

DIRECT AND INDIRECT LEAST SQUARES APPROXIMATING POLYNOMIALS FOR THE FIRST DERIVATIVE FUNCTION

T. VAN HECKE,* *Ghent University*

Abstract

Finite difference methods are useful to give a discrete approximation of the derivative function f' based on a set of data points (x_i, f_i) ($i = 0, 1, 2, \dots, n$). If a continuous function is required to represent the derivative function and only a scatter of data points is available, finite difference formulas are insufficient. This paper describes two different approaches to derive an analytical description of the derivative function based on data points. Their performances are compared on several test functions where Monte Carlo simulations give statistics on the errors.

Keywords: derivative; polynomial approximation; finite difference; least squares difference

2010 Mathematics Subject Classification: Primary 65D25

Secondary 65D10

1. Introduction

Derivatives are used for multiple purposes to represent change or in algorithms such as the one of Newton-Raphson or Levenberg-Marquadt. In order to obtain an analytical description of the derivative function one can start by constructing the approximating polynomial obtained by a regression method (e.g. least squares regression), which is differentiated afterward. Polynomials are a preferred class of functions due to the ease of differentiation. But it is an interesting research question to investigate whether a method, where fitting is followed by differentiating, performs better than a method where these actions are performed in reversed order. This differentiation process will require a numerical method such as finite difference methods. In literature [5] one finds derivatives of interpolation functions as continuous approximations of the

* Postal address: Ghent University, Faculty of Engineering and Architecture, Voskenslaan 270, Ghent, Belgium, Tanja.VanHecke@ugent.be

first derivative function. However for a limited number of interpolation points this approach leads to the classical finite difference formulas, while for a larger number of interpolation points the strong oscillations of the approximation function are an undesired side effect. Another approach is based on nonparametric regression analysis [1]. Differentiating the nonparametric estimate with respect to the independent variable to obtain the first order derivative of the regression function only works well if the original regression function is extremely well estimated. When the data is noisy, it can lead to wrong derivative estimates. Two main approaches to overcome this problem are smoothing splines [4] and local polynomial regression [1]. They are based on limiting the approximation analysis to local regions for functions having varying estimation difficulty over the interval. In this paper we want to compare different global approaches when the measurements are contaminated by overall noise.

2. Mathematical background

2.1. Combining least squares and finite difference

Least Squares Difference (LSD) [2] is a well known method to fit data given by the points (x_i, y_i) ($i = 0, 1, \dots, n$), where a distance function

$$D = \sum_{i=0}^n (y_i - P(x_i))^2 \quad (1)$$

is minimized by optimized estimations of the model coefficients a_j , $j = 0, 1, \dots, k$ in case of a polynomial approximation $P(x) = \sum_{j=0}^k a_j x^j$ of degree k ($k < n$).

Finite differences [3] are widely used to approximate derivatives. Among the lowest order finite difference approximations for the first derivative $y'(x_i)$ are

- forward, first order: $y'(x_i) \approx \frac{y_{i+1} - y_i}{h}$
- backward, first order: $y'(x_i) \approx \frac{y_i - y_{i-1}}{h}$
- central, second order: $y'(x_i) \approx \frac{y_{i+1} - y_{i-1}}{2h}$
- central, fourth order: $y'(x_i) \approx \frac{-y_{i+2} + 8y_{i+1} - 8y_{i-1} + y_{i-2}}{12h}$
- central, sixth order: $y'(x_i) \approx \frac{y_{i+3} - 9y_{i+2} + 45y_{i+1} - 45y_{i-1} + 9y_{i-2} - y_{i-3}}{60h}$

with $h = x_{i+1} - x_i$. We refer to a method as a p^{th} -order method if the truncation error

is of the order of $O(h^p)$.

A direct method M_1 can be applied to generate a polynomial approximation on the points (x_i, y_i) ($i = 0, 1, \dots, n$). The differentiation of the fitted polynomial is the approximation result for the required derivative function y' . Indirect methods start by using finite difference methods to generate approximations for the derivative values \hat{y}'_i for each of the data points (x_i, y_i) . The least squares regression analysis is applied afterward on (x_i, \hat{y}'_i) to generate a continuous approximation of the derivative function y' . The indirect method M_2 is based on the first order forward finite difference formula, where the indirect methods M_3 , M_4 and M_5 are based on the second, fourth and sixth order central finite difference formulas respectively. To evaluate the methods M_j ($j = 1, 2, 3, 4, 5$) we consider an error function E_j with

$$E_j = \sum_{i=1}^{n-1} (\hat{y}'_i^{[j]} - y'(x_i))^2, \quad j = 1, 2, 3, 4, 5 \quad (2)$$

with estimation $\hat{y}'_i^{[j]}$, being an approximation generated by method M_j for the value of the derivative function in $x = x_i$.

2.2. Error analysis

We assume the output $y(x)$ to be troubled by zero mean Gaussian noise $\epsilon(x)$ such that

$$y(x) = T(x) + \epsilon(x), \quad (3)$$

with $T(x)$ the pure output. We call $\hat{a} = (a_0, a_1, a_2, \dots, a_k)^T$ the resulting vector of the least squares optimization with the direct method M_1 , which makes

$$\hat{T}'_1(x) = c(x)^T \hat{a} \quad \text{with } c(x)^T = (0, 1, 2x, 3x^2, \dots, kx^{k-1}). \quad (4)$$

This implies that the variance of the estimated $\hat{T}'_1(x)$ is related to the covariance matrix $\Sigma_{\hat{a}}$ of the least squares estimates of the polynomial coefficients as

$$\text{Var}(\hat{T}'_1(x)) = c(x)^T \Sigma_{\hat{a}} c(x). \quad (5)$$

The accuracy of the estimation of the derivative function obtained with M_1 will be directly influenced by the quality of the least square estimation of vector a as a result of the noise ϵ .

With $\hat{b} = (b_0, b_1, b_2, \dots, b_{k-1})^T$ the resulting vector of the least squares optimization with the indirect method M_j ($j = 2, 3, 4, 5$), the estimated derivative function can be written as

$$\hat{T}_j'(x) = d(x)^T \hat{b} \quad \text{with } d(x)^T = (1, x, x^2, \dots, x^{k-1}). \quad (6)$$

This implies that the variance of the estimated $\hat{T}_j'(x)$ is related to the covariance matrix $\Sigma_{\hat{b}}$ of the least squares estimates of the polynomial coefficients as

$$\text{Var}(\hat{T}_1'(x)) = d(x)^T \Sigma_{\hat{b}} d(x). \quad (7)$$

The accuracy of the estimation of the derivative function obtained with M_j ($j = 2, 3, 4, 5$) will be directly influenced by the quality of the least square estimation of vector b that reflects the noise as well as the order of the applied finite difference scheme.

3. Validation results

We compared the methods discussed in the previous paragraph on the testfunctions

$$\begin{aligned} T_1(x) &= e^x \\ T_2(x) &= \cos(3x - 2). \end{aligned} \quad (8)$$

Within the interval $[-1, 2]$ we generated data (x_i, y_i) with $x_i = -1 + ih$, $y_i = T(x_i) + \epsilon_i$ ($i = 0, 1, \dots, 30$) and step size $h = 0.1$. The added noise ϵ_i is Gaussian distributed with zero mean and variance σ . In Figure 1 a polynomial degree of $k = 7$ and $\sigma = 0.01$ is used for different test functions T_j ($j = 1, 2$). The residuals $(\hat{y}_i^{[j]} - T'(x_i))$ are plotted for $x \in [-1, 2]$ when the methods M_1 , M_2 and M_3 are applied on T_1 and T_2 . We see that the loss in accuracy due to lower order finite difference formulas is most prominent near the borders of the interval.

A Monte-Carlo simulation with 100 trials leads to box plots of the errors E_j ($j = 1, 2, 3, 4, 5$) for $\sigma = 0.2$ and $\sigma = 0.4$ ($h = 0.3$) as in Figure 2 with $k = 6$ and $k = 8$ when the test function $T_1(x) = e^x$ is used. The loss of accuracy due to highly oscillatory behavior when the error scale for $k = 8$ is compared with $k = 6$. When $k = 6$ the

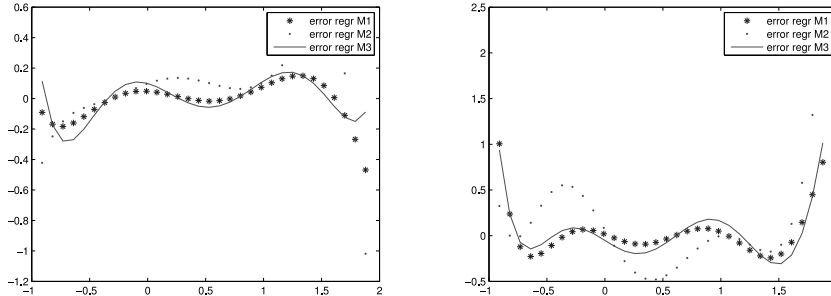


FIGURE 1: Residuals for the derivative function with method M_1 (*), method M_2 (.) and method M_3 (-) in case of $T_1(x) = e^x$ (left) and $T_2(x) = \cos(3x - 2)$ (right).

method M_2 performs worst as the finite difference order is too low and cannot be corrected by an accurate regression polynomial.

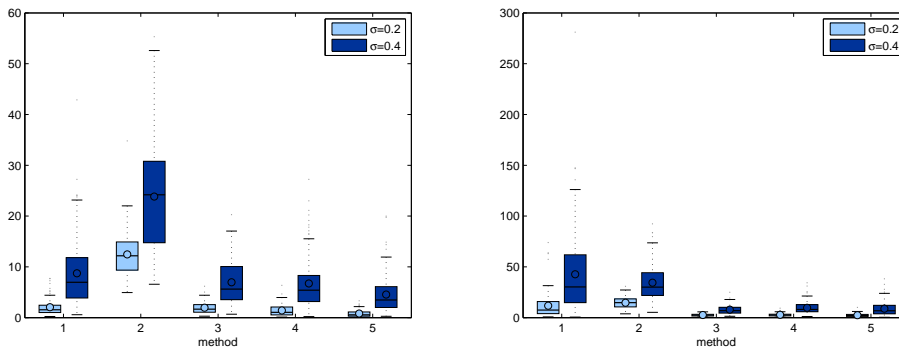


FIGURE 2: Boxplots of a 100-trials Monte-Carlo simulation of the residuals for the derivative function with methods M_i ($i = 1, 2, 3, 4, 5$) in case of $k = 6$ (left) and $k = 8$ (right) and $T(x) = e^x$.

4. Conclusion

Our comparative analysis revealed that when higher order central finite difference formulas are used, the indirect method where least squares approximation is preceded by finite difference formulas, performs better than the direct method M_1 , where the continuous least squares approximation function is derived to create a continuous approximation of the derivative function. Results from polynomials of lower degree

used for least squares regression with limited accuracy, can be improved when combined with higher order finite difference formulas.

References

- [1] FAN, J. AND GIJBELS, I. (1996). *Local Polynomial Modelling and its Applications*, Chapman and Hall.
- [2] FORNBERG, B. (1988). Generation of Finite Difference Formulas on Arbitrarily Spaced Grids. *Mathematics of Computation* **51 (184)**, 699-706.
- [3] HAZEWINKEL, M. (2001). *Finite difference calculus*, Encyclopedia of Mathematics, Springer.
- [4] STONE, C. (1985). Additive regression and other nonparametric models, *Ann. Statist.* **13(2)**, 689-705.
- [5] SULI, E. AND MAYERS, D. (2003). *An Introduction to Numerical Analysis*, Cambridge University Press.