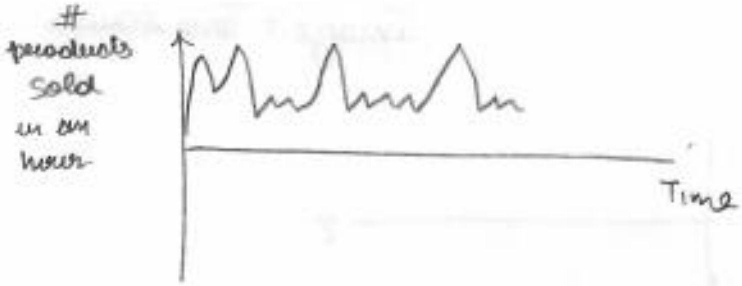
Featurizations & feature Engineering:

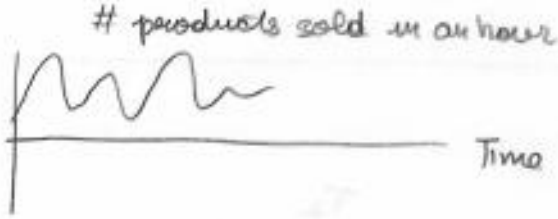
① Most important aspect : Fe FE

Time Series : heart rate

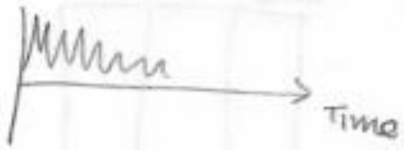
1·)



11·) ecommerce



111·) speech /audio



1v) Stock market

v.) Text :- Sequence data

$w_1\ w_2\ w_3\ w_4 \ldots\ldots$

sequence of things

Time series → numerical vector. → Applying model

Image data : Face detection, Face recognition

⇓

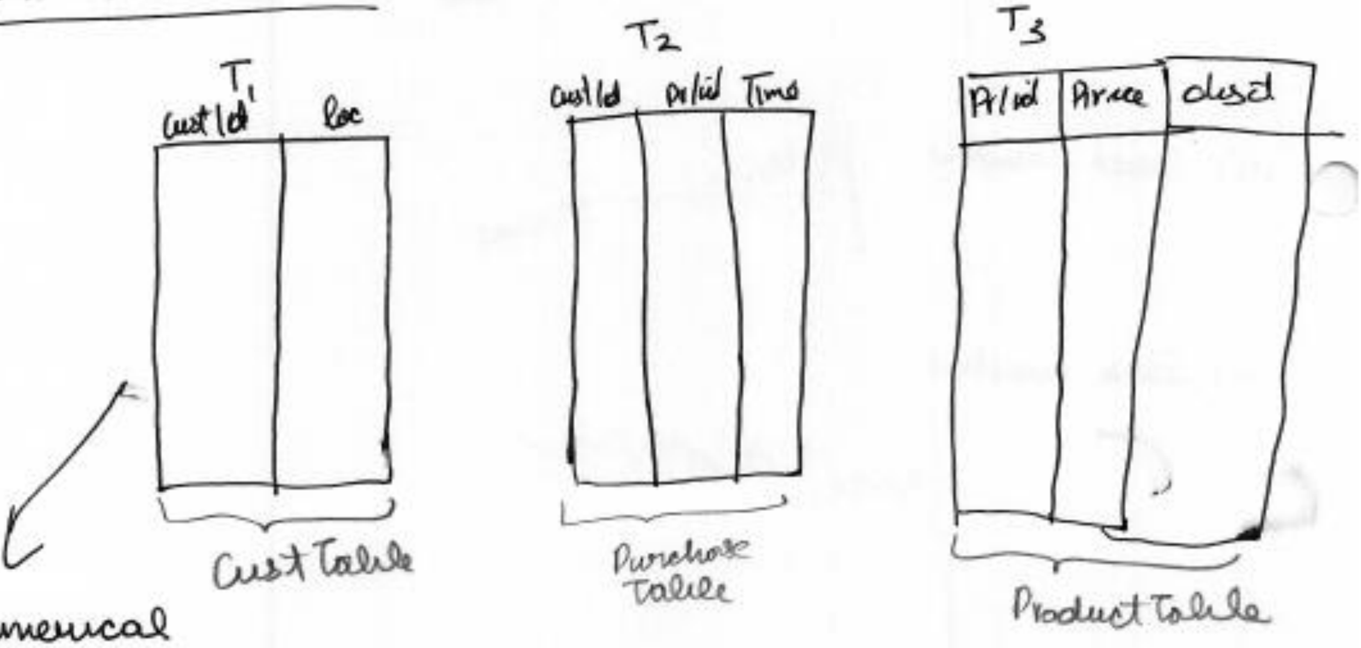
Numerical data

X-Rays, MRI scans, video

↑

Image + Time series

Data - Database Table.

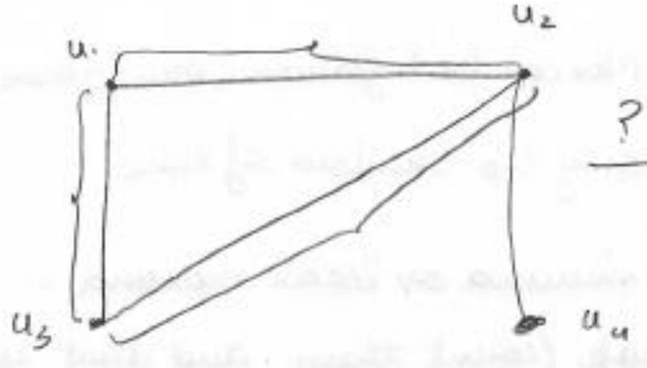


Numerical feature?

↓

for ML

Graph-data → recommend a friend on FB.



$\xrightarrow{?}$ numerical vector

↓

So that you can apply ML models.

$u_1, u_3 \longrightarrow u_4$

$\left\{ \begin{array}{l} \circ \text{ vertex} \\ \rightarrow \text{edge} \end{array} \right\}$

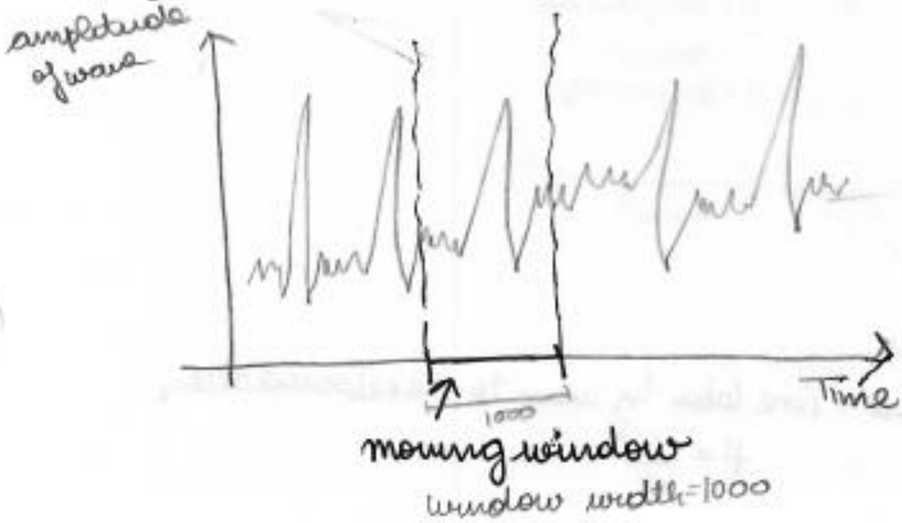
Tons of type of data

→ featurization : decades performing research on it

Moving window for Time Series Data

◇ Simplest featurization of Time-Series

ECG :- (Electro Cardio Gram)



(1) Choose window of data
and than compute mean value / Standard - deviation

(2) median,

(3) Max, Min. → Max can be 1 feature, Min other feature

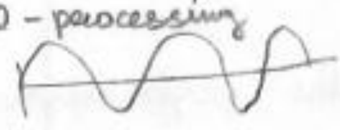
(4) Max - Min → only the window of time

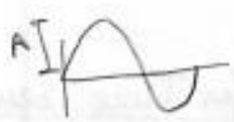
(5) Number of local minima or local maxima.
Check number of peak (count them and what is the local maxima).

Fourier Decomposition / Transform :-

physics, applied math, Electronics, Communication, CS

Signal - processing

freq. :- 

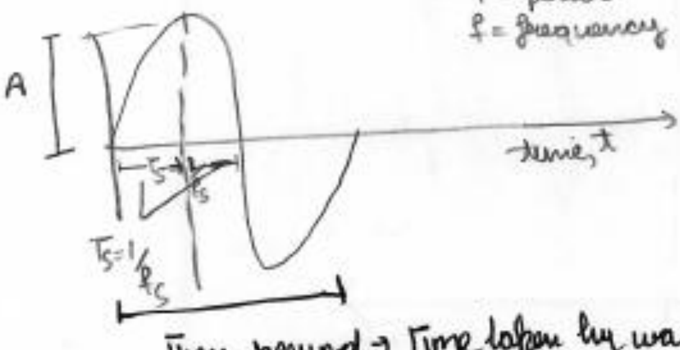


amplitude :-

phase :- 

oscillating Sine wave

A = amplitude
T = period
f = frequency



Time period → Time taken by wave to take 1 oscillation

$$f = \frac{1}{T}.$$

for eg. one oscillation per second = 1 Hz

two oscillation per second = 2 Hz

heart - rate → $\underbrace{60\text{-}100}$ $\underbrace{\text{beats}}$ per minute

⇓ oscillates

↑

1 to 1.6 beats per seconds

⇓

$\boxed{1\,Hz \text{ to } 1.6\,Hz}$

normal heart beats 60-100 beats per minute

If heart beats 30-150 beats per minute than abnormality can be seen which can be used for prediction

2.) ecommerce sales :-

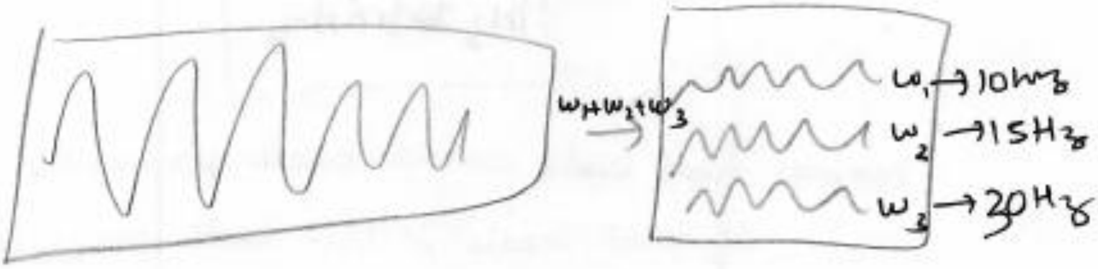
daily pattern :-

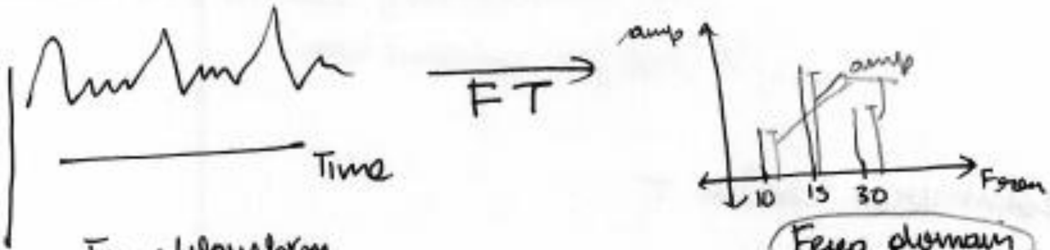
weekly pattern :-

annual pattern : - Christmas | holiday → sales would be more

Fourier Transform

Given any composite wave.
we can decompose it into sum of sin waves.



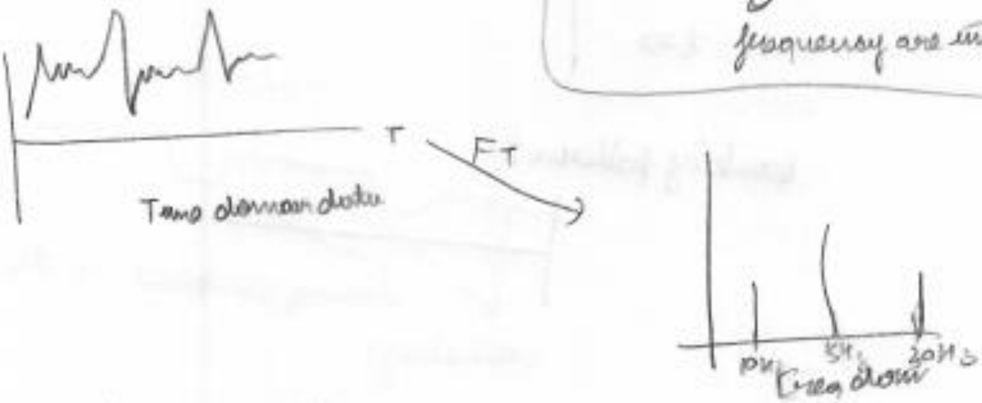


Time Waveform

Freq domain

useful whenever you have repeating pattern

frequency are important



feature - vector

fourier representation is important when you have repeating waveforms
↑ frequency

Case Study → Fitbit's accelerometer → to track movement everyday
(use fourier transforms windows

Deep learnt features : LSTM

30-50 years → Time series

① heart beat → design special features
↑ these will not work special features

② speech signal → design
(acoustics)

③ ecommerce time series data

Deep learning → lots of data → "automatically" learn the best featurization for your data also called Deep learnt features
↓
best features today

{Time series, sequence of data, image data}

decades.

{ Google Now : Speech recognisation } Deeplearning
Siri

? Why not use Deep learning for all problem when you have lots of data ?

① Deep learning (2018) works best data (Image, audio, complex data).

② Simple problem predicting probability add being clicked logistic regression work well.

③ Using non deep learning method like logistic, trees we can interpt the data (instead of black box).

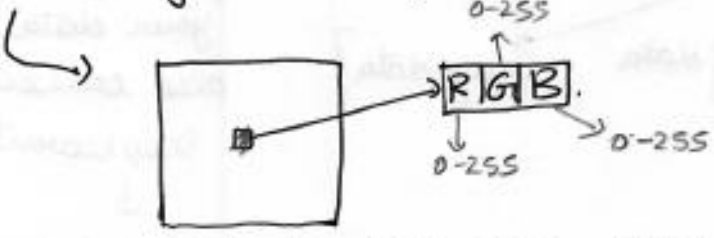
④ When we need · low latency rate than deep learning is not so good.

Image histogram

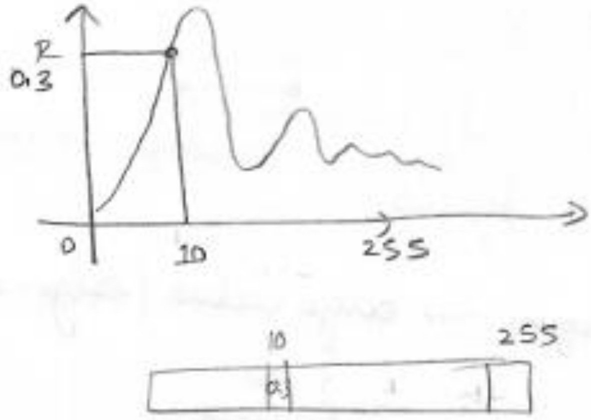
☆ Images :- faces, object, scans, x-rays, autonomous cars,
↓
30+ years of research.

2012 : Deep learnt features + CNN (Convolution Neural Network) is used nowadays.

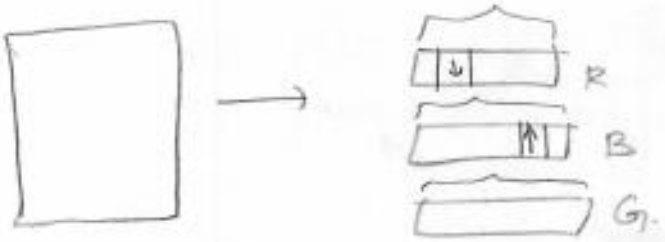
1) Color histogram , edge histogram.



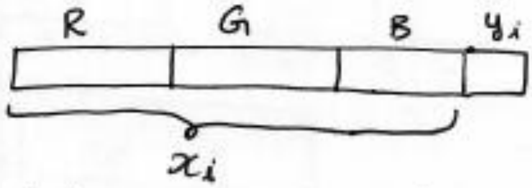
Step 1 Take all Red values for each pixel
(n×m) data points → histogram



Task: sky or not?



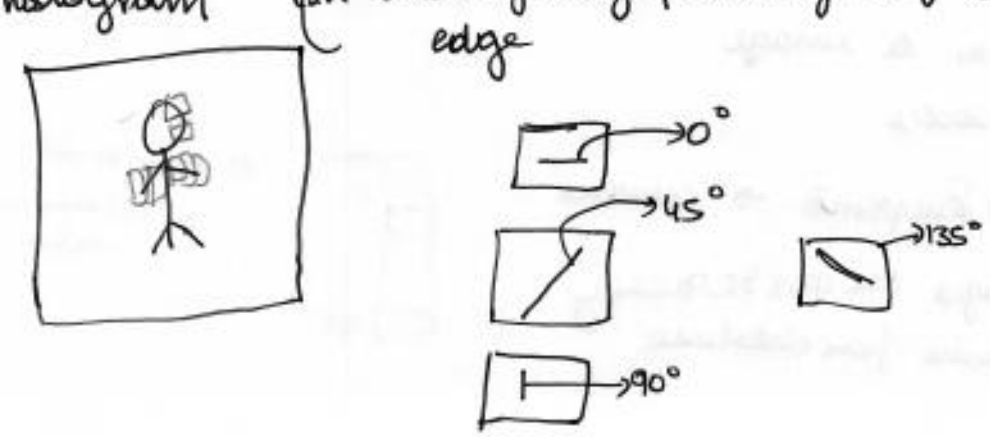
intuitively



Can you do face detection — Skin colour occur in some particular Shade so can be don

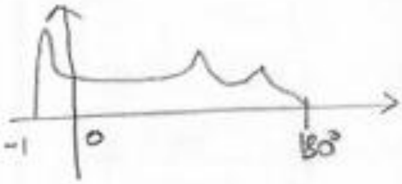
Color histogram → cannot recognize shape
but useful for detecting colour

2.) Edge histogram (In area of Image processing lot of algo to detect edge)

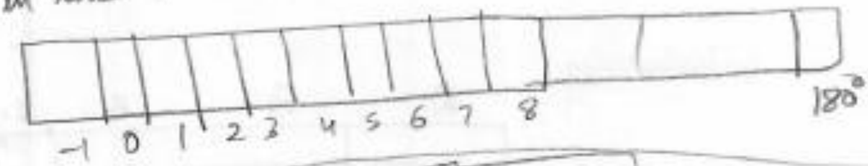


Break image into grid

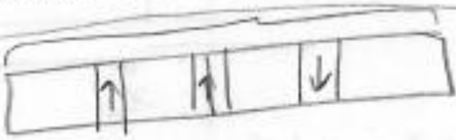
each region → edge value | edge-angle
↓
histogram.



Store edge in this.



faces :-



↓

haar features (used for face recognization)

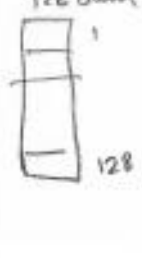
Keypoints : SIFT

Scale Invariant Feature Transforms (SIFT)

→ objects in a image
→ early 2000s

detects → keypoints → corners

comparing 2 images one can be query and one can be from database



each key point
128 dimensional vector

☆ SIFT properties like :-

i) Scale invariance
ii) rotation invariance

It has library called open CV.

☆ **Deep learning features : CNN**

→ Time series → LSTM → featurize

→ Images → CNN → Convolution Neural Networks

CNN beats the best features

CNN → automatically create images / features.

X-ray → lots of data / Images → CNN

↳ tumor or not

much faster than decades of research.

Today – CNN

Relation Data & featurizations

eg.

CustId	Zipcode

Customer Table

CustId	productId	Time
2	5	

Customer viewing / visitation data

CustId	pid	Time

Purchase data

prodId	Product type

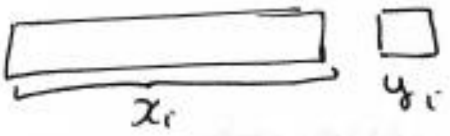
Product data

Relational Data Base → Oracle, MySQL, SQL Server.

Task: predict if a cust would purchase a product in the next 7 day

Cust Id, PId ⟶ 1/0

buy   not buy

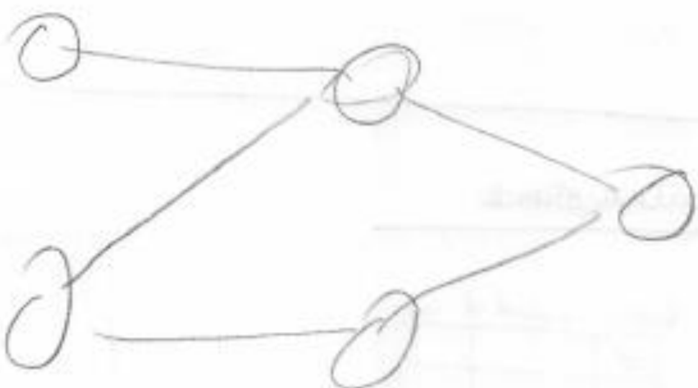


domain knowledge

times the customer id viewed the Id page in 24hours

times the cust id visited any product Type as prod

Graph data



feature Engineering: Indicator variable

eg ① "height" as a feature

→ real valued feature

→ $h > 150$ —— ①

$h \leq \underline{150}$ —— ⓪

↳ Threshold

problem specific

domain specific

h → real-valued feature

h → binary indicator var feature

→ 1 if $h > 150$

→ 0 otherwise

Feature binning ⇒ Binning is a process of converting regression problem into classification problem.

→ extension to indicator variables:

→ eg :- height as a variable

if $h < 120$ cm
  return 1
if $h < 150$ cm AND $h \geq 120$ cm
  return 2
if $h < 180$ cm AND $h \geq 150$ cm
  return 3
if $h > 180$ return 4

4 thresholds here

Interaction variables

Task: $\underbrace{h, w, hL, ec}_{x_i} \longrightarrow \underbrace{\text{gender}}_{y_i}$

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

x_i

$f_1 = 1$ / logical 2way interaction feature

else

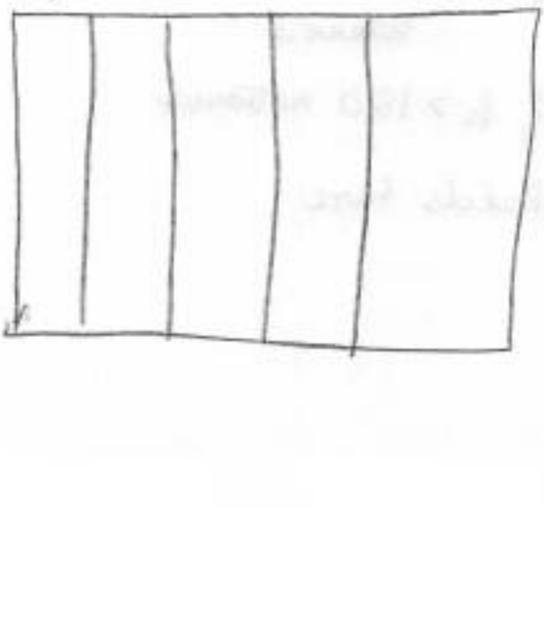
$f_1 = 0$

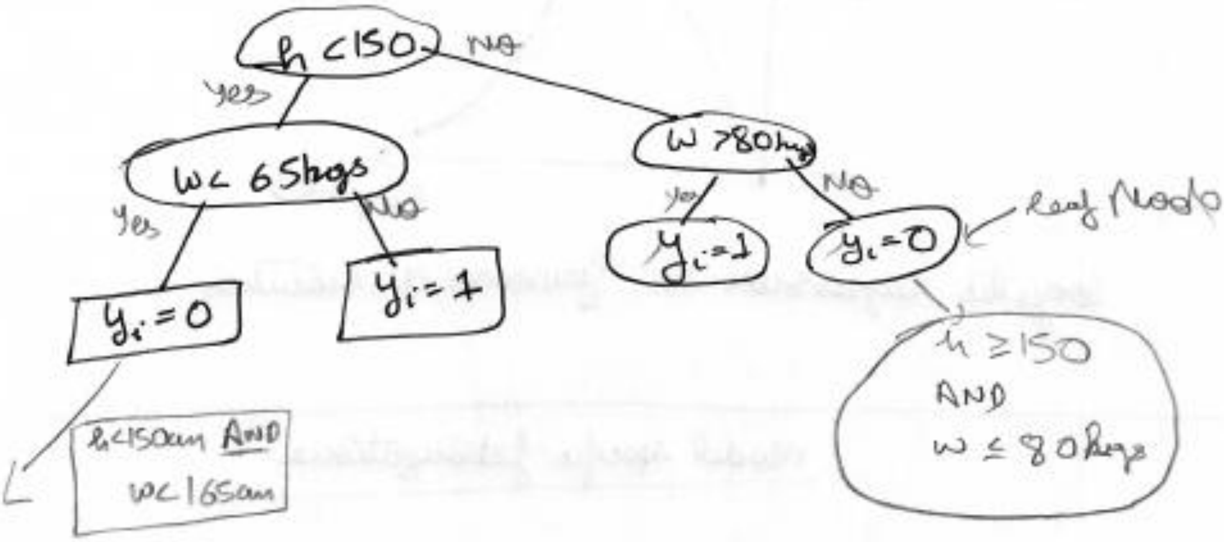
(eg) 2 (a) $h \times w$ (b) $\sqrt{h} + w$ } → mathematical 2-way interaction feature

eg $h < 150\,cm$ And $w < 65\,kg$ AND $hl > 5\,cm$ } → 3way interaction feature / logical interaction

Q Given a task, how do you find good interaction features?

Sol. (DT) $h, w, hL, y_i = \text{gender}$





Mathematical transforms

$x \rightarrow$ single feature

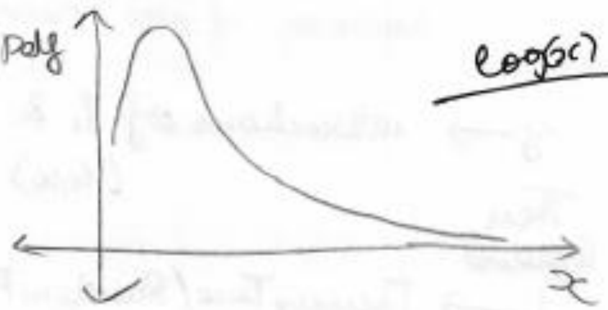
- ✓ $\log(x)$, e^x
- ✓ $\text{sqrt}(x)$; $\sqrt[3]{x}$
- ✓ $x^2, x^3, x^4, \ldots$ polynomial features
- ✓ $\sin(x)$, $\cos(x)$, $\tan(x)$ etc

Q what is best transform among above operations?

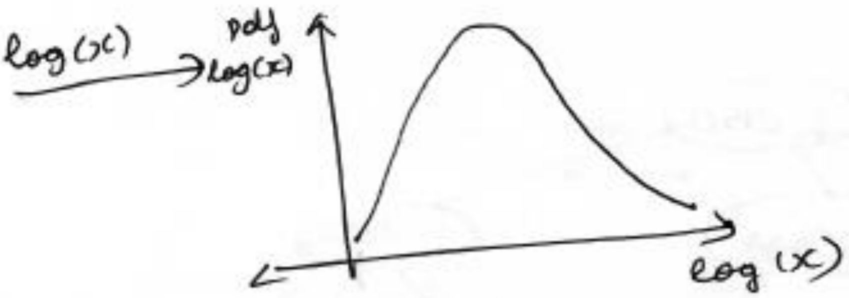
$x \rightarrow$ powerlaw distribution

$\downarrow$

$\boxed{\log x} \rightarrow \{$Gaussian distr$\}$



log(x) → next page



logistic regression is Gaussian distribution

Model specific featurizations

(e.g) $f_1 \longrightarrow$ powerlaw distribution

log-Reg $\longrightarrow$ Gaussian Naive Bayes

features are Gaussian distributed

$\log(f_1)$

we are making this transformations

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

$\underbrace{y \approx f_1 - f_2 + 2 f_3}$ ← you know this from domain knowledge

linear combination of f_i's

As you know this from domain knowledge such cases than linear models are better ie linear regression

In such cases Decision tree may not work well as they work on if else condition

eg(3) $y \rightarrow$ interactions of f_1 & f_2 ← Domain knowledge.
(h,w)

Then (garden)

→ Decision Tree / Random Forest / GBDT

problem like this linear model or linear SVM
may not work well

text Data → Bag of Words → linear models

Bag of Words ↓ Very high dim space → hyperplanes → lr. SVM, logistic regression

+ve −ve

Very high dim space ↓ Sometimes DT may not work v. well

Feature orthogonality

* The more different / orthogonal the features are, the better would your models be.

Case I $f_1, f_2, f_3 \xrightarrow{M} y$

if $f_1 \xrightarrow{\text{correlated}} y$, $f_2 \xrightarrow{\text{correlated}} y$, $f_3 \xrightarrow{\text{correlated}} y$

$f_1 \xrightarrow{\text{n correlated}} f_2$, $f_1 \xrightarrow{\text{n correlated}} f_3$

$f_2 \xrightarrow{\text{n correlated}} f_3$

Case II f_1, f_2, f_3

$f_1 \xrightarrow{\text{cor}} y$; $f_2 \xrightarrow{\text{corr}} y$; $f_3 \xrightarrow{\text{corr}} y$

$f_1 \xrightarrow[\text{correlate}]{\text{not}} f_2$, $f_1 \xrightarrow[\text{cor}]{\text{not}} f_3$

overall impact $f_1, f_2, f_3 \xrightarrow{M} y$ much better than Case I

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

Then impact of f_4 giving good model is high

(idea) $f_1, f_2, f_3 \xrightarrow{M} y_i$

$M :- \tilde{y}_i$

f_1	f_2	f_3	y_i	$\tilde{y}_i$

$\underbrace{\qquad\qquad\qquad}_{x_i}$

Q How do I design a new feature f_4 which is.

$f_4 \xrightarrow{corr} y$

$f_4 \xrightarrow[cr]{not} f_1, f_2, f_3$.

Sol

error:-

$\forall x: \quad e_i = y_i - \tilde{y}_i$

$f_4 \xrightarrow{Correlated} e_i$

Try to make new feature f_4 correlated with e_i

Domain specific featurizations

heart attack → ECG data

important to research & study

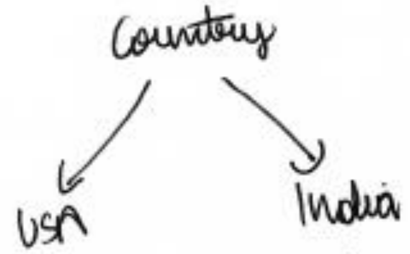
existing featurizations

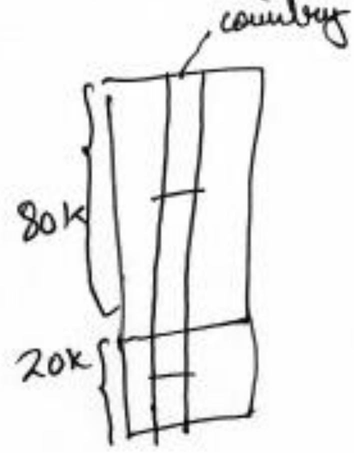
doctors / Specialists

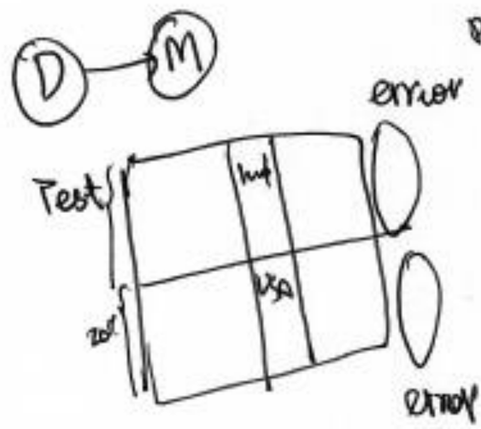
new features

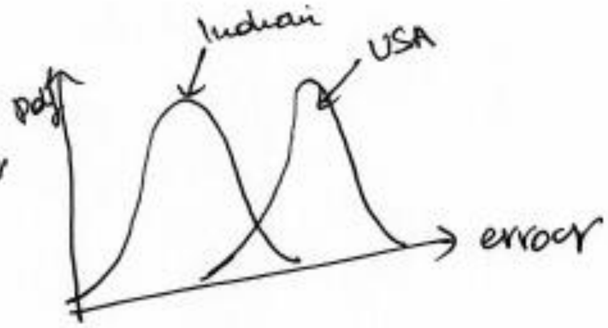
Research the domain before making models

Feature slicing









strategy → D

$$D \rightarrow \underset{80K}{D_{India}} \rightarrow M_1$$

$$D \rightarrow \underset{20K}{D_{USA}} \rightarrow M_2$$

slice data using features

Calibration of models: Need for calibration

2class classifications $\quad y_i \in \{0,1\}$

$$D = \{x_i, y_i\} \longrightarrow \text{Model } \boxed{f(x)}$$

$$x_q \ \boxed{f(x_q) = y_q}$$

$D_{tr} = \{x_i, y_i\} \rightarrow$ Mode $f(x)$

$P(y_q = 1 | x_q\ f)$