

Time Dependent Data Exploration And Preprocessing: Doing It All by SAS.

Joseph Twagilimana, University of Louisville, Louisville, KY

ABSTRACT

This paper presents exploration and preprocessing methodology of transactional data in order to transform the data into a multivariate time series and select an adequate model for analysis. Unlike time series data, where observations are equally spaced by a specific time interval, in transactional data, observations are not spaced with respect to any particular time period. Our approach is illustrated using observations of length of stay (LOS) of a patient at a hospital Emergency Department (ED). The challenges of analyzing these data include autocorrelations of the observations, non-linearity, and the fact that observations were not recoded at regular time intervals. First, using the SAS procedure, PROC HPF, we transformed the transactional data set into multivariate time series data. Next, a series of specialized plots such as histograms, kernel density plots, boxplots, time series plots, and correlograms were produced using the SAS procedure PROC GPLOT to capture the essentials of the data to discover relationships in the variables, and to select an optimal model of analysis. As a result of this step by step preprocessing methodology, adequate models of analysis of LOS were identified and the dimension of the data set was reduced from 3345 observations to only 256 observations.

INTRODUCTION

The purpose of this paper is to examine the use of transactional time series data in a hospital emergency department (ED) in order to schedule efficiently Medical Personnel in the ED. Data range from those having strong time-dependency to those with little or no time relationship. When data are time-dependent, that is, sequentially collected in time, it is quite likely that the error terms are autocorrelated rather than independent. In that case, the usual linear models are not applicable since the fundamental assumption of independence of error required for linear models is violated.

When the data have been collected at constant intervals of time, such as each hour or each day, each week, each month and so on, time series analysis methods can be applied. However much time-dependent data are collected at irregular intervals of time and are referred to as *transactional data*. Time series analysis methods cannot be directly applied to transactional data. SAS has provided a procedure of transforming *transactional data* into time series data, PROC EXPAND. The problem with this procedure is that it cannot handle duplicate identifiers, resulting in procedural errors. Instead, the Proc HPF (for SAS High Performance Forecasting System), initially intended for forecasting, can be used to transform *transactional data* with duplicate identifiers and missing records into time series. This transformation is based on the fact that, by default, its forecast values are exactly the same as the actual values. This means that if you have observed n values and want to forecast the next k values with the Proc HPF, the result will be that the first n values will be identical to the observed values. Before performing this transformation, we need to get a deep insight view into the data so that we can wisely select an adequate accumulation time interval. An adequate choice of time interval retains the essential nature of the data and will also reduce the high dimensionality of the data from thousands of observation to some hundreds, without any loss of useful information.

Data exploration and preprocessing consist first of all in checking these assumptions concerning the data. In this paper, we use a clinical data set provided by electronic medical records from an Emergency Department (ED) to illustrate an exploratory and preprocessing methodology by the SAS system in order to select an adequate model for analysis and forecasting of the length of stay (LOS) at the ED. The Emergency Room is open 24 hours a day, 7 days a week. Patients having non-urgent to emergent conditions use the services of an ED. For every patient that visits the ED, the LOS is measured by subtracting triage time from release time. Hence the variable LOS is sequentially measured in time. Assumptions made about the variable LOS are:

- **Autocorrelation.** When observations on a variable are sequentially collected in time, a correlation between actual and previous observations of the variable, also called a serial correlation or autocorrelation, can be expected.
- **Stationarity.** It is also assumed that emergency departments are crowded at some times of the day and almost empty at other times. This means that the mean and variance of the LOS may vary with the time so that the time series resulting from the transformation of LOS is non-stationary. Therefore, in the preprocessing, it is necessary to check for stationarity.
- **Associations.** The data set may contain some others variables that are related to LOS. These variables may help explain the variation in the LOS variable. Association detection is also a task of data preprocessing.

In this paper, graphics and statistics tests are intensively used as hypotheses testing and preprocessing methodologies. Graphics include histograms, probability density plots and boxplots. Statistics tests include tests for normality (Kolmogorov-Smirnov, Cramer-von Mises, Anderson-Darling), tests for autocorrelation (Durbin-Watson test), and tests for stationarity (Dickey-Fuller unit root test). Another methodology used is the **SAS high forecast procedure** (Proc HPF) to transform transactional data into time series data.

VISUALIZING AND ANALYZING DATA DISTRIBUTIONS

Graphics are useful in that they let the researcher have an idea of how the data are distributed. As many statistical tests require data to be approximately normally distributed, it is important that when we investigate data distributions, a test for normality should be performed. Normality can be investigated graphically (Fig.1 and Fig.2) using a histogram plot with an overlaid approximating normal curve (Fig.1), or by one of the three (Kolmogorov-Smirnov, Cramer-von Mises, Anderson-Darling) statistical tests for normality offered by SAS in the univariate procedure (Proc Univariate) (Table.1). A combination of graphical methods and statistics tests may improve our judgments about the distributions of the data.

The Histogram in Figure1. shows that patients can stay as many as nine hours in the ED and reference lines show that more than 50% stay between One and Four hours. The distribution is clearly skewed to the right and the histogram shows that the lognormal distribution describe the data better than the normal. The code used to plot the histogram in figure one is presented below in Table1.

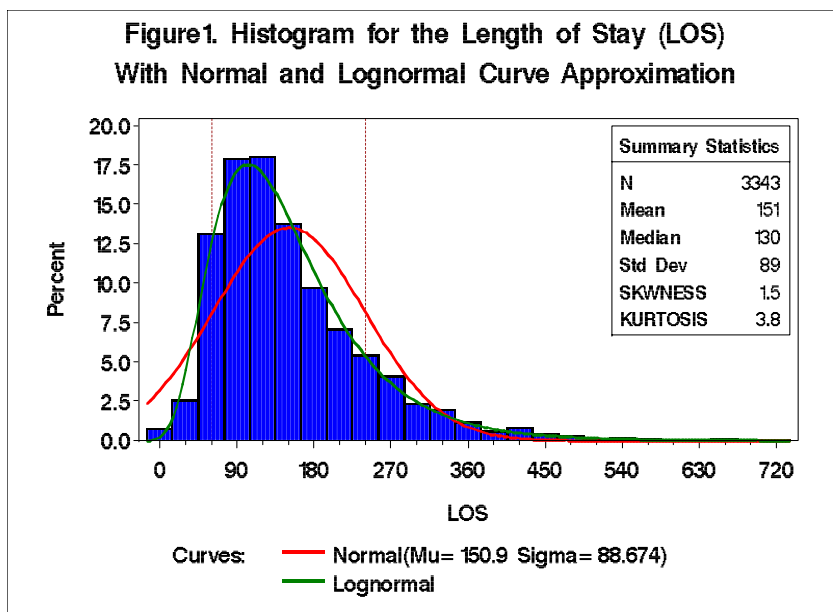


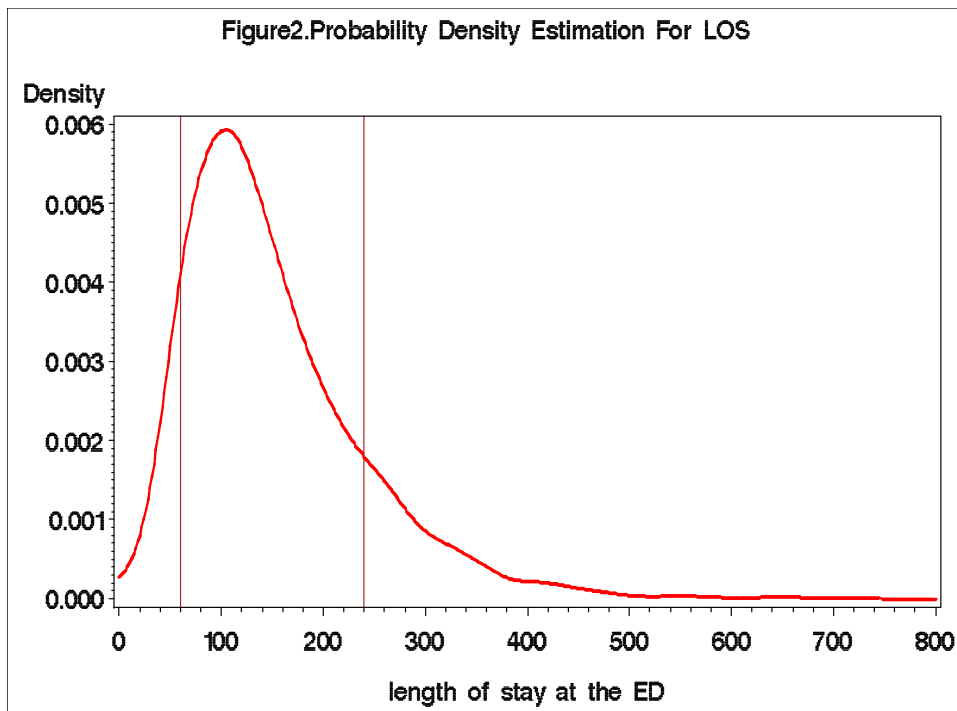
Table1: Code for the histogram in Figure 1.

```

options reset=all ctext = bl htext = 3.5pct ftext = swissb border;
symbol1 color=red i=join w=2 v=none ;
symbol2 color=green i=join w=2 v=none;
proc capability data=sasuser.Ibex2datetime;
var LOS;
histogram LOS /vaxis=axis1 cbarline=black cfill=blue normal(color=red w=3)
              lognormal (theta=est color=green w=3) ;
              inset n = "N"(5.0) mean = "Mean"(5.0)
MEDIAN="Median"(5.0)
              std = "Std Dev" (5.0) SKEWNESS="SKWNESS"(3.1)
KURTOSIS="KURTOSIS"(3.1)/
              pos = ne
              height = 3
              header = 'Summary Statistics';
              axis1 label=(a=90 r=0);
Title1 BOLD C=BL H=18pt FONT=swissb"Figure1. Histogram for the
              Length of Stay (LOS)";
Title2 BOLD C=BL H=18pt FONT=swissb"With Normal and Lognormal Curve
              Approximation";

run;
quit;

```



A better estimation of the density distribution can be obtained using a kernel density estimate (Figure2.). This can be done using the SAS procedure Proc KDE, as in the code provided in Table 2.

Table2: Code for Probability Density Estimation For LOS .

```

goptions reset=all;
goptions reset=all ctext = b1 htext = 3.5pct ftext = swissb border;
proc kde data= sasuser.Ibex2datetime grid1=0 gridu=800 method=SNR out=one;
var LOS;
run;
Title"Figure2.Probability Density Estimation For LOS";
proc gplot data=one;
symbol1 color=red i=spline w=3 v=none ;
plot density*LOS;
label LOS="length of stay at the ED";
axis1 label=(a=90 r=0);
run;

```

TRANSFORMING THE LOS VARIABLE

The lognormal approximation curve in figure one suggests that the lognormal distribution may be a good approximation of the distribution of the variable LOS. A lognormal distribution can be useful for modeling variables which have a peak in the distribution near zero and then taper off gradually for larger values. A random variable X has a lognormal distribution if its natural logarithm, $Y = \text{LOG}(X)$ has a normal distribution. We performed a logarithmic transformation and created a new variable LogLOS=Log (LOS) and then used Proc Univariate to check graphically and statistically the hypothesis of the distribution being lognormal. Testing that the variable LOS has a lognormal distribution is the same as testing if the transformed variable LogLOS is normally distributed.

The histogram in Figure 3 shows a nonnormal distribution and a very long, thin tail that dies from a value close to 2.6. This value is at a distance below three standard deviations (0.66) from the average (4.84). $4.84 - 3 * 0.66 = 2.86$. Therefore we may consider these values as outliers and discard them. When these outliers are discarded, we obtain a normal distribution, confirmed with both graphics (Figure4.) and statistics tests (Table 4). Although we cannot accept the null hypothesis, a failure to reject it indicates that it is fairly safe to assume normality.

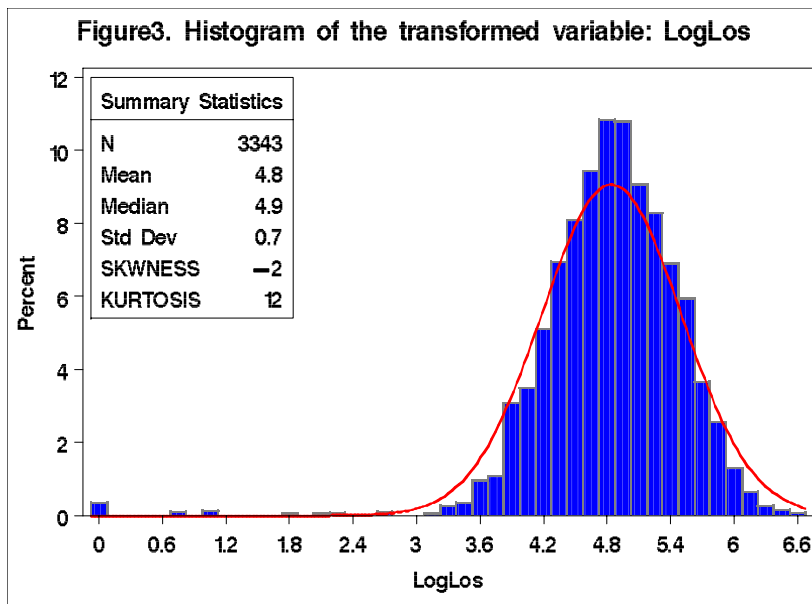


Table3. discarding outliers from the LOS variable.

```

goptions reset=all;
goptions reset=all ctext = black htext = 3.5pct ftext = swissb border;
symbol1 color=red i=join w=2 v=none ;
data sasuser.Ibex2datetime_logTransNooutlier;
set sasuser.Ibex2datetime_logTrans;
if logLOS lt 2.86 then delete;
run;
proc univariate data=sasuser.Ibex2datetime_logTransNooutlier ;
var logLOS ;
histogram logLOS / cbarline=grey cfill=blue normal normal(color=red w=3)
midpoints = 3 to 6.5 by 0.15 ;
inset n= "N" (5.0) mean = "Mean" (3.1) Median="Median"(3.1)std = "Std
Dev" (3.1)
SKEWNESS="SKWNESS"(3.1) KURTOSIS="KURTOSIS"(3.1)/
pos = nw height = 3.5 header = 'Summary Statistics';;
Title1 BOLD C=BL H=18pt FONT=swissb"Figure5. Histogram of LogLOS Without
Outliers";
run;

```

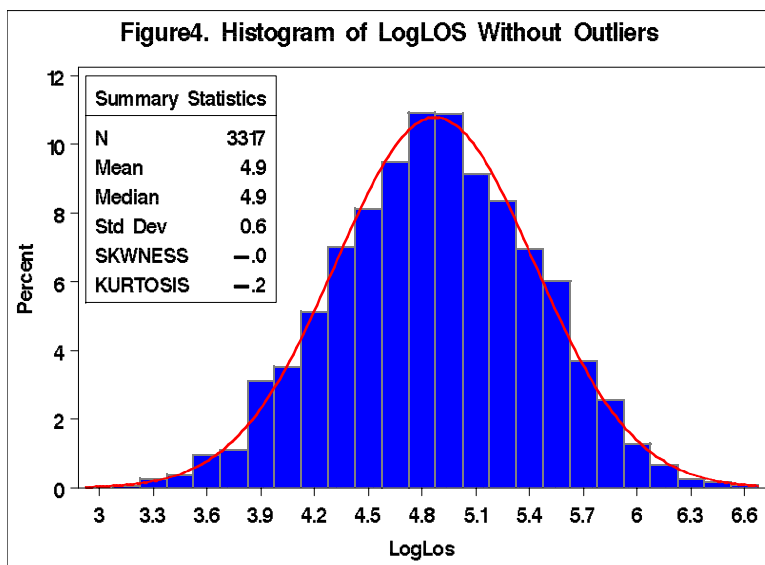


Figure4. shows that the histogram of LogLOS, after outliers are discarded shows a normal distribution. The following code (Table5.)was used to discard the outliers.

Table 4: Test statistics also support our hypothesis. Each one of these statistics supports the null hypothesis that LogLOS is normally distributed.

Basic Statistical Measures			
Location		Variability	
Mean	4.870702	Std Deviation	0.55793
Median	4.867534	Variance	0.31129
Mode	4.682131	Range	3.95124
		Interquartile Range	0.77405
Tests for Normality			
Test	--Statistic--		-----p Value-----
Kolmogorov-Smirnov	D	0.011952	Pr > D >0.1500
Cramer-von Mises	W-Sq	0.07348	Pr > W-Sq >0.2500
Anderson-Darling	A-Sq	0.592018	Pr > A-Sq 0.1277

USING PROC HPF TO TRANSFORM TRANSACTIONAL DATA INTO TIME SERIES.

Time series are sets of ordered observations or measurements $\{x_1, x_2, \dots, x_n\}$ recorded over time. Usually, the observations are successive and equally spaced in time. When observations are not equally spaced, the SAS procedure Proc Expand can be used to transform the data into equally spaced observations. The drawback of using Proc Expand is that when many observations are recoded at the same time, then the procedure will produce an error of duplicate ID. Fortunately Proc HPF can handle duplicate ID and even missing values. Normally, the procedure PROC HPF is used for forecast purposes. In the procedure, HPF, an option "lead=" must be used to indicate the number of forecasts desired. When that option is set to zero, no forecast will be produced. Instead, the option "accumulate=" will tell SAS to sum or average over a time period specified in the "interval=" option. The option "interval= Hourn." will accumulate over an n-hour period. In Table 5 we describe the codes that are used to transform the data set used in this paper to time series data with an accumulation interval of 8 hours.

Table 5: Code using Proc HPF to transform the transactional data set into a time series data set.

```
proc sort data= sasuser.Ibex2datetime_logTransNooutlier;
    out=two;
    by datetime;
run;
Proc Hpf data=two out=three lead=0;
    id datetime interval=Hour8. accumulate=Total;
    Forecast Los LogLOS visits Age;
run;
data sasuser.HpflogTransfnooutlier;
    set three;
    Los=Los/visits;
    LogLOS=LogLOS/visits;
    Age=round(Age/visits,1);
run;
Quit;
```

AUTOCORRELATION AND STATIONARITY DETECTION IN TIME SERIES.

AUTOCORRELATION

Let $\{X_t\}$ be a time series with a finite second moment, that is $E(X_t^2) < \infty$. The mean $E(X_t)$ is generally a function of t and is denoted as $\mu_t = E(X_t)$. The covariance between X_t and X_{t+k} is defined as $\gamma_X(k) = Cov(X_t, X_{t+k}) = E[(X_t - \mu_t)(X_{t+k} - \mu_{t+k})]$ for all integers t and k . This type of covariance is called an autocovariance. The time series $\{X_t\}$ is said to be stationary if $\mu_t = E(X_t)$ is independent of t ; that is, $\mu_t = E(X_t) = \mu$ for all t and $\gamma_X(k)$ is dependent only on k so that $\gamma_X(k) = \gamma_k$. For $k=0$, we obtain the variance of the time series $\{X_t\}$; that is, $Var(X_t) = \gamma_0$. The set of autocovariance values γ_k is called the autocovariance function (ACVF) at lag k . The set of standardized autocovariance coefficients, or autocorrelation

coefficients, $\rho_k = \frac{\gamma_k}{\gamma_0}$, constitutes the autocorrelation function (ACF). Let $\{x_1, x_2, \dots, x_N\}$ be N observations of a stationary time series $\{X_t\}$. The autocovariance and autocorrelation functions of the data are respectively approximated by

$$\hat{\gamma}_k = \frac{1}{N} \sum_{t=1}^{N-k} (x_{t+k} - \bar{x})(x_t - \bar{x}) \quad \text{and} \quad \hat{\rho}_k = \frac{\hat{\gamma}_k}{\hat{\gamma}_0} \quad \text{where } k = 0, 1, \dots, N.$$

In this case, the autocorrelation function (ACF) is the set $\{\rho_0 = 1, \rho_1, \rho_2, \dots, \rho_N\}$ of the autocorrelation coefficients at lags $0, 1, 2, \dots, N$. To detect for autocorrelation in the time series, we plot the autocorrelation function against the lag variable k . The graph of the ACF is called correlogram. Under the hypothesis of independence or non-correlation and for large N , the 95% confidence interval for the autocorrelation coefficients is approximately $\pm 1.96/\sqrt{N}$. Table 6 shows the SAS codes that were used to draw the correlogram of the data set used in this paper. For this data set, $N = 256$ and the 95% confidence interval is approximately ± 0.1225 .

Table 6: code for plotting the correlogram.

```

goptions reset=all ctext = bl htext = 3pct ftext = swissb border;
proc arima data=sasuser.HpflogTransfnooutlier;
  identify var=LogLos nlag=24 outcov=acf;
run; quit;
data acf1;
  set acf;
if lag=0 then up0=1;
else up=1.96/sqrt(n);
  lo=-up;
  run;
proc gplot data=acf1
axis1 order=-1 to 1 by 0.2 label=(a=90) length=4 in;
  axis2 order=0 to 25 by 5 length=6 in;
  plot (corr up lo)*lag /frame overlay vaxis=axis1 haxis=axis2 vref=0 cvref=red;
  symbol1 color=blue i=needle v=none w= 6 ;
  symbol2 color=red i=join v=none w=2; symbol3 color=red i=join v=none w=2;
  label corr='r(k)=Autocorrelation';
Title BOLD C=BL H=18pt FONT=swissb'Figure5. Correlogram for LogLos With 8 hours
Time interval';
Run;

```

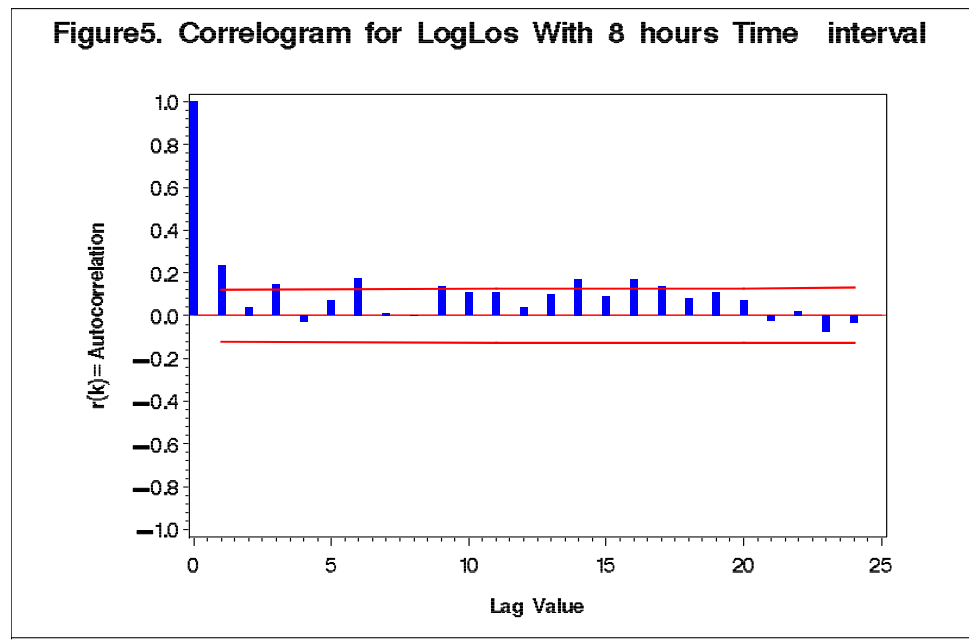


Fig.5. The correlogram shows a moderate positive autocorrelation. The autocorrelation is significant only at lags 1, 3, 6, 14 and 16. All other autocorrelation coefficients lie in the 95% confidence interval under the null hypothesis of no autocorrelation. For a size of 256 we should expected $12(=256*0.04)$ autocorrelation coefficients to be significant. Therefore, for the analysis, we can only consider the first autocorrelation to be significant and the others to have appeared by chance.

The error autocorrelation detection can be performed using the Durbin-Watson Statistic. This test is performed in SAS in the procedure Proc Autoreg, leaving empty the right side of the statement “model=” and requesting the Durbin-Watson statistic and the p-values by the options dw and dwprob. Lag means the time difference between observations. Any number of lags can be specified in the test. In the code of Table7, we have tested correlation up to lag 24 to cover a full day. The null hypothesis in the Durbin-Watson test is: “there is no autocorrelation in the errors.”

Table7: Code for testing autocorrelation by the Durbin-Watson Statistic

```
proc autoreg data=sasuser.HpflogTransfnooutlier ;
model logLOS = / dwprob dw=24;
run;
```

A part of the output of this code is shown in Table 8. The output consists of the Durbin-Watson Statistic value DW with the p-value to test if there is a correlation at lag k or not.

Table 8: Durbin-Watson Statistics

Order	DW	Pr < DW	Pr > DW
1	1.5181	<.0001	1.0000
2	1.9138	0.2657	0.7343
3	1.7167	0.0158	0.9842
4	2.0225	0.6439	0.3561
5	1.8079	0.0991	0.9009
...			

NOTE: Pr<DW is the p-value for testing positive autocorrelation,
and Pr>DW is the p-value for testing negative autocorrelation.

STATIONARITY

Time series analysis is about inference about the unknown structure (distribution) of the process $\{x_t\}$ using the available data (x_1, x_2, \dots, x_n) . The structure is then used in forecasting and controlling the future values of the processes. In order for this analysis to make sense, some kind of stationarity as defined in the previous section, is required. Stationarity in time series can be detected graphically, by computing and graphing the means for each time interval. The time interval in our data set is an hour. Table8 describes the codes used to calculate the means and standard deviation (SD) of LOS by hour. If the time series were stationary, we should expect the plots to be approximately on a horizontal line, meaning that Means and SD are not time dependent. Fig6a and Fig6b show the graph produced by this code. It clear from this graph that the series is not stationary since means are different across time.

Table8: Codes for graphic check of stationarity.

```

goptions reset = all ctext = bl htext = 4pct ftext = swissb border;
symbol1 i=spline v=dot c=red w=2 h=1.5;
symbol2 i=spline v=dot c=blue w=2 h=1.5;
legend1 across=1
      cborder=black
      position=(top inside left)
      value=(tick=1 'MEANS' tick=2 'STDs')
      offset=(3,-13)
      label=none;
proc means data=sasuser.HpfibexLogTransfnooutlier noprint;
class Time;
var LOS;
output out=LOSMeans1 mean=mean var=var std=std;
run;
axis1 minor=none label=( angle=90 'SDs and Means of Los')
order=(60 to 160 by 10);
axis2 minor=none label=('Time') ;
proc gplot data=LOSMeans1;
plot Mean*Time=1 std*Time=2 /vaxis=axis1 haxis=axis2 overlay
legend=legend1;
Title BOLD C=BL H=18pt FONT=swissb'Fig6a. STDs and Means Over Time : Accum.
Interval = 1 Hour';
run;
proc means data=sasuser.Hpf8ibexLogTransfnooutlier2 noprint;
class Time;
var LOS;
output out=LOSMeans2 mean=mean var=var std=std;
run;
axis1 minor=none label=( angle=90 ' SDs and Means of Los')
order=(30 to 150 by 10);
axis2 minor=none label=('Time') ;
proc gplot data=LOSMeans2;
plot Mean*Time=1 std*Time=2 /vaxis=axis1 haxis=axis2 overlay
legend=legend1;
Title BOLD C=BL H=18pt FONT=swissb'Fig6b. STDs and Means Over Time : Accum
Interval = 8 Hours';
run;

```

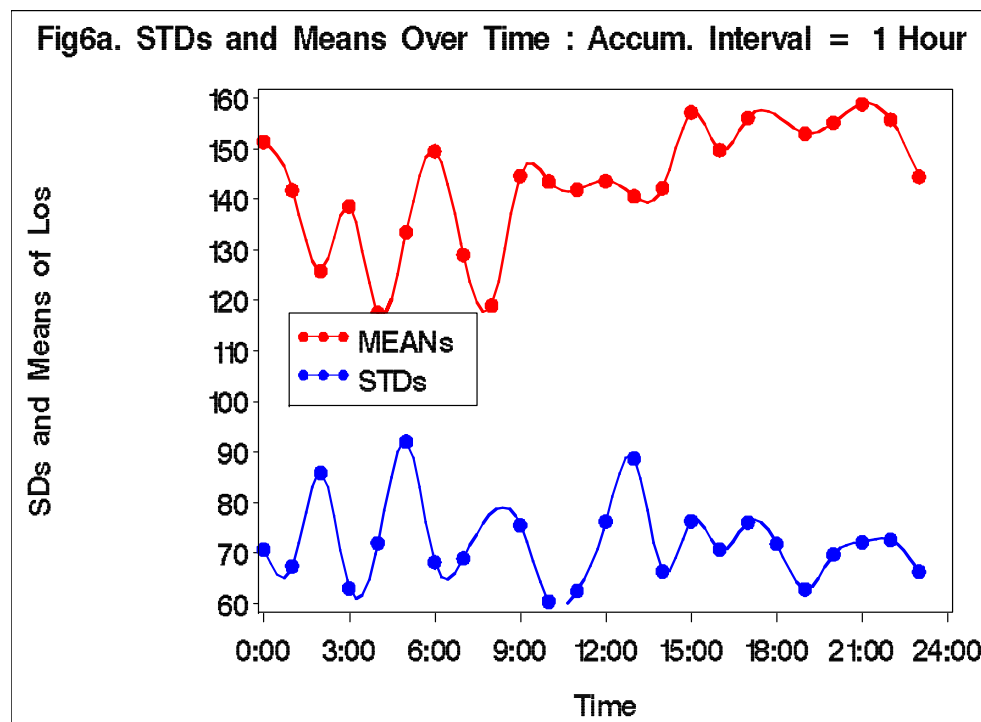
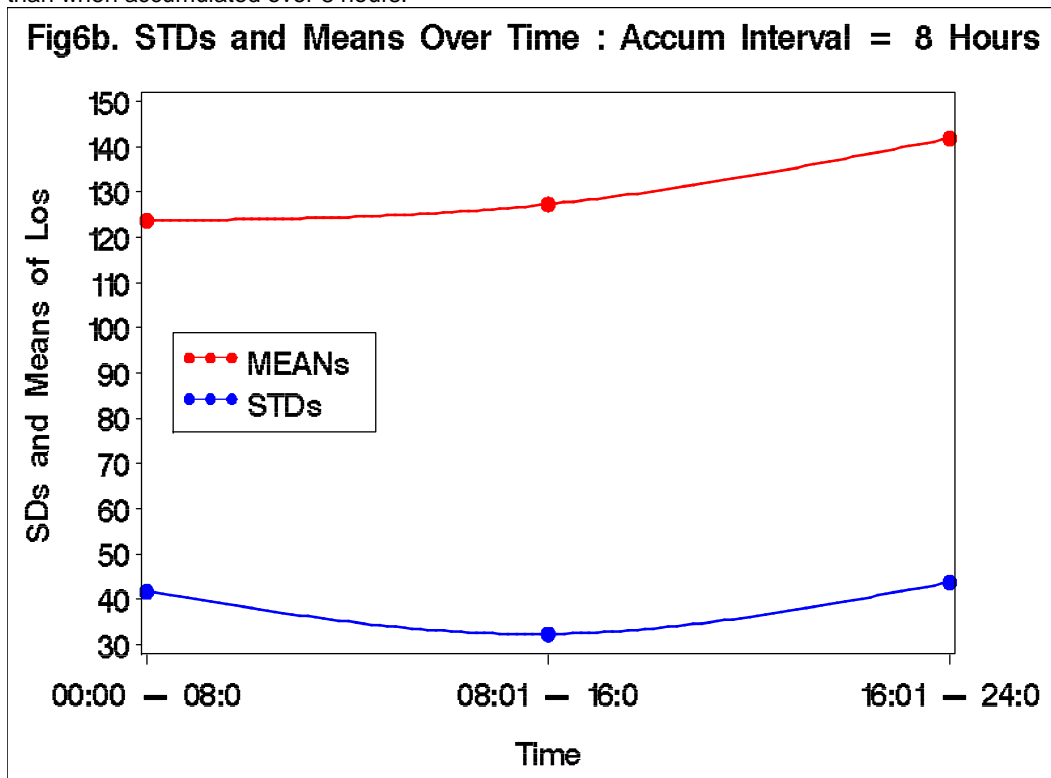


Fig6a and 6b. PLOT of Standard Deviations and Means Over Time. The means and STDS vary over time, which implies that the time series LOS is not stationary. When the data is accumulated over an hour, the variation is higher than when accumulated over 8 hours.



UNIT ROOT TESTS

Unit root tests are important statistic tests used in testing the stationarity of a time series. When a time series has a unit root, the series is nonstationary and the ordinary least squares (OLS) estimator is not normally distributed. In this paper we examine a unit root test for only an autoregressive and a moving average processes of order one. Given a

time series $\{X_t\}$, an autoregressive process of order one or AR(1) equation is given by: $X_t = \phi X_{t-1} + Z_t$ where $\{Z_t\} \sim WN(0, \sigma^2)$ and ϕ is an unknown constant. A unit root test would be a test of the null hypothesis that

$H_0: \phi = 1$, usually against the alternative hypothesis that $H_1: |\phi| < 1$. A First Order Moving Average or MA(1)

Process is defined by $X_t = Z_t + \theta Z_{t-1}, t = 0, \pm 1, \dots$ where $Z_t \sim WN(0, \sigma^2)$ and θ is a constant. A unit root test for MA(1) would be a test of the null hypothesis that $H_0: \theta = -1$, usually against the alternative hypothesis that

$H_1: |\theta| < 1$. For the two models, when H_0 is not rejected, the time series is considered nonstationary. It can be brought to stationary by using a differencing operator. When testing for stationarity, a higher order Autoregressive model can be specified.

Table 9: Dickey-Fuller unit and Phillips-Perron Unit Root Tests codes.

```
proc arima data=sasuser.Hpf8ibexLogTransfnooutlier2;
  identify var=LOS stationarity=( ADF=(1,2,5));
run; /*ADF=augmented Dickey-Fuller*/
proc arima data=sasuser.Hpf8ibexLogTransfnooutlier2;
  identify var=LOS stationarity=(PP=5);
run; /*PP= Phillips- Perron */
```

In Table 9, the first proc ARIMA performs Augmented Dickey-Fuller tests with autoregressive orders 1, 2 and 5 and the second procedure performs Augmented Phillips-Perron tests with autoregressive orders ranging from 0 to 5. Higher orders can be specified. Stationarity can also be checked using a boxplot.

TESTING FOR HETEROSCEDASTICITY

One of the key assumptions of the ordinary regression model is that the errors have the same variance throughout the sample. This is also called the homoscedasticity model. If the error variance is not constant, the data are said to

be heteroscedastic. Since ordinary least-squares regression assumes constant error variance, heteroscedasticity causes the OLS estimates to be inefficient. Models that take into account the changing variance can make more efficient use of the data. Therefore, it is recommended to check for homoscedasticity in the preprocessing stage. The test for heteroscedasticity with PROC AUTOREG is specified by the ARCHTEST option (Table 10).

Table10:SAS code for homoscedacity testing

```
proc autoreg data=sasuser.HpfibexLogTransfnooutlier;
  model LOS = / nlag=12 archtest dwprob;
  output out=r r=yresid;
run;
```

Table11: Output of Tests for heteroscedasticity for LOS with an accumulation interval of 8 Hours

Order	Q	Pr > Q	LM	Pr > LM
1	1.2376	0.2659	1.2043	0.2725
2	1.2376	0.5386	1.2101	0.5460
3	1.3224	0.7238	1.2833	0.7331
4	1.3421	0.8542	1.2920	0.8627
5	1.3437	0.9304	1.2921	0.9357
6	6.4119	0.3787	6.1978	0.4014
7	6.8140	0.4485	7.1051	0.4180
8	7.1834	0.5170	7.3418	0.5002
9	7.1873	0.6176	7.3893	0.5967
10	8.1111	0.6180	8.2836	0.6012
11	9.0873	0.6138	8.9393	0.6275
12	9.1121	0.6933	8.9478	0.7074

The Q and LM (Lagrange Multiplier) statistic tests (Table11) indicate that when the transactional variable is accumulated over an interval of 8 hours, the variable LOS does not show heteroscedascity at all lags.

CONCLUSION

Data preprocessing is a prerequisite step in data analysis. Data pre-processing can improve the quality of the data, for example by identifying and removing outliers, fixing errors for data mining, checking assumptions of any eligible data analysis method, and identifying the optimal model for a final analysis. In this paper, we showed how this important data preprocessing in data analysis can be handled by the SAS procedure. In data used for illustration of the process, we identified and removed outliers, identified a theoretical distribution of the data, and showed that linear models were not appropriate for the analysis of the data. By using a log transformation of the data, we obtained a distribution close to a normal distribution which is an important requirement in testing hypotheses. We also have found that observations in the data were autocorrelated. As we plan to analyze the data by an ARIMA (Autoregressive Integrated Moving Average) model, the next step will be to identify the parameters of the model. An

ARIMA Model is principally characterized by three parameters denoted p, d, q where q is the parameter of the moving average, d is the degree of differencing to transform the data to stationarity and p is the autoregressive parameter. After that, the final analysis will be done by the SAS procedure PROC ARIMA and compared to the procedure Proc HPF which does an automatic selection of the forecasting model.

REFERENCES

1. <http://support.sas.com/rnd/app/examples/ets/gplot/>
2. Updated Hospital Diversion Guidelines <http://www.njha.com/publications/pubcatalog/DiversionGuidelines.pdf>,
3. Hospital Status program, Emergency Health services : <http://hospitals.ehsf.org/ehsf.html> .
4. Chris Chatfield (2001) "Time-Series Forecasting"
5. Mohsen Pourahmadi (2001) "Foundation Of Time Series Analysis and Prediction Theory"
6. Peter J.Brockwell, Richard A.Davis (2002) "Introduction to Time Series and Forecasting"

CONTACT INFORMATION

Joseph Twagilimana
 Department of Mathematics
 University of Louisville
 Louisville, KY 40292
 502-852-6826
 j0twag01@louisville.edu