

[Ending Soon] BIG INDEPENDENCE SALE - Flat 30% Off +  
a surprise 🎁 on all Master's Program | Use Code:  
FREEDOM

Enroll Now

([https://courses.analyticsvidhya.com/collections?utm\\_source=blog&utm\\_medium=flashstrip](https://courses.analyticsvidhya.com/collections?utm_source=blog&utm_medium=flashstrip))

 LOGIN / REGISTER (<https://id.analyticsvidhya.com/auth/login/?>

NEXT=<https://www.analyticsvidhya.com/blog/2021/04/how-the-gradient-boosting-algorithm-works/>)



(<https://www.analyticsvidhya.com/blog/>)



([https://courses.analyticsvidhya.com/collections/?utm\\_source=blog&utm\\_medium=topbanner](https://courses.analyticsvidhya.com/collections/?utm_source=blog&utm_medium=topbanner))

[ADVANCED \(https://www.analyticsvidhya.com/blog/category/advanced/\)](https://www.analyticsvidhya.com/blog/category/advanced/)

[MACHINE LEARNING \(https://www.analyticsvidhya.com/blog/category/machine-learning/\)](https://www.analyticsvidhya.com/blog/category/machine-learning/)

[MATHS \(https://www.analyticsvidhya.com/blog/category/maths/\)](https://www.analyticsvidhya.com/blog/category/maths/)

[PYTHON \(https://www.analyticsvidhya.com/blog/category/python-2/\)](https://www.analyticsvidhya.com/blog/category/python-2/)

[STRUCTURED DATA \(https://www.analyticsvidhya.com/blog/category/structured-data/\)](https://www.analyticsvidhya.com/blog/category/structured-data/)

[SUPERVISED \(https://www.analyticsvidhya.com/blog/category/supervised/\)](https://www.analyticsvidhya.com/blog/category/supervised/)

## How the Gradient Boosting Algorithm works?

NTARBANI (<https://www.analyticsvidhya.com/blog/author/ntarbani/>), APRIL 19, 2021 [LOGIN TO BOOKMARK THIS ARTICLE \(HT...](#)

We use cookies on Analytics Vidhya websites to deliver our services, analyze web traffic, and improve your experience on the site. By using Analytics Vidhya, you agree to our Privacy Policy

(<https://www.analyticsvidhya.com/privacy-policy/>) and Terms of Use (<https://www.analyticsvidhya.com/terms/>)

This article was published as a part of the [Data Science Blogathon](https://datahack.analyticsvidhya.com/contest/data-science-blogathon-7/)

(<https://datahack.analyticsvidhya.com/contest/data-science-blogathon-7/>). Accept

## Introduction

Gradient boosting algorithm is one of the most powerful algorithms in the field of machine learning. As we know that the errors in machine learning algorithms are broadly classified into two categories i.e. Bias Error and Variance Error. As gradient boosting is one of the boosting algorithms it is used to minimize bias error of the model.

Unlike, Adaboosting algorithm, the base estimator in the gradient boosting algorithm cannot be mentioned by us. The base estimator for the Gradient Boost algorithm is fixed and i.e. *Decision Stump*. Like, AdaBoost, we can tune the `n_estimator` of the gradient boosting algorithm. However, if we do not mention the value of `n_estimator`, the default value of `n_estimator` for this algorithm is 100.

Gradient boosting algorithm can be used for predicting not only continuous target variable (as a Regressor) but also categorical target variable (as a Classifier). When it is used as a regressor, the cost function is Mean Square Error (MSE) and when it is used as a classifier then the cost function is Log loss.

### This article is going to cover the following topics related to Gradient Boosting Algorithm:

- 1) Manual Example for understanding the algorithm
- 2) Python Code for the same example with different estimators.
- 3) Finding the best estimators using GridSearchCV.
- 4) Applications
- 5) Conclusion.

### 1) Manual Example for understanding the algorithm:

Let us now understand the working of the Gradient Boosting Algorithm with the help of one example. In the following example, Age is the Target variable whereas Likes, Exercising, Gender, and Vehicle are independent variables. As in this example, the target variable is continuous, Gradient Boosting Regression is used here. (https://www.analyticsvidhya.com/privacy-policy/) and Terms of Use (https://www.analyticsvidhya.com/terms/).

Accept

LikesExercising	GotoGym	DrivesCar	Age
False	True	True	14
False	True	False	15
False	True	False	16
True	True	True	26
False	True	True	36
True	False	False	50
True	True	True	69
True	False	False	72
True	False	True	73

### 1<sup>st</sup>-Estimator

For estimator-1, the root level (level 0) will consist of all the records. The predicted age at this level is equal to the mean of the entire Age column i.e. 41.33 (addition of all values in Age column divided by a number of records i.e. 9). Let us find out what is the MSE for this level. MSE is calculated as the mean of the square of errors. Here error is equal to actual age - predicted age. The predicted age for the particular node is always equal to the mean of age records of that node. So, the MSE of the root node of the 1<sup>st</sup> estimator is calculated as given below.

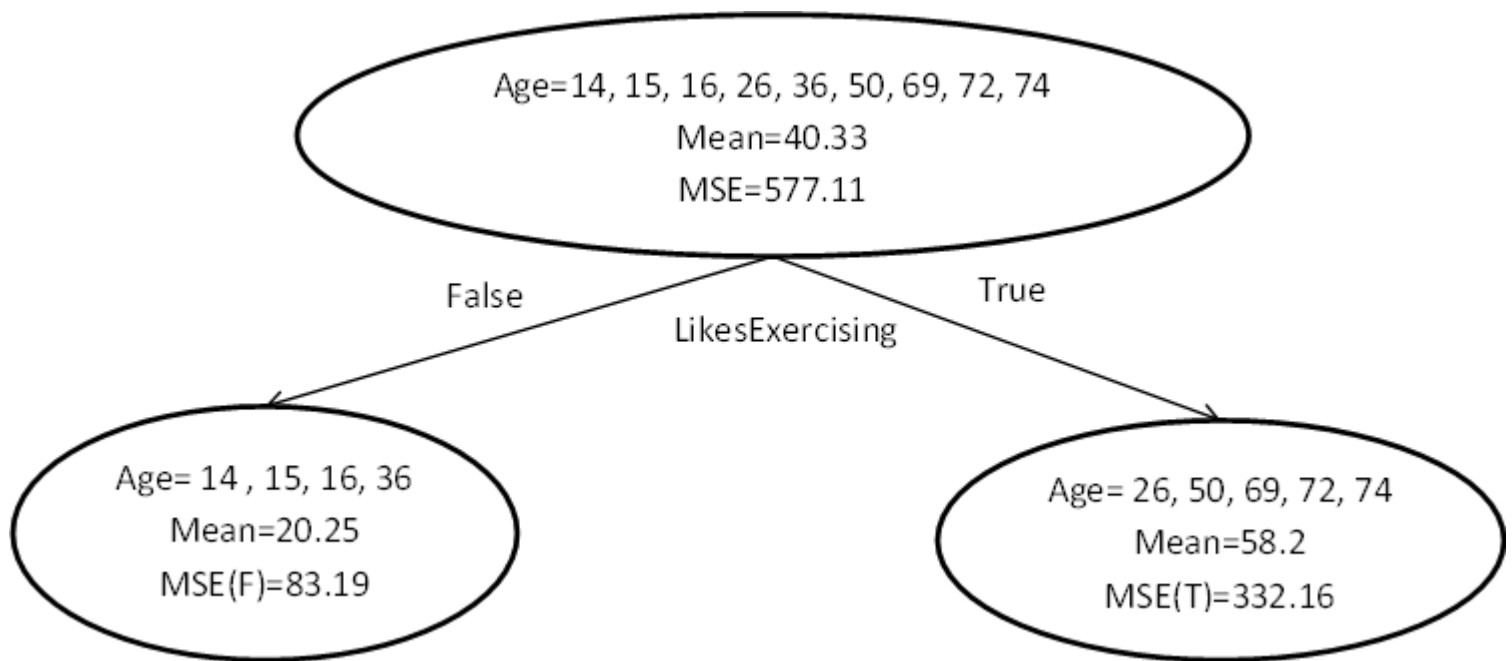
$$\text{MSE} = (\sum (\text{Age}_i - \mu)^2) / 9 = 577.11$$

The cost function here is MSE and the objective of the algorithm here is to minimize the MSE.

Now, one of the independent variables will be used by the Gradient boosting to create the Decision Stump. Let us suppose that the LikesExercising is used here for prediction. So, the records with False LikesExercising will go in one child node, and records with True LikesExercising will go in another child node as shown below.

We use cookies on Analytics Vidhya websites to deliver our services, analyze web traffic, and improve your experience on the site. By using Analytics Vidhya, you agree to our Privacy Policy (<https://www.analyticsvidhya.com/privacy-policy/>) and Terms of Use (<https://www.analyticsvidhya.com/terms/>).

Accept



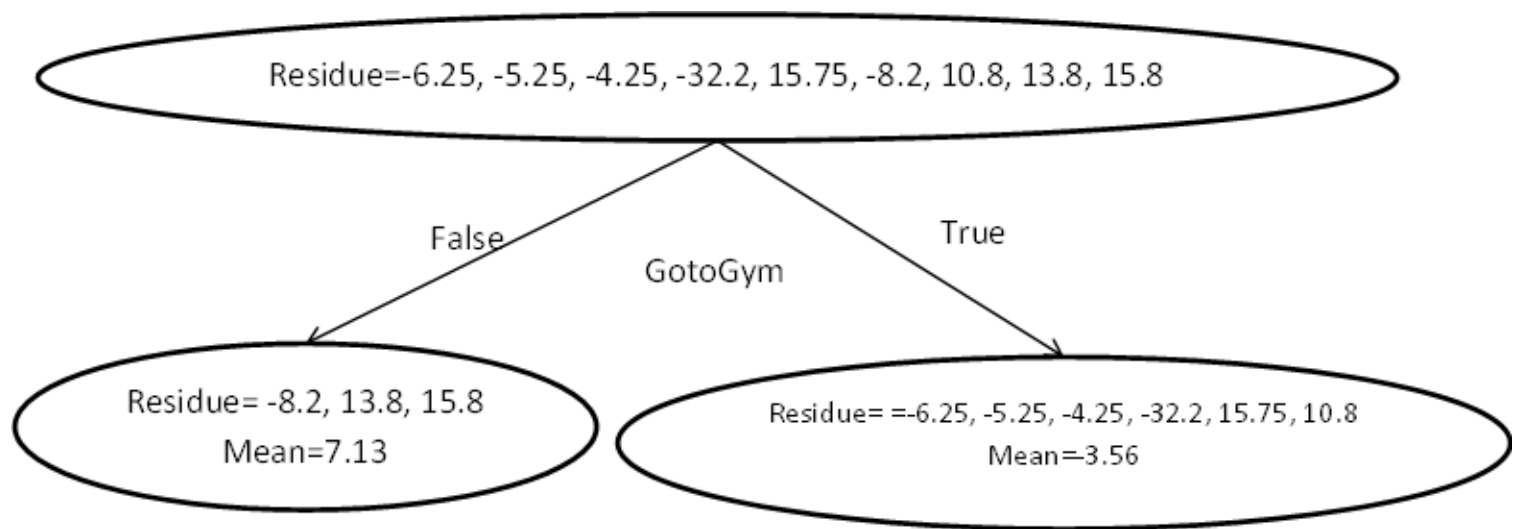
Let us find the means and MSEs of both the nodes of level 1. For the left node, the mean is equal to 20.25 and MSE is equal to 83.19. Whereas, for the right node, the mean is equal to 58.2 and MSE is equal to 332.16. Total MSE for level 1 will be equal to the addition of all nodes of level 1 i.e.  $83.19 + 332.16 = 415.35$ . We can see here, the cost function i.e. MSE of level 1 is better than level 0.

## 2<sup>nd</sup>-Estimator:

Let us now find out the estimator-2. Unlike AdaBoost, in the Gradient boosting algorithm, residues ( $\text{age}_i - \mu$ ) of the first estimator are taken as root nodes as shown below. Let us suppose for this estimator another dependent variable is used for prediction. So, the records with False GotoGym will go in one child node, and records with True GotoGym will go in another child node as shown below.

We use cookies on Analytics Vidhya websites to deliver our services, analyze web traffic, and improve your experience on the site. By using Analytics Vidhya, you agree to our Privacy Policy (<https://www.analyticsvidhya.com/privacy-policy/>) and Terms of Use (<https://www.analyticsvidhya.com/terms/>).

Accept



The prediction of age here is slightly tricky. First, the age will be predicted from estimator 1 as per the value of LikesExercising, and then the mean from the estimator is found out with the help of the value of GotoGym and then that means is added to age-predicted from the first estimator and that is the final prediction of Gradient boosting with two estimators.

Let us consider if we want to predict the age for the following records:

LikesExercising	GotoGym	DrivesCar
False	True	True

Here, LikesExercising is equal to False. So, the predicted age from the first estimator will be 20.25 (i.e. mean of the left node of the first estimator). Now we need to check what is the value of GotoGym for the second predictor and its value is True. So, the mean of True GotoGym in the 2<sup>nd</sup> estimator is -3.56. This will be added to the prediction of the first estimator i.e. 20.25. So final prediction of this model will be  $20.25 + (-3.56) = 16.69$ .

Let us predict of ages of all records we have in the example.

We use cookies on Analytics Vidhya websites to deliver our services, analyze web traffic, and improve your experience on the site. By using Analytics Vidhya, you agree to our Privacy Policy (<https://www.analyticsvidhya.com/privacy-policy/>) and Terms of Use (<https://www.analyticsvidhya.com/terms/>).

Accept

LikesExercising	GotoGym	DrivesCar	Age	Prediction from 1 <sup>st</sup> Estimator	Mean from 2 <sup>nd</sup> Estimator	Final Prediction	Final Residues
False	True	True	14	20.25	-3.56	16.69	-2.69
False	True	False	15	20.25	-3.56	16.69	-1.69
False	True	False	16	20.25	-3.56	16.69	-0.69
True	True	True	26	58.2	-3.56	54.64	-28.64
False	True	True	36	20.25	-3.56	16.69	19.31
True	False	False	50	58.2	7.13	65.33	-15.33
True	True	True	69	58.2	-3.56	54.64	14.36
True	False	False	72	58.2	7.13	65.33	6.67
True	False	True	73	58.2	7.13	65.33	7.67

Let us now find out the Final MSE for above all 9 records.

$$\text{MSE} = ((-2.69)^2 + (-1.69)^2 + (-0.69)^2 + (-28.64)^2 + (19.31)^2 + (-15.33)^2 + (14.36)^2 + (6.67)^2 + (7.67)^2) / 9$$

$$= 194.2478$$

So, we can see that the final MSE is much better than the MSE of the root node of the 1<sup>st</sup> Estimator. This is only for 2 estimators. There can be n number of estimators in gradient boosting algorithm.

## 2) Python Code for the same:

Let us now write a python code for the same.

We use cookies on Analytics Vidhya websites to deliver our services, analyze web traffic, and improve your experience on the site. By using Analytics Vidhya, you agree to our Privacy Policy (<https://www.analyticsvidhya.com/privacy-policy/>) and Terms of Use (<https://www.analyticsvidhya.com/terms/>).

Accept

```
# Importing required modules
import numpy as np
import pandas as pd
from sklearn.ensemble import GradientBoostingRegressor
from sklearn.preprocessing import LabelEncoder

# Let us create the Data-Frame for above
X=pd.DataFrame({'LikesExercising':[False,False,False,True,False,True,True,True,True],
                'GotoGym':[True,True,True,True,True,False,True,False,False],
                'DrivesCar':[True,False,False,True,True,False,True,False,True]})
Y=pd.Series(name='Age',data=[14,15,16,26,36,50,69,72,74])

# Let us encode true and false to number value 0 and 1
LE=LabelEncoder()
X['LikesExercising']=LE.fit_transform(X['LikesExercising'])
X['GotoGym']=LE.fit_transform(X['GotoGym'])
X['DrivesCar']=LE.fit_transform(X['DrivesCar'])
```

```
#We will now see the effect of different numbers of estimators on MSE.
```

We use cookies on Analytics Vidhya websites to deliver our services, analyze web traffic, and improve your experience on the site. By using Analytics Vidhya, you agree to our Privacy Policy (<https://www.analyticsvidhya.com/privacy-policy/>) and Terms of Use (<https://www.analyticsvidhya.com/terms/>).

Accept

```
# 1) Let us now use GradientBoostingRegressor with 2 estimators to train the model and to predict the age for the same inputs.
GB=GradientBoostingRegressor(n_estimators=2)
GB.fit(X,Y)
Y_predict=GB.predict(X) #ages predicted by model with 2 estimators
Y_predict
# Output
#Y_predict=[38.23 , 36.425, 36.425, 42.505, 38.23 , 45.07 , 42.505, 45.07 ,47.54]
#Following code is used to find out MSE of prediction with Gradient boosting algorithm having estimator 2.
MSE_2=(sum((Y-Y_predict)**2))/len(Y)
print('MSE for two estimators :',MSE_2)
#Output: MSE for two estimators : 432.482055555546
```

```
# 2) Let us now use GradientBoostingRegressor with 3 estimators to train the model and to predict the age for the same inputs.
GB=GradientBoostingRegressor(n_estimators=3)
GB.fit(X,Y)
Y_predict=GB.predict(X) #ages predicted by model with 3 estimators
Y_predict
# Output
#Y_predict=[36.907, 34.3325, 34.3325, 43.0045, 36.907 , 46.663 , 43.0045, 46.663 , 50.186]
#Following code is used to find out MSE of prediction with Gradient boosting algorithm having estimator 3.
MSE_3=(sum((Y-Y_predict)**2))/len(Y)
print('MSE for three estimators :',MSE_3)
#Output: MSE for three estimators : 380.05602055555556
We use cookies on Analytics Vidhya websites to deliver our services, analyze web traffic, and improve your experience on the site. By using Analytics Vidhya, you agree to our Privacy Policy
(https://www.analyticsvidhya.com/privacy-policy/) and Terms of Use (https://www.analyticsvidhya.com/terms/).
```

Accept



```
# 3) Let us now use GradientBoostingRegressor with 50 estimators to train the model and to predict the age for the same inputs.
GB=GradientBoostingRegressor(n_estimators=50)
GB.fit(X,Y)
Y_predict=GB.predict(X) #ages predicted by model with 50 estimators
Y_predict
# Output
#Y_predict=[25.08417833, 15.63313919, 15.63313919, 47.46821839, 25.08417833, 60.89864242, 47.46821839, 60.89864242, 73.83164334]
#Following code is used to find out MSE of prediction with Gradient boosting algorithm having estimator 50.
MSE_50=(sum((Y-Y_predict)**2))/len(Y)
print('MSE for fifty estimators :',MSE_50)
```

#Output: MSE for fifty estimators: 156.5667260994211

### **Observation:**

As we can see here, MSE reduces as we increase the estimator value. The situation comes where MSE becomes saturated which means even if we increase the estimator value there will be no significant decrease in MSE.

### **3) Finding the best estimator with GridSearchCV:**

Let us now see, how GridSearchCV can be used to find the best estimator for the above example.

We use cookies on Analytics Vidhya websites to deliver our services, analyze web traffic, and improve your experience on the site. By using Analytics Vidhya, you agree to our Privacy Policy (<https://www.analyticsvidhya.com/privacy-policy/>) and Terms of Use (<https://www.analyticsvidhya.com/terms/>).

Accept

```

from sklearn.model_selection import GridSearchCV
model=GradientBoostingRegressor()
params={'n_estimators':range(1,200)}
grid=GridSearchCV(estimator=model,cv=2,param_grid=params,scoring='neg_mean_squared_error')
grid.fit(X,Y)
print("The best estimator returned by GridSearch CV is:",grid.best_estimator_)
#Output
#The best estimator returned by GridSearch CV is: GradientBoostingRegressor(n_estimators=19)
GB=grid.best_estimator_
GB.fit(X,Y)
Y_predict=GB.predict(X)
Y_predict
#output:
#Y_predict=[27.20639114, 18.98970027, 18.98970027, 46.66697477, 27.20639114,58.34332496, 46.666974
77, 58.34332496, 69.58721772]
MSE_best=(sum((Y-Y_predict)**2))/len(Y)
print('MSE for best estimators :',MSE_best)
#Following code is used to find out MSE of prediction for Gradient boosting algorithm with be
st estimator value given by GridSearchCV
#Output: MSE for best estimators : 164.2298548605391

```

### **Observation:**

You may think that MSE for n\_estimator=50 is better than MSE for n\_estimator=19. Still GridSearchCV returns 19 not 50. Actually, we can observe here is until 19 with each increment in estimator value the reduction in MSE was significant, but after 19 there is no significant decrease in MSE with increment in estimators. So, n\_estimator=19 was returned by GridSearchSV.

We use cookies on Analytics Vidhya websites to deliver our services, analyze web traffic, and improve your experience on the site. By using Analytics Vidhya, you agree to our Privacy Policy

**4) Applications:** <https://www.analyticsvidhya.com/privacy-policy/> and Terms of Use (<https://www.analyticsvidhya.com/terms/>)

Accept

i) Gradient Boosting Algorithm is generally used when we want to decrease the Bias error.

ii) Gradient Boosting Algorithm can be used in regression as well as classification problems. In regression problems, the cost function is MSE whereas, in classification problems, the cost function is Log-Loss.

## 5) Conclusion:

In this article, I have tried to explain how Gradient Boosting Actually works with the help of a simple example.

Gradient Boosting Algorithm is generally used when we want to decrease the Bias error.

Here, the example of GradientBoostingRegressor is shown. GradientBoostingClassifier is also there which is used for Classification problems. Here, in Regressor MSE is used as cost function there in classification Log-Loss is used as cost function.

The most important thing in this algorithm is to find the best value of `n_estimators`. In this article, we have seen how GridSearchCV can be used to achieve that.

After reading this article, you should first try to use this algorithm for practice problems and after that, you must learn how to tune the hyper-parameters for this algorithm.

*The media shown in this article are not owned by Analytics Vidhya and is used at the Author's discretion.*

You can also read this article on our Mobile APP



([https://play.google.com/store/apps/details?id=com.analyticsvidhya.android&utm\\_source=blog\\_article&utm\\_campaign=blog&pcampaignid=MKT-Other-global-all-co-prtnr-py-PartBadge-Mar2515-1](https://play.google.com/store/apps/details?id=com.analyticsvidhya.android&utm_source=blog_article&utm_campaign=blog&pcampaignid=MKT-Other-global-all-co-prtnr-py-PartBadge-Mar2515-1)).



(<https://apps.apple.com/us/app/analytics-vidhya/id1470025572>).

We use cookies on Analytics Vidhya websites to deliver our services, analyze web traffic, and improve your experience on the site. By using Analytics Vidhya, you agree to our Privacy Policy (<https://www.analyticsvidhya.com/privacy-policy/>) and Terms of Use (<https://www.analyticsvidhya.com/terms/>).

Accept

**TAGS :** [BLOGATHON \(https://www.analyticsvidhya.com/blog/tag/blogathon/\)](https://www.analyticsvidhya.com/blog/tag/blogathon/), [GRADIENT BOOSTING](#)

[\(HTTPS://WWW.ANALYTICSVIDHYA.COM/BLOG/TAG/GRADIENT-BOOSTING/\)](https://www.analyticsvidhya.com/blog/tag/gradient-boosting/), GRADIENT BOOSTING MACHINE

[\(HTTPS://WWW.ANALYTICSVIDHYA.COM/BLOG/TAG/GRADIENT-BOOSTING-MACHINE/\)](https://www.analyticsvidhya.com/blog/tag/gradient-boosting-machine/).

NEXT ARTICLE

### **Guide For Feature Extraction Techniques**

[\(https://www.analyticsvidhya.com/blog/2021/04/guide-for-feature-extraction-techniques/\)](https://www.analyticsvidhya.com/blog/2021/04/guide-for-feature-extraction-techniques/)

...

PREVIOUS ARTICLE

### **Machine Learning Basics For Data Science Enthusiasts**

[\(https://www.analyticsvidhya.com/blog/2021/04/machine-learning-basics-for-data-science-enthusiasts/\)](https://www.analyticsvidhya.com/blog/2021/04/machine-learning-basics-for-data-science-enthusiasts/)

[\(https://www.analyticsvidhya.com/blog/author/ntarbani/\)](https://www.analyticsvidhya.com/blog/author/ntarbani/)

[Ntarbani \(https://www.analyticsvidhya.com/blog/author/ntarbani/\)](https://www.analyticsvidhya.com/blog/author/ntarbani/)

---

## **LEAVE A REPLY**

Your email address will not be published.

We use cookies on Analytics Vidhya websites to deliver our services, analyze web traffic, and improve your experience on the site. By using Analytics Vidhya, you agree to our Privacy Policy [\(https://www.analyticsvidhya.com/privacy-policy/\)](https://www.analyticsvidhya.com/privacy-policy/) and Terms of Use [\(https://www.analyticsvidhya.com/terms/\)](https://www.analyticsvidhya.com/terms/).

Accept

Comment

Name (required)

Email (required)

Website

**SUBMIT COMMENT**



Find Data Courses

We use cookies on Analytics Vidhya websites to deliver our services, analyze web traffic, and improve your experience on the site. By using Analytics Vidhya, you agree to our Privacy Policy (<https://www.analyticsvidhya.com/privacy-policy/>) and Terms of Use (<https://www.analyticsvidhya.com/terms/>).

Accept

## POPULAR POSTS

---

40 Questions to test a Data Scientist on Clustering Techniques (Skill test Solution)

(<https://www.analyticsvidhya.com/blog/2017/02/test-data-scientist-clustering/>)

45 Questions to test a data scientist on basics of Deep Learning (along with solution)

(<https://www.analyticsvidhya.com/blog/2017/01/must-know-questions-deep-learning/>)

Top 30 MCQs to Ace Your Data Science Interviews (<https://www.analyticsvidhya.com/blog/2021/04/top-30-mcqs-to-ace-your-data-science-questions-interviews/>)

Commonly used Machine Learning Algorithms (with Python and R Codes)

(<https://www.analyticsvidhya.com/blog/2017/09/common-machine-learning-algorithms/>)

Effective Data Visualization Techniques in Data Science Using Python



(<https://www.analyticsvidhya.com/blog/2021/08/effective-data-visualization-techniques-in-data-science-using-python/>)

3 Interesting Python Projects With Code for Beginners! (<https://www.analyticsvidhya.com/blog/2021/07/3-interesting-python-projects-with-code-for-beginners/>)

COVID-19: A Medical diagnosis using Deep Learning (<https://www.analyticsvidhya.com/blog/2021/08/covid-19-a-medical-diagnosis-using-deep-learning/>)

30 Questions to test a data scientist on Tree Based Models


(<https://www.analyticsvidhya.com/blog/2017/09/30-questions-test-tree-based-models/>)

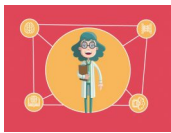
**Free course search to find  
online courses from all the to  
universities and platforms.**

Coursa

We use cookies on Analytics Vidhya websites to deliver our services, analyze web traffic, and improve your experience on the site. By using Analytics Vidhya, you agree to our Privacy Policy

**CAREER RESOURCES** (<https://www.analyticsvidhya.com/privacy-policy/>) and Terms of Use (<https://www.analyticsvidhya.com/terms/>). 

Accept



## 16 Key Questions You Should Answer Before Transitioning into Data Science (<https://www.analyticsvidhya.com/16-key-questions-data-science-career-transition/?>

&utm\_source=Blog&utm\_medium=CareerResourceWidget)

NOVEMBER 23, 2020



## What is A Business Analyst and What is the Role of a business analyst in a Company? (<https://www.analyticsvidhya.com/blog/2021/06/what-is-a-business-analyst-and-what-is-the-role-of-a-business-analyst-in-a-company/?>

&utm\_source=Blog&utm\_medium=CareerResourceWidget)

JUNE 28, 2021



## Data Engineering – Concepts and Importance

(<https://www.analyticsvidhya.com/blog/2021/06/data-engineering-concepts-and-importance/?>

&utm\_source=Blog&utm\_medium=CareerResourceWidget)

JUNE 14, 2021

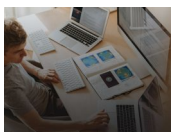


## Here's What You Need to Know to Become a Data Scientist!

(<https://www.analyticsvidhya.com/blog/2021/01/heres-what-you-need-to-know-to-become-a-data-scientist/?>

&utm\_source=Blog&utm\_medium=CareerResourceWidget)

JANUARY 22, 2021



## These 7 Signs Show you have Data Scientist Potential!

(<https://www.analyticsvidhya.com/blog/2020/12/these-7-signs-show-you-have-data-scientist-potential/?>

&utm\_source=Blog&utm\_medium=CareerResourceWidget)

DECEMBER 3, 2020

## RECENT POSTS

### Introducing Panda Gym -Understanding The Reinforcement Learning

(<https://www.analyticsvidhya.com/blog/2021/08/introducing-panda-gym-understanding-the-reinforcement-learning/>)  
We use cookies on Analytics Vidhya website to deliver our services, analyze web traffic, and improve your experience on the site. By using Analytics Vidhya, you agree to our Privacy Policy (<https://www.analyticsvidhya.com/privacy-policy/>) and Terms of Use (<https://www.analyticsvidhya.com/terms/>).  
AUGUST 14, 2021

Accept

## Creating Continuous Action Bot using Deep Reinforcement Learning

(<https://www.analyticsvidhya.com/blog/2021/08/creating-continuous-action-bot-using-deep-reinforcement-learning/>)

AUGUST 14, 2021

## Better EDA with 3 Easy Python Libraries for Any Beginner

(<https://www.analyticsvidhya.com/blog/2021/08/better-eda-with-3-easy-python-libraries-for-any-beginner/>)

AUGUST 14, 2021

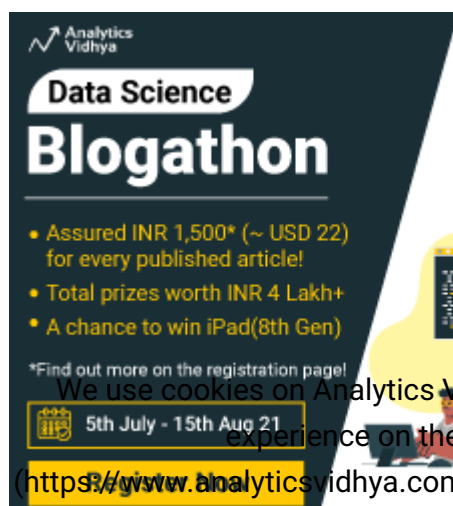
## What is relational about Relational Databases? (<https://www.analyticsvidhya.com/blog/2021/08/what-is-relational-about-relational-databases/>)

AUGUST 14, 2021



(<https://ascendpro.analyticsvidhya.com/?>

utm\_source=blog&utm\_medium=stickybanner1#av-roadmap)



(<https://datahack.analyticsvidhya.com/contest/data-science->

\*Find out more on the registration page!  
We use cookies on Analytics Vidhya websites to deliver our services, analyze web traffic, and improve your experience on the site. By using Analytics Vidhya, you agree to our Privacy Policy (<https://www.analyticsvidhya.com/privacy-policy/>) and Terms of Use (<https://www.analyticsvidhya.com/terms/>).

blogathon-10/?utm\_source=blog&utm\_medium=stickybanner2)



(<https://www.analyticsvidhya.com/>)

## Download App



(<https://play.google.com/store/apps/details?id=com.analyticsvidhya.android>)



(<https://apps.apple.com/us/app/analytics-vidhya/id1470025572>)

## Analytics Vidhya

About Us (<https://www.analyticsvidhya.com/about-me/>)

Our Team (<https://www.analyticsvidhya.com/about-me/team/>)

Careers (<https://www.analyticsvidhya.com/about-me/career-analytics-vidhya/>)

Contact us (<https://www.analyticsvidhya.com/contact/>)

## Data Science

Blog (<https://www.analyticsvidhya.com/blog/>)

Hackathon (<https://datahack.analyticsvidhya.com/>)

Discussions (<https://discuss.analyticsvidhya.com/>)

Apply Jobs (<https://www.analyticsvidhya.com/jobs/>)

## Companies

Post Jobs (<https://www.analyticsvidhya.com/corporate/>)

Trainings (<https://courses.analyticsvidhya.com/>)

Hiring Hackathons (<https://datahack.analyticsvidhya.com/>)

Advertising (<https://www.analyticsvidhya.com/contact/>)

## Visit us

in



(<https://www.facebook.com/analyticsvidhya/>)



(<https://www.youtube.com/channel/UCdteHtH4hg3o2343iObA>)

We use cookies on Analytics Vidhya websites to enhance your navigation, improve site usage, analyze site usage, and assist in our marketing efforts. By using Analytics Vidhya, you agree to our [Privacy Policy](https://www.analyticsvidhya.com/privacy-policy/) and [Terms of Use](https://www.analyticsvidhya.com/terms/).

Accept