D.E.M.E.TRA.

PSR Campania 2014-2020 - Sottomisura 16.1 Azione 2

Caserta – 15 Settembre 2020

REPORT M3

elenco e tipologia di sistemi di machine learning e valutazione delle tipologie più adeguate per il progetto

---

PSR Campania 2014-2020 - Sottomisura 16.1 Azione 2 - Focus Area 3A

CUP: B28H19005170006

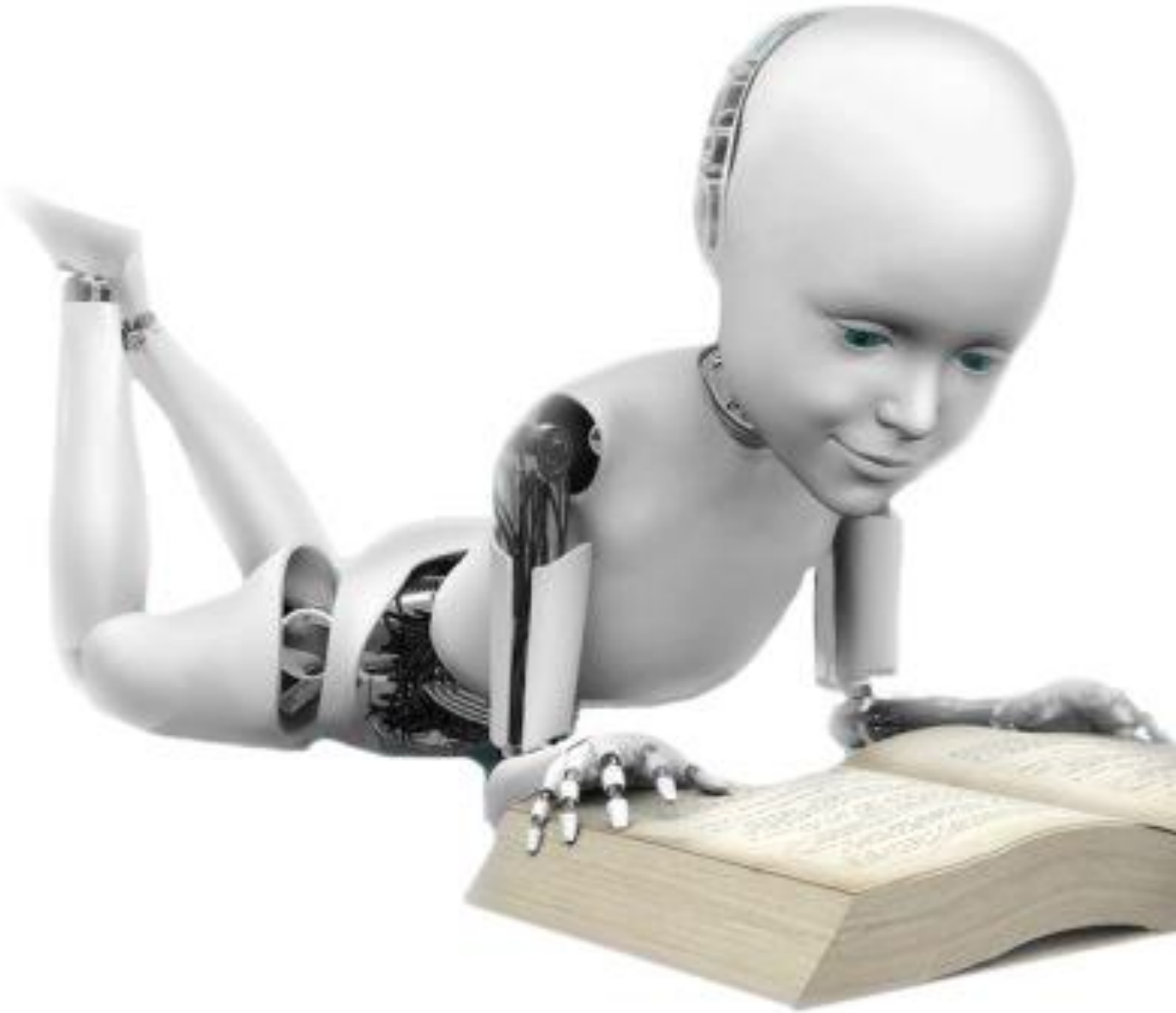
Fondo europeo agricolo per lo sviluppo rurale: *l'Europa investe nelle zone rurali*

Unione Europea

REGIONE CAMPANIA

Il presente elaborato è la delivery del task#3: "Mappatura sistemi di machine learning" del WP.03: "valutazione componenti tecnologici per sistemi di monitoraggio e certificazione produzione nocciole" del progetto D.E.M.E.TRA. e nel rispetto del cronoprogramma tale attività è iniziata il 01 Agosto 2020 ed è stata completata il 31 Agosto 2020.

Questa specifica attività del progetto D.E.M.E.TRA. punta a svolgere uno studio finalizzato a mappare le tipologie di algoritmi di machine learning attualmente disponibili e utilizzabili in capo agricolo che risultano compatibili con le finalità e gli obiettivi del progetto. Lo studio è rivolto principalmente ad esaminare gli strumenti già attualmente in uso e che possono essere facilmente integrati nel progetto.



Come funziona il Machine Learning	l'apprendimento automatico si distingue in due macro categorie: apprendimento supervisionato: quando si danno al computer esempi completi da utilizzare come indicazione per eseguire il compito richiesto; apprendimento non supervisionato: quando si lascia lavorare il software senza alcun "aiuto".
Apprendimento Supervisionato	Al computer vengono "dati in pasto" sia dei set di dati come input sia le informazioni relative ai risultati desiderati con l'obiettivo che il sistema identifichi una regola generale che colleghi i dati in ingresso con quelli in uscita (gli vengono cioè forniti degli esempi di input e di output in modo che impari il nesso tra loro), in modo da poter poi riutilizzare tale regola/modello per altri compiti simili. «Nell'apprendimento supervisionato il lavoro di risoluzione viene lasciato al computer. Una volta compresa la funzione matematica che ha portato a risolvere uno specifico insieme di problemi, sarà possibile riutilizzare la funzione per rispondere a qualsiasi altro problema similare», Adam Geitgey.
Apprendimento NON Supervisionato	In questa seconda categoria di Machine Learning al sistema vengono forniti solo set di dati senza alcuna indicazione del risultato desiderato. Lo scopo di questo secondo metodo di apprendimento è "risalire" a schemi e modelli nascosti, ossia identificare negli input una struttura logica senza che questi siano preventivamente etichettati.
Apprendimento Semi-Supervisionato	In questo caso si tratta di un modello "ibrido" dove al computer viene fornito un set di dati incompleti per l'allenamento/apprendimento; alcuni di questi input sono "dotati" dei rispettivi esempi di output (come nell'apprendimento supervisionato), altri invece ne sono privi (come nell'apprendimento non supervisionato). L'obiettivo, di fondo, è sempre lo stesso: identificare regole e funzioni per la risoluzione dei problemi, nonché modelli e strutture di dati utili a raggiungere determinati obiettivi.
Apprendimento per Rinforzo	In questo caso, il sistema (computer, software, algoritmo) deve interagire con un ambiente dinamico (che gli consente di avere i dati di input) e raggiungere un obiettivo (al raggiungimento del quale riceve una ricompensa), imparando anche dagli errori (identificati mediante "punizioni"). Il comportamento (e le prestazioni) del sistema è determinato da una routine di apprendimento basata su ricompensa e punizione. Con un modello del genere, il computer impara per esempio a battere un avversario in un gioco (o a guidare un veicolo) concentrando gli sforzi sullo svolgimento di un determinato compito mirando a raggiungere il massimo valore della ricompensa; in altre parole, il sistema impara giocando (o guidando) e dagli errori commessi migliorando le prestazioni proprio in funzione dei risultati raggiunti in precedenza.

Ci sono poi altre sottocategorie di Machine Learning che in realtà servono a darne una sorta di classificazione "pratica" perché, di fatto, identificano degli approcci pratici di applicazione degli algoritmi di Machine Learning (da cui si possono quindi derivare delle categorie di "apprendimento" dei sistemi).

Parliamo per esempio dei cosiddetti "alberi delle decisioni" basati su grafi attraverso i quali si sviluppano modelli predittivi grazie ai quali è possibile scoprire le conseguenze (output) di determinate decisioni (input).

Altro esempio concreto viene dal "clustering" ossia dai modelli matematici che consentono di raggruppare dati, informazioni, oggetti, ecc. "simili"; si tratta di una applicazione pratica del Machine Learning dietro al quale esistono modelli di apprendimento differenti che vanno dall'identificazione delle strutture (cosa definisce un cluster e qual è la sua natura) al riconoscimento degli "oggetti" che devono far parte di un gruppo piuttosto che di un altro.

Dai modelli probabilistici al Deep Learning

C'è poi la sottocategoria dei "modelli probabilistici" che basano il processo di apprendimento del sistema sul calcolo delle probabilità (il più noto è forse la "rete di Bayes", un modello probabilistico che rappresenta in un grafo l'insieme delle variabili casuali e le relative dipendenze condizionali).

Infine, ci sono le notissime reti neurali artificiali che utilizzano per l'apprendimento certi algoritmi ispirati alla struttura, al funzionamento ed alle connessioni delle reti neurali biologiche (cioè quelle dell'essere umano). Il modo in cui si allenano le reti neurali, può essere definito come deep learning: perché la rete dei neuroni è organizzata su diversi livelli gerarchici. Il primo livello è costituito da neuroni in entrata che rilevano i dati, iniziano con la loro analisi e inviano i risultati al prossimo nodo neurale. Alla fine le informazioni, di volta in volta sempre più precise, raggiungono il livello di uscita e la rete emette un valore. I livelli, nel complesso molto numerosi, che si trovano tra entrata e uscita, sono chiamati livelli nascosti.

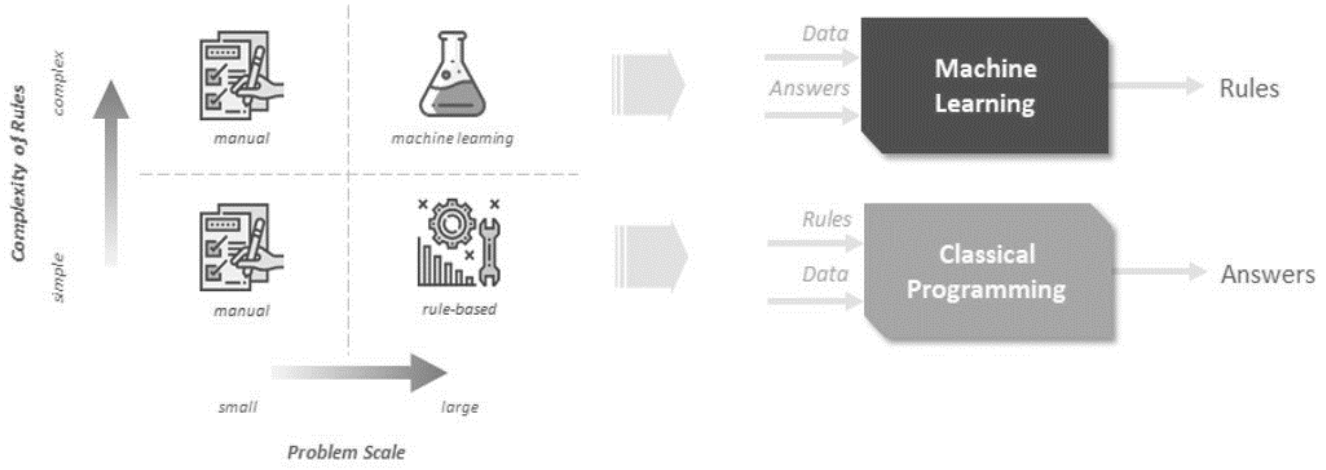
L'analisi approfondita è un processo ciclico, in cui definiamo un modello per il nostro problema, ne testiamo i risultati e lo perfezioniamo (aggiungendo parametri, cambiando pesi e misure, adattandone la struttura) o lo scartiamo e ne proviamo uno nuovo.

Processo iterativo

In molti casi, si scopre che i migliori risultati (cioè quelli che restituiscono la percentuale minima di errore) non vengono raggiunti da un unico modello, ma da un insieme di modelli che cooperano tra loro.

Applicazioni
del Machine Learning

Il machine learning consiste in una serie di tecniche che permettono ai sistemi informatici di prevedere, classificare, ordinare, prendere decisioni e in generale estrarre conoscenze dai dati senza bisogno di definire regole esplicite. Si tratta di un campo dell'intelligenza artificiale che ha lo scopo di insegnare ai computer ad eseguire compiti basati su esempi senza una programmazione definita. A causa della sempre maggiore complessità del business e della quantità di dati da analizzare, stiamo passando da un'intelligenza basata su regole a un'intelligenza basata sui dati, come illustrato nello schema che segue.



I dati vengono utilizzati per addestrare i modelli e consentire loro di imparare a estrarne le conoscenze che gli servono. Questi modelli vengono addestrati utilizzando diversi algoritmi, determinati in base al tipo di compito che vogliamo far eseguire alla nostra soluzione.

Tipologie
di Machine Learning

La scelta del modello è determinata dal problema aziendale. La fase di inquadramento del problema ci aiuta a definire che tipo di risposte stiamo cercando: vogliamo spiegare la situazione attuale? Vogliamo prevedere gli esiti futuri? Stiamo cercando di classificare le nostre attività? A seconda delle risposte a queste domande potremo scegliere uno o più modell

Classification

Avendo a disposizione una serie di dati appartenenti ad un numero finito di categorie note, assegnare ogni campione alla giusta categoria

Clustering

Avendo a disposizione una serie di dati appartenenti ad un numero finito di gruppi non noti, creare i gruppi in modo che gli elementi al loro interno abbiano dei punti in comune

Natural Language Processing

Dallo scritto al parlato; dal parlato allo scritto; il riconoscimento di entità nominate; l'estrazione automatica di approfondimenti come il tono di una recensione o il riassunto di un testo più lungo

Forecasting

Analisi delle caratteristiche chiave di una serie ordinata di campioni, per rilevare la stagionalità delle tendenze e l'influenza di fattori esterni. Previsione dei valori futuri di una variabile target in base alla sua relazione con altre variabili o a determinate serie storiche

Strumenti utili
per il Machine Learning

Negli ultimi anni sono stati lanciati strumenti per automatizzare l'addestramento, la convalida e la selezione degli algoritmi: marchi come DataRobot o C3.ai promettono di automatizzare molte fasi del processo dei dati per renderle accessibili alla comunità di «non Data Scientists». In generale, questi strumenti possono offrire un buon valore aggiunto per compiti specifici (es. confronto e punteggio dei modelli), ma presentano alcune ovvie limitazioni dovute alla loro specifica progettazione e all'automazione che introducono. In sintesi, ci sembra non si sia ancora presentata l'occasione di applicarli wall-to-wall in un'implementazione di machine learning complessa.

Per costruire modelli ed eseguire algoritmi si possono sfruttare diverse librerie ML in grado di semplificare il lavoro del Data Scientist in diverse attività: tutte gratuite e open source, forniscono una potente serie di strumenti per la costruzione e l'esecuzione di modelli di machine learning.

Scikit-learn	Scikit-learn è una libreria di machine learning per Python. Noi la utilizziamo principalmente per vari algoritmi di classificazione, regressione e clustering che possono essere considerati «classici» algoritmi di machine learning (macchine vettoriali di supporto o foreste casuali, per citarne alcuni). Si tratta di una libreria ricca di contenuti, affidabile e facile da usare che si integra perfettamente con le librerie di Python NumPy (numerica) e SciPy (scientifica).
Tensorflow & Keras	Tensorflow è una libreria di algebra computazionale progettata per i flussi di dati e la programmazione differenziabile di una vasta gamma di compiti, nonché forse la più utilizzata per costruire e addestrare reti neurali. Tensorflow è stato prima sviluppato dal team di Google Brain per uso interno, poi rilasciato come risorsa open source. Strumento dalle grandi prestazioni, sia in termini di addestramento che di inferenza, ha una sintassi che può rivelarsi piuttosto complicata. Keras è una libreria di reti neurali scritta in Python che gira su Tensorflow, eliminando la maggior parte delle complicazioni della costruzione di reti neurali su Tensorflow. Tensorflow 2.0, rilasciato in versione alfa in aprile 2019, include Keras come API di default di alto livello per la costruzione e l'addestramento dei modelli di machine learning.
Pytorch & Fastai	Come per Tensorflow e Keras, Pytorch e Fastai sono due librerie Python che si completano a vicenda. Pytorch è una libreria di machine learning Python, utilizzata soprattutto per applicazioni come l'elaborazione del linguaggio naturale e la visione artificiale e originariamente sviluppata dal gruppo di ricerca sull'intelligenza artificiale di Facebook. Fastai è una libreria che funziona su Pytorch e semplifica al massimo il processo di costruzione, formazione e messa a punto di modelli di machine learning, permettendo di costruire prototipi e modelli in tempi brevissimi.

La classificazione dei dati multispettrali e iperspettrali è diventata sempre più importante per rilevare il cambiamento del suolo utilizzando le immagini satellitari.

La classificazione non supervisionata raggruppa prima i pixel in "cluster" in base alle loro proprietà. Per creare "cluster", gli analisti utilizzano algoritmi di clustering di immagini quali K-means e ISODATA. Esistono molti software di uso libero nel campo del telerilevamento. Dopo aver scelto un algoritmo per effettuare il clustering, si determinano il numero di gruppi che si vogliono generare. Questi saranno ancora cluster non classificati perché nella fase successiva si procederà ad identificare manualmente ciascun cluster con classi di copertura territoriale.

Nel complesso, la classificazione senza supervisione è la tecnica più basilare poiché non ha bisogno di campioni ed è un modo semplice per segmentare e comprendere un'immagine.

classificazione dei dati satellitari

Nella Classificazione Supervisionata invece, si selezionano campioni rappresentativi per ciascuna classe di copertura del suolo e il software utilizza questi siti per l'apprendimento e li applica all'intera immagine. Si utilizza la firma spettrale definita nel set utilizzato per l'apprendimento. La procedura in sintesi prevede la selezione delle aree di riferimento, la generazione di un file delle firme spettrali e la classificazione finale.

Sia la classificazione supervisionata che la non supervisionata sono basate sulla creazione di pixel quadrati e ogni pixel ha una classe.

Invece la classificazione delle immagini Object-based raggruppa i pixel in forme e dimensioni rappresentative con una segmentazione multirisoluzione.

La segmentazione multirisoluzione produce oggetti immagine omogenei raggruppando i pixel. Genera contemporaneamente oggetti con diverse scale in un'immagine. Questi oggetti sono più significativi perché rappresentano le caratteristiche dell'immagine. Ma soprattutto, si può classificare gli oggetti in base a texture, contesto e geometria.

Case History #1:

confronto tra due algoritmi di machine learning, Random Forest e Support Vector Machine

L'obiettivo del case history è di testare gli algoritmi Support Vector Machine (SVM) e Random Forest (RF) implementati nel pacchetto opensoruce di R/rminer e disponibile in The Comprehensive R Archive Network repository. Le performance sono valutate usando l'indice K, l'indice di accuratezza e gli errori di classificazione.

SVM utilizza un iperpiano come predittore: il piano divide il data set in due gruppi e gli oggetti sono trasformati in classi usando una funzione matematica detta kernel.

RF consiste in una collezione di alberi decisionali e si basa su tecniche di statistica di bootstrap. RF campiona casualmente il data set disegnando l'albero decisionale; il predittore viene applicato ad ogni ramo e viene scelta la variabile migliore. Le predizioni sono organizzate in un nuovo campione che viene analizzato. In questo studio, RF e SVM sono stai applicati, tramite un framework, a un set di campioni stratificati variabili tra 2% e il 20% della popolazione (pixel dell'immagine). Il framework campiona e bilancia automaticamente il numero dei pixel di una classe, scartando quelli non significativi. Per ogni set di campioni stratificati, vengono creati 10 test set di training e 10 test set di validazione. Il test set di training serve per istruire il modello. Il test set di validazione è ottenuto dalla differenza tra il data set intero e il data set di training.

Il data set di controllo è una classificazione indipendente basata su fotointerpretazione. Per valutare gli algoritmi sono stati calcolati gli indici di accuratezza (Acc), gli errori di classificazione (CE) e indice Kappa Index (K). Gli indici variano tra 0 a 100% e sono stimati con tre diversi approcci: (i) pixel del training set e K-fold cross-validation, (ii) pixel del test set di validazione e (iii) pixel dell'intero test set.

Tramite K-fold cross-validation e test set di validazione i risultati evidenziano che SVM ha performance migliori di RF, mentre con il test set completo, RF ha performance migliori di SVM.

Questo report è stato realizzato con lo studio e l'approfondimento di testi scientifici, articoli e pubblicazioni redatti dai seguenti autori:

Riferimenti Bibliografici

Redazione Network-Digital-360

M. Piragnolo, A. Vettore, F. Pirotti CIRGEO, Centro Interdipartimentale di Ricerca di Geomatica, Università di Padova

Cover Classification." Remote Sensing of Environment 197 (August)

Manuel Torres Artificial Intelligence Manager in Techedge

Renzo Carlucci GEOmedia

Redattori del Report

Contributi tecnici	Marco Vitale
Editing testo	Giuseppe Ciccarelli
Supervisione	Valeria Pucitti
Approvazione	Partner G.O. D.E.M.E.TRA.

D.E.M.E.TRA.

PSR Campania 2014-2020 - Sottomisura 16.1 Azione 2

Report realizzato con il contributo del PSR Campania 2014-2020

Sottomisura 16.1 Azione 2 - Focus Area 3A

Unione Europea
Fondo europeo agricolo per lo sviluppo rurale: *l'Europa investe nelle zone rurali*
REGIONE CAMPANIA