Principles of Forecasting: A Handbook for Researchers and Practitioners,
J. Scott Armstrong (ed.): Norwell, MA: Kluwer Academic Publishers, 2001

DIFFUSION OF FORECASTING PRINCIPLES: AN ASSESSMENT OF FORECASTING SOFTWARE PROGRAMS

Len Tashman* and Jim Hoover**
*School of Business Administration, University of Vermont
Burlington, VT 05405
**United States Department of the Navy
2000 Navy Pentagon (N412H)
Washington, D.C. 20350-2000

ABSTRACT

Do forecasting software programs facilitate good practices in the selection, evaluation, and presentation of appropriate forecasting methods? Using representative programs from each of four market categories, we evaluate the effectiveness of forecasting software in implementing relevant principles of forecasting. The categories are (1) Spreadsheet add-ins, (2) Forecasting modules of general statistical programs, (3) Neural network programs, and (4) Dedicated business-forecasting programs. We omitted one important category – Forecasting engines for demand planning software - because software developers in that market refused to submit their products for review.

In the aggregate, forecasting software is attending to about 50 percent of the basic principles of forecasting. The steepest shortfall occurs in assessment of uncertainty: programs are often secretive about how they calculate prediction intervals, and uninformative about the sources of uncertainty in the forecasts. For the remaining areas of evaluation – preparing data, selecting and implementing methods, evaluating forecast accuracy, and presenting forecasts – we rated the packages as achieving 42-51 percent of the maximum possible ratings (the ratings assigned for best practices).

Spreadsheet add-ins (16% of best-practices rating) have made rudimentary regression tools and some extrapolative forecasting techniques accessible to the spreadsheet analyst; however, they do not incorporate best practices in data preparation, method selection, forecast accuracy evaluation, or presentation of forecasts.

Forecasting modules of general statistical programs (42% of best-practices rating) provide effective data preparation tools; however, with the exception of one of these programs, they do not adequately help users to select, evaluate and present a forecasting method. To implement best practices, the forecaster must perform macro programming and multiple-step processing.

Neural network packages (38% of best-practices rating) facilitate many best practices in preparing data for modeling and in evaluating neural network models. They do not use the more traditional models as comparative benchmarks, however, to test whether the neural net improves accuracy enough to justify its added complexity and lack of transparency.

Dedicated business-forecasting programs (60% of best-practices rating) have the best record for implementation of forecasting principles. Data preparation is generally good, although it could be more effectively automated. The programs are strong in method selection, implementation, and evaluation. However, they lack transparency in their assessments of uncertainty and offer forecasters little help in presenting the forecasts. Three of the dedicated business-forecasting programs contain features designed to reconcile forecasts across a product hierarchy, a task this group performs so commendably it can serve as a role model for forecasting engines in demand-planning systems.

Keywords: Automatic forecasting, batch forecasting, combining forecasts, event adjustments, fit period, forecast horizon, intermittent demand, judgmental override, method evaluation, method selection, out-of-sample test, prediction interval, product hierarchy, trading day variation.

Although journals are the primary means for reporting scientific advances in forecasting, software programs are the critical paths for implementation. An innovation that is not incorporated in software may take a decade or more to be transmitted through textbooks and eventually accepted in forecasting practice.

Today most forecasting programs are fast and efficient processors of data. Most operate seamlessly, taking good advantage of menus and dialog boxes, and fully supporting both spreadsheets and databases. Many offer automatic forecast-method selection, which is especially useful when you are forecasting a large batch of time series. If you know where you are going, the software will get you there expeditiously.

Few users of forecasting software, however, consider themselves methodological experts. Many take it for granted that the methods the software offers or automatically selects will prove the most suitable. Further, software developers' advertisements and fliers often herald the accuracy of their forecasting algorithms, giving the impression that results obtained can be trusted.

But, can forecasting practitioners rely on software to steer them in the right direction? Unsuspecting users may be misled into selecting inappropriate methods by unsupported claims for a method's forecasting performance. Alternatively, the software may fail to offer the information needed to make an appropriate

method selection and evaluation. A software program cannot be held accountable for a user's lack of theoretical and practical knowledge; however, it should be expected to help the forecaster adhere to certain principles. In this chapter, we evaluate various categories of forecasting software in the light of those principles.

The forecasting software market is broad and varied. Some practitioners rely on spreadsheet programs or forecasting add-ins to spreadsheets, taking advantage of the spreadsheet's wide installation base. Also for convenience, those using general statistical packages for data analysis may gravitate to the forecasting modules within these programs. Dedicated business-forecasting programs offer methods and features directed to extrapolation of time series data and many single-equation econometric techniques as well. The more sophisticated and expensive *batch* versions of these programs serve as forecasting engines in demand planning environments. For some sophisticated forecasters, econometric software can help to develop multi-equation causal models to forecast business and economic series. Finally, some forecasters have begun using software based on artificial neural networks for financial and economic forecasting.

Our primary goal in this chapter is to see how well forecasting software incorporates the principles described in Principles of Forecasting (Armstrong, 2001b). We hope that our evaluation will guide forecasters and software developers toward enhancements that extend their implementation of best practices.

Our selection of software packages is restricted to the statistical branch of forecasting methods, as depicted in Exhibit 4 of the introduction to Principles of Forecasting (Armstrong, 2001a), and further restricted to commercially available packages that apply forecasting methods to time-series data. We have decomposed this segment of the software market into four categories:

Spreadsheet add-ins,

Forecasting modules of broad-scope statistical programs,

Neural networks, and.

Dedicated business-forecasting programs.

We had planned to include a fifth category, forecasting engines for demand planning, but omitted it because software developers in that market were unwilling to submit their products for review. Demand planning typically involves automatic forecasting for a hierarchical structure of time series, reconciling discrepancies between item-level and group-level forecasts and developing supply-chain strategies on the basis of the

forecasts. To partially compensate partially for this omission, we have examined the product-hierarchy features found in several of the dedicated business-forecasting programs.

Here is a listing of the software programs reviewed for this study.

Program	Version/Date	Web Site
Spreadsheet Add-Ins		
Excel Data Analysis Tools	Excel 97: 1996	www.microsoft.com/office/excel
CB Predictor	V1 : 1999	www.decisioneering.com
Insight.xla	V1 : 1998	www.analycorp.com
Forecasting Modules of Statistical Programs		
Minitab	V11 : 1997	www.minitab.com
SAS /ETS	Version 7 : 1997-1999	www.sas.com
Soritec for W 95/NT	Student V 1.0	www.fisisoft.com
SPSS - Trends	Release 8.0: 1998	www.spss.com
Neural Network Programs		
NeuroShell Predictor	V2: 1998	wardsystems.com
NeuroShell Professional Time Series	V2.1: 1999	wardsystems.com
SPSS Neural Connection	V2: 1998	www.spss.com
Dedicated Business-Forecasting Programs		
Autobox	V5.0 : 1999	www.autobox.com
Forecast Pro	V4 and Unlimited : 1999	www.forecastpro.com
SmartForecasts	V5 : 1999	www.smartcorp.com
Time Series Expert	V2.31 : 1998	isro.ulb.ac.be/compstat.html
tsMetrix	V2.0 : 1997	www.rer.com

We have not considered software for multi-equation econometric modeling, both because of its highly specialized features and because it has yet to be shown that multi-equation models add value in forecasting (Allen and Fildes, 2001); however, many of the programs we have considered include single-equation econometric techniques based on regression models. We have omitted conjoint analysis programs because the methodology does not operate on time series data. Wittink and Bergestuen (2001) evaluate this methodology. Finally, we excluded non-statistical tools for enhancing judgment and providing decision support. However, some of the included programs permit judgmental inputs to forecasting.

PRINCIPLES AND STANDARDS FOR FORECASTING SOFTWARE

Forecasting software is not designed to deal with all aspects of forecasting. Users must perform such tasks as setting objectives, structuring problems and identifying information sources before initiating data analysis. They must rely on judgment in performing other tasks.

We identified six categories on the *Forecasting Standards Checklist* (Armstrong 2001b) to which forecasting software should be expected to make a contribution: preparation of data, method selection, method implementation, method evaluation, assessment of uncertainty, and forecast presentation. These categories contain 80 principles, of which we selected 30 as pertinent to forecasting software. The selection criterion was straightforward: that implementation of the principle could be abetted if it were automated or routinized within a programmed procedure. The principles we excluded represent forecasting strategies and perspectives that precede the use of software. For example, we included Principle 6.8 – *Compare the track records of various methods* – but not Principle 6.4 – *Use quantitative methods rather than qualitative methods*.

In addition to the 30 principles selected from the checklist, we added 15 software features *(swf)* to the evaluation criteria. Six of these features expedite the forecasting process (e.g., by enabling the user to withhold data for an out-of-sample evaluation of forecasting accuracy or to choose the criterion of best-fit). The other nine *swf* represent special principles applicable to forecasting within the structure of a product hierarchy, such as the reconciliation of item and group forecasts.

Tables 1-7 contain our ratings of how effectively a principle of forecasting has been implemented into forecasting software. We used the following rating system:

++: Effectively implemented principle (a best practice)

+: Partially implemented principle

o: Principle is ignored

–: Principle is undermined.

na: Principle is not applicable

With the exception of the neural network packages, we installed and tested the software programs on our own (standard-issue) PCs. These ratings represent our consensus judgments. When we initially differed, we exchanged written explanations. Ultimately, we came to agreement in all areas. The testing and rating of the neural network programs was performed by Tom Rubino, an expert on neural network software. Through written communication, we attempted to ensure consistency between the evaluation of neural network programs and the evaluations of the other categories of forecasting software.

We sent a preliminary version of the chapter and program ratings to all software providers. Responses were received regarding 8 of the 15 programs. In several cases, we made follow-up inquires. When a software

provider raised questions about specific ratings, we reexamined our ratings and answered their arguments. Based on this review process, we revised approximately 5% of the preliminary ratings.

Our primary objective was to examine the effectiveness of forecasting software in implementing forecasting principles. Our tables are not sufficiently detailed, however, to present full evaluations of individual software packages. We have not addressed some key features, such as the time investment for learning to operate a package, ease of use for those with a modest statistical background, complexity of user interfaces, quality and accessibility of technical support, availability of training, and price. In addition, we do not discuss matching the methodological strengths of software packages to user needs. Software customers should consult review articles about individual packages. You can access a Web site for software reviews through the Principles of Forecasting Web site: http://hops.wharton.upenn.edu/forecast. In addition, Rycroft (1999) lists software products with their features, prices, and developer information.

PREPARATION OF DATA

Some time series are too short, too volatile, or too unstable to be forecast on the basis of a statistical method. Hence, before undertaking a statistical-forecasting effort, the forecaster should determine whether the series is forecastable. One benchmark of forecastability is provided by the performance of a random-walk model, also called a *naïve-1*, which issues forecasts of 'no change' from the forecast origin to each period being forecast. As such, its forecast errors measure the degree of change in the series. Another approach is to decompose the time series into systematic and random components and assess the magnitude of the random component (noise) of the series.

Before you can identify a suitable forecasting method, it is often necessary to clean the time series, correcting errors, interpolating missing values, and identifying and possibly down-weighting outliers. Forecasters often overlook this critical principle in practice, inviting large forecast errors.

Data adjustments and transformations may also be necessary. Monthly time series, especially retail-level sales, exhibit *trading day variation* that, if undetected, can distort the calculation of seasonal indexes. Many weekly, monthly and quarterly series have seasonal components that can make it difficult to identify trends, special events, and the effects of causal variables. The modeling of time series for which the frequency distribution of observations or errors is notably skewed, or in which the degree of variation around trend changes systematically over time may benefit from transformations that normalize the data or stabilize the

spread of the series about its trend. A transformation may also be warranted on theoretical grounds to convert a variable to a change or percentage change. The most common transformation is that from the raw data to the (natural) logarithm of the series, which can accomplish all of the above objectives simultaneously, unless the data contain zeros or negative numbers.

Visual inspection of graphs (time plots) helps to reveal unusual data points, seasonality, and the presence and type of trends. In forecasting for a product hierarchy, time plots can also identify problems with individual time series – such as erratic behavior, intermittent demand (intervals with zero demand), or shifts in volume due to product replacements.

In Table 1, we give our ratings of how forecasting software performs in data-preparation tasks.

{Insert Table 1 about here}

Spreadsheet add-ins have not compensated for the spreadsheet's omission of automated data preparation features. You can use the basic spreadsheet to manually prepare data for forecasting. Because the add-ins do not automate this task, we assigned spreadsheet add-ins the rating o:ignored.

Forecasting modules of statistical programs have generic routines to facilitate the preparation of data for any statistical task. However, the user must often navigate between main-menu categories, a minor inconvenience. Their graphing capability is much more comprehensive than with other categories of software, but the user must define the data, axis, and variables each time. The programs offer a comprehensive set of transformations.

Neural net programs require thorough data preparation and two of the three NN packages we examined are among the best in conforming to these principles. These two adjust for seasonality and trading days and at least partially facilitate data cleaning and transformations. In the remaining NN package, Neural Connection from SPSS, the user can prepare the data within SPSS and pass it through to Neural Connection. None of the three NN programs use the more traditional models as comparative benchmarks to test whether the neural net improves accuracy enough to justify its added complexity and lack of transparency.

Dedicated business-forecasting programs treat data preparation as a critical step. They contain basic spreadsheets for entering and editing data, dialog boxes with options for missing values and outliers, and transformation menus. The reference manuals include warnings about the effects of missing data and the need to correct the problem before modeling the series. In most of these programs, users can make trading day

adjustments using an X-11 routine or a function that assigns trading day weights to the data. Their graphing capabilities, however, are uneven: scatter diagrams are not always offered, seasonally adjusted series are sometimes cumbersome to plot, and visual quality is too often inadequate for detecting data irregularities.

METHOD SELECTION

To select an appropriate method, forecasters need domain knowledge. They also need to examine the features of the time series. Time plots of transformed and adjusted data can reveal trend and cyclical patterns. It is sometimes difficult, however, to judge seasonality from a time plot because it is not always easy to see whether the peaks and troughs repeat *regularly* over the years. A helpful supplemental graph is the ladder chart, in which the horizontal axis lists the season (month or quarter) and values are plotted for each season's low, average, and high during the past several years. Levenbach and Cleary (1981, p.308) provide a useful illustration.

In addition, a statistical test for seasonality – often based on autocorrelations at the seasonal lags – can be a valuable feature of method selection. In a monthly time series, for example, seasonality would be indicated by a high (auto) correlation between values that are separated by multiples of 12 (and sometimes 13) periods. However, you normally need at least three years of monthly data for a statistical assessment of seasonality.

Although visually identifying trends and cycles may narrow the choice of plausible forecasting methods, you are often left with a number of candidates worthy of further screening. Comparing the forecasting track records of these finalists can be informative. The M3-Competition (Hibon & Makridakis, 2000) showed that automatic method-selection algorithms based on such comparisons were among the most accurate approaches to extrapolation of time series. In forecasting comparisons, it is important to discourage overfitting and unnecessary model complexity. Method selection based on a statistic that is adjusted for degrees of freedom is helpful because it penalizes complexity; however, the penalties are probably not strong enough. An information criterion, such as the Akaike Information Criterion AIC or the Bayesian Information Criterion BIC, provides a basis for method selection that imposes a stronger handicap on complex procedures.

When possible, analysts should base method selection (and evaluation) on out-of-sample tests rather than fit to the data. Out-of-sample accuracy is normally measured by holding out some portion of the historical time series from the data that is used to select and estimate the forecasting method. For example, the most recent 12 months may be withheld from a time series of 60 months to test the forecasting accuracy of a method fit to the

first 48 months of data. The software program should permit users to readily designate fit and test (holdout) periods.

Detecting patterns from graphs is important in selecting a forecasting method, as is managerial judgment about pattern changes. If several forecasting methods differ in the emphasis they give to different features of the data, the forecaster may find it advantageous to diversify the forecasting portfolio by combining forecasts from several methods. The combined-forecast errors are almost always smaller than the average of the errors from the individual forecasts, and sometimes as low as the errors from the best of the individual forecasts (Armstrong, 2001c).

For forecasting the large number of time series typically involved in a product hierarchy, automatic method selection is mandatory. Tashman and Leach (1991) identified five types of automatic method selection in the software of the 1980s. The 1990s have seen an explosion in the number and variety of these methodologies.

For causal methods, where you base forecasts on explanatory variables, the inclusion of lagged variables and lagged errors (*dynamic terms* in Table 2) can often improve model performance by accounting for effects that are distributed over more than one time period. In a regression model, you must specify the form of each causal variable as well as a time pattern for its effect on the variable to be forecast. Alternatively, you can incorporate causal variables into ARIMA models, which establish forms and time patterns on the basis of correlations in the data.

{Insert Table 2 about here}

Spreadsheet add-ins encourage users to look at graphs before selecting a forecasting method. One program, CB Predictor, automatically compares eight extrapolation techniques and ranks them based on fitting accuracy. The user can select from three statistical criteria for the ranking – RMSE, MAD, or MAPE. This feature affords excellent flexibility, although software should also offer method rankings based on an information criterion (which penalizes complexity) and on an out-of-sample error measure. Combining forecasts can be implemented manually in spreadsheets. Regression modeling in the add-ins is rudimentary and does not allow for dynamic terms.

Forecasting modules of statistical programs do not attend carefully to method-selection principles. Three of the four packages we reviewed fail to incorporate any best practices. Because the programs include a wide offering

of statistical techniques, forecasters can implement many of the principles using these products, but only through multiple manual steps and, in some cases, by writing specific programming instructions.

Neural net programs conform well to principles concerning methods selection. They base selection on best track record. They determine model parameters by "training", using a portion of the data, and then comparing the results against a test set of data. The programs discourage model complexity, although the NN approach itself is complex.

Dedicated business-forecasting programs are selective in their provision of best practices. By offering tests for trend and seasonality, most encourage the forecaster to match the forecasting method to the features of the data. Most facilitate method selection by providing comparisons of track records. In doing so, several programs take account of method complexity or out-of-sample performance. All but one of these five programs provides excellent regression facilities. Only one program offers a formal process for combining forecasts.

METHOD IMPLEMENTATION

After choosing a method, the forecaster faces a variety of decisions about method implementation. One concerns the portion of the data to serve as a fit period, with the remaining part of the series held out to establish an out-of-sample, forecasting track record. If a program provides the ability to automatically designate the period of fit, forecasters can easily test preliminary models over different time periods. It also enables the forecaster to conveniently update the coefficients of the preferred model by reincorporating some or all of the held-out data.

A program provides further flexibility if it permits the forecaster to choose a statistical criterion to define *best fit*. The typical default for both extrapolative and explanatory methods is minimization of a function of the squared errors. Alternatives include minimization of absolute errors or absolute percentage errors, criteria considered less sensitive to distortion from outliers. In automatic method-selection procedures, the rankings of the component methods can differ for different best-fit criteria.

If a special event occurred during the fit period, the forecaster can model it through dummy variables (regression), intervention analysis (ARIMA), or event indexes (exponential smoothing). If the forecaster expects a new event (one with no prior history) to occur in the forecast period, he or she must use judgment, either as an input to the model or in overriding a model forecast. Williams and Miller (1999) show how judgmental adjustments may be built into exponential smoothing methods.

Programs that offer exponential smoothing or ARIMA models automatically assign weights to the data such that in general the recent past is given more emphasis than the distant past. In many cases, this decaying pattern of weights is intuitively plausible. Other procedures, such as ordinary least squares regression, assume equal weighting of data unless the forecaster specifies unequal weights. Program options for weighted least squares give the forecaster greater flexibility. They also enable more efficient estimation of models in which the variance of the dependent variable has shifted over time.

When implementing a regression analysis, it is valuable to be able to use extrapolation procedures to forecast explanatory variables. This capability also permits the forecaster to compare extrapolation and causal methods.

{Insert Table 3 about here}.

Spreadsheet add-ins provide almost no flexibility in method implementation. Excel's Data Analysis Tools lists exponential smoothing as a method but offers only the smoothing procedure appropriate for non-trended, non-seasonal data. The other add-in programs provide a greater complement of smoothing procedures, allowing forecasters to extrapolate data for trended and seasonal series. For regression modeling, CB Predictor was the most effective of all the programs examined for integrating extrapolative forecasts of explanatory variables. For each explanatory variable, this program automatically chooses the best fit from among eight extrapolative procedures, and then enters the forecasts from this procedure into the regression equation. In all programs, judgmental forecasts can be entered manually.

Forecasting modules of statistical programs allow users to select the fit period and to weight the data, but they provide limited opportunities for judgmental adjustments. One of these programs, SAS/ETS, offers a choice of best-fit criteria, an ability to define and adjust for expected events (interventions) and a linkage from extrapolation methods to the forecasting of explanatory variables in a regression model.

Neural net programs permit users to select the fit (training) period, to choose an optimization criterion, to weight the data, and to integrate extrapolative forecasts of the input variables. Judgmental adjustments are possible in only one of the three packages and none permit adjustments for expected events. Overall, however, method implementation is a strong feature of NN programs.

Dedicated business-forecasting programs offer forecasters considerable flexibility in method implementation: the forecaster can conveniently specify the period of fit, adjust for special events, integrate judgment, and override statistical forecasts. The last capability can be abused, but it is often necessary in extrapolative and causal modeling. These programs offer no choice of best-fit criterion: in all cases, they are hard-wired to minimize the sum or mean of the squared errors. Only two of the programs automatically integrate extrapolation methods for forecasting the explanatory variables into the regression routine.

METHOD EVALUATION

A forecaster should determine whether the software program (a) assesses the validity of underlying model assumptions and (b) provides suitable measures of forecasting accuracy.

Model diagnostics

Analysts normally subject a forecasting model to a battery of diagnostic tests before using it to generate forecasts. Typically, they perform tests on the fitted (within-sample) errors. The diagnostics look mainly for non-random patterns in the fitted errors. In the absence of statistically significant indications of non-random behavior, the forecaster can proceed to test the model's forecasting accuracy. Contrary diagnostic results, such as a systematic pattern of errors, warn the forecaster that the model may need refining.

Analysts should also perform statistical significance tests on the estimated coefficients. The lack of statistical significance can point to problems with the data (e.g.; explanatory variables that provide overlapping information), to shortcomings in the model's ability to detect relationships, or to the presence of superfluous terms (and hence the desirability of a simpler model).

Forecasting Accuracy Measurement

The software should make a clear distinction between within-sample and out-of-sample accuracy. For causal methods, it is useful to make a further distinction, between ex post and ex ante out-of-sample tests. Ex ante tests are based on forecasts of both the dependent and explanatory variables. They provide true tests of forecasting accuracy but do not distinguish the extent to which forecast errors are attributable to the model or to misforecasts of the causal variables. Ex post tests assume that the explanatory variables are known in the forecast period; hence errors must be attributable to the model, not to misforecasts of the regressors.

The software should calculate forecast errors at each forecast horizon - delineating one period-ahead errors, two period-ahead errors, etc – ideally, using a variety of error measures. The traditional *standard error* is

calculated from the squared-errors; but this measure has fallen from favor because of its poor reliability (Armstrong, 2001d). Common alternatives are based on absolute errors (*MAD* for mean absolute deviation), *absolute* percent errors *(MAPE* for mean absolute percent error), and relative absolute errors *(RAE)* – the latter measuring a method's errors relative to those of a benchmark *naïve* method. If the forecasting track record is to be based on multiple time series, such that errors need to be averaged across different series, then you need averages based on percent errors or relative errors to avoid distortions.

{Insert Table 4 about here}

Spreadsheet add-ins provide rudimentary residual plots but no formal tests of model assumptions. They do not distinguish fitted values from forecasts. The Excel DAT regression tool provides only the two traditional fit measures – R-square and the standard error of estimate – and its exponential smoothing tool supplies no error measures. Insight.xla does not augment the spreadsheet's offering of error measures. CB Predictor is better in this area as it offers a variety of statistical measures of fitting accuracy, including the MAPE, but it does not measure out-of-sample forecasting accuracy - a serious omission in all add-in programs.

Forecasting modules of statistical programs have good to excellent model-validation tests for regression models, and comprehensive graphing facilities that allow for a variety of residual plots. Unfortunately, little of value has carried over to the extrapolation methods. Most of these programs do not effectively distinguish within-sample from out-of-sample evaluations and do not provide adequate variety in error measurement. One package reports only the sum of squared errors (SSE). Another supplies the MAPE for exponential smoothing but only the residual variance (MSE) and R-square for ARIMA and regression – an inconsistency that makes comparison of methods difficult.

Neural net programs make a clear distinction between in-sample and out-of-sample forecast accuracy. They also provide multiple measures of accuracy, although the list of measures could be broadened to include such measures as trimmed means, which adjust for outliers. The programs do not offer diagnostic tests of model assumptions, tests that could help users to identify correct neural architectures.

Dedicated business-forecasting programs effectively support method evaluation. Almost all contain model diagnostics, distinguish within-sample from out-of-sample accuracy, provide an adequate variety of error measures for both, and report forecast errors by forecast horizon. Despite a tendency in this market segment to

"show clients only what they can understand," the quantity and quality of evaluation tools and measures has kept pace with the research literature.

ASSESSMENT OF UNCERTAINTY (PREDICTION INTERVALS)

When the forecasting method is based on a theoretical model of how the time series was generated, you can derive prediction intervals (also called forecast intervals, interval forecasts, and confidence intervals for a forecast) objectively from the underlying model assumptions (plus an appeal to the normal distribution of errors). We call these *theoretical* prediction intervals.

Theoretical PIs, however, do not capture the full degree of uncertainty in the point forecasts, either because they do not take into account the possibility that the model being used is inadequate or that inputs to the model (forecasts of explanatory variables) are incorrect. In some cases, too, the theoretical PI is based on the assumption that the estimated coefficients of the model represent the true coefficients. The software program should document the various sources of error represented in the PI and highlight those that are omitted. For example, the software should reveal that its regression-model PIs account for sampling and estimation errors but assume, heroically, that the regressors are forecast without error, and that the model specification is appropriate for the forecast period. Tashman, Bakken, and Buzas (2000) show that accounting for regressor forecast error could easily double PI width.

Programs employ a variety of algorithms to calculate prediction intervals for exponential smoothing methods. Newbold and Bos (1989) show that some of these algorithms are based on untenable assumptions about the underlying pattern of the data and may be worse than nothing at all when conditions are changing. The software should inform the forecaster what methodology it uses to calculate the PI.

The inherent limitations of theoretical PIs make empirical PIs an attractive alternative. An empirical PI is derived from an actual or simulated distribution of prediction errors for a specific forecast horizon. Analysts usually maintain the normality assumption, so they can compute appropriate multiples of the forecast standard error. A disadvantage of the empirical PI is that its width is liable to shift irregularly over the forecast horizons, especially for small numbers of forecasts. In the case of a small number of forecasts, the empirical PI for a longer-term forecast can turn out to be narrower than that for a shorter-term forecast. It should be possible to smooth the empirical PIs; however, to date, only one program offers such a feature and does so in an arbitrary manner.

The principles of combining forecasts can be extended to combining prediction intervals from alternative methods. Unfortunately, the performance of prediction intervals – theoretical, empirical and combined – has not been examined in the forecasting competitions to date.

{Insert Table 5 about here}

No category of software effectively implements the principles for assessment of uncertainty.

Spreadsheet add-ins offer the standard, theoretical prediction intervals for regression models. CB Predictor also provides empirical prediction intervals for its moving-average and exponential-smoothing procedures. These empirical PIs, however, are based on within-sample prediction errors. The manual contains no descriptions or explanations. This program also allows users to input forecasts to the complementary risk analysis program, Crystal Ball, which enables Monte Carlo simulations for the assessment uncertainty in the forecasts. Insight.xla permits users to simulate prediction intervals using the normal distribution and assumptions about the forecast standard error. The simulation facility may encourage forecasters to calculate alternatives to theoretical prediction intervals.

Forecasting modules of statistical programs supply theoretical prediction intervals for regression and ARIMA models. Most of these programs provide only point forecasts for smoothing procedures. The outdated rationale is that smoothing procedures are not based on a theoretical view of the pattern in the data. SAS/ETS is the exception in this software category, providing prediction intervals for smoothing procedures that are based on analogous theoretical (ARIMA) models. Overall, forecasting modules of statistical programs are not adequate for forecasting via exponential smoothing.

Neural net programs currently do not supply theoretical, empirical or qualitative assessments of uncertainty. This area needs the attention of NN analysts.

Dedicated business-forecasting programs are rarely explicit about the procedures they use for calculation of theoretical prediction intervals. No program clearly explains the assumptions and limitations behind the procedure adopted. This omission is ironic in light of the detailed discussions these programs give to the forecast methods themselves. Moreover, the width of PIs supplied for equivalent methods differs across software programs. Time Series Expert and Forecast Pro use ARIMA model representations to calculate PIs for exponential smoothing, a technique devised in part by TSE co developer Guy Melard (Broze and Melard, 1990).

Forecast Pro uses Chatfield-Yar procedures (1990, 1991) for seasonal smoothing models. Empirical PIs, developed or bootstrapped from forecast errors, are not generally provided - SmartForecasts is the exception in this category - but can be manually calculated in those programs that provide rolling out-of-sample forecast errors.

FORECAST PRESENTATION

Critical to the forecasting process is the organization's acceptance and integration of the forecasts into the managerial process. The forecaster must strive to demystify the forecasting methodology and demonstrate that the forecasts have a plausible and trustworthy foundation. The forecast presentation should include a description of assumptions, an explanation of and justification for the method selected, a graphical demonstration that the forecasts are a plausible progression from historical patterns, a description and illustration of how the forecasts are generated, and a discussion of the uncertainty surrounding the forecasts. Gaining acceptance for forecasts is partly an educational process: the more decision makers learn about forecasting and statistical methodology, the better they will be able to recognize effective forecasting efforts.

Software should help practitioners in all phases of the presentation process. It should also possess the mundane ability to export data and forecasts to end-users. The main sources of assistance are forecast reports, user guides and reference manuals.

{Insert Table 6 about here}

Spreadsheet add-ins contribute little to the presentation of forecasts other than the provision of forecasts in an exportable format. One add-in produces an effective graphic of the time series, point and interval forecasts, as well as a rudimentary forecast report. It gives no indication, however, that forecasting models are based on assumptions about reality that users must understand and validate.

Forecasting modules of statistical programs match dedicated business-forecasting programs in data exportability, but, with one exception, fall behind them in the graphic presentation of forecasts, in making the theoretical assumptions transparent, and in explaining how forecasts have been generated. This category of software does not supply forecast reports.

Among the three *neural net programs*, one package provided effective justifications for the models selected. The programs only partially implement principles of good graphical presentation, given the absence of upper and lower bounds of uncertainty. None provide forecast reports.

Dedicated business-forecasting programs generally match the spreadsheets and add-ins in providing exportable formats and presentation graphics. Most of these programs reveal the theoretical assumptions and forecasting methodology and explain how they produce forecasts. Most limit their forecast reports to numerical tabulations that are stingy on explanations and illustrations. These programs could do a much better job in facilitating forecast presentation.

FORECASTING ACROSS A PRODUCT HIERARCHY

Product hierarchies are families of related product lines. For example, a brand of toothpaste may come in several flavors and each flavor may be packaged in several tube sizes. The forecaster's task is to project volume of demand for each stock-keeping unit (sku) – tube size of a specific flavor – as well as total demand for each flavor and overall demand for the brand. Forecasts made at each level of the hierarchy must be reconciled.

The forecaster can choose from several strategies in order to reconcile multi-level forecasts: (a) Develop a model-based forecast for each sku and aggregate sku forecasts to obtain flavor and brand totals (a *bottom-up* strategy); (b) Directly forecast the aggregate brand data and use these forecasts to create or modify the forecasts for flavor totals and individual tube sizes (a *top-down* strategy); (c) Create model-based forecasts for each *flavor,* summing these forecasts to obtain a forecast for the brand total and disaggregating to obtain individual tube size forecasts (a *middle-out* strategy). In addition, reconciliation can be accomplished by applying historical proportions to disaggregate a group forecast.

In forecasting demands for the typically large number of items in a product hierarchy, the forecaster must rely mainly on automatic procedures for selecting the forecasting method. The software should be able to detect data features – trend and seasonality, for example - and choose an appropriate forecasting procedure for each of the items to be forecast. Since forecasts at different levels must be reconciled, the software should offer bottom-up, top-down and middle-out approaches to reconciliation.

If not adjusted for, special events, such as irregular promotions and natural catastrophes, can distort the forecasting equation. Some programs offer event adjustment procedures for the family of exponential smoothing methods that prevent confounding of trend and seasonal indexes. The user must identify the timing

of the special event. The event-adjustment capability, however, cannot be applied to assess the impacts of quantitative event variables.

Intermittent (also called interrupted) series present another challenge in forecasting the product hierarchy. Such series reflect a pattern of demand in which orders occur in clumps with periods of zero demand. Demands for high-cost computer components and aviation replacement parts tend to be intermittent. Simple exponential smoothing can be improved upon in such cases (Willemain, et al, 1994) by procedures that project the demand interval as well as the average demand and by simulation (bootstrapping) of potential demand from the distribution of actual demands.

The software should flag for manual review time series for which out-of sample forecast errors exceed user-specified limits. If the forecaster wishes to make a judgmental override of a forecast, the program should automatically reconcile the change across the product hierarchy. The program should also enable the forecaster to compare the accuracy of different strategies for reconciliation.

Only three of the 15 packages we evaluated include systematic features for linking and reconciling forecasts - Autobox, Forecast Pro, and SmartForecasts. Batch versions of these packages can and do serve as forecasting engines for demand planning – these are versions that have no restrictions on the number of time series they can accommodate.

Many companies are using forecasting modules that are part of larger demand-planning, supply-chain management or enterprise-resource-planning systems. These systems link forecasting engines with relational databases and with business applications programs, and they are often sold with installation, training, and consulting services as complete forecasting solutions. The databases, business applications functions, and individualized services can easily multiply costs over that of a forecasting engine by a factor of 10 or more.

The developers of these encompassing systems have sought to limit outside scrutiny of their software products, fearing that a negative evaluation or a comprehensive cost-benefit comparison might damage sales. Hence, we were unable to enlist the participation of such firms in our evaluation of forecasting software. A listing of companies providing software in this category can be found at the American Production and Inventory Control Society (APICS) Web site: www.apics.org. In Table 7, we present our standards for multilevel (product hierarchy) forecasting and our evaluation of the implementation of these standards by the three programs that offer multilevel capabilities. You can use this checklist to evaluate vendors or internally-generated forecasting systems.

{Insert Table 7 about here}

These three packages offer forecasters the functionality of the dedicated business-forecasting program and, in addition, provide the ability to automatically forecast a large batch of time series. Batch forecasting as a task, however, disconnects the forecaster from the data, thus restricting application of some best practices. The forecaster must keep in mind the advantages of reviewing certain time series individually.

The strength of these batch-forecasting programs for a product hierarchy lies in their automatic forecasting and reconciliation features. For two of the three programs, automatic forecasting is rooted in the family of smoothing methods and works by comparing forecasting errors from alternative smoothing specifications. In the third program, automatic forecasting is based on ARIMA models. The results of the M3-competition showed that the method-selection procedures in these programs worked well as compared to the application of any single forecasting method to all time series in the batch. (Hibon and Makridakis, 2000) For other tasks, including flagging problem series and forecasts and comparing alternative reconciliation strategies, the forecaster would benefit from using supplemental software.

SUMMARY OF RATINGS BY SOFTWARE PROGRAM AND CATEGORY

In this section, we summarize our evaluations by program and category of software. A ++ rating for every principle in Tables 1-6 would earn a program a score of 66. In Table 8, we present the aggregate ratings as a percent of this maximum-possible rating. For example, a score of 50% indicates that our ratings of this software summed to a raw score of 33, 50% of the maximum.

We remind you that we have omitted several considerations that can loom large in a software-purchase decision, such as the time investment for learning to operate a package, ease of use for those with modest statistical backgrounds, complexity of user interfaces, quality and accessibility of technical support, availability of training, and price.

{Insert Table 8 about here}

Spreadsheet add-ins as a group (16% of maximum) currently implement few principles of forecasting and cannot be recommended to the practitioner as an adequate forecasting solution. They offer only a few smoothing procedures and a rudimentary regression tool. They do not offer adequate data preparation features, provide limited choices in method selection and estimation, and do not assist in forecast presentation. Most serious is the omission of features for evaluations of (out-of-sample) forecast accuracy.

Forecasting modules of statistical programs (42% of maximum) were effective in data preparation; however, with one exception, the critical tasks of method selection and evaluation are left to trial and error on the part of the forecaster. The exception is SAS/ETS, whose forecasting functionality can be recommended to users of SAS. We advise users of the other general statistical packages, however, to obtain a dedicated business-forecasting program for their time series needs.

Neural net programs (*38*% of maximum) differ widely in data-preparation features but all are strong in forecast method selection and implementation. The neural net programs fall short of best practices in the evaluation of forecast accuracy and assessment of uncertainty. In addition, these programs do not use the more traditional models as comparative benchmarks to test whether the neural net model improves accuracy enough to justify its added complexity and lack of transparency. Success in convincing decision makers of the validity of neural net forecasts is a function of the skill and persuasiveness of the forecaster.

Dedicated business-forecasting programs (60% of maximum) have the superior record in implementation of best practices. In this market category, forecasters can expect at least partial implementation of forecasting principles from the beginning to the end of the forecasting process. Yet as the 60 percent figure suggests, these programs fall far short of consistent implementation of best practices. Data preparation is generally good but could be more effectively automated. There are major weaknesses in the assessment of uncertainty and in forecast presentation. The strengths of these programs lie in method selection, implementation, and evaluation. They all clearly distinguish fitting from forecasting accuracy.

SUMMARY OF RATINGS BY PRINCIPLE

Table 9 shows aggregate program and software category ratings on each principle of forecasting.

{Insert Table 9 about here}

In the aggregate, forecasting software is realizing about 50 percent of relevant forecasting best practices. The steepest shortfall occurs in assessment of uncertainty (22% of maximum): the packages are frequently *ad hoc* and secretive about the production of prediction intervals and uninformative about the sources of uncertainty in the forecasts. We recommend that software developers give greater attention to empirical and combined-method prediction intervals. Cox and Loomis (2001) found that assessment of uncertainty was also the weakest area of coverage in forecasting textbooks.

Software has upgraded its procedures for method selection (43% of maximum) and method implementation (42% of maximum) during the past decade but has not succeeded in coalescing around common standards. Too often, selection rules seem motivated more by marketing considerations than by forecasting research. The distinction between within-sample and out-of-sample performance must be sharpened and the emphasis shifted to the latter. Very infrequently does software offer any alternative offered to the least-squares (or minimum mean squared error) criterion for model fit. The tools for incorporation of expert judgment are crude, and research shows that refinements in this area could be of great value.

Method evaluation (51% of maximum) and forecast presentation (48% of maximum) are relatively strong areas for software and can be further strengthened with little new technology. We recommend that the software present its point and interval forecasts within a process that makes assumptions explicit and indicates whether and how the validity of the assumptions has been tested.

IMPLICATIONS FOR PRACTITIONERS

Forecasting practitioners may consider themselves bound by organizational habits or financial constraints to spreadsheets or general statistical packages. Our evaluations suggest that analysts cannot currently apply best forecasting practices within the spreadsheet medium without substantial manual effort or programming. For these forecasters, a dedicated business-forecasting program would improve implementation of best forecasting practices without sacrificing ease of data handling. The cost of entry versions of these software packages is about that of a spreadsheet program with the add-in.

General statistical packages contain a regression capability and a number of methods for extrapolating time series. We found that one such system, SAS/ETS, was as effective as any of the dedicated business-forecasting programs in implementing best forecasting practices.

At the current level of implementation of best practices, we recommend that users of the other general statistical packages choose a dedicated business forecasting program for forecasting time series data. General statistical programs have lagged behind in implementing best practices in method selection, evaluation, and assessment of uncertainty.

The current generation of neural net programs should be viewed as supplements rather than replacements for traditional business forecasting methods. Developers have begun to introduce neural net functionality into dedicated business-forecasting programs and general statistical programs: its further diffusion could provide forecasters with an integrated solution.

For forecasting a product hierarchy, the three programs we reviewed have state-of-the-art features for automatic method selection, forecast reconciliation, and coping with special events and intermittent demands. Indeed, they provide a benchmark for judging forecasting engines in demand planning software.

Because demand-planning software is much more costly than other categories of forecasting software, and because they fear negative reviews could damage sales, the developers of demand-planning software have restricted their general evaluation. Potential users should carefully evaluate each program's forecasting functionality and how effectively it implements the forecasting principles, and not merely dwell on its data-management features. They should request input from existing adopters as well as from organizations that have tested but rejected the demand-planning package. They should test a program’s forecasting accuracy against one of the forecasting engines reviewed here.

Forecasting software will evolve over time as new products enter the market and as today's products change to implement more of the principles of forecasting. That is the nature of software development. While no current package implements best practices in every area, we hope that software programs increasingly strive to help clients apply principles of forecasting, to encourage users to follow appropriate procedures outside the realms of the programs, and to make analysts aware of the limitations of forecasting methodologies. In effect, software programs should be the primary means for implementing the principles of forecasting.

In the meantime, analysts must be cognizant of software limitations and go beyond the software to implement best practices.

IMPLICATIONS FOR DEVELOPERS

Splendid improvements have been made in the past decade in method selection algorithms, speed of computation, and data management. Developers wisely continue to refine these important areas as they screen new technologies and procedures for incorporation as program enhancements.

For financial reasons, developers increasing view program features and even program calculations as proprietary, not to be subjected to the scrutiny of clients, competitors, and reviewers. Forecasting software is becoming less transparent in describing method-selection procedures, specific tests of statistical significance, and the basis for calculating interval forecasts. Programs do not make clear to the practitioner why forecasts could go wrong. Advances in automating method-selection procedures can come at the expense of forecaster involvement in this important judgmental process. This is unwise. Practitioners cannot defend forecasts if they do not understand why a particular forecasting method was chosen.

We urge software firms to encourage forecasters' active involvement in method selection. Method evaluation schemes should be designed to provide appropriate and efficient feedback. Forecasting manuals should more fully explain that forecast models make simplifying assumptions, that only some of these can be formally tested, and that, as a consequence, the accuracy of any forecasts cannot be judged as precisely as the statistics of model-fit indicate. Transparent explication of the underlying sources of uncertainty behind any forecast would give forecasters valuable insight. It would also improve the business world's perception of the potential of and limitations of forecasting methodology and practice.

REFERENCES

Allen, G. & R. Fildes (2001), "Econometric forecasting" in J.S. Armstrong (ed.), *Principles of Forecasting.* Norwell, MA: Kluwer Academic Publishers

Armstrong, J. S. (2001a), " Introduction," in J.S. Armstrong (ed.), *Principles of Forecasting.* Norwell, MA: Kluwer Academic Publishers

Armstrong, J. S. (2001b), " Standards and practices for forecasting" in J.S. Armstrong (ed.), *Principles of Forecasting***.** Norwell, MA: Kluwer Academic Publishers

Armstrong, J. S. (2001c), "Combining forecasts" in J.S. Armstrong (ed.), *Principles of Forecasting.* Norwell, MA: Kluwer Academic Publishers

Armstrong, J. S. (2001d), "Evaluating forecasting methods" in J.S. Armstrong (ed.), *Principles of Forecasting.* Norwell, MA: Kluwer Academic Publishers

Broze, L. & G. Melard (1990), "Exponential smoothing: Estimation by maximum likelihood," *Journal of Forecasting,* 9, 445-455.

Chatfield, C. & M. Yar (1991), "Prediction intervals for multiplicative Holt-Winters, " *International Journal of Forecasting* 7, 31-37.

Cox, J. & D. Loomis (2001), "Diffusion of forecasting principles: An assessment of books relevant to forecasting" in J.S. Armstrong (ed.), *Principles of Forecasting***.** Norwell, MA: Kluwer Academic Publishers

Hibon, M. & S. Makridakis (2000), "The M3-competition", *International Journal of Forecasting* (forthcoming)

Levenbach H. & J. Cleary (1981), *The Beginning Forecaster.* Belmont, CA: Lifetime Learning Publications

Newbold, P. & T. Bos (1989), "On exponential smoothing and the assumption of deterministic trend plus white noise data-generating models," *International Journal of Forecasting,* 5, 523-527.

Rycroft, R.S. (1999) Microcomputer software of interest to forecasters in comparative review: Updated again", *International Journal of Forecasting*, 15, 93-120.

Tashman, L.J. & M.L. Leach (1991), " Automatic forecasting software: A survey and evaluation," *International Journal of Forecasting,* 7, 209-230.

Tashman, L.J., T. Bakken & J. Buzas (2000), "Effect of regressor forecast error on the variance of regression forecasts," *Journal of Forecasting* (forthcoming)

Willemain, T.R., C.N. Smart, J.H. Shocker and P.A. DeSautels (1994), "Forecasting intermittent demand: A comparative evaluation of Croston's method," *International Journal of Forecasting,* 10, 529-538.

Williams, D.W. & D. Miller (1999), " Level-adjusted exponential smoothing for modeling planned discontinuities," *International Journal of Forecasting,* 15, 273-289.

Wittink, D.R & T. Bergestuen (2001), "Forecasting with conjoint analysis" in J.S. Armstrong (ed.), *Principles of Forecasting.* Norwell, MA: Kluwer Academic Publishers

Yar, M. & C. Chatfield (1990), "Prediction intervals for Holt-Winters forecasting procedure, " *International Journal of Forecasting* 6, 1-11.

Acknowledgments

The authors thank Tom Rubino for his contributions on the neural network programs, to Robert Rycroft, Tim Davidson and four anonymous reviewers for their critical perspectives on the chapter and to the software developers who took the time to examine our evaluations and correct our errors.

Statement of author/developer affiliations

Len Tashman is a professor in the School of Business Administration of the University of Vermont. He has written dozens of software reviews for forecasting journals and newsletters, maintains a Web site for software reviews, and has served as a beta tester for many software programs. He has collaborated with Business Forecast Systems in delivery of professional education workshops and has made extensive use of Forecast Pro in the classroom. To ensure objectivity, ratings of Forecast Pro were given particular scrutiny by his co-author.

Jim Hoover is an officer in the United States Navy. He has previously written software reviews of Smart Forecasts for Windows, one for the *International Journal of Forecasting*, the other for *The Forum* (now called *The Oracle of IIF*). He has used the following programs on the job: Crystal Ball, Forecast Pro, Insight.xla, Minitab, and SPSS Trends, as well as internally-developed enterprise forecasting programs. He has no relationship with any software developer.