**SUMMARY**

1) You used two hidden layers. Try using one or three hidden layers and see how doing so affects validation and test accuracy

The training process seems to be overfitting. While the model keeps learning the training data (increasing training accuracy and decreasing training loss), its ability to generalize to unseen data suffers (validation accuracy drops and validation loss increases). Adding more hidden layers doesn't improve performance, suggesting the model might be too complex for the task.

2)Try using layers with more hidden units or fewer hidden units: 32 units, 64 units, and so on.

While training the model seemed effective (training loss decreased), its performance on unseen data worsened (validation accuracy peaked then dropped). Adding more complexity (more nodes) hurt performance, suggesting the model might be memorizing training data instead of learning general patterns (overfitting).

3)Try using the mse loss function instead of binary\_crossentropy

Switching from binary\_crossentropy to MSE loss function initially seemed promising: training and validation losses behaved similarly until a sudden divergence after two epochs. However, this change came at a cost - validation accuracy started dropping after the fourth epoch, suggesting potential issues with the model's ability to generalize.

4)Try using the tanh activation (an activation that was popular in the early days of neural networks) instead of relu

Despite rising training accuracy, the model struggled to generalize: validation accuracy peaked in the second epoch and then dropped. Interestingly, using ReLu activation resulted in both higher validation loss increases and greater fluctuations in validation accuracy compared to Tanh, suggesting ReLu might be less suitable for this specific scenario.

5)Use any technique we studied in class, and these include regularization, dropout, etc., to get your model to perform better on validation

While the model seemed to learn well from the training data (steadily increasing training accuracy), its ability to generalize to unseen data was limited. Validation accuracy peaked after 8 epochs and then almost dropped back down. Interestingly, using the dropout technique significantly improved accuracy without impacting validation accuracy much, suggesting it helped prevent overfitting and improve the model's ability to handle new data.