

# Various Preprocessing and Postprocessing Methods to Improve Forecast

Dissertation

submitted in partial fulfillment of the requirements  
for the degree of

**Master of Technology**

by

**Pankaj E. Gulhane**

(Roll no. 03329007)

under the guidance of

**Prof. V. M. Gadre**

**Prof. Bernard Menezes**



Kanwal Rekhi School of Information Technology

Indian Institute of Technology Bombay

2005



# Dissertation Approval Sheet

This is to certify that the dissertation entitled  
**Various Preprocessing and Postprocessing Methods  
to Improve Forecast**

by

**Pankaj E. Gulhane**

(Roll no. 03329007)

is approved for the degree of **Master of Technology**.

---

Prof. V. M. Gadre

Prof. Bernard Menezes

(Supervisors)

---

Prof. Madhu Belur

(Internal Examiner)

---

Prof. D. Manjunath

(External Examiner)

---

Prof. S. V. Kulkarni

(Chairperson)

Date: \_\_\_\_\_

Place: \_\_\_\_\_



---

# INDIAN INSTITUTE OF TECHNOLOGY BOMBAY

## CERTIFICATE OF COURSE WORK

This is to certify that **Mr. Pankaj E. Gulhane** was admitted to the candidacy of the M.Tech. Degree and has successfully completed all the courses required for the M.Tech. Programme. The details of the course work done are given below.

Sr.No.	Course No.	Course Name	Credits
<b>Semester 1 (Jul – Nov 2003)</b>			
1.	HS699	Communication and Presentation Skills (P/NP)	4
2.	IT605	Distributed Systems	6
3.	IT619	IT Foundations Laboratory	10
4.	IT623	Foundation course of IT - Part II	3
5.	IT653	Network Security	3
6.	IT694	Seminar	4
7.	MA411	Introduction to Probability Theory	6
<b>Semester 2 (Jan – Apr 2004)</b>			
8.	CS602	Applied Algorithms	6
9.	CS604	Combinatorics (Audit)	6
10.	EE678	Wavelets	6
11.	IT606	Embedded Systems	6
12.	IT680	Systems Laboratory	6
<b>Semester 3 (Jul – Nov 2004)</b>			
13.	CS681	Performace Analysis	6
14.	HS617	Intellectual Property Rights (Institute Elective)	6
<b>M.Tech. Project</b>			
16.	IT696	M.Tech. Project Stage - I (Jul 2004)	18
17.	IT697	M.Tech. Project Stage - II (Jan 2005)	30
18.	IT698	M.Tech. Project Stage - III (Jul 2005)	42

I.I.T. Bombay

Dy. Registrar(Academic)

Dated:



## **Abstract**

We study the effect of decomposing a series into multiple components and performing forecasts on each component separately. The focus here is on sales data - most of the series considered display both seasonality and trend. Hence the original series is decomposed into trend, seasonality and an irregular component. Multiple forecasting ‘experts’ are used to forecast each component series. These ranges from different feedforward neural network topologies to Holt-Winter, ARIMA (of various orders) and double exponential smoothing. We compare the forecast errors with and without decomposition. We study the result of combining using the mean/median of all expert forecasts. Since our space of composite experts runs into the thousands, we experiment with more limited cardinalities using greedy elimination and best expert pair.



# Contents

<b>Abstract</b>	<b>i</b>
<b>List of figures</b>	<b>v</b>
<b>List of tables</b>	<b>vii</b>
<b>1 Introduction</b>	<b>1</b>
<b>2 Background Study and Literature Survey</b>	<b>5</b>
2.1 Time Series Analysis . . . . .	5
2.1.1 Types of time series . . . . .	5
2.1.2 Mathematical Models . . . . .	7
2.2 Wavelets . . . . .	8
2.2.1 Locally Stationary Wavelet . . . . .	8
2.2.2 Variance Estimation . . . . .	9
2.3 Error Measures . . . . .	10
2.4 Decomposition . . . . .	13
2.4.1 Classical . . . . .	14
2.4.2 Seasonality Adjustment . . . . .	17
2.4.3 Multiscale Decomposition . . . . .	18
2.5 Multiple Experts . . . . .	19
2.5.1 Time Series Similarity . . . . .	19
2.5.2 Clustering . . . . .	21
2.5.3 Combining Methods . . . . .	23
<b>3 Our Approach</b>	<b>27</b>
3.1 Decomposition . . . . .	29

---

3.2	Combining Forecasts . . . . .	29
3.2.1	Brute Force . . . . .	30
3.2.2	Clique Approach 1 . . . . .	31
3.2.3	Clique Approach 2 . . . . .	31
3.2.4	Greedy Algorithm . . . . .	32
<b>4</b>	<b>Results</b>	<b>35</b>
4.1	Decomposition . . . . .	35
4.1.1	Improvement using Decomposition . . . . .	35
4.1.2	Multiplicative model . . . . .	35
4.1.3	Decomposition Method . . . . .	36
4.2	Similarity Measure . . . . .	38
4.3	Combining Methods . . . . .	39
4.3.1	Mean and Median . . . . .	39
4.3.2	Expert Comparison . . . . .	41
4.3.3	Brute Force . . . . .	43
4.3.4	Clique Approaches . . . . .	43
4.3.5	Greedy . . . . .	46
4.3.6	Minimizing Maximum MAPE . . . . .	47
4.4	Error Measure . . . . .	49
<b>5</b>	<b>Conclusion and Future Work</b>	<b>51</b>
	<b>Bibliography</b>	<b>53</b>
	<b>Publications</b>	<b>57</b>

# List of Figures

2.1	Algorithm for decomposing time series into trend, season and irregular components . . . . .	16
3.1	Clique formation algorithm . . . . .	32
3.2	Greedy Elimination Algorithm for combining various forecasts . . . . .	33
3.3	Greedy Accretion Algorithm for combining various forecasts . . . . .	34
4.1	Abraham . . . . .	40
4.2	Sweet . . . . .	40
4.3	Red . . . . .	40
4.4	Wine . . . . .	40
4.5	Algorithm for combining two forecast using wavelet method . . . . .	42
4.6	Mean of MAPE of various subset of a clique . . . . .	46
4.7	Progress of greedy elimination algorithm using mean and median(series: fortif) . . . . .	48
4.8	Progress of greedy elimination algorithm using mean and median (zoomed in) when number of experts go down from 200 to 1 (series: fortif) . . . . .	49
4.9	Best, mean and the worst MAPE for combination of $k$ experts . . . . .	49



# List of Tables

3.1	Time series used for analysis in this paper . . . . .	28
4.1	Improvement in MAPE using decomposition method, *indicates decomposition was used . . . . .	36
4.2	MAPE Comparison of various decomposition models . . . . .	36
4.3	MAPE Comparison of various decomposition methods . . . . .	37
4.4	MAPE using wavelet decomposition method, *indicates decomposition was used . . . . .	38
4.5	Distance between two time series using time domain method . . . . .	39
4.6	Distance between two time series using Wavelet method . . . . .	39
4.7	Table compaing the forecast of three experts with their combined forecast .	41
4.8	Table compares two methods with table 4.7 . . . . .	42
4.9	MAPEs using holt-winter and mean/median of all combinations of experts	43
4.10	Best model and best pair mapes using 2400 experts . . . . .	44
4.11	Comparison of MAPE of various subset sizes . . . . .	45
4.12	Comparison of two clique approaches, number in bracket (best clique/max clique size) . . . . .	45
4.13	MAPE of various clique sizes. number in bracket indicate maximum clique size we got for that series . . . . .	47
4.14	Comparison of various greedy approaches, *indicate that median was used for combining various forecasts. Figure in bracket indicates the number of experts that were present in the subset selected. . . . .	48
4.15	Inability of MAPE to reflect improvement in seasonality prediction. 1 <sup>st</sup> and 3 <sup>rd</sup> column shows seasonality MAPE whereas 2 <sup>nd</sup> and 4 <sup>th</sup> column shows overall MAPE after combining all three components . . . . .	50



# Chapter 1

## Introduction

Sales forecasting is an important part of supply chain management - both at the retailer end and at the distributors, manufacturers and suppliers. Timely and accurate sales forecasts are crucial in bridging the gap between supply and demand, thereby decreasing inventory holding costs while maintaining a negligible probability of stock-out. Much work in sales forecasting has centered around the comparison of linear statistical models such as ARIMA[1] [2], the Holt-Winter approach (exponential smoothing of level, trend and seasonality)[3] and the use of artificial neural networks (ANNs). These methods are becoming more sophisticated day by day so as to capture more and more features from a time series.

Forecasting problem can be stated as predicting future values  $x_{t+h}$ , given a part of a time series  $x_t, x_{t-1}, \dots, x_{t-w+1}$  (called pattern), where  $w$  is the length of the window on the time series and  $h \geq 1$  is called the prediction horizon.

In parallel to the development of forecasting methods, researcher is also trying various pre-processing and post-processing techniques to improve forecasts. One of the goals of this study is to investigate the effect of a specific form of pre-processing - series decomposition. Very recent work in this area [4] confirms our findings that de-trending and deseasonalizing the data greatly help in improving forecasts.

For deseasonalizing, we can use seasonality adjustment methods. Their need is very well known and very well appreciated. Seasonality adjustment is useful in getting the clear picture of the state of time series (e.g. economy, or demand state). But there has been very few efforts in using these techniques for forecasting. If we dig into the seasonality adjustment methods, we will find that it can be effectively used for separating trend, cycle, seasonal and irregular components. These component series can be treated independently and forecasting method can be applied on each of these series to get final forecast.

Similarly, decomposition of a series at various levels of detail can be useful. This approach is called multi-resolution analysis. This is especially useful when we don't know the seasonality period. Though this method is able to decompose the series into trend like and seasonal like components, getting a trend/season is really difficult. Even the detail level components may not be stationary or near to stationary. With prior knowledge of seasonality period, we can deploy seasonality adjustment methods to get trend, season, and irregular component nearly unaffected by each other and we can also get irregular component nearly stationary.

We know that a particular kind of series can be analyzed in better way by a particular method than that of any other method. When forecast for new series is required, then the most appropriate method should be chosen. Giving an example, when we have decomposed series components, then trend can be predicted very well by exponential smoothing method, but we can not use exponential smoothing method for forecasting irregular component. To address this problem, model selection criteria are used. There has been a lot of research going into this area but there is no one solution which will work in all conditions. However it seems clustering is making an impact in this area for selecting a model for a given series. In clustering, similar type of series are clustered together and the best method for that cluster is applied.

To make clustering method work, we need to know the similarity measure between two time series. One of the similarity measure is taking difference square of absolute values of two time series for a given period of time. But this time measure is inadequate, many time it triggers false alarms. It has been advisable to match the attributes of time series rather than their absolute values for similarity measure[5]. Measure which seems to satisfy this requirement is similarity measure using wavelets.

However, only model selection is not enough for getting better forecast. It is known that model selection is often unstable and may cause an unnecessarily high variability in the final estimation/prediction[6]. So final estimate/prediction/forecast should consider of combining different forecasts from different forecasting methods/models. The quest for combining various forecasts from different methods in an optimal way is almost three decade old(Clemen[7] 1989). To improve accuracy of combining method, past knowledge of performance of various participating methods can be used.

Our work differs from existing work in the way we choose and combine [7][8] a mul-

titude of experts - both neural and statistical to forecast each component series. The neural experts are feedforward ANNs with different topologies and the statistical experts are ARIMA models of assorted orders. We perform forecasts for each series separately which are then used to derive the forecast for the original series.

We have discussed above issues in detail, in the following chapters. Chapter 2 covers necessary background for the rest of the thesis. It talks briefly about time series analysis, wavelets and error measures. We have also discussed about the decomposition and combining techniques to improve the forecast in that chapter. Our approach is explained in the chapter 3. It discusses mainly the approach which should be followed to get better forecast. Results of various experiments are included in the chapter 4. Final chapter talks about the conclusion and the future work.



# Chapter 2

## Background Study and Literature Survey

### 2.1 Time Series Analysis

A time series is a sequence of observations taken sequentially in time. An intrinsic feature of time series is that, typically, adjacent observations are dependent. Time series analysis is concerned with the technique of analysis of this dependence.

The main objective of the time series analysis is to model a process, which is generating the data, to provide compact description and to understand the generating process. To allow for the possibly unpredictable nature of the future observations, selection of the probability model for the data is very important. We can define a **time series model**[2] for the observed data  $\{x_t\}$ , to be a specification of the joint distribution of a sequence of random variables  $\{X_t\}$  of which  $x_t$  is postulated to be a realization.

#### 2.1.1 Types of time series

Observations within time series can take either continuous values or values from a fixed set. If values taken by observations are from the fixed set, then that time series comes under *categorical time series*. Further time series can be either deterministic or statistical. In statistical time series, future value can be described only in terms of probability distribution. A statistical phenomenon that evolves in time according to probabilistic laws is called a *stochastic process*.

### 2.1.1.1 Stationary Series

In this class of stochastic process, process is assumed to be in *statistical equilibrium*. A stochastic process is called *strictly stationary* if its properties are unaffected by the shift in time origin, i.e. joint probability distribution associated with  $m$  observations at any point in time with any shift  $k$  is same[1]. Depending on the value of  $m$ , process can be first( $m = 1$ ) or second( $m = 2$ ) order stationary.

**First order stationary:** A time series is a first order stationary if the expected value of  $X(t)$  remain same for all  $t$ . A process is a first order stationary when series is trend and seasonality free.

**Second order stationary:** A time series is a second order stationary if it is first order stationary and covariance between  $X(t)$  and  $X(s)$  is a function of lag  $(t - s)$  only.

Loosely speaking,  $\{X_t\}$  is said to be stationary if its statistical properties are similar to those of the time-shifted series  $\{X_{t+h}, t = 0, \pm 1, \dots\}$  for each integer  $h$ . If both mean ( $\mu_X(t)$ ) and covariance function ( $\gamma_X(t + h, t)$ ) of  $\{X_t\}$  are independent of  $t$  for each  $h$ , then  $\{X_t\}$  is a *weakly stationary process*.

The auto-covariance function of a process  $\{X_t\}$  is denoted by  $\gamma_X(r, s) = Cov(X_r, X_s)$ , and for stationary process, it depends on the distance between  $r$  and  $s$  only, i.e.  $\gamma_X(r, s) = \gamma_X(|r - s|)$ . As this function is independent of the reference point in the time, we can say that the auto-covariance function of a stationary process is homogeneous over time.

One very important property of a stationary process( $x_t$ ) it that, any process  $L_t$  obtained by linear operation on stationary process ( $x_t$ ) for fixed  $n$  terms is also stationary. This leads to an elegant theory from the point of view of both estimation and forecasting.

### 2.1.1.2 Categorical Time Series

Categorical data is identified by category rather than by real value. The data set can be partitioned into a number of categories. Each element in the data set belongs to exactly one category. For example, gender data is categorical data. Each item in a data set of people can be placed in one of the categories from the set  $\{\text{Male}, \text{Female}\}$ .

Let  $C = \{c_1, \dots, c_k\}$  be a set of states that can be taken by  $X_t$ , for each  $t = 0, \pm 1, \dots$  in such a way that  $p_j = P(X_t = j) > 0$ . Then  $X_t$  can be called as categorical time series [9]. We can obtain real time series by associating a real value number with every category. i.e. assume  $\beta = (\beta_1, \dots, \beta_k)' \in R^k$  to be such that  $X_t(\beta)$  is the real time series. Here

analysis will be performed on the real series and will depend on the  $\beta$  chosen.

Categorical time series can also be useful when, we are interested in relative value (e.g high or low) than their absolute value.

## 2.1.2 Mathematical Models

Few mathematical models, which are commonly used, are listed below. First section describes models which are applied to the stationary series, whereas another section discusses about *Holt-Winter* method which is applied to a series containing both trend and seasonality. Later model is more relevant to demand series as it contain both trend and seasonality.

### 2.1.2.1 Stochastic Models

- **Autoregressive models(AR):** In this model, current value of a process is expressed as a finite, linear aggregate of previous values of the process. Let us denote the values of a process by  $z_t$ , then AR process of order  $p$  can written as

$$\tilde{z}_t = \phi_1 \tilde{z}_{t-1} + \phi_2 \tilde{z}_{t-2} + \dots + \phi_p \tilde{z}_{t-p} + a_t$$

where  $a_t$  is shock(error term).

- **Moving Average models(MA):** In this model,  $\tilde{z}_t$  is linearly dependent on finite number of previous  $a$ 's. Thus

$$\tilde{z}_t = a_t - \theta_1 a_{t-1} - \theta_2 a_{t-2} - \dots - \theta_q a_{t-q}$$

is called moving average(MA) process of order  $q$ .

- **Mixed autoregressive- Moving Average models(ARMA):** To achieve greater flexibility, we can include both AR and MA terms in the model. Thus resultant model becomes.

$$\tilde{z}_t = \phi_1 \tilde{z}_{t-1} + \phi_2 \tilde{z}_{t-2} + \dots + \phi_p \tilde{z}_{t-p} + a_t - \theta_1 a_{t-1} - \theta_2 a_{t-2} - \dots - \theta_q a_{t-q}$$

### 2.1.2.2 Holt Winter Method

The basic Holt-Winter forecasting method with multiplicative seasonality (exponential smoothing of level ( $S_t$ ), trend ( $T_t$ ) and seasonal index ( $I_t$ )) is described by

$$S_t = \alpha(D_t/I_{t-p}) + (1 - \alpha)(S_{t-1} + T_{t-1})$$

$$T_t = \beta(S_t - S_{t-1}) + (1 - \beta)T_{t-1}$$

$$I_t = \gamma(D_t/S_t) + (1 - \gamma)I_{t-p}$$

Here  $p$  is the number of observation points in a cycle ( $p = 4$  for quarterly data).  $\alpha, \beta$  and  $\gamma$  are the smoothing constants. The forecast at time  $t$  for time  $t + i$  is  $(S_t + i \times T_t)I_{t-p+i}$ .

## 2.2 Wavelets

A wave is usually defined as an oscillating function of time or space, such as a sinusoid. A wavelet is a small wave, which has its energy concentrated in time to give a tool for the analysis of transient, non-stationary, time-varying phenomena. Wavelet allows simultaneous time and frequency analysis with a flexible mathematical foundation.

Formally, we can define wavelet to be any function ( $\psi \in L^2$ ), which satisfy following admissibility condition.

$$\int_{-\infty}^{\infty} \frac{|\hat{\psi}(\omega)|^2}{|\omega|} d\omega < \infty$$

Wavelet system is a set of building blocks to construct or represent signal or function. It is two dimensional expansion set (usually basis) for some class of one or more dimensional signal. Formally, if  $\psi_{j,k}$ ,  $j, k = 1, 2, \dots$  is the wavelet set, where  $j, k$  are the scale (dilation) and location (translation) parameters respectively, then  $f(t)$  can be written as

$$f(t) = \sum_j \sum_k a_{j,k} \psi_{j,k}(t)$$

for some set of coefficients  $a_{j,k}$ , called *discrete wavelet transform (DWT)* of  $f(t)$ .

Wavelets can decompose signals in terms of scale of details. Generally at first level we decompose signal into two components, high frequency and low frequency component. As high frequency component can be resolved easily in time, we don't have to process it again to get more information out of it. About low frequency which can not be still resolved in time, we again decompose it using high pass filter and low pass filter. If we apply HPF and LPF to high frequency component also, we will get wavelet packet transform. Following sub-sections discusses about the most widely used features of wavelets.

### 2.2.1 Locally Stationary Wavelet

Wavelet transform[10] decomposes series into various scales, but at each scale, there are only half the number of points as compared with the number of points present in scale

immediately lower than this, where lowest scale indicate the original series. This is called as decimated decomposition, which is very useful in compression. However this useful property renders wavelet transform to be shift variant. To make wavelet transform shift invariant, we use *Redundant, or Non-Decimated, or Stationary, or Maximal Overlap(MODWT)* wavelet transform. MODWT is an attempt to get away from effects attributed to the choice of a starting time by essentially including all possible placements of averaging interval. In this transform we have same number of points at every scale(or level). This property is widely used in time series analysis using wavelet transform[11]. The most simple stationary wavelet transform is *à trous*, this transform is variant of Haar transform in stationary wavelet transform domain.

### 2.2.2 Variance Estimation

Estimating scale characteristic of process which do not show location dependency, we need to analyze long term process. The variance which can be really useful to long term process is *Allan variance*. In such processes it is possible that, autocorrelation of the process will decay slow, such that effect will persist over long time scale.

*Allan variance*[12]  $\sigma_X^2(\tau)$  at a particular scale  $\tau \in Z$  is a measure of how averages, over window length  $\tau$ , changes from one period to the next. If  $\bar{X}_t(\tau) = 1/\tau \sum_{n=0}^{\tau-1} X_{t-n}$ , then

$$\sigma_X^2(\tau) = 1/2E(|\bar{X}_t(\tau) - \bar{X}_{t-\tau}(\tau)|^2).$$

It can be shown that, Allan variance is proportional to the Haar wavelet variance. Haar wavelet variance is the scale variance based on the discrete haar wavelet transform. Let  $\{\hat{d}_{j,k}\}$  denote wavelet coefficient of the signal  $\{X_t\}_{t=0,\dots,T-1}$ , and  $j = -1, \dots, -\lg(T)$ , where more negative is more coarser. Let high pass filter coefficient  $\{g_{(k)}\}$  of the Haar transform is  $\{1/\sqrt{2}, -1/\sqrt{2}\}$ . Then for  $j = -1$ , at scale  $\tau_j = 2^{-j-1}$ ,

$$\hat{d}_{-1,k} = \frac{1}{\sqrt{2}}(X_{2k+1} - X_{2k}), \quad k = 0, \dots, T/2 - 1$$

and it can be seen that  $\text{var}\{\hat{d}_{j,k}\} = \sigma_X^2(1)$ , if we generalize this, we get

$$\text{var}\{\hat{d}_{j,k}\} = E\hat{d}_{j,k}^2 = \tau_j \sigma_X^2(\tau_j).$$

Using above equation, an unbiased estimator for Allan variance, also called as non-

overlapped estimator, is the normalized sum of the squared wavelet coefficients

$$\hat{\sigma}_X^2(\tau_j) = \frac{2}{T} \sum_{k=0}^{\frac{T}{2^j}-1} \hat{d}_{jk}^2.$$

Above estimator has one property that each data point  $X_t$  contribute to exactly one coefficient  $\hat{d}_{jk}$ . We can see that one can improve above estimator by summing over not just  $T/2$  values of these time series values, but over all  $T - 1$  possible one. Then the resulting estimator will have smaller variance and also possesses independence with respect to the choice of the origin of the series  $X_t$ . This is called *Maximal-overlap* estimator, denoted by  $\tilde{\sigma}_X^2(\tau_j)$ , is based on the non-decimated wavelets transform. More general wavelet can be used in place of Haar wavelet in variance estimator. It can be seen that, it can be useful to use wavelet other than Haar, because different wavelet cover slightly different frequency ranges and posses differing phase behavior.

The importance of the *scale-variance* is that, it permits a new decomposition of the process variance. The fascinating thing is, we can decompose the variance of a process into quantities which measure the fluctuation separately scale by scale:

$$\text{var}\{X_t\} = \frac{1}{2} \sum_{j=-\infty}^{-1} \sigma_X^2(\tau_j) = \sum_j \text{var}\{\tilde{d}_{jk}\}/2\tau_j.$$

## 2.3 Error Measures

Error measure plays an important role in calibrating and refining forecasting model/method. This calibration helps analyst to improve forecasting method. Comparisons of errors across series typically involve many methods and many series. Because use of multiple measures can be cumbersome, a single error measure is desirable. The choice of an error measure may vary according to the situation, number of time series available and on whether the task is to select the most accurate method or to calibrate a given model.

There exist various error measure, every error measure tries to capture different aspect of the loss function. Depending on the aspect of interest of loss function, selection of error measure and hence the forecasting method can be selected. For example, in SCM, if distributor want to optimize on stock present in inventory, then maximum error make more sense than average absolute error, to make sure that inventory is not getting understocked or overstocked. Above example clearly shows that selection of error measure is more managerial decision.

In forecasting arena, quest is for finding best fit for the actual series using various forecasting methods, so researchers were using *Root Mean Square Error*. But *RMSE* has some limitations which made researcher to search for new error measures. Because of the quest for the unit-free error measure which is one of the main limitations of *RMSE*, many error measures were introduced. Empirical comparisons of these various error measures have been done by Armstrong[13] to compare forecasting methods. He recommended Geometric Mean of the Relative Absolute Error *GMRAE* to calibrate a model for a set of time series.

There exist many error measure depending on the situation. Below is the list of few error measures which are either popular or recommended.

1. Mean Squared Error(*MSE*) :

$$MSE = \sum_{j=1}^N (observation_j - prediction_j)^2 / N$$

This is one the most widely used error measure till late 80s, although it is not unit-free measure. The problem with this measure is, it depend on the values of series, i.e. it is scale dependent, which renders it useless for comparing different forecasting methods across various series. Scale dependency make this measure unreliable for model calibration.

2. Root Mean Squared Error(*RMSE*) :

$$RMSE = \sqrt{\sum_{j=1}^N (observation_j - prediction_j)^2 / N}$$

This measure is a variation of *MSE*, so it posses same advantage and disadvantage as that of *MSE*.

3. Normalized Mean Squared Error(*NMSE*) :

$$\sum_{j=1}^N (observation_j - prediction_j)^2 / \sum_{j=1}^N (observation_j - mean)^2$$

This is an effort to make *MSE* unit-free.

4. Mean Absolute Percentage Error(*MAPE*) :

$$MAPE = 100 \times \frac{\sum_{i=0}^N (|forecasted_i - actual_i|) / actual_i}{N}$$

This error measure is the most widely used unit-free error measure. A disadvantage of *MAPE* is it is relevant only for ratio-scaled data (i.e. data with a meaningful zero). Another disadvantage of the *MAPE* is that it puts a heavier penalty on forecasts that exceed the actual than on those that are less than the actual. For example, the MAPE is bounded on the low side by an error of 100%, but there is no bound on the upper side. Another disadvantage is, it is not commutative. Strongly rejected by fields on statistical grounds.

5. Unbiased Absolute Percentage Error (*UAPE*) aka Symmetric Mean Absolute Percentage Error(*SMAPE*):

$$UAPE = \frac{100}{N} \times \sum_{i=0}^N \frac{|forecasted_i - actual_i|}{(actual_i + forecasted_i)/2}$$

An adjustment to the denominator of the APE so that the average of the actual and the forecast values is used instead of the actual value. The UAPE is constrained to be between 0 and 200(only for +ve values), The error is symmetric with respect to the scale of the errors. *UAPE* also avoids problems associated with dividing by small numbers when the actual is close to zero. Its unbiasedness, simple summarization makes it worthy for the further research. One disadvantage is, although the *UAPE* is symmetric when actual and forecast values are interchanged, it in fact creates a new problem of asymmetry which is more likely to be of practical concern than the problem resulting from the interchange, as same amount of error in either direction of actual value will give us two different *UAPE* values, which forfeits the purpose of symmetric APE for errors [14].

6. Median Absolute Percentage Error (MdAPE): This measure is almost similar to the *MAPE*, but in *MAPE* mean is used for summarization whereas in *MdAPE*, median is used for summarization across series. Median can remove higher as well as lower than the middle value, reducing the effect of outliers. Though outliers can be trimmed off by assigning upper and lower limits, but still proper choice of limits become the most concern issue. One disadvantage of median summarization is lack of sensitivity.
7. Relative Absolute Error(*RAE*) : This approach employs relative error and compare the forecast error from a given model against those from another model. Most of the

the time, *Random Walk* is used as alternative model because of its simplicity. One method of calculating RAE is to divide absolute error of the proposed model by the absolute error of the random walk. For summarizing across series, we can take either arithmetic mean (*AMRAEs*) or Geometric mean (*GMRAEs*) or median (*MdRAE*) of *RAEs*. This measure is generally used for comparison over a small set of time series. One disadvantage is that it can not be used for decision making [15].

8. Direction Variation Symmetry (*DVS*) :

$$DVS = (1/(N - 1)) \times \sum_{j=2}^N \Psi(\Delta actual_j \times \Delta forecast_j)$$

where

$$\Delta actual_j = actualValue_j - actualValue_{j-1},$$

$$\Delta forecast_j = forecastValue_j - forecastValue_{j-1}$$

and  $\Psi$  is heavy-side function, i.e.  $\Psi(x) = 1$  if  $x > 0$  and  $\Psi = 0$  otherwise. This can be useful, when we are interested in the direction of the future value, than the future value itself.

9. Mean Absolute Deviation (*MAD*) : This error measure is extensively used in inventory control as it is closely related to decision making.

## 2.4 Decomposition

A time series( $x_t$ ) can be seen as composition of many individual component time series. Some components out of these, can be predictable whereas other components may be almost random which can be difficult to predict. Decomposing a series into such components enables us to forecast better. These component series can catch different aspects of time series which can be predicted more accurately if analyzed separately. Depending on the aspects caught by different component series, there exist various decomposition methods.

For time series forecasting following methods of decomposition are used. Each decomposition method listed down is a research area in itself.

### 2.4.1 Classical

This approach make an assumption that any time series  $x_t$  can be represented by combination of four components i.e. *trend component*( $t_t$ ), *cyclic component*( $c_t$ ), *seasonal component*( $s_t$ ), and *irregular component*( $i_t$ ). Informally, these components can be described as follow:

- Trend : In this context, trend does not imply a monotonically increasing or decreasing series but simply the lack of a constant mean. That is, different sections of a series may have quite different sample means indicating that the population mean is time dependent.
- Cycle : It refers to patterns, or waves, in the data that are repeated after approximately equal intervals with approximately equal intensity. Period of repetition is larger than seasonal period.
- Season : It refers to a cycle of fixed period ( e.g. weekly, monthly, yearly etc.).
- Irregular : It refers to variation not covered by the above.

The usual decomposition into trend, cycle, seasonal and irregular component was motivated mainly by business analysts, who wanted to have information about the actual and historical situation of the business cycle. One important part is estimation and removal of the seasonal component to get a clearer picture of the state of time series. Another advantage of decomposition is that it gives us irregular component as an approximately stationary series, and there exist many methods which can perform better on stationary series[1] in terms of forecast accuracy. Many time trend component is mixed with the cycle component to simplify the model. From this point onward we will call trend component as combination of both trend component and cyclic component unless and until specified.

Above four components can combine in all possible ways, but following are the most popular models for modeling time series i.e. multiplicative or additive form

$$x_t = t_t \times c_t \times s_t \times i_t$$

OR

$$x_t = t_t + c_t + s_t + i_t$$

OR

$$x_t = t_t \times c_t \times (s_t + i_t)$$

All relations are valid and one of them is applicable, depending on the type of time series we are dealing with. Correct relationship (additive or multiplicative or any other) for a given time series can be found out by visual inspection. Though most of the times, for demand series, we use multiplicative model. This choice is influenced by the statement made by Brown in 1959, stating that ‘you will be very likely to find that the standard deviation of demand is nearly proportional to the total annual usage, or to the average monthly usage’. This statement was verified by Snyder[3]. If we take trend at point  $t$  as mean of demand  $x_{t-6}, \dots, x_{t+6}$  in monthly series, then above argument clearly suggest trend in demand time series is multiplicative in nature.

Identification of the system is very important as it directly affects the forecast accuracy. Some experiments were carried by Gardner[16] in which he found that seasonal additive model was performing better than seasonal multiplicative model for demand series for the store, and seasonal additive model was more robust to outliers than seasonal multiplicative model. Seasonal adjustment is not always better, sometimes series is influenced by seasonality, that does not mean that season can be identified and estimated using seasonal adjustment method. If this is the case then it’s better not to perform seasonal adjustment.

We can convert the multiplicative model into additive by taking log of the series. Though all statistical properties of the original series may no longer hold, it gives us the opportunity to exploit the benefits of additive model. If we want to exploit some statistical properties of the original series, we have to apply anti-log over log transformed decomposed series.

When we separate trend component, seasonal component and irregular component, then irregular component is called *detrended seasonally adjusted series*. There are many methods for seasonality adjustment which can be used for decomposition of series into various components. These seasonality adjustment methods exist since many years but still there are many questions regarding reliability/efficiency of these methods.

Now we will formally define the terms and will look into various decomposition methods

- $t_t$  : This can be called as non-periodic components in time series

- $c_t$  : This is periodic low frequency in time series. It's period is more than period of seasonality.
- $s_t$  : This is low frequency periodic component in time series.
- $i_t$  : This is generally high frequency component of time series. It can be looked as the residual series after removing seasonality, trend, cyclic component.

One simple method[17] to decompose the series into its component is described in algorithm (figure 2.1). This method doesn't take care about multiplicative nature of series.

```

1: Let's call original series as  $x_t$ .
2: Assign  $p = seasonal\_period$ 
3: Calculate  $\bar{x}_t = \frac{1}{p} \times \sum_{i=0}^{p-1} x_{t-i}$  {now  $\bar{x}_t$  is relatively free from seasonal component}
4:  $t_t = \bar{x}_t$ 
5: for All t do
6:   Calculate  $y_t = x_t / \bar{x}_t$  {Approximately trend free component}
7:   Calculate seasonal component( $z_t$ ) for each point in period  $p$  {e.g. if series is monthly demand series, then  $p = 12$  then calculate seasonality index for each month.}
8:    $z_j = \frac{1}{\lfloor \frac{t}{p} \rfloor} \sum_{i=0}^{\lfloor \frac{t}{p} \rfloor} (y_{i \times p + j})$  for  $\forall j \in \{1, \dots, p\}$  {All  $z_j$  should sum up to 12}
9:    $s_t = z_{t \% p + 1}$ 
10:  Calculate  $i_t = \frac{y_t}{s_t}$ 
11: end for

```

Figure 2.1: Algorithm for decomposing time series into trend, season and irregular components

In decomposing time series into various component, there lies a problem. It is widely known that we can not remove seasonality completely until and unless trend is removed and we can not remove trend completely unless and until seasonality is removed. We break this deadlock by removing or estimating trend component first and then we remove/estimate the seasonality. The choice of estimating trend is very valid as we know that sum of the seasonality indices is constant[17] and expectation of the irregular component is constant. This statistical property of seasonality and irregular component helps us when we take mean of  $x_t$  over seasonality period  $p$ , by this we are sure that this mean contain very less seasonality and also expected value of irregular component is almost a

constant(i.e. zero). This lead us to seasonality free and almost irregular component free trend estimation.

The method(figure 2.1) described above is certainly not sufficient, and most important is, it doesn't take care about multiplicative relationship. Another thing is, we have to iterate through above method so as to get rid off trend and seasonality completely from irregular component. This method doesn't give exact details about when to stop.

## 2.4.2 Seasonality Adjustment

The main challenge is to remove seasonality from the given time series. Once seasonality is removed, removing trend become simpler (either by Moving Mean/Median method). So potentially all seasonality adjustment methods could be used for the decomposing time series into its component series. Following are some seasonality adjustment methods which are widely used in econometrics domain[18].

- *Macauley's approach aka Classical decomposition*(Year 1931) : This method is one of the oldest and a bit basic method. It works better for additive model. It proceeds as follows
  1. Calculation of the seasonal component for each month of the time series by using ratios of the actual to a 12 month (if monthly data) centered moving average and then averaging to arrive at 12 seasonal indices;
  2. Estimation of the trend by using a linear or higher order polynomial;
  3. Division of the moving average data by the trend estimate to obtain estimates of the cyclical component.
  4. After calculation of seasonal, cyclic, trend component, we can find irregular component.

This method laid foundation for X11 Method. X11/X12-ARIMA is the current standard method of seasonality adjustment.

- *Census*(Year 1954) : Census method II was modified to X11 method.
- *X11 Method*(Year 1963) : Most widely used method for seasonal adjustment.

- *DAINTIES* (Year 1979) : It was being used in major databases to adjust seasonality. Precise implementation in databases is not available.
- *BV4* (Year 1983) : This method is based on curve fitting. Trend component is approximated by a polynomial of order 3 and seasonal component is approximated by eleven trigonometric functions.

### 2.4.3 Multiscale Decomposition

In this method, we use *MODWT* (section 2.2.1) for decomposing the series. Using this approach we decompose the signal/series into range of frequency scales. This gives us number of series at different scales. Each scale indicate the amount of details present in the series. This decomposed representation is an equivalent representation of the original signal. We can apply various forecasting models on these series at different scales either independently or exploiting relationship between various scales.

There are many methods based on multi-resolution analysis, these methods differ in the way of selecting wavelet coefficients for prediction from each scale[19][20], and in the way they are combining the results from each scale. In [19], prediction module at each scale selects only those points which are multiple of  $2^j$  at  $j^{th}$  scale. This method tries to fit  $AR(p)$  model at each scale, and parameters for the  $AR$  model is estimated using Neural Networks. In [21], a neural network is used to assign weights to different scales while combining these scales to give final predicted value. This method is using all wavelet coefficients at all levels.

Wavelet transform is a kind of additive process. As we discussed before (section 2.4.1), different components of time series can be in additive or multiplicative relationship. If time series is multiplicative and we are applying wavelet transform then we may not get uncorrelated scale components. This will affect our forecasting method at each scale, because any forecasting method will be looking into past and trying to forecast next point at that scale, but the values at that scale may be affected/controlled by information present at another scale. To overcome this problem we can use log transformation on series before applying wavelet transform.

For non-stationary series[2], wavelet can be used to find different components of the series. As we know that wavelet transform decomposes the signal into various frequency

ranges, we can map this to previous section 2.4.1 where trend and cycle component are low frequency components (lower than seasonal component), in monthly demand series, we know that seasonality is yearly (i.e. 12) and remaining are higher frequency components which are irregular components. So having known seasonality period and behavior of trend we can decompose the series into trend, season, and irregular components so that information present at each scale is fairly independent. This independence in different components can improve forecasting accuracy.

## 2.5 Multiple Experts

There exist numerous series with different attributes. It is not possible for any one method to perform better on all kind of series. Some series can be forecasted better by application of one forecasting methods, whereas same methods will perform badly when applied to another kind of series. In every series, some attributes of series are prominent while other attributes are very mild. A forecasting method can perform better if it can catch those prominent attributes. Different forecasting methods are designed to catch different attributes. Though finding the correct method for a given series is difficult, some-what correct forecasting method can be found using model selection (like clustering, AIC etc).

As no single method can consider all important aspects of time series, we can consider using multiple forecasting experts. Now using multiple set of experts, we can try to consider as many attributes of a series as we can. To get final forecast, a function can be applied on the forecast of various experts.

### 2.5.1 Time Series Similarity

Similarity measure plays a very important role in data mining algorithms. Similarity measure can also be used for finding the distance between two time series. Calculating distance between two time series is very important for clustering algorithm (section 2.5.2).

Similarity of time series data should be based on certain characteristics of the data rather than on the raw data itself. Ideally, these characteristics are such that the similarity of the time series is simply given by the (traditional) similarity of the characteristics. This forces us to extract characteristics of a given time series. One effort in this direction can be found in [22], where 15 attributes have been used to describe any time series. On attribute

based dimensions, we can find similarity in better manner. Clustering of time series based on this approach is reported to be working better than normal distance measures (refer [22]).

### 2.5.1.1 Distance Measure or Similarity Measure

Distance between two time series can be found in time domain as well as wavelet domain. We carried few experiments (refer section 4.2 which shows that the wavelet based distance estimator performs better than time domain in most of cases.

#### Time based Distance Calculation

Distance  $D(x, y)$  between two equal length sequence  $x$  and  $y$  can be calculated using following formula[5].

$$D(x, y) = \left( \sum_{i=0}^{n-1} (y_i - x_i)^2 \right)^{\frac{1}{2}}$$

The effect of vertical shift between two sequences has not been considered in above definition. It depends on the application, whether application want distance to include vertical shift or not. If we are thinking  $D(x, y)$  as similarity measure then above definition will fail for vertically shifted similar sequences.

To overcome the problem of vertical shift mentioned above, we have to do one minor modification in above definition, which is as follows.

$$D(x, y) = \left( \sum_{i=0}^{n-1} ((y_i - \bar{y}) - (x_i - \bar{x}))^2 \right)^{\frac{1}{2}}$$

where  $\bar{x}, \bar{y}$  are the mean values of sequences  $x$  and  $y$  respectively.

#### Wavelet based Distance Calculation

This is the simple and powerful techniques which allows for the rapid evaluation of similarity between time series in large data bases[23]. In wavelet approach, in addition to locality, it possesses very desirable ability of filtering the polynomial behavior to some predefined degree. There are also other benefits of using wavelet transform for similarity finding, which are listed below

- Some wavelet transform have compact support, that enable it to capture the local properties of data.

- The time complexity of finding wavelet transform is linear with the length of data.
- The wavelet transform is hierarchical and allows much finer tuning for variety of application.
- Wavelet transform have infinite set of possible basis function. Thus they provide access to information that can be obscured by other methods.

This method measure the correlation( $C(f, g)$ ) between Haar wavelet coefficients  $c_f^{i,j}$  and  $c_g^{k,l}$  of two respective time series  $f$  and  $g$ .

$$C(f, g) = \sum_{\{i,j,k,l\}}^{m,n} c_f^{i,j} c_g^{k,l} \delta_{i,j,k,l}$$

where  $\delta_{i,j,k,l} = 1$  iff  $i = k \& j = l$

Normalization is necessary in order to arrive at the correlation product between  $[0, 1]$  and will simply take form

$$C_{normalized}(f, g) = \frac{C(f, g)}{\sqrt{C(f, f)C(g, g)}}$$

The distance of two representations can be easily obtained as

$$Distance(f, g) = -\log |C_{normalized}(f, g)|$$

The above mentioned method has performed better in finding the similarity between various time series, we can see the results in section 4.2

In above method, we can use different wavelets for similarity search. It has been found that some wavelet function does better in dimension reduction than other wavelet function[24]. Here the process of dimension reduction means selecting a subset of wavelet coefficients(mostly from transformed space) called features.

## 2.5.2 Clustering

There has been lot of research done in time series forecasting and it has been found that there is no one method universally superior to another. There can be a method which can forecast better for some type of series whereas for other type of series some other type of method will outperform. This lead us to a problem of selecting the most appropriate method for a given series. Here the most appropriate method will be one which is giving

less forecasting error. The problem is selecting most appropriate forecasting model for a given time series. Many model selection criteria have been discussed in literature, but one methods seems to be more promising than the other methods. This method is *Mixture of expert models(MEM)*[25] for time series forecasting. In this method, problem of learning mapping from input series to output forecast is considered. There is different mapping for different regions of the input space(i.e. input series). This method uses wavelet transformation for improvement of quality of information available to the models.

MEM focuses on the problem of learning a mapping in which the form of the mapping is different for different regions of the input space. Although a single homogeneous adaptive model could be applied to this problem, we might expect that the task would be better performed if we assign different expert models to tackle each of the different regions, and then use an extra gating model, which also checks the input vector, to decide which one of the experts should be used to determine the output. This implementation of the gating model in MEM is based on the clustering algorithm, applied to Haar wavelets transformed data, which assumes the previous knowledge of the number of input space classes (clusters) and uses the Euclidian distance as the similarity measure. The idea of MEM is ‘to divide for conquer’. A complex problem is subdivided into simpler subproblems that are treated individually. The implementation of the MEM method involves following phases.

1. *Changing the base of input vector space* : The base of input space is changed by applying the Haar wavelets transform. In this way each sample is described by an overall shape plus details, thus allowing the clustering algorithm to group patterns that have closer shapes.
2. *Input space partitioning* : The transformed data plus any pertinent data available for the training of the expert models are partitioned by the clustering algorithm into a predefined number of classes. As output, we obtain the samples and the centers of mass for each cluster (input space classes).
3. *Training of expert models* : Several models are trained for all clusters, i.e. for every cluster we train all models, so there are total number of models times number of clusters.
4. *Testing and benchmarking* : Given an independent set of test patterns, firstly test pattern is classified among the input space classes identified in phase II. This is

done on the basis of minimum Euclidian distance from centroid of class. Finally, for every class, the winner expert model denoted as  $B(i)$  is selected, with the minimum forecasting error, where  $i$  is one of the input space class.

5. *Forecasting* : Given a wavelet-transformed pattern, taken from a time series, Firstly input space class is identified by selecting its corresponding nearest centroid. Then we select the model  $B(i)$  to treat this sample and produce the required forecast.

The main advantages of the MEM method are:

1. The use of the Haar wavelets transform to perform a base change of the input vector space, giving an overall shape description of each pattern to the clustering algorithm.
2. The independence of models and data, what makes it possible to train and adjust the different predictive models in a individualized form and in parallel.
3. The possibility of using different classes and variations of adaptive models, selecting those ones that best fit to particular regions of the input space.

### 2.5.3 Combining Methods

There are many forecasting techniques used in time series forecasting. Typically one technique is selected based on a selection criterion (e.g., AIC), hypothesis testing, and/or graphical inspection, and selected model is used for the forecasting. However, model selection is often unstable and may cause an unnecessarily high variability in the final estimation/prediction[6]. To overcome the above problems, combining techniques can be useful. In combining, we combine the forecast from different forecasting experts to generate final forecast. This combining function can be as simple as taking mean to as sophisticated as applying dynamic function over dynamic set of forecasting experts.

The idea of combining forecasts implicitly assumed that one model could not identify the underlying process, but different forecasting models could capture different aspects of the information available for prediction.

Combining forecasts can be used for risk minimization as it reduces the variance of the final forecast. That means, worst MAPE will follow non-increasing improvement as the number of experts in combining increases (refer figure 4.9).

Combining techniques can also be used to improve forecast accuracy. There is lot of research going to make sure that combined forecast is better than all the participating forecasting experts.

### 2.5.3.1 Choosing Participating Experts

Which forecast should participate in the combining is important if we want to improve forecast. There is split in the opinion on this front. Some researcher feels that all participating forecast should differ substantially from one another with respect to the data used and also with respect to the procedures for analyzing the data (e.g., extrapolation or econometric or judgmental). Whereas some researcher says combining forecast from similar models may be useful[6].

Combining can be helpful even with the same method but with different parameters, e.g. combining rolling and recursive[26] forecast can also improve the final MAPE. The difference between rolling and recursive forecasting method is just the window size that is being used for forecasting next point. In rolling forecast, this window size is some constant value whereas in recursive forecasting this window size is infinite, i.e. it consider all its past data points to forecast next point.

To decide which experts to combine, Scott[8] suggests possible use of Realistic simulation, Rule based forecasting. It also says that we can use prior evidence on which methods have been most accurate in a given type of situation, though we tried this method results are not very encouraging (table 4.8).

### 2.5.3.2 How to Combine

As noted by Clemen[7], past research has produced two primary conclusions, one expected and one surprising. The expected conclusion is that combined forecasts reduce error (in comparison with the average error of the participating forecast methods). The unexpected conclusion is that the simple average performs as well as more sophisticated statistical approaches. Even weighted average that depend upon estimated correlation performs poorly than above method[27][28].

There are two most popular and simple methods to combine forecasts. One is taking mean and another is taking median of forecasts from different forecasting methods. Out of these two methods many time *median* outperform *mean* method[8]. Our experiments(see

section 4.3) have also validated this argument.

### 2.5.3.3 Combining Guidelines

From experiments (section 4.3) it can be seen that most of the time combining improves forecast but that is not always the case. If any of participating series is very badly forecasted, then the overall accuracy of combined forecast deteriorates. Under such conditions combining is not useful. So under what conditions we should be using combining method? One argument says that combining is more useful for long-range forecasting because of the greater uncertainty. An alternative viewpoint is that random errors are more significant for short-range forecasts; because these errors are off-setting, a combined forecast should reduce the errors.



# Chapter 3

## Our Approach

For our forecasting experiments, we used a total of 11 series from the Time Series Library representing monthly sales (table 3.1). Except for Hsales, these represent sales of Fast Moving Consumer Goods (FMCGs).

Our approach tries to improve forecast with existing forecasting techniques. These techniques could be use of neural networks, genetic algorithms or any forecasting software. Forecasting techniques/tools used are mentioned below.

1. SAS : This is a statistical analysis software which is widely used in industries for data analysis. It also provides time series forecasting module, which helps in forecasting future values after trying many statistical models for forecasting (like ARMA, ARIMA, Log ARIMA, etc.) for a given time series.
2. Neural Networks : We used neural network library which takes time series and a training period. This library also tries various topologies with various number of input nodes and different sizes of context window.
3. Genetic Algorithms : This is also another library which takes input series and number of inputs for training and forecasts future values.
4. Holt-Winter Method (section 2.1.2.2) : This is the most widely used method. There are two variants of this method: one is additive and another is multiplicative. This method can be used using SAS software.

We used first 30 points for training if required, and considered forecasted values from 31<sup>st</sup> point for calculation of the *MAPE*. 30 is the minimum number of points that enable us to analyze two values for the same month, as we need atleast two values for fitting any equation over the time series.

Series Name	Description
Abraham12	Monthly gasoline demand Ontario gallon millions 1960 - 1975. Source: Abraham & Ledolter(1983).
Drywhite	Monthly Australian sales of dry white wine: thousands of litres. Jan 1980-Jul 1995. Source: ABS.
Fortif	Monthly Australian sales of fortified wine: thousands of litres. Jan 1980-Jul 1995. Source: ABS.
Hsales	Monthly sales of new one-family houses sold in the USA since 1973. Source: Makridakis, Wheelwright and Hyndman (1998).
Paper	Monthly sales of paper Jan 1963 – Dec 1972 printing and writing paper (10-year monthly)
Redwine	Monthly Australian sales of red wine: thousands of litres. Jan 1980 - Jul 1995. Source: ABS.
Rose	Monthly Australian sales of rose wine: thousands of liters. Jan 1980-Jul 1995. Source: ABS.
Spaper	CFE specialty writing papers monthly sales. Source: Makridakis & Wheelwright (1989).
Sparkling	Monthly Australian sales of sparkling wine: thousands of liters. Jan 1980-Jul 1995. Source:ABS.
Sweetwhite	Monthly Australian sales of sweet white wine: thousands of liters. Jan 1980 - July 1995. Source: ABS.
Wine	Monthly Australian wine sales:thousands of liters. Jan 1980 - July 1995 (total wine). Source: ABS.

Table 3.1: Time series used for analysis in this paper

We employed decomposition and expert combining for each of the series as explained below.

## 3.1 Decomposition

Using a form of pre-processing like decomposition, forecast can be improved (Table 4.1), so we decomposed a series,  $D$ , into three component series - Trend ( $T$ ), Seasonality ( $S$ ) and the Irregular component ( $IC$ ) as described in algorithm 2.1. Assuming that series can be combined using the operators,  $+$  and  $\times$ , we have eight ways of combining these series (i.e.  $T \times (S + I)$ ,  $S + T + I$ , etc). Table 4.2 shows comparison of three models. Pure multiplicative model ( $D = T \times S \times I$ ) seemed to be superior in representing sales data and so we used it in the experiments reported here. Each of these components were forecasted using various tools mentioned earlier with different parameters.

The success of decomposition depends on how well trend, seasonality and irregular components are isolated. Ideal decomposition would be making these component series orthogonal to each other. Effect of decomposition on forecast can be seen in the table 4.3

## 3.2 Combining Forecasts

We have employed the services of 40, 2 and 30 experts for forecasting  $T$ ,  $S$  and  $IC$  respectively. The names and notations for the experts for each component are listed below:

- *Trend*:
  - 20 Neural Network : NN1, NN2, ..., NN20
  - 8 AR : AR(2), AR(3), ...,AR(9)
  - 8 AR<sup>2</sup><sup>1</sup>: AR2(2), AR2(3), ...,AR2(9)
  - 2 Holt (non adaptive, yearly adaptive): HNA, HYA
  - 2 Double Exponential (non adaptive, yearly adaptive): DENA, DEYA
- *Season*:
  - Previous Value: RW (Random Walk)
  - Winter (yearly adaptive): W

---

<sup>1</sup>The parameters can take only positive values.

- *Irregular:*
  - 11 Neural Network: NIC1, NIC2, ... , NIC11
  - 8 AR2: AR2(2), AR2(3), ....., AR2(9)
  - 11 ARMA: ARMA(0,1), ARMA(0,2), ARMA(1,0), ARMA(1,1), ARMA(1,2), ARMA(2,0), ARMA(2,1), ARMA(2,2), ARMA(3,0), ARMA(3,1), ARMA(3,2)

All neural network models are monthly adaptive i.e. their weights are updated after every point. On other hand, for statistical models, weights are either updated monthly, yearly or unmodified after 30 months. We then combined all these experts resulting in 2400 experts ( $40 \times 2 \times 30$ ).

The main problem, which we are targeting, is to come up with a static subset which can forecast better than the best individual forecast. As can be seen in the table 4.10, even taking a subset of two experts can bring down MAPE, another observation from the same table is that the best subset (i.e. best pair) may not be with top two individual experts. This makes the problem more interesting, as final subset may contain few experts which may not be performing very well individually.

To come with the subset of experts, we came up with various heuristics, which are explained in following sub-sections. These heuristics work on the whole series, and heuristics are mainly for the analysis, but they can be deployed with little modifications.

### 3.2.1 Brute Force

For  $N$  experts, we can try all possible  $2^N$  possibilities of combining to get the optimal combination. Here we are essentially searching for the subset, which on combining will give us the least error (optimal combination). Trying all possibilities may not be practical, as even for a small value of  $N = 30$ , its computation will take enormous amount of time, and we have 2400 experts to consider so this option can be ruled out.

Instead of trying all possibilities, we can try choosing  $k$  experts from the set of  $N$  experts, where  $k$  can take values from 1 to  $K$  ( $K \leq N$ ). For 2400 experts, we can not test all possible combinations even upto  $K = 3$  in reasonable time, so we reduced number of experts from 2400 to 30. Selection of these 30 experts was done on the basis of irregular component expert, we chose the best combination of the trend and the seasonal component expert per irregular component expert. Now with the 30 experts we can

compute the optimal subset of experts to combine upto size  $K$  in reasonable time (for  $K = 4$  refer table 4.11). This establishes the benchmark upto subset size 4.

### 3.2.2 Clique Approach 1

Searching optimal combination would take exponential amount of time, so we will search for a near-optimal subset. We developed a heuristic for searching of a near-optimal subset in the huge search space of all possible combinations of experts. We know that the when two experts are combined then their  $MAPE$  follows following relationship:

$$MAPE_{A,B} \leq \frac{MAPE_A + MAPE_B}{2}$$

Using this information, we can map the problem in combinatorics domain. We will map each expert to a vertex in a graph and we will add an edge between two vertices (experts  $A$  and  $B$ ) iff following relation is true:

$$MAPE_{A,B} \leq MAPE_A$$

and

$$MAPE_{A,B} \leq MAPE_B$$

Now with the above constructed graph, we can try to find maximal clique. This maximal clique will have all those experts, which on combining will give better forecast. But our approach didn't work (refer table 4.12), we found that the maximal clique generated by the program was not best among all possible cliques.

### 3.2.3 Clique Approach 2

In the above approach, we were very stringent about the presence of an edge, but this can eliminate lot of possible combinations of experts which could have performed better. If we add all possible edges in the graph then finding all possible cliques of all possible sizes would be exponential and for 2400 experts, time we will need, will be in years. So we need to prune the search space so as to finish computations in reasonable time. For this, we tried to study the relationship between cliques and all its possible sub-cliques, we found that if the mean of MAPE of lower level sub-cliques is less than than the mean of the MAPE of another set of sub-cliques, then there are more chances that cliques formed

by using former sub-cliques will have lesser MAPE than the another clique formed by the set of later sub-cliques (refer figure 4.3.4). Exact algorithm is described in figure 3.1.

With this observation, we were able to prune the search space. Figure 4.3.4 essentially tells us that, if we combine, sub-cliques with lesser MAPE, then it is very probable that clique formed by these sub-cliques will have lower MAPE.

```

1: Initialize  $N = 100000$ 
2: Initialize  $currentCliqueSize = 1$ 
3: Initialize clique array from 1 to number of experts.
4: while  $|array| > 0$  do
5:   Sort all  $currentCliqueSize$  level cliques on their  $MAPE$  values.
6:   for  $i = 0$  to last clique do
7:     See how many  $currentCliqueSize + 1$  cliques by considering  $i^{th}$  clique
8:     Store higher level cliques in an array
9:     if number of cliques formed, exceed  $N$  then break;
10:  end for  $currentCliqueSize++ = 1$ 
11:  switch to new array of higher cliques
12: end while

```

Figure 3.1: Clique formation algorithm

In above approach, cliques finding takes enormous amount of time and at the end of program execution we may not get significant improvement in higher clique sizes(4.13).

### 3.2.4 Greedy Algorithm

As we saw, above approaches are exponential complexity approaches. As we have large number of experts, we may want some simple polynomial time heuristics for finding suitable combination. For solving above problem, we applied greedy approach. In greedy approach, we expect that the best solution for the sub-problem would be the part of the final solution. We developed greedy elimination, to find a suitable combination of experts. We start with all experts and then, at each step we eliminate the expert whose absence decreases the MAPE by the largest amount (or increases it by the smallest amount). This is repeated until all except the last expert have been eliminated. Initially with every removal of an expert there will be decrease in  $MAPE$  as expected (see figure 4.3.5), further removal of an expert after certain point, will break monotonic improvement and from this

point onward *MAPE* will start oscillating, net effect being improvement in *MAPE* (see figure 4.3.5) till some point.

```

1:  $S$  is the set of all experts.
2: while  $|S| > 0$  do
3:   for Each  $\{i\} \in S$  do
4:     for Each point  $j$  in the series do
5:       Compute  $F_{S-\{i\}}(j)$ 
6:     end for
7:     Compute  $MAPE_{S-\{i\}} = 100 \times \sum_j \left| \frac{F_{S-\{i\}}(j) - d(j)}{d(j)} \right|$ 
8:   end for
9:   Let  $k$  be the forecaster that minimizes  $MAPE_{S-\{i\}}$ .
10:   $S \leftarrow S - \{k\}$ 
11: end while

```

Figure 3.2: Greedy Elimination Algorithm for combining various forecasts

We tried this approach on 30 experts and it gave us better results. This being polynomial time algorithm, we were able to apply this algorithm on 2400 experts as well, here also it gave better results than any other approaches we applied. In all cases this algorithm performed better than the best expert or best pair (refer table 4.14). To check the effect of shrinking sets, we also devised *greedy accretion* which works in reverse direction, i.e. it starts with an empty set and at each iteration it adds an expert iff its presence bring down the MAPE most or increases the MAPE by least amount (algorithm 3.3). Table 4.14 shows that there is not much difference in two approaches.

As we can see, greedy approach is sub-optimal because local best choice may not be present in the global optimal solution, e.g. two worst experts may come together to give best forecast but greedy approach will remove them in the course of its run.

```
1:  $|S|$  is an empty set.
2:  $|G|$  be the set of all experts
3: while  $|S| < TotalExperts$  do
4:   for Each  $\{i\} \in \{G - S\}$  do
5:     for Each point  $j$  in the series do
6:       Compute  $F_{S+\{i\}}$ 
7:     end for
8:     Compute  $MAPE_{S+\{i\}} = 100 \times \sum_j \left| \frac{F_{S+\{i\}}(j) - d(j)}{d(j)} \right|$ 
9:   end for
10:  Let  $k$  be the forecaster that minimizes  $MAPE_{S+\{i\}}$ .
11:   $S \leftarrow S + \{k\}$ 
12: end while
```

Figure 3.3: Greedy Accretion Algorithm for combining various forecasts

# Chapter 4

## Results

In all experiments described below, we will be using series which are mentioned in table 3.1. We will be using first 30 points for training, if required, for the reason explained in section 3. All MAPE calculations start from 31<sup>st</sup> point.

### 4.1 Decomposition

In all experiments, we decomposed the series using algorithm 2.1 unless and until specified.

#### 4.1.1 Improvement using Decomposition

In following experiment, we compared the effect of decomposition on the forecast. We fed series as it is to two forecasting tools SAS and GA module. We then decomposed the series and applied SAS and GA on component series independently, and then combined component series to give us forecast for the actual series. Table 4.1 shows considerable amount of improvement for almost all series.

#### 4.1.2 Multiplicative model

We can decompose the given series in many ways as mentioned in previous section. To select the decomposition model, we carried out experiments comparing various decomposition models. We compared two mostly used models and one hybrid model in which trend is in multiplication with sum of seasonality and the irregular component, this model is inspired by work in [3]. After going through table 4.2, we can see multiplicative model is surely the winner model for demand series.

Name of Series	SAS	SAS*	GA	GA*
Abraham	2.607	2.496	3.001	3.200
Dry	8.007	7.242	9.821	8.888
Fortif	7.848	6.755	9.012	7.951
Hsales	6.786	5.945	9.552	7.672
Paper	4.469	4.207	6.546	4.708
Red	8.683	8.244	11.278	9.781
Rose	11.160	11.157	17.168	13.148
Spaper	7.277	6.437	10.639	8.670
Spark	11.670	11.583	14.410	13.226
Sweet	13.673	12.647	18.631	15.004
Wine	7.179	6.224	8.277	7.387

Table 4.1: Improvement in MAPE using decomposition method, \*indicates decomposition was used

Name of Series	$T \times S \times I$	$T \times (S + I)$	$T + S + I$
Abraham	2.8544	3.2369	4.8581
Dry	8.7494	8.7866	16.7566
Fortif	8.6129	8.9013	12.6264
Hsales	8.6752	9.2515	10.0515
Paper	4.7071	4.7928	23.9112
Red	9.4960	9.4118	20.4183
Rose	14.3090	14.3846	24.4252
Spaper	8.5504	8.7234	19.6980
Spark	13.9247	13.8166	23.4578
Sweet	16.6548	17.0456	23.7166
Wine	7.6445	8.0324	13.8647

Table 4.2: MAPE Comparison of various decomposition models

### 4.1.3 Decomposition Method

We are trying to decompose the given series into its basic component series and then forecasting them individually and then combining these decomposed component series

back into one final forecasted series. Basic component series can be anything depending on the techniques which will be used to forecast the component series. In demand series, series can be decomposed into trend, season and irregular component, which is one of the possible decompositions.

#### 4.1.3.1 Classical decomposition

Decomposition method directly affects the performance of forecast. We compared two decomposition methods keeping rest of the setup same. First method is classical decomposition (Algorithm 2.1), second method is modified version of the first one. In second method, we give more weightage to recent years than equal weightage as done in earlier method.

$$S_t = 0.4 \times \frac{D_t}{T_t} + 0.3 \times S_{t-p} + 0.2 \times S_{t-2p} + 0.1 \times S_{t-3p}$$

Series	Old decomposition	New decomposition
Abraham	3.149	2.868
Dry	8.724	8.404
Fortif	8.241	7.621
Hsales	8.181	9.193
Paper	4.738	4.937
Red	9.429	9.073
Rose	12.853	12.469
Spaper	8.416	8.234
Spark	13.226	12.163
Sweet	14.814	15.263
Wine	7.304	7.201

Table 4.3: MAPE Comparison of various decomposition methods

#### 4.1.3.2 Wavelet Decomposition

We tried to decompose series using NDWT. We used Haar wavelet with 3 levels of detail for decomposing the series. Table 4.4 shows that decomposition using NDWT with haar wavelet is not performing well at all. This shows the importance of right decomposition

method to improve the results. However we may get some better results using another wavelet function. After comparing wavelet decomposition with classical decomposition, we decided to continue with the classical decomposition as it always giving us better results.

Name of Series	SAS	SAS*
Abraham	2.6071	4.4697
Dry	8.0071	15.4835
Fortif	7.8481	14.5621
Hsales	6.7863	8.6586
Paper	4.4693	16.0513
Red	8.6826	18.1152
Rose	11.1599	17.1667
Spaper	7.2765	8.7522
Spark	11.6700	16.2504
Sweet	13.6730	16.1076
Wine	7.1786	9.2688

Table 4.4: MAPE using wavelet decomposition method, \*indicates decomposition was used

## 4.2 Similarity Measure

In this experiment, we took 5 different series, all truncated to first 128 points. Then we carried similarity measure on these series. This similarity measure is based on distance between two series. Here we are taking care of vertical shift. Following are results after applying methods from section 2.5.1.1.

Figure 4.2 shows that *abraham* is more similar to *red* than *wine* series as indicated by table 4.5. However we can see table 4.6 is clearly indicating that *red* series is more similar to *abraham* than any other series. It has been found that wavelet similarity measure works for the most of the series.

Series	Abraham	Wine	Red	Rose	Sweet
Abraham	0	<b>2.8414</b>	2.9567	3.0089	2.9981
Wine	2.8414	0	<b>0.5783</b>	0.5885	0.5852
Red	2.9567	0.5783	0	0.0575	<b>0.0483</b>
Rose	3.0089	0.5885	0.0575	0	<b>0.0170</b>
Sweet	2.9981	0.5852	0.0483	<b>0.0170</b>	0

Table 4.5: Distance between two time series using time domain method

Series	Abraham	Wine	Red	Rose	Sweet
Abraham	0	0.1141	<b>0.0472</b>	0.3416	0.1568
Wine	<b>0.1141</b>	0	0.2248	0.6043	0.3500
Red	<b>0.0472</b>	0.2248	0	0.2892	0.1521
Rose	0.3416	0.6043	0.2892	0	<b>0.2720</b>
Sweet	0.1568	0.3500	<b>0.1521</b>	0.2720	0

Table 4.6: Distance between two time series using Wavelet method

## 4.3 Combining Methods

This section discusses about the various combining experiments. All experiments in following subsections (except subsection 4.3.1) will be using 2400 experts as described in section 3 until and unless specified.

### 4.3.1 Mean and Median

In this experiment we tried to check for improvement in the forecast after deploying combining methods. As mentioned earlier in section 2.5.3, even simple mean or median techniques to combine can improve forecast accuracy. In this experiment we took three forecasting experts namely Holt-Winter(HW), Neural-Network(NN) and Genetic Algorithm(GA).

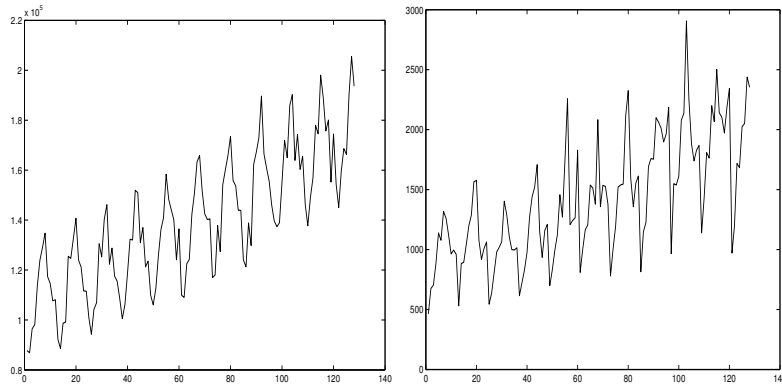


Figure 4.1: Abraham

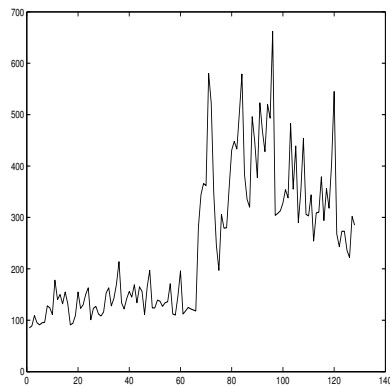


Figure 4.2: Sweet

Figure 4.3: Red

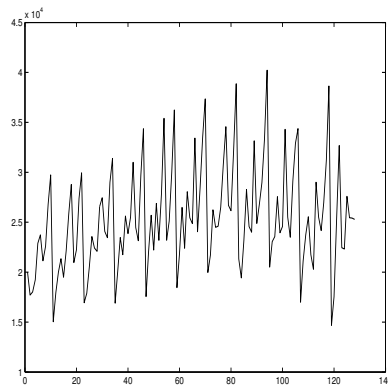


Figure 4.4: Wine

#### 4.3.1.1 Past Performance

There is one problem in the above simple mean and median combining method, if one of these functions is independent of the reference point in the time, we can say that the auto-covariance function of a forecasting method is performing really bad, then overall forecast suffers. It lacks in anticipating the past knowledge of the performance of different forecasting methods. We tried to fill this gap and took weighted average of forecasted values of different methods. The weights were decided by looking into past performance of various participating forecasting methods.

**Bayesian Method** One possible method of assigning weights to different experts is to consider how many times the given expert performed best among the set, i.e. till the point of forecast, how many times expert forecasted value with lease *MAPE* from the given set of experts. Assign that as the probability with which given expert will be forecasting with

Name of series	HW	NN	GA	Mean	Median
Abraham	3.3108	3.2159	3.0669	<b>2.9221</b>	2.9380
Dry	8.4137	8.8189	8.5554	8.1386	<b>8.0827</b>
Fortif	<b>7.9032</b>	8.4899	8.1055	7.9426	8.1740
Hsales	10.0606	8.5460	13.3592	9.5438	<b>8.2752</b>
Paper	5.0890	4.9809	<b>4.6595</b>	4.7719	4.7469
Redwine	9.5616	10.0073	10.1340	9.4751	<b>9.4672</b>
Rose	15.5460	14.1500	12.9741	13.2615	<b>12.9647</b>
Spaper	8.7556	8.9656	8.5815	8.4613	<b>8.4530</b>
Spark	12.9858	13.7458	13.4010	<b>12.8165</b>	12.9855
Sweet	18.0528	17.1736	<b>14.6793</b>	14.8038	15.2708
Wine	7.7193	7.4898	7.3366	<b>7.0878</b>	7.2503

Table 4.7: Table compaing the forecast of three experts with their combined forecast

the least *MAPE*. Treating these probabilities of experts as weights for those experts and taking weighted mean gives us final forecast.

**Wavelet Method** One method of assigning weights to experts could depend on the similarity of forecast with the actual series. We can calculate similarity between the forecasted series and original series till the point of prediction, then normalize all weights so that it would add up to one. In this method similarity was calculated using wavelet method, as we saw that similarity can be better calculated using wavelet similarity measure(see section 2.5.1.1). This method is a bit similar to the method described in section 2.5.2, with gating function applied at output side i.e. after getting forecast from different methods to get final output.

### 4.3.2 Expert Comparison

This experiment involves use of simple method to combine forecasts from 2400 different experts. We compared this simple combining strategy with heavily used Holt-Winter monthly adaptive method. Table 4.9 shows that even a naive combining method using the median of all experts' forecasts does better in over 50% of the series compared with monthly adaptive Holt-Winter forecasting. This method is completely non-cheating and

```

1: Call forecast generated by first method as  $f_t$  and forecast generated by another method as  $g_t$ .
2: for Each forecasted value  $i$  do
3:   Compute similarity measure with original series  $x_t$  for  $f_t$  and  $g_t$  for  $(t < i)$  using wavelet similarity measure.
4:   Call similarity measure  $S_f$  and  $S_g$ .
5:   Assign weights  $w_f$  and  $w_g$  to forecasted values from  $f_t$  and  $g_t$ .
6:    $w_g = \frac{S_f}{S_g + S_f}$ 
7:    $w_f = 1 - w_g$ 
8:   output forecasted value as  $\hat{x}_t = w_f \times f_i + w_g \times g_i$ .
9: end for

```

Figure 4.5: Algorithm for combining two forecast using wavelet method

series	Bayes Method	WC
Abraham	2.8986	2.9168
Dry white	8.1229	8.1512
Fortif	7.9021	7.9301
Hsales	9.7551	8.7420
Paper	4.7520	4.7770
Redwine	9.4866	9.5824
Rose	13.2949	13.7100
Spaper	8.6215	8.4868
Spark	12.7805	12.8156
Sweet	14.3988	14.4874
Wine	7.1305	7.1021

Table 4.8: Table compares two methods with table 4.7

hence, very practical method.

Next experiment deals with the analysis of participating forecasting experts in pairs. Table 4.10 shows the top two experts for each series and their MAPEs. It also shows the best combination of two experts. While there is negligible improvement in combining two experts in the Hsales series, the improvement for Rose and Sweet is 5-6% and averages 2% in the other cases. One immediate observation is the diversity of experts that yield

Series	HW adaptive	Median	Mean
Abraham	3.286	3.073	3.035
Dry	8.324	8.937	9.054
Fortif	7.641	8.177	8.213
Hsales	10.453	8.220	8.322
Paper	5.065	4.882	5.013
Red	9.231	9.469	9.428
Rose	14.473	12.712	12.568
Spaper	8.568	8.696	8.843
Spark	12.872	13.456	13.452
Sweet	16.880	15.314	15.220
Wine	7.568	7.381	7.458

Table 4.9: MAPEs using holt-winter and mean/median of all combinations of experts

best forecasts across series. Also, it is usually the case that neither of the experts in the best pair combination is the best in its own right.

### 4.3.3 Brute Force

For applying brute force approach, we selected 30 experts and decided to compute all the possible subsets upto size  $K$ . For deciding the value of  $K$  we carried out experiment for finding all possible combinations of 20 experts, i.e. all  $2^{20}$  possibilities. In most of the cases, we found that the subset of size 3 is giving the best result. So we decided to compute all possible subset upto size of 4.

### 4.3.4 Clique Approaches

We compared two clique approaches (section 3.2.2, 3.2.3) on 30 experts for different series. We also compared what is the maximum size clique generated by both approaches along with which clique size gives us best result for that approach. Table 4.12 shows comparison of two approaches along with best clique size and maximum clique size generated.

The motivation behind approach 2 was the observation that, best clique of size  $k$  will have less average of its  $k$  sub-cliques of size  $k - 1$ . This can be seen in the following plot

Series	Top 2 experts	Best pair
Abraham	2.870 (NT15, W, ARMA(3,2)) 2.873 (AR(7), W, ARMA(3,2))	2.846 (AR(7), RW, NIC1) + (AR(7), W, ARMA(3,2))
Dry	8.724 (NT8, W, NIC3) 8.746 (AR(9), W, NIC3)	8.606 (NT8, W, NIC8) + (AR(9), W, NIC4)
Fortif	7.889 (AR(5), W, AR2(6)) 7.904 (AR(5), RW, AR2(6))	7.834 (AR(9), W, AR2(7)) + (HNA, RW, NIC1)
Hsales	7.588 (AR(5), W, AR2(6)) 7.593 (AR(2), W, AR2(6))	7.587 (AR(5), W, AR2(7)) + (AR(5), W, AR2(9))
Paper	4.634 (NT6, W, AR2(7)) 4.639 (NT19, W, AR2(7))	4.525 (NT18, W, AR2(3)) + (NT19, W, AR2(6))
Red	9.311 (AR2(8), W, ARMA(1,1)) 9.331 (AR(8), W, ARMA(2,1))	9.194 (NT13, W, AR2(5)) + (AR(2), RW, AR2(2))
Rose	12.772 (AR(9), W, NIC6) 12.908 (AR(9), W, NIC5)	12.030 (NT2, RW, AR2(3)) + (NT15, W, ARMA(0,2))
Spaper	8.000 (NT19, W, ARMA(1,1)) 8.008 (DENA, W, ARMA(1,1))	7.955 (NT19, W, ARMA(2,1)) + (DENA, W, ARMA(1,2))
Spark	12.957 (AR2(9), W, AR2(7)) 13.000 (AR2(9), RW, AR2(7))	12.776 (NT2, W, ARMA(2,0)) + (AR2(9), W, AR2(8))
Sweet	14.701 (AR(6), W, AR2(7)) 14.814 (NT15, W, NIC10)	13.915 (NT4, W, NIC11) + (AR(6), W, AR2(9))
Wine	7.261 (NT14, W, ARMA(0,1)) 7.266 (NT14, W, ARMA(1,0))	7.040 (NT9, W, NIC6) + (NT15, W, ARMA(3,0))

Table 4.10: Best model and best pair maps using 2400 experts

Series	Best Model	Best Pair	Best 3	Best 4
Abraham	2.8702	2.8556	2.8212	2.832
Dry white	8.7244	8.6078	8.6239	8.6395
Fortif	7.8887	7.8403	7.8127	7.7934
Hsales	7.5883	7.587	7.5853	7.6141
Paper	4.6342	4.5453	4.5357	4.5459
Redwine	9.311	9.2034	9.1013	9.1046
Rose	12.7722	12.1265	11.9780	11.9437
Spaper	8.0000	7.9577	7.9514	7.9728
Spark	12.9569	12.8023	12.7922	12.8393
Sweet	14.7013	14.1737	13.9672	13.755
Wine	7.2607	7.0794	6.9876	6.9728

Table 4.11: Comparison of MAPE of various subset sizes

Series	Approach 1	Approach 2
Abraham	2.82122(3/12)	2.82122(3/7)
Dry white	8.60781(2/10)	8.60781(2/7)
Fortif	7.81272(3/11)	7.81272(3/6)
Hsales	7.58701(2/6)	7.58535(3/9)
Paper	4.53571(3/7)	4.53571(3/7)
Redwine	9.1013 (3/8)	9.1013(3/7)
Rose	11.9129(6/8)	11.9129(6/6)
Spaper	7.95137(3/5)	7.95137(3/5)
Spark	12.8023(2/7)	12.7922(3/8)
Sweet	13.9672(3/11)	13.714 (5/6)
Wine	6.97076(5/12)	6.9728 (4/8)

Table 4.12: Comparison of two clique approaches, number in bracket (best clique/max clique size)

4.3.4. In this plot, we sorted cliques of size 4 on their MAPE values, and plotted the mean of MAPE of 4 possible subset/cliques of size 3, 6 possible pairs and 4 individual

experts. We can say that if given two cliques  $A, B$  of size 4, we can compute the mean of the MAPE of sub-cliques of size 3 for each of the clique  $A, B$  and whichever is giving less mean MAPE may have lesser MAPE for the clique size of 4.

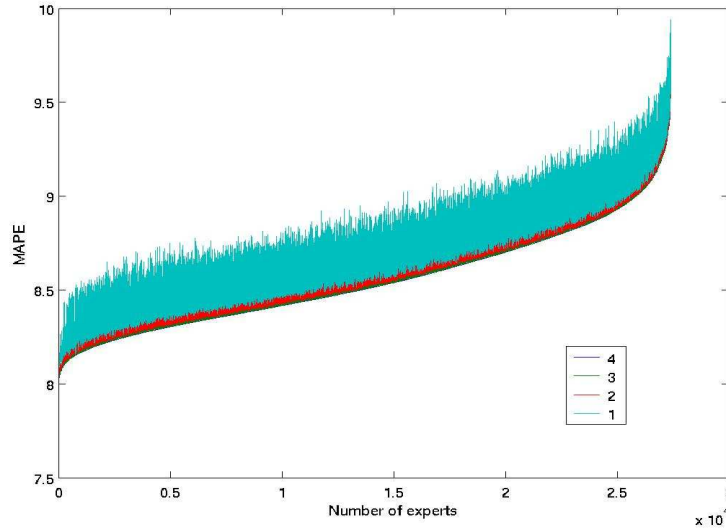


Figure 4.6: Mean of MAPE of various subset of a clique

We applied second approach on set of 2400 experts. We considered only 100000 cliques of size  $k$  for finding cliques of size  $k + 1$ . These cliques were sorted on the basis of their's MAPE values. Following table indicates the best MAPE values for various clique sizes.

Above approaches take a long time even for clique size of 7, so we need to search for the another heuristic for faster execution and may be better performance in terms of accuracy.

### 4.3.5 Greedy

From a given set of 2400 experts, to select subset of experts which can give us better forecast than individual, we applied greedy elimination (section 3.2) and accretion (3.3) algorithms. We also applied this method across trend, season and irregular component and then multiplied trend, season, and irregular component. We call this combination as horizontal combination, i.e. we apply combining techniques horizontally and then apply composition on the three components to get final forecast.

We can plot the MAPE improvement followed by the greedy elimination algorithm in terms of number of experts and corresponding *MAPE*. Two lines indicate different

Series	1	2	3	4	5	6
Abraham(8)	2.87021	2.84551	2.8227	2.813	2.81677	2.81966
Dry(8)	8.72444	8.60551	8.5701	8.54225	8.53954	8.55426
Fortif(9)	7.88873	7.83382	7.80779	7.81271	7.81551	7.82351
Hsales(7)	7.58835	7.58701	7.58503	7.58412	7.58529	7.5856
Paper(10)	4.63417	4.52492	4.51745	4.51968	4.51736	4.52278
Red(7)	9.31102	9.19565	9.14795	9.14374	9.15343	9.15183
Rose(6)	12.7722	12.0298	11.8226	11.8752	11.9065	11.9611
Spaper(5)	8.00003	7.95517	7.94133	7.92723	7.92782	
Spark(6)	12.9569	12.7758	12.7822	12.7746	12.7792	12.7985
Sweet(6)	14.7013	14.1116	13.7185	13.716	13.7906	13.9703
Wine(6)	7.26066	7.09664	7.07771	7.08006	7.07131	7.08478

Table 4.13: MAPE of various clique sizes. number in bracket indicate maximum clique size we got for that series

combining methods i.e. mean and median method. In plot 4.3.5, we can see that as number of experts reduces from 2400 to 200, *MAPE* continues to fall, but after that point, *MAPE* does not follow monotonic improvement in forecast as can be seen in figure 4.3.5. In zoomed portion, we can see that there are too much oscillations in the median approach, this may be because of removal of balancing expert from the set.

### 4.3.6 Minimizing Maximum MAPE

Combining method helps greatly in reducing the worst MAPE of forecast. This can be seen as the minimization of risk as mentioned in section 2.5.3. Figure 4.9 shows the plot of the best, worst and mean MAPE possible using  $k$  experts, where  $k$  can take value of 1 to  $N$ ,  $N$  being the total number of experts under consideration. Horizontal line indicates the mean of MAPE of all possible combinations in that category. We tried all possible combinations of experts from a set of  $N = 20$  experts.

As we can see in figure, the worst MAPE monotonically decreases and so is the mean. This suggests us to use combining techniques instead of using a method or a static subset of

Series	Elimination	Elimination*	Accretion	Elimination Horz
Abraham	2.77945(16)	2.73782(9)	2.7811(27)	2.8439
Dry white	8.5017(8)	8.55744(5)	8.5226(13)	8.5635
Fortif	7.73728(17)	7.7691(11)	7.741(17)	7.7523
Hsales	7.58185(14)	7.59972(23)	7.583(15)	7.5922
Paper	4.52254(8)	4.48844(11)	4.5264(9)	4.5358
Redwine	9.07615(12)	8.94274(13)	9.0768(15)	9.1314
Rose	11.7262(27)	11.6629(27)	11.7413(53)	11.9516
Spaper	7.93661(9)	7.90036(11)	7.9318(4)	7.955
Spark	12.703(25)	12.4796(15)	12.7109(27)	12.7547
Sweet	13.2899(20)	14.0783(17)	13.2705(17)	13.3598
Wine	6.92932(23)	6.89241(11)	6.9306(11)	6.9774

Table 4.14: Comparison of various greedy approaches, \*indicate that median was used for combining various forecasts. Figure in bracket indicates the number of experts that were present in the subset selected.

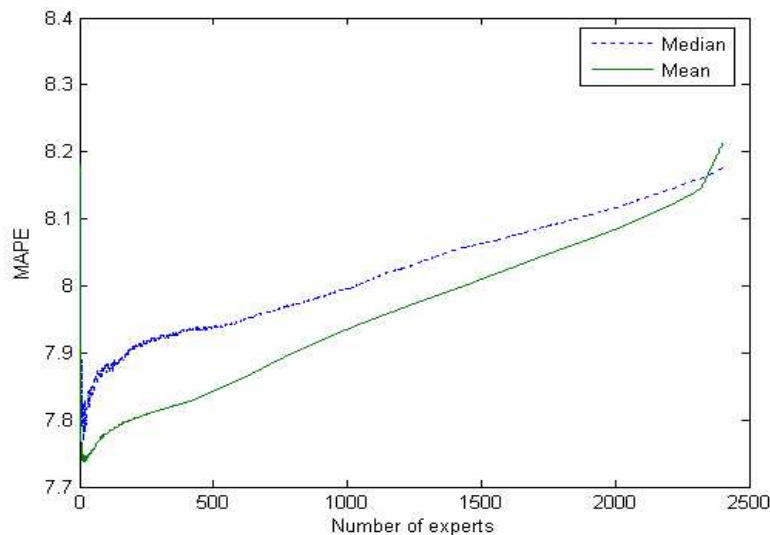


Figure 4.7: Progress of greedy elimination algorithm using mean and median(series: fortif)

experts throughout, for minimizing risk. However as number of experts goes on increasing, best MAPE also starts increasing. So we have to stop at a point where best MAPE is not worse and risk is within expected bound.

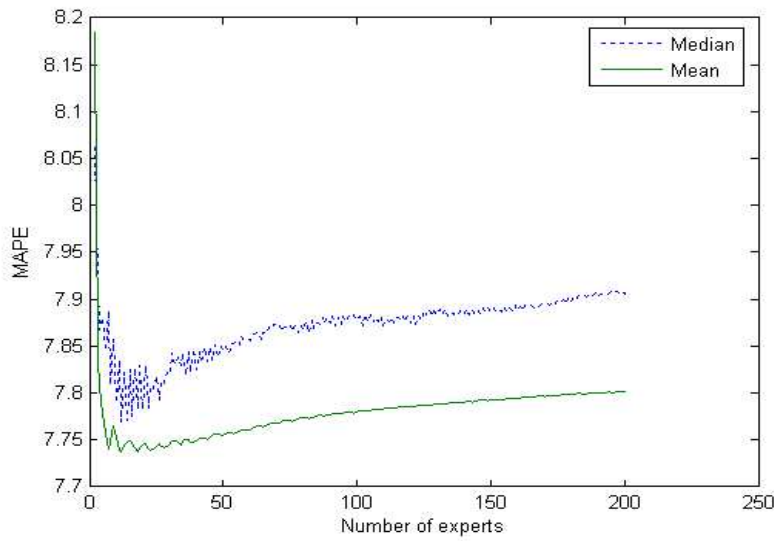


Figure 4.8: Progress of greedy elimination algorithm using mean and median (zoomed in) when number of experts go down from 200 to 1 (series: fortif)

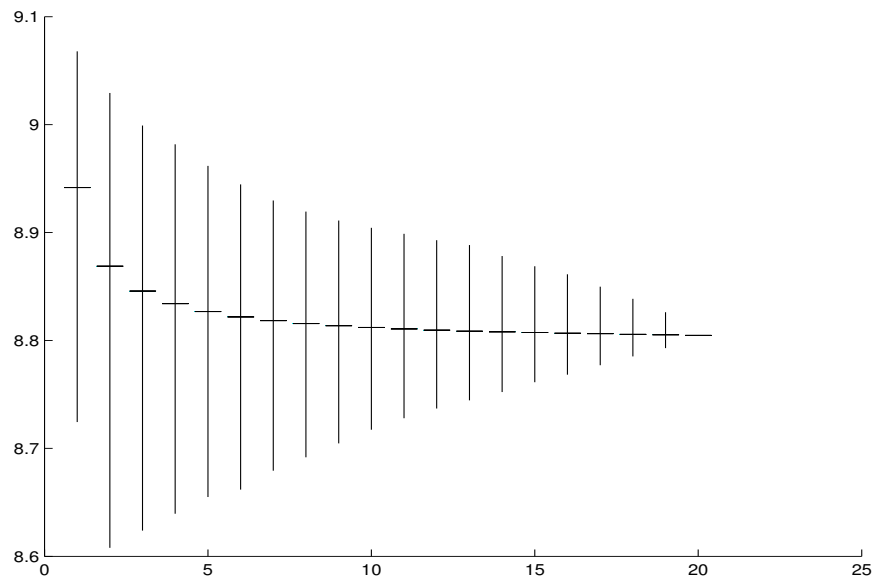


Figure 4.9: Best, mean and the worst MAPE for combination of  $k$  experts

## 4.4 Error Measure

Though *MAPE* is the most widely used error measure for measuring forecast accuracy, over the course of project, we found that *MAPE* is not sufficient and lacks in one of the most important property, it doesn't give us the least *MAPE* (i.e. the best forecast) even when we combine trend, seasonal, irregular components of series with the least *MAPE* in

their category. This is a major problem, as it will leave us direction-less which forecasted value for trend, season and irregular component we should be selecting to get best possible forecast.

To illustrate above scenario, we carried out an experiment(table 4.15) where we forecasted seasonal component using two methods, one was using neural network(NN) and another was using random walk(carbon copy i.e. current value is the forecast for the next value). Seasonality prediction using random walk(RW) was giving us less *MAPE*, but when we combined this seasonality with the trend and irregular component, we found that effective value of *MAPE* was more than the one in which seasonality was predicted using neural networks. In this experiment we left trend and irregular component untouched.

Name of Series	NN	final(NN)	RW	final(RW)
Abraham	0.4860	3.1279	0.3641	3.1413
Dry	1.3361	8.9403	1.0100	8.9643
Fortif	1.4647	8.2829	0.9736	8.3347
Hsales	1.6982	8.9358	1.4440	9.2167
Paper	1.0302	4.7786	0.7554	4.7781
Red	2.0710	10.1590	1.0470	9.6191
Rose	2.8134	13.3718	1.6031	13.3825
Spaper	1.2493	8.4829	0.9940	8.4239
Spark	2.1028	13.7452	1.4441	13.8467
Sweet	3.3417	15.4858	2.0418	16.0429
Wine	1.6522	7.6832	0.8293	7.4378

Table 4.15: Inability of MAPE to reflect improvement in seasonality prediction. 1<sup>st</sup> and 3<sup>rd</sup> column shows seasonality MAPE whereas 2<sup>nd</sup> and 4<sup>th</sup> column shows overall MAPE after combining all three components

Many modern forecasting methods use *MAPE* as an error measure, to compare our work with the existing work we had to choose *MAPE* as an error measure criteria.

# Chapter 5

## Conclusion and Future Work

We found that decomposition improves forecasts greatly though there is considerable room for improvement. We used a myriad of forecasting techniques, both neural and statistical for each component series. A simple strategy like using the median of all forecasts is about 4% worse on an average than the best model while greedy elimination is about 4% better than the best model. No single technique is good across all series or even over every part of a single series. Hence, combining helps to minimize the risk caused by an expert making outrageous forecasts.

We have used classical decomposition, but we feel that this method could be refined a lot. Currently we have many experts, these experts could be ranked and eliminated to make search space less. We are trying to find out static set of experts, next step to this would be to find out dynamic set of experts over different points in time and different weighing functions.



# Bibliography

- [1] G. E. Box, G.M.Jenkins, and G.C.Reinsel. *Time Series Analysis*. Pearson Education, 2003.
- [2] P. J. Brockwell and R. A. Davis. *Introduction to Time Series and Forecasting*. Springer, second edition edition, 2001.
- [3] R.D. Snyder, A. Koehler, and K. Ord. Forecasting for inventory control with exponential smoothing. *International Journal of Forecasting*, 18(1):5–18, January-March 2002.
- [4] Min Qi G. and Peter Zhang. Neural network forecasting for seasonal and trend time series. *European Journal of Operational Research*, 160(2):501–514, 2005.
- [5] Kin pong Chan and Ada Wai-Chee Fu. Efficient time series matching by wavelets. In *ICDE*, pages 126–133, 1999.
- [6] Hui ZouE and Yuhong Yang. Combining time series models for forecasting. *International Journal of Forecasting*, 20(1):69–84, January-March 2004.
- [7] Robert T. Clemen. Combining forecast: A review and annotated bibliography. *International Journal of Forecasting*, 5(4):559–583, 1989.
- [8] J. Scott Armstrong. Combining forecasts: The end of the beginning or the beginning of the end? *International Journal of Forecasting*, 5(4):585–588, October 1990.
- [9] Pedro A. Morettin. From fourier to wavelet analysis of time series. Technical report, Department of statistics, University of Sao Paulo, Brazil, 1999.
- [10] R. M. Rao and A. S. Bopadirdikar. *Wavelet Transform Introduction to Theory and Applications*. Pearson Education, 2002.

- 
- [11] Donald Percival and Andrew Walden. *Wavelet Methods for Time Series Analysis*. 0521640687. Cambridge university press, 2000.
- [12] Guy P. Nason and Rainer von Sachs. Wavelets in time series analysis. *Philosophical Transactions of the Royal Society of London A*, 357(1760):2511–2526, 1999.
- [13] J.Scott Armstrong and Fred Collopy. Error measures for generalizing about forecasting methods: Empirical comparisons. *International Journal of Forecasting*, 8(1):69–80, June 1992.
- [14] Paul Goodwin and Richard Lawton. On the asymmetry of the symmetric mape. *International Journal of Forecasting*, 15(4):405–408, October 1999.
- [15] F. Collopy and J.S.Armstrong. Another error measure for selection of the best forecasting method: The unbiased absolute percentage error. 2000.
- [16] Jr. Everette S. Gardner and Joaquin Diaz-Saiz. Seasonal adjustment of inventory demand series: a case study. *International Journal of Forecasting*, 18(1):117–123, January-March 2002.
- [17] D.L. Rubinfeld and R.S. Pindyck. *Econometric Models and Economic Forecasts*. 0079132928. McGraw-Hill/Irwin, 4 edition, July 1 1998.
- [18] Bjorn Fisher. Decomposition of time series comparing different methods in theory and practice. Technical report, Eurostat, 1995.
- [19] Olivier Renaud, Jean-Luc Starck, and Fionn Murtagh. Prediction based on a multiscale decomposition. *International Journal of Wavelets, Multiresolution and Information Processing*, 1(2):217–232, 2003.
- [20] Zheng Gonghui, Jean-Luc Starck, Jonathan Campbell, and Fionn Murtagh. Multiscale transforms for filtering financial data streams. *Journal of Computational Intelligence in Finance*, 7:18–35, 2003.
- [21] Bai-Ling Zhang, R. Coggins, M.A. Jabri, D. Dersch, and B. Flower. Multiresolution forecasting for futures trading using wavelet decompositions. *Neural Networks, IEEE Transactions*, 12(4):765–775, july 2001.

- 
- [22] Xiaozhe Wang, Kate A. Smith, Rob J. Hyndman, and Daminda Alahakoon. A scalable method for time series clustering. Unrefereed research papers, 1 April 2004.
- [23] Zbigniew R. Struzik and Arno Siebes. The haar wavelet transform in the time series similarity paradigm. In *PKDD '99: Proceedings of the Third European Conference on Principles of Data Mining and Knowledge Discovery*, pages 12–22. Springer-Verlag, 1999.
- [24] Ivan Popivanov and Renee J. Miller. Similarity search over time-series data using wavelets. In *ICDE '02: Proceedings of the 18th International Conference on Data Engineering (ICDE'02)*, pages 212–222. IEEE Computer Society, 2002.
- [25] Ruy-L., Ricardo-J., and Raul-P. Time-series forecasting through wavelets transformation and a mixture of expert models. *Neurocomputing, Elsevier Science*, 28(1-3):145–156, 1999.
- [26] Michael W. McCracken and Todd E. Clark. Improving forecast accuracy by combining recursive and rolling forecasts. Technical Report 0420, Department of Economics, University of Missouri, December 2004. available at <http://ideas.repec.org/p/umc/wpaper/0420.html>.
- [27] Michale Hibon and Theodoros Evgeniou. To combine or not to combine: selecting among forecasts and their combinations. *International Journal of Forecasting*, 21(1):15–24, January-March 2005.
- [28] Lilian M. de Menezes, Derek W. Bunn, and James W. Taylor. Review of guidelines for the use of combined forecasts. *European Journal of Operational Research*, 120(1):190–204, January 2000.



# Publications

1. Pankaj Gulhane, Bernard Menezes, Kunduru Timma Reddy, Kalam Shah *Forecasting using Decomposition and Combinations of Experts*, International Conference on Artificial Intelligence (ICAI 2005), June 2005.
2. Pankaj Gulhane, Bernard Menezes, Kunduru Timma Reddy, Kalam Shah *New Perspectives on Forecasting in Centralized Supply Chain Management* , International Symposium on Forecasting (ISF 2005), June 2005. *only abstract*



# Acknowledgements

First and foremost I would like to express my gratitude and appreciation to my both advisors **Prof. V. M. Gadre** and **Prof. B. L. Menezes**. I am indebted to them for their guidance, encouragement, support and trust in me. I strongly feel that I was privileged to interact with the best advisors in the institute. From both of my advisors I learned a lot, most valuable being *to stay focussed* and *to enjoy challenges*.

I would also like to thank *Forecasting Group* of IIT Bombay. I would like to thank *Kalam Shah*, *Kunduru Timma Reddy* and *Deven Puri* for helping me with various forecasting methods. I am also thankful to *Abhishek Seth*, *Suneel Sarawat*, and *Jayakrishnan* for helping in exploring various aspects of the forecasting. I would like to extend my thanks to *Ashish Singh* and *Nagesh P.C.* for patiently listening to my new ideas about forecasting.

Last, but certainly not the least, I would like to thank the people who mean a lot to me, my family. I deeply thank my mother, father, brother and sister for unflinching encouragement and support and their true and endless love.

I would also like to thank the entire KReSIT and IITB family for making my stay here wonderful. Thank you **IIT Bombay**.

**Pankaj E. Gulhane**

KReSIT,

IIT Bombay

June 13, 2005

