Scalable Large-Scale Classification with Latent Variable Augmentation

Francisco J. R. Ruiz
Columbia University
University of Cambrdige

Michalis K. Titsias
Athens University of
Economics and Business

David M. Blei
Columbia University

1 Introduction

Categorical distributions are applied in many areas of machine learning, such as large-scale classification (Gupta et al., 2014) or neural language models (Bengio et al., 2006). They also play an important role in economics and discrete choice models (McFadden, 1978; McFadden and Train, 2000).

One way to parameterize categorical distributions is through the concept of utility. In particular, for each possible value of the outcome $k \in \{1, \dots, K\}$, we define a utility ψ_k. In a classification scenario, the utility for class k is typically a function of observed features; in discrete choice models, the utility is the baseline preference for choice k. Under this construction, the observed outcome y depends on all the utilities ψ_k, corrupted by random noise ε_k, such that

$$y = \arg\max_k \left(\psi_k + \varepsilon_k\right). \tag{1}$$

Here, we have not specified the distribution of the noise terms ε_k. If they are independent and Gumbel distributed, the probability $p(y = k \mid \psi)$ is given by the well-known softmax; if they are independent and Gaussian distributed, we recover the multinomial probit model. Here, we consider that the errors are independent and they follow some distribution $\phi(\cdot)$, i.e., $\varepsilon_k \sim \phi(\cdot)$.

Though useful, this construction may lead to computational issues when the number of classes or choices K is large. Typically, the computational cost to process the probability of each observation is $\mathcal{O}(K)$, which becomes prohibitive specially when embedded in an iterative procedure such as gradient descent.

In this paper, we put forward a scheme to speed up the training procedure in models that incorporate utility-based categorical distributions. Our approach is based on two ideas: latent variable augmentation and stochastic variational inference (Hoffman et al., 2013). Essentially, we first introduce an auxiliary latent variable into the model, and then we perform variational inference on the augmented space. Importantly, the resulting variational lower bound involves a summation over the number of classes, so we can leverage stochastic variational inference techniques to subsample a subset of classes at each iteration. We describe our method in detail in Section 2.

Related work. There exist several other methods that try to deal with the high computational cost of the softmax for large values of K. Among those, there are methods that attempt to perform the exact computations (Gopal and Yang, 2013; Vijayanarasimhan et al., 2014), sampling-based methods (Bengio and Sénécal, 2003; Mikolov et al., 2013; Devlin et al., 2014; Ji et al., 2016), approximate distributed approaches (Grave et al., 2017), approximations such as noise contrastive estimation (Smith and Jason, 2005; Gutmann and Hyvärinen, 2010) or random nearest neighbor search (Mussmann et al., 2017), hierarchical or stick-breaking models that forgo the utility-based perspective (Kurzynski, 1988; Morin and Bengio, 2005; Tsoumakas et al., 2008; Beygelzimer et al., 2009; Dembczyński et al., 2010; Khan et al., 2012), and variational lower bounds (Titsias, 2016).

Our work differs from previous approaches in that it is valid for a general class of methods including the softmax and the multinomial probit model. Furthermore, it is derived from a variational inference perspective and thus it is a rigorous lower bound on the marginal likelihood $p(y \mid \psi)$.

31st Conference on Neural Information Processing Systems (NIPS 2017), Long Beach, CA, USA.

2 Method Description

Consider the model in Eq. 1, where $\varepsilon_k \sim \phi(\cdot)$. By marginalizing out the error terms, we can express the probability of outcome k as an integral,

$$\begin{aligned} p(y = k \,|\, \psi) &= \text{Prob}\left(\psi_k + \varepsilon_k \geq \psi_{k'} + \varepsilon_{k'} \;\; \forall k' \neq k\right) \\ &= \int_{-\infty}^{+\infty} \phi(\varepsilon_k) \left(\prod_{k' \neq k} \int_{-\infty}^{\varepsilon_k + \psi_k - \psi_{k'}} \phi(\varepsilon_{k'}) d\varepsilon_{k'} \right) d\varepsilon_k \\ &= \int_{-\infty}^{+\infty} \phi(\varepsilon) \left(\prod_{k' \neq k} \Phi(\varepsilon + \psi_k - \psi_{k'}) \right) d\varepsilon. \end{aligned} \tag{2}$$

Here, we have defined $\Phi(\varepsilon) = \int_{-\infty}^{\varepsilon} \phi(\tau) d\tau$ as the cumulative density function of the error terms, and we have renamed the auxiliary (dummy) variable ε_k as ε to avoid cluttering of notation. The expression in Eq. 2 is analogous to the one by Girolami and Rogers (2006) for the multinomial probit model, although we do not necessarily assume a Gaussian density $\phi(\cdot)$. Note that, for a Gumbel distribution $\phi(\cdot)$, Eq. 2 simplifies to the well-known softmax expression.

We now consider the *augmented model* in which we instantiate the auxiliary variable ε. We can write its joint probability as

$$p(y = k, \varepsilon \,|\, \psi) = \phi(\varepsilon) \prod_{k' \neq k} \Phi(\varepsilon + \psi_k - \psi_{k'}). \tag{3}$$

By construction, the marginal $\int p(y = k, \varepsilon \,|\, \psi) d\varepsilon$ recovers the original model,[1] $p(y = k \,|\, \psi)$. Crucially, the joint in Eq. 3 involves a product over the classes or choices $k' \neq k$. This means that the log joint involves a summation over k', which allows us to apply stochastic optimization techniques in the variational inference framework, following Hoffman et al. (2013).

Variational inference. We perform variational inference (Jordan et al., 1999; Blei et al., 2017) on the augmented model $p(y, \varepsilon \,|\, \psi)$. In particular, we form the evidence lower bound (ELBO) $\mathcal{L}$ as

$$\log p(y = k \,|\, \psi) \geq \mathcal{L} \triangleq \mathbb{E}_{q(\varepsilon)} \left[\log \phi(\varepsilon) + \sum_{k' \neq k} \log \Phi(\varepsilon + \psi_k - \psi_{k'}) - \log q(\varepsilon) \right]. \tag{4}$$

The optimal $q(\varepsilon)$ is equal to the conditional posterior of ε given y, i.e.,

$$q^\star(\varepsilon) = p(\varepsilon \,|\, y = k, \psi) \propto \phi(\varepsilon) \prod_{k' \neq k} \Phi(\varepsilon + \psi_k - \psi_{k'}). \tag{5}$$

This choice of $q(\varepsilon)$ leads to equality in the bound; however, it is generally not available in closed form, and we need to choose alternative forms for the variational distribution. We next describe how to choose $q(\varepsilon)$ for three different models: the softmax, the multinomial probit, and the multinomial logistic. These models only differ in the distribution over the error terms, $\phi(\cdot)$.

2.1 Softmax Augmentation

In the softmax model, the distribution over the error terms is a standard Gumbel,

$$\phi_{\text{softmax}}(\varepsilon) = \exp\{-\varepsilon - e^{-\varepsilon}\}, \quad \text{and} \quad \Phi_{\text{softmax}}(\varepsilon) = \exp\{-e^{-\varepsilon}\}. \tag{6}$$

In this model, it turns out that the optimal variational distribution $q^\star(\varepsilon)$ in Eq. 5 (which achieves the bound with equality) has closed-form expression:

$$q^\star_{\text{softmax}}(\varepsilon) = \text{Gumbel}(\varepsilon\,;\, \log \eta^\star, 1), \qquad \text{with } \eta^\star = 1 + \sum_{k' \neq k} e^{\psi_{k'} - \psi_k}. \tag{7}$$

[1] A different latent variable augmentation technique for variational inference in multinomial models has also been applied by Linderman et al. (2015). However, while their focus is to achieve conjugacy, ours is scalability.

Algorithm 1: Variational inference on the augmented softmax

for $t = 1, 2, \ldots,$ **do**
  Sample a minibatch of data, $\mathcal{B} \subseteq \{1, \ldots, N\}$
  `# Local step (E step):`
  for $n \in \mathcal{B}$ **do**
    Sample a set of labels, $\mathcal{S}_n \subseteq \{1, \ldots, K\} \setminus \{y_n\}$
    Update the natural parameter, $\eta_n \leftarrow 1 + \frac{K-1}{|\mathcal{S}_n|} \sum_{k \in \mathcal{S}_n} e^{\psi_{nk} - \psi_{ny_n}}$
  end
  `# Global step (M step):`
  Sample a set of labels, $\mathcal{S}_n \subseteq \{1, \ldots, K\} \setminus \{y_n\}$ for each $n \in \mathcal{B}$
  Set $g^{(t)} \leftarrow -\frac{w}{\sigma_w^2} - \frac{N}{|\mathcal{B}|} \frac{K-1}{|\mathcal{S}_n|} \sum_{n \in \mathcal{B}} \frac{1}{\eta_n} \sum_{k \in \mathcal{S}_n} \nabla_w e^{\psi_{nk} - \psi_{ny_n}}$
  Update $w \leftarrow w + \rho^{(t)} g^{(t)}$
end

Even though $q^\star_{\text{softmax}}(\varepsilon)$ has an analytic form, its natural parameter is computationally hard to compute because it involves a summation over $K - 1$ classes. Instead, we set

$$q_{\text{softmax}}(\varepsilon\,;\,\eta) = \text{Gumbel}(\varepsilon\,;\,\log(\eta), 1). \tag{8}$$

Substituting this choice for $q_{\text{softmax}}(\varepsilon\,;\,\eta)$ into Eq. 4 gives the following ELBO:

$$\mathcal{L}_{\text{softmax}} = 1 - \log(\eta) - \frac{1}{\eta}\left(1 + \sum_{k' \neq k} e^{\psi_{k'} - \psi_k}\right). \tag{9}$$

This bound coincides with the log-concavity bound (Bouchard, 2007; Blei and Lafferty, 2007), although we have derived it from a completely different perspective. This derivation allows us to optimize η in a computationally efficient manner, as we describe next.

First, note that the Gumbel$(\varepsilon\,;\,\log\eta, 1)$ is an exponential family distribution whose natural parameter is η. This allows us to apply stochastic variational inference as described by Hoffman et al. (2013). We iterate between a local step in which we optimize η, and a global step in which we optimize the parameters ψ. In the local step (E step), we subsample a set of classes (choices), $\mathcal{S} \subseteq \{1, \ldots, K\} \setminus \{y\}$ of size $S = |\mathcal{S}|$, and we estimate the natural parameter as $\hat{\eta} = 1 + \frac{K-1}{S} \sum_{k' \in \mathcal{S}} e^{\psi_{k'} - \psi_y}$. In the global step (M step), we take a gradient step with respect to (the parameters of) ψ, fixing $\eta = \hat{\eta}$.

Example: Classification. Consider a classification scenario with N datapoints x_n and their corresponding labels $y_n \in \{1, \ldots, K\}$. Each observation is associated with its parameters ψ_{nk}, typically chosen as $\psi_{nk} = w_k^{(0)} + x_n^\top w_k$ for some feature vector x_n and weights w_k. We wish to learn the weights w_k. Algorithm 1 summarizes the procedure to obtain the *maximum a posteriori* (MAP) estimates[2] of the weights, assuming a Gaussian prior $p(w) = \mathcal{N}(w \,|\, 0, \sigma_w^2 I)$. In the local step, we optimize η_n for each observation n; in the global step, we optimize the weights w_k.

Numerical stability. One way to prevent potential numerical instabilities due to the $\exp\{\cdot\}$ function is to sample the labels $\mathcal{S}_n$ only once. If we do this, then some terms cancel out in the expression for the gradient in the M step and we can perform numerically stable computations.

2.2 Multinomial Probit Augmentation

In the multinomial probit model, the distribution over the error terms is a standard Gaussian,

$$\phi_{\text{probit}}(\varepsilon) = \mathcal{N}(\varepsilon\,;\,0, 1) = \frac{1}{\sqrt{2\pi}} e^{-\frac{1}{2}\varepsilon^2}, \quad \text{and} \quad \Phi_{\text{probit}}(\varepsilon) = \int_{-\infty}^{\varepsilon} \mathcal{N}(\tau\,;\,0, 1) d\tau. \tag{10}$$

The ELBO becomes

$$\mathcal{L}_{\text{probit}} = \mathbb{E}_{q(\varepsilon)}\left[-\frac{1}{2}\log(2\pi) - \frac{1}{2}\varepsilon^2 + \sum_{k' \neq k} \log \Phi_{\text{probit}}(\varepsilon + \psi_k - \psi_{k'}) - \log q(\varepsilon)\right]. \tag{11}$$

[2] If we are interested in posterior inference on the weights instead of the MAP solution, we can proceed similarly, simply taking the expectation of the exponential function, $\mathbb{E}_{q(w)}\left[e^{\psi_{nk} - \psi_{ny_n}}\right]$.

model	log-likelihood train	log-likelihood test	accuracy train	accuracy test
exact softmax	-0.1835	-0.3594	0.956	0.903
one-vs-each (Titsias, 2016)	-0.2635	$\mathbf{-0.3618}$	0.923	$\mathbf{0.900}$
softmax augmentation	-0.2925	$\mathbf{-0.3662}$	0.917	$\mathbf{0.901}$
multinomial probit augmentation	-0.2656	-0.4122	0.922	0.895
multinomial logistic augmentation	-0.2730	-0.3805	0.918	0.898

Table 1: Performance on the extended MNIST data with 100 classes.

In contrast to the softmax case, this expectation is intractable. Fortunately, we can perform inference using the reparameterization trick (Kingma and Welling, 2014; Titsias and Lázaro-Gredilla, 2014; Rezende et al., 2014) by building unbiased Monte Carlo estimates of the gradient of the ELBO with respect to the variational parameters. We choose a Gaussian variational distribution, $q_{\text{probit}}(\varepsilon\,;\,\mu,\sigma) = \mathcal{N}(\varepsilon\,;\,\mu,\sigma^2)$, and we use amortized inference (Dayan et al., 1995; Gershman and Goodman, 2014) to avoid optimizing the local variational parameters μ and σ independently for each datapoint. In particular, we let the local variational parameters $\mu = \mu(x)$ and $\sigma = \sigma(x)$ be functions of the observed input x. We parameterize these functions using neural networks (Mnih and Gregor, 2014; Kingma and Welling, 2014; Rezende et al., 2014), typically called *recognition networks*.

The reparameterization trick requires evaluating the log joint, $\log p(y, \varepsilon \mid \psi)$. To avoid the expensive computation involved in the summation over k', we use unbiased stochastic estimates obtained via subsampling of observations and classes (Hoffman et al., 2013).

2.3 Multinomial Logistic Augmentation

We obtain the multinomial logistic model by placing a standard logistic distribution over the errors,

$$\phi_{\text{logistic}}(\varepsilon) = \sigma(\varepsilon)\sigma(-\varepsilon), \quad \text{and} \quad \Phi_{\text{logistic}}(\varepsilon) = \sigma(\varepsilon), \tag{12}$$

where $\sigma(\varepsilon) = \frac{1}{1+e^{-\varepsilon}}$ is the sigmoid function. In this case, the ELBO can be expressed as

$$\mathcal{L}_{\text{logistic}} = \mathbb{E}_{q(\varepsilon)}\left[\log\sigma(\varepsilon) + \log\sigma(-\varepsilon) + \sum_{k'\neq k}\log\sigma(\varepsilon + \psi_k - \psi_{k'}) - \log q(\varepsilon)\right]. \tag{13}$$

Note the resemblance between this expression and the one-vs-each bound of Titsias (2016), which is a lower bound on the softmax model. Similarly to the multinomial probit case, this expectation is intractable, and thus we use the reparameterization trick. We choose a logistic variational distribution,[3] $q_{\text{logistic}}(\varepsilon\,;\,\mu,\beta) = \frac{1}{\beta}\sigma\Big(\frac{\varepsilon-\mu}{\beta}\Big)\sigma\Big(-\frac{\varepsilon-\mu}{\beta}\Big)$, and we fit μ and β using amortized inference.

3 Experiments

In our preliminary experiments, we build a dataset with $K = 100$ classes starting from MNIST.[4] To do that, we split the dataset (training and test) into 10 groups, each containing 10% of the instances. On each group, we perform a random permutation of the pixels. This effectively leads to a classification scenario with $K = 100$ classes, in which each class has 600 training samples.

We compare the following methods: exact softmax; one-vs-each bound (Titsias, 2016); softmax augmentation (Section 2.1); multinomial probit augmentation (Section 2.2); and multinomial logistic augmentation (Section 2.3). For the augmentation approaches, as well as for the one-vs-each method, we use $|\mathcal{S}_n| = S = 10$ samples. We use a minibatch size of $|\mathcal{B}| = 200$ datapoints, and we perform MAP estimation on the classification weights and intercepts (the prior variance is $\sigma_w^2 = 1$).

In Table 1, we report log-likelihood and accuracy on both the training and the test sets. The softmax augmentation procedure performs similarly to the one-vs-each bound, and they are close to the exact softmax. The probit augmentation and logistic augmentation perform better on the training set but worse on test; we believe this is due to overfitting of the recognition networks. In our ongoing work we plan to understand this, and also to carry out experiments on datasets with larger values of K.

[3] We could consider other functional forms for $q_{\text{probit}}(\varepsilon)$ and $q_{\text{logistic}}(\varepsilon)$. We leave that for future work.

[4] http://yann.lecun.com/exdb/mnist

Acknowledgments

This work is supported by NSF IIS-1247664, ONR N00014-11-1-0651, DARPA PPAML FA8750-14-2-0009, DARPA SIMPLEX N66001-15-C-4032, the Alfred P. Sloan Foundation, and the John Simon Guggenheim Foundation. Francisco J. R. Ruiz is supported by the EU H2020 programme (Marie Skłodowska-Curie grant agreement 706760).

References

Bengio, Y., Schwenk, H., Senécal, J.-S., Morin, F., and Gauvain, J.-L. (2006). Neural probabilistic language models. In *Innovations in Machine Learning*. Springer.

Bengio, Y. and Sénécal, J.-S. (2003). Quick training of probabilistic neural nets by importance sampling. In *Artificial Intelligence and Statistics*.

Beygelzimer, A., Langford, J., Lifshits, Y., Sorkin, G. B., and Strehl, L. (2009). Conditional probability tree estimation analysis and algorithms. In *Uncertainty in Artificial Intelligence*.

Blei, D., Kucukelbir, A., and McAuliffe, J. (2017). Variational inference: A review for statisticians. *Journal of American Statistical Association*.

Blei, D. M. and Lafferty, J. D. (2007). A correlated topic model of Science. *The Annals of Applied Statistics*, 1(1):17–35.

Bouchard, G. (2007). Efficient bounds for the softmax and applications to approximate inference in hybrid models. In *Advances in Neural Information Processing Systems, Workshop on Approximate Inference in Hybrid Models*.

Dayan, P., Hinton, G. E., Neal, R. M., and Zemel, R. S. (1995). The Helmholtz machine. *Neural Computation*, 7(5):889–904.

Dembczyński, K., Cheng, W., and Hüllermeier, E. (2010). Bayes optimal multilabel classification via probabilistic classifier chains. In *International Conference on Machine Learning*.

Devlin, J., Zbib, R., Huang, Z., Lamar, T., Schwartz, R., and Makhoul, J. (2014). Fast and robust neural network joint models for statistical machine translation. In *Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*.

Gershman, S. J. and Goodman, N. D. (2014). Amortized inference in probabilistic reasoning. In *Proceedings of the Thirty-Sixth Annual Conference of the Cognitive Science Society*.

Girolami, M. and Rogers, S. (2006). Variational Bayesian multinomial probit regression with Gaussian process priors. *Neural Computation*, 18(8):1790–1817.

Gopal, S. and Yang, Y. (2013). Distributed training of large-scale logistic models. In *International Conference on Machine Learning*.

Grave, E., Joulin, A., Cissé, M., Grangier, D., and Jégrou, H. (2017). Efficient softmax approximation for GPUs. In *arXiv:1609.04309*.

Gupta, M. R., Bengio, S., and Jason, W. (2014). Training highly multiclass classifiers. *Journal of Machine Learning Research*, 15(1):1461–1492.

Gutmann, M. and Hyvärinen, A. (2010). Noise-contrastive estimation: A new estimation principle for unnormalized statistical models. In *Artificial Intelligence and Statistics*.

Hoffman, M. D., Blei, D. M., Wang, C., and Paisley, J. (2013). Stochastic variational inference. *Journal of Machine Learning Research*, 14:1303–1347.

Ji, S., Vishwanathan, S. V. N., Satish, N., Anderson, M. J., and Dubey, P. (2016). Blackout: Speeding up recurrent neural network language models with very large vocabularies. In *International Conference on Learning Representations*.

Jordan, M. I., Ghahramani, Z., Jaakkola, T. S., and Saul, L. K. (1999). An introduction to variational methods for graphical models. *Machine Learning*, 37(2):183–233.

Khan, M. E., Mohamed, S., Marlin, B. M., and Murphy, K. P. (2012). A stick-breaking likelihood for categorical data analysis with latent Gaussian models. In *Artificial Intelligence and Statistics*.

Kingma, D. P. and Welling, M. (2014). Auto-encoding variational Bayes. In *International Conference on Learning Representations*.

Kurzynski, M. (1988). On the multistage Bayes classifier. *Pattern Recognition*, 21(4):355–365.

Linderman, S., Johnson, M. J., and Adams, R. (2015). Dependent multinomial models made easy: Stick-breaking with the Polya-gamma augmentation. In *Advances in Neural Information Processing Systems*.

McFadden, D. (1978). Modeling the choice of residential location. In *Spatial Interaction Theory and Residential Location*.

McFadden, D. and Train, K. (2000). Mixed MNL models for discrete response. *Journal of Applied Econometrics*, 15(5):447–470.

Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., and Dean, J. (2013). Distributed representations of words and phrases and their compositionality. In *Advances in Neural Information Processing Systems*.

Mnih, A. and Gregor, K. (2014). Neural variational inference and learning in belief networks. In *International Conference on Machine Learning*.

Morin, F. and Bengio, Y. (2005). Hierarchical probabilistic neural network language model. In *Artificial Intelligence and Statistics*.

Mussmann, S., Levy, D., and Ermon, S. (2017). Fast amortized inference and learning in log-linear models with randomly perturbed nearest neighbor search. In *Uncertainty in Artificial Intelligence*.

Rezende, D. J., Mohamed, S., and Wierstra, D. (2014). Stochastic backpropagation and approximate inference in deep generative models. In *International Conference on Machine Learning*.

Smith, N. A. and Jason, E. (2005). Contrastive estimation: Training log-linear models on unlabeled data. In *Association for Computational Linguistics*.

Titsias, M. K. (2016). One-vs-each approximation to softmax for scalable estimation of probabilities. In *Advances in Neural Information Processing Systems*.

Titsias, M. K. and Lázaro-Gredilla, M. (2014). Doubly stochastic variational Bayes for non-conjugate inference. In *International Conference on Machine Learning*.

Tsoumakas, G., Katakis, I., and Vlahavas, I. (2008). Effective and efficient multilabel classification in domains with large number of labels. In *ECML/PKDD Workshop on Mining Multidimensional Data*.

Vijayanarasimhan, S., Shlens, J., Monga, R., and Yagnik, J. (2014). Deep networks with large output spaces. In *arXiv:1412.7479*.