Esami analizzati

- Simulazione di metà corso
- Scritto del 2023-02-12
- Scritto del 2022-12-21
- Scritto del 2023-01-16
- Scritto del 2023-12-19

General tips

> Selezionare la sentenza CORRETTA…

Solitamente prende 4 sentenze vere e 3 di queste le nega, quindi se sei indeciso seleziona quella non negata, è vero anche per quando le sentenze sono ERRATE, ma più raro che il trick funzioni

La risposta più lunga e dettagliata solitamente è la corretta, è più importante il dettaglio della lunghezza in questo caso

Alberi di decisione

- Possono essere utilizzati sia con features discrete che continue (basta usare un threshold)
- Il costo computazioneale della predizione è molto basso
- Hanno una forte tendenza all'overfitting
- Possono esprimere qualunque funzione di classificazione
- La profondità dell'albero è minore o uguale al numero di features (con features discrete)

Random Forest

- Richiedono tecniche opportune per la creazione di alberi di decisone diversi relativi a uno stesso dataset
- NON tendono a migliorare l'explainability (spiegabilità) degli alberi di decisione riducendo l'instabilità nella selezione degli attributi
- Tentano di mitigare il fenomeno dell'overfitting tipico degli alberi di decisione
- È una tecnica di apprendimento ad "ensemble" basata su di una combinazione di alberi di decisione

Naive Bayes

- Numero di parametri nel caso di feature booleane

$$num\_classi + num\_classi * num\_features$$

- Numero di parametri nel caso generico

$$num\_classi + num\_classi * \sum_{f \in F} possibile\_values(f) \quad where \quad F = set\_of\_features$$

- È una tecnica di tipo generativo, in queanto cerca di determinare la distribuzione delle varie categorie dei dati
- Si suppone (ingenuamente e per semplicità) che le features siano indipendenti tra loro
- Fornisce un modo computazionale efficiente per approssimare la distribuzione congiunta di probabilità delle features
- (ERRATA) Non può essere utilizzata se le features non sono tra loro indipendenti, date le classi (Errata per il "date le classi nella frase")

Apprendimento supervisionato

- Apprendimento di funzioni basato su esempi di training composti da coppie input-output
- Può comprendere sia problemi di regressione che di classificazione
- La definizione della ground truth può richiedere l'intervento umano ed essere onerosa
- NON richiede la costante supervisione di un esperto durante il training

Backpropagation

- Si effettua solo durante il "training", non nella fase di "inference" (calcolo in avanti)

- Ha un costo (in termini di tempo) paragonabile al calcolo in avanti lungo la rete
- È l'algoritmo per il calcolo della derivata parziale della loss rispetto a ogni parametro della rete
- L'algoritmo calcola il gradiente un layer alla volta, sfruttando la regola matematica per la derivazione di funzioni composte
- Si riduce a semplici calcoli algebrici facilmente parallelizzabili in strutture di calcolo tipo GPU
- Richiede la memorizzazione delle attivazioni di tutti i neuroni della rete durante la forward pass
- (ERRATA) Viene effettuata unicamente lungo le skip connections delle reti residuali, per evitare perdita del gradiente
- (ERRATA) Tipicamente, il gradiente viene artificialmente rinforzato ad ogni layer attraversato per contrastare il fenomeno della sua scomparsa (vanishing)

Learning rate

- Un learning rate alto tipicamente velocizza il training ma potrebbe saltare sopra al minimo
- E' un iper-parametro che definisce la lunghezza del passo durante la discesa del gradiente
- Il learning rate può variare durante il training
- NON è una metrica che misura la capacità di apprendimento del modello

Gradiente

- Ridurre la dimensione del minibatch AIUTA ad uscire da minimi locali
- Aumentare il learning rate AIUTA ad uscire da minimi locali
- Aggiungere un "momento" al gradiente, cioè parte del gradiente del passo precedente AIUTA ad uscire da minimi locali
- Fare clipping del gradiente in un range prefissato NON AIUTA ad uscire da minimi locali
- La tecnica della discesa del gradiente potrebbe convergere ad un minimo locale
- Il risultato della tecnica di discesa del gradiente dipende dalla inizializzazione dei parametri del modello
- È opportuno decrementare il learning rate verso la fine dell'apprendimento
- (ERRATA) Può essere applicata solo se la funzione da minimizzare ha una superficie concava

Scomparsa del gradiente (vanishing gradient)

- La scomparsa del gradiente (progressiva diminuzione della sua intensità) è dovuta a backpropagation in reti profonde
- La scomparsa del gradiente (progressiva diminuzione della sua intensità) NON è dovuta a dati troppo rumorosi
- La scomparsa del gradiente (progressiva diminuzione della sua intensità) NON è dovuta a dati malamente processati
- La scomparsa del gradiente (progressiva diminuzione della sua intensità) NON è dovuta a troppi pochi dati di trainig a disposizione
- La scomparsa del gradiente (progressiva diminuzione della sua intensità) NON è dovuta a training eccessivamente lungo
- Se il gradiente tende a zero i parametri non sono più aggiornati e la rete smette di apprendere
- Il problema è mitigato dall'uso di link residuali all'interno della rete
- Il problema è fortemente attenuato dall'uso di ReLU (o sue varianti) come funzione di attivazione per i livelli nascosti della rete

Dimensione MiniBatch

- AUMENTANDO la dimensione del minibatch durante il training la backpropagation è effettuata con MENO frequenza ma l'aggiornamento dei parametri è PIÙ accurato
- DIMINUENDO la dimensione del minibatch durante il training la backpropagation è effettuata con PIÙ frequenza ma l'aggiornamento dei parametri è MENO accurato

Entropia

È una misura del grado di disordine della variabile aleatoria

$$H(X) = -\sum_{i=1}^{n} P(X = i) \log_2 P(X = i)$$

- Aggiungere alla funzione obiettivo una componente tesa ad AUMENTARE l'entropia ha l'effetto di ridistribuire la probabilità in modo più bilanciato tra tutte le classi
- Aggiungere alla funzione obiettivo una componente tesa a DIMINURE l'entropia ha l'effetto di focalizzare le scelte sui casi più probabili

- È MASSIMA quando la probabilità è distribuita equamente tra tutti i valori
- È MINIMA quando la probabilità è distribuita tutta in un unico valore
- Il range dei valori per l'entropia è tra 0 e $log(n)$ dove n sono i possibili valori della variabile aleatoria discreta

Cross Entropy

$$H(P,Q) = -\sum_{i=1}^{n} P(X=i)\log_2 Q(X=i)$$

- NON è simmetrica
- È uguale alla divergenza di Kullback-Leibler $KL(P,Q)$ più l'entropia $H(P)$ di P
- Misura la loglikelihood di Q data la distribuzione P (attenzione all'ordine!)
- Ha un valore MINIMO quando $P = Q$

Mutua Informazione (Information Gain per alberi di decisione)

- È una funzione simmetrica
$$I(X,Y) = I(Y,X)$$
- Può essere utilizzata per guidare la selezione degli attributi durante la costruzione di un albero di decisione
- Misura il guadagno di informazione su Y dopo aver osservato X
- NON coincide con l'entropia $H(Y \mid X)$ di Y dato X

Apprendimento auto-supervisionato

- Qualora i dati di input possano essere considerati come annotazioni (labels) per guidare l'apprendimento, come nel caso degli autoencoders
- NON quando il modello è in grado di riconfigurare in modo automatico la propria architettura
- NON quando il modello è supposto contribuire alla creazione di nuovi dati di training
- NON quando l'apprendimento pervede una sinergia tra l'uomo e la macchina

Probabilità condizionata $P(A \mid B)$

$$P(A \mid B) = \frac{P(A \cap B)}{P(B)}$$

$$P(A \mid B) = P(A) \cdot \frac{P(B \mid A)}{P(B)}$$

- $P(A \mid B) \geq P(A \cap B)$ (poichè è un rapporto di due numeri tra 0 e 1)

Distribuzione congiunta (per N variabili aleatorie discrete)

- È la distribuzuione di probabilità di tutte le possibili tuple di valori per le variabili
- NON richiede il calcolo di un numero esponenziale di parametri
- Permette il calcolo di probabilità condizionali tra le features
- Permette il calcolo di eventi condizionati
- Permette di fare predizioni
- Consente una visione distinta delle singole features
- Consente il calcolo delle probabilità marginali delle singole features
- Il suo calcolo presenta problemi di scalabilità all'aumentare delle features

Overfitting

- È problmatico se ho pochi dati di training
- È problmatico se dispongo di un modello molto espressivo
- È problmatico se ho un training molto prolungato
- NON è problematico avere dati molto rumorosi
- Per contrastare il fenomeno

  - Si può usare l'early stopping
  - Si può usare la data augmentation
  - Si può usare l'introduzione di dropout layers
  - NON si può usare l'aggiunta di skip connections

Dado/moneta truccata (MLE)

$$P(X^n = \alpha_i \mid \theta) = c_{\alpha_i} \cdot \prod_i \theta_i^{\alpha_i}$$

Where α_i is the number of i in the sequence and c_{a_i} is a combinatorial constant not depending on θ

- Per calcolare quale delle due situazioni è più probabile calcolando solo la produttoria, tanto c è costante per entrambi i casi
- Esempio
  - Due dadi, uno normale e uno truccato che restituisce un 6 con probabilità 0.5 e gli altri valori con probabilità 0.1
  - Faccio due lanci con lo stesso dado e osservo un 3 e un 6, cosa posso concludere?
  - Maximum Likelihood Estimate del dado truccato: $1/10 \cdot 1/2 = 1/20$
  - Maximum Likelihood Estimate del dado normale: $1/6 \cdot 1/6 = 1/36$
  - Maggiore nel primo caso, quindi è più probabile che sia il dato truccato

Modelli generativi

Modelli che cercano di apprendere la distribuzione di probabilità dei dati

- Generative Adversarial Networks (GAN), Variational AutoEncoders (VAE) e Diffusion Models sono esempi di tecniche generative profonde
- Un tipico esempio di questa tecnica è il NAIVE BAYES
- NON sono modelli meta-teorici rivolti alla automatizzazione della generazione di reti neurali
- NON è il processo che automatizza la generazione di reti neurali
- NON è l'uso di attacchi avversariali allo scopo di aumentare la robustezza dei modelli
- NON è l'applicazione di tecniche genetiche al deep learning

Tecniche discriminative

- Tecniche di classificazione che si focalizzano sulla definizione delle frontiere di decisione (decision boundaries)
- NON tecniche tipiche di unsupervised learning che tentano di separare i dati in cluster distinti
- NON tecniche che cercano di discriminare i dati in base alle diverse distribuzioni di probabilità delle varie classi
- NON tecniche che cercano di indentificare gli outliners all'interno del dataset
- NON sono tipicamente meno espressive delle tecniche generative
- NON si applicano per lo più in ambito di apprendimento non supervisionato
- NON cercano di determinare le distribuzioni di probabilità delle varie classi di dati

AlexNet

- È la prima rete convoluzionale PROFONDA vincitrice della ImageNet competition
- Realizzata nel 2012

Regressione Lineare

- NON cerca di determinare un iperpiano di separazione tra due categorie di dati
- Il problema di ottimizzazione ammette una soluzione in forma chiusa
- La funzione di loss è tipicamente una distanza quardatica tra i valori predetti e quelli osservati
- Cerca di stabilire una relazione tra i valori di una variabile di output e i valori di una o più features di input

Classificazione lineare

- Potrebbe non fornire risultati soddifacenti quando la classificazione dipende da un confronto tra le features
  - Se esiste una elevata correlazione tra le features NON importa

  - Se non tutte le features non sono rilevanti ai fini della classificazione NON importa
  - Se le features sono indipendenti tra loro, data la classe, NON importa

Regressione logistica

- I parametri del modello sono tipicamente calcolati mediante discesa del gradiente
- La predizione dipende dal bilanciamento dei dati di training rispetto alle classi
- I parametri del modello NON possono essere tipicamente calcolati in forma chiusa, mediante una formula esplicita
- Il calcolo della predizione si basa sulla loglikelihood dei dati di training, in quanto si tratta di una tecnica discriminativa
- Si basa su una combinazione lineare delle features di input
- La probabilità della predizione cresce se ci sia allontana dalla superficie tra le classi
- Nel caso di una classificazione binaria, la superficie di confine tra le classi è un iperpiano
- Potrebbe essere in difficoltà quando la classificazione dipende da un confronto tra le features
- Permette di associare una probabilità alla predizione della classe
- NON ci sono problemi quando non tutte le features di input sono rilevanti ai fini della classficazione
- NON ci sono problemi quando esiste una elevata correlazione tra le features
- NON ci sono problemi quando le feature sono indipendenti tra loro, data la classe

Regressione multinomiale

- Il peso con cui è valutata ogni feature è tipicamente diverso per ogni classe
- Il peso delle features indica la loro importanza ai fini della classificazione
- Per n features e m classi, il numero di parametri del modello è $n \cdot m + m$
- Per n features e m classi, il numero di parametri del modello cresce come $O(nm)$
- Per ogni input, NON È GARANTITO che esista almeno una classe con probabilità > 0.5
- (ERRATA) I pesi delle features sono sempre tutti positivi, i bias possono essere negativi

Derivata della funzione logistica

$$\frac{d}{dx}\sigma(x) = \sigma(x) \cdot (1 - \sigma(x))$$

- Tende a 0 quando $x \to -\infty$
- Hai il suo massimo in corrispondenza dello 0
- NON è una funzione simmetrica
- NON è una funzione monotona

Accuratezza, Precisione e Recall

- Accuratezza (Accuracy) = Istanze classificate correttamente

$$Accuratezza = \frac{TP + TN}{All}$$

- Precisione (Precision) = Precisione sui positivi

$$Precisione = \frac{TP}{TP + FP}$$

- Richiamo (Recall) = Percentuale dei positivi classificati come tali

$$Richiamo = \frac{TP}{TP + FN} = \frac{TP}{P}$$

Autoencoders

- Applicati per rimozione del rumore (denoising)
- Applicati per riduzione delle dimensioni (dimensionality reduction)
- Applicati per rilevamento di anomalie (anomaly detection)
- NON applicati per Segmentazione di Immagini (semantic segmentation)
- NON richiedono l'uso di livelli densi

- Encoder e decoder NON devono essere simmetrici
- La rappresentazione interna (latent space) prodotta dall'encoder ha solitamente una dimensione ridotta rispetto a quella di partenza
- NON è una rete che codifica sè stessa

Neuroni Artificiali

Definisce un semplice modello matematico che simula un neurone biologico

- Tipicamente calcola una combinazione lineare dei suoi input, seguita dalla applicazione di una funzione di attivazione NON lineare
- Il suo numero di parametri è lineare nel numero dei suoi input
- NON può apprendere qualunque funzione dei suoi input
- Può apprendere ANCHE funzioni lineari, ma non solo

Campo ricettivo di neuroni artificiali (receptive field)

Definisce la porzione dell'input che influenza l'attivazione di un determinato neurone

$$r_{i-1} = s_i \cdot r_i + (k_i - s_i)$$

- Esempio: Due layer Conv2D con stride 1, il primo con kernel 5x5 e il secondo con kernel 3x3
  - $r_2 = 1$
  - $r_1 = 1 \cdot r_2 + (3 - 1) = 3$
  - $r_0 = 1 \cdot r_1 + (5 - 1) = 7$
- Dipende quindi da
  - Profondità del layer a cui si trova il neurone
  - Dimensioni e stride dei kernel precedenti
- Aumenta rapidamente con l'attraversamento di livelli con downsampling

Transposed Convolutions

- Possono essere interpretate come convoluzioni normali con stride sub-unitario
- Sono prevalentemente utilizzate in architetture per Image-to-Image processing, come autoencoders o U-Nets
- Sono essenzialmente equivalenti alla applicazione di un livello di upsampling seguito da una convoluzione normale
- NON richiedono la trasposizione dell'input prima di calcolare la convoluzione del Kernel

Dimensione outupt in Conv2D

Nota: Le divisioni sono divisioni intere

$$Width_{new} = \frac{Width_{old} - Kernel_{width} + 2 \cdot Padding}{S} + 1$$

$$Height_{new} = \frac{Height_{old} - Kernel_{height} + 2 \cdot Padding}{S} + 1$$

$$Channel_{new} = Number\_of\_kernel$$

Stride

- Uno stride non unitario (>1) fa DIMINUIRE la dimensione spaziale (dell'output)

Numero di parametri in Conv2D

1 for bias

$$number\_parameters = out\_channels * (in\_channels * kernel\_height * kernel\_width + 1)$$

- Dipende quindi da dimensione spaziale del kernel e numero di canali in input e in output

Inception module

- Sfutta kernel di dimensione diversa
- Tende a ridurre il costo computazionale sfruttando convoluzioni unarie per diminuire il numero dei canali
- NON utilizza al proprio interno delle skip-connections per bypassare l'applicazione di parte dei kernel
- È un componente tipico della rete Inception-v3

Intersection over Union

- È frequentemente utilizzata come misura di similitudine tra bounding boxes
- Restituisce un valore nel range [0,1]
- È una funzione simmetrica dei suoi input
- È una metrica principalmente utilizzata nel campo della Object Detection

Transformers

- Hanno una tipica struttura encoder-decoder, ognuno formato da uno stack di sotto-componenti modulari
- Sono alla base delle reti della famiglia BERT e GPT
- Utilizzano pesantemente il meccanismo di attenzione
- NON Aggiungono ad ogni livello della rete un encoding posizionale per enfatizzare la posizione realtiva dei tokens

Funzione Softmax

- Permette di calcolare/Restituisce una distribuzione di probabilità sulle classi
- Produce valori compresi nell'intervallo $[0, 1]$
- Generalizza la funzione logistica al caso multiclasse
- Per un dato input, la somma dei suoi valori su tutte le classi è sempre 1
- (ERRATA) Per una data classe, la somma dei suoi valori su tutti gli input è sempre 1 (Attenzione all'ordine!)

Funzione ReLU

$ReLU(x) = x$ se $x > 0$, altrimenti 0

- Può essere usata per i layer convoluzionali
- Lei o le sue varianti son usate per i livelli interni delle reti neurali profonde
- La sua derivata è una funzione a gradino
- È una funzione monotona non decrescente

Funzione di MaxPooling

- Dato un tensore in input restituisce il valore massimo del tensore
- Solitamente si usa per fare pooling di un layer (diminuire dimensione da un layer all'altro)
- La sua derivata è uguale ad 1 in corrispondenza del massimo e 0 altrove

Struttura rete per classificazione di immagini

- Sequenza alternata di convoluzioni e downsampling, seguita da flattening e pochi livelli densi finali

Deep Features

- Features sintetizzate in modo automatico a partire da altre features

Generative adversarial network (GAN)

- Composte da Generatore e discriminatore
  - Allenati alternativamente
- Possono soffrire del fenomeno di "mode collapse", cioè la tendenza a focalizzare la generazione su un unico o pochi esempi

- NON è una rete che permette di generare attachci per un qualunque modello predittivo
- NON hanno una struttura encoder-decoder, simile a quella di un autoencoder
- NON basano il loro training su una funzione di loglikelihood relativa ai dati generati

Long-Short Term Memory Models (LSTM)

- Sono una particolare tipologia di Rete Ricorrente
- Utilizzati prevalentemente per elaborazione di sequenze di dati
  - NON per segmentazione di immagini mediche
  - NON per predirre traiettorie per agenti a guida autonoma
  - NON per elaborazione di immagini
- Utilizzano delle particolari porte (gates) per gestire l'evoluzione della cella di memmoria durante l'elaborazione di una sequenza di dati

Funzione di loss

- Rete neurale per classificazione di categorie MULTIPLE con softmax come attivazione finale tipicamente usa come loss la CATEGORICAL CROSSENTROPY
- Rete neurale per classificazione BINARIA con sigmoid come attivazione finale tipicamente usa come loss la BINARY CROSSENTROPY

U-Net

- È un componente tipico dei modelli generativi a diffusione
- Può essere usata per la rimozione del rumore (denoising) di immagini
- Il suo campo tipico di applicazione è la segmentazione semantica
- Il suo campo tipico di applicazione NON è la generazione musicale
- Il suo campo tipico di applicazione NON è la object detection
- Il suo campo tipico di applicazione NON è la natural language processing

Reti per classificazione di immagini

- Inception-v3
- VGG19
- ResNet
- NON la U-Net (usata invece per la segmentazione di immagini biomediche)

Optmizer in Tensorflow/Keras

Definisce l'algoritmo che calcola i gradienti della loss e aggiorna i pesi del modello

- NON contrasta l'overfitting
- NON aggiunge penalità ai pesi del layer su cui viene istanziato
- NON salva i migliori pesi del modello durante il processo di training

K-Means

Il suo obbiettivo è raggruppare i punti di un cluster attorno al loro centroide