Maths ML After Statistics (I)

Slop Maths is fun.

eq of a line

$y = mx + c$

The major diff

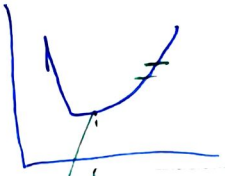
$\text{Slop} \Rightarrow \frac{\text{change in } y}{\text{change in } x} = \frac{y_2 - y_1}{x_2 - x_1}$

gradient:

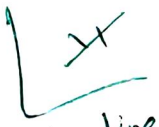
↳ how steep the line is

$\frac{a}{y_1 - y} = \frac{3}{0} = \infty$? not define

if $y = x^2$ → the how to calculate the slop of the in line is easy



→ because there not same perpersion



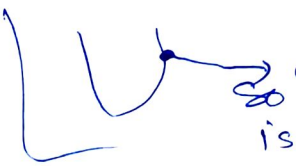
in line slope had constant rate

can we do $\frac{y_2 - y_1}{x_2 - x_1}$ ×

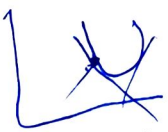
↳ in this case slope is Vary

↓↓

Steepness varies the (so slop of 2 point not same)



so slope w.r.t 1 point that is called → instantaneous rate of change w.r.t y (called)



tangent / slope
↳ is a line that touches a curve at one point

then How Calculate

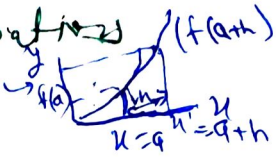
$\frac{y_2 - y_1}{x_2 - x_1}$ → $\frac{dy}{dx} = f'(x) = \lim_{h \to 0} \frac{f(a+h) - f(a)}{h}$ → first principle

Slope or tangent

instantaneous change rate of change in y w.r.t x)

this is called differentiation

h tends to 0 not 0



Recap

① $\frac{dx^n}{dx} = nx^{n-1}$

① $\frac{d(1)}{dx}$ 0 $\frac{d(x)}{dx} = 1x^{1-1} = 1x^0 = 1$

upward is slope +ve or -ve

So line in upword called positive

Chain rule

$$\frac{d\,f(g(x))}{dx} = f'g(x) \cdot g'(x)$$

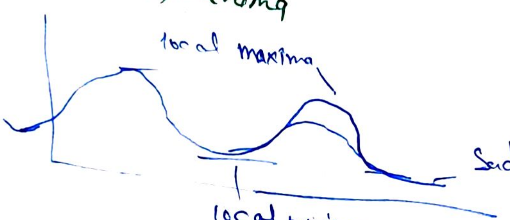
eg $(\sin(x^2))$ eg $\cos x^2 \cdot 2x$

$f(\ + g(x))$ $= 2x \cos x^2$

$f(g(x))$

Use Case → in gradient descent (for ML optimization techniques)

Maxima / minima



Slop = 0 (flatten/s

In case of maxima / minim

$\frac{dy}{dx} = 0$ flatten

Physics

Q A trajectory of ball is followed by $y = 4 + 10t - 6t^2$

what is max height that ball can reach?

we know that maxi heigh & min depth slop is 0 we know so we find the

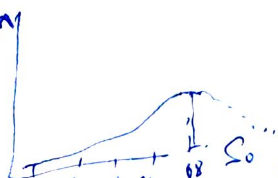
→ So find slope

$$\frac{dy}{dt} = \frac{d}{dt}(4 + 10t - 6t^2) = 0 + 10 - 12t$$
$$= 10 - 12t$$

So = Slop 0 at max/min

$10 - 12t = 0 \Rightarrow t = \frac{10}{12} = 0.83$

at 0.83sec, it will reach max ht



So height is $= 4 + 10t - 6t^2$

$= 4 + 10 \times 0.83 - 6 \times (0.83)^2 \Rightarrow 8.166$ m at 0.83

why you say max it may be min

Slop = 0

So in this do second order derivative

Maths ③ → Stats
→ tool kits
→ EDA

Do second order derivative

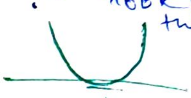
→ 10 − 12t

→ When a fn slope is 0 at x and the second order derivative at x is :-
- → less than 0, it is local maximum
- → greater than 0, it local minima
- → equal to 0, you cannot say any thing

$\frac{d}{dt}(10-12t) = -12$ → +ve

$\frac{dy}{dx} < 0$ → it means $\frac{dy}{dx} = 0$ is of local maxima.

optimization funct look like this



min

because we need minima error in our model

18 April

→ Maths → behind ML → for intuition ① ✓
↳ implementation ② ✓
→ maths ③ little bit

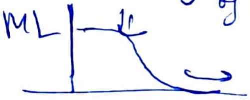
→ Murphy Book Stats

→ No fear from maths

→ Be punctual

→ Intro ML
Terminology
Pre-process / feature Engineering
→ Matrices

→ Learning curve of ML

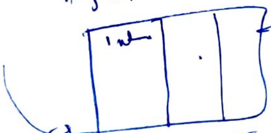


Show examples / motivation of ML
→ Cadbury Ads example
- ↳ Alexa
- ↳ Automation
- ↳ Whether forecast
- ↳ GPT, Bard, LLM (stats MODL → NLP → NLP) this is process

→ No student is bad Teacher is bad
how I takes
① ask Question
② Practice / Practice & Practice

mail example

Ajay algorithm



learn these thing
this is data

Ajay → promotion, discount, offers } → Promotion
↳ Fb, tinder, bu, Snap → Social tab

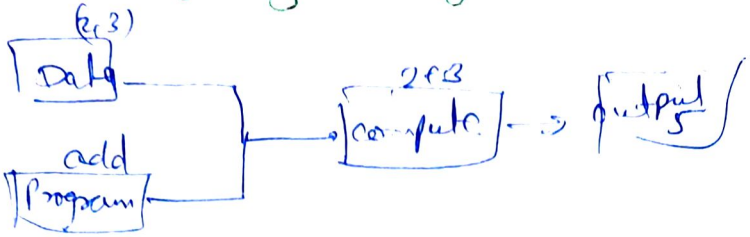
type of People
Not doing (is no) / methodology → Success fun

Q What is Machine learning?
✓ Machine learns pattern from the data and tries to replicate the same in future.

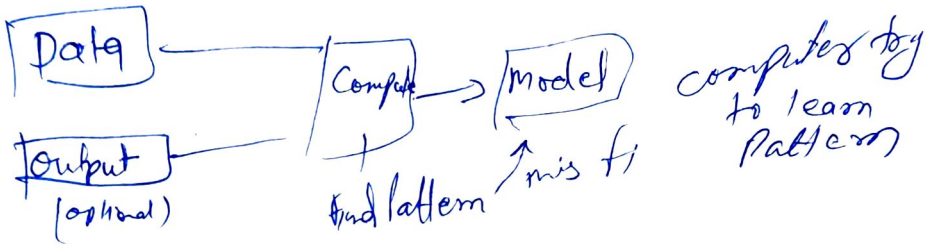
Arthur Samuel 1959 → The subfield of CS that gives computers the ability to learn without explicitly programmed

• tradition Programming Paradigm

⇒ Traditional Programming Paradigm



⇒ ML Programming Paradigm



Tom Mitchell (1997)
Geoffrey Hinton
Andrew Ng

1995-9
1997 → AI
1997
2011
2016
2017
2017

Linear Regression
Logistic Regression
} old

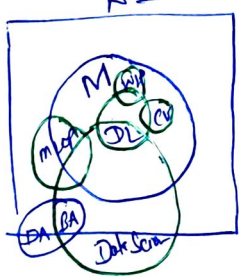
NVIDIA
AMD
Synopsis

ML
hardware
Processor
Software

1997 —— 2019
develop
Processor

AI Vs ML Vs DL Vs DS

→ Introduction to ML



→ types of → SL

Supervised

Historical data

Predict Price of House

feature targets
labels

Regression
y is Continuous
ex. Price

Classification
y is discrete (finite)
spam/ham

USL
No Supervision
y is not present
No Historical data
Then Can you find trends in that data

Semi-supervised learning
Combination
→ So either you drop labels & do SL

ex: netflix recommendation

Reinforcement

self driving Car

⇒ Test the model on some other data
↓
take this model
to test

we do ← so Time/Money take

Train test split

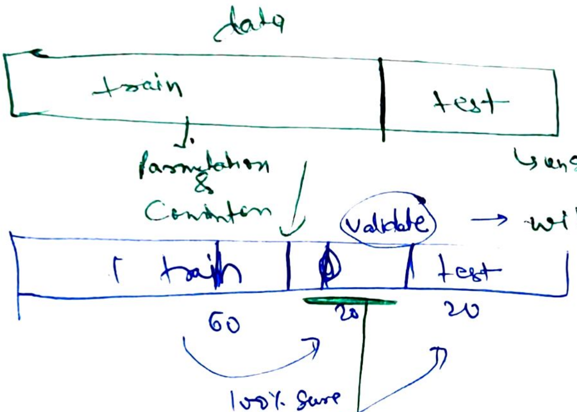
Train } 4:1
Test { ↳ representation of unseen data
→ used to test model accuracy

⇒ What ratio should devide?
① No any ideal ans / No any rule of

- 50:50 → × why (more data to learn)
- big data (1m rows) ← 70:30, 80:20
- 90:10 → small data (< 1m rows)

train > test

DS/M → appear for interview; mock interview practice



validate → will act as test while training

• data leakage

① Why Validation is need
↓
Hyperparameters tuning

ex: Batsman Virat Kholi → practice (trained, retrain for)

Coach R.D ↓ tuned ↓ Hyperparameter tuning

Parameter → Mathematical relationships

• Stude Stude ① NCERT training ② Sample Paper Tuning ③ Board Exams ‖ test of Model

Data leakage
Ex: board exam papers are given to students

you have passed test information to the training of the model ‖ model will perform well on both ⇐ deploy (use) ⇐ Model perform fail ⇐ losses of huge (Time/Money)

• Cross Validation

ML framework

f is set of rules
- a predefined process
- set of tools
- guidelines
- Standarized process

• area of house | room | price (y)

↳ 80% → training of mode → understanding the pattern → Mathematical relationship

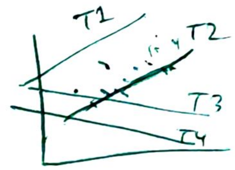
20%.

$$y = f(x)$$

y → output; x → input; f → predictive functions

→ **Model training** → Given the data estimate the prediction function to minimizing error
- prediction function ↓ ML algos
- minimizing error → Evaluation matrics

① If you learn from Training data then apply Ytrain prediction



→ Overfitting & Underfitting

train → Model(train) → Accuracy ↑ 95%.

test → Model(test) ⇒ Accuracy is low 60%.

↳ Model perform well on train data but worse on test data

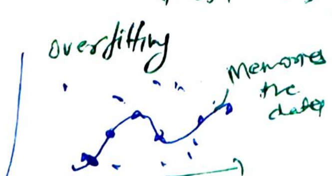
Vision sir (100)

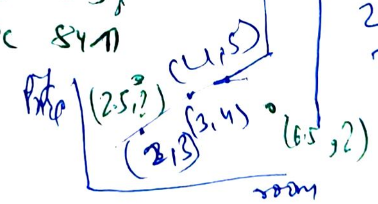
Tony Stark → memorised 100
Vision → understand the Concept

Under → Train → accuracy ↓ 80%
test → accuracy 45↓

Best Generalization Model

Train Acc. ↑ 85%
test → Acc 84↑





y = x + 1

No room	Price
2	3
3	4
4	5

Why?? → will you be able to predict the Price

↑ They don't fall in the Pattern

Generalize Model

How to identify ??
Model is Underfitting → not well while training
Overfitting → train is high ↑ test ↓
diff more than 5%
Generalized model → train ↑ test ↑
diff ≃5% range

Pre assumption
Bias Variance

* training error is also known as bias
High training error means high Bias
* testing error is also known as Variance.
High testing error means high Variance

Underfitting :- High
Train → Accuracy ↓ or High training error. → High Bias
test → Acc ↓ → High testing error → High Variance

→ Underfitting → High Bias, High Variance

Train Validate for retrain to imp → use Validation data not test data

Bias ↓ assumption is less

Overfitting → (low bias, High Variance)
You assumed many

Variance

Why Variance is analogous to High testing error?
→ Test data varied from the Pattern, that why high testing error

Train → trained → Accuracy ↑↑ → mean low bias (low Train error)
Test → tested → Accuracy (High test error) → High Variance

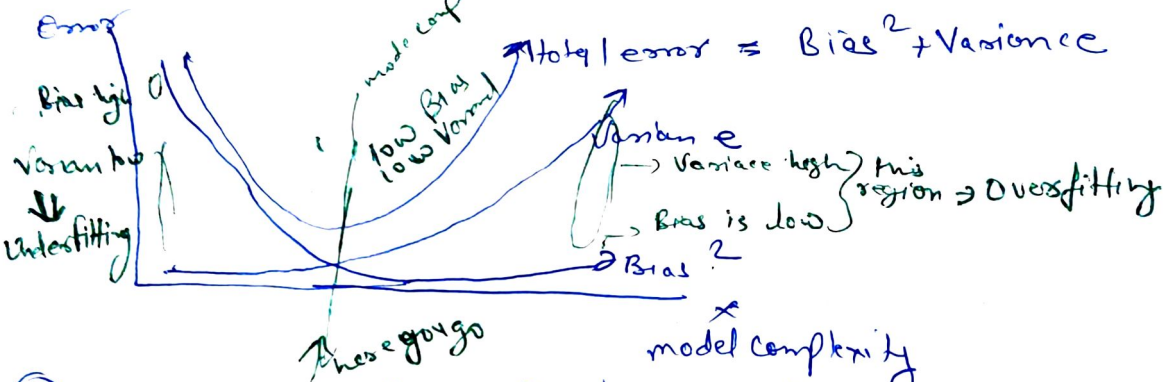
Generalised Model : (low bias, low Variance) → train acc ↑ (low bias) → low error in train
test acc ↑ (low Variance) → low error in test

Variance is spread

bias → how close to the target you are

→ Bias-Variance Tradeoff :

it describe the relationship b/w a model's complexity the accuracy of its predictions and how well it can make prediction on previously unseen data that were not used to train the model.



(1) where you want to go (know)

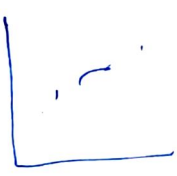
→ in order to reduce the Variance (induce high Bias.)
↓↓
simply inject the bias (increase bias) (means drop unnecessary feature)

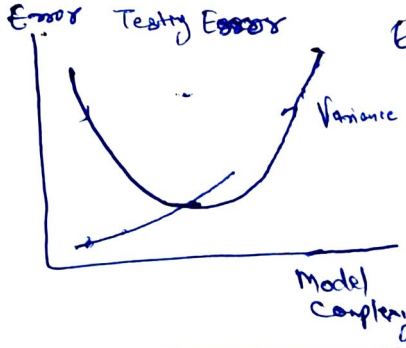
- Selete only relevent features
- Dont learn Noise / " " outliers
- Use some algorithm which is not overfit

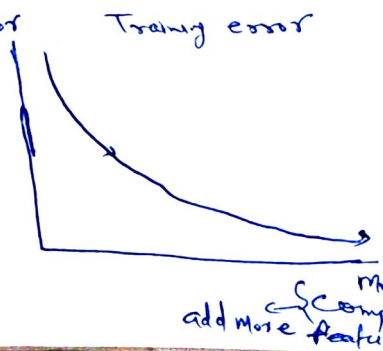
* In order to reduce Variance (reduce the testy error)
So introduce some bias in trainy of model by selecting relevent features of high Alogs

* In order to reduce high bias → Use some more datapoint, do more feature engineering
Or use some other Algorithms (last)

PCA (in ML)
* Vectors







(9)

Vectors

$5\hat{i}+4\hat{j}$ → $w^T x + b = 0$

$5\hat{i}+4\hat{j}$ → $\begin{bmatrix}5\\4\end{bmatrix}$

$OA^2 = OB^2 + AB$

Same my

These can be multiple vectors in space of same magnitude & direction, the origin is different.

Operations On Vectors

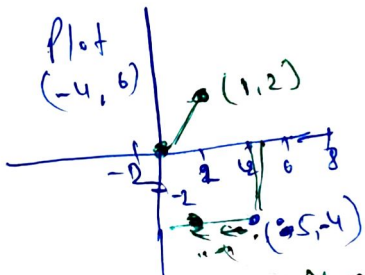
represent R^2 → 2D space C.

Real numbs

$\hat{i} = \begin{bmatrix}5\\-4\end{bmatrix}$; $\hat{j} = \begin{bmatrix}-4\\6\end{bmatrix}$

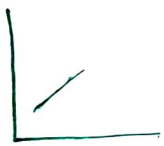
$i, j \in R^2$

⇒ Add ⇒ $i+j = \begin{bmatrix}1\\2\end{bmatrix}$ $\begin{bmatrix}5+(-4)\\(-4)+6\end{bmatrix}$

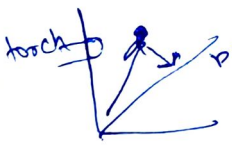


x → 5+(-4)
y → -4 → 6

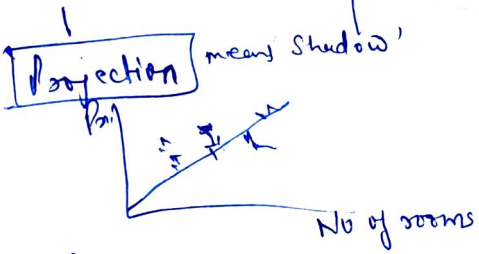
vecor is point



Projection of this is



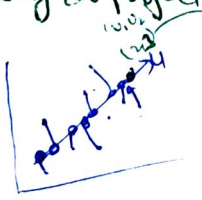
Projection means shadow



Project

Projection of a data point will be smaller than best fit line ~~that~~ that in same direction

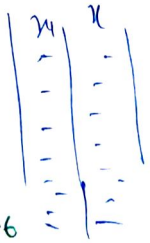
(5) Why do Projection



$\hat{u}_0 = \begin{bmatrix}u_1\\u_2\end{bmatrix}$ & 8 magnitude

$2 + 8 = 10$

10 mill 100 cols

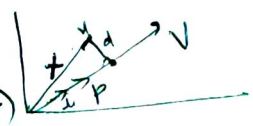


8 Data point = 16 cor

2 cols = 16 coordin

Can we all datapoint represent at u

How we choose $u = \frac{v}{\|v\|}$ (unit vector (magnitude is 1))



according to vector Rule

$x = p + d$

$d = x - p$ ⇒ $p + ku$

$d = x - ku$

Then dot product will be 0 $a \cdot b = ab\cos\theta$

$\cos 90 = 0$

$\Rightarrow 0$

$p \cdot d = 0$

$k4 \quad x(x^2 - k4)$

$\Rightarrow kx4 - k^2 4 \cdot 4 = 0$

$= k(x4 - k(4 \cdot 4))$ magnitude of $4 = 1$

$x4 - k4 \cdot 4 = 0$ $4 \cdot 4 = 1$

$k = x4$

$\hat{m}$ $\frac{v}{\|v\|}$ unit

$\hat{m}$ East

$\hat{m}$

$\Rightarrow$ East unit vector

Linear Algebra

What is linear algebra ?

& use

How to do ?

$3x + 2y = 8$

$5x + 4y = 18$

$\begin{bmatrix} 3 & 2 \\ 5 & 4 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} 8 \\ 18 \end{bmatrix}$

Matrix Addition

$\begin{bmatrix} 2 & 2 \\ 4 & 1 \end{bmatrix} + \begin{bmatrix} 2 & 4 \\ 2 & 3 \end{bmatrix}$

Matrix Multiplication → not index Multiplication

A = 3×4

B = 4×3

if $m \times \underbrace{n}_{\text{cols}} \quad \underbrace{n}_{\text{rows}} \times p$ (same) $\Rightarrow$ Result is $m \times p$

$\begin{bmatrix} 2 & 2 \\ 4 & 1 \end{bmatrix} \times \begin{bmatrix} 2 & 3 \\ 1 & 4 \end{bmatrix} = \begin{bmatrix} 2+2 & 2+1 \end{bmatrix}$

a ⊙ B

- Transpose - $A = \begin{bmatrix} 2 & 2 \\ 4 & 1 \end{bmatrix} \Rightarrow A^T = \begin{bmatrix} 2 & 4 \\ 2 & 1 \end{bmatrix}$

use in PCA

* Determinant of Matrix

in Python (det(A))

↳ $A = \begin{bmatrix} a & b & c \\ d & e & f \\ g & h & i \end{bmatrix}$

$Det(A) = a \begin{vmatrix} e & f \\ h & i \end{vmatrix} - b \begin{vmatrix} d & f \\ g & i \end{vmatrix} + c \begin{vmatrix} d & e \\ g & h \end{vmatrix}$

$\Rightarrow aei + bfg + cdh - afh - bdi - ceg$

$a(ei - fh)$

* inverse matrix

↳ $A \cdot A^{-1} = I$ (identity)

↳ for 2×2 (square matrix only) 2×2, 3×3, 4×4

$$\rightarrow Inv(A) = A^{-1} = \frac{1}{ad-bc}\begin{bmatrix} d & -b \\ -c & a \end{bmatrix}$$

→ ~~Adjoint of A~~ Adjoint of A

2 → 1/2
3 → 1/2

10m up & again come to same

→ for order 3×3

Work done is 0 but distance 20m

↳ eg: $A = \begin{bmatrix} a & b & c \\ d & e & f \\ g & h & i \end{bmatrix}$

order 3 or above

• the $inv(A) = \frac{1}{det(A)}$

$$inv(A) = \frac{1}{detA}\begin{vmatrix} \begin{vmatrix} e & f \\ h & i \end{vmatrix} & \begin{vmatrix} d & f \\ g & i \end{vmatrix} & \begin{vmatrix} d & e \\ g & h \end{vmatrix} \\ \begin{vmatrix} b & c \\ h & i \end{vmatrix} & \begin{vmatrix} a & c \\ g & i \end{vmatrix} & \begin{vmatrix} a & b \\ g & h \end{vmatrix} \\ \begin{vmatrix} b & c \\ e & f \end{vmatrix} & \begin{vmatrix} a & c \\ d & f \end{vmatrix} & \begin{vmatrix} a & b \\ d & e \end{vmatrix} \end{vmatrix}$$

Q $A = \begin{bmatrix} 4 & 7 \\ 2 & 6 \end{bmatrix} = A^{-1}$

$$= \frac{1}{4\times6 - 7\times2}\begin{bmatrix} 6 & -7 \\ -2 & 4 \end{bmatrix}$$

$$= \frac{1}{10}\begin{bmatrix} 6 & -7 \\ -2 & 4 \end{bmatrix}$$

$$A^{-1} = \begin{bmatrix} 0.6 & -0.7 \\ -0.2 & 0.4 \end{bmatrix}$$

order matter

$$\Rightarrow A \cdot A^{-1} = I \Rightarrow \begin{bmatrix} 4 & 7 \\ 2 & 6 \end{bmatrix} \cdot \begin{bmatrix} 0.6 & -0.7 \\ -0.2 & 0.4 \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

Relationship

Vector (vector) — convert into matrix → Matrix

• $v_1 = 3x + 4y \rightarrow [3 \cdot 4]$

$v_2 = x + 3y \rightarrow [1 \cdot 3]$

Z score → standardizes (using matrix)

scale up

scale down

$$\begin{bmatrix} x_1 \\ y_1 \end{bmatrix} = \begin{bmatrix} x \\ y \end{bmatrix}\begin{bmatrix} s_x & 0 \\ 0 & s_y \end{bmatrix}$$

Rotation :

$$\begin{bmatrix} x_1 \\ y_1 \end{bmatrix} = \begin{bmatrix} x \\ y \end{bmatrix}\begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix}$$

→ decomposition matrix

$$\begin{bmatrix} 1 & 0 & 0 & 2 \\ 0 & 1 & 0 & -1 \\ 0 & 0 & 1 & 1 \end{bmatrix}$$

★ Vector to matrix calculation

non-linear equation

$$2x + y - 2 = 2$$
$$x + 3y + 2z = 1$$
$$x + y + z = 2$$

$$\begin{bmatrix} 2 & 1 & -1 & 2 \\ 1 & 3 & 2 & 1 \\ 1 & 1 & 1 & 2 \end{bmatrix}$$

→ 2 approach

(1) Row echleon methods or Column echleon methods

(2) inverse method

Variance ✓
Bias ✓

CRISP DM framework

Cross industry standard process for Data Mining

Today
- Data Preparation
- Feature Engineering
- technique
→ Simple linear Regression
→ eigen Value ∴ Vector
→

(Q) ML → Can you predict this item demand for next 6 month

- 1st do POC (Proof of concept)

1st do EDA → ∴ → [univariate, bivariate, MultiVariant]
(Obj, Numeric)

① Data Preparation

① Missing Value treate
② Duplicates vate
③ Outliers → Class imbalance
④ feature engineering
  ① Select relevant feature
  ② Create new feature → speed, Dobn, new acc
  ③ Modify the existing feature

Age	
15 yr	years remain
16 year	
15	

⑤ Data Encoding
↳ String → to num → one hot

① Missing Values.

① Missing at completely Random
② MNR (Missing not at random) → Salary

↳ Deals
① Connect buss
② missing Val < 1% (& data large) so drop
③ Small data → Do Imputation
  - Numerical
    - outliers treatments is done → mean
    - If is not do → median
  - Categorical → Mode

④ OR Constant
↳ (Replace with 0) (Marks → Absent → So 0) or -1 OR No, that is not possible in that column

⑤ Create new columns flages

Val	Mg	N ob
a	20	0
b	0	0
c	3	1
d	4	0

⑥ if Missing Value in Column > 30% drop the Column, → Add Noise → unwanted data which is not naturally occurs

Interpolations — If time series data → impute Previous / next / Avg of Prev

+ Explain the ML Pipline
- Explain Lifecycle of ML

⇒ Data Interpolation

- It is a process of estimating data in a range. or getting Unknown Values from known values.

⇒ 3 way → filling the data using same trend with help of interpolation

1. Linear Interpolation
2. Cubic
→ Polynomial

Can we create some Syntatic data to fill this line

x = [1 2 3 4 5]
y = [1, 3, 5, 7, 9]
(linear) interpolate scatter

```
x_new = np.linespace(1,5,10)
y_interp = np.interp(x_new, x,y)
⇒ Scatter (x_new, y_interp)
```

Cubic

np. OR use.

```
from scipy.interpolate import interp1d
F = interp1d(x,y, kind='cubic')
x_new = np.linespace(1,5,10)
y_interp = f(x_new)
```

Use Case → time series
↳ ① missing data → for analysis of data ↳ Based on trend of data find
② when not enough data ↳ your Airplane Crash

Linear $ax + b = 0$
quad $ax^2 + bx + c = 0$
cub $ax^3 + bx^2 + cx + d = 0$ — Polynomial
$a, b, c \neq 0$

```
P = np.polyfit(x,y,2)   # 2 rows
→ x_new = np.linspace(1,5,10)
y_inp = np.polyval(P, x_new)
Plt.Scatter(x_new, x_interp)
```

→ Outlier treatment — distplot, boxplot
① Dropping the outliers
② Capping → replacing outliers with the nearest values that is not outliers
③ replace with mean & median
④ scaling and transformation

```
Ex: = df['Salary'].quantile(0.25)
                            (0.75)
>> IQR = Q3 - Q1
>>> lower_fence = Q1 - 1.5*IQR
    upper fence = Q3 + 1.5*IQR
```

Q1 - 1.5×IQR fence

Everything between max & min is not outlier

← low, upper → outlier

```
df_filter = df[(df.Salary >= lower_fence) & (df.Salary <= upper_fence)]
```

- df['Salary_mean_impute'] = np.where((df.Salary >= upper_fence) | (df.Salary <= lower_fence), df['Salary'].mean(), df['Salary'])

CAP ↳ decreased cap.

```
lower_cap = df['Salary'].quantile(0.05)
upper_cap = df(       "       (0.95)
```

- Eigen Value & Eigen Vector → useful in PCA → for reduce component
  ↳ dimensionality reduction techniques.

Eigen vector do not change direction
if some transformation is applied to them.

↳ 3×3, 4×4, 5×5,

↳ for any square matrix A, if there is a matrix V such that if I multiply matrix V with a scalar λ that

$AV = \lambda V$ holds true

↳ λ is Eigen Value & V Eigen Value

V = V so direction not change only change magnitude

in easy

$AV = \lambda V$

$AV - \lambda V = 0$

$V(A - \lambda) = 0$

$V(A - \lambda I) = 0$ (I = identity matrix)

eg. $\begin{bmatrix} 3 & 6 \\ 5 & 4 \end{bmatrix} \cdot V = \lambda V$

$V(A - \lambda I) = 0$

$V\left[\begin{bmatrix} 3 & 6 \\ 5 & 4 \end{bmatrix} - \lambda \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}\right] = 0$

$V\left(\begin{bmatrix} 3 & 6 \\ 5 & 4 \end{bmatrix} - \begin{bmatrix} \lambda & 0 \\ 0 & \lambda \end{bmatrix}\right) = 0$

$V\begin{pmatrix} 3-\lambda & 6 \\ 5 & 4-\lambda \end{pmatrix} = 0$

determinant $V\left|(3-\lambda)(4-\lambda) - 30\right| = 0$

$V\left[3(4-\lambda) - \lambda(4-\lambda) - 30\right]$

$V(12 - 3\lambda - 4\lambda + \lambda^2 - 30) = 0$

$\lambda = 9, -2$ so there is 2 Eigen Value in magnitude

$\lambda_1 = 9$

$\lambda_2 = -2$

PCA

↳ $\begin{bmatrix} \text{cov}(a,a) & \text{cov}(a,b) \\ \text{cov}(b,a) & \text{cov}(b,b) \end{bmatrix}$

$\lambda_1 V_1$ $\lambda_2 V_2$

PC1 PC2

eg

x_1	x_2
feature	f_2
2	3
3	4
7	2
4	2

$q x_1 - x_1$

$\lambda_1 = 9$, $\lambda_2 = -2$ vs $\lambda_1 = 9$ V_1

$\lambda_2 = -2$ V_2 λ_1 V_1

$PC_2 = \lambda_1 f_1 + \lambda_2 f_2$ → new data point with PCA

(15)

→ Imbalance class
→ feature extraction
Engineering, Scaling & Data encoding
→ Practical
→ pdf in PDA
→ simple linear Regression

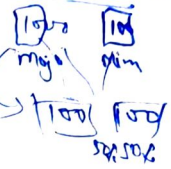
→ Imbalance
→ why happen it
→ What is Ration
→ How to find it

for classification data

1000
10 diabetic
1%, 2%, 5%, 10%
} What Accuracy = 90% Among
→ Major, Minor

types
(1) Under sampling → resample by that major class data. Disadv ⊙ Data lose
(2) Over sampling
(3) SMOTE

So



→ from sklearn.utils import resample
.. resample(df_minor, replace=True, n_samples=len(df_majority), random_state=1)

→ for sklearn.datasets import make_classification
X, y = make_classification(n_samples=1000, n_redundant=0, n_features=2, n_clusters_per_class=1, weights=[0.90], random_state=1)

→ Pip install imblearn
• from imblearn.over_sampling import SMOTE
→ oversample = SMOTE()
X, y = oversample.fit_resample(df_final[['f1','f2']], df_final['target'])

feature extraction

Not All feature to use in Model creation

① Create new feature
② Modify the existing feature
↳ 15+ → 15
16+ → 18
18+ → 16
} Date | age/month/years

distance	time	speed
50	2	50
100	3.5	
—	—	
=	=	

Curse of dimensionality
y with increase in no of features.
① Mode train cost ↑
②

(optional) • feature Scaling (PCA → Transformations) ① Interpretation easy ② less time in Computation ③ gradient descent optimization

Area	No. P	temp	Smoothing
1000	2	100	60
1500	3	50	50
2500	1	25	51
2500	0	5	5+

Why Same Scale

2×3 = 6
5×2 = 10
3×5 = 15
250×328 = ? → more Compute

distance	age	area	neat
50	2	100	200

Types of Scaling in ML

① Standardisation (ML algo) $Z = \frac{x_i - \bar{x}}{\sigma}$ SND ⇒ $\mu = 0$, $\sigma = 1$

② Normalization (min-max scal') → DL Algo Pref (0,1)

$X_{scale} = \frac{x_i - x_{min}}{x_{max} - x_{min}}$

③ Unit Vector

$OA^2 + OB^2 = AB^2$
$= \sqrt{\quad} = \sqrt{13}$
unit $\frac{\sqrt{13}}{\sqrt{13}} = 1$

⇒ Select the right feature

(1) Filter method
(2) Embedd method → Lasso Regression (for feature selection), Decision tree, random forest
(3) Wrapper method → Recursive feature elimination
forward selection, Backward selection

How?
f_1 | f_2 | f_3 price
(1) Correlation
(f_1, y)
(f_2, y)
(2) Multicollinearity (VIF) → correlation among feature correlated
x_1, x_2 δ x_1, x_2

7th Nov

- Data Encoding

(1) Nominal / OHE / Dummy variable → (1) Categorical to Numeric
(2) No order in the data
(2) Lable or ordinal - 1, 2, 3, 4, 5 → (dis Adv) → ML can assume order (Dis > DE)
↳ A, B, C
(3) Target Guided ordinal encoding
useful for ↳ ~~level of teaching~~ nominal higher categories

disadv
- Column has each 900 cols

↳ Sol select top 5 or top 10 ... category in feature of no. y

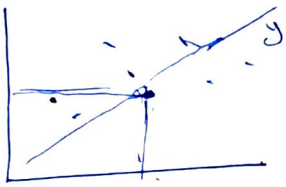
28 Apr - ML-5 LR

Linear Regression

↳ Intuition
↳ Mathematical
↳ Implementation

→ Simple LR

ML
SL / USL
Reg / Class
Ycont / Ydis
Yknot



$\hat{y} = mx + c$

^ - estimators

$\hat{y} = mx + c$ or $\hat{y} = \hat{\beta}_0 + \hat{\beta}_1 x$ ($\hat{\beta}_1 = m$, $\hat{\beta}_0 = c$)

$\hat{y}$ Predict, m coefficient / Slope, c intercept

$h_\theta(x) = \theta_0 + \theta_1 x$

or $h_\theta(x) = b + w x$

(Q) Which Line will be give you least error??

$y_{actual} - y_{predict}$ or $(y_{act} - y_{pre})^2$ =

So We want to have that combination of $(m, c) / \theta_0, \theta_1 / (b, w) / b_0, b_1$ where the overall error is least and the line will be best line.

So we say $= \sum_{i=1}^{n} \min (y_i - y_{pred})^2$

and this is called OLS (Ordinary Least Squared Error)

↳ This is the simplest / most ~~formulate~~ initial formula to find best fit line.

$\min \sum_{i=1}^{n} (y_{act} - y_{pred})^2$ → min the sum of square diff for all y_{act} & y_{pred} for all datapoints.

$\min \sum_{i=1}^{n} (y_i - (\beta_0 + \beta_1 x_i))^2$

optimal (β_0, β_1) → best fit line

(Q) Which technique is to find best fit line in LR

↓

OLS

Now if i divide OLS / total no of data point ⇒ then it called

Mean Square Error = $\min \frac{1}{n} \sum (y_i - (\beta_0 + \beta_1 x_i))^2$

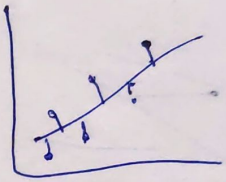
So High MSE = High cost ↑

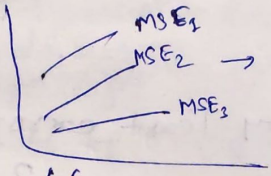
Cost function

Cost function :- A function/metric/Expression that ensures to Penalize you whenever you do something wrong (more error)

① MSE

$\rightarrow \frac{SSE}{n} = \frac{OLS}{n} = \frac{1}{n} \sum_{i=1}^{n} (y_{act} - (\beta_0 + \beta_1 x_i))^2$





$\rightarrow L_\alpha(m_\alpha, c_\alpha)$

↳ whichever has least MSE

So we optimize the process

↓ How

Gradient Descent

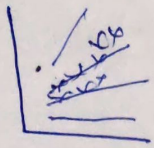
if outlier not treated use

$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^{n} (y_i - y_{pred})^2}$ =

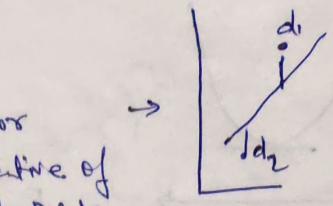
$OLS \rightarrow SSE \rightarrow \frac{SSE}{n} \Rightarrow Avg(mse) \approx \sqrt{MSE} \Rightarrow RMSE$

R^2 s
↳ not a Cost function

ML6 - 4th May Gradient & LR



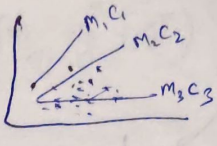
best fit line
↳ how
Minimize the error
⇓
best representative of all the data points

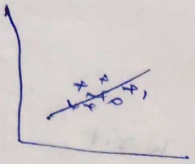


Error $= d_1 + d_2$
(should be minimum)

Total Error $= \sum_{i=1}^{n} \varepsilon_i$

how calculate ⇓
↳ $\sum_{i=1}^{n} (y_{act} - y_{pred})^2$





for Least error
$= \sum (y_i - mx_i - c)^2$
or β_0, β_1
$w, b \rightarrow DL$
θ_0, θ_1

for Least Error :- $\sum_{i=1}^{n} (y_i - mx_i - c)^2$

also write like this
- $J(m, c)$
- $J(\beta_0, \beta_1)$
- $J(\theta_0, \theta_1)$
- $J(w, b)$

So we Solve by 2 Approach

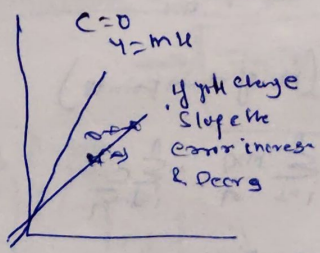
Close form Soln

→ formulate a Problem Statement into Maths equation
like ⇒ $ax^2 + bx + c = 0$
↓
$\alpha, \beta = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$

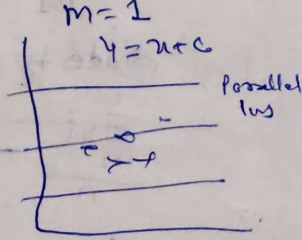
Iterative form Solution
⇓
Gradient Descent

Total Error $= \sum (y_i - mx_i - c)^2$
↳ m
↳ c

Scenario 1
$c = 0$
$y = mx$

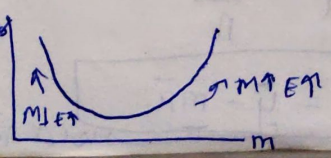


Scenario 2
$m = 1$
$y = x + c$

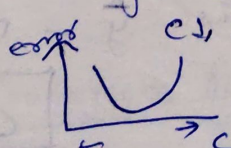


⇒ So observe
↳ if you increase m or decrease m error will change

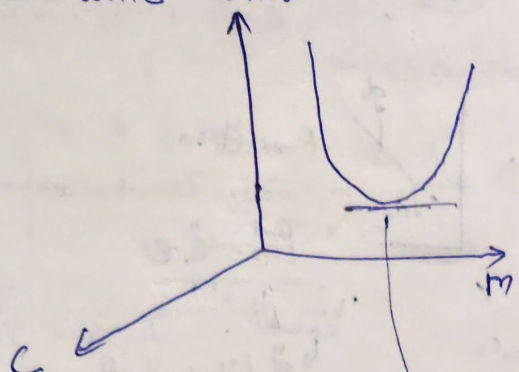
We Say ↳ Slop ↑ T. Error ↑
Slop ↓ T. Error ↑ →



If c ↑ error ↑



We write Error

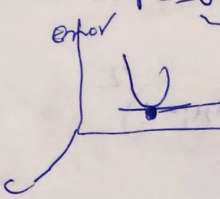


$Err = \sum (y_i - mx_i - c)^2$

$y = x^2$ = Parabola

because we want least error

Slope = 0 → so we do $\frac{dE}{\partial m} = 0$ w.r.t m & c



slope

$\frac{\partial E}{\partial c} = 0$

Differentiation

$\frac{\partial E}{\partial c} = 0$ why equal to 0?

→ Error should be least → So Slope or error = 0

$\Rightarrow \frac{\partial}{\partial c} \sum_{i=1}^{n} (y_i - mx_i - c)^2$

w.r.t c ↪

$\Rightarrow \sum_{i=1}^{n} 2(y_i - mx_i - c)(-1)$

$\Rightarrow \sum_{i=1}^{n} -2(y_i - mx_i - c)$

$\Rightarrow \frac{\partial E}{\partial c} = 0 \Rightarrow$

$\Rightarrow \sum -2(y_i - mx_i - c) = 0$

divide by -2

$\sum_{i=1}^{n} y_i - \sum mx_i - \sum_{i=1}^{n} c = 0$

divide by n (no. of data points)

$\frac{\sum_{i=1}^{n} y_i}{n} - \sum_{i=1}^{n} \frac{mx_i}{n} - \sum_{i=1}^{n} \frac{c}{n}$

$mx \to \bar{x}$ $y \to \bar{y}$ avg

So we can write like this: $\bar{y} - m\bar{x} - \frac{nc}{n} = 0$

$c = \bar{y} - m\bar{x}$ → what is m?

$n x^{n-1} \leftarrow x^n$

$\frac{dy_i}{\partial c} - \frac{\partial mx_i}{\partial c} - \frac{dc}{\partial c}$

Constant differentiation is zero

∴ for m diff records

↳ make equations

$$\frac{\partial E}{\partial m} = 0$$

$$E = \sum_{i=1}^{n} (y_i - m x_i - c)^2$$

↳ $\bar{y} - m\bar{x}$ → we find

$$\hookrightarrow E = \sum (y_i - m x_i - \bar{y} + m\bar{x})^2$$

$$\frac{\partial E}{\partial m} = \sum_{i=1}^{n} \frac{\partial}{\partial m} (y_i - m x_i - \bar{y} + m\bar{x})^2$$

$$= \sum_{i=1}^{n} 2 (y_i - m x_i - \bar{y} + m\bar{x}) (-x_i + \bar{x}) = 0$$

Solve this in any way
divide by 2 both side

$$m = \frac{\sum_{i=1}^{n} (x_i - \bar{x})(y_i - \bar{y})}{\sum_{i=1}^{n} (x_i - \bar{x})^2}$$

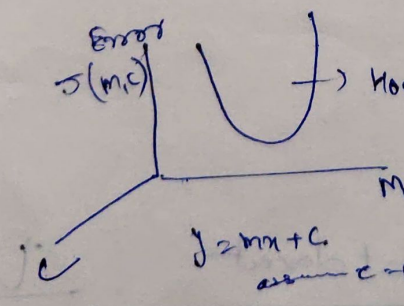
if multiple variable

x_1, x_2, x_3, x_4

it $\frac{\partial E}{\partial x_1}, \frac{\partial E}{\partial x_2}, \frac{\partial E}{\partial x_3} \cdots \frac{\partial E}{\partial x_n}$ → then more complex

↓↓ Iterative solution

↓↓ Gradient descent

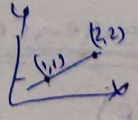


$$\rightarrow J(\theta_0, \theta_1) = \frac{1}{n} \sum_{i=1}^{n} (y_i - y_{pred})^2$$

it

x	y
1	1
2	2
3	3

$\theta = mx$



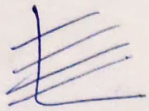
↓

→ Initialize random m, c.
→ move in the direction.
→ least error will be optimal m, c.

→ How do by Convergence Algorithm

* Convergence Algorithm

- keep making new best fit line in the direction of dp until Error (CF) is reduced as compared to previous line.



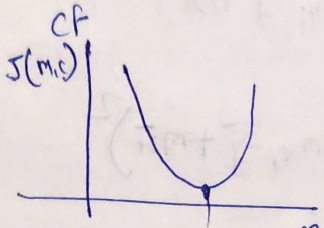
till when ??

↓↓

& where m, c is optimal

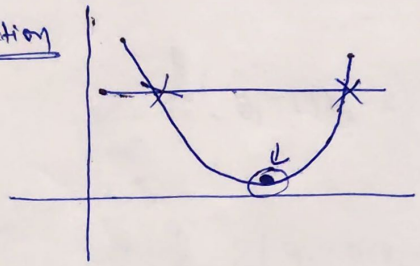
{ Error is least (or) Cf is least

Cost function



Cf is least

↳ Point of mining → global mining



function

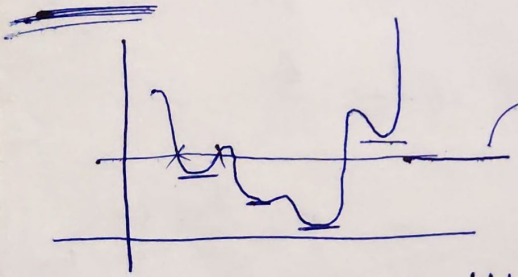
Convex function

→ Cuts at only two points

↳ convex

↓

one mining



non-convex fn

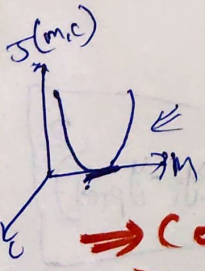
Cuts more than two points

↓↓

local, global minima

Pt - 3 · Global minimal

Pt - 1, 2, 3 → local minima



⇒ Convergence Algorithms

Repeat until convergence

Called gradient descent

$$m_{new} = m_{old} - \eta \frac{\partial(cf)}{\partial m_{old}}$$

$$c_{new} = c_{old} - \eta \frac{\partial(cf)}{\partial(c_{old})}$$

↓

Fast this

⇒ Time ↑

Cost ↑

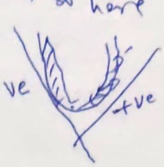
4th May 2:00

blog.dailys...dev/gradient-desc... hill
developers.google/ml/crashcourse/ ... https/graph, ... backing

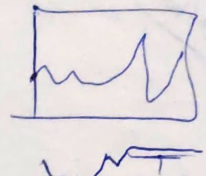
Gradient Descent

Slope — Coming down

↳ So here



In Deep learning



↳ Slope guides for Convergence

$m_{new} = m_{old} - \eta \frac{\partial(f)}{\partial m_{old}}$

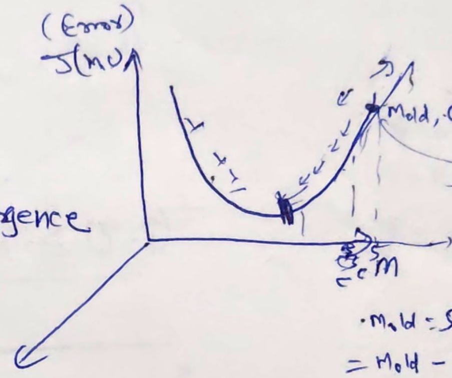
eta

it is (0.1 - 0.001)

learning rate → It decides the Convergence Speed.

η - too small → very slow for Convergence

η - too high → Exploding gradient problem or will never Converge



So you ↓ m

- Mold = 5
- = Mold - (some Value)
- = 5 - 3

↳ what? Value

$\left\{ \frac{\partial Cf}{\partial m_{old}} \right\}$

if -ve slop

$m_{new} = m_{old} - \frac{\partial Cf}{\partial m_{old}}$

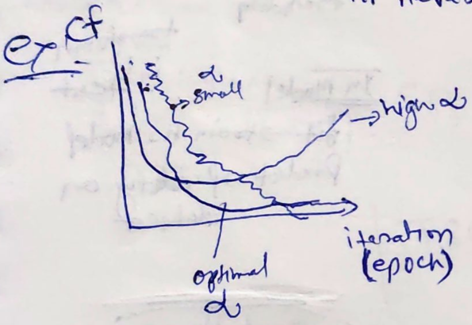
$= m_{old} - (\text{-ve slope})$

$\eta_{new} = m_{old} + \text{ve value}$

+ve slop

$m_{new} = m_{old} - (\text{+ve val})$ Slope +ve val

$m_{new} << m_{old}$



→ Contour plot

(24)

⇒ Linear Regression Implementation

Workbook

ML pipeline
→ Read dataset
→ Prepare the data
→ x, y
— train-test
— scaling
→ Model training
→ Model evaluation
→ Model prediction

(random_state = 111)
↳ reproduce the same result

Seed
in StandardScaling also

fit means → Calculate the Parameters
transform → Apply the Parameters
↓
traindata

transform in test

By Model
fit → train the model
Predict → Predicting on dataset

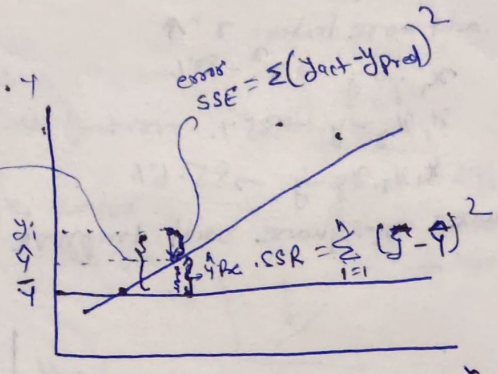
Evaluation
① R^2
② ~~RSME~~ Adjusted r^2

③ MSE
④ RMSE
⑤ MAE

Evaluation Metrix (Regression)

1. R square
2. Adjusted R-square
3. MSE
4. RMSE
5. MAE

R-SQUARE



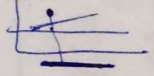
find R-square your Model is better.

Sum of Square due to regression

SSR → is the Σ y square diff of ~~Predicted~~ Avg y - ypred

If $R^2 = 85$ so we say 85% Variance ~~Predict~~ by R^2 will explain by this feature x.

→ can R square -ve?
→ yes but why
↳ if y vary far



1 - SSE/TSS

SSE = is the Σ y Square diff of y Actual yi - y Predicted.

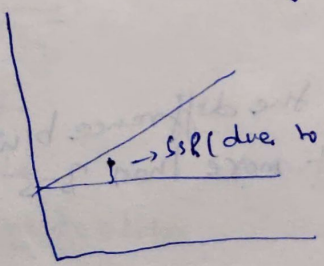
TSS = is the Σ of Square diff of yi actual - y Avg

$$\frac{SSE}{n} = MSE$$

$$R\text{square} = 1 - \frac{SSE}{TSS} \text{ or } \frac{SSR}{TSS}$$

also called Coffecient of determination → out of total error, SSR is the variation explained by linear regression line.

How R square Explain Variance & Unexplain Variance



→ Can I say this much of Variation is y is explained by Regression

R-Square = SSR/TSS → out of total variation, SSR is Variation explained.

↳ = SSR/TSS × 100 → The % variation in y explained by x.

(2) Adjusted r Square :-

We know → r^2 = % age explained variance in y due to X.

Example: House price

x_1 x_2 x_3 y

$x_1, x_2, x_3 \ldots x_{100}$

add more feature r^2 ↑

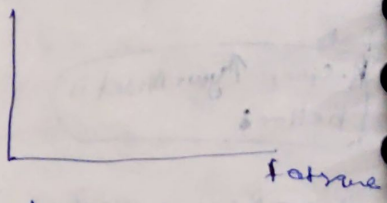
$x_1 \rightarrow y \rightarrow r^2 - 80\%$.

$x_1, x_2 - y \rightarrow 85\%$.

$x_1, x_2, x_3 \rightarrow y \rightarrow 85.6\%$

* As we add more features r-square will improve (or remain as it is constant)

id $x_1, x_2 \ldots x_{100}$

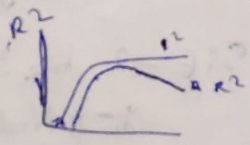


then How to know which feature is not contribute much.

So use

Adjusted r-square

↳ It penalizes r^2 as we add new features.



$$Ar^2 = \frac{1 - (1-R^2)(N-1)}{N-P-1}$$

N = no of dp

P = no features

Example

Scenario 1 $R^2 = 87$, $N = 11$, $P = 2$

$$Ar^2 = 1 - \frac{(1-0.8)(11-1)}{11-2-1} = \frac{1-0.2\times 9}{8} = 0.75$$

$R^2 = 87$, $N = 11$, $P = 8$

$$Ar^2 = \frac{1-(0.2)(10)}{11-8-1} = 1 - \frac{2}{2} \Rightarrow 1-1 \Rightarrow 0 \text{ Ans}$$

(1) $Ar^2 < r^2$

(2) only add feature in the model if the diffrence b/w r square & adjusted r square is not more than 3-5%.

(3) MSE (mean Square ERROR)

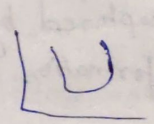
$$\frac{1}{n}\sum_{i=1}^{n}(y_{act}-y_{pr})^2$$

(*) Error is quantified
(*) Lower the MSE better the model will be

(*) Can we use another Cost function
Yes

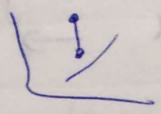
Advantage

(1) * It is differentiable
So it is use as Cost function



You can differentiate anypoint & it is convex so it is 1 minimal.

(2) Emphasis (Adv) on large error $(y_{act}-y_{pr})^2$



Disadvantage

(1) Not robust to outlier
(2) It is not in same unit as y

$y_{ac} = 3m$
$y_1 = 6m$
$error = (3-6)^2$
$= 9m^2$.

(4) MAE

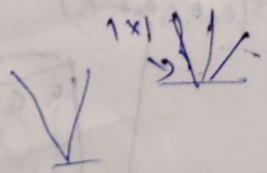
$$= \frac{1}{n}\sum_{i=1}^{n}|y_{act}-y_{pred}|$$

Adv

↳ Less sensitive to outlier
→ more interpretable.
↳ It is of same unit

DisAdv

→ Not differentiable at $x=0$
→ time consuming



→ Lower the MAE Better will be the model

(5) RMSE (Root mean Square Error)

$RMSE = \sqrt{MSE}$.

Adv

(1) Same unit
(2) Differentiable
(3) Less sensitive to outlier

11 May
MLR

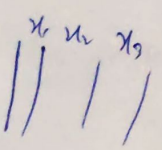
(28)

Agenda

- Assumption of LR
- MLR
- Polynomial reg
- Lass, ridge, Elastic net
- Cross validation & Hyperparameter
- Logistic Regression

Cost fun = loss function

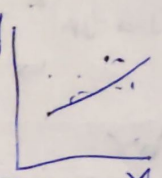
⇒ Multi Linear Regression



Why we go MLR

(1) All the information y is not captured by x in order to capture more & more information we require more x.

MLR ⇒ More than 1 I.V (Independent Variable is used.

1-IV ⇒ $y_{pre} = \theta_0 + \theta_1 x$ → 

→ interpret will only change

MLR $h_\theta(x) = \theta_0 + \theta_1 x_1 + \theta_2 x_2 + \theta_3 x_3 \cdots \theta_n x_n$

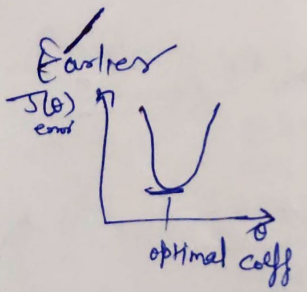
$\theta_1 \cdots \theta_n$ → are optimal coefficient

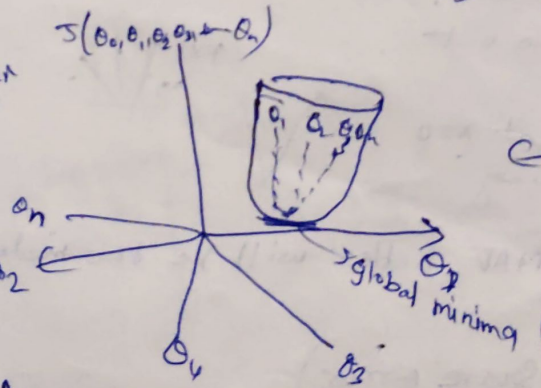
So what is Cost function of this

$$Cf = \frac{1}{n} \sum_{i=1}^{n} (y_{act} - y_{pred})^2$$

↳ $y_{act} - (\theta_0 + \theta_1 x_1 + \theta_2 x_2 \cdots + \theta_n x_n))^2$

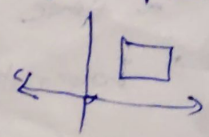
Earlier



MLR → 

MLR Visualization

↳ It is Plane Hyperplane



→ In MLR, the Intercept is the point where hyperplane intersects the Vertical

Example

(1)	# hr or hours	Marks
(2)	Sale price year	Age

Avg marks obtain by asld even if he θ0 θi did not study

$\therefore y = 50 + 2.5x$

→ θ_1 ⇒ with 1 unit ↑ in # of hours studied, the marks increase by 2.5 unit

$y_{pred} = 100 - 3.2(Age)$

⇒ Then how to interpret in MLR

SP, RPM

$y_{pred} = 3 - 0.52(Age of) + 2.5\,RPM$

θ_0 θ_1 θ_2

Mathematically change but fundamentally same

$\theta_1(-0.52)$ → with 1 unit increase in age of car the SP of car decreases by 0.5 units on Avg keeping RPM constant? (means 0)

Whatever you taking others are constant

$B_2(2.5)$

If data lie

H.W Ques
→ Sklearn
→ Explain/Unexplain variance R^2
→ Stats model
→ feature selection

Polynomial Regression:

(1) Simple polynomial regression (1 Dependent Var (DV), 1 Independent Var (IV))

Polynomial degree 0, $h_\theta(x) = \theta_1 x^0$

" 1 → $h_\theta(x) = \theta_0 x^0 + \theta_1 x_1^{(1)}$ → Simple LR

" 2 → $h_\theta(x) = \theta_0 x^0 + \theta_1 x_1 + \theta_2 x_1^2$

" 3 → $h_\theta(x) \Rightarrow \theta_0 x^0 + \theta_1 x_1 + \theta_2 x_1^2 + \theta_3 x_1^3$

n = $h_\theta(x)$ $\theta_0 x^0 + \theta_1 x_1 + \ldots \theta_n x_1^n$

d=3, d=2, d=1

It capture the non-linear Relationship

* As you increase the degree, you might get an overfitting model.

⇒ Non-linear relationship → also other model Decision Tree → non linear

PR → Disadv: Overfitting the model

For multiple X

Polynomial degree 2: $\theta_0 + \theta_1 x_1 + \theta_2 x_2 + \theta_3 x_3 + \theta_4 x_1^2 + \theta_5 x_2^2 + \theta_6 x_3^2 + \theta_7 x_1 x_2 + \theta_8 x_2 x_3 + \theta_9 x_3 x_1$

$x_1 x_2$ → Cross product keeping power in mind

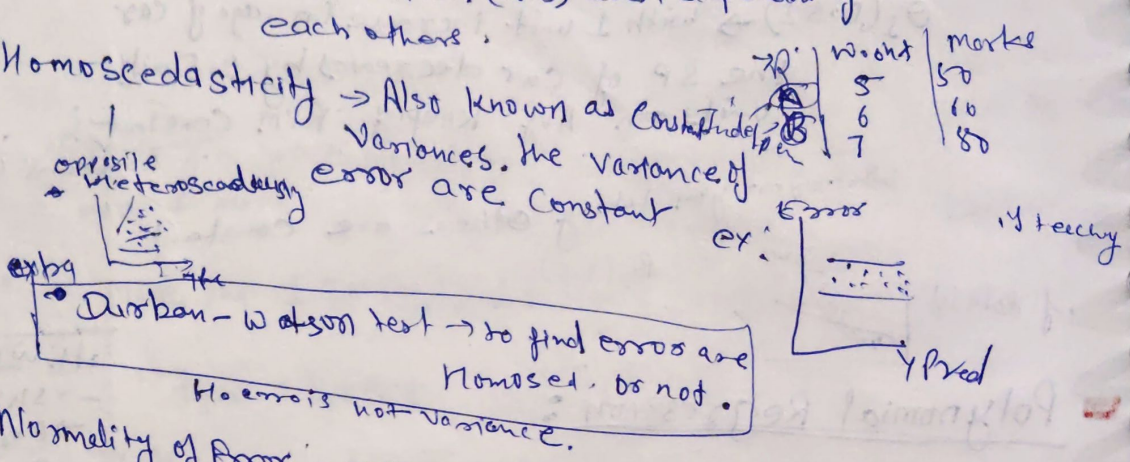
(30)

Assumption of Linear Regression

(1) Linearity → X and y Should have linear Relationship.

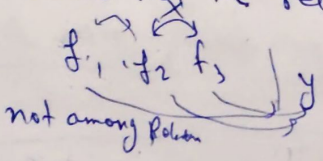
(2) Independence → Observation (rows) are Independent of each others.

(3) Homoscedasticity → Also known as Constant Variances. The variance of error are constant



(4) Normality of Error :- error should be normally distribute.

(5) The feature Should not be related or Shout have relation



(6) what part x_1 is contributing to y & what part x_2 is contributing to y

Can we say $x_1 = x_2$ ↳ not answer

$8x_1 + 2x_2 = y$
$10x_1 = y$
$2x_1 + 8x_2 = y$
$10x_2 = y$

*so we say Multi-Col-linearity
↳ may together linear relation

$x_1 \to x_2$ } → Correlation
$x_1 \to y$ }

$x_1 \approx (x_2, x_3)$
$x_1 \approx (x_2, x_3, x_4)$ } multi-collinearity

Multi Collin → where a feature exhibits a linear relationship with more then one variable

↳ It never talk about $x \to y$ target

$x_1 \sim x_2 x_3$ ..
$x_5 \sim x_1 x_2 x_3 \ldots x_4$

What → One feature explained by other feature.

Why → you can not interpret correctly what is the contribution of each individual feature w.r.t y

you only lose interpretion

it is severe Problem → Ans → No
severe
↳ No effect on Prediction

then Problem → you lose interpretability

② $x_1\ x_2\ x_3\ x_4$ → y
×
② Computationally expensive → model training

Solution | ① Detect Multi-Collinearity or Measure by
VIF (Variance inflation factor)
↳ a measure of amount of multi collinearity in regression

$$VIF_i = \frac{1}{1-R_i^2}$$ → R^2 → % variation in y explained by X

How 1 feature explain by other features
$x_4 \sim (x_2\ x_3\ x_1 \dots x_n)$

How $x_1\ x_2\ x_3\ x_4$
$x_1 \sim x_2, x_3, x_4$
$x_2 \sim x_1, x_3, x_4$
$x_3 \sim x_1, x_2, x_4$
$x_4 \sim x_1, x_2, x_3$
The x_i is treated as y & calculate the R-square

VIF ~ 0 to ∞

So → Since VIF doesn't impact prediction, you should always ask business team before dropping.

VIF ≥ 10, then drop feature one by one not together.

2 Question → Why VIF ≥ 10 → $10 = \frac{1}{1-R^2} \Rightarrow 1-R^2 = \frac{1}{10} \Rightarrow R^2 = 1-\frac{1}{10} = \frac{10-1}{10} = \frac{9}{10} = 0.9$

$x_1 \sim x_1, x_2, x_3$
↳ 90% of variance in X is explained by $x_2\ x_3\ x_4$

if 1000 features then How.
No library
① 1000 time VIF
RFE (Recursive feature Elimination) ↳ Sklearn
② PCA

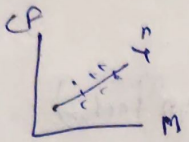
Kinds of Multi
- Data based
- Structural

(42)

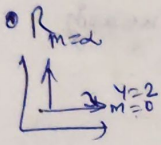
Regularisation ⇒ To add something / to regularisation

to regularize
|
to penalize

e.g. data → train, test
Accn low bias, Accn High Variance } overfitting



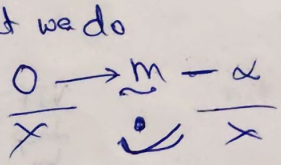
m is very high → model has memorized the dP
"It means overfitting

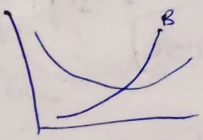


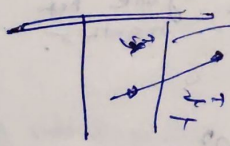
$y = mx + c \Rightarrow m = 0$ then y = constant
So No use of x in predicting y → Underfitting
↓
High bias

or $y = mx + c \Rightarrow m = \alpha$ (m is very-very large)
you are giving all importance to x_1 → it is memorising x_1
↓
Overfitting

So what we do







→ Introduce some bias
↓
error in it

$$Cf \rightarrow \frac{1}{n}\sum_{i=1}^{n}(y_{act} - mx - c)^2 + m^2$$

error = 0 → so we add some

or introduce some bias

① Ridge $- Cf + \lambda m^2$ ($\lambda \to \infty$) ⇒ (L_2 Regularization, L_2 Norm) ↑ bias / Training error
② Lasso $- Cf + \lambda |m|$
③ Elastic net $- Cf + \lambda m^2 + \lambda |m|$

~~Cf~~ = Ridge $= Cf + \lambda m^2$

↓ λ decide by regularization in hyperparameters tuning
or alpha

Related

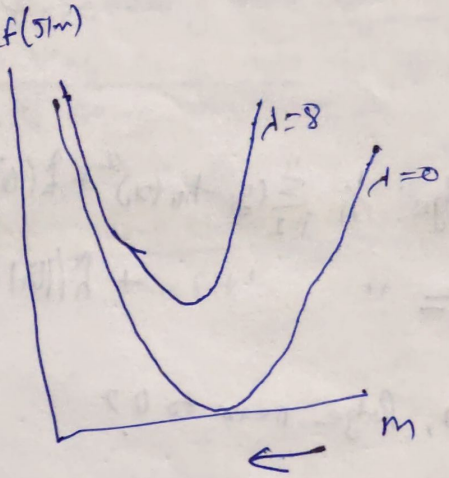
Relationship b/w λ, Cost function, m

$\uparrow\uparrow$ Cf$\uparrow$

So minima is shift

So m value is $\downarrow$

$\lambda\uparrow$ m$\downarrow$



② Lasso (L1 penalty, L1 norm)

$Lasso = Cf + \lambda \cdot \sum_{i=1}^{n} |slope|$

→ As you ↑ λ the less significal feature will be removed in Lasso →

$\lambda\uparrow\uparrow$ Cf$\uparrow$ minima shift

$\lambda\uparrow$ slope$\downarrow$

Slope can be 0.

Ridge (

→ L_2 norm / penalty

→ $\lambda\uparrow\uparrow$ m$\downarrow$

→ m will never become 0, it will be close to 0

→ Reduces the overfitting (by reducing the Coefficent)

→ effective in handling multi Collinearity (reducing ~~Collinearity~~ all the Coeff)

$x_1 \sim [x_2, x_3]$ x_1 is also passed x_2, x_3 is also passed

memorization happen → overfitting → Ridge reduces overfitting → ↓ reduce multi Colin as we see

Disadv - Doesn't make least important feature Coefficient to 0. → overcome by Lasso → feature Selection

Lasso (final)

→ L_1 norm / penalty

→ $\lambda\uparrow$ m$\downarrow$

→ m will become 0

Adv leads to feature Selection

- Removing insignificant feature (also called Automatic feature Selection / sparsity)

→ Dis Adv → lead

(3) Elastic net → Combination → Reducing overfitting (L_2)
→ Feature Selection (L_1)

$\lambda_1 \lambda_2$
↳ hyperparameters

→ General formula → Ridge: $\frac{1}{n} \sum_{i=1}^{n} (y_i - h_\theta(x_i))^2 + \lambda(\theta_1^2 + \theta_2^2 + \theta_3^3 + \dots \theta_n)$

Lasso: = " " $+ \lambda(|\theta_1| + |\theta_2| + |\theta_3|)$

(8) Why Lasso makes coffe 0, Ridge near to 0?
Ans

May 18 may -
- Cross Validation & hyper parameter tuning
- Logistic regress (depth)
- Evaluation matric
⇒ Practical
→ Multi class classification

Cross Validation :- Experimenting with different arrangement of same data to build different models of same algorithms.

* when if you don't fix random state each time r-square will changes.
↳ for

→ Exp1 - model 1 ...
Exp2
Exp3

* Many experiments and average out the metrics increases the confidence in the model.

Data
→ training → Train → build → Validate
→ training → Validation
→ testing

example

training of different model:

	train	validate	r-squ/accuracy
Exp-1	Train	Val	
Exp-2	"	"	
⋮	⋮	⋮	

in Exp 1 data not Repeat in Exp 2 data
↳ Now arrange of same data in different way

Types

① Leave one Out Cross Validation (LOOCV)

→ not use in Industry but core
↔ test data is unseen data

Exp1 [1] 2 | 3 | ... | 100
Exp2 1 [2] 3 | 4 | ... | 100
Validation

1-validate
99 - training data
2nd data for validate
99 - train data

Avg-r-square [100] mode

Adv

dis Adv - ① time Complexity
② Model Overfitting

② leave P out cross Validation
↖ overcome

DS1
| 1 | 2 | 3 | 4 | ... |
Val — Train

③ K-fold Cross Validation

$n = 100$ ⇒ k group ($k=5$) → No of groups that you want to divide the data

$\frac{100}{5} = 20$ data point

Example

Exp-1 | 20 | 20 | 20 | 20 | 20 | → R^2_1
(Validate: first 20; train: remaining)

Adv
- So time Complexity Reduce

100 data point

Exp-2 | 20 | 20 | 20 | 20 | 20 | → R^2_2
(Train | Val | train)

| 20 | 20 | 20 | 20 | 20 | → R^3
(Val: third block)

→ R^4 (Val)

→ R^2_5 (Val)

Avg(R^2)

④ Stratified K-fold Cross Validation (imbalanced data)

Class 1 = 400 (40%)
Class 0 = 100

Ex:- k=5, n=100

Exp 1 | 20 | ... |
Val
10 row class 0
10 row class 1

Exp 5 | ... | 20 |
Val
↑ 10th of 0 class
↑ 10th of 1 class

* HyperParameter

What is •
Why necessary •

→ you remember Ridge → $\sum_{i=0}^{n} (y_{act} - y_{pr})^2 + \lambda(\text{slope})^2$

$(y_{act} - m x_i - c)^2$

these are parameter

but C, M is learned from data (during the training change)

def → It is External Configuration of a model that are not learned from the data but are set prior to training process

(External Parameter) λ → dictates the behavior of training algos

these Parameter called Hyperparameter

• you don't know what value you put λ

$\lambda = (1, 2, 4, 19, 20, 100, 1000)$

↳ Train → best performance → Hyperparameter tuned.

Tune the → this process called Hyper parameter tuning

Ⓐ
• | fold | | 20 |
7/8, 1/8th, 1/8th, 1/4th

CV = 5
| class 0 | cls 1, cls 2 |
train | Vale
4/5
1/10 class
1/10 th

bike
Read also
impact your drive

18 Apr
1:30

Relation b/w hyperparameter & CV

(3) Hyperparameter tuning with cross Validations

How → by

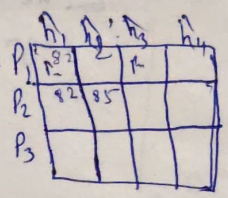
① Grid Search CV (Grid search + Cross Validation)

② Randomized search CV

① Grid Search CV

h → 1, 2, 3, 5, 10

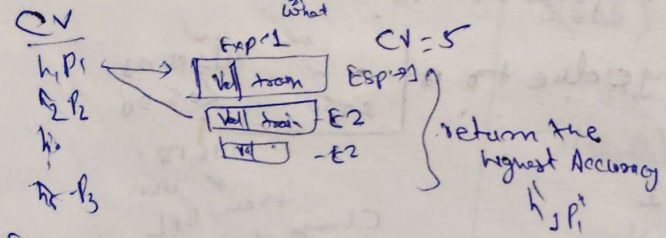
p → p1, p2, p3



which give high R2 value

All possible combination of hyperparameter is taken

→ So where does CV comes to role → in B.



→ for each combination of h & p k-fold CV will happen

→ How many Model they build

↳ 3 × 5 × 5

↓

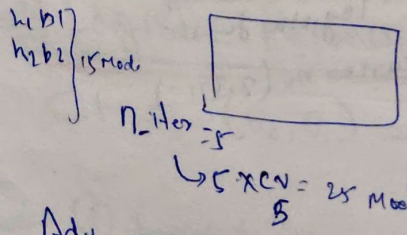
Disadv GS CV

- Time Complexity increase with huge data from model training

② Randomised Search CV

→ We will not see all possible combination.

→ Out of all the possible combination only select some random combination



↳ 5 × cv = 25 Models

5

Adv → Time Complexity decrease

⇒ If data is small → instead of train test split → you can go with k-fold cross validation.

⇒ Lassoridge & Elastic net → Cross Validation

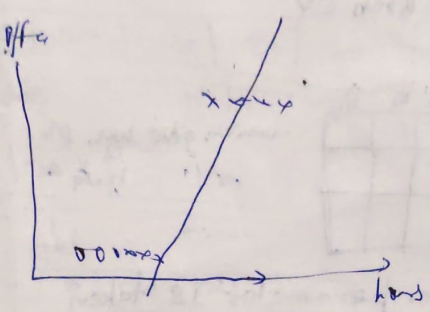
18 May → 2:25

Classification → binary
→ Multi Class

2:30 ① Logistic Regression

→ SVM
→ Gradient
→ XGBoost
→

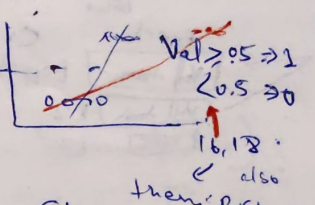
Hours	Pass/fail
2	0
8	1
3	0
4	0
6	1



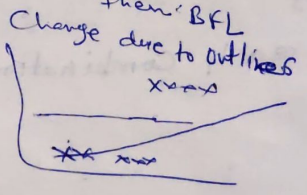
← here we cannot do Regression line.

② Why we can not use regression line?

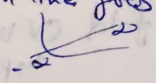
① Best fit line will change due to presence of outliers.



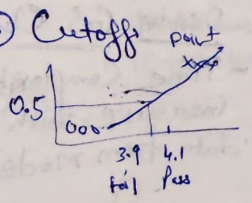
← also then BFL Change due to outliers



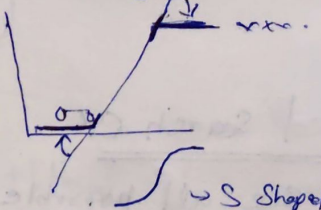
② range of output (y) → 0 to 1
→ but regression line goes −∞ to +∞



③ Cutoff



→ So Squashing of line do to solve 2 problem
- ∞ to ∞ → 0, 1
- outliers → no problem now



explain logistic Regression

→ S Shaped curve or Sigmoid curve
↓↓
this curve called logistic function

$$\sigma = \frac{1}{1+e^{-z}}$$

e - euler no (2.371...)

$$\frac{1}{1+e^{z}}$$

$z = -\infty \Rightarrow \frac{1}{1+e^{(\infty)}} = \frac{1}{1+e^{\infty}} = \frac{1}{\infty} = 0$

$z = \infty \Rightarrow \frac{1}{1+e^{-\infty}} = \frac{1}{1+\frac{1}{e^{\infty}}} = \frac{1}{1+\frac{1}{\infty}} = \frac{1}{1+0} = 1$

$h_\theta(x) = \theta_0 + \theta_1 x$ → Best fit line

$h_\theta(x) = \sigma(\theta_0 + \theta_1 x)$

$h_\theta(x) = \frac{1}{1+e^{-(\theta_0+\theta_1 x)}}$ → Logistic Regression model

Cost function

$$\frac{1}{n}\sum \frac{1}{2}(y_{act} - y_{pre})^2$$

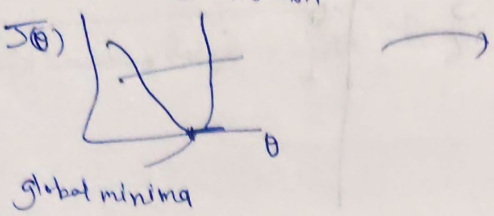
To get optimal θ_0 & θ_1 minimize the Cost functions.

linear Reg e

Linear Regression

$$J(\theta_0,\theta_1) = \frac{1}{n}\sum_{i=1}^{n}(y_i - h_\theta(x_i))^2$$

Convex function



Logistic regression

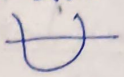
$$J(\theta_0,\theta_1) = \frac{1}{n}\sum_{i=1}^{n}(y_i - h_\theta(x_i))^2$$

$$h_\theta(x_i) = \frac{1}{1+e^{-(\theta_0+\theta_1 x_i)}}$$

but this is non convex function

So there is local minima

Convex function
↓
when horizontal line cut 2 point



So here Cost function is use

• Log Loss function

$$J(\theta_1,\theta) = -y_i \log(h_\theta(x)_i) - (1-y_i)\log(1-h_\theta(x_i))$$

$$J(\theta_1,\theta_j) = \begin{cases} -\log h_\theta(x) & \text{if } y_i = 1 \\ -\log(1-h_\theta(x_i)) & y_i = 0 \end{cases}$$

To minimize chang θ_0 & θ_1

by convergence Algorithm

Repeat until convergence

{

$$\theta_j := \theta_j - \alpha \frac{\partial}{\partial \theta} J(\theta_0,\theta_1)$$

}

toge optimal θ_0, θ_1

For multiple Variable

$$h_\theta(x) \; \frac{1}{1+e^{-(\theta_0+\theta_1 x_1+\theta_2 x_2 \cdots \theta_n x_4)}}$$

⇒ **Logistic Regression with Regularisation :**

$$Cf = J(\theta_0,\theta_1) = -y \log(h_\theta(x))$$

① Lasso :

② Ridge :

③ elastic net :

$$C = \frac{1}{\lambda}$$

Logistic Regression

$$\frac{1}{1+e^{-z}} \rightarrow$$

No of hours studied	Pass/fail
5	1
2	0
3	0
6	1
=	=
=	=

if N samples $\begin{matrix} 0 \\ 1 \end{matrix}$

- for Samples labelled 1 : Estimate θ_1 such that $\hat{p}(x)$ is as close as 1 possible.
- for Sample labellele ; Estimate θ_1 such that $1-\hat{p}(x)$ is as class to 1 possible.
  or
  $\hat{p}(x)$ is close to 0 possible

$\prod$ it means multiply all the labels $\rightarrow \prod \hat{p}(x)_i$ s in $y_i=1$ $\cdot \prod 1-p(x_i)$ s in $y_i=0$

$$L(\theta) = \prod \left(p(x_i)^{y_i} \times (1-p(x)_i)^{1-y_i} \right)$$

Likelihood

$a^n \Rightarrow n \log a$

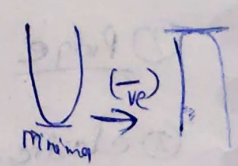
do log both side

$$\log(L(\theta)) = \sum_{i=1}^{n} y_i \log p(x_i) + (1-y_i) \log$$

log likelihood estimate

Maximum Likelihood estimation

So we $-ve$ So the maximum error reverse to minimum error



19 May → Classification Evaluation Metrics & Implementation (Iris dataset)

i

→ newton-cg, sag, saga → these are optimization algorithms

→ Lasso, Ridge → alpha = λ $C = \frac{1}{\lambda}$ (because SVM, DT, ensemble that's by use C)

Smaller values specify stronger regularization

$\frac{p}{1-p}$

Odds ⇒ p(suss) / p(failure)

log odds

$\log\left(\frac{p}{1-p}\right)$

p ↑ log odds ↑

↑ odd ↑ log ↑

Monotonic relationship
↳ one direction

Evaluation metrics for classification

1. Confusion matrix
2. Accuracy / misclassification rate
3. Precision
4. Recall
5. F-Beta Score

6. True Positive rate (sensitivity)
7. false Positive rate
8. True Negative rate (Specificity)
9. ROC-AUC
10. Precision-Recall / Sensitivity-specificity trade off.

1:37

Predicted Value	Actual Value: True(1)	Actual Value: False(0)
T(1)	2 TP	2 FP
F(0)	2 FN	1 TN

	0	1	2
0	True0	false0	false0
1	false1	True1	false1
2	false1	false1	True1

① Accuracy $= \frac{TP+TN}{TP+FP+FN+TN}$

*Misclassification rate = opposite of Accuracy

$\frac{FP+FN}{total} = 1 - Accuracy$

③ Precision :-

Imbalance class
100 obs → 90 c1, 10 c0

If → | 90 | 10 |
|---|---|
| 0 | 0 |

what accuracy without model = $\frac{90+0}{100} = 90\%$

Accuracy doesn't give more Priority to minority class

Out of all predicted ones, how many are actual ones

↳ Out of all actual Value, how many are correctly predicted.

$P = TP/(TP+FP)$

Precision of +ve cases:

④ Recall : Out of all predicted values, how many are correctly predicted with actual value.

for +ve class

↳ $\frac{TP}{TP+FN}$

Not Class

USECASE-1 Spam classifier

So, we wanted a model with high precision

↳ Threshold/cutoff ↓ — Increase the cutoff 0.5 to 0.9

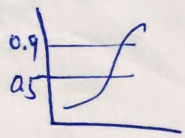
Precision Used

TP ⇒ Mail - Spam(1) } Correct
Model Spam(1) }

TN ⇒ Mail - Not Spam(0) } Correct
Model - Not Spam(0) } Case

FP ⇒ Mail → Not spam (input) Blunder } Wrong
Model → spam } Predict

FN ⇒ Mail - spam → Model → Not a spam



19 May
2:40

If change the criteria
50% → Spam
After Accuracy the λ 0.9
Minimise the case FP
goes to 30% → Spam

USE CASE 2 :

	1	0
Pred 1	TP	FP
0	FN	TN

TP = Actual - diabetics
FN Model Not "
TN

FP - Act → No diabetic
Model - Predicted diabetic → 1st level check, 2nd level check for cross check

FN → Act - diabetic
model - No diabetic → Blunder ✓ so it is important

⇒ FN is important (more) Why??
Use Recall

H.W ① Conviction of a person in the Court trial.
② Stock Mkt Crash or not

⑤ f-beta Score $(1+\beta^2)\frac{P \times R}{P+R}$

P → Precision
R → Recall

① if FP and FN both are important $\beta = 1$

$$F_1 \text{ Score} = 2\frac{P \times R}{P+R}$$

② if FP is more important than FN

$\beta = 0.5$

$$F_{0.5} \text{ Score} = (1+0.25)\frac{P \times R}{P+R}$$

③ if FN is more important than FP

$\beta = 2$

$$F_2 \text{ Score} = (1+4)\frac{P \times R}{P+R}$$

⑥ True Positive Rate ⇒ Out of all actual 1, it is actually Predited one.

↳ Sensitivity, Recall $TPR = \frac{TP}{TP+FN}$ — Some Ans / Totall Ans

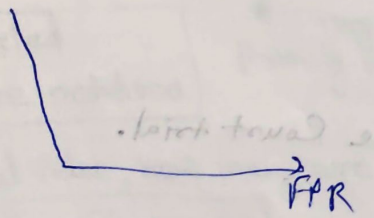
⑦ false Positive Rate : $\frac{FP}{FP+TN}$

⑧ True Negative Rate $TNR = \frac{TN}{FN+TN}$

↳ specifisity ⇒ $TNR \Rightarrow 1 - fPR$

⑨ ROC-AUC :↳ Recievor Operating Characteristic Area under Curve.

TPR Vs FPR



x_1	y_1	Cutoff y_1 0.1	0.2	0.3	0.4	1

for different cutoff

X	Y	Prob	cutoff 0.5 ypred
5	1	0.9	1
6	0	0.8	1
2	1	0.6	1
8	0	1	0
4	1	1	1
⋮	⋮	⋮	⋮

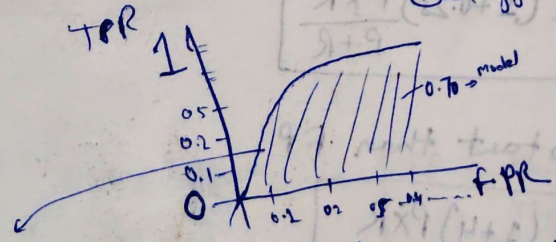
↳ TPR FPR

for different cutoff calcat

0.1	0.2	0.3	0.4	.....	1
0	0	1	1		
⋮	⋮	⋮	⋮		
TPR, FPR	TPR FPR	1	1	–	–

→ different Predict. Class

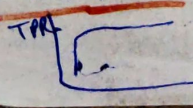
So Calculate all TPR & fPR of different cutoff & Plot a chart



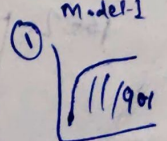
Not plot cut-off only Plot their Value of TPR & FPR

0.70 → Model will correctly Predict 70% of time.

• Higher the AUC betters the model will be



① Model-1 ② Model-2

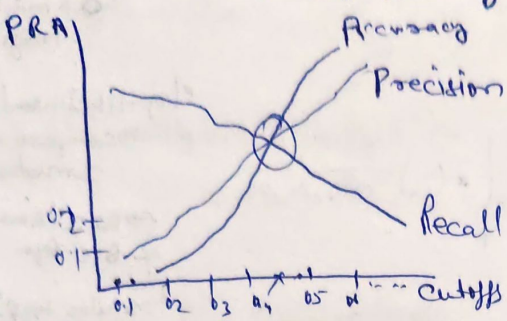


Cutoff → by business te

(18) Cutoff → by business team (b.

if No business team (default is 0.5)

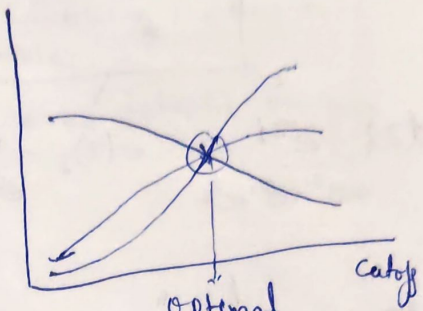
Precision - Recall - Accuracy



Where PRA cut that is more accurately cutoff and give the priority to all (PRA)

OR

Sensitivity - Specificity - Acuuy



are both same Concept

Linear Rc
Logistic
Hypothesis
Multiclass classification
• Decision tree then
• SVM
• ensemble
• PCA
• Anomaly
• Unsupervised
• Time series

Class 25 May

Multiclass Classification & Hyperparameters tuning

① OVR ** (One Vs Rest)

Solution → ② Multinomial

make data → make_classification (sample=1000, n_feature=10, n_informative=5, n_redundant=5, n_classes=2,

ROC-AUC we for compare 2 models Performance

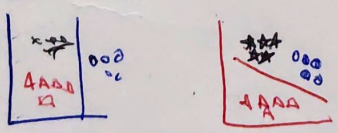
Model1 10 feat → R square / Cost / Adjusted R-Square

Model2 20 feat

↳ why because Curse of dimensionality
↳ Time complexity
↳ overfitting
↳ No better Interpretion

→ One Vs Rest

① 3 class



② → Multinomial method | Softmax regression

Multinomial loss — specific of Logistic

→ we don't decomposed the problem into binary classification.

→ Modify the loss / Cost function

→ Single Model

different { M1 - 98% (Acc, P, R)
M2 - 78% (Acc, P, R)

Absolute	Relative
98% Sci	TPR
98% Ans	FPR
XAO compare	Compare 2 model

↳ Sigmoid $\frac{1}{1+e^{-z}} = \frac{1}{1+\frac{1}{e^z}} \Rightarrow \frac{1}{\frac{e^z+1}{e^z}} \Rightarrow \boxed{\frac{e^z}{1+e^z}}$

• CHI test for (categorical) in Stats Model
→ OLS
→ f-test (Continuous)
↳ Comparing Variance

Softmax $\sigma(z)_i = \frac{e^{z_i}}{\sum_{j=1}^{K} e^{z_j}}$ j is no of class

$\sigma(z_1) = \frac{e^{z_1}}{e^{z_1}+e^{z_2}+e^{z_2}}$ $\sigma(z)_2 = \frac{e^{z_2}}{e^{z_1}+e^{z_2}+e^{z_3}}$ $\sigma(z_3) = \frac{e^{z_3}}{e^{z_1}+e^{z_2}+e^{z_3}}$

• f-statistics
↳ 0 → model is insignificant

• log-likelihood
↳ compare 2 model
↳ AIC (Akaike Info Criterion)
↳ BIC Bayesian
↳ similar to R^2 but R^2 better

1,
Cost fn (log loss) $= -\left[\frac{1}{n}\sum_{n=1}^{n} y_i \log \hat{y}_i + (1-y_i)\log(1-\hat{y}_i)\right]$

↳ Multiclass (modify.) $\Rightarrow -\frac{1}{n}\sum_{i=1}^{n}\sum_{k=1}^{K} y_k^{(i)} \log \hat{y}_k^{(i)}$

k → no of class
i = no of datapoints

↳ t
↳ $P > 1+1$
↳ when $df \geq 30$

x_1	x_2	y	$y_{k=1}$	$y_{k=2}$	$y_{k=3}$
x_{11}	x_{12}	1	1	0	0
x_{21}	x_{22}	2	0	1	0
x_{31}	x_{32}	3	0	0	1

Class-1 $\Rightarrow y_1^{(1)} \log \hat{y}_1^{(1)} + y_2^{(1)} \log \hat{y}_2^{(1)} + y_3^{(1)} \log \hat{y}_3^{(1)} \ldots$