



ITIS-LS “Francesco Giordani” Caserta

prof. Ennio Ranucci

a.s. 2023-2024

Colab, Python, Chatgpt e AI



Colab, o "Colaboratory", ti permette di scrivere ed eseguire Python nel tuo browser con

- Nessuna configurazione necessaria
- Accesso alle GPU senza costi
- Condivisione semplificata

Accedere al proprio Account Google, aprire una nuova scheda e scrivere Colab nella barra degli indirizzi.

Aprire il sito Google Colab

Cliccare su Annulla nella finestra "Apri blocco note" e sul segno "+" che precede "codice"

Copiare il seguente codice python della riga della finestra che si apre:

```
def contatoreCaratteri(listaA):  
    listaB = []  
    for parola in listaA:  
        listaB.append(len(parola))  
    return listaB
```

```
contatoreCaratteri(['Ennio','Salvatore','Lorenzo'])
```

poi cliccare sul play per eseguire il codice incollato.

Ora utilizziamo chatgpt per scrivere codice python relativo all'IA

ChatGPT 3.5 ▾



You

scrivi codice python per una esercitazione per studenti relativa all'IA, usa dataset italiano



ChatGPT

Certamente! Ecco un esempio di esercitazione in Python che coinvolge un semplice problema di classificazione utilizzando un dataset italiano. In questo esempio, useremo il famoso dataset Iris per la classificazione dei fiori.

```
python
```

Copy code

```
# Importa le librerie necessarie
```

```
import pandas as pd
```

```
from sklearn.model_selection import train_test_split
```

```
from sklearn.ensemble import RandomForestClassifier
```

```
from sklearn.metrics import accuracy_score, classification_report
```

```
# Carica il dataset Iris (puoi sostituire con un dataset italiano)

iris_url = "https://archive.ics.uci.edu/ml/machine-learning-databases/iris/iris.data"
column_names = ['sepal_length', 'sepal_width', 'petal_length', 'petal_width', 'class']
iris_data = pd.read_csv(iris_url, header=None, names=column_names)

# Visualizza le prime righe del dataset
print("Primi 5 record del dataset Iris:")
print(iris_data.head())

# Suddividi il dataset in features (X) e target (y)
X = iris_data.drop('class', axis=1)
y = iris_data['class']

# Suddividi il dataset in training set e test set
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)

# Crea un modello di classificazione (usiamo RandomForest come esempio)
model = RandomForestClassifier(n_estimators=100, random_state=42)

# Addestra il modello sul training set
model.fit(X_train, y_train)

# Effettua predizioni sul test set
predictions = model.predict(X_test)

# Valuta le prestazioni del modello
accuracy = accuracy_score(y_test, predictions)
report = classification_report(y_test, predictions)
```

```
# Visualizza i risultati

print("\nAccuracy del modello:", accuracy)

print("\nReport di classificazione:")

print(report)
```

Ricorda che questo è solo un esempio generale. Puoi sostituire il dataset Iris con un dataset italiano più specifico per il tuo scopo. Assicurati che il dataset abbia etichette (target) e features che possano essere utilizzate per l'addestramento di un modello di machine learning.

ChatGPT 3.5 ▾



You

spiega dettagliatamente l'obiettivo, il codice e il risultato



ChatGPT

Obiettivo:

L'obiettivo di questo esempio è introdurre gli studenti al concetto di apprendimento automatico (machine learning) attraverso un problema di classificazione. Verrà utilizzato un dataset italiano (anche se, nell'esempio, viene usato il dataset Iris per la sua disponibilità e facilità di utilizzo).

Codice:

1. **Importazione delle librerie:** Vengono importate le librerie necessarie per manipolare i dati (pandas), suddividere il dataset in set di training e test (train_test_split), costruire un modello di classificazione (RandomForestClassifier) e valutare le prestazioni del modello (accuracy_score, classification_report).
2. **Caricamento del dataset:** Viene caricato il dataset Iris da un URL usando pandas. Il dataset contiene informazioni su quattro caratteristiche di tre specie di fiori Iris (setosa, versicolor, virginica). Puoi sostituire questo dataset con uno italiano che sia adatto al tuo scopo.
3. **Visualizzazione dei dati:** Vengono visualizzate le prime 5 righe del dataset per mostrare come sono strutturati i dati.
4. **Preparazione dei dati:** Il dataset viene suddiviso in features (X) e target (y). Le features sono le colonne che saranno utilizzate per fare predizioni, mentre il target è la variabile che si cerca di predire.
5. **Suddivisione in set di training e test:** Il dataset viene suddiviso in un set di training (usato per addestrare il modello) e un set di test (usato per valutare le prestazioni del modello) utilizzando la funzione **train_test_split**.
6. **Creazione e addestramento del modello:** Viene creato un modello di classificazione utilizzando un algoritmo di Random Forest con 100 alberi (**RandomForestClassifier**). Il modello viene poi addestrato sul set di training.

7. **Predizioni e valutazione:** Il modello addestrato viene utilizzato per fare predizioni sul set di test. Successivamente, vengono valutate le prestazioni del modello calcolando l'accuracy (percentuale di predizioni corrette) e generando un report di classificazione che include metriche come precision, recall e f1-score per ciascuna classe.

Risultato: Il risultato sarà l'accuracy del modello e un report di classificazione che fornisce una valutazione dettagliata delle prestazioni del modello su ciascuna classe nel set di test. Gli studenti possono esplorare il concetto di overfitting, tuning degli iperparametri e altre pratiche comuni nell'apprendimento automatico. L'obiettivo è far capire loro come costruire, addestrare e valutare un modello di machine learning per risolvere un problema di classificazione.