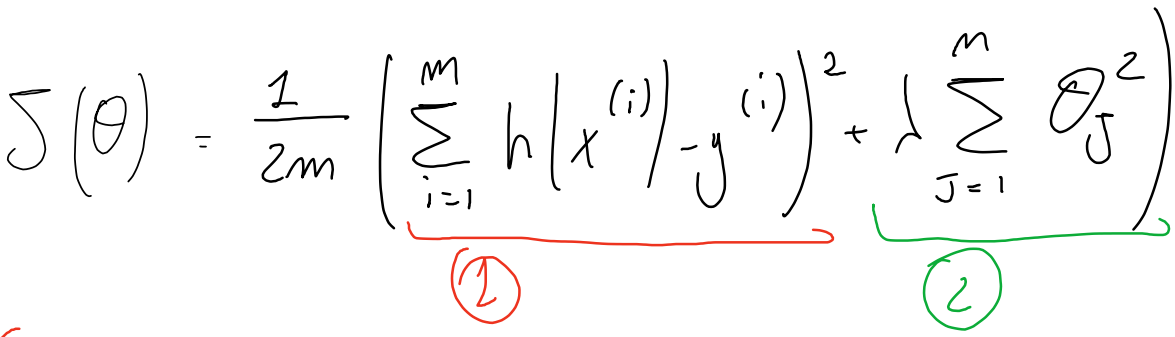
4. Come scegliere il prossimo ML system

mercoledì 30 ottobre 2024 11:04

Ipotizziamo di fare un debugging di un algoritmo di apprendimento, in particolare di una regressione lineare regolarizzata utilizzata per prevedere i prezzi delle case. Qui vengono esplorate alcune strategie per migliorare le prestazioni del modello, nel caso in cui i test mostrino errori molto elevati.



1 rappresenta la differenza tra i valori previsti dal modello h(x) e i valori reali y , sommati per tutti i punti di training.

2 Aggiunge una penalità in base ai parametri θ_j del modello, dove λ controlla l'intensità della regolarizzazione. Questo termine riduce l'overfitting impedendo ai coefficienti di diventare troppo grandi.

Strategie per Migliorare le Prestazioni

Se il modello ha errori molto elevati, si possono provare diverse tecniche per ridurli:

- Aumentare gli esempi di addestramento: può migliorare la generalizzazione, soprattutto se il modello soffre di varianza elevata.
- Ridurre il numero di feature: aiuta a ridurre la complessità del modello, utile se il modello soffre di overfitting.
- Aggiungere nuove feature: può aiutare se il modello è troppo semplice (sottofitting) e non riesce a catturare tutta la variabilità nei dati.
- Aggiungere feature polinomiali: aumenta la capacità del modello di adattarsi a relazioni più complesse tra le variabili.
- Modificare il valore di \lambda :
- Ridurre \lambda. se il modello è troppo rigido (sottofitting) a causa di una regolarizzazione eccessiva.
- Aumentare \lambda se il modello è troppo complesso (overfitting) e ha bisogno di essere semplificato.

ANALISI DIAGNOSTICA

Un'analisi diagnostica è un test progettato per dare informazioni su cosa stia funzionando o meno in un algoritmo di apprendimento e offrire indicazioni su come migliorare le sue prestazioni. Qundi si tratta di strumenti che aiutano a capire se l'algoritmo sta generalizzando bene, soffre di overfitting o underfitting, e come intervenire per correggere eventuali problemi.

Perché i Diagnostici sono Utili

Implementare diagnostici può richiedere tempo, ma rappresenta un investimento prezioso. Permettono di risparmiare tempo e risorse a lungo termine, poiché aiutano a evitare tentativi ed errori casuali nel miglioramento del modello. I diagnostici forniscono informazioni chiare su quali aspetti del modello possono essere migliorati e aiutano a scegliere le tecniche di ottimizzazione più efficaci.

Tipologie di Diagnostici

Ecco alcuni esempi di diagnostici comuni in machine learning:

1. Analisi del Bias e della Varianza:
   • Bias elevato indica che il modello è troppo semplice e non riesce a catturare le informazioni dei dati (sottofitting).
   • Varianza elevata suggerisce che il modello è troppo complesso e si adatta troppo strettamente ai dati di training (overfitting).
   • Con questo diagnostico, si può decidere se è necessario rendere il modello più complesso o più semplice.
2. Learning Curve (Curva di Apprendimento):
   • Grafica l'errore del modello in funzione del numero di esempi di addestramento.
   • Permette di capire se il problema è dovuto alla quantità di dati (es., se l'errore decresce lentamente, può essere utile raccogliere più dati).
3. Validazione della Regolarizzazione:
   • Testare diversi valori del parametro di regolarizzazione λ per capire l'impatto sull'overfitting.
   • Aiuta a determinare se l'algoritmo richiede più o meno regolarizzazione.
4. Errore di Addestramento vs Errore di Test:

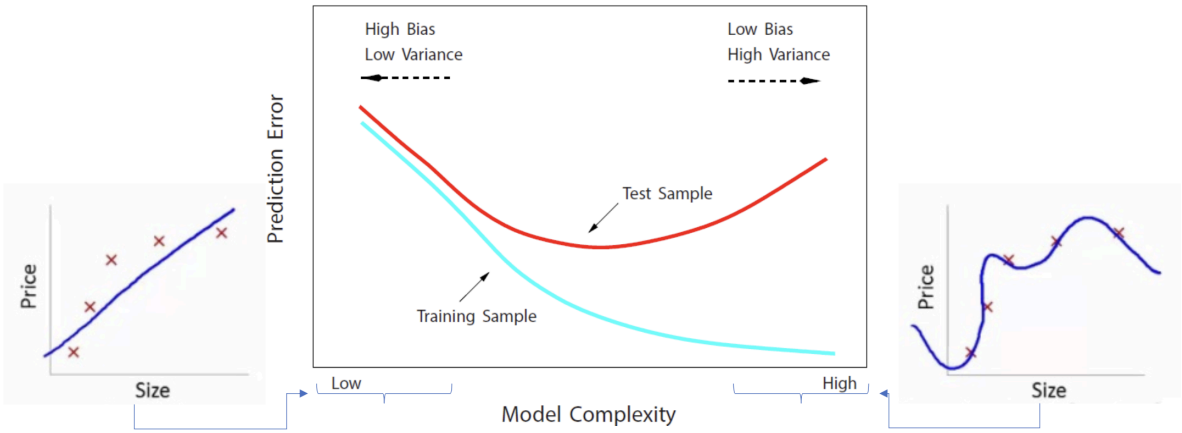
• Comparare l'errore sui dati di addestramento con quello sui dati di test.
• Se l'errore di test è molto maggiore rispetto a quello di addestramento, è segnale di overfitting.

Vantaggi dei Diagnostici

Implementare diagnostici in modo sistematico aiuta a prendere decisioni informate su come modificare il modello o i dati, evitando di fare tentativi casuali. Questo approccio permette di migliorare le prestazioni dell'algoritmo in modo più rapido ed efficiente, rendendo i diagnostici uno strumento prezioso nel processo di sviluppo del machine learning.

Diagnosing bias/variance

Let us recap the Bias variance behavior w.r.t. the complexity of the model



Bias e Varianza:

Bias Alto e Varianza Bassa (sul lato sinistro della complessità del modello): Quando il modello è troppo semplice, tende a sottostimare la relazione tra le variabili, portando a errori elevati. Questo fenomeno è noto come sottofitting.
Bias Basso e Varianza Alta (sul lato destro della complessità): Quando il modello è troppo complesso, si adatta troppo strettamente ai dati di addestramento, generando errori più elevati nei dati di test. Questo è chiamato overfitting.

Curva dell'Errore di Addestramento e Test:

La linea blu rappresenta l'errore sui dati di addestramento, mentre la linea rossa rappresenta l'errore sui dati di test.
Man mano che aumenta la complessità del modello, l'errore sui dati di addestramento diminuisce rapidamente, ma l'errore sui dati di test inizia ad aumentare dopo un certo punto, indicando overfitting.

Strategie per la Diagnosi
Analisi Bias-Varianza:

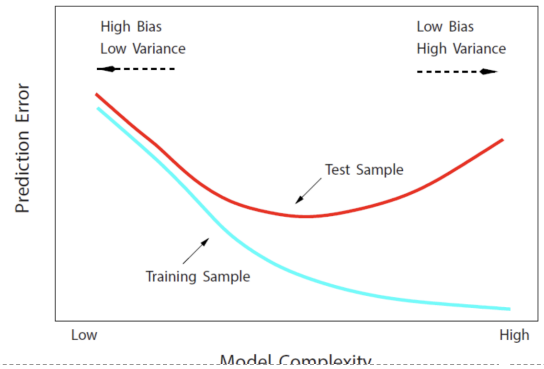
Se il modello ha un errore elevato sia nei dati di addestramento che in quelli di test, probabilmente ha un bias elevato (overfitting). Aumentare la complessità del modello potrebbe aiutare.
Se il modello ha un errore basso nei dati di addestramento ma elevato nei dati di test, allora soffre di varianza elevata (overfitting). Ridurre la complessità del modello può essere utile.
Modifiche e Ottimizzazione:

Aumentare i dati di addestramento: utile per ridurre la varianza.
Modificare la complessità del modello: ridurre o aumentare il numero di feature o aggiungere feature polinomiali a seconda del problema di underfitting o overfitting.
Regolarizzazione: scegliere il valore ottimale di λ per trovare un equilibrio tra bias e varianza.

Diagnosing bias/variance

Suppose your learning algorithm is performing less well than you were hoping. ($J_{cv}(\theta)$ or $J_{test}(\theta)$ is high) Is it a bias problem or a variance problem?

SINTOMI

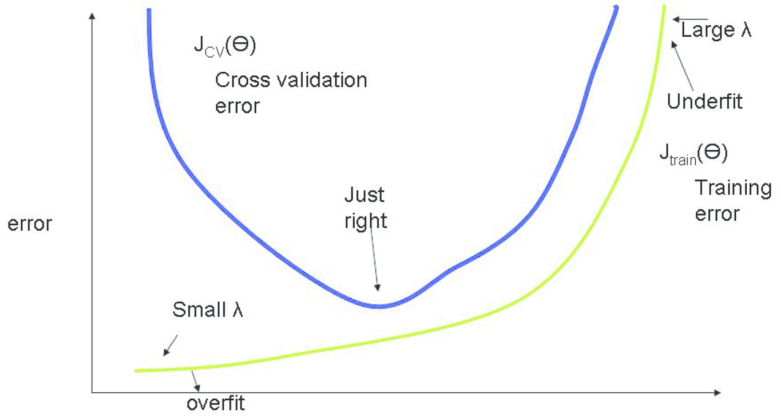


Bias (underfit):

$J_{train}(\theta)$ will be high
$J_{cv}(\theta) \approx J_{test}(\theta)$

Variance (overfit):

$J_{train}(\theta)$ will be low
$J_{cv}(\theta) \gg J_{train}(\theta)$



Cosa Sono le Learning Curves

Una learning curve è un grafico che mostra l'andamento dell'errore del modello in funzione del numero di esempi di addestramento (m). Serve principalmente per:

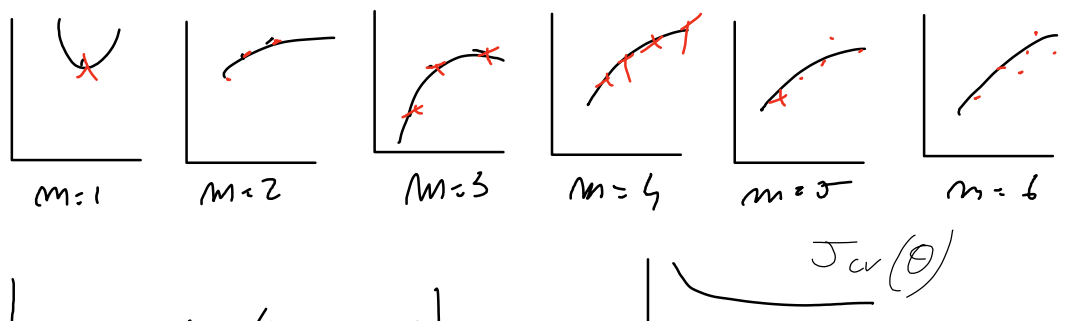
Verificare il modello: è utile per capire se il modello sta apprendendo correttamente dai dati.
Migliorare le prestazioni: permette di individuare problemi di sottofitting o overfitting.
Elementi del Grafico della Learning Curve
Nella learning curve, vengono tracciate due funzioni dell'errore rispetto alla dimensione del set di addestramento m:

Errore sul set di addestramento
Jtrain(θ): rappresenta l'errore commesso dal modello sui dati di addestramento. Di solito, questo errore diminuisce all'aumentare dei dati di addestramento.
Errore di cross-validazione (Jcv(θ)): indica l'errore del modello sui dati di validazione, quindi una stima di come il modello generalizza su dati non visti.

Come Funziona la Learning Curve
Per costruire una learning curve:Si inizia con un numero ridotto di esempi di addestramento (m=1), calcolando l'errore sui dati di addestramento e di cross-validazione.
Man mano si aumentano i dati di addestramento (m=2, m=3, ecc.) fino a utilizzare l'intero set, osservando come variano gli errori.

Interpretazione della Learning Curves
Underfitting: se l'errore di addestramento e di cross-validazione sono entrambi elevati, il modello è troppo semplice (alto bias).
Overfitting: se l'errore di addestramento è basso, ma quello di cross-validazione è alto, il modello è troppo complesso e si adatta troppo ai dati di addestramento (alta varianza).

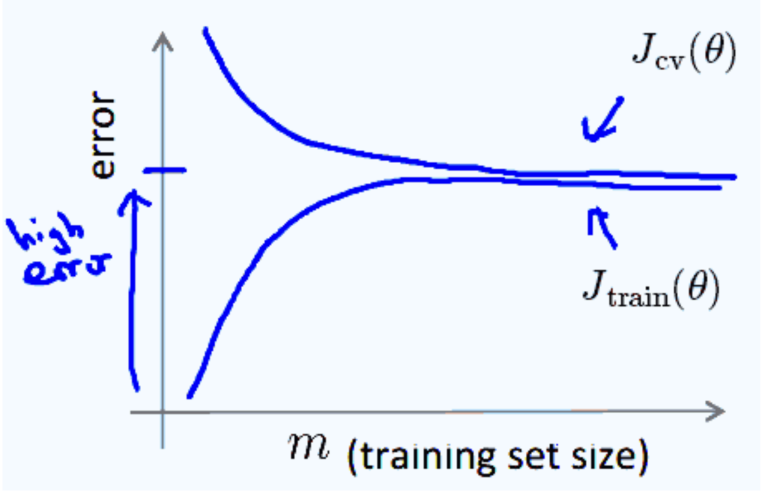
Supponiamo di avere: $h_\theta(x) = \theta_0 + \theta_1 x_1 + \theta_2 x^2 +$



Learning curve ottenute

ESEMPIO
$h_\theta(x) = \theta_0 + \theta_1 x_1$ ADATTATO da retta a tale polinomio

Aumentando la dimensione del dataset (indicato come m), la retta di regressione non cambia significativamente e continua a non rappresentare adeguatamente i dati, segnalando che il modello è troppo semplice.



Curve di apprendimento (learning curves):

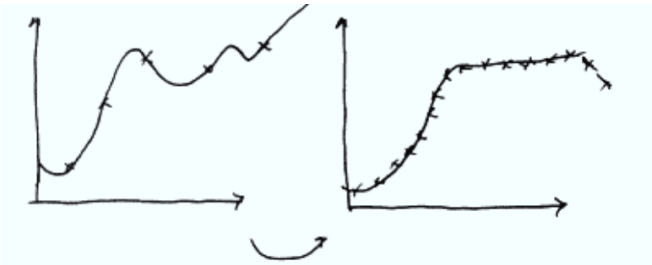
- La slide mostra le curve di apprendimento per errore di allenamento J_{TRAIN} e errore di validazione (J_{VAL}) rispetto alla dimensione del set di dati di allenamento.
- In caso di high bias, entrambe le curve tendono a un valore di errore elevato, e l'errore di validazione si avvicina all'errore di allenamento . Questo indica che il modello non riesce a catturare la complessità dei dati.

IN DEFINITIVA: Se un algoritmo di apprendimento ha un problema di high bias, aumentare il numero di esempi nel dataset non migliorerà le sue prestazioni. Questo perché l'errore è principalmente causato dalla mancanza di complessità del modello, non dalla scarsità di dati.

DIAGNOSE HIGH VARIANCE (OVERFITTING)

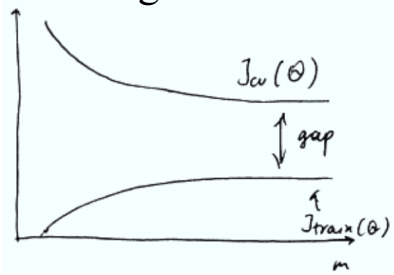
42 ORA ABBIAMO QUESTO POLINOMIO

$h_\theta(x) = \theta_0 + \theta_1 x + \theta_2 x^2 + \dots \theta_{100} x^{100}$



All’inizio, il modello è altamente sovra-allenato e si adatta perfettamente ai dati di addestramento, ma questo significa che generalizza male sui nuovi dati.
Quando la dimensione del set di dati ((θ) aumenta gradualmente, dato che diventa più difficile adattarsi a un set di dati più grande.

L'errore di validazione diminuisce lentamente con l’aumento dei dati, ma persiste un ampio divario tra *J*train e *J*cv
Questo divario è un sintomo di overfitting e indica che il modello ha un’alta varianza.
Soluzione proposta: L'immagine suggerisce che, se un algoritmo mostra alta varianza, una possibile soluzione è aggiungere molti più dati per cercare di ridurre l'overfitting e migliorare la capacità di generalizzazione del modello.



+ GAP + VARIANZA

LA SITUAZIONE CHE DOBBIAMO CAPIRE DATO UN MODELLO PER PRONOSTICARE IL PREZZO O LA VALUTA DELLE CASE, ESSO DA ERRORI PER UNO DEI DUE PROBLEMI O PER IL BIAS O PER UN ALTA VARIANZA.

Debugging a learning algorithm

$$J(\theta) = \frac{1}{2m}\left(\sum_{i=1}^{m}\left(h\left(x^{(i)}-y^{(i)}\right)^2 + \lambda\sum_{j=1}^{n}\theta_j^2\right)\right.$$

POSSIBILI RISOLUZIONI PER MIGLIORARE I MODELLI

Get more training examples (Aggiungere più esempi di addestramento):Utile per ridurre high variance. Con più dati, il modello si allena su una varietà maggiore, riducendo il rischio di sovra-allenamento.
Try smaller sets of features (Provare un set di caratteristiche più piccolo):Utile per ridurre high variance. Utilizzare meno caratteristiche semplifica il modello, evitando che si adatti troppo ai dettagli specifici del set di addestramento.
Try getting additional features (Aggiungere nuove caratteristiche):Utile per ridurre high bias. Nuove caratteristiche possono aiutare il modello a rappresentare meglio le relazioni nei dati, aumentando la sua capacità di adattarsi correttamente ai dati di addestramento.
Try adding polynomial features (Aggiungere caratteristiche polinomiali):Utile per ridurre high bias. Caratteristiche polinomiali rendono il modello più complesso e capace di catturare pattern non lineari, migliorando l'accuratezza su dati complessi.
Diminuire λ:Utile per ridurre high bias. Ridurre il parametro di regolarizzazione
λ rende il modello meno rigido, permettendo di adattarsi meglio ai dati.
(Aumentare λ) Utile per ridurre high variance. Aumentare λ introduce più regolarizzazione, rendendo il modello meno complesso e riducendo il rischio di adattamento eccessivo ai dati di addestramento.