Improving Bayesian Synthetic Likelihood via Transformations

Associate Professor Chris Drovandi

School of Mathematical Sciences
Queensland University of Technology Centre for Data Science
ARC Centre of Excellence for Mathematical and Statistical Frontiers (ACEMS)

Collaborators: Jacob Priddle, David Frazier and Scott Sisson

2nd July 2020

Bayesian Statistics

In Bayesian statistics we are interested in sampling from the posterior:

$$p(\theta|y) \propto p(y|\theta)p(\theta),$$

where $p(y|\theta)$ is the likelihood, $p(\theta)$ is the prior, $y = (y_1, \ldots, y_m)^\top$ is the observed data and $\theta \in \Theta \subset \mathbb{R}^p$ is an unknown parameter.

Simulator Models

Simulator models are a type of stochastic model that is often used to approximate a real-life process, such as:

- the movement patterns of invasive species of animals;
- biological mechanisms, or
- the outbreak of an infectious disease,

Unfortunately, for these types of models, $p(y|\theta)$ is generally computationally intractable.

Approximate Bayesian Computation (ABC)

ABC is the current state-of-the-art likelihood-free Bayesian method.

ABC prefers θ that produce simulated data $x \sim p(\cdot|\theta)$ that is 'close' to y in terms of summary statistics that are generated according to $S(\cdot) : \mathbb{R}^m \to \mathbb{R}^d$.

Targets the posterior conditional on observed summary $p(\theta|s_y) \propto p(s_y|\theta)p(\theta)$ where $s_y = S(y)$.

Estimates $p(s_y|\theta)$ non-parametrically via simulations.

Choice of $S(\cdot)$ is a trade-off between information loss and dimensionality.

ABC Likelihood Approximation

Simulate n iid datasets, denoted $x_{1:n} = (x_1, \ldots, x_n)$, from the model based on θ.

Calculate n sets of summary statistics, $s_{1:n} = (S(x_1), \ldots, S(x_n)) = (s_1, \ldots, s_n)$.

The intractable $p(s_y|\theta)$ is replaced with the estimated ABC likelihood,

$$\hat{p}_\epsilon(s_y|\theta) = \frac{1}{n}\sum_{i=1}^{n} K_\epsilon(\rho(s_y, s_i)).$$

- $\rho(\cdot)$ is called the discrepancy function;
- $K_\epsilon(\cdot)$ is a kernel weighting function with bandwidth ϵ, and
- ϵ is called the ABC tolerance (bias/variance trade-off).

QUT Centre for Data Science ACEMS

ABC Limitations

ABC has several drawbacks, including:

- Highly sensitive to choice of tuning parameters ϵ, $\rho(\cdot)$ and to a lesser extent $K_\epsilon(\cdot)$;
- No standard way to select ϵ or ρ, and
- Suffers from the curse of dimensionality with respect to the size of the summary statistic.

Synthetic Likelihood

In synthetic likelihood methods, we assume a parametric form of the likelihood, which acts as a surrogate for the true likelihood.

In general, synthetic likelihood methods

- can scale better to a high-dimensional summary statistic
- do not require as much tuning

Synthetic likelihood

The synthetic likelihood method[1] uses a working normal likelihood:

$$p(s_y|\theta) \approx p_A(s_y|\theta) = \mathcal{N}(s_y|\mu(\theta), \Sigma(\theta)).$$

$\mu(\theta)$ and $\Sigma(\theta)$ are typically unknown but can be estimated according to:

$$\mu_n(\theta) = \frac{1}{n}\sum_{i=1}^{n} S(x_i) \text{ and}$$

$$\Sigma_n(\theta) = \frac{1}{n-1}\sum_{i=1}^{n}(S(x_i) - \mu_n(\theta))(S(x_i) - \mu_n(\theta))^\top,$$

where $x_1, \ldots, x_n \overset{\text{iid}}{\sim} p(\cdot|\theta)$. The synthetic likelihood estimate is then

$$p_A(s_y|\theta) \approx \mathcal{N}(s_y|\mu_n(\theta), \Sigma_n(\theta)).$$

[1] Wood (2010). Nature. 466:1102.

Bayesian Synthetic Likelihood

The synthetic likelihood has been considered in a Bayesian framework[2], called Bayesian Synthetic likelihood (BSL).

The BSL posterior approximation is then:

$$p_{\text{BSL}}(\theta|s_y) \propto p_{\text{BSL}}(s_y|\theta)p(\theta), \text{ where}$$

$$p_{\text{BSL}}(s_y|\theta) = \int \mathcal{N}(s_y|\mu_n(\theta), \Sigma_n(\theta)) \prod_{i=1}^{n} p(S(x_i)|\theta) dS(x_1) \cdots S(x_n).$$

Typically MCMC is used to sample from $p_{\text{BSL}}(\theta|s_y)$. n is a tuning parameter chosen to maximise the computational efficiency.

[2] Price et al (2018). JCGS. 27:1-11

Limitations of BSL

1. The number of simulations per iteration, n, needs to be large for estimating high dimensional covariance matrix.
2. The distribution of the summary statistic must be roughly Gaussian.
3. Reliance on MCMC to explore parameter space (not ideal in high dimensions).
4. Can perform poorly when model cannot recover observed statistic (i.e. model misspecification).

In this talk, we address 1 and 2; 3 and 4 are addressed in other work.

Scaling of the *unconstrained* log SL variance

Assumption 1. The simulated summaries are generated iid and distributed according to $\mathcal{N}(\mu(\theta), \Sigma(\theta))$.

Result 1. Consider the standard synthetic likelihood estimator as $\hat{p}_d(s_y|\theta) = \mathcal{N}(s_y; \hat{\mu}_n(\theta), \hat{\Sigma}_n(\theta))$. For n and d large:

$$\text{Var}\left[\log \hat{p}_d(s_y|\theta)\right] = \mathcal{O}\left(\frac{d^2 n^2}{(n-d)^3}\right).$$

Letting $n \propto d^2$, we have that $\text{Var}\left[\log \hat{p}_d(s_y|\theta)\right] = \mathcal{O}(1)$.

Thus, n must scale *quadratically* with d to control the variance of the *unconstrained* log synthetic likelihood estimator.

Scaling of the *diagonal* log synthetic likelihood variance

Assumption 2. The simulated summaries are generated iid and distributed according to $\mathcal{N}(\zeta(\theta), \Omega(\theta))$, where $\Omega(\theta)$ is diagonal.

Result 2. Synthetic likelihood estimator: $\hat{p}_{d,w}(s_y|\theta) = \mathcal{N}(s_y; \hat{\zeta}_n(\theta), \hat{\Omega}_n(\theta))$. For n and d large:

$$\text{Var}\left[\log \hat{p}_{d,w}(s_y|\theta)\right] = \mathcal{O}\left(\frac{dn^2}{(n-d)^3}\right).$$

Letting $n \propto d$, we have that $\text{Var}\left[\log \hat{p}_{d,w}(s_y|\theta)\right] = \mathcal{O}(1)$.

Thus, n must scale *linearly* with d to control the variance of the *diagonal* log synthetic likelihood estimator.

Significant computational benefits are possible in BSL algorithms if the summary statistics are uncorrelated.

However, it is challenging to find a summary statistic vector that is both independent across its dimensions and retains a large proportion of the information content intrinsic to the observed data.

Our solution: Whitening Bayesian Synthetic Likelihood[3].

[3] Priddle et al (2020). *arXiv preprint arXiv:1909.04857*.

Covariance Estimation

In previous BSL research we have considered the following covariance matrix shrinkage estimator[4]:

$$\Sigma_{n,\gamma} = \Sigma_d^{1/2}(\gamma \hat{R} + (1-\gamma) I_d)\Sigma_d^{1/2},$$

where $\hat{R} = \Sigma_d^{-1/2}\Sigma_n\Sigma_d^{-1/2}$ where $\Sigma_d = \text{diag}(\Sigma_n)$.

By using the Warton estimator, we can reduce n.

However, when there is significant correlation between the marginal summary statistics, the Warton estimator is a poor approximation of Σ_n. This leads to poor posterior approximations.

[4] Warton (2008). JASA 103(481):340-349.

Whitening Bayesian Synthetic Likelihood

We consider a whitening transformation

$$\tilde{s} = Ws$$

which transforms so that $\text{Var}(s) = \Sigma$ becomes $\text{Var}(\tilde{s}) = I_d$.

W is estimated at some initial parameter value θ^0. Now the transformation is approximate at a given θ value.

We hold W constant, and use the wBSL estimator in an MCMC algorithm.

Given the summary statistics are approximately decorrelated, we can apply heavier shrinkage.

There are infinitely many W that decorrelate the marginal summary statistics at θ^0.

Different W are more effective in decorrelating summaries generated away from θ^0.

We find PCA whitening works best. $W_{\text{PCA}} = \Lambda^{-1/2} U^\top$, where $\Sigma = U \Lambda U^\top$ is the eigendecomposition of the covariance matrix.

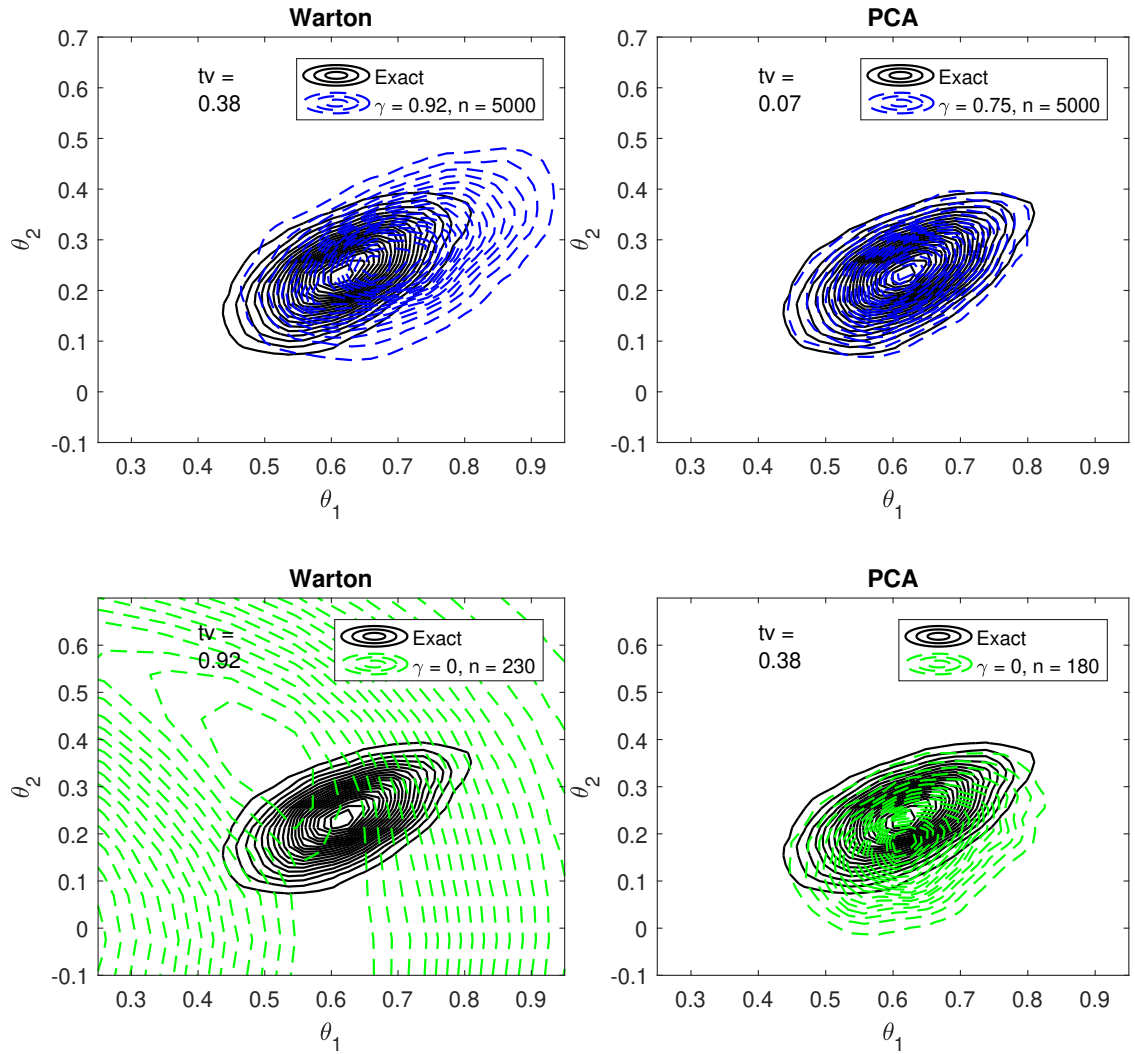
Toy example – MA(2) model

The MA(2) model represents a univariate series of temporally dependent observations as

$$x_t = w_t + \theta_1 w_{t-1} + \theta_2 w_{t-2}$$

where $w_i \sim \mathcal{N}(0, \sigma^2)$ for $i = -1, 0, 1, \ldots, T_0$, $t = 1, \ldots, T_0$, and has parameter constraints $-1 < \theta_2 < 1$, $\theta_1 + \theta_2 > -1$ and $\theta_1 - \theta_2 < 1$.

Here y is 200 observations from the MA(2) process with $\theta_{\text{true}} = (\theta_1, \theta_2)^\top = (0.6, 0.2)^\top$. We take $s_y = y$ to be the full dataset.



Results for the MA(2) example

Toad example

We consider an individual-based model for the movement of a species of Fowler's toads[5]. We consider their random return model.

The model assumes toads take refuge during the day and forage throughout the night.

The overnight displacement is drawn from the levy alpha-stable distribution $S(\alpha, \xi)$ with stability α and scale ξ.

Toads return at the end of their foraging with probability p_0. The return site is determined at random from previous refuge sites.

[5] Marchand et al 2017. Ecological Modelling. 360:63-69

y is the displacements of $n_t = 66$ toads over $n_d = 63$ days.

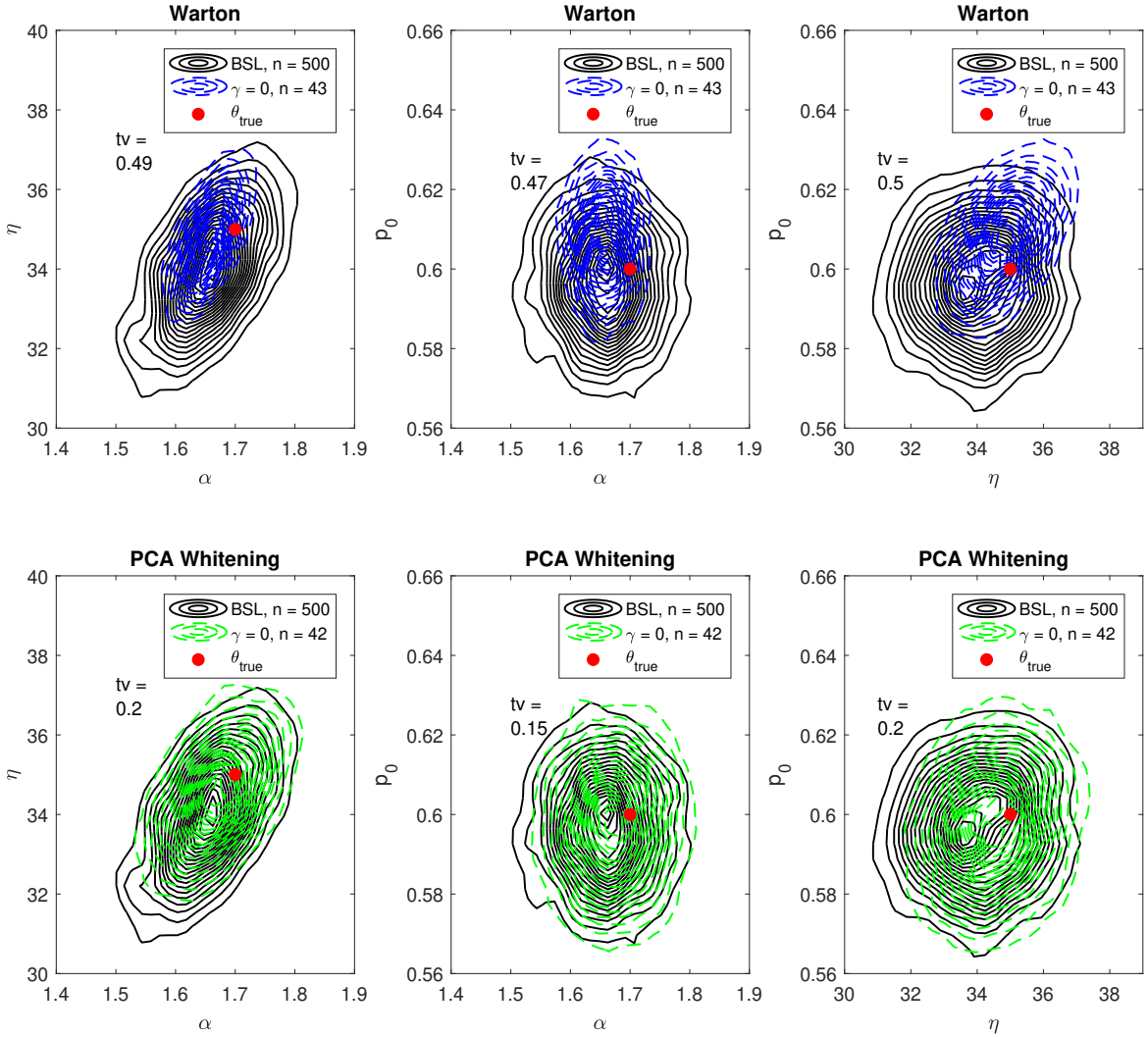
y is summarised to 4 sets comprising the moving distances for time lags of $1, 2, 4, 8$ days.

Number of returns for all four time lags (defined as distance < 10m).

For the non-returns we consider log difference between adjacent p-quantiles with $p = 0, 0.1, \ldots, 1$ and also the median. Repeat for each time lag.

48 statistics. Difficult for conventional ABC to deal with.

We use a simulated dataset with $\theta_{\text{true}} = (\alpha, \xi, p_0)^\top = (1.7, 35, 0.6)^\top$.



Results the Toad example

Semi-parametric BSL (semiBSL)

semiBSL[6] provides additional robustness for a non-Gaussian distributed summary statistic.

We model each univariate summary statistic $\mathcal{S}^j$ using kernel density estimation. That is, given n iid model simulations $x_1, \ldots, x_n$, the KDE is given by:

$$\hat{g}_{S^j}(s) = \frac{1}{n} \sum_{i=1}^{n} K_h(s - S(x_i)^j),$$

where $K_h(u) = h^{-1} K(u/h)$ and h is the bandwidth.

We use a Gaussian kernel for K and select h using rule of thumb.

[6] An et al (2020). Statistics and Computing, 30(3):543-557.

Semi-parametric BSL (semiBSL)

The Gaussian copula is used to model the dependence structure:

$$c(u) = \frac{1}{\sqrt{\det(R)}} \exp\left\{-\frac{1}{2}\eta^\top (R^{-1} - I_d)\eta\right\}$$

where R is the correlation matrix, $\eta = (\Phi^{-1}(u_1), \ldots, \Phi^{-1}(u_d))^\top$, Φ^{-1} is the inverse cdf of the $\mathcal{N}(0,1)$ and $u_j = G_{Sj}(s_y^j)$ for $j = 1, \ldots, d$.

The summary statistic likelihood estimate is then:

$$\frac{1}{\sqrt{\det(\hat{R})}} \exp\left\{-\frac{1}{2}\hat{\eta}_{s_y}^\top (\hat{R}^{-1} - I_d)\hat{\eta}_{s_y}\right\} \prod_{j=1}^{d} \hat{g}_j(s_y^j)$$

Transformation kernel density estimation (TKDE)

Standard KDE often does not provide adequate smoothing over all features of the distribution (global bandwidth), and can fail for heavy tailed distributions.

We consider Transformation kernel density estimation (TKDE)[7].

Firstly, transform the data so that the standard (global bandwidth) KDE is more accurate (close to Gaussian), then transform back to the original domain.

[7] Wand et al (1991). JASA. 86(414):343-353.

Transformation in TKDE

We consider the hyperbolic power transformation[8]:

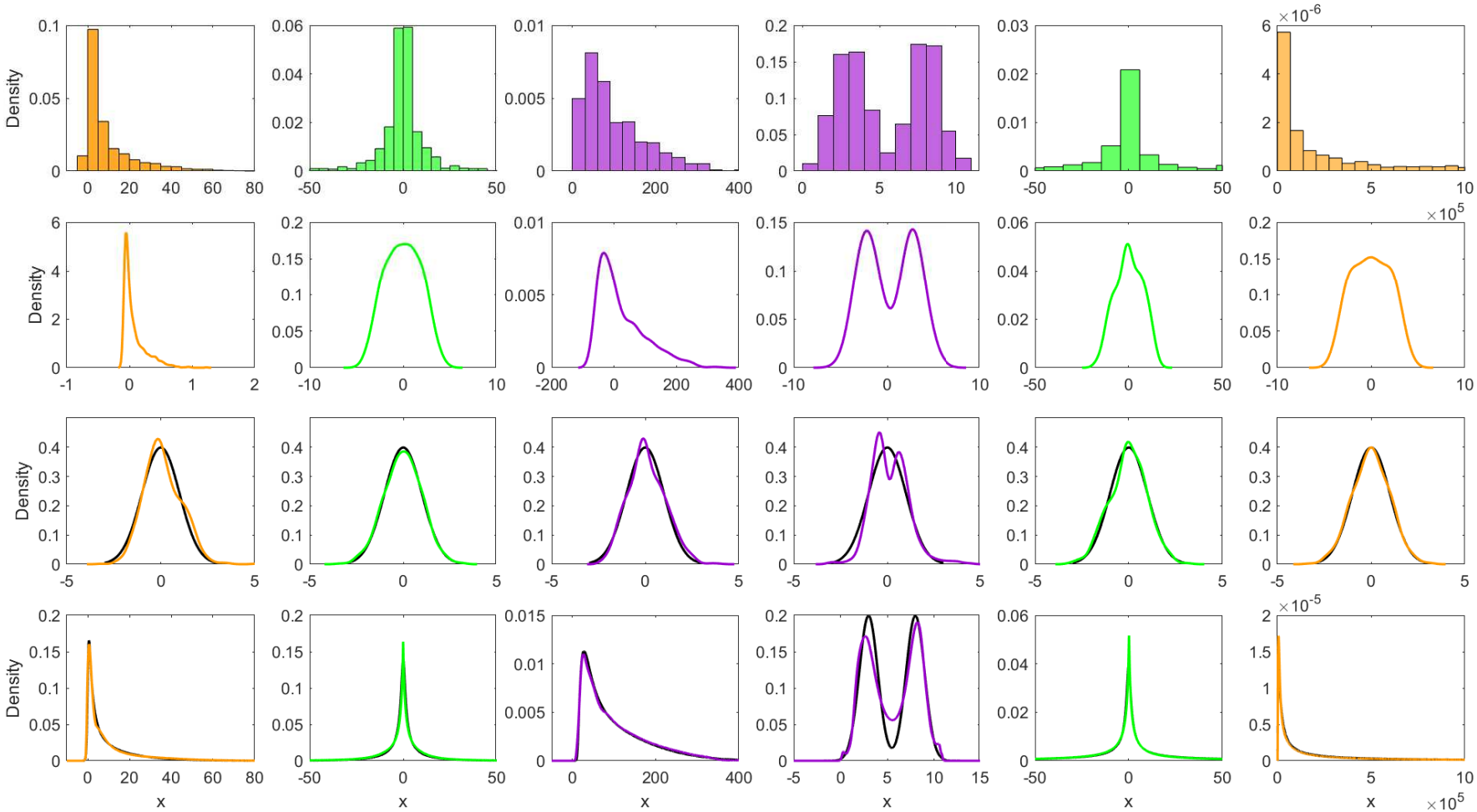
$$\mathcal{G}_\omega(s) = \begin{cases} \nu \sinh(\psi_- s) \operatorname{sech}^{\lambda_-}(\beta_- s)/\psi_- & s \leq 0 \\ \nu \sinh(\psi_+ s) \operatorname{sech}^{\lambda_+}(\psi_+ s)/\psi_+ & s > 0, \end{cases}$$

where s is median centered, $\omega = \{\nu, \psi_-, \lambda_-, \psi_+, \lambda_+\}$, $\nu, \psi_-, \psi_+ > 0$ and $|\lambda_-|, |\lambda_+| \leq 1$. λ_-, λ_+ are the power parameters; ψ_-, ψ_+ are the scale parameters, and α is the normalising constant. We approximate the MLE of ω numerically.

An initial log transformation may also be applied for positively skewed data, negatively skewed data, or data with heavy kurtosis.

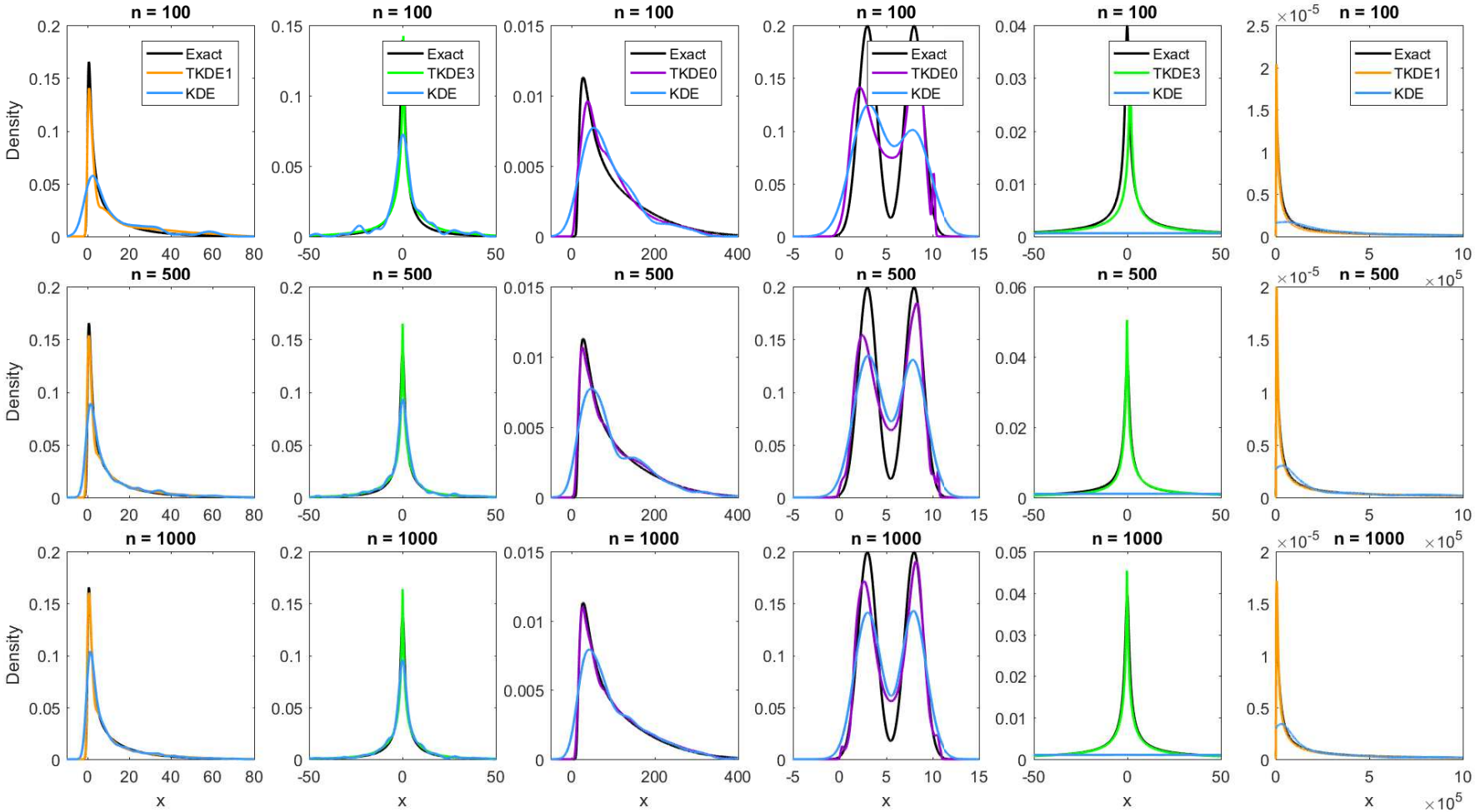
[8] Tsai et al (2017). CSDA. 115:250-266.

Transformation kernel density estimation



Intermediate densities of TKDE procedure.

Transformation kernel density estimation



Comparison of KDE and TKDE estimators.

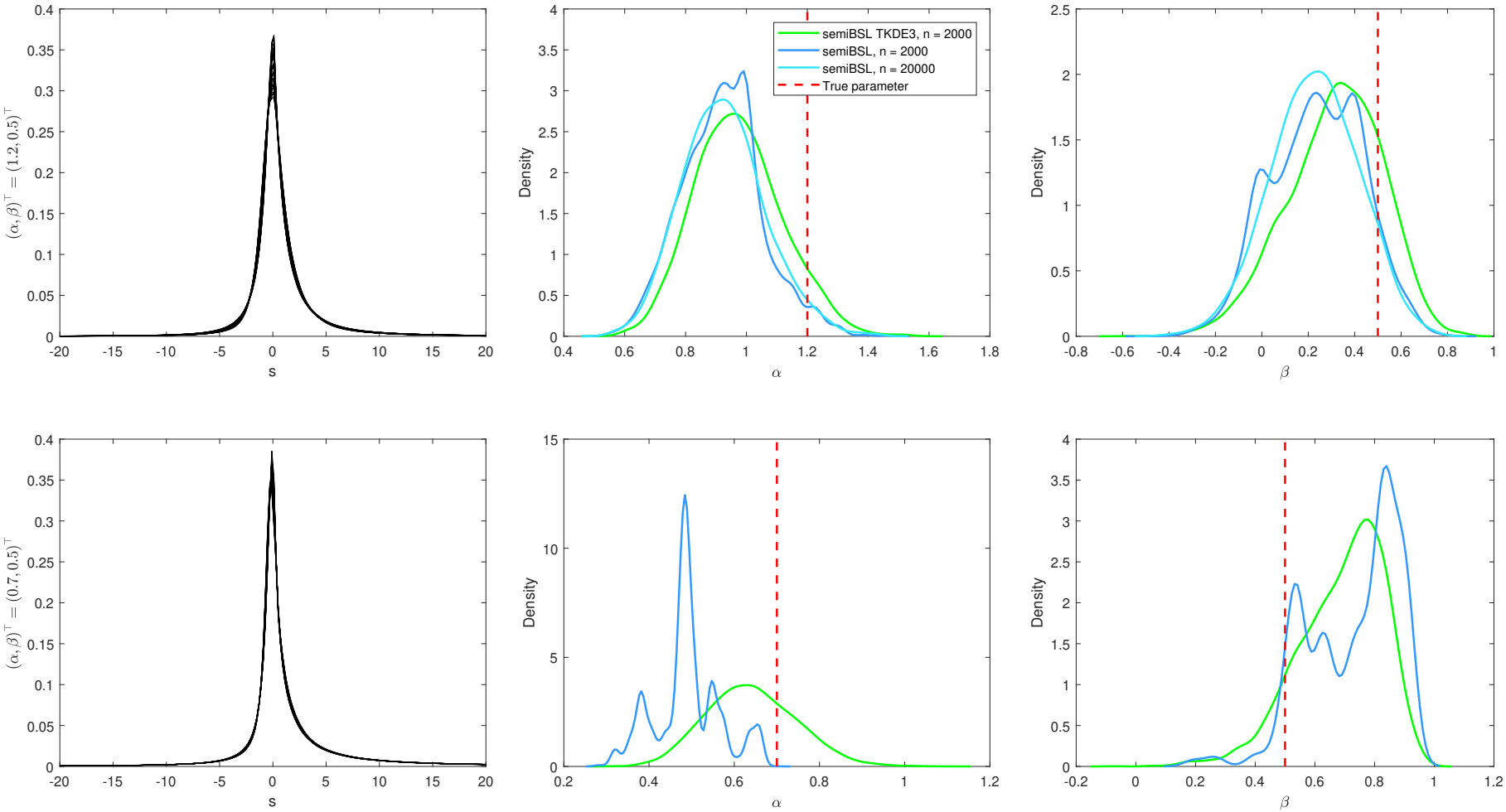
semiBSL TKDE – α-stable stochastic volatility model

Used to model financial returns. The return process is given by:

$$y_t = \exp\left(\frac{x_t}{2}\right) v_t$$
$$x_t \sim \mathcal{N}(\mu + \phi(x_{t-1} - \mu), \sigma_t).$$

We assume $v_t \sim \mathcal{SD}(\alpha, \beta, \gamma, \delta)$, where α, β, γ and δ control the tail heaviness, the skewness, the scale and the location, respectively.

We consider two observed datasets and infer $\theta = (\alpha, \beta)^\top$ with fixed $\mu = 5$, $\phi = 1$, $\gamma = 1$, $\delta = 0$ and $\sigma = 0.2$. We set $\theta_{\text{true}} = (1.2, 0.5)^\top$ and $\theta_{\text{true}} = (0.7, 0.5)^\top$ and $s_y = y$ (50 observations).



Results for α-stable stochastic volatility model.

Whitening or TKDE?

wsemiBSL TKDE

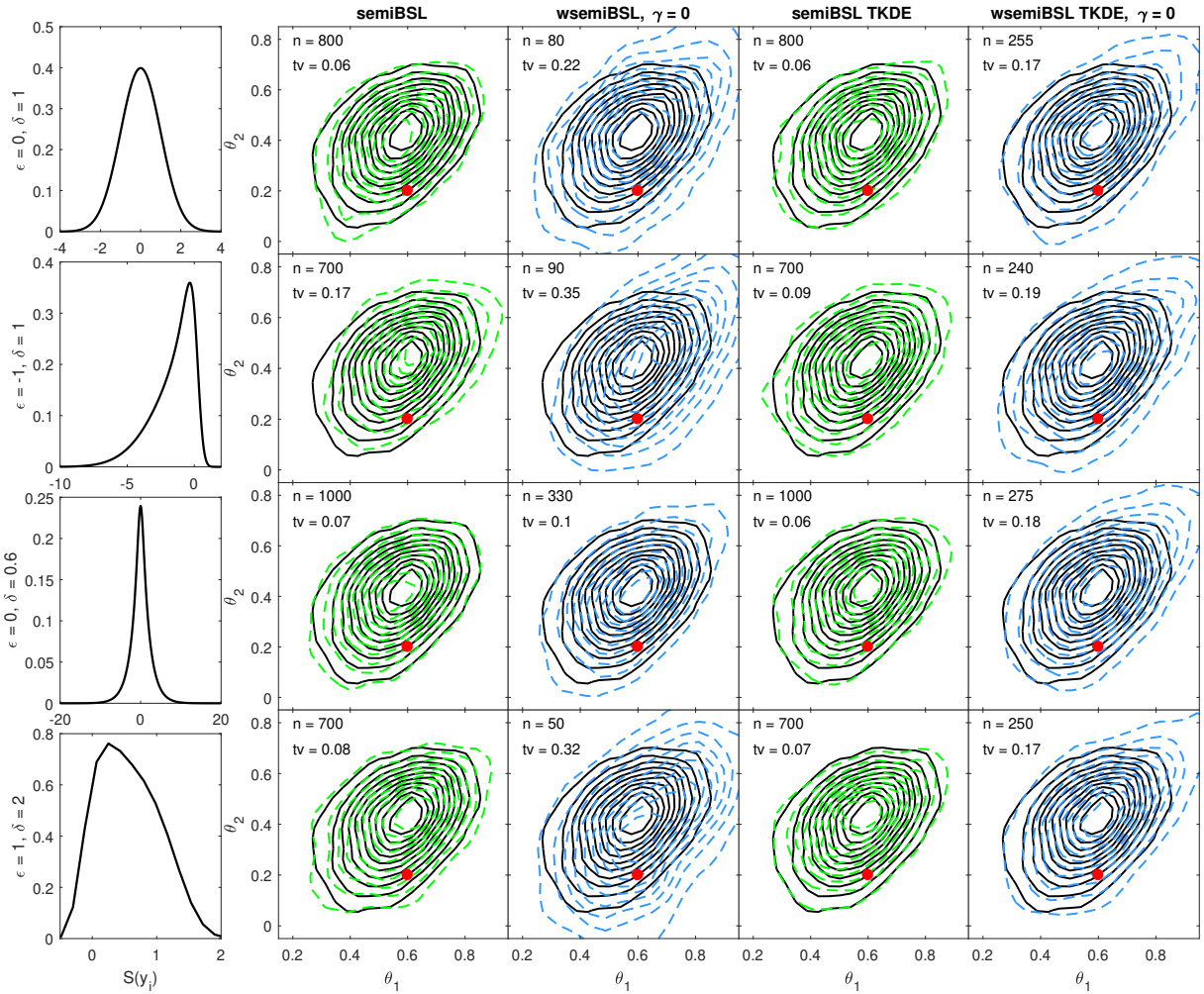
Our whitening method can be applied to semiBSL, call this wsemiBSL. wsemiBSL applies whitening transformation to the standard Gaussian quantiles:

$$\tilde{\eta} = W\eta \tag{1}$$

and not directly to the summary statistic vector.

Furthermore, we can apply whitening in conjunction with TKDE to achieve computational gains on top of the improved robustness, call this wsemiBSL TKDE.

wsemiBSL TKDE – MA(2) example



wsemiBSL TKDE posterior approximations for MA(2) example.

R Package

We have an evolving R package for BSL[9]. TKDE still to be implemented.

[9] An et al (2020). https://arxiv.org/abs/1907.10940.

Limitations/Future Work

Limitations:

- Efficiency gains of whitening is dampened by irregular marginals.

Future Work:

- Improve robustness of BSL methods to non-linear dependence structures between marginal summary statistics.
- Robustness of semiBSL to model misspecification.

References

This talk is primarily based on the two articles:

Priddle et al (2020) Efficient Bayesian Synthetic Likelihood with Whitening Transformations. *arXiv preprint arXiv:1909.04857*.

Priddle and Drovandi (2020) Transformations in Semi-Parametric Bayesian Synthetic Likelihood.

Other key references

Price et al (2018). Bayesian synthetic likelihood. JCGS.

An et al (2020) Robust Bayesian Synthetic Likelihood via a Semi-Parametric Approach. Statistics and Computing.

Tsai et al (2017). On hyperbolic transformations to normality. CSDA.

Wand et al (1991). Transformations in density estimation. JASA.

Warton (2008). Penalized normal likelihood and ridge regularization of correlation and covariance matrices. JASA.

Contact Details

c.drovandi@qut.edu.au chrisdrovandi.weebly.com @chris_drovandi

QUT Centre for Data Science ACEMS