Appuntamento dal medico: studio sulle assenze dei pazienti alle visite mediche

Nabil El Asri[1], Riccardo Levi[1], Lorenzo Lorgna[1], Mario Pedol[1], Giorgio Pizzati[2]

[1]*CdLM Data Science, Università degli studi di Milano Bicocca*
[2]*CdLM Fisica, Università degli studi di Milano Bicocca*

Fissato un appuntamento dal medico, è possibile prevedere se un paziente si presenterà o meno in base ai suoi dati anagrafici, clinici e al tempo che dovrà aspettare prima della data stabilita?

Questa domanda vuole essere il punto di partenza del nostro studio che, cercando di prevedere le possibili assenze dei pazienti tramite tecniche di apprendimento supervisionato, si propone di migliorare la programmazione delle agende dei medici e di identificare le possibili cause che portano le persone a non presentarsi. Inoltre, quest'ultimo punto permetterebbe agli studi medici di provare ad ovviare al problema, ad esempio, mediante una comunicazione chiara e tempestiva con il paziente.
Al di là di questi possibili miglioramenti, è importante sottolineare anche che non presentandosi ad un appuntamento, quindi non effettuando una visita, si mette a rischio la propria salute, ad esempio, non diagnosticando in tempo gravi patologie.

1 Introduzione

Non presentarsi ad un appuntamento dal medico rappresenta un problema attuale e tutt'ora aperto di grande rilievo nel sistema sanitario internazionale. Questo fenomeno può portare ad evidenti conseguenze dal punto di vista economico e si possono individuare anche ripercussioni sulla salute del paziente. Dal momento in cui nell'ambito dell'assistenza sanitaria il tempo delle persone rappresenta una delle risorse più importanti, è chiaro che il problema degli appuntamenti mancati risulta essere un problema sia del centro medico sia del paziente.

Dal punto di vista economico, secondo uno studio condotto nello scenario americano da Health Management Technology [1], non presentarsi ad una visita ed eventuali ritardi costano al sistema sanitario americano più di 150 mld di dollari all'anno.

Oltre alle principali conseguenze economiche è bene sottolineare che questo fenomeno può avere anche un impatto sulla salute dei pazienti. Infatti, può compromettere la continuità delle cure e l'efficacia dei farmaci con il rischio che malattie acute diventino croniche e con complicazioni a seguire. A questo si aggiunge il fatto che spesso le persone che non si presentano al primo appuntamento, tendono a non tornare più nello studio medico entro 18 mesi, come è stato messo in luce dallo studio condotto da Athenahealth [2].

Indagare sulle possibili cause di un mancato appuntamento risulta essere dunque il punto di partenza per cercare di contrastare questo fenomeno. Secondo una ricerca [3], le principali cause che portano a ciò sono legate al fatto che il paziente o si dimentica oppure ha problemi lavorativi, personali, organizzativi o logistici.

Per far fronte agli scenari sopra riportati sono state pensate soluzioni di diversa natura [4]: dal venire incontro ai pazienti con problemi di trasporti con servizi di *rideshare* come Uber, all'effettuare diverse chiamate di remainder a partire da alcuni giorni prima dell'appuntamento fissato. Non mancano soluzioni basate su modelli predittivi come lo studio effettuato dalla Duke University [5] che ha permesso di prevedere correttamente il non presentarsi all'appuntamento medico di circa 5mila pazienti.

2 Descrizione del dataset

L'analisi condotta fa riferimento a un *dataset* [6] dove sono raccolte alcune informazioni relative a circa 110 mila appuntamenti presso diversi studi medici situati in Brasile nei primi 7 mesi del 2016.

Gli attributi che lo compongono si riferiscono alle caratteristiche degli appuntamenti e ai dettagli anagrafici e clinici dei pazienti. Nello specifico sono:

- *Absence* (Binaria): Variabile target che indica se un paziente si è presentato o meno dal medico.
- *Appointment ID* (nominale): Identificativo dell'appuntamento.
- *Patient ID* (nominale): Identificativo del paziente.
- *Gender* (Binaria): Genere del paziente.
- *Schedule Day* (date-time): Data della prenotazione.
- *Appointment Day* (data-time): Data dell'appuntamento.
- *Age* (intero): Età del paziente.
- *Neighbourhood* (nominale): Quartiere dove è situata la clinica.
- *Scholarship* (binaria): Borsa di studio.
- *Hypertension* (binaria): Paziente affetto da ipertensione arteriosa.
- *Diabetes* (binaria): Paziente affetto da diabete.
- *Alcoholism* (binaria): Paziente alcolista.
- *Handicap* (ordinale): Grado di disabilità.
- *SMS_recived* (binaria): Ricezione o meno di un sms come remainder del giorno prefissato.

2.1 I punti critici

Durante una prima esplorazione dei dati sono emerse tre importanti osservazioni:

1. Come mostrato in *Figura 1* la variabile target presenta una distribuzione non equa delle sue modalità con circa l'80% di presenze dei pazienti agli appuntamenti contro il 20% delle assenze. Tale osservazione tornerà particolarmente utile nella fase di validazione dei modelli.

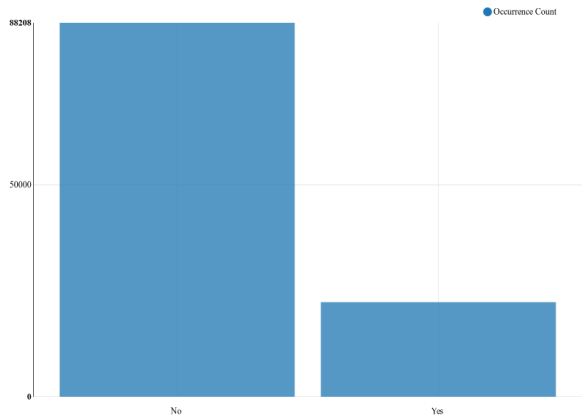

Figura 1: *Distribuzione della variabile Absence*

2. L'attributo *"Age"*, riportante l'età dei pazienti presenta due principali problemi:

   2.1 È presente un record con valore pari a -1.

   2.2 Si osservano diversi pazienti particolarmente anziani che riportano un'età di 115 anni e che quindi destano sospetti in quanto possono essere frutto di errori di imputazione del dato.

3. Osservando l'attributo *"Patient ID"* è possibile notare che per alcuni record ad un paziente sono associati più appuntamenti.

4. Esaminando gli attributi che danno le informazioni sui tempi delle prenotazioni, "*Appointment Day"* e "*Schedule day"*, emergono dei *record* dove la data di prenotazione è la stessa o successiva a quella dell'appuntamento.

Nel paragrafo che segue si proverà a trattare questi punti.

3 Sistemazione del dataset

Prima di procedere con l'analisi si è reso necessario svolgere diverse operazioni sulla struttura del *dataset*.

Come primo punto, si è voluto fare un controllo sulla presenza di valori mancanti. Non essendo stato trovato nessun attributo che presentasse tale problematica si è spostata l'attenzione sulla variabile *"Age"*, dove è presente un record negativo pari a -1, come già osservato al punto 2 del paragrafo 2.1. Poiché non è stato possibile interpretarlo è stato eliminato.

Continuando ad operare sullo stesso attributo, si è voluto selezionare l'intervallo di età [18, 80]. Tale scelta si è basata dopo aver analizzato la probabilità di presentarsi ad un appuntamento rispetto all'età come mostrato in *Figura 2*. Si nota infatti una distribuzione lineare solo per il *range* di età considerato.

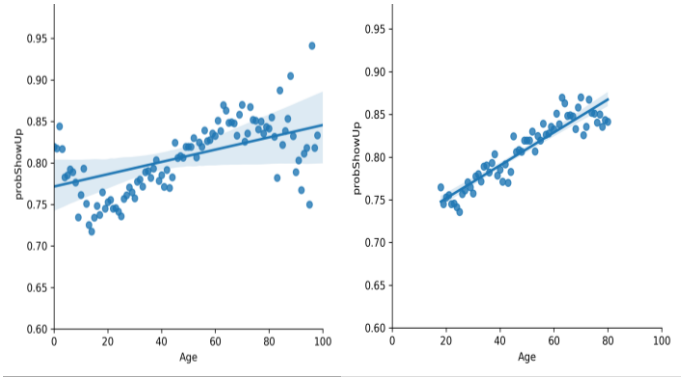

Figura 2: *Distribuzione della variabile No-show*

Questo fenomeno può essere interpretato come, ad esempio, una dipendenza dei minorenni dai genitori che non hanno la possibilità di scegliere se andare dal medico o meno. Lo stesso ragionamento è portato avanti per le persone più anziane che generalmente dipendono dai figli o dai tutori.

Rivolgendo ora l'attenzione sull'attributo *"Patient ID"*, è importante notare che sono presenti pazienti che hanno preso un appuntamento più di una volta durante l'anno. Si è deciso dunque di includere nel *dataset* solo l'appuntamento più recente per ogni persona. Questa operazione, tuttavia, non ha influito sulla distribuzione delle modalità della variabile target.

3.1 Feature Engineering

Dopo aver pulito il *dataset* dai valori problematici si è voluto generare una nuova variabile, *"Delta T"*, che indicasse il tempo intercorso tra le prenotazioni e gli effettivi appuntamenti. L'attributo è stato ricavato come differenza tra *"Appointment Day"*, data dell'appuntamento e *"Schedule Day"*, data della prenotazione.

Come già ci si poteva aspettare dell'osservazione 4 del paragrafo 2.1 nella nuova variabile emergono 2 valori negativi che stanno a significare che la data in cui è stata effettuata la prenotazione è successiva a quella dell'appuntamento. Questi record vengono quindi rimossi in quanto non è possibile ottenere una stima ragionevole.

Ci si concentra ora sull'attributo *"Neighbourhood"*, che indica 81 possibili quartieri dove sono situati gli studi medici. Si prende in considerazione di mappare ogni quartiere con un numero oppure di binarizzare i *record* con la creazione di *log_2(81) =7 feature* binari. Purtroppo, questi metodi, si basano sull'assunzione che il *feature* di partenza sia ordinale, mentre nel nostro caso è nominale in quanto non è possibile stabilire un ordine. In alternativa, si potrebbero creare 81 attributi binari che indicano se un record appartenga ad un certo quartiere o meno. In questo modo, però, si andrebbe incontro ad un aumento notevole dei *features* da analizzare. Preso atto delle considerazioni appena fatte è stata presa la decisione di escludere la variabile in questione.

Inoltre, si è ritenuto opportuno formare due nuovi attributi, *"Previous Appointment"* e *"Miss Previous Appointment"*, ottenuti dai valori passati di *"Patient ID"* prima che venissero eliminati, che indicano se un paziente ha avuto appuntamenti nei mesi precedenti e quante volte non si è presentato.

Infine, come ultima operazione si è voluto eliminare dal *dataset* gli attributi *"Patient ID"* e *"Appointment ID"* utilizzati come identificativi del paziente e dell'appuntamento. Inoltre, è stato rimosso anche l'attributo *"Schedule Day"*, giorno della prenotazione, in quanto l'informazione è già contenuta nella nuova variabile *"Delta T"*, mentre si è voluto mantenere *"Appointment Day"* per valutare se un particolare giorno dei mesi considerati possa influire o meno sul fenomeno oggetto di studio.

In ultima analisi, poiché l'unica variabile non numerica o binaria risulta essere *"Appointment Day"*, si è voluto convertirla in *timestamp*. Con *timestamp* si intende il numero intero di secondi intercorsi tra il 1/01/1970 alle 00:00 e l'evento in analisi. Per avere a che fare con un *range* contenuto è stato deciso di utilizzare come data di riferimento il 1/01/2016 alle 00:00, evento antecedente a tutti gli appuntamenti del *dataset*, evitando così valori negativi.

4 Modellistica

Per questo studio si è deciso di svolgere una prima classificazione senza usare nessuna metodologia di *feature selection*.

Le principali tecniche di *Machine Learning* supervisionato che sono state implementate sono riassumibili in quattro differenti famiglie di modelli:

- Modelli probabilistici: tra i più utilizzati vi sono il *Naive Bayes*, che segue il teorema di *Bayes*, e il *NBTree*, un albero di decisione che poggia sul modello precedente
- Modelli di regressione: poggiano sulla Regressione Logistica *(SLogistic, Logistic)*, modello particolarmente flessibile per il fatto

che gli attributi di input possono essere di differente natura, mentre l'attributo di output è dicotomico

- Modelli euristici: in questa classe di modelli, basata sugli alberi decisionali, è possibile menzionare il *Random Forest (default, implementazione Weka)*, classificatore ottenuto tramite l'aggregazione di più alberi decisionali che permette di gestire anche dati di tipo categorico, e il *J48*, albero di regressione implementato da Weka

- Modelli di separazione: i seguenti modelli partizionano lo spazio degli attributi. Tra questi distinguiamo il *Support Vector Machine (SPegasos)* e il *Multilayer Perceptron*. Quest'ultimo è costituito da neuroni artificiali che comunicano unidirezionalmente, dagli attributi di input all'attributo di classe considerato

5 Dataset sbilanciato

Come già precedentemente osservato nel paragrafo 2.1 e 3 il *dataset* presenta un forte squilibrio delle modalità della variabile target.

Tale sbilanciamento può incidere in maniera significativa sui risultati dei modelli utilizzati, in particolare è ragionevole pensare che un modello tenderà a prevedere correttamente un elevato numero di record dell'evento più frequente ignorando quello meno frequente. Di conseguenza, considerando l'Accuratezza di un modello definita come:

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

Dove:

TP = Veri Positivi, valori positivi classificati correttamente dal modello.

TN = Veri Negativi, Valori negativi classificati correttamente dal modello.

FP = Falsi Positivi, valori negativi classificati positivamente dal modello.

FN = Falsi Negativi, valori positivi classificati positivamente dal modello.

ci si può aspettare un alto livello di Accuratezza per le modalità più frequenti e uno più basso per le meno frequenti.
Si è quindi ritenuto più corretto effettuare una valutazione delle performance dei modelli usando delle misure più opportune.

5.1 Misure di Valutazione di un modello con dataset sbilanciato

Considerando la problematica appena vista, si è voluto procedere con delle misure in grado di fornire una valutazione della performance più precisa. In particolare, si sono calcolate:

I. *Precision*: determina la frazione di record che effettivamente si rivela essere positiva nel gruppo che il classificatore definisce come classe positiva.

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

Assume valori compreso tra 0 e 1, più alta è la precision più è basso il numero di falsi positivi commessi.

II. *Recall*: misura la frazione di record positivi correttamente predetti dal modello di classificazione.

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

Assume valori tra 0 e 1, un alto recall significa pochi record positivi erroneamente predetti come classe negativa.

III. *F-measure*: media armonica tra *recall* e *precision*.

$$F - measure = \frac{2 * r * p}{r + p}$$

Compreso tra 0 e 1, assume 0 solo se almeno una delle misure precedenti vale 0, mentre assume valore 1 se sia precision che recall sono pari a 1.

IV. *AUC*: Area Under Curve della curva ROC, grafico mette in relazione la percentuale dei falsi positivi con quella dei veri positivi.

5.2 Cost Sensitive Learning

Un possibile approccio per trattare il fenomeno della *class imbalance* è quello avvalersi del metodo *cost*

sensitive learning basato sull'utilizzo di una matrice dei costi associata a quella di confusione. La prima stabilisce in base ai falsi positivi e negativi quanto costo bisogna impiegare per classificare un certo oggetto (si tenderà ad avere un costo più alto per la classe più rara erroneamente classificata). La seconda, invece, considera il numero di osservazioni del *Test Set* [1] classificate correttamente, Veri Positivi e Negativi, e quelle classificate in modo errato Falsi Positivi e Negativi.

Con questa metodologia, il classificatore apprende dalla matrice dei costi che associa ad ogni istanza di quella di confusione uno specifico peso. L'obiettivo finale è quello di minimizzare il costo totale [7].

$$Cost = C_{--} * TN + C_{-+} * FP + C_{+-} * FN + C_{++} * TP$$

6 Analisi dei risultati

Prendendo in considerazione i modelli precedentemente selezionati e applicandoli al *dataset* in questione sono state valutate le performance dei singoli modelli con l'obiettivo di individuare il migliore tra essi.

6.1 Holdout

Il *dataset* viene suddiviso in due sottoinsiemi disgiunti, il *training* [2]e il *test set*, attraverso un procedimento di *stratified sampling*[3] in cui la variabile di stratificazione è "*Absence"*. I vari modelli vengono addestrati usando il *training set* e le *performance* sono valutate utilizzando il *test set*.

Models	Recall	Precision	F-measure	Accuracy	AUC
MLP	0,012	0,457	0,023	0,802	0,719
J48	0,021	0,371	0,039	0,800	0,706
NBtree	0,020	0,427	0,038	0,801	0,696
RF1	0,016	0,642	0,032	0,804	0,693
SLogistic	0,009	0,364	0,018	0,801	0,657
SPegasos	0,015	0,346	0,030	0,800	0,657
Logistic	0,010	0,362	0,020	0,801	0,656
RF2	0,192	0,335	0,244	0,765	0,649
NB	0,066	0,309	0,108	0,787	0,644

Tabella 1: classificatori con holdout

Osservando i risultati nella *tabella 1*, è possibile constatare che il *Random Forest (implementazione Weka RF2)* e il *Naive Bayes* risultano essere i migliori modelli in termini di *F-measure*.

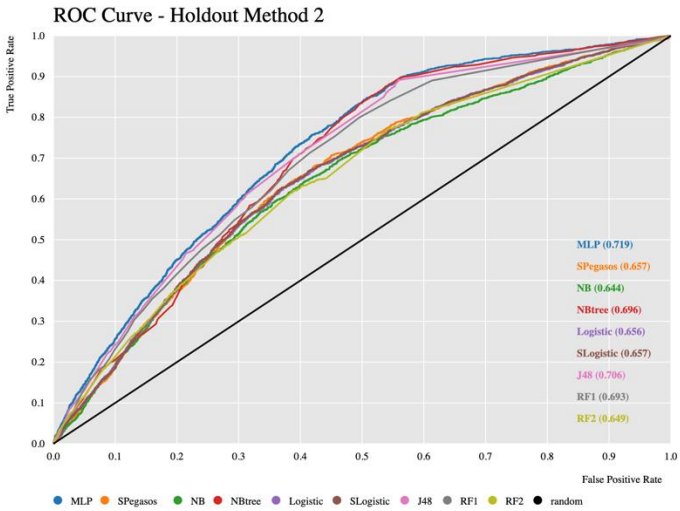

Figura 3: *Roc Curve con metodo Holdout*

Per valutare ulteriormente le performance dei classificatori è stata presa in considerazione l'analisi della curva ROC mostrata in *Figura 3*, che permette di effettuare un confronto tra i vari modelli facendo a meno della distribuzione della classe e del costo dell'errore. Quindi, tutti i modelli presi in considerazione si comportano meglio di un classificatore *"Zero Rule"*, ovvero di un classificatore causale che non si basa su alcuna informazione. Tuttavia, tale misura, l'AUC, non pone enfasi su una classe piuttosto che un'altra e per questo non possibile stabilire quale modello classifichi meglio l'evento più raro, obiettivo dell'analisi. Di conseguenza è stato scelto quello con la F-measure più elevata.

6.2 Cross validation

Il *dataset* viene partizionato in k fogli e ogni record è utilizzato esattamente lo stesso numero di volte nel *training set* ed esattamente una volta nel *test set*. Durante ogni iterazione, una delle partizioni viene scelta per il *testing*, mentre tutte le altre sono utilizzate per il *training*. Questa procedura è dunque ripetuta k volte. Il valore di k scelto è pari a 3.

[1] *Test set:* partizione del *dataset* dove si assume che i valori della variabile di classe siano ignoti, tale sottoinsieme viene usato per testare il modello.

[2] *Training Set*: Partizione del *dataset* usata per addestrare il modello di classificazione

[3] *Stratified sampling:* Tecnica di campionamento dove i record vengono presi in modo che all'interno del campione vengano rispettate le proporzioni tra gli attributi presenti nel *dataset* di partenza

Models	Recall	Precision	F-measure	Accuracy	AUC
MLP	0,021	0,513	0,041	0,807	0,730
J48	0,059	0,349	0,100	0,797	0,708
RF1	0,009	0,510	0,017	0,807	0,701
NBtree	0,006	0,310	0,033	0,806	0,696
pegasos	0,027	0,326	0,050	0,801	0,661
SLogistic	0,012	0,370	0,023	0,805	0,661
Logistic	0,014	0,372	0,027	0,805	0,661
RF2	0,148	0,321	0,203	0,775	0,655
NB	0,086	0,323	0,135	0,788	0,642

Tabella 2: *classificatori con 3-fold*

Anche in questo caso osservando i risultati della *tabella 2* in termini di *F-measure* è possibile giungere alla conclusione che il *Random Forest (implementazione weka RF2)* e il *NB* risultano essere i migliori modelli.

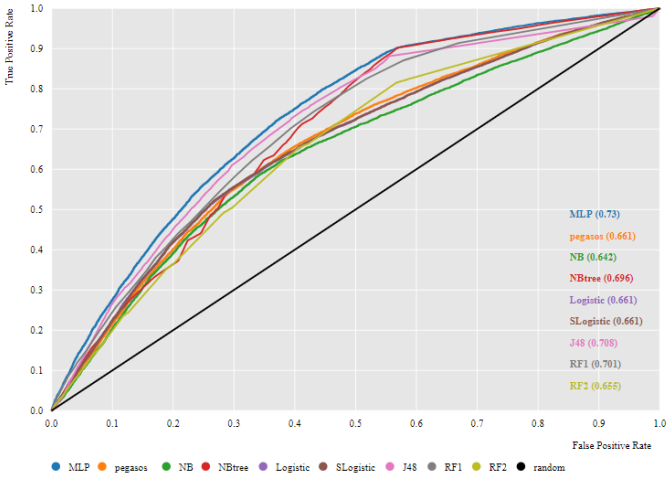

Figura 4: *Roc Curve con metodo 3-fold*

Procedendo con un'ulteriore valutazione delle performance considerando il valore AUC come mostrato in *Figura 4* dei diversi modelli è possibile osservare che il modello *Random Forest (implementazione weka RF2)* ottenuto tramite la tecnica *Cross validation* assume un valore di AUC maggiore rispetto a quello ottenuto con la tecnica *Holdout*. Anche in questo caso tutti i modelli si comportano meglio di un classificatore *"Zero Rule"*.

6.3 Feature selection sulla F-measure

Dopo una preliminare valutazione dei modelli, si è deciso di applicare una *Feature Selection* di tipo *wrapper*. Tale tecnica consiste nell'individuare, tramite un classificatore, l'insieme ottimale degli attributi input con il fine di massimizzare la misura di performance prefissata, in questo caso si è scelta la *F-measure*.

Nella *tabella 3* che segue sono stati riportati i risultati dei modelli ai quali è stata applicata la tecnica appena spiegata mediante il nodo *AttributeSelectClassifier*.

Models	Recall	Precision	F-measure	Accuracy	AUC
MLP	0,011	0,552	0,022	0,807	0,727
J48	0,069	0,364	0,116	0,797	0,705
RF2	0,077	0,330	0,125	0,791	0,669
pegasos	0,027	0,341	0,050	0,802	0,655
Logistic	0,016	0,383	0,031	0,805	0,652
SLogistic	0,013	0,398	0,026	0,805	0,650
NB	0,071	0,305	0,115	0,789	0,644

Tabella 3: *classificatori con feature selection*

Come si può notare le performance dei modelli sembrano essere peggiorati o invariati rispetto alla *3-fold Cross Validation.*

Tuttavia, si sono notati alcuni leggeri miglioramenti in termini di F-meaure nei modelli Logistic, SLogistic e J48. Per il primo sono stati considerati nella classificazione i feature *"Scholarship"*, *"Age"*, *"DeltaT"*, *"Alcoholism"*, per il secondo invece rispetto al precedente non è stato selezionato *"Alcoholism"*. Infine, per il *J48* sono stati ritenuti ottimali gli attributi *"DeltaT"*, *"Alcoholism"*, *"Handicap"*, *AppointmentDay"*, *"Diabetes"*.

6.4 Cost sensitive learning

Per far fronte al problema di *class imbalance* è stato scelto come possibile approccio il *Cost sensitive learning*. Per ogni modello, facendo uso della tecnica di cross validation, il rispettivo training è stato eseguito tramite il nodo *Weka CostSensitiveClassifier*, basato su una specifica matrice di costo. Quest'ultima è stata calcolata in maniera *brute force* considerando diversi valori di costo, cercando di ottimizzare il valore di F-measure.

Models	Recall	Precision	F-measure	Accuracy	AUC
J48	0,171	0,340	0,409	0,776	0,546
NBtree	0,622	0,300	0,405	0,646	0,637
RF2	0,504	0,290	0,368	0,666	0,605
NB	0,431	0,315	0,364	0,709	0,603
Logistic	0,376	0,339	0,357	0,738	0,600
MLP	0,578	0,163	0,254	0,345	0,656

Tabella 4: *Cost Sensitive Learning*

Osservando i valori ottenuti riportati nella *tabella 4* è possibile constatare che, adottando la tecnica *Cost sensitive learning*, il *J48* e il *NBTree* risultano essere i modelli migliori in termini di F-measure.

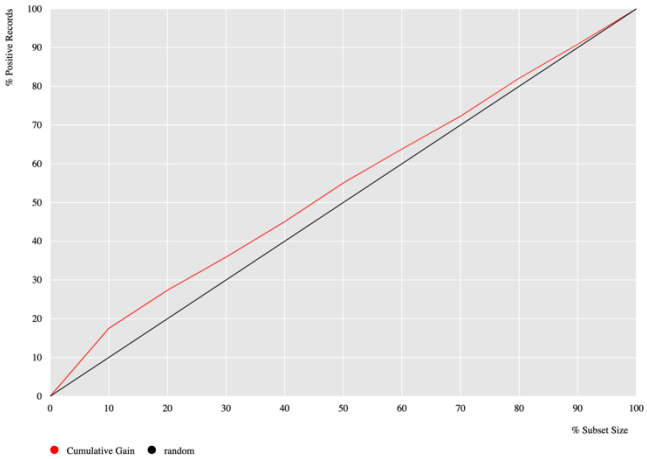

Figura 5: *Cumulative Gain Chart*

Prendendo in considerazione il modello *J48*, il migliore ottenuto in termini di F-measure in seguito alle tecniche precedentemente esposte, è stato analizzato il *Cumulative Gain Chart* rappresentato nella *Figura 5*. Il seguente grafico permette di osservare la percentuale dei record positivi identificati rispetto al totale dei record positivi presi in considerazione. Come si può vedere dal grafico il modello *J48* ha performance leggermente migliori rispetto ad un classificatore *"Zero Rule"*.

Procedendo, invece, con un'analisi specifica dei costi dei singoli modelli è possibile osservare che il modello *NBTree* risulta essere il migliore dal momento in cui per esso è minimo il costo totale calcolato (36543). In termini di costo ottimale, dunque minimo, si distingue anche il modello *J48* che registra un costo simile, leggermente inferiore, a quello del modello precedente (36733). Focalizzando l'attenzione, invece, sul *MLP* è possibile notare che il seguente modello pur avendo un valore di AUC elevato, come riscontrato dalla *Tabella 4*, presenta un elevato costo rispetto ai precedenti (50615). Questo è dovuto al fatto che il modello tende ad ignorare la classe più rara.

7 Conclusioni

In conclusione a questo studio, si è voluto identificare la tecnica di Machine Learning supervisionato migliore per la classificazione dei pazienti che non si presentano ad un appuntamento dal medico.

Il problema più grande che si è incontrato è stato la presenza di una forte *class imbalance* della variabile target. Da una prima analisi, infatti, non si è ritenuto corretto valutare le performance dei modelli utilizzati solo in termini di Accuratezza e AUC ma si è preferito basarsi sulla F-measure che ha permesso di tenere in considerazione tale fenomeno.

Da ciò è possibile trarre una prima conclusione, in particolare si è ritenuto che il modello più performante fosse il *Random Forest* di implementazione Weka.

Per cercare di ottenere performance migliori e al tempo stesso diminuire il numero di attributi da prendere in considerazione è stata applicata anche una tecnica di *feature selection*. I risultati ottenuti non sono stati soddisfacenti, infatti solo un modello ha registrato un leggero miglioramento in seguito all'applicazione di tale tecnica.

Trattando il problema con un approccio che fa riferimento all'analisi dei costi, una delle possibili soluzioni presenti in letteratura, si è riscontrato rispetto al caso precedente che il modello migliore risulta essere il *NBTree* seguito dal *J48*. Ponendo l'obiettivo di minimizzare il costo del modello è emerso che il *Random Forest*, modello candidato precedentemente ad essere il migliore per performance, presuppone un costo elevato.

7.1 Sviluppi futuri

Pensando a possibili sviluppi futuri si potrebbe approcciare al metodo *Cost sensitive learning* con una matrice dei costi differente con l'obiettivo di minimizzare il costo del modello scelto. Infatti, la matrice dei costi individuata in questo lavoro non ha permesso di affrontare il problema della *class imbalance* in maniera efficiente. Emerge dunque la necessità di adottare un algoritmo che possa individuare la matrice dei costi ottimale ai nostri scopi, con l'intento di aumentare le performance del modello considerato.

Un ulteriore sviluppo futuro potrebbe consistere in un'analisi più accurata dei parametri ottimali da utilizzare nel processo di *feature selection*. Ciò porterebbe ad una più corretta individuazione dei parametri, selezionandone solo alcuni di quelli di partenza, da utilizzare nelle fasi di training dei modelli per ottenere conseguentemente dei risultati migliori.

Infine, un ultimo approccio che si potrebbe seguire è quello dell'*equal size sampling*, tecnica con la quale si campiona in modo casuale il *dataset* rispetto la variabile target, con la stessa proporzione per gli eventi rari ed eventi frequenti. Tuttavia, creando un *dataset* non sbilanciato si sollevano le problematiche legate al fatto che si eliminano dei dati che si potrebbero rivelare utili al fine della classificazione.

8 Riferimenti

[1] Jamie Gier, *Missed appointments cost the U.S. healthcare system $150B each year*, 2017, https://www.scisolutions.com/uploads/news/Missed-Appts-Cost-HMT-Article-042617.pdf

[2] Chris Hayhurst, *No-show effect: Even one missed appointment risks retention*, 2019, https://www.athenahealth.com/knowledge-hub/financial-performance/no-show-effect-even-one-missed-appointment-risks-retention

[3] Ullah, Saif & Sangeetha, Rajan & Todd, Liu & Ellen, Demagistris & Regina, Jahrstorfer & Anandan, Swapna & Gentile, Christina & Gill, Angad, *Why do Patients Miss their Appointments at Primary Care Clinics?,* 2018, Journal of Family Medicine and Disease Prevention. 4. 10.23937/2469-5793/1510090

[4] Sachin H. Jain, Missed Appointments, Missed Opportunities: Tackling *The Patient No-Show Problem*, 2019, https://www.forbes.com/sites/sachinjain/2019/10/06/missed-appointments-missed-opportunities-tackling-the-patient-no-show-problem/?sh=fc7eec4573bd

[5] Xiruo Ding, Ziad F Gellad, Chad Mather, III, Pamela Barth, Eric G Poon, Mark Newman, Benjamin A Goldstein, *Designing risk prediction models for ambulatory no-shows across different specialties and clinics*, Journal of the American Medical Informatics Association, Volume 25, Issue 8, August 2018, Pages 924–930, https://doi.org/10.1093/jamia/ocy002

[6] Joni Hoppen, *Medical Appointment No Shows Why do 30% of patients miss their scheduled appointments?, https://www.kaggle.com/joniarroba/noshowappointments*

[7] Ling, Charles & Sheng, Victor. (2010), *Cost-Sensitive Learning and the Class Imbalance Problem.*, 2010, Encyclopedia of Machine Learning