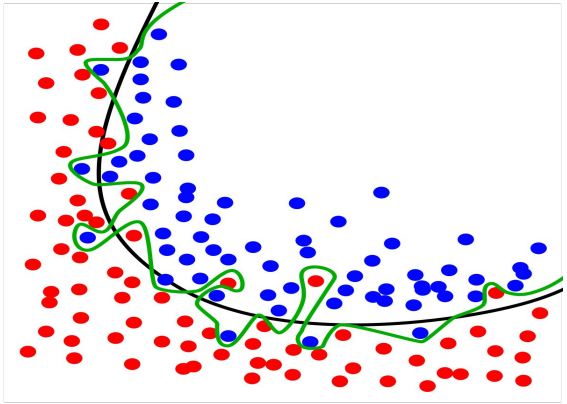
Using Overfitting to evaluate Models

CS550 - Machine Learning and Business Intelligence



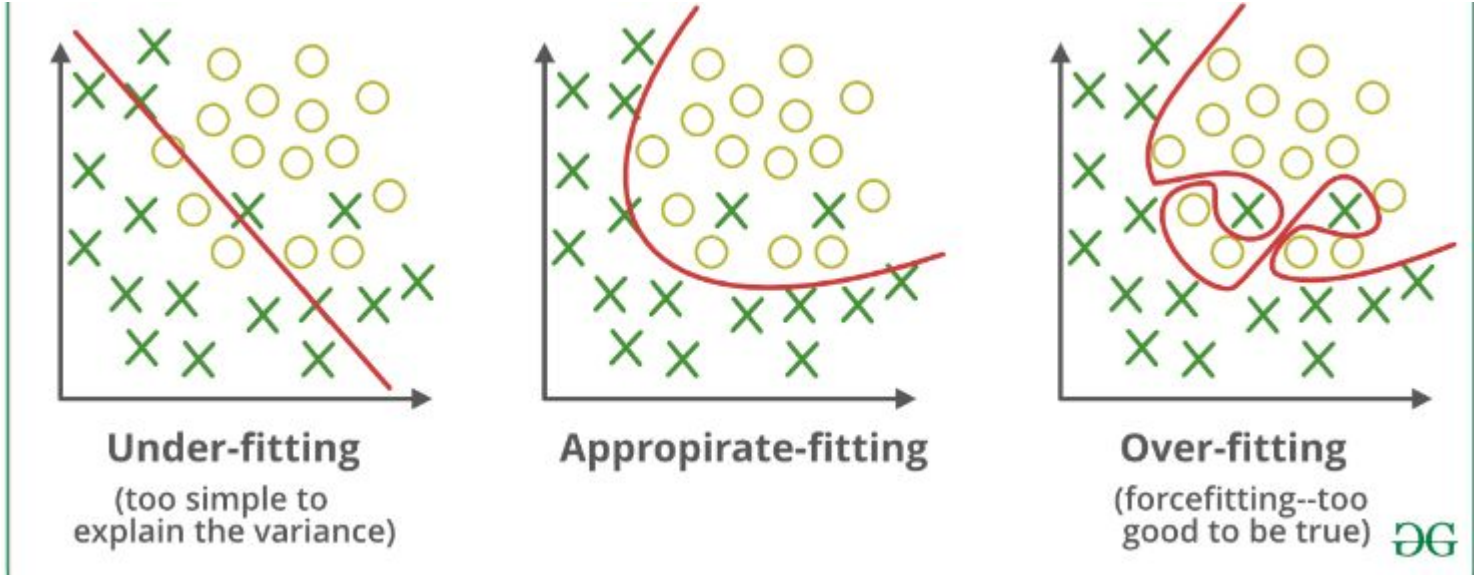
Submitted by: Divya Pandey(19665)
Instructor: Dr. Henry Chang

Table of Content

Introduction

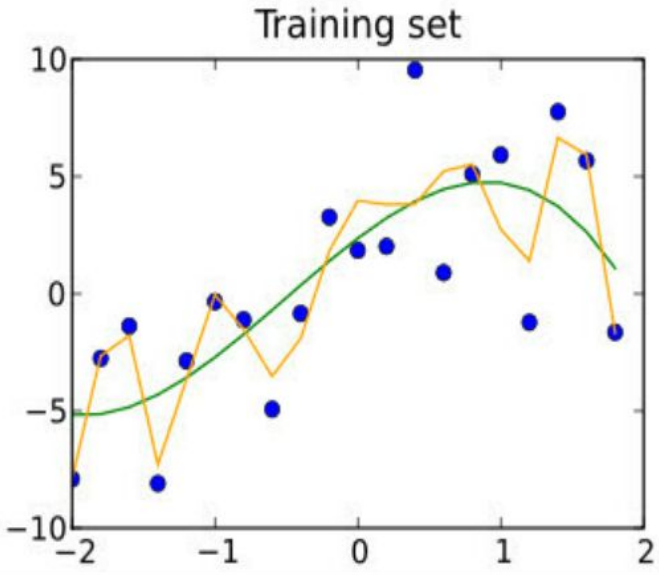
- Overfitting refers to an unwanted behavior of a machine learning algorithm used for predictive modeling.
- We can identify if a machine learning model has overfit by first evaluating the model on the training dataset and then evaluating the same model on a holdout test dataset.
- Overfitting occurs when you achieve a good fit of your model on the training data, while it does not generalize well on new, unseen data
- We can identify overfitting by looking at validation metrics, like loss or accuracy.

Design

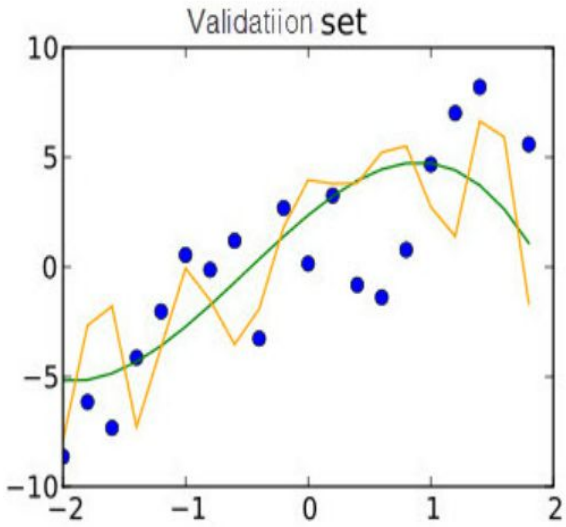

Under-fitting
(too simple to explain the variance)
Appropirate-fitting
Over-fitting
(forcefitting--too good to be true)

Design

Training Set



Validation Set



Implementation-Linear Regression

Linear Regression:

X Value	Y Value	X*Y	X*X
1	1.8	1.8	1
2	2.4	4.8	4
3.3	2.3	7.59	10.89
4.3	3.8	16.34	18.49
5.3	5.3	28.09	28.09
1.4	1.5	2.1	1.96
2.5	2.2	5.5	6.25
2.8	3.8	10.64	7.84
4.1	4.0	16.4	16.81
5.1	5.4	27.54	26.01

Implementation(Linear Regression)

$\Sigma X = 31.8$
$\Sigma Y = 32.5$
$\Sigma XY = 120.8$
$\Sigma X^2 = 121.34$

Slope(b) = $(N\Sigma XY - (\Sigma X)(\Sigma Y)) / (N\Sigma X^2 - (\Sigma X)^2)$ ======>**b1**
=0.86

Intercept(a) = $(\Sigma Y - b(\Sigma X)) / N$ =====>**a1**
== 0.5152

Implementation(Non-Linear Regression)

Non-Linear Regression:

X Value	Y Value	X*Y	X*X
1	1.8	1.8	1
4	2.4	9.6	8
10.89	2.3	25.047	118.5921
18.49	3.8	70.262	341.8801
28.09	5.3	148.877	789.0481
1.96	1.5	2.94	3.8416
6.25	2.2	13.75	39.0625
7.84	3.8	29.792	61.4656
16.81	4.0	67.24	282.5761
26.01	5.4	140.454	676.5201

Implementation(Non-Linear Regression)

$\Sigma X = 121.34$
$\Sigma Y = 32.5$
$\Sigma XY = 509.762$
$\Sigma X^2 = 2321.9862$

Slope(b) = $(N\Sigma XY - (\Sigma X)(\Sigma Y)) / (N\Sigma X^2 - (\Sigma X)^2)$ =====>**b2**
= 0.1345579143

Intercept(a) = $(\Sigma Y - b(\Sigma X)) / N$ =====>**a2**
=1.6168

Implementation(Training Phase)

Training Phase:

x	y	$\hat{y}=a1 + b1 * x$	$\hat{y}=a2 + b2 * x^2$
1	1.8	1.38	1.8
2	2.4	2.24	1.93
3.3	2.3	3.358	2.099
4.3	3.8	4.218	2.229
5.3	5.3	5.078	2.359
1.4	1.5	1.724	1.852
2.5	2.2	2.67	1.995
2.8	3.8	2.928	2.034
4.1	4.0	4.046	2.203
5.1	5.4	4.906	2.333

Implementation(Validation Phase)

Validation Phase:

x	y	**Model 1 $\hat{y}$=a1 + b1 * x**	**Model 2 $\hat{y}$=a2 + b2 * x^2**
1.5	1.7	1.81	1.865
2.9	2.7	3.014	2.047
3.7	2.5	3.702	2.151
4.7	2.8	4.562	2.281
5.1	5.5	4.906	2.333

Implementation(Validation Phase)

MSE for $\hat{y}$=a1 + b1 * x Model 1
=[(1.81-1.7)^2+(3.014-2.7)^2+(3.702-2.5)^2+(4.562-2.8)^2+(4.906-5.5)^2]/5
= 1.002596

MSE for $\hat{y}$=a2 + b2 * x 2 Model 2
=[(1.865-1.7)^2+(2.047-2.7)^2+(2.151-2.5)^2+(2.281-2.8)^2+(2.333-5.5)^2]/5
=2.174937

Model 1 is better model so we are choosing value **$\hat{y}$=a1 + b1 * x** in Test Phase

Test(Test Phase)

Test Phase:

x	$\hat{y}=a1 + b1 * x$
1.4	1.7192
2.5	2.6652
3.6	3.6112
4.5	4.3852
5.4	5.1592

Enhancement Ideas

- We can use Cross Validation Method also , where we iteratively train the algorithm on k-1 folds while using the remaining holdout fold as the test set.

- This method allows us to tune the hyperparameters of the neural network or machine learning model and test it using completely unseen data.

Conclusion

- Overfitting is a possible cause of poor generalization performance of a predictive model.

- Overfitting can be analyzed for machine learning models by varying key model hyperparameters.

- Lower the MSE value better the model.

References

- Brownlee, J. (2020, November 26). How to identify overfitting machine learning models in Scikit-Learn. Retrieved January 28, 2023, from https://machinelearningmastery.com/overfitting-machine-learning-models/

- *Brownlee, J. (2020, November 26). How to identify overfitting machine learning models in Scikit-Learn. Retrieved January 28, 2023, from https://machinelearningmastery.com/overfitting-machine-learning-models/*