

# PIP INSTALL AXELROD

## PRISONERS' DILEMMA

- ▶ both sides are better off **Cooperating** (3)
- ▶ there is always a temptation to **Defect** (5)

|   | C    | D    |
|---|------|------|
| C | 3, 3 | 0, 5 |
| D | 5, 0 | 1, 1 |

WHEN INTERACTING WITH A  
SNEAKY OPPONENT  
SHOULD PEOPLE HOLD A GRUDGE  
AGAINST THEM?

|                 | 1 | 2 | 3 | 4 | 5 | 6 |
|-----------------|---|---|---|---|---|---|
| SneakyTitForTat | C | C | D | D | C | C |
| Grudger         | C | C | C | D | D | D |

```
>>> import axelrod as axl

>>> first_match = axl.Match([
...     axl.SneakyTitForTat(),
...     axl.Grudger()],
...     turns=20)

>>> first_match.play()[:6]
[('C', 'C'), ('C', 'C'), ('D', 'C'),
 ('D', 'D'), ('C', 'D'), ('C', 'D')]

>>> first_match.final_score()
(20, 55)

>>> second_match = axl.Match([
...     axl.SneakyTitForTat(),
...     axl.TitForTat()],
...     turns=20)

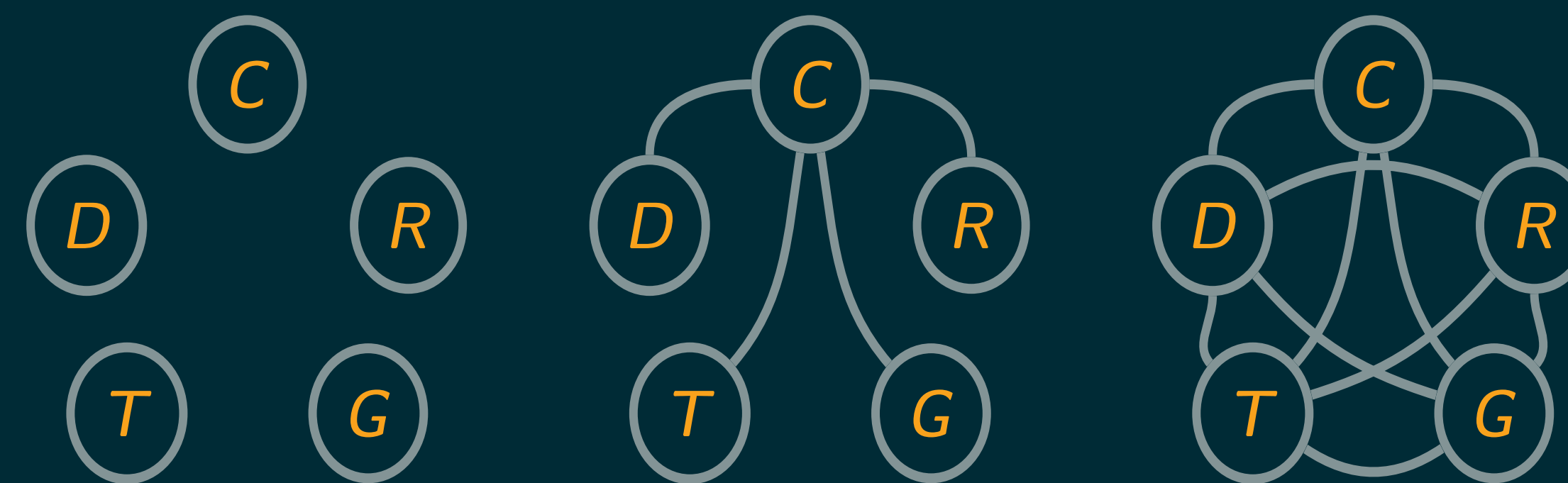
>>> _ = second_match.play()
>>> second_match.final_score()
(57, 57)
```

```
>>> assert axl.__version__ == "3.5.0"
```

```
$ python -m doctest poster.tex
```

```
# 3 expected failures for readability
```

WHAT IS THE OPTIMAL STRATEGIC  
PLAY AGAINST THE MANY FACES OF  
WAR?

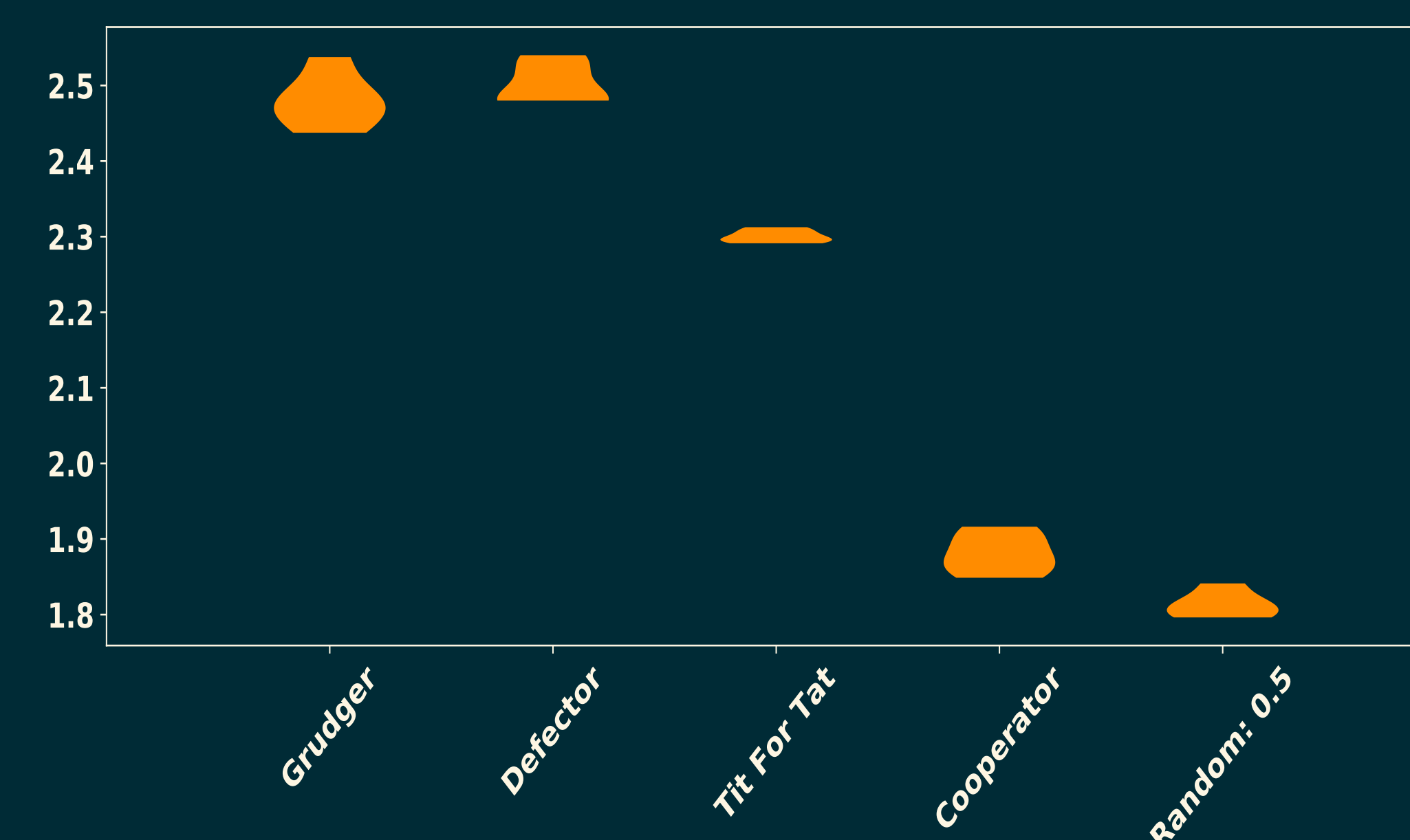


```
>>> import axelrod as axl

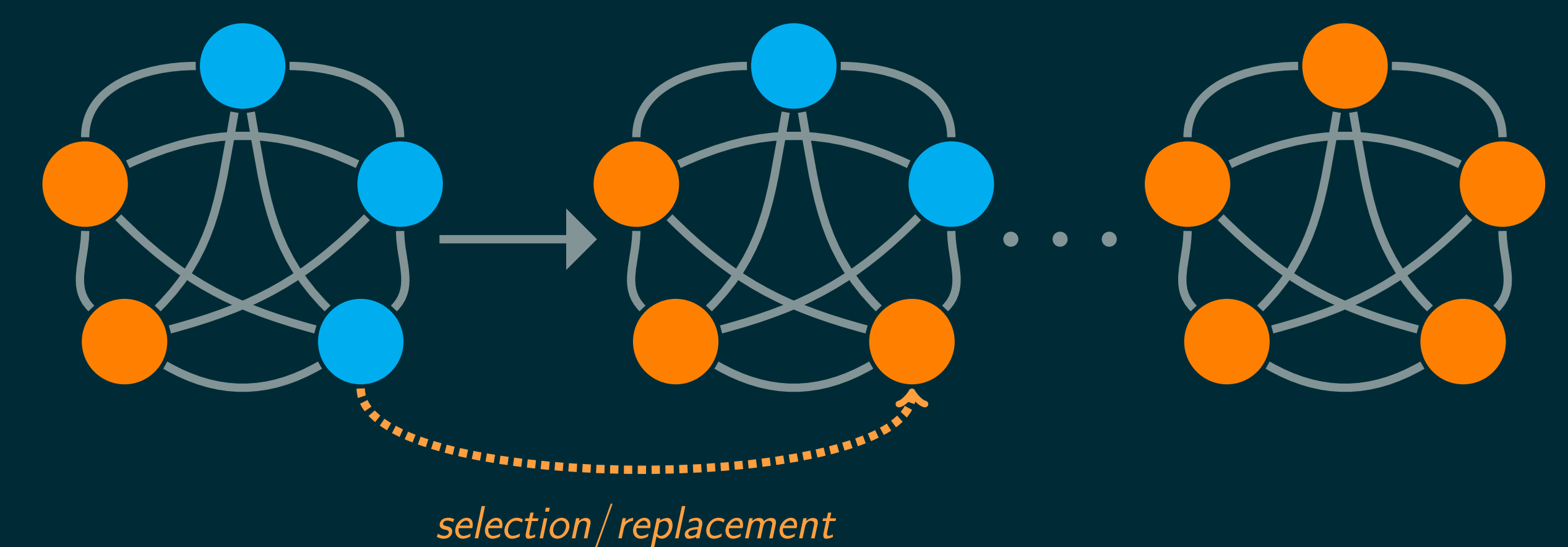
>>> axl.seed(0)
>>> players = [axl.Cooperator(), axl.Random(),
...             axl.TitForTat(), axl.Grudger(),
...             axl.Defector()]

>>> tournament = axl.Tournament(players)
>>> results = tournament.play()
>>> results.ranked_names
['Grudger', 'Defector', 'Tit For Tat',
 'Cooperator', 'Random: 0.5']

>>> plot = axl.Plot(results)
>>> p = plot.boxplot()
>>> p.show()
```



SHOULD THE NORTH JOIN HANDS  
WITH THE SOUTH TO DEFEAT THE  
NIGHT KING?



```
>>> import axelrod as axl
>>> import random

>>> N = 5
>>> players = []
>>> axl.seed(5)
>>> for _ in range(N):
...     player = random.choice([axl.Defector,
...                               axl.Cooperator])
...     players.append(player())

>>> mp = axl.MoranProcess(players=players, turns=200)
>>> mp.play()
[Counter({'Cooperator': 3, 'Defector': 2}),
 Counter({'Cooperator': 3, 'Defector': 2}),
 Counter({'Cooperator': 3, 'Defector': 2}),
 Counter({'Cooperator': 2, 'Defector': 3}),
 Counter({'Cooperator': 2, 'Defector': 3}),
 Counter({'Cooperator': 1, 'Defector': 4}),
 Counter({'Cooperator': 1, 'Defector': 4}),
 Counter({'Cooperator': 1, 'Defector': 4}),
 Counter({'Defector': 5})]
```

