

Note: crossed fingers or overlapping landmarks can confuse the mediapipe and cause “jittering” in the landmark positions. If this is a problem consider switching to simple MLP Linear/Dense Network. EDIT it was not

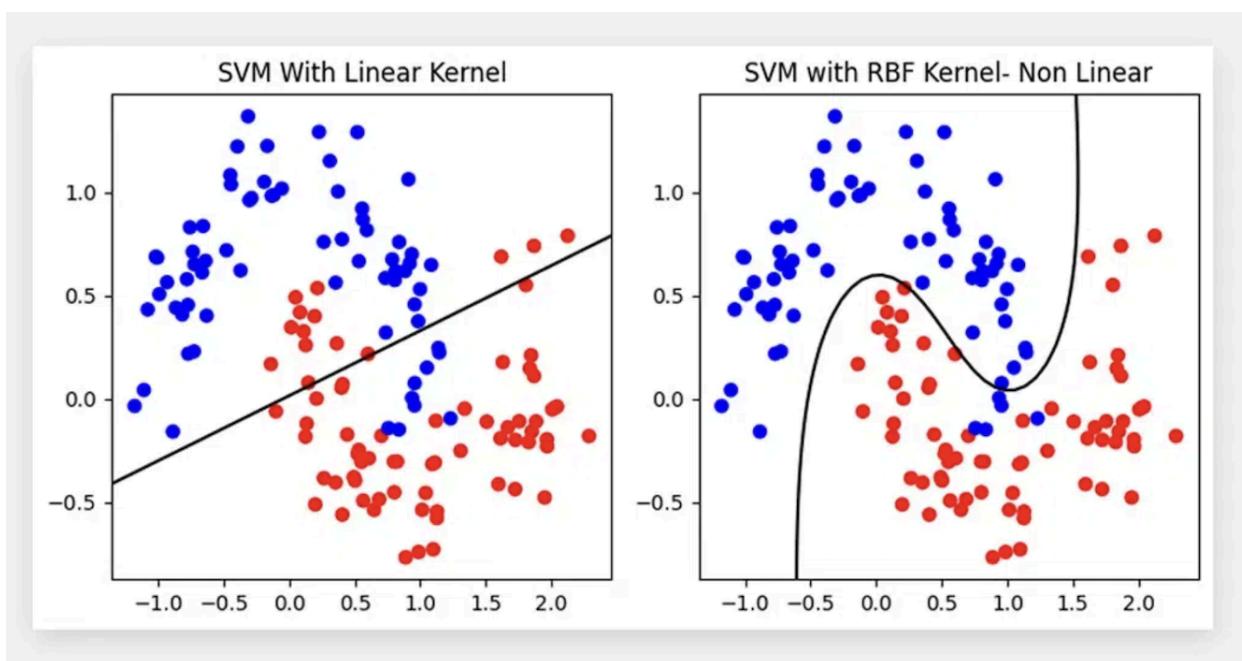
| Model | Speed (Run-time) | Accuracy (Small Data) | Smoothness | Justification |
|---------------|----------------------------------|--------------------------|------------|--|
| KNN | Slow (Get slower with more data) | High | Low | Too slow for real-time, lazy learning is not optimal for potentially thousands of csv lines, every new line has to be compared to these as well. |
| Random Forest | Fast | Medium | Medium | overly sensitive to tiny jitters in hand position, Hand gestures are curved, our data is highly geometrical and non categorical, which is not a good fit |
| Neural Net | Fast | Low (Overfits easily) | High | Overkill for this task |
| RBF SVM | Fast | High | High | Best Balance with eager learning and finding boundary . RBF bell curve works best with curved |

geometric features like the ones we have

Support Vector Machine (rbf SVM) is a great choice for gesture recognition because it often creates cleaner decision boundaries for high-dimensional data (like 129 features).

A basic Linear SVM tries to draw a straight line to separate "Thumbs Up" from "Peace Sign." But real gestures are complex; the difference between a "Fist" and "Thumbs Up" isn't a straight line.

The RBF Kernel allows the model to draw curved, flexible loops around your data clusters. It can wrap a boundary tightly around the "Thumbs Up" examples, excluding everything else.



<https://www.geeksforgeeks.org/machine-learning/linear-vs-non-linear-classification-analyzing-differences-using-the-kernel-trick/>

Rbf is best because we want Soft, curved bubbles around gesture cluster that the kernel tricks provide, not a line like in linear.

KNN would be a good starting point, but it is worse in High Dimensions or with lots of features. We have 129 so RBF would be the best choice