Bayes Linear Methods III - Analysing Bayes linear influence diagrams and Exchangeability in [B/D]

David Wooff and Michael Goldstein
University of Durham

[B/D] Home Page: http://fourier.dur.ac.uk:8000/stats/bd/

Abstract

Bayes linear methodology provides a quantitative structure for expressing our beliefs and systematic methods for adjusting these beliefs given observational data. In this report we introduce two aspects of the methodology. Firstly, we discuss Bayes linear influence diagrams, which are tools for representing and interpreting both the qualititative and quantitative structure. Secondly, we address the basic role of exchangeability. The computer language [B/D] (an acronym for *beliefs adjusted by data*) has been developed to implement Bayes linear methods, and can be used to construct such influence diagrams. In this technical report, we show how [B/D] is used to generate various types of influence diagram for a simple, but non-trivial, problem. We also present an example which shows how [B/D] is used to exploit exchangeability. The example concerns a collection of exchangeable dynamic linear models, representing the reduction of alumina by electrolysis.

1 Introduction

This tutorial document illustrates two important aspects of Bayes linear methodology: the generation of Bayes linear influence diagrams, and the exploitation of exchangeability. In each case we give examples and show how they are analysed using [B/D]. Section 2 considers the construction of influence diagrams for a problem in which two correlated linear regressions are used to model a certain chemical process. Section 3 shows how we model exchangeable structures, and how [B/D] can be used to exploit the exchangeability to suggest sampling strategies under various circumstances.

Two related technical reports are [6] and [15]. The former introduces the basic machinery of Bayes linear methodology and in particular describes the objectives of such analyses and the notation used. The latter shows how Bayes linear methods are applied to a simple example, and then how the [B/D] language can be used to carry out the analysis. This technical report assumes familiarity with its two predecessors. Detailed instructions on constructing influence diagrams using [B/D] can be found in [16].

2 Bayes linear influence diagrams

Graphical portrayals of the static and dynamic features of belief analysis and revision are basic to the understanding of large stochastic systems. An important graphical tool consists of the Bayes linear influence diagram. These diagrams show a number of nodes connected by arcs. The nodes represent meaningful collections of random quantities. An arc represents a potential for a source node to influence a destination node. Some nodes are or can be shaded to summarise the features of belief revision for the corresponding collection of random quantities. The arcs can be labelled to summarise information flow across the diagram, thus quantifying the degree of influence.

Bayes linear influence diagrams are introduced in [3], and further developed and considered in relation to a case study concerning stock flow and prediction in the brewing industry, in [7]. A more detailed exposition, using related brewing data, is given in [9]. Bayes linear influence diagrams are applied to the problem of revising beliefs about variance-covariance matrices in [14]. A useful starting reference for influence diagrams in general is given by [12].

2.1 Example problem

The following problem is considered in [1, Chapter 8]. A certain chemical process leads to a product Y and a by-product Z. The yields of both products are thought to be related to the temperature of the process, X^*. Twelve experiments are performed with different temperature settings (degrees fahrenheit) to study the effect of temperature. The data are shown in table 1, and plotted in Figure 1. In performing the analysis, we transform the temperature measurements to $X = (X^* - 177.86)/100$, where 177.86 is the mean temperature setting.

Table 1: Yield of two products for a chemical process

Temperature X^*	Main product Y	By product Z
161.30	63.70	20.30
164.00	59.50	24.20
165.70	67.90	18.00
170.10	68.80	20.50
173.90	66.10	20.10
176.20	70.40	17.50
177.60	70.00	18.20
181.70	73.70	15.40
185.60	74.10	17.80
189.00	79.60	13.30
193.50	77.10	16.70
195.70	82.80	14.80

2.2 The model

The model suggested to explain relationships between the quantities is as follows:

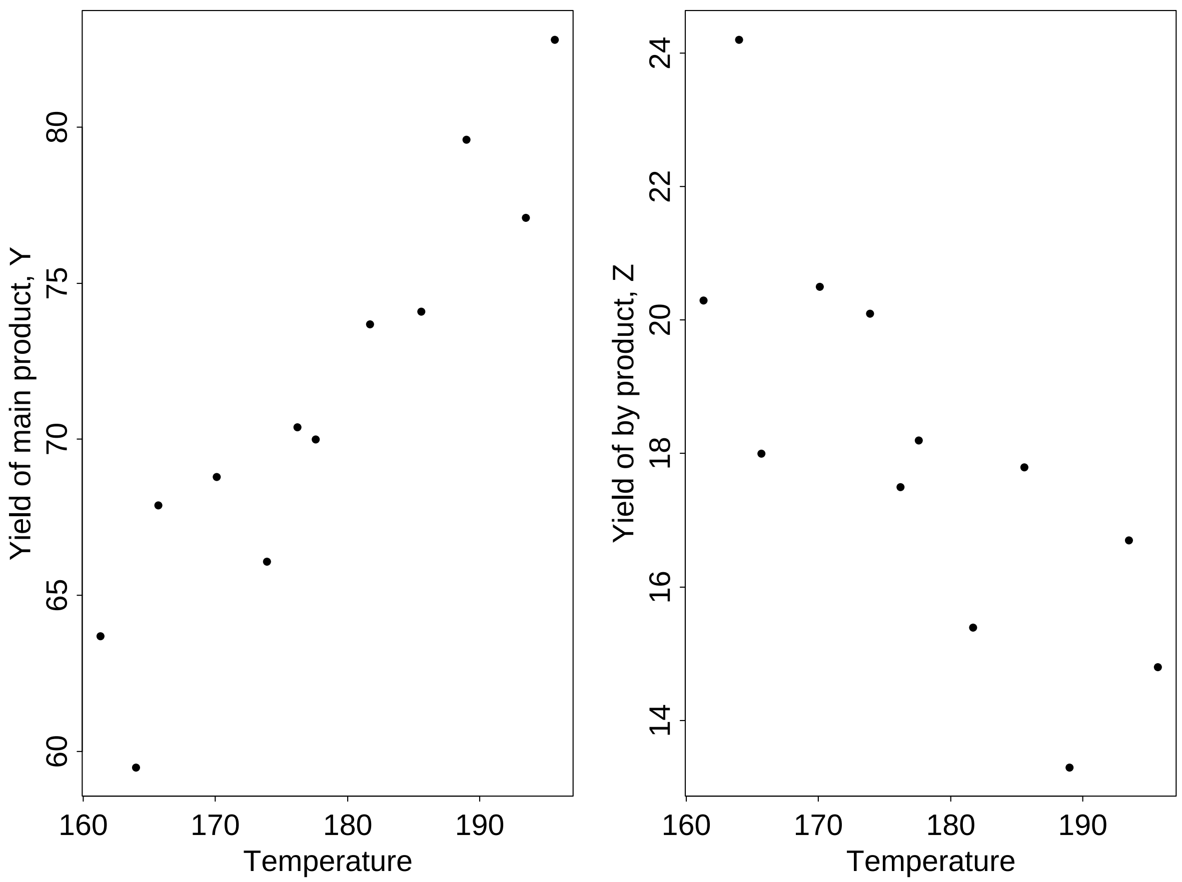
$$\begin{aligned} y_i &= a + bx_i + e_i \\ z_i &= c + dx_i + f_i, \quad i = 1, \ldots, 12. \end{aligned} \tag{1}$$

The model reflects the beliefs that the relationships between the yields Y, Z and the temperature X are approximately linear in X over the given range of temperature values. The intercept terms a, c indicate the yields for average temperature settings, whilst the slopes of the regressions are given by b, d. The models incorporate error components e_i, f_i. Separate runs of the experiment are independent; however, in any particular run it is felt that the error components will be correlated because slight aberrations in reaction conditions or analytical procedures could simultaneously affect both product yields. We will thus suppose that $e_1, e_2, \ldots$ are an uncorrelated sequence of error components with expectation zero and variance σ_e^2; that $f_1, f_2, \ldots$ are an uncorrelated sequence of error components with expectation zero and variance σ_f^2; and that all pairs of error components e_i, f_j are uncorrelated except for $\mathrm{Cov}(e_i, f_i) = \sigma_{ef}$.

2.3 Prior beliefs

It is necessary to specify prior beliefs over the four quantities A, B, C, D, and for the error components. Non-informative reference prior distributions are used for all these quantities by [1]. As we are concerned only with developing the example insofar as it illustrates the construction of influence diagrams, we provide here fictitious but credible prior beliefs about these quantities. In doing so, we have attempted to portray degrees of uncertainty that might plausibly be held by the process production manager. (For genuine *solvable* problems,

Figure 1: Yield of two products for a chemical process



we should be able at least to elicit prior information for the first and second order structure.) For the error quantities, we specify $\sigma_e^2 = 6.25$, $\sigma_f^2 = 4$, and $\sigma_{ef} = 2.5$, so that the the correlation between the two error components for any given run is about 0.5. (We chose these error variances by examining the residuals from separate least squares fits, so that the analysis would not be complicated by obvious conflicts between prior beliefs and the data.) We specify the following expectations and covariances between the regression parameters:

$$\mathrm{E}(\begin{bmatrix} a \\ b \\ c \\ d \end{bmatrix}) = \begin{bmatrix} 75 \\ 40 \\ 20 \\ -30 \end{bmatrix}, \tag{2}$$

$$\mathrm{Var}(\begin{bmatrix} a \\ b \\ c \\ d \end{bmatrix}) = \begin{bmatrix} 4 & -6 & -1 & 0 \\ -6 & 225 & 0 & -90 \\ -1 & 0 & 1 & -2.4 \\ 0 & -90 & -2.4 & 144 \end{bmatrix}. \tag{3}$$

The variance-covariance specifications indicate somewhat more uncertainty about the first regression equation than the second. Each slope quantity is considerably more uncertain than the corresponding intercept quantity: a production manager may know roughly the yield for an average temperature, but may have only sketchy beliefs concerning the direction and magnitude of the slope. We have specified the same degree of weak negative correlation, -0.3, between slope and intercept for each regression.

If we suppose that, at any given temperature, the total amount of product yield $(Y+Z)$ will fluctuate between fairly narrow limits, then the two yields should be strongly negatively correlated. We choose to introduce this information into the model by specifying negative correlations (each -0.5) between the two intercepts and between the two slopes. We treat the slope and intercept for different regressions as being uncorrelated, as we have exhausted our intuition about the physical process. This completes the prior specification process.

2.4 Establishing and generating the model

Before constructing any influence diagrams, it is necessary to prepare the model. To do this, consider the fragment of code shown in Figure 2, which does the following.

- The ELEMENT: command is used to introduce the four random quantities A, B, C, D together with prior expectations for them.
- The BASE: command is used to specify the collection $G = \{A, B, C, D\}$.
- The VAR: command is used to specify the variance-covariance structure over the collection G. These are explicit variance specifications.
- The ASSIGN: command is used to specify the modelled linear relationships. At this point, i is an index. The intention is later to construct the 24 quantities $y_1, \ldots, z_{12}$ from these assignments.
- The FVAR: command is used to specify beliefs for the error components. This command is used especially when the random quantities involved are uncorrelated with most other quantities, so that the complete description of beliefs about these quantities, and between these and other quantities, can be specified as simple functions. For random quantities such as these, the default action of [B/D] is to treat as zero any expectations and correlations not explicitly stated.
- The DATA: command is used to input the data shown in Table 1. The data follows directly after the command. For larger data sets it would be more usual to input the data from a different file.
- The FOR: command loop transforms each temperature, x_i by subtracting the average temperature and dividing by 100.
- The INDEX: and COBUILD: commands construct the 24 quantities $y_1, \ldots, y_{12}, z_1, \ldots, z_{12}$, taking into account the previously constructed quantities A, B, C, D and the functionally defined error components. The covariance structure between A, B, C, D and the y_i's and z_j's will also be correctly constructed. The

Figure 2: Generating the model

```
element: a=75, b=40, c=20, d=-30
base: g = a, b, c, d
var: v(1,g)
 4
-6   225
-1     0     1
 0   -90  -2.4  144

assign: y.i=a+(x(.i))b+e.i
assign: z.i=c+(x(.i))d+f.i
fvar: v(1,e.i,e.j)=(.i=.j)*6.25
fvar: v(1,f.i,f.j)=(.i=.j)*4
fvar: v(1,e.i,f.j)=(.i=.j)*2.5

data: <x,ydata,zdata>12
  161.30   63.70   20.30
  164.00   59.50   24.20
  165.70   67.90   18.00
  170.10   68.80   20.50
  173.90   66.10   20.10
  176.20   70.40   17.50
  177.60   70.00   18.20
  181.70   73.70   15.40
  185.60   74.10   17.80
  189.00   79.60   13.30
  193.50   77.10   16.70
  195.70   82.80   14.80

for: i=1,1,12 | data: x([i])=(x([i])-177.86)/100
index: i=1,1,12
cobuild: y.i,z.i

for: i=1,1,12
   data: y.[i](1)=ydata([i])
   data: z.[i](1)=zdata([i])
end:
```

Figure 3:

```
base: Y=y.$
base: Z=z.$
grid0:
g,      a,    b,    c,    d,    Y,     Z
0.15   0.80  0.80  0.80  0.80  0.40   0.40
0.50   0.80  0.60  0.40  0.20  0.65   0.35
0.07   0.06  0.06  0.06  0.06  0.035  0.035
 1     1     1     1     1     5      4
0.05   0.90  0.90  0.90  0.90  0.40   0.40
0.50   0.80  0.60  0.40  0.20  0.75   0.25

arc: Y,a,1 | arc: Y,b,1 | arc: Y,c,3 | arc: Y,d,1 | arc: Y,g,1
arc: Z,a,1 | arc: Z,b,1 | arc: Z,c,3 | arc: Z,d,1 | arc: Z,g,1

option: influence
control: overwrite
adjust: [ a / Y + Z ]
adjust: [ b / Y + Z ]
adjust: [ c / Y + Z ]
adjust: [ d / Y + Z ]
adjust: [ g / Y + Z ]
```

COBUILD: command, which is an extension of the BUILD: command introduced in [15], generates collections of quantities from given linear relationships simultaneously.

- Finally, within the FOR: loop, the DATA: command is used to attach individual observations to their corresponding random quantities.

See [15] for various tools for examining the specifications. For example, the LOOK: command with parameters e, b, d, a, v1, and r1 may be used respectively to view expectations, bases (collections), data, assignments, variance-covariance specifications, and variance-covariance specifications in correlation format.

2.5 Assessing information content and flow for each yield separately

There are many different explorations of the beliefs and data that we could make. For our first example, we will assess the value of observing the collection $Y = Y_1, \ldots, Y_{12}$ and the extra value of observing the collection $Z = Z_1, \ldots, Z_{12}$. We will draw influence diagrams to portray the expected effects of adjustment, calculated a priori, together with diagnostics comparing what we expected to what actually happened. We will examine individual aspects of the belief adjustment with respect to each of the four slope and intercept quantities A, B, C, D, as well as overall aspects of the belief adjustment with respect to the collection G containing them. To do this, it is necessary first to design the influence diagram, and second to carry out the relevant adjustments.

Generating the diagram

The steps required to produce the influence diagram Figure 4 are shown in Figure 3 as follows.

- The BASE: command is used to establish the collections Y and Z, each containing twelve quantities (the \$ symbol is used as a wildcard).

- The GRID0: command is used to specify node coordinates, size, and colour for the influence diagram. The command line is followed by seven rows of information, divided into columns, with each column corresponding to a different node. The information in each column is, respectively:

  1. The name of the element or base (collection) which the node is to represent.
  2. The x coordinate of the centre of the node. Coordinates are for the unit square, with (0,0) representing the bottom left position, and (1,1) the top right.
  3. The y coordinate of the centre of the node.
  4. The node radius. (A value of zero supplied here leads to [B/D] choosing a default radius which depends on screen resolution.)
  5. The node colour. For most implementations there will be at least 16 colours available, represented by the integers 0..15, representing white, black, and so forth. (See [16] for details.)
  6. The x coordinate of the centre of the text label associated with the node. Usually, the label is the same as the name of the element or collection represented by the node.
  7. The y coordinate of the centre of the text label associated with the node.

- The ARC: command is used to specify that an arc should be drawn between two given nodes, the name of the source node (representing information) being given before the name of the destination node (the collection being adjusted). The integer supplied determines the kind of arc drawn; the simplest are:

  - 1 - a directed arc.
  - 3 - a labelled directed arc, with the label portraying various kinds of information flow.
  - 5 - an unlabelled directed arc, together with a path correlation.
  - 7 - a labelled directed arc, together with a path correlation indicator.

  Thus, in this example we specify that labelled directed arcs should be drawn from each information source (the two nodes Y, Z) to each node to be adjusted (A, B, C, D and G). The LOOK: command with argument grid can be used to review grid and arc specifications.

- The OPTION: command is used with the `influence` parameter to direct [B/D] to produce an influence diagram.

- The CONTROL: command is used with the overwrite control to direct [B/D] *not* to clear the graphics screen at the end of every adjustment. This allows the influence diagram to be constructed from a sequence of related and/or unrelated adjustments. (With the overwrite control switched off, [B/D] waits for you to click on the graphics screen at the end of an adjustment, and any further adjustment will clear the graphics screen.)

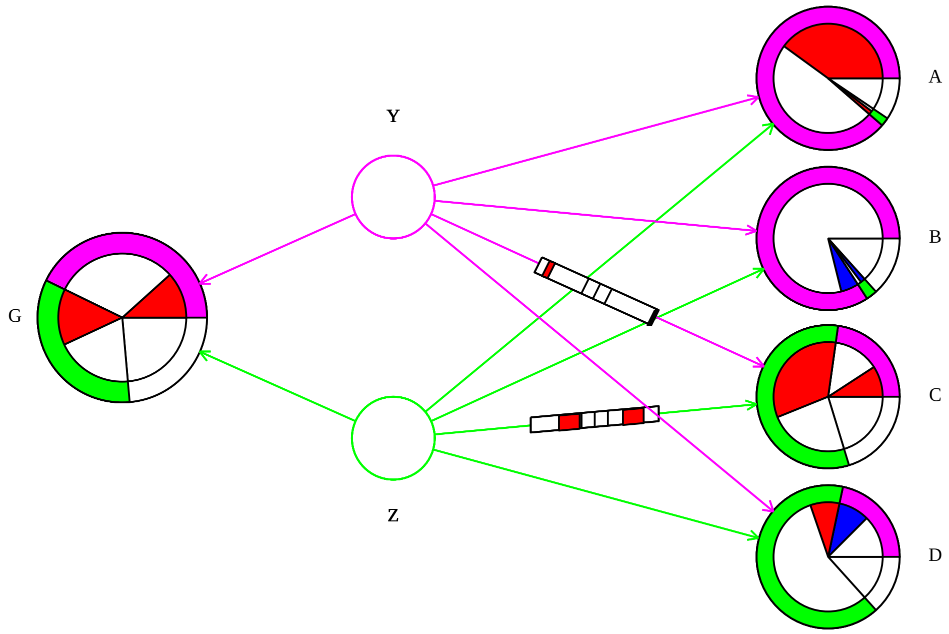
- Finally, a sequence of the adjustment commands, ADJUST:, are given for each of A, B, C, D and G. For example, the first will direct [B/D] to perform the adjustment of the element A firstly by the collection Y, and then *partially* by the collection Z.

Interpreting the diagram

The code in Figure 3 produces the influence diagram shown in Figure 4. Note that this diagram is generated from a colour postscript file produced directly from [B/D]; its translation into a monochrome image results in gray scale shadings which make the different colours hard to distinguish. For the inner node shading, the heavier shading corresponds to blue, and the lighter shading corresponds to red. The inner node of node D contains examples of both. Note also that all the features portrayed on the influence diagram are available as standard output, via the SHOW: command, as described in [15]. Table 2 contains all of the information summarised on the influence diagram.

The two sources of information (Y and Z) are drawn in the centre of the diagram. Nodes for the individual random quantities A, B, C, D are drawn on the right, and the node representing their collection, G, is drawn on the left. Directed arcs are drawn from the information sources to the nodes being adjusted to represent

Figure 4:



information flow. We called for the arcs to node C to be labelled. Various features of the adjustment are shown by (1) shading of the outer node; (2) shading of the inner node; (3) shading of the arc half label nearest the source node, and of the arc half label nearest the destination node.

1. The outer node sector represents 100% of the prior variance in the node. (For collections, a scale-free statistic of prior uncertainty in the collection as a whole is given instead.) Each information source I has the capacity to "explain" a proportion of the variance, or uncertainty, in a destination node V. We call this capacity the **resolution** in the node attributable to the information source I, abbreviated $\mathrm{R}_I(V)$. For the single adjustment of node V by node I, we shade the corresponding part of the outer sector, starting at 0 degrees and working counter-clockwise. The shading is in the same colour as the information source (and the arc connecting the two nodes) to aid identification of important contributory information sources. Consider for example the adjustment of the intercept term C by the information collection Y. C has prior variance 1.0, and adjusted variance 0.7720, a resolution of $\mathrm{R}_Y(C) = 0.2280$. Consequently, 22.8% of the outer sector is shaded. The remaining 77.2% represents variation (or uncertainty) in C unexplained by Y.

   The extra shading of the outer node expresses the portion of the remaining uncertainty in C that is explained by further fitting on the information source Z. This extra resolution is about 57.0% of prior, so that the overall fit by both Y and Z resolves $22.8 + 57.0 = 79.8$% of the uncertainty, leaving 20.2% of original variation in C unexplained by either Y or Z. The adjustment by Z *in addition* to the adjustment by Y is called a **partial** adjustment. The variance explained by Z alone is not shown by node shading.

2. Each adjustment corresponds to a shading of the outer node to represent variance explained. The matching inner node is now shaded according to the value of the **size ratio** for the adjustment, a diagnostic comparing predicted to observed effects (see [6] for a definition). Informally, large (small) values of the size ratio indicate that there were larger (smaller) squared changes in expectation than expected. They are represented by red (blue) shadings on the inner sectors of the nodes. The amount of shading corresponds to the magnitude of the diagnostic, with no shading for ratios equal to one (meaning that changes were about as expected). About half an inner sector is shaded when the changes in expectation are of the order of two standard deviations. In our example, the size ratio for the adjustment of node C by information source Y is $\mathrm{Size}_y(c) = 2.85 \approx 1.7^2$, so that the actual change in expectation is about 1.7 standard deviations larger than expected. As a result, the inner sector for this adjustment is roughly half shaded red.

   For the partial adjustment of C by the extra information source Z, having taken into account Y, the corresponding size ratio is $5.76 = 2.4^2$. Thus, not only is the contribution to variance reduction attributable to Z over and above Y substantial, but also the associated actual changes in expectation attributable to the partial adjustment are surprisingly large. Hence a substantial portion of the corresponding inner sector is shaded red.

3. When several information sources are being used to adjust an uncertain quantity, we portray the partial value of each information source by appropriate shading of labels attached to arcs. We measure two kinds of information flow between each information source I and node V being adjusted: (i) the capacity of each source used singly to reduce uncertainty in V, informally the information "leaving" I and (ii) the loss in variance reduction that would result if the node I were withdrawn from the overall adjustment, informally the information "arriving" at V from I. As for node shadings, each variance resolution is associated with a diagnostic measure comparing expected to actual behaviour.

   To illustrate, we have drawn two labels on the arcs connecting Y and Z to C. The information leaving node Y for node C is summarised in the half label nearer the node Y, whereas the information arriving at C from Y is summarised in the half label nearer the node C. We consider each half separately as representing 100% of the prior variance in the destination node.

   We partition the half label nearest Y into (1) the proportion of variance in C explained by the overall fit, $\mathrm{R}_{Y\cup Z}(C) = 79.8\%$; (2) the proportion of variance in C explained by Y alone, $\mathrm{R}_Y(C) = 22.8\%$. A portion of the latter is then shaded to depict the diagnostic measure. (As we adjusted firstly by Y, these half-label shadings correspond exactly to the node C shadings for the first adjustment. This will not be the case for the half-label shadings for information sources used in subsequent adjustments, such as Z here.)

   We partition the half label nearest C into (1) the proportion of variance in C explained by the overall fit, $\mathrm{R}_{Y\cup Z}(C) = 79.8\%$; (2) the proportion of explained variance in C lost by withdrawing the information source Y alone. This is a useful measure in the sense of parsimonious fitting, as some information sources might be found to be redundant as all their explanatory power is carried by other information sources. In this example, the only other information source is Z, and it alone explains about 75.2% of the variance in C. Consequently, the loss in explanation of variance when Y is removed from the overall fit is $79.8 - 75.2 = 4.6\%$ of prior. The value 4.6% is indicated on the influence diagram by a vertical line, giving a bar of width 4.6% which indicates the information arriving at C from Y. As before, a portion of this bar is now shaded to depict the standard diagnostic measure (the size ratio for the adjustment with Y extracted is $\text{Size}_Z(C) = 2.89 = 1.7^2$ standard deviations; somewhat larger changes in expectation than expected).

In conclusion, this label shows that for Y an appreciable amount of information leaves Y, but very little arrives. In each case, the associated standardised changes in expectation are rather larger than expected. If we do not have access to the information source Z, Y alone has some potential to reduce uncertainty in C. Otherwise, if Z is available, Y is not additionally valuable as it can contribute only an extra 4.6% variance reduction.

The influence diagram facilitates rapid appraisal of features of adjustments over complex stochastic structures. Suppose that we return to Figure 4 and review the main features. For the individual terms A, B, C, D most of the variance is explained by Y, Z together, as indicated by outer node shading. In each case, Y (magenta) is the first information source fitted, followed by Z (light green). Clearly Y is more informative for A, B and Z is more informative for C, D, as would be expected from the model (1), but Y singly is also quite informative for C, D, and Z for A, B. Both intercept terms A, C are associated with rather larger changes in expectation than were anticipated (indicated by red shading of the inner nodes) from both information sources. For the slope terms B, D, changes in expectation were less surprising: principally, the change in expectation for D from fitting on Y was rather less than expected. For the overall collection G, it is clear that both information sources are valuable, and that both lead to rather larger than expected changes in expectation.

For other than trivial examples, an initial graphical review of a complex system is followed by further graphical and numerical focussing on particular aspects to which we have been directed by our various diagnostic tools. In this way, we deconstruct the complex system into smaller manageable areas, pursuing the anomalies exhibited by the influence diagrams to whatever level of detail is necessary to unravel and comprehend surprising features of the adjustment process. For example, we have seen some rather larger than expected changes in expectation, indicating potentially serious conflicts between data and belief specifications. Here, we might begin by examining the model at the element (random quantity) level, to try to localise and identify contradictory features.

Table 2: Information flow summaries

Node shading	A	B	C	D	G
Resolution due to Y	0.8853	0.8406	0.2280	0.2163	0.4290
Size ratio	3.34	0.88	2.85	0.33	1.88
Partial resolution due to Z	0.0205	0.0270	0.5700	0.6511	0.3350
Size ratio	4.57	0.16	5.76	1.33	3.00
Resolution due to $Y \cup Z$	0.9058	0.8676	0.7979	0.8674	0.7640
Size ratio	2.20	0.73	1.27	0.51	2.03
Arc label shading					
Resolution, leaving Y	0.8853	0.8406	0.2280	0.2163	0.4290
Size ratio	3.34	0.88	2.85	0.33	1.88
Resolution, arriving from Y	0.7134	0.6522	0.0458	0.0270	0.3692
Size ratio	6.20	1.61	4.75	0.21	3.51
Resolution, leaving Z	0.1923	0.2153	0.7522	0.8404	0.3948
Size ratio	2.48	0.25	2.89	0.65	1.50
Resolution, arriving from Z	0.0205	0.0270	0.5700	0.6511	0.3350
Size ratio	4.57	0.16	5.76	1.33	3.00

Figure 5:

```
for: i = 1, 1, 3   |   base: batch1 = batch1,y.[i],z.[i]
for: i = 4, 1, 6   |   base: batch2 = batch2,y.[i],z.[i]
for: i = 7, 1, 9   |   base: batch3 = batch3,y.[i],z.[i]
for: i = 10, 1, 12 |   base: batch4 = batch4,y.[i],z.[i]

gclear:
xgrid:$
grid0:
g,    a,    b,    c,    d,    batch1, batch2, batch3, batch4
0.5  0.2   0.4   0.6   0.8    0.2     0.4     0.6     0.8
0.85 0.15  0.15  0.15  0.15   0.6     0.6     0.6     0.6
0.07 0.07  0.07  0.07  0.07    0       0       0       0
1    1     1     1     1       4       5       6       8
0.5  0.2   0.4   0.6   0.8    0.13    0.33    0.67    0.87
0.98 0.02  0.02  0.02  0.02   0.6     0.6     0.6     0.6


for: i=1,1,4
   arc: batch[i],a,1
   arc: batch[i],b,1
   arc: batch[i],c,1
   arc: batch[i],d,1
   arc: batch[i],g,5
end:


adjust: [ a / batch1 + batch2 + batch3 + batch4 ]
adjust: [ b / batch1 + batch2 + batch3 + batch4 ]
adjust: [ c / batch1 + batch2 + batch3 + batch4 ]
adjust: [ d / batch1 + batch2 + batch3 + batch4 ]
adjust: [ g / batch1 + batch2 + batch3 + batch4 ]
```

2.6 Influence diagrams summarising sequential knowledge acquisition

Our second example extends the use of portraying partial adjustments on influence diagrams, and introduces the notion of path correlation. Consider the model (1), and suppose that we observe runs of the experiment in batches with three observations of both product yields in each batch. Thus batch 1 consists of the observations $y_1, z_1, y_2, z_2, y_3, z_3$, and so forth. We will adjust the unknowns by each of the four batches sequentially. (Where experimentation is very expensive, it is obviously useful to identify the point at which further experimentation is expected to contribute little extra knowledge.)

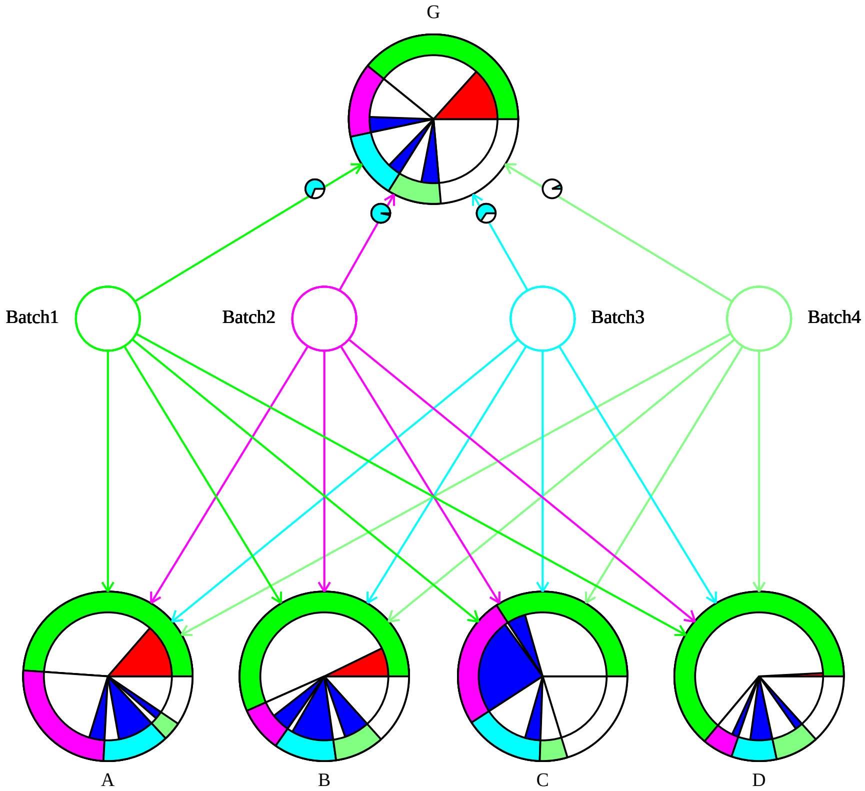
Generating the diagram

The steps required to produce the influence diagram Figure 6 are shown in Figure 5 as follows.

- Note that any options and controls established earlier in Figure 2 or Figure 3 remain in force.
- FOR: commands are used to establish the bases (collections) $batch1, \ldots, batch4$.
- The GCLEAR: command is used to clear the graphics screen of the influence diagram previously drawn.

- We want to draw an entirely new influence diagram, with different node coordinates and so forth. Thus we use the XGRID: command to delete currently existing node coordinate information before designing the new diagram. The $ symbol is a wildcard, representing all nodes.
- The GRID0: command is used as described previously to design the influence diagram.
- Several ARC: commands are used to connect nodes on the influence diagram. The arcs between the batch nodes and the nodes representing the individual quantities A, B, C, D are to be unlabelled directed. The arcs between the batch nodes and the node G representing the collection of individual quantities are to be unlabelled directed and display path correlations.
- Finally, the ADJUST: command is used several times. Each node A, B, C, D, G is adjusted initially by $batch1$, and then partially by $batch2$, $batch3$, and $batch4$ in turn.

Figure 6:



Interpreting the diagram

How should we interpret the influence diagram shown in Figure 6? Only the path correlation circles shown on arcs entering the G node are new features - the remainder we interpret as described above. For example, examining the outer shadings reveals that the information source *batch*1 is expected to explain at least 50% of the variance in each of the individual quantities excepting the intercept C, for which *batch*2 also is a valuable information source. Alternatively, for node D, the partial adjustment by *batch*2 following an initial adjustment by *batch*1 isn't expected to contribute much more to variance resolution.

Examining the diagnostic inner node shadings, we see mostly blue shadings, indicating smaller than expected changes in expectation. In particular, the change for node C induced by observing *batch*2 is very much smaller than expected. A few of the inner node shadings are red, but not so large as to indicate a serious contradiction between the belief inputs and the actual observations. The same can be said for the overall adjustment summary for the G node: there is an initial slightly larger than expected change in expectation due to observing the first batch; followed by three further diagnostically innocuous adjustments. The outer shadings for the G node suggest that each of the four batches contribute to the explanation of uncertainty across the collection G.

The small circles at the ends of the arcs drawn to the G node represent **path correlations** between the sources of information. These express the observed degree of concordance between two composite information sources in determining the revision of beliefs. Values near 1.0 indicate that two information sources are complementary: together they give larger changes in belief than would be given by the sum of their individual effects. Conversely, values near -1.0 indicate contradictory evidence, with some consequent cancelling out of their respective changes in belief. We show the magnitude of correlation by degree of shading of the small circle, and direction by colour: light blue for positive, light red for negative. In Figure 4, all the path correlations are positive. Respectively, they are 0.69, 0.97, 0.66, and 0.07. Thus, for example, the information in *batch4* is only weakly related to the information carried by the other three batches taken together.

3 Exploring sensitivity for exchangeable regressions

3.1 Exchangeability

The Bayes linear approach, which is based on partial belief specification with expectation as primitive, allows the straightforward construction of models reflecting second-order exchangeability. The approach requires only a small number of belief statements over observables.

Consider, for example, a simple case: a single infinite, exchangeable sequence of vectors $\{Y_1, Y_2, \dots\}$, for example, blood pressure, temperature, etc., measured on a sequence of patients. All that we must specify is the common expectation vector and variance matrix, for each Y_i, and the common covariance matrix between each pair Y_i and Y_j. From these three specifications, we can generate an exchangeability representation for the sequence as a combination of an underlying population vector $\mathcal{M}(Y)$ and individual variation vector $\mathcal{R}_i(Y)$, so that, for each i, $y_i = \mathcal{M}(Y) + \mathcal{R}_i(Y)$, where $\mathcal{R}_1(Y), \mathcal{R}_2(Y), \dots$ is an uncorrelated sequence of vectors, each with zero mean vector and the same variance matrix, all of which are uncorrelated with $\mathcal{M}(Y)$ (see [5]). Under this representation, we require as specifications the expectation vector and variance matrix over the mean components, $\mathrm{E}(\mathcal{M}(Y))$ and $\mathrm{Var}(\mathcal{M}(Y))$; and also the residual vector variance matrix, $\mathrm{Var}(\mathcal{R}_i(Y))$.

Learning about future values Y_k from observations on a collection $Y_1, \dots, Y_n$ is equivalent to learning about $\mathcal{M}(Y)$ in the exchangeability representation, so that prediction and estimation are equivalent. We may show further that the vector of sample means from the observed sample $Y_1, \dots, Y_n$ is Bayes linear sufficient for prediction of future observations, and that the canonical directions for an adjustment under exchangeability do not depend on the sample size. Therefore, the specification and adjustment of exchangeable beliefs is particularly simple.

Discussions of the principles and practice of adjustment for exchangeable beliefs are given in [5] and [2], and in detail in [10].

3.2 Problem summary

In [10] we considered an industrial process for extracting aluminium by electrolysis from a solution of alumina. Experiments are performed, under similar operating conditions, to measure the percentage concentration of alumina in solution every ten minutes, terminating when the concentration falls to a prespecified level. The measurements are the responses Y_{ij} representing, for runs j, the concentrations of alumina remaining in solution at thirteen time points i timed from the end of the experiment (the alumina level is essentially fixed at the end of the experiment). A key property of the process is that the experiments are exchangeable, and one aim of the analysis will be to assess the underlying mean values of the process. Data is available for three runs of the process.

The following exchangeable regressions model (an adaptation of the localised regression model introduced in [11] and of the dynamic linear model discussed in [13]), which takes into account various physical determinants and complicating features, was suggested to represent the process.

$$\begin{aligned} Y_{ij} &= a_j + ib_{ij} + \epsilon_{ij}. && (4) \\ \text{where} \quad \epsilon_{ij} &= E_{ij} + V_{ij} + H_{ij}, \\ V_{ij} &= V_{(i-1)j} + F_{ij}, && (5) \\ H_{ij} &= \phi H_{(i-1)j} + U_{ij}, \quad i \geq 2. && (6) \end{aligned}$$

The intercepts $\{a_j\}$ and the slopes $\{b_{ij}\}$ are second-order exchangeable over runs, j, and the $\{a_j\}$ are uncorrelated with the $\{b_{ij}\}$. We allow for the regression slopes to drift slowly over time, and so introduce strong correlations between neighbouring slopes and choose a correlation function so that the slopes have the markov property in being mutually uncorrelated given their neighbours. One way of achieving this aim is to use the following correlation function:

$$\text{Corr}(b_{ir}, b_{js}) = \rho^{\frac{|i-j|}{12}}, \quad \text{for all } r, s \quad \text{and} \quad i, j = 1, \ldots, 13,$$

where, for convenience, we define $0 < \rho < 1$ to be the correlation between the most distant slopes b_1 and b_{13}, so that values of $\rho^{\frac{1}{12}}$ describe the degree of local stability. As we expect only minor regression drift, we feel that values of about $\rho = 0.9$, giving a correlation of about 0.9913 between neighbouring slopes, might be appropriate. Smaller values of ρ will still yield high neighbouring correlations, but the dependencies between slopes will tail off much more quickly.

The error term ϵ_{ij} is constructed from uncorrelated components representing various error terms: simple noise, a random walk with drift, and an autoregressive term. (Full details of the development of the error structure can be found in [4].)

To assess the sensitivity of the model to changes in the stability parameter, and so discover whether our actual choice needs careful thought, we construct the model for a range of values of ρ and estimate the regression slopes. We assess the overall implications of changing the stability parameter by evaluating the structure of the resolution transform for each ρ.

3.3 Modelling and analysis

Model generation

Typical analyses including exchangeable quantities involve representations where we can resolve the information over the exchangeable quantities into mean and residual components. In particular, an exchangeable sequence can be treated as a special kind of element, where the sequence is summarised by its name, expectation, mean–component variance, and mean–plus–residual variance, together with the notion of repetition. Such elements are distinguished from ordinary elements by being associated with a *pair* of variance-covariance specifications over itself and with other quantities. To this purpose, [B/D] allows the creation of several alternative variance–covariance specifications over the same named quantity. Different belief storage areas, named *belief store 1, belief store 2, ...*, are set aside for this purpose. In this example, for all the quantities involved in the model we intend storing overall variances (for the mean plus residual components) in belief store 1, and any underlying mean component variances in belief store 2.

Thus, the model definition part of the [B/D] program for this example, shown in Figure 7, begins with the creation of one *element* named A to represent all three intercepts $\{a_j\}$. A has overall variance 0.058 (comprising 0.020 as the residual variance, and 0.038 as the underlying mean component variance) which we enter into variance–covariance store 1. Its mean–component variance is entered into store 2, and its prior expectation is set at 1.4.

The remainder of the code shown in Figure 7 deals with the specification and creation of the error structure as follows. The non-zero beliefs are introduced as constants, and the elements V_1 and H_1 are defined to terminate the random walk and autoregressive components (5), (6). Otherwise, beliefs over the error quantities are defined functionally, using the FVAR: command. These error quantities are essentially residual only and will relate to belief store 1: there is no need to enter zero variances into belief store 2 for their notional mean components because unspecified variances and covariances are taken to be zero by default.

We specify the expectation and variances and covariances over the slopes functionally, using the FVAR: and FE: commands. The overall (i.e. mean plus residual) variance structure is stored in belief store 1, and the variance structure for the underlying mean components is stored in belief store 2. The term named "#rho", intended to be the correlation ρ between neighbouring slopes, is a function which will be defined as appropriate before entry to the main creation and analysis subroutine.

Linear and linear recursive relationships may be established by [B/D] *assignments* by using the ASSIGN: command. When particular recursive relationships are encountered, the program tries, by tracking back through the recursion, to construct from it a definition which is finally non-recursive because of termination criteria already set. From the three assignments shown in Figure 7, and the earlier explicit and functional belief specifications, [B/D] is now able to deduce expectations, variances, and covariances for the $\{Y_{ij}\}$. Recall that the element whose name is "H.1" was explicitly introduced earlier to act as a termination criterion for the recursive component. For example, when the program later calls for H_3 it deduces that $H_3 = \phi^2 H_1 + \phi U_2 + U_3$ by using the recursive assignment and this termination criterion.

Figure 7:

```
keep log=alumlog
element: A=1.4
var: v(2,A)=0.038
var: v(1,A)=0.058
c: phi=0.7
c: varE=0.01
c: varF=0.01
c: varH=0.0204
element: v.1,H.1
var: v(1,V.1)=(%varF)
var: v(1,H.1)=(%varH/(1-sqr(%phi)))

fvar: v(1,U.t,U.i)=(.t=.i)*%varH
fvar: v(1,F.t,F.i)=(.t=.i)*%varF
fvar: v(1,E.t,E.i)=(.t=.i)*%varE
fvar: v(1,B.t,B.u)=0.0017*#rho^(abs(.t-.u))
fvar: v(2,B.t,B.u)=0.0016*#rho^(abs(.t-.u))
fe: B.t=0.1

assign: H.t=(%phi)H.(t-1)+U.t
assign: V.t=V.(t-1)+F.t
assign: Y.t=A+(.t)B.t+E.t+V.t+H.t
```

Belief construction and sensitivity analysis

We have defined - partly functionally and partly explicitly - the model, the quantities of interest, and the belief specifications that we need to carry out the adjustment. In particular, we have organised matters so that variances and covariances pertaining to underlying mean components are stored in variance–covariance store 2, and those pertaining to the mean–plus–residual components are in store 1. It remains only to define a range of values for ρ (we choose initially the three values $\rho = 0.05, \rho = 0.50, \rho = 0.95$), and for each such value to construct the corresponding $\{Y_{ij}\}$, and then perform the belief adjustments desired. The fragment of code in Figure 8 contains three parts: calls to the main analysis subroutine with different values of ρ, the subroutine itself, and the data. We concentrate only on describing the subroutine.

For each value of ρ, the INDEX: and COBUILD: commands are used to construct thirteen $\{b_i\}$ and thirteen $\{Y_i\}$ quantities. These will have expectations, variances, and underlying mean-component variances computed from their definitions: the former functionally, and the latter as linear combinations of intercept, slope, and error quantities. (Much computation is involved at this point, so the program may take some time to generate the quantities, depending on computer platform.)

Once constructed, the regression parameters $\{a, b_1, \dots, b_{13}\}$ are collected into the base named C, and the observables are gathered into the base whose name is Y. Observations from three experiments are then attached: these are shown at the foot of the listing.

For the analysis, we exploit the notion of *Bayes linear sufficiency* via exchangeability. Here the vector of averages of the observables, $\bar{Y} = \frac{1}{3}\sum Y_j$, over the three experiments is Bayes linear sufficient for the regression parameters. To perform the Bayes linear adjustment of the regression parameters by the observables, it is necessary to set various *belief source* controls. These indicate to [B/D] the belief stores in which the overall and underlying mean-component variance matrices are stored:

- The setting of the priorvar control determines the location of the variance-covariance specifications for $\mathrm{Var}(C)$: belief store 2. These constitute the prior variance matrix for the regression parameters.

Figure 8:

```
c q=0.05
m @analyse
c q=0.50
m @analyse
c q=0.95
m @analyse
return:

@analyse
f: rho=%q^(1/12)
index: i=1,1,13 | cobuild: Y.i, B.i
base: C=A, B$
base: Y=Y$
data: <Y> 3@alumdata
control: priorvar=2,infovar=1,modelvar=2,betweenvar=2
control: exchangeable,usedata
adjust: [ C / Y]
show: a
control: -usedata,obs=1
adjust: [ Y / Y ]
print: Most distant correlation: (%q)
print: canonical resolutions:   $
for: i=1,1,13 | print: [i]-(cr([i])) $
print: | print n for 50% reduction:     $
for: i=1,1,13 | print: [i]-(trunc(1/cr([i])-1)+1) $
print: | print: n for 50% reduction "(T): (findsize(Y,0.5))
return:


@alumdata
1.79 2.14 2.13 2.07 2.08 1.88 1.94 2.01 2.35 2.23 2.58 2.48 2.82
1.93 1.76 1.61 2.32 1.87 1.80 2.21 2.23 2.42 2.58 2.60 2.65 2.70
1.54 1.48 1.57 1.28 1.50 1.79 1.88 2.11 2.48 2.28 3.39 3.44 2.80
```

Table 3: Estimates of the regression parameters

Parameter	Prior expectation	Adjusted expectation $\rho = 0.05$	$\rho = 0.50$	$\rho = 0.95$
a	1.4	1.5206	1.5256	1.5096
b_1	0.1	0.0996	0.0877	0.0950
b_2	0.1	0.0944	0.0858	0.0949
b_3	0.1	0.0847	0.0827	0.0947
b_4	0.1	0.0856	0.0823	0.0947
b_5	0.1	0.0708	0.0767	0.0942
b_6	0.1	0.0653	0.0746	0.0941
b_7	0.1	0.0770	0.0806	0.0949
b_8	0.1	0.0815	0.0846	0.0956
b_9	0.1	0.0996	0.0955	0.0971
b_{10}	0.1	0.0894	0.0918	0.0971
b_{11}	0.1	0.1187	0.1105	0.0999
b_{12}	0.1	0.1108	0.1062	0.0995
b_{13}	0.1	0.0979	0.0972	0.0980

- The setting of the infovar control determines the location of the variance-covariance specifications for $\text{Var}(Y)$: belief store 1. These constitute the overall variance matrix for the observables.
- The setting of the modelvar control determines the location of the variance-covariance specifications for the underlying mean components $\text{Var}(\mathcal{M}(Y))$ of the observables: belief store 2.
- The setting of the betweenvar control determines the location of the variance-covariance specifications $\text{Cov}(C, Y) = \text{Cov}(C, \mathcal{M}(Y))$: belief store 2. These constitute the covariance matrix between the regression parameters and the observables.

The exchangeable and usedata controls specify that [B/D] should take into account data on the observables, and should use the internal routines which exploit exchangeability.

We next make two kinds of adjustments. Firstly, we adjust the regression parameters by the data (three observations over the $\{Y_i\}$), and display their adjusted expectations using the SHOW: command. (As the observables are Bayes linear sufficient for the regression parameters, [B/D] automatically obtains the general adjustment via the sample means only.) The adjusted expectations for the regression parameters are shown in Table 3 for the three values of the stability parameter ρ. This output indicates that although the model and observations are consistent with changes in expectation for these parameters (the adjusted expectation for the intercept rises slightly, etc.), changing the stability parameter makes little difference, so that the model is not particularly sensitive to choice of ρ for predicting individual Y_i values.

Our second adjustment assesses variance sensitivities in relation to changes in sample size. It is a property of such second-order exchangeable adjustments that the analysis for a sample of size n_1 can be deduced with almost no additional computation from the same analysis performed for a sample of size n_2. The analysis is particularly simple for adjustments where the observables are Bayes linear sufficient for the collection to be adjusted, as is the case here. Consequently, we tend to make an initial analysis assuming a sample of size one from which we may deduce easily the analysis for a general sample size n. Therefore, we use the usedata and obs controls to indicate that the analysis should assume an initial sample size of one and that it should ignore the actual observations available, and we then perform the adjustment of the underlying mean component vector $\mathcal{M}(Y)$ by the observables, exploiting Bayes linear sufficiency automatically. [B/D] deduces here that by Y we mean $\mathcal{M}(Y)$ from the settings of the belief source controls described above.

Much of the output available from an adjustment is available interactively as further input to the program. These include adjusted expectations, variances and covariances; the resolution transform and its canonical structure, and so forth. Various other functions are available to exploit exchangeability as appropriate. For the sensitivity study here, we output, for each value of ρ, the following:

Table 4: Measures for assessing sensitivity over the model

ρ	n_{tr}	n_{max}	λ_1	λ_2	λ_3	λ_4	λ_5	λ_6	λ_7	λ_8	λ_9	λ_{10}	λ_{11}	λ_{12}	λ_{13}
0.05	2	23	0.68	0.64	0.59	0.55	0.499	0.458	0.417	0.362	0.298	0.231	0.163	0.099	0.043
0.50	6	79	0.59	0.43	0.34	0.29	0.252	0.216	0.181	0.146	0.112	0.081	0.054	0.031	0.013
0.95	57	1001	0.60	0.31	0.05	0.03	0.027	0.022	0.017	0.013	0.010	0.007	0.004	0.003	0.001

- the values of all thirteen canonical resolutions;
- the sample size required to obtain a resolution of at least 50% in each canonical direction;
- via the **findsize**($\cdot$) function, the sample size required to guarantee a resolution of at least 50% over the collection $\mathcal{M}(Y)$ considered jointly (see [6] for a description of uncertainty measures over collections).

For the analysis of sensitivity over the model we examine both the canonical resolutions and some implications of changing the sample size. For three values of ρ, Table 4 shows two sample sizes and thirteen canonical resolutions. The first sample size, n_{tr}, is the sample size needed to achieve a 50% reduction in uncertainty over the collection overall, as measured by the trace of the resolution transform. The second, n_{max}, is the sample size needed to guarantee a variance reduction at least 50% in *every* linear combination of the quantities $\{\mathcal{M}(Y_i)\}$. The canonical resolutions indicate the effective dimension of the model and the speed of variance reduction as the sample sizes increase.

We discover that these values are highly sensitive to the choice of stability parameter ρ. For $\rho = 0.95$ the smallest canonical resolution is $\lambda_{13} = 0.001$, so that for a sample size $n = 1$ we can guarantee a reduction in uncertainty of only 0.1% in every linear combination of the mean components. This guaranteed reduction rises to 50% if we take a sample size $n = 1001$, whereas we need take only $n = 79$ to achieve the same reduction when we choose $\rho = 0.5$. For uncertainty in the collection overall the picture is similar. Examining the canonical resolutions, for $\rho = 0.95$, the model is dominated by two canonical quantities with resolutions of 0.60 and 0.31 respectively. The remaining canonical quantities have small resolutions, so that large sample sizes will be needed to reduce their variances appreciably. Therefore the learning process for the model with $\rho = 0.95$ is essentially two-dimensional. (This should not surprise us: taking $\rho = 1$ forces (4) to become a simple regression model with two parameters, intercept and common slope.)

As we reduce the stability parameter, the dynamics of adjustment change: we increase the number of canonical quantities having noticeable variance reductions for small sample sizes, and we learn more quickly about the collection overall, so that for $\rho = 0.05$ we can learn about almost all combinations of the Y values. Therefore, the model is very sensitive to choice of ρ, for learning about changes in Y over time.

3.4 Other remarks

To explore further the differences between the prior and adjusted expectation and variance-covariance specifications, we could use the techniques described in [4] and [8]. For example, these techniques would identify the linear combinations of the mean-component quantities $a, b_1, \ldots, b_{13}$ whose specifications under alternative models were most and least different.

4 Acknowledgements

The program [B/D] has been developed by us with the help of grants from the Science and Engineering Research Council. We are grateful to our colleague Dr Peter Craig for invaluable technical aid.

References

[1] G. E. P. Box and G. C. Tiao. *Bayesian inference in statistical analysis.* Addison-Wesley, 1973.

[2] M. Farrow and M. Goldstein. Bayes linear methods for grouped multivariate repeated measurement studies with application to crossover trials. *Biometrika*, 80(1):39–59, 1993.

[3] M. Goldstein. Influence and belief adjustment. In J.Q. Smith and R.M. Oliver, editors, *Influence Diagrams, Belief Nets and Decision Analysis*. Wiley, Chichester, 1990.

[4] M. Goldstein. Belief transforms and the comparison of hypotheses. *The Annals of Statistics*, 19:2067–2089, 1991.

[5] M. Goldstein. Revising exchangeable beliefs: subjectivist foundations for the inductive argument. In P. Freeman and A. F. M. Smith, editors, *Aspects of Uncertainty: A Tribute to D. V. Lindley*. Wiley, 1994.

[6] M. Goldstein. Bayes linear methods I - Adjusting beliefs: concepts and properties. Technical Report 1995/1, Department of Mathematical Sciences, University of Durham, 1995.

[7] M. Goldstein, M. Farrow, and T. Spiropoulos. Prediction under the influence: Bayes linear influence diagrams for prediction in a large brewery. *The Statistician*, 42(2):445–459, 1993.

[8] M. Goldstein and D. A. Wooff. Robustness measures for Bayes linear analyses. *Journal of Statistical Planning and Inference*, 40(2-3):261–277, 1994.

[9] M. Goldstein and D. A. Wooff. Bayes linear computation: concepts, implementation and programming environment. *Statistics and Computing*, 5:327–341, 1995.

[10] M. Goldstein and D. A. Wooff. Adjusting exchangeable beliefs. *Biometrika*, 85(1):39–54, 1998.

[11] A. O'Hagan. Curve fitting and optimal design for prediction (with Discussion). *Journal of the Royal Statistical Society*, 40(1):1–42, 1978.

[12] J. Q. Smith and R. M. Oliver, editors. *Influence Diagrams, Belief Nets and Decision Analysis*. Wiley, Chichester, 1990.

[13] M. West and P. J. Harrison. *Bayesian forecasting and dynamic models*. Springer, New York, 2nd edition, 1997.

[14] D. J. Wilkinson and M. Goldstein. Bayes linear adjustment for variance matrices. In J.-M. Bernardo et al., editors, *Bayesian Statistics 5*, pages 791–800, Oxford, 1996. University Press.

[15] D. A. Wooff. Bayes linear methods II - An example with an introduction to [B/D]. Technical Report 1995/2, Department of Mathematical Sciences, University of Durham, 1995.

[16] D. A. Wooff. [B/D] Manual. Technical Report 1995/4, Department of Mathematical Sciences, University of Durham, 1995.