

# A Survey on the Means of Transportations

Marco Scutari

2022–2023


Consider a simple survey whose aim is to investigate the usage patterns of different means of transport, with a focus on cars and trains. For simplicity, let's say that you record the following (discrete) variables and values:

- *Age (A)*: *young* for individuals below 30 years old, *adult* for individuals between 30 and 60 years old, and *old* for people older than 60.
- *Sex (S)*: *male* or *female*.
- *Education (E)*: *up to high school* or *university degree*.
- *Occupation (O)*: *employee* or *self-employed*.
- *Residence (R)*: the size of the city the individual lives in, recorded as either *small* or *big*.
- *Travel (T)*: the means of transport favoured by the individual, recorded either as *car*, *train* or *other*.

## Part 1: Construct the Network Structure

The nature of the variables recorded in the survey suggests how they may be related with each other. Consider the following pieces of information from the press:

Exhibit #1:



Education ► Schools Teachers Universities Students

Higher education

### University gender gap at record high as 30,000 more women accepted

Ucas says young women a third more likely to go to university than men, and overall admissions are down on last year

Press Association  
Sun 27 Aug 2017  
19.01 EDT



▲ Students check their A-level results. Photograph: Andrew Matthews/PA

## Record numbers of teenagers going to university in England and Scotland, study finds

One in three youngsters in England and one in four in Scotland have been awarded university places this year

**Alison Kershaw** | Sunday 17 September 2017 19:19 |



Hike comes as overall numbers fall, due to fewer mature and EU students ( Getty )

## The Telegraph

> Lifestyle > Education & Careers

### Britain's highest paying degrees, according to UK graduate salaries



Exhibit #4:



Exhibit #5:



## Exhibit #6:

### Car, bike, train, or walk: how people get to work mapped

The 2011 census reveals the main way people commute to work in 34,753 'output areas' across England and Wales, each of 1,500 people. Find out what happens where you live - which are the top areas for cycling, driving and walking? And why the local concentrations of each?

Travel by local authority										
Click heading to sort table. <a href="#">Download this data</a>										
Place	Total people	Work at home, %	Tube, metro, light rail, tram, %	Train, Bus, coach, Taxi, %	%	%	Motorcycle etc %	Driving %	Passenger in car or van, %	Cycle, On foot, %
ENGLAND	38,881,374	3.5	2.6	3.5	4.9	0.3	0.5	36.9	3.3	1.9
NORTH EAST	1,924,206	2.2	1.5	0.7	5.6	0.5	0.3	36.9	4.2	1.1
NORTH WEST	5,184,216	2.8	0.4	1.7	5.2	0.5	0.4	39	3.8	1.4
YORKSHIRE AND THE HUMBER	3,875,219	2.9	0.3	1.5	5.3	0.4	0.4	38.4	4	1.6
EAST MIDLANDS	3,336,532	3.3	0.2	0.9	4	0.3	0.5	42.2	3.9	1.8
WEST MIDLANDS	4,067,119	3	0.2	1.6	4.8	0.3	0.4	40.6	3.8	1.2
EAST	4,245,544	3.8	0.8	4.8	2.5	0.3	0.5	41.4	3.4	2.4
LONDON	6,117,482	3.3	14.7	8.7	9.2	0.3	0.8	18.3	1.1	2.6
SOUTH EAST	6,274,341	4.5	0.2	5	3	0.3	0.6	41.3	3.2	2
SOUTH WEST	3,856,715	4.6	0.1	1	3.1	0.2	0.7	41.4	3.4	2.3

1. To start, create an empty DAG with a node for each of the variables list above.
2. Consider the information available from the British press. What arcs do you think should be included in the network? Add them individually with `set.arc()` and think of how their inclusion can be justified.
3. Create the same DAG as in the previous point in one go using model formulas (`model2network()`), adjacency matrices (`amat()`) and arc sets (`arcs()`).
4. Identify the parents, the children and the spouses of each node. Verify empirically that they form the Markov blanket of that node. What do the Markov blankets tell us?
5. Construct the CPDAG. Which arcs are directed, which are undirected? Why? What does that tell us?

## Commented Code for Part 1: Construct the Network Structure

1. To make code more compact, you can name the nodes using the initials of the variable names.

```
survey.dag = empty.graph(nodes = c("A", "S", "E", "O", "R", "T"))
```

2. Age and Sex are not influenced by any of the other variables, hence we do not need any arcs pointing to either either of them. Age has a direct influence on Education. The number of people attending universities has increased over the years: so younger people are more likely to have a university degree than older people.

```
survey.dag = set.arc(survey.dag, from = "A", to = "E")
```

Sex also influences Education: the gender gap in university applications has been widening for many years, with women outnumbering and outperforming men.

```
survey.dag = set.arc(survey.dag, from = "S", to = "E")
```

Education strongly influences Occupation because higher education levels help in accessing more prestigious professions.

```
survey.dag = set.arc(survey.dag, from = "E", to = "O")
```

Education influences Residence as well because people often move to attend a particular university or to find a job that matches the skills they acquired in their studies.

```
survey.dag = set.arc(survey.dag, from = "E", to = "R")
```

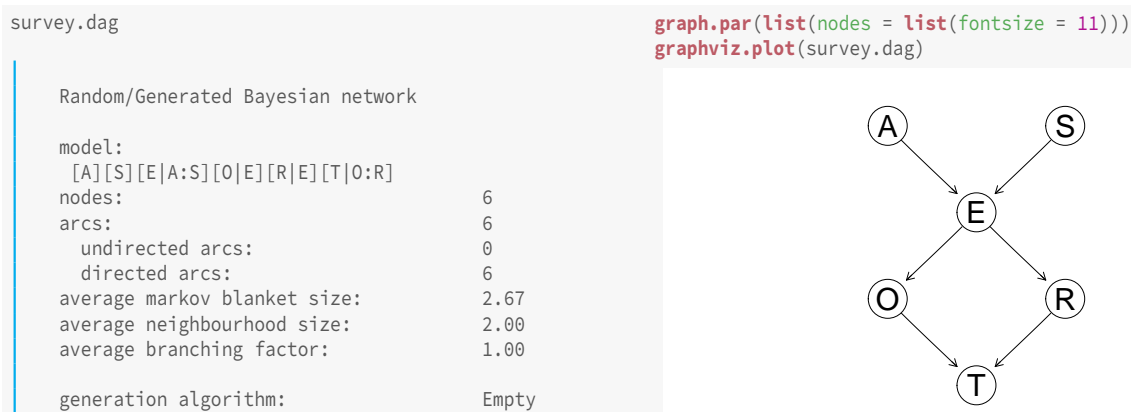
Finally, the preferred means of transport are directly influenced by both Occupation and Residence. For Occupation, the reason is that a few jobs require periodic long-distance trips, while others require more frequent trips but on shorter distances.

```
survey.dag = set.arc(survey.dag, from = "O", to = "T")
```

For Residence, the reason is that both commute time and distance are deciding factors in choosing between travelling by car or by train.

```
survey.dag = set.arc(survey.dag, from = "R", to = "T")
```

In conclusion, the DAG:



### 3. The DAG, created with model2network():

```
survey.dag = model2network("[A][S][E|A:S][O|E][R|E][T|O:R]")
```

With amat():

```
adjacency.matrix = matrix(0, nrow = nnodes(survey.dag), ncol = nnodes(survey.dag),
  dimnames = list(nodes(survey.dag), nodes(survey.dag)))
adjacency.matrix["A", "E"] = 1
adjacency.matrix["S", "E"] = 1
adjacency.matrix["E", "O"] = 1
adjacency.matrix["E", "R"] = 1
adjacency.matrix["O", "T"] = 1
adjacency.matrix["R", "T"] = 1
amat(survey.dag) = adjacency.matrix
```

With arcs():

```
arc.set = matrix(c("A", "E",
  "S", "E",
  "E", "O",
  "E", "R",
  "O", "T",
  "R", "T"),
  byrow = TRUE, ncol = 2, dimnames = list(NULL, c("from", "to")))
arcs(survey.dag) = arc.set
```



4. The Markov blanket of each node is defined as the union of the parents, the children and the spouses (that is, the children's other parents). We can verify that it holds empirically:

```
for (node in nodes(survey.dag)) {  
  
  pa = parents(survey.dag, node)  
  ch = children(survey.dag, node)  
  sp = spouses(survey.dag, node)  
  mb = mb(survey.dag, node)  
  
  cat("node", node, "\n")  
  cat("* parents:", pa, "\n")  
  cat("* children:", ch, "\n")  
  cat("* spouses:", sp, "\n")  
  cat("* Markov blanket:", mb, "\n")  
  cat("@ same?", setequal(mb, union(pa, union(ch, sp))), "\n")  
  
}#FOR  
  
node A  
* parents:  
* children: E  
* spouses: S  
* Markov blanket: E S  
@ same? TRUE  
node E  
* parents: A S  
* children: O R  
* spouses:  
* Markov blanket: A O R S  
@ same? TRUE  
node O  
* parents: E  
* children: T  
* spouses: R  
* Markov blanket: E R T  
@ same? TRUE  
node R  
* parents: E  
* children: T  
* spouses: O  
* Markov blanket: E O T  
@ same? TRUE  
node S  
* parents:  
* children: E  
* spouses: A  
* Markov blanket: A E  
@ same? TRUE  
node T  
* parents: O R  
* children:  
* spouses:  
* Markov blanket: O R  
@ same? TRUE
```

What Markov blankets tell us:

- Travel (T) is completely explained by Occupation (O) and Residence (R). This is important for this model because it suggests that you can predict preferences for different means of transportation from just these two variables.
- Occupation is completely explained by Travel, Residence and Education (E).
- Symmetrically, Residence is completely explained by Travel, Occupation and Education.
- Education is completely explained by Age (A), Sex (S), Occupation and Residence.
- Age is completely explained by Education and Sex.
- Symmetrically, Sex is completely explained by Age and Education.

Note that "explained" is different from "caused"!

## 5. The CPDAG is identical to the DAG for this BN:

```
cpdag(survey.dag)

Random/Generated Bayesian network

model:
  [A][S][E|A:S][O|E][R|E][T|O:R]
nodes:                                6
arcs:                                 6
  undirected arcs:                     0
  directed arcs:                       6
average markov blanket size:          2.67
average neighbourhood size:           2.00
average branching factor:              1.00

generation algorithm:                  Empty
```

All arcs are directed because:

- $A \rightarrow E \leftarrow S$  is a v-structure.
- $O \rightarrow T \leftarrow R$  is a v-structure.
- $E \rightarrow O$  is a compelled arc because  $O \rightarrow E$  would introduce the additional v-structure  $A \rightarrow E \leftarrow O$ .
- Similarly,  $E \rightarrow R$  is a compelled arc because  $R \rightarrow E$  would introduce the additional v-structure  $A \rightarrow E \leftarrow R$ .

The fact that all arc directions are uniquely identified suggests that you could learn them well from data because the BN is part of a equivalence class that contains a single model.

## Part 2: Set the Parameters

A BN is the combination of a DAG and a probability distribution, structured into a set of local distributions of the form  $P(X_i | \Pi_{X_i})$ . In **Part 1** you have created the former, and now you can create the latter from the available information. Recall some key facts from the news articles we used to as a basis to construct the DAG:

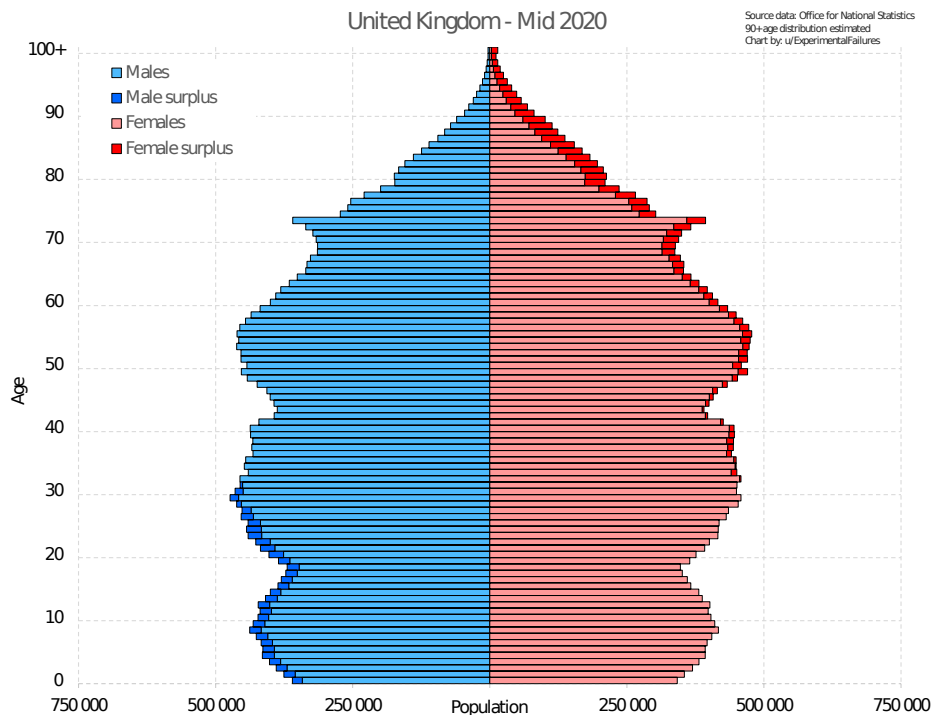
- More women enroll in (and thus graduate from) university than men.
- More people enroll in university compared to past years and to previous generations.
- University graduates get higher salaries than non-graduates, some more than others.
- A large share of jobs requiring a university degree are in the largest city in the country.
- The use of different means of transportation is very different in a big city (where you have public transport and you can bike) compared to small cities or rural areas (car driving is the only option).

You can use these facts to create the probability tables that contain the parameters of the local distributions, which look like this:

A young adult old "?" "?" "?"	R E high uni small "?" "?" big "?" "?"	, , R = small O T emp self car "?" "?" train "?" "?" other "?" "?"
S M F "?" "?"	, , S = M A E young adult old high "?" "?" "?" uni "?" "?" "?"	, , R = big O T emp self car "?" "?" train "?" "?" other "?" "?"
O E emp high uni self "?" "?"	, , S = F A E young adult old high "?" "?" "?" uni "?" "?" "?"	

For the Age and Sex, consider the population pyramid plot from the Office for National Statistics.

*Exhibit #7:*



Discuss the following points:

1. How does the choice of the arcs impact the number of parameters that we have to specify?
2. How are the conditional probabilities for the same variable but for different values of their parents related? Should they be similar? Should they be very different?

### Commented Code for Part 2: Set the Parameters

These are the conditional probabilities I chose when I constructed this example in 2013:

```
A.lv = c("young", "adult", "old")
S.lv = c("M", "F")
E.lv = c("high", "uni")
O.lv = c("emp", "self")
R.lv = c("small", "big")
T.lv = c("car", "train", "other")
A.prob = array(c(0.30, 0.50, 0.20), dim = 3, dimnames = list(A = A.lv))
A.prob

| A
| young adult old
| 0.3 0.5 0.2

S.prob = array(c(0.60, 0.40), dim = 2, dimnames = list(S = S.lv))
S.prob

| S
| M F
| 0.6 0.4

O.prob = array(c(0.96, 0.04, 0.92, 0.08), dim = c(2, 2), dimnames = list(O = O.lv, E = E.lv))
O.prob

| E
| O high uni
| emp 0.96 0.92
| self 0.04 0.08
```



```

R.prob = array(c(0.25, 0.75, 0.20, 0.80), dim = c(2, 2), dimnames = list(R = R.lv, E = E.lv))
R.prob

      E
R      high uni
small 0.25 0.2
big   0.75 0.8

E.prob = array(c(0.75, 0.25, 0.72, 0.28, 0.88, 0.12, 0.64, 0.36, 0.70, 0.30, 0.90, 0.10),
               dim = c(2, 3, 2), dimnames = list(E = E.lv, A = A.lv, S = S.lv))
E.prob

, , S = M

      A
E      young adult old
high  0.75 0.72 0.88
uni   0.25 0.28 0.12

, , S = F

      A
E      young adult old
high  0.64 0.7 0.9
uni   0.36 0.3 0.1

T.prob = array(c(0.48, 0.42, 0.10, 0.56, 0.36, 0.08, 0.58, 0.24, 0.18, 0.70, 0.21, 0.09),
               dim = c(3, 2, 2), dimnames = list(T = T.lv, O = O.lv, R = R.lv))
T.prob

, , R = small

      O
T      emp self
car   0.48 0.56
train 0.42 0.36
other 0.10 0.08

, , R = big

      O
T      emp self
car   0.58 0.70
train 0.24 0.21
other 0.18 0.09

cpt = list(A = A.prob, S = S.prob, E = E.prob, O = O.prob, R = R.prob, T = T.prob)
survey.bn = custom.fit(survey.dag, cpt)

```

You will notice that nodes that have more parents, like Education or Travel, have more parameters—conditional probability tables are larger. This is natural: the more parents a node has, the more dimensions the local distributions has, and the more parameters. It becomes more and more difficult to specify conditional probabilities from expert knowledge as they grow in number. Even estimating them becomes difficult, as we will see later in the course: collecting enough data to estimate them with sufficient precision will be a challenge.

In most cases, individual probability distributions in the same conditional probability table should be more similar than they are different: phenomena tend to have smooth behaviour, so small changes in the parents should correspond to small changes in the probability. In particular, it is uncommon to have extreme probabilities (with values very close to 0 or to 1) for nodes that do not represent human decisions. They will cause numerical problems because they prevent us from reversing the direction of conditioning via Bayes' theorem. However, they should be different enough that the arc represents a meaningful probabilistic dependence between the nodes and the parents.

### Part 3: Validate and Query the Network

BNs as machine learning models are meant to provide a working model of reality good enough for a computer system to understand and use. We can ask the computer systems questions and check whether the answers make sense to validate the BN. We will see how that works in detail later: for now, let's use the `setEvidence()` and `querygrain()` from the **gRain** package after converting the BN we created in **Part 2**

with `as.grain()`.

1. Do women have different preferences than men towards different means of transportation?
2. Does the answer change if we limit ourselves to men and women with university degrees?
3. Do young people use the train more rather than driving compared to old people?

### Commented Code for Part 3: Validate and Query the Network

1. The marginal distribution of Travel in the general population can be obtained as follows.

```
options(digits = 4)
junction = as.grain(survey.bn)
querygrain(junction, nodes = "T")$T
|  T
|  car train other
|  0.5618 0.2809 0.1573
```

After we tell the BN we are talking about women (first) or men (second), we can get the marginal probability of Travel for each Sex.

```
jF = setEvidence(junction, nodes = "S", states = "F")
querygrain(jF, nodes = "T")$T
|  T
|  car train other
|  0.5621 0.2806 0.1573

jM = setEvidence(junction, nodes = "S", states = "M")
querygrain(jM, nodes = "T")$T
|  T
|  car train other
|  0.5617 0.2810 0.1573
```

The difference between the two distributions is minimal, which suggests that preferences do not change between sexes.

2. The difference in preferences between men and women disappears completely if we focus on university graduates.

```
options(digits = 4)
jE = setEvidence(junction, nodes = "E", states = "uni")
jF = setEvidence(jE, nodes = "S", states = "F")
querygrain(jF, nodes = "T")$T
|  T
|  car train other
|  0.5690 0.2731 0.1579

jM = setEvidence(jE, nodes = "S", states = "M")
querygrain(jM, nodes = "T")$T
|  T
|  car train other
|  0.5690 0.2731 0.1579
```

The reason for that is simple: Sex and Travel are d-separated by Education, so by construction

$$P(T \mid S, E) = P(T \mid E)$$

because graphical separation implies conditional independence.

```
dsep(survey.dag, x = "T", y = "S", z = "E")
| [1] TRUE
```

### 3. Again, the difference is minimal.

```
jT = setEvidence(junction, nodes = "T", states = "train")
querygrain(jT, nodes = "A")$A
| A
|   young  adult   old
| 0.2996 0.4994 0.2011

jT = setEvidence(junction, nodes = "T", states = "car")
querygrain(jT, nodes = "A")$A
| A
|   young  adult   old
| 0.3002 0.5003 0.1995
```