Informe del projecte d'Aprenentatge Computacional

Grup 11 - Detecció d'Ictus a Través de Models d'Aprenentatge Computacional

David Morillo, Adrià Muro i Lucia Garrido

Índex de l'informe

Context Mèdic

L'ictus, o accident cerebrovascular, és una de les principals causes de mort i discapacitat arreu del món. Es produeix quan el subministrament de sang al cervell es veu interromput, causant danys cerebrals que poden ser irreversibles si no es detecten i tracten ràpidament. Hi ha dos tipus principals d'ictus: isquèmic (causat per un bloqueig) i hemorràgic (causat per un vessament de sang al cervell).

Factors de risc com hipertensió, diabetis, obesitat, hàbits de vida no saludables i antecedents mèdics familiars augmenten significativament la probabilitat de patir un ictus. Tot i això, moltes persones amb risc no tenen accés a diagnòstics preventius o assistència mèdica regular.

A través de l'ús de dades demogràfiques i mèdiques bàsiques, aquest projecte busca aprofitar la intel·ligència artificial per omplir aquest buit i oferir eines per a la detecció precoç i accessible, especialment en comunitats amb recursos limitats.

Objectius Del Projecte

La detecció d'ictus és un repte crític en l'àmbit mèdic, ja que la seva identificació precoç pot salvar vides i reduir danys irreversibles. A través de l'ús de models de Machine Learning, aquest projecte pretén aportar una eina que ajudi a identificar pacients amb risc d'ictus a partir de dades clíniques i demogràfiques.

El projecte posa especial èmfasi en dos aspectes:

- **Accesibilitat**: Estudiar si models menys dependents de dades estrictament mèdiques poden oferir una detecció preliminar fiable.
- **Eficàcia**: Prioritzar el record (recall) en les prediccions per assegurar una detecció òptima dels casos positius, reduint al mínim el risc de falsos negatius.

Aquest enfocament té el potencial d'ampliar l'abast de les eines diagnòstiques a entorns amb recursos limitats, oferint una solució preventiva que pugui salvar vides.

Planificació del Treball

La planificació d'aquest projecte es va dividir en diverses fases per garantir un desenvolupament estructurat i eficient. Cada etapa es va completar seguint un cronograma preestablert i ajustant les tasques en funció dels resultats obtinguts i els desafiaments trobats. A continuació, es detallen les principals fases del projecte:

Fase 1: Preparació del Dataset

Tasques realitzades:

- Exploració inicial del dataset per entendre les característiques i les seves distribucions.
- Tractament de valors manquants i anomalies, com dades extremes o inconsistents.
- Codificació de variables categòriques per facilitar el seu ús en models de Machine Learning.
- Divisió del dataset en conjunts d'entrenament, validació i test.
- Aplicació de tècniques per abordar el desbalanceig de classes, com SMOTE.

Fase 2: Selecció i Entrenament de Models

Tasques realitzades:

- Definició dels algoritmes inicials a utilitzar: Logistic Regression, Random Forest, KNN, Naive Bayes, AdaBoost i XGBoost.
- Entrenament dels models amb els conjunts de dades preparats.
- Avaluació preliminar dels models utilitzant mètriques com Accuracy, Recall i AUC-ROC.

Fase 3: Optimització i Avaluació

Tasques realitzades:

- Ajust del threshold dels models per prioritzar el recall en detriment d'altres mètriques, assegurant la detecció dels casos positius.
- Comparació dels resultats amb i sense certes característiques per identificar-ne l'impacte en el rendiment dels models.
- Generació de gràfics comparatius, com les corbes ROC i matrius de confusió, per analitzar les diferències entre models.

Fase 4: Visualització i Interpretació dels Resultats

Tasques realitzades:

- Creació de gràfics de suport per il·lustrar els resultats obtinguts.
- Interpretació de les mètriques per destacar els punts forts i les limitacions de cada model.
- Redacció de conclusions basades en els resultats i en la viabilitat d'aplicar els models en entorns clínics i no clínics.

Fase 5: Redacció i Presentació

Tasques realitzades:

- Documentació de tot el procés i resultats en aquest README.
- Preparació d'una presentació clara i concisa per exposar el projecte en 10 minuts, seguint un esquema lògic i amb suport visual.
- Assaigs previs de la presentació per ajustar el temps i anticipar preguntes.

Dataset Utilitzat

El dataset utilitzat prové de **Kaggle** i conté dades de 5.110 pacients amb les següent característiques:

- **id**: Identificador únic del pacient.
- **gender**: "Male", "Female" o "Other".
- **age**: Edat del pacient.
- **hypertension**: 0 si el pacient no té hipertensió, 1 si en té.
- **heart_disease**: 0 si no té malalties cardíacques, 1 si en té.
- **ever_married**: "No" o "Yes".
- **work_type**: "children", "Govt_job", "Never_worked", "Private" o "Self-employed".
- **Residence_type**: "Rural" o "Urban".
- **avg_glucose_level**: Nivell mitjà de glucosa a la sang.
- **bmi**: Índex de massa corporal.
- **smoking_status**: "formerly smoked", "never smoked", "smokes" o "Unknown".
- **stroke**: 1 si el pacient ha patit un ictus, 0 en cas contrari.

El dataset presentava un desbalanceig significatiu: només aproximadament 250 casos (5%) tenien *stroke = 1*. Aquest problema es va abordar aplicant **SMOTE** (Synthetic Minority Over-sampling Technique) per equilibrar les classes durant l'entrenament, garantint que les proves reflectissin la realitat i no un escenari artificialment balancejat.

Llibreries Utilitzades

- **numpy**, **pandas**: Per a la gestió i manipulació de dades.
- **scikit-learn**: Per a la divisó de dades, implementació de models i càlcul de mètriques.
- **imblearn**: Per aplicar SMOTE.
- **matplotlib**, **seaborn**: Per a la generació de gràfics.
- **Mòduls personalitzats**:
    - **dataloader_module**: Carrega i prepara les dades.
    - **metrics_module**: Calcula les mètriques necessàries.
    - **graphs_module**: Genera gràfics com ROC curves.

Models de d'AC Aplicats

- Logistic Regression
- Random Forest
- Naive Bayes
- K-Nearest Neighbors (KNN)

- AdaBoost
- XGBoost

Els models es van avaluar, per una banda amb totes les característiques, i per altra amb les següents columnes excloses (dades clíniques):

- bmi
- obesity
- avg_glucose_level
- hypertension
- heart_disease

Els criteris d'èxit incloïen la **accuracy**, el **recall** (prioritzat en aquest projecte), i l'àrea sota la corba **ROC** (AUC-ROC) per a una supervisió d'un comportament esperat.

No-Implementació de Grid Search

Durant el desenvolupament del projecte, vam considerar la possibilitat d'utilitzar GridSearchCV per a la cerca d'hiperparàmetres amb l'objectiu de maximitzar tant el recall com l'accuracy de manera equilibrada. No obstant això, després de diverses proves i avaluacions, vam observar que la combinació d'aquestes dues mètriques en la cerca d'hiperparàmetres no produïa els resultats esperats.

En concret, vam trobar que:

- **Mètriques Nefastes**: Quan intentàvem maximitzar simultàniament el recall i l'accuracy, els models resultants presentaven mètriques de rendiment subòptimes. Això es devia al fet que la combinació d'aquestes dues mètriques en una sola funció d'avaluació no reflectia adequadament el compromís necessari entre elles.

- **Compromís entre Recall i Accuracy**: Maximitzar el recall sovint implica acceptar un major nombre de falsos positius, mentre que maximitzar l'accuracy pot implicar un compromís en la detecció de casos positius. Aquest compromís inherent va dificultar la cerca d'una configuració d'hiperparàmetres que equilibrés ambdues mètriques de manera satisfactòria.

- **Complexitat i Temps de Càlcul**: La cerca d'hiperparàmetres amb GridSearchCV és computacionalment intensiva, especialment quan es treballa amb conjunts de dades grans i graelles d'hiperparàmetres extenses. Els resultats subòptims obtinguts no justificaven el temps i els recursos invertits en aquesta cerca.

Per aquestes raons, vam decidir no implementar GridSearchCV en la nostra metodologia final. En lloc d'això, vam optar per ajustar-ho manualment a través del threshold, prioritzant el recall per assegurar-nos que els casos positius fossin detectats de manera efectiva, mentre manteníem una accuracy acceptable.

Aquesta decisió ens va permetre obtenir models amb un millor rendiment global, adaptats a les necessitats específiques del projecte, sense comprometre excessivament cap de les dues mètriques clau.

No-Implementació de models de Clustering

Des d'un inici es va voler implementar com a model secundari (recolzant un model de classificació binari), el model de **K-means**. Aquest faria servir característiques numèriques i categòriques per a agrupar als pacients en grups, i amb ajuda del ground truth, separar aquests en grups per tractaments diferents. Junt amb la resta del projecte aquesta idea va anar avançant. A la segona setmana d'estar-hi treballant, havíem estat experimentant amb mètodes de visualització com PCA i t-SNE.

Tot i els esforços inicials, després d'una reunió d'equip i la revisió dels resultats preliminars, es va decidir abandonar la idea per les següents raons:

- **Viabilitat de la Separació de Grups**: No vam veure viable la separació clara entre grups útils per al tractament basat en les dades disponibles. Tot i la visualització, els grups generats pel K-means no mostraven una estructura clara que permetés una diferència significativa per al diagnòstic o tractament de l'ictus.
- **Complexitat Afegida**: Introduir un model de clustering amb dades de pacients que ja tenien un diagnòstic definit podria afegir complexitat sense aportar un valor addicional clar. L'objectiu principal del projecte era la detecció de l'ictus a partir de dades disponibles, no la segmentació de pacients per altres criteris. A part, seria gairebé impossible associar un tipus de tractament a un pacient, donat el nostre coneixement nul en medicina, i no sabriem el nombre de clústers que necessitariem per començar. Al no tenir un ground truth, tampoc podríem verificar que les nostres prediccions fóssin les correctes.
- **Inestabilitat en les Assignacions de Grups**: El model de clustering presentava instabilitat en les assignacions de grups, especialment quan es treballava amb dades més petites (les de test) o amb dades amb molt soroll. Això va dificultar l'ús d'aquests grups com a base per al tractament.

Per aquests motius, vam decidir no implementar el model de clustering a la metodologia final del projecte. En lloc d'això, ens vam centrar en els models de classificació binària, que ens permetien obtenir resultats més fiables i directament relacionats amb la predicció d'ictus.

Reproducció del Projecte

Aquest apartat detalla com reproduir el projecte pas a pas i com accedir als resultats obtinguts. Hem dissenyat el projecte perquè sigui fàcil d'executar per a qualsevol persona amb coneixements bàsics de Python i Machine Learning.

Requisits

Python 3.8+: Per executar els scripts i notebooks, cal tenir instal·lada una versió moderna de Python (3.8 o superior). Podeu descarregar-lo des de python.org.

Llibreries necessàries: Les dependències es poden instal·lar fàcilment executant la següent comanda al terminal des del directori principal del projecte:

pip install -r requirements.txt

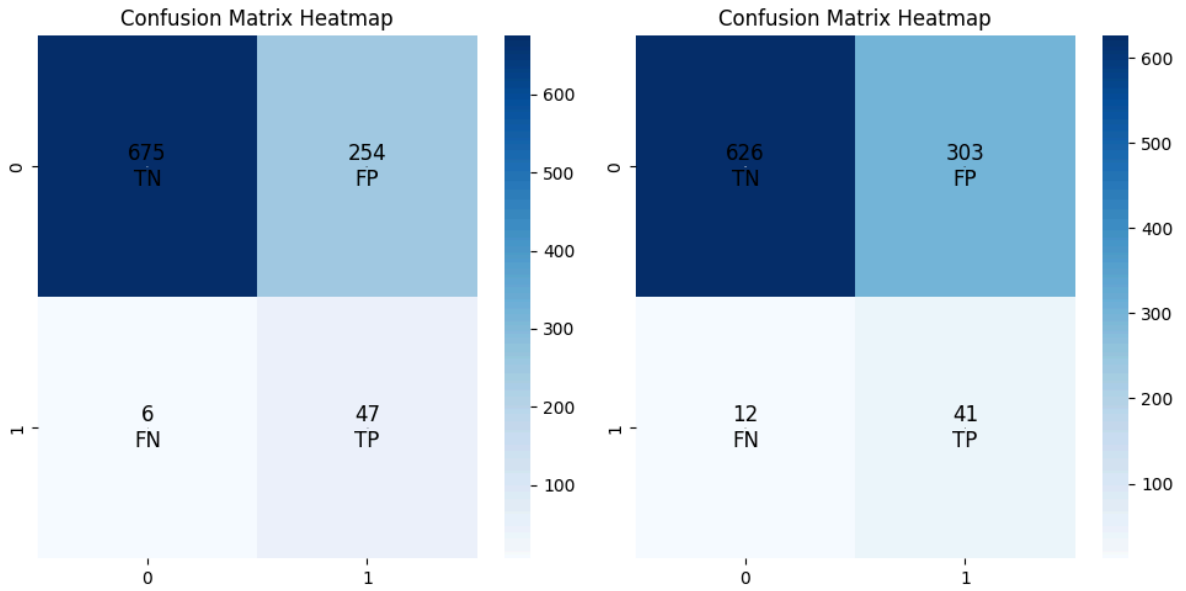
Resultats

Els resultats inclouen:

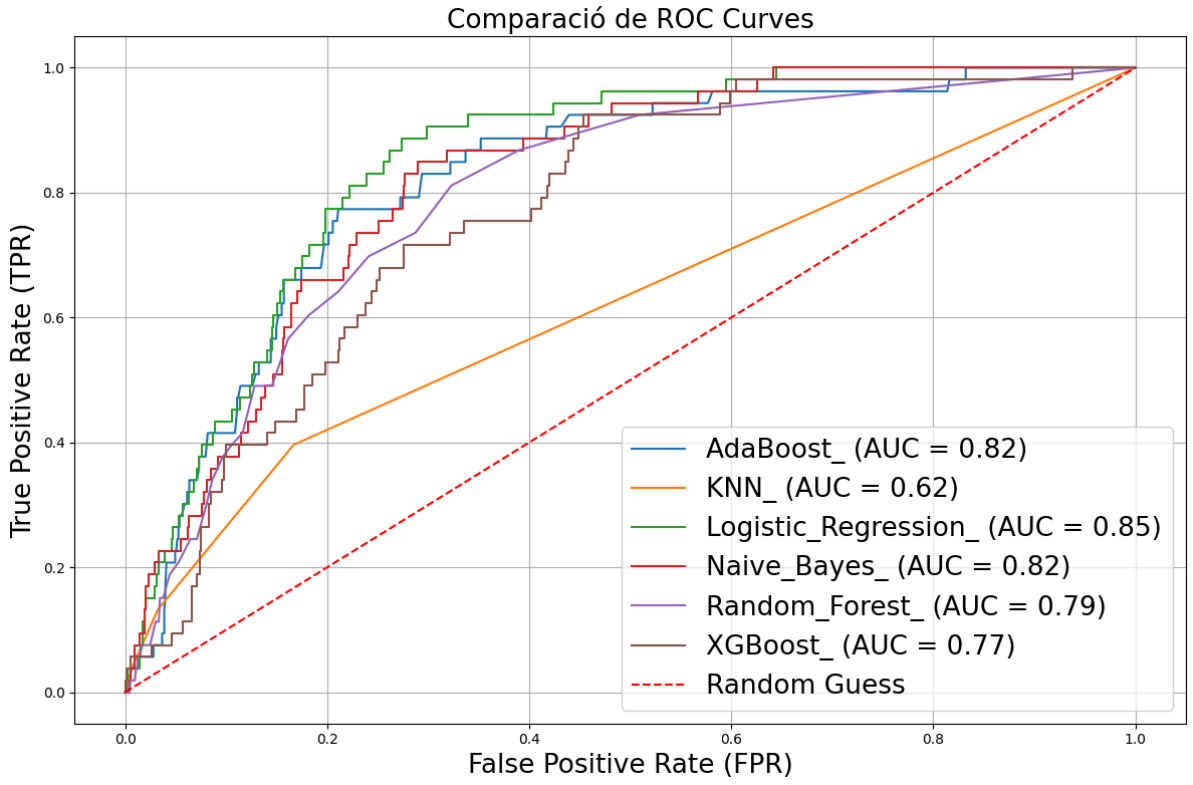
- **Mètriques dels millors models seleccionats:**

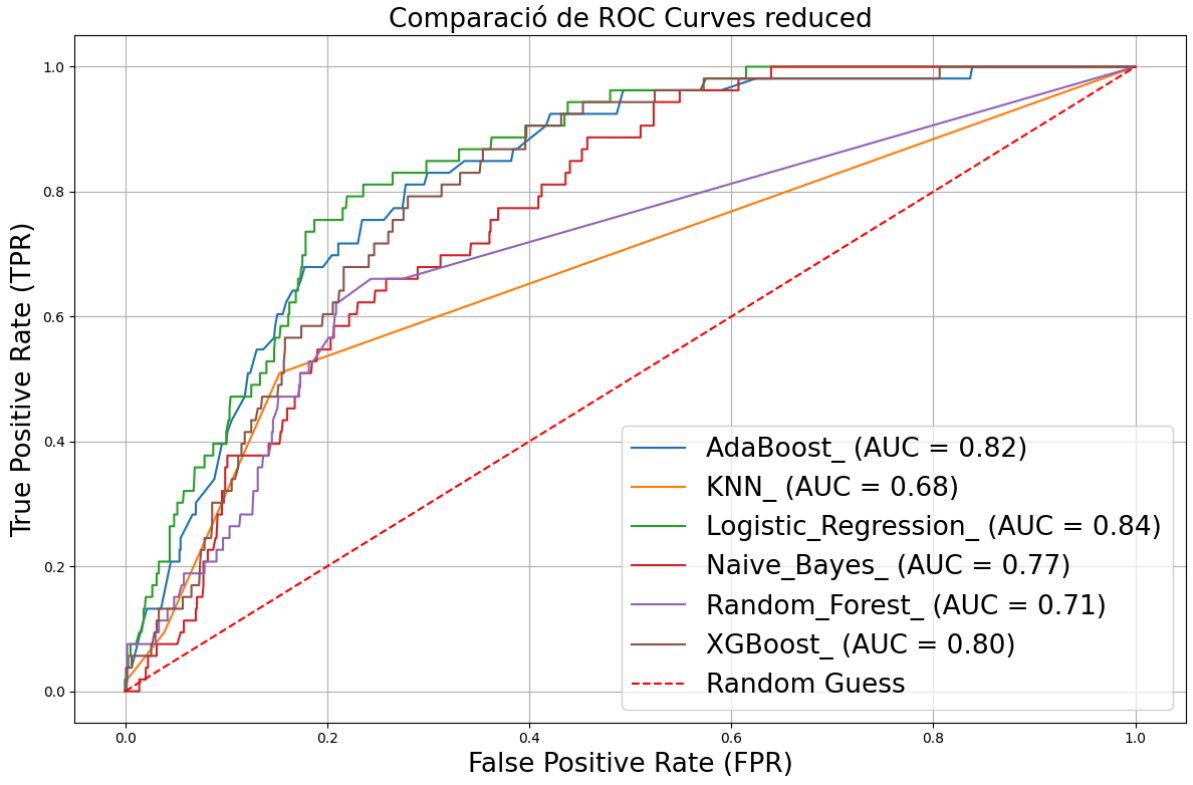
	Logistic Regression	XGBoost
Precision	0.1561	0.1192
Accuracy	0.7352	0.6792
Recall	0.8868	0.7736
F1 Score	0.2655	0.2065

- **Matrius de confusió que mostren dels millors models:**

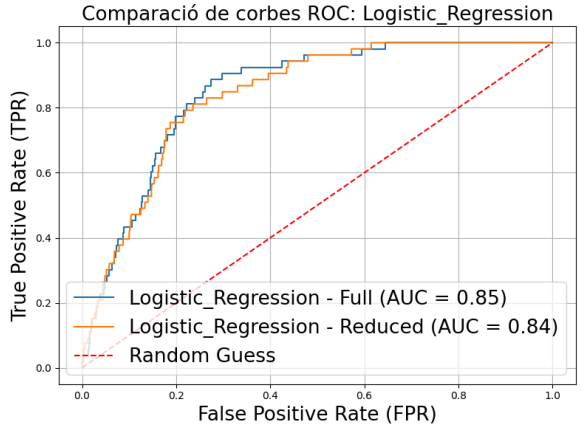


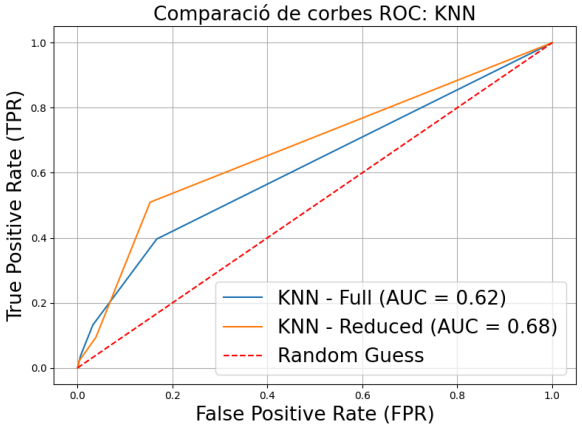
- **ROC curves que comparen tots els models amb les dades completes i amb les dades reduïdes:**

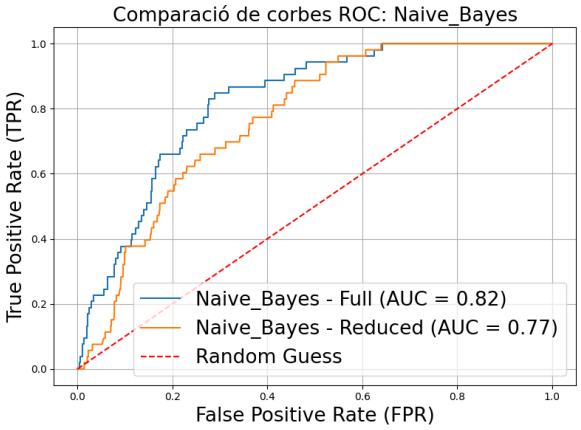


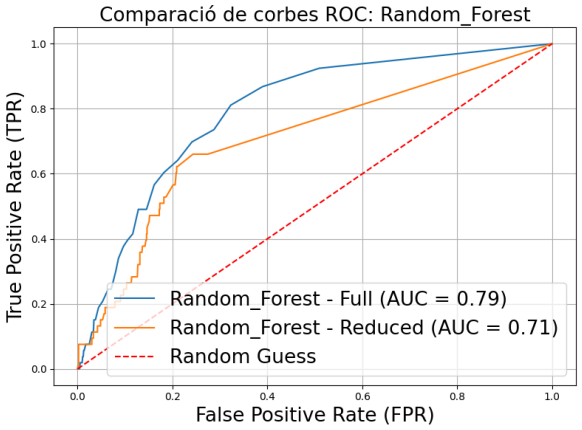


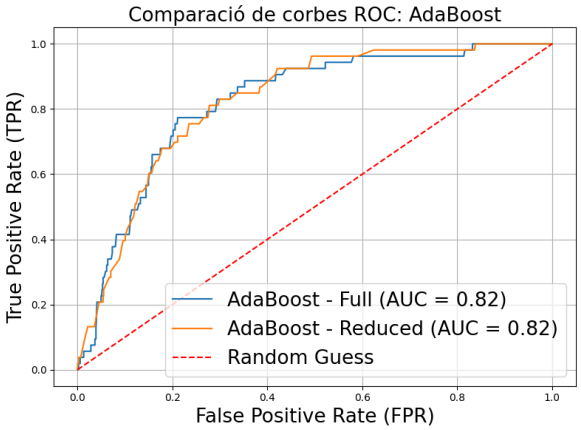
- **Comparatives de corbes ROC entre models:**

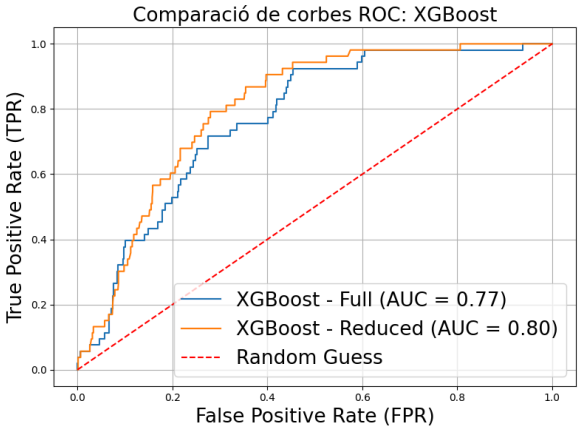












Observació dels Resultats

Per analitzar el rendiment dels models en diferents condicions, vam generar gràfiques ROC tant utilitzant totes les característiques com després de reduir-les. Aquestes comparatives ens permeten entendre millor com afecta la quantitat d'informació disponible a la capacitat predictiva dels models, i identificar quins són més sensibles o robustos davant aquests canvis.

Observacions sobre les gràfiques ROC comparant tots els models

1. **Comparativa amb totes les característiques:**
    - Les gràfiques ROC mostren que, quan s'utilitzen totes les característiques disponibles, els models tenen una capacitat notable de predicció d'ictus. Això es tradueix en corbes més properes a l'angle superior esquerre, indicant una elevada sensibilitat i especificitat.
    - Tant la Regressió Logística com XGBoost aconsegueixen àrees sota la corba (AUC) altes, destacant com els models més eficients del conjunt.

2. **Comparativa amb reducció de característiques:**
    - Quan es redueixen les característiques, sorprenentment, els models continuen sent capaços de predir ictus amb un grau de fiabilitat raonable, cosa que suggereix que les característiques de caràcter mèdic tenen menys pes del que es pensava.

Observacions sobre les gràfiques ROC entre models

Per avaluar com afecta la reducció de característiques al rendiment dels models, vam decidir generar gràfiques ROC que comparen cada model abans i després d'eliminar característiques. Aquest enfocament ens permet analitzar visualment com canvia la capacitat predictiva dels models, ajudant-nos a identificar quins són més robustos davant la pèrdua d'informació.

1. **Models que van millorar:**
    - **KNN** i **XGBoost** van mostrar una millora inesperada en la seva capacitat predictiva. Això pot ser degut a l'eliminació de variables sorolloses que afectaven negativament el rendiment. La simplificació del conjunt de dades sembla haver optimitzat la seva eficàcia.

2. **Models que van empitjorar:**
    - **Naive Bayes** i **Random Forest** van experimentar una disminució notable en el rendiment. Com s'esperava, la pèrdua d'informació va tenir un impacte negatiu, especialment en aquests models, que depenen fortament d'un

conjunt de dades complet i ric en característiques per maximitzar la seva precisió.

3. **Models equilibrats:**
    - **Regressió Logística** i **AdaBoost** van mantenir un rendiment relativament equilibrat. Aquest comportament indica que aquests models són menys sensibles a la pèrdua d'informació, probablement gràcies a la seva naturalesa estadística o al seu disseny per gestionar dades menys complexes.

Els models mostren una disminució de la precisió quan es redueixen les característiques, tot i que encara és possible detectar ictus amb una fiabilitat raonable. Els models de **Regressió Logística** i **XGBoost** van ser els més robustos, especialment quan es prioritzava el recall.

Conclusions

- És possible detectar casos d'ictus amb dades que tenim, amb una **Accuracy** de **74%** i un **Recall** del **89%** per la regressió logística.
- La priorització del recall és clau per minimitzar els falsos negatius, donada la gravetat d'un ictus no detectat.
- Tot i que les mètriques com l'accuracy i la precisió no han estat prioritzades en aquest projecte, som conscients que els seus valors han estat relativament baixos. Aquesta situació és conseqüència del compromís inherent entre maximitzar el recall i mantenir un bon rendiment global. Tot i això, considerem que millorar aquests aspectes podria oferir models amb un rendiment més equilibrat, i es planteja com un objectiu a abordar en futures iteracions del projecte
- És possible detectar ictus amb característiques mesurables a casa, encara que els valors de les mètriques disminueixin lleugerament, depenent del model.

Millora Futura

Incorporar dades addicionals per augmentar la robustesa dels models, especialment de casos positius (ictus). Això inclou més tipus de dades (historial alimentari, activitat física, etc.), i quantitats més grans de dades de pacients amb ictus.

Repositori

El repositori es troba a GitHub i té la següent estructura:

```
├── data/                    # Conté el dataset
├── not_implemented/         # Scripts o idees que no es van arribar a implementar
├── notebooks_and_scripts/   # Notebooks dels diferents models utilitzats en el projecte per predir ictus
├── ROC_Data/                # Conjunt de dades utilitzats per representar les ROC Curves
├── .gitignore               # Fitxer que exclou arxius del control de versions
├── README.md                # Documentació principal
└── requirements.txt         # Llista de dependències
```