07/01/26

Machine Learning : Day 1

1) What is Machine Learning?
- ML is learning from data.
- ML does not think like humans
- ML does not understand logic or meaning
- ML works on numbers & patterns only
- Main Goal → Reduce prediction error.

2) What ML learns from data?
- ML learns correlations, not causes
- Learns patterns between input (x) & output (y)
- Uses past data to predict unseen data.
- ML optimizes error, not truth.

3) Why I am Taking Finance as context?
- Finance data is structured
- Clear outcomes (approve / reject, profit / loss).
- Many variables affect decision
- Easy to reason & validate result.

4) ML structure (Problem statment)
- Feature (X): Input columns
- Target (Y): output column
- Objective : (Depends on situation)

Example : X: Income, age, credit score
Y: loan approval (Yes / No).

5) What is EDA used for?
- Understand data distribution
- Find outlier & noise
- Identify useless column.

- Understand data Limitation
- EDA tells what data can & cannot teach model.

6) Relation between EDA and Model?
- EDA does not choose model directly
- EDA shows:
  (i) risks
  (ii) constraints
  (iii) possible failure
- It helps to Ans: "where will model fail".

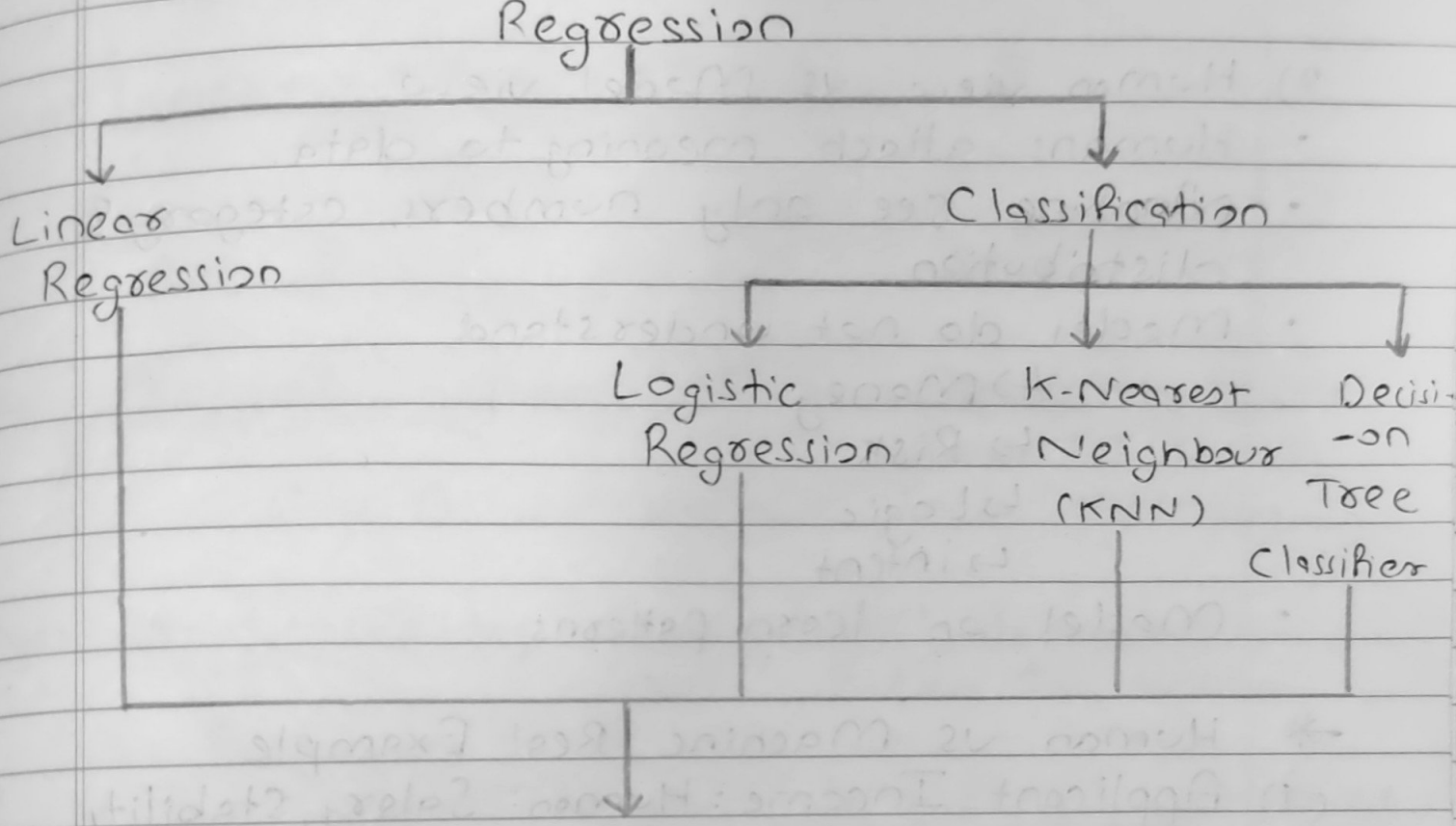
7) What is a good model?
→ A good model:
- Generalizes good on unseen data.
- Is stable under small changes
- Is not sensetive on noise

→ A good model is not:
- Highest accuracy
- Most complex model.

8) Common Mistakes
- Chasing accuracy / R^2 blindly
- Ignoring Preprocessing → core work
- Encoding is boring
- Trusting model blindly
- Assuming model blindly

* Summary
(i) ML learn patterns, not causes
(ii) EDA defines data limit
(iii) Understanding > Accuracy
(iv) Model optimizes error, not truth.

* Machine Learning Algorithm & Process



8/1/26

Data Understanding : Day 2

1) What is a dataset?
- A dataset is a historical snapshot
- Each row = information captured at one moment
- Data does not represent full reality
- For ML to predict this enough but irl its not complete.

2) Human view vs Model view
- Humans attach meaning to data.
- Machine see only numbers, category & distribution.
- Model do not understand
  - ↳ Money
  - ↳ Risk
  - ↳ Logic
  - ↳ intent
- Model on learn patterns

* Human vs Machine Real Example

(i) Applicant_Income : Human: Salary, Stability
Model: Numeric value
Risk: Model trust income blindly, ignores job stability.

(ii) Coapplicant_Income: Human: Family support
Model: 0 vs Non zero
Risk: Overvalues presence of coapplicant

(iii) Loan_Amount: Human: High Amt = High Risk
Model: Numeric & related with Income.
Risk: Double counting Income information.

(iv) Loan_Term: Human: EMI flexibility
Model: Low variance Numberic.
Risk: adds little useful insights

(v) Gender / Married / Education:
Human: Social factors
Model: Encoded labels
Risk: Biased decisions.

(vi) Property_Location: Human: Opportunity resale
Model: Encodel labels
Risk: Baised decisions, False ordering.

*(vii) Credit_History: Human: Past Behaviour
Model: Binary Prediction
Problem: Bank uses credit history
(ii) Data records decisions influenced by it.
→ So Model learns from data weather 'Loan Approved / Not' based on credit score.
Model learns Policy, not risk.
→ By doing this well get accuracy but model will be at risk.

(viii) Loan_status: Human: Approved / Not depending on all features
Model: Correlation & finding pattern
Risk: Wrong learning == Wrong outcome.

Page No.
Date

* Why High Accuracy can be dangerous?
→ Features that produce High Accuracy may
(i) Dominate model learning (Loan_status: Yes/No
(ii) Hide weaker but meaningful insights
(iii) Encode bias.

Note: High Accuracy ≠ good / Reliable model.

* Summary:
⇒ Data understanding is about knowing what the model is learning & what is it blind towards.

8/1/26

EDA : Day 2

1) What is EDA actually for?
- EDA is not for plots.
- It is used to understand data behaviour.
- EDA helps to predict how Model will react.
- Goal of EDA → Reduce risk.

2) Why EDA & How it Answers?

With EDA	Without EDA
• Target can be predicted from data. • Where will model fail? • Noise & dominant features. • Mistakes to avoid.	• Model results are misleading. • Accuracy can lie • Failures are unexpect-ed.

* Analysis on Real Dataset

① Target variable Analysis
- Always Analyze target first
- Check: Majority vs Minority (Balance)
- If target is unbalanced : Accuracy unreliable

→ Our Analysis : "Loan_Status".
- Loan given Yes : 68-70%.
  Loan given No : 32-30%.

→ This gives clear idea that data is unbalanced & it will predict Yes more often.

(2) Numeric features vs Target
- EDA checks:- Overlap between them
  - Presence of outliers

→ Our Analysis: Credit_History vs Loan Status. When Credit history is 1.0 (Yes) more loans are given as compared to 0.0 (No). This is leakage risk as it changes model behaviour.

(3) ApplicantIncome vs Loan Status.
→ We analyzed that the person who has income of 15000. So there is a possibility that he might get loan / he might not. There is no biasness here. This destroy our belief: High Income = Loan given.

(4) Property_Area vs Loan_Status.
→ Our Analysis says semi Urban have been given more loan. But is it true, No future trends will change the belief.

* Concept of Overlap
(i) If loan Approve & rejected values overlap heavily then no clear decision & struggles.

(ii) Model learn from seperation.

8/1/26

Project 1: Email Spam Classifier

* Dataset: Spam_ham.csv
  Size: 5000+ rows
  Columns: (i) Label - (spam, ham)
  (ii) text
  (iii) Label_num - (0, 1)

* Structure

```
app/
  └> app.py
model/
  └> m1.pkl
     m2.pkl
training/
  └> train.py
  └> csv file
```

* Understanding Project.

(1) Training → train.py

(i) Import necessary libraries: nltk, stopwords, PorterStemmer, RandomReg

(ii) Load dataset

(iii) Setup portstemmer & stopwords

(iv) Processing text: Lowering text
Removing nlts & spaces
& performing (iii)

(v) Create a vectorizer for Countvectorizer

(vi) X & y assigning

(vii) Traintest split

(viii) Create model RandomForestRegressor & fit

(ix) Save the model in joblib.

9/1/26

Data Cleaning & Feature Handling : Day 3

(1) Purpose of Day 3

- Convert raw data into model ready data.
- Decide what information model should trust
- Preprocessing = Part of modeling

(2) Missing values handling

- Missing values are information, not errors
- One strategy doesn't work for all.

* Categorical Columns

(i) Fill with mode

(ii) Reson: Preserves category distribution

* Numeric Column

(i) Fill with median → Loan Amount
  ↳ Data is Skewed
    Outliers are present
    Mean will dissort values

(ii) Fill with mode → Loan_Amt_Term (Duration)
  ↳ Most loan duration is (240-360)
  ↳ makes sense.

* Special Column

(i) Fill with 0 → Creclit History
  ↳ Missing means No credit in past
  ↳ Missingness is signal.

- No Universal Missingness value strategy.

(3) Target Encoding (Loan_Status)
- Values Y, N
- Encoding: Y → 1
  N → 0

Reason:
- Binary Outcome
- Mapping is clear & safe
- Better than hidden LabelEncoders.

(4) Categorical Feature Encoding
- Encoding categorical values
- Reason:

(i) No natural order
(ii) Prevent Fake ordering

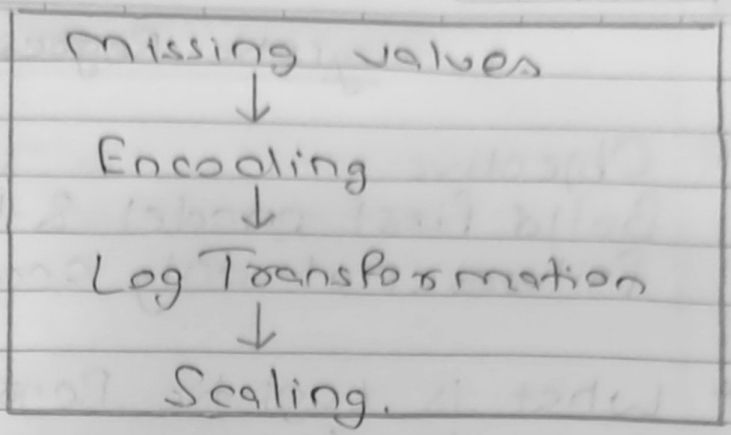
(5) Log Transformation (log1p) *Important
→ Application done to →
  (i) Applicant_Income
  (ii) CoApplicant_Income
  (iii) Loan_Amount

- Reason

(i) Compress Extreme values
(ii) Reduces dominance of outliers
(iii) Log fixes shape.
(iv) If data is skewed prefer log1p once.

(6) Scaling → (i) Linear model are sensetive to magnitude.
  (ii) Scaling ensures all features contribute comparably.

* **Workflow** –



(7) Feature Cleanup

(i) Removed: LoanID & RawIncom, Loan as we Kept Log versions

(ii) Converted: Boolean columns → 0/1

- Reason:
- Avoid duplicate info
- Prevent leakage
- Ensure model stability

———— X ———————— X ————————

9/1/26

Logistic Regression - Day 3

* Objective
Build first model & learn how to judge its failures, not only accuracy.

* What is Logistic Regression
→ Computes Linear combination of Features
Output between 0 & 1.
Value represents $P(y=1 / x)$

* Coefficient Interpretation
(i) Positive coefficient - increase approval
Negative - decrease approval
Large value - strong value

(4) Confusion Matrix

	Predicted No	Predicted Yes
Actual No	TN	FP
Actual Yes	FN	TP

- Accuracy → (TP + TN) / Total
- Precision → TP / (TP + FP)
  ↳ when model says 'Yes', how often it is right.
- Recall → TP / (TP + FN)
  ↳ out of all positive how many we caught.

Note: (1) Precision ↑ Recall ↓

(2) Precision ↓ Recall ↑

(5) Error & Business Risk

In banking:

(i) FP is most dangerous
↳ Loan given to risky applicant
↳ Financial loss

(ii) FN (Missed opportunity)
↳ safer than FP

∴ Precision > Recall in loan approval.

* Note: Model evaluation is a business decision, not just math.

10/1/25

Decision Thresholds & Risk : Day 4

(1) What does Logistic model give?
- It does not give decisions
- It gives probablity.
- Model only how confident it is.

⇒ Decision are made using threshold.

(2) Decision threshold.
* Threshold = cut off to convert probablity → Decision

- Rule:

(i) if problity > threshold → Approve (1)
(ii) else → Reject (0).

- Important

→ 0.5 is arbitary
→ It has 0 busines value

(3) Precision & Recall business view
- Precision
  → When we approve how often we are correct
- Recall
  → Out of how many good customer, we approve.

(4) Hands on result

Threshold	FP	Precision	Recall	Behaviour
0.5	24	0.80	0.92	Risky
0.7	15	0.85	0.79	Balanced
0.8	4	0.93	0.47	Strict

⇒ 0.7 is best choice.

15/1/26

Day 6: Model Evaluation (Industry)

* Note
(i) If False Negative are worse → we focus on recall.
(ii) If false positive are worse → we focus on Precision.

1. Why is model evaluation important?
→ Model evaluation is used to decide weather a model deployed or rejected.
Wrong evaluation leads to:
(i) Financial loss
(ii) Risk exposure
(iii) Wrong deployment decisions

2. Problem context
→ FN (False Negative) is most costly to bank (Bad customer approved → loss to bank).

3. Baseline model
→ A simple run used as minimum performance model.
The model says predict all customer as BAD

Result:
→ FN: 0 ; Busines: 0;

* Any model we make must perform better

4. Dataset Splitting
→ The method to split real-world usage & avoid leakage.

(i) Train set → Model learning
(ii) Validation set → Threshold tuning & comparision
(iii) Test set → Final.

* Stratify=y ; This is used to save data from imbalance.

5. Confusion Matrix
TN - Good customer approved
FP - Good customer rejected
FN - Bad customer approved
TP - Bad customer rejected

6. Threshold tuning.
→ Default 0.5 is not optimal of buisness problem.
• predict_proba → Used to predict how bad is something.

* Final verdict
→ The model reduces bad loan approvals significantly compared to default logistic reg.
Risk control using threholdtuning.