Paper 2973-2019

# Fit Statistic Smoothing: Minimizing Bias in Time Series Forecasting

Zachary D. Blizard, Hanesbrands Inc.

## ABSTRACT

Firms depend on SAS® software to build, execute, and maintain forecasting models. Practitioners, whether using SAS/ETS® or SAS® Forecast Studio, can fit various models to a series and deploy the most accurate one according to a specified selection criterion. Practitioners will often combine the forecasts from different models (i.e. ensemble modeling). The ensembling process involves combining forecasts whose associated models are ranked using a single error metric. This practice is not the best option available because no single error metric is globally optimal. Instead, forecasters should combine the forecasts from champion models selected using numerous error metrics. This results in a more stable and robust forecast because the outcome is an average of many forecasts, each from the champion of a unique selection criteria. This paper describes this process and demonstrates its benefits.

## INTRODUCTION

Stable and robust forecasts are not necessarily accurate. A model that forecasts a constant of 30 is stable and robust, but it can hardly be described as accurate in its modeling of a stochastic process, such as demand. For forecasts to be accurate, they must capture and extrapolate the systematic variation present in the series. Different models may detect the signal equally well, and when this is the case, most practitioners will opt for the most parsimonious model. However, there may be some that detect the signal equally well and are equally parsimonious. This makes model selection challenging. For example, SAS® Forecast Studio offers over 40 pre-made time series forecasting models, all of which attempt to capture and extrapolate the signal in a time series ("SAS® High-Performance Forecasting 4.1: User's Guide," 2011). These models often appear to fit a time series equally well, which is a reason why practitioners rely on error metrics as selection criteria when choosing a model to deploy. SAS® offers numerous error metrics to choose from ("SAS® High-Performance Forecasting 4.1: User's Guide," 2011). Table 1 lists available error metrics. The metrics highlight in grey are the ones utilized by the method being proposed in this paper. Table 2 lists the names of the premade time series models ("SAS® High-Performance Forecasting 4.1: User's Guide," 2011).

Routinely, no available model, even the champion model (the best model selected based on the fit statistic), produces an accurate forecast. To address this issue, practitioners will often combine models or model forecasts, using one of various methods, to produce more accurate forecasts (Blair, Leonard, and Elsheimer, 2012; Zou, and Yang, 2004; Lean, Shouyang, Lai, and Nakamori, 2005). Combining modeled forecasts by averaging is a common, straight-forward technique for ensembling, which involves summing the forecasted values across all models for each time interval and dividing by the number of models. Nevertheless, this process by which combined model forecasts are generated and used is not guaranteed to be optimal.

Table 1. Available Error Metrics for Selection Criteria

| | | |
|---|---|---|
| Adjusted R-squared | Akaike Information Criterion | Akaike Information Criterion, finite sample size corrected |
| Amemiya's Adjusted R-squared | Amemiya's Prediction Criterion | Geometric Mean Absolute Error Percent of Standard Deviation |
| Geometric Mean Percent Error | Geometric Mean Predictive Percent Error | Geometric Mean Relative Absolute Error |
| Geometric Mean Symmetric Percent Error | Maximum Absolute Error Percent of Standard Deviation | Maximum Error |
| Maximum Percent Error | Maximum Predictive Percent Error | Maximum Relative Error |
| Maximum Symmetric Percent Error | Mean Absolute Error | Mean Absolute Error Percent of Standard Deviation |
| Mean Absolute Percent Error | Mean Absolute Predictive Symmetric Percent Error | Mean Absolute Scaled Error |
| Mean Absolute Symmetric Percent Error | Mean Error | Mean Percent Error |
| Mean Predictive Percent Error | Mean Relative Absolute Error | Mean Relative Error |
| Mean Squared Error | Mean Symmetric Percent Error | Median Absolute Error Percent of Standard Deviation |
| Median Absolute Percent Error | Median Absolute Predictive Percent Error | Median Relative Absolute Error |
| Median Absolute Symmetric Percent Error | Minimum Absolute Error Percent of Standard Deviation | Minimum Error |
| Minimum Percent Error | Minimum Predictive Percent Error | Minimum Relative Error |
| Minimum Symmetric Percent Error | R-square | Random Walk R-square |
| Root Mean Squared Error | Schwarz Bayesian Information Criterion | Sum of Squares Error |
| Total Corrected Sum of Squares for the dependent variable | Total Sum of Squares | Unbiased Mean Square Error |
| Unbiased Root Mean Square Error | | |

The models used to create the ensemble are typically top performing models from the set of all models. For example, a forecaster may use three of the top five best performing models, according to their MAPE score, to produce a combined model. The order of best to worst performers is determined by the selection criterion chosen at the beginning of the models' fitting process. Thus, the "best" models are only the "best" according to the individual error

metric used as the selection criterion. As was shown earlier, there are many available error metrics. From the practitioner's perspective, it is difficult to know what fit statistic is the most appropriate to use, especially since she\he is forecasting for thousands of different series. This is not, however, a valid reason for using a single error metric in all cases, especially from a statistical point of view, which is why a new and more robust way to combine forecasts, and to forecast generally, is being proposed in this paper.

Table 2. Available Time Series Models

| | |
|---|---|
| Airline Model | Log ARIMA(2,1,2)(0,1,1)s NOINT |
| ARIMA(0,1,1)s NOINT | Log Damped Trend Exponential Smoothing |
| ARIMA(0,1,1)(1,0,0)s NOINT | Log Double (Brown) Exponential Smoothing |
| ARIMA(0,1,2)(0,1,1)s NOINT | Log Linear (Holt) Exponential Smoothing |
| ARIMA(0,2,2)(0,1,1)s NOINT | Log Linear Trend |
| ARIMA(2,0,0)(1,0,0)s | Log Linear Trend with Autoregressive Errors |
| ARIMA(2,1,0)(0,1,1)s NOINT | Log Linear Trend with Seasonal Terms |
| ARIMA(2,1,2)(0,1,1)s NOINT | Log Mean |
| Damped Trend Exponential Smoothing | Log Random Walk with Drift |
| Double (Brown) Exponential Smoothing | Log Seasonal Dummy |
| Linear (Holt) Exponential Smoothing | Log Seasonal Exponential Smoothing |
| Linear Trend | Log Simple Exponential Smoothing |
| Linear Trend with Autoregressive Errors | Log Winters Method – Additive |
| Linear Trend with Seasonal Terms | Log Winters Method – Multiplicative |
| Log Airline Model | Mean |
| Log ARIMA(0,1,1)s NOINT | Random Walk with Drift |
| Log ARIMA(0,1,1)(1,0,0)s NOINT | Seasonal Dummy |
| Log ARIMA(0,1,2)(0,1,1)s NOINT | Seasonal Exponential Smoothing |
| Log ARIMA(0,2,2)(0,1,1)s NOINT | Simple Exponential Smoothing |
| Log ARIMA(2,0,0)(1,0,0)s NOINT | Winters (Additive) Method |
| Log ARIMA(2,1,0)(0,1,1)s NOINT | Winters (Multiplicative) Method |

## FIT STATISTIC SMOOTHING

Instead of using one error metric to rank models and then combining a few of the best performers, a more robust, and arguably better, method is to gather the 31 champion forecasts for all the error metrics available in SAS® Forecast Studio (the highlighted error metrics) and average the forecasts produced from the champion models. The following are benefits of this method:

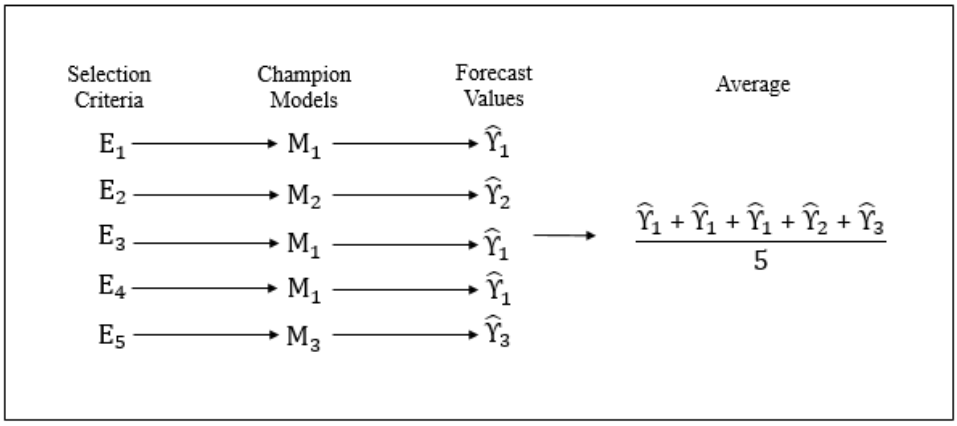
- The forecast will never be the worst. However, it will not necessarily be the most accurate relative to some of its component forecasts.
- Instead of rolling the dice on one error metric, this new process effectively hedges against all error metrics to minimize the random variability in accuracy.

- It has the strengths of each of the component champion forecasts.
- It will be robust against runaway step shifts and slopes.
- Its accuracy will be less uncertain.
- No arbitrary choice of one error metric will be involved on the part of the forecaster.

Though some of the component forecasts will likely be more accurate over a 1 to 2 year time horizon, the practitioner will not know this beforehand. It will only be known in an ad hoc manner, after new actuals are presented. Therefore, in the presence of such uncertainty, choose the less volatile method that will smooth the random variability in error metrics and result in a consistently robust approach.

A natural weighting process is present in this new manner of combining forecasts. There will be various models that are selected multiple times across the set of error metrics. Thus, these models will have forecast values that are included more than once in the average, which means they have more of an influence on the combined forecast values. This satisfies intuition because if a model is chosen multiple times across error metrics, then this seems to indicate that it is particularly good at modeling the data and, therefore, ought to be given extra influence in the final forecast values. The diagrams in Figure 1 offers a simple illustration of this process.

Figure 1. Weighting Process




$$\frac{3\hat{Y}_1 + \hat{Y}_2 + \hat{Y}_3}{5} = \alpha\hat{Y}_1 + \beta\hat{Y}_2 + \delta\hat{Y}_3, \text{where } \alpha + \beta + \delta = 1$$

$$= 0.60\hat{Y}_1 + 0.20\hat{Y}_2 + 0.20\hat{Y}_3 = \overline{Y}$$

The process in Figure 1 is as follows:

1. The process begins with a set of selection criterions to be used for choosing champion models. Five error metrics are used as criteria in the illustration, $E_1 - E_5$.
2. The five metrics choose from a pool of five models, $M_1 - M_5$, resulting in three champion models being selected. $M_1$ is the selected champion for three of the error metrics, and $M_2$ and $M_3$ are selected for one error metric each.
3. The champion models then produce five forecast values, three of which are the same because they come from the same model, $M_1$.

4. These values are summed together and then divided by five, the number of models.
5. The math at the bottom of the diagram in Figure 3 illustrates how the weights are calculated. Since $\hat{Y}_1$ is added together three times, it can be treated as $3\hat{Y}_1$. When the fraction is split into the three parts, the actual weights corresponding to the forecast values are more easily seen. The weight on the forecast value from the most frequently selected champion, denoted by $\alpha$, is equal to 0.60. The second weight, denoted by $\beta$, is equal to 0.20, and the third weight, denoted by $\delta$, is also equal to 0.20. The weights sum to 1. Thus, the forecast value of the most frequent champion is more heavily weighted and will have a greater impact on the final ensemble forecast value.

Depending on the error metric used as a selection criteria, the champion model will change, as can be seen in Table 3.

Table 3. Variability in Selected Models

| Champion Models by Selection Criterion (Top 5) | | | |
|---|---|---|---|
| Top 5 | Mean Squared Error (MSE) | Geometric Mean Percent Error (GMAPE) | Akaike Information Criterion (AIC) |
| 1 | Winters (Additive) Method | Winters (Multiplicative) Method | ARIMA(0,1,2)(0,1,1)s NOINT |
| 2 | Seasonal Exponential Smoothing | Log Winters Method – Additive | ARIMA(2,1,2)(0,1,1)s NOINT |
| 3 | Log Winters Method – Additive | Winters (Additive) Method | Log ARIMA(0,1,2)(0,1,1)s NOINT |
| 4 | Log Seasonal Exponential Smoothing | Log Seasonal Exponential Smoothing | Log ARIMA(2,1,2)(0,1,1)s NOINT |
| 5 | Winters (Multiplicative) Method | Seasonal Exponential Smoothing | Airline Model |

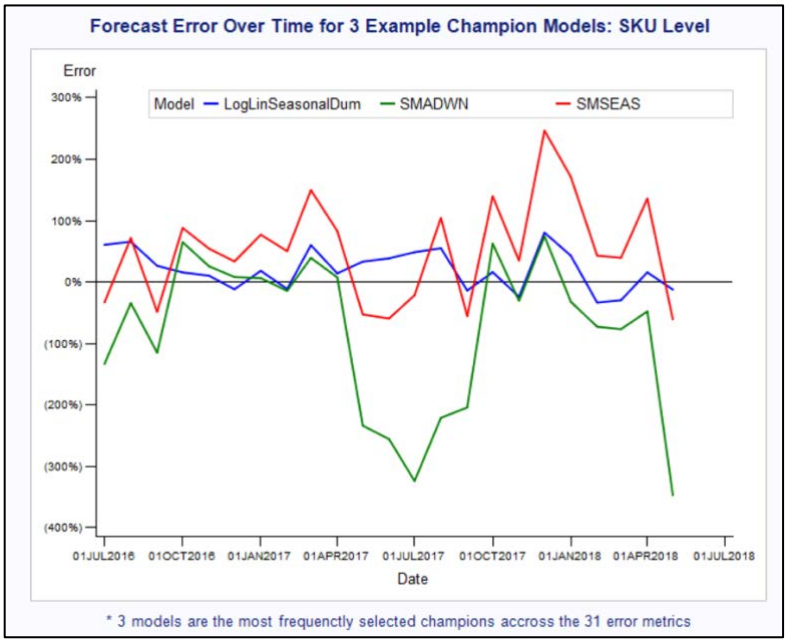
The three fit statistics in Table 3, MSE, GMAPE, and AIC are chosen because of their common use as selection criteria. As can be seen, the champion for each statistic of fit is different. The top five best performing models and their order are different, except for the fourth-place model for MSE and GMAPE, which is the LOGSEASONAL model. This demonstrates that the error metric is an important determinant of the selected model. The selected models can vary significantly, and in turn, produce very different forecasts. Depending on the selected champion model, the forecast accuracy will differ as can be seen in Figure 2.

Figure 2 plots the percent change of the residuals (Y-axis) across the forecast time period (X-axis) for four of the most frequently chosen champion models. The percent changes of the residuals are calculated by comparing the forecasted values to the actual values in the holdout sample. In Figure 2, the most visibly inaccurate forecast results from the SMADWN model, and the most visibly accurate results from using the LOGLINSEASONALDUM model. What is surprising, however, is that the SMADWN is the selected champion most frequently across the various error metrics. This suggests that depending on the metric a practitioner chooses for selecting a model to forecast with, she\he has a high likelihood of deploying an inferior model, and overlooking other, superior models that produce more accurate forecasts.

At the moment of forecasting, there's no way to know if the chosen selection criterion will result in the choosing of the best or the worst champion model. This adds a lot of uncertainty to the overall process. By averaging all champion forecasts across all the error

metrics, the practitioner knows that his\her forecast will fall somewhere in the middle of the distribution of the least to most accurate champion forecasts. This process is attempting to minimize the overall residual variability. Using a simple average dilutes the influence of runaway slopes and step shifts that might appear in individual forecasts. Bearing in mind the forecasts included in the average are champions, there won't be as wide of a range of forecast values. This suggests that the process is highly risk-resistant and eliminates the threat of using the least appropriate error metric. It also eliminates the risk of using a forecast with a runaway slope or step shift. The forecast will also be less volatile due to the process of averaging and incorporating the weights.

Figure 2. Forecast Error Variability of Champion Models




## GENERATING THE REPOSITORY AND THE FORECASTS

A simple example demonstrates the process being espoused in this paper. The forecaster needs a model repository. The sections of code below illustrate how a repository is filled. The portion of code below creates two models and saves them in the repository. The first is an ARIMA with seasonal dummy variables and no intercept, named SEASONALDUMMIES. The last is a simple exponential smoothing model, $s_t = \alpha x_t + (1- \alpha)_{st-1}$, named SIMPLE. These two models are premade SAS® models (they come ready to use with SAS® Forecast Studio, so they weren't conjured up by the author). The HPF procedures, such as PROC HPFARIMASPEC and PROC HPFESMSPEC, are available with a Forecast Server license. Since there are over 40 premade time series models available, the program contains the same number of sections to add each model to the repository. The time series forecasting models available in SAS® Forecast Studio are used in this example for simplicity. A forecaster will usually want to fill his\her repository with all potentially viable models, which will include both premade and custom made models.

```
%let model_repo = rep.SAS_System_Models;

PROC HPFARIMASPEC
    MODELREPOSITORY = &model_repo
    SPECNAME = SEASONALDUMMIES
```

```
    SPECLABEL = ""
    SPECTYPE = ARIMA
    SPECSOURCE = FSUI;

    FORECAST SYMBOL = AIR TRANSFORM = NONE NOINT;
    INPUT PREDEFINED = SEASONAL TRANSFORM = NONE;

    ESTIMATE
    METHOD = CLS
    CONVERGE = 0.001
    MAXITER = 50
    DELTA = 0.001
    SINGULAR = 1.0E-7;
run;

PROC HPFESMSPEC
    MODELREPOSITORY = &model_repo
    SPECNAME = SIMPLE
    SPECLABEL = ""
    SPECTYPE = ESM
    SPECSOURCE = FSUI;

    ESM METHOD = SIMPLE TRANSFORM = NONE
    LEVELREST = (0.001, 0.999)
    TRENDREST = (0.001, 0.999)
    DAMPREST = (0.001, 0.999)
    SEASONREST = (0.001, 0.999);
run;
```

Once the repository is filled with the desired models, the practitioner needs a way to select the best models according to each available error metric. A simple macro can do this. First, create a string containing the names of the error metrics and then place the HPFSELECT procedure within the macro. Loop through the string, select each champion model, and output the resulting forecast values into their own table. After the macro is finished running, join the tables together. To get the combined forecast, use simple averaging.

```
%let input_set = Work.Data_TS;
%let Time_ID = Date;
%let Dep_Var = Shipments;
%let interval = month;
%let full_sort_list = Level Date;
%let BY_List = id_desc;
%let Metrics_List =
ADJRSQ|AIC|AICC|AADJRSQ|APC|GMAPES|GMAPE|GMAPPE|GMRAE|GMASPE|MAE|
MAPES|MAPE|MAPPE|MASE|SMAPE|MRAE|MSE|MDAPES|MDAPE|MDAPPE|MDRAE|MDASPE
|MINAPES|RSQUARE|RWRSQ|RMSE|SBC|SSE|UMSE|URMSE;
%MACRO Error_Fitter();
%let iterator = 1;

%do %while(&iterator <= %sysfunc(countw(superq(&Metrics_List), "|")));
    %let Error_Metric = %scan(&Metrics_List, &iterator, "|");

%let output_fcst = %sysfunc(cat(work.&Error_Metric., _Champion));

proc hpfselect rep = PERM.SAS_System_Models name = &Error_Metric.selection;
select criterion = &Error_Metric;
```

```
spec ADDWINTERS AIRLINE ARIMA000011NOINT ARIMA011100NOINT ARIMA012011NOINT
ARIMA022011NOINT ARIMA200100 ARIMA210011NOINT ARIMA212011NOINT DAMPTREND
DOUBLE LINEAR LINEARSEASONALDUMMIES LINEARTREND LINEARTRENDAR LOGADDWINTERS
LOGAIRLINE LOGARIMA000011NOINT LOGARIMA011100NOINT LOGARIMA012011NOINT
LOGARIMA022011NOINT LOGARIMA200100 LOGARIMA210011NOINT LOGARIMA212011NOINT
LOGDAMPTREND LOGDOUBLE LOGLINEAR LOGLINEARSEASONALDUMMIES LOGLINEARTREND
LOGLINEARTRENDAR LOGMEAN LOGRWWD LOGSEASONAL LOGSEASONALDUMMIES LOGSIMPLE
LOGWINTERS LSMADWN LSMDAMP LSMDOUB LSMLIN LSMSEAS LSMSIMP LSMWINT MEAN RWWD
SEASONAL SEASONALDUMMIES SIMPLE SMADWN SMDAMP SMDOUB SMLIN SMSEAS SMSIMP
SMWINT WINTERS;
run;

proc hpfengine data = work.ML_Pract repository = PERM.SAS_System_Models out
= work.&Error_Metric.Champion globalselection = &Error_Metric.selection
print = (select estimates) lead=24;
by &BY_List;
    id &Time_Id interval=&interval;
    forecast &Dep_Var;
run;

%let iterator = %eval(&iterator+1);

%end;

%MEND Error_Fitter;
%Error_Fitter()
```

## EMPIRICAL PERFORMANCE

Two simulated time series are assessed in the following demonstration, which are referred to as Division 1 and SKU 1. Division 1 is a fictional business division, and its' series is a high-level aggregation of many different fictional products. SKU 1 represents a stock keeping unit (SKU) within Division 1, therefore, the series itself has greater volatility. For both, the series are monthly data from January 2009 to mid-summer 2018. The models estimated with this data are trained using the data from 2009 to 2016, therefore, the forecast range is from 2016 to summer 2018. Thirty-one fit statistics are considered when selecting champion models, and the premade SAS® Forecast Studio models are each estimated using the data. The process results in 31 champions. The bar charts in Figure 3 and Figure 4 show the chosen champion models for Division 1 (red) and SKU 1 (blue) respectively, and the frequency of selection. For modeling SKU 1, seven champion models are selected. The most frequently selected model, across the statistics of fit, is the SMADWN, which is selected by around 26% of the error metrics as the champion. For modeling Division 1, seven champion models are also selected. The most frequently selected model, across the fit statistics, is the LINEARSEASONALDUMMIES model, which is selected by over 50% of the error metrics as the champion.

For Division 1, since seven different models are selected as champions across the thirty-one error metrics, the weighted ensemble forecast has the following form:

$$\bar{Y} = \alpha\hat{Y}_1 + \beta\hat{Y}_2 + \delta\hat{Y}_3 + \varepsilon\hat{Y}_4 + \theta\hat{Y}_5 + \tau\hat{Y}_6 + \varphi\hat{Y}_7$$

where $\hat{Y}_1$, $\hat{Y}_2$, $\hat{Y}_3$, $\hat{Y}_4$, $\hat{Y}_5$, $\hat{Y}_6$, and $\hat{Y}_7$ are the forecast values from the seven champion models, ordered from most to least frequently selected. The weights have the following approximate values:

$$\alpha = 0.52,\ \beta = 0.16,\ \delta = 0.16,\ \varepsilon = 0.06,\ \theta = 0.03,\ \tau = 0.03,\ \text{and}\ \varphi = 0.03$$

The resulting equation for the weighted ensemble forecast is as follows:

$$\bar{Y} = 0.52\hat{Y}_1 + 0.16\hat{Y}_2 + 0.16\hat{Y}_3 + 0.06\hat{Y}_4 + 0.03\hat{Y}_5 + 0.03\hat{Y}_6 + 0.03\hat{Y}_7$$

Clearly from the bar charts, depending on the fit statistic used as the selection criteria, a wide variety of champions will be selected. Even in the case of Division 1, where one champion model is selected over 50% of the time, the likelihood that a practitioner happens to select the supposed best champion (out all champions) comes down to a coin flip.

Figure 3. Division Level Champions

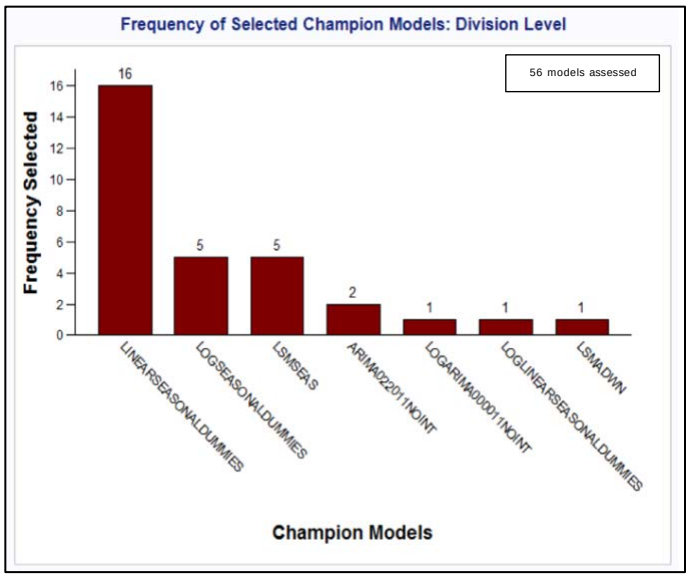
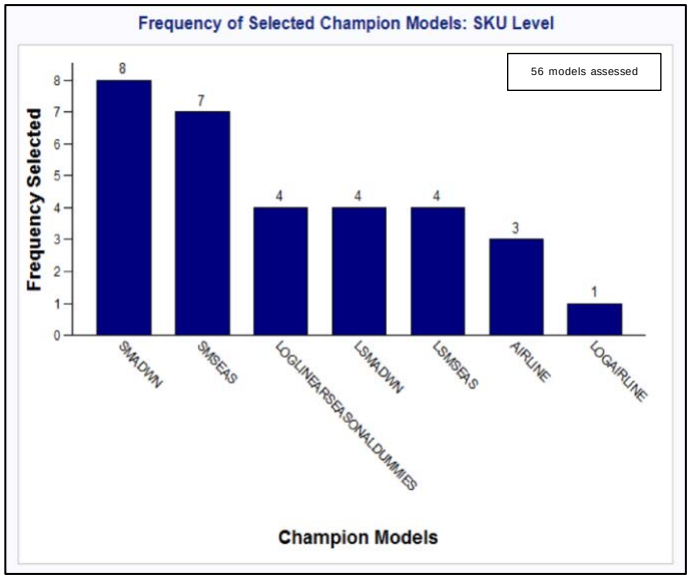



Figure 4. SKU Level Champions




The two line graphs in Figure 5 and Figure 6 plot the forecast errors of the three most frequent champions for Division 1 and SKU 1 respectively, along with the combined forecast. The combined forecast isn't the most accurate, but it is never the worst. It outperforms the others in terms of stability (less variance), and is buffered against large swings in the forecast. For example, looking at the plot for SKU 1, the most frequent model, SMADWN, misses greatly between April 2017 and October 2017. The combined forecast,

instead, is buffered against this under-projection. To reiterate, the practitioner has a high likelihood of choosing SMADWN due to it being the most frequent champion, which would result in a poor forecast. If she\he utilizes the process espoused in this paper, she\he will avoid this problem and will be more certain of the outcome.

Figure 5. Division Level Forecast Errors

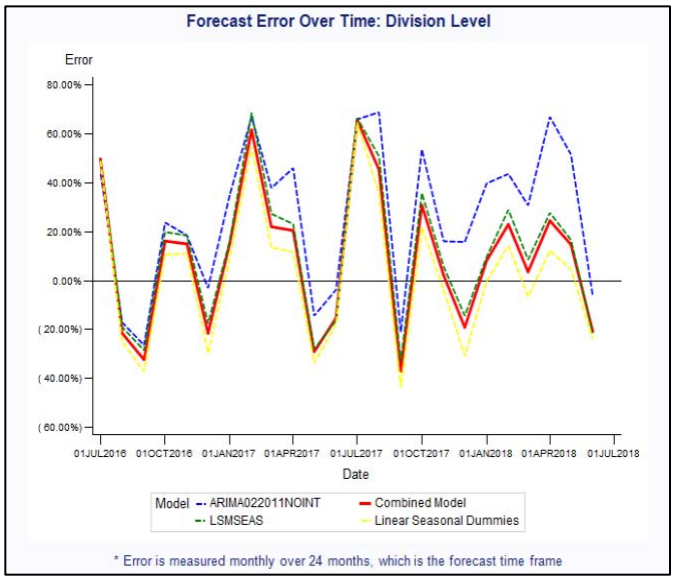
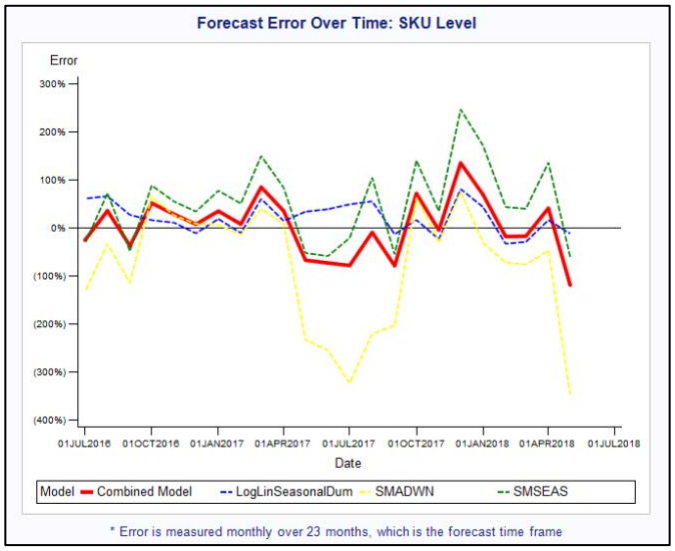


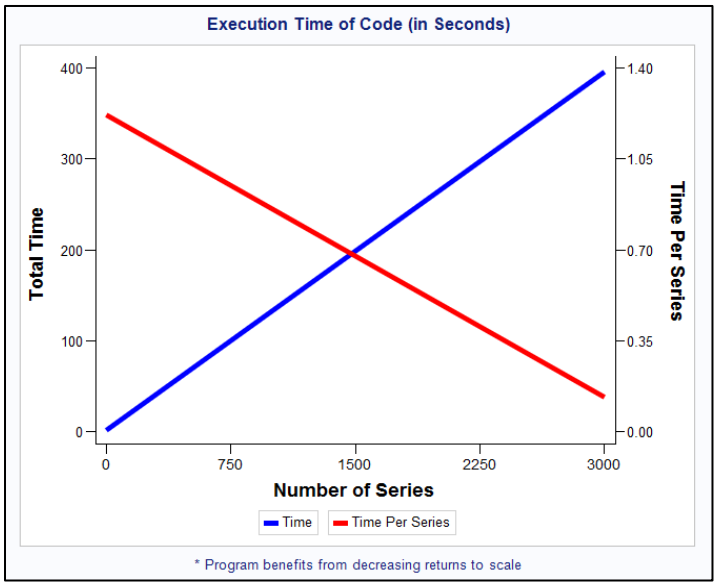

Figure 6. SKU Level Forecast Errors




Not only does this method result in a superior forecast, the SAS® program required to produce the results requires very little time to execute. Therefore, a better result is produced without a time trade off. In Figure 7, a line graph shows the execution times when producing an ensemble forecast for a single series and when producing ensemble forecasts for 3,000 time series. The relative time required for the program to execute decreases significantly as more ensembles are produced. This indicates that the program benefits from significant returns to scale! Executing the program to produce one ensemble forecast for a single time series takes approximately 1.22 seconds to complete. When 3,000 ensemble forecasts are produced for 3,000 time series, the program takes approximately 396 seconds to execute. This means that, per generated ensemble forecast, the time required goes from

1.22 to 0.13 seconds. Therefore, not only does this process take very little time to execute and produce improved forecasts, but it is also highly efficient and benefits from returns to scale.

Figure 7. Program Run Time




## CONCLUSION

Reducing random variation is fundamental in modern business forecasting. My new technique accomplishes this by ensembling the champion forecasts selected from all available error metrics and weighting them on their frequency of selection across the error metrics. This method smooths the random variability in forecast accuracy and is less volatile and more robust than choosing one fit statistic to select and deploy champion forecasts.

## REFERENCES


Blair, E., Leonard, M., and Elsheimer (2012). Combined forecasts: what to do when one model isn't good enough. *SAS Institute Inc. SAS Global Forum 2012*.

SAS High-Performance Forecasting 4.1: User's Guide. *SAS Institute Inc. 2011*. Cary, NC: SAS Institute Inc.

Zou, H., and Yang, Y. (2004) Combining time series models for forecasting. *International Journal of Forecasting, Volume 20*(1), Pages 69-84.

Lean, Y., Shouyang, W., Lai, K., and Nakamori, Y. (2005). Time series forecasting with multiple candidate models: selecting or combining? *Journal of Systems Science and Complexity, Volume 18*(1)


## ACKNOWLEDGMENTS


I would like to sincerely thank Jay Laramore, Analytical Consultant at SAS, Scott Ryan, Senior Manager of Advanced Analytics at Hanesbrands Inc., Scott Leslie PhD, Manager of Advanced Analytics at MedImpact Healthcare Systems, Inc., and Ethan Gordon, philosophy graduate student at Texas Tech University for their helpful suggestions, invaluable recommendations, and constant support. I would also like to thank Jihoon Lee, Manager of Customer Analytics at Lowes, for being the first person to encourage me to write this paper.

## CONTACT INFORMATION

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

Zachary Blizard
zachblizz@yahoo.com