# Feature engineering

→ There are a lot of features in any dataset
The dataset may not be pure and may contain a lot of noise
If such a raw dataset is fed into a ML model the performance of the model will suffer.

ML algorithms require the data to have a specific characteristic

Feature engineering is the process of transforming raw data into features that are more suitable for the model

It is the process of selecting, extracting & transforming the most relevant processes

These techniques highlight the most relevant patterns

## Goals of feature engineering

1. Preparing proper input dataset compatable with the model

2. Improving performance of ML model by cleaning the dataset

# Clean data Principles

These principles must be followed for good accuracy

1. Each Variable must have their own column → each column is a Variable

   ie Name must be split into M name
   L name

2. No cell must contain missing values

3. End result must be a numeric Value

# List Of Techniques

1. Imputation
2. Outliers
3. Binning
4. Data encoding
   - → one hot encoding
   - → categorical encoding
5. Grouping
6. Feature split
7. Scaling
8. Log transforms

# Imputation

(a) Dataset contain missing values

(b) In order to remove the missing values, three things can be done

(A) Dropping

(B) Numerical Imputation

(C) Catagorical Imputation

(A) Dropping → If blank values are more in number, then drop the column or row

threshold → 70%

i) DROP columns where more than 70% data is missing

ii) DROP rows where more than 70% data is missing

(B) Numerical Imputation → If blank values are less then replace it with some values

Replacing can be done by

i) Mean of non blank values

ii) Median of non blank values

(C) Catagorical Imputation → Replace blank data by a catagoery

If blank values make sense then replace them by a catagory like "other" "N.A." or "0"

# Missing Value Patterns

In real life datasets, certain datapoints may have missing values due to human errors and unavailable respondants

1. Missing Completly at Random

Data is missing at random without any relation to the characteristics of data.

eg receptionist forgot to enter the age of a patient at random.

2. Missing at Random.

Data is missing but missing because someone has not disclosed it.

The probability of the missing value depends on some observable data

eg. Young people may not disclose their name

name → Missing data
Age: Young → Observed data

Missing data is due to an observed data.

3. Missing Not at random.

Data is missing because people want to hide it

eg. Rich people may not disclose their income.

The income is not disclosed because it is too high or too low

The missing is related to characteristic of missing unobserved data itself

# Outliers

(a) Outliers are exceptions

(b) Outliers are not to be treated like normal data points

(c) Example ages of students in a school

16, 18, 15, 22, 16, 14, 19, 55, 14, 12, 13

outlier : odd one out

(d) If we feed such data into model, then model will give wrong output

(e) even simple average of all students age will be a wrong parameter to predict student age

(f) If such outliers are very few in number then they must be removed

(g) Outliers can also be due to human error in data entry
eg 15 was written as 55 by mistake

# Outlier detection

Outliers can be detected by stastical methods

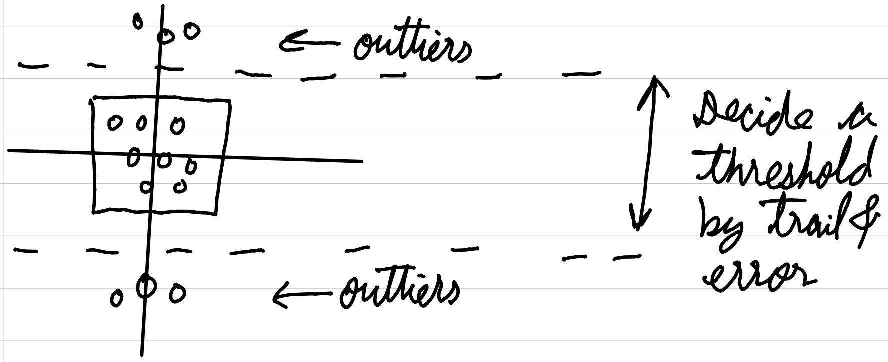
Ⓐ Standard deviation

if a value has distance from average that is higher than $t * S.D$

$$x - \bar{x} > \underbrace{t}_{\text{threshold}} \times S.D.$$

Ⓑ Thresholding

Depending on dataset, visualize the dataset




$x > t_1$ or $x < t_2$ are outliers

Ⓒ Percentile

Decide a percentile for outliers

eg Top 5% are outliers

or Bottom 2.5% are outliers

But adjustment needs to be done on basis of graph visualization

# Outlier Handling

What to do after outliers are found?

Ⓐ Remove outliers

→ If outliers are real values
eg man of 55 does attend school

Ⓑ Cap them to highest value / lowest value

55 becomes 23

Ⓒ Put realistic values for human error

eg a data point is "-17" we can understand that -ve is typo error

# Binning

(a) Binning means placing data into Bins or catagories

(b) Binning is used to prevent overfitting

(c) Binning is used to reduce data

(d) Binning leads to sacrifising the data & makes data more regularized

## (A) Numerical Binning

| Value | | Bin |
|---|---|---|
| 0 - 30 | → | Low |
| 31 - 70 | → | Mid |
| 71 - 100 | → | High |

## (B) Catagorical Binning

| Value | | Bin |
|---|---|---|
| Mumbai | → | Maharashtra |
| Pune | → | Maharashtra |
| Kolkata | → | Bengal |

# One Hot encoding

ⓐ Used to convert catagorical data to numeric data

| User | city |
|---|---|
| 1 | Mumbai |
| 2 | Pune |
| 1 | Pune |
| 3 | Kolkata |
| 2 | Kolkata |
| 1 | Kolkata |
| 1 | Mumbai |

| User | Mumbai | Pune |
|---|---|---|
| 1 | 1 | 0 |
| 2 | 0 | 1 |
| 1 | 0 | 1 |
| 3 | 0 | 0 |
| 2 | 0 | 0 |
| 1 | 0 | 0 |
| 1 | 1 | 0 |

Kolkata city values can be infered

} kolkata

ⓑ N distinct values converted to N-1 columns
3 values to 2 columns for city

# Catagorical Encoding

In this, Simply replace the catagorical values by numbers

| User | City |
|---|---|
| 1 | Mumbai |
| 2 | Pune |
| 1 | Pune |
| 3 | Kolkata |
| 2 | Kolkata |
| 1 | Kolkata |
| 1 | Mumbai |

| User | City |
|---|---|
| 1 | 1 |
| 2 | 2 |
| 1 | 2 |
| 3 | 3 |
| 2 | 3 |
| 1 | 3 |
| 1 | 1 |

# Grouping

Pivot tables can be used in similar way as one hot encoding but with non binary values

| User | City | Visit |
|---|---|---|
| 1 | Mumbai | 1 |
| 2 | Pune | 2 |
| 1 | Pune | 1 |
| 3 | Kolkata | 1 |
| 2 | Kolkata | 4 |
| 1 | Kolkata | 3 |
| 1 | Mumbai | 3 |

| User | Mumbai | Pune | Kolkata |
|---|---|---|---|
| 1 | 4 | 1 | 3 |
| 2 | 0 | 2 | 4 |
| 3 | 0 | 0 | 1 |

N columns for N cities

Count the values (Unlike one hot encoding)

# Feature split

ⓐ Sometimes, a feature might be represented in a different manner in dataset

ⓑ Seperation of relevant features need to be done

| Name | | F. Name | L. name |
|---|---|---|---|
| Harry Potter | ⇒ | Harry | Potter |
| Tom Riddle | | Tom | Riddle |

ⓒ Extracting date can be done by various methods

i) Traditional data split into dd mm yy or other format & remove unnecessary columns
ii) Extracting timeperiod eg time passed since
iii) Extracting features eg weekday
Holiday
old/new

# Log Transform

ⓐ Sometimes data is skewed

ⓑ Log transform helps to even such data

ⓒ It also decreases the effect of outliers

ⓓ eg age difference between ages 10 - 15 is more relevent than age difference in ages 65-70

5 years of difference is higher for small magnitude

$$x = \log(x+1)$$

Only works for +ve values.

# Scaling

(a) Numerical features differ between themselves and dont have same range

eg age & income wont have the same range

(b) But ML model expects everything to be in the same range

(A) Normalization based on range

$$X_{norm} = \frac{X - X_{min}}{X_{max} - X_{min}}$$

MinMax normalization (range will be 0-1)

(B) standardization (z-score)

$$z = \frac{x - \bar{x}}{\sigma}$$

standard deviation is used to convert the values (range not 0-1)
downscales all values, makes some -ve values

Mnemonic

# F I S H B L O G

Feature extraction
Imputation
Scaling
Handling Outliers
Binning
Log transform
One Hot encoding
Grouping