Computer-supported Decision Making
Including: Which movie should I watch tonight?

Helge Langseth

Joint work with Thomas D. Nielsen, Aalborg University

NTNU
1 Bayesian networks
   - The “Explosion”-example
   - Bayesian networks: A formal modelling framework

2 Latent variables for data compression and augmentation
   - Latent variables
   - Text “understanding” – The reuters dataset

3 Movie recommendations
   - Introduction
   - A latent variable model
   - Movie “understanding” – The MovieLens dataset

4 Conclusions
A gas leak scenario

1. A gas leak ($L$) can lead to an explosion ($X$);
2. An explosion ($X$) can lead to one or more casualties ($C$);
3. Gas leaks ($L$) detected by a gas detector ($G$) are not harmful;
4. The environment ($E$) influences the gas leak frequency ($L$) ... and the reliability of the gas detector ($G$).
A gas leak scenario

1. A gas leak ($L$) can lead to an explosion ($X$);
2. An explosion ($X$) can lead to one or more casualties ($C$);
3. Gas leaks ($L$) detected by a gas detector ($G$) are not harmful;
4. The environment ($E$) influences the gas leak frequency ($L$)
5. ... and the reliability of the gas detector ($G$).

Relevant questions we may pose:

Deductive reasoning: “To what extent will an improved gas detector reduce expected number of casualties/year?”

Abductive reasoning: “Having seen a series of explosions recently, can I say something about the environment?”

General case: “What is $P$ (Query variables|Observed variables)?”

Desired: A modelling framework where (causal) knowledge can be encoded and relevant queries answered.
A simple BN example: “Explosion”

- $E$: Environment
- $L$: Leak
- $G$: GD failed
- $X$: Explosion
- $C$: Casualties

$P(E, L, G, X, C)$
A simple BN example: “Explosion”

\[ P(E, L, G, X, C) \]

\[ \text{pa} (X) = \{L, G\} \]
A simple BN example: “Explosion”

$\begin{align*}
E &: \text{Environment} \\
L &: \text{Leak} \\
G &: \text{GD failed} \\
X &: \text{Explosion} \\
C &: \text{Casualties}
\end{align*}$

$\text{pa}(X) = \{L, G\}$
$\text{nd}(X) = \{E, L, G\}$

$P(E, L, G, X, C)$
A simple BN example: “Explosion”

\[ \text{E: Environment} \]
\[ \text{L: Leak} \]
\[ \text{G: GD failed} \]
\[ \text{X: Explosion} \]
\[ \text{C: Casualties} \]

\[ \text{pa}(X) = \{L, G\} \]
\[ \text{nd}(X) = \{E, L, G\} \]
\[ X \perp E | \{L, G\} \]

\[ P(E, L, G, X, C) \]
A simple BN example: “Explosion”

\[ \begin{align*}
E: \text{Environment} \\
L: \text{Leak} \\
G: \text{GD failed} \\
X: \text{Explosion} \\
C: \text{Casualties}
\end{align*} \]

\[ \begin{array}{c|cc}
G & E = \text{hostile} & E = \text{normal} \\
\hline
\text{yes} & p_H & p_N \\
\text{no} & 1 - p_H & 1 - p_N
\end{array} \]

\[ P(G | \text{pa}(G)) \]

\[ \text{pa}(X) = \{L, G\} \]

\[ \text{nd}(X) = \{E, L, G\} \]

\[ X \perp\!\!\!\!\!\!\!\perp E | \{L, G\} \quad (\text{Hence, } P(X | E, L, G) = P(X | L, G)) \]

\[ P(E, L, G, X, C) = P(E) \cdot P(L | E) \cdot P(G | E, L) \cdot P(X | E, L, G) \cdot P(C | E, L, G, X) \]

\[ = P(E) \cdot P(L | E) \cdot P(G | E) \cdot P(X | L, G) \cdot P(C | X) \]

Fast inference algorithms utilise these independence properties.
MUNIN: An expert system for electromyography

Left Medianus - Abductor Pollicis Brevis

Right Medianus - Abductor Pollicis Brevis

Left Ulnaris - Abductor Digiti Minimi

Right Ulnaris - Abductor Digiti Minimi

Left Axillaris - Deltoideus

Right Axillaris - Deltoideus

Left Suralis

Right Suralis

MYOTONIC DYSTROPHY
Consider a text document represented by the variables $X_i$: “Is word $i$ of the vocabulary used in the document?”

We have $n$ (no. terms in vocabulary) variables for each document.

Example:

- Vocabulary: \{text, image, Bayes, network, analysis\}
- Document: Bayesian text analysis rocks.
- Representation: $(X_1, \ldots, X_5) = (1, 0, 1, 0, 1)$
Consider a text document represented by the variables $X_i$: “Is word $i$ of the vocabulary used in the document?” We have $n$ (no. terms in vocabulary) variables for each document.

**Example:**

- Vocabulary: \{text, image, Bayes, network, analysis\}
- Document: Bayesian text analysis rocks.
  - word 3
  - word 1
  - word 5
  - N/A
- Representation: $(X_1, \ldots, X_5) = (1, 0, 1, 0, 1)$

**Question:**
Can we automatically examine a number of documents and find their “meaning”, thus get a representation better suited for analysis?

- Fewer, more cleverly defined features.
- New features capturing semantics (“meaning”) of text, not syntax (“writing style”).
One solution:

- Introduce $Z = (Z_1, \ldots, Z_q)$, a vector of latent variables, representing a “compressed” representation (assuming $q \ll n$).
- Each $Z_j$ defines a “topic” (aggregated meaning) of document; the presence of a topic influences the probability of seeing a specific word in the document.
- In this factor analysis model, the factor variables model the correlation among the attributes.
Example:

- The often-used *reuters-22173* dataset was analysed.
  - Vocabulary containing a subset of 500 words (chosen automatically).

- Some of the discovered topics:
  - negotiation, agreement, deal, hope, meeting, international, . . .
  - USDA, agriculture, crop, export, wheat, tonnes, . . .
  - estimate, expect, statistics, reserve, fall, rise, season, qtr, . . .
  - . . .

Key conclusion:

*Latent variables* appear useful to detect/automatically learn and represent generalisations of high-dimensional data!
Collaborative filtering: To predict the utility of items for the active user based on a database of rating data.

Input data: Our data is a sparse matrix of ratings.

<table>
<thead>
<tr>
<th>User</th>
<th>Star Wars</th>
<th>Platoon</th>
<th>Olsen</th>
<th>banden</th>
<th>1</th>
<th>Festen</th>
<th>5</th>
<th>Das boot</th>
<th>5</th>
<th>...</th>
</tr>
</thead>
<tbody>
<tr>
<td>User 1</td>
<td>5</td>
<td>1</td>
<td>4</td>
<td>5</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>User 2</td>
<td>1</td>
<td>5</td>
<td>2</td>
<td>5</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>User 3</td>
<td>5</td>
<td>5</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>User 4</td>
<td>2</td>
<td>1</td>
<td>3</td>
<td>2</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>User 5</td>
<td>3</td>
<td>4</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>⋮</td>
<td>⋮</td>
<td>⋮</td>
<td>⋮</td>
<td>⋮</td>
<td></td>
<td>⋮</td>
<td>⋮</td>
<td>⋮</td>
<td>⋮</td>
<td>⋮</td>
</tr>
</tbody>
</table>

Notation: $R_{p,i}$ is the rating User $p$ gives to Movie $i$, so $R_{2,1} = 1$.

Goal: Predict values for unobserved ratings given a database of ratings. E.g., how will User 5 rate Festen? (I.e., what is a good guess for $R_{5,5}$?)
What determines how a user will rate a movie?

- Let movie no. \( i \) be represented by \( M_i \), a point in \( \mathbb{R}^q \):
  - Each of the \( q \) dimensions of \( M_i \) has some (implicit) meaning, e.g., degree of chick-movie, size of production, etc.
  - Finding a good encoding is done automatically during learning, but we can inspect the representation afterwards.
  - Movie representations spread around zero; we model them as standard Gaussians a priori.
  - Each user \( p \) has a preference for the \( q \) aspects of a movie. This is modelled by a \( q \)-dimensional vector \( \nu_p \).
Proposed Model

What determines how a user will rate a movie?

- Let **movie no.** \( i \) be represented by \( M_i \), a point in \( \mathbb{R}^q \).
- Let **user no.** \( p \) be represented by \( U_p \), a point in \( \mathbb{R}^r \):
  - Each of the \( r \) dimensions of \( U_p \) has some (implicit) meaning.
  - User representations spread around zero; we model them as standard Gaussians a priori.
  - Each **movie** \( i \) has a connection to the \( r \) aspects of a user. This is modelled by a \( r \)-dimensional vector \( w_i \).
What determines how a user will rate a movie?

- Let movie no. $i$ be represented by $M_i$, a point in $\mathbb{R}^q$.
- Let user no. $p$ be represented by $U_p$, a point in $\mathbb{R}^r$.
- Let $\psi_p$ be average rating for user $p$ (modelling “grumpiness”), and $\phi_i$ be average rating for movie $i$ after adjusting for which users have rated it (modelling “quality”).
What determines how a user will rate a movie?

- Let movie no. \( i \) be represented by \( M_i \), a point in \( \mathbb{R}^q \).
- Let user no. \( p \) be represented by \( U_p \), a point in \( \mathbb{R}^r \).
- Let \( \psi_p \) be average rating for user \( p \) (modelling “grumpiness”), and \( \phi_i \) be average rating for movie \( i \) after adjusting for which users have rated it (modelling “quality”).

Rating model:

\[
R_{p,i} | \{ M_i = m_i, U_p = u_p \} = v_p^T m_i + w_i^T u_p + \phi_i + \psi_p + \epsilon
\]

where \( \epsilon \) is a noise term.
The model is a Bayesian Network
The 10 items closest to *Star Wars* and *Three Colors: Blue*:

<table>
<thead>
<tr>
<th>Close to <em>Star Wars</em></th>
<th>Close to <em>Three Colors: Blue</em></th>
</tr>
</thead>
<tbody>
<tr>
<td>1. The Empire Strikes Back</td>
<td>1. Welcome to the Dollhouse</td>
</tr>
<tr>
<td>3. Star Trek II</td>
<td>3. Three Colors: White</td>
</tr>
<tr>
<td>4. Return of the Jedi</td>
<td>4. Wings of Desire</td>
</tr>
<tr>
<td>5. Raiders of the Lost Ark</td>
<td>5. Everyone Says I Love You</td>
</tr>
<tr>
<td>7. <em>Private Parts</em></td>
<td>7. Dead Man Walking</td>
</tr>
<tr>
<td>8. Star Trek VI</td>
<td>8. The Nightmare Before Christmas</td>
</tr>
<tr>
<td>10. Men in Black</td>
<td>10. To Die For</td>
</tr>
</tbody>
</table>

**NOTE!**
Only the rating matrix is used to find these patterns!

**About the model building:**
- We chose $r = 1$ and $q = 3$ for this analysis
- MovieLens data – 100 000 ratings (943 users, 1682 movies)
- **Automatic (EM) learning** used to find all parameters
The value of $\psi_i$ may say something about the “movie quality”:

<table>
<thead>
<tr>
<th>Movies with highest $\psi_i$</th>
<th>Movies with highest average rating</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. The Shawshank Redemption ▲</td>
<td>1. Entertaining Angels: The Dorothy Day Story ▼</td>
</tr>
<tr>
<td>2. Schindler’s List ▲</td>
<td>2. Someone Elses America ▼</td>
</tr>
<tr>
<td>3. Star Wars ▲</td>
<td>3. Aiqing wansui ▼</td>
</tr>
<tr>
<td>4. Casablanca ▲</td>
<td>4. Santa with Muscles ▼</td>
</tr>
<tr>
<td>5. The Usual Suspects ▲</td>
<td>5. The Saint of Fort Washington ▼</td>
</tr>
<tr>
<td>6. Rear Window ▲</td>
<td>6. Star Kid ▼</td>
</tr>
<tr>
<td>8. The Silence of the Lambs ▲</td>
<td>8. Prefontaine ▼</td>
</tr>
<tr>
<td>9. One Flew Over the Cuckoo’s Nest ▲</td>
<td>9. They Made Me a Criminal ▼</td>
</tr>
<tr>
<td>10. 12 Angry Men ▲</td>
<td>10. A Great Day in Harlem ▼</td>
</tr>
</tbody>
</table>

▲: **In Top 25** of IMDBs list of 250 best movies
▼: **Not at all** in IMDBs list of 250 best movies
Conclusions

- Making decisions under uncertainty is a task we are faced with every day. Still, it is hard to automate this reasoning process in a computer. **Bayesian Networks** is one framework that shows promise in this regard.

- **Latent variable models** can effectively be used to synthesise/aggregate information from complex high-dimensional data.

- We have seen by example that the latent variables give a reasonable aggregation in the movie domain, as they offer a relevant **semantic interpretation**.

- Although we have only discussed qualitative properties of the model, it also gives **very good quantitative results** wrt. quality of recommendations: We obtain a *mean absolute error* of 0.69 on the MovieLens dataset; other systems typically in the area 0.73 – 0.74.