Moringa School | Sept

www.moringaschool.com

DSPT08 - Phase 2

Regression

14/10/2024

Session Expectations

Online edition	The usual
→ Participate in the discussions → Mute yourself when you are not speaking	→ Be present → Be critical → Be curious → Joining using laptop

Regression

Regression is a statistical method used to examine the relationship between a dependent variable (outcome) and one or more independent variables (predictors)

Training and test set

- **Training Set** - Include independent and dependent variables data used to train a model to predict
- **Test Set** - Validate how good a model is used to predict
  - ○ Sklearn selection -

    from sklearn.model_selection import train_test_split

  - ○ Pandas - df.sample(0.8 , random_state= 70)
  - ○ Numpy - np.random.rand(len(df)) =< 0.8

Linear Regression

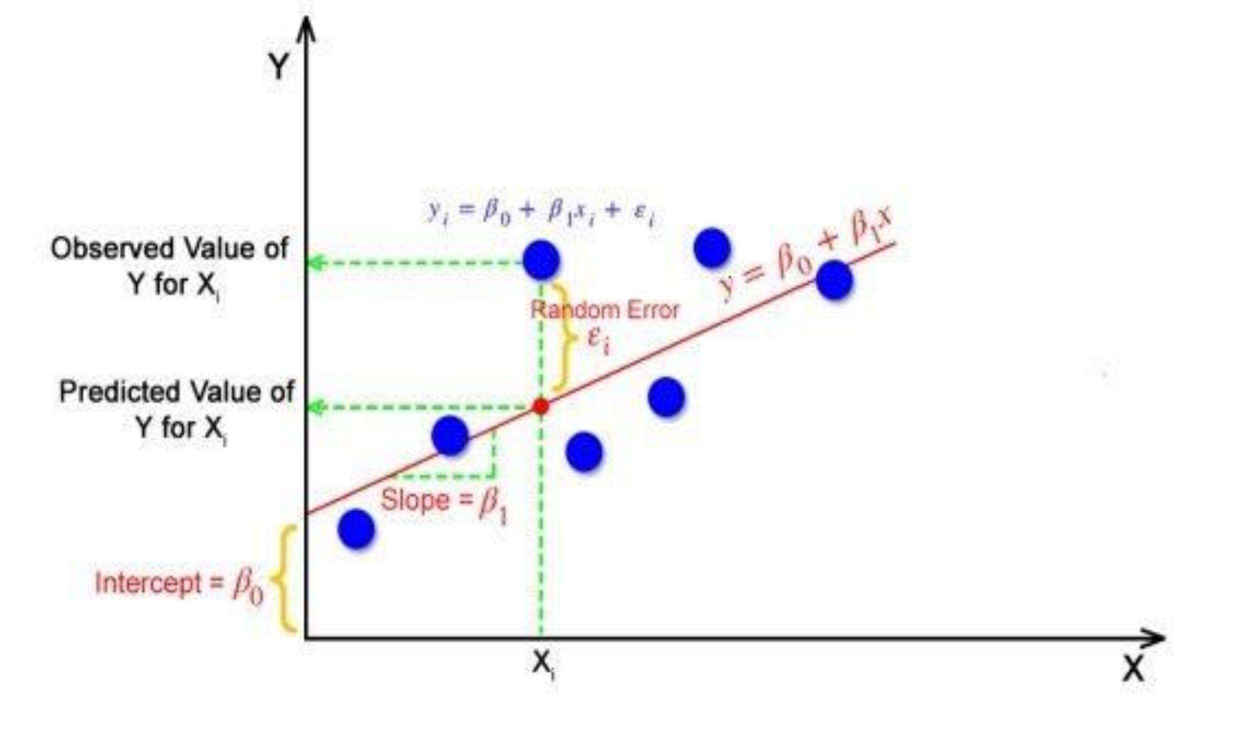
Linear regression $y = mx + b$ - *Single variable*

$y= b + m1x1 + m2x2 + m3x3 + mnxn$ - *Multiple independent variables*

In machine learning, we get to optimize m & b

Linear Regression

- Linear regression analysis is **used to predict the value of a variable based on the value of another variable**.


Y
$y_i = \beta_0 + \beta_1 x_i + \varepsilon_i$
Observed Value of Y for X_i
$y = \beta_0 + \beta_1 x$
Random Error
ε_i
Predicted Value of Y for X_i
Slope = β_1
Intercept = β_0
X_i
X

Cost Function/ Error Function

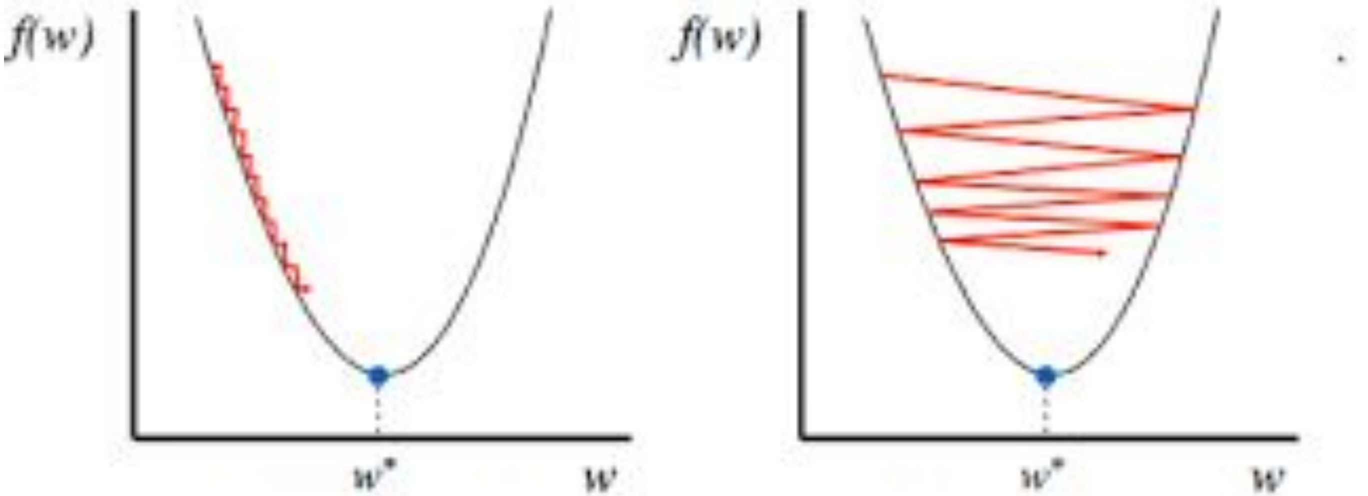
Cost function - helps to access how good our regression (best line) fit/predicts

- Compute the difference between data for the dependent var and prediction from regression line
- Square that value
- Add you all the values you've obtained
- Divide by the number of the data
- By doing so the error function is average value of (y-(mx+b)2

Learning Rate

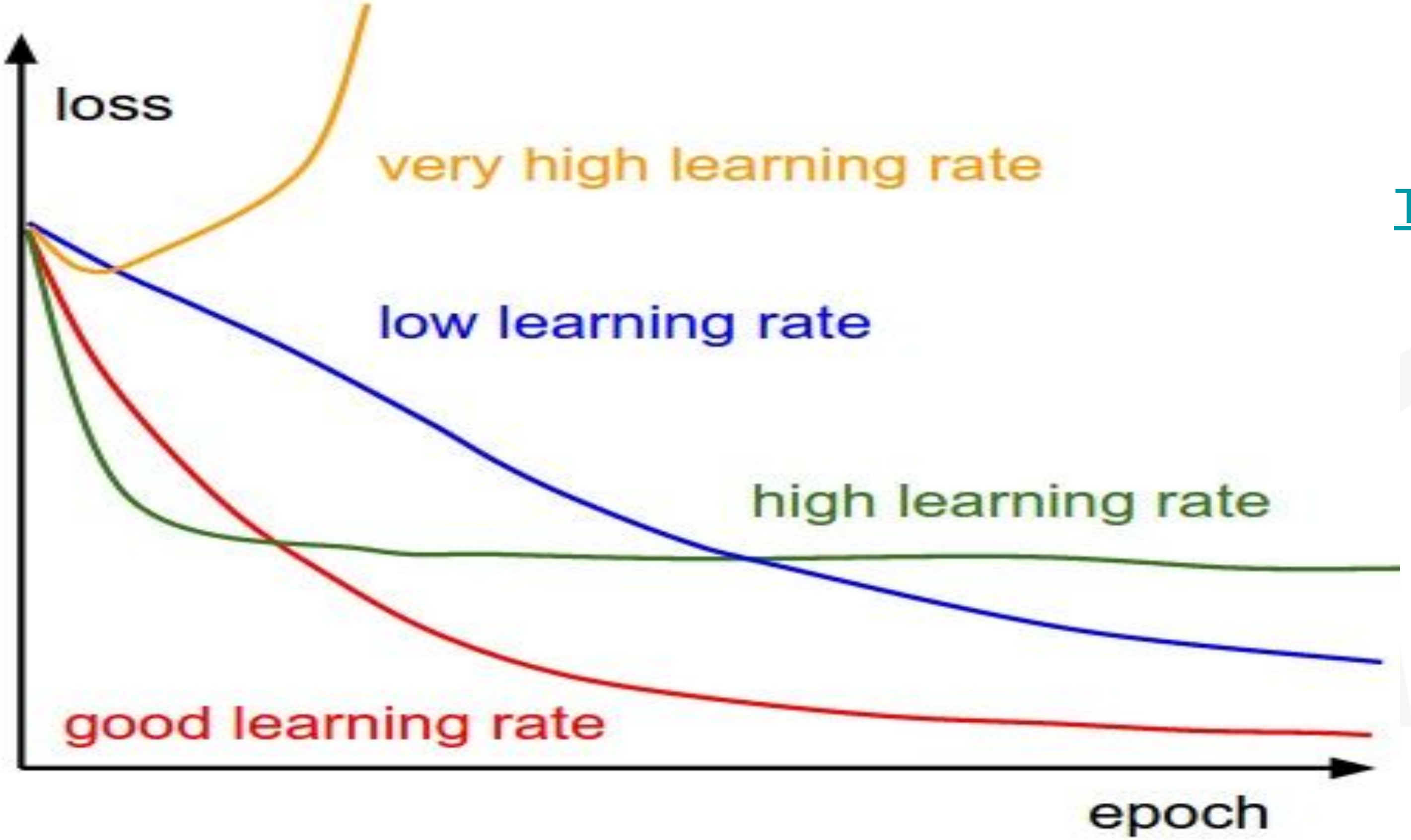
- Change on m & b iteration is based on learning rate
- Learning rate is is a parameter that controls how much a change in model response to the estimated error each time the model predicts
- A low learning rate means slow model to train
- A high learning rate fails to take the optimum data points

NOTE: The learning rate affects how quickly our model can converge to a local minima (aka arrive at the best accuracy)



Too small: converge very slowly

Too big: overshoot and even diverge


loss
very high learning rate
low learning rate
high learning rate
good learning rate
epoch

The example

Gradient Descent

- Gradient Descent is an optimization algorithm used to minimize the cost function (or loss function) by iteratively adjusting the model's parameters (weights and biases). It is widely used in regression and machine learning models to find the best-fit parameters for accurate predictions

- Gradient Descent has direction - vector

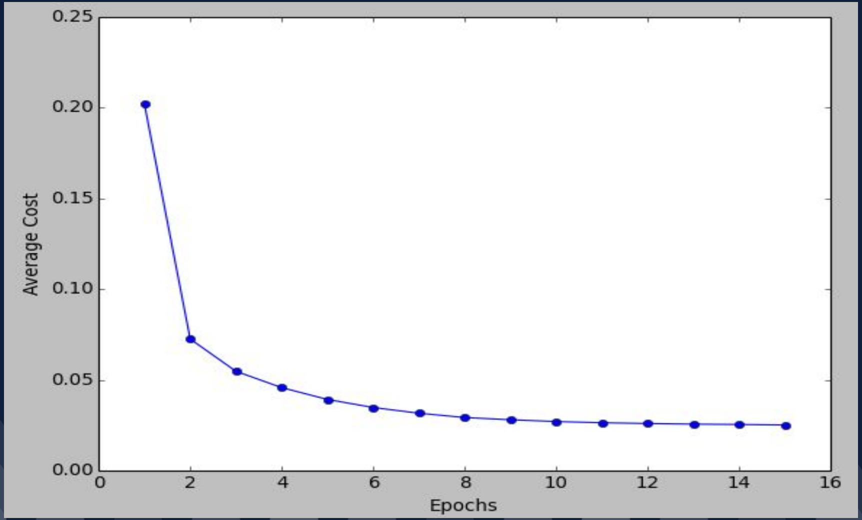
Explanation

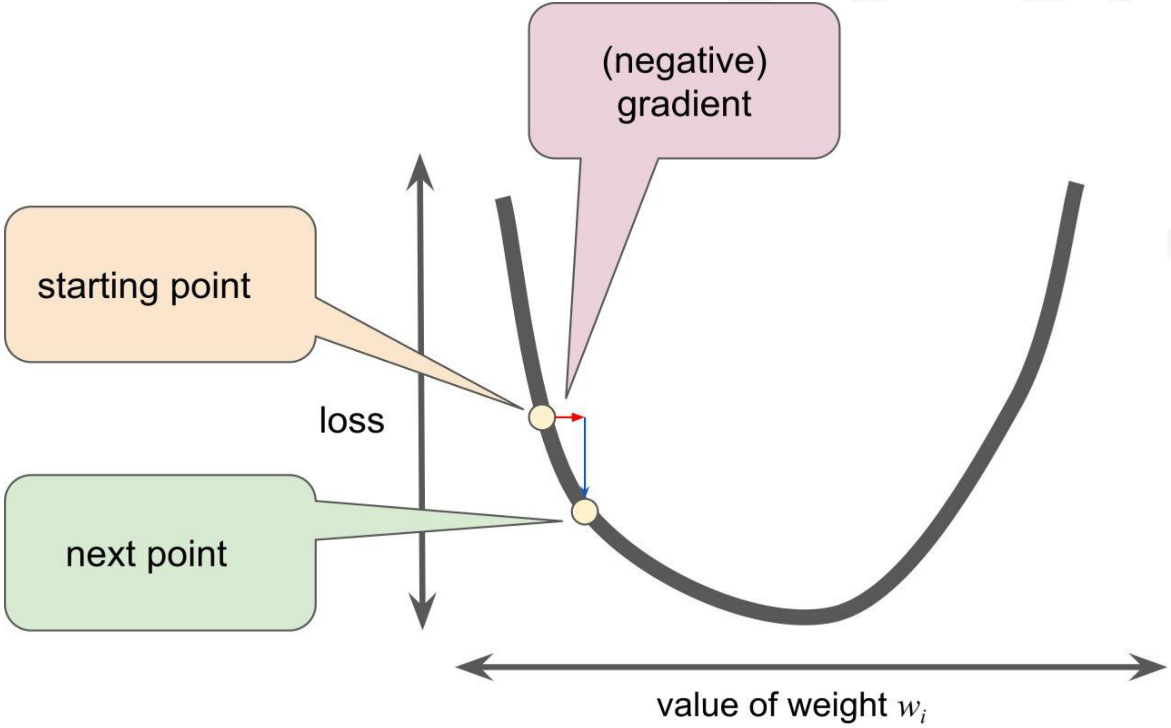
Types of Gradient Descents

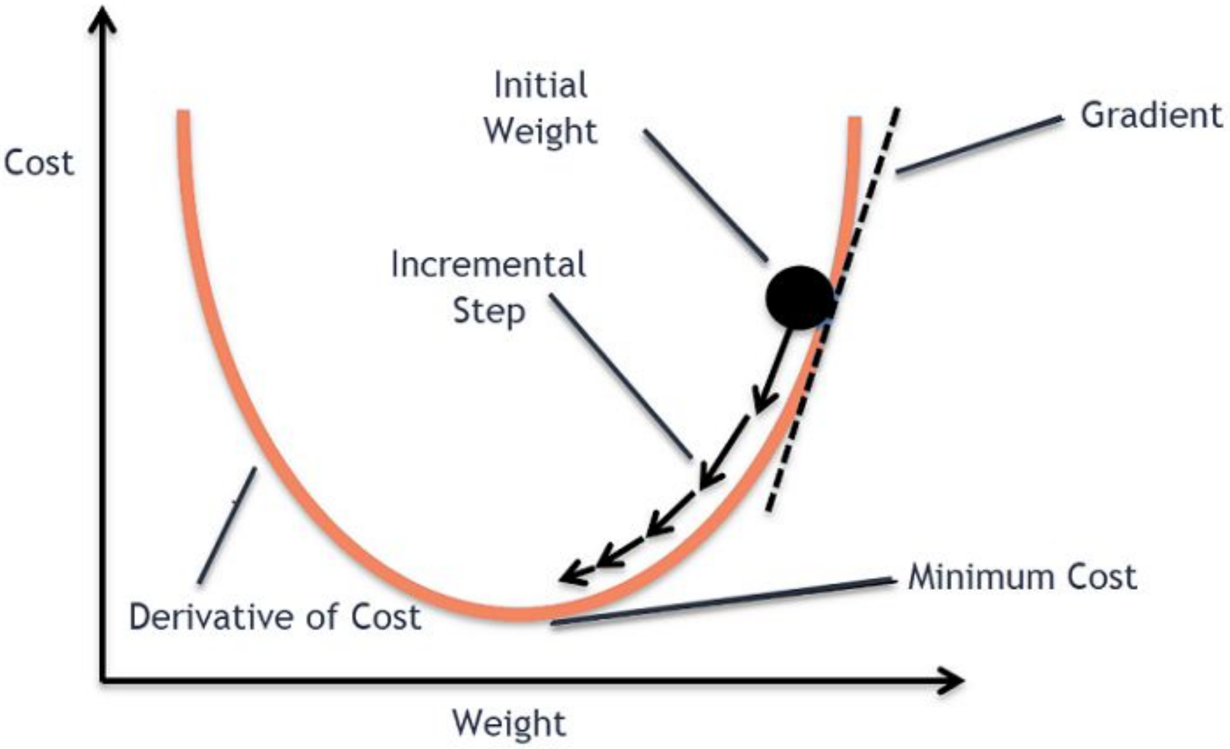
Batch Gradient Descent -

All the training points are considered   to a single step as and the mean of the gradients of all points is considered.

The technique is good for conex or smooth error minifolds




(negative) gradient
starting point
loss
next point
value of weight w_i


Cost
Initial Weight
Gradient
Incremental Step
Minimum Cost
Derivative of Cost
Weight

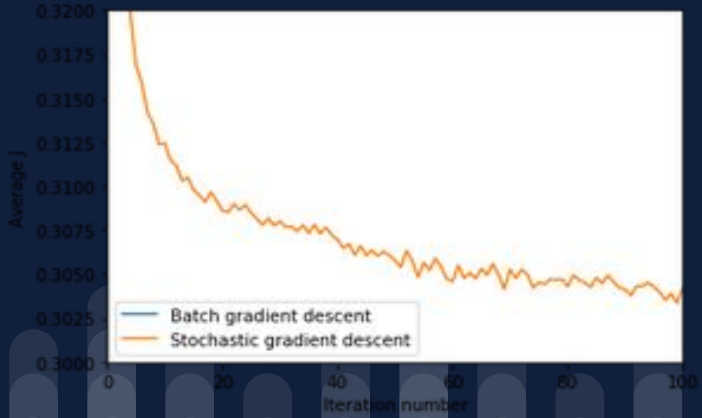
Types of Gradient Descents

Stochastic Gradient Descent -

We consider just one example at a time to take a single step

Applied in large dataset especially for deep learning

If you have 10 M data points you run 10 Min irierations which is not effective

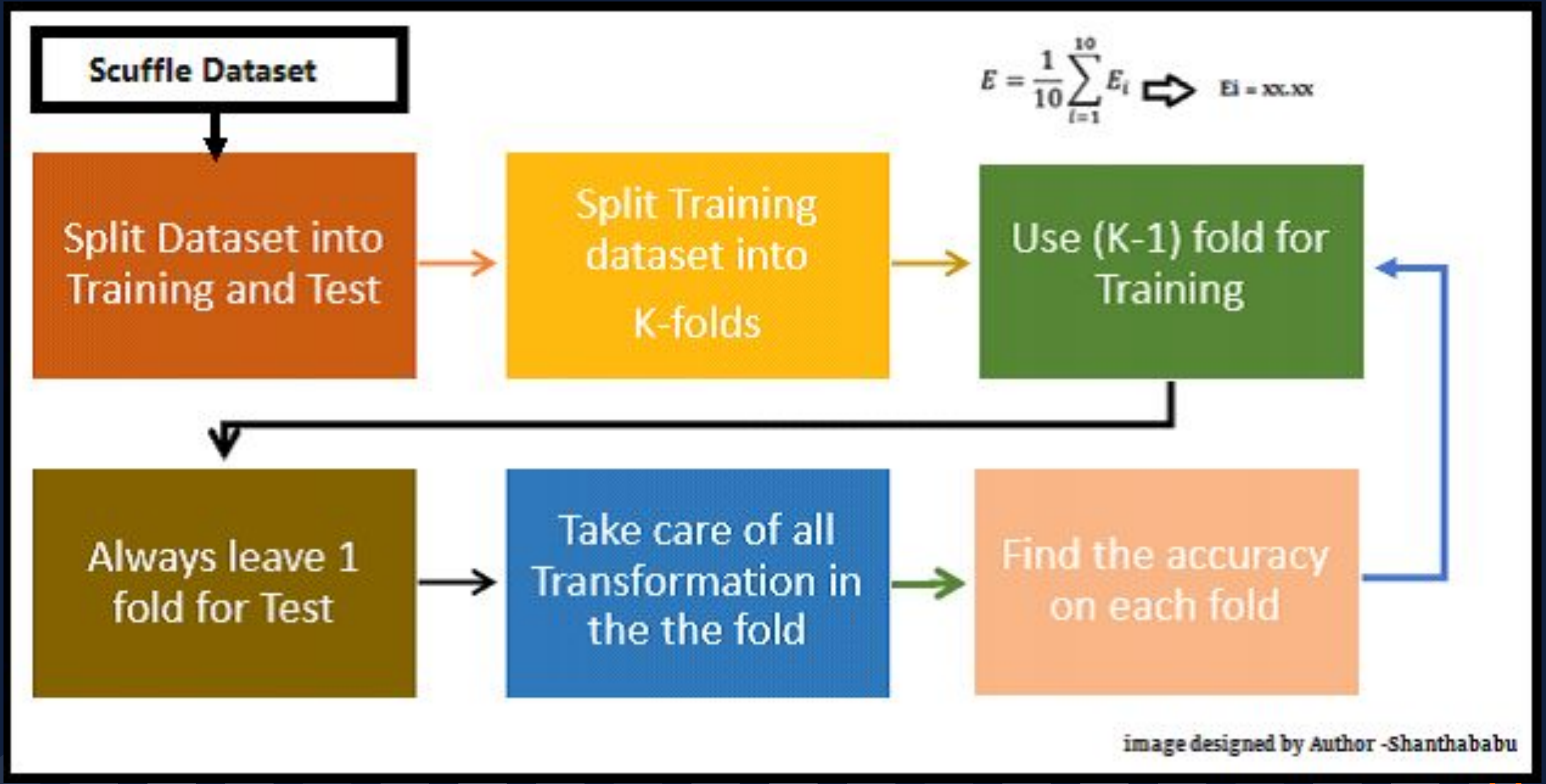


Types of Gradient Descents

Mini batch Gradient Descent-

- Applies the above techniques combined
- Batch uses smoother curves while SGD uses large dataset
- Here, mini batch is used to fixed the number of training examples, thus applying both technique reduces the computation power,
- The question is how?

Linear Optimization - Cross validation



Scuffle Dataset

$E = \frac{1}{10}\sum_{i=1}^{10} E_i$ ⇨ Ei = xx.xx

Split Dataset into Training and Test

Split Training dataset into K-folds

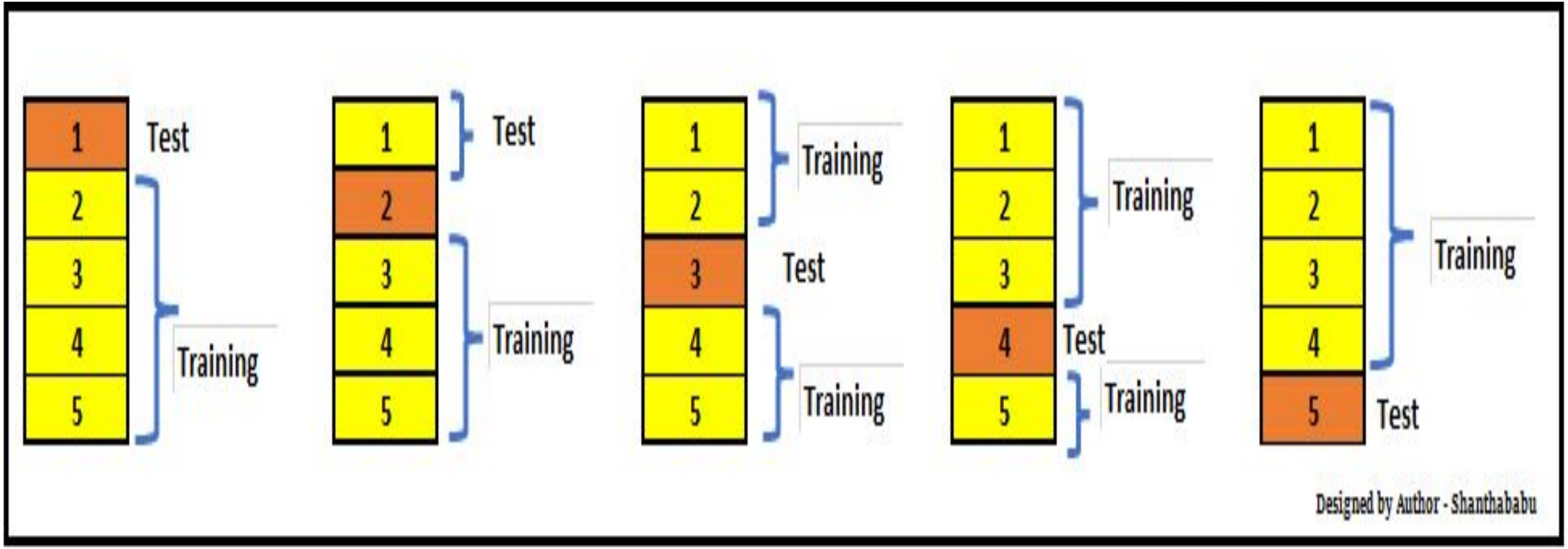
Use (K-1) fold for Training

Always leave 1 fold for Test

Take care of all Transformation in the the fold

Find the accuracy on each fold

image designed by Author -Shanthababu


1
2
3
4
5
Test
Training
1
2
3
4
5
Test
Training
1
2
3
4
5
Training
Test
Training
1
2
3
4
5
Training
Test
Training
1
2
3
4
5
Training
Test
Designed by Author - Shanthababu

Thumb rule in K-Fold Cross validation

- K should be always >= 2 to the number of records
- The optimized value of K is 10 and used with data of good size
- Too large K value leads to less variance across the training set and limit model performance across the iterations.
- The number of folds is indirectly proportional to the dataset, i.e if dataset size is too small the number folds increases
- Large values of K increases the running time of the cross-validation process

Multilinearity

Multicollinearity occurs in regression models when two or more independent variables (predictors) are highly correlated with each other

- In multivariate regression independent variables shouldn't be strongly correlated
- If there is high relationship it affects the coefficients of the our model assigned to each indep variable
- Variance Inflation Factor (VIF) - measure how much the variance of a regression coefficient model increases if your independent variable are correlated.
- A VIF of 5 indicate there is association between variables while 10 is considered to be highly correlated variables.
- Assumptions for testing   multicollinearity

Polynomial Regression

- Linear regression deals with correlated variables having a mutual relationship
- Polynomial Applies when the data has a correlation but no relationship in linearity
- I.e relationship between the independent variables x and dependent variable y expressed in nth degree
- Polynomial is done to reduce under fitting by increasing complexity
- Example on Canvas ...

Code Implementation

Overview of regression model and model evaluation

EDA & Regression model _project & Data

Thanks!

Have an amazing week ahead. Stay safe.

- Sam

We value you!