Statistical Physics 2019/20
MATH327
Kurt Langfeld & David Schaich

# Virtual Computer Lab

Friday 24 April

# Computer project Part A feedback

Generally high marks on Exercises 1–2,  Exercise 3 more challenging

**Philosophy:** Exercises 1–3 are warm-ups for Part B
Exact analytic solutions possible, allowing checks

# Computer project Part A feedback

Generally high marks on Exercises 1–2,    Exercise 3 more challenging

**Philosophy:** Exercises 1–3 are warm-ups for Part B
Exact analytic solutions possible, allowing checks

Shortcuts also possible, which won't work for Part B
$\longrightarrow$ If having trouble with Part B, try redoing Part A without shortcuts

# Computer project Part A feedback

Generally high marks on Exercises 1–2, Exercise 3 more challenging

**Philosophy:** Exercises 1–3 are warm-ups for Part B
Exact analytic solutions possible, allowing checks

Shortcuts also possible, which won't work for Part B
$\longrightarrow$ If having trouble with Part B, try redoing Part A without shortcuts

**Logistics:** Submitting code in separate (.m / .py / etc.) file
may help speed up marking

# Speeding up Exercise 3

Some reported trouble running all $N = 1, \cdots, 500$ in Exercise 3
(may have same issue in Exercise 5)

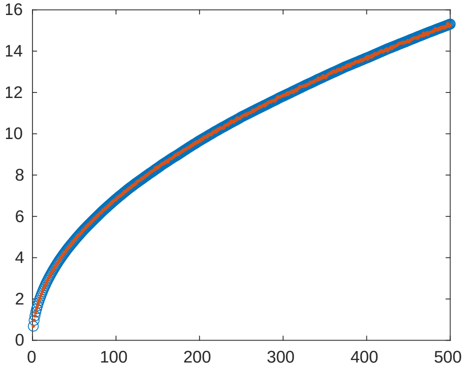
Trick: $N = 500$ case includes all $1 \leq N \leq 499$ for free
$\longrightarrow$ roughly 250 **times** faster!

# Speeding up Exercise 3

Trick: $N = 500$ case includes all $1 \leq N \leq 499$ for free

$\longrightarrow$ roughly 250 **times** faster!

Potential concern: **Correlations** if $N$-step result is based on $(N-1)$-step result




Blue circles from correlated computation

Every red point computed independently

$\longrightarrow$ zooming in reveals more fluctuations

Fit *residuals* roughly double

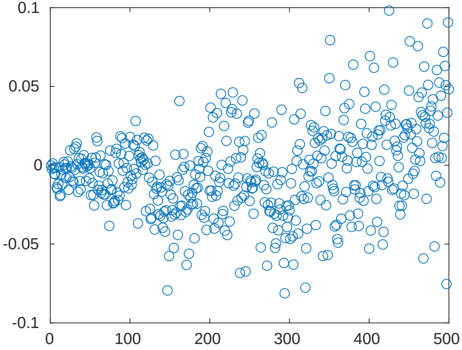
$\nwarrow$(difference between fit and points)

# Speeding up Exercise 3

Trick: $N = 500$ case includes all $1 \leq N \leq 499$ for free

$\longrightarrow$ roughly 250 **times** faster!

Potential concern: **Correlations** if $N$-step result is based on $(N-1)$-step result




With $n = 100,000$,

statistical fluctuations neglected by trick

no more than ~1% effect

# An easier (if smaller) speed-up

Computing a “**running sum**” can also speed things up

Example: These produce the same result (0.3334)

```
rng(314156);
n = 10^7;
x = 0.0;
for i = 1:n
  x = x + rand^2;
end
ave = x / n
```

```
rng(314156);
n = 10^7;
x = zeros(1, n + 1);
for i = 1:n
  x(i + 1) = x(i) + rand^2;
end
ave = x(n + 1) / n
```

The code on the left uses millions of times less data storage

# One limited shortcut

Many avoided doing the fit to $\ell_2 = D\sqrt{N}$
by defining $D \equiv \ell_2/\sqrt{N}$ and averaging estimates for each $N = 1, \cdots, 500$

This won't suffice for $\ell_\theta = D\, N^\alpha$ in Part B, where both $D$ and $\alpha$ unknown

# One limited shortcut

Many avoided doing the fit to $\ell_2 = D\sqrt{N}$
by defining $D \equiv \ell_2/\sqrt{N}$ and averaging estimates for each $N = 1, \cdots, 500$

This won't suffice for $\ell_\theta = D\,N^\alpha$ in Part B, where both $D$ and $\alpha$ unknown

Let's review how to do bona fide fits
by confirming that the law of diffusion is dimension-independent