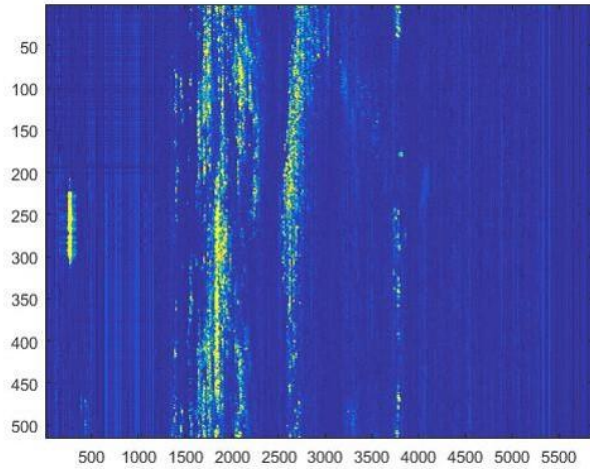




# DOBIN – DIMENSION REDUCTION FOR OUTLIER DETECTION

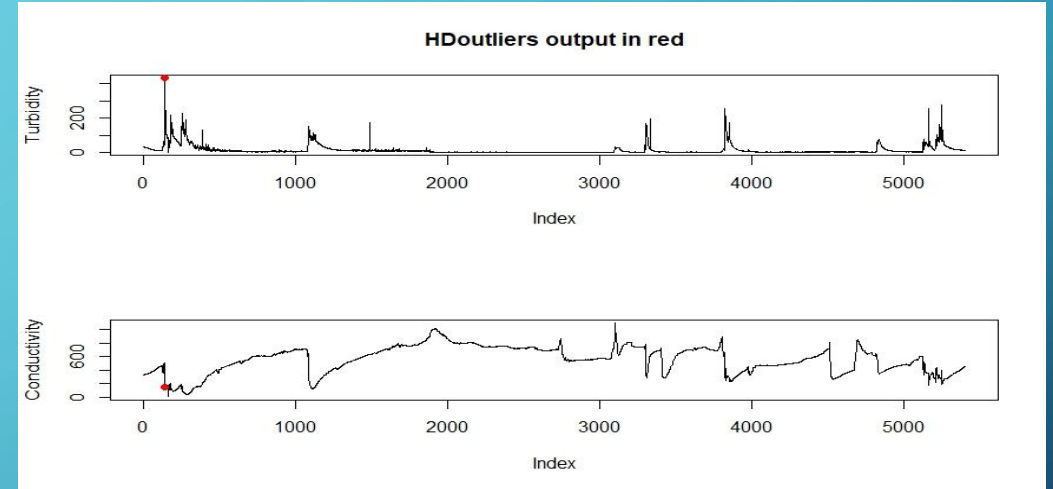
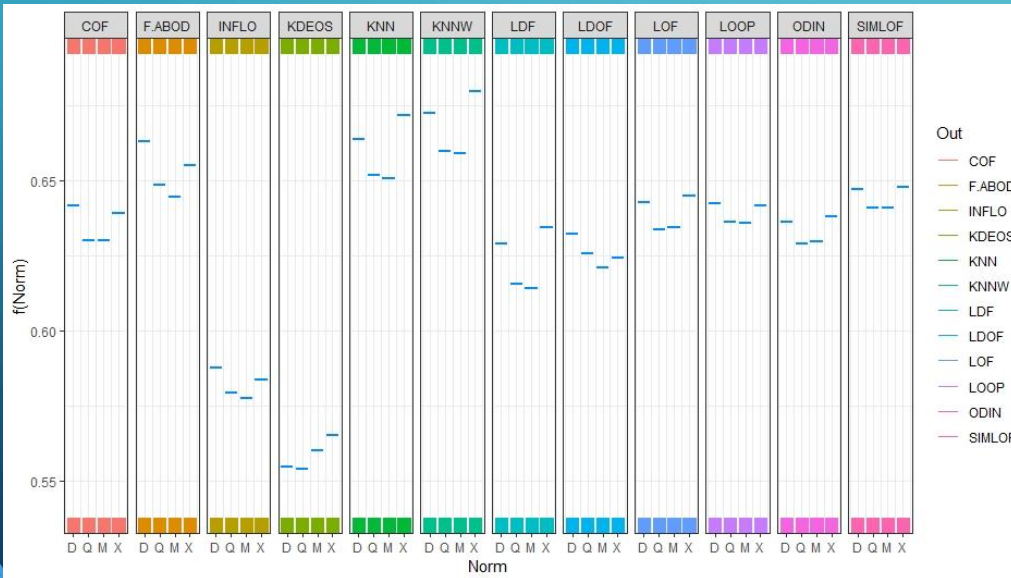
JOINT WORK WITH ROB HYNDMAN

DOB IN : TO INFORM AGAINST, SPECIALLY TO THE POLICE




Water quality

## Intrusion detection



## Meta-learning study on outlier detection

A decorative graphic on the left side of the slide, consisting of a network of light blue lines and small circles, resembling a circuit board or a neural network diagram.

BUT, NOT MUCH FOCUS  
ON DIMENSION  
REDUCTION METHODS  
FOR OUTLIER DETECTION

A decorative graphic on the left side of the slide, consisting of a network of light blue lines and small circles, resembling a circuit board or a neural network diagram.

HIGH DIMENSIONAL OUTLIERS  
MAY NOT BE OUTLIERS IN  
LOW DIMENSIONAL  
PROJECTIONS!

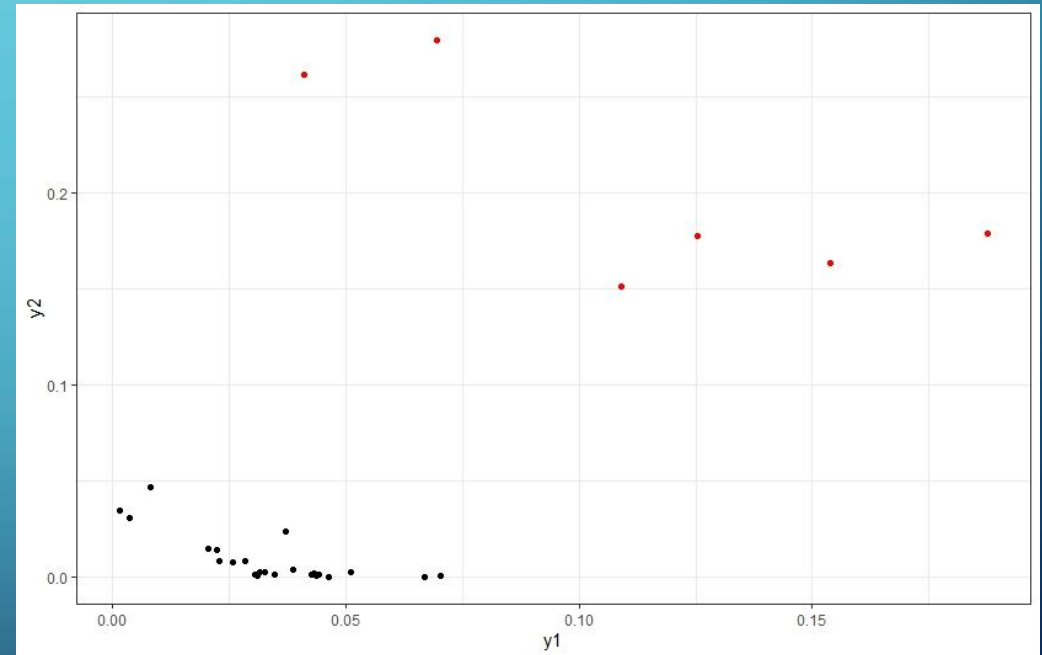
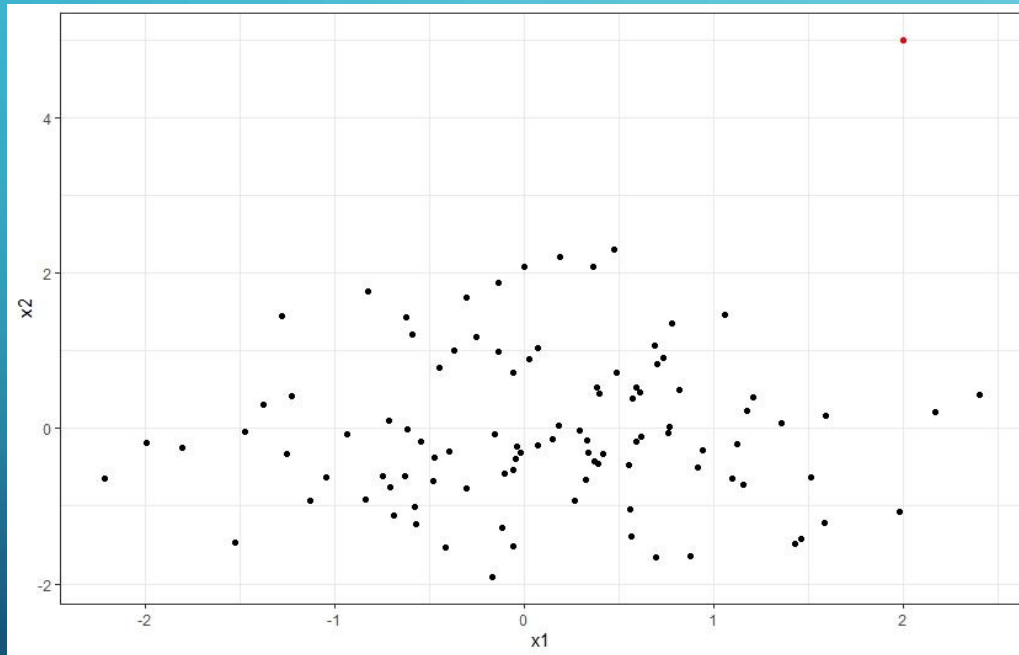
# DOBIN

- Distance-based Outlier Basls using Neighbours
- dob in : To inform against, specially to the police
- Finds a set of basis vectors tailored for outlier detection
  - First basis vector – in the direction of highest outlyingness
  - Second basis vector – in the next highest direction of outlyingness
  - And so on ...

# THE $Y$ SPACE

- For each point  $x$  in the original space, find  $k$  nearest neighbours  $z_1, z_2, \dots, z_k$
- $x = (x_1, x_2, \dots, x_n)$  and  $z_i = (z_{i1}, z_{i2}, \dots, z_{in})$
- $y = [(x_1 - z_{i1})^2, \dots, (x_n - z_{in})^2]$
- $y = (x - z_1)^{E2}$ ,  $E2$  denotes element wise squares
- So, the  $Y$  space is an **inter-distance space**
- If  $x \in R^n$ , then  $y \in R^{n+}$
- Remove points with small inter-distances,  $y_1 + y_2 + \dots + y_l < M$  for some  $M$

# EXAMPLE: $Y$ SPACE



# COMPARISON

## $X$ SPACE

- A point is an input observation
- Distance is not a linear combination of  $x$  coordinates

## $Y$ SPACE

- A point gives the inter-distance-vector between two neighbouring points in the  $X$  space
- Distance in  $X$  is a linear combination of  $y$  coordinates



# DISTANCES

- Distance between two points  $dist(x_i, x_j)^2 = (x_i - x_j)^T (x_i - x_j)$
- More generally  $dist(x_i, x_j)^2 = (x_i - x_j)^T S (x_i - x_j)$ , where  $S$  is a symmetric, positive definite matrix
- $dist(x_i, x_j)^2 = \langle \eta, (x_i - x_j)^{E2} \rangle$ , element wise squares
- Using  $Y$  space
- $dist(x_i, x_j)^2 = \langle \eta, y_l \rangle$
- $\sum dist(x_i, x_j)^2 = \sum \langle \eta, y_l \rangle$

# MAXIMISING DISTANCE

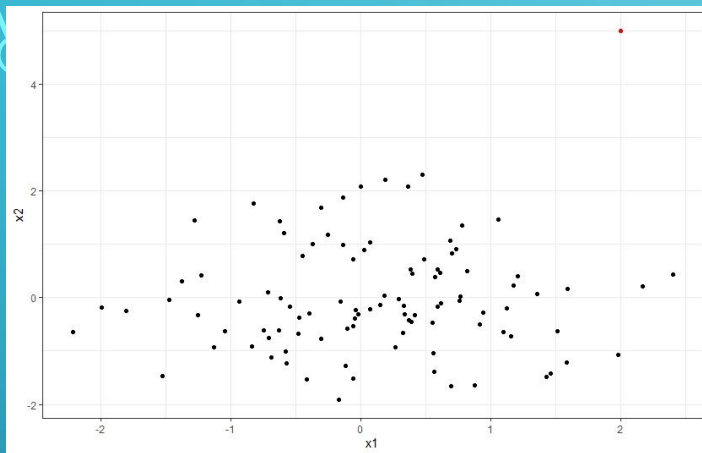
- Want to maximise sum of distances between points
- $\max \sum dist(x_i, x_j)^2$
- Our problem: Find  $\eta$  such that

$$\max \sum \langle \eta, y_l \rangle$$

$$\text{Subject to } \|\eta\| = 1$$

- Because  $dist(x_i, x_j)^2 = \langle \eta, y_l \rangle$

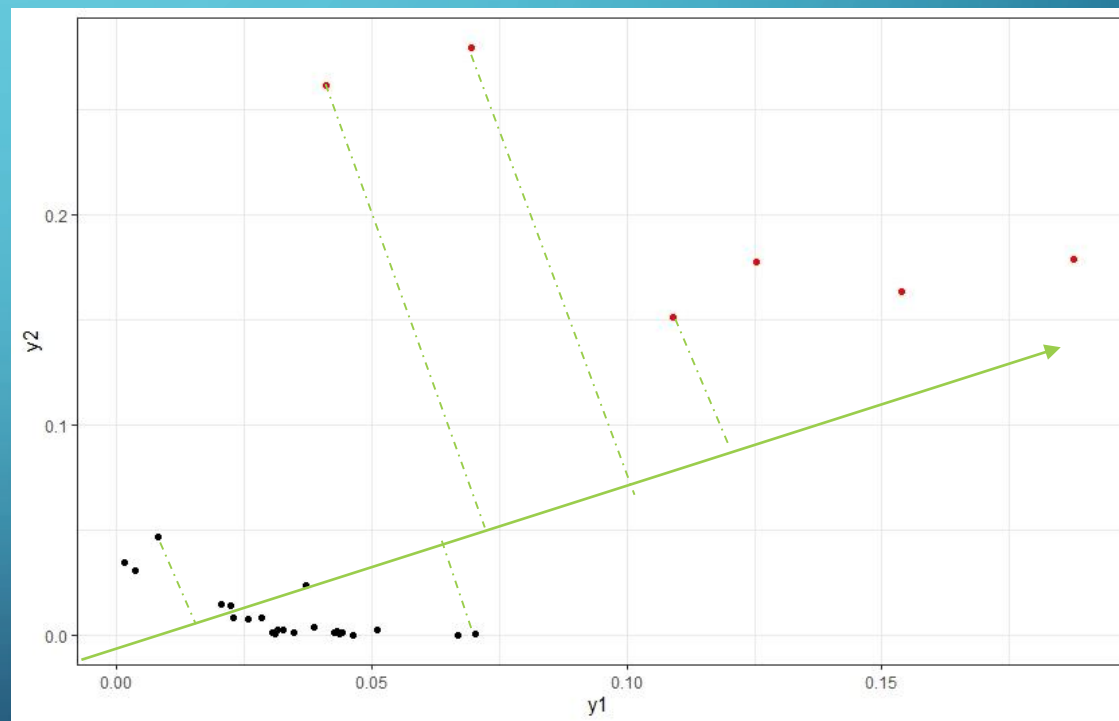
$X$  Space



$\max \sum \langle \eta, y_l \rangle = \text{sum of all projections in } \eta \text{ direction}$

# GEOMETRICALLY

$Y$  Space



# SOLVING IT

- Using Lagrange multipliers

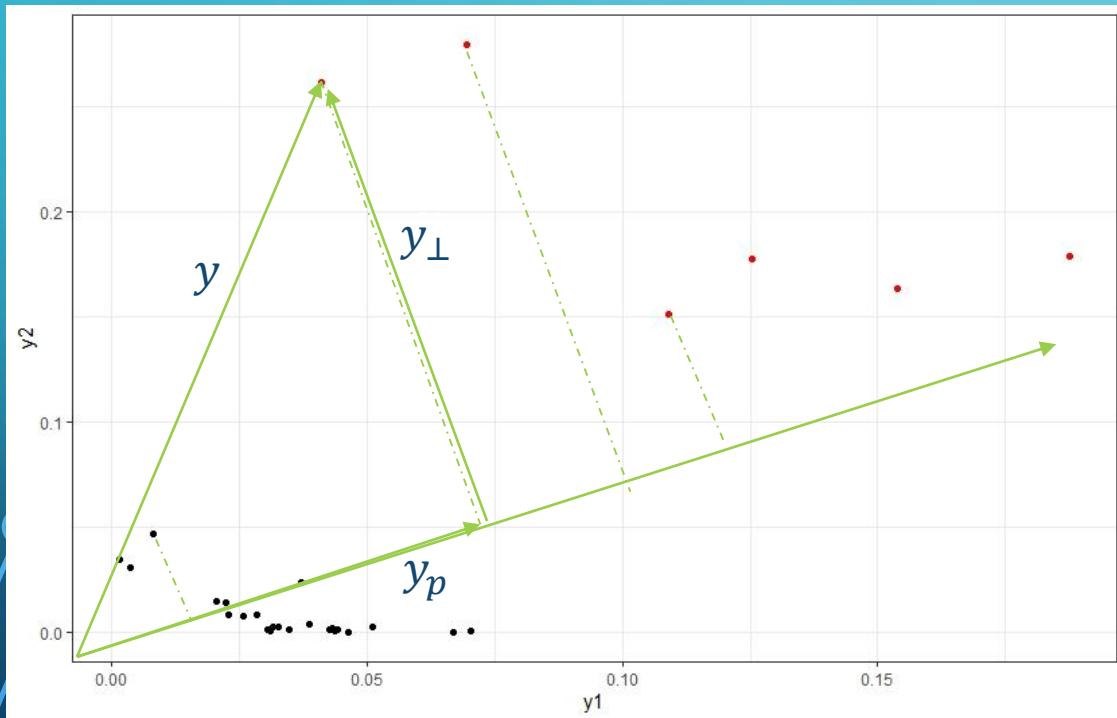
$$\eta = \frac{\sum y_l}{\|\sum y_l\|}$$

A decorative graphic on the left side of the slide, consisting of a network of light blue lines and small circles, resembling a circuit board or a neural network diagram.

WE FOUND THE DIRECTION THAT MAXIMIZES  
DISTANCES BETWEEN POINTS.

BUT HOW DO WE GET A BASIS?

# VECTORS PERPENDICULAR TO $\eta$



Let  $\eta_1 = \eta$

$$y = y_p + y_\perp$$

But  $y_p = \langle y, \eta_1 \rangle \eta_1$

$$y_\perp = y - \langle y, \eta_1 \rangle \eta_1$$

All  $y_\perp$  are perpendicular to  $\eta_1$

## SECOND VECTOR $\eta_2$

All  $y_{\perp}$  are perpendicular to  $\eta_1$

$$\eta_2 = \frac{\sum y_{\perp}}{\|\sum y_{\perp}\|}$$

So  $\eta_2$  is perpendicular to  $\eta_1$

## A BASIS

- Continue this way

$$y_{\perp} = y - \langle y, \eta_1 \rangle \eta_1 - \langle y, \eta_2 \rangle \eta_2$$

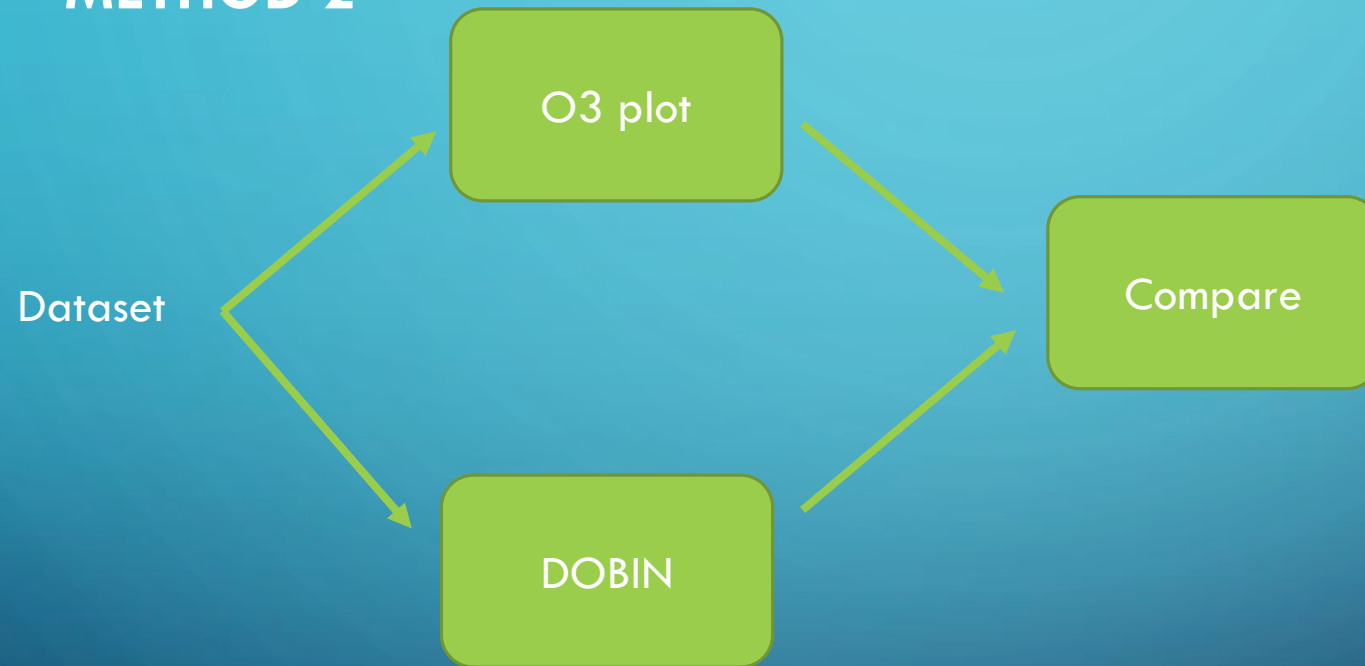
$$\eta_3 = \frac{\Sigma y_{\perp}}{\|\Sigma y_{\perp}\|}$$

Basis  $(\eta_1, \eta_2, \dots, \eta_n)$

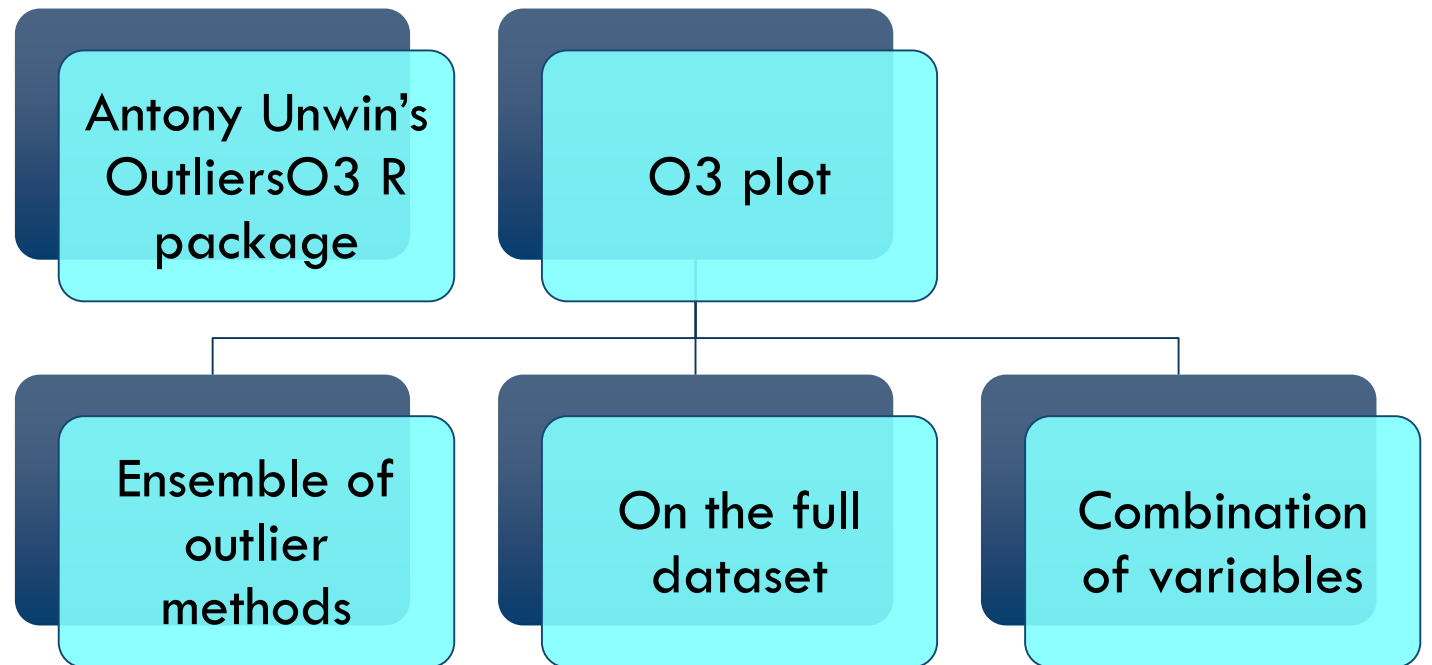


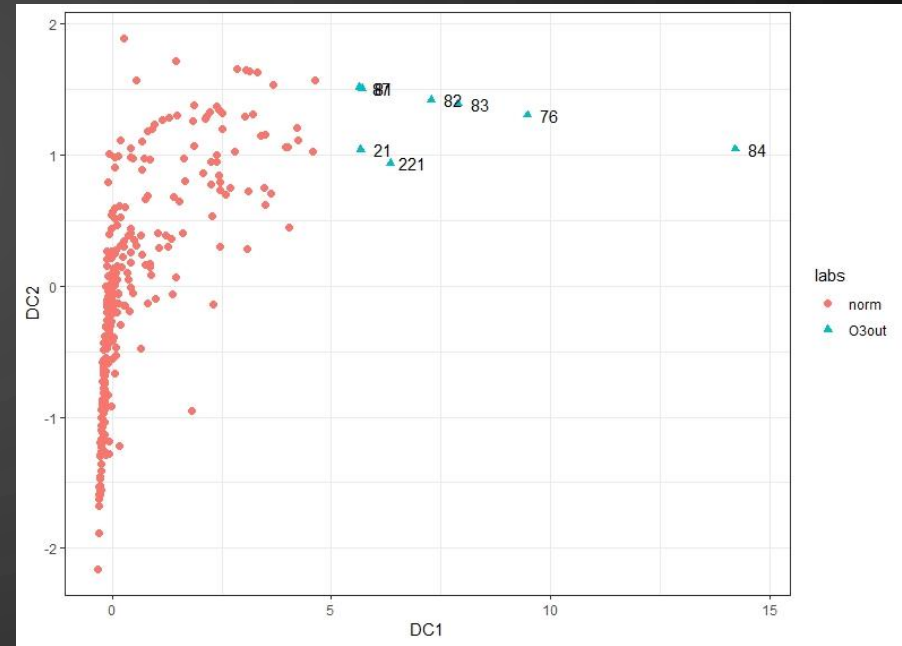
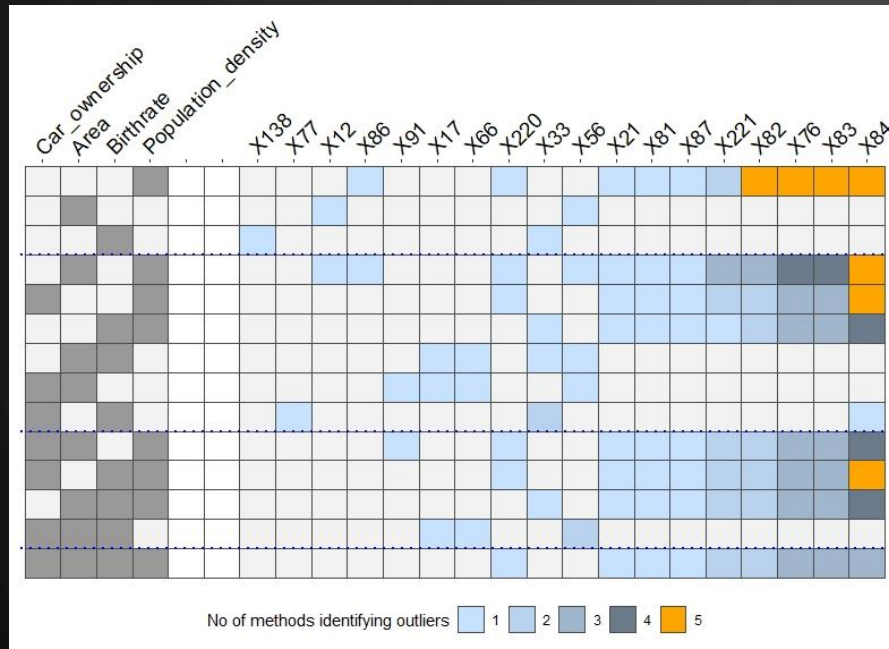
# HOW DO WE TEST DOBIN?

## METHOD 2



# O3 PLOTS



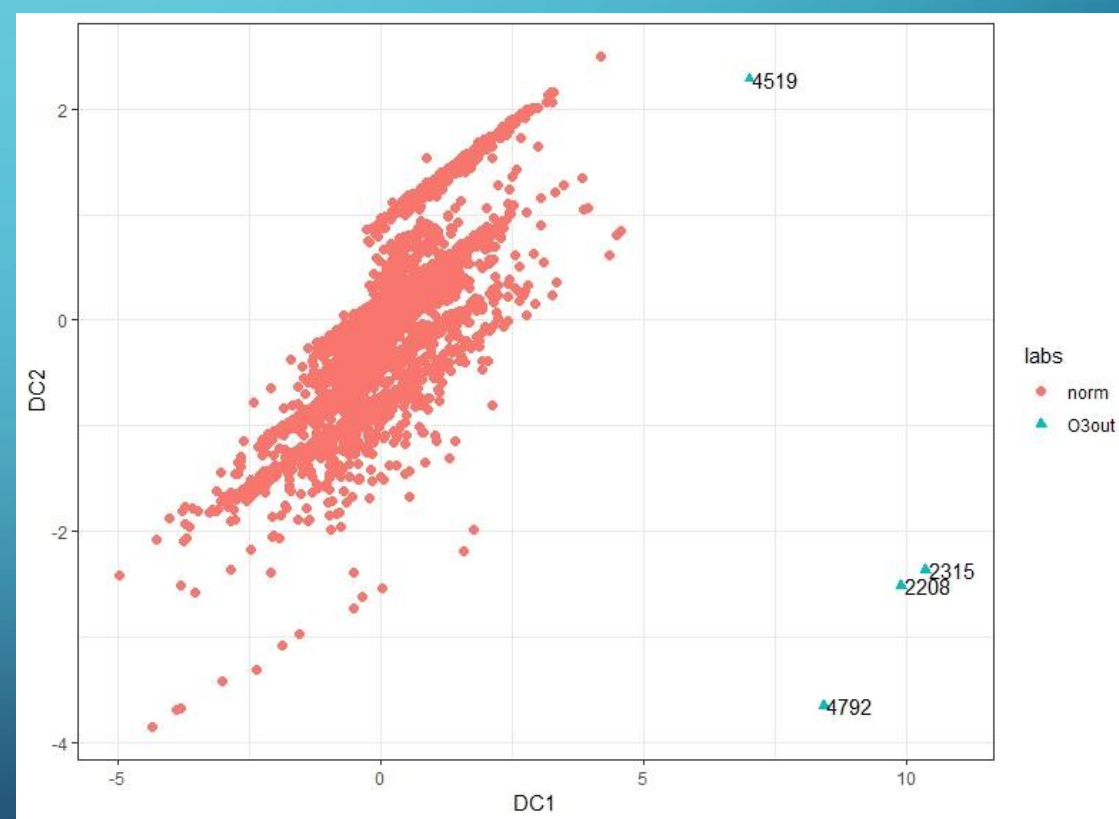
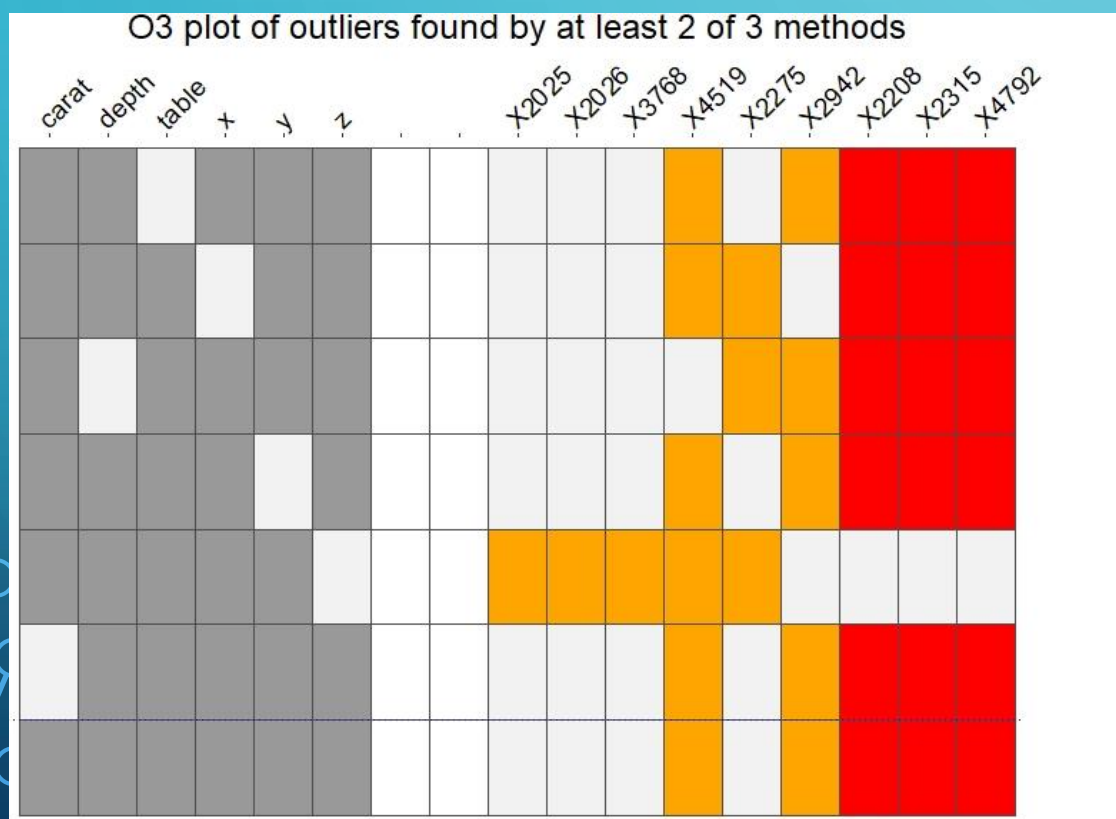


# ELECTION2005 DATASET

R package *mbgraphic*

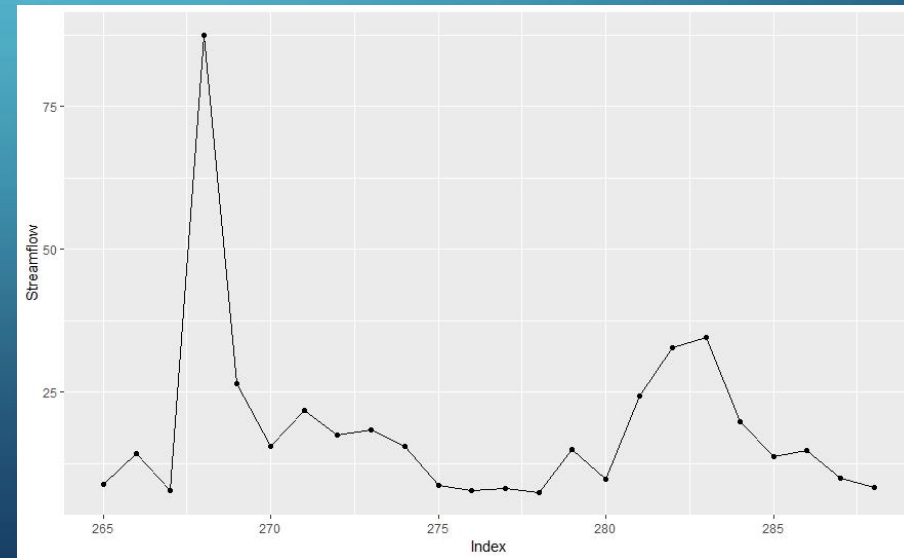
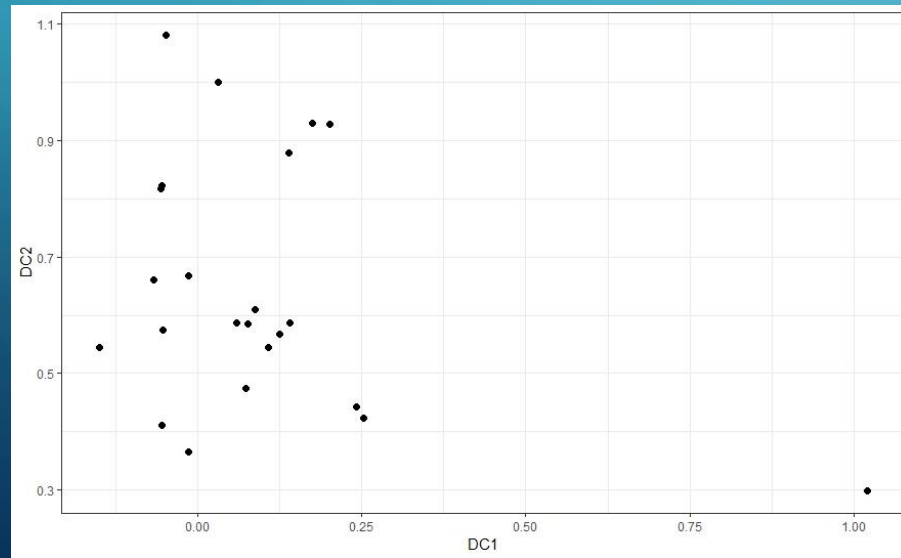
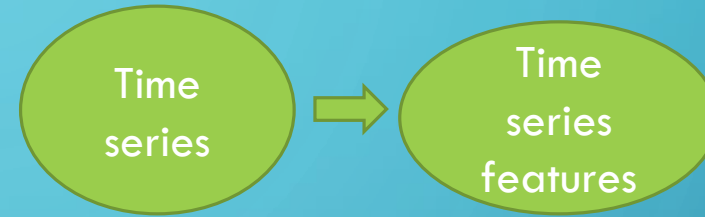
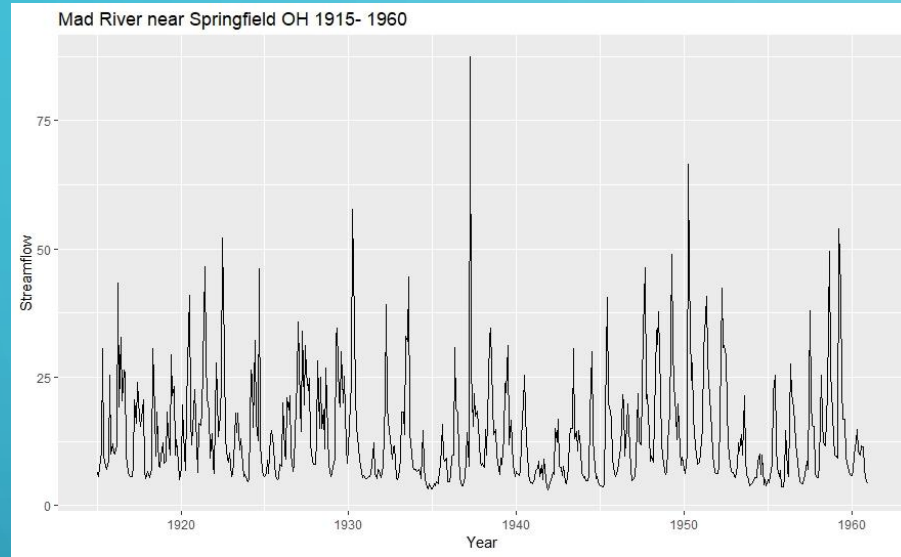
# DIAMONDS DATASET

R package `ggplot2`



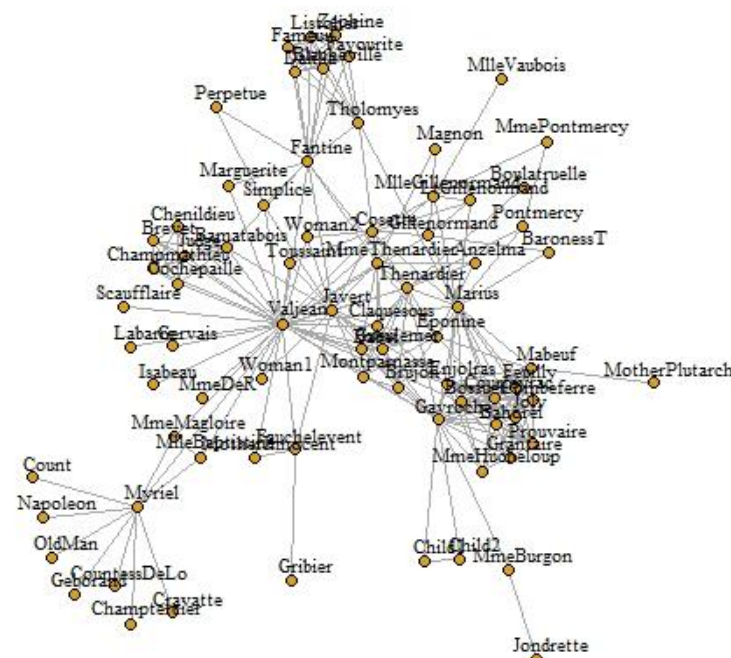
# TIME SERIES DATA - MAD RIVER NEAR SPRINGFIELD, OH

R package *tsdl*



# GRAPH DATA LESMIS DATASET

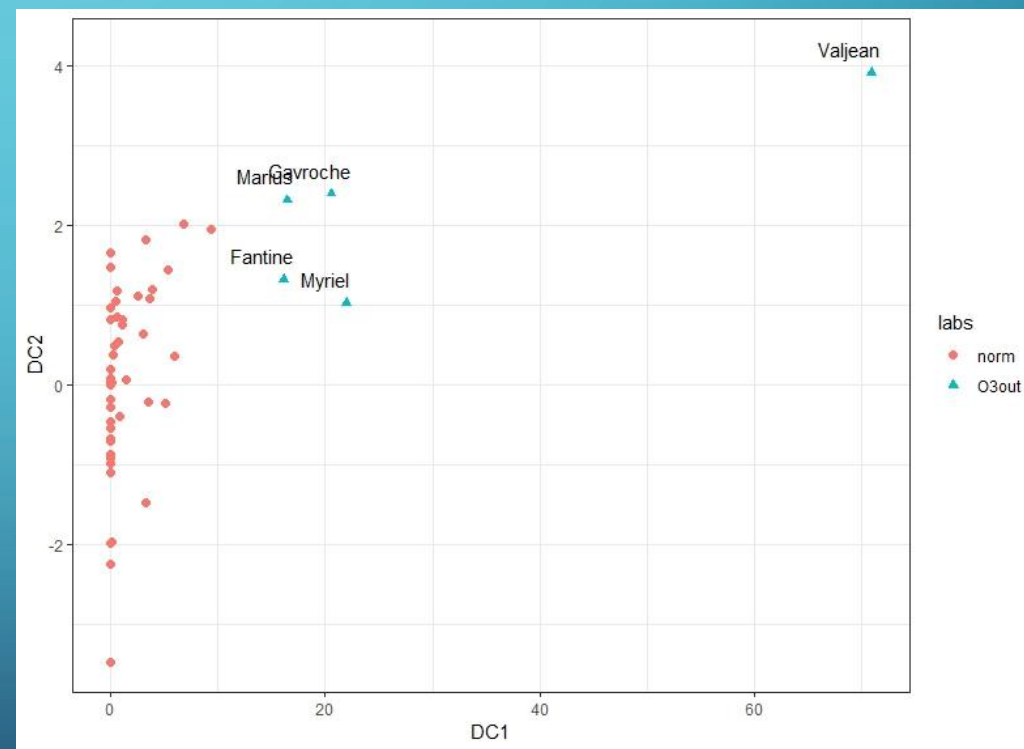
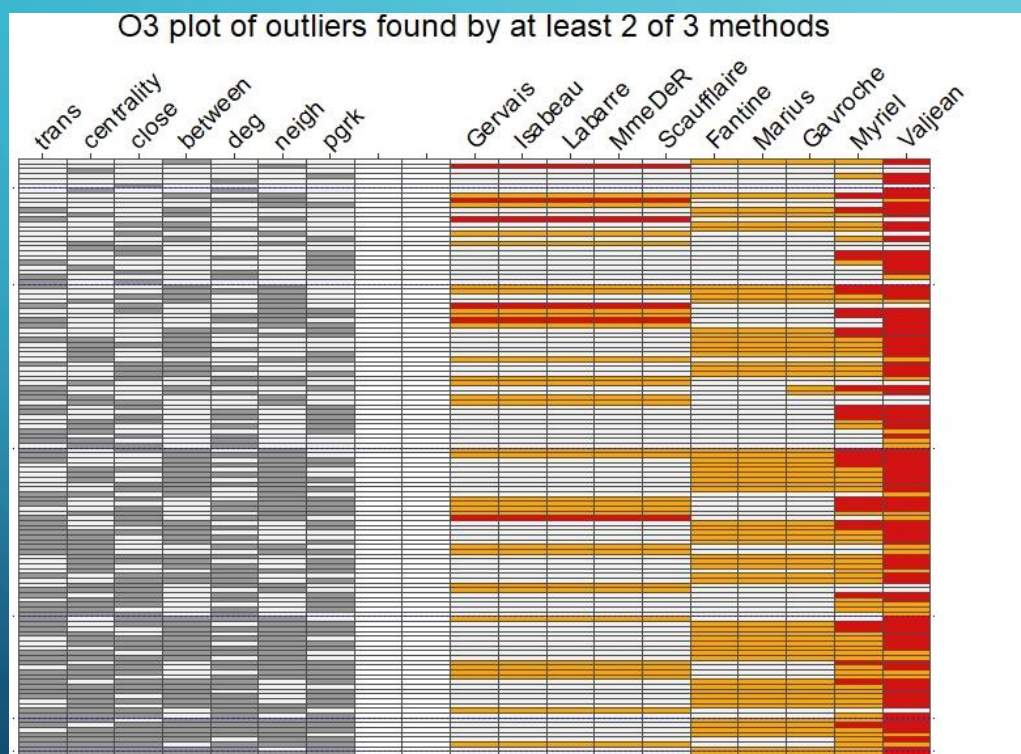
- R package SOMbrero
- Character co-appearance network of characters in *Les Misérables*
- It is a graph object
- We compute
  - Centrality, transitivity, closeness, betweenness, degree, average nearest neighbour degree, page rank





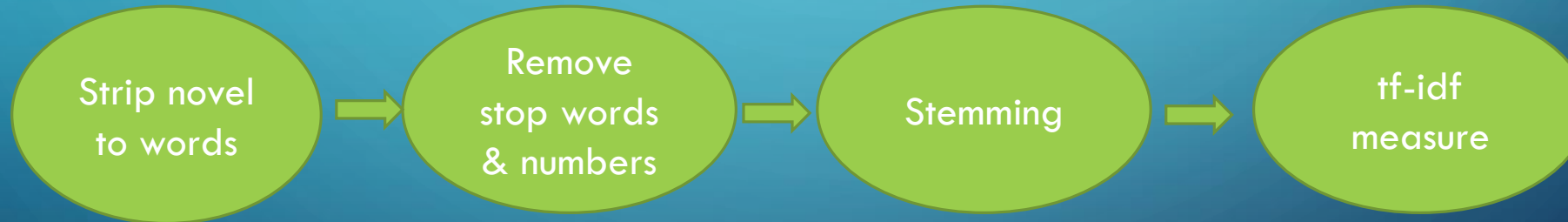
# LESMIS DATASET

R package SOMbrero



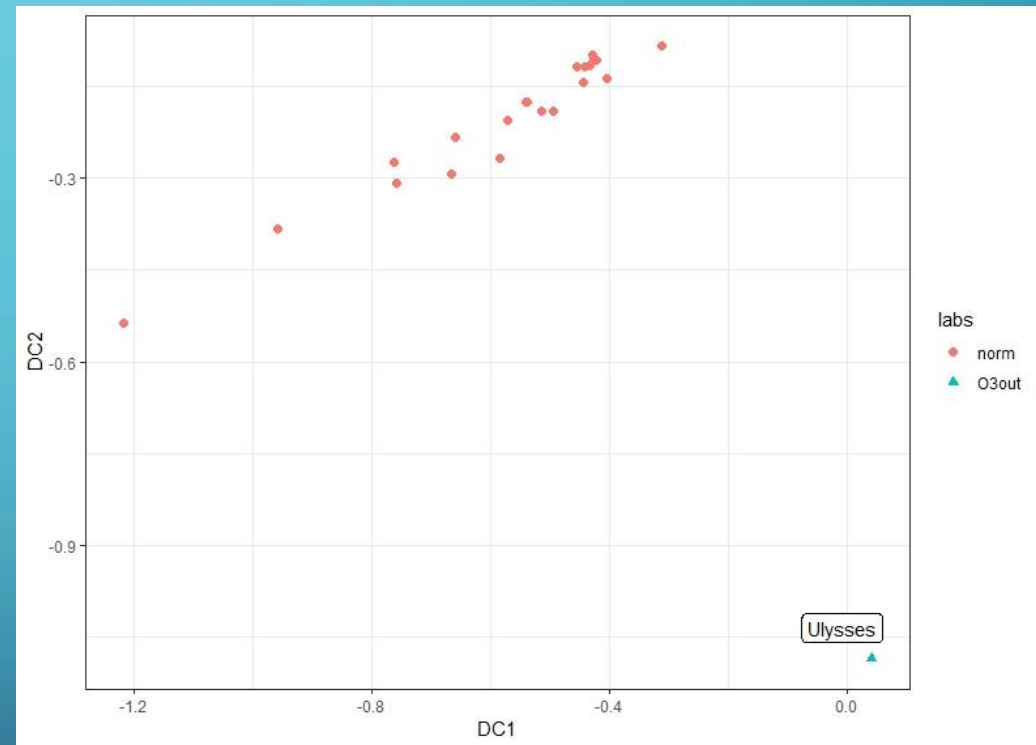
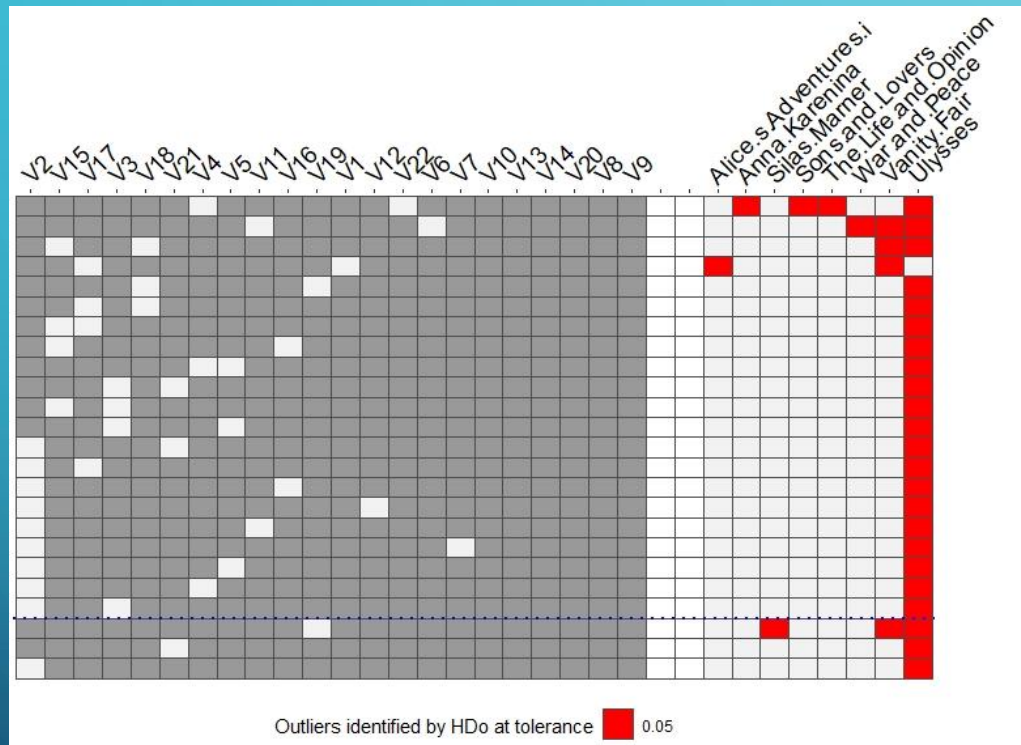
# CLASSICS FROM GUTENBERG

- 22 classics downloaded from Gutenberg project
- Alice in Wonderland, Anna Karenina, Bleak House, Emma, Frankenstein, Gullivers Travels, Jude the Obscure, Lord Jim, Mansfield Park, Middlemarch, Moby Dick, Northanger Abbey, Persuasion, Pride and Prejudice, Sense and Sensibility, Silas Marner, Sons and Lovers, The Life and Opinions of Tristram Shandy, Wizard of Oz, Ulysses, Vanity Fair and War and Peace





# CLASSICS FROM GUTENBERG



# USES OF DOBIN

- Dimension reduction for outlier detection
- Visualization of outliers

# THANK YOU!

- R package *dobin* is on CRAN [http://bit.ly/cran\\_dobin](http://bit.ly/cran_dobin)
- Slides at <https://sevvandi.netlify.com/>
- Paper on RG: [http://bit.ly/paper\\_dobin](http://bit.ly/paper_dobin)
-  @sevvandik
-  sevvandi

