Learning with High and Low Accuracy Observations.

GdR MascotNum, IHP, Paris

Federico Zertuche[1], Celine Helbert[2] and Anestis Antoniadis[1] .

May 17, 2013.

[1]LJK, UJF, Grenoble.
[2]EC ICJ, Lyon.

Context and Notation

The main problem is to predict and output y given an input x.

Perform well on average:

- Output and input modelled by random variables $\mathbf{X}$ and $\mathbf{Y}$.
- The best prediction is $\mathbb{E}[\mathbf{Y}|\mathbf{Y}(X) = y]$

The given data $(X, Y) : ((x_1, y_1), ..., (x_n, y_n))$ are called observations. We will try to learn from the observations by devicing an strategy and we will try to determine how good it is.

We present 2 alternatives to the linear Cokriging method:

- One in which we estimate the relationship by local polynomials and
- One in which we use adaptive wavelets.

Learning With Gaussian Processes.

High and Low Accuracy Observations.

- The Linear Model.
- Non-linear Model.

Coarse to Fine Wavelet Regression

Learning with Gaussian Processes

- Observations of the form (X, Y) where X is deterministic ,
- and Y is modeled by a Gaussian process $\mathbf{Y}$ that depends on x.

Sometimes we will note $\mathbf{Y}(x)$ instead of $\mathbf{Y}$ to highlight this dependence.

Learning with Gaussian Processes

We had that $\mathbf{Y}(x) \sim \mathcal{GP}(m(x), k(x, x'))$.

The best estimation and error of estimation are

- $\widehat{y_*} = \mathbb{E}[\mathbf{Y}(x_*)|\mathbf{Y}(X) = Y]$ and
- $\widehat{\sigma^2}(y_*) = Var[\mathbf{Y}(x_*)|\mathbf{Y}(X) = Y]$.

With explicit formullas

- $\widehat{y_*} = m(x_*) - k(x_*, X)k(X, X)^{-1}(Y - m(X))$ and
- $\widehat{\sigma^2}(y_*) = k(x_*, x_*) - k(x_*, X)k(X, X)^{-1}k(x_*, X)^T$.

Learning with Gaussian Processes

The covariance and mean functions are parametrized.

- $m(x) = \mu_c$
- $k(h) = \sigma^2 \prod_{k=1}^{n} g_k(h^2; \theta^k)$ for $h = (x^k - x'^k)$ and $x \in \mathbb{R}^n$.

Estimate all the parameters to build the prediction. Maximize minus the log-likelihood of $\mathbf{Y}(X)$ at Y - which is a Gaussian Vector.

We note $k(h) = \sigma^2 g(h)$.

Learning with Gaussian Processes

To sumarize:

- Define a mean and covariance function.
- use the likelihood of the parameters of $\mathbf{Y}(X)$ at Y to estimate them.
  - In our case μ, σ^2 are constants and if $x \in \mathbb{R}^n$, $\theta \in \mathbb{R}^n$
- make a prediction by using the formulas above.

The problem of learning with Gaussian processes is the problem of learning the free parameters of the mean and covariance function.

Learning with G.P. when low and high accuracy responses are available.

The main objectve: predict an output given two sets of observations of the same type.

For example, we solve

$$y' = sin(x^4 y^2), x \in [0, 100] \tag{1}$$

by using Euler's method with two different discretization steps $h_l = 2h$ and

$h_h = h$ to obtain $\tilde{y}_l$ and $\tilde{y}_h$.

Learning with G.P. when low and high accuracy responses are available.

$\tilde{y}_l$ is easier to calculate but less accurate than $\tilde{y}_h$. That is why we consider an observation set as follows:

- For h_l, $X_l : 0, 2h, 4h, ..., 100,\ Y_l : \tilde{y}_l(0), \tilde{y}_l(2h), ..., \tilde{y}_l(100)$.
- For h_h, $X_h : 0, 6h, 12h, ..., 100,\ Y_h : \tilde{y}_h(0), \tilde{y}_h(6h), ..., \tilde{y}_h(100)$.

X_l has more elements than X_h. So is Y_l with respect to Y_h.

X_h is not necessarily a subset of X_l. In fact, $X_h \not\subset X_l$ is a more general setting and more convenient in terms of exploration.

We try to learn y_h as it is more accurate.

Learning with G.P. when low and high accuracy responses are available.

We try to make a prediction at a point (x_*^h, y_*^h) related to $\tilde{y}_h$ by using all the data available.

- $\widehat{y_*^h} = \mathbb{E}[\mathbf{Y}_h(x_*^h)|\mathbf{Y}_h(X_h) = Y_h(X_h), \mathbf{Y}_l(X_l) = Y_l(X_l)]$
- $\widehat{\sigma^2}(y_*^h) = Var(\mathbf{Y}_h(x_*^h)|\mathbf{Y}_h(X_h) = Y_h(X_h), \mathbf{Y}_l(X_l) = Y_l(X_l))$

Learning with G.P. when low and high accuracy responses are available.

As before, the mean and covariance functions of $\mathbf{Y}_l$ and $\mathbf{Y}_h$ depend on μ_l, σ_l^2 and θ_l and μ_h, σ_h^2 and θ_h respectively.

To estimate them we would like to maximize the likelihood of $(\mathbf{Y}_l(X_l), \mathbf{Y}_h(X_h))$ who is Gaussian vector.

Linear Model.

Once again, the observations are $(X_l, Y_l) : (x_1^l, y_1^l), \ldots (x_n^l, y_{nl}^l)$ and $(X_h, Y_h) : (x_1^h, y_1^h), \ldots (x_{nh}^h, y_{nh}^h)$

- X_l and X_h are deterministic;
- Y_l and Y_d are modeled by $\mathbf{Y}_l(x) \sim \mathcal{GP}(\mu_l, \sigma_l^2 g_l(x, x'))$ and $\mathbf{Y}_d(x) \sim \mathcal{GP}(\mu_d, \sigma_d^2 g_d(x, x'))$.
- Y_l and Y_d are independent;
- $Y_h(x) = rY_l(x) + Y_d(x)$.

Learning with G.P. when low and high accuracy responses are available: Linear Model.

We can estimate all the parameters. μ_l, σ_l^2 and θ_l and σ_d^2 and θ_d and (μ_d, r) by using

$$(\widehat{\mu_d}, \widehat{r}) = [N^T(\sigma_d^2 g_d(X_h, X_h))^{-1}N]^{-1}[N^T(\sigma_d^2 g_d(X_h, X_h))^{-1}Y_h(X_h)]$$

where $N = \begin{pmatrix} \mathbf{1}_{length(nh)} & Y_l(X_h) \end{pmatrix}^T$.

In order to estimate (μ_d, r) we need $Y_l(X_h)$.

Learning with G.P. when low and high accuracy responses are available: Linear Model.

It turns out that if $\mathbf{Y}_h(x) = r\mathbf{Y}_l(x) + \mathbf{Y}_d(x)$, the prediction and prediction error formulas are

$$r\mathbb{E}[\mathbf{Y}_l(x_*^h)|\mathbf{Y}_l(X_l) = Y_l(X_l)] + \mathbb{E}[\mathbf{Y}_d(x_*^h)|\mathbf{Y}_d(X_l) = Y_d(X_l)]$$

and

$$r^2 Var(\mathbf{Y}_l(x_*^h)|\mathbf{Y}_l(X_l) = Y_l(X_l)) + Var(\mathbf{Y}_d(x_*^h)|\mathbf{Y}_d(X_l) = Y_d(X_l)).$$

Non-linear model.

Is the linear model a good representation of the relationship between (X_l, Y_l) and (X_h, Y_h)?

Consider the following example in which we try to determine the influence of some parameters on the solution of a differential equation.

Non-linear model.

Is the linear model a good representation of the relationship between (X_l, Y_l) and (X_h, Y_h)?

Consider the following example in which we try to determine the influence of some parameters on the solution of a differential equation.

Non-linear model.

For each $t \in \{1, \ldots, n\}$, solve

$$\begin{cases} a^2 \nabla_x^2 p(th, x) = \dfrac{p(th, x) - p((t-1)h, x)}{\Delta t}, \forall x \in \Omega \\ \nabla_x p(th, x) \cdot n = 0, \forall x \in \partial\Omega_1 \cup \partial\Omega_2 \cup \partial\Omega_3 \\ \nabla_x p(th, x) \cdot n = 1, \forall x \in \partial\Omega_0 \\ p(0, x) = 1, \forall x \in \Omega \end{cases} \quad (2)$$

for $p(th, x)$. where $a^2 = \frac{k}{\gamma(nC_f + C_s)}$ and n is orthogonal to the border of the domain $\partial\Omega$.
The domain Ω is a rectangle with sides $\partial\Omega_1, \partial\Omega_2, \partial\Omega_3$ and the top side $\partial\Omega_0$.

Non-linear model.

For each value $\tilde{\pi} = (\tilde{k}, \tilde{\gamma}, \tilde{C}_f)$ and $t \in \{1, \ldots, n\}$, we solve the projected problem on E_l and E_h. We consider the maximum value of the response on space for $t = n$, that we note $\max_x p(n, x)$, as the responses $Y_l(\tilde{\pi})$ and $Y_h(\tilde{\pi})$.

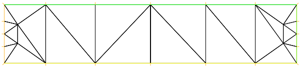


Figure : E_l discretization grid.

Non-linear model.

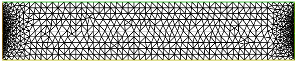


Figure : E_h discretization grid.

Non-linear model.

The responses $Y_l(\Pi)$ and $Y_h(\Pi)$ are plotted on figure 3 below. The relationship is clearly non-linear.

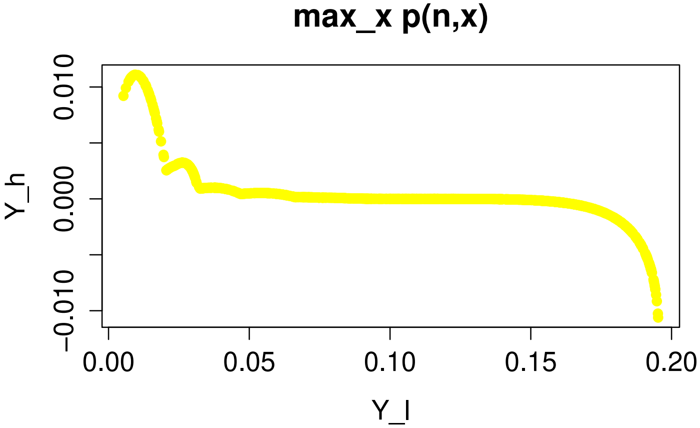


Figure : $Y_l(\Pi)$ versus $Y_h(\Pi)$.

Local-Polynomial Regression

Use the observed data (X_l, Y_l) and (X_h, Y_h) to estimate φ by using locally linear polynomials.

Set $\mathbf{Y}_h = \widehat{\varphi}(\mathbf{Y}_l(x)) + \mathbf{Y}_d(x)$.

The estimated relationship $\widehat{\varphi}$, is locally linear.

Local-Polynomial Regression

The parameters of $\mathbf{Y}_l$ are estimated by maximizing the likelihood of the given observations Y_l.

To estimate those of $\mathbf{Y}_d$, we first fit $\widehat{\varphi}$ using (X_l, Y_l) and (X_h, Y_h).

Once we have a formula for $\widehat{\varphi}$, we set Y_d as $Y_h - \widehat{\varphi}(Y_l)$.

We use the likelihood of Y_d to build the corresponding parameter estimators. Finally, we build the prediction and error formulas by plugging in the estimates on the equations of **Proposition 2**.

Local-Polynomial Regression

For $x \in [0, 3]$ let

$$f_l(x) = 3\sin(x) + 1 \tag{3}$$

$$f_h(x) = \sin(3x). \tag{4}$$

Local-Polynomial Regression

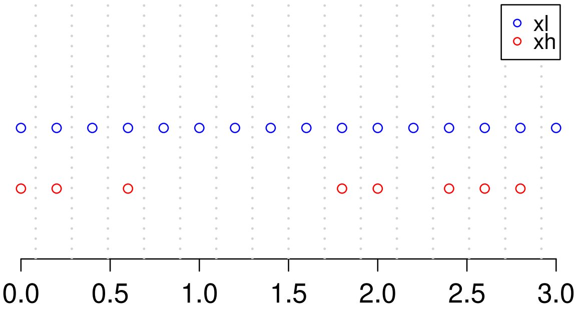
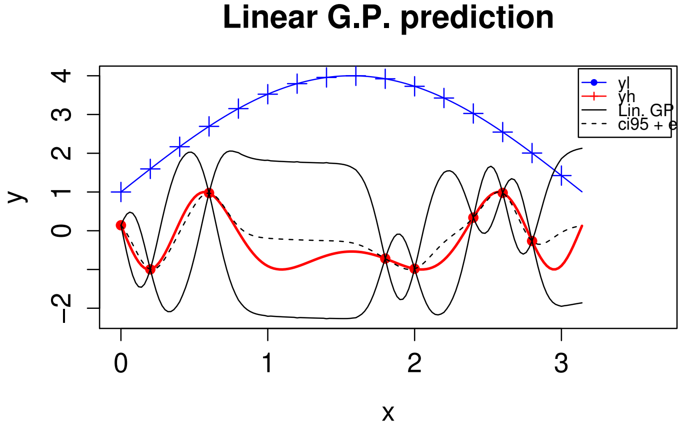


Figure : Locations X_l and X_h used on figures 6, 7 and 5.

Local-Polynomial Regression

Figures 6 and 7 show the result obtained by applying the linear and non-linear learning procedures to the observations made over the points on figure 4.



Local-Polynomial Regression

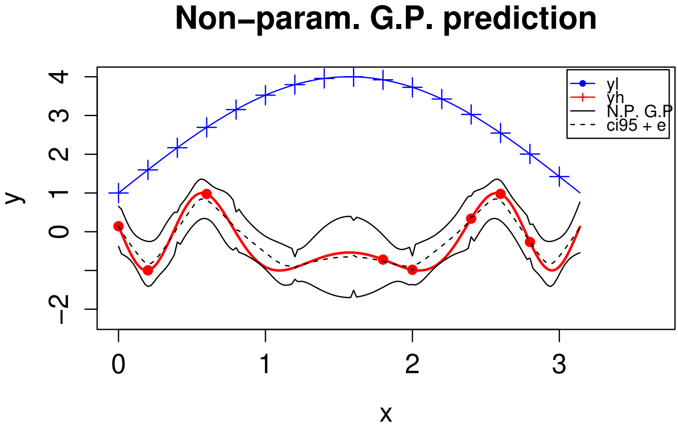


Figure : Fit produced by the non parametric learning method.

Coarse to Fine Wavelet Regression

A coarse-to-fine algorithm to build a prediction using adaptive wavelets when high accuracy and low accuracy inputs are available is proposed as an alternative to the Gaussian process method. [CnK06, CK03]

Wavelet Regression

Wavelets functions are built from a single compactly supported function Ψ by scaling and translating it as shown on figure 8.

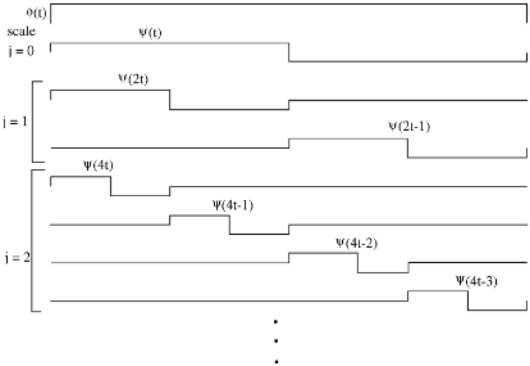


Figure : Several levels of the Haar wavelet. Each level i is formed by contracting and translating by a constant the wavelet functions of the previous scale i-1. (The image was taken from http://cnx.org/content/m10764/latest/) .

Wavelet Regression

Wavelets represent the details of a function at a scale or resolution. To explain the wavelet transform consider the following example:

Resolution	Averages	Detail Coefficients
4	[9 7 3 5]	
2	[8 4]	[1 -1]
1	[6]	[2]

The wavelet transform of [9 7 3 5] is [6 2 1 -1].

Wavelet Regression

Given a set of observations $(X, Y) : (x_1, y_1), \ldots (x_n, y_n)$, we solve the least squares problem

$$\sum_{i=1}^{n_j} (y_i - f(x_i))^2 \tag{5}$$

where

$$f(x) = \sum_{\lambda \in \Lambda} d_\lambda \psi_\lambda(x) \tag{6}$$

d_λ are unknown constants and ψ_λ are wavelet functions. To build an approximation of the unknown function that generated the observations.

Coarse to Fine Wavelet Regression

We will chose the wavelet basis functions ψ_λ by looking at the size of its corresponding coefficients d_λ and the number of points in their support.

If there are not enough points, we will add observations where needed.

Coarse to Fine Wavelet Regression

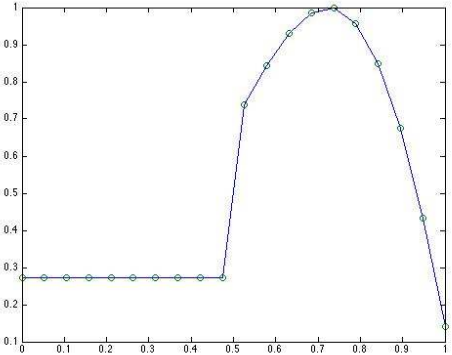


Figure : Test function with 20 observation points.

Coarse to Fine Wavelet Regression

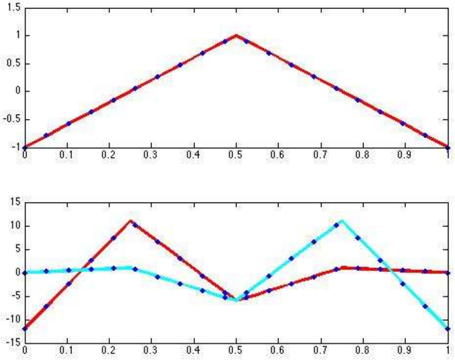


Figure : Initial wavelet basis with observation points.

Coarse to Fine Wavelet Regression

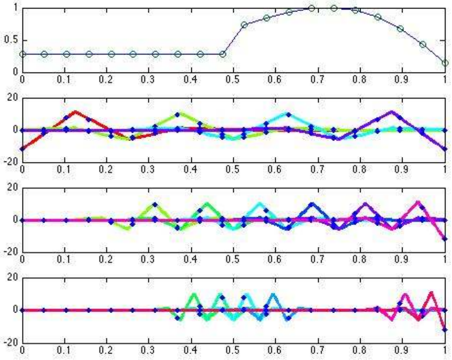


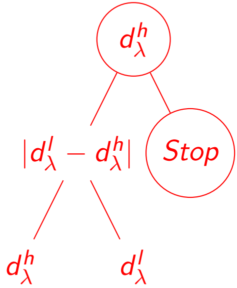
Figure : Chosen wavelet functions.

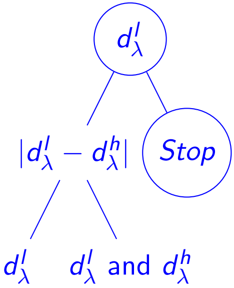
Multi-Fi. Coarse to Fine Wavelet Regression

We will use the low accuracy observations (X_l, Y_l) to help us to determine where to explore F_h to improve our approximation f_h. The idea is that in order to solve the minimization problem (5) after we added some wavelets we will need, eventually, to add observations.

Because observations generated by F_l are easier to obtain, we would prefer to explore F_l where it is similar to F_h. For that, we determine the coefficients related to each data set. We note them d_λ^l and d_λ^h respectively. Then, we determine which wavelets to add as follows:

Multi-Fi. Coarse to Fine Wavelet Regression





Multi-Fi. Coarse to Fine Wavelet Regression

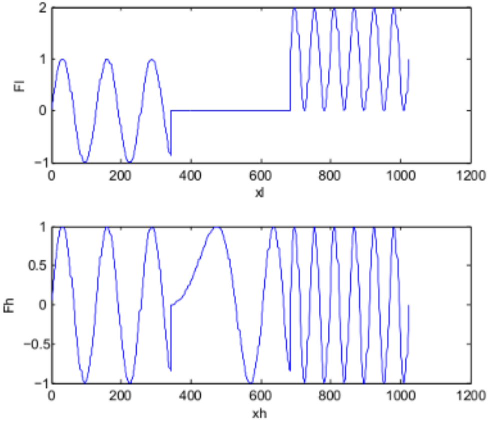


Figure : F_h and F_l are the same on the first third of the domain. F_l is a rough approximation of F_h on the second third and a translation on the y-axis on the last third.

Multi-Fi. Coarse to Fine Wavelet Regression

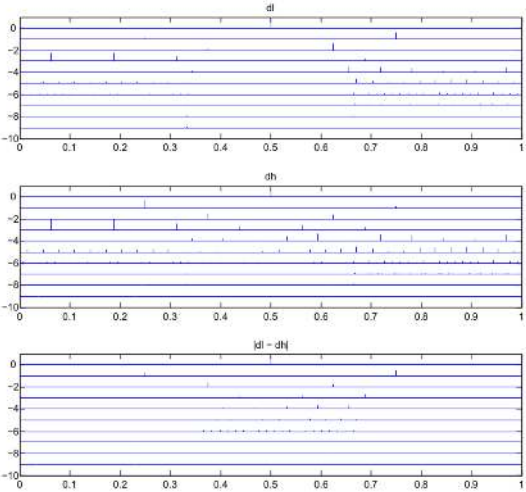


Figure : The fist plot are the wavelet coefficients of the approximation f_l of the

Multi-Fi. Coarse to Fine Wavelet Regression

Suppose that we start at level -1 on the first plot of figure 13. We see on the left a small coefficient. The recursive algorithm would stop at level -1. But, as we can see, there are 3 big coefficients on level -3. The idea is to design an statistical test to determine when to refine the decomposition based on the the articles [AG02, AA04].

Also in [AA04, AG02] a method to find the discontinuity points of an unknown function using wavelets is proposed. Applying such method on the example would help us to determine the form of the subsets of $[0, 1]$ in which F_l is similar to F_h.

Felix Abramovich and Anestis Antoniadis.
Optimal testing in a fixed-effects functional analysis of variance model.

International Journal of . . ., 2(4):324—-349, 2004.

A Antoniadis and Irene Gijbels.
Detecting abrupt changes by wavelet methods.
Journal of Nonparametric Statistics, 14(No 1-2):7–29, 2002.

D Castano and A Kunoth.
Adaptive fitting of scattered data by spline.
Curves and Surfaces, 2003.

Daniel Castaño and Angela Kunoth.
Robust regression of scattered data with adaptive spline-wavelets.
IEEE transactions on image processing : a publication of the IEEE Signal Processing Society, 15(6):1621–32, June 2006.