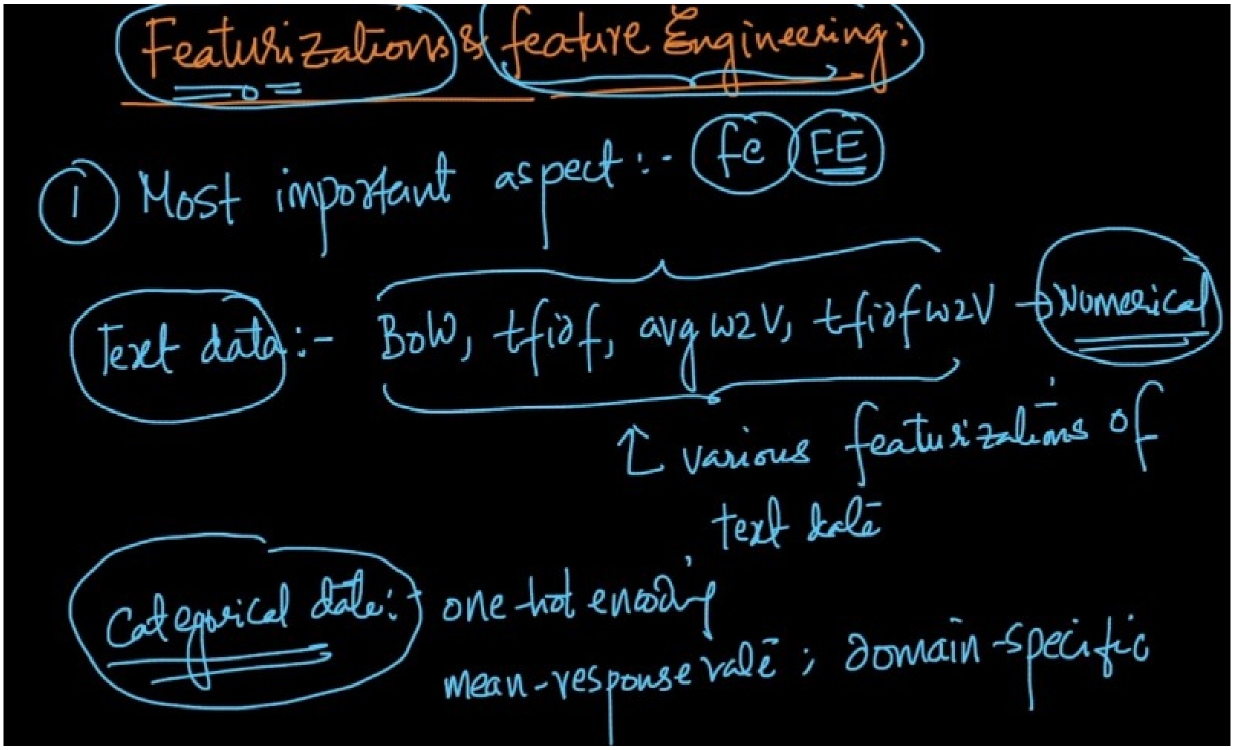
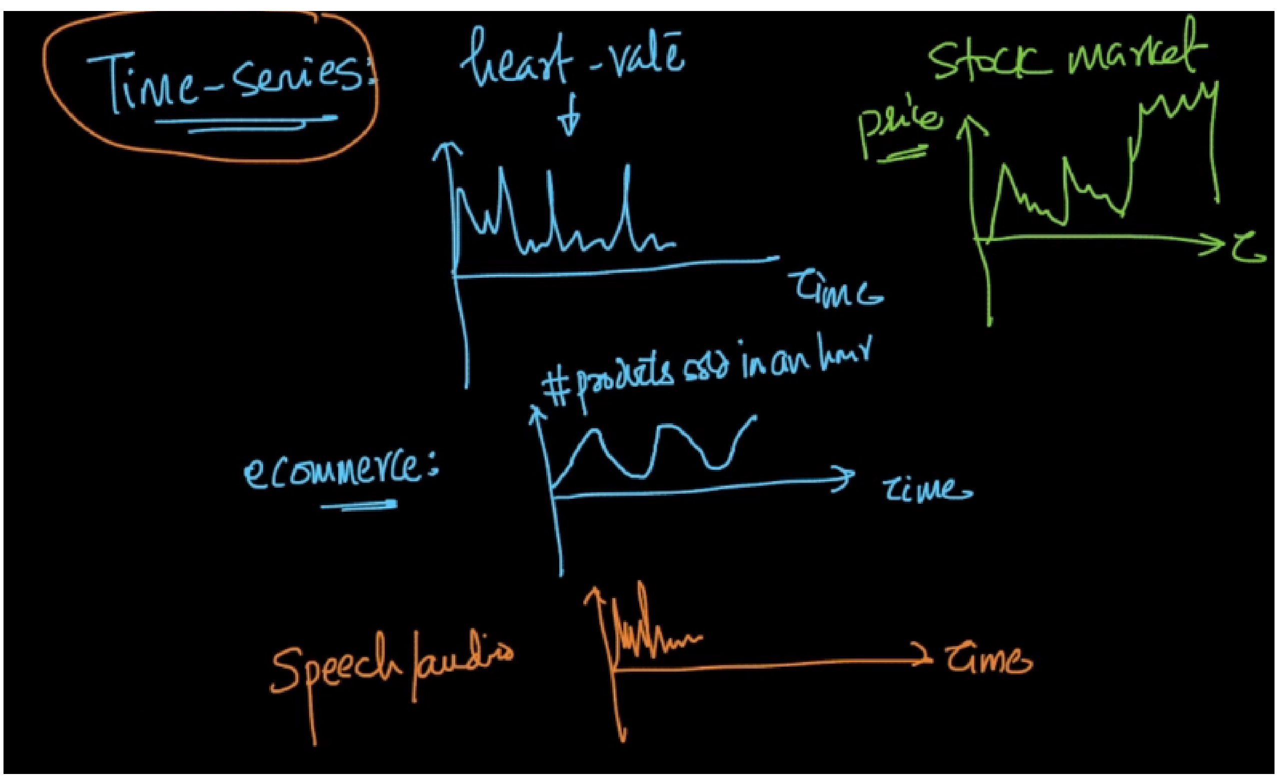
Featurisations and feature engineering: Text data is converted to these following features to handle.

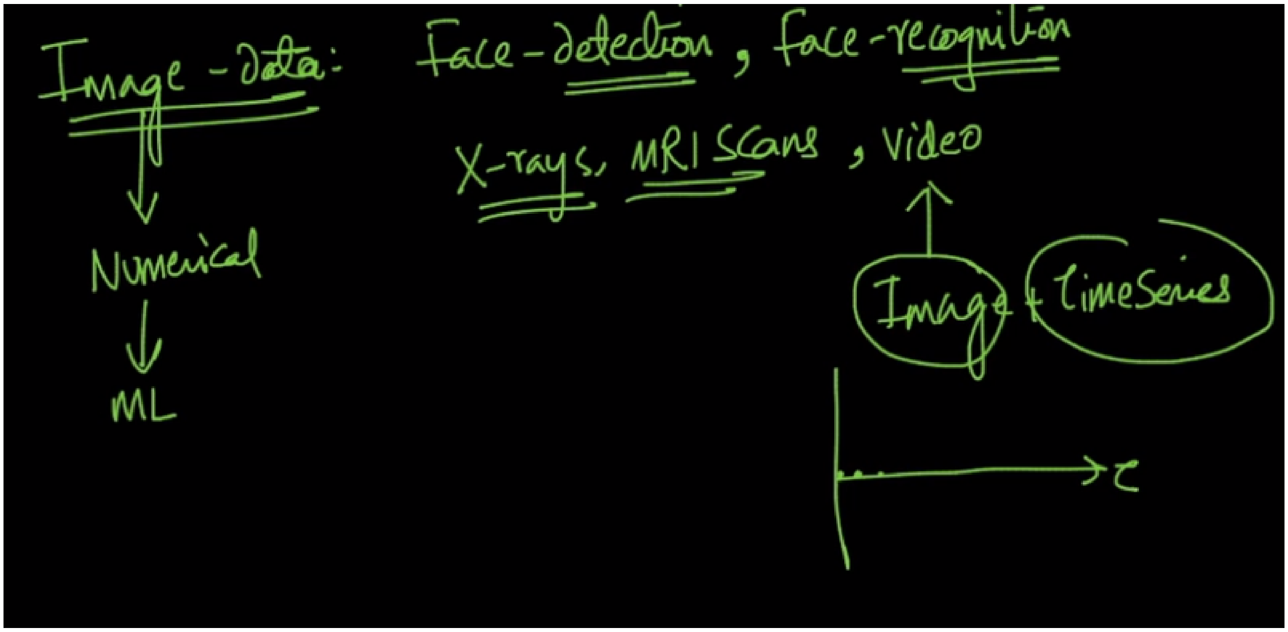
All Algo's process the Numerical data.
For categorical data we use one-hot encoding, mean response rate, domain specific ways of handling the data.



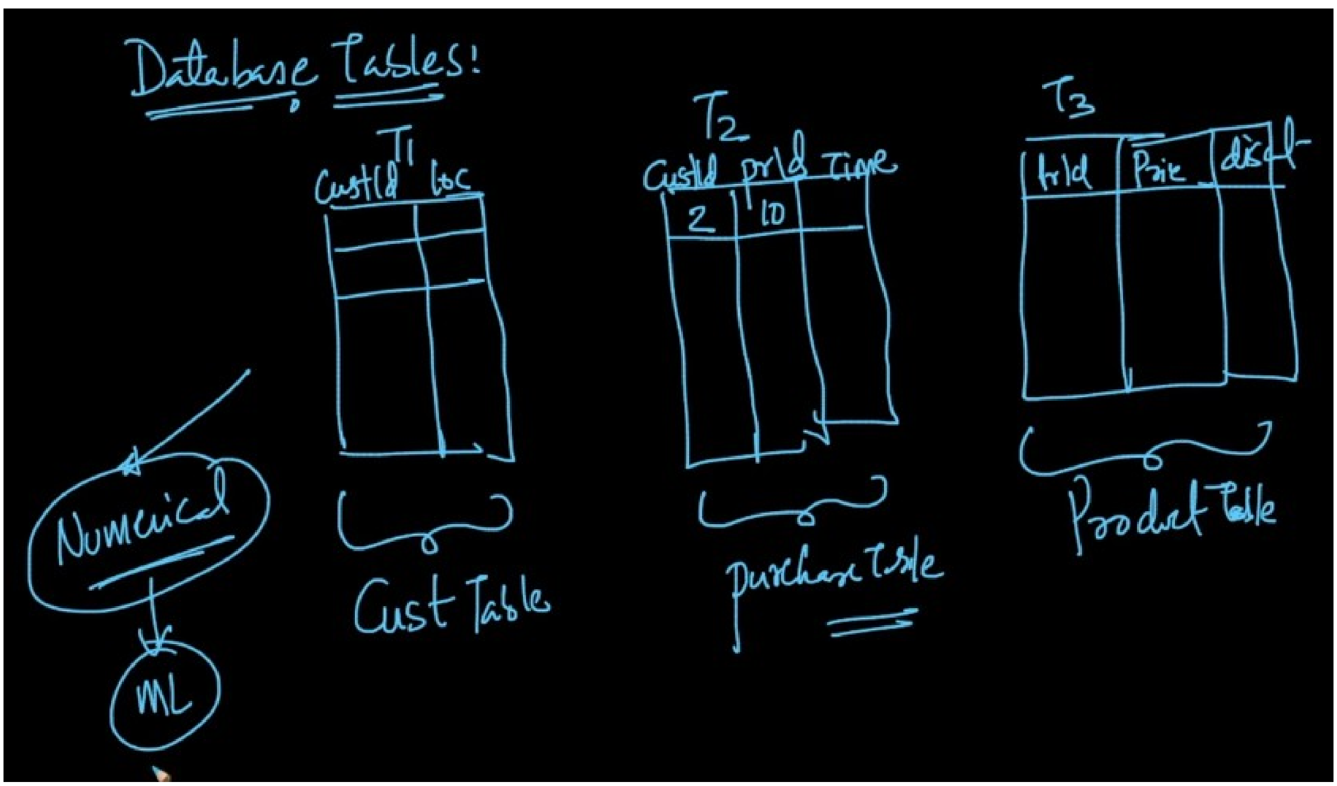
Time – Series data: Heart rate pattern of time series, Stock market data.



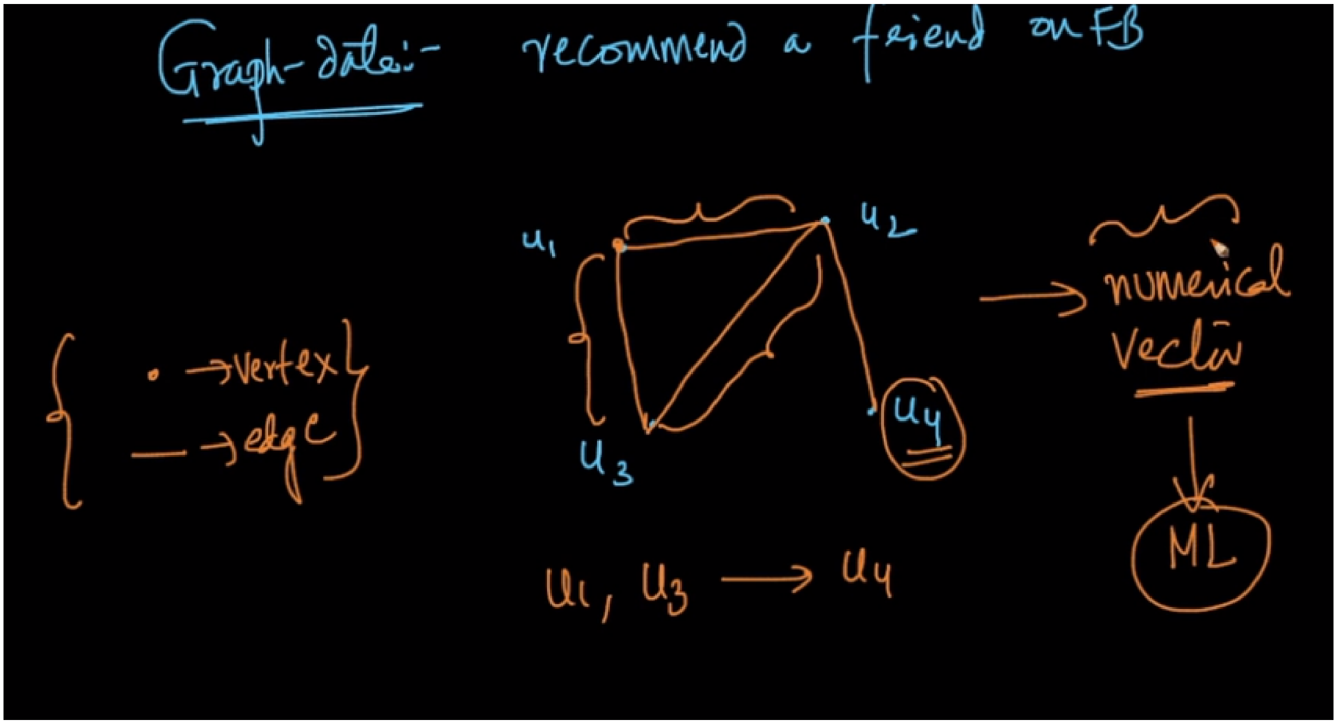
Text can also be the sequence data, the words in the data can be taught of a sequence.



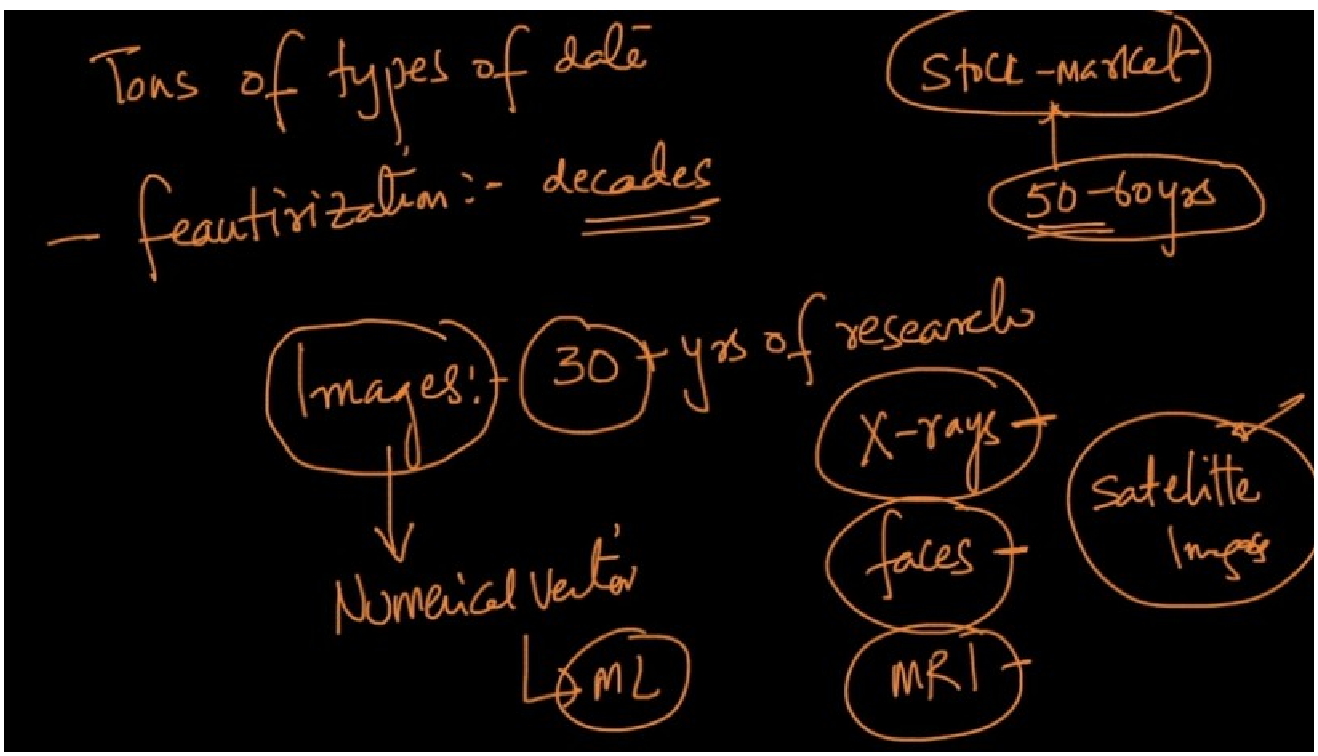
Data from database tables:



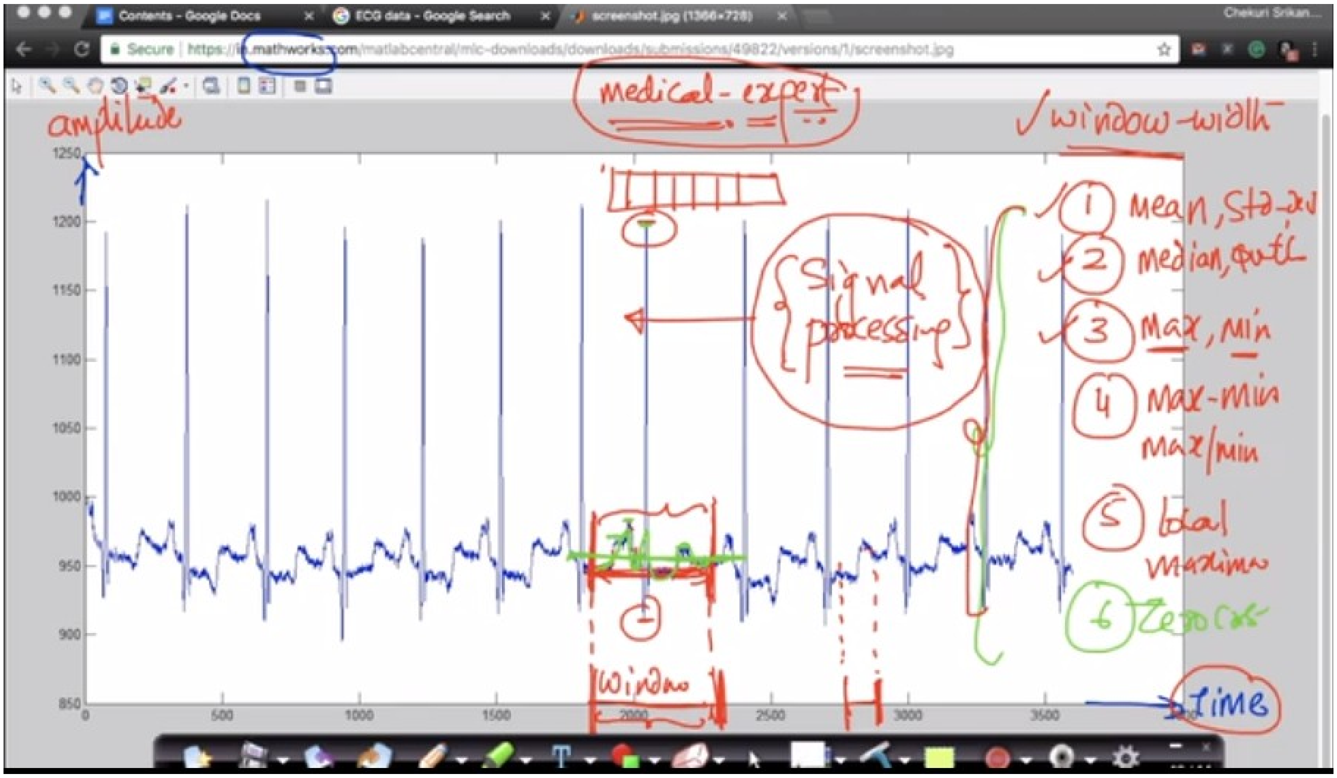
The other type of data is called the graph data – On facebook each friend can be taught of a vertex and each line can be taught of an edge in the graph.



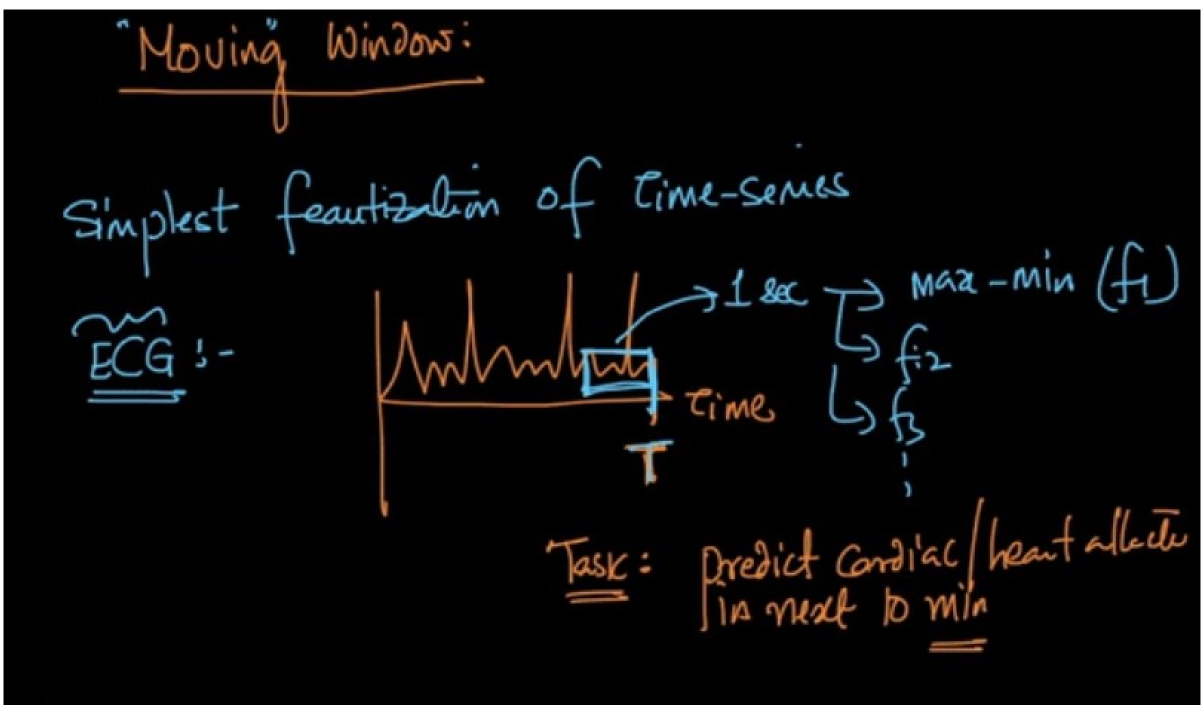
Edges are the relationships of the people. How many common friends can be.



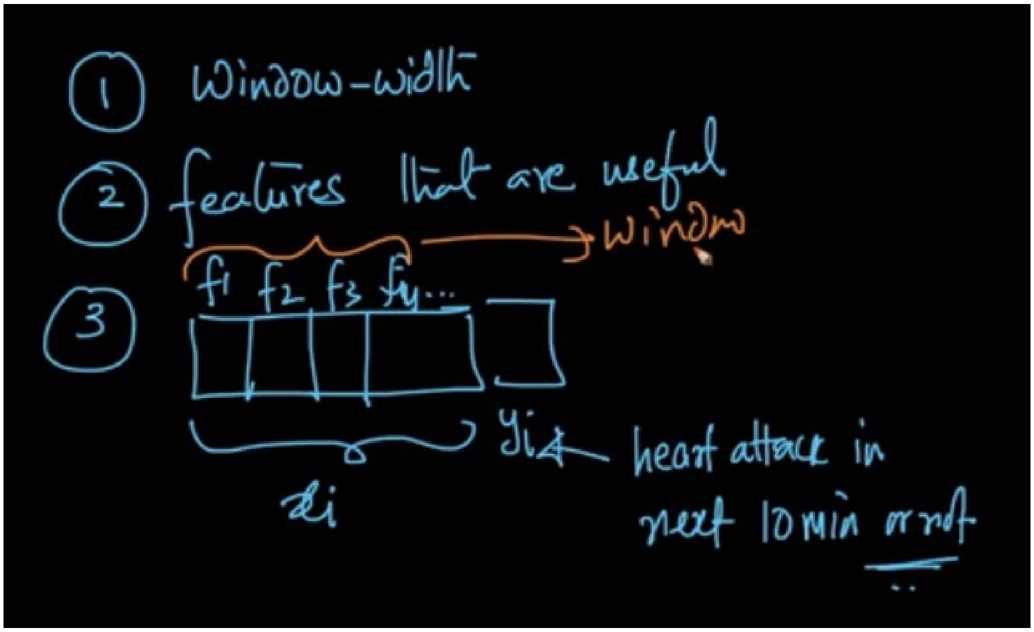
Moving window for Time Series data:
Example: ECG.
Moving window takes the window of certain width, right window width is the problem specific.
Each value in the window can be put in a vector.



Task: Will the patient get heart attack in the next time-stamps.

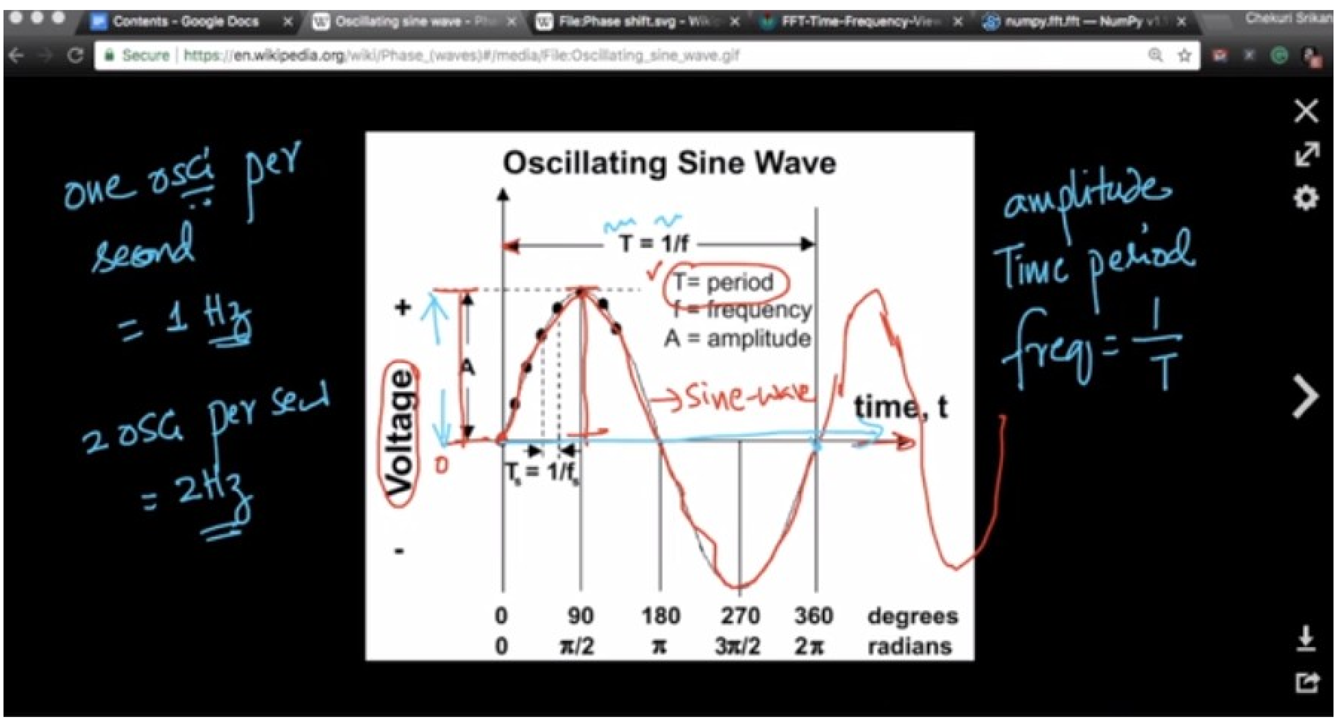


Steps of featurizing of time series data:



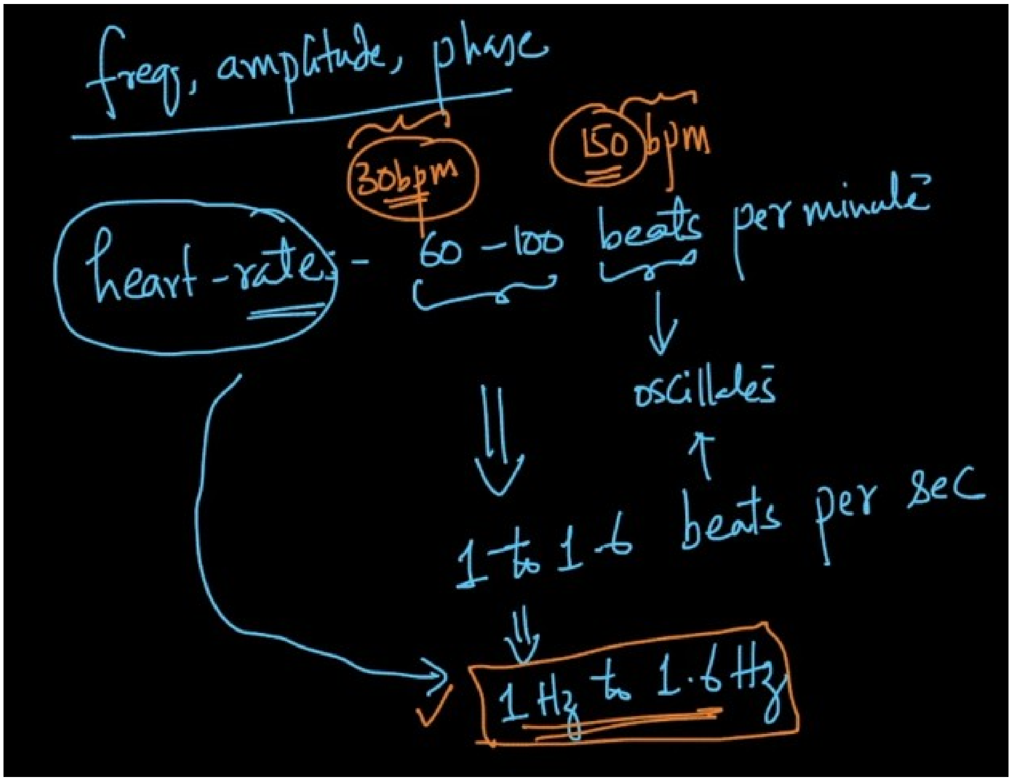
The each window has these features.

Fourier decomposition:

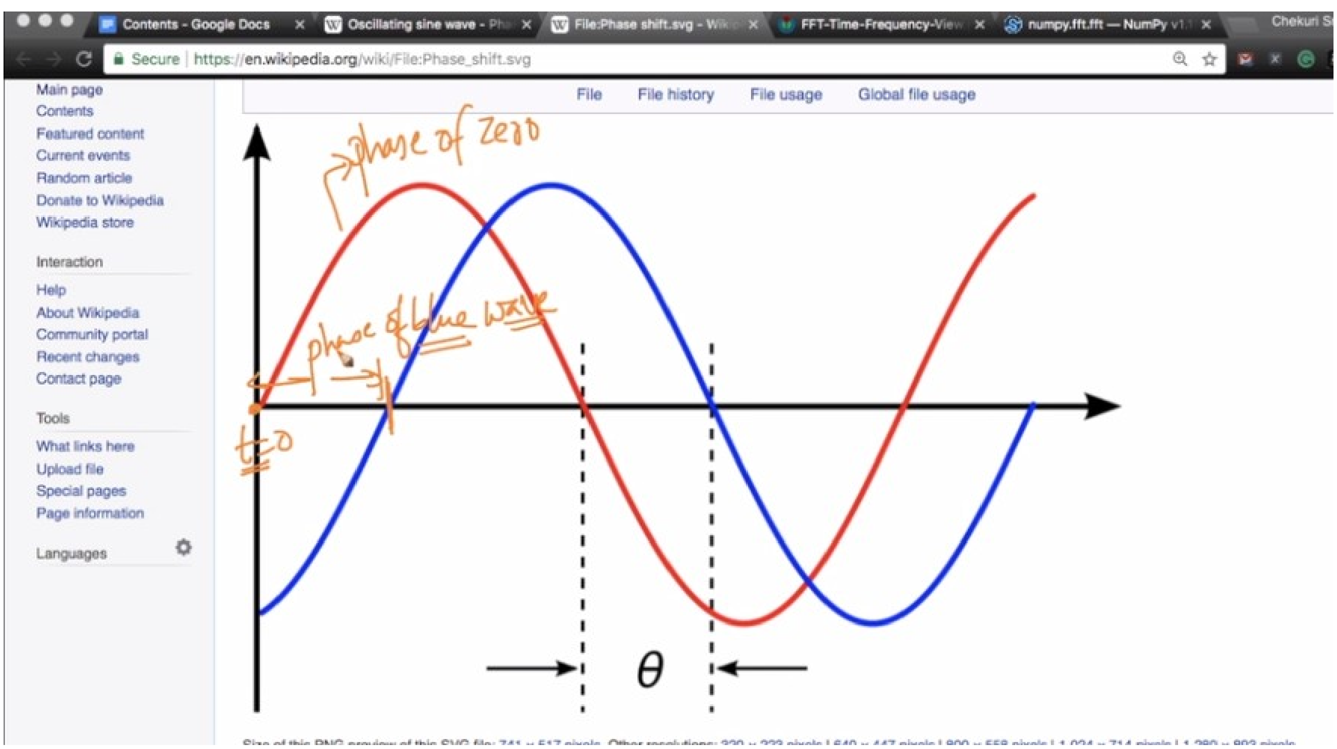


Frequency, amplitude, phase -

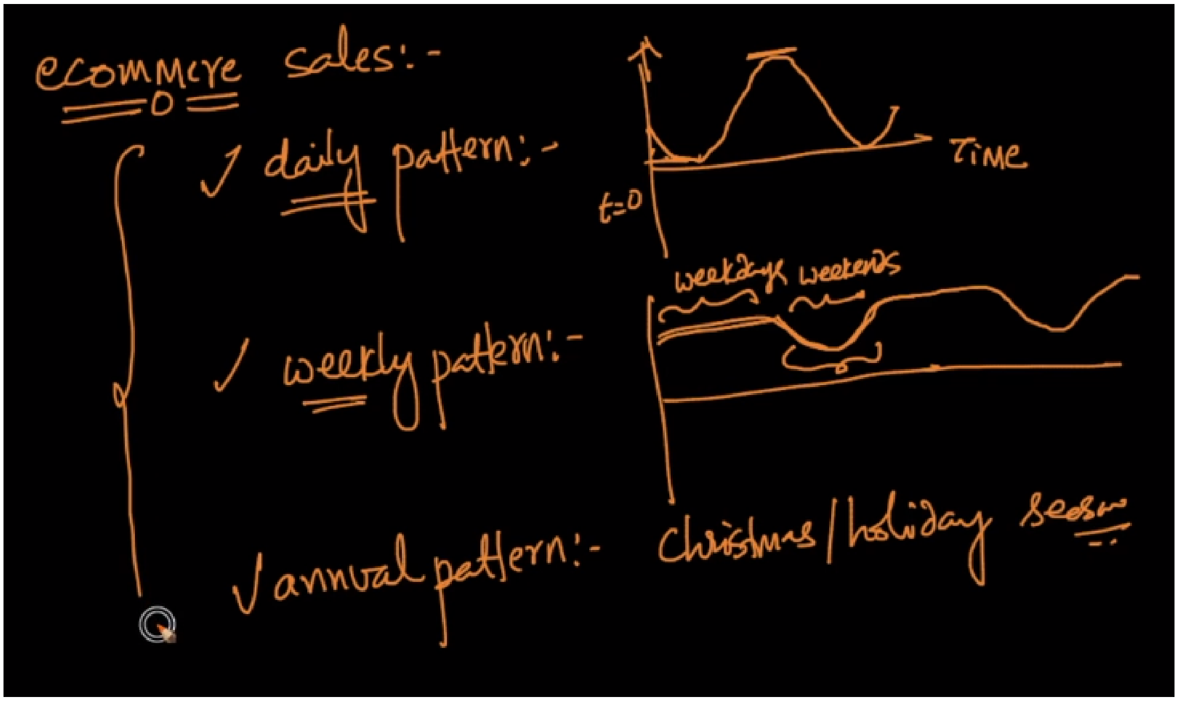
Hz is the number of oscillations completed in one second.



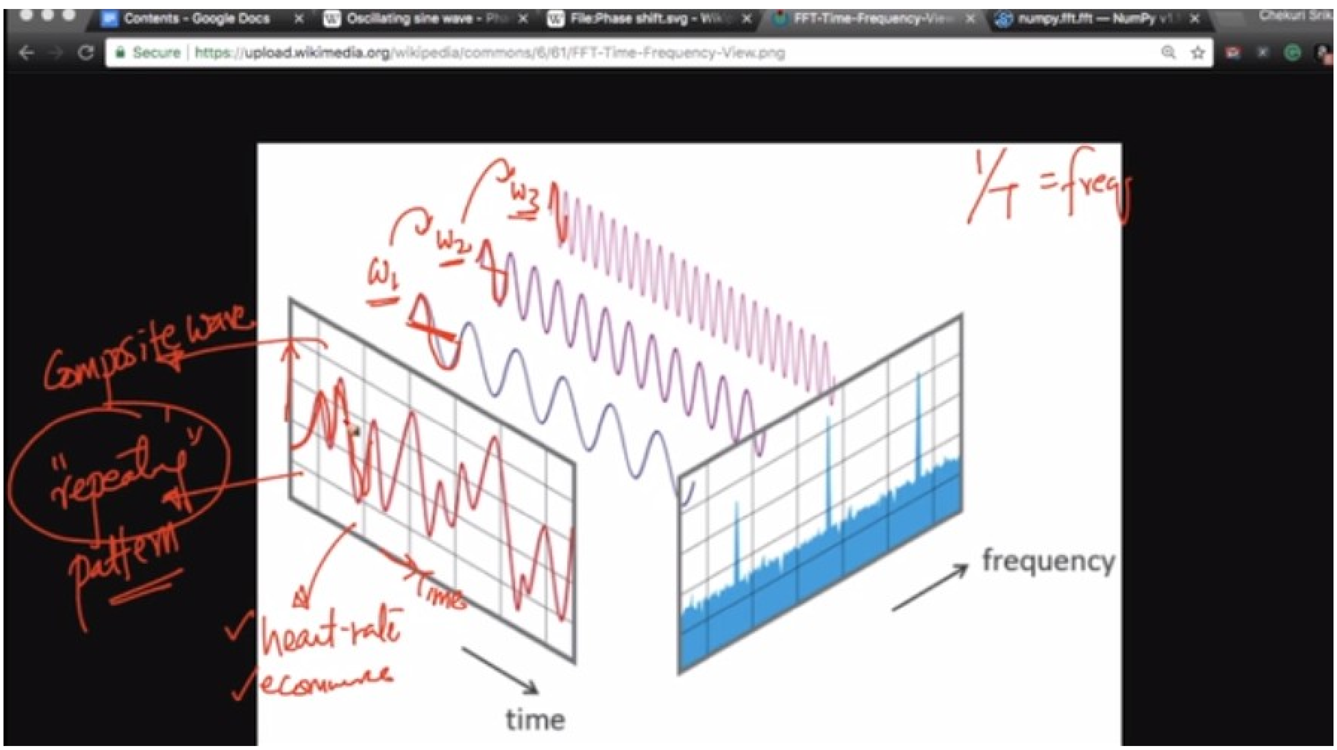
Phase of the wave:
This is called the phase of the wave.



Common patterns in E – commerce sales:



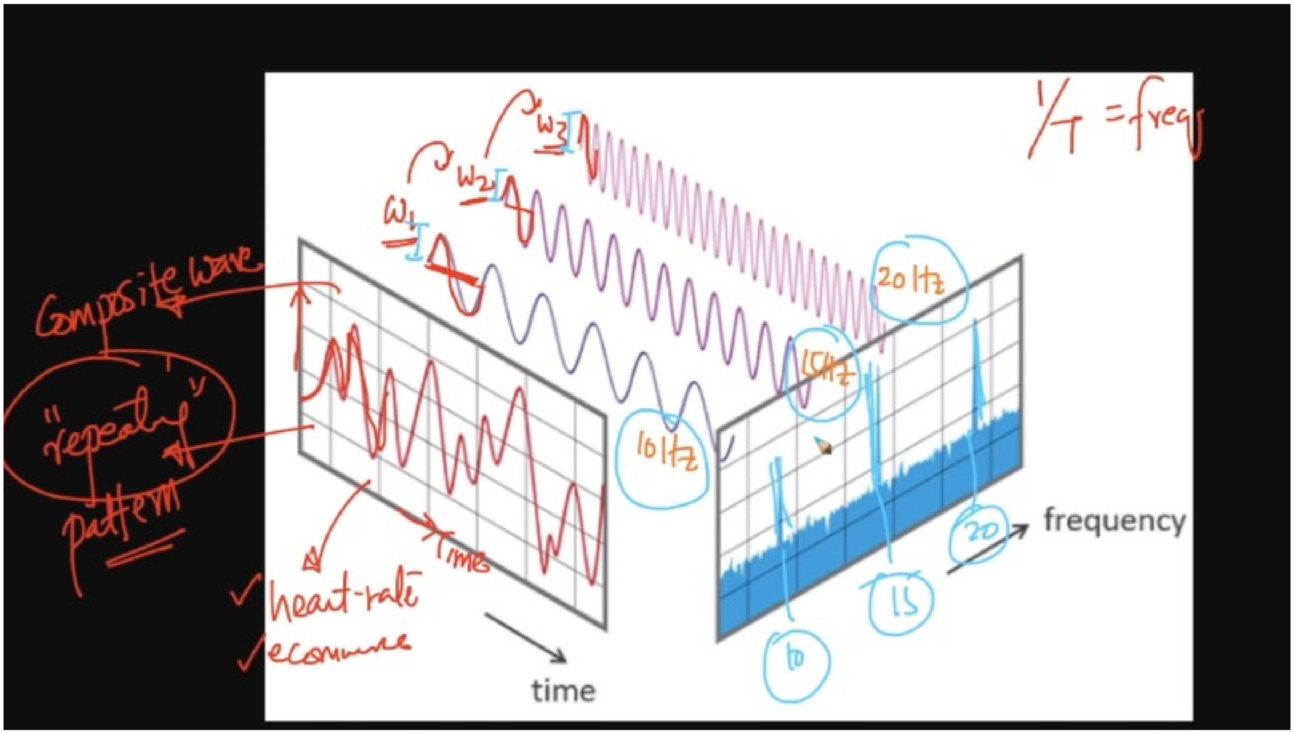
If there are repeating patterns in the data then we can express the data as the sum of different waves(w1, w2, w3, ...) that make the actual wave.



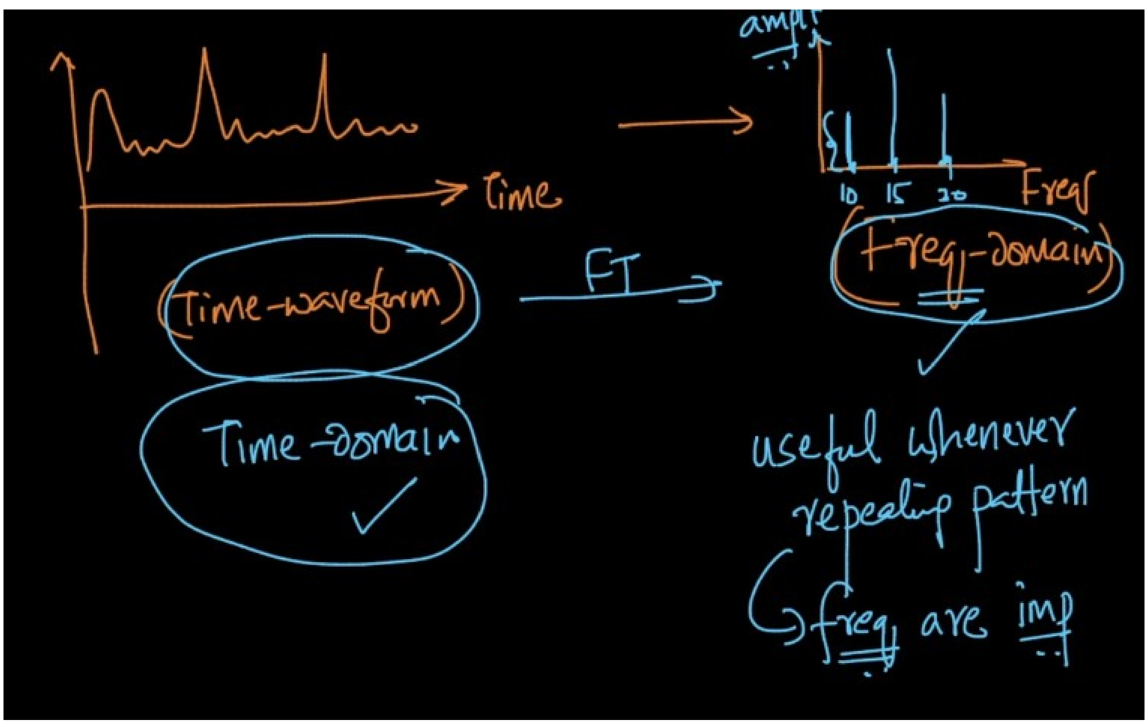
This is called the time form of the data, We can move to the frequency type of data.

The Fourier transform of the data can be done to several different types of waves of various amplitudes.

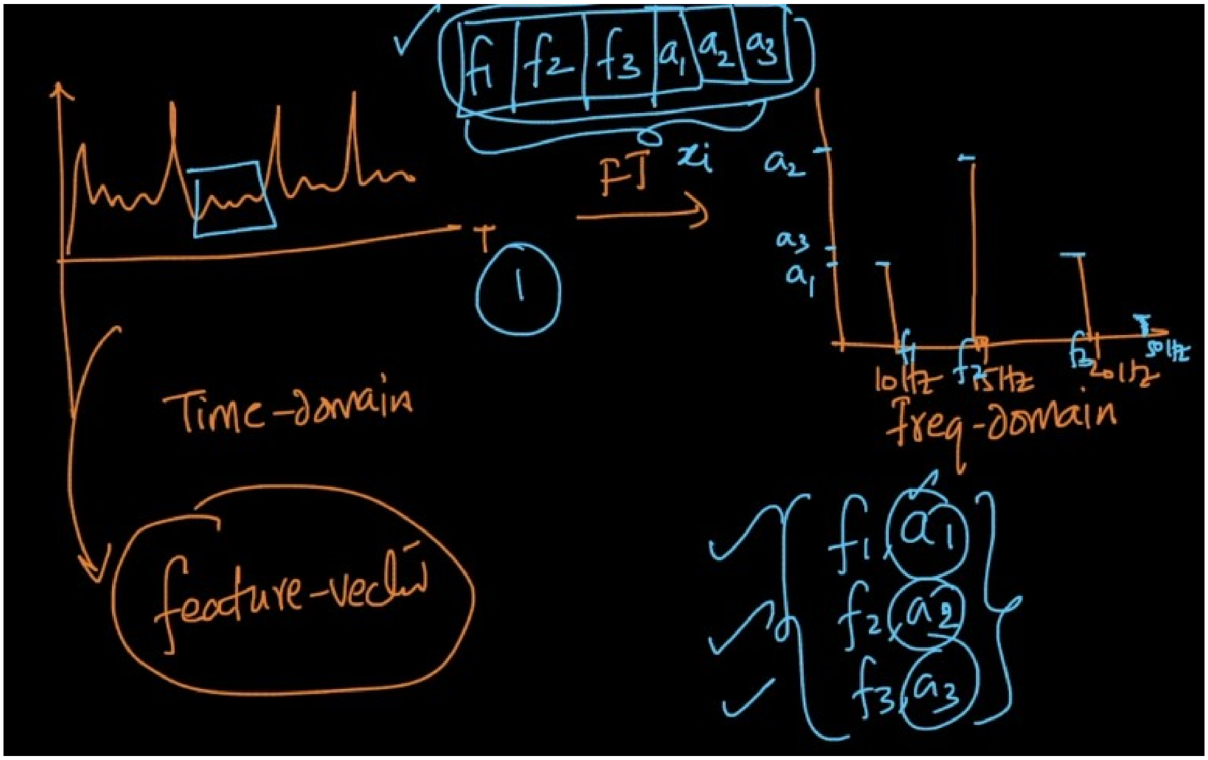
We can change the wave form to the frequency domain of different frequencies(10 Hz, 20 Hz, …).



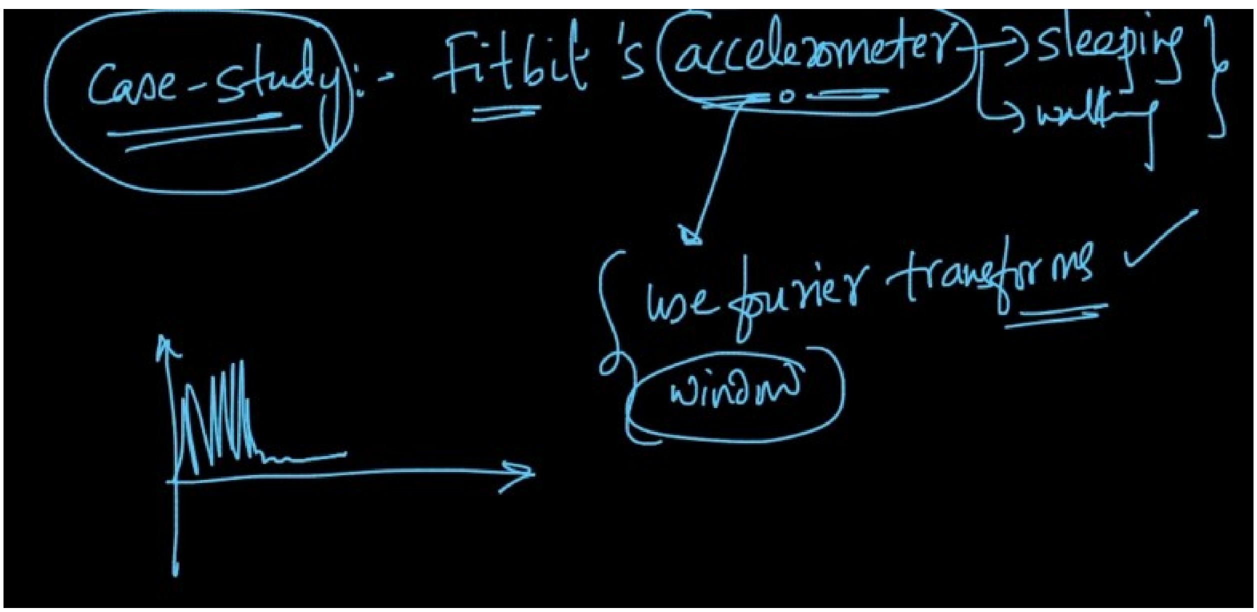
When ever there is a repeating pattern, frequencies are important.



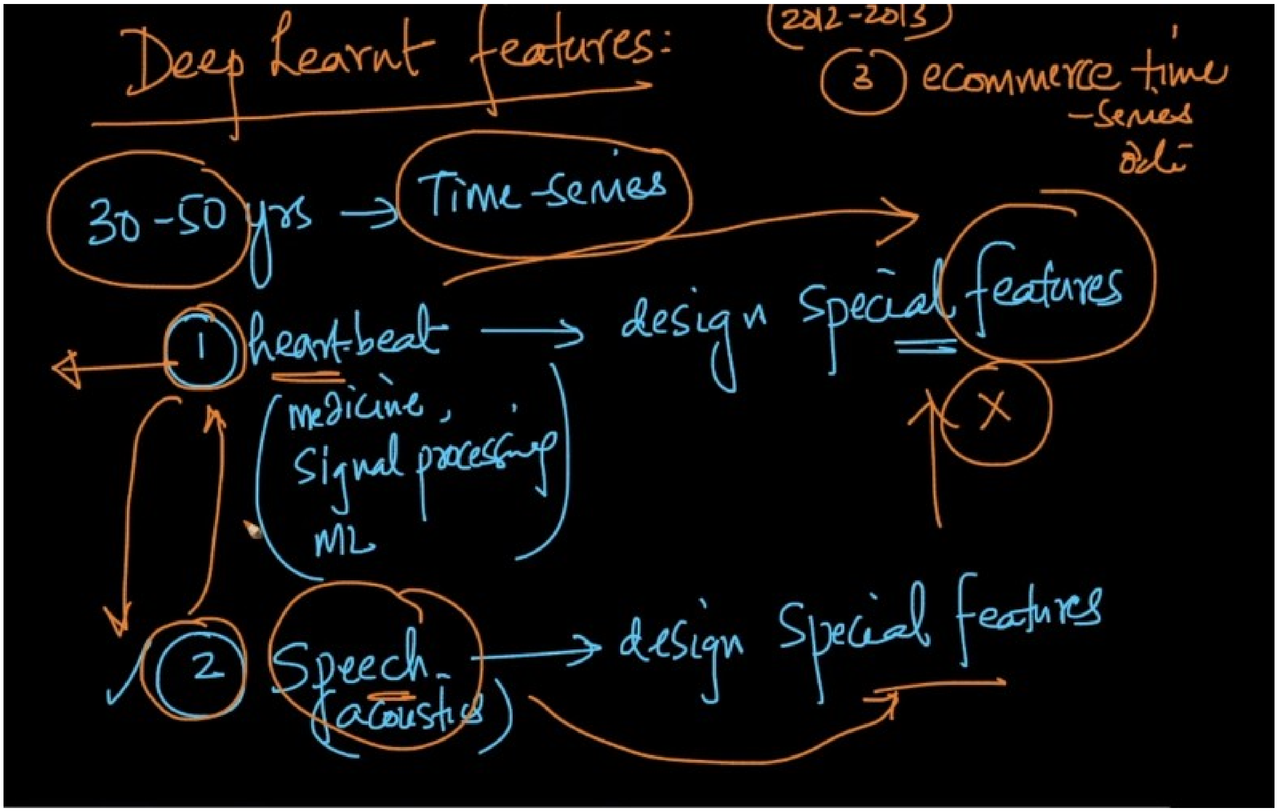
Time – series data:
Mathematical conversion of the data from wave to frequency domain.



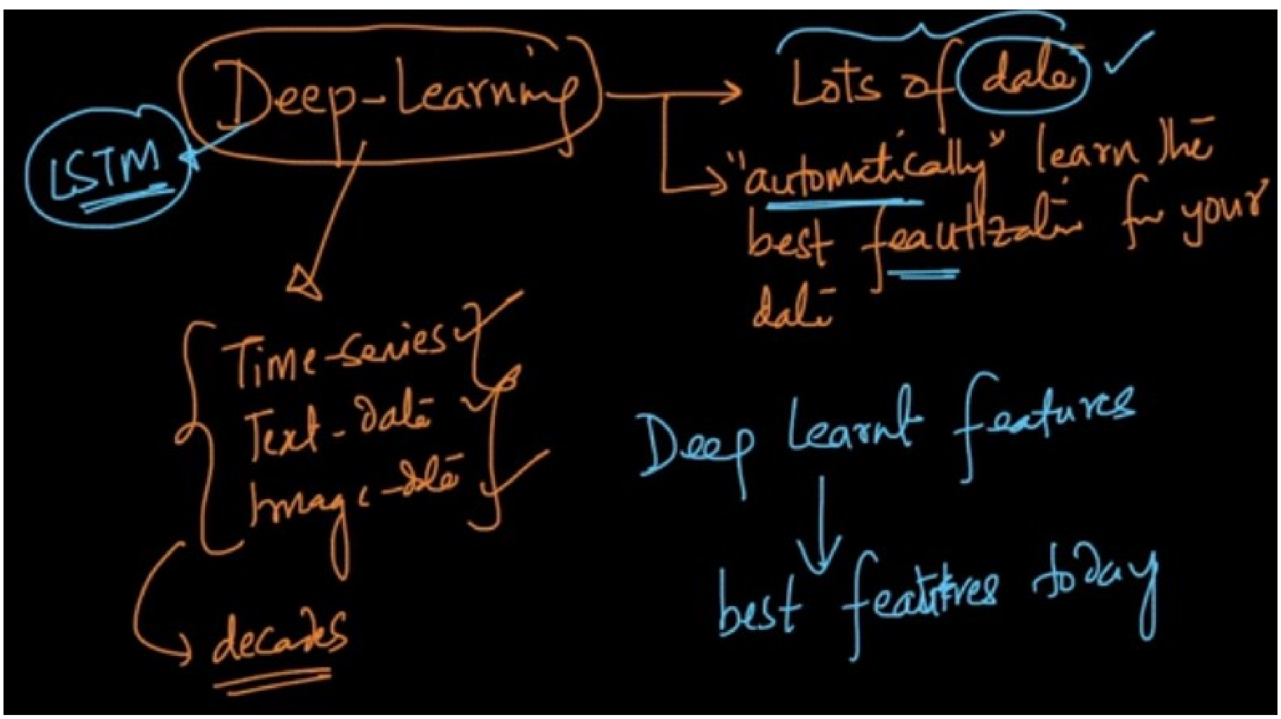
This is called the Fourier representation of the data. This is called the frequency of the data.



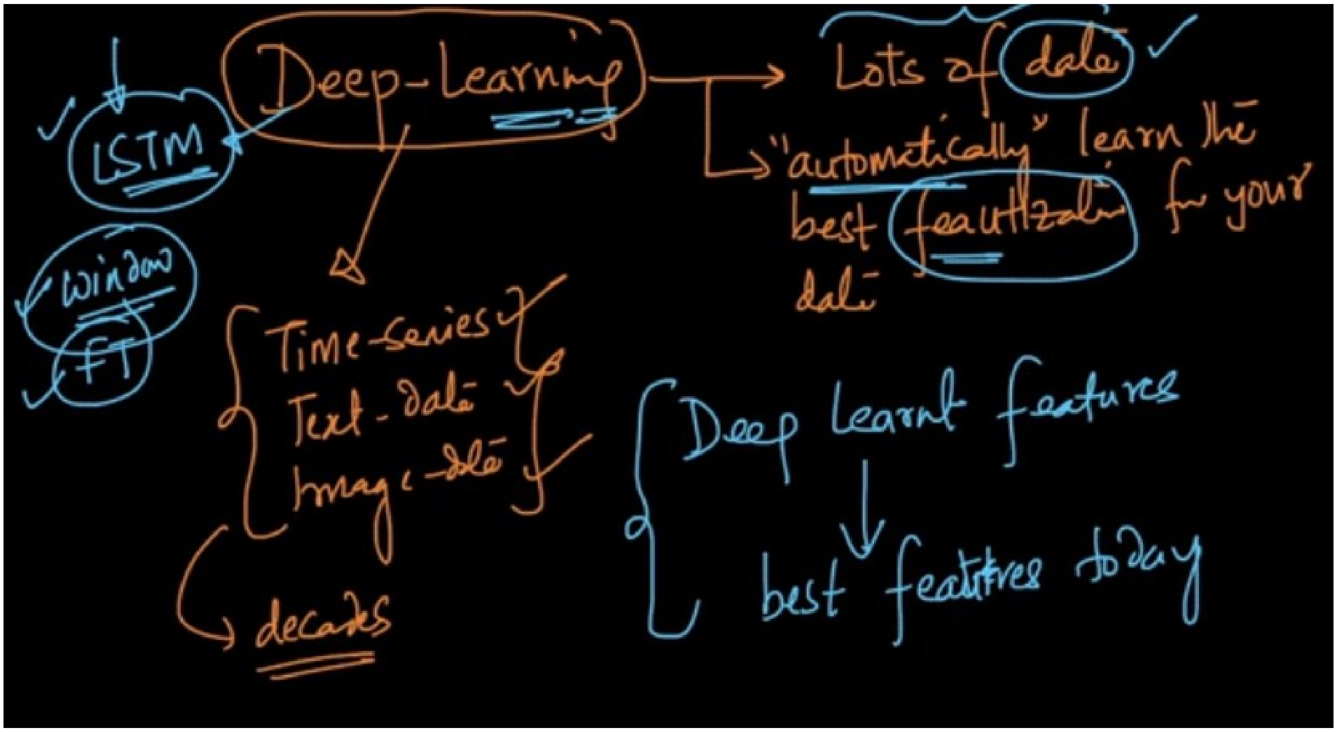
Deep learning features: LSTM



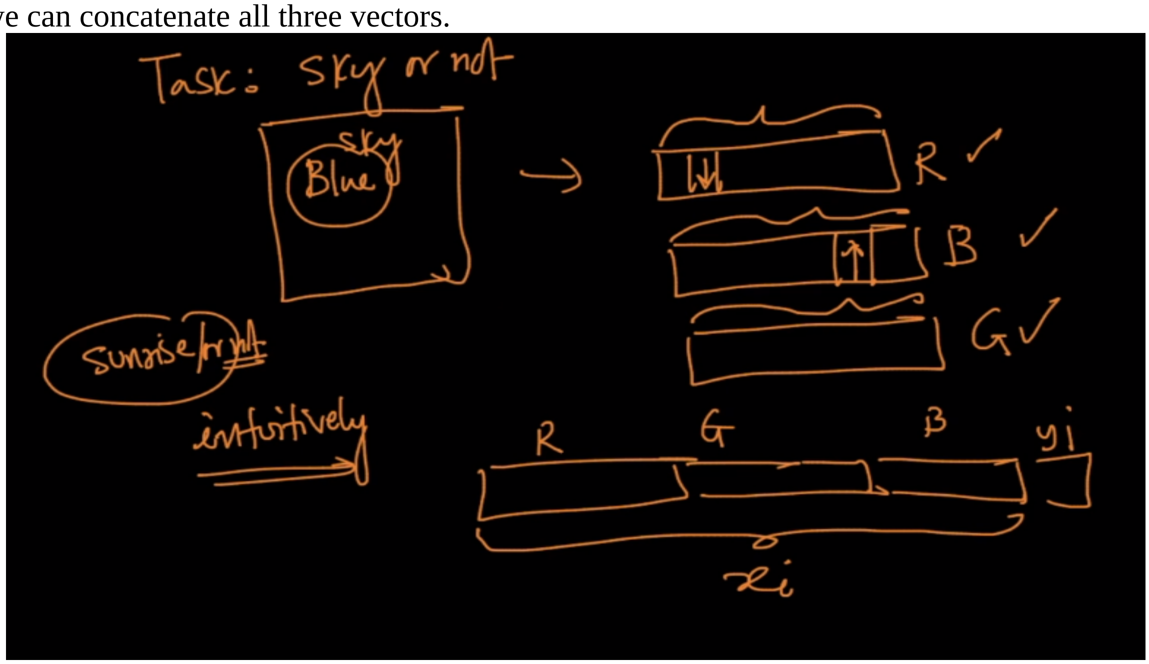
Deep learning takes lots of data, then it can learn the best features for your data.



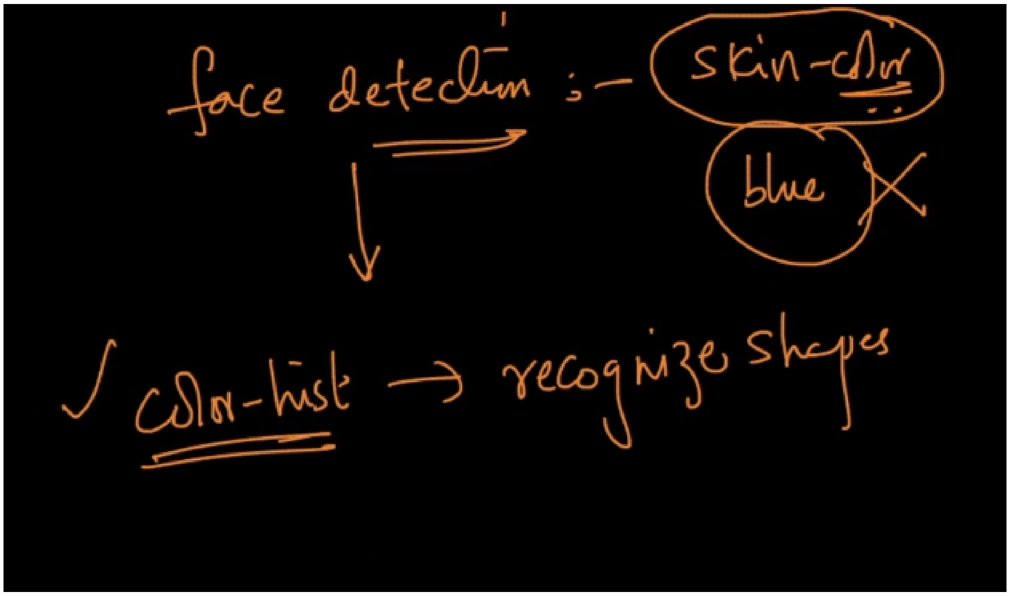
Instead of using the right features for a particular problem, we can use the LSTM's for making the best features.



Today's application are mostly done in deep learning.
Image histogram:
Images: Faces, Object, scans, X – rays, autonomous cars.
Take the Red values for each pixels in the image, then we can plot a histogram of the Red color. I can convert the histogram to a vector.
Similarly can also be done to a green and blue colors of the image.
We get three vectors for every image.
Now we can concatenate all three vectors.

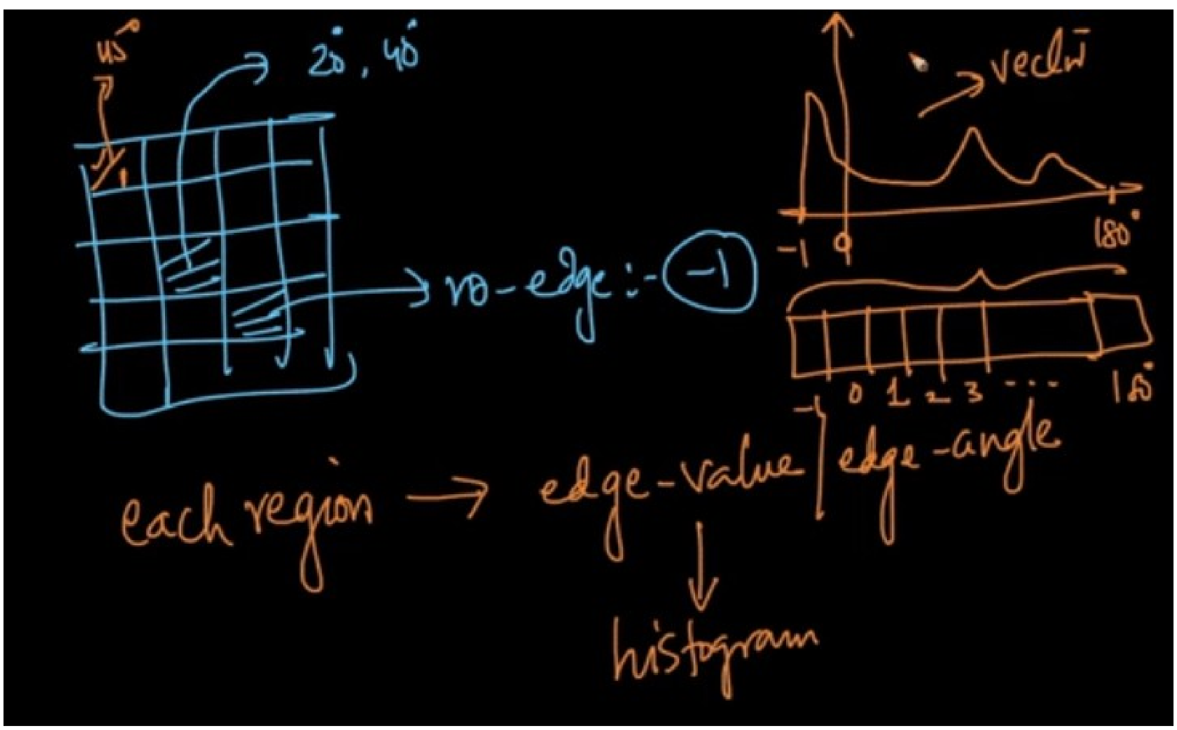


For a sky blue color then there will be more blue color in the image.
For skin color, we don't observe more blue color.

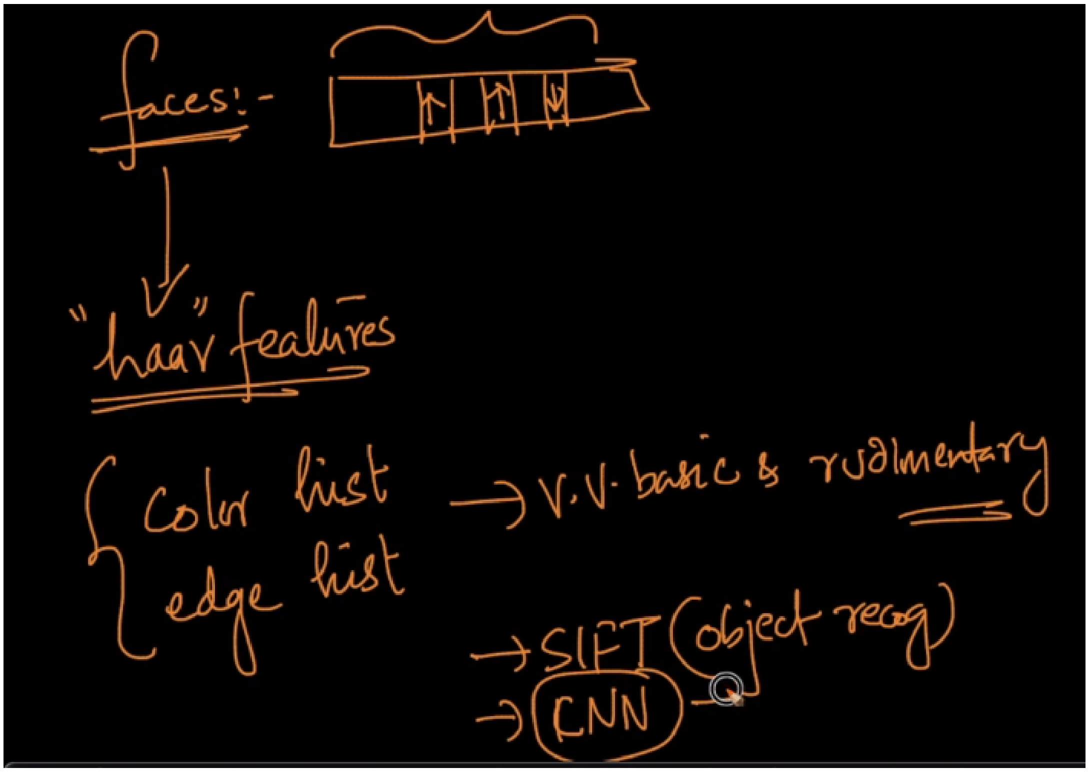


Edge histograms:
We break up this big image into a grid. And calculate the angle of the edge on the grid.
We take the dominant edge as the angle for that grid.

For each region, we can get the edge – value/ edge – angle. Now, I can plot the histogram of the angles.

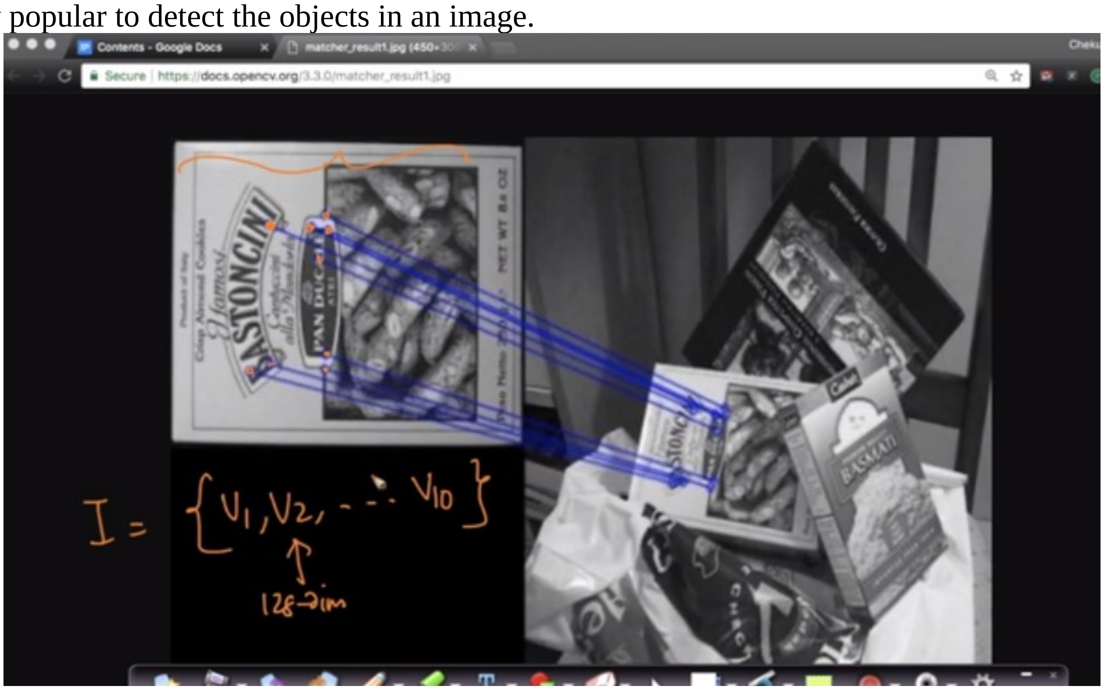


In cases of faces, the edge angles can easily be distinguished and therefore we can make the conclusion of edges in the image.
Haar features for image recognition:
The color histograms and edge histograms very basic.

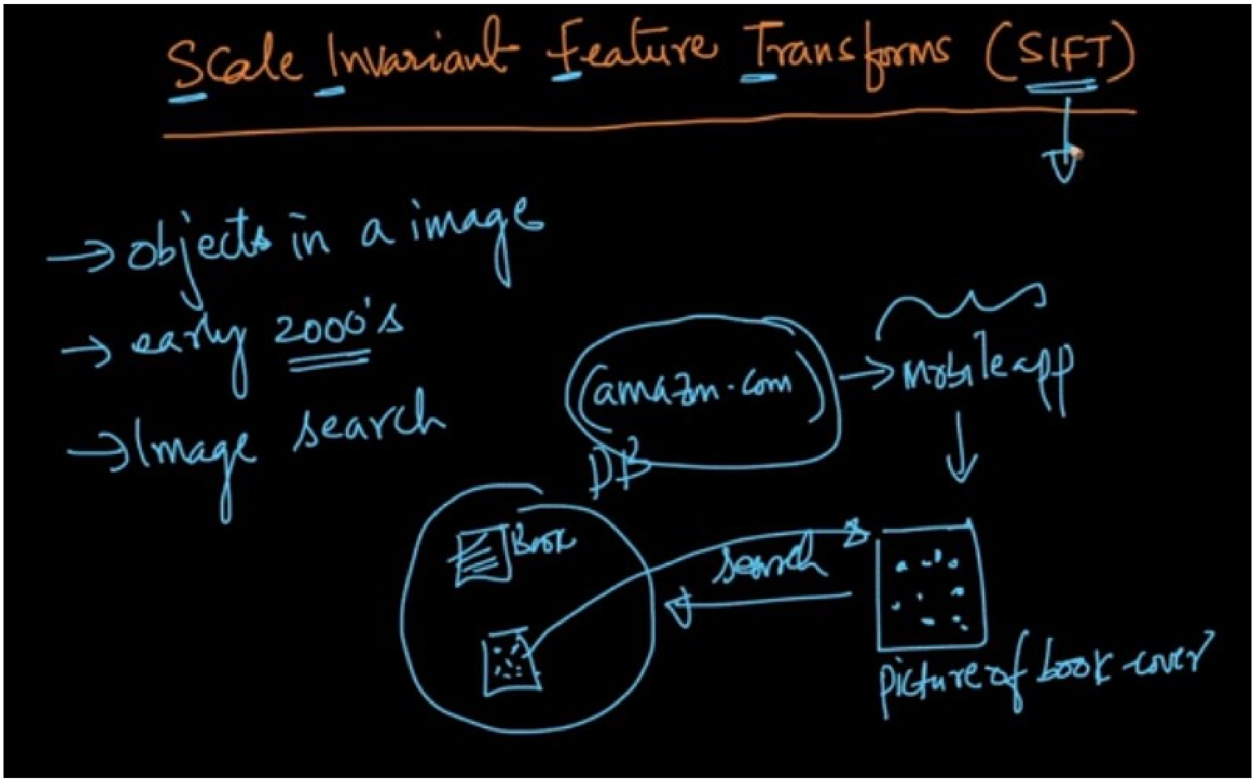


Keypoints: SIFT(Scale invariant Feature Transforms)

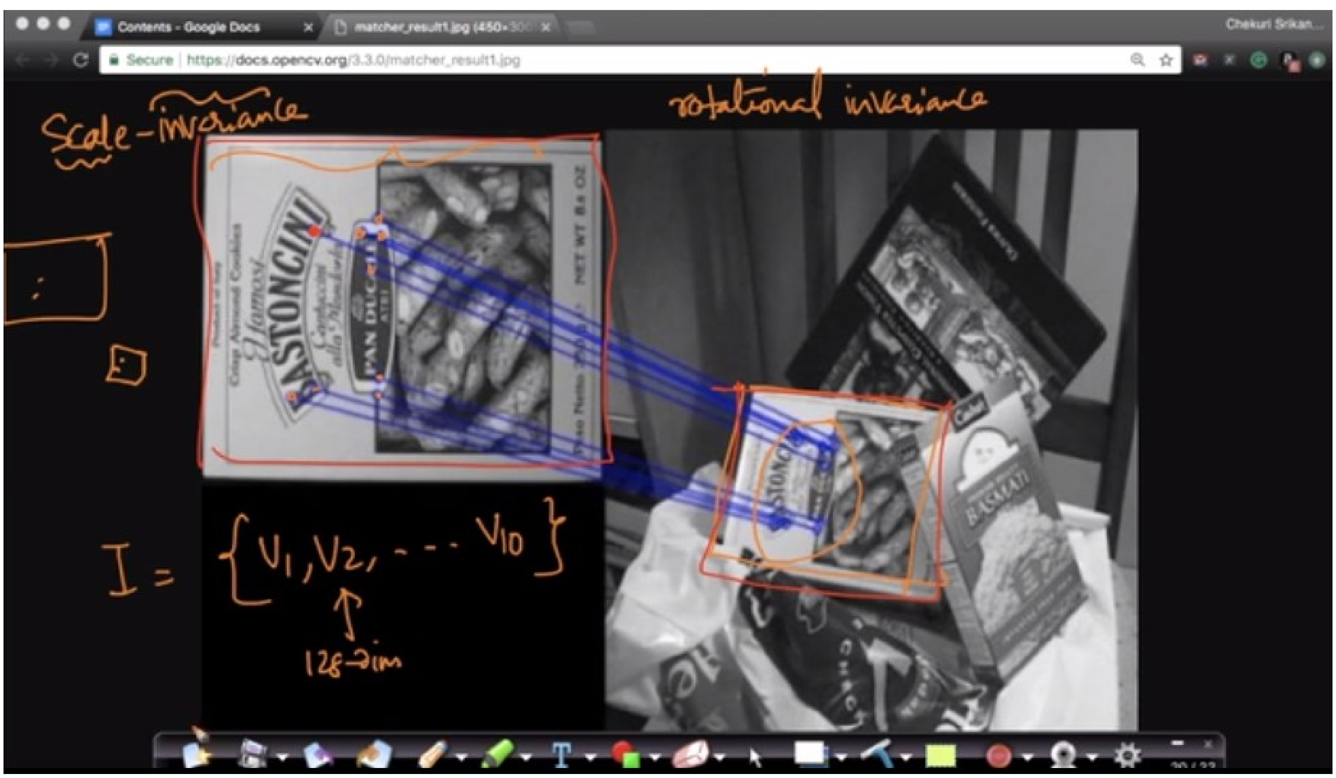
It is very popular to detect the objects in an image.



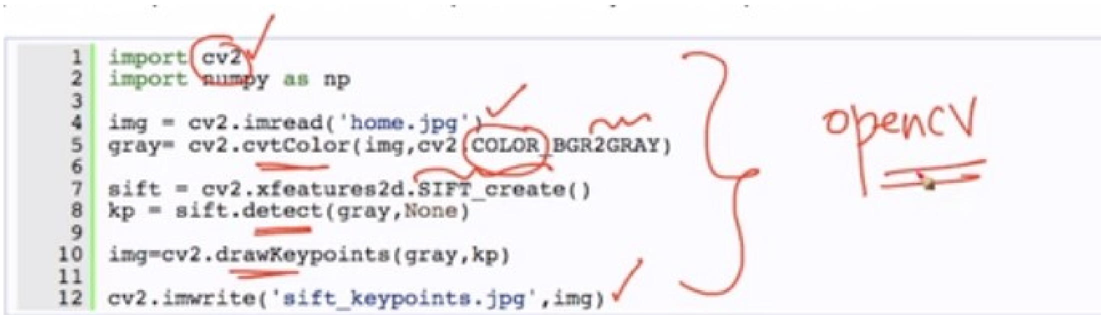
We can take the key points in the picture of the book cover and search for the key points on the data base, match come up with the matched product.



It is a scale invariant feature transform. That size of the object in the image does nor matter. It does not change much, rotational in variance.



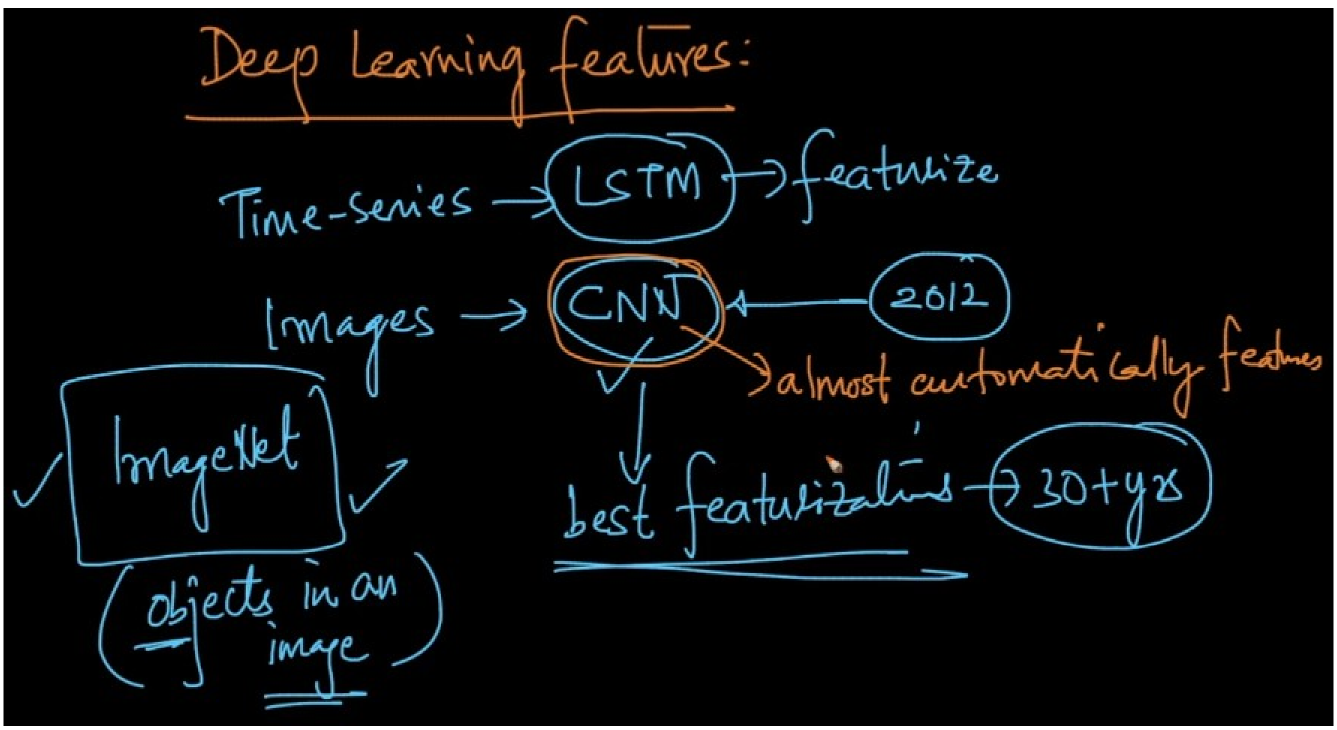
This has the very nice property called rotational slightly and change in shape.
We can use OpenCV for making the SIFT features.



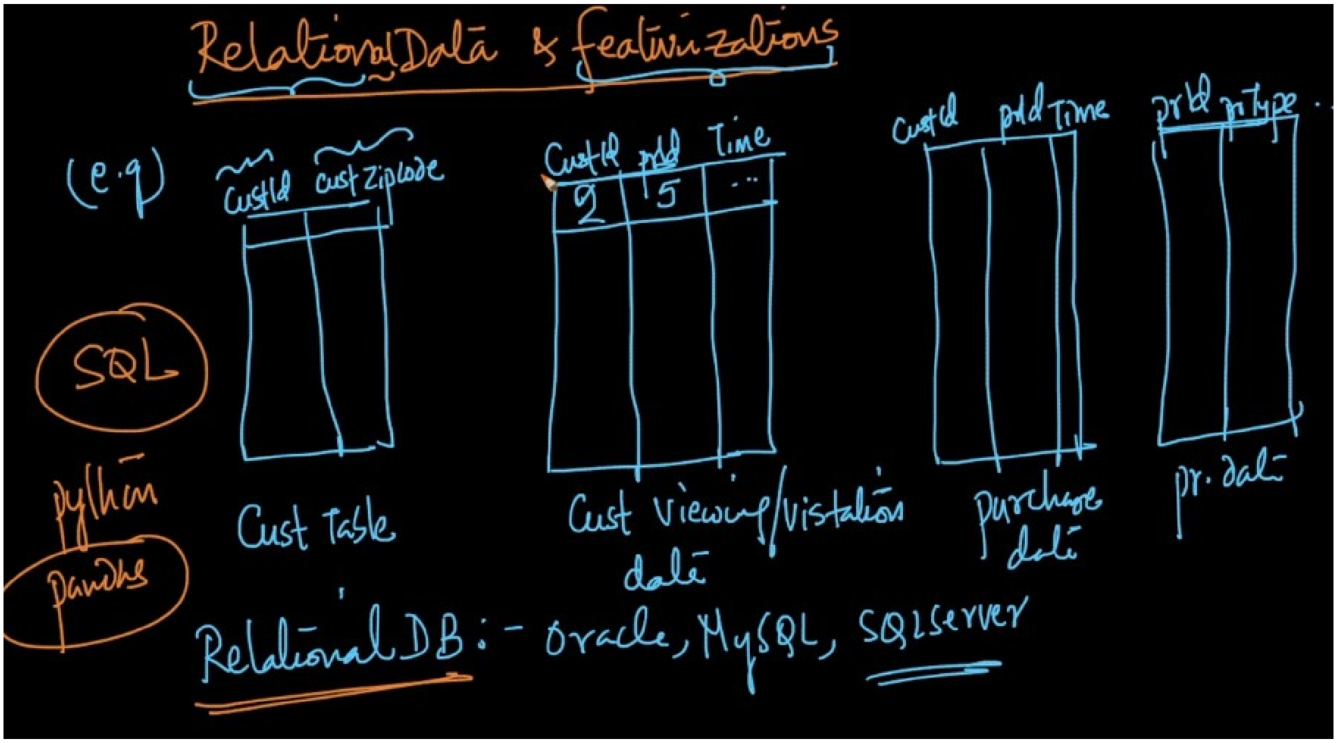
Deep learning features: CNN

For time series data we have LSTM to featurize the data. For image data we have the CNN best featurization.

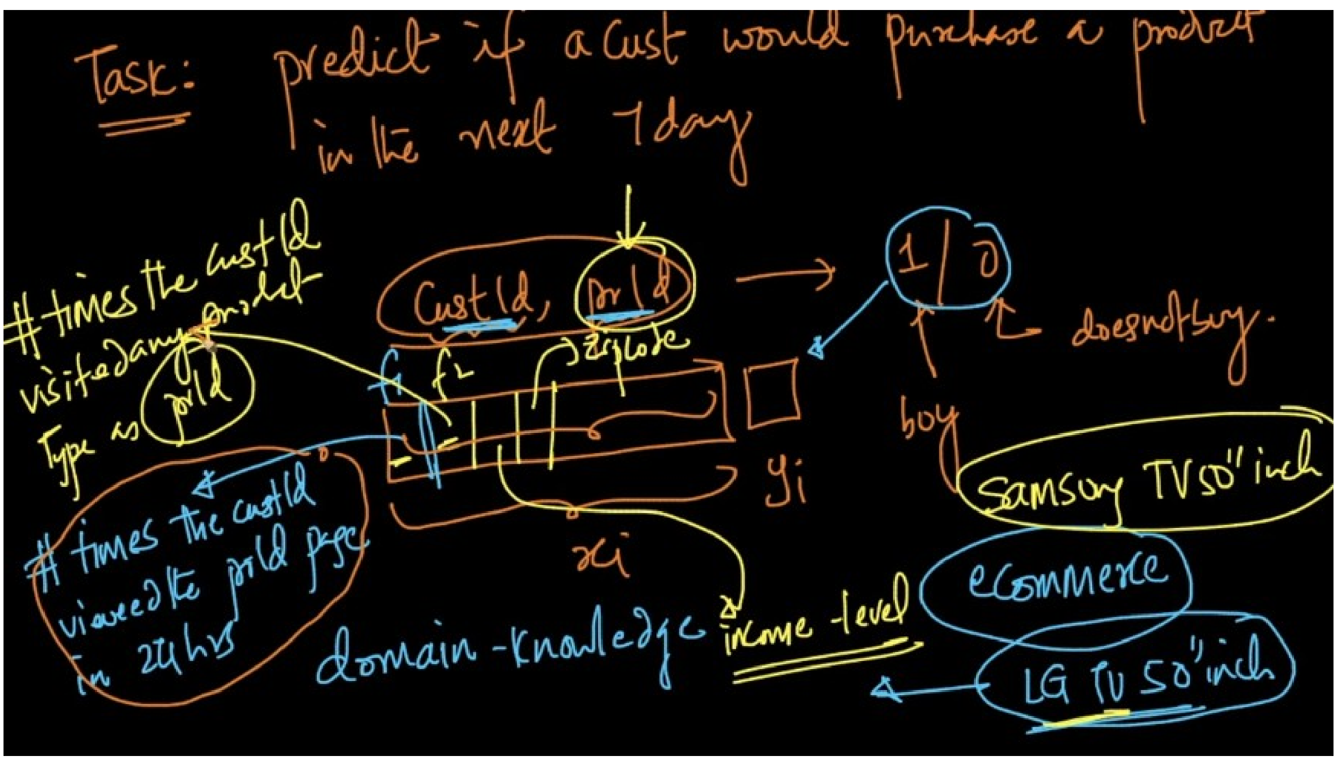
They almost automatically detect the images.



They are so powerful that we can give lots of data and make the classification.
Designing CNN’s for much faster than decades of research.
Relational data and featurization:
Rational Data: The data is stored in the from of tables in relational databases.



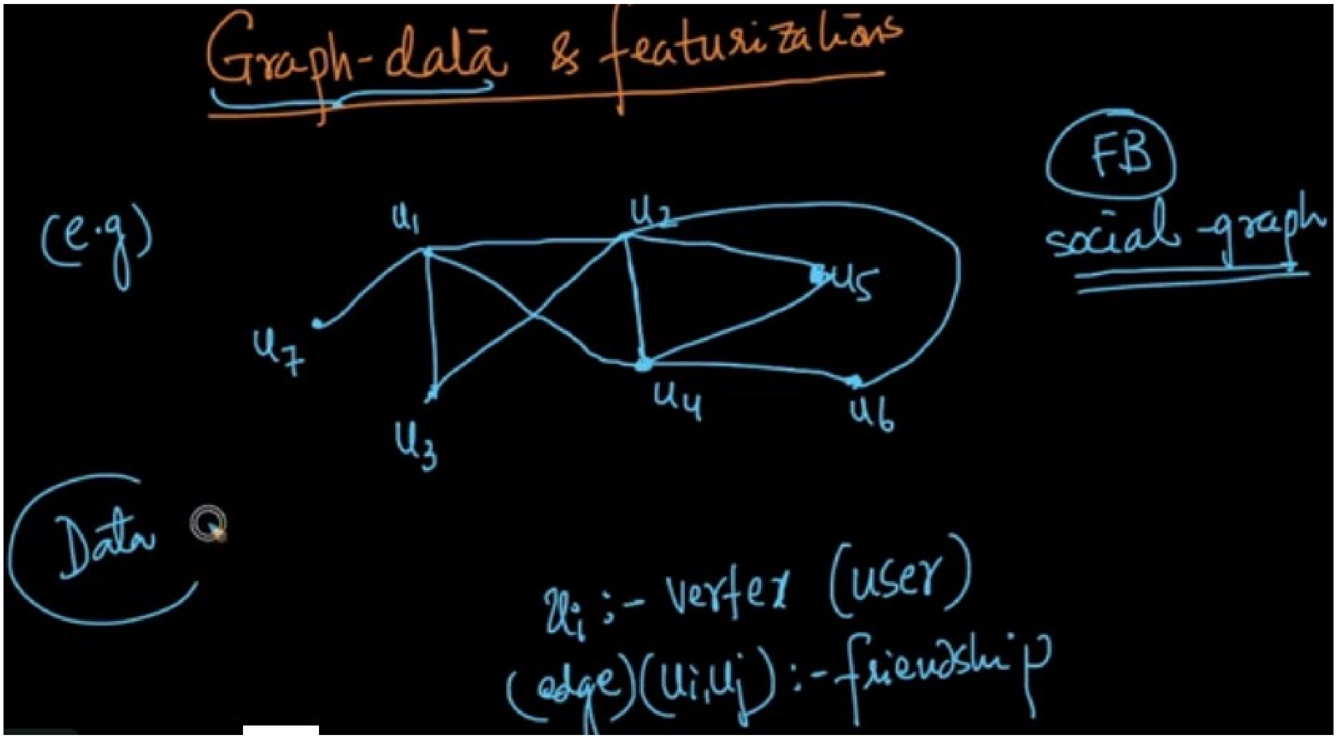
Task:



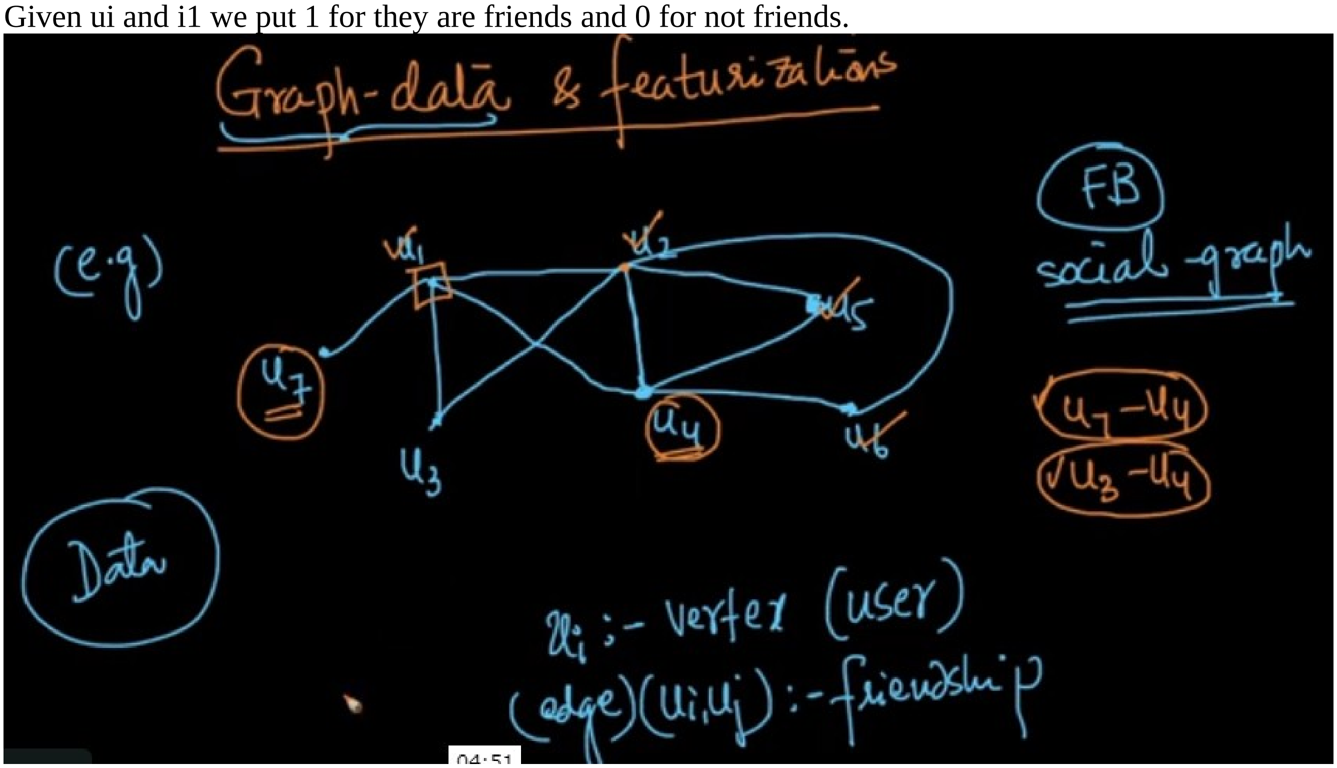
These are called the relational tables.
We use the Sql to make the grouping of the data and make the inference.

Graph data and featurization: Featurization is very domain specific.

Users are vertices and edges are relationships.

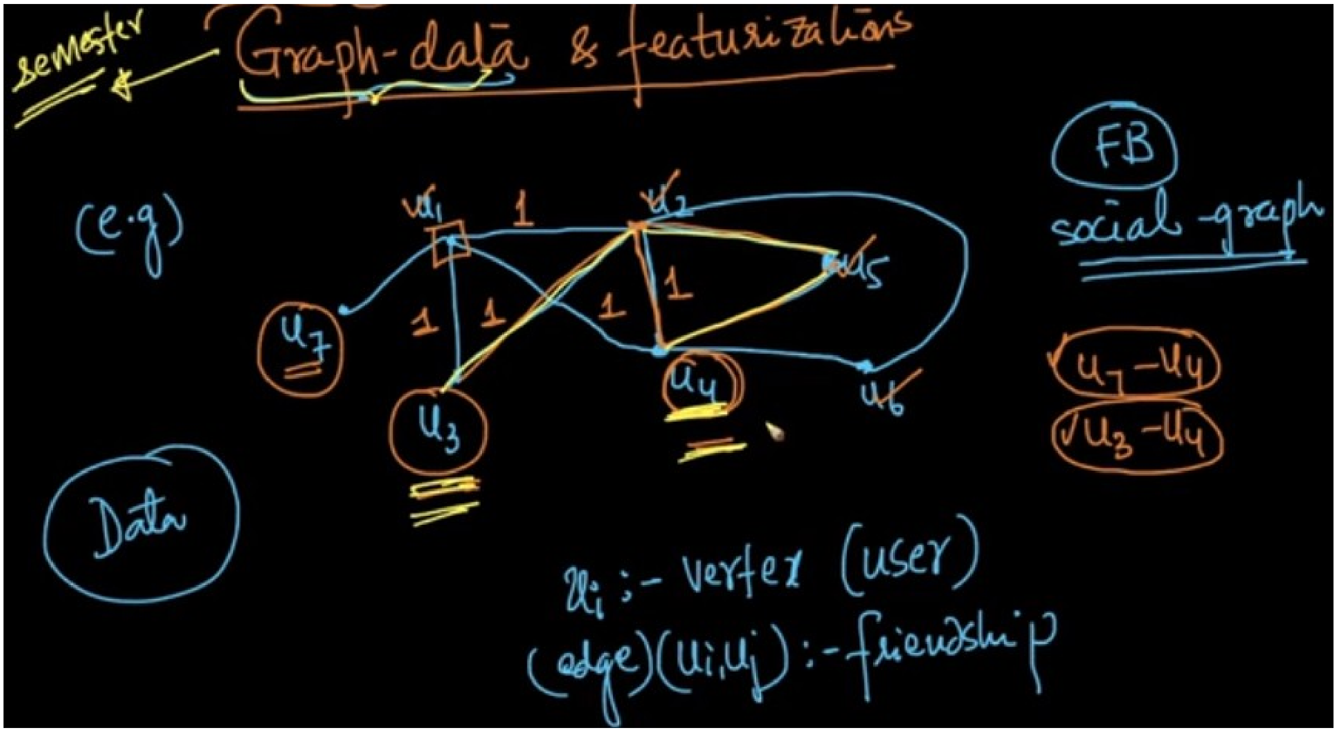


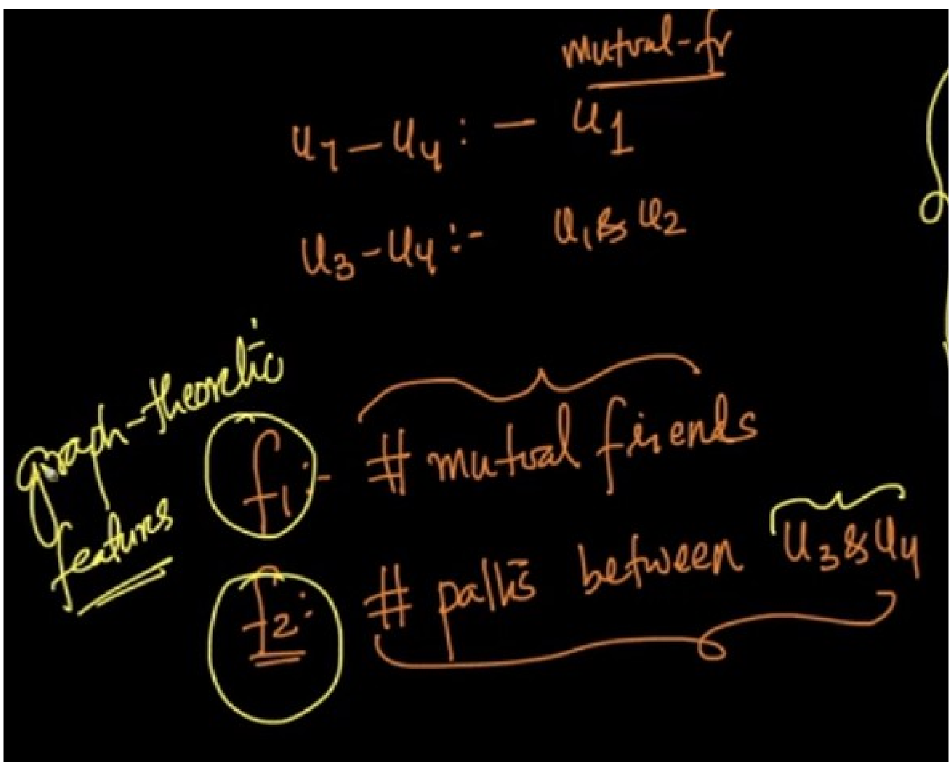
Task: Recommend new friends for a user ui.
Given ui and i1 we put 1 for they are friends and 0 for not friends.



The more the mutual friends, we have more the to be friends probability.

More the number of paths there is the higher chance of becoming the friends.



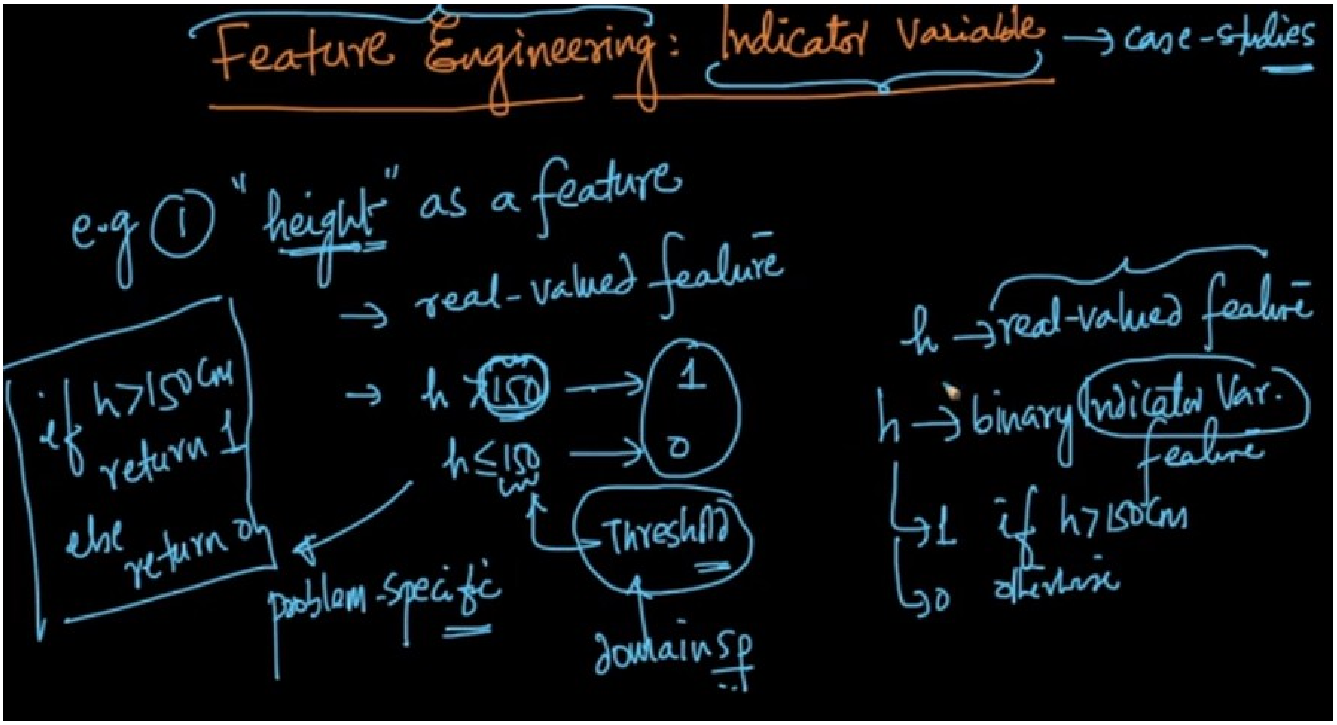


There are more spanning trees like min – spanning trees, shortest path. These features are called graph based features.

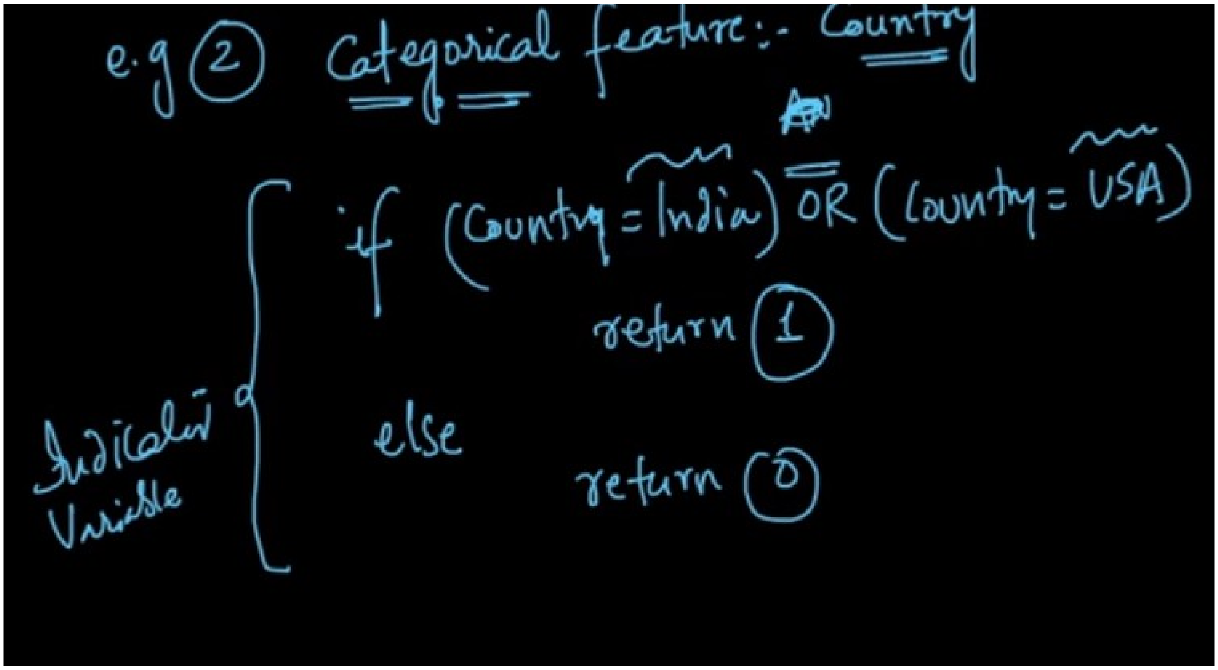
Indicator variables:

Since, height as the feature, this is the real valued feature.

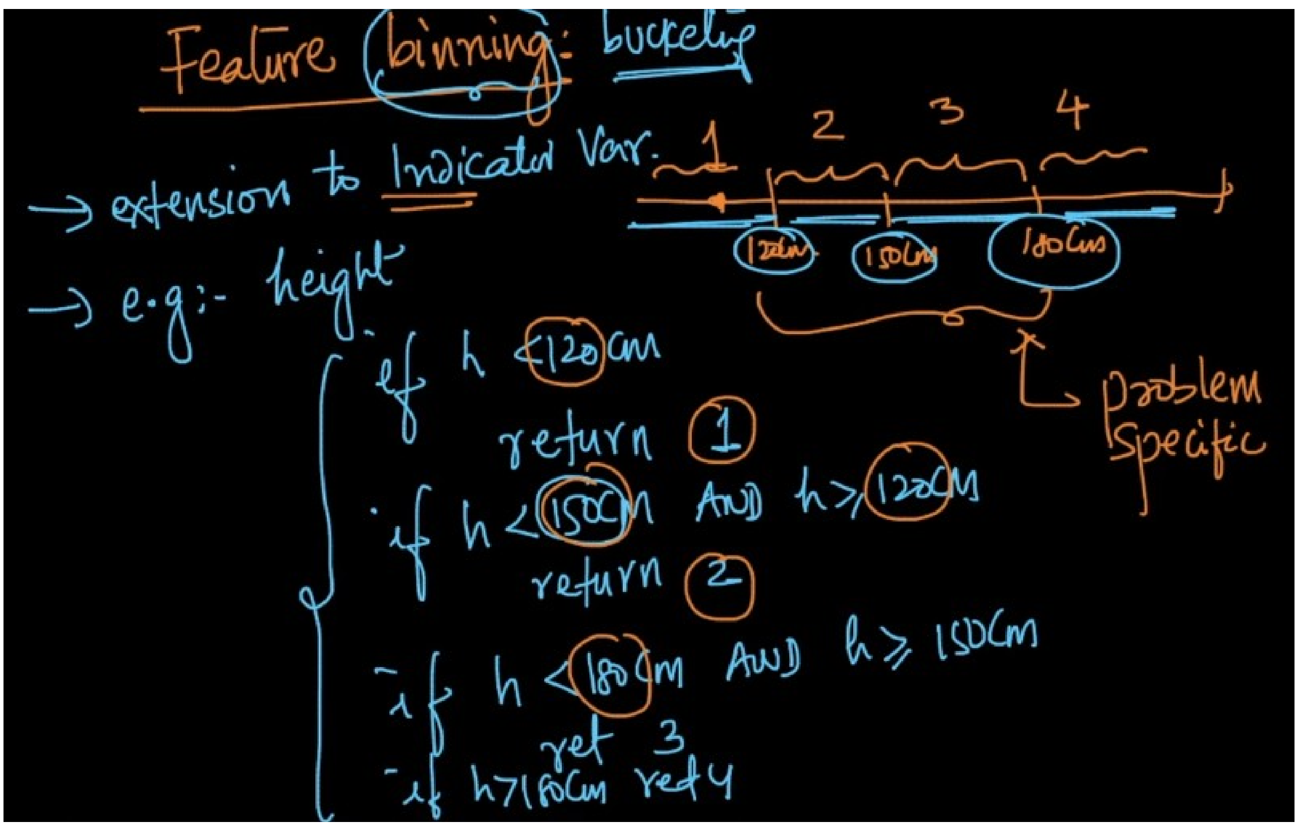
We can convert the feature into more less abstract with a threshold. This value is called the real valued feature, binary indicator variable.



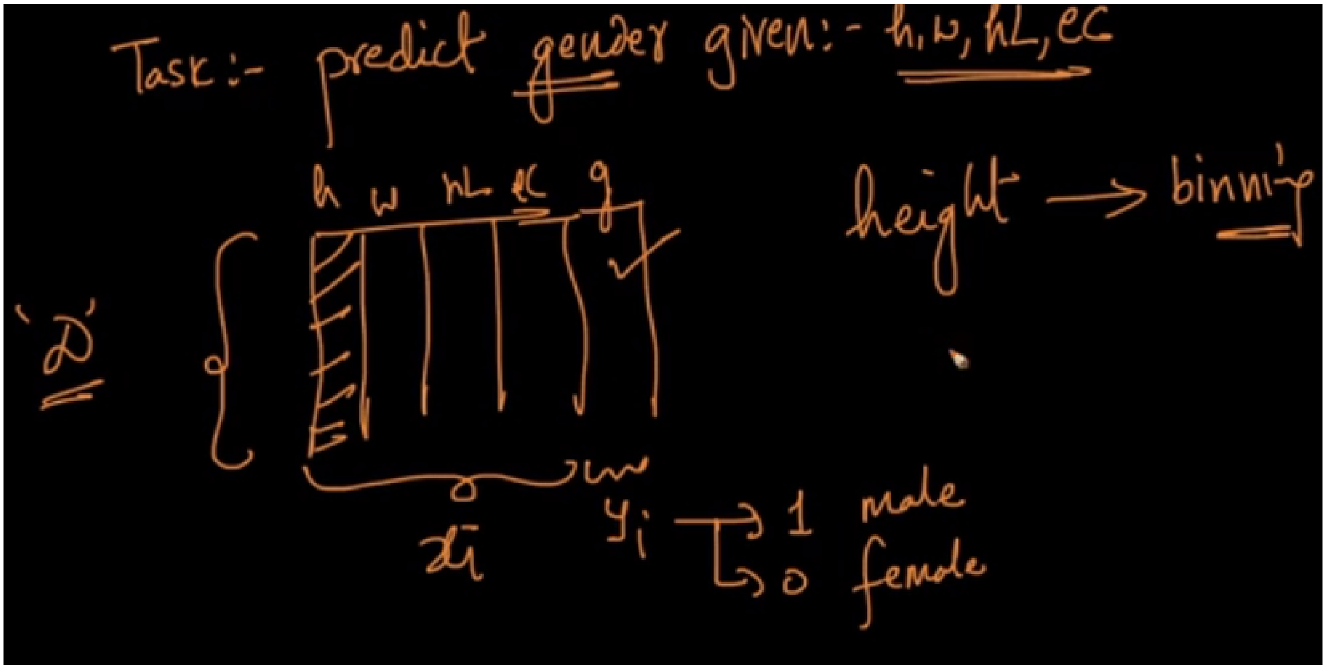
We can convert the categorical feature to a binary by making the conditions. These are called the indicator variables.



Feature binning: It is an extension to logical indicator.

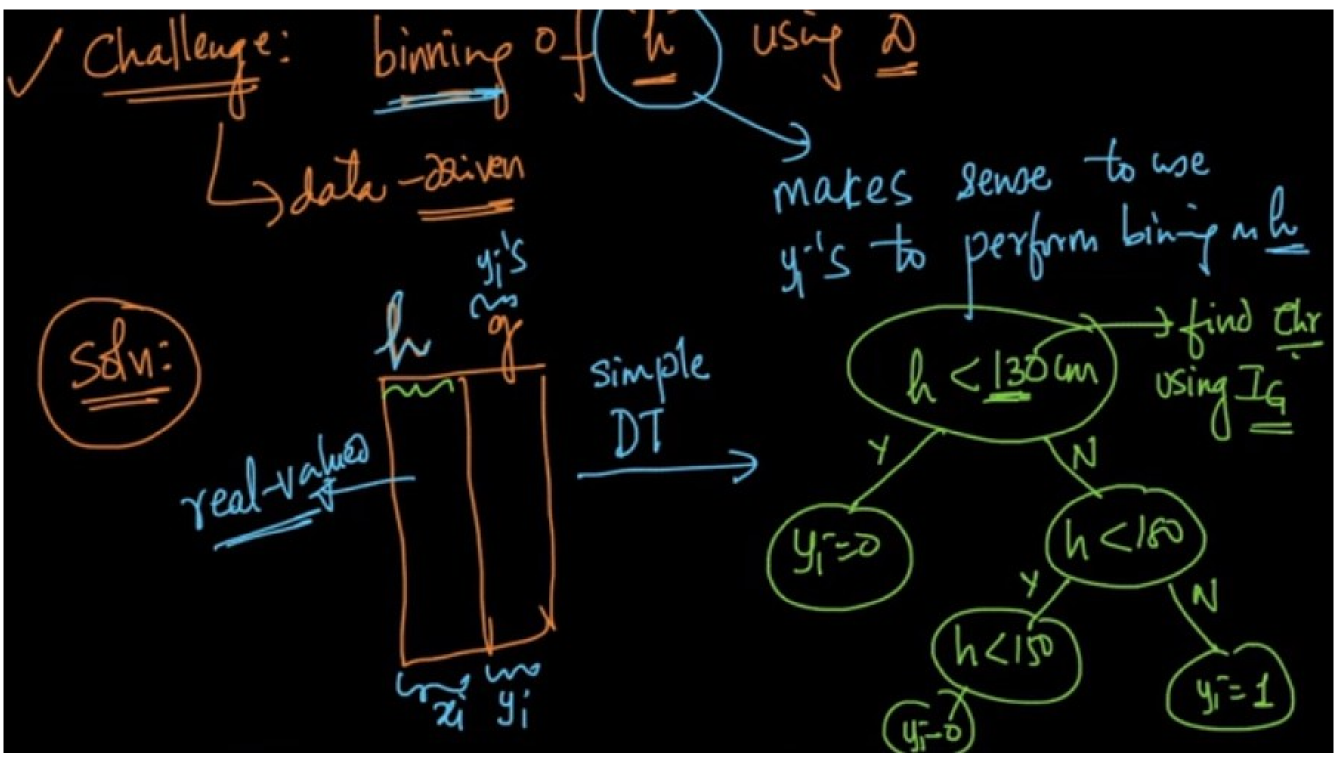


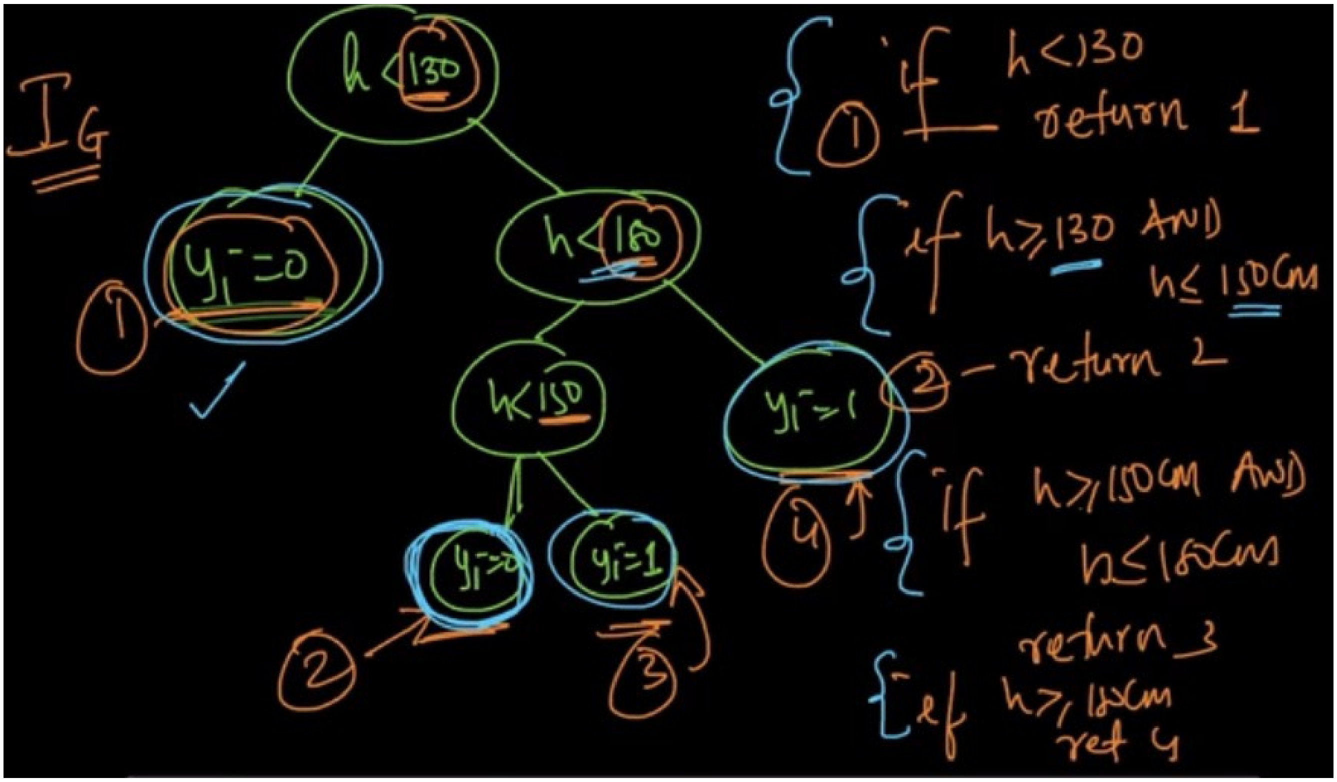
Question:



Binning the feature:

Decision trees can be helpful in making the binning, because it uses the information gain as the criteria.



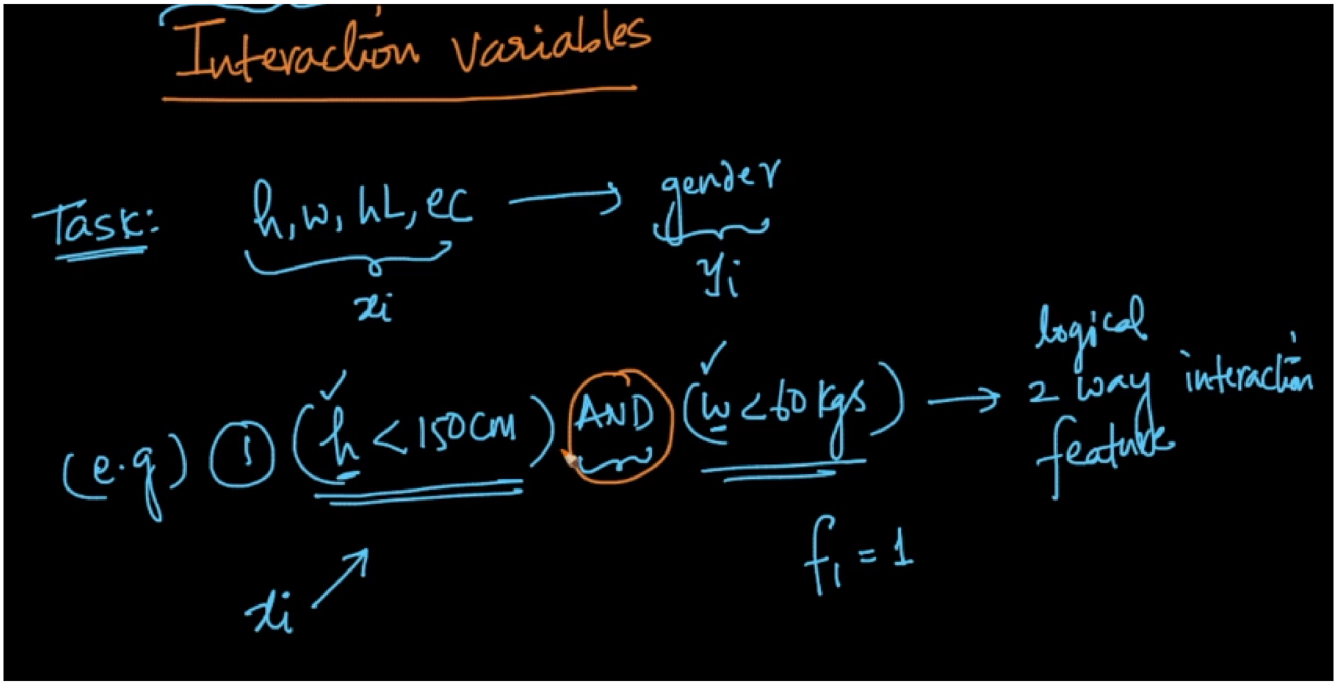


binned real-valued feature
using y_i's & feature itself using DT

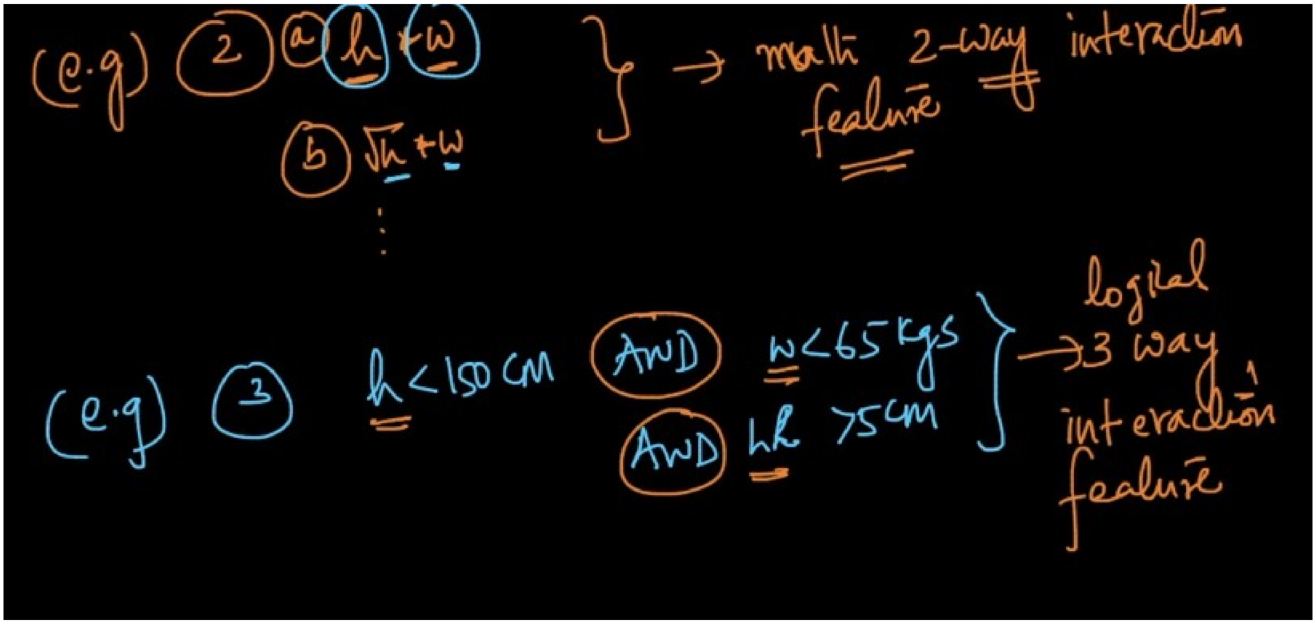
Data

Lots of Kaggle competitions → use DT based feature binning ✓

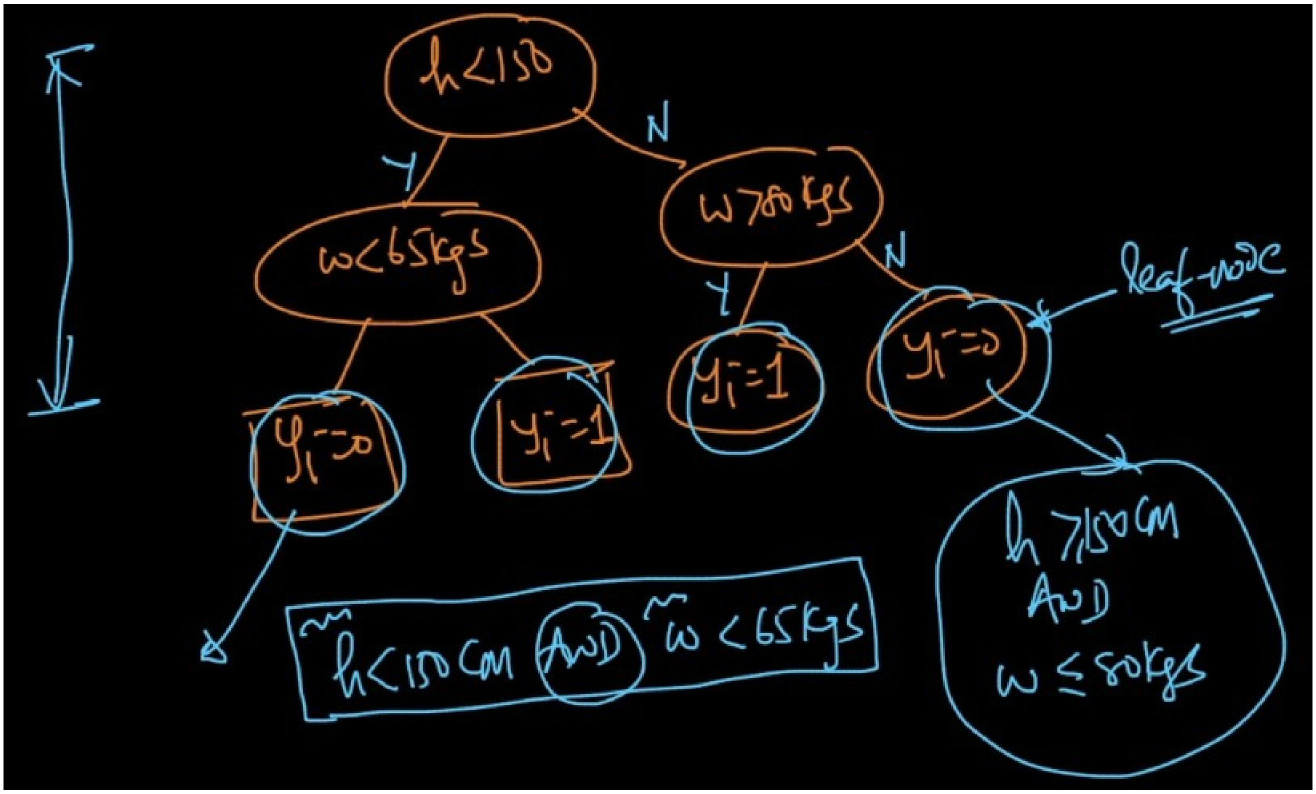
Interaction variables:
Using the two variables to make the new feature this is called the logical two way interaction.



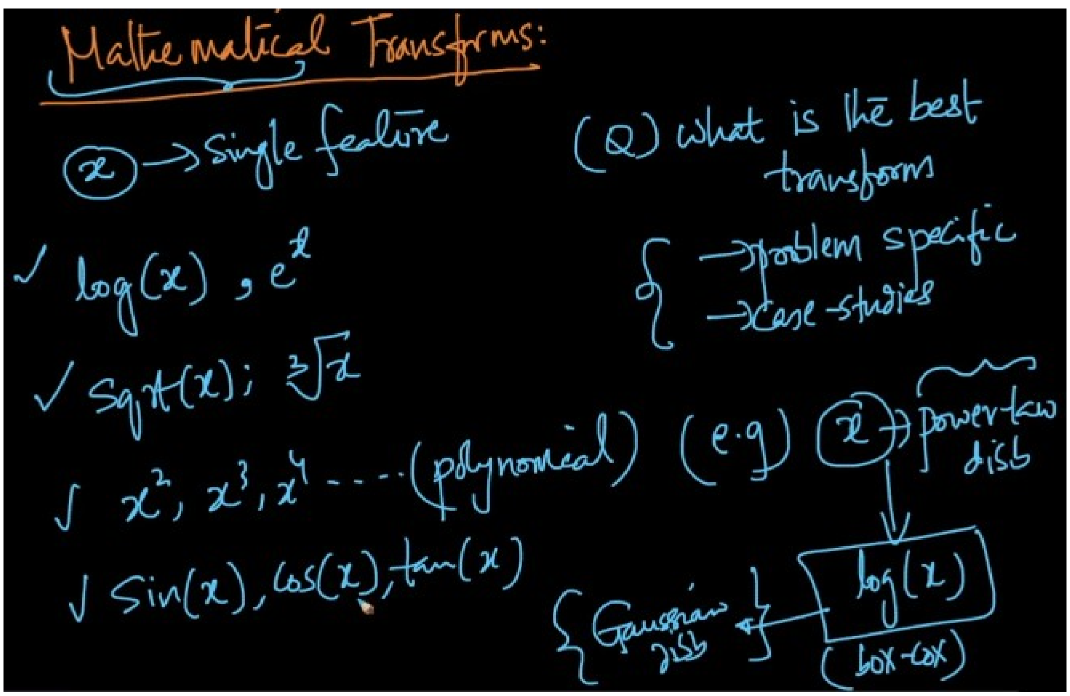
We can use the height * weight as the numerical feature. This is the mathematical 2 – way interaction feature.



How to find the good interaction features?
We can use the decision tree to fix the interaction feature.

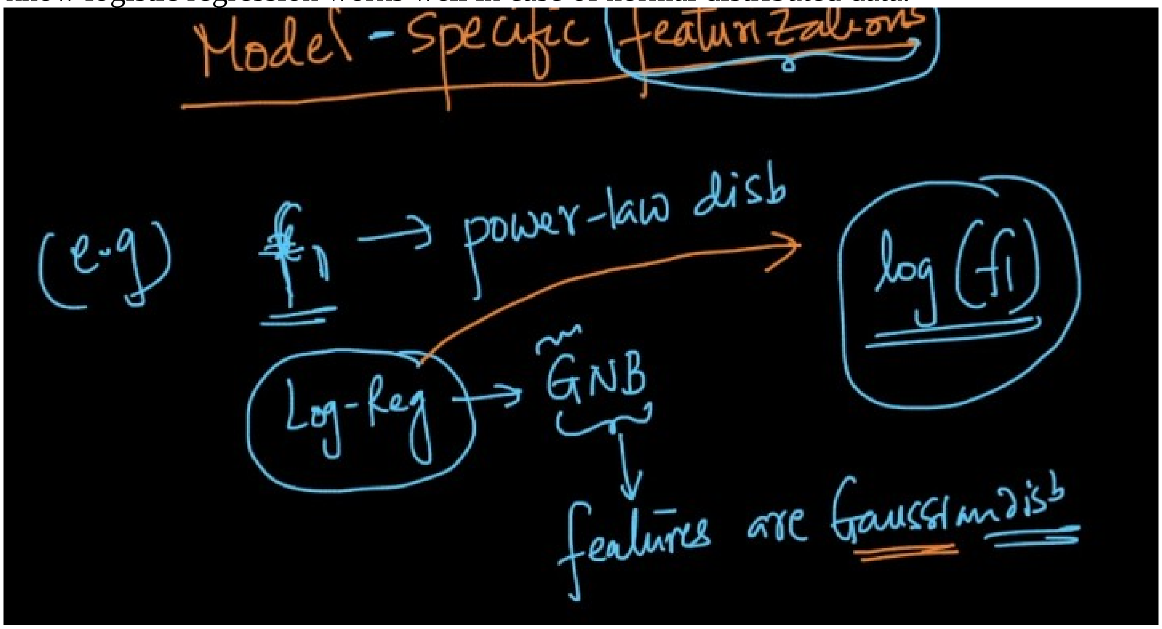


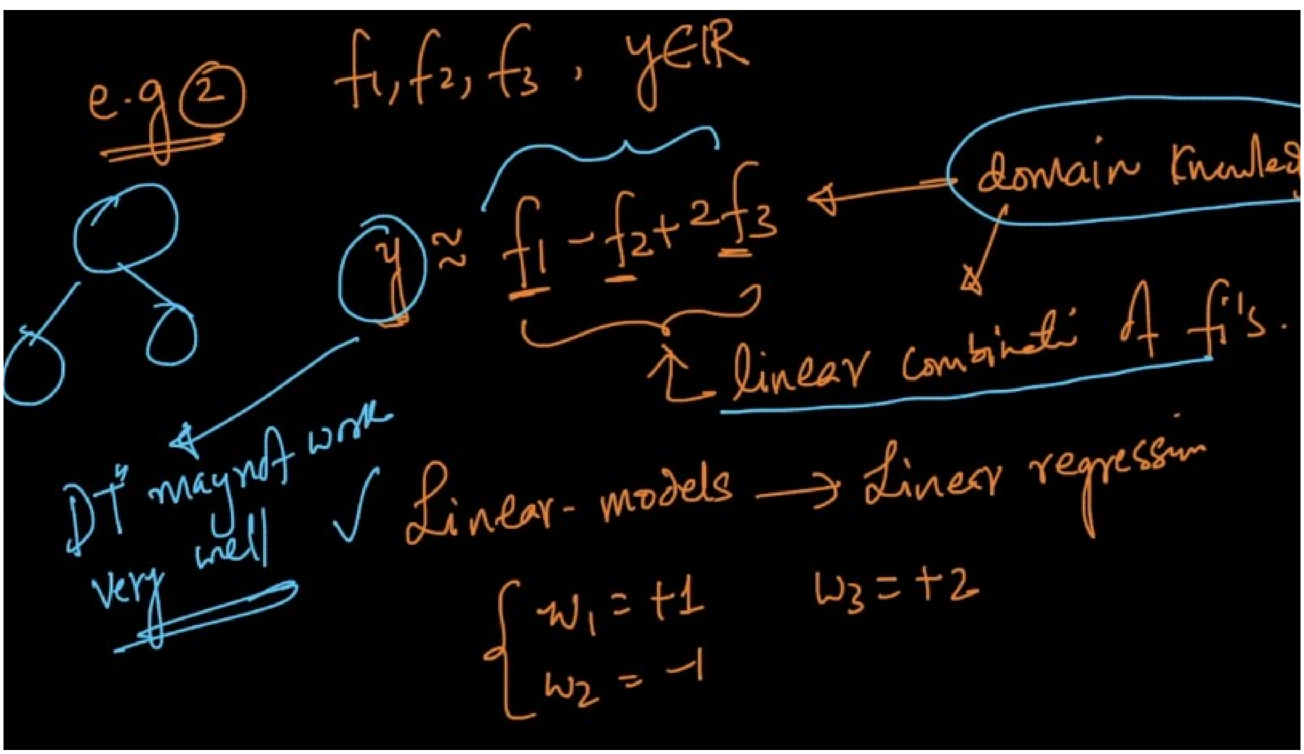
This variables are made using the information gain.
Mathematical transforms: What is the best transform?



We can apply log(x) to make the power law distributed feature to a Gaussian.
Model specific featurizations :

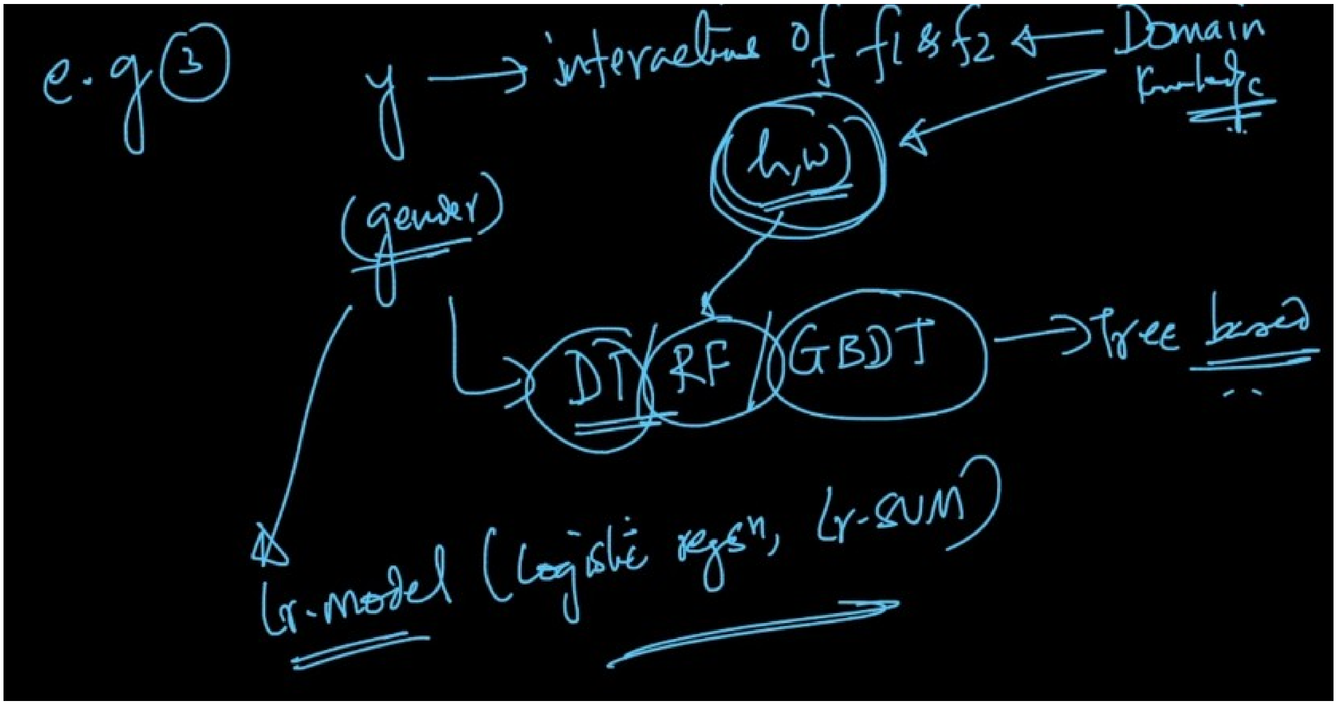
As we know logistic regression works well in case of normal distributed data.



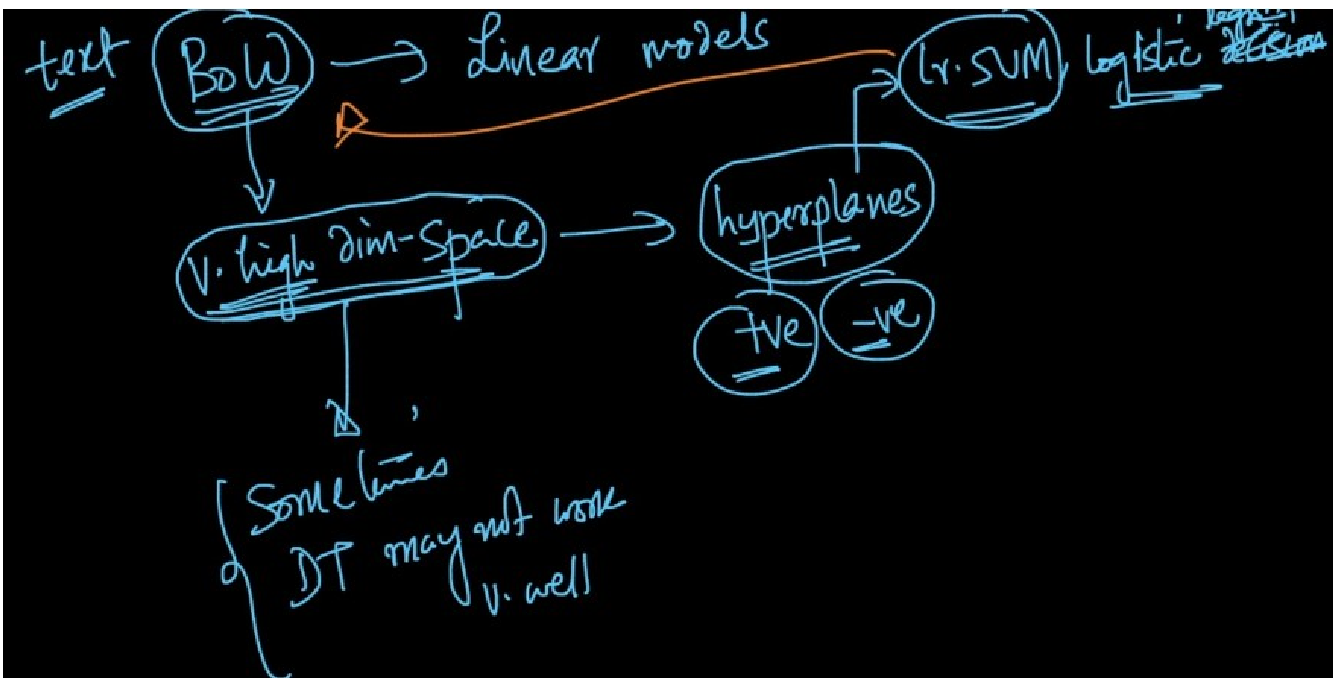


If we know y is dependent on the features f1 and f2 ← domain knowledge.

Which feature works is based on the problem specific.



If there are lots of features, example Bag of words(very high dimensional space).

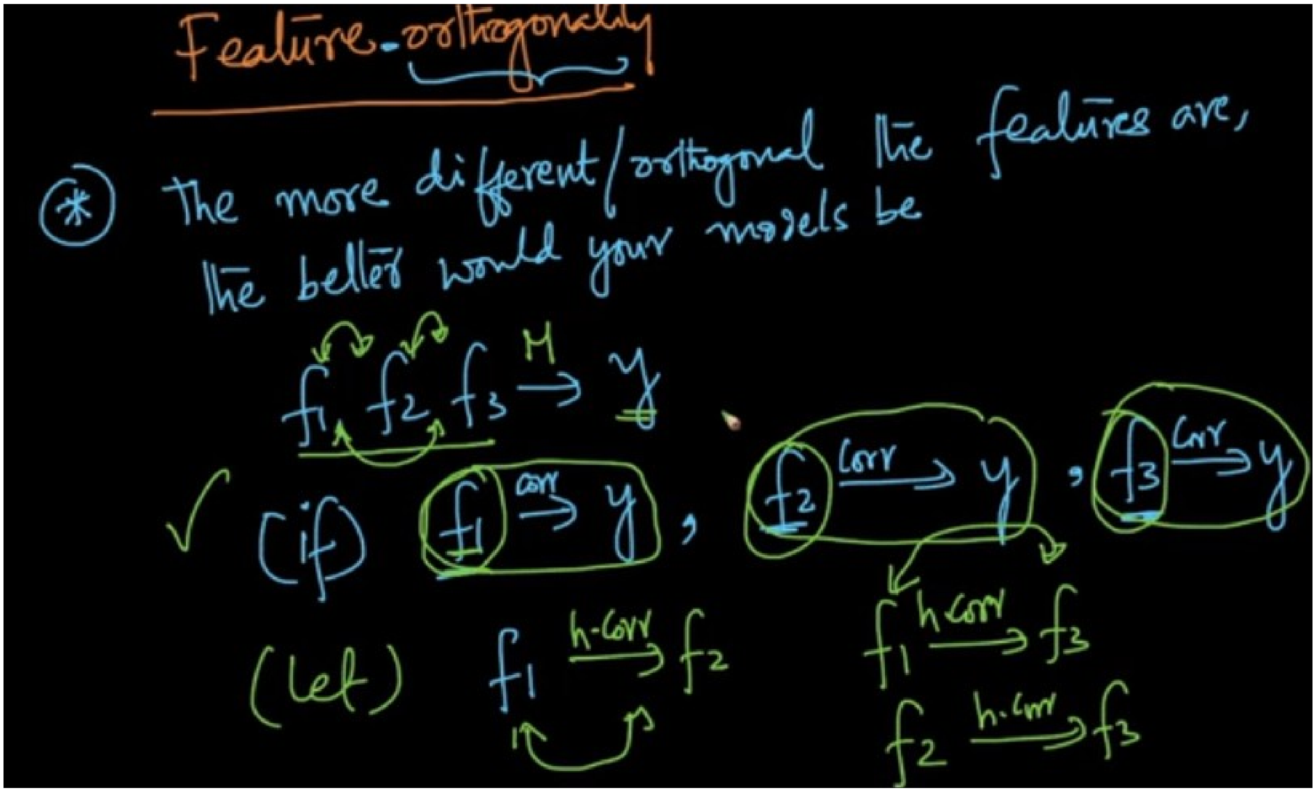


Feature orthogonality:

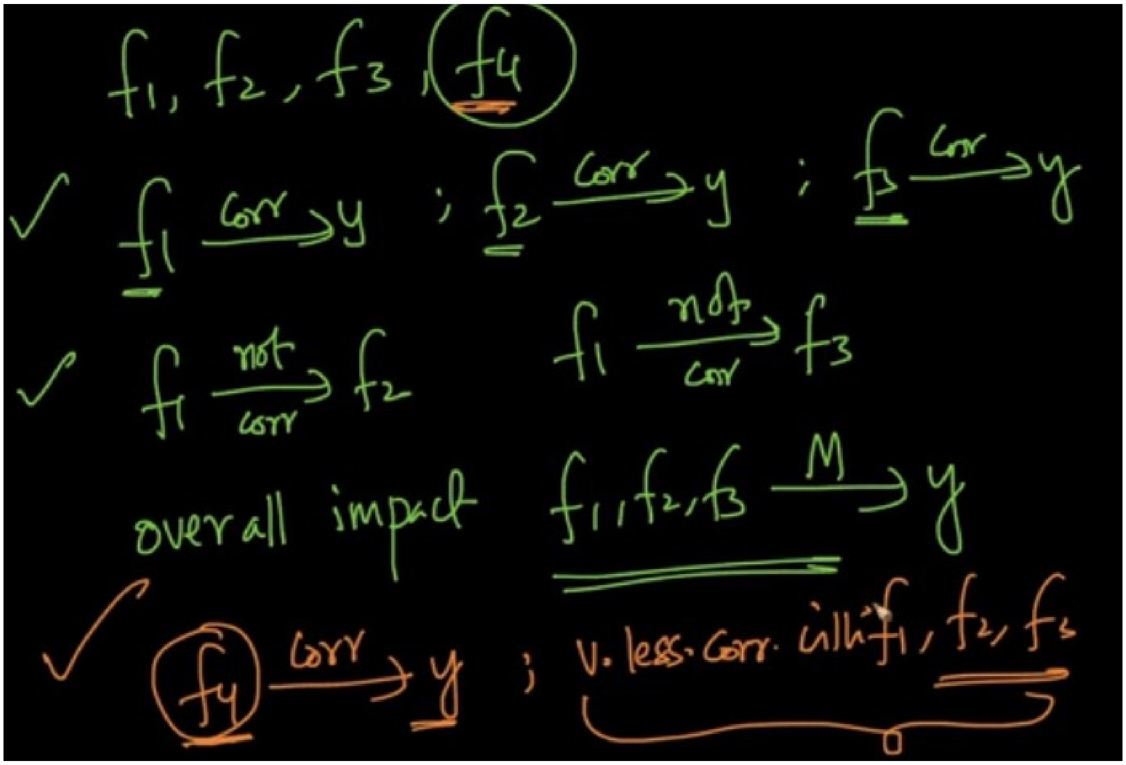
The more different/orthogonal the features are, the better would your models be

f1, f2, f3 → y

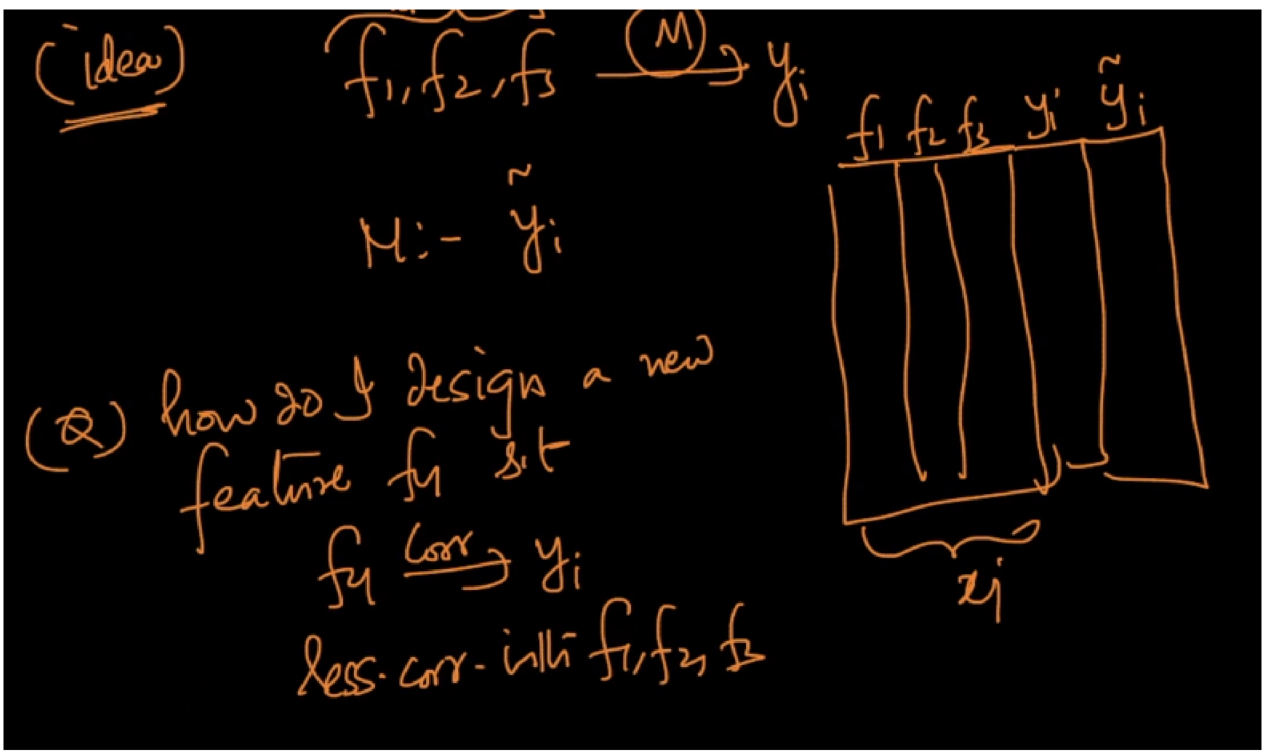
if f1 →y, f2 → y, f3 → y.

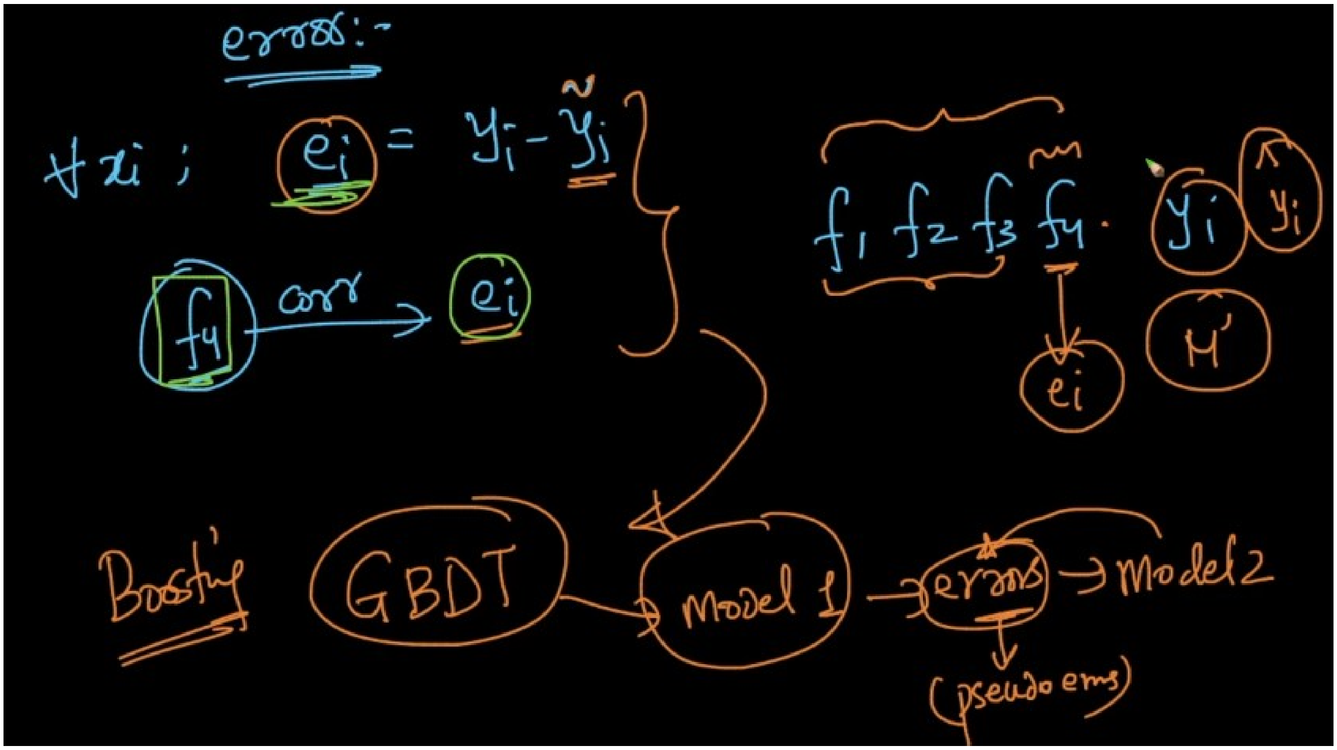


The different / orthogonal the features are the more the model will pereform.



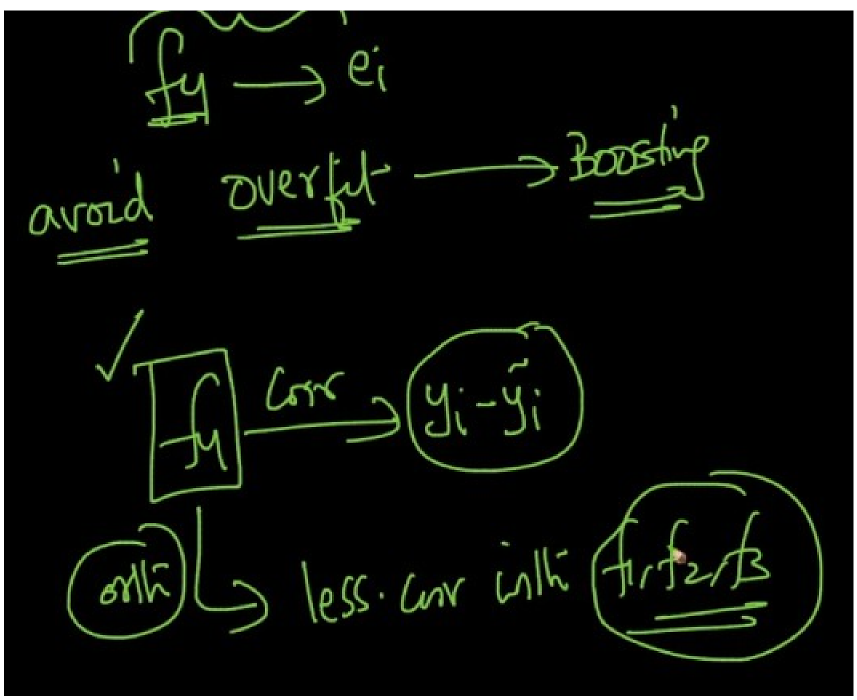
The new feature that is designed must be more correlated with the target variable than the other features.





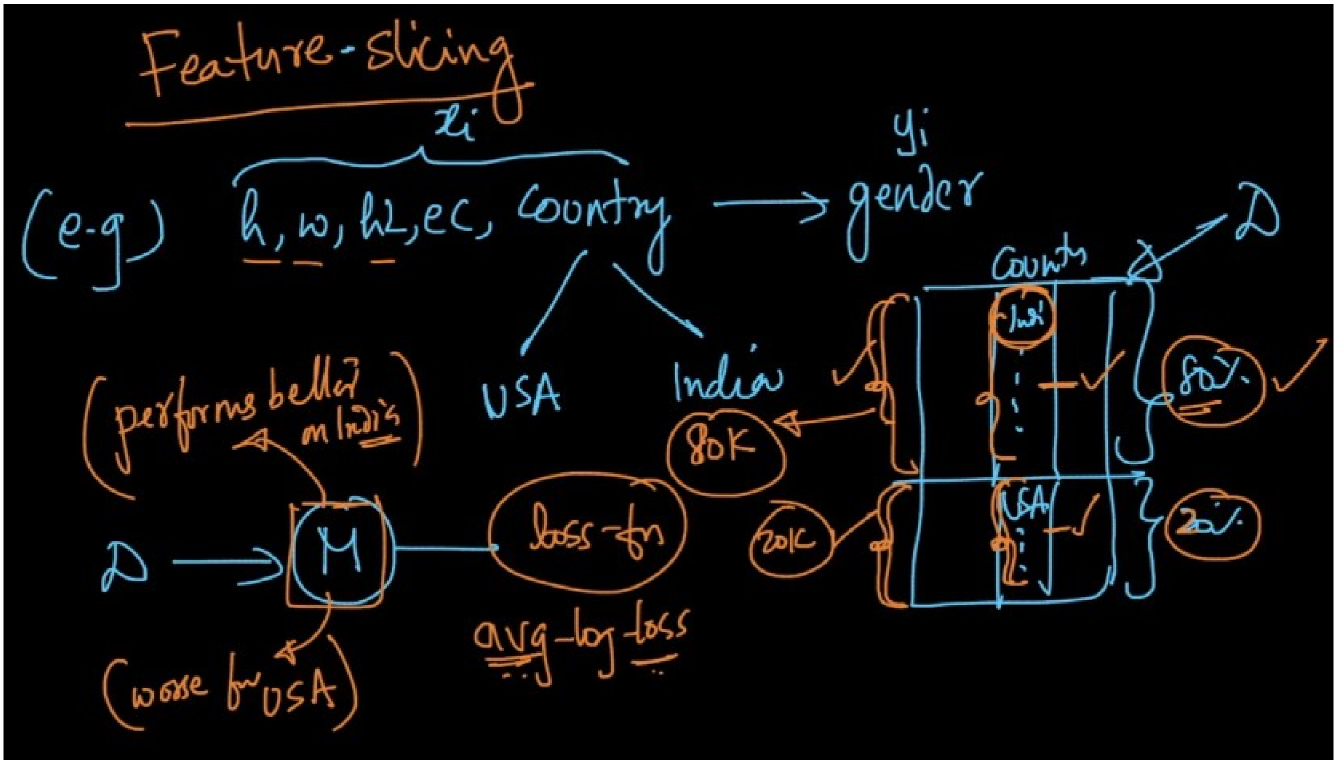
Here we use the concept of boosting to make the new feature. The new feature that is made must not over fit the data.

As the new feature is more correlated to the errors and orthogonal to the other features then the model will perform well.

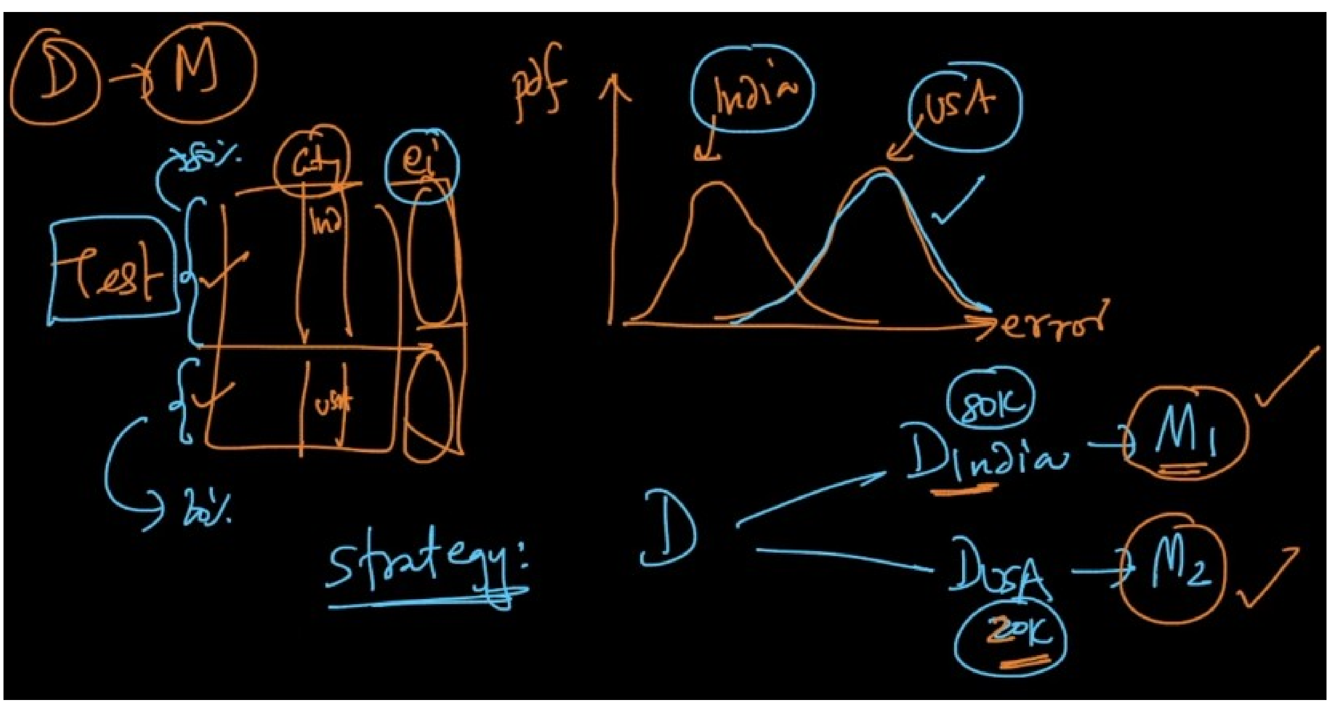


There are more chances of over fit the data.

Feature slicing:
Example:



Steps to overcome the problem:
The distributions of the errors in the two classes are more different, By this we can know the model is performing well on which class of the data.
The feature slicing is more appropriate for training the individual models.



The idea of slicing the data based on features, the two conditions must be satisfied.