WORKSHEET

U7.5 Resampling

Localisation · University · about 30 min

BugBotLab

Name _____ Class _____ Date _____

What this lesson is about

Keeping the good guesses without losing the diversity that lets the filter recover.

- Low variance resampling
- Particle deprivation
- Watching it work

Questions

6 marks in all

1. What does this program print? [1 mark]

```
particles = ["A", "B", "C", "D"]
weights = [0.05, 0.5, 0.3, 0.15]
N = 4
step = 1.0 / N
r = 0.12
c = weights[0]
i = 0
fresh = []
for m in range(N):
    u = r + m * step
    while u > c and i < N - 1:
        i += 1
        c += weights[i]
    fresh.append(particles[i])
print(" ".join(fresh))
```

2. With low variance resampling and N = 500, a particle has weight 0.013. How many copies does it get? [1 mark]

- ☐ **A** 6 or 7
- ☐ **B** Exactly 7
- ☐ **C** Any number from 0 to 500, depending on luck
- ☐ **D** 13

3. After many rounds of resampling without jitter, the filter is confidently and permanently wrong. What has happened? [1 mark]

- ☐ **A** Particle deprivation: every particle is an exact copy of one ancestor, so nothing is left elsewhere to rescue it
- ☐ **B** The weights have underflowed to zero and the floor has taken over
- ☐ **C** The motion noise was too large, so the cloud spread off the mat
- ☐ **D** The effective sample size has grown above N

4. Which of these defend against particle deprivation? [1 mark]

Tick every answer that is true.

- ☐ **A** Add a small amount of jitter to each copy
- ☐ **B** Resample only when neff falls below N/2
- ☐ **C** Inject a few particles scattered over the whole map each tick
- ☐ **D** Resample on every tick whatever neff is
- ☐ **E** Use a smaller sigma in the measurement model

5. Put the steps of low variance resampling in order. [1 mark]

Number the lines 1 to 5 to put them in the right order.

___ `Set c to the first weight and i to 0`

___ `For each m, set the pointer u = r + m * step`

___ `Copy particles[i], plus a little jitter, into the new set`

___ `Set step = 1/N and draw a single number r between 0 and step`

___ `While u > c, move i on and add weights[i] to c`

6. Why is low variance resampling preferred to drawing N independent samples? [1 mark]

- ☐ **A** It is less noisy, never gives zero copies to a particle that deserves one, and runs in O(N)
- ☐ **B** It needs no random numbers at all, so it is deterministic
- ☐ **C** It keeps the old weights, so no information is lost
- ☐ **D** It guarantees the resampled cloud has no duplicate particles

The task: resample

Weight a scattered cloud against one reading, then resample it. Print `before:` and `after:`, the effective sample size each side, and `spread:`, the standard deviation of the resampled cloud.

```
from bugbot import *
import math, random
connect()

N, SIGMA = 500, 3.0
particles = [random.uniform(0, 200) for i in range(N)]
```

Plan your program here, then type it in and press Run.

Do it on the robot

www.bugbotlab.com/learn/u7-5-resampling/

The simulator checks it and tells you when it passes. Nothing to install, no account.

Challenges

1. Resample ten times in a row without jitter and print how many distinct values are left.
2. Add jitter and repeat. How many now?
3. Inject 5 percent random particles each round and describe what it costs you when the filter is already right.