LAB05 REPORT

```
MVG cassifier - Error rate: 7.00%
Tied gaussian - Error rate: 9.30%
Naive Bayes Gaussian - Error rate: 7.20%
LDA classifier - Error rate: 9.3%
LDA classifier with PCA pre processing - Error rate: 9.2%
```

Here are the error rates for the various classifiers, this result suggest the class are separable well enough to be distinguished with a threshold (in the case of the LDA classifier), but the probabilistic interpretation of the MVG and the approximation of the classes as gaussian distribution works a little bit better, giving a little bit better result.
We can also conclude that the various features are pretty unrelated because the Naive Bayes Gaussian classifier (which assumes features independence) only has a .20 increment in the error rate compared to the MVG. On the other hand, Tied covariances model performs worse than MVG and Naive Bayes, this is probably due to the approximation of the covariance matrix as a single one for all classes.

```
Covariance matrix true:
[[ 1.44809527e+00 -1.47222433e-02  5.57010301e-03  1.57415883e-02  1.94971163e-02 -1.76682539e-04]
 [-1.47222433e-02  5.53390796e-01 -1.12168681e-02 -9.06473359e-03 -1.46589901e-02  1.63492048e-02]
 [ 5.57010301e-03 -1.12168681e-02  5.57480229e-01  2.75609663e-02 -3.76966451e-03 -1.45976943e-02]
 [ 1.57415883e-02 -9.06473359e-03  2.75609663e-02  5.69657013e-01 -1.16983404e-02  3.49931863e-02]
 [ 1.94971163e-02 -1.46589901e-02 -3.76966451e-03 -1.16983404e-02  1.34201767e+00  1.69454096e-02]
 [-1.76682539e-04  1.63492048e-02 -1.45976943e-02  3.49931863e-02  1.69454096e-02  1.30371880e+00]]
Covariance matrix fake:
[[ 6.00956506e-01  5.15866517e-05  1.90589145e-02  1.92529876e-02  1.28039402e-02 -1.34721598e-02]
 [ 5.15866517e-05  1.44722543e+00 -1.61340110e-02 -1.58561474e-02 -2.64529141e-02  2.29139833e-02]
 [ 1.90589145e-02 -1.61340110e-02  5.65348901e-01 -1.84344435e-03 -6.91446277e-03  1.68928322e-02]
 [ 1.92529876e-02 -1.58561474e-02 -1.84344435e-03  5.41615202e-01  5.25171375e-03  1.35717775e-02]
 [ 1.28039402e-02 -2.64529141e-02 -6.91446277e-03  5.25171375e-03  6.96067641e-01  1.58438399e-02]
 [-1.34721598e-02  2.29139833e-02  1.68928322e-02  1.35717775e-02  1.58438399e-02  6.86519710e-01]]
```

As we can see from the covariance matrixes, the variance value are far greater compared to the covariance with the other features, this suggest us the features are NOT strongly correlated.

```
Pearson correlation for class true:
[[ 1.00000000e+00 -1.64459687e-02  6.19940380e-03  1.73317836e-02  1.39859734e-02 -1.28588787e-04]
 [-1.64459687e-02  1.00000000e+00 -2.01948630e-02 -1.61447883e-02 -1.70101823e-02  1.92481371e-02]
 [ 6.19940380e-03 -2.01948630e-02  1.00000000e+00  4.89072205e-02 -4.35821698e-03 -1.71229097e-02]
 [ 1.73317836e-02 -1.61447883e-02  4.89072205e-02  1.00000000e+00 -1.33794547e-02  4.06054941e-02]
 [ 1.39859734e-02 -1.70101823e-02 -4.35821698e-03 -1.33794547e-02  1.00000000e+00  1.28109397e-02]
 [-1.28588787e-04  1.92481371e-02 -1.71229097e-02  4.06054941e-02  1.28109397e-02  1.00000000e+00]]
Pearson correlation for class fake:
[[ 1.00000000e+00  5.53156127e-05  3.26977873e-02  3.37466904e-02  1.97968638e-02 -2.09743833e-02]
 [ 5.53156127e-05  1.00000000e+00 -1.78367604e-02 -1.79095288e-02 -2.63560127e-02  2.29882544e-02]
 [ 3.26977873e-02 -1.78367604e-02  1.00000000e+00 -3.33139656e-03 -1.10223563e-02  2.71155043e-02]
 [ 3.37466904e-02 -1.79095288e-02 -3.33139656e-03  1.00000000e+00  8.55322509e-03  2.22569065e-02]
 [ 1.97968638e-02 -2.63560127e-02 -1.10223563e-02  8.55322509e-03  1.00000000e+00  2.29196624e-02]
 [-2.09743833e-02  2.29882544e-02  2.71155043e-02  2.22569065e-02  2.29196624e-02  1.00000000e+00]]
```

The pearson correlation confirms what we already saw from the covariance matrixes. this means the Naive Bayes Classifier is a viable option for out classification task, and this is also confirmed by the error rates we get.

As seen from laboratory 4, features 1-2-3-4 can be correctly modeled as gaussians, while 5-6 are not well represented using gaussians. We now try to analyze the model performance while only using features 1-2-3-4.

```
MVG classifier with 4 features - Error rate: 7.95%
Tied gaussian with 4 features - Error rate: 9.50%
Naive Bayes Gaussian with 4 features - Error rate: 7.65%
```

Excluding the last two features does not help the classification; in fact it makes the classification worse! Despite the inaccuracy of the assumption, the gaussian models are still able top extract useful informations to improve classification accuracy.

```
MVG classifier with feature 1-2 - Error rate: 36.50%
Tied gaussian with feature 1-2 - Error rate: 49.45%
Naive Bayes Gaussian with feature 1-2 - Error rate: 36.30%
MVG classifier with feature 3-4 - Error rate: 9.45%
Tied gaussian with feature 3-4 - Error rate: 9.40%
Naive Bayes Gaussian with feature 3-4 - Error rate: 9.45%
```

Using only the first 2 features greatly impacts our classifier: despite them being described very well by gaussian models, the error rates go up by a lot. Using only the feature 3-4 the error rates don't increase too much.
This can be explained by looking at the gaussian distribution of the features: the first two features have very similar distribution and it’s difficult for our classifier to find a suitable threshold, whereas in the distribution of features 3-4 there is a clear distinction between the two distribution and the chosen threshold is a lot more effective.
I would like to bring our attention to how the Tied gaussian classifier behaves: using only the first two features, we see a big error (almost 50%) and is effectively the worst, while using features 3-4 the error goes down to normal ranges and it’s actually the best.
This is due to the variances we observed in the Laboratory 4: the first two features have very different variances while having the same mean: this is shown by the tied gaussian model error rate, while the features 3-4 have a very similar variance, and the tied gaussian model works much better!

```
---------PCA (m=0)--------------
MVG classifier with PCA (m=0) pre processing - Error rate: 49.60%
Tied gaussian with PCA (m=0) pre processing - Error rate: 50.40%
Naive Bayes Gaussian with PCA (m=0) pre processing - Error rate: 50.40%
---------PCA (m=1)--------------
MVG classifier with PCA (m=1) pre processing - Error rate: 9.25%
Tied gaussian with PCA (m=1) pre processing - Error rate: 9.35%
Naive Bayes Gaussian with PCA (m=1) pre processing - Error rate: 9.25%
---------PCA (m=2)--------------
MVG classifier with PCA (m=2) pre processing - Error rate: 8.80%
Tied gaussian with PCA (m=2) pre processing - Error rate: 9.25%
Naive Bayes Gaussian with PCA (m=2) pre processing - Error rate: 8.85%
---------PCA (m=3)--------------
MVG classifier with PCA (m=3) pre processing - Error rate: 8.80%
Tied gaussian with PCA (m=3) pre processing - Error rate: 9.25%
Naive Bayes Gaussian with PCA (m=3) pre processing - Error rate: 9.00%
---------PCA (m=4)--------------
MVG classifier with PCA (m=4) pre processing - Error rate: 8.05%
Tied gaussian with PCA (m=4) pre processing - Error rate: 9.25%
Naive Bayes Gaussian with PCA (m=4) pre processing - Error rate: 8.85%
---------PCA (m=5)--------------
MVG classifier with PCA (m=5) pre processing - Error rate: 7.10%
Tied gaussian with PCA (m=5) pre processing - Error rate: 9.30%
Naive Bayes Gaussian with PCA (m=5) pre processing - Error rate: 8.75%
```

Now we analyze the effect of PCA as pre-processing: the best results are obtained using PCA with an m value equal to 5, but it's still a little bit worse than no preprocessing.

Overall the best model for our dataset is the standard MVG