



**The Association of System
Performance Professionals**

The **Computer Measurement Group**, commonly called **CMG**, is a not for profit, worldwide organization of data processing professionals committed to the measurement and management of computer systems. CMG members are primarily concerned with performance evaluation of existing systems to maximize performance (eg. response time, throughput, etc.) and with capacity management where planned enhancements to existing systems or the design of new systems are evaluated to find the necessary resources required to provide adequate performance at a reasonable cost.

This paper was originally published in the Proceedings of the Computer Measurement Group's 2009 International Conference.

For more information on CMG please visit <http://www.cmq.org>

Copyright 2009 by The Computer Measurement Group, Inc. All Rights Reserved

Published by The Computer Measurement Group, Inc., a non-profit Illinois membership corporation. Permission to reprint in whole or in any part may be granted for educational and scientific purposes upon written application to the Editor, CMG Headquarters, 151 Fries Mill Road, Suite 104, Turnersville, NJ 08012. Permission is hereby granted to CMG members to reproduce this publication in whole or in part solely for internal distribution with the member's organization provided the copyright notice above is set forth in full text on the title page of each item reproduced. The ideas and concepts set forth in this publication are solely those of the respective authors, and not of CMG, and CMG does not endorse, guarantee or otherwise certify any such ideas or concepts in any application or usage. Printed in the United States of America.

Survival Analysis In Computer Performance Analysis

Brian Barnett Bank of America
Frank Berezny
Perry Gibson

Abstract

Survival Analysis is a statistical technique that deals with biological events and mechanical failure. It uses a nonparametric approach that handles continuously distributed and discrete data, detects significance, enables predictions, evaluates time to events (response time) and allows event censoring (equipment swaps). This paper will provide insight into how Cox Regression Analysis works and how it was implemented using SAS to identify critical components of computer performance and capacity problems. This methodology worked and improved performance in areas that were previously intractable. One example is provided.

Background

There are many ways to analyze computer performance and capacity problems. Among the methods used are rules of thumb, queueing theory, physical laws and statistics. Each method is useful depending on personal preference, background, experience and the type of problem at hand. Statistics is of great value when the requirements are to evaluate large amounts of data and to determine whether the measured changes are real or are most likely due to chance. In cases where the objective is to automate the analysis of large amounts of data so that alerts are generated when performance metrics exceed certain thresholds, then statistics is the best choice. Statistics can also be the best choice when large amounts of data are analyzed to enable discovery of causative and predictive factors.

Many of us took statistics classes and learned about descriptive statistics such as the mean, median, percentiles and standard deviation (σ). We also learned about normally distributed data: where in a normal distribution about 68% of the values lie within one standard deviation of the mean, approximately 95% of the values are within two standard deviations and 99.7% of the values are within three standard deviations. After studying the Central Limit Theorem, we understood that if we average measurements of some quantity, the distribution of our averages tends toward a normal one. No matter what the shape of the original distribution is, the sampling distribution of the means approaches a normal distribution. In most cases, a normal distribution is approached very quickly as the size of each sample mean increases. It also simplifies the approach to doing data analysis because so many statistical procedures require that the data be normally distributed. But what if the data is not normally distributed? Can you convert the captured data into a normal distribution? What do you do if you normalize the baseline (control) data but not the test data?

We started a project where the plan was to use Multivariate Adaptive Statistical Filtering (MASF¹) techniques to determine if response times had increased to a point where they were significantly elevated over their reference set. This project consisted of an SOA which uses AJAX (Asynchronous JavaScript and XML) extensively in the UI layer. There is an Oracle backend and usage is about 6k users per day with about 4k users online concurrently at any one time. It was here that we ran into large numbers of alert “surprises” because the data was not normally distributed. Numerous alerts were being generated even when the reference threshold was set to 3σ . In other cases, alerts were not elicited even though a threshold was set relatively low, such as 2σ .

Non-normally Distributed Data

Non-normally Distributed Data

When data is normally distributed, we have a good idea how the data values are distributed about the mean. When data is not normally distributed and we do not know the distribution, we cannot be so certain how the values are distributed.

When the data is not normally distributed, you cannot be certain about how many values you will find as you look further from the mean. How skewed the data is becomes very important (Figure 1). With a normal distribution, you know that at 2σ you will have accounted for 95% of all the data. With a non-normal distribution, you would account for **at least** 75% of the data, but you could have accounted for all of the data. While 99.7% of the values of a normal distribution are within 3σ of the mean, when the data is not normally distributed **at least** 89% of the values are within 3σ . As the degree of skewing goes up, the percentage of data within 3σ decreases closer to 89%. It is easy to see that with non-normal data, using a threshold of even three standard deviations could provide many surprises.

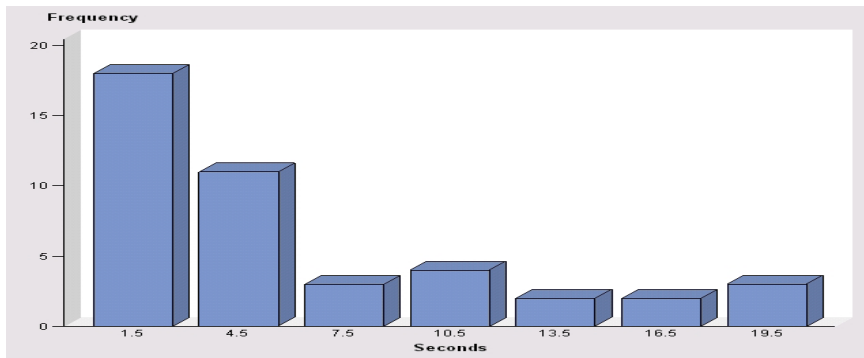


Figure 1
Highly Skewed Data Distribution

If the distribution of the population is normal, then the sampling distribution of the mean will be normal for any sample size N (including $N = 1$). If a population distribution is not normal, but has a bump in the middle and no outliers or a strong skew, then a sample of even modest size samples (e.g., $N = 30$) will have a sampling distribution of the mean that is very close to normal. However, if the population distribution is far from normal (see Figure 1 with extreme outliers and a strong skew), then to produce a sampling distribution of the mean that is close to normal it may be necessary to draw a very large sample (e.g., $N = 500$ or more). Your sampling period might not have enough data to normalize the mean. In our world, the data distribution was highly non-normal and we had to reevaluate the handling and analysis of our data.

Non-normality and Multivariate Adaptive Statistical Filtering (MASF)

The first place that we reevaluated the underlying data distributions was in applying the MASF¹ technique. We normalized a reference set by taking the average of response times (for every service) every five minutes over a four-hour period. The combined average for the 48 samples ($12 \times 4 = 48$ samples/day) became the basis of the threshold. Converting the average into a threshold was done by multiplying the mean by different multiples of the standard deviation. When the baseline was three standard deviations above the mean, we found that some services were still getting many high response time alerts. This continued even after raising the threshold level to 4σ , 5σ or even 7σ above the mean. The data was coming from a population that was highly skewed and the normalization process did not have a large enough sample to get a normalized mean distribution. It was at this point that we realized that we had to reevaluate how we handled non-normal data.

The Assumption of Normality

The assumption that data is normally distributed when it is not can produce erroneous conclusions. Techniques for handling normal data distributions (parametric statistics) do not apply to non-normal distributions. Those distributions that are not normally distributed are handled by a separate set of nonparametric statistical techniques. Many people assume that the data they are dealing with is normally distributed, but this assumption can result in erroneous conclusions when a parametric statistical test is applied to non-normal data.

Why should we assume that any data we come across is normally distributed? For example, Figure 1 is bounded by zero on the low side and has no bound on the high side. Why is there a long tail skewed to the right? Is there a way we can find out what is driving the elongation of these response times? Are all requests in a class of transactions calling for similar data to be returned? Could the elongation be due to how users interact with the system? Behavior is not random and consequently does not always produce output that is normally distributed; therefore, there is no reason to assume that data is normally distributed. Blindly making the assumption that our data is normally distributed, or close to normally distributed, can lead to some undesirable results, with perhaps the worst consequence being that others will not trust the conclusions and predictions we make. We need to

¹ Buzen, J. and Shum, A., "MASF –Multivariate Adaptive Statistical Filtering," Proceedings of the Computer Measurement Group, 1995

make sure that the sampling distribution applies to the methodology used for analysis. It would also be of benefit to use techniques that do not require us knowing the underlying data distribution to help answer our questions.

Survival Analysis

One branch of statistics that does not require knowledge of the data distribution is Survival Analysis. The origin of survival analysis goes back to mortality tables from centuries ago; however, it was not until World War II that a new era of survival analysis emerged. This new era was stimulated by interest in reliability (or failure time) of military equipment.³ This topic is called reliability theory or reliability analysis in engineering, and duration analysis or duration modeling in economics or sociology. Survival Analysis is a statistical tool that has many potential uses in computer measurement.

Survival Analysis has traditionally dealt with topics such as, how long will cancer patients live when given a new treatment, the time for a drug addict to relapse, the time for a piece of equipment to fail, time to divorce and development of insurance company mortality models. The parameter that is common across these examples is that each has a time to an event: time to death, time to relapse, time for a server or car to fail, time from marriage to divorce or the mean time to some other failure. The time to an event (the dependent variable) is very often not normally distributed, so this alone would prevent us from using standard regression techniques like Multiple Regression Analysis.

Another problem that prevents us from using techniques that are more common is, depending on how the event times are clocked, it is possible to have a completion time and no start time or a start time with no end time. When looking at system availability or reliability, some equipment, servers or application, would have a time to failure and others would not. The important question and challenge is how do you handle this situation? Do you throw away information if there is no time to a failure? There is no way for this to be handled in traditional statistics, but fortunately, Survival Analysis does have a methodology to handle this and it is called censoring. Censoring is a way to keep observations with incomplete time to event data in the study. For example, in Cox regression, which is one type of Survival Analysis, the computation of the regression coefficients is based only on the uncensored cases, but all cases are used when estimating the baseline hazard. We will focus on Cox regression in this paper.

When the independent variables vary with time, Cox regression can handle them. An example would be to look at response time (a timed event) relative to CPU utilization, available memory, available network bandwidth, hour of day and day of week. The timed event depends on three quantitative variables and two qualitative variables. The variables that the timed event depends on are called covariates, or independent variables, and each could cause the response time to increase or decrease. The analyst would want to discover which covariate(s) caused how much change allowing for prediction of what would happen if the covariates values were altered in any way. The covariates can also vary with time (hour and day in this case) as well and we would want a model that would include the effect on response time as covariates varied by time. Cox regression can handle this type of analysis.

Computer Performance Evaluation (CPE) and Survival Analysis both have timed events: response and failure times. Why not use Survival Analysis to help us analyze CPE?

Basic Survival Analysis Concepts as Applied to CPE

Before proceeding further, let us look at some Survival Analysis terminology and concepts. Some of the concepts will take a bit of getting used to, but to use these newer methods requires some understanding. The information in this section amplifies what was introduced above.

Censoring

Survival Analysis allows data to be included in the model even though there may not be an associated timed event. The cause of or reason for the missing timed event value does not matter; the transaction with a missing timed event is marked as being censored. This is analogous to end effects where there is an end time without a start time or a start time with no end time. We do not want to throw away all of these observations and desire that they be considered with other data collected. How we define a test interval also affects the number of timed events. If you only looked at failures during the first 24 hours after a software release, some parts of the release could show no problems while others could fail, so the parts that had no problems would be censored since they had no end event to measure.

Under some circumstances, data could be censored even if it had an associated timed event. For example, if data is being returned from a database and the time returned is a timeout value, that datum should be censored. The timeout is actually a non-event. Including it could introduce errors into the model.

Study Time Interval

Sequential calendar or clock time is not used in Survival Analysis. Events are aligned to a single timeline from start to finish. What matters is the time from a defined starting point to when the event is experienced or censoring has occurred. If we examine response times from 12 PM to 1 PM (a well-defined interval) all of the event times are evaluated as if they started from the same starting point (time $t = 0$) in the interval no matter what the clock time was.

In drug trials, this is important. It may not be possible to have subjects all start a trial on the same day, week or month. If we want to see if response times have increased after a release we would use survival statistics to compare the before and after release response times as if they were run at the same starting time and over the same interval. Care has to be taken to ensure that no other changes occurred to the application or system that would introduce errors into the results.

Time Dependence

Survival Analysis also can account for time-dependent covariates. Some covariates are constant during the period of analysis, while others can, and do, change over time. For example, the number of CPUs a server has and physical memory are constant over time. However, the available memory can, and does change over time. With Survival Analysis, we can combine quantitative, categorical and time-dependent variables in the same model. This would be very difficult to do using classical methods.

Covariates

Covariates are the independent predictors. They can be categorical (race, sex, citizen, operating system, 32 vs. 64 bit) or continuous elements (e.g., age, weight, available memory, network bandwidth).

The Hazard Rate and Hazard Ratio

The hazard rate (also called the hazard function or hazard ratio) is what much of Survival Analysis is about. The hazard rate/ratio (HR) has these properties:

- It is always a positive number greater than or equal to zero.
- It is often thought of as a probability but it is not because it can be larger than 1.0.
- It can only be estimated.
- The hazard rate is applied to individuals not to populations, unless all individuals in the population are the same. This could be the case in CPE when there is a uniform workload.
- If you looked at the surviving population at a given time interval after the start of an evaluation, you would be seeing the *odds* of the event happening.
- You must know the units of the hazard rate (seconds, hours, days, months, years, etc) in order to interpret it².
- A simple way to think of the hazard ratio is in terms of a test group relative to a control group. For categorical variables, the covariate is set to one if the person, or thing, is a member of the test group and to zero if the person or thing is a member of the control group. The calculated HR is the ratio between the test and the control group; it tells you how much these groups are different.
- For time-invariant continuous covariates, when the covariate does not change over time (e.g., physical memory), the hazard ratio is the amount of change in the hazard of the event occurring for each unit change in the covariate³. With a quantitative covariate, $(HR-1)*100$ estimates the percentage change in the HR when there is a one unit change in the independent variable. For instance, a hazard ratio of 1.02 means that there is a 2% increase in the rate of the event occurring for a one unit increase in the covariate, controlling for other covariates in the model. A hazard ratio of 0.1 means that there is a reduction in the hazard of the test group of 90% relative to the reference group. If the covariate were CPU utilization, the unit increase would be 1% CPU utilization. A hazard ratio of 1.1 for the covariate age would mean that a one-year increase in age would be

² Sahai H, Kurshid A. Statistics in epidemiology: methods techniques and applications. CRC Press 1996.

³ Armitage P, Berry G. Statistical Methods in Medical Research (3rd edition). Blackwell 1994.

associated with a .1 (10%) increase in the hazard rate. A ten percent increase in CPU utilization, or a ten-year increase in age, would correspond to $1.02^{10} = 1.22 =$ equating to increasing the hazard rate by a factor of 1.22, or 22%. In this way, predictions can be made.

- For time-varying continuous covariates, the hazard ratio is the amount the hazard ratio changes given a unit change in the time-dependent function of the covariate.
- Qualitative covariates take a bit more effort in coding and interpreting. We will explain how they are coded later in the paper. The HR compares the value of the variable when it is set to one to the value of the variable when it is set to zero, holding all other variables constant. The HR provides the odds of a positive outcome compared to a negative outcome.
- Cox regression assumes 'proportional hazards': the hazard ratio will remain constant over time. The hazard rates can change over time, but not the ratio. If we compare a pre-performance fix (control) to a post performance fix (test), the ratios between these will remain constant over time.
- When the hazard ratio is not repeatable, for example in the case of death, take the reciprocal (i.e. 1/hazard ratio) to obtain the expected time until the event occurs. Assume a person has a hazard ratio of dying from a heart attack of 0.04 and the units are in years. You can expect that person to live $1/0.04$ or 25 years before dying from a heart attack, assuming that the hazard ratio remains constant. This is the same as saying the odds that the person will die within the next time interval is 1:25 or that there is a $1/26=3.85\%$ chance of dying within the next time interval.

Chi-square

The Chi-square (statistical significance) value is computed as a function of the log-likelihood for the model with all independent variables and the log-likelihood of the model in which all independent variables are forced to zero. The Chi-square is calculated as $\text{Chi-square} = (\text{Parameter estimate}/\text{standard error})^2$. If the Chi-square value meets the criteria for statistical significance, we reject the null hypothesis and assume that the independent variable(s) are significantly related to survival times.

Cox Regression

Cox regression⁴, also called proportional hazard regression, is a Survival Analysis method for finding the effects that one, or more, covariates have on the time it takes an event to happen. It does not require that you select a survival model, or probability distribution, to represent survival times because the model uses the underlying hazard rate rather than survival time. Some other methods require that you first select a model or probability distribution. No assumption about the shape of the hazard function is required. It is called a proportional hazard because if you have two different sets of observations, a test group and a control group for the same covariate, the ratio of the hazard functions for these two sets of observations do not change with time. The ratio remains constant over time even though the hazards may change.

The calculated coefficients, β , in a Cox regression relate to the hazard ratio. A negative coefficient indicates that the ratio will be lower. In terms of an outcome such as death or availability, this is desirable, because the reduced hazard means that the chances of surviving, or being available, longer are greater. If we are analyzing response times, the negative coefficient means that we will expect the transaction to take longer to complete, which is not so desirable. Positive coefficients produce a higher ratio which translates to a higher chance of dying sooner (this is bad) or a transaction completing faster (this is good). The hazard ratio is calculated as e^{β} .

Cox regression makes the process of evaluating time-dependent covariates easier. Some variables are fixed over time because their values cannot change. For example, installed physical memory and the number of CPUs a server has is the same for all separate transactions examined during the measurement interval. However, if a covariate can have different values for different transactions over time, then it is time-dependent. For the most part, Cox regression is easy to implement and analyze.

A real life example would be looking at failure time after a software release. If you did the release at night, you would have a low load on the system right after the release. As the day went on the load would increase (arrival rates, CPU utilization etc) and application failures would be more likely to occur. These covariates would be time dependent.

⁴ Cox, D.R. (1972), "Regression Models and Life Tables" (with discussion) Journal of the Royal Statistical Society, B34 187-220.

PROC PHREG

SAS implements Cox regression through PROC PHREG (pronounced p h reg). There are also other packages that implement Cox regression, such as Stata, Limdep and SPSS. While the author's preference is SAS, the principles are the same.

A basic PROC PHREG input and output is shown below, along with how they are related. A simple example of using PROC PHREG follows. The timed event is ResponseTime in this example. A variable that contains the censored values (Censored) is created. The value of this variable can be one or zero depending on how it is defined. In the examples below, a censored event has a value of zero. This is indicated in the model statement by Censored(0). If a 1 indicated censored, it would be written as Censored(1).

Covariates are entered on the right side of the model statement. Here there are three time-independent variables named s1, s2 and s3.

```
Proc PHREG data = test;
model ResponseTime*Censored(0) = s1 s2 s3 / TIES = EFRON;
run;
```

PROC PHREG Output

Model Information						
Data Set			test			
Dependent Variable	ResponseTime		ResponseTime			
Censoring Variable	censored					
Censoring Value(s)	0					
Ties Handling	EFRON					
Number of Observations Read			214110			
Number of Observations Used			214110			
Model Fit Statistics						
Criterion		Without Covariates		With Covariates		
-2 LOG L		4827871.4		4783687.2		
AIC		4827871.4		4783847.2		
SBC		4827871.4		4784669.2		
Testing Global Null Hypothesis: BETA=0						
Test		Chi-Square		DF	Pr > ChiSq	
Likelihood Ratio		44184.1687		80	<.0001	
Score		71950.1032		80	<.0001	
Wald		55882.4797		80	<.0001	
Analysis of Maximum Likelihood Estimates						
Parameter	DF	Parameter Estimate	Standard Error	Chi-Square	Hazard Pr > ChiSq	Ratio
β						
s1	1	-0.74455	0.03952	354.8785	<.0001	0.475
s2	1	-0.71369	0.02649	725.8828	<.0001	0.490
s3	1	-0.71978	0.04457	260.8166	<.0001	0.487

Figure 2
PROC PHREG - Cox Regression Output

Cox Regression Output Interpretation

Dependent Variable Names the dependent variable.
Censoring Variable Names the Censoring variable.
Censoring Values(s) The value of the censored variable.

Ties	If input data has ties PHREG handles this a TIES statement.. TIES=EFRON provides approximations to the EXACT method without using the high CPU that goes along with the EXACT method. If the time scale is discrete, use the TIES=DISCRETE method.
Testing Global Null Hypothesis:	BETA=0 The null hypothesis is that all the coefficients are zero. Since the Likelihood Ratio, score and Wald test have $Pr > ChiSq$ that is <0.05 , we can reject the null hypothesis and say that at least one of the 80 covariates is not 0. The Likelihood Ratio is obtained from the -2 LOG L by subtracting the With Covariates from the Without Covariates.
Parameter	Covariate (independent variable), predictor variable (s1, s2, s3 in this example).
Number of Observations	The total number of observations in the input. Each covariate should have a minimum of 100 observations.
Model Fit Statistics	see Testing Global Null Hypothesis
DF	Degrees of Freedom. Each variable has 1 DF. The total, in this case is 80.
Parameter Estimate	β is the regression coefficient. The sign of the coefficient tells the direction the relationship takes. A positive sign indicates shorter times to the timed event. A negative coefficient indicates a longer time to the timed event. For life and death studies, a negative coefficient is good. For response times a positive coefficient means shorter response times.
Standard Error	Standard deviation of those sample means over all possible samples.
Chi-square	goodness-of-fit $(\beta/\text{Standard Error})^2$
Pr > ChiSq	If the result is significant, the null hypothesis is rejected and the covariate is related to survival time
Hazard Ratio	The hazard ratio is e^β .

Covariates that we think are related to the timed event are added to the PHREG model statement. In the past, we were always told to treat all variables as independent, so this is a change from how we are used to processing data. This allows us to add covariates that we believe, based on experience, are predictors of the event time and to add interactions between covariates. An interaction between two variables, s1 and s2, is written as s1*s2. PROC PHREG would evaluate this in the following manner: hold all covariates constant and compare s1*s2 when both s1 and s2 are one to when both s1 and s2 are zero. There can be any number of interactions, e.g. s1*s2*s3. We have to be careful in defining the time interval we use to test. The interval needs to have a well-defined start and end time. Measurements that start or stop in another interval should be censored. Units have to be consistent. If one variable is measured in days and another in hours, results will be incorrect. In our next example there are two continuous quantitative variables named AvailableMemory and PagesPerSecond that we will handle as time-independent since data was collected over a short time interval. The form of PROC PHREG SAS code is:

```
Proc PHREG data = work.cpu_survival;
model ResponseTime * Censored(0) = AvailableMemory PagesPerSec ;
run;
```

PROC PHREG Output

Below is a part of the essential output produced by PROC PHREG.

Parameter	DF	Parameter Estimate	Standard Error	Chi-Square	Pr > ChiSq	Hazard Ratio	Label
		β					
AvailableMemory	1	0.00181	0.0002936	38.0736	<.0001	1.002	AvailableMemory
PagesPerSec	1	-0.12184	0.04587	7.0544	0.0079	0.885	PagesPerSec

Figure 3
PROC PHREG - Cox Regression Output 2

Using what we have discussed so far, we can see that AvailableMemory has a positive parameter estimate, which produces an increase in the Hazard ratio. This in turn translates to a faster expected response time. In our

previous example, the hazard of dying was referenced, but here the hazard is that of a response time ending, meaning a faster response. The PagesPerSec has a negative Parameter Estimate, which results in a smaller Hazard ratio and an increase in transition page time. The parameter estimate, β , when raised to the e^β gives the Hazard ratio. The $Pr > ChiSq$ indicates that the Hazard ratio is not due to chance alone. For the quantitative variable, AvailableMemory, we get $(1.002-1)*100=0.2$ meaning we will get a 0.2 percent change in the Hazard ratio for each unit change in AvailableMemory. For PagesPerSec we have $(0.885-1)*100 = -11.5$, meaning each unit increase in PagesPerSec will decrease the hazard by 11.5% (increase ResponseTime).

A Real Example

A portion of a system showed increased response times that were traced to Oracle page transitions. Traditional analytical methods did not elicit enough information to throw light on what the problem was. The initial analysis resulted in identifying the categorical variables shown in Table 1: Division, Source, Target, OrgUnit, OrgUnitSubType and Region along with the continuous variable %ProcessorTime.

Our goal was to find the variables, or combinations of variables, that have the greatest negative Parameter Estimates. Combining variables blindly is not useful in these cases. PHREG allows us to model either single variables or combinations of variables (called interactions). Consider the 64 (n) categorical variables previously identified. If we just looked at potential combinations of these variables taken four (r) at a time, we would have $(N!/(R!(N-R)!))=635,376$ potential model candidates with no repetition allowed and where order is not important. In these cases, PHREG holds all variables in the Model statement constant except the one it is evaluating. This is also true for interaction covariates where the evaluation is done with all the covariates in the interaction set being evaluated set to true while all other covariates are held constant.

This model selection process is where the CPE analyst should use expertise to construct a reasonable model grouping. Stepwise selection is available in Proc PHREG but it can produce errors⁵. We do not need to automate variable selection even though there are a large number of potential covariates. Initially, we included one continuous variable (% ProcessorTime) with six categorical variables. Cox regression requires that all the predictors be numeric and the qualitative variable values must be zero or one. Variables that could take only two values, such as 64 bit processor, could be coded 1 for 64 bit and 0 for 32 bit. When the variable can take on more than two values, we have to create dummy variables. For example, the variable 'Source' can have dummy variable names of Source1 to Source13 each with a possible value of 0 or 1 Another variable, 'Target', would also have dummy variable names of 1 to 13 for it as well. Additionally, we can create dummy variables for OrgUnit and OrgUnitType1. We will leave Division, Region and Branch out of this part of the study. One other thing that we can do is group covariates to get interactions between the covariates. The process to arrive at a combination grouping would be:

What is the impact of Source (From page) on page transition time?

What is the impact of Target (To page) on page transition time?

What is the impact of Source (From page) and Target (To page) on page transition time?

What is the impact of OrgUnit on page transition time?

What is the impact of Source (From page), Target (To page) and OrgUnit on page transition time?

What is the impact of OrgUnitSubType on page transition time?

What is the impact of Source (From page), Target (To page), OrgUnit and OrgUnitSubType all taken together at the same time on page transition time?

Combination 7 can have 249,900 combinations. However, preliminary analysis revealed that if we eliminated all cases where there were less than 100 observations, the number of useful combinations dropped to 89.

We have the following variables:

DIV1-DIV2

Source1 – Source13 (the 13 From pages)

Targ1 – Targ13 (the 13 To pages)

OrgUnit1 – OrgUnit8

OrgunitSubType1- OrgunitSubType17

Region1 – Region11

s1 to s89 (Interaction covariates)

⁵ Ernest S. Shtatland, Harvard Medical School, Harvard Pilgrim Health Care, Boston, MA

Ken Kleinman, Harvard Medical School, Harvard Pilgrim Health Care, Boston, MA

Emily M. Cain, Harvard Medical School, Harvard Pilgrim Health Care, Boston, MA "Model Building In PROC

PHREG With Automatic Variable Selection and Information Criteria" SUGI 30, 2006-30.

Proc PHREG allows you to create variables as if it were a SAS data step (see Proc PHREG below where only a portion of the created variables is displayed). The variables that have been created are entered into the Model statement. Variables s1 to s89 are interactions between Source, Target, OrgUnit and OrgUnitSubType.

Table 1 CATEGORICAL VARIABLES

Count	Division	Source and Target	OrgUnit	OrgUnitSubType	Region	Branch
1	Div1	Choose Options	BR	ABA/Joint Ventures	NewEngland	1
2	Div2	ErrorView	DIV	B2B Sales	MiddleAtlantic	2
3		Input Assets	BRST	B2C Administration	EastNorthCentral	3
4		Input Borrower Info	CO	B2C Centralized Originations	WestNorthCentral	4
5		Input Declarations	RGN	B2C Centralized Processing	SouthAtlantic	5
6		Input Employment/Income	DEPT	CMD Corporate	EastSouthCentral	6
7		Input FHA/VA Addendum	TMDV	CMD Relation	WestSouthCentral	7
8		Input Government Info	TMBR	Central Funding	Mountain	8
9		Input Liabilities		Central Underwriting	Pacific	9
10		Input REO Details		NSC	Hawaii	10
11		Input Subject Property		One Time Close	Alaska	11
12		Next Steps		Production Branch		12
13		Validate Application		Production Only Branch		13
14				Regional Operations Center		14
15				Remote Sales Team		15
16				Standard Branch		16
17				Unknown		17
18					
19						N

Our initial model used 155 categorical variables and one continuous variable, but obviously not all of the variables are significant. We need to determine which variables are to be included in the model and if we can obtain any insights from the model. The initial process starts by running PROC PHREG and looking for all variables where $P > \chi^2 > 0.05$. These variables are removed from the model. PROC PHREG is then rerun and any non-significant variables are once again removed. This process continues until all remaining variables are significant.

The last step is to pick the best subset of all the variables. The criterion used to determine this subset is based on the global score chi-square statistic. For two models A and B, each having the same number of explanatory variables, model A is considered to be better than model B if the global score chi-square statistic for A exceeds that for B⁶.

```
Proc PHREG data = eclib001.New&date._3;
model trantime*Censored(0) = ProcessorTime DIV1 DIV2
REG1 REG2 REG3 REG4 REG5 REG6 REG7 REG8 REG9 REG10 REG11
Source1 Source2 Source3 Source4 Source5 Source6 Source7 Source9 Source10 Source11 Source12 Source13
Target1 Target2 Target3 Target4 Target5 Target6 Target7 Target8 Target9 Target10 Target11 Target12 Target13
orgun1 orgun2 orgun3 orgun4 orgun5 orgun6 orgun7 orgun8 orgun9 orgun10 orgsub1 orgsub2 orgsub3 orgsub4
orgsub5 orgsub6 orgsub7 orgsub8 orgsub9 orgsub10 orgsub11 orgsub12 orgsub13 orgsub14 orgsub15 orgsub16
```

⁶ Furnival, G. M. and Wilson, R. W. (1974), "Regression by Leaps and Bounds," *Technometrics*, 16, 499–511.

```

orgsub17 s1 s2 s3 s4 s5 s6 s7 s8 s9 s10 s11 s12 s13 s14 s15 s16 s17 s18 s19 s20 s21 s22 s23 s24 s25 s26 s27
s28 s29 s30 s31 s32 s33 s34 s35 s36 s37 s38 s39 s40 s41 s42 s43 s44 s45 s46 s47 s48 s49 s50 s51 s52 s53
s54 s55 s56 s57 s58 s59 s60 s61 s62 s63 s64 s65 s66 s67 s68 s69 s70 s71 s72 s73 s74 s75 s76 s77 s78 s79
s80 s81 s82 s83 s84 s85 s86 s87 s88 s89 /TIES=EFRON;

```

/*examples of creating dummy variables in PHREG*/

```

Source1=(Source=1); Source2=(Source=2); Target1=(Target=1);
Target2=( Target =2); Orgun1=(OrgUnit=1); Orgun2=(OrgUnit=2);
S1=Source1*Target4*OrgUn1*OrgSub13; S2=Source1*Target4*OrgUn1*OrgSub16;
Run;

```

Output from the first pass PHREG

Parameter	DF	Parameter Estimate	Standard Error	Chi-Square	Pr > ChiSq	Hazard Ratio	Label
ProcessorTime	1	-0.00101	0.0000595	289.9069	<.0001	0.999	
ProcessorTime							
DIV1	1	-0.40378	0.19667	4.2153	0.0401	0.668	
DIV2	1	-1.81729	0.77847	5.4496	0.0196	0.162	
...							
Reg10	1	0.06043	0.16255	0.1382	0.7101	1.062	
Source2	1	0.21445	0.13580	2.4937	0.1143	1.239	ErrorView

Figure 4
PROC PHREG - Cox Regression Output 3

Table 2 Final Model

Parameter	Parameter Estimate	Standard Error	Chi-Square	Pr > ChiSq	Hazard Ratio	Label
Target9	-2.29593	0.02226	10636.610	<.0001	0.101	Input Liabilities
Target10	-1.98572	0.02060	9288.757	<.0001	0.137	Input REO Details
Target11	-2.00305	0.02097	9124.893	<.0001	0.135	Input Subject Property
Target6	-1.88543	0.01984	9035.396	<.0001	0.152	Input Employment/Incom
Target4	-2.03023	0.02195	8555.058	<.0001	0.131	Input Borrower Info
Target3	-1.34359	0.02078	4181.848	<.0001	0.261	Input Assets
Target7	-1.89767	0.03090	3771.013	<.0001	0.150	Input FHA/VA Addendum
Target8	-1.50254	0.02819	2841.865	<.0001	0.223	Input Government Info
Target5	-0.93833	0.02107	1983.176	<.0001	0.391	Input Declarations
Source4	-0.63418	0.01965	1041.217	<.0001	0.530	Input Borrower Info

Results

The output from this analysis clearly shows how Cox regression analyzed a large group of variables and identified a smaller subset of variables that had the greatest negative impact on response time. Variables with a non-significant Chi Square were eliminated and other variables were ordered according to their effect on page transition times. Decreased response times in production were attained because of a better understanding of what was producing the elongated responses times. This validated the usefulness of Cox regression: it worked.

1. We went from 64 variables in Table 1 to a model with just ten statistically significant covariates (Table 2). Improving the performance of these covariates would give the biggest-bang-for the buck in terms of improving overall page transition times.
2. The magnitude of the Chi Square metric is a measure of how big the relative effect is when comparing the same number of variables. Arranging the covariates in decreasing order of Chi Square arranges the covariates in decreasing order of impact. This comparison and arrangement can be done with individual

covariates as well as with interaction groupings. The interaction groupings can be compared to groups with equal numbers of members.

3. Nine Target (to) pages have the largest impacts in increasing page transition times irrespective of anything else. Here is where page transition times can be improved the most by loading stable data into cache after the first time a page is called.
4. Only one source page (sourc4) displayed a reduction in the hazard ratio but this was less than the effect of the target pages.
5. For Region 10 (Hawaii) we expected to see a negative parameter estimate (longer response times) because that region is the furthest from our data center. The parameter estimate was positive and the $P > \text{Chi Sq}$ was not significant (Figure 4). Post hoc analysis confirmed that Region 10 should have been removed from the analysis. While Hawaii is about 3,700 miles removed from the database, it also had significantly less traffic than other regions. If the analyst knows from experience that the covariate in question does affect event time, the covariate can still be left in the model. It does no harm.
6. Creating a dummy covariate that would be 1 for post-tuning changes and 0 for the original data would create a simple test for determining the overall effectiveness of any tuning changes. The HR for the dummy variable in this case would indicate how much of an impact caching had on the overall page transition times. The individual HRs would show how much each covariate was impacted by caching. After tuning changes were made to speed up Target page loading, Target pages no longer had the most individual negative impact on response times. The biggest impact became the four covariate interactions
7. One point should be added to the analysis. If you repeat the analysis with a different input data, assuming no changes that would affect performance, the results can be expected to be equivalent but not the same. If a change was made that could improve performance, you would look for a change in the HR.

Survival Analysis Summary

We have looked at problems that non-normal and unknown data distributions can cause an analyst. From this, we looked at a way to solve this type of problem using Survival Analysis, and in particular, Cox regression. The primary reason for adopting Cox regression as a tool is that it is not necessary to know what the underlying distribution is. Data without a defined end time can be included in the model along with time invariant and time-varying covariates. Additionally, large numbers of covariates can be used and the methodology will eliminate non-significant covariates. The analyst has the capability to add covariates that are thought, or are suspected, to impact a time to an event. If any of these covariates not to be significant, they can be removed from the Cox regression model. The basic concepts behind Cox regression were explained along with a real example showing that the methodology works. Survival Analysis is in fact a statistical tool that has many potential uses in computer measurement analysis.
