20231003 - ML

Linear Regression

y = m \* x + b (modelo linear)

m = steepness

b = distance(error)

Mean square error function

LLM nesta altura não é preciso para estudar

Nas proximas aulas o chatGPT as vistas do utilizador é somente uma black box, vamos a ver em termos algebricos.

“Algoritmo de mecanismo de atenção”: antes cada palavra era apenas uma, mas agora o mecanismo de atenção, associam-se com o sentido de cada um.

Variavel dependente: Y

Variável independente: X

Instancias: elementos (cada instancia é um ponto) = uma linha

Caracteristicas: colunas

Dimensionalidade:

Densidade alta: qto maior a dimensionalidade maior deve ser a densidade

Quando a taxa de erro não está muito boa:

Etiquetas finitas: classificação

Etiquetas infinitas: regressão

**Occam’s razor**: Descreva de que forma se relaciona com a maldição da dimensionalidade

Complexo: numero de parametros

Os complexos ao competir com os simples dependem com a quantidade de instancias

Seria justo se seriam a mesma quantidade de parametros

O modelo x resultou melhor, mas não sabem qual o motivo. Quais os parametros que cada modelo está a usar

ℎ𝜃 𝑥 = 𝜃1. 𝑥 + 𝜃2

Parametros = 𝜃1, 𝜃2

Mais parâmetros mais curvas

Uma reta não consegue atingir todos os pontos porque as instancias não são expressas em uma reta.

Overfiting para o treino é bom, para a produção

O valor dos parametros são os parafusos que quero ajustar

Nós não conseguimos fazer otimização independente

Loss function diz se os parametros estão bem ajustados

Função J: função custo

J: é decisiva, o som do motor

Why ?: Por conta da Derivada

Tenho modelo 2 parametros, eu crio a função objetivo e diz se estou longe ou perto do objetivo

Algoritmo do gradiente: é eu ter uma função e faz a derivada parciais em relação aos parametros para calcular maximo e minimos das funções

O minimo da função J é o melhor do modelo.

Parametros =

𝜃1= 2

𝜃2= 3,4

Derivadas parciais de 𝜃1 e 𝜃2: vai dar a partir de um sitio a direção até chegar as derivadas parciais chegar a zero 0 que é o sitio ideal

Taxa de aprendizagem diz se os saltos são pequeninos ou grandes

Derivadas da função de custo

𝜃1. 𝑥 + 𝜃2 - yi

a + b = a’

Para cada configuração dos dois 𝜃 tem a distancia dos erros

𝜽(t+1)= 𝜽(t)- Δ ∫ / ∫𝜽 \* J(𝜽)

Forma fechada: quando as função de custo começam a complicar não existe forma fechada

Funciona bem com poucos parametros e é muito custosa porque inverte matrix

Toda a gente acaba por ir para o metodo gradiente

Estamos a misturar os 𝜽 a esquerda

Não sei onde é o minimo local

Modelo de ML é um conjunto de parametros relacionar input e output

Quando é que se para o modelo ?

parei ao fim de 100 interações.

Paciencia é um parametro de aumento de erro ou melhoria