

8. Large Data

- *Large n*: Overplotting, slow-down of real-time processes, eg brushing, slow startup.
- *Large p*: Too many variable circles, too many possible views to see.

More information Swayne et al (98).

8 - 1

Some Work Arounds

- Overplotting: use smallest glyph - pixel points.
- Slow startup: Save data in binary format after first read.
- Slow real-time processes:
 - ▶ Brushing: Modify user behavior.
 - ▶ Use `-only` option to look at a subset of the whole data set.
 - ▶ Don't use textured dotplots.
- Too many variable circles: modify the XGobi resource file to make very small variable circles.
- Reduce dimensionality before visualization: Principal components, projection pursuit variable selection.

8 - 2

Linked Brushing with Large Data

Modifying the user behavior and software user interface.

- *Small data*:
 - ▶ drag brush
 - ▶ large glyphs
- *Large data (200 000 cases)*:
 - ▶ jump brush
 - ▶ inactivate the brush during motion
 - ▶ update the brush only after button release
 - ▶ use single pixel glyphs to alleviate overplotting

8 - 3

Tours with Large Data

- *Grand Tour*: efficient, of order n , independent of p . It gets less continuous, and more jerky, as sample size increases.
- *Guided Tour*: Projection pursuit is slow. Need to do dimension reduction off-line.
- *Manual Controls*: Plot updates cannot keep up cursor motion.

8 - 4

Subsetting Tools

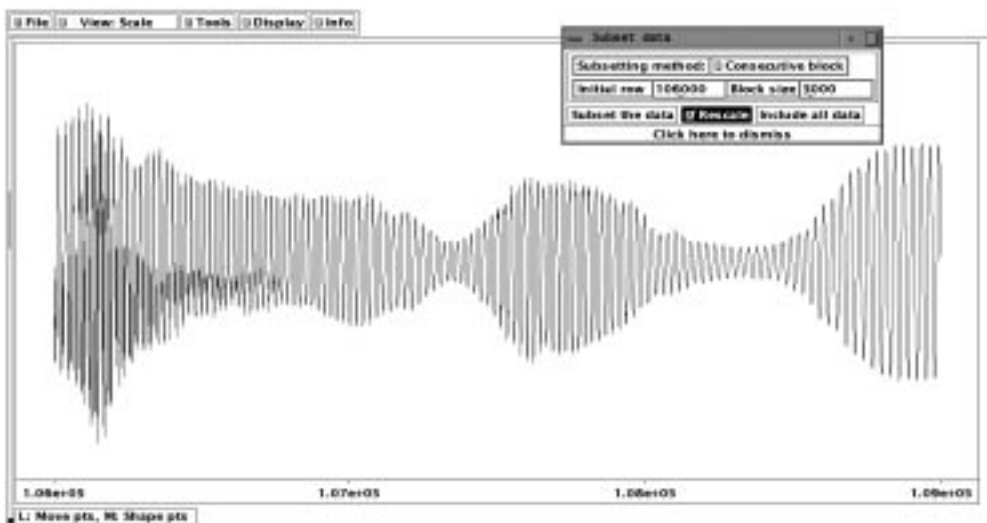
Command line options:

- only n/N Randomly choose n out of N cases
- only a,n Subset n cases starting from case a
- subset n Read in the whole data set but display a random sample of size n .

Subset tool within XGobi allows specifying a contiguous block of cases, random sampling, selecting every n 'th case, selecting all cases chosen in Identify mode, or all cases with the same label.

8 - 5

Subsetting Tools



8 - 6