Towards Encrypted Inference for Arbitrary Models

Louis J. M. Aslett (louis.aslett@durham.ac.uk)
Department of Mathematical Sciences
Durham University

Scalable Statistical Inference Workshop
Isaac Newton Institute
5 July 2017

Durham University

Acknowledgements

- Chris Holmes and i-like project — this work germinated while postdoc.
- Ryan Christ for interesting discussions.

i-like.org.uk

Introduction

Motivation

Security in statistics applications is a growing concern:

- computing in a ‘hostile’ environment (e.g. cloud computing);
- donation of sensitive/personal data (e.g. medical/genetic studies);
- complex models on constrained devices (e.g. smart watches)
- running confidential algorithms on confidential data (e.g. engineering reliability)

Perspectives on “privacy”

- Differential privacy
    - on outcomes of ‘statistical queries’
    - guarantees of privacy for individual observations

Perspectives on “privacy”

- Differential privacy
  - on outcomes of ‘statistical queries’
  - guarantees of privacy for individual observations
- Data privacy
  - at rest
  - during fitting
  - data pooling

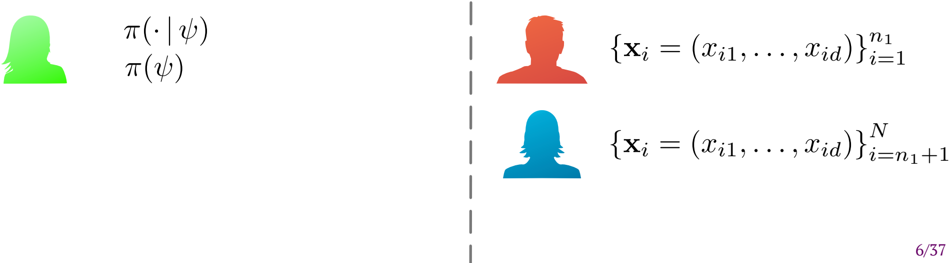
Perspectives on “privacy”

- Differential privacy
    - on outcomes of ‘statistical queries’
    - guarantees of privacy for individual observations
- Data privacy
    - at rest
    - during fitting
    - data pooling
- Model privacy
    - prior distributions
    - model formulation

The perspective for today …

- **Eve** has a private model, including prior information which may itself be private.
- **Cain** and **Abel** have private data which is relevant to the fitting of Eve's model.

Can Eve fit a model, pooling data from Cain and Abel without observing their raw data and without revealing her model and prior information? Abel also doesn't trust Cain …



$\pi(\cdot \mid \psi)$

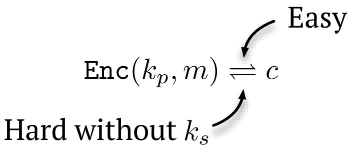
$\pi(\psi)$

$\{\mathbf{x}_i = (x_{i1}, \dots, x_{id})\}_{i=1}^{n_1}$

$\{\mathbf{x}_i = (x_{i1}, \dots, x_{id})\}_{i=n_1+1}^{N}$

Cryptography the solution?

Encryption can provide security guarantees …



Easy

$\texttt{Enc}(k_p, m) \rightleftharpoons c$

Hard without k_s

$\texttt{Dec}(k_s, c) = m$

… but is typically ‘brittle’.

Cryptography the solution?

Encryption can provide security guarantees …

Easy

$$\texttt{Enc}(k_p, m) \rightleftharpoons c \qquad\qquad \texttt{Dec}(k_s, c) = m$$

Hard without k_s

… but is typically ‘brittle’.

Arbitrary addition and multiplication is possible with **fully homomorphic encryption** schemes (Gentry, 2009).

$$\begin{tikzcd}
m_1 \arrow[d, "\texttt{Enc}(k_p, \cdot)"] & m_2 \arrow[d] \arrow[r, "+"] & m_1 + m_2 \\
c_1 & c_2 \arrow[r, "\oplus"] & c_1 \oplus c_2 \arrow[u, "\texttt{Dec}(k_s, \cdot)"']
\end{tikzcd}$$

Back to the problem ...

$\pi(\cdot \mid \psi)$

$\pi(\psi)$

$\{\mathbf{x}_i = (x_{i1}, \dots, x_{id})\}_{i=1}^{n_1}$

$\{\mathbf{x}_i = (x_{i1}, \dots, x_{id})\}_{i=n_1+1}^{N}$

Back to the problem ...

$\pi(\cdot \,|\, \psi)$

$\pi(\psi)$

$\{\mathbf{x}_i = (x_{i1}, \ldots, x_{id})\}_{i=1}^{n_1}$

$\{\mathbf{x}_i = (x_{i1}, \ldots, x_{id})\}_{i=n_1+1}^{N}$

↓

$$\mathbf{x}_i^\star = \mathrm{Enc}(k_p, \mathbf{x}_i)$$

Back to the problem ...

$\pi(\cdot \mid \psi)$

$\pi(\psi)$

$\{\mathbf{x}_i = (x_{i1}, \dots, x_{id})\}_{i=1}^{n_1}$

$\{\mathbf{x}_i = (x_{i1}, \dots, x_{id})\}_{i=n_1+1}^{N}$

$$\mathbf{x}_i^\star = \mathrm{Enc}(k_p, \mathbf{x}_i)$$

$$\pi(\psi \mid X) \propto \mathrm{Dec}\left[k_s, \prod_{i=1}^{N} \pi(\mathbf{x}_i^\star | \mathrm{Enc}(k_p, \psi)) \times \mathrm{Enc}(k_p, \pi(\psi))\right]$$

Back to the problem ...

$\pi(\cdot \,|\, \psi)$

$\pi(\psi)$

$$\pi(\psi \,|\, X) \propto$$

$$\mathrm{Dec}\left[k_s, \prod_{i=1}^{N} \pi(\mathbf{x}_i^\star | \mathrm{Enc}(k_p, \psi)) \times \mathrm{Enc}(k_p, \pi(\psi))\right]$$

✗ Likelihood restricted to low degree polynomials

✗ Can only handle very small N due to multiplicative depth

✗ MAP/posterior? How? MCMC?

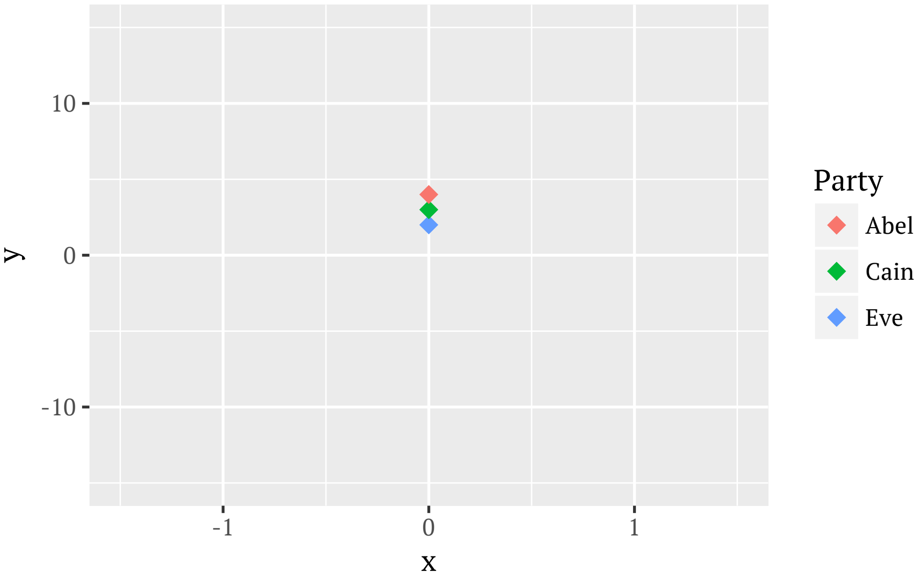
$\{\mathbf{x}_i = (x_{i1}, \ldots, x_{id})\}_{i=1}^{n_1}$

$\{\mathbf{x}_i = (x_{i1}, \ldots, x_{id})\}_{i=n_1+1}^{N}$

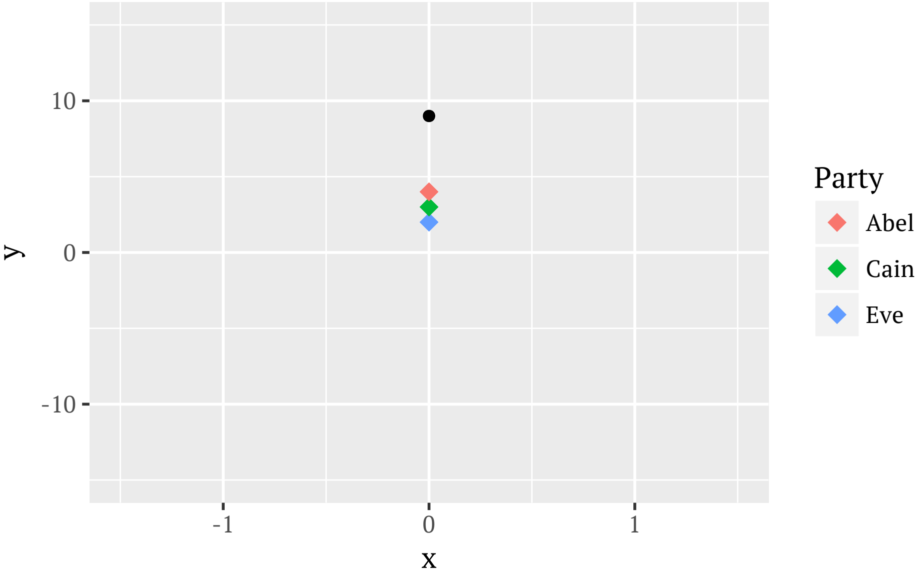
$$\mathbf{x}_i^\star = \mathrm{Enc}(k_p, \mathbf{x}_i)$$

✗ Who holds secret key?

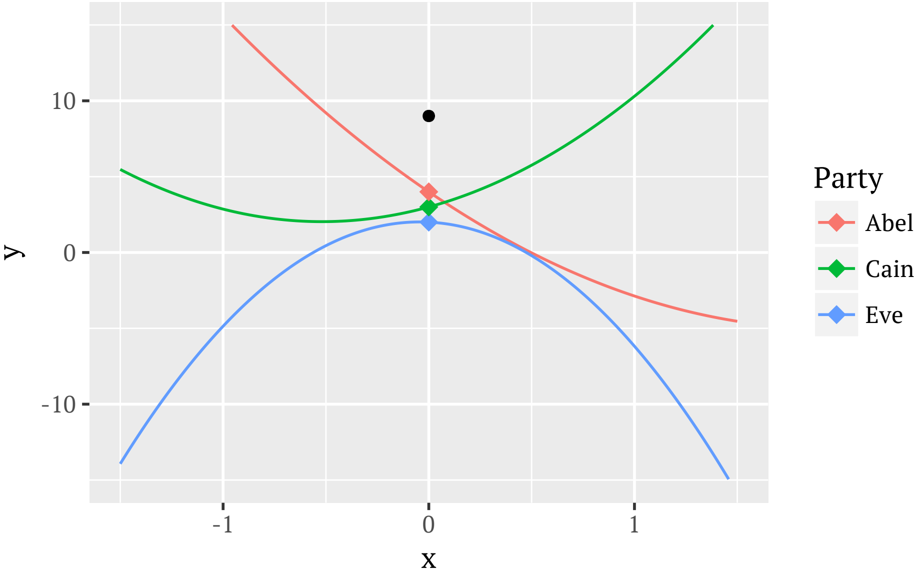
(Simplified) look at Homomorphic Secret Sharing



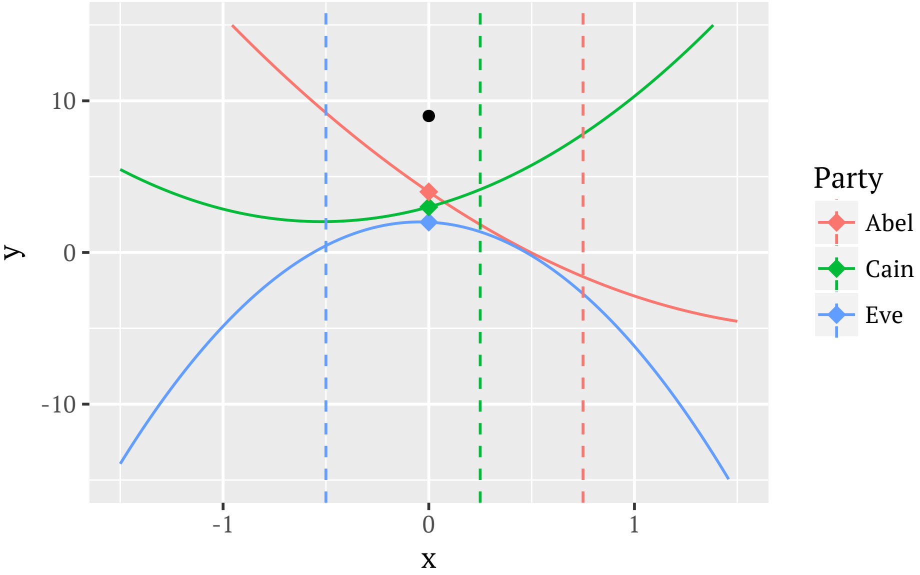
(Simplified) look at Homomorphic Secret Sharing



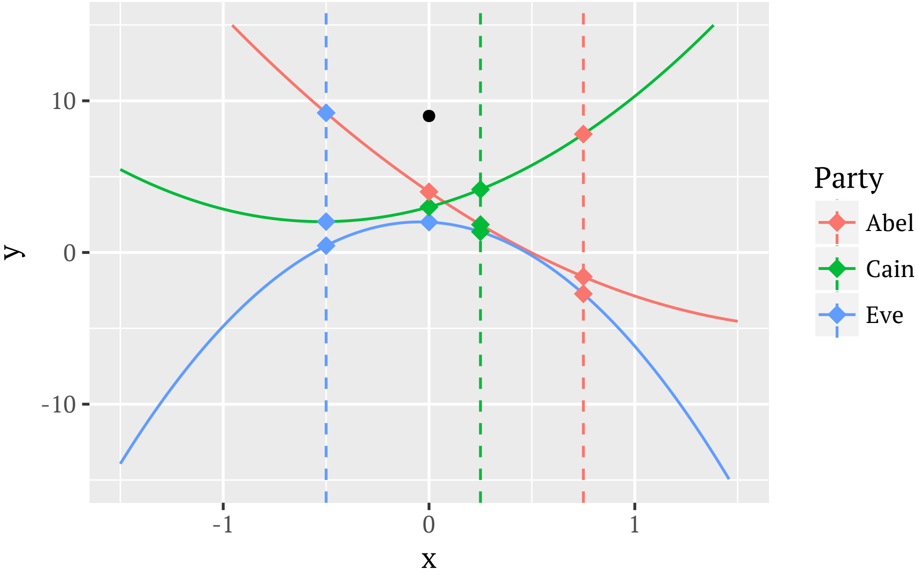
(Simplified) look at Homomorphic Secret Sharing



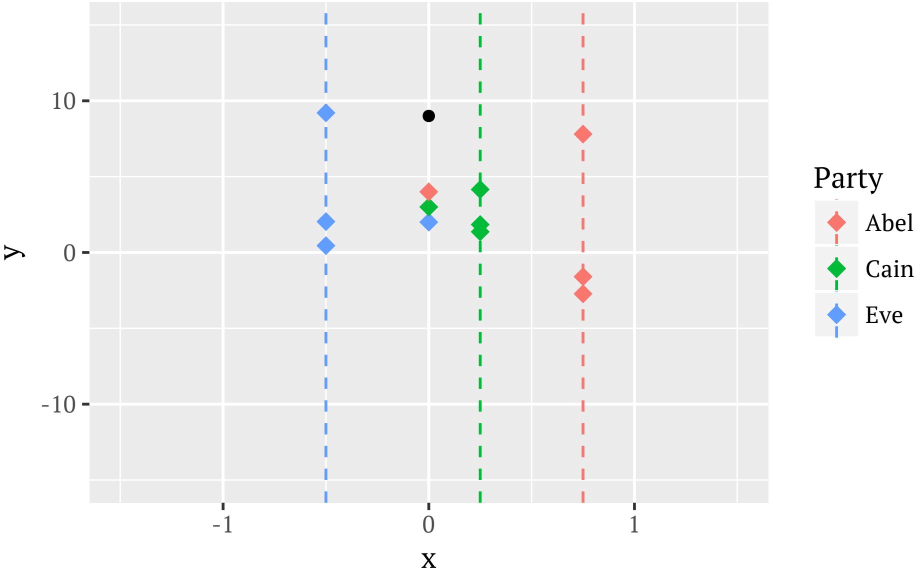
(Simplified) look at Homomorphic Secret Sharing



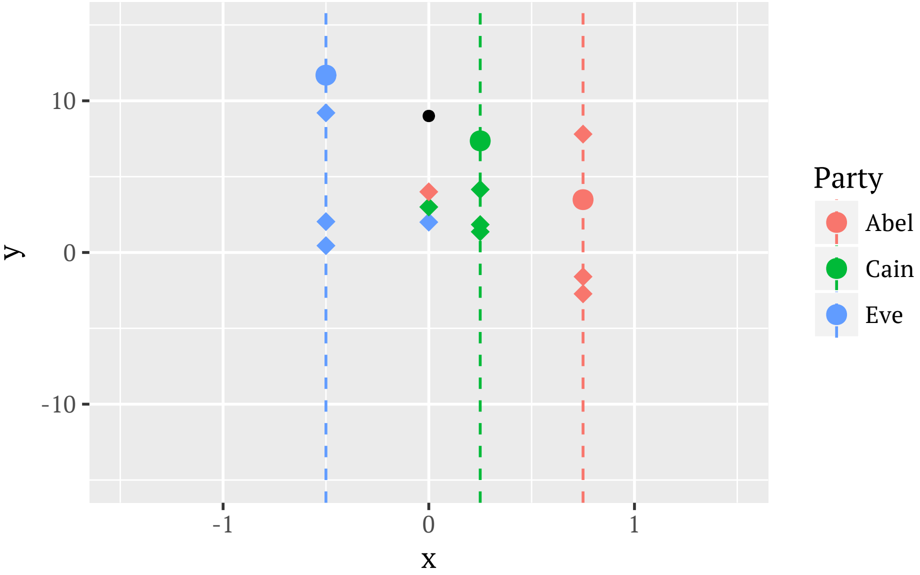
(Simplified) look at Homomorphic Secret Sharing



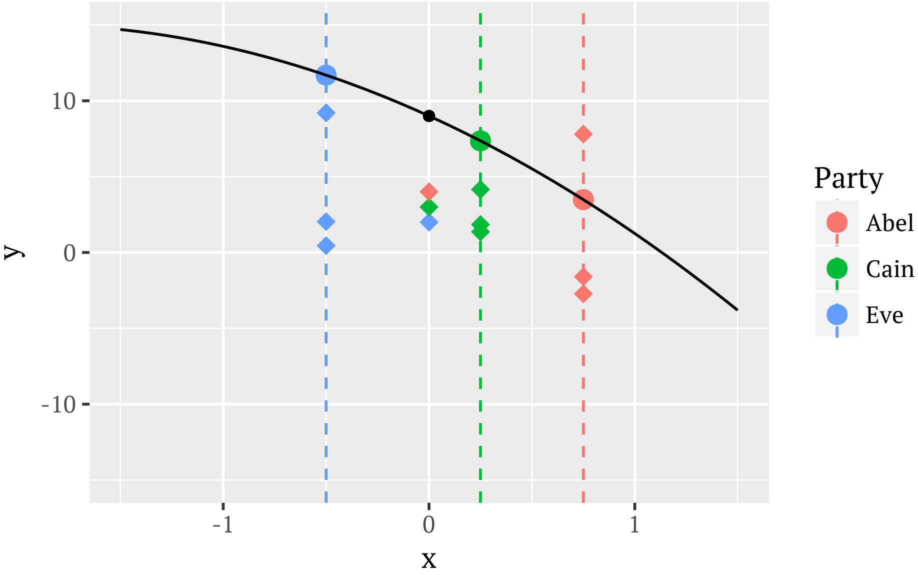
(Simplified) look at Homomorphic Secret Sharing



(Simplified) look at Homomorphic Secret Sharing



(Simplified) look at Homomorphic Secret Sharing



Eve, Cain & Abel

$\pi(\cdot \mid \psi)$

$\pi(\psi)$

$$\pi(\psi \mid X) \propto$$

$$\mathrm{Dec}\left[k_s, \prod_{i=1}^{N} \pi(\mathbf{x}_i^\star | \mathrm{Enc}(k_p, \psi)) \times \mathrm{Enc}(k_p, \pi(\psi))\right]$$

✗ Likelihood restricted to low degree polynomials

✗ Can only handle very small *N* due to multiplicative depth

✗ MAP/posterior? How? MCMC?

$$\{\mathbf{x}_i = (x_{i1}, \ldots, x_{id})\}_{i=1}^{n_1}$$

$$\{\mathbf{x}_i = (x_{i1}, \ldots, x_{id})\}_{i=n_1+1}^{N}$$

$$\mathbf{x}_i^\star = \mathrm{Enc}(k_p, \mathbf{x}_i)$$

~~✗ Who holds secret key?~~

Approximate Bayesian Computation

Approximate Bayesian Computation

1. Sample $\psi_j \sim \pi(\psi),\ j \in \{1, \ldots, m\}$
2. For each ψ_j, simulate a dataset Y_j from $\pi(\cdot \mid \psi_j)$ of the same size, N, as X.
3. Accept ψ_j if $d(S(X), S(Y_j)) < \varepsilon$.

Where $S(\cdot)$ is some (vector) of summary statistics; $d(\cdot, \cdot)$ is a distance metric; and ε is a user defined threshold.

When $S(\cdot)$ is sufficient and $\varepsilon \to 0$, this procedure will converge to the usual Bayesian posterior.

Approximate Bayesian Computation

1. Sample $\psi_j \sim \pi(\psi),\ j \in \{1, \dots, m\}$
2. For each ψ_j, simulate a dataset Y_j from $\pi(\cdot \,|\, \psi_j)$ of the same size, N, as X.
3. Accept ψ_j if $d(S(X), S(Y_j)) < \varepsilon$.

Where $S(\cdot)$ is some (vector) of summary statistics; $d(\cdot, \cdot)$ is a distance metric; and ε is a user defined threshold.

When $S(\cdot)$ is sufficient and $\varepsilon \to 0$, this procedure will converge to the usual Bayesian posterior.

Benefit: Eve can do steps 1 & 2 and encrypt her simulated data, eliminating need for function privacy.

Approximate Bayesian Computation

1. Sample $\psi_j \sim \pi(\psi),\ j \in \{1, \ldots, m\}$
2. For each ψ_j, simulate a dataset Y_j from $\pi(\cdot \,|\, \psi_j)$ of the same size, N, as X.
3. Accept ψ_j if $d(S(X), S(Y_j)) < \varepsilon$.

Where $S(\cdot)$ is some (vector) of summary statistics; $d(\cdot, \cdot)$ is a distance metric; and ε is a user defined threshold.

When $S(\cdot)$ is sufficient and $\varepsilon \to 0$, this procedure will converge to the usual Bayesian posterior.

Benefit: Eve can do steps 1 & 2 and encrypt her simulated data, eliminating need for function privacy.

Problems: $d(\cdot, \cdot)$ can only be low degree polynomials;
Must compute $S(\cdot)$ secretly for Cain and Abel's pooled data;
Naïve ABC performs poorly & choosing ε blindfolded.

Naïve encrypted ABC (I) – Eve & data owners $1, \dots, P$

1. Eve samples $\psi_j \sim \pi(\psi),\ j \in \{1, \dots, m\}$; simulates datasets Y_j of size N from $\pi(\cdot \,|\, \psi_j)$; and computes $S(Y_j)$.

Naïve encrypted ABC (I) – Eve & data owners $1, \ldots, P$

1. Eve samples $\psi_j \sim \pi(\psi),\ j \in \{1, \ldots, m\}$; simulates datasets Y_j of size N from $\pi(\cdot \mid \psi_j)$; and computes $S(Y_j)$.

2. Eve computes HSS shares $S^{\star p}(Y_j)$, $p \in \{1, \ldots, P+1\}$;
   - send $S^{\star p}(Y_j)$ to data owner p
   - retain $S^{\star P+1}(Y_j)$

Naïve encrypted ABC (I) – Eve & data owners $1, \ldots, P$

1. Eve samples $\psi_j \sim \pi(\psi),\ j \in \{1, \ldots, m\}$; simulates datasets Y_j of size N from $\pi(\cdot \mid \psi_j)$; and computes $S(Y_j)$.
2. Eve computes HSS shares $S^{\star p}(Y_j)$, $p \in \{1, \ldots, P+1\}$;
   - send $S^{\star p}(Y_j)$ to data owner p
   - retain $S^{\star P+1}(Y_j)$
3. Data owners $k \in \{1, \ldots, P\}$ create HSS shares $S^{\star p}(X_k)$, $p \in \{1, \ldots, P+1\}$
   - send $S^{\star p}(X_k)$ to data owner p (retaining when $p = k$)
   - send $S^{\star P+1}(X_k)$ to Eve

Naïve encrypted ABC (I) – Eve & data owners $1, \ldots, P$

1. Eve samples $\psi_j \sim \pi(\psi),\ j \in \{1, \ldots, m\}$; simulates datasets Y_j of size N from $\pi(\cdot \mid \psi_j)$; and computes $S(Y_j)$.

2. Eve computes HSS shares $S^{\star p}(Y_j)$, $p \in \{1, \ldots, P+1\}$;
   - send $S^{\star p}(Y_j)$ to data owner p
   - retain $S^{\star P+1}(Y_j)$

3. Data owners $k \in \{1, \ldots, P\}$ create HSS shares $S^{\star p}(X_k)$, $p \in \{1, \ldots, P+1\}$
   - send $S^{\star p}(X_k)$ to data owner p (retaining when $p = k$)
   - send $S^{\star P+1}(X_k)$ to Eve

4. All compute $S^{\star p}(X) = \tilde{S}\left(\bigcup_k S^{\star p}(X_k)\right)$, where $\tilde{S}(\cdot)$ is a **homomorphically computable pooling function**.

Naïve encrypted ABC (I) – Eve & data owners $1, \ldots, P$

1. Eve samples $\psi_j \sim \pi(\psi),\ j \in \{1, \ldots, m\}$; simulates datasets Y_j of size N from $\pi(\cdot \,|\, \psi_j)$; and computes $S(Y_j)$.

2. Eve computes HSS shares $S^{\star p}(Y_j)$, $p \in \{1, \ldots, P+1\}$;
   - send $S^{\star p}(Y_j)$ to data owner p
   - retain $S^{\star P+1}(Y_j)$

3. Data owners $k \in \{1, \ldots, P\}$ create HSS shares $S^{\star p}(X_k)$, $p \in \{1, \ldots, P+1\}$
   - send $S^{\star p}(X_k)$ to data owner p (retaining when $p = k$)
   - send $S^{\star P+1}(X_k)$ to Eve

4. All compute $S^{\star p}(X) = \tilde{S}\left(\bigcup_k S^{\star p}(X_k)\right)$, where $\tilde{S}(\cdot)$ is a **homomorphically computable pooling function**.

5. All compute $d_j^{\star p} = d\left(S^{\star p}(X), S^{\star p}(Y_j)\right)$, where $d(\cdot)$ is a **homomorphically computable distance metric**.

Naïve encrypted ABC (II) – Eve & data owners $1, \ldots, P$

6. All send their shares, $d_j^{\star p}$, to a randomly chosen data owner $k \in 1, \ldots, P$

Naïve encrypted ABC (II) – Eve & data owners $1, \ldots, P$

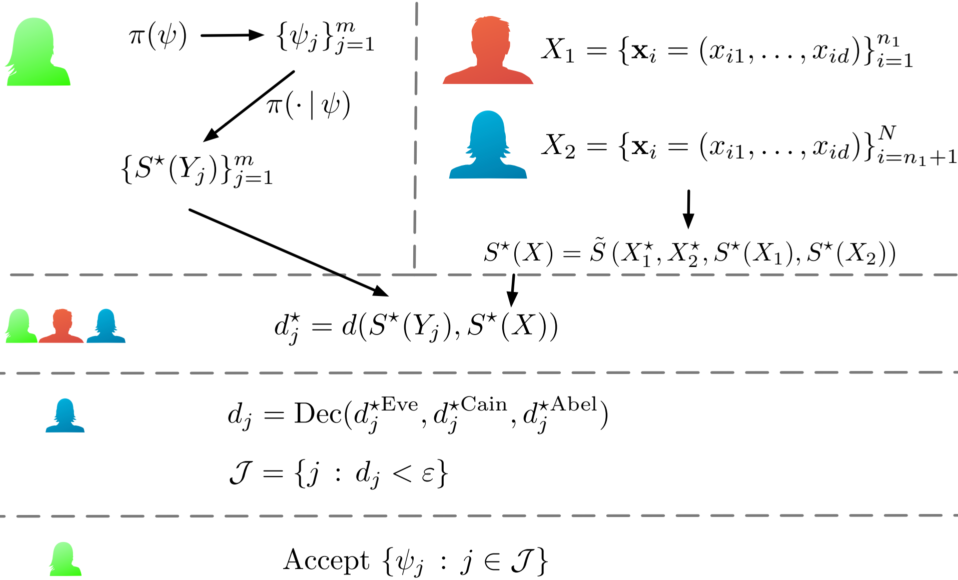
6. All send their shares, $d_j^{\star p}$, to a randomly chosen data owner $k \in 1, \ldots, P$

7. Data owner k reconstructs $d_j = \text{Dec}(d_j^{\star 1}, \ldots, d_j^{\star P+1})$

Naïve encrypted ABC (II) – Eve & data owners $1, \dots, P$

6. All send their shares, $d_j^{\star p}$, to a randomly chosen data owner $k \in 1, \dots, P$
7. Data owner k reconstructs $d_j = \text{Dec}(d_j^{\star 1}, \dots, d_j^{\star P+1})$
8. Data owner k sends to Eve a list of those indices j such that $d_j < \varepsilon$.

Naïve encrypted ABC (III) – in pictures



$$\pi(\psi) \longrightarrow \{\psi_j\}_{j=1}^{m}$$

$$\pi(\cdot \mid \psi)$$

$$\{S^\star(Y_j)\}_{j=1}^{m}$$

$$X_1 = \{\mathbf{x}_i = (x_{i1}, \dots, x_{id})\}_{i=1}^{n_1}$$

$$X_2 = \{\mathbf{x}_i = (x_{i1}, \dots, x_{id})\}_{i=n_1+1}^{N}$$

$$S^\star(X) = \tilde{S}\left(X_1^\star, X_2^\star, S^\star(X_1), S^\star(X_2)\right)$$

$$d_j^\star = d(S^\star(Y_j), S^\star(X))$$

$$d_j = \mathrm{Dec}(d_j^{\star\mathrm{Eve}}, d_j^{\star\mathrm{Cain}}, d_j^{\star\mathrm{Abel}})$$

$$\mathcal{J} = \{j \,:\, d_j < \varepsilon\}$$

Accept $\{\psi_j \,:\, j \in \mathcal{J}\}$

Points to note

- Samples ψ_j are never seen by Cain and Abel
- Eve learns only an accept/reject
  - Final distances between summary statistics decrypted by Cain or Abel
- Cain and Abel do not learn about each other's data
  - only see composite distance between pooled summary stats and Eve's simulation
  - can make distances information theoretically secure by adding random values generated by Cain, Abel and Eve
- **BUT**, Cain and Abel do have to know $S(\cdot)$, which in most ABC settings is model dependent $\implies$ risk to Eve

Obstacles to cryptographic ABC

- Homomorphically computable pooling of summary statistics
- Summary statistics that don't reveal model
- Homomorphically computable distance metric
- Blindfold selection of ε

Obstacles to cryptographic ABC

- Homomorphically computable pooling of summary statistics
- Summary statistics that don't reveal model
- Homomorphically computable distance metric
- Blindfold selection of ε
    - Propose using ABC-PMC/SMC, with distance chosen to retain $\alpha\%$ of samples instead. Eve then uses accepted ψ_j on step t to propose step $t+1$ and repeat algorithm.
    - Standard idea — details omited.

Cryptographically Secure Inference

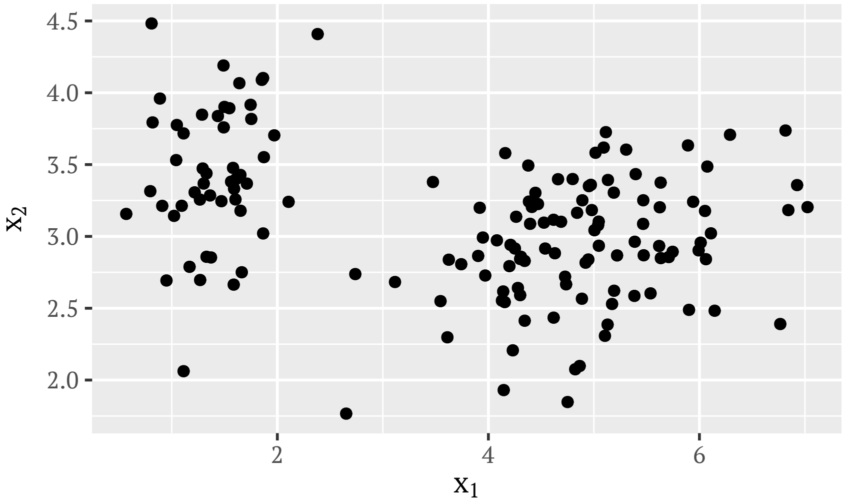
Collection of Coarse Random Marginals (CCRM)

Construct in the manner of a decision forest:

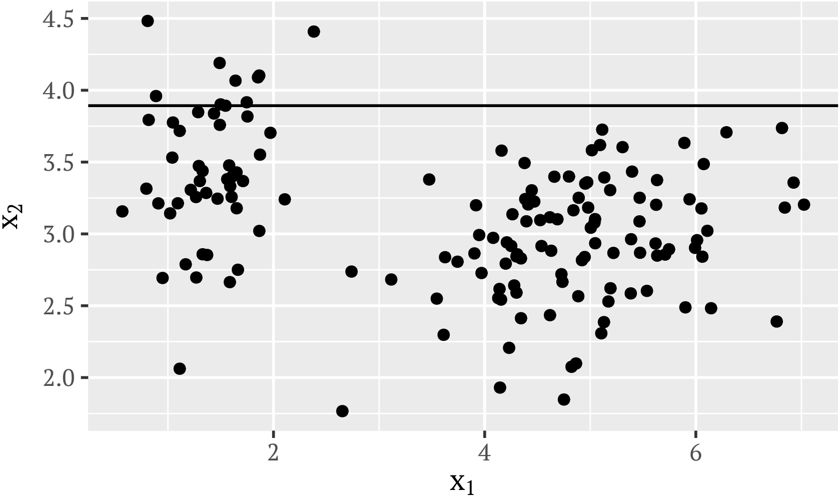
- Grow T trees, each to predetermined fixed depth L
- Choose variable $v \in \{1, \dots, d\}$ uniformly at random
- Each split point uniformly at random in range of $x_{\cdot v}$
  - Thus Cain and Abel must provide range of each variable in the data, though this range need not be tight
  - e.g. release $(\min_i x_{iv} + \eta, \max_i x_{iv} + \eta)$ for $\eta \sim N(0, \sigma^2)$ with σ^2 chosen not to exclude too large a range
- s $= S(\cdot)$ is then the counts of observations in each terminal leaf
  - vector of $T2^L$ counts
  - $\tilde{S}(\cdot)$ is then simply vector addition
- Define

$$d(S(X), S(Y_j)) = \sum_{i=1}^{T2^L} \left(s_i^X - s_i^{Y_j} \right)^2$$

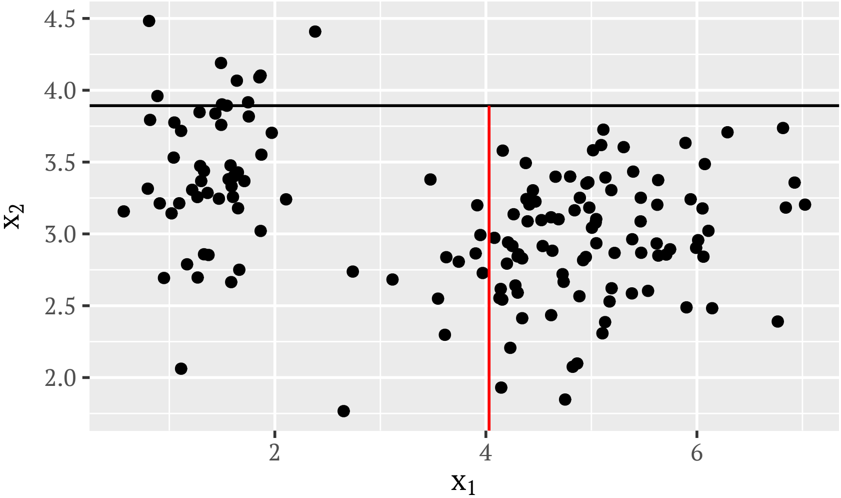
Collection of Coarse Random Marginals (CCRM)



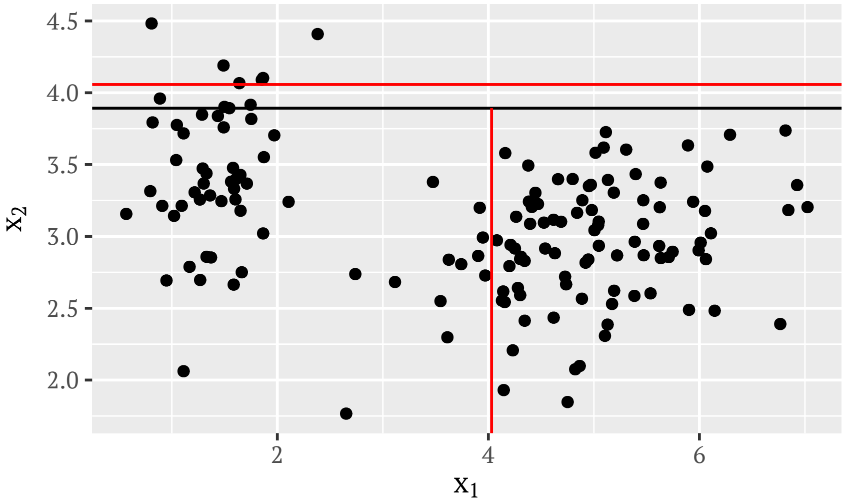
Collection of Coarse Random Marginals (CCRM)



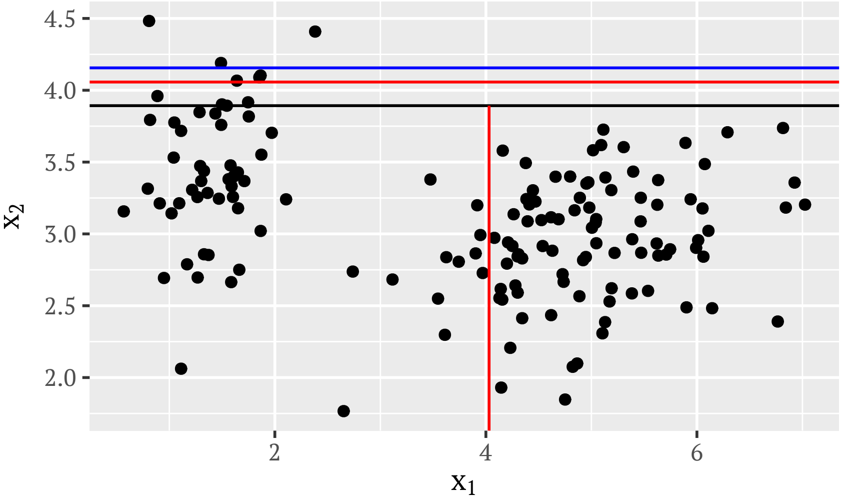
Collection of Coarse Random Marginals (CCRM)



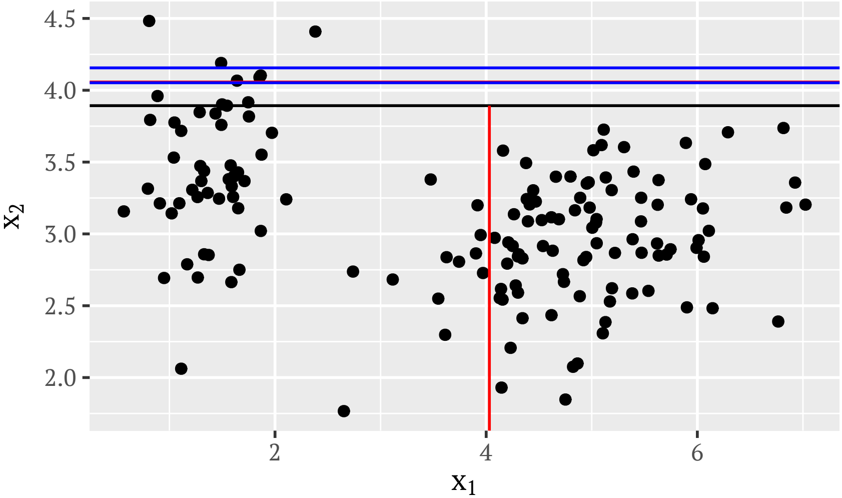
Collection of Coarse Random Marginals (CCRM)



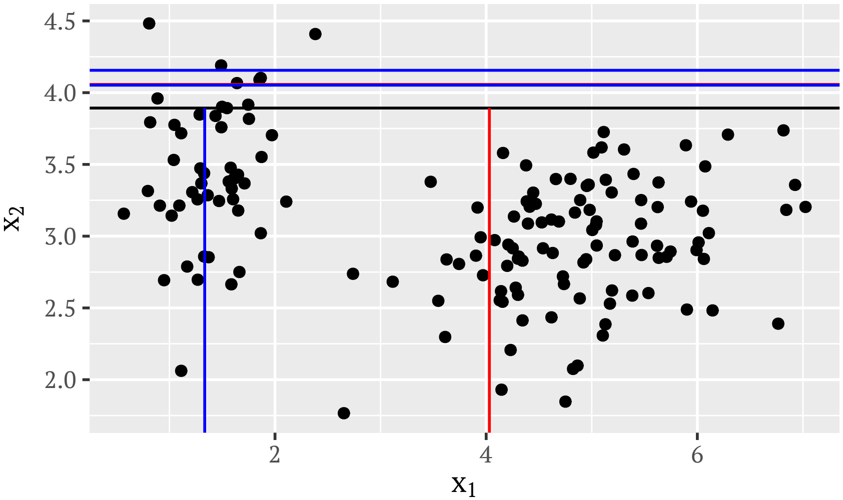
Collection of Coarse Random Marginals (CCRM)



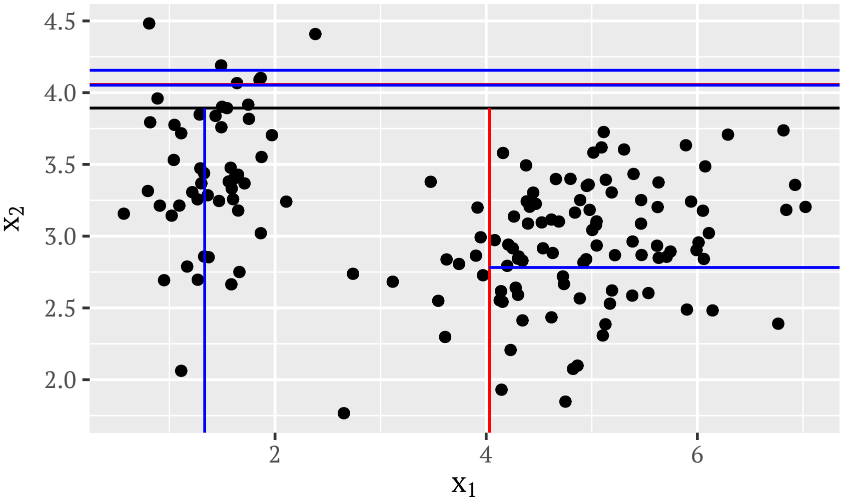
Collection of Coarse Random Marginals (CCRM)



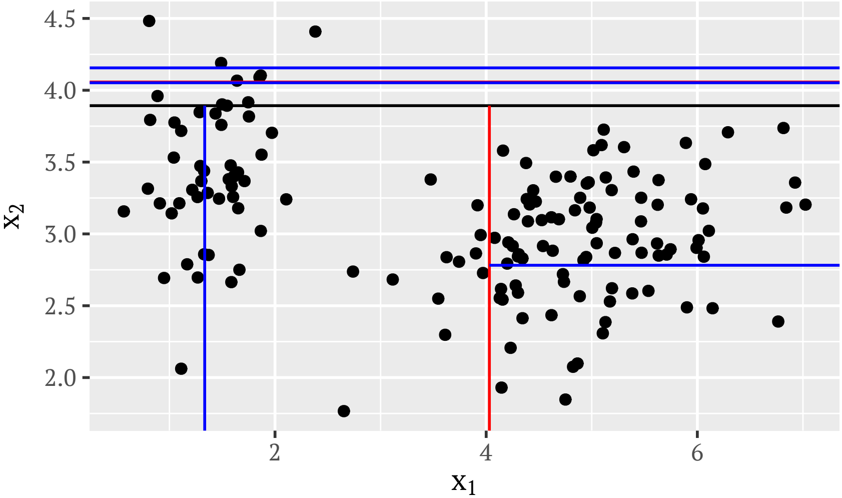
Collection of Coarse Random Marginals (CCRM)



Collection of Coarse Random Marginals (CCRM)



Collection of Coarse Random Marginals (CCRM)



$$S(X) = (\dots, 3, 3, 0, 3, 43, 33, 64, 24, \dots)$$

CCRM solutions

- Homomorphically computable pooling of summary statistics
  - **simple vector addition**
- Summary statistics that don't reveal model
  - **CCRM is completely random, grown the same way for all models and data sets. Only weak information about range of each variable leaked.**
- Homomorphically computable distance metric
  - **sum of squared differences**

Variance of distance metric per CRM

Lemma *Let the random variable* V *be multinomially distributed with success probabilities* $p = (p_1, \ldots, p_k)$ *for* n *trials. Then,*

$$
\begin{aligned}
&\operatorname{Var}\left(\sum_{i=1}^{k}(V_i - c_i)^2\right) \\
&= \sum_{i=1}^{k}\Bigg[\left({}^{n}C_{n-4} - n^2(n-1)^2\right)p_i^4 + \left(6\,{}^{n}C_{n-3} + 2n(n-1)(4c_i - n)\right)p_i^3 \\
&\quad + \left(7n(n-1) - n^2 - 4c_i n(2n-3)(1+c_i)\right)p_i^2 + \left(n + 4c_i n(c_i - 1)\right)p_i \\
&\quad + \sum_{\substack{j=1 \\ i\neq j}}^{k}\Big[-n(2c_i-1)(2c_j-1)p_ip_j + 2n(n-1)(2c_j-1)p_i^2p_j \\
&\quad + 2n(n-1)(2c_i-1)p_ip_j^2 - 2n(n-1)(2n-3)p_i^2p_j^2\Big]\Bigg]
\end{aligned}
$$

$\Longrightarrow$ can be used to weight random marginals differently.

ABCDE: Approximate Bayesian Computation Done Encrypted

Tying it all together:

- ABC-PMC/SMC
- Homomorphic Secret Sharing with data pooling
- CCRM summary statistic protecting model/prior privacy
- Pooled $S(\cdot)$ computable encrypted from multiple data owners
- Distance computable encrypted and not learned by modeller
- Variance of each CRM computable encrypted for weighting

Selected connections in ABC literature

- Bernton, E., Jacob, P. E., Gerber, M., & Robert, C. P. (2017). Inference in generative models using the Wasserstein distance. *arXiv:1701.05146*.

- Gutmann, M. U., Dutta, R., Kaski, S., & Corander, J. (2017). Likelihood-free inference via classification. *Statistics and Computing*, 1-15.

- Fearnhead, P., & Prangle, D. (2012). Constructing summary statistics for approximate Bayesian computation: semi-automatic approximate Bayesian computation. *Journal of the Royal Statistical Society: Series B*, 74(**3**), 419-474.

Examples

Toy example

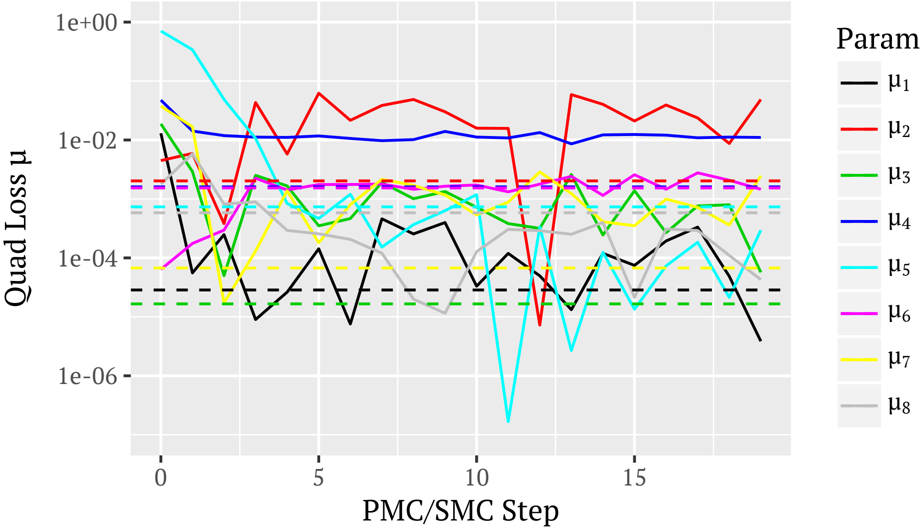
Super simple first example, 8-dimensional multivariate Normal.

$$X \sim \mathrm{N}(\boldsymbol{\mu} = \mathbf{0}, \Sigma = I)$$
$$\mu_i \sim \mathrm{N}(\eta_i, \sigma = 2)$$

where η_i chosen independently uniformly at random on the interval $[-1, 1]$ for repeated experiments.

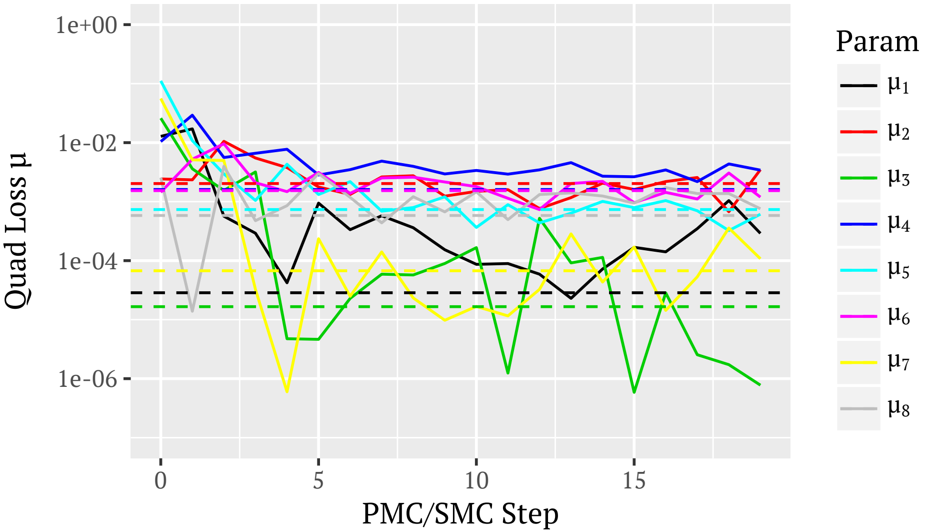
- Simulate $n = 1000$ observations
- Range of all dimensions taken to be $[-4, 4]$ for construction of CCRM, without checking true range of X
- Standard ABC used $S(X) = (\bar{x}_1, \ldots, \bar{x}_8)$

Toy example: 8D Normal, marginal quadratic loss



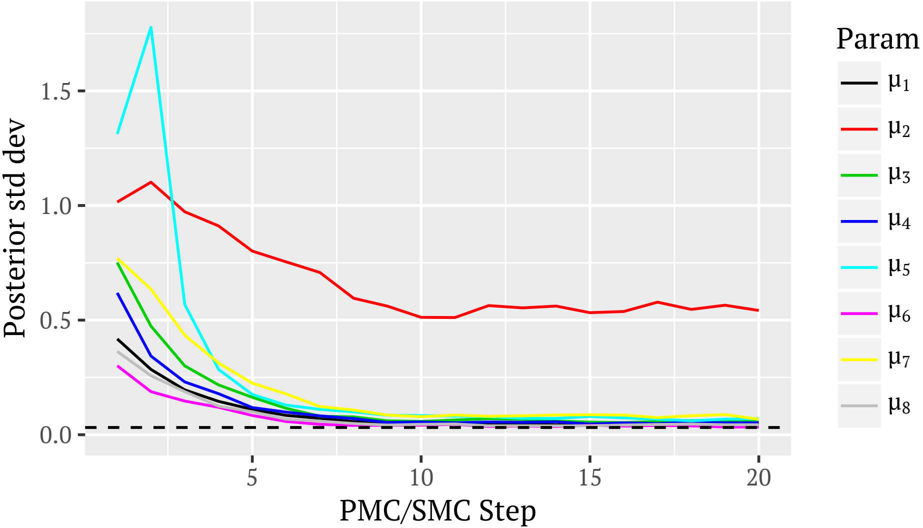
$$n = 10^3, T = 20, L = 2, m = 10^4, \alpha = 0.01$$

Toy example: 8D Normal, marginal quadratic loss



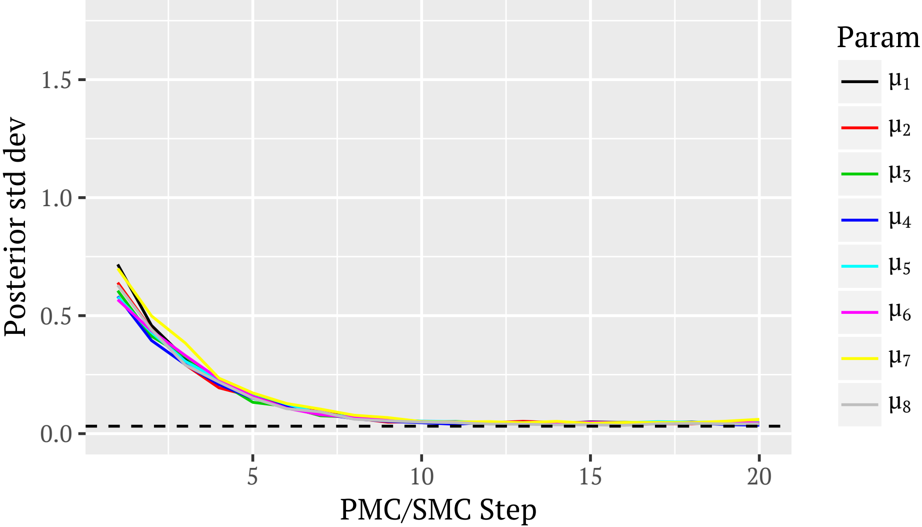
$$n = 10^3, T = 1000, L = 2, m = 10^4, \alpha = 0.01$$

Toy example: 8D Normal, marginal posterior σ



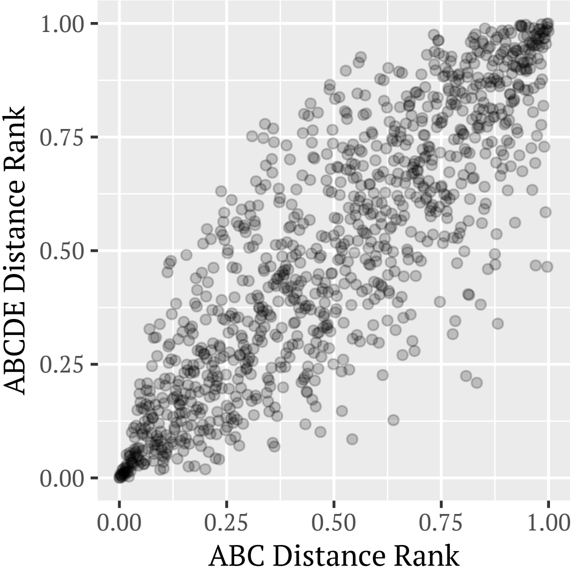
$$n = 10^3, T = 20, L = 2, m = 10^4, \alpha = 0.01$$

Toy example: 8D Normal, marginal posterior σ



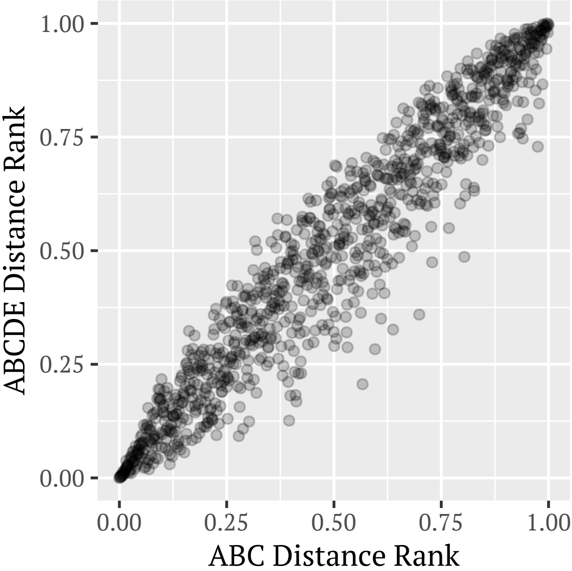
$$n = 10^3, T = 1000, L = 2, m = 10^4, \alpha = 0.01$$

Toy example: distance concordance



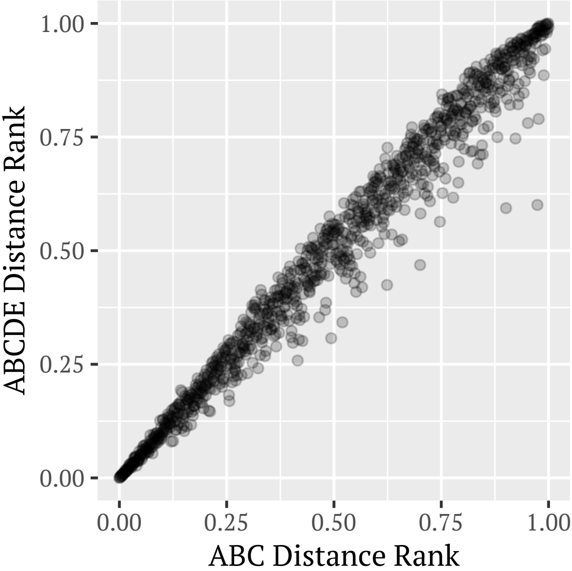
$T = 20$

Toy example: distance concordance



$T = 100$

Toy example: distance concordance



$T = 1000$

Expected quadratic loss

Can understand lowest ABC error achievable without Monte Carlo error:

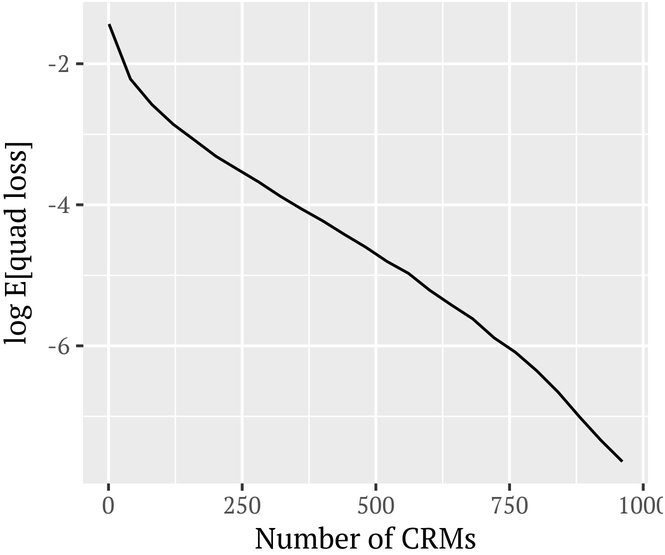
$$\mathbb{E}\left[(\mu - \hat{\mu})^2 \,|\, T = t\right]$$
$$= \frac{1}{|\mathcal{A}^t|} \int_{\mathcal{A}^t} \left(\mu - \int_{-\infty}^{\infty} \theta \, \mathbb{P}\left(S(x) = S(x^{\text{obs}}) \,|\, da_1, \dots, da_t \right) \, \pi(d\theta) \right)^2$$

because for 1-level CRMs:

$$\mathbb{P}\left(S(x) = S(x^{\text{obs}}) \,|\, da_1, \dots, da_t \right)$$
$$= \prod_{k=1}^{t} \binom{n}{m_k} F_{v_k}(X < a_k)^{m_k} (1 - F_{v_k}(X < a_k))^{n-m_k}$$

where $m_k = \#\{i : x_i^{\text{obs}} < a_k\}$.

Expected quadratic loss



g-and-k distribution (Haynes et al. 1997)

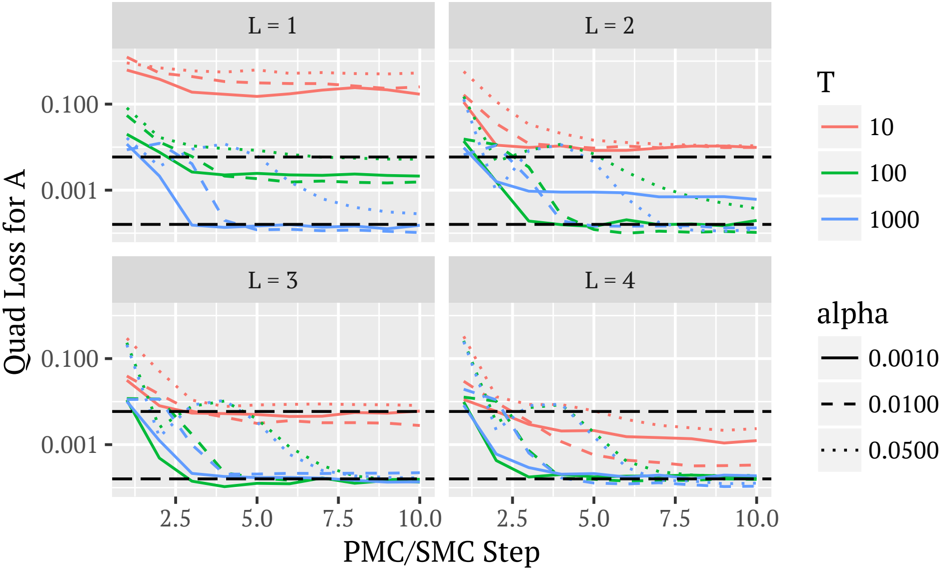
Defined via inverse distribution function

$$F^{-1}(x \,|\, A, B, g, k) = \\ A + B \left[1 + 0.8 \frac{1 - \exp\left(-g\Phi^{-1}(x)\right)}{1 + \exp\left(-g\Phi^{-1}(x)\right)}\right] \left(1 + \Phi^{-1}(x)^2\right)^k \Phi^{-1}(x)$$

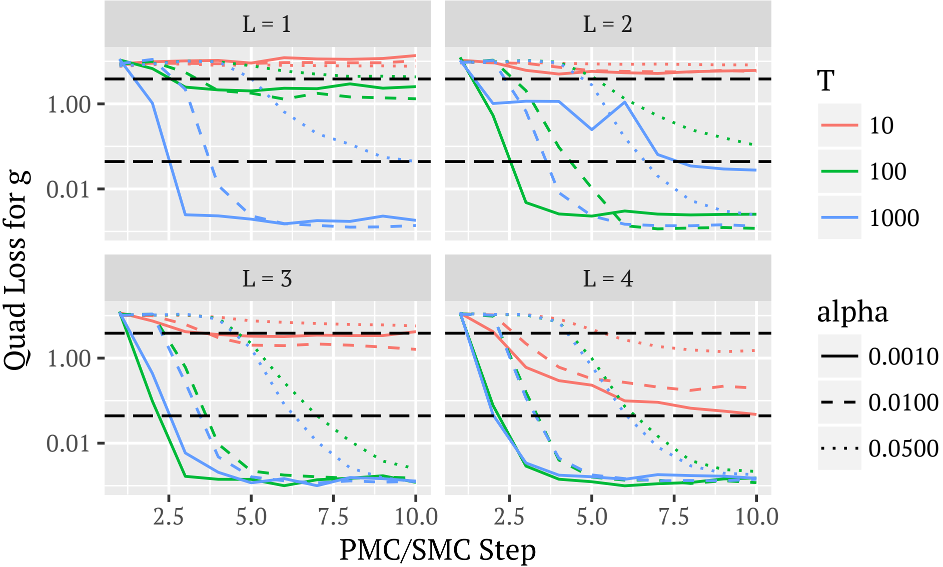
Following Allingham et al. (2009) and Fearnhead & Prangle (2012), take:

- $A = 3, B = 1, g = 2, k = \frac{1}{2}$
- simulate $n = 10000$ observations
- standard ABC uses the order statistics, $S(X) = (x_{(1)}, \dots, x_{(n)})$

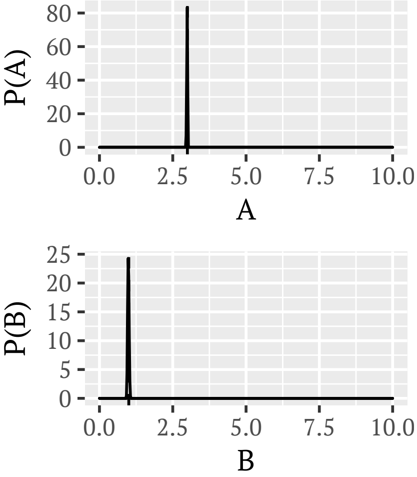
g-and-k: quadratic loss



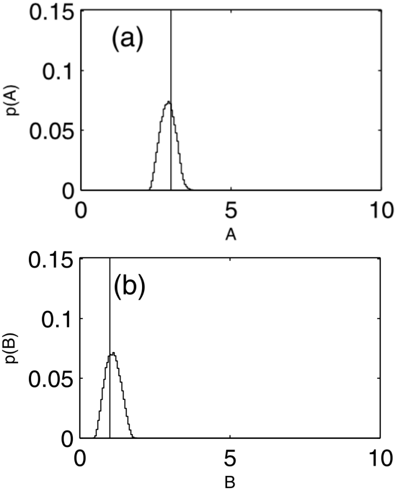
g-and-k: quadratic loss



g-and-k: density plots

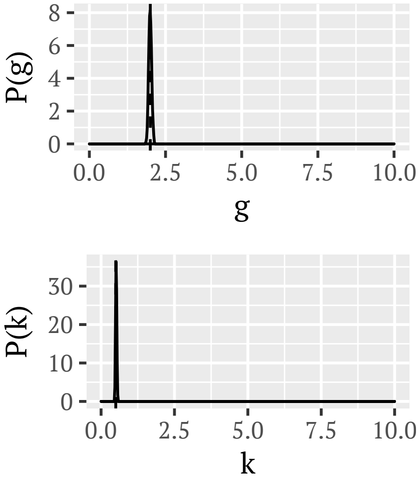


$T = 1000, L = 3, m = 10^5, \alpha = 0.01$

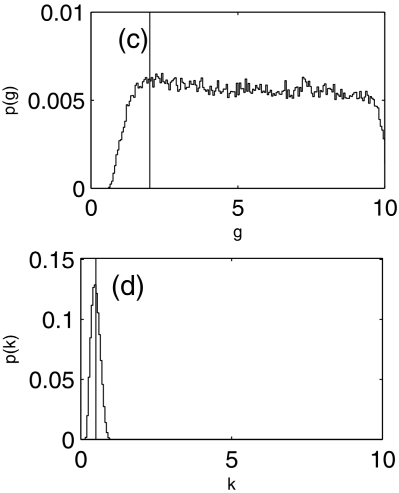


Allingham et al (2009)

g-and-k: density plots



$T = 1000, L = 3, m = 10^5, \alpha = 0.01$



Allingham et al (2009)

Tuberculosis Transmission (Tanaka et al. 2006)

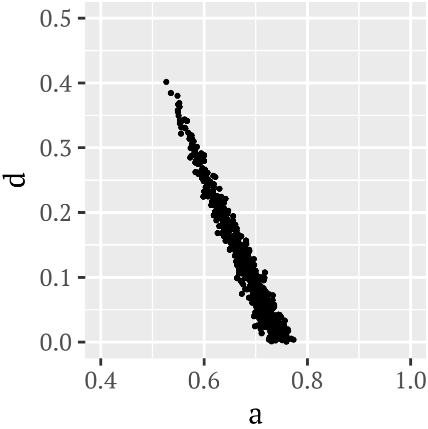
Model of transmission of disease,

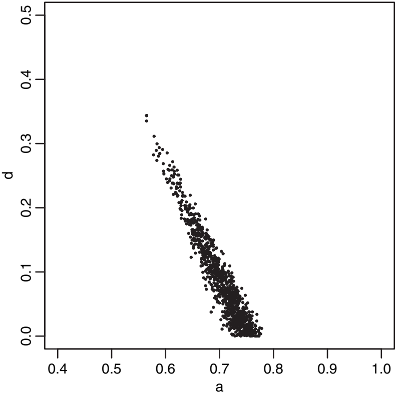
- ‘birth’ of new infections, rate α
- ‘death’ recovery or mortality of carrier, rate δ
- ‘mutation’ genotype of bacterium mutates within carrier, rate θ (infinite-alleles assumption)

$X_i(t)$ num infections type i at time t; $G(t)$ num unique genotypes.

- San Francisco tuberculosis data 1991/2, 473 samples (no time)
- Fearnhead & Prangle (2012) transform $(\alpha/(\alpha+\delta+\theta), \delta/(\alpha+\delta+\theta))$
- $S(X) = (G(t_{\text{end}})/473, 1 - \sum_i (X(t_{\text{end}})/473)^2)$

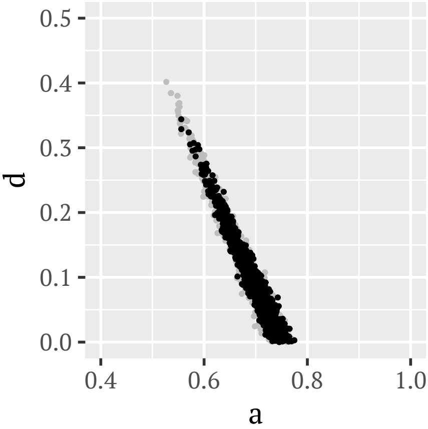
Posterior samples

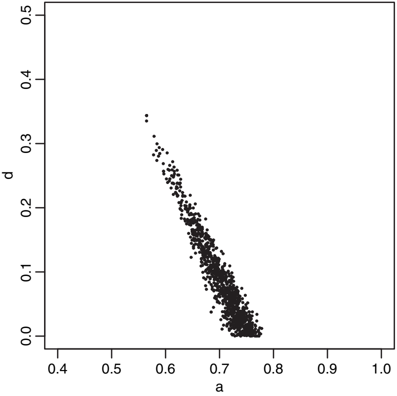




Semi-automatic ABC

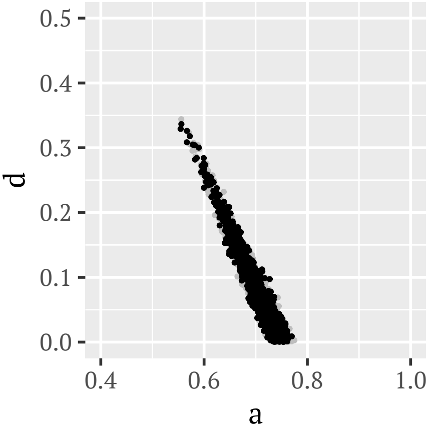
Posterior samples

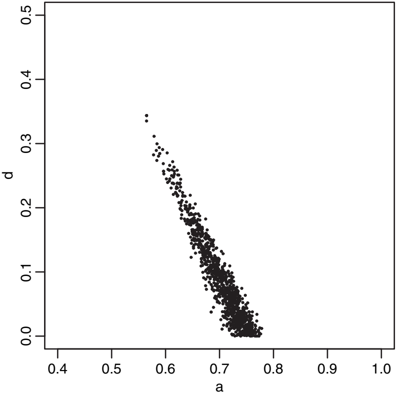




Semi-automatic ABC

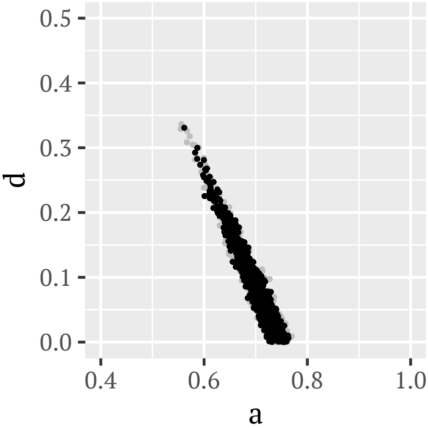
Posterior samples

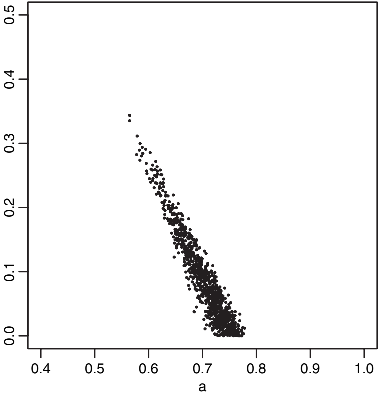




Semi-automatic ABC

Posterior samples





Semi-automatic ABC

Conclusions

- So far, this …
  - Provides encrypted inference whilst preserving model, prior and data privacy
  - Enables pooling of multiple data owners
  - Theoretically arbitrary low-dimensional models
- … but this is work-in-progress! Currently in progress:
  - Method of ensuring differential privacy
  - Encrypted software implementation of this scheme
  - Best use of weights
  - Fuller understanding of accuracy for CCRM choices
  - Data as a service
- Perhaps also useful as a model independent summary statistic for unencrypted ABC too? **Scalable!**
- Questions, comments and discussion over tea welcome!

Conclusions

- So far, this …
  - Provides encrypted inference whilst preserving model, prior and data privacy
  - Enables pooling of multiple data owners
  - Theoretically arbitrary low-dimensional models
- … but this is work-in-progress! Currently in progress:
  - Method of ensuring differential privacy
  - Encrypted software implementation of this scheme
  - Best use of weights
  - Fuller understanding of accuracy for CCRM choices
  - Data as a service
- Perhaps also useful as a model independent summary statistic for unencrypted ABC too? **Scalable!**
- Questions, comments and discussion over tea welcome!

Thank you!