# Graphical Models - Assignment 2

Silvana Luque, Simon Schmetz

2025-05-05

```r
library(readr)
library(dplyr)
library(knitr)
library(graph)
library(gRim)
library(gRain)
library(bnlearn)
library(Rgraphviz)
```

## Taks

a) Provide a brief introduction to the data set
b) Try to define a possible network structure and give a brief explanation as to why this structure could make sense. Include a graph of the proposed model.
c) Fit the model and use it to make an out of sample prediction (like what we did in class to predict the probability a patient, who smokes and has not been to China, has tuberculosis ).
d) Try to fit the graph structure of the model using one or more of the different approaches we saw in class. Do the fits make sense? If not you could use whitelists or blacklists to make certain links impossible.
e) Which of the proposed graphical structures is the best?
f) Provide a brief summary of your results.

## a) Introduction

The following work on Bayesian Network was done as part of the Master in Statistics for Data Science at the Universidad Carlos III de Madrid and contains the design of a Graphical Bayesian Network model. The analysis was performed using the dataset titled "Weightlifting Injuries in Master Athletes" (https://zenodo.org/records/6679575), which comprises survey responses from 976 master weightlifters (aged 35–88, 51.1% female) across Australia, Canada, Europe, and the USA. Collected in June 2021, the data includes information on acute weightlifting-related injuries, chronic diseases, training practices, and sport history. Out a range of available variables, the following variables were extracted in the preprocessing of the data and then used for anaylsis:

- sex: sex of participant (m/f)
- agegrp3: age group of participant (35-49, 50-59, 60+)
- yrs_experience: years of experience in weightlifting (much = >2 years, little = <2 years)
- injury: sustained injuries in past (1/0) in any of the categories "shoulder", "knees", "back", "hips"
- train_warm: warm up before training in minutes (1/2+)
- coached: coached by a professional (1/0), either in person or online

the work begins with loading the data and some initial preprocessing. Then, graphical models are designed and fitted, first manually based on made assumptions and then based on design algorithms. The results are compared and a summary of the findings is provided.

```r
### Data Preprocessing
## Load Data
data <- read_csv("data/wlinj_dryad.csv")
```

```
## Rows: 976 Columns: 44
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr  (3): sex, agegrp3, hips
## dbl (41): id, age, age_start, yrs_experience, shoulder, knees, back, wrist, ...
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```r
## transform data

# aggregate yrs_experience
data <- data %>%
  mutate(across(all_of(c("yrs_experience")), ~ case_when(
    . <= 2 ~ "little",
    TRUE ~ "much"
  )))

# aggregate train_warm
data <- data %>%
  mutate(across(all_of(c("train_warm")), ~ case_when(
    . <= 1 ~ "1",
    TRUE ~ "2+"
  )))

# aggregate coached
data <- data %>%
  mutate(coached = if_else(pcoach == 1 | premote == 1, 1, 0))

# aggregate injury
data <- data %>%
  mutate(injury = if_else(shoulder == 1 | knees == 1 | back == 1 | hips == 1, 1, 0))

# select vars
data = data[c("sex", "agegrp3", "yrs_experience", "injury","train_warm","coached")]

# turn vars into factors
data <- data %>%
  mutate(across(everything(), as.factor))

# make sure data is a dataframe
data <- as.data.frame(data) # important dont know why but R is dogshiiiiiiiiit so I spend 20 min to fin

#print data
head(data)
```

```
##   sex agegrp3 yrs_experience injury train_warm coached
## 1   m   35-44         little      1         2+       1
```

```
## 2   m   45-59           much      1           1        1
## 3   m   35-44           much      1          2+        1
## 4   m   35-44           much      1          2+        1
## 5   f   45-59           much      0           1        1
## 6   m     60+           much      1           1        0
```

# b&c) Manual Model Design

The following section contains the manual design of a graphical model, which is then fitted to the data and used to make predictions.

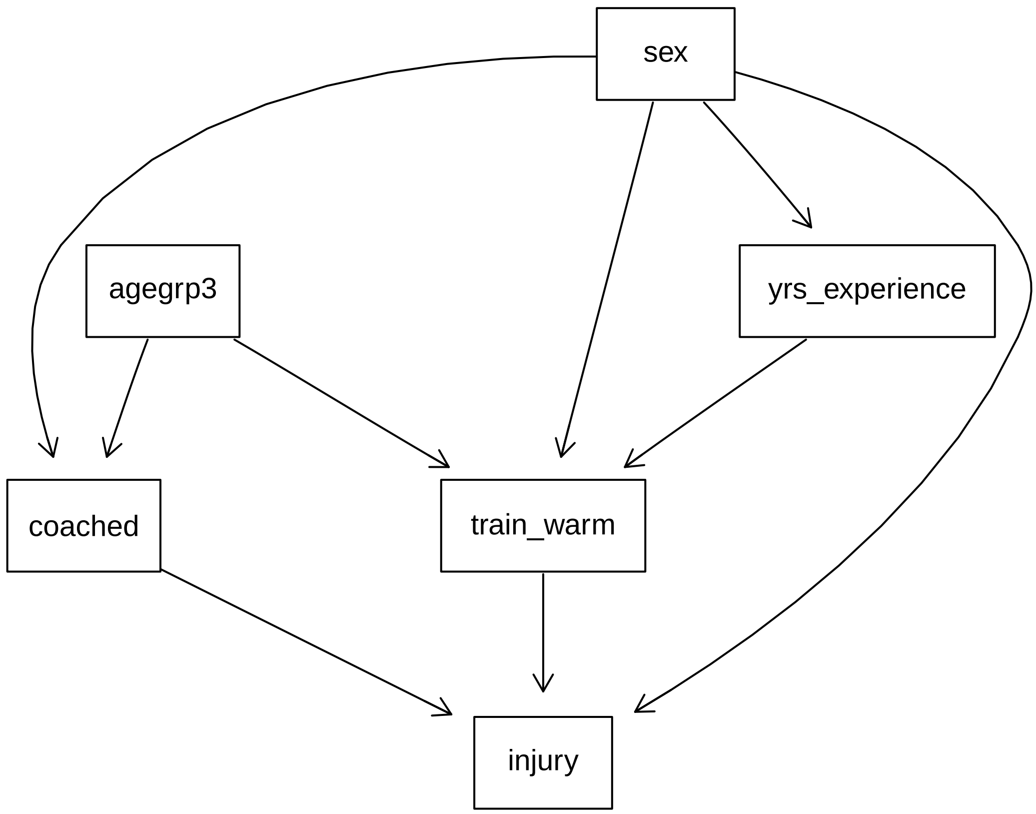
## b) Design Graph (Manually)

We begin by designing a graphical model based on what variables we asssume to be dependent. We begin by assuming that:

- **yrs_experience** is dependent on sex, as weight lifting appears to have started as a male dominated sport
- **train_warm** is dependent on yrs_experience, sex and agegrp3, as we assume that wiser are more likely to warm up before training and we expect that age, experience and not being male makes people wiser
- **coached** is dependent on agegrp3 and sex as we assume that older people prefer to do sport in groups and males to prefer to do sport alone
- **injury** is dependent on coached and train_warm, as we assume that people who are coached and/or properly warmed up are less likely to get injured, as well as dependent on sex as we expect men to tend to train more excessively than woman

These assumptions yield the following model:

```
dag <- model2network("[sex][agegrp3][yrs_experience|sex][train_warm|sex:yrs_experience:agegrp3][coached
graphviz.plot(dag)
```




## c) Fit Model and predict out of Sample

Next, we fit the model using the data and print out the probablity tables of the resulting model.

```
# Using maximum likelihood.
model_fit = bn.fit(dag,data,method = "bayes")
model_fit
```

```
##
##   Bayesian network parameters
##
##   Parameters of node agegrp3 (multinomial distribution)
##
## Conditional probability table:
##      35-44     45-59       60+
## 0.3892869 0.4046401 0.2060730
##
##   Parameters of node coached (multinomial distribution)
##
## Conditional probability table:
##
## , , sex = f
##
##        agegrp3
## coached     35-44     45-59       60+
##       0 0.1208178 0.1108887 0.1501241
##       1 0.8791822 0.8891113 0.8498759
##
## , , sex = m
```

```
##
##        agegrp3
## coached     35-44     45-59       60+
##       0 0.3078975 0.3851291 0.6192547
##       1 0.6921025 0.6148709 0.3807453
##
##
##   Parameters of node injury (multinomial distribution)
##
## Conditional probability table:
##
## , , sex = f, train_warm = 1
##
##       coached
## injury         0         1
##      0 0.5383387 0.5522145
##      1 0.4616613 0.4477855
##
## , , sex = m, train_warm = 1
##
##       coached
## injury         0         1
##      0 0.3479638 0.3847162
##      1 0.6520362 0.6152838
##
## , , sex = f, train_warm = 2+
##
##       coached
## injury         0         1
##      0 0.4763314 0.4970782
##      1 0.5236686 0.5029218
##
## , , sex = m, train_warm = 2+
##
##       coached
## injury         0         1
##      0 0.2619962 0.3741659
##      1 0.7380038 0.6258341
##
##
##   Parameters of node sex (multinomial distribution)
##
## Conditional probability table:
##         f        m
## 0.511259 0.488741
##
##   Parameters of node train_warm (multinomial distribution)
##
## Conditional probability table:
##
## , , sex = f, yrs_experience = little
##
##           agegrp3
## train_warm     35-44     45-59       60+
```

```
##          1  0.7137767 0.5996678 0.7448980
##          2+ 0.2862233 0.4003322 0.2551020
##
## , , sex = m, yrs_experience = little
##
##            agegrp3
## train_warm     35-44     45-59       60+
##          1  0.6146497 0.4172414 0.7117647
##          2+ 0.3853503 0.5827586 0.2882353
##
## , , sex = f, yrs_experience = much
##
##            agegrp3
## train_warm     35-44     45-59       60+
##          1  0.6031291 0.6119709 0.6030383
##          2+ 0.3968709 0.3880291 0.3969617
##
## , , sex = m, yrs_experience = much
##
##            agegrp3
## train_warm     35-44     45-59       60+
##          1  0.5803727 0.5656830 0.6377049
##          2+ 0.4196273 0.4343170 0.3622951
##
##
##   Parameters of node yrs_experience (multinomial distribution)
##
## Conditional probability table:
##
##               sex
## yrs_experience          f          m
##         little 0.12862863 0.06753927
##         much   0.87137137 0.93246073
```

We then find a combination of variables that doesn't exist in our data

```r
# find out of sample combination
all_combinations <- expand.grid(lapply(data, levels))
missing_combinations <- anti_join(all_combinations, data, by = names(data))
missing_combinations[1,]
```

```
##   sex agegrp3 yrs_experience injury train_warm coached
## 1   f     60+         little      0          1       0
```

and use that combination to perform a out of sample prediction for injury, yielding the following probability:

```r
# predict
cpquery(model_fit, (injury=="0"), (sex=="f" & agegrp3=="60+" & yrs_experience=="little" & train_warm  ==
```

```
## [1] 0.5520968
```

**d) Automatic Graph Design**

**e) Compare Graphs**

**f) Summary and Conclusions**