Hand-in exercise

```
library(spatstat)
```

In this exercise we work with the `spatstat` dataset `cells`. First we show that a Poisson model is clearly inadequate and then we investigate whether a determinantal point process (DPP) model is a good fit.

Part 1

- Plot the `cells` dataset and briefly (1-2 lines) argue why you don't think a Poisson model is adequate for this data.
- Generate 1999 realisations from a uniform point process model with 42 points in the unit square (use `runifpoint()`) and save them as `sims`.
- Use the generated simulations (`sims`) as the `simulate` argument in `envelope.ppp()` along with your choice of summary statistic (argument `fun`) to make envelopes for the `cells` data (also specify `savefuns = TRUE`). Save the envelope as `env`.
- Load the `GET` package and make the global rank envelope test using the envelope `env` you generated above and plot the result.

Part 2

- Fit a Bessel DPP to the `cells` data using `dppm()` (*hint:* `dppBessel()` is useful).
- Generate 39 realisations from the fitted Bessel DPP model model in the unit square (use `simulate()`) and save them as `sims2`.
- Use the generated simulations (`sims2`) as the `simulate` argument in `envelope.ppp()` with argument `fun = pcf` to make envelopes for the `cells` data. Save the envelope as `env2`.
- Plot the envelopes and add the theoretical pcf for this DPP model to the plot (*hint*: extract pcf with `pcfmodel()` and add to existing plot with `plot(, add = TRUE)`).
- Does the mean pcf from the simulations agree with the theoretical pcf? Does the empirical pcf?
- (Optional) Make global envelopes based on `nsim = 39` simulations. Try both without specifying `ginterval` and with `ginterval = c(.02,.25)`. What is the difference?
- (Optional) Try to (briefly) explain the large value of g(r) around r = 0.15.