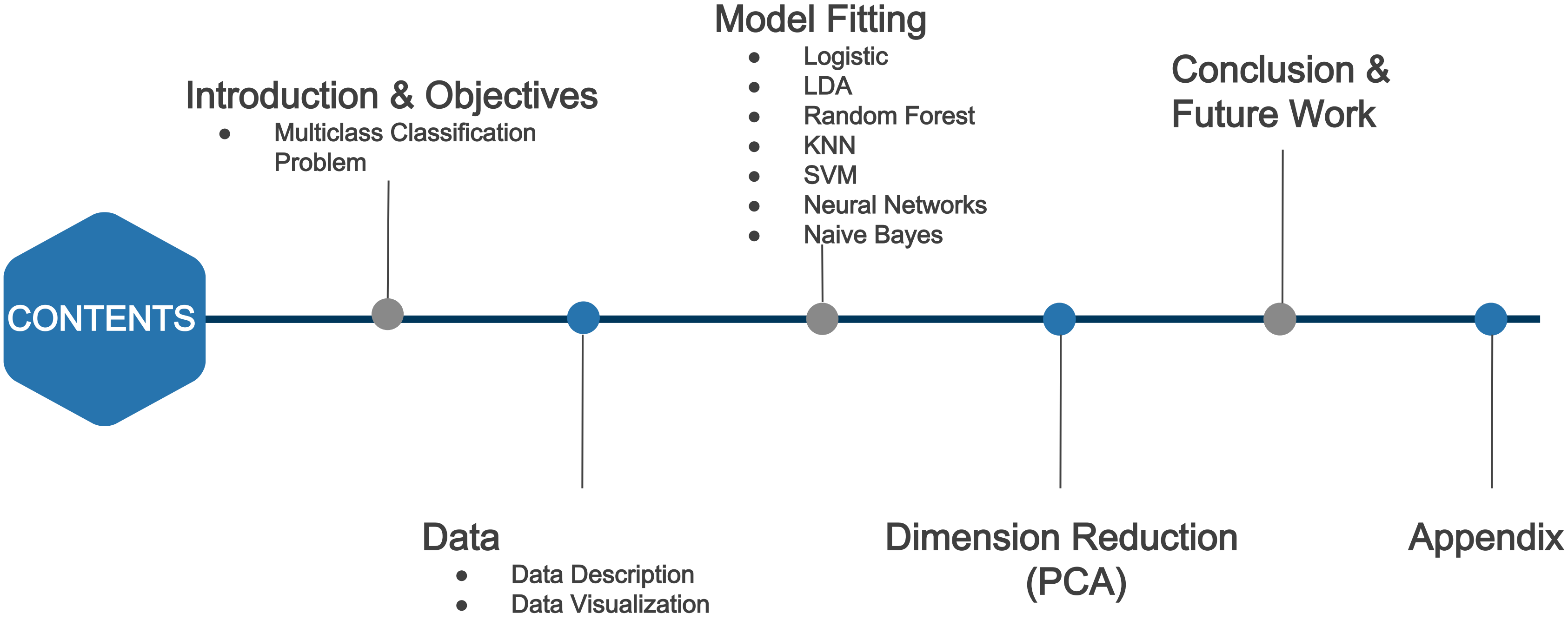
ECON 412 FINAL PROJECT

HUMAN ACTIVITY CLASSIFICATION USING MACHINE LEARNING ALGORITHMS

Roya Latifi
Yiping Liu
Yiran Sun

OUTLINE



CONTENTS

Introduction & Objectives

- Multiclass Classification Problem

Data

- Data Description
- Data Visualization

Model Fitting

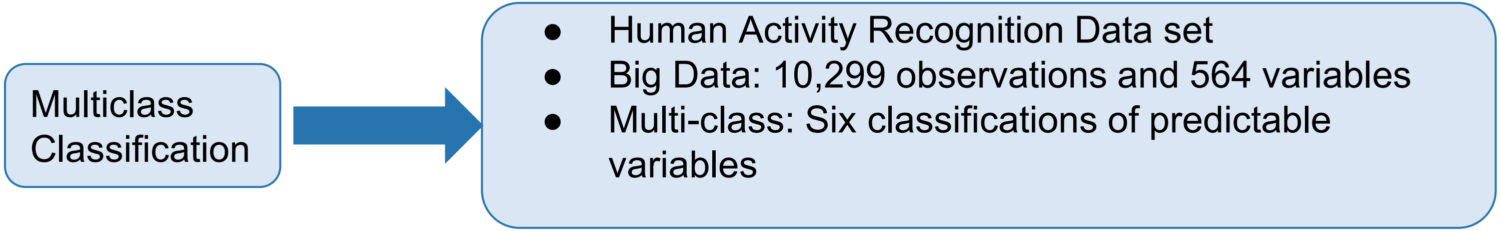
- Logistic
- LDA
- Random Forest
- KNN
- SVM
- Neural Networks
- Naive Bayes

Dimension Reduction (PCA)

Conclusion & Future Work

Appendix

Introduction and Objective



Multiclass Classification

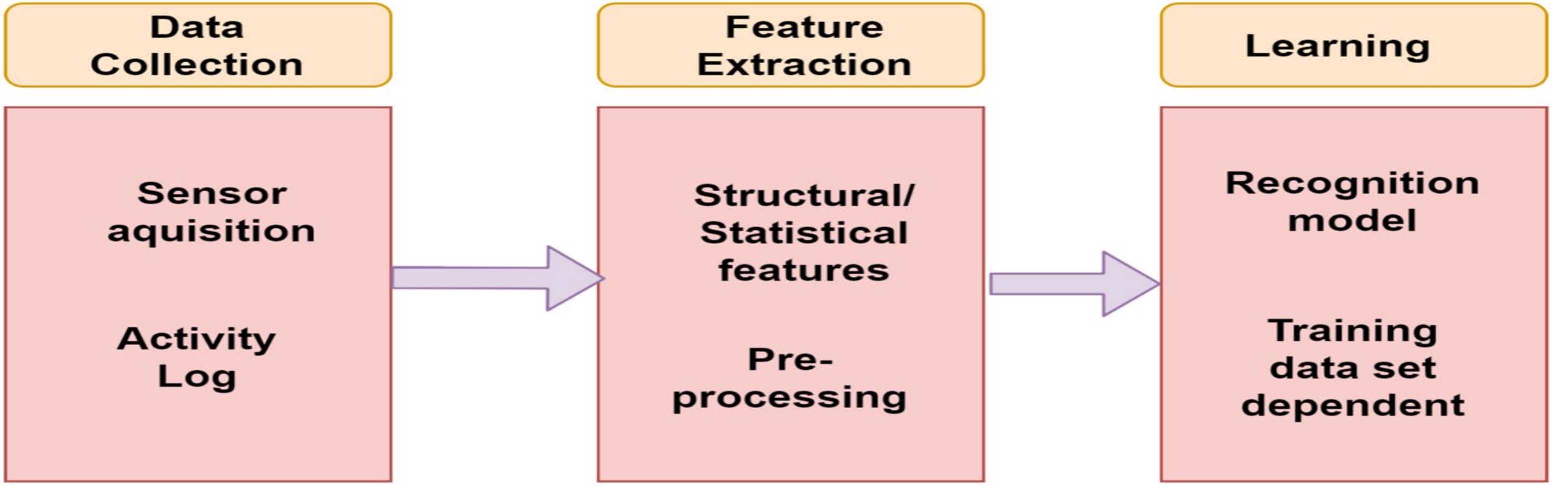
- Human Activity Recognition Data set
- Big Data: 10,299 observations and 564 variables
- Multi-class: Six classifications of predictable variables

❑ The Human Activity Recognition database was built from the recordings of **30 participants** performing activities of daily living (ADL) while carrying a waist-mounted smartphone with embedded inertial sensors.

❑ Our objective is to classify activities into one of the **six activities performed** .

❑ The variables are further calculated from **3-axial linear acceleration and 3 -axial angular velocity** . They are captured by embedded accelerometer and gyroscope at a constant rate of 50Hz in the experiment.

Data Processing



Data Collection

Sensor aquisition

Activity Log

Feature Extraction

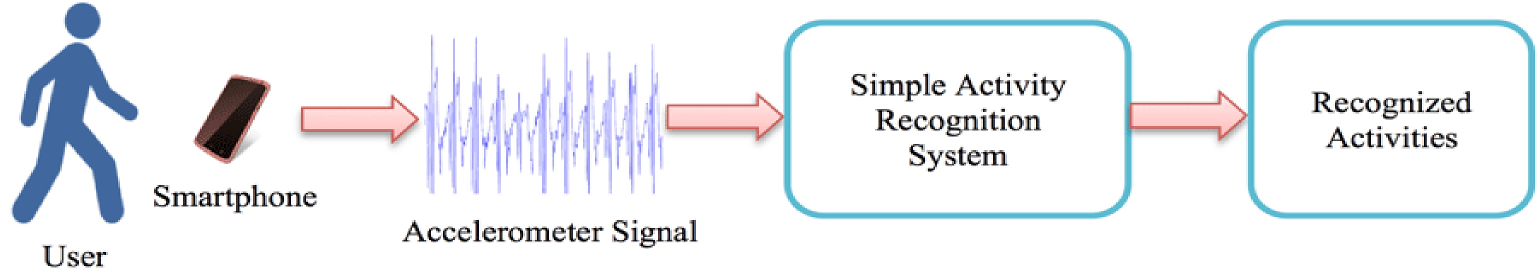
Structural/ Statistical features

Pre-processing

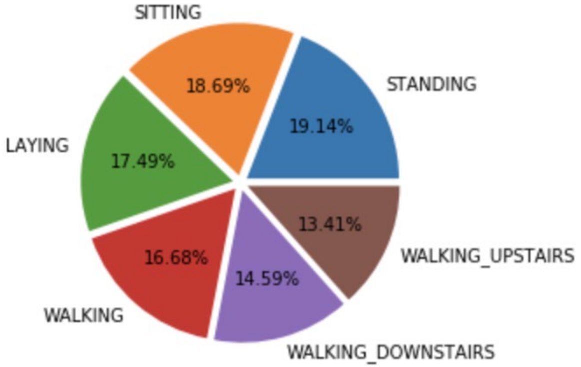
Learning

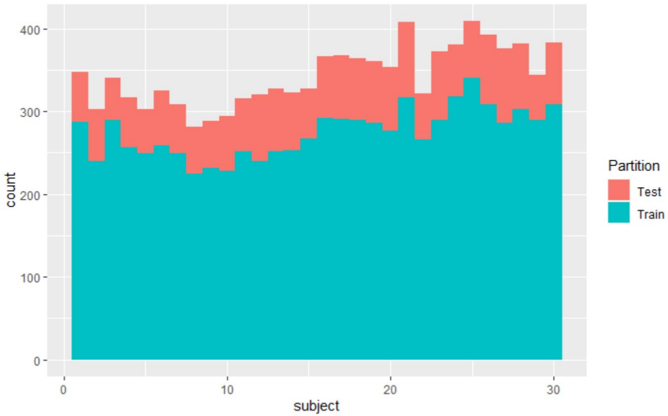
Recognition model

Training data set dependent

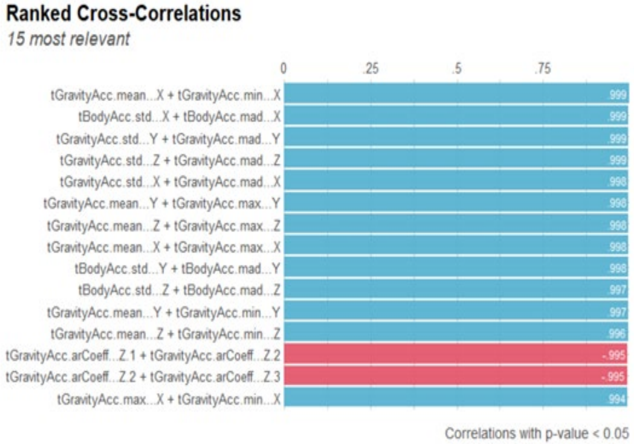


Data Visualization





- ❖ **Data Checking and Processing** :
  i. No missing data in the database
  ii. Independent variables distributed evenly
  iii. splitting training and validation data randomly
- ❖ Follow experiment steps in grouping variables; **Visualize correlation in each group**



correlation plot for raw signals

Model Results Summary

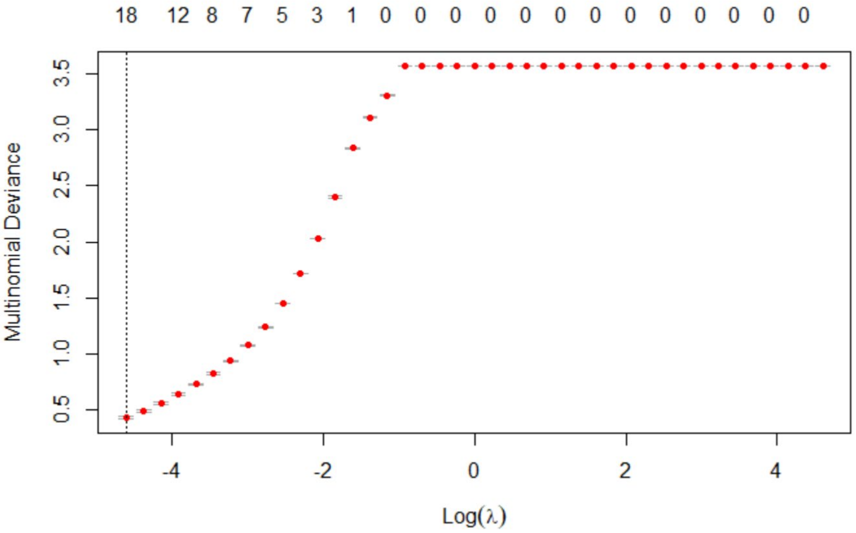
Models	All Features Accuracy (Training Sample)	All Features Accuracy (Testing Sample)	PCA Features Accuracy
Logistic Regression	(L1 Penalty) 96.24%	95.97%	95.39%
	(L2 Penalty) 98.19%	97.43%	95.58%
LDA	97.91%	98.20%	93.46%
Random Forest	100%	97.86%	94.32%
Naive Bayes	77.63%	76.70%	85.34%
KNN	96.46%	96.02%	95.97%
SVM	99.47%	98.50%	95.73%
Neural Networks	99.30%	97.04%	94.51%

Best Performing Model

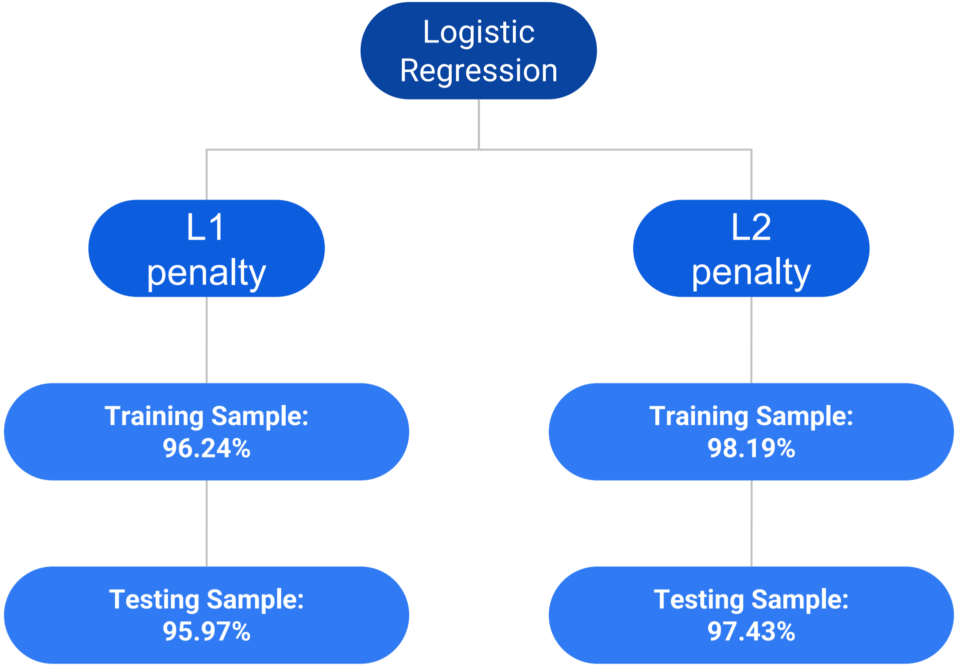
PCA Best Model

Logistic Regression

★ Best lambda selection



➢ Different Alpha Setting



Logistic Regression

L1 penalty

L2 penalty

Training Sample: 96.24%

Testing Sample: 95.97%

Training Sample: 98.19%

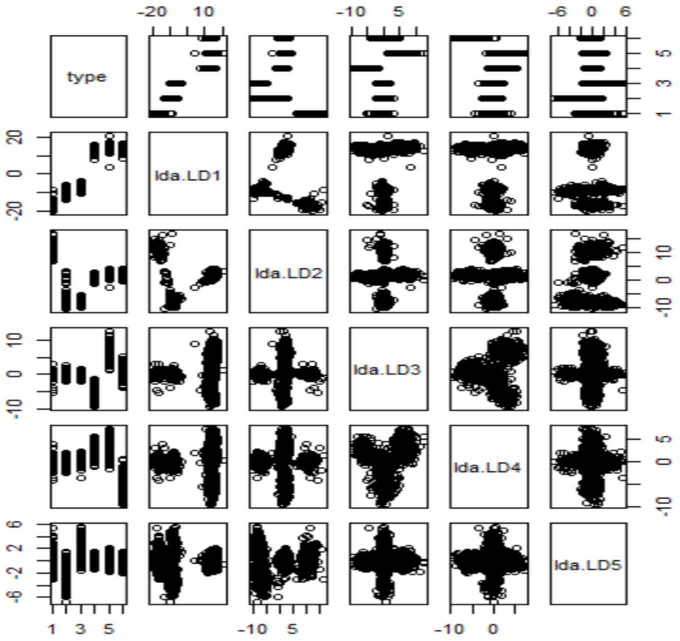
Testing Sample: 97.43%

LDA

Data Set	Accuracy
Training	97.91%
Testing	98.20%

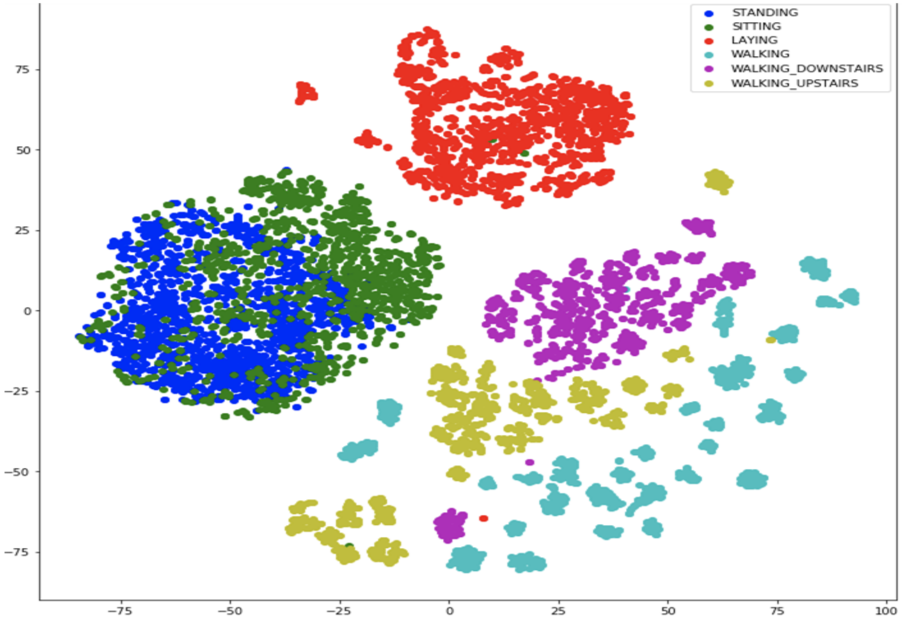
- ➔ High Accuracy in training and testing data set
- ➔ The accuracy drops in cross validation and also the PCA regularization in the next step
- ➔ Meet Collinearity Problem

❖ 5-Fold Cross Validation

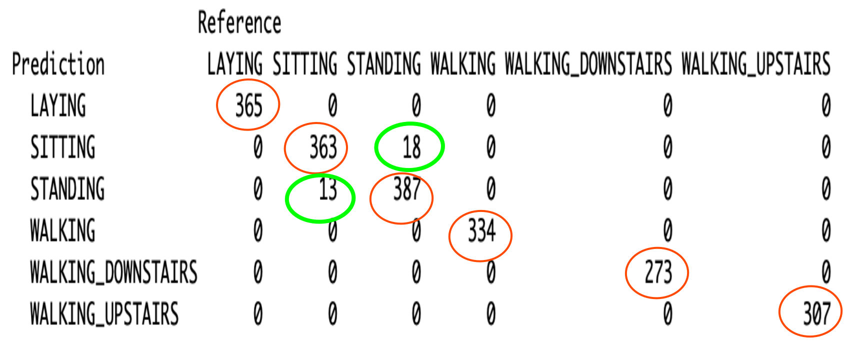


SVM

Multiclass Classification:



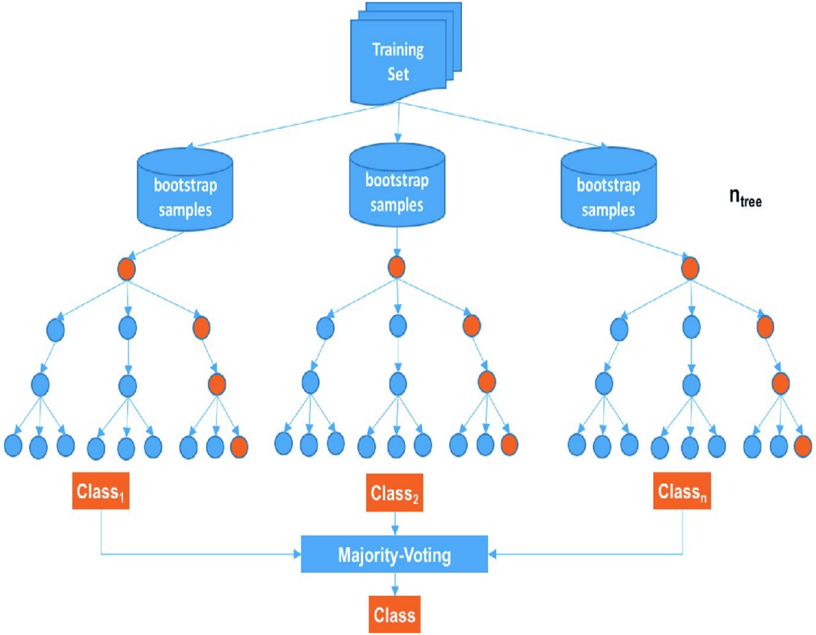
Confusion Matrix:



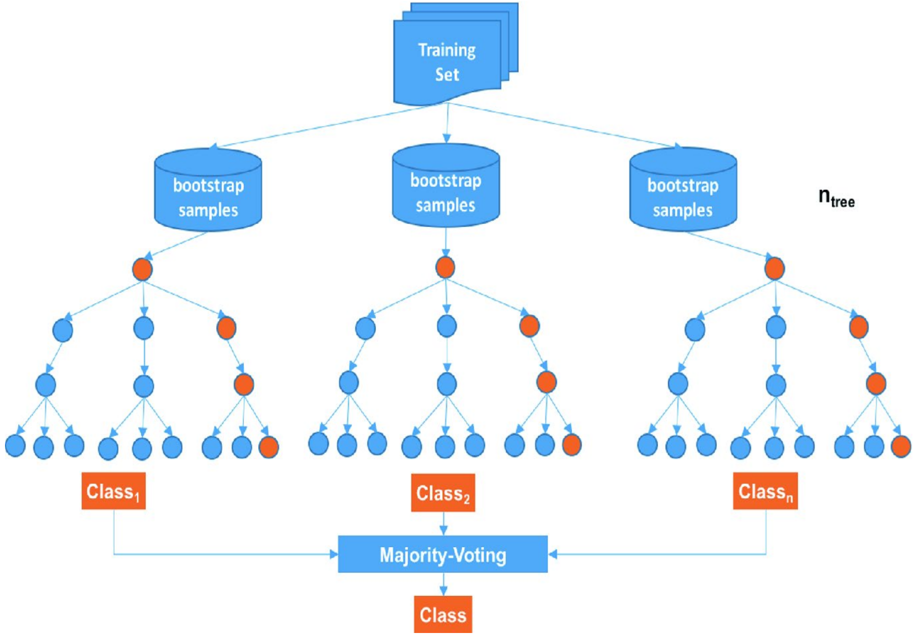
```
                    Reference
Prediction           LAYING SITTING STANDING WALKING WALKING_DOWNSTAIRS WALKING_UPSTAIRS
  LAYING                365       0        0       0                  0                0
  SITTING                 0     363       18       0                  0                0
  STANDING                0      13      387       0                  0                0
  WALKING                 0       0        0     334                  0                0
  WALKING_DOWNSTAIRS      0       0        0       0                273                0
  WALKING_UPSTAIRS        0       0        0       0                  0              307
```

- CV Accuracy: 98.33 %
- Training Sample Accuracy: 99.47 %
- Testing Sample Accuracy: 98.50 %

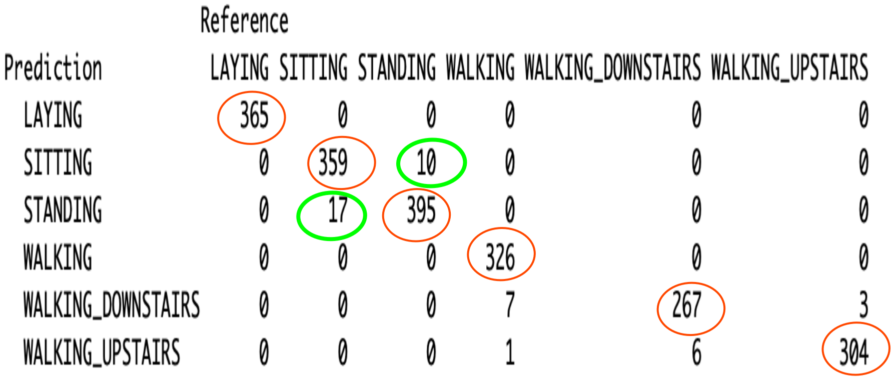
Random Forest



Random Forest



Confusion Matrix:



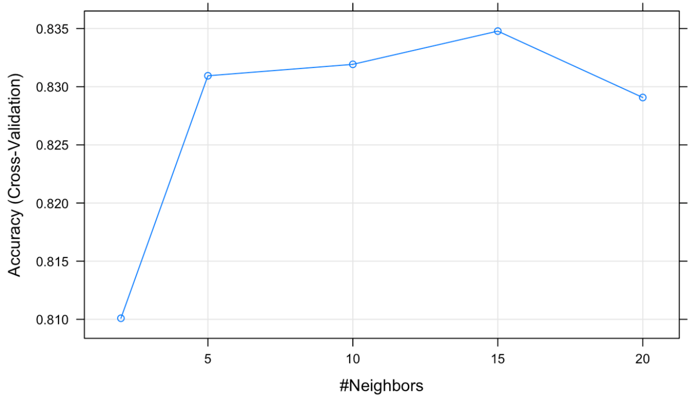
```
                    Reference
Prediction           LAYING SITTING STANDING WALKING WALKING_DOWNSTAIRS WALKING_UPSTAIRS
  LAYING                365       0        0       0                  0                0
  SITTING                 0     359       10       0                  0                0
  STANDING                0      17      395       0                  0                0
  WALKING                 0       0        0     326                  0                0
  WALKING_DOWNSTAIRS      0       0        0       7                267                3
  WALKING_UPSTAIRS        0       0        0       1                  6              304
```

- Resampling: Cross-Validated (5 fold)
- Number of trees: 500, mtry = 33

- CV Accuracy: 97.59 %
- Training Sample Accuracy: 100 %
- Testing Sample Accuracy: 97.86 %

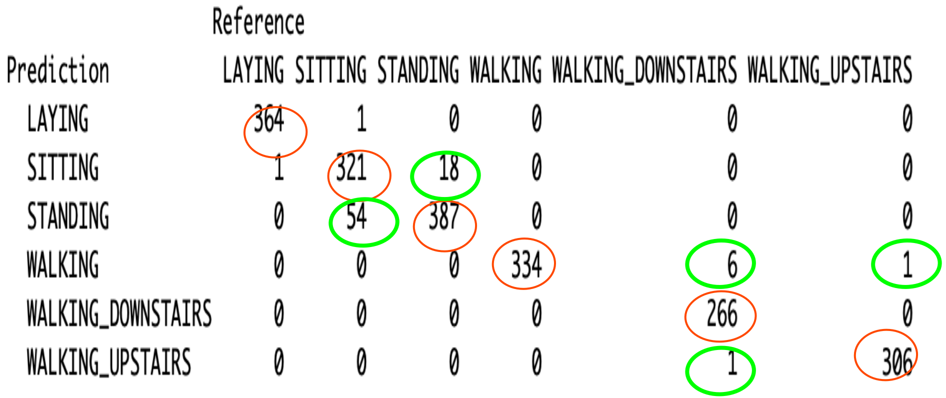
KNN

Cross Validation Results (5 Fold):



- Accuracy was used to select the optimal model using the largest value.
- 5-fold cross-validation was used to find the best k for our dataset that gives the highest accuracy.
- The optimal value k for the model was 15.

Confusion Matrix:

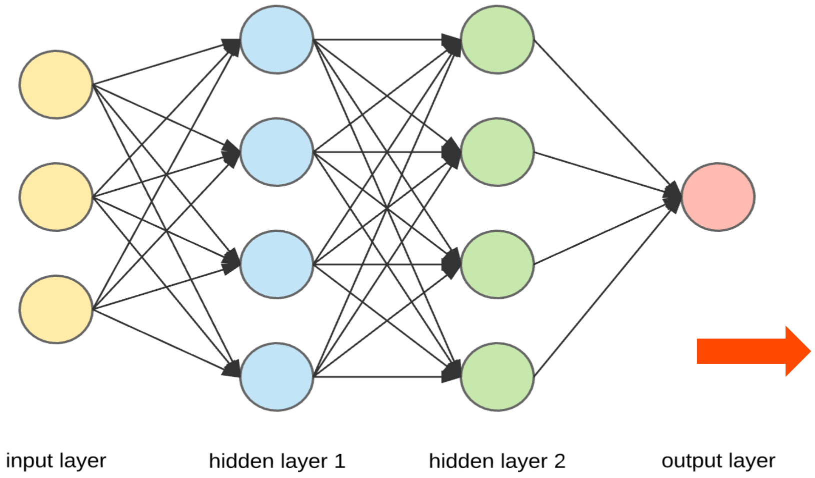


```
                    Reference
Prediction          LAYING SITTING STANDING WALKING WALKING_DOWNSTAIRS WALKING_UPSTAIRS
  LAYING               364       1        0       0                  0                0
  SITTING                1     321       18       0                  0                0
  STANDING               0      54      387       0                  0                0
  WALKING                0       0        0     334                  6                1
  WALKING_DOWNSTAIRS     0       0        0       0                266                0
  WALKING_UPSTAIRS       0       0        0       0                  1              306
```

☺ Best Model with CV: K = 15

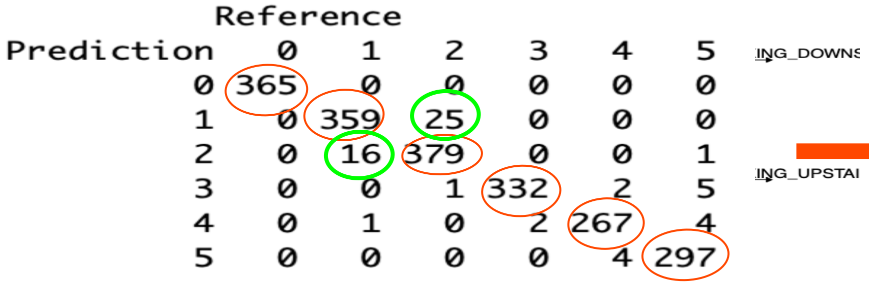
- CV Accuracy: 83.47 %
- Training Sample Accuracy: 96.46 %
- Testing Sample Accuracy: 96.02 %

Neural Networks



Hidden Configuration	Training Accuracy	Testing Accuracy
(2)	99.37 %	95.43 %
(6,3)	99.04 %	95.53 %
(6)	99.30 %	97.03 %
(10)	99.73 %	96.94 %
(6,6)	99.19 %	96.26 %

Best Model: 1 hidden layer, 6 neurons, Threshold = 0.3



```
          Reference
Prediction   0   1   2   3   4   5
         0 365   0   0   0   0   0
         1   0 359  25   0   0   0
         2   0  16 379   0   0   1
         3   0   0   1 332   2   5
         4   0   1   0   2 267   4
         5   0   0   0   0   4 297
```

Best Neural Network Model:
(1 hidden layer, 6 neurons)

- Training Sample Accuracy: 99.30 %
- Testing Sample Accuracy: 97.03 %

Naive Bayes

Theorem:

Posterior Probability

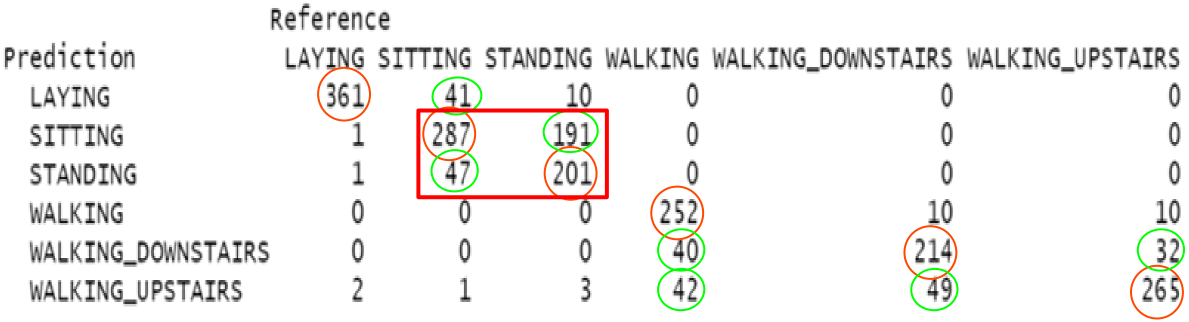
Likelihood

Prior Probability

$$p(C_k \mid \mathbf{x}) = \frac{p(C_k)\, p(\mathbf{x} \mid C_k)}{p(\mathbf{x})}$$

$$p(C_k \mid x_1, \dots, x_n) = \frac{1}{Z} p(C_k) \prod_{i=1}^{n} p(x_i \mid C_k)$$

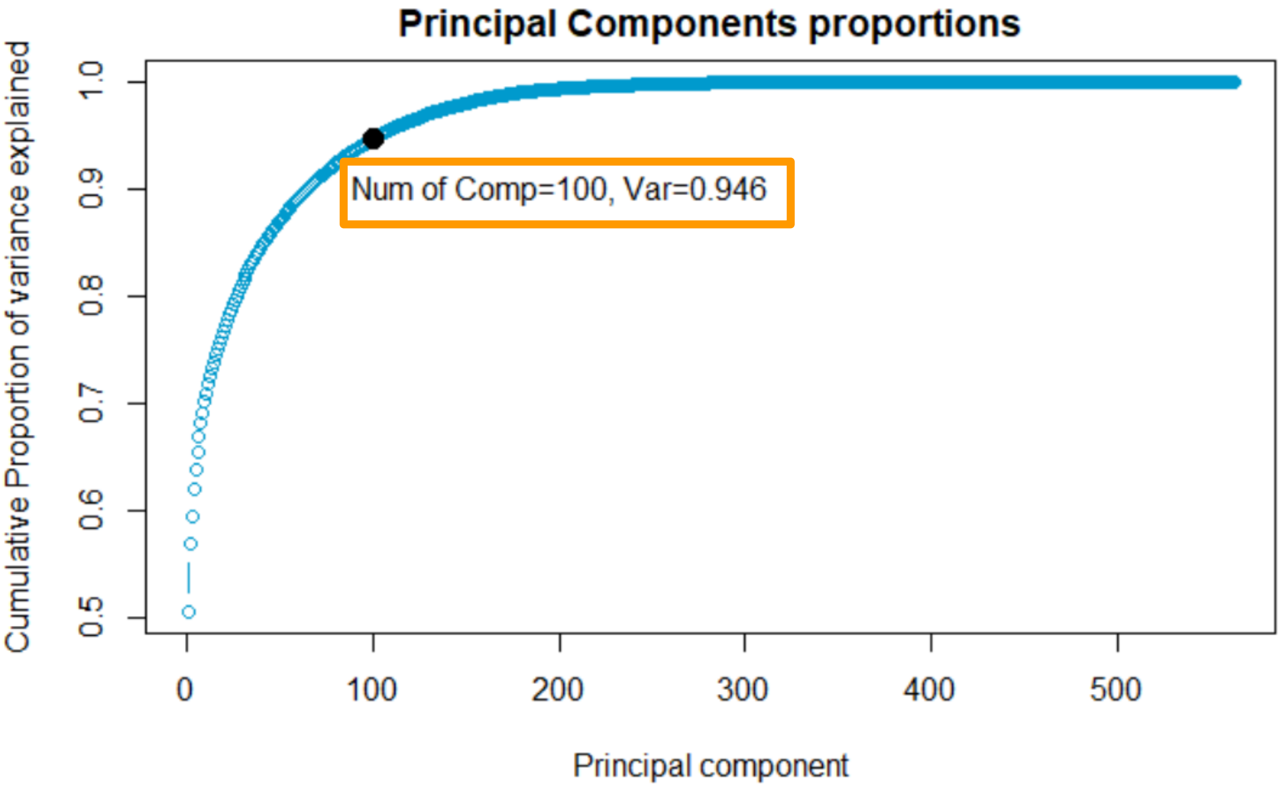
$$\hat{y} = \underset{k \in \{1,\dots,K\}}{\operatorname{argmax}}\ p(C_k) \prod_{i=1}^{n} p(x_i \mid C_k).$$



```
                   Reference
Prediction          LAYING SITTING STANDING WALKING WALKING_DOWNSTAIRS WALKING_UPSTAIRS
  LAYING               361      41       10       0                  0                0
  SITTING                1     287      191       0                  0                0
  STANDING               1      47      201       0                  0                0
  WALKING                0       0        0     252                 10               10
  WALKING_DOWNSTAIRS     0       0        0      40                214               32
  WALKING_UPSTAIRS       2       1        3      42                 49              265
```

Training Acc: 77.63 %
Testing Acc: 76.70 %

Dimension Reduction ——PCA



Dimension Reduction Application

Models	All Features Accuracy	PCA Features Accuracy
Logistic	(L1 Penalty) 0.9597	0.9539
	(L2 Penalty) 0.9743	0.9558
LDA	0.9820	0.9364
Random Forest	0.9786	0.9432
Naive Bayes	0.7670	0.8534
KNN	0.9602	0.9597 ★
SVM	0.9850	0.9573
Neural Network	0.9704	0.9451

Why PCA impaired some models' performance?

Loss of Information.
Unsupervised algorithm!

Conclusion and Future Work

Conclusion

- Almost all models perform very well on this feature-engineered dataset
- Before regularization, **SVM** would be our choice.
- After PCA regularization, **KNN** outperforms others.
- The PCA regularization doesn't bring expected positive influence.

Future Work

- **Selecting Variables** wisely
- Further deep work on **Neural Network:**
  - **Hyperparameter Tuning**
  - **Variations:** Recurrent Neural Networks, Long Short-Term Memory, Convolutional Neural Networks
- **More Algorithms:** Gradient Boosting Machine (Adaboost) etc.

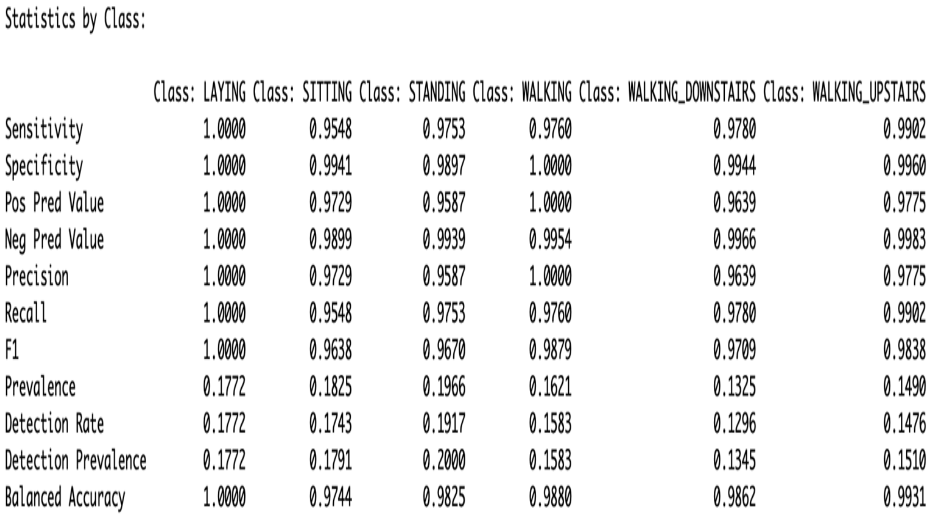
Reference

- **Dataset:** https://www.kaggle.com/uciml/human-activity-recognition-with-smartphones
- https://en.wikipedia.org/wiki/Support_vector_machine

Thank You

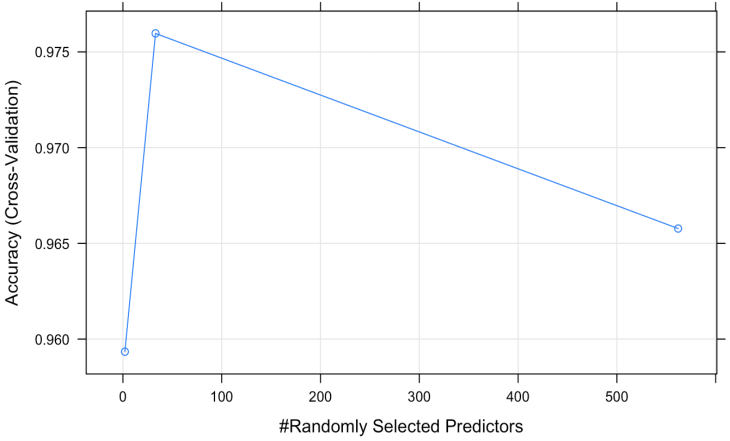
Appendix

Random Forest



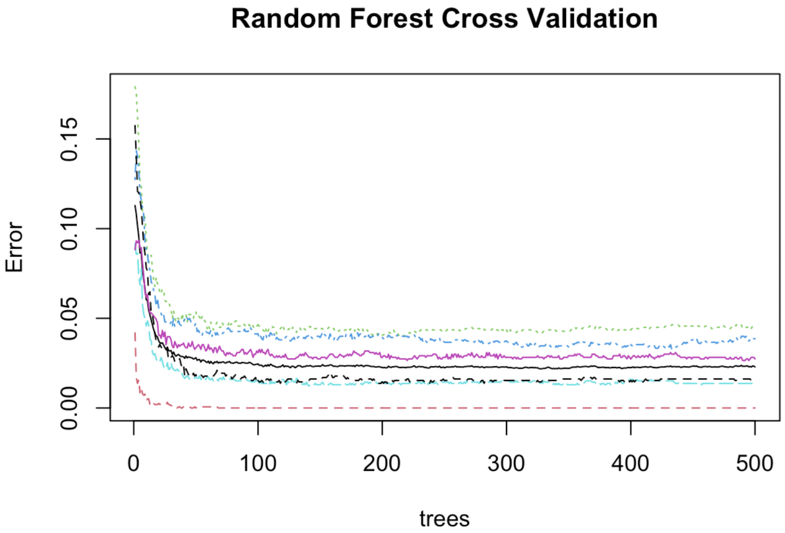
```
Statistics by Class:

                     Class: LAYING Class: SITTING Class: STANDING Class: WALKING Class: WALKING_DOWNSTAIRS Class: WALKING_UPSTAIRS
Sensitivity                 1.0000         0.9548          0.9753         0.9760                    0.9780                  0.9902
Specificity                 1.0000         0.9941          0.9897         1.0000                    0.9944                  0.9960
Pos Pred Value              1.0000         0.9729          0.9587         1.0000                    0.9639                  0.9775
Neg Pred Value              1.0000         0.9899          0.9939         0.9954                    0.9966                  0.9983
Precision                   1.0000         0.9729          0.9587         1.0000                    0.9639                  0.9775
Recall                      1.0000         0.9548          0.9753         0.9760                    0.9780                  0.9902
F1                          1.0000         0.9638          0.9670         0.9879                    0.9709                  0.9838
Prevalence                  0.1772         0.1825          0.1966         0.1621                    0.1325                  0.1490
Detection Rate              0.1772         0.1743          0.1917         0.1583                    0.1296                  0.1476
Detection Prevalence        0.1772         0.1791          0.2000         0.1583                    0.1345                  0.1510
Balanced Accuracy           1.0000         0.9744          0.9825         0.9880                    0.9862                  0.9931
```

Cross Validation Results (5 Fold):



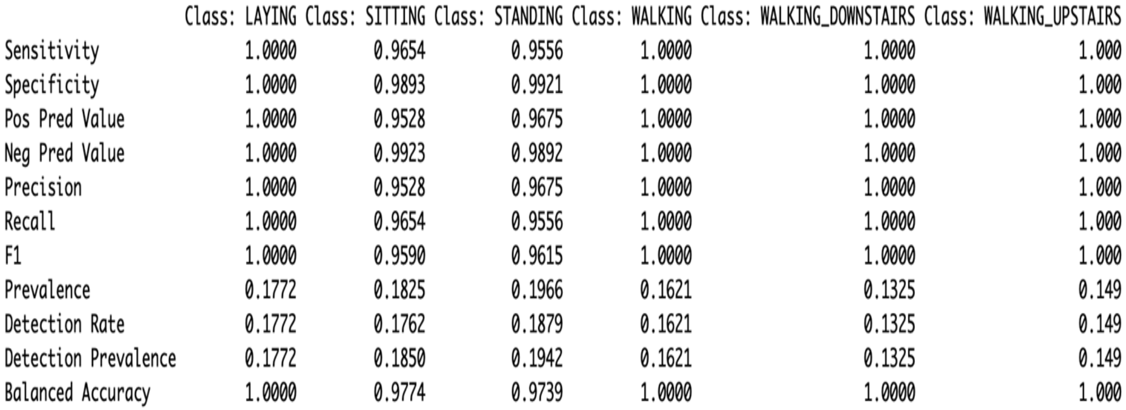
Appendix

Random Forest



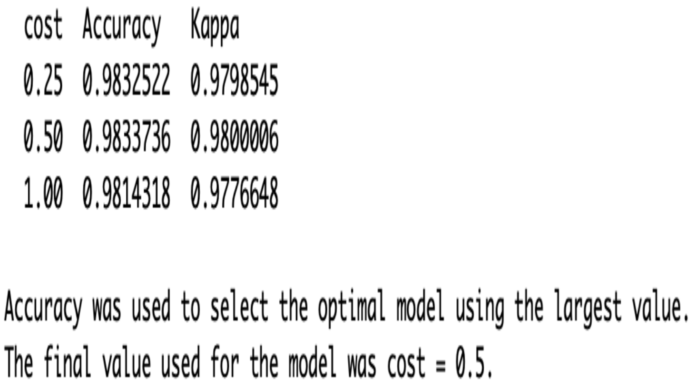
Appendix

SVM



```
                     Class: LAYING Class: SITTING Class: STANDING Class: WALKING Class: WALKING_DOWNSTAIRS Class: WALKING_UPSTAIRS
Sensitivity                 1.0000         0.9654          0.9556         1.0000                    1.0000                   1.000
Specificity                 1.0000         0.9893          0.9921         1.0000                    1.0000                   1.000
Pos Pred Value              1.0000         0.9528          0.9675         1.0000                    1.0000                   1.000
Neg Pred Value              1.0000         0.9923          0.9892         1.0000                    1.0000                   1.000
Precision                   1.0000         0.9528          0.9675         1.0000                    1.0000                   1.000
Recall                      1.0000         0.9654          0.9556         1.0000                    1.0000                   1.000
F1                          1.0000         0.9590          0.9615         1.0000                    1.0000                   1.000
Prevalence                  0.1772         0.1825          0.1966         0.1621                    0.1325                   0.149
Detection Rate              0.1772         0.1762          0.1879         0.1621                    0.1325                   0.149
Detection Prevalence        0.1772         0.1850          0.1942         0.1621                    0.1325                   0.149
Balanced Accuracy           1.0000         0.9774          0.9739         1.0000                    1.0000                   1.000
```

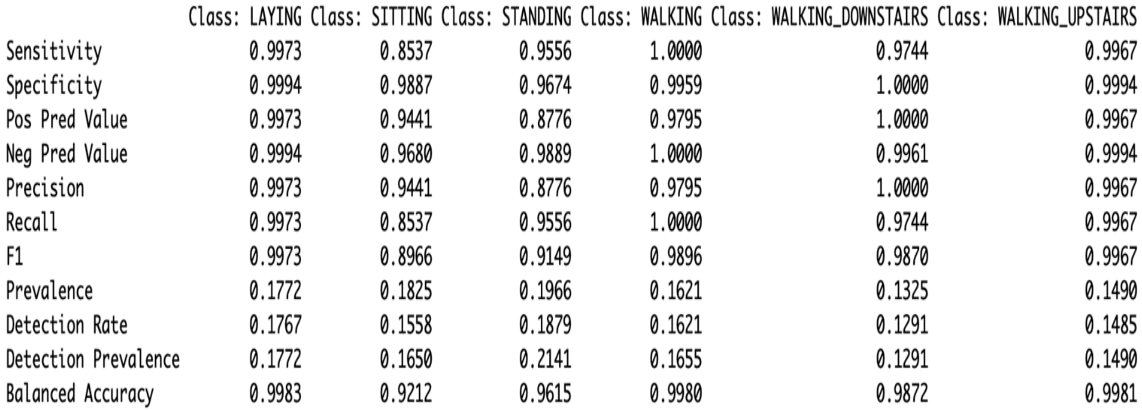
Cross Validation Results (5 Fold):



```
cost  Accuracy   Kappa
0.25  0.9832522  0.9798545
0.50  0.9833736  0.9800006
1.00  0.9814318  0.9776648

Accuracy was used to select the optimal model using the largest value.
The final value used for the model was cost = 0.5.
```

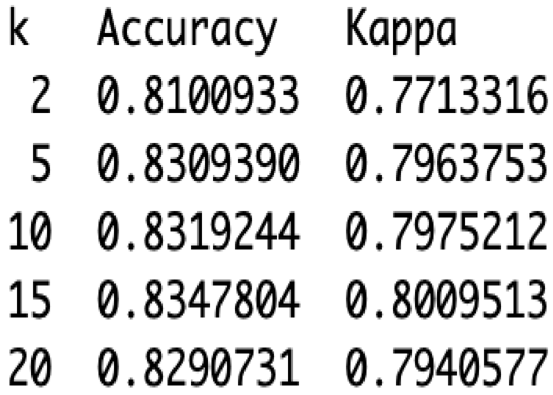
Appendix

KNN



```
                     Class: LAYING Class: SITTING Class: STANDING Class: WALKING Class: WALKING_DOWNSTAIRS Class: WALKING_UPSTAIRS
Sensitivity                 0.9973         0.8537          0.9556         1.0000                    0.9744                   0.9967
Specificity                 0.9994         0.9887          0.9674         0.9959                    1.0000                   0.9994
Pos Pred Value              0.9973         0.9441          0.8776         0.9795                    1.0000                   0.9967
Neg Pred Value              0.9994         0.9680          0.9889         1.0000                    0.9961                   0.9994
Precision                   0.9973         0.9441          0.8776         0.9795                    1.0000                   0.9967
Recall                      0.9973         0.8537          0.9556         1.0000                    0.9744                   0.9967
F1                          0.9973         0.8966          0.9149         0.9896                    0.9870                   0.9967
Prevalence                  0.1772         0.1825          0.1966         0.1621                    0.1325                   0.1490
Detection Rate              0.1767         0.1558          0.1879         0.1621                    0.1291                   0.1485
Detection Prevalence        0.1772         0.1650          0.2141         0.1655                    0.1291                   0.1490
Balanced Accuracy           0.9983         0.9212          0.9615         0.9980                    0.9872                   0.9981
```

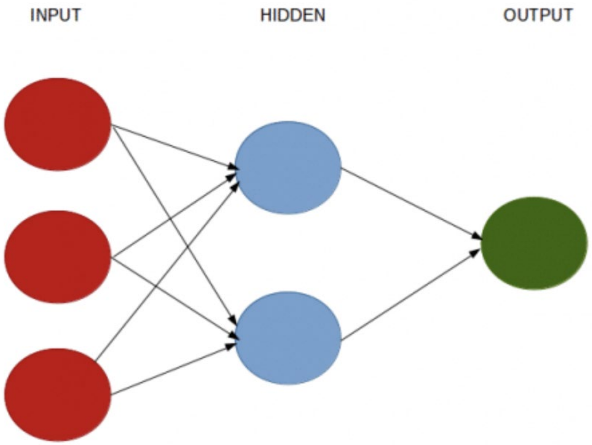
Cross Validation Results (5 Fold):

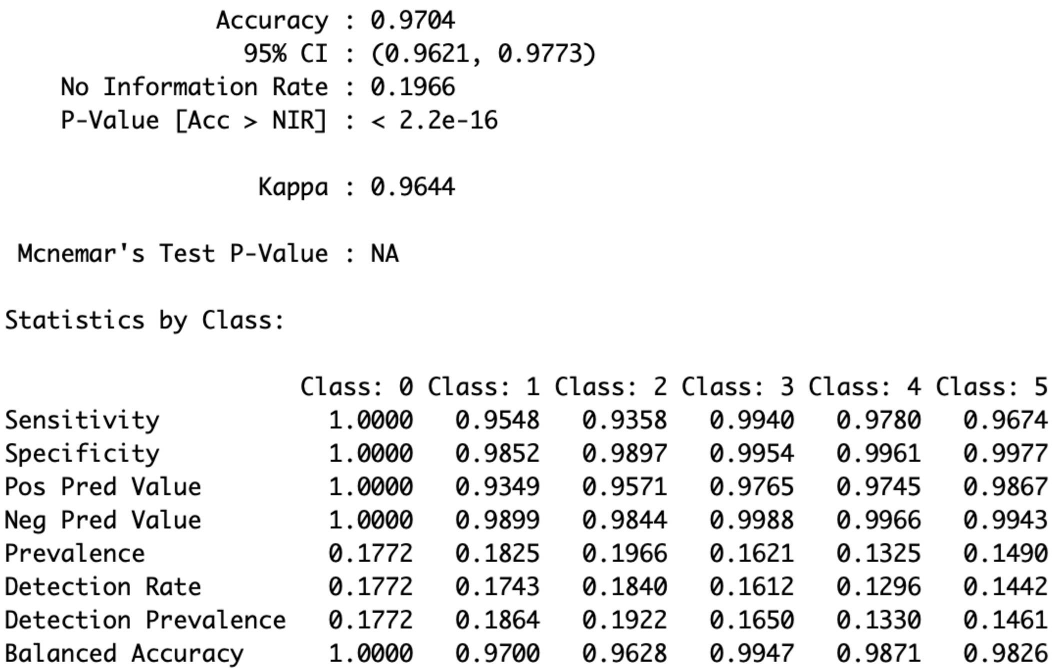


```
 k  Accuracy   Kappa
 2  0.8100933  0.7713316
 5  0.8309390  0.7963753
10  0.8319244  0.7975212
15  0.8347804  0.8009513
20  0.8290731  0.7940577
```

Appendix

Neural Networks





```
               Accuracy : 0.9704
                 95% CI : (0.9621, 0.9773)
    No Information Rate : 0.1966
    P-Value [Acc > NIR] : < 2.2e-16

                  Kappa : 0.9644

 Mcnemar's Test P-Value : NA

Statistics by Class:

                     Class: 0 Class: 1 Class: 2 Class: 3 Class: 4 Class: 5
Sensitivity            1.0000   0.9548   0.9358   0.9940   0.9780   0.9674
Specificity            1.0000   0.9852   0.9897   0.9954   0.9961   0.9977
Pos Pred Value         1.0000   0.9349   0.9571   0.9765   0.9745   0.9867
Neg Pred Value         1.0000   0.9899   0.9844   0.9988   0.9966   0.9943
Prevalence             0.1772   0.1825   0.1966   0.1621   0.1325   0.1490
Detection Rate         0.1772   0.1743   0.1840   0.1612   0.1296   0.1442
Detection Prevalence   0.1772   0.1864   0.1922   0.1650   0.1330   0.1461
Balanced Accuracy      1.0000   0.9700   0.9628   0.9947   0.9871   0.9826
```