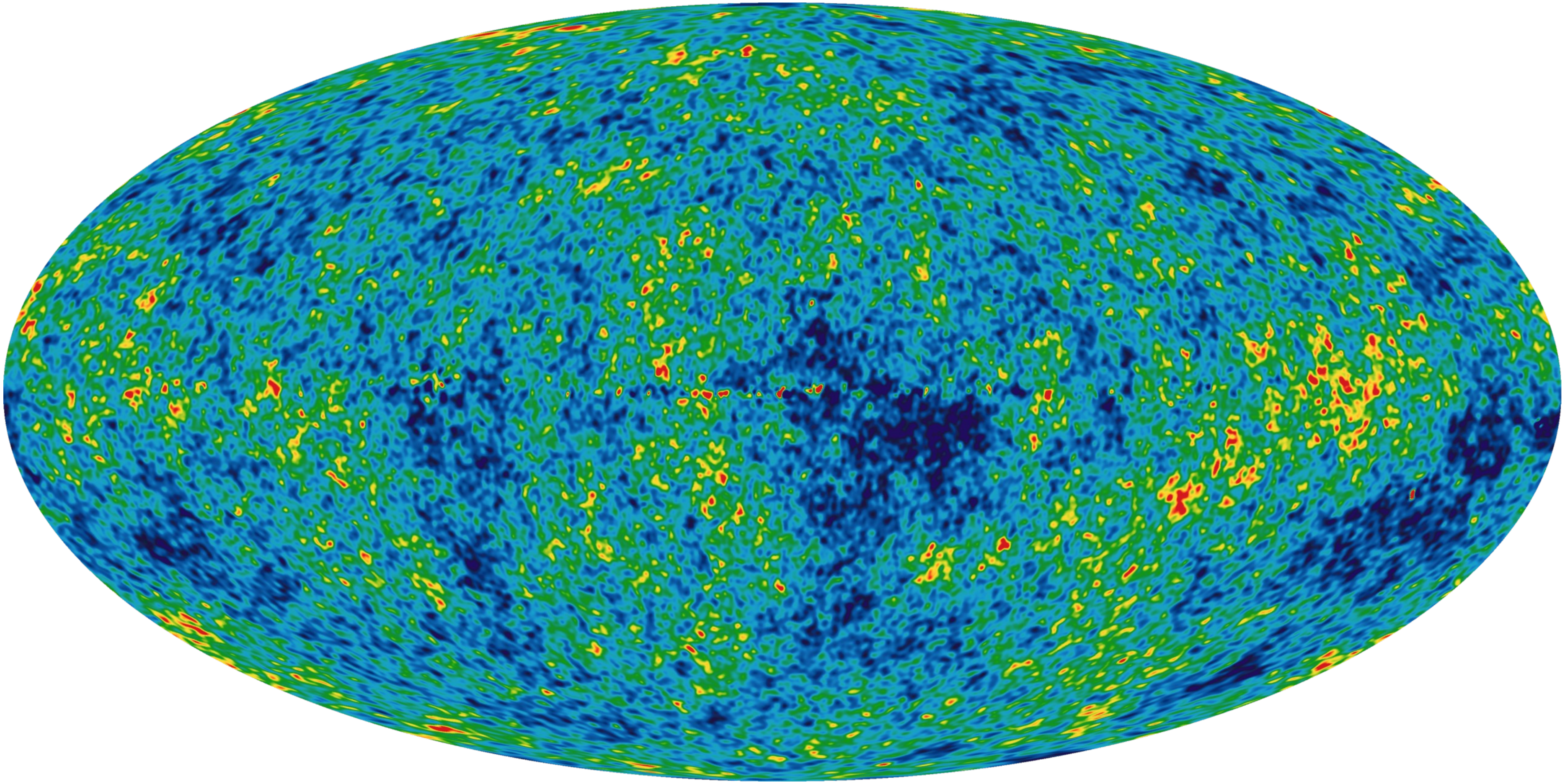
UNIVERSITY OF CAMBRIDGE

Perambulation through random numbers

Gareth Conduit

Theory of Condensed Matter group

Randomness at the start of the universe



Random numbers

Randomness is the apparent lack of pattern or predictability in events

Generation of random numbers

Using a sequence of random numbers to calculate deterministic quantities

1) Numerical integration

2) Heavy tailed distributions

3) Bootstrap sampling for machine learning

Generation of random numbers

Hardware random number generator e.g. thermal noise (classical), shot noise of electron flow (quantum), radioactive decay (quantum)

Time interval between successive events and compare to mean interval

Sycamore – Google quantum computer – high bandwidth quantum random number generator and tester

Software whitening

Underlying distribution function may be incorrect e.g. radioactive decay is exponential, can be corrected in software:

"Given an unfair coin that is heads *p* of the time and tails 1-*p* of the time, how do simulate a fair coin?"

Example problem

Underlying distribution function may be incorrect e.g. radioactive decay is exponential, can be corrected in software:

"Given an unfair coin that is heads *p* of the time and tails 1-*p* of the time, how do simulate a fair coin?"

HHHHTHHHHHHTTTHHHHTTTTHTHHHTTHHHTHHH

Idea: average over pairs of results

Underlying distribution function may be incorrect e.g. radioactive decay is exponential, can be corrected in software:

"Given an unfair coin that is heads p of the time and tails 1-p of the time, how do simulate a fair coin?"

```
HHHTHHHHHTTTHHHTTTTHTHHHTTHHHTHHH
```

HH	p^2	HT	$p(1-p)$
TT	$(1-p)^2$	TH	$(1-p)p$

Removed lowest order bias

Underlying distribution function may be incorrect e.g. radioactive decay is exponential, can be corrected in software:

"Given an unfair coin that is heads *p* of the time and tails 1-*p* of the time, how do simulate a fair coin?"

```
HHHHTHHHHHHTTTHHHHTTTTHTHHHTTHHHTHHH
    10    1  0   1   010  1 0  10
```

Results alternate, now equal number of 1s and 0s, but have introduced a higher order bias

Pair tosses and take first result from alternating pairs

Underlying distribution function may be incorrect e.g. radioactive decay is exponential, can be corrected in software:

"Given an unfair coin that is heads *p* of the time and tails 1-*p* of the time, how do simulate a fair coin?"

```
HHHHTHHHHHHTTTHHHHTTTTHTHHHTTHHHTHHH
    0     1           1   1 0   0
```

Protocol to handle bias (more 1s than 0s) by John von Neumann

Hardware generated random numbers speed limited

Underlying distribution function may be incorrect e.g. radioactive decay is exponential, can be corrected in software:

"Given an unfair coin that is heads *p* of the time and tails 1-*p* of the time, how do simulate a fair coin?"

```
HHHHTHHHHHHTTTHHHHTTTTHTHHHTTHHHTHHH
    0     1           1   1 0   0
```

Protocol to handle bias (more 1s than 0s) by John von Neumann

Software whitening lowers bandwidth of numbers

Software generated random numbers

Pseudorandom number generator is an algorithm giving an apparently random sequence of numbers

Faster than hardware generation and reproducible as it starts from a seed

Algorithms for software generated random numbers

Pseudorandom number generator is an algorithm giving an apparently random sequence of numbers

Faster than hardware generation and reproducible as it starts from a seed

Common algorithms include linear congruential, linear feedback shift registers, and Mersenne twister

Linear congruential generator

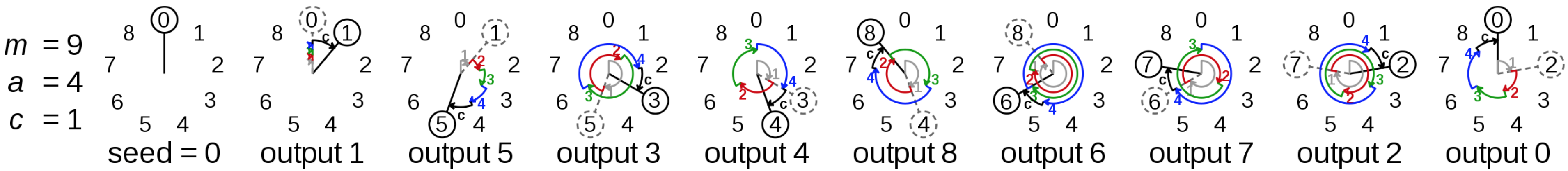
Relies on the non-invertability of modular mathematics

$X_{n+1} = (a\,X_n + c) \bmod m$ $r_n = X_n/(1 + m)$

Linear congruential generator

Relies on the non-invertability of modular mathematics

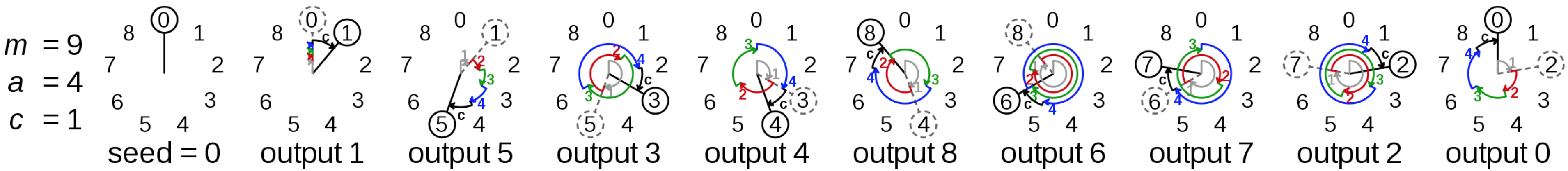
$$X_{n+1} = (a X_n + c) \bmod m \qquad r_n = X_n/(1 + m)$$



Linear congruential generator

Relies on the non-invertability of modular mathematics

$X_{n+1} = (a\, X_n + c) \bmod m \qquad r_n = X_n/(1 + m)$



glibc chooses m=2^{31}, a=1103515245, c=12345

Low quality random numbers, for example employing a single seed prohibits any random numbers in sequence from being identical until the sequence repeats

Applications of random number generators

Require a **random** number: gambling and cryptography

Calculating a **deterministic** results

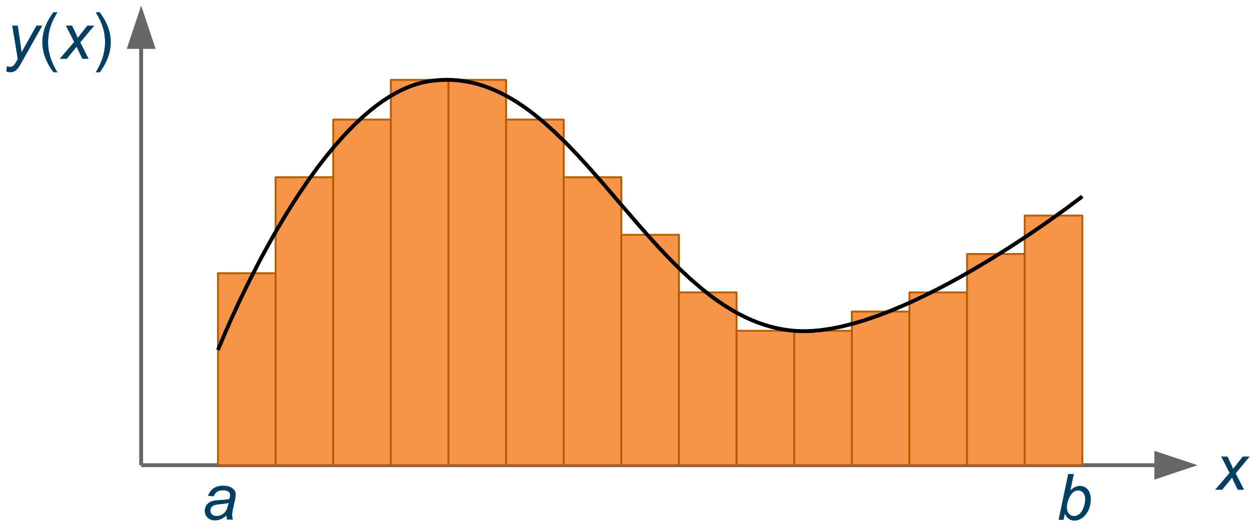
1) Numerical integration

2) Monte Carlo studies

3) Bootstrap sampling for machine learning

Numerical integration: midpoint rule

Fit rectangles to integrate under curve

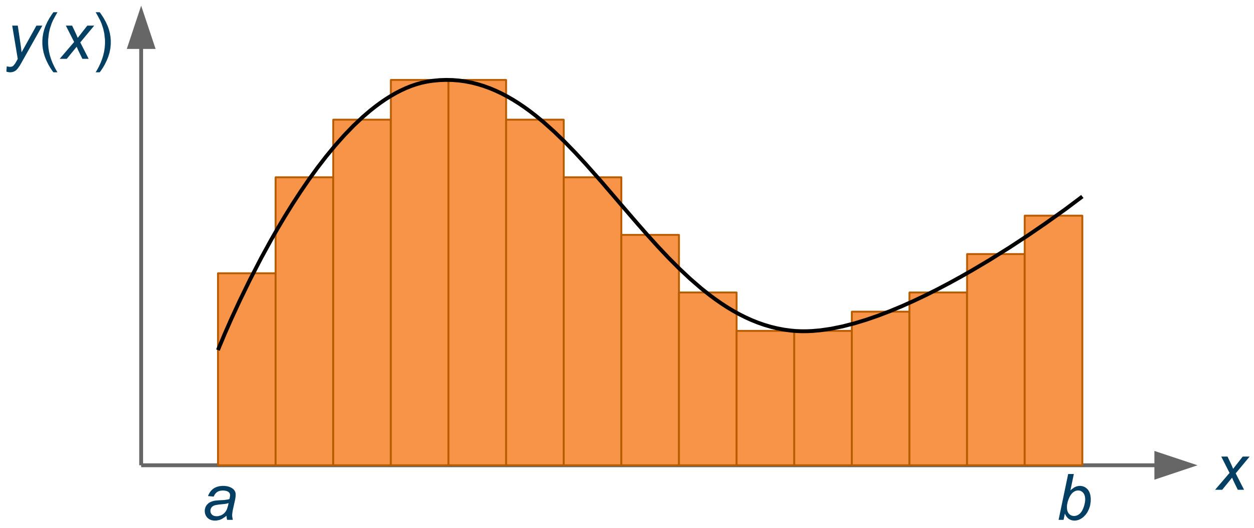


Area ≈ $\Delta x\,[\,y(a + \Delta x/2) + y(a + 3\Delta x/2) + \cdots + y(b - \Delta x/2)\,]$

Error ~ $1/N^2$

Numerical integration: Simpson's rule

Fit quadratics to integrate under curve



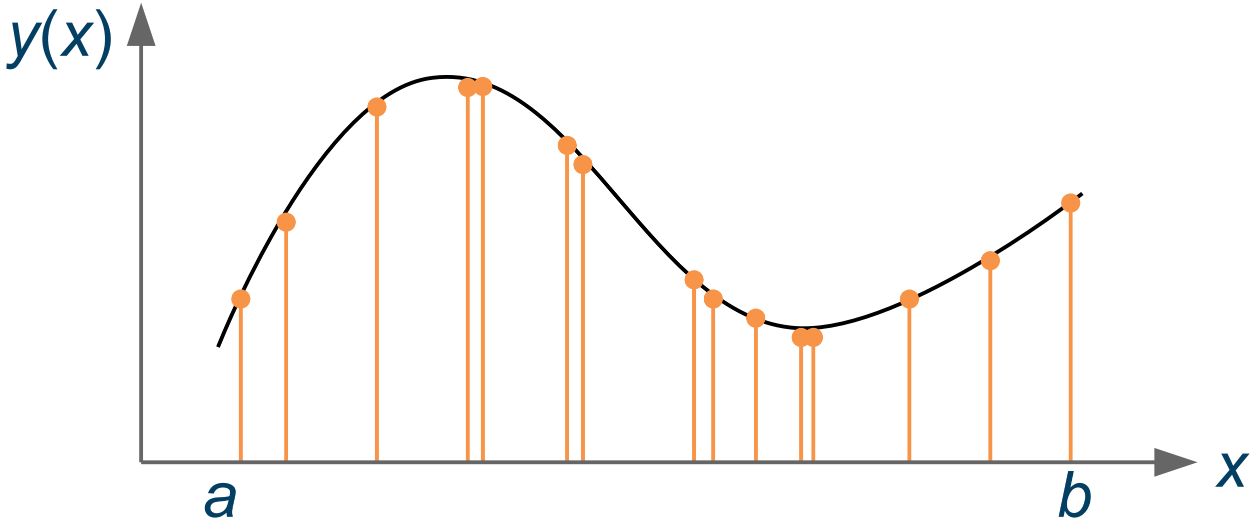
Area ≈ $\Delta x\,[\,y(a) + 4y(a + \Delta x) + 2y(a + 2\Delta x) + \cdots + y(b)\,]\,/\,3$

Error ~ $1/N^2$

Simpson's rule error ~ $1/N^4$

Numerical integration: random numbers

Sample the curve at N random points x_n

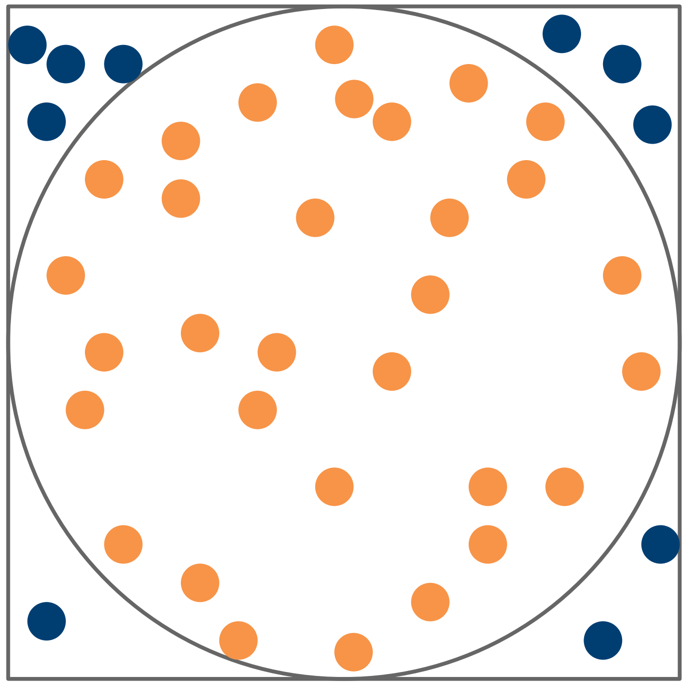


Area ≈ $(b - a) \sum_n y(x_n) / N$

Error ~ $1/N^{1/2}$

Two-dimensional integration

Estimate π by determining ratio of circle to square, function 1 inside circle, 0 outside of it



Simpson's rule error ~$1/N^{4/2} = 1/N^2$

Monte Carlo error ~ $1/N^{1/2}$

High dimensional integration

Integrate wavefunction for 100 electrons gives 300-dimensional integral

$$E = \frac{\int \langle \psi(\boldsymbol{R}) | \hat{H} | \psi(\boldsymbol{R}) \rangle \, d\boldsymbol{R}}{\int \langle \psi(\boldsymbol{R}) \, \psi(\boldsymbol{R}) \rangle \, d\boldsymbol{R}}$$

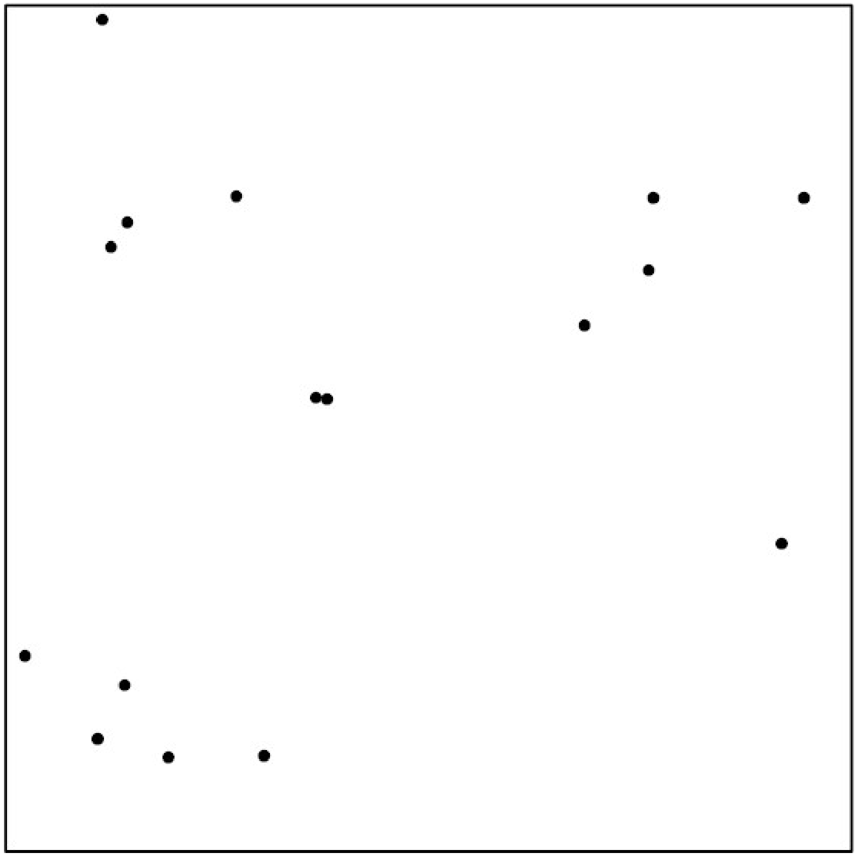
$$\boldsymbol{R} = \{\boldsymbol{r}_1, \boldsymbol{r}_2, \boldsymbol{r}_3, \cdots, \boldsymbol{r}_{100}\}$$

Simpson’s rule error ~$1/N^{4/300} = 1/N^{1/75}$

Monte Carlo error ~ $1/N^{1/2}$

Computationally efficient to do >8 dimensional integrals randomly, also beneficial for badly behaved integrands

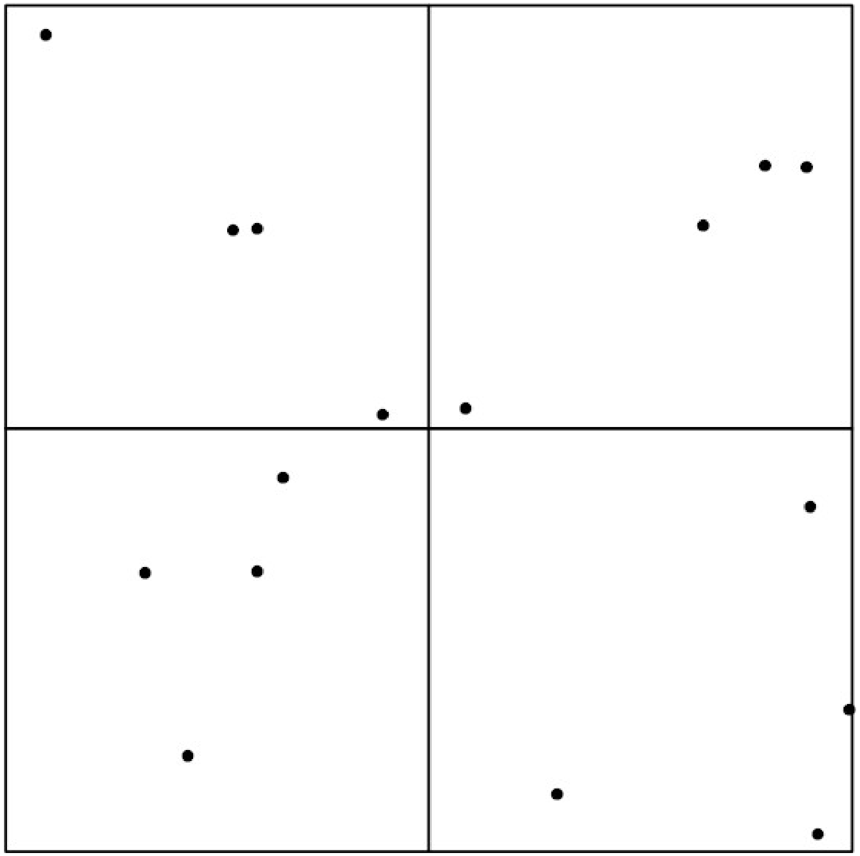
Clustering of pseudorandom numbers



Can we find another distribution of numbers that has the beneficial properties of randomness but avoids clustering?

Stratified sampling

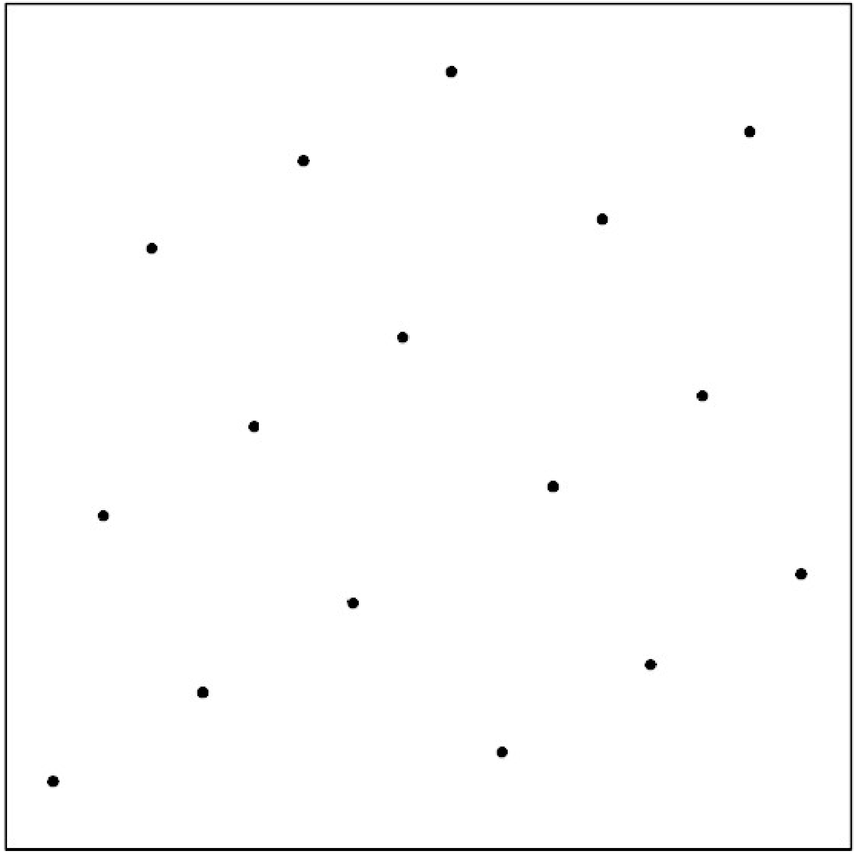
Partition space and sample each randomly



Leads to improved convergence still with $N^{-1/2}$ behavior

Low discrepancy sequence

Merge benefits of randomness yet approximately equidistributed



Algorithms include van der Corput (1D), Halton, and Sobol

Application to numerical integration

Performance of quasirandom numbers

Pseudorandom $\sim 1/N^{1/2}$

Quasirandom $\sim (\log N)^d/N$

Monte Carlo integration of a wavefunction

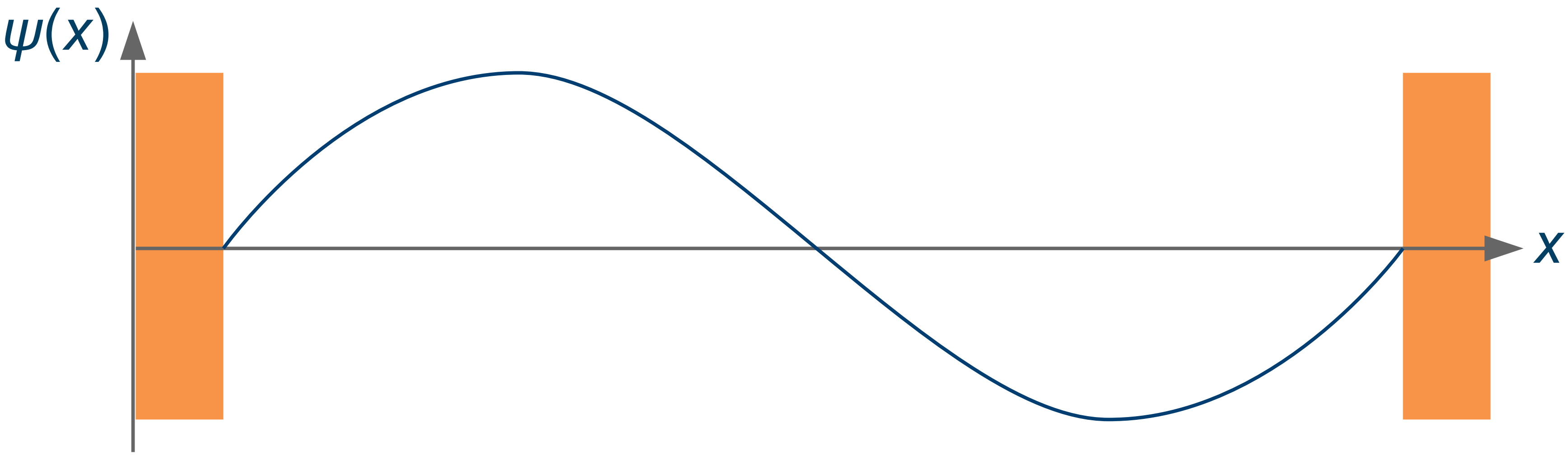
Evaluate a quantum expectation value

$$A = \frac{\int \langle \psi | \hat{A} | \psi \rangle d\,\boldsymbol{R}}{\int \langle \psi \; \psi \rangle d\,\boldsymbol{R}}$$

Wavefunction of a particle in a box

Evaluate a quantum expectation value

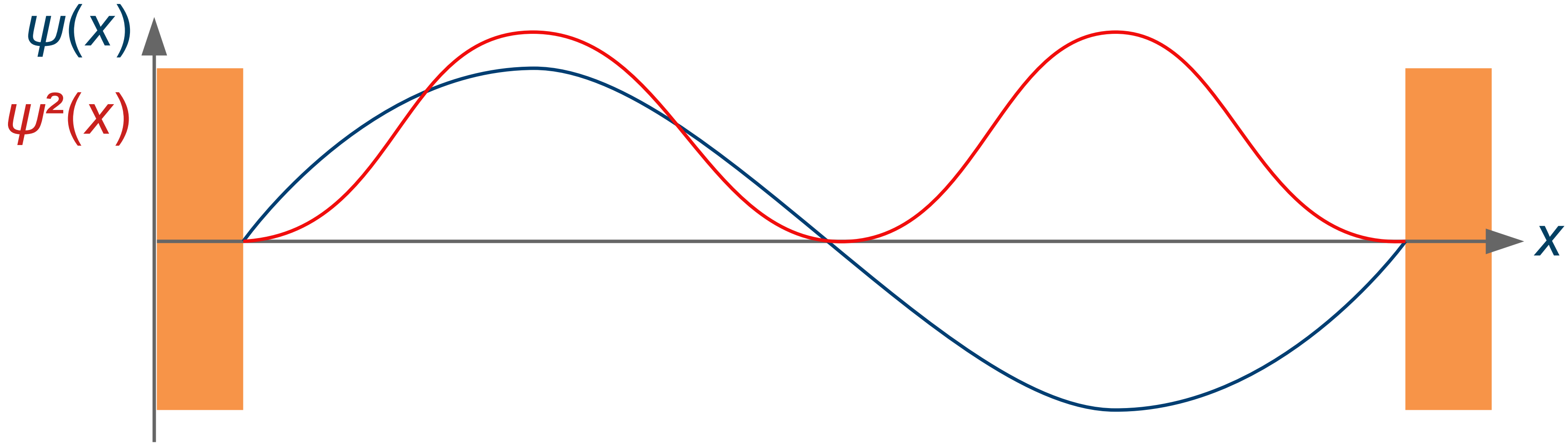
$$A = \frac{\int \langle \psi | \hat{A} | \psi \rangle d\boldsymbol{R}}{\int \langle \psi \; \psi \rangle d\boldsymbol{R}}$$



Integrand is not smooth

Evaluate a quantum expectation value

$$A = \frac{\int \langle \psi | \hat{A} | \psi \rangle d\,\boldsymbol{R}}{\int \langle \psi \; \psi \rangle d\,\boldsymbol{R}}$$

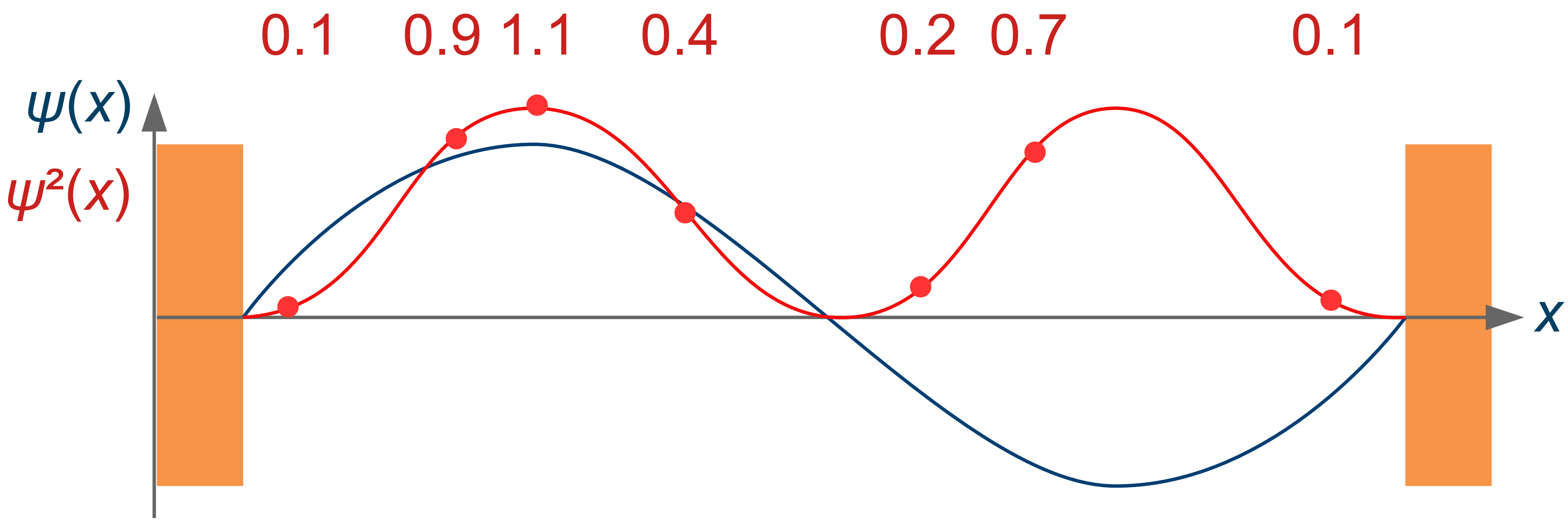


Integrand is not smooth

Evaluate a quantum expectation value

$$A = \frac{\int \langle \psi | \hat{A} | \psi \rangle d\,\boldsymbol{R}}{\int \langle \psi \ \psi \rangle d\,\boldsymbol{R}}$$

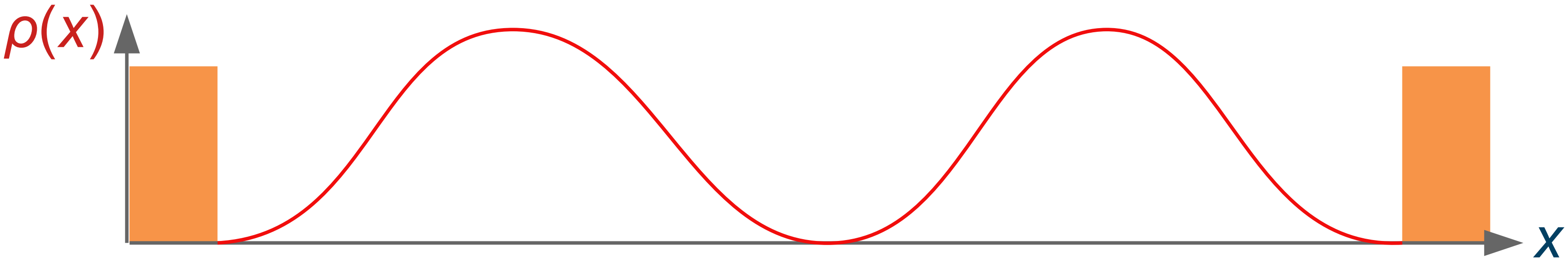
Mean = 0.50
Uncert = 0.16



Weight the sampling to focus on largest contribution

Sampling weighted by ψ^2 makes integrand more uniform

$$A=\frac{\int \langle \psi|\hat{A}|\psi\rangle d\mathbf{R}}{\int \langle \psi\ \psi\rangle d\mathbf{R}}=\int_{\psi^2} \frac{\langle \psi|\hat{A}|\psi\rangle}{\langle \psi\ \psi\rangle} d\mathbf{R}$$

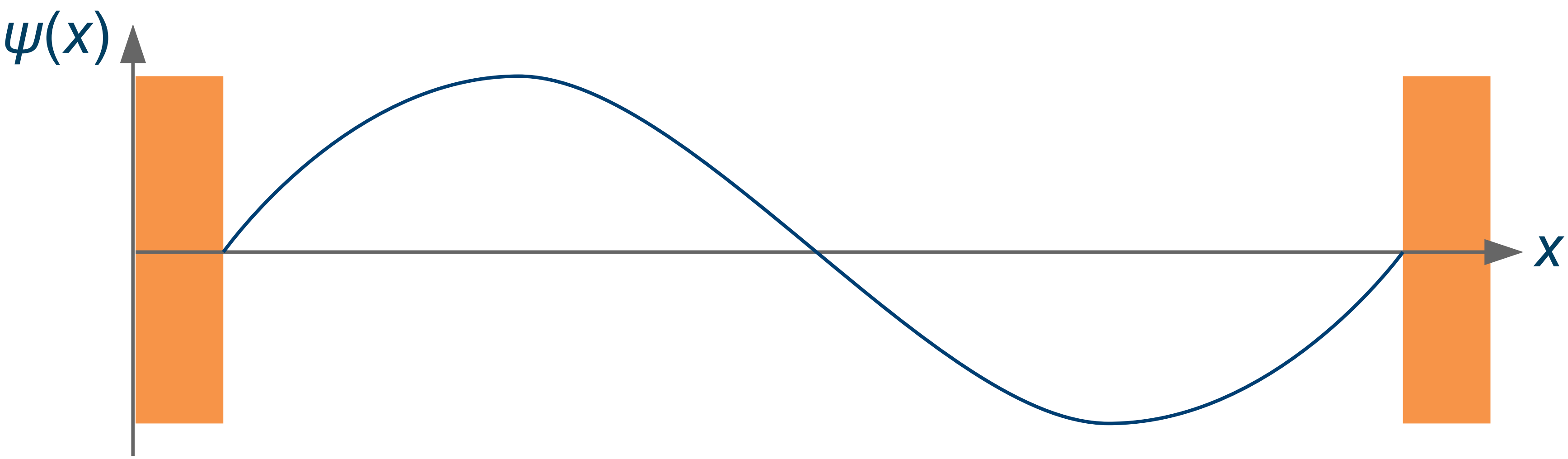


What happens at a node?

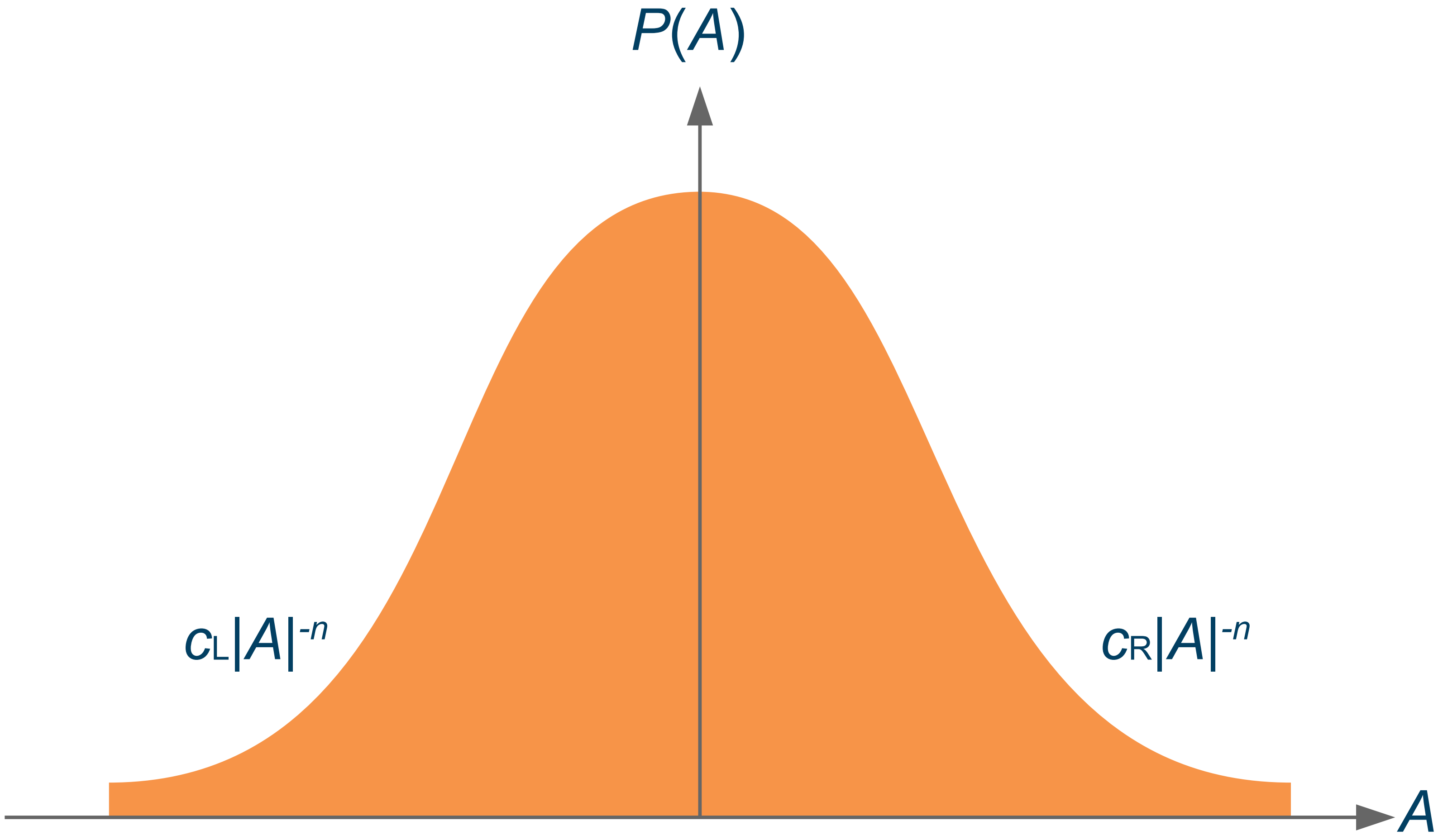
Evaluate a quantum expectation value

$$A = \frac{\int \langle \psi | \hat{A} | \psi \rangle d\boldsymbol{R}}{\int \langle \psi \ \psi \rangle d\boldsymbol{R}} = \int_{\psi^2} \frac{\langle \psi | \hat{A} | \psi \rangle}{\langle \psi \ \psi \rangle} d\boldsymbol{R}$$

What happens where wave function has a node?



Heavy tailed probability distribution

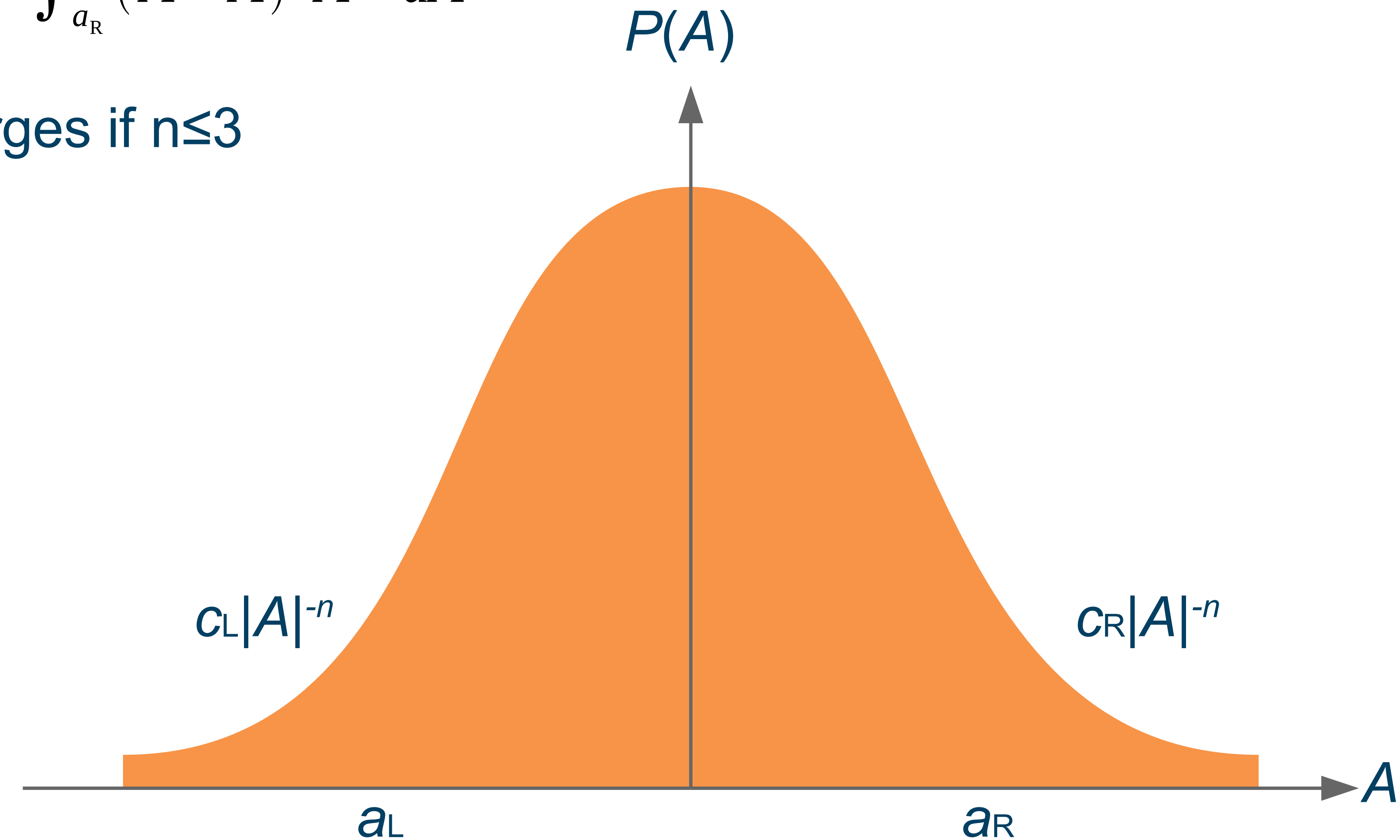


Uncertainty of sampling the heavy tail diverges

Uncertainty in expected value

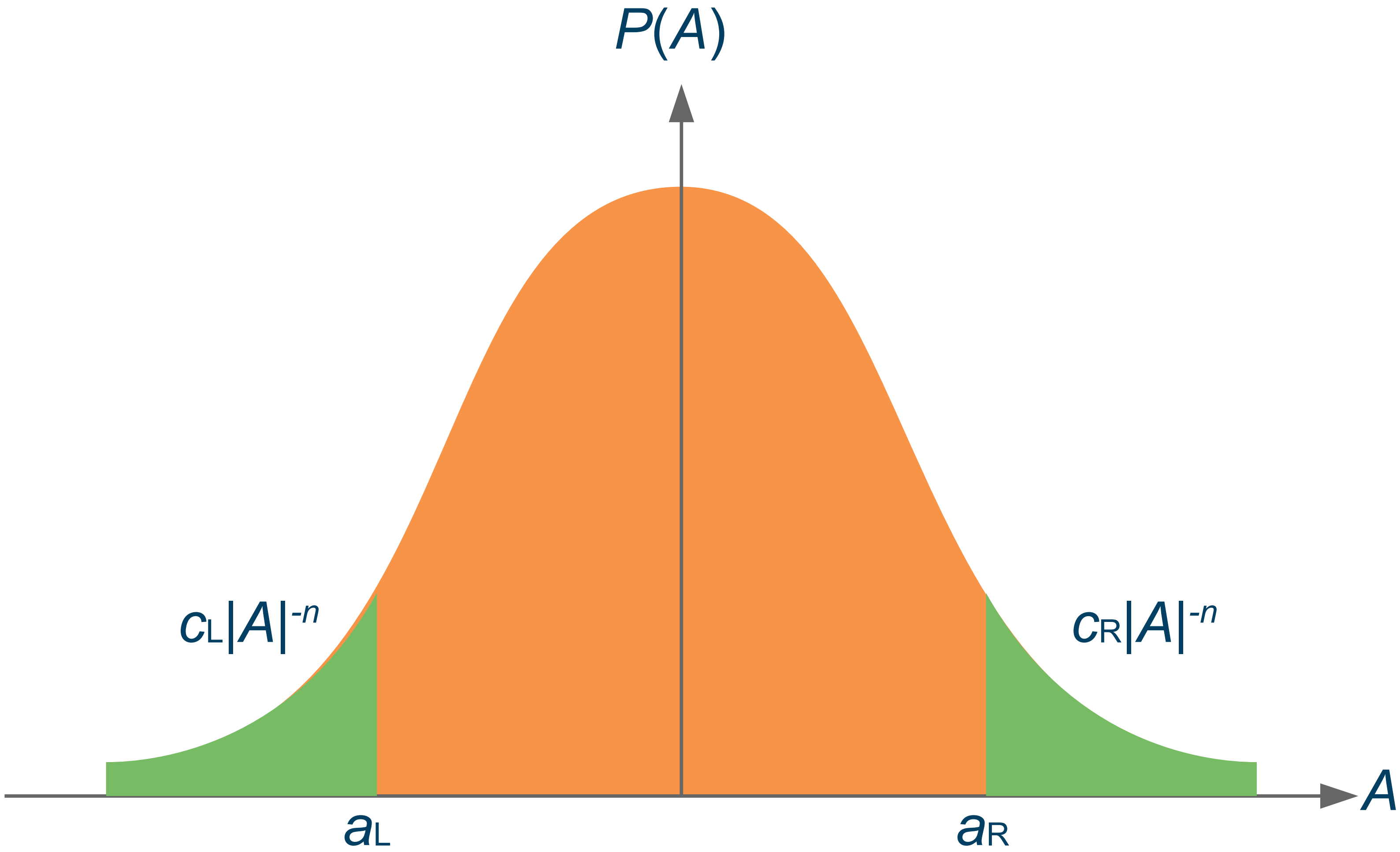
$$\sigma_A \approx \int_{a_R}^{\infty} \left(A - \bar{A}\right)^2 A^{-n} dA$$

Diverges if n≤3



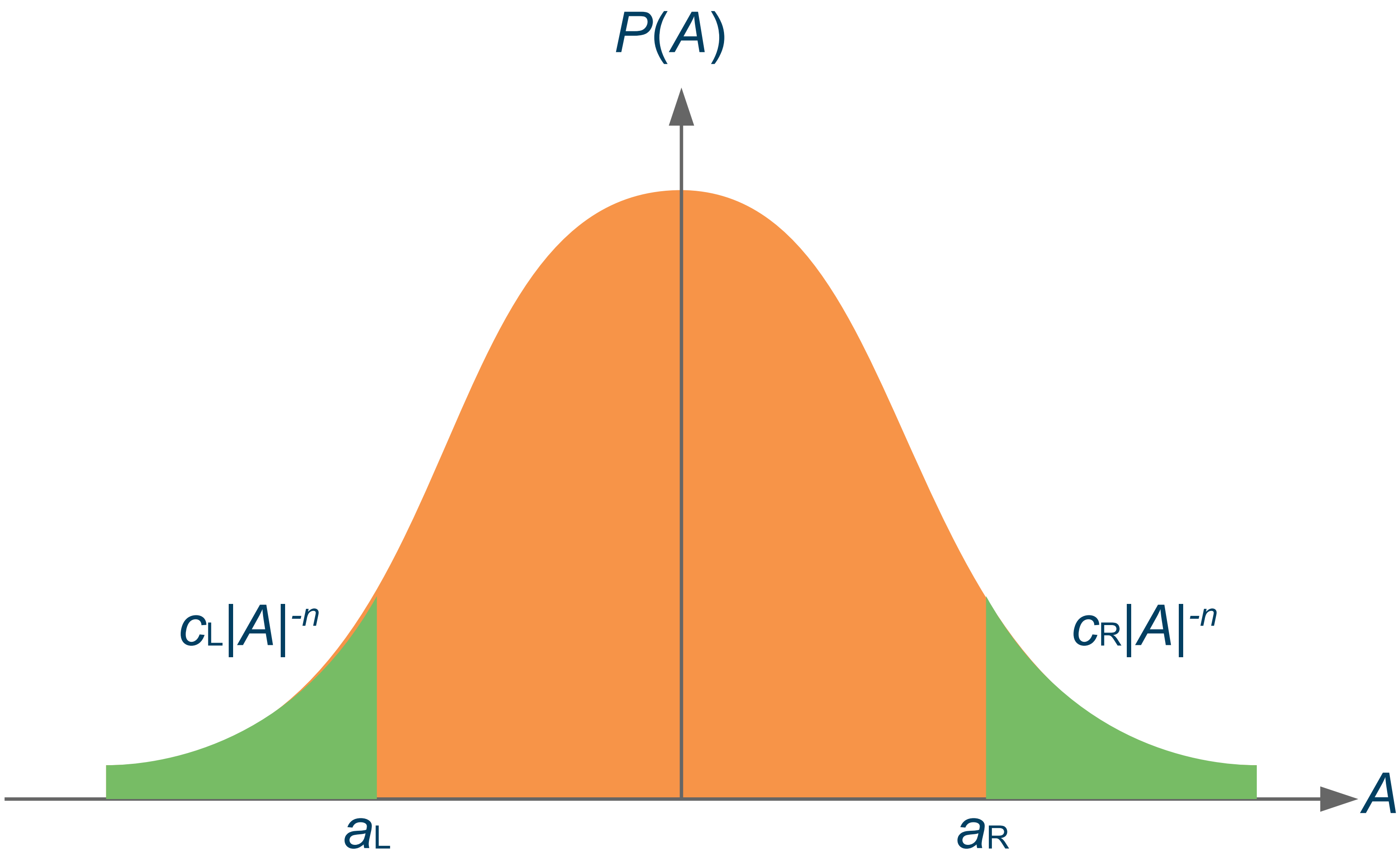
Fit and replace points in the tail

Uncertainty in c_L, c_R is well-defined

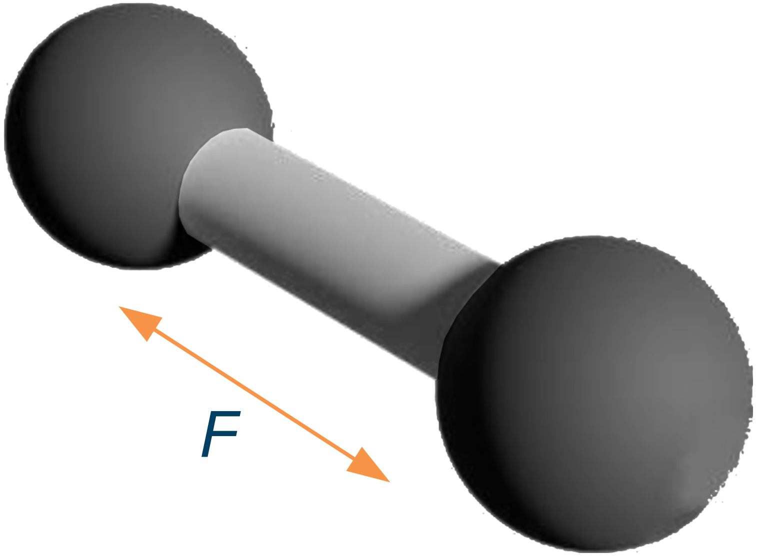


Fit and replace points in the tail

$$\langle A \rangle \approx \frac{1}{M} \sum_{a_L < A < a_R} A + c_L \int_{-\infty}^{a_L} A |A|^{-n} dA + c_R \int_{a_R}^{\infty} A |A|^{-n} dA$$

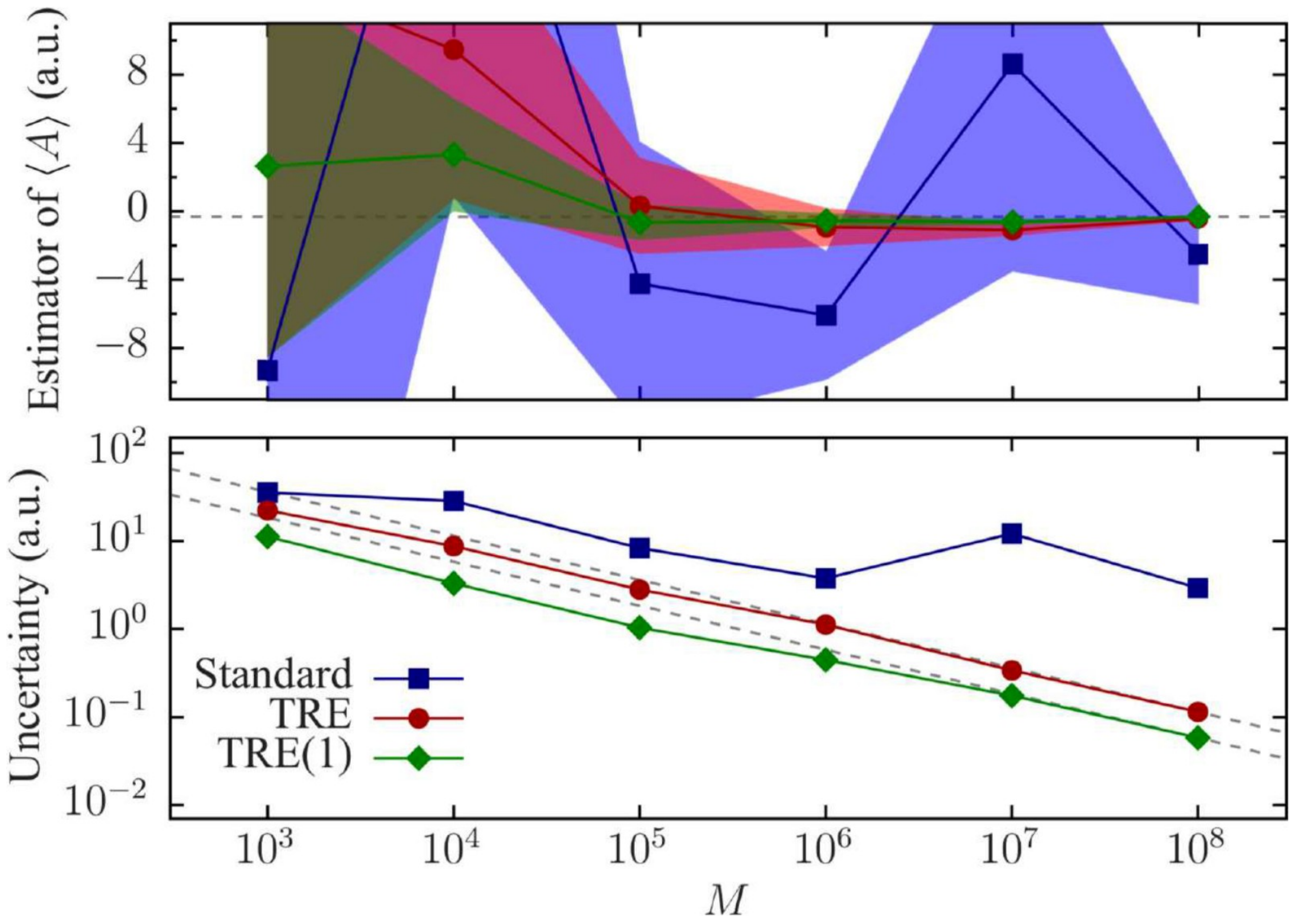


Interatomic force in a carbon molecule



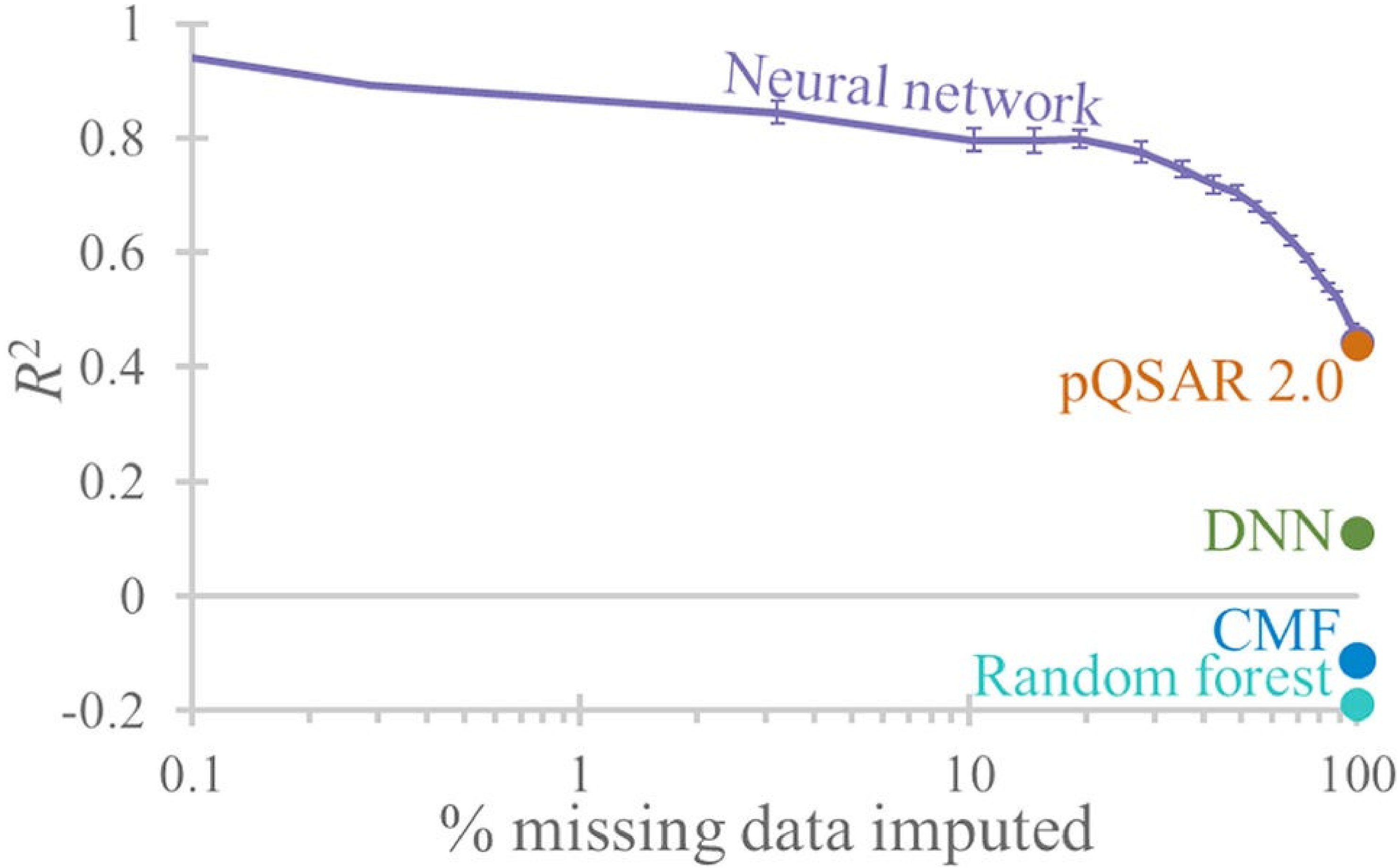
Tail $|F|^{-5/2}$ so well defined expected force but divergent uncertainty

Accelerate calculation of interatomic force



P. López Ríos & G.J. Conduit, Phys. Rev. E **99**, 063312 (2019)

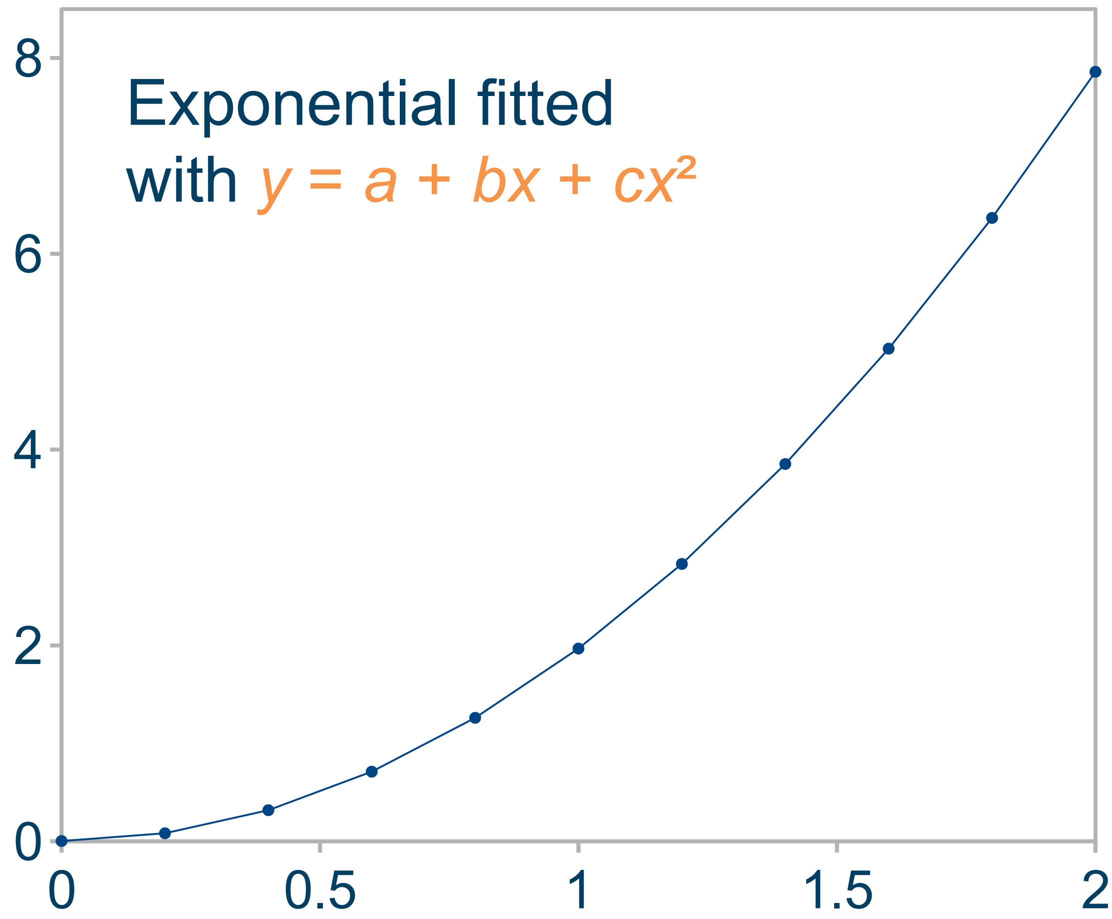
Estimating uncertainties in machine learning



Estimating uncertainties in machine learning

Orig data

x	y
0	1.00
0.2	1.22
0.4	1.49
0.6	1.82
0.8	2.23
1	2.72
1.2	3.32
1.4	4.06
1.6	4.95
1.8	6.05
2	7.39



Bootstrap sample randomly with replacement

Orig data

x	y
0	1.00
0.2	1.22
0.4	1.49
0.6	1.82
0.8	2.23
1	2.72
1.2	3.32
1.4	4.06
1.6	4.95
1.8	6.05
2	7.39

Model 1

x	y
0	1.00

Select the second entry

Orig data

x	y
0	1.00
0.2	1.22
0.4	1.49
0.6	1.82
0.8	2.23
1	2.72
1.2	3.32
1.4	4.06
1.6	4.95
1.8	6.05
2	7.39

Model 1

x	y
0	1.00
0.6	1.82

Select the third entry

Orig data

x	y
0	1.00
0.2	1.22
0.4	1.49
0.6	1.82
0.8	2.23
1	2.72
1.2	3.32
1.4	4.06
1.6	4.95
1.8	6.05
2	7.39

Model 1

x	y
0	1.00
0.6	1.82
1.2	3.32

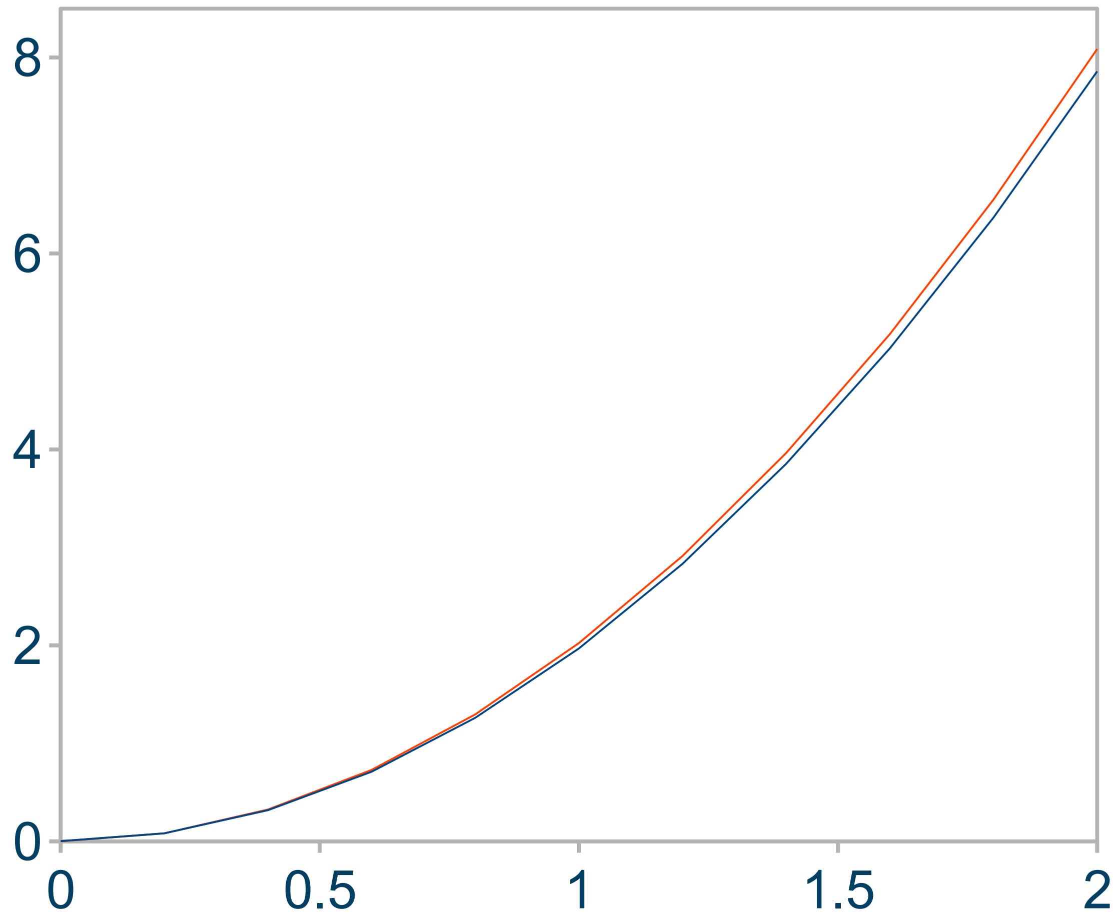
Entire sample for first model

Orig data		Model 1	
x	**y**	**x**	**y**
0	1.00	0	1.00
0.2	1.22	0.6	1.82
0.4	1.49	1.2	3.32
0.6	1.82	1.6	4.95
0.8	2.23	0.4	1.49
1	2.72	1.4	4.06
1.2	3.32	1.6	4.95
1.4	4.06	0.4	1.49
1.6	4.95	0	1.00
1.8	6.05	1.6	4.95
2	7.39	1.4	4.06

First bootstrap model

Model 1

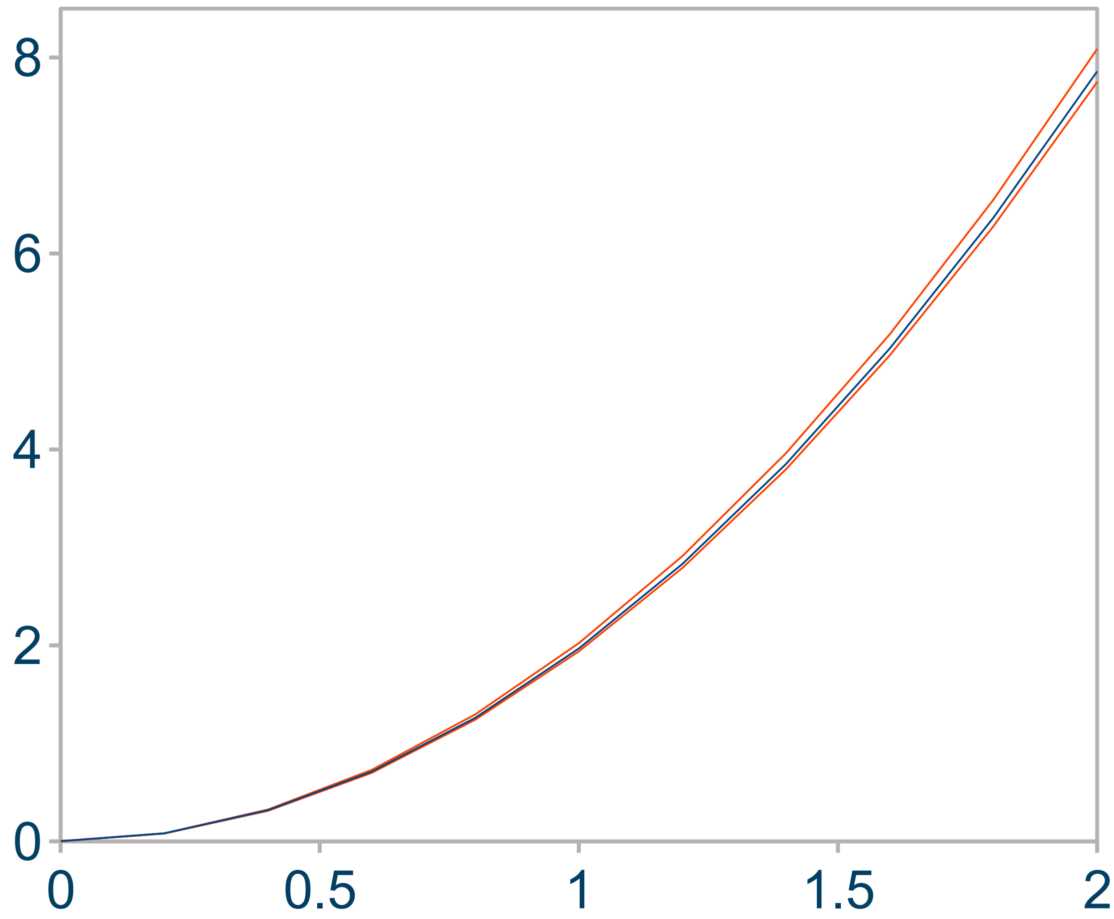
x	y
0	1.00
0.6	1.82
1.2	3.32
1.6	4.95
0.4	1.49
1.4	4.06
1.6	4.95
0.4	1.49
0	1.00
1.6	4.95
1.4	4.06



Second bootstrap model

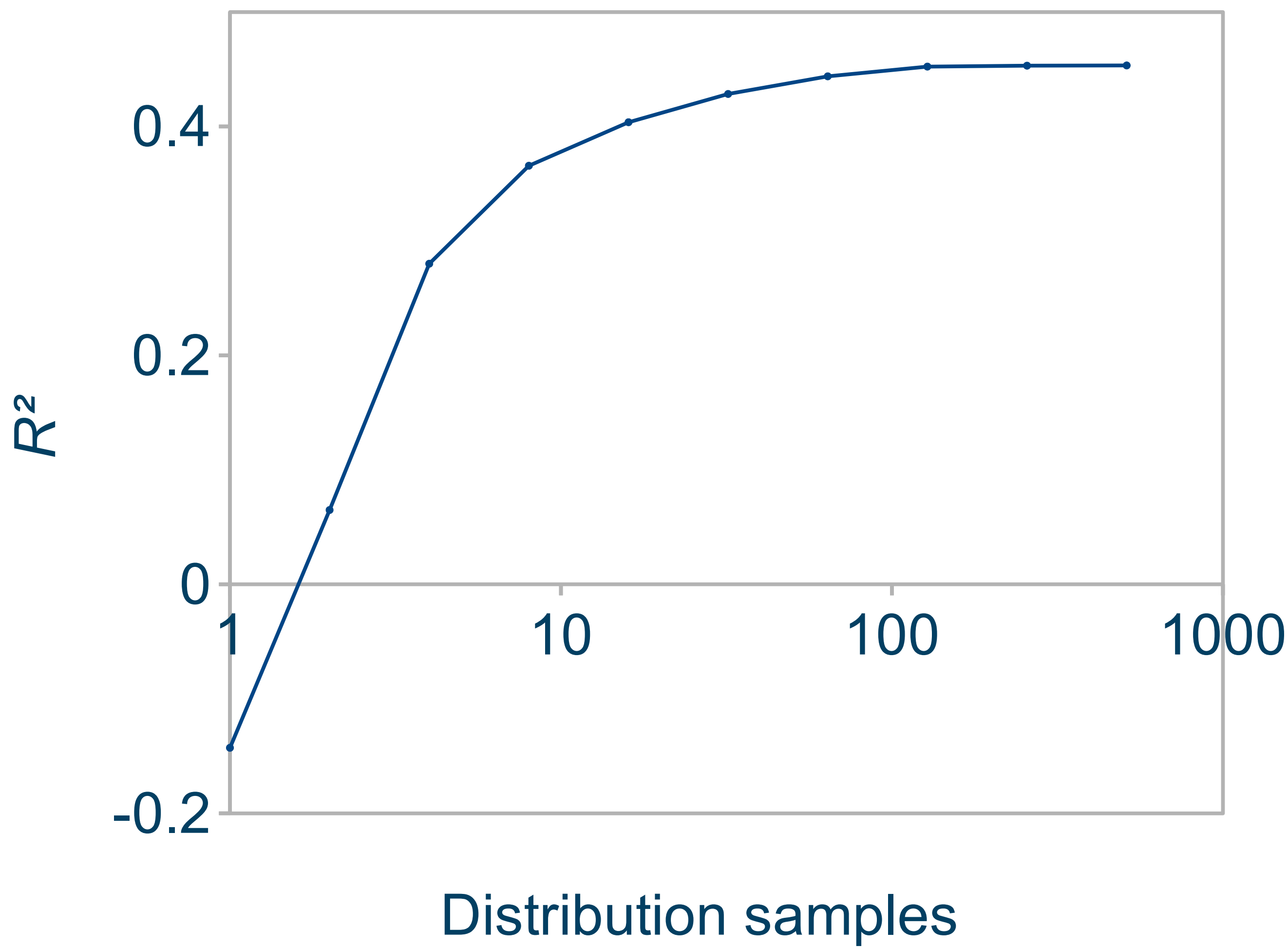
Model 2

x	y
1.6	4.95
1.6	4.95
0	1.00
1.4	4.06
2	7.39
1.6	4.95
1.4	4.06
1.6	4.95
1.8	6.05
0.8	2.23
0.8	2.23



Problems with bootstrap

Slow convergence of mean prediction and counterintuitive behavior with one distribution sample



Constrain distribution: sample without replacement

Orig data1		Orig data 2		Model 1		Model 2	
x	**y**	**x**	**y**	**x**	**y**	**x**	**y**
0	1.00	0	1.00				
0.2	1.22	0.2	1.22				
0.4	1.49	0.4	1.49				
0.6	1.82	0.6	1.82				
0.8	2.23	0.8	2.23				
1	2.72	1	2.72				
1.2	3.32	1.2	3.32				
1.4	4.06	1.4	4.06				
1.6	4.95	1.6	4.95				
1.8	6.05	1.8	6.05				
2	7.39	2	7.39				

Z. Mashreghi, D. Haziza, C. Léger
Statistics Surveys, 10, 1 (2016)

Constrain distribution: first entry

Orig data1		Orig data 2	
x	y	x	y
0	1.00	0	1.00
0.2	1.22	0.2	1.22
0.4	1.49	0.4	1.49
0.6	1.82	0.6	1.82
0.8	2.23	0.8	2.23
1	2.72	1	2.72
1.2	3.32	1.2	3.32
1.4	4.06	1.4	4.06
1.6	4.95	1.6	4.95
1.8	6.05	1.8	6.05
2	7.39	2	7.39

Model 1		Model 2	
x	y	x	y
1.6	4.95		

Constrain distribution: second entry

Orig data1		Orig data 2	
x	**y**	**x**	**y**
0	1.00	0	1.00
0.2	1.22	0.2	1.22
0.4	1.49	0.4	1.49
0.6	1.82	0.6	1.82
0.8	2.23	0.8	2.23
1	2.72	1	2.72
1.2	3.32	1.2	3.32
1.4	4.06	1.4	4.06
		1.6	4.95
1.8	6.05	1.8	6.05
2	7.39	2	7.39

Model 1		Model 2	
x	**y**	**x**	**y**
1.6	4.95		
0.4	1.49		

Constrain distribution: third entry

Orig data1		Orig data 2	
x	**y**	**x**	**y**
0	1.00	0	1.00
0.2	1.22	0.2	1.22
0.4	1.49		
0.6	1.82	0.6	1.82
0.8	2.23	0.8	2.23
1	2.72	1	2.72
1.2	3.32	1.2	3.32
1.4	4.06	1.4	4.06
		1.6	4.95
1.8	6.05	1.8	6.05
2	7.39	2	7.39

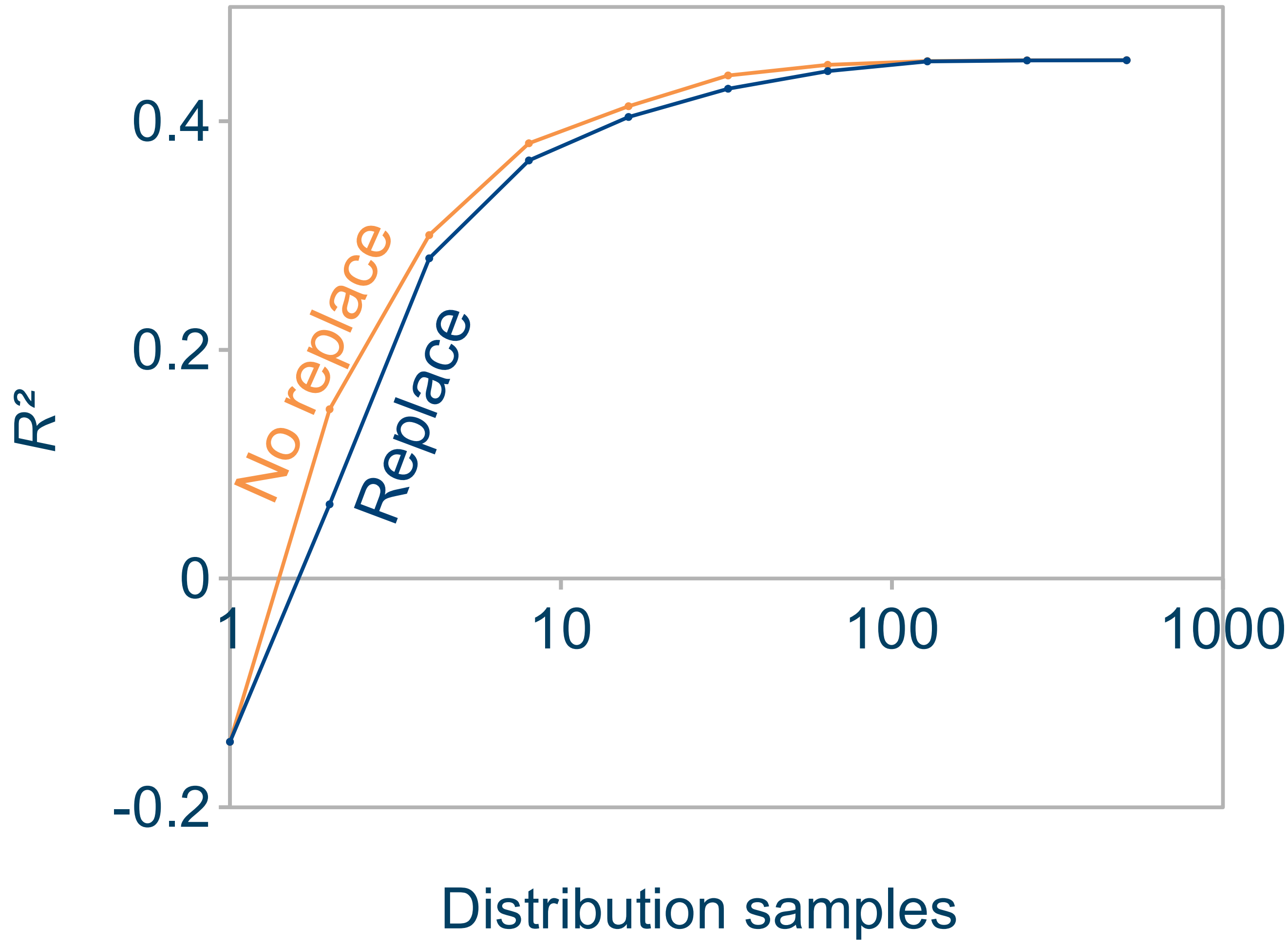
Model 1		Model 2	
x	**y**	**x**	**y**
1.6	4.95		
0.4	1.49		
2	7.39		

Constrain distribution: data for two models

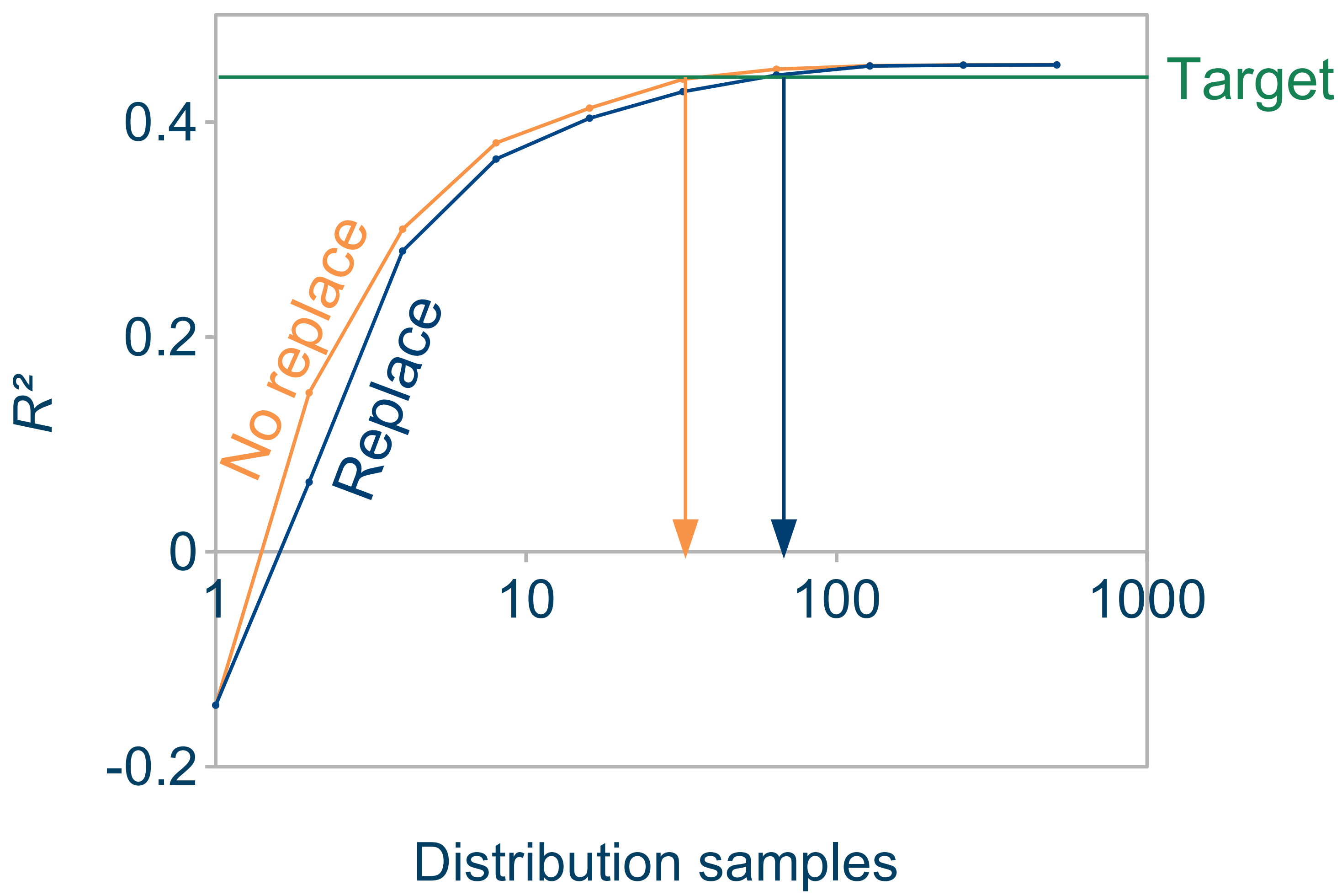
Orig data1		Orig data 2	
x	y	x	y

Model 1		Model 2	
x	y	x	y
1.6	4.95	1.6	4.95
0.4	1.49	0.2	1.22
2	7.39	1	2.72
1.4	4.06	0.8	2.23
0.4	1.49	2	7.39
0.2	1.22	1.2	3.32
1.4	4.06	1.8	6.05
1.8	6.05	0.8	2.23
1	2.72	0.6	1.82
0	1.00	0.6	1.82
1.2	3.32	0	1.00

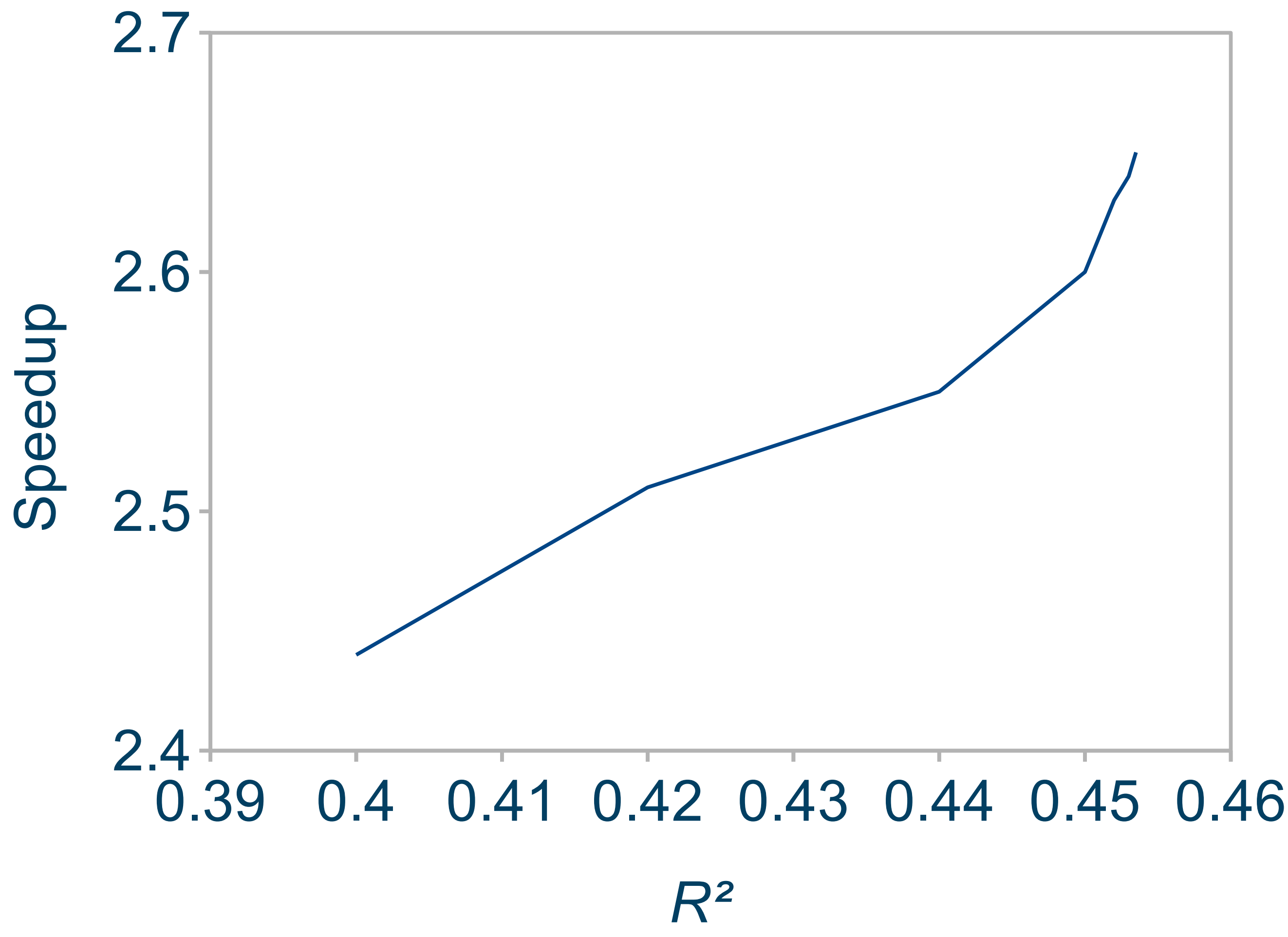
Compare accuracy for two bootstrap strategies



Constrained probability distribution



Speedup offered by constraining sampling



Correct distribution if either one or infinite number of models

Summary

Random numbers frequently used to calculate deterministic quantities

Low discrepancy random numbers increase efficiency of sampling over random numbers

Analytical knowledge of the heavy tail permits calculation of expectation values

Constrained bootstrap sampling increases efficiency by over x2