**Doves and Hawks**

# **Andrew Muratov**

# Introduction

The hawk-dove model is an evolutionary game theoretical model developed by John Maynard Smith (1982) depicting the fundamental conflict between prosocial (altruism and cooperation) and antisocial behavior (selfishness). The model describes the contest between two fundamentally different behavioral strategies, hawks (selfishness) and doves (pro-sociality), when competing over a shared resource.

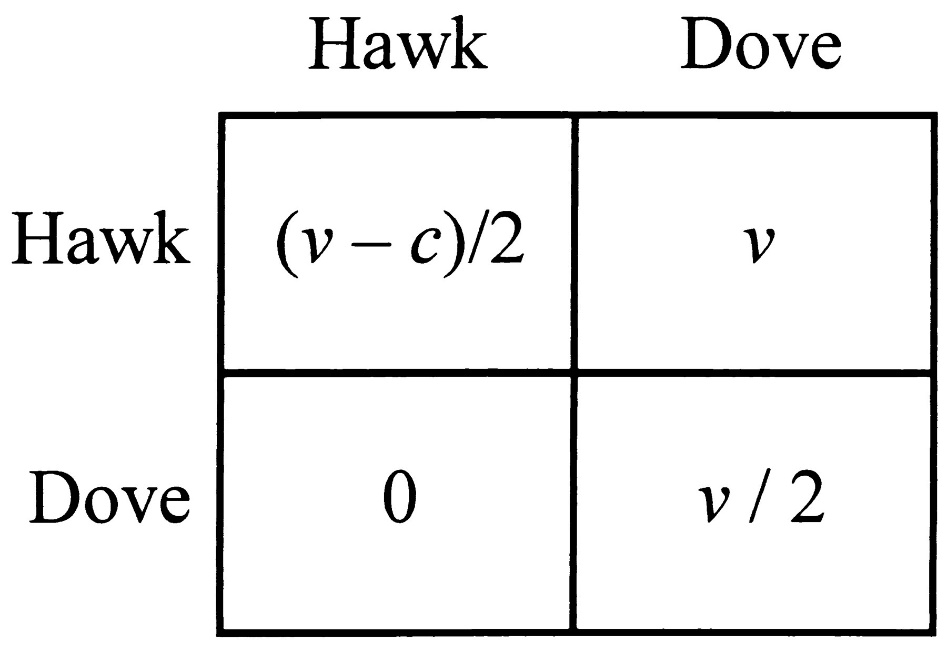
This contest reveals the evolutionary paradox of prosocial behavior (i.e., if natural selection is based on competition, then prosocial traits should not evolve). The hawk-dove model provides a simplistic framework to investigate the conditions that favor the evolution of prosocial behavior.

Overall, hawks outcompete doves within groups, but a group of doves outcompete a group of hawks. For either hawks or doves to evolve, the balance of selection within and between groups must tip in their respective favor.

I used this evolutionary model in my project to see what species of birds had a better chance of survival. I had thought that the Hawks were the more supreme species because they keep all of the recourses to themselves, but as it turns out, the Doves had many more times as many recourses than the Hawks, making their evolutionary model the best for all species.

V – amount of resources

C – Fight cost



# Simulation

I wrote a program using the Python programming language to simulate the interaction of Doves and Hawks. The program implements random interactions between the two classes of birds. Every instance of the class has an attribute to store its resource amount from all previous interactions. Every simulation initializes the number of birds in each species and runs a given number of interactions after which, it returns the table.

The criteria for the table was to choose the top 3 birds with the highest resource amount. If there are more doves than hawks, the doves win, and if there are more hawks than doves, the hawks win.

# Results

At the end of the program, we made for loop to give many values for the fight cost.

# Extras

The code of the program:

*import random  
  
n = 0  
m = 0  
  
# This functions gets the top three birds from the table, and prints which bird there's more of.  
# Hawks if there are more hawks, and Doves if there are more doves.  
  
def criteria(table\_of\_birds):  
 num\_doves = 0  
 num\_hawks = 0  
 top\_birds = table\_of\_birds[0:3]  
  
 for i in top\_birds:  
 if i[0] == "dove":  
 num\_doves += 1  
 else:  
 num\_hawks += 1  
  
 if num\_doves > num\_hawks:  
 return "Doves are better"  
 else:  
 return "Hawks are better"  
  
def simulation(n\_hawk, n\_dove, r, c):  
 class Dove:  
 def \_\_init\_\_(self):  
 global n  
 self.resource = 0  
 self.type = 'dove'  
 self.id = n  
 n += 1  
  
 def interaction(self, bird):  
 if bird.type == 'hawk':  
 self.resource += 0  
 elif bird.type == 'dove':  
 self.resource += r / 2  
  
  
 class Hawk:  
 def \_\_init\_\_(self):  
 global m  
 self.resource = 0  
 self.type = 'hawk'  
 self.id = m  
 m += 1  
  
 def interaction(self, bird):  
 if bird.type == 'hawk':  
 self.resource += r / 2 - c  
 elif bird.type == 'dove':  
 self.resource += r  
  
 list\_of\_birds = []  
 n\_interactions = (n\_dove + n\_hawk) \* 200  
  
 for i in range(n\_hawk):  
 list\_of\_birds.append(Hawk())  
 for i in range(n\_dove):  
 list\_of\_birds.append(Dove())  
  
 for i in range(n\_interactions):  
  
 j = random.randint(0, len(list\_of\_birds) - 1)  
 k = random.randint(0, len(list\_of\_birds) - 1)  
  
 bird1 = list\_of\_birds[j]  
 bird2 = list\_of\_birds[k]  
  
 bird1.interaction(bird2)  
 bird2.interaction(bird1)  
  
 table = [(bird.type, bird.id, bird.resource) for bird in list\_of\_birds]  
  
 table = sorted(table, key=lambda row: row[2], reverse=True)  
  
 return table  
for fight\_cost in range(0, 1000, 10):  
 table = simulation(50, 200, 100, fight\_cost)  
 print(criteria(table), fight\_cost)  
# print(\*table, sep="\n")*