Fast and Natural Newton method

Anil K Nelakanti
Discussion Group
Xerox Research Center Europe

A fast natural Newton method.
Nicolas Le Roux and Andrew Fitzgibbon, ICML 2010.

Learning Problems

Learning problems fit a model and use that to make inferences.
Fit a model $\mathcal{M}(\theta)$ to data D.

Retrieve fixed unknown θ given data D: frequentist

$$\mathcal{L}(\ D\ ;\ \mathcal{M}(\theta)\ )$$

Retrieve probable θ for fixed given data D: bayesian

$$\mathcal{L}(\ D \mid \mathcal{M}(\theta)\ )$$

Learning as Optimization

Consider a loss function $L(x, \theta)$ that gives the error in the model $\mathcal{M}(\theta)$ at point x. The error in the model is now

$$\mathbf{L}(\theta) = \int_x L(x, \theta) p(x) dx$$

Solve for parameters that minimize the error function.

$$\arg\min_{\theta} \mathbf{L}(\theta)$$

The problem boils down to an optimization in the parameter space.

Error Minimization

Find x_n such that

$$f(x_n) < f(x_n + \epsilon) \quad \forall \text{ small } \epsilon$$

Involves generating a sequence of iterates $\{x_k\}$ as

$$x_{k+1} := x_k + \alpha p_k$$

move α units along p_k from x_k such that

$$f(x_{k+1}) < f(x_k)$$

Continuous Optimization

1. First-order methods

   guaranteed convergence[local,global]: Q-linear.
   * Coordinate descent - descend along the axes.
   * Steepest descent - opposite to the gradient.
2. Second-order methods

   convergence[only local]: Q-quadratic.
   * Newton method - Hessian modified gradient.
   * Quasi-Newton - approx. hessian: DFP, BFGS
   * Damped Newton - guarantees conv.
     Levenberg Marquardt - sum-of-squares with zero means
   * Trust region methods.
3. Conjugate Gradients:
   * n steps in n-dim with exact arithmetic.
   * not good with noisy estimates.

Second-order offers faster convergence than first-order.

Learning as Optimization

Consider second-order minimization of

$$\arg\min_{\theta} \mathbf{L}(\theta)$$

$$\theta_{k+1} := \theta_k + \mathcal{H}^{-1} g$$

$g = \nabla \mathbf{L}$ is gradient of $\mathbf{L}$.
$\mathcal{H}^{-1}$ is the inverse of Hessian defined on the parameter space at θ giving the local curvature.

Hessian captures the information of how different the gradient would be if the position in the parameter space was slightly different.

Distribution of Gradient

Let estimated gradient $\hat{g}$ be a sampling from the distribution: a gaussian centered at the mean g with covariance C,

$$\text{Likelihood: } \hat{g}|g \sim \mathcal{N}(g, \frac{C}{n})$$

C is the actual covariance

$$C = \int_x (\frac{\partial \mathbf{L}(x,\theta)}{\partial \theta} - g)(\frac{\partial \mathbf{L}(x,\theta)}{\partial \theta} - g)^T p(x) dx$$

and is approximated by $\hat{C}$ the estimated covariance.

$$\hat{C} = \frac{1}{n} \sum_x (\frac{\partial \mathbf{L}(x,\theta)}{\partial \theta} - \hat{g})(\frac{\partial \mathbf{L}(x,\theta)}{\partial \theta} - \hat{g})^T$$

Posterior distribution of gradients

Consider a gaussian prior with variance σ for the zero centered gradient distribution:

$$\text{Prior: } g \sim \mathcal{N}(0, \sigma^2 I)$$

Posterior is proportional to likelihood times prior

$$\text{Posterior: } g|\hat{g} \sim \mathcal{N}([I + \frac{C}{n\sigma^2}]^{-1}\hat{g}, [nC^{-1} + \sigma^{-2}I]^{-1})$$

The best direction of descent is now (replacing C by $\hat{C}$)

$$\delta\theta \propto [I + \frac{\hat{C}}{n\sigma^2}]^{-1}\hat{g}$$

called the natural gradient.

Covariance matrix captures how the gradient would vary if the training data were any different.

Specifics of Learning

$$\delta\theta \propto [I + \frac{\hat{C}}{n\sigma^2}]^{-1}\hat{g}$$

There are two specific characteristics of the natural gradient that suit the learning problem.

1. Space of parameters:
   Space of parameter θ is not Euclidean but Riemannian. Its distance measure depends on the Riemannian metric.
2. Noise in estimates:
   We cannot trust our gradients to be perfect since our data is noisy. Covariance matrix captures this information.

The natural gradient meets both these requirements typical of learning problems.

Natural Gradient == Hessian ?

$$[I + \frac{\hat{C}}{n\sigma^2}]^{-1}\hat{g} \text{ vs } \mathcal{H}^{-1}\hat{g}$$

Hessian captures local curvature of the manifold at the given θ.
It is identity when the curvature of error function is spherical at given θ.

Covariance captures the effect of noise in data on the gradients.
It is identity when the axes are orthogonal $\Rightarrow$ the space of parameters is Euclidean.

They represent very different informations.
Hessian is the same as covariance when the model perfectly fits the data at the given θ.

Is Prior good enough?

Assume (as usual with second-order methods):

- the cost function is locally quadratic and
- the dependence of $\mathcal{H}$ on x is weak, then

$$g(\theta) = \int_x \frac{\partial \mathbf{L}}{\partial \theta} p(x) dx = \mathcal{H}(\theta - \theta^*)$$

θ^* being the optimal value, it is clearly dependent on $\mathcal{H}$.

Change prior! Use gaussian prior for $(\theta - \theta^*)$ instead.

$$p(\theta - \theta^*) = \mathcal{N}(0, \sigma^2 I) \text{ and } g(\theta) = \mathcal{H}(\theta - \theta^*)$$

$$\text{New Prior: } g \sim \mathcal{N}(0, \sigma^2 \mathcal{H}^2)$$

Natural + Newton

Corrected Posterior: $g|\hat{g} \sim \mathcal{N}\left(\left[I + \frac{C\mathcal{H}^{-2}}{n\sigma^2}\right]^{-1} \hat{g}, \left[\frac{\mathcal{H}^{-2}}{\sigma^2} + nC^{-1}\right]^{-1}\right)$

Replacing C by $\hat{C}$, the descent direction $\mathcal{H}^{-1}g$ comes to

$$\delta\theta \propto \left[I + \frac{\mathcal{H}^{-1}\hat{C}\mathcal{H}^{-1}}{n\sigma^2}\right]^{-1} \mathcal{H}^{-1}\hat{g} = \left[I + \frac{\hat{D}}{n\sigma^2}\right]^{-1} \hat{d}$$

- Plug in your favorite Newton method to compute $\hat{d}$.
- Use your favorite natural gradient algorithm to compute $\delta\theta$.

Covariance with Memory

For being conservative at using estimates use a weighted update:

$$\begin{aligned} \mu_n &= \frac{\sum_i \gamma^{n-1} d_i}{\sum_i \gamma_i} \\ D_n &= \frac{\sum_i \gamma^{n-1}(d-\mu_n)(d-\mu_n)^T}{\gamma_n - \frac{\sum_i \gamma^{-2i}}{\gamma_n}} \end{aligned}$$

- The smaller the value of γ the shorter the effect of an example.
- The uncentered covariance is an alternative that makes the update simpler.
- The prior of g is not updated by posterior from earlier step since their distribution changes at every step.

Parameter tuning and Practical issues

- The change in covariance matrix at each step is small; update once every few steps.
- σ^2 is replaced by the running average of squared norms $\|\mu\|^2$.
- A step further to limit the effect of covariance matrix to avoid variations is by constraining the rank of $\delta\theta$.
- As $n \to \infty$, the covariance term vanishes leaving Hessian reassuring asymptotic convergence to the actual parameters.

Thank you.

Nicolas Le Roux & Andrew Fitzgibbon.
A fast natural Newton method.
ICML, 2010.

Amari Shun-ichi.
Natural Gradient Works Efficiently in Learning.
Neural Computation, 1998.