
Wrap Up

CS 536: Machine Learning
Littman (Wu, TA)

Administration

Next week, *you* talk!

- Different rooms M, back here W.

Quite impressive.

Today:

- EM for HMMs
- quick review/questions (focus on 2nd half, allowed to make connections!)
- possibly ERL videotape

EM for HMMs: Setting

We have a (long) sequence of observations.

Assume generated by n -state HMM.

Want to find the most likely HMM given the data.

If we assume uniform prior of HMMs, we want the HMM that is most likely given the data.

Recall EM

Applying the EM perspective requires:

1. Figure out the “missing” information that links the model and the data.
2. Show how to use the missing information to compute a maximum likelihood model.
3. Show how a model and the data can be used to estimate the missing information.

Missing Information in HMM

This information intentionally missing.

M-step

Given the state transitions, how do we build a maximum likelihood HMM model?

Can simply count the number of times each state was visited and the number of times each next state was visited to compute transition probabilities.

E-step

Given an observation sequence and an HMM, how compute the probability that each transition was made?

Forward-backward, Baum-Welch

Define:

$$\alpha_t(i) = \Pr(q_t = S_i \mid O_1 O_2 \dots O_T, \lambda)$$

$$\beta_t(i,j) = \Pr(q_t = S_i \wedge q_{t+1} = S_j \mid O_1 O_2 \dots O_T, \lambda)$$

We can compute these quantities efficiently using dynamic programming (again!).

Calculating a Model

$\sum_t^{T-1} \alpha_t(i)$ = Expected number of transitions out of state i during the path

$\sum_t^{T-1} \beta_t(i,j)$ = Expected number of transitions from state i to state j during the path

Note that $\sum_t^{T-1} \beta_t(i,j) / \sum_t^{T-1} \alpha_t(i)$ is an estimate of $\Pr(j|i)$ transition or a_{ij} .

EM iterates between these steps.

EM News

Bad News

- There are lots of local minima

Good News

- Local minima are usually adequate.

Notice

- EM does not estimate the num. of states.
- Can force HMM to have some zero-probability links. Set $a_{jj}=0$ in model $\lambda(0)$.
- Easy extension: Real valued outputs

Second Half

- 10/27/03: [Ch. 13: Reinforcement Learning](#)
- 10/29/03: [Reinforcement Learning Sampler](#)
- 11/03/03: [Learning in Game Theory](#)
- 11/05/03: [Support Vector Machines](#)
- 11/10/03: Support Vector Machines, continued
- 11/12/03: [Boosting](#) (Rob Schapire)
- 11/17/03: [Unsupervised Learning](#)
- 11/19/03: [Expectation Maximization](#)
- 11/24/03: [Hidden Markov Models](#)
- 12/01/03: [Datamining on Streams](#) (Yihua Wu)

Summary (1)

- RL: MDPs, actions, states, rewards, discounting, how Q-learning update rule works, function approximation can fail, POMDP model.
- Learning in Game Theory: matrix games, types of games (zero-sum, team, general-sum), minimax, definition of Nash equilibrium, candy game, prisoner's dilemma, repeated games, why are threats Nash?, stochastic games, relation to MDPs, why is Q-learning not the answer? Why is Nash-Q not the answer?

Summary (2)

- SVMs: margin idea, make learning an optimization problem, kernel trick, why high dimensional features can help, what's a "support vector"?
- boosting: weighted sum of weak classifiers (know how distributions are defined, how final classifier constructed), decision stumps (bag-of-word-type rules for text classification).

Summary (3)

- unsupervised learning: K-means (what are the steps, why does it converge?, where can we get the number of centers?), single linkage hierarchical clustering (how to use it to achieve richness, scale invariance, and consistency; but not all 3!), LSA applications.
- EM: Using EM (selecting the right missing information to fill in), expectation (in the coin example), maximization (somewhat generally).

Summary (4)

- HMMs: definition, how relates to POMDPs and MDPs, dynamic programming for computing most likely states and trajectories (exponential otherwise!)
- Streams: how does hoeffding bound help? Why can't we revisit the data? How does non-stationarity cause problems?

Midterm Summary (1)

"Things to know"

- Classification problems: error, training set, testing set, overfitting, cross-validation.
- Decision trees: training, use, interpretation.
- ANN: gradient ascent, single vs multi-layer, how to...
- Nearest neighbor: 1-NN, k-NN, how to...
- Naive Bayes

Midterm Summary (2)

- Hypothesis testing: confidence intervals, normal approx of binomial
- Bayes' rule for discrete values distribution
- VC dimension: for discrete values, for continuous values

Thanks to Mark Sharp for taking notes, typing them in, and sending them to me!