Learning and Advice Seeking with Knowledge

1 Learning

The Learning problem is to be able to learn from both data and external sources of knowledge. In this case, there are two possibilities, either there exists a feasible solution by formulating the external knowledge as constraints or we need to find a re-conciliatory solution that allows trade off between data and the external knowledge source.

1.1 Constrained optimization

In our problem we take the probability, $P(y|x) \propto \psi$ and there use the softmax function to convert the function output to a probability distribution over different values that y can take as shown in equation 1

$$P = \frac{e^{\psi(y|x)}}{\sum_{y'} e^{\psi(y'|x)}} \tag{1}$$

We know that, $\frac{\partial logP}{\partial \psi} = I - P$. The conditional gradient optimization method allows us to find a ψ within a feasible set as allowed by external knowledge constraints, if such a solution ψ exists. Any constrained optimization procedure will yield the same solution with different efficiency. The conditional gradient algorithm adapted to our solution is as shown below in Algorithm 1. We start with this as it by passes projection in projected gradient ascent.

Result: ψ^* within feasible set ψ_C
while *Until Convergence* **do**
1. $\psi_0 = 0$, set initial function;
2. $s = \text{argmax}_{\psi \in \psi_C} \dot{\psi(I - P)}$, find max function;
3. $\psi^{t+1} = (1 - \gamma)\psi^t + \gamma s$, 1 step of FGB;
4. $\gamma = \frac{2}{2+\gamma}$

end

Algorithm 1: Conditional Gradient FGB with knowledge

In Phillip's advice case we know the knowledge is of the form $Precondition => I(y = c|x),\ c \in C$, where C is the label set. So the ψ corresponding to each piece of knowledge if it applies will be ∞ or $-\infty$ (because it is deterministic). If it was probabilistic, it would be $log(\frac{P(y=c|x)}{1-P(y=c|x)})$. So step 3. would essentially mean, moving a small step γ in the direction of ∞ or $-\infty$. When we have multiple pieces of knowledge in Phillip's work we aggregate this direction as $n_t - n_f$ and step size λ and therefore step 3. amounts to doing $\psi^{t+1} = \psi^t + \lambda(n_t - n_f)$. If a feasible set does not exist, we need a re-conciliatory trade of direction so we can write $\psi^{t+1} = \psi^t + \nu(I - P) + (1 - \nu)\lambda(n_t - n_f)$, where $\nu \in [0, 1]$.

1.2 Generalized knowledge gradient and objective function

In general for any kind of knowledge, we can say $\psi^{t+1} = \psi^t + \nu(I - P) + (1 - \nu)C$, where C is the gradient shift specific to each kind of knowledge. Therefore, we need an objective function g for which $\frac{\partial g}{\partial \psi} = \nu(I - P) + (1 - \nu)C$. Integrating we get the objective function g as shown in equation 2. This is a convex combination of the log probability and log odds scaled by C. In general we can say knowledge is of the form $precondtion => (P(y = c|x) = p\ or\ higher\ or\ lower)$, where $c \in C$ and C is set of labels.

$$\begin{aligned} g &= \int_{\psi} \nu(I - P) + \int_{\psi} (1 - \nu)C \\ &= \nu logP + (1 - \nu)\psi C \\ &= \nu logP + (1 - \nu)log(\frac{P}{1 - P})C \end{aligned} \tag{2}$$

1.3 Phillip's advice

Precondition is this framework is conjunction of literals and post condition has probability set to 1 for particular labels c if it applies. C in Phillip's method is $\lambda(n_t - n_f)$. This can also be written as $\lambda \sum_{a \in A} sign(I_a - P)$. Instead, we can sum or average them. The comparison is in the table below for the toy heart attack data set (4 iterations). As can be seen, the average, sum and Phillip's method match the scale of the data gradient and hence smoother updates in that order. But the count $n_t - n_f$ is efficient. We can decay $\lambda^{t+1} = \frac{2}{2+\lambda^t}$ in every iteration of boosting as in the conditional gradient algorithm in algorithm 1. This smoothes the update as also shown in the table below.

Accuracy	1	2	3	4
Phillip's method	0.545	1.0	1.0	1.0
Sum method	0.545	0.545	1.0	1.0
Average method	0.545	0.545	0.545	1.0
Phillip's method with decay	0.545	0.545	1.0	1.0

1.4 Cost sensitive data

For false positive, i.e precondition: $I = 0 \wedge P > 0.5$, In order to make it harder for P to predict positive, we subtract constant $k1$ from ψ, where $k1 = log(\frac{\alpha}{1-\alpha})$ and $\alpha \in [0, 0.5]$. So, the lower α is, the harder it makes it. Therefore, post condition is P is lower. $\alpha = 0.5$ means you get back the original ψ. Similarly for false negative, we add a constant $k2 = log(\frac{\beta}{1-\beta})$ and $\beta \in [0.5, 1]$. The higher β, the harder it is to predict negative. Therefore, in this case, $C = \lambda(log(\frac{\beta}{1-\beta}) - log(\frac{\alpha}{1-\alpha})) = \lambda log(\frac{\beta-\alpha}{\alpha-\beta})$, where λ is to scale down to match the data gradient and decays as $\lambda^{t+1} = \frac{2}{2+\lambda^t}$.

2 Advice seeking

In Phillip's advice the gradient $(I_a - P)$ that produces the greatest shift in $(I - P)$ can be used to rank multiple experts. This must only be done after every K iterations of boosting to avoid asking too many questions. Using a validation set doesn't make sense here as the training data could be noisy.

In cost sensitive data, after K iterations, if the prediction cost is high on a validation set from some measure, say F6 score or weighted AUC-ROC, a question can be posed to the expert as to the degree of sensitivity of false prediction. α is sensitivity of predicting false positives and β false negatives. So the experts number say between $1 - 10$ is scaled appropriately as α and β.

3 Connection to A* search

A* search expands the node according to minimum $f(x) + h(x)$, $x \in X$ where X is the set of examples at a node, f is the true distance to goal and h is the heuristic distance to goal. The constraint is that $h(x)$ must be admissible and consistent. If we treat the information loss on predictions using knowledge gradients as $f(x)$ and information loss on predictions using data gradients as $h(x)$, then, by the definition of expert, the two conditions. This works because the data cannot possibly lower information loss faster than expert knowledge can.

4 Summary and key takeaways

A generalized objective function to accomodate any kind of external knowledge is derived based on a reasonable premise arising from traditional constrained convex optimization. Phillip's advice and Shuo's advice (modified) is shown to fit in this framework.