Conditional uncorrelation (normal distribution), chained linear predictions: examples

Video-presentation: http://personales.upv.es/asala/YT/V/condnocoEN.html

This code ran without errors in Matlab `R2022a`

Table of Contents

Conditional Independence (revision)

Suppose we have three random variables a, b, c.

- b and c are said to be "conditionally independent when $a = a_m$ " if

$$p(b|c = c_m, a = a_m) = p(b|a = a_m)$$

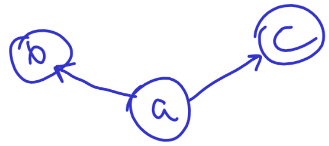
or

$$p(c|b = b_m, a = a_m) = p(c|a = a_m).$$

Equivalently, if $p(b, c|a = a_m) = p(b|a = a_m) \cdot p(c|a = a_m)$.

- If b and c and are conditionally independent given any possible observation of a, then we say that they are "conditionally independent given a".

This relationship allows expressing probabilistic models in graphical diagrams such as:



which are called "Bayesian networks" in a general case.

Condicionally uncorrelated Variables (normal case)

Statistical independence is harder to prove than incorrelation (at least, approximately, given a sufficient amount of data). No correlation is equivalent to inability to set up any useful "linear" predictor.

If we have a symmetric covariance matrix:

$$\Sigma := \begin{pmatrix} \Sigma_{aa} & \Sigma_{ab} & \Sigma_{ac} \\ \Sigma_{ba} & \Sigma_{bb} & \Sigma_{bc} \\ \Sigma_{ca} & \Sigma_{cb} & \Sigma_{cc} \end{pmatrix}$$

then, the best (minimum error variance) linear prediction of (b, c) given a is:

$$\begin{pmatrix} \widehat{b}_{|a} \\ \widehat{c}_{|a} \end{pmatrix} = \begin{pmatrix} \Sigma_{ba} \\ \Sigma_{ca} \end{pmatrix} \Sigma_{aa}^{-1} \cdot a$$

with a prediction error variance given by:

$$\Sigma_{err} := \begin{pmatrix} \Sigma_{bb} & \Sigma_{bc} \\ \Sigma_{cb} & \Sigma_{cc} \end{pmatrix} - \begin{pmatrix} \Sigma_{ba} \\ \Sigma_{ca} \end{pmatrix} \Sigma_{aa}^{-1} \begin{pmatrix} \Sigma_{ba}^T & \Sigma_{ca}^T \end{pmatrix} = \begin{pmatrix} \Sigma_{bb} - \Sigma_{ba}\Sigma_{aa}^{-1}\Sigma_{ba}^T & \Sigma_{bc} - \Sigma_{ba}\Sigma_{aa}^{-1}\Sigma_{ca}^T \\ \Sigma_{cb} - \Sigma_{ca}\Sigma_{aa}^{-1}\Sigma_{ba}^T & \Sigma_{cc} - \Sigma_{ca}\Sigma_{aa}^{-1}\Sigma_{ca}^T \end{pmatrix}$$

Hence, if $\Sigma_{err\,(1,2)} = \Sigma_{bc} - \Sigma_{ba}\Sigma_{aa}^{-1}\Sigma_{ca}^T = 0$, i.e., $\Sigma_{bc} = \Sigma_{ba}\Sigma_{aa}^{-1}\Sigma_{ca}^T$, then Σ_{err} is diagonal, so extra information about $e := c - \widehat{c}_{|a}$ is of no use to predict (linearly) b with greater precision than the already available $\widehat{b}_{|a}$, and likewise happens with predictions of c with additional info on b. They are conditionally incorrelated.

In the case of normal distribution, this would serve to "exactly" determine the conditional distribution in mean and variance, so we will call them "conditionally uncorrelated", but, obviously, it is only true in certain cases, for example with the extra assumption of normality. Otherwise, the true "conditional" probability distribution may not have the mean and variance from the best linear prediction formulae.

In fact, in the multivariate normal distribution case, uncorrelation implies independence, so these variables would also be **conditionally independent**. So, the focus on "uncorrelation" in this material is just for numerical/illustrative reasons.

Example 1: sum of two random variables

The "sum" $a = b + c$, assuming variance of b=2, variance of c=1, b y c independent, has a covariance matrix:

$$E[ab] = E[(b+c)b] = E[b^2] + E[bc] = E[b^2],$$

$$E[ac] = E[(b+c)c] = E[bc] + E[c^2] = E[c^2],$$

$$E[a^2] = E[(b+c)^2] = E[b^2 + c^2 + 2bc] = E[b^2] + E[c^2]$$

```
VCabc=[3 2 1;2 2 0;1 0 1]
```

```
VCabc = 3×3
     3     2     1
     2     2     0
     1     0     1
```

```
Resid=VCabc(2:3,2:3)-VCabc(2:3,1)*inv(VCabc(1,1))*VCabc(1,2:3)
```

```
Resid = 2×2
    0.6667   -0.6667
   -0.6667    0.6667
```

```
[V,D]=eig(Resid)
```

```
V = 2×2
   -0.7071   -0.7071
   -0.7071    0.7071
D = 2×2
    0.0000         0
         0    1.3333
```

The null eigenvalue indicates that there is a "deterministic" relationship in there (clearly, if I know the sum and one of the variables, the remaining one can be deterministically obtained).

Example 2

```
Sigma=@(s_bc) [2 3 4;3 10 s_bc;4 s_bc 20]; %cov(b,c) adjustable parameter
S0=Sigma(0) %b and c are uncorrelated.
```

```
S0 = 3×3
     2     3     4
```

```
     3    10     0
     4     0    20
```

```
eig(Sigma(0)) %must be positive def.
```

```
ans = 3×1
    0.2648
   10.8492
   20.8860
```

2.1: Incorrelated, but "condicionally" correlated

```
S0(2:3,2:3)% diagonal
```

```
ans = 2×2
    10     0
     0    20
```

```
vzaerrbc=S0(2:3,2:3)-S0(2:3,1)*inv(S0(1,1))*S0(1,2:3) %NOT diagonal
```

```
vzaerrbc = 2×2
    5.5000   -6.0000
   -6.0000   12.0000
```

```
eig(vzaerrbc)% some randomness remains, not like in the "sum" example.
```

```
ans = 2×1
    1.9263
   15.5737
```

Given the conditional correlation, the best prediction of "c", given "a" and "b", will have a residual prediction error variance given by:

```
S0(3,3)-S0(3,1:2)*inv(S0(1:2,1:2))*S0(1:2,3)
```

```
ans = 5.4545
```

which is better than "20" (equal to the marginal) with only the knowledge of "b", and better than the variance of the best prediction knowing only "a", which is:

```
S0(3,3)-S0(3,1)*inv(S0(1,1))*S0(1,3)
```

```
ans = 12
```

*Note that the result is the same if I used the information on "b" with the residual covariance arising from a previous use of "a":

```
vzaerrbc(2,2)-vzaerrbc(2,1)*inv(vzaerrbc(1,1))*vzaerrbc(1,2)
```

```
ans = 5.4545
```

2.2: correlated but condicionally uncorrelated

```
bc_ci=S0(2,1)*inv(S0(1,1))*S0(1,3)
```

```
bc_ci = 6
```

```
Sci=Sigma(bc_ci)
```

```
Sci = 3×3
     2     3     4
     3    10     6
     4     6    20
```

```
eig(Sci) %es def+
```

```
ans = 3×1
    0.7521
    7.3408
   23.9070
```

```
Resid_given_a=Sci(2:3,2:3)-Sci(2:3,1)*inv(Sci(1,1))*Sci(1,2:3) %sale diagonal, cond. ir
```

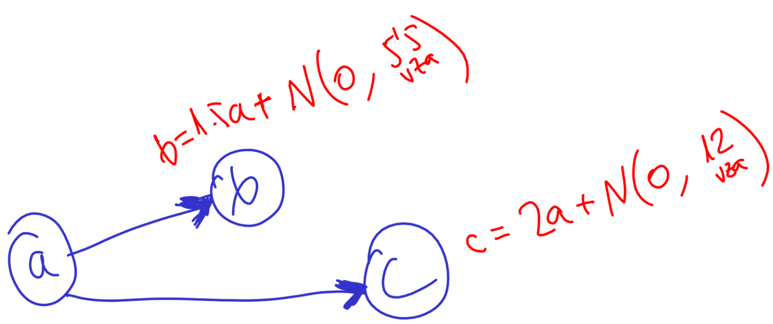
```
Resid_given_a = 2×2
    5.5000         0
         0   12.0000
```

Bayesian network representation

- Option 1:

```
bestpred_given_a=Sci(2:3,1)*inv(Sci(1,1))
```

```
bestpred_given_a = 2×1
    1.5000
    2.0000
```



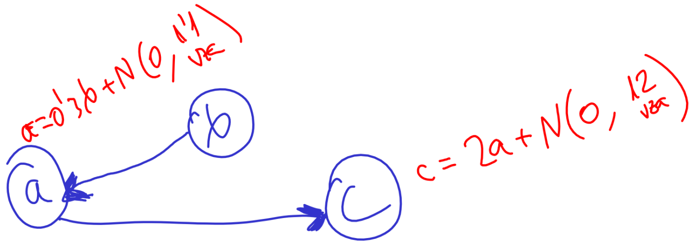
- Option 2:

```
bestpred_given_b=Sci([1 3],2)*inv(Sci(2,2))
```

```
bestpred_given_b = 2×1
    0.3000
    0.6000
```

```
Resid_given_b=Sci([1 3],[1 3])-Sci([1 3],2)*inv(Sci(2,2))*Sci(2,[1 3])
```

```
Resid_given_b = 2×2
    1.1000    2.2000
    2.2000   16.4000
```



- Option 3:

```
bestpred_given_c=Sci([1 2],3)*inv(Sci(3,3))
```

```
bestpred_given_c = 2×1
    0.2000
    0.3000
```

```
Resid_given_c=Sci([1 2],[1 2])-Sci([1 2],3)*inv(Sci(3,3))*Sci(3,[1 2])
```

```
Resid_given_c = 2×2
    1.2000    1.8000
    1.8000    8.2000
```