VARIATIONAL STOCHASTIC GRADIENT DESCENT

Michael Tetelman
InvenSense, Inc
michael.tetelman@gmail.com

ABSTRACT

In Bayesian approach to probabilistic modeling of data we select a model for probabilities of data that depends on a continuous vector of parameters. For a given data set Bayesian theorem gives a probability distribution of the model parameters. Then the inference of outcomes and probabilities of new data could be found by averaging over the parameter distribution of the model, which is an intractable problem. In this paper we propose to use Variational Bayes (VB) to estimate Gaussian posterior of model parameters for a given Gaussian prior and Bayesian updates in a form that resembles SGD rules. It is shown that with incremental updates of posteriors for a selected sequence of data points and a given number of iterations the variational approximations are defined by a trajectory in space of Gaussian parameters, which depends on a starting point defined by priors of the parameter distribution, which are true hyper-parameters. The same priors are providing a weight decay or L2 regularization for the training. Then a selection of L2 regularization parameters and a number of iterations is completely defining a learning rule for VB SGD optimization, unlike other methods with momentum (Duchi et al., 2011; Kingma & Ba, 2014; Zeiler, 2012) that need selecting learning, regularization rates, etc., separately. We consider application of VB SGD for important practical case of fast training neural networks with very large data. While the speedup is achieved by partitioning data and training in parallel the resulting set of solutions obtained with VB SGD forms a Gaussian mixture. By applying VB SGD optimization to the Gaussian mixture we can merge multiple neural networks of same dimensions into a new single neural network that has almost the same performance as an original Gaussian mixture.

1 BAYESIAN METHOD

In Bayesian approach to probabilistic modeling of data we select a family of models for probabilities of data that generally depends on a continuous vector of parameters (MacKay, 1995; Bishop, 1995).

Let $P_1(y|\vec{x},\vec{w})$ be a conditional probability of label y given input vector $\vec{x}$ that depends on a vector of continuous parameters $\vec{w}$.

Then for an observed data given as pairs $\{\vec{x}_t, y_t\}, t = 1 \dots T$, the Bayesian theorem defines a probability distribution of model parameters:

$$Prob(\vec{w}) \propto P_0(\vec{w}) \prod_{t=1}^{T} P_1(y_t|\vec{x}_t,\vec{w}). \tag{1}$$

Here, $P_0(\vec{w})$ is a prior probability distribution of model parameters $\vec{w}$.

With Bayesian method the inference of outcomes, probabilities of new data and other values of interest could be found by computing averages with the parameter distribution of the model. For example, the probability of a label y given a new never observed input $\vec{x}$ is obtained by the following expression:

$$Prob(y|\vec{x}) = \int d\vec{w} P_1(y|\vec{x},\vec{w}) \left(P_0(\vec{w}) \prod_{t=1}^{T} P_1(y_t|\vec{x}_t,\vec{w}) \right) / \int d\vec{w} \left(P_0(\vec{w}) \prod_{t=1}^{T} P_1(y_t|\vec{x}_t,\vec{w}) \right)$$

However, computing Bayesian integrals over parameters $\vec{w}$ with the parameter distribution above is a difficult problem. A standard approach is to find a single point $\vec{w}_0$ in $w-$parameter space - a maximum of the parameter distribution. With this maximum likelihood method a parameter distribution is simplified to become a delta-function

$$Prob(\vec{w}) = \delta(\vec{w} - \vec{w}_0), \vec{w}_0 = \arg\max_{\vec{w}} \left(P_0(\vec{w}) \prod_{t=1}^{T} P_1(y_t|\vec{x}_t,\vec{w} \right).$$

Variational Bayes method allows to obtain an approximation of the probability distribution over parameters in a form that could make possible computing of the integrals (Bishop, 2006).

With VB we can find distributions that are less trivial than a delta-function and still manageable to compute the averages of interests.

In this paper we propose to use Variational Bayes to estimate Gaussian posterior of parameters for a given Gaussian prior and Bayesian updates with a given model of data.

To do that we will use the following trick and Jensen's inequality for average of exponential to transform the Bayesian integral with some probability $P(\vec{w})$ to a better form:

$$\int d\vec{w} P(\vec{w}) = \int d\vec{w} Q(\vec{w}|\phi) \frac{P(\vec{w})}{Q(\vec{w}|\phi)} \geq \exp\left(\int d\vec{w} Q(\vec{w}|\phi) \ln \frac{P(\vec{w})}{Q(\vec{w}|\phi)} \right). \quad (2)$$

Here, a new probability distribution $Q(\vec{w}|\phi)$ is a variational approximation for probability distribution $P(\vec{w})$. The distribution $Q(\vec{w}|\phi)$ depends on a set of parameters ϕ. By maximizing the integral on right side of the equation above over parameters ϕ we can find the distribution $Q(\vec{w}|\phi)$ that is a best approximation for $P(\vec{w})$. The right side of eq.2 contains a negative of a well-known Kullback-Leibler (KL) divergence for distributions Q and P. So the best Q is the one that minimizes KL-divergence in eq.2.

2 VARIATIONAL BAYES SGD

We will consider a distribution $Q(\vec{w})$ that is a product of Gaussian distributions for all components of vector $\vec{w}$:

$$Q(\vec{w}|\vec{\mu},\vec{\sigma}) = \prod_i \frac{e^{-\frac{(w_i-\mu_i)^2}{2\sigma_i^2}}}{\sqrt{2\pi\sigma_i^2}}$$

to approximate the distribution $Prob(\vec{w})$ in eq.1.

A direct computing of integral in KL-divergence with Gaussian distribution $Q(w)$ and $Prob(w)$ is still a difficult problem. This problem could be solved with the following iterative approach.

The distribution $Prob(\vec{w})$ in eq.1 consists of a product of prior distribution for $\vec{w}$ and probabilities of observed data points up to some normalization constant. We can consider an effect of observed data as a Bayesian update of the prior distribution $P_0(\vec{w})$ to the posterior distribution $Q(\vec{w})$. To make this update accurate we can do it incrementally in N iterations by using a fraction of a data point contribution at the time.

Let's use a Gaussian prior $P_0(\vec{w})$. Then it is equal to $Q_0(\vec{w}) = Q(\vec{w}|\vec{\mu_0},\vec{\sigma_0})$ for some parameters $(\vec{\mu_0},\vec{\sigma_0})$.

Because for large enough N the contribution of data in eq.1 can be represented as a product of factors where each factor is close to 1

$$Prob(\vec{w}) \propto Q_0(\vec{w}) \left[\prod_{t=1}^{T} P_1(y_t|\vec{x}_t, \vec{w})^{\frac{1}{N}} \right]^N$$

we can replace $Q_0(\vec{w})P_1(y_t|\vec{x}_t, \vec{w})^{1/N}$ with $Q_1(\vec{w})$, where $Q_1(\vec{w})$ minimizes KL-divergence

$$Q_{1,t}(\vec{w}) = Q(\vec{w})|\vec{\mu}_{1,t}, \vec{\sigma}_{1,t}), (\vec{\mu}_{1,t}, \vec{\sigma}_{1,t}) = \arg\max_{\vec{\mu},\vec{\sigma}} \int d\vec{w} Q(\vec{w}|\vec{\mu}, \vec{\sigma}) \ln \frac{Q_0(\vec{w})[P_1(y_t|\vec{x}_t, \vec{w})]^{\frac{1}{N}}}{Q(\vec{w}|\vec{\mu}, \vec{\sigma})} \tag{3}$$

$Q_1(\vec{w})$ is a Bayesian update of prior $Q_0(\vec{w})$ from a $1/N$-fraction of a data point t. By repeating these Bayesian updates for each data point and iteration n we will find a sequence of approximations

$$Q_n(\vec{w}) \to Q_{n+1}(\vec{w}), Q_{n+1}(\vec{w}) = Q(\vec{w}|\vec{\mu}_{n+1}, \vec{\sigma}_{n+1}),$$

$$(\vec{\mu}_{n+1}, \vec{\sigma}_{n+1}) = \arg\max_{\vec{\mu},\vec{\sigma}} \int d\vec{w} Q(\vec{w}|\vec{\mu}, \vec{\sigma}) \ln \frac{Q_n(\vec{w})[P_1(y_t|\vec{x}_t, \vec{w})]^{\frac{1}{N}}}{Q(\vec{w}|\vec{\mu}, \vec{\sigma})}$$

with a final $Q_N(\vec{w})$ approximating $Prob(\vec{w})$ in eq.1.

We will compute the integral above in the limit of small variances σ_i^2 by expanding $P_1(\vec{w})$ around $\vec{\mu}$ and keeping only leading terms, then

$$\int d\vec{w} Q(\vec{w}|\vec{\mu}, \vec{\sigma}) \ln P_1(\vec{w}) \approx \ln P_1(\vec{\mu}) + \sum_i \frac{1}{2} \sigma_i^2 \frac{\partial^2}{\partial w_i^2} \ln P_1(\vec{w})|_{\vec{w}=\vec{\mu}}.$$

Now, by maximizing over $\vec{\mu}$ and $\vec{\sigma}$ we can obtain the VB SGD update rules for a single data point:

$$\mu_{n+1,i} = \mu_{n,i} + \frac{\sigma_{n,i}^2}{N} \frac{\partial}{\partial w_i} \ln P_1(y_t|\vec{x}_t, \vec{w})|_{\vec{w}=\vec{\mu}_n}, \frac{1}{\sigma_{n+1,i}^2} = \frac{1}{\sigma_{n,i}^2} - \frac{1}{N} \frac{\partial^2}{\partial w_i^2} \ln P_1(y_t|\vec{x}_t, \vec{w})|_{\vec{w}=\vec{\mu}_n} \tag{4}$$

The term with second derivative in the equation 4 after iterating over a whole data set can be considered as an average over empirical distribution $q(x, y)$: $\sum_{x,y} q(x, y)\delta^2 \ln P_1(y|x, w)$. That average satisfies the following identity: $< \delta^2 \ln P >=< \delta^2 P/P > - < (\delta \ln P)^2 >$. If a model probability P is close enough to an empirical probability we can neglect a term with second derivative of P and keep only term with a square of first derivative of the log of probability.

Then finally, we have the VB SGD update rule for σ with the first order gradient:

$$\frac{1}{\sigma_{n+1,i}^2} = \frac{1}{\sigma_{n,i}^2} + \frac{1}{N} \left(\frac{\partial}{\partial w_i} \ln P_1(y_t|\vec{x}_t, \vec{w}) \right)^2 |_{\vec{w}=\vec{\mu}_n} \tag{5}$$

3 Merging Multiple Models

When training multiple models of same dimensions on different partitions of data the VB SGD gives us a Gaussian distribution for each model and a distribution of the whole ensemble is a mix of Gaussian distributions. We apply VB SGD to find a single Gaussian distribution that approximates the mix: $G_{mix}(\vec{w}) = 1/T \sum_t G_t(\vec{w})$.

The update rule is the same as in eq.4, only instead of P_1 we use ratio $G_{mix}(\vec{w})/Q(\vec{w}|\vec{\mu}_n, \vec{\sigma}_n)$.

REFERENCES

C. M. Bishop. *Neural networks for pattern recognition*. Oxford University Press, 1995.

Christopher M. Bishop. *Pattern Recognition and Machine Learning*. Information Science and Statistics. Springer, New York, 2nd edition, 2006.

John Duchi, Elad Hazan, and Yoram Singer. Adaptive subgradient methods for online learning and stochastic optimization. *JMLR*, 999999:2121–2159, 2011.

Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *CoRR*, abs/1412.6980, 2014. URL `http://arxiv.org/abs/1412.6980`.

D. J. C. MacKay. Probable networks and plausible predictions — a review of practical Bayesian methods for supervised neural networks. *Network: Computation in Neural Systems*, 6:469–505, 1995.

Matthew D. Zeiler. ADADELTA: An Adaptive Learning Rate Method. *CoRR*, abs/1212.5701, 2012.