

Project #01

Binary classification on custom DataSet

Project #01

Indice

1. Studio del corso di introduzione all'IA con Python
2. Setup ambiente di sviluppo
3. Progetto di classificazione di immagini

1. Studio

Alcuni argomenti del corso “Introduzione all'IA con Python”

- Configurazione dell'ambiente di lavoro
- Tipi ed esempi di machine learning
- Librerie Python
- Reti neurali di regressione
- Convoluzioni e filtri
- Reti neurali convoluzionali
- Trasferimento di apprendimento

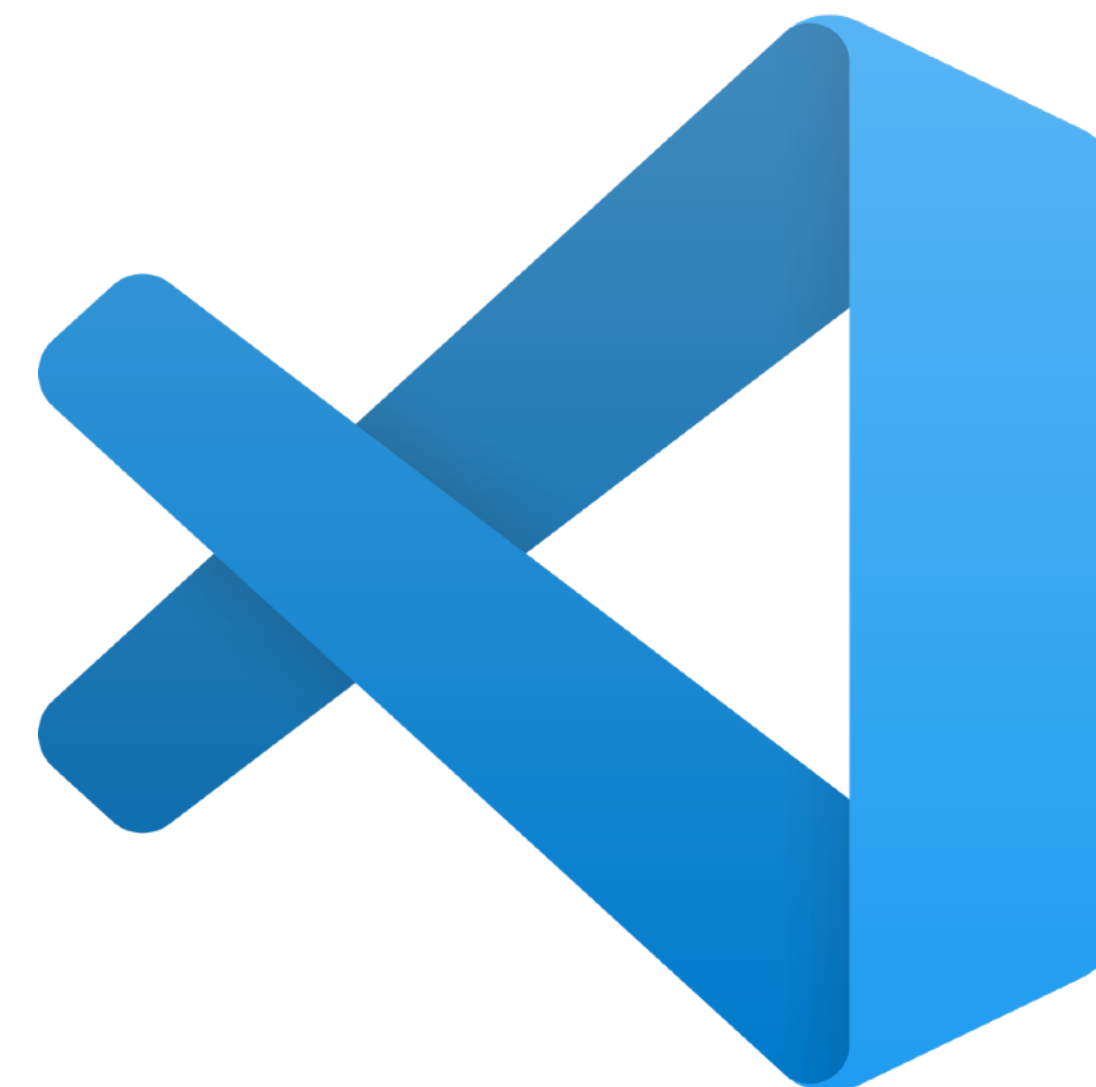
2. Setup ambiente di sviluppo

Librerie e pacchetti installati:

- Anaconda Navigator (2.4.2)
- Python (3.11.5)
- Miniconda 3 (per Conda 23.5.2)
- PyTorch (2.0.1)
- Pandas (2.1.0)
- Scikit-learn (1.3.0)
- TensorFlow (2.13.0)
- TensorFlow Hub (0.14.0)
- OpenCV for python (4.8.0.76)

IDE:

- Visual Studio Code
 - ➡ **Jupyter Notebook**
(IPython notebook)



3. Progetto di classificazione di immagini

3.1 Generalità

- Progetto di intelligenza artificiale di un Images Classifier con Tensorflow, che possa classificare immagini del felino giaguaro e del roditore idrochèro.
- Per realizzare questo progetto è stato seguito un corso di *Rodrigo Montemayor* intitolato "Introduzione all'IA con Python" e pubblicato sulla piattaforma Domestika.

3. Progetto di classificazione di immagini

3.2 Creazione del dataset

- È stato creato un dataset di immagini dei due animali: giaguaro (felino) e idrochèro (roditore).
- Le immagini che compongono il dataset sono state raccolte da internet e salvate in due archivi .zip da 481 immagini l'uno.

3. Progetto di classificazione di immagini

3.3 Utilizzo di Image Data Generator

- Viene importato Image Data Generator di TensorFlow.
- Questo modulo semplifica notevolmente la gestione delle immagini, incluso il *data augmentation** per evitare il sovradattamento e la preparazione dei dati per il modello.

* È un insieme di tecniche che ampliano il dataset a disposizione senza effettivamente raccogliere nuovi elementi: la *data augmentation* applica ai dati già esistenti dei cambiamenti casuali controllati, realizzandone delle copie modificate.

3. Progetto di classificazione di immagini

3.4 Modello preaddestrato

- È stato usato un modello preaddestrato in formato “Saved Model”
- Il modello è basato su MobileNet V2 ed è stato importato nel progetto grazie a TensorFlow Hub.
- I modelli preaddestrati hanno già imparato a riconoscere molte caratteristiche delle immagini e possono essere sfruttati come base solida per il compito di classificazione.

3. Progetto di classificazione di immagini

3.5 Creazione del modello personalizzato

- Viene aggiunto un ultimo strato "Dense Layer" per la classificazione finale. Ciò è una pratica comune.
- Questo strato aggiunto al modello preaddestrato consentirà di adattare il modello alle specifiche categorie di animali (giaguaro e idrochèro).

3. Progetto di classificazione di immagini

3.6 Predizioni e valutazione del modello

- Fase di predizione su immagini di giaguari e idrocheri per valutare le prestazioni del modello.
- Il modello riesce bene a distinguere tra le due categorie.

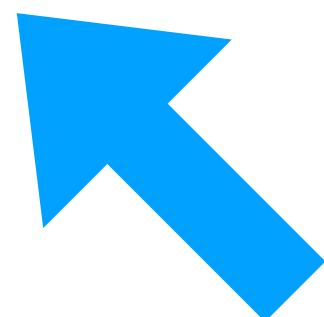
```
root = '/Users/vittoriostile/Developer/Python/content/idrochero.jpeg'

prediction = to_classify(root)
print(prediction)
```

✓ 0.0s Python

1/1 [=====] - 0s 37ms/step

1



3. Progetto di classificazione di immagini

3.7 Predizioni e valutazione del modello

- Fase di predizione su immagini di giaguari e idrocheri per valutare le prestazioni del modello.
- Il modello riesce bene a distinguere tra le due categorie.

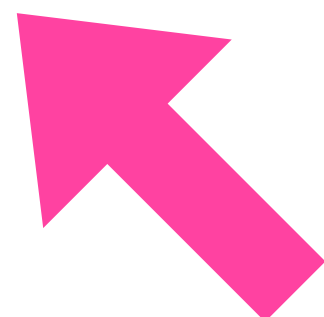
```
root = '/Users/vittoriostile/Developer/Python/content/giaguaro.jpeg'

prediction = to_classify(root)
print(prediction)
```

✓ 0.4s Python

1/1 [=====] - 0s 352ms/step

0



Grazie !!!

VITTORIO **STILE**