Geometric Hypergraphs pt3

Recap

- Geometric hypergraphs arise when one
  - Associates a geometric object (line segment, circle, rectangle, sphere….) to each vertex
  - Define an edge as occuring between vertices when the respective geometric objects relate in a certain way (intersect, containment, do not intersect, have a minimum distance between them of a certain value or less etc….)
- Geometric hypergraphs tend to have many “nice properties” – they tend to be “very special” in some senses (implies probably easier to solve problems on them)

Having the geometric embedding or not

- We gave an example of MaxCon robust fitting – we start with the data (line segments). So we know the embedding Graph <-> Data (segments).
- This means we can also apply our algorithms "in Data space". Indeed, we might not even want to construct the graph (or any significant part of it).
- We can think of the "things we can do" in data space as "oracles" to our underlying graph. We will see that for MaxCon there is a data space oracle for "Clique" (and hence for independence in the complement graph). Checking for being a clique or an independent set in the hypergraph can often be costly whereas with the data space oracle it might be "fast".

Having the geometric embedding or not

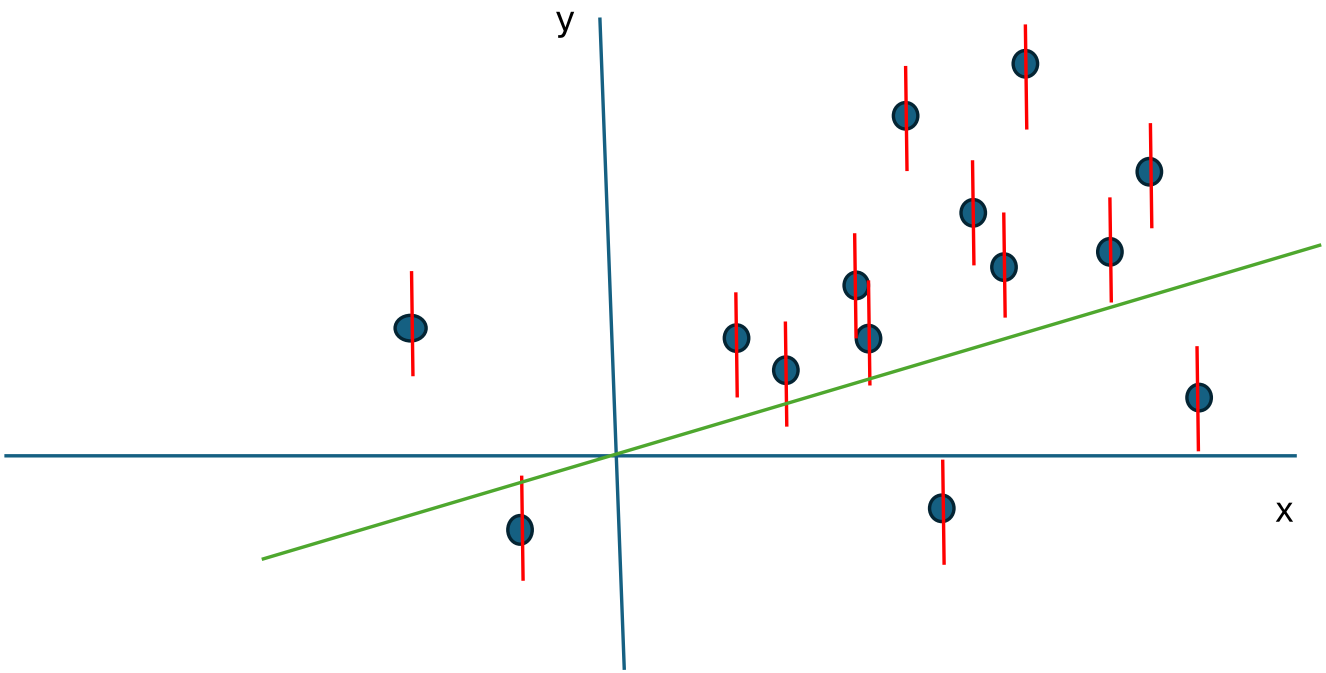
- Indeed, we can in principle identify three situations
  1. We have the data (only). We could in principle create the hypergraph but we might not want to (particularly k-uniform hypergraphs for large k)
  2. We have the (hypergraph) only. In such a situation it is *usually* VERY difficult to derive the embedding into data space.
  3. We have both the data space embedding and the graph.
- We are mainly going to concentrate on the first of these. Moreover, in large part concentrating on MaxCon for computer vision…

Other problems…

- We are mainly going to concentrate on the first of these. Moreover, in large part concentrating on MaxCon for computer vision…however, many other types of application can be found.
- For MaxCon we have only looked at the relatively uninteresting problem of 1-D location estimation.
- Consider now, regression of a 1D subspace in 2D
  - Given a number of points in 2D (x_i,y_i), and a vertical tolerance on the y-coordinate, find the 1D subspace (line through the origin) that "meets" as many as possible tolerance intervals

Other problems…

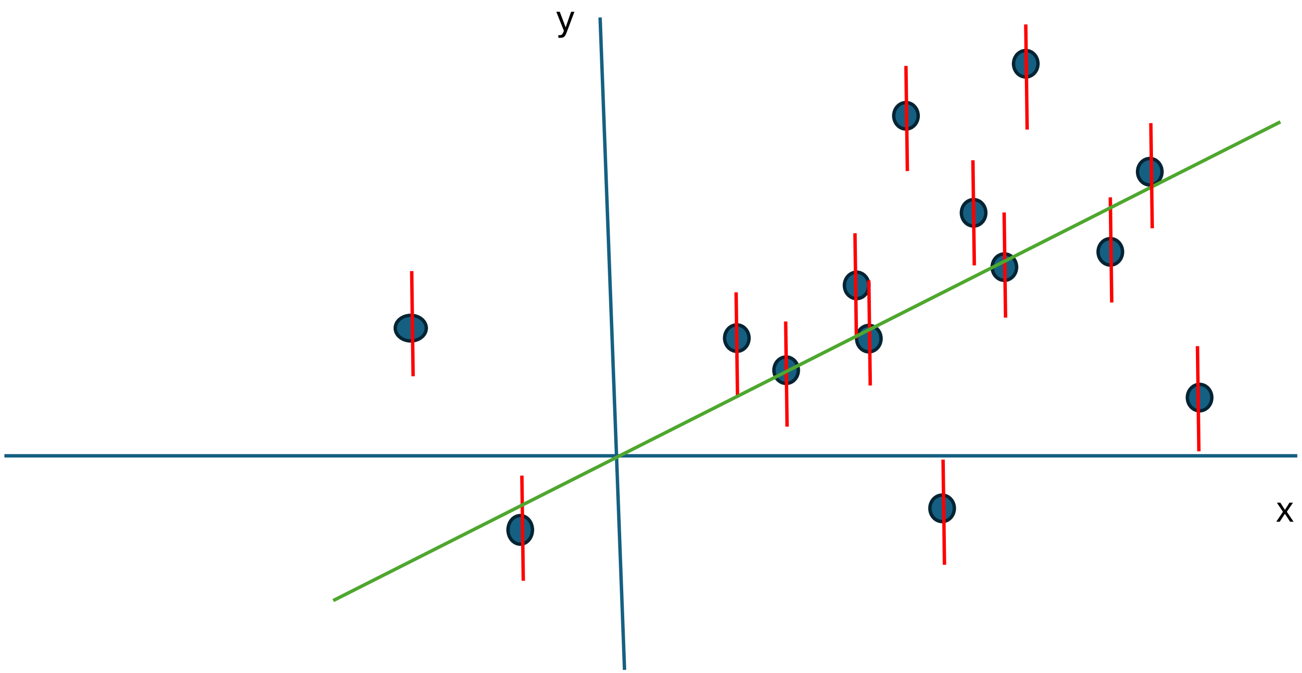
- Consider now, regression of a 1D subspace in 2D
  - Given a number of points in 2D (x_i,y_i), and a vertical tolerance on the y-coordinate, find the 1D subspace (line through the origin) that "meets" as many as possible tolerance intervals



"stab as many red intervals as possible" by a line (hypothesis) going through the origin.

Other problems...

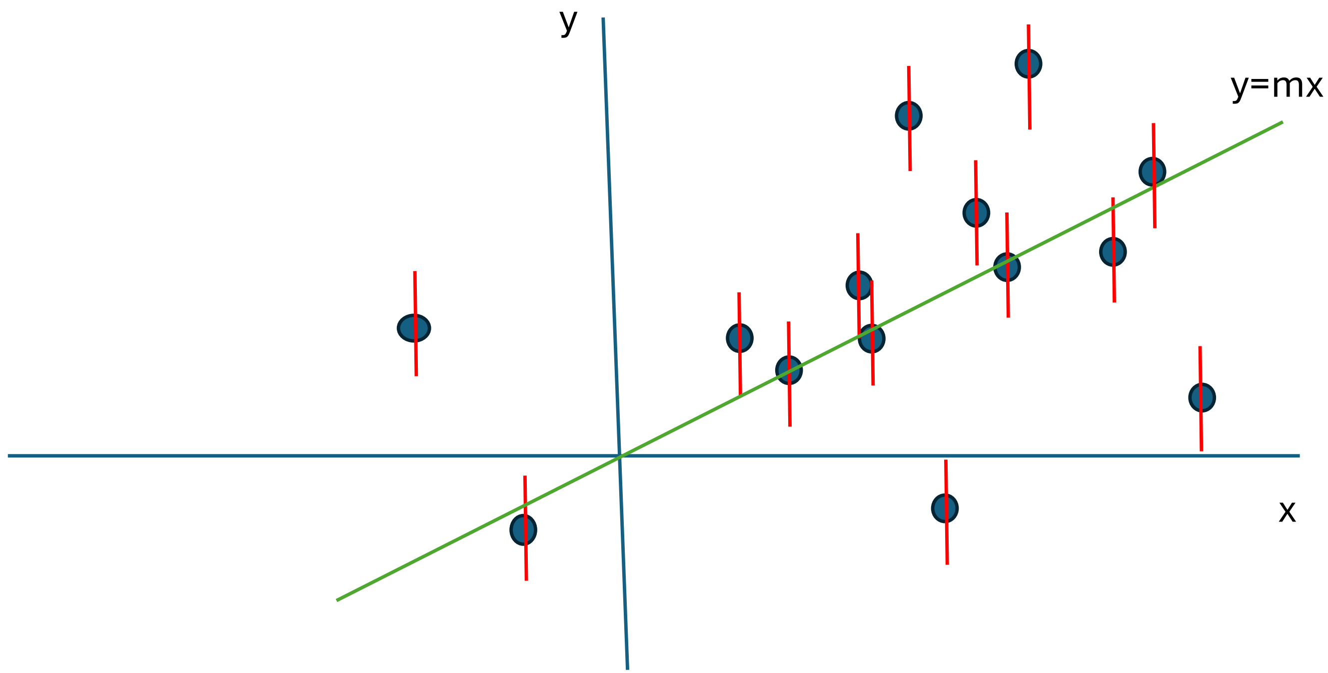
- Consider now, regression of a 1D subspace in 2D
  - Given a number of points in 2D (x_i,y_i), and a vertical tolerance on the y-coordinate, find the 1D subspace (line through the origin) that "meets" as many as possible tolerance intervals



"stab as many red intervals as possible" by a line (hypothesis) going through the origin.
A better hypothesis

Other problems…

$$\text{Maximize}_{m} \; |\,I\,| : I=\{(x_i,y_i):\; y_i-\epsilon \le m\, x_i \le y_i+\epsilon\}$$



"stab as many red intervals as possible" by a line (hypothesis) going through the origin.

A better hypothesis

What sort of hypergraph class is this?

The vertical intervals are “stabbable” by the same subspace iff the circular arc intervals overlap.
So…the graph we get is a circular arc graph.

i.e., we have shown that the class of graphs for MaxCon in 1D subspace regression is a subset of circular arc graphs

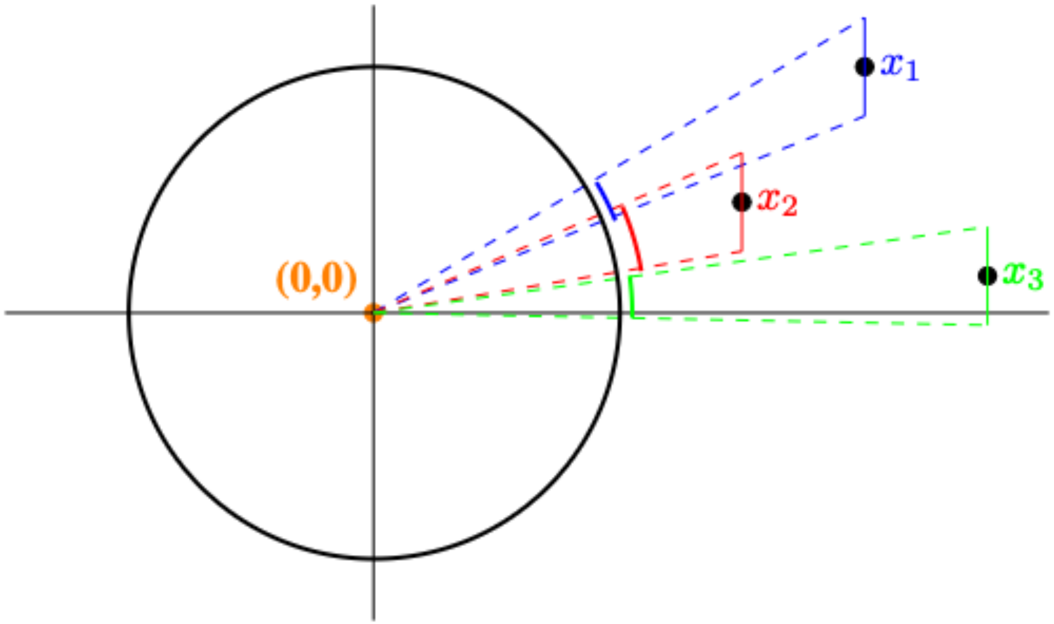


Figure 1: 1D subspace fitting and circular arc graphs

What sort of hypergraph class is this?

The vertical intervals are “stabbable” by the same subspace iff the intervals on the left hand line overlap. So…the graph we get is an interval graph.

i.e., we have shown that the class of graphs for MaxCon in 1D subspace regression is a subset of interval graphs

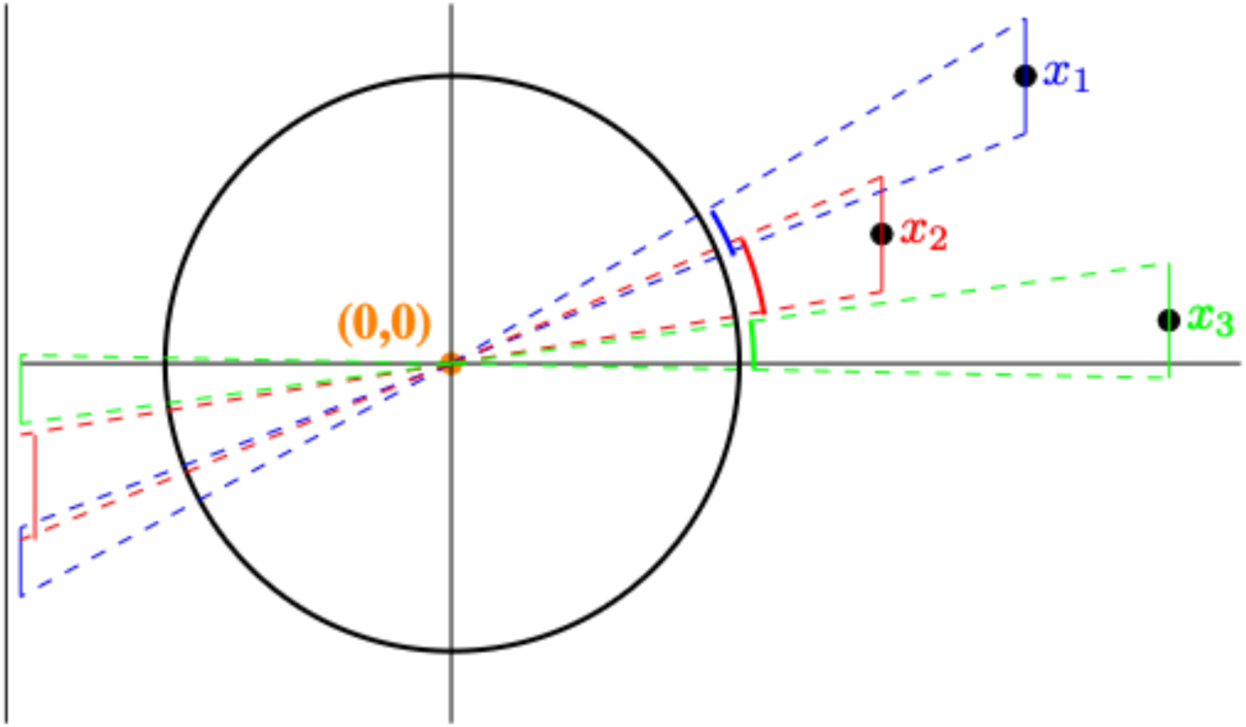


Figure 2: 1D subspace fitting and interval graphs

How can our 1D subspace regression problems map onto both circular arc and interval graphs??

The freely available SAGE package for maths has an interface to the graphclasses.org database

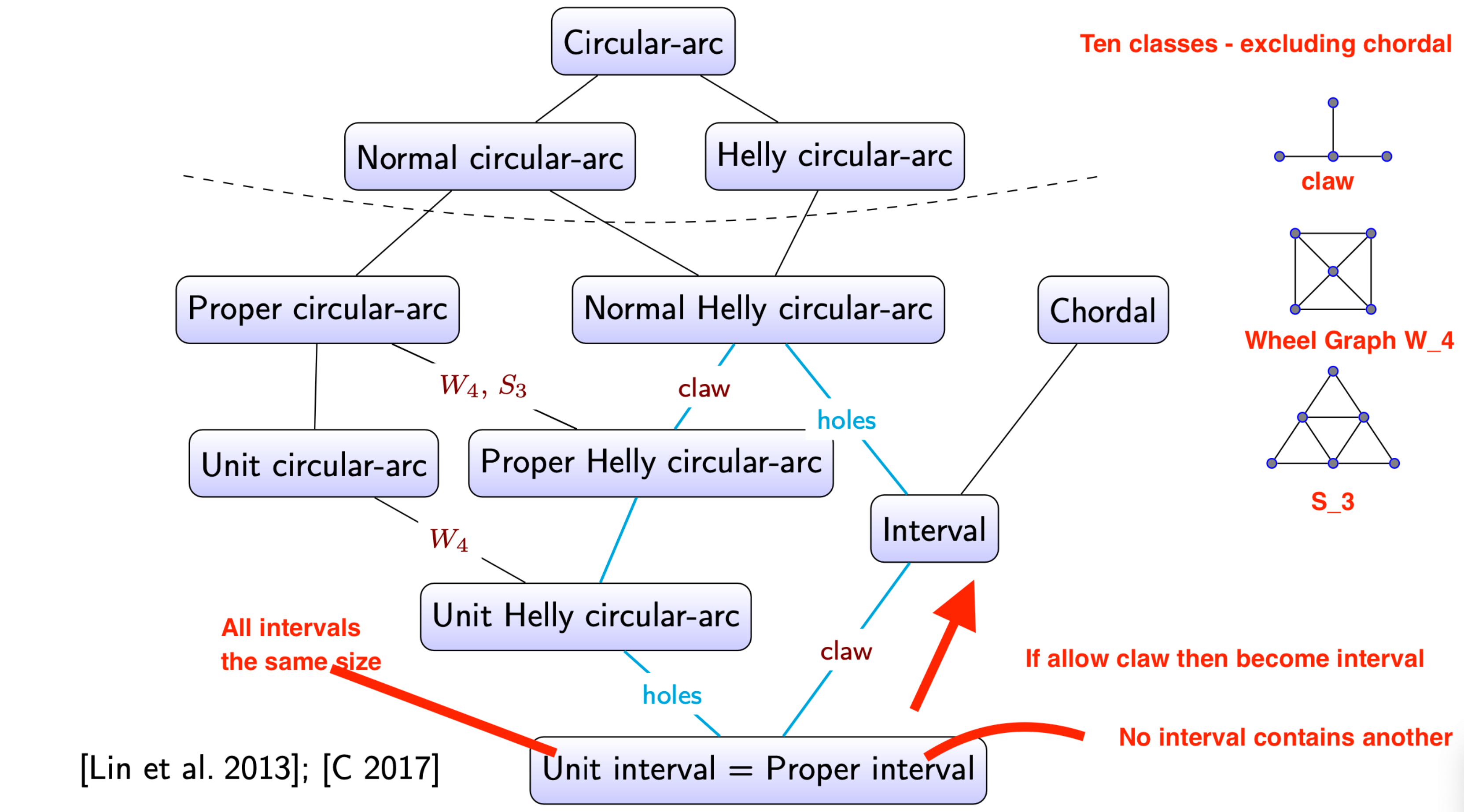
```
sage: sage: p=d.shortest_path("gc_234","gc_133")
sage: len(p) - 1
-1
sage: sage: p=d.shortest_path("gc_133","gc_234")
sage: len(p) - 1
1
sage: for c in p:
....:     print(graph_classes.get_class(c))
....:
circular arc graphs
interval graphs
sage:
```

So circular arc graphs are NOT contained in interval graphs

BUT interval graphs ARE contained in circular graphs

So...interval graphs are contained in circular arc graphs (surprised?)

The Gory Detail of the Difference between Circular Arc and Interval Graphs



MaxCon problems so far…

- 1D location problem maps onto (a subset of?) unit interval graphs
- 1D subspace regression problem (in 2D) maps onto (a subset of?) interval graphs

What do these have in common?

They are both situations where the model (location in 1D, subspace in 2D) is defined by only one point (the point itself in the 1D location problem and the projection of that point – onto the unit circle – ofr the 1D subspace problem in 2D)

In both problems a single point is always “feasible” (can be fit within tolerance by a model).

MaxCon problems so far…

They are both situations where the model (location in 1D, subspace in 2D) is defined by only one point (the point itself in the 1D location problem and the projection of that point – onto the unit circle – ofr the 1D subspace problem in 2D)

In both problems a single point is always “feasible” (can be fit within tolerance by a model).

Also, they both have TWO points as the minimal situation where “infeasible” situations can arise. i.e., not every pair of points can be fit (within tolerance) by a single model.

This minimal infeasible set size (2 here) is the key to determining whether we get a graph or a hypergraph.

MaxCon problems so far…

The minimal infeasible set size is the key to determining whether we get a graph or a hypergraph.

Now consider not subspaces in 2D (which have to go through the origin) but lines: $y=mx+b$

Now, for a general line in 2D, ANY 2 points CAN ALWAYS be fit by a single line. So the minimal infeasible set size is 3. So the relevant graph (capturing infeasible and feasible subsets of data) has edges of size 3: 3-uniform hypergraph

The is already a quite useful problem setting

Maximum Consensus line fitting – the image at left is made by a streak from a moving spaced object against a stationary background of stars… Find the moving object…

Find the line that has more points within some specified tolerance of that line….