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

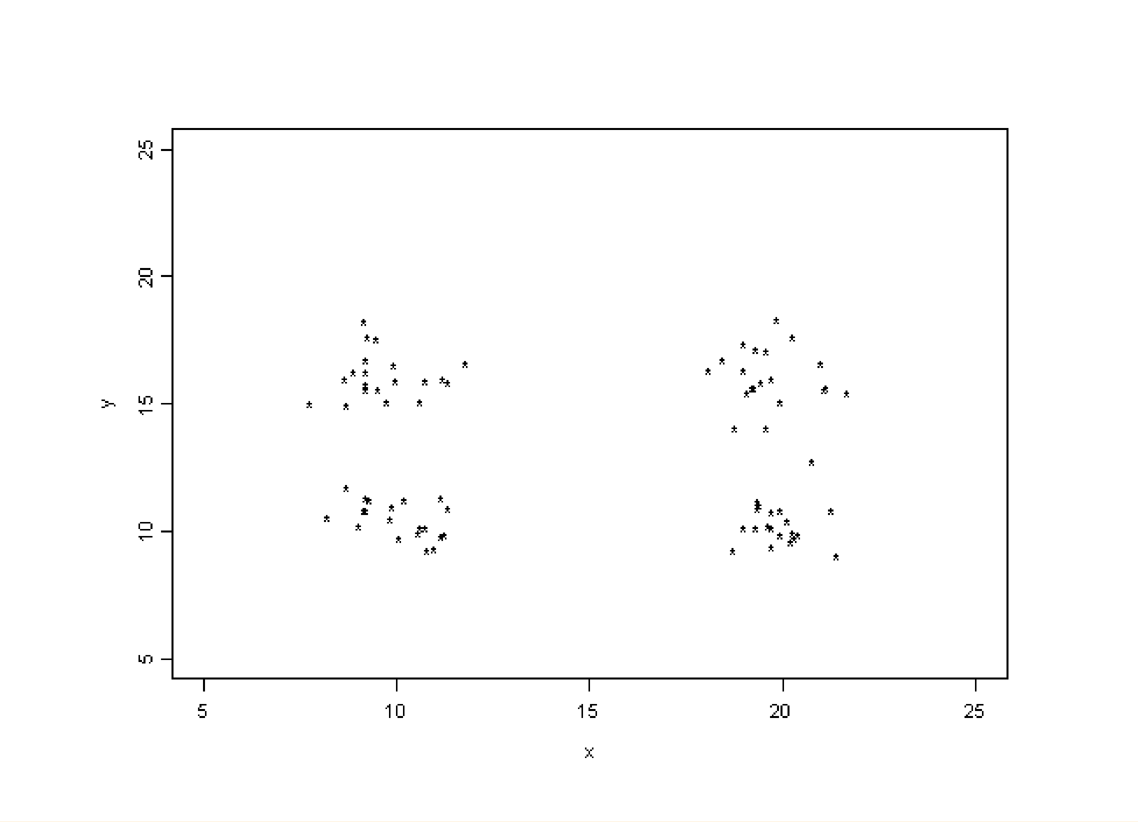
- **Homogeneous subgroups** in a dataset

Clustering Goals

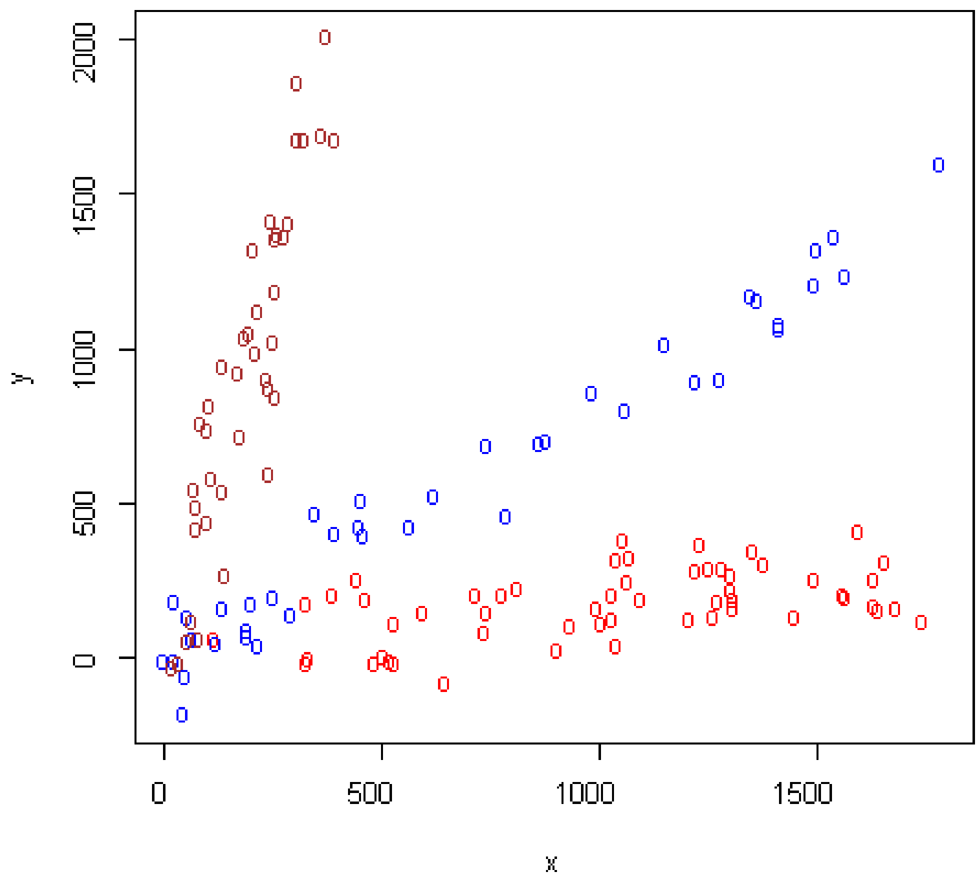
- ➤ **Homogeneous subgroups** in a dataset
- ➤ **Interesting patterns** in a dataset

Clustering Algorithms

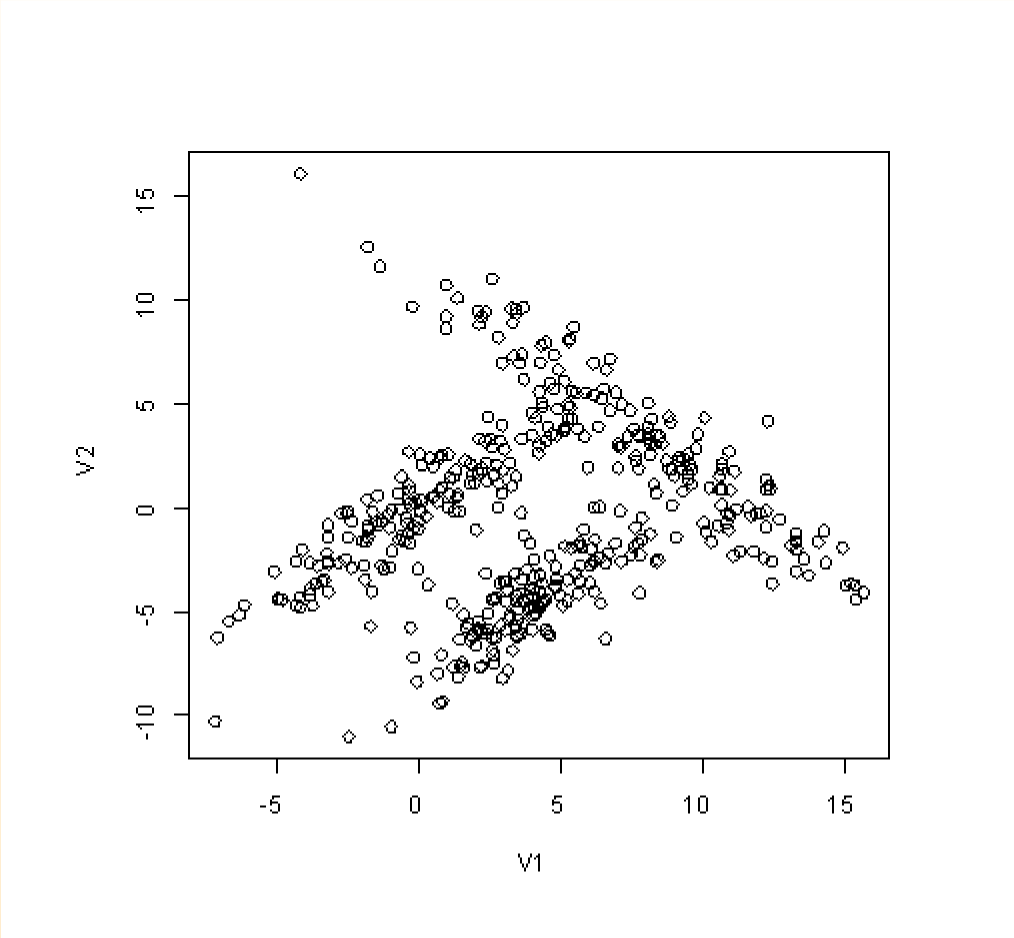
Clustering algorithms are effective when the clusters are separated groups of points



But some patterns **cannot be found** this way ...



Tilted Pi Pattern



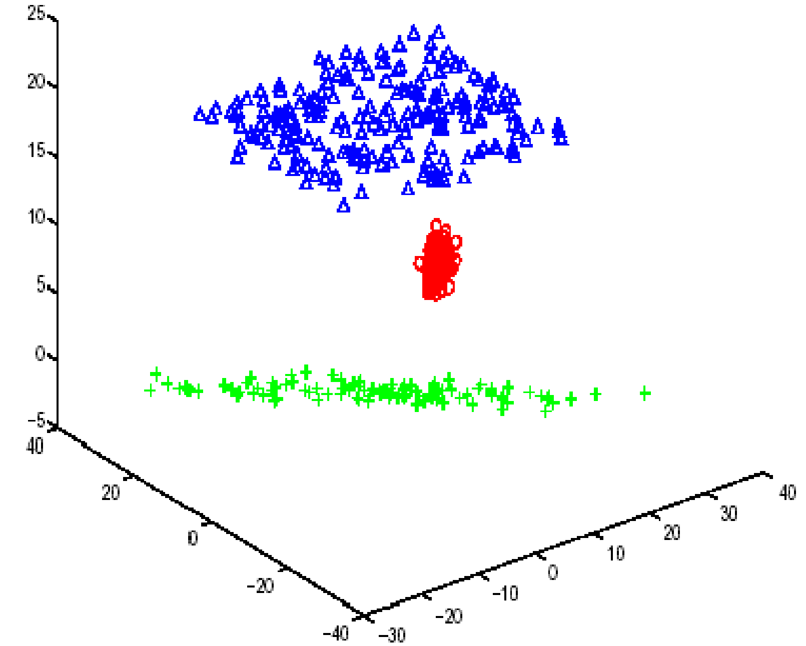
Our Goal

Our Goal

- To find groups clustered around hyperplanes of different dimensions

$$0 \leq l_i \leq d - 1 \qquad i = 1, 2, ..., N$$

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

Formulating the Problem

➤ In general, a $\mathbf{d} - \mathbf{j}$ dimensional hyperplane ($\mathbf{j} \leq \mathbf{d}$) is given by the equation

$$Ax = B$$

- *A is an orthogonal $\mathbf{j} \times \mathbf{d}$ matrix*
- *B is a $\mathbf{j}$-dimensional vector.*

Formulating the Problem

➤ In general, a $\mathbf{d-j}$ dimensional hyperplane ($\mathbf{j} \leq \mathbf{d}$) is given by the equation

$$Ax = B$$

- *A is an orthogonal $\mathbf{j} \times \mathbf{d}$ matrix*
- *B is a $\mathbf{j}$-dimensional vector.*

➤ Therefore we search for $\mathbf{N}$ groups with "central hyperplanes"

$$(\mathbf{A_1}, \mathbf{B_1}), (\mathbf{A_2}, \mathbf{B_2}), ..., (\mathbf{A_N}, \mathbf{B_N})$$

Generalized LGA

$$GLGA = LGA + GAP$$

- *LGA finds the "best" partition of the data around* **k** *hyperplanes of dimension* **d-1**.

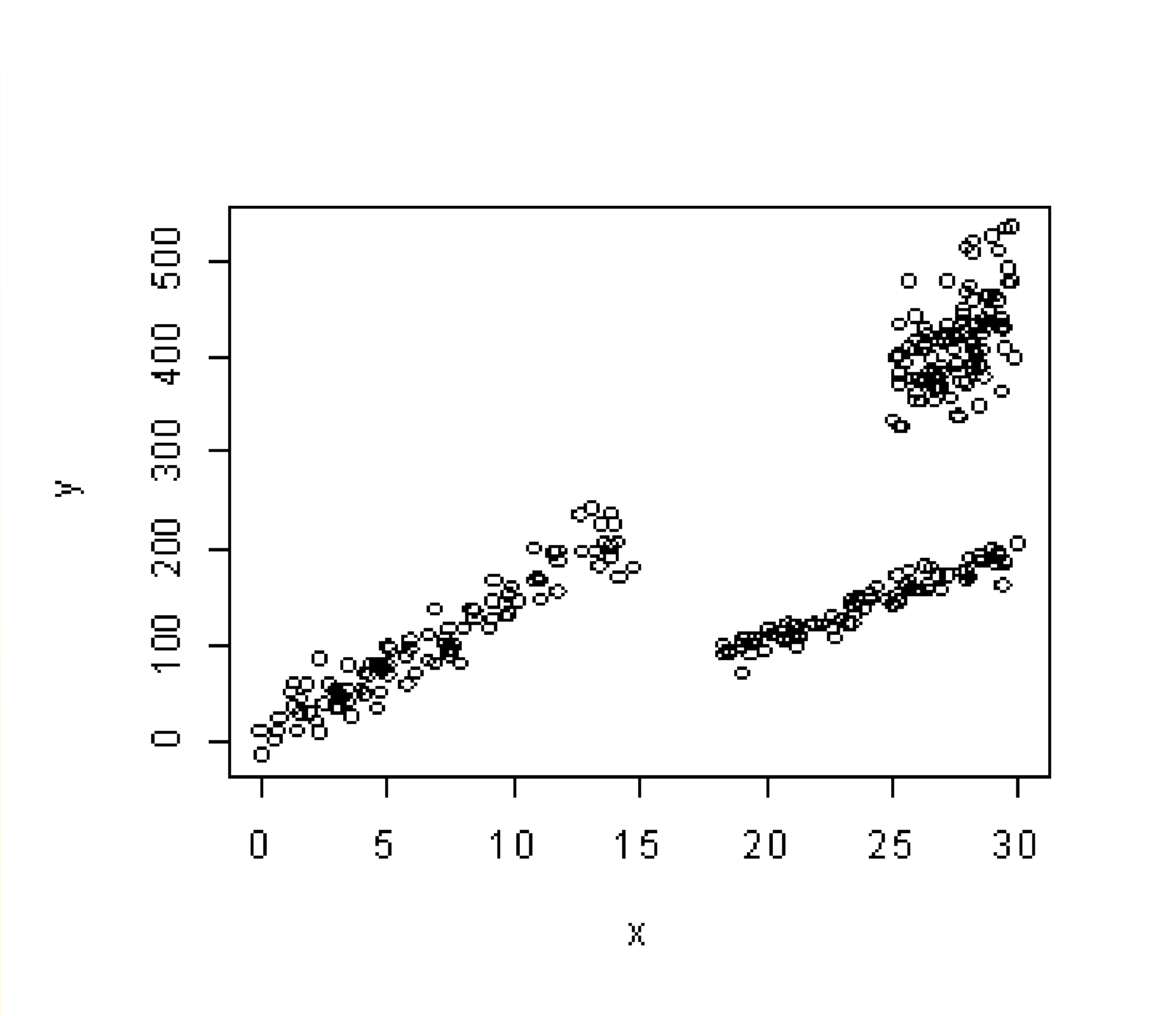
Generalized LGA

$$GLGA = LGA + GAP$$

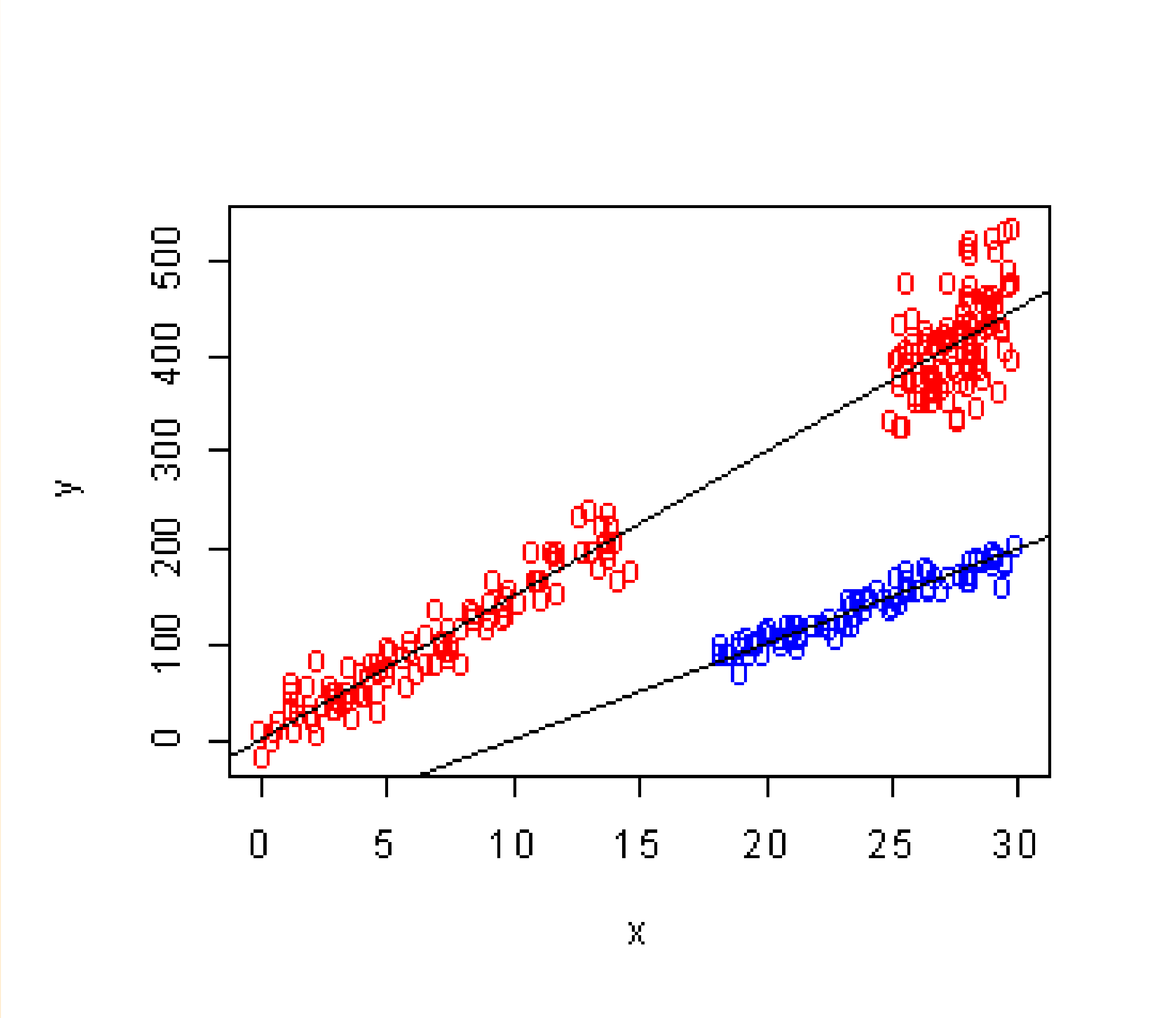
- *LGA finds the "best" partition of the data around* **k** *hyperplanes of dimension* **d-1***.*
- *GAP sequentially considers the possibility of increasing the number of clusters by one and stops when the addition of a cluster doesn't provide a significant improvement.*

Simple Example

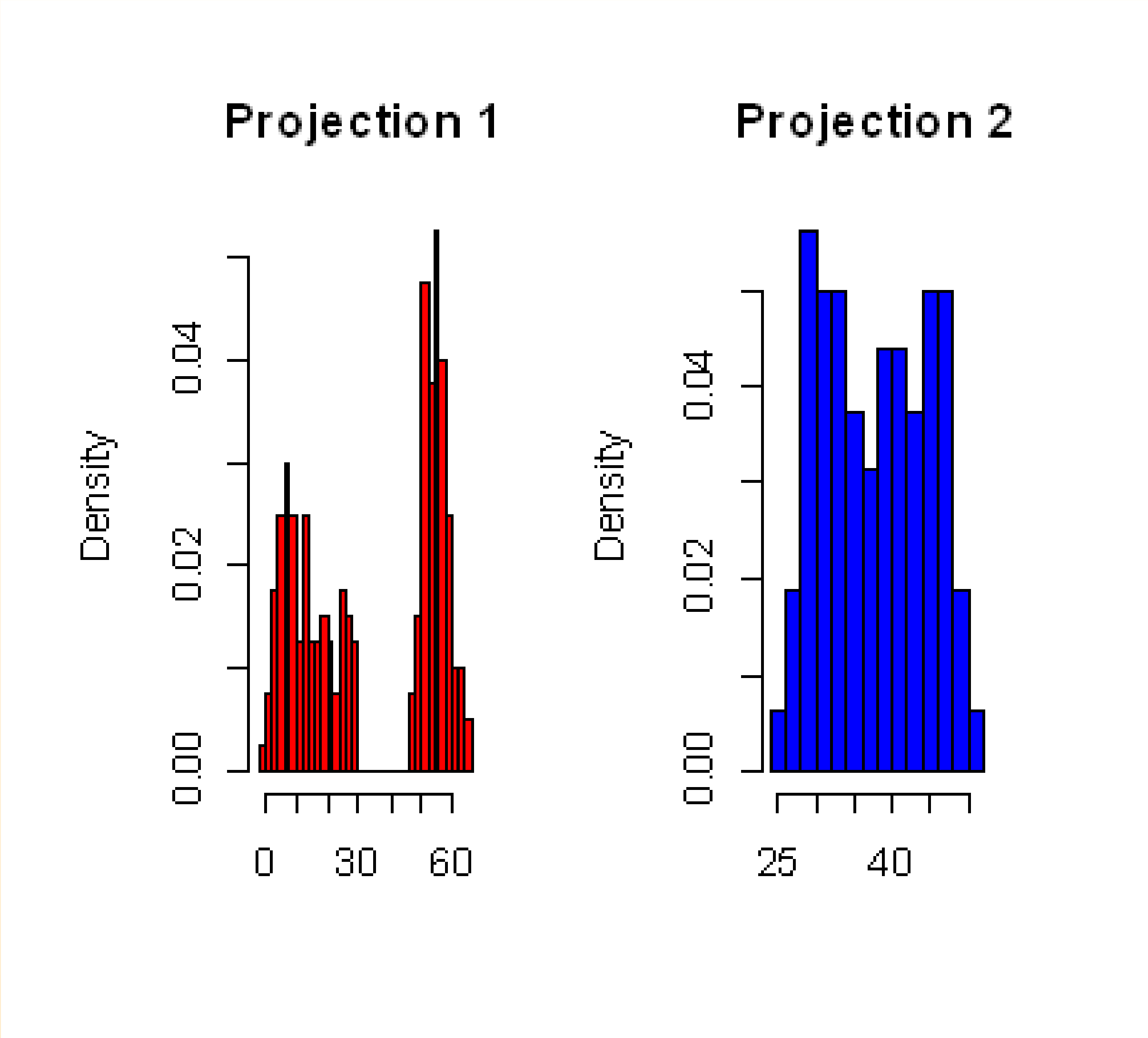
$$d = 2 \text{ and } N = 3$$



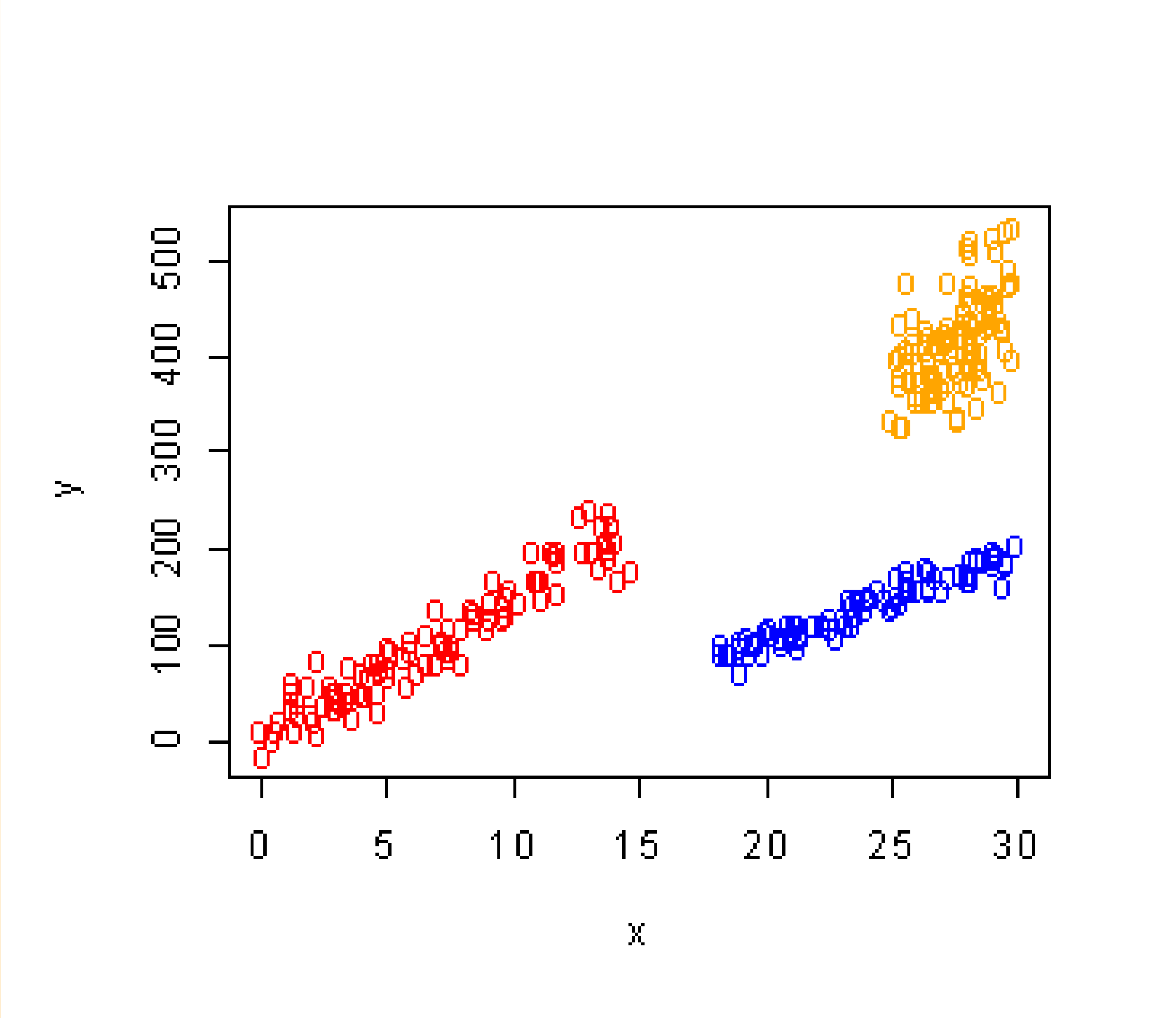
Finding 1-d Hyperplanes (Lines)



Projecting on the Lines and Finding 0-d Hyperplanes (Points)



The Final Result



The Basic LGA

Goal: to find $\mathbf{k}$ groups around hyperplanes of dimension $\mathbf{d-1}$

The Basic LGA

Goal: to find $\mathbf{k}$ groups around hyperplanes of dimension $\mathbf{d-1}$

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.

Unsupervised Learning Setup

- Clustering and linear grouping are often used in the context of **unsupervised learning**.

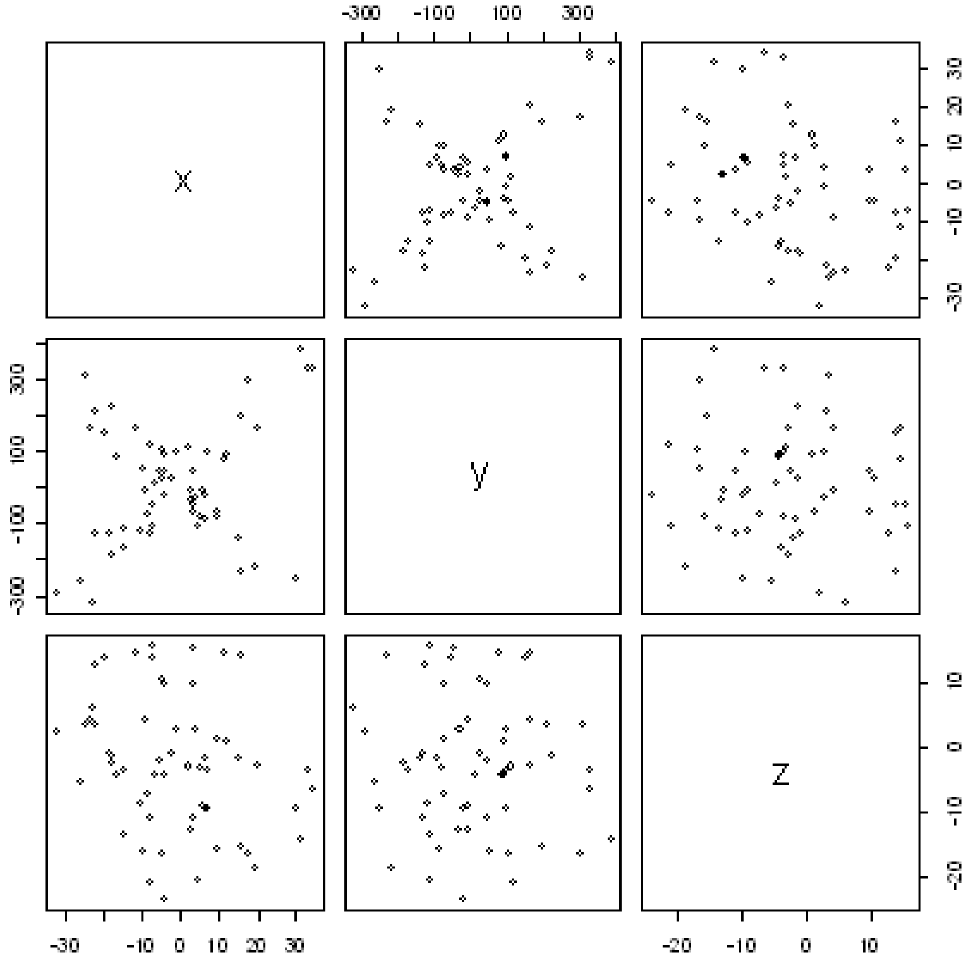
Unsupervised Learning Setup

- 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**.

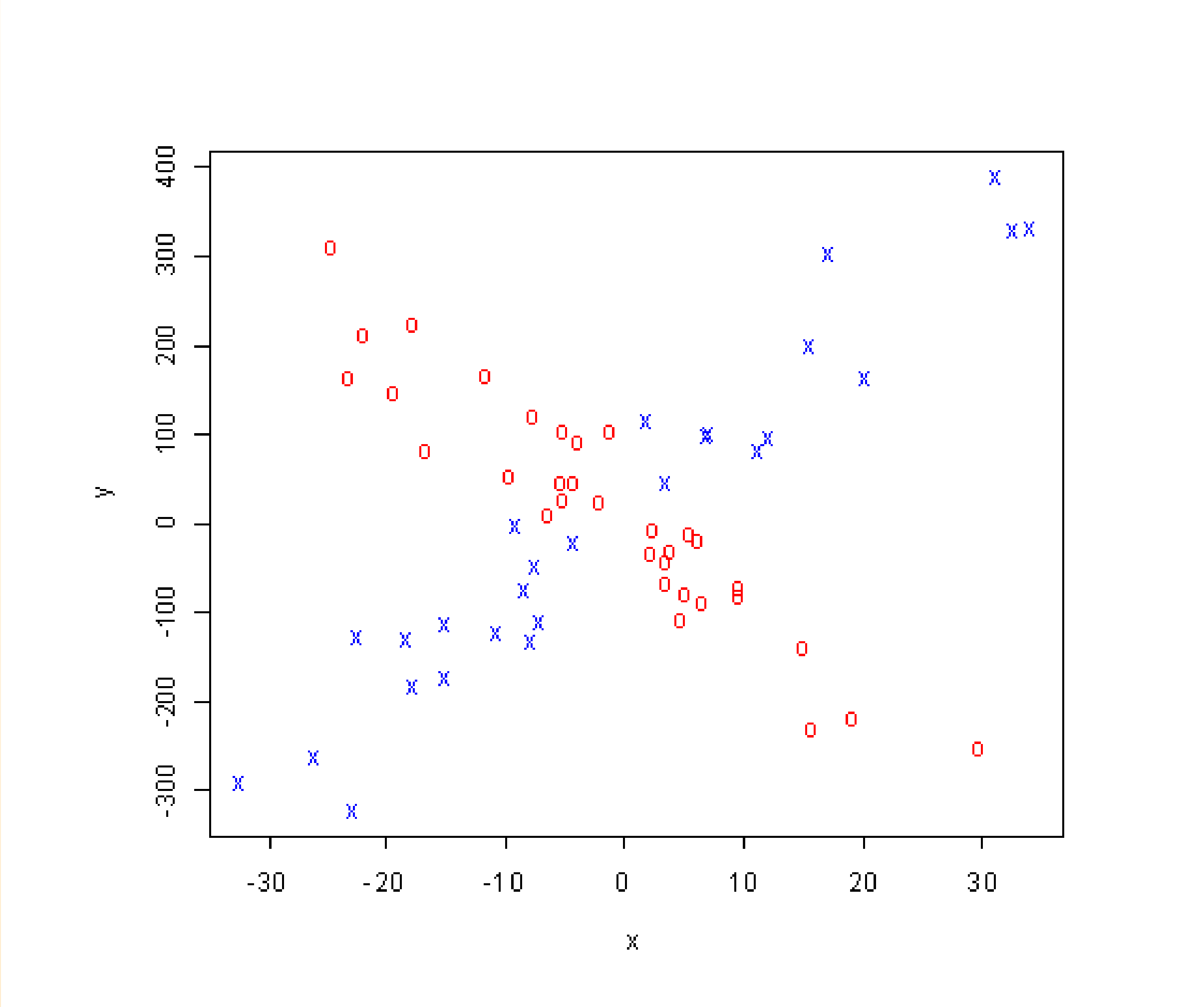
Unsupervised Learning Setup

- 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**.

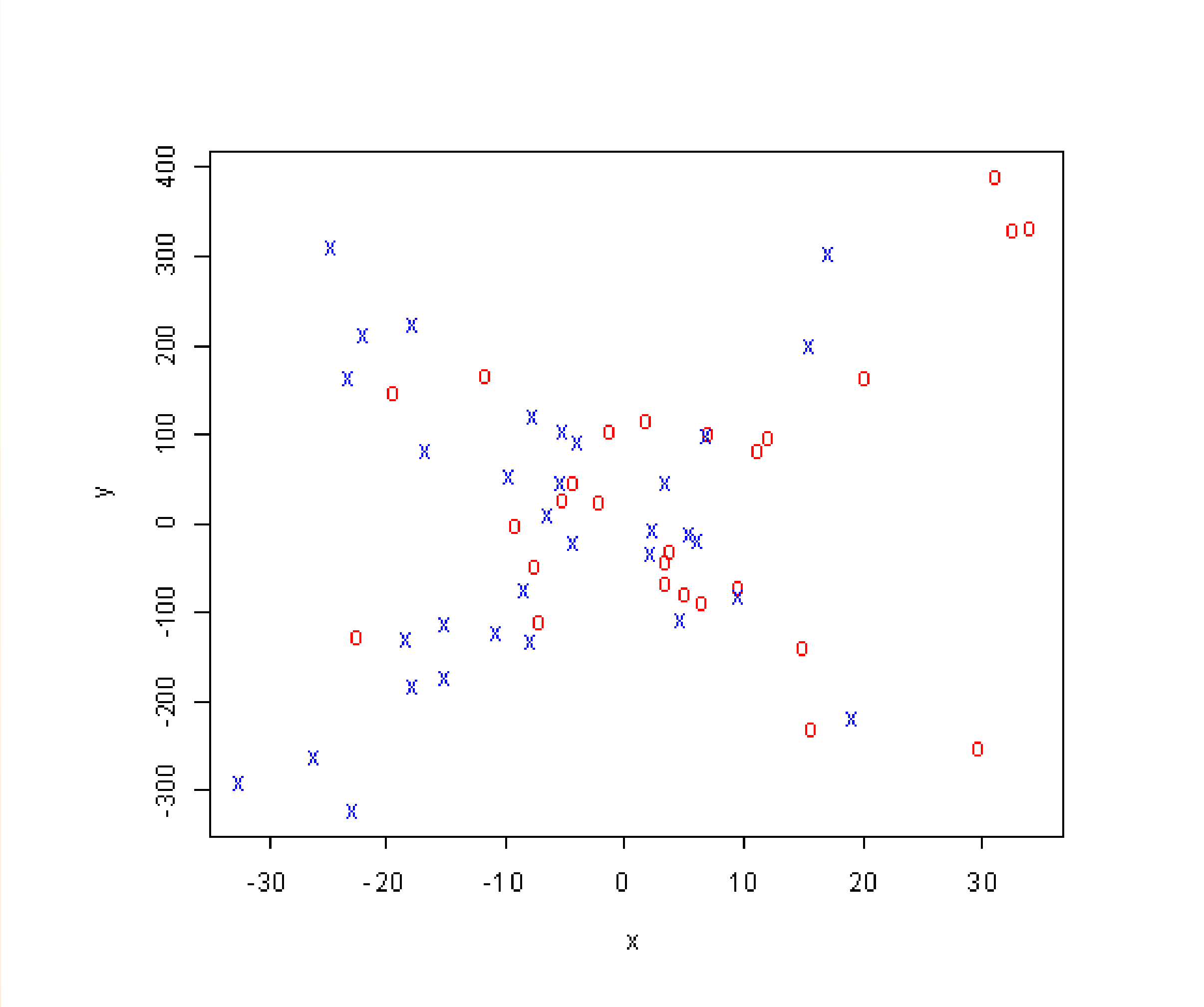
Example



Response Variable = Y



Response Variable = Z



Orthogonal Regression

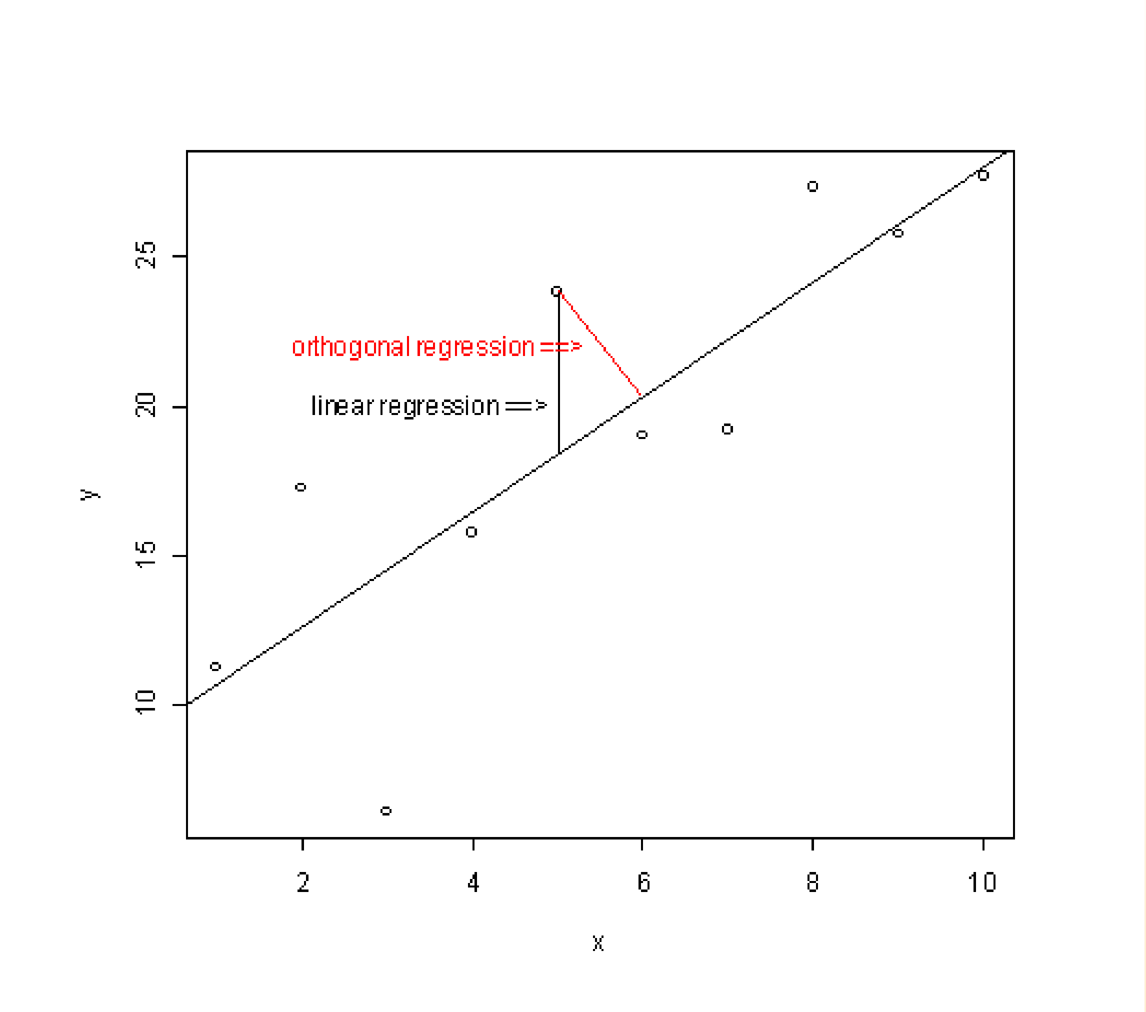
Linear Residual = Vertical distance

Orthogonal Regression

Linear Residual = Vertical distance

Orthogonal Residual = Euclidean distance

Orthogonal Residuals



Orthogonal Regression

Orthogonal Regression

Given $z_1, z_2, ..., z_n$ in 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:

$$\text{Minimize}_{\|\alpha\|=1,\beta} \sum (\alpha' z_i - \beta)^2$$

Orthogonal Regression

$$\bar{z} \;=\; \frac{1}{n}\sum z_i \qquad \text{(Sample Mean)}$$

$$S \;=\; \frac{1}{n}\sum (z_i - \bar{z})(z_i - \bar{z})' \qquad \text{(Sample Covariance)}$$

The OR estimates are:

$$\hat{\alpha} \;=\; \textbf{normalized first eigenvector of S}$$

$$\hat{\beta} \;=\; \hat{\alpha}'\bar{z}$$

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 $z_1, z_2, ..., z_n$ and the desired number k of groups

OUTPUT: The "best partition" of the dataset into k groups centered around hyperplanes of dimension $d-1$

LGA Step-by-Step

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.

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.

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

$$
\begin{aligned}
p &= \frac{\begin{pmatrix} n_1 \\ d \end{pmatrix} \begin{pmatrix} n_2 \\ d \end{pmatrix} \cdots \begin{pmatrix} n_k \\ d \end{pmatrix}}{\begin{pmatrix} n_1 + n_2 + \cdots n_k \\ dk \end{pmatrix}} \\
0.95 &= 1 - (1 - p)^m \\
m &= \frac{\log(0.05)}{\log(1 - p)}
\end{aligned}
$$

The Number of Random Starts

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

	$k=2$		$k=3$		$k=4$	
d	1:1	1:2	1:1:1	1:2:3	1:1:1:1	1:2:3:4
2	7(7)	9(10)	23(24)	42(43)	73(77)	201(206)
3	9(9)	13(13)	34(35)	82(83)	127(135)	580(586)
4	10(10)	17(17)	44(45)	145(145)	187(203)	1462(1431)
5	11(12)	52(51)	53(56)	244(239)	253(280)	3446(3207)

The Number of Random Starts

The needed number $\mathbf{m}$ of random starts depends on:

- *The the number $\mathbf{k}$ of groups,*
- *The relative size of the groups,*
- *The dimension $\mathbf{d}$ of the data.*

The Number of Random Starts

The needed number $\mathbf{m}$ of random starts depends on:

- *The the number* $\mathbf{k}$ *of groups,*
- *The relative size of the groups,*
- *The dimension* $\mathbf{d}$ *of the data.*

- $\mathbf{m}$ *doesnt depend much on the data size,* $\mathbf{n}$*.*

The Number of Groups

The Number of Groups

- The number k of groups is an input of our algorithm

The Number of Groups

➤ 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 of Groups

- 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*

Graphical Approach

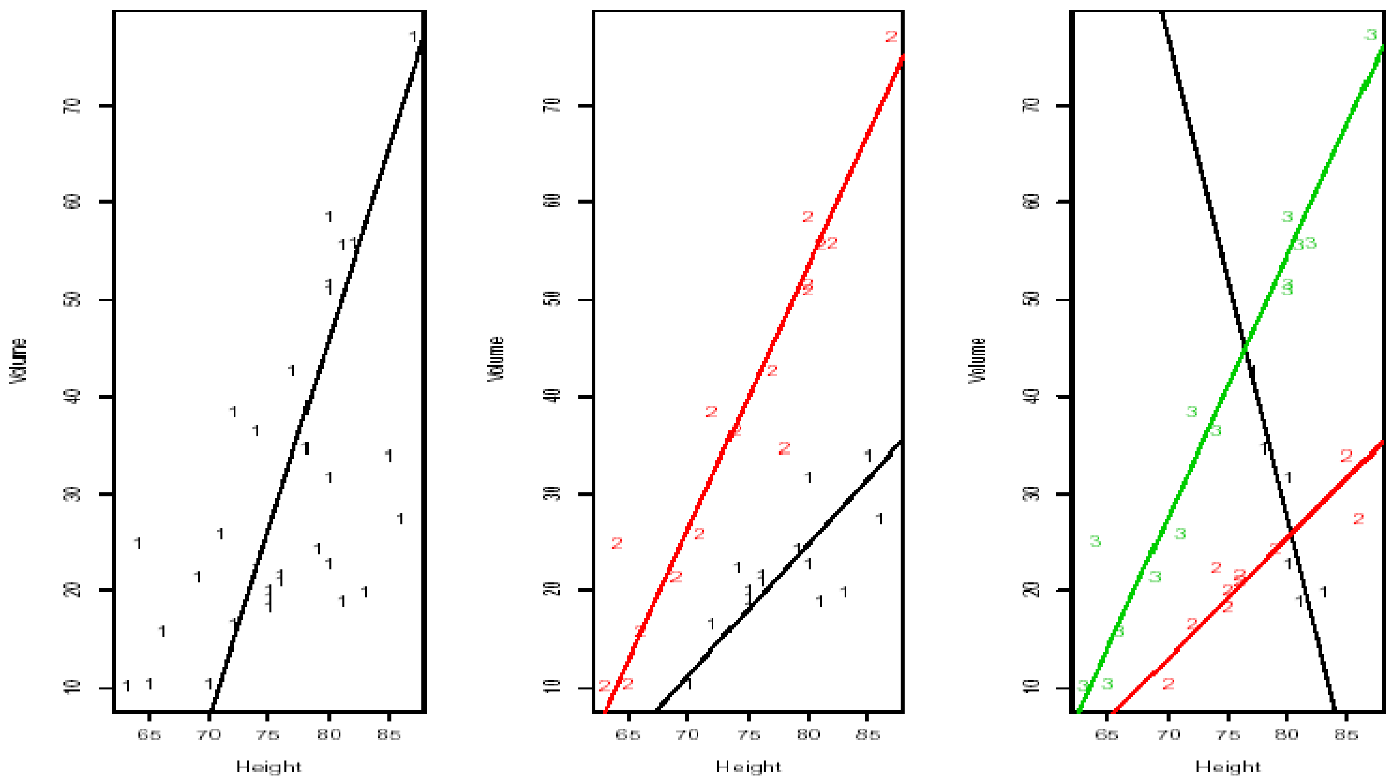


Figure 4: The height and volume of young and old trees.

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)

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

The GAP Statistic (continued)

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

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

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)

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

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

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

$$S_{k+1} = \text{Standard Deviation of } \log\left(SSR_{k+1}\,(b)\right)$$

Application to Allometry Data

Application to Allometry Data

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

Application to Allometry Data

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).

Application to Allometry Data

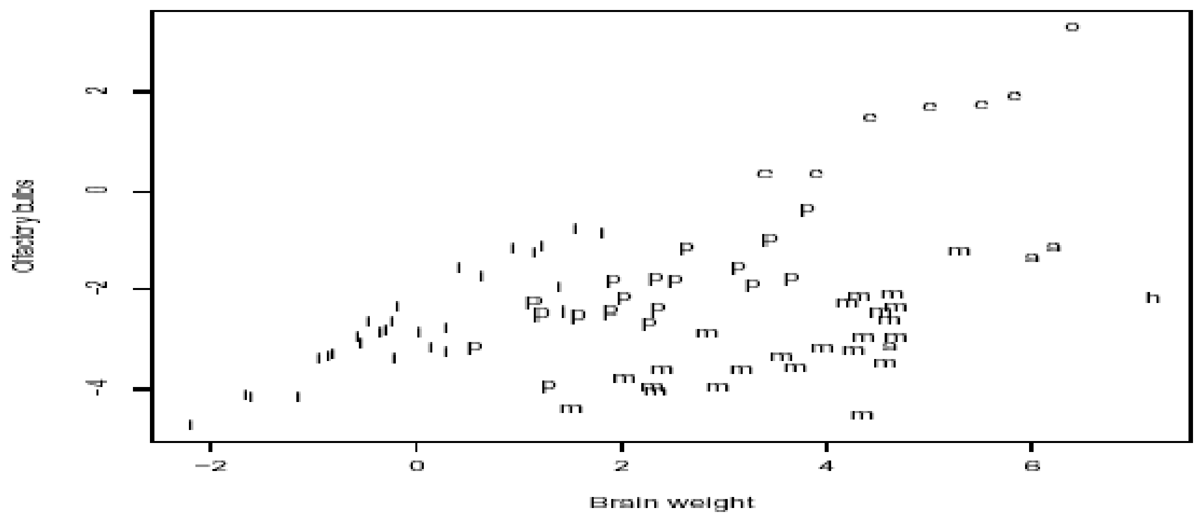
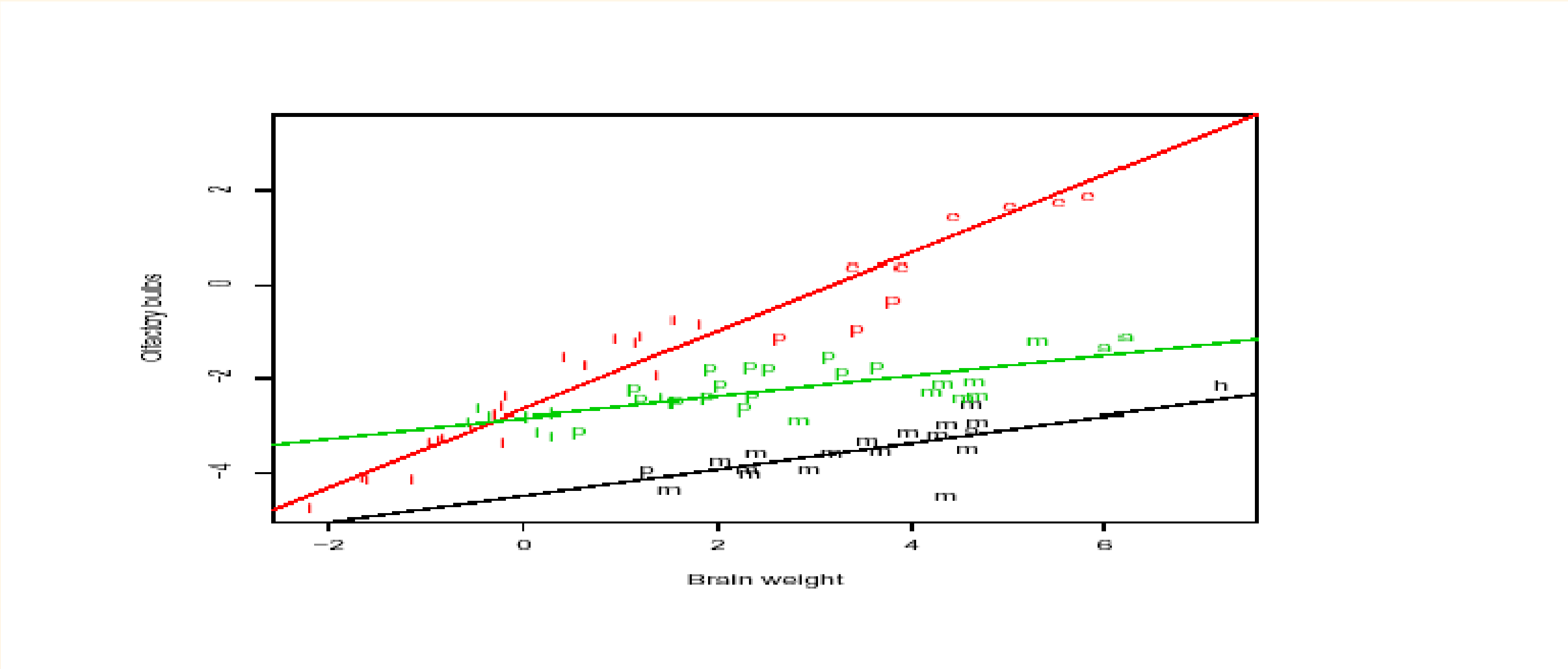


Figure 6: Logarithms of Olfactory Bulb vs. Brain Weight for some mammal species: Insectivores (i), Carnivores (c), Prosimians (p), Apes (a), Monkeys (m), Human (h) and Horse (o).

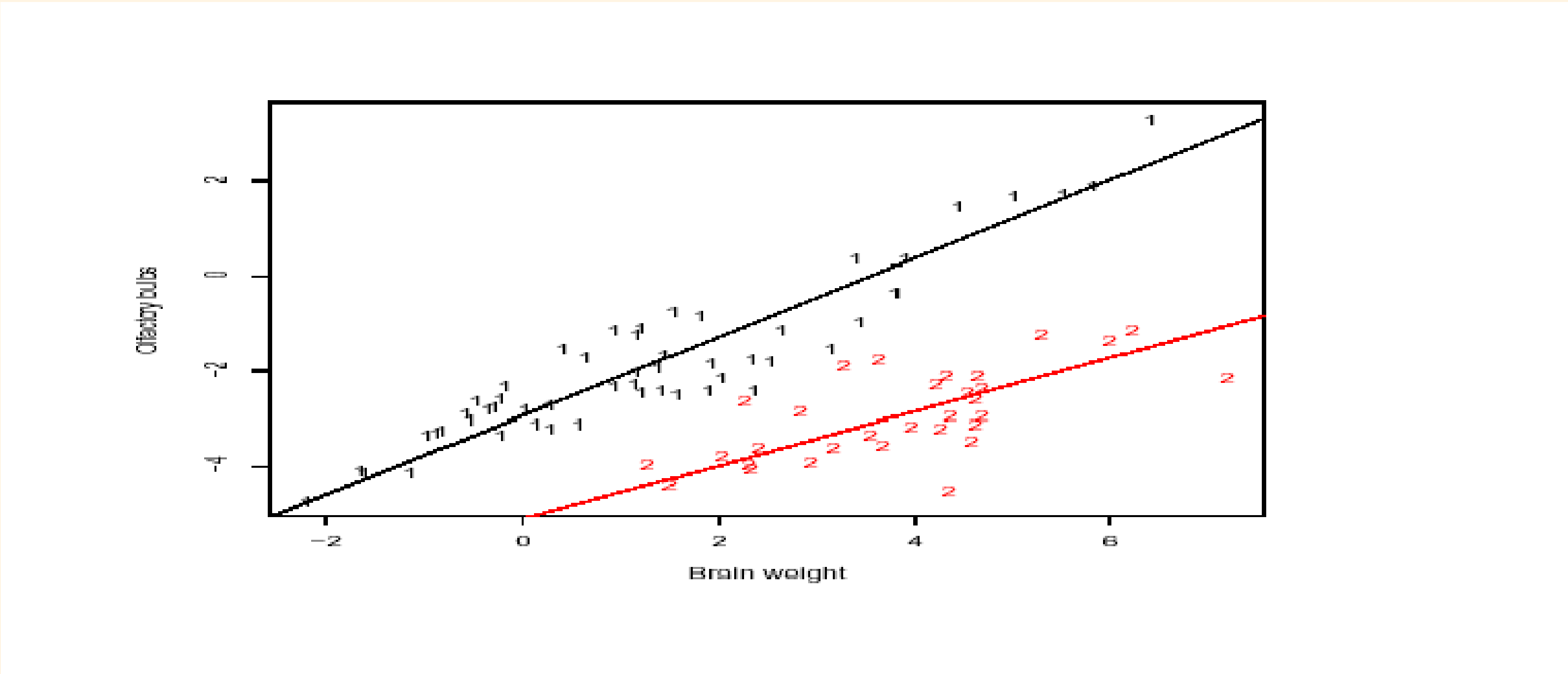
Application to Allometry Data

LGA with $k = 3$ (Dr. Jerison's hypothesis)



Application to Allometry Data

LGA with $k = 2$ (GAP result)



	Prof. Jerison	
I	insectivores, carnivores, horses,	
II	prosimians (primitive primates)	
III	anthropoids (monkeys, apes, human)	
	LGA with k=3	
I	insectivores, carnivores, horses,	red
II	prosimians and apes	green
III	monkeys and human	black
	LGA & GAP	
I	insectivores, carnivores, horses, prosimians	black
II	monkeys, apes and human	red

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
-

Variables	Description
PTS	# of Goal Scored + # of Assists
P/M	Plus/Minus Rating + team scored, - oponent team scored
PIM	Total penalty time (minutes)
PP	Total number of power-play goals scored

- We applied OR-grouping with k=3

- The results:

Group	PTS	P/M	PIM	PP
1	-0.156	0.015	0.001	0.988
2	-0.221	0.029	-0.003	0.975
3	0.113	-0.010	0.001	-0.994

Sharp Shooters - Team Players

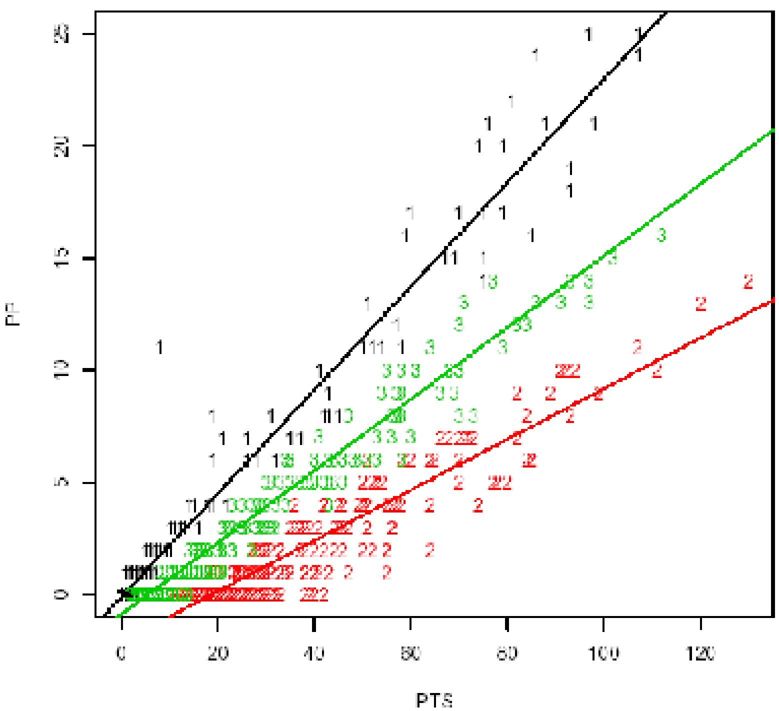


Figure 8: Plot of PP versus PTS for the NHL 94-95 competition with the three groups detected by LGA.

Application to Genomics

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).

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.

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 ATGTTA.**

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 ATGTTA.**
- 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 ATGTTA.**
- 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

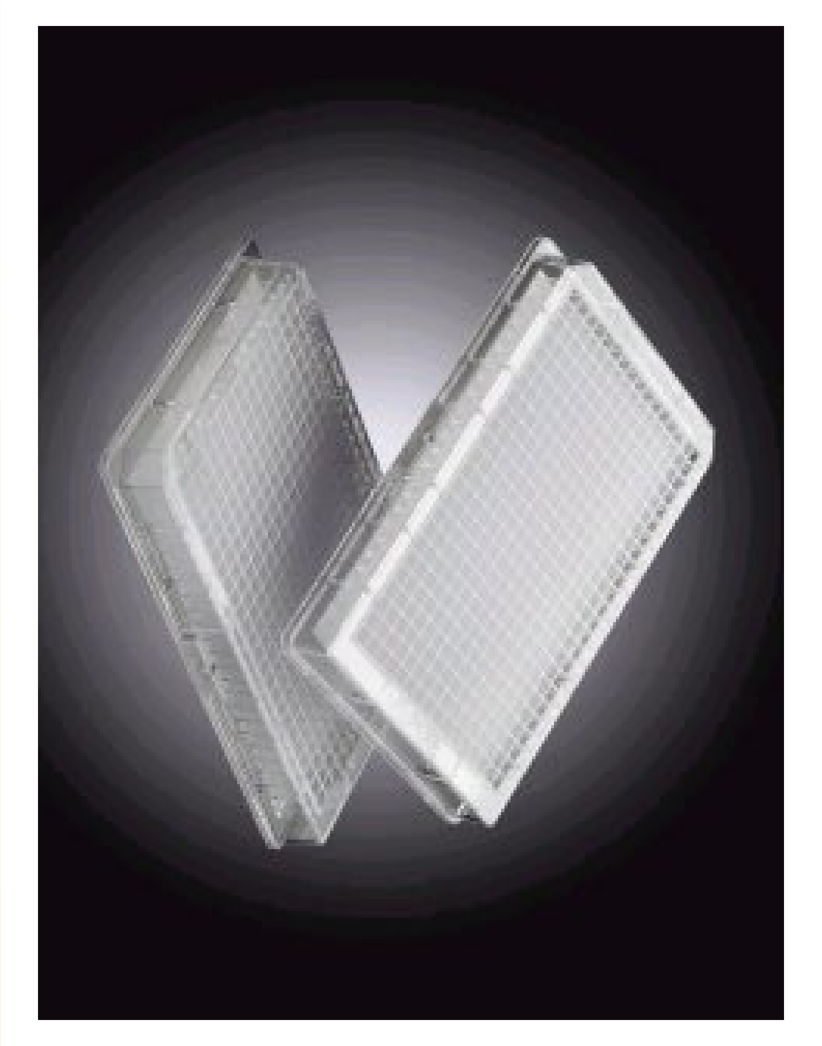
The TaqMan assay is a popular high-throughput genotyping technology (Livak et al. 1995)

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

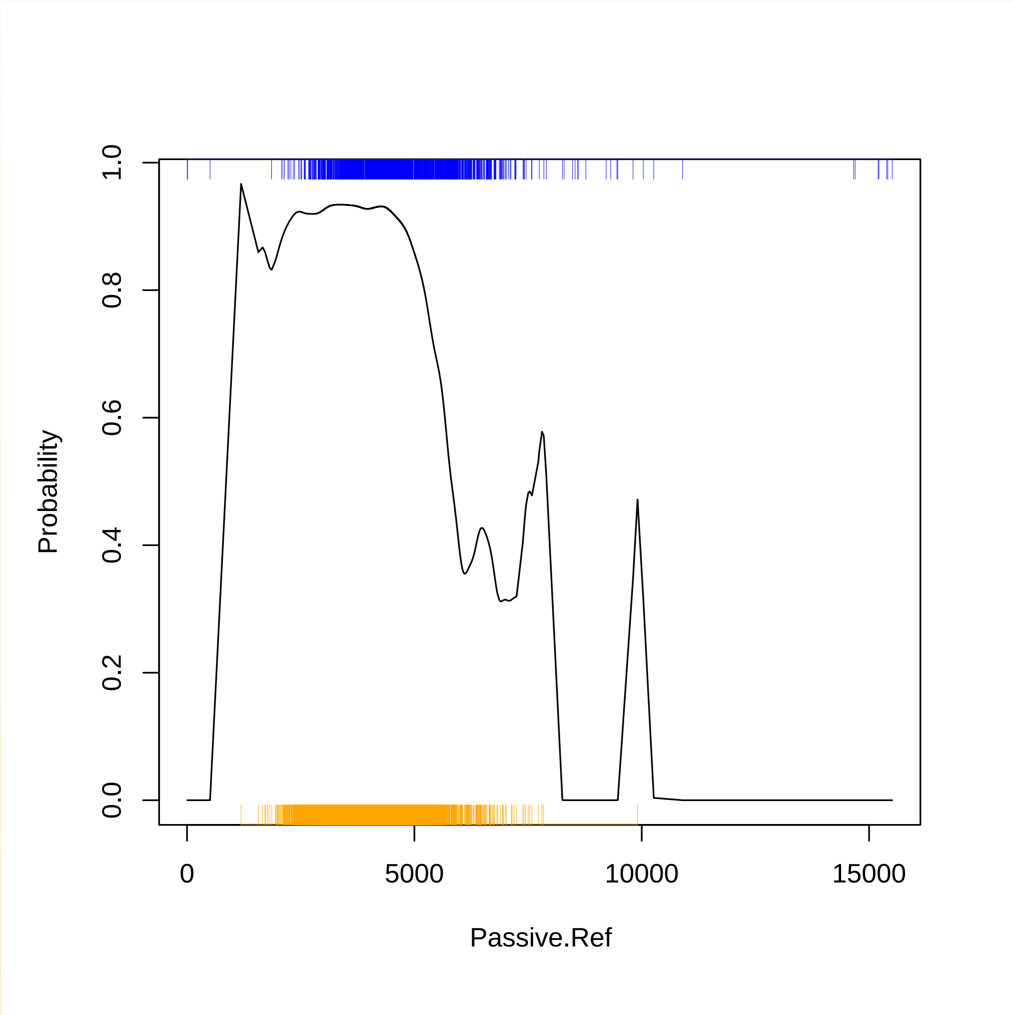


Plate3 - Raw Data

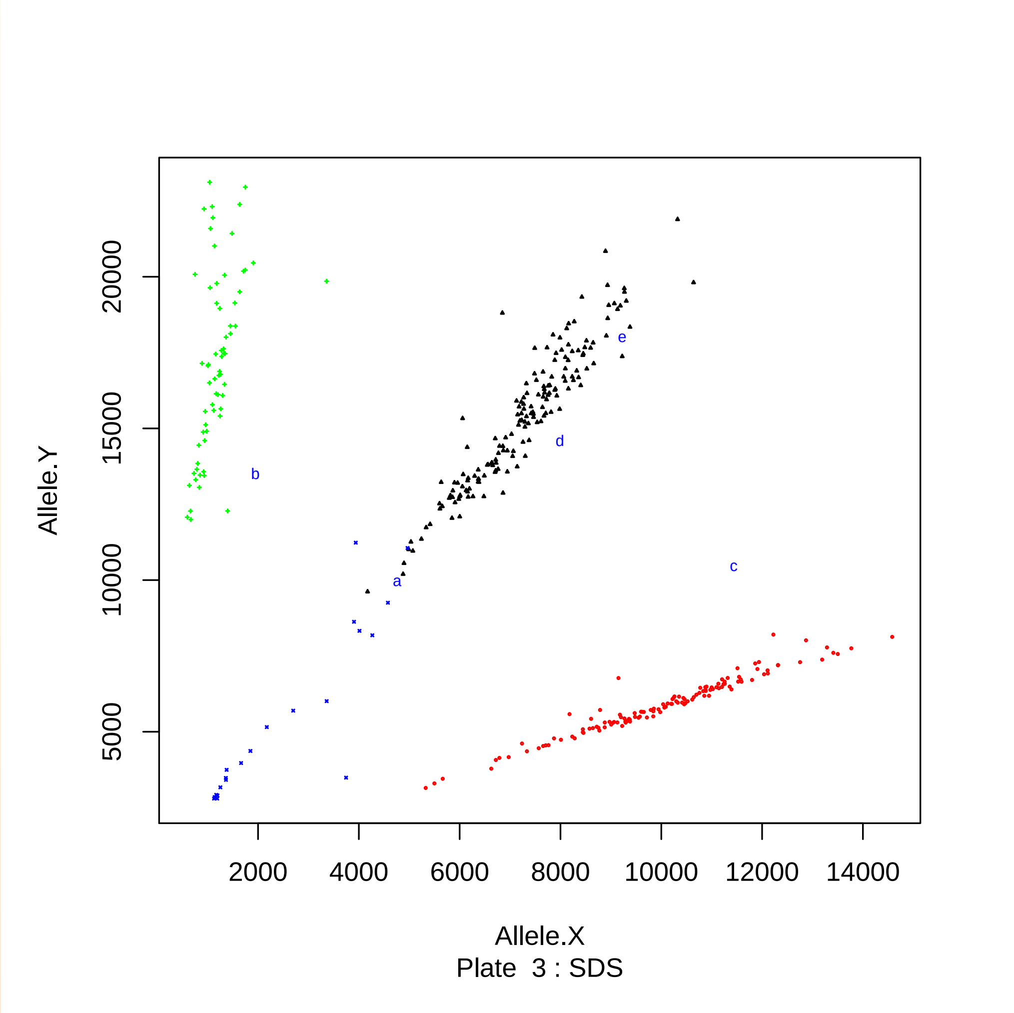


Plate3 - ROX Normalized Data

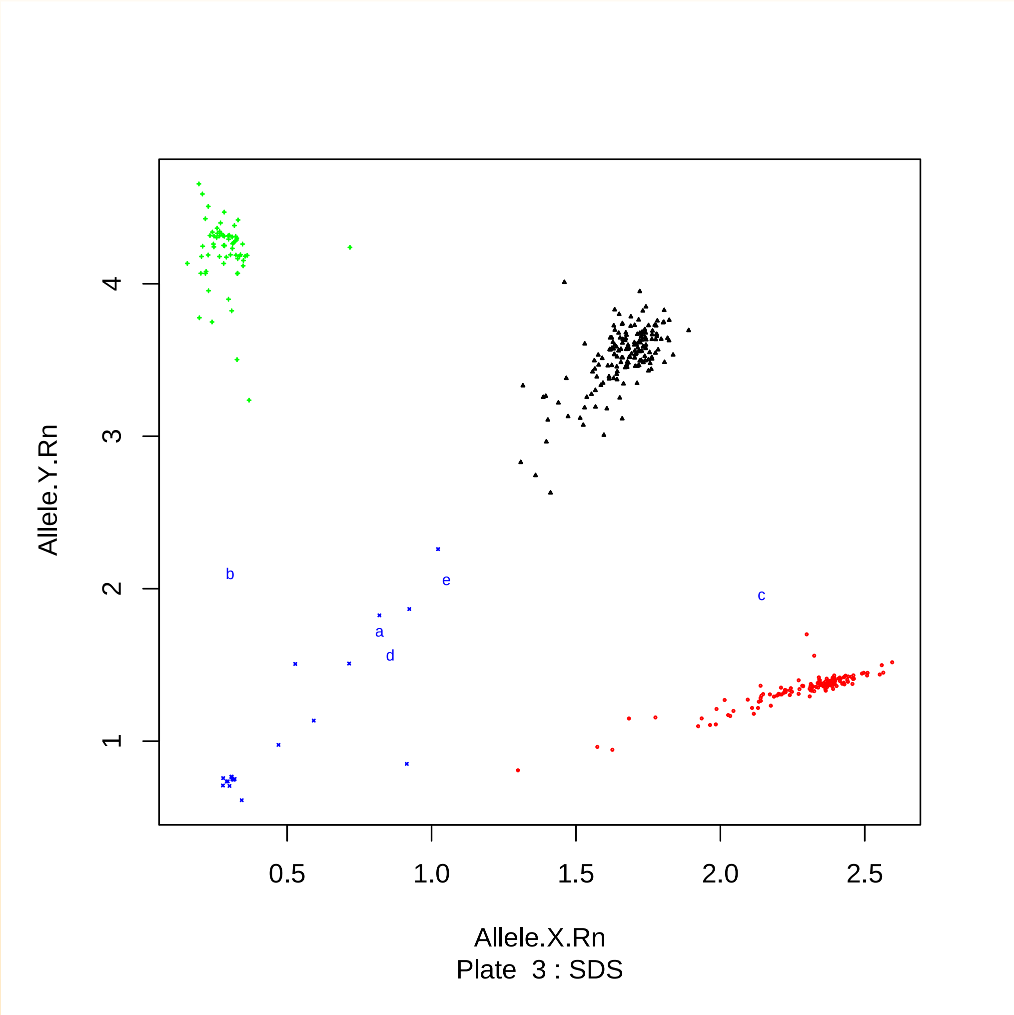
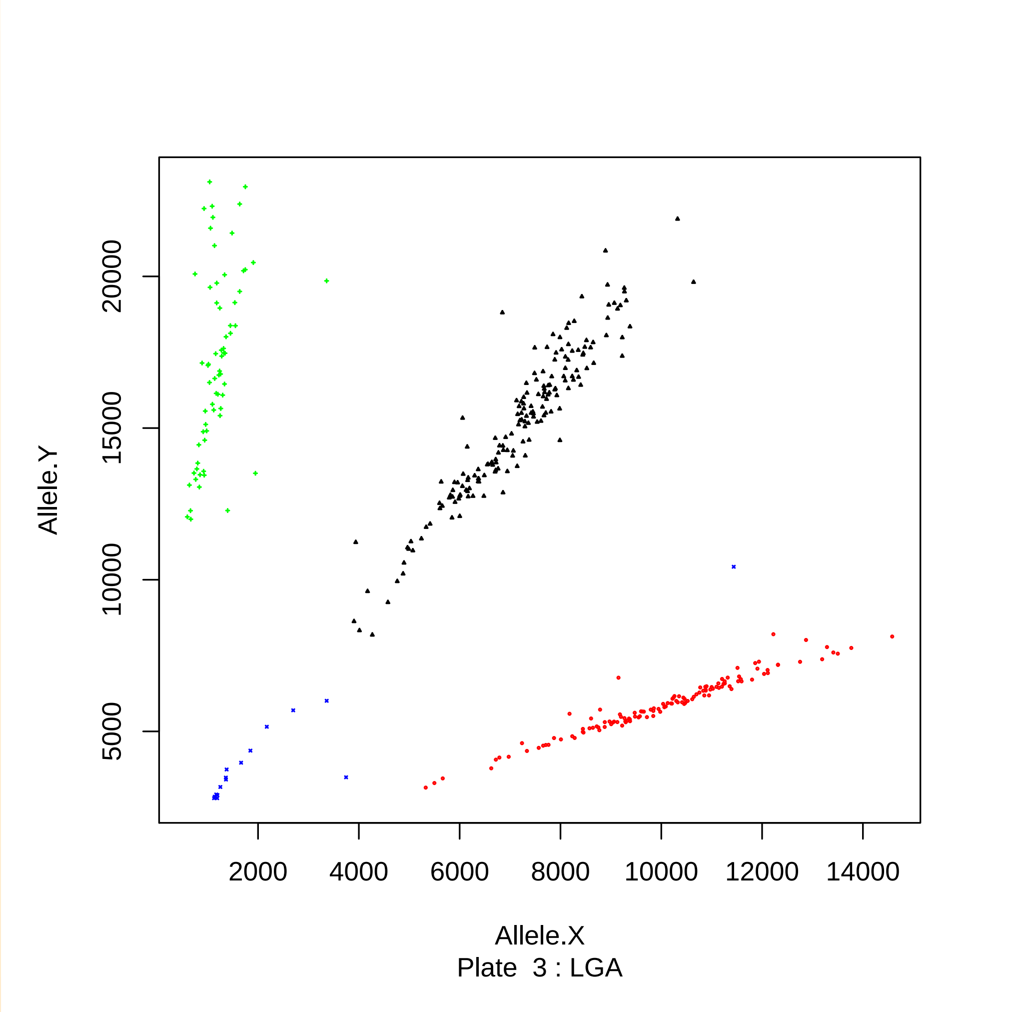


Plate3 - Raw Data - LGA Grouping



Summary

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 “nuisanse” variables*

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)).

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 "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**)

➤ 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 "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**)
- 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 “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**)
- 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 “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**)
- 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!