

PRILOGA K PROGRAMU: DOKUMENTACIJA (model 1)

Opis programa

Program je narejen na statistični podlagi in sicer je osnovna ideja zasnovana na predpostavljeni normalni porazdelitvi ter na uporabi povprečja in standardnega odklona, saj je povprečje tisto, ki ga želimo nadzorovati (in spreminjati), standardni odklon pa je od njega neposredno odvisen oz. sta tesno povezana.

Ideja je zasnovana na generiranju naključnih populacij oz. populacij, ki kar se da natančno predstavljajo preteklo in trenutno dodeljevanje popustov. Za natančnejšo analizo in natančnejše rezultate naključno populacijo generiramo n -krat, da zaradi naključne izbire dobimo realnejše podatke. Pri pridobivanju rezultatov vedno izbiramo mediano.

Algoritem dela s pomočjo štirih populacij, ki jih uporabljamo skozi korake algoritma. Te so:

- Učna populacija P_u : je populacija preteklega opazovanega obdobja. Iz nje, ob predpostavki da je porazdeljena normalno, dobimo povprečje in standardni odklon.
- Realna populacija P_r : realni dodeljeni popusti od nekega določenega datuma naprej.
- Polje simuliranih populacij $P_s[]$: element polja je naključna populacija, ki je po predpostavki normalno porazdeljena ter ima željeno ciljno povprečje in standardni odklon. Povprečje se od povprečja populacije P_u razlikuje za μ_d (za koliko odstotkovnih točk želimo spremeniti povprečje), standardni odklon pa se od standardnega odklona populacije P_u razlikuje za σ_d (za koliko % želimo spremeniti standardni odklon).
- Nova populacija P_n : realni dodeljeni popusti tekočega dne.

Potrebne pretekle podatke dobimo iz baze, kjer so shranjeni znani pretekli podatki (v našem primeru je to .csv datoteka).

Potrebni podatki so:

- ZAVAROVALNA_POGODBA_ID - identifikacijska številka zavarovalne pogodbe
- DT_SKLENITVE (KOLEDARSKI DATUM) - datum vnosa police v bazo ali datum sklenitve zavarovalne pogodbe podan v obliki 1111mmdd, kjer 1111 predstavlja leto, mm mesec in dd dan sklenitve

- PE - poslovna enota oz. segment podana v obliki PE kk, kjer kk predstavlja kratico mesta
- DODELJENI_POPUSTI_952_ODS - dodeljeni popust zavarovalne pogodbe, podan z vrednostjo med 0 in 1, kjer je decimalni del od celega ločen s piko

Podatki so v .csv datoteki med seboj ločeni s podpičji.

Spodaj vidimo sestavo tabele, v kateri je tudi naveden primer vnosa ene vrstice oz. podatkov ene zavarovalne police.

ZAVAROVALNA_POGODBA_ID	DT_SKLENITVE	PE	DODELJENI_POPUSTI_952_ODS
15230422	20131016	PE MB	0.33

Psevdokoda

Razlago spremenljivk in programa najdete v poglavju Koraki programa.

Data: $P_u, \mu_d, \sigma_d, stPop$

Result: M, nmg // M-meja za avtoriziranje, nmg-namig avtorizantu

Initialization of local variables:

// s $|P_n|$ označimo moč populacije P_n ;

$m :=$ pričakovano število popustov;

$P_s[] :=$ polje vseh simuliranih populacij velikosti $stPop$;

$nmgArray[] :=$ polje namigov za trenutni dan velikosti $stPop$;

$MArray[] :=$ polje mej za trenutni dan velikost $stPop$;

$\mu = \text{mean}(P_u) + \mu_d$;

$\sigma = \text{std}(P_u) + \sigma_d$;

$M_z := \mu + \sigma$ // začetna meja;

$k := 1$ // koeficient za premikanje meje;

for $i:=1$ **to** $stPop$ **do**

$P_s[i] :=$ naključna normalno porazdeljena populacija velikosti m z μ in σ ;

end

$P_r := \emptyset$;

$P_n := \emptyset$;

for vsak dan **do**

for $i:=1$ **to** $stPop$ **do**

$k := \frac{\mu \cdot m - \sum_{p \in P_r} p}{\sum_{p \in P_s} p}$;

$MArray[i] := k \cdot M_z$ // nova trenutna meja;

$h = \frac{\mu \cdot m - \sum_{p \in P_r} p - \sum_{p \in P_s[i], p < M} p}{\sum_{p \in P_s[i], p \geq M} p}$;

$nmgArray[i] := 1 - h$;

$P_n :=$ nove prejete vrednosti;

 iz $P_s[i]$ odstrani $|P_n|$ naključnih vrednosti;

$P_r := P_r + P_n$;

end

$M = \text{mediana}(MArray)$ //meja za ta dan;

$nmg = \text{mediana}(nmgArray)$ // namig za ta dan;

end

Algorithm 1: Psevdokoda programa

Koraki programa

1. Izračunamo σ (standardni odklon) in μ (povprečje) učne populacije P_u .
2. Generiramo polje velikosti $stPop$ simuliranih populacij $P_s[]$. Posamezna populacija $P_s[i]$ je velikosti m , katero običajno določimo na podlagi števila popustov v enakem obdobju leto nazaj in indeksa rasti.
3. Inicializiramo polja $MArray[]$ in $nmgArray[]$ velikosti $stPop$, za shranjevanje mej in namigov posameznih naključnih populacij.
4. Realna populacija P_r je na začetku prazna.
5. Nova populacija P_n je na začetku prazna.

Dnevno ponavljamo naslednje korake:

Ponavljamo za $i := 1$ do $stPop$ točke 6 do 12:

6. Izračunamo k iz enačbe:

$$\mu = \frac{\sum_{p \in P_r} p + k \cdot \sum_{p \in P_s[i]} p}{m},$$

kjer je μ povprečje od P_u , P_r je realna populacija, P_n nova populacija, $P_s[i]$ simulirana populacija in m velikost simulirane populacije.

Če $k = 1$, pomeni, da mejo ohranjamo oz. se ta ne spremeni.

7. Za mejo $MArray[i]$ postavimo vrednost $k \cdot (\mu + \sigma)$.
8. Izračunamo h iz enačbe:

$$\mu = \frac{\sum_{p \in P_r} p + \sum_{p \in P_s, p < M} p + h \cdot \sum_{p \in P_s, p \geq M} p}{m}$$

9. Za namig $nmgArray[i]$ postavimo vrednost $1 - h$
10. Novo populacijo P_n prepisemo z novimi prejetimi vrednostmi.
11. Iz simulirane populacije $P_s[i]$ naključno odstranimo toliko elementov oz. popustov, kolikor jih je v novi populaciji P_n .
12. K realni populaciji P_r dodamo novo populacijo P_n .
13. Ko zaključimo z vsemi simuliranimi populacijami določimo namig $nmg = \text{mediana}(nmgArray)$ in mejo $M = \text{mediana}(MArray)$

Funckije

Funkcija *generirajModel*, ki je tudi glavna funkcija programa, ima naslednje parametre:

- *datumOd* - začetni datum učnega obdobja
- *datumDo* - končni datum učnega obdobja
- *datumOd2* - začetni datum realnega obdobja
- *datumDo2* - končni datum realnega obdobja
- *privzetaMejaZaAvtorizacijo* - sedanja oz. trenutna fiksna privzeta meja za avtoriziranje. Potrebujemo jo za primerjavo števila potrebnih avtorizacij pri tej meji s številom potrebnih avtorizacij pri meji, ki jo vrne program, da bi ugotovili, za koliko se število avtorizacij zmanjša (ali poveča)
- *dPovprecje* (μ_d) - za koliko odstotkov želimo spremeniti povprečje (če ga hočemo zmanjšati, mora biti negativno, vrednosti morajo biti med 0 in 1)
- *dSD* (σ_d) - analogno kot dPovprecje, le da gre za standardni odklon
- *dAvtorizacije* - za koliko znižamo popust, če se znajde v avtorizaciji
- *stPop* - koliko naključnih populacij bomo generirali
- *segment* oz. zavarovalniška enota (PE) - opsijski parameter

Funkcija *dolociMejo* vrne mejo za avtoriziranje za izbran datum.

Program vrne naslednje podatke:

- *stAvtorizacij* (koliko avtorizacij je v testnem obdobju bilo potrebno) in *stAvtorizacijPrivzeto* (koliko avtorizacij bi bilo potrebno pri neki fiksni privzeti meji)
- Razliko med *stAvtorizacij* in *stAvtorizacijPrivzeto* in razmerje oz. odstotek zmanjšanja števila avtorizacij.

Navodila uporabniku

Uporabnik mora vnesti parametre funkcije *generirajModel* (opis parametrov je zapisan pri opisu funkcije). Po vnosu parametrov program vrne *mejo za avtoriziranje* za vsak dan testnega obdobja. Parametre funkcije *dolociMejo* uporabnik ne rabi navajati, saj se prenesejo iz funkcije *generirajModel*.

Primer preprostega klica funkcije

```
simulator <- generirajModel("20140101","20140131",  
  "20140201","20140228", 0.37, dPovprecje = -0.02,  
  dSD = -0.03, dAvtorizacije = -0.05, stPop = 10)  
print(simulator)
```

Funkcija bo simulirala obdobje od 1. februarja do 28. februarja s podanimi parametri in nato shranila v objekt `simulator` število avtorizacij z uporabo meje, ter primerjavo z številom avtorizacij, če bi uporabljali privzeto mejo. Te podatke nato izpišemo.