He is going through the mock mini project: practical 7

Compare analytic and simulation-based approaches in the context of (univariate) KDEs.

See Section 3.3.9 where there is a big derivation of the confidence intervals – see est.density function: analytical application of upper and lower limit of the confidence intervals in 3.3.9

For analytical intervals:

* Section 3.3.9 (Full implementation in example 3.11)

For simulation-based approaches (as in Lab 6) – his **talking points**

* There is a slight modification of est.density in Lab 6 which is combined with the rejection sampler – the output is **cbind(x.grid, est)** – this is used for the grid values in the rejection sampler
* The next step is to do a combined function with est.density and rejection sampler **(we almost did this in lab 6)** – what we need to do in the summative

From the bootstrapped intervals (this has to do with lab 6) – his **whiteboard notes**

* In the mock:
  + Combine the est.density versions so that we get all outputs x.grid, est, lower, upper – see prac7 with solutions for this combination on line 118
  + Use Silverman bandwith because it has a smaller interval – **ask him about this point** – this gives you analytical Cis
* Lab 6 – we already *almost* produced boostrap Cis for KDEs

Mock Solutions:

* Use kde.reject and the chunk after – overlaying densities over the previous estimate – rather than 1 at a time like in Lab 6
* From all of the points, extract all the relevant quantiles to give you a confidence interval – want 2.5% spot and 97.5% spot for each x point – which is what quantiles loop does
* Then you can plot the kde, bootstrap and analytical curves – bootstrap struggles at the peak but they seem to tell the same story

Relevant points from lecture notes:

* Pointwise “percentile” bootstrapped Cis (compare with Section 4.2 of Michaelmas)

Use est.density rather than density because extracting what you want is easier.

Could use apply to construct confidence intervals – it is neater.

Want unimodal data – not bimodal for example.

The mean function is computed different to the median (the 50th% quantile). Bootstrap assesses uncertainty of initial estimate, but it is not used to adjust/ refine your initial estimate.

Notes on Bootstrap

* The original fhat\_h(x) “remains” the estimate
* The bootstrapped mean 1/B \Sum b = 1 to b of fhat^[b](x) is **not** used

Notions used in the practical 7

* If you used bootstrap based on data generated from the **fitted model** (here: KDE) is called parametric bootstrap – misleading because there are no parameters – hence NPS…
* Bootstrap based on resampling from the original data – {x1,…,xn} - (as in Example 3.2 of 1st term NPS notes) is also called non-parametric bootstrap – this is the same as sampling data from the empirical distribution function - edf (see algorithm: bootstrap estimation of variance)

Both of the above methods are valid to generate percentile confidence intervals.

He goes back to practical 7 solution using nonparametric bootstrap (i.e. resampling from the edf):

* This process is much easier to generate our KDEs and overlay them
* And, hence generate the analytical and simulation-based CIs

In the practical – the bootstrap (edf) is a middle ground between the analytical and normal bootstrap. There isn’t a uniform way of saying whether bootstrap edf or bootstrap is better or worse.

Bootstrap edf can go wrong with instable data – i.e. with more influential values which are picked more often.

**Hints for how this relate to bootstrap in localised regression for the assignment:**

* Take Yi = m(xi) + \epsiloni with data (xi,yi), i = 1,…,n

We can use either of these bootstrap methods

* See lecture 5 of Michaelmas Dinos Lecture notes – “Nonparametric paired bootstrap” (based on edf) which you can use similarly here (resample pairs (xi,yi),…)
* “Parametric/ semiparametric bootstrap”
  + \hat{Y}\_i ^ [b] = mhat(xi) + epsilon\_hat\_i, mhat(xi) comes from the fitted model
  + To get epsilon\_hat\_i there is some creativity needed example – lecture 6 notes from Dinos – page 25 bottom of page – semiparametric residual bootstrap
  + epsilon\_hat\_i = \hat{Y}\_i ^ [b] - mhat(xi), then you can reshuffle them