WWU MÜNSTER

NOT PART OF WINTER TERM 2022 / 23

Continuous Space

Statistical Relational Artificial Intelligence (StaRAI)

$$\begin{pmatrix}
\Sigma_{r_1r_1} & \cdots & \Sigma_{r_1r_n} & \Sigma_{r_1s_1} & \cdots & \Sigma_{r_1s_m} & \Sigma_{r_1t_1} & \cdots & \Sigma_{r_1t_l} \\
\vdots & \ddots & \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\
\Sigma_{r_nr_1} & \cdots & \Sigma_{r_nr_n} & \Sigma_{r_ns_1} & \cdots & \Sigma_{r_ns_m} & \Sigma_{r_nt_1} & \cdots & \Sigma_{r_nt_l} \\
\Sigma_{s_1r_1} & \cdots & \Sigma_{s_1r_n} & & & & \Sigma_{s_1t_1} & \cdots & \Sigma_{s_1t_l} \\
\vdots & \ddots & \vdots & & & & \vdots & \ddots & \vdots \\
\Sigma_{s_mr_1} & \cdots & \Sigma_{s_mr_n} & & & & \Sigma_{s_mt_1} & \cdots & \Sigma_{s_mt_n} \\
\Sigma_{t_1r_1} & \cdots & \Sigma_{t_1r_n} & \Sigma_{t_1s_1} & \cdots & \Sigma_{t_1s_m} & \Sigma_{t_1t_1} & \cdots & \Sigma_{t_1t_l} \\
\vdots & \ddots & \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\
\Sigma_{t_lr_1} & \cdots & \Sigma_{t_lr_n} & \Sigma_{t_ls_1} & \cdots & \Sigma_{t_ls_m} & \Sigma_{t_lt_1} & \cdots & \Sigma_{t_lt_l}
\end{pmatrix}$$

$$\begin{pmatrix} \sigma^2_{R(X)} \\ \\ \sigma^2_{T(Z)} \end{pmatrix} \quad \begin{pmatrix}
0 & \sigma^2_{R(X)}\beta_{rs} & m\sigma^2_{R(X)}\beta_{rs}\beta_{st} \\
\sigma^2_{R(X)}\beta_{rs} & & (\sigma^2_{S(Y)} + mn\sigma^2_{R(X)}\beta^2_{rs})\beta_{st} \\
m\sigma^2_{R(X)}\beta_{rs}\beta_{st} & (\sigma^2_{S(Y)} + mn\sigma^2_{R(X)}\beta^2_{rs})\beta_{st} & m\beta^2_{st}(\sigma^2_{S(Y)} + mn\sigma^2_{R(X)}\beta^2_{rs})
\end{pmatrix}$$

Tanya Braun

Contents

Models with Continuous Variables

- Discretisation of continuous variables
  - Discrete model again
  - Own set of problems
    - Hard to find good discretisation
    - High granularity might be necessary
      → large ranges → large factors
    - Lose characteristics of variable
      - Not each value necessarily associated with a probability
      - Nearby values have similar probabilities → hard to capture in a discrete distribution (no notion of closeness between range values)
- Therefore, use models with continuous variables

Outline: 8. Continuous Space

A. Basics
- Continuous variables, probability density function, cumulative probability distribution
- Joint distribution, marginal density, conditional density

B. Gaussian models
- (Multivariate) Gaussian distribution
- (Parameterised) Gaussian Bayesian networks

C. Probabilistic Soft Logic (PSL)
- Modelling, semantics, inference task

Probability Density Function

- Continuous random variable R
  - Range $\mathcal{R}(R) = [0,1]$
- Function $p : \mathbb{R} \rightarrow \mathbb{R}$ is a probability density function (PDF) for R if it is a non-negative, integrable function s.t.

$$\int_{\mathcal{R}(R)} p(r)dr = 1$$

- For any a (and b) in event space

$$P(R \leq a) = \int_{-\infty}^{a} p(r)dr \qquad P(a \leq R \leq b) = \int_{a}^{b} p(r)dr$$

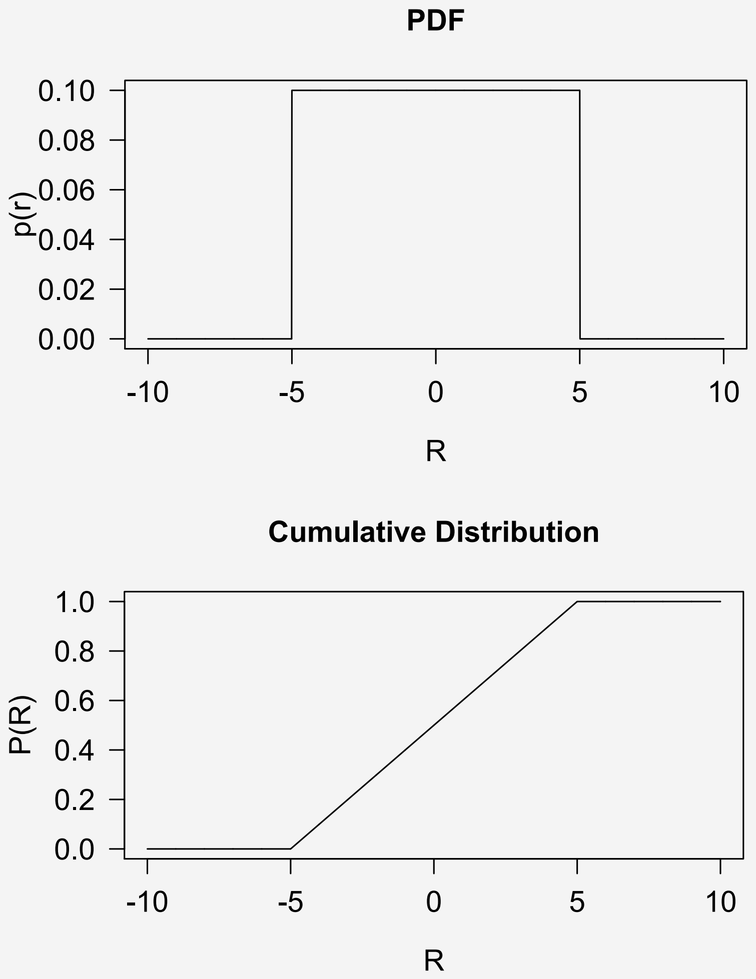
  - Function P is a cumulative distribution for R
  - Intuitively, value of $p(r)$ at point r is the incremental amount that r adds to the cumulative distribution during integration

PDFs: Uniform Distribution

- Continuous random variable R has a uniform distribution over $[a, b]$, denoted $R \sim \text{Unif}[a, b]$, if it has the PDF

$$p(r) = \begin{cases} \frac{1}{b-a} & b \geq r \geq a \\ 0 & \text{otherwise} \end{cases}$$

  - Density can be larger than 1 if $b - a < 1$
    - Can be legal if the total area under the pdf is 1



Joint/Multivariate Distribution

- Let P be a joint distribution over continuous random variables $R_1, \dots, R_n$
- Function $p(r_1, \dots, r_n)$ is a joint density function of $R_1, \dots, R_n$ if
  - $p(r_1, \dots, r_n) \geq 0$ for all values $r_1, \dots, r_n$ of $R_1, \dots, R_n$
  - p is an integrable function
  - For any choice $a_1, \dots, a_n$ and $b_1, \dots, b_n$,

$$P(a_1 \leq R_1 \leq b_1, \dots, a_n \leq R_n \leq b_n)$$

$$= \int_{a_1}^{b_1} \dots \int_{a_n}^{b_n} p(r_1, \dots, r_n) dr_1 \dots r_n$$

Marginal Density

- Given a joint density, integrate out the non-query random variables
  - E.g., given $p(r, s)$ a joint density for random variables R, S, then

$$p(r) = \int_{-\infty}^{+\infty} p(r, s)\, ds$$

- Shorthand notations
  - $p_R = p(r)$ marginal density
  - $p_{R,S} = p(r, s)$ joint density

Conditional Density Function

- Discrete case: $P(S|R=r) = \frac{P(S,R=r)}{P(R=r)}$
  - Problem in continuous case: $P(R=r) = 0$
    $\rightarrow P(S|R=r)$ undefined
- To avoid problem, condition on event
  $r - \epsilon \leq R \leq r + \epsilon$ and consider limit when $\epsilon \longrightarrow 0$

$$P(S|r) = \lim_{\epsilon \longrightarrow 0} P(S|r - \epsilon \leq R \leq r + \epsilon)$$

- If a continuous joint density $p(r,s)$ exists, derive form of this expression:

$$p(s|r) = \frac{p(r,s)}{p(r)}$$

  - If $p(r) = 0$, conditional density undefined
  - Chain rule and Bayes' rule hold as well:

$$p(r,s) = p(r)p(s|r) \qquad p(s|r) = \frac{p(s)p(r|s)}{p(r)}$$

Outline: 8. Continuous Space

A. Basics

- Continuous variables, probability density function, cumulative probability distribution
- Joint distribution, marginal density, conditional density

B. Gaussian models

- (Multivariate) Gaussian distribution
- (Parameterised) Gaussian Bayesian networks

C. Probabilistic Soft Logic (PSL)

- Modelling, semantics, inference task

Models with Continuous Variables

- Problem: Space of possible parameterisation essentially unbounded
- Special case: (Multivariate) Gaussian distributions
  - Two parameters per variable: mean, variance
  - Strong assumptions, e.g.,
    - Exponential decay away from its mean
    - Linearity of interactions between random variables

      → Assumptions often invalid but still work as a good approximation for many real-world distributions
  - Many generalisations exist which use Gaussians as a foundation
    - Non-linear interactions
    - Mixture of Gaussians

PDFs: Gaussian/Normal Distribution

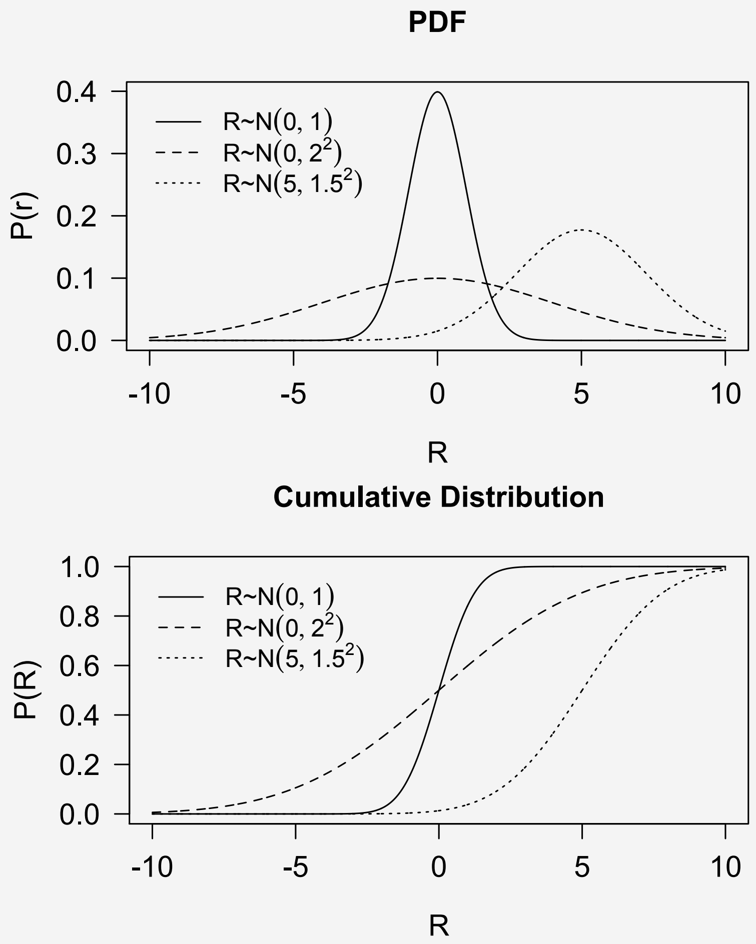
- Continuous random variable R has a Gaussian distribution with mean μ and variance σ^2, denoted $R \sim \mathcal{N}(\mu, \sigma^2)$, if it has the PDF

$$p(r) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(r-\mu)^2}{2\sigma^2}}$$

  - Expected value and variance of R given by μ and σ^2
    - Standard deviation: σ

Standard Gaussian $R \sim \mathcal{N}(\mu = 0, \sigma^2 = 1)$:

$$p(r) = \frac{1}{\sqrt{2\pi}} e^{-\frac{(r)^2}{2}}$$



Multivariate Gaussian

- Univariate Gaussian: Two parameters mean μ and variance σ^2
- Multivariate Gaussian distribution over continuous random variables $R_1, \dots, R_n$ characterised by n-dimensional mean vector $\boldsymbol{\mu}$ and symmetric $n \times n$ covariance matrix Σ
  - I.e., $\mathcal{N}(\boldsymbol{\mu}; \Sigma)$
- Density function defined as

$$p(\boldsymbol{r}) = \frac{1}{\sqrt{(2\pi)^n |\Sigma|}} \exp\left[-\frac{1}{2}(\boldsymbol{r} - \boldsymbol{\mu})^T \Sigma^{-1} (\boldsymbol{r} - \boldsymbol{\mu})\right]$$

  - $\boldsymbol{r} = (r_1, \dots, r_n)^T$
  - $|\Sigma|$ determinant of Σ
  - For well-defined density, Σ must be positive-definite
    - For any $\boldsymbol{r} \in \mathbb{R}^n$ such that $\boldsymbol{r} \neq 0 : \boldsymbol{r}^T \Sigma \boldsymbol{r} > 0$
    - Guaranteed to be non-singular → non-zero determinant

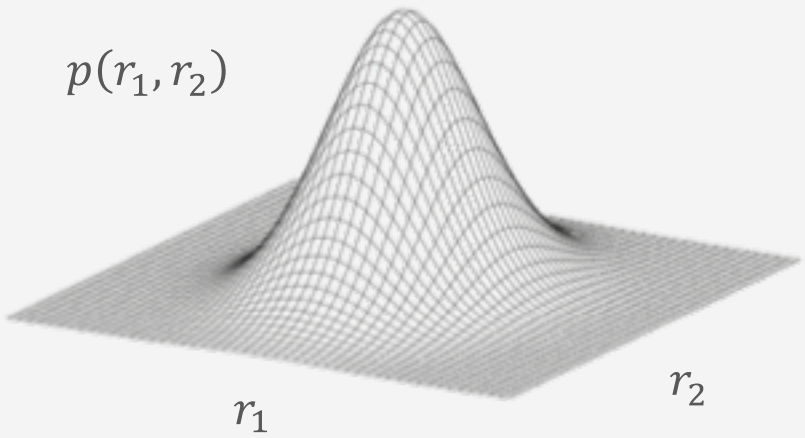
Standard multivariate Gaussian $R_1, \dots, R_n$ with
- $\boldsymbol{\mu} = \boldsymbol{0}$ (all-zero vector)
- $\Sigma = \boldsymbol{I}$ (identity matrix)

Example

- Joint Standard Gaussian distribution over two random variables R_1, R_2, i.e.,
  - $\mu = (0 \quad 0)^T, \Sigma = \boldsymbol{I}_2$
- Joint Gaussian distribution over three random variables R_1, R_2, R_3
  - Mean vector, covariance matrix:

$$\mu = \begin{pmatrix} 1 \\ -3 \\ 4 \end{pmatrix} \qquad \Sigma = \begin{pmatrix} 4 & 2 & -2 \\ 2 & 5 & -5 \\ -2 & -5 & 8 \end{pmatrix}$$

  - Covariance $Cov[R_1; R_3]$ ($Cov[R_2; R_3]$) negative, i.e., R_3 negatively correlated with R_1 (R_2)
    - When R_1 (R_2) goes up, R_3 goes down



Marginalisation

- Trivial with covariance matrix:
  - Compute pairwise covariances, i.e., generating the distribution in its covariance form
  - Given covariance form Σ: Read off from $\boldsymbol{\mu}, \Sigma$
- Assume a joint Gaussian distribution over $\{\boldsymbol{R}, \boldsymbol{T}\}$ where $\boldsymbol{R} \in \mathbb{R}^n$ and $\boldsymbol{T} \in \mathbb{R}^m$
  - One can decompose mean and covariance:

$$p(\boldsymbol{r}, \boldsymbol{t}) = \mathcal{N}\left(\begin{pmatrix}\boldsymbol{\mu}_R \\ \boldsymbol{\mu}_T\end{pmatrix}; \begin{bmatrix}\Sigma_{\boldsymbol{RR}} & \Sigma_{\boldsymbol{RT}} \\ \Sigma_{\boldsymbol{TR}} & \Sigma_{\boldsymbol{TT}}\end{bmatrix}\right)$$

  - where
    - $\boldsymbol{\mu_R} \in \mathbb{R}^n$, $\boldsymbol{\mu_T} \in \mathbb{R}^m$,
    - $\Sigma_{\boldsymbol{RR}}$ an $n \times n$ matrix, $\Sigma_{\boldsymbol{RT}}$ an $n \times m$ matrix,
      $\Sigma_{\boldsymbol{TR}} = \Sigma_{\boldsymbol{RT}}^T$ an $m \times n$ matrix, $\Sigma_{\boldsymbol{TT}}$ a $m \times m$ matrix
- Then, marginal distribution over $\boldsymbol{T}$ given by Gaussian distribution of $\mathcal{N}(\boldsymbol{\mu_T}; \Sigma_{\boldsymbol{TT}})$

Example

- Given joint Gaussian distribution over three random variables R_1, R_2, R_3
  - Mean vector, covariance matrix:

$$\mu = \begin{pmatrix} 1 \\ -3 \\ 4 \end{pmatrix} \qquad \Sigma = \begin{pmatrix} 4 & 2 & -2 \\ 2 & 5 & -5 \\ -2 & -5 & 8 \end{pmatrix}$$

- $p(R_1, R_2)$ given by Gaussian distribution with

$$\mu = \begin{pmatrix} 1 \\ -3 \end{pmatrix} \qquad \Sigma = \begin{pmatrix} 4 & 2 \\ 2 & 5 \end{pmatrix}$$

Dual: Information/Precision Form

- Rewrite $\exp\left[-\frac{1}{2}(\boldsymbol{r}-\boldsymbol{\mu})^T\Sigma^{-1}(\boldsymbol{r}-\boldsymbol{\mu})\right]$ by setting $\Gamma = \Sigma^{-1}$ and multiplying out:

$$-\frac{1}{2}(\boldsymbol{r}-\boldsymbol{\mu})^T\Gamma(\boldsymbol{r}-\boldsymbol{\mu}) = -\frac{1}{2}[\boldsymbol{r}^T\Gamma\boldsymbol{r} - 2\boldsymbol{r}^T\Gamma\boldsymbol{\mu} + \boldsymbol{\mu}^T\Gamma\boldsymbol{\mu}]$$

- $\boldsymbol{\mu}^T\Gamma\boldsymbol{\mu}$ is constant over the different $\boldsymbol{r}$, therefore,

$$p(\boldsymbol{r}) \propto \exp\left(-\frac{1}{2}[\boldsymbol{r}^T\Gamma\boldsymbol{r} - 2\boldsymbol{r}^T\Gamma\boldsymbol{\mu}]\right)$$
$$= \exp\left[-\frac{1}{2}\boldsymbol{r}^T\Gamma\boldsymbol{r} + \boldsymbol{r}^T\Gamma\boldsymbol{\mu}\right]$$
$$= \exp\left[-\frac{1}{2}\boldsymbol{r}^T\Gamma\boldsymbol{r} + (\Gamma\boldsymbol{\mu})^T\boldsymbol{r}\right]$$

$$\boldsymbol{r}^T\Gamma\boldsymbol{\mu}$$
$$= (\boldsymbol{r}^T\Gamma\boldsymbol{\mu})^{T^T} \quad \triangleright\ A^{T^T} = A$$
$$= \left((\Gamma\boldsymbol{\mu})^T\boldsymbol{r}^{T^T}\right)^T \quad \triangleright\ (AB)^T = B^T A^T$$
$$= \left((\Gamma\boldsymbol{\mu})^T\boldsymbol{r}\right)^T \quad \triangleright\ A^{T^T} = A$$
$$= (\Gamma\boldsymbol{\mu})^T\boldsymbol{r} \quad \triangleright\ k^T = k,\ k \text{ a scalar}$$

  - $\Gamma\boldsymbol{\mu}$ called potential vector

Dual: Information/Precision Form

- For a decomposition $\{\boldsymbol{R}, \boldsymbol{T}\}$ where $\boldsymbol{R} \in \mathbb{R}^n$ and $\boldsymbol{T} \in \mathbb{R}^m$:

$$\Gamma = \Sigma^{-1} = \begin{bmatrix} \Sigma_{RR} & \Sigma_{RT} \\ \Sigma_{TR} & \Sigma_{TT} \end{bmatrix}^{-1} = \begin{bmatrix} \Gamma_{RR} & \Gamma_{RT} \\ \Gamma_{TR} & \Gamma_{TT} \end{bmatrix}$$

- Getting to Σ
  - $\Sigma_{RR} = \left(\Gamma_{RR} - \Gamma_{RT}\Gamma_{TT}^{-1}\Gamma_{TR}\right)^{-1}$
  - $\Sigma_{TT} = \left(\Gamma_{TT} - \Gamma_{TR}\Gamma_{RR}^{-1}\Gamma_{RT}\right)^{-1}$
  - $\Sigma_{RT} = -\Gamma_{RR}^{-1}\Gamma_{RT}\left(\Gamma_{TT} - \Gamma_{TR}\Gamma_{RR}^{-1}\Gamma_{RT}\right)^{-1} = \Sigma_{TR}^{T}$
  - $\Sigma_{TR} = -\Gamma_{TT}^{-1}\Gamma_{TR}\left(\Gamma_{RR} - \Gamma_{RT}\Gamma_{TT}^{-1}\Gamma_{TR}\right)^{-1} = \Sigma_{RT}^{T}$
- Getting to Γ
  - $\Gamma_{RR} = \left(\Sigma_{RR} - \Sigma_{RT}\Sigma_{TT}^{-1}\Sigma_{TR}\right)^{-1}$
  - $\Gamma_{TT} = \left(\Sigma_{TT} - \Sigma_{TR}\Sigma_{RR}^{-1}\Sigma_{RT}\right)^{-1}$
  - $\Gamma_{RT} = -\Sigma_{RR}^{-1}\Sigma_{RT}\left(\Sigma_{TT} - \Sigma_{TR}\Sigma_{RR}^{-1}\Sigma_{RT}\right)^{-1} = \Gamma_{TR}^{T}$
  - $\Gamma_{TR} = -\Sigma_{TT}^{-1}\Sigma_{TR}\left(\Sigma_{RR} - \Sigma_{RT}\Sigma_{TT}^{-1}\Sigma_{TR}\right)^{-1} = \Gamma_{RT}^{T}$

Conditioning

- Conditioning a Gaussian on observations $\boldsymbol{E} = \boldsymbol{e}$ easy to perform in the information form by setting $\boldsymbol{E}$ to $\boldsymbol{e}$ in one of the following

$$p(\boldsymbol{r}) \propto \exp\left[-\frac{1}{2}(\boldsymbol{r}-\boldsymbol{\mu})^T\Gamma(\boldsymbol{r}-\boldsymbol{\mu})\right]$$

$$\propto \exp\left[-\frac{1}{2}\boldsymbol{r}^T\Gamma\boldsymbol{r} + (\Gamma\boldsymbol{\mu})^T\boldsymbol{r}\right]$$

- Assuming a decomposition into $\boldsymbol{R}$ and $\boldsymbol{E}$, i.e.,

$$p(\boldsymbol{r},\boldsymbol{e}) = \mathcal{N}\left(\begin{pmatrix}\boldsymbol{\mu}_R \\ \boldsymbol{\mu}_E\end{pmatrix}; \begin{bmatrix}\Sigma_{\boldsymbol{RR}} & \Sigma_{\boldsymbol{RE}} \\ \Sigma_{\boldsymbol{ER}} & \Sigma_{\boldsymbol{EE}}\end{bmatrix}\right)$$

$$\propto \exp\left[-\frac{1}{2}\left(\begin{pmatrix}\boldsymbol{r} \\ \boldsymbol{e}\end{pmatrix} - \begin{pmatrix}\boldsymbol{\mu}_R \\ \boldsymbol{\mu}_E\end{pmatrix}\right)^T \begin{bmatrix}\Gamma_{\boldsymbol{RR}} & \Gamma_{\boldsymbol{RT}} \\ \Gamma_{\boldsymbol{TR}} & \Gamma_{\boldsymbol{TT}}\end{bmatrix}\left(\begin{pmatrix}\boldsymbol{r} \\ \boldsymbol{e}\end{pmatrix} - \begin{pmatrix}\boldsymbol{\mu}_R \\ \boldsymbol{\mu}_E\end{pmatrix}\right)\right]$$

In the exponential function:

$$-\frac{1}{2}\left(\begin{pmatrix} r \\ e \end{pmatrix} - \begin{pmatrix} \mu_R \\ \mu_E \end{pmatrix}\right)^T \begin{bmatrix} \Gamma_{RR} & \Gamma_{RE} \\ \Gamma_{ER} & \Gamma_{EE} \end{bmatrix} \left(\begin{pmatrix} r \\ e \end{pmatrix} - \begin{pmatrix} \mu_R \\ \mu_E \end{pmatrix}\right)$$

$$= -\frac{1}{2}\begin{pmatrix} r-\mu_R \\ e-\mu_E \end{pmatrix}^T \begin{bmatrix} \Gamma_{RR} & \Gamma_{RE} \\ \Gamma_{ER} & \Gamma_{EE} \end{bmatrix} \begin{pmatrix} r-\mu_R \\ e-\mu_E \end{pmatrix}$$

$$= -\frac{1}{2}(r-\mu_R)^T\Gamma_{RR}(r-\mu_R) - \frac{1}{2}2(r-\mu_R)^T\Gamma_{RE}(e-\mu_E) - \underbrace{\frac{1}{2}(e-\mu_E)^T\Gamma_{EE}(e-\mu_E)}_{\text{Does not depend on } r}$$

$$\propto -\frac{1}{2}(r-\mu_R)^T\Gamma_{RR}(r-\mu_R) - (r-\mu_R)^T\Gamma_{RE}(e-\mu_E)$$

$$= -\frac{1}{2}(r-\mu_R)^T\Gamma_{RR}(r-\mu_R) - (r-\mu_R)^T\Gamma_{RE}(e-\mu_E) - A + A$$

$$p(r) \propto \exp\left[-\frac{1}{2}(r-\mu)^T\Sigma^{-1}(r-\mu)\right] = \exp\left[-\frac{1}{2}(r-\mu)^T\Gamma(r-\mu)\right]$$

Use $-A$ to get expression into the form $(r-\mu)^T\Gamma(r-\mu)$ by factoring out Γ_{RR}

$$A = \frac{1}{2}(e-\mu_E)\Gamma_{ER}\Gamma_{RR}^{-1}\Gamma_{RR}\Gamma_{RR}^{-1}\Gamma_{RE}(e-\mu_E)$$

$$\exp\left[-\frac{1}{2}\left(\left(r-\mu_R+\Gamma_{RR}^{-1}\Gamma_{ER}(e-\mu_E)\right)^T \Gamma_{RR}\left(r-\mu_R+\Gamma_{RR}^{-1}\Gamma_{RE}(e-\mu_E)\right)\right)\right]\exp[A]$$

$$\propto \exp\left[-\frac{1}{2}\left(\left(r\underbrace{-\mu_R+\Gamma_{RR}^{-1}\Gamma_{ER}(e-\mu_E)}\right)^T \underbrace{\Gamma_{RR}}\left(r\underbrace{-\mu_R+\Gamma_{RR}^{-1}\Gamma_{ER}(e-\mu_E)}\right)\right)\right]$$

$$\mu^* = \mu_R - \Gamma_R^{-1}\Gamma_{ER}(e-\mu_E) \qquad \Sigma^* = \Gamma_{RR}$$

Conditioning

- Conditioning a Gaussian on observations $\boldsymbol{E} = \boldsymbol{e}$ with remaining random variables $\boldsymbol{R}$
- Result:

$$\boldsymbol{R}|\boldsymbol{E} = \boldsymbol{e} \sim \mathcal{N}(\mu^*, \Sigma^*)$$

  - Information form:
    - $\mu^* = \mu_{\boldsymbol{R}} - \Gamma_{\boldsymbol{R}}^{-1}\Gamma_{\boldsymbol{ER}}(\boldsymbol{e} - \mu_{\boldsymbol{E}})$
    - $\Sigma^* = \Gamma_{\boldsymbol{RR}}$
  - Covariance form:
    - $\mu^* = \mu_{\boldsymbol{R}} + \Sigma_{\boldsymbol{RE}}\Sigma_{\boldsymbol{EE}}^{-1}(\boldsymbol{e} - \mu_{\boldsymbol{E}})$
    - $\Sigma^* = \Sigma_{\boldsymbol{RR}} - \Sigma_{\boldsymbol{RE}}\Sigma_{\boldsymbol{EE}}^{-1}\Sigma_{\boldsymbol{ER}}$
  - Mean moved from $\mu_{\boldsymbol{R}}$ according to correlation and variance on observations $\Sigma_{\boldsymbol{RE}}\Sigma_{\boldsymbol{EE}}^{-1}(\boldsymbol{e} - \mu_{\boldsymbol{E}})$
  - Covariance does not depend on observations $\boldsymbol{e}$

Query Answering: Summary

- For marginalisation, read off parameters in covariance form
  - Marginal query for $\boldsymbol{T}$: $\mathcal{N}(\boldsymbol{\mu}_{\boldsymbol{T}}; \Sigma_{\boldsymbol{TT}})$
- For conditioning, one needs to invert the covariance matrix to obtain the information form
  - Conditioning on $\boldsymbol{E} = \boldsymbol{e}$:

$$\boldsymbol{R} | \boldsymbol{E} = \boldsymbol{e} \sim \mathcal{N}(\mu^*, \Sigma^*)$$

    - In covariance form
      - $\mu^* = \mu_{\boldsymbol{R}} + \Sigma_{\boldsymbol{RE}} \Sigma_{\boldsymbol{EE}}^{-1} (\boldsymbol{e} - \mu_{\boldsymbol{E}})$
      - $\Sigma^* = \Sigma_{\boldsymbol{RR}} - \Sigma_{\boldsymbol{RE}} \Sigma_{\boldsymbol{EE}}^{-1} \Sigma_{\boldsymbol{ER}}$
  - Matrix inversion can be very costly!

Linear Gaussian Model

- Let S be a continuous random variable with continuous parents $R_1, \dots, R_k$
- S has a linear Gaussian model if there are parameters $\beta_0, \dots, \beta_k$ and σ^2 such that

$$p(S|r_1, \dots, r_k) = \mathcal{N}(\beta_0 + \beta_1 r_1 + \cdots + \beta_k r_k; \sigma^2)$$
$$= \mathcal{N}(\beta_0 + \boldsymbol{\beta}^T \boldsymbol{r}; \sigma^2)$$

← (vector notation)

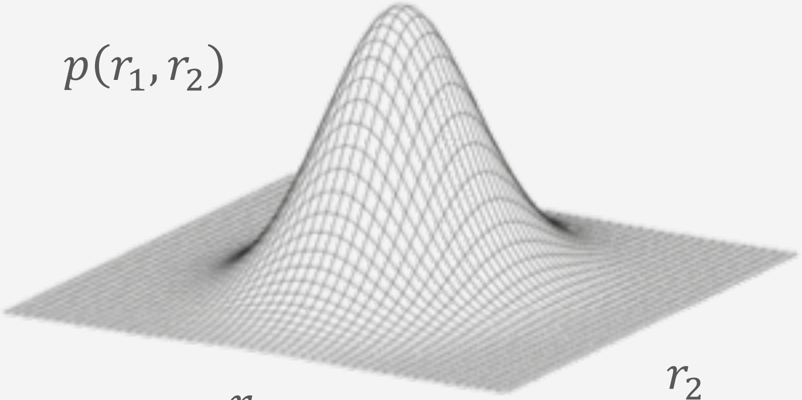
  - $p(S|r_1, \dots, r_k)$ a conditional probability distribution (CPD)
  - Interpretations
    - β_0 is an initial mean μ_0 that is moved according to the influences by the parents
    - S is a linear function of $R_1, \dots, R_k$ with the addition of Gaussian noise: $S = \beta_0 + \beta_1 r_1 + \cdots + \beta_k r_k + \epsilon$
      - ϵ a Gaussian random variable with mean 0 and variance σ^2, representing the noise in the system
  - Does not allow σ^2 to depend on parent values
    - But can be a useful approximation

Independencies in Gaussians

- Let random variables $R_1, \dots, R_n$ have a joint distribution $\mathcal{N}(\boldsymbol{\mu}; \Sigma)$
- Then, R_i, R_j independent if and only if $\Sigma_{ij} = 0$
  - Joint distribution needs to be Gaussian for this equivalence to hold
    - If the distribution is not Gaussian, $\Sigma_{ij} = 0$ might be the case and there still might be a dependence between R_i, R_j
- Conditional independence can be read of in the inverse of the covariance matrix, Σ^{-1}
  - Given a Gaussian distribution $p(r_1, \dots, r_n) = \mathcal{N}(\boldsymbol{\mu}; \Sigma)$
  - Then, $\Sigma_{ij}^{-1} = 0$ iff $p \vDash \left(R_i \perp R_j | \{R_1, \dots, R_n\} \setminus \{R_i, R_j\}\right)$

Example

- Joint Standard Gaussian distribution over two random variables R_1, R_2, i.e.,
  - $\mu = (0 \quad 0)^T, \Sigma = \boldsymbol{I}_2$
  - R_1, R_2 independent as $\Sigma_{ij} = \Sigma_{ji} = 0$
- Gaussian for R_1, R_2, R_3 from before
  - Covariance and inverse covariance matrix:
    $$\Sigma = \begin{pmatrix} 4 & 2 & -2 \\ 2 & 5 & -5 \\ -2 & -5 & 8 \end{pmatrix} \quad \Sigma^{-1} = \begin{pmatrix} 0.3125 & -0.125 & 0 \\ -0.125 & 0.5833 & 0.3333 \\ 0 & 0.3333 & 0.3333 \end{pmatrix}$$
  - R_1, R_3 conditionally independent given R_2
  - $\Sigma^{-1}_{13} = 0$ iff
    $p \models (R_1 \perp R_3 | \{R_1, R_2, R_3\} \setminus \{R_1, R_3\}) = (R_1 \perp R_3 | R_2)$



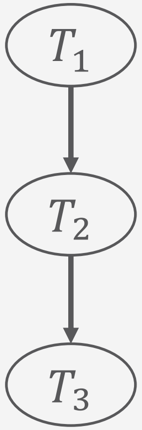
Gaussian Bayesian Network (GBN)

- Factorisation of a joint distribution into factors also possible with linear Gaussians as local CPDs
- A BN is a directed acyclic graph G whose nodes are discrete random variables $\{R_1, \ldots, R_n\}$ and whose full joint P_G factorises according to the local CPTs, i.e.,

$$P_G = \prod_i P\big(R_i | parents(R_i)\big)$$

- Gaussian BN is a BN where
  - R_i are continuous random variables
  - All CPDs are linear Gaussians

- E.g., $T_1 \to T_2 \to T_3$ (also depicted below)
  - $p(T_1) = \mathcal{N}(1; 4)$
  - $p(T_2|T_1) = \mathcal{N}(-3.5 + 0.5 \cdot T_1; 4)$
  - $p(T_3|T_2) = \mathcal{N}(1 + (-1) \cdot T_2; 3)$



Connection to Multivariate Gaussian

- Linear GBN an alternative representation to multivariate Gaussian distribution
  - A linear Gaussian BN always defines a joint multivariate Gaussian distribution
- Let S be a linear Gaussian of its parents $R_1, \dots, R_k$
  - $\mathcal{N}(\beta_0 + \boldsymbol{\beta}^T \boldsymbol{r}; \sigma^2) = \mathcal{N}(\beta_0 + \beta_1 r_1 + \cdots + \beta_k r_k; \sigma^2)$
  - $R_1, \dots, R_k$ jointly Gaussian with $\mathcal{N}(\boldsymbol{\mu}; \Sigma)$
- Distribution of S is a Gaussian $p(S) = \mathcal{N}(\mu_S; \sigma_S^2)$ with

$$\mu_S = \beta_0 + \boldsymbol{\beta}^T \boldsymbol{r}$$
$$\sigma_S^2 = \sigma^2 + \boldsymbol{\beta}^T \Sigma \boldsymbol{\beta}$$

- Joint distribution over $\{R_1, \dots, R_k, S\}$ is a Gaussian with

$$Cov[R_i; S] = \sum_{j=1}^{k} \beta_j \Sigma_{ij}$$

General Procedure for Conversion

- Let $(R_1, \dots, R_n)$ be the random variables of a GBN
  - Each R_i is a Gaussian $\mathcal{N}(\beta_0 + \boldsymbol{\beta}^T \boldsymbol{r}; \sigma^2)$ conditional on its parents $parents(R_i)$
  - $(R_1, \dots, R_n)$ follows a topological ordering θ such that
  $$\forall R_j \in \{R_1, \dots, R_n\} : \forall R_i \in parents(R_j) : R_i \prec_\theta R_j$$
  - Build a matrix $B^{n \times n}$ that has a non-zero entry β_{ij} if there exists a parent-child relation $R_i \to R_j$ with β_{ij} being the factor for R_i in the $\boldsymbol{\beta}$ of R_j
  $$B = \begin{pmatrix} 0 & \beta_{12} & \dots & \beta_{1n} \\ 0 & 0 & & \beta_{2n} \\ & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 0 \end{pmatrix}$$
    - i chooses the row(s) , j chooses the column(s)
    - B is upper-triangular because no loops allowed in BNs
      - Including self-loops → $\beta_{ii} = 0$ as well

Ross D. Shachter and C. Robert Kenley: Gaussian Influence Diagrams. In: *Management Science*, 1989.

General Procedure for Conversion

- Joint distribution $p(r_1, \dots, r_n)$ given by $\mathcal{N}(\boldsymbol{\mu}, \Sigma)$
  - Means
    $\boldsymbol{\mu} = \left(\mu_1, \beta_{0,2} + \boldsymbol{\beta}_2^T \boldsymbol{r}, \dots, \beta_{0,n} + \boldsymbol{\beta}_n^T \boldsymbol{r}\right)^T$
  - Covariance (*recursive rules*): $j \in \{2, \dots, n\}, i = 1 \dots j-1$

$$\Sigma_{11} \leftarrow \sigma_1^2$$
$$\Sigma_{ij} \leftarrow \Sigma_{ii} B_{ij}$$
$$\Sigma_{ji} \leftarrow \Sigma_{ij}^T$$
$$\Sigma_{jj} \leftarrow \sigma_j^2 + \Sigma_{ji} B_{ij}$$

    - First index chooses the row(s), second index chooses the column(s)

- Example: Given B

$$B = \begin{pmatrix} 0 & \beta_{12} & \dots & \beta_{1n} \\ 0 & 0 & & \beta_{2n} \\ & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & 0 \end{pmatrix}$$

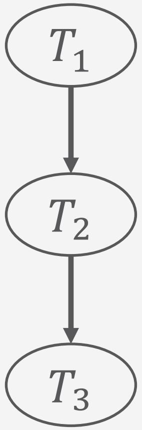
filling Σ layer-wise

$$\Sigma = \begin{pmatrix} \Sigma_{11} & \Sigma_{12} & \dots & \Sigma_{1n} \\ \Sigma_{21} & \Sigma_{22} & & \Sigma_{2n} \\ \vdots & & \ddots & \vdots \\ \Sigma_{n1} & \Sigma_{n2} & \dots & \Sigma_{nn} \end{pmatrix}$$

Ross D. Shachter and C. Robert Kenley: Gaussian Influence Diagrams. In: *Management Science*, 1989.

GBN: Conversion Example

- GBN



$p(T_1) = \mathcal{N}(1; 4)$

$p(T_2|T_1) = \mathcal{N}(-3.5 + 0.5 \cdot T_1; 4)$

$p(T_3|T_2) = \mathcal{N}(1 + (-1) \cdot T_2; 3)$

- Goal: Joint distribution

$$p(t_1, t_2, t_3) = \mathcal{N}(\boldsymbol{\mu}; \Sigma)$$

$$\boldsymbol{\mu} = \begin{pmatrix} \mu_1 \\ \mu_2 \\ \mu_3 \end{pmatrix}$$

$$\Sigma = \begin{pmatrix} \Sigma_{11} & \Sigma_{12} & \Sigma_{13} \\ \Sigma_{21} & \Sigma_{22} & \Sigma_{23} \\ \Sigma_{31} & \Sigma_{32} & \Sigma_{33} \end{pmatrix}$$

- Matrix B
  - $B_{12}: T_1 \to T_2, \beta_1 = 0.5$
  - $B_{23}: T_2 \to T_3, \beta_1 = -1$
  - Rest: zeroes

$$B = \begin{pmatrix} 0 & 0.5 & 0 \\ 0 & 0 & -1 \\ 0 & 0 & 0 \end{pmatrix}$$

- Means
  - $\mu_1 = 1$
  - $\mu_2 = -3.5 + 0.5 \cdot \mu_1 = -3$
  - $\mu_3 = 1 + (-1) \cdot \mu_2 = 4$

$$\boldsymbol{\mu} = \begin{pmatrix} 1 \\ -3 \\ 4 \end{pmatrix}$$

GBN: Conversion Example

- Filling Σ:
  - $\Sigma = \begin{pmatrix} \Sigma_{11} & \Sigma_{12} & \Sigma_{13} \\ \Sigma_{21} & \Sigma_{22} & \Sigma_{23} \\ \Sigma_{31} & \Sigma_{32} & \Sigma_{33} \end{pmatrix}$
- Need B and the recursive rules
  - $B = \begin{pmatrix} 0 & 0.5 & 0 \\ 0 & 0 & -1 \\ 0 & 0 & 0 \end{pmatrix}$
  - First index: row(s)
  - Second index: column(s)

$$\Sigma_{11} \leftarrow \sigma_1^2$$
$$\Sigma_{ij} \leftarrow \Sigma_{ii} B_{ij}$$
$$\Sigma_{ji} \leftarrow \Sigma_{ij}^T$$
$$\Sigma_{jj} \leftarrow \sigma_j^2 + \Sigma_{ji} B_{ij}$$

- $\Sigma_{11} = \sigma_1^2 = 4$
- $j = 2, i = 1$
  - $\Sigma_{12} = \Sigma_{11} B_{12} = 4 \cdot 0.5 = 2$
  - $\Sigma_{21} = \Sigma_{12}^T = 2^T = 2$
  - $\Sigma_{22} = \sigma_2^2 + \Sigma_{21} B_{12} = 4 + 2 \cdot 0.5 = 5$

$$\begin{pmatrix} 4 & \Sigma_{12} & \Sigma_{13} \\ \Sigma_{21} & \Sigma_{22} & \Sigma_{23} \\ \Sigma_{31} & \Sigma_{32} & \Sigma_{33} \end{pmatrix}$$

$$\begin{pmatrix} 4 & 2 & \Sigma_{13} \\ \Sigma_{21} & \Sigma_{22} & \Sigma_{23} \\ \Sigma_{31} & \Sigma_{32} & \Sigma_{33} \end{pmatrix}$$

$$\begin{pmatrix} 4 & 2 & \Sigma_{13} \\ 2 & \Sigma_{22} & \Sigma_{23} \\ \Sigma_{31} & \Sigma_{32} & \Sigma_{33} \end{pmatrix}$$

$$\begin{pmatrix} 4 & 2 & \Sigma_{13} \\ 2 & 5 & \Sigma_{23} \\ \Sigma_{31} & \Sigma_{32} & \Sigma_{33} \end{pmatrix}$$

GBN: Conversion Example

- Filling Σ:
  - $\Sigma = \begin{pmatrix} \Sigma_{11} & \Sigma_{12} & \Sigma_{13} \\ \Sigma_{21} & \Sigma_{22} & \Sigma_{23} \\ \Sigma_{31} & \Sigma_{32} & \Sigma_{33} \end{pmatrix}$
- Need B and the recursive rules
  - $B = \begin{pmatrix} 0 & 0.5 & 0 \\ 0 & 0 & -1 \\ 0 & 0 & 0 \end{pmatrix}$

$\Sigma_{11} \leftarrow \sigma_1^2$
$\Sigma_{ij} \leftarrow \Sigma_{ii} B_{ij}$
$\Sigma_{ji} \leftarrow \Sigma_{ij}^T$
$\Sigma_{jj} \leftarrow \sigma_j^2 + \Sigma_{ji} B_{ij}$

  - First index: row(s)
  - Second index: column(s)

- $j = 3, i = 12$
  - $\Sigma_{(12)3}$
    $= \Sigma_{(12)(12)} B_{(12)3}$
    $= \begin{pmatrix} 4 & 2 \\ 2 & 5 \end{pmatrix} \begin{pmatrix} 0 \\ -1 \end{pmatrix} = \begin{pmatrix} -2 \\ -5 \end{pmatrix}$
  - $\Sigma_{3(12)}$
    $= \Sigma_{(12)3}^T$
    $= \begin{pmatrix} -2 \\ -5 \end{pmatrix}^T = \begin{pmatrix} -2 & -5 \end{pmatrix}$
  - Σ_{33}
    $= \sigma_3^2 + \Sigma_{3(12)} B_{(12)3}$
    $= 3 + \begin{pmatrix} -2 & -5 \end{pmatrix} \begin{pmatrix} 0 \\ -1 \end{pmatrix}$
    $= 3 + 5 = 8$

$$\begin{pmatrix} 4 & 2 & \Sigma_{13} \\ 2 & 5 & \Sigma_{23} \\ \Sigma_{31} & \Sigma_{32} & \Sigma_{33} \end{pmatrix}$$

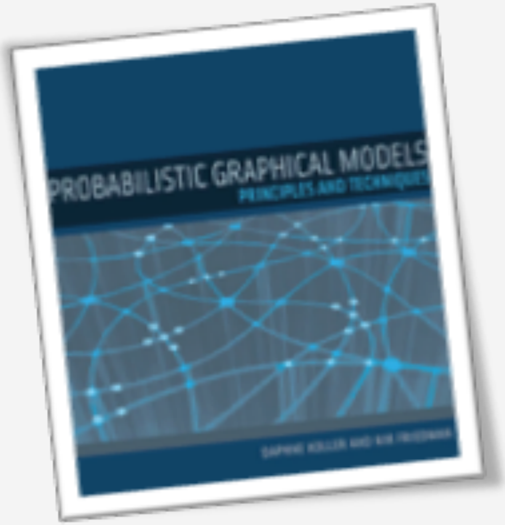
$$\begin{pmatrix} 4 & 2 & -2 \\ 2 & 5 & -5 \\ \Sigma_{31} & \Sigma_{32} & \Sigma_{33} \end{pmatrix}$$

$$\begin{pmatrix} 4 & 2 & -2 \\ 2 & 5 & -5 \\ -2 & -5 & \Sigma_{33} \end{pmatrix}$$

$$\begin{pmatrix} 4 & 2 & -2 \\ 2 & 5 & -5 \\ -2 & -5 & 8 \end{pmatrix}$$

Inference in GBNs

- Inference in linear Gaussians with Variable Elimination
  - Representation through linear Gaussian CPDs instead of CPTs/factors
  - Modified operations for multiply/sum-out
- Message passing formulation
  - Approximate belief propagation
- Sampling in the continuous space
  - Rejection sampling, importance sampling, MCMC methods for GBNs
- Actually using the full joint
  - Marginalisation, conditioning as sketched in Basics

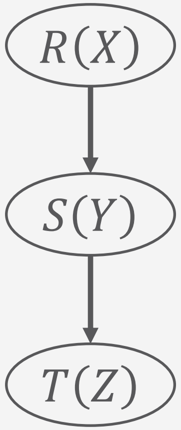


See Ch. 14 of PGM book for further information

Lifting the Full Joint

- Lifting conversion method by Shachter and Kenley for parameterised GBNs
  - GBN with PRVs $A_1, \dots, A_m$ as nodes
    - PDF for each A_i applies to each $R \in gr(A_i)$
    - $m \ll n, n = |\bigcup_i gr(A_i)|$
    - Semantics: grounding and forming full joint $p(\bigcup_i gr(A_i))$
    - Simple case: For all parent-child relations $R(\boldsymbol{X}) \rightarrow S(\boldsymbol{Y})$, it holds that $\boldsymbol{X} \cap \boldsymbol{Y} = \emptyset$
      - Each child instance has the same parent instances as its siblings
    - General case a bit more involved [Hartwig et al., 2021]

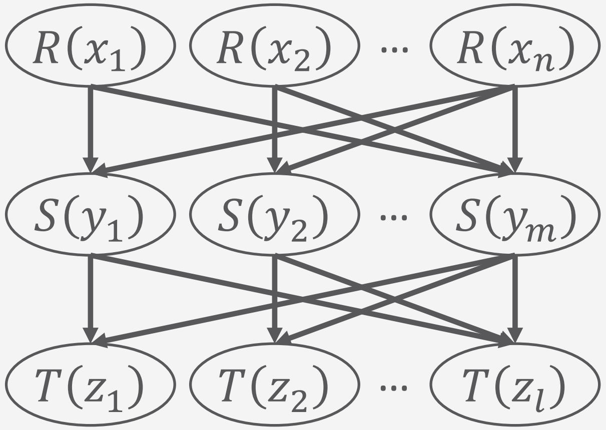
$\Sigma_{11} \leftarrow \sigma_1^2$
$\Sigma_{ij} \leftarrow \Sigma_{ii} B_{ij}$
$\Sigma_{ji} \leftarrow \Sigma_{ij}^T$
$\Sigma_{jj} \leftarrow \sigma_j^2 + \Sigma_{ji} B_{ij}$



$p(R(X)) = \mathcal{N}(1; 4)$

$p(S(Y)|R(X)) = \mathcal{N}(-3.5 + 0.5 \cdot T_1; 4)$

$p(T(Z)|S(Y)) = \mathcal{N}(1 + (-1) \cdot T_2; 3)$

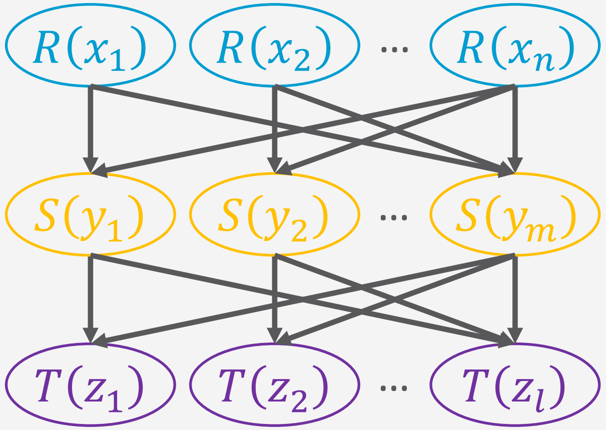


Mattis Hartwig and Ralf Möller. Lifted Query Answering in Gaussian Bayesian Networks. In: *PGM-20 Proceedings of the 10th International Conference on Probabilistic Graphical Models*, 2020.

Lifting the Full Joint: Simple Case

- With PRVs, matrix B and covariance matrix have liftable blocks for each PRV
  - Given the case of no overlaps in logical variables: B

$$
\begin{array}{c c}
 & \begin{matrix} \quad R(X) \quad & \qquad\qquad S(Y) \qquad\qquad & \qquad T(Z) \end{matrix} \\
\begin{matrix} R(X) \\ \\ \\ S(Y) \\ \\ \\ T(Z) \end{matrix} &
\begin{pmatrix}
0 & \cdots & 0 & \beta_{r_1 s_1} & \cdots & \beta_{r_1 s_m} & 0 & \cdots & 0 \\
\vdots & \ddots & \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\
0 & \cdots & 0 & \beta_{r_n s_1} & \cdots & \beta_{r_n s_m} & 0 & \cdots & 0 \\
0 & \cdots & 0 & 0 & \cdots & 0 & \beta_{s_1 t_1} & \cdots & \beta_{s_1 t_l} \\
\vdots & \ddots & \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\
0 & \cdots & 0 & 0 & \cdots & 0 & \beta_{s_m t_1} & \cdots & \beta_{s_m t_l} \\
0 & \cdots & 0 & 0 & \cdots & 0 & 0 & \cdots & 0 \\
\vdots & \ddots & \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\
0 & \cdots & 0 & 0 & \cdots & 0 & 0 & \cdots & 0
\end{pmatrix}
\end{array}
$$



Lifting the Full Joint: Simple Case

Each $R(x_i)$ has the same influence on each $S(y_j)$. Given $P(s(Y)|r(X)) = \mathcal{N}(\beta_0 + \beta_1 r(X); \sigma^2)$,

$$\beta_{r_i s_j} = \beta_1$$

for all $i \in \{1, \dots, n\}, j \in \{1, \dots, m\}$. The same holds for $S(y_j)$ and $T(z_k)$ (as well as $R(x_i)$ and $T(z_k)$, which has $\beta_{r_i t_k} = 0$).

Lifted B'

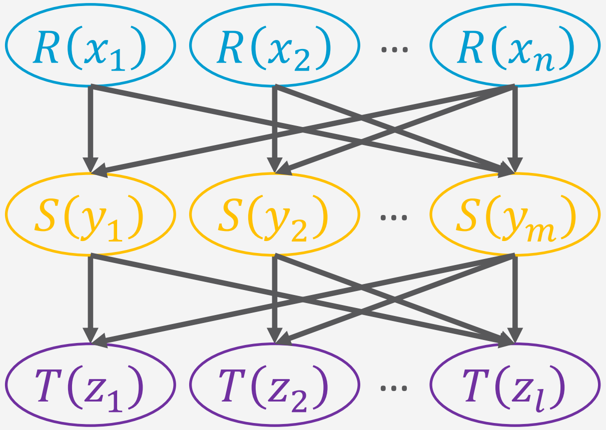
$$\begin{pmatrix} 0 & \beta_{rs} & 0 \\ 0 & 0 & \beta_{st} \\ 0 & 0 & 0 \end{pmatrix}$$

Lifted B'

$$\begin{pmatrix} 0 & 0.5 & 0 \\ 0 & 0 & -1 \\ 0 & 0 & 0 \end{pmatrix}$$

Columns: $R(X)$, $S(Y)$, $T(Z)$; rows: $R(X)$, $S(Y)$, $T(Z)$

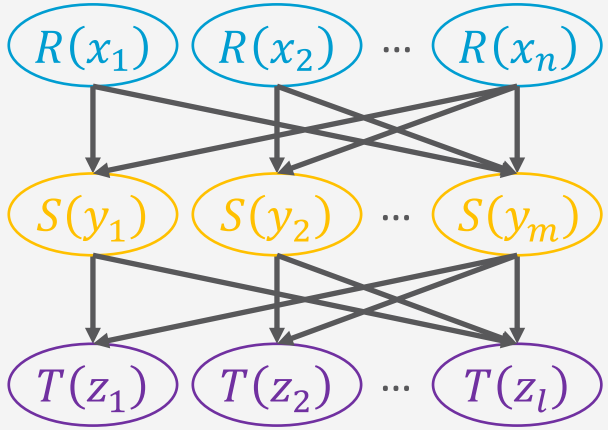
$$\begin{pmatrix}
0 & \dots & 0 & \beta_{r_1 s_1} & \cdots & \beta_{r_1 s_m} & 0 & \dots & 0 \\
\vdots & \ddots & \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\
0 & \dots & 0 & \beta_{r_n s_1} & \cdots & \beta_{r_n s_m} & 0 & \dots & 0 \\
0 & \dots & 0 & 0 & \dots & 0 & \beta_{s_1 t_1} & \cdots & \beta_{s_1 t_l} \\
\vdots & \ddots & \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\
0 & \dots & 0 & 0 & \dots & 0 & \beta_{s_m t_1} & \cdots & \beta_{s_m t_l} \\
0 & \dots & 0 & 0 & \dots & 0 & 0 & \dots & 0 \\
\vdots & \ddots & \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\
0 & \dots & 0 & 0 & \dots & 0 & 0 & \dots & 0
\end{pmatrix}$$



Lifting the Full Joint: Simple Case

- With PRVs, matrix B and covariance matrix have liftable blocks for each PRV
  - Given the case of no overlaps in logical variables: Σ

$$
\begin{array}{c} \\ R(X) \\ \\ S(Y) \\ \\ T(Z) \end{array}
\begin{array}{c}
\begin{array}{ccc} R(X) & S(Y) & T(Z) \end{array} \\
\begin{pmatrix}
\Sigma_{r_1 r_1} & \cdots & \Sigma_{r_1 r_n} & \Sigma_{r_1 s_1} & \cdots & \Sigma_{r_1 s_m} & \Sigma_{r_1 t_1} & \cdots & \Sigma_{r_1 t_l} \\
\vdots & \ddots & \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\
\Sigma_{r_n r_1} & \cdots & \Sigma_{r_n r_n} & \Sigma_{r_n s_1} & \cdots & \Sigma_{r_n s_m} & \Sigma_{r_n t_1} & \cdots & \Sigma_{r_n t_l} \\
\Sigma_{s_1 r_1} & \cdots & \Sigma_{s_1 r_n} & \Sigma_{s_1 s_1} & \cdots & \Sigma_{s_1 s_m} & \Sigma_{s_1 t_1} & \cdots & \Sigma_{s_1 t_l} \\
\vdots & \ddots & \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\
\Sigma_{s_m r_1} & \cdots & \Sigma_{s_m r_n} & \Sigma_{s_m s} & \cdots & \Sigma_{s_m s_m} & \Sigma_{s_m t_1} & \cdots & \Sigma_{s_m t_n} \\
\Sigma_{t_1 r_1} & \cdots & \Sigma_{t_1 r_n} & \Sigma_{t_1 s_1} & \cdots & \Sigma_{t_1 s_m} & \Sigma_{t_1 t_1} & \cdots & \Sigma_{t_1 t_l} \\
\vdots & \ddots & \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\
\Sigma_{t_l r_1} & \cdots & \Sigma_{t_l r_n} & \Sigma_{t_l s_1} & \cdots & \Sigma_{t_l s_m} & \Sigma_{t_l t_1} & \cdots & \Sigma_{t_l t_l}
\end{pmatrix}
\end{array}
$$



Lifting the Full Joint: Simple Case

$$\begin{pmatrix}
\Sigma_{r_1r_1} & \cdots & \Sigma_{r_1r_n} & \Sigma_{r_1s_1} & \cdots & \Sigma_{r_1s_m} & \Sigma_{r_1t_1} & \cdots & \Sigma_{r_1t_l} \\
\vdots & \ddots & \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\
\Sigma_{r_nr_1} & \cdots & \Sigma_{r_nr_n} & \Sigma_{r_ns_1} & \cdots & \Sigma_{r_ns_m} & \Sigma_{r_nt_1} & \cdots & \Sigma_{r_nt_l} \\
\Sigma_{s_1r_1} & \cdots & \Sigma_{s_1r_n} & \Sigma_{s_1s_1} & \cdots & \Sigma_{s_1s_m} & \Sigma_{s_1t_1} & \cdots & \Sigma_{s_1t_l} \\
\vdots & \ddots & \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\
\Sigma_{s_mr_1} & \cdots & \Sigma_{s_mr_n} & \Sigma_{s_ms} & \cdots & \Sigma_{s_ms_m} & \Sigma_{s_mt_1} & \cdots & \Sigma_{s_mt_n} \\
\Sigma_{t_1r_1} & \cdots & \Sigma_{t_1r_n} & \Sigma_{t_1s_1} & \cdots & \Sigma_{t_1s_m} & \Sigma_{t_1t_1} & \cdots & \Sigma_{t_1t_l} \\
\vdots & \ddots & \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\
\Sigma_{t_lr_1} & \cdots & \Sigma_{t_lr_n} & \Sigma_{t_ls_1} & \cdots & \Sigma_{t_ls_m} & \Sigma_{t_lt_1} & \cdots & \Sigma_{t_lt_l}
\end{pmatrix}$$

$$\Sigma_{r_1r_1} = \sigma^2_{R(X)}$$
$$\Sigma_{r_1r_2} = \sigma^2_{R(X)} B_{r_1r_2} = \sigma^2_{R(X)} B'_{11} = \sigma^2_{R(X)} \cdot 0 = 0$$
$$\Sigma_{r_2r_1} = 0$$
$$\Sigma_{r_2r_2} = \sigma^2_{R(X)} + \Sigma_{r_2r_1} B_{r_1r_2} = \sigma^2_{R(X)} + 0 = \sigma^2_{R(X)}$$
$$\ldots$$

$$\begin{array}{c} \\ R(X) \end{array}
\overset{R(X)}{\begin{pmatrix} \sigma^2_{R(X)} & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & \sigma^2_{R(X)} \end{pmatrix}}
= \begin{pmatrix} 4 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & 4 \end{pmatrix} \longrightarrow$$

on−diagonal: $\sigma^2_{R(X)}$

off−diagonal: 0

$$\Sigma_{11} \leftarrow \sigma_1^2$$
$$\Sigma_{ij} \leftarrow \Sigma_{ii} B_{ij}$$
$$\Sigma_{ji} \leftarrow \Sigma_{ij}^T$$
$$\Sigma_{jj} \leftarrow \sigma_j^2 + \Sigma_{ji} B_{ij}$$

Lifted B'

$$\begin{pmatrix} 0 & \beta_{rs} & 0 \\ 0 & 0 & \beta_{st} \\ 0 & 0 & 0 \end{pmatrix}$$

Lifting the Full Joint: Simple Case

$$\begin{pmatrix} \sigma^2_{R(X)} & \cdots & 0 & \Sigma_{r_1 s_1} & \cdots & \Sigma_{r_1 s_m} & \Sigma_{r_1 t_1} & \cdots & \Sigma_{r_1 t_l} \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ 0 & \cdots & \sigma^2_{R(X)} & \Sigma_{r_n s_1} & \cdots & \Sigma_{r_n s_m} & \Sigma_{r_n t_1} & \cdots & \Sigma_{r_n t_l} \\ \Sigma_{s_1 r_1} & \cdots & \Sigma_{s_1 r_n} & \Sigma_{s_1 s_1} & \cdots & \Sigma_{s_1 s_m} & \Sigma_{s_1 t_1} & \cdots & \Sigma_{s_1 t_l} \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ \Sigma_{s_m r_1} & \cdots & \Sigma_{s_m r_n} & \Sigma_{s_m s} & \cdots & \Sigma_{s_m s_m} & \Sigma_{s_m t_1} & \cdots & \Sigma_{s_m t_n} \\ \Sigma_{t_1 r_1} & \cdots & \Sigma_{t_1 r_n} & \Sigma_{t_1 s_1} & \cdots & \Sigma_{t_1 s_m} & \Sigma_{t_1 t_1} & \cdots & \Sigma_{t_1 t_l} \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ \Sigma_{t_l r_1} & \cdots & \Sigma_{t_l r_n} & \Sigma_{t_l s_1} & \cdots & \Sigma_{t_l s_m} & \Sigma_{t_l t_1} & \cdots & \Sigma_{t_l t_l} \end{pmatrix}$$

$$\Sigma_{(r_1 \ldots r_n)s_1}$$
$$= \Sigma_{(r_1 \ldots r_n)(r_1 \ldots r_n)} B_{(r_1 \ldots r_n)s_1}$$
$$= \begin{pmatrix} \sigma^2_{R(X)} & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & \sigma^2_{R(X)} \end{pmatrix} \begin{pmatrix} \beta_{rs} \\ \vdots \\ \beta_{rs} \end{pmatrix} = \begin{pmatrix} \sigma^2_{R(X)}\beta_{rs} \\ \vdots \\ \sigma^2_{R(X)}\beta_{rs} \end{pmatrix} = \begin{pmatrix} 4 \cdot 0.5 \\ \vdots \\ 4 \cdot 0.5 \end{pmatrix} = \begin{pmatrix} 2 \\ \vdots \\ 2 \end{pmatrix}$$

$$\Sigma_{s_1 s_1}$$
$$= \sigma^2_{S(Y)} + \Sigma_{s_1 (r_1 \ldots r_n)} B_{(r_1 \ldots r_n)s_1}$$
$$= \sigma^2_{S(Y)} + \begin{pmatrix} \sigma^2_{R(X)}\beta_{rs} & \ldots & \sigma^2_{R(X)}\beta_{rs} \end{pmatrix} \begin{pmatrix} \beta_{rs} \\ \vdots \\ \beta_{rs} \end{pmatrix} = \sigma^2_{S(Y)} + n\sigma^2_{R(X)}\beta^2_{rs}$$
$$= 4 + n \cdot 4 \cdot 0.5^2 = 4 + n$$

$$\Sigma_{11} \leftarrow \sigma_1^2$$
$$\Sigma_{ij} \leftarrow \Sigma_{ii} B_{ij}$$
$$\Sigma_{ji} \leftarrow \Sigma_{ij}^T$$
$$\Sigma_{jj} \leftarrow \sigma_j^2 + \Sigma_{ji} B_{ij}$$

Lifted B'

$$\begin{pmatrix} 0 & \beta_{rs} & 0 \\ 0 & 0 & \beta_{st} \\ 0 & 0 & 0 \end{pmatrix}$$

Lifting the Full Joint: Simple Case

$$\begin{pmatrix} \sigma^2_{R(X)} & \cdots & 0 & \Sigma_{r_1s_1} & \cdots & \Sigma_{r_1s_m} & \Sigma_{r_1t_1} & \cdots & \Sigma_{r_1t_l} \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ 0 & \cdots & \sigma^2_{R(X)} & \Sigma_{r_ns_1} & \cdots & \Sigma_{r_ns_m} & \Sigma_{r_nt_1} & \cdots & \Sigma_{r_nt_l} \\ \Sigma_{s_1r_1} & \cdots & \Sigma_{s_1r_n} & \Sigma_{s_1s_1} & \cdots & \Sigma_{s_1s_m} & \Sigma_{s_1t_1} & \cdots & \Sigma_{s_1t_l} \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ \Sigma_{s_mr_1} & \cdots & \Sigma_{s_mr_n} & \Sigma_{s_ms} & \cdots & \Sigma_{s_ms_m} & \Sigma_{s_mt_1} & \cdots & \Sigma_{s_mt_n} \\ \Sigma_{t_1r_1} & \cdots & \Sigma_{t_1r_n} & \Sigma_{t_1s_1} & \cdots & \Sigma_{t_1s_m} & \Sigma_{t_1t_1} & \cdots & \Sigma_{t_1t_l} \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ \Sigma_{t_lr_1} & \cdots & \Sigma_{t_lr_n} & \Sigma_{t_ls_1} & \cdots & \Sigma_{t_ls_m} & \Sigma_{t_lt_1} & \cdots & \Sigma_{t_lt_l} \end{pmatrix}$$

$\Sigma_{s_1s_1}$

$$\Sigma_{(r_1\ldots r_n s_1)s_2} = \Sigma_{(r_1\ldots r_n s_1)(r_1\ldots r_n s_1)} B_{(r_1\ldots r_n s_1)s_2}$$

$$= \begin{pmatrix} \sigma^2_{R(X)} & \cdots & 0 & \sigma^2_{R(X)}\beta_{rs} \\ \vdots & \ddots & \vdots & \vdots \\ 0 & \cdots & \sigma^2_{R(X)} & \sigma^2_{R(X)}\beta_{rs} \\ \sigma^2_{R(X)}\beta_{rs} & \cdots & \sigma^2_{R(X)}\beta_{rs} & \sigma^2_{S(Y)} + n\sigma^2_{R(X)}\beta^2_{rs} \end{pmatrix} \begin{pmatrix} \beta_{rs} \\ \vdots \\ \beta_{rs} \\ 0 \end{pmatrix}$$

$$= \begin{pmatrix} \sigma^2_{R(X)}\beta_{rs} \\ \vdots \\ \sigma^2_{R(X)}\beta_{rs} \\ n\sigma^2_{R(X)}\beta^2_{rs} \end{pmatrix} = \begin{pmatrix} 4 \cdot 0.5 \\ \vdots \\ 4 \cdot 0.5 \\ n \cdot 4 \cdot 0.5^2 \end{pmatrix} = \begin{pmatrix} 2 \\ \vdots \\ 2 \\ n \end{pmatrix}$$

$$\Sigma_{s_2s_2}$$
$$= \sigma^2_{S(Y)} + \Sigma_{s_2(r_1\ldots r_n s_1)} B_{(r_1\ldots r_n s_1)s_2}$$
$$= \sigma^2_{S(Y)} + \begin{pmatrix} \sigma^2_{R(X)}\beta_{rs} & \ldots & \sigma^2_{R(X)}\beta_{rs} & n\sigma^2_{R(X)}\beta^2_{rs} \end{pmatrix} \begin{pmatrix} \beta_{rs} \\ \vdots \\ \beta_{rs} \\ 0 \end{pmatrix}$$
$$= \sigma^2_{S(Y)} + n\sigma^2_{R(X)}\beta^2_{rs} = 4 + n$$

$$\Sigma_{11} \leftarrow \sigma_1^2$$
$$\Sigma_{ij} \leftarrow \Sigma_{ii} B_{ij}$$
$$\Sigma_{ji} \leftarrow \Sigma_{ij}^T$$
$$\Sigma_{jj} \leftarrow \sigma_j^2 + \Sigma_{ji} B_{ij}$$

Lifted B'

$$\begin{pmatrix} 0 & \beta_{rs} & 0 \\ 0 & 0 & \beta_{st} \\ 0 & 0 & 0 \end{pmatrix}$$

$$\begin{matrix} \sigma^2_{R(X)} & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & \sigma^2_{R(X)} \end{matrix} \longrightarrow \begin{matrix} \text{on-diagonal: } \sigma^2_{R(X)} \\ \text{off-diagonal: } 0 \end{matrix}$$

$$\begin{matrix} \sigma^2_{R(X)}\beta_{rs} & \cdots & \sigma^2_{R(X)}\beta_{rs} \\ \vdots & \ddots & \vdots \\ \sigma^2_{R(X)}\beta_{rs} & \cdots & \sigma^2_{R(X)}\beta_{rs} \end{matrix} \longrightarrow \text{all: } \sigma^2_{R(X)}\beta_{rs}$$

$$\begin{matrix} \sigma^2_{S(Y)} + n\sigma^2_{R(X)}\beta^2_{rs} & \cdots & n\sigma^2_{R(X)}\beta^2_{rs} \\ \vdots & \ddots & \vdots \\ n\sigma^2_{R(X)}\beta^2_{rs} & \cdots & \sigma^2_{S(Y)} + n\sigma^2_{R(X)}\beta^2_{rs} \end{matrix} \longrightarrow \begin{matrix} \text{on-diagonal: } \sigma^2_{S(Y)} + n\sigma^2_{R(X)}\beta^2_{rs} \\ \text{off-diagonal: } n\sigma^2_{R(X)}\beta^2_{rs} \end{matrix}$$

$$\begin{matrix} & R(X) & S(Y) & T(Z) \end{matrix}$$

$$\begin{matrix} R(X) \\ \\ S(Y) \\ \\ T(Z) \end{matrix} \begin{pmatrix} \Sigma_{r_1r_1} & \cdots & \Sigma_{r_1r_n} & \Sigma_{r_1s_1} & \cdots & \Sigma_{r_1s_m} & \Sigma_{r_1t_1} & \cdots & \Sigma_{r_1t_l} \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ \Sigma_{r_nr_1} & \cdots & \Sigma_{r_nr_n} & \Sigma_{r_ns_1} & \cdots & \Sigma_{r_ns_m} & \Sigma_{r_nt_1} & \cdots & \Sigma_{r_nt_l} \\ \Sigma_{s_1r_1} & \cdots & \Sigma_{s_1r_n} & \Sigma_{s_1s_1} & \cdots & \Sigma_{s_1s_m} & \Sigma_{s_1t_1} & \cdots & \Sigma_{s_1t_l} \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ \Sigma_{s_mr_1} & \cdots & \Sigma_{s_mr_n} & \Sigma_{s_ms} & \cdots & \Sigma_{s_ms_m} & \Sigma_{s_mt_1} & \cdots & \Sigma_{s_mt_n} \\ \Sigma_{t_1r_1} & \cdots & \Sigma_{t_1r_n} & \Sigma_{t_1s_1} & \cdots & \Sigma_{t_1s_m} & \Sigma_{t_1t_1} & \cdots & \Sigma_{t_1t_l} \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ \Sigma_{t_lr_1} & \cdots & \Sigma_{t_lr_n} & \Sigma_{t_ls_1} & \cdots & \Sigma_{t_ls_m} & \Sigma_{t_lt_1} & \cdots & \Sigma_{t_lt_l} \end{pmatrix}$$

Lifting the Full Joint: Simple Case

$$
\begin{array}{c|ccc}
 & & T(Z) & \\
R(X) & m\sigma^2_{R(X)}\beta_{rs}\beta_{st} & \cdots & m\sigma^2_{R(X)}\beta_{rs}\beta_{st} \\
 & \vdots & \ddots & \vdots \\
 & m\sigma^2_{R(X)}\beta_{rs}\beta_{rs} & \cdots & m\sigma^2_{R(X)}\beta_{rs}\beta_{rs}
\end{array}
\longrightarrow \text{all: } m\sigma^2_{R(X)}\beta_{rs}\beta_{st} = m \cdot 4 \cdot 0.5 \cdot (-1) = -2m
$$

$$
\begin{array}{c|ccc}
 & & T(Z) & \\
S(Y) & (\sigma^2_{S(Y)} + mn\sigma^2_{R(X)}\beta^2_{rs})\beta_{st} & \cdots & (\sigma^2_{S(Y)} + mn\sigma^2_{R(X)}\beta^2_{rs})\beta_{st} \\
 & \vdots & \ddots & \vdots \\
 & (\sigma^2_{S(Y)} + mn\sigma^2_{R(X)}\beta^2_{rs})\beta_{st} & \cdots & (\sigma^2_{S(Y)} + mn\sigma^2_{R(X)}\beta^2_{rs})\beta_{st}
\end{array}
\longrightarrow \text{all: } (\sigma^2_{S(Y)} + mn\sigma^2_{R(X)}\beta^2_{rs})\beta_{st} = -4 - mn
$$

$$
\begin{array}{c|ccc}
 & & T(Z) & \\
T(Z) & \sigma^2_{T(Z)} + m\beta^2_{st}(\sigma^2_{S(Y)} + mn\sigma^2_{R(X)}\beta^2_{rs}) & \cdots & m\beta^2_{st}(\sigma^2_{S(Y)} + mn\sigma^2_{R(X)}\beta^2_{rs}) \\
 & \vdots & \ddots & \vdots \\
 & m\beta^2_{st}(\sigma^2_{S(Y)} + mn\sigma^2_{R(X)}\beta^2_{rs}) & \cdots & \sigma^2_{T(Z)} + m\beta^2_{st}(\sigma^2_{S(Y)} + mn\sigma^2_{R(X)}\beta^2_{rs})
\end{array}
$$

$\longrightarrow$ on–diagonal: $\sigma^2_{T(Z)} + m\beta^2_{st}(\sigma^2_{S(Y)} + mn\sigma^2_{R(X)}\beta^2_{rs}) = 3 + 4m + m^2n$

off–diagonal: $m\beta^2_{st}(\sigma^2_{S(Y)} + mn\sigma^2_{R(X)}\beta^2_{rs}) = 4m + m^2n$

Lifted Joint

- Only two structures required for covariance matrix
  - A matrix

$$
\begin{array}{c} \\ R(X) \\ S(Y) \\ T(Z) \end{array}
\begin{array}{c}
\begin{array}{ccc} R(X) & S(Y) & T(Z) \end{array} \\
\begin{pmatrix}
0 & \sigma^2_{R(X)}\beta_{rs} & m\sigma^2_{R(X)}\beta_{rs}\beta_{st} \\
\sigma^2_{R(X)}\beta_{rs} & n\sigma^2_{R(X)}\beta^2_{rs} & \left(\sigma^2_{S(Y)} + mn\sigma^2_{R(X)}\beta^2_{rs}\right)\beta_{st} \\
m\sigma^2_{R(X)}\beta_{rs}\beta_{st} & \left(\sigma^2_{S(Y)} + mn\sigma^2_{R(X)}\beta^2_{rs}\right)\beta_{st} & m\beta^2_{st}\left(\sigma^2_{S(Y)} + mn\sigma^2_{R(X)}\beta^2_{rs}\right)
\end{pmatrix}
\end{array}
$$

$$
= \begin{pmatrix}
4 & 2 & -2m \\
2 & n & -4 - mn \\
-2m & -4 - mn & 4m + m^2 n
\end{pmatrix}
$$

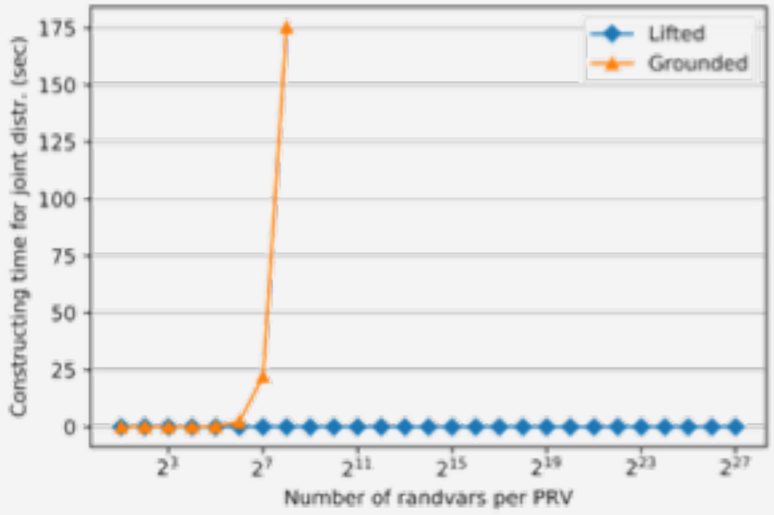
  - A vector for on-diagonal covariance entries
    - Individual variances
      - Have to be stored anyway

$$
\begin{array}{c} R(X) \\ S(Y) \\ T(Z) \end{array}
\begin{pmatrix} \sigma^2_{R(X)} \\ \sigma^2_{S(Y)} \\ \sigma^2_{T(Z)} \end{pmatrix}
= \begin{pmatrix} 4 \\ 4 \\ 3 \end{pmatrix}
$$

Mattis Hartwig and Ralf Möller. Lifted Query Answering in Gaussian Bayesian Networks. In: *PGM-20 Proceedings of the 10th International Conference on Probabilistic Graphical Models*, 2020.

Lifted Joint

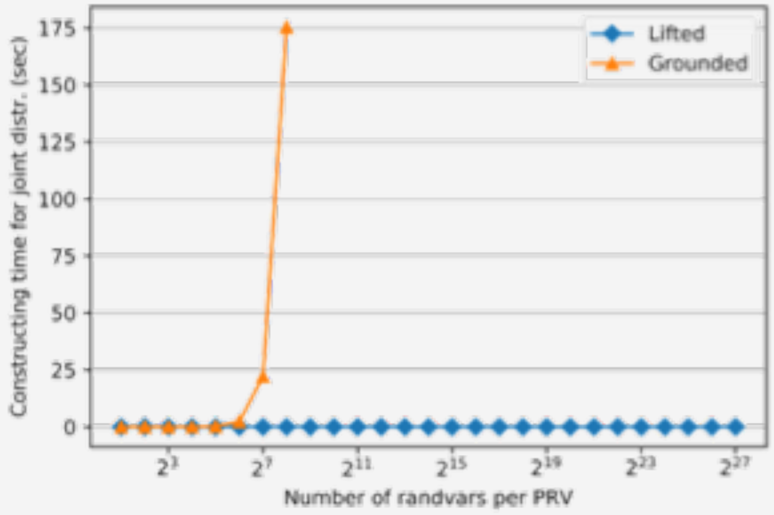
- Only two structures required for covariance matrix
- Depend only on the number of PRVs, not the domain sizes!



Mattis Hartwig and Ralf Möller. Lifted Query Answering in Gaussian Bayesian Networks. In: *PGM-20 Proceedings of the 10th International Conference on Probabilistic Graphical Models*, 2020.

Lifted Query Answering

- Marginal queries: Read off values in (lifted) covariance representation
- Conditional queries $\boldsymbol{R}|\boldsymbol{E} = \boldsymbol{e} \sim \mathcal{N}(\mu^*, \Sigma^*)$
  - $\mu^* \ = \mu_{\boldsymbol{R}} + \Sigma_{\boldsymbol{RE}}\Sigma_{\boldsymbol{EE}}^{-1}(\boldsymbol{e} - \mu_{\boldsymbol{E}})$
  - $\Sigma^* \ = \Sigma_{\boldsymbol{RR}} - \Sigma_{\boldsymbol{RE}}\Sigma_{\boldsymbol{EE}}^{-1}\Sigma_{\boldsymbol{ER}}$
  - Matrix multiplication, inversion required
    - Possible to compute in a lifted manner due to block structure
      - Proof in paper by Hartwig and Möller (2020)
  - Evidence is ground
    - Probably no symmetries in observations with real numbers as range values → unlikely to get identical observations
      - Fig.: 50% of ground instances get random values assigned as evidence



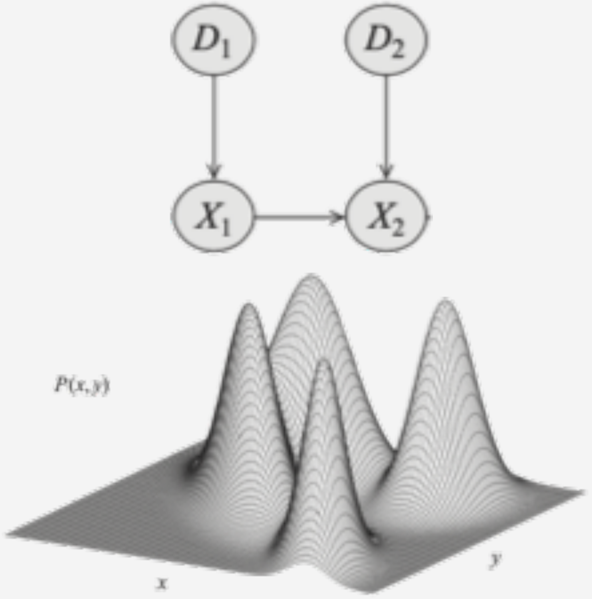
Mattis Hartwig and Ralf Möller. Lifted Query Answering in Gaussian Bayesian Networks. In: *PGM-20 Proceedings of the 10th International Conference on Probabilistic Graphical Models*, 2020.

Interim Summary

- Linear Gaussian models
  - Linear dependency between child and parent random variables
  - Full joint given by vector of means and covariance matrix
    - Information form as inverse of covariance form
  - Query answering
    - Marginal using covariance matrix
    - Conditional using information form
- Gaussian BNs
  - Explicitly encode independencies in network structure
    - Conditional linear Gaussian
  - GBN = multivariate Gaussian distribution
  - Lifting for PRVs without an overlap in logvars between parent and child

Hybrid Models

- Models that contain discrete (D_i in fig.) and continuous random variables (X_i in fig.)
- Some general results
  - Even representing the correct marginal distribution in a hybrid network can require space that is exponential in the size of the network
  - Query answering problem is NP-hard even if the GBN is a polytree where all discrete random variables are Boolean-valued and where every continuous random variable has at most one discrete ancestor
    - There are not even approximate algorithms to solve the problem in polynomial time with a useful error bound without further restrictions



Joint marginal of X_1, X_2

Figures from PGM book by Koller and Friedman, p. 615+616.

Outline: 8. Continuous Space

A. *Basics*
- Continuous variables, probability density function, cumulative probability distribution
- Joint distribution, marginal density, conditional density

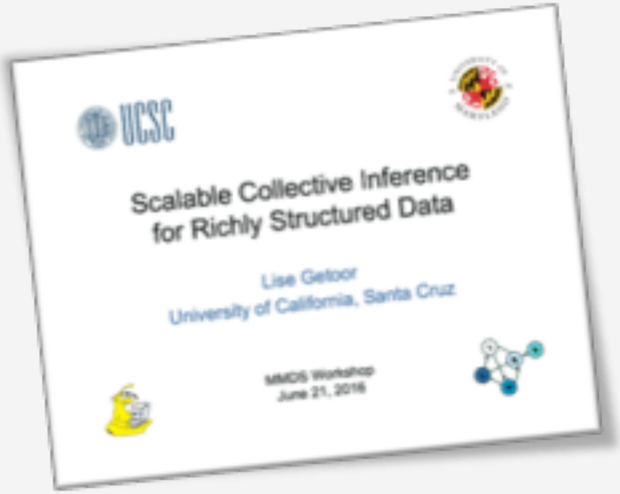
B. *Gaussian models*
- (Multivariate) Gaussian distribution
- (Parameterised) Gaussian Bayesian networks

C. ***Probabilistic Soft Logic (PSL)***
- Modelling, semantics, inference task

Probabilistic Soft Logic (PSL)

- Logic-based approach
- Probabilistic programming language
  - Predicate = relationship or property
  - Atom = continuous random variable
  - Rule = dependency or constraint
  - Set = define aggregates
- PSL program = rules + input database
- Implementation: https://psl.linqs.org

Based on slides by Lise Getoor, “Scalable Collective Inference for Richly Structured Data”, MMDS Workshop 2016.



Syntax & Semantics

- Let $\boldsymbol{R}$ be a set of weighted logical rules, each R_j has the form

$$w_j : \bigwedge_{i \in I_j^-} x_i \Rightarrow \bigvee_{i \in I_j^+} x_i$$

  - $w_j \geq 0$
  - Sets I_j^-, I_j^+ index conjuncted / disjuncted literals
  - Equivalent clausal form:

$$\left(\bigvee_{i \in I_j^+} x_i\right) \vee \left(\bigvee_{i \in I_j^-} \neg x_i\right)$$

- Probability distribution (compare: MLNs)

$$P(\boldsymbol{x}) \propto \exp\left(\sum_{R_j \in \boldsymbol{R}} w_j \left(\bigvee_{i \in I_j^+} x_i\right) \vee \left(\bigvee_{i \in I_j^-} \neg x_i\right)\right)$$

MPE Inference

- MPE: Find the most probable assignment to the unobserved random variables
  - I.e., given a model ground over an input database,

$$\underset{\boldsymbol{x}}{\operatorname{argmax}} \sum_{R_j \in \boldsymbol{R}} w_j \left(\bigvee_{i \in I_j^+} x_i \right) \vee \left(\bigvee_{i \in I_j^-} \neg x_i \right)$$

  - Combinatorial, NP-hard

- Approximation:
  View as optimising rounding probabilities

Expected Score

- Expected score of a clause is the weight times the probability that at least one literal is true:

$$w_j \left(1 - \prod_{i \in I_j^+} (1 - p_i) \prod_{i \in I_j^-} p_i \right)$$

- Then, expected total score is

$$\widehat{W} = \sum_{R_j \in \boldsymbol{R}} w_j \left(1 - \prod_{i \in I_j^+} (1 - p_i) \prod_{i \in I_j^-} p_i \right)$$

- But, $\operatorname{argmax}_{\boldsymbol{p}} \widehat{W}$ highly non-convex due to product

At least one literal true
→ or-semantics → trick:
Instead of computing
$P(A \vee B)$
$= P(A) + P(B) - P(A \wedge B)$
compute
$P\big(\neg\neg(A \vee B)\big)$
$= 1 - P(\neg A \wedge \neg B)$

Approximate Inference

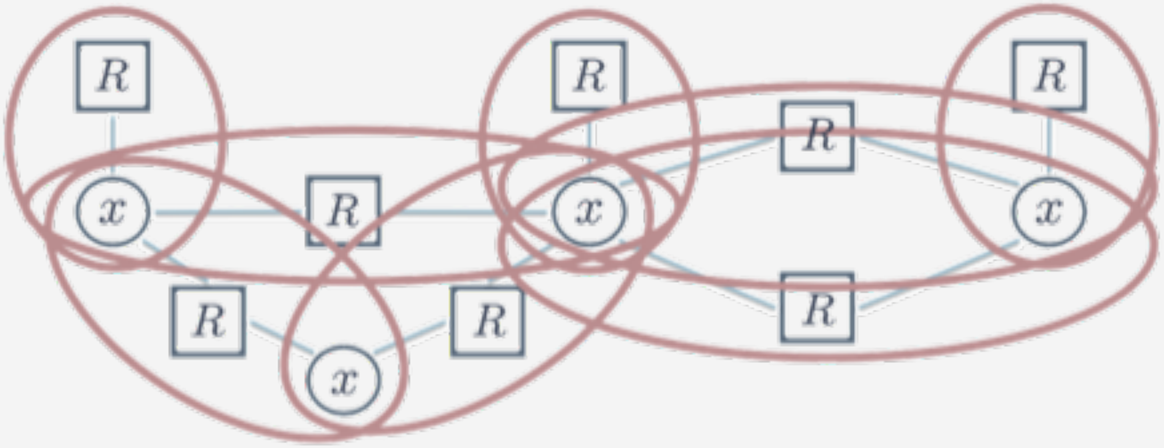
- Instead: Optimise a linear program that bounds expected score

$$\sum_{R_j \in \boldsymbol{R}} w_j \left(1 - \prod_{i \in I_j^+} (1 - p_i) \prod_{i \in I_j^-} p_i \right) \geq \left(1 - \frac{1}{e}\right) \sum_{R_j \in \boldsymbol{R}} w_j \min \left\{ \sum_{i \in I_j^+} p_i + \sum_{i \in I_j^-} (1 - p_i)\, , 1 \right\}$$

  - Can give $\left(1 - \frac{1}{e}\right)$-optimal *discrete* solution

Scalable Approximate Inference

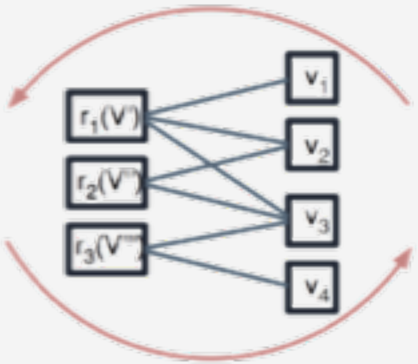
- Linear programming algorithms do not scale well to big probabilistic models



- Instead of solving the problem as one big optimisation, decompose the problem based on its graphical structure
  - Compare: cliques/clusters

Consensus Optimisation

- Decompose problem and solve sub-problems independently (in parallel), then merge results
  - Sub-problems are ground rules
  - Auxiliary variables enforce consensus across sub-problems
- Framework:
  Alternating direction method of multipliers (ADMM) (Boyd, 2011)
  - Guaranteed to converge for convex problems
  - Inference with ADMM fast, scalable, straightforward to implement (Bach et al, 2017)

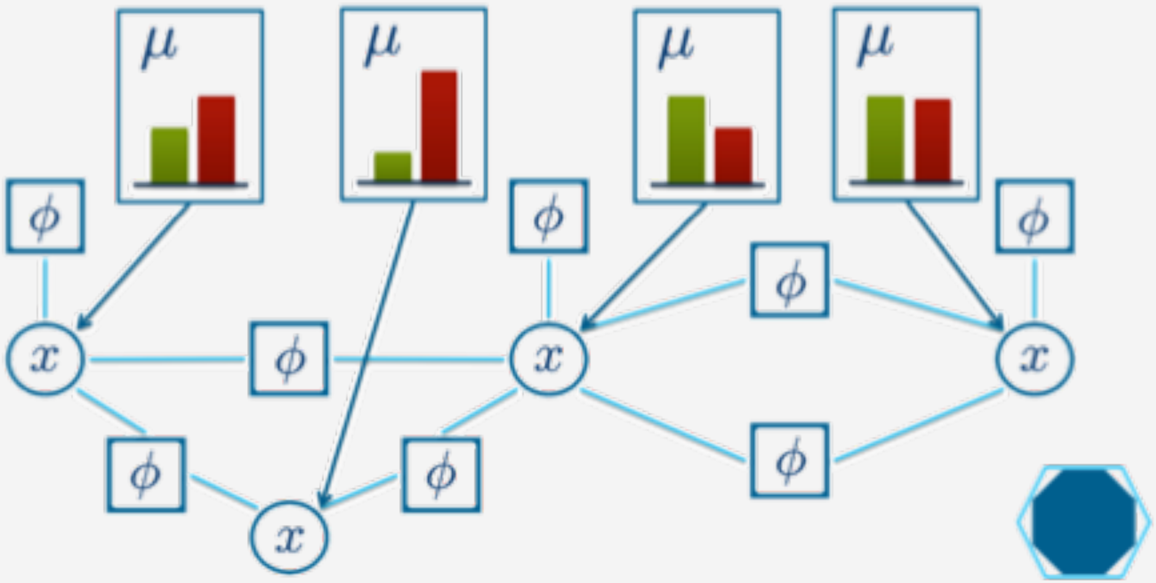


S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein. Distributed Optimization and Statistical Learning Via the Alternating Direction Method of Multipliers. *Now Publishers*, 2011.
Stephan H. Bach. Matthias Broecheler, Bert Huang, and Lise Getoor: Hinge-Loss Markov Random Fields and Probabilistic Soft Logic. In: *Journal of Machine Learning Research 18*, 2017.

Local Consistency Relaxation

- Relax search over consistent marginals to simpler set

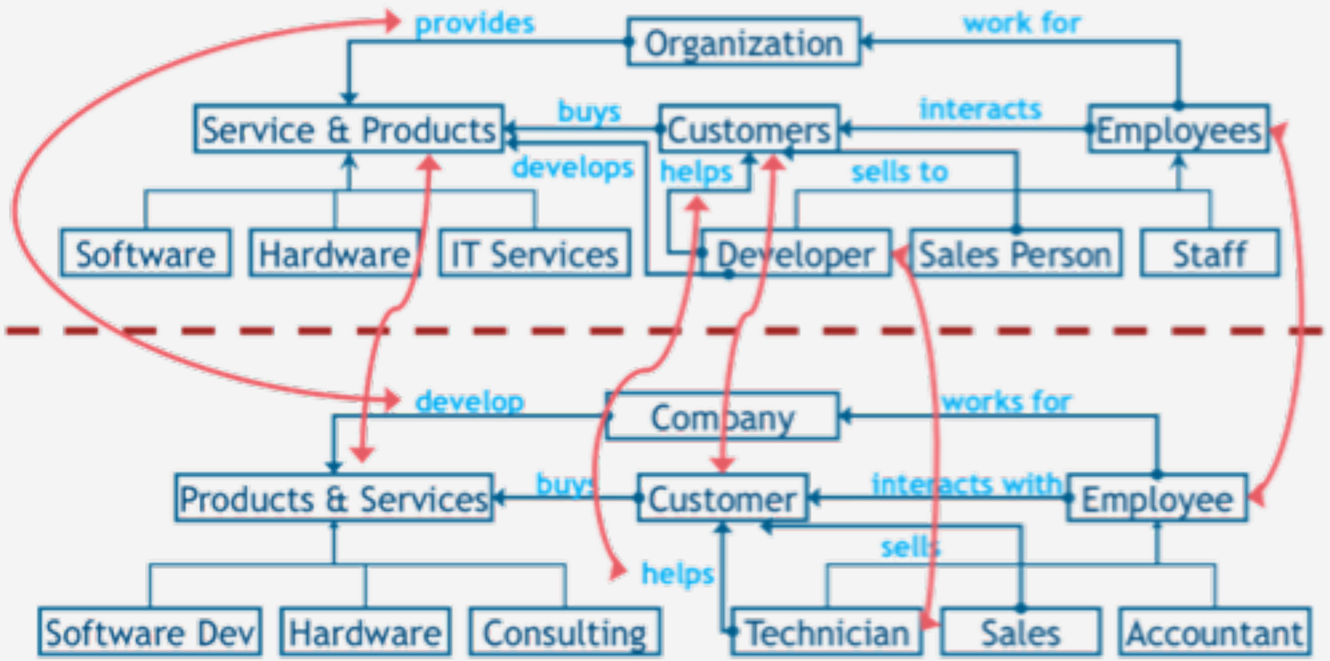
$$\underset{\boldsymbol{\mu}\in[0,1]^n}{\operatorname{argmax}} \sum_{R_j\in\boldsymbol{R}} w_j \min\left\{\sum_{i\in I_j^+} \mu_i + \sum_{i\in I_j^-} (1-\mu_i)\,,1\right\}$$



Continuous Variables & Similarity

- Continuous values interpreted as similarities
  - E.g., multiple ontologies → alignment
    - Match/Don't match → similar to what extent?

  ⇒ Soft logic



Soft Logic

- Logical operators defined for continuous values in the $[0,1]$ interval
  - Interpret as similarities or degree of truth
- Łukasiewicz logic
  - $p \land q = \max\{p + q - 1, 0\}$
  - $p \lor q = \min\{p + q, 1\}$
  - $\neg p = 1 - p$
- PSL: Use Łukasiewicz logic to interpret rules
  - Hinge-loss MNs (or Markov random fields as called in the publications by the PSL team) formalise this

Hinge-loss MNs

- Relaxed, logic-based MNs can reason about both discrete and continuous graph data scalably and accurately
  - General objective

$$\underset{\boldsymbol{y}\in[0,1]^n}{\operatorname{argmax}} P(\boldsymbol{y})$$
$$= \underset{\boldsymbol{y}\in[0,1]^n}{\operatorname{argmax}} \sum_{j=1}^{m} w_j \min\left\{\sum_{i\in I_j^+} y_i + \sum_{i\in I_j^-}(1-y_i), 1\right\}$$
$$= \underset{\boldsymbol{y}\in[0,1]^n}{\operatorname{argmin}} \sum_{j=1}^{m} w_j \max\left\{1 - \sum_{i\in I_j^+} y_i - \sum_{i\in I_j^-}(1-y_i), 0\right\}$$

  - Notion of distance to satisfaction

Stephen H. Bach. Matthias Broecheler, Bert Huang, and Lise Getoor: Hinge-Loss Markov Random Fields and Probabilistic Soft Logic. In: *Journal of Machine Learning Research 18*, 2017.

Distance to Satisfaction

$$\underset{y\in[0,1]^n}{\text{argmin}} \sum_{j=1}^{m} w_j \max\left\{1 - \sum_{i\in I_j^+} y_i - \sum_{i\in I_j^-} (1 - y_i), 0\right\}$$

- Maximum value of any unweighted term is 1
  - Term is *satisfied*
- Unsatisfied term → distance to satisfaction
  - How far it is from achieving its maximum value
  - Each unweighted objective term measures how far the linear constraint is away from being satisfied:

$$1 - \sum_{i\in I_j^+} y_i - \sum_{i\in I_j^-} (1 - y_i) \leq 0$$

Stephen H. Bach. Matthias Broecheler, Bert Huang, and Lise Getoor: Hinge-Loss Markov Random Fields and Probabilistic Soft Logic. In: *Journal of Machine Learning Research 18*, 2017.

Relaxed Linear Constraints

- Instead of requiring logical clauses, each term can be defined using any function $\ell_j(\boldsymbol{y})$ linear in $\boldsymbol{y}$

$$\underset{\boldsymbol{y}\in[0,1]^n}{\operatorname{argmin}} \sum_{j=1}^{m} w_j \max\{\ell_j(\boldsymbol{y}), 0\}$$

  - Each term represents the distance to satisfaction of a linear constraint $\ell_j(\boldsymbol{y}) \leq 0$
    - Can use logical clauses or something else based on domain knowledge
    - Also called hinge losses
    - Sometimes $\max\{\ell_j(\boldsymbol{y}), 0\}$ gets squared to better trade off conflicting objective terms
  - Weight indicates how important it is to satisfy a constraint relative to others by scaling the distance to satisfaction

Stephen H. Bach. Matthias Broecheler, Bert Huang, and Lise Getoor: Hinge-Loss Markov Random Fields and Probabilistic Soft Logic. In: *Journal of Machine Learning Research 18*, 2017.

Hinge-loss MNs

- Let $\boldsymbol{y} = (y_1, \ldots, y_n)$ be a vector of n random variables and $\boldsymbol{x} = (x_1, \ldots, x_{n'})$ be a vector of n' random variables with joint range $\boldsymbol{D} = [0,1]^{n+n'}$
- Let $\boldsymbol{\phi} = (\phi_1, \ldots, \phi_m)$ be a vector of m continuous potentials of the form
$$\phi_j(\boldsymbol{y}, \boldsymbol{x}) = \left(\max\{\ell_j(\boldsymbol{y}, \boldsymbol{x}), 0\}\right)^{p_j}$$
    - $\ell_j(\boldsymbol{y}, \boldsymbol{x})$ linear function of $\boldsymbol{y}, \boldsymbol{x}$
    - $p_j \in \{1,2\}$
- For$(\boldsymbol{y}, \boldsymbol{x}) \in \boldsymbol{D}$ and given a vector of m weights $\boldsymbol{w} = (w_1, \ldots, w_m)$, constrained hinge-loss energy function $f_{\boldsymbol{w}}$ is defined as
$$f_{\boldsymbol{w}}(\boldsymbol{y}, \boldsymbol{x}) = \sum_{j=1}^{m} w_j \phi_j(\boldsymbol{y}, \boldsymbol{x})$$

Stephen H. Bach. Matthias Broecheler, Bert Huang, and Lise Getoor: Hinge-Loss Markov Random Fields and Probabilistic Soft Logic. In: *Journal of Machine Learning Research 18*, 2017.

Hinge-loss MNs

- Let $c = (c_1, \dots, c_r)$ be a vector of linear constraint functions which further restrict the domain $\boldsymbol{D}$ to $\boldsymbol{D}'$
- Hinge-loss MN over random variables $\boldsymbol{y}$ and conditioned on random variables $\boldsymbol{x}$ is a PDF defined as follows
  - if $(\boldsymbol{y}, \boldsymbol{x}) \notin \boldsymbol{D}'$, then $P(\boldsymbol{y}|\boldsymbol{x}) = 0$
  - if $(\boldsymbol{y}, \boldsymbol{x}) \in \boldsymbol{D}'$, then

$$P(\boldsymbol{y}|\boldsymbol{x}) = \frac{1}{Z(\boldsymbol{w}, \boldsymbol{x})} \exp\left(-f_{\boldsymbol{w}}(\boldsymbol{y}, \boldsymbol{x})\right)$$

    - where

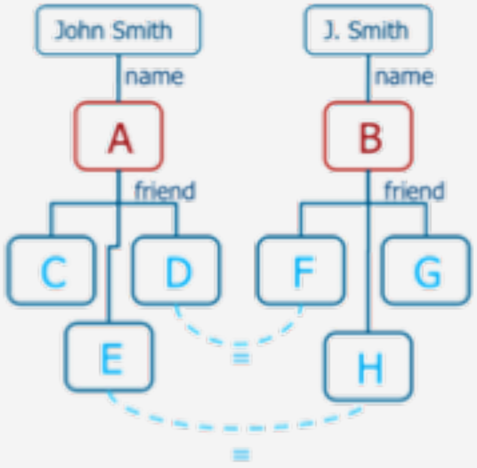
$$Z(\boldsymbol{w}, \boldsymbol{x}) = \int_{\boldsymbol{y}|(\boldsymbol{y},\boldsymbol{x}) \in \boldsymbol{D}'} \exp\left(-f_{w}(\boldsymbol{y}, \boldsymbol{x})\right) d\boldsymbol{y}$$

- Define hinge-loss MNs using PSL

Stephen H. Bach. Matthias Broecheler, Bert Huang, and Lise Getoor: Hinge-Loss Markov Random Fields and Probabilistic Soft Logic. In: *Journal of Machine Learning Research 18*, 2017.

Application: E.g., Entity Resolution

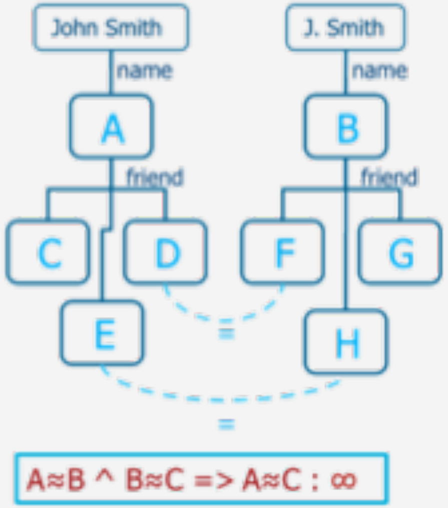
- Goal: Identify references that denote the same person
  - Use model to express dependencies

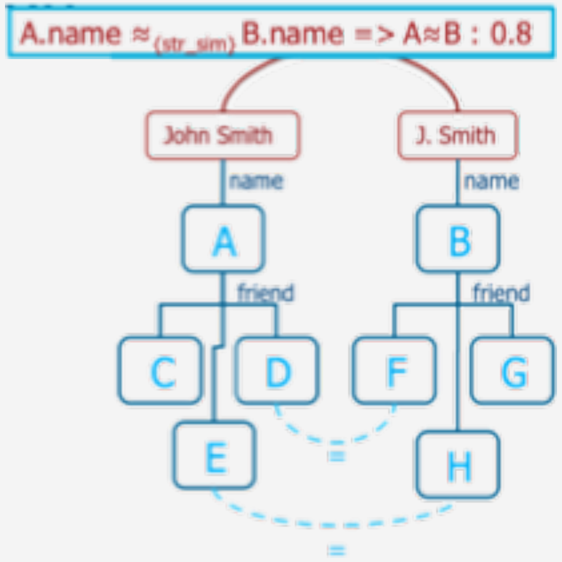


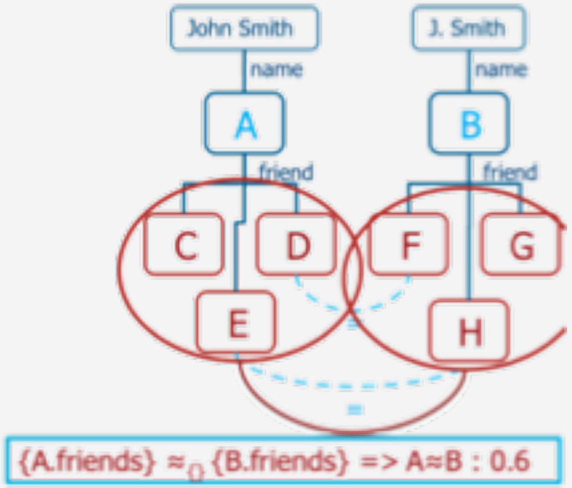
“If A=B and B=C, then A and C must also denote the same person“

“If two people have similar names, they are probably the same”

“If two people have similar friends, they are probably the same”







Interim Summary

- PSL
  - Logic programming language
  - Approximations
    - Linear program that bounds MPE solution from below
    - Decomposition of PGM to optimise set of subproblems (consensus optimisation)
    - Local consistency relaxation
  - Soft logic: Łukasiewicz logic
    - Interpret continuous values as similarities/degree of truth
- Hinge-loss MNs
  - Notion of distance to satisfaction
  - Define using PSL

Outline: 8. Continuous Space

A. *Basics*
- Continuous variables, probability density function, cumulative probability distribution
- Joint distribution, marginal density, conditional density

B. *Gaussian models*
- (Multivariate) Gaussian distribution
- (Parameterised) Gaussian Bayesian networks

C. *Probabilistic Soft Logic (PSL)*
- Modelling, semantics, inference task

The End