**Introduction to Support Vector Machine and A Comparison Between Linear Kernel & Poly Kernel**

Jonathan Nguyen

ICS 465

University of Hawai’i

**ABSTRACT**

Machine Learning is created to teach computer how to classify the inputs that it was given. Support Vector Machine (SVM) is one of the most popular and effective algorithm for classify inputs. The purpose of this paper is to test on SVM linear to see how accurate and effective it can be on classify data.

**INTRODUCTION**

A Support Vector Machine is a discriminative classifier formally defined by a separating hyperplane, which means it’s a supervised learning method. [1] To begin with, the SVM will be fed with inputs data that are labeled into classes (e.g. [1,-1], [0,1,2]...). These data are called *training data*, then the SVM will use these inputs to apply to its *decision function*:



In which, *n* is the number of data, xi is the

image of a support vector in input space, and

*alphai* is the weight of a support vector in

the feature spaces. *K* is the kernel function.[2]

There are 4 popular kernel functions such as *linear, polynomial, rbf, and sigmoid*. In this paper, I will only compare between linear and polynomial kernel to see how they perform on both linear and non-linear dataset.

**TESTING**

To perform this test, I’m using Python as programming language, scikit-learn for the SVM library, matplotlib for graphing, and numpy for generate data.

**Dataset:**

For the dataset that I use for this experiment, I generate them randomly using numpy.

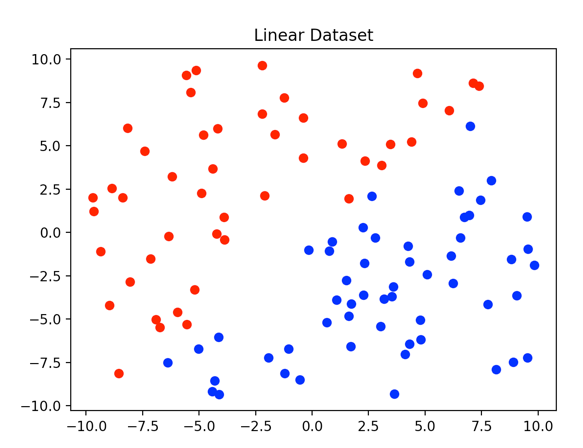
**Labeling:**

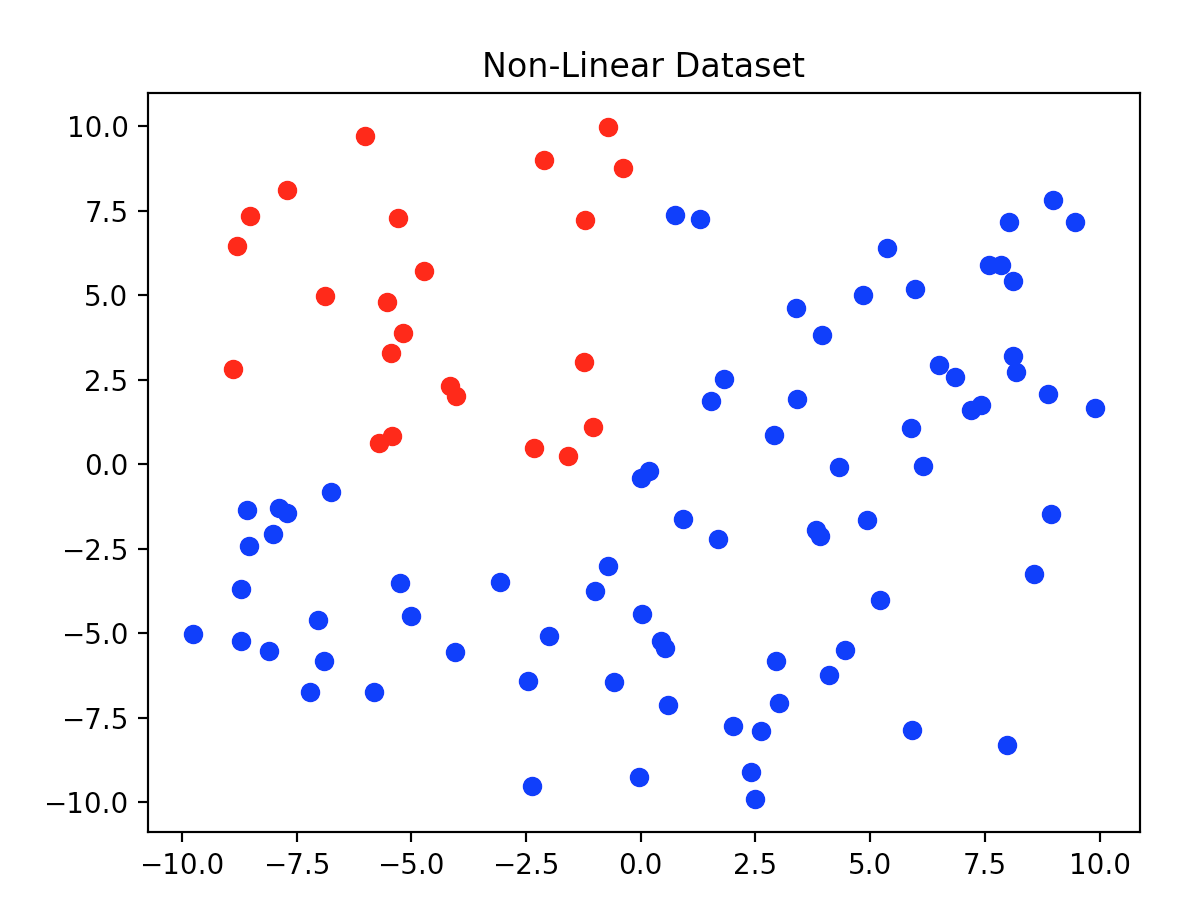
For linear dataset, I labeled them by setting:

1. labels = [1 if y > x else -1 for x, y in dataset]

For non-linear dataset, I labeled them by setting:

1. labels = [1 if (y > 0 and x < 0 ) else -1 for x, y in dataset]

Here are the example datasets:



**Linear dataset test:**

Test 1: Training size

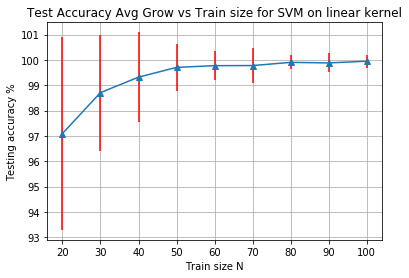
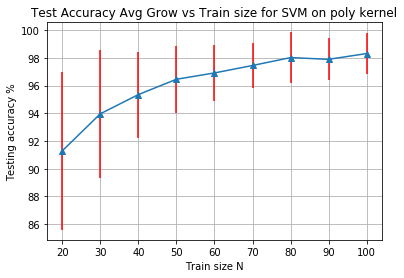
In this test, I will test the accuracy between linear and poly kernel by increasing number of training size.

*Objective:* To see which kernel has the most accurate classification

Constant:

* Testing size: 500
* X values: -100 🡪 100
* Y values: -100 🡪 100
* Margin: 0.5

Results:

As you can see the results from the charts above, linear kernel quickly approach 100% accuracy mark, meanwhile, the highest accuracy that the poly kernel can get is around 98%.

Therefore, this round linear kernel won by 2%.

**Non-linear dataset test:**

Test 1: Training size

In this test, I will test the accuracy between linear and poly kernel by increasing number of training size.

*Objective:* To see which kernel has the most accurate classification

Constant:

* Testing size: 500
* X values: -100 🡪 100
* Y values: -100 🡪 100
* Margin: 0.5

Results: