รายงาน

เรื่อง Classification of Iris Data Set

โดย

07610477	นายศักดิ์ณรงค์	สมบัติเจริญ
620710405	นางสาวณัฐธิดา	ลาภธนชัย
620710407	นางสาวเพชรัตน์	สุขอุบล
620710408	นางสาวสุธาทิพย์	แย้มกลิ่น

เสนอ

อาจารย์ จิตดำรง ปรีชาสุข

รายงานนี้เป็นส่วนหนึ่งของรายวิชา 522151 Foundation of data science

ภาคเรียนที่ 2 ปีการศึกษา 2563

คณะวิทยาศาสตร์ มหาวิทยาลัยศิลปากร

TERM PROJECT
522151-2561 FOUNDATION OF DATA SCIENCE

CLASSIFICATION PROJECT

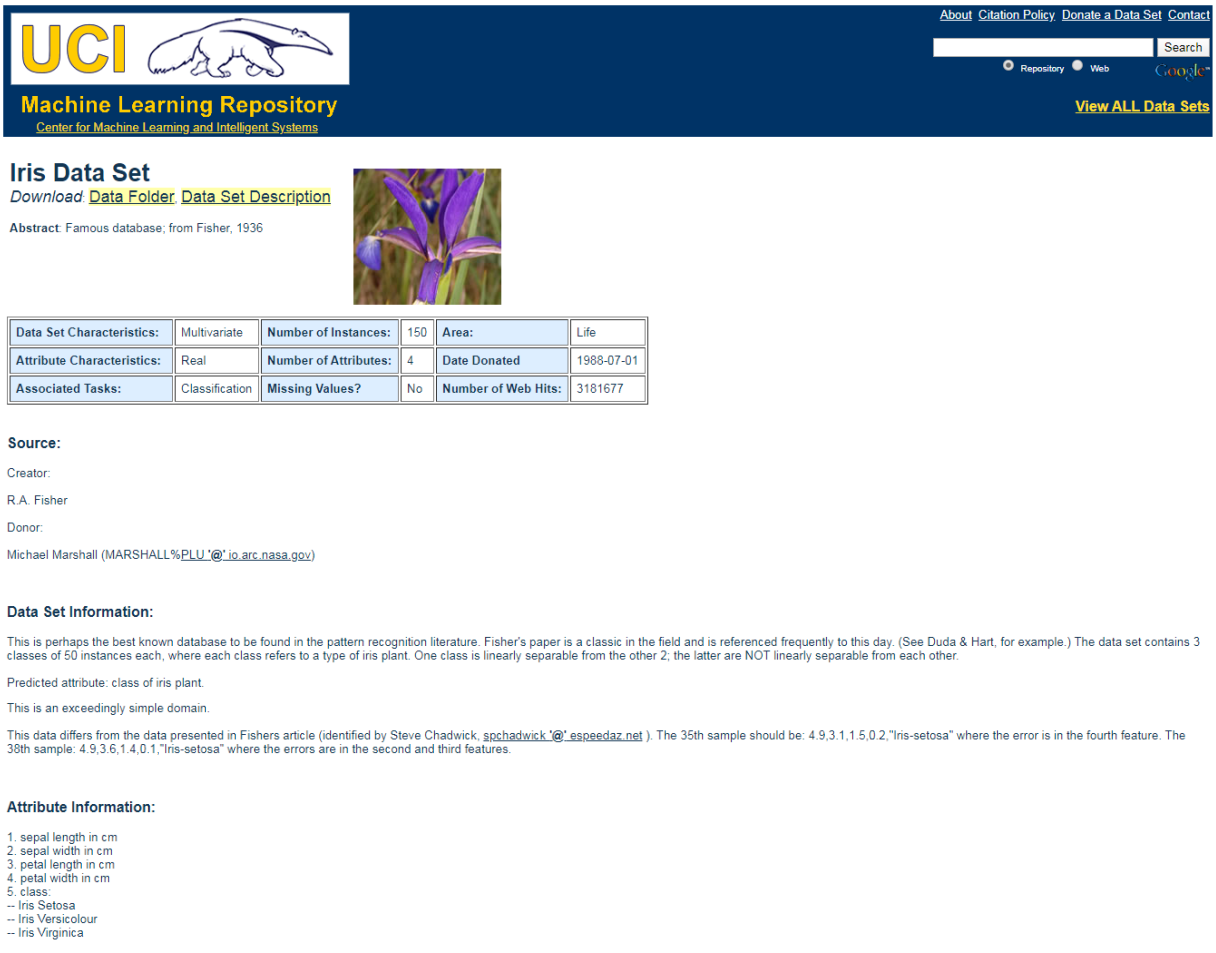
MEMBERS : 07610477 นายศักดิ์ณรงค์ สมบัติเจริญ

620710405 นางสาวณัฐธิดา ลาภธนชัย

620710407 นางสาวเพชรัตน์ สุขอุบล

620710408 นางสาวสุธาทิพย์ แย้มกลิ่น

DATASET : Iris Data Set [https://archive.ics.uci.edu/ml/datasets/Iris]



UCI Machine Learning Repository
Center for Machine Learning and Intelligent Systems

About Citation Policy Donate a Data Set Contact
Search
Repository Web
View ALL Data Sets

Iris Data Set

Download: Data Folder, Data Set Description

Abstract: Famous database; from Fisher, 1936

Data Set Characteristics:	Multivariate	**Number of Instances:**	150	**Area:**	Life
Attribute Characteristics:	Real	**Number of Attributes:**	4	**Date Donated**	1988-07-01
Associated Tasks:	Classification	**Missing Values?**	No	**Number of Web Hits:**	3181677

Source:

Creator:

R.A. Fisher

Donor:

Michael Marshall (MARSHALL%PLU '@' io.arc.nasa.gov)

Data Set Information:

This is perhaps the best known database to be found in the pattern recognition literature. Fisher's paper is a classic in the field and is referenced frequently to this day. (See Duda & Hart, for example.) The data set contains 3 classes of 50 instances each, where each class refers to a type of iris plant. One class is linearly separable from the other 2; the latter are NOT linearly separable from each other.

Predicted attribute: class of iris plant.

This is an exceedingly simple domain.

This data differs from the data presented in Fishers article (identified by Steve Chadwick, spchadwick '@' espeedaz.net). The 35th sample should be: 4.9,3.1,1.5,0.2,"Iris-setosa" where the error is in the fourth feature. The 38th sample: 4.9,3.6,1.4,0.1,"Iris-setosa" where the errors are in the second and third features.

Attribute Information:

1. sepal length in cm
2. sepal width in cm
3. petal length in cm
4. petal width in cm
5. class:
-- Iris Setosa
-- Iris Versicolour
-- Iris Virginica

Dataset Name

- **Attributes Number :** 4
- **Attributes Name :** sepal length, sepal width, petal length, petal width
- **Attributes Characteristics :** Real
- **Associated Tasks :** Classification

- **Attributes Detail :** 1. sepal length in cm
  2. sepal width in cm
  3. petal length in cm
  4. petal width in cm
  5. class:
  -- Iris Setosa
  -- Iris Versicolour
  -- Iris Virginica
- **Class Name :** Iris Setosa (50), Iris Versicolour (50), Iris Virginica (50)
- **Number of instances :** 150

Technique

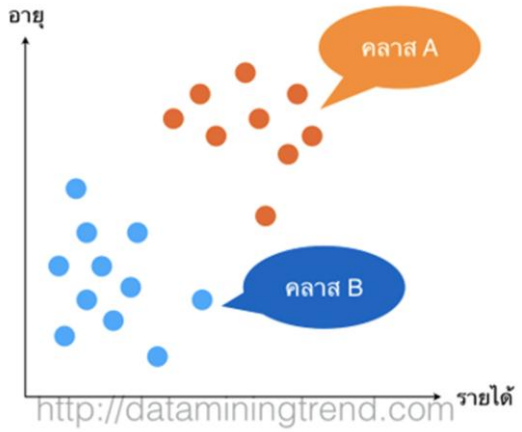
- **Name : SVM (Support Vector Machine)**

เป็นอัลกอริทึมที่สามารถนำมาช่วยแก้ปัญหาการจำแนกข้อมูล ใช้ในการวิเคราะห์ข้อมูลและจำแนกข้อมูล โดยอาศัยหลักการของการหาสัมประสิทธิ์ของสมการเพื่อสร้างเส้นแบ่งแยกกลุ่มข้อมูลที่ถูกป้อนเข้าสู่กระบวนการสอนให้ระบบเรียนรู้ โดยเน้นไปยังเส้นแบ่งแยกแยะกลุ่มข้อมูลได้ดีที่สุด

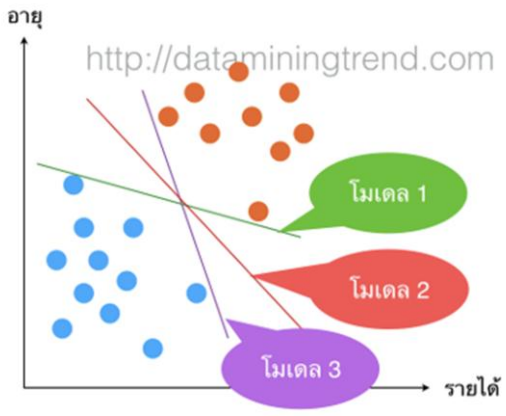
แนวความคิดของ Support Vector Machine

เกิดจากการที่นำค่าของกลุ่มข้อมูลมาวางลงในฟีเจอร์สเปซ (Feature Space) จากนั้นจึงหาเส้นที่ใช้แบ่งข้อมูลทั้งสองออกจากกันโดยจะสร้างเส้นแบ่ง (Hyperplane) ที่เป็นเส้นตรงขึ้นมา และเพื่อให้ทราบว่าเส้นตรงที่แบ่งสองกลุ่มออกจากกันนั้น เส้นตรงใดเป็นเส้นที่ดีที่สุด

สำหรับรากฐานเดิมของ Support Vector Machine ถูกนำมาใช้กับข้อมูลที่เป็นเชิงเส้น แต่ในความเป็นจริงแล้วข้อมูลที่นำมาใช้ในระบบการสอนให้ระบบเรียนรู้ส่วนใหญ่มักเป็นข้อมูลแบบไม่เป็นเชิงเส้น ซึ่งสามารถแก้ปัญหาดังกล่าวด้วยการนำ Kernal Function มาใช้



รูปที่ 1



รูปที่ 2

สมมติว่าต้องการคัดแยกข้อมูลออกเป็น 2 กลุ่ม โดยใช้เส้นแบ่งที่เป็นเส้นตรง จะเห็นว่ามีเส้นตรงจำนวนมากที่สามารถคัดแยกได้ แต่เส้นตรงเส้นไหนที่ดีที่สุด เราจะนิยาม Margin เป็นผลรวมระยะห่างของเส้นตรงที่เป็นเส้นแบ่ง ถึงเส้นตรงที่ผ่านข้อมูลที่ใกล้ที่สุดและขนานกับเส้นแบ่งของทั้งสองกลุ่ม จะเห็นว่า H1 แม้จะสามารถแบ่งข้อมูลทั้งสองกลุ่มออกได้เช่นกัน แต่ ระยะในการแบ่งจากเส้นแบ่งไปถึงข้อมูลที่ใกล้ที่สุดนั้นมีขนาดน้อย แต่จากเส้น H2 จะเป็นเส้นที่แบ่งกลุ่มที่กว้างมากที่สุดของทั้งสองกลุ่มคือให้ค่า maximum margin เราเรียกข้อมูลที่อยู่บน margin นี้ว่า Support Vector จากการกระจายตัวของข้อมูลในรูปที่ 1 จะเห็นว่าสามารถแบ่งแยกออกเป็น 2 กลุ่มได้อย่างชัดเจน ซึ่งโดยปกติแล้วเราจะใช้ linear model (สมการเส้นตรง) เพื่อทำการแบ่งข้อมูลออกเป็น 2 คลาส ทว่า linear model นี้สามารถเป็นไปได้หลากหลายเส้นดังใน รูปที่ 2

- **Command name of technique used in R :**

```
> data_iris <- iris
> str(data_iris)
        'data.frame':           150 obs. of  5 variables:
         $ Sepal.Length: num  5.1 4.9 4.7 4.6 5 5.4 4.6 5 4.4 4.9 ...
         $ Sepal.Width : num  3.5 3 3.2 3.1 3.6 3.9 3.4 3.4 2.9 3.1 ...
         $ Petal.Length: num  1.4 1.4 1.3 1.5 1.4 1.7 1.4 1.5 1.4 1.5 ...
         $ Petal.Width : num  0.2 0.2 0.2 0.2 0.2 0.4 0.3 0.2 0.2 0.1 ...
         $ Species     : Factor w/ 3 levels "setosa","versicolor",..: 1 1 1 1 1 1 1 1 1 1 ...
> summary(data_iris)
          Sepal.Length       Sepal.Width        Petal.Length       Petal.Width          Species
        Min.   :4.300   Min.   :2.000   Min.   :1.000   Min.   :0.100   setosa     :50
        1st Qu.:5.100   1st Qu.:2.800   1st Qu.:1.600   1st Qu.:0.300   versicolor :50
        Median :5.800   Median :3.000   Median :4.350   Median :1.300   virginica  :50
        Mean   :5.843   Mean   :3.057   Mean   :3.758   Mean   :1.199
        3rd Qu.:6.400   3rd Qu.:3.300   3rd Qu.:5.100   3rd Qu.:1.800
        Max.   :7.900   Max.   :4.400   Max.   :6.900   Max.   :2.500

> install.packages(“caTools”)
> library(caTools)
> set.seed(123)
> data_split <- sample.split(data_iris$Species,SplitRatio = 0.7)
> training_set <- subset(data_iris,data_split==TRUE)
          Sepal.Length   Sepal.Width   Petal.Length   Petal.Width   Species
      1   5.1            3.5           1.4            0.2           setosa
      3   4.7            3.2           1.3            0.2           setosa
      6   5.4            3.9           1.7            0.4           setosa
      7   4.6            3.4           1.4            0.3           setosa
      9   4.4            2.9           1.4            0.2           setosa
      10  4.9            3.1           1.5            0.1           setosa
```

```
> test_set <- subset(data_iris,data_split==FALSE)
```

	Sepal.Length	Sepal.Width	Petal.Length	Petal.Width	Species
2	4.9	3.0	1.4	0.2	setosa
4	4.6	3.1	1.5	0.2	setosa
5	5.0	3.6	1.4	0.2	setosa
8	5.0	3.4	1.5	0.2	setosa
11	5.4	3.7	1.5	0.2	setosa
16	5.7	4.4	1.5	0.4	setosa

```
> nrow(training_set) #105 from 150 (70%)
      [1] 105
> nrow(test_set) #45 from 150 (30%)
      [1] 45
> training_set[,1:4] = scale(training_set[,1:4])
> training_set[,1:4]
```

	Sepal.Length	Sepal.Width	Petal.Length	Petal.Width
1	-0.8671874	1.1068187	-1.323982	-1.300121
3	-1.3414306	0.3909093	-1.380494	-1.300121
6	-0.5115051	2.0613647	-1.154448	-1.039849
7	-1.4599914	0.8681822	-1.323982	-1.169985
9	-1.6971129	-0.3250002	-1.323982	-1.300121
10	-1.1043090	0.1522728	-1.267471	-1.430257

```
> test_set[,1:4] = scale(test_set[,1:4])
> test_set[,1:4]
```

	Sepal.Length	Sepal.Width	Petal.Length	Petal.Width
2	-1.2144832	-0.22500949	-1.348549	-1.322425
4	-1.5896669	-0.01406309	-1.292203	-1.322425
5	-1.0894220	1.04066889	-1.348549	-1.322425
8	-1.0894220	0.61877609	-1.292203	-1.322425
11	-0.5891772	1.25161528	-1.292203	-1.322425
16	-0.2139936	2.72824005	-1.292203	-1.057940

```
> install.packages("e1071")
> library(e1071)
> mymodel <- svm(Species~., data = iris)
> mymodel
      Call : svm(formula = Species ~ ., data = iris)
      Parameters :
       SVM-Type  :  C-classification
       SVM-Kernel  :  radial
            cost :  1
      Number of Support Vectors :  51
```

```
> classifier1 = svm(formula = Species~., data = training_set, type = 'C-classification', kernel = 'radial')
> classifier1
        Call : svm(formula = Species ~ ., data = training_set, type = "C-classification", kernel = "radial")
        Parameters :

        SVM-Type   :  C-classification
        SVM-Kernel :  radial
              cost :  1
        Number of Support Vectors:  40
> classifier2 = svm(formula = Species~ Petal.Width + Petal.Length, data = training_set,
                            type = 'C-classification', kernel = 'radial')
> classifier2
        Call : svm(formula = Species ~ Petal.Width + Petal.Length, data = training_set,
                            type = "C-classification", kernel = "radial")
        Parameters :
        SVM-Type   :  C-classification
        SVM-Kernel  :  radial
              cost :  1
        Number of Support Vectors:  29
> test_pred1 = predict(classifier1, type = 'response', newdata = test_set[-5])
> head(test_pred1)
           2        4        5        8        11       16
        setosa  setosa  setosa  setosa  setosa  setosa
        Levels: setosa  versicolor  virginica
> test_pred2 = predict(classifier2, type = 'response', newdata = test_set[-5])
> head(test_pred2)
           2        4        5        8        11       16
        setosa  setosa  setosa  setosa  setosa  setosa
        Levels: setosa  versicolor  virginica
> cm1 = table(test_set[,5], test_pred1)
> cm1
        test_pred1
                        setosa      versicolor      virginica
        setosa          15          0               0
        versicolor      0           13              2
        virginica       0           1               14

> cm2 = table(test_set[,5], test_pred2)
> cm2
         test_pred2
                        setosa      versicolor      virginica
        setosa          15          0               0
        versicolor      0           13              2
        virginica       0           2               13
```

```
> ACC <- sum( diag (cm1) ) / nrow ( test_set )
> ACC
      [1] 0.9333333
```

- **Performance measurement :**

Confusion Matrix of “Test Data (iris)”

		Prediction		
		setosa	versicolor	virginica
Actual	setosa	15	0	0
	versicolor	0	13	2
	virginica	0	1	14

True Position of Model (TP) = $15+13+14 = 42$

Confusion Matrix of “setosa”

TP=15

		Prediction		
		setosa	versicolor	virginica
Actual	setosa	15	0	0
	versicolor	0	13	2
	virginica	0	1	14

TN=15

		Prediction		
		setosa	versicolor	virginica
Actual	setosa	15	0	0
	versicolor	0	13	2
	virginica	0	1	14

FP=0

		Prediction		
		setosa	versicolor	virginica
Actual	setosa	15	0	0
	versicolor	0	13	2
	virginica	0	1	14

FN=0

		Prediction		
		setosa	versicolor	virginica
Actual	setosa	15	0	0
	versicolor	0	13	2
	virginica	0	1	14

Precision = $\frac{15}{15+0+0} = 1$ Recall = $\frac{15}{15+0+0} = 1$

Confusion Matrix of “versicolor”

TP=13

Actual \ Prediction	setosa	versicolor	virginica
setosa	15	0	0
versicolor	0	13	2
virginica	0	1	14

TN=16

Actual \ Prediction	setosa	versicolor	virginica
setosa	15	0	0
versicolor	0	13	2
virginica	0	1	14

FP=1

Actual \ Prediction	setosa	versicolor	virginica
setosa	15	0	0
versicolor	0	13	2
virginica	0	1	14

FN=2

Actual \ Prediction	setosa	versicolor	virginica
setosa	15	0	0
versicolor	0	13	2
virginica	0	1	14

$$\text{Precision} = \frac{\mathbf{13}}{\mathbf{13+1+0}} = \mathbf{0.929} \qquad \text{Recall} = \frac{\mathbf{13}}{\mathbf{13+2+0}} = \mathbf{0.867}$$

Confusion Matrix of “virginica”

TP=14

Actual \ Prediction	setosa	versicolor	virginica
setosa	15	0	0
versicolor	0	13	2
virginica	0	1	14

TN=17

Actual \ Prediction	setosa	versicolor	virginica
setosa	15	0	0
versicolor	0	13	2
virginica	0	1	14

FP=2

Actual \ Prediction	setosa	versicolor	virginica
setosa	15	0	0
versicolor	0	13	2
virginica	0	1	14

FN=1

Actual \ Prediction	setosa	versicolor	virginica
setosa	15	0	0
versicolor	0	13	2
virginica	0	1	14

$$\text{Precision} = \frac{\mathbf{14}}{\mathbf{14+2+0}} = \mathbf{0.875} \qquad \text{Recall} = \frac{\mathbf{14}}{\mathbf{14+1+0}} = \mathbf{0.933}$$