Davide	*Ghiotto*	*1236660*

Midterm test No. 3

22 / 12 / 2020

Questions

1. Paste below your assignment ID.
   21

2. Download human GO annotations (GAF format) from GOA at EBI (ftp.ebi.ac.uk/pub/databases/GO/goa/HUMAN/).

   For these answers I used the code provided in "example.py".
   For every accession if it was present in my dataset, I counted the number of annotations and the terms as belonging to my set. In the other case the annotations/proteins/terms were part of the rest.
   To calculate the leaf terms associated with my list of proteins I just extracted all the leaf terms and filtered out the ones that weren't inside my terms set.

   a. How many annotations are associated with your list of proteins?
      11992

   b. How many annotations are associated with the rest of human proteins?
      279318

   c. How many different (unique) terms are associated with your proteins?
      7434

   d. How many different (unique) terms are associated with the rest of human proteins?
      21763

   e. How many leaf terms are associated with your proteins?
      1828

3. Which are the most abundant GO terms considering your list of proteins? Provide no more than 10 terms separately for each namespace (sub-ontology).
   For this answer I used the code provided in the script "parse_goa,py": basically the code builds a map where for every term is associated the count of numbers of times is repeated.
   Then from this map is extracted an array, sorted by the value (decreasing) and foreach root the code loops over the first 10 elements.

   - biological_process
     - GO:0071222 150 cellular response to lipopolysaccharide
     - GO:0032496 138 response to lipopolysaccharide
     - GO:0006954 85 inflammatory response
     - GO:0045944 55 positive regulation of transcription by RNA polymerase II
     - GO:0019221 46 cytokine-mediated signaling pathway
     - GO:0045087 43 innate immune response
     - GO:0007165 43 signal transduction
     - GO:0007186 40 G protein-coupled receptor signaling pathway
     - GO:0010628 37 positive regulation of gene expression
     - GO:0006915 37 apoptotic process

   - molecular_function
     - GO:0005515 232 protein binding
     - GO:0042802 50 identical protein binding
     - GO:0005524 31 ATP binding
     - GO:0019899 26 enzyme binding
     - GO:0046872 25 metal ion binding
     - GO:0005125 24 cytokine activity
     - GO:0042803 22 protein homodimerization activity
     - GO:0000978 21 RNA polymerase II cis-regulatory region sequence-specific DNA binding
     - GO:0044877 20 protein-containing complex binding
     - GO:0008270 19 zinc ion binding

   - cellular_component
     - GO:0005737 117 cytoplasm
     - GO:0005886 113 plasma membrane
     - GO:0005829 102 cytosol
     - GO:0005634 94 nucleus
     - GO:0005615 87 extracellular space
     - GO:0005576 79 extracellular region
     - GO:0005654 70 nucleoplasm
     - GO:0070062 50 extracellular exosome
     - GO:0005887 43 integral component of plasma membrane
     - GO:0009986 39 cell surface

4. Which are the most abundant GO terms considering your list of proteins after integrating ancestors terms? Provide no more than 10 terms separately for each namespace (sub-ontology). **This answer follows the same procedure of the 3th with the difference than this time it sums also the number of ancestors inside the terms count map.**
    - biological_process
        - GO:0033993 283 response to lipid
        - GO:0009607 283 response to biotic stimulus
        - GO:0009605 283 response to external stimulus
        - GO:0050896 283 response to stimulus
        - GO:0010033 283 response to organic substance
        - GO:0043207 283 response to external biotic stimulus
        - GO:0002237 283 response to molecule of bacterial origin
        - GO:1901700 283 response to oxygen-containing compound
        - GO:0042221 283 response to chemical
        - GO:0008150 283 biological_process

    - molecular_function
        - GO:0003674 281 molecular_function
        - GO:0005488 272 binding
        - GO:0005515 261 protein binding
        - GO:0043167 111 ion binding
        - GO:0097159 100 organic cyclic compound binding
        - GO:0003824 99 catalytic activity
        - GO:1901363 96 heterocyclic compound binding
        - GO:0005102 90 signaling receptor binding
        - GO:0098772 85 molecular function regulator
        - GO:0019899 82 enzyme binding

    - cellular_component
        - GO:0005575 283 cellular_component
        - GO:0110165 281 cellular anatomical entity
        - GO:0043226 190 organelle
        - GO:0043227 180 membrane-bounded organelle
        - GO:0043229 163 intracellular organelle
        - GO:0005622 163 intracellular anatomical structure
        - GO:0043231 153 intracellular membrane-bounded organelle
        - GO:0016020 151 membrane
        - GO:0005737 120 cytoplasm
        - GO:0005886 113 plasma membrane

5. For each term GO_i build a confusion matrix as defined below and calculate the "fold increase" within your set of proteins compared to the rest of human proteins.
   Hint: The fold increase can be calculated dividing the ratio *having-/not-having the property* of the *selected* with the ratio *having-/not-having* of the *not selected*.

	Having the property	**Not having the property**
Selected	No. proteins with GO_i in your set	No. proteins without GO_i in your set
Not selected	No. proteins with GO_i in the rest of human proteins	No. proteins without GO_i in the rest of human proteins

For these answers I used the code at the end of "exercise.py" where it calculates exactly the confusion matrix reported above. Than in order to answer question "a" I just counted the terms with fold-increase greater than 1.
For the second I just loop over the first 10 elements ordered decreasingly by fold-increase (for each subontology).

a. How many terms have a fold increase larger than 1.
   7279

b. Report terms with the largest fold increase, 10 terms for each namespace (sub-ontology).
   - biological_process
     - GO:0032496 fold increase 19379.236749116608
     - GO:0071222 fold increase 10303.749116607774
     - GO:0071219 fold increase 1160.0247349823321
     - GO:0002237 fold increase 1019.9598289008742
     - GO:0009635 fold increase 409.42049469964667
     - GO:0018119 fold increase 409.42049469964667
     - GO:0017014 fold increase 409.42049469964667
     - GO:0032497 fold increase 341.1837455830389
     - GO:0071216 fold increase 316.3703822679087
     - GO:0071223 fold increase 307.06537102473493

   - molecular_function
     - GO:0000048 fold increase 409.42049469964667
     - GO:0048248 fold increase 409.42049469964667
     - GO:0035662 fold increase 409.42049469964667
     - GO:0102953 fold increase 341.1837455830389
     - GO:0103068 fold increase 341.1837455830389
     - GO:0034617 fold increase 341.1837455830389
     - GO:0070891 fold increase 341.1837455830389
     - GO:0004517 fold increase 272.9469964664311
     - GO:0002951 fold increase 204.71024734982333
     - GO:0004949 fold increase 204.71024734982333

- cellular_component
    - GO:0035976 fold increase 272.9469964664311
    - GO:0043514 fold increase 204.71024734982333
    - GO:0046696 fold increase 136.47349823321554
    - GO:0002096 fold increase 136.47349823321554
    - GO:1990008 fold increase 136.47349823321554
    - GO:1990005 fold increase 136.47349823321554
    - GO:0099013 fold increase 136.47349823321554
    - GO:0034592 fold increase 136.47349823321554
    - GO:0034466 fold increase 136.47349823321554
    - GO:0098898 fold increase 136.47349823321554

6. For each confusion matrix generated above, calculate the enrichment with Fisher's exact test.
   Hint: You can use https://pypi.org/project/fisher/
   For these answers I just install the library and passed the same data as the confusion matrix defined in the previous question. The value we are interested in is the right-tail p-value that identifies the terms with extraordinary presence of the property over the entire collection of terms. For question "b" was just about printing out also the p-values for each of the best terms (ordered by fold-increase) of the previous question.

    a. Which P-value between the left- and right-tail tells which are the enriched terms in your set of proteins?
    Hint: Make a comparison with terms with high "fold increase" calculated above if you are not sure.

    The enriched terms are the ones with low right tail values (usually is <0.05)

    b. Report enriched terms in the selected set and the corresponding P-values (left-/right-/two-tails).
    - molecular_function
        - GO:0000048 p-values: left 1.0 - right: 6.746099091595784e-11 - two tail: 1.380004214544527e-08
        - GO:0048248 p-values: left 1.0 - right: 6.746099091595784e-11 - two tail: 1.380004214544527e-08
        - GO:0035662 p-values: left 1.0 - right: 6.746099091595784e-11 - two tail: 1.380004214544527e-08
        - GO:0102953 p-values: left 0.9999999999511309 - right: 3.923173346728469e-09 - two tail: 6.692032603651698e-07
        - GO:0103068 p-values: left 0.9999999999511309 - right: 3.923173346728469e-09 - two tail: 6.692032603651698e-07
        - GO:0034617 p-values: left 0.9999999999511309 - right: 3.923173346728469e-09 - two tail: 6.692032603651698e-07
        - GO:0070891 p-values: left 0.9999999999511309 - right: 3.923173346728469e-09 - two tail: 6.692032603651698e-07
        - GO:0004517 p-values: left 0.9999999993652802 - right: 2.2260320195718512e-07 - two tail: 2.2260320195718512e-07
        - GO:0002951 p-values: left 0.9999999555752237 - right: 1.2169999886769644e-05 - two tail: 1.2169999886769644e-05
        - GO:0004949 p-values: left 0.9999999555752237 - right: 1.2169999886769644e-05 - two tail: 1.2169999886769644e-05

- cellular_component
    - GO:0035976 p-values: left 0.9999999993652802 - right: 2.2260320195718512e-07 - two tail: 2.2260320195718512e-07
    - GO:0043514 p-values: left 0.9999999555752237 - right: 1.2169999886769644e-05 - two tail: 1.2169999886769644e-05
    - GO:0046696 p-values: left 0.9999999961518344 - right: 6.59963259172205e-07 - two tail: 6.59963259172205e-07
    - GO:0002096 p-values: left 0.9999969560068148 - right: 0.0006262181492702775 - two tail: 0.0006262181492702775
    - GO:1990008 p-values: left 0.9999969560068148 - right: 0.0006262181492702775 - two tail: 0.0006262181492702775
    - GO:1990005 p-values: left 0.9999969560068148 - right: 0.0006262181492702775 - two tail: 0.0006262181492702775
    - GO:0034466 p-values: left 0.9999969560068148 - right: 0.0006262181492702775 - two tail: 0.0006262181492702775
    - GO:0034592 p-values: left 0.9999969560068148 - right: 0.0006262181492702775 - two tail: 0.0006262181492702775
    - GO:0099013 p-values: left 0.9999969560068148 - right: 0.0006262181492702775 - two tail: 0.0006262181492702775
    - GO:0098898 p-values: left 0.9999969560068148 - right: 0.0006262181492702775 - two tail: 0.0006262181492702775

- biological_process
    - GO:0032496 p-values: left 0.9999999999891656 - right: 0.0 - two tail: 7.229847760928413e-07
    - GO:0071222 p-values: left 0.99999999999063 - right: 1.3716963647540804e-261 - two tail: 2.5110414852648167e-07
    - GO:0071219 p-values: left 0.9999999999643543 - right: 6.205485312033692e-253 - two tail: 6.335232225750574e-07
    - GO:0002237 p-values: left 0.9999999999501292 - right: 0.0 - two tail: 6.515191656258862e-07
    - GO:0009635 p-values: left 1.0 - right: 6.746099091595784e-11 - two tail: 1.380004214544527e-08
    - GO:0017014 p-values: left 1.0 - right: 6.746099091595784e-11 - two tail: 1.380004214544527e-08
    - GO:0018119 p-values: left 1.0 - right: 6.746099091595784e-11 - two tail: 1.380004214544527e-08
    - GO:0032497 p-values: left 0.9999999999511309 - right: 3.923173346728469e-09 - two tail: 6.692032603651698e-07
    - GO:0071216 p-values: left 0.9999999999637004 - right: 9.490035321168261e-231 - two tail: 8.373958328490208e-07
    - GO:0071223 p-values: left 1.0 - right: 1.7095071472497441e-15 - two tail: 3.04106914067791e-07