
Dodatak 1.5. – Stratificirani uzorak u Pythonu

Ovdje ćemo prikazati kako smo simulirali ankete spomenute u ovom odjeljku. Prvo ćemo brojeve iz tablice 1.4 pretvoriti u listu podataka.

```
[1]: import random

def generiraj_listu(n, permutiraj=False):
    """
        Kreira listu u kojoj se broj i pojavljuje
        ↳ n[i] puta za i=0,1,2,3.

        Parametri:
        - n: lista ili tuple od 4 cijela broja koji
        ↳ predstavljaju broj ponavljanja za svaki i
        - permutiraj: bool, ako je True, lista se
        ↳ slučajno permutira

        Povratna vrijednost:
        - lista s traženim brojevima
        """
    if len(n) != 4:
        raise ValueError("Ulazna lista mora sadržavati
        ↳ točno 4 elementa.")

    rezultat = []
    for i in range(4):
        rezultat.extend([i] * n[i])

    if permutiraj:
        random.shuffle(rezultat)

    return rezultat
```

Svaki član liste biti će jedan broj između 0 i 3, a ti brojevi predstavljati će informacije o pripadniku populacije:

- kojoj dobnoj skupini pripada;
- je li završio studij ili ne.

```
[2]: # kodiranje informacija

# 0 -> (0,0) -> 18-24, bez završenog studija
n0 = 29850 - 3980
```

```

# 1 -> (0,1) -> 18-24, završen studij
n1 = 3980

# 2 -> (1,0) -> 25+, bez završenog studija
n2 = 229800 - 88894

# 3 -> (1,1) -> 25+, završen studij
n3 = 88894

slucajno_poredana_lista = generiraj_listu([n0,
↪n1, n2, n3], True)

```

Prvo simulirano istraživanje (opisano na stranici 27) gdje slučajno biramo ispitanike i bilježimo podatke onih one koji su nam dali odgovor je implementirano u idućoj ćeliji.

```

[3]:
    broj_ispitanika = 0
    broj_ispitanika_sa_studijem = 0
    ciljani_broj_ispitanika = 1000
    brojac_pokusaja = 0

    while broj_ispitanika < ciljani_broj_ispitanika:
↪and brojac_pokusaja < len(slucajno_poredana_lista):
        brojac_pokusaja += 1
        slucajni_broj = random.uniform(0,1)
        tip_ispitanika =
↪slucajno_poredana_lista[brojac_pokusaja]

        ispitanik_nije_pristao = ((tip_ispitanika // 2 ==
↪0) and (slucajni_broj > 0.75)) or ((tip_ispitanika // 2 == 1)
↪and (slucajni_broj > 0.2))

        if ispitanik_nije_pristao:
            continue

        broj_ispitanika += 1
        broj_ispitanika_sa_studijem += tip_ispitanika % 2

        print("Ispitano: {}".format(broj_ispitanika))
        print("Studij završilo: {} ({:.2f} %)".
↪format(broj_ispitanika_sa_studijem,
↪(broj_ispitanika_sa_studijem * 100 / broj_ispitanika)))

```

```

Ispitano: 1000
Studij završilo: 284 (28.40 %)

```

Dobili smo tisuću odgovora, ali smo pokušali pitati puno više potencijalnih ispitanika.

```
[4]: brojac_pokusaja
```

```
[3]: 3613
```

Rezultati ankete u tablici 1.5 dobiveni su na idući način.

```
[4]: broj_ispitanika_po_skupinama = [0, 0]
      broj_ispitanika_sa_studijem_po_skupinama = [0, 0]
      ciljani_broj_ispitanika_po_skupinama_po_skupinama = [100, 1000]
      brojac_pokusaja = 0

      while broj_ispitanika_po_skupinama < ciljani_broj_ispitanika_po_skupinama_po_skupinama and brojac_pokusaja < len(slucajno_poredana_lista):
          brojac_pokusaja += 1
          slucajni_broj = random.uniform(0,1)
          tip_ispitanika = slucajno_poredana_lista[brojac_pokusaja]
          dobna_skupina = tip_ispitanika // 2
          obrazovna_skupina = tip_ispitanika % 2

          ispitanik_je_pristao = ((dobna_skupina == 0) and (slucajni_broj < 0.75)) or ((dobna_skupina == 1) and (slucajni_broj < 0.2))

          if ispitanik_nije_pristao or broj_ispitanika_po_skupinama[dobna_skupina] >= ciljani_broj_ispitanika_po_skupinama_po_skupinama[dobna_skupina]:
              continue

          broj_ispitanika_po_skupinama[dobna_skupina] += 1
          broj_ispitanika_sa_studijem_po_skupinama[dobna_skupina] += tip_ispitanika % 2

          print("Ispitano: {}".format(broj_ispitanika_po_skupinama))
          print("Studij završilo: {}".format(broj_ispitanika_sa_studijem_po_skupinama))
```

```
Ispitano: [100, 1000]
```

Studij završilo: [17, 370]