10-708 L7 13/05/2020

L7 - Maximum likelihood learning of undirected GM

EX: What is key algorithm for PGMs?

Q: EM

- 10-708: see lots of algorithms; develop taste and understanding
- PGMs: do inference on unobserved; then apply completely observed tools (heuristic)
- better researchers dig out foundations

E.g. EM as co-ordinate ascent algorithm (characterising it this way can place it in a class)

EX: See rationale behind algorithm

EX: use graphical models to pull together local structures

MLE for BNs:

(*) most important: - $\theta_{ijk}^{ML} = \frac{n_{ijk}}{\sum_{i,j',k} n_{ij'k}}$ (Q: looks like counts/empirical probability)

(however ICGE GM B)

(*) due to factorisability of DGM

Q: does this apply to UGM?

MLE for undirected GMs

(*) UGM: Hammersley-Clifford means we can define a UGM in terms of a Gibbs distribution and partition function

$$P(x_1, \ldots, x_n) = \frac{1}{Z} \prod_{c \in C} \psi_c(x_c) \qquad Z = \sum_{x_1, \ldots, x_n} \prod_{c \in C} \psi_c(x_c)$$

- Z - partition fn
  - normalisation constant of product of unnormalised potentials

$\prod \psi_c(x_c)$

EX: Suppose ψ contains hidden parameters (e.g. strength of config per clique) (potentials) and you want to estimate, given complete observations

Q Can you compute parameters within ψ?
easily

Try to engage with line of thought

- A: NO; you will have the following:- $\frac{1}{Z}\prod_{c\in C}\psi_c(\underline{x}_c, \underline{z})$

- so parameters will appear inside clique potential functions
- You will need to marginalise along the lines of

$\prod_{c\in C}\sum_{\underline{z}}\psi_c(\underline{x}_c, \underline{z})$ (ψc not probability; but assume an analogous operation)

- so product-sum difficult for ML estimation

- $\psi_c(\underline{x}_c)$ X ARE
- Not compute the parameters wrong → you have written latent variables

(*) Coupling is the answer

EX: Nothing explicit to optimise against (if doing MLE); as you will have unknown parameters; and hence $\frac{1}{Z}\prod_{c\in C}\psi_c(\underline{x}_c)$ will be. be unknown

EX: some graphical models structures can also be described by UGM.

(*) log-likelihood for UGMs with tabular clique potentials.

Sufficient stats:

- UGM (V, E): the no. times config $\underline{x}$ i.e. $\underline{X}=\underline{x}$ is observed in a data set $D=\{\underline{x}_1, \ldots, \underline{x}_N\}$ can be represented as follows:-

Define

- $m(\underline{x}) = \sum_n \delta(\underline{x}, \underline{x}_n)$ (total count) (6/7)

(4/7) (4/7)

- $m(\underline{x}_c) = \sum_{?} m(\underline{x})$ (clique count) (6/7)

- Total counts - no. of time a configuration appears in dataset
- Clique counts - no. of times a particular configuration within a clique appears in the dataset

(*) Clique counts obtained by marginalising our total counts (W)(A2)

Assume discreteness

Log-likelihood:- $p(\mathcal{D}|\theta) = \sum_c \sum_{\underline{x}_c} m(\underline{x}_c) \log \psi_c(\underline{x}_c) - N \log Z$ (*)

(W)(A3): check you understand how log-like is specified (quick)

Ex: log-like:- Sum over all possible configurations of $\underline{x}$
- use delta function to dump those values of $\underline{x}/\underline{x}_n$ (?) which are consistent with your observations of the data, count 1 everytime you see it.

- connects log-likelihood with ms (i.e. counts) obtained from data

(*) Do you understand how log-like and observations/sufficient statistics for MRFs are related?

(?)

Q: What is θ (parameters?) (so you remember L3!)

(W)(A4): Refresh param. of MRFs!

- we are compute scientists, not mathematicians
- do not mattle to theoreticians (actually dealing with messiness)
- You were close → CPDs; but do not obey constraints of prob.
- An unnormalised table of nos $\psi_c(\underline{x}_c)$ - $\underline{x}_c$ associated with a no. e.g α

(*) Derivative of LL:-

- Standard calculus:

1st term $\dfrac{\partial \mathcal{L}_1}{\partial \psi_c(\underline{x}_c)} = \dfrac{m(\underline{x}_c)}{\psi_c(\underline{x}_c)}$

2nd term: (W)(A5) - Review this (quick)

Conditions on clique marginals

- Get optimal ψ_c^* → find it vanishes

(*) At ML setting of parameters, for each clique, model marginals equal to observed marginals (empirical counts)

EX: Only get marginal probability of a clique $p^*_{MLE}(x_c)$, we want the potential function (estimates of) of each clique.

- These are not the same in UGMs (✓)

(*) Only provides condition that must be satisfied when we have ML param, does not specify how to get ML param

EX: Serious work into doing this

MLE for UGMs

EX: Previous iterations relied on these concepts (decision tree style questions)

- triangulated
- clique potentials defined on maximal cliques
- full tables or compact

} - skip
- see Koller for historical account

---

2 workhorse algorithms (most insightful) ✓ nos. behind energy fg

- IPF (Iterative proportional fitting) - MRFs tabular pot.
- GIS (generalised iterative scaling) - MRFs with features potential

- EX: Key is how the differences in problem scopes yield ~~to~~ differences in algorithmic approach
  Algebraic tricks → make problem easier (significantly)

IPF

- Identity from LL optimisation → anti-climactic
- How to recover from this?

(*) From LL:

$$\frac{\partial L}{\partial \psi_c(x_c)} = \frac{m(x_c)}{\psi_c(x_c)} - N \frac{p(x_c)}{\psi_c(x_c)}$$

(W)(AB): - Review this algo.

$$p^*_{MLE}(x_c) = \frac{m(x_c)}{N} = \hat{p}(x_c)$$

(*) Derive:-

$$\frac{\tilde{p}(x_c)}{\psi_c(x_c)} = \frac{p(x_c)}{\psi_c(x_c)}$$

- Turn identity into a fixed point equation (endow identity with time component)

for ψ_c

$$\psi_c^{(t+1)} = \psi_c^{(t)}(x_c) \frac{\hat{p}(x_c)}{p^{(t)}(x_c)}$$

(*) update fn: $\frac{\hat{p}(x_c)}{p^{(t)}(x_c)}$ - proportion of empirical marginal (countable from data) over current version of estimated marginal, based on your model. (derivable from $\psi_c^{(t)}(x_c)$)

(*) In UGM, even with observed data, have to do inference

Accessible question: i) does it converge etc.?

Properties of IPF updates:

IPF is a fixed point program over time, but also over potential functions

(*) w(?): our potentials

(*) A co-ordinate ascent algorithm, attaining an optima on a partic. direction when other directions fixed.

- convergence somewhere.
- (*) Also known as I-projection (distn from one space to another where only one potential is allowed to change)

- our space of possible distn families (via attained via max-entropy)

(*) understood via KL divergence view → comes up in V.I/D.L (Jordan II)

KL divergence view

- ML can be reframed as KL divergence
- co-ordinate ascent change of IPF through KL divergence (via info theory)

$$\max \ell \iff \min KL(\hat{p}(\underline{x}) \| p(\underline{x}|\underline{\theta})) = \sum_{\underline{x}} \hat{p}(\underline{x}) \log \frac{\hat{p}(\underline{x})}{p(\underline{x}|\underline{\theta})}$$

- Partition arguments of distribution into:-

$\underline{x}_c$ and $\underline{x}_{c|}$ $\quad$ c| - complement of c

- To carve out a partic. potential clique

(*) combine & KL with conditional chain rule (2) (45): review ✓

IPF minimises KL divergence

(*) changing ψ_c (clique potential) has no effect on c.d.

(2nd term unaffected)

$$(*) \quad KL(\hat{p}(\underline{x}) \| p(\underline{x}|\underline{\theta})) = KL(\hat{p}(\underline{x}_c) \| p(\underline{x}_c|\underline{\theta})) + \sum_c \hat{p}(\underline{x}_c) KL(\hat{p}(\underline{x}_{c|}|\underline{x}_c) \| p(\underline{x}_{c|}|\underline{x}_c))$$

- i.e. setting $p(\underline{x}_c) = \hat{p}(\underline{x}_c)$

(44) (46): quick review

- IPF
- start with random guess of potential nos. $\psi_c^{(0)}(\underline{x}_c)$
- multiply by a ratio $\dfrac{\hat{p}(\underline{x}_c)}{p^{(t)}(\underline{x}_c)}$ (proportional)
- convexity only qualifies whether local or global (in this context)
- initialise random no. generator 100 times and run (to deal with converg/local/global)

Ex: 4th year PHD did not know how to run K-means

- You want to know theorem; but you have to get it working (OTI)
- (W): Implement these(?)

- Potential functions $\psi_c(x_c)$
- Some issues:- Spell-checking example
  - Build affinity models of charact. streams
  - e.g. consec appearances of 3 streams
  - use this criterion to score likelihood

$\psi_c(x_c) = \psi_c(x_1, x_2, x_3)$

'zyz' and 'ink'

- define potential function over a triplet of characters
- 26^3 different features and put a function on it?
- 26^5

(*) Above shows infeasibility of tabular potentials (catch to enumerat. of joint)

Feature based clique pot. (W): note instantiation of cliques in practice

---

(*) Features

(Q): Find a way of appropriately compressing the granularities

- Handcrafted feature design → role of human knowledge in A.I
- use feature-engin to save on rep. cost?
- Distinct from current, incomplete ML models (e.g. gigantic placeholder)
- Ex: Against the idea of human knowledge being ignored in ML

(*) Micropotentials for k features and weights defined over 3 charact. potential OR

- $\psi_c(c_1, c_2, c_3) = \exp\left\{\sum_{k=1}^{K} \theta_k f_k(c_1, c_2, c_3)\right\}$

(*) Overall potential (clique) is exp weighted sum of micropot.

(*) micropotent. distinct from tabular potentials.

(*) K parameters over K features → more compact

Combining features

(*) sliding window/overlapping sliding window

(*) Note how we can modify standard cts Gibbs. rep. for exp.
- Allowance of exponential / GLIMS

- (???): not entirely clear how to apply IPF in this case due to coupling of estimated θ_k and designed $f_k(c_1, c_2, c_3)$

Net of feature based UGMs:

- Scaled likelihood:- $\hat{\ell}(\theta; D) = \ell(\theta; D/N) = \frac{1}{N} \sum_n \log p(\underline{x}_n | \underline{\theta})$

$$= \sum_{\underline{x}} \hat{p}(\underline{x}) \log p(\underline{x} | \underline{\theta})$$

$$= \sum_{\underline{x}} \hat{p}(\underline{x}) \sum_i \theta_i f_i(\underline{x}) - \log \underline{z}(\underline{\theta})$$

(i) (ii)

(*) Calculus + derivatives → not fruitful.

Ex: Nonlinearities cause issues (e.g. log/inverse)
- Linearise it so argument can be exposed to linear attack

- log nes linear upper bound

- $\log \underline{z}(\underline{\theta}) \leq \mu \underline{z}(\underline{\theta}) - \log \mu - 1$

- Bound holds $\forall \mu$: $\mu = \underline{z}^{-1}(\theta^{(t)})$ — fixed point it. strategy — assume this (there is a previous version of $\underline{z}$)

(*) GIS derivation ⓦ ⓐ: Review

- define $\Delta\theta_i^{(t)} = \theta_i - \theta_i^{(t)}$ and introduce

- still nasty: → every

(*) note exp of weighted sum:- $\exp\left\{\sum_i \Delta\theta_i^{(t)} f_i(x)\right\}$

Ex: we make distinction between weight θ_K and $(\Delta\theta_i^{(t)})$ and features $f_i(x)$; algebraically the same.

- Treat f_i as weights; $\Delta\theta_i^{(t)}$ as arguments
- (*) impose prob. constraints (normally applying to weights) to f_i our assumed "weights".

} relax

(*) $\exp(\cdot)$ is convex → use Jensen's

- Algebraic trick often used in ML
- getting $\sum_i f_i(x)\exp(\Delta\theta_i^{(t)})$ → only linearly coupled with others.

(*) use low bound of scaled LL:- GIS

(*) use calculus:-

(*) giving update steps:-

(*) note:- $\dfrac{\sum_x \hat{p}(x) f_i(x)}{\sum_x p^{(t)}(x) f_i(x)}$

"weighted sum of feature by emp. prob. $\hat{p}(x)$"

"features weighted by inferred probability $p^{(t)}(x)$"

(*) Iterative re-scaling → connection with IPF.

(*) Summary GIS/IPF

- fixed point iterations on LL obj.
- one for tabular, feature based

(*) where does exponential come from? (☺: A move from Gauss → Gibbs?)

(☺)(R7): Review exp. family form (LS)

(*) Note on MLE; expectations of sufficient statistics under model match exp. feature average

- Note eq.

(*) Begin with exp. family → get a consequence.

- Reverse attitude
- Maximum entropy

- Impose constraint on distri: so you can't give me arbitrary ones
  - Fixed feat. exp: $\sum_{\underline{x}} p(\underline{x}) f_i(\underline{x}) = \alpha_i$ (from data) — feature must match
  - encode few assumpt. about world
- entropy as amt of randomness / amt of assumptions made

simplest distr. consistent with data

$$\max_p H(p(\underline{x})) = -\sum_{\underline{x}} p(\underline{x}) \log p(\underline{x})$$

s.t. $\sum_{\underline{x}} p(\underline{x}) f_i(\underline{x}) = \alpha_i$

$\sum_{\underline{x}} p(\underline{x}) = 1$ $\longrightarrow$ $p(\underline{x}|\underline{\theta}) = \frac{1}{Z(\underline{\theta})} \exp\left\{\sum_i \theta_i f_i(\underline{x})\right\}$

(*) Variational definition: - define a distrib. as a solution to a constrained optimisation problem.

(☺)(A8): Review Lagrangian sol.

(*) natural consequence gives: an exponential family distri.

(*) Benefit of information theoretic principles on ML → (☺) explore

---

- More general max entropy method
- incorporate prior distr on $\underline{x}$; reference it. ($u(\underline{x})$)
- Estimated distr has least addit. assumptions from priors
- use KL-divergence rather than entropy

$$\min_p \; KL(p(\underline{x}) \| h(\underline{x})) = \sum_{\underline{x}} p(\underline{x}) \log \frac{p(\underline{x})}{h(\underline{x})} = -H(p) - \sum_{\underline{x}} p(\underline{x}) \log h(\underline{x})$$

$$\text{s.t.} \sum_{\underline{x}} p(\underline{x}) f_i(\underline{x}) = \alpha_i$$

$$\sum_{\underline{x}} p(\underline{x}) = 1$$

$$\Longrightarrow p(\underline{x}|\underline{\theta}) = \frac{1}{\underline{z}(\underline{\theta})} h(\underline{x}) \exp\left\{\sum_i \theta_i f_i(\underline{x})\right\}$$

(*) Constraints from data

(Q) Since do constraints α_i come from?

- data itself is the constraint

- (h)(PG): automatic consistency?

- Geometric interp; general process :-

Either :-

1) Assume all exponential family distbs as "model" :-

$$\mathcal{E} = \{p(\underline{x}) : p(\underline{x}|\underline{\theta}) = \frac{1}{\underline{z}(\underline{\theta})} h(\underline{x}) \exp\left\{\sum_i \theta_i f_i(\underline{x})\right\}$$

OR

2) Assume all distbs satisfying moment constraints

$$\mathcal{M} = \{p(\underline{x}) : \sum_{\underline{x}} p(\underline{x}) f_i(\underline{x}) = \sum_{\underline{x}} \hat{p}(\underline{x}) f_i(\underline{x})\}$$

do not acknowledge roots

- Information geometry Pythagorean theorem :-

(*) Summary

(*) Exp family viewed as a sol. to variational exp → maximum entropy

→ inspires V.I; deep gen. models

(*)

Supplementary → structure learning (see supp.) / 2020 (

Case Study: CRFs (Lafferty) - at CMU

- insight of experienced modelling

- Lafferty paper → impressive ; clear rationale, motivation

(*) 'local normalisability is a double-edged sword

- makes computing simple'

(?) an - in context of HMM

(*) What you want is global normalisability

- use scores rather than enforcing local normalisibility

- use potentials

$$\exp\{\Sigma \theta_i f_i\}$$

- features corresp. to nodes

- use human knowledge for features

- 'Art of modelling' → totally comprehens.