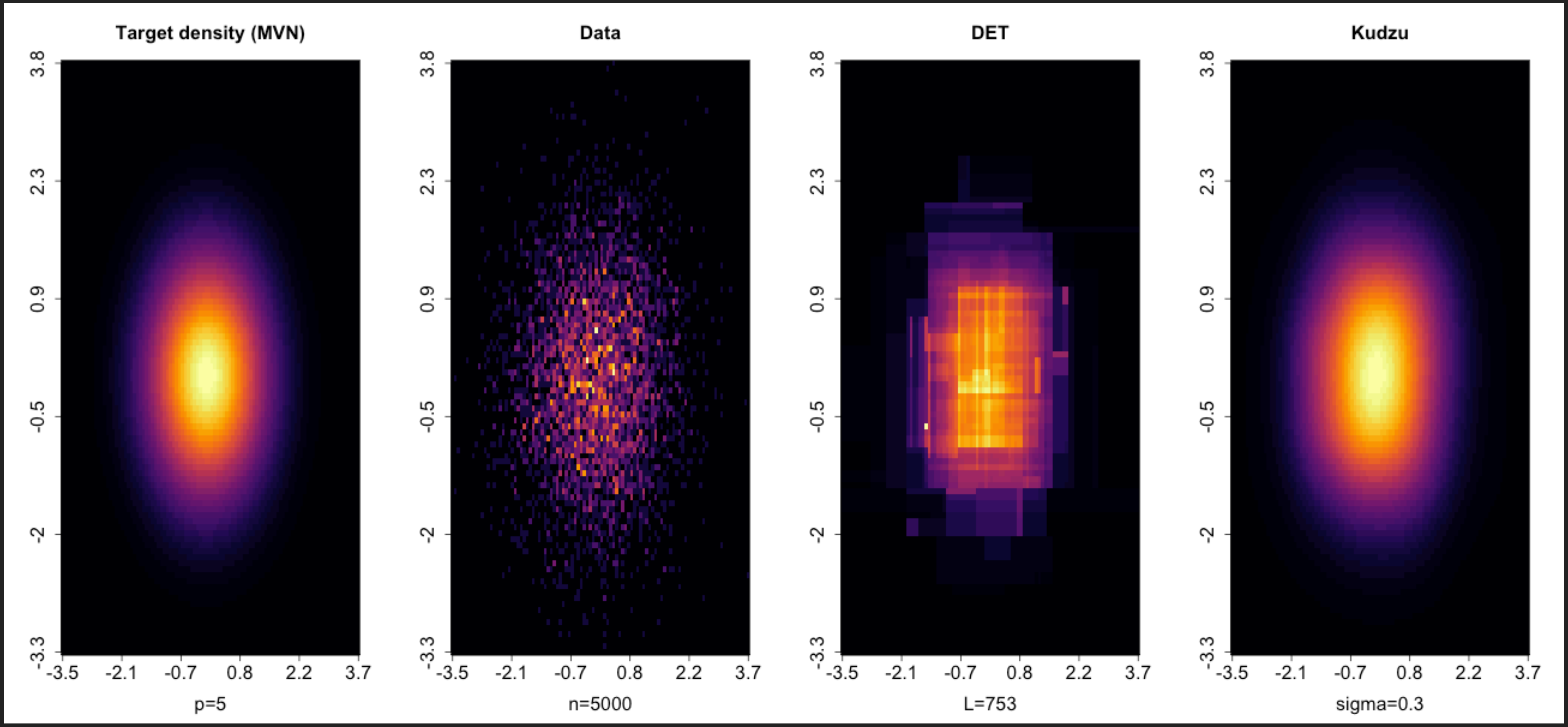


ROBERT GRANT — BAYESCAMP & KINGSTON UNIVERSITY, UK

SCALABLE HIGH-DIMENSIONAL NON-PARAMETRIC DENSITY ESTIMATION, WITH BAYESIAN APPLICATIONS

DENSITY ESTIMATION

- Our goal: turn a dataset into a probability density function
- The PDF should be smooth
- The method should work for many variables / dimensions
- Extant methods are either parametric, or limited to few dimensions (*p*), or computationally intensive (i.e. bad for the planet), and forbidding to most users

MOTIVATION

- Density estimation is useful for probabilistic prediction, visualisation, simulation, etc.
- My original motivation was Bayesian updating:
  - Model fitted on data X_1, giving a posterior sample from $P(\theta \mid X_1)$.
  - Now data X_2 arrives. Re-analysis of (X_1, X_2) will take too long.
  - Use X_1's posterior $P(\theta \mid X_1)$ as the prior, and compute likelihood only on X_2. No guarantee of a convex, unimodal p[oste]rior, so we need a non-parametric method. *p* could be large.

CORE IDEA

- Fit a density estimation tree (DET; Ram & Gray 2011) and smooth it.
- This produces a kudzu density function, named after a vine (a.k.a. Japanese arrowroot) which grows rapidly over trees, smoothing out their shape.

クズ
kudzu

- Thanks: Gordon Hunter, Lucio Morettini

DENSITY ESTIMATION TREES

- Proposed by Ram & Gray (2011) with little subsequent uptake.
- CART algorithm, but using integrated squared error (ISE), which controls over-fitting to some extent (and smoothing helps too).
- Trees scale well to high n and high p (compute time and accuracy).
- Terminal nodes of the tree are L "leaves". The tree is defined by two L-by-p matrices (top and bottom edges) and a L-vector of densities.
- ISE collapses to an extremely simple formula for DETs.

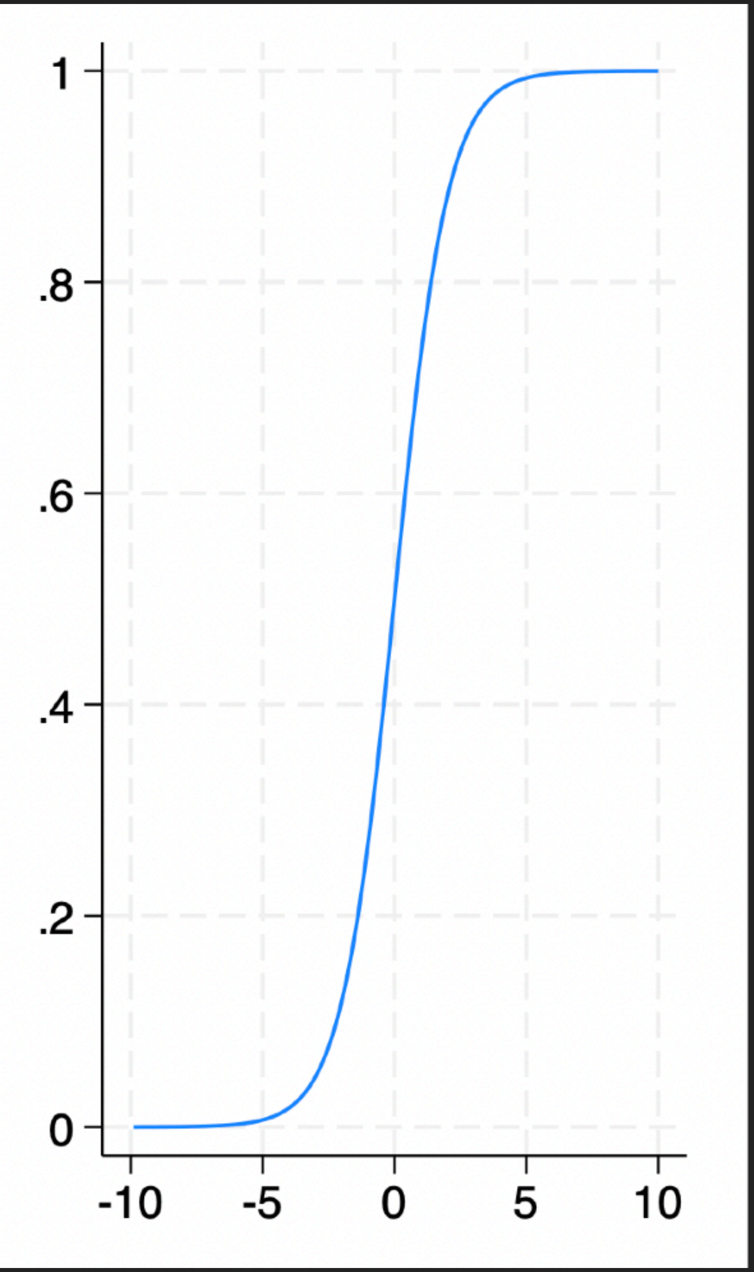
😱 (Technical aside: The volume of the leaf is important in DET, so the p variables/unknowns must jointly define a metric space: no ordinal or nominal variables, though integers are fine.)

KUDZU DENSITY FUNCTION (1)

$$\frac{h}{1+e^{-\frac{x-\mu}{\sigma}}}$$

- We replace each edge of each leaf with a smooth ramp (monotonic, two horizontal asymptotes).
- They are centred on the edges, and have bandwidth σ. Inverse logistic function is in the class of computationally minimal smooth ramps (one power series).

😱 (Technical aside: you can also conceive of it as a convolution of each leaf-dimension with the logistic PDF.)



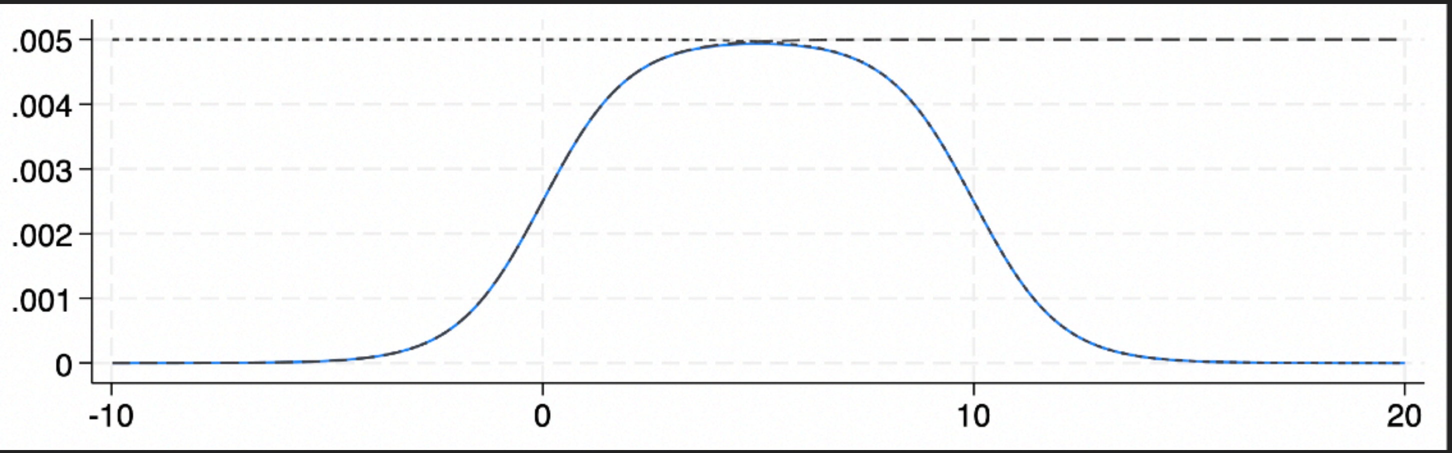
KUDZU DENSITY FUNCTION (2)

- Each dimension of each leaf is the product of the top and bottom ramps, and the predicted DET density

$$\hat{f}_{\text{kudzu}}(x_j|\ell) \propto \hat{f}_{\text{DET}}(x_j|\ell)\left(\frac{1}{1+e^{\frac{\mu_{b\ell j}-x_j}{\sigma}}}\right)\left(\frac{1}{1+e^{\frac{x_j-\mu_{t\ell j}}{\sigma}}}\right)$$

- Each leaf is the product of these *p* dimensions (which are orthogonal and independent)

$$\hat{f}_{\text{kudzu}}(\boldsymbol{x}|\ell) \propto \hat{f}_{\text{DET}}(\boldsymbol{x}|\ell)\prod_{j=1}^{p}\left(\frac{1}{1+e^{\frac{\mu_{b\ell j}-x_j}{\sigma}}}\right)\left(\frac{1}{1+e^{\frac{x_j-\mu_{t\ell j}}{\sigma}}}\right)$$



KUDZU DENSITY FUNCTION (3)

- The whole tree is the sum of the leaves

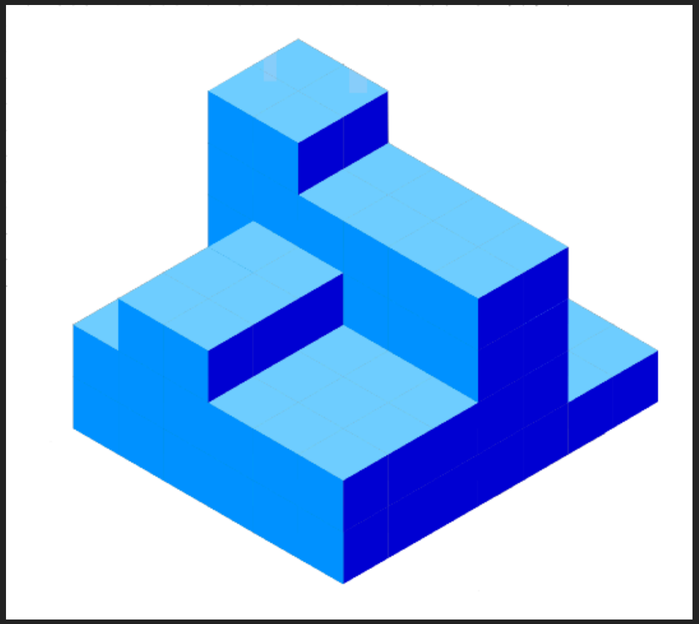
$$\hat{f}_{\text{kudzu}}(\boldsymbol{x}) \propto \sum_{\ell=1}^{L} \hat{f}_{\text{DET}}(\boldsymbol{x}|\ell) \prod_{j=1}^{p} \left(\frac{1}{1+e^{\frac{\mu_{b\ell j}-x_j}{\sigma}}} \right) \left(\frac{1}{1+e^{\frac{x_j-\mu_{t\ell j}}{\sigma}}} \right)$$

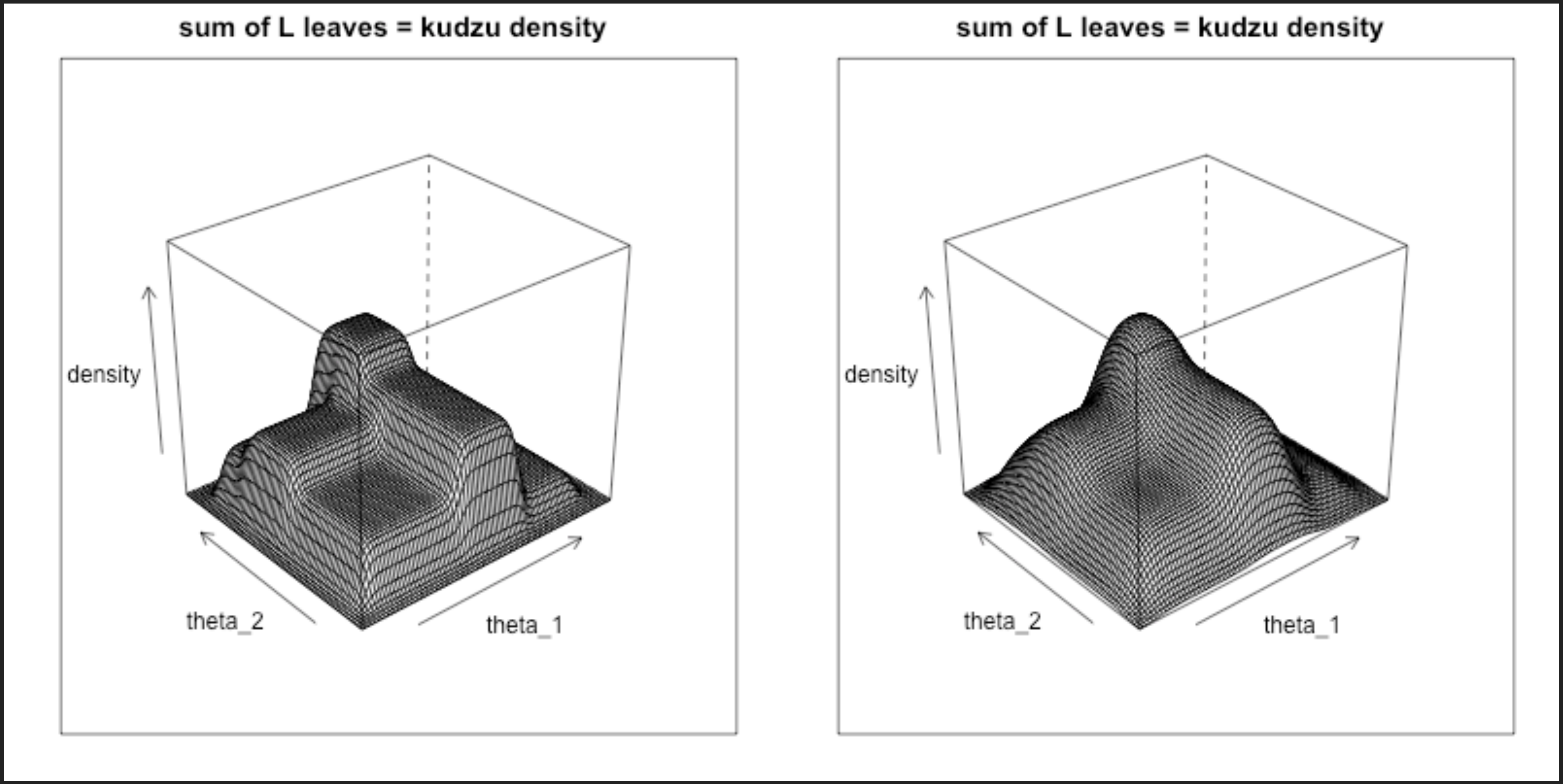
- But it might not integrate to one, so can optionally be normalised by dividing by the definite integral out to ±ϕσ (beyond which it is negligible).

$$\hat{f}_{\text{kudzu}}(\boldsymbol{x}) = \frac{\sum_{\ell=1}^{L} \hat{f}_{\text{DET}}(\boldsymbol{x}|\ell) \prod_{j=1}^{p} \left(\frac{1}{1+e^{\frac{\mu_{b\ell j}-x_j}{\sigma}}} \right) \left(\frac{1}{1+e^{\frac{x_j-\mu_{t\ell j}}{\sigma}}} \right)}{\sum_{\ell=1}^{L} \hat{f}_{\text{DET}}(\boldsymbol{x}|\ell) \prod_{j=1}^{p} \sigma \frac{e^{u_{\ell j}}}{e^{u_{\ell j}}-1} \ln \left(\frac{e^{u_{\ell j}+\phi}+e^{-(u_{\ell j}+\phi)}+2}{e^{\phi}+e^{-\phi}+2} \right)}$$

- When we integrate, we store all the leaf integrals and leaf-dimension integrals.

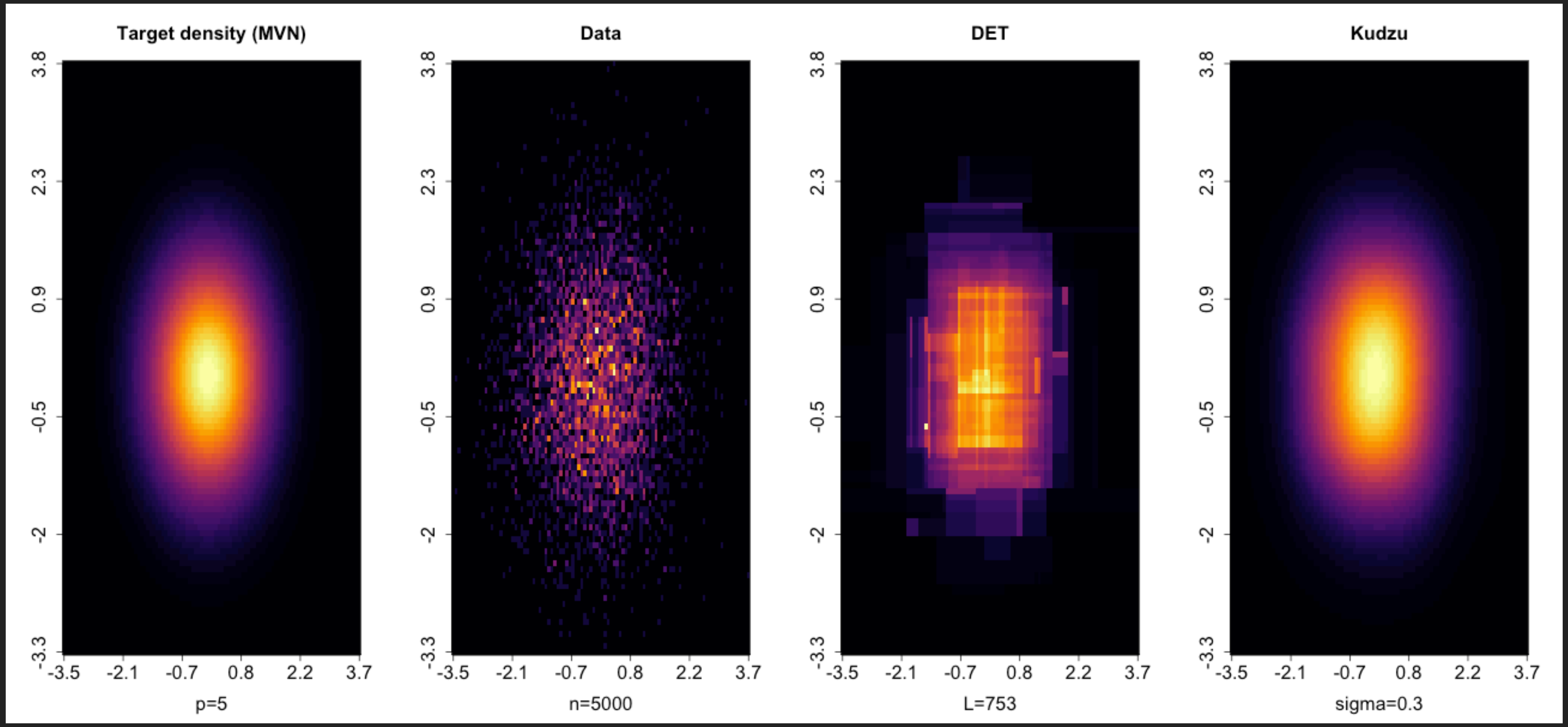
KUDZU DENSITY FUNCTION (4)





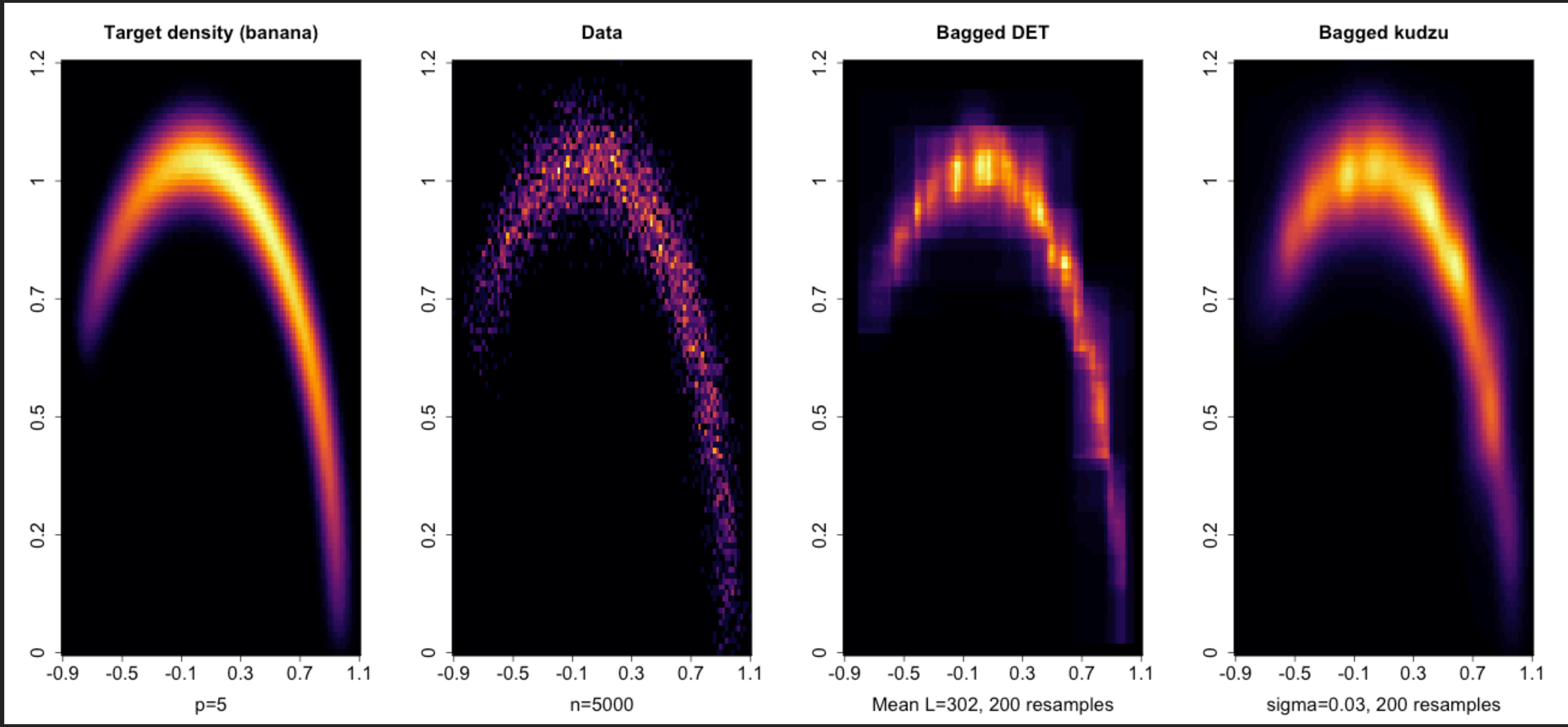
PERFORMANCE

- DET fit time is $O(p)$ and approximately $O(n \log n)$. Density evaluation is very fast with maximum $2Lp$ power series. We only evaluate neighbouring leaves and use stored integral components for marginalisation.



ENSEMBLES

- Trees struggle with shapes that cannot line up with the axes.
- Ensembles of kudzu density functions are promising, and I have implemented bagging so far.



STATA / MATA IMPLEMENTATION

- Sharing for alpha testing at robertgrantstats.co.uk/kudzu
- `webuse iris, clear`
- `kudzu_tree seplen-petwid, minnl(5) maxnodes(10)`
- [… boring matrix storage omitted …]
- `kudzu_predict, kmat("K") at(6.0 3.1 6.8 0.4)`
- `kudzu_rng, kmat("K") n(1000)`
- Export BUGS/JAGS, Stan and bayesmh evaluator code for p[oste]rior.

FIND OUT MORE

- Thank you for listening. Follow progress at robertgrantstats.co.uk/kudzu
- References:

  - P Ram & A Gray (2011). "Density estimation trees", KDD '11: Proceedings of the 17th ACM SIGKDD international conference on knowledge discovery and data mining. pp. 627–635.
  - L Breiman et al (1984). "Classification and regression trees".
  - DW Scott (2015). "Multivariate density estimation: theory, practice, and visualization." Wiley.

- **robert@bayescamp.com**

- (By the way, I'm job hunting for 2025.)