M.Sc. Practical HT2007 week 6

Graphical models with ordinal data

This practical exercise uses MIM and the interface package mimR to analyse data from a study of types of housing in Denmark. The data were originally published by Madsen (1976) but have been analysed often since, for example in MASS, where the dataset is available as `housing`. The variables involved are

Satisfaction: The degree of satisfaction: low, medium, high.
Influence: The degree of influence on decisions: low, medium, high.
Contact: The degree of contact experienced: low, high.
Type: The type of housing: tower block, apartment, atrium house, and terraced house.

In principle, mimR will start up MIM when needed. However, this may give error messages on some machines. To avoid this, it is best to start MIM manually before loading mimR.

So, first start MIM.

Next start R and change directory to the appropriate one.

Then load mimR (which also loads MASS) and for visualization also Rgraphviz.

```
> library(mimR)
> library(Rgraphviz)
```

Next load data.

```
> data(housing)
> housing[1:5, ]
```

```
     Sat   Infl  Type Cont Freq
1    Low    Low Tower  Low   21
2 Medium    Low Tower  Low   21
3   High    Low Tower  Low   28
4    Low Medium Tower  Low   34
5 Medium Medium Tower  Low   22
```

We need to transform data to a gmData object. Since the data frame uses Frequency as a separate response variable, we first have to change format to a cross-classification table.

```
> housingTab <- xtabs(Freq ~ Sat + Infl + Type + Cont, data = housing)
> ht <- as.gmData(housingTab)
```

The gmData object can now be displayed by the command

```
> ht
```

```
  name letter factor levels
1  Sat      a   TRUE      3
2 Infl      b   TRUE      3
3 Type      c   TRUE      4
```

```
4 Cont      d   TRUE      2
Data origin :      table
```

Note that the variable names are taken from the original dataframe. The 'letters', a,b,c,d are names used by MIM.

Two of the variables are clearly ordinal. This is declared as follows

```
> ordinal(ht) <- c("Sat", "Infl")
```

The gmData object has now absorbed this information.

```
> ht
```

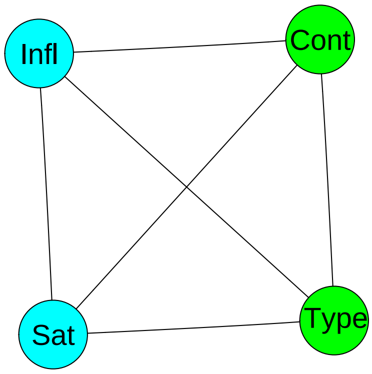
```
  name letter factor levels
1  Sat      a   TRUE      3
2 Infl      b   TRUE      3
3 Type      c   TRUE      4
4 Cont      d   TRUE      2
Ordinal     :      Sat Infl
Data origin :      table
```

Next we specify the saturated model as a MIM-model. The model specification syntax for discrete data writes interaction terms in usual R syntax, then adds a double slash:

```
> msat <- mim("Sat:Infl:Cont:Type//", data = ht)
```

The dependence graph of the model can be displayed as

```
> display(msat)
```



Next, we try to simplify the model by independence tests for all pairs of variables.

At this point, MIM is used as the calculator and mimR will automatically start MIM whenever needed if this has not been done already.

```
> testdelete("Sat:Infl", msat)
```

```
test: Chi-squared method: asymptotic
stat: 135.69 df: 32 P: 0
```

```
> testdelete("Sat:Cont", msat)

test: Chi-squared method: asymptotic
stat: 32.871 df: 24 P: 0.107

> testdelete("Sat:Type", msat)

test: Chi-squared method: asymptotic
stat: 99.094 df: 36 P: 0

> testdelete("Infl:Cont", msat)

test: Chi-squared method: asymptotic
stat: 33.721 df: 24 P: 0.09

> testdelete("Infl:Type", msat)

test: Chi-squared method: asymptotic
stat: 43.755 df: 36 P: 0.175

> testdelete("Cont:Type", msat)

test: Chi-squared method: asymptotic
stat: 64.349 df: 27 P: 0
```

It appears that the least significant edge is between Influence and Type, but the relationships between Satisfacton and Contact and between Influence and Contact are not significant either.

By using stepwise search with options, this can all be done with a single command:

```
> stepwise(msat, arg = "o")

Coherent Backward Single-step Selection.
Fixed edges: none.
Critical value:   0.0500
Decomposable mode, Chi-squared tests.
DFs adjusted for sparsity.
Model: abcd
Deviance:   0.0000 DF:   0 P:  1.0000
    Edge        Test
Excluded   Statistic DF          P
    [ab]    135.6898 32      0.0000 +
    [ac]     99.0937 36      0.0000 +
    [ad]     32.8715 24      0.1068
    [bc]     43.7552 36      0.1754
    [bd]     33.7206 24      0.0898
    [cd]     64.3488 27      0.0001 +
Formula: Sat:Infl:Type:Cont//
-2logL: 13544.49 DF: 0
```

Since we may be worried about the validity of the asymptotics, we also use Monte-Carlo tests:

```
> stepwise(msat, arg = "om")

Coherent Backward Single-step Selection.
Fixed edges: none.
```

```
Critical value:   0.0500
Decomposable mode, Chi-squared tests.
Exact tests, Monte Carlo sampling.
DFs adjusted for sparsity.
Model: abcd
Deviance:   0.0000 DF:   0 P:  1.0000
    Edge        Test
Excluded   Statistic DF          P
    [ab]    135.6898 32      0.0000 +
    [ac]     99.0937 36      0.0061 +
    [ad]     32.8715 24      0.1323
    [bc]     43.7552 36      0.1900
    [bd]     33.7206 24      0.1152
    [cd]     64.3488 27      0.0012 +
Formula: Sat:Infl:Type:Cont//
-2logL: 13544.49 DF: 0
```

This makes no essential difference, so we choose to continue with standard asymptotic options.

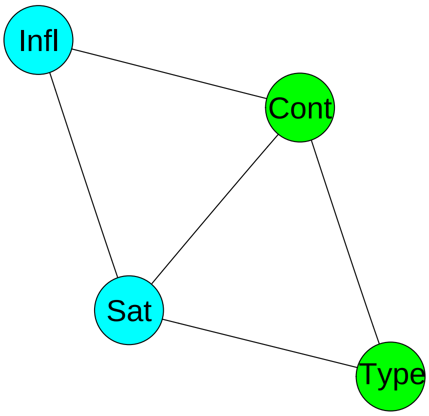
We can also go the entire way with a stepwise search and make a new model object from this search. The argument ”u” ensures that the search is unrestricted and all purely graphical models are investigated.

```
> mstep <- stepwise(msat, arg = "u")
```

```
Coherent Backward Selection.
Fixed edges: none.
Critical value:   0.0500
Unrestricted mode, Chi-squared tests.
Model: abcd
Deviance:   0.0000 DF:   0 P:  1.0000
    Edge        Test
Excluded   Statistic DF          P
    [ab]    135.6898 32      0.0000 +
    [ac]     99.0937 36      0.0000 +
    [ad]     32.8715 24      0.1068
    [bc]     43.7552 36      0.1754
    [bd]     33.7206 24      0.0898
    [cd]     64.3488 27      0.0001 +
Removed edge [bc]
Model: acd,abd
Deviance:  43.7552 DF:  36 P:  0.1754
    Edge        Test
Excluded   Statistic DF          P
    [ad]     29.6432 12      0.0032 +
    [bd]     24.4012  6      0.0004 +
Selected model: acd,abd
```

The dependence graph of the selected model can be displayed.

```
> display(mstep)
```



This analysis did not exploit that some of the variables were ordinal. If we repeat the same analysis, but use the appropriate ordinal test, we get a clearer picture from the outset. Note that the ordering of the variables now matter and the non-ordinal variable should be mentioned before the ordinal.

```
> testdelete("Sat:Infl", msat, arg = "j")
```

```
test: Jonckheere-Terpstra method: asymptotic
stat: 91192.5 df: 32 P: 0
```

```
> testdelete("Cont:Sat", msat, arg = "w")
```

```
test: Wilcoxon method: asymptotic
stat: 60178 df: 24 P: 0
```

```
> testdelete("Type:Sat", msat, arg = "k")
```

```
test: Kruskal-Wallis method: asymptotic
stat: 87.349 df: 18 P: 0
```

```
> testdelete("Cont:Infl", msat, arg = "w")
```

```
test: Wilcoxon method: asymptotic
stat: 70812 df: 24 P: 0
```

```
> testdelete("Type:Infl", msat, arg = "k")
```

```
test: Kruskal-Wallis method: asymptotic
stat: 23.077 df: 18 P: 0.1876
```

```
> testdelete("Cont:Type", msat)
```

```
test: Chi-squared method: asymptotic
stat: 64.349 df: 27 P: 0
```

With a single command, this can all be done as

```
> stepwise(msat, arg = "ow")
```

```
Coherent Backward Single-step Selection.
Fixed edges: none.
```

```
Critical value:   0.0500
Decomposable mode, Chi-squared tests.
DFs adjusted for sparsity.
Model: abcd
Deviance:   0.0000 DF:   0 P:  1.0000
    Edge        Test
Excluded   Statistic DF          P
    [ab]  91192.5000 32      0.0000 +
    [ca]     87.3486 18      0.0000 +
    [da]  60178.0000 24      0.0000 +
    [cb]     23.0770 18      0.1876
    [db]  70812.0000 24      0.0000 +
    [cd]     64.3488 27      0.0001 +
Formula: Sat:Infl:Type:Cont//
-2logL: 13544.49 DF: 0
```

MIM gives a slightly weird output here. It writes ”Chi-squared tests”, but in fact, it has used the appropriate ordinal test for each edge-deletion.

Taking ordinality into account leads to the same final model as before, but now the only non-significant edge, even at the initial stage, is that between Type and Influence.

We may still try to simplify the model found, now considering simplifications which remove second-order interactions. There is no easy way to take ordinality into account for this.

```
> m2factor1 <- editmim(mstep, deleteTerm = "Sat:Cont:Type")
> summary(m2factor1)
```

```
Formula: Type:Cont + Sat:Type + Sat:Infl:Cont//
Variables in model  :  Type Cont Sat Infl
deviance: 55.845 DF: 42 likelihood: 13600.34
```

```
> m2factor2 <- editmim(mstep, deleteTerm = "Sat:Cont:Infl")
> summary(m2factor2)
```

```
Formula: Sat:Type:Cont + Infl:Cont + Sat:Infl//
Variables in model  :  Sat Type Cont Infl
deviance: 45.551 DF: 40 likelihood: 13590.05
```

And they can then be compared to the model selected by the stepwise search

```
> modelTest(mstep, m2factor1)
```

```
Test of H0 :  Sat:Type:Cont + Sat:Infl:Cont//
Against    :  Type:Cont + Sat:Type + Sat:Infl:Cont//

test: Chi-squared method: asymptotic
stat: 12.09 df: 6 P: 0.06
```

```
> modelTest(mstep, m2factor2)
```

```
Test of H0 :  Sat:Type:Cont + Sat:Infl:Cont//
Against    :  Sat:Type:Cont + Infl:Cont + Sat:Infl//
```

```
test: Chi-squared method: asymptotic
stat: 1.795 df: 4 P: 0.773
```

They both give an acceptable fit, although the model `m2factor1` is close to being significant. Note that the mimR output has swapped H_0 and the alternative hypothesis in the text. This will be corrected in the next version of mimR.

If we remove both second-order terms, we get

```
> m2factor <- editmim(m2factor1, deleteTerm = "Sat:Cont:Infl")
> summary(m2factor)
```

```
Formula: Type:Cont + Infl:Cont + Sat:Cont + Sat:Infl + Sat:Type//
Variables in model  :  Type Cont Infl Sat
deviance: 57.64 DF: 46 likelihood: 13602.14
```

This model can be compared to the model found by selecting among graphical models.

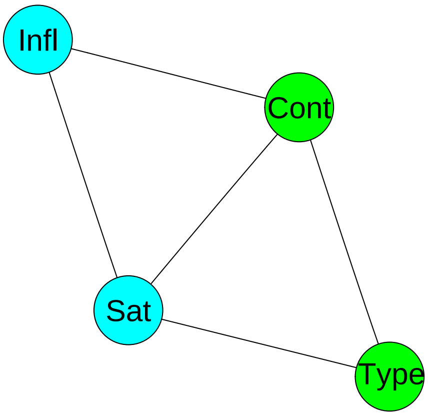
```
> modelTest(mstep, m2factor)
```

```
Test of H0 :  Sat:Type:Cont + Sat:Infl:Cont//
Against    :  Type:Cont + Infl:Cont + Sat:Cont + Sat:Infl + Sat:Type//

test: Chi-squared method: asymptotic
stat: 13.885 df: 10 P: 0.178
```

giving quite a reasonable fit.

We can display the final model

```
> display(m2factor)
```



but since mimR does not have the facility of displaying the interaction graph, we need to do so in MIM if we so wish.

The final model has been decomposed into one with just two cliques. A phenomenon, known as *collapsibility*, now ensures that we can proceed with analyzing the data in each of the marginal tables without paradoxes such as the Yule-Simpson. In fact we could have looked for removing interactions of higher order in each of these. For example, if we first specify the saturated marginal model as:

```
> marg1 <- mim("..", data = ht, marginal = c("Sat", "Infl", "Cont"))
> summary(marg1)
```

```
Formula: Sat:Infl:Cont//
Variables in model  :  Sat Infl Cont
deviance: 0 DF: 0 likelihood: 9419.526
```

and the model without second-order interactions

```
> marg12factor <- mim("Sat:Infl+Sat:Cont+Infl:Cont", data = ht,
+     marginal = c("Sat", "Infl", "Cont"))
> summary(marg12factor)
```

```
Formula: Sat:Infl+Sat:Cont+Infl:Cont
Variables in model  :  Sat Infl Cont
deviance: 1.795 DF: 4 likelihood: 9421.322
```

we can compare

```
> modelTest(marg1, marg12factor)
```

```
Test of H0 :  Sat:Infl:Cont//
Against    :  Sat:Infl+Sat:Cont+Infl:Cont

test: Chi-squared method: asymptotic
stat: 1.795 df: 4 P: 0.773
```

and get the same value.

Similarly with the other marginal and the corresponding model without second-order interactions

```
> marg2 <- mim("..", data = ht, marginal = c("Sat", "Type", "Cont"))
> marg22factor <- editmim(marg2, deleteTerm = "Sat:Cont:Type")
> modelTest(marg2, marg22factor)
```

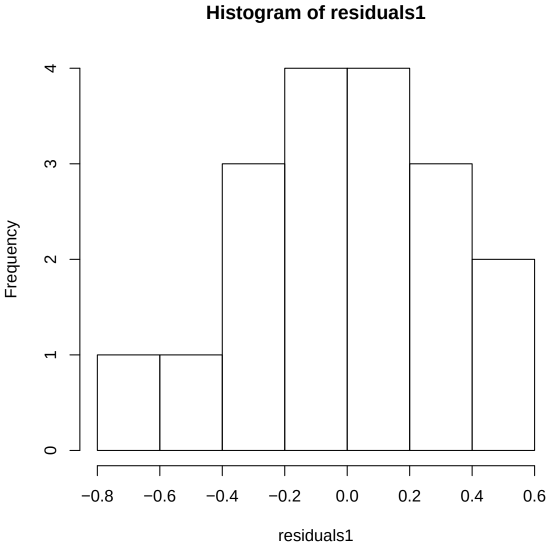
```
Test of H0 :  Sat:Type:Cont//
Against    :  Type:Cont + Sat:Cont + Sat:Type//

test: Chi-squared method: asymptotic
stat: 12.09 df: 6 P: 0.06
```

The interpretation of the final model is as follows:

- For given level of satisfaction and contact there is no (obvious) relationship between the type of housing and the feeling of having influence. This statement reflects the conditional independence between Influence and Type.
- The association between Satisfaction and Influence is the same for both levels of Contact.
- The association between Satisfaction and Type of housing is the same for both levels of Contact.

To investigate further whether the latter two relations hold water, we may further examine the residuals in the two marginal models:

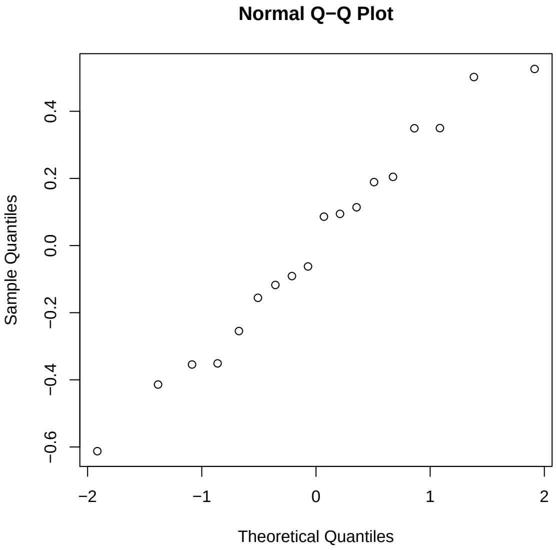
```
> observed1 <- fitted(marg1)[, 4]
> expected1 <- fitted(marg12factor)[, 4]
> residuals1 <- (observed1 - expected1)/sqrt(expected1)
> hist(residuals1)
```



These look amost suspiciously small, certainly nothing indicates the presence of outliers.

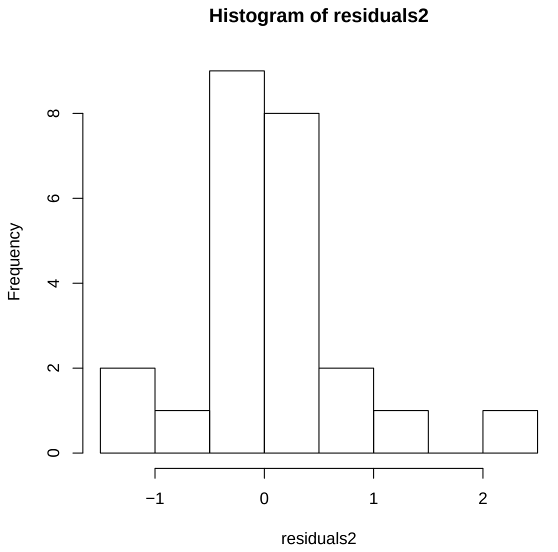
Also, they are not far from normally distributed:

```
> qqnorm(residuals1)
```



Considering the other marginal we get

```
> observed2 <- fitted(marg2)[, 4]
> expected2 <- fitted(marg22factor)[, 4]
> residuals2 <- (observed2 - expected2)/sqrt(expected2)
> hist(residuals2)
```

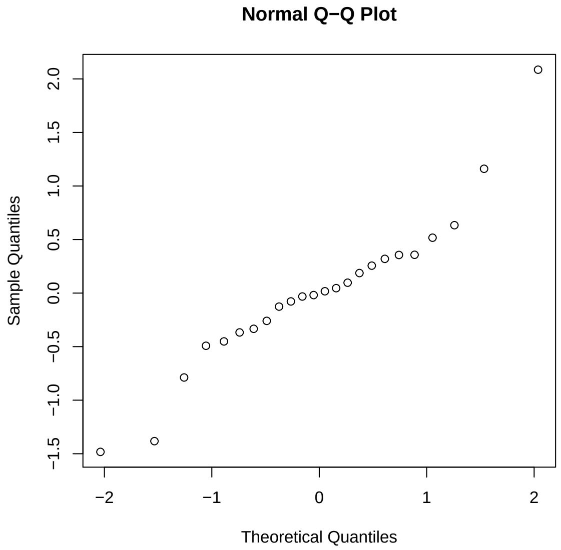


Some of these could look suspiciously large

```
> residuals2[residuals2 > 2]
[1] 2.086083
```

But there is only a single outlier and the value does not appear dramatic.

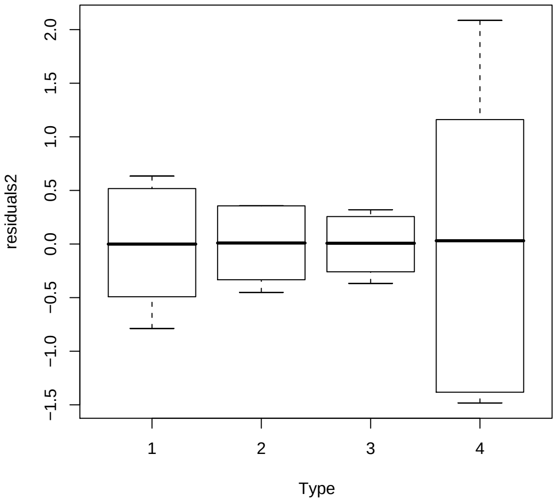
```
> qqnorm(residuals2)
```



On the other hand, the distribution seems to be too heavy tailed, so the model does not fit terribly welll.

We can for example study the dependence of the residuals on the levels of the categorial variables by producing new dataframes:

```
> residuals2 <- cbind(fitted(marg2)[, 1:3], residuals2)
> plot(residuals2 ~ Type, data = residuals2)
```



which indicates that there may be something different going on in terraced houses. . .

The advantage of using mimR over MIM is that such analyses as above (and many others) are now very easy to do.

References

Madsen, M. (1976). Statistical analysis of multiple contingency tables. two examples. *Scandinavian Journal of Statistics*, 3:97–106.