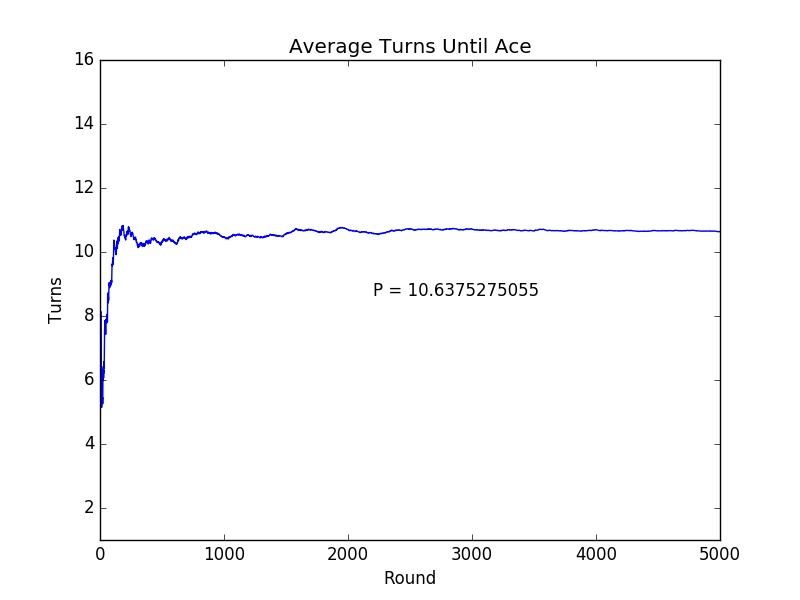
Francisco Fierro

EE 380

Lab 2

Problem 1:



import numpy as np

import matplotlib.pyplot as plt

import math

rounds = 5000

deck = np.arange(1,53)

foundAtArray = np.zeros(rounds)

foundAt = 0

for i in range(1, rounds):

deck = np.random.permutation(deck)

for j in range(52):

foundAt = foundAt + 1

if(deck[j] % 13 == 0):

break

foundAtArray[i] = foundAt/i

plt.plot(foundAtArray)

plt.title("Average Turns Until Ace")

plt.xlabel("Round")

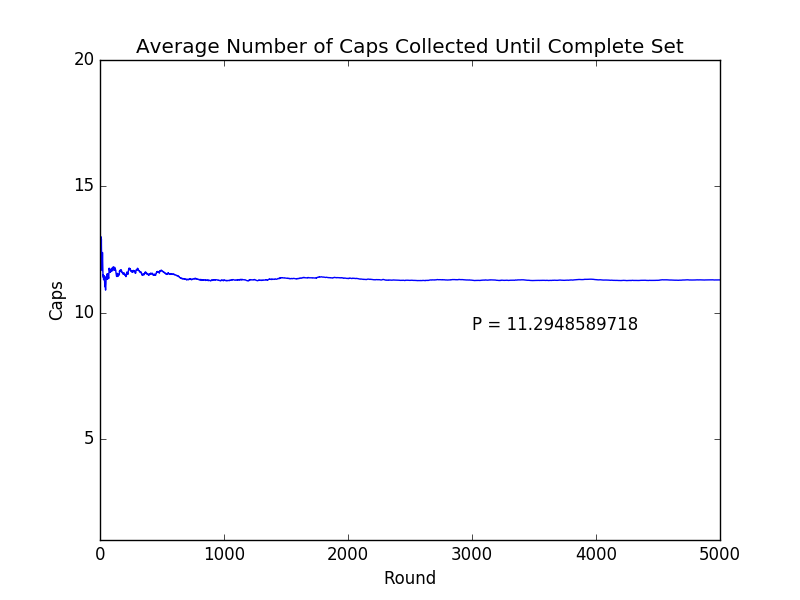
plt.ylabel("Turns")

plt.axis([1,rounds,1, int(math.ceil(foundAtArray[rounds-1]) + 5)])

plt.annotate("P = " + str(foundAtArray[rounds -1]), xy = (rounds - 2800, foundAtArray[rounds - 1] -2 ))

plt.show()

Problem 2:



import numpy as np

import matplotlib.pyplot as plt

import math

rounds = 5000

complete = False

collection = np.zeros(5)

capsCollected = 0

AvgCapsUntilComplete = np.zeros(rounds)

count = 0

for i in range(1,rounds):

for j in range(5):

collection[j] = 0

while(True):

count = 0

cap = np.random.randint(0,5)

capsCollected = capsCollected + 1

collection[cap] = 1

for k in range(5):

count = count + collection[k]

if(count == 5):

break

AvgCapsUntilComplete[i] = capsCollected / i

plt.plot(AvgCapsUntilComplete)

plt.title("Average Number of Caps Collected Until Complete Set")

plt.xlabel("Round")

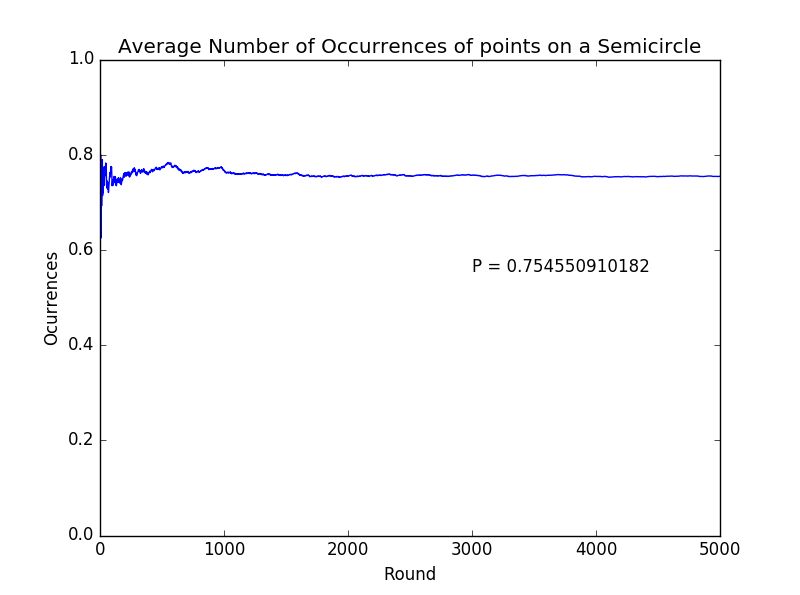
plt.ylabel("Caps")

plt.axis([1,rounds,1, 20])

plt.annotate("P = " + str(AvgCapsUntilComplete[rounds -1]), xy = (rounds - 2000, AvgCapsUntilComplete[rounds - 1] -2 ))

plt.show()

Problem 3:



import numpy as np

import matplotlib.pyplot as plt

import math

import numpy as np

rounds = 5000

avg = np.zeros(rounds)

inSemi = 0

p = np.zeros(3)

for i in range(1, rounds):

for j in range(3):

p[j] = np.random.random() \* 2 \* 180

p = np.sort(p)

if(p[1] - p[0] == 180):

inSemi = inSemi + 1

else:

if(p[1] - p[0] <= 180):

delta1 = p[1] -p[0]

if(p[0] - p[2] + 360 + delta1 <= 180 or p[2]- p[1] + delta1 <= 180):

inSemi = inSemi + 1

else:

inSemi = inSemi + 1

avg[i] = inSemi / i

plt.plot(avg)

plt.title("Average Number of Occurrences of points on a Semicircle");

plt.xlabel("Round")

plt.ylabel("Ocurrences")

plt.annotate("P = " + str(avg[rounds -1]), xy = (rounds - 2000, avg[rounds - 1] - .2))

plt.show()