Mixture of contaminated normal distributions with different variables inflation factor within classes

Jorge Sanchez

2024-02-01

Introduction

The traditional contaminated mixture model assumed that the contamination is the same for all variables within groups. The contaminated mixture model each group with a mixture of two normal distributions with two components. The first normal distribution models the non-contaminated samples while the second component models the contaminated samples. The contamination is control by two parameters which are the proportion of non-contaminated samples in each group α_g and the inflation factor η_g that is the same for all variables within group.

$$f(\mathbf{x}|\vartheta) = \sum_{g=1}^{G} \pi_g \left[\alpha_g \mathcal{N}(x|\boldsymbol{\mu}_g, \boldsymbol{\Sigma}_g) + (1-\alpha_g)\mathcal{N}(x|\boldsymbol{\mu}_g, \eta_g \boldsymbol{\Sigma}_g) \right]$$

There are cases where the assumption that the inflation factor is the same for all variables measured in an observation within groups might be unrealistic. It is possible that the characteristics or variables being contaminated are a few instead of a all variables. To model this scenario the previous equation can be modified by replacing the scalar η_g that is the inflation factor for all variables within group g by a matrix N_g which is a diagonal matrix where each element of the diagonal η_{gj} for $j = 1, \ldots, p$ represent the inflation factor for the corresponding variable.

$$f(\mathbf{x}|\vartheta) = \sum_{g=1}^{G} \pi_g \left[\alpha_g \mathcal{N}(x|\boldsymbol{\mu}_g, \boldsymbol{\Sigma}_g) + (1-\alpha_g)\mathcal{N}(x|\boldsymbol{\mu}_g, \dot{N}_g \boldsymbol{\Sigma}_g \dot{N}_g^T) \right]$$

$$\dot{N}_g = \begin{bmatrix} \sqrt{\eta_{g1}} & 0 & \ldots & 0 \\ 0 & \sqrt{\eta_{g2}} & \ldots & 0 \\ \ldots & \ldots & \ldots & \ldots \\ 0 & 0 & \ldots & \sqrt{\eta_{gp}} \end{bmatrix}$$

Simulation study: Comparison between mixtures with equal and different inflation factors within group

Overview

In this section, the behavior of contaminated mixture normal models assuming equal and different variables inflation factor within group is investigated. To generate the data the following process is conducted to generate three datasets where 75% of the samples composed the training subset and the remaining are part of the test subset.

a) A contaminated dataset of 100 samples in $p = 3$ dimensions and same variable inflation factor within group and $G = 1$;
b) A contaminated dataset of 100 samples in $p = 3$ dimensions with different variable inflation factor within group and $G = 1$;
c) A contaminated balanced dataset of 100 samples in $p = 3$ dimensions and $G = 2$ with equal proportions for both groups and one group with same variable inflation factor and another group with different variable inflation factor;

Dataset A

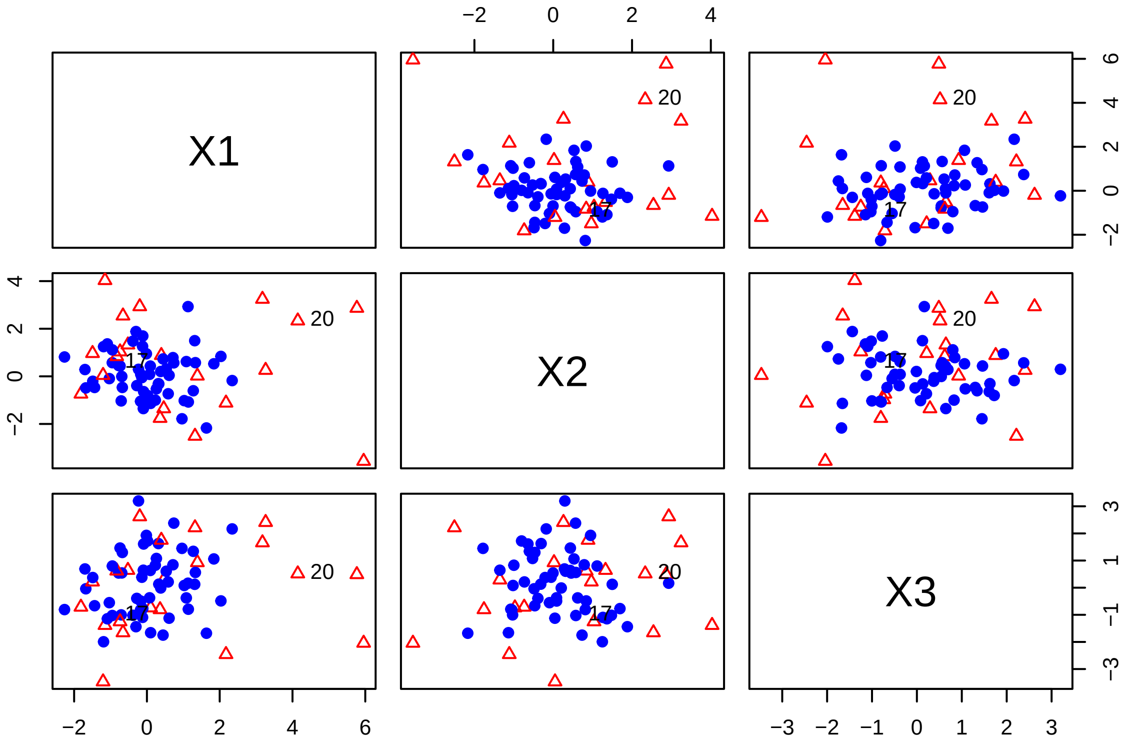
The parameters for dataset A are

$$\mu = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix}, \Sigma = I_3 = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}, \alpha = 0.8, \eta = 5$$

```
#pairs(DatasetA$Xtrain, col = c("red","blue")[DatasetA$vtrain + 1],
#      pch = ifelse(DatasetA$vtrain == 0,24,19), cex = 1)

pairs(DatasetA$Xtrain, panel = function(x,y, ...) {
  points(x,y,
         col = c("red","blue")[DatasetA$vtrain + 1],
         pch = ifelse(DatasetA$vtrain == 0,24,19),
         cex = 1,
         ...)
      text(x[indAreal_T0_P1_Testv],y[indAreal_T0_P1_Testv],
           labels=c(indAreal_T0_P1_Testv),pos = 4)

} )
```



Looking at the training data above, it is clear that the inflation factor is the same across all variables with a few contaminated samples further than the contaminated samples cloud. This dataset is fitted firstly with the contaminated mixture of normals model assuming equal variable inflation factor within group and secondly with the model assuming different variable inflation factor within group.

The parameter estimates were obtained by using the command Cnmixt until covergence from the library ContaminatedMixt. Next, the self-coded E-step was used to obtain the labels whether a sample is non-contaminated or contaminated. The function Cnmixt assumes equal variable inflation factor within group and their parameters are shown below.

$$\mu = \begin{bmatrix} 0.07 \\ 0.15 \\ 0.13 \end{bmatrix}, \Sigma = \begin{bmatrix} 1.05 & 0 & 0 \\ 0 & 1.05 & 0 \\ 0 & 0 & 1.05 \end{bmatrix}, \alpha = 0.78, \eta = 4.95$$

Using the real parameters and running e-step to predict ν values for Dataset A.

It is assumed the real parameter estimates are known and they are plug-in the self-coded E-step function to predict samples contamination in both training and test sets. The confusion matrix where the rows are the real values and the columns are the predicted ones shows that 52%(11), 50%(2) of the contaminated samples are correctly identified in the training and test set respectively.

```
knitr::kable(table(FactArealTrainv,FactpredArealTrainv), format = "markdown")
```

	Cont	Non-cont
Cont	11	10
Non-cont	4	50

```
#knitr::kable(df_A, format = "markdown")
format_table(df_ArealTrain)
```

```
##        Metric Equal_IF Different_IF
## 1    Accuracy     0.84         0.81
## 2   Precision     0.91         0.73
## 3      Recall     0.48         0.52
## 4 Sensitivity     0.48         0.52
## 5 Specificity     0.98         0.93
## 6    F1 score     0.62         0.61
```

```
knitr::kable(table(FactArealTestv,FactpredArealTestv), format = "markdown")
```

	Cont	Non-cont
Cont	2	2
Non-cont	0	21

```
#knitr::kable(df_A, format = "markdown")
format_table(df_ArealTest)
```

```
##        Metric Equal_IF Different_IF
## 1    Accuracy     0.92         0.92
## 2   Precision     1.00         1.00
## 3      Recall     0.50         0.50
## 4 Sensitivity     0.50         0.50
## 5 Specificity     1.00         1.00
## 6    F1 score     0.67         0.67
```

Using the real contamination information ν's and running m-step to obtain parameter estimates for Dataset A.

The parameter estimates obtained by using the real contamination information are

$$\mu = \begin{bmatrix} 0.06 \\ 0.11 \\ 0.17 \end{bmatrix}, \Sigma = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}, \alpha = 0.72, \dot{N} * \dot{N}^T = N = \begin{bmatrix} 7.46 & 0 & 0 \\ 0 & 5.14 & 0 \\ 0 & 0 & 3.41 \end{bmatrix}$$

It can be seen that the estimate for α is close to the estimate obtained by the model assuming equal variable inflation factor within group. Moreover, the estimates for η assuming different inflation factor are in the diagonal of matrix N and they are all different instead of all of them equal to the true parameter value that is 5. Nevertheless, X_2 is the variable con inflation factor closer to 5, while X_1 and X_3 have a higher and smaller inflation factors.

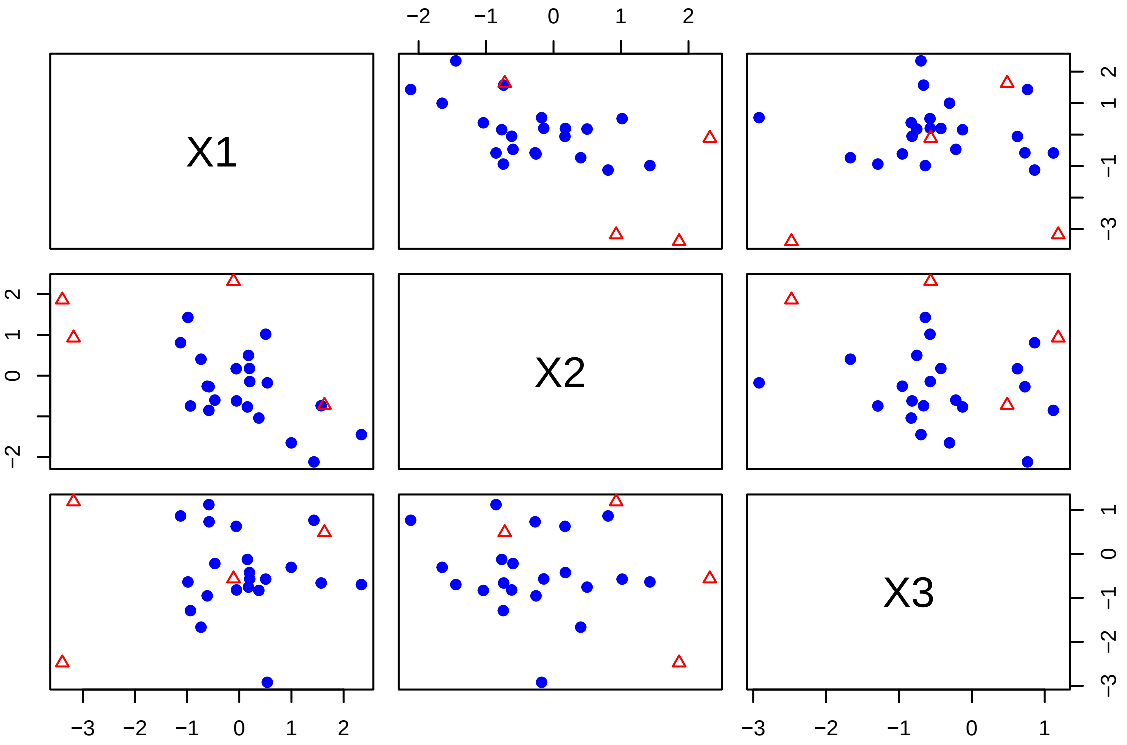
Predicting contamination information in samples by using parameters estimates for a model assuming equal variable inflation factors as initial values for Dataset A.

The initial values for the parameters of the model were taken from the contaminated mixture model produced by the function Cnmixt from the ContaminatedMixt package that assumes same variable inflation factor within group. Next, an iteration of a self-coded m-step that assumes different variable inflation factor within group for EII model was run to obtain the estimated parameters shown below. It is possible to see that there are some differences between the estimates obtained for both models. The main difference between the estimates is observed in the values that the model with different variables inflation factor take for α which suggest a 51% of non-contaminated samples and implies a greater contamination that the other model. Also, the inflation factors in the diagonal of the matrix $\dot{N}$ are different and much smaller than 4.95.

$$\mu = \begin{bmatrix} 0.07 \\ 0.16 \\ 0.13 \end{bmatrix}, \Sigma = \begin{bmatrix} 1.31 & 0 & 0 \\ 0 & 1.31 & 0 \\ 0 & 0 & 1.31 \end{bmatrix}, \alpha = 0.93, \dot{N} * \dot{N}^T = N = \begin{bmatrix} 14.07 & 0 & 0 \\ 0 & 6.53 & 0 \\ 0 & 0 & 1.93 \end{bmatrix}$$

Next, a self-coded function E-step was used to obtain the estimates of labels corresponding whether the observations are non-contaminated or contaminated in the test set. It was observed that the estimates for α and N were affected for the choose of the initial values of ν_i which denotes if the i^{th} sample is non-contaminated if $\nu_i = 1$ otherwise is 0. The pairs plot of the samples in the test subset shows a little more dispersion for the pair X_1 and X_2 than for the pair X_1 and X_3.

```
pairs(DatasetA$Xtest, col = c("red","blue")[DatasetA$vtest + 1],
      pch = ifelse(DatasetA$vtest == 0,24,19), cex = 1)
```



The confusion matrix where the rows are the actual values and the columns the predicted values for the model assuming same variable inflation factor within group , shows that it was possible to identify half of the contaminated observations and all the non-contaminated observations correctly.

```r
#knitr::kable(t_AEq, format = "markdown")
knitr::kable(table(Fact_ATrainv,pred_AEqTrain), format = "markdown")
```

	Cont	Non-cont
Cont	10	11
Non-cont	1	53

```r
#knitr::kable(t_AEq, format = "markdown")
knitr::kable(table(Fact_ATestv,pred_AEqTest), format = "markdown")
```

	Cont	Non-cont
Cont	2	2
Non-cont	0	21

However, taking a look of the performance of the model assuming different variables inflation factor within group is also able to identify half of the non-contaminated samples and misclassified 3 non-contaminated samples as contaminated. As the data is generated for a model with same variable inflation factor within group, there is not surprise that it overcame their counterpart that assume different variable inflation factors within group. It is possible to confirm this result taking a look to the other metrics specially F1_Score.

```r
knitr::kable(t_ADif, format = "markdown")
```

	1
0	4
1	21

```r
#knitr::kable(df_A, format = "markdown")
format_table(df_A)
```

```
##        Metric Equal_IF Different_IF
## 1    Accuracy     0.92         0.84
## 2   Precision     1.00             
## 3      Recall     0.50             
## 4 Sensitivity     0.50             
## 5 Specificity     1.00         1.00
## 6    F1 score     0.67             
```

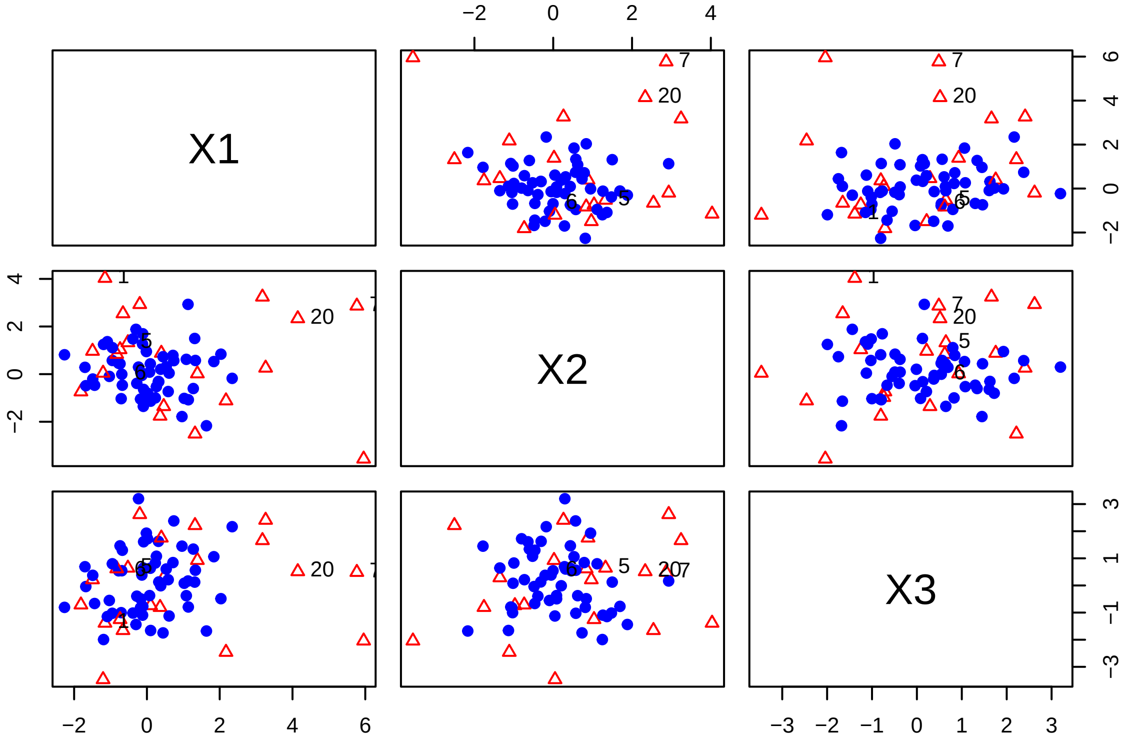
Dataset B

The parameters for dataset B are

$$\mu = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix}, \Sigma = I_3 = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}, \alpha = 0.8, \dot{N}_g = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \sqrt{5} & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

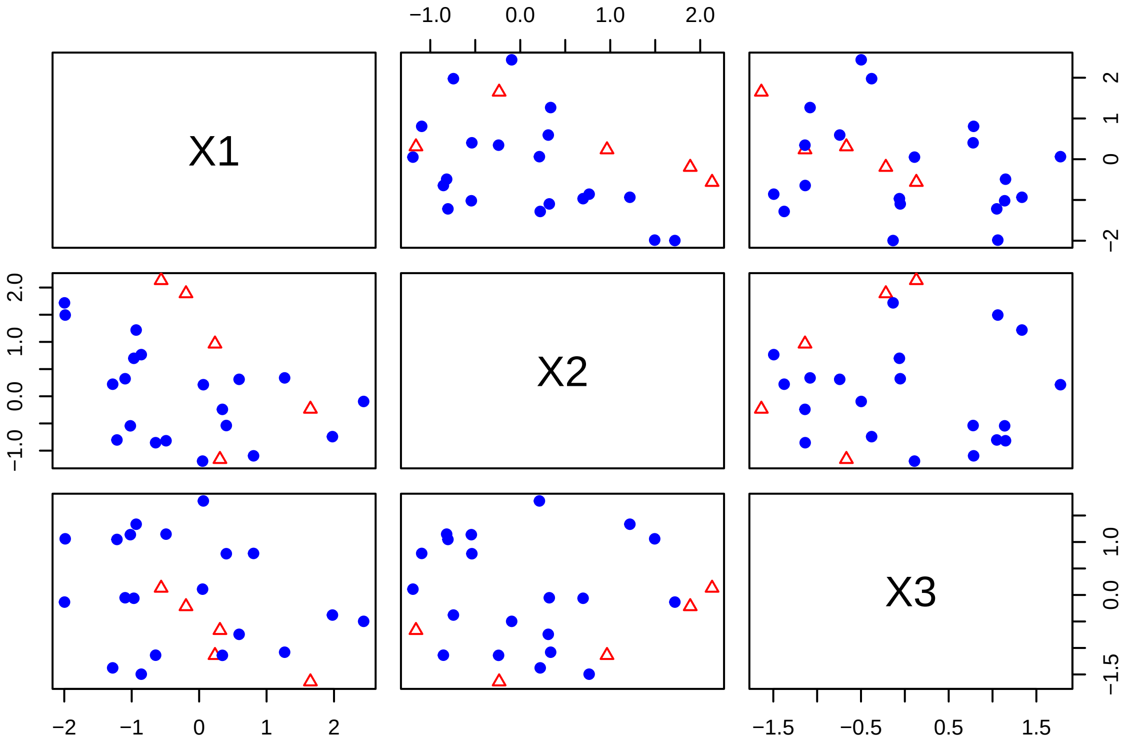
The main different between the second dataset versus the previous one is that the data is generated from allowing different variables inflation factor within the group. It is possible to see that X_2 has an inflation factor higher than X_1 and X_3. The experiment consist of fitting first the variable assuming the same variable inflation factor in the group and next allowing different variable inflation factor within the group and comparing the results obtained.

```
pairs(DatasetA$Xtrain, panel = function(x,y, ...) {
  points(x,y,
         col = c("red","blue")[DatasetA$vtrain + 1],
         pch = ifelse(DatasetA$vtrain == 0,24,19),
         cex = 1,
         ...)
      text(x[indBreal_T0_P1_Testv],y[indBreal_T0_P1_Testv],
           labels=c(indBreal_T0_P1_Testv),pos = 4)
})
```



It is possible to observe that in the training and test subset there is a much higher dispersion in the variable $X2$ in comparison with X_1 and X_3.

```
pairs(DatasetB$Xtest, col = c("red","blue")[DatasetB$vtest + 1],
      pch = ifelse(DatasetB$vtest == 0,24,19), cex = 1)
```



Similar procedure is carry out fitting the data using Cnmixt function for model EII and assuming same variable inflation factor in the group and the parameters obtained are:

$$\mu = \begin{bmatrix} 0.06 \\ -0.08 \\ 0.07 \end{bmatrix}, \Sigma = \begin{bmatrix} 1.22 & 0 & 0 \\ 0 & 1.22 & 0 \\ 0 & 0 & 1.22 \end{bmatrix}, \alpha = 0.99, \eta = 1$$

Using the real parameters and running e-step to predict ν values for Dataset B.

It is assumed that the real parameters are known and they are used as an input for a self-coded E-step as it was done previously to produce estimates for ν's. From the tables below it is possible to see that half of the contaminated samples and any of them were identified in the train and test subsets respectively.

```
knitr::kable(table(FactBrealTrainv,FactpredBrealTrainv), format = "markdown")
```

	Cont	Non-cont
Cont	5	10
Non-cont	2	58

```
#knitr::kable(df_A, format = "markdown")
format_table(df_BrealTrain)
```

```
##          Metric Equal_IF Different_IF
## 1    Accuracy     0.80         0.84
## 2   Precision                  0.71
## 3      Recall                  0.33
## 4 Sensitivity                  0.33
## 5 Specificity     1.00         0.97
## 6    F1 score                  0.45
```

```
knitr::kable(table(FactBrealTestv,FactpredBrealTestv), format = "markdown")
```

	Cont	Non-cont
Cont	0	5
Non-cont	0	20

```
#knitr::kable(df_A, format = "markdown")
format_table(df_BrealTest)
```

```
##        Metric Equal_IF Different_IF
## 1    Accuracy     0.80         0.80
## 2   Precision
## 3      Recall
## 4 Sensitivity
## 5 Specificity     1.00         1.00
## 6    F1 score
```

Using the real contamination information ν's and running m-step to obtain parameter estimates for Dataset B

The parameter estimates obtained by using the real contamination information are

$$\mu = \begin{bmatrix} 0.1 \\ -0.07 \\ 0.11 \end{bmatrix}, \Sigma = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}, \alpha = 0.8, \dot{N} * \dot{N}^T = N = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 4.15 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Predicting contamination information in samples by using parameters estimates for a model assuming equal variable inflation factors as initial values for Dataset B.

It is possible to see that the estimations for μ, Σ are quite close, while there is a noticeable different in the estimates for α since the estimate parameter assumes there is less contaminated samples than what the true parameter states. The estimation for η could be confused by the estimated for λ as the model fitted is EII. It is possible to see that the value the estimate takes is 1.22 (see diagonal of the variance covariance matrix).

Next, the same data is fit with the model allowing different variables inflation factor within group and the parameter estimates are shown below:

$$\mu = \begin{bmatrix} 0.06 \\ -0.08 \\ 0.07 \end{bmatrix}, \Sigma = \begin{bmatrix} 1.22 & 0 & 0 \\ 0 & 1.22 & 0 \\ 0 & 0 & 1.22 \end{bmatrix}, \alpha = 1, \dot{N} * \dot{N}^T = N = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 5.64 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

The parameters obtained for μ, Σ, and λ are very similar as the initial values for the parameters come from the estimates for the model EII with same variable inflation factor within group. The main difference between the estimates produced by these two models lies in the estimates for α that is 0.51 which means that the inflation factor is being overestimated as the percentage of non-contaminated samples is 0.9. Also, It is possible to see that the variable inflation factor estimates are different for variables X_1, X_2 and X_3. Also, the estimate for the variable inflation factor corresponding to X_2 is smaller that the true variable.

Taking a look of the performance of the model assuming same variable inflation factor within group we can see that it is not able to identify any of the 5 contaminated samples. The plot of the test subset shows that the contaminated samples are quite close to the non-contaminated samples.

```
knitr::kable(t_BEq, format = "markdown")
```

	1
0	5
1	20

The model allowing different variables inflation factor within group performs better than its counterpart that assumes same variable inflation factor in the group. The former is able to detect 2 of the 5 contaminated samples.

```
knitr::kable(t_BDif, format = "markdown")
```

	1
0	5
1	20

The rest of the metrics confirms that when the data was coming from different variables inflation factors the model allowing different inflation factor seems to be more appropiate than assuming equal inflation factor within group. Nevertheless, more exploration is needed.

```
#knitr::kable(df_A, format = "markdown")
format_table(df_B)
```

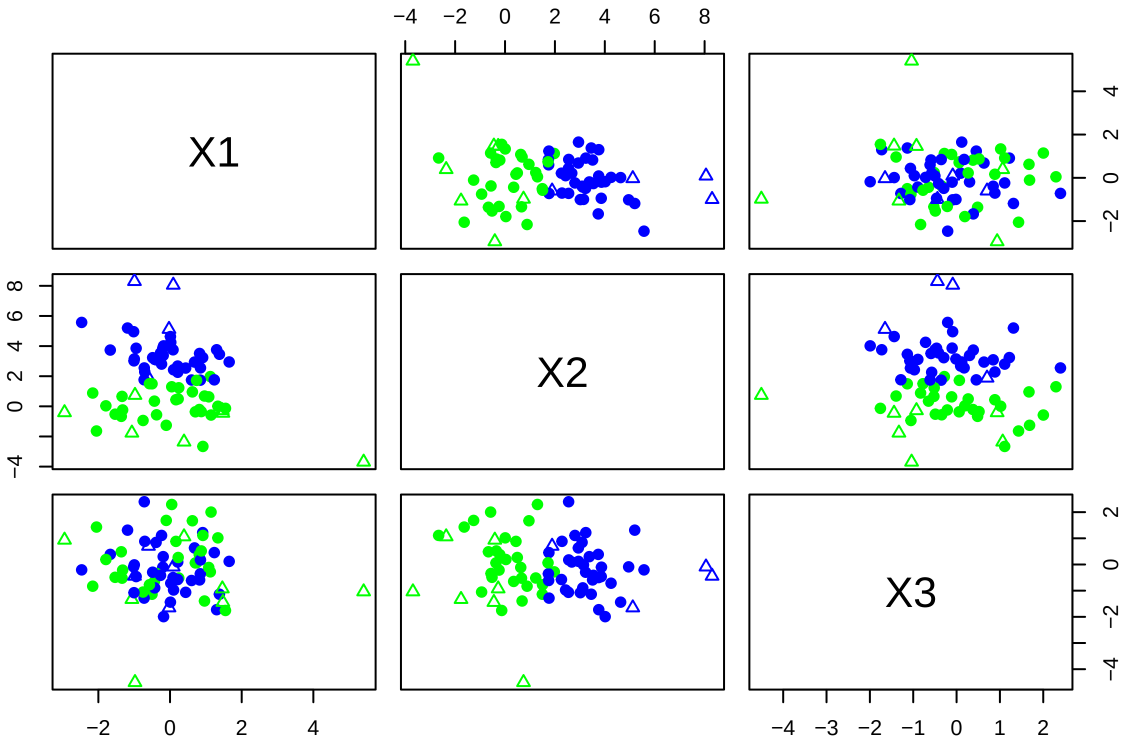
```
##        Metric Equal_IF Different_IF
## 1    Accuracy     0.80         0.80
## 2   Precision
## 3      Recall
## 4 Sensitivity
## 5 Specificity     1.00         1.00
## 6    F1 score
```

Dataset C

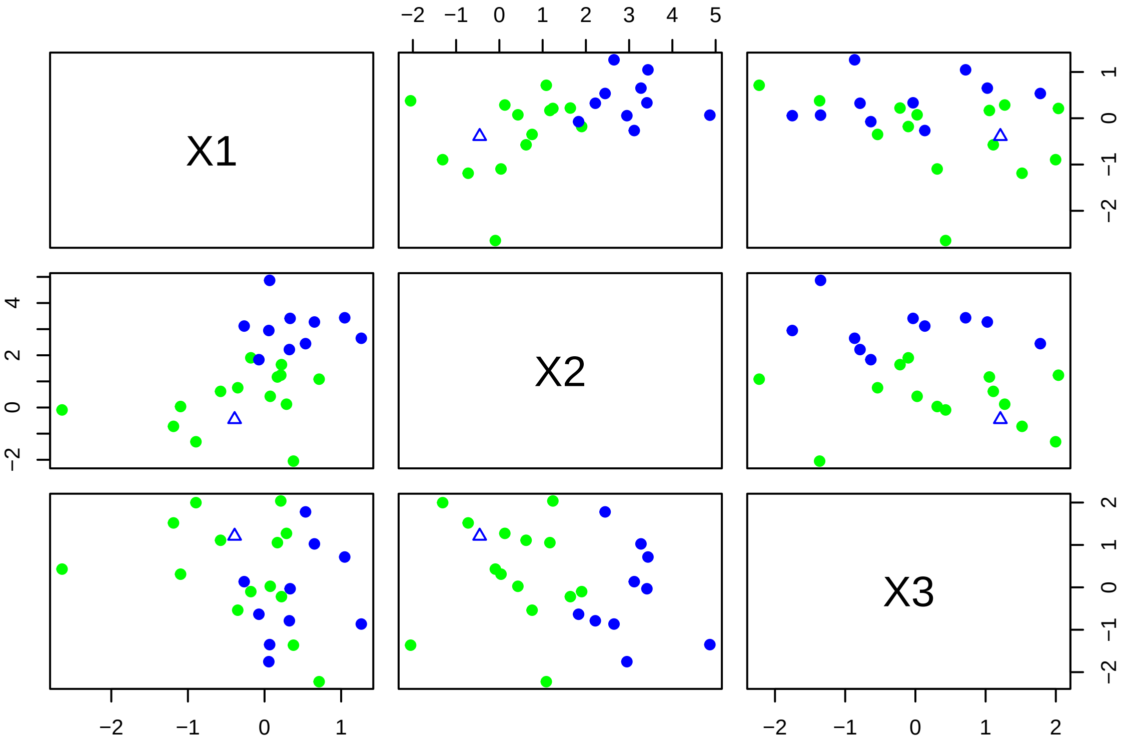
The parameters for dataset C

$$\mu_1 = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix}, \mu_2 = \begin{pmatrix} 0 \\ 3 \\ 0 \end{pmatrix} \Sigma_1 = I_3, \Sigma_2 = I_3, \alpha_1 = 0.8, \alpha_2 = 0.9, \eta_1 = 5, N_2 = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \sqrt{10} & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

```
pairs(DatasetC$Xtrain, col = c("green","blue")[DatasetC$ltrain],
      pch = ifelse(DatasetC$vtrain == 0,24,19), cex = 1)
```



```
pairs(DatasetC$Xtest, col = c("green","blue")[DatasetC$ltest],
      pch = ifelse(DatasetC$vtest == 0,24,19), cex = 1)
```


X1
X2
X3
−2 −1 0 1 2 3 4 5
−2 −1 0 1
−2 0 2 4
−2 −1 0 1 2