QUADRATIC MAJORIZERS FOR LOW-DEGREE POLYNOMIALS

JAN DE LEEUW

ABSTRACT. Recent theory for sharp and uniform quadratic majorization on the line is applied to polynomials, specifically to quartics. In cases where quadratic majorization is impossible, we introduce the new concept of quadratic D-approximation. We hope our results for low-order polynomials illustrate the basic concepts, and will eventually be generalized to more complicated multivariate functions.

1. INTRODUCTION

Suppose we want to minimize a function f over a set X and our current best approximation to the minimizer is $y \in X$. A *majorizer* of f in y is a function g such that $f(y) = g(y)$ and $f(x) \leq g(x)$ for all $x \in X$. Thus g is not below f on X and g touches f in y. To emphasize that g is a function of x that majorizes f in y we write $g(x|y)$.

In a step of the *majorization algorithm* we update the minimizer by minimizing a majorizer g over x. Suppose the minimum is attained in $\hat{x}$. Then $f(\hat{x}) \leq g(\hat{x}|y)$ because g is a majorizer of f, and $g(\hat{x}|y) \leq g(y|y)$ because $\hat{x}$ is a minimizer of g over X. Moreover $g(y|y) = f(y)$ because the functions g and f touch in y. Thus

$$f(\hat{x}) \leq g(\hat{x}|y) \leq g(y|y) = f(y) \tag{1}$$

Date: December 7, 2006.

2000 *Mathematics Subject Classification.* 29M20.

Key words and phrases. Successive Approximation, Iterative Majorization, Convexity.

and minimizing the majorizer decreasing the value of the objective function. Equation (1) is known as the *sandwich inequality*.

Iterative majorization algorithms, which construct majorizers at the current best point and then minimize them, only make sense if minimizing the majorizers is much simpler than minimizing the original objective function. In De Leeuw and Lange [2006] quadratic majorizers are studied, in the unconstrained and unidimensional case in which X is the real line.

A quadratic g that touches f at y is of the form $g(x|y) = f(y) + b(x-y) + \frac{1}{2}a(x-y)^2$. If f is differentiable, then it follows that $b = f'(y)$. For twice-differentiable f in addition $a \geq f''(y)$. Thus quadratic majorizers in the twice-differentiable case are necessarily of the form

$$g(x|y) = f(y) + f'(y)(x-y) + \frac{1}{2}a(x-y)^2, \tag{2}$$

with $a \geq f''(y)$.

We arrive at a necessary and sufficient condition by defining

$$\delta(x|y) = \frac{f(x) - f(y) - f'(y)(x-y)}{\frac{1}{2}(x-y)^2}. \tag{3}$$

Then (2) defines a quadratic majorizer at y if and only if $a \geq \delta(x|y$ for all x. Define

$$\alpha(y) = \sup_x \delta(x|y).$$

Then quadratic majorizers exist if and only if $\alpha(y) < +\infty$. In that case the quadratic

$$g(x|y) = f(y) + f'(y)(x-y) + \frac{1}{2}\alpha(y)(x-y)^2$$

is the *sharp quadratic majorizer* or SQM.

If there is a real number $\overline{\alpha}$ such that $f''(x) \leq \overline{\alpha}$ for all x, then clearly $\overline{\alpha} \geq \alpha(y)$ and the quadratic

$$g(x|y) = f(y) + f'(y)(x-y) + \frac{1}{2}\overline{\alpha}(x-y)^2$$

is the *uniform quadratic majorizer* or UQM.

In any case, the resulting quadratic majorization algorithm is of the form

$$x^{(k+1)} = x^{(k)} - \frac{1}{\alpha^{(k)}} f'(x^{(k)}),$$

where $\alpha^{(k)} = \alpha(x^{(k)})$ for SQM and $\alpha^{(k)} = \overline{\alpha}$ for UQM.

2. SQM's for Polynomials

In this short note we apply the theory of UQM and SQM to univariate polynomials, in particular cubics and quartics. Of course minimizing univariate polynomials is easy to do, and there is no need to apply majorization. We merely use the example to illustrate the concepts we discussed in the introduction.

Suppose $f(x) = \sum_{s=0}^{p} \alpha_s x^s$ is any polynomial. We can write f in the form

$$f(x) = \sum_{s=0}^{p} \frac{1}{s!} f^{(s)}(y)(x-y)^s.$$

By using the definition (3) we see that

$$\delta(x|y) = f^{(2)}(y) + 2\sum_{s=1}^{p-2} \frac{1}{(s+2)!} f^{(s+2)}(y)(x-y)^s. \tag{4}$$

Result (4) is new, and was not given in De Leeuw and Lange [2006]. It is actually more general than it seems, because it can easily be extended to convergent power series.

It follows that for a cubic function f the function $\delta(x|y)$ is linear in x, and there is no quadratic majorizer. For a quartic f with leading coefficient p we see that $\delta(x|y)$ is a convex quadratic if $p > 0$, which means it is unbounded above and no quadratic majorizer exists. If $p < 0$, however, the quadratic is concave and has a maximum, so we can find the sharp quadratic majorizer (in closed form). It is possible, however, that the sharp quadratic majorizer is concave.

3. Numerical Example

Consider the quartic $f(x) = -x^4 - 3x^3 + x^2 + x + 1$. The function (RED) and its first (BLUE) and second (GREEN) derivatives are plotted in Figure 1.

[Figure 1 about here.]

Table 1 shows that the function has two local maxima and one local minimum.

[Table 1 about here.]

The upper bound for the second derivative is 8.75. Thus the majorization algorithm with uniform quadratic majorization will converge to the local minimum with speed $1 - 5.75/8.75 \approx 0.3428571$.

In Figure 2 we show quadratic majorizers at $y = -1$ and $y = +1$. The uniform majorizer is in BLUE, the sharp majorizer in GREEN. For the sharp majorizer the second derivative bound is 8.50 at -1 and -3.50 at +1. At the local minimum the convergence speed of sharp majorization, with a second derivative bound of 7.75, is $1 - 5.75/7.75 \approx 0.2580645$. Observe that, as shown in Van Ruitenburg [2005] and De Leeuw and Lange [2006] the SQM g has two support points, i.e. it touches the function f in exactly two points.

[Figure 2 about here.]

Observe that at $y = +1$ the sharp quadratic majorizer is concave. This means that the algorithm will stop in one iteration, correctly concluding that f is unbounded below. The example also shows that concave majorizers can be useful in univariate minimization, contrary to what De Leeuw and Lange [2006] say. The algorithm based on uniform quadratic approximation will eventually come to the same conclusion, but more slowly. In general if we start close enough to $-\frac{1}{4}$ both the SQM and the UQM algorithms will converge quickly, in about 10-15 iterations. SQM uses two or three fewer iterations, but of course each iteration involves more work.

If we start further away, the algorithms are usually very different. Figure 3 show what happens if we start at $y = -2$. SQM moves the iteration to 0.8 and then in the next iteration dives over the edge and stops at infinity. UQM moves to -1.20 and continues to converge smoothly to the local minimum at -0.25.

[Figure 3 about here.]

Appendix A. Code

In this appendix we give R functions for UQM and SQM applied to quartic polynomials (with a negative leading coefficient). The code heavily relies on Kurt Hornik's polynom package.

```
library(polynom)

uqm<-function(x,y,eps=1e-6,itmax=100,verbose=TRUE) {
f0<-polynomial(x)
f1<-deriv(f0)
f2<-deriv(f1)
f3<-deriv(f2)
f4<-deriv(f3)
bu<-predict(f2,solve(f3))
xold<-y; fold<-predict(f0,y); gold<-predict(f1,y); itel=1
repeat {
        xnew<-xold-gold/bu; fnew<-predict(f0,xnew); gnew<-predict(f1,xnew)
        if (verbose)
                cat("Iteration: ",formatC(itel,digits=6,width=6),
                "xo: ",formatC(xold,digits=6,width=12,format="f"),
                "xn: ",formatC(xnew,digits=6,width=12,format="f"),
                "fo: ",formatC(fold,digits=6,width=12,format="f"),
                "fn: ",formatC(fnew,digits=6,width=12,format="f"),
                "go: ",formatC(gold,digits=6,width=12,format="f"),
                "gn: ",formatC(gnew,digits=6,width=12,format="f"),
                "\n")
```

```
        if (((fold - fnew) < eps) || (itel == itmax))
            break()
        xold<-xnew; fold<-fnew; gold<-gnew; itel<-itel+1
        }
return(list(x=xnew,f=fnew,g=gnew))
}

sqm<-function(x,y,eps=1e-6,itmax=100,verbose=TRUE) {
f0<-polynomial(x)
f1<-deriv(f0)
f2<-deriv(f1)
f3<-deriv(f2)
f4<-deriv(f3)
bu<-predict(f2,solve(f3))
xold<-y; fold<-predict(f0,y); gold<-predict(f1,y); itel=1
repeat {
        y2<-predict(f2,xold)
        y3<-predict(f3,xold)
        y4<-predict(f4,xold)
        c0<-y2+(y4*(xold^2)/12)-(y3*xold/3)
        c1<-(y3/3)-(y4*xold/6)
        c2<-y4/12
        dd<-polynomial(c(c0,c1,c2))
        de<-deriv(dd)
        by<-predict(dd,solve(de))
        xnew<-xold-gold/by; fnew<-predict(f0,xnew); gnew
            <-predict(f1,xnew)
        if (verbose)
                cat("Iteration:␣",formatC(itel,digits=6,
                    width=6),
                "xo:␣",formatC(xold,digits=6,width=12,
                    format="f"),
                "xn:␣",formatC(xnew,digits=6,width=12,
                    format="f"),
```

```
                    "fo: ",formatC(fold, digits=6,width=12,
                        format="f"),
                    "fn: ",formatC(fnew, digits=6,width=12,
                        format="f"),
                    "go: ",formatC(gold, digits=6,width=12,
                        format="f"),
                    "gn: ",formatC(gnew, digits=6,width=12,
                        format="f"),
                    "\n")
        if (((fold - fnew) < eps) || (itel == itmax))
            break()
        xold<-xnew; fold<-fnew; gold<-gnew; itel<-itel+1
        }
return(list(x=xnew, f=fnew, g=gnew))
}
```

References

J. De Leeuw and K. Lange. Sharp Quadratic Majorization in One Dimension. Preprint 503, UCLA Department of Statistics, 2006. URL http://preprints.stat.ucla.edu/503/sharp_third.pdf.

J. Van Ruitenburg. Algorithms for Parameter Estimation in the Rasch Model. Measurement and Research Department Reports 2005-04, CITO, Arnhem, Netherlands, 2005.

Department of Statistics, University of California, Los Angeles, CA 90095-1554

E-mail address, Jan de Leeuw: deleeuw@stat.ucla.edu

URL, Jan de Leeuw: http://gifi.stat.ucla.edu

List of Figures

1 Quartic Example 10
2 Uniform and Sharp Majorization 11
3 First Step from $y = -2$ 12

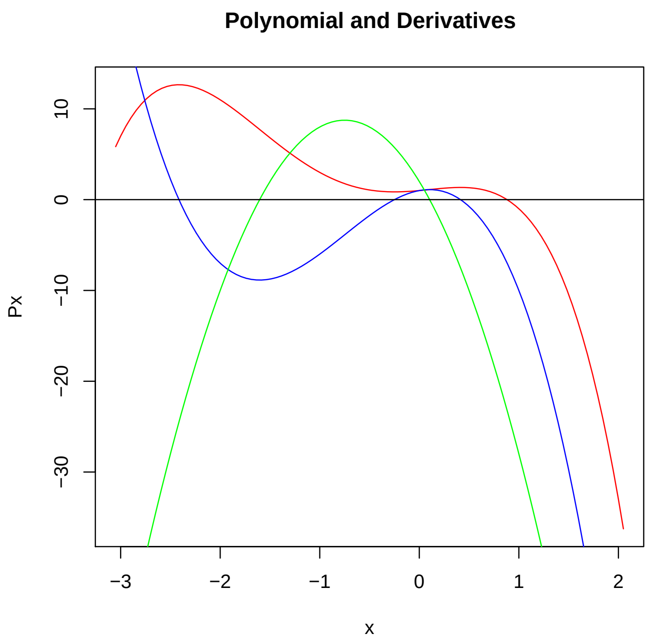

FIGURE 1. Quartic Example

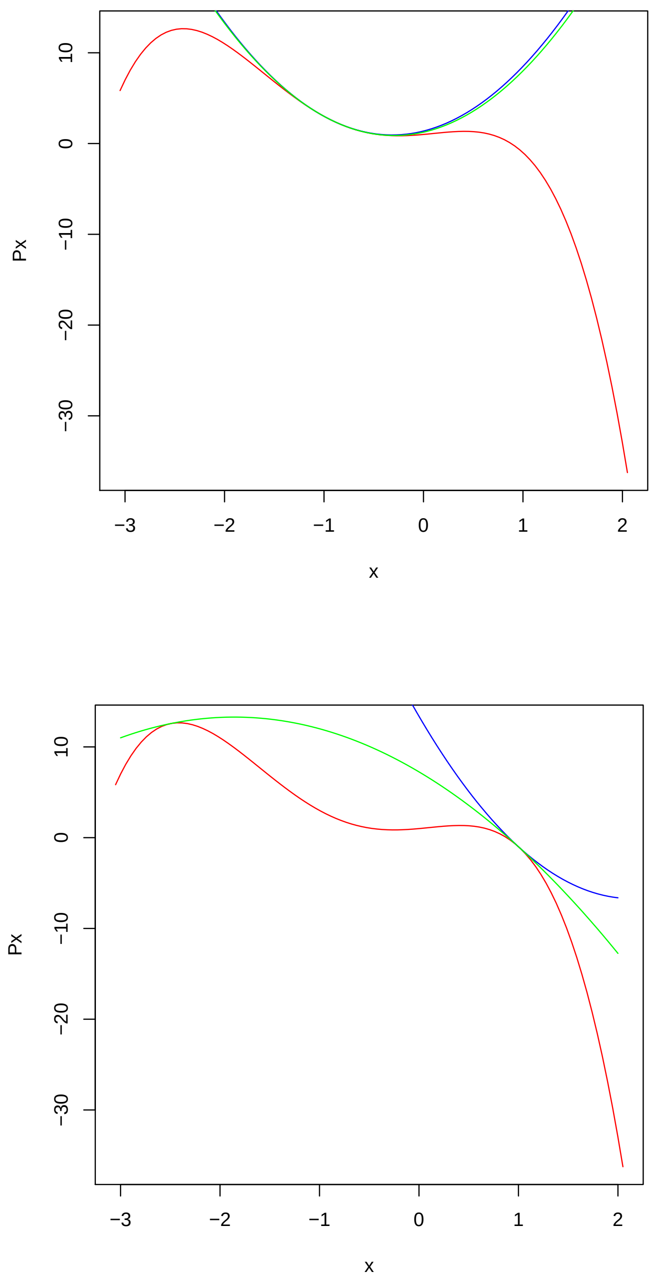

FIGURE 2. Uniform and Sharp Majorization

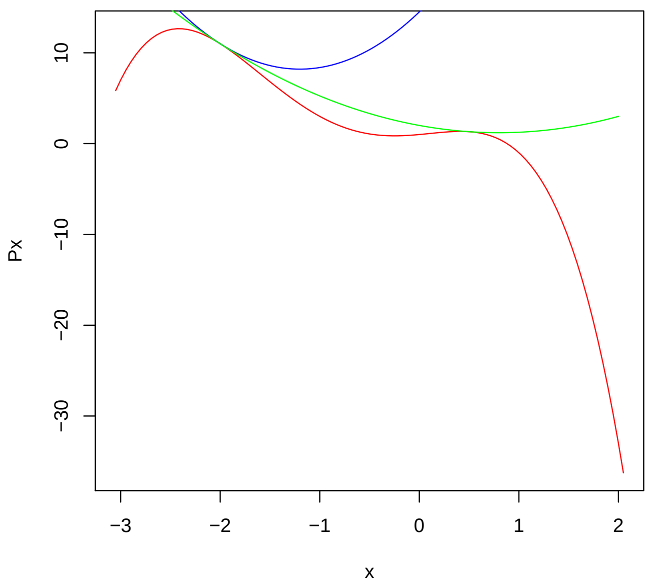

FIGURE 3. First Step from $y = -2$

Figures 13

List of Tables

1 Stationary Points 14

TABLE 1. Stationary Points

x	f	f''
$-1-\sqrt{2} \approx -2.4142136$	12.6568542	-24.485281
$-\frac{1}{4} \approx -0.2500000$	0.8554688	5.750000
$-1+\sqrt{2} \approx +0.4142136$	1.3431458	-7.514719