Matching Random Features

Cosma Shalizi (CMU Statitics & Machine Learning)

23 June 2022

An apology

The S key on my laptop is dying, and I can't spell

Summary

- Most simulation-based inference shares a basic pattern:
  Twist the knobs until the simulations look like the data in ***this*** *respect*
- Deciding *what aspects* of the data to match demands a lot of human time
- I think that *decided what to match* can be largely automated:
  Matching the values of functions chosen ***at random*** *from a rich enough basis is informative and (maybe) efficient*
- In fact, I hope to convince you that
  For nice d-parameter models, we match at most $2d+1$ *random functions*
- (You can now resume checking e-mail)

Agenda

- Convince you there's a problem
- Sketch a solution
- Show you some pictures
- Hand-wave at the glorious future

How simulation-based inference usually works (1)

- Scientist shows up with data $x_{1:n}$
- They also show up with a model for $X_{1:n}$, distribution $P_{n,\theta}$, $\dim(\theta) = d$
- Scientist can *simulate* it from any θ, as many times as desired, getting $\tilde{X}_{1:n}(\theta)$
- Good luck *calculating* $p(x_{1:n};\theta)$, let alone maximizing it
- Instead, we pick some **features**, i.e., statistics, say $f = (f_1, \ldots f_k)$
- We *hope* that a couple of convergences happen as $n \rightarrow \infty$:

$$P_{n,\theta}f \equiv \int f(x_{1:n}) dP_{n,\theta}(x_{1:n}) \quad \rightarrow \quad \phi(\theta) \ (\textit{stability}) \tag{1}$$

$$f(X_{1:n}) \quad \overset{P}{\rightarrow} \quad \phi(\theta) \ (\textit{LLN}) \tag{2}$$

$$\tag{3}$$

- We *hope* that ϕ^{-1} exists

How simulation-based inference usually works (2)

- The ideal would be: observe $X_{1:\infty} \sim P_{\infty,\theta}$, so
$$\theta = \phi^{-1}(f(X_\infty))$$
- We have $X_{1:n}$ instead of X_∞ so
$$\theta^{\dagger\dagger\dagger} = \phi^{-1}(f(X_{1:n}))$$
- We don't know ϕ^{-1} and anyway $f(X_{1:n})$ might be outside its domain so
$$\theta^{\dagger\dagger} = \underset{\theta\in\Theta}{\mathrm{argmin}} \|f(X_{1:n}) - \phi(\theta)\|^2$$
- We don't know ϕ so
$$\theta^{\dagger} = \underset{\theta\in\Theta}{\mathrm{argmin}} \|f(X_{1:n}) - P_{n,\theta}f\|^2$$
- We don't know $P_{n,\theta}f$ so we Monte Carlo it:
$$\hat{\theta} = \underset{\theta\in\Theta}{\mathrm{argmin}} \left\|f(X_{1:n}) - \frac{1}{s}\sum_{i=1}^{s} f(\tilde{X}^{(i)}(\theta))\right\|^2$$
- $\hat{\theta}$ is a feasible, simulation-based estimator
  - Factor of $(1 + 1/s)$ in the asymptotic variance of $\hat{\theta}$

How simulation-based inference usually works (3)

- $k \geq d$ or else there's no inverse ϕ^{-1}
  - Peano curves need not apply
- We want ϕ^{-1} to be smooth, so noise in $f(X_{1:n})$ doesn't wreck us
- Method of simulated generalized moment: f_i come from genuine or generalized moments
- Indirect inference: f_i are point estimates of the parameters of an auxiliary model
- ABC: summary statistics work similarly
  - Won't compare to all the strategies for summary-statistic selection here but see the paper
- In every case, we want $\frac{\partial\phi_i}{\partial\theta_j}$ to be large

Picking which features to match is a *pain*

- Ideal f: a d-dimensional sufficient statistic
  - If you have that, why are you doing simulation-based inference?
- Bad features are insensitive to the generative-model parameters, $\partial\phi_i/\partial\theta_j \approx 0$
- Feature sources: tradition, analogy, understanding, omnivorousness, trial and error
  - Tradition: "worked in my last paper"; "my adviser said it worked for him"; "I think I saw it in JRSS-A once"; "heard about it at JSM"; "StackOverflow recommends"
  - Analogy: "this is *almost* like the model in the JRSS-A paper, but…"
  - Understanding: "there should be heavy tails, so estimate a Pareto exponent"; "we care about the dynamics, so estimate an AR(k)"
  - Omnivorousness: throw *everything* in and hope *something* works (Wood 2010)
  - Trial and error: tinker until you can get the estimator to converge on your own simulation output
- Easily one of the most *human*-time-consuming parts of any SBI project

What we'd like

- A way of *automatically* picking features
- without much understanding of the model
- leading to statistically-efficient estimates
- with not too many features

The proposal

- Start with a d-parameter model, where Θ is a d-dimensional manifold, and $\theta \rightarrow \theta^*$ implies $P_{n,\theta} \overset{d}{\rightarrow} P_{n,\theta^*}$
- Pick $k = 2d+1$ random smooth functions from a rich enough set $\mathcal{F}$, say $F = (F_1, \ldots F_k)$
    - In fact $P_{n,\theta}f$ should be C^1 in θ for $f \in \mathcal{F}$
- Pick $s \geq 1$ simulations per parameter value
- Estimate:

$$\hat{\theta} = \underset{\theta \in \Theta}{\operatorname{argmin}} \left\| F(X_{1:n}) - \frac{1}{s}\sum_{i=1}^{s} F(\tilde{X}^{(i)}(\theta)) \right\|^2$$

- Why should this work at all?

Random features in machine learning

- Fix a nice-enough function class $\mathcal{F}$, say $f(x;\omega)$
    - Think: Fourier basis functions (but many others can work)
- We want to approximate functions of the form

$$m(x) = \int a(\omega) f(x;\omega) d\omega$$

    - Think: functions with nice Fourier transforms
- Pick Ω_i iidly from a distribution ρ
- Then with high probability, a function of the form

$$\sum_{i=1}^{k} a_i f(x;\Omega_i)$$

gives an $O(1/\sqrt{k})$ approximation, in L_2 and L_∞, to $m(x)$ (Rahimi and Recht 2008)
- If $\omega = (w,b)$ and $f(x;\omega) = \cos x \cdot w + b$, then we can get an $O(1/\sqrt{k})$ approximation to any function with a nice Fourier transform by a linear combination of k random Fourier features
- Random features have turned into an *incredibly* useful technique for ML

Limiting the number of random features

- Error of $O(1/\sqrt{k}) \Rightarrow k = O(\epsilon^{-2})$
- How *few* features k could we get away with?
- Clearly $k \geq d$ is required
- Could we get $k = O(d)$?

A wonderful old result in differential geometry / topology

- Say M is a d-dimensional manifold
- Consider maps $\phi : M \mapsto \mathbb{R}^k$
- Say each coordinate $\phi_i : M \mapsto \mathbb{R}$ is C^1
- ϕ is an **embedding** if ϕ^{-1} exists and is also C^1 (so ϕ is a **diffeomorphism**)
- **Theorem** (Whitney 1936): Once $k = 2d$, an embedding exists; once $k \geq 2d+1$, the **generic** or **typical** C^1 mapping is an embedding
  - That is, embeddings are an open, dense set
  - Topological equivalent of "almost everywhere"

Applying the embedding theorem

> Suppose $\mathcal{F}$ is a nice class of bounded continuous functions, and we sample $F_1, \ldots F_k$, collectively F, from $\mathcal{F}$ according to ρ which is supported everywhere on $\mathcal{F}$. Suppose that $\forall \theta, \forall f \in \mathcal{F}$, $\lim_{n\to\infty} P_{n,\theta}(f)$ exists. Define $\Phi(\theta) = \lim_{n\to\infty} P_{n,\theta} F$. Then generically Φ^{-1} exists and is C^1, so long as $k \geq 2d+1$.

- We can smoothly translate between θ and the (limiting) expectation values of $2d+1$ test functions

- $\Rightarrow$ Try to do inference by matching the values of $2d+1$ random test functions!

Sample convergence

- We don't get to see expectation values, so what we really want is

$$f(X_{1:n}) \to P_{n,\theta}(f)$$

  for all $f \in \mathcal{F}$
- i.e. some sort of law of large numbers or ergodic theorem
- A concentration-of-measure or large-deviations result would be ideal
- For lots of processes, time (or space) averages over short blocks will be well-behaved *and* informative

Putting the pieces together: time series

- Pick nice $\mathcal{F}$: bounded, continuous, rapidly-ergodic test functions over blocks of length (say) m whose span defines a big function space, and a distribution ρ over $\mathcal{F}$
  - For vector-valued time series, random-frequency cosines over blocks of length m
- Start with a d-dimensional manifold of models
- Sample $F = (F_1, \ldots F_{2d+1})$ iidly from ρ
- Observe $X_{1:n}$
- Minimize $\left\| F(X_{1:n}) - \frac{1}{s}\sum_{i=1}^{s} F(\tilde{X}^{(i)}(\theta)) \right\|^2$

Does it work?

- Four increasingly tricky cases:
  1. Estimate θ when $X \sim \mathcal{N}(\theta, 1)$, IID
  2. Estimate the location of a t distribution with the *same* random features
  3. Estimate r in the logistic map, $x_t = 4x_{t-1}(1 - x_{t-1})$
  4. Estimate r and σ when the logistic map with Gaussian observational noise

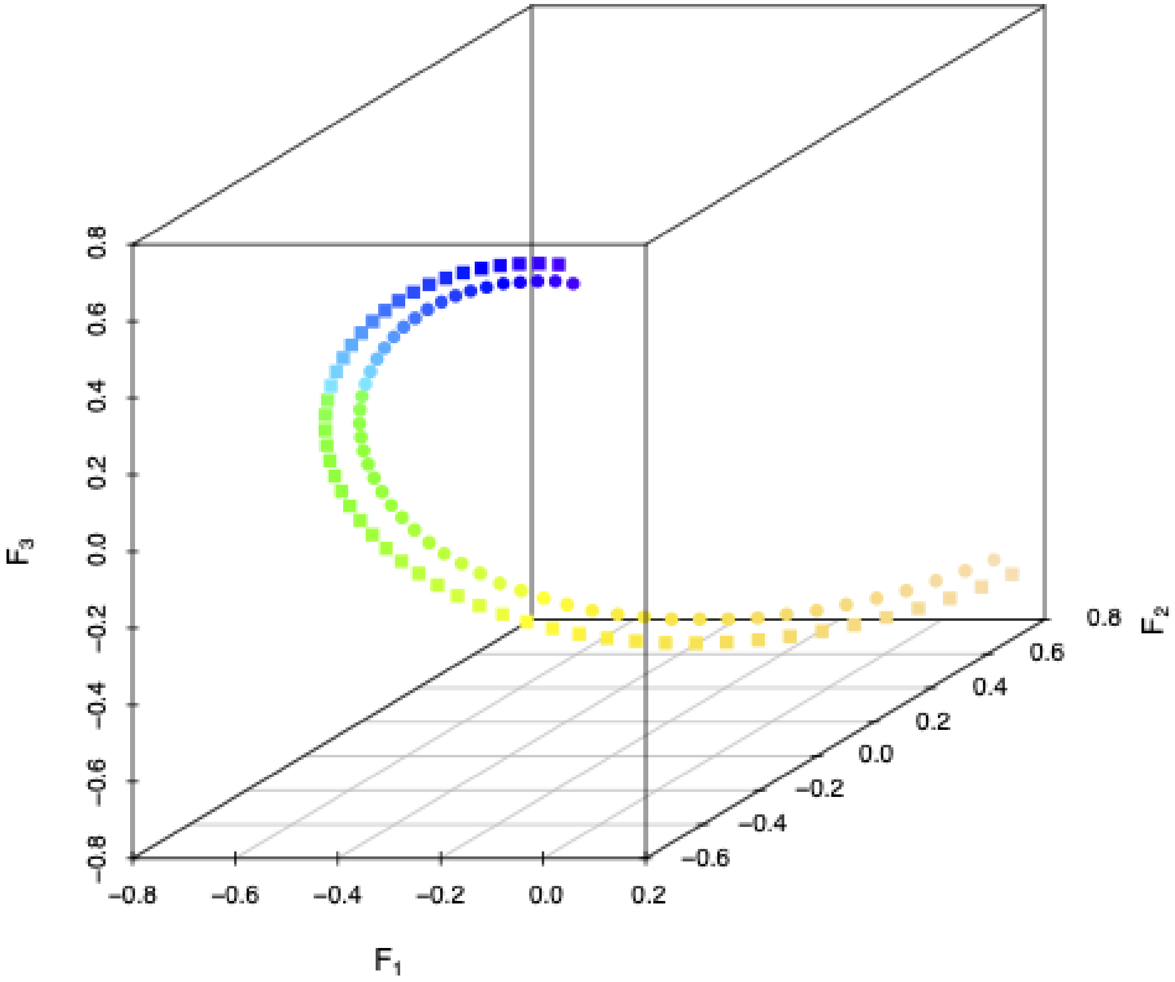
IID Gaussian Location

- $X_t \sim_{IID} \mathcal{N}(\theta, 1)$
- "Baby's first inference problem": If the method can't do *this*, forget it
  - Obviously it can do this or I wouldn't be showing it to you!
- $d = 1$ so use $k = 3$ random univariate Fourier features, i.e.,

$$F_i(X_{1:n}) = \frac{1}{n}\sum_{t=1}^{n} \cos\left(X_t W_i + B_i\right)$$

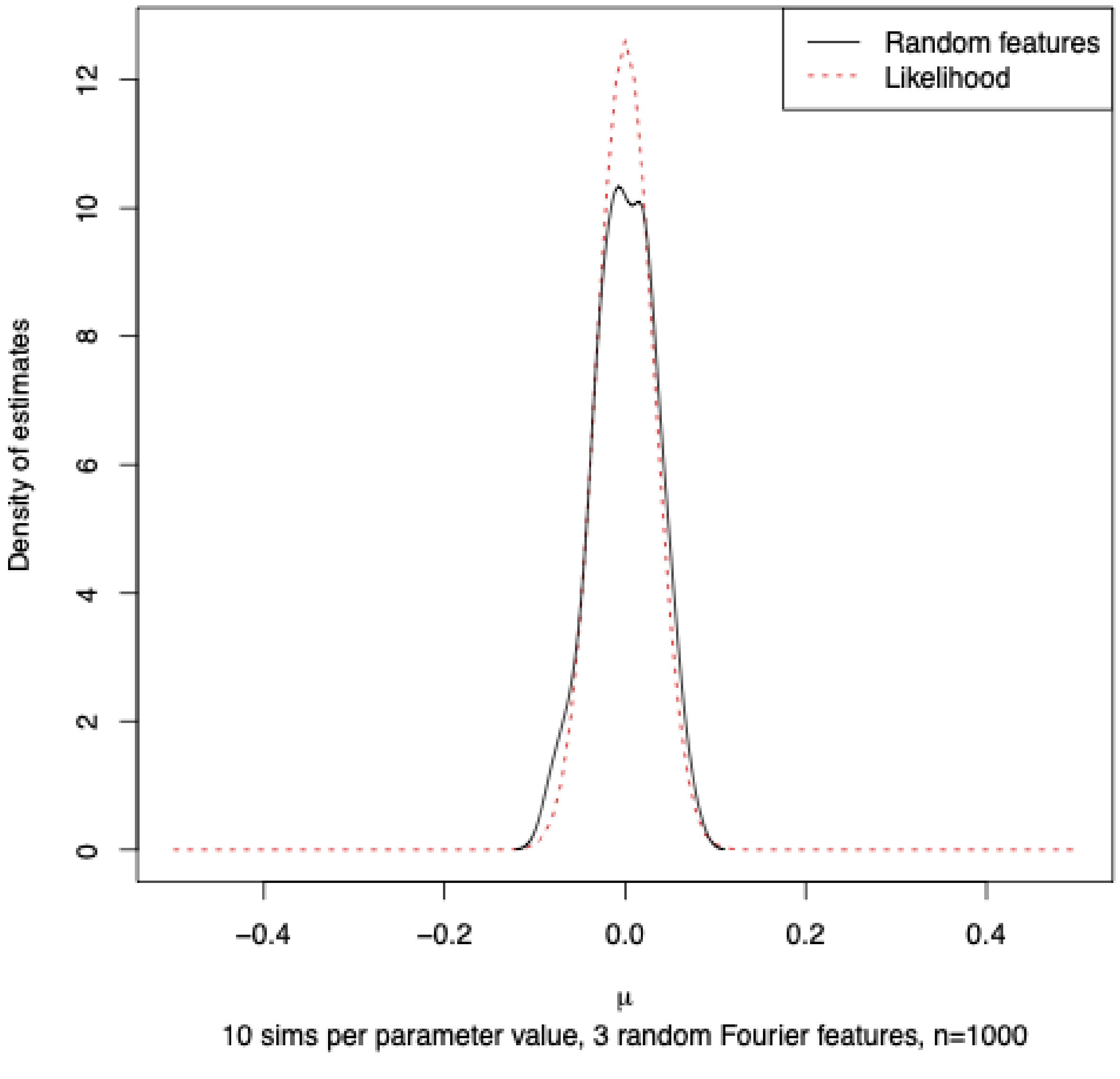
with $W_i \sim \mathcal{N}(0,1)$ and $B_i \sim \mathrm{Unif}(-\pi, \pi)$

IID Gaussian Location (2)



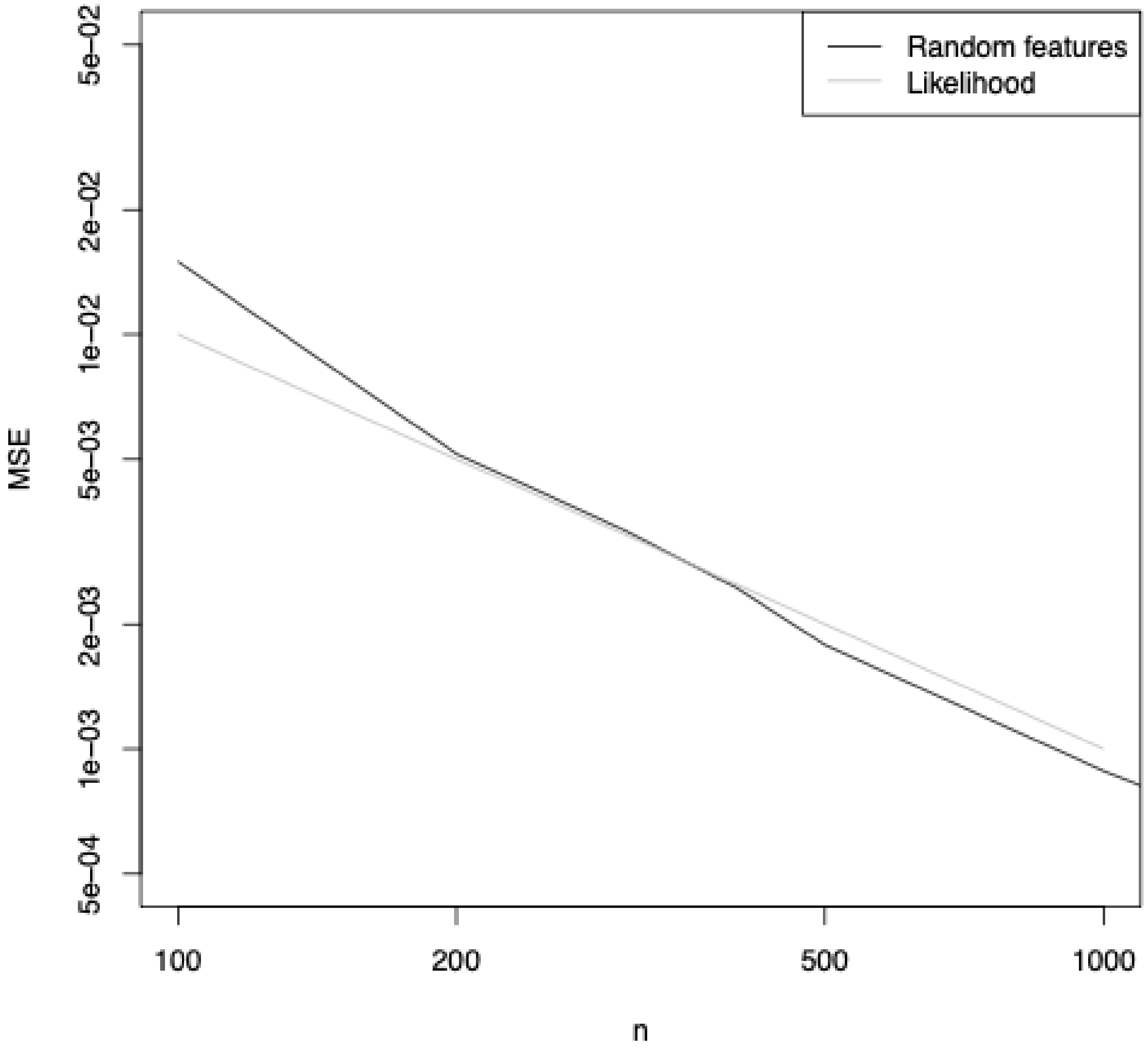
Values of 3 random Fourier features (axes) as we vary the location θ of $\mathcal{N}(\theta, 1)$ (color), for two different $n = 100$ noise realizations

IID Gaussian Location (4)



Sampling distribution (smoothed) from matching the 3 random Fourier features, vs. theoretical distribution of the MLE

IID Gaussian Location (5)

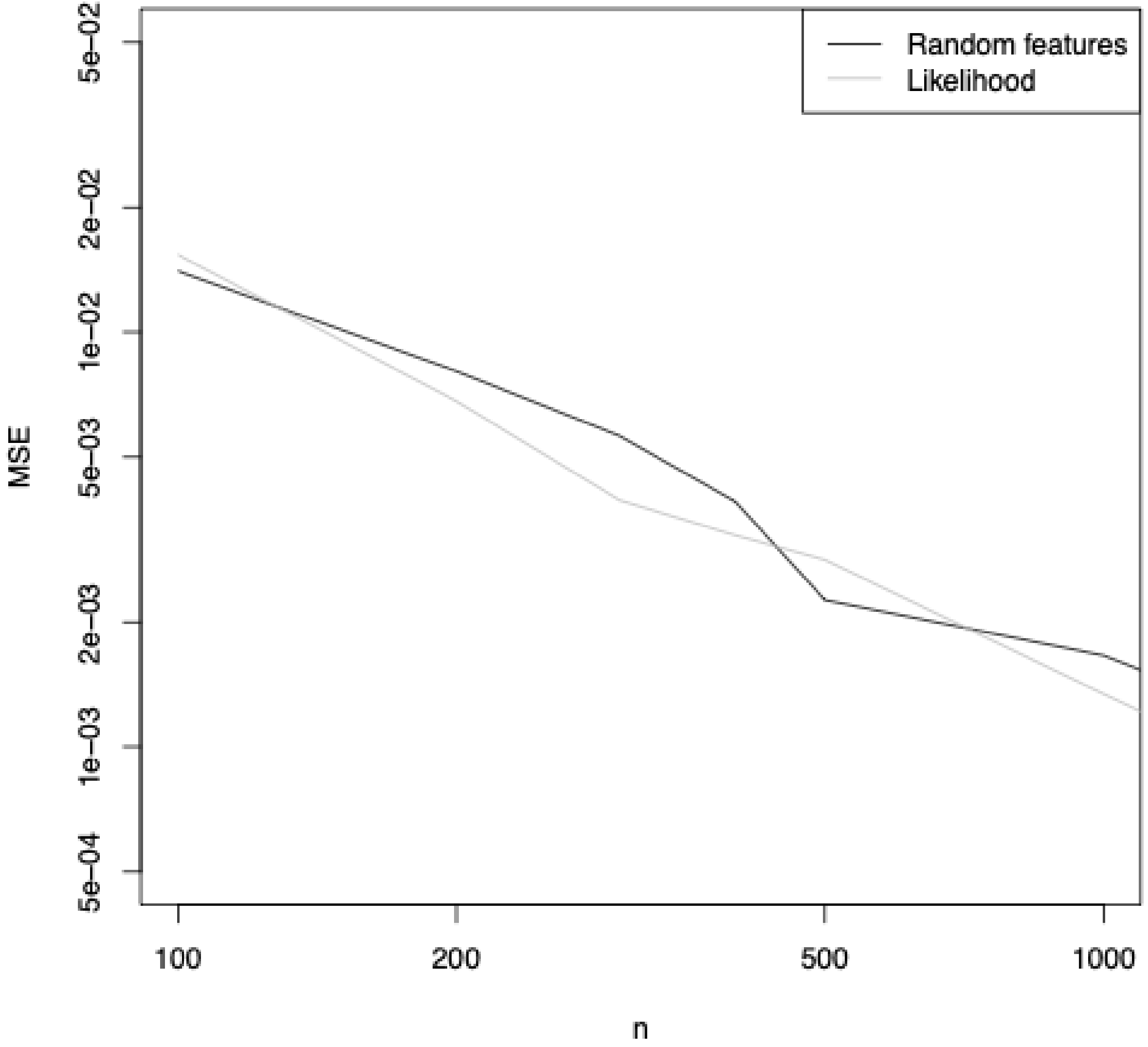


MSE of $\hat{\theta}$ from matching the 3 random Fourier features, vs. theoretical MSE of the MLE. 100 replicate per sample size, $s = 10$ per parameter value.

IID t-Distribution Location

- $X_t = \theta + Z_t$ where $Z_t \sim t_5$, a t-distribution with 5 degrees of freedom
- Also a one-parameter location family, but heavy-tailed
- I use the *same* 3 random Fourier features that I did for the Gaussian

IID t-Distribution Location (2)



MSE of $\hat{\theta}$ from matching the ***same*** *3 random Fourier features as in the Gaussian, vs. numerical MLE. 100 replicates per sample size, $s = 10$ per parameter value.*

(numerical likelihood maximization using `MASS:fitdistr`)

Logistic Map: A Chaotic Dynamical System

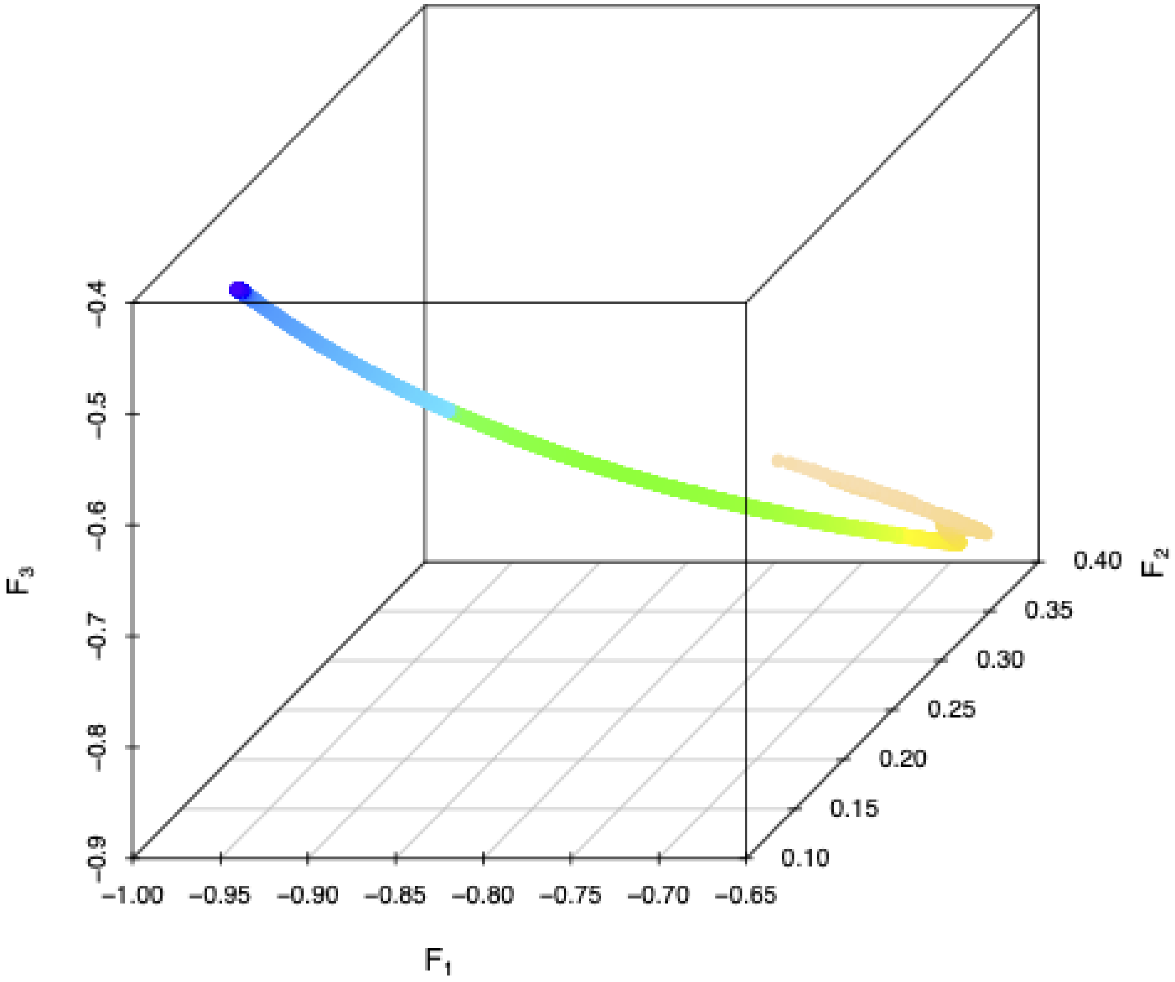
$$X_0 \sim \mathrm{Unif}(0,1) \quad (4)$$

$$X_{t+1} = 4rX_t(1-X_t) \quad (5)$$

- A *deterministic* dynamical system (extensive theory; see Devaney (1992))

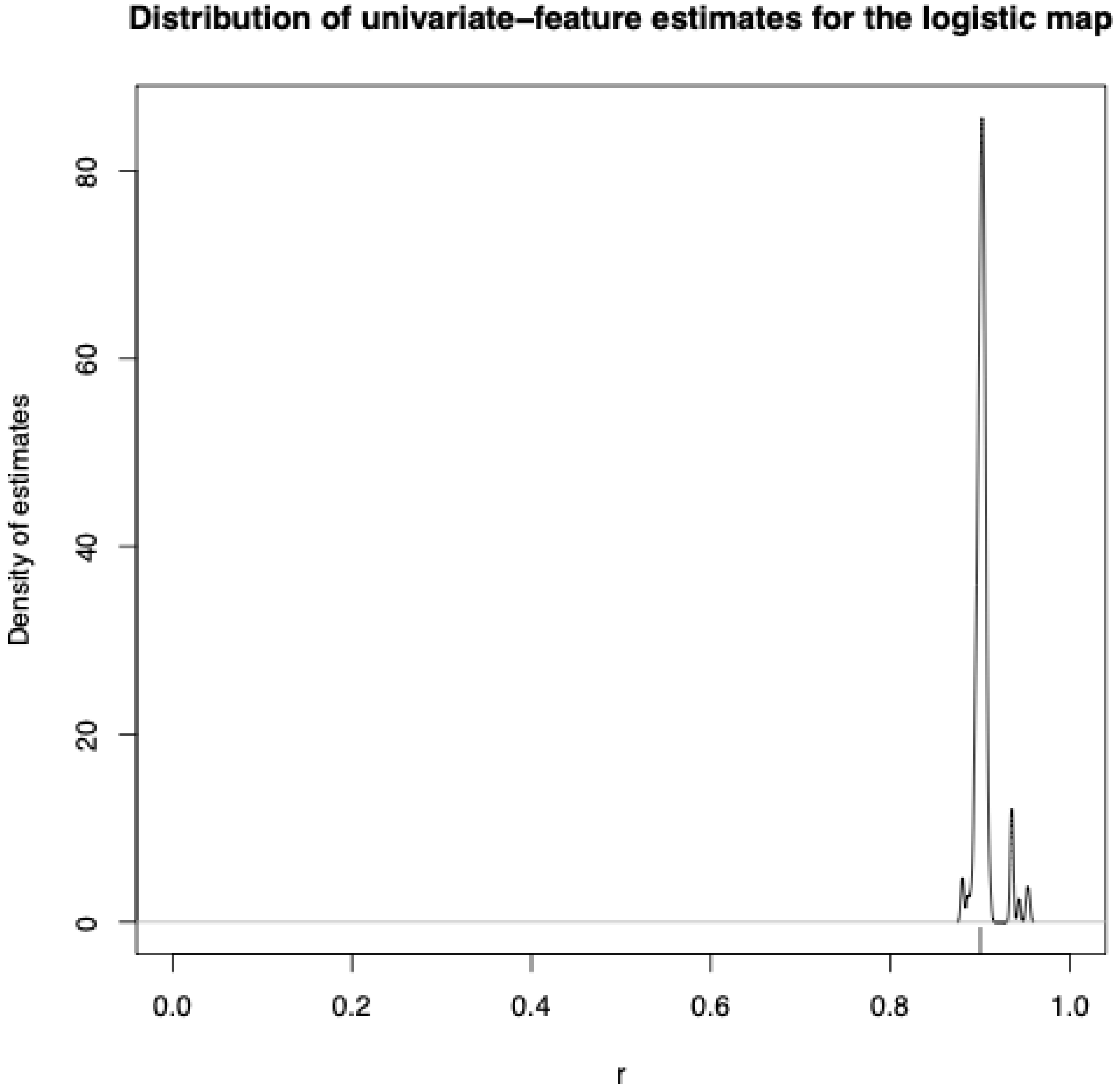
- Rapidly goes to an attractor $\subset [0,1]$
  - Strictly speaking, non-stationary because Lebesgue measure isn't invariant
- For large enough r, **chaotic**: sensitive dependence on initial conditions
  - In particular, $r = 0.9$ is chaotic
- Likelihood and the MLE are not useful here
  - Any particular trajectory is either certain (given initial condition) or impossible

Logistic Map (2)



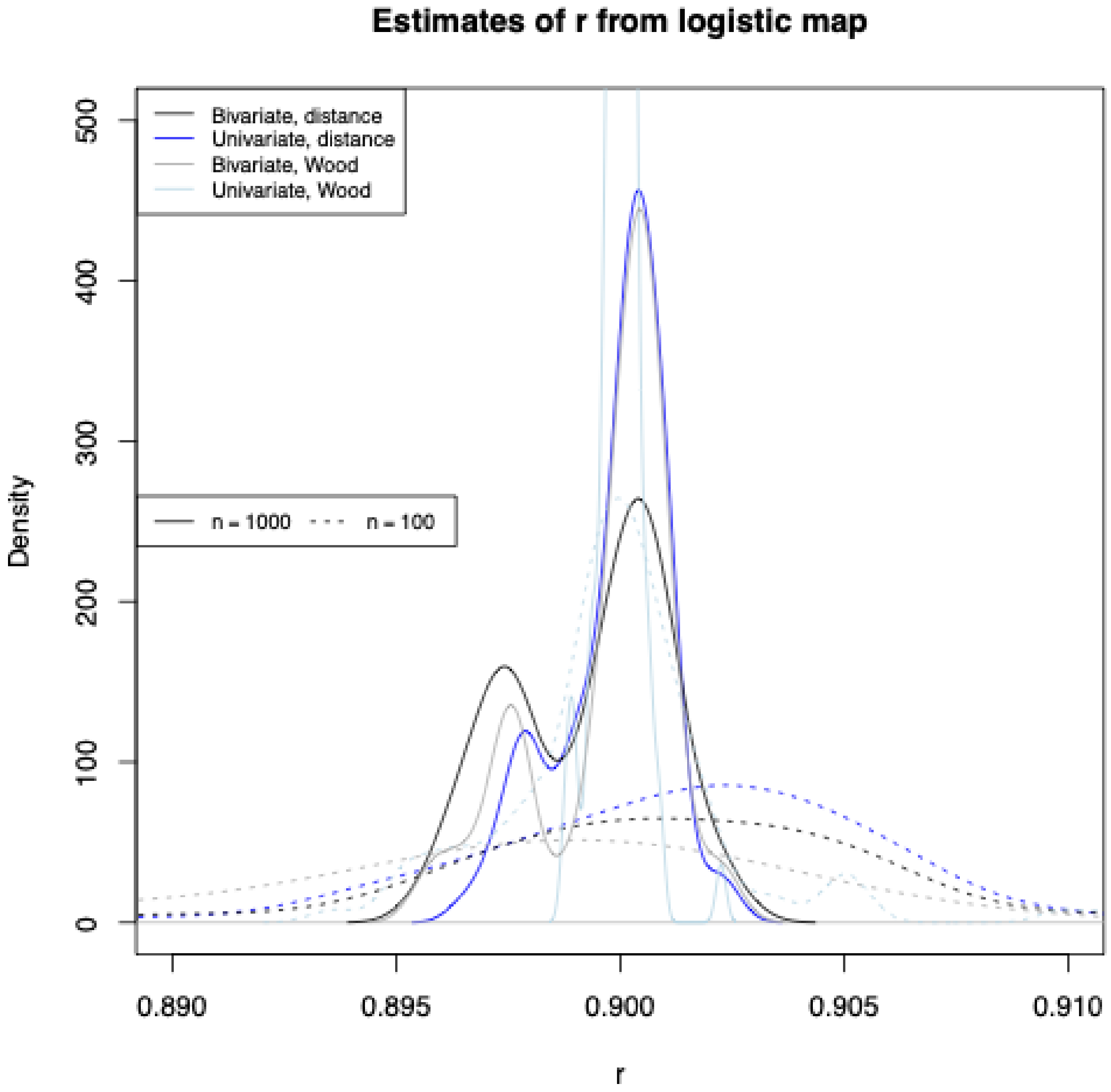
With the same *three univariate random Fourier features as the previous examples (averaging $s = 10$ trajectories of length $n = 100$ per r value)*

Logistic Map (3)



Distribution of estimates using the 3 univariate random Fourier features when $n = 100$, with $s = 10$ per parameter value, true $r = 0.9$

Logistic Map (4)



Distribution of estimates using both univariate (as before) and bivariate random Fourier features, but at $n = 100$ *(as before) and* $n = 1000$*, and comparing simple distance minimization with the Wood (2010) sorta-kinda-likelihood*

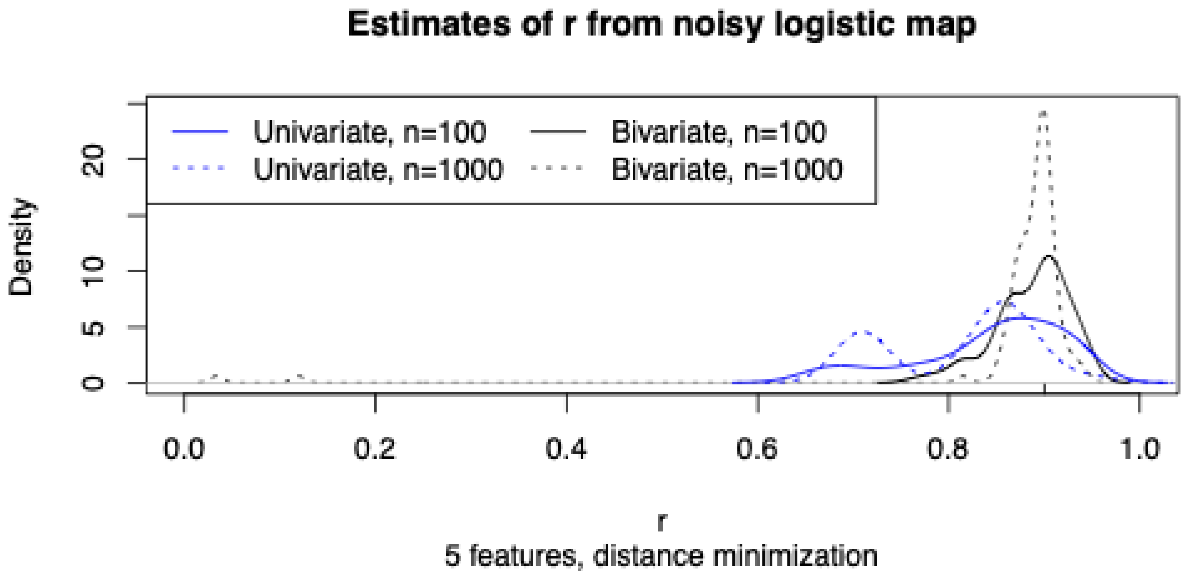
Logistic Map Plus Observational Noise

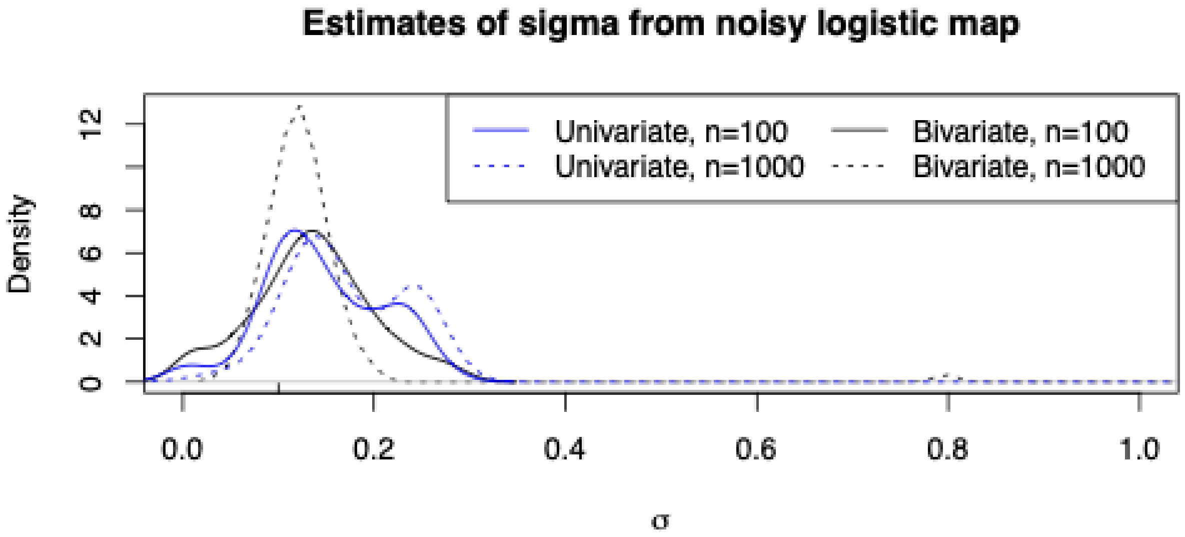
$$
\begin{aligned}
S_0 &\sim \mathrm{Unif}(0,1) && (6)\\
S_{t+1}|S_t &= 4rS_t(1-S_t) && (7)\\
X_t|S_t &= \mathcal{N}(S_t,\sigma^2) && (8)
\end{aligned}
$$

- X_t manifest, S_t latent

- Chaotic dynamical system observed through noise
- Likelihood is easier to comprehend, but calculating…

Logistic Map Plus Observational Noise (2)





So what have we learned?

- In some easy problems, random-feature matching is competitive with the MLE
- In some more challenging problems, it at least works
- *With no fine-tuning*

So what do we still need to do?

- How much efficiency *do* we typically lose, compared to the MLE, for using just $2d + 1$ features?
  - Can we get back some efficiency if we let k grow?
- For time series, should we let the block length grow with n, and how fast?
  - Should be able to do at least $O(\log n)$ but what's the rate?
- What to do for non-stationary time series?
  - More exactly, ones which aren't asymptotically mean-stationary and/or where the stationary state *is* uninformative
- What's the right class of test functions for networks?
  - Or for random fields on networks?

So what should you take away from all this?

- If your scientists' model is not too pathological and has d parameters, then the expectations of $2d + 1$ random basis functions will, generically, identify the parameters
- If those random features have decent ergodic properties, you can get consistent parameter estimates by adjusting θ until the value of the features under simulation matches the values seen in data
  - The usual asymptotics apply (conditional on features)
- The *same* sets of random features can work for many *different* models, depending on the sample space
- We should think about good sets of random features for data like networks, and about non-stationarity

In conclusion

- **Skeptic**: Noooo, you can't estimate a complicated generative model by just twisting the parameter knobs until the averages of totally random functions, picked without any understand or insight into the model and without any regard for its inner workings, just happen to approach the values of those functions on the data, that's not how simulation-based inference works

- **Fool**: Hahaha sine wave printer go brrrrr

- Thank you for your attention!

I'm glad you asked that quetion!

Backup: The usual asymptotics

$$
\begin{aligned}
\hat{\theta} &\equiv \operatorname*{argmin}_{\theta \in \Theta} M_n(\theta) && (9)\\
m(\theta) &\equiv \lim_{n \to \infty} M_n(\theta) \text{ (assumption)} && (10)\\
\theta^* &\equiv \operatorname*{argmin}_{\theta \in \Theta} m(\theta) && (11)\\
\nabla M_n(\hat{\theta}) = 0 &\approx \nabla M_n(\theta^*) + \nabla\nabla M_n(\theta^*)(\hat{\theta} - \theta^*) \text{ (Taylor expansion)} && (12)\\
\hat{\theta} &\approx \theta^* - \left(\nabla\nabla M_n(\theta^*)\right)^{-1} \nabla M_n(\theta^*) \approx \theta^* - \left(\nabla\nabla m(\theta^*)\right)^{-1} \nabla M_n(\theta^*) && (13)\\
\mathbb{V}\left[\hat{\theta}\right] &\approx \left(\nabla\nabla m(\theta^*)\right)^{-1} \mathbb{V}\left[\nabla M_n(\theta^*)\right] \left(\nabla\nabla m(\theta^*)\right)^{-1} && (14)\\
\text{if } M_n(\theta) &= \frac{1}{2}\|P_{n,\theta} f - f(X_{1:n})\|^2 && (15)\\
\text{and so } m(\theta) &= \frac{1}{2}\|\phi(\theta) - \phi(\theta^*)\|^2 && (16)\\
\mathbf{g} &\equiv \nabla \phi(\theta^*) && (17)\\
\mathbf{v}_n &\equiv \mathbb{V}\left[f(X_{1:n})\right] && (18)\\
\text{then } \mathbb{V}\left[\hat{\theta}\right] &\approx (\mathbf{g}^T \mathbf{g})^{-1} \mathbf{g}^T \mathbf{v}_n \mathbf{g} (\mathbf{g}^T \mathbf{g})^{-1} && (19)
\end{aligned}
$$

- So we get an (asymptotic) sandwich variance matrix and associated standard errors
- If $\nabla M_n(\theta^*)$ has a Gaussian limiting distribution (perhaps because $M_n(\theta^*)$ does, perhaps because it's an average of many small weakly-dependent terms), we can get (asymptotic) Gaussian confidence sets for θ
- With random F, this all works *conditional on* F

Backup: Embedding and "geometry from a time series"

- The embedding theorem is extensively used to study *deterministic* dynamical systems
- Say $s_{t+1} = g(s_t)$ for deterministic g
- We observe $x_t = f(s_t)$
- Now $x_{t+1} = f(g(s_t)) =$ a different function of s_t
- So $x_{t+k}, x_{t+k-1}, \ldots x_t$ is a k dimensional function of s_t
- Embedding: if g and f are generic smooth functions, the **time-delay vector** $(x_{t+k}, x_{t+k-1}, \ldots x_t)$ and the latent state s_t are diffeomorphic
- $\Rightarrow$ the map which takes $(x_{t+k}, x_{t+k-1}, \ldots x_t)$ to $(x_{t+k+1}, x_{t+k}, \ldots x_{t+1})$ is equivalent to g ("up to a smooth change of coordinates")
  - $\therefore$ x_{t+k+1} is a deterministic function of $(x_{t+k}, x_{t+k-1}, \ldots x_t)$!
- $\Rightarrow$ we can do prediction, study the properties of the dynamical system and the manifold, etc., just by using the time-delay vectors
- Original idea: Packard et al. (1980); connection to Whitney, Takens (1981); best versions of the embedding theorems (for these purposes) Sauer, Yorke, and Casdagli (1991)

Backup: A bit more about applying the embedding theorem

- Assume the distributions form a manifold
- If we could just observe $k = 2d + 1$ generic functionals of the distribution, we could invert them to recover the position on the manifold
- But: neighborhoods on the manifold are generated by sets of the form $\{\theta : |P_\theta f - a| \leq b\}$ for some bounded, continuous f

    - because: convergence on the manifold implies convergence in distribution
- So we can restrict ourselves from arbitrary smooth functionals of the distribution to expectations of bounded, continuous test functions
- But the span of nice function bases is dense in the space of test functions so we can restrict ourselves to expectations of those basis functions

Backup: Not just the Fourier basis

- The Fourier basis works because the characteristic function $\mathbb{E}\left[e^{it\cdot X}\right]$ determines the distribution of X and vice versa
- And *that* is (basically) because of the Stone-Weierstrass theorem
- So we could use any set of basis functions which *also* satisfy S-W
    - E.g., variously centered and stretched logistic functions
- I didn't find anything which worked *better* than the Fourier basis on my examples, but that's not a proof

Backup: Not just time averages (in principle)

- Lots of complicated functions that aren't just averages over time *also* concentrate on expectation values
    - This is important for, e.g., indirect inference
- It could be that we'll get *more* information from using global but concentrating random functions
- I have not been able to make this work
    - It turns out that $\cos\left(W \cdot X_{1:n} + B\right)$, $W \sim \mathcal{MVN}(0, \Sigma)$ is very close to 0 under just about any distribution for $X_{1:n}$
- I *suspect* this will nonetheless be helpful for strong non-stationarity however

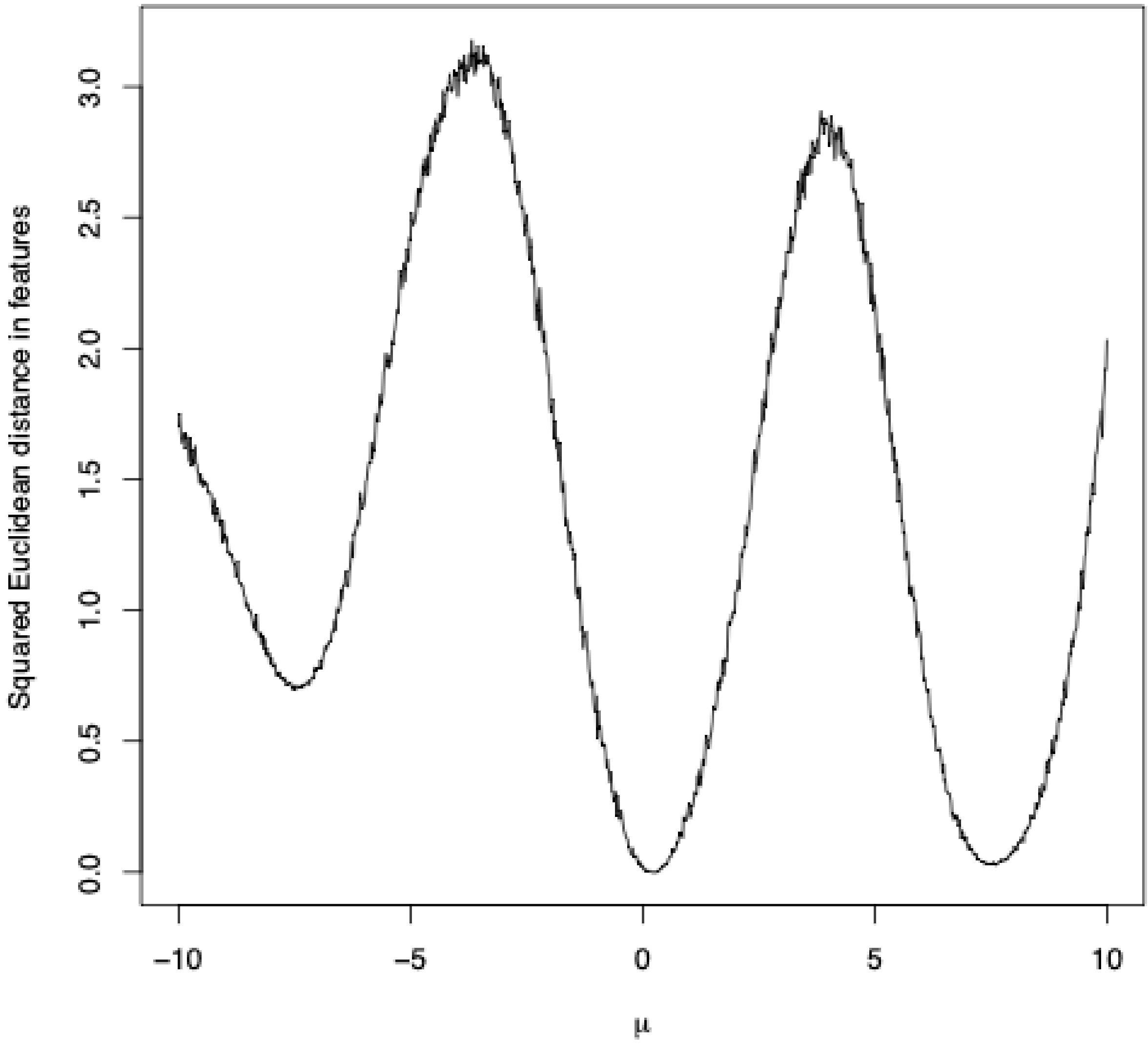
Backup: "Wood likelihood"

- A trick I learned from Wood (2010)
- Instead of just minimizing distance to the *expected* value of F, use simulations to estimate both $\mathbb{E}\left[F\right]$ and $\mathbb{V}\left[F\right]$ as a function of θ
- Then calculate the Gaussian likelihood for the *observed* $F(x_{1:n})$
- Maximize
- More forgiving of sloppy matches to highly-variable features, etc., etc.
- In my trials so far, sometimes helps, sometimes doesn't, would be nice to know why…

Backup: The actual implementation

- Just writing argmin is a theorist's way of saying "that's not my department"
- The actual objective function is very rough here
    - Could smooth it out, some, by re-using the same random number seeds for different θs, as in (Gouriéroux and Monfort 1996) but even that wouldn't help much with something like the logistic map
- I am just brute-forcing it by using "generalized" simulated annealing
    - i.e., simulated annealing with heavy-tailed proposals
    - Implemented in the `GenSA` package (Yang Xiang et al. 2013)
- Code: CMU is actually trying to patent it
    - Commercial partners very welcome
    - Get in touch about non-commercial uses, we'll make it happen

Backup: Distance between Gaussians in univariate random features



Distance, using the same 3 random Fourier features, between one $n = 100$ sample ($\mathcal{N}(0,1)$) and the average of $s = 10$, $n = 100$ simulations at varying $\mathcal{N}(\mu,1)$.

Backup: More exact statements of the random-function approximation results

(This is not the best version of the results, or even the best version of Rahimi and Recht (2008))

- Say that $\mathcal{F}$ consists of functions of form $\psi(\omega \cdot x)$ where $\psi : \mathbb{R} \mapsto \mathbb{R}$ is Lipschitz, $x \in \mathbb{R}^q$, and ρ is a fixed distribution over $\mathbb{R}^q$ with finite second moments
- $\mathcal{M}$ = integral mixtures from $\mathcal{F}$, so $\int a(\omega)\psi(\omega \cdot x)d\omega$ where $\sup \left|\frac{a(\omega)}{\rho(\omega)}\right| < \infty$
- Pick your favorite $m \in \mathcal{M}$

- Sample $\Omega_1, \ldots \Omega_k$ iidly from ρ
- There exist constants $a_1, \ldots a_k$ such that

  $$\mathbb{P}\left(\left\|\sum_{i=1}^{k} a_i \psi(\Omega_i \cdot x) - m(x)\right\|_2 \leq \frac{c_1}{\sqrt{k}}\left(1 + \sqrt{2 \log \frac{1}{\delta}}\right)\right) \geq 1 - \delta$$

  and

  $$\mathbb{P}\left(\left\|\sum_{i=1}^{k} a_i \psi(\Omega_i \cdot x) - m(x)\right\|_\infty < \frac{c_1}{\sqrt{k}}\left(\sqrt{\log \frac{1}{\delta}} - c_2\right)\right) \geq 1 - \delta$$

  - Probability here is over the random choice of basis functions
- Moreover: $\mathcal{M}$ is dense in the RKHS whose kernel is defined by $k(x, y) = \int \rho(\omega)\psi(\omega \cdot x)\psi(\omega \cdot y) d\omega$
  - E.g., for random Fourier feature with a Gaussian frequency distribution, this says $\mathcal{M}$ is dense in the RKHS corresponding to the Gaussian kernel function…

Backup/Aside: A little Bayes result

- Pick $F_1, \ldots F_k$ before seeing the data or thinking about the model, from the random Fourier basis with a Gaussian distribution over frequencies
- Start with a prior Π
- Repeatedly sample $\theta \sim \Pi$ and then $\tilde{X}|\theta \sim P_{n,\theta}$, so we get $\theta_1, \theta_2, \ldots \theta_m$ and corresponding $\tilde{X}$s
- Set

  $$\hat{a} = \operatorname*{argmin}_{a \in \mathbb{R}^k} \sum_{j=1}^{m} \|\theta_j - \sum_{i=1}^{k} a_i F_i(\tilde{X}(\theta_j))\|^2$$

- Now, with high probability over the choice of F_i,

  $$\sum_{i=1}^{k} \hat{a}_i F_i(x) - \mathbb{E}\left[\theta | X = x\right] = O(1/\sqrt{k})$$

  in L_2 and L_∞, *if* the posterior mean is in the RKHS generated by a Gaussian kernel
  - $O(1/\sqrt{k})$ to the RKHS approximation to the posterior mean if not
  - … if you trust your prior

References

Devaney, Robert L. 1992. *A First Course in Chaotic Dynamical Systems: Theory and Experiment.* Reading, Massachusetts: Addison-Wesley.

Gouriéroux, Christian, and Alain Monfort. 1996. *Simulation-Based Econometric Methods.* Oxford, England: Oxford University Press.

Packard, Norman H., James P. Crutchfield, J. Doyne Farmer, and Robert S. Shaw. 1980. "Geometry from a Time Series." *Physical Review Letters* 45:712–16. https://doi.org/10.1103/PhysRevLett.45.712.

Rahimi, Ali, and Benjamin Recht. 2008. "Uniform Approximation of Functions with Random Bases." In *46th Annual Allerton Conference on Communication, Control, and Computing*, edited by P. Moulin and C. Beck, 555–61. Urbana-Champaign, Illinois: IEEE. https://doi.org/10.1109/ALLERTON.2008.4797607.

Sauer, Tim, James A. Yorke, and Martin Casdagli. 1991. "Embedology." *Journal of Statistical Physics* 65:579–616. https://doi.org/10.1007/BF01053745.

Takens, Floris. 1981. "Detecting Strange Attractors in Fluid Turbulence." In *Symposium on Dynamical Systems and Turbulence*, edited by D. A. Rand and L. S. Young, 366–81. Berlin: Springer-Verlag. https://doi.org/10.1007/BFb0091924.

Whitney, Hassler. 1936. “Differentiable Manifolds.” *Annals of Mathematics* 37:645–80. https://doi.org/10.2307/1968482.

Wood, Simon N. 2010. “Statistical Inference for Noisy Nonlinear Ecological Dynamic Systems.” *Nature* 466:1102–4. https://doi.org/10.1038/nature09319.

Yang Xiang, Sylvain Gubian, Brian Suomela, and Julia Hoeng. 2013. “Generalized Simulated Annealing for Efficient Global Optimization: The GenSA Package for R.” *The R Journal* 5 (1). https://journal.r-project.org/archive/2013/RJ-2013-002/index.html.