



Classifying data using Support Vector Machines(SVMs) in Python

Last Updated : 01 Sep, 2023



Introduction to SVMs: In machine learning, support vector machines (SVMs, also support vector networks) are supervised learning models with associated learning algorithms that analyze data used for classification and regression analysis. A Support Vector Machine (SVM) is a discriminative classifier formally defined by a separating hyperplane. In other words, given labeled training data (supervised learning), the algorithm outputs an optimal hyperplane which categorizes new examples.

What is Support Vector Machine?

An SVM model is a representation of the examples as points in space, mapped so that the examples of the separate categories are divided by a clear gap that is as wide as possible. In addition to performing linear classification, SVMs can efficiently perform a non-linear classification, implicitly mapping their inputs into high-dimensional feature spaces.

What does SVM do?

Given a set of training examples, each marked as belonging to one or the other of two categories, an SVM training algorithm builds a model that assigns new examples to one category or the other, making it a non-probabilistic binary linear classifier. Let you have basic understandings from this [article](#) before you proceed further. Here I'll discuss an example about SVM classification of [cancer](#) UCI datasets using machine learning tools i.e. scikit-learn compatible with Python. **Pre-requisites:** [Numpy](#), [Pandas](#), [matplotlib-lib](#), [scikit-learn](#) Let's have a quick example of support vector classification. First we need to create a dataset:

python3

```
# importing scikit learn with make_blobs
from sklearn.datasets import make_blobs

# creating datasets X containing n_samples
# Y containing two classes
X, Y = make_blobs(n_samples=500, centers=2,
                  random_state=0, cluster_std=0.40)
import matplotlib.pyplot as plt
# plotting scatters
plt.scatter(X[:, 0], X[:, 1], c=Y, s=50, cmap='spring');
plt.show()
```

Output:



What Support vector machines do, is to not only draw a line between two classes here, but consider a region about the line of some given width. Here's an example of what it can look like:

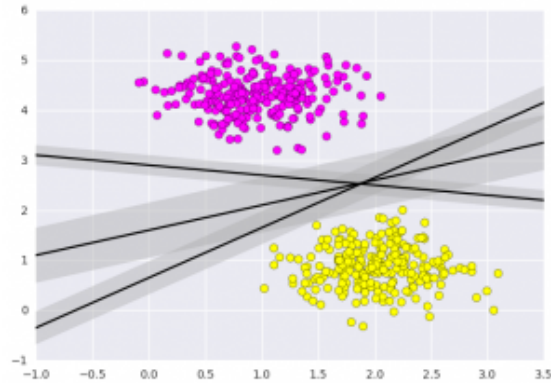
python3

```
# creating linspace between -1 to 3.5
xfit = np.linspace(-1, 3.5)

# plotting scatter
plt.scatter(X[:, 0], X[:, 1], c=Y, s=50, cmap='spring')

# plot a line between the different sets of data
for m, b, d in [(1, 0.65, 0.33), (0.5, 1.6, 0.55), (-0.2, 2.9, 0.2)]:
    yfit = m * xfit + b
    plt.plot(xfit, yfit, '-k')
    plt.fill_between(xfit, yfit - d, yfit + d, edgecolor='none',
                    color='#AAAAAA', alpha=0.4)
```

```
plt.xlim(-1, 3.5);  
plt.show()
```



Importing datasets

This is the intuition of support vector machines, which optimize a linear discriminant model representing the perpendicular distance between the datasets. Now let's train the classifier using our training data. Before training, we need to import cancer datasets as csv file where we will train two features out of all features.

python3

```
# importing required libraries
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt

# reading csv file and extracting class column to y.
x = pd.read_csv("C:\\...\\cancer.csv")
a = np.array(x)
y = a[:,30] # classes having 0 and 1

# extracting two features
x = np.column_stack((x.malignant,x.benign))

# 569 samples and 2 features
x.shape

print (x),(y)
```

```
[[ 122.8  1001. ]
 [ 132.9  1326. ]
 [ 130.   1203. ]
 ...,
 [ 108.3   858.1 ]
 [ 140.1  1265. ]
 [  47.92  181.  ]]

array([ 0.,  0.,  0.,  0.,  0.,  0.,  0.,  0.,  0.,  0.,  0.,
        0.,  0.,
         0.,  0.,  0.,  0.,  0.,  0.,  1.,  1.,  1.,  0.,  0.,
        0.,  0.,
         0.,  0.,  0.,  0.,  0.,  0.,  0.,  0.,  0.,  0.,  0.,
        1.,  0.,
         0.,  0.,  0.,  0.,  0.,  0.,  0.,  1.,  0.,  1.,  1.,
        1.,  1.,
         1.,  0.,  0.,  1.,  0.,  0.,  1.,  1.,  1.,  1.,  0.,
        1., ...,
         1.])
```

Fitting a Support Vector Machine

Now we'll fit a Support Vector Machine Classifier to these points. While the mathematical details of the likelihood model are interesting, we'll let read about those elsewhere. Instead, we'll just treat the scikit-learn algorithm as a black box which accomplishes the above task.

python3

```
# import support vector classifier
# "Support Vector Classifier"
from sklearn.svm import SVC
clf = SVC(kernel='linear')

# fitting x samples and y classes
clf.fit(x, y)
```

After being fitted, the model can then be used to predict new values:

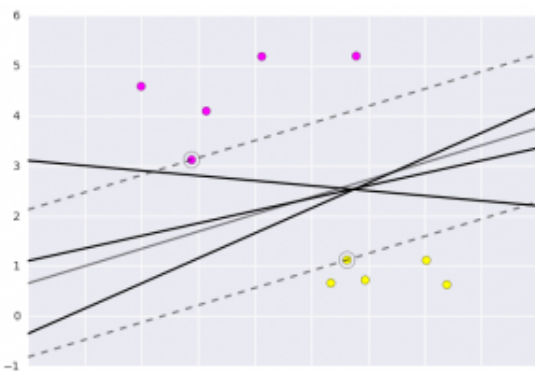
python3

```
clf.predict([[120, 990]])

clf.predict([[85, 550]])
```

```
array([ 0.])
array([ 1.])
```

Let's have a look on the graph how does this show.



This is obtained by analyzing the data taken and pre-processing methods to make optimal hyperplanes using matplotlib func If you like GeeksforGeeks and would like to contribute, you can also write an article using [write.geeksforgeeks.org](https://www.geeksforgeeks.org) or mail your article to review-team@geeksforgeeks.org. See your article appearing on the GeeksforGeeks main page and help other Geeks.

 Comment

More info 

Advertise with us

Next Article 

Support Vector Regression (SVR)
using Linear and Non-Linear Kernels
in Scikit Learn

Similar Reads

Machine Learning Algorithms

Machine learning algorithms are essentially sets of instructions that allow computers to learn from data, make predictions, and improve their performance over time without being explicitly programmed. Machine...

 9 min read

Top 15 Machine Learning Algorithms Every Data Scientist Should Know in 2025

Machine Learning (ML) Algorithms are the backbone of everything from Netflix recommendations to fraud detection in financial institutions. These algorithms form the core of intelligent systems, empowering...

 15 min read

Linear Model Regression



Linear Model Classification



Regularization



K-Nearest Neighbors (KNN)



Support Vector Machine (SVM) Algorithm

Support Vector Machine (SVM) is a supervised machine learning algorithm used for classification and regression tasks. While it can handle regression problems, SVM is particularly well-suited for...

🕒 10 min read

Classifying data using Support Vector Machines(SVMs) in Python

Introduction to SVMs: In machine learning, support vector machines (SVMs, also support vector networks) are supervised learning models with associated learning algorithms that analyze data used for...

🕒 4 min read

Support Vector Regression (SVR) using Linear and Non-Linear Kernels in Scikit Learn

Support vector regression (SVR) is a type of support vector machine (SVM) that is used for regression tasks. It tries to find a function that best predicts the continuous output value for a given input value. SVR...

🕒 5 min read

Major Kernel Functions in Support Vector Machine (SVM)

In previous article we have discussed about SVM(Support Vector Machine) in Machine Learning. Now we are going to learn in detail about SVM Kernel and Different Kernel Functions and its examples. Types o...

🕒 4 min read

ML | Stochastic Gradient Descent (SGD)

Gradient Descent is an iterative optimization process that searches for an objective function's optimum value (Minimum/Maximum). It is one of the most used methods for changing a model's parameters in orde...

🕒 10 min read

Decision Tree



Ensemble Learning



A-143, 7th Floor, Sovereign Corporate
Tower, Sector- 136, Noida, Uttar Pradesh
(201305)



Registered Address:

K 061, Tower K, Gulshan Vivante
Apartment, Sector 137, Noida, Gautam
Buddh Nagar, Uttar Pradesh, 201305



Advertise with us

Company

About Us
Legal
Privacy Policy
In Media
Contact Us
Advertise with us
GFG Corporate Solution
Placement Training Program
GeeksforGeeks Community

DSA

Data Structures
Algorithms
DSA for Beginners
Basic DSA Problems
DSA Roadmap
Top 100 DSA Interview Problems
DSA Roadmap by Sandeep Jain
All Cheat Sheets

Web Technologies

HTML
CSS
JavaScript
TypeScript
ReactJS
NextJS
Bootstrap
Web Design

Computer Science

Operating Systems
Computer Network
Database Management System
Software Engineering
Digital Logic Design
Engineering Maths
Software Development
Software Testing

System Design

High Level Design
Low Level Design
UML Diagrams
Interview Guide

Languages

Python
Java
C++
PHP
GoLang
SQL
R Language
Android Tutorial
Tutorials Archive

Data Science & ML

Data Science With Python
Data Science For Beginner
Machine Learning
ML Maths
Data Visualisation
Pandas
NumPy
NLP
Deep Learning

Python Tutorial

Python Programming Examples
Python Projects
Python Tkinter
Web Scraping
OpenCV Tutorial
Python Interview Question
Django

DevOps

Git
Linux
AWS
Docker
Kubernetes
Azure
GCP
DevOps Roadmap

Interview Preparation

Competitive Programming
Top DS or Algo for CP
Company-Wise Recruitment Process
Company-Wise Preparation

Design Patterns
OOAD
System Design Bootcamp
Interview Questions

Aptitude Preparation
Puzzles

School Subjects

Mathematics
Physics
Chemistry
Biology
Social Science
English Grammar
Commerce
World GK

GeeksforGeeks Videos

DSA
Python
Java
C++
Web Development
Data Science
CS Subjects