

Programma python

Scrivere un programma python che:

- prenda in ingresso:
 - un nome di file che contiene dei dati di pazienti con cancro al seno, in forma tabellare
 - una classificazione (e.g. Pam50Subtype)
 - un attributo (e.g. stage)
 - l'indicazione se l'attributo è continuo o discreto
- stampi per ogni classe della classificazione, se l'attributo è continuo il suo valore medio (tra i soggetti della classe), altrimenti il dizionario di coppie (valore, numero di soggetti con quel valore)

Esempio esecuzione

Attributo continuo

```
> python compute_stats.py
Inserire nome file: breast_cancer.txt
Inserire classificazione: Treatment
Inserire attributo: size
attributo continuo?(S/N): S
Treatment      size
RT              20.326531
NONE            23.784314
HT/RT           25.483395
HT              25.021739
CT/HT           22.692308
CT              23.470588
CT/HT/RT        35.000000
CT/RT           31.621429
```

Esempio esecuzione

Attributo discreto

```
> python compute_stats.py
Inserire nome file: breast_cancer.txt
Inserire classificazione: Pam50Subtype
Inserire attributo: stage
attributo continuo?(S/N): N
Pam50Subtype   stage
Basal          1->18 0->29 3->7 2->34
Her2           1->11 0->28 3->10 2->18
Normal         1->11 0->13 3->3 2->25 4->2
LumB           1->51 0->67 3->11 2->81 4->5
LumA           1->94 0->145 3->12 2->122 4->2
```

Programma python: suggerimento

Si possono implementare 4 funzioni separate:

1. Una che legga il file, individui le colonne corrispondenti a classificazione ed attributo richiesti, e restituisca un dizionario classe → lista di valori dell'attributo dei soggetti con tale classe

2. una che calcola la media di un elenco di valori (continui)
3. una che calcola il dizionario di occorrenze di un elenco di valori (discreti)
4. una che realizzi il programma richiesto usando le funzioni di cui sopra

Shell: esercizio #1

Dato il file `breast_cancer.txt`, stampare a schermo l'ID dei 5 pazienti diagnosticati in più tarda età che non hanno mutazioni alla proteina P53.

Nota: la colonna `P53_mutation_status` vale WT, wildtype, in questo caso.

Risultato atteso

```
MB-0203
MB-0322
MB-5193
MB-0454
MB-0319
```

Shell: esercizio #2

Dato il file `sequences.fasta` ed i due seguenti "motivi":

- Da una a quattro **lisine** (K); seguite da una **lisina** o da una **arginina** (R); un aa. qualunque; una **lisina** o **arginina**; un aa. qualunque.
- Un amminoacido **sconosciuto** (X).

Calcolare quante sequenze contengono il primo motivo, quante il secondo, e quante entrambi, cercando di minimizzare il numero di invocazioni di `grep`.

Risultato atteso

616, 7, 5.

Modalità di esecuzione e consegna

1. Avviare la macchina in modalità ESAME
2. Autenticarsi con nome utente `sci-esame` e password fornita dal docente
3. Il testo del compito ed i file necessari si trovano in una cartella `Testo` sul Desktop
4. Realizzare il programma python come file `programma.py` e scrivere gli esercizi da linea di comando in un file di testo `linea_di_comando.txt`
5. Creare sul Desktop una cartella con `nome_cognome` e metterci i due file realizzati.
6. Eseguire il logout ma NON spegnere la macchina