



# Functional Data Analysis of Borrmann Curves

Ligia Andrea Martín Montoya

Universidad Nacional de Colombia

Supervisor: Martin Tolkiehn

September 6, 2011

## Abstract

The present study involves the Functional Data Analysis to visualize those principal features that affect the Borrmann curves using MATLAB. In order to get rid of the noise, the smoothing using Roughness Penalty and the shift of the curves were done. Using PCA has been found the first principal components that ensure how the height and the width are the most representative landmarks. Finally the Rotating Factors shows an other point of view and represents an important tool to interpret the principal components and to make a qualitative analysis about the curves.

# Contents

<b>1</b>	<b>Introduction</b>	<b>3</b>
<b>2</b>	<b>Borrmann spectroscopy</b>	<b>3</b>
<b>3</b>	<b>Functional Data Analysis</b>	<b>4</b>
3.1	Turning Row Discrete Data into Smooth Functions . . . . .	4
3.1.1	Smoothing Functional Data by Least Squares . . . . .	4
3.1.2	Smoothing data with Roughness Penalty . . . . .	5
3.1.3	Choosing the Smoothing Parameter $\lambda$ . . . . .	6
3.1.4	Alignment: Newton Raphson Algorithm . . . . .	8
3.2	Principal Components Analysis (PCA) . . . . .	10
3.2.1	Rotating Factors: VARIMAX . . . . .	12
<b>4</b>	<b>Conclutions</b>	<b>14</b>
<b>5</b>	<b>Matlab code</b>	<b>15</b>
5.1	Declaration . . . . .	15
5.2	Parameter $\lambda$ . . . . .	16
5.3	Smoothing: Roughness Penalty . . . . .	16
5.4	Shift . . . . .	17
5.5	PCA . . . . .	17

# 1 Introduction

The Functional Data Analysis (FDA) is a powerful tool to analyze the Borrmann curves (Intensity  $I$  vs. incident angle  $\theta$ ) when the energy is varied. These curves exhibit the tendency to have a peak of intensity with some width around the Bragg angle that in principle could be adjusted like Gaussians. Ideally the curves have the peak centered in the Bragg angle without dependency of the energy value. Experimentally it has been found that the curves exhibit shifts in the position of the peak and also a lot of noise; it has been also observed in the curves that there is an enhancement on the intensity peak and a variable width.

In order to analyze the data in a proper way, it is desirable to convert the discrete curves in a functional form and for this goal is very useful to express the data like a linear combination of basis functions and then proceed to make a smooth of them. The B-spline functions were chosen as a basis set because they are the best choice of approximation system for non-periodic functional data.

The FDA does not only show the way to find the number of basis functions or the correct smooth parameter, it is also a good method to shift the data. Once the smoothing and alignment of the curves is done, the PCA (Principal Component Analysis) is a big branch of the FDA which is used here to find and analyze the increment or decrement of the peak or the width of the curves and to visualize the results.

## 2 Borrmann spectroscopy

The Borrmann effect occurs in the transmission case of X-ray diffraction (see Figure 1). In this geometry is possible to study in a more deep way the electric quadrupole absorption that is not possible to see in the Bragg geometry.

This effect occurs under conditions of Laue diffraction when the crystal is many times thicker than the absorption length. A single standing wave field forms parallel to the crystal planes with nodes at the atomic planes and polarization parallel to the planes (see Figure 1).

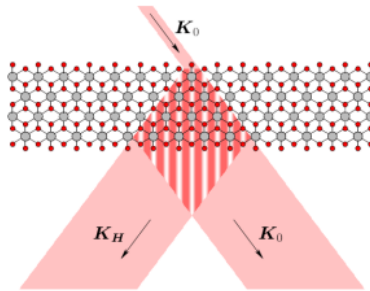


Figure 1: *Borrmann Effect*

### 3 Functional Data Analysis

#### 3.1 Turning Row Discrete Data into Smooth Functions

There are  $n$  incident angles for every  $E$  values of energy, so we have a  $(E \times n)$  matrix data  $\mathbf{R}$  and the first aim is to express the data in terms of a linear combination of basis functions. In fact the original data will be equal to the fit data  $\mathbf{Y}$  plus some error:

$$\mathbf{R} = \mathbf{Y} + \varepsilon \quad (1)$$

Where

$$\mathbf{Y} = \mathbf{c}'\phi \quad (2)$$

Here  $\phi$  is the set of B-spline functions (Figure 2) with a size of  $(n \times K)$  where  $K$  is the number of basis functions and  $\mathbf{c}$  is the  $K \times n$  coefficient matrix.

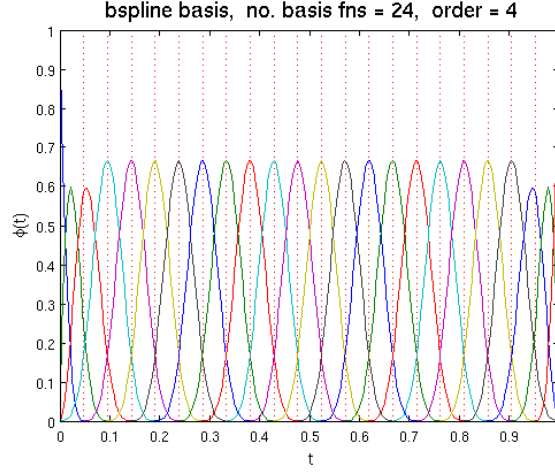


Figure 2: *B-spline*  $K = 24$

##### 3.1.1 Smoothing Functional Data by Least Squares

The principal idea is to use the Least Squared Criterion (LSC) to minimize the difference between our data  $\mathbf{R}$  and the ideal smooth curve  $\mathbf{Y}$ :

$$LSC = [\mathbf{R} - \mathbf{Y}]^2 = (\mathbf{R} - \mathbf{Y})'(\mathbf{R} - \mathbf{Y}) \quad (3)$$

replacing (2) in (3) it can be

$$LSC = (\mathbf{R} - \mathbf{c}'\phi)'(\mathbf{R} - \mathbf{c}'\phi) \quad (4)$$

Taking the first derivate of the Least Square Criterion equal to zero yields the equation:

$$2\phi\phi'\mathbf{c} - 2\phi'\mathbf{y} = \mathbf{0} \quad (5)$$

$\hat{\mathbf{c}}$  that minimize the least squares solution is:

$$\hat{\mathbf{c}} = (\phi'\phi)^{-1}\phi'\mathbf{R} \quad (6)$$

So,

$$\mathbf{Y} = \phi\hat{\mathbf{c}} = \phi(\phi'\phi)^{-1}\phi'\mathbf{R} \quad (7)$$

Choosing 24 B-spline funtions for  $n = 39$  and  $E = 261$  we find  $\mathbf{Y}$  like:

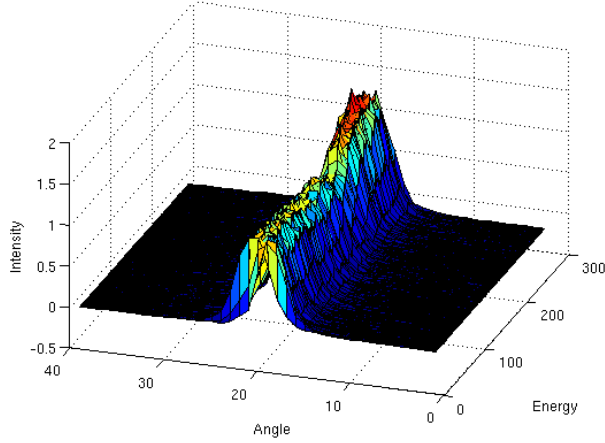


Figure 3: *Smooth LSC*

### 3.1.2 Smoothing data with Roughness Penalty

There is explicitly two conflict goals in the curve estimation. On one hand is desirable to ensure that the estimated curve gives a good fit to the data, while on the other hand one does not wish to make the fit too good, because  $\mathbf{Y}$  can be excessively wiggly. In order to find the criterion, an attempt would be to use the derivatives.

Let's define:

$$PEN_2 = \int D^2Y \quad (8)$$

Now defining a new criterion that include  $PEN_2$

$$PENSSSE_\lambda = (\mathbf{Y} - \mathbf{R})^2 + \lambda PEN_2 \quad (9)$$

Where  $\lambda$  is the smooth parameter.

From (2):

$$PEN_2 = \mathbf{C}'(\int \mathbf{D}^2\phi\mathbf{D}^2\phi')\mathbf{C} \equiv \mathbf{C}'\mathbf{R}_{\text{mat}}\mathbf{C} \quad (10)$$

Replacing (10) in (9) and taking the first derivative equal to zero, the new  $\hat{\mathbf{c}}$  obtained is:

$$\hat{\mathbf{c}} = (\phi' \phi + \lambda \mathbf{R}_{\text{mat}})^{-1} \phi' \mathbf{y} \quad (11)$$

Note that if  $\lambda$  is zero,  $\hat{\mathbf{c}}$  becomes the same as the one found by LSC, and by increasing  $\lambda$  the effect of the smoothing is more visible:

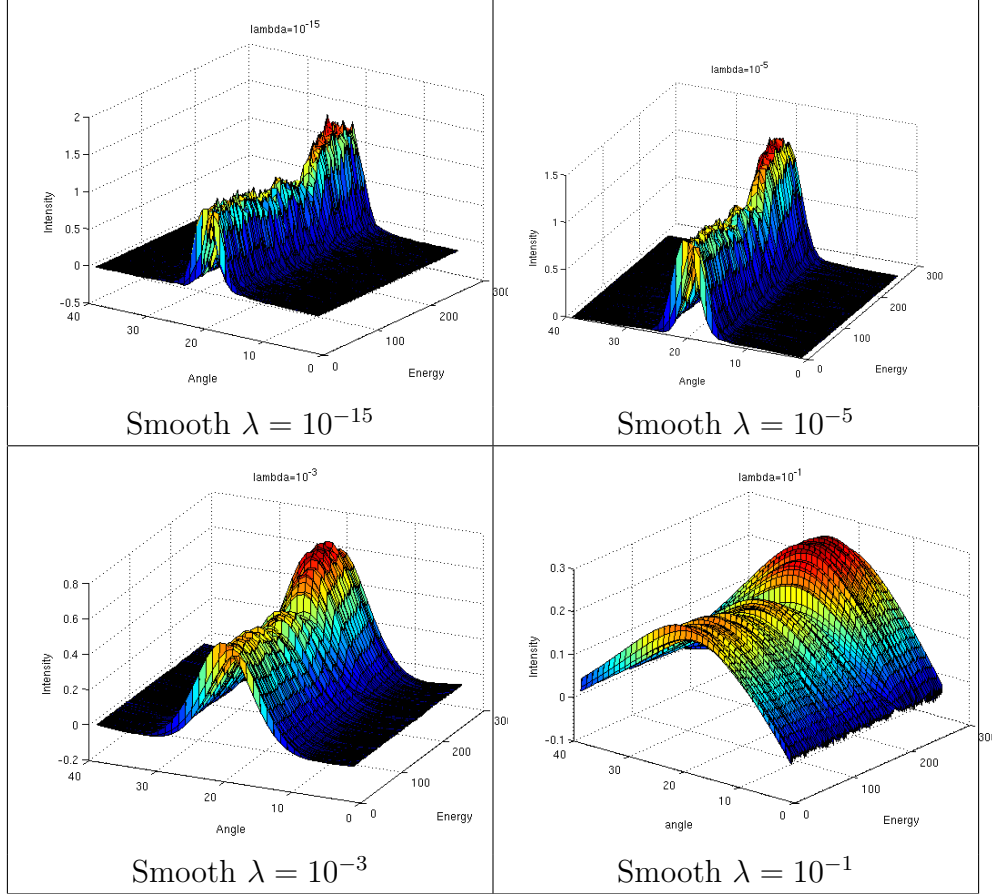


Figure 4: *parameter  $\lambda$*

### 3.1.3 Choosing the Smoothing Parameter $\lambda$

In a next step a good criterion of the parameter  $\lambda$  has to be found. The Generalized Cross-Validation GCV method is defined usually by:

$$GCV(\lambda) = \left( \frac{n}{n - df(\lambda)} \right) \left( \frac{SSE}{n - df(\lambda)} \right) \quad (12)$$

where:

$$SSE = \sum_{n=1}^n [\mathbf{R}_n - \mathbf{Y}_n]^2 \quad (13)$$

and;

$$df(\lambda) = \text{trace}[\phi(\phi'\phi + \lambda \mathbf{R}_{\text{mat}})^{-1}\phi'] \equiv \text{trace}[\mathbf{S}] \quad (14)$$

The aim now is to find the new parameter  $l$  related with  $\lambda$  by  $\lambda = 10^{-l}$  which minimizes the GCV. This parameter is actually different for every fixed energy data. In the figure (4) are shown some examples of the GCV plot for four different energy values. The minimum value is indicated in color blue:

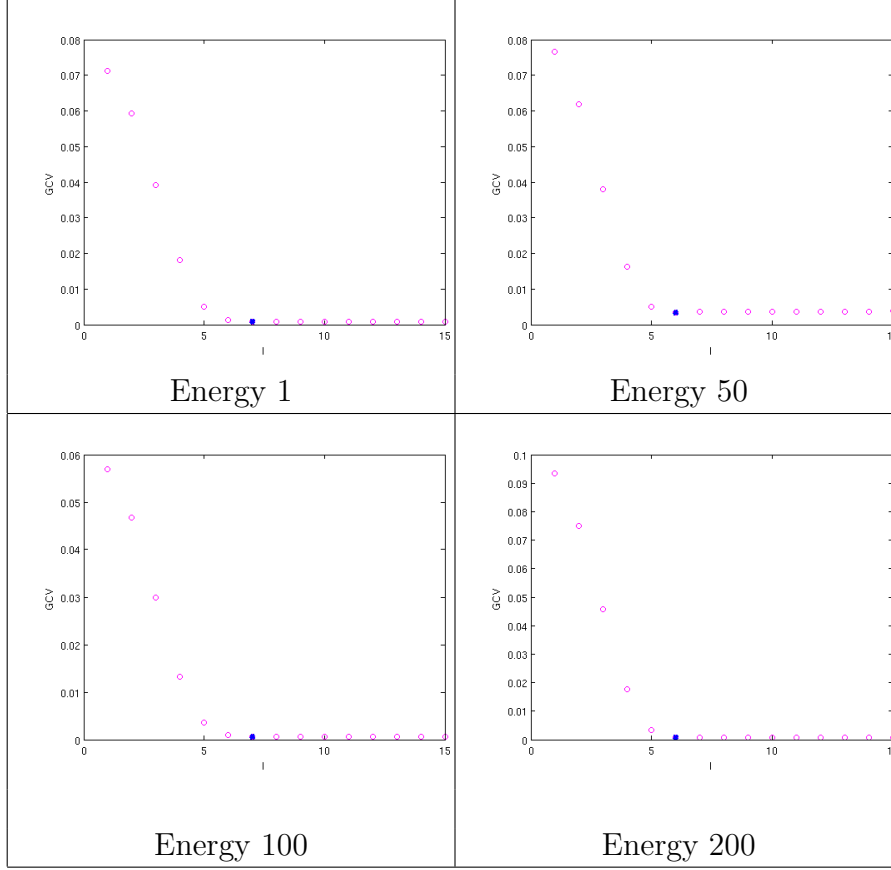


Figure 5: *GCV curves*

In the GCV plots it can be clearly seen that there is not a relevant difference for big  $l$  values ( $\geq 6$ ), this is the reason why in this case the smoothing with  $\lambda = 10^{-15}$  is very similar to the one done with  $\lambda = 10^{-5}$  (see Fig.4). Taking the mean of all the  $l$  values the best smooth parameter is  $\lambda = 10^{-6.24}$  is obtained.

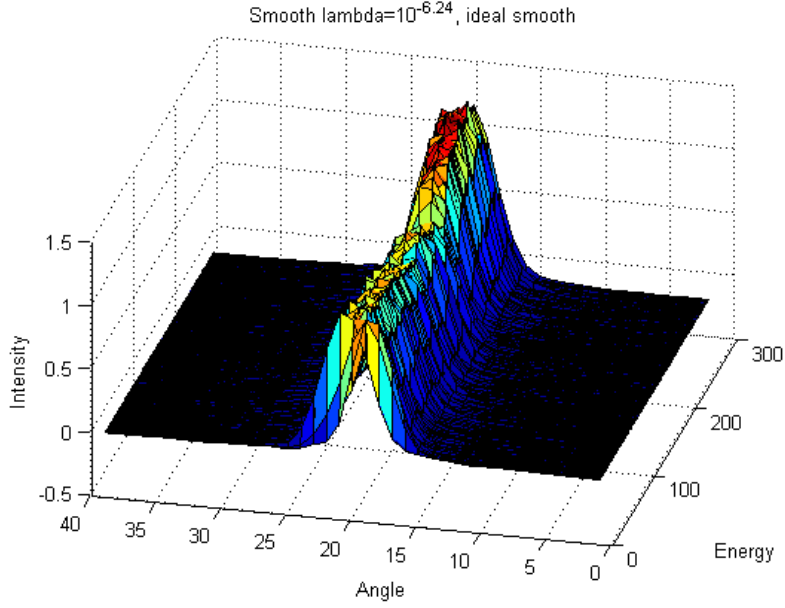


Figure 6: *Smooth curves with  $\lambda = 10^{-6.24}$*

#### 3.1.4 Alignment: Newton Raphson Algorithm

In order to align the intensity peak of each curve the *Newton Raphson* algorithm has been used.

To shift the smooth data one should take into account that this shift is not the same for each curve. One way is to use the *Landmark Registration*, which is a method to identify one special feature using, in the most of the cases, derivatives and then proceed to shift all the curves. In the present study this feature can be seen as the negative slope in the first derivative, nevertheless it is more convenient to use an algorithm that works generally and not only with one special feature or landmark. The *Newton Raphson algorithm* is very useful for this purpose because allows to identify this shift without taking into account the type of data we are working with.

The *Newton Raphson algorithm* is an iterative method and works as follows:

- **Step 0**

Beginning with some initial shift estimates  $\delta_i^{(0)}$ , perhaps by aligning with respect to some feature, or even  $\delta_i^{(0)} = 0$ . But the better the initial estimation, the faster and more reliably the algorithm would converges.

- **Step 1**

Estimate the average  $\hat{\mu}$  of the smooth curves.



where

$$\hat{\mu} = \frac{1}{E} \sum_{i=1}^E \mathbf{Y}_i \quad (15)$$

• **Step  $v$**

for  $v=1,2,\dots$  Modify the estimation  $\delta_i^{(v-1)}$  on the previous iteration by:

$$\delta_i^{(v)} = \delta_i^{(v-1)} - \alpha \frac{(\frac{\partial}{\partial \delta_i}) REGSSE}{(\frac{\partial^2}{\partial \delta_i^2}) REGSSE} \quad (16)$$

where  $REGSSE$  (the global registration criterion) is the global sum of squared vertical discrepancies between the shifted curves and the sample mean curve. The  $REGSSE$  is defined as:

$$REGSSE = \sum_{i=1}^N \int [\mathbf{Y}_i(\mathbf{t} + \delta_i) - \hat{\mu}(\mathbf{t})]^2 d\mathbf{t} \quad (17)$$

It follows that the first and the second derivative of the global criterion are:

$$\frac{\partial}{\partial \delta_i} REGSSE = 2 \int [\mathbf{Y}_i(\mathbf{t} + \delta_i) - \hat{\mu}(\mathbf{t})] \mathbf{D} \mathbf{Y}_i(\mathbf{t}) d\mathbf{t} \quad (18)$$

$$\frac{\partial^2}{\partial \delta_i^2} REGSSE = 2 \int [\mathbf{Y}_i(\mathbf{t} + \delta_i) - \hat{\mu}(\mathbf{t})] \mathbf{D}^2 \mathbf{Y}_i(\mathbf{t}) d\mathbf{t} + 2 \int [D \mathbf{Y}_i(\mathbf{t})]^2 d\mathbf{t} \quad (19)$$

and  $\alpha$  is a step-size parameter that can sometimes simply be set to one. It is usual to drop the first term in (18) since it vanishes at the minimizing values, and the convergence without this term tends to be more reliable when current estimates are substantially far from the minimizing values.

Taking  $\alpha = 1$ ,  $\delta_i^{(0)} = 0$  and dropping the first term of (19), one can compare the not aligned data with the aligned one:

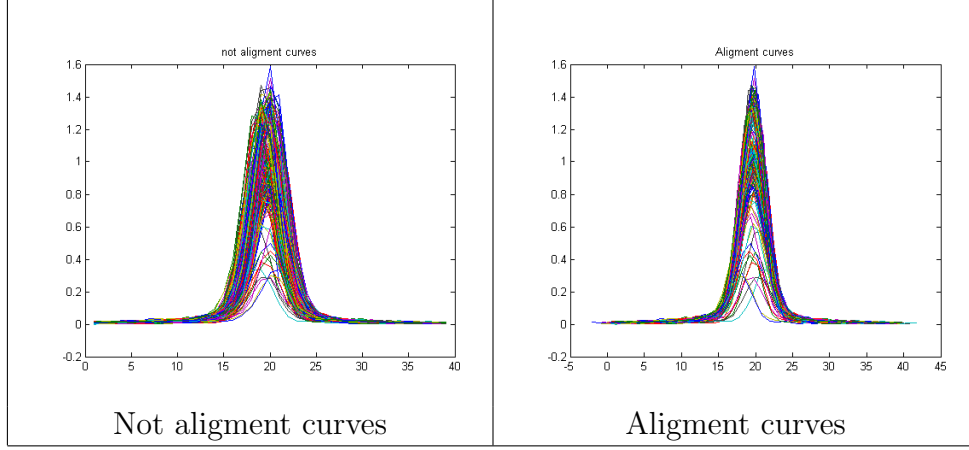


Figure 7: *Not alignment and alignment curves respectively*

and finally the 3D plot of the smooth and align data is obtained (Figure 8).

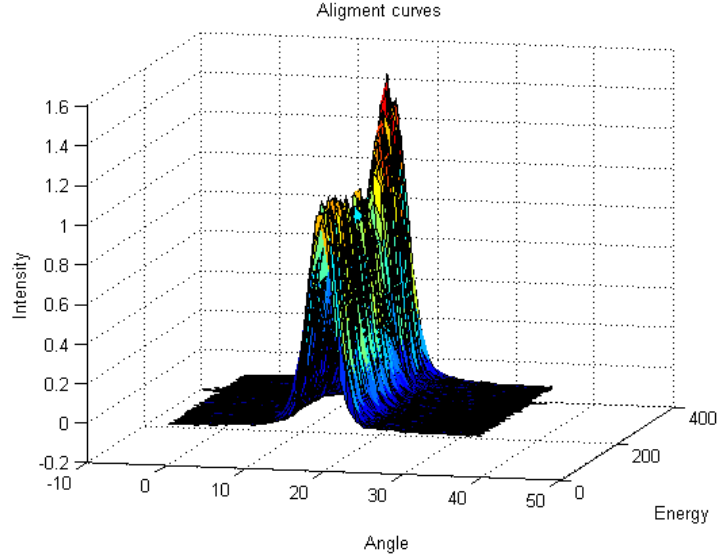


Figure 8: *3D plot of alignment curves*

### 3.2 Principal Components Analysis (PCA)

The aim of this study is to investigate the characterization of the PCA in terms of the eigen-analysis of the variance-covariance function or operator. Subtracting the mean  $\hat{\mu}(t)$  from the smoothed and aligned data  $\mathbf{Y}$ . The mean square criterion for finding the first principal component weight vector can be written as:

$$\max \xi' \mathbf{V} \xi \quad (20)$$

Or the equivalent eigenvector problem:

$$\mathbf{V} \xi = \rho \xi \quad (21)$$

where  $\mathbf{V}$  is the variance-covariance matrix, defined as

$$\mathbf{V} = \mathbf{N}^{-1} \mathbf{Y}' \mathbf{Y} \quad (22)$$

where  $N$  is the number of curves (in our case  $E$ ) and  $\xi$  are the principal components which we are searching for.

Solving the eigenvector problem the principal components could be obtained.

One way to visualize it is plotting components as perturbations of the mean.

$$\hat{\mu} \pm 0.2C\xi_i \quad (23)$$

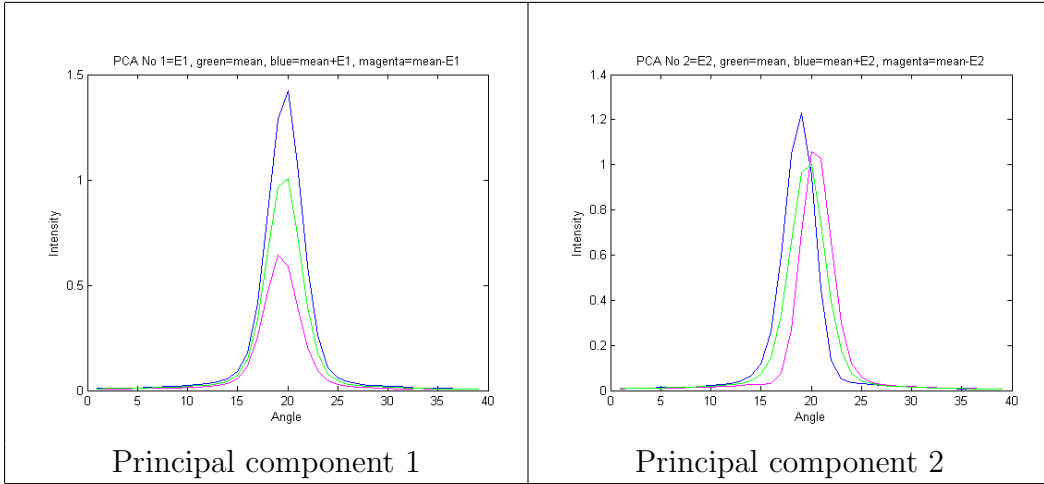
where

$$C^2 = n^{-1} \|\hat{\mu} - \bar{\mu}\|^2 \quad (24)$$

and

$$\bar{\mu} = n^{-1} \int \hat{\mu}(t) dt \quad (25)$$

The plots of the effect of adding and subtracting to the mean the three principal components are shown in Figure 9.



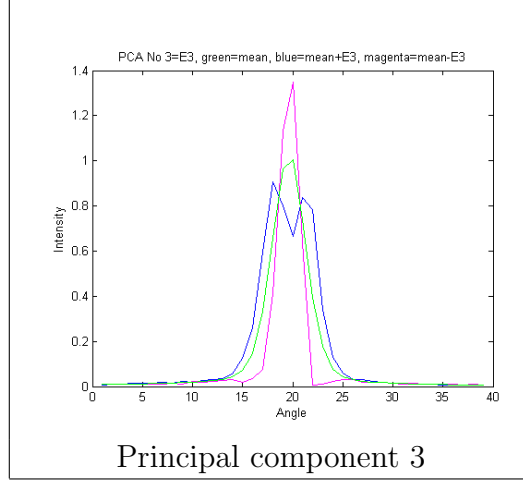


Figure 9: *Principal components*

Here we can note that the first principal component shows the effect of the differences in height between curves, and taking in account that is the first component we can conclude that is the most representative effect in the curves. This change in the height of the peaks is because of the absorption level of the atoms in the crystal.

The second principal component, which represents the second one most significative gives informations about the possible shift in the curves and the third components represents the difference in the width of the curves.

### 3.2.1 Rotating Factors: VARIMAX

The principal components  $\xi$  can be viewed as defining orthonormal set of  $K$  functions for expanding the curves to minimize a summed integrated squared error criterion:

$$\sum_{i=1}^E \|\mathbf{Y}_i - \hat{\mathbf{Y}}_i\|^2 \quad (26)$$

where

$$\hat{\mathbf{Y}}_i = \sum_{k=1}^K \mathbf{f}_{ik} \xi_k \quad (27)$$

and

$$\mathbf{f}_{ik} = \mathbf{Y}_i \xi_k \quad (28)$$

This does not mean, however, that there aren't other orthonormal sets that will do just as well, for example:

$$\psi = \mathbf{T} \xi \quad (29)$$

where  $\mathbf{T}$  is the rotation matrix.

From a geometrical perspective, the vector of functions  $\psi$  is a rigid rotation of  $\xi$ . In order to find this new set of rotated components we attempt to use VARIMAX method that maximize the diagonal elements of  $\psi$ . Using VARIMAX in Matlab has been found the three first rotate factors, that are showing as perturbations of the mean (see Figure 9).

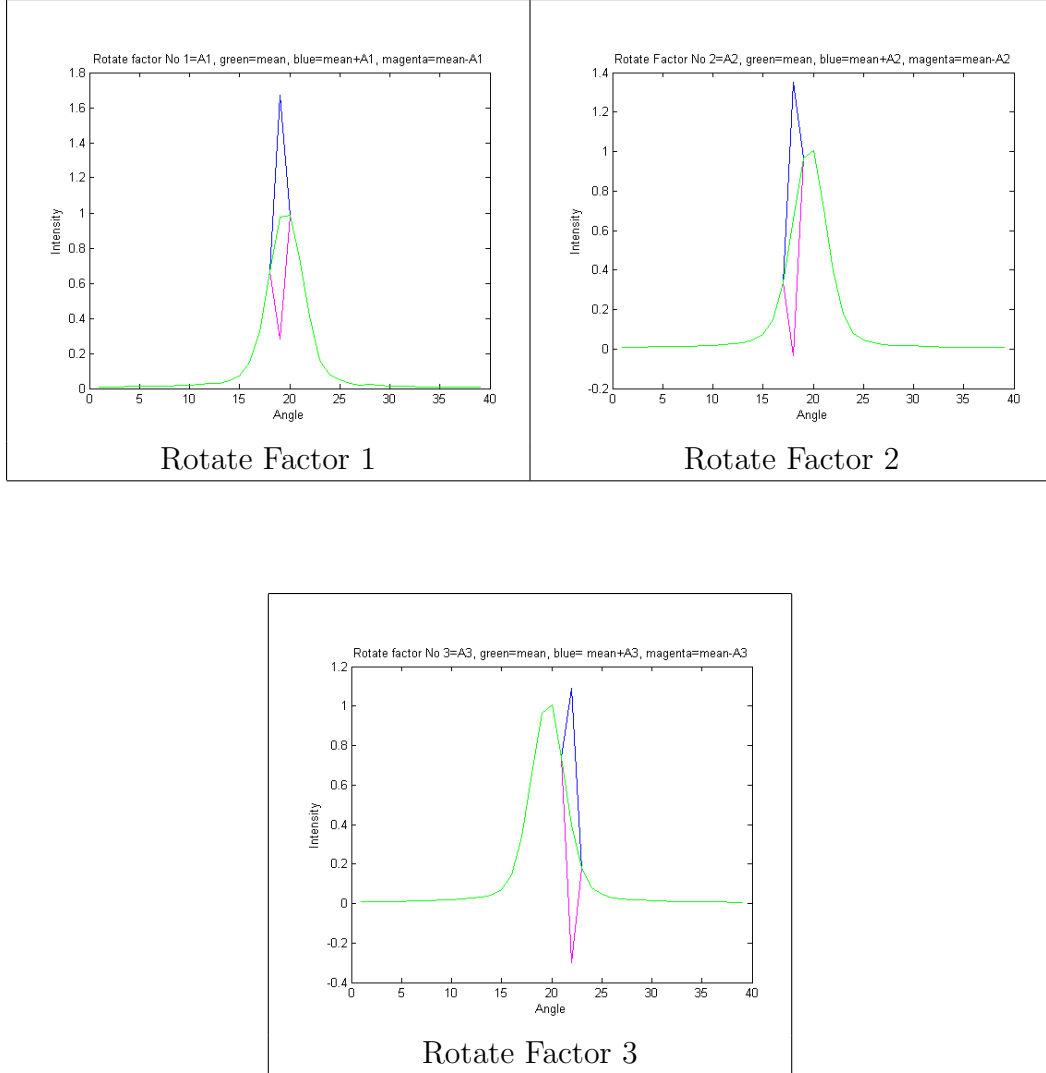


Figure 10: *Rotating Factors*

The principal aim to find them is that sometimes the rotating factors are easier to interpret. For example, it can be clearly seen that the first rotating factors indicate change in the height and width.

Principal Component analysis is a great tool to identify different characteristics of one type of data usually hidden by the noise; As an example, in the present study it has been found the effect of the height, the width and the shift separately and it was also

possible to find that the most representative effect is the changes in height because it was found in the first Principal Component.

## 4 Conclutions

After the Smothing and the Shifting of the data have been done, it is possible to find the correct curve ( intensity  $I$  vs. energy  $E$ ) taking the maximum values of each primary curves (intensity  $I$  vs. angle  $\theta$ ) (see Figure 11).

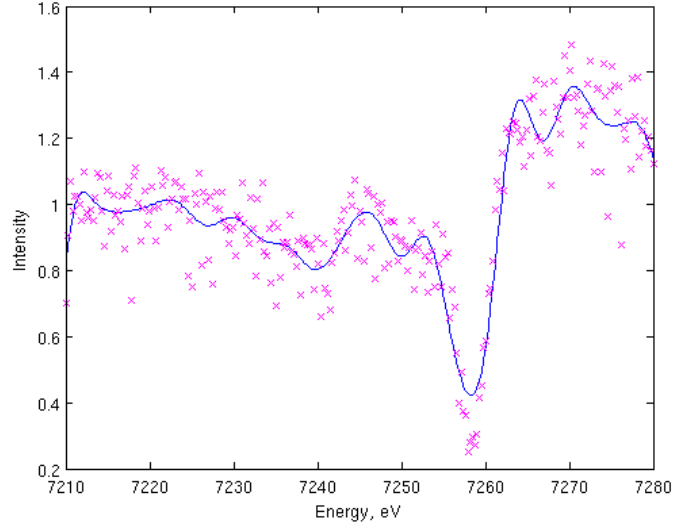


Figure 11: *Intensity vs. energy. In blue the smooth and shifted data; in magenta the original data.*

Using the PCA and the rotating factors it was possible to find that the most important features in the smooth and shift data are: the variable height and width of the primary curves (intensity  $I$  vs. angle  $\theta$ ).

## 5 Matlab code

### 5.1 Declaration

```
##### FDA code #####

K=41; % number of basis funtions
o=4; % function's order
n=39;%Number of meassure angles
NE=261; % number of curves at different energies values

addpath /afs/desy.de/user/m/martil/fda/fdaM % add path with the bspline functions

R=dlmread('datos.dat');%data
```

## 5.2 Parameter $\lambda$

```
##### Parameter#####

base=create_bspline_basis([0,1],K,o);% bspline basis K basis functions order o

phi=eval_basis(0:1/(n-1):1,base); %evaluate the basis

harmaccelLfd=vec2Lfd([0,1],[0,1]);%2 order derivate operator

Rmat=eval_penalty(base,harmaccelLfd);

%Choosing the smoothing parameter
for l=1:15
    lambda=10^(-l);

    S=phi*inv(phi'*phi+lambda.*Rmat)*phi';
    Y1=S*R';
    Dif=(R'-Y1).^2;
    df= trace(S);
    a=(n/(n-df).^2);
    G=cumsum(Dif);
    for i=1:NE
        GCV(i,l)=a.*G(n,i); %squared errors, residuals
    end
end

    minimo=min(GCV',[],1);
for i=1:NE
    for j=1:15
        if GCV(i,j)==minimo(i)
            domain(i)=j;
        end
    end
end

l=mean(domain);
```

## 5.3 Smoothing: Roughness Penalty

```
##### SMOOTH #####

lambda=10^(-l);

Rmat=eval_penalty(base,harmaccelLfd);

S=phi*inv(phi'*phi+lambda.*Rmat)*phi';

Y1=S*R';

surf(Y1)%plot 3D
```



## 5.4 Shift

```
##### SHIFT #####
MEAN=sum(Y1,2) ./NE;%mean
DERIV=gradient(Y1');%derivates of the curves

%difference between the curves and the mean
for E=1:NE
    for i =1:n
        dif(i,E)=Y1(i,E)-MEAN(i);
    end
end
int1=2*trapz(dif.*DERIV');%first derivative of REGSSE
int2=2*trapz(DERIV'.^2);%second derivative of REGSSE

deltai=-int1./int2;%shift
xi=1:1:n;
x= repmat(xi',1,NE);
corrimento=repmat(deltai,n,1);
Xshift=x-corrimento;

surf(Xshift, repmat(1:NE, [n,1]), R)%alignments curves
```

## 5.5 PCA

```
%visualise results
timeAV=trapz(MEAN1');
c=sqrt(norm(MEAN'-timeAV)/n);
E=princomp(Y1');%principal components
A=rotatefactors(E);%VARIMAX

%ploting principal components as perturbations of the mean
p=4;
u=2;
plot(MEAN+p*c*E(:,u))
hold on
plot(MEAN-p*c*E(:,u), 'm')
hold on
plot(MEAN, 'g')
hold off
%rotating factors as perturbations of the mean
plot(MEAN+p*c*A(:,u))
hold on
plot(MEAN-p*c*A(:,u), 'm')
hold on
plot(MEAN, 'g')
hold off
```

## References

- [1] Functional Data Analysis, Springer Series in Statistics, Second Edition *J.O. Ramsay, B.W. Silverman*
- [2] Functional Data Analysis with R and MATLAB, Springer *J.O. Ramsay, Giles Hooker, Spencer Graves*
- [3] Borrmann spectroscopy *S P Collins, M Tolkiehn, R F Pettifer and D Laundry*