3/11 AI-ML

* GPT-5
* Best method for feature selection
* Agentic AI

* Feature Selection

→ Definable quality or characteristic of an item in dataset.
→ They are variables - bcoz value changes.
→ Independent (input), Dependent (output).
→ Categorize data points in various ways.

★ Goals

1. Identify the most imp input variables so that the model can effectively predict dependent variable.
2. Best feature selection ensures model performs accurately using the most relevant features.

★ Target Variable

1. Target (dependent) var. is what the model is trained to predict
2. Input (indep.) var. helps in predicting this target.

⁉ Why feature selection is imp?

- Helps determine which model uses fewer parameters but still performs efficiently.
- Reduces overfitting, training time, and noise in the data.
- Improves model interpretability.
- Research Point: How to optimize and reduce the feature selection process.

Calculate multilayer perceptron layer - mathematically
• Softmax
• ReLU

dimentionality Reduction

Feature Engineering
- Selection
  - Supervised
    - Filter (used for doing pre-processing step). Statistical Technique → Model Independent → Interpret features → XAI (+ why?) | Feature Selection → Before training model. → diff.
      - Variance (numeric) $\frac{1}{2}\Sigma(x_i-\bar{x})^2$
      - Correlation (Cont.-num) $r = \frac{Cov(X,Y)}{\sigma_x \sigma_y}$
      - Fisher (Cont.)
      - Chi-square (Categorical/Non-ve) $(\chi^2 = \Sigma \frac{(O-E)^2}{E})$
      - ANOVA. (Cont + Categorical) $(F = \frac{\text{Btw class variance}}{\text{within class variance}})$ F-test
    - Embedded
      - Regularization (L1, L2)
    - Wrapper
      - Forward
      - Backward
      - Exhaustive
      - Recursive
  - Unsupervised
    - Filter
      - Variance
      - Correlation
      - PCA
    - Hybrid
    - Wrapper
- Extraction
  - Positional Embedding
  - PCA
  - Word 2 Vector
  - CNNs.

What if we create a model which helps to reduce features selection & reduces noise, redundancy & irrelevant features

New formula for new features

• Who will win?
↳ Agentic AI or
↳ Core ML ?

✱ Image Feature Extraction

→ Kernel (matrix - will work on tensor)
↳ 32, 64 (1 Kernel box moving & taking only imp features which are relevant to use for model training)

✱ TF-IDF - for extracting features from text.

Que. Diff b/w feature selection & feature Extraction.

Algorithm :- Takes data as input & train the data for model.

Model :- Trained Algorithm

- Size of model depends on features & not data size.

→ Time & Space Complexity.
- SLMs.

🔍 Difference Between Feature Selection and Feature Extraction

No.	Feature Selection	Feature Extraction
1.	Selects a **subset of existing features** from the original dataset.	**Transforms existing features** into new ones.
2.	Does **not alter** the original features.	Creates **new features** by combining or transforming the old ones.
3.	Goal: **Choose the most relevant features.**	Goal: **Reduce dimensionality** by forming new, compact representations.
4.	Works by **removing irrelevant or redundant features.**	Works by **projecting data into a new feature space.**
5.	Example methods: Filter, Wrapper, Embedded, Hybrid.	Example methods: PCA, LDA, Autoencoders, Word2Vec, TF–IDF.
6.	Easier to interpret because original features are retained.	Harder to interpret as new features may not have clear meaning.
7.	Maintains the **physical meaning** of the data.	May **lose the original meaning** of variables.
8.	Mainly focuses on **relevance and importance** of features.	Focuses on **representation and compression** of data.
9.	Less computationally intensive compared to extraction.	More computationally intensive due to mathematical transformations.
10.	Often improves model performance without changing data structure.	Changes data structure entirely, creating new feature dimensions.
11.	Can be applied **before or during** model training.	Usually applied **before** model training.
12.	Examples: Removing correlated columns, using feature importance scores.	Examples: Using PCA to cre principal components, Word2Vec embeddings in NLP, CNN layers
13.	Useful when you have **many features but want to keep interpretability.**	Useful when you have **high-dimensional data** and want compact representation.

- Removes irrelevant feat.
Chi-space test
Correlation filtering
-ve
+ve

Q Benefits of feature selection.

1. Impactful
2. Removes Redundancy (size, time, complexity)
3. Improves Efficiency
4. Boosts Performance.
5. Better Recall
6. More Accurate
7. More Precise
8. Data features affect how data models configure their weights using training which drives performance.
9. Reduce Overfitting
10. Reduce Comp Cost
11. Smooth Implementation

a. Data Cleaning
b. Extraction
c. Selection

Uni Varient – (ANOVA, Chi-square, Fisher Score)

Multi Varient –

Pairwise Filter – Correlation b/w target & other

Kaggle
↳ data has to be injected.
↳ Then model is trained
↳ Ask is customization of model.
↳ How ML work - include data pre-processing.

- ★ Telecomm. Pentesting
- ★ 5G. Tech.
- ★ SS7 Protocol

Filter Method

- Univarient
  - (Varience, Chi-sq. ANOVA, Fisher)
  - evaluate each feature indiv. with respect to target variable.
- Multivarient
- Pairwise
  - (Pearson, Cramer, Dist Correlation)
  - ↳ Evaluate relationships b/w pairs of features or b/w feat. & the target
  - ↳ Multi-colinearity (feat-feat redundancy)
  - ↳ Visualize with heatmap.
  - ↳ Complexity ↑ses.

ANOVA :- If features values are similar across all classes, it's not useful. If a feature's values differ significantly across classes, it's impt in classification.
If variance is high – it is good.

Standard Scalar
↳ Converting huge or larger value into ↓ smaller one

Import lib (numpy, panda, sns, matplotlib)
Import dataset (raw)(load)

A. data preprocess.
- Cleaning → Removing null, Adding missing values
- Statistics (can also visualize here)
- Convert into numerical value ↑ categorical (Some algo can't digest categorical value)
- Standard Scalar (mean | max kinda thing)
- Visualize. (SMOTE)

A.1 Feature Selection.
- filter
  - Mini Variant
  - Pair Variant
- Wrapper
- Emb.

A.2 Splitting the data →

Training (70%) dataset		Testing (30%) dataset	
Training X	Testing X	Training Y	Testing Y
70% rows.	30% rows.	70%	30%
700	300	700	300

feature part | Ans part
X | Y
↓ picking & storing only features
[:-1]
Y → [-1]

★ Data Pipeline
★ Hyper parameter

↓
Fun^n → {Algorithmobject.fit}
eg; dst.fit(x_train, y_train)

Accuracy → predict fun^n → x_test & y_test value.
↓
Precesicion → Recall → F1. (Evaluation Matrices ★ Confusion matrix → Both Logistic & Linear)
↳ will def. come in exam

Dump the model (pkl file).

★ One can take 3 or 4 algorithm for training
If you want more algo then you have to do ensemble learning.

★ Hyperparameter point.

20/11.

SVM - Vector - Supervised. 1995 - Vladimir N. Vapnik.

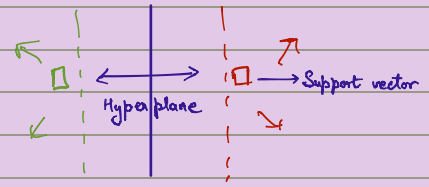
↳ data points.

↓ Classification

↓ Regression

→ Find the best boundary - Hyperplane.
→ maximize the margin b/w 2 classes.

→ Classification - find optimal hyperplane - maximize margin b/w closest datapts.
line.
→ 2D - space. or a plane in n-dimensional space.
→ Multiple hyperline can be found. - Choose the best.
→ Lines adjacent to optimal hyperplane are called support vector. & these vectors run thru the data points that determine maximal margin.



- Why maximise?
- Diff b/w linear reg & SVM.

SVM uses Kernel / filter

↓

Can handle both linear & non-linear.

↓

difficult

so ↳ use Kernel-trick. (transformes higher dim. space to enable linear seperation)
2D → 3D.

↓

a. linear
b. Polynomial
c. Radial basis fun (RBF).

for hyperplane : $y_i (w \cdot x + b) = 0$, seperates +1 & -1.

↙ ans. ↓ weight ↓ input ↓ bias.

(2,2) A (4,4) B (4,0) C

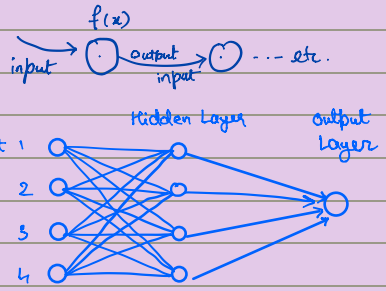
Bias

$b = -1 - 4w_1$ judging Lesson

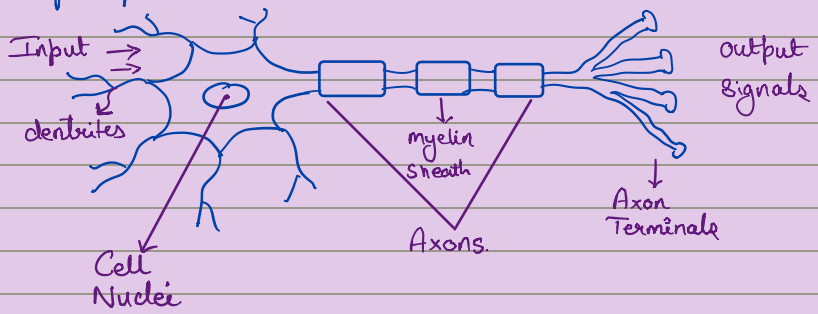
Margin Cal - $\frac{2}{(|w|)}$.

21/11

Artificial Neural Network & Back Propogation



perceptron - 1 neuron - calculation can come in exam



★ playground.tensorflow.org.

- Noise helps to generalise your model.
- Test data (around 60 to 80 %.)
- Epoch - iteration
- Learning rate - should be manageable
- Activation Fun - 'if' condition. (ReLU, Tanh, Sigmoid, Linear)
  - ↳ helpful to find non-linear pattern
  - a. ReLU - max fun ⇒ -ve = 0 +ve = that value
  - b. Sigmoid - $x_1 \times w_1 + x_2 \times w_2$ (boundary 0 to 1)
  - c. Tanh - one neuron to another layer (-1 to 1 is boundary).
- Combining neurons at a Single time is called layer.
- Perceptron calculation - single
- ANN - collection of perceptrons
- Multi-layer perceptron.
- Hidden Layer
- Dropout Fun - forget that specific neuron/task.
- Regularization - Punishment to the model
  - ↳ Reg. Rate -
  - Types → L1
  - Types → L2
- Global minima, maxima
- Separately define activ fun for diff layers, if u want.

Classification
- Binary → Sigmoid
- Multi-class → Softmax (output) (Layer).

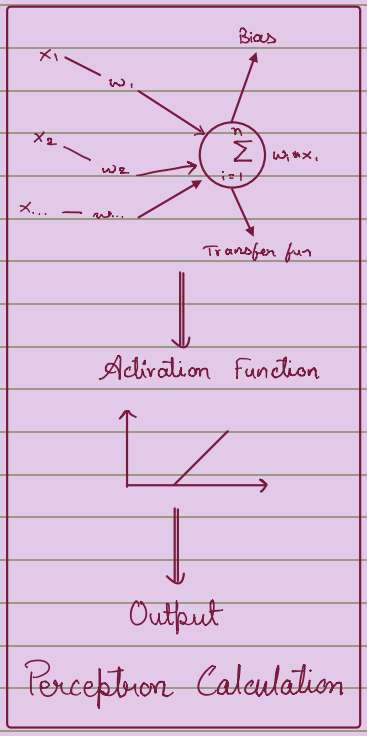
Non-linear statistical model → Complex relationship b/w inputs & outputs

Few shot learning
Zero shot learning

Random function approximation.

Input (x1 = 0 or 1) weight ... } given

1. Threshold value = 1.5 Can be anything.
2. $x1 * w1 + x2 * w2 + x3 * w3$
3. = Z. (1.4)
   - Sum of all
   - ↳ If it is more than ur threshold value then activation fun is done.
   - ↳ Type of acti. fun would be given
4. Activate the output ⇒ Apply activation fun.



★ Steps.
1. Randomly assign weight
2. Give Dataset
3. Check
4. Feedback (loss fun)
5. Adjust weight value.
6. Testing

Sigmoid eqn $f(z) = \frac{1}{1+e^{(-z)}} = \frac{1}{1+e^{(1.4)}}$
↓ upar vale ka ans

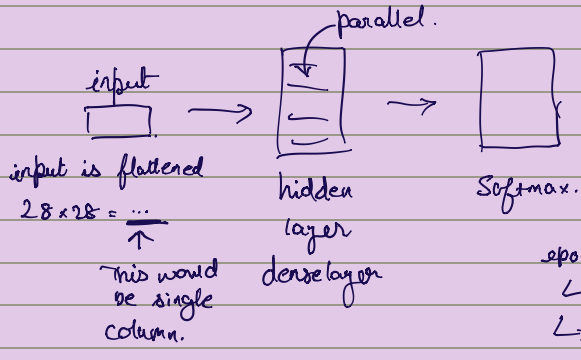
net model

ANN = Feed forward = Multi-layer perceptron

perceptron = 1 neuron
layers = multiple neurons

Hidden layer = ReLU (binary +ve)
Riki ReLU (-ve x -0.1)

Multi-Class Classification = Softmax
(0, 1, 2, 3, 4)



Alexnet
VGG

parameters should take less value.

epoch.
↳ accuracy will ↑
↳ loss fun will ↓.

Loss fun → single record

Cost fun → avg loss of all records

Optimizer (for loss fun).

Diff b/w ANN & CNN

Why CNN is better?

sgd. = Stochastic gradient descent (optimizer)

Sparse categorical cross entropy → for multiclass classification.

runpod.ai

Training = GPUs.
↳ model.fit

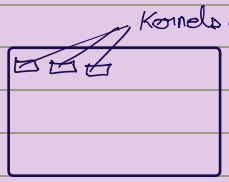
26/11

CNN = ANN + polling
(max, avg)

3 Channel - image
RGB

CNN
↓
Conv2D, MaxPooling

RGB → feature map → ReLU → Output
(Kernels)



Feed forward
↓
No memory

ANN
CNN ANN LSTM

RNN = Recurrent Neural Network
(It remembers only current flight)

Current Time = T past = T-1
future = T+1

The RNN model hallucinates more
(less context window)

Single-time step of input is provided
↓
Calculate current state (using current input) (ht)
& the previous state (ht-1)
↓
Final current state is used to calculate output
⇓
[Back Propogation means if any error occurs then it goes back (it recalls) & then it corrects the ans]

↳ Unidirectional ht-1 — ht — ht+1
↳ Sequential * Evaluates current input + learned from past input.
↳ Has memory

x → Input	O → Output
h(t) → hidden state (memory)	V → comm. from one time-step to other. (weights)

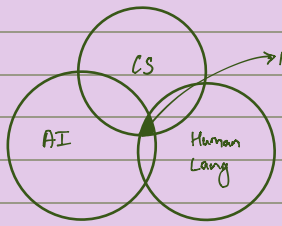
h(t) = f(U x(t) + W h(t-1))

- Backward Pass
- Forward Pass.

CNN - feature extr. dimensi
RNN - seq. pattern
is no parallelism

LSTM
Long-short term memory
↓
It has gates → which forgets any specific state & even remembers the important thing.

* NLP :- Natural Language Processing



nltk.

* Pipeline.
↓
Make ur own GPT.

* Titan.

* BERT
↓
Bidirectional
↓
Parallel Processing

⇒ One Hot Encoding

⇒ TF-IDF (Term Frequency - Inverse Document Frequency).
- TF: No of times a word came.
- IDF: importance of word

⇒ Bag of Words = Word-to-Numbers (tokens).
↳ It's a technique in NLP.
↳ Each doc is treated as bag or collection of words & then count the frequency of each word in doc.

① Word Ambiguity
↳ Resolving can be a research paper
- Go to the bank & take out money
- I went to narmada bank.

★ Encoder ★ Decoder → for lang. translation.

Input Embedding

Your	Cat	is	a	lovely	cat
↓	↓	↓	↓	↓	↓
105	1687	548	357	66	576

attention words here → Query, Key, Value

In RNN → One word goes & tokens are made → Sequential

Attention (GPT). → Query × Key = Scores ⇒ Takes whole para at once.

Multi-headed Attention (Multiple attentions together)

$$\text{Attention}(Q, K, V) = \text{Softmax}\left(\frac{K^T Q}{\sqrt{d_K}}\right)V$$

★. Unit 4

Computer Vision :- Subfield of DL & AI where humans teach computers to see & interpret world around them.

- 2 Swedish sci: Hubel & Wiesel - placed a cat in a restricting harness & an electrode in its visual cortex.

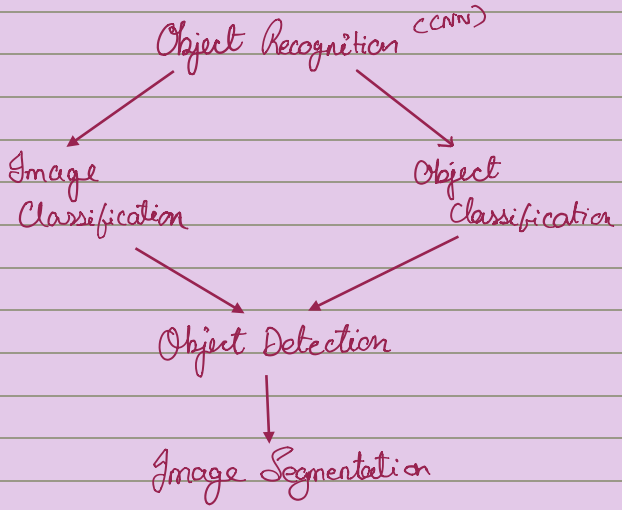
2001 - Face detection

2012 VGG, Alexnet

Murder - pic - 2D - 3D - NL

Detection

Recognition



HOG feature Extractor & SVM
↳ Better before CNN.

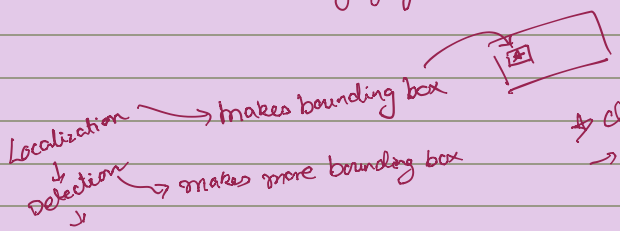
★ Challenges

finding weights & vectors from pic

SIFT (Scale - Invariant Feature Transform)

ORB (Oriented FAST & Rotated BRIEF

Viola - Jones Algorithm - Face detection } open CV.
↳ Haar - like feature extraction }
↳ Blinking of eye.



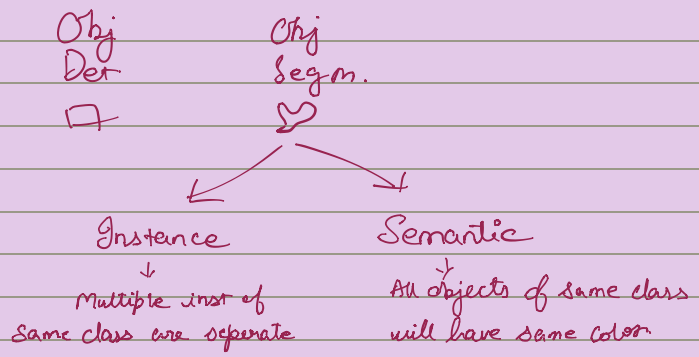
Localization → makes bounding box
Detection → makes more bounding box

* challenges
→ Always rectangular.

Image Segmentation

↳ Marks the presence of object through pixel-wise masks generated for each object in the image

Mask R-CNN = Detecting cancer.



★ Applications of ML

★ Challenges of ML

★ ML Types

A] Supervised

Linear Regression
- Best fit line
- $Y = B_0 + B_1 X + \varepsilon$ (single & multi)
- Errors → RMSE
  - ↳ MSE :
  - ↳ R^2 :
  - ↳ SSE : Sum of squared errors
- Adv.
  - Fast
  - Good for Linear Rela.
  - Quantifies Relationship
- Disadv.
  - Only supports linear Rel^n
  - Sensitive to outliners
  - Not for complex data

Logistic Regression
- Classification
- Categorical outcomes / discrete values (0 & 1)
- Sigmoid fun.
- $S(z) = \frac{1}{1+e^{-z}}$ ← (linear eq)
- Compare with threshold
- Evaluation : F1, acc, pre., recall
- Types
  - ↳ Binomial
  - ↳ Multinomial
  - ↳ Ordinal
- Advantages
  - Probabilistic output
  - Simple, Efficient
  - Well for linearly seperable data
- Disadvantages
  - Assumes linear relationship
  - Sensitive to outliners
  - Requires independent features
- Error : MLE (Max likelihood estimation)
  log loss

B] Unsupervised

Hierarchical Cluster Analysis (• Maths.)
- Group unlabelled datasets
- Tree-shaped structure – Dendrogram
- No predetermination of pre-defined clusters (unlike K-means)
- 2 Approaches → Agglomerative : bottom-up (all data-points as single clusters & merging them until one cluster is left)
  → Divisive : top-down

⊙ Steps : Data Preprocessing
1. Distance Matrix Calculation : Euclidean dist. – (pair wise dist b/w all data points)
2. Cluster Initiation → 3. Iterative Merging or Splitting
4. Linkage Criteria Selection → Single, complete, avg (how dist b/w clusters are calculated)
5. Stopping Criterion : • max cluster • max threshold • max granularity
6. Dendrogram Generation
7. Cluster Extraction (choosing desired level of granularity – cutting dendrogram) – low cut – more cluster (fine grained) – high cut – few (coarse-grained)

• Advantages	• Disadvantages
→ Easy → Flexible Clusters	→ Sensitive to dist & linkage
→ No predefined Clusters	→ Comp. expensive → No reverse
→ Interpretable dendrogram	→ Not for large datasets or real-time
→ wide range of data types (numerical categorical)	

K-Means Clustering
- Group similar objects into a cluster.
- Clustering means dividing entire data into groups based on patterns – Not prediction but club similar observations

Decision Tree

- For both class & reg. (more)
- Splitting data into branches based on features. (Recursive Partitioning)
- Attribute → Deci. Rule → Output
- Algo

↳ ID3 - Info. Gain (works for categorical values)

↳ C4.5 - Gain Ratio (Contin. variable)

↳ CART - Gini Index (both classif. & regression)

a. Entropy – Measure of uncertainity or randomness
b. Info. Gain – ↓ in entropy after data is split. ↑IG ✓
c. Gini Index – Correct variable for Splitting nodes.

- Advantages
  - — Handles non-linear relⁿ
  - — Easy, simple, fast
  - — Feature imp. analysis
  - — Both categorical & numerical
  - — little data processing
- Disadvantages
  - — Greedy — Overfitting
  - — Instability — Bias to dominant class.

Loan, Marketing, Medical

- Best attribute is selected by ASM - Attribute Selection Measures
- → Evaluation : Accuracy, Precision, Recall
- → Stopping criteria : • Tree length • nodes belongs to same class • features done.

K-Nearest Neighbour

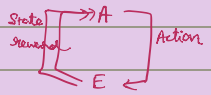
- Reg + Class (more)
- Similarity b/w new data & available categories
- Non-parametric (no assumptions)
- Lazy learner - bcoz it stores the dataset & at the time of classification, it performs action.
- K-selection methods
  - • Not small (overfit) • Not big (underfit)
  - • Always odd • $\sqrt{N} = K$ • Select range
  - • Error vs Accuracy map • Mean Error Rate
- Euclidean formula

Naive Bayes

- Text classification (High dimensional Training dataset)
- Probabilistic - fast, simple
- Considers every features independent of each other (Naive) + Bayes theorem
- $P(A|B) = \frac{P(B|A)\ P(A)}{P(B)}$ (Posterior) (likelihood) (Prior) (marginal)
- Binary + Multi-class classification
- Dis - Assumes all features are independent.
- Gaussian, Multinomial & Bernoulli - Types
- Credit Scoring, real-time predictions, medical data classification

Reinforcement Learning

- Optimal behaviour to obtain max reward
- Absense of supervisor, maximize the rewards → trial & error search
- Depends on how delayed or immediate it gets rewarded
- Agent, Environment, State, Action, Reward ⇒ Components



- Exploration (high at start) e-epsilon
- Exploitation (risk). (↓ess over time) wants cumulative reward.

co2 agent

- Core Components :
  1. Policy :- Defines model behaviour
  2. Reward Signal :- Feedback
  3. Value Function :- Long term benefits
  4. Model :- Simulates env. to predict outcome of actions

Usage: Robotics, Autonomous vehicles, Gaming, Finance, Healthcare, NLP, Industrial Control System.

- Types
  1. Online – direct interaction with env.
  2. Offline – not direct int with env. – learns through logged data.

Support Vector Machine

- Reg + Class (more)
- Create a best line / decision boundary (hyperplane) that can segregate n-dim. space into classes
- Extreme vectors – support vectors
- Types → Linear – divided into 2 classes
  - → Non-linear – can't divide by line.
- Best line selection – margin & best seperation

NLP

↳ Subfield of ML

↳ Helps to analyse, generate & understand human generated lang. to derive meaningful insights.

Text - Preprocessing

↳ Tokenization - breaking down complex piece of text into smaller units called tokens

↳ Stemming - converting words into its base or root form. - removes suffix.

↳ lemmatization - more sophisticated includes grammar & part of speech.

Text Feature Extraction

1. Bag-of-words / Vocabulary - Set of unique words
2. n-grams - n items - +ve integer - context of words in text (uni, bi, tri) irrespective of grammar & order
3. Count vectorization - text → numerical (it keeps tracking no of occurrence of words).
4. Word Embeddings

⇒ Text Sort : assigning text to predefined categories

Named Entity Recognition

↳ Identify & extract named entities like name, org, location, date

↳ Working

a. Tokenization
b. Entity identification
c. Entity classification
d. Contextual analysis
e. Post-processing

Part-of-Speech Tagging (POS) → Categorize words based on their grammatical fun.

↳ Identifying parts of speech of words ⇝ NLTK lib in python. - pos_tag method

Noun (NN) Ajdective (JJ) Article (DT) Conjuction (CC)
Verb (VB) Pronoun (PRP) Adverb (RB) Preposition (IN)

↳ Subject - does action - simple, compound, complex.
Object - on which action happens
Modifier - which describe or modify subject

Text Generation

↳ Generating new text based on given input or model. Eg; machine translation, text summarization chatbots, content creation, lang. translation

* Techniques

1. Markov Chains: predicts next word based on previous input
2. Seq2Seq: 2 RNNs - encoder & decoder - lang translation
3. GANs: Generator & discriminator
4. Transformers: Attention mechanism - GPT, BERT, Translation, Summarization

Text-to-Speech

• 2 modules - text analysis module & speech synthesis module

Speech-to-text

2 Modules: Acoustic models & language model - Virtual assistance, dictation & Automatic Speech recognition (ASR)

Computer Vision

↳ Subfield of DL * Hubel & Wiesel

↳ Humans teach computers to see & interpret the world

1956	1959	1960s	1970s	2001	2012
AI	Neurophy.	1st comp. vision	AI winter	Face Detection	Alexnet, BERT

Object Recognition

↳ Technique of identifying the object present in the image or video

↳ Recognize the content of the image

HOG
SVM
SURF
SIFT
ORB
Viola-Jones algo
CNN

Image Classification

↳ Takes the image & classifies it into class.

Object Localization

↳ Locates the presence of the object in image & represents it with bounding box.

↳ (position, width, height)

Object Detection

↳ Mix of IC & OL – takes image as input & makes bounding boxes with class label

↳ Deals with multi-class & multi-occurrences

* Challenges
  a. Boxes always rectangular
  b. Detecting parameters are difficult

Image Segmentation

↳ Detects presence of an object pixel wise masks generated for each object

↳ Shapes ✓ → medical, satellite

2 Types

Instance
- Multiple instances of same class are seperate segments

Semantic
- All objects of same class forms a single classification

Applications

- Driverless car
- Surveillance & Security
- Medical Image Processing