FUNCTIONAL CLUSTER ANALYSIS

10

Cluster analysis is an exploratory method used to find groups or clusters of similar data points. Classical *hierarchical* cluster analysis requires a matrix containing the distances between the items to be clustered. To compute a distance matrix, a *metric* or distance measure between any two data points is required. Functional methods offer many methods for computing distance matrixes, as was seen in Chapter 1, where the distance measure was taken as the integrated squared distance or l_2 distance between the two functions first derivatives.

Here we consider two examples involving daily measurements of precipitation and mean daily temperature at 35 Canadian weather stations over a one-year period. The functions provide an estimate of the expected daily temperatures at these stations. Ideally, additional years of observations would be desirable for analysis. We proceed by regularizing or smoothing the data, and then using the resulting functions to cluster the weather stations. As in the example in Chapter 1, the distance measure is obtained from the first derivatives of the smoothed functions.

CLUSTERING PRECIPITATION DATA

We first consider the daily precipitation data, and begin by smoothing the data using the construct function `fVector`. Daily precipitation is often highly variable with no precipitation on some days and a large amount of precipitation on others. Moreover, dry spells can last for quite some time, as can rainy periods. We are interested in the "expected" precipitation function, but we have only one year of measurements. Because we are interested in the first derivative function (the rate of change of the expected precipitation) and not in the measurement errors about the function, cross validation for these errors is not really helpful and we simply smooth until we seem to have an appropriate amount of smoothing by examining the first derivative of the smoothed function:

```
> sPrec <- fVector(fWeather$fPrec,
             penalty=list(lambda=100000, linDop=fDop(2)))
> par(mfrow=c(2, 1))
> plot(sPrec, main="Precipitation Functions")
> plot(fVector(sPrec, linDop=fDop(1)),
       main="Precipitation Derivatives")
```

In this code the unsmoothed precipitation functions are contained in the `fPrec` variable in the `fWeather` data frame (see the help file for `fWeather`). The smoothed functions are displayed in Figure 10.1. Looking at this Figure, the precipitation functions and their derivatives are reasonably smooth, giving a fairly good idea of the trend in the precipitation over the year that the data was measured.

Standardizing

Some patterns are evident in the expected precipitation functions, but precipitation is highly variable, depending to a considerable extent on local situations. For example, the western sides of mountains on the west coast of North America tend to get more rain as the weather station elevation increases (because of lifting), but the "trend" in weather variation is identical at all elevations. Because of this increase, we should be less interested in clustering based solely on the amount of precipitation, but rather on the rate of precipitation over the course of the year. Therefore, we standardize all weather stations to a fixed amount of fifty inches. This is accomplished by first

integrating the smoothed precipitation functions over the year to get the total amount of precipitation, and then adjusting each function so that its integral is 50:

```
> precInt <- fInt(sPrec)/50
> ssPrec <- fVector(t(t(getCoef(sPrec))/precInt),
                    getBasis(sPrec), getNames(sPrec))
> par(mfrow=c(2, 1))
> plot(ssPrec, main="Standardized Precipitation")
> plot(fVector(ssPrec, linDop=fDop(1)), main=
       "Derivative of the Standardized Precipitation")
```

This result is displayed in Figure 10.2, which shows that the patterns of precipitation are now much more apparent. Indeed, some stations report the bulk of their precipitation over the winter months, while in others, most precipitation occurs in the summer.

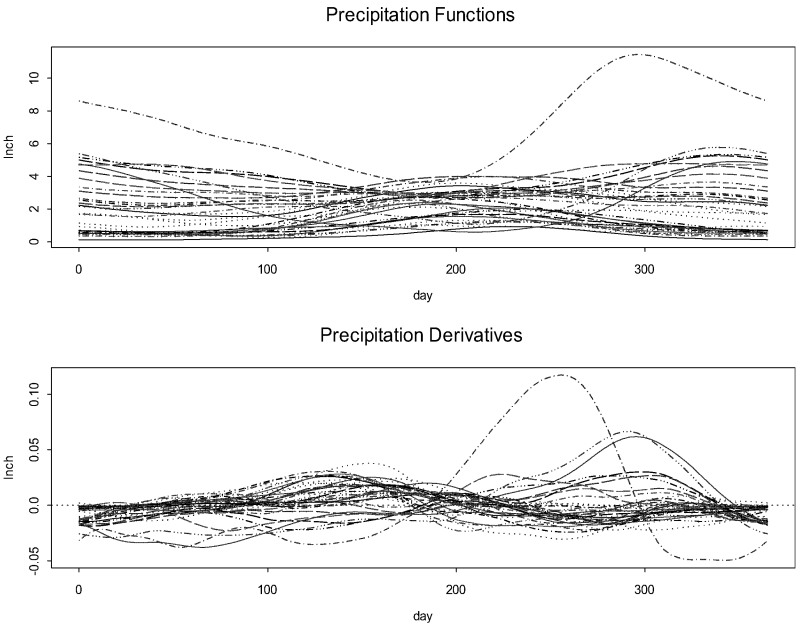

Figure 10.1: *Plot of the "expected" precipitation functions for 35 Canadian weather stations (top), with the first derivative (bottom).*

Clustering

To perform a hierarchical cluster analysis, we must first compute the between-station distance matrix using S+FDA function `fDist`. Here we use the integrated squared differences in the rate of change of precipitation as our clustering criterion. The S-PLUS function `hclust` is then used to cluster the data using a complete-linkage algorithm:

```
> ssPrecDist <- sqrt(fDist(ssPrec, linDop=fDop(1)))
> ssPrecClust <- hclust(ssPrecDist)
```

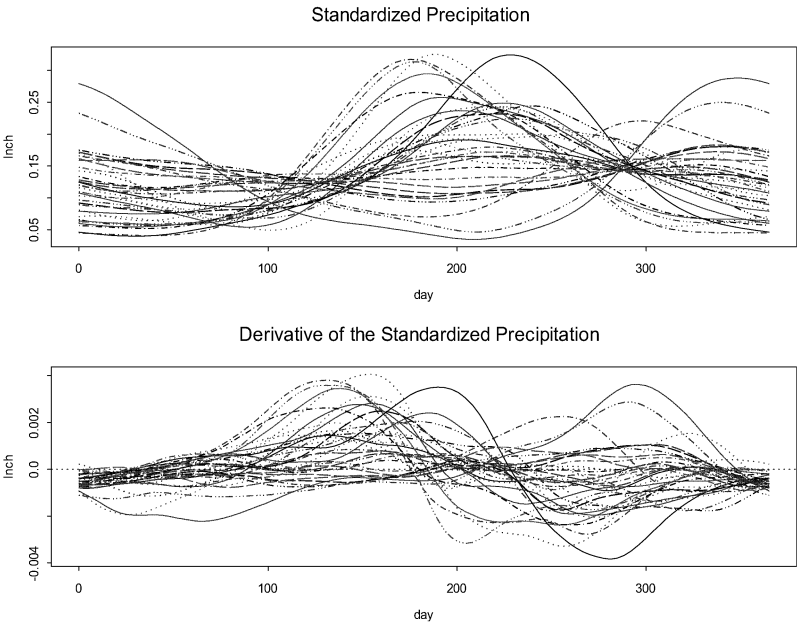

Figure 10.2: *The precipitation functions standardized to 50 inches per year.*

Complete linkage is chosen because we want the maximum within cluster distance to be small. Rather than plot the cluster tree, we plot the means of the seven cluster solution. The `cutree` function is used to identify stations within the seven clusters, as follows:

```
> ii <- cutree(ssPrecClust, k=7)
> ssPrecMean <- ssPrec[1:7]
> for(i in 1:7)
      ssPrecMean[i] <- mean(ssPrec[ii==i])
> par(mfrow=c(1, 1))
> plot(ssPrecMean,
```

```
        main="Mean Functions for Seven Clusters")
> legend(0, 0.325, as.character(1:7), lty=1:7)
```

The result is displayed in Figure 10.3, which shows that the cluster mean functions exhibit distinct patterns of precipitation.

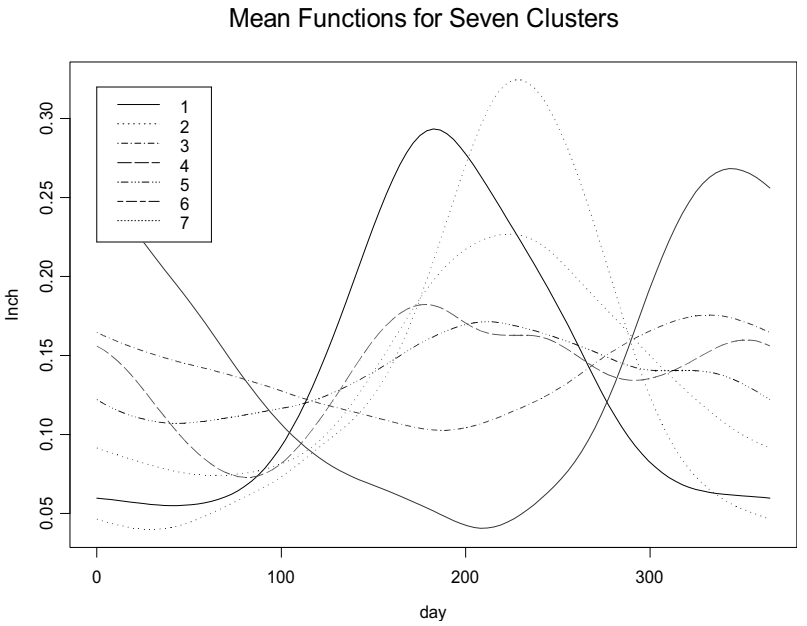

Figure 10.3: *Mean functions for the seven clusters.*

To see if the clustering result makes sense, we find the cities corresponding to the weather stations for each cluster:

- 1) Calgary, Edmonton, Prince Albert, Regina, The Pass, Winnipeg
- 2) Churchill, Dawson, Inuvik, Iqaluit, Schefferville, Thunder Bay, Uranium City, Whitehorse, Yellowknife
- 3) Charlottetown, Fredericton, Halifax, Prince Rupert, St. Johns, Sydney, Yarmouth
- 4) Kamloops, Prince George
- 5) Arvida, Bagotville, London, Montreal, Ottawa, Quebec, Sherbrooke, Toronto
- 6) Vancouver, Victoria

- 7) Resolute

Some of these results are expected, e.g., we would expect the far northern cities in cluster 2 to be similar, and Vancouver and Victoria, both in cluster 6, clearly share the same weather pattern being less that fifty miles apart and separated only by a body of water. On the other hand, we have no reason to believe that Halifax, on the east coast, and Prince Rupert, on the west coast, would have the same weather patterns, although they are both coastal cities. Clearly clustering based upon precipitation patterns is useful in finding groups of weather stations with related weather patterns, but precipitation patterns alone are insufficient to characterize the weather data.

CLUSTERING TEMPERATURE DATA

We now consider the temperature data. Unlike the precipitation data, here we do not standardize to a constant mean temperature. We look at the rate of change of the average daily temperature, rather than at the expected average daily temperature function. As with the precipitation data, smoothing is used to obtain an "expected" daily temperature from a single year of data.

The S+FDA statements used to smooth the data, perform a cluster analysis, and compute and plot the cluster mean functions are as follows:

```
> sTemp <- fVector(fWeather$fTemp,
                   penalty=list(lambda=50000, linDop=fDop(2)))
> sTempDist <- sqrt(fDist(sTemp, linDop=fDop(1)))
> sTempClust <- hclust(sTempDist)
> jj <- cutree(sTempClust, k=7)
> sTempMean <- sTemp[1:7]
> for(i in 1:7)
      sTempMean[i] <- mean(sTemp[jj==i])
> par(mfrow=c(2,1))
> plot(sTempMean, main=
      "Temperature Cluster Mean Functions")
> plot(fVector(sTempMean), main=
       "Derivatives of Temperature Cluster Mean Functions")
> legend(300, 0.4, as.character(1:7), lty=1:7)
```

The result is shown in Figure 10.4.

The temperature-based clusters are composed of the following stations. Here the number in parenthesis is the cluster number for the plot legend.

1. (3) Calgary, Edmonton, Kamloops, Prince George, Whitehorse
2. (1) Dawson, Prince Albert, Regina, The Pas, Uranium City, Winnipeg, Yellowknife
3. (6) Charlottetown, Halifax, St. Johns, Sydney, Yarmouth
4. (5) Churchill, Iqaluit, Schefferville
5. (4) Arvida, Bagotville, Fredericton, London, Montreal, Ottawa,

Quebec, Sherbrooke, Thunder Bay, Toronto

6. (7) Prince Rupert, Vancouver, Victoria

7. (2) Inuvik, Resolute

Again there are stations whose cluster assignment make sense (e.g., cluster 6 (7)), as well as clusters which are difficult to interpret (e.g, cluster 2 (1)).

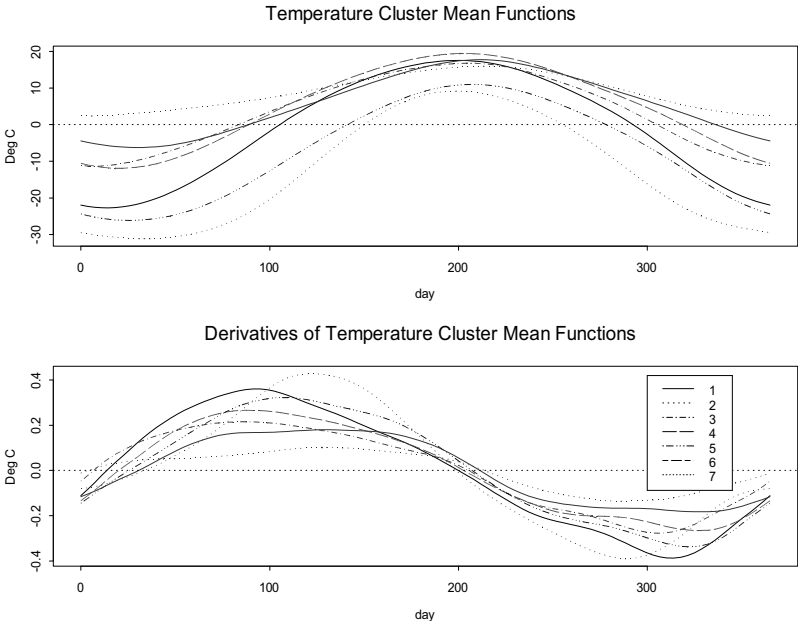

Figure 10.4: *The cluster mean functions for temperature (top) and its derivative (bottom).*

SUMMARY

Cluster analysis is an exploratory technique. Functional data methods offer the advantage of allowing a greater variety of clustering matrixes to choose from. The examples involving the clustering of Canadian weather stations are meant to be illustrative, since the known locations of weather stations can be used to infer which ones should exhibit similar weather patterns.The objective is not so much to find "real" clusters of stations, but rather to learn how the weather patterns at the different stations are related. Some of the clusters obtained consist of stations that are located in the same region, which we would expect similar to have weather patterns. Other aspects of the clustering are harder to interpret (e.g., assignment of Prince Rupert and Halifax to the same cluster), although they may also indicate relationships in weather patterns for stations at some distance from each other. A cluster analysis that accounted for both precipitation and temperature (and other weather related variables such as humidity) might be preferable, provided a suitable clustering metric could be found.

Methods for determining the number of clusters in functional cluster analysis are identical to those in the classical case, and thus are not discussed further here.

If groupings for some of the data are known in advance, it may be preferable to use a discriminant function analysis to find the variables and matrix that best classify the remaining observations. In the chapter on functional generalized linear models, we use a form of discriminant function analysis, functional logistic models, to classify the weather stations.