

## **Introduzione al Machine Learning - Hybrid Cohort-based Training**

Questo corso copre le basi dell'apprendimento automatico, esplorando vari tipi di modelli, tra cui regressione lineare, logistica, e reti neurali. Si approfondiscono concetti come l'overfitting e l'ottimizzazione, utilizzando tecniche avanzate come il gradient descent, la regolarizzazione, e il feature scaling. Gli studenti apprenderanno anche la classificazione multiclasse, la gestione del bias e della varianza, e l'uso del transfer learning. Verranno introdotte le reti neurali convoluzionali (CNN) per la visione artificiale e le reti ricorrenti (RNN) per le serie temporali, inclusi metodi come la Data Augmentation e il Long Short-Term Memory (LSTM). I moduli prevedono sessioni pratiche e laboratori per l'implementazione di modelli in Python e TensorFlow.

### **Profilo dei destinatari**

In generale, il corso è rivolto a chiunque abbia una conoscenza di base degli strumenti informatici e desideri applicare tecniche di Machine Learning e Deep Learning per analizzare dati e prendere decisioni basate su modelli predittivi.

### **Obiettivi del corso**

Dopo aver completato questo corso, i partecipanti saranno in grado di:

- Acquisire e pre-elaborare dati per l'analisi e lo sviluppo di modelli di Machine Learning.
- Comprendere e implementare tecniche di Machine Learning, incluse regressione lineare, logistica e reti neurali.
- Gestire problemi di overfitting e migliorare i modelli tramite regolarizzazione e ottimizzazione avanzata.
- Utilizzare librerie Python (Numpy, Pandas, Matplotlib, TensorFlow) per sviluppare, addestrare e valutare modelli di ML.
- Applicare tecniche di Data Augmentation e Transfer Learning per migliorare l'efficacia dei modelli e affrontare dataset complessi.

### **Prerequisiti**

Esperienza di base sull'uso di strumenti informatici. Conoscenza di base del linguaggio Python. Concetti fondamentali di algebra lineare e statistica (utili per comprendere i modelli di Machine Learning).

## **Contenuti del corso**

### Modulo 1: Introduzione al Machine Learning

Questo modulo esplora le basi del Machine Learning, i diversi tipi di apprendimento automatico (supervisionato, non supervisionato, semi-supervisionato), il concetto di overfitting e underfitting, e l'importanza dei dati. Gli studenti acquisiranno familiarità con le principali librerie Python che verranno usate nei moduli successivi (pandas, numpy, matplotlib).

Sessione dal vivo (4 ore):

- Presentazione
  - Introduzione al Machine Learning: la nuova "elettricità";
  - Un nuovo paradigma incentrato sui dati: pro e contro;
  - Quando usare il Machine Learning;
  - Esempi di applicazioni di ML;
  - Tipi di Machine Learning:
  - Supervised Learning (Regressione Lineare, Regressione Logistica);
  - Unsupervised Learning (Clustering, Rilevamento di anomalie, Riduzione dimensionale);
  - Semi-supervised Learning;
  - Batch vs Online Learning, ML basato su istanze vs ML basato su modelli.
- Laboratorio: Introduzione agli strumenti di Machine Learning
  - Uso di Jupyter Notebook per il Machine Learning;
  - Esempi pratici con le librerie Python (Numpy, Pandas, Matplotlib)
  - Quiz: valutazione dei concetti principali.

Obiettivi:

- Comprendere i concetti base del Machine Learning e il suo impatto;
- Identificare i diversi tipi di Machine Learning e quando applicarli;
- Utilizzare gli strumenti principali per creare modelli di ML in Python;
- Utilizzare l'interfaccia di Jupyter Notebook per analisi e sviluppo di modelli di ML.

## Modulo 2: Regressione Lineare e Logistica, Overfitting e Regolarizzazione

Questo modulo esplora la regressione lineare e logistica, con un focus sull'uso delle funzioni di costo, il gradient descent, la regolarizzazione, e la gestione dell'overfitting. Gli studenti apprenderanno concetti chiave come la curva di apprendimento, la vettorizzazione e il ruolo del feature scaling. Verranno trattati esempi pratici per visualizzare e comprendere meglio il comportamento dei modelli.

Sessione dal vivo (6 ore):

- Presentazione
  - Regressione Lineare:
    - Funzione di costo (Cost function);
    - Visualizzazione della Cost Function e degli esempi;
    - Gradient descent e Learning rate;
    - Feature Scaling e Regressione Polinomiale;
    - Curve di apprendimento e Vettorizzazione.
  - Regressione Logistica (Classificazione):
    - Classificazione e Logistic Loss;
    - Funzione di costo per la Regressione Logistica;
    - Funzione 'sigmoid' e Decision Boundary;
    - Gradient Descent per Logistic Regression.
  - Overfitting:
  - Funzione di costo con regolarizzazione;
  - Regressione Lineare con regolarizzazione;
  - Regressione Logistica con regolarizzazione.
- Laboratorio:
  - Laboratorio sulla Regressione Lineare;
    - Implementazione della regressione lineare con Python e Scikit-Learn;
    - Esempi di visualizzazione della Cost Function e ottimizzazione con Gradient Descent;
    - Applicazione del Feature Scaling e Regressione Polinomiale;
    - Creazione di curve di apprendimento per valutare l'andamento del modello.
    - Quiz
  - Laboratorio sulla Regressione Logistica:
    - Implementazione della Regressione Logistica per la classificazione binaria;
    - Utilizzo della funzione 'sigmoid' e ottimizzazione con Gradient Descent;
    - Definizione e analisi del Decision Boundary.
    - Quiz
  - Laboratorio su Overfitting e Regolarizzazione:
    - Applicazione della regolarizzazione alla Regressione Lineare;
    - Utilizzo della regolarizzazione nella Regressione Logistica;
    - Confronto tra modelli regolarizzati e non, valutazione dei risultati.
    - Quiz

#### Sessione in autoapprendimento:

- Approfondimento sulla Regressione Lineare: condizioni da soddisfare per l'applicazione delle Regressione Lineare
- Esempio di modello basato sulla Regressione Logistica preso da Kaggle.
- Generazione di dati sintetici in Python

#### Attività individuale:

- Definizione e implementazione di un modello di regressione lineare o logistica per un dataset scelto dallo studente.

#### Attività di gruppo:

- Presentazione dei progetti individuali;

#### Obiettivi:

- Comprendere i concetti chiave della regressione lineare e logistica;
- Implementare i modelli di regressione utilizzando Python e Scikit-Learn;
- Identificare l'overfitting e applicare la regolarizzazione per migliorare i modelli;
- Ottimizzare i modelli attraverso feature scaling e regolarizzazione;

### Modulo 3: Ottimizzazione Avanzata, Classificazione Multiclasse, Bias e Varianza

Questo modulo esplora l'ottimizzazione avanzata, la scelta delle funzioni di attivazione, la classificazione multiclasse e la gestione del bias e della varianza. Gli studenti apprenderanno tecniche avanzate per migliorare i modelli di Machine Learning e verranno introdotti al ciclo completo di sviluppo di un progetto ML, incluso il Transfer Learning e l'analisi degli errori.

Sessione dal vivo (6 ore):

- Presentazione:
  - Scelta della Funzione di Attivazione:
    - Esempi pratici, ReLu, LeakyReLu, Tanh;
  - Classificazione Multiclasse:
    - Esempi e implementazione della classificazione multiclasse;
    - Utilizzo dell'output Softmax;
  - Ottimizzazione Avanzata:
    - Stochastic Gradient Descent, RMSprop, Adam optimizers;
  - Strategie per eseguire una verifica di un modello di ML:
    - Training, Cross Validation, Test sets;
    - Esempi
  - Bias e Varianza:
    - Identificazione di problemi di Bias o Varianza;
    - Regolarizzazione e stabilire una 'Baseline of performance';
    - Curva di apprendimento (Learning Curves) e loro analisi;
    - Bias e Varianza nelle reti neurali;
- Laboratorio:
  - Sviluppo ed ottimizzazione di un modello di Machine Learning Multiclasse,
  - Analisi degli errori e miglioramento del modello, Verifica del modello su Training, Cross Validation e Test sets;
  - Identificazione di problemi di bias e varianza, e correzione mediante regolarizzazione e ottimizzazione.
  - Laboratorio su Data Augmentation e Transfer Learning
  - Quiz

Sessione in autoapprendimento:

- Ciclo completo di un progetto di ML: Implementazione di un progetto ML completo, dall'acquisizione dei dati all'analisi degli errori e al miglioramento continuo del modello.

Attività individuale:

- Generazione di dati sintetici; fitting di un modello a scelta dello studente,
- Identificazione di eventuali problemi di underfitting e overfitting, applicazione di tecniche di riduzione dell'overfitting

Attività di gruppo:

- Presentazione e discussione dei progetti individuali;

Obiettivi:

- Comprendere le tecniche avanzate di ottimizzazione, inclusi Gradient Descent, RMSprop e Adam;
- Implementare modelli di classificazione multiclasse utilizzando Softmax;
- Identificare problemi di bias e varianza e applicare soluzioni efficaci;
- Migliorare i modelli con tecniche di Data Augmentation e Transfer Learning;
- Realizzare un ciclo completo di sviluppo di un progetto di Machine Learning.

## Modulo 4: Introduzione alle Reti Neurali

Questo modulo esplora le reti neurali, concentrandosi sulla loro struttura, il processo di addestramento e l'implementazione pratica tramite Tensorflow. Saranno presentate le tecniche per la costruzione di reti neurali artificiali, l'ottimizzazione dei modelli, e l'applicazione delle reti neurali a problemi di classificazione.

Sessione dal vivo (6 ore):

- **Presentazione:**
  - Introduzione alle Reti Neurali:
    - Definizione di rete neurale;
    - Struttura base: neuroni, strati e connessioni;
    - Il concetto di apprendimento automatico nelle reti neurali;
    - Applicazioni pratiche delle reti neurali.
  - Perceptron e Reti Neurali Artificiali (ANN):
    - Il neurone artificiale (Perceptron);
    - Funzione di attivazione (ReLU, Sigmoid, Softmax);
    - Architettura di una rete neurale feedforward;
    - Propagazione in avanti (Forward Propagation).
  - Addestramento delle Reti Neurali:
    - Funzione di perdita (Loss function);
    - Ottimizzazione tramite discesa del gradiente (Gradient Descent);
    - Propagazione all'indietro (Backpropagation);
    - Overfitting e tecniche di regolarizzazione (dropout, weight regularization);
    - Divisione in training, validation e test set.
  - Introduzione a Tensorflow:
    - Cos'è Tensorflow: un framework per il Deep Learning;
    - Installazione e configurazione di Tensorflow;
    - Creazione di un modello base con Tensorflow: definizione dei layers e compilazione del modello.
- **Laboratorio:**
  - **Costruzione di una Rete Neurale con Tensorflow:**
    - Definizione dei layers con Keras API;
    - Selezione delle funzioni di attivazione e della funzione di perdita;
    - Compilazione del modello: ottimizzatore (Adam, SGD) e metriche;
    - Addestramento del modello su un dataset con `model.fit()`;
    - Visualizzazione della curva di apprendimento.
    - Quiz
  - **Valutazione e Ottimizzazione del Modello:**
    - Valutazione del modello su dati di test con `model.evaluate()`;
    - Miglioramento delle prestazioni con tecniche di tuning (learning rate, batch size);
    - Introduzione al concetto di Transfer Learning per reti pre-addestrate.
    - Quiz
  - **Laboratorio Pratico:**
    - Utilizzo del dataset MNIST per il riconoscimento delle cifre;

- Costruzione, addestramento e valutazione di un modello di rete neurale con Tensorflow;
- Analisi dei risultati e miglioramento del modello.

Sessione in autoapprendimento:

- Approfondimento su Tensorflow e tecniche di ottimizzazione delle reti neurali.

Attività individuale:

- Definizione e implementazione di un progetto individuale per la classificazione di immagini o dati;
- Generazione o acquisizione dei dati necessari al progetto.

Attività di gruppo:

- Presentazione dei progetti individuali;

Obiettivi:

- Comprendere la struttura e il funzionamento delle reti neurali;
- Implementare una rete neurale utilizzando Tensorflow e Keras;
- Ottimizzare un modello di rete neurale tramite tecniche avanzate;
- Valutare e migliorare i modelli attraverso esperimenti pratici.

## Modulo 5: Reti Neurali Avanzate e Analisi di Time Series

Questo modulo esplora l'uso delle Reti Neurali Convoluzionali (CNN) e le Reti Neurali Ricorrenti (RNN) applicate a problemi di Computer Vision e Time Series. Gli studenti impareranno a identificare e pre-processare dati da varie fonti, utilizzando tecniche avanzate come Data Augmentation e Transfer Learning. Saranno presentati metodi per migliorare le prestazioni delle reti e realizzare previsioni su dati sequenziali.

Sessione dal vivo (6 ore):

- Presentazione:
  - Introduzione alle Reti Neurali Convoluzionali (CNN);
  - Introduzione alla Computer Vision e utilizzo del dataset MNIST;
  - Pooling Layers nelle CNN e loro importanza per migliorare le prestazioni;
  - Introduzione a Transfer Learning e Data Augmentation per migliorare le performance;
  - Reti Neurali Ricorrenti (RNN) e Long Short-Term Memory (LSTM) per l'analisi di Time Series;
  - Utilizzo di Moving Averages e finestre temporali per il Forecasting.
- Laboratorio:
  - Creazione di una CNN per il riconoscimento di caratteri scritti a mano;
  - Utilizzo di ImageDataGenerator per il Training e la Data Augmentation;
  - Implementazione di una RNN per la previsione su Time Series, utilizzando TensorFlow;
  - Uso del Dropout e del Transfer Learning con una rete pre-addestrata (es. Inception);
  - Visualizzazione dell'effetto della convoluzione e miglioramento della precisione con dataset complessi.

Sessione in autoapprendimento:

- Fondamenti di Machine Learning per Time Series;
- Previsione con modelli di Machine Learning su Time Series;
- Esempi pratici di predizione su sequenze temporali con reti neurali presi da Kaggle

Attività individuale:

- Definizione del project work individuale;
- Generazione o acquisizione di un dataset per l'addestramento di un modello CNN o RNN;
- Implementazione di una CNN per la classificazione di immagini o di una RNN per la previsione su Time Series.

Attività di gruppo:

- Presentazione dei progetti individuali;
- Definizione degli ambiti di integrazione e collaborazione tra i partecipanti.

Obiettivi:

- Identificare e pre-processare i dati per l'addestramento di reti neurali complesse;
- Utilizzare CNN e RNN per problemi di Computer Vision e Time Series;
- Migliorare le performance di un modello con tecniche come Data Augmentation e Transfer Learning;
- Valutare la qualità dei modelli tramite test sets e validazione;
- Applicare tecniche avanzate di predizione su sequenze temporali utilizzando LSTM.