ABSTRACT
Singular spectrum analysis (SSA) is a relatively new approach to modeling time series data. Now supported in SAS/ETS software, the SSA method of time series analysis applies nonparametric techniques to decompose time series into principal components. SSA is particularly valuable for long time series, for which patterns (such as trends and cycles) are difficult to visualize and analyze.

This paper provides an introduction to singular spectrum analysis and demonstrates how to use SAS/ETS software to perform SSA. As an illustration, monthly data on U.S. temperatures over the last century are analyzed to discover significant patterns.

INTRODUCTION
Time series data often contain trends, cycles, anomalies, and other components. For long time series, these patterns are often difficult to visualize and discover. Singular spectrum analysis (SSA) applies nonparametric techniques that adapt the commonly used principal components analysis (PCA) for decomposing time series data. These components can help the analyst discover and understand the various patterns contained in the time series. Once these components are understood by the analyst, each component series can be modeled and forecasted separately; then, the component series forecasts can be aggregated to forecast the original series under investigation.

BACKGROUND
This section provides a brief theoretical background on singular spectrum analysis. It is intended to provide the analyst with motivation, orientation, and references. An introductory discussion of singular spectrum analysis can be found in Golyandina, Nekrutkin, and Zhigljavsky (2001) and in Elsner and Tsonis (1996).

Given a time series $y_t$ for $t = 1, \ldots, T$ and a window length $2 \leq L < T/2$, singular spectrum analysis decomposes the time series into spectral groupings using the following steps:

1. **Embedding step:** Using the time series, form a $K \times L$ trajectory matrix $X = \{x_{k,l}\}_{k=1,l=1}^{K,L}$ such that

   $x_{k,l} = y_{k-l+1}$ for $k = 1, \ldots, K$ and $l = 1, \ldots, L$ where $K = (T - L + 1)$. By definition, $L \leq K < T$ because $2 \leq L < T/2$.

2. **Decomposition step:** Apply the singular value decomposition to the trajectory matrix $X = UV^T$ where $U$ represents the $(K \times L)$ matrix that contains the left-hand-side (LHS) eigenvectors, $Q$ represents the diagonal $(L \times L)$ matrix that contains the singular values, and $V$ represents the $(L \times L)$ matrix that contains the right-hand-side (RHS) eigenvectors.

   Therefore, $X = \sum_{i=1}^{L} X^{(i)} = \sum_{i=1}^{L} u_i q_i v_i^T$ where $X^{(i)}$ represents the $(K \times L)$ principal component matrix, $u_i$ represents the $(K \times 1)$ left-hand-side (LHS) eigenvector, $q_i$ represents the singular value, and $v_i$ represents the $(L \times 1)$ right-hand-side (RHS) eigenvector that is associated with the $i$th window index.

3. **Grouping step:** For each group index, $m = 1, \ldots, M$, define a group of window indices $I_m \subset \{1, \ldots, L\}$. Let $X_{I_m} = \sum_{i \in I_m} X^{(i)} = \sum_{i \in I_m} u_i q_i v_i^T$ represent the grouped trajectory matrix for group $I_m$. 
Note that if groupings represent a spectral partition, \( \bigcup_{m=1}^{M} I_m = \{1, \ldots, L\} \) and \( I_m \cap I_n = \emptyset \) for all \( m \neq n \), then according the singular value decomposition theory, \( X = \sum_{m=1}^{M} X_{I_m} \).

4. **Averaging step:** For each group index, \( m = 1, \ldots, M \), compute the diagonal average of
\[
\tilde{x}_{t}^{(m)} = \frac{1}{n_t} \sum_{t-s_t}^{s_t} x_{(t-l+1),l}^{(m)}
\]
where \( s_t = 1, e_t = t, n_t = t \) for \( 1 \leq t < L \)
\( s_t = 1, e_t = L, n_t = L \) for \( L \leq t \leq (T-L+1) \)
\( s_t = (T-t+1), e_t = L, n_t = (T-t+1) \) for \( (T-L+1) < t \leq T \)

Note that if groupings represent a spectral partition, \( \bigcup_{m=1}^{M} I_m = \{1, \ldots, L\} \) and \( I_m \cap I_n = \emptyset \) for all \( m \neq n \), then by definition, \( y_t = \sum_{m=1}^{M} \tilde{x}_{t}^{(m)} \). Hence, singular spectrum analysis additively decomposes the original time series, \( y_t \), into \( m \) component series: \( \tilde{x}_{t}^{(m)} \) for \( m = 1, \ldots, M \).

5. **Forecasting step (optional):** If the groupings represent a spectral partition, then each component series, \( \tilde{x}_{t}^{(m)} \) for \( m = 1, \ldots, M \), can be modeled and forecasted independently using an appropriate time series model (ARIMAX, unobserved component model, exponential smoothing model, and others) possibly using different time series models which include different input series (causal factors) and calendar events (interventions).

Let \( \hat{x}_{t}^{(m)} \) for \( m = 1, \ldots, M \) represent the component series forecasts derived from the \( m \)th independent time series model. Then, the forecast for the original time series, \( \hat{y}_t \), can be derived by simply aggregating the component series forecasts:
\[
\hat{y}_t = \sum_{m=1}^{M} \hat{x}_{t}^{(m)}
\]

The SSA forecasting step represents a clever forecast model combination technique.

**SAS IMPLEMENTATION**

The singular spectrum analysis described in the previous section can be performed with SAS/ETS software. This section describes how the TIMESERIES procedure analyzes time-stamped and time series data.

**PROC TIMESERIES Statement**

The PROC TIMESERIES statement has the following additional options that are related to SSA (for options related to other analyses, see the SAS/ETS User’s Guide):

- **OUTSSA=SAS-data-set**
  
  names the output data set to contain the singular spectrum analysis result series.

- **PLOTS=option | (options)**

  - **SSA** plots the singular spectrum analysis results.

- **PRINT=option | (options)**

  - **SSA** prints the singular spectrum analysis results.
SSA Statement

The SSA statement is a new statement that can be used with the TIMESERIES procedure to specify options that are related to singular spectrum analysis (SSA) of the accumulated time series. Only one SSA statement is allowed.

The SSA Statement has the following syntax:

SSA < / options > ;

The following options can be specified in the SSA statement following the slash (/):

**LENGTH=** *number*

specifies the window length to be used in the analysis. It represents the maximum lag used in the SSA trajectory matrix. The number specified by the LENGTH= option must be greater than 1 and less than 1000. When the SEASONALITY= option is provided or implied by the INTERVAL= option in the ID statement, the default window length is the smaller of two times the length of the seasonal cycle and one-half the length of the time series. When no seasonality value is available, the default window length is the smaller of 12 and one-half the length of the time series.

For example, the following SSA statement specifies a window length of 10:

```plaintext
ssa / length=10;
```

As another example, the following SSA statement specifies a window length of 24 if the INTERVAL=MONTH or SEASONALITY=12 options are specified:

```plaintext
ssa;
```

If the specified window length is greater than what is feasible based on one-half the length of the accumulated time series, the window length is reduced and a warning message is printed to the log.

**THRESHOLD=** *percent*

specifies the threshold value used to determine the size of the last group based on the cumulative percent of the singular values. The percentage specified by the THRESHOLD= option must be greater than zero and less than 100. The default is 90% (THRESHOLD=90).

For example, the following SSA statement specifies a threshold of 80%:

```plaintext
ssa / threshold=80;
```

As another example, the following SSA statement specifies a threshold of 90%:

```plaintext
ssa;
```

The size of the last group must be at least 1 but less than the window length, and the threshold is adjusted to achieve this requirement.

For example, the following SSA statement specifies a threshold of 0% and implies that the size of the last group is one less than the window length:

```plaintext
ssa / threshold=0;
```

As another example, the following SSA statement specifies a threshold of 100% and implies that the size of the last group is one:

```plaintext
ssa / threshold=100;
```

**GROUPS=**(numlist) …(numlist)

specifies the list of groups of window lags to be stored in OUTSSA= data set or plotted. The window lags must be separated by spaces or commas. For example, GROUPS=(1 3) (2 4) specifies that first and third window lags form the first group and the second and fourth window lags form the second group. The default is to evenly divide the window into two groups based on the window length specified by the LENGTH= option.
For example, the following SSA statement specifies three groups:

```
ssa / groups=(1 3)(2 4 5)(6);
```

The first group contains the first and third principal components, the second group contains the second, fourth, and fifth principal components, and the third group contains the sixth principal component.

For example, the following SSA statement specifies two groups:

```
ssa;
```

The first group contains the principal components whose spectrum sums to greater than the threshold of 90%; the second group contains the remaining principal component.

**Specifying the Window Length**

You can explicitly specify the maximum window length, \(2 \leq L \leq 1000\), by using the `LENGTH=` option in the SSA statement, or you can implicitly specify the window length by using `INTERVAL=` option in the ID statement or the `SEASONALITY=` option in the PROC TIMESERIES statement.

Either way the window length is reduced based on the accumulated time series length \(T\) to enforce the requirement that \(2 \leq L < T/2\).

**Specifying the Groups**

You can explicitly specify the grouping \(I_m \subset \{1, \ldots, L\}\) by using the `GROUPS=` option in the SSA statement, or you can implicitly specify the grouping with the `THRESHOLD=` option in the SSA statement. The `THRESHOLD=` option is useful for removing noise or less dominant patterns from the accumulated time series.

Let \((0 < \alpha < 1)\) be the cumulative percent singular value threshold. Then \(I_M\) (the last group) is determined by the following threshold:

\[
\min\left(\left\{l_\alpha - 1 \left(\frac{\sum_{i=1}^{l_\alpha} q_i}{\sum_{i=1}^{l_\alpha} q_i} \right) \geq \alpha\right\}\right)\text{ where } I_M = \{l_\alpha, \ldots, L\} \text{ where } 1 < l_\alpha \leq L
\]

Using this rule, the last group \(I_M = \{l_\alpha, \ldots, L\}\) describes the least dominant patterns in the time series, and the size of the last group is at least one and less than the window length, \(L \geq 2\).

**EXAMPLE**

To illustrate the use of SSA in SAS/ETS software, monthly data on U.S. temperatures over the last century are analyzed to discover significant patterns.

**Basic Time Series Analysis**

The monthly temperature anomaly (degrees Celsius) for the U.S. over the last 128 years provided by National Oceanographic Atmospheric Administration (NOAA) was analyzed. The temperature anomaly is seasonally adjusted using the reference decade of the 1960s. First the time series was plotted using the TIMESERIES procedure as follows:

```
proc timeseries data=NOAA out=ZERO plot=(SERIES CYCLES);
  id DATE interval=MONTH;
  var TEMPERATURE;
run;
```

`DATA=NOAA` specifies that the data set Work.NOAA contains the temperature anomaly records. The `ID` statement specifies that the time ID variable is `DATE` and the time interval is `MONTH`. The `VAR` statement specifies that the variable under analysis is `TEMPERATURE`. The `PLOT=(SERIES CYCLES)` option request that the series and the year-over-year monthly cycles be plotted.
Figure 1 illustrates the SERIES plot. The X axis represents the time ID (DATE), and the Y axis represents the temperature anomaly (TEMPERATURE). As you can see from this plot, it is difficult to see any patterns in the time series because of its length and variation.

Figure 1. Monthly Series Plot of the Temperature Anomaly
Figure 2 illustrates the CYCLES plot. The X axis represents the monthly seasonal index (January=1, ..., December=12), and the Y axis represents the temperature anomaly (TEMPERATURE). Each line represents one year (128 seasonal cycles). As you can see from this plot, the series has no discernible monthly pattern as expected because the time series is seasonally adjusted.

![Seasonal Cycles Plot of the Temperature Anomaly](image)

**Figure 2. Seasonal Cycles Plot of the Temperature Anomaly**

**Singular Spectrum Analysis**

Next, singular spectrum analysis is applied using a threshold value for the eigenspectrum. The time series is analyzed using the TIMESERIES procedure as follows:

```sas
proc timeseries data=noaa out=_NULL_ plot=(series cycles SSA);
   SSA / LENGTH=120 THRESHOLD=80;
   id date interval=month;
   var temperature;
run;
```

The SSA statement LENGTH=120 option specifies that a window length of 120 (ten years), and the THRESHOLD=80 option specifies an eigenspectrum threshold value of 80%. Including SSA as one of the values in the PLOT= option requests that the SSA analysis be plotted.
Figure 3 illustrates the eigenspectrum plot. The first graph illustrates the eigenspectrum, and the second graph illustrates the cumulative percentage of the eigenspectrum on the Y axis. The X axis represents the window lags. As you can see from this graph, the eigenspectrum decreases rapidly after the seventh lag. Close inspection reveals that there are four “steps” of equal value in the eigenspectrum plot: (1)(2)(3 4)(5 6 7).

Next, singular spectrum analysis is applied using grouping of the eigenspectrum. The time series is analyzed using the TIMESERIES procedure as follows:

```sas
proc timeseries data=noaa out=_NULL_ plot=(series cycles ssa) OUTSSA=SSA;
  ssa / length=120 GROUP=(1)(2)(3 4)(5 6 7);
  id date interval=month;
  var temperature;
run;
```

The SSA statement GROUP=(1)(2)(3 4)(5 6 7) specifies that the series be decomposed into four spectral groups. The first group contains the first lag, the second group contains the second lag, the third group contains the third and fourth lags, and the fourth group contains the fifth, sixth, and seventh lags. The OUTSSA=SSA option specifies that the spectral grouping be stored in the Work:SSA data set. Since three spectral groups are requested, the data set contains four variables (GROUP1, GROUP2, GROUP3, and GROUP4).
Figure 4 illustrates the first group. In the first graph, the black line represents the original series and the blue line represents the first group. In the second graph, the blue line represents the first group. As you can see from the plot, the first group represents the dominant trend in the temperature anomaly series. From this plot, it appears that temperatures have increased about one degree over the last century.

Figure 4. First Spectral Group of the Temperature Anomaly
Figure 5 illustrates the second group. As you can see from this plot, the second group represents the dominant long-term cycle in the temperature anomaly series.

Figure 5. Second Spectral Group of the Temperature Anomaly
Figure 6 illustrates the spectral density plot for the second group. From this plot, there appears to be an approximately 22-year cycle (SEASONALITY=252) possibly related to the Hale solar cycle.

Figure 6. Spectral Density of the Second Spectral Group
Figure 7 illustrates the third group. As you can see from this plot, the third group represents the dominant short-term cycle in the temperature anomaly series. It appears that the variation is small for the reference decade of the 1960s.

Figure 7. Third Spectral Group of the Temperature Anomaly
Figure 8 illustrates the spectral density plot for the third group. From this plot, there appears to be a monthly cycle (SEASONALITY=12).

Figure 8. Spectral Density of the Third Spectral Group
Figure 9 illustrates the fourth group. As you can see from this plot, the fourth group represents the dominant medium-term cycle in the temperature anomaly series.

Figure 9. Fourth Spectral Group of the Temperature Anomaly
Figure 10 illustrates the spectral density plot for the fourth group. From this plot, there appears to be an approximately five-year cycle (SEASONALITY=60) possibly related to the El Niño and La Niña cycle.
As you can see from the preceding analysis, this long series is effectively decomposed into spectral groups. Figure 11 illustrates all four spectral groupings. No model assumptions are made other than the window length (LENGTH= option) and spectral groupings (GROUP= option). This demonstrates the value of singular spectrum analysis in discovering patterns (especially cyclical patterns) in long series.

Figure 11. SSA Results for Temperature Anomaly

The preceding analysis decomposed the time series into additive components. Multiplicative components can be achieved by taking the log transform of the (positive-valued) time series.

Unobserved Component Model (UCM) Analysis

Now that the time series has been effectively decomposed into spectral groups, the first spectral grouping is analyzed using a basic trend model (state space model) using the UCM procedure:

```sas
proc ucm data=SSA;
   id date interval=month;
   model GROUP1;
   LEVEL;
   SLOPE PLOT=SMOOTH;
run;
```

The DATA= option of PROC UCM statement specifies the input data set. The ID statement specifies that the time ID variable is DATE and the time interval is MONTH. The MODEL statement specifies that the variable under analysis is GROUP1 (the first spectral group). The LEVEL statement specifies that a time-varying level component be included in the state-space model, and the SLOPE statement specifies that a time-varying slope component be included in the state space model.
Figure 12 illustrates the slope component over time.

Figure 12. Slope Component of the Temperature Anomaly
Figure 13 illustrates the distribution of the slope component. The results of this analysis (not shown) indicate that the mean/median filtered slope component is 0.00081 degrees per month (about 1 degree per century) with standard deviation of 0.00125 per month and that the final filtered slope component is (−0.0031) degree per month (about −3 degrees per century).

Other analyses can be applied to each of the spectral groups: time domain analysis, frequency domain analysis, component analysis, distribution analysis, forecasting, and others.

Temperatures for Each State

For another example, the temperatures of the lower 48 states over the last 100 years provided by NOAA are analyzed. The same analysis previously described is repeated for each state. The final filtered slope component for each state is collected. Figure 14 illustrates the distribution of the slope components. Figure 14 plots the slope components on the map.
Figure 14: Slope Component Distribution for each State
Comparison of the Aggregate and Disaggregate Analyses

The first example applied SSA to the aggregate of the temperatures (U.S. temperature anomaly). The second example applied SSA to the disaggregate temperatures (state temperature records) and analyzed the distribution. The value of this comparison enables you to consider potential aggregation bias and to better understand localized effects.

CONCLUSION

Singular spectrum analysis (SSA) is a very powerful tool for detecting patterns in long time series with few model assumptions. SSA effectively decomposes time series into spectral groupings. These spectral groupings can be individually analyzed using time series analysis techniques such as forecasting and state-space component analysis. This paper uses temperature records to illustrate how SAS/ETS software can be used to perform SSA.

Other cyclical time series can use this technique—for example, load forecasting (electric, gas, and water consumption), service centers (manpower, call centers, and customer support), and telecommunications (phone service, data centers, and Web servers). The geographic analysis shows how SSA can be used to determine localized trends for resource allocation—for example, new utility construction, new service locations, new telecommunication infrastructure, and others.

REFERENCES


CONTACT INFORMATION
Your comments and questions are valued and encouraged. Contact the author at:

   Michael Leonard
   SAS
   SAS Campus Drive
   Cary, NC 27513
   919-531-6967
   Michael.Leonard@sas.com

SAS and all other SAS Institute Inc. product or service names are registered trademarks or trademarks of SAS Institute Inc. in the USA and other countries. ® indicates USA registration.

Other brand and product names are trademarks of their respective companies.