Collective Classification Regularization and Leave One Out Loss Minimization

David McAllester

Toyota Technological Institute at Chicago

Collective Classification

[Koller, Taskar, et al.], [Lafferty, McCallum, Pereira 01].
Assume a fixed distribution over pairs $\langle X, Y \rangle$.

We are interested in the case where Y is:

- An assignment of a word sequence to an acoustic signal X.
- An assignment of labels to the pages of web site X.
- An assignment of part of speech tags to word sequence X.
- An assignment of a parse tree to word sequence X.
- An assignment of a semantic interpretation to a statement in a dialogue (reference and logical form).

Loss Functions

For a density model $\hat{P}(Y|X)$ we have log loss

$$l(\langle X, Y \rangle) = \ln \frac{1}{\hat{P}(Y|X)}$$

For any unbounded loss it seems sufficient to consider thresholded loss.

$$\mathrm{P}\left[\mathrm{l}(\langle X, Y \rangle) \geq \mathrm{c}\right]$$

Other losses are possible. Taskar, Guestrin, and Koller consider Hamming distance — the number of places in Y that a labeling algorithm gets wrong.

ML Collective Logistic Regression

$$\hat{P}(Y|X,\Lambda) = \frac{1}{Z(X,\Lambda)} \quad e^{\Sigma_i \; \lambda_i f_i(X,Y)} \qquad (1)$$

$$Z(X,\Lambda) = \sum_{Y} \quad e^{\Sigma_i \; \lambda_i f_i(X,Y)}$$

Maximum likelihood (ML) training takes a sample $\langle x_1,\ y_1\rangle$, $\ldots$, $\langle x_T, ,\ y_T\rangle$ and sets

$$\Lambda^* = \operatorname*{argmax}_{\Lambda} \ \prod_{t=1}^{T} \hat{P}(y_t|x_t,\Lambda)$$

Characterizing Λ^*

$$\mathrm{c(f_i)} = \sum_t f_i(x_t, y_y)$$

$$\hat{\mathrm{c}}(f_i,\ \Lambda) = \sum_t \sum_y \hat{P}(y|x_t, \Lambda) f_i(x_t, Y)$$

$$\hat{\mathrm{c}}(f_i, \Lambda^*) = \mathrm{c(f_i)} \tag{2}$$

$\hat{P}(Y|X, \Lambda^*)$ maximumizes entropy s.t. (2).

MAP Collective Logistic Regression

$$P(\Lambda) = \prod_i \frac{1}{\sqrt{2\pi}\sigma_i} \exp\left(\frac{-\lambda_i^2}{2\sigma^2}\right)$$

$$\Lambda^* = \operatorname*{argmax}_\Lambda P(\Lambda) \prod_t \hat{P}(y_t | x_t, \Lambda)$$

$$\hat{\mathrm{c}}(f_i, \Lambda^*) = \mathrm{c}(\mathrm{f_i}) - \frac{\lambda_\mathrm{i}}{\sigma_\mathrm{i}^2}$$

The Exponential Prior

We now require $\lambda_i \geq 0$.

$$P(\Lambda) = \prod_i \alpha_i e^{-\alpha_i \lambda_i}$$

$$\hat{\mathrm{c}}(f_i, \Lambda^*) = \mathrm{c}(\mathrm{f_i}) - \alpha_\mathrm{i}$$

Language Modeling

$$\hat{P}(w_3|w_1w_2, \Lambda) = \frac{1}{Z} \; e^{\lambda_{w_1w_2w_3} + \lambda_{w_2,w_3} + \lambda_{w_3}}$$

This does surprisingly well when used with Gaussian priors [Chen and Rosenfeld 99], [Goodman 03].

Leave One Out Minimization: Regression Trees

Suppose Y is a bounded real number and let T be a decision tree without leaf labels.

Let $T(S)$ be the result of labeling each leaf by its mean in the sample S.

$$\ell(T(S)) = \mathrm{E}\left[(\mathrm{Y} - \mathrm{T(S)(X)})^2\right]$$

Let $\hat{l}(T, S)$ be the leave-one-out estimate of $\ell(T(S))$.

Regression Tree Generalization Theorem

$$\forall^{\delta} S \quad \forall T \;\; \ell(T(S)) \leq \hat{\ell}(T,\; S) + O\left(\sqrt{\frac{|T| + \ln(1/\delta)}{m}}\right)$$

Leave One Out Minimization: Kneser-Ney Language Modeling

$$\hat{P}(y|x,U) = \begin{cases} \frac{\mathrm{c}(\langle x,y\rangle)-\alpha}{\mathrm{c(x)}} & \text{if } \mathrm{c}(\langle x,y\rangle) > 0 \\ \\ \beta_1(x)\beta_2(x)U(y) & \text{otherwise} \end{cases}$$

$$\beta_1(x) = \alpha|V_x|$$

$$\beta_2(x) = \frac{1}{U(V/V_x)}$$

$$V_x = \{y : \langle x,y\rangle \in S\}$$

Setting U

Let $\hat{l}(U, S)$ be the leave-one-out estimate of $\mathrm{E}\left[\ell(\hat{\mathrm{P}}(\mathrm{Y}|\mathrm{X},\ \mathrm{U}))\right]$.

$$U^* = \underset{U}{\operatorname{argmin}}\ \hat{\ell}(U,\ S)$$

$$U^*(y) \approx \frac{|\{t :\ \mathrm{c}(\langle x_t, y_t\rangle) = 1 \wedge \mathrm{y_t} = \mathrm{y}\}|}{|\{t :\ \mathrm{c}(\langle x_t, y_t\rangle) = 1\}|}$$

Leave One Out Minimization: Expectation Maximization

[Johnson, personal communication]
When computing the expected counts for data point t use the previous iteration counts minus the counts for t.

Leave One Out Optimization
The General Case

Divide the model parameters into Λ and Γ.

$$\Gamma = \Gamma(\Lambda,\ S)$$

In regression trees $\Lambda = T$ and Γ is the set of leaf labels.

In a Kneser-Ney bigram model Λ is the unigram distribution and Γ is the bigram distribution.

$$\Lambda^* = \operatorname*{argmin}_{\Lambda} \hat{\ell}(\Lambda, S)$$

$$\hat{\ell}(\Lambda,\ S) = \text{LOO estimate of } \ \ell(\Lambda, \Gamma(\Lambda,\ S))$$

Summary

1. Collective classification seems central to ambitious AI problems.
2. Collective Logistic Regression seems to be a natural approach to 1).
3. Leave one out minimization seems to be a well motivated approach to "regularization".
4. It may be possible to adapt leave one out minimization to collective logistic regression.