Regression On Networks

Static processes

- NN Prediction
- Markov Random Fields
- Kernel-Based Methods

Interested in $\{X_i\}_{i \in V}$ on a network.

NN: $\hat{x}_i = \dfrac{\sum_{j \in N_i} x_j}{|N_i|}$ and compare to a threshold

one type of leave one out prediction.

Markov Random Fields (MRF)

Let $X \in \mathbb{R}^{N_V}$ discrete R.V.s on V

X is a MRF on G if

(a) $\mathbb{P}(X = x) > 0$

(b) $\mathbb{P}(X_i = x_i \mid X_{(-i)} = x_{(-i)}) = \mathbb{P}(X_i = x_i \mid X_{N_i} = x_{N_i})$

- Does not ensure this gives a full dist.

<u>Hammersley-Clifford Thrm</u>

MRF equivalent to a Gibbs RF.

$$\mathbb{P}(X=x) = \left(\frac{1}{K}\right) \exp\left(U(x)\right)$$

$U(\cdot)$ energy function. $K = \sum_{x} \exp\left(U(x)\right)$ partition function.

This gives a factorization over G

$$U(x) = \sum_{c \in C} U_c(x)$$

C - set of cliques of all sizes in G.

- Truncate these sums to only include only simple cliques

- Assume functional form on lower order terms

$$U_c = \begin{cases} 0 & \text{if } n_v > 2 \\ u_c & n_v \leq 2 \end{cases}$$

If X_i are binary

$$U(x) = \sum_{i=V} \alpha_i x_i + \sum_{(i,j)\in E} \beta_{ij} x_i x_j$$

$$\mathbb{P}(X_i = 1 \mid X_{N_{V_i}} = x_{N_i}) = \frac{\exp\left(\alpha_i + \sum_{N_i} \beta_{ij} x_j\right)}{1 + \exp\left(\alpha_i + \sum_{N_i} \beta_{ij} x_j\right)}$$

Very logistic-y: but way over parameterized

To reparameterized

(a) Assume $\alpha_i \equiv \alpha$, $\beta_{ij} \equiv \beta$.

$$\log\left(\frac{\mathbb{P}(X_i = 1 \mid N_i)}{\mathbb{P}(X_i = 0 \mid N_i)}\right) = \alpha + \beta \sum_{N_i} x_j$$

(b) $\alpha_i = \alpha + |N_i|\beta_2 \quad \beta_{ij} = \beta_1 - \beta_2$

$$\log\left(\frac{\mathbb{P}(X_i = 1 \mid N_i)}{\mathbb{P}(X_i = 0 \mid N_i)}\right) = \alpha + \beta_1 \sum_{N_i} x_j + \beta_2 \sum_{N_i} (1 - x_j)$$

<u>To Do</u>: Infer parameters, predict missing values

Idea: ML: $\log \mathbb{P}_\theta(X = x) = U(x;\theta) - \log k(\theta)$

Bad b.c. hard to calculate $k(\theta)$

Use pseudo-likelihood; instead optimize

$$\sum_{i\in V} \log \mathbb{P}_\theta(X_i = x_i \mid X_{N_{V_i}} = x_{N_i})$$

<u>Rmk</u>: ignores dependencies beyond neighborhood

$(\hat{\alpha}, \hat{\beta})_{MPL}$ = can be used in standard software logistic regression.

Prediction w/ MRF

- Don't observe full X, $X = (X^{obs}, X^{miss})^T$

- impute $\mathbb{P}_\theta(X^{miss} \mid X^{obs} = x^{obs})$

- Given θ, vertex-by-vertex with a Gibbs sampler

$$\mathbb{P}(x_i \mid X^{obs}, X^{miss}_{(-i)} = x^{(m-1), miss}_{(-i)})$$