GitHub: https://github.com/helga119/Machine-Learning-Tutorial.git

Title

Exploring the Behaviour of Support Vector Machines: A Tutorial on Linear vs RBF Kernels

Objective

One of the most important choices when creating a Support Vector Machine (SVM) model is selecting the appropriate kernel. The SVM uses kernels as tools to differentiate between various data classes. A kernel can be thought of as a technique used by the SVM to draw a boundary line separating such classes.

The purpose of this tutorial is to examine ad compare the behaviour of SVM using two very popular kernels: the Radial Basis Function kernel and the linear kernel. Using the Breast Cancer dataset as a real-world example to enact this comparison, we will focus our attention on how these kernels affect the decision boundary and the model's performance. I'll take you through each stage of putting both SVM models into practice and then we will assess the model's performance.

Why is this topic important?

it is very important we choose the right kernel if we want to build an effective SVM model. the difference between the two is that a linear kernel separates data using a straight line (or a flat plane in higher dimensions). It is very straightforward, and this makes it a very good choice for problems where it is easy to divide data cleanly with a straight boundary. But linear kernels do struggle when data is more complex in pattern or more complicated or such that it overlaps, in these situations RBF kernel is much better.

The RBF kernel is very important because it can create curved and more flexible boundary lines as compared to the linear kernel. This allows the SVM to handle data with complex relationships. however, this does come with a drawback in that RBF can be much slower to compute and harder to understand compared to linear. So, knowing when to use each is integral for the success of a model. it is easy to understand that if the problem is simple use linear if the problem is containing data with more complex relationships such as overlapping classes and such, use RBF.

What is SVM?

As geeksforgeeks.org state 'Support Vector Machine is a supervised learning algorithm used for tasks like classification and regression.' Its main goal is to find the optimal boundary line, also known as the hyperplane, that can separate data into respective classes. For example, in the Brest Cancer Dataset we will be looking at SVM has the ability to separate the data

points into the two classes benign and malignant based on the features in the dataset. it does this by the algorithm identifying the boundary line that maximizes the margin, between the two classes.

Understanding kernels

Kernels are integral to Support Vector Machines (SVM), they essentially make the algorithms adapt to the dataset structure. they do this by the by transforming the data to a higher dimensional plane and this is why the SVM can effectively divide data points into classes. The kernel you choose will weigh heavily on the success of the models and will determine the type of boundary line. A linear kernel is great for simply dividing the classes by a straight line it is very easy to understand- and is integral for problems where readability is key.

However, the RBF kernel is much better at more complex data, and it does this by mapping the data into a higher dimensional plane. the procedure allows the SVM to create more flexible boundary lines that can divide more complex data points into classes, so choosing the correct kernel is very important and you would need to look at the dataset and pick out the behaviours of the data points first before choosing a kernel to tell if a linear kernel of RBF kernel would be best. Although, the linear kernels do not incur computational cost but the RBF does, even though RBF is flexible in separating complex datasets. It is first priority that you look at your dataset first to define if there are significant overlaps or not and from them decide which you will choose. you can use visualization or statistical analysis to do this.

Mathematics behind kernels

To first understand how SVM works its important that the mathematics are explained, the kernel will be integral in how the SVM separates data points into classes and the mathematics behind them will show you how they are able to flexibly do. Below I will go through the mathematics for the linear and RBF kernels.

Linear Kernel

The liner kernel would be the easiest kernel to leanr. As geeksforgeeks.org state 'It computes the dot product between two vectors, measuring their similarity in the original input space.'

(image from geeksforgeeks.org)

*It is defined as , **K(x, y) = x * y** , where x and y are the input vectors.*

- For example, if x = [4,7] and y =[9,1] their dot product is:

K(x, y) = 4 x 9 + 7 x1 = 36 + 7 = 43

A large number means there is a higher similarity between the two vectors.

- During the process as SVM classifies the linear kernel it will create a hyperplane that separates the different classes in the input space, this separation line will make a straight line in a 2d space and a flat plane in a 3d space, this only works in the data points do not overlap.
- Since the linear kernel does all of this within the input space this makes it computationally effective and will work great will large datasets.

RBF (Radial Basis Function) kernel

Needless to say the RBF kernel, is defined as a non-linear kernel. differently to the linear kernel for this to classify complex datasets it needs to map them to a higher dimensional plane and it does this by using the exponential function. This allows the SVM to determine a boundary line in a new space.

Kernal Function (image from geeksforgeeks.org)

It is defined as

$K(x, y) = exp(-gamma * ||x - y||^2)$ *, where x and y are the input vectors, gamma is a positive parameter, and ||x - y|| is the Euclidean distance between x and y.*

- The part '||x-y||2 ' represents the squared Euclidean distance between two data points, x and y. it calculates how far these points are from each other and points that are further from each other have a lower similarity score while points closer to each other have a higher score.
- the gamma parameter decides how much the individual data points will influence the boundary line. A lower gamma will create a simpler boundary, and a higher gamma will create a more complex boundary this will capture the finer intricacies in the data. However, selecting an overly low gamma value can cause problems and lead to underfitting to the data and a higher gamma value can lead to drawbacks as it can cause overfitting to the data.

Why RBF Creates Curved Boundaries

The RBF basically creates a new dimensional space that it can transform data to and that allows it to divide nonlinear classes with a linear decision line.

1. first it will begin in the input space then if the data is overlapping or showing a complex relationship it will not be divisible by a straight line

2. In that case the RBF would map these to a higher dimensional plane where it can then linearly separate the data points into classes

3. When that is done the linear decision boundary will be projected back into the input space and it will create a curved boundary line that can now divide the complex data

For example:

think of a dataset with only two classes, where one of the classes is nested in another class and both classes form a circle. In the input space this can't be separated by a straight line because they are both circles. however what RBF can do is that is can map this to a higher dimensional space, then a linear boundary can sperate these into respective classes and when this is projected back the boundary would appear as a circle.

Support Vector Machines for Breast Cancer Classification

This tutorial we will explore the use of SVM to divide the breast cancer cases as either benign or malignant. To note in medical tasks where it is important to read clear results the SVM excels in this. We will be comparing linear and rbf kernels to see which of these will perform better and to dissect the different results we get to see which model will be best for our problem. we will be using metrics to evaluate each model such as precision, recall, F-1 and that will determine which model performs best and the contexts of why.

The breast cancer dataset contains 30 features that we will use to determine the cancer cases this includes texture, perimeter, radius of the tumour. The diagnoses columns have been encoded as binary values so if the tumour is cancerous it will be 1 if it is not it will be 0. and all unnecessary columns are removed from the dataset we will be examining.

```
data = data.drop(columns=['id', 'Unnamed: 32'])
data['diagnosis'] = data['diagnosis'].map({'M': 1, 'B': 0})
```

Before we begin our analysis we will be splitting the features and target variable into training and test sets , doing a 70-30 split . 70% will be reserved for training and 30% will be reserved for testing. We will be using StandardScaler to scale the features since SVM is sensitive to that, this point is very important and should not be missed because this can affect you models results.

```
X = data.drop(columns=['diagnosis'])
y = data['diagnosis']
```

```
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.3, random_state=42)
```

```
scaler = StandardScaler()
X_train = scaler.fit_transform(X_train)
X_test = scaler.transform(X_test)
```

Here we will be comparing both models the linear and the RBF models and the same regularization parameter was used for both at C = 1.0 with default hyperparameters to create a fair comparison.

```
linear_svm = SVC(kernel='linear', C=1.0, random_state=42)
linear_svm.fit(X_train, y_train)

rbf_svm = SVC(kernel='rbf', C=1.0, gamma='scale', random_state=42)
rbf_svm.fit(X_train, y_train)
```

```
SVC
SVC(random_state=42)
```

```
y_pred_linear = linear_svm.predict(X_test)
y_pred_rbf = rbf_svm.predict(X_test)
```

Results

The models are evaluated with the following results:

1. Linear kernel SVM performance

- Accuracy: 97.66%
- Precision:0.98(for benign and malignant classes)
- Recall: 0.98 (for benign) and 0.97 (for malignant)
- F1-score: 0.98
- Confusion matrix:

```
Confusion Matrix - Linear Kernel:
 [[106   2]
 [  2  61]]
```

2. RBF kernel SVM performance

- Accuracy: 97.66%

- Precision:0.98(for benign and malignant classes)
- Recall: 0.98 (for benign) and 0.97 (for malignant)
- F1-score: 0.98
- Confusion matrix:

```
Confusion Matrix - Linear Kernel:
 [[106   2]
 [  2  61]]
```

From the results above we can see that the dataset is mostly linearly separable since both results were so similar, the RBF would not be needed for our problem as it is more computationally expensive and would only garner the similar results, as the must lower computationally costing and easier to understand linear kernel

Discussion

The results from both the linear and RBF kernels demonstrated identical performance metrics for this dataset, indicating that the data is predominantly linearly separable. However, in many cases, the choice of kernel can lead to differences in performance, making it essential to understand these variations for effective model evaluation.

When one kernel outperforms another in terms of accuracy, precision, recall, or F1-score, it suggests that the kernel is better at capturing the dataset's structure. For instance, an RBF kernel may excel in datasets with non-linear patterns, while a linear kernel may be better for simpler problems. When performance differences are marginal, opting for the simpler linear kernel is often preffered because of its computational efficiency.

From the result we can see almost identical metrics showcasing a very similar performance from both models for this dataset. however, in most cases, depending on the kernel you choose you would get vastly different results comparing performances which is why its important to look at the accuracy, precision, recall, or F1-score, to determine which model outperforms the other. It’s also important to consider the computational costs as well as these metrics as larger datasets or real time applications might consider using linear kernel compared to RBF and complex datasets would prefer RBF. as well be mindful to watch out for overfitting as a good accuracy score on the training set but a poor score on the test set can signal overfitting, which is more likely to occur with RBF. now that you have learned about these two kernels, I hope you know now which better to us to solve your needs.

References

1. Tibrewal, T.P. (2023) Support Vector Machines (SVM): An Intuitive Explanation, Low Code for Data Science. Available at: https://medium.com/low-code-for-advanced-data-science/support-vector-machines-svm-an-intuitive-explanation-b084d6238106.
2. Support Vector Machine (SVM) in 2 minutes (no date) www.youtube.com. Available at: https://www.youtube.com/watch?v=_YPScrckx28.
3. Starmer, J. (2019) *Support Vector Machines Part 1 (of 3): Main Ideas!!!*, *www.youtube.com*. Available at: https://www.youtube.com/watch?v=efR1C6CvhmE.
4. GeeksforGeeks (2024) How to Choose the Best Kernel Function for SVMs, GeeksforGeeks. GeeksforGeeks. Available at: https://www.geeksforgeeks.org/how-to-choose-the-best-kernel-function-for-svms/ (Accessed: 20 April 2024).
5. SVM Kernels : Data Science Concepts (no date) www.youtube.com. Available at: https://www.youtube.com/watch?v=OKFMZQyDROI.
6. SVM with polynomial kernel visualization (no date) www.youtube.com. Available at: https://www.youtube.com/watch?v=3liCbRZPrZA.
7. Mahesh Huddar (2023) What is Kernel Trick in Support Vector Machine | Kernel Trick in SVM Machine Learning Mahesh Huddar, YouTube. Available at: https://www.youtube.com/watch?v=Js2oAqEN_h0 (Accessed: 2 December 2024).
8. Udacity (2015) Kernel and Gamma - Intro to Machine Learning, YouTube. Available at: https://www.youtube.com/watch?v=pH51jLfGXe0 (Accessed: 2 December 2024).
9. ritvikmath (2021) 'SVM Dual : Data Science Concepts', YouTube. Available at: https://www.youtube.com/watch?v=6-ntMIaJpm0 (Accessed: 23 December 2022).
10. Siddhardhan (2021) 7.3.2. Math behind Support Vector Machine Classifier, YouTube. Available at: https://www.youtube.com/watch?v=y0_Qq6fXzCs&list=PLfFghEzKVmjvzS4DILijsdQk27Ew7xIPu&index=2 (Accessed: 2 December 2024).
11. GeeksforGeeks (2024) Implementing SVM from Scratch in Python, GeeksforGeeks. Available at: https://www.geeksforgeeks.org/implementing-svm-from-scratch-in-python/#step-2-svm-class-definition (Accessed: 2 December 2024).
12. Introduction to Support Vector Machines (SVM) (2020) GeeksforGeeks. Available at: https://www.geeksforgeeks.org/introduction-to-support-vector-machines-svm/.
13. Team, P. (2022) Support Vector Machine, Python Geeks. Available at: https://pythongeeks.org/support-vector-machine/ (Accessed: 2 December 2024).
14. Bento, C. (2020) Support Vector Machines explained with Python examples, Medium. Available at: https://towardsdatascience.com/support-vector-machines-explained-with-python-examples-cb65e8172c85.

15. RBF SVM Parameters in Scikit Learn (2023) GeeksforGeeks. Available at: https://www.geeksforgeeks.org/rbf-svm-parameters-in-scikit-learn/.
16. Radial Basis Function Kernel - Machine Learning (2020) GeeksforGeeks. Available at: https://www.geeksforgeeks.org/radial-basis-function-kernel-machine-learning/.
17. How to Make Better Models in Python using SVM Classifier and RBF Kernel (2023) GeeksforGeeks. Available at: https://www.geeksforgeeks.org/how-to-make-better-models-in-python-using-svm-classifier-and-rbf-kernel/.
18. Creating linear kernel SVM in Python (2018) GeeksforGeeks. Available at: https://www.geeksforgeeks.org/creating-linear-kernel-svm-in-python/.
19. GeeksforGeeks (2024) What is the influence of C in SVMs with linear kernel?, GeeksforGeeks. Available at: https://www.geeksforgeeks.org/what-is-the-influence-of-c-in-svms-with-linear-kernel/ (Accessed: 2 December 2024).
20. Support Vector Regression (SVR) using linear and non-linear kernels (no date) scikit-learn. Available at: https://scikit-learn.org/stable/auto_examples/svm/plot_svm_regression.html.
21. Malik, U. (2018) Implementing SVM and Kernel SVM with Python's Scikit-Learn, Stack Abuse. Stack Abuse. Available at: https://stackabuse.com/implementing-svm-and-kernel-svm-with-pythons-scikit-learn/.
22. Oleszak, M. (2021) SVM Kernels: What Do They Actually Do?, Medium. Available at: https://towardsdatascience.com/svm-kernels-what-do-they-actually-do-56ce36f4f7b8.
23. *Dot Product* (2024) *GeeksforGeeks*. Available at: https://www.geeksforgeeks.org/dot-product/.