8-16-1996

A Decision Support System for Building ARIMA Models: Using A Relational Database As A Knowledge Base

Peter Chung Chang Luan
Computer Support Services, College of Business, Arizona State University

Robert D. St. Louis
School of Accountancy, College of Business, Arizona State University

A Decision Support System for Building ARIMA Models: Using A Relational Database As A Knowledge Base

Peter Chung Chang Luan, Computer Support Services, College of Business, Arizona State University
Robert D. St. Louis, School of Accountancy, College of Business, Arizona State University

I. Introduction

Many efforts have been made to combine knowledge with traditional modeling tasks. This study focuses on the task of model building. Model building is an iterative, trial and error process that consists of formulating and evaluating alternative models. The tentative model allows the problem at hand to be subjected to further analysis, manipulation, and interpretation. Obtaining a good model requires an understanding of the basic problem, familiarity with relevant underlying theory, and awareness of the subject domain. It also requires proper application of both model knowledge and domain knowledge to all sub-tasks. These requirements demand intensive knowledge processing. Many tools are available to build a knowledge based system (KBS) to support model building. It is natural to adopt some tools from artificial intelligence, and it also is natural to adopt database tools.

Using relational database (RDB) technology to build a KBS has many advantages. Relational database management systems (RDBMSs) are commercially available. In the database field, there is a trend to replace other types of database management systems with RDBMSs. Moreover, the RDBMSs are the primary database management systems for client-server computing environments. Using a RDBMS to establish a "data warehouse" [Inmon, 1993] on a server with structured query language (SQL) as an access tool on a client, allows users to apply graphical user interfaces (GUI) to access data, resulting in competitive business advantages [SAS Institute Inc., 1994].

Another reason why RDBMSs are so popular is they enable a database administrator (DBA) to keep the data consistent and accurate. By keeping the relations in conformance with certain rules, DBA can reasonably safeguard against duplicate records, accidental deletions, and inconsistencies during updates.

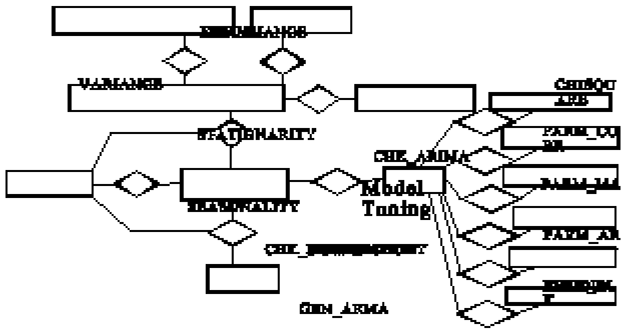
If we use a RDBMS for constructing knowledge bases (KB), several important advantages will be obtained. First, model building knowledge will be stored in a data management system as a warehouse. This means the knowledge will be available to all modelers, regardless of the statistical or operations research software that is being used. Secondly, it's more likely that model building knowledge will be consistent and easily updated. Third, client-server technology is readily available to provide widespread access to the knowledge through distributed processing. Fourth, the infrastructure may be simple enough to allow problem domain experts to encode their knowledge directly into the KB with little or no assistance.

For illustrative purposes, knowledge about building Autoregressive Integrated Moving Average (ARIMA) time series models are acquired. Using this set of knowledge and a KBS proposed [Luan, 1995], a prototypical application is developed. The ARIMA procedure, developed by Box and Jenkins (1976), analyzes and forecasts univariate time series data, transfer function data, or intervention data. It is a procedure that is well developed in theory and well accepted in practice.

2. Representing the KnowledgeRepresenting the Knowledge

Based on the entity-relationship (ER) data model developed in the [Luan, 1995], model building knowledge can be represented using relational tables. Overall, the structure of the knowledge set may be graphically represented by the ER diagram in Figure 1 showing all entities and relationships in building an ARIMA model. Basic statistical concepts such as stationarity, variance, and means are represented as entities. Association between concepts are represented as joint relationships. In Figure 1, through four joint relationships CHK_STATIONARITY, CHK_SEASONALITY, CHK_ARMA, and CHK_MODEL, we are able to associate the concept CHK_ARIMA with four sub-concepts STATIONARITY, SEASONALITY, ARMA and MODEL_TUNING. Table 1 is a partial schema description of entities and relationships in Figure 1.

The status of sub-concepts determine the row status of the major concept such as CHK_ARIMA. The row status of each concept then determines proper actions to take if the current concept is not optimum. For example, when the status code for R_STA in CHK_ARIMA is "N", regardless the status code for R_ARMA, R_SEASON, or R_MODEL in concept 3, user is asked to obtain stationarity of the data series and plot the data first.



Sub-concepts may associate themselves with other sub-concepts. For example: concept STATIONARITY is associated with three other concepts by the relationships CHK_VARIANCE, CHK_MEANS, and CHK_COVARIANCE. The concept STATIONARITY captures patterns of status values of the sub-concepts variance, mean, and autocorrelation function (ACF), which is essentially the possible situations in identifying true stationarity.

The links among concepts are achieved by absorbing sub-concepts as attributes as the primary key. The primary key for each concept consists a combination of conditions that lead to an action. That is, entities are condition-action tables. Nonkey attributes either suggest further actions or initiate certain procedures to revise current concept.

Table 1: Concepts List

Concept Number	Concept Name	Attribute List
1	Model directory	(Entity_Name, Number_of _Condition_Attributes, Number_of_Action_Attributes, Description)
2	GOOD_ ARIMA	(C_ARIMA, A_ARIMA, F_STATUS)
3	CHK_	(R_STA, R_ARMA, R_ SEASON, R_MODEL,

	ARIMA	C_ARIMA)
4	CHK_STATIONARITY	(C_STA, A_STA, R_STA)
5	STATIONARITY	(R_VAR, R_MEANS, R_ACF, C_STA)
6	VARIANCE	(C_VAR, A_VAR, R_VAR)
7	VAR_P	(PATTERN_VARIANCE, C_VAR)
8	GEN_ ARMA	(ACF_EXP, ACF_DAMP, ACF_SPIKES, PACF_EXP, PACF_DAMP, PACF_SPIKES, C_ARMA)
9	R_ARMA	(C_ARMA, A_ARMA, R_ARMA)
10	SEASONALITY	(R_STA, R_ARMA, R_ SEASON, A_SEASON)
11	MODEL_ TUNING	(Conf_ Int, Parm_ Corr, Parm_ AR, Chisquare, Parm_ MA, Residual, C_MODEL)
12	R_MODEL	(C_MODEL, A_MODEL, R_MODEL)
13	Concept Label	(Concept_Name, Attribute_Name, Attribute_label, TYPE, Order)

3. Associating ARIMA Modeling with KB

Table 1 developed in the previous section may be useful in the process of building a ARIMA model. A user may start with model directory table, concept 1. It provides a list of all the models in the KB; their names, number of condition attributes and number of action attributes in the top concept, and their general description. A user can start with a concept that has "forecasting" in its general description field.

Given a series like GDP (Gross Domestic Product), F_STATUS in concept 2, GOOD_ARIMA, is determined by C_ARIMA. C_ARIMA in turns is determined in concept 3, CHK_ARIMA. The first attribute of concept 3, R_STA, concerns whether GDP is a stationary series or not.

For checking stationarity, row status of R_STA in concept 4, CHK_STATIONARITY, is determined by C_STA. C_STA is determined by four attributes in concept 5, STATIONARITY. The first attribute of concept 5, R_VAR, is determined by concept 6 while second attribute R_MEANS depends on another concept.

The GDP data plot shows no apparent variance pattern and an upward trend of the means after year 1970's. From concept 5, then it seems appropriate to adjust the mean by taking a difference of the data series. After taking the difference, the data still fanning out which means the variance still has some kind of patterns. According to concept 6, since the standard deviation proportional to the expected value of the data series, then a logarithm is recommended to remove the nonstationarity in the variance. Since the preference is to remove the nonstationarity in the variances first, so we apply the logarithm first and then difference the data. After taking these transformations, the series seems to be stationary.

Since there is no seasonal pattern showing in the ACF and the ACF plot is showing some types of exponential decay within the first 10 lags (ACF_EXP=1), according concept 8, there is no need to adjust the seasonal part of the model. According to concept 9, concept 10 and concept 3, we can then conclude that we have a stationary series.

After the data series is stationary, we can then determine the ARMA model parameters. The plot of the PACF has one significant spike, PACF_SPIKES=1 and then becomes stable after. According to concept 8, model AR(1) is recommended, and R_ARMA="Y." From concept 11, MODEL_TUNING, the estimate of AR(1) model seems to have satisfactory result. The T rations of the estimate coefficient are significant. There is low correlation (Parm_Corr) among the parameter estimates. With respect to the autocorrelation, since most of the associated probabilities are higher than 0.05, there is no significant pattern in the residuals (Residual). The coefficient of AR factors (Parm_AR) is not close to one. From the status of all these attributes, according to concept 11, row number 7 tells us that C_MODEL=5 which in concept 11 concludes that there is no need to adjust the model any further, "Exit."

4. Summary and Conclusions

Based on the ER data model developed, we represented the knowledge for building ARIMA models using a set of relational tables. We showed the feasibility of how we may apply these tables to build a ARIMA model for a sample data series. All the manipulations of concepts shown above have been implemented in SQL.

In addition to storing model building knowledge, concepts in the KB provide several important functions. "Model directory" concept and "Attribute label" concept serves as a dictionary which describes the organization of the knowledge set. This includes information such as KB names, number of attributes in each KB, name of each attribute, and attribute type.

Concepts serve as control points. We can use certain concepts to determine the type of processing to be taken place. Each concept has the actual value or information of each attribute in the concept. Depending on the value status of an attribute, we may associate it with different types of processing, either with a procedure or associate it with a process.

Concepts serve as points of integration. Information from statistical computation procedures or other devices can be integrated or monitored by each concept to constantly adjust the data series or the descriptive model until reaching an optimal or satisfactory result. More sophisticated visual aids may be used in the future. For linking proper knowledge among tables, an algorithm were developed to deal with the iterative process of retrieving knowledge from these tables.

Note: References available upon request from Peter C. C. Luan.