Progetto d'esame per il corso di Bioinformatica: Predizione della funzione delle proteine con metodi di Machine Learning

L'obiettivo del progetto e' di predire la funzione delle proteine di *Drosophila melanogaster* (moscerino della frutta, organismo modello per gli insetti) ed opzionalmente di *Homo sapiens* (uomo), utilizzando metodi di machine learning (ML).

1 Attività del progetto

Ogni gruppo (massimo 2-3 studenti per gruppo) svolga le seguenti attività:

1. Scelga almeno due metodi di machine learning fra:

- Percettrone lineare
- Percettrone multi-strato (MLP)
- Support Vector Machine (SVM)
- Albero di decisione
- K-nearest-neighbour
- Regressione Logistica
- Naive Bayes
- Bagging
- Boosting
- Random Forests
- Metodi network-based semi-supervisionati (ad es: Label Propagation, COSNet, RANKS [4, 12])
- Altri metodi di ML (da concordare con il docente)

2. Scelga un linguaggio di programmazione ed eventuali librerie per l'implementazione dei metodi. Ad es: linguaggio C, oppure Python, Java, R. Esistono ottime librerie di ML per ognuno di questi linguaggi (ad es: Weka per Java, caret e decine di package per R, scikit-learn per Python).

3. Applichi i metodi alla predizione dei termini BP (Biological Process), MF (Molecular Function)e CC (Cellular Component) della GO (Gene Ontology) per l'organismo modello *Drosophila melanogaster*.

4. (Opzionale) Applichi i metodi alla predizione dei termini BP, MF e CC per *Homo sapiens*.

Di seguito sono descritti i dati e le annotazioni da utilizzare (disponibili sul server del Dipartimento di Informatica) ed il set-up sperimentale da seguire.

2 Dati

I dati (feature di ingresso per gli algoritmi di ML) rappresentano grafi indiretti sotto forma di matrici pesate di adiacenza. Ogni riga (colonna) si riferisce quindi ad una diversa proteina dell'organismo ed ogni entry al peso dell'arco che connette due proteine. Gli algoritmi network-based usano direttamente tale matrice per l'apprendimento.

Gli algoritmi induttivi supervisionati possono utilizzare la riga iesima di tali matrici di adiacenza come un vettore di feature da associare alla iesima proteina. Tale vettore rappresenta il dato di ingresso per la learning machine. In altre parole le righe di tali matrici corrispondono agli esempi e le colonne alle feature associate a ciascun esempio

Nel caso del moscerino la matrice di adiacenza è costruita tramite l'integrazione di 8 differenti tipologie di dati ottenuti da database pubblici (Tabella 1) ed include 3195 proteine.

Database	Type of data
PRINTS [1]	Motif fingerprints
PROSITE [6]	Protein domains and families
Pfam [3]	Protein domain
SMART[8]	Simple Modular Architecture Research Tool (database annotations)
InterPro [9]	Integrated resource of protein families, domains and functional sites
Protein Superfamilies[5]	Structural and functional annotations
EggNOG [10]	Evolutionary genealogy of genes: Non-supervised Orthologous Groups
Swissprot [2]	Manually curated keywords describing the function of the proteins at different degrees of abstraction

Table 1: Data base e tipi di dati usati per costruire la matrice di adiacenza del grafo delle proteine di *Drosophila melanogaster*.

Nel caso dell' uomo la matrice di adiacenza e' stata ottenuta direttamente da *STRING* [11], il database di riferimento per le relazioni proteina-proteina ed include 19247 proteine.

Le matrici delle annotazioni (associazioni proteina - termine GO) sono state ottenute dal database Swissprot (http://www.expasy.org/). Ogni riga della matrice delle annotazioni corrisponde ad una proteina e le colonne si riferiscomo ai termini GO. Le entry della matrice sono 1 se la proteina e' annotata per il termine GO, 0 altrimenti. Sia per il moscerino che per l'uomo sono considerati solo i termini con almeno 5 annotazioni. Il numero delle classi (termini) risultanti e' per il moscerino 235 (CC), 234(MF), 1951 (BP) e per l'uomo 601 (CC), 899 (MF) e 3958 (BP).

Disponibilità dei dati. Tutti i dato sono scaricabili da: `http://homes.di.unimi.it/valentini/DATA/ProgettoBioinf1617`. I file sono in formato testo compresso (gzip):

A. File delle matrici di adiacenza: righe e colonne rappresentano proteine. La prima riga contiene gli identificatori delle proteine (colonne). Dalla seconda riga in poi la prima entry e' il nome della proteina seguita da valori numerici (una entry per ogni colonna). Le entry sono separate da tab:

- Dros.adjmatrix.txt.gz : matrice di adiacenza del moscerino
- Homo.adjmatrix.txt.gz : matrice di adiacenza dell'uomo

B. File delle annotazioni: le righe corrispondono a proteine, le colonne a termini GO. La prima riga contiene gli identificatori delle classi GO (colonne). Dalla seconda riga in poi la prima entry e' il nome della proteina seguita da valori numerici (0 oppure 1, una entry per ogini colonna). Le entry sono separate da tab:

- Dros.BP.ann.txt.gz : Tabella delle annotazioni BP per il moscerino
- Dros.MF.ann.txt.gz : Tabella delle annotazioni MF per il moscerino
- Dros.CC.ann.txt.gz : Tabella delle annotazioni CC per il moscerino
- Homo.BP.ann.txt.gz : Tabella delle annotazioni BP per l'uomo
- Homo.MF.ann.txt.gz : Tabella delle annotazioni MF per l'uomo
- Homo.CC.ann.txt.gz : Tabella delle annotazioni CC per l'uomo

3 Set-up sperimentale

Per valutare le performance del metodo si usi la tecnica sperimentale della 5-fold cross-validation. Si usino le seguenti metriche:

1. Misure "per class": Area Under the Receiver Operating Characteristic curve (AUROC) and the Area Under the Precision Recall Curve (AUPRC);
2. Misure "per-example" la Precision, Recall ed F-score gerarchica.

Se indichiamo con $TP_j(t)$, $TN_j(t)$ and $FP_j(t)$ rispettivamente il numero dei true positive, true negative e false positive per la proteina j computati con la soglia t applicata all'output della learning machine, possiamo definire la "per-example" multiple-label precision $Prec(t)$ e sensibilità (recall) $Rec(t)$ per una data soglia t:

$$Prec(t) = \frac{1}{n}\sum_{j=1}^{n}\frac{TP_j(t)}{TP_j(t)+FP_j(t)} \qquad Rec(t) = \frac{1}{n}\sum_{j=1}^{n}\frac{TP_j(t)}{TP_j(t)+FN_j(t)} \quad (1)$$

dove n è il numero degli esempi (proteine). In altre parole $Prec(t)$ ($Rec(t)$) è la multi-label precision (recall) mediata su tutti gli esempi. La F-score multi-label dipende da t ed in accordo con il setting sperimentale di CAFA2 [7], l' F-score massimo ottenibile($Fmax$) è il seguente:

$$Fmax = max_t \frac{2Prec(t)Rec(t)}{Prec(t)+Rec(t)} \quad (2)$$

Per la computazione dell' AUROC e dell'AUPRC esistono diverse librerie disponibili. Potete implementare direttamente l'F-score gerarchico oppure usare la funzione find.best.f disponibile nel file R `F-hier.R`.

4 Report e codice

Ogin gruppo deve preparare un report che descriva brevemente il lavoro svolto (metodo scelto, set-up sperimentale, risultati ottenuti). La consegna deve essere effettuata almeno 7 giorni prima della data fissata per l'orale.

N.B.: La parte sperimentale deve essere descritta in modo che gli esperimenti siano riproducibili.

In appendice al report aggiungete i listati del codice che avete utilizzato. Il sw deve essere documentato in modo che possa essere usato facilmente anche senza leggere il codice stesso.

References

[1] Terri K. Attwood, Paul Bradley, Darren R. Flower, Anna Gaulton, Neil Maudling, AL Mitchell, G Moulton, A Nordle, K Paine, P Taylor, et al. Prints and its automatic supplement, preprints. *Nucleic acids research*, 31(1):400–402, 2003.

[2] The UniProt Consortium. Uniprot: a hub for protein information. *Nucleic Acids Research*, 43(D1):D204–D212, 2015.

[3] Robert D Finn, Jaina Mistry, Benjamin Schuster-Böckler, Sam Griffiths-Jones, Volker Hollich, Timo Lassmann, Simon Moxon, Mhairi Marshall, Ajay Khanna, Richard Durbin, et al. Pfam: clans, web tools and services. *Nucleic acids research*, 34(suppl 1):D247–D251, 2006.

[4] M. Frasca, A. Bertoni, and G. Valentini. Unipred: Unbalance-aware network integration and prediction of protein functions. *J. Comput. Biol.*, 22(12):1057–1074, 2015.

[5] Julian Gough, Kevin Karplus, Richard Hughey, and Cyrus Chothia. Assignment of homology to genome sequences using a library of hidden markov models that represent all proteins of known structure. *Journal of molecular biology*, 313(4):903–919, 2001.

[6] Nicolas Hulo, Amos Bairoch, Virginie Bulliard, Lorenzo Cerutti, Edouard De Castro, Petra S Langendijk-Genevaux, Marco Pagni, and Christian JA Sigrist. The prosite database. *Nucleic acids research*, 34(suppl 1):D227–D230, 2006.

[7] Y. Jiang et al. An expanded evaluation of protein function prediction methods shows an improvement in accuracy. *Genome Biology*, 17(184), 2016.

[8] Ivica Letunic, Richard R Copley, Birgit Pils, Stefan Pinkert, Jörg Schultz, and Peer Bork. Smart 5: domains in the context of genomes and networks. *Nucleic acids research*, 34(suppl 1):D257–D260, 2006.

[9] Nicola J Mulder, Rolf Apweiler, Teresa K Attwood, Amos Bairoch, Alex Bateman, David Binns, Peer Bork, Virginie Buillard, Lorenzo Cerutti, Richard Copley, et al. New developments in the interpro database. *Nucleic acids research*, 35(suppl 1):D224–D228, 2007.

[10] Jean Muller, Damian Szklarczyk, Philippe Julien, Ivica Letunic, Alexander Roth, Michael Kuhn, Sean Powell, Christian von Mering, Tobias Doerks, Lars Juhl Jensen, et al. eggnog v2. 0: extending the evolutionary genealogy of genes with enhanced non-supervised orthologous groups, species and functional annotations. *Nucleic acids research*, 38(suppl 1):D190–D195, 2010.

[11] Damian Szklarczyk, Andrea Franceschini, Stefan Wyder, Kristoffer Forslund, Davide Heller, Jaime Huerte-Cepas, Milan Simonovic, Alexander Roth, Alberto Santos, Kalliopi P. Tsafou, Michael Kuhn, Peer Bork, Lars J. Jensen, and Christian von Mering. String v10: protein˘2013protein interaction networks, integrated over the tree of life. *Nucleic Acids Research*, 2014.

[12] G. Valentini, Armano G., M. Frasca, J. Lin, M. Mesiti, and M. Re. RANKS: a flexible tool for node label ranking and classification in biological networks. *Bioinformatics*, 32:2872–2874, 2016.