4.5 Approximate Bayesian Computation (ABC)

Likelihood-free inference via rejection from the prior

Where we are so far: likelihood-based Bayes

So far, our Bayesian workflows assumed the likelihood is **easy to work with**:

- We can **evaluate** $\pi(y \mid \theta)$ at many values of θ.
- We can evaluate it **cheaply**, so MCMC / optimisation are feasible.

But sometimes:

- the likelihood is **not available in closed form**, or
- it is **too expensive** to evaluate.

Terminology: methods that do inference without evaluating $\pi(y \mid \theta)$ are called likelihood-free inference.

Motivating example: complex weather models (Example 4.6)

Weather forecasting models can be extremely complex:

- huge numbers of parameters,
- complicated dynamics,
- potentially no exact tractable likelihood.

Even if a likelihood can be written down in principle, it may be impractical:

- evaluating $\pi(y \mid \theta)$ may be too slow,
- and re-running a full MCMC pipeline every time new data arrive is costly.

Key idea for ABC: often we *can* still **simulate data** from the model for a given θ, even if we cannot compute the likelihood.

ABC in one sentence

Approximate Bayesian Computation (ABC) is a likelihood-free approach that replaces

$$\text{“evaluate likelihood of observed data”} \quad \longrightarrow \quad \text{“simulate fake data and compare to observed data”}.$$

ABC typically needs:

- a prior $\pi(\theta)$,
- a simulator $y^\star \sim \pi(\cdot \mid \theta)$ (data-generating process),
- a notion of **closeness** (distance and tolerance ε).

Today: two basic ABC variants:

1. ABC with rejection (compare full data),
2. Summary ABC with rejection (compare summary statistics).

ABC with rejection: algorithm (Definition 4.2)

Goal: approximate the posterior $\pi(\theta \mid y)$ without computing $\pi(y \mid \theta)$.

ABC rejection algorithm:

1. Sample $\theta^\star \sim \pi(\theta)$ (from the prior).
2. Simulate $y^\star \sim \pi(\cdot \mid \theta^\star)$ (generate synthetic data).
3. Accept $\theta^\star$ if $\|y - y^\star\| < \varepsilon$ for some $\varepsilon > 0$, otherwise reject.
4. Repeat steps 1–3 until you have enough accepted samples.

Interpretation:

- We keep parameter values that can generate data *similar* to what we observed.
- Similarity is controlled by a **tolerance** ε.

What distribution are we sampling from? (Definition 4.3)

Define the acceptance region

$$A_\varepsilon(y^\star) = \{y^\star \ : \ \|y^\star - y\| < \varepsilon\}\,.$$

Then the ABC-rejection samples target the **approximate posterior**

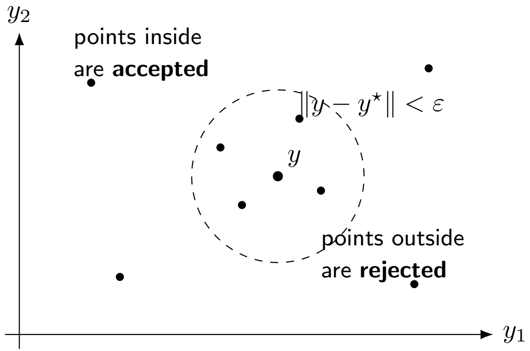
$$\pi_\varepsilon(\theta \mid y) \ \propto \int \pi(y^\star \mid \theta)\,\pi(\theta)\,\mathbf{1}_{A_\varepsilon}(y^\star)\,dy^\star.$$

Plain English:

- For each θ, we look at the probability that simulated data $y^\star$ lands within distance ε of the observed data y.
- Multiply by the prior, and renormalise.

A picture: acceptance as a “ball” around the data

Think of y and $y^\star$ as vectors in $\mathbb{R}^n$.



Key message: as dimension n grows, it becomes much harder for random $y^\star$ to fall inside a small ε-ball.

Example 4.7: model setup

We observe data

$$y_1, \ldots, y_{10} \sim \mathrm{Beta}(3, \beta).$$

We place a prior on the unknown parameter β:

$$\beta \sim \mathrm{Unif}[0, 5].$$

ABC idea here:

- propose $\beta^\star$ from the prior,
- simulate synthetic data $y_1^\star, \ldots, y_{10}^\star$ from $\mathrm{Beta}(3, \beta^\star)$,
- accept if simulated data is "close" to observed data.

Example 4.7: distance and acceptance rule

Choose a distance (here: squared error across all data points):

$$D = \sum_{i=1}^{10} \left(y_i - y_i^\star\right)^2.$$

Acceptance rule:

$$\text{accept } \beta^\star \quad \text{if} \quad D < \varepsilon,$$

with (in the example) $\varepsilon = 0.75$.

Notes:

- This D is just one choice; many distances are possible.
- ε controls the approximation–computation trade-off.

Example 4.7: R code (ABC rejection)

```
# Set Up Example
set.seed(1234)
n <- 10
y <- rbeta(n, 3, 2)

# Set Up ABC
n.iter <- 50000
b.store <- numeric(n.iter)
epsilon <- 0.75

# Run ABC
for(i in 1:n.iter){

  # Propose new beta
  b <- runif(1, 0, 5)

  # Simulate data
  y.star <- rbeta(n, 3, b)
```

Example 4.7: reading the output

In the provided run (with $\varepsilon = 0.75$):

- many proposals are rejected (acceptance rate is low),
- the histogram of accepted $\beta^\star$ values approximates the posterior shape,
- the red vertical line at $\beta = 2$ is the **true** value used to generate y (only for this toy demo).

You can summarise the approximate posterior via:

- posterior mean (sample mean of accepted draws),
- credible intervals (quantiles of accepted draws).

But: all results can change substantially with ε.

The big practical question: how do we choose ε?

Choosing ε is **application-specific** and often difficult.

Two competing effects:

- **Small** $\varepsilon \Rightarrow$ better approximation to $\pi(\theta \mid y)$, but very low acceptance (slow / few samples).
- **Large** $\varepsilon \Rightarrow$ high acceptance, but poor approximation (can collapse back to the prior).

In practice, common diagnostics are:

- acceptance rate (how many draws are not `NA`),
- comparing histograms of the approximate posterior vs the prior,
- stability of posterior summaries across nearby ε values.

Example 4.8: very small tolerance ($\varepsilon = 0.12$)

With $\varepsilon = 0.12$, almost all proposals are rejected.

```r
epsilon <- 0.12

# ... same loop ...
sum(is.na(b.store)) # almost all rejected
hist(b.store, freq = FALSE, xlab = expression(beta), main = "")
abline(v = 2, col = "red")

mean(b.store, na.rm = TRUE)
quantile(b.store, c(0.025, 0.975), na.rm = TRUE)
```

Interpretation:

- The few accepted samples might be close to the truth,
- but with tiny sample size, the histogram / quantiles are very noisy.

Example 4.9: very large tolerance ($\varepsilon = 2$)

With $\varepsilon = 2$, almost all proposals are accepted.

```
epsilon <- 2

# ... same loop ...
sum(is.na(b.store)) # very few rejected
hist(b.store, freq = FALSE, xlab = expression(beta), main = "")
abline(v = 2, col = "red")
```

Interpretation:

- If almost everything is accepted, the accepted $\beta^\star$ values look like the prior,
- meaning we have learned very little from the data.

Limiting behaviour (Proposition 4.2)

ABC with rejection interpolates between the prior and the true posterior:

$$\lim_{\varepsilon \to \infty} \pi_\varepsilon(\theta \mid y) \overset{D}{=} \pi(\theta), \qquad \lim_{\varepsilon \to 0} \pi_\varepsilon(\theta \mid y) \overset{D}{=} \pi(\theta \mid y).$$

Why this makes sense:

- If ε is huge, almost every $y^\star$ counts as "close" $\Rightarrow$ we accept almost every $\theta^\star$ from the prior.
- If ε is tiny, only $\theta^\star$ that can generate (almost) the exact observed data are accepted $\Rightarrow$ we recover the true posterior in the limit.

A practical trade-off

The proposition suggests: smaller ε is better.

But in real computation:

- Very small ε can mean **near-zero acceptance**, leading to:
    - huge run time, or
    - so few accepted samples that Monte Carlo noise dominates.
- Larger ε increases acceptance but can blur the posterior and bias inference.

Rule of thumb: monitor both

acceptance rate   and   how different the posterior looks from the prior.

Why ABC with full data struggles: curse of dimensionality

ABC rejection compares full datasets via $\|y - y^\star\|$.

As the number of data points n increases:

- y and $y^\star$ live in higher-dimensional space,
- the probability that a random $y^\star$ lands close to y becomes tiny,
- to accept anything we often must increase ε,
- which **degrades** the approximation.

Example 4.10 (conceptual): for the Beta example with $n = 200$, we may need $\varepsilon > 15$ just to get non-zero acceptance *when comparing full data*.

Idea: compare summary statistics instead

Instead of comparing the entire datasets, compare a **summary**:

$$S(y) \in \mathbb{R}^k,$$

where k is small (e.g. $k = 1$ for the mean).

Benefit:

- lower-dimensional matching is easier,
- acceptance rates improve dramatically for large n.

Cost:

- we introduce an *additional approximation* because $S(y)$ throws away information.

Summary ABC with rejection: algorithm (Definition 4.4)

Summary ABC rejection algorithm:

1. Sample $\theta^\star \sim \pi(\theta)$.
2. Simulate $y^\star \sim \pi(\cdot \mid \theta^\star)$.
3. Accept $\theta^\star$ if
$$\|S(y) - S(y^\star)\| < \varepsilon,$$
otherwise reject.
4. Repeat.

Key difference from basic ABC: we match in the space of summary statistics.

Approximate posterior for Summary ABC (Proposition 4.3)

Define acceptance set in summary space:

$$A_\varepsilon(y^\star) = \{y^\star : \ \|S(y^\star) - S(y)\| < \varepsilon\} .$$

Then the Summary-ABC approximate posterior can be written as

$$\pi_\varepsilon(\theta \mid S(y)) \ \propto \ \int \pi(y^\star \mid \theta)\, \pi(\theta)\, \mathbf{1}_{A_\varepsilon}(y^\star)\, dy^\star .$$

Same structure as before, but with acceptance defined using $S(\cdot)$ rather than the full data.

Sufficient statistics: when Summary ABC can be exact

Using summaries generally increases approximation, *unless* the summary contains all information about θ.

Definition 4.5 (Sufficient statistic): A statistic S is sufficient for θ if the conditional distribution

$$\pi\big(y \mid S(y)\big)$$

does **not** depend on θ.

Intuition:

- Once you know $S(y)$, the remaining details of y are irrelevant for learning θ.

Consequence (Proposition 4.4)

If S is sufficient, Summary ABC can recover the true posterior in the small-tolerance limit:

$$\lim_{\varepsilon \to 0} \pi_\varepsilon(\theta \mid S(y)) \overset{D}{=} \pi(\theta \mid y).$$

Reality check:

- Sufficient statistics typically exist only for “nice” families (e.g. Beta, Gamma, Poisson with standard sampling models).
- In those cases we often can do exact Bayes anyway, so ABC is not necessary.

Example 4.11: Beta example with mean as summary

Repeat the Beta example but now with:

- $n = 200$ observations,
- summary statistic $S(y) = \bar{y}$ (sample mean),
- tolerance $\varepsilon = 0.001$.

Important: the mean is *not* sufficient for β in $\mathrm{Beta}(3, \beta)$, so there is an additional approximation even as $\varepsilon \to 0$.

Example 4.11: R code (Summary ABC with mean)

```r
set.seed(1234)
n <- 200
y <- rbeta(n, 3, 2)

n.iter <- 50000
b.store <- numeric(n.iter)
epsilon <- 0.001

for(i in 1:n.iter){

  b <- runif(1, 0, 5)
  y.star <- rbeta(n, 3, b)

  # summary distance in 1D (mean)
  d <- (mean(y) - mean(y.star))^2

  if(d < epsilon){
    b.store[i] <- b
```

What improved, and what did we pay?

With summary statistics:

- **Improvement:** matching in low dimension is easier, so we can use a much smaller ε and still accept a reasonable number of draws.
- **Cost:** we are no longer targeting $\pi(\theta \mid y)$ but rather something closer to

  $$\pi(\theta \mid S(y)),$$

  which can lose information when S is not sufficient.

Takeaway:

- Summary ABC reduces the curse of dimensionality,
- but forces you to think carefully about which summaries preserve the information you care about.

Key takeaways (4.5)

- ABC is **likelihood-free**: it replaces likelihood evaluation by **simulation + comparison**.
- ABC rejection:
  - propose $\theta^\star$ from the prior,
  - simulate $y^\star$,
  - accept if $\|y - y^\star\| < \varepsilon$.
- Tolerance ε controls:

$$\varepsilon \downarrow \Rightarrow \text{better approximation but lower acceptance.}$$

- Limits:

$$\varepsilon \to \infty \Rightarrow \text{recover prior,} \qquad \varepsilon \to 0 \Rightarrow \text{recover posterior (in ideal limit).}$$

- Curse of dimensionality motivates **Summary ABC**: match $\|S(y) - S(y^\star)\|$ instead.
- If S is **sufficient**, Summary ABC can be exact as $\varepsilon \to 0$; otherwise it introduces extra approximation.

Practical diagnostics for ABC

When running ABC in practice, always check:

- **Acceptance rate:** too low $\Rightarrow$ too few samples; too high $\Rightarrow$ likely close to the prior.
- **Sensitivity to ε:** do posterior summaries change a lot when you adjust ε slightly?
- **Prior vs posterior:** does the approximate posterior look meaningfully different from the prior?
- **Choice of distance and summaries:** are you matching features of the data that matter for θ?

Next steps (later chapters): more efficient ABC variants (e.g. sequential methods) and links to MCMC ideas.