Credit card, yes or no?

Marc Castañé del Barrio

Resum—Aquest treball de Fi de Grau (TFG) presenta un model de classificació de clients bancaris utilitzant tècniques d'aprenentatge automàtic implementades en Jupyter Notebook. L'objectiu principal és esenvolupar un model que pugui predir de manera precisa la probabilitat que un client es converteixi en morós o no, utilitzant característiques demogràfiques, financeres i de comportament del client. S'utilitza una varietat d'algoritmes de classificació com DecisionTreeClassifier, SVC, XGBClassifier i RandomForestClassifier per entrenar i avaluar el model. Les dades utilitzades per a l'estudi consisteixen en una mostra de clients històrics del banc txec Berka. Els resultats obtinguts demostren una alta precisió de classificació i permeten identificar els factors clau que influeixen en el risc financer dels clients. Aquest treball proporciona una eina útil per a la pressa de decisions basadad en la classificació de clients en el sector bancari, amb aplicacions en la gestió de riscos i l'optimització d'estratègies de màrqueting i vendes. La investigació realitzada contribueix al camp de l'anàlisi de dades bancàries i proporciona informació valuosa per a la presa de decisions efectiva en el sector bancari.

Paraules clau— DecisionTreeClassifier, Support Vector Classifier, XGB Classifier, Random Forest Classifier, Aprenentatge Automàtic, Llindar, Estandarditzar, Accuracy, Precisió, Recall, híper-paràmetres, sobreclassificació.

Abstract— This Bachelor's Degree thesis (TFG) presents a model for classifying bank clients using machine learning techniques implemented in Jupyter Notebook. The main objective is to develop a model that can accurately predict the probability of a client becoming delinquent or not, using demographic, financial, and behavioral client features. A variety of classification algorithms such as DecisionTreeClassifier, SVC, XGBClassifier, and RandomForestClassifier are used to train and evaluate the model. The data used for the study consists of a sample of historical clients from the Czech bank Berka. The results obtained demonstrate high classification accuracy and allow for identifying key factors that influence clients' financial risk. This work provides a useful tool for decision-making based on client classification in the banking sector, with applications in risk management and optimization of marketing and sales strategies. The conducted research contributes to the field of banking data analysis and provides valuable information for effective decision-making in the banking sector.

Index Terms— Decision Tree Classifier, Support Vector Clasifier, XGB Classifier, Random Forest Classifier, Machine Learning, Threshold, Standardise, Accuracy, Precision, Recall, Hyper Parameters, Over-classification.

- *E-mail de contacte: Marccdb8888@gmail.com*
- *Treball tutoritzat per: Daniel Franco Puntes (departament)*
- *Curs 2022/23*

1 INTRODUCCIÓ - CONTEXT DEL TREBALL

Avui en dia la Intel·ligència Artificial, els models predictius amb Machine learning o Deep learning, el Big Data, tot lo relacionat amb la tecnología s'ha tornat imprescindible i quan es va proposar el tema vaig veure la posibilitat d'una aportació interessant.

El meu Treball de Fi de Grau tracta d'analitzar si un client bancari nou és bo o dolent per al banc a partir de les dades que es tenen sobre ell. La identificació de clients amb possibles riscos és una tasca clau per als bancs, ja que els ajuda a evitar possibles pèrdues. En aquest treball, proposo l'ús de diferents tècniques d'anàlisi de dades per a l'avaluació dels clients bancaris.

El sector financer és un dels sectors principals en quant a la tecnología, ja que manegen molts diners i a més a més no són seus, així que han d'estar el més avançats possibles per a poder donar un servei òptim i mai es poden permetre el luxe de tenir dades inconsistents, és a dir, un client no pot tenir 500€ si ho mira des del móvil i 5000€ si ho mira des de l'ordinador.

Els bancs com la majoria d'empreses tracten de millorar els seus serveis per a obtenir majors ingressos, la qual cosa deriva a majors beneficis.

Actualment els gerents bancaris tenen una vaga idea sobre si un client és bo (al qual ens interessaria oferir-li més serveis) o si un client és dolent (al qual hauríem de vigilar-lo més detalladament per a minimitzar les pèrdues del banc). Afortunadament, els bancs disposen de moltíssimes dades emmagatzemades sobre els seus clients, els comptes, els préstecs concedits… Amb totes aquestes dades es pot fer un estudi profund i obtenir grans conclusions, en el meu cas, l'objectiu principal será analitzar aquestes dades bancàries per tal d'extrapolar-ne el tipus de client que fa un bon candidat per a una targeta de crèdit.

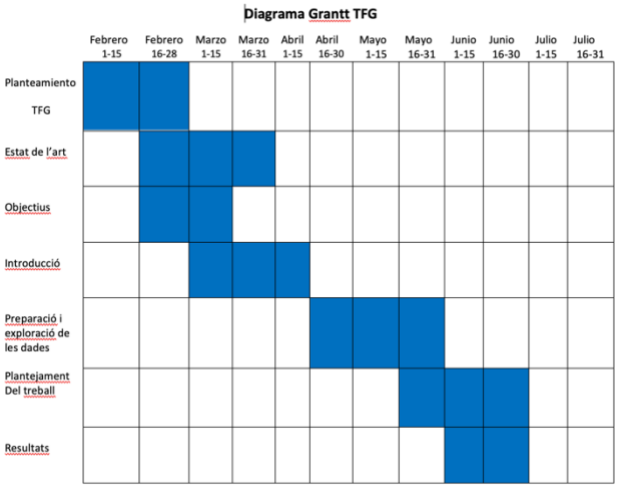
Les dades que s'utilitzarán serán les del dataset Berka que és una col·lecció d'informació financera anónima real d'un banc txec.

Per començar, s'analitzarán les dades demogràfiques del client, com la seva edat, gènere i ubicació geogràfica. Aquestes dades poden donar una idea de la situación actual del client i de la probabilitat que el client pugui ser un deutor problemàtic.

A continuació, s'analitzarán les dades financeres del client, com el seu historial, els seus ingressos i despeses, i el seu patrimoni net. Aquestes dades ja sí que indiquen amb més certeza la capacitat económica del client per fer front als seus pagaments i la seva capacitat per absorbir pèrdues financeres.

Per a l'anàlisi, s'utilitzaràn diferents tècniques estadístiques, per exemple anàlisi de regressió, anàlisi de discriminació o anàlisi de components principals, però encara no s'ha decidit quin o quins exactamente. Això permetrà identificar les variables més significatives per a més endavant fer la predicció de la solvència del client.

Finalment, es farà un model predictiu que combini les dades juntament amb les tècniques d'anàlisi per a la valoració del client. La idea és que gràcies a aquest model els bancs o qui sigui puguin predir la capacitat d'un nou client ajudant-lis en la seva presa de decisions i en la seva gestió de riscos. En resum, aquest TFG té com a objectiu avaluar si és possible determinar si és bo o dolent donar-li una tarjeta de crèdit a un client bancari a partir de les dades que es tenen sobre ell mitjançant un model predictiu amb Machine Learning.



2 OBJECTIUS

- **Determinar la qualitat del client:**
    - Determinar la solvència financera d'un client concret (Determinar si el client té bona pinta o no, però de manera superficial, per a fer un cribatge inicial)
    - Valorar l'emissió d'una targeta de crèdit a un client nou (Quan la suma dels atributs rellevants donin >= que el valor que es determini)
    - Determinar la probabilitat que un client actual esdevingui un deutor problemàtic (client que no és deutor problemàtic però és probable per les característiques concretes que es torni deutor problemàtic)

3 PROPOSTA

Realitzar un model regressió logística que permeti classificar i predir als clients dels bancs en funció d'un conjunt de variables que es considerin relevants. Aquestes variables serán les que influeixin més en la relació client-banc.

4 ESTAT DE L'ART

A l'era moderna, els equips de ciència de dades dels bancs construeixen models predictius utilitzant l'aprenentatge automàtic.

Hi ha molts models de Machine learning per a tasques de classificació com poden ser: Random Forest model, Regressió logística, XGBoost, SVM o també xarxes neuronals, a més, dins de cada model, es poden perfeccionar els resultats mitjançant ajustaments més acurats a dels hiperparàmetres.
El dataset que s'utilitza va sortir a l'any 1999 i com era d'esperar hi ha més d'un treball amb la intención de crear un model predictiu dels clients nous. A continuació es comenten 2 d'aquests treballs.

L'objectiu principal d'un dels 2 treballs es comprobar si un client és un bon candidat o no per a una targeta de crèdit.

En aquest treball el primer que van fer va ser un data profiling on es netegen les dades, s'explora la informació, es verifiquen les relacions de les taules, tenint en compte que volien una base de dades relacional...

Un cop fet això, van optar per fer un ranking dels atributs per així eliminar els atributs poc rellevants o sense rellevància pel seu atribut de decisió que era un creat per ells anomenat: CardType. Van utilitzar l'eina see5, que utilitza un algorisme d'arbre de decisió per fer la classificació i així poder predir quins comptes eren titulars de targetes de crèdit

Finalmente, es va utilizar l'algorisme Cobweb implementat pel toolkit de Weka per a obtenir els clústers dels titulars de targetes, és a dir, per clasificar als clients en grups.

Els resultats van ser molt bons determinant, per exemple, que tots els clients de targetes de credit que han contractat un préstec solen tenir un bon estat de reemborsament o que no hi ha una diferència notòria entre el homes i les dones en l'emissió de targetes de crèdit.

Laltre treball comença semblant, és a dir preparant-se les dades en l'entorn que millor li convé, en aquest cas fent SQL queries per importar-ho a MySQL database i ho connecta amb Python per a utilitzar la llibrería Pandas.

Un cop es tenen les dades ben estructurades van optar per crear gràfiques entre la variable de préstecs i la resta de variables i d'aquesta manera quedar-se amb les característiques més rellevants. Com que hi ha atributs numèrics i categòrics, va utilitzar un escalador de scikit-learn per a passar els valors numèrics acategòrics, hi ha molts escaladors com: StandardScaler, MinMaxScaler and RobustScaler, en aquest cas s'utilitza MinMaxScaler per passar els atributs numèrics als valors binaris 0 o 1 i pels atributs categòrics es va utilizar OneHotEncoder també per posar els valors en 0 o 1.

En aquest cas, s'escull el model d'aprenentatge automàtic de Random Forest pel seu gran rendiment i els resultats del set d'entrenament van ser del 97% d'accuracy i els del set de test Baixa al 89,2%.

En conclusió, els models predictius utilitzant aprenentatge automatic satisfan els requisits de l'entitat financera, i quant millor netegis les dades i més hiper-paràmetres provis, millors resultats obtindràs.

5 DATASET BERKA

El dataset Berka és un conjunt de dades reals sobre la informació financera d'un banc txec i les dades están anònimes.

Taules Dataset Berka:

El data set consta de 8 arxius els quals cadascun representa una taula són els següents:

Compte: Cada registre descriu característiques estàtiques d'un compte.

Client: Cada registre descriu característiques d'un client.

Districte: Cada registre descriu característiques demogràfiques d'un districte.

Préstec: Cada registre descriu un préstec concedit a un compte concret.

Comanda: Cada registre descriu les característiques d'una comanda de pagament.

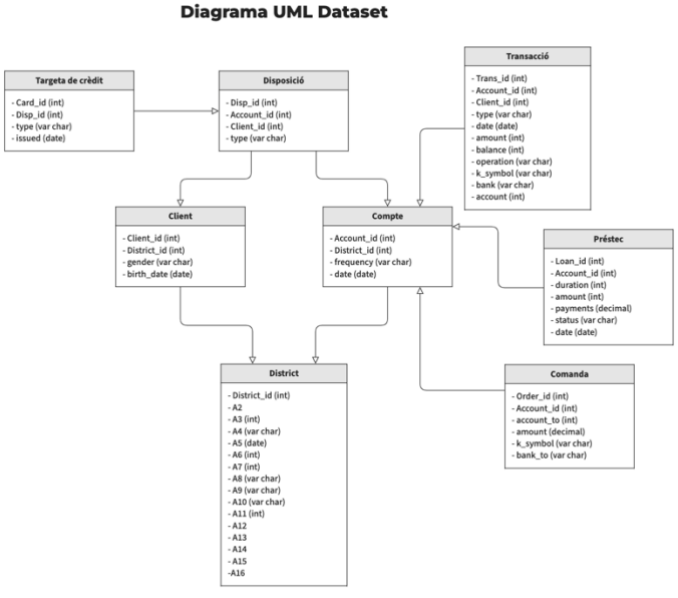
Disposició: Cada registre relaciona a un client amb un compte.

Transacció: Cada registre descriu una transacció d'un compte.

Targeta de crèdit: Cada registre descriu una targeta de crèdit emesa a un compte.

Tipus de relacions entre les taules:

- Un compte només pot tenir com a màxim un préstec.
- La relació de "Disposició" cap a "Client" descriu les característiques de les persones que poden manipular els comptes.
- Cada compte té característiques estàtiques que están en la taula "Compte" i característiques dinàmiques que están en les taules de "Transacció" i "Comanda".
- Les relacions de "Préstec" i "Targeta de crèdit" és que són serveis que el banc pot oferir als seus clients.
- La relació "Districte" et dona informació pública disponible sobre els districtes, amb la qual pots obtenir informació rellevant sobre els clients.
- Un compte pot tenir més d'una targeta de crèdit.
- Un compte pot ser utilitzat per més d'un client i un client pot utilitzar més d'un compte.
- Els clients i els comptes es relacionen a la taula "Disposició".



7 PREPARACIÓ DE LES DADES

Com ja he comentat anteriorment, el data set de Berka consta de 8 arxius i el primer que he hagut de fer ha estat netejarlos i ajuntar-los per a fer el model de predicció.

Aquests arxius són .asc però es podien tractar com a csv, així que vaig convertir els 8 arxius en dataframes amb la funció: pd.read_csv('nom_arxiu, sep=';')

Un cop he investigat tots els dataframes, m´he posat a fer un data Profiling per a quedar-me amb les dades que m'interesen pel model de predicció.

El dataframe Tarjetas_df(Figura 1) en un principi tenia 4 camps que eren: card_id, disp_id, type i issued, per una banda vaig canviar el nom de type per card_type per ser més concret i com que el camp issued està amb la data, vaig optar per canviar-ho a els dies que porta la targeta en actiu i per tant canvio el nom a card_issued_date.

```
Tarjetas_df.head()
```

	card_id	disp_id	type	issued
0	1005	9285	classic	931107 00:00:00
1	104	588	classic	940119 00:00:00
2	747	4915	classic	940205 00:00:00
3	70	439	classic	940208 00:00:00
4	577	3687	classic	940215 00:00:00

Figura 1

Aquí podem veure el resultat del dataframe Tarjetas_df(Figura 2).

```
Tarjetas_df
```

	card_id	disp_id	card_type	card_issued_date
0	1005	9285	classic	310
1	104	588	classic	383
2	747	4915	classic	400
3	70	439	classic	403

Figura 2

```
clientes_df.head()
```

	client_id	birth_number	district_id_x
0	1	706213	18
1	2	450204	1
2	3	406009	1
3	4	561201	5
4	5	605703	5

Figura 3

En el cas de Clientes_df(Figura 3) el birth_number ens dona informació sobre l'edat del client i també el sexe, així que el que vaig fer va ser separar-ho en 2 camps i posar els dies en anys que té més sentit.

Podem observar tant el camp client_age que és l'edat del client, com el camp client_gender que és el gènere del client (Figura 4).

```
clientes_df
```

	client_id	district_id_x	client_age	client_gender
0	1	18	29.071233	F
1	2	1	54.942466	M
2	3	1	59.268493	F
3	4	5	43.112329	M

Figura 4

```
account_df.head()
```

	account_id	district_id_x	frequency	date
0	576	55	POPLATEK MESICNE	930101
1	3818	74	POPLATEK MESICNE	930101
2	704	55	POPLATEK MESICNE	930101
3	2378	16	POPLATEK MESICNE	930101
4	2632	24	POPLATEK MESICNE	930102

Figura 5

En els comptes (Figura 5) s'han canviat els camps: date i frequency, el de frequency perquè està en txec i ho vaig passar a anglès i el de date per posar la data en que es va obrir el compte.

Aquí (Figura 6) els camps ja tenen un sentit més lògic.

```
account_df
```

	account_id	district_id_x	statement_freq	account_date_opened
0	576	55	MONTHLY	0
1	3818	74	MONTHLY	0
2	704	55	MONTHLY	0
3	2378	16	MONTHLY	0

Figura 6

En el cas del dataframe de disposició (Figura 7) només vaig canviar el nom de disp_type que en un principi era type, però la resta tot igual.

```
disp_df.head()
```

	disp_id	client_id	account_id	disp_type
0	1	1	1	OWNER
1	2	2	2	OWNER
2	3	3	2	DISPONENT
3	4	4	3	OWNER
4	5	5	3	DISPONENT

Figura 7

El cas del district_df (Figura 8) és una mica diferent ja que hi havia 16 camps però tots eren iguals, només variaven en funció del district_id que era qui determinava la resta dels camps. A més a més, els noms originals eren (A1, A2, A3...) i els vaig passar a noms que determinessin en què consistia el camp. Exemple, qualsevol persona que visqués a Barcelona tindria el mateix nombre d'habitants, de regió, de població, de crims...

```
district_df.head()
```

	district_id_x	district_name	region	num_inhabitants	num_munipalities_gt499	num_munipalities_500to1999	num_munipalities_2000to9999
0	1	Hl.m. Praha	Prague	1204953	0	0	0
1	2	Benesov	central Bohemia	88884	80	26	6
2	3	Beroun	central Bohemia	75232	55	26	4

Figura 8

```
loan_df.head()
```

	loan_id	account_id	date	amount	duration	payments	status
0	5314	1787	930705	96396	12	8033.0	B
1	5316	1801	930711	165960	36	4610.0	A
2	6863	9188	930728	127080	60	2118.0	A
3	5325	1843	930803	105804	36	2939.0	A
4	7240	11013	930906	274740	60	4579.0	A

Figura 9

En el cas dels préstecs (Figura 9) vaig decidir quedar-me amb la variable status ja que serà la variable objectiu pel model de predicció, i té 4 possibles valors que són A, B, C i D i vaig fer que si és A o C es posi 1 i si es B o D es posi 0 (després explico el perquè) però així és una variable de Sí o No.

A més a més, em vaig quedar amb el camp de account_id per agrupar-los i vaig fer que el amount fos la mitjana dels préstecs agrupats per la compta a la qual pertanyen. (Figura 10)

```
loan
```

	account_id	loan_status	loan_amount
0	2	1	80952.0
1	19	0	30276.0
2	25	1	30276.0
3	37	0	318480.0

Figura 10

```
order_df.head()
```

	order_id	account_id	bank_to	account_to	amount	k_symbol
0	29401	1	YZ	87144583	2452.0	SIPO
1	29402	2	ST	89597016	3372.7	UVER
2	29403	2	QR	13943797	7266.0	SIPO
3	29404	3	WX	83084338	1135.0	SIPO
4	29405	3	CD	24485939	327.0	

Figura 11

A les comandes (Figura 11)en un principi tenia aquests camps però vaig optar per fer el mateix que als préstecs, és a dir que vaig agrupar per account_id i vaig fer que el camp amount fos la mitjana de les comandes agrupades per la compta (Figura 12).

```
order
```

	account_id	order_amount
0	1	2452.000000
1	2	5319.350000
2	3	1667.000000
3	4	1681.500000

Figura 12

Finalment, al dataframe de les transaccions vaig fer el mateix que al de order.

Aquí podem veure el dataframe original de trans_df

```
trans_df.head()
```

	trans_id	account_id	date	type	operation	amount	balance	k_symbol	bank	account
0	695247	2378	930101	PRIJEM	VKLAD	700.0	700.0	NaN	NaN	NaN
1	171812	576	930101	PRIJEM	VKLAD	900.0	900.0	NaN	NaN	NaN
2	207264	704	930101	PRIJEM	VKLAD	1000.0	1000.0	NaN	NaN	NaN
3	1117247	3818	930101	PRIJEM	VKLAD	600.0	600.0	NaN	NaN	NaN
4	579373	1972	930102	PRIJEM	VKLAD	400.0	400.0	NaN	NaN	NaN

(Figura 13).

Figura 13

Aquest és el resultat del dataframe de transaccions (Figura 14).

trans

	account_id	trans_amount
0	1	1569.767782
1	2	6593.052929
2	3	2521.553846
3	4	1886.943011

Figura 14

Un cop tots els dataframes estaven bé i netejats, em vaig posar a ajuntar-los per a que més endavant pogués fer el model de predicció.

Vaig optar per juntar per les relacions directes, ja que era el més sensat, així que vaig ajuntar el dataframe Disp amb el de Tarjetas amb la primary key de disp_id.

I aquests dataframe el vaig ajuntar amb els clients amb el camp: client_id.

Per una altra banda vaig agafar els dataframes de préstecs, transaccions i orders i els vaig ajuntar amb el de comptes i tots tres relacionats amb el camp account_id.

Acte seguit vaig ajuntar el dataframe dels clients, que ja tenien les disposicions i les targetes, amb el dataframe de comptes, que ja tenia els préstecs, les comandes i les transaccions. Finalment vaig ajuntar el dataframe de districtes i ja tenia els 8 dataframes junts. I era un dataframe de 756 files x 33 columnes.

Un cop estaven junts, vaig decidir que per fer el model de predicció tots els camps de ID eren irrellevants i fins i tot afectarien negativament a la predicció, així que vaig decidir treure els camps de ID. Quedant un dataframe de 756 files x 29 columnes.

Per acabar, ja havia vist que del dataframe de districtes, tots els clients amb el mateix districte, tenien els camps repetits, així que era redundant tenir el 18 camps del districte i que només amb un era suficient, així que em vaig quedar només amb el de districte_id i la resta els vaig treure. Finalment em queda un dataframe de 756 files x 12 columnes.

8. METODOLOGIA

Un cop es va aconseguir un dataframe amb les dades necessàries per a realitzar el model de predicció en un principi la idea era fer un model de regressió logística perquè era el que més m'agradava a mí en la teoria, ja que m'encanten les matemàtiques però a l'hora de realitzar aquest model ja vaig començar a veure que no tenia molt bon futur, així que vaig optar per canviar de model.

La variable objectiu escollida és la de Loan_Status ja que diu si el client paga o no els seus préstecs i és molt interesant.

Variable Loan_Status:
- A: Contracte acabat, sense problemes
- B: Contracte acabat, préstec no pagat
- C: Contracte en funcionament, de moment tot correcte
- D: Contracte en funcionament, client en deute

És per això que els resultats de A i C els he posat en 1 i B i D en 0

Amb les dades netejades, el primer de tot va ser separar la variable Loan_Status de la resta, ja que era la variable objectiu i separar les dades en dades d'entrenament i dades de test (X_train, X_test, y_train i y_test), en aquest cas s'han separat en 75% d'entrenament i 25% de test.

Un cop realitzat el Train_Test_Split s'han preprocessat les variables, separant les variables numèriques de les categòriques. En el cas de les variables numèriques s'han estandarditzat les dades amb StandardScaler que el que fa és estandarditzar les dades eliminan la mitjana i escalant les dades de manera que la seva variança sigui igual a 1 y per una altra banda, les variables categòriques han estat tractades amb ColumnTransformer("One_hot_Encoder") que mitjançant un Umbral, transforma les variables categòriques en variables numèriques binàries (0 o 1).

Després de preprocessar les dades de manera separada, es tornen a juntar i finalment ja estan preparades per a ser utilitzades en els respectius models de classificació.

Finalment, el treball ha estat realitzat amb 4 models de classificació diferents, els quals han estat: DecisionTreeClassifier, SVC, XGBClassifier i RandomForestClassifier, el procés ha estat semblant sempre, primer s'han executat els 4 models amb els paràmetres per defecte i després mitjançant la funció GridSearchCV de la llibreria sklearn.model_selection, s'han obtingut els paràmetres òptims per a cada model de classificació de manera individual. En la següent imatge es pot veure el codi realitzat al model DecisionTreeClassifier amb la funció de Grid_Search (Imatge 1).

```
In [59]: # Define los parámetros que deseas probar
         param_grid = {
             'criterion': ['gini', 'entropy'],
             'max_depth': [None, 5, 10, 15, 20],
             'min_samples_split': [2,3,4, 5,7, 10,13,15],
             'min_samples_leaf': [1, 2, 4,5, 6]
         }

         # Crea un objeto GridSearchCV y especifica el modelo a ajustar
         grid_search = GridSearchCV(
             estimator=DecisionTreeClassifier(),
             param_grid=param_grid,
             scoring='accuracy',
             cv=5,
             n_jobs=-1
         )

         # Ajusta el modelo utilizando GridSearchCV
         grid_search.fit(X_train_processed, y_train)

         # Imprime los mejores parámetros y la mejor puntuación obtenida
         print("Mejores parámetros:", grid_search.best_params_)
         print("Mejor puntuación:", grid_search.best_score_)

         Mejores parámetros: {'criterion': 'gini', 'max_depth': 5, 'min_samples_leaf': 4, 'min_samples_split': 4}
         Mejor puntuación: 0.8977332712311753
```

Imatge 1

9. MODELS DE CLASSIFICACIÓ UTILITZATS

Els models de classificació que han estat estat utilitzats són els següents:

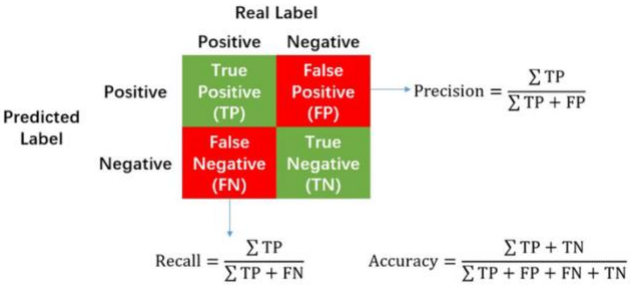
1. DecisionTreeClassifier:
- Es basa en arbres de decisió, on cada node representa una característica i cada branca representa una regla de decisió basada en aquesta característica.
- Divideix el conjunt de dades en funció de les característiques per a formar una estructura d'arbre.
- Pot gestionar tant dades numèriques com categòriques.
- Propens a 8obre ajust8r les dades d'entrenament si no es controla adequadament.
- És ràpid i fàcil d'interpretar.

2. SVC (Support Vector Classifier):
- És un model basat en vectors de suport.
- Busca trobar un 8obre aj que millor separi les diferents classes a l'espai de característiques.
- Pot gestionar dades lineals i no lineals utilitzant diferents funcions de kernel.
- És efectiu en conjunts de dades amb dimensions altes.
- Pot ser computacionalment costós en conjunts de dades grans.

3. XGB (XGBoost):
- És una implementació optimitzada de l'algoritme Gradient Boosting.
- Utilitza una combinació d'arbres de decisió febles i els millora de manera seqüencial mitjançant l'ajust de pesos a cadascun d'ells.
- Proporciona un millor rendiment en comparació amb els arbres de decisió individuals.
- Pot gestionar automàticament característiques mancants i variables categòriques.
- És escalable i té una bona capacitat de generalització.

4. RandomForestClassifier:
- És un conjunt d'arbres de decisió que s'ajusten de forma independent a subconjunts aleatoris de característiques i dades.
- Combina la predicció de cada arbre mitjançant votació o promig, depenent del problema (classificació o regressió).
- Ajuda a reduir el 8obre ajust i proporciona una millor generalització en comparació amb un sol arbre de decisió.
- Pot gestionar característiques numèriques i categòriques.
- És menys propens al 8obre ajust en comparació amb DecisionTreeClassifier.

En general, la elecció del model dependrà del conjunt de dades, el problema específic i els requisits de rendiment. És recomanable experimentar amb diferents models i ajustar els seus hiperparàmetres per determinar quin s'adapta millor a la teva situació particular.



10. ACCURACY, PRECISION I RECALL

Imatge 2

L'accuracy, la precisió i la recall (Imatge 2) són mètriques comunes utilitzades per avaluar el rendiment dels models de classificació en l'àmbit de l'aprenentatge automàtic.

- Accuracy : L'accuracy mesura la proporció de prediccions correctes realitzades pel model en relació al nombre total de prediccions. Es calcula dividint el nombre de prediccions correctes pel nombre total de prediccions. L'accuracy és una mètrica important per determinar la precisió global del model. Un valor

d'accuracy proper a 1 indica un bon rendiment del model, mentre que un valor proper a 0 indica un rendiment deficient.

- Precisió: La precisió mesura la proporció de prediccions positives correctes entre totes les prediccions positives realitzades pel model. Es calcula dividint el nombre de prediccions positives correctes pel nombre total de prediccions positives (prediccions positives correctes + prediccions positives incorrectes). La precisió és particularment útil quan l'objectiu és reduir els falsos positius. Una alta precisió indica que el model té una baixa taxa de falsos positius.

- Recall (recala o record): El recall mesura la proporció de prediccions positives correctes entre tots els exemples positius presents en les dades d'entrada. Es calcula dividint el nombre de prediccions positives correctes pel nombre total d'exemples positius (prediccions positives correctes + falsos negatius). El recall és especialment útil quan l'objectiu és reduir els falsos negatius. Un recall alt indica que el model té una baixa taxa de falsos negatius.

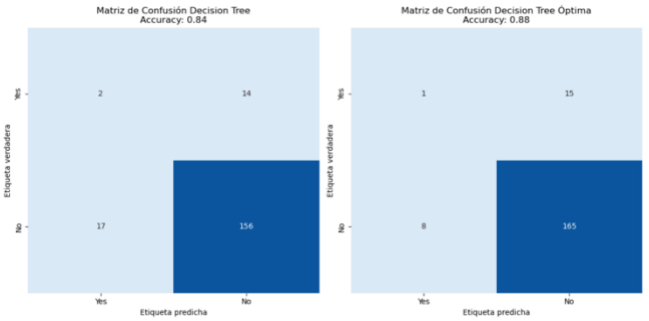
En resum, l'accuracy proporciona una visió general del rendiment global del model, mentre que la precisió i el recall se centren en la qualitat de les prediccions positives. És important considerar totes aquestes mètriques conjuntament per a una avaluació completa del rendiment d'un model de classificació.

11. RESULTATS

Finalment els resultats han estat els següents:

Amb el model de classificació "Decision Tree Classifier" he obtingut els pitjors resultats, només executant el model amb els paràmetres per defecte ja es podia observar com aquest model per aquest cas no és gaire òptim. El model ha tingut una accuracy del 84% que ajustant els paràmetres s'ha pogut augmentar fins a un 88%, però tot i així s'ha quedat amb uns resultats molt inferiors a la resta de models de classificació.

A la imatge (Resultat 1) també podem observar que en ha passat d'encertar 158/189 a encertar 166/189, això ha estat un canvi notori

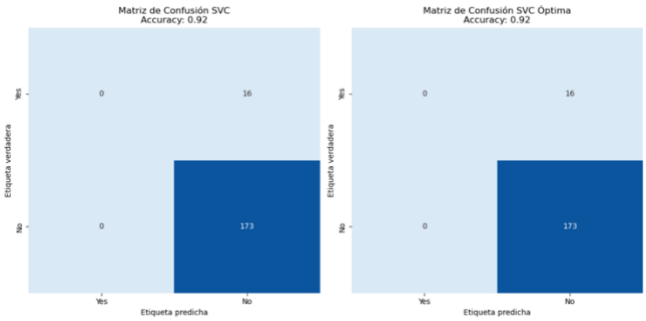


Resultat 1

A diferència del model anterior, la resta han tingut uns resultats pràcticament iguals, amb la mateixa accuracy excepte un, que ha encertat un valor més.

En el cas del model de classificació Support Vector Classification (SVC) , els resultats han estat de una accuracy del 92% tant amb els hiperparàmetres per defecte com amb els hiperparàmetres òptims, és a dir, que els paràmetres òptims són els per defecte.

Aquí a sota, podem veure com el model necessitava més dades d'exemple ja que bàsicament ha predit que tots els clients no són bons clients, i com que la mayoría són clients dolents, ha aconseguit un 92% d'accuracy. També es va provar entrenant el model amb més dades d'entrenament lo que comportava menys dades de test, però tot i així, predia que tots els clients eren dolents.



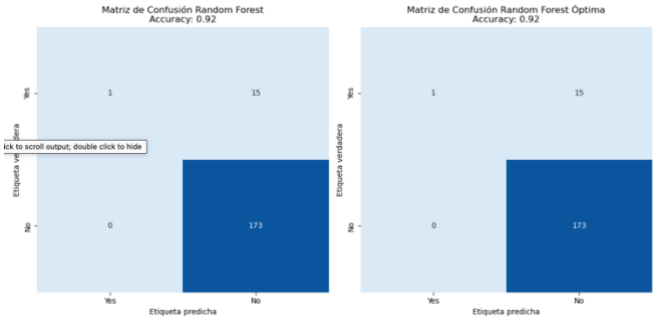
Ha encertat 173/189 que en veritat són els 173 clients que a la variable Loan_Status tenien una B o una D (Resultat 2).

Resultat 2

El següent model utilitzat ha estat el model de classificació Random Forest Classifier el qual ha millorat els resultats respecte el model SVC ja que tot i que ambdós models tenen un 92% d'accuracy és perquè no estàn els decimals, ja que el model Random Forest sí que ha encertat un client bo pel banc.

El model amb els paràmetres per defecte ha encertat 1/1 que el client era bo. A la resta ha posat que eren clients dolents. No sembla un bon model o sembla que es necessiten més dades per a poder entrenar el model correctament, ja que tot i que ha encertat a un client bo pràcticament no s'arrisca.

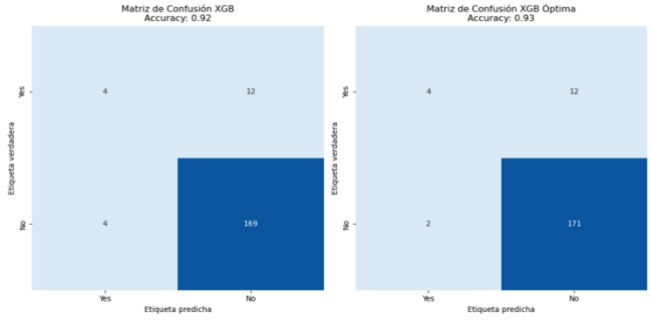
Ha encertat 174/189 resultats, és millor que el CVC però no és un model ideal per a realitzar aquesta tasca (Resultat 3).



RESULTAT 3

Finalment, tenim el model de XGB el qual ha aconseguit ni més ni menys que un 93% d'accuracy quan s'utilitzen els paràmetres òptims.

Amb els paràmetres per defecte s'ha quedat amb un 92% d'accuracy. Això sí, el model XGBoost (XGB) s'ha arriscat i ha encertat 4 clients bons pel banc tant amb els paràmetres per defecte com amb els òptims, la diferència ha estat que el model per defecte ha fallat 4 clients que no eren clients bons però ell els ha catalogat com a bons clients pel banc (Resultat 4).



Resultat 4

12. CONCLUSIONS

En conclusió, després de realitzar un estudi exhaustiu i comparar els models de classificació XGBoost (XGB) amb Random Forest, Decision Tree i SVC, he arribat a la decisió de quedar-me amb el model XGB com la millor opció per a aquesta tasca.

Durant el procés d'avaluació, XGB ha demostrat tenir el millor rendiment dels 4 models de classificació. En primer lloc, s'ha destacat per la seva gran precisió en la classificació, aconseguint resultats consistents i confiables.

D'altra banda, XGB ha destacat per la seva capacitat per afrontar problemes de desequilibri de classes, una característica important en el nostre conjunt de dades. Ha estat capaç d'adaptar-se i aprendre eficientment dels casos minoritaris, evitant una possible sobreclassificació dels casos més freqüents.

També és important destacar la seva rapidesa en l'entrenament i la inferència. XGB ha demostrat ser més eficient en termes de temps de processament, permetent-nos obtenir resultats en un període de temps més curt. Això és especialment rellevant quan es tracta d'analitzar conjunts de dades més grans o quan es requereixen actualitzacions freqüents del model.

En resum, després de comparar els models de classifica ció XGB, Random Forest, Decision Tree i SVC, he conclòs que XGB és la millor opció per a la tasca en qüestió. La seva precisió, robustesa enfront de desequilibris de classes i eficiència en termes de temps de processament el converteixen en una elecció òptima per a aquest estudi.

Agraïments

Vull expressar el meu sincer agraïment a totes les persones que han contribuït i donat suport al desenvolupament d'aquest treball de Fi de Grau.
Primer de tot, vull agrair al meu tutor acadèmic, Daniel Franco, per la seva orientació, suport i paciència al llarg de tot el procés d'investigació i totes les presentacions col·lectives realitzades durant aquests mesos. També m'agradaria agrair a en Antonio Espinosa que ha estat sempre implicat en el meu trball i en el dels meus companys i finalment vull agrair a tothom que hagi estat implicat en el meu aprenentatge durant aquests 4 anys, tant als professors, com als amics com als companys.

BIBLIOGRAFIA

[1] https://sorry.vse.cz/~berka/challenge/pkdd1999/berka.htm
[2] https://towardsdatascience.com/loan-default-prediction-an-end-to-end-ml-project-with-real-bank-data-part-1-1405f7aecb9e
[3] https://github.com/zhouxu-ds/loan-default-prediction
[4] https://github.com/justinng1/berka
[5] https://webpages.charlotte.edu/mirsad/itcs6265/group1/domain.html
[6] https://relational.fit.cvut.cz/dataset/Financial

APÈNDIX

A1. DESCRIPCIÓ DELS ATRIBUTS DE LES TAULES:

- **Compte:** Els comptes tenen 4 atributs els quals són:
  - account_id: Identificació del compte
  - district_id: Ubicació de la sucursal
  - date: Dia de creació del compte en format YYMMDD
  - frequency: Càrregues i abonaments

- **Client:** Els clients tenen 4 atributs els quals són:
  - client_id: Identificació del client
  - district_id: Ubicació del client
  - birth_number: Sexe i naixement del client, en format YYMMDD pels homes i YYMMDD+50DD per les dones.

- **Districte:**
  - district_id: Codi del districte
  - A2: Nom del districte
  - A3: Regió
  - A4: Nombre d'habitants
  - A5: Nombre de municipis amb menys de 500 habitants
  - A6: Nombre de municipis entre 500 i 1999 habitants
  - A7: Nombre de municipis entre 2000 i 9999 abitants
  - A8: Nombre de municipis amb més de 10000 habitants
  - A9: Nombre de ciutats
  - A10: Proporció d'habitants urbans
  - A11: Salari mitjà
  - A12: Taxa d'atur al 1995
  - A13: Taxa d'atur al 1996
  - A14: Nombre d'empresaris per 1000 habitants
  - A15: Nombre de delictes comesos al 1995
  - A16: Nombre de delictes comesos al 1996

- **Préstec:**
  - loan_id: Identificació del registre del préstec
  - account_id: Identificació del compte
  - duration: Duració del préstec
  - amount: Quantitat del préstec
  - payments: Pagaments mensuals del préstec
  - status: Estat del pagament del préstec: "A" (contracte finiquitat sense problemes), "B" (contracte finiquitat préstec no pagat), "C" (contracte en curs, tot bé pel moment), "D" (contracte en curs, client en deute)
  - date: Data en que el préstec ha estat constituït

- **Comanda:**
  - order_id: Identificació del registre de la comanda
  - account_id: Identificació del compte
  - account_to: Titular del compte
  - amount_to: Import transferit
  - k_symbol: tipus de pagament: "POJISTNE" (pagament de l'assegurança), "UVER" (pagament del préstec), "LEASING" (arrendament), "SIPO" (pagament de la llar)
  - bank_to: Entitat destinatària

- **Disposició:**
  - disp_id: Identificació del registre
  - client_id: identificació del client
  - account_id: Identificació del compte
  - type: tipus de disposició: propietari o usuari

- **Transacció:**
  - trans_id: Identificació del registre de la transacció
  - account_id: Identificació del compte
  - client_id: Identificació del client
  - type: Tipus de transacció: "PRIJEM" (ingressar capital), "VYDAJ" (retirar capital)
  - date: Data de la transacció en el format YYMMDD
  - amount: Quantitat de la transacció
  - balance: Capital un cop realitzada la transacció
  - operation: Mode en el que s'ha realitzat la transacció
  - k_symbol: Caracterització de la transacció
  - bank: Banc del soci
  - account: Compte del soci

- **Trageta de crèdit:**
  - card_id: Identificació de la targeta de crèdit
  - disp_id: Disposició al compte
  - type: Tipus d etargeta d ecrèdit: "gold", "junnior", "classic"

ISSUED: 23/07/02