Cosa sono le scienze sociali computazionali?

Scienze sociali computazionali come un territorio che contiene all'interno delle sue discipline. Un **territorio interdisciplinare** in cui tecniche e strumenti di computazione avanzati, assistiti dall'informatica sono usati per fare delle cose comprendere, modellare (i modi in cui questi dati si legano assieme), se utilizzo un modello posso prevedere quei dati all'interno, costruire modelli della realtà, simulare e analizzare dei fenomeni sociali. (ricerca sociale + data science)

Modello di previsione del prezzo dell'affitto

Tre elementi chiave in molte definizioni:

- Interdisciplinarità (ci sono più discipline) (post – disciplinarità)
- Modellizzazione (algoritmi) "costruire modelli"
- Dinamica e complessità sociale (il fatto che i fenomeni vengono svolti nel tempo e il fenomeno della società che invece è molto complesso)

    - **Obiettivi:**
- Trovare **strutture emergenti**, ricorrenti e dotate di **senso** fra le **tracce** della vita contemporanea (capire se ci sono delle strutture all'interno di dati, dentro questi dati posso trovare senso? Cosa c'è dentro questi dati? Ci sono le tracce della vita contemporanea di tutti noi come ad es. braccialetti per fare l'attività fisica
- Formulare **teorie** robuste del comportamento sociale (È possibile da tutti questi dati tirare fuori delle teorie?)
- Prendere **decisioni** guidate dai dati (data – driven)
    - **Aumento esponenziale della comunicazione → Complessità → Big data**

    BIG (data) BANG

- **CSS =** Computanional social science
  L'esigenza di usare strumenti computazionali per capire la società
- **BBS = British – based School of Complexity (**uscire dalla sociologia letteraria per giungere ad una sociologia tecnica, computazionale più complessa)
- **Viene pubblicato un paper "The Coming Crisis of Empirical Sociology" (2007)** = (l'immenente crisi della sociologia empirica) → interviste, questionari ecc. (es. Non rispondere ai questionari), è esplosa la privacy nella società contemporanea
- **Viene pubblicato un altro paper "Computational Social Science" (2009) =** in cui si prende atto di una nuova scienza e questo articolo diventa una sorta di manifesto, come si deve articolare questa disciplina che risponde alla crisi della sociologia e di altre scienze sociali. In questo articolo si parla anche di "post – disciplinarità" (oltre la disciplina).
- **Accademic divide =** divisione delle università ricche e povere (due mondi diversi accademici, il mondo dei poveri e il mondo dei ricchi. Questa cosa è legata anche nel marketing, nelle ricerche di mercato (soldi per lavorare sui dati che sono dei dati legati alle grandi aziende, le aziende danno i soldi alle università che sono più avanti, fanno queste partnership "Google – Apple ecc." università sempre più grandi interessate meno alla scienza e più al profitto.
- **Journal of Computational Social Science =** rivista ufficiale di coloro che si occupano delle scienze computazionali

Tecniche di CSS

- **Data mining** (scavare nei dati)

- A partire dal data mining si è sviluppata la **nuova intelligenza artificiale** = lavorando sulle macchine fatte di neuroni artificiali
- **Social network analysis**
- **Simulazione al computer** (terza grande branca della computational social science) (Agent - based simulation (di questo non parleremo**)**
- **(Sistemi georeferenziati – GIS) (**neanche di questo)

- **Raccolta, esplorazione, analisi e visualizzazione dei dati** (modi di rappresentare i dati suggestivi, informativi che certe volte rappresentano anche l'arte "dati avanzati")

- **Big data (le tre "V") =** sono belli grandi (**v**olume, **v**elocità, **v**arietà)
- **Tecniche di analisi vecchie e nuove**
- **Ricerca tradizionale e ricerca contemporanea: dalla conferma all'esplorazione** (voglio scoprire dentro i dati un modo per fare più soldi, l'importante è fatturare)
- **Prevale la raccolta e l'analisi dei dati a scapito della validità e della qualità dei dati**

CSS e programmi di ricerca contemporanea
Ci sono degli ambiti disciplinari che precedono le scienze sociali computazionali dei quali possiamo trovare delle cose comuni come ad es.

- **Sociologia analitica** (su micro e macrosociologia) → nella sociologia si sono sempre contrapposte queste due visioni
- **Scienza delle reti (network analysis) non solo quelle sociali**
- **Memetica (genetica delle idee e della cultura) Richard Dawkins,** il gene egoista, 1976 = la cultura funziona come una genetica, come viene utilizzato un meme e qual è la sua cultura? Il meme è un pezzo di idee di una società che viene tramandato dal codice genetico
- **Cliometria e cliodinamica per l'analisi storica =** misurare la storia e studiare la storia nei suoi aspetti dinamici, studiare la storia da un punto prettamente quantitativo. La cliometria ha poi influenzato la cliodinamica, utilizza gli strumenti computazionali allo studio della storia, è più recente.
- **Digital humanities (informatica umanistica)** l'informatica applicata allo studio delle scienze umane e **culturomics** (studio del comportamento e della cultura attraverso i testi digitalizzati) es. lo studio degli spartiti musicali attraverso l'informatica
  - **Cultural analytics (LIBRO: Raffaello Cortina, tutto ciò che accade nell'informatica nel mondo della cultura)**
    - **Economia cognitiva per l'analisi di scelte e decisioni**

DATA MINING (tante tecniche "scavare nei dati")

- Una famiglia di tecniche di **estrazione, esplorazione e analisi dei dati**, di tipo **statistico** o **matematico-computazionale,** basata su macchine **"intelligenti" (**sia controllate sia NON controllate dall'uomo "apprendimento supervisionato" o macchine che procedono e fanno tutto da sole), che ha lo scopo d'identificare **pattern significativi d'informazione e relazioni**, in database multidimensionali "tracce" elettroniche e documenti vasti e complessi, altrimenti **non gestibili** attraverso le tecniche più tradizionali

    - Ricavare informazioni e conoscenza da dati vasti, "rumorosi" e apparentemente senza significato (dati "impliciti") e lo facciamo con: → Knowledge Discovery in Dataset (KDD) grazie agli algoritmi sono in grando di estrapolare i dati che ci interessano, machine learning (ML), IA, **Mash-up** (vuole estrarre informazioni ma vuole mettere ciò che non serve di lato) il Mash-up = mescola tre tipi di dati diversi che da soli non significano niente ma che messi assieme assumo significatoad esempio.
    - **Data mining** (scavare dati all'interno di archivi digitali fisici) **e Web mining** (ricaviamo dei dati dai siti web)
- **Text mining e natural Language processing (NLP) → processamento del linguaggio naturale,** noi prendiamo le cose per come le persone leggono o le scrivono e le analizziamo, un modo di analizzare i testi così come la natura crea.
- **Web usage mining** (come le persone scavano I dati e sono collegati ai siti web**, Web content mining** (che testi ci sono all'interno dei siti web, **Web structure mining** (il modo in cui la rete si da una forma e come i link tra di loro riescono a costruire una struttura interessante (quale sito rimanda ad un sito e che rimanda ad un altro sito?
- All'interno di questo web mining c'è il
  Web semantico e XML (Hypertext Markup Language)
  L'estrazione può avvenire in vari modi con:
- **Crawling =** ci sono degli informatici che utilizzano dei ragnetti che attraversano questi tre link (ragnetto che ricostruisce la rete)
- **Parsing** = strutture che servono per capire i pezzi di come è stata composta una pagina web
- **Scraping** = grattare i contenuti dei siti web
- Spigots = bastoncelli che possiamo infilare nei buchi del web ed estrapoliamo fuori cosa c'è dentro
- **API** = Application Programming interface (interfacce di programmazione per l'applicazione) sono delle porte che i programmatori predispongono nel caso in cui quasi un altro programmatore voglia entrare nell'applicazione es. programmatore di facebook che predispone una porta ad un altro programmatore per mostrargli i dati che io metto a disposizione ma come in tutte le porte occorre una chiave: questa chiave ce l'ha solo il programmatore di Facebook che può decidere se dare l'accesso o meno
- **Data streaming** = i dati arrivano a ondate

RSS (Google sheets) (esercizio in aula)

Big data in action

- **Chi produce big data?**
- **Dove si conservano i dati e chi offre risorse e strumenti per i big data?**

- High-end vs commodity hardware (I dati vengono divisi e portati nel computer)
- **Hadoop (HDFS + mApReduce + YARN** per gestire le risorse di calcolo) Hadoop è fatto con un sistema che distribuisce i dati, su questo hardisk è montata una mappa che serve a fare dei calcoli e Yarn è il software che ci fa lavorare.
    - Hardware Failure (le copie di quello rotto vengono utilizzare al posto dell'hardisk bruciato, questo software viene distribuito
    - Le distribuzioni di Hadoop (es. Cloudera) con strumenti di alto livello (Hive) e altre librerie dedicate (es. Spark)
    - Dentro hadoop ci sono dentro sia dati strutturati cioè quelli organizzati secondo righe e colonne (matrice per colonne) dati analizzati attraverso il linguaggio del

database come Siquel (SQL – structured query language) (Impala) e non strutturati (NOSQL) (Cassandra, MongoDB (database), Neo4j ecc.)
    - Noi ci affideremo ai **servizi pay-per use: IBM Cloud, Microsoft Azure, Google Cloud Platform, Oracle, Amazon EMR – Elastic MapReduce (file distribuiti) Amazon web servicy**

DATA STACK e applicazioni di data analytics

Catasta per fare l'analytics, la base è l'infrastruttura fisica

- **Infrastruttura fisica (data center o cloud provider) dove i dati stanno e si modificano**
- **Piattaforma dati (anche da diversi database) dentro le macchine dobbiamo mettere le macchine, il dato va spostato e messo poi dentro la struttura fisica o reale**
- **Applicazioni (calcolo, advanced analytics, interfacce)**
    - **Fogli di calcolo (Google Sheets)**
    - **Business Intelligence per dashboard** (interfaccia con dei grafici, scegliamo delle variabili e ci vengono mostrate delle informazioni es. possiamo cambiare l'anno di cui vogliamo vedere i dati) interattive destinate anche a utenti meno esperti **(Microsoft Power BI** (lo possiamo usare), **Tableau, Google Data Studio)**
        - **Low – code** (con codice o senza codice) o no – code analytics tramite programmazione visuale (**KNIME**, Rapid – Miner)
        - **Code – based analytics (Python, R, Scala) anche per le applicazioni in tempo reale, tramite IDE** (Integrated Development Environment) "software entro i quali vengono fatte delle… **come R studio, Jupyter Notebook, Visual Studio, Pycharm ecc.**

          Data analytics, Data Science e IA
        - Utilizzano gli stessi metodi per fini diversi:
        - La **Data analytics** si concentra sull'estrazione di informazioni e risposte significative dai dati esistenti, con l'obiettivo di rispondere alle domande specifiche poste da organizzazioni e aziende
        - L'**IA** studia come realizzare sotware che replichino l'intelligenza umana e animale, per automatizzare attività che prima erano prerogativa dell'uomo
        - La **Data Science** è una combinazione di tecniche di analisi dei dati, di apprendimento automatico e di intelligenza artificiale per rispondere a domande complesse e sviluppare modelli predittivi

          DATA ANALYTICS
        - Come trasformare dati grezzi in qualcosa di utile e di valore
        - Tre tipi di data analytics:
    - **Descriptive: (Business Intelligence – BI)**
      Che cosa è successo? Ad es. in azienda (quali sono gli andamenti della nostra azienda?)
    - Il report statistico (tabelle e grafici descrittivi)
        - Key Performance Indicators (KPI) → Per avere un quadro di come stanno andando gli affari
        - Dashboard Interattive e management cockpit (con I KPI) "manipolare i dati"
    - **Predictive (Advanced analytics)**
      Perché è successo? Che cosa succederà? Es. come mai ho venduto di meno in questo periodo?
    - **Modelli** (prevedere)
    - **Strumenti diagnostici, business alert, anomaly detection** (c'è qualche anomalia)

- **Forestcast, ROI (Return of Investment), Propensy models (churns e retentions**) es. Azienda che deve fidelizzare dei clienti "servizio telefonico", questi client rimarranno con me o se ne andranno?
- **Segmentazione (di un mercato)**
- **Presciptive (Advanced Analytics)**
  Che cosa fare? Cosa posso fare?
- **Recommendation systems (trading automatizzato, programmatic advertisement)**
  "captare l'offerta"

  PROFESSIONI E RUOLI NELLA DATA ANALYTICS
  - Utente di business (in realtà, solo fruitori)
  - **Business analyst (o data analyst)**
    Connette mondo del business e mondo dei dati (spiega al mondo del business i dati che li racconta e li fa diventare qualcosa che possono essere utilizzati per ottenere valore)
  - Deve essere in grado di estrarre insight da grandi quantità di dati
  - Digital data storytelling
- **Data Scientist**
  Pulisce e trasforma i dati
  Usa matematica, statistica e informatica, per ricavare valore dai dati
  Conosce gli algoritmi di analisi dei dati e costruisce modelli
  Effettua analisi predittive e prescrittive

-Data engineer

Fa funzionare le infrastrutture

Strumenti del nostro laboratorio

- **Account developer (Amazon, Spotify)**
- **Data storage e repository**
- **Software:**
  Applicazioni per l'analisi avanzata dei dati → Algoritmi
- **Linguaggi e ambienti di programmazione (IDE)**
- **Visualizzazione dei dati (dataviz)**

  LA CONSOLE AWS (Amazon Web Services)
- **Dove si crea?**
  - **https://aws.amazon.com/it/**
  - **S3 (Simple storage service) Bucket : mettiamo dentro tutti i dati**
  - **Togliere il flag da "blocca", la disattivazione (mettere il segno flag), bucket a Parigi, bucket: amazg**
  - **Sign in: Root user (e-mail e password)**

  14/10/2024

  Chiave per accedere:
  IAM (pannello di controllo):

CHIAVE DI ACCESSO: AKIA4SZHNUEZDMW4CUG5

CHIAVE DI ACCESSO SEGRETA: tNvB3X6v3O2tkfeiGYijqW2sA5q9h3w03LMPHGsp

Nodi computazionali (Knime) : trascinare un nodo

Big data e modelli predittivi
(Introduzione a KNIME)

Cenni di Machine learning

Affidare alle macchine il compito di imparare delle cose si colloca in un ampio progetto che è il progetto di affidare alle macchine il compito dell'uomo. Possono le macchine essere intelligenti?

Fornire alle macchine degli strumenti per conoscere

- **Machine Learning**/ **Deep learning** (apprendimento attraverso le reti neurali) / **Generative AI**/ **AGI** (intelligenza artificiale generale)
- IA "debole" → algoritmi (non riesce a raggiungere i diversi risultati delle altre intelligenze artificiali. Inoltre, simula una o pochissime facoltà dell'intelligenza umana. Nel caso del machine learning viene simulato ciò che viene chiamato "l'apprendimento", "debole" perché gli algoritmi sono più semplici.
- Come possiamo noi replicare questo processo di apprendimento nelle macchine?
- Le persone lo fanno attraverso tre modi:
  - **Apprendimento non - supervisionato** (imparare esplorando il mondo e connettendo)
  - **Apprendimento supervisionato** (imparare dagli esempi)
    Creare un "programma", unendo dati e risultati, pre prevedere casi nuovi
    I dati sono organizzati in feature e target (o label)
  - **Apprendimento rinforzato** (imparare dagli errori

1° apprendimento supervisionato, 2° non supervisionato, 3° rinforzato

In cosa consiste l'apprendimento supervisionato? Cosa fanno le macchine per imparare sull'apprendimento supervisionato?

Le macchine hanno la possibilità di vedere dei dati e dei risultati che riguardano quel blocco di dati

Es. consumatore (comprare/non comprare uno yogurt) → caratteristiche delle persone che ci indicheranno se le persone compreranno o non compreranno uno yogurt. Chi lo acquista, chi non lo fa cosa succede? La macchina cerca di costruire un programma che riesce a unire i dati con i risultati per prevedere il comportamento di nuovi individui.

Come sono organizzati questi dati?

Abbiamo delle caratteristiche delle persone che chiamiamo feature e il target (compro/non compro)

KNIME: label

Apprendimento supervisionato

Le tecniche dell'apprendimento cambieranno con le variabili target

- Variabili target **numerica** (appartamento: quanto è l'affitto?)
  Sceglierò: Algoritmo di regressione
- Variabile target **dicotomica o binaria**

Sceglierò: algoritmo di regressione logistica
Oppure: algoritmo di classificazione (**alberi decisionali, random forest**, support vector machine ecc.) classificare un'etichetta

- Variabile target **nominale**
  Sceglierò: algoritmo di classificazione (alberi decisionali, random forest, support vector machine ecc.)

Come si creano questi modelli di machine learning supervisionato?
Cosa faremo con KNIME?
Quali sono le caratteristiche che ci interessano del modello? Cioè, le feature (caratteristiche) "variabili che io penso possano essere responsabili del fenomeno (compro/non compro) o anche per riconoscere il tipo di e-mail? Quali sono le caratteristiche che mi fanno vedere quel fenomeno?

- **Selezione delle feature**
- **Scelta dell'algoritmo di M-L**
- Dobbiamo stabilire come utilizzare la matrice etichettata, dobbiamo stabilire quanta parte di questa matrice mi servirà per l'apprendimento e quanta parte della matrice io destinerò per gli esempi? Per l'apprendimento? metodo → **Hold – out method**
  Training set (ca 70% dei dati), validation set (15%), test set (15%) un 15% lo uso per fare la validazione cioè: vediamo se ora che la macchina ha appreso gli esempi, vediamo se la macchina prevede degli esempi (?) capiamo se il modello prevede correttamente l'output → previsioni / questo modello prevede correttamente o no? Il 30% viene utilizzato per capire se le previsioni sono corrette
- In base ai risultati della valutazione del training set, faccio una valutazione ed eventuale raffinamento del modello (ottenere una percentuale ancora corretta)
- Infine, avviene la fase di **Distribuzione (deployment)** → con un software, un programma che può essere venduto a chi vuol fare un'analisi di mercato

La regressione lineare multipla (la retta deve passare vicino ai punti)

- Un'estensione della regressione lineare semplice
- Il metodo dei minimi quadrati
- Modello multivariato
  Cosa conta davvero nella previsione di un risultato (target)?
  $Y = b0 + b1x1 + b2x2 + \dots + bnxn$ $y = a + bx$

- **Affitto** = b0 (intercetta) + b1 superficie + b2 stanze + b3 piano
  Prezzo dell'affitto e caratteristiche dell'appartamento

KNIME

- **Dove si scarica?**
  https://www.knime.com/downloads
- **Come è fatto?**
  Nodi e workflow
  - Input e output
  - Manipolazione
  - Analytics
  - Visualizzazione e reporting
  - Flow control
  - Altro: API, R, Python
- Extensions
- SCENARIO: Un'agenzia immobiliare
  Rapporto finale KNIME

Deployment (distribuzione)

Questi modelli vanno distribuiti, modelli con dati completamente nuovi.

Fitting = il modello deve calzare sui dati

- Over – fitting (alta validità interna, scarsa validità esterna)
- Under – fitting (modello troppo semplice)
- Usare il modello con dati nuovi: problemi di fitting
  Modello di machine learning che impara a distinguere dei pallini gialli e le crocette verdi, il modello è la linea rossa che cerca di dividere i pallini gialli dalle crocette verdi
  1) Fitting = veste bene il modello
  2) Over – fitting = oltre / alta validità interna
  3) Under – fitting =

Gli alberi decisionali

- A quale categoria appartiene un elemento? Algoritmi di machine learning che rispondono a questa domanda (es. a quale categoria appartiene questa persona?) voglio sapere se questa persona rimarrà fedele alla compagnia telefonica che io gestisco o mi tradirà. Churner = fedele
- Un albero decisionale è una rappresentazione grafica del processo che porta, attraverso una serie di scelte, ad una categoria o scelta
- Radice, foglie, rami
- Gli alberi decisionali ci forniscono delle idee molto chiare su ciò che sarà la decisione di un consumatore, per esempio, queste decisioni poco chiare le riscontriamo nel deep learning che saprà prevedere il futuro ma non sappiamo come fa a farlo. Le spiegazioni sono esplicite e chiare (vs deep learning)
- Tre categorie:
  - Se il target è dicotomico si dice che utilizziamo gli alberi di classificazione (classification trees) (compra non compra?) decidere dove un oggetto viene collocato in una classifica

- Se il target è cardinale, alberi di regressione (regression trees)
- Esistono anche alberi di classificazione e regressione
- Algoritmi principali: ID3, C4.5 e C5.0, CART (Classification and Regression Trees). CHAID (Chi. Square automatic interaction detection), QUEST (Quick unbiased
- Altri algoritmi: CAL5, FACT, LMDT, T1, PUBLIC, MARS

Come funzionano questi algoritmi?

- Serve una misura della capacità classificatoria delle variabili
- Guadagno d'informazione, indice di Gini, riduzione della varianza, (chi quadrato x2)
- L'algoritmo ID3:
  Misurare l'entropia della matrice (il disordine o l'incertezza di un sistema)

  E = (formula)

Entropia = quanto è incerto l'esito di y

- Il guadagno d'informazione è la differenza tra entropia totale ed entropia considerando un'altra variabile
- Sapere di avere una probabilità mi aumenta l'incertezza, quindi l'algoritmo confronta: opzione dicotomica
- Esempio: Ciccio va o non va al concerto?
  Ciccio maschio hip hop basso 5 cd

Pulire e preparare i dati

- **Missing values** (buchi mancanti da togliere)
- **Outliers** (estremi, vanno tolti perché distorcono i parametri della macchina)
- **Binning** (ricodifica delle variabili cardinali) divide in fasce
- **Normalizzazione** variabili cardinali che entrano in un range prestabilito ovvero tra 0 e 1
- **Standardizzazione** (nuova distribuzione) media zero e standardizzazione pari a 1
- **Riequilibrio delle classi bilanciate** (molto, abbastanza (1 categoria) poco, per nulla (1 categoria)
- **Data reduction** (si fa con l'analisi fattoriale)
- **Campionamento dei casi** (quando i casi sono troppi e l'algoritmo non ce la fa a svolgere tutti i casi, allora occorre estrarre un campionamento

Costruire il modello

Prima master
Poi quello storico cvs

28/10/2024

Matrice di confusione

Knime: qual è il valore reale della matrice?
No e si che si riferiscono ai valori predetti
Accuracy: rapporto tra indovinati e le previsioni 956+ 145 e si divide per tutto

Valutare il modello: la matrice di confusione

- **Incrociare valori osservati e valori previsti (nodo SCORER)**
- **Accuratezza (accuracy**) = previsioni corrette (giallo)/" diviso "Totale casi (verde)
- **Precisione (precision**)= Veri positivi (previsioni positivi corrette) /Totale **positivi REALI**
- **Sensibilità o richiamo (sensitivity o recall**) = Veri positivi (previsioni positivi corrette) /Totale **positivi PREVISTI**
- **Specificità (Specifity)** = Veri negativi (previsioni negative corrette) /Totale **negativi REALI "in che misura io misuro i veri negativi?"**
- **Qual è la proporzione di falsi negativi?**
  P (proporzione) di falsi negativi (su totale negativi PREVISTI) (mancato allarme) → sono malato ma non se ne accorge nessuno **e**
  p di falsi positivi (su totale positivi PREVISTI) (falso allarme) → non sono malato ma risulto malato nel test covid

knime: compagnia telefonica vodafone
No = vuol dire sì (positivo)
SI = vuol dire no (negativo)

Valutare il modello: altre misure

- Dobbiamo utilizzare la **Kappa di Cohen** e cosa misura?
- La concordanza tra valori reali e le previsioni
- Varia tra 0 e 1 (deve essere maggiore di 0,4 "se superiamo lo 0,4 va benissimo"
- L'altra misura è la F- measure (o F1 o F-score) e a cosa serve?
  Serve all'ottimizzazione degli algoritmi, quando si vogliono considerare assieme precisione (previsione positivi) e sensibilità -racall (peso dei falsi negativi)
  - >0,9 eccellente
  - 0,8 – 0,9 buono
  - 0,5 – 0,8 medio (è più bella la kappa di Cohen)
  - >0,5 scarso

Knime: accuracy 0,783

Raffinare il modello: due strategie

- **Iterativa** (aggiusta e valuta)
- **Computazionale** (ensemble learning o apprendimento d'insieme)
  - **Si tratta di generare, comparare e aggregare grandi quantità di modelli (alberi) più semplic**i con feature e casi diversi (alto impatto computazionale, opacità euristica) "vengono campionate le righe", funziona ma perché funziona?

- Seconda strategia computazionale viene eseguita da **tre approcci**:

1) **Boosting e gradient boosting**
2) **Bagging (abbraviazione di bootstrap aggregation) (es. Random Forest)** " quello che utilizzeremo noi"
3) **Stacking** (accatastamento → comprende i clustering ensemble = vengono messi assieme una serie di clustering)
    - L'importanza delle piccole differenze di performance (medicina, marketing)
    - **Curva ROC** (Receiver Operating Characteristics)
    - Cosa mettiamo sull'asse delle x → tasso falsi positivi (falsi allarmi) (asse x) vs tasso veri positivi (sensibilità) (asse y)
    - Pari o superiore a 0,5 SOGLIA DA 0 A 1
    - **Calcolo FP e VP (coordinate), per ogni step da 0 a 1 della soglia negativo/positivo**
    - **La linea a 45° rappresenta il caso**
    - **Più è vicina a 1 e migliore sarà**
    - **L'area sotto la curva**

	P	**Churner (CAT) (0,5)**	**CAT (0,3)**	**CAT (0,7)**
Franco	0,63	C	C	nc
Ciccio	0,32	nc (non churner)	C	nc
Mimmo	0,8	c	C	C
Enzo	0,4	nc	C	nc
Totò	0,2	nc	NC	nc
Pippo	0,7	C (churner)	C	C

Albero decisionale random forest (quanti modelli creare), ottenere valore di accuracy molto elevato, curva ROC
Kappa di cohen superiore a 0,4
Accuracy più alto è meglio è
f- measures vicino a 1

11/11/2024

CLUSTERING

E' una tecnica di Analisi multivariata (più variabili) che ci consente di raggruppare i casi all'interno di gruppi che sono molto omogenei al loro interno ed eterogenei tra di loro.

Principi di clustering

- Raggruppare i casi in teorie
- Come raggiungiamo questo obiettivo? Come raggruppiamo queste persone? Ci sono delle tecniche di clustering
- Tecniche di clustering:

- Tecniche gerarchiche
- Tecniche iterative o delle partizioni ripetute (K- means)
- Tecniche della densità locale (MDS)
- Reti neurali (deep learning) "profondo" rip. (supervisionato → quando abbiamo degli esempi e non supervisionato → non abbiamo degli esempi, la macchina da sè), le macchine analizzano delle matrici di dati e capiscono se è possibile fare dei raggruppamenti e dividerli in varie somiglianze

- Principio gerarchico: (raccogliere delle classi in dei contenitori) strategia dall'alto verso il basso o dal basso verso l'alto, l'idea è quella di usare una scala concettuale dal basso verso l'alto. Posso anche fare dall'alto verso il basso, dividere gli oggetti di ogni giorno.

- Basso verso l'alto agglomerativi o dall'alto verso l'alto divisive

- Tecniche iterative o delle partizioni ripetute (K – means)

  Il ricercatore decide quanti gruppi vuole, mette gli oggetti a caso decidendo dove volerli raggruppare. Tecniche iterative → si comincia da un gruppo, si prende a caso un oggetto e si decide se spostarla o lasciarla li casualmente.
  Omogeneità → i gruppi sono simili tra loro
  Eterogeneità → ogni gruppo è diverso dall'altro

- Tecniche della densità locale (MDS)
  Gli oggetti sono sempre delle persone. L'addensamento localizzato in un punto che suggerisce la presenza di un cluster.

- Reti neurali (modelli matematici che simulano l'attività del cervello (chat gpt, intelligenza artificiale generativa) (deep learning)

  L'algoritmo k-means:
  - Il ricercatore sceglie quanti gruppi vuole, Scelta del numero dei gruppi (la ricercatrice fa delle prove) "prova ed errore"
  - Iterazione intorno ai centroidi (sposta gli oggetti vicino al centroide)

- **Vantaggi:** semplicità computazionale (matrice delle distanze), efficienza, scalabilità, non richiede campioni probabilistici
- Trattamento degli outlier (valori estremi di una distribuzione) e normalizzazione (le variabili devono rientrare su una stessa scala)

Clustering in KNIME

- Trattamento degli outlier e normalizzazione
- Nodo numeric outliers
- Nodo normalizer
- Nodo k – means
- Nodo line – plot

Rossi → cluster 1 pluralisti
Blu → consumatori attenti cluster 2
Cluster 0 → esigenti verdi

Reti Neurali

- Intelligenza artificiale connessionista: un approccio bottom-up (a partire dai neuroni si è pensato che si potesse riprodurre le facoltà umane) (fino a prima della Seconda guerra mondiale filosofi e scienziati si sono cimentati nell'intelligenza umana manipolandone i simboli. Se una macchina sembra essere una persona, allora diceva Touring che è intelligente). 1948 (storia per la tesi, ricerca sul neurone e neurone artificiale e quindi le reti neurali).
- Neuroni. Cervello, apprendimento (deep learning)
- Neurone: dentriti, soma e assone

  X1 – w1—b → (sommatoria)/ F → y
  X2 – w2 → (sommatoria)/F →y
  Somma moltiplicata questi pesi (w) dell'input più o meno questo bias (errori), La F è una funzione che stabilisce quando scatta la corrente.
  Modellino chiamato Percettrone → ovvero un neurone artificiale (idea del 1948 ma fu realizzata con l'elettronica del tempo nel 1956)

- Input (x), bias (b), **funzione di attivazione**: Y cioè il risultato
- **Y = f (x1*w1*+x2*w2+b)** che ci ricorda la regressione lineare multipla
- **Se la somma cioè Y supera una certa soglia sarà 1**
- **Se la somma cioè Y non supera una certa soglia sarà 0**
- Sigmoid
- Tanh
- Rectified Linear Unit (ReLU)
- Funzione identità: y = x1 ecc (lineare)

- **Funzione gradino** (step function) → **Stabilita una soglia, Y = 0 o Y=1 (c'è un'interruzione)**
- **Funzione logistica (sigmoide) → Y varia in modo continuo tra 0 e 1 (derivabile)**
- **Tangente Iperbolica → come una sigmoide, varia tra -1 e +1 (asintoto a -1, accelera il learning)**
- **Softmax → la dobbiamo scegliere quando avremo una variabile nominale,** Si usa quando abbiamo più neuroni in output (score di probabilità per ogni modalità) es. chat gpt
- **RELU (Rectified Linear Unit) → Y > 0, se X > 0** (funziona benissimo per la computer-vision) es. riconoscimento di immagini (non c'è un'interruzione)
- **I neuroni vengono combinati in un'architettura organizzata in layer (strati):**
  primo strato **(input), hidden (**in mezzo, cioè il secondo strato), **output** (terzo strato)
- Ogni neurone ha il proprio insieme di pesi e bias
- I neuroni dello stesso layer usano di solito la stessa funzione di attivazione (ogni neurone ha la sua sommatoria e la sua funzione)
- **Dense layer:** tutti i neuroni del layer con tutte le possibili connessioni attivate precedente con tutti i layer di quello successivo

Reti neurali – Alcune architetture e scopi

- **Feedforward neural networks (FFNN)** (non ci sono percorsi che agiscono all'entrata o fuoriuscita della rete)
- **Classificazione, regressione, fraud detection** es. dati o frode delle banche
- **Backpropagation (retroazione)** (reti che utilizzano qualche feedback del segnale di uscita alle reti neurali) "propagazione dell'errore"
- Errore quadratico, funzione di costo, gradiente, learning rate, epoche
- **Convolution Neural Networks (CNN)** (**Reti neurali convoluzionali**) es. creare delle immagini

Come funzionano?

- **Per dati a griglia** (pixel che vengono colorati in una griglia, ogni pixel è un'informazione e se ogni griglia corrisponde al pixel vedremo l'immagine)

- **Recurrent Neural Networks (RNN) (Reti neurali ricorrenti, 2014)**

  Come funzionano?

  - **Dati sequenziali, diacronici, serie storiche, testi** es. chat gpt
  - **Correzioni al problema della short-term memory**
  - **Si utilizzano dei gate (correttori) che regolano il flusso delle informazioni, selezionando quelle più rilevanti**
  - Queste agevolazioni si chiamano:

- Gated Recurrent United Layer
-

- **Probabilistic neural networks (PNN) (rientrano nelle FFNN)**
  Vengono utilizzate per fare **classificazione, clustering** o **riconoscimento di pattern** (es. KNIME consumatore/carrello)

18/11/2024

- Queste reti neurali si basano sul teorema di Bayes, che viene utilizzato in statistica per risolvere delle probabilità. (Quando vogliamo conoscere la probabilità che si verifichi un evento, se vogliamo delle info in più possiamo aggiornare delle probabilità che potranno essere solo teoriche) es. moneta (testa/croce), teorema che aggiorna delle probabilità sulla base della nostra conoscenza es. tempo
- Si stimano la probabilità delle distribuzioni dei dati di input (suppongo che accadano delle cose)
  Come sono realizzate queste reti PNN?

  Con quattro layer:
- Input layer (tanti nodi quanto sono le feature (variabili) primo layer
- Il secondo layer che chiamiamo Pattern layer (layer dei prototipi/nodi), costruiamo dei modellini statistici e dentro questi layer, ogni nodo sarà un modellino statistico.
- Tanti nodi quanti sono i casi (ogni caso appartiene a una classe)
- Come è fatto questo prototipo/nodo?
  Con la **funzione gaussiana della distanza tra prototipi** (o pattern) e **input (prende dei valori e li fa cambiare con questa funzione qui)**
- Mentre il terzo layer viene chiamato layer di somma (i nodi prendono i nodi in uscita e fanno la somma di questi
- Summation layer (PDF o probability function di ogni classe, **sommando le funzioni**)
- Output (decision node) – viene scelta la categoria con probabilità più alta
- I dati di input vanno normalizzati (min-max, 0-1)
- **Vantaggi:** non sensibili ad outlier, non-linearità, apprendimento veloce, accuratezza
- **Svantaggi:**

  Addestrare una rete neurale (backpropagation)
- **Come funziona? E cosa significa che la rete apprende?**

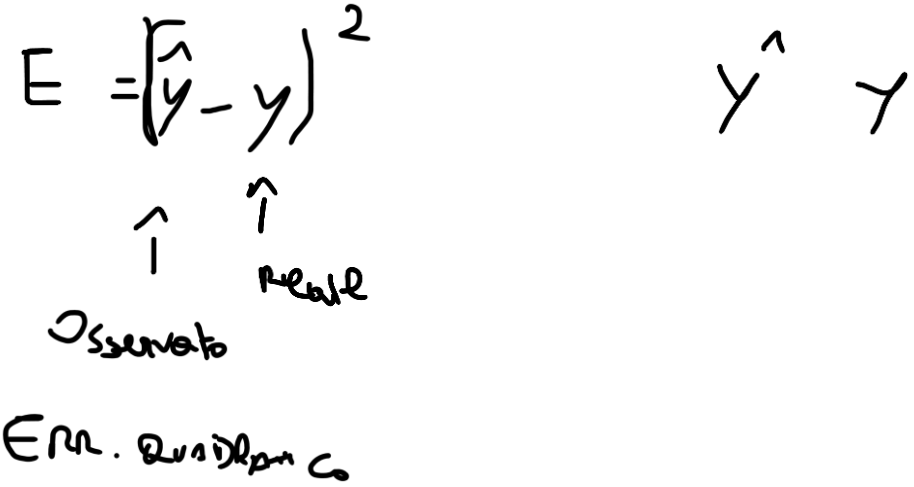
L'apprendimento di una rete consiste nella sua modifica dei pesi iniziali, quando li modifichiamo, la rete apprende. Trovare pesetti giusti per ogni link in modo che la rete dia la massima performance.
Come funziona?

- Assegnazione di pesi casuali alla rete e caricamento del training set, (set di dati che avrà delle variabili e poi delle y di dati)
- Misurazione dell'errore tra valori attesi e previsti tramite una **loss function** (calcolare un errore tra i valori attesi "etichetta che la rete doveva indovinare" e l'etichetta che la rete ha previsto lavorando con quelle informazioni attese.
  Es. etichetta 2 mentre il pc dice 0 (valore che è stato previsto)
- **I pesi verranno aggiornati continuamente, di epoca in epoca, e sulla base di funzione di costo,** cerco di ottenere dei valori migliori di pesi per aggiornare la mia rete, tramute una versione efficiente di **gradient descent** (es. **Adam del 2015**)
- **Algoritmo che è capace di esplorare la funzione di costo e di trovare dei valori minimi → gradient descent**
- Gradient descent (ci dice se la curva sta scendendo o se sta salendo)
- Tangente che ci dice se l'errore Se sta scendendo, continua ad aumentare il valore di w1 oppure
- Se sta salendo, torna indietro quindi scende il valore di w1
- Un parametro del gradient descent è il learning rate
- La procedura si ripete fino a raggiungere una configurazione ottimale

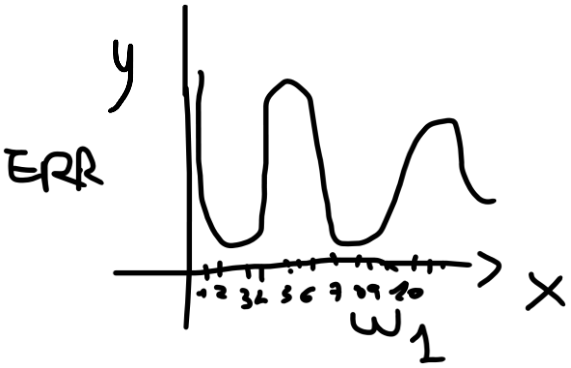
obiettivo	Dimensione del layer di output	Funzione di attivazione	Loss function
Classificazione binaria	1	sigmoide	Binary cross enthropy
Classificazione multiclasse			

Framework open source per le reti neurali

- **TensorFlow:** una libreria open source che ci consente di progettare delle reti neurali, per l'apprendimento automatico
- **Sviluppata da Google brain (2015)** → fa funzionare molte app di Google (es. intelligenza artificiale)
- **L'importanza delle GPU (processore grafica del computer) e non CPU "scheda anvidia"**
- **Keras** è una libreria "amichevole" per la prototipazione di reti neurali
- **È scritto in Python**
- Supporta come back-end TensorFlow, Microsoft Cognitive Toolkit (CNTK) e Theano



Errore = valore osservato – valore reale elevato al quadrato



Quanto vale l'errore rispetto a → 1,2,3,4,5,6,7,8,9,10

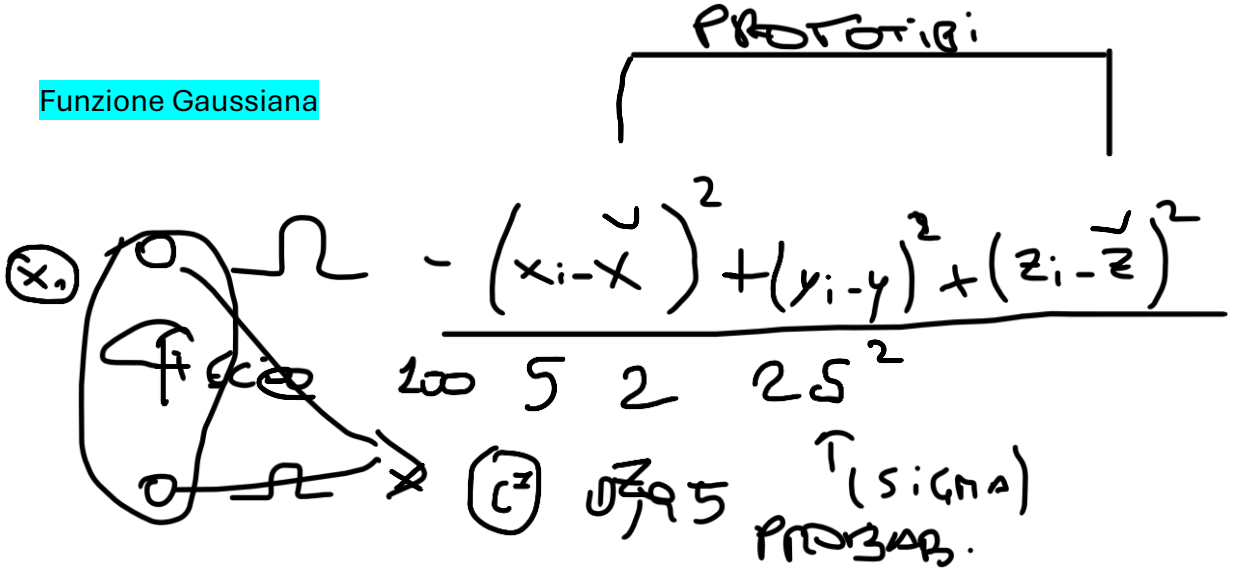
Dobbiamo minimizzare la funzione di costo

Al variare del pesetto, quanto variano gli errori?

Mi vado a calcolare l'errore → es. 4 perché la linea è più bassa in quel punto

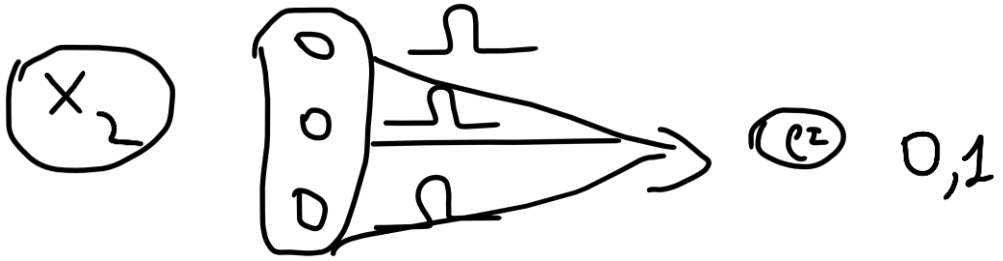
Dovremmo trovare uno spazio con miliardi di dimensioni

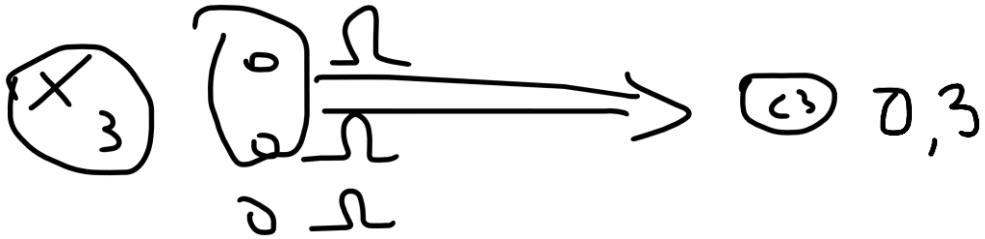
Funzione Gaussiana



٧

FRANCO 2007 2





FRANCO

0,32

0,87 → Prob. più ALTA

0,01