Smart Portfolio Management con tecniche di *Machine Learning*

Antonio Branda[1], Niccolò Gava[1], Fabio Grasso[2], Francesco Lamaro[1], Susanna Levantesi[2]

Abstract

Obiettivo del lavoro è lo sviluppo di un sistema di supporto per la realizzazione del trading algoritmico, ovvero l'utilizzo di un algoritmo di filtraggio delle informazioni che fornisca raccomandazioni personalizzate per i clienti di un dato intermediario finanziario finalizzate ad una gestione ottimizzata del portafoglio titoli. La metodologia utilizzata è di tipo *machine learning* (ML), con l'applicazione di tre algoritmi basati sugli alberi: *Random Forest*, *Decision Tree* e *Gradient Boosting*, ed analizza tre variabili target: bisogno di liquidità, crescita ed investimento. Il lavoro evidenzia come l'utilizzo di un modello dinamico possa migliorare la qualità della consulenza per l'allocazione del portafoglio titoli, in funzione del profilo di rischio/rendimento dei clienti.

Keywords: *Smart Portfolio Management, Machine Learning, Random Forest, Decision Tree, Gradient Boosting, Regression Analysis.*

1. Introduzione

Questo lavoro si pone come obiettivo quello di sviluppare un sistema di supporto per la realizzazione del trading algoritmico, basato su tecniche di *machine learning*, capace di suggerire al cliente il mix di prodotti finanziari migliori in un'ottica di massimizzazione del profitto derivante dall'investimento, permettendo di simulare differenti scenari di *What If* per la gestione individuale del portafoglio prodotti.

I modelli di *machine learning* sono non parametrici, quindi non postulano una forma funzionale che colleghi la variabile target alle variabili esplicative: il loro punto di forza è infatti l'elevata flessibilità nell'apprendere dai dati, così come la loro elevata capacità predittiva. Tuttavia, essi hanno alcuni punti deboli che sono principalmente rappresentati dal rischio di *overfitting* e dall'interpretabilità dei risultati generati dagli algoritmi. In questo lavoro consideriamo algoritmi di *machine learning* basati su strutture ad albero: *Decision Trees* (alberi di decisione) (Breiman e Friedman (1984)), *Random Forest* (Breiman (2001)) e *Gradient Boosting* (Friedman (2001)).

La metodologia proposta è stata applicata al caso empirico del portafoglio clienti di Poste Italiane. Le fasi principali dello studio sono state dapprima la ricognizione del patrimonio informativo endogeno ed esogeno, a cui è seguita l'individuazione delle variabili necessarie per l'implementazione di modelli di *machine learning*, l'individuazione degli *stakeholders* coinvolti e la comprensione dei limiti dell'attuale modello utilizzato da Poste Italiane.

A tale scopo, è stata condotta un'approfondita analisi descrittiva che ha evidenziato come Poste Italiane, ad oggi, utilizzi di un modello basato su profili e regole statiche senza tener conto del fatto che le esigenze, i comportamenti e la propensione al rischio dei clienti si modificano nel corso del tempo. Un modello dinamico, in cui la profilazione dei clienti cambia nel tempo sulla base di una serie di variabili di cui si registrano i flussi, appare senz'altro più adeguato se si vuole tenere conto in tempo reale dei mutamenti del profilo di rischio dei clienti.

---

[1] Infoedge Technology Srl.

[2] Dipartimento di Scienze Statistiche, Sapienza Università di Roma.

A questa prima fase di analisi è poi seguita una valutazione quantitativa dell'attuale modello di consulenza guidata di Poste Italiane effettuata anche tramite l'impiego di statistiche descrittive. Infine, sono stati implementati i tre modelli di *machine learning* sulla base delle serie storiche relative agli anni 2017-2018 e considerando come variabile target i bisogni di liquidità, crescita ed investimento dei clienti di Poste Italiane.

Il modello proposto può costituire la base metodologica per lo sviluppo di un'applicazione dedicata, con un'interfaccia smart utilizzabile tramite diversi *devices* che potrebbe significativamente migliorare la *customer experience* dei clienti di Poste Italiane.

Il lavoro è organizzato come segue. Nel paragrafo 2 è presentata la metodologia utilizzata con la descrizione dei modelli degli algoritmi di machine learning basati sulla tecnica ad albero. Il paragrafo 3 è dedicato all'applicazione del modello con la presentazione dell'analisi statistica descrittiva del dataset, la descrizione delle variabili e dei risultati dei modelli. Infine, il paragrafo 5 conclude il lavoro.

2. Metodologia

In questo paragrafo vengono illustrate le caratteristiche ed il funzionamento degli algoritmi di *machine learning* utilizzati nello studio: *Decision Trees* (2.1), *Random Forest* (2.2) *e Gradient Boosting Machine* (2.3).

2.1. Decision Trees

Un decision tree (DT), suddivide un dataset iniziale appartenete allo spazio $\mathbb{R}$ in una sequenza di sottosistemi binari, dando così vita ad una struttura ad albero (Hastie e al. (2016)). Anche i sottosistemi vengono a loro volta suddivisi ricorsivamente in nuove regioni. Il riscontro di un'osservazione, in questo modo, viene predetto utilizzando la media ottenuta dalle osservazioni considerate durante la fase di training nella regione alla quale la stessa osservazione appartiene (James e al. (2017)).

Sia $(R_j)_{j \in J}$ una partizione dello spazio $\mathbb{R}$, dove J rappresenta il numero di regioni distinte non sovrapposte. Lo stimatore DT, dato un insieme di variabili $\boldsymbol{x} = x_1, \dots, x_p$, è definito come segue:

$$\hat{f}^{DT}(\boldsymbol{x}) = \sum_{j \in J} \hat{y}_{R_j} \mathbb{I}_{\{\boldsymbol{x} \in R_j\}}$$

dove $\mathbb{I}_{\{.\}}$ è la funzione indicatrice. Le regioni $(R_j)_{j \in J}$ possono essere determinate minimizzando la somma dei quadrati dei residui. La stima della variabile target $\hat{y}_{R_j}$ può essere determinata attraverso i valori medi della variabile che appartiene alla stessa regione R_j.

La grandezza dell'albero è controllata fissando un limite alla crescita dello stesso, in modo tale da evitare che il processo di suddivisione (*splitting*) proceda fino a quando i nodi terminali non diventino nodi puri (un nodo viene definito puro quando tutti i dati appartengono alla stessa classe). Il numero di nodi terminali è rappresentato dal parametro di complessità *pc*. Piccoli valori di *pc* producono alberi di grandi dimensioni, aumentando così il rischio di overfitting, mentre al contrario valori più grandi possono sottostimare la variabile d'interesse.

I DT hanno come vantaggio principale quello di poter essere interpretati facilmente e di essere in grado di catturare qualunque tipo di correlazione nei dati. Tuttavia, vi è una concreta incertezza nel predire i dati e, altresì, piccole fluttuazioni negli input possono produrre alberi molto differenti. La *peformance* predittiva dei DT può essere affinata aggregando diversi alberi di decisione tra di loro, riducendo in questo modo la varianza rispetto ad un singolo albero. Questa tecnica appartiene ai metodi di *ensamble*, tra cui è possibile trovare anche il *Random Forest ed il Gradient Boosting Machine*.

2.2. Random Forest

Il metodo *Random Forest* (RF) consiste nell'aggregazione di diversi DT, ottenuti generando dal dataset originario degli elementi attraverso il boostrap training (Breiman (2001)).

L'idea principale dietro l'algoritmo è quella di generare una perturbazione casuale (*random*) nel sistema di apprendimento, in modo tale da differenziare gli alberi e combinare le loro predizioni attraverso tecniche di aggregazione.

Il metodo è basato sul *«bagging» (bootstrap aggregation)* ma la sua peculiarità risiede nella modalità in cui considera i predittori, la quale permette di prevenire la predominanza di forti predittori nei diversi split di ogni albero (James e al. (2017)). La tecnica utilizzata viene chiamata *out-of-bag* (OOB) e semplifica la stima degli errori de predizioni. La RF è pertanto definita come segue:

$$\hat{f}^{RF}(\boldsymbol{x}) = \frac{1}{B}\sum_{b=1}^{B}\hat{f}^{DT}(\boldsymbol{x}|b)$$

dove B rappresenta il numero di elementi *bootstrap* e $\hat{f}^{DT}(\boldsymbol{x}|b)$ lo stimatore DT sviluppato sul campione (*sample*) b.

Il numero di alberi deve essere scelto in modo tale da massimizzare la percentuale di varianza spiegata e da minimizzare la media dei quadrati dei residui. Il numero deve essere abbastanza grande da consentire ad ogni predittore di essere selezionato (Probst e Boulesteix (2018)).

2.3. Gradient Boosting

Il *Gradient Boosting Machine* (GBM) è un metodo proposto da Friedman (2001) che utilizza gli alberi di decisione di dimensioni fisse come deboli "apprendisti" (*learners*). In questo caso si tratta di un approccio sequenziale, a differenza del RF, in cui l'approccio è di parallelizzazione.

Precisamente, ogni DT utilizza le informazioni del DT precedente per migliorare (*boosting*) l'errore (il gradiente) (Ayyadevara (2018), p.117). In breve, l'algoritmo può essere descritto come segue. Sia F_{m-1} l'attuale modello di *fit*, attraverso il GBM, F_m può essere definito come:

$$F_m(\boldsymbol{x}) = F_{m-1}(\boldsymbol{x}) + \lambda \cdot \gamma_m \cdot h_m(\boldsymbol{x})$$

dove λ è il tasso di apprendimento che ridimensiona il contributo di ogni *weak learner* e $h_m(\boldsymbol{x})$ i *weak learners*, definiti come segue:

$$h_m = -\sum_{i=1}^{p}\nabla_F L(y_i, F_{m-1}(\boldsymbol{x_i}))$$

dove L é la funzione di perdita (*loss function*). Il *weak learner* h_m cerca quindi di minimizzare la funzione L, dato l'ensamble precedente F_{m-1}.

L'accuratezza dell'algoritmo dipende da tre parametri fondamentali: il numero di alberi, la loro profondità (quindi il numero massimo di nodi in ogni albero) e λ, il tasso di apprendimento detto anche "*shrinking*".

3. Applicazione del modello

Il modello è stato applicato al portafoglio clienti di Poste Italiane relativamente ad un campione della popolazione di individui che nel corso del biennio 2017/2018 hanno richiesto una consulenza.

A tal proposito è stata effettuata un'analisi statistica con l'obiettivo di estrarre le informazioni necessarie per la fase di implementazione del modello di *machine learning*. Tale analisi ha inoltre permesso di evidenziare l'inadeguatezza del metodo corrente di consulenza basato, come precedentemente descritto, su un modello di tipo statico.

3.1. Analisi statistica descrittiva del dataset

L'analisi statistica descrittiva preliminare si è innanzitutto incentrata sul piano geografico. Il grafico in *figura 1* riporta il totale delle consulenze (su NDG - codice anagrafico alfanumerico identificativo dell'intestatario – distinti ovvero conteggiando gli individui che si sono recati più volte a ricevere la consulenza nei due anni considerati come singolo individuo) per regione sulla popolazione totale della regione stessa (dati ISTAT al 1° Gennaio 2018) in termini percentuali. Emerge che le regioni con il maggior numero di consulenze (in termini relativi) sono l'Umbria e il Molise (1% della popolazione totale) con rispettivamente 8.920 e 2.908 consulenze osservate. Al contrario, la regione in cui si osservano meno consulenze è il Trentino Alto-Adige con 3.990 consulenze su una popolazione totale al 1° Gennaio del 2018 di 1.067.648 abitanti.

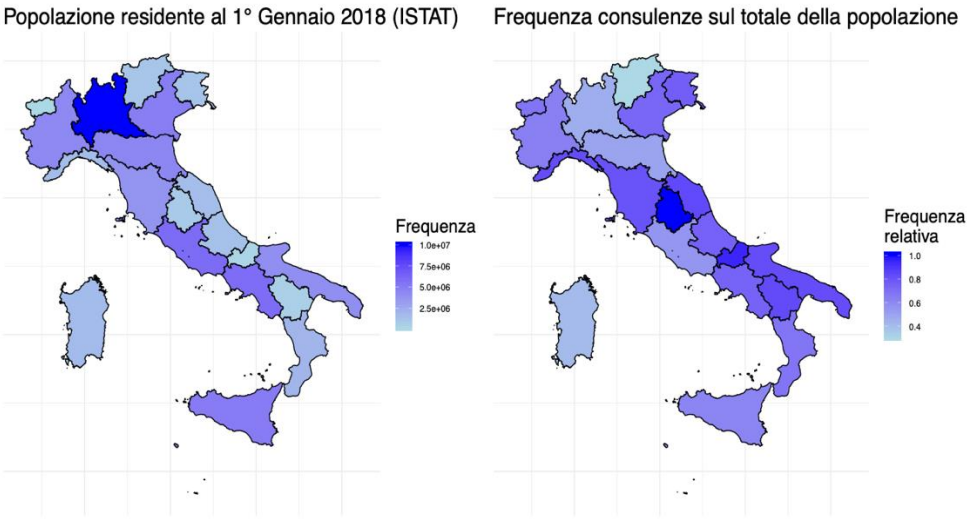

Fig. 1 – Analisi geografica: frequenza relativa delle consulenze

Dalla *figura 2* emerge, invece, come il controvalore delle consulenze (espresso in €), che nel periodo analizzato (2017-2018) ha avuto complessivamente un valore aggregato medio di 2mln € per regione, sia proporzionale al numero di consulenze effettuate: maggiore è il numero di consulenze

più elevato è il giro d'affari. In termini assoluti, il maggior numero di consulenze si è registrato in Lombardia (58.363) e Campania (56.916) regioni in cui si registra anche il maggior giro d'affari, rispettivamente 3.808.074.169 € e 3.402.775.842 € mentre il minor giro d'affari si è registrato in Valle D'Aosta con 66.959.352€. Notiamo a tal proposito che non vi sono apparenti sbilanciamenti in termini di controvalore/consulenza nelle singole regioni.

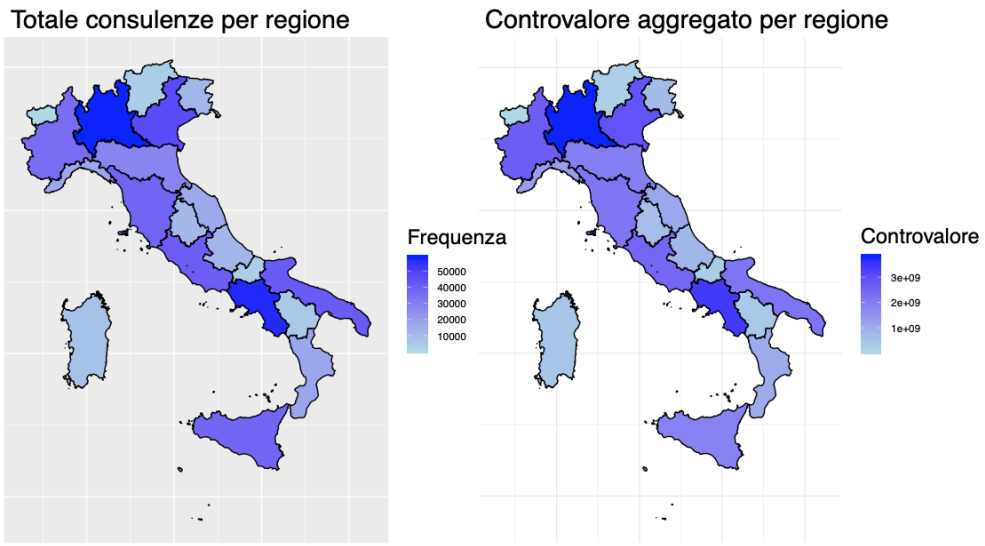

Fig. 2 – Analisi geografica: controvalore delle consulenze

Dalla *tabella 1* si evince che, a meno dei dati mancanti, il modello attuale riconosce:

- L'area MA SUD (dove MA sta per Macro Area) come l'area territoriale meno propensa al rischio (il 31% delle consulenze totali della classe di rischio 1, il 27% della classe di rischio 2 e il 25% della classe 3 sono state osservate in quest'area);
- L'area MA NORD OVEST come l'area territoriale più predisposta a rischiare (le classi di rischio 4 e 5 sono quelle più osservate in quest'area con rispettivamente il 28% e il 26% delle consulenze);

Globalmente, raggruppando le aree in nord, centro e sud Italia è emerso che mentre il sud continua a rimanere l'area meno propensa (classi 1 e 2 più frequenti qui), il centro diventa l'area più rischiosa (classe di rischio 5 più frequente) e il nord come area intermedia (classe di rischio 3 e 4 più frequenti). Osserviamo comunque che le differenze non sono particolarmente significative.

	MA SICILIA	MA SUD	MA CENTRO	MA CENTRO NORD	MA NORD EST	MA NORD OVEST
Classe rischio 1	2.218	5.913	2.822	3.313	2.101	2.860
Classe rischio 2	8.833	25.727	12.587	16.947	11.066	18.479
Classe rischio 3	21.146	70.817	35.146	53.702	37.616	62.896
Classe rischio 4	5.180	17.689	12.730	17.622	11.001	25.192
Classe rischio 5	517	1.582	1.622	2.070	985	2.438

Tab. 1 – Analisi geografica: consulenze (totali) per area territoriale

Considerando la variabile età si è potuto osservare che l'età media più bassa si registra in Campania (53 anni), che abbiamo visto essere la seconda regione per numero di consulenze e giro

d’affari. Al contrario, la regione mediamente più anziana è la Liguria (60 anni). Globalmente, da uno sguardo alla mappa emerge che i soggetti che effettuano la consulenza sembrerebbero essere più anziani nelle regioni del Nord Italia e più giovani nelle regioni del Centro-Sud. Dunque, in termini di consulenza, il Sud Italia emerge come territorio mediamente più giovane e meno propenso a rischiare. In aggiunta, la classe che ricorre maggiormente alla consulenza è quella tra i 65 e i 70 anni d’età; viceversa le classi di età che ricorrono meno sono quelle con età inferiore a 20 anni e superiore ad 85. La distribuzione della variabile età, evidenzia che l’età minima osservata è di 3 anni mentre l’età massima osservata è pari a 105. Il primo quartile e il terzo ci suggeriscono che il 50% degli utenti osservati hanno un'età compresa tra i 46 anni e i 69 anni mentre in media i clienti che ricorrono alla consulenza hanno un’età di 57 anni.

La *figura 3* sulla ripetibilità per regione riporta i clienti che si sono presentati alla consulenza per più di due volte. Osservando il grafico sulla ripetibilità per regione non si nota un particolare discostamento rispetto a quanto osservato sul totale delle consulenze per regione, da ciò ne deriva che le regioni in cui vengono fatte più consulenze, sono anche quelle in cui si è registrata la maggior ripetibilità.

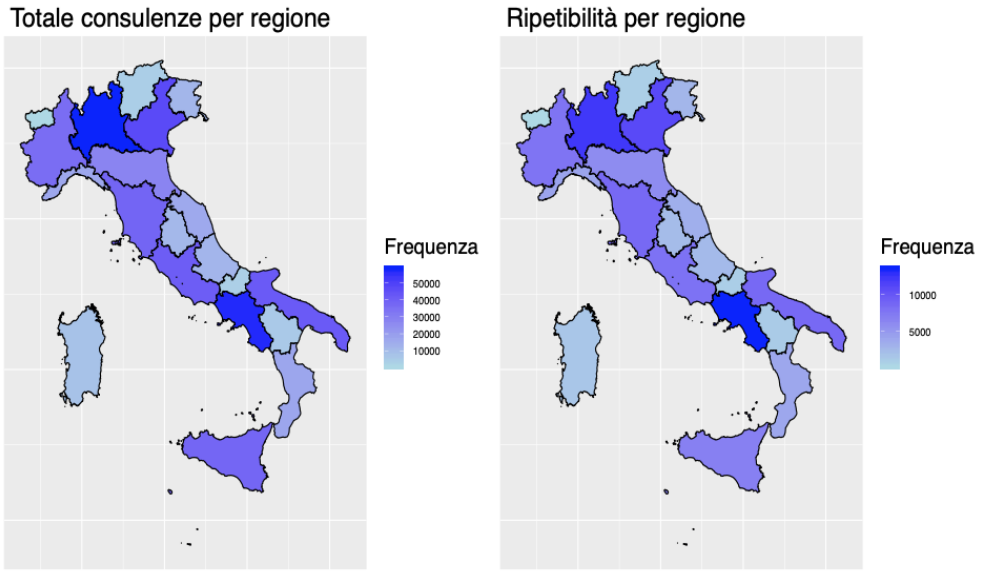

Fig. 3 – Analisi geografica: ripetibilità per regione

Questo scenario potrebbe cambiare notevolmente con lo sviluppo di un’apposita applicazione su *smartphone* o altri *devices* per effettuare la consulenza, che renderebbe maggiormente fruibile la consulenza comportando verosimilmente un aumento non solo del numero di consulenze, ma anche del ripetersi di queste per singolo individuo sull’intero territorio nazionale.

Analizzando più nel dettaglio la ripetibilità per regione, è risultato interessante notare che nel Sud Italia la ripetibilità è stata più consistente rispetto al Nord nel corso dei due anni (2017-2018). La Campania è risultata ragione maggiormente propensa alla ripetibilità della consulenza. Qui 535 consulenze sono di soggetti che si sono presentati più di 10 volte. Tuttavia, la ripetibilità massima è stata registrata in Puglia (un soggetto ha effettuato la consulenza 52 volte). La regione con la minor ripetibilità è invece la Valle d’Aosta: si tratta dell’unica regione in cui i clienti non si sono recati a richiedere la consulenza più di 6 volte.

Ancora sulla ripetibilità, il 10% delle consulenze (56.686 su 544.093) sono di clienti che hanno effettuato le consulenze 2 volte. La ripetibilità massima osservata sui due anni (2017-2018) è di un

cliente che si è recato a fare la consulenza 52 volte. Analizzando questo individuo è emerso che si tratta di un soggetto in Puglia con 45 anni d'età, con un profilo di rischio medio (classi di rischio 2 e 3) e un patrimonio medio di 56.414,49 € (inferiore al patrimonio medio osservato su tutti gli NDG che risulta pari a 95.347,32 €).

Misurando poi la ripetibilità per classi di età dall'analisi risulta che su 544.093 consulenze osservate, solo 116.865 (21,48%) sono di soggetti che si sono presentati più di una volta. Nel grafico in *figura 4* ci si è focalizzati su queste 116.865 consulenze. In particolare, sono stati presi tutti i clienti ricorsi alla consulenza più di una volta e le frequenze sono state normalizzate tra 0 ed 1 in modo da essere confrontabili. Si evince che per le classi di età estreme la ripetibilità non è presente. Viceversa, nelle classi d'età centrali la ripetibilità è più ricorrente. Nella classe di età maggiormente ricorsa alla consulenza (65-70 come visto nella slide precedente) notiamo che degli 11.931 clienti ricorsi alla consulenza più di una volta, circa il 60% (7.303) si è recato esattamente 2 volte, mentre il 6% di questi (653) si è presentato più di 5 volte.

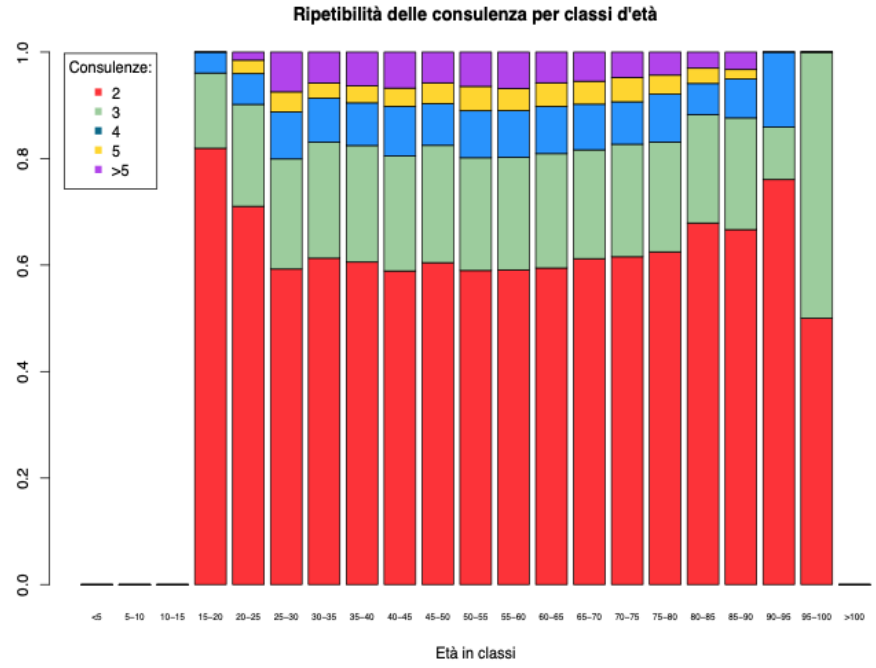

Fig. 4 – Analisi della ripetibilità: ripetibilità della consulenza per classi di età

Il 56,7% delle consulenze (308.415 su 544.093) è stato osservato con riferimento alla terza classe di rischio. Trattandosi della classe intermedia era lecito attendersi questa collocazione. Al contrario, la classe di rischio in cui si è osservato il minor numero di consulenze è stata la quella per soggetti maggiormente propensi al rischio (ovvero la classe 5), con appena l'1,78% (9.512) delle consulenze totali. Per quanto riguarda un'analisi approfondita sulle classi di rischio è, innanzitutto, emerso che minore è l'età maggiore è la classe di rischio associata. I valori medi e i quantili nella *tabella 2* ci suggeriscono, inoltre, che le prime due classi di rischio non sono particolarmente discriminate dall'età (caratteristiche molto simili). Tuttavia, possiamo affermare che tendenzialmente il modello attuale riconosce i più giovani come più propensi al rischio rispetto ai meno giovani.

	CLASSE 1	CLASSE 2	CLASSE 3	CLASSE 4	CLASSE 5
N	22.230	109.382	308.415	94.554	9.512
MANCANTI	0	0	2	0	0
MEDIA	59	59	56	54	52
DEV. STD.	17,60	17,49	14,98	14,16	12,77
Q1	46	46	45	45	41
MEDIANA	61	65	57	56	54
Q3	74	75	68	65	61
MINIMO	8	3	7	18	19
MASSIMO	99	103	99	102	77

Tab. 2 – Analisi delle classi di rischio: ripartizione dell'età per classe di rischio

Un'analisi descrittiva sul controvalore è stata poi effettuata ed ha evidenziato l'assenza di dati mancanti per questa variabile. Il valore minimo osservato del controvalore è di 35,03 € contro un massimo di 15.618.186 €. Da ciò deduciamo immediatamente che il range di questa variabile è particolarmente elevato e dunque la variabilità è molto alta. I quantili della distribuzione ci suggeriscono a tal proposito che al 50% delle consulenze associamo un controvalore compreso tra 7.850 € e 74.966,82 €. Il controvalore medio risulta essere pari a 63.084,30 €. In realtà, data la presenza di valori particolarmente alti del controvalore (i tre controvalori massimi osservati sono stati pari a 15.618.186 €, 7.676.743 € e 5.676.681 €), per avere un'idea rappresentativa della media si è utilizzato un indice robusto agli *outliers*, ovvero una media *trimmed* al 5%. Eliminando il 5% dei controvalori su entrambe le code della distribuzione, si è ottenuta una media di 47.301,54 €. Il giro d'affari su tutti e due gli anni ammonta complessivamente a 34.324.012.814€. Distinguendo sui due anni notiamo che, dato il maggior numero di consulenze, come lecito aspettarsi il giro d'affari è maggiore nel 2018 (371.512 consulenze) rispetto al 2017 (172.581). Mediamente nel 2017 si è osservato un controvalore di 43.553€ contro i 72.158 € del 2018.

Anche in questo caso è stata fatta una distinzione per classe di rischio. Innanzitutto, dalla *tabella 3*, notiamo che non sono presenti valori mancanti. Nonostante la classe di rischio 5 sia quella con il numero più basso di consulenze è anche quella a cui in media è associato il maggior giro d'affari (92.257 €). Guardando alla media semplice, sembrerebbe che per la classe di rischio 2 e la classe di rischio 3 non vi siano grandi differenze in termini di controvalore. In realtà ciò è riconducibile alla presenza nella classe di rischio 2 del controvalore massimo pari a 15.618.186 € (*outlier*). La media trimmed al 5% evidenzia, infatti, una differenza più significativa tra le medie. I controvalori osservati presentano un'elevata variabilità.

	CLASSE 1	CLASSE 2	CLASSE 3	CLASSE 4	CLASSE 5
N	22.230	109.382	308.415	94.554	9.512
MANCANTI	0	0	0	0	0
MEDIA	36.468	59.320	60.903	77.879	92.257
MEDIA TRIMMED 5%	25.338,55	42.907,53	45.881,80	60.745,71	72.236,98
DEV. STD.	74.405,56	117.254,3	105.096,5	121.500,3	137.010,9
Q1	1.643	5.147	7.887	13.457	15.768
MEDIANA	13.782	25.008	29.651	41.015	46.495
Q3	42.071	68.090	72.915	95.522	114.648
MINIMO	50	38	36	35	50
MASSIMO	2.377.780	15.618.186	7.676.743	3.783.180	3.130.562

Tab. 3 – Analisi del controvalore: ripartizione del controvalore per classe di rischio

Sulla data di creazione della consulenza, distinguendo le consulenze sui due anni, è emerso che il numero totale di consulenze nel 2017 è stato pari a 172.581 contro le 371.512 del 2018, ad

indicare un aumento di circa il 115.27% in quest'ultimo anno rispetto al primo. Distinguendo per mese, il mese con il maggior numero di consulenze è stato giugno nel 2017 con 26.706 consulenze contro le 68.670 di novembre nel 2018. Cumulativamente sui 2 anni, invece, notiamo che i mesi in cui i clienti ricorrono maggiormente alla consulenza sono ottobre e novembre con rispettivamente 85.897 e 85.911 consulenze. La presenza di valori particolarmente bassi per i primi mesi del 2017 è giustificabile in virtù di un sistema non ancora maturo (non tutti gli uffici erano ancora abilitati al *MiFiD*). Il sistema si consolida a partire da Maggio 2017. Verso la fine dell'anno i clienti sembrerebbero essere maggiormente propensi alla consulenza rispetto agli altri mesi ma comunque non è emersa alcuna stagionalità nei dati analizzati.

Concludiamo l'analisi statistica descrittiva osservando che i 5 grafici in figura dimostrano come, per ogni classe di rischio il modello statico utilizzato da Poste Italiane sottostimi il bisogno di investimento dei clienti del 13% (media totale) e sovrastimi il loro bisogno di crescita dell'8% (media totale). Ciò denota un forte potenziale di miglioramento da parte del modello di profilazione, che suggerendo un bisogno di investimento più adeguato potrebbe incrementare il potenziale di rendimento finanziario del cliente. In particolare, in riferimento al bisogno 1 (liquidità) il concordato è sempre maggiore, in media, rispetto a quanto suggerito dal modello per ogni classe di rischio. Gli utenti tendono quindi a tenere in liquidità più di quanto il modello gli consiglia. Sul bisogno 2 (crescita) il modello suggerisce di allocare mediamente la stessa somma per le classi di rischio 4 e 5, mentre i valori si discostano per le prime 3 classi di rischio. In relazione a questo bisogno, i clienti nelle prime due classi di rischio allocano un controvalore prossimo a quello suggerito dal modello di Poste Italiane. Diverso è il caso delle classi di rischio 3, 4 e 5 dove il modello sovrastima notevolmente il controvalore effettivamente allocato dai clienti. Sull'ultimo dei bisogni (investimento) il controvalore cresce al crescere della classe di rischio. Il modello suggerisce, in media, un controvalore inferiore a quanto effettivamente investito dai clienti.

In sintesi, dall'analisi statistica emerge che lo scostamento tra il controvalore concordato e quello suggerito dalla consulenza è spesso di segno positivo. Dunque, il bisogno di investimento (e di crescita) non è pienamente colto dall'attuale sistema di consulenza guidata: il cliente mediamente investe più risorse di quanto suggerito dalla consulenza guidata. Poiché l'analisi statistica preliminare ha evidenziato uno scostamento tra quanto viene suggerito ai clienti dal modello attuale e quanto effettivamente concordato con i consulenti questo si traduce nell'impossibilità, oggi, di anticipare quelli che sono gli effettivi bisogni/desideri dei clienti di Poste Italiane da parte della consulenza. Per questo motivo si è proceduto allo sviluppo di un modello di *machine learning* finalizzato al trading algoritmico

3.2. Descrizione delle variabili e risultati dei modelli

Le variabili individuate in base all'analisi statistica ed utilizzate per l'implementazione degli algoritmi di *machine learning* sono state in totale 20, escludendo a priori tutte le variabili con potere predittivo irrilevante. L'elenco delle variabili utilizzate è riportato nella *tabella 4*.

DT_CREAZIONE_CONSULENZA	TIPOLOGIA	SOTTOTIPO
CONTROVALORE	LIQUIDITA_DA_CC	LIQUIDITA_DA_LIBRETTI
LIQUIDITA_DA_NL	CD_PROFILO	STRATEGIA
CLUSTER_NDG	CODICE_PROFILO_EC	ETA_NDG
PATRIMONIO_TOTALE_NDG	REGIONE	CTV_ATTUALE_BIS1
CTV_ATTUALE_BIS2	CTV_ATTUALE_BIS_3	

Tab 4 – la master table: l'elenco dei predittori

Le variabili relative ai bisogni di liquidità, crescita e investimento sono state utilizzate come variabile (trivariata) dipendente del modello, mentre le rimanenti 17 hanno costituito l'insieme dei predittori. Stante il numero di predittori sufficientemente ridotto, non sono state ritenute necessarie operazioni di riduzione di dimensionalità (es. Analisi in Componenti Principali o Analisi Fattoriale). Nell'ambito del *machine learning* tali procedure sono generalmente utilizzate, non tanto per selezionare le variabili ritenute più significative, quanto per migliorare i tempi di esecuzione del modello (l'algoritmo restituisce l'importanza delle variabili di per sé).

Il modello di *machine learning* è stato implementato sfruttando le seguenti librerie su Python3:

- ***NumPy*** (statistiche, array, matrici …)
- ***Pandas*** (lettura dati, manipolazione dati …)
- ***Scikit-Learn*** (algoritmi di machine learning, modelli…)

È stato svolto un lavoro di *pre-processing* imputando i dati mancanti (ovvero sostituendo i *missing values* con la moda per le variabili categoriche e con una media aritmetica per quelle continue), normalizzando il *dataset* (l'*outcome* è stato ridefinito in modo tale che la somma dei controvalori concordati dei tre bisogni fosse pari ad 1) e ricodificando le variabili categoriche mediante il metodo del *OneHot-Encoding* (*splitting* delle variabili categoriche in tante variabili *dummy* dando luogo alle categorie della variabile stessa facendo così salire in numero delle variabili del dataset a 51). Tutto questo lavoro ha permesso di migliorare notevolmente le prestazioni dell'algoritmo in termini sia di efficacia che di efficienza. Osserviamo inoltre che dato l'elevato numero di dati a disposizione si è inoltre suddiviso il *dataset* in *training set* (70%) e *validation set* (30%).

Dapprima è stato applicato l'algoritmo *Random Forest*, che è stato poi confrontato con altre due metodologie basate sugli alberi: *Decision Tree* e *Gradient Boosting*. Al fine di individuare il modello ottimale in termini di riduzione dell'errore (*Mean Squared Error – MSE*), sono state effettuate una serie di simulazioni volte a calibrare opportunamente i parametri di configurazione. In particolare, il *tuning* degli iper-parametri ci ha permesso di individuare che valori dare al modello con riferimento a:

- Numero di alberi: indica il numero di alberi della foresta. Generalmente maggiore è il numero di alberi, e migliore sarà l'algoritmo nell'apprendimento (a discapito di un peggioramento nelle performance del modello in termini di tempo di esecuzione). La scelta del numero ottimale di alberi è stata effettuata graficamente in base al valore dell'MSE: dal grafico a fianco emerge che l'MSE decresce rapidamente all'aumentare del numero di alberi da 0 a 15, meno rapidamente fino a 50 e poi si stabilizza. Per questa ragione si è fissato il numero di alberi a 50 (parametro *n_estimators* in Python).
- Profondità massima: indica la profondità massima di ciascun albero di cui è composta la foresta. Ci si aspetta che più profondo è l'albero, maggiore è il numero di split effettuati, e maggiore è l'informazione catturata dal modello. Anche qui la scelta è stata effettuata graficamente attraverso l'MSE. In questo caso è opportuno scegliere il valore del parametro sia considerando il valore a partire dal quale l'MSE si stabilizza, sia alla distanza tra le due curve (maggiore è la distanza, maggiore è l'overfitting). Il trade-off ottimale tra queste due

misure si raggiunge in corrispondenza di una profondità pari a 15 (parametro *max_depth* in Python).

Confrontando i risultati ottenuti con i diversi parametri (*Grid Search*), si è poi deciso di fissare gli iper-parametri del modello a quelli sopra individuati: numero di alberi=50 e profondità=15.

Prima di testare il modello sul *test set*, per validare la scelta dei parametri selezionati per l'implementazione del modello finale, si è effettuata una *K-Fold Cross Validation*. Questa tecnica consiste nel testare le performance del modello sul set di addestramento prima di effettuare il test definitivo sul *test set* al fine di evitare che risultati apparentemente buoni siano frutto del caso (Data Leakage, Shabtai e al. (2012)) ed, eventualmente, di apportare eventuali aggiustamenti ai parametri del modello. Nel caso specifico, il *training set* è stato suddiviso in 10 set di dati (*10-Fold Cross Validation*) e su questi è stato stimato il modello *Random Forest* (profondità=15 e numero di alberi=50). Su tutti e 10 i *sub-sets* il modello ha performato in modo pressoché identico in termini di *score* (R^2) ottenendo mediamente un valore di 0,91. Il modello selezionato può, dunque, essere ritenuto ragionevole ed essere esteso all'utilizzo di dati ancora non osservati.

In seguito alla calibratura dei parametri, è stato testato il modello. Mediamente il valore previsto dal modello si discosta dal valore vero di 0,008 (margine d'errore pressoché nullo). Il modello stesso è inoltre in grado di cogliere il 91% della variabilità totale dell'*outcome* misurato. L'analisi è stata ripetuta eliminando tutte le variabili irrilevanti, ottenendo gli stessi risultati ma riducendo notevolmente la complessità del modello, dato il minor numero di parametri da dover stimare.

L'analisi è stata poi effettuata usando come i modelli *Decision Tree* e *Gradient Boosting* (ricercando anche qui la combinazione ottimale di parametri che nel caso del Decision Tree ha riguardato esclusivamente la scelta della profondità mentre per il Gradient Boosting ha interessato anche il *learning rate* e il numero di stimatori). I risultati sono stati confrontati con quelli ottenuti con l'algoritmo *Random Forest* e date le prestazioni inferiori rispetto a quest'ultimo, sia in termini di varianza spiegata/errore quadratico medio ma anche in termini di tempo (nel caso del Gradient Boosting), si è scelto in via definitiva il metodo *Random Forest*.

Concludiamo osservando i grafici riportati in figura 5 dove abbiamo rappresentato il controvalore medio concordato, suggerito dal modello statico di Poste Italiane, e quello suggerito dal nostro modello di *machine learning* (SBC – Simulatore Bisogni Concordati) distintamente per mese (il controvalore medio è stato normalizzato così che la somma sui tre bisogni sia pari ad 1). A titolo esemplificativo, sul bisogno di crescita di dicembre notiamo che se mediamente il controvalore concordato è stato pari a 0,429 e Poste Italiane ha suggerito 0,648, il modello SBC avrebbe suggerito al cliente un valore pressoché identico a quello effettivamente stabilito, ovvero 0,436. In sintesi, il modello SBC coglie i bisogni dei clienti con molta più precisione rispetto al modello attualmente adottato.

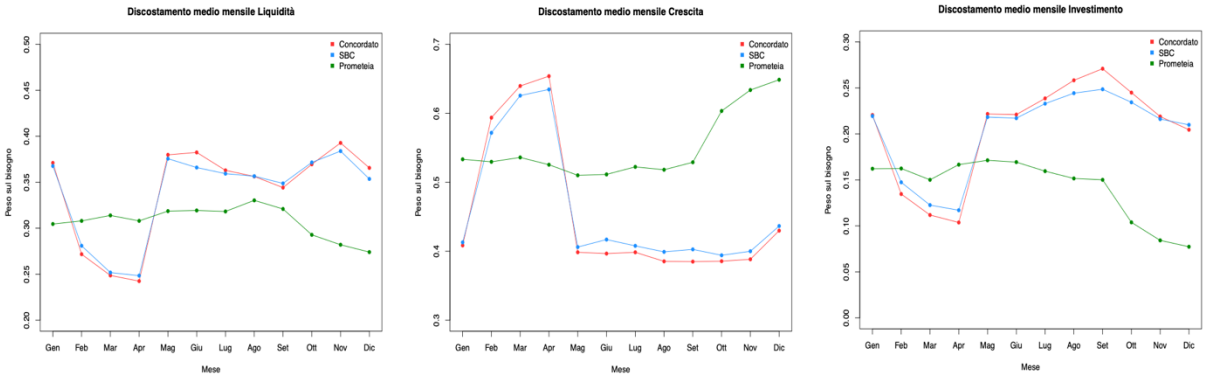

Fig. 5 – Margine d'errore del modello di machine learning

4. Conclusioni

Lo studio ha evidenziato come i modelli di *machine learning*, in particolare l'algoritmo *Random Forest*, siano in grado di riprodurre con un elevato grado di accuratezza i valori della variabile output, ovvero i controvalori concordati dei tre bisogni (liquidità, crescita ed investimento), delle serie storiche del biennio 2017-18, aprendo così il campo all'introduzione di un modello dinamico atto a garantire il miglioramento delle performance e della *customer experience* del portafoglio clienti di Poste Italiane. Sotto questo punto di vista, un plausibile modello dinamico da implementare potrebbe essere strutturato nelle seguenti fasi:

1) Profilazione a priori: sulla base di variabili note all'ingresso (età, propensione al rischio da questionario *MiFiD*).
2) Analisi del comportamento finanziario del cliente e del mercato finanziario.
3) Profilazione a posteriori: sulla base delle variabili di cui al punto 2).
4) Definizione del portafoglio modello: problema di allocazione delle risorse del cliente (asset allocation) che si esprime matematicamente come un problema di ottimo vincolato: si esplicita una funzione obiettivo e si traducono i vincoli in specifiche equazioni e disequazioni.
5) Assegnazione di un portafoglio modello in funzione del profilo del cliente.

Il bisogno di liquidità dovrebbe inoltre essere fissato (soglia minima) in funzione della giacenza media del conto corrente, mentre i bisogni di crescita e investimento saranno stabiliti assegnando un portafoglio di strumenti finanziari secondo aliquote fissate in funzione del profilo.

Infine, in termini di vantaggi per il cliente, adottando un sistema di raccomandazione dinamico basato sull' *intelligenza artificiale* contro l'attuale modello di consulenza guidata di tipo statico basato su regole fisse, è possibile ottenere una più adeguata allocazione del portafoglio titoli nonché una migliore *customer experience* raggiungibile anche tramite l'implementazione di una applicazione dedicata, fruibile tramite *smartphone*, *laptop* ed altri *devices* di ultima generazione.

Bibliografia

1. Ayyadevara V.K. (2018) Gradient Boosting Machine. In: Pro Machine Learning Algo-rithms. Apress, Berkeley, CA
2. Breiman, L., Random forests.Machine learning45(1): 5–32. (2001)

3. Breiman, L., Friedman, J., Olshen, R. and Stone C., Classification and regres-sion trees. Chapman & Hall/CRC, Boca Raton, Florida. (1984)
4. Friedman, J.H., Greedy function approximation: A Gradient Boosting Machine. Annals ofStatistics 29: 1189–1232 (2001).
5. Hastie, J., Hastie, T., Tibshirani, R. 2016.The Elements of Statistical Learning, 2nd ed.Data Mining, Inference, and Prediction. New York: Springer. ISBN 10: 0387848576.
6. James, G., Witten, D., Hastie, T., Tibshirani, R., An Introduction to Statistical Learning: with Applications in R. Springer Texts in Statistics. ISBN 10: 1461471370. (2017)
7. Liaw,A.,Packagerandomforest.https://cran.rproject.org/web/packages/randomForest/randomForest.pdf. (2018)
8. Mehra, C. S., Prugel-Bennett, A., Gerding, E., Robu, V. (2014). Constructing smart portfolios from data driven quantitative investment models. 2014 IEEE Conference on Computational Intelligence for Financial Engineering & Economics (CIFEr), London, 2014, pp. 166-173, doi: 10.1109/CIFEr.2014.6924069.
9. Mehra, C. S. (2016). Constructing smart financial portfolios from data driven quantitative investment models. *University of Southampton, Faculty of Physical Science and Engineering, Doctoral Thesis*, 261pp.
10. Oshiro, T.M., Perez, P.S., Baranauskas, J.A. (2012) How Many Trees in a Random Forest?. In: Perner P. (eds) Machine Learning and Data Mining in Pattern Recognition. MLDM 2012. Lecture Notes in Computer Science, vol 7376. Springer, Berlin, Heidelberg.
11. Probst, P., Boulesteix, A.L. (2018) To Tune or Not to Tune the Number of Treesin Random Forest.Journal of Machine Learning Research, 18: 1-18.
12. Ridgeway, Greg. 2007. Generalized Boosted Models: A Guide to thegbmPackage.https://cran.r-project.org/web/packages/gbm/gbm.pdf(accessed on 21 May 2018).
13. Shabtai, A., Elovici, Y., Rokach, L. (2012). A Survey of Data Leakage Detection and Prevention Solutions. SpringerBriefrs in Computer Science. doi: 10.1007/978-1-4614-2053-8_1
14. Therneau, T. M., Atkinson, E. J. et al. (2017). An introduction to recur-sive partitioning using the RPART routines. https://cran.r-project.org/web/packages/rpart/vignettes/longintro.pdf