Grouping Objects by Linear Pattern

Ruben Zamar

Department of Statistics
University of British Columbia
Vancouver, Canada

Coauthors and collaborators:

Stefan Van Aelst (Ghent University)
Steven Wang (York University)
Rong Zhu (McMaster University)
Matias Salibian-Barrera (UBC)
Justin Harrington (UBC)
Will Welch (UBC)
Outline

➤ Grouping by linear patterns
➤ Our basic building block (LGA)
➤ The number of random starting points
➤ The number of groups
➤ The generalized LGA (GLGA)
➤ Application to Biology (Allometry data)
➤ Application to sport (hockey data)
➤ Application to Genomics (SNP data)
Clustering Goals
Clustering Goals

- Homogeneous subgroups in a dataset
Clustering Goals

➤ Homogeneous subgroups in a dataset

➤ Interesting patterns in a dataset
Clustering Algorithms

Clustering algorithms are effective when the clusters are separated groups of points.
But some patterns **cannot be found** this way ...
Tilted Pi Pattern
Our Goal
Our Goal

➢ To find groups clustered around hyperplanes of different dimensions

\[ 0 \leq l_i \leq d - 1 \quad i = 1, 2, ..., N \]
Example $d = 3$ and $N = 3$

- $l_1 = 1$ points clustering around a line.
- $l_2 = 0$ points clustering around a point.
- $l_3 = 2$ points clustering around plane.
Formulating the Problem
In general, a $d - j$ dimensional hyperplane ($j \leq d$) is given by the equation

$$Ax = B$$

- $A$ is an orthogonal $j \times d$ matrix
- $B$ is a $j$-dimensional vector.
Formulating the Problem

**In general, a** \( d - j \) **dimensional hyperplane** \((j \leq d)\) **is given by the equation**

\[
Ax = B
\]

- **A** is an orthogonal \( j \times d \) matrix
- **B** is a \( j \)-dimensional vector.

**Therefore we search for** \( N \) **groups with “central hyperplanes”**

\[(A_1, B_1), (A_2, B_2), \ldots, (A_N, B_N)\]
Generalized LGA

\[ GLGA = LGA + GAP \]

- LGA finds the “best” partition of the data around \( k \) hyperplanes of dimension \( d-1 \).
Generalized LGA

\[ \text{GLGA} = \text{LGA} + \text{GAP} \]

- LGA finds the “best” partition of the data around \( k \) hyperplanes of dimension \( d-1 \).
- GAP sequentially considers the possibility of increasing the number of clusters by one and stops when the addition of a cluster doesn’t provide a significant improvement.
Simple Example

\[ d = 2 \quad \text{and} \quad N = 3 \]
Finding 1-d Hyperplanes (Lines)
Projecting on the Lines and Finding 0-d Hyperplanes (Points)
The Final Result
The Basic LGA

**Goal**: to find $k$ groups around hyperplanes of dimension $d - 1$
The Basic LGA

**Goal**: to find \( k \) groups around hyperplanes of dimension \( d - 1 \)

Some proposed methods to find linear groups:

- Murtagh and Raftery (1984)
- Gawrysiak et al. (2000)
- Spath (1982, 1985)
- Desarbo, Oliver and Rangaswamy (1989)
- Wedel and Kistemaker (1989)
- Kamgar-Parsi, Kamgar-Parsi and Wechsler (1990)
- Gawrysiak, Okoniewski and Rybinski (2000)
The Basic LGA

**Goal**: to find $k$ groups around hyperplanes of dimension $d - 1$

Some proposed methods to find linear groups:

- Murtagh and Raftery (1984)
- Gawrysiak et al. (2000)
- Spath (1982, 1985)
- Desarbo, Oliver and Rangaswamy (1989)
- Wedel and Kistemaker (1989)
- Kamgar-Parsi, Kamgar-Parsi and Wechsler (1990)
- Gawrysiak, Okoniewski and Rybinski (2000)

These methods assume a specified output variable.
Clustering and linear grouping are often used in the context of unsupervised learning.
Clustering and linear grouping are often used in the context of **unsupervised learning**.

Unsupervised learning is characterized by the absence of a specified **output variable**.
Unsupervised Learning Setup

- Clustering and linear grouping are often used in the context of **unsupervised learning**.
- Unsupervised learning is characterized by the absence of a specified **output variable**.
- Moreover, different linear groups may involve **different subsets of variables**.
Example
Response Variable = \( Y \)
Response Variable = $Z$
Orthogonal Regression

Linear Residual  =  Vertical distance
Orthogonal Regression

Linear Residual = Vertical distance

Orthogonal Residual = Euclidean distance
Orthogonal Residuals
Orthogonal Regression
Orthogonal Regression

Given \( z_1, z_2, \ldots, z_n \) in \( \mathbb{R}^d \), the fitting \((d - 1)\)-dimensional hyperplane

\[
(\hat{\alpha}, \beta) = \{ z : \hat{\alpha}'z = \hat{\beta}, \|\hat{\alpha}\| = 1 \}
\]

is defined as the solution to the problem:

\[
\text{Minimize } \sum_{i=1}^{n} (\alpha'z_i - \beta)^2
\]
Orthogonal Regression

\[ \bar{z} = \frac{1}{n} \sum z_i \quad \text{(Sample Mean)} \]

\[ S = \frac{1}{n} \sum (z_i - \bar{z})(z_i - \bar{z})' \quad \text{(Sample Covariance)} \]

The OR estimates are:

\[ \hat{\alpha} = \text{normalized first eigenvector of } S \]

\[ \hat{\beta} = \hat{\alpha}' \bar{z} \]
The LGA Algorithm
The LGA Algorithm

**INPUT**: $d$-dimensional data points $z_1, z_2, ..., z_n$ and the desired number $k$ of groups
The LGA Algorithm

**INPUT**: $d$-dimensional data points $z_1, z_2, \ldots, z_n$ and the desired number $k$ of groups

**OUTPUT**: The “best partition” of the dataset into $k$ groups centered around hyperplanes of dimension $d - 1$
1) **Initialization**: Initial hyperplanes are defined by the exact fitting of $k$ sub-samples of size $d$
LGA Step-by-Step

1) **Initialization**: Initial hyperplanes are defined by the exact fitting of $k$ sub-samples of size $d$

2) **Forming $k$ groups**: Each data point is assigned to its closest hyperplane using Euclidean distances.
1) **Initialization:** Initial hyperplanes are defined by the exact fitting of $k$ sub-samples of size $d$

2) **Forming $k$ groups:** Each data point is assigned to its closest hyperplane using Euclidean distances.

3) **Computing $k$ Hyperplanes:** New hyperplanes are computed applying orthogonal regression to each group.
LGA Step-by-Step

1) **Initialization**: Initial hyperplanes are defined by the exact fitting of $k$ sub-samples of size $d$

2) **Forming $k$ groups**: Each data point is assigned to its closest hyperplane using Euclidean distances.

3) **Computing $k$ Hyperplanes**: New hyperplanes are computed applying orthogonal regression to each group.

4) Steps 2) and 3) are repeated several times
The Number of Random Starts

\[ p = \frac{\binom{n_1}{d} \binom{n_2}{d} \cdots \binom{n_k}{d}}{\binom{n_1 + n_2 + \cdots + n_k}{d}} \]

\[ 0.95 = 1 - (1 - p)^m \]

\[ m = \frac{\log (0.05)}{\log (1 - p)} \]
The Number of Random Starts

Table 1: Number of random starts for 95% probability of at least one good subset.

<table>
<thead>
<tr>
<th></th>
<th></th>
<th>k = 2</th>
<th>k = 3</th>
<th>k = 4</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>1:1</td>
<td>1:2</td>
<td>1:1:1</td>
</tr>
<tr>
<td>d</td>
<td></td>
<td>1:1:1</td>
<td></td>
<td>1:1:1:1</td>
</tr>
<tr>
<td>2</td>
<td></td>
<td></td>
<td></td>
<td>23(24)</td>
</tr>
<tr>
<td>3</td>
<td></td>
<td>9(9)</td>
<td>13(13)</td>
<td>34(35)</td>
</tr>
<tr>
<td>4</td>
<td></td>
<td>10(10)</td>
<td>17(17)</td>
<td>44(45)</td>
</tr>
<tr>
<td>5</td>
<td></td>
<td>11(12)</td>
<td>52(51)</td>
<td>53(56)</td>
</tr>
</tbody>
</table>
The Number of Random Starts

The needed number $m$ of random starts depends on:

- *The number* $k$ *of groups,*
- *The relative size of the groups,*
- *The dimension* $d$ *of the data.*
The Number of Random Starts

The needed number $m$ of random starts depends on:

- *The number* $k$ *of groups,*
- *The relative size of the groups,*
- *The dimension* $d$ *of the data.*
- $m$ *doesnt depend much on the data size,* $n.$
The Number of Groups
The number $k$ of groups is an input of our algorithm.
The number $k$ of groups is an input of our algorithm.

- $k$ may be suggested by additional subject field information (species, gender, location, etc.)
The number $k$ of groups is an input of our algorithm

- $k$ may be suggested by additional subject field information (species, gender, location, etc.)

- Finding the number of groups may be the most important goal of the research
Graphical Approach
Graphical Approach

- *Plots may provide visual information*
Graphical Approach

- Plots may provide visual information
- Mainly helpful for 2 or 3 dimensional data
Graphical Approach

- Plots may provide visual information
- Mainly helpful for 2 or 3 dimensional data
- Eyes may fail to identify linear patterns in heavily overlapping regions
Figure 4: The height and volume of young and old trees.
The GAP Statistic

Tibshirani, Walther and Hastie (2001) proposed the GAP statistic to determine the number of clusters in a data set.
The GAP Statistic

Tibshirani, Walther and Hastie (2001) proposed the GAP statistic to determine the number of clusters in a data set.

GAP compares the pooled within-cluster sum of squares around the cluster centers with its expectation under a null reference distribution.
The GAP Statistic

Tibshirani, Walther and Hastie (2001) proposed the GAP statistic to determine the number of clusters in a data set.

GAP compares the pooled within-cluster sum of squares around the cluster centers with its expectation under a null reference distribution.

The null distribution is obtained by generating uniformly distributed points on the hyper-rectangle aligned with the principal components of the data.
The GAP Statistic

Tibshirani, Walther and Hastie (2001) proposed the GAP statistic to determine the number of clusters in a data set.

GAP compares the pooled within-cluster sum of squares around the cluster centers with its expectation under a null reference distribution.

The null distribution is obtained by generating uniformly distributed points on the hyper-rectangle aligned with the principal components of the data.

The (modified) GAP statistic for linear grouping is obtained by replacing “distance to the center” by “distance to the hyperplane”.
The GAP Statistic (continued)

\[ GAP(k) = \left[ \frac{1}{B} \sum_{b=1}^{B} \log (SSR_k(b)) \right] - \log (SSR_k) \]
The GAP Statistic (continued)

\[
\text{GAP}(k) = \left[ \frac{1}{B} \sum_{b=1}^{B} \log (\text{SSR}_k(b)) \right] - \log (\text{SSR}_k)
\]

\[\hat{k} = \text{smallest } k \text{ such that } \text{GAP}(k) \geq \text{GAP}(k + 1) - s_{k+1}\]
The GAP Statistic (continued)

\[
GAP (k) = \left[ \frac{1}{B} \sum_{b=1}^{B} \log (SSR_k (b)) \right] - \log (SSR_k)
\]

\[
\hat{k} = \text{smallest } k \text{ such that } GAP (k) \geq GAP (k + 1) - s_{k+1}
\]

\[
s_{k+1} = S_{k+1} \sqrt{1 + (1/B)}
\]
The GAP Statistic (continued)

\[ GAP(k) = \left[ \frac{1}{B} \sum_{b=1}^{B} \log (SSR_k(b)) \right] - \log (SSR_k) \]

\[ \hat{k} = \text{smallest } k \text{ such that } GAP(k) \geq GAP(k + 1) - s_{k+1} \]

\[ s_{k+1} = S_{k+1} \sqrt{1 + (1/B)} \]

\[ S_{k+1} = \text{Standard Deviation of } \log (SSR_{k+1}(b)) \]
Application to Allometry Data
Biologists investigate the relationships between sizes of organs for different species.
Biologists investigate the relationships between sizes of organs for different species.

The (transformed) sizes of organs are linearly related.
Application to Allometry Data

Biologists investigate the relationships between sizes of organs for different species.

The (transformed) sizes of organs are linearly related.

Linear associations differ across species because of different living habits, environment, food sources, etc.
Application to Allometry Data

Biologists investigate the relationships between sizes of organs for different species.

The (transformed) sizes of organs are linearly related.

Linear associations differ across species because of different living habits, environment, food sources, etc.

Grouping according to different linear patterns is necessary.
Application to Allometry Data

Biologists investigate the relationships between sizes of organs for different species.

The (transformed) sizes of organs are linearly related.

Linear associations differ across species because of different living habits, environment, food sources, etc.

Grouping according to different linear patterns is necessary.

Biologists make manual assignments based on their scientific experience (Jerison 1973).
Figure 6: Logarithms of Olfactory Bulb vs. Brain Weight for some mammal species: Insectivores (i), Carnivores (c), Prosimians (p), Apes (a), Monkeys (m), Human (h) and Horse (o).
Application to Allometry Data

LGA with $k = 3$ (Dr. Jerison’s hypothesis)
Application to Allometry Data

LGA with $k = 2$ (GAP result)
<table>
<thead>
<tr>
<th>LGA with $k=3$</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>I</strong></td>
<td>insectivores, carnivores, horses,</td>
</tr>
<tr>
<td><strong>II</strong></td>
<td>prosimians and apes</td>
</tr>
<tr>
<td><strong>III</strong></td>
<td>monkeys and human</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>LGA &amp; GAP</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>I</strong></td>
<td>insectivores, carnivores, horses, prosimians</td>
</tr>
<tr>
<td><strong>II</strong></td>
<td>monkeys, apes and human</td>
</tr>
</tbody>
</table>
Application to Sport Data
Application to Sport Data

- Performance of 871 players in the 94/95 Hockey League
## Application to Sport Data

### Performance of 871 players in the 94/95 Hockey League

<table>
<thead>
<tr>
<th>Variables</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>PTS</td>
<td># of Goal Scored + # of Assists</td>
</tr>
<tr>
<td>P/M</td>
<td>Plus/Minus Rating</td>
</tr>
<tr>
<td></td>
<td>+ team scored,</td>
</tr>
<tr>
<td></td>
<td>- oponent team scored</td>
</tr>
<tr>
<td>PIM</td>
<td>Total penalty time (minutes)</td>
</tr>
<tr>
<td>PP</td>
<td>Total number of power-play goals scored</td>
</tr>
</tbody>
</table>
We applied OR-grouping with k=3

The results:

<table>
<thead>
<tr>
<th>Group</th>
<th>PTS</th>
<th>P/M</th>
<th>PIM</th>
<th>PP</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>-0.156</td>
<td>0.015</td>
<td>0.001</td>
<td>0.988</td>
</tr>
<tr>
<td>2</td>
<td>-0.221</td>
<td>0.029</td>
<td>-0.003</td>
<td>0.975</td>
</tr>
<tr>
<td>3</td>
<td>0.113</td>
<td>-0.010</td>
<td>0.001</td>
<td>-0.994</td>
</tr>
</tbody>
</table>
Figure S: Plot of PP versus PTS for the NHL 94-95 competition with the three groups detected by LGA.
Application to Genomics
The Gene/Environment Team at the ICAPTURE Center is currently using the fluorescent based Taqman technology to genotype 10,000 enrolled patients for 160 single nucleotide polymorphisms (SNP).
The Gene/Environment Team at the ICAPTURE Center is currently using the fluorescent based Taqman technology to genotype 10,000 enrolled patients for 160 single nucleotide polymorphisms (SNP).

SNP (pronounced ‘snip’), is a small genetic variation that can occur within a person’s DNA sequence.
The Gene/Environment Team at the ICAPTURE Center is currently using the fluorescent based Taqman technology to genotype 10,000 enrolled patients for 160 single nucleotide polymorphisms (SNP).

SNP (pronounced ‘snip’), is a small genetic variation that can occur within a person’s DNA sequence. Example: the DNA segment AGGTTA changes to ATGTTA.
The Gene/Environment Team at the ICAPTURE Center is currently using the fluorescent based Taqman technology to genotype 10,000 enrolled patients for 160 single nucleotide polymorphisms (SNP).

SNP (pronounced ‘snip’), is a small genetic variation that can occur within a person’s DNA sequence. Example: the DNA segment AGGTGA changes to ATGTGA.

On average, SNPs occur in the human population approximately 1 percent of the time.
Application to Genomics

- The Gene/Environment Team at the ICAPTURE Center is currently using the fluorescent based Taqman technology to genotype 10,000 enrolled patients for 160 single nucleotide polymorphisms (SNP).

- SNP (pronounced ‘snip’), is a small genetic variation that can occur within a person’s DNA sequence. **Example: the DNA segment AGGTTA changes to ATGT TA.**

- On average, SNPs occur in the human population approximately 1 percent of the time.

- SNPs found within a coding sequence are of particular interest (more likely to alter the biological function of a protein).
The TaqMan Technology
The TaqMan Technology

The TaqMan assay is a popular high-throughput genotyping technology (Livak et al. 1995)
The TaqMan Technology

The TaqMan assay is a popular high-throughput genotyping technology (Livak et al. 1995)

Individual samples are arranged in a 384-well plate
The TaqMan Technology

The TaqMan assay is a popular high-throughput genotyping technology (Livak et al. 1995)
Individual samples are arranged in a 384-well plate
ROX Normalization
ROX Normalization

- **VIC** (for Allele X), **FAM** (for Allele Y) and **ROX** (Passive Reference) fluorescence values are measured concurrently for each well.
ROX Normalization

- **VIC** (for Allele X), **FAM** (for Allele Y) and **ROX** (Passive Reference) fluorescence values are measured concurrently for each well.

- **ROX** account for well-to-well differences and for differences in the PCR mastermix.
ROX Normalization

- **VIC** (for Allele X), **FAM** (for Allele Y) and **ROX** (Passive Reference) fluorescence values are measured concurrently for each well.
- **ROX** account for well-to-well differences and for differences in the PCR mastermix.
- **ROX** dye intensities are assumed unchanged after PCR amplification and hence can be used to normalize the data.
Call Rate vs ROX
Plate 3 - Raw Data
Plate 3 - ROX Normalized Data
Plate 3 - Raw Data - LGA Grouping
Summary
Summary

➤ LGA finds groups that follow different linear relationships
Summary

➤ LGA finds groups that follow different linear relationships
  • *LGA can find overlapping linear patterns*
Summary

➤ LGA finds groups that follow different linear relationships

- LGA can find overlapping linear patterns
- LGA tolerates the presence of “nuisance” variables
Summary

➤ LGA finds groups that follow different linear relationships
  - \textit{LGA can find overlapping linear patterns}
  - \textit{LGA tolerates the presence of “nuisance” variables}

➤ We are currently “fine-tuning” an algorithm to genotype SNPs using LGA (Gyan, G., Van Aelst, S., Welch, W. and Zamar, (2006)).
Summary

➤ LGA finds groups that follow different linear relationships
  - LGA can find overlapping linear patterns
  - LGA tolerates the presence of “nuisanse” variables

➤ We are currently “fine-tuning” an algorithm to genotype SNPs using LGA (Gyan, G., Van Aelst, S., Welch, W. and Zamar, (2006)).

➤ Justin Harrington constructed an R-package to implement LGA and GLGA (available from http://md.stat.ubc.ca/lga )
Summary

- LGA finds groups that follow different linear relationships
  - LGA can find overlapping linear patterns
  - LGA tolerates the presence of “nuisance” variables
- We are currently “fine-tuning” an algorithm to genotype SNPs using LGA (Gyan, G., Van Aelst, S., Welch, W. and Zamar, (2006)).
- Justin Harrington constructed an R-package to implement LGA and GLGA (available from http://md.stat.ubc.ca/lga)
- Scaled up algorithm to handle higher dimensions and very large datasets (Harrington, J. and Salibian-Barrera, M., 2006)
Summary

➢ LGA finds groups that follow different linear relationships
  ◦ *LGA can find overlapping linear patterns*
  ◦ *LGA tolerates the presence of “nuisance” variables*

➢ We are currently “fine-tuning” an algorithm to genotype SNPs using LGA (Gyan, G., Van Aelst, S., Welch, W. and Zamar, (2006)).

➢ Justin Harrington constructed an R-package to implement LGA and GLGA *(available from http://md.stat.ubc.ca/lga)*

➢ Scaled up algorithm to handle higher dimensions and very large datasets (Harrington, J. and Salibian-Barrera, M., 2006)

➢ Robust LGA using trimmed means to deal with outliers (Pison, G., Van Aelst, S. and Zamar, R.H., 2006)
Summary

- LGA finds groups that follow different linear relationships
  - *LGA can find overlapping linear patterns*
  - *LGA tolerates the presence of “nuisance” variables*

- We are currently “fine-tuning” an algorithm to genotype SNPs using LGA (Gyan, G., Van Aelst, S., Welch, W. and Zamar, (2006)).

- Justin Harrington constructed an R-package to implement LGA and GLGA (available from http://md.stat.ubc.ca/lga)

- Scaled up algorithm to handle higher dimensions and very large datasets (Harrington, J. and Salibian-Barrera, M., 2006)

- Robust LGA using trimmed means to deal with outliers (Pison, G., Van Aelst, S. and Zamar, R.H., 2006))

- We plan to extend this approach to find nonlinear patterns.
Summary

➤ LGA finds groups that follow different linear relationships
  • *LGA can find overlapping linear patterns*
  • *LGA tolerates the presence of “nuisance” variables*

➤ We are currently “fine-tuning” an algorithm to genotype SNPs using LGA (Gyan, G., Van Aelst, S., Welch, W. and Zamar, (2006)).

➤ Justin Harrington constructed an R-package to implement LGA and GLGA ([available from http://md.stat.ubc.ca/lga](http://md.stat.ubc.ca/lga))

➤ Scaled up algorithm to handle higher dimensions and very large datasets (Harrington, J. and Salibian-Barrera, M., 2006)

➤ Robust LGA using trimmed means to deal with outliers (Pison, G., Van Aelst, S. and Zamar, R.H., 2006))

➤ We plan to extend this approach to find nonlinear patterns.

Thanks for your attention!