CS 165: Foundations of Machine Learning **Winter 2019**

Lecture 15: Data issues

Lecturer: Anima Anandkumar *Scribes: Yujia Huang, Florian Schaefer*

15.1 Review

Uncertainty quantification In the last lecture we talked about *uncertainty quantification* and proposed to (conceptually) divide the errors of our estimation procedure into *aleatoric error* and *epistemic* error. Aleatoric errors arise from our data containing only limited information about or target of estimation (for example due to noise). Epistemic error (Estimation variance) in contrast arise if there are regions of feature space where we were not able to collect (sufficient amounts of) data and where our model is not able to reliable extrapolate from the seen data.

Bayes by Backprop Principle While providing good prediction accuracy, neural typically do not provide an accurate measure of their own uncertainty. Conceptually, Bayesian methods provide a simple and principled way to obtain estimators with better uncertainty quantification. In the context of neural nets, this amounts to compute a posterior distributions on the set of weights, which is conceptually very expensive to compute due to the large number of weights and correspondingly high dimension of the state space. As discussed in the last lecture, we can nevertheless find an*approximation* of the posterior within a parametric class of distributions by using variational methods.

15.2 Shortcomings of uncertainty based sampling

We next discussed the shortcomings of active learning, and how some of them can be alleviated by more general frameworks like for interactive learning such as bandits and Reinforcement Learning (RL). The first downside of active learning is that it is mypopic (short-sighted) in that it only aims to minimize the uncertainty after the next step. Furthermore, it is very dependent on the an accurate measure of the uncertainty. In particular, the lack of an explicit *exploration* policy makes it very vulnerable to overconfidence due to "unknown unknowns". Finally, the data collection process will be very heavily skewed towards the particular learning objective used for the learning, limiting its usefulness for other downstream tasks. Finally, active learning does not allow to account for the impact that the process of collecting data can have on the target of estimation. Bandits and more general techniques from reinforcement learning allow to add a planning element to the policy, allowing to optimize the policy over larger time horizons. The more explicit treatment of the exploration-exploitation trade-off can furthermore reduce the effects of poorly calibrated uncertainty estimates.

15.3 Data annotation issues

Continuing a discussion from last lecture we looked at the problem of label uncertainty caused by, for example, mistakes of the annotators. While this problem is often addressed by letting multiple annotators work on the same data and then determining the *ground truth* by majority vote, this need not be the best

policy. Instead one can train a neural net on the uncorrected data and then use the output of the neural net as ground truth to estimate the confusion matrices of the annotators. By using the confusion matrix thus obtained to retrain a new neural network, decreasing the weighting of bad annotators, we obtain an iterative scheme to estimate the true labels. This begs the question whether, under a fixed budget of annotations, we should should annotate the same sample by multiple annotators, or whether we should only annotate each sample once, to annotate the maximal number of data points. Indeed, under the assumption

- that the best predictor (without label noise) is accurate enough,
- that all workers have the same quality
- that the probability of each annotations to be correct is larger than 83%,

that the generalization error is minimized by using only a single annotation per sample. More precisely, the following theorem can be shown:

Theorem 15.1 (Khetan et al. (2017)) *Define $N := nr$ to be the number of total annotations collected on n training examples with redundancy r. Suppose $\min_{f\in\mathcal{F}} R_{l,\mathcal{D}}(f) \leq 1/4$. For any hypothesis class $\mathcal{F}$ with a finite VC dimension V and any $\delta < 1$, there exists a universal constant C such that if N is large enough and satisfies*

$$N \geq \max\left\{Cr\left(\left(\sqrt{V}+\sqrt{\log(1/\delta)}\right)/(1-2\alpha)\right)^2, 2^{12}m\log(2^6 m/\delta)\right\}, \tag{15.1}$$

then for binary classification with 0-1 loss function l, the $\hat{f}$ and π returned by the algorithm after $T = 2$ iterations satisfy

$$R_{l,\mathcal{D}} - \min_{f\in\mathcal{F}} R_{l,\mathcal{D}} \leq \frac{C\sqrt{r}}{1-2\beta_\epsilon}\left(\sqrt{\frac{V}{N}}+\sqrt{\frac{\log(1/\delta)}{N}}\right). \tag{15.2}$$

Here, β_ϵ measures the inaccuracy of the annotators, with larger β_ϵ leading to a larger bound. Compared to classical VC bound, the quality of annotators acts like a multiplication effect. If we fix N, the best r would be 1. However, we may need large enough number of samples to get good estimates of annotator quality. An open question is how to bound the number of samples needed if we don't know the annotator quality.

15.4 Data Augmentation

If data is scarse, significant improvements can often be achieved by augmenting the existing data new data, obtained from suitable transformations of the existing data. In computer vision, one often copies of the original dataset, after applying rotations, cropping, and/or various filters and noise structures. In speech recognition, one can add background noise or spectral transforms. These approaches can be seen as very simple ways to introduce modelling assumptions (translation invariance, invariance to noise etc) into the estimation procedure. One promising avenue towards improving the interpretability of machine learning models is to find more general ways of imposing such structure into otherwise black-box models. An example of this is given by Arabshahi et al. (2018), who propose a method to combine symbolic and numerical input data for the learning of functions. In this work the authors propose to learn neural networks that implement basic mathematical expressions on both numerical values and symbolic representations The model can then be trained not only with numerical data, but also symbolic information about, for example, algebraic identities, that can be randomly generated and labeled by a computer algebra software. A notable feature of this method is that the resulting model generalizes to expressions that have a larger depth than the expressions that were used as training data. The main remaining open problem is scalability, in particular how to limit error propagation when dealing with equations of large depth.

References

Arabshahi, F., S. Singh, and A. Anandkumar
2018. Combining symbolic expressions and black-box function evaluations in neural programs. *arXiv preprint arXiv:1801.04342*.

Khetan, A., Z. C. Lipton, and A. Anandkumar
2017. Learning from noisy singly-labeled data. *arXiv preprint arXiv:1712.04577*.