Relazione SBD - Progetto

Introduzione

Lo studio da noi introdotto si basa sul dataset SN182-Indagine sui bilanci delle famiglie italiane (2014) della Banca d'Italia, focalizzandoci sugli aspetti chiave delle abitudini finanziarie di queste ultime. Le variabili che abbiamo selezionato sono:

- **Y.x** : variabile dipendente principale nel nostro studio, legata al reddito disponibile netto;
- **carta** : variabile indipendente, riferita al possesso di carta di credito;
- **bancomat** : variabile indipendente, riferita all'utilizzo di carte di debito;
- **cartapre** : variabile indipendente, riferita al possesso di carte prepagate;
- **spesecon** : variabile indipendente, riferita alla spesa mensile abituale in contanti.

Queste variabili sono state scelte in particolare per analizzare la diffusione degli strumenti di pagamento elettronici e non (**spesecon**) e l'impatto potenziale che essi hanno sul reddito disponibile netto (**Y.x**).

ANALISI REGRESSIONE LINEARE SEMPLICE

Obiettivo dell'analisi

L'obiettivo di questa analisi è capire come alcuni **strumenti di pagamento** (come carte di credito, prepagate, bancomat, contanti) influiscano sul **reddito netto** delle persone.

Per poter fare ciò, si è applicata un **modello di regressione lineare semplice** su ogni variabile (*reddito netto = variabile dipendente, strumenti di pagamento = variabile indipendente*)

Dati

I dati che sono stati utilizzati per l'applicazione di questo modello sono stati raccolti dentro una ***copia*** del dataset originale, chiamata **data.EX.** Dalla quale sono state eliminate tutte le osservazioni con valori mancanti (NA), per garantire una qualità statistica dell'analisi.

I dati in questione sono :

- **Y.x** : *reddito netto*
- **carta:** *possesso carta di credito*
- **bancomat**: *possesso bancomat*
- **cartapre**: *possesso carta prepagata*
- **spesecon**: *spese effettuate con contanti*

Costruzione dei modelli

Per ciascuna variabile sono stati creati dei modelli di regressione tramite il comando **model_EX**:

```
model_EX = lm(formula = Y.x ~ carta, data = data.EX )
```

ognuno di questi comandi stima il reddito netto in funzione di una diversa variabile esplicativa, ovvero con questo modello vogliamo capire come il tipo di carta posseduta possa influenzare il reddito degli individui.

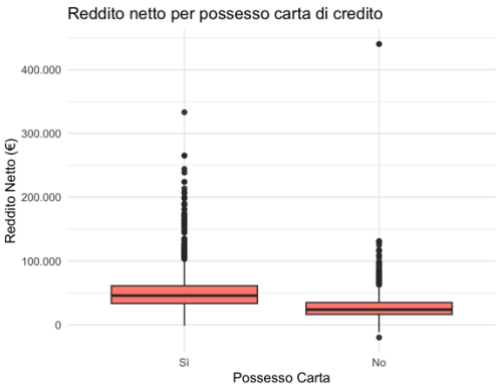
Visualizzazione dei risultati

Per ciascun modello sono stati realizzati dei grafici di dispersione (**boxplot**), attraverso il comando

```
ggplot(data.EX, aes(x = factor(carta), y = Y.x)) +
  geom_boxplot(fill = "tomato", alpha = 0.7) +
  labs(x = "Possesso Carta", y = "Reddito Netto (€)", title = "Reddito netto per possesso carta di credito") +
```

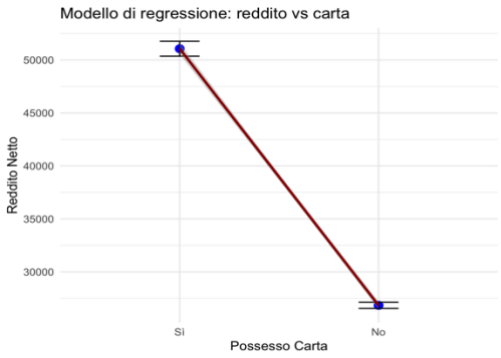
Relazione SBD - Progetto

```
  scale_x_discrete(labels = c("1" = "Sì", "2" = "No")) +
  scale_y_continuous(labels = scales::label_comma(big.mark = ".", decimal.mark = ",")) +
  theme_minimal()
```



Chi possiede una carta di credito ha in media un reddito più alto.
La variabilità del reddito tra i possessori è maggiore, come suggerisce la larghezza del box e la presenza di outlier.
Chi non possiede la carta ha redditi più bassi e distribuzione più compatta, con meno valori estremi.

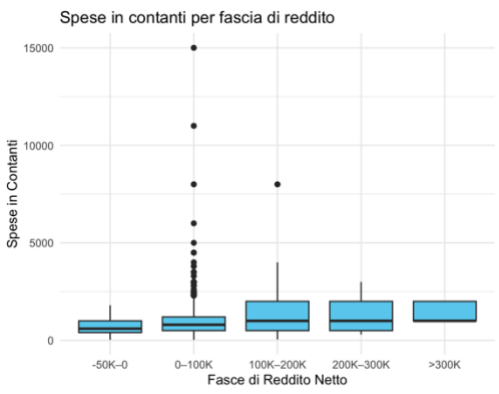
```
ggplot(data.EX, aes(x = factor(carta), y = Y.x)) +
  stat_summary(fun = mean, geom = "point", size = 3, color = "blue") +
  stat_summary(fun.data = mean_cl_normal, geom = "errorbar", width = 0.2) +
  geom_smooth(method = "lm", se = TRUE, aes(group = 1), color = "darkred") +
  labs(x = "Possesso Carta", y = "Reddito Netto", title = "Modello di regressione: reddito vs carta")+
  scale_x_discrete(labels = c("1" = "Sì", "2" = "No")) +
   theme_minimal()
```



Questo grafico rappresenta visivamente una regressione lineare che mette in relazione il possesso di una carta di credito con il reddito netto.
I punti blu indicano il reddito medio per ciascun gruppo.
Le barre nere rappresentano l'intervallo di confidenza attorno a quei valori medi.
Chi possiede una carta di credito ha un reddito medio più alto (~€50.000) rispetto a chi non la possiede (~€30.000).
Il netto salto verso l'alto suggerisce una differenza significativa tra i due gruppi.
L'intervallo di confidenza è stretto, quindi la stima è affidabile.

Abbiamo fatto lo stesso anche per la variabile bancomat e carta prepagata e i risultati sono praticamente gli stessi. Ora analizziamo la relazione tra reddito netto e spesa in contanti.

```
breaks <- c(-50000, 0, 100000, 200000, 300000, 500000)
labels <- c("-50K–0", "0–100K", "100K–200K", "200K–300K", ">300K")
data.EX$reddito_cat <- cut(data.EX$Y.x,  breaks = breaks,labels = labels,)
ggplot(data.EX, aes(x = reddito_cat, y = spesecon)) +
  geom_boxplot(fill = "skyblue") + labs(x = "Fasce di Reddito Netto", y = "Spese in Contanti",
    title = "Spese in contanti per fascia di reddito") +
   theme_minimal()
```



In questo grafico notiamo che all'aumentare del reddito, la capacità di spendere in contanti aumenta, ma anche le abitudini diventano più eterogenee. C'è più dispersione e meno uniformità nei comportamenti di spesa.

Relazione SBD - Progetto

Nella seconda fascia notiamo che ci sono molti individui che spendono in contanti più della media del loro gruppo. Questo può indicare persone con stili di vita particolarmente "cash-oriented o persone che evitano strumenti elettronici.

Per ciascun modello è stato esaminato il sommario statistico prodotto da ***summary ()***. Questo comando fornisce una panoramica dettagliata del modello di regressione corrispondente.
In particolare, si concentra sul p-value, i coefficienti stimati e gli errori standard, così da poter analizzare al meglio la relazione tra le variabili.

Calcolo indici di bontà

Per spiegare al meglio la variabilità della nostra variabile dipendente in funzione di quella indipendente possiamo calcolare e analizzare i seguenti indici:

- **SSTOT (DEVIANZA TOTALE):** va a misurare la variabilità totale di Y.x rispetto alla sua media, ovvero rappresenta la totale dispersione attorno la media.
- **SSR (DEVIANZA SPIEGATA):** Rappresenta la **parte di variabilità** che il modello riesce a spiegare tramite la variabile indipendente. Più è alta, meglio il modello sta funzionando.
- **SSE(DEVIANZA RESIDUA):** Misura ciò che il modello **non riesce a spiegare**. È la variabilità "rimasta fuori" dal modello.
- **R^2(COEFF. DI DETERMINAZIONE):** È un valore che fornisce indicazioni riguardanti la bontà di adattamento di un modello statistico ai dati. È compreso tra 0 e 1 e si calcola con: **R^2=SSR/SSTOT**

In sintesi: questi indici servono a **quantificare la bontà del modello**, quindi ci suggeriscono se ci conviene o meno usare una determinata variabile esplicativa in questo caso carta, per prevedere il reddito netto.

ANALISI REGRESSIONE MULTIPLA

Obiettivo

L'obiettivo di questa sezione è analizzare come il reddito netto sia influenzato da tre variabili esplicative di tipo categorico, tenendo conto delle altre variabili.
Le tre variabili in questione sono:

- **carta** (possesso o meno di una carta di credito);
- **bancomat** (possesso o meno di un bancomat);
- **cartapre** (possesso o meno di una carta prepagata).

Pulizia e preparazione del Dataset

Abbiamo innanzitutto creato una copia del dataset originale chiamata data.RM, rimuovendo eventuali osservazioni contenenti valori mancanti (NA) per le variabili di interesse:

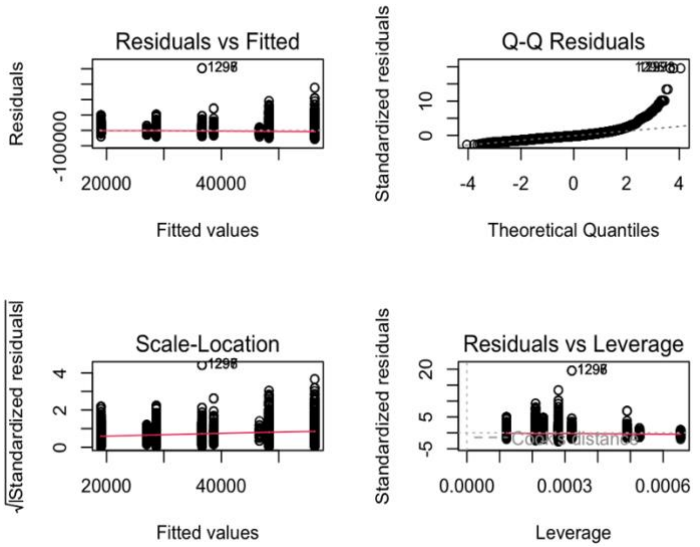
```
data.RM <- na.omit(m4[c("Y.x","bancomat","cartapre","carta")])
```

Interpretazione dei coefficienti

Dalla funzione summary(mod_multiple) otteniamo i coefficienti stimati, che ci indicano quanto varia il reddito netto medio al variare di ciascuna variabile esplicativa. Da qui riusciamo a individuare anche R^2 aggiustato che è una stima più affidabile della bontà del modello, soprattutto quando stai facendo un confronto tra modelli con un diverso numero di predittori. Nel nostro caso essendo pari a

Relazione SBD - Progetto

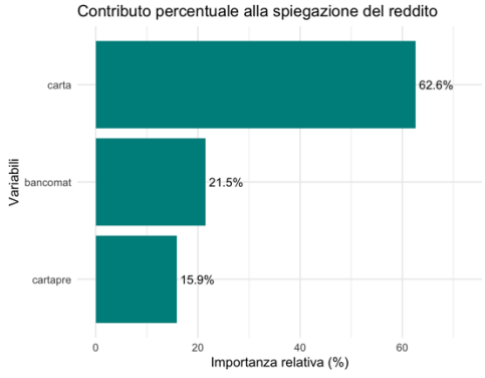
0,26 significa che il tuo modello di regressione riesce a spiegare il 26% della variabilità della variabile dipendente, tenendo conto del numero di variabili indipendenti utilizzate.
par(mfrow = c(2, 2))
plot(mod_multiple)
Con queste operazioni stiamo ottenendo i quattro grafici diagnostici fondamentali per valutare la bontà del tuo modello di regressione lineare multipla.



Residuals vs Fitted Controlla la linearità e l'omogeneità della varianza.
Normal Q-Q Verifica se i residui seguono una distribuzione normale (i punti dovrebbero seguire la diagonale).
Scale-Location Controlla l'omoschedasticità, ovvero se la variabilità dei residui è costante.
Residuals vs Leverage Aiuta a identificare punti influenti (outlier o osservazioni con forte impatto sul modello).

```
rel_imp <- calc.relimp(mod_multiple, type = "lmg",
rela = TRUE)

rel_df <- as.data.frame(rel_imp@lmg)
rel_df$Variabile <- rownames(rel_df)
colnames(rel_df)[1] <- "Importanza"
rel_df$Importanza <- rel_df$Importanza * 100
ggplot(rel_df, aes(x = reorder(Variabile, Importanza), y = Importanza)) +
  geom_col(fill = "darkcyan") + geom_text(aes(label = paste0(round(Importanza, 1), "%")),
            hjust = -0.1, color = "black", size = 3.5) +
  coord_flip() +  labs(x = "Variabili", y = "Importanza relativa (%)",
      title = "Contributo percentuale alla spiegazione del reddito") +
  theme_minimal() +
  ylim(0, max(rel_df$Importanza) * 1.15)
```



Con questo blocco di codice stiamo calcolando e visualizzando graficamente l'importanza relativa delle variabili esplicative in termini di quanto ciascuna contribuisca a spiegare la variabile dipendente.
"Carta" spiega da sola il 62,6% della varianza del reddito tra le variabili considerate: è quindi la variabile più rilevante.
"Bancomat" contribuisce con un 21,5% ha un ruolo importante, ma meno incisivo.
"Cartapre" ha il contributo minore con 15,9%.
Questo significa che la variabile "carta" è di gran lunga la più informativa per spiegare le differenze nel reddito e probabilmente, chi ha redditi più alti ha più probabilità di possedere una carta di credito. Al contrario, la carta prepagata

Relazione SBD - Progetto

è meno associata al reddito, forse perché è utilizzata più trasversalmente o da chi non può accedere a carte tradizionali.

ANALISI REGRESSIONE LOGISTICA

La regressione logistica serve a modellizzare una variabile 0-1 in funzione di una o più variabili indipendenti. Essa stima la probabilità che si verifichi un evento sulla base di un determinato set di dati variabili indipendenti.
Nel nostro contesto abbiamo deciso di utilizzarla per stimare la probabilità che un individuo possieda una carta in funzione delle variabili indipendenti Y.x e C(reddito netto e consumi).

Preparazione dei dati

1. La variabile m4$carta è stata trasformata in una variabile categorica con etichette "possiede" (1) e "non possiede" (2).
2. È stata creata una variabile binaria carta_binaria (1 = "possiede", 0 = "non possiede") per l'analisi.

m4$carta == "possiede" questa parte confronta ogni elemento della variabile m4$carta con la stringa "possiede" e restituisce un vettore di valori logici (TRUE o FALSE):

- TRUE se il valore è "possiede"
- FALSE se è "non possiede"

Creaimo attraverso la funzione **table** una tabella di contingenza che confronta i valori della variabile binaria carta_binaria con quelli della variabile categorica m4$carta.
La funzione table() conta quante volte ogni combinazione di carta_binaria e m4$carta appare nel dataset.

carta_binaria	possiede	non possiede
0	0	13091
1	6275	0

La trasformazione da m4$carta a carta_binaria ha funzionato alla perfezione, non c'è alcuna incongruenza. Abbiamo individuato 13.091 individui che non possiedono la carta e 6.275 che la possiedono.
In questo modo possiamo usare tranquillamente carta_binaria per analisi statistiche o modelli predittivi, sapendo che riflette con precisione la codifica originale.

Modello di regressione logistica

mod1=glm(carta_binaria~Y.x + C, data=m4, family = binomial)

carta_binaria è la variabile dipendente binaria.
Y.x + C sono le variabili indipendenti (o esplicative). Il modello cerca di capire come queste variabili influenzano la probabilità che carta_binaria sia 1.
family = binomial ciò specifica che stiamo usando una regressione logistica, cioè un modello adatto per una variabile dipendente binaria.
In pratica stiamo stimando la probabilità che un individuo possegga la carta in funzione di Y.x e C.

Eseguendo la funzione summary(mod1) otteniamo una sintesi dettagliata del modello.

Relazione SBD - Progetto

glm(formula = carta_binaria ~ Y.x + C, family = binomial, data = m4)
Coefficients:

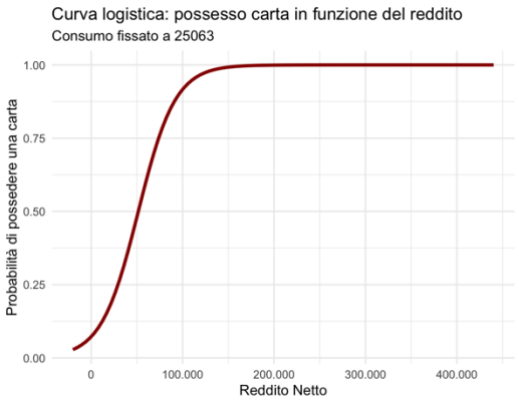
	Estimate	Std. Error	z value	Pr(>|z|)	
(Intercept)	-3.251e+00	4.885e-02	-66.56	<2e-16	***
Y.x	4.947e-05	1.504e-06	32.88	<2e-16	***
C	2.804e-05	2.241e-06	12.51	<2e-16	***

Signif. codes: 0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
(Dispersion parameter for binomial family taken to be 1)
Null deviance: 24396 on 19365 degrees of freedom
Residual deviance: 19029 on 19363 degrees of freedom
AIC: 19035
Number of Fisher Scoring iterations: 5

L’ intercetta(-3.251): indica la log-odds di possesso della carta quando Y.x = 0 e C = 0.
Sia Y.x che C sono positivi quindi per ogni incremento unitario in Y.x/C, le log-odds di possesso aumentano leggermente.
Tutti i p-value sono < 2e-16, quindi le variabili sono altamente significative nel modello.
Il modello funziona bene: spiega in modo significativo chi possiede la carta
Y.x e C hanno effetto positivo, anche se piccolo
L’intercetta negativa implica che senza i predittori, la probabilità base di possesso è molto bassa.

`mod1$coefficients[1]` è l’intercetta del modello, che rappresenta le log-odds
`exp(mod1$coefficients[1])` trasforma le log-odds in odds veri e propri.
`exp(-3.251) ≈ 0.0386` questo significa che, quando Y.x = 0 e C = 0, le probabilità di possedere una carta sono circa 3.86 a 100, o meno del 4%.

```
reddito_seq <- seq(min(m4$Y.x), max(m4$Y.x), length.out = 100)
consumo_medio <- mean(m4$C, na.rm = TRUE)
df_pred <- data.frame(Y.x = reddito_seq, C = consumo_medio)
df_pred$prob <- predict(mod1, newdata = df_pred, type = "response")
ggplot(df_pred, aes(x = Y.x, y = prob)) +
  geom_line(color = "darkred", linewidth = 1.2) +
  labs(x = "Reddito Netto", y = "Probabilità di possedere una carta",
       title = "Curva logistica: possesso carta in funzione del reddito",
       subtitle = paste("Consumo fissato a", round(consumo_medio, 0))) +
  scale_x_continuous(labels = label_comma(big.mark = ".", decimal.mark = ",")) +
  theme_minimal()
```



Stiamo osservando l'effetto isolato del reddito sul possesso della carta, ipotizzando che tutti i soggetti abbiano un uguale livello di consumo medio.
Ai livelli più bassi di reddito, la probabilità di possedere una carta è bassa, nonostante i consumi siano medi → possibile difficoltà di accesso a strumenti finanziari.
Con l'aumentare del reddito, la probabilità sale rapidamente → le persone con consumi simili ma più reddito sono più bancabili, più stabili, più affidabili agli occhi del sistema.

Relazione SBD - Progetto

Oltre una certa soglia (es. 100.000 €), la probabilità si stabilizza verso l'1 → praticamente tutti hanno la carta.
fissare i consumi medi ci permette di vedere che non è quanto si consuma a determinare il possesso della carta, ma quanto si guadagna. Questo sottolinea che l'accesso a strumenti finanziari è più legato al reddito che allo stile di vita.

```
predict.probs = predict(mod1, type="response")
```

predict() applica il modello ai dati: stima il valore della variabile dipendente per ogni riga del dataset.
type = "response" fa sì che le predizioni siano espresse come probabilità comprese tra 0 e 1 (anziché in log-odds).
In questo modo otteniamo un vettore predict.probs della stessa lunghezza del dataset, dove ogni valore rappresenta la probabilità stimata che l'individuo possegga una carta.
Con head(predict.probs) otteniamo i primi 6 valori, ovvero le prime 6 probabilità stimate dal modello.
Questo aiuta a fare un controllo veloce per capire se il modello sta restituendo valori plausibili (compresi tra 0 e 1) e dà un'idea generale delle previsioni prodotte.

```
     1          2          3          4          5          6
0.30406423 0.11010849 0.14449225 0.25850515 0.09776987 0.09776987
```

Questo suggerisce che nessuna delle prime sei persone ha una probabilità elevata di possedere una carta.
Le probabilità sono tutte ben sotto il 50%, quindi se volessimo fare una classificazione binaria (es. possiede = 1 se prob > 0.5), tutte queste osservazioni verrebbero previste come "non possiede".
mod1$fitted.values restituisce tutte le probabilità stimate.
Dato questo assumiamo che la soglia di classificazione sia p.soglia =0,3.

predict.class = ifelse(predict.probs>=p.soglia,1,0) serve a trasformare le probabilità stimate dal modello in una classificazione binaria basata su una soglia decisionale chiamata p.soglia.
Si applica questa logica:
Se la probabilità stimata è maggiore o uguale a p.soglia, assegna 1 (cioè "si prevede che possieda la carta"), altrimenti assegna 0 ("si prevede che non la possieda").

```
conf.mat = table(Predetti=predict.class,
                 Osservati = carta_binaria)
```

Crea una matrice di confusione che è uno strumento fondamentale per valutare quanto bene il modello logistico riesce a classificare correttamente chi possiede o non possiede la carta.

```
                Osservati
Predetti         0    1
       0.     9744 1690
       1      3347 4585
```

Da questo determiniamo:
TN = conf.mat[1,1] quante volte il modello ha predetto 0 correttamente.
TP = conf.mat[2,2] quante volte il modello ha predetto 1 correttamente.
FN = conf.mat[1,2] quante volte il modello ha predetto 0 mentre il valore oss. era 1.
FP = conf.mat[2,1] quante volte il modello ha predetto 1 mentre il valore oss. era 0.

Da questi possiamo calcolare la:

Relazione SBD - Progetto

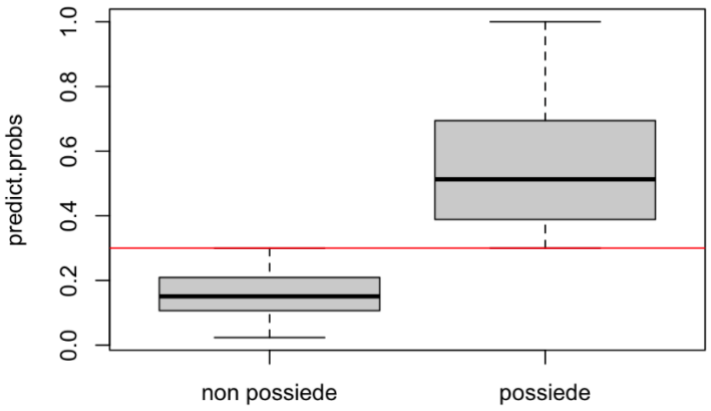
sensitivity = TP/(TP+FN) misura la precisione del modello nel classificare le osservazioni avente valore pari a 1, viene anche detta TPR(true positive rate).
specificity = TN/(TN+FP) misura la precisione del modello nel classificare correttamente tutte le osservazioni i tali che Yi=0, viene anche detta TNR.

Analisi Esplorativa

Boxplot

predict.class <- factor(predict.class, levels = c(0, 1), labels = c("non possiede", "possiede"))
Con questa riga di codice stiamo trasformando il vettore predict.class, che contiene le previsioni binarie numeriche (0 = non possiede, 1 = possiede), in un fattore leggibile con etichette testuali.

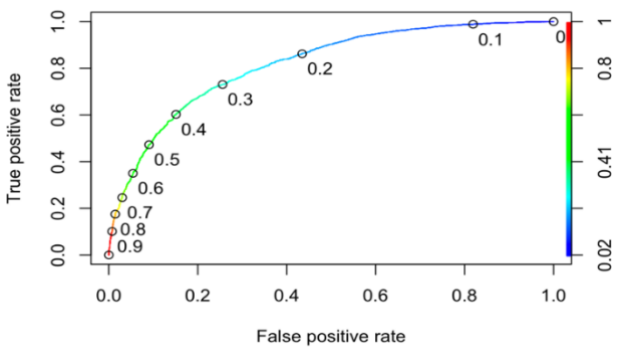
boxplot(predict.probs~predict.class); abline(h=0.3, col="red") Questa istruzione crea un grafico a scatola che mostra la distribuzione delle probabilità predette in base alle classi predette dal modello, e aggiunge una linea rossa orizzontale per visualizzare la soglia decisionale sul grafico(0,3).



Il boxplot per "non possiede" mostra probabilità tipicamente basse, concentrate sotto la soglia rossa: perfetto, perché indica coerenza tra bassa probabilità e previsione di non possesso.
Il boxplot per "possiede" mostra probabilità più alte, molte sopra la soglia: anche qui bene perché indica che il modello assegna alte probabilità a chi predice come possessore.
Abbiamo una buona separazione tra le due classi predette, segno che il modello è in grado di distinguere in modo sensato tra chi possiede e chi no.

Curva ROC

ROCRpred = prediction(predict.probs, carta_binaria)
Prepara i dati predetti e osservati per valutare le performance di classificazione del modello.
ROCRperf = performance(ROCRpred, measure = "tpr", x.measure = "fpr")
Questa serve per creare una curva ROC (Receiver Operating Characteristic).
performance() calcola una metrica di performance a partire da questo oggetto.
measure = "tpr" indica che vogliamo calcolare la True Positive Rate (sensibilità).
x.measure = "fpr" indica che vogliamo usare la False Positive Rate sull'asse delle x.



La curva Roc è un grafico diagnostico che misura la performance in termini di classificazione del modello al variare delle diverse soglie che applichiamo alla probabilità per classificare le osservazioni come 0 oppure 1.
Questo grafico ci aiuta a capire quanto bene il modello distingue tra le due classi, al variare della soglia di decisione.

performance(ROCRpred, measure = "auc")@y.values[[1]] serve a estrarre il valore dell'AUC dalla curva ROC.

Relazione SBD - Progetto

Un valore pari **0,8** indica che il classificatore ha **buona capacità discriminante**, quindi ha **l'80% di probabilità di distinguere correttamente** tra un soggetto che possiede una carta e uno che non la possiede. Il modello quindi è affidabile e ben costruito.

ANALISI CLUSTER

Obiettivo

L'obiettivo del clustering è quello di suddividere un insieme di dati in gruppi omogenei, detti appunto cluster, in modo che gli elementi all'interno dello stesso cluster siano simili tra loro, mentre quelli di cluster diversi siano il più possibile differenti.

Preparazione dei dati

La prima operazione che abbiamo effettuato è proprio questa:

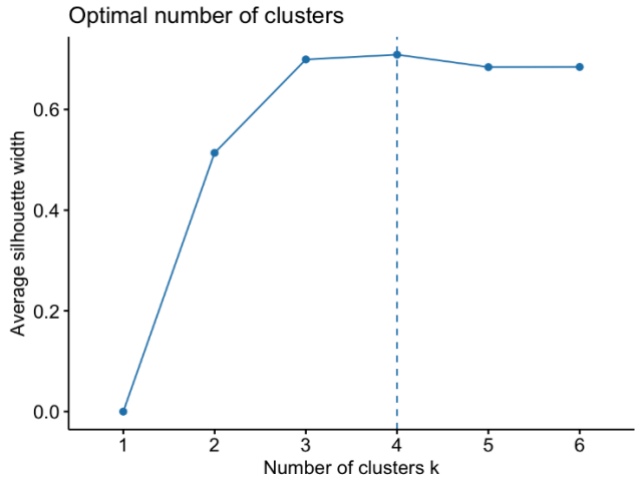
```
m4$carta    <- ifelse(m4$carta == 1, 1, 0)
m4$bancomat <- ifelse(m4$bancomat == 1, 1, 0)
m4$cartapre <- ifelse(m4$cartapre == 1, 1, 0)
```

Questa operazione serve a trasformare le variabili in formato binario(0/1), assicurando pulizia dei dati, standardizzazione e la corretta preparazione per l'analisi di clustering. In questo modo, si garantisce che ogni osservazione venga trattata in modo uniforme dall'algoritmo di segmentazione.
data.CL <- na.omit(m4[, c("ireg","carta", "bancomat", "cartapre")]) in questo modo stiamo creando un sottoinsieme dei dati contenente solo le informazioni rilevanti e prive di valori mancanti, in modo da poter eseguire l'analisi in modo affidabile.
colnames(data.CL) Mostra i nomi delle colonne del dataset
dim(data.CL) Restituisce il numero di righe e colonne del dataset.
str(data.CL) Mostra la struttura interna del dataset, verifica che tutte le variabili siano nel formato giusto(numerico per il clustering).
class(data.CL) Dice che tipo di oggetto è data.CL
summary(data.CL) Fornisce una statistica descrittiva di base per ogni variabile, è un primo sguardo utile per identificare valori anomali, sbilanciamenti o outlier.
Con questo blocco di codice stiamo facendo un controllo diagnostico di base.
data.CL$area_geografica <- cut(data.CL$ireg, breaks = c(0, 8, 14, 18, 20), labels = c("Nord", "Centro", "Sud", "Isole")) Con questa operazione stiamo creando una nuova variabile chiamata area_geografica che classifica ogni riga del dataset data.CL in una delle quattro macro-aree geografiche italiane, in base al valore numerico della variabile ireg. Ciò è utile per passare da un codice numerico regionale a una classificazione territoriale significativa.
CL_scaled <- scale(data.CL[, c("carta", "bancomat", "cartapre")]) Questa riga serve a normalizzare (o standardizzare) le tre variabili binarie. La funzione scale() centra ogni variabile (sottrae la media) e ridimensiona (divide per la deviazione standard).
Il risultato è che ciascuna variabile avrà media 0 e deviazione standard 1.
La variabile ireg non è stata inclusa nel clustering perché non rappresenta un comportamento ma una localizzazione geografica.

```
fviz_nbclust(CL_scaled, kmeans, method = "silhouette", k.max = 6)
```

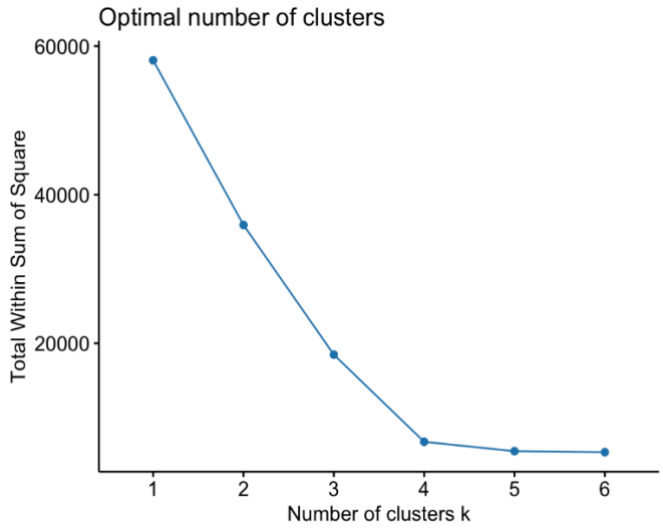
Visualizza la bontà del clustering per un certo range di valori di k.
CL_scaled: sono i tuoi dati standardizzati, su cui vuoi fare il clustering.
kmeans: specifica l'algoritmo da usare
method = "silhouette": usa l'indice di silhouette come criterio di bontà per valutare ogni k.
k.max = 6: considera da 1 fino a 6 cluster.

Relazione SBD - Progetto



Questo grafico mostra l'indice silhouette medio in funzione del numero di cluster scelto per l'algoritmo di k-means.
Il grafico suggerisce che 4 cluster è la scelta più solida che ci permette di catturare la struttura dei dati bilanciando chiarezza interna e separazione tra gruppi.

```
fviz_nbclust(CL_scaled, kmeans, method = "wss", k.max = 6)
```



Questa funzione crea il grafico dell'"elbow" (gomito) usando il metodo WSS (Within Cluster Sum of Squares), e serve a identificare il numero ottimale di cluster da utilizzare per il k-means.

Il punto di svolta visivo, dove la curva forma una sorta di gomito, è il numero di cluster ottimale(k=4).
Quindi dato che entrambi i grafici si trovano d'accordo sulla scelta di k=4 possiamo dire che k = 4 rappresenta il miglior compromesso tra compattezza e interpretabilità.
Nonostante questo, preferiamo scegliere k=3 per i seguenti motivi:
La segmentazione risulta più semplice da interpretare o comunicare;
Riduciamo la complessità del modello;
I profili risultanti con 3 cluster sono comunque ben distinti e coerenti.

Modello cluster (k-means)

```
CL_cluster <- kmeans(CL_scaled, centers = 3, iter.max = 100,
nstart = 10)
```

Abbiamo appena segmentato gli utenti in 3 gruppi simili tra loro in base all'uso di carta, bancomat e prepagata.
names(CL_cluster) è un'operazione molto utile per esplorare ciò che contiene l'oggetto CL_cluster dopo il clustering, ovvero:
"cluster" indica l'assegnazione di ciascun punto al cluster

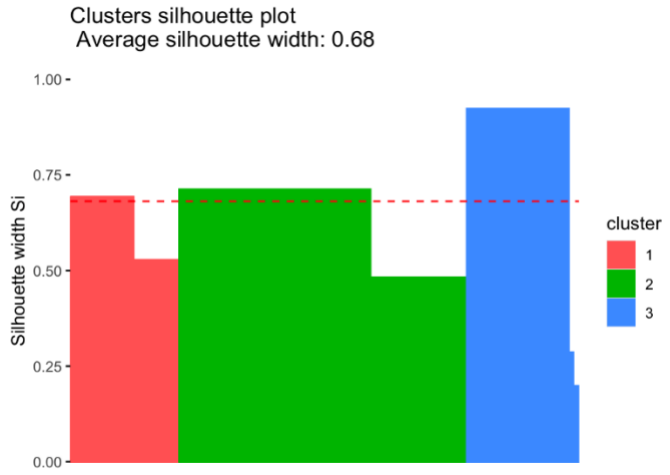
Relazione SBD - Progetto

"centers" indica le coordinate dei centroidi (uno per cluster)
"totss" indica la varianza totale del dataset
"withinss" indica la varianza interna per ciascun cluster
"tot.withinss" indica la somma totale delle varianze intra-cluster (WSS)
"betweenss" indica la varianza spiegata dai cluster (tra i centroidi)
"size" indica il numero di osservazioni per ogni cluster
"iter" indica il numero di iterazioni effettuate
"ifault" indica lo stato dell'algoritmo (0 = ok)

data.CL$cluster <- CL_cluster$cluster è utile perché permette di lavorare, visualizzare e analizzare i risultati del clustering usando il dataset data.CL, che ora contiene anche l'informazione di a quale cluster appartiene ogni osservazione.
distances = get_dist(CL_scaled, method="euclidean") serve a calcolare la matrice delle distanze tra le osservazioni.
sil_k_means = silhouette(CL_cluster$cluster, distances)
L'indice di silhouette misura quanto un'osservazione è ben assegnata al proprio cluster rispetto agli altri, confrontando la distanza media dagli altri punti nel cluster vs la distanza media dal cluster più vicino. È un valore compreso tra -1 e 1.

- Se pari a 1 l'unità è classificata bene;
- Se pari a 0 l'unità è nel mezzo dei cluster(bridge point);
- Se pari a -1 l'unità è classificata male.

cbind(area = as.character(data.CL$area_geografica), sil_k_means) cosi stiamo combinando i risultati del clustering con l'informazione geografica, in modo da poter analizzare quanto bene le osservazioni di ciascuna area sono state assegnate ai cluster.
fviz_silhouette(sil_k_means) serve a visualizzare graficamente gli indici silhouette calcolati per ciascuna osservazione, permettendo di valutare quanto bene ogni punto è stato assegnato al suo cluster.



La silhouette media di 0.68 è ottima e ciò indica che i cluster sono ben formati e separati.
I tre cluster appaiono bilanciati visivamente, ciascuno con molte osservazioni ben assegnate (barre alte).
Non si notano barre fortemente negative, il che rafforza la validità della scelta di k = 3.

table(data.CL$area_geografica, data.CL$cluster) così stiamo creando una tabella di contingenza che mostra la distribuzione dei cluster all'interno delle diverse aree geografiche (Nord, Centro, Sud, Isole). Questa è una mossa molto utile per interpretare i cluster dal punto di vista territoriale.

```
           1    2    3
Nord    2095 5008 1258
Centro  1229 2682  752
```

Relazione SBD - Progetto

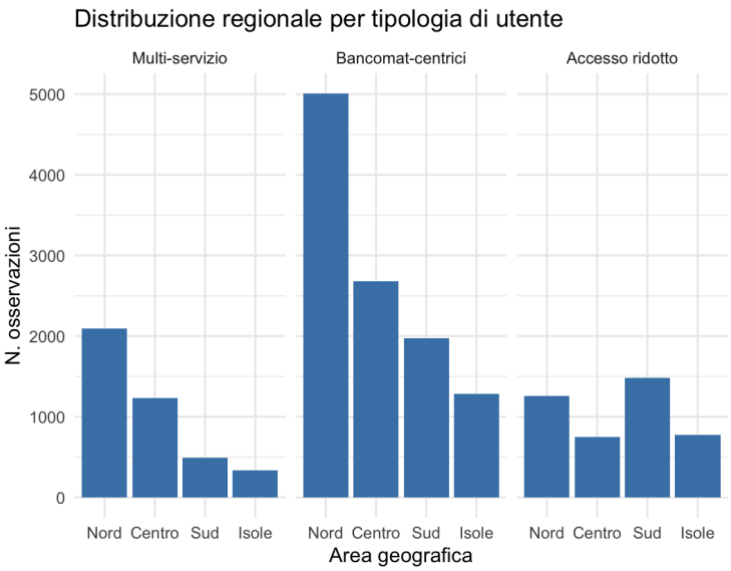
```
  Sud     493 1975 1479
  Isola   335 1281  779

risultato <- aggregate(. ~ cluster, data = data.CL[, c("carta",
"bancomat", "cartapre", "cluster")], mean)
print(risultato, row.names = FALSE)

cluster      carta  bancomat   cartapre
      1 0.58622351 0.9918112 1.00000000
      2 0.33509958 1.0000000 0.00000000
      3 0.04053421 0.0000000 0.05060918
```

Così stiamo calcolando e visualizzando il profilo medio di ciascun cluster rispetto all'uso dei tre strumenti bancari: carta, bancomat, e cartapre.
Questa tabella è il cuore dell'interpretazione comportamentale dei cluster. Mostra la media di utilizzo per ciascuno strumento finanziario in ogni gruppo individuato dal k-means.
Questi tre cluster evidenziano tre chiaramente tre tipologie di utenti:
multiservizi: utilizzano tutti i tipi di carte
bancomat centrici: utilizzano praticamente solo il bancomat
accesso ridotto: non usano quasi nessuna carta

```
data.CL$etichetta_cluster <- factor(data.CL$cluster,
                                    levels = c(1, 2, 3),
                                    labels = c("Multi-servizio",
"Bancomat-centrici", "Accesso ridotto"))
ggplot(data.CL, aes(x = factor(area_geografica))) +
  geom_bar(fill = "steelblue") +
  facet_wrap(~ etichetta_cluster) +
  labs(x = "Area geografica", y = "N. osservazioni", title =
"Distribuzione regionale per tipologia di utente") +
  theme_minimal()
```



Questo codice serve a visualizzare la distribuzione delle osservazioni per area geografica all'interno di ciascun cluster, sotto forma di grafico a barre suddiviso in pannelli.
Il Nord ha il maggior numero di utenti "multi-servizio" e "bancomat-centrici", a indicare un alto livello di bancarizzazione e diversificazione nei metodi di pagamento.
Il Centro ha una distribuzione più contenuta e bilanciata, con prevalenza di bancomat-centrici.
Il Sud registra un'elevata presenza di "bancomat-centrici", ma anche la quota più alta di utenti con accesso ridotto, suggerendo disparità digitali e bancarie.
Le Isole mostrano lo stesso pattern del Sud, con una netta prevalenza di "accesso ridotto" rispetto ai "multi-servizio".

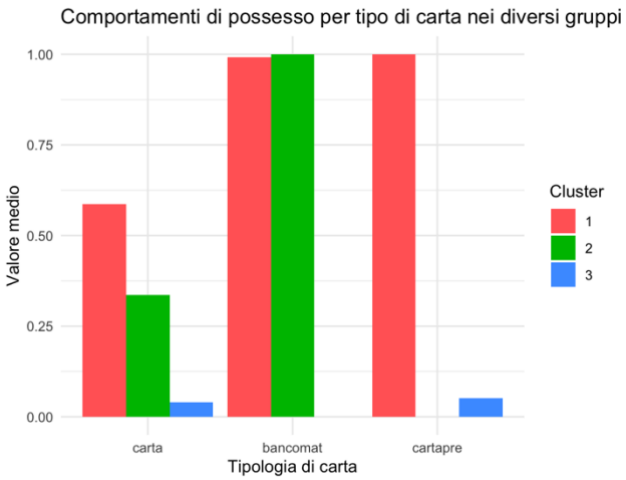
prop.table(table(data.CL$etichetta_cluster))

Relazione SBD - Progetto

Multi-servizio	Bancomat-centrici	Accesso ridotto
0.2143964	0.5652174	0.2203862

La maggioranza assoluta degli individui appartiene al gruppo “bancomat-centrici”, suggerendo che il bancomat è lo strumento più diffuso nel campione.
Il fatto che solo 1 persona su 5 sia “multi-servizio” sottolinea che l’utilizzo integrato di più strumenti finanziari non è ancora predominante.
La presenza significativa di soggetti con accesso ridotto (22%) è un segnale importante: indica che oltre un quinto della popolazione esaminata può essere parzialmente escluso dal sistema bancario tradizionale.

```
dati_long <- melt(risultato, id.vars = "cluster")
ggplot(dati_long, aes(x = variable, y = value, fill = factor(cluster))) +
  geom_col(position = "dodge") +
  labs( x = "Tipologia di carta", y = "Valore medio", fill = "Cluster",
    title = "Comportamenti di possesso per tipo di carta nei diversi gruppi"
  ) +
  theme_minimal()
```



Questo codice crea un grafico a barre a gruppi che confronta il possesso medio (cioè la percentuale) di tre strumenti di pagamento (carta, bancomat, cartapre) nei diversi cluster di utenti.
Questo grafico è estremamente efficace per:

- Visualizzare e confrontare i **comportamenti d’uso tra gruppi di utenti**.
- Identificare **gruppi digitalizzati vs esclusi**.
- Comunicare sinteticamente risultati emersi dall’analisi dei cluster.

CONSIDERAZIONI FINALI

Motivazioni della scelta metodologica

1. **Regressione lineare**: è stata utilizzata per isolare l'effetto di singoli strumenti di pagamento sul reddito netto, ottenendo così una comprensione diretta delle relazioni lineari.
2. **Regressione multipla**: ci ha consentito di valutare l'impatto congiunto di più variabili, tenendo conto delle interazioni tra loro.
3. **Regressione logistica**: è stata utilizzata per stimare la probabilità di possesso di una carta di credito in base al reddito e alle spese di consumo, fornendo un modello predittivo robusto.
4. **Analisi cluster**: abbiamo segmentato le famiglie in gruppi omogenei in base all'uso degli strumenti finanziari, rivelando differenze comportamentali significative tra le aree geografiche.

Vantaggi dell'approccio

- **Completezza**: L'uso combinato di diverse tecniche ha permesso di esplorare il fenomeno da più prospettive, garantendo una comprensione approfondita.
- **Flessibilità**: Le analisi hanno adattato metodi statistici sia per variabili continue che categoriche, massimizzando l'utilizzo dei dati disponibili.

Relazione SBD - Progetto

- **Interpretabilità**: I risultati sono stati presentati in modo chiaro, con visualizzazioni che facilitano la comunicazione dei risultati anche ad un pubblico non tecnico.

Limiti e possibilità di miglioramento

1. **Dipendenza dai dati**: L'analisi è vincolata alla qualità e alla rappresentatività del dataset. L'inclusione di dati più recenti o variabili aggiuntive (es. livello di istruzione, età) potrebbe migliorare la robustezza dei risultati.
2. **Semplicità dei modelli**: Nonostante l'uso della regressione multipla, alcune relazioni potrebbero essere più complesse e richiedere modelli non lineari o machine learning avanzato.
3. **Soggettività nella scelta dei cluster**: La decisione di utilizzare 3 cluster, sebbene giustificata, potrebbe essere validata ulteriormente con metodi alternativi o criteri più rigorosi.
4. **Mancanza di analisi causali**: Le correlazioni identificate non implicano causalità. Studi futuri potrebbero integrare metodi sperimentali per indagare relazioni causali.

Prospettive future

Per approfondire lo studio, sarebbe interessante:

- Espandere l'analisi a dataset più recenti per valutare l'evoluzione temporale delle abitudini finanziarie.
- Introdurre tecniche di machine learning per identificare pattern non lineari o interazioni complesse tra variabili.
- Condurre indagini qualitative per comprendere i motivi alla base delle differenze geografiche emerse nell'analisi cluster.

In conclusione, questo progetto ha dimostrato l'utilità di un approccio multidisciplinare nell'analisi dei dati finanziari, offrendo insights rilevanti per le istituzioni finanziarie. Nonostante i limiti, la metodologia adottata si è rivelata efficace nel descrivere e interpretare le dinamiche osservate, ponendo le basi per future ricerche più approfondite.

LAVORO A CURA DI:
ANTONIO CIUFFREDA,605734
DIEGO LO CURTO,575524
FEDERICO RENZI,608983
VALENTINO LUCCI,608725
GABRIELE SANTELLI,606286
RICCARDO DI GREGORIO,606017