Lecture #19

Regression

over-view of previous lecture

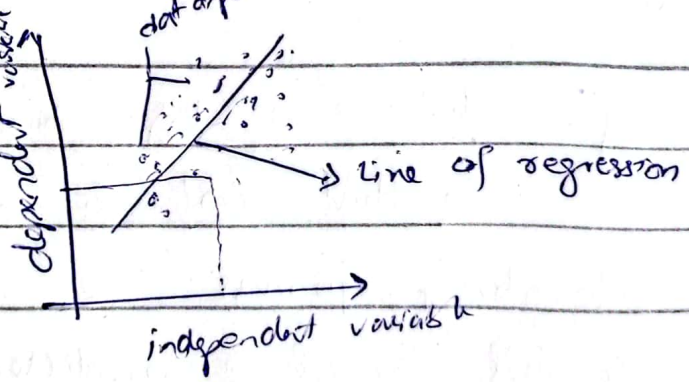
- output would be in numerical value.
- target variable ⇒ numerical

Algorithms

① Linear Regression

↳ supervised machine learning for predicting a continuous output based on one or more input features.

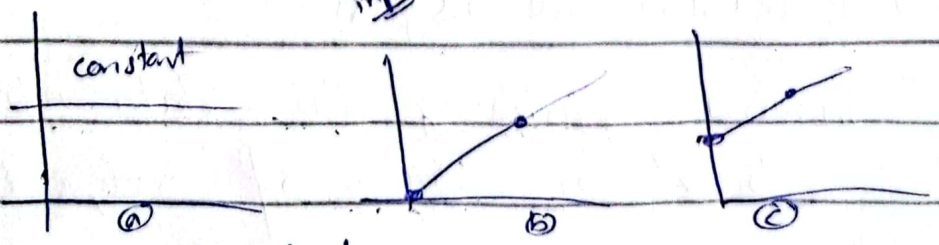
- a line touching / close to data points
- helps to predict actual data



....

Model : $f_{w,b}(x) = wx + b$ (w, b → parameters)

(x → input)



(i) one feature ⇒ uni-variant regression

(ii) multiple features ⇒ multi-variant regression

w → weight
b → bias

Our goal: finding those parameters (w, b) that when value of x gives the output close to our data actual value.

⇒ model training ⇒ learning the value of w and b so that we can get the result.

Loss/Cost function (Tell us difference b/w actual & predicted)
↳ distance function

input data → plotted — exap:

اس w اور b randomly لیا اور wx پر لیا - آپ کا
رزلٹ دیکھیں - (Prediction Result) دو (Actual) سے
کتنا فاصلہ ہے .
اس فاصلہ کو ماپنے کے لیے (cost function) استعمال کرتے ہیں

if distance is large tune our value of w so that cost is reduced.
iteratively process
untill actual & predicted distance is minimum.

(2) Decision Tree Regressor

- involves partitioning the data into sub-sets based on value of independent variables.

⇒ leaf node ⇒ average score/value

③ Random forest regressor

④ Gradient Boosting Regressor

"build multiple decision trees sequentially, each one correcting the error"

Implementation

dataset - USA housing

⇒ ensure data is correct, clean and ready to requirement

① Data Pre-processing
- check data
- check shape (5000, 7) (rows, cols)
- view/check data (example)
- .info()
- handling null value
- drop/substitute
- .describe() # 5-number summary

→ Visualization (through heat map)

② ⇒ Getting model ready

(i) split the dataset → input & target variable

input: grab everything except target variable

target: we're trying to predict

(ii) using sklearn pre-processing (i.e standard scaling)

(iii) split the training data into — train, test

∴ (using train_test_split)

(iv) Model Training

- import model
- create instance
- pass train data into model

(v) Model Testing

- find prediction using test data
- compare with actual & predicted (np.column_stack)

one way to evaluate regression model

Residual Analysis (विश्लेषण)

- a in linear regression is a way to check how well the model fits the data
- involves looking at difference b/w actual data points & predictions from the model
- in good model ⇒ residuals should be randomly scattered around zero or plot

(over estimation , under - estimation)

visualization → distribution plot → symmetric → good regressor

→ scatter plot

(vi) Evaluation

— from metrics

(i) mean squared-error

(ii) root mean square error

→ and some is → (vi) for all algorithms.