Structure Learning

Lecturer : Barak Gross & Mark Berlin

Supervised By : Prof. Haim Kaplan

Background

- Now we look for a MN network.
- We already saw two approaches:
  - Constrained Based (Works better then in the Bayesian case).
    - Lack robustness to empirical noise.
    - Produce only a structure (for full description we need to run parameter estimation).
  - Score Based(Harder to compute likelihood function than in the Bayesian case).

- Is learning the global independences is appropriate?
- In the Bayesian case distinguished between learning global structure (as directed graph) and local structure(form of CPDs).
- We want to find a compact factorization but a complex graph structure.
- We will focus on the score based approach.

Reminder

Local Independencies

- We had 2 types of independencies in BN
  - local (Each node is independent of its non-descendants given it parents)
  - global (Induced by d-separation).
- MN has 3:

Pairwise Independencies - $\mathcal{I}_P(\mathcal{H}) = \{(X \perp Y | \mathcal{X} \setminus \{X, Y\}) \,|\, \{X, Y\} = e \notin \mathcal{H}\}$.

Markov Blanket - all neighbours of a node X, i.e $\mathbf{MB}_{\mathcal{H}}(\mathbf{X}) = \{Y \in \mathcal{X} | \{X, Y\} \in \mathcal{H}\}$.

Local Independencies - $\mathcal{I}_l(\mathcal{H}) = \{(X \perp \mathcal{X} \setminus (\{X\} \cup MB_{\mathcal{H}}(X)) \,|\, MB_{\mathcal{H}}(X)) \,|\, x \in \mathcal{X}\}$.

and the **Global** we saw (separation).

Learning with Independence Tests

- We assume:
  - $\forall u \;\; P^*(u) > 0$
  - P^* has a perfect map H^*
  - $\forall u \in V(H^*) \;\; \deg_{H^*} u < d^*$
- Still, using previous definitions we can't check independence traceably.
- Also, we need exponential number of samples in order to reduce statistical error.

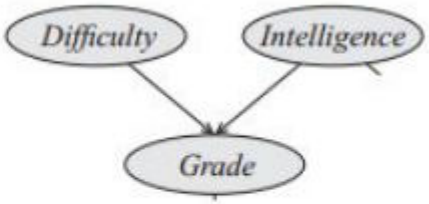
- Lets try to use the degree bound…
- Let $X, Y \in V(H^*)$. If they are not neighbors , then we can separate using their Markov Blanket.
- We can find Z s.t. Z separates X from Y, and $|Z| \leq \min(|MB_H(X)|, |MB_H(Y)|)$
- Since H^* is perfect map we can show: $(X, Y) \notin H^* \Leftrightarrow \exists Z\ s.t. |Z| \leq d^* \wedge P^* \vDash (X \perp Y|Z)$

Mixed Feelings

- Good news:
  - We used polynomial number of independence tests, resulting in a polynomial time algorithm.
  - If a single independence test fails at probability of at most ε , then at probability of at most $\sum_{k=0}^{d^*}\binom{n-2}{k}\epsilon$ one of the independence failed.
  - Under our assumption , if is small enough then we get the perfect map under high probability .
- Bad news:
  - A lot of important assumption (bounded degree , perfect map existence , etc.). If they are violated then we can get incorrect data.
  - In practice we may need a lot of samples to answer the independence test correctly.

Bad Example

- A perfect map (I(P)=I(H)) - usually doesn't exist even for positive P:
  - Example: We need edges D--G and I--G
  - Can we omit edge between I and D?
    - No, from D-separation, (I depend D|G) which will be violated.
  - Only minimal I-map is the fully connected graph.
    - For removing an edge, gives an unwanted independence
  - Which does not capture I ⊥ D which holds in P
  - Thus the minimal I-map is not a perfect map
    - i.e., I(P)≠I(H)

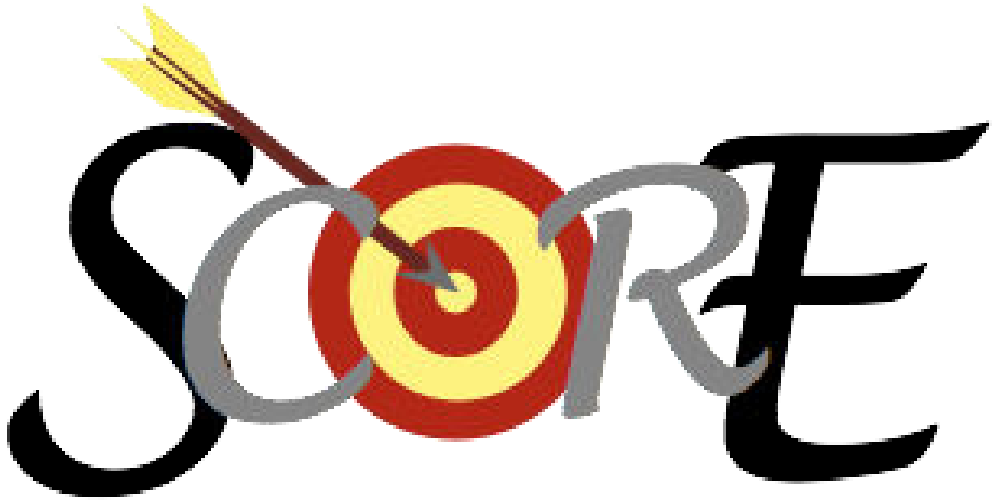


- Observe that in our case $\mathcal{D} \perp I|\emptyset$, meaning our algorithm will remove the edge between $\mathcal{D}$ and I, even though it is supposed to be there.

We are stuck

- This approach can be useful tool for obtaining qualitative insight into global structure of a distribution
- Good starting point for the search in the score based methods.

It’s Time To Score



Hypothesis Spaces

- Structure learning formulated as an optimization problem.
  - A set of possible networks
  - Objective Function
  - Search Strategy

- Several ways of formulating search space.
- Depends on the level of granularity at which we consider the network parametrization.
  - Coarsest Grained:
    - Space of different structure.
    - Model complexity measured in terms of clique size.
  - Next level:
    - Factor graphs.
    - Model complexity measured as size of factors.
  - Finest level of granularity:
    - Individual features in a log-linear model.
    - Measure sparsity at level of features included in the model.

Comparisons

- More fined grained hypothesis allows to select a parametrization that matches the property of our distribution without overfitting.
- Factor graphs:
  - We can distinguish between a large factor on k variables to $\binom{k}{2}$ pairwise disjoint factors.
- Log-Linear Models:
  - Distinguish between full factor on k variables and a single log-linear feature over same variables.

- Sparsity in log-linear model doesn't correspond directly to sparsity in the model structure.
- Single feature $f(d)$ introduces edges connecting all variables in d .
- Even models with small number of features can give rise to dense graphs.
- In finer graphs, search algorithms take smaller steps in the space, potentially increasing cost of learning procedures.

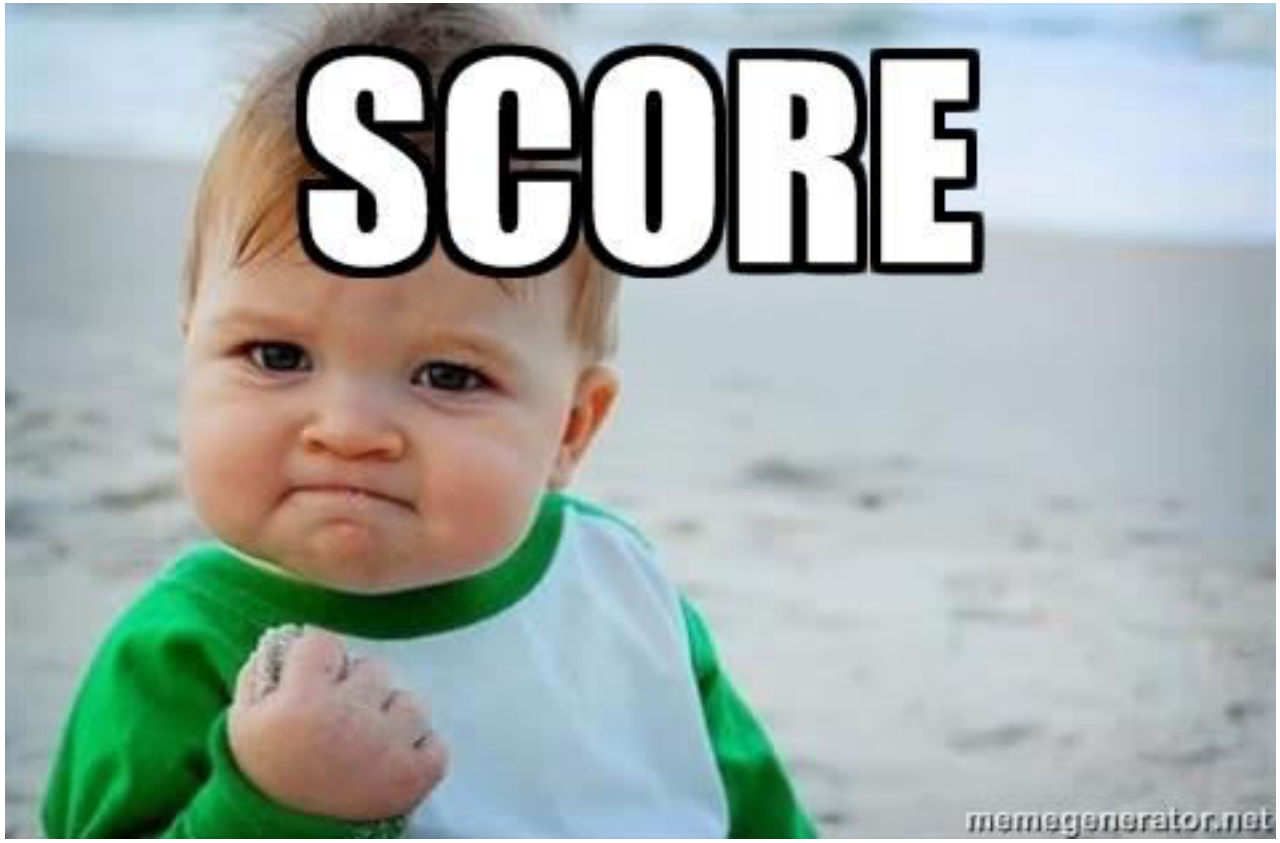
Search Space Of Log-Linear Models

- Ω = Set of features who can have non-zero weights.
- Select model structure M defined by some subset $\Phi[\mathcal{M}\ ] \subset \Omega$.
- Let $\Theta[\mathcal{M}]$ be set of parameterizations that are compatible with the model structure.

- Now we can define a compatible parametrization of a log-linear distribution:

$$P(\mathcal{X}|\mathcal{M},\theta) = \frac{1}{Z}\exp\left\{\sum_{i\in\Phi[\mathcal{M}]}\theta_i f_i(\xi)\right\} = \frac{1}{Z}\exp\{\, f^T\theta\}$$

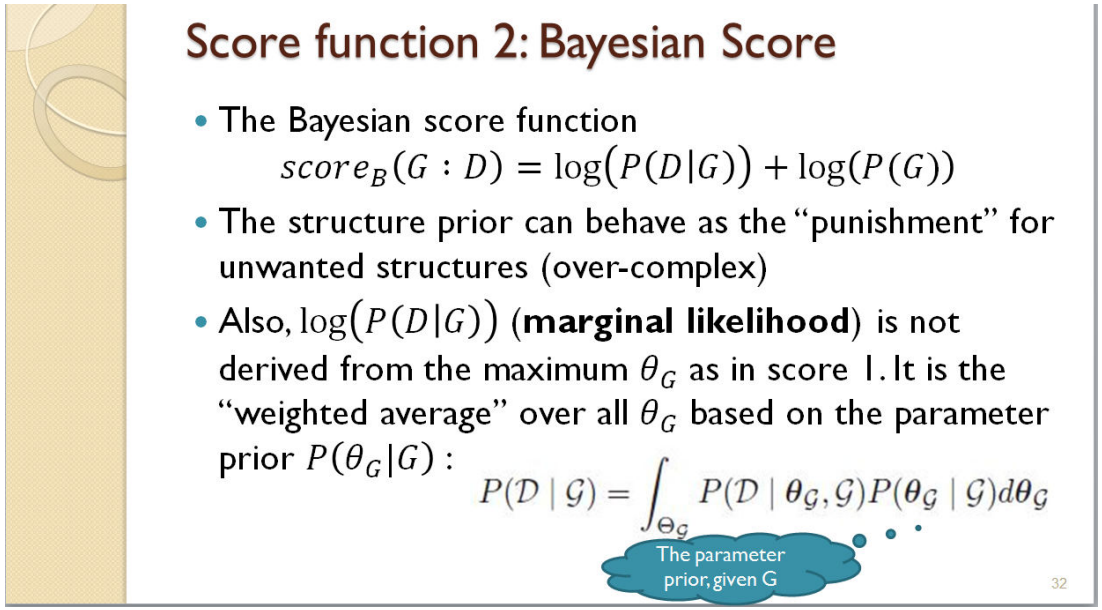
- We may insert some structural constraints.
- Popular choice : bounded tree-width.
  - Prevent overly dense network.
  - Reduced the chance of overfitting.
  - Learning become more efficient.
- Computing tree-width, and keeping it low is hard ☹.
- Many real-world distribution can't be represented by low tree-width graphs.


SCORE
memegenerator.net

Same as always

- The likelihood score:
  - $Score_L(\mathcal{M}:\mathcal{D}) = \max_{\theta \in \Theta[\mathcal{M}]} \ln P(\mathcal{D}|\mathcal{M},\theta) = \ell\left(\left\langle \mathcal{M}, \hat{\theta} \right\rangle : \mathcal{D}\right)$
- We discussed this function two weeks ago and got to the conclusion that it is too simple…

Bayesian Score



In Bayesian network we could evaluate efficiently.

Too hard for MN …

Laplace and BIC score

Bic Score

- Lets try to approximate the score asymptotically.

$$Score_{BIC}(\mathcal{M}|\mathcal{D}) = \ell\left(\left\langle \mathcal{M}, \widehat{\theta} \right\rangle : \mathcal{D}\right) - \frac{\dim(\mathcal{M})}{2} \ln \mathrm{M}$$

- The dimension of the model is the rank of the matrix whose rows are complete assignments to ξ_i to χ , whose columns are features f_i, and whose entries are $f_j(\xi_i)$.

Laplace Approximation

$$\text{score}_{Laplace}(\mathcal{M} \ : \ \mathcal{D}) = \ell(\langle \mathcal{M}, \tilde{\boldsymbol{\theta}}_{\mathcal{M}} \rangle : \mathcal{D}) + \ln P(\tilde{\boldsymbol{\theta}}_{\mathcal{M}} \mid \mathcal{M}) + \frac{\dim(\mathcal{M})}{2} \ln(2\pi) - \frac{1}{2} \ln |A|,$$

where $\tilde{\boldsymbol{\theta}}_{\mathcal{M}}$ are the parameters for $\mathcal{M}$ obtained from *MAP estimation*:

$$\tilde{\boldsymbol{\theta}}_{\mathcal{M}} = \arg\max_{\boldsymbol{\theta}} P(\mathcal{D} \mid \boldsymbol{\theta}, \mathcal{M}) P(\boldsymbol{\theta} \mid \mathcal{M}), \tag{20.28}$$

and A is the negative *Hessian* matrix:

$$A_{i,j} = -\frac{\partial}{\partial \theta_i \partial \theta_j} \left(\ell(\langle \mathcal{M}, \boldsymbol{\theta} \rangle : \mathcal{D}) + \ln P(\boldsymbol{\theta} \mid \mathcal{M}) \right),$$

evaluated at the point $\tilde{\boldsymbol{\theta}}_{\mathcal{M}}$.

- But is hard to compute the Hessian

Parameter Penalty Scores

- Alternative to marginal likelihood.
- Evaluate maximum posterior probability

$$Score_{MAP}(\mathcal{M}:\mathcal{D}) = \quad \ell\left(\langle \mathcal{M}, \tilde{\theta}_{\mathcal{M}} \rangle : \mathcal{D}\right) + \ln P(\tilde{\theta}_{\mathcal{M}} | \mathcal{M})$$

- Intuition : The prior “regularizes” the likelihood.

- MAP score is distribution over parameters (not structures).
- Any parameterization can be viewed as parameterization of the “universal” model with weights zero for features not in $\Phi[\mathcal{M}]$.
- Assuming that weight zero => the prior ignores this parameter.
- Score simply evaluates different parameterizations of the universal model.

L2 Regularization

- L2 Regularization will tend to drive the parameters toward zero, few will actually hit zero, and so structural sparsity will not be achieved.
- L2-Regularized MAP will generally give rise to fully connected structure.
- Not generally used for model selection.

L1 Regularization

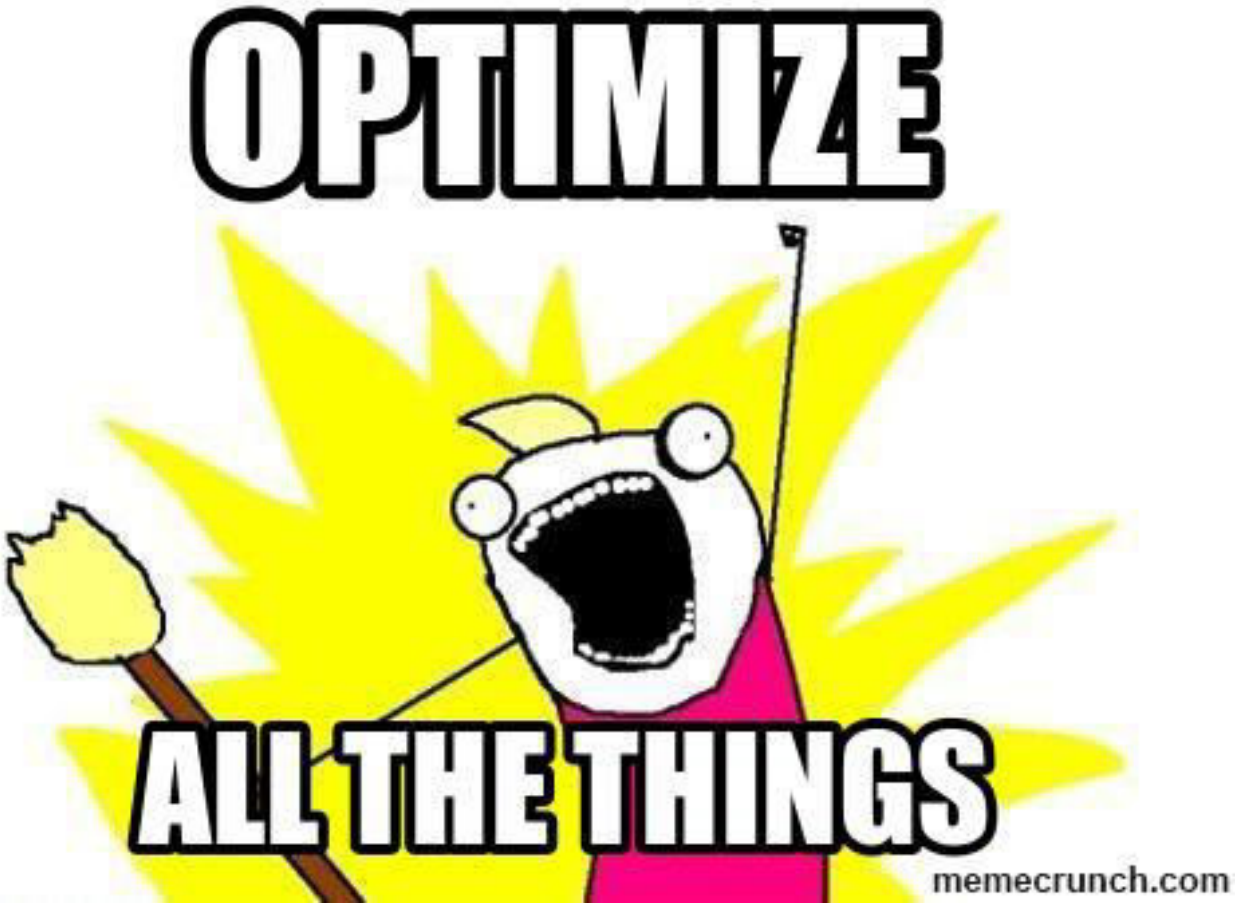
- Has the effect of driving parameters to zero.
- Give rise to sparse set of features.
- Later (if we will have time), we will see that L1 prior has other useful properties when used as a basis for a structure selection objective.

Block L1 Regularization

- Feature-level sparsity doesn't necessarily induce network sparsity.
- Lets try partition to blocks.
- We partition all the parameter into groups $\Theta_i = \{\theta_{i,1} \dots \theta_{i,k}\}$
- We define the next variant of L1 regularization:

$$-\sum_{i=1}^{l} \left| \sqrt{\sum_{j=1}^{k_i} \theta_{i,j}^2} \right|$$

Time to Optimize



Greedy Structure Search

- Local Search.
- General template.
- At each point the of the search , optimizes the model parameters relative to current feature set and structure score.
- Estimates the improvement of different structure modification steps.
- Selects some subset of modifications to implement and returns to the parameter optimization task.
- Repeated until a termination condition is reached.

Procedure Greedy-MN-Structure-Search (

Ω, // All possible features

$\mathcal{F}_0$, // initial set of features

$\text{score}(\cdot : \mathcal{D})$, // Score

)

$\mathcal{F}' \leftarrow \mathcal{F}_0$ // New feature set

$\boldsymbol{\theta} \leftarrow \mathbf{0}$

do

$\mathcal{F} \leftarrow \mathcal{F}'$

$\boldsymbol{\theta} \leftarrow$ Parameter-Optimize$(\mathcal{F}, \boldsymbol{\theta}, \text{score}(\cdot : \mathcal{D}))$

// Find parameters that optimize the score objective, relative to current feature set, initializing from the current parameters

for each $f_k \in \mathcal{F}$ such that $\theta_k = 0$

$\mathcal{F} \leftarrow \mathcal{F} - f_k$

// Remove inactive features

for each operator o applicable to $\mathcal{F}$

Let $\hat{\Delta}_o$ be the approximate improvement for o

Choose some subset $\mathcal{O}$ of operators based on $\hat{\Delta}$

$\mathcal{F}' \leftarrow \mathcal{O}(\mathcal{F})$ // Apply selected operators to $\mathcal{F}$

while termination condition not reached

return $(\mathcal{F}, \boldsymbol{\theta})$

Successor Evaluation

- Considerably more expensive than for BNs .
- At each stage need to evaluate the score for all candidates we wish to examine .
- Requires estimating parameters for the structure .
- Use heuristic that a single change to the structure does not result in drastic changes to model .

Choice Of Scoring function

- The greedy algorithm can be applied to any objective function.
- Choosing objective function directly influence our ability to optimize.
- We can’t rely on this objectives to induce sparsity in the model structure.
- We should choose the richest model and optimize its parameter.

- We can get more compact models using constraints.
- Generally introduce nontrivial combinatorial trade-offs between features.
- Multiple local optima
  - Generally intractable to find a global optimal solution.
- Another suggestion: When the score doesn't improve much – halt!
  - Usually good features are introduced early.
  - No guarantee to get even close to optimum.

- The penalties are discrete
  - They are important – they penalize the complexity of structure.
  - But now the score function is non-concave
    - No guarantee of convergence to the global optimum.
  - This problem also have risen in the Bayesian case
    - Could be alleviated by methods that avoid local maxima
      - Tabu search, random restarts, data perturbation , etc.
  - In Markov we have another solution: L1-Regularized likelihood.
    - Concave
    - Unique global optima
    - Give rise to sparse models

L1-Regularization For Structure Learning

- $Score_{\mathcal{L}_1}(\theta : \mathcal{D}) = \ell(\langle \mathcal{M}, \theta \rangle : \mathcal{D}) - \|\theta\|_1$

•Can be optimized in a way that guarantees convergence to the globally optimal solution.

•Optimizing L1-regularized log-likelihood is a convex optimization problem with no local optima.

- Introduce all of the possible features, optimize the resulting parameter θ relative to our objective.
- The penalty will drive some of the parameters to 0.
- Structure selection becomes parameter optimization.
  - Not feasible
- Generally implemented as double loop algorithm.

- Benefits to this regularized objective:
  - Do not need to consider feature deletion in the search.
  - We can consider feature introduction step in any order, and still achieve convergence to global optimum.
  - Simple and efficient test for determining convergence.
  - PAC bound.

Proposition 20.5 *Let $\Delta_L^{\text{grad}}(\theta_k : \boldsymbol{\theta}^l, \mathcal{D})$ denote the gradient of the likelihood relative to θ_k, evaluated at $\boldsymbol{\theta}^l$. Let β be the hyperparameter defining the L_1 prior. Let $\boldsymbol{\theta}^l$ be a parameter assignment for which the following conditions hold:*

- *For any k for which $\theta_k^l \neq 0$ we have that*

$$\Delta_L^{\text{grad}}(\theta_k : \boldsymbol{\theta}^l, \mathcal{D}) - \frac{1}{\beta}\text{sign}(\theta_k^l) = 0.$$

- *For any k for which $\theta_k^l = 0$ we have that*

$$|\Delta_L^{\text{grad}}(\theta_k : \boldsymbol{\theta}^l, \mathcal{D})| < \frac{1}{2\beta}.$$

Then $\boldsymbol{\theta}^l$ is a global optimum of the L_1-regularized log-likelihood function:

$$\frac{1}{M}\ell(\boldsymbol{\theta} : \mathcal{D}) - \frac{1}{\beta}\sum^{k} |\theta_i|.$$

Implication

- Convergence can be tested easily at each step.
- Usually we optimize the likelihood using the L-BFGS algorithm.
- There are some problems using it since L1-regularized likelihood is not continuously differentiable.

PAC Bound

Theorem 20.4

Let $\mathcal{X}$ be a set of variables such that $|Val(X_i)| \leq d$ for all i. Let P^* be a distribution, and $\delta, \epsilon, B > 0$. Let $\mathcal{F}$ be a set of all indicator features over all subsets of variables $\boldsymbol{X} \subset \mathcal{X}$ such that $|\boldsymbol{X}| \leq c$, and let

$$\Theta_{c,B} = \{\boldsymbol{\theta} \in \Theta[\mathcal{F}] \ : \ \|\boldsymbol{\theta}\|_1 \leq B\}$$

be all parameterizations of $\mathcal{F}$ whose L_1-norm is at most B. Let $\beta = \sqrt{c \ln(2nd/\delta)/(2M)}$. Let

$$\boldsymbol{\theta}^*_{c,B} = \arg \max_{\boldsymbol{\theta} \in \Theta_{c,B}} \mathbb{D}(P^* \| P_{\boldsymbol{\theta}})$$

be the best parameterization achievable within the class $\Theta_{c,B}$. For any data set $\mathcal{D}$, let

$$\hat{\boldsymbol{\theta}} = \arg \max_{\boldsymbol{\theta} \in \Theta[\mathcal{F}]} \text{score}_{L_1}(\boldsymbol{\theta} \ : \ \mathcal{D}).$$

Then, for

$$M \geq \frac{2cB^2}{\epsilon^2} \ln \left(\frac{2nd}{\delta} \right),$$

with probability at least $1 - \delta$,

$$\mathbb{D}(P^* \| P_{\hat{\boldsymbol{\theta}}}) \leq \mathbb{D}(P^* \| P_{\boldsymbol{\theta}^*_{c,B}}) + \epsilon.$$

- L1-regularized learning provides us with a model that is close to optimal, using polynomial number of samples.

Conclusion

- We say similarity and difference between learning Markovian Structure and Bayesian Structure.
- We saw score based and constrained based approaches, and concluded that we prefer score based.
- Model selection is actually a parameter optimization problem of “universal” model.
- We say several priors and objectives.
- Greedy algorithm is a little problematic to implement but it has a general idea that can be implemented.
- We saw that L1 regularization is pretty good prior.

ANDDDD…. That is it ☺