How to implement custom loss function that has more parameters with XGBClassifier in scikit-learn?

Asked 3 years, 4 months ago Modified 2 years, 9 months ago Viewed 4k times

4

I have following problem with implementing custom loss function with scikit-learn:

I would like to implement Focal Loss as my objective function in XGBClassifier. However, I dont know how to pass additional arguments as a parameter(objective parameter):

```
 def focal_loss(y_pred, y_true, alpha=0.25, gamma=1):
  a,g = alpha, gamma
  def fl(x,t):
    p = 1/(1+np.exp(-x))
    return -( a*t + (1-a)*(1-t) ) * (( 1 - ( t*p + (1-t)*(1-p)) )**g) * ( t*np.log(p)+(1-
t)*np.log(1-p) )
  partial_fl = lambda x: fl(x, y_true)
  grad = derivative(partial_fl, y_pred, n=1, dx=1e-6)
  hess = derivative(partial_fl, y_pred, n=2, dx=1e-6)
  return grad, hess

xgb = xgb.XGBClassifier(objective=focal_loss)
```

What should I do in following situation? Is there maybe ready version of Focal Loss ready to use? Thanks in advance.

scikit-learn xgboost

Share Edit Follow Flag

asked May 24, 2020 at 21:44

Avistian
85 1 5

1 Answer

Sorted by: Highest score (default)

1

```
def focal_loss(alpha, gamma):
    def custom_loss(y_pred, y_true):
        a,g = alpha, gamma
        def fl(x,t):
            p = 1/(1+np.exp(-x))
            return -( a*t + (1-a)*(1-t) ) * (( 1 - ( t*p + (1-t)*(1-p)) )**g) * (
t*np.log(p)+(1-t)*np.log(1-p) )
        partial_fl = lambda x: fl(x, y_true)
```

```
            grad = derivative(partial_fl, y_pred, n=1, dx=1e-6)
            hess = derivative(partial_fl, y_pred, n=2, dx=1e-6)
            return grad, hess
        return custom_loss

xgb = xgb.XGBClassifier(objective=focal_loss(alpha=0.25, gamma=1))
```

Using Python Closures!!

Share Edit Follow Flag

answered Dec 6, 2020 at 21:45

Milind Dalvi

156 1 3