CS 312 Assignment 6 Report; Team - 21

Josyula V N Taraka Abhishek (200010021), M V Karthik (200010030)

March 16, 2022

1 Abstract

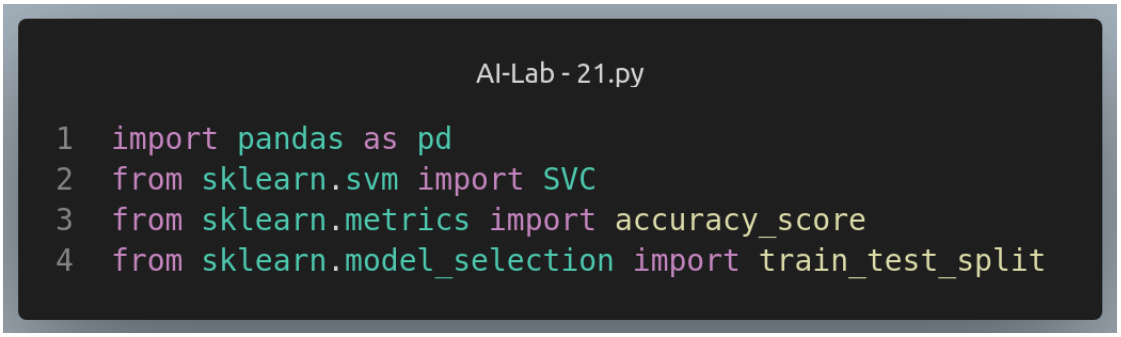
Making Support vector machine and Using it to classify emails into spam or non-spam categories and report the classification accuracy for various SVM parameters and kernel functions.

2 Dataset

We are given with email features like frequency of occurrences of certain keywords, length of capitalized words etc. A data set containing about 4601 instances are available here. We classified 70% of data randomly into training data and test data.

3 Modules used

Below are modules we used in making SVC:



AI-Lab - 21.py

```python
import pandas as pd
from sklearn.svm import SVC
from sklearn.metrics import accuracy_score
from sklearn.model_selection import train_test_split
```

Figure 1: Caption

3.1 pandas

pandas is a Python package providing fast, flexible, and expressive data structures designed to make working with “relational” or “labeled” data both easy and intuitive. It aims to be the fundamental high-level building block for doing practical, real-world data analysis in Python. Additionally, it has the broader goal of becoming the most powerful and flexible open source data analysis/manipulation tool available in any language. It is already well on its way toward this goal.

3.2 sklearn

Scikit-learn (formerly scikits.learn and also known as sklearn) is a free software machine learning library for the Python programming language.[3] It features various classification, regression and clustering algorithms including support vector machines, random forests, gradient boosting, k-means and DBSCAN, and is designed to interoperate with the Python numerical and scientific libraries NumPy and SciPy.

4 Kernels

The following subsections explain the kernels used in the code to implement SVM for the given dataset.

4.1 Linear kernel

4.2 Quadratic kernel

In machine learning, the polynomial kernel is a kernel function commonly used with support vector machines (SVMs) and other kernelized models, that represents the similarity of vectors (training samples) in a feature space over polynomials of the original variables, allowing learning of non-linear models.

Intuitively, the polynomial kernel looks not only at the given features of input samples to determine their similarity, but also combinations of these. In the context of regression analysis, such combinations are known as interaction features. The (implicit) feature space of a polynomial kernel is equivalent to that of polynomial regression, but without the combinatorial blowup in the number of parameters to be learned. When the input features are binary-valued (booleans), then the features correspond to logical conjunctions of input features.

4.3 Radial Basis Function(RBF) Kernel

The radial basis function kernel, or RBF kernel, is a popular kernel function used in various kernelized learning algorithms.

	C	Linear	Poly	RBF
0	0.001	0.893555	0.614048	0.614048
1	0.01	0.923244	0.625634	0.716148
2	0.1	0.927589	0.728458	0.912382
3	1	0.931209	0.847936	0.933382
4	10	0.931933	0.918899	0.93483
5	100	0.929037	0.912382	0.929761
6	500	0.928313	0.908762	0.919623

Table 1: Test accuracy for different values of c and kernes

	C	Linear	Poly	RBF
0	0.001	0.887888	0.603727	0.602484
1	0.01	0.922671	0.619876	0.707143
2	0.1	0.931366	0.723292	0.913975
3	1	0.934161	0.852484	0.947516
4	10	0.935404	0.949689	0.968323
5	100	0.934783	0.970186	0.987267
6	500	0.936025	0.985093	0.991925

Table 2: Train accuracy for different values of c and kernes

5 Results and observations

We made svm with gradient descent optimizer which has maximum iterations of 10^6 with different values of c for each kernel i.e linear, quadratic, RBF kernels.

For **C = 100, we obtain the highest accuracy for testing training dataset in the linear kernel mode implementation.** Also, for C greater than 100 onwards we got maximum accuracy for the quadratic kernel. At the last, for C greater than 100, onwards we got maximum accuracy for training testing accuracy in the RBF kernel.

5.1 Accuracy vs kernels

At small c values linear has more accuracy than others. At high c values like 100 RBF has maximum accuracy of 93%.

5.1.1 Accuracy score vs C - Regularization factor behaviour

For linear kernel, on increasing C value from small values, accuracy is improving and stabilizing at small values. So, small values like 0.5 are having highest accuracy of about 93%.

For Quadratic kernel(polynomial kernel), Also increases accuracy with c value. stability occurs in between 10 and 100.

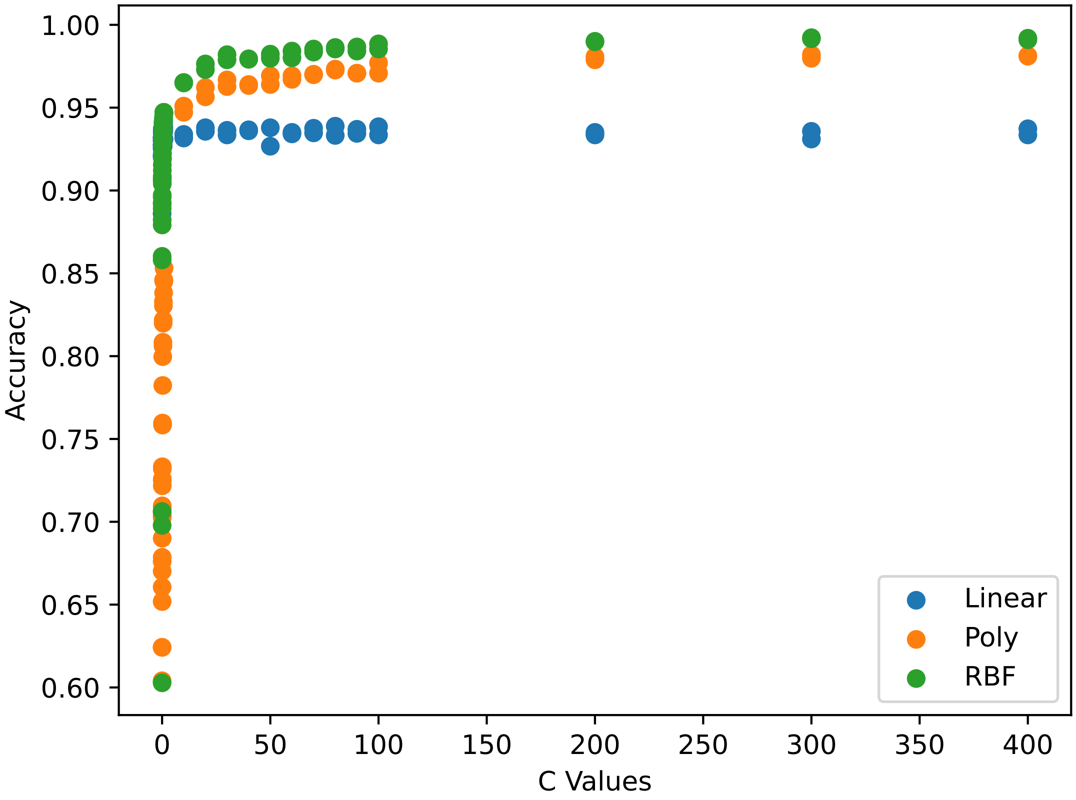


Figure 2: Accuracy vs c-values for train data

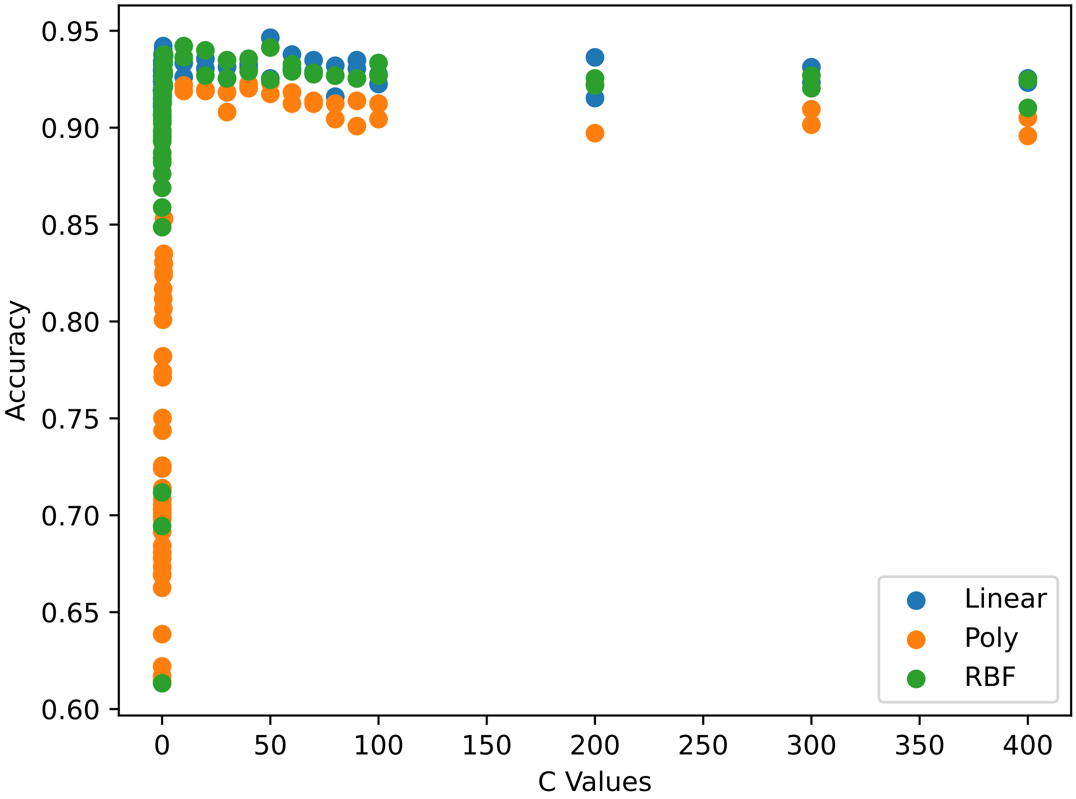


Figure 3: Accuracy vs c-values for test data

For RBF kernels, Accuracy keeps on increases and stabilization is not seen before 1000. Because time complexity we can't go beyond 1000.

6 Conclusion

Because for For very tiny values of C, we are getting more mis-classified examples, often even if our training data is linearly separable. So, this is the final conclusion across different C parameter and various kernels for the given dataset.