Previsione del reddito: metodi di classificazione supervisionata

Elaborato finale di Machine Learning svolto da Bettani Elia, Gregori Lorenzo, Pellegata Alessandra, Rola Stefano

Abstract

In uno studio effettuato su circa 32 000 individui sono state rilevate alcune variabili di natura socio-economica. I dati sono stati reperiti dall'archivio dell'Università della California Irvine (UCI data repository). Partendo da questo studio l'obiettivo è costruire un modello classificatore per effettuare delle previsioni robuste sul reddito annuale in funzione di tali caratteristiche.

Tramite l'utilizzo del software Knime è stato possibile confrontare la bontà di classificazione di diversi modelli predittivi, e scegliere il più efficiente in termini di prestazioni. Per questo motivo dopo una prima fase in cui è stata svolta un'analisi descrittiva del dataset e in cui sono state applicate tutte le procedure di pre-processing, è stata valutata l'efficienza predittiva dei classificatori e scelto il modello più performante, in grado cioè di classificare al meglio le osservazioni del test set.

Sommario

1. Introduzione

Con l'aumentare continuo della disponibilità di dati in formato elettronico cresce il bisogno di metodi automatici per la loro analisi. Lo scopo del Machine Learning è sviluppare metodi che automaticamente cerchino patterns nei dati e usino tali patterns per effettuare delle previsioni o per descrivere la struttura intrinseca dei dati. Dunque possiamo suddividere la disciplina in apprendimento supervisionato e non supervisionato, a seconda dello studio che si intende effettuare. Lo scopo dell'apprendimento supervisionato è prevedere il valore di un particolare attributo Y a partire dai valori di altri attributi X; le tecniche di apprendimento non supervisionato invece mirano a esplorare strutture particolari contenute nei dati che riassumano le relazioni sottostanti ad essi.

A partire da informazioni di natura socio-economica rilevate su 32561 individui il seguente elaborato affronta un problema di predictive learning (ossia apprendimento supervisionato) in riferimento alla variabile target *income*.

2. Dataset

Nel dataset considerato sono state rilevate 15 variabili, di cui 6 continue e 9 categoriali. Tra queste, è stata considerata come variabile indipendente *income*, variabile dicotomica che esprime il reddito su due livelli: individui che hanno un reddito annuale maggiore di 50mila dollari e individui che hanno il reddito annuale minore o uguale alla stessa somma. La natura dicotomica della variabile indipendente ha suggerito di costruire dei modelli classificatori per effettuarne delle previsioni. Nel dataset la variabile *income* si divide con le seguenti frequenze nei due livelli: 23.93% per *income* >50k e 76.06% per *income* <=50k.
Le osservazioni che compongono il dataset sono 32561; un'ulteriore analisi ha permesso di stabilire una percentuale pari al 7% di osservazioni che presentavano almeno un valore mancante.

2.1 Variabili di input

Di seguito è riportata una breve descrizione delle variabili di input del dataset in analisi.

- *age*: variabile quantitativa discreta che indica l'età associata all'individuo oggetto di studio.
- *workclass*: variabile categoriale che indica la tipologia d'impiego. Può assumere i valori: Private, Self-emp-not-inc, Self-emp-inc, Federal-gov, Local-gov, State-gov, Without-pay, Never-worked.
- *fnlwgt*: variabile continua contenente un valore numerico pesato per le caratteristiche socio-economiche dell'individuo in analisi.
- *education*: variabile categoriale che indica il livello massimo di istruzione. Può assumere i valori: Bachelors, Some-college, 11th, HS-grad, Prof-school, Assoc-acdm, Assoc-voc, 9th, 7th-8th, 12th, Masters, 1st-4th, 10th, Doctorate, 5th-6th, Preschool.
- *education-num*: variabile numerica che associa un numero progressivo ad ogni categoria della variabile *education*. Può assumere valori da 1 a 16.
- *marital-status*: variabile categoriale che indica lo stato civile dell'individuo oggetto di studio. Assume i seguenti valori: Married-civ-spouse, Divorced, Never-married, Separated, Widowed, Married-spouse-absent, Married-AF-spouse.
- *occupation*: variabile categoriale che indica l'ambito lavorativo in cui l'individuo risulta impiegato. Assume i valori: Tech-support, Craft-repair, Other-service, Sales, Exec-managerial, Prof-specialty, Handlers-cleaners, Machine-op-inspct, Adm-clerical, Farming-fishing, Transport-moving, Priv-house-serv, Protective-serv, Armed-Forces.
- *relationship*: variabile categoriale indicante il legame familiare più importante per l'individuo in analisi. Si divide nelle seguenti categorie: Wife, Own-child, Husband, Not-in-family, Other-relative, Unmarried.
- *race*: variabile categoriale che indica l'origine geografica delle persone incluse nello studio. Si esplica in: White, Asian-Pac-Islander, Amer-Indian-Eskimo, Other, Black.
- *sex*: indica il sesso della persona, tramite Male e Female
- *capital-gain*: variabile continua che rappresenta il capitale guadagnato dall'individuo nell'anno in cui è stata svolta l'analisi (1994).
- *capital-loss*: variabile continua che rappresenta la perdita di capitale dell'individuo nell'anno in cui è stata svolta l'analisi (1994).
- *hours-per-week*: variabile continua che rappresenta il numero di ore lavorate settimanalmente da ogni individuo oggetto di analisi.
- *native-country*: variabile categoriale che indica il paese di provenienza dell'osservazione presa in considerazione. I paesi che vengono inclusi nell'indagine sono: United-States, Cambodia, England, Puerto-

Rico, Canada, Germany, Outlying-US(Guam-USVI-etc), India, Japan, Greece, South, China, Cuba, Iran, Honduras, Philippines, Italy, Poland, Jamaica, Vietnam, Mexico, Portugal, Ireland, France, Dominican-Republic, Laos, Ecuador, Taiwan, Haiti, Columbia, Hungary, Guatemala, Nicaragua, Scotland, Thailand, Yugoslavia, El-Salvador, Trinadad&Tobago, Peru, Hong, Holand-Netherlands.

3. Pre-processing

La parte di *pre-processing* si divide in questo caso in due fasi principali: in una prima fase vengono individuati i missing values e si studiano dei metodi per il loro trattamento. Nella seconda fase di *pre-processing* sono state rinominate e trattate alcune variabili del dataset, così da ottenere infine un dataset risistemato e ottimale per essere sottoposto alle analisi.

3.1 Trattamento dei Missing Values

Inizialmente è stato possibile osservare come solamente 3 variabili del dataset contengano dei missing values: *workclass*, che contiene 1836 valori mancanti; *occupation*, che ne contiene 1843 e *native-country* che ne contiene 583; osserviamo che sono tutte variabili categoriali. Visto il totale delle osservazioni, pari a 32561, si è ritenuto il numero di valori mancanti presenti nel dataset accettabile per procedere con le analisi. Le strategie che sono state applicate per la gestione dei missing values sono svariate. La prima utilizzata è anche la più comune, e consiste nel rimuovere tutte le osservazioni che contengono almeno un valore mancante al loro interno. La seconda strategia invece sostituisce al valore missing la modalità più frequente della variabile in esame, senza eliminare alcuna riga dal dataset. Un ulteriore metodo prevede la sostituzione del valore mancante con la modalità più frequente condizionata al valore assunto dalla variabile indipendente *income*. L'ultimo metodo sperimentato, infine, consiste nel dividere il dataset in due partizioni: la prima contenente le righe corrispondenti ai valori osservati della variabile in analisi, la seconda contenente le righe corrispondenti a valori mancanti della variabile in analisi. Viene poi costruito un modello a partire dalla prima partizione e vengono effettuate le previsioni sulla seconda partizione, sostituendo ai valori mancanti di quella variabile le previsioni ottenute in precedenza.

Un'analisi più approfondita ha portato a osservare che i valori mancanti sono concentrati sulle stesse osservazioni, che costituiscono una percentuale minima dei dati a disposizione, pari al 7.4% dei records. Di conseguenza, la perdita di informazione a seguito della rimozione di tali osservazioni non risulta onerosa ai fini dello studio.
A seguito della rimozione dei records contenenti missing values, il dataset conta 30162 osservazioni.

3.2 Trattamento delle variabili

Nella seconda fase di *pre-processing* per prima cosa è stata esclusa dalle analisi la variabile *fnlwgt*, in quanto risulta ridondante ai fini della spiegazione della variabile di output *income.* Tale variabile infatti assegna un peso al record in funzione delle caratteristiche socio-economiche dell'individuo in analisi.

È stata poi aggregata la variabile *education-num,* che inizialmente prevedeva 16 classi, in una nuova variabile distribuita su 5 livelli ordinati di uguale frequenza. La suddivisione su 5 livelli è funzionale alla distribuzione delle frequenze all'interno delle categorie originali. Questa suddivisione inoltre rappresenta una buona approssimazione dei corrispondenti cicli di istruzione della scuola italiana. Per evitare ridondanze e possibili problematiche legate al fenomeno della multicollinearità, infine, è stata successivamente rimossa la variabile *education.*

In un secondo momento è stato oggetto di interesse il fatto che le distribuzioni interne delle variabili *capital-gain* e *capital-loss* presentassero numerosi valori nulli. Per ovviare a questo problema, insieme a quello della riduzione della dimensionalità totale del dataset, le due variabili precedentemente citate sono state accorpate in un'unica variabile chiamata *capital-net*. Tale variabile rappresenta la differenza tra il capitale guadagnato e quello perso di ogni individuo del dataset.

Infine è stata presa in considerazione la distribuzione di frequenza della variabile *native-country.*

Dal seguente grafico, infatti, emerge chiaramente come la maggior parte degli individui soggetti all'analisi provenga dagli USA: circa il 91% delle osservazioni appartiene a questa categoria.

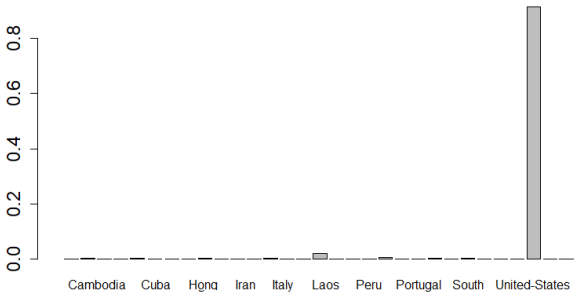

Figura 1. Istogramma raffigurante le frequenze relative associate ad ogni livello della variabile native-country.

È stato deciso quindi di effettuarne una *binarization*, codificando con valore 1 gli individui nati negli Stati Uniti d'America e con valore 0 quelli nati in altri paesi: ciò permette infatti di ridurre la sproporzione numerica tra le due classi. Nonostante questo accorgimento, la proporzione tra le due classi rimane comunque molto sbilanciata.

4. Classificazione

Prima di iniziare a costruire i modelli per la classificazione il dataset è stato diviso in training e test set tramite un campionamento stratificato della variabile target. Il training set contiene i 2/3 delle osservazioni, mentre il test set 1/3 delle osservazioni. La partizione viene effettuata con lo scopo di testare su un campione rappresentativo dei dati di partenza le performance di vari modelli classificatori.

Dunque una volta terminata la fase di *pre-processing* e disponendo quindi di training e test set privi di missing e pronti all'analisi, viene affrontato un problema di classificazione. L'obiettivo è specificare una serie di modelli predittivi e valutare, secondo alcune misure classificative, l'efficacia dei vari modelli costruiti al fine di scegliere quello più performante. Nello specifico si sono costruiti i seguenti modelli:

1. Modelli Euristici: Decision Trees e Random Forest
2. Modelli di regressione e separazione: Regressione Logistica, Support Vector Machine e Artificial Neural Network
3. Modelli probabilistici: Naive Bayes, Naive Bayes Tree e Bayes Network

4.1 Regressione Logistica

Il primo modello preso in esame è il modello logistico. In particolare sono stati sviluppati una regressione logistica semplice e il modello logistico multinomiale. In entrambi i casi l'analisi preliminare ha confermato come tutte le variabili siano interessanti ai fini della spiegazione dell'output. Le accuracy di entrambi i modelli inoltre risultano pari circa a 0.84, sebbene il modello di regressione logistica multinomiale sia preferibile a livello decimale. Nello specifico il modello di regressione logistica multinomiale presenta un livello di accuracy in percentuale pari a 83.829 %, le osservazioni correttamente classificate ammontano a 8429, quelle classificate erroneamente a 1626 e la statistica k di Cohen è pari a 0.534.

4.2 Decision Trees e Random Forest

Successivamente in un metanodo sono stati raggruppati i modelli decision tree, random forest e J48 sulla base della somiglianza nel modo di operare di questi metodi classificativi euristici; tutti i classificatori implementati infatti prevedono in modo analogo lo split delle variabili indipendenti del modello. Questa volta la scelta del modello più efficace sulla base dell'accuracy è più intuitiva, poiché il modello J48 presenta una statistica più elevata in relazione agli altri due modelli sviluppati e pari a 0.849, contro 0.827 raggiunto da decision tree e 0.828 da random forest.

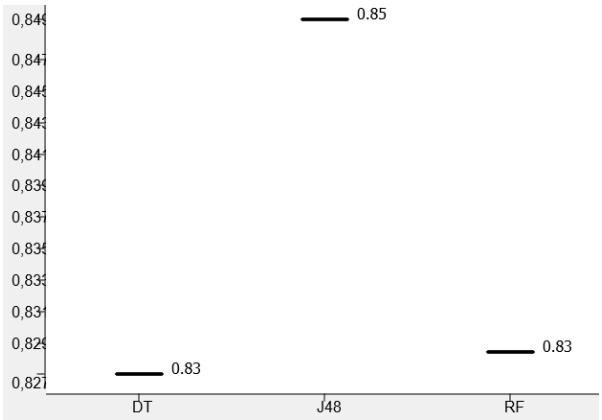

Figura 2. Livelli di Accuracy raggiunti da Decision Tree, J48 e Random Forest

Il modello vincitore perciò è rappresentato da un random forest che classifica tramite l'algoritmo J48 di Weka, un software neozelandese per l'apprendimento automatico.

Sulla base della matrice di confusione, infatti, il modello J48 costruito permette di classificare correttamente un numero di istanze pari a 8543, mentre quelle erroneamente classificate sono pari a 1512.

4.3 Support Vector Machine

Un altro tipo di classificazione effettuata sui dati è quella basata sul metodo del Support Vector Machine, che minimizza l'errore empirico di classificazione massimizzando al contempo il margine geometrico. Se la distribuzione dei dati non è approssimabile linearmente, questo metodo sfrutta la trasformazione kernel per cercare la divisione ottimale dello spazio multidimensionale. Nel dataset in esame risulta necessario utilizzare una trasformazione kernel. In particolare nel workflow di Knime sono stati implementati tre diversi approcci del metodo SVM: una variante stocastica del Pegasos chiamata SPegasos e due learner SMO utilizzando due diversi tipi di kernel: PolyKernel e Puk.

Svolgendo le analisi sui tre diversi metodi è stato immediato verificare che il miglior classificatore in termini di accuracy risulta essere quello che utilizza la funzione PolyKernel: infatti l'accuracy associata a tale modello è pari a 0.837, con 8415 osservazioni classificate correttamente e 1647 classificate erroneamente. È possibile notare che le accuracy associate agli altri modelli sono di poco inferiori a quella del modello scelto, raggiungendo un valore di 0.834 per SPegasos e 0.827 per il modello SMO con funzione Puk.

4.4 Multilayer Perceptron

Il quarto metodo utilizzato è il Multilayer Perceptron, che consiste nel costruire un'architettura di neuroni che comunicano in modo unidirezionale dalle variabili indipendenti di input alla variabile target, che nel dataset in esame è rappresentata dalla variabile *income*. I modelli MLP possono avere architetture differenti, che variano a seconda di due parametri: il numero di strati nascosti e il numero di neuroni contenuti in ciascuno strato.

Nel workflow di Knime sono stati inizialmente implementati due diversi tipi di MLP: il primo metodo, fornito da Weka, sfrutta l'intelligenza artificiale per ottenere la migliore combinazione tra numero di strati nascosti e numero di neuroni contenuti per strato; nel secondo metodo, invece, sono stati specificati entrambi i parametri, scegliendo di utilizzare un solo strato con 30 neuroni. Prima di poter utilizzare il secondo metodo, è stato necessario ricorrere a una opportuna normalizzazione delle variabili per far sì che l'algoritmo funzionasse correttamente.

Tra i due metodi utilizzati il primo è risultato significativamente più performante; infatti l'accuracy ottenuta, 0.837, risulta sensibilmente più elevata rispetto a quella del secondo metodo, pari a 0.244. Data la scarsità della performance ottenuta con il secondo metodo, si è deciso di approfondire ulteriormente l'analisi variando l'architettura del MLP, in particolare provando a costruire modelli con diverse quantità di neuroni per verificare se ci fosse un miglioramento in termini di accuracy. Nello specifico sono stati implementati diversi modelli MLP al variare del numero di neuroni contenuti in uno strato nascosto, rispettivamente 1, 2, 4, 8 e 16 neuroni.

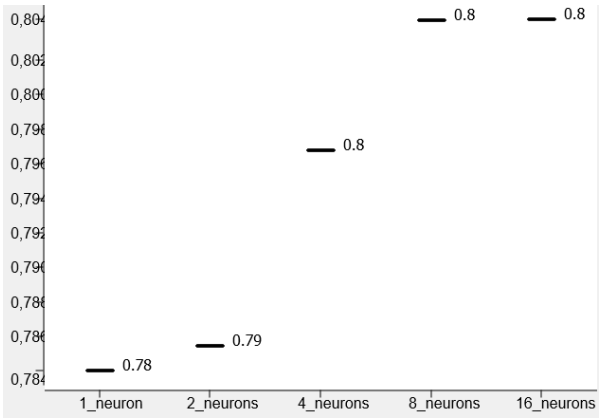

Figura 3. Livelli di accuracy del modello MLP al variare del numero di neuroni.

Il livello di accuracy del modello MLP con un solo neurone è nettamente aumentato rispetto a quello del modello MLP con 30 neuroni; inoltre, come si può dedurre dal grafico, il livello di accuracy aumenta con l'aumentare del numero di neuroni, fino alla soglia di 8 neuroni, in cui il valore dell'accuracy è pari a 0.803, lo stesso del modello

con 16 neuroni. Quindi a parità di livello di accuracy, è stato scelto il modello con 8 neuroni, che ha un'architettura più semplice e quindi preferibile alle altre. Un ultimo confronto da effettuare è tra il modello con 8 neuroni e il modello MLP fornito da Weka descritto in precedenza: tra essi il secondo risulta essere più performante in termini di accuracy.

È stato notato come lo svolgimento dell'algoritmo dei MultiLayer Perceptron risulti essere computazionalmente molto oneroso, gravando sui tempi di esecuzione del workflow.

4.5 Naïve Bayes

Il Naïve Bayes è un classificatore bayesiano che richiede l'assunzione di indipendenza condizionale e permette di calcolare la probabilità a posteriori della variabile dipendente date le covariate.

In questa sezione del workflow sono stati costruiti due modelli bayesiani utilizzando due nodi differenti: uno implementa il Naïve Bayes classico, l'altro implementa la versione costruita da Weka, che si serve dell'intelligenza artificiale. Il modello implementato da Knime fornisce un'accuracy di 0.785, classificando correttamente 7897 osservazioni ed erroneamente 2158. Il modello fornito da Weka, invece, ottiene un valore di accuracy pari a 0.818, con 8225 osservazioni correttamente classificate e 1830 osservazioni erroneamente classificate. Tra questi due modelli la differenza tra i valori delle accuracy è più significativa: la scelta è quindi ricaduta sul modello costruito da Weka.

4.6 Bayes Network e Tree Augmented Naïve Bayes

L'ultimo modello utilizzato è il Bayes Network. Il Bayes Network generalizza il classificatore Naïve Bayes sfruttando il concetto di sparsità, che permette di evitare l'assunzione di indipendenza condizionale. L'architettura dei network prevede che la variabile di target, in questo caso la variabile *income*, sia collegata a ciascuna delle variabili esplicative. La particolarità del modello BayesNet è che le variabili esplicative sono collegate tra di loro, senza però effettuare collegamenti circolari. Una variazione del modello BayesNet è il Tree Augmented Naive Bayes, in cui ogni esplicativa può avere al massimo una variabile esplicativa parente. Si differenzia dal TANB il modello Naive Bayes Tree, il quale prevede la costruzione di un albero decisionale al posto dell'architettura descritta in precedenza.

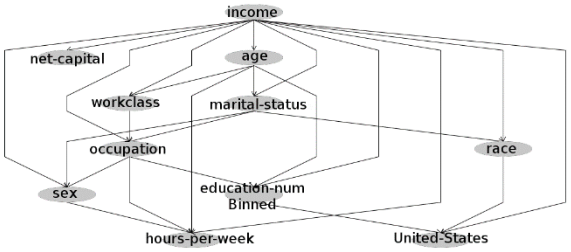

Figura 4. Architettura del classificatore BayesNet

Nel workflow vengono presentati tre tipi diversi di modelli: Tree Augmented Naive Bayes, Naive Bayes Tree e Bayes Net Classifier, che sono descritti nel paragrafo precedente. Andando a osservare e confrontare le accuracy dei modelli citati, è emerso che il miglior modello, ovvero il BNC, presenta un livello di accuracy pari a 0.855.

Gli altri due modelli presentano una accuratezza di poco inferiore, pari a 0.854 nel caso del modello NBTree e pari a 0.853 nel caso del TANB. È inoltre interessante osservare che il livello di accuracy di questi modelli è mediamente più elevato rispetto a quello di altri tipi di classificatori utilizzati.

5. Confronto tra i modelli

In un metanodo sono state raccolte gli indicatori per valutare l'efficienza dei classificatori implementati. La prima e più importante quantità considerata è l'accuracy, ossia la percentuale delle osservazioni correttamente classificate sul totale delle osservazioni.

$$Accuracy = \frac{TP + TN}{TP + FP + TN + FN}$$

Sono state perciò confrontate le accuracy dei sei modelli migliori valutati in precedenza.

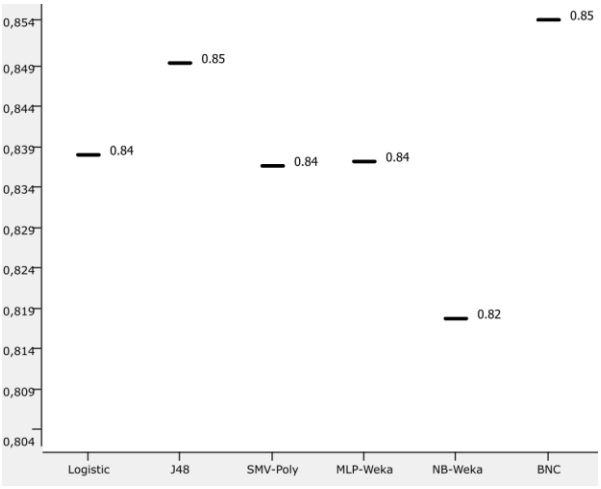

Figura 5. Livelli di accuracy dei modelli classificatori

Secondo questa misura il modello più performante risulta essere il BNC, con un'accuracy associata pari a 0.855.

Per approfondire l'analisi sono state considerate altre misure calcolate a partire dalle matrici di confusione dei diversi modelli. Nel dettaglio abbiamo utilizzato recall, precision, specifity e f-measure, di cui di seguito si riportano le formule:

$$Recall = \frac{TP}{TP + FN}$$

$$Precision = \frac{TP}{TP + FP}$$

$$Specifity = \frac{TN}{TN + FP}$$

F-measure è una media armonica pesata di precision e recall:

$$F = 2 * \frac{Prec * Rec}{Prec + Rec}$$

La principale differenza tra queste misure di precisione e l'accuracy è che le prime vengono calcolate distinguendo per le due classi della variabile *income*, ottenendo quindi due valori per ciascun modello. Si procede dunque analizzando questi valori per verificare se il modello BNC sia effettivamente preferibile anche in questi termini. Le prime misure analizzate sono recall e specifity, che vengono considerate congiuntamente poiché il valore della recall calcolata sulla classe <=50K è lo stesso del valore della specifity calcolata sulla classe >50K: sono infatti misure speculari. Osservando i valori della recall per la classe <=50K calcolata nei vari modelli, si può notare che particolarmente significativa è quella del modello NB, con un valore pari a 0.967; tuttavia il valore della recall per il modello NB nella classe >50K è di 0.368, e viene considerato non soddisfacente. Valutando congiuntamente i valori della recall nelle due classi, il modello BNC risulta essere nuovamente il miglior compromesso per la classificazione della variabile *income*.

MODELLI*INCOME*	**<=50K**	**>50K**
LOGISTIC	0.928	0.566
J48	0.943	0.567
SVM-POLY	0.926	0.569
MLP-WEKA	0.908	0.626
NB	0.967	0.368
BNC	0.936	0.61

Figura 6. Valori di recall nelle classi di income

In tutti i modelli si nota come, essendo il numero di osservazioni appartenenti alla classe <=50K pari al 75% delle osservazioni totali, il valore della recall dei vari modelli nella classe <=50K sia mediamente più alto dei valori nell'altra classe.

Si passa poi a considerare la precision, che indica la percentuale dei valori veramente >50K sul totale delle osservazioni classificate >50K.

MODELLI*INCOME*	**<=50K**	**>50K**
LOGISTIC	0.866	0.724
J48	0.868	0.769
SVM-POLY	0.866	0.717
MLP-WEKA	0.88	0.692
NB	0.822	0.788
BNC	0.879	0.76

Figura 7. Valori di precision nelle classi di income

Dai livelli di precision emerge che il modello BNC risulta nuovamente il migliore, affiancato dal modello J48, ugualmente molto performante. Questo si poteva già osservare con le misure di recall e accuracy.

Per avere un'ulteriore conferma dei risultati ottenuti si è utilizzata un'ultima misura di paragone, la f-measure, che essendo una media pesata risulta più accurata ai fini dell'analisi.

MODELLI*INCOME*	<=50K	>50K
LOGISTIC	0.896	0.635
J48	0.904	0.652
SVM-POLY	0.895	0.635
MLP-WEKA	0.893	0.657
NB	0.889	0.502
BNC	0.907	0.677

Figura 8. Valori di f-measure nelle classi di income

Osservando i valori riportati in tabella per entrambe le classi di *income* risulta che il modello BNC è quello che classifica al meglio i dati, di poco migliore rispetto al modello J48.

È emerso chiaramente dalle analisi come quanto suggerito inizialmente dall'accuracy sia stato confermato da tutte le altre misure: possiamo quindi ritenere essa un buon indicatore della bontà di classificazione dei dati.

5.1 Intervalli di confidenza

Avendo trovato due modelli molto validi è parso opportuno effettuare un confronto più approfondito tra i due.

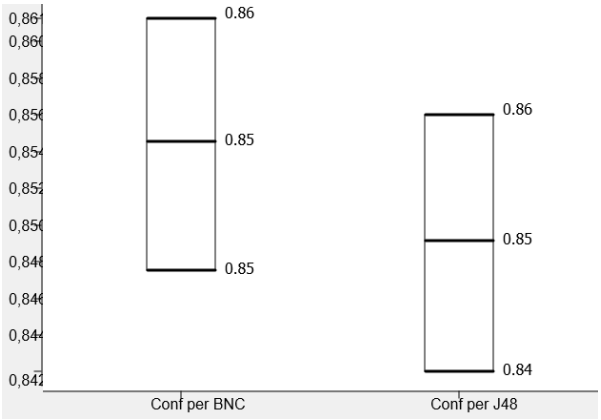

Figura 9. Intervalli di confidenza per accuracy

È stato costruito un metanodo contenente gli intervalli di confidenza per l'accuracy dei modelli BNC e J48. Come si può osservare dal boxplot l'intervallo di confidenza per il modello BNC è mediamente migliore rispetto a quello del modello J48. Nonostante la differenza non sia significativa, scegliamo di approfondire l'analisi del modello BNC.

5.2 Iterated holdout e keyfolds

Successivamente è stato costruito il modello BNC su due partizioni alternative del dataset, una ottenuta con il metodo dell'iterated holdout e una ottenuta con il metodo k-folds. Il modello applicato a questi diversi tipi di partizione restituisce 5 valori di accuracy quando il modello è applicato al training test, mentre un solo valore di accuracy per il modello applicato al test set. Il motivo per cui nel training set esistono diversi valori di accuracy è il fatto che queste partizioni sono state reiterate per 5 volte ciascuna.

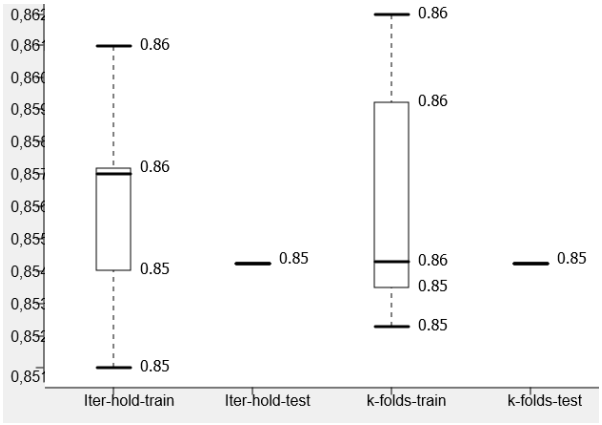

Figura 10. Livelli di accuracy per partizioni con iterated holdout e keyfolds.

Nel grafico soprastante vengono riportati i vari livelli di accuracy per le due partizioni implementate sul modello BNC. I livelli di accuracy rimangono costanti costruendo e testando il modello sulle due differenti partizioni: questo garantisce che il modello scelto abbia delle buone performance sul dataset indipendentemente dal tipo di partizione applicata.

6. Conclusioni

In questo elaborato è stata inizialmente affrontata un'analisi preliminare del dataset, per ottenere informazioni di natura descrittiva sulle variabili oggetto di studio: tale fase di pre-processing ha permesso di trasformare e selezionare le variabili da includere nel modello. Il passo successivo di costruzione dei classificatori ha permesso di individuare come modello più efficiente il BayesNet Classifier: tutti gli indicatori di performance (accuracy, recall, precision, …) portano a selezionare tale modello. La struttura particolare del classificatore BNC permette infatti performance migliori con variabili indipendenti di natura differente, in particolare di tipo categoriale: probabilmente è questa peculiarità che permette nel nostro caso al

modello BNC di essere il più efficace tra i differenti classificatori costruiti. Ulteriore conferma di questo risultato è fornita dal partizionamento alternativo secondo i metodi iterated holdout e keyfolds: implementando il modello BNC anche su tali partizioni la sua accuracy rimane ottimale e intorno ai valori raggiunti in precedenza.
Un futuro sviluppo dell'analisi potrebbe concentrarsi sullo studio del singolo modello BNC, valutandone in maniera approfondita aspetti statisticamente rilevanti quali la significatività dei parametri e le correlazioni tra le variabili ai fini della spiegazione del reddito.

Riferimenti

[1] Fonte del dataset: https://archive.ics.uci.edu/ml/datasets/Adult
[2] Informazioni sul dataset: http://www.cs.toronto.edu/~delve/data/adult/adultDetail.html
[3] https://www.kaggle.com/uciml/adult-census-income
[4] N. Friedman et al., "Bayesian Network Classifiers", 1997
[5] https://statisticalatlas.com/United-States/Educational-Attainment#definitions
[6] Pang-Ning Tan, Michael Steinbach, Vipin Kumar, "Introduction to Data Mining", 2006