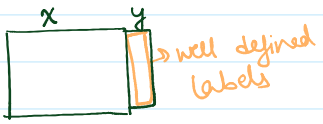
Regression Techniques - Theory

17 January 2023 23:17

Past revision ✓

- Why Train test split is done ? ✓
- Why is Feature Engineering required ? → to get those features which improve accuracy of my model.
  - o What is Standardization and why do we do it ✓
  - o What is Normalization and why do we do it ✓
- What are Performance Metrics for -
  - o For Regression (Names?) → R^2, adjusted R^2
  - o For Classification
    - What is a confusion matrix? ✓
    - What is Accuracy? ✓
    - What are Type I and Type II errors with examples? ✓ ✓

ML → well defined labels
↳ Supervised Learning
↳ Regression & classification ✓



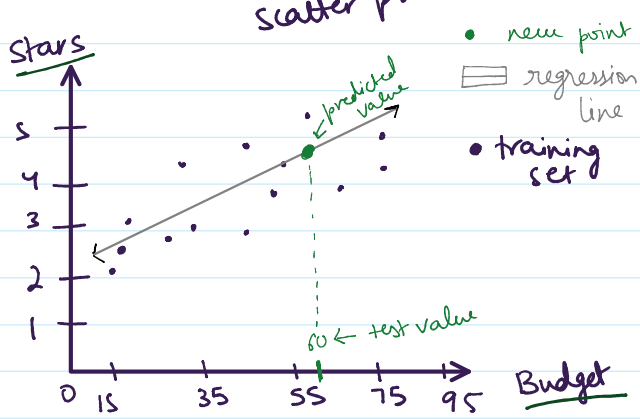
Linear Regression basic understanding

Given a set of features (X-variables : Independent variables) we need to predict another variable (y: Dependent variable)

There is some relationship b/w X, y which the regression model understands and fits a straight line to X, y so that any future value can be predicted.

eg.

Budget of film(in L)	Stars
20	2.5
18	2.4
34	2.67
50	4.85
16	3.01
12 34	4.66
⋮	⋮



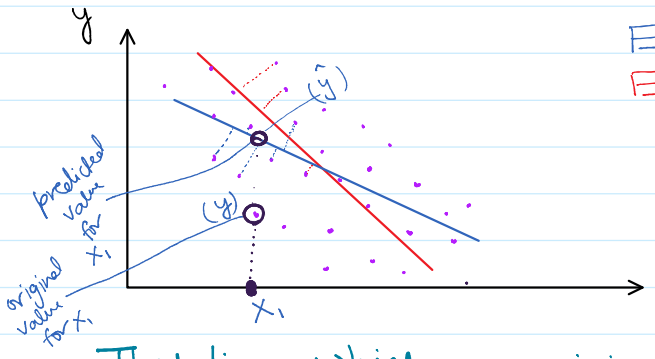
In statistics simple linear regression models the relationship between 2 continuous variables using a straight line. Using the fitted line and given a new set of independent (X) variables, the model will be able to predict the y-value

Bi-variate analysis

2 variables are used in a simple linear regression model using a line of best fit.

Scatter plot & Line of best fit

Scatter plot & Line of best fit



possible line 1 — prediction error

possible line 2 — prediction error

how to calculate which line is better?

That line which gives minimum prediction error on my training dataset is the best model that I can get.

Calculation of Prediction errors:

NOTATION:

y_i → original y-value (TRUE VALUE)

$\hat{y}_i$ → predicted y-value (FROM LINE)

Errors and squared errors

Error (ε_i): $y_i - \hat{y}_i$ = error for i^{th} point. (can be + or −)

If I calculate $\sum(y_i - \hat{y}_i)$ + & − values will cancel each other.

so I calculate $\sum_i (y_i - \hat{y}_i)^2$ sum of squared errors

The line of best fit is the one which minimizes this sum of squared errors.

The technique used to minimize this is called OLS estimation

OLS: ordinary least squares estimation.

Equation of line: $(y = mx + c)$

Regression equation: $y = \beta X + \alpha + \varepsilon$

for each point the equation is $y_i = \beta X_i + \alpha + \varepsilon$

β → slope coefficient

ε → error

α → intercept coefficient

Once OLS method has calculated $\boxed{\alpha, \beta}$ by minimizing ε^2 given any X-value a predicted y can be calculated as:

$$\hat{y}_i = \alpha + \beta X_i$$

Method of least squares OLS

Once α, β are calculated by OLS which minimizes ε^2 my regression model is fit and ready to be used.

Cost function minimisation

In ML lingo cost function is something like $\sum_i (y_i - \hat{y}_i)^2$ which we need to minimize to fit a model.

Assumptions of linear model:

1. Independence of observations ✓
2. ~~No influential values~~
3. X variables are uncorrelated ✓
4. Linear relation among X,y
5. Errors are normally distributed
6. Error terms must have constant variance ✓

✓ → most imp.

Before using the regression model make sure these assumptions hold true.

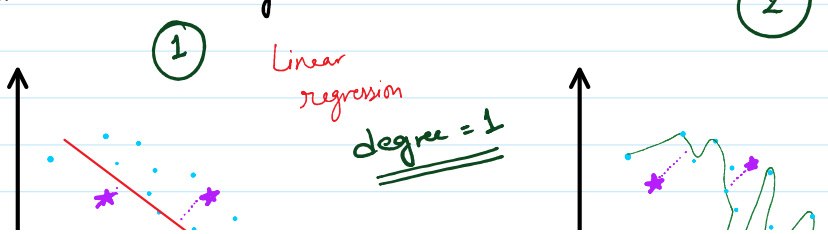
MLRM : Multiple Linear Regression Model

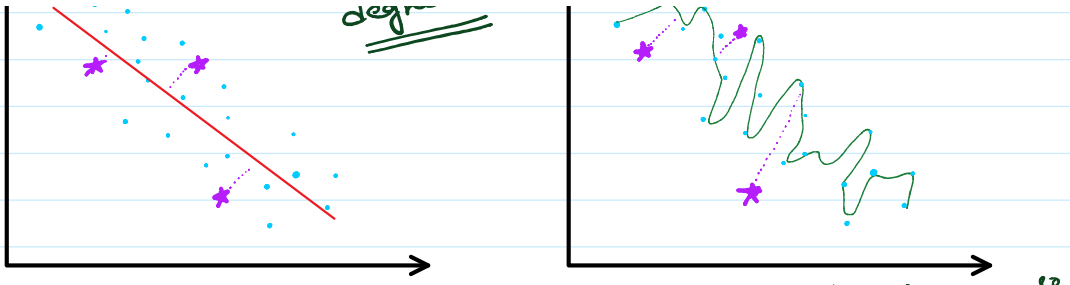
$$y = \alpha + \beta_1 X_1 + \beta_2 X_2 + \cdots + \beta_n X_n + \varepsilon$$

Using OLS $\alpha, \beta_1, \beta_2, \ldots, \beta_n$ can be similarly calculated.

Overfitting

Overfitting means fitting your Model to your training dataset in such saturation that it only performs well on the training data & not in the real world because it has learnt the dataset itself.



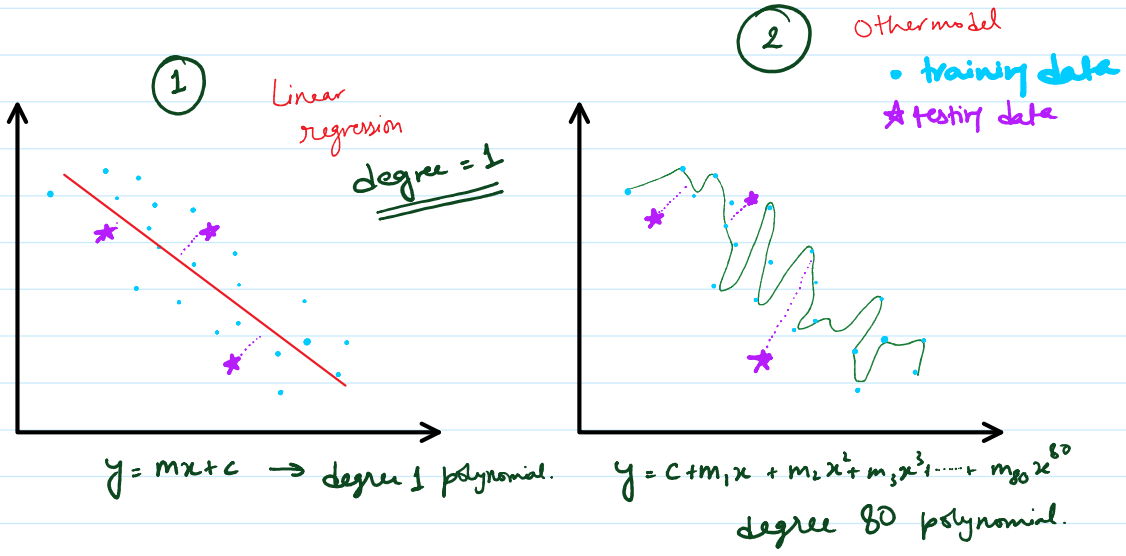


a higher degree polynomial (complex model) can overfit the data easily providing very less error on training dataset but is not useful when new data(*) in the real world is given to it.

Bias and variance defn and tradeoff

Bias and Variance are 2 extremely important concepts in ML which we'll discuss now. Please note that this topic is not just related to Regression algorithms, but with ML as a whole. Bias or Variances exist for all sorts of ML algorithms, not just regression. Bias and Variance have very simple definitions as given below -

- **Bias** : Error in Training data is called Bias.
- **Variance** : Error in Test data is known as variance.



In model 1 Bias is high and variance is low
In model 2 Bias is low and variance is high

testing training

Because the **Straight Line** can't be

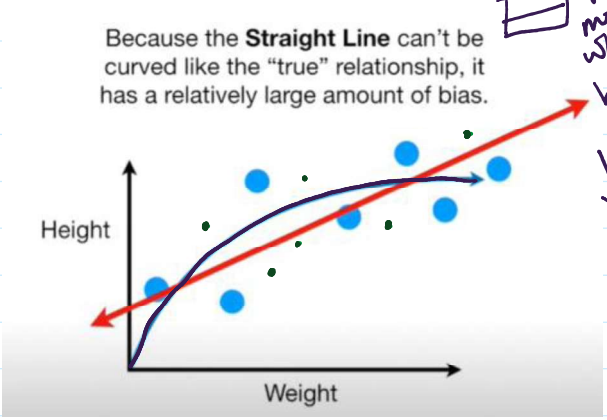
Because the **Squiggly Line** can handle the arc

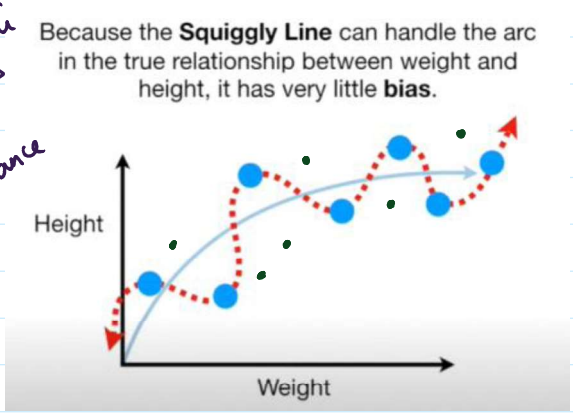
test

● testing ● trainig

best model w/ low bias & low variance

test set





Squiggly line is an overfit model and straight line gives consistent results in all datasets (training and testing)

An ideal model has low bias and low variance but in real world its not possible

A good model minimizes both

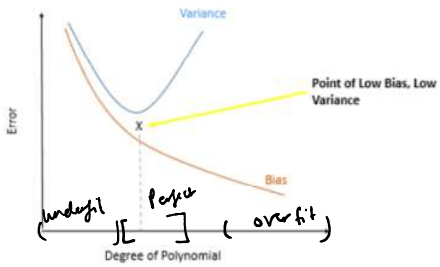
Overfit model has high variance and low bias (squiggly line)

Underfit model has both high (straight line in above img)

Impact of additional Features

As we add on more features to our model training dataset, the degree of the polynomial increases. With this increase in degree of polynomial, the Train error will always decrease because any addition of a feature to the model will explain the target variable by chance up to some extent. This is the main reason why models are Overfitted, because data scientists use a lot of insignificant variables for training the model. In other words, we can say that the model goes towards low bias as we add more features.

But the behavior with Variance is not same. The below chart explains how Bias and Variance in a model is impacted as we add more and more features –



(best model)

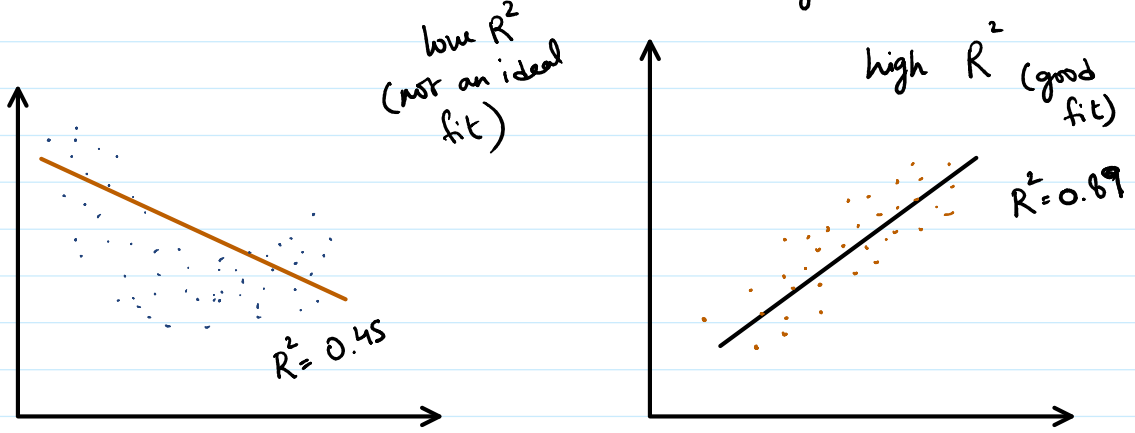
R squared and adjusted R squared

R^2 is a performance metric used with Regression models to judge how well a model is for prediction and how well it fits the training data.

R^2 → lies b/w 0 & 1

0 means nothing is explained by model
1 means very accurate model

In Practice models > 0.7 R^2 are treated as good models



difference

Since we know adding complexity in the model will reduce bias but at the same time variance will increase.

simple R^2 always increases when new features are added to data eg.

$M1 \Rightarrow \quad y = \beta_1 X_1 + \beta_2 X_2 + \varepsilon_1$

$M2 \Rightarrow \quad y = \beta_1 X_1 + \beta_2 X_2 + \beta_3 X_3 + \varepsilon_2$

Since M2 has an additional feature $R^2(M1) \leq R^2(M2)$ it does not take into account that error in test dataset will increase. So to ammend this limitation adjusted R^2 was invented.

Adjusted R^2 puts a penalty when increasing the features so adjusted R^2 (M2) could be lower also.

$\Rightarrow$ Adjusted R^2 takes into account the variance tradeoff also and hence it's a better metric to see for regression.

Other metrics:

Other than R^2 we can calculate the following.

MAE: mean absolute error

$$\frac{1}{n} \sum_{i=1}^{n} |y_i - \hat{y}_i| \rightarrow \text{mean of absolute error.}$$

MSE: mean squared error

$$\frac{1}{n} \sum_{i=1}^{n} (y_i - \hat{y}_i)^2 \rightarrow \text{mean of squared error}$$

$n \sum_{i=1}$

MAPE: mean absolute percent error → gives ans in %

The formula for MAPE is given as –

$$MAPE = 1/n \left(\Sigma \left| \frac{(\hat{y} - y)}{y} \right|\right)$$

$$\frac{1}{n} \sum \left(\left| \frac{y_i - \hat{y}_i}{y_i} \right| \right)$$

Move to python nb