Machine Learning for Data Science (CS4786)
Lecture 25

Graphical Models, Wrapping up

April 30, 2015

Course Webpage :
http://www.cs.cornell.edu/Courses/cs4786/2015sp/

Announcements

- Assignment 3 due on May 5th, Bonus question can be turned in on May 6th
- Competition 2, due May 11th.
  - Speech data: task is to identify the word spoken
  - Feature extracted version provided, each utterance is a sequence of 83 time units/windows and for each time unit 13 MFC coefficients are provided
  - Each utterance is one of 7 words that are spoken
- You are provided with a training set with both the feature extracted speech and example labels
- Provided test set with only feature extracted speech, provide labels for these test examples.
- Competition is set up on Kaggle

Inference and Learning in Graphical Models

- Model data as a graphical model (use hidden or latent varibles)
- Inference:
  - What is the probability of some unobserved variable(s) given/conditioned on observation
  - What are the marginal probability of variables in the model
- Learning: based on observation pick the best parameters that explain the data
  - MLE:

$$\theta^* = \text{argmax}_{\theta \in \Theta} P(\text{Observations}|\theta)$$

  - MAP:

$$\begin{aligned}\theta^* &= \text{argmax}_{\theta \in \Theta} P(\theta|\text{Observations}) \\ &= P(\theta|\text{Observations}) \times P(\theta)\end{aligned}$$

Learning in Graphical Models: EM

- Power of wishful thinking: start with a wild guess
- E-step: perform inference to infer distributions over latent variables given observation (under current guess of parameters)

$$Q^t(\text{Latent}) = P_{\theta^{t-1}}(\text{Latent}|\text{Observation})$$

- Under the inferred distribution over latent variables, find parameters that optimize joint likelihood of variables

$$\theta^t = \text{argmax}_{\theta \in \Theta} \sum_{\text{Latent}} Q^t(\text{Latent}) \log P_\theta(\text{Observed}, \text{Latent})$$
$$= \text{argmax}_{\theta \in \Theta} \mathbb{E}_{\text{Latent} \sim Q^t} [\log P_\theta(\text{Observed}, \text{Latent})]$$

Inference required for EM (learning in general)

Exact Inference

Calculate the marginals/conditionals given parameter exactly

- Variable Elimination:
  - Always guaranteed to work
  - Can be computationally prohibitive
- Belief Propagation/Message Passing
  - Guaranteed to work only on tree structures and few other structure
  - Highly parallelizable, for many problems works well in practice

Exact inference in worst case is computationally hard!

Approximate Inference: Via Sampling

Relax, get approximate answers

- Often the generative story is easy: sample from generative process given parameters
- Law of large numbers say if draw large enough sample

$$\text{True Marginal} \approx \text{Empirical Marginal}$$

For EM algorithm,

$$\mathop{\mathbb{E}}_{\text{Latent}\sim Q^t}[\log P_\theta(\text{Obs}, \text{Lat})] \approx \frac{1}{m}\sum_{j=1}^{m} \log P_\theta(\text{Obs}, \text{Lat}_j)$$

Approximate Inference

- Variational inference:
  - Instead of true posterior, calculate posterior in a restricted family of distributions close to true one
  - Latent variables get their own set of parameters which we pick on the fly to make then close to true posterior
- Approximate message passing, expectation propagation, ...

Bigger Picture

- Dimensionality reduction, clustering and more generally learning

  There are no free lunches :(

- Probabilistic modeling makes assumptions or guesses about way data is generated or how variables are related
- Caution:
  - In the real world no modeling assumption is really true . . . there are good fits and bad fits
  - Choosing a model: Bias Vs Variance, Approximation error Vs estimation error, Expressiveness Vs amount of data
  - Choose the right model for the right job, there are no universally good answers
  - Feature extraction is an art (not covered in class)

Supervised Learning

- Training data: $(x_1, y_1), \ldots, (x_n, y_n)$ provided (typically assumed to be drawn from a fixed unknown distribution)
- Goal: Find a mapping $\hat{h}$ from input instances to outcome that minimizes $\mathbb{E}\Big[\ell(\hat{h}(x), y)\Big]$
  (ℓ is a loss function that measures error in prediction)

Generative Vs Discriminative Approaches

Generative approach:

- Input instances x_t's are generated based on/by y_t's
- We try to model $P(y, x) = P(x|y)P(y)$
- Example: Naive Bayes

Discriminative approach:

- We model $P(Y|X)$ or the boundary of classification
- Rationale: we are only concerned with predicting output y's given input x
- Example: linear regression, logistic regression

Probabilistic Story Vs Optimization Story

- Maximizing likelihood is same as minimizing negative log likelihood.
- Think of - log likelihood as loss function

$$-\log(P_\theta(Y|X)) \rightarrow \text{loss}(h_\theta(X), Y)$$

ie. θ parameterizes hypothesis for prediction or boundary

- MLE = Find hypothesis minimizing empirical loss on data
- Log Prior can be viewed as "regularization" of hypothesis

$$-\log(P(Y|X,\theta)) - \log(P(\theta)) \rightarrow \text{loss}(h_\theta(X), Y) + R(\theta)$$

- MAP = Find hypothesis minimizing empirical loss + regularization term
- Not all losses can be viewed as negative log likelihood

Semi-supervised Learning

- Can we used unlabeled examples to learn better?
- For instance, if we had a generative graphical model for the data: do example
- If we had prior information about the marginal distribution of X's and its relation to $P(Y|X)$

Active Learning

- Humans label the examples, can we get the learning algorithm in the loop?
- Learning algorithm picks the examples it wants labeled
- Eg. Margin based active learning, query points where model that fits observed data well so far disagree most

Domain Adaptation

- We learn a particular task on one corpus but want to use this learnt model to adapt with much fewer examples on another corpus
- Typical assumption: $P(Y|X)$ in both corpus remain fixed
- Marginal distributions change across the corpuses

Other Learning Frameworks

1. Transfer learning, multitask learning
2. Collaborative Filtering
3. Structured prediction
4. Online learning
5. ...