Featurization And Feature Engineering

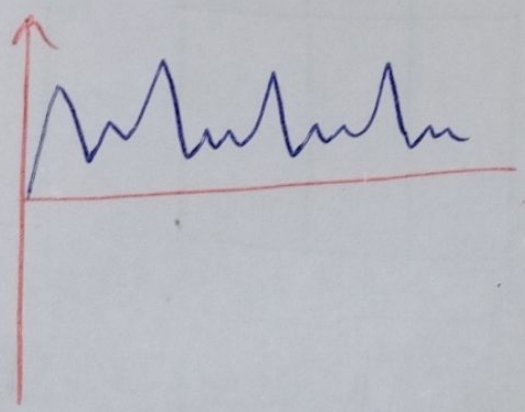
- most important
- convert data to numerical vector

Text data: BOW, tfidf, avg w2v, tfidf w2v

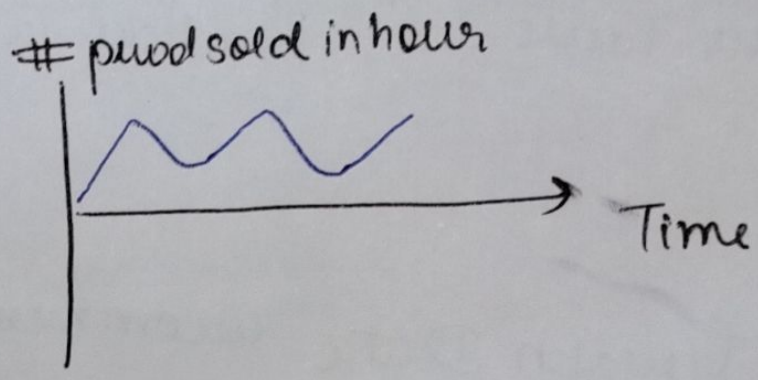
↓ various featurization

categorical data: one hot encoding
mean response rate, domain specific

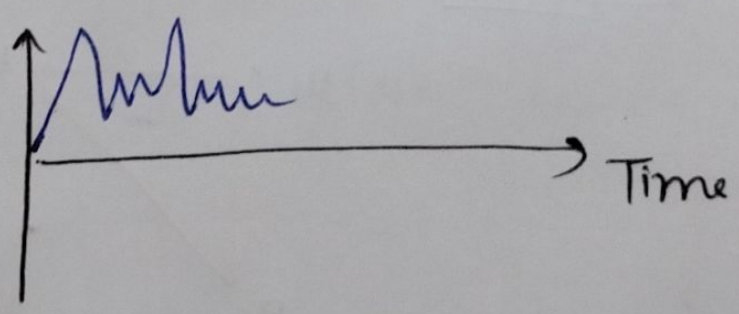
Time Series heart rate



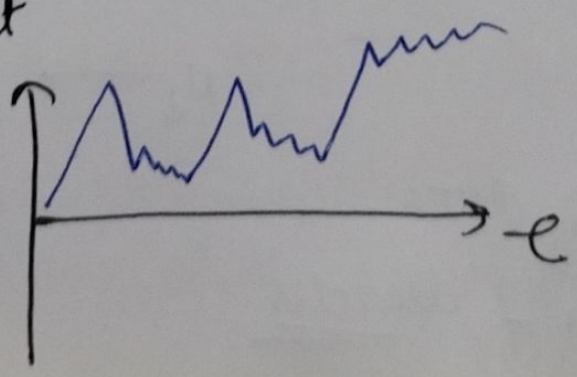
e Commerce



Speech



Stock Market price



Text: Sequence data

w_1, w_2, w_3, w_4 ——→

Image data Face detection, face recognition
X-rays, MRI scans, video ↑
Image + Time Series

Database Tables

T1

C Id	Locati

Cus Table

T2

C Id	pu Id	Time

Purchase Table

T3

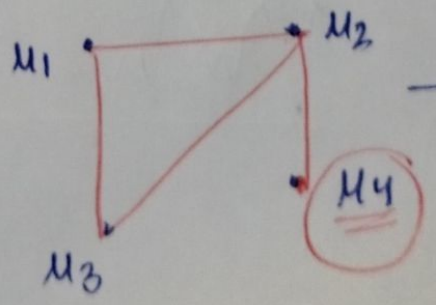
	Price	disc

Product Table

Numerical features → ML

Graph Data recommend a friend on FB

[→ vertex
→ edge]



→ Numerical Vector → ML

$u_3 \longrightarrow u_4$

Tons of types of data
→ featurization:- decades

→ some commonly used featurization
(general purpose)

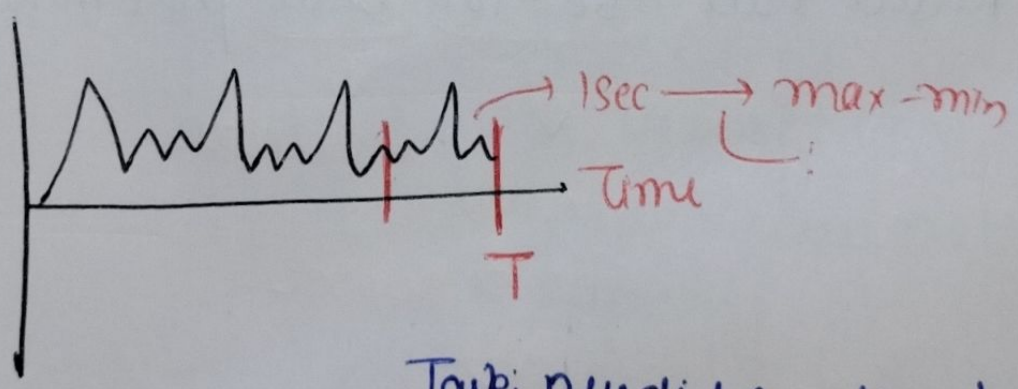
Moving Window

Simplest featurization of Time Series

ECG: ① Take a window of Time

② Then compute mean, Std dev ...
median, quantile
Max, Min
Max - Min
Max / Min
local maxima & minima
Zero crossing (Times it crosses mean line)

ECG:

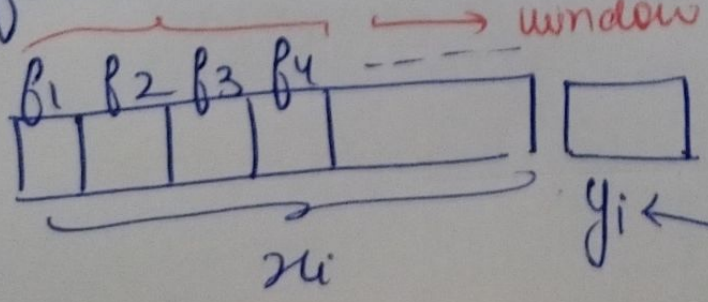


Task: predict cardiac heart attack in next 10 min

① Window - width

② features in window which are useful

③

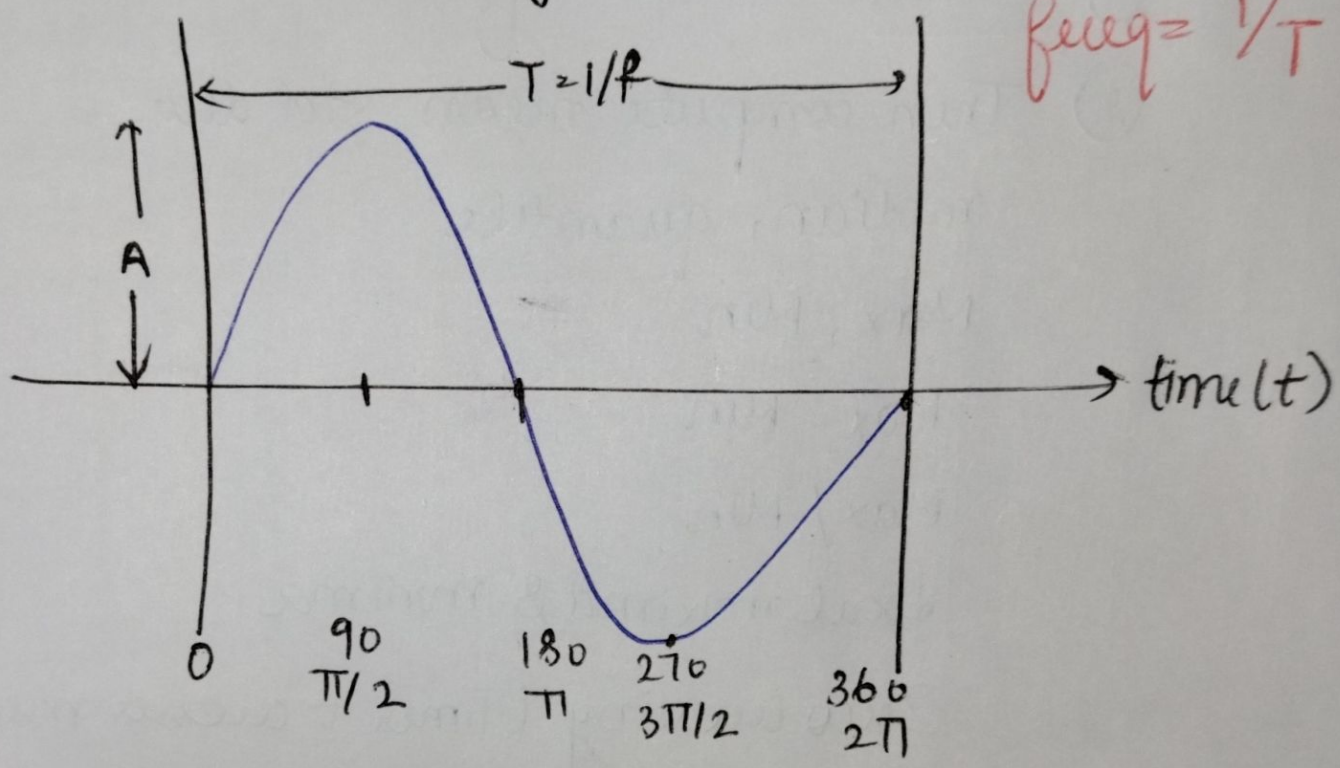


Fourier Decomposition / Transform

method to represent Time Series

physics, applied math, communication, CS, Signal processing

Oscillating Sine Wave



freq = 1/T

Heart rate: 60 – 100 beats per minute

↓ ↓ oscillates

1 to 1.6 beats per sec

⇓

1 Hz to 1.6 Hz

W_3
W_2
W_1

$\frac{1}{T} = freq$

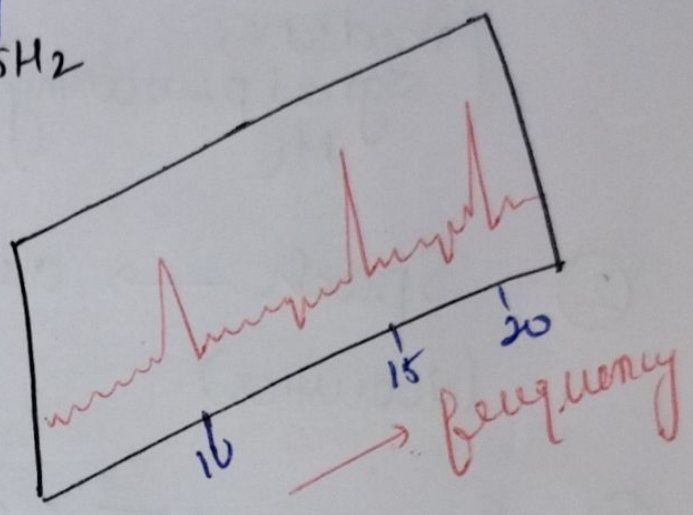
$20H_2$
$15H_2$
$10H_2$

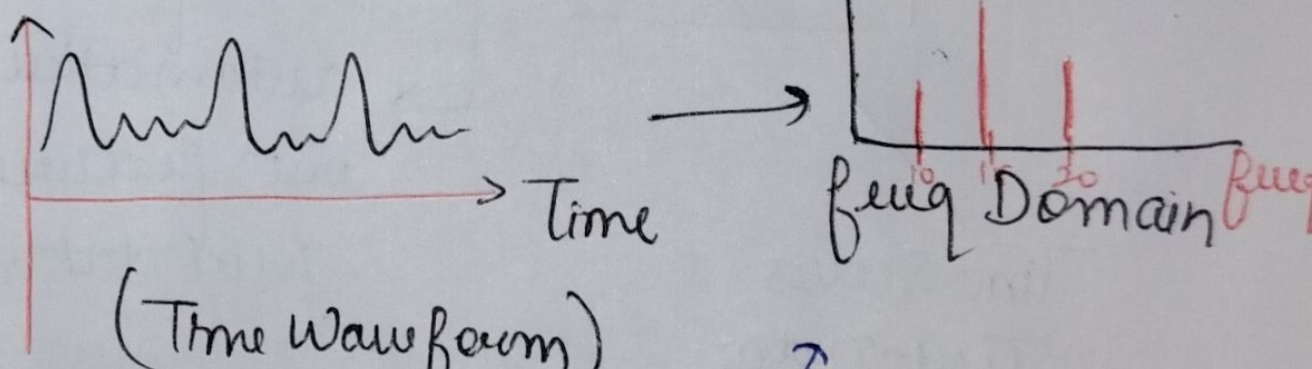
Composite Wave

repeating pattern

heartbeat / eCommerce

Time





(Time Waveform)

Time domain

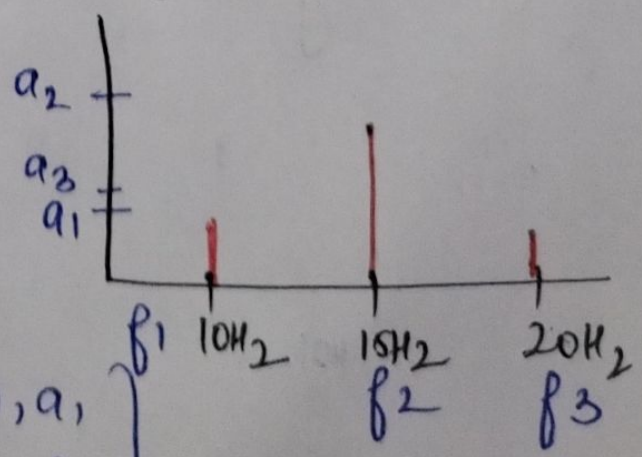
Fourier Transform

useful inherent pattern is repeating

$|f_1|f_2|f_3|a_1|a_2|a_3|$ → freq matters

x_i

FT



Time domain

Feature Vector

$\{f_1, a_1\}$
$\{f_2, a_2\}$
$\{f_3, a_3\}$

Deep Learnt features

50-yrs → Time Series

① Heartbeat → design special features
(medicine, signal processing ML)

② Speech → design special features
(acoustics)

③ Ecommerce Time Series data

Till 2012-2013

Deep Learning → Lots of data
automatically learn the best featurization for your data

Time Series
Text Data
Image Data

5-6 yrs (2012-2017)

{ Google improved: Speech recognition
Siri:
Song Recommendation } → deep learnt features

Image Histograms

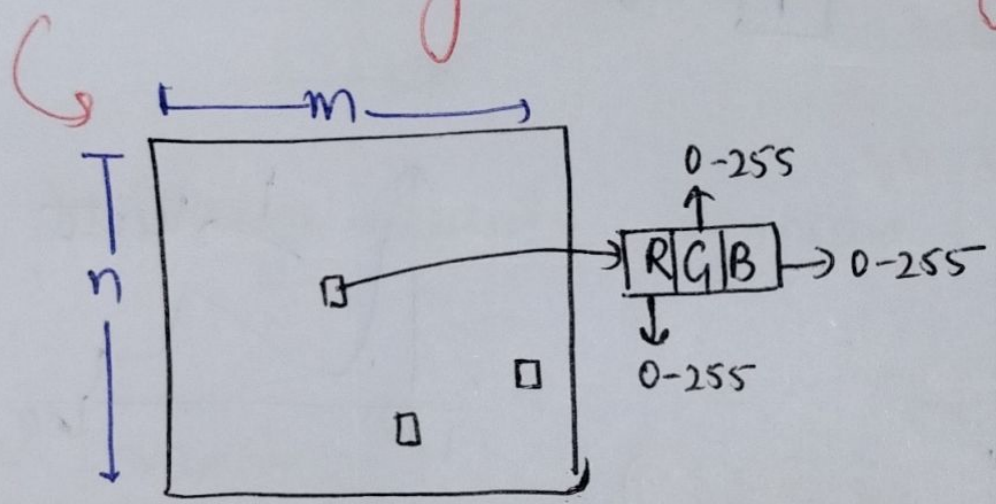
Images: face, object, Scan, X rays, autonomous Cars

↓

30+ yrs of research

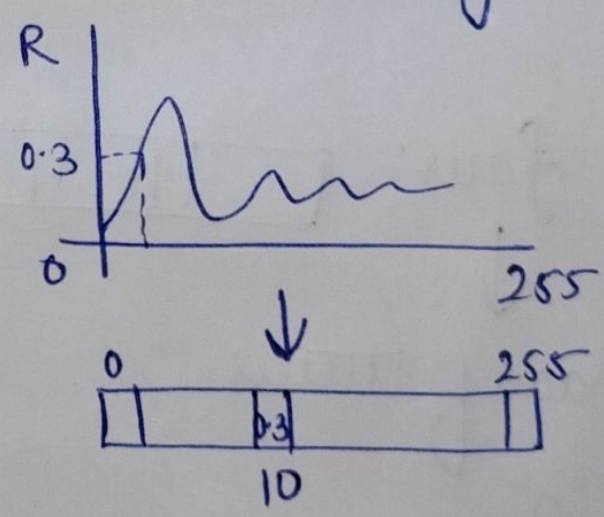
2012: Deep learning (CNN)

① Colour Histograms ② Edge Histograms

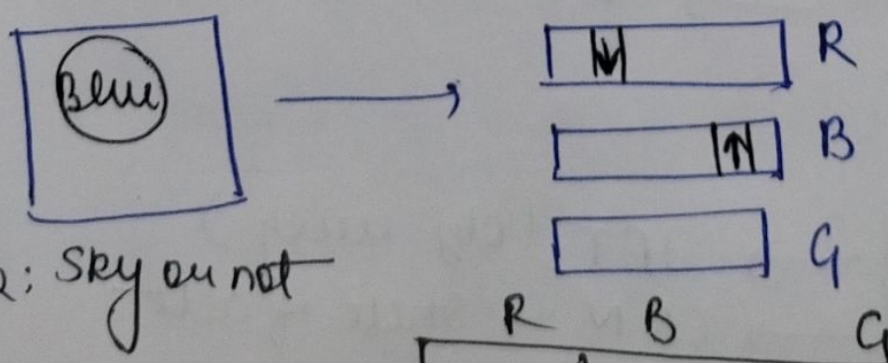


① Red Values for each pixel

(n×m) data points → histogram



② Green And Blue



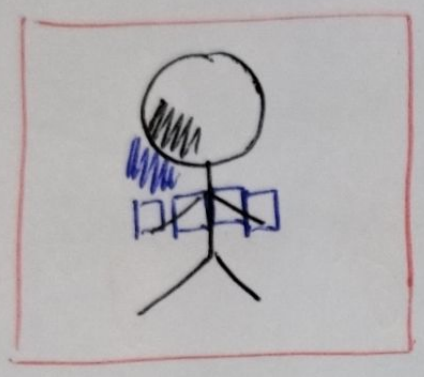
Task: Sky or not

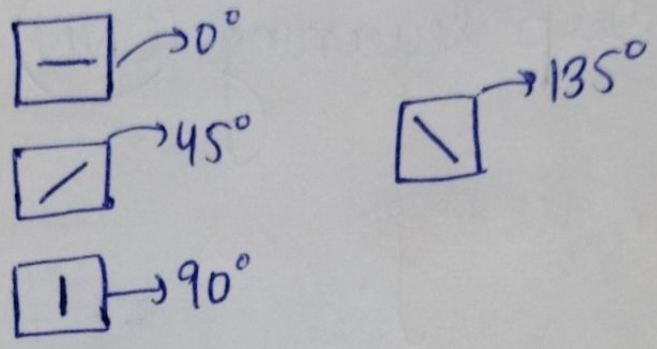
face detection: skin colour ✗

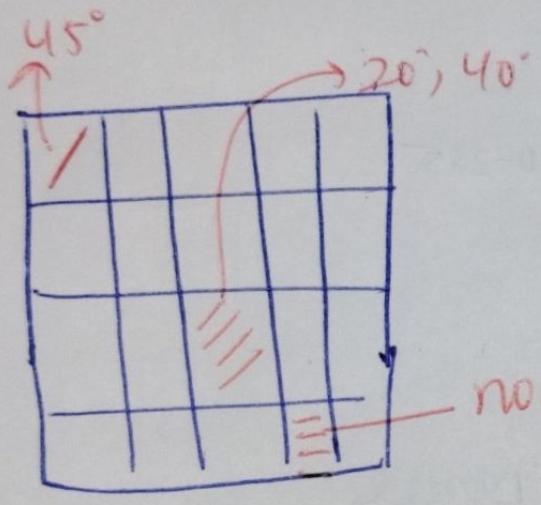
↓

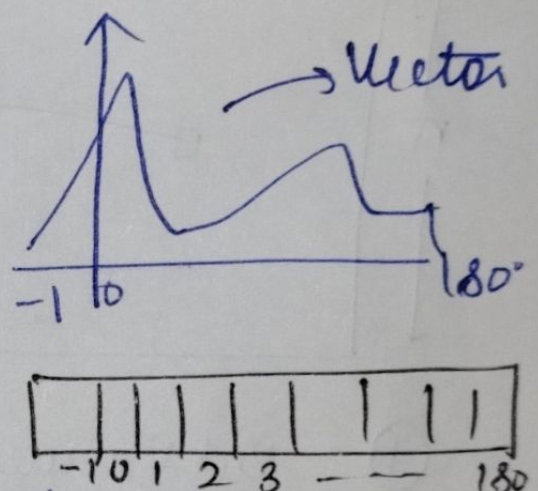
Colour hist → ✗ recognize shapes

Edge Histogram (Image processing)







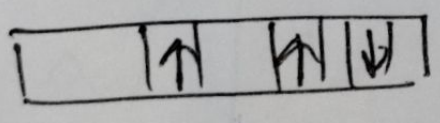


each region: edge value / edge angle

↓

histogram

faces:



↓

Haar features

{ Colour hist
  edge hist } → v.v basic & rudimentary

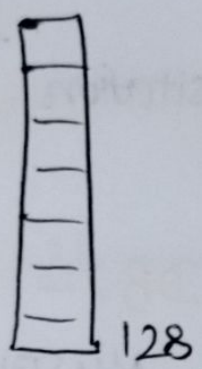
[→ SIFT (obj recog.)
  → CNN → state of art

Scale Invariant feature transforms (SIFT)

→ Image featurization

→ object detection (2000's)

→ creates vector for each point (128 dim vectors)



$I = \{v_1, v_2 \text{ --- } v_{10}\}$

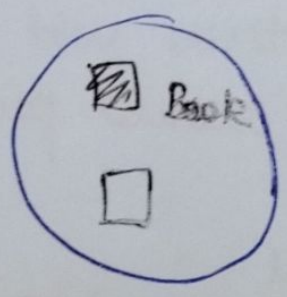
↑ 128 dim

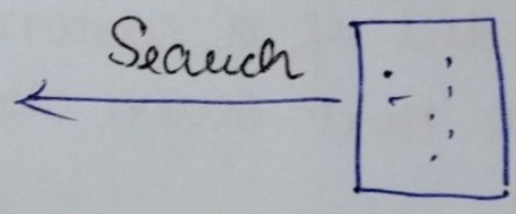
→ Image Search (amazon.com) → mobile app

↓

Take picture of anything and search

{ Scale Invariance
Rotational Invariance





Deep Learning features

Time Series → LSTM → featurize

Images → CNN ← 2012

↓ → almost automatically

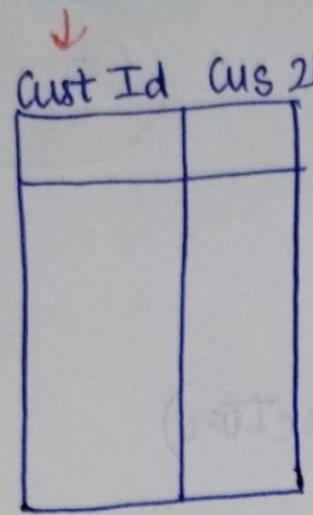
best featurization (30+ yrs)

X-rays → lost of data / Images

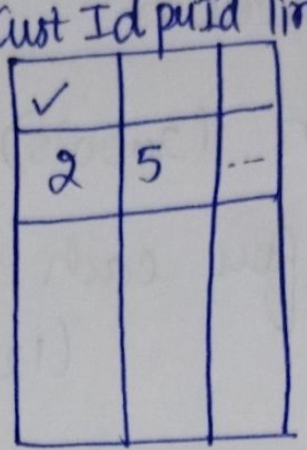
↳ tumor or not → (CNN)

Relational Data And featurization

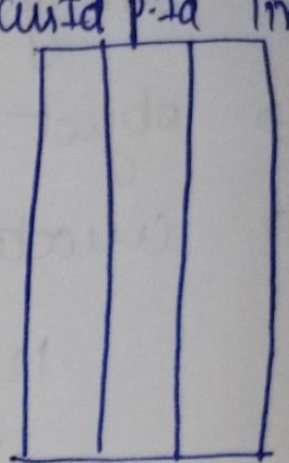
eg



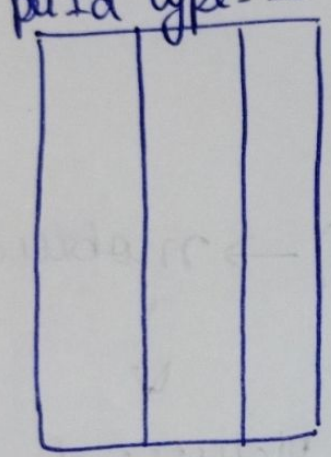
Cust Table



Cust viewing/visitation data



purchase data



pid. data

Relational DB:- Oracle, MySQL, SQL Server

Task: predict if a customer would purchase product in next seven days.

#cust Id visit any pr. belonging to same type

Cust Id, pr Id → I/O

buy

does not buy

#the cust id viewed pr. id page in whis domain knowledge

β_1 β_2 β_3

x_i

y_i

Income level

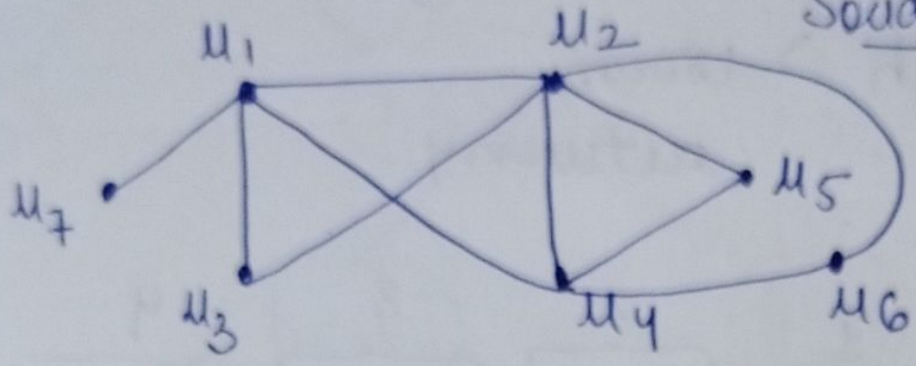
Relational data → SQL

↓

featurization

Graph data & featurization

eg.



Data

Social graph (Facebook)

u_i:- users (users)

edge (u_i, u_j): friends

Task: Recommend new friends for a user (u_4)

ML

$u_i, u_1 \longrightarrow$ 1/0

mutual friends

$u_7 - u_4$:— u_1

$u_3 - u_4$:- u_1, u_2

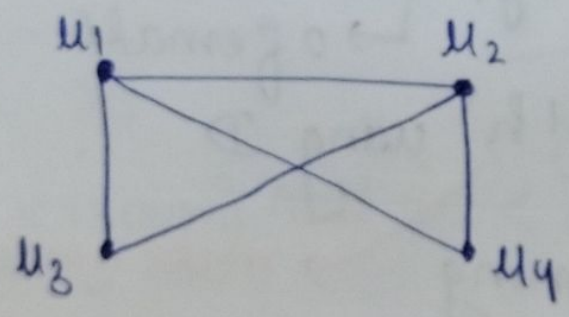
✓ $u_7 - u_4$
✓ $u_3 - u_4$
} Recommend

graph based features

f_1: # mutual friends

f_2: # paths b/w vertices u_3 & u_4

(eg)



Feature Binning (bucketing)

→ extension to Indicator variables

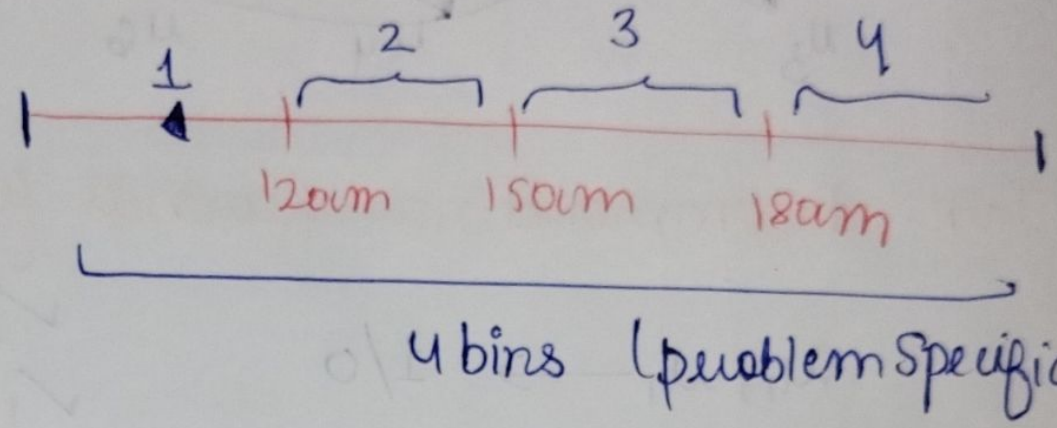
→ eg: height

if height < 120cm
return 1

if height < 150cm And $h \geq 120$cm

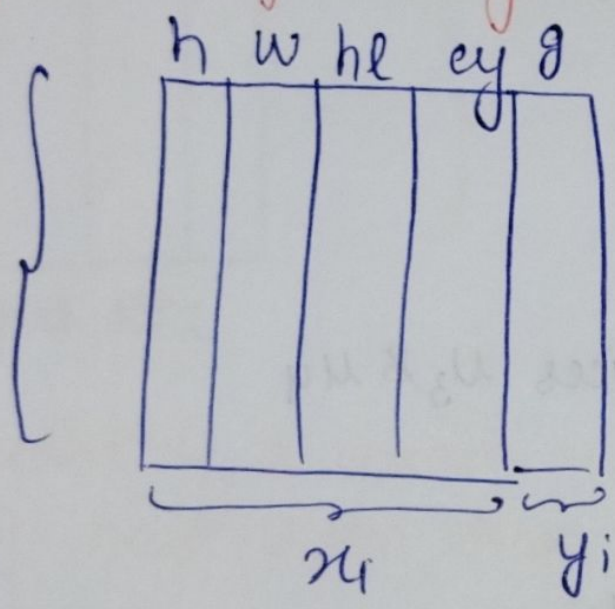
if $h < 180cm$ and $h \geq 150cm$
return 3

if $h > 180cm$
return 4



4 bins (problem specific)

Task : predict gender given: h, w, h, eye colour

'D' 

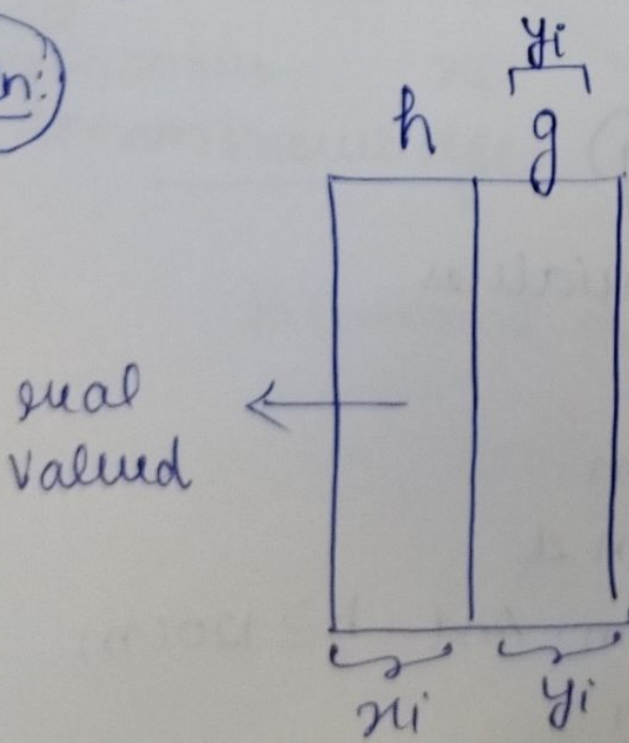
y_i → 1 male
→ 0 female

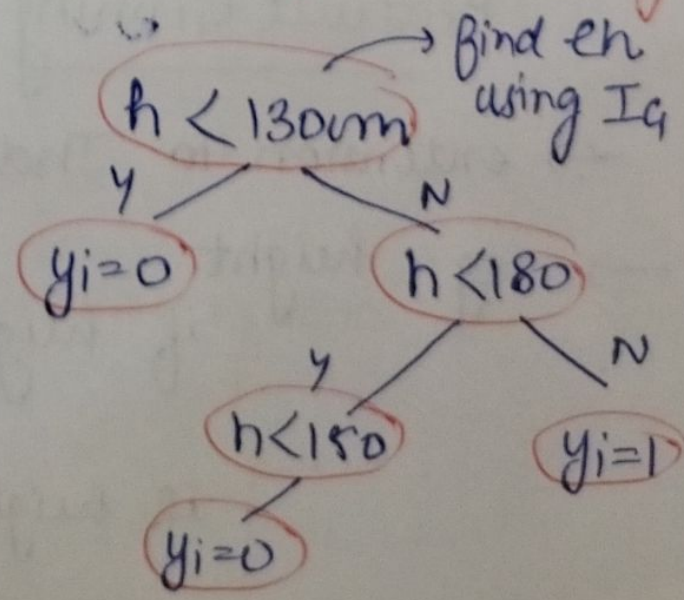
height → binning

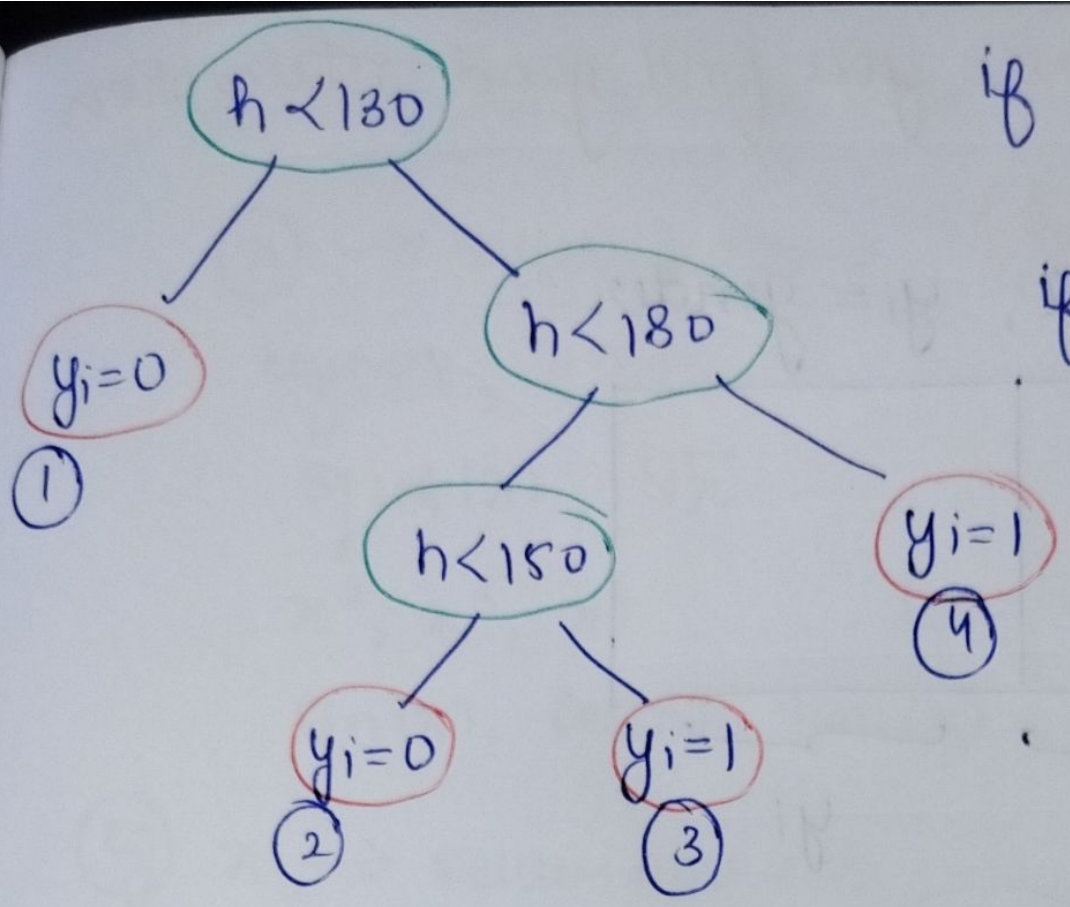
Challenge : binning of 'h' using D

↳ data driven binning → make sense to use y_i to perform binning

Soln:







if $h < 130$
return 1

if $h \geq 130$ And $h \leq 150$ cm
return 2

if $h \geq 130$ And $h \leq 180$
return 3

if $h \geq 180$ cm
return 4

binned real valued features using y_i's & features itself
using DT

Interaction Variables

Task: predict gender

$h, w, hl, ec \longrightarrow$ gender

x_i $\quad$ y_i

(eg) (1) $(h < 150cm)$ AND $(w < 60kg) \longrightarrow$ 2way interaction feature (logical)

x_i $\quad$ $f_i = 1$

(eg) (2) (a) $h * w$
(b) $\sqrt{h} * w$
} math 2way interaction feature

(eg) (3) $h < 150cm$ AND $w < 65kg$ AND $hl > 5cm$ } $\longrightarrow$ 3way interaction feature (logical)

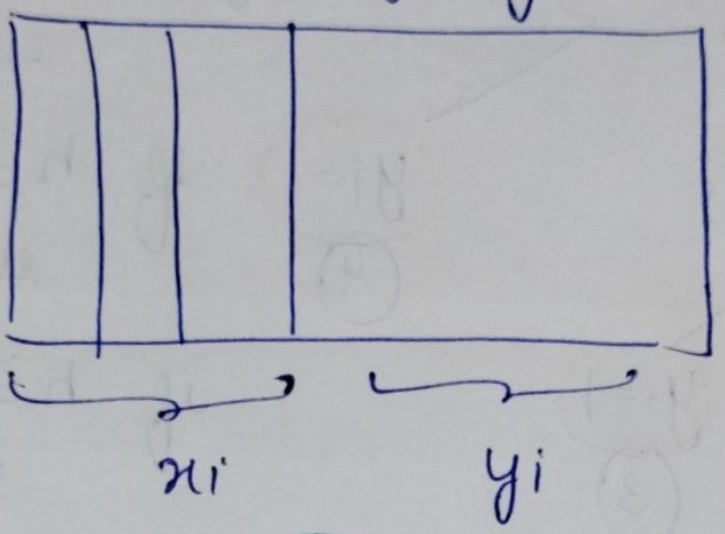
(Q) Given task, how do you find good interaction features

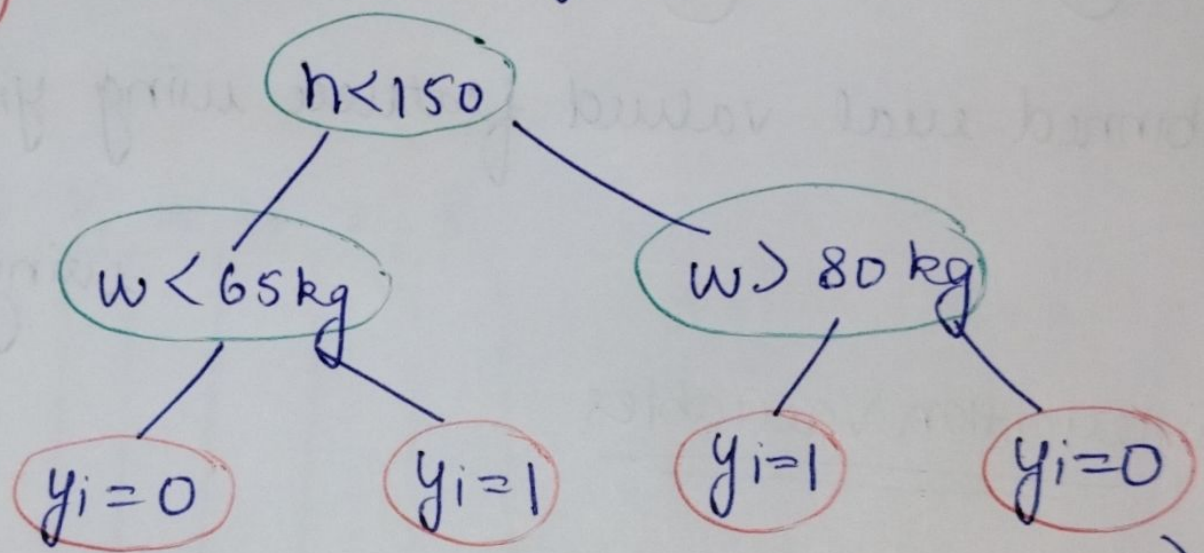
(Soln) DT

↓

method to perform feature engineering

h, w, hl, y_i = gender





h <150cm AND w<65kg

new binary feature
(Interaction variable)

h ≥150cm AND w ≤80kg

Corresponding to every leaf node create a new feature.

Mathematical Transformers

(x) → single feature

$\log(x)$, e^x

Sqrt(x); $\sqrt[3]{x}$

x^2, x^3, x^4 ..... (polynomial)

Sin(x), Cos(x), tan(x)

(Q) what is the best transform
→ problem specific
→ case study

(eg) x → power law dist

↓

$\log(x)$ (Gaussian dist)

box cox transform

Model Specific featurizations

(eg) f_1 → power law dist ($\log(f_1)$)

Logistic Reg. → gaussian Naive Bayes

↓

features are gaussian dist

(eg)2. $f_1, f_2, f_3, y \in \mathbb{R}$

$y = f_1 - f_2 + 2f_3$ ← domain knowledge

linear combination of f_i's

DT may not work very well

Linear model → Linear Regression

$w_1 = +1$, $w_2 = -1$, $w_3 = +2$

eg ③ → y → interaction of f_1 & f_2 ← Domain knowledge

(gender) (h,w)

↳ DT/ RF/ GBDT Tree based

linear model (log. Reg, lr svm)

text BOW → linear models

↓

v. high dim-space → hyperplanes → lr SVM, lg. Reg

+ve -ve

Feature Orthogonality

* The more different / orthogonal the features are the better the model would be

$f^1, f^2, f^3 \xrightarrow{M} y$

if $f_1 \xrightarrow{corr} y$, $f_2 \xrightarrow{corr} y$, $f_3 \xrightarrow{corr} y$

$f_1 \xrightarrow{h.corr} f_2$ $f_1 \xrightarrow{h.corr} f_3$

$f_2 \xrightarrow{h.corr} f_3$

---

f^1, f^2, f^3, f^4 $f_4 \xrightarrow{corr} y$; v. less corr with f_1, f_2, f_3

$f_1 \xrightarrow{corr} y$ $f_2 \xrightarrow{corr} y$ $f_3 \xrightarrow{corr} y$

$f_1 \xrightarrow[corr]{not} f_2$ $f_1 \xrightarrow[corr]{not} f_3$ overall impact

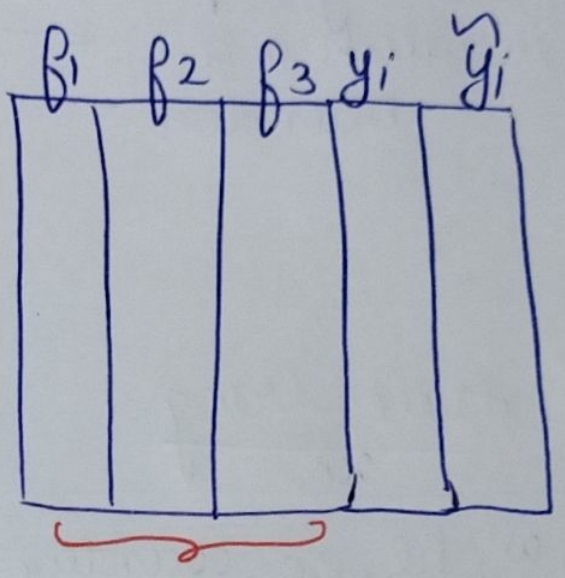
$f_1, f_2, f_3 \xrightarrow{M} y$ (better)

how to create f_4

$f_4 \xrightarrow{corr} y$; v. less corr with f_1, f_2, f_3

idea $f_1, f_2, f_3 \xrightarrow{M} y$

M:- $\tilde{y}_i$



(Q) how to assign new feature f_4 s.t

$f_4 \xrightarrow{corr} y_i$

less corr with f_1, f_2, f_3

error: $y_i \sim \tilde{y}_i = e_i \quad \forall x_i$;

$f_4 \xrightarrow{corr} e_i$

Similar to

GBDT → model1 → errors → model2

↓

(pseudo errors)

$f_4 \rightarrow e_i$

avoid overfit → Boosting

$f_4 \xrightarrow{corr} y_i \sim \tilde{y}_i$

with → less corr with f_1, f_2, f_3

Domain Specific featurization

Heart attack ⟶ ECG data

doing no prior research is a blunder ←

impt to research existing featurizations by doc/specialists

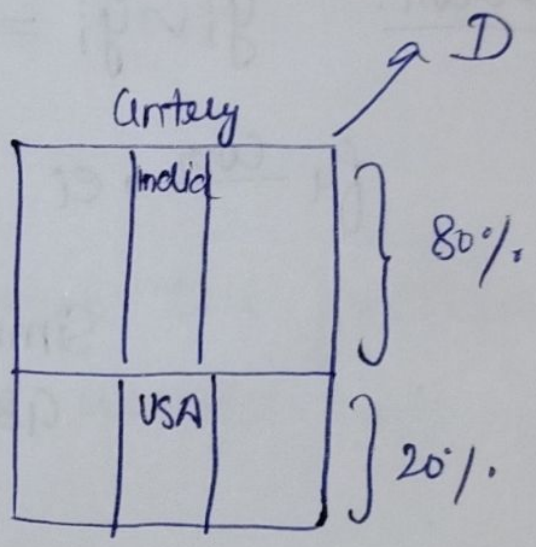
↓

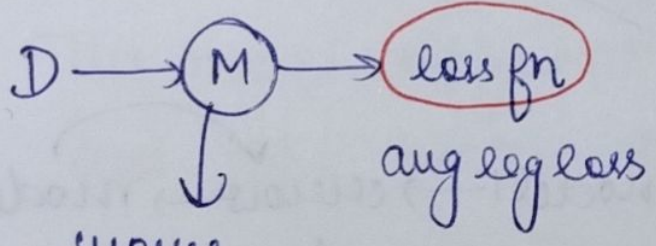
create new features

Feature Slicing

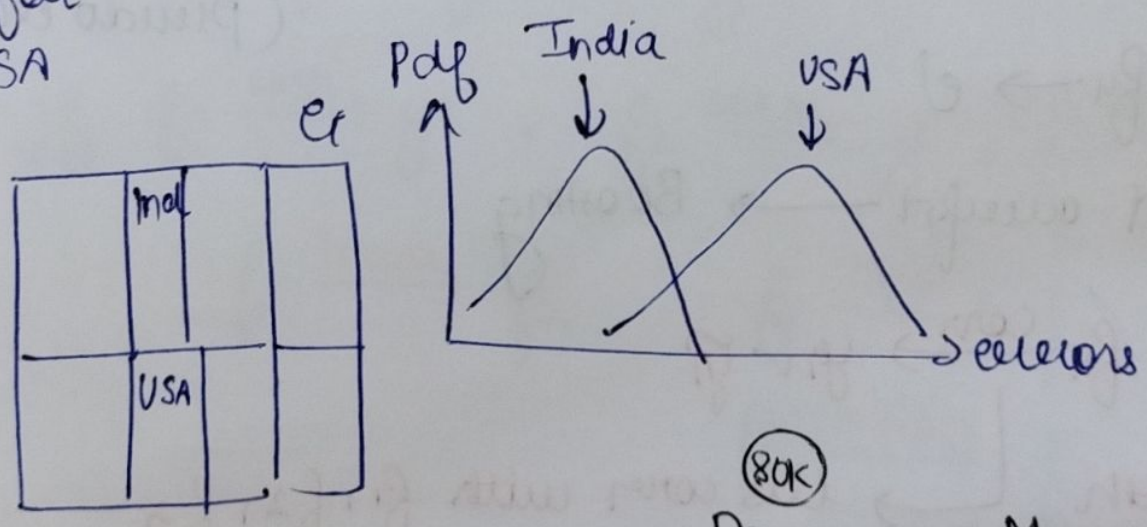
(eg) x_i: h, w, hl, cc, country ⟶ y_i: gender

USA, India

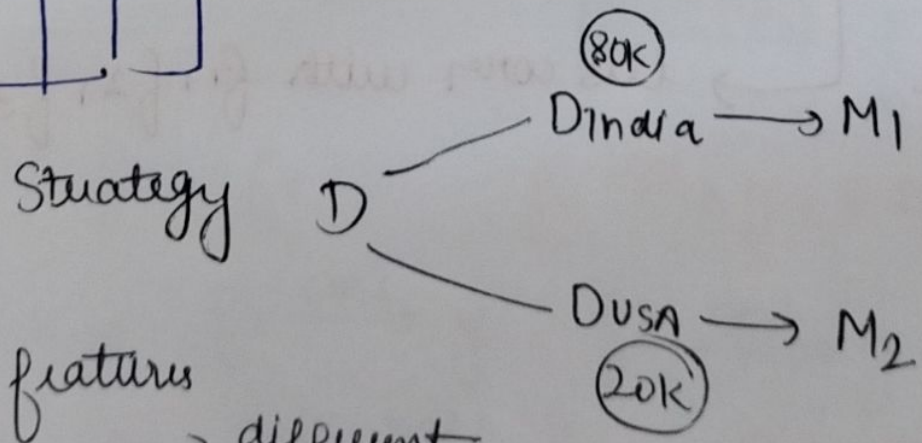




Test



Strategy



Slicing data on features

① cat_1 & cat_2 ⟶ different

② sufficient # for each data