1. **Discussion**:
   1. **Interpretation of Results and their significance**

The following section outlines various methods utilized in our project, providing an interpretation of results along with an explanation of their significance for each method.

* **Logistic Regression**:

The results of our Logistic Regression suggest that the model has moderate predictive performance. The balance between precision and recall, as indicated by the F1 score, suggests that the model is good at identifying the states but may not be highly precise or sensitive.

* **SVM one-versus-one:**

In comparison to the Logistic Regression, the SVM Classifier with a one-vs-one strategy appears to perform better across all metrics, with higher accuracy, precision, recall, and F1 score. These results suggest that the SVM model is more effective in predicting the states, demonstrating a higher overall performance.

* **SVM one-versus-all**

Our SVM Model demonstrates excellent performance with our dataset, utilizing both one-vs-all and one-vs-one strategies. This robust performance may indicate the presence of outliers in the data, which SVMs are known to be more resistant to. Furthermore, the consistent and strong predictive performance across accuracy, precision, recall, and F1 score metrics for both one-vs-all and one-vs-one strategies suggests that the choice of strategy does not significantly impact the SVM Classifier's effectiveness in this case. This observation is indicative of a well-balanced dataset that includes examples for all classes and features that contain the necessary information for constructing a reliable decision boundary.

* **Naive Bayes Classifier**

Our Naive Bayes Model performs the worst so far, with it not being able to cross the 50% threshold. This might indicate that the datapoints are not conditionally independent, leading to many wrong predictions by the model. The Naive Bayes model appears to have a lower overall performance compared to the SVM model discussed earlier. The lower accuracy, precision, recall, and F1 score indicate that the Naive Bayes model might not be well-suited for the task or that the data may not align well with the assumptions of the Naive Bayes algorithm.

* **Custom Neural Network**

For our custom neural network, we can see the highest scores of any model trained so far. This high accuracy shows that most of its predictions are correct, signifying its overall performance. The high precision and recall show that the model is great at minimizing false positives while predicting positive instances of each class. The high F1 score also confirms that the model has a good balance of its precision and recall scores. The training scores being remarkably close to actual test scores also signify that the model has not been overfitted to the training data even with the substantial number of epochs.

* 1. **limitations and implications**

In the forthcoming section, we delve into an in-depth exploration of the respective limitations and implications associated with each approach.

* **Logistic Regression**:

1. Data Imbalance:

- The metrics are influenced by an unequal distribution of classes, potentially leading to bias and reduced accuracy for minority classes.

2. Generalization Challenges:

- The model's ability to generalize to unseen data, crucial for real-world applications, is not guaranteed.

* **SVM one-versus-one:**

1. Dependence on Data Characteristics:

- The observed superior performance of SVM with a one-vs-one strategy is contingent on the specific characteristics of the dataset used. Its effectiveness may vary with different data distributions.

2. Sensitivity to Hyperparameters:

- Our reported results are based on the specific hyperparameters used during training. The model's performance can change based on different HP.

3. Computational Complexity:

- SVM, especially with one-versus-one classification, tends to have higher computational requirements compared to some other algorithms. This could limit its scalability for large datasets or real-time applications.

* **SVM one-versus-all:**

1. Generalization to Outliers:

- While SVMs are known for their resilience to outliers, the assumption of outlier resistance might not be held universally. The model's performance in the presence of extreme outliers can be lower.

2. Dependency on Balanced Dataset:

- The inference of a well-balanced dataset contributing to the model's success implies potential challenges if the dataset distribution changes. The model's reliance on balanced class representation can limit its effectiveness in imbalanced scenarios.

**Implications**:

1. Robust Outlier Handling:

- The SVM's robust performance suggests its capability to handle outliers effectively. This makes it a favorable choice when dealing with datasets that may contain noise or anomalies.

2. Strategy Insensitivity:

- The consistent performance across one-vs-all and one-vs-one strategies implies that, in this specific case, the choice of strategy does not significantly impact the SVM Classifier's effectiveness

.

* **Naive Bayes Classifier**

1. Sensitivity to Outliers for Extreme Values:

- Naive Bayes can be sensitive to outliers for extreme values, leading to inaccurate probability estimations.

2. Difficulty Handling Non-Normal Distributions:

- Naive Bayes assumes that features follow a Gaussian (normal) distribution. If the data distribution deviates from this assumption, the model's ability to accurately capture the underlying patterns may be compromised, as evidenced by the lower accuracy, precision, recall, and F1 score.

3. Limited Capability for Capturing Complex Relationships:

- The Naive Bayes model, being a simple probabilistic model, struggles to capture complex relationships between features

4. Sensitive to Feature Scaling:

- Naive Bayes assumes that features are conditionally independent given the class, but it does not account for variations in feature scales. Sensitivity to feature scaling could result in certain features dominating the model's decision-making process, especially if their scales differ significantly.

* **Custom Neural Network**

1. Computational Intensity:

- Custom neural networks, especially sophisticated architectures, can be computationally intensive during training and inference for high number of epochs. This can limit their applications requiring real-time predictions when the resource is limited.

2. Interpretability Challenges:

- The inherent complexity of neural networks can result in a lack of interpretability, making it challenging to understand the underlying decision-making process. This limitation could be critical in applications where interpretability is a primary concern. As discussed during our class, the higher the performance, the lower the interpretability.

3. Data Dependency:

- The exceptional performance observed may be highly dependent on the characteristics of the training data. The model might struggle when faced with novel patterns or datasets with different distributions.