1 Introduction

The purpose of this report is to conduct a classification task to predict a woman's contraceptive method (no use, long-term methods, or short-term use) based on her demographic and socioeconomic characteristics.

2 Exploratory Data Analysis

2.1 The Dataset

This dataset is a subset of the 1987 National Indonesia Contraceptive Prevalence Survey (Dua and Graff 2019). The samples are married women who were either not pregnant, or unaware of pregnancy at the time of interview. The cmc.data file in the Data Folder has provided a code book with column names(as shown below) from the Attribute Information section in the CodeBook. It is clear the dataset requires preparation due to no column names. Therefore, using the codebook provided with the dataset, column names were assigned to the corresponding columns.

-

- **WifeAge**: Wife's age
- **WifeEducation**: Wife's education with 1 as the lowest, 2, 3, and 4
- **HusbandEducation**: Husband's education with 1 as the lowest, 2, 3, and 4
- **NumberOfChildrenEverBorn**: Number of children ever born
- **WifeReligion**: Wife's religion with 0 as Non-Islam, and 1 as Islam
- **WifeNowWorking**: Wife's working status with 0 as Yes, they work, 1 as No
- **HusbandOccupation**: Husband's occupation with either 1, 2, 3 or 4
- **StandardOfLivingIndex**: Standard-of-living index with 1 as low, 2, 3, 4
- **MediaExposure**: Media exposure with 0 as Good, and 1 as Not good
- **ContraceptiveMethodUsed**: Contraceptive method used which the target variable with 1 as No-use, 2 as Long-term Use, and = 3 as Short-term Use

3 3. Data Preparation

3.1 Load Data

The data was loaded from the UCI repository and inspected below:

```
> set.seed(123)
> data <- read.csv('cmc.data', header=TRUE)
> head(data)
```

```
  X24 X2 X3 X3.1 X1 X1.1 X2.1 X3.2 X0 X1.2
1  45  1  3   10  1    1    3    4  0    1
2  43  2  3    7  1    1    3    4  0    1
3  42  3  2    9  1    1    3    3  0    1
4  36  3  3    8  1    1    3    2  0    1
5  19  4  4    0  1    1    3    3  0    1
6  38  2  3    6  1    1    3    2  0    1
```

It is clear the dataset requires preparation due to no column names. Therefore, using the codebook provided with the dataset, column names were assigned to the corresponding columns.

3.2 Add Column Names

```
> set.seed(123)
> colnames(data)<- c("WifeAge", "WifeEducation",
+                    "HusbandEducation", "NumberOfChildrenEverBorn",
+                    "WifeReligion", "WifeNowWorking",
+                    "HusbandOccupation", "StandardOfLivingIndex",
+                    "MediaExposure", "ContraceptiveMethodUsed")
> head(data)
```

```
  WifeAge WifeEducation HusbandEducation NumberOfChildrenEverBorn WifeReligion
1      45             1                3                       10            1
2      43             2                3                        7            1
3      42             3                2                        9            1
4      36             3                3                        8            1
5      19             4                4                        0            1
6      38             2                3                        6            1
  WifeNowWorking HusbandOccupation StandardOfLivingIndex MediaExposure
1              1                 3                     4             0
2              1                 3                     4             0
3              1                 3                     3             0
4              1                 3                     2             0
5              1                 3                     3             0
6              1                 3                     2             0
  ContraceptiveMethodUsed
1                       1
2                       1
3                       1
4                       1
5                       1
6                       1
```

3.3 Detecting missing values

In order to remove missing values, they are initially detected where it is clear that all 1473 are complete values and that none need to be removed.

```
> set.seed(123)
> #load the library
> library(tidyr)
> library(ggplot2)
> # number of complete cases
> sum(complete.cases(data))

[1] 1472

>
```

3.4 Exploratory Analysis

The dataset has 1473 number of instances and 10 attributes. From the summary below it shows the majority of the data as ordinal which would mean the columns would need to be converted to factors to be able to capture the predictors properly for the classification.

```
> set.seed(123)
> # find dimensions of the data
> dim(data)

[1] 1472   10

> # find summary of the data
> summary(data)

    WifeAge      WifeEducation   HusbandEducation NumberOfChildrenEverBorn
 Min.   :16.00   Min.   :1.000   Min.   :1.00     Min.   : 0.000
 1st Qu.:26.00   1st Qu.:2.000   1st Qu.:3.00     1st Qu.: 1.000
 Median :32.00   Median :3.000   Median :4.00     Median : 3.000
 Mean   :32.54   Mean   :2.959   Mean   :3.43     Mean   : 3.262
 3rd Qu.:39.00   3rd Qu.:4.000   3rd Qu.:4.00     3rd Qu.: 4.250
 Max.   :49.00   Max.   :4.000   Max.   :4.00     Max.   :16.000
  WifeReligion    WifeNowWorking   HusbandOccupation StandardOfLivingIndex
 Min.   :0.0000   Min.   :0.0000   Min.   :1.000     Min.   :1.000
 1st Qu.:1.0000   1st Qu.:0.0000   1st Qu.:1.000     1st Qu.:3.000
 Median :1.0000   Median :1.0000   Median :2.000     Median :3.000
 Mean   :0.8505   Mean   :0.7493   Mean   :2.138     Mean   :3.134
 3rd Qu.:1.0000   3rd Qu.:1.0000   3rd Qu.:3.000     3rd Qu.:4.000
 Max.   :1.0000   Max.   :1.0000   Max.   :4.000     Max.   :4.000
 MediaExposure     ContraceptiveMethodUsed
 Min.   :0.00000   Min.   :1.000
```

```
1st Qu.:0.00000   1st Qu.:1.000
Median :0.00000   Median :2.000
Mean   :0.07405   Mean   :1.921
3rd Qu.:0.00000   3rd Qu.:3.000
Max.   :1.00000   Max.   :3.000
```

To understand the socio-economic characteristic of the participants the main predictors used in this will include all features except media exposure. The age of the wives(WifeAge), the number of children ever born(NumberOfChildrenEverBorn), the standard of living index (StandardOfLivingIndex) and the wife's education (WifeEducation) will be visualised in the following figures. In Figure 1 below it analyses the first numerical columns of the dataset the Wife Age and Figure 2 shows the statistics of the second numerical column of the number of children ever born, in box plots.However During the conversion process to a latex document some figures have been moved to other areas of the document. Most noteably the target variable visualisation, Figure 5, can be found below Section 6.

Wife'a Age

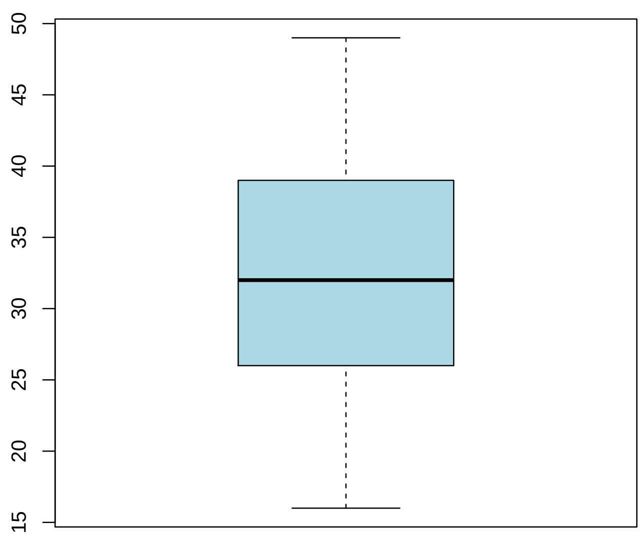

Figure 1: WifeAge

The average age of the wives is 32.5 in the dataset with the minimum age of 16 found is and the maximum age found being 49 in the data. The majority of the data found women between the ages of 26 and 38 with no outliers found in the data.

Number of Children Ever Born

```
> set.seed(123)
> # create boxplot
> boxplot(data$NumberOfChildrenEverBorn, ylab = "Number",
+         main = "Number Of Children Ever Born",
+         col="lightblue", outcol="orange")
```

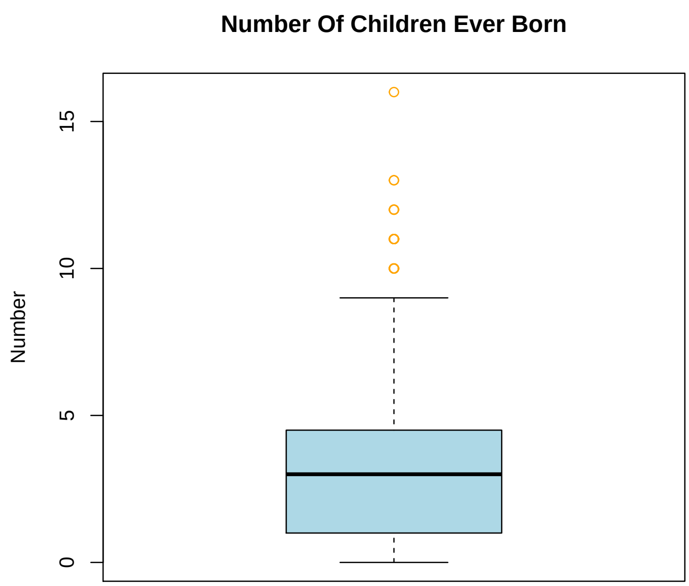

Figure 2: Number Of Children Ever Born

From Figure 2, the average number of children ever born is 3 within the dataset. When the minimum found is zero and the maximum found being 16. The majority of the wives in the dataset have between 1 to 4 children however there are outliers between 10 and 16. By analysing these statistics it is expected that the minimum value of would be zero.

Wife's Education

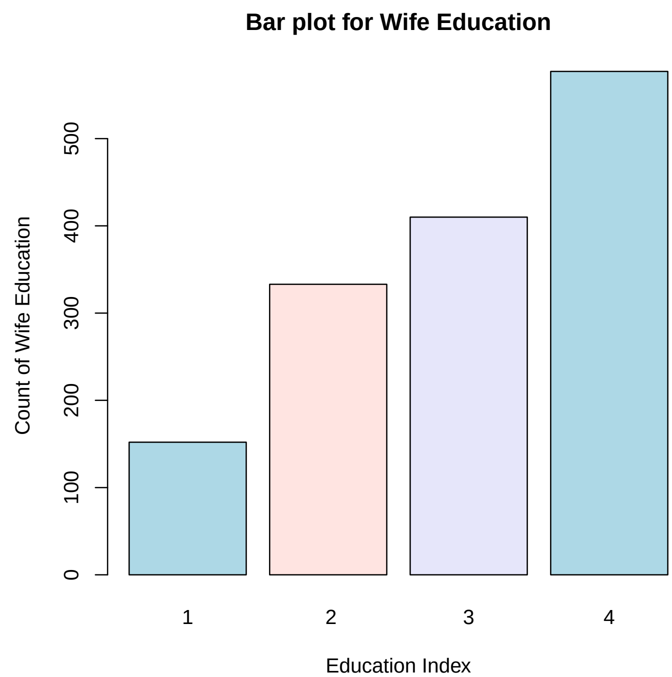

Figure 3: Bar plot for Wife Education

From Figure 3, The highest level of education for the wives in the dataset was four, according to the figure there are more women in the dataset with high level of education that lower level.

StandardOfLivingIndex

From Figure 4, the most frequent Standard Of Living Index in the dataset was four also, this suggest that the dataset consists of women with a high standard of living.

3.5 Target Variable - ContraceptiveMethodUsed

The target variable is within the focus of the classification experiments will be the contraceptive method used (ContraceptiveMethodUsed) column which is apply to a bar plot in Figure 5.

It is clear from Figure 5, that the highest count in the dataset is the no use method with 628 counts found. This suggests the majority of the sample have no contraception in place. The second highest count of contraceptive method

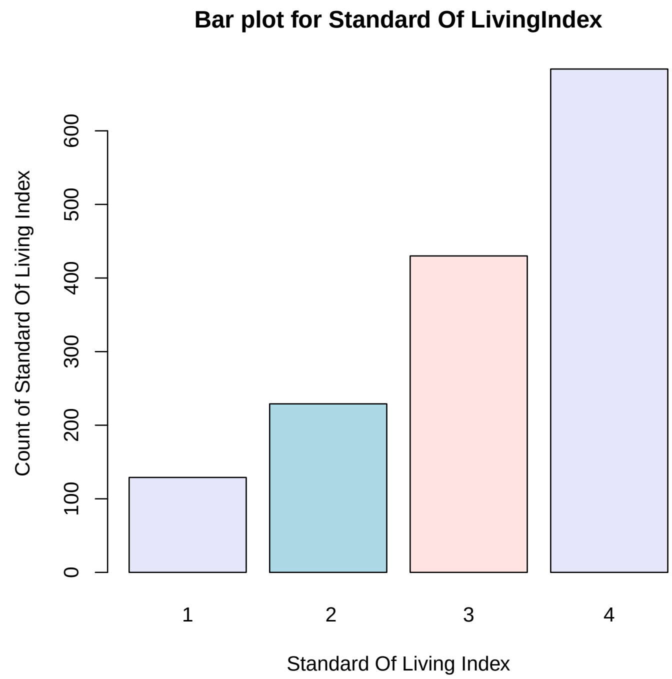

Figure 4: Bar plot for Standard Of Living Index

used would be short term with 511 counts however long-term contraception is the lowest with a count of 333. This suggests, that when implementing the prediction, the predictors could be more likely to predict wives have no method of contraception than do.

Count

```
> set.seed(123)
> #calculate the count of each contraceptive method used
> contraceptiveCount <- table(data$ContraceptiveMethodUsed)
> #show the count
> contraceptiveCount

  1   2   3 
628 333 511 
```

Table 1: Count of Target Variable

No-use	Long-term	Short-term
628	333	511

4 Supervised Learning Experiments

4.1 Methodology

Each experiment will implement a classification Before implementing the experiments the data is split with 80 percent of the data used for training and 20 percent reserved for testing on each model. This will apply an additional level of accuracy for evaluating the results of each classifier. The seed is set as 123 for each experiment and a five fold cross validation method will be used on all 5 experiment.

To implement each experiment the each column will be converted to a factor and the media exposure column will be removed as the experiments aims to use only socio-economic factors related directly to the wife or husband. The three contraceptive methods used have been converted into factors of 0, 1, and 2.

```
> #remove MediaExposure
> data$MediaExposure <- NULL
> #convert columns to factors
> data$HusbandEducation <- as.factor(data$HusbandEducation)
> data$WifeReligion <- as.factor(data$WifeReligion)
> data$WifeNowWorking <- as.factor(data$WifeNowWorking)
> data$HusbandOccupation <- as.factor(data$HusbandOccupation)
> data$WifeAge <- as.factor(data$WifeAge)
> data$WifeEducation <- as.factor(data$WifeEducation)
> data$NumberOfChildrenEverBorn <- as.factor(data$NumberOfChildrenEverBorn)
> data$StandardOfLivingIndex <- as.factor(data$StandardOfLivingIndex)
> data$ContraceptiveMethodUsed <- as.factor(data$ContraceptiveMethodUsed)
```

Splitting the Dataset

```
> library(caret)
> set.seed(123)
> #divide data to 80% and 20%
> trainingInst <- createDataPartition(data$ContraceptiveMethodUsed,
+                                    p=0.8, list=FALSE)
> #create training set
> test.set <- data[-trainingInst,]
> #create testing set
> train.set <- data[trainingInst,]
```

Apply five fold cross validation

```
> #import caret and lattice packages
> library(caret)
> library(lattice)
> #set the seed
> set.seed(123)
> #apply 5 fold cross validation
> control <- trainControl(method="cv", number=5)
>
```

4.2 Linear Discriminant Analysis

Initially, a Linear Discriminant Analysis(LDA) model is used to predict the method of contraceptive use.

The model was chosen due to the ability to predict 3 classes. It uses the prior probability and the density function to chose the class with the highest probability and assumes that every class share the same standard deviation. As there are eight predictors the density function will have a mean vector of all predictors and covariance. More than one predictor also suggests a multivariate normal distribution. This makes a decision based on the probability of an observation to be part of a certain class or not(James et al. 2013). Since this model is linear discriminant analysis there are no parameters needed to tune.

```
> #import library
> library(e1071)
> library(caret)
> #set the seed
> set.seed(123)
> #fit the LDA classifier
> lda.fit <-train(ContraceptiveMethodUsed ~ ., data = train.set,
+ trControl=control,
+ preProcess = c("center"))
> #create the predictions
> predictions <- predict(lda.fit, test.set)
> cmlda <- confusionMatrix(predictions, test.set$ContraceptiveMethodUsed)
```

```
> #import xtable
> set.seed(123)
> library(xtable)
> xtable(lda.fit$results[,c(1,2,3,4)],
+        digits = c(0,2,2,4,4))
```

```
> set.seed(123)
> # show statistics
> cmlda$overall
```

Table 2: Accuracy and Kappa for LDA on Training Data

	mtry	Accuracy	Kappa	AccuracySD
0.01	0.25	0.4529	0.0595	
0.01	0.50	0.4775	0.1314	
0.01	1.00	0.4894	0.1671	

```
     Accuracy          Kappa  AccuracyLower  AccuracyUpper   AccuracyNull
  0.511945392    0.225140088    0.453134609    0.570512216    0.426621160
AccuracyPValue  McnemarPValue
  0.001995222    0.002988819
```

Table 3: Accuracy and Kappa of LDA on Testing Data

Accuracy	Kappa
0.50	0.19

When analysing the accuracy and the kappa statistics we can understand the effectiveness of the model of predicting the target variable. The accuracy is quite low at 0.50 . However the kappa value can provide a more robust indication with 0.19 which shows the model has slight agreement in predicting the correct target variable.

The results of the LDA will be compared against the three Support Vector Machine models and the Random Forest model in Section 5 to identify the experiment most appropriate for the prediction of types of contraception used.

4.3 Support Vector Machine(SVM)

The Support Vector Machine(SVM) is a non parametric which contrasts with LDA based on assumptions of distribution of data. SVM allows for manipulation of parameters to tune for different application. The model can have high accuracy and offers the opportunity to modify the kernel to test different methods of ”linear”, ”polynomial”, and ”radial”.

The support vector machine is an extension of support vector classfier which handle non linear classification. The idea is to add dimensions to the data by applying some form of transformation to the decision variable. However there are issues over the number of features growing too quickly and therefore the computation can be computationally expensive. SVM contain hyperplanes which are better at splitting the space compared to linear discriminant analysis (James et al. 2013). The model contains a cost parameter c is a parameter that needs to be tuned as it defines how much the model is penalised when violating constraints.

The following experiments in Section 4.4 with linear SVM, Section 4.5 with polynomial SVM and Section 4.6 with radial SVM will train and test models with the to compare the accuracy and kappa statistic in Section 5.

Table 4: Accuracy and Kappa for SVM Linear on Training Data

C	Accuracy	Kappa	AccuracySD
1.00	0.54	0.2866	0.0287

4.4 Linear Support Vector Machine

For Linear Support Vector Machine the cost(C) value will be tuned.

```
> #import relevant packages
> library(MASS)
> library(kernlab)
> library(caret)
> #set the seed
> set.seed(123)
> svmLinear.fit <- train(ContraceptiveMethodUsed ~ ., data = train.set,
+                        method="svmLinear",
+                        trControl=control,
+                        preProcess = c("center"))
> prediction <- predict(svmLinear.fit, test.set)
> cmsvml <- confusionMatrix(data = prediction,
+                           reference = test.set$ContraceptiveMethodUsed)
```

```
> set.seed(123)
> print(xtable(svmLinear.fit$results[,c(1,2,3,4)], digits = c(0,2,2,4,4)),
+       include.rownames = FALSE)
```

```
% latex table generated in R 4.0.2 by xtable 1.8-4 package
% Fri Apr 23 15:56:23 2021
\begin{table}[ht]
\centering
\begin{tabular}{rrrr}
  \hline
C & Accuracy & Kappa & AccuracySD \\
  \hline
1.00 & 0.54 & 0.2866 & 0.0287 \\
   \hline
\end{tabular}
\end{table}
```

From fitting the model on the training set it states that the accuracy is 0.54 with a kappa statistic of 0.28 when C is equal to one. Which is still relatively low. This will be tested against with the testing set below to provide our final results.

```
> set.seed(123)
> #show statistics
> cmsvml$overall
```

Table 5: Accuracy and Kappa of SVM Linear on Testing Data

Accuracy	Kappa
0.49	0.21

```
      Accuracy          Kappa  AccuracyLower  AccuracyUpper   AccuracyNull
    0.49146758     0.20568757     0.43285796     0.55025144     0.42662116
AccuracyPValue  McnemarPValue
    0.01474874     0.58353376
```

When analysing the accuracy and the kappa statistics we can understand the effectiveness of the model of predicting the target variable. The accuracy is quite low at 0.49 suggesting the model which is reasonable. However the kappa value is 0.21 which shows the model has an agreement in predicting the correct target variable. This could have issues with underfitting is as the learned classifier is too simplistic and does not capture the structure of the data to make predictions with higher accuracy. The results will be compared against the four other experiments in Section 5 to identify the experiment most appropriate for the prediction of types of contraception used.

4.5 Polynomial Support Vector Machine

For Polynomial Support Vector Machine the degree, scale and cost values will be tuned:

```
> set.seed(123)
> #tuning the parameters
> Cs <- c(0.01,0.1,0.5,1,2)
> degrees <- c(2,3)
> scales <- c(1)
> tunegrid <- expand.grid(C=Cs, degree = degrees, scale = scales)
>

> library(kernlab)
> set.seed(123)
> svmPoly.fit <- train(ContraceptiveMethodUsed ~ ., data = train.set,
+                      method="svmPoly",
+                      trControl=control,
+                      preProcess = c("center"))
> prediction <- predict(svmPoly.fit, test.set)
> cmpoly<- confusionMatrix(prediction, test.set$ContraceptiveMethodUsed)

> set.seed(123)
>   library(xtable)
> print(xtable(svmPoly.fit$results[,c(1,2,3,4)], digits = c(0,2,2,4,4)),
+       include.rownames = FALSE)
```

Table 6: Accuracy and Kappa of Polynomial SVM on Training Data

degree	scale	C	Accuracy
1	0.00	0.2500	0.4266
1	0.00	0.5000	0.4266
1	0.00	1.0000	0.4266
1	0.01	0.2500	0.4613
1	0.01	0.5000	0.4783
1	0.01	1.0000	0.4911
1	0.10	0.2500	0.5079
1	0.10	0.5000	0.5139
1	**0.10**	**1.0000**	**0.5224**
2	0.00	0.2500	0.4266
2	0.00	0.5000	0.4266
2	0.00	1.0000	0.4487
2	0.01	0.2500	0.4826
2	0.01	0.5000	0.4927
2	0.01	1.0000	0.5105
2	0.10	0.2500	0.4792
2	0.10	0.5000	0.4817
2	0.10	1.0000	0.4801
3	0.00	0.2500	0.4266
3	0.00	0.5000	0.4283
3	0.00	1.0000	0.4758
3	0.01	0.2500	0.4953
3	0.01	0.5000	0.4927
3	0.01	1.0000	0.4876
3	0.10	0.2500	0.4792
3	0.10	0.5000	0.4800
3	0.10	1.0000	0.4843

Table 7: Accuracy and Kappa of SVM Polynomial on Testing Data

Accuracy	Kappa
0.51	0.24

From fitting the model on the training set it states the highest accuracy is 0.52 with a cost of 1. This will be tested against with the testing set below to provide our final results.

```
> set.seed(123)
> cmpoly$overall
```

```
      Accuracy          Kappa  AccuracyLower  AccuracyUpper   AccuracyNull
   0.515358362    0.237482589    0.456523309    0.573879750    0.426621160
AccuracyPValue  McnemarPValue
   0.001368253    0.022379416
```

When analysing the accuracy and the kappa statistics we can understand the effectiveness of the model of predicting the target variable. The accuracy is quite low at 0.51 suggesting the model is more likely to predict the wrong value than the right one. However the kappa value indicates 0.24 which shows the model has slight agreement in predicting the correct target variable.

4.6 Radial Support Vector Machine

For Radial Support Vector Machine the sigma and cost values will be tuned

```
> library(e1071)
> library(caret)
> set.seed(123)
> svmRadial.fit <- train(ContraceptiveMethodUsed ~ ., data = train.set,
+                         method="svmRadial",
+                         trControl=control,
+                         preProcess = c("center"))
> prediction <- predict(svmRadial.fit, test.set)
> cmsvmRad <- confusionMatrix(prediction, test.set$ContraceptiveMethodUsed)
```

```
> set.seed(123)
> library(xtable)
> print(xtable(svmRadial.fit$results[,c(1,2,3,4)],
+              digits = c(0,2,2,4,4)), include.rownames = FALSE)
```

```
> set.seed(123)
> cmsvmRad$overall
```

Table 8: Accuracy and Kappa of SVM Radial on Training Data

sigma	C	Accuracy	Kappa
0.01	0.25	0.4529	0.0595
0.01	0.50	0.4775	0.1314
0.01	1.00	0.4894	0.1671

Table 9: Accuracy and Kappa of SVM Radial on Testing Data

Accuracy	Kappa
0.50	0.19

```
      Accuracy          Kappa  AccuracyLower  AccuracyUpper   AccuracyNull
  4.982935e-01   1.890228e-01   4.396063e-01   5.570156e-01   4.266212e-01
AccuracyPValue  McnemarPValue
  7.956537e-03   8.984545e-06
```

When analysing the accuracy and the kappa statistics we can understand the effectiveness of the model of predicting the target variable. The accuracy is lower than desired at 0.50. However the kappa value can provide a more 0.19 which shows the model has slight agreement in predicting the correct target variable.

4.7 Random Forest

To create a statistical model for the contraceptive dataset, the randomForest package is used.

There are two drawbacks to the randomForest function. First, it simply can not handle missing values, so users must impute data before running it through the feature. The Second issue is the total number of levels for each categorical attribute is limited to 32 (Zhao 2013).

Both these issues have been resolved in the pre-processing stage of the data in Section 3

As a result, the procedure is implemented as follows:

Tuning the Parameters

A variety of parameters influence the training of certain models, impacting their performance significantly. Random Forest will tune the number of trees or branches that grow at each time split (ntree) and the number of variables randomly gathered to be sampled and tested at each split (ntry)

A grid search is used to tune, experimenting for every possible combination of parameter values

Tuning algorithm will facilitate retaining control of the training process and generate stronger results. There are several other approaches, but these two parameters are expected to have the greatest influence on model accuracy. We

Table 10: Accuracy and Kappa for RF on Training Data

mtry	Accuracy	Kappa
2	0.4792	0.1290
3	0.4970	0.1815
4	0.5071	0.2044

will train and test all 9 combinations of those parameters in this random forest with ntree set to 100, 500, 1000 and mtry set to 2, 3, 4.

You can also perform a grid search to tune over the parameters defined.

```
> library(caret)
> set.seed(123)
> mtrys <- c(2,3,4)
> tunegrid <- expand.grid(mtry = mtrys)
```

Fitting the Classifier

The caret package will use the random forest method to create a random forest classifier train the classifier with the training data set before testing the prediction of the model with the test set. This will be passed through the confusionMatrix function to collect the results below:

```
> set.seed(123)
> rf.fit <- train(ContraceptiveMethodUsed ~ ., data = train.set,
+                  method = "rf", trControl = control, tuneGrid = tunegrid)
> rf_prediction <- predict(rf.fit, test.set)
> cmrf <- confusionMatrix(rf_prediction, test.set$ContraceptiveMethodUsed)
```

```
> set.seed(123)
> library(xtable)
> print(xtable(rf.fit$results[,c(1,2,3)], digits = c(0,0,4,4)),
+       include.rownames = FALSE)
>
```

```
> set.seed(123)
> cmrf$overall
```

```
      Accuracy          Kappa  AccuracyLower  AccuracyUpper   AccuracyNull
  5.017065e-01   1.872399e-01   4.429844e-01   5.603937e-01   4.266212e-01
AccuracyPValue  McnemarPValue
  5.736811e-03   6.503033e-10
```

When analysing the accuracy and the kappa statistics we can understand the effectiveness of the model of predicting the target variable. The accuracy is lower at 0.50 suggesting the model is fifty- fifty in correctly classifying the contraceptive method used. However the kappa value is 0.19 which showing the model has slight agreement in predicting the correct target variable.

Table 11: Accuracy and Kappa of RF on Testing Data

Accuracy	Kappa
0.50	0.19

Table 12: Accuracy of LDA, SVM and RF on the testing set

model	Accuracy	Kappa
LDA	0.51	0.23
SVM with linear kernel	0.49	0.21
SVM with polynomial kernel	**0.52**	**0.24**
SVM with radial kernel	0.50	0.19
Random forest	0.50	0.19

5 Summary Of Results

5.1 Summary of the results

Table 12 shows the accuracy obtained by LDA, SVM and RF.

The results of the model show that the support vector machine with a polynomial kernel has performed the best with the accuracy of 0.52 and a kappa statistic of 0.24. Each model has a similar value for accuracy at around 0.5. This suggests that the model is performing better chance as with three classes, which would be around 33 percent without a model. Overall, the results indicate that the eight predictors only produce around a fifty percent chance of correctly classifying which contraceptive method a woman would use.

The results from the visualisation do suggest that the characteristics such as the womens education, age, number of children born and the standard of living index would provide an insight into the contraceptive method used do improve the predictions however as most the scores are left to be desired the model could do with some improvements.

These could be through additional research into other characteristics such as health data or a more diverse population. Although there is 1473 participants and increase in the amount of data could help benefit the training stages of each model.

Also, each experiment could be simplified into a binary classification to predict whether contraception is used or not, instead of differentiating between which type of contraceptive method used.

6 References

Dua, D., and Graff, C., (2019). UCI Machine Learning Repository. Contraceptive Method Choice Data Set. [online]. Irvine, CA: University of California, School of Information and Computer Science. Available from: https://archive.ics.uci.edu/ml/datasets/Contraceptive+Method+Choice

[Accessed 23 April 2021].

JAMES, G., WITTEN, D., HASTIE, T. and TIBSHIRANI, R., 2013.
An introduction to statistical learning: With applications in R. New York: Springer.

ZHAO, Y., 2013. R and Data Mining: Examples and Case Studies. San Diego: Elsevier Science and Technology.

```
> # create table for barplot for target variable
> set.seed(123)
> library(plyr)
> contraceptiveBPlot <- table(data$ContraceptiveMethodUsed)
> barplot(contraceptiveBPlot, main = "Bar plot for Contraceptive Method Used",
+         xlab = "Type of Contraceptive Method Used",
+         ylab = "Count of Contraceptive Method Used",
+         names.arg = c("1=No-use", " 2=Long-term", "3=Short-term"),
+         col = c("lightblue", "mistyrose", "lavender"))
>
```

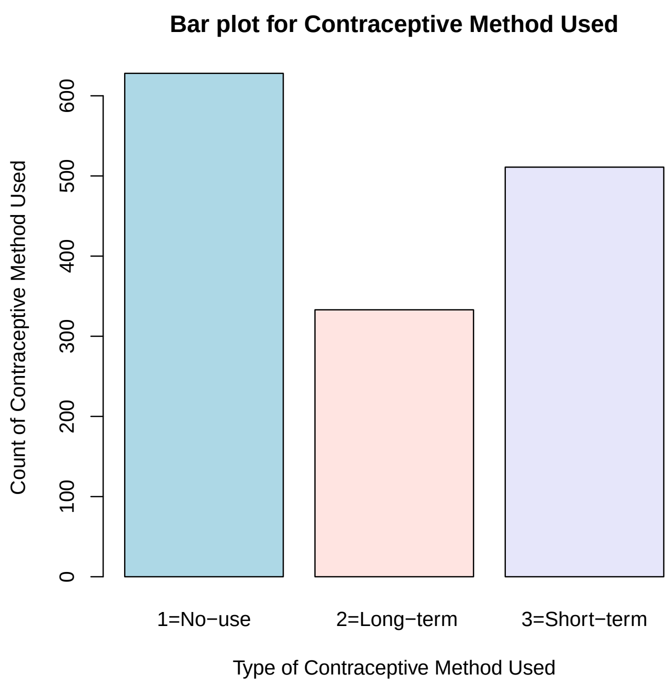

Figure 5: Bar plot for Contraceptive Method Used