Documentation: Model Enhancement for Celestial Body Classification

Original Code:

Source : https://valueml.com/classification-of-celestial-bodies-using-cnn-in-python/

1. **Model Architecture**:
    - Convolutional layers: 3 (32, 64, and 64 filters respectively)
    - Pooling layers: 3 (MaxPooling)
    - Fully Connected layers: 2 (128 units followed by 1 unit with a sigmoid activation)
2. **Training Configuration**:
    - Epochs: 25
    - Batch Size: 25
    - Optimizer: Adam
    - Loss Function: Binary Crossentropy
3. **Results**:
    - Training Loss: 0.0316
    - Training Accuracy: 98.56%
    - Validation Loss: 0.0341
    - Validation Accuracy: 98.40%

Modified Code:

1. **Model Architecture Enhancements**:
    - Added Dropout layers after each Conv2D and Dense layers to reduce overfitting:

```
my_model.add(layers.Dropout(0.2)) # Added after the first Conv2D layer
my_model.add(layers.Dropout(0.3)) # Added after the second Conv2D layer
my_model.add(layers.Dropout(0.3)) # Added after the third Conv2D layer
my_model.add(layers.Dropout(0.5)) # Added before the final Dense layer
```

2. **Training Configuration Updates**:
    - Increased Epochs to 50 for extended training.
    - Adjusted the Batch Size to 32 based on the data set size.

      Implemented Learning Rate Scheduler (**ReduceLROnPlateau**) to adaptively adjust the

```
lr_schedule = callbacks.ReduceLROnPlateau(monitor='val_loss', factor=0.2, patience=5, min_lr=0.001)
```

3. **Results**:
    - Training Loss: 0.0286 (Reduced from 0.0316)
    - Training Accuracy: 99.21% (Improved from 98.56%)
    - Validation Loss: 0.0344 (Slightly increased from 0.0341)
    - Validation Accuracy: 98.44% (Slightly improved from 98.40%)

Comparison:

- **Training Loss**: Reduced from 0.0316 to 0.0286.
- **Training Accuracy**: Improved from 98.56% to 99.21%.
- **Validation Loss**: Slightly increased from 0.0341 to 0.0344.
- **Validation Accuracy**: Slightly improved from 98.40% to 98.44%.

These modifications were aimed at enhancing the model's performance by incorporating dropout layers for regularization, increasing the number of epochs, adjusting the batch size, and implementing a learning rate scheduler. The results indicate a slight improvement in the model's training accuracy, while the validation accuracy remained relatively consistent, suggesting a more generalized performance.