Computing Lab

Song Liu (song.liu@Bristol.ac.uk)

Requirement

- To do today’s computing lab, you first need a working binary **logistic regression classifier**.

Factorization to Dependency

- Recall **a graph encoding the factorization** of a probability distribution, **encodes the dependency information** between R.V.s in such a probability distribution.

- Today, we try to discover the dependency between random variable by learning the factorization from a dataset.

US Senate

- Senate is a legislative branch of US government.
- Senators are (usually) affiliated to **Democratic** or **Republic** party.
- There are **100** senators, **2** per state.

- Our dataset contains rollcall votes of the 109^{th} US senate.
  - Rollcall: senators say “yay” or “nay” when names were called.
- **Target**:

Analyze **dependencies between senators** using rollcall data.

Graphical Modelling

- Let $x^{(i)}$ be a vote that a senator i casts on a bill.
- $x^{(i)}$ can take two values: $\{1, -1\}$.
  - "Yay": 1, "Nay": -1

- Let us model $p\left(x^{(1)} \dots x^{(100)}\right)$ as a Markov Net:
- $p\left(x^{(1)} \dots x^{(100)}\right) \propto \prod_{c \in C} g_c(x^{(c)})$
- Assume p is a **pairwise Markov network**
  - Which means, all g_c can be further factorized over edges in clique c: $g_c = \prod_{(u,v) \in c} g_{u,v}(x^{(u)}, x^{(v)})$.

Graphical Modelling

- It means $p\left(x^{(1)} \dots x^{(100)}\right) \propto \prod_{u,v \in E} g'_{u,v}\left(x^{(u)}, x^{(v)}\right)$
- For some unknown $g'_{u,v}$ function.

- Model $g'_{u,v} \coloneqq \exp\left(w_{u,v} x^{(u)} x^{(v)}\right)$ for all u, v, introducing an unknown parameter $w_{u,v}$. If $w_{u,v} = 0$, there is no edge $\left(X^{(u)}, X^{(v)}\right)$ in the factorization graph G.

- **Target**: **Estimate** $w_{u,v}$ **from data**

Intractable Normalization

- Probability mass function parameterized by $w_{u,v}$ can be written as:
- $p\left(x^{(1)} \dots x^{(100)} | \boldsymbol{W}\right) = \frac{1}{Z(w_{u,v,\forall u,v})} \exp\left(\sum_{u,v,\forall u,v} w_{u,v} x^{(u)} x^{(v)}\right)$
- $Z\left(w_{u,v,\forall u,v}\right) = \sum_{\boldsymbol{x} \in \{-1,1\}^{100}} \exp\left(\sum_{u,v,\forall u,v} w_{u,v} x^{(u)} x^{(v)}\right)$
- $Z\left(w_{u,v,\forall u,v}\right)$ is **not** computationally tractable (why?)

Conditional Graphical Modelling

- Luckily, we can use conditional graphical model to reduce the complexity of the probability mass function model:
- 1. Write down $p\left(x^{(1)} | x^{(2)} \ldots x^{(100)}, w_{1,v,\forall v}\right)$ given the joint probability model $p\left(x^{(1)} \ldots x^{(100)} | \boldsymbol{W}\right)$.
- 2. Write down the likelihood function given dataset:

$D := \{\boldsymbol{x}_1, \boldsymbol{x}_2 \ldots \boldsymbol{x}_n\}$, where $\boldsymbol{x}_i \in R^{100}$ is the votes of 100 senators on the bill i. Assume D contains IID observations.

- 3. Maximize the above likelihood w.r.t. $w_{1,v,\forall v}$
  - Hint: use your logistic regressor.

Partisanship vs. Bipartisanship

- Try to find the edges in G connected to the following vertices:

- $x^{(v)} =$ “Barack H. Obama”
- $x^{(v)} =$ “Hillary R. Clinton”
- $x^{(v)} =$ “John S. McCain”

- Who has the most bipartisan link?