Parametric regression estimation. Part II: the constrained least squares

Przemysław Śliwiński

***Abstract*—Here we end our least-squares journey…**

I. Introduction

A. CVX introduction…[1]

We have (again):

- A random input sequence, $\{x_n\} \in [-\pi, \pi]$ for all $n = \ldots, -1, 0, 1, \ldots$.
- A system

$$m(x) = \sum_{k=1}^{K} \varphi_k(x) \cdot \alpha_k, \tag{1}$$

  with some K, where $\{\alpha_k\}$ are unknown, and where, for instance, $\varphi_k(x) = \cos(kx)$.
- A random additive output noise, $\{z_n\} \sim N\left(0, \sigma_z^2\right)$.

B. Exercises

Exercise 1: Taking

$$z_n \sim N(0,1), x_n \sim U[-\pi, \pi],$$

and, *cf.* (1)

$$m(x) = -1 \cdot \cos(x) + 2 \cdot \cos(3x) - 1 \cdot \cos(5x),$$

generate $N = 256$ input-output pairs.

$$\{(x_n, y_n = m(x_n) + z_n)\}.$$

Exercise 2: Employ the **Matlab CVX**[2] library to find the vector, $\hat{A}_{\bar{K}} = \left[\begin{array}{ccc} \hat{\alpha}_1 & \cdots & \hat{\alpha}_K \end{array}\right]$, of the *empirical coefficients* (aka *least-squares estimates*, *parameters*) of the model, *cf.* (1)

$$\hat{m}(x) = \sum_{k=1}^{K} \cos(kx) \cdot \hat{\alpha}_k, \text{ for } \bar{K} = 64, 128, 256, 512$$

of the actual system $m(x)$ using the generated measurement set $\{(x_n, y_n)\}$ under the extra assumption that

$$\|\hat{\alpha}_k\|_p \leq R, \text{ some } p \in [1, \infty] \text{ and } R > 0. \tag{2}$$

Exercise 3: Find the p and R such that the empirical mean square error

$$\widehat{err} = \sum_{q=-Q}^{Q} \left[\hat{m}(x_q) - m(x_q)\right]^2, \text{ for } x_q = \frac{q\pi}{Q} \text{ and some } Q.$$

Exercise 4: Compare the results with the outcome of the Gasser-Müller and least-squares algorithm without constraints for the same settings.

[1]Aka *Machine Learning* (ML) bread and butter/piece of cake/nuts and bolts…

[2]`http://cvxr.com/cvx/`

II. Tips'n'tricks

A. CVX II

```
% Measurements number N and model size K
N = ...; K = .. .;
Q = ...;

% Constraint parameters
p = ...; R = ...;

% Matrix of regressors FI
% Vector of parameters A
FI = ...;
Y = ...;

cvx_begin
    variable A(K)
    minimize(norm(FI * A - Y, 2))
    subject to
       norm(A, p) <= R
cvx_end
```

B. Intuitive illustration

Compare shapes and areas of the norms in various L_p as in figure below. Relate them to the constrain in (2) and explain the impact on the LS solution.

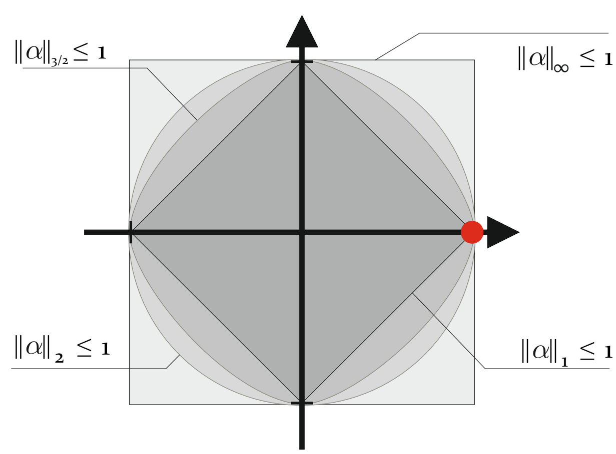

Fig. 1. Shapes of various unit balls for various L_p norms, $p = 1, 1.5, 2, \infty$