Report and summary of the project Data Augmentation for cnn models

This report summarizes the implementation and results of applying data augmentation techniques to a Convolutional Neural Network (CNN) for image classification. The objective was to improve model generalization and accuracy by artificially expanding the training dataset.

---

1. Augmentation Techniques Used

The augmentation pipeline was implemented using TensorFlow/Keras's ImageDataGenerator class. The following transformations were applied to the training images:

- **Rotation:** rotation_range=40 - Randomly rotated images by up to 40 degrees. This helps the model recognize objects regardless of their orientation.
- **Horizontal Flip:** horizontal_flip=True - Randomly flipped images horizontally. This is effective for objects where a horizontal flip doesn't change their classification (e.g., a cat is still a cat when flipped).
- **Zoom:** zoom_range=0.2 - Randomly zoomed into images by up to 20%. This forces the model to learn features at different scales.
- **Shifts:**
  - width_shift_range=0.2
  - height_shift_range=0.2 - Randomly shifted images horizontally and vertically by up to 20% of their total width and height. This makes the model less sensitive to the precise position of an object within the frame.
- **Brightness Adjustment:** brightness_range=[0.5, 1.5] - Randomly adjusted the brightness of images. This helps the model generalize to different lighting conditions.

---

2. Results and Insights

Two identical CNN models were trained on the Cats vs. Dogs dataset: a **Baseline Model** (without augmentation) and an **Augmented Model** (with the transformations listed above). The final performance was compared based on their validation accuracy and loss curves.

Model	Final Validation Accuracy
Baseline Model	0.689
Augmented Model	0.7

Insights:

- **Overfitting Reduction:** The baseline model exhibited classic signs of **overfitting**. Its training accuracy reached nearly 100%, indicating it had memorized the training data. However, its validation accuracy plateaued at a lower value, and the gap between training and validation accuracy grew significantly.

- **Improved Generalization:** The augmented model showed a different training pattern. Its training accuracy was lower and more volatile, as the model was forced to learn more robust features from the constantly changing images. This led to a higher final **validation accuracy** and a smaller gap between training and validation performance. The model generalized better to new, unseen data because it was trained on a more diverse dataset.
- **Conclusion:** The data augmentation techniques successfully improved the model's ability to generalize, leading to a small but meaningful **increase in validation accuracy**. The results demonstrate that augmenting the training data is an effective strategy for creating more robust and accurate CNNs, especially when the initial dataset is limited.