Multivariable Regression Questions

Multivariable Regression Overview: → "please refer to notes"

How to find More Complex Boundaries:

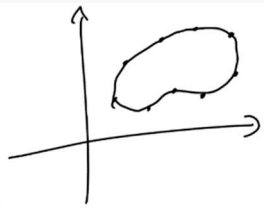
MCQ Questions:

1.

Predicting a Function

Send Feedback

A Prediction function like in the figure below to fit all training data will lead to .....



Options

Attempts left: 1/2

This problem has only one correct answer

- Overfitting model
- Underfitting model
- Balanced model

Hurray! Correct Answer

Solution Description

It will lead to overfitting, as it completely fits our training data, it remembers our training data points, and it will fail for the new dataset.

2.

Figure is an Example Of

Send Feedback

Prediction function as shown in the figure below is an example of:

Options

This problem has only one correct answer

- Balanced
- UnderFit
- OverFit

Hurray! Correct Answer

Solution Description

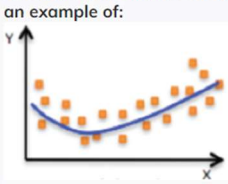
The prediction function as shown in the figure below is an example of underfitting. A better fit for this dataset is a quadratic equation.

3.

Figure is an Example Of

Send Feedback

Prediction function as shown in the figure below is an example of:



Options

This problem has only one correct answer

- Balanced
- UnderFit
- OverFit

Hurray! Correct Answer

Solution Description

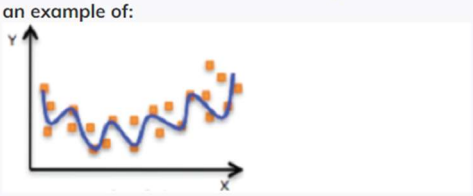
The prediction function as shown in the figure below is an example of balanced model.

4.

Figure is an Example Of

Send Feedback

Prediction function as shown in the figure below is an example of:



Options

This problem has only one correct answer

- UnderFit
- Balanced
- OverFit

Hurray! Correct Answer

Solution Description

The prediction function as shown in the figure below is an example of overfitting.

Add Features to Boston Dataset:

In this problem you need to load the Boston dataset from sklearn toy datasets. After loading, you need to split the dataset into testing and training datasets. Now, fit the linear regression model on the training dataset and store the training and testing scores.

After this add two more columns to the dataset. These columns are the squared values of the 'AGE' and the 'RM' columns respectively.

Again split the dataset into testing and training datasets. Now, fit the linear regression model on the training dataset and store the training and testing scores.

Compare the scores of both the models and print "Score improved" if both training and testing scores improved after squaring two columns. Otherwise, print "Score not improved".

Output

If both training and testing scores improved after squaring two columns print:

"Score improved"

Else print:

"Score not improved"

Source Code:

```
import numpy as np
import pandas as pd
from sklearn.datasets import load_boston
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LinearRegression
from sklearn.metrics import mean_squared_error, r2_score

# Load the Boston dataset
boston = load_boston()
data = pd.DataFrame(boston.data, columns=boston.feature_names)
target = pd.DataFrame(boston.target, columns=["MEDV"])

# Split the dataset into training and testing sets
X_train, X_test, y_train, y_test = train_test_split(data, target, test_size=0.2, random_state=42)

# Fit the linear regression model and store the training and testing scores
lr = LinearRegression()
lr.fit(X_train, y_train)
train_score_before = lr.score(X_train, y_train)
test_score_before = lr.score(X_test, y_test)

# Add squared columns for 'AGE' and 'RM'
X_train['AGE^2'] = X_train['AGE'] ** 2
X_train['RM^2'] = X_train['RM'] ** 2
X_test['AGE^2'] = X_test['AGE'] ** 2
X_test['RM^2'] = X_test['RM'] ** 2

# Fit the linear regression model with squared columns
lr.fit(X_train, y_train)
train_score_after = lr.score(X_train, y_train)
test_score_after = lr.score(X_test, y_test)

# Compare scores and print the result
if train_score_after > train_score_before and test_score_after > test_score_before:
    print("Score improved")
else:
    print("Score not improved")
```

Complexity Analysis of Normal Equation

Gradient Descent:

MCQ Questions:

1.

Gradient Descent Definition

Send Feedback

Gradient Descent is an optimization algorithm used for _____

Options Attempts left: 1/2

This problem has only one correct answer

- minimizing the cost function in various machine learning algorithms
- Certain Changes in algorithm
- maximizing the cost function in various machine learning algorithms
- remaining same the cost function in various machine learning algorithms

Hurray! Correct Answer

Solution Description

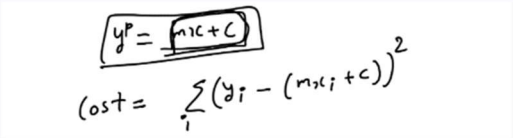
Gradient Descent is an optimization algorithm used for minimizing the cost function in various machine learning algorithms

2.

Cost in Gradient Descent

Send Feedback

Following is the equation to calculate the cost in Linear Regression. In Gradient Descent our task is to:



Options

This problem has only one correct answer

- ○ Find m and c to maximize the cost
- ◉ Find m and c to minimize the cost
- ○ Make the cost zero

✓ Hurray! Correct Answer

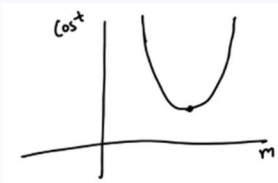
Solution Description

In Gradient Descent our task is to find m and c to minimize the cost.

3.

Cost Function

Send Feedback



In above graph between cost and m, dot on parabola represent:

Options

This problem has only one correct answer

- ○ It is a point where cost is zero
- ◉ It is a point where cost is minimum
- ○ It is a point where cost is maximum

✓ Hurray! Correct Answer

Solution Description

It is the lowest point in the parabola so the cost is minimum.

4.

Cost Function

Send Feedback

Choose all correct statements:

Options

This problem may have one or more correct answers

- ☐ Left tangent on parabola is having +ve slope
- ☐ Right tangent on parabola is having -ve slope
- ☑ Left tangent on parabola is having -ve slope ✓
- ☑ Right tangent on parabola is having +ve slope ✓

✓ Hurray! Correct Answer

Solution Description

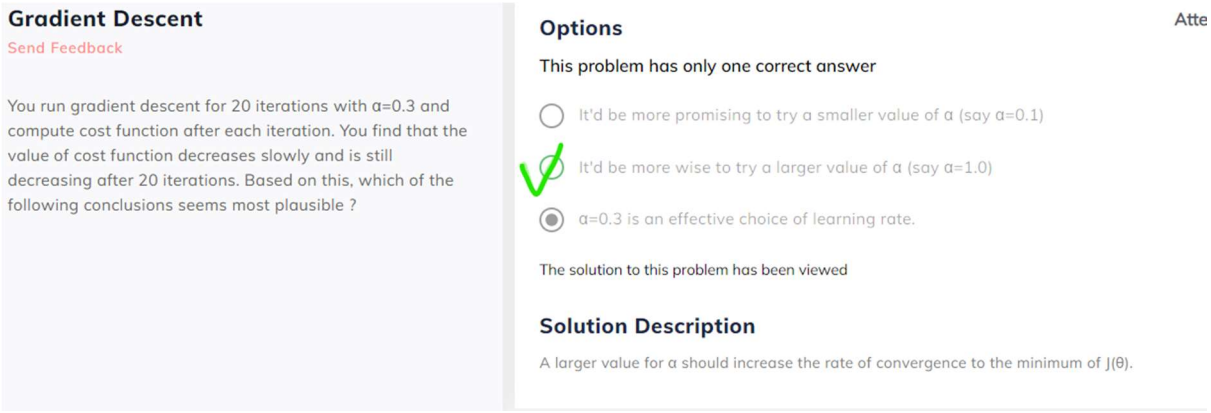
1. Left tangent on the parabola is having the -ve slope.

2. Right tangent on the parabola is having +ve slope.

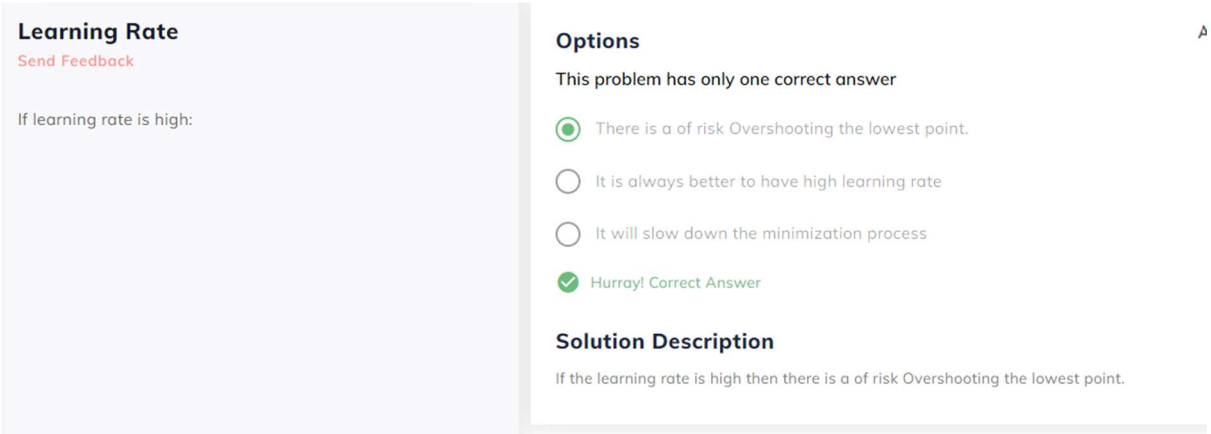
Learning Rate:

MCQ Questions:

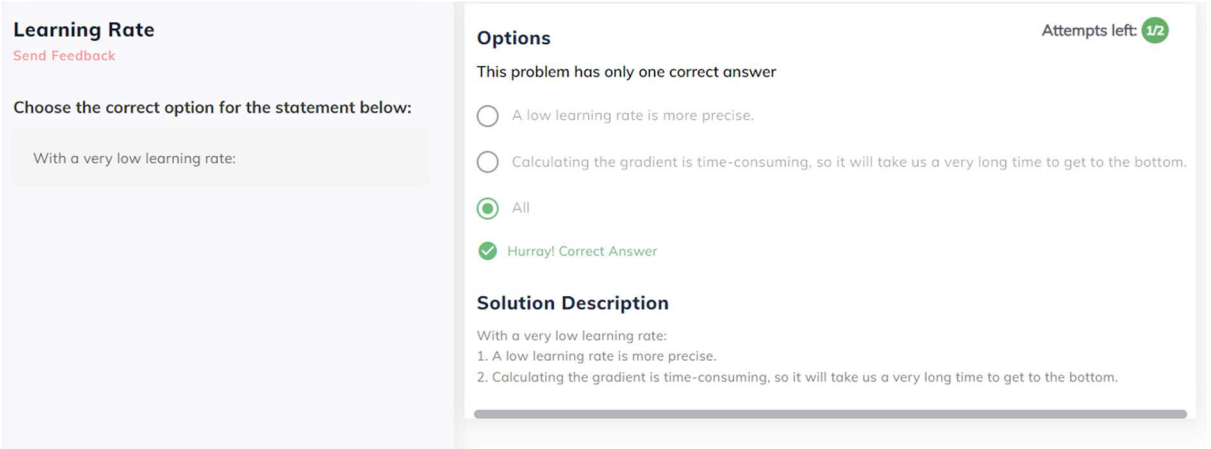
1.



2.



3.

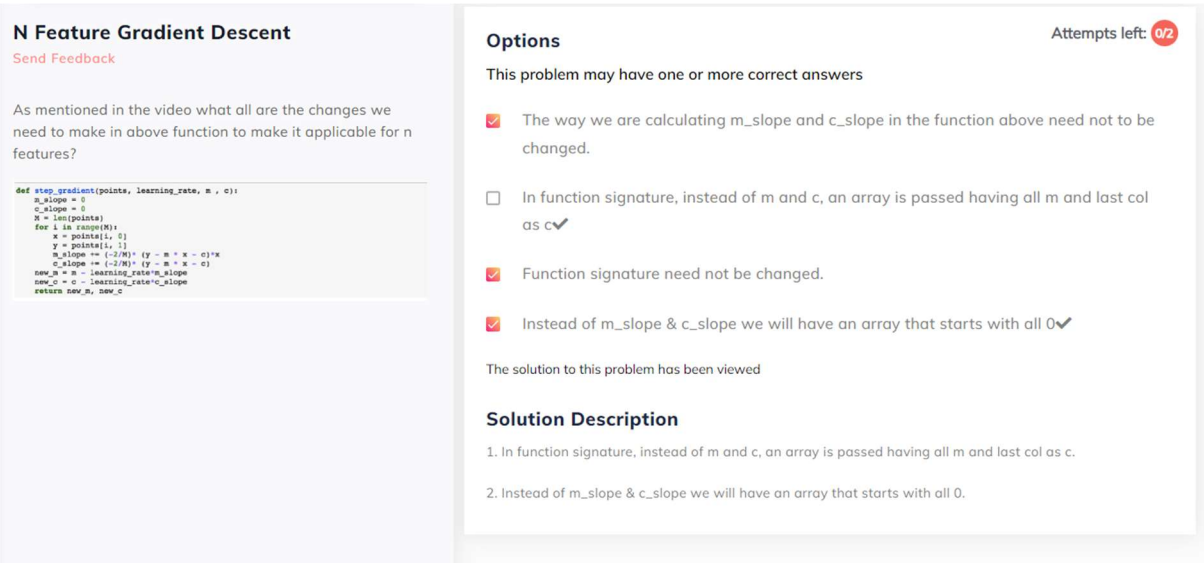


Code Gradient Descent: →"**please refer** to **notes**"

Code Gradient Descent:

MCQ Questions:

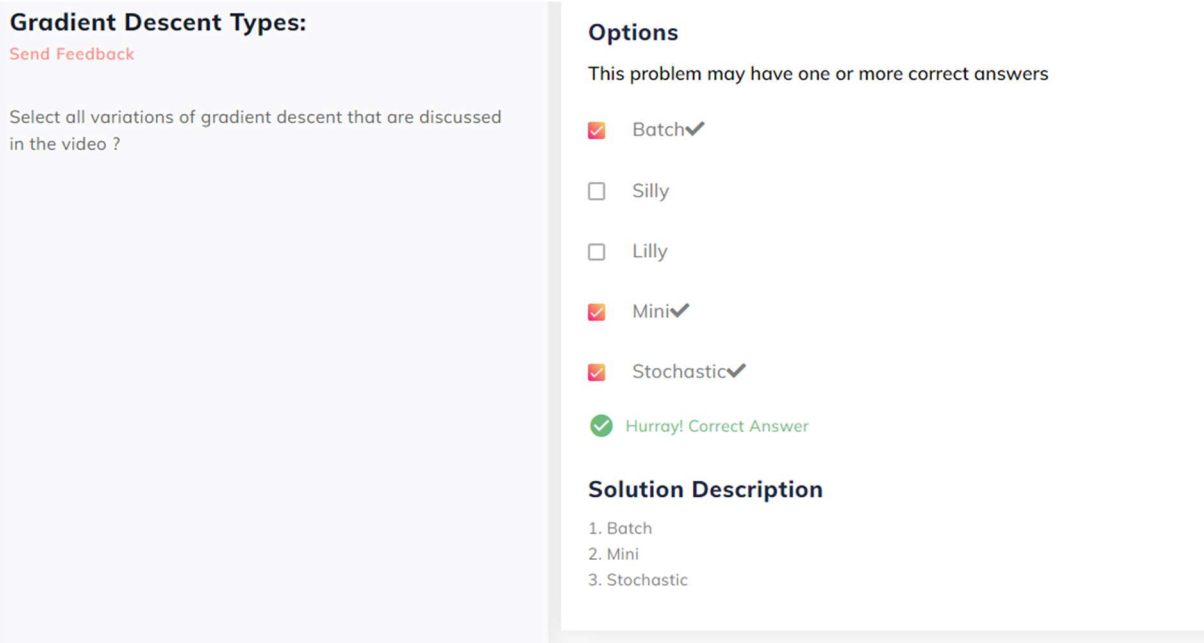
1.



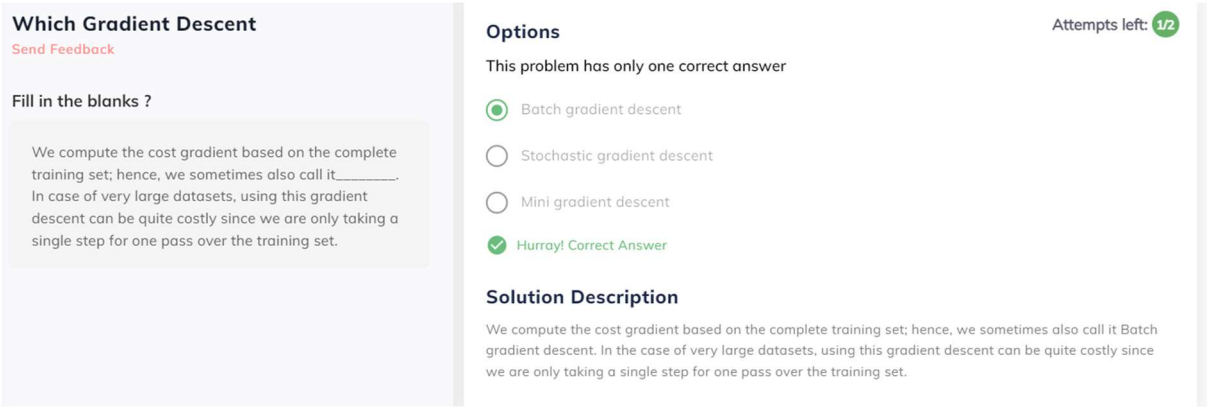
Variations of Gradient Descent

MCQ Questions:

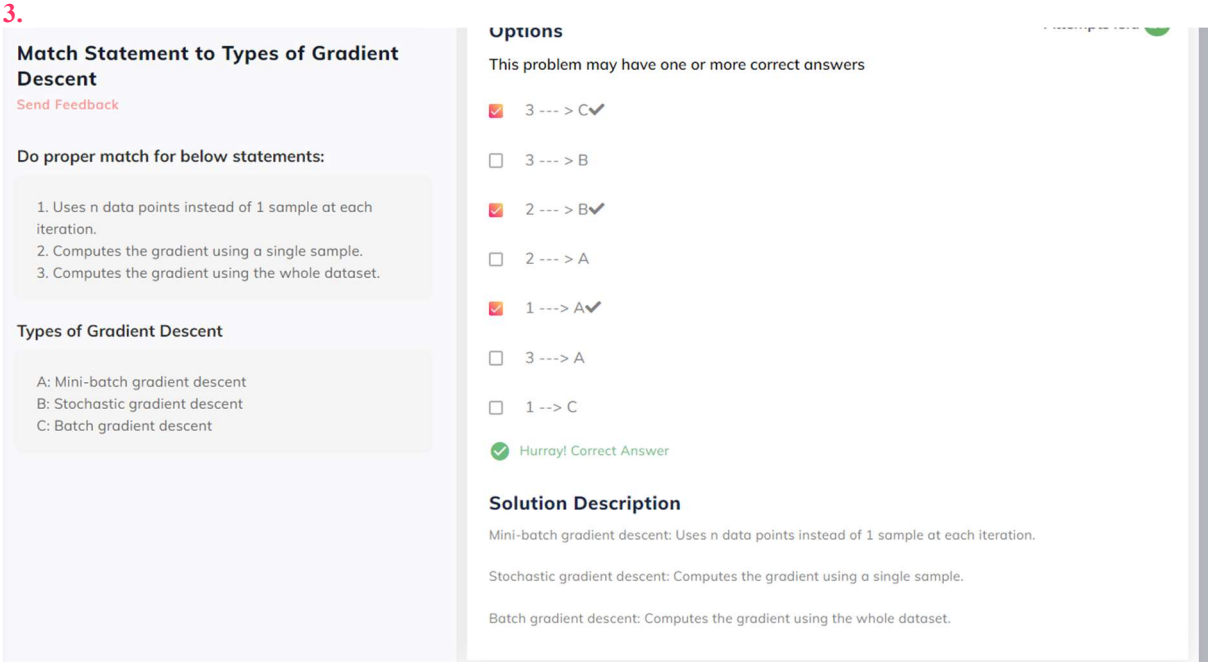
1.



2.



3.



→ Gradient Descent Notes

→ Multivariable Regression And Gradient Descent Notebook