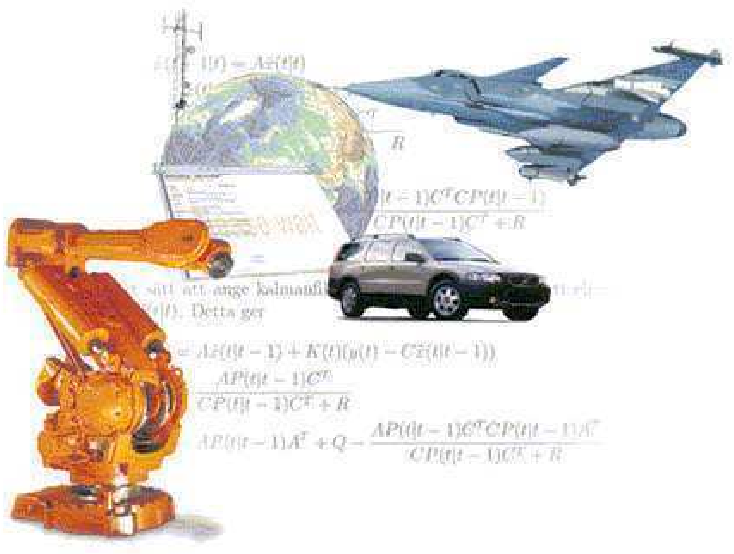
Semi-Supervised Regression and System Identification



Lennart Ljung
Coauthor: Henrik Ohlsson

Division of Automatic Control
Linköping University
Sweden

The Problem

Given a standard regression problem:

$$y(t) = f(\varphi(t)) + \text{noise}$$

f unknown, $y(t)$ and $\varphi(t)$ observed for $t = 1, \ldots, N_L$. Find f!

Given a standard regression problem:

$$y(t) = f(\varphi(t)) + \text{noise}$$

f unknown, $y(t)$ and $\varphi(t)$ observed for $t = 1, \ldots, N_L$. Find f!

Or rather, for any given φ^* find a good value of $f(\varphi^*)$ ("Model on Demand", "Just in time model")

The Problem

Given a standard regression problem:

$$y(t) = f(\varphi(t)) + \text{noise}$$

f unknown, $y(t)$ and $\varphi(t)$ observed for $t = 1, \ldots, N_L$. Find f!

Or rather, for any given φ^* find a good value of $f(\varphi^*)$ (“Model on Demand”, “Just in time model”)

Extra feature: We have several measurements of $\varphi(t), t = N_L + 1, \ldots, N_L + N_U$ without corresponding values of $y(t)$ (“unlabeled observations”)

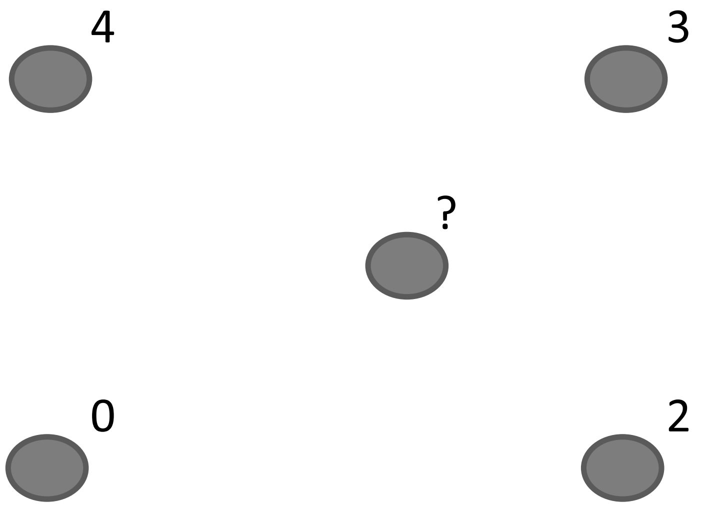
Information in Unlabeled Regressors?

Information in Unlabeled Regressors?

Yes, if the regressors live in a confined region, like a manifold:

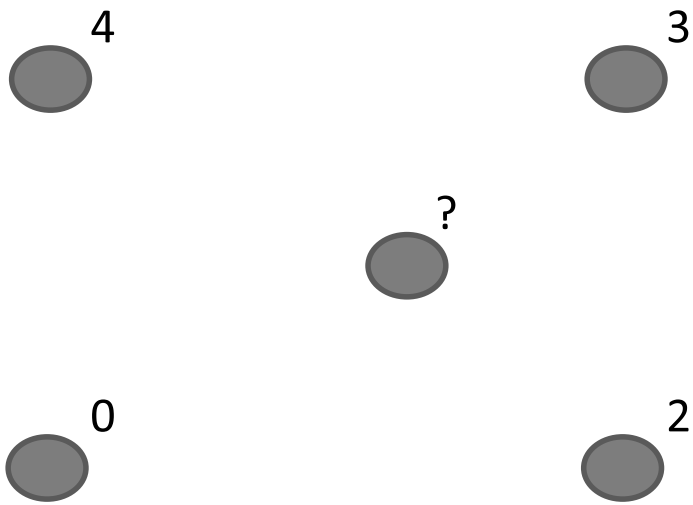
Information in Unlabeled Regressors?

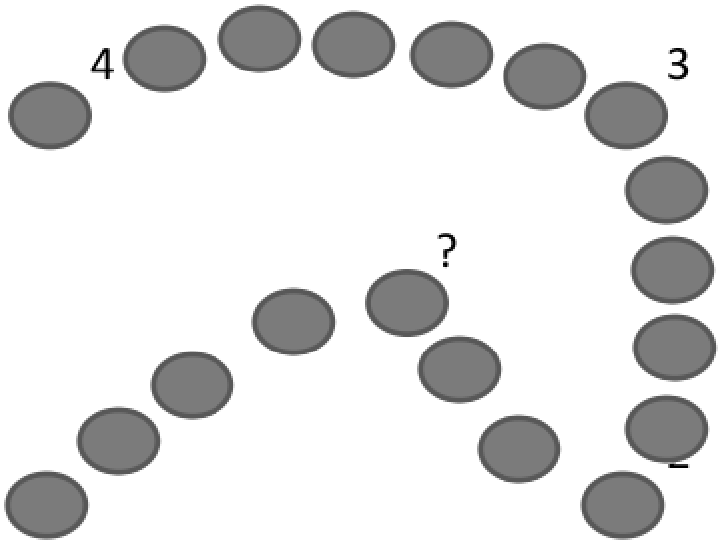
Yes, if the regressors live in a confined region, like a manifold:



Information in Unlabeled Regressors?

Yes, if the regressors live in a confined region, like a manifold:





(Un-/Semi-)Supervised

- Supervised: All Regressors labeled $(y(t), \varphi(t))$ [Standard Regression problems]
- Unsupervised: No labels known. [Clustering, Classification]
- Semi-supervised: Some labels known $\{y(t), \varphi(t), t = 1, \ldots, N_L\}$. Some additional regressors known without labels $\{\varphi(t), t = N_L + 1, \ldots, N_L + N_U\}$
- Estimation problem: still to "predict" $y^* = f(\varphi^*)$ for any given φ^*
  - ”predict”: .....

The Suggested Method: WDMR

WDMR: Weight Determination by Manifold Regression

The Suggested Method: WDMR

WDMR: Weight Determination by Manifold Regression

(Manifold) Smoothness Assumption: $f(\varphi_1)$ and $f(\varphi_2)$ close if φ_1 and φ_2 are close (in a relevant metric).

WDMR: Weight Determination by Manifold Regression

(Manifold) Smoothness Assumption: $f(\varphi_1)$ and $f(\varphi_2)$ close if φ_1 and φ_2 are close (in a relevant metric).

Problem: associate $\varphi(i)$ with good values $\hat{f}_t = f(\varphi(t))$ for all regressors, both labeled and unlabeled.

The Suggested Method: WDMR

WDMR: Weight Determination by Manifold Regression

(Manifold) Smoothness Assumption: $f(\varphi_1)$ and $f(\varphi_2)$ close if φ_1 and φ_2 are close (in a relevant metric).

Problem: associate $\varphi(i)$ with good values $\hat{f}_t = f(\varphi(t))$ for all regressors, both labeled and unlabeled.

Take care of two kinds of information:

- $f(\varphi(t)) \approx y(t)$ for the measured labels
- $f(\varphi(t)) \approx$ a weighted sum of $f(\varphi(j))$ for neighboring $\varphi(j) \approx \varphi(t)$ for all regressors

The Suggested Method: WDMR

WDMR: Weight Determination by Manifold Regression

(Manifold) Smoothness Assumption: $f(\varphi_1)$ and $f(\varphi_2)$ close if φ_1 and φ_2 are close (in a relevant metric).

Problem: associate $\varphi(i)$ with good values $\hat{f}_t = f(\varphi(t))$ for all regressors, both labeled and unlabeled.

Take care of two kinds of information:

- $f(\varphi(t)) \approx y(t)$ for the measured labels
- $f(\varphi(t)) \approx$ a weighted sum of $f(\varphi(j))$ for neighboring $\varphi(j) \approx \varphi(t)$ for all regressors

Formalize the second information using a *kernel* $K(\cdot,\cdot)$:

$$\hat{f}_t \approx \sum_{t=1}^{N_L+N_U} K_{t,j}\hat{f}_j; \qquad K_{t,j} = K(\varphi(t),\varphi(j))$$

Now, weigh together the two sources of information

$$\lambda \sum_{i=1}^{N_\mathrm{l}+N_\mathrm{u}} (\hat{f}_i - \sum_{j=1}^{N_L+N_U} K_{i,j}\hat{f}_j)^2 + (1-\lambda)\sum_{t=1}^{N_L}(y(t) - \hat{f}_t)^2 \qquad (*)$$

$$K_{i,j} = K(\varphi(i), \varphi(j))$$

Pick a ("regularization parameter") λ that balances the fit to measured labels and the smoothness prior. Minimize w.r.t. $\hat{f}_t, t = 1, \ldots, N_L + N_U$.

WDMR, cont'd

Now, weigh together the two sources of information

$$\lambda \sum_{i=1}^{N_\mathrm{l}+N_\mathrm{u}} (\hat{f}_i - \sum_{j=1}^{N_L+N_U} K_{i,j}\hat{f}_j)^2 + (1-\lambda)\sum_{t=1}^{N_L}(y(t)-\hat{f}_t)^2 \qquad (*)$$

$$K_{i,j} = K(\varphi(i), \varphi(j))$$

Pick a ("regularization parameter") λ that balances the fit to measured labels and the smoothness prior. Minimize w.r.t. $\hat{f}_t, t = 1, \ldots, N_L + N_U$.

That gives the estimated function value $\hat{f}(\varphi)$ for any regressor φ you include among the unlabeled regressors.

Now, weigh together the two sources of information

$$\lambda \sum_{i=1}^{N_\mathrm{l}+N_\mathrm{u}} (\hat{f}_i - \sum_{j=1}^{N_L+N_U} K_{i,j}\hat{f}_j)^2 + (1-\lambda)\sum_{t=1}^{N_L}(y(t)-\hat{f}_t)^2 \qquad (*)$$

$$K_{i,j} = K(\varphi(i), \varphi(j))$$

Pick a ("regularization parameter") λ that balances the fit to measured labels and the smoothness prior. Minimize w.r.t. $\hat{f}_t, t = 1, \ldots, N_L + N_U$.

That gives the estimated function value $\hat{f}(\varphi)$ for any regressor φ you include among the unlabeled regressors.

Note that $(*)$ is quadratic in $\hat{f}_t$, so the solution is easy to obtain.

Choice of Kernel $K(\cdot, \cdot)$

Many possibilities: K-Nearest Neighbors, Gaussian Kernels ....

Choice of Kernel $K(\cdot, \cdot)$

Many possibilities: K-Nearest Neighbors, Gaussian Kernels ....

Choice that can adapt the metric to manifolds and other high-density regions: Local Linear Embedding (LLE):

Choice of Kernel $K(\cdot,\cdot)$

Many possibilities: K-Nearest Neighbors, Gaussian Kernels ....

Choice that can adapt the metric to manifolds and other high-density regions: Local Linear Embedding (LLE):

Find the Kernel $K_{i,j} = K(\varphi(i),\varphi(j))$ by minimizing (wrt $K_{i,j}$)

$$\varepsilon(K) = \sum_{i=1}^{N} \left\| \varphi(i) - \sum_{j=1}^{N} K_{i,j}\varphi(j) \right\|^2 \tag{3a}$$

under the constraints

$$\begin{cases} \sum_{j=1}^{N} K_{i,j} = 1, \qquad K_{i,i} = 0 \\ K_{i,j} = 0 \text{ if } \|\varphi(i) - \varphi(j)\| > C_i(R) \text{ or if } i = j. \end{cases} \tag{3b}$$

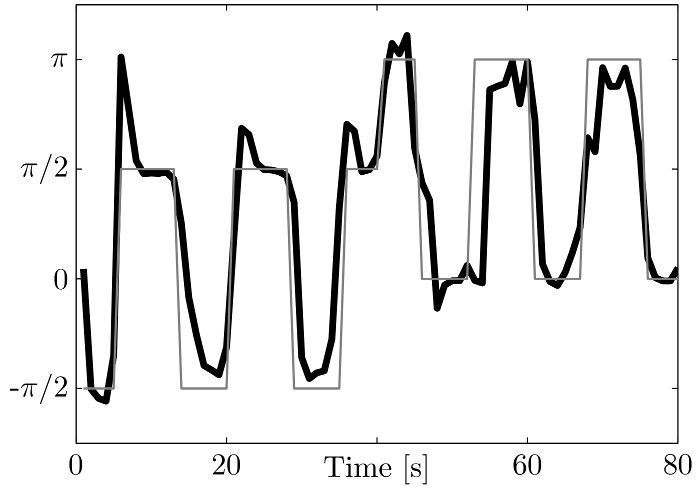
Here, $C_i(R)$ is chosen so that only R weights $K_{i,j}$ become nonzero for every i. R is a design variable.

Example 1: fMRI Signals

The patient in the magnet camera is moving his eye focus left - right - up - down. 128 voxels in the visual cortex are monitored by fMRI, giving a vector $\varphi(t) \in R^{128}$ sampled every two seconds. The output $y(t)$ is the viewing angle $0, \pi, \pi/2, -\pi/2$.

120 (labeled) samples were collected. 80 of those were treated as "labeled" ($N_L = 80$) and the remaining 40 were treated as "unlabeled" ($N_U = 40$).

Below we show the predicted y-values $([-\pi/2 \quad \pi])$, (thick line) for these unlabeled measurements together with the corresponding true angles (thin line).



Equation: (Biological interpretation ....)

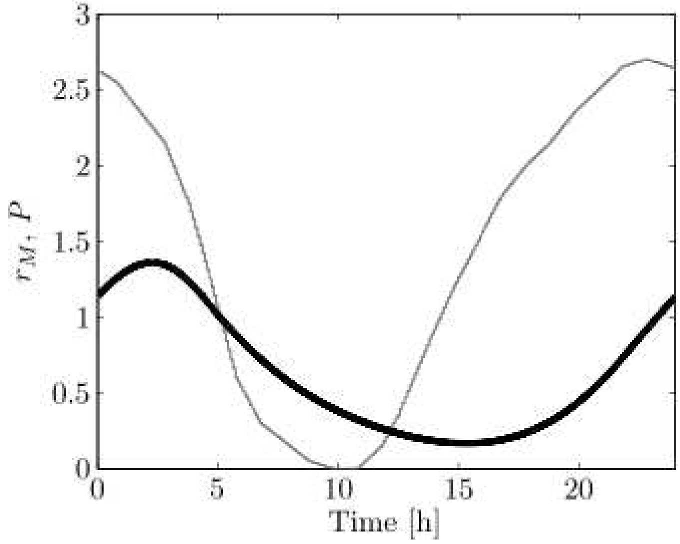
$$\frac{dM(t)}{dt} = \frac{r_M(t)}{1 + P(t)^2} - 0.21M(t),$$
$$\frac{dP(t)}{dt} = M(t-4)^3 - 0.21P(t),$$

Input $r_M(t)$ (periodic, cheap), Output $P(t)$ (expensive)

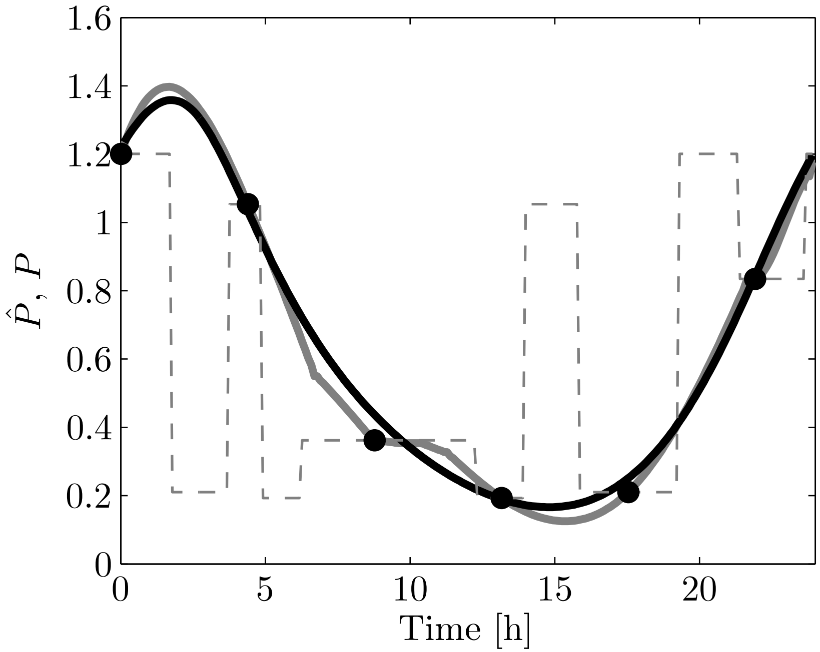
Predict P from r_M. Use $[r_M(t) \;\; r_M(t-4)]^T$ as regression vector

Circadian Clock, cont'd

Measured Data



Estimated output (black line)
true output (grey line)
measured outputs (6 dots)
[Dashed: Near.t Neighb. estimates]



Note that only 6 P-values are measured.
(t=0, 4, 9, 14,17, 22)

Example 3: The Narendra-Li Example

$$x_1(t+1) = \left(\frac{x_1(t)}{1+x_1^2(t)} + 1\right)\sin(x_2(t))$$

$$x_2(t+1) = x_2(t)\cos(x_2(t)) + x_1(t)\exp\left(-\frac{x_1^2(t)+x_2^2(t)}{8}\right) + \frac{u^3(t)}{1+u^2(t)+0.5\cos(x_1(t)+x_2(t))}$$

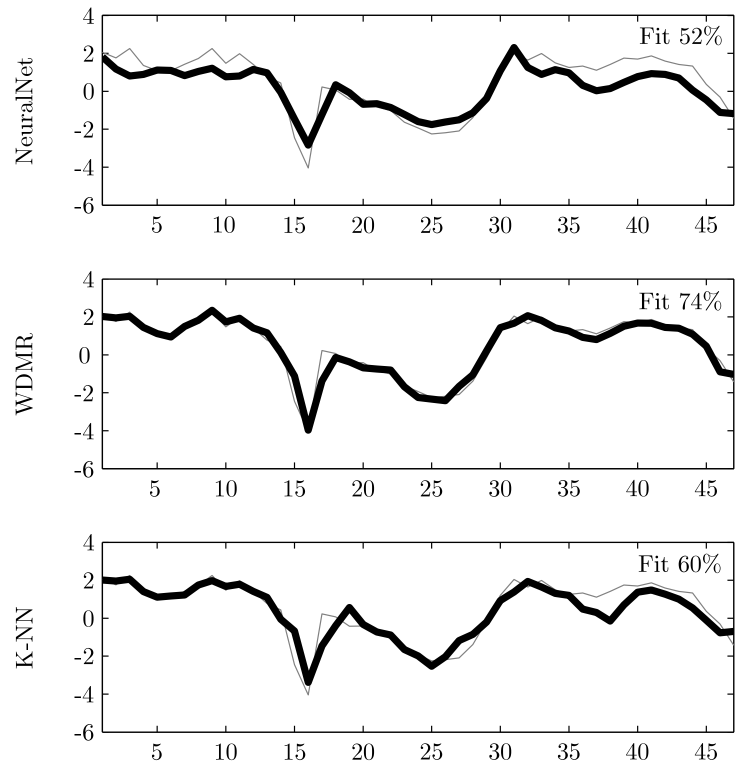
$$y(t) = \frac{x_1(t)}{1+0.5\sin(x_2(t))} + \frac{x_2(t)}{1+0.5\sin(x_1(t))} + e(t)$$

u: RBS for estimation data; sinusoid for validation data

Regression vector.

$\varphi(t) = [y(t-1)y(t-2), y(t-3), u(t-1), u(t-2), u(t-3)]^T$ [NLARX]

One-step ahead prediction on validation data



Top: Standard Neural Network (18 hidden units — best) 52% fit
Middle: WDMR ($\lambda = 0.5$, LLE kernel R=5) 74% fit
Bottom: K-nearest neighbor (K = 15 — best) 60% fit

Conclusions

- Ideas from “machine learning” have relevance for system identification
- A method WDMR was suggested for non-linear regression
  - Local, non-parametric flavor
  - Links to LLE
- It shows promising results for several examples of different characters
- More to understand regarding the potential for system identification!