ANSWERS

U4.2 Averaging

Noise and filtering · University · about 25 min

What this lesson is about

The square root of n, why it stops paying, and what it costs in time.

- The square root of n
- The cost is time
- Independence, again

Questions

6 marks in all

1. A depth sensor has sigma = 4 cm. How many independent readings must be averaged to get an estimate with sigma = 1 cm? [1 mark]

> **Answer:** 16. sigma / sqrt(n) = 1 needs sqrt(n) = 4, so n = 16.

2. What does this print? [1 mark]

```
SIGMA = 6.0
for n in (1, 4, 16, 64):
    print(n, SIGMA / n ** 0.5, round(n * 0.1, 1))
```

> **Answer:**
>
> ```
> 1 6.0 0.1
> 4 3.0 0.4
> 16 1.5 1.6
> 64 0.75 6.4
> ```
>
> Each fourfold increase in n halves the noise, 6 to 3 to 1.5 to 0.75 cm, while the time taken at 0.1 s per reading grows fourfold. That is the diminishing return.

3. With the noise set to sigma = 8 cm and a new reading every 0.1 s, how many seconds of standing still does an estimate good to 1 cm take? [1 mark]

> **Answer:** 6.4 (accept within 0.05). 8 / sqrt(n) = 1 needs n = 64 readings, and 64 x 0.1 s = 6.4 s.

4. Averaging 100 readings has already been done. What does going to 400 readings buy? [1 mark]

- ☐ **A** The noise halves again, at four times the time
- ☐ **B** The noise falls to a quarter
- ☐ **C** The noise halves and the bias halves
- ☐ **D** Nothing, because 100 readings is the limit

> **Answer:** A. sqrt(400 / 100) = 2, so the noise halves while the time goes from 10 s to 40 s. Bias is untouched however many readings are taken.

5. Averaging 16 readings should divide the noise by 4. In which of these cases will it not? [1 mark]

Tick every answer that is true.

- ☐ **A** The 16 readings are taken in a tight loop faster than the sensor updates
- ☐ **B** A slow draught nudges every reading in the window the same way
- ☐ **C** The 16 readings are taken 0.1 s apart with the robot still
- ☐ **D** The sensor's sigma is 8 cm instead of 4 cm

> **Answer:** A, B. Root n only holds for independent readings. Repeated samples and noise that is correlated across the window both break independence; a larger sigma is still divided by 4.

6. A robot drives at 20 cm/s while averaging 60 readings taken 0.1 s apart. How many centimetres does it travel during the average? [1 mark]

> **Answer:** 120. 60 x 0.1 s = 6 s, and 6 s x 20 cm/s = 120 cm. The average describes over a metre of driving, which is why the rule is average while still, filter while moving.

The task: average it down

The sensor has about 4 cm of noise on it. Standing still, produce an estimate of the gap to the wall good to about a centimetre, and print `readings:` (how many you took) and `distance:` (your answer). Good to a centimetre at one sigma still leaves a third of answers further out than that, so take enough readings that a centimetre is two sigma.

```
from bugbot import *
connect()

# how many readings put two sigma at 1 cm, with sigma = 4?
```

The hint students can ask for: Aim for 1 cm at two sigma, 0.5 cm at one: work out n, then take that many, a tenth of a second apart.

A solution

```
from bugbot import *
connect()

n = 80
readings = []
for i in range(n):
    readings.append(distance())
    wait(0.1)
print("readings:", n)
print("distance:", round(sum(readings) / n, 2))
```

Any program that meets the task's checks is marked correct in the simulator; this is one way, not the only way.