text SPI\_DroughtPeriod, time scale, aggregate type, drought level, and a .txt, .csv, or .xlsx extension (i.e., 451968\_SPI\_DroughtPeriod\_01\_M\_-1.txt or 50136\_SPI\_DroughtPeriod\_06\_M\_-1.5.txt). After the header and column labels are printed, the remaining lines list the start date, end date, duration, peak value, sum, average, and median.

|  |  |
| --- | --- |
|  |  |

# Error Logging

The Windows GUI displays basic exceptions to the user. Exceptions encountered using the command line are logged to an exception file in the output directory (if the directory is not the cause of the problem) or the application directory. Exceptions encountered while processing a file or directory for either the command line or Windows application are logged to the exception file. All unhandled exceptions are logged to the Windows Application Event Log.





# Contact Information

For questions about this application, please contact the National Drought Mitigation Center.

Contact for this application should be directed to:

[NDMCdata@unl.edu](mailto:NDMCdata@unl.edu)

Please use “NDMC SPI application” as at least part of the subject line and please include as much information as you can, including screenshots and specifics about versions of the applications and your operating system, if you are contacting us with questions about errors.