Progetto di Classificazione delle Stelle Variabili

[Progetto ML e SII – Damiano di Padova 559784, Gabriele Rizzitiello 546817]

- Link al progetto: https://github.com/damdip/VariableStarClassification

Contesto

Le stelle variabili, astri che esibiscono fluttuazioni di luminosità nel tempo, rivestono un ruolo cruciale nell'astrofisica moderna. L'analisi delle loro variazioni di luce fornisce informazioni preziose sulla loro struttura interna, evoluzione e distanze cosmiche. La classificazione accurata di queste stelle è quindi fondamentale per una vasta gamma di studi, dalla determinazione della scala delle distanze nell'universo, all'indagine sulla formazione stellare e sulla fisica degli ambienti estremi.

Stato dell'arte

Le tecniche tradizionali di classificazione delle stelle variabili si basano su metodi manuali e sull'analisi di caratteristiche specifiche delle curve di luce. Tuttavia, l'avvento di survey astronomici di nuova generazione, come il Large Synoptic Survey Telescope (LSST), genera enormi volumi di dati che rendono impraticabili i metodi manuali. Di conseguenza, l'applicazione di tecniche di machine learning (ML) è diventata essenziale per automatizzare e accelerare il processo di classificazione.

Introduzione - Cos'è una stella variabile?

Una stella variabile è una stella la cui luminosità apparente cambia nel tempo. La **luminosità apparente** (o magnitudine apparente) è la misura della luminosità di un corpo celeste osservata dalla Terra. Maggiore è la luminosità del corpo celeste, minore è la sua magnitudine. Per esempio, Sirio è la stella più luminosa del cielo notturno nello spettro visibile, mentre Betelgeuse domina nel vicino infrarosso (banda J). La banda V, simile alla sensibilità dell'occhio umano, è la più comunemente utilizzata per le osservazioni.

Questo accade perché un oggetto celeste estremamente luminoso può apparire debole se a grande distanza. La **magnitudine assoluta** esprime invece la luminosità intrinseca di un corpo celeste. Ad esempio, la magnitudine apparente di Sirio visto da una distanza di 1 UA è -30,30, mentre osservato dalla Terra varia tra -1,44 e -1,46.

Le variazioni di luminosità delle stelle variabili possono andare da pochi millesimi a venti magnitudini in periodi che vanno da frazioni di secondo ad anni. Le stelle variabili si dividono in:

• **Variabili intrinseche**: la cui luminosità cambia realmente, ad esempio a causa di cambiamenti nelle dimensioni dell'astro. Esempi includono le variabili pulsanti, eruptive e cataclismiche.

• **Variabili estrinseche**: il cui cambiamento di luminosità è apparente, dovuto a fattori esterni come eclissi da parte di una compagna in un sistema binario. Esempi includono le variabili a eclisse e rotazionali.

Le stelle variabili sono fondamentali per la comprensione dell'astrofisica stellare, poiché permettono di studiare le proprietà fisiche delle stelle e i processi dinamici al loro interno. Anche il Sole varia la sua luminosità dello 0,1% durante il suo ciclo undecennale.

Obiettivi

Per il progetto in questione sono stati considerati diversi approcci di machine learning riferiti al task di classificazione di stelle variabili, in particolare: **Random Forests ,Multinomial Logistic Regression** , **Reti Neurali Artificiali**, **K-Nearest Neighbors (KNN)**

Struttura del Progetto

La struttura del progetto è stata progettata per favorire la modularità e la manutenibilità del codice:

ClassificationOfVariableStars/

|-- **data**/

| |-- raw/ # Dati grezzi originali

| |-- processed/ # Dati preprocessati

|-- **analysis**/ # Analisi dei dati

| |-- data_preparation.py # Pulizia dei dati

| |-- data_plotting.py

|-- **src**/ # Script principali

| |-- data_loader.py # Caricamento dei dati

| |-- preprocessing.py # Preprocessing dei dati

| |-- cross_validation.py # Implementazione K-fold

|-- **models**/ # Modelli salvati

| |--final_model.pkl # Implementazione K-fold

|-- variable_star_classifier.py # Modello Random Forest

|-- variable_star_classifier_knn.py # Modello KNN

|-- README.md # Descrizione del progetto

Preparazione dei Dati

I dati sono stati caricati da un file CSV utilizzando pandas. Il dataset LINEAR contiene 7194 stelle variabili con variabili come periodo (P), ampiezza (A), e tipo di curva di luce (LCtype).

Dataset

Vediamo nello specifico quali sono le variabili di nostro interesse presentate da LINEAR:

"*LINEARobjectID*" è un ID interno che può essere utilizzato per trovare i dati della curva di luce corrispondenti.

"*LCtype*" è il tipo di curva di luce dalla classificazione visiva (i numeri tra parentesi indicano i conteggi):

• 0 = altro (318, inclusi alcuni quasar);

• 1 = RR Lyr ab (2923);

• 2 = 2 RR Lyr c (990);

• 3 = Algol-like con 1 minimo (20);

• 4 = Algol-like con 2 minimi (357);

• 5 = binario di contatto (2385);

• 6 = delta Scu/SX Phe (112);

• 7 = variabile di lungo periodo, incluse variabili semi-regolari (77);

• 8 = candidati heartbeat (1);

• 9 = BL Her (6);

[manca la classe 10]

• 11 = Cefeidi anomale (5); "P" è il periodo più adatto (confermato visivamente). Quando il periodo non può essere determinato in modo affidabile, viene impostato su -9,90 (6958 sorgenti, da 7194, hanno P>0, con minP=0,042d e maxP=1964 d).

"*A*" è l'ampiezza della curva di luce, stimata in modo non parametrico dalla distribuzione di magnitudine cumulativa come intervallo tra i punti 5\% e 95\%

"*mmed*" è la magnitudine LINEARE mediana

"*stdev*" è la deviazione standard delle magnitudini LINEARI (utilizzando tutti i punti)

"*rms*" è la stima robusta della deviazione standard basata sull'intervallo interquartile

"*Lchi2pdf*" è il logaritmo in base 10 del chi2 per grado di libertà calcolato assumendo nessuna variazione (intorno alle magnitudini medie non ponderate) e utilizzando stime di incertezza di magnitudine LINEARE (i candidati sono stati selezionati utilizzando chi2pdf > 3)

"*nObs*" è il numero di osservazioni LINEARI "skew e kurt" sono asimmetria e curtosi nelle curve di luce LINEARI.

"*LR*" è analogo a Lchi2pdf, eccetto per il fatto che il 5 percento dei punti più periferici è escluso dal calcolo. "CUF" è il flag di incertezza della classificazione, nell'intervallo 0-5. Se CUF>1, allora t2 (più probabile) e t3 (meno probabile) sono offerti come tipi alternativi (6142 voci hanno CUF<=1).

Pulizia e analisi dei Dati

- Rimozione dei valori NaN e dei duplicati
- Normalizzazione e standardizzazione per migliorare le performance del modello.

Feature Engineering

- Selezione delle feature più rilevanti tramite analisi di correlazione.

Implementazione dei Modelli

Random Forest (RF)

- Modello: RandomForestClassifier da scikit-learn.
- K-Fold Cross Validation (k=5) per valutare la generalizzabilità.

K-Nearest Neighbors (KNN)

- Modello: KNeighborsClassifier da scikit-learn.
- Ottimizzazione del parametro k con ricerca sistematica.

Multinomial Logistic Regression

- Modello: Logistic regression da scikit-learn.
- K-Fold Cross Validation (k=5) per valutare la generalizzabilità.

Reti neurali Fully Connected

- Modello: Costruito con keras.
- Tre layer, il primo con 32 nodi, il secondo con 16 e il terzo con il numero di classi da classificare, funzione ReLu e softmax per l'ultimo strato. 50 epoche con dimensione del batch pari a 8 istanze

Valutazione dei Modelli

Metriche di Valutazione

- Accuracy, Precision, Recall, F1-Score.

Interpretazione dei Risultati

Analisi delle metriche per identificare punti di forza e debolezza, relativi al confronto tra i modelli:

• **Random Forest**:

- Vantaggi: Robusto, gestisce dati complessi.
- Svantaggi: Computazionalmente costoso (si intende per dataset più corposi), rischio di overfitting.

• **KNN**:

- Vantaggi: Semplice, non richiede addestramento esplicito.
- Svantaggi: Prestazioni scadenti e insensibili alla variazione del parametro k.

• **FC**:

- Vantaggi: Buone prestazioni per i dati tabellari anche con poca “manutenzione” su struttura della rete e iperparametri.
- Svantaggi: Addestramento più costoso e lungo rispetto agli altri modelli (dipendenza dal numero di epoche).

Conclusioni

Il progetto ha dimostrato l'efficacia del machine learning nella classificazione delle stelle variabili. Il modello Random Forest e a seguire la rete neurale hanno mostrato prestazioni superiori grazie alla sua robustezza, mentre il KNN ha evidenziato una maggiore sensibilità ai parametri.

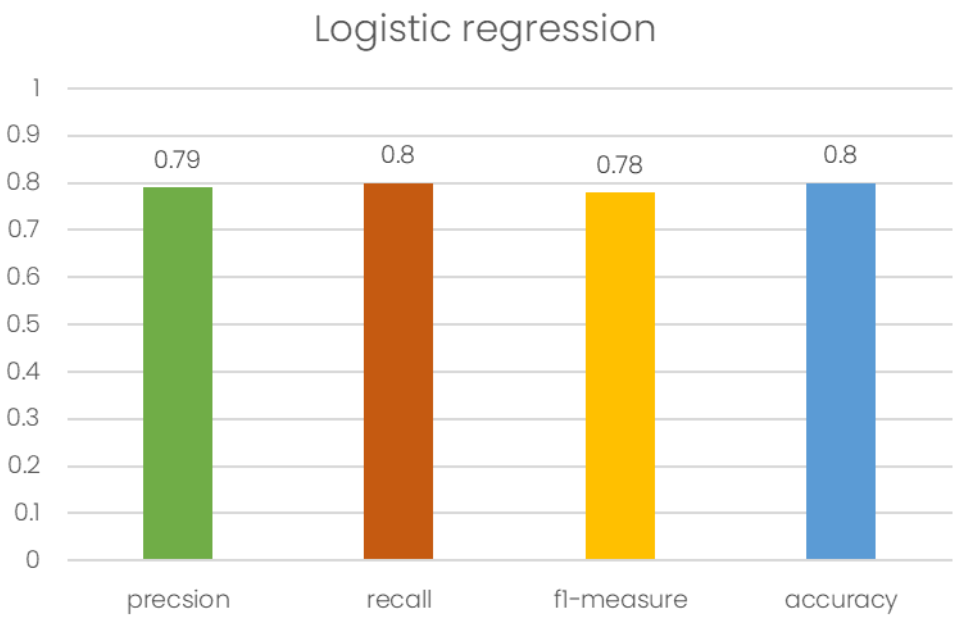
Sviluppi Futuri

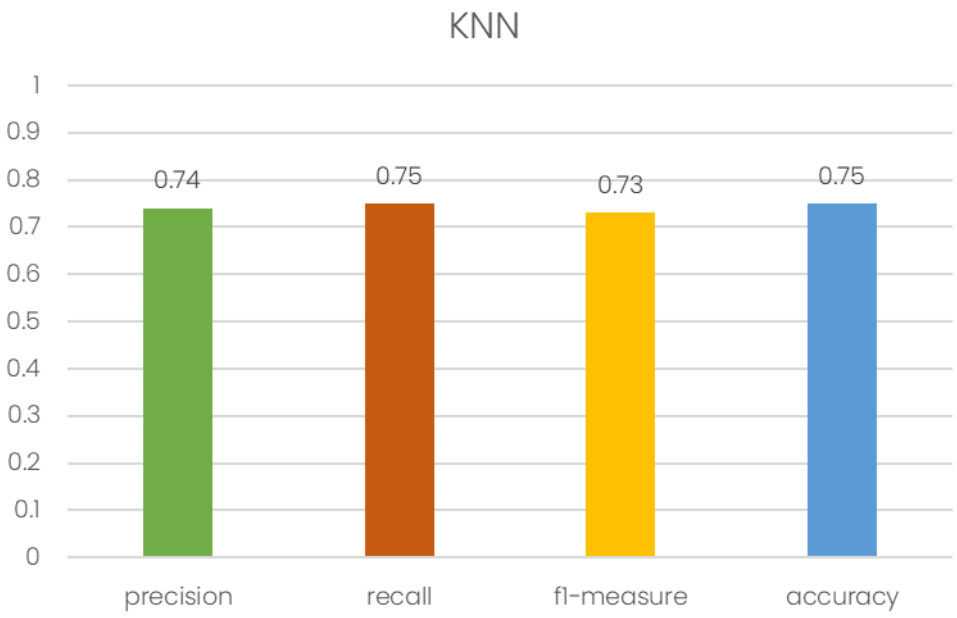
1. Esplorazione di modelli avanzati (Gradient Boosting, Reti Neurali più complesse).
2. Ampliare il dataset e comparare le prestazioni del modello sui nuovi dati.
3. Tecniche di feature selection e analisi dati più avanzata.
4. Sviluppo di un'applicazione web per la classificazione automatica.

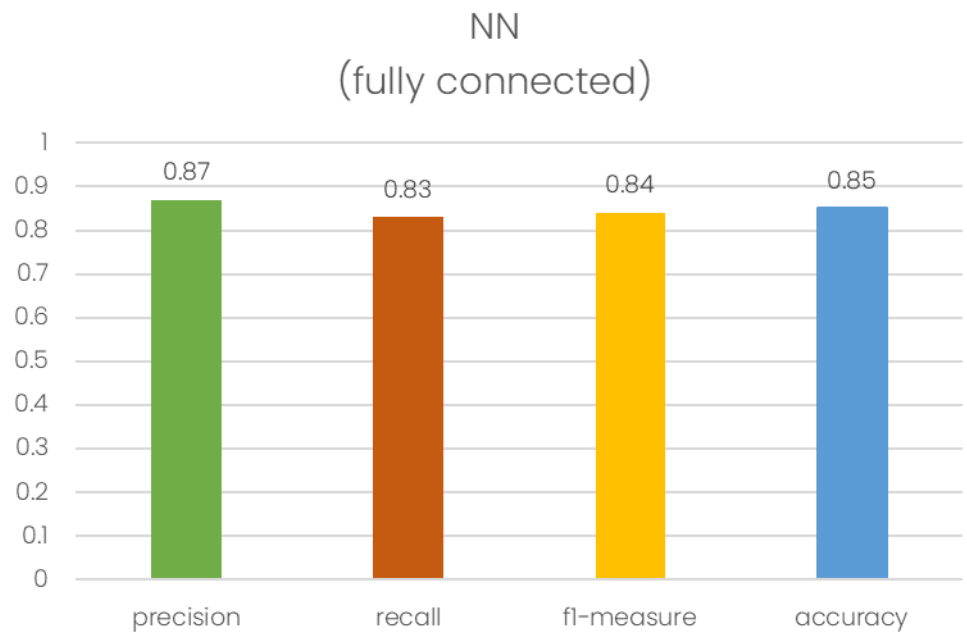
Dipendenze del Progetto

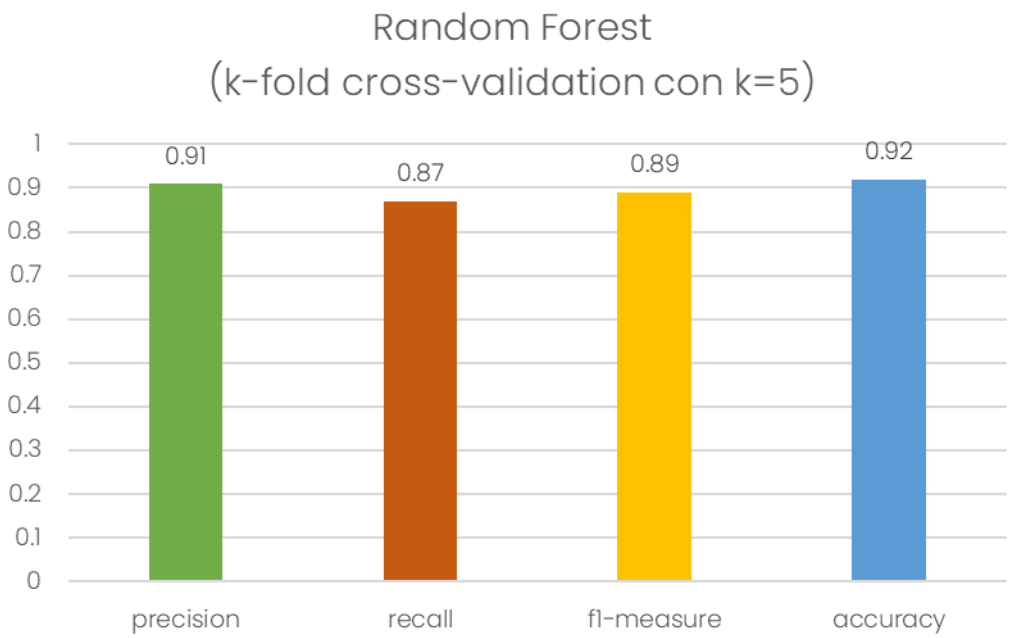
Pandas, Numpy, Scikit-learn, Matplotlib, Joblib, Keras, Tensorflow, Seaborn

Risultati ottenuti









Necessari alla realizzazione del seguente progetto sono stati i paper:

- "**Variable Star Signature Classification using Slotted Symbolic Markov Modeling**"
- **"Classification of variable stars"**

rispettivamente lavoro di *K.B. Johnston, A.M. Peter* (Florida Institute of Technology) e *Tirth Surti, Abhijit Devalapura* e *Christina Sze* (Stanford University).

Dataset: https://faculty.washington.edu/ivezic/linear/PaperIII/PLV.html