Applied Machine Learning with Python

Applied Machine Learning With Python

M T W T F S S
Page No.:
Date:
YOUVA

ML - The study of computer program or algorithms that can learn by example.

* Algorithms learn rules from labelled examples

* **Training data** - The labelled examples used for training of algorithms are Training dataset.

* **Test data** - The dataset used to evaluate the Algorithm wheather it is working correctly to unknown dataset which is not in Training.

Supervised learning - Algorithms that uses labelled datasets to predict targets values of unknown datasets.

i) **Regression** - Target values are in continous forms. their is relationship between input given and target value and that relationship is continuous

ii) **Classification** - Target values are discrete values or discrete classes. And target value is need to be identified as wheather it is belonging to perticular class or not.

iii) **Disision Tree** - Target values are determined by set of (if else) questions

Unsupervised - Algorithm do not use lebelled data and finds structure in unlabelled data directly to make predictions.

i) clustering - Find group of similar instance in the data.

ii) Outlier detection - Finding unusual patterns

Basic Machine learning workflow.

Representation → Evaluation → optimization

Choose:
i) A features Representation
ii) Type of classifier Algorithm to use

e.g. image pixel with K-nearest neighbour classifier

Choose:-
i) what criteria used to distinguish good v/s bad classifier

e.g. % of correct predictions on test set

Choose:
How to sear for settings or paramet that gives best classifi for this evalu criteria.

e.g. Try a ran of values of 'K' in K-near neighbor class

Examining the data

1) certain preprocessing is needed for dataset.
e.g. to see whether some cells in dataset are missing or not. which affects model badly.

2) sometimes examining dataset can make you realize that no machine learning algorithm required for problem solving.

3) for Inspecting the dataset we use visualisation methods like feature plots.

Scikit learn -

i) scipy - i) with scikit learn it provides support for sparse matrices a way to store large tables that consists mostly of zeros.
ii) provide useful scientific tools like -
a) statistical distributions,
b) optimization of functions.
c) mathematical functions.

ii) Numpy - i) provides fundamental data structures used by scikit learn like multidemensional array.
ii) Typically data input to scikit is Numpy array.

iii) Pandas - i) provides key data structures like dataframes.
ii) Also support Reading/writing data in dataset.

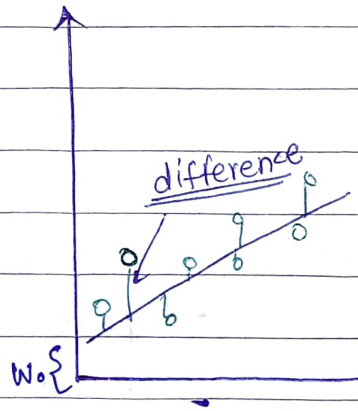
iv) Matplotlib - plotting libraries to show spread of data.

Regression

Takes set of training instances and learn a mapping to target value.

Target value is continous (floating point)
Real valued target values are required regressio model.

Least square linear regression



W_0 and W_1 are parameters defined by many methods correspond to different fit criteria and goals and ways to controls model complexity.

$$RSS = \sum_{i=1}^{n} [y_i - (w_0 + w_1 x_i)^2]$$

The learning algorithm finds the parameters that optimize an objective function that minimize the error between predicted target value and actual target value

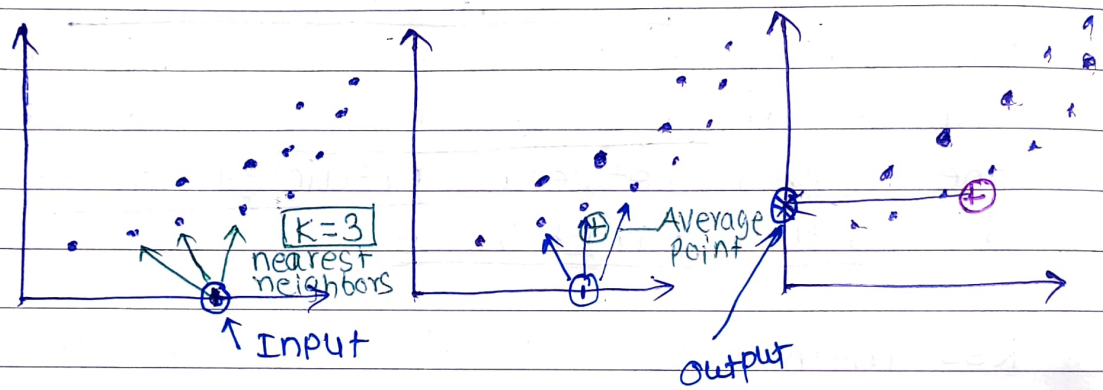
RSS method used to minimize the difference of predicted and actual values.

For least square Linear Regression line is always straight.

* The value of W_0 & W_1 ... W_n are finded by least square method i.e. the line which gives lesser RSS is choosen.

KNN & Regression

1) KNN Regression models finds the nearest neigh to input and no. of neighbor based on value of [K] then
2) Take average of all this points (neighbors) and according to regression model it defines target value for given input.



R^2 - (r-squared) Regression Score - Measure of determining how well Regression model is Predicting the data.

Score (0 to 1)

0 - corresponds to constant model which gives mean value of all training data given.

1 - corresponds to perfect predictions.

Also known as coefficient of determination.

Key Parameters

Model complexity

1) n_neighbors - number of nearest neighbors [Default = 5]

Model fitting

2) metric - distance function. [Default ⇒ p=2] Euclid minow

Ridge Regression (L2 Regularization)

Ridge regression learns W_1, W_0 using same least square criterion but adds a penalty for large variations in W parameters

$$(RSS)_{Ridge}(W, W_0) = \sum_{i=1}^{n} [y_i - (W_1 x + W_0)]^2 + \alpha \sum_{j=1}^{n} W_j^2$$

once the parameters are learned, the ridge-regression prediction formula is the same as ordinary least squares.

Regularisation - method that prevents overfitting by restricting the model, typically to reduce complexity by addition parameters.

Regularisation is controlled by $\boxed{\alpha}$ term higher the $\boxed{\alpha}$ higher is the regularisation and less complex model and, less likely to be overfitted.

Lasso regression (L1 Regularization)

minimizate the sum of the absolute values of the coefficients

$$RSS_{Lasso}(W, W_0) = \sum_{i=1}^{n} [y_i - (Wx_i + W_0)]^2 + \alpha \sum_{j=1}^{n} |W_j|$$

This has the effect of setting parameters weights in W to zero for the least influential variables. This is called sparse solution kind of feature selection.

The parameter α controlls regulisation
default value of $\alpha = 1.0$.

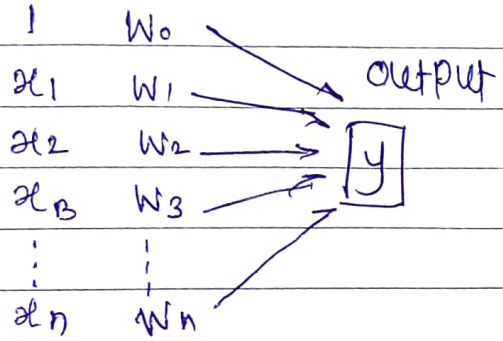
When to use Lasso Vs Ridge

Many small/medium sized effects : use ridge
Few medium/large sized effects : use Lasso

Regularization becomes less important as amount of training data increases.

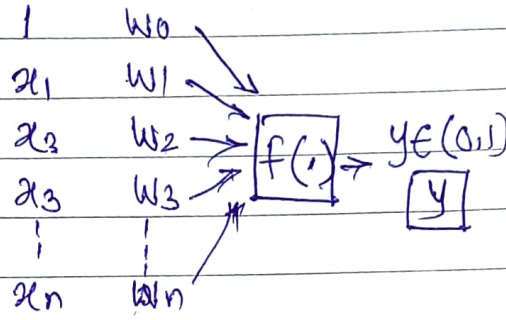
Logistic Regression

Linear regression



$$\hat{y} = w_0 + w_1x_1 + w_2x_2 \cdots w_nx_n$$

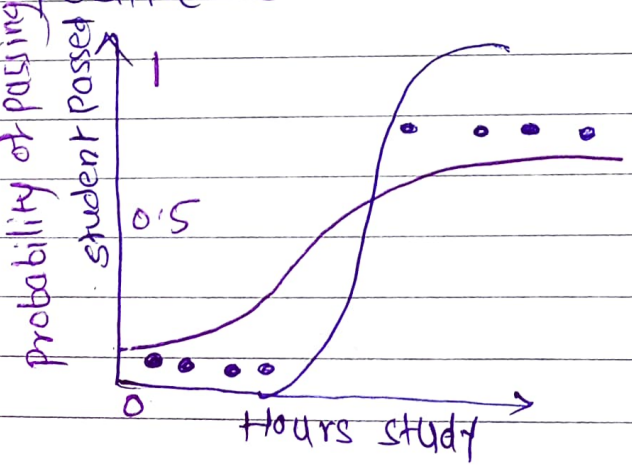
Logistic Regression



$$\hat{y} = \text{logistic}(w_0 + w_1x_1 \cdots w_n$$

$$\hat{y} = \frac{1}{1 + \exp[-(w_0 + w_1x_1 \cdots w_n$$

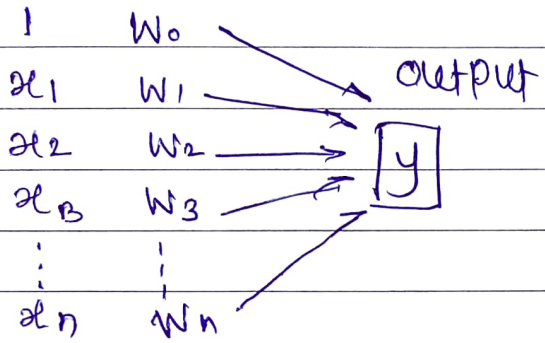
Logistic regression gives output values that falls between (0 to 1) that signifies output or target value falls in positive or negative class. It is the probability the input objects belong to positive class.



Regularization becomes less important as amount of training data increases.

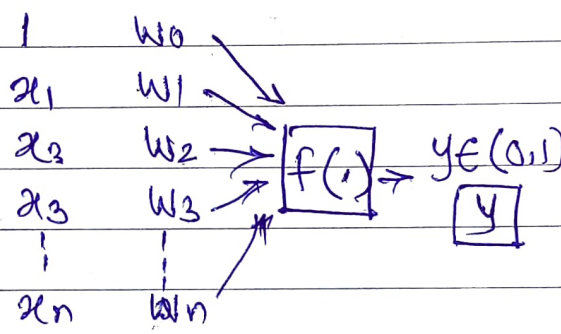
Logistic Regression

Linear regression



$$\hat{y} = w_0 + w_1 x_1 + w_2 x_2 \cdots w_n x_n$$

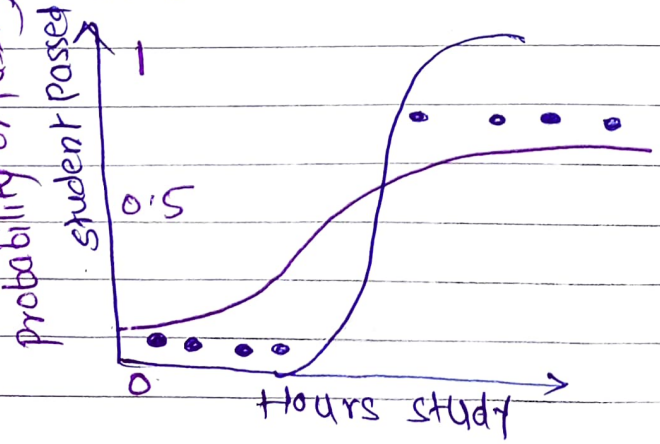
Logistic Regression



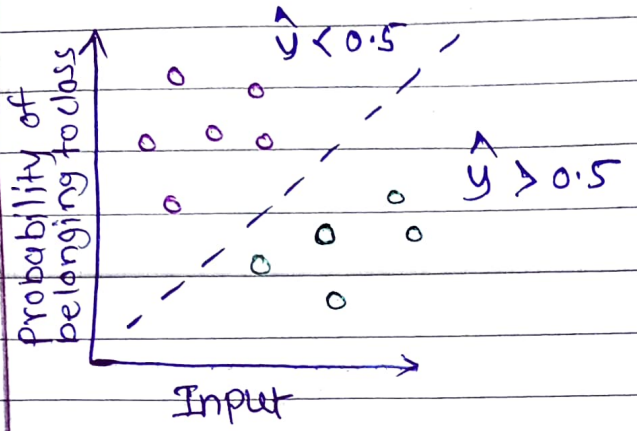
$$\hat{y} = \text{logistic}(w_0 + w_1 x_1 \cdots w_n$$

$$\hat{y} = \frac{1}{1 + \exp[-(w_0 + w_1 x_1 \cdots w_n$$

Logistic regression gives output values that falls between (0 to 1) that signifies output of target value falls in positive or negative class It is the probability the input objects belong to positive class.

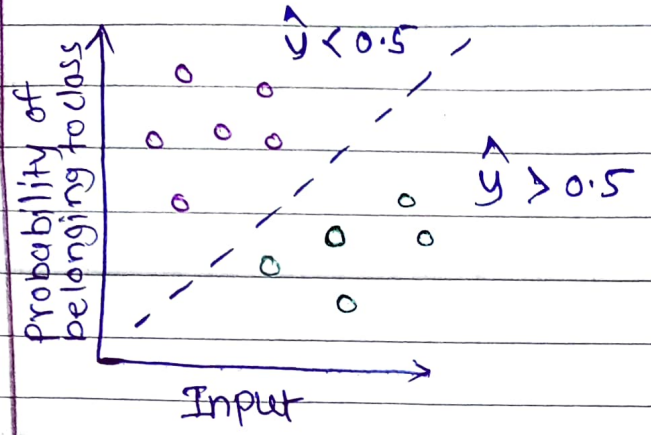


Logistic regression is used for Binary classifi



Logistic regretion is used for Binary classification.

Logistic regression is used for Binary classi



logistic regretion is used for Binary classification.

Classification

Takes set of training instances and learn a mapping to a target value.

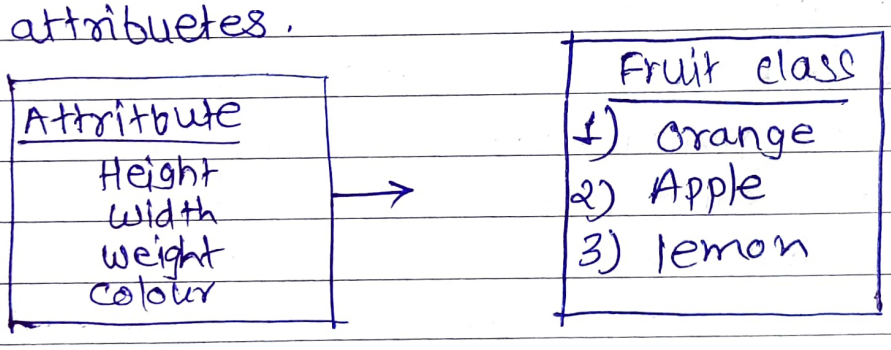
for classification target value is discreate class value.

1) Binary

target value is 0 (negative) or 1 (positive)
e.g. fraudelent credit card transaction

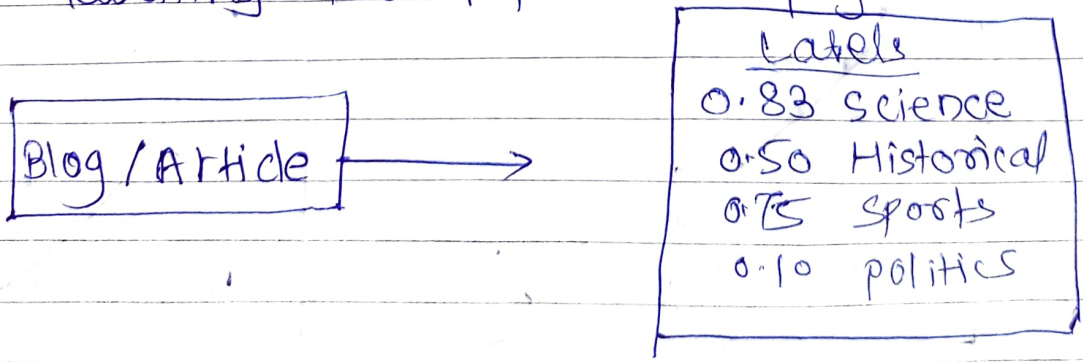
2) Multiclass

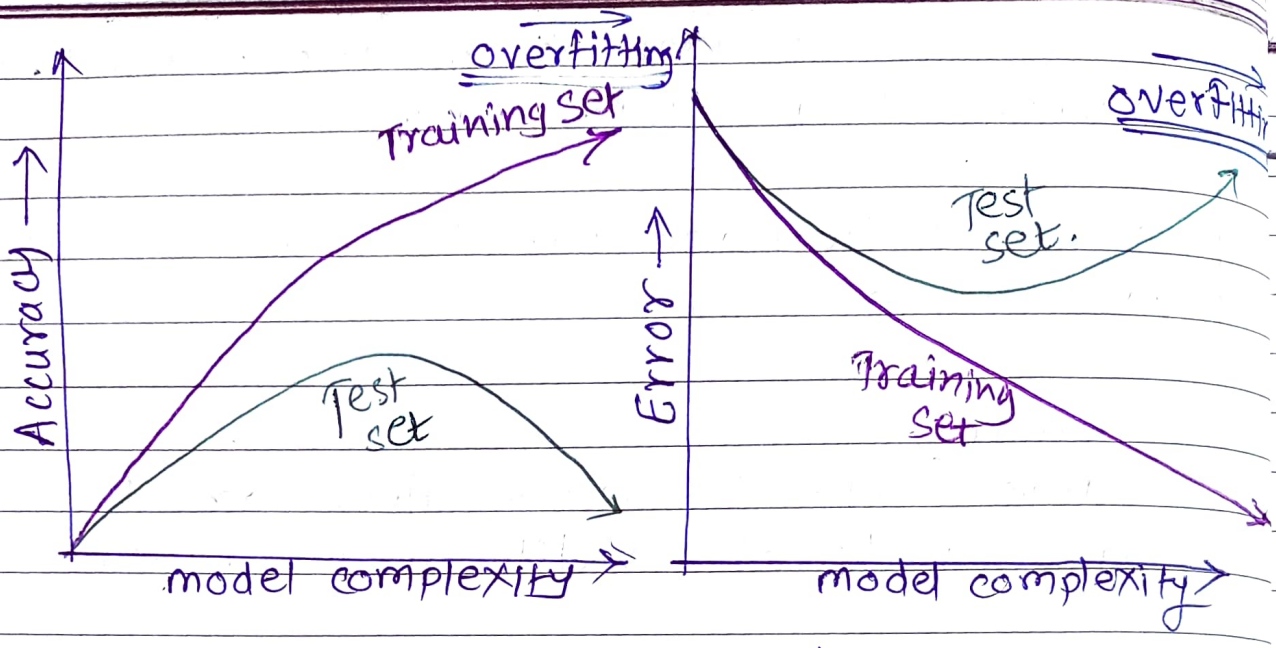
target value is one out of a set of descrete value
e.g. lebelling the Name of fruit from given attributes.



3) muultilabel classification

target values are multiple.
e.g. labelling the topic on webpage.





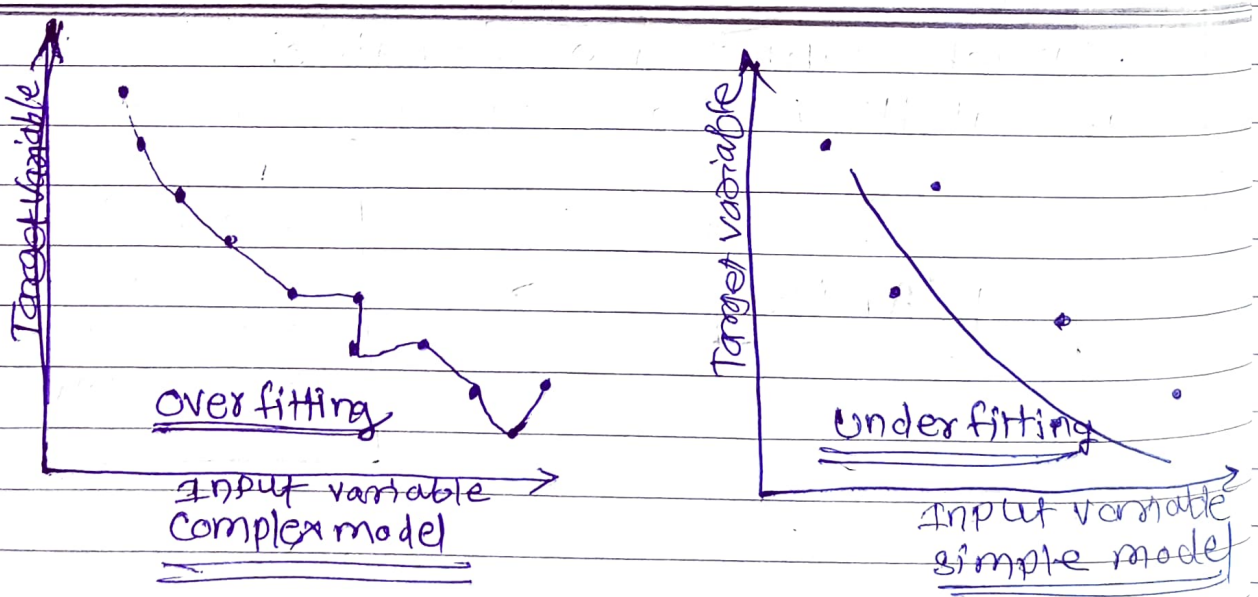
Accuracy and Errors are need to be optimal If model overfits Accuracy of test set decr and Error increases.

overfitting

Models that are to specific for training datas are not giving good results for test datas they are said to be overfitted.

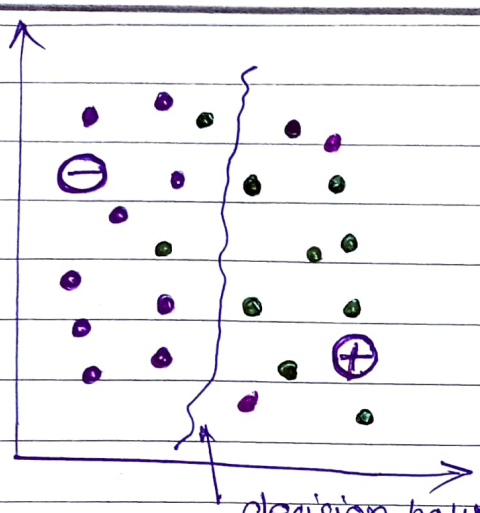
Underfitting

Models that are too simple, not fitting train data well are underfitted.

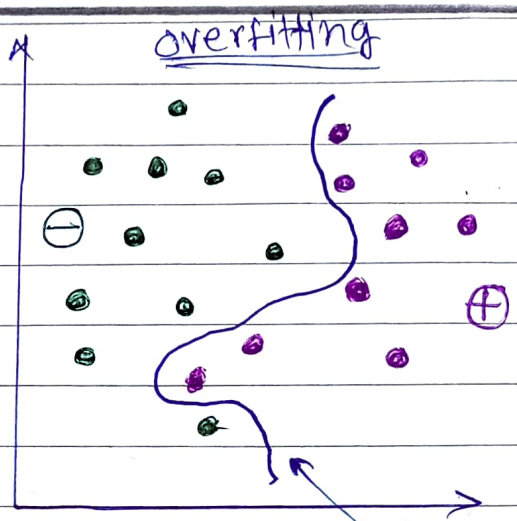


Regression Underfitting and overfitting.

Classification overfitting and underfitting



decision boundary is not clear.
It allows some points falls at across hence not overfitted and good model.



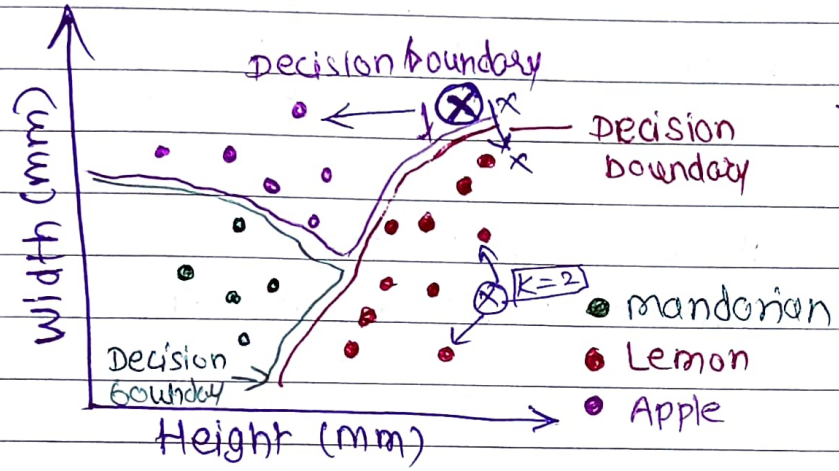
decision boundary Too overfitted

Decision boundary defines complexity of classifier model. It should not be too distinct to cover every point. And able to ignore minor variations.

K-nearest neighbor classifier.

M T W T F S S
Page No.:
Date:
YOUVA

K-NN - It is also called as instance based or memory based classifier. because it memorize the instances given to it and make further predictions based on that memory.



Fruit dataset

How to predict

i) when new instance is occured we find the the nearest point to than instance (query) and lebal of that point is assigned and predic

ii) For $K > 1$ the two or more according to K no. of points nearest to query point are finded.

iii) Point nearest to same region is mapped and not from different region/class. this classes/ regions are distinguished with decision boundaries.

Needs four things to be specified.

i) A distance metrics
Typically Euclidean metric used by detaur in scikit with $p=2$ minkowski metric used.

ii) How many nearest neighbors. $k=?$

iii) Influence of most nearest neighbors.
optimal weighting functions on the neighbou points.

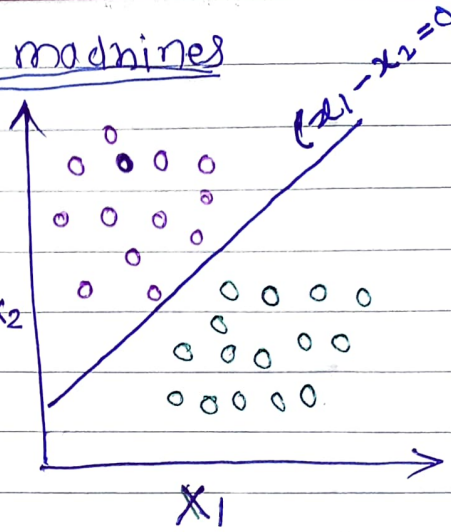
iv) How to aggregate the classes of neighbour p
simple majority vote: class with most representatives with nearest neigh

Linear classifiers : support vector machines

Feature vector → [f] → ŷ → (+1) / (−1)

Input

output

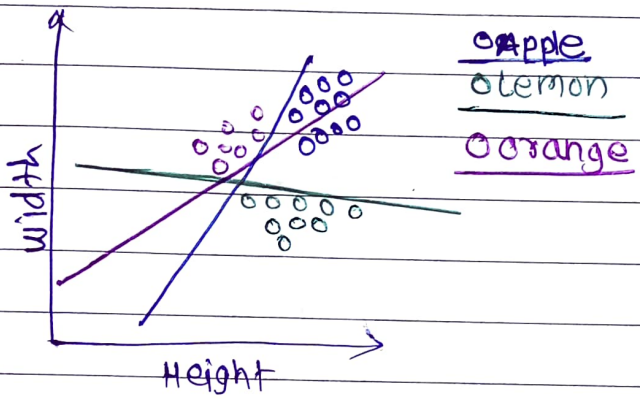


$$f(x, W, W_0) = \text{sign}(W \cdot x + W_0)$$
$$= \text{sign}\left[\Sigma\, W(i) \cdot x(i) + W_0\right]$$

dot product

Multiclass classification

Scikit learn make easy to make multiclass classifier by creating Binary classification model for each class against all other classes.



for more features space becomes 3D and so on. ---

Scikit learn makes Binary classifiers for each class for. e.g. - class 1 = Apple then one Binary classifier determines wheather object is a apple or not apple.

Similarly all the classes works and classfies object on multiple classes with many parameters.

Decision Trees

M T W T F S S

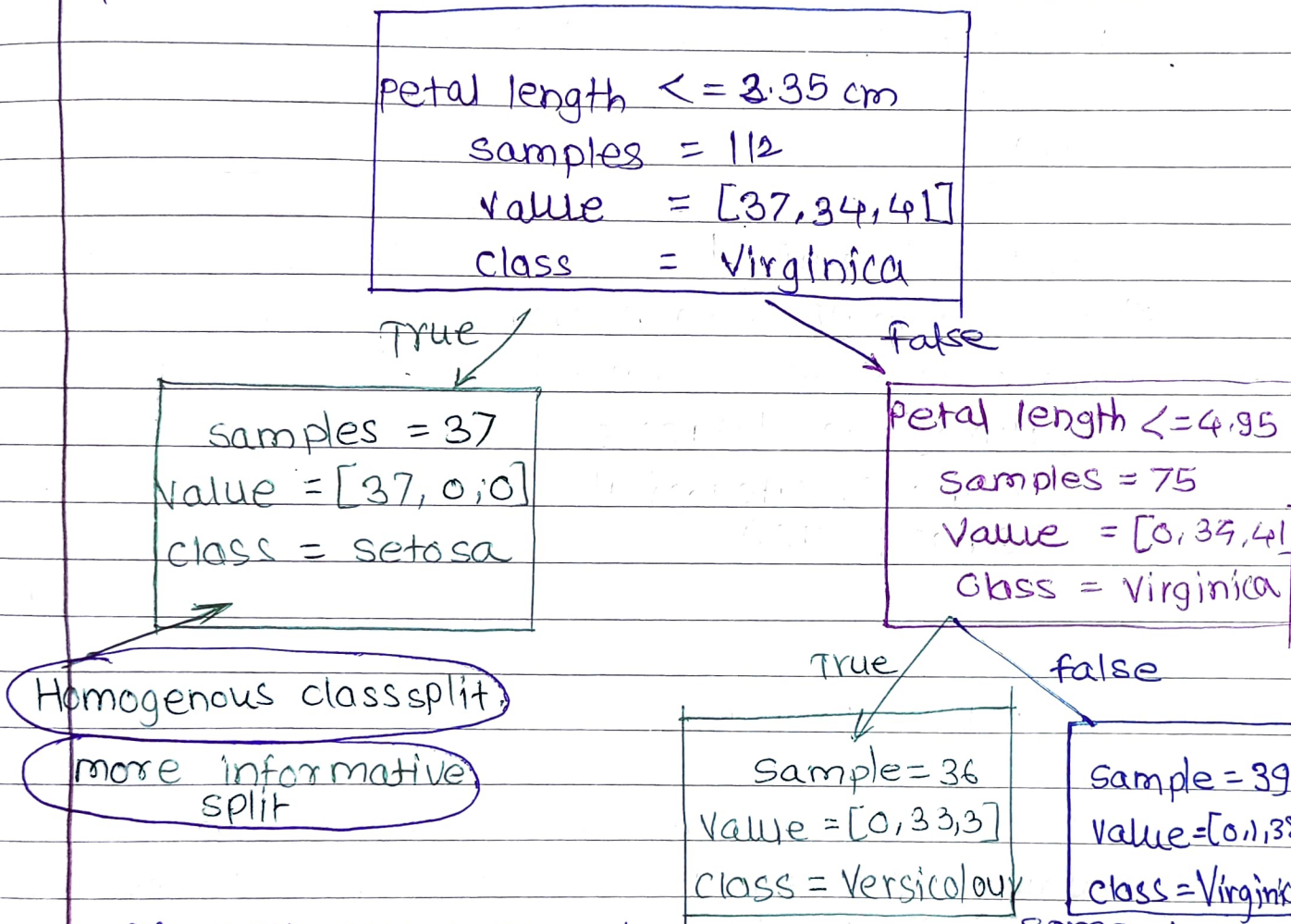
Page No.:

Date:

YOUVA

Decision Trees - It is consists of series of if-else rules which leads to target values

Informativeness of Splits - As split occurs the results must be homogenous as more as possible.



If split occurs as above e.g. then some Node does'nt having homogenous data so further classification cannot happen as certain nodes while further classification required at some Nodes.

Decision Trees be also overfitted which is prevented by restricting 1) depth of Tree d
2) Max leaf nodes - Tota
3) Min sample leaf - # of data leaves (or to avoid splitting

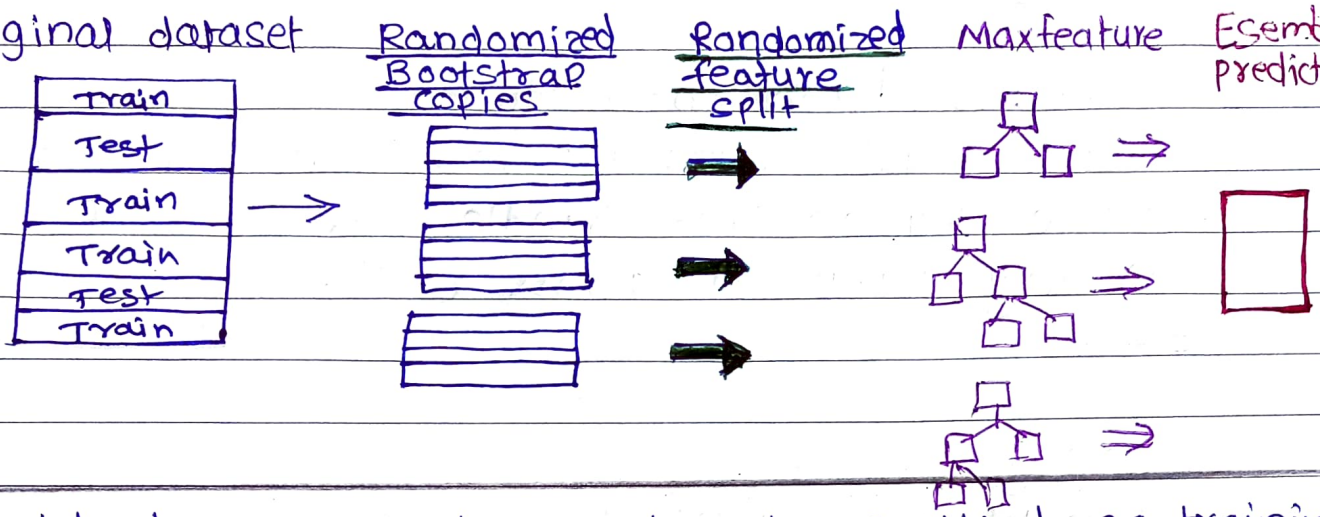
Pros and cons of Decision trees.

Pros	Cons
1) Better visualized 3) No Normalisation needed 3) Work well with mixed features in datasets (continous, categorical Binary etc.)	1) Tendency to ove Even after tun 2)

Random forests

It is a ensemble of many decision trees which uses Randomized bootstrap copies of d and Randomized feature splits which le to different kinds of decision trees which are ensembled (brought together) to make Predictions.

Bootstrap sample - Samples of training sets a are selected randomly from original data hence each sample is slightly different than other.



1) original Dataset is Randomaly splitted as training and test datasets this various copies of original data set is called bootstrap samples.
2) Feature are extracted randomly from each bootstrap coopy and
3) separate models are made for each bootstra
4) Esembling of this trees are made
5) Predictions are made for every tree in forest and combination of this prediction done as -
   i) Regression :- mean of individual tree prediction
   ii) Classification :- a) Eeach tree gives probability of eachclass
      b) probabilities are averaged.
      c) predict class with high probability,

max-feature parameter controlls the no. of feauture to be selected for each tree.
i) For High value of max-features =< no. of total feature gives similar forests with similar trees
ii) For low value of max-features = 1 trees are diverse.

M T W T F
Page No.:
Date:

Pros

i) widely used, Excellent prediction
ii) Don't require carefull Normalisation of features
iii) Like decision tree handle mixture of features.
iv) Easily parallised across CPU's.

cons:

i) models are difficult to be interpreted by Humans
ii) May not be good for High dimensional tasks like text classifiers. compared to linear mode

key parameters

1) n_estimators : Number of trees to ensemble should be larger for larger datasets (default = 10)

2) max_features : Influence on diversity of trees default works better

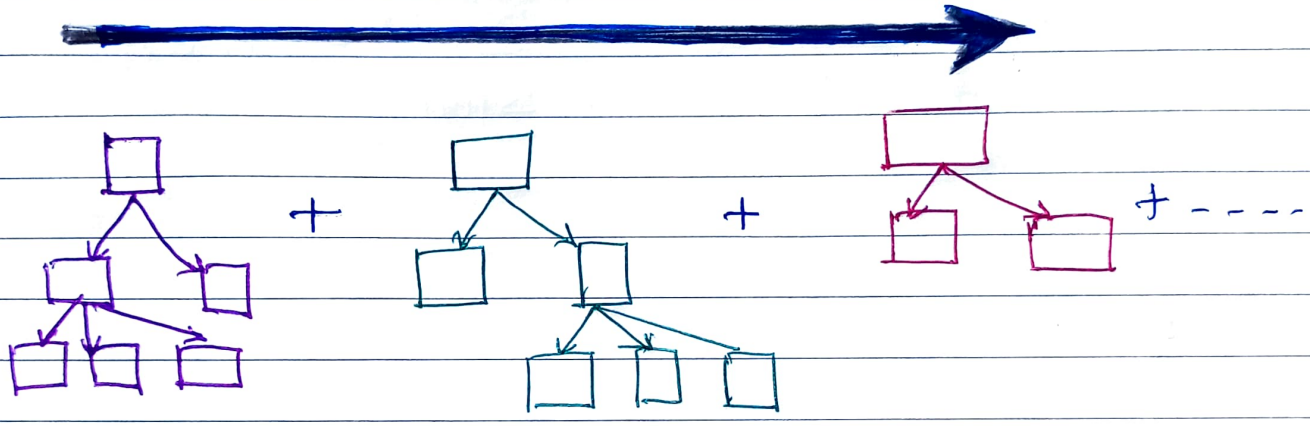
3) max_depth :- controls depth of each tree. (default = None)

4) n_jobs :- How many cores to use in parallel during traing.

choose fixed settings for random state parameter if need reproducible results.

Gradient Boosted decision Trees (GBDT)

No. of shallow trees that fixes the mistakes made by previous trees. hence with many trees is with series reduces mistakes made by trees and leads to achieve higher accuracy.



Learning Rate - The parameter which controls how new tree try to force to correct the mistakes of previous tree.

High learning rate → High complexity
Low learning rate → Low complexity.

* Pros and cons are similar to Random forests.

Key parameters

1) **n_estimators** - # of small trees to use which is used first always to best exploit memory and CPU.
2) **max_depth** - Typically very less of (3 to 5)
3) **Learning_rate** - controls ephasis on fixing errors

Cross Validation

Process in which we evaluate the models with different train test splits. we create different models and evaluate them with differentsets

K fold cross validation (3 fold)

dataset	model 1	model 2	model 3
fold 1	Train	Train	Test
fold 2	Train	Test	Train
fold 3	Test	Train	Train

choosing different values to train and test may give different Accuracy

Hence we define different models and obtain accuracy as a Average of all folds.

Leave one out cross validation. (N sample in dataset)

Dataset	model 1	model 2	N model
fold 1 = 1 data	Test	Train	Train
	Train	⋮	⋮
	⋮	Test	⋮
	⋮	⋮	Train
fold N	Train	Train	Test

only one data used as Test data and all other are Training data for N data items their are N models.

Cross Validation

M T W T F S S | Page No.: | Date: | YOUVA

Process in which we evaluate the models with different train test splits. we create different models and evaluate them with different sets

K fold cross validation (3 fold)

dataset	model 1	model 2	model 3
fold 1	Train	Train	Test
fold 2	Train	Test	Train
fold 3	Test	Train	Train

choosing different values to train and test may give different Accuracy

Hence we define different models and obtain accuracy as a Average of all folds.

Leave one out cross validation. (N sample in dataset)

Dataset	model 1	model 2	N model
fold 1 = 1 data	Test	Train	Train
	Train	⋮	⋮
	⋮	Test	⋮
	⋮	⋮	Train
fold N	Train	Train	Test

only one data used as Test data and all other are Training data for N data items their are N models.

Unsupervised learning

No target values, labells

Clustering

1) Find groups in the data -
2) Assign every point in dataset to one of this group -

Clustering - Finding a way to divide a dataset into groups. (clusters).

* Data points within the same cluster should be close or similar in some way.
* Data points within the different clusters should be far or different in som away.

Hard clustering - Each point belongs exactly one cluster

Soft (fuzzy) clustering - Each point having certain weight, score or probability of membership of certain cluster.

K-means clustering -

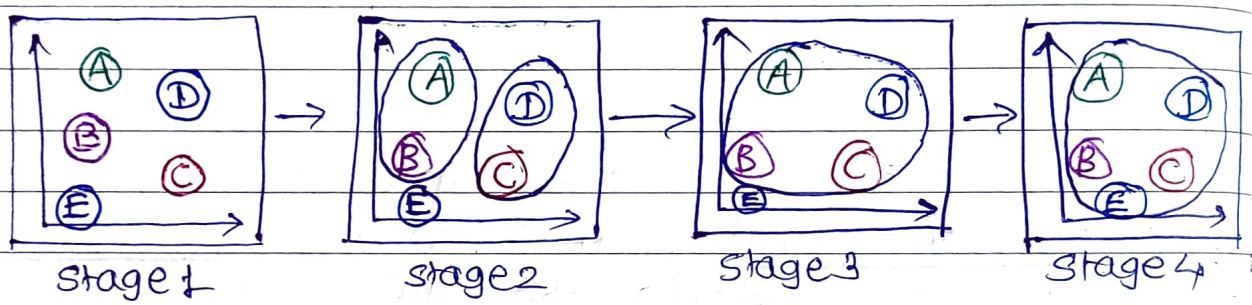
1) Pick a number of clusters 'k' you want to find then pick a random point for clusters.

2) A - Assign every point in cluster to single Point selected randomly, which is nearest one.

B - Update each cluster centre by replacing with mean of all points of same clu

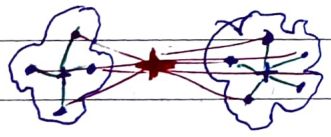
3) Repeat the (2) process until cluster converge to stable solution.

Agglomerative clustering

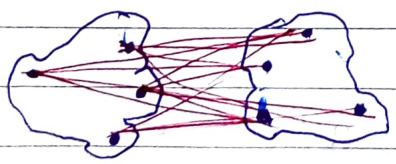


Linkage criteria - in scikit learn

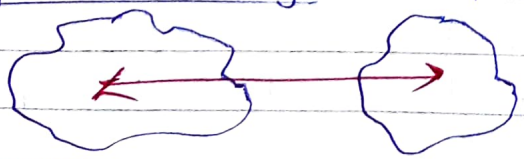
1) Ward's method - Least increase in total varia around cluster centroids.



2) Average linkage - Average distance between clusters.



3) Complete linkage - Max distance between cl

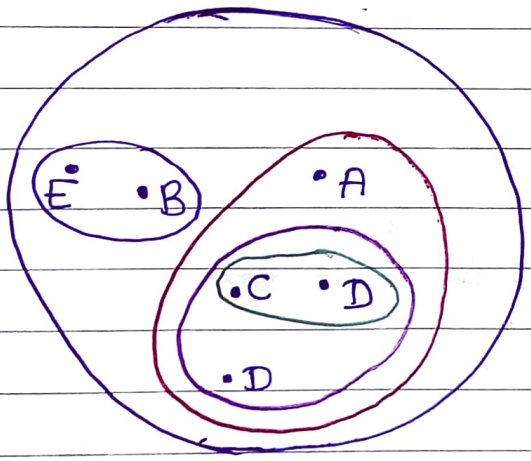


* Average linkage works well in most of the Applications

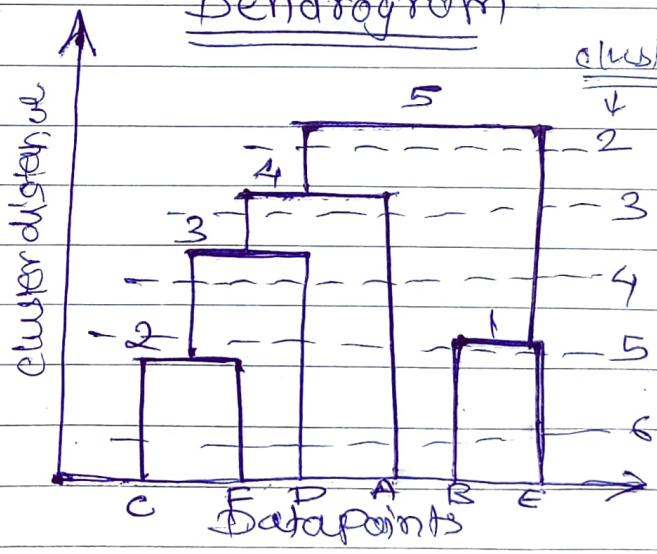
Agglomerative clustering

1) Initially each instance is separate cluster then
2) Each cluster get linked to other clusters which are certain distances according to linkage types specified.
3) This process of combining clusters is repeated until no. clusters shrinks to specific number of clusters.

Hierarchical clustering

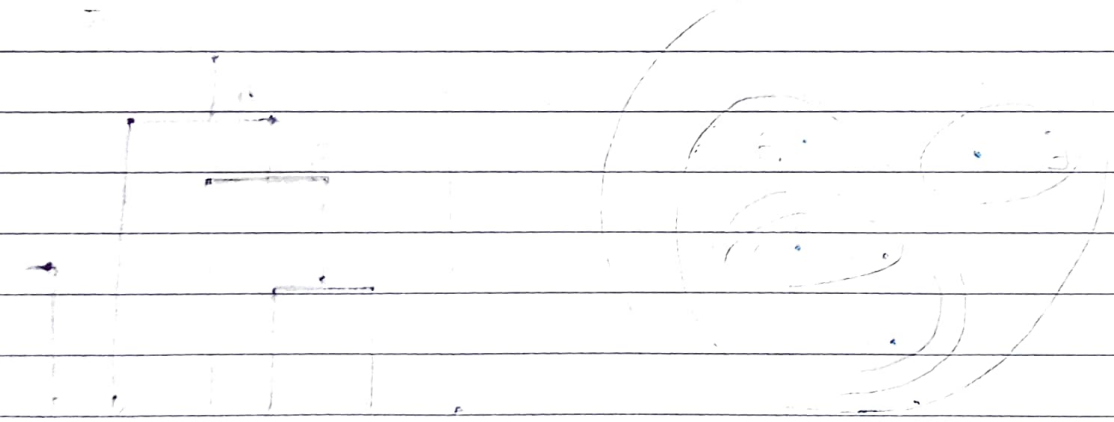


Dendrogram



1) One cluster is a part of/instance of another cluster.
2) Distance metric is made by finding distance between two points form a matrix of such distance metrics.
3) The two object of clusters merged together as single cluster. and now again process is repeated.

★ Dentrogram plots point v/s distance of points on x and y axises.

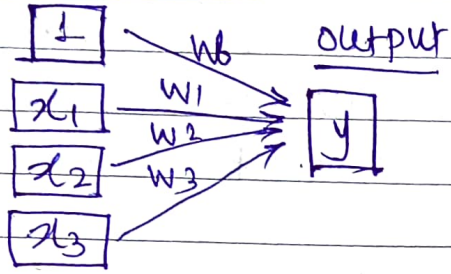


Neural Networks for regression

M T W T F S S
Page No.:
Date:
YOUVA

Linear Regression

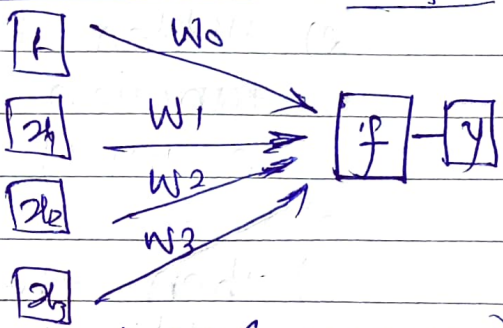
Input feature



$$\hat{y} = w_0 + \hat{w}_1 x_1 + \hat{w}_2 x_2 + \hat{w}_3 x_3$$

Logistic Regression

Input feature

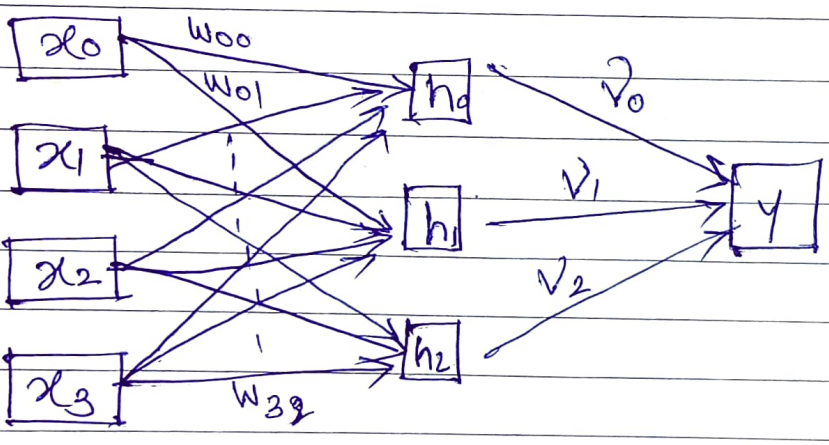


output

$$\hat{y} = \text{logistic}(w_0 + w_1 x_1 + w_2 x_2 + w_3 x_3)$$

(MLP) Multi layer perceptron with one Hidden layer.

Input features Hidden layer output



$$\hat{y} = v_0 h_0 + v_1 h_1 + v_2 h_2$$

$$h_i = \tanh(w_{i0} x_0 + w_1 x_i + w_2 x_{ii} + w_3 x_{iii})$$

Activation function

☆ There can more than one hidden layers.

M T W T F S S
Page No.:
Date:
YOUVA

1) By addition of addition process layer called hid layer in Logistic Regression represented by se of boxes [h0] [h1] [h2] which called hidden boxes

2) Hidden units in hidden layers computes non function for each feature with weighted su in input. resulting in output values $V_0, V_1,$

3) then the MLP computes weighted sum of th hidden unit outputs to form final outp value ($\hat{Y}$ hat).

4) Non linear function that hidden unit applie is called Activation functions.

Data leakage

1) When data we using to train contains information about what you are trying to predict.

2) Introducing information about targets during training that would not legitmately be available during actual use.

E.g. - Including the label to be predicted as a feature
- Including test data with training data :
data to be predicted as which fruit has one feature with its actual label.

e.g. Predict if user on financial site will open an account or not? and asking account id. which could be only filled after opening account.

Model Evaluation

M T W T F S S
Page No.:
Date:
YOUVA

Represent - Train - Evaluate - Refine cycle

Representation
Extract and select object features

→

Train models
Fit the estimator to the data

↑

↓

Feature and model Refinement

←

Evaluation

Evaluation method are need to be choose wisely that match the goal of application.

We compute selected Evaluation metric for the multiple different models. and then select the model with best evaluation metric.

1) Accuracy with imbalanced class
Just looking at accuracy for evaluating the model is not sufficient sometimes.

e.g. suppose we have two classes
1) Relevant - Positive class
2) Irrelevant - Negative class

out of 1000 randomly selected items, on average only 1 item is Relevant (Positive) and all other are Irrelevant (Negative) then even if we only define a dummy classifier that too gives accuracy of **99.9%**

Dummy classifier - scikit provides a dummy classifier class which produce model which don't train model on training dataset just

gives results based on majority of class as depending upon wheather majority of instances belongs to Negative or positive class.

Dummy classifier is not used for actual applicati

Hence Accuracy cannot be always reliable to be used as Evaluation metric of model.

$$\text{Accuracy} = \frac{\#\text{ correct predictions}}{\#\text{ instances}}$$

If Accuracy of model is same as dummy classifier Accuracy is close to Null accuracy line.
1) Because of large imbalance of class
2) Missing important features or parameters

2) Confusion metrics - Accuracy is not always para we also need to look for the consequences o mistakes made by models.

e.g. model that predict BrainTumour.

True \ Predicted	Negative	Positive
Positive	FN (4)	TP (7)
Negative	TN (107)	FP (26)

Here even if person have tumor and model predicts it as negati that is not acceptable.

$$\text{Accuracy} = \frac{\#TP + \#TN}{\#TP + \#TN + \#FP + FN} = \frac{107+7}{107+7+26+4} =$$

$$\text{Error} = \frac{\#FN + \#FP}{\#FN + \#FP + \#TP + \#TN} = \frac{4+26}{4+26+107+7} =$$

Accuracy = 1 - Error

Precision -

$$\text{Precision} = \frac{TP}{TP+FP} = \frac{7}{7+26}$$

Recall -

$$\text{Recall} = \frac{TP}{TP+FN} = \frac{7}{4+7}$$

Recall and Precision trades are needed as high or low depending upon the application
e.g. Tumour detector require high recall.

F-score - Generalises **F-1** score for combining precision and recall into a single number.

$$F_1 = \frac{2 \cdot \text{Precision} \cdot \text{recall}}{\text{Precision} + \text{recall}} = \frac{2TP}{2TP + FN + FP}$$

$$F_\beta = (1+\beta^2) \cdot \frac{\text{Precision} \cdot \text{recall}}{(\beta^2 \cdot \text{Precision}) + \text{recall}} = \frac{(1+\beta^2) \cdot TP}{(1+\beta^2)TP + \beta FN + FP}$$

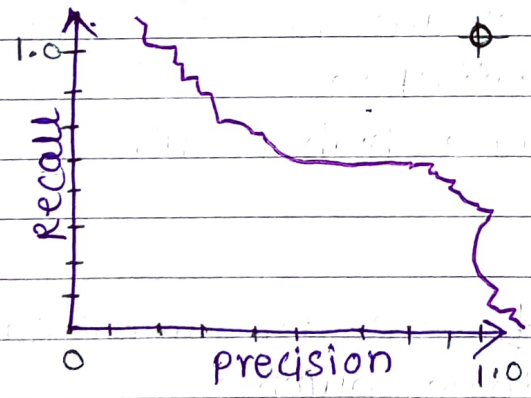
β Allows users to adjustment of metric to contro the emphasis on recall vs precision.

β = 2 - Recall oriented users
Falls negative hurt performance more than Falls Positive.
e.g. Tumour detection.

β = 0.5 - Precision oriented users
False negative hurt less than false positiv

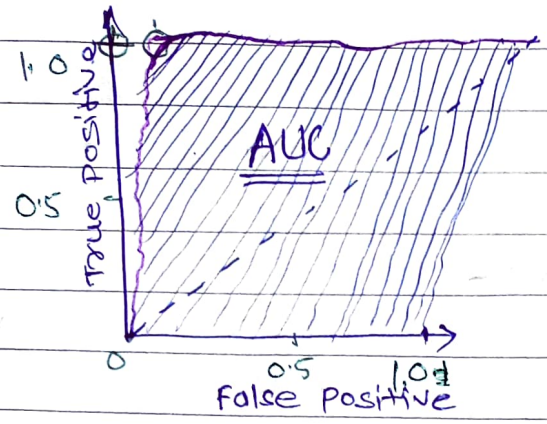
Hence apart from accuracy various parameter can be used as Evaluation metric for model depending upon the goal of model.

Precision Recall curve
Ideal point - Top right corner
Precision = 1
Recall = 1



ROC curve - (

Ideal point - Top left corner



Ideal
False positive = 0
True positive = 1

AUC should be High

AUC = 0 Bad classifiers
AUC = 1 good classifiers.

Multiclass Evaluation

Regression Evaluation

Typically used R^2 score for evaluating regression.

Alternative metrics are -

1) mean absolute error - absolute diff. of target & pre
2) mean squared error - Squared diff. of target & pre
3) median absolute error - median of error distribu

To determine wheather R^2 score is good or not we take a help of dummy regression model and compare R^2 score of both models.