Overview

- Weak Supervision Learning
- Simple Estimation trick
- Correlation → Inverse covariance & Graph.

Given $x^{(1)}, x^{(2)}, \dots x^{(n)} \in \mathbb{R}^d$ (Data points)

$\lambda_1, \dots \lambda_n : \mathbb{R}^d \to \{-1, 1\}$

Find $P(y^{(i)} | \lambda, x^{(i)})$ $\quad y \in \{-1,1\}$

<u>Idea</u> : λ_i is a noisy voter / function (inaccurate)

Ex: λ_1 : "The classfier says yes"

λ_3 : " Name in DB "

Programmed labels

Model 0: Independent Guess

w/ Prob P_i $\quad \lambda_i(x) = y$

$\lambda_i(x)$

Sadly we don't see y.

$$P(\lambda_i(x)=1 | y=1) = P(\lambda_i(x) = -1 | y = -1) = P_i$$

Given	λ_1	λ_2	λ_3	...	λ_n	Unobserved
Data	1	1	1		1	1
$x^{(i)}$	1	-1	1		1	1
$x^{(2)}$		⋮				
⋮						
$x^{(n)}$	-1	-1	-1		-1	-1

$P(y | \lambda, x)$

① $\mathbb{E}[\lambda_i Y] = P_i \cdot 1 + (1-P_i)(-1) = 2P_i - 1 \triangleq a_i$

$\{-1,1\}$ $\qquad a_i \in [-1, 1]$

② $\mathbb{E}[\lambda_i \lambda_j] = 1$ if $i = j$ $\quad (\mathbb{E}[\lambda_i^2] = 1)$

③ $\mathbb{E}[\lambda_i \lambda_j]$ if $i \neq j$

$$= \underbrace{P_i P_j}_{\text{Agree}} \cdot 1 + \underbrace{(1-P_i)(1-P_j)}_{\text{Both wrong}} 1$$
$$+ P_i (1-P_j)(-1) + (1-P_i)P_j(-1)$$

$$= a_i a_j$$

For a matrix $M \in \mathbb{R}^{m \times m}$ $\quad M_{ij} = \mathbb{E}[\lambda_i \lambda_j]$

We can estimate M from data (observation) without y.

"Agree v.s. Disagrees" w/o y.

Simple Algorithm

$$M_{ij} M_{jk} = a_i a_j^2 a_k$$

$$\frac{M_{ij} M_{jk}}{M_{ik}} = a_j^2 \qquad \text{solve upto the sign of } \underline{a}$$

We know magnitude, not sign.

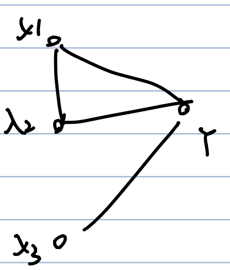
M_{ij} is observed, $\quad M_{ij} = a_i a_j$ if we know $\text{sign}(a_i)$

$\text{sign}(M_{ij}) = \text{sign}(a_i)\,\text{sign}(a_j) \Rightarrow \text{sign}(a_j)$

- what if $M_{ij} = 0 \Rightarrow a_i = 0$ or $a_j = 0$

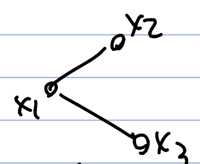
<u>Recap</u>: Simple solution to "EM-like"

What if correlated



$$\mathbb{E}[\lambda_1 \lambda_2 | Y] = \mathbb{E}[\lambda_1 | Y]\,\mathbb{E}[\lambda_2 | Y]$$

Structure of Inverse Covariance Matrices



$X_2 \sim N(X_1, 1)$

$X_3 \sim N(X_1, 1)$

$X_1 \sim N(0,1)$

$X_2 = X_1 + \varepsilon_2 \qquad \varepsilon_2 \sim N(0,1)$

$X_3 = X_1 + \varepsilon_3 \qquad \varepsilon_3 \sim N(0,1)$

① $\mathbb{E}[X_1] = 0 \qquad \mathbb{E}[X_2] = \mathbb{E}[X_1] + \mathbb{E}[\varepsilon_2] = 0$

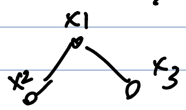
$\qquad\qquad\qquad \mathbb{E}[X_3] = 0$

② $\mathbb{E}[X_1^2] = 1. \qquad \mathbb{E}[X_2^2] = \mathbb{E}[(X_1 + \varepsilon_2)^2] = \mathbb{E}[X_1^2] + \mathbb{E}[\varepsilon_2^2]$

$\qquad\qquad\qquad\qquad\qquad = 2 = \mathbb{E}[X_3^2]$

③ $\mathbb{E}[X_1X_2] = \mathbb{E}[X_1^2] + \cancel{\mathbb{E}[X_1\varepsilon_2]}^{\,0} = \mathbb{E}[X_1^2] = 1$

$\Sigma = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 2 & 1 \\ 1 & 1 & 2 \end{bmatrix}$



$\Sigma^{-1} = \begin{bmatrix} 3 & -1 & -1 \\ -1 & 1 & 0 \\ -1 & 0 & 1 \end{bmatrix}$

We set a probability distribution $p: \mathbb{R}^d \to [0,1]$

factorizes or agrees with a graph $G = (V, E)$

if $p(x) = c_0 \prod_{(x_1,x_2)\in E} p_e(x_i, x_j) \cdot \prod_{x_i \in V} p_v(x_i)$

constant

Gaussians

$\log P(x) = \log \exp\{x^T \Sigma^{-1} x\} \cdot c$

$= \log c_0 + \sum_{(x_1,x_2)\in E} \log P_E(x_i, x_j) + \sum_{x_i\in V} \log P_V(x_i)$

$A = \Sigma^{-1} \qquad = \sum_{ij} A_{ij} x_i x_j$

for $(i,j) \in E$, $\partial x_i \partial x_j \sum_{k,l} A_{kl} x_k x_l = A_{ij} + A_{ji}$

because A is symmetric

$\Rightarrow 2A_{ij}$

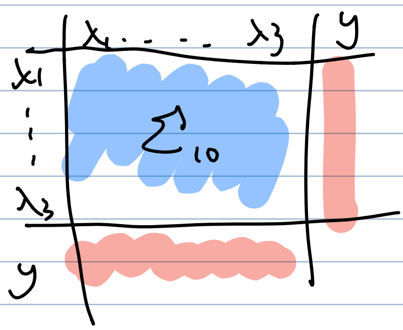
If we are differentiate factored expression.

$(i,j) \notin E \Rightarrow$ Factored term must be 0.

$\Rightarrow A_{ij} = \Sigma^{-1} = 0.$

So Gaussian have this structure. if (i,j) are not connected, $\Sigma^{-1}_{ij} = 0.$

Back to our problem



Assume we know graph structure

$\Rightarrow$ zeros in Σ^{-1}

Let $O = \{1,2,3\}^2$ observe.

$(\Sigma_1^{-1})_O = (\Sigma_O - vv^T)^{-1}$

some rank 1 fn.

$(i,j) \notin E \Rightarrow \Sigma^{-1}_{ij} = 0$

$$0 = \Sigma_0^{-1} + zz^T$$

so for every missing edge we get a Gaussian.

$$(\Sigma_0^{-1})_{ij} = B_i B_j$$

$$B_{ij}^2 = z_i^2 z_j^2 \leftrightarrow \log B_{ij}^2 = \log z_i^2 + \log z_j^2$$

Structure about inverse covariance ~~to~~ create a sequence of linear equations → solve those