Data Science: Deep Learning Prerequisites – Linear Reg in Python Notes

**Section 2: 1-D Linear Regression Theory and Code**

**Linear regression in python**

this isa lecture

ohm’s law V=IR is a linear equation

V=vol tate

I=current

R=resistance

Current (I) is the independent variable

given a set of inputs and outputs you can plot on graph

creates an almost perfect line

always some element of error

drawing the line of best fit

try to calculate the slope of the lin

R=slope=V/I

linear can apply to anything

blood pressure versus age

blood pressure vs weight

All data is the same

there is a systematic way to find line of best fitSimple linear regression

this is yi=axi+b

y*i*is expected y

xi is the input ex

we want the line of best fit to be as close to possible (minimizing error)

finding differences does not work

errors of +5 and -5 = 0

Step 1: need to find the “sum of squared errors”

Error (E)=summation (yi-yhati)2

Step 2: take the derivive of Error function to minimize the result

example

E=0.5t2-t

E`=t-1

so by setting derivative equal to 0

0=t-1

t=1

in our problem

Error (E)=summation (yi-yhati)2

So pluggin in our line of best fit function

E=summation(yi-(axi+b))2 or E=summation(yi-axi-b)2

yi and xi are given due to datapoints

have to take partial derivatives since multiple variables <https://www.youtube.com/watch?v=SbfRDBmyAMI>

<https://www.google.com/search?q=finding+derivatives+polynomials&ie=utf-8&oe=utf-8&client=firefox-b-1-ab#kpvalbx=1>

<https://www.youtube.com/watch?v=-_8DFxTl0Ls>

<http://polisci.msu.edu/jacoby/msu/ppl801/handouts/Handout,%20Summations,%202010.pdf>

see notes sheet