

Семинарски рад

Анђела Дамњановић

2023-04-21

Увод

Циљ истраживања

Узорковање у статистици подразумева одабир дела популације за добијање потребних података за анализу. То чини процес прикупљања података лакшим, бржим и јефтинијим. Понекад је могуће проучити читаву популацију, али ако је она превелика, онда је практичније одабрати *узорак* (део популације над којим ће бити спроведено истраживање). Најбитнија особина коју би узорак требало да има јесте *репрезентативност*, тј. карактеристике јединки које улазе у узорак требало би да што верније осликавају карактеристике читаве популације. Такође, обим узорка би требало да буде пропорционалан укупном обиму популације. Узорак који је већи него што је потребно биће боље репрезентативан за популацију и стога ће дати тачније резултате. Међутим, након одређене тачке, повећање тачности ће бити мало. Насупрот томе, узорак који је мањи него што је потребно не би имао довољну статистичку моћ да одговори на примарно истраживачко питање, а статистички незнатан резултат могао би бити само због неадекватне величине узорка (тип 2 или лажно негативна грешка).

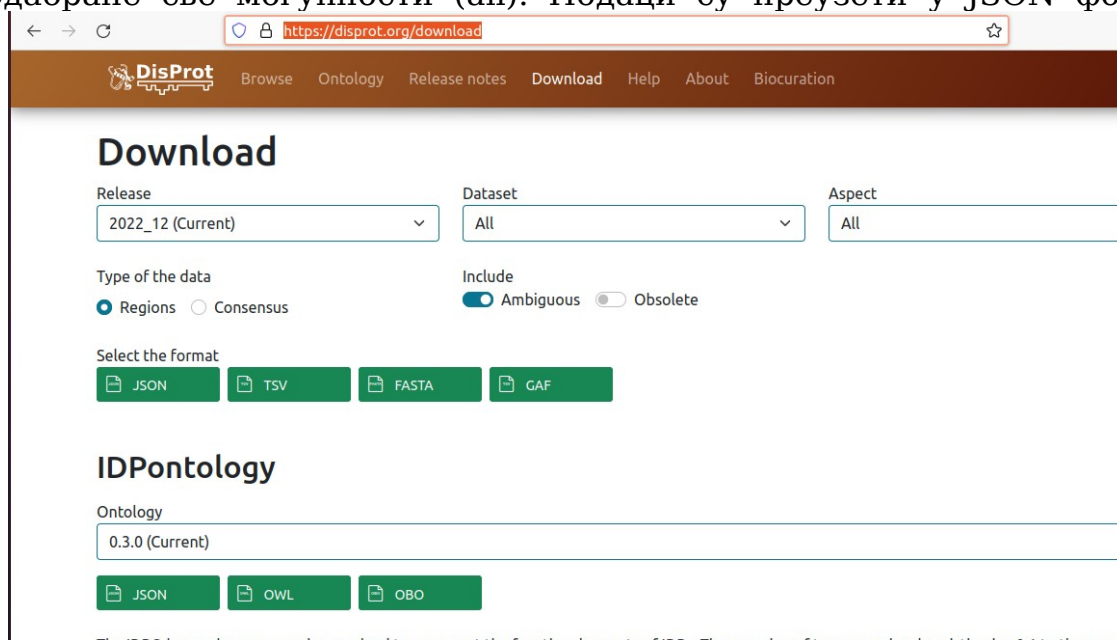
Узорковање је подељено у две категорије: вероватносно и невероватносно. Прва врста подразумева се свакој јединки из популације може придружити одређена вредност - вероватноћа да ће она бити изабрана у узорак. Предност оваквог начина узорковања је тај што се могу добити оцене параметара, као и оцене грешки. Са друге стране, невероватносно узорковање је метод одабира јединки из популације коришћењем субјективног (тј. неслучајног) метода. Пошто оно не захтева комплетан оквир анкете, то је брз, лак и јефтин начин добијања података. Међутим, да би се из узорка извукли закључци о популацији, мора се претпоставити да је узорак репрезентативан за популацију. Ово је често ризична претпоставка у случају невероватносног узорковања због потешкоћа у процени да ли претпоставка важи. Поред тога, пошто се елементи бирају произвољно, не постоји начин да се процени вероватноћа да ће било који елемент бити укључен у узорак. Такође, није дата гаранција да свака ставка има шансу да буде укључена, што онемогућава и процену варијабилности

узорка или идентификацију могуће пристрасности, те се овакве технике не могу оценити.

Циљ овог рада јесте демонстрација различитих техника узорковања, оцењивања, као и анализа добијених резултата.

Упознавање са подацима за рад

Скуп података који је коришћен за израду овог рада може се наћи у бази неуређених протеина DisProt (<https://disprot.org/download>). Када се приступи сајту, добијају се могућности одабира верзије података, врсте протеина, као и аспеката. Након извршеног одабира, потребно је одабрати и формат. За потребе овог рада преузета је актуелна верзија (у време израде то је 2022_12), док су за скупове протеина и аспекте одабране све могућности (all). Подаци су преузети у JSON формату.



Слика 5 Подешавања

Све информације које се налазе у овој бази података су проверене и не очекују се елементи ван граница, тако да нема потребе за претпроцесирањем, већ је могуће одмах радити са готовим подацима. У самој бази налазе се подаци за 2470 протеина. Неке од карактеристика протеина које се налазе у JSON фајловима су:

1. фамилија протеина - носи информације о идентификатору фамилије протеина, имену, почетку и крају
2. почетак и крај неуређене секвенце
3. стринговни приказ секвенце протеина
4. таксономија- у којој врсти је пронађен дати узорак
5. идентификатор у оквиру базе неуређених секвенци- идентификатор таксономије

6. број неуређених региона који се налазе у датом протеину, као и информације о самим регионима

7. региони: садрже информације о стању протеина, онтологији, крају и идентификатору региона, референце (текст са информацијама), идентификатор термина

8. неке занимљиве статистике као што је проценат протеина који садржи неуређене регионе.

Претпроцесирање података и налажење директних и индиректних некомплементарних секвенци

Да бисмо за ове податке могли да покренемо програм StatRepeats, морамо их прво преbacити у .fasta формат јер програм ради искључиво са овим форматом ¹. За добијање одговарајућег .fasta фајла довољно је покренути програм toFasta.ipynb који ће обрадити прослеђени JSON фајл (који треба да буде назван podaci.json) и на основу њега направити датотеку preprocessed.fasta коју сада можемо проследити програму. Поменути програм користи библиотеку Bio, која је део колекција некомерцијалних Пајтон алата за рачунарску биологију и биоинформатику и садржи класе које представљају биолошке секвенце и анотације секвенци. Библиотека такође може да чита и пише у различите формате датотека, па је за инсталацију те библиотеке потребно у терминалу откуцати наредбу `pip install Bio`, или, уколико покрећете програм из Јупитера, `!pip install Bio` ². Добијени .fasta фајл садржи 26.488 линија, а део његовог садржаја може се видети на Слици 6.

¹ У биоинформатици и биохемији, fasta формат је текстуални формат за представљање или нуклеотидних секвенци или аминокиселинских (протеинских) секвенци, у којима су нуклеотиди или аминокиселине представљени помоћу једнословних кодова. Састоји се од заглавља, које почиње знаком > којег прати идентификатор и даље опис протеина/ДНК секвенце и тела у коме се налази сама секвенца

² Уколико се програм не покреће из Јупитера, већ из терминала, довољно је копирати код у .ру фајл и покренути га из терминала. Са друге стране, ако се програм покреће из Јупитера, линије кода које имају улогу да врше испис могу се откоментарисати

```
home > andjela > Desktop > ipSeminarski > preprocessed.fasta
1 >DP000003 Human adenovirus C serotype 5
2 MASREEEQRETTPERGRGAARRPPTMEDVSSPSPPPPRAPPKKMRRIESEDEEDSS
3 QDALVPRTPSPRPSTSAADLAIAPKKKKRPSPKPERPSPPEVIVDSEEEERDVALQMVG
4 FSNPPVLIKHGKGGKRTVRRLNEDDPVARGMRTQEEEEEPSEAESEITVMNPLSVPIVSA
5 WEKGMEAAARALMDKYHVDNDLKANFKLLPDQVEALAAVCKTWLNEEHRGLQLTFTSKKTF
6 VTMGRFLQAYLQSF AEVTKHHEPTGCALWLHRC AEIEGELKCLHGSIMINKHVIEMD
7 VTSENGQRALKEQSSKAKIVKNRWGRNVVQISNTDARCCVHDAACPANQFSGKSCGMFFS
8 EGAQAQVAFKQIKAFMQALYPNAQTGHGHLMLRCECNSKPGHAPFLGRQLPKLTPFAL
9 SNAEDLDADLISDKSVLASVHHPALIVFOCCNPVYRNSRAQGGGPNCDFKISAPDLLNAL
10 VMVRSLSWSENFTELPRMVVPEFKWSTKHQYRNVSLPVAHSDARQNPFDF
11 >DP000004 Homo sapiens
12 MKTQRDGHSLGRWSLVLLLLGLVMPLAIIAQVLSYKEAVLRAIDGINQRSSDANLYRLLD
13 LDPRPTMDGDPDTPKPVSVFTVKETVCPRTTQSQSPEDCDFKDGGLVKRCMGTVTNLNARGS
14 FDISCDKNRFRALLGDFRKSKEKIGKEFKRIVQRIKDFLRNLVPRTES
15 >DP000005 Escherichia phage lambda
16 MDAQTRRRERRAEKQAQWKAANPLLVGVSAPVNRPILSLNRKPKSRVESALNPIDLTVL
17 AEYHKIESNLQRIERKNQRTWYSKPGERGITCSGRQKIKGSKIPI
18 >DP000006 Equus caballus
19 MGDVEKGKKIFVQKCAQCHTVKEGKHKHTGPNLHGLFGRKTGQAPGFTYTDANKNKGITW
20 KEETLMYELENPKKIYIPGTMIFAGIKKTEREDLIAYLKKATNE
21 >DP000007 Homo sapiens
22 MPKRGGKGAEDGDELRTPEAKKSKTAACKNDKEAAGEGPALYEDPPDQKTSPSGKPA
23 TLKICSWNVVDGLRAWIKKGLDWVKEEAPDILCLQETKCSENKLPALQELPGLSHQYWS
24 APSDKEGSGVGLLSRQCPLKVSYGIGDEEHDQEGRVIVAEFDSFVLVTAYVNPAGRGLV
25 RLEYRQWDEAFRKLGLASRKPLVLCGLNVAHEEIDLNRNPKGNKNAGFTPQERQGF
26 GELLQAVPLADSFRHLYPNTPYAYTFWYMMNARSKNVGWRLDYFLLSHSLPALCDSKI
27 RSKALGSDHCPITLYLAL
28 >DP000008 Mus musculus
29 MLWQKSTAPEQAPAPRPYQGVVRVKEPVKELLRRKRGHTSVGAAGPPTAVVLPHPQPLATY
30 STVGPSCLDMEVSASTVTEEGTLCAGWLSQAPATLQPLAPWPTYEYVSHEAVSCPYST
31 DMVVPVQVPSYTVVGPSSVLTYASPLITNVTPRSTATPAVGPQLEGPEHQAPLTYFPWP
32 QPSTLPTSSLYQVPPARTLSGDEEVLPTSTPEPVLQMDNDPBRATSSITDKLLEFF
```

Слика 6-Изглед добијеног .fasta фајла

Сада, када имамо податке у оном формату који нам одговара, можемо да покренемо програм StatRepeats који ће нам пронаћи директне некомплементарне секвенце. Пошто нема смисла да узимамо секвенце дужине 1, то овде нећемо радити. За потребе овог рада, програм је покретан више пута, за различите минималне дужине секвенци, па је због тога написан скрипт extractAllInfo.py чија је улога да покрене програм за различите улазне вредности, а који је потребно покренути у терминалу (пре тога је неопходно позиционирати се у терминалу у онај фолдер који садржи извршну верзију StatRepeats програма). Свака наредба која се налази у горепоменутом скрипту позива програм StatRepeats за своје параметре и као резултат извршавања програма креира фајл са називом directNall.out (где N означава минималну дужину секвенце за коју је покренут, а ознака direct говори да ли је програм покренут за директне понављајуће секвенце). Сваки новогенерисани фајл садржи следеће информације:

1. ако у протеину не постоји секвенца дужа од минималне дужине, исписује само идентификатор датог протеина и информацију да су сва слова у секвенци из дозвољеног алфабета

2. ако у протеину постоји/постоје секвенца/е дуже од минималне дужине, онда се, поред информација које се исписују и у случају да нема секвенци исписују и следеће ствари: индекс почетка прве понављајуће секвенце, индекс краја прве понављајуће секвенце, индекс почетка друге понављајуће секвенце, индекс краја друге понављајуће секвенце, дужина поновљене секвенце, део секвенце који се понавља ,

као и укупан број понављајућих секвенци свих дужина које је програм нашао (Слика 7)

[illegible]

На слици видимо део генерисаног садржаја. Из њега можемо видети да протеини са идентификаторима DP00062, DP00063 и DP00064 немају понављајућу секвенцу дужу од оне којом је покренут програм, док DP00065 има. Секвенце које задовољавају тај услов су затим наведене, као и њихова статистика, тј. број појављивања секвенце те дужине у протеину

За каснију анализу могу нам бити корисне информације о броју секвенци које испуњавају услов, па преглед нудимо у оквиру табеле на Слици 8.

Minimal sequence length	Number of direct sequences	Number of indirect sequences
2	4 561 323	4 683 431
3	408 546	397 087
8	3 845	2 132
10	2 005	1 128
17	762	379
20	583	284
50	153	48
100	58	/

Table 1: Number of direct and indirect sequences depending on the minimal length of the sequence

Слика 8 Табела која приказује број директних и индиректних секвенци у зависности од минималне дужине секвенце

Међутим, иако овај фајл садржи све што нам је потребно да бисмо наставили рад, за даљу обраду било би доста једноставније када бисмо ове податке ипак чували у различитим фајловима. Да бисмо ово успели, довољно је покренути skript.py из терминала (поново морамо бити позиционирани у онај директоријум где нам се налази извршни StatRepeats програм). Једина разлика између овог и extractAllInfo.py програма је тај што је овај скрипт покренут са опцијом -load directN која функционише тако што аутоматски генерише 3 фајла:

1. directN.fasta.id у коме се у свакој линији налази по један идентификатор протеина из базе (пошто је програм покретан увек над истим подацима, сви генерисани .id фајлови имаће исти садржај- Слика 9)

2. directN.fasta.load који садржи информације о директним секвенцама и то: идентификатор протеина, индекс почетка прве понављајуће секвенце, индекс краја прве понављајуће секвенце, индекс почетка друге понављајуће секвенце, индекс краја друге понављајуће секвенце, дужина поновљене секвенце, део секвенце који се понавља , као и укупан број понављајућих секвенци свих дужина које је програм нашао -Слика 10

3. directN.fasta.stat у коме се, као што име и сугерише, налазе статистички подаци о сваком протеину из базе садржи податке о укупном броју секвенци одређене дужине, при чему се тај број креће од минимума који је задат па све до најдуже секвенце која се поклапа (Слика 11).

1 DP00003,
2 DP00004,
3 DP00005,
4 DP00006,
5 DP00007,
6 DP00008,
7 DP00009,
8 DP00011,
9 DP00012,
10 DP00013,
11 DP00015,
12 DP00016,
13 DP00017,
14 DP00018,
15 DP00020,
16 DP00021,
17 DP00023,
18 DP00024,
19 DP00025,
20 DP00026,
21 DP00027,
22 DP00028,
23 DP00029,
24 DP00030,
25 DP00031,
26 DP00032,
27 DP00033,
28 DP00034,
29 DP00036,
30 DP00040,
31 DP00041,
32 DP00042,
33 DP00043,
34 DP00044,
35 DP00047,

Слика 9 Изглед генерисаног .id фајла за директне секвенце дуже од 17

```
1 DP00017,197,214,199,216,18,APAPAPAPAPAPAPAP,APAPAPAPAPAPAPAP
2 DP00025,879,915,893,929,37,PTPTTPEVPSEPETPTPTTPEVPSEPETPTPTTPEVP,PTPTTPEVPSEPETPTPTTPEVPSEPETPTPTTPEVP
3 DP00025,879,901,907,929,23,PTPTTPEVPSEPETPTPTTPEVP,PTPTTPEVPSEPETPTPTTPEVP
4 DP00034,86,184,248,266,19,EGGSEGGGSEGGGSEGGG,EGGSEGGGSEGGGSEGGG
5 DP00034,244,262,249,267,19,GGGSEGGGSEGGGSEGGG,GGGSEGGGSEGGGSEGGG
6 DP00034,87,184,244,261,18,GGGSEGGGSEGGGSEGGG,GGGSEGGGSEGGGSEGGG
7 DP00065,1169,1253,1171,1255,85,DSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDS,DSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDS
8 DP00065,1169,1251,1173,1255,83,DSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDS,DSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDS
9 DP00065,1169,1249,1175,1255,81,DSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDS,DSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDS
10 DP00065,1169,1247,1177,1255,79,DSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDS,DSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDS
11 DP00065,1169,1245,1179,1255,77,DSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDS,DSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDS
12 DP00065,1169,1243,1181,1255,75,DSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDS,DSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDS
13 DP00065,1169,1241,1183,1255,73,DSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDS,DSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDS
14 DP00065,1169,1239,1185,1255,71,DSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDS,DSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDS
15 DP00065,1169,1237,1187,1255,69,DSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDS,DSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDS
16 DP00065,1169,1235,1189,1255,67,DSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDS,DSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDS
17 DP00065,1169,1233,1191,1255,65,DSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDS,DSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDS
18 DP00065,1169,1231,1193,1255,63,DSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDS,DSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDS
19 DP00065,1169,1229,1195,1255,61,DSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDS,DSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDS
20 DP00065,1169,1227,1197,1255,59,DSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDS,DSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDS
21 DP00065,1169,1225,1199,1255,57,DSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDS,DSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDS
22 DP00065,1169,1223,1201,1255,55,DSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDS,DSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDS
23 DP00065,1169,1221,1203,1255,53,DSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDS,DSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDS
24 DP00065,1169,1219,1205,1255,51,DSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDS,DSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDS
25 DP00065,1169,1217,1207,1255,49,DSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDS,DSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDS
26 DP00065,1169,1215,1209,1255,47,DSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDS,DSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDS
27 DP00065,1169,1213,1211,1255,45,DSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDS,DSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDS
28 DP00065,1169,1211,1213,1255,43,DSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDS,DSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDS
29 DP00065,1169,1209,1215,1255,41,DSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDS,DSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDS
30 DP00065,1169,1207,1217,1255,39,DSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDS,DSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDS
31 DP00065,1169,1205,1219,1255,37,DSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDS,DSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDS
32 DP00065,1169,1203,1221,1255,35,DSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDS,DSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDS
33 DP00065,1169,1201,1223,1255,33,DSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDS,DSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDS
34 DP00065,1169,1199,1225,1255,31,DSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDS,DSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDS
```

Слика 10 Изглед генерисаног .load фајла за директне секвенце дуже од 17

```
Total for length 17 is 5.
Total for length 19 is 4.
Total for length 21 is 4.
Total for length 23 is 4.
Total for length 25 is 4.
Total for length 27 is 4.
Total for length 29 is 30.
Total for length 31 is 1.
Total for length 33 is 1.
Total for length 35 is 1.
Total for length 37 is 1.
Total for length 39 is 1.
Total for length 41 is 1.
Total for length 43 is 1.
Total for length 44 is 1.
Total for length 45 is 1.
Total for length 47 is 1.
Total for length 49 is 1.
Total for length 51 is 1.
Total for length 53 is 1.
Total for length 55 is 1.
Total for length 57 is 1.
Total for length 59 is 1.
Total for length 61 is 1.
Total for length 63 is 1.
Total for length 65 is 1.
Total for length 67 is 1.
Total for length 69 is 1.
Total for length 71 is 1.
Total for length 73 is 1.
Total for length 75 is 1.
Total for length 77 is 1.
Total for length 79 is 1.
```

Слика 11 Изглед генерисаног .stat фајла за директне секвенце дуже од 17

Треба напоменути да је и овде, као и у претходном случају, N минимална дужина секвенце, а direct у називу фајла индикатор да се ради о фајлу који садржи директне секвенце.

Напомена: може се чинити да је овакав приступ непрактичан јер се издавају дупле информације, међутим аутор овакав избор оправдава

следећим чињеницама: када је у питању обрада неуређених секвенци, довољне су нам информације које се налазе у .load фајловима, међутим, уколико желимо да обрађујемо статистику везану за сваки протеин, то не можемо лако урадити из .stat фајла јер он не садржи id протеина за који су информације везане, док фајл који садржи свеобухватне информације то све садржи.

Налажење позиција неуређених региона

Да бисмо наставили са радом, део информација које је потребно обрадити обухвата дохватање и обраду информација о неуређеним регионима у оквиру протеина. Сваки протеин може имати један или више неуређених региона. Како су нам информације о њиховим почецима и крајевима од кључне важности, у овом поглављу бавићемо се управо издавањем ових региона из датих протеина. Информације које ћемо издвојити из сваког протеина су следеће:

1. позиција почетка неуређеног региона
2. позиција краја неуређеног региона
3. идентификатор региона и
4. део секвенце који је обухваћен овим регионом.

Да бисмо ово урадили потребно је покренути програм `extractRegions.ipynb`³ који пролази кроз преузети JSON фајл, извлачи информације од интереса и уписује их у фајл `regioni.txt`. Како на сајту базе DisProt постоји могућност да се преузме и .fasta фајл који садржи информације о свим неуређеним регионима, и он је преузет са циљем да се провере резултати написаног програма. За проверу кардиналности региона у .fasta фајлу, написан је програм `Main.java`, док се кардиналност скупа региона који смо ми добили може видети у поменутом `extractRegions.ipynb` фајлу, а може се погледати и број линија `regioni.txt` фајла. У свим наведеним случајевима кардиналност скупова је иста - 10.544 региона. Изглед добијеног фајла приложен је на Слици 12.

³ опет исто важи за покретање из терминала

```

DP00003r002 294-334 EHVIEMDVTSNGQALKEQSSKAKIVKNRWGRNVQISNT
DP00003r004 454-464 VYRNSRAQGGG
DP00004r001 134-170 LLGDFFRKSKEKIGKEFKRIVQRIKDFLRNLVPRTES
DP00004r002 134-170 LLGDFFRKSKEKIGKEFKRIVQRIKDFLRNLVPRTES
DP00004r004 134-170 LLGDFFRKSKEKIGKEFKRIVQRIKDFLRNLVPRTES
DP00004r005 150-162 FKRIVQRIKDFLR
DP00004r006 150-162 FKRIVQRIKDFLR
DP00005r001 1-107 MDAQTRRRERRAEKQAQWKAANPLLVGVSAPVNRPILSNRKPKSRVESALNPIDLTVLAEYHKQIESNLQRIERKNQRTWYSKPGERGITCSGRQIKIGKSJ
DP00005r004 1-107 MDAQTRRRERRAEKQAQWKAANPLLVGVSAPVNRPILSNRKPKSRVESALNPIDLTVLAEYHKQIESNLQRIERKNQRTWYSKPGERGITCSGRQIKIGKSJ
DP00005r005 1-107 MDAQTRRRERRAEKQAQWKAANPLLVGVSAPVNRPILSNRKPKSRVESALNPIDLTVLAEYHKQIESNLQRIERKNQRTWYSKPGERGITCSGRQIKIGKSJ
DP00005r006 1-107 MDAQTRRRERRAEKQAQWKAANPLLVGVSAPVNRPILSNRKPKSRVESALNPIDLTVLAEYHKQIESNLQRIERKNQRTWYSKPGERGITCSGRQIKIGKSJ
DP00005r007 1-107 MDAQTRRRERRAEKQAQWKAANPLLVGVSAPVNRPILSNRKPKSRVESALNPIDLTVLAEYHKQIESNLQRIERKNQRTWYSKPGERGITCSGRQIKIGKSJ
DP00005r008 1-107 MDAQTRRRERRAEKQAQWKAANPLLVGVSAPVNRPILSNRKPKSRVESALNPIDLTVLAEYHKQIESNLQRIERKNQRTWYSKPGERGITCSGRQIKIGKSJ
DP00005r009 1-107 MDAQTRRRERRAEKQAQWKAANPLLVGVSAPVNRPILSNRKPKSRVESALNPIDLTVLAEYHKQIESNLQRIERKNQRTWYSKPGERGITCSGRQIKIGKSJ
DP00005r010 1-107 MDAQTRRRERRAEKQAQWKAANPLLVGVSAPVNRPILSNRKPKSRVESALNPIDLTVLAEYHKQIESNLQRIERKNQRTWYSKPGERGITCSGRQIKIGKSJ
DP00005r011 1-107 MDAQTRRRERRAEKQAQWKAANPLLVGVSAPVNRPILSNRKPKSRVESALNPIDLTVLAEYHKQIESNLQRIERKNQRTWYSKPGERGITCSGRQIKIGKSJ
DP00005r012 1-22 MDAQTRRRERRAEKQAQWKAAN
DP00005r013 34-47 NRPILSLRKPKSR
DP00005r014 34-47 NRPILSLRKPKSR
DP00005r015 1-22 MDAQTRRRERRAEKQAQWKAAN
DP00005r016 1-22 MDAQTRRRERRAEKQAQWKAAN
DP00005r017 1-107 MDAQTRRRERRAEKQAQWKAANPLLVGVSAPVNRPILSNRKPKSRVESALNPIDLTVLAEYHKQIESNLQRIERKNQRTWYSKPGERGITCSGRQIKIGKSJ
DP00005r018 1-36 MDAQTRRRERRAEKQAQWKAANPLLVGVSAPVNR
DP00006r011 1-104 MGDVEKGKKIFVQKCAQCHTVEKGGKHKTGPNLHGLFGRKTGQAPGFTYTDANKNKGITWKEETLMEYLENPKKYIPGTMIFAGIKKTEREDLIAYLKKATN
DP00006r012 1-104 MGDVEKGKKIFVQKCAQCHTVEKGGKHKTGPNLHGLFGRKTGQAPGFTYTDANKNKGITWKEETLMEYLENPKKYIPGTMIFAGIKKTEREDLIAYLKKATN
DP00006r013 2-105 GDVEKGKKIFVQKCAQCHTVEKGGKHKTGPNLHGLFGRKTGQAPGFTYTDANKNKGITWKEETLMEYLENPKKYIPGTMIFAGIKKTEREDLIAYLKKATN
DP00006r014 2-105 GDVEKGKKIFVQKCAQCHTVEKGGKHKTGPNLHGLFGRKTGQAPGFTYTDANKNKGITWKEETLMEYLENPKKYIPGTMIFAGIKKTEREDLIAYLKKATN
DP00007r002 1-42 MPKRGGKGAVAEDGDELTERPEAKKSKTAACKNDKEAAGEGP
DP00007r006 1-36 MPKRGGKGAVAEDGDELTERPEAKKSKTAACKNDKE
DP00007r007 32-43 KNDKEAAGEGPA
DP00007r008 2-40 PKRGGKGAVAEDGDELTERPEAKKSKTAACKNDKEAAGE

```

Слика 12 Део извучених неуређених региона

Напомена: када бисмо посматрали само позиције почетка и краја неуређених региона, имали бисмо привид да постоје дупликати којих се треба отарасити. Међутим, како сви ови региони имају свој јединствен идентификатор, сви добијени резултати су задржани.

Обједињавање резултата

Да бисмо могли да применимо алгоритме за налажење правила придруживања, морамо прво наше резултате објединити у један фајл, како би сви потребни подаци били на једном месту. Пошто података о поновцима дужине 2 и више име преко четири милиона (види Слику 8), за потребе овог рада биће коришћени подаци о поновцима минималне дужине 3, којих има око 400000.

Међутим, како је и фајл са 400000 линија велики и захтева доста времена за обраду, приступ који је био коришћен током овог истраживања је следећи: покретањем програма `splitFile.java` фајл ће се, покретањем неколико кориснички дефинисаних нити (класа `FileThreadRunnable.java`) поделити на 8 мањих фајлова. Након тога, када је подела фајла завршена, потребно је покренути редом програме `append1.py`, `append2.py`, `append3.py`, `append4.py`, `append5.py`, `append6.py`, `append7.py` и `append8.py`. Када је и то урађено, над резултујућим `dataset.csv` фајлом потребно је покренути програм `cleanCSV.java` који ће

уклонити заглавља која се понављају. Сада коначно имамо документ над којим можемо да примењујемо узorkовање.

Анализа обележја

Као што можемо видети из резултујућих датотека, као обележја од посебног интереса издвојили смо:

1. ниво неуређености протеина, који би такође могао да утиче на чињеницу да ли у протеину има или нема понављајућих секвенци
2. индикатор пресека поновака и неуређених региона и
3. индикатор да ли је поновак у потпуности садржан у региону.

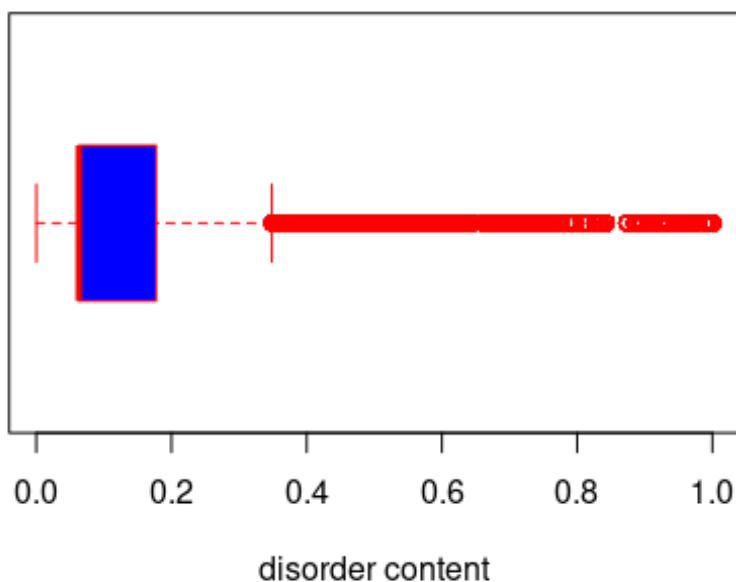
Како подататака за анализу има јако много, желели бисмо да извучемо неке најбитније статистике које би могле бити од помоћи са каснијом анализом, али би уједно могле и да нас усмере када је избор параметара за функције у питању. Да бисмо добили сумарне статистике ових обележја, потребно је покренути програм `summary.R` који ће извести тражене статистике и уписати их у фајл `summary.txt` (резултати су издвојени у посебан фајл ради постизања боље прегледности.)

Као што се и може видети из резултата, када је у питању ниво неуређености протеина, средња вредност на читавој популацији је веома мала, а вредност медијане још мања, што нас може навести на, помало неинтуитивну, претпоставку да ће се и поновци углавном налазити у протеинима који имају низак ниво неуређености. Пошто из приложених статистика не можемо видети како су тачно распрострањени подаци, направићемо по дијаграм.

```
data<-read.csv("datasetClean.csv")

b<-boxplot(data$disorder_content, xlab="disorder content",
main="Disorder content-direct repeats", col="blue", border="red",
horizontal = TRUE)
```

Disorder content-direct repeats



```
#donja i gornja granica
```

```
b$conf
```

```
##           [,1]
```

```
## [1,] 0.06251745
```

```
## [2,] 0.06278095
```

```
#broj autlajera
```

```
length(b$out)
```

```
## [1] 303778
```

```
#standardna devijacija
```

```
dev=sd(data$disorder_content)
```

```
dev
```

```
## [1] 0.2374755
```

Kao što se sa добијених дијаграма може видети, највећи део података заиста јесте сконцентрисан око средње вредности, међутим, стандардна девијација у оба случаја је прилично велика, што значи да подаци у просеку и нису толико близу средини колико би се у први мах помислило.

Но, упркос интересантности нивоа неуређености, много су нам занимљивије статистике које се тичу индикатора пресека и

садржавања. Пошто су све вредности или 0 или 1, вредности минимума и максимума ових колона није нам од значаја. Даље, из чињенице да је медијана 0 (за оба индикатора) можемо да закључимо да има више поновака који се не секу и не припадају неуређеним регионима. Но, то је било и очекивано, па не можемо да кажемо да нам је медијана донела неки помак. Са друге стране, ако анализирамо средње вредности, ту долазимо до напретка. Можемо уочити да су средње вредности оба индикатора веома ниске, што нас упућује на то да су подаци који ће бити корисни веома ретки. Такође, чињеница да има више региона који садрже цео поволак него оних који се секу са неким поновком може бити веома интересантна.

Уколико желимо да, ради боље прегледности, видимо и проценат података који има неку од две вредности, то можемо урадити покретањем следећег кода.

```
data<-read.csv("datasetClean.csv")
table1<-table(data$intersect_ind)
table2<-table(data$contains_ind)

prop.table(table1)

##
##           0           1
## 0.94222637 0.05777363

prop.table(table2)

##
##           0           1
## 0.87982 0.12018
```

Из приложених табела можемо уочити да су вредности од интереса када су оба индикатора у питању веома ниска, што нас упућује на то да ћемо код неких техника узорковања морати да узмемо већи проценат јединки у узорак да бисмо имали колико толико корисне резултате.

Различити планови узорковања и оцењивања

Овај рад ће обухватати неколико вероватносних метода: прост случајан узорак са и без понављања, стратификован узорак, кластеровање и систематски узорак. Невероватносно узорковање неће бити разматрано због разлога наведених у уводном делу.

Напомена: сви програми писани су за конзолно извршавање.

Прост случајан узорак без понављања

Наша разматрања кренућемо од најједноставнијег облика узорковања - *простог случајаног узорка*. Овај начин узорковања подразумева да је вероватноћа да се јединка из популације нађе у узорку иста за сваку јединку. Уколико узорковање вршимо тако да се једна јединка у узорку може појавити највише једанпут, говоримо о *простом случајном узорку без понављања*. Ако из популације обима N треба да извучемо узорак обима n , онда је вероватноћа да јединка буде изабрана у узорак $\frac{1}{N}$.

Међутим, због величине скупа који представља нашу популацију може се десити да је вредност израза $\frac{1}{N}$ веома велика, па ће онда, последично, вероватноћа избора јединке бити једнака 0, што је ситуација коју желимо да избегнемо, те стога морамо пажљиво бирати параметар n . Да би вероватноћа одабира јединке била већа од 0, морамо одабрати или веома мали или веома велики број јединки. Како смо видели у уводном делу о узорковању, лоше је изабрати премали узорак, те ћемо морати да бирамо веома велики проценат популације. Самим тим, природно је да очекујемо да ће оцене које добијемо веома добро апроксимирати стварне вредности.

За израчунавање оцена тотала коришћењем методе простог случајног узорка, потребно је покренути програме `psuBezPonavljanja.R` (оцењујемо ниво неуређености протеина), `psuBezPonavljanjaSadrzi.R` (који оцењује вредност индикатора протеина који садрже поновак), те `psuBezPonPresek.R` (за оцену вредности тотала индикатора пресека). Сваки од ових програма ради узорковање за по 4 вредности параметра n , за сваку од тих вредности врши оцењивање тотала, дисперзије оцене, оцену дисперзије оцене и, на крају, исписује добијене податке у фајл. Резултујући документи названи су редом `psuBPneuredjenost.txt`, `psuBPsadrzi.txt` и `psuBPpresek.txt`. Ради лакше анализе добијених резултата, користићемо следећу табелу у којој се налазе обједињени резултати записани у свим фајловима.

Назив фајла	n	Оцењена вредност тотала	Стварна вредност тотала	Оцена дисперзије оцене
psuBPneuredj enost.txt	1884800	325890.4	325890.2	1.240694
psuBPneuredj enost.txt	1884780	325890	325890.2	2.36862
psuBPneuredj enost.txt	1884770	325889.5	325890.2	2.932581
psuBPneuredj enost.txt	1884760	325890.3	325890.2	3.49572
psuBPsadrzi.t xt	1884800	226519.6	226518	2.326253
psuBPsadrzi.t xt	1884780	226518	226518	4.441048
psuBPsadrzi.t xt	1884770	226515.2	226518	5.498411
psuBPsadrzi.t xt	1884760	226515.5	226518	6.555837
psuBPpresek. txt	1884800	108892.3	108893	1.197595
psuBPpresek. txt	1884780	108890.4	108893	2.286306
psuBPpresek. txt	1884770	108891	108893	2.830694
psuBPpresek. txt	1884760	108891.6	108893	3.375093

Као што смо и очекивали, све оцене су веома блиске стварној вредности тотала. Ако мало боље погледамо табелу, уочићемо да се свуда дешава да оцене тотала које су добијене на узорку мањег обима буду боље од оцена добијених за обимније узорке, чак се и дешава да узорак најмањег обима даје најбољу процену. Да ли то значи да су те оцене заиста боље? Да бисмо добили одговор на то питање довољно је да погледамо колону која садржи оцене дисперзије оцене и видимо да иако су мање обимни узорци имали приближнију вредност оцене, имали су и већу дисперзију, те можемо закључити да су ипак, обимнији узорци поузданији.

Прост случајан узорак са понављањем

У претходном делу илустровали смо случајан узорак без понављања. Сада ћемо посватити пажњу другој врсти простог случајног узорака - *узорковање са понављањем*. Као што и само име каже ова техника узорковања подразумева да се свака јединка може наћи 0 или више пута у узорку. Ако је потребно да из популације обима N извучемо узорак обима n , онда је вероватноћа да јединка буде изабрана у узорак $\frac{1}{N^n}$.

Но, и овде наилазимо на проблем - пошто је обим популације веома велики, то значи да обим узорака који бирамо мора бити веома мали да би вероватноћа избора била већа од нуле. Стога, овај начин узорковања неће бити анализиран, већ ће служити да покажемо колико је пропорционалност битна особина. Демонстрација узорковања са понављањем може се видети покретањем програма `rsuSaPonavljanjem.R`, где су обједињене оцене за сва 3 обележја за вредности $n=10$ и $n=15$. Као и у случају са простим случајним узорковањем без понављања, сви резултати уписани су у документ `rsuSP.txt`. И опет ћемо ради лакше прегледности извући потребне податке у табелу:

n	Обележје које се оцењује	Оцењена вредност тотала	Стварна вредност тотала
10	неуређеност протеина	135975.3	325890.2
15	неуређеност протеина	125234.6	325890.2
10	индикатор пресека	0	108893
15	индикатор пресека	0	108893
10	индикатор садржања	0	226518
15	индикатор садржања	0	226518

Као што се из добијених резултата може видети, оцене су веома лоше, што је и било очекивано с обзиром на обим узорака који је

извучен.

Стратификован узорак

Следећи тип вероватносног узорка који ћемо размотрити јесте *стратификовани узорак*. Овакав начин узорковања подразумева да се почетни скуп подели у неколико стратума (скупова) тако да су јединке које припадају једном стратуму сличне (у териминима вредности обележја), док су јединке које се налазе у различитим стратумима међусобно различите. Потребно је нагласити да је неопходно да стратуми буду међусобно дисјунктни, али и да покривају целу популацију.

Подела индикатора по стратумима је веома интуитивна, па ћемо и поделити популацију у 2 стратума у односу на вредност индикатора. Са друге стране, подела јединки на основу нивоа неуређености протеина можда није на први поглед очигледна као у претходном случају због великог броја различитих вредности које ово обележје може да узме. За потребе овог рада популација ће бити подељена у 5 стратума тако да први стратум обухвата јединке чија је вредност неуређености $[0, 0.2)$, други стратум ће обухватати вредности из скупа $[0.2, 0.4)$, трећи из интервала $[0.4, 0.6)$, четврти из $[0.6, 0.8)$, док ће остале вредности из интервала $[0.8, 1]$ припадати последњем, петом стратуму.

Даље, када смо поделили популацију по стратумима, методом простог случајно узорка без понављања из сваког стратума извучимо одређени број јединки. У овом раду, из сваког стратума извучена је половина јединки које се у њему налазе. У случају оцено нивоа неуређености протеина, извршена је и још једна подела тако да је укупан број извучених јединки приближно једнак броју извучених јединки код простог случајног узорка без понављања како бисмо лакше упоредили оцено.

Примена стратификованог узорка демонстрирана је у програмима `stratifikovaniBezPonavljanjaSadrzi.R`, `stratifikovaniBezPonavljanja.R` и `stratifikovaniBezPonavljanjaPresek.R`. Сваки програм обухвата прављење стратума (на начин описан изнад), извлачење узорка из сваког стратума (опет, на начин који је описан горе), рачунање оцено тотала и оцено дисперзије оцено тотала, те испис добијених резултата у фајлове `StratifikovaniSadrzi.txt`, `StratifikovaniNeuredjenost.txt` и `StratifikovaniPresek.txt` респективно.

Као и до сад, нудимо табеларни приказ добијених резултата:

Назив фајла	n	Оцењена вредност тотала	Стварна вредност тотала	Оцена дисперзије оцене
Stratifikovani Sadrzi.txt	942411	226518	226518	0
Stratifikovani Presek.txt	942410	108893	108893	0
Stratifikovani Neuredjenost .txt	942412	325904	325890.2	3427.94
Stratifikovani Neuredjenost .txt	1413416	325846.1	329890.2	1442.739
Stratifikovani Neuredjenost .txt	1696340	325870.5	325890.2	380.8346
Stratifikovani Neuredjenost .txt	1884715	325889.2	329890.2	0.2522913

Из табеле можемо видети да се стратификовано узорковање показало веома добро када је оцена оба индикатора у питању (прецизна оцена иако је у узорак ушло само 50% јединки читаве популације), док је оцена нивоа неуређености протеина донела мало лошије резултате за узорке мањег обима, али изузетно добре резултате за веома обимне узорке. Ако упоредимо ове оцене са резултатима добијеним простим случајним узорковањем, видећемо да стратификовани узорак даје боље резултате за свако од посматраних обележја.

Узорковање кластеровањем

Сада ћемо демонстрирати технику узорковања која је сушта супротност стратификованом узорку - *кластеровање*. Идеја овог начина узорковања јесте да се све јединке популације поделе у *кластере*, али тако да су јединке које се налазе у оквиру једног кластера међусобно различите, док су сви кластери међусобно слични, тј. идеја је да сваки кластер што верније осликава целу популацију.

Зато је прво неопходно поделити јединке у кластере. За потребе овог рада то ћемо учинити на следећи начин: прво ћемо све јединке распоредити у стратуме као што смо урадили и у претходном случају.

Затим је те јединке потребно распоредити у кластере тако да сваки кластер што верније осликава популацију. Размотрићемо 2 начина за поделу јединки у кластере: прва опција је да кластери буду исте величине, док је друга опција да јединке поделимо тако да се 40% свих јединки налази у првом кластеру, 30% у другом, 20% у трећем, а по 5% у четвртном и петом кластеру⁴ (овај број кластера узет је насумично, да би се поклапао са бројем стратума). Након поделе јединки у кластере, потребно је изабрати неки подскуп кластера и над њиме вршити израчунавања.

Пример кластеровања и рачунања оцена овом техником узорковања демонстриран је у програмима `klasterovanje.R`, `klasterovanjeSadrzi.R` и `klasterovanjePresek.R`, док су резултати извршавања уписани редом у фајлове `klasterovanjeNeuredjenost.txt`, `klasterovanjeSadrzi.txt` и `klasterovanjePresek.txt`.

Табелирани резултати израчунавања могу се наћи у табели испод:

Назив фајла	n	Оцењена вредност тотала	Стварна вредност тотала	Назив оцене	Оцена дисперзиј е оцене
klasterovanjeSadrzi.txt	1130893	226518.3	226518	псу без понављања	1.111111
klasterovanjePresek.txt	1130893	108893.3	108893	псу без понављања	1.111111
klasterovanjeSadrzi.txt	1036652	275679	226518	Хорвиц-Томпсон	NaN
klasterovanjePresek.txt	1036652	132529.1	108893	Хорвиц-Томпсон	NaN
klasterovanjeSadrzi.txt	1036652	99820	226518	псу без понављања	27080843 53
klasterovanje	1036652	207641.7	108893	псу без	62581200

⁴ у тексту стоји проценат јединки по кластеру у односу на укупан број јединки у популацији, али је први кластер прављен тако да садржи по 40% јединки из сваког стратума како би кластер што боље осликавао целокупну популацију. Аналогно важи и за остале кластере

Назив фајла	n	Оцењена вредност тотала	Стварна вредност тотала	Назив оцене	Оцена дисперзиј е оцене
njePresek. txt				понављањ а	3
klasterova njeNeuredj enost.txt	1036652	402370.3	325890.2	Хорвиц- Томпсон	NaN
klasterova njeNeuredj enost.txt	1036652	306408.3	325890.2	псу без понављањ а	62190250 48
klasterova njeNeuredj enost.txt	1130893	320552.0 96585091	325890.2	псу без понављањ а	37475457

Као што можемо видети из табеле, резултати кластеровања су или веома добри, или веома лоши. У случајевима када смо популацију делили на кластере једнаких величина, оцене за оба индикатора су се показале као веома добре, док је оцена нивоа неуређености лошија. Са друге стране, у случајевима када су кластери неједнаких величина, све оцене које су добијене су веома лоше.

Систематски узорак

Демонстрираћемо још један веома једноставан начин узорковања - *систематски узорак*. Основна идеја овог узорковања подразумева да се из популације обима N извлачи узорак обима n . Уколико $k = \frac{N}{n}$ није природан број, онда се он заокружи. Следећи корак јесте насумично одабрати један број из интервала $[1, k]$ и узима се јединка која се налази на тој позицији у узорку. Након ње, у узорак се узима свака k -та јединка. Као што се да претпоставити, резултати ће умногоме зависити од распореда јединки у популацији.

За потребе овог рада биће коришћене оне вредности n које су делиоци броја N , ради лакшег рачуна. Настављамо као и до сад, оцене нивоа неуређености протеина и дисперзије тих оцена могу се добити покретањем програма `sistematski.R`, док се оцене индикатора могу добити извршавањем програма `sistematskiSadrzi.R` и `sistematskiPresek.R` редом. Резултати извршавања могу се наћи у документима `sistematskiNeuredjenost.txt`, `sistematskiSadrzi.txt` и `sistematskiPresek.txt`. И опет, у наставку се може наћи табеларни приказ добијених резултата:

Назив фајла	n	Оцењена вредност средине	Стварна вредност средине	Дисперзија оцене
sistematskiNeuredjenost.txt	314137	0.1729133	0.1729023	0.00000000003738765
sistematskiNeuredjenost.txt	628274	0.1729045	0.1792023	0.000000000001635157
sistematskiPresetek.txt	314137	0.06464695	0.05777363	0.00005871577
sistematskiPresetek.txt	628274	0.05720275	0.05777363	0.0000001670036
sistematskiSadrzi.txt	314137	0.1314076	0.12018	0.0000829115
sistematskiSadrzi.txt	628274	0.1220582	0.12018	0.00000227279

Као што се види из табеле, оцене које смо добили су веома добре, иако је обим узорка био доста мањи него у претходним случајевима.

Анализа резултата и закључак

Студије се спроводе на узорцима јер је обично немогуће проучити целу популацију. Закључци извучени из узорака имају за циљ да се генерализују на популацију, те је зато веома битно да узорак буде што репрезентативнији.

У оквиру овог рада приказане су различите технике вероватносног узорковања: прост случајан узорак са и без понављања, стратификован узорак, кластеровање и систематски узорак.

Прво је демонстрирано просто случајно узорковање без понављања. Главни проблем код ове технике био је тај што је она изискивала узорак великог обима да би могла бити примењена, те смо могли унапред очекивати да ће оцене бити веома добре и блиске стварним вредностима. То се и десило. На основу добијених оцена за ниво неуређености протеина можемо закључити да је $n=1884760$ довољно обиман узорак (можда је и мањи узорак такође довољно добар, али то нисмо могли проверити), сви резултати после тога веома мало се мењају, а не поправљају драстично вредност оцене. Међутим, оно што се мења са повећањем узорка јесте оцена дисперзије оцене, као и сама дисперзија оцене. Што је узорак обимнији то је дисперзија мања, а

нижа вредност дисперзије значи већу прецизност. Са друге стране, када су оцене индикатора у питању, већи узорак заиста јесте дао најбољу оцену тотала, а и најмање вредности дисперзије.

Затим је прост случајан узорак са понављањем коришћен како би се доказала тврдња да премали узорак није добар, те стога резултати ове технике узорковања неће бити даље анализирани.

Након простог случајног узорковања упознали смо се са стратификованим узорком. Он се показао као веома добар када је оцена индикатора у питању јер је оцена тотала добијена на основу само половине популације одговарала стварној вредности тотала. Одговарајућа оцена дисперзије оцене за оба индикатора је 0. Насупрот томе, оцена нивоа неуређености протеина доста је осциловала. Све оцене овог обележја истог су реда величине као и стварна вредност тотала, чак су и веома блиске стварној вредности с обзиром из ког обима су добијени. Но, оно што разликује ове оцене су вредности оцене њихових дисперзија. Показало се да најмањи узорак има доста велику оцену дисперзије, док се та вредност смањује са повећањем узорка. На крају, за најобимнији узорак, вредност оцене дисперзије оцене пада испод 1, што ову технику узорковања чини најбољом до сад.

Потом је описано узорковање кластеровањем. Када су јединке биле подељене у једнаком обиму у кластере оцене оба индикатора су биле веома добре, чак су и оцене њихових дисперзија биле тек нешто мало изнад 1, док је оцена нивоа неуређености била досад најлошија у поређењу са свим оценама. Међутим, када су кластери били неједнаке величине, све оцене које су добијене су веома лоше, а самим тим и њихове дисперзије⁵. Постоји могућност да је узрок оваквих резултата лоша подела јединки у кластере.

На крају, испробан је систематски узорак. Можемо видети да су све оцене које су добијене систематским узорком веома добре, као и оцене њихових дисперзија, које су веома блиске 0. То нас упућује на чињеницу да су јединке лепо распоређене у оквиру популације. Када имамо ову информацију, можемо закључити да проблем код кластеровања заиста јесте био у начину распоређивања јединки у кластере, те да би можда боља идеја била да су се јединке распоређивале на основу систематског узорка, него на основу стратума.

У табелама испод упоредићемо технике узорковања:

⁵ израчунате су само дисперзије оцене добијене простим случајним узорковањем јер су оцене добијене Хорвиц-Томпсоновим методом још лошије од њих

Ранг	Поређење узорковања индикатора	техника за оцену
1.	стратификовани узорак	
2.	систематски узорак	
3.	кластеровање са једнаким величинама кластера	
4.	прост случајан узорак без понављања	
5.	прост случајан узорак са понављањем	
6.	кластеровање са неједнаким величинама кластера	
Ранг	Поређење узорковања индикатора	техника за оцену
1.	систематски узорак	
2.	стратификовани узорак	
3.	прост случајан узорак без понављања	
4.	кластеровање са једнаким величинама кластера	
5.	прост случајан узорак са понављањем	
6.	кластеровање са неједнаким величинама кластера	

Као што можемо видети, најбоље оцене добијене су стратификованим и систематским узорком, средње добре оцене добијене су простим случајним узорком без понављања и кластеровањем са једнаким величинама кластера, док су најлошије оцене добијене кластеровањем са неједнаким величинама кластера и простим случајним узорком без понављања.

Литература

- 1 Бабу Мадан (2016), "Допринос неуређених секвенци функционисању протеина, комплексности ћелије и људским болестима", може се наћи на линку. (<https://www.ncbi.nlm.nih.gov/pmc/articles/PMC5095923/>)

- 2 Уверски Владимир (2019), “Неуређени региони и њихова”мистериозна” (мета)физика”, (<https://www.frontiersin.org/articles/10.3389/fphy.2019.00010/full>)
- 3 База неуређених секвенци DisProt, (може се наћи на: <https://disprot.org/>)
- 4 Документација за Python Biopython, аpyori
- 5 Онлајн документација везана за .fasta фајлове, која се може пронаћи на линку <https://www.ncbi.nlm.nih.gov/WebSub/html/help/fasta.html>
- 6 Пратећа документација за коришћење програма StatRepeats
- 7 Званични сајт за језик Python, где је могуће наћи и туторијале и инсталације уколико је потребно, <https://www.python.org/>
- 8 Званични сајт компаније Oracle, где је могуће наћи информације о програмском језику Java, као и инсталације уколико је то потребно, <https://www.oracle.com/in/java/>
- 9 Званични сајт за програмски језик Java, где је могуће наћи инсталацију Јава виртуелне машине уколико је то потребно, <https://www.java.com/en/download/manual.jsp>
- 10 Методе узорковања, <https://edu.gcfglobal.org/en/statistics-basic-concepts/sampling-methods/1/>
- 11 Материјали са вежби и предавања
- 12 Андраде, Читарањан, “Величина узорка и његова важност у истраживању”, <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6970301/>
- 13 Фабер, Хорхе, “Како величина узорка утиче на исход истраживања”, <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4296634/>
- 14 Чланак о дисперзији у статистици, <https://www.wallstreetmojo.com/dispersion/>