

TO find R2 score value by using following machine learning method:

1. Multiple Linear Regression:

R2 Score value in **MLR** = **0.7894790349867009**

2. Support Vector Machine:

Kernal	C=1	C=10	C=100	C=1000
Linear	-0.010102665316 081394	0.462468414233 9678	0.628879285732 0361	0.764931173859 7825
rbf	-0.083382385936 19329	-0.032273293906 71052	0.320031783205 0831	0.810206487480 8204
Poly	-0.075699655708 6089	0.038716222760 231456	0.617956962405 9799	0.856648767594 659
sigmoid	-0.075429242811 07188	0.039307143782 74347	0.527610354651 0407	0.287470694869 7563

In **svm (kernal=linear,c=1)** gives **r2_square** value is **0.856648767594659**

3. DecisionTree:

Splitter =	criterion=squared_error	criterion=friedman_mse	criterion=absolute_error	criterion=poisson
best	0.682250159645 5095	0.69676684346 89598	0.696225572173 8235	0.718509077002 2624
random	0.685227293182 3247	0.739997945917 5379	0.747379837018 312	0.724476326600 1724

Here (criterion="squared_error", splitter="best") gives model r2_square value is 0.747379837018312

4. Random Forest:

criterion	n_estimators	max_features	R2_score
squared_error	100	sqrt	0.8722842204761092
absolute_error	100	sqrt	0.8757337256192934
friedman_mse	100	sqrt	0.8680297251949043
poisson	100	sqrt	0.8716059600152415
squared_error	100	log2	0.8712922689780294
absolute_error	100	log2	0.8723776640868625
friedman_mse	100	log2	0.871378594460033
poisson	100	log2	0.8721964653950868

<i>squared_error</i>	100	None	0.8547494377130381
<i>absolute_error</i>	100	None	0.8499397360490405
<i>friedman_mse</i>	100	None	0.8549703309699938
<i>poisson</i>	100	None	0.8556145935293948
<i>squared_error</i>	500	sqrt	0.8718583712387833
<i>absolute_error</i>	500	sqrt	0.8746432159550745
<i>friedman_mse</i>	500	sqrt	0.8720804525715373
<i>poisson</i>	500	sqrt	0.8734047305211126
<i>squared_error</i>	500	log2	0.8721191310381918
<i>absolute_error</i>	500	log2	0.8738158110772026
<i>friedman_mse</i>	500	log2	0.8718839306472905
<i>poisson</i>	500	log2	0.8709770263689216
<i>squared_error</i>	500	None	0.8562648066140802
<i>absolute_error</i>	500	None	0.8537889935368864
<i>friedman_mse</i>	500	None	0.8526928046116207
<i>poisson</i>	500	None	0.855938083383492
<i>squared_error</i>	1000	sqrt	0.8720462505768847
<i>absolute_error</i>	1000	sqrt	0.8738259802417034

<i>friedman_mse</i>	1000	<i>sqrt</i>	0.8729093810690829
<i>poisson</i>	1000	<i>sqrt</i>	0.8718917358341729
<i>squared_error</i>	1000	<i>log2</i>	0.8722115789189225
<i>absolute_error</i>	1000	<i>log2</i>	0.8740641167816261
<i>friedman_mse</i>	1000	<i>log2</i>	0.8724954886409342
<i>poisson</i>	1000	<i>log2</i>	0.8715554217382954
<i>squared_error</i>	1000	<i>None</i>	0.8544108919870267
<i>absolute_error</i>	1000	<i>None</i>	0.8551830267037819
<i>friedman_mse</i>	1000	<i>None</i>	0.8548508184154748
<i>poisson</i>	1000	<i>None</i>	0.8552923617139219

Best model criterion=*absolute_error*";n_estimators=100;max_features='sqrt'
 Gives R2_SCORE Value is **0.8757337256192934**

5.ADA BOOOSTING:

N_estimators	Loss	r2_score
1	Square	0.8809576998632572
1	linear	0.8809576998632572

1	exponential	0.8809576998632572
50	square	0.5185321334257109
50	linear	0.8447476929154872
50	exponential	0.6292859305668668
100	square	0.4661497994260698
100	linear	0.8447476929154872
100	exponential	0.538576314578121
500	square	0.4661497994260698
500	linear	0.8447476929154872
500	exponential	0.538576314578121

Best model n_estimators=1;loss='square,linear,exponential' Gives
R2_SCORE Value is **0.8809576998632572**

6.LightGradient Boosting(LGBM):

loss	criterion	max_features	r2_square
Squared error	friedman_mse	None	0.8833974199848822
Absolute_error	friedman_mse	None	0.8588019148018358
huber	friedman_mse	None	0.8914390761897757
quantile	friedman_mse	None	0.6289330239690663

Squared error	<i>friedman_mse</i>	sqrt	0.8900729638277464
absolute_error	<i>friedman_mse</i>	sqrt	0.8767758265920338
huber	<i>friedman_mse</i>	sqrt	0.8904302901210221
quantile	<i>friedman_mse</i>	sqrt	0.6600838602848385
squared_error	<i>friedman_mse</i>	log2	0.8900729638277464
absolute_error	<i>friedman_mse</i>	log2	0.8767758265920338
huber	<i>friedman_mse</i>	log2	0.8904302901210221
quantile	<i>friedman_mse</i>	log2	0.6600838602848385
squared_error	<i>squared_error</i>	None	0.8833974199848822
absolute_error	<i>squared_error</i>	None	0.8588019148018358
huber	<i>squared_error</i>	None	0.8914390761897757
quantile	<i>squared_error</i>	None	0.6266819312165877
Squared error	<i>squared_error</i>	sqrt	0.8900729638277464
Absolute _error	<i>squared_error</i>	sqrt	0.8767758265920338
huber	<i>squared_error</i>	sqrt	0.8904302901210221
quantile	<i>squared_error</i>	sqrt	0.6597975662815307
squared_error	<i>squared_error</i>	log2	0.8900729638277464

absolute_error	squared_error	log2	0.8767758265920338
huber	squared_error	log2	0.8904302901210221
quantile	squared_error	log2	0.6597975662815307

Best model loss='huber', criterion='squared_error,friedman_mse', max_features='None' Gives R2_SCORE Value is 0.8914390761897757

7. Extreme Gradient Boosting :

```
verbosity= 0,1,2,3; booster=gbtree, gblinear; tree_method=auto, exact, approx, hist
```

n_estimators	learning_rate	max_depth	min_child_weight	verbosity	tree_method	R2_Score
10	0.1	1	10	0	auto	0.5572062730789185
10	0.1	1	10	0	exact	0.5572062730789185
10	0.1	1	10	0	approx	0.5572062730789185
10	0.1	1	10	0	hist	0.5572062730789185
10	0.1	1	10	1	auto	0.5572062730789185
10	0.1	1	10	1	exact	0.5572062730789185
10	0.1	1	10	1	approx	0.5572062730789185
10	0.1	1	10	1	hist	0.5572062730789185
10	0.1	3	20	2	auto	0.7720925807952881
10	0.1	3	20	2	exact	0.7719619870185852

10	0.1	3	20	2	approx	0.7720925807952881
10	0.1	3	20	2	hist	0.7720925807952881
10	0.1	5	50	3	auto	0.7559833526611328
10	0.1	5	50	3	exact	0.7556763887405396
10	0.1	5	50	3	approx	0.7559833526611328
10	0.1	5	50	3	hist	0.7559833526611328
40	0.1	1	10	0	auto	0.7502782344818115
40	0.1	1	10	0	exact	0.7503102421760559
40	0.1	1	10	0	approx	0.7502782344818115
40	0.1	1	10	0	hist	0.7502782344818115
40	0.1	1	10	1	auto	0.7502782344818115
40	0.1	1	10	1	exact	0.7503102421760559
40	0.1	1	10	1	approx	0.7502782344818115
40	0.1	1	10	1	hist	0.7502782344818115
40	0.1	3	20	0	auto	0.8945130705833435
40	0.1	3	20	0	exact	0.8941095471382141
40	0.1	3	20	0	approx	0.8945130705833435
40	0.1	3	20	0	hist	0.8945130705833435

40	0.1	3	20	1	auto	0.8945130705833435
40	0.1	3	20	1	exact	0.8941095471382141
40	0.1	3	20	1	approx	0.8945130705833435
40	0.1	3	20	1	hist	0.8945130705833435
40	0.1	3	20	2	auto	0.8945130705833435
40	0.1	3	20	2	exact	0.8941095471382141
40	0.1	3	20	2	approx	0.8945130705833435
40	0.1	3	20	2	hist	0.8945130705833435
40	0.1	5	50	3	auto	0.886320173740387
40	0.1	5	50	3	exact	0.8869958519935608
40	0.1	5	50	3	approx	0.886320173740387
40	0.1	5	50	3	hist	0.886320173740387
100	0.1	1	10	0	auto	0.7909070253372192
100	0.1	1	10	0	exact	0.7909550666809082
100	0.1	1	10	0	approx	0.7909070253372192
100	0.1	1	10	0	hist	0.7909070253372192
100	0.1	1	10	1	auto	0.7909070253372192
100	0.1	1	10	1	exact	0.7909550666809082

100	0.1	1	10	1	approx	0.7909070253372192
100	0.1	1	10	1	hist	0.7909070253372192
100	0.1	3	20	1	auto	0.8905522227287292
100	0.1	3	20	1	exact	0.8899841904640198
100	0.1	3	20	1	approx	0.8905522227287292
100	0.1	3	20	1	hist	0.8905522227287292
100	0.1	3	20	2	auto	0.8905522227287292
100	0.1	3	20	2	exact	0.8899841904640198
100	0.1	3	20	2	approx	0.8905522227287292
100	0.1	3	20	2	hist	0.8905522227287292
100	0.1	5	50	3	auto	0.8821432590484619
100	0.1	5	50	3	exact	0.8829681873321533
100	0.1	5	50	3	approx	0.8821432590484619
100	0.1	5	50	3	hist	0.8821432590484619

Best model n_estimators=40, learning_rate=0.1,max_depth=30 , min_child_weight=20 ,verbosity=0,tree_method=auto.. Gives R2_SCORE Value is 0.8945130705833435
Finalized R2_score value =0.8945130705833435