RnavGraph

Interactive visual clustering

Adrian Waddell
and Wayne Oldford

SSC June 14, 2011

UNIVERSITY OF
WATERLOO

uwaterloo.ca

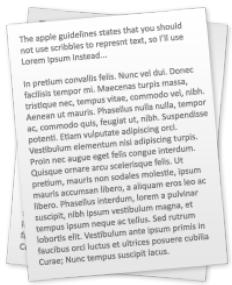
Challenge

- p values on each of n individuals

Challenge

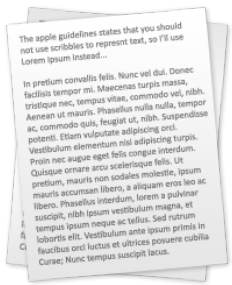
- *p* values on each of *n* individuals

- data can have a complex structure



Challenge

- *p* values on each of *n* individuals
- data can have a complex structure

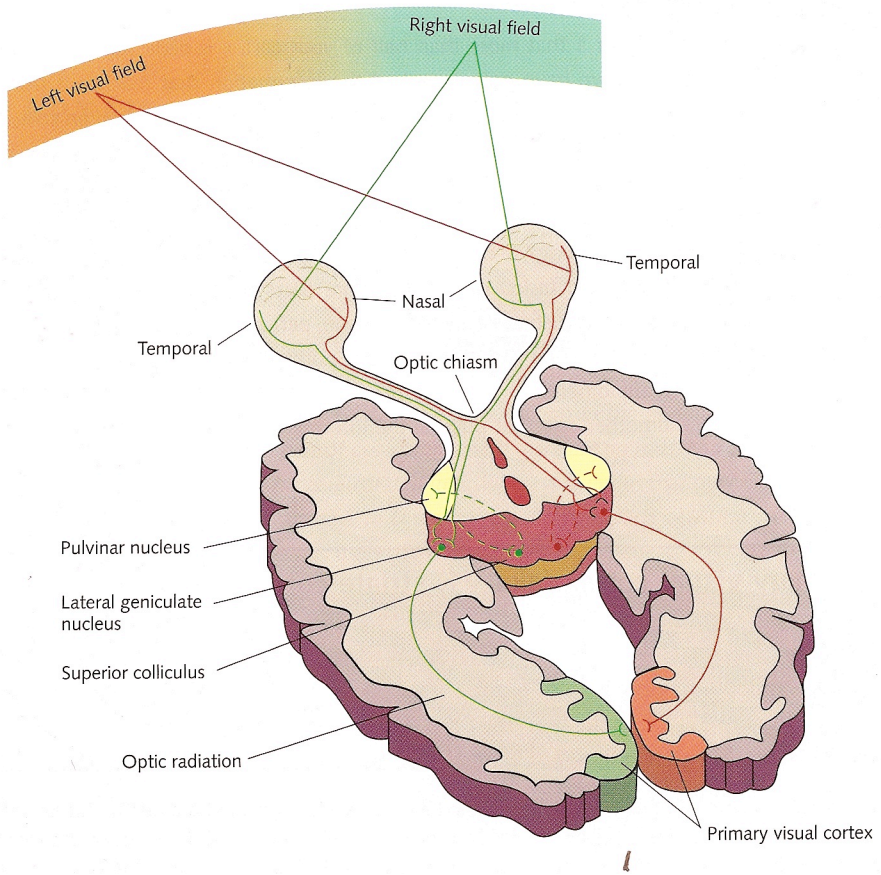





- *n*, or *p*, or both can be very large

Why Visualization?

- powerful human visual system
  - patterns, relations, textures,...
  - recognize structure
  - discover structure
- data analysis objectives
  - relations, clusters, classes,
    outliers, diagnostics, unusual structure, ...



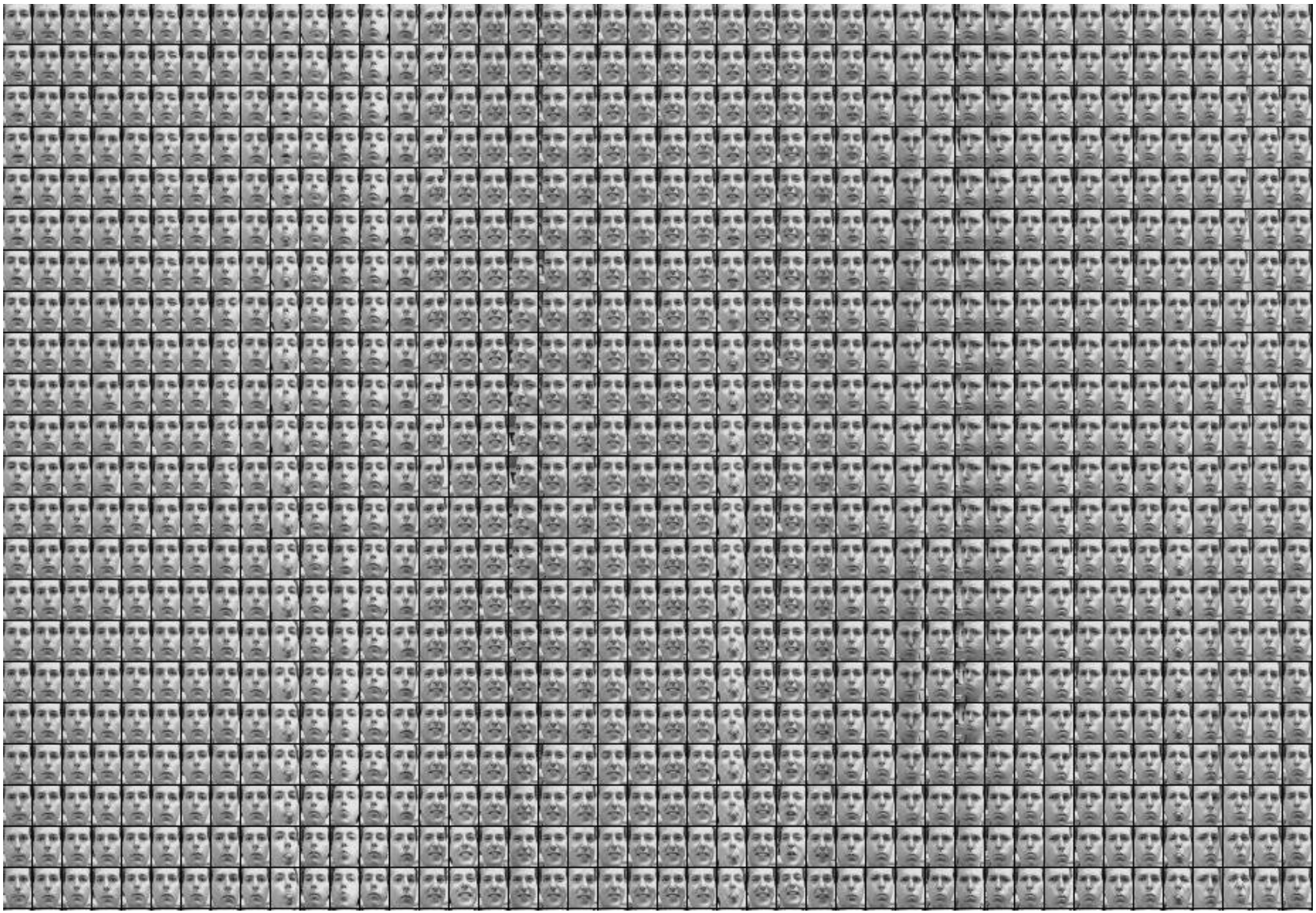
Dealing with high dimensionality $\longrightarrow$ large p

- visually, we are constrained to small p
  - locations: $p < 4$
  - use color, shape, texture, movement,...
- large number of low dimensional views
  - $\binom{p}{d}$ d-dimensional views
  - How? Which ones? And, in what order?

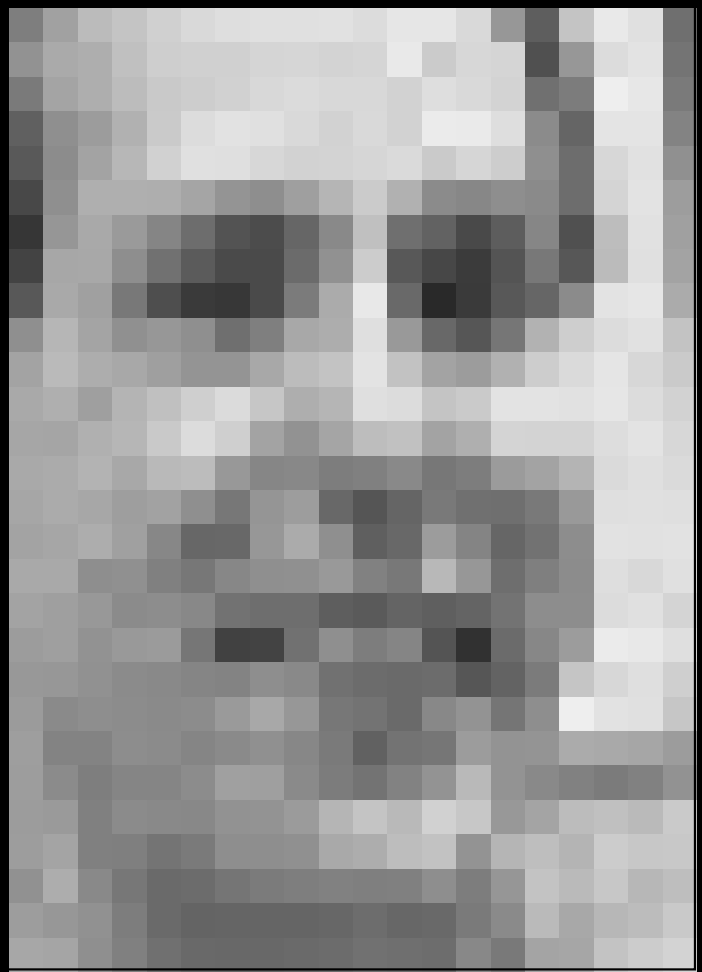
Proposed Approach

- reduce numbers
  - p overall dimensionality
  - $\binom{p}{d}$ subset of all low-d views
- interactive/tailored low-d views
- connect low-d views via interactive navigation graphs

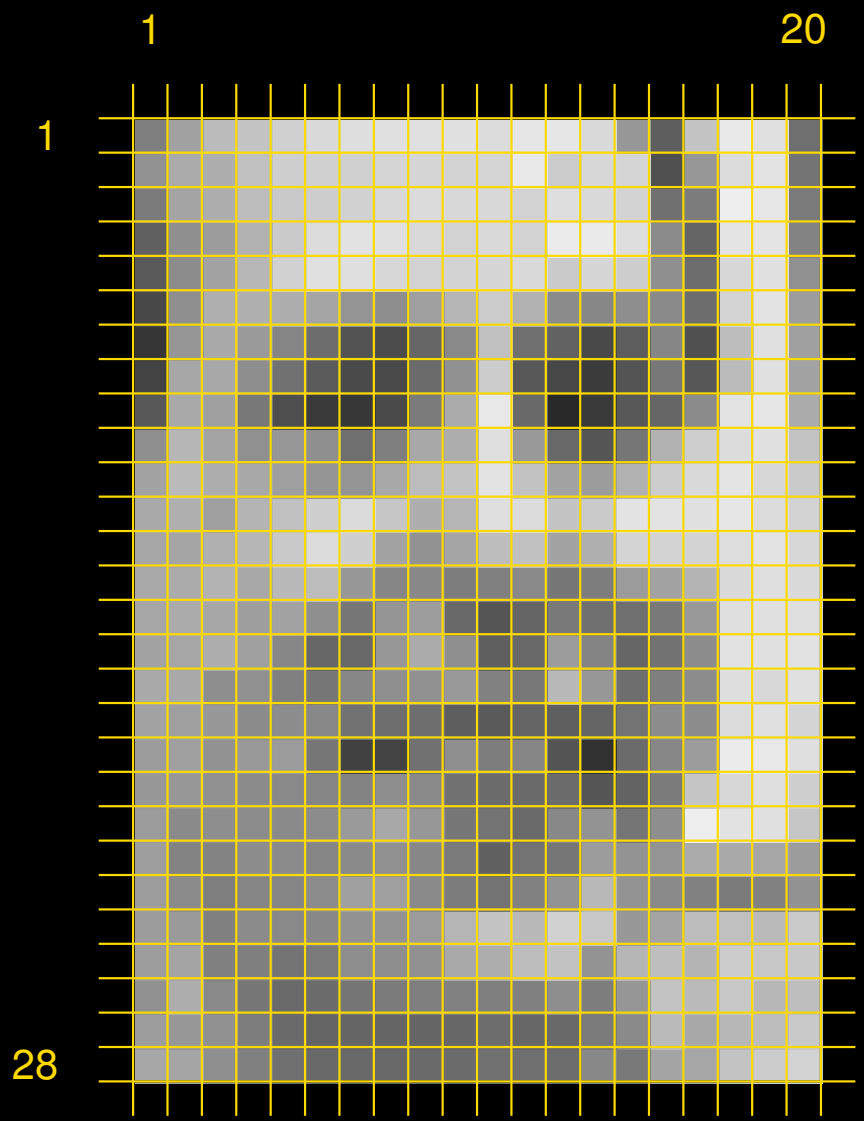
Example: Frey faces from 1965 movie frames



Reduce dimensionality

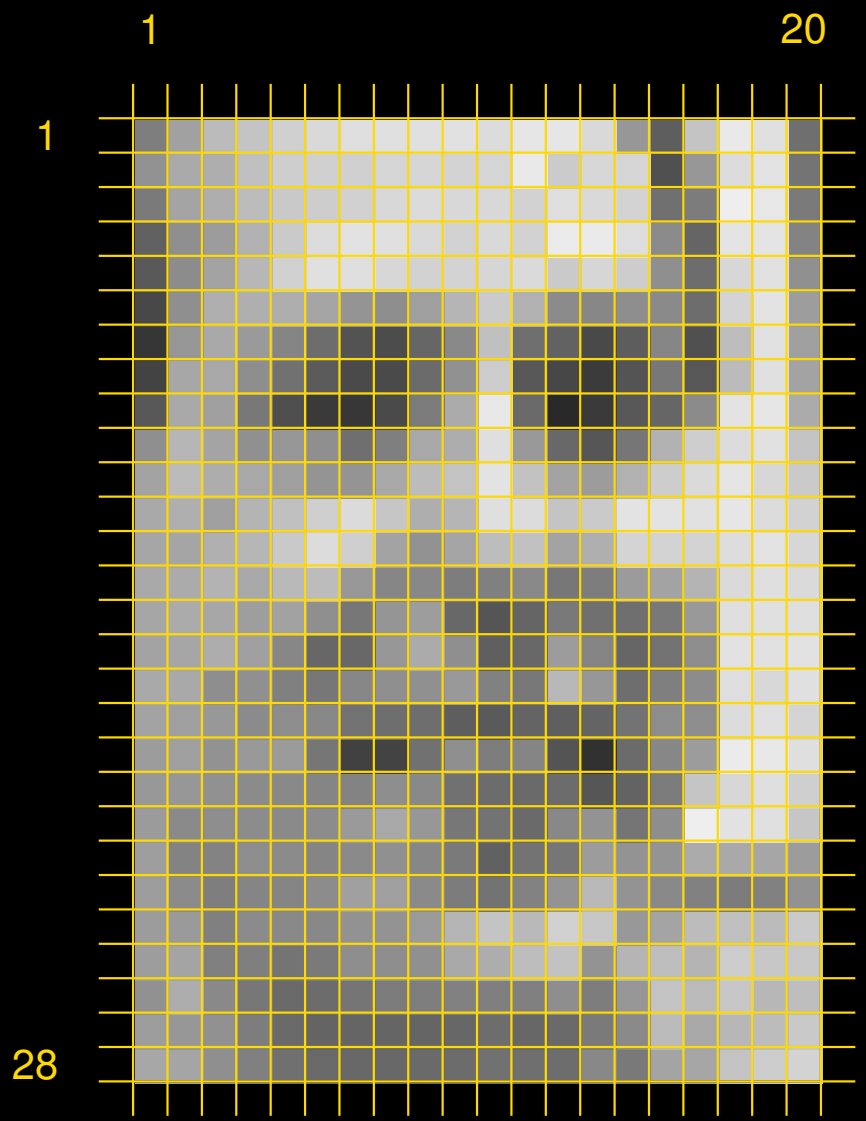


Reduce dimensionality



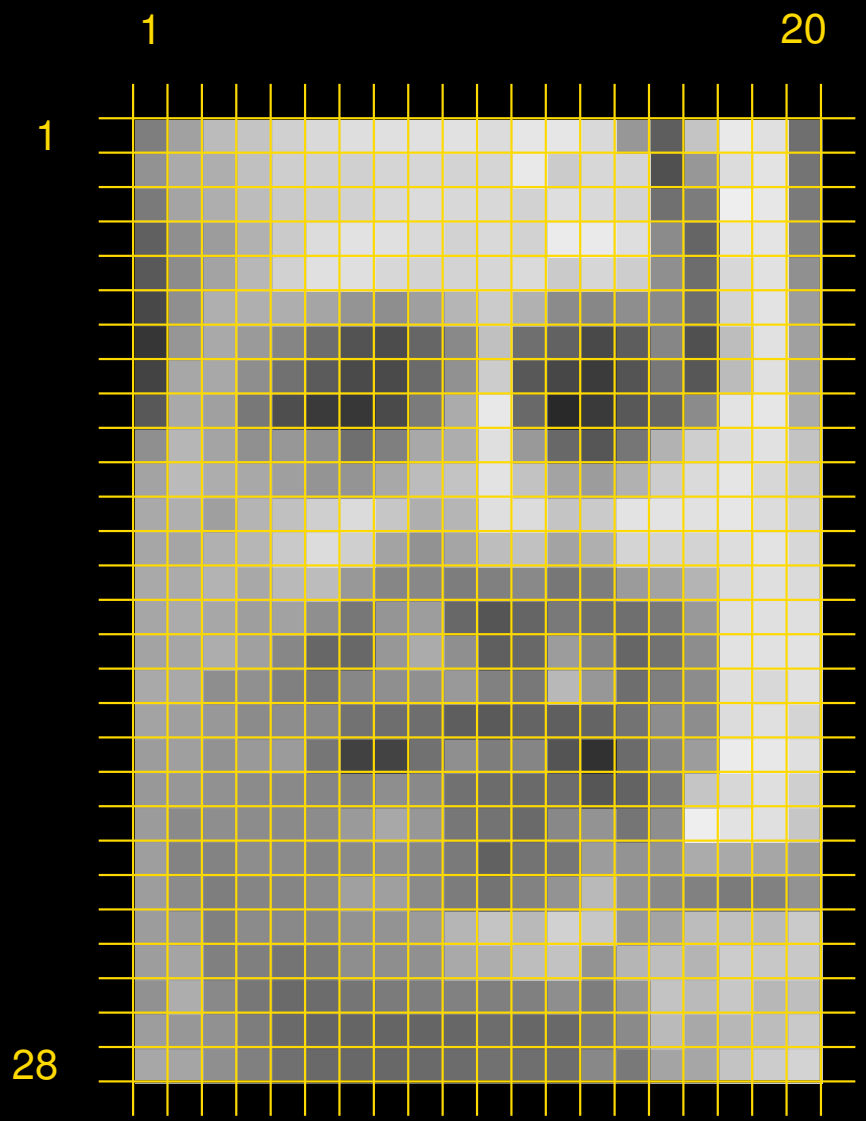
- $28 \cdot 20 = 560$ dimensions

Reduce dimensionality



- $28 \cdot 20 = 560$ dimensions
- explore via low dimensional spaces

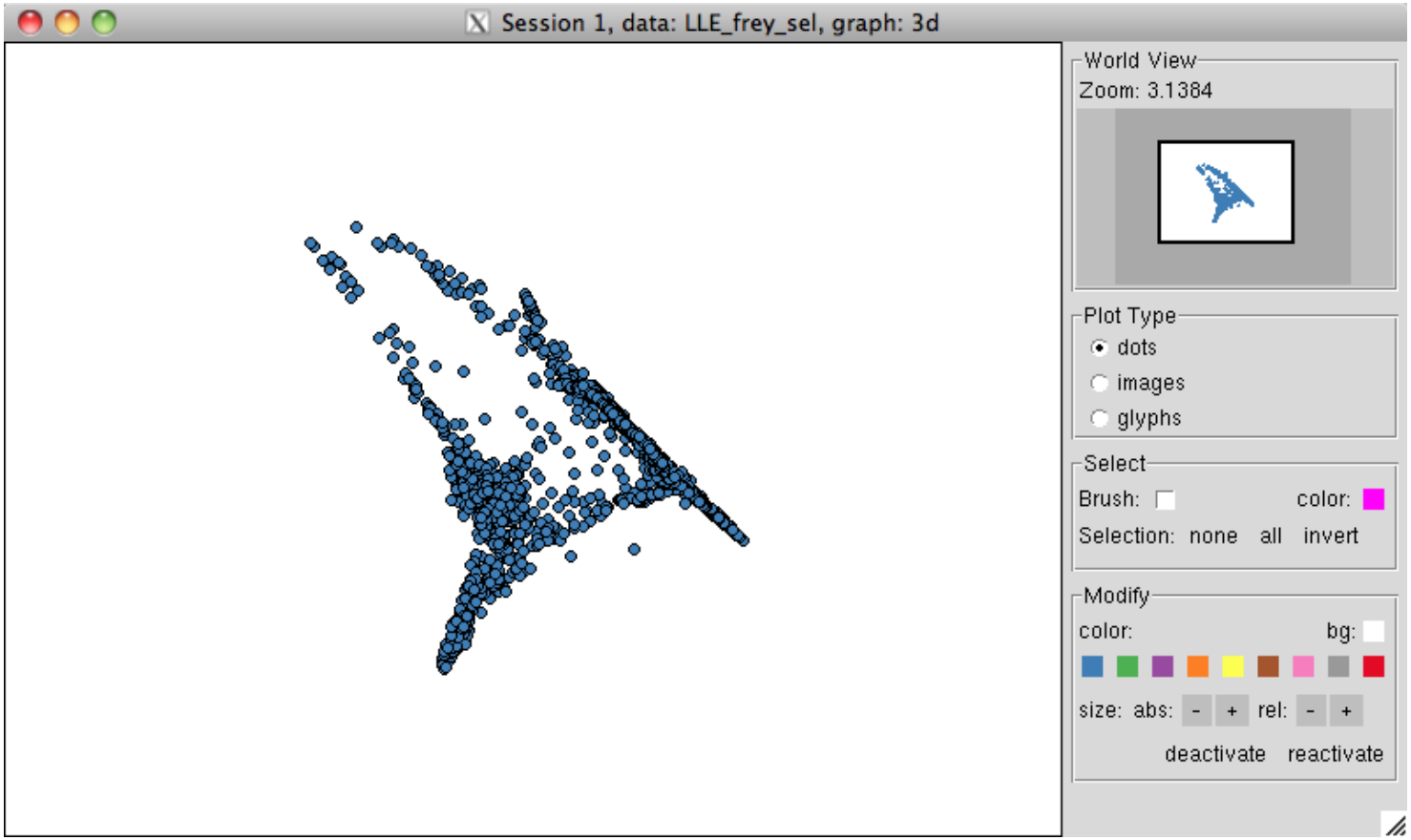
Reduce dimensionality



- $28 \cdot 20 = 560$ dimensions
- explore via low dimensional spaces
- Using LLE: local linear embedding
- $k = 12$ neighbors
- reduce to 5 dimensions

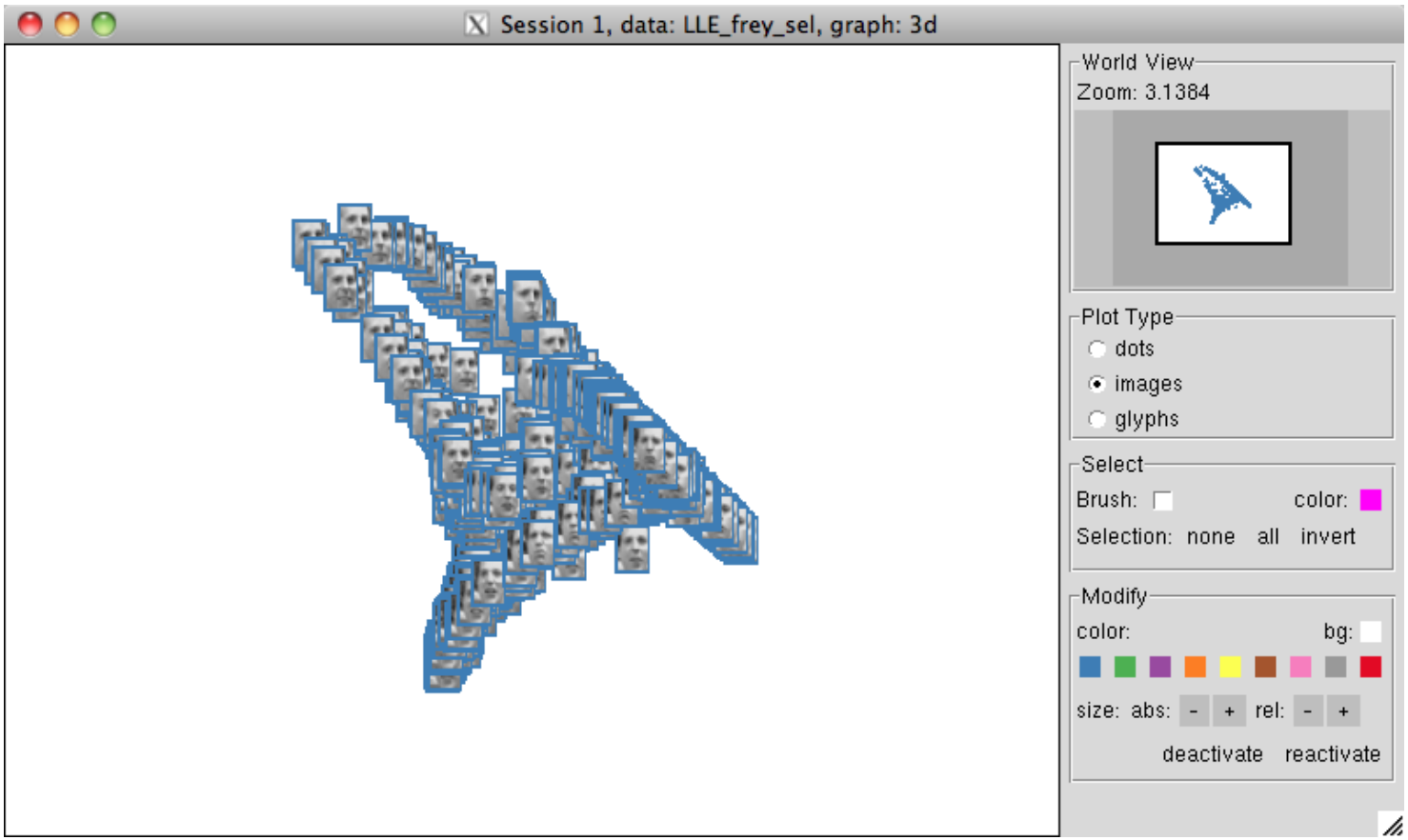
$$(x_1, x_2, x_3, x_4, x_5)$$

Interactive 2*d* view



- (x_1, x_2) - 2*d* dot plot

Interactive 2*d* view



- (x_1, x_2) - 2*d* image plot

Interactive *2d* view

- Point selection, brushing

Interactive *2d* view

- Deactivate complementary points

Interactive *2d* view

- Images, zoom, relocate

Interactive *2d* view

- Resize, zoom

Interactive *2d* view

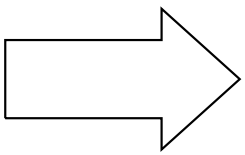
- Zoom out, reactivate, resize, pan

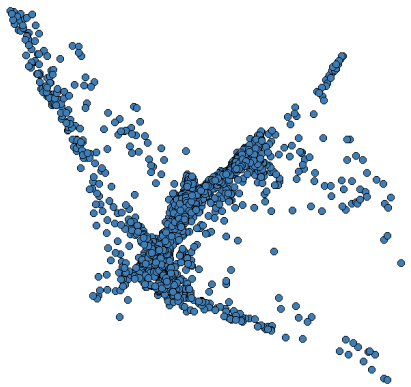
Interactive *2d* view

- Zoom out, back to dots

Connecting the views

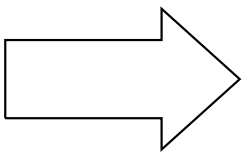
(x_1, x_2)

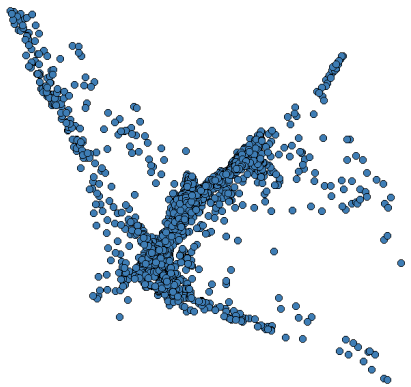




(x_2, x_3)

Connecting the views

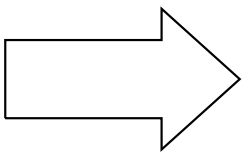


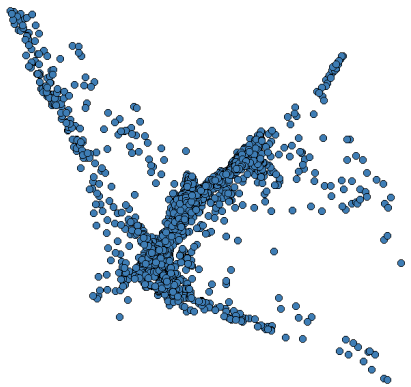


(x_1, x_2) (x_2, x_3)

- $3d$ rigid rotation

Connecting the views

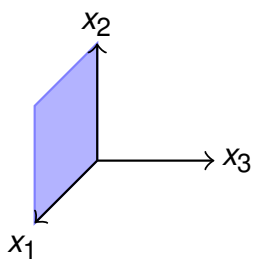




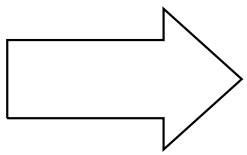
(x_1, x_2)

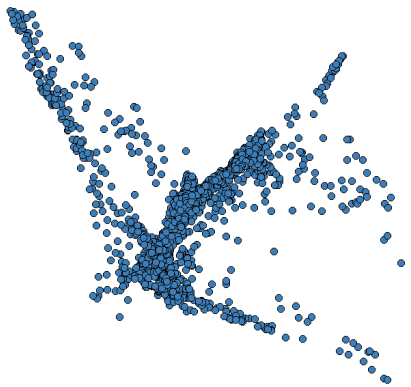
(x_2, x_3)

- $3d$ rigid rotation



Connecting the views

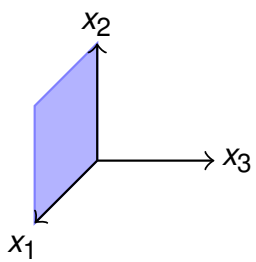


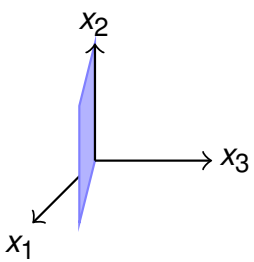


(x_1, x_2)

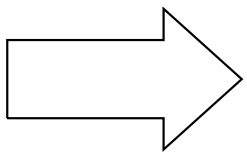
(x_2, x_3)

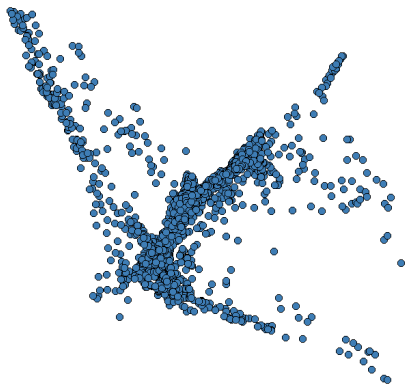
- 3d rigid rotation





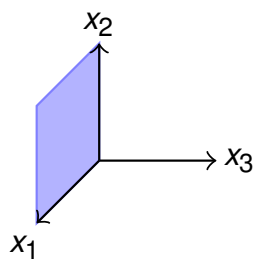
Connecting the views

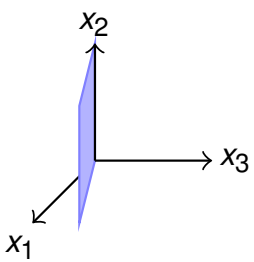


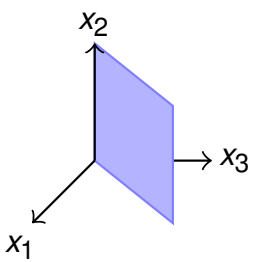


(x_1, x_2) (x_2, x_3)

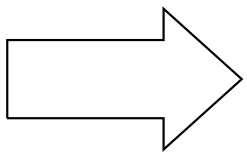
- $3d$ rigid rotation

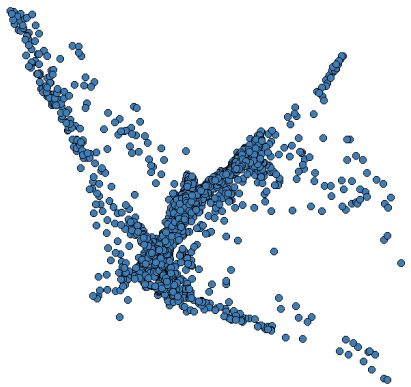






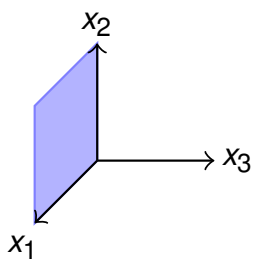
Connecting the views

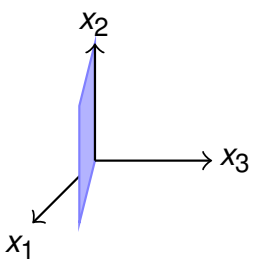


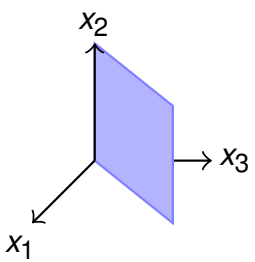


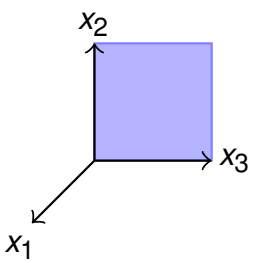
(x_1, x_2) (x_2, x_3)

- $3d$ rigid rotation



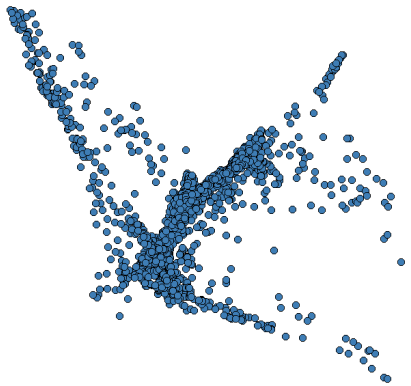






Connecting the views

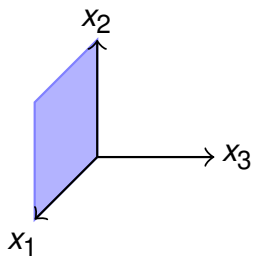
(x_1, x_2)

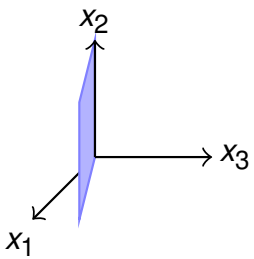


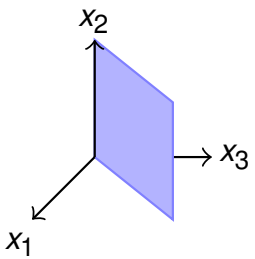
(x_2, x_3)

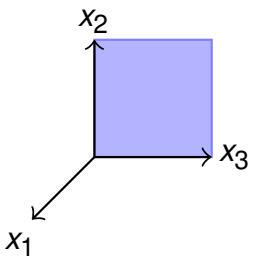
$3d$ - transition

- $3d$ rigid rotation

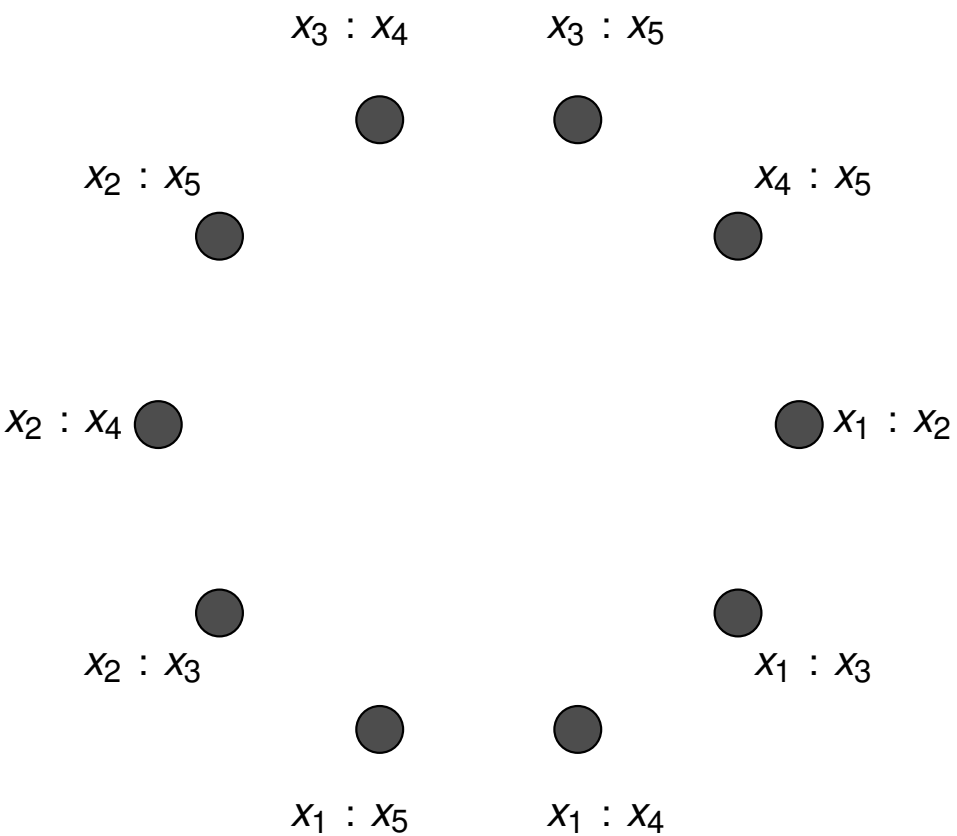








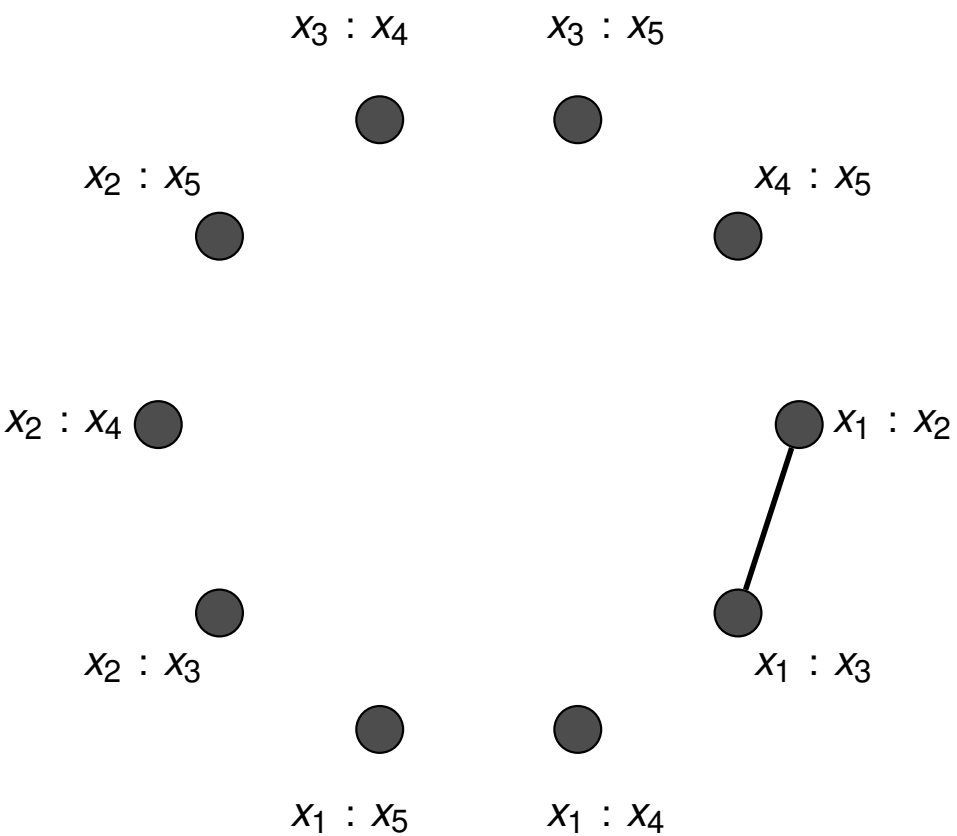
A 3*d* transition graph



- Each node is a 2*d* view

$$\binom{5}{2} = 10$$

A 3*d* transition graph

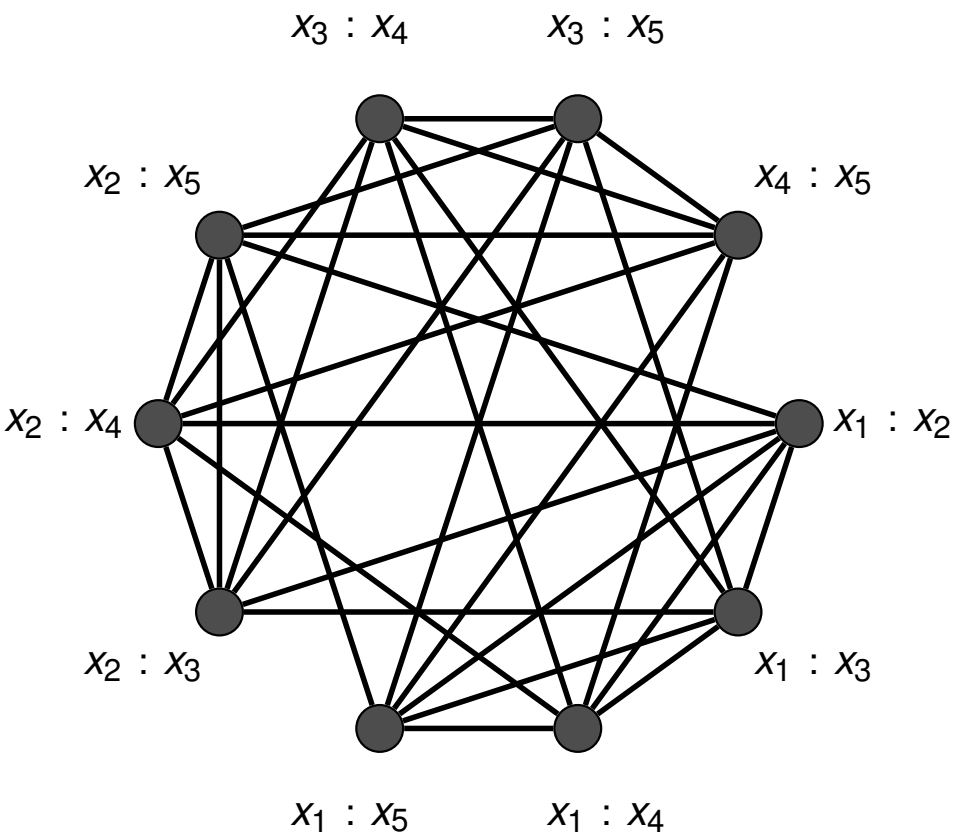


- Each node is a 2*d* view

$$\binom{5}{2} = 10$$

- Each edge is a 3*d* transition

A 3*d* transition graph

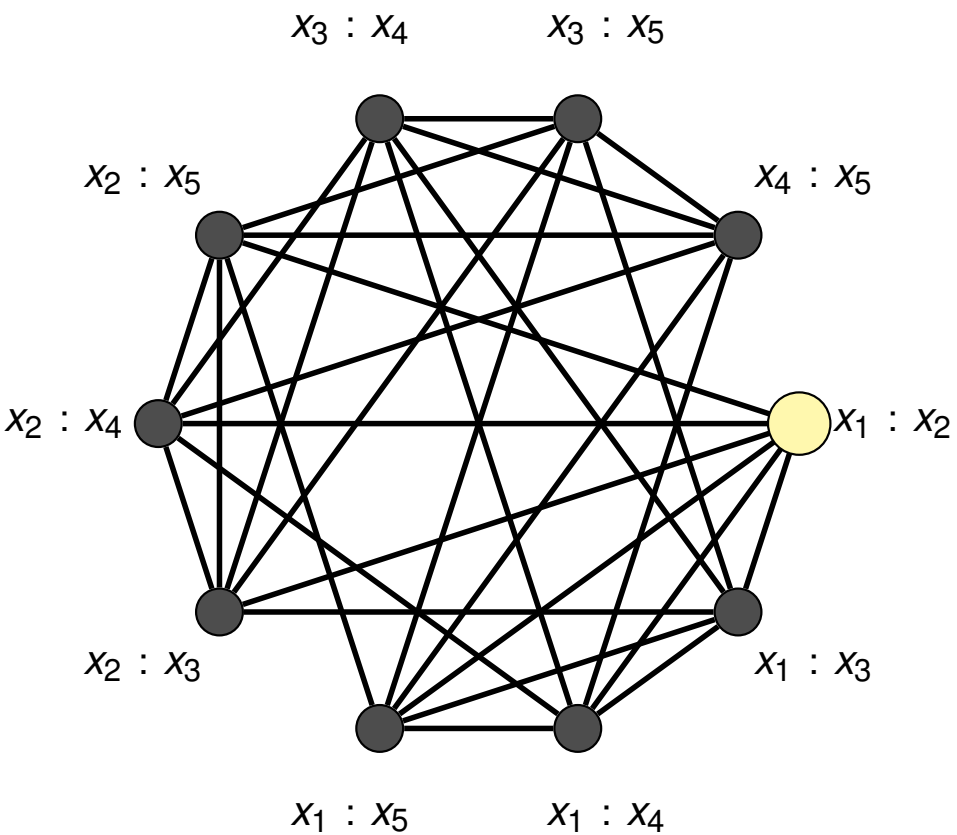


- Each node is a 2*d* view

$$\binom{5}{2} = 10$$

- Each edge is a 3*d* transition

A 3*d* transition graph

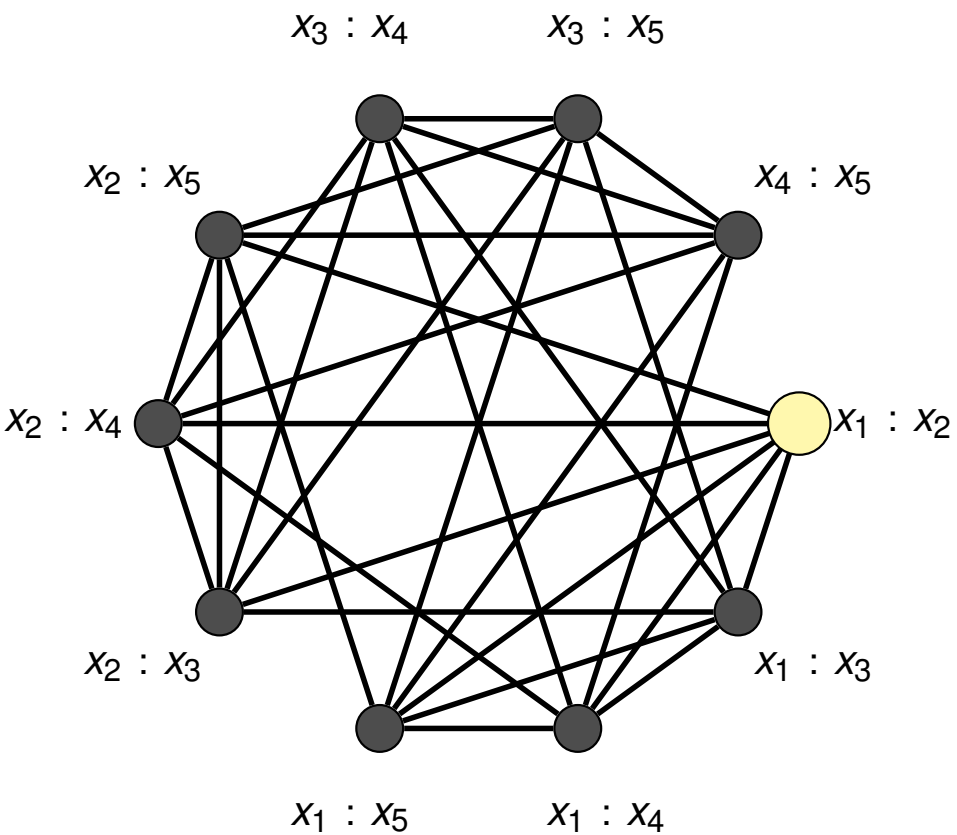


- Each node is a 2*d* view

$$\binom{5}{2} = 10$$

- Each edge is a 3*d* transition
- You are here bullet

A 3*d* transition graph

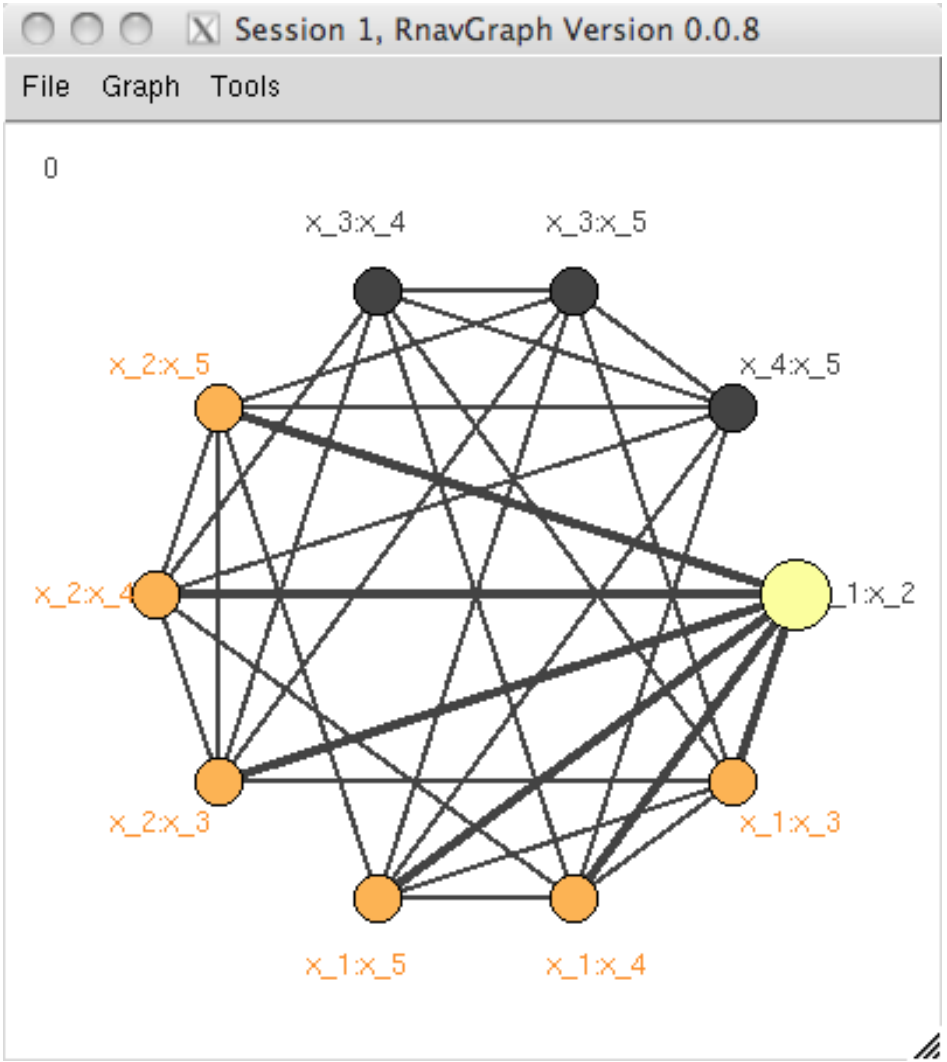


- Each node is a 2*d* view

$$\binom{5}{2} = 10$$

- Each edge is a 3*d* transition
- You are here bullet
- This is a navigation graph (NavGraph)

A 3*d* transition graph



- Each node is a 2*d* view

$$\binom{5}{2} = 10$$

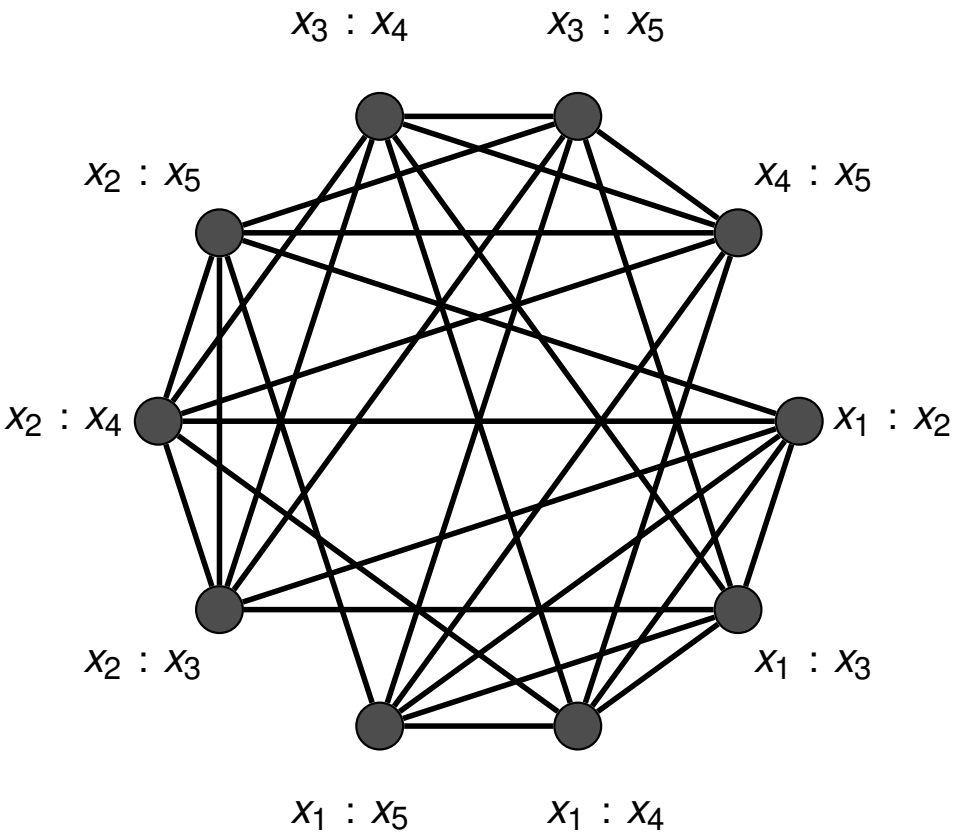
- Each edge is a 3*d* transition
- You are here bullet
- This is a navigation graph (NavGraph)

Interactive navigation graph

Move the bullet $2d$ view changes in response

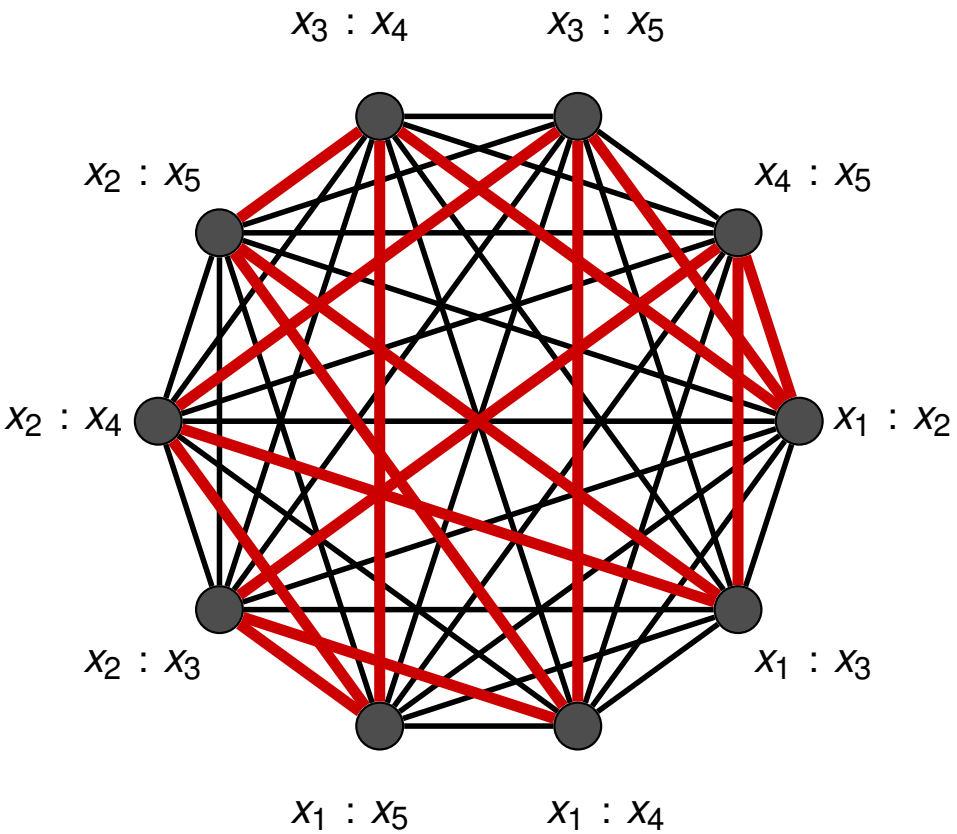
- Can stop anywhere and interact with the low-d view

3*d* and 4*d* transition graphs



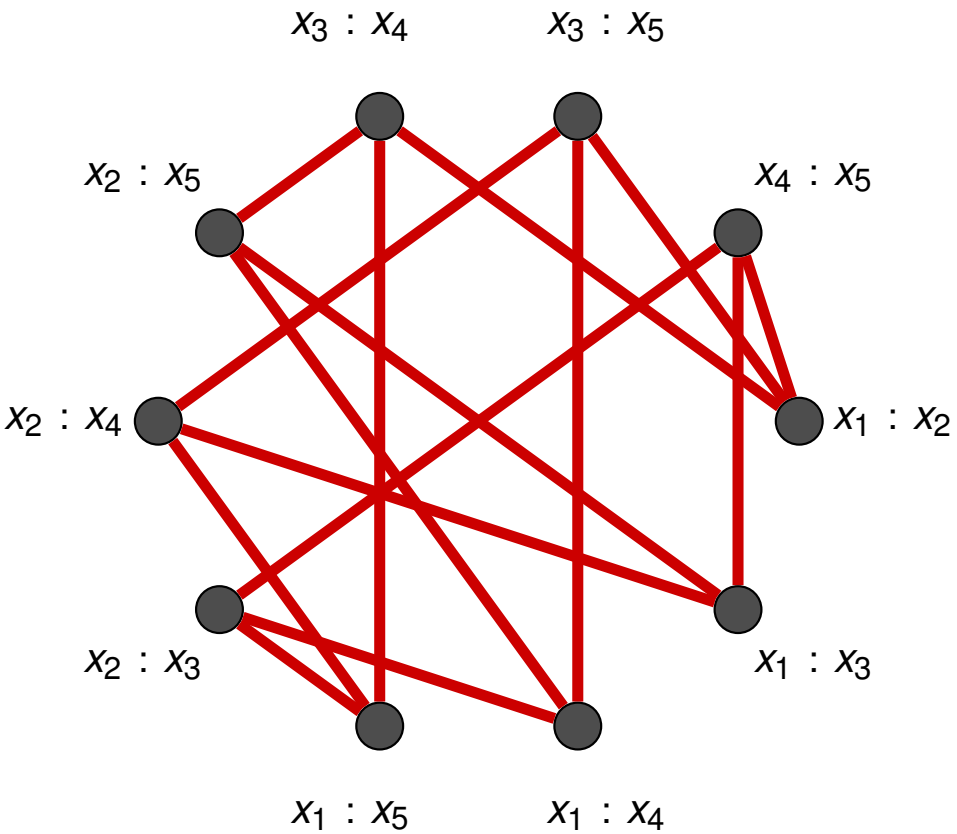
- complement of 3*d* transition graph yields 4*d* transition graph

3*d* and 4*d* transition graphs



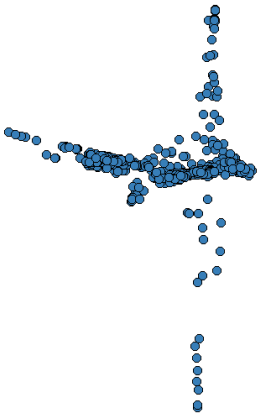
- complement of 3*d* transition graph yields 4*d* transition graph

3*d* and 4*d* transition graphs

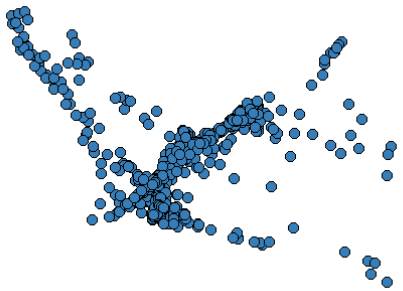


- complement of 3*d* transition graph yields 4*d* transition graph

$4d$ **transition** $(x_1 : x_5) \longrightarrow (x_2 : x_3)$

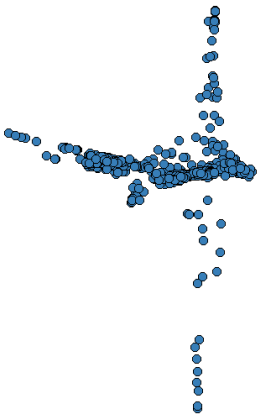


(x_1, x_5)

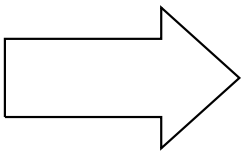


(x_2, x_3)

$4d$ **transition** $(x_1 : x_5) \longrightarrow (x_2 : x_3)$

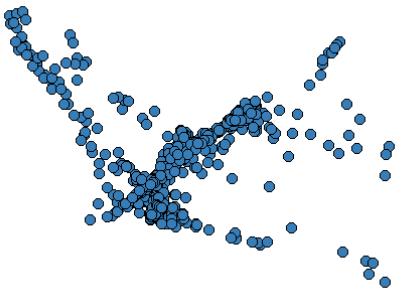


(x_1, x_5)

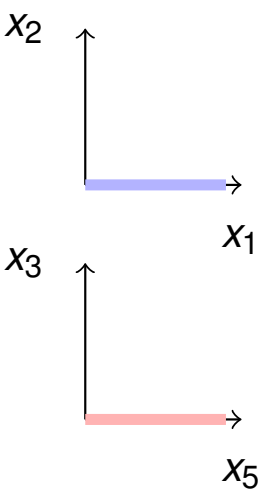


x-axis

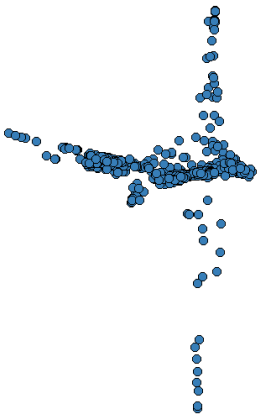
y-axis



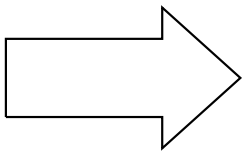
(x_2, x_3)



$4d$ **transition** $(x_1 : x_5) \longrightarrow (x_2 : x_3)$

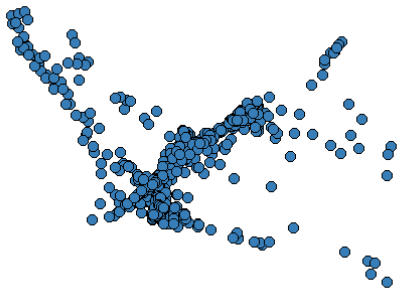


(x_1, x_5)

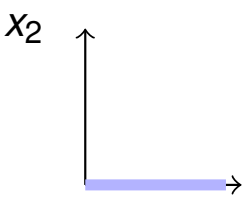


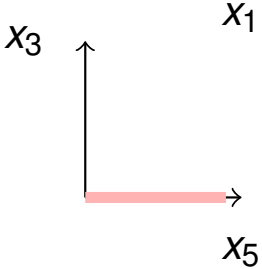
x-axis

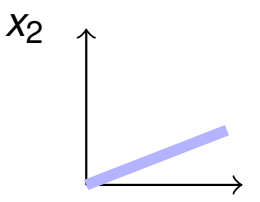
y-axis

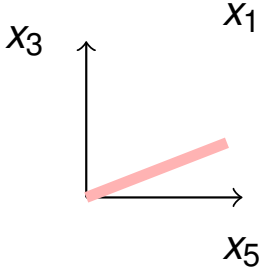


(x_2, x_3)

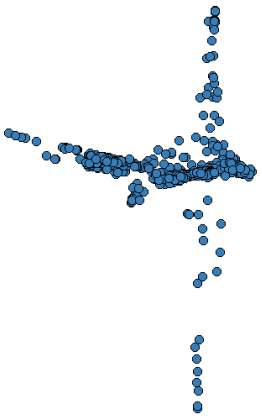




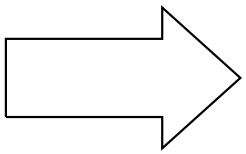


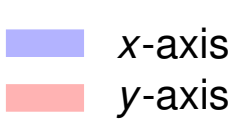


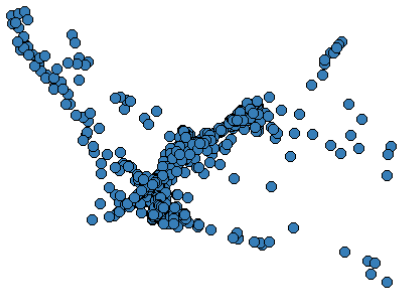
$4d$ **transition** $(x_1 : x_5) \longrightarrow (x_2 : x_3)$



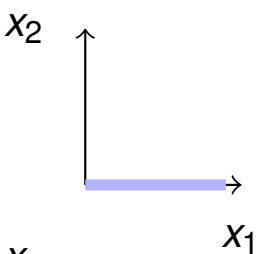
(x_1, x_5)

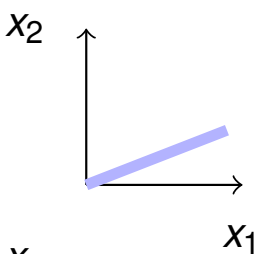


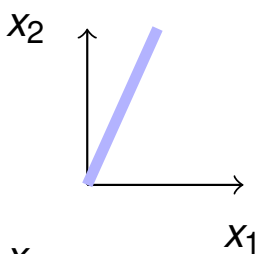




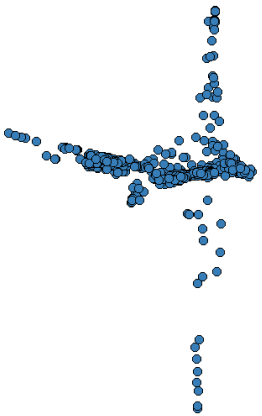
(x_2, x_3)



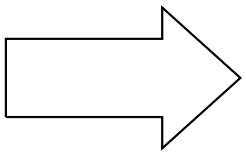




$4d$ **transition** $(x_1 : x_5) \longrightarrow (x_2 : x_3)$

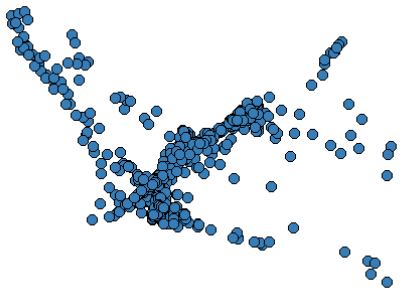


(x_1, x_5)

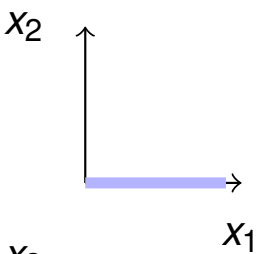


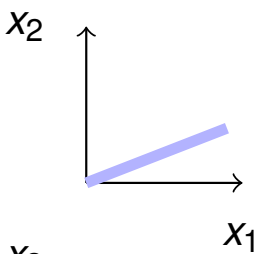
x-axis

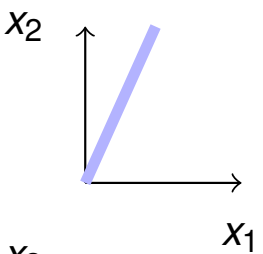
y-axis

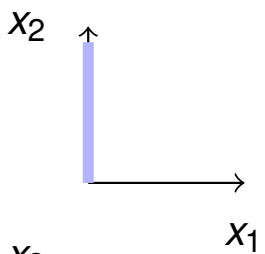


(x_2, x_3)









$4d$ **transition** $(x_1 : x_5) \longrightarrow (x_2 : x_3)$

- $4d$ transition is NOT a rigit rotation

Try the package yourself

- This is only a part of our `R` package
- Try it yourself
- Package is on CRAN: `install.packages('RnavGraph')`
- Install dependencies and suggested packages
- Read the vignette
- Try the demos

Papers

- Oldford and Waddell
  - Visual clustering of high-dimensional data by navigating low-dimensional spaces (**ISI Dublin, 2011**)
  - RnavGraph: A visualization tool for navigating through high dimensionaldata (**ISI Dublin, 2011**)
  - `RnavGraph` R package, available on CRAN
- Hurley and Oldford
  - Graphs as a navigational infrastructure for high dimensional data spaces (**Comp Stats 2011**)
  - Pairwise display of high dimensional information via Eulerian tours and Hamiltonian decompositions (**JCGS, 2010**)
    - Eulerian tour algorithms for data visualization and the `PairViz` R package (**Comp Stats 2011**)
    - **PairViz** R package, available on CRAN