



HAL
open science

LE BRUIT ET LE MODELE Stratégies d'identification d'éléments discrets

Pascal Terracol

► **To cite this version:**

Pascal Terracol. LE BRUIT ET LE MODELE Stratégies d'identification d'éléments discrets. 2020.
hal-02938504

HAL Id: hal-02938504

<https://hal.science/hal-02938504>

Preprint submitted on 14 Sep 2020

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.

LE BRUIT ET LE MODELE

Stratégies d'identification d'éléments discrets

Pascal Terracol

Laboratoire EVCAU - ENSA Paris Val de Seine

Université Paris Diderot USPC

pascal.terracol@paris-valdeseine.archi.fr

Résumé. *Les nuages de points générés par les techniques de photogrammétrie et leur utilisation en numérisation d'édifices posent la question - notamment en matière de relevé archéologique - de l'identification d'objets géométriques à partir d'ensembles de données discrètes et volumineuses. Cet article propose quelques solutions et éléments prospectifs afin de reconstruire le continu à partir du discret.*

Mots-clés : *Photogrammétrie ; Nuage de points ; Modèles analytiques.*

Abstract. *Cloud points provided by photogrammetry schemes ask about identification on discrete items. This paper make a survey on several tools and prospective sights to avoid the gap between discrete and continuity.*

Keywords: *Photogrammetry ; Cloud of points ; Analytics models.*

Sommaire

Introduction	2
Un outil analytique : la méthode des moindres carrés	3
Régression linéaire	3
Régression planaire	5
Voutes et régressions circulaires	6
Dômes et régressions sphériques	7
Le potentiel d'un traitement en intelligence artificielle	8
Qualification sémantique	9
Bibliographie	9
Notice Biobibliographique	9

Introduction

A l'issue d'une rencontre par *blogs*¹ interposés, les archéologues d'Aix en Provence m'ont invité à expliciter un ensemble de concepts théoriques sur la géométrie. La problématique à laquelle ils sont confrontés réside dans les masses de données issues des relevés photogrammétriques qu'ils réalisent sur le terrain. Ces objets qui deviennent des artefacts du réel sont allègrement composés de plusieurs millions de points. Ces données sont très agréables à consulter. Elles complètent bien efficacement les séries de clichés permettant d'explorer un site en 3D tout en évitant d'y revenir afin de combler les lacunes souvent inhérentes à l'exercice du relevé, mais le volume de ces données pose la question de la référence géométrique. En particulier, celui du choix arbitraire d'un élément signifiant, comme un rayon de courbure, de la signature d'une volute, de la qualification géométrique d'un bloc afin d'en identifier la cohérence d'appareillage ou de l'identification de sa présence dans un appareil hétéroclite construit à l'issue d'un réemploi.

Ainsi le bruit convoque-t-il de manière sous-jacente un modèle canonique qui est l'expression du geste architectonique de son auteur, qu'il en soit l'architecte ou le sculpteur. Au-delà des tentatives d'analyses conduites sous de multiples faisceaux épistémologiques convoquant les ontologies formulées par les anthropologues ; les historiens ; l'interprétation des épigraphes ; celles des archéologues et bien sûr des architectes, le dépassement du matériau brut tel qu'il nous parvient sous la forme de nuages de points dans un format numérique, codant à la fois leurs coordonnées, la signature chromatique et également les normales, permet grâce à la géométrie une approche complémentaire et précieuse. Il s'agit donc de décoder à partir d'une représentation discrète et bruitée, un modèle continu inscrit dans la typologie de la géométrie analytique et de l'associer avec le corpus des autres disciplines.

Pour cela l'identification du vocabulaire de l'architecture articulé au modèle morphologique et à la description de ses composantes : spécifiquement des directrices et des génératrices définissant des surfaces et permettant de qualifier géométriquement ces éléments, nécessite de faire des choix parmi l'éventail bruité des données. Quels points choisir pour identifier une génératrice ? Quelle mesure pour un rayon ? Comment qualifier une courbure ? Les candidats sont nombreux - discrets - là où l'on souhaiterait identifier une continuité, les arcs de cercles fragmentés et souvent lacunaires le disputent à l'arc théorique, la volute ébréchée compose avec son équivalent issu de la géométrie pure élaborée - elle - sur des fonctions polaires², la droite cherche son chemin selon une ligne polygonale hasardeuse et riche de plusieurs milliers de candidats potentiels.

¹ Jean-Jacques Malmay, "Comment calculer les dimensions d'un bloc avec son nuage de points ?," sur *RAAN*, 26/06/2016, <https://raan.hypotheses.org/805>.

² [Terracol 2017] pp. 192 ; 203.

Le choix arbitraire hésite entre l'unilatéral, la conjecture et le premier venu, les tropismes se perdent sous la dictature de la multitude. A la richesse du modèle sur-représenté s'oppose un tracé régulateur que l'on souhaitera canonique, pur, unique et représentatif des blocs qui nous parviennent.

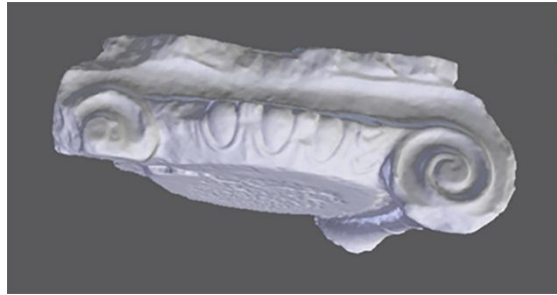


Figure 1 : Nuage de points composé de 2 467 465 points. Chapiteau Ionique Delos. Jean Jacques Malmay. Maison Méditerranéenne des Sciences de L'Homme, Institut de Recherche sur l'architecture antique (IRAA) – USR 3155 (AMU- CNRS) Aix en Provence. Représenté ici sous Blender v.2.79.

Un outil analytique : la méthode des moindres carrés

Régression linéaire

Largement utilisée en statistiques et probabilités, la méthode dite des moindres carrés que l'on doit à Gauss-Newton a pour objectif d'approcher au mieux par une fonction linéaire un nuage de points discrets. En utilisant les coordonnées X_i et Y_i et en calculant les minimums des dérivées partielles, le calcul analytique permet de construire la fonction $y = bx + a$ en minimisant l'erreur via une régression linéaire.

Ainsi la formulation des deux inconnues a et b s'écrivent alors sous la forme :

$$a = \frac{\left(\sum_{i=1}^n y_i \right) \left(\sum_{i=1}^n x_i^2 \right) - \left(\sum_{i=1}^n x_i \right) \left(\sum_{i=1}^n x_i y_i \right)}{n \left(\sum_{i=1}^n x_i^2 \right) - \left(\sum_{i=1}^n x_i \right)^2}$$

$$b = \frac{n \left(\sum_{i=1}^n x_i y_i \right) - \left(\sum_{i=1}^n x_i \right) \left(\sum_{i=1}^n y_i \right)}{n \left(\sum_{i=1}^n x_i^2 \right) - \left(\sum_{i=1}^n x_i \right)^2}$$

CodeCogsEqn (0).png (les formules à la suite sont nommées
« CodeCogsEqn (i).png » ces légendes sont laissées provisoirement
pour la compréhension du texte et la mise en page et sont
à supprimer à l'issue de celle-ci)

et l'implémentation de ce modèle sous RhinoScript peut s'écrire ainsi :

Option Explicit 'Script version Monday, 4 January 2018 20:09:36

Call Main()

Sub Main()

Dim i, a, b

Dim x(100000)

Dim y(100000)

Dim arrPoints, arrPt

'Saisie graphique des points

arrPoints = Rhino.GetPointCoordinates

If IsArray(arrPoints) Then

i = 0

For Each arrPt In arrPoints

x(i) = arrPt(0)

y(i) = arrPt(1)

i = i + 1

Next

Call my_Sum(x, y, i-1)

End If

End Sub

Sub my_Sum(x, y, n)

Dim i, dblaa, dblbb, dbldnum, dblsumx, dblsumy, dblsumx2, dblsumxy

Dim xmin, xmax, ymin, ymax, arrStartPoint, arrEndPoint, arrLine

dblaa = 0.0

dblbb = 0.0

dblsumx = 0.0

dblsumy = 0.0

dblsumxy = 0.0

dblsumx2 = 0.0

xmin = 100000000.0

ymin = 100000000.0

xmax = -100000000.0

ymax = -100000000.0

For i = 0 To n

If x(i) > xmax Then xmax = x(i) End If

If y(i) > ymax Then ymax = y(i) End If

If x(i) < xmin Then xmin = x(i) End If

If y(i) < ymin Then ymin = y(i) End If

Next

For i = 0 To n

dblsumx = dblsumx + x(i)

dblsumy = dblsumy + y(i)

*dblsumxy = dblsumxy + (x(i) * y(i))*

*dblsumx2 = dblsumx2 + (x(i) * x(i))*

Next

*dbldnum = (n + 1) * (dblsumx2) - (dblsumx * dblsumx)*

*dblaa = ((dblsumy * dblsumx2) - (dblsumx * dblsumxy)) / (dbldnum)*

*dblbb = (((n + 1) * dblsumxy) - (dblsumx * dblsumy)) / dbldnum*

'tracons la droite d'après le boxing

*arrStartPoint = Array(xmin, (dblbb * xmin) + dblaa, 0.0)*

*arrEndPoint = Array(xmax, (dblbb * xmax) + dblaa, 0.0)*

Call Rhino.AddLine(arrStartPoint, arrEndPoint)

End Sub

LE BRUIT ET LE MODELE

Pascal Terracol p. 4

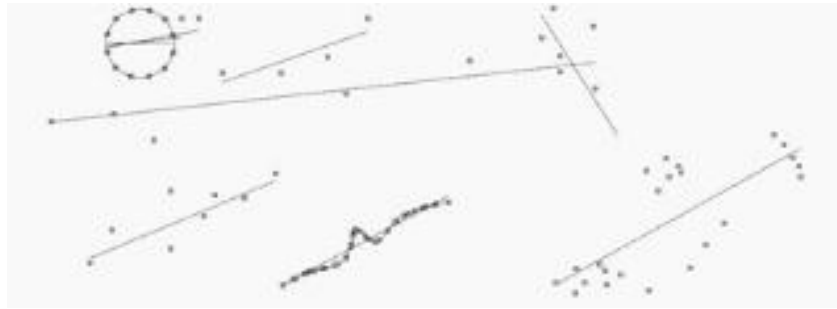


Figure 2. Quelques exemples d'ajustement linéaires bidimensionnels

Régression planaire

La généralisation de ce modèle bidimensionnel à un modèle tridimensionnel peut s'envisager en utilisant les coordonnées discrètes X_i, Y_i, Z_i afin d'approcher ce nuage de points par le modèle linéaire du plan correspondant :

$$z = F(x, y) = Ax + By + C \quad \text{CodeCogsEqn (1) .png}$$

Pour cela nous devons minorer la somme :

$$\varepsilon_{xy} = \sum |F(x_i, y_i) - z_i|^2 \quad \text{CodeCogsEqn (2) .png}$$

qui s'écrit donc :

$$\varepsilon_{xy} = \sum |Ax_i + By_i + C - z_i|^2 \quad \text{CodeCogsEqn (3) .png}$$

Ecrivons que les 3 dérivées partielles sont nulles :

$$\frac{\partial \varepsilon_{xy}}{\partial A} = 0 \quad \frac{\partial \varepsilon_{xy}}{\partial B} = 0 \quad \frac{\partial \varepsilon_{xy}}{\partial C} = 0 \quad \text{CodeCogsEqn (4) .png}$$

$$\frac{\partial \varepsilon_{xy}}{\partial A} = 0 \Leftrightarrow \sum 2(Ax_i + By_i + C - z_i) x_i = 0 \quad \text{CodeCogsEqn (5) .png}$$

$$\frac{\partial \varepsilon_{xy}}{\partial B} = 0 \Leftrightarrow \sum 2(Ax_i + By_i + C - z_i) y_i = 0 \quad \text{CodeCogsEqn (6) .png}$$

$$\frac{\partial \varepsilon_{xy}}{\partial C} = 0 \Leftrightarrow \sum 2(Ax_i + By_i + C - z_i) 1 = 0 \quad \text{CodeCogsEqn (7) .png}$$

ce qui nous conduit à écrire le système d'équations linéaires suivant :

$$\begin{pmatrix} \sum x_i^2 & \sum x_i y_i & \sum x_i \\ \sum x_i y_i & \sum y_i^2 & \sum y_i \\ \sum x_i & \sum y_i & 1 \end{pmatrix} \begin{pmatrix} A \\ B \\ C \end{pmatrix} = \begin{pmatrix} \sum x_i z_i \\ \sum y_i z_i \\ \sum z_i \end{pmatrix} \quad \text{CodeCogsEqn (8) .png}$$

dont la résolution et le calcul des 3 inconnues A, B, C permet d'identifier le plan de régression de notre distribution de points X_i, Y_i, Z_i .

Voutes et régressions circulaires

Les voutes peuvent être décrites par un modèle se rapprochant des régressions linaires précédentes à la condition d'opérer un changement de système de coordonnées³. Pour cela les coordonnées polaires ou cylindriques permettent de décrire un arc de cercle ou un segment de cylindre, circulaire à partir des coordonnées discrètes X_i, Y_i, Z_i exprimées dans un système de coordonnées cartésiennes.

Dans un système de coordonnées cylindriques où l'axe du cylindre est aligné sur l'axe des x un point M s'exprime selon :

$$M(\rho, \theta, x) \text{ avec } M \begin{cases} x = x \\ y = \rho \cos \theta \\ z = \rho \sin \theta \end{cases} \quad \text{CodeCogsEqn (9) .png} \quad \text{CodeCogsEqn (10) .png}$$

$$\text{et } \rho = \sqrt{y^2 + z^2} \quad \text{CodeCogsEqn (11) .png}$$

$$\cos \theta = \frac{y}{\sqrt{y^2 + z^2}} \quad \text{CodeCogsEqn (12) .png}$$

$$\sin \theta = \frac{z}{\sqrt{y^2 + z^2}} \quad \text{CodeCogsEqn (13) .png}$$

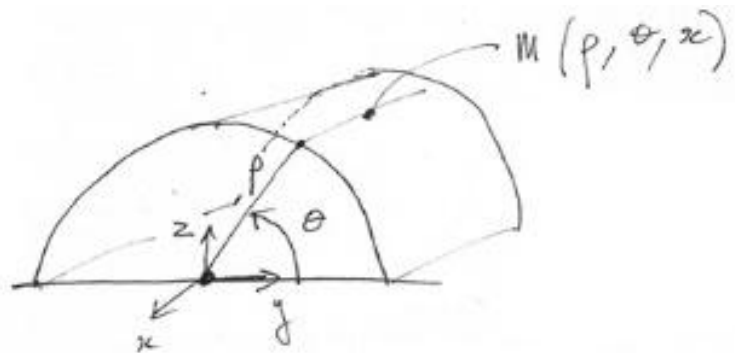


Figure 3 : Coordonnées cylindriques d'un point

Or les représentants du nuage de points sont

$$\forall i \in [1, n] / M_i(x_i, y_i, z_i) \quad \text{CodeCogsEqn (14) .png}$$

et dans notre système de représentation cylindrique nous avons

³ Il existe des formulations analytiques plus élaborées notamment celles de [Walter, Golub, Strebél 1994] permettant d'approcher un nuage de point par un cercle en calculant notamment son centre et son rayon. Notre proposition en terme de modèle dans un format plus accessible nécessite la connaissance et le relevé de l'axe longitudinal du cylindre mais possède l'avantage d'être plus explicite sur le plan de la géométrie.

$$\forall i \in [1, n] / M_i(\rho_i, \theta_i, x_i) \quad \rho_i = \sqrt{y_i^2 + z_i^2} \quad \text{CodeCogsEqn (15) .png}$$

$$\text{de plus } \cos \theta_i = \frac{y_i}{\rho_i} \text{ et } \sin \theta_i = \frac{z_i}{\rho_i} \quad \text{CodeCogsEqn (16) .png CodeCogsEqn (17) .png}$$

$$\text{d'où } \theta_i = \arccos\left(\frac{y_i}{\rho_i}\right) \quad \text{ainsi que } \theta_i = \arcsin\left(\frac{z_i}{\rho_i}\right)$$

CodeCogsEqn (18) .png

CodeCogsEqn (19) .png



Figure 4 : Dans ces conditions, approcher le rayon ρ du cylindre revient à se rapprocher au problème précédant (titre 1.1) et à effectuer une régression linéaire à partir des ρ_i et θ_i .

Dômes et régressions sphériques

Les points d'un dôme peuvent être repérés dans un système de coordonnées sphériques :

$$M(\rho, \varphi, \theta) \quad \text{avec} \quad M \begin{cases} x = \rho \sin \theta \cos \varphi \\ y = \rho \sin \theta \sin \varphi \\ z = \rho \cos \theta \end{cases}$$

CodeCogsEqn (20) .png

CodeCogsEqn (21) .png

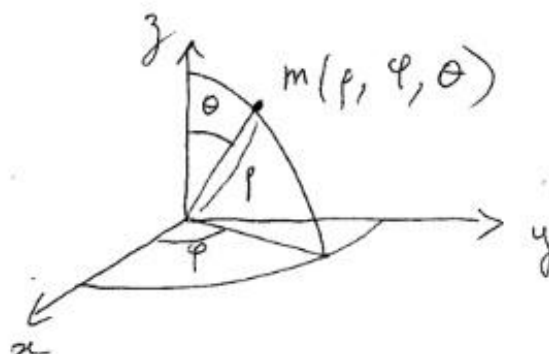


Figure 5. Coordonnées sphériques

Par ailleurs nous avons :

$$\rho = \sqrt{x^2 + y^2 + z^2} \quad \text{et} \quad \cos \theta = \frac{z}{\sqrt{x^2 + y^2 + z^2}}$$

CodeCogsEqn (22).png

CodeCogsEqn (23).png

Les représentants du nuage de points sont : $\forall i \in [1, n] / M_i(x_i, y_i, z_i)$ et dans notre système de représentation sphérique nous pouvons les exprimer sous la forme :

$$\forall i \in [1, n] / M_i(\rho_i, \varphi_i, \theta_i) \quad \text{avec :}$$

CodeCogsEqn (24).png

$$\rho_i = \sqrt{x_i^2 + y_i^2 + z_i^2} \quad \text{et} \quad \cos \theta_i = \frac{z_i}{\rho_i} \quad \text{avec} \quad \rho_i \neq 0$$

CodeCogsEqn (25).png

CodeCogsEqn (26).png

CodeCogsEqn (27).png

donc $\theta_i = \arccos\left(\frac{z_i}{\rho_i}\right)$

CodeCogsEqn (28).png

nous pouvons alors exprimer φ_i

CodeCogsEqn (29).png

avec $\theta_i \neq 0$ et $\varphi_i = \arccos\left(\frac{x_i}{\rho_i \sin \theta_i}\right)$

CodeCogsEqn (30).png

CodeCogsEqn (31).png

Dans ces conditions il est possible de réaliser une régression planaire à l'aide des ρ_i, φ_i et θ_i

et d'en déduire le rayon de la sphère⁴ ρ : CodeCogsEqn (32).png

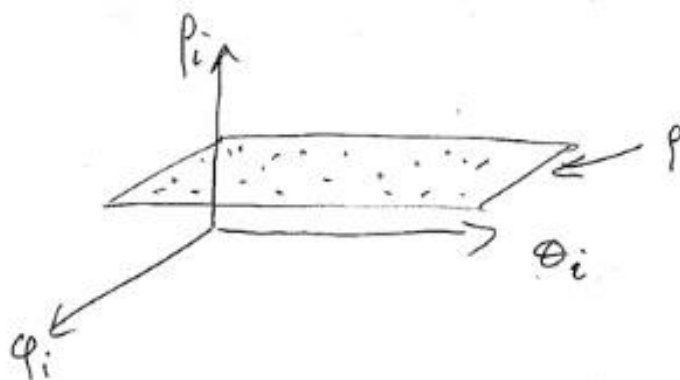


Figure 6 : régression planaire

Le potentiel d'un traitement en intelligence artificielle

Les nuages de points constituent le type de données que l'on aimerait soumettre aux outils d'intelligence artificielle (IA). Plusieurs éléments se dessinent comme porteur de potentiel et de direction de recherche que nous souhaitons en particulier pouvoir expérimenter avec des doctorants :

- les stratégies de classification utilisant les outils de la géométrie analytique articulées à ceux de l'analyse probabiliste semblent des éléments d'expérimentation précieux ;

⁴ Au delà des limites de cette formulation, ici encore l'analyse matricielle fournit avec [Robinson (Stephen M.)] des outils bien plus sophistiqués quoique moins accessibles du point de vue de la géométrie mais complémentaires à celle-ci.

- Les outils de régression - linéaires ou autres – et les systèmes de type ConvNet faisant appel à la norme L2 en utilisant des stratégies de minoration ou de majoration, il n'est pas impossible de voir apparaître des solutions hardware dédiés à l'apprentissage de données s'appuyant sur les produits présentées par Nvidia l'été dernier au Siggraph 2017 ;
- La possibilité du traitement autonome sans intervention humaine dans des visées de classification⁵ (par exemple le bloc A et le bloc B ont le même rayon de courbure) semble accessible à l'issue d'une phase d'apprentissage et de supervision.

Qualification sémantique

En précisant ces quelques outils portés à la disposition des doctorants: ceux de la géométrie analytique et leur équivalent plus sophistiqués ceux de l'analyse matricielle, cette conclusion prospective évoque une articulation entre les catégories de la géométrie et le potentiel de l'intelligence artificielle en reprenant l'argument de Jean Jacques Malmay⁶ qui évoque une complémentarité sémantique entre le vocabulaire de l'architecture et celui de la géométrie. Grâce à l'IA cette complémentarité se transformera certainement en une continuité.

Bibliographie

Alpaydin Ethem. (2016). *Machine learning - The new AI*. Cambridge MA : The MIT Press, 208p.
 Audet, D., Boucher, C., Caumartin, A., Skeene, C. (1986). *Probabilités et Statistiques*. Chicoutimi, Quebec, Canada : Gaëtan Morin Editeurs, 408 p.
 Gander, W., Golub, G.H., Strebel R. (1994). *Least-square fitting of circles and ellipses*. BIT, 1994 (34):558{578).
 Ginouvès R., Martin R *et alii*. (1985) *Dictionnaire méthodique de l'architecture grecque et romaine*. Rome : Ecole française de Rome – Athènes : Ecole française d'Athènes. CNRS.
 Robinson (Stephen M.) *Fitting Spheres by the Method of Least Squares*, in Comm. ACM , 1961, p. 491.
 Terracol Pascal. *Vocabulaire de géométrie pour l'architecture – Ontologies pour modèles de synthèse*. Paris : Presses de ponts et chaussées, 280 p.

Weisstein, Eric W. "*Least Squares Fitting*." From MathWorld--A Wolfram Web Resource. <http://mathworld.wolfram.com/LeastSquaresFitting.html>

Notice Biobibliographique

Pascal Terracol est architecte DPLG Docteur en Géographie de l'université Paris 1 Panthéon Sorbonne. Il est maître de conférences à l'ENSA de Paris Val de Seine et chercheur au laboratoire

⁵ Tel que le décrit [Alpaydin] p.179.

⁶ « Dans le domaine de l'archéologie, la description d'une forme architecturale est réalisée le plus souvent en employant le vocabulaire conventionnel de l'architecture qui varie selon les époques ou les sociétés étudiées. En archéologie classique, l'ouvrage consacré est le Dictionnaire méthodique de l'architecture grecque et romaine de R. Ginouvès *et alii*. Désigner une forme à l'aide de ce vocabulaire revient à reconnaître son rôle architectonique. Dès lors que le terme fonctionnel et la forme ne coïncident pas exactement ou qu'aucun terme adéquat ne permet de désigner ou d'expliquer une forme, le recours au vocabulaire de la géométrie peut s'avérer d'un grand secours. »

EVCAU (Espace Virtuel de Conception Architecturale et Urbaine). Il co-dirige le séminaire doctoral ACN Architecture et Cultures Numériques.

Thèmes de recherche : Géométrie analytique et veille technologique pour le projet d'architecture, du détail au grand paysage.

Dernière publication :

Terracol Pascal. *Vocabulaire de géométrie pour l'architecture – Ontologies pour modèles de synthèse.*

Paris : Presses de ponts et chaussées, 280 p.

pascal.terracol@paris-valdeseine.archi.fr

Option Explicit
'Regression Planaire Script written by <Pascal Terracol>
'Script version Tuesday, 27 February 2020 20:09:36
Call Main()

```
Sub Main()  
  Dim i, a, b  
  Dim x(6000)  
  Dim y(6000)  
  Dim z(6000)  
  
  Dim arrPoints, arrPt  
  arrPoints = Rhino.GetPointCoordinates  
  
  If IsArray(arrPoints) Then  
    i = 0  
    For Each arrPt In arrPoints  
      x(i) = arrPt(0)  
      y(i) = arrPt(1)  
      z(i) = arrPt(2)  
      i = i + 1  
    Next  
    i = i - 1  
    Rhino.Print("nb points:")  
    Rhino.Print(i)  
    Call my_Sum(x, y, z, i)  
  End If  
End Sub
```

```
Sub my_Sum(x, y, z, n)  
  Dim i  
  Dim dblsumx, dblsumy, dblsumz, dblsumx2, dblsumy2, dblsumxy, dblsumxz, dblsumyz  
  Dim xmin, xmax, ymin, ymax, zmin, zmax  
  Dim arrStartPoint, arrEndPoint, arrLine  
  Dim a,b,c,d,e,f,g,h,ii ' Matrice A  
  Dim ta,tb,tc,td,te,tf,tg,th,ti ' matrice Transposé tA  
  Dim DetA ' Determinant de A  
  Dim AA,BB,CC' coefficients du plan  
  
  Rhino.print n  
  a = 0.0  
  b = 0.0  
  c = 0.0  
  d = 0.0  
  e = 0.0  
  f = 0.0  
  g = 0.0  
  h = 0.0  
  ii = 0.0  
  ta = 0.0  
  tb = 0.0  
  tc = 0.0  
  td = 0.0  
  te = 0.0
```

```

tf = 0.0
tg = 0.0
th = 0.0
ti = 0.0
DetA = 0.0
dblsumx = 0.0
dblsumy = 0.0
dblsumz = 0.0
dblsumx2 = 0.0
dblsumy2 = 0.0
dblsumxy = 0.0
dblsumxz = 0.0
dblsumyz = 0.0
xmin = 100000000.0
ymin = 100000000.0
zmin = 100000000.0
xmax = -100000000.0
ymax = -100000000.0
zmax = -100000000.0
'Boxing min max
For i = 0 To n
    If x(i) > xmax Then
        xmax = x(i)
    End If
    If y(i) > ymax Then
        ymax = y(i)
    End If
    If z(i) > zmax Then
        zmax = z(i)
    End If
    If x(i) < xmin Then
        xmin = x(i)
    End If
    If y(i) < ymin Then
        ymin = y(i)
    End If
    If z(i) < zmin Then
        zmin = z(i)
    End If
Next

For i = 0 To n
    dblsumx = dblsumx + x(i)
    dblsumy = dblsumy + y(i)
    dblsumz = dblsumz + z(i)
    dblsumxy = dblsumxy + (x(i) * y(i))
    dblsumxz = dblsumxz + (x(i) * z(i))
    dblsumyz = dblsumyz + (y(i) * z(i))
    dblsumx2 = dblsumx2 + (x(i) * x(i))
    dblsumy2 = dblsumy2 + (y(i) * y(i))
Next
a = dblsumx2
b = dblsumxy
c = dblsumx

```

```

d = b
e = dblsumy2
f = dblsumy
g = dblsumx
h = dblsumy
ii = n
DetA = (a * e * ii) + (b * f * g) + (c * d * h) - (c * e * g) - (f * h * a) - (ii * b * d)
Rhino.Print("DetA")
Rhino.Print(DetA)
'inversion
ta = (e * ii - f * h) / DetA
tb = (c * h - b * ii) / DetA
tc = (b * f - c * e) / DetA
td = (f * g - d * ii) / DetA
te = (a * ii - c * g) / DetA
tf = (c * d - a * f) / DetA
tg = (d * h - e * g) / DetA
th = (b * g - a * h) / DetA
ti = (a * e - b * d) / DetA
'resolution
AA = (ta * dblsumxz) + (tb * dblsumyz) + (tc * dblsumz)
BB = (td * dblsumxz) + (te * dblsumyz) + (tf * dblsumz)
CC = (tg * dblsumxz) + (th * dblsumyz) + (ti * dblsumz)

Rhino.Print("Boxing")
Rhino.Print(xmin)
Rhino.Print(xmax)
Rhino.Print(ymin)
Rhino.Print(ymax)
Rhino.Print(zmin)
Rhino.Print(zmax)
'tracons la droite d'après le boxing
'
arrStartPoint = Array(xmin, (dblbb * xmin) + dblaa, 0.0)
arrEndPoint = Array(xmax, (dblbb * xmax) + dblaa, 0.0)
arrStartPoint = Array(xmin, ymin, (AA * xmin) + (BB * ymin) + CC)
arrEndPoint = Array(xmax, ymax + (AA * xmax) + (BB * ymax) + CC)
Call Rhino.AddLine(arrStartPoint, arrEndPoint)
End Sub

```