Visualizing Relationships

Statistics 4868/6610 Data Visualization

Prof. Eric A. Suess

2/24/2016

Introduction

Today we will discuss the end of Chapter 6, Comparison.

- python for data manipulation.
- We will discuss the Histogram Matrix and small multiples.
- Forecasting Time Series
- Hiereachical and Grouped Time Series
- parallel processing
- Start Chapter 8

python

For those of you that are interested in learning python for data manipulation and analysis, there are lots of good resources.

I would first suggest the video series on python from TheNewBoston.

Or you could start with the interactive course from codecademy.

I would second suggest the book by Norm Matloff, UC Davis, Fast Lane to Python.

And I would third suggest the book Python for Data Analysis: Data Wrangling with Pandas, NumPy, and IPython by Wes McKinney.

A nice install of python is the Enthought Canopy Express.

python

The author presents a nice python program to wrangle the transform-birth-rate data file. It takes rows, skipping the first column, stacking the data into two columns, the year and the rate.

python

```
import csv

reader = csv.reader(open('birth-rate.csv',
  'r'), delimiter=",")
writer = csv.writer(open('birth-rate-
  yearly5.csv', 'w'), delimiter=",")

rows_so_far = 0

data = ["year", "rate"]
writer.writerow(data)

for row in reader:
    if rows_so_far == 0:
        header = row
        rows_so_far += 1
    else:
```

python

```
        for i in range(len(row)):
            if i > 0 and row[i]:
                data = [header[i], row[i]]
                writer.writerow(data)

        rows_so_far += 1
```

python

To run the code at a command line.

```
$ python transform-birth-rate.py >
    birth-rate-yearly.csv
```

R

Or try the **dplyr** package in R written by Hadley Wickham.

- dplyr Cheetsheet
- dplyr tutorial

Or try the newer **tidyr** package in R written by Hadley Wickham.

Histogram Matrix

My second favorite plot is the Histogram Matrix using the **lattice** library in R.

Histogram Matrix

There are three examples in the book.

- birth-rate data
- tv data
- rotten tomatos and IMDB

small multiples

According to the author,...

The technique of putting a bunch of small graphs together in a single graphic is commonly referred to as **small multiples**.

It encourages readers to make comparisons across groups and categories, and within them.

Plus, you can fit a lot in one space if your graphic is organized.

Additional Topics

While we are discussing **Correlation** and **Regresson** and **Comparison**, we can discuss **Forecasting** and **Hierarchical Time Series**.

And add to it **parallel processing**, which is available in R as part of base R through the **parallel** package that is now part of base R. Many libraries have function that are giving parallel options in functions now.

Forecasting

There is an excellent OTexts Forecasting book available online, Forecasting: principles and practice.

```
library(fpp)
```

Last time we discuss Linear Regression and here is some discussion about the use of Dummy Variables in a time series setting.

We have already discussed Time Series Decomposition, you might check out X-12-ARIMA and STL.

Forecasting

Other topics to consider that can be useful when trying to visualize time series data.

1. Exponential Smoothing
2. Hierarchical time series forecasting

Exponential Smoothing

Chapter 7 in fpp describes

1. Simple Exponential Smoothing
2. Holt's method
3. Holt-Winter's method
4. beyond

You can do all of this in MS Excel and Minitab also.

Forecasting Hierarchical or Grouped Time Series

Chapter 9 in fpp describes Hierarchical or grouped time series. These ideas are very similar to the ideas presented in our book for Comparison.

But since this is for time series data, forecasting can be used to project into the future.

Forecasting Hierarchical or Grouped Time Series

Rob J. Hyndman, one of the authors of fpp, recently gave a talk about Visualization and forecasting of big time series data

He maintains the Time Series Data Library now part of DataMarket.

He has worked on the R libraries

```
library(forecast)
```

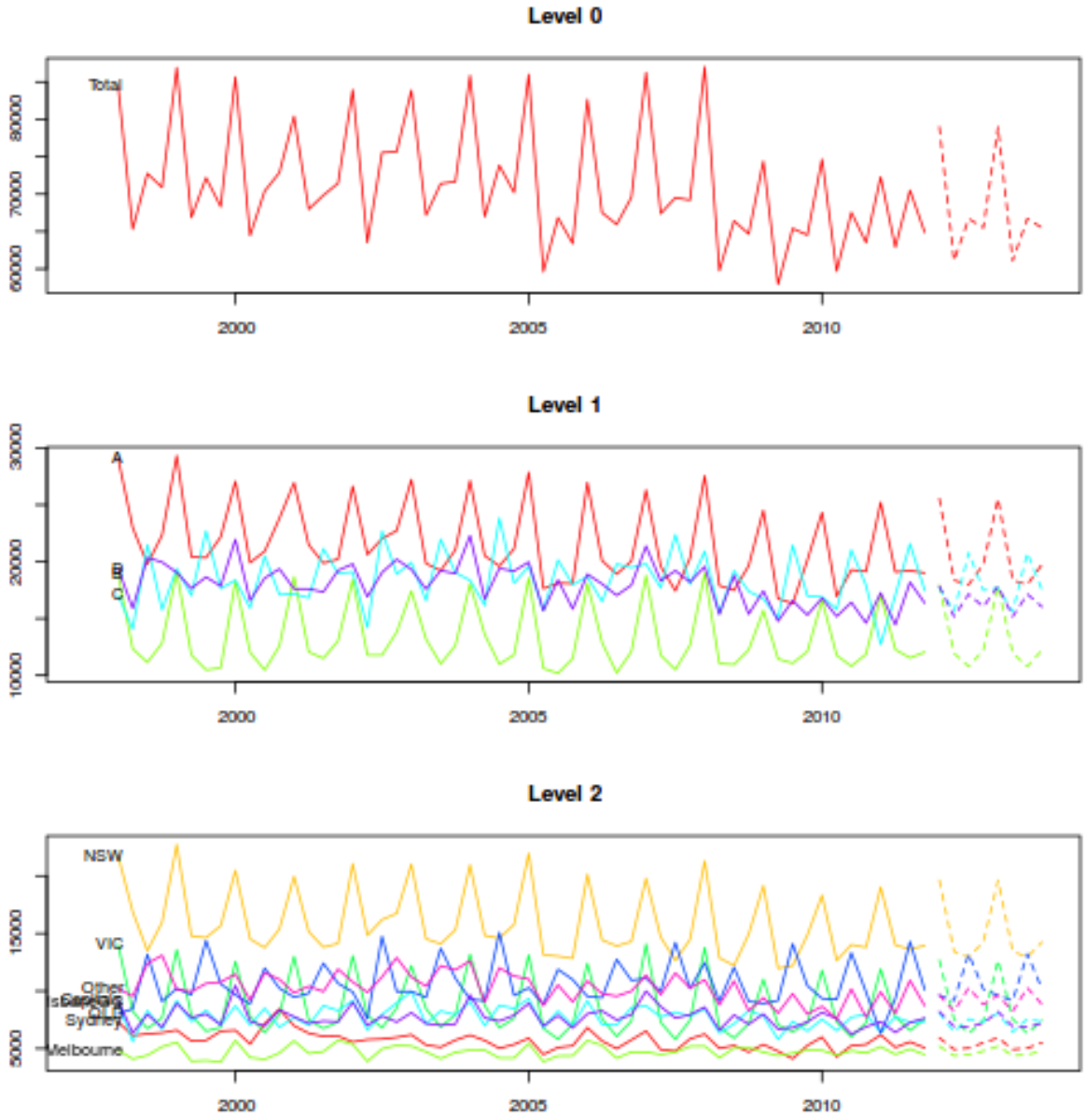
```
library(hts)
```

Forecasting Hierarchical or Grouped Time Series

Read Example 9.7 in fpp and try the R code for the Australian tourism hierachy.

Forecasting Hierarchical or Grouped Time Series

```
library(fpp)
library(hts)
y <- hts(vn, nodes=list(4,c(2,2,2,2)))
allf <- forecast(y, h=8, parallel=TRUE,
  num.cores=4)
plot(allf)
```



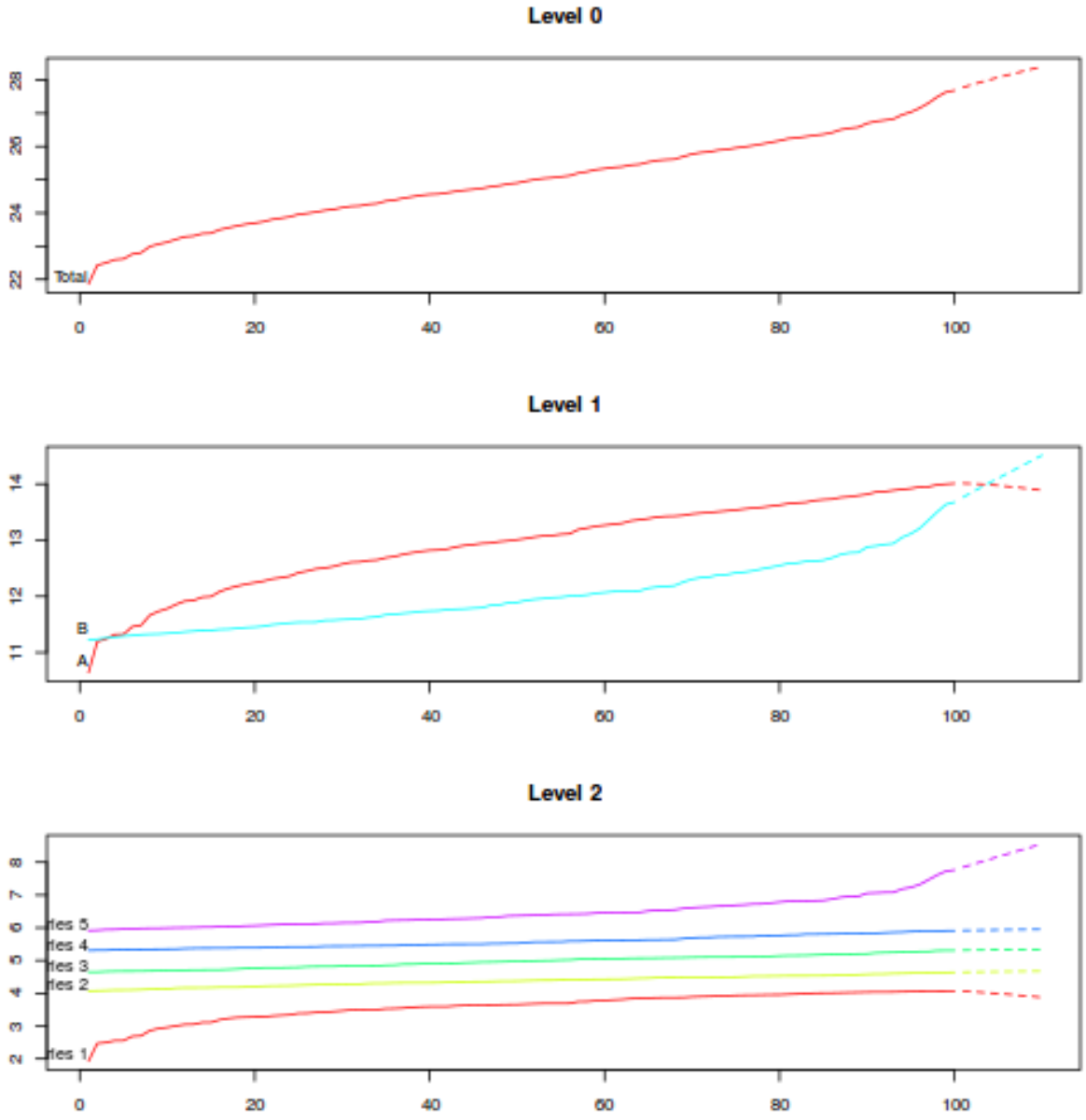
Forecasting Hierarchical or Grouped Time Series

```
# Example 1
# The hierarchical structure looks like
# 2 child nodes associated with level 1,
# which are followed by 3 and 2 sub-child
# nodes respectively at level 2.

nodes <- list(2, c(3, 2))
abc <- ts(5 + matrix(sort(rnorm(500)),
  ncol = 5, nrow = 100))
```

```
x <- hts(abc, nodes)
```

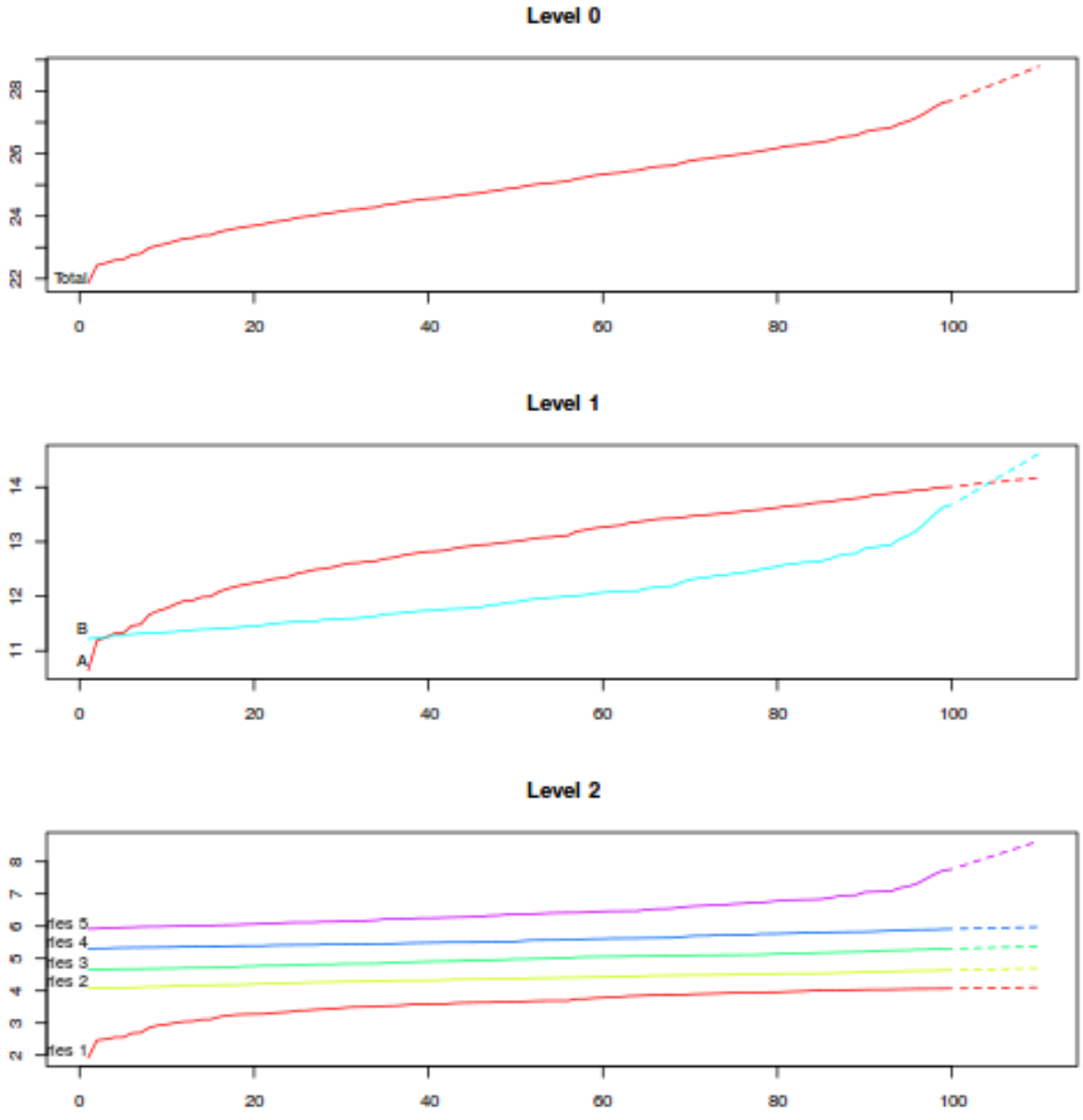
Forecasting Hierarchical or Grouped Time Series

```
# etc

fc <- forecast(x, h=10, fmethod="ets",
  parallel=TRUE, num.cores=2)

plot(fc)
```



Forecasting Hierarchical or Grouped Time Series

```
# arima

fc <- forecast(x, h=10, fmethod="arima",
     parallel=TRUE, num.cores=4)

plot(fc)
```

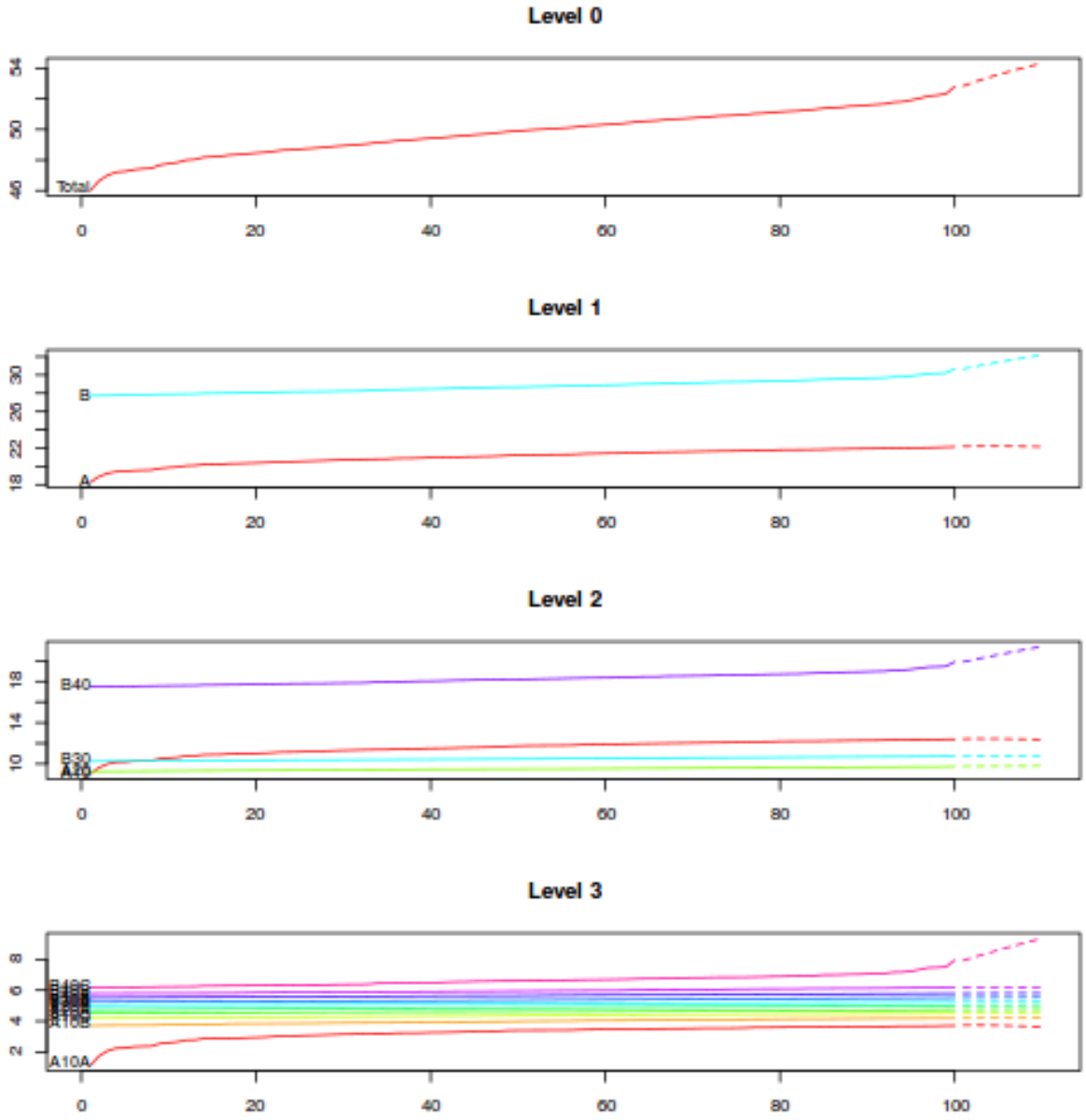


Forecasting Hierarchical or Grouped Time Series

```
# Example 2
# Suppose we've got the bottom names that
# can be useful for constructing the node
# structure and the labels at higher levels.
# We need to specify how to split them
# in the argument "characters".

abc <- ts(5 + matrix(sort(rnorm(1000)),
      ncol = 10, nrow = 100))
colnames(abc) <- c("A10A", "A10B", "A10C",
      "A20A", "A20B", "B30A","B30B", "B40A",
      "B40B", "B40C")

y <- hts(abc, characters = c(1, 2, 1))
```

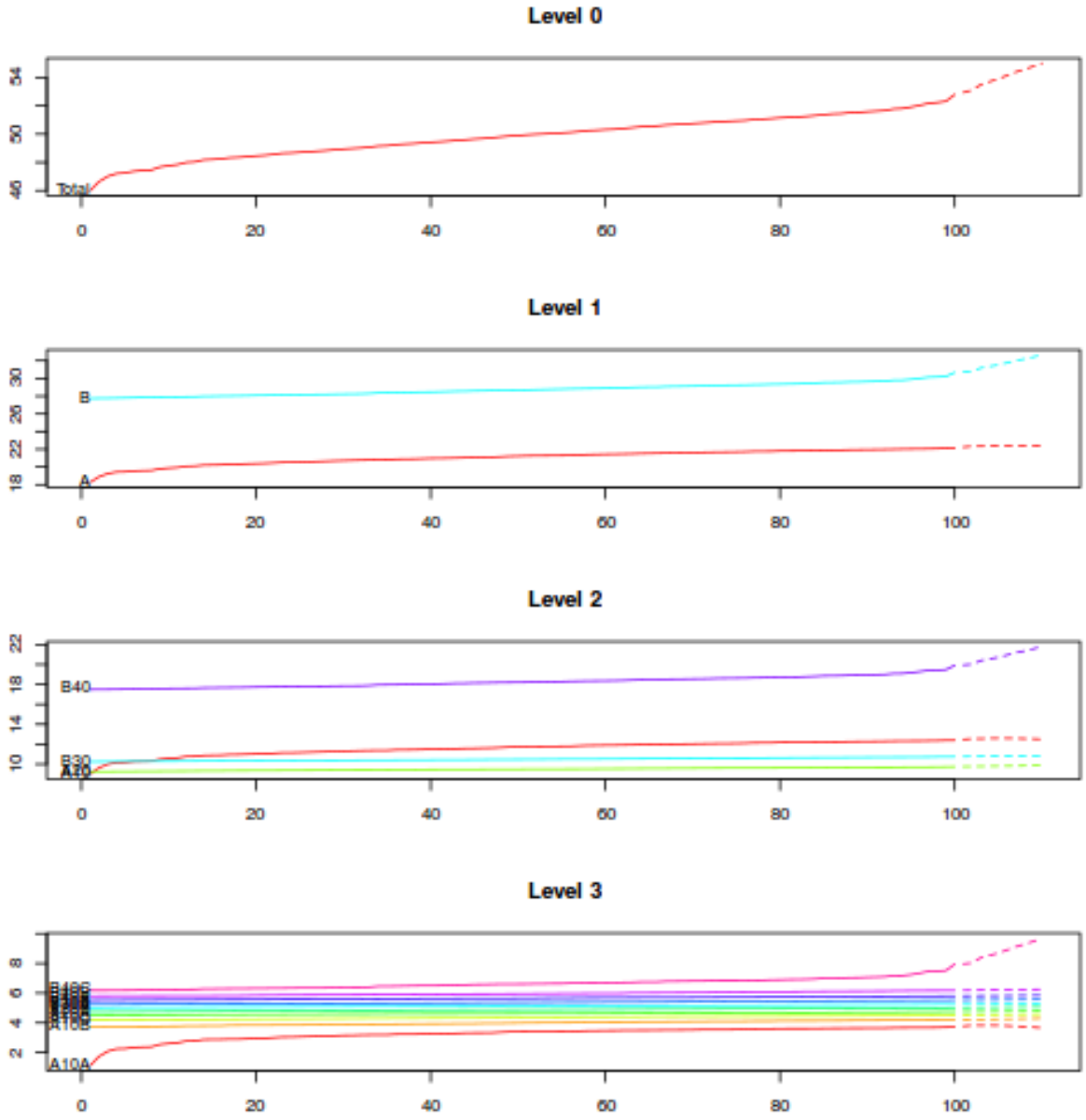
Forecasting Hierarchical or Grouped Time Series

```
# etc

fc <- forecast(y, h=10, fmethod="ets",
     parallel=TRUE, num.cores=2)

plot(fc)
```



Forecasting Hierarchical or Grouped Time Series

```
# arima

fc <- forecast(y, h=10, fmethod="arima",
     parallel=TRUE, num.cores=8)

plot(fc)
```


Level 0
Total
Level 1
B
A
Level 2
B40
B30
A20
Level 3
A10A