

Estimation of Smooth Functions via Convex Programs

Eunji Lim¹ & Mina Attallah¹

¹ College of Business and Public Administration, Kean University, Union, NJ 07083, USA

Correspondence: Eunji Lim, College of Business and Public Administration, Kean University, Union, NJ 07083, USA.
 Tel: 1-908-737-4709. E-mail: elim@kean.edu

Received: May 17, 2016 Accepted: June 6, 2016 Online Published: June 28, 2016

doi:10.5539/ijsp.v5n4p150 URL: <http://dx.doi.org/10.5539/ijsp.v5n4p150>

Abstract

One of the numerically preferred methods for fitting a function to noisy data when the underlying function is known to be smooth is to minimize the roughness of the fit while placing a limit on the sum of squared errors. We show that the fit can be formulated as a solution to a convex program. Since convex programs can be solved by various methods with guaranteed convergence, our formulation enables one to use these methods to compute the fit numerically. Numerical results show that our formulation is successfully applied to the problem of sensitivity estimation of option prices as functions of the underlying stock price.

Keywords: nonparametric regression, smoothing spline, convex programming, penalized least squares

1. Introduction

We consider the problem of fitting a function $g : [a, b] \rightarrow \mathbb{R}$ to noisy data when the underlying function $f_* : [a, b] \rightarrow \mathbb{R}$ is assumed to satisfy a certain smoothness condition. We presume that we observe m noisy measurements Y_{i1}, \dots, Y_{im} of f_* at each $x_i \in [a, b]$ for $1 \leq i \leq n$, and that

$$Y_{ij} = f_*(x_i) + \epsilon_{ij},$$

for $1 \leq i \leq n$ and $1 \leq j \leq m$, where $\epsilon_{i1}, \dots, \epsilon_{im}$ are independent and identically distributed (iid) with a mean of 0 and a variance of $\sigma_i^2 < \infty$. We are particularly interested in the case where the underlying function f_* is known to be k times differentiable (for $k \geq 2$) and its k th derivative is square integrable. This situation naturally arises when the $(k-2)$ th derivative of f_* is observed to be smooth, and hence, one tries to find the fit g by minimizing the roughness of the $(k-2)$ th derivative of the fit g . Since the “roughness” of a function $f : [a, b] \rightarrow \mathbb{R}$ is measured by $\int_a^b \{f^{(2)}(x)\}^2 dx$, the roughness of the $(k-2)$ th derivative of g is measured by $\int_a^b \{g^{(k)}(x)\}^2 dx$. While minimizing the roughness $\int_a^b \{g^{(k)}(x)\}^2 dx$, one needs to ensure that the fitted function is close enough to the estimated values of f_* . This can be done by imposing a limit on the sum of squared distances between the fitted function and the estimates of f_* as follows:

$$\sum_{i=1}^n (\bar{Y}_i - g(x_i))^2 / n \leq S$$

for some positive constant $S > 0$, where $\bar{Y}_i = \sum_{j=1}^m Y_{ij} / m$ for $1 \leq i \leq n$. This leads to the following formulation for computing the fit:

$$\text{Minimize } \int_a^b \{g^{(k)}(x)\}^2 dx \quad \text{subject to } \sum_{i=1}^n (\bar{Y}_i - g(x_i))^2 / n \leq S \quad (1)$$

over $g \in \mathcal{D}^k$, where

$$\mathcal{D}^k = \left\{ f : [a, b] \rightarrow \mathbb{R} : f \text{ is } k \text{ times differentiable and } \int_a^b \{f^{(k)}(x)\}^2 dx < \infty \right\}.$$

Formulation (1) was introduced by Schoenberg (1964) and was studied extensively in the numerical analysis community by a number of authors including Reinsch (1967), Reinsch (1971), Wahba (1975), and Gander (1980). Formulation (1) is often contrasted, in the statistics literature, with the following formulation:

$$\text{Minimize } \sum_{i=1}^n (\bar{Y}_i - g(x_i))^2 / n + \lambda_n \int_a^b \{g^{(k)}(x)\}^2 dx \quad (2)$$

over $g \in \mathcal{D}^k$ for a sequence of nonnegative real numbers $(\lambda_n : n \geq 1)$. In Formulation (2), the term $\int_a^b \{g^{(k)}(x)\}^2 dx$ measures the roughness of the fit g and the term $\sum_{i=1}^n (\bar{Y}_i - g(x_i))^2 / n$ measures the sum of squared errors. The parameter λ_n controls the trade-off between the roughness and the goodness-of-fit.

One of the advantages of Formulation (1) over Formulation (2) is that a good estimate of S is easily provided in Formulation (1), while the performance of Formulation (2) is highly sensitive to the choice of $(\lambda_n : n \geq 1)$ and selecting $(\lambda_n : n \geq 1)$ is not straightforward. For this reason, Formulation (1) is preferred in the numerical analysis community. A good estimate of S in Formulation (1) can be obtained using the laws of large numbers as follows. By the weak law of large numbers, $\sum_{j=1}^m \epsilon_{ij} / \sqrt{m}$ converges in distribution to $N(0, \sigma_i^2)$ as m increases to infinity, where $N(\mu, \sigma^2)$ denotes a normal random variable with a mean of μ and a variance of σ^2 . Hence, $\bar{\epsilon}_i^2$, where $\bar{\epsilon}_i = (1/m) \sum_{j=1}^m \epsilon_{ij}$, can be approximated by $(N(0, \sigma_i^2))^2 / m$ for m sufficiently large. By applying the strong law of large numbers in n , $\sum_{i=1}^n (N(0, \sigma_i^2))^2 / n$ can be approximated as $\sum_{i=1}^n \sigma_i^2 / n$ for n sufficiently large. Therefore, the following approximation is possible

$$\frac{1}{n} \sum_{i=1}^n (\bar{Y}_i - f_s(x_i))^2 = \frac{1}{n} \sum_{i=1}^n \bar{\epsilon}_i^2 \approx \frac{1}{nm} \sum_{i=1}^n (N(0, \sigma_i^2))^2 \approx \frac{1}{nm} \sum_{i=1}^n \sigma_i^2$$

for n and m sufficiently large. The symbol \approx is used to informally express “approximate equality.” In practice, σ_i^2 is estimated by the sample variance S_i^2 of Y_{i1}, \dots, Y_{im} for $1 \leq i \leq n$; i.e., $S_i^2 = \sum_{j=1}^m (\epsilon_{ij} - \bar{\epsilon}_i)^2 / (m - 1)$. A good estimate of S is thus $\sum_{i=1}^n S_i^2 / (nm)$.

Despite its practical importance, there exist few numerical procedures that compute the solution of Formulation (1) with guaranteed convergence. Traditionally, the solution of (1) is computed as follows. For each $S > 0$, there is a unique $\lambda_n = \lambda_n(S)$ such that the solution of (2) for this λ_n is the solution of (1). Furthermore, the solution of (2) for this λ_n , denoted by g_{λ_n} , satisfies $\sum_{i=1}^n (\bar{Y}_i - g_{\lambda_n}(x_i))^2 / n = S$. Since (2) can be solved by solving a set of linear systems (pages 410–412 of Györfi et al., 2002), $\lambda_n(S)$ can be computed iteratively by means of the Newton procedure starting from an initial guess of $\lambda_n(S)$. This procedure does not guarantee global convergence to the solution of (1); see Reinsch (1971).

In this paper, we show that (1) can be reformulated as a convex program (Proposition 1). Convex programs can be solved using various methods that guarantee global convergence to the solution; see the Lagrangian method on page 217 of Zangwill (1969) for an example of methods that solve convex programs. Our formulation thus enables one to compute the solution of (1) with guaranteed convergence by using those methods and powerful software packages that are designed to solve convex programs.

This paper is organized as follows. Section 2 describes the proposed formulation in detail. Numerical results in Section 3 illustrate that our formulation is successfully applied to the problem of sensitivity estimation of option prices as functions of the underlying stock price. Concluding remarks are included in Section 4.

1. Proposed Formulation

In this section, we describe how Formulation (1) can be reformulated as a convex program. We first present some preliminary results.

A spline function with degree $r > 1$ with knots x_1, \dots, x_n , where $a < x_1 < \dots < x_n < b$, is a function $s : [a, b] \rightarrow \mathbb{R}$ having the following two properties: (a) In each of the intervals $[a, x_1], [x_1, x_2], \dots, [x_{n-1}, x_n], [x_n, b]$, $s(x)$ is given by some polynomial of degree r or less, and (b) $s(x)$ is $r - 1$ times continuously differentiable on $[a, b]$. We denote the set of spline functions with degree r by $\mathcal{S}_r([a, b])$. Even though $\mathcal{S}_r([a, b])$ seems to be infinite dimensional, it turns out to be finite dimensional with the dimension equal to $r + n + 1$. We describe one of the bases for $\mathcal{S}_r([a, b])$, which is the set of B-splines; the B-splines are preferred in numerical studies since they have bounded supports, and hence, produce well-conditioned numerical settings. We introduce additional knots $x_{-r}, \dots, x_0, x_{n+1}, \dots, x_{n+r+1}$ so that

$$x_{-r} < x_{-r+1} < \dots < x_0 < a < x_1 < \dots < x_n < b < x_{n+1} < \dots < x_{n+r+1}.$$

The B-spline $B_{i,r}$ of degree r is defined recursively by

$$B_{i,0}(x) = \begin{cases} 1, & \text{if } x_i \leq x < x_{i+1} \\ 0, & \text{otherwise} \end{cases} \tag{3}$$

for $i = -r, \dots, n + r$ and $x \in \mathbb{R}$ and

$$B_{i,l}(x) = \frac{x - x_i}{x_{i+l} - x_i} B_{i,l-1}(x) + \frac{x_{i+l+1} - x}{x_{i+l+1} - x_{i+1}} B_{i+1,l-1}(x) \tag{4}$$

for $i = -r, \dots, n+r-l, l = 1, \dots, r$, and $x \in \mathbb{R}$. By Theorem 14.1 on page 262 of Györfi et al. (2002), $\{B_{i,r} : i = -r, \dots, n\}$ restricted to $[a, b]$ is a basis of $S_r([a, b])$. We are ready to present the main result of this paper.

Proposition 1 Assume $2 \leq k \leq n$. Consider the minimization problem

$$\begin{aligned} & \text{minimize} && \int_a^b \{g^{(k)}(x)\}^2 dx && (5) \\ & \text{subject to} && \frac{1}{n} \sum_{i=1}^n (\bar{Y}_i - g(x_i))^2 \leq S \end{aligned}$$

over $g \in \mathcal{D}^k$ for some constant $S > 0$. Then, there exists a solution $\hat{g}_n \in \mathcal{D}^k$ to Problem (5). Furthermore, \hat{g}_n can be represented as

$$\hat{g}_n(x) = \sum_{i=-(2k-1)}^n \hat{c}_i B_{i,2k-1}(x)$$

for $x \in [a, b]$, where the \hat{c}_i s are the solution to the following convex program in the decision variables $c_{-(2k-1)}, \dots, c_n, y_1, \dots, y_n \in \mathbb{R}$.

$$\begin{aligned} & \text{minimize} && \int_a^b \left(\sum_{i=-(2k-1)}^n c_i B_{i,2k-1}(x) \right)^2 dx = \sum_{i=-(2k-1)}^n \sum_{j=-(2k-1)}^n c_i c_j \int_a^b B_{i,2k-1}^{(k)}(x) B_{j,2k-1}^{(k)}(x) dx \\ & \text{subject to} && \sum_{i=1}^n (\bar{Y}_i - y_i)^2 / n \leq S, && (6) \\ & && \sum_{i=-(2k-1)}^n c_i B_{i,2k-1}(x_j) = y_j, j = 1, \dots, n. \end{aligned}$$

Proof. We let $C = \{(y_1, \dots, y_n) \in \mathbb{R}^n : \sum_{i=1}^n (\bar{Y}_i - y_i)^2 / n \leq S\}$. It should be noted that C is a nonempty, closed and bounded subset of \mathbb{R}^n .

By Lemmas 20.2 and 20.3 on pages 415 and 416 of Györfi et al. (2002), for any $y = (y_1, \dots, y_n)$ in C , there exists a unique spline function h of degree $2k - 1$ satisfying $h(x_i) = y_i$ for $i = 1, \dots, n$ and $h^{(l)}(a) = h^{(l)}(b) = 0$ for $l = k, \dots, 2k - 1$. Furthermore, $\int_a^b \{h^{(k)}(x)\}^2 dx \leq \int_a^b \{g^{(k)}(x)\}^2 dx$ for any $g \in \mathcal{D}^k$ satisfying $g(x_i) = y_i$ for $1 \leq i \leq n$. Since $\{B_{i,2k-1} : -(2k - 1) \leq i \leq n\}$ is a basis of $S_{2k-1}([a, b])$ (Theorem 14.1 on page 262 of Györfi et al., 2002), there exists a unique $c_{-(2k-1)}, \dots, c_n$ such that $h(x) = \sum_{i=-(2k-1)}^n c_i B_{i,2k-1}(x)$ for $x \in [a, b]$. The c_i s are determined by the following linear system with $n + 2k$ variables and $n + 2k$ linear equations:

$$\begin{aligned} \sum_{i=-(2k-1)}^n c_i B_{i,2k-1}(x_j) &= y_j \\ \sum_{i=-(2k-1)}^n c_i B_{i,2k-1}^{(l)}(a) &= 0 \\ \sum_{i=-(2k-1)}^n c_i B_{i,2k-1}^{(l)}(b) &= 0 \end{aligned} \tag{7}$$

for $j = 1, \dots, n$ and $l = k, \dots, 2k - 1$. By the uniqueness of $c_{-(2k-1)}, \dots, c_n$, the linear system (7) is nonsingular, and hence, the map from y in \mathcal{D}^k to $\int_a^b \{h^{(k)}(x)\}^2 dx$ is continuous. Hence, there exists $\hat{y} = (\hat{y}_1, \dots, \hat{y}_n) \in C$ that minimizes $\int_a^b \{g^{(k)}(x)\}^2 dx$ over $g \in \mathcal{D}^k$. Let \hat{h} be the unique spline function satisfying $\hat{h}(x_i) = \hat{y}_i$ for $i = 1, \dots, n$ and $\hat{h}^{(l)}(a) = \hat{h}^{(l)}(b) = 0$ for $l = k, \dots, 2k - 1$. Let $\hat{c}_{-(2k-1)}, \dots, \hat{c}_n$ be the solution to (7) when $(y_1, \dots, y_n) = (\hat{y}_1, \dots, \hat{y}_n)$. Obviously, \hat{h} is a feasible solution to (6). For any feasible solution $c_{-(2k-1)}, \dots, c_n, y_1, \dots, y_n$ of (6), $\sum_{i=-(2k-1)}^n c_i B_{i,2k-1}(x)$ belongs to \mathcal{D}^k , and hence, $\int_a^b (\sum_{i=-(2k-1)}^n c_i B_{i,2k-1}^{(k)}(x))^2 dx$ is less than or equal to $\int_a^b \{\hat{h}^{(k)}(x)\}^2 dx$. Thus, the \hat{c}_i s and the \hat{y}_i s are the solution to (6). \square

We close this section by describing how to evaluate the B-splines and the integration of the product of their k th derivatives that appear in (6). The B-splines can be evaluated recursively through Equations (3) and (4). The k th derivative of the B-spline can be evaluated recursively through the following relation: for a B-spline of degree r ,

$$dB_{i,r}(x)/dx = (r/(x_{i+r} - x_i))B_{i,r-1}(x) - (r/(x_{i+r+1} - x_{i+1}))B_{i+1,r-1}(x) \tag{8}$$

for $i = -r, \dots, n$ and $x \in [a, b]$; see Lemma 14.6 on page 265 of Györfi et al. (2002).

There are a couple of ways to compute the integration in (6). First, $\int_a^b B_{i,2k-1}^{(k)}(x)B_{j,2k-1}^{(k)}(x)dx$ can be computed by evaluating $B_{i,2k-1}^{(k)}$ using the recursion in (8) and by numerically evaluating the integration. Second, one can use the closed form formula for the integration of the product of the k th derivatives of the B-splines given by Equation (7) on page 1026 of Vermeulen et al. (1992) and the closed form formulas for the B-splines (Equation (1.20) on page 8 of Dierckx, 1993) to directly compute the k th derivatives and the integration of their products.

3. Numerical Results

3.1 A Motivation from Finance

This paper is motivated by the need to estimate the price of a stock option and its derivatives as functions of the underlying stock price. The first and second derivatives of an option price play an important role when financial institutions manage a portfolio of stocks and stock options in an attempt to hedge the risks associated with the portfolio. For example, consider a call option that gives the holder of the option the right to buy the underlying stock by a certain date for a certain price. Since the price of such a call option depends on the underlying stock price, the option price can be denoted by $f_*(x)$, where x is the underlying stock price per share. The delta (Δ) of the option is defined by the first derivative df_*/dx of the option price with respect to the underlying stock price. It is well known that a portfolio consisting of a short position of the call option and a long position of Δ shares of the underlying stock is expected to grow at a risk-free interest rate. Since the value of delta changes as the underlying stock price changes over time, the number of shares of the underlying stock in the portfolio must be changed periodically to stay in the risk-free position. This step is called rebalancing. When rebalancing a portfolio, the gamma (Γ) of the option, which is the second derivative d^2f_*/dx^2 of the option price with respect to the underlying stock price, is used since the value of gamma tells us how much delta changes, and hence, how many shares of the underlying stock should be sold or bought in order to stay in a delta neutral position.

Recently, financial institutions have been issuing stock options with much more complex payoff structures than that of a call option. For such options, the option price cannot be expressed in a closed-form formula, and hence, one needs to use simulation to estimate the option price. Simulation of the option price consumes a significant amount of time. In order to facilitate quick decisions, traders in financial institutions conduct simulations before they actually need to rebalance a portfolio. Since the underlying stock price in the future cannot be predicted accurately, the traders conduct simulations for all possible underlying stock prices on the day when rebalancing takes place. The question thus boils down to how to estimate the option price f_* and its first and second derivatives, df_*/dx and d^2f_*/dx^2 , over a range $[a, b]$ of possible underlying stock prices using simulation.

One simple strategy for estimating f_* , df_*/dx and d^2f_*/dx^2 for $x \in [a, b]$ is to choose various possible values for the stock price, say x_1, \dots, x_n , from $[a, b]$, estimate the option price at each x_i for $1 \leq i \leq n$ using simulation, use finite differences of the estimated option prices to estimate delta, and use finite differences of the estimated delta values to estimate gamma. A serious drawback of this approach is that the estimated delta and gamma values often lead to noisy curves as functions of the underlying stock price. It is especially frustrating for traders to see gamma values fluctuating around zero because positive gamma values suggest purchasing additional shares of the underlying stock, while negative gamma values suggest selling some of the shares. Figure 1 shows an example of the estimated gamma values plotted against the underlying stock price. The stock prices S_1, S_2 , and S_3 are close to one another, but the graph suggests different strategies because the gamma values are positive, negative, and positive at S_1, S_2 , and S_3 , respectively. This degree of randomness in the gamma curve is not acceptable in practice.

To overcome this drawback, we propose fitting a curve $g : [a, b] \rightarrow \mathbb{R}$ to the estimated values of f_* so that the fitted curve has a smooth second derivative. Since the “roughness” of a function $f : [a, b] \rightarrow \mathbb{R}$ is measured by $\int_a^b \{f^{(2)}(x)\}^2 dx$, the roughness of the second derivative of g is measured by $\int_a^b \{g^{(4)}(x)\}^2 dx$. While minimizing $\int_a^b \{g^{(4)}(x)\}^2 dx$, we want to make sure that the fitted function is close to the estimated values of f_* by placing a limit on the sum of squared distance between the fitted values and the estimated values. This leads to the following optimization problem:

$$\begin{aligned} & \text{minimize} && \int_a^b \{g^{(4)}(x)\}^2 dx \\ & \text{subject to} && \frac{1}{n} \sum_{i=1}^n (\bar{Y}_i - g(x_i))^2 \leq S \end{aligned}$$

over $g \in \mathcal{D}^4$, where Y_{ij} is the j th replication of the estimated value of the option price at x_i for $1 \leq i \leq n$ and $1 \leq j \leq m$. The above formulation is a special case of Formulation (1) with $k = 4$.

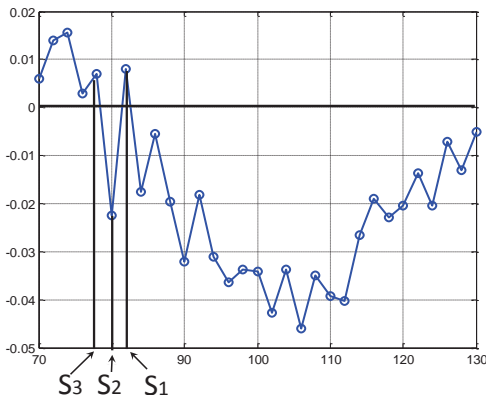


Figure 1. The horizontal axis is the underlying stock price, and the vertical axis is gamma.

3.2 Applying Our Formulation to Sensitivity Estimation of Option Prices

We consider the case where $f_*(x)$ is the expected payoff of a certain equity-linked security (ELS) when the underlying stock price is denoted by x . The payoff function of the ELS has the following structure. Suppose that the ELS is issued at time 0 and matures at time T . We denote the underlying stock price at time $t \in [0, T]$ by S_t . There are q days when early redemption is possible. On each of those days d_i for $1 \leq i \leq q$, the ELS expires with a payoff of $\$r_i$ if S_{d_i}/S_0 exceeds some threshold b_i . Otherwise, the ELS does not expire until maturity. If there is no early redemption and S_T/S_0 does not drop below a limit b until maturity, then the ELS expires with a payoff of $\$1$ at maturity. Otherwise, the ELS expires with a payoff of $\$S_T/S_0$ at maturity.

We let $a = 90, b = 110$, and $x_i = 90 + (20)(i/n) - (10/n)$ for $1 \leq i \leq n$. For each x_i , a sample path of a geometric Brownian motion is generated as a trajectory of the stock price between now and maturity, and the corresponding payoff of the ELS is computed. Y_{ij} is the payoff computed this way in the j th replication of the geometric Brownian motion at x_i . The parameters used for the experiment are $T = 365$ days, $q = 6, d_1 = 61, d_2 = 122, d_3 = 182, d_4 = 243, d_5 = 304, d_6 = 365, b_1 = 0.9, b_2 = 0.9, b_3 = 0.85, b_4 = 0.85, b_5 = 0.8, b_6 = 0.8, r_1 = 1.05, r_2 = 1.10, r_3 = 1.15, r_4 = 1.20, r_5 = 1.25, r_6 = 1.30$, and $b = 0.7$. The remaining time until maturity is 60 days, the annual volatility is 30%, the annual risk-free interest rate is 5%, and the initial stock price at time 0 is $\$125$.

We set $m = 50$, so 50 sample paths for the geometric Brownian motion are generated at each x_i to compute Y_{i1}, \dots, Y_{i50} for $1 \leq i \leq n$. We compute $\bar{Y}_i = \sum_{j=1}^{50} Y_{ij}/50$ for $1 \leq i \leq n$ and use $(x_1, \bar{Y}_1), \dots, (x_n, \bar{Y}_n)$ to compute the proposed estimator \hat{g}_n by solving (6) with CVX, a package for specifying and solving convex programs (Grant & Boyd, 2014). The constant S in Formulation (6) is replaced with $\sum_{i=1}^n S_i^2$, where S_i^2 is the sample variance of Y_{i1}, \dots, Y_{im} for $1 \leq i \leq n$.

To measure the accuracy of the proposed estimator, we compute the following integrated mean square error (IMSE) between the underlying function f_* and \hat{g}_n : $\sum_{i=1}^n (\hat{g}_n(x_i) - f_*(x_i))^2 / n$, where $f_*(x_i)$ is estimated from the average of 400,000 iid replications of Y_{ij} at each x_i . Table 1 reports the averages and the standard deviation of the IMSE, computed based on 200 iid replications of \hat{g}_n , for a variety of n values. The IMSE decreases as n increases, which displays the convergence of \hat{g}_n to f_* as $n \rightarrow \infty$.

Table 1. The averages and the standard deviation of the IMSE when $f_*(x)$ is the price of the ELS.

n		5	10	20	40	80	160
IMSE	Average (10^{-4})	8.6	4.4	2.1	1.1	0.7	0.6
	Standard deviation (10^{-4})	6.1	2.9	1.5	0.8	0.5	0.3

4. Concluding Remarks

In this paper, we study a numerically preferred formulation for fitting a smooth function to noisy data. Numerical results illustrate that our formulation successfully computes the fit. They also suggest that the fit converges to the true function as the number of observations in the data set increases to infinity. Future research topics include studies on asymptotic

properties of the fit when the number of observations in the data set increases to infinity.

References

- Dierckx, P. (1993). *Curve and surface fitting with splines*. New York: Oxford University Press Inc.
<http://dx.doi.org/10.5860/choice.31-2162>
- Gander, W. (1980). Least squares with a quadratic constraint. *Numer. Math.*, *36*, 291-307.
<http://dx.doi.org/10.1007/bf01396656>
- Grant, M., & Boyd, S. (2014). CVX: Matlab software for disciplined convex programming, version 2.1. <http://cvxr.com/cvx>, March 2014.
- Györfi, L., Kohler, M., Krzyżak, A., & Walk, H. (2002). *A distribution-free theory of nonparametric regression*. New York: Springer. <http://dx.doi.org/10.1007/b97848>
- Reinsch, C. H. (1967). Smoothing by spline functions. *Numer. Math.*, *10*, 177-183. <http://dx.doi.org/10.1007/bf02162161>
- Reinsch, C. H. (1971). Smoothing by spline functions II. *Numer. Math.*, *16*, 451-454.
<http://dx.doi.org/10.1007/bf02169154>
- Schoenberg, I. J. (1964). Spline functions and the problem of graduation. *Proc. Nat. Acad. Sci. (USA)*, *52*, 947-950.
<http://dx.doi.org/10.1073/pnas.52.4.947>
- Vermeulen, I. J., Bartels, R. H., & Heppler, G. R. (1992). Integrating products of B-splines. *Siam J. Sci. Stat. Comput.*, *13*, 1025-1038. <http://dx.doi.org/10.1137/0913060>
- Wahba, G. (1975). Smoothing noisy data with spline functions. *Numer. Math.*, *24*, 383-393.
<http://dx.doi.org/10.1007/bf01437407>
- Zangwill, W. I. (1969). *Nonlinear programming: a unified approach*. New Jersey: Prentice-Hall Inc.

Copyrights

Copyright for this article is retained by the author(s), with first publication rights granted to the journal.

This is an open-access article distributed under the terms and conditions of the Creative Commons Attribution license (<http://creativecommons.org/licenses/by/3.0/>).