

EM for M-Estimation

Ali Rahimi (ali@mit.edu)

May 28, 2002

Abstract

1 Introduction

Parameter estimation is the problem of fitting a function to observed data. Formally, given data $Y = \{y_i | i = 1..n\}$, we would like to find the best parameter θ such that the functions $f_i(\theta)$ are as close as possible to their corresponding y_i . Often the L2 norm is used as a measure of closeness, and the parameter estimation problem can be phrased as:

$$\theta^* = \arg \min_{\theta} \sum_i \|r_i(\theta)\|^2 = \arg \min_{\theta} \sum_i \|f_i(\theta) - y_i\|^2, \quad (1)$$

Real data is often plagued with outliers which manifest themselves as data points which do not agree with the same θ fit that is suitable for other data points. Unfortunately, these outliers have a particularly strong influence on the final choice of θ under the L2 norm. As will become clear later, this is a very sensible thing for the L2 norm to do *if the data actually come from $f_i(\theta)$ but are corrupted by white Gaussian noise*.

To diminish the influence of outliers, Huber [?] suggests using M-Estimation, which involves replacing the L2 norm with a robust error norm:

$$\theta^* = \arg \min_{\theta} \sum_i \rho[r_i(\theta)] = \sum_i \rho(f_i(\theta) - y_i). \quad (2)$$

Instead of retaining the L2 norm's quadratic growth for larger values of d_i , ρ is made to saturate as the error gets larger. This makes the fit for θ robust to outliers. Figure ?? compares various robust error norms against the L2 norm. The suggested method for minimizing this function is an iterative method such as gradient descent or Newton Raphson.

This note explores various robust error norms and justifies them as statistical generative models. These generative models yield easy to optimize problems which require little effort from the implementer. The running example is that of robust linear regression. We are given pairs of (y_i, X_i) , and are told the y_i 's are obtained by corrupting $\theta^\top X_i$ using some kind of noise.

The final example in this note is an algorithm for recovering the warping parameters between two images: Assume a function $W(x, I; \theta)$ which warps a pixel at location x with intensity I to a new position according to the warping parameter θ . Then given two images I_1 and I_2 , we would like to find the warping parameters θ which minimizes appearance difference between $I_1(x)$ and $W(I_2(x), x; \theta)$. We show how various metrics on appearance affect the final result using real imagery.

2 Justifying Error Norms

We can legitimize equation (1) as a maximum likelihood estimator (as is done in [?]). Assume a generative model for the data:

$$\begin{aligned} y_i &\sim f_i(\theta) + \omega_i \\ \omega_i &\sim \mathcal{N}(0, 1) = \alpha \exp\left(-\frac{1}{2}x^\top x\right), \end{aligned} \quad (3)$$

where $\mathcal{N}(y; \mu, \Lambda)$ is a Gaussian pdf with mean μ and covariance Λ . Each observed data point y_i is thought of as having originated from a true $f_i(\theta)$, but corrupted by white Gaussian noise. The model can be written in terms of the likelihood of θ :

$$p(Y|\theta) = \prod_i \mathcal{N}(y_i; f_i(\theta), 1).$$

The maximum likelihood θ is the one which maximizes the above function. Taking logs, multiplying by -2 , and removing terms not dependent on θ , we obtain:

$$\theta^* = \arg \max_{\theta} p(Y|\theta) = \arg \max_{\theta} \|y_i - f_i(\theta)\|^2,$$

which is the same objective as equation (1).

To justify robust error norms, let's assume the data is noise-corrupted as before, but that the ω_i have distribution

$$\omega_i \sim \exp\left(-\frac{1}{2}\rho(x)\right).$$

We can perform the same simplifications as before and find that the ML θ is the one from equation (2).

Because ρ flattens out for large deviations, the pdf for w_i has heavy tails. Figure ?? compares some common ρ 's against the L2 norm. The implicit philosophy of robust estimation is that a more accurate generative model yields more robust estimates. For example, if the data doesn't come from a Gaussian, don't use the L2 norm as the log likelihood. The ρ 's of figure ?? imply different generative models where the noise distribution has heavy tails. This paper provides several generative models to justify heavy-tailed distributions as appropriate generative models.

There are several approaches for picking generative models for regression problems. Huber [?] used an adversary model for choosing an error norm (using minimax). The resulting norm can be exponentiated and normalized to yield a generative model. Alternatively, if we have enough intuition about the underlying process, we can write down a distribution intuitively. This is the approach used in sections 6 and 7. Another approach is to plot a histogram of the residuals of a manual fit. After picking a θ , the histogram of $y_i - f_i(\theta)$ will convey the distribution of the additive noise w_i . Often, the use of a heavy-tailed distribution for regression is enough to claim robustness.

3 Finding Robust Solutions

Various authors [?, ?] recommend minimizing equation (2) using Newton-Raphson. This involves analytically computing the first and second derivatives of the objective with respect to θ . These derivatives define a quadratic form which is used as a local approximation to the objective function at each iteration.

Computing the first and second derivatives of the objective function often ends up being cumbersome. However, with an appropriate choice of ρ we may be able to simplify the optimization significantly by using Expectation Maximization instead of Newton-Raphson

The following section briefly reviews Newton-Raphson and EM.

3.1 Review of Newton-Raphson

??

$$\begin{aligned} \epsilon(\theta) &= \sum_{i=1}^N (f_i(\theta) - y_i)^2 = \sum r_i^2 \\ \nabla \epsilon(\theta) &= \sum_i r_i \nabla f_i(\theta) \\ \nabla^2 \epsilon(\theta) &= \sum_i r_i \nabla^2 f_i(\theta) + \nabla f_i(\theta) \nabla f_i(\theta)^\top \\ &\approx \sum_i \nabla f_i(\theta) \nabla f_i(\theta)^\top \end{aligned}$$

The first term of the second derivative is often dropped based on the assumption that near the peak, $f_i(\theta) - y_i$ is small.

$$\begin{aligned}\epsilon(\theta) &= \sum_{i=1}^N \rho(r_i) \\ \nabla\epsilon(\theta) &= \sum_i \rho'(r_i) \nabla f_i(\theta) \\ \nabla\epsilon^2(\theta) &= \sum_i \rho'(r_i) \nabla^2 f_i(\theta) + \rho''(r_i) \nabla f_i(\theta) \nabla f_i(\theta)^\top \\ &\approx \sum_i \rho''(r_i) \nabla f_i(\theta) \nabla f_i(\theta)^\top\end{aligned}$$

3.2 Review of EM

$$\theta^{i+1} = \arg \max_{\theta} g(\theta, q^{i+1}).$$

The maximizing q is always:

$$q^{i+1}(h) = p(h|\theta^i, y) \tag{4}$$

The terms of g relevant in maximizing over θ are:

$$\theta^{i+1} = \arg \max_{\theta} E_{q^{i+1}(h)} [\log p(y, h|\theta)] \tag{5}$$

4 Generative Model for Linear Regression

I'll use linear regression as an example of robust estimation throughout this paper. The examples model each datum y_i as emanating from a linear combination of the explanatory variables x_i and the variable of regression θ . In other words,

$$f_i(\theta) = \theta^\top x_i,$$

The following few sections explore various choices of corrupting noise and the corresponding estimation algorithms.

5 Robust Norm To Downweight Large Values

To illustrate robust regression using Newton-Raphson, we assume that each data point comes from a heavy-tailed student-t distribution with one degrees of freedom, also known as a Cauchy distribution (see figure ??):

$$p(y|\theta) = \prod_{i=1}^N \text{student-t}_1(y_i|f_i(\theta), \sigma^2) \propto \prod \frac{1}{\sigma\pi} \left(1 + \frac{[y_i - f_i(\theta)]^2}{\sigma^2}\right)^{-1}$$

To minimize function, we need to find the first and second derivatives. To simplify matters, we can equivalently look for the minimum of:

$$\epsilon(\theta) = \sum_i \ln((f_i(\theta) - y_i)^2 + \sigma^2) = \sum_i \ln \epsilon_i.$$

The derivatives of $\epsilon(\theta)$ are

$$\begin{aligned}\nabla\epsilon(\theta) &= 2 \sum_i (f_i(\theta) - y_i) \epsilon_i^{-1} \nabla f_i(\theta) \\ \nabla^2\epsilon(\theta) &= 2 \sum_i (f_i(\theta) - y_i) \epsilon_i^{-1} \nabla^2 f_i(\theta) + \epsilon_i^{-1} \nabla f_i(\theta) \nabla f_i^\top(\theta) - 2(f_i(\theta) - y_i)^2 \epsilon_i^{-2} \nabla f_i(\theta) \nabla f_i^\top(\theta) \\ &= \sum_i \epsilon_i^{-1} \nabla f_i(\theta) \nabla f_i^\top(\theta)\end{aligned}$$

Comparing these derivatives to the L2 optimization of section ?? reveals that each data point is weighted according to the square of its distance to the model. The scale factor σ^2 provides a baseline to even out or accentuate the variations in these weights.

6 Robust Norm for Tunable Noise

Finding second derivatives is often tedious, and Newton-Raphson may have poor convergence properties on some problems. In this section, I justify the Cauchy robust norm by deriving it from an equivalent generative model. The resulting likelihood function is minimized using EM, which yields a reweighted least squares strategy where the weight of each point diminishes roughly according to the square of its distance to the model fit.

Suppose you are given a sequences of data points corrupted by white Gaussian noise, but are told that the variance of the noise w_i varies from datum to datum. The variance of w_i is chosen independently for each sample. If the variance were given for each datum, we could solve the estimation problem using non-linear weighted least squares. But since we don't, we have to get clever.

Let's assume that the variance of each data point is chosen from an inverse χ^2 distribution. The pdf of an Inv- χ^2 with ν degrees of freedom and spread s^2 is:

$$\chi^{-2}(\sigma^2|\nu, s^2) = \frac{(\nu s^2/2)^{(\nu/2)}}{\Gamma(\nu/2)} (\sigma^2)^{-(\nu/2+1)} e^{-\nu s^2/(2\sigma^2)}.$$

Then each datum is drawn from

$$p(y_i|x_i, \theta, \sigma^2) = \mathcal{N}(y_i|f_i(\theta), \sigma^2).$$

The joint distribution of the y 's and the variances, conditioned on the mean parameter and the hyperparameters for the variances is:

$$p(\sigma^2, y|\nu, s^2, \theta, x) = \prod_i p(y_i|\theta, x_i, \sigma_i^2) p(\sigma_i^2|\nu, s^2).$$

We would like to find the best θ and the prototypical variance, s^2 . Since the variances σ_i^2 aren't observed, we must integrate them out in the objective function:

$$(\theta^*, s^{2*}) = \arg \max_{\theta, s^2} \ln \int_{\sigma^2} p(\sigma^2, y|\nu, s^2, \theta, x) = \arg \max_{\theta, s^2} \sum_{i=1}^N \ln \int_{\sigma_i^2} p(y_i|\theta, x_i, \sigma_i^2) p(\sigma_i^2|\nu, s^2).$$

According to (4), the E-step must compute

$$q(\sigma^2) = \prod_i q_i(\sigma_i^2) = \prod_i p(\sigma_i^2|\theta^{old}, v^{old}, s^{2old}, y_i, x_i),$$

and the M-step, as dictated by (5), must maximize

$$\arg \max_{\theta, s^2} E_{\sigma^2} \left[\ln \prod_{i=1}^N \mathcal{N}(y_i|x_i, \theta, \sigma_i^2) \chi^{-2}(\sigma_i^2|\nu, s^2) \middle| \theta^{old}, v^{old}, s^{2old}, x \right]. \quad (6)$$

where the expectation is over $q(\sigma^2)$.

6.1 E Step

We would like to find

$$q_i(\sigma_i^2) = p(\sigma_i^2|\theta^{old}, v^{old}, s^{2old}, x_i, y_i) \propto \mathcal{N}(y_i|f_i(\theta), \sigma_i^2) \chi^{-2}(\sigma_i^2|\nu, s^2)$$

Expanding and recollecting suggestively, we obtain:

$$q_i(\sigma_i^2) \propto \exp\left(-\frac{[y_i - f_i(\theta)]^2 + \nu s^2}{2\sigma_i^2}\right) (\sigma_i^2)^{-\nu/2+3/2},$$

which is a scaled inverse χ^2 distribution with $\nu+1$ degrees of freedom and scale parameter equal to the new spread:

$$q_i(\sigma_i^2) = \chi^{-2}\left(\sigma_i^2|\nu+1, \frac{[y_i - f_i(\theta)]^2 + \nu s^2}{\nu+1}\right).$$

It will turn out later that the M-step will require the expectation $E1/\sigma_i^2$. Lets compute it for an inverse χ^2 distribution with ν degrees of freedom and spread s^2 :

$$\int_0^\infty \frac{1}{\sigma^2} \chi^{-2}(\sigma^2|\nu, s^2) d\sigma^2 = \frac{(\nu/2)^{(\nu/2)}}{\Gamma(\nu/2)} s^\nu \int_0^\infty (\sigma^2)^{-(\nu/2+2)} \exp\left(-\frac{\nu s^2}{2\sigma^2}\right) d\sigma^2$$

The integrand is a scaled inverse Gamma distribution with $\alpha = \nu/2 + 1$ and $\beta = \nu s^2/2$, the normalizing factor for which is $\beta^\alpha/\Gamma(\alpha)$. So the expectation becomes:

$$\frac{(\nu/2)^{(\nu/2)}}{\Gamma(\nu/2)} s^\nu \frac{\Gamma(\nu/2 + 1)}{(\nu/2)^{(\nu/2+2)} (s^2)^{(\nu/2+1)}} = s^{-2}.$$

The expectation of σ_i^{-2} under q_i is therefore:

$$E\sigma_i^{-2} = W_i = \frac{\nu + 1}{[y_i - f_i(\theta)]^2 + \nu s^2} \quad (7)$$

W_i can be interpreted as a shrunk inverse deviation: $[y_i - f_i(\theta)]^2$ is the deviation of the datum from the current model. Adding νs^2 and dividing $\nu + 1$ averages this spread with the prior spread dictated by the hyperparameters. ν determines how much influence the prior has in this shrinking.

6.2 M Step

We can split up (6) into two parts, each depending on a subset of the parameters of interest:

$$\arg \max_{\theta, s^2} E_{\sigma^2} \left[\sum_{i=1}^N \ln \mathcal{N}(y_i|x_i, \theta, \sigma_i^2) \right] + E_{\sigma^2} \left[\sum_{i=1}^N \ln \chi^{-2}(\sigma_i^2|\nu, s^2) \right] \quad (8)$$

Let's maximize over θ first. After dropping the second expectation and all other irrelevant terms:

$$\begin{aligned} \theta^* &= \arg \min_{\theta} \sum_{i=1}^N \int \frac{[y_i - f_i(\theta)]^2}{\sigma_i^2} q_i(\sigma_i^2) d\sigma_i^2 \\ &= \arg \min_{\theta} \sum_{i=1}^N [y_i - f_i(\theta)]^2 W_i. \end{aligned} \quad (9)$$

The maximizing θ minimizes the weighted least squares problem where each datum is weighted by $W_i = E\sigma_i^{-2}$. Because W_i is a measure of the deviation from the model, points which are far from the model are weighted less than points which are close.

To find the most likely variance hyperparameter, observe the second expectation in (8). After dropping irrelevant terms,

$$s^{2*} = \arg \max_{s^2} \sum_i \int \left[\frac{\nu}{2} \ln(s^2) - \frac{\nu s^2}{2} \right] q_i(\sigma_i^2) d\sigma_i^2$$

After dropping constants and differentiating with respect to s^2 , solving for s^2 after equating the result with 0, we obtain:

$$s^{2*} = N \left(\sum_{i=1}^N W_i \right)^{-1}$$

Recall that W_i is a shrunken inverse deviation. The best s^2 is the reciprocal of the sum of the inverse deviations.

6.3 Summary

The best θ is found by minimizing (8), which is a weighted non-linear least-squares problem. The weight of each points is the average between the expected spread (as determined by s^{2*}) and the deviation of the point from the model (which is just $[y_i - f_i(\theta)]^2$). Once the new θ is found, the spread s^{2*} is updated so as to better influence future points. If the shrinking towards a common spread is undesirable, ν can be set to a small value. That way each point will determine its weight independently of all others.

7 Robust Norm For Modeling Inliers and Outliers

This section proposes a ρ based on a mixture model. For each datum, we assume a coin is flipped. If the coin lands head, the emitted value is a white Gaussian noise corrupted $f_i(\theta)$. If the coin lands tails, we emit garbage:

$$p(y_i|\theta) = \mathcal{N}(y_i|f_i(\theta), \Lambda)p(h=1) + p_0(y_i)p(h=0), \quad (10)$$

where $p(h=0)$ is the probability that garbage is assigned to a datum. The distribution p_0 captures our knowledge about the garbage noise process. For simplicity, one might set it to be uniform over the range of allowable values for y_i .

Equation (10) replaces the heavy-tailed noise corrupting model of equation (3) with a more versatile noise model where outliers do not translate with the prediction. Figure ?? compares this robust norm against other heavy-tailed distributions. The corresponding ρ prescribed by equation (10) is:

$$\rho = \log \left(\exp \left(-\frac{1}{2}x^\top \Lambda^{-1}x \right) + |\Lambda|p_0 \right),$$

where we have assumed that $p_0(y_i) = p_0$ is constant. Figure ?? compares this norm with other popular norms.

Since we don't know which data reflect θ and which are garbage, we integrate over h for each data point. We'd like to find the θ which maximizes

$$\theta^* = \arg \max_{\theta} \prod_x p(y_i|\theta) \quad (11)$$

The use of EM as a maximization strategy reveals a reweighted least-squares strategy, but with weights which are easier to compute.

7.1 E Step

q in equation (4) can be found by normalizing:

$$q_i(h_i) = p(h_i|\theta^{old}, y_i, x_i) = \frac{p(y_i|\theta, x_i, h_i)p(h_i|\theta^{old})}{\sum_i p(y_i|\theta, x_i, h_i=i)}$$

This is the posterior probability that a datum is good, given our current estimate of θ . For the generative model of this section:

$$q^{i+1}(h(x)=1) = \frac{\mathcal{N}(I_2(x); I_1(x + u(x; \theta^i)), \Lambda)}{\mathcal{N}(I_2(x); I_1(x + u(x; \theta^i)), \Lambda) + p_0(I_2(x))},$$

the complement of which is $q(h(x)=0)$. Note that there is one q per pixel. In mixture models such as this one, q is a soft assignment of each pixel to a cluster. In this case, $q(h(x)=1)$ is the probability that the pixel at location x is due to motion, and $q(h(x)=0)$ is the probability that it was generated by the clutter.

The optimization (5) for θ can be expressed as:

$$\theta^{i+1} = \arg \max_{\theta} \sum_x q^{i+1}(h(x)=1) \log \mathcal{N}(I_1(x + u(x; \theta)), \Lambda) + \sum_x q^{i+1}(h(x)=0) \log p_0(I_2(x))$$

Only the first term is relevant in this optimization:

$$\theta^{i+1} = \arg \min_{\theta} \sum_x q^{i+1}(h(x)=1) \|I_2(x) - I_1(x + u(x; \theta))\|_{\Lambda}^2. \quad (12)$$

Where $\|x - x_0\|_{\Lambda}^2$ is the Mahalanobis distance of x from x_0 . Let d denote the reconstruction error image

$$d^i(x) = I_2(x) - I_1(x + u(x; \theta^i)),$$

and substitute for q in (12) to get the final optimization criterion:

$$\theta^{i+1} = \arg \min_{\theta} \sum_x \frac{\exp \left(-\frac{1}{2} \|d^i(x)\|_{\Lambda}^2 \right)}{\exp \left(-\frac{1}{2} \|d^i(x)\|_{\Lambda}^2 \right) + |\Lambda|^{-1}c} \|d(x)\|_{\Lambda}^2, \quad (13)$$

where c is $p_0(I_2(x))$. Note that the only term which is a function of the current θ here is inside $d(x)$, and that $d^i(x)$ depend on the previous iterate of θ instead.

8 Generative Model for Parametric Motion Estimation

In keeping with the previous section, let's assume the following generative model for our running example:

$$I_2(x) = I_1(x + u(x; \theta)) + \omega(x), \quad (14)$$

where $\omega(x)$ are iid zero mean and unit covariance Gaussians. We've limited the flexibility of the initial problem definition somewhat, so that the warping function u can now only move pixels around, whereas before, it could also change their intensity values. This model says that image I_2 is generated by warping image I_1 and adding noise to the resulting image.

Rewriting this model in terms of probability distributions, and simplifying as per the previous section, yields the objective:

$$\begin{aligned} \theta^* &= \arg \max_{\theta} \prod_x p(I_2(x) | \theta, I_1) \\ &= \arg \min_{\theta} \sum_x \|I_2(x) - I_1(x + u(x; \theta))\|^2, \end{aligned}$$

or with a robust model for $\omega(x)$,

$$\theta^* = \arg \min_{\theta} \sum_x \rho [I_2(x) - I_1(x + u(x; \theta))]$$

References

- [1] T.P. Minka. Expectation-maximization as lower bound maximization. Technical report, Media Lab, <http://www.media.mit.edu/~tpminka/papers/em.html>, 2001.