

Mind Map Outliner

Topic Subtopic Relationship Summary Boundary Insert

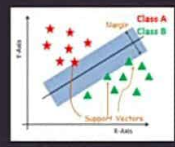
ZEN Pitch

Panel

Handwritten Digit Recognition using SVM Classifier

8 Algorithm Support Vector Machine - Classifier

Non-Linear SVM



Main Components

- Hyperplane**
It is a decision plane which separates between a set of objects
- Support Vectors**
Data point close to the Hyperplane, which defines the separating line better by calculating margins
- Margin**
It is a gap between the two lines on the closest class points

SVM Kernel

Linear Kernel

$K(x, xi) = \sum(x * xi)$
sum of the multiplication of each pair of input values

Default

Radial Basis Function Kernel

$$K(x, xi) = \exp(-\gamma * \sum(|x - xi|^2))$$

It can map an input space in infinite dimensional space

Gamma range 0 to 1

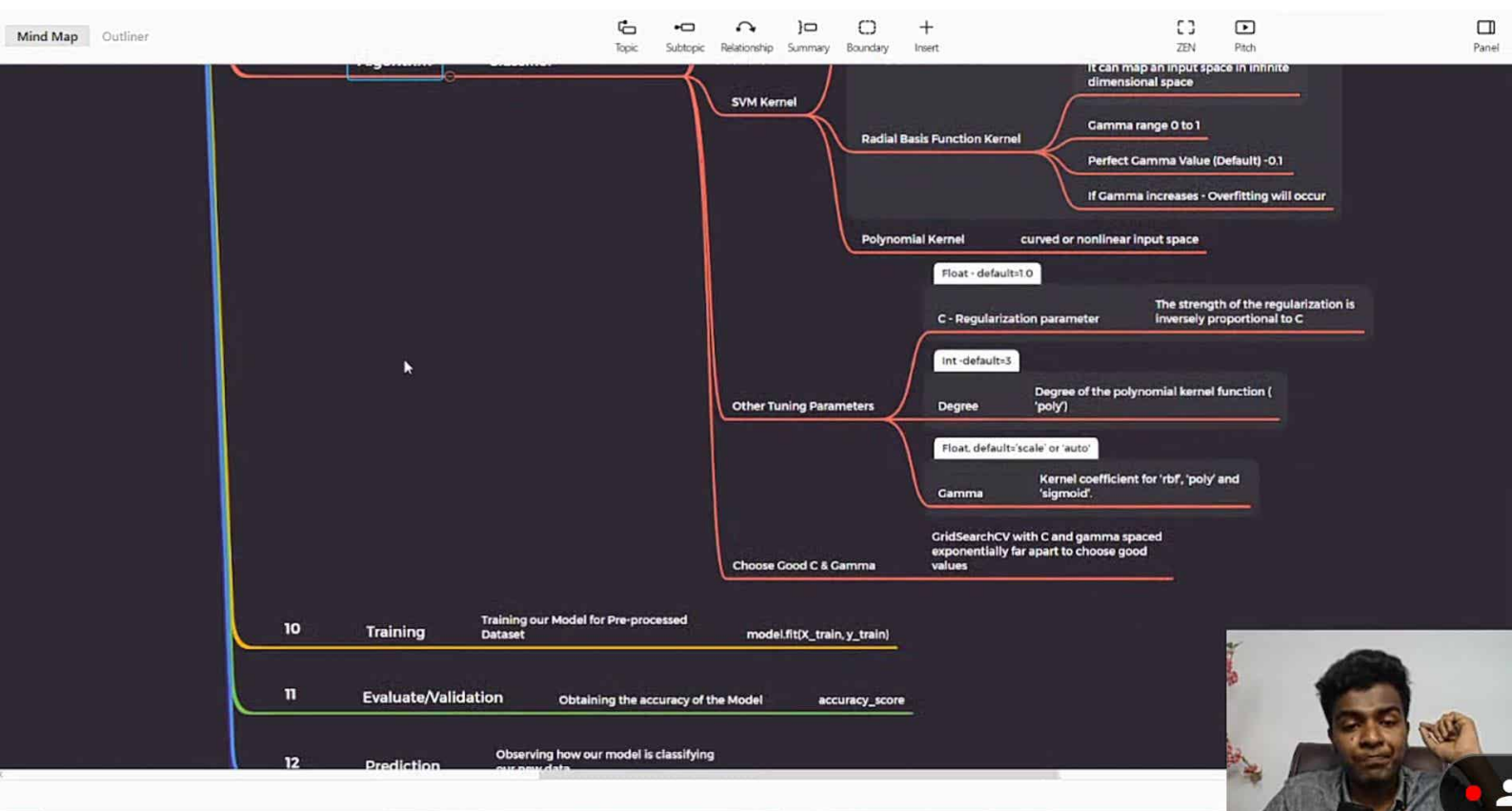
Perfect Gamma Value (D)

If Gamma increases - Overfitting

Polynomial Kernel

curved or nonlinear input space







colab.research.google.com/drive/1EK2eiDs7ikL1T0zSulcowcAiy0S8Yri#scrollTo=KgOT1pQe-Sij

Day5_HandwrittenDigitRecognition_SVM.ipynb

File Edit View Insert Runtime Tools Help All changes saved

+ Code + Text


RAM  Disk  Editing

```
import numpy as np
from sklearn.datasets import load_digits
```

Load Dataset

Classes	10
Samples per class	~180
Samples total	1797
Dimensionality	64
Features	integers 0-16

0s completed at 7:44 PM





colab.research.google.com/drive/1EK2eiDs7ikL1T0zSulcowcAiy0S8Yri#scrollTo=mEkoGFM3_Rl6

Day5_HandwrittenDigitRecognition_SVM.ipynb


File Edit View Insert Runtime Tools Help Saving...

+ Code + Text

RAM  Disk 

Editing

↑ ↓ ↺ 💬 ⚙️ 📄 🗑️ ⋮

0s  dataset = load_digits()

Summarize Dataset

```
[ ] print(dataset.data)
    print(dataset.target)


    print(dataset.data.shape)
    print(dataset.images.shape)
```

✓ 0s completed at 7:45 PM

3_AdSalePrediction.pptx

Type here to search

30°C Rain showers



```
+ Code + Text
```

```
print(dataset.images.shape)
```

```
dataimageLength = len(dataset.images)
```

```
print(dataimageLength)
```

```
[[ 0.  0.  5. ...  0.  0.  0.]
```

```
[ 0.  0.  0. ... 10.  0.  0.]
```

```
[ 0.  0.  0. ... 16.  9.  0.]
```

```
...
```

```
[ 0.  0.  1. ...  6.  0.  0.]
```

```
[ 0.  0.  2. ... 12.  0.  0.]
```

```
[ 0.  0. 10. ... 12.  1.  0.]]
```

```
[0 1 2 ... 8 9 8]
```

```
(1797, 64)
```

```
(1797, 8, 8)
```

```
1797
```

Visualize the Dataset

```
n=9 #No. of Sample out of Samples total 1797

import matplotlib.pyplot as plt
plt.gray()
plt.matshow(dataset.images[n])
plt.show()

dataset.images[n]
```

Segregate Dataset into X(Input/IndependentVariable) & Y(Output/Depend

0s completed at 7:45 PM

colab.research.google.com/drive/1EK2eiDs7ikLf1T0zSulcowcAiy0S8Yri#scrollTo=sGYJfmAvCj3a

Day5_HandwrittenDigitRecognition_SVM.ipynb

CommentShareSettingsProfile

FileEditViewInsertRuntimeToolsHelpAll changes saved

+ Code+ Text

RAMDiskEditing

Segregate Dataset into X(Input/IndependentVariable) & Y(Output/DependentVariable)

Input - Pixel | Output - Class

+ Code+ Text

X = dataset.images.reshape((dataimageLength,-1))

X

[] Y = dataset.target

Y

Splitting Dataset into Train & Test

0s completed at 7:51 PM

3_AdSalePrediction.pptxShow all

colab.research.google.com/drive/1EK2eiDs7ikL1T0zSulcowcAiy0S8Yri#scrollTo=1760lRsBHNw2

Day5_HandwrittenDigitRecognition_SVM.ipynb

CommentShare

FileEditViewInsertRuntimeToolsHelp

+ Code+ Text

RAMDisk

Editing

Splitting Dataset into Train & Test

from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test = train_test_split(X, Y, test_size = 0.25, random_state = 0)
print(X_train.shape)
print(X_test.shape)

(1347, 64)
(450, 64)

0s

completed at 7:53 PM



3_AdSalePrediction.pptx

colab.research.google.com/drive/1EK2eiDs7ikLf1T0zSulcowcAiy0S8Yri#scrollTo=ul79KjPYH3Ud

Day5_HandwrittenDigitRecognition_SVM.ipynb

File Edit View Insert Runtime Tools Help

+ Code + Text

RAM  Disk  Editing

Training

```
from sklearn import svm
model = svm.SVC(kernel='linear')
model.fit(X_train,y_train)
```

```
SVC(C=1.0, break_ties=False, cache_size=200, class_weight=None, coef0=0.0,
    decision_function_shape='ovr', degree=3, gamma='scale', kernel='linear',
    max_iter=-1, probability=False, random_state=None, shrinking=True,
    tol=0.001, verbose=False)
```

0s completed at 7:54 PM

3_AdSalePrediction.pptx

Show all

colab.research.google.com/drive/1EK2eiDs7ikLf1T0zSulcowcAiy0S8Yri#scrollTo=Cf1TTkbAO8i9

Day5_HandwrittenDigitRecognition_SVM.ipynb

File Edit View Insert Runtime Tools Help All changes saved

Comment Share

RAM Disk

Editing

+ Code + Text

Predicting, what the digit is from Test Data

```
n=13
result = model.predict(dataset.images[n].reshape((1,-1)))
plt.imshow(dataset.images[n], cmap=plt.cm.gray_r, interpolation='nearest')
print(result)
print("\n")
plt.axis('off')
plt.title('%i' %result)
plt.show()
```

[3]

0s completed at 8:08 PM



3_AdSalePrediction.pptx Show all

colab.research.google.com/drive/1EK2eiDs7ikLf1T0zSulcowcAiy0S8Yri#scrollTo=Cf1TTKbAO8i9

Day5_HandwrittenDigitRecognition_SVM.ipynb

File Edit View Insert Runtime Tools Help [All changes saved](#)

+ Code + Text

RAM  Disk  Editing

Prediction for Test Data

```
[14] y_pred = model.predict(X_test)
      print(np.concatenate((y_pred.reshape(len(y_pred),1), y_test.reshape(len(y_test),1)),1))
```

```
[4 4]
[6 6]
[0 0]
[7 7]
[7 7]
[6 6]
```

✓ 0s completed at 8:11 PM



3_AdSalePrediction.pptx Show all

colab.research.google.com/drive/1EK2eiDs7ikLf1T0zSulcowcAiy0S8Yri#scrollTo=irsP8j9KJnl-

Day5_HandwrittenDigitRecognition_SVM.ipynb

File Edit View Insert Runtime Tools Help All changes saved

+ Code + Text

RAM  Disk  Editing

```
from sklearn.metrics import accuracy_score
print("Accuracy of the Model: {0}%".format(accuracy_score(y_test, y_pred)*100))
```

Accuracy of the Model: 97.11111111111111%

Play with the Different Method

```
from sklearn import svm
model1 = svm.SVC(kernel='linear')
model2 = svm.SVC(kernel='rbf')
model3 = svm.SVC(gamma=0.001)
model4 = svm.SVC(gamma=0.001,C=0.1)
```

0s completed at 8:12 PM



3_AdSalePrediction.pptx Show all

colab.research.google.com/drive/1EK2eiDs7ikLf1T0zSulcowcAiy0S8Yri#scrollTo=BVv7Pux6LdpH

Day5_HandwrittenDigitRecognition_SVM.ipynb

File Edit View Insert Runtime Tools Help All changes saved

+ Code + Text

RAM  Disk  Editing

```
model2.fit(X_train,y_train)
model3.fit(X_train,y_train)
model4.fit(X_train,y_train)

y_predModel1 = model1.predict(X_test)
y_predModel2 = model2.predict(X_test)
y_predModel3 = model3.predict(X_test)
y_predModel4 = model4.predict(X_test)

print("Accuracy of the Model 1: {0}%".format(accuracy_score(y_test, y_predModel1)*100))
print("Accuracy of the Model 2: {0}%".format(accuracy_score(y_test, y_predModel2)*100))
print("Accuracy of the Model 3: {0}%".format(accuracy_score(y_test, y_predModel3)*100))
print("Accuracy of the Model 4: {0}%".format(accuracy_score(y_test, y_predModel4)*100))
```

0s completed at 8:12 PM

3_AdSalePrediction.pptx Show all