SVM Example 2

Abhirup Sen

24/05/2021

Get the dataset

```
dataset = read.csv('Social_Network_Ads.csv')
dataset = dataset[3:5]
```

```
dataset$Purchased = factor(dataset$Purchased, levels = c(0,1))
```

Splitting the dataset into the traning set and test set

```
library(e1071)
library(caTools)
set.seed(123)
split = sample.split(dataset$Purchased, SplitRatio = 0.75)
training_set = subset(dataset, split == TRUE)
test_set = subset(dataset, split == FALSE)
```

Features Sclaing

```
training_set[-3] = scale(training_set[-3])
test_set[-3] = scale(test_set[-3])
```

Fitting SVM to the Traning Set

```
classifier = svm(formula = Purchased ~., data = training_set,
                type = 'C-classification',
                kernel='linear')
summary(classifier)
```

```
##
## Call:
## svm(formula = Purchased ~ ., data = training_set, type = "C-classification",
##     kernel = "linear")
##
##
## Parameters:
##    SVM-Type:  C-classification
##  SVM-Kernel:  linear
##        cost:  1
##
```

```
## Number of Support Vectors:  116
##
##  ( 58 58 )
##
##
## Number of Classes:  2
##
## Levels:
##  0 1
```

Predict the test set result

```
y_pred = predict(classifier, newdata = test_set[-3])
```

Making the confusion matrix

```
cm = table(test_set[,3],y_pred)
library(caret)
```

```
## Loading required package: lattice
```

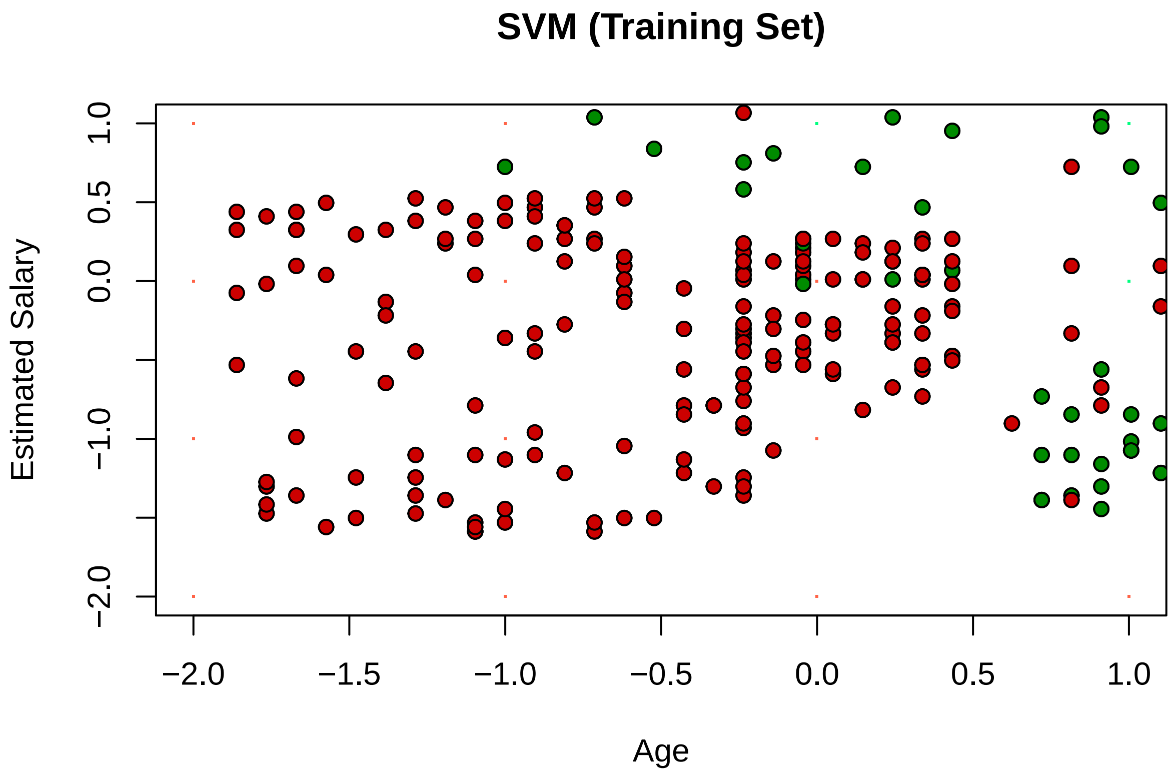
```
## Loading required package: ggplot2
```

```
confusionMatrix(cm)
```

```
## Confusion Matrix and Statistics
##
##    y_pred
##      0  1
##   0 57  7
##   1 13 23
##
##                Accuracy : 0.8
##                  95% CI : (0.7082, 0.8733)
##     No Information Rate : 0.7
##     P-Value [Acc > NIR] : 0.01646
##
##                   Kappa : 0.5495
##
##  Mcnemar’s Test P-Value : 0.26355
##
##             Sensitivity : 0.8143
##             Specificity : 0.7667
##          Pos Pred Value : 0.8906
##          Neg Pred Value : 0.6389
##              Prevalence : 0.7000
##          Detection Rate : 0.5700
##    Detection Prevalence : 0.6400
##       Balanced Accuracy : 0.7905
##
##        ’Positive’ Class : 0
##
```

visualising the traning set results

```
set = training_set
x1 = seq(min(set[,1]-1,max(set[,1])+1, by = 0.01))
x2 = seq(min(set[,2]-1,max(set[,2])+1, by = 0.01))
grid_set = expand.grid(x1,x2)
colnames(grid_set) = c('Age', 'EstimatedSalary')
y_grid = predict(classifier, newdata = grid_set)
plot(set[,-3],
     main ='SVM (Training Set)',
     xlab = 'Age', ylab ='Estimated Salary',
     xlim =range(x1), ylim=range(x2))
#contour(x1,x2, matrix(as.numeric(y_grid),length(x1),length(x2)), add = TRUE)
points(grid_set, pch ='.',col = ifelse(y_grid==1, 'springgreen','tomato'))
points(set, pch = 21, bg = ifelse(set[,3]==1,'green4','red3'))
```


SVM (Training Set)
Estimated Salary
Age