Gradient Descent (BGD) - SLR

Wednesday, 6 March 2024 6:55 pm

→ Gradient descent is a first-order iterative optimization algorithm for finding a local minimum of a differentiable function.

→ The idea is to take repeated steps in the opposite direction of the gradient (derivative) of the function at the current point, b/c this is the direction of steepest descent.

```
from sklearn.datasets import make_regression
import matplotlib.pyplot as plt
import numpy as np
```

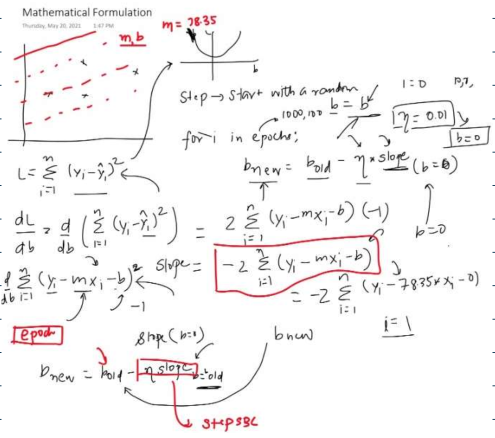
```
X,y = make_regression(n_samples=100, n_features=1, n_informative=1, n_targets=1,noise=20)
```

→ random_state parameter

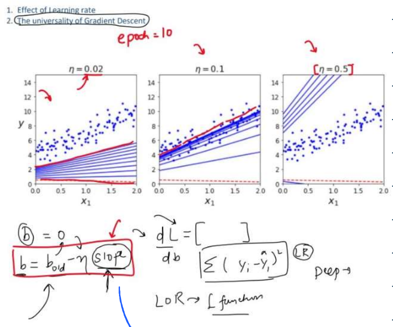
```
plt.scatter(X,y)
```

$$b_{new} = b_{old} - \underbrace{\eta \; slope}_{\text{step size}}\Big|_{b = b\_old}$$

When to stop?
- while diff b/w old & new > 0.0001
- Iteration limit e.g. 100, 1000 (epochs)



→ The further you are, the bigger the step size.

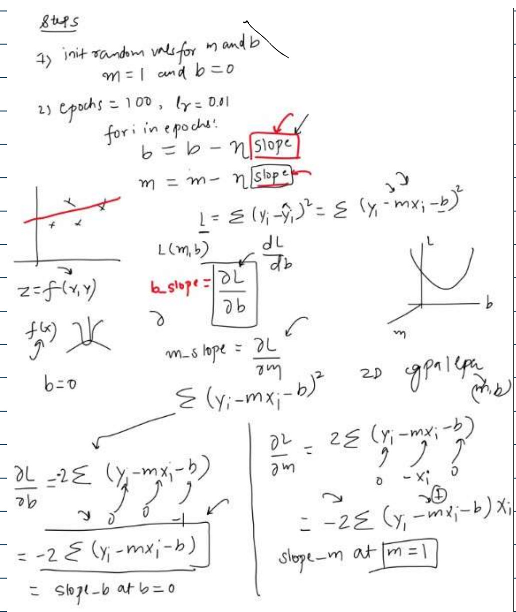


* Learning rate Hyperparameter

* if η is too low, the GD algo will converge towards the right solution VERY SLOWLY (More iterations/ephocs → ↑ calculations → slow algo)

Important graphs:
- → Cost vs parameter AS epochs progress
- → Cost vs epochs → tells us when to stop
- → parameter vs epochs

→ This equation is independent of the ML algo. We just require a differentiable Loss function.

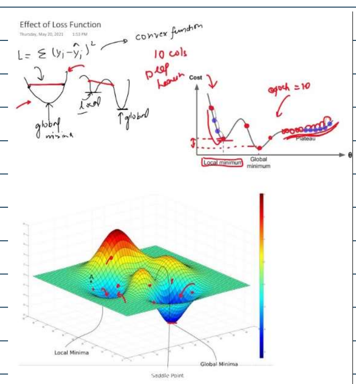


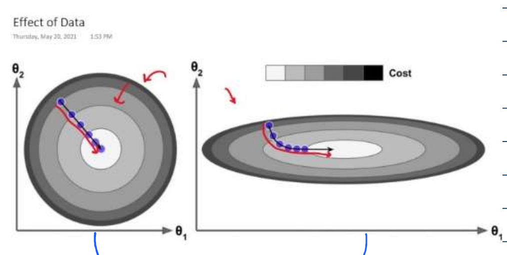
→ Effect of Learning rate:

$\eta \downarrow$ ephocs ↑ to converge to right solution

$\eta \uparrow$ Risky, may never converge to the right solution

optimal η (learning rate) is crucial.





↓ if Features are on same scale

↓ if Features aren't on same scale ↑ time

That's why scaling is important in LR.

* Accuracy is not a challenge of using gradient descent.
The challenges of using gradient descent are convergence, stability & computational complexity.

* **Question 15** Single Correct Option, +3.00, -1.00

Which variant of Gradient Descent is most suitable for online learning scenarios?

A Batch Gradient Descent

B Stochastic Gradient Descent

C Mini-batch Gradient Descent

D Adaptive Gradient Descent

✖ Your Answer is Incorrect

Correct Answers: B

Your Answers: C

In online learning, the model learns and updates its parameters continuously as new data becomes available. SGD is particularly well-suited for this setting because it updates the model's parameters after processing each individual data point. This allows the model to adapt quickly to new data and adjust its predictions accordingly.

Unlike Batch Gradient Descent, which requires processing the entire training dataset before updating the parameters, SGD updates the parameters based on the gradient computed from a single randomly selected data point. This makes SGD computationally efficient and enables it to handle large-scale online learning tasks.

Furthermore, SGD's stochastic nature introduces noise in parameter updates, which can help the model escape local minima and explore different regions of the parameter space. This stochasticity can be beneficial in scenarios where the data is non-stationary or when there are abrupt changes in the underlying distribution.

Therefore, due to its ability to handle continuous learning, computational efficiency, and noise introduction, Stochastic Gradient Descent is the most suitable variant of Gradient Descent for online learning scenarios.

* The learning rate controls how much the model parameters are updated in each iteration of gradient descent. A small learning rate will cause the model to converge slowly, while a large learning rate may cause the model to diverge.

* Which variant of Gradient Descent is most suitable for online learning scenarios?

A Batch Gradient Descent

B Stochastic Gradient Descent

C Mini-batch Gradient Descent

D Adaptive Gradient Descent

✔ Your Answer is Correct

Correct Answers: B

Your Answers: B

In online learning, the model learns and updates its parameters continuously as new data becomes available. SGD is particularly well-suited for this setting because it updates the model's parameters after processing each individual data point. This allows the model to adapt quickly to new data and adjust its predictions accordingly.

Unlike Batch Gradient Descent, which requires processing the entire training dataset before updating the parameters, SGD updates the parameters based on the gradient computed from a single randomly selected data point. This makes SGD computationally efficient and enables it to handle large-scale online learning tasks.

Furthermore, SGD's stochastic nature introduces noise in parameter updates, which can help the model escape local minima and explore different regions of the parameter space. This stochasticity can be beneficial in scenarios where the data is non-stationary or when there are abrupt changes in the underlying distribution.

Therefore, due to its ability to handle continuous learning, computational efficiency, and noise introduction, Stochastic Gradient Descent is the most suitable variant of Gradient Descent for online learning scenarios.