# CLASSIFICATION OF EXERCISE QUALITY BASED ON ACCELOROMETER DATA


**Abstract**

**Introduction & Assignment Goals:** Implement a machine learning algorithm which could classify the data on exercise performed by different individuals from personal activity monitoring devices and classify them based on efficacy or effectiveness of the performed activity. The training data classifies the activities into five groups ('classe') and the algorithm needs to be able to classify the accelerometer data into one of these five classification groups accurately.


**Methodology**

Methodology followed is briefly outlined in Figure 1. It was noted that there were large number of variables in the original data. Given the large number of variables involved in the study, two approaches were taken for implementing the machine learning algorithm: (1) the algorithm was trained directly on the data; (2) Dimension reduction was performed using PCA and only selected variables were used for training the model.

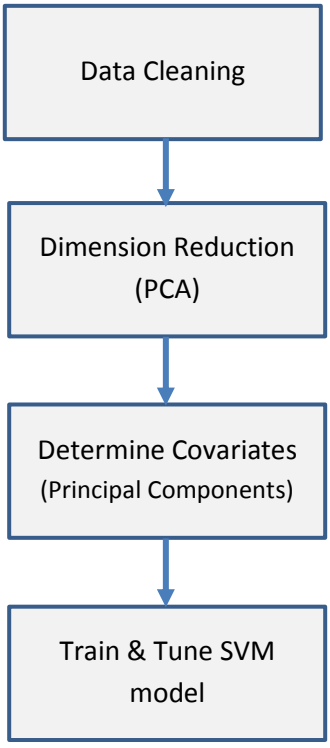


Figure 1: Outline of Methodology Used

## 1. Data Preprocessing

### 1.1. Covariate Selection & Data Extraction

The original data consists of a total of 152 variables encompassing a total of 19622 rows. This can be seen using the dim (dimension) command of R. By utilizing the 'str' command, it was possible to see that there were many columns which had a significant number of 'NA'. These columns were removed from the original data set by sub-setting the data. Furthermore, columns the data pertaining to data collection scheme (user_name, raw_times etc.) were also excluded from the data. This resulted in a dataset with 53 rows (including the outcome variable 'classe').

### 1.2. **Dimension Reduction using PCA**:

Dimension reduction was performed on the existing data sets by utilizing the principal component analysis function in R (preProcess, with method = "PCA"). Figure 2 presents the plot of variance explained by PCA against the number of principal components. Four principal components (PC) explained ~75% of the total variance, while seven PCs explained roughly 90%. It was further seen that it only took 18 principal components to explain 99% of the total variation. Therefore, using the dimension reduction technique, it was possible to minimize the covariates from 52 variables to as few of seven. The final number of PCs to be included can be determined based on the accuracy level of the machine learning algorithm utilized.

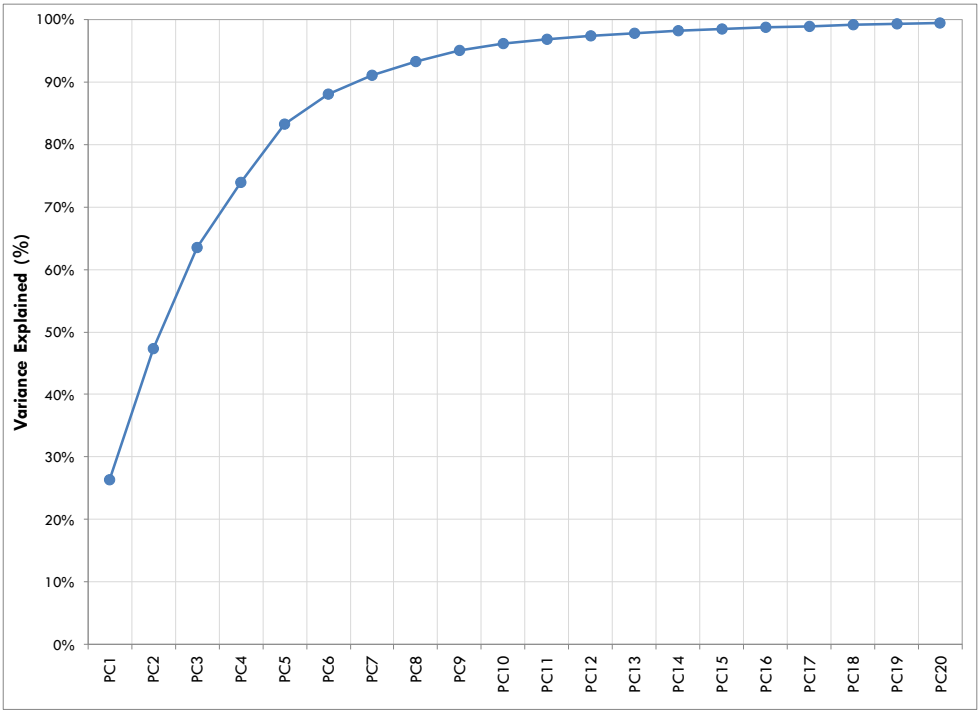


**Figure 2: PCA Analysis**

## 2. Selection of Machine Learning Algorithms

The outcome variable ('classe') consisted of five categories ('A' to 'E') dependent on the quality of the exercise performed. Hence, this is a classification problem, and three potential machine learning algorithms 'Random Forest', 'Classification Tree' or 'Support Vector Machines' were evaluated on the training data. For the purposes of this report, the details & results specific to SVM are only presented here. The RBF (radial basis function) and linear kernel, which are widely used for classification problems, was used for training the model.

### 2.1. **Cross Validation & Tuning of SVM**

Even though there are ~19000 rows of data, however the data is from 6 participants of the exercise study. *Hence, rather than splitting the data into training and test, cross validation would be an appropriate method*. *Repeated k-fold cross-validation* which repeats cross validation process multiple times and average the performance across the multiple runs.

This validation technique requires two variables: (1) k – number of folks; (2) number of repeats of the k fold. For the purposes of this assignment, a 10 fold cross validation, with 10 repeats was initially utilized. It was seen that training time was significantly high for such a permutation (>2 hours). Therefore, the number of folds and repeats were reduced to 3 fold, 3 repeats. For the RBF kernel, the results from the cross validation (Table 1) was then used to identify the right set of parameters for the final SVM model. Best value of the SVM parameters were Sigma = 0.0130848 and C =32 (*svm.tune$best*).

| C | Accuracy | Kappa | Accuracy SD | Kappa SD |
|---|---|---|---|---|
| **0.25** | 0.9015 | 0.8751 | 0.0036 | 0.0046 |
| **0.5** | 0.9307 | 0.9121 | 0.0040 | 0.0051 |
| **1** | 0.9538 | 0.9415 | 0.0038 | 0.0049 |
| **2** | 0.9672 | 0.9584 | 0.0033 | 0.0042 |
| **4** | 0.9772 | 0.9711 | 0.0029 | 0.0036 |
| **8** | 0.9842 | 0.9800 | 0.0025 | 0.0032 |
| **16** | 0.9877 | 0.9845 | 0.0011 | 0.0014 |
| **32** | 0.9888 | 0.9859 | 0.0014 | 0.0017 |
| **64** | 0.9888 | 0.9858 | 0.0015 | 0.0019 |

Table 1: Tuning of SVM Model w Cross Validation

## 3. Results

### 3.1. Impact of Kernel Selection

Linear and RBF kernels were utilized for the SVM training. Linear kernel provided lower classification accuracy (78%) while RBF gave a higher accuracy of ~98%. However, the training time for the linear kernel was much lower than the RBF, hence there needs to be careful evaluation between the tradeoff between computation time vs. acceptable accuracy of the model.

| Kernel | Min. | 1st Qu. | Median | Mean | 3rd Qu. | Max. |
|---|---|---|---|---|---|---|
| RBF | 0.9881 | 0.9891 | 0.9904 | 0.9899 | 0.9906 | 0.9912 |
| Linear | 0.7873 | 0.7942 | 0.7942 | 0.7942 | 0.7945 | 0.7982 |

Table 2: Comparison of Kernels of SVM

### 3.2. Sensitivity Analysis with the number of PCAs

In order to explore the impact of dimension reduction on the classification accuracy, a sensitivity analysis was performed by varying the number of principal components used as the covariates (or predictors). The parameters of the SVM were established from the tuning process (stated in *section 2.1)*.The result of this experiment is provided in Table 3. It can be seen that the accuracy improved as the number of PCs were increased from 7 to 30 (higher proportion of the variance were explained with greater number of PCs). It can be seen that by incorporating all the PCs, the accuracy improved from 93.6% (30 PCs) to 96.39% - This is a 3% improvement, even though 30 PCs accounted for 99.96% of the total variance.

| No. of PCA | 7 | 10 | 15 | 20 | 25 | 30 | All PCAs |
|---|---|---|---|---|---|---|---|
| % Variance Explained | 91.05% | 96.14% | 98.49% | 99.42% | 99.82% | 99.96% | 100% |
| Accuracy | 0.6896 | 0.8317 | 0.8836 | 0.9076 | 0.9235 | 0.9365 | 0.98339 |

Table 3: Effect of Number of PCAs on Classification Accuracy

## 4. Conclusions

SVM algorithm was trained on the accelerometer data provided. The RBF accuracy was close to 98% when repeated cross validation was utilized for the training. The RBF kernel also outperformed the linear kernel.

## References


1. Hsu C., Chang C., and Lin C. 2016. “A Practical Guide to Support Vector Classification”. https://www.csie.ntu.edu.tw/~cjlin/papers/guide/guide.pdf, Last Referenced: August 2016.
2. Rickert J. “The 5th Tribe, Support Vector Machines and caret”, **https://www.r-bloggers.com/the-5th-tribe-support-vector-machines-and-caret/**; Last Referenced: August 2016.