ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ

ΣΧΟΛΗ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ ΚΑΙ ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ

ΘΕΩΡΙΑ ΠΛΗΡΟΦΟΡΙΑΣ

2^{η} ΣΕΙΡΑ ΓΡΑΠΤΩΝ ΑΣΚΗΣΕΩΝ

Ειρήνη Δόντη

ΑΜ: 03119839

9ο εξάμηνο

Αθήνα 2023-2024

Για την επίλυση των μαθηματικών σχέσεων της εκάστοτε άσκησης χρησιμοποιούμε κώδικα σε Python.

Άσκηση 1

1.

Ισχύει ότι $p = 0.1$. Όλες οι τυχαίες μεταβλητές έχουν την ίδια πιθανότητα και συνεπώς ισχύει ότι: $H(X) = -p\log_2(p) - (1-p)\log_2((1-p)) = 0.469$ Οποιαδήποτε ακολουθία μήκους n bits με k άσους εμφανίζεται με πιθανότητα $p^k(1-p)^{n-k}$, οπότε θέλουμε να βρούμε τα k για τα οποία ισχύει ότι:

$2^{-200(0.469+0.1)} \leq p^k(1-p)^{200-k} \leq 2^{-200(0.469-0.1)}$ Με χρήση κώδικα Python που παρατίθεται, βρίσκουμε ότι η παραπάνω ανισότητα ισχύει για $k \in [14,26]$

2.

Το πλήθος των ακολουθιών του τυπικού συνόλου ως ποσοστό του συνόλου όλων των δυνατών ακολουθιών είναι ίσο με: $\frac{\sum_{14}^{26}\binom{200}{k}}{2^{200}} = 2.2215 \cdot 10^{-28}$

3.

Για να υπολογίσουμε την πιθανότητα $\Pr\{A_\varepsilon^{(n)}\}$ του τυπικού συνόλου, δεδομένου ότι γνωρίζουμε ότι οι ακολουθίες που περιέχονται στο σύνολο είναι εκείνες που έχουν από 14 μέχρι 26 άσους, υπολογίζουμε απλά το άθροισμα (με την βοήθεια κώδικα στη Python) ως $\sum_{14}^{26}\binom{200}{k}p^k(1-p)^{200-k} = 0.8762$

Άσκηση 2

1.

Ισχύει ότι: E{logP(X1)} = P{X1 = 1}logP{X1 = 1} + P{X1 = 0}logP{X1 = 0} = p logp + (1 − p)log(1 − p) = −H(p)

2.

Υποθέτουμε ότι δεν είναι ερώτημα, αλλά υπόθεση για τα ερωτήματα που ακολουθούν.

3.

Δεδομένου ότι οι μεταβλητές είναι i.i.d. και έχουν την ίδια μάζα πιθανότητας, ισχύει ότι: φ(x1,x2,…,xn) = $-\frac{1}{n}$log(p(x1,x2,…,xn)) = $-\frac{1}{n}\log\prod_{i=1}^{n} p(xi)$ =

= $-\frac{1}{n}\sum_{i=1}^{n}\log(p(xi))$ = $-\frac{1}{n}\{\sum_{i=1}^{k}\log(p) + \sum_{i=1}^{n-k}\log(1-p)\ \}$ = $-\frac{1}{n}\{\,k\log(p) + (n-k)\log(1-p)\}$, εφόσον P{xi = 1} = p και P{xi = 0} = 1-p και συμβολίζουμε με k το πλήθος των xi που έχουν την τιμή 1 για μία δεδομένη ακολουθία x1,x2,…,xn.

4.

Επειδή οι τυχαίες μεταβλητές είναι ανεξάρτητες, η ζητούμενη πιθανότητα τμηματοποιείται ανάλογα με τους πόσους άσους περιέχει η ακολουθία x1, x2,…, xn. Για δεδομένα k, n η πιθανότητα να έχουμε μία οποιαδήποτε ακολουθία μήκους n με k άσους δίνεται με τον τύπο $\binom{n}{k}p^k(1-p)^{n-k}$. Οπότε, για δεδομένο n, εξετάζουμε την παραπάνω πιθανότητα ∀kϵ[0, n], εξετάζοντας αν είναι μεγαλύτερες από ε και στη συνέχεια να τις προσθέσουμε.

Pr{|φ(x1,x2,…,xn) + E{log(p(X1))}| > ε} = $\sum_{k\epsilon Kn}\binom{n}{k}p^k(1-p)^{n-k}$, με Kn το σύνολο kϵ[0, n] για τα οποία ισχύει ότι: |φ(x1,x2,…,xn) + E{log(p(X1))}| > ε ή

$-\frac{1}{n}\{k\log(0.1) + (n-k)\log(0.9)\}$ -H(p) > 0.5. Οι τιμές της ζητούμενης πιθανότητας (με χρήση κώδικα Python) για n = 20 είναι 0.0113 και για n = 40 είναι 0.0015 αντίστοιχα. Παρατηρούμε ότι, με τον διπλασιασμό της τιμής του n, η τιμή της ζητούμενης πιθανότητας μειώνεται με εκθετικό ρυθμό.

5.

Παρατηρούμε πως οι τιμές της πιθανότητας δε φαίνεται να προσεγγίζουν το 0, όπως ήταν θεωρητικά αναμενόμενο. Αυτό μπορεί να συμβαίνει εξαιτίας μικρών τιμών του n.

Παράθεση Κώδικα σε Γλώσσα Προγραμματισμού Python

Άσκηση 1

Είσοδος:

1.

```
from math import log, comb

p = 0.1
H_x = -p*log(p,2) - (1-p)*log(1-p,2)
print("H(x) =",H_x)

lowlimit = 2**(-200*(0.469+0.1))
uplimit = 2**(-200*(0.469-0.1))
ones_typical_set = [k for k in range(201) if lowlimit <= (1-
p)**(200-k)*p**k <= uplimit]
print(f"num of 1s that sequence is in typical set:
[{ones_typical_set[0]},{ones_typical_set[-1]}]")
```

2.

```
s = 0

least_ones = ones_typical_set[0]
max_ones = ones_typical_set[-1]

for k in range(least_ones, max_ones+1):
    s += comb(200, k)
pcnt = s/pow(2,200)
print(f"percentage of sequences in typical set to all
sequences: {pcnt}")
```

3.

```
#Probability of the given typical set

typical_set_pr = 0
for k in range(least_ones, max_ones+1):
    typical_set_pr += comb(200, k)*pow(p,k)*pow(1-p,200-k)
typical_set_pr = round(typical_set_pr, 4)
print(f"Probability of typical set: {typical_set_pr}")
```

Έξοδος:

1.

```
H(x) = 0.4689955935892812

num of 1s that sequence is in typical set: [14,26]
```

2.

```
percentage of sequences in typical set to all sequences:
2.221524186196646e-28
```

3.

```
Probability of typical set: 0.8762
```

Άσκηση 2

Είσοδος:

4.

```
from math import log, comb

def get_probability_for_k_n(k,n):
    p = 0.1
    return comb(n,k)*pow(p,k)*pow(1-p, n-k)

def check_k(k,n):
    err = 1/2
    p = 0.1
    H = -p*log(p,2) - (1-p)*log(1-p,2)
    val = ((-1/n) * (k*log(p,2) + (n-k)* log(1-p,2))) - H
    return abs(val) > err

#n = 20

s = 0
n = 20
for k in range(n+1):
      if check_k(k,n):
         s += get_probability_for_k_n(k,n)
print(f"for n: {n}, probability: {round(s,4)}")

#n = 40

s = 0
n = 40
for k in range(n+1):
      if check_k(k,n):
         s += get_probability_for_k_n(k,n)
print(f"for n: {n}, probability: {round(s,4)}")
```

Έξοδος:

4.

```
for n: 20, probability: 0.0113
for n: 40, probability: 0.0015
```