Polynomial regression under shape constraints

Francois Wahl, T Espinasse

▶ **To cite this version:**

Francois Wahl, T Espinasse. Polynomial regression under shape constraints. 2014. hal-01073514v1

HAL Id: hal-01073514

https://inria.hal.science/hal-01073514v1

Preprint submitted on 3 Nov 2014 (v1), last revised 4 Mar 2015 (v2)

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

To appear in the *Journal of Statistical Computation and Simulation*
Vol. 00, No. 00, Month 20XX, 1–14

Polynomial regression under shape constraints

F. Wahl[a*b] and T. Espinasse[a]

[a] *Université Lyon 1, Institut Camille Jordan, 43 blvd du 11 novembre 1918, 69622 Villeurbanne - France*

[b] *IFP Energies nouvelles, Etablissement de Lyon, BP3, 69360 Solaize - France*

(*Received 00 Month 20XX; final version received 00 Month 20XX*)

Calculating regression under shape constraints is a problem addressed by statisticians since long. This paper shows how to calculate a polynomial regression of any degree and of any number of variables under shape constraints, which include bounds, monotony, concavity constraints. Theoretical explanations are first introduced for monotony constraints and then applied to ad hoc examples to show the behavior of the proposed algorithm. Two real industrial cases are then detailed and worked out.

Keywords: multivariate polynomial regression; monotony requirements; constrained regression; numerical algorithm; quadratic programming

AMS Subject Classification: 62H12, 62J05

1. Motivation

Fitting a multivariate regression function to a set of n given observed points is a common industrial problem. Additionaly, very often experts seek to impose some shape constraints on the resulting function, like monotony constraints or concavity.

Industrial problems are very commonly ill posed, and do not follow the theoritical standards of ideal situations for a lot of reasons. First of all, in an industrial context, obtaining experimental values can be difficult: experiences are not as perfectly controlled as wished even in a laboratory environment since they depend on a wide range of variables which can be difficult to master individually. Secondly, measurements are difficult to acquire, depending on the examined quantities and rely on some devices or captors which have their own defaults Thirdly, experiences are subjects to constraints like time and money and it can be very expensive to acquire the new values sought in the experiment.

For all these reasons, in this industrial context, observed values may share one of the following features: very few experimental points are available; they belong to multivariate settings, e.g. five or more dimensions being a very common situation; some of the points are suspicious but not really detectable (specially in more than two or three dimension).

As a result, the obtained regression can sometimes exhibit strange behaviors: oscillation in the responses functions, not desired minima or maxima, false tendencies if the calculated values increase instead of decreasing (or the other way round). To compensate all these flaws and obtain an acceptable result, experts try to use a posteriori knowledge on the regression behavior. The resulting function will be accepted only if for example monotony behaviors are observed on the whole domain of interest (even if it has been

*Corresponding author. Email: Francois.Wahl@univ-lyon1.fr

established on a sub-domain only), and this can only be obtained by chance without a proper methodology.

Explaining how to incorporate these constraints in an a priori manner is the purpose of this article. Moreover, resulting functions should be easy to calculate, avoiding tedious procedure for fitting extra hyper-parameters or heavy computer resources for predicting a new point if possible.

The example which first motivated this work is a case in process engineering detailed in the section entitled 'hydrotreatment of naphta'. One of the goal of hydrotreatment processes is to remove sulfur in petroleum feedstock, in order to fulfil environmental requirements. Indeed, in the underground, petroleum always contains some percentage of sulfur, and this very nocive compound must be eliminated. To simply describe a very complicated chemical transformation, the feed is heated to a high temperature (between 200^oC and 400^oC), and put under heavy pressure of hydrogen (from 10b to 140b). Under these severe conditions, when contacting a specific catalyst, chemical bounds linking sulfur to carbon chemical compounds are broken, and in this way, sulfur can be extracted from the original feed.

This transformation can be quite cumbersome to modelise : the feed contains a huge number of different types of molecules, and the reactions involved in the process, in presence of the catalyst, are not fully understood, ... One of the simplest possibility is to adjust a degree 2 polynomial in order to obtain an approximate model of the response.

However, this very easily constructed model should exhibit some expected behaviors (see the detailed description in the corresponding section). For example, when the temperature increases, the sulfur content at the outlet should decrease, in accordance to the Arrhenius law governing the chemical reactions.

But the polynomial expression of the answer, obtained through classical least square regression modeling does not guarantee satisfying all these expectations. The objective of this work is to develop a regression model that allows us to incorporate monotony constraints into the estimation of the response. Note that in this example, few experimental points are available, and the expected function is multivariate: the dimension of the input space is 4.

As we shall see, polynomial regression functions can be constrained to fulfill all the needed requirements. They stay very simple to calculate, and very smooth. Moreover, by construction, they are guaranteed to respect all the constraints in the entire domain of interest, avoiding unexpected changes of behavior like (even slight) changes of concavity, which may occur with the use of exponential or gaussian kernel function for example. This last point is crucial as the regression function will be extensively used. Slightest defaults may become apparent.

Our methodology can be applied to any type of constraints as long as they are linear with respect to the coefficients of the polynomial. We show in this paper how to express monotony constraints into linear form. These transformations can as well be applied to bound constraints or concavity (convexity) constraints.

This paper is organized as follows : after a short bibliography, the theory is exposed for monotony constraints, first in dimension 1, before extending the idea to more dimensions. Simulations studies are then demonstrated with ad hoc examples, and two industrial cases: hydrotreatment of naphta is finally detailed, and a case in laser-plasma experiments is presented.

2. Selected bibliography

Imposing shape restrictions is a very usual demand in regression analysis, and is still a very active domain of research. Shape restrictions include equality constraints and prior knowledge on particular points, for which values are certain, like intercept, maximum or minimum values or inequality constraints like monotony requirements or positivity constraints on the function and its derivative , concavity or convexity (see Lauer [1]).

In univariate settings, one can say that each regression method has its extension with shape restrictions. Among others, we can refer to Barlow et al. [2] with the Pool Adjacent Violators Algorithm (PAVA) for solving monotonic regression problems. Starting in dimension one, Burdakov et al[3] propose to pool every point violating a constraint with the next adjacent value.Ramsay [4] introduced the use of regression splines for monotone regression functions. Another type of method for regression subject to monotonicity constraints is kernel-type estimators (Hall and Huang 2001 [5]; Dette et al. 2006 [6]). Local polynomial is the base of the work of Marron et al. [7].

Extensive bibliography can be found in Mammen [8] or Scheder [9].

Until recently, relatively to the univariate case, few works exist in multivariate settings. We distinguish two types of approaches, the first based on a 'fit then monotonise' strategy (see [8]), and the laters on smoothing non parametric regression like kernel or 'SVR' regression or kriging.

In dimension 2 or greater, the authors in [10] extend the PAVA procedure via graph theory. The numerical experiments show that GPAV algorithms enjoy both low computational burden and high accuracy. It can be run with a lot of data and several variables. But the solution is not guaranteed to be C^2, and may exhibit a staircase behavior, with large regions of constant behavior followed by an abrupt step to the next level.

In kernel or non parametric regression, all the proposed smoothing methods suffer from the same drawback, the curse of dimensionality: for example with monotony requirements, to be sure that the constraints apply on the whole domain, a very usual way is to define a grid of points and apply the needed constraints on every node. Obviously, the number of conditions grows exponentially with the dimension of the input space and this way of proceeding is only possible in low dimension problems. Besides, there is no guarantee that between the nodes of the grid, constraints are still valid. Finally, each prediction on a new point requests to solve a new complex problem if one does not interpolate between the points of the grid.

Dette et al [6] postulate that the experimental points available are sampled from a cumulative distribution function (cdf) to be estimated. This cdf is monotonically increasing by construction. Starting with one dimensional increasing curve, the algorithm is extended to more than one dimension.

Racine and Parmeter [11] propose a generalization of the classical kernel regression where the estimated response is given by $\hat{y}(x) = \sum_{i=1}^{n} p_i A(x, x_i) y_i$ and A is a kernel estimator (for example, Nadaraya Watson), $(x_i, y_i)_{i=1,n}$ are known observed points and x a new point where the response has to be predicted. The weights p_i have to be adjusted to satisfy the monotonicity constraints.

The equation calculated by SVR algorithm is given by $y(x) = \sum_{i=1,n} \alpha_i H(x, x_i)$ when the kernel H contains a bias term, where α_i are suitable parameters. In SVR, the coefficients are found by solving a QP optimization problem (see Lauer and Bloch [1]). In case of additional linear constraints (with respect to the α_i), the number of conditions is only augmented, the solving mechanism remains the same.

In kriging, one can refer to the work of Da Veiga and Marrel ([12]), which relies on conditional expectations of the truncated multinormal distribution. Antoniadis and coauthors [13] propose a constrained regression function using penalized wavelet regression

techniques.

A few words are needed on polynomial regression under shape constraints. This has been studied in dimension 1, with Turlach "On Monotone Regression" [14] or in a non parametric settings with the use of Bernstein polynomials in [15] or [9].

3. Theory

To overcome the limitations of non parametric regressions and be formally sure that shape constraints are verified everywhere, whatever x considered, we restrict ourselves to polynomial regression, and we make the assumption that the observed points $x_{(i)}$ take their value in some hypercube, meaning that each independent coordinate is bounded between a minimum and a maximum value. For convenience and without any loss of generality this minimum is taken to be 0 and the maximum +1.

3.1. *Notations*

Let us consider an input space of dimension v. $x = (x_1, x_2, ..., x_v)$ is a point in this space, $x_{(i)}$ is a point in a set of data indexed by i. We denote P a multivariate polynomial of degree d, of the variables $x_1, x_2, ..., x_v$. $P_{(10\cdots0)}$ refers to the derivative of P with respect to x_1.

3.2. *In dimension 1*

Let us examine a very simple example, in dimension 1 ($v = 1$) where we try to fit a degree 3 polynomial ($d = 3$) expressed as $P(x) = \beta_0 + \beta_1 x + \beta_2 x^2 + \beta_3 x^3$ on a set of n given points $(x_{(i)}, y_{(i)})_{i=1,n}$, with the constraint that the resulting solution should be monotically increasing on the domain of definition of x, the interval $[0, 1]$.

The derivative $P_{(1)}(x) = \beta_1 + 2\beta_2 x + 3\beta_3 x^2$ is linear with respect to the coefficients β_1, β_2 and β_3. To empathize this, we rewrite $P_{(1)}(x)$ as $P_{(1)}(x) = z(t_1, t_2) = \beta_1 + 2\beta_2 t_1 + 3\beta_3 t_2$ taking $t_1 = x$ and $t_2 = x^2$. Now, if $z(t_1, t_2)$ is positive in every four corner of the square $[0, 1]^2$, then by convexity, $z(t_1, t_2)$ will be positive everywhere in $[0, 1]^2$, and so will $P_{(1)}(x)$ $\forall x \in [0, 1]$.

In fact, all the possible values for $[t_1, t_2]$ are included in the triangle defined by the vertices [0,0], [1,0],[1,1], by convexity of the function $t \to t^2$ for $t \in [0, 1]$. Consequently, to be sure of the sign of the derivative, it is only necessary to check the three linear following inequalities:

$$\beta_1 \geq 0,\ \beta_1 + 2\beta_2 \geq 0,\ \beta_1 + 2\beta_2 + 3\beta_3 \geq 0 \tag{1}$$

corresponding to the equation of $z(t_1, t_2)$ in the three corners $[0, 0]$, $[1, 0]$ and $[1, 1]$.

Mathematically, the least square problem to be solved can be expressed as $\underset{\beta}{argmin} \sum_{i=1,n} (y_{(i)} - P(x_{(i)}))^2$, s.t. constraints (1), which is a classical convex quadratic programming problem (see [16]).

This example is illustrated on the following figure, with the function

$$y = 1.5x + \frac{3}{4\pi} sin(4\pi x),$$

which is approached by a polynomial regression of degree 3. Ten values for x are randomly

taken in the interval [0,1], and the corresponding y are calculated. A random normal noise of standard deviation $\sigma = 0.1$ is added to each y. The green squares indicate the chosen points.

Three curves are drawn on the figure: in plain red, the calculated constrained regression, in dotted blue, the non constrained standard multivariate polynomial regression, in plain black, the true function. As can be seen on the graphic, the regression without any shape constraints is not monotone.

With only ten points, we use the root mean square error defined as $RMSE = \sum_{i=1}^{n}(\hat{y_{(i)}} - y_{(i)})^2$ as an indicator of the quality of the regression, where n is the number of points, and $\hat{y}_{(i)}$ the calculated i-th value. Without constraints, the regression gives $RMSE = 0.1060$, and with constraints, the same indicator is only slightly worse in this case, $RMSE = 0.1081$.

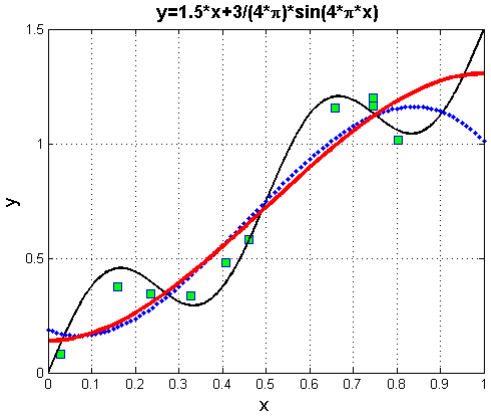


Figure 1. an example with a degree 3 polynomial

To guarantee the shape requirement is satisfied, only 3 linear conditions are added to the initial optimization problem.

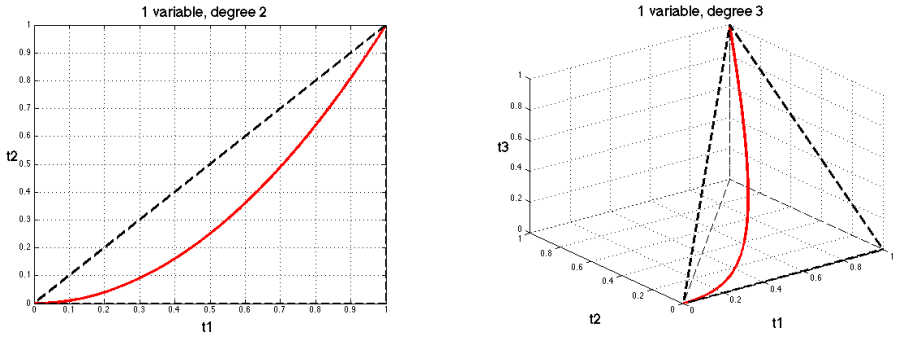
In a more general setting, still in dimension $v = 1$, if the polynomial to fit is of degree d, the number of constraints will be also d: the constraints will be applied to the derivative $P_{(1)}(x)$ which is a polynomial of degree $d-1$ corresponding to some linear function $z(t) = z(t_1, t_2, ..., t_{d-1})$ with $t_1 = x$, ..., $t_{d-1} = x^{d-1}$. When $x \in [0, 1]$, a point of coordinate $t = (x^1, x^2, ..., x^{d-1})$ is always inscribed in the convex polytope with d vertices $(0, 0, ..., 0)$, $(1, 0, ..., 0)$, $(1, 1, ..., 0)$, ..., $(1, 1, ..., 1)$, and this leads effectively to write d constraints, corresponding to the d vertices.

The figure 2 illustrates this statement.

3.3. *In dimension* > 1

Now, we switch to a more general situation, where x is v-dimensional, with a monotony constraint required for the first coordinate x_1.

To check the condition in every point of the hypercube covered by $x_1, x_2, ..., x_v$, we examine the derivative of P with respect to x_1, $P_{(10...)}(x)$, and we have to verify that $P_{(10...)}(x) \geq 0$ or (≤ 0) in the entire domain. As usual, we rewrite $P_{(10...)}(x)$ as $P_{(10...)}(x) = z(t_1, t_2, ..., t_m)$ where each t_k for $k = 1, m$ corresponds to one of the m monomial in the expression of $P_{(10...)}$. Indeed, as in dimension 1, one way to be sure P is monotone with respect to x_1 is to impose the conditions that z should be positive (or negative) in every corner C_i of the corresponding region for t. When written in this way, the problem to



(a) in dimension 1. (b) in dimension 2.

Figure 2. Two parametric curves in dimension 1 defined by a single variable polynomial (left, equation $t_1 = x$, $t_2 = x^2$) and in dimension 2 (right, Equation $t_1 = x$, $t_2 = x^2$, $t_3 = x^3$), showing that they are included in a triangle and a tetrahedra

solve in dimension v can be rephrased :

$$\underset{\beta}{argmin} \sum_{i=1,n} (y_{(i)} - P(x_{(i)}))^2, \text{s.t. constraints } z(C_i) > 0, \forall C_i \tag{2}$$

This a classical quadratic optimization optimization problem with linear inequality constraints, nowadays easily solvable by usual available mathematical software, save for the number of constraints: if the principle is simple, the realization is much more tedious since the number m of necessary monomials to express $P_{(10...)}$ will increase exponentially with the dimension d and the number of variables v, and so will the number of constraints (2^m).

We first show on a simple example how to extend the previous property explained in dimension 1, in order to reduce drastically the number of constraints. Then we introduce a general proposition which gives a means to automatically generate the constraints needed.

We take an arbitrary example with 2 variables, and a degree 3 polynomial: $P(x) = \beta_0 + \beta_{10}x_1 + \beta_{20}x_1^2 + \beta_{11}x_1x_2 + \beta_{21}x_1^2x_2$

After derivating $P(x)$ with respect to x_1, we obtain: $P_{(10)}(x) = \beta_{10} + 2\beta_{20}x_1 + \beta_{11}x_2 + 2\beta_{21}x_1x_2$. We rewrite $P_{(10)}(x) = \alpha_{00} + \alpha_{10}x_1 + \alpha_{01}x_2 + \alpha_{11}x_1x_2$ to simplify the notation and we see that:

(1) if $\alpha_{00} \geq 0$ and $\alpha_{00} + \alpha_{10} \geq 0$, then $\alpha_{00} + \alpha_{10}x_1 \geq 0, \forall x_1 \in [0,1]$
(2) if $\alpha_{00} \geq 0$ and $\alpha_{00} + \alpha_{01} \geq 0$, then $\alpha_{00} + \alpha_{01}x_2 \geq 0, \forall x_2 \in [0,1]$
(3) if $\alpha_{00} + \alpha_{01}x_2 \geq 0$ and $\alpha_{00} + \alpha_{10} + \alpha_{01}x_2 + \alpha_{11}x_2 \geq 0$, then $\alpha_{00} + \alpha_{10}x_1 + \alpha_{01}x_2 + \alpha_{11}x_1x_2 \geq 0, \forall x_1, x_2 \in [0,1]^2$.
(4) $\alpha_{00} + \alpha_{10} + \alpha_{01}x_2 + \alpha_{11}x_2 \geq 0$ is in turn implied by $\alpha_{00} + \alpha_{10} \geq 0$ and $\alpha_{00} + \alpha_{10} + \alpha_{01} + \alpha_{11} \geq 0$.

Gathering everything, we obtain 4 conditions, expressed in this case with the α on the left and equivalently with the β on the right as:

$$\begin{array}{cccc} \alpha_{0,0} & \geq 0 & \beta_{1,0} & \geq 0 \\ \alpha_{0,0} + \alpha_{1,0} & \geq 0 & \beta_{1,0} + 2\beta_{2,0} & \geq 0 \\ \alpha_{0,0} + \alpha_{0,1} & \geq 0 & \beta_{1,0} + \beta_{1,1} & \geq 0 \\ \alpha_{0,0} + \alpha_{1,0} + \alpha_{0,1} + \alpha_{1,1} & \geq 0 & \beta_{1,0} + 2\beta_{2,0} + \beta_{1,1} + 2\beta_{2,1} & \geq 0 \end{array}$$

Obviously, necessary and sufficient conditions for constraining a multivariate regression polynomial to be monotone over some domain are highly non linear and very hard to

handle, as soon as the number of variables and/or the degree of the polynomial is greater than 2. The following result states in a general case, whatever the number of variables and the degree of the polynomial, sufficient conditions for constraining the polynomial to be monotone over the whole domain of the input variables. If the maximum degree for each variables is 1, than these conditions are also necessary.

In the following, let $P_{(10\cdots)}(x_1, \cdots, x_v) = \sum_{i_1 \leq d_1, \cdots, i_v \leq d_v} \alpha_{i_1 \cdots i_v} x_1^{i_1} \cdots x_v^{i_v}$ be the derivative w.r.t. x_1 of some polynomial $P(x_1, \cdots, x_v)$, where the maximum degree for the i-th variable in $P_{(10\cdots)}$ is d_i, and the total number of monomials m. The $\alpha_{i_1 \cdots i_v}$ are introduced to render the proposition (3.1) more general and to avoid to deal with the coefficients coming from the derivation of the $x_1^{i_1}$ when the exponent i_1 is between 1 and d_1.

The following proposition gives a way to reduce the number of constraints in (2) from 2^m to a maximum of $\prod_{i=1,v} (d_i + 1)$.

PROPOSITION 3.1 *If*

$$\forall (j_1, \cdots, j_v) \in [0, d_1] \times \cdots \times [0, d_v], \sum_{i_1 \leq j_1, \cdots, i_v \leq j_v} \alpha_{i_1 \cdots i_v} \geq 0,$$

Then,

$$\forall (x_1, \cdots, x_v) \in [0, 1]^v, \sum_{i_1 \leq d_1, \cdots, i_v \leq d_v} \alpha_{i_1 \cdots i_v} x_1^{i_1} \cdots x_v^{i_v} \geq 0.$$

If $\max_{i=1,v}(d_i) = 1$, *then the previous condition is also necessary.*
The maximum number of constraints is $\prod_{i=1,v} (d_i + 1)$.

The sufficient part of the proposition is proved in appendix A. The necessary conditions are easily deduced when the maximum degree for each variable is one, since they are obtained when each variable takes the value 0 or 1. The maximum number of constraints is the product of the number of possible values for each $(j_i)_{i=1,v}$.

In the previous example, the number of variables in $P_{(10)}(x)$ is 2 and the maximum degree for each variable is 1. Therefore, the expected number of constraints is 4. The set of constraints in this case has been already given.

To give an idea of how much it reduces the number of constraints, anticipating a little bit one of our industrial example about a real example of radiative shock experiments, in section (4.4), a degree 3 polynomial with 6 variables is needed. The response should be monotone with to respect to every six variables, three of them inducing an increase of y and the other three a decrease. This polynomial includes 84 monomials. If all the terms are kept, with our methodology, a single monotony requirement will give rise to $3^6 = 729$ constraints instead of the 2^{84} initial. As explained in section (4.4), since 6 monotony constraints are required, we need (only) $6 * 3^6 = 4374$ linear inequalities.

3.4. *Optimization*

The optimization problem is solved with the active set algorithm which is standard in QP problems. This method gives an exact solution for which all the constraints are fulfilled and is preferred in this case to other methods since no approximation is required: with the still great number of constraints involved, slight approximations in the solution may lead to some inequalities being not verified, and violation of monotony requirements.

Caution must be taken since the constraints are collinear and their high number may induce numerical difficulties. We note β_{ls} the least squares solution of the unconstrained

problem, and X the matrix of predictors with n lines and m columns. $X = US^tV$ is the singular value decomposition of X, where S is the diagonal matrix of singular values, U and V unitary matrices. The constraints can be put in matrix form as $C\beta \geq 0$. Taking $\beta' = S^tV\beta$ by means of variable change in the parameter space, the least square solution is now given by $\beta'_{ls} =^t Uy$ and the matrix of constraints become $C' = CVS^{-1}$

With this suitable replacement, the problem (2) is recast in :

$$\underset{\beta'}{argmin}||\beta'_{ls} - \beta'||^2, \text{s.t. constraints } C'\beta' \geq 0 \tag{3}$$

$\beta' = 0$ leading to $\beta = 0$ always fulfills all the constraints, meaning that the constraints form a cone, and giving an easy starting point to the algorithm. The solution in this formulation is the orthogonal projection of β'_{ls} onto the cone of constraints.

A final remark is worth mentioning: the solution β_{sol} to (3) will be sparse. Indeed, in the parameter space, the equation $\sum_{i=1,n}(y_{(i)} - P(x_{(i)}))^2 = cst$, where cst is a constant, describes an (hyper)ellipsoid. Due to the well known Karish-Kuhn-Tucker conditions (see [16]), β_{sol} is the point where the ellipsoid is tangent to the cone of constraints. At this point, some of the constraints will be active, that is equal to zero. But constraints in C very often differ only from each other by a single coefficient. Suppose that two constraints C_1 and C_2 corresponding to two lines i_1 and i_2 in the matrix C differ only at the j-th column, and are active at the same time, giving two equations $C_{i1}\beta = 0$ and $C_{i2}\beta = 0$. Then the corresponding coordinate β_j of the vector β will be zero. Due the large number of constraints, this situation will occur more often than not, and result in zero coefficients in the solution.

4. Examples

4.1. *Simulated example in dimension 1*

In this example, 100 points are generated from the equation $y = -6x^3 + 10x^2 - 3x$, on the interval $[0, 1]$. A random gaussian noise of standard deviation 0.1 is added to y. Results are shown on the following figure.

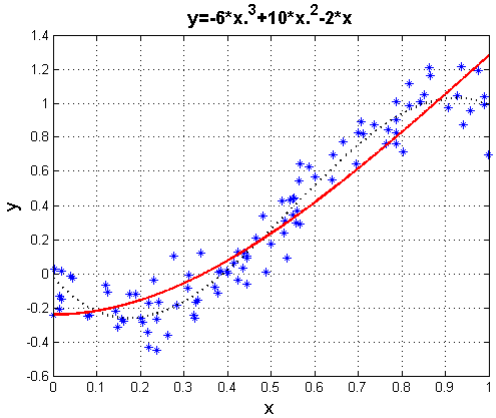


Figure 3. regression in dim 1 with a degree 3 polynomial ($d = 3$), and a monotony constraint on x_1

The dashed black is the calculated regression function without any constraint, assuming a degree 3 polynomial. The plain red line is the regression function when the function is supposed to be increasing with a positive concavity.

4.2. *Simulated example in dimension 2*

100 points are generated with the equation $y = -6x_1^3x_2 + 10x_1^2 - 3x_1$. A gaussian noise with a standard deviation of 0.1 is again added to y.

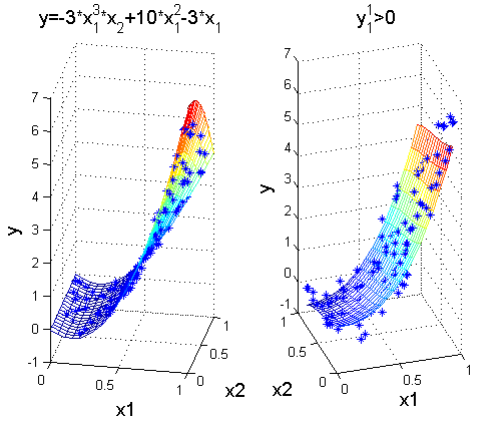


Figure 4. regression in dim 2 with a degree 3 polynomial ($d = 3$), and a monotony constraint on x_1

It can be seen that y is first decreasing with x_1 and then increasing. On the left panel, the original function is plot. On the right panel, we show the calculated regression with the constraint that y should increase with x_1. The figures are rotated to clearly show the behavior of the original and calculated functions.

4.3. *Real example: hydrotreatment of naphta*

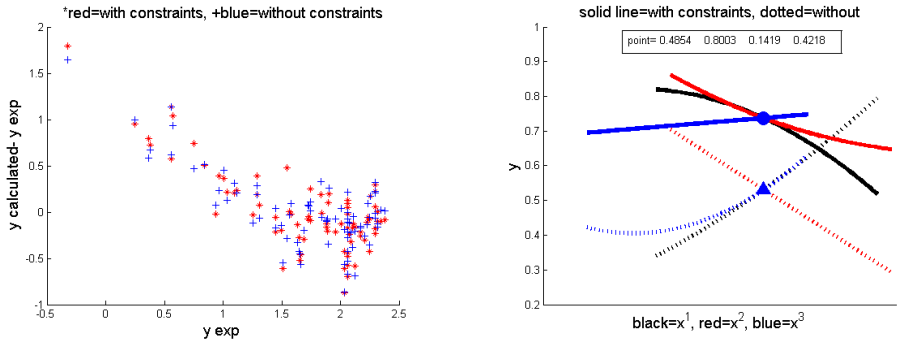
In petroleum process engineering, hydrotreating consists in treating a petroleum cut under hydrogen pressure in an industrial reactor. After being extracted, the oil coming from the underground has first to be refined and fractionated in different cuts and then to be prepared for a future commercial use. Specifically, in naphtha cuts, impurities (mainly sulphur and nitrogen) must be removed, before any further use.

A pseudo-kinetic model is commonly proposed to approximate this process and is given by the following equation :

$$ln(\frac{C}{C_0}) = -k.t.exp(-\frac{E_a}{RT}).P_{H_2}^m.P_{H_2S}^s$$

with the following variables :
C the concentration of the chemical to be removed remaining at the outlet of the reactor
C_0 the initial concentration
T the temperature of the process
P_{H_2} the partial hydrogen pressure
P_{H_2S} the partial H_2S pressure, since the reaction is inhibited by the presence of H_2S inside the reactor
t the contact time. In fact the real quantity followed by the experimenters is named LHSV for Liquid Hourly Space Velocity, is defined as the volumic rate of the naphta feed at the inlet divided by the volume of the catalyst bed and is equal to $1/t$.
k, E, m and s are parameters and must be estimated from experimental measurements.

Taking logarithm on each side of this formula, the equation can be easily linearized and rewritten $y = \sum_{i=1,4} \beta_i x_i$, where $y = ln(-ln(\frac{C}{C_0}))$, $x_1 = 1/T$, $x_2 = ln(LHSV)$, $x_3 = ln(P_{H_2})$, $x_4 = ln(P_{H_2S})$.



(a) Residue diagram. (b) Spider plot.

Figure 5. HDS Data and Regression. The left panel compares residues obtained by regressing with constraints in red to those obtained without constraints in blue. The right panel shows how the response varies from a given point. Solid lines are for the constrained regression and dotted for the unconstrained one

But unfortunately, this expression is unable to take into account the full complexity of the process, and empirical terms must be added. Finally, a degree 2 polynomial in the variables $x = (x_1, x_2, x_3, x_4)$ is postulated. Moreover some constraints must be respected: the process is more efficient (which means that C decreases or equivalently y increases) when :

- the temperature T increases or x_1 decreases
- $LHSV$ decreases or x_2 augments
- P_{H_2} or x_3 goes higher.

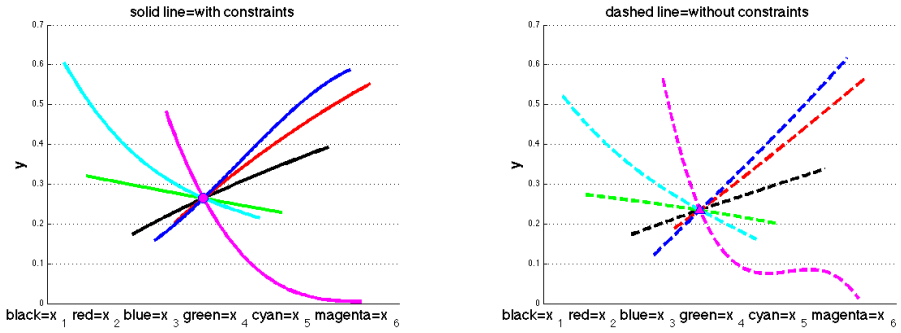
In figure 5 we compare the results when regressing with and without constraints. The left panel exhibits the residues (y calculated - y experimental), showing only minor differences when the experimental points are predicted by both methods: Root Mean Square Error is RMSE=0.438 with constraints and 0.411 without. But the obtained equations are really different as shown on the right.

On the right panel, we see a kind of spider plot, showing the behavior of the response when only one variable varies at a time, starting from a given point in the domain (here: $[x_1 = 0.71, x_2 = 0.64, x_3 = 0.174, x_4 = 0.062]$). The dotted lines correspond to the regression without constraints, the solid line to the regression with constraints. The plain triangle marks the response for the regression without constraints, the circle for the regression with constraints. x-axis are translated so that every curve crosses at the center of the graphic. Black lines correspond to variations along T or x_1, red lines to variations with $LHSV$ or x_2, blue lines to variations with P_{H_2} or x_3. Behaviors for the regressions without constraints are obviously wrong: the black dotted line is increasing instead of decreasing and the blue has a minimum.

4.4. *Real example: radiative shock experiments*

Magnetic cataclysmic variables are binary systems containing a magnetic white dwarf which accretes matter from a secondary star. The radiation collected from these objects mainly comes from an area near the white dwarf surface, named the accreted column, which is difficult to observe directly. The POLAR experiments aim is to mimic this shock formation in laboratories using high-power laser facilities as described in [17]. The plasma produced by the laser beams collides with an obstacle, and the reverse shock produced is similar to the astrophysical one.

Numerical simulations of these experiments are performed at CEA/DAM le-de-France with the laser-plasma interaction hydrodynamic code FCI2. A set of about 2000 numerical experiments were run with six input variables varying on the interval [0,1] after



(a) With constrained. (b) Without constraintst.

Figure 6. polynomial fit to the synthetic data of radiative shock experiments: spider plot for the constrained regression on the left panel and spider plot for the unconstrained multivariate regression on the right

renormalisation. For clarity in this paper, these variables are named x_1 to x_6. The data have been kindly provided to us by Jean Giorla.

The variables x_1 to x_3 describe the 1D-geometry (thicknesses of the two target layers and distance between the target and the obstacle) and the variables x_4 to x_6 are relative to the absorbed energy (the laser power and duration, a physical parameter involved in the electronic diffusion equation). Physical reasons indicate that the collision time y of the plasma impacting the obstacle is monotonically increasing with the first three variables and decreasing with the three others.

200 observations among the 2000 available ones were extracted by Latin hypercube sampling techniques to construct the models. The following figure 6 shows the results, assuming a degree 3 polynomial. The lines correspond to the conditional mean of the response with respect to the indicated variable. The plain lines on the left panel correspond to the proposed methodology and the dotted lines on the right to a multivariate linear regression on the same data. While the general behaviors of the curves are very similar, we can see that the magenta curve for x_4 on the right panel is not monotone. In this example, the $RMSE$ calculated over the remaining1800 values changes from 0.006 for the unconstrained case to 0.014 for the constrained regression, that is approximately two times higher.

4.5. *Shape requirements*

As in [1] or [12], the same method can be applied as long as the corresponding constraints stay linear with respect to the coefficients of the polynomial model. This includes :

- monotony constraints;
- concavity or convexity constraints as they result on an upper or lower zero bound on the second derivative, which remains a polynomial;
- bound constraints on the function itself;
- equality constraints;
- any kind of linear constraint on the coefficients.

An other advantage of the method is that expert knowledge can be incorporated in the polynomial to more easily obtain the desired behavior. If one expects a linear variation with respect to the first variable, while the second variable should correspond to a third degree polynomial, then the corresponding terms can be omitted in the fit to force the response to present the correct shape. This could have been done in the radiative shock experiments example for the second response (in red in figure 6).

However, some problems, clearly, would not correspond to this method. For example, consider the function $y = 1 - 4(x - 1/2)^2$, drawn on the figure 4.5 in black. At $x = 1/2$, this function reaches its maximum, $y = 1$. Twenty values for x are drawn uniformly on $[0, 1]$, and a random gaussian noise of standard deviation 0.1 is added to the resulting values of y. The points are shown in green square on the figure 4.5. They are fitted with a 2 degree polynomial, drawn in red, with the additional constraint that the maximum should not exceed 1.

We can see that the obtained fit respects the constraint, but is obviously not what is expected: constraints seem too stringent.

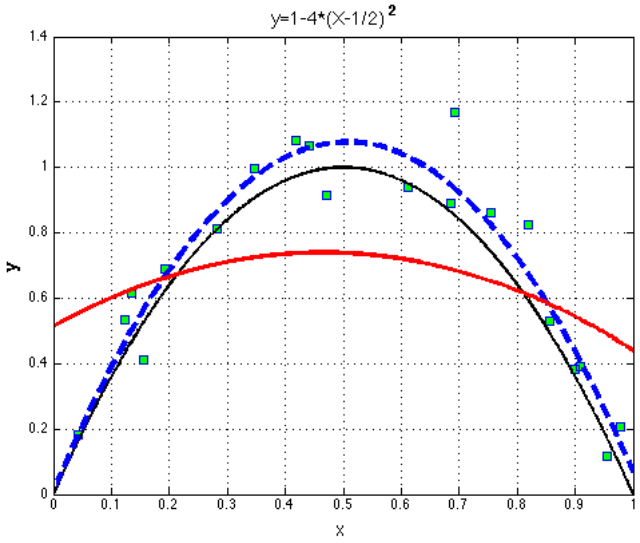


Figure 7. fit of the function $y = 1 - 4(x - 1/2)^2$. The original function is in black. The fitted least square 2 degree polynomial is in blue, the obtained constrained function with a maximum not exceeding 1 in red.

To conclude, the proposed procedure is adapted to polynomial regression, a problem occurring very often in industrial applications, specially with few available experimental data and in multidimensional cases. It should be understood that the response should vary smoothly enough, with no discontinuity in the response and its first derivative. The proposed methodology is very flexible, easy to understand for practitioners and well adapted to industrial problems.

Acknowledgements

The author would like to thank Jean Giorla from CEA/DAM le-de-France for giving access to the data of the radiative shock experiments and Damien Hudebine from IFPEN for the example in hydrotreatment

References

[1] F Lauer and G Bloch. Incorporating prior knowledge in supportvector regression. Machine Learning. 2008;.
[2] Barlow, R, Bartholomew, D, Bremner, J and Brunk, H. Statistical inference under order restrictions: Theory and application of isotonic regression. John Wiley and Sons Inc; 1978.
[3] Burdakov O, Syoev O, Grimvall G, Hussian M. An o(n) algorithm for isotonic regression

problems. The Proceedings of the 4th European Congress of Computational Methods in Applied Science and Engineering 'ECCOMAS 2004', Ed P Neittaanmaki et al. 2004;.
[4] Ramsay J. Monotone regression splines in action. Statistical Science. 1988;3(4):425–461.
[5] Hall P, Huang L. Nonparametric kernel regression subject to monotonicity constraints. Annals of Statistics. 2001;29:624–647.
[6] Dette, H Neumeyer, N, and Pilz, KF. A simple nonparametric estimator of a monotone regression function. Bernoulli. 2006;12:469–490.
[7] Turlach M, Marron JS, Turlach BA, Wand MP. Local polynomial smoothing under qualitative constraints. In: L Billard and NI Fisher (Eds), Graph-Image-Vision. 1997;28:647–652.
[8] E Mammen, J S Marron, B A Turlach and M P Wand. Ageneral projection framework for constrained smoothing. Statistical Science. 2001;16(3):232–248.
[9] Scheder R. Shape constraints in multivariate regression [dissertation]. Ruhr University; 2007.
[10] Burdakov O, G G, Hussian M. Generalized a generalized pav algorithm for monotonic regression en several variables. Nonconvex Optimization and Its Applications, Springer-Verlag. 2006;.
[11] Racine J, Parmeter C. Constrained nonparametric kernel regression: estimation and inference. disponible sur http://webuvicca/econ/research/papers/pdfs/racinepdf. 2008;.
[12] Da Veiga S, Marrel A. Gaussian process modeling with inequality constraints. Annales de la facult des sciences de Toulouse Mathmatiques. 2012 4;21(3):529–555.
[13] Anestis A, Bigot I Jand Gijbels. Penalized wavelet monotone regression. Statistics and Probability Letters. 2007;77(16):1608 – 1621.
[14] Turlach B. On monotone regression. In: NZSA 2011 Conference, 62nd Annual Meeting of the New Zealand Statistical Association; August. Auckland, New Zealand: in the invited session on Smoothing and Applications, NZSA 2011 Conference; 2011.
[15] Wang J, Ghosh S. Shape restricted nonparametric regression with bernstein polynomials. Computational Statistics and Data Analysis. 2012;56(9):2729–2741.
[16] Nocedal J, Wright S. Numerical optimization. 2nd ed. New York: Springer; 2006.
[17] Falize E, the PAMAL group. Recent advances in the experimental simulation of x-ray binary stars accretion shocks. In: 9th Conference on High Energy Density Laboratory Astrophysics; May; Tallahassee, Florida; 2012.

Appendix A. demonstration of proposition1

In this section, we first prove Proposition 3.1 by induction, and discuss a few about simple possible ameliorations we do not want to develop in this paper, for computational reasons.

For simplicity reasons, this proposition is written for a polynomial

$$P(x) = \sum_{i_1 \leq d_1, \cdots, i_v \leq d_v} \alpha_{i_1 \cdots i_v} x_1^{i_1} \cdots x_v^{i_v}$$

in which the i-th variable is at most of degree d_i, and where x stands for $(x_1, \cdots, x_v)$. This implies that the resulting polynomial is at most of degree $\prod_{i=1,v} d_i$. This statement includes polynomials of degree d (for example quadratic polynomials) since in this case the coefficients for which the sum of the corresponding exponents $\sum_{j=1,v} i_j > d$ will be equal to zero.

In the following, i^+ means $sup(i, 0)$ for some integer i.

In a preparatory lemma (lemma A.1), we consider the polynomial $R(x)$, constructed from the initial $P(x)$ in which the exponent of variable i_1 (respectively$\cdots i_v$) has been decremented by k_1 (respectively$\cdots k_v$) for some integers $0 \leq \ k_1 \leq d_1, \cdots, 0 \leq \ k_v \leq d_v$

when it is possible:

$$R(x) = \sum_{i_1 \leq d_1, \cdots, i_p \leq d_p} \alpha_{(i_1, \cdots i_p)} x_1^{(i_1 - k_1)^+} x_2^{(i_2 - k_2)^+} \cdots x_p^{(i_p - k_p)^+}$$

$S(x)$ and $T(x)$ result from the decomposition $R(x) = S(x) + x_1 T(x)$, in which we have assumed for convenience that $k_1 \geq 1$

$$S(x) = \sum_{i_1 \leq k_1, \cdots, i_p \leq d_p} \alpha_{(i_1, \cdots i_p)} x_2^{(i_2 - k_2)^+} \cdots x_p^{(i_p - k_p)^+}$$

$$T(x) = \sum_{k_1 < i_1 \leq d_1, \cdots, i_p \leq d_p} \alpha_{(i_1, \cdots i_p)} x_1^{(i_1 - 1 - k_1)^+} x_2^{(i_2 - k_2)^+} \cdots x_p^{(i_p - k_p)^+}.$$

LEMMA A.1 *decreasing one degree*
If

$$\forall x \in [0,1]^p, S(x) \geq 0 \text{ and } T(x) \geq 0$$

then $R(x) \geq 0$

The proof of lemma A.1 is immediate since x_1 takes its value in $[0, 1]$. We are now ready for the demonstration of Proposition 3.1 which is first recalled.

PROPOSITION A.2 *If*

$$\forall (j_1, \cdots, j_v) \in [0, d_1] \times \cdots \times [0, d_v], \sum_{i_1 \leq j_1, \cdots, i_v \leq j_v} \alpha_{i_1 \cdots i_v} \geq 0,$$

Then,

$$\forall (x_1, \cdots, x_v) \in [0,1]^v, \sum_{i_1 \cdots, i_v \leq d} \alpha_{i_1 \cdots i_v} x_1^{i_1} \cdots x_v^{i_v} \geq 0.$$

Proof. Proposition 3.1 is obviously verified for $n = 1$. By induction, we assume that Proposition 3.1 is demonstrated until $n - 1$ for some $n > 1$, and we want to prove that if for all $(n_1, \cdots, n_p)$ such that $\sum n_i \leq n$,

$$\forall j_1 \leq n_1, \cdots, j_p \leq n_p, \sum_{i_1 \leq j_1, \cdots, i_p \leq j_p} \alpha_{(i_1, \cdots i_p)} \geq 0,$$

then $R(x) \geq 0$ when x is in $[0,1]^p$.

We assume $j_1 - k_1 > 0$ for convenience and we decompose again $R(x)$ as $R(x) = S(x) + x_1 T(x)$. Since our induction hypothesis are verified for both $S(x)$ and $T(x)$, $S(x) \geq 0$ and $T(x) \geq 0$ and we apply Lemma A.1 to get the result. Otherwise if $j_i - k_i = 0, \forall i$, then $R(x)$ is equal to $\sum_{i_1 \leq j_1, \cdots, i_p \leq j_p} \alpha_{(i_1, \cdots i_p)}$, and this quantity has been assumed to be greater or equal to 0. ■