



Analisi dei dati e elementi di Machine Learning con Python

CODICE

NOVANMLPY

DURATA

3 Giorni

PREZZO

1.800,00€ (iva escl.)

LINGUA

Italiano

MODALITÀ

Virtual Classroom
Corso in aula

SCHEDULAZIONE

- A Richiesta

Il corso si rivolge a persone che hanno conoscenze di Python 3 e un minimo di esperienze con la libreria Numpy per il calcolo numerico efficiente.

È inoltre richiesta conoscenza di matematica a livello universitario.

Il corso ha un forte carattere pratico e hands-on.

Prerequisito tecnico per i partecipanti è di aver installato sul proprio computer un ambiente Python 3 con un editor e un sistema per l'installazione di pacchetti di terze parti.

Il sistema proposto è:

- IDE PyCharm - basta la versione "Community" gratuita, vedere:

<https://www.jetbrains.com/pycharm/download>

- Ambiente Anaconda (o similmente Miniconda o Miniforge) come distribuzione di Python e gestione pacchetti

Metodologia didattica:

Vengono alternanti momenti di spiegazione teorica con sessioni di live-coding e sessioni di esercizi da svolgere individualmente.

PREREQUISITI

- Conoscenza di base, anche minima di SQL (qualsiasi database relazionale).

CONTENUTI

[Introduzione Panoramica sulle varie tecnologie di apprendimento supervisionato e non: problemi di regressione, classificazione e clusterizzazione.

Blocco 1

[Teoria] Problemi di regressione e gli algoritmi più popolari (modelli lineari, decision tree).

[Live Coding] Creare un modello regressivo per predire il prezzo di una casa con Scikit-Learn: definire il problema; ottenere i dati; importare e pulire i dati; plottare i dati per una prima analisi; dividere i dati in set per l'allenamento e la validazione; creare modelli regressivi con diversi algoritmi disponibile in Scikit-Learn e confrontarne la performance.



[Esercizio/Lab] Creare un modello regressivo per predire l'efficienza di consumo di un'automobile.

Blocco 2

[Teoria] La libreria Pandas per gestire dati in formato tabellare. Gestione di valori NaN; selezionare, filtrare e aggregare i dati; gestione di serie temporali usando interpolazione e indicizzazione; gestione di dati categorici.

[Live Coding] Leggere serie temporali di dati meteorologici (temperatura, precipitazione) di alcune città e calcolare indicatori climatici come la temperatura media minima e massima per mese e plottare il risultato.

[Esercizio/Lab] Ottenere, interpolare e plottare una serie temporale di dati riguardante un parcheggio pubblico.

Blocco 3

[Teoria] Problemi di classificazione e introduzione alle reti neurali.

[Live Coding] Classificazione di fotografie di segnaletica stradale con una rete semplice (fully connected, un unico strato nascosto) usando la libreria PyTorch.

[Esercizio/Lab] Classificazione dei caratteri "0" - "9" scritti a mano del dataset MNIST con una rete semplice e PyTorch.

Blocco 4

[Teoria] Cenni a reti neurali convoluzionali e deep learning.

[Live Coding] Migliorare la performance della rete precedente per la classificazione della segnaletica aggiungendo strati convoluzionali (architettura LeNet).

[Esercizio/Lab] I partecipanti possono applicare la stessa modifica alla loro rete per il riconoscimento dei caratteri.

Blocco 5

[Teoria] Problemi di clusterizzazione e gli algoritmi più popolari (k-means, DBSCAN).

[Live Coding] Un esempio di customer segmentation su dataset di kaggle.com usando Scikit-Learn e l'algoritmo DBSCAN.

[Esercizio/Lab] Ripetere l'esercizio con un altro dataset di kaggle.com a scelta del partecipante.

Prezzi e corsi potrebbero subire variazioni; si consiglia di verificare sul sito www.novanext.it/training.