AI new handrawn

Topics:

- Introduction
- Types of Learning
- Features
- Training - Testing - Validation set
- Bias and Variance
- Underfitting + Overfitting
- Measuring performance
- Matrics Estimation

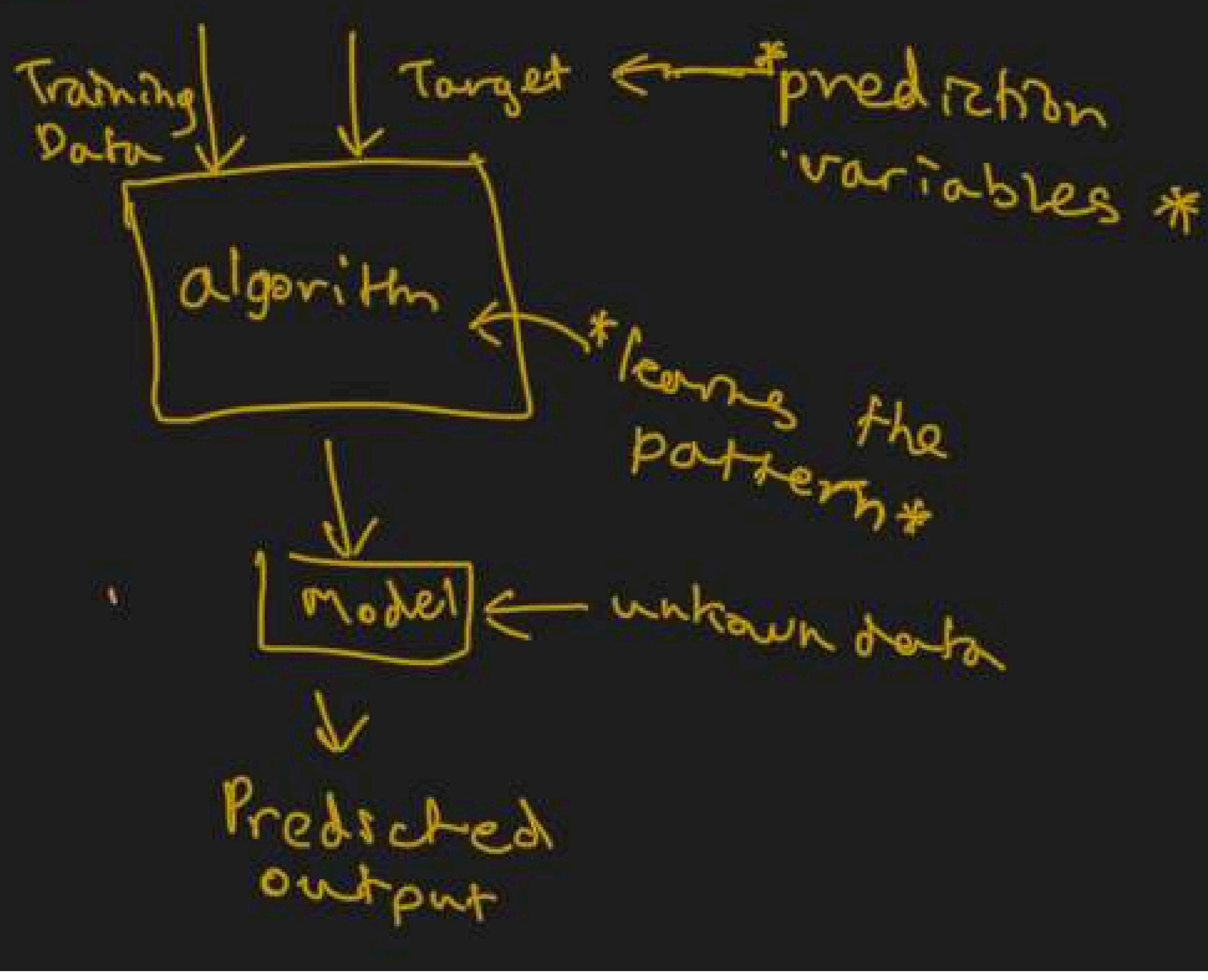
Day 1:

many subsets
like: ML, DL, RL, genetic algorithms

AI = ability to guess. A set of technologies that allows computers to perform functions. that mimic human intelligence.

Computer Program:

AI Model: (* = notes)



What is Learning?

A computer program is said to learn from experience (E) with respect to tasks (T) and measure performance (P) If it's performance at task (T) as measured by performance (P) improves with the experience (E).

(TEP model of Learing)

What is a Model?

A model is the output of the learning process. It is also a mathematical representation of the relationship within the data.

What is a Learning Algorithm?

It is a method/procedure to train a model from data In simple words: Learing Algorithm builds the model and that model is used to predict unkown data.

Types of Learning:

Supervised Learning
↓
Trainable Data with Labels

[label = target output]

Unsupervised Learning
↓
Trainable Data without labels

Reinforcement Learning
↓
No data without labels; instead learing with experience in the environment with rewards and penalties

---

Features

Features, are the individual measurable properties or charecteristic of the data used as an input to the model.

Training

	c_1	c_2	c_3	c_4	Output
r_1	1	2	3	4	
r_2	1	6	7	8	
r_3	10	9	7	6	

Features

Types of Features

1. Categorical features: Represents category /group
   Ex. Blood group, Spam classification, multi-class classification.

2. Ordinal features: Represents category without intervals (thresholds) ranked based
   Ex. Good, bad, large, medium, small
3. Numerical features: Represents number
   Ex. height, weight, temperature
4. Boolean Features: True/False

Splitting data into training, testing and validation sets.

- Training set: The data used to train and teach about pattern to the model
  - The training data should not be imbalanced that is the data should not contain only one type of value, it should be the combination of all types of values.
- Validation set: Testing during training is known as validation
  - The validation set helps fine-tune the model after each epoch (iteration)
- Testing set: set of data, used to test the model after completing training.
  - Provides a final model performance in terms of accuracy and precision

\# Training data > validation data

\# Training data > testing data

Normally # testing data = # validation data

Ex. T: 60% V: 20% Te: 20%

Features	Out
~60% Train	
20% validation	
20% test	

similar/about

sometimes not there thus test being 40%

Types of learning (detailed)

Supervised learning: data with labels. This means that, each training example consists of input-output pairs, where the input is the data that we need to be analyzed and the output is the corresponding label or target value you want to predict.

- 2 types of learning tasks:
  - Regression: Type of predictive modeling technique used to predict continuos values. (There can be an infinite amount of numerical values between 0 and 1.) The main objective is to establish a relationship between independent variables (features) and a dependent variable (target), to predict a future value. Ex. Linear regression.
  - Classification: A task where model predicts the discrete labels (variables) or categories for given input data. The goal is to assign each input sample to one of several pre-defined classes based on it's feature. Ex. Logistic regression.

Bias and Variance

- Bias

Bias is the difference between the actual value and the predicted value.

High bias can not capture the important features of the data.

- Variance

This can be described as the model sensitivity to fluctuations in the data. More specifically, variance is the variability of the model, that is how much it can adjust to the new subset of the training data set

Variance should be minimum and closer to average for same class of data.

Variance should be more for different class of data.

High variance is the signal to change the algorithm

$$\text{Variance representation} = E\left[\left(\hat{y} - E(\hat{y})\right)^2\right]$$

$E(\hat{y})$ = Expectation/mean of the predicted value

$\hat{y}$ = predicted value of the target variable

Over fitting & Under Fitting

over fitting: when model is over trained and tries to learn the details and noise in the training data resulting in high error rates is known as over fitting.

- Noise here is outliers or irrilevent/meaningless data, missing values
- Low bias & High variance
- Low training error & High testing error
- Can also arise due to class imbalencement

Under fitting: Our model is ~~too~~ simple and has very few variables on data points that it considers each of them as a seperate class. which results into high bias, and low variance.

- High training error & high testing error

a = actual value
b = predicted value

bias = b − a $\{\hat{y} - y\}$

Measuring Performance [performance metrics]

Sum^2 error (also known as residual sum of square error)

$$SSE = \sum e_i = \sum_{i=0}^{n} (y - \hat{y})^2$$

Ex.

find out

	x	y	$\hat{y}$	$(y-\hat{y})^2$
$i=0 \rightarrow$	0	1	2	1
	1	3	1	4
	2	2	0	4
	3	5	4	1
$n \rightarrow$	4	7	6	1

given

$\sum = 11$

error

Math ex. $1+2+3+4+5$

$$\sum(5) = \frac{5(5+1)}{2}$$

$\rightarrow 15$

Mean Absolute error

MAE represents the average of the absolute difference between the actual and predicted value and it is shown as:

$$MAE = \frac{1}{N}\sum_{i=1}^{N}|y - \hat{y}|$$

N = # of data points
$\frac{1}{N}$ = mean

x	y	$\hat{y}$	$\lvert y-\hat{y}\rvert$
0	1	2	1
1	3	1	2
2	2	0	2
3	5	4	1
4	7	6	1

*unlike MSE, it measures the average of the residuals

$\sum = 7 \rightarrow \frac{7}{5} = 1.4$

Each data is having an average error of 1.4

* SSE limitation: only measures the variability of the data and doesn't provide information about the goodness of the fit of the model, doesn't generalize model well
SSE always provides a non-negative value which turns rewards into penalties
~prone to under fitting

Mean² Error

Mean² Error

MSE represents the average of the squared diff. between the original and predicted values

* Measures the variance of the residuals. ← diff. gotten

$$MSE = \frac{1}{N}\sum_{i=0}^{N}(y-\hat{y})^2$$

x	y	$\hat{y}$	$(y-\hat{y})^2$
0	1	2	1
1	3	1	4
2	2	0	4
3	5	4	1
4	7	6	1

$\sum = 11 \rightarrow \frac{11}{5} = 2.2$

* lower the error, the more accuracy

Always: MAE < RMSE < MSE < SSE

Root mean² error

RMSE = indicates how close the predicted value are to the actual value

$$RMSE = \sqrt{\frac{1}{N} \sum_{i=0}^{N} (y - \hat{y})^2}$$

$\sqrt{2.2} = 1.48$

from MSE ex.

* shows closeness to the actual value.

Hw: Understand - Matrix inverse, and how to find it
- Transpose
- Matrix multiplication
- Variance & co-variance
- STD
- x and $\hat{y}$
- I

x	y	$\hat{y}$	Calculate all errors
10	1	2	
20	3	1	
30	2	0	
40	5	4	
50	7	6	
60	8	5	
70	8	7	

x	y	$\hat{y}$	calculate all errors
10	1	2	-1
20	3	1	2
30	2	0	2
40	5	4	1
50	7	6	1
60	8	5	3
70	8	7	1

← diffs

$SSE = 1+4+4+1+1+9+1 = \underline{21}$

$MAE = 1+2+2+1+1+3+1 = \frac{11}{7} \approx \underline{1.57}$

$MSE = 21 \div N \to \frac{21}{7} = \underline{3}$

$RMSE = \sqrt{3} \approx \underline{1.73}$

Matrixes

Inverse → $\begin{bmatrix} a & b \\ c & d \end{bmatrix}$

↳ $\frac{1}{ad-bc}[\ldots]^{-1} \longrightarrow \begin{bmatrix} a & b \\ c & d \end{bmatrix}^{-1} = \begin{bmatrix} d & -b \\ -c & a \end{bmatrix}$

Ex. $\begin{bmatrix} 7 & 9 \\ 1 & 1 \end{bmatrix}$

↳ $\frac{1}{7-9} = -\frac{1}{2}$

↳ $-\frac{1}{2}\begin{bmatrix} 1 & -9 \\ -1 & 7 \end{bmatrix}$

↳ $\begin{bmatrix} -\frac{1}{2} & \frac{9}{2} \\ \frac{1}{2} & -\frac{7}{2} \end{bmatrix}$

Transpose: $\begin{bmatrix} a & b & c \\ d & e & f \end{bmatrix} \rightarrow \begin{bmatrix} a & d \\ b & e \\ c & f \end{bmatrix}$

Multiplication:

$A = \begin{bmatrix} 3 & 1 & 4 \end{bmatrix}$ $\quad B = \begin{bmatrix} 4 & 3 \\ 2 & 5 \\ 6 & 8 \end{bmatrix}$

$(1 \times \boxed{3})$ ← have to be equal → $(\boxed{3} \times 2)$

AB?

result will be (1×2)

$3(4) + 1(2) + 4(6)$ ↙ * A by first col of B

$= 38$

$9 + 5 + 32 =$ ← second col

46

Result: $\begin{bmatrix} 38 & 46 \end{bmatrix}$

* Multiply all rows by each column one by one:

cols A (sum)

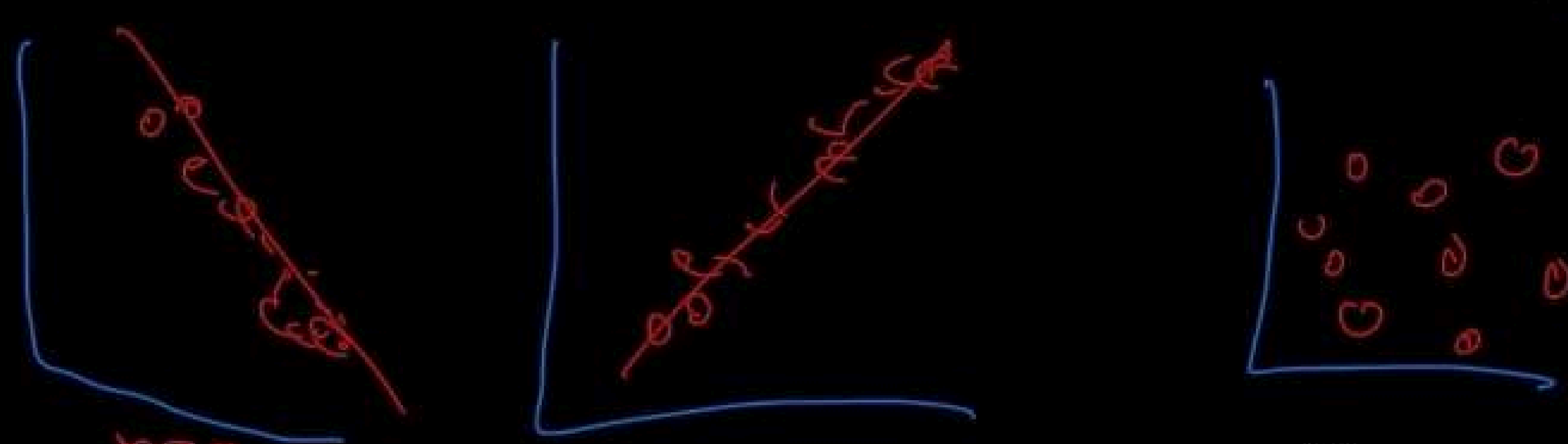
$$\begin{array}{l} \text{1st row} \\ \text{2nd} \\ \text{3rd} \end{array} \begin{bmatrix} 1 & 2 & 3 \\ 1 & 2 & 3 \\ 1 & 2 & 3 \end{bmatrix}$$

Co-variance: how related variables X and Y are related

$$\text{Co-var} = \frac{\sum (x - \text{mean}X) * (Y - \text{mean}Y)}{N-1}$$

positive = both var's change in same dir.
inverse = opp. dir
no relation:

Corr Coefficient
↳ (same thing but between -1 and 1)



STD: (how spread out the values are

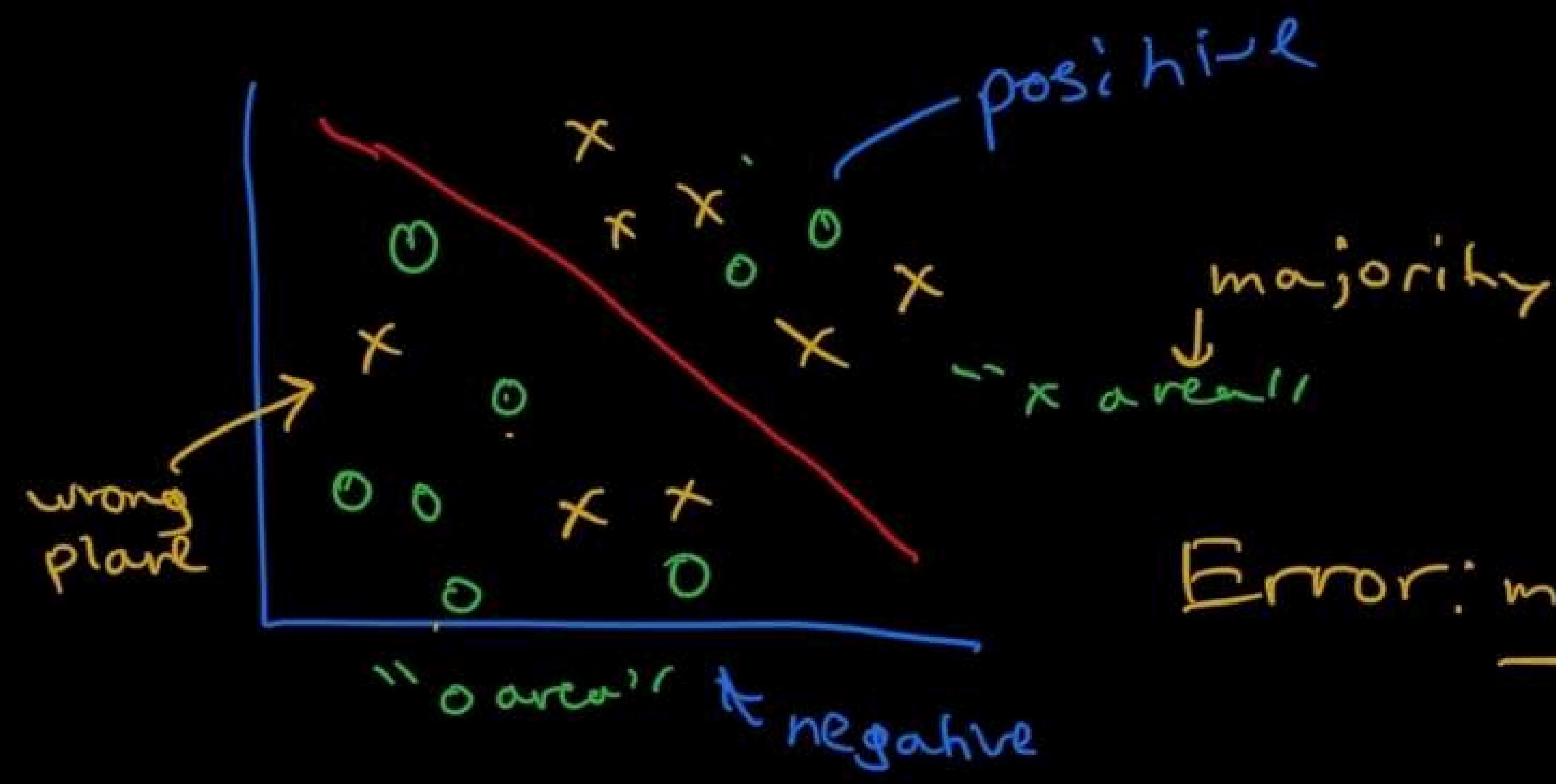
1. Find mean of values
2. Find diffs from mean (value - mean) ← deviation
3. Find all deviation²
4. Sum of squares Σ
5. $\frac{\text{sum}}{n-1}$] variance
   ↗ sample size (n)
6. $\sqrt{\text{variance}}$

] how to find variance

Var = STD²
(Σ)

Classification Performance

Evaluation



Error: $\frac{\text{misplaced points}}{\text{total}} = \frac{5}{16}$

Pos = Plane #1(x):

True Pos = 5

False Pos = 2

Neg = Plane #2(o):

True neg: 6

False neg: 3

ACCuracy = $\frac{\text{correctly classified points}}{\text{total}} = \frac{11}{16}$

Error = 1 − ACC

True Positive = proportion of all actual positive that are correctly classified

False Positive = proportion of all actual negative that are incorrectly classified as positive

True Negative = proportion of all actual neg. that are correctly classified

False Neg = proportion of all actual pos. that are incorrectly classified as neg.

Confusion Matrix: a table used to evaluate the performance of a classification model
(error matrix)

	Actual Pos.	Actual neg.
Pred. Pos.	TP	FP
Pred neg.	FN	TN

Truthness

Ex.

5	2
3	6

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

the proportion of all the model's pos. classification that are actually pos. to everything classified as pos.

Recall = true pos. rate = $\frac{TP}{TP+FN}$

↳ ratio between true pos rate to all pos. samples

* The dimension of the confusion matrix can change as per the number of classes defined