Data Mining HW4

Construct support vector machine classifiers for MNIST\_small data. Report the best tuning parameters and what kernel works the best. Make a confusion matrix and provide your conclusions and discussion.

##### load dataset  
  
mnist\_train = read.csv('./MNIST\_train\_small.csv', header=TRUE)  
mnist\_test = read.csv('./MNIST\_test\_small.csv', header=TRUE)  
  
x\_train = mnist\_train[, 2:785]  
y\_train = as.factor(mnist\_train[, 1])  
x\_test = mnist\_test[, 2:785]  
y\_test = as.factor(mnist\_test[, 1])  
  
student = 20152410

##### construct support vector machine classifier  
  
library(e1071)

## Warning: package 'e1071' was built under R version 3.6.3

library(caret)

## Warning: package 'caret' was built under R version 3.6.3

## Loading required package: lattice

## Loading required package: ggplot2

library(kernlab)

##   
## Attaching package: 'kernlab'

## The following object is masked from 'package:ggplot2':  
##   
## alpha

method\_list = c('svmLinear', 'svmPoly', 'svmRadial')  
fold\_number = 3  
tune\_length = 10  
train\_control = trainControl(method='cv',  
 number=fold\_number,  
 search='random')

### linear svm  
  
start = Sys.time()  
  
set.seed(student)  
linear\_model = train(x\_train,  
 y\_train,  
 method=method\_list[1],  
 trControl=train\_control,  
 metric='accuracy',  
 tuneLength=tune\_length)

## Warning in train.default(x\_train, y\_train, method = method\_list[1], trControl =  
## train\_control, : The metric "accuracy" was not in the result set. Accuracy will  
## be used instead.

## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.  
  
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.  
  
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.  
  
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.  
  
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.  
  
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.  
  
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.  
  
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.  
  
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.  
  
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.  
  
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.  
  
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.  
  
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.  
  
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.  
  
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.  
  
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.  
  
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.  
  
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.  
  
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.  
  
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.  
  
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.  
  
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.  
  
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.  
  
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.  
  
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.  
  
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.  
  
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.  
  
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.  
  
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.  
  
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.  
  
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.

load\_time = Sys.time() - start  
load\_time

## Time difference of 5.971888 mins

linear\_model

## Support Vector Machines with Linear Kernel   
##   
## 6000 samples  
## 784 predictor  
## 10 classes: '0', '1', '2', '3', '4', '5', '6', '7', '8', '9'   
##   
## No pre-processing  
## Resampling: Cross-Validated (3 fold)   
## Summary of sample sizes: 3999, 4001, 4000   
## Resampling results across tuning parameters:  
##   
## C Accuracy Kappa   
## 0.1101521 0.9186653 0.9095505  
## 0.2812232 0.9116661 0.9017667  
## 2.6407019 0.9050007 0.8943553  
## 8.8305640 0.9050007 0.8943553  
## 36.4625175 0.9050007 0.8943553  
## 53.6442018 0.9050007 0.8943553  
## 75.7921019 0.9050007 0.8943553  
## 228.4851141 0.9050007 0.8943553  
## 433.6071618 0.9050007 0.8943553  
## 551.1111075 0.9050007 0.8943553  
##   
## Accuracy was used to select the optimal model using the largest value.  
## The final value used for the model was C = 0.1101521.

linear\_pred = predict(linear\_model, newdata=x\_test)  
linear\_table = table(linear\_pred, y\_test)  
linear\_cm = confusionMatrix(linear\_table, mode='everything')  
linear\_cm

## Confusion Matrix and Statistics  
##   
## y\_test  
## linear\_pred 0 1 2 3 4 5 6 7 8 9  
## 0 93 0 3 0 0 2 2 1 0 1  
## 1 0 106 2 0 2 0 0 0 0 2  
## 2 0 0 100 2 1 2 0 1 1 0  
## 3 0 0 1 77 0 3 0 0 4 0  
## 4 0 0 1 0 100 0 3 1 2 3  
## 5 1 0 0 2 1 85 1 0 2 2  
## 6 0 0 0 1 0 2 93 0 2 0  
## 7 0 0 0 1 0 0 0 102 2 2  
## 8 0 0 4 2 0 1 0 1 77 0  
## 9 0 0 0 1 4 0 0 2 0 93  
##   
## Overall Statistics  
##   
## Accuracy : 0.926   
## 95% CI : (0.908, 0.9415)  
## No Information Rate : 0.111   
## P-Value [Acc > NIR] : < 2.2e-16   
##   
## Kappa : 0.9177   
##   
## Mcnemar's Test P-Value : NA   
##   
## Statistics by Class:  
##   
## Class: 0 Class: 1 Class: 2 Class: 3 Class: 4 Class: 5  
## Sensitivity 0.9894 1.0000 0.9009 0.8953 0.9259 0.8947  
## Specificity 0.9901 0.9933 0.9921 0.9912 0.9888 0.9901  
## Pos Pred Value 0.9118 0.9464 0.9346 0.9059 0.9091 0.9043  
## Neg Pred Value 0.9989 1.0000 0.9877 0.9902 0.9910 0.9890  
## Precision 0.9118 0.9464 0.9346 0.9059 0.9091 0.9043  
## Recall 0.9894 1.0000 0.9009 0.8953 0.9259 0.8947  
## F1 0.9490 0.9725 0.9174 0.9006 0.9174 0.8995  
## Prevalence 0.0940 0.1060 0.1110 0.0860 0.1080 0.0950  
## Detection Rate 0.0930 0.1060 0.1000 0.0770 0.1000 0.0850  
## Detection Prevalence 0.1020 0.1120 0.1070 0.0850 0.1100 0.0940  
## Balanced Accuracy 0.9897 0.9966 0.9465 0.9433 0.9574 0.9424  
## Class: 6 Class: 7 Class: 8 Class: 9  
## Sensitivity 0.9394 0.9444 0.8556 0.9029  
## Specificity 0.9945 0.9944 0.9912 0.9922  
## Pos Pred Value 0.9490 0.9533 0.9059 0.9300  
## Neg Pred Value 0.9933 0.9933 0.9858 0.9889  
## Precision 0.9490 0.9533 0.9059 0.9300  
## Recall 0.9394 0.9444 0.8556 0.9029  
## F1 0.9442 0.9488 0.8800 0.9163  
## Prevalence 0.0990 0.1080 0.0900 0.1030  
## Detection Rate 0.0930 0.1020 0.0770 0.0930  
## Detection Prevalence 0.0980 0.1070 0.0850 0.1000  
## Balanced Accuracy 0.9669 0.9694 0.9234 0.9476

### polynomial svm  
  
start = Sys.time()  
  
set.seed(student)  
ploy\_model = train(x\_train,  
 y\_train,  
 method=method\_list[2],  
 trControl=train\_control,  
 metric='accuracy',  
 tuneLength=tune\_length)

## Warning in train.default(x\_train, y\_train, method = method\_list[2], trControl =  
## train\_control, : The metric "accuracy" was not in the result set. Accuracy will  
## be used instead.

## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.  
  
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.  
  
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.  
  
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.  
  
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.  
  
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.  
  
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.  
  
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.  
  
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.  
  
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.  
  
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.  
  
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.  
  
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.  
  
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.  
  
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.  
  
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.  
  
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.  
  
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.  
  
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.  
  
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.  
  
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.  
  
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.  
  
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.  
  
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.  
  
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.  
  
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.  
  
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.  
  
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.  
  
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.  
  
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.  
  
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.

load\_time = Sys.time() - start  
load\_time

## Time difference of 42.96775 mins

ploy\_model

## Support Vector Machines with Polynomial Kernel   
##   
## 6000 samples  
## 784 predictor  
## 10 classes: '0', '1', '2', '3', '4', '5', '6', '7', '8', '9'   
##   
## No pre-processing  
## Resampling: Cross-Validated (3 fold)   
## Summary of sample sizes: 3999, 4001, 4000   
## Resampling results across tuning parameters:  
##   
## degree scale C Accuracy Kappa   
## 2 1.737694e-05 2.32211366 0.3176675 0.2304178  
## 2 1.966788e-02 4.75246581 0.9389992 0.9321668  
## 2 2.287321e-02 0.04064384 0.9178332 0.9086216  
## 2 2.720802e-02 200.76500754 0.9408326 0.9342062  
## 2 3.332947e-02 0.22875819 0.9426657 0.9362414  
## 3 5.290869e-05 0.96647039 0.6203325 0.5759114  
## 3 1.219221e-04 2.11694269 0.8718316 0.8573970  
## 3 1.697705e-03 0.06655475 0.8289986 0.8096041  
## 3 8.564345e-02 0.59816536 0.9461656 0.9401350  
## 3 1.587170e-01 187.67195635 0.9456660 0.9395794  
##   
## Accuracy was used to select the optimal model using the largest value.  
## The final values used for the model were degree = 3, scale = 0.08564345 and C  
## = 0.5981654.

ploy\_pred = predict(ploy\_model, newdata=x\_test)  
ploy\_table = table(ploy\_pred, y\_test)  
ploy\_cm = confusionMatrix(ploy\_table, mode='everything')  
ploy\_cm

## Confusion Matrix and Statistics  
##   
## y\_test  
## ploy\_pred 0 1 2 3 4 5 6 7 8 9  
## 0 93 0 2 0 0 3 1 1 0 1  
## 1 0 106 1 0 2 0 0 0 0 2  
## 2 0 0 105 2 0 1 0 2 0 0  
## 3 0 0 1 78 0 1 0 0 2 0  
## 4 0 0 0 0 103 0 1 0 0 1  
## 5 1 0 0 1 1 85 1 0 0 1  
## 6 0 0 0 1 0 1 96 0 0 0  
## 7 0 0 1 1 0 0 0 102 2 1  
## 8 0 0 1 3 0 2 0 1 86 0  
## 9 0 0 0 0 2 2 0 2 0 97  
##   
## Overall Statistics  
##   
## Accuracy : 0.951   
## 95% CI : (0.9357, 0.9635)  
## No Information Rate : 0.111   
## P-Value [Acc > NIR] : < 2.2e-16   
##   
## Kappa : 0.9455   
##   
## Mcnemar's Test P-Value : NA   
##   
## Statistics by Class:  
##   
## Class: 0 Class: 1 Class: 2 Class: 3 Class: 4 Class: 5  
## Sensitivity 0.9894 1.0000 0.9459 0.9070 0.9537 0.8947  
## Specificity 0.9912 0.9944 0.9944 0.9956 0.9978 0.9945  
## Pos Pred Value 0.9208 0.9550 0.9545 0.9512 0.9810 0.9444  
## Neg Pred Value 0.9989 1.0000 0.9933 0.9913 0.9944 0.9890  
## Precision 0.9208 0.9550 0.9545 0.9512 0.9810 0.9444  
## Recall 0.9894 1.0000 0.9459 0.9070 0.9537 0.8947  
## F1 0.9538 0.9770 0.9502 0.9286 0.9671 0.9189  
## Prevalence 0.0940 0.1060 0.1110 0.0860 0.1080 0.0950  
## Detection Rate 0.0930 0.1060 0.1050 0.0780 0.1030 0.0850  
## Detection Prevalence 0.1010 0.1110 0.1100 0.0820 0.1050 0.0900  
## Balanced Accuracy 0.9903 0.9972 0.9702 0.9513 0.9757 0.9446  
## Class: 6 Class: 7 Class: 8 Class: 9  
## Sensitivity 0.9697 0.9444 0.9556 0.9417  
## Specificity 0.9978 0.9944 0.9923 0.9933  
## Pos Pred Value 0.9796 0.9533 0.9247 0.9417  
## Neg Pred Value 0.9967 0.9933 0.9956 0.9933  
## Precision 0.9796 0.9533 0.9247 0.9417  
## Recall 0.9697 0.9444 0.9556 0.9417  
## F1 0.9746 0.9488 0.9399 0.9417  
## Prevalence 0.0990 0.1080 0.0900 0.1030  
## Detection Rate 0.0960 0.1020 0.0860 0.0970  
## Detection Prevalence 0.0980 0.1070 0.0930 0.1030  
## Balanced Accuracy 0.9837 0.9694 0.9739 0.9675

### radial svm  
  
start = Sys.time()  
  
set.seed(student)  
radial\_model = train(x\_train,  
 y\_train,  
 method=method\_list[3],  
 trControl=train\_control,  
 metric='accuracy',  
 tuneLength=tune\_length)

## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.

## Warning in train.default(x\_train, y\_train, method = method\_list[3], trControl =  
## train\_control, : The metric "accuracy" was not in the result set. Accuracy will  
## be used instead.

## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.  
  
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.  
  
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.  
  
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.  
  
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.  
  
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.  
  
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.  
  
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.  
  
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.  
  
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.  
  
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.  
  
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.  
  
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.  
  
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.  
  
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.  
  
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.  
  
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.  
  
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.  
  
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.  
  
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.  
  
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.  
  
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.  
  
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.  
  
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.  
  
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.  
  
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.  
  
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.  
  
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.  
  
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.  
  
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.  
  
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.

load\_time = Sys.time() - start  
load\_time

## Time difference of 41.59553 mins

radial\_model

## Support Vector Machines with Radial Basis Function Kernel   
##   
## 6000 samples  
## 784 predictor  
## 10 classes: '0', '1', '2', '3', '4', '5', '6', '7', '8', '9'   
##   
## No pre-processing  
## Resampling: Cross-Validated (3 fold)   
## Summary of sample sizes: 3999, 4001, 4000   
## Resampling results across tuning parameters:  
##   
## sigma C Accuracy Kappa   
## 0.004451774 6.87490728 0.9396661 0.9329064  
## 0.004779173 0.19853618 0.9011652 0.8900698  
## 0.005353820 0.36343771 0.9148328 0.9052872  
## 0.006283408 0.05255669 0.8518305 0.8351070  
## 0.006547637 0.27951279 0.9148326 0.9052857  
## 0.017157958 6.19099659 0.9551656 0.9501470  
## 0.018133226 11.54186541 0.9558322 0.9508878  
## 0.018391833 6.53863380 0.9559988 0.9510729  
## 0.019130171 8.30370240 0.9563322 0.9514437  
## 0.020183007 11.81963185 0.9564990 0.9516292  
##   
## Accuracy was used to select the optimal model using the largest value.  
## The final values used for the model were sigma = 0.02018301 and C = 11.81963.

radial\_pred = predict(radial\_model, newdata=x\_test)  
radial\_table = table(radial\_pred, y\_test)  
radial\_cm = confusionMatrix(radial\_table, mode='everything')  
radial\_cm

## Confusion Matrix and Statistics  
##   
## y\_test  
## radial\_pred 0 1 2 3 4 5 6 7 8 9  
## 0 93 0 2 0 0 1 1 1 0 1  
## 1 0 106 1 0 0 0 0 0 0 1  
## 2 0 0 106 2 0 1 0 1 0 0  
## 3 0 0 1 78 0 1 0 0 1 1  
## 4 0 0 0 0 105 0 1 1 0 1  
## 5 1 0 0 1 0 89 1 0 0 0  
## 6 0 0 0 0 1 1 96 0 0 0  
## 7 0 0 1 1 0 0 0 101 3 1  
## 8 0 0 0 4 0 1 0 1 86 0  
## 9 0 0 0 0 2 1 0 3 0 98  
##   
## Overall Statistics  
##   
## Accuracy : 0.958   
## 95% CI : (0.9436, 0.9696)  
## No Information Rate : 0.111   
## P-Value [Acc > NIR] : < 2.2e-16   
##   
## Kappa : 0.9533   
##   
## Mcnemar's Test P-Value : NA   
##   
## Statistics by Class:  
##   
## Class: 0 Class: 1 Class: 2 Class: 3 Class: 4 Class: 5  
## Sensitivity 0.9894 1.0000 0.9550 0.9070 0.9722 0.9368  
## Specificity 0.9934 0.9978 0.9955 0.9956 0.9966 0.9967  
## Pos Pred Value 0.9394 0.9815 0.9636 0.9512 0.9722 0.9674  
## Neg Pred Value 0.9989 1.0000 0.9944 0.9913 0.9966 0.9934  
## Precision 0.9394 0.9815 0.9636 0.9512 0.9722 0.9674  
## Recall 0.9894 1.0000 0.9550 0.9070 0.9722 0.9368  
## F1 0.9637 0.9907 0.9593 0.9286 0.9722 0.9519  
## Prevalence 0.0940 0.1060 0.1110 0.0860 0.1080 0.0950  
## Detection Rate 0.0930 0.1060 0.1060 0.0780 0.1050 0.0890  
## Detection Prevalence 0.0990 0.1080 0.1100 0.0820 0.1080 0.0920  
## Balanced Accuracy 0.9914 0.9989 0.9752 0.9513 0.9844 0.9668  
## Class: 6 Class: 7 Class: 8 Class: 9  
## Sensitivity 0.9697 0.9352 0.9556 0.9515  
## Specificity 0.9978 0.9933 0.9934 0.9933  
## Pos Pred Value 0.9796 0.9439 0.9348 0.9423  
## Neg Pred Value 0.9967 0.9922 0.9956 0.9944  
## Precision 0.9796 0.9439 0.9348 0.9423  
## Recall 0.9697 0.9352 0.9556 0.9515  
## F1 0.9746 0.9395 0.9451 0.9469  
## Prevalence 0.0990 0.1080 0.0900 0.1030  
## Detection Rate 0.0960 0.1010 0.0860 0.0980  
## Detection Prevalence 0.0980 0.1070 0.0920 0.1040  
## Balanced Accuracy 0.9837 0.9642 0.9745 0.9724