

---

# Face Recognition

---

Computer Vision

*University of Barcelona*

December 27, 2015

*Author:*

CAMPS SEREIX, Julià

# 1 Gender Recognition



Figure 1: Content in ARFace.internal(:,1).

As we can observe in this figure, the content on the elements of ARFace.internal are face images, as I commented on the questions answers.



Figure 2: Relation between the contents in ARFace.internal and ARFace.person.

As we can observe in this figure, the content on the elements of ARFace.person are the subject labels for each of the images present in ARFace.internal, as I commented on the questions answers.

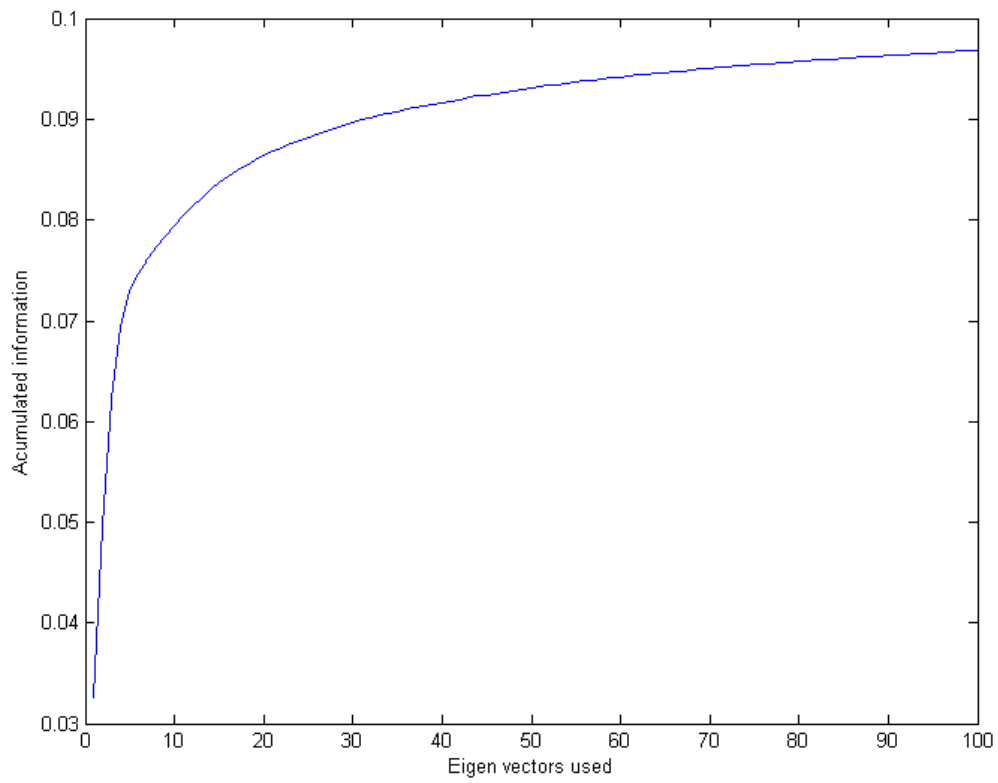


Figure 3: Accumulated information when increasing the number of eigen-vectors for the PCA representation.

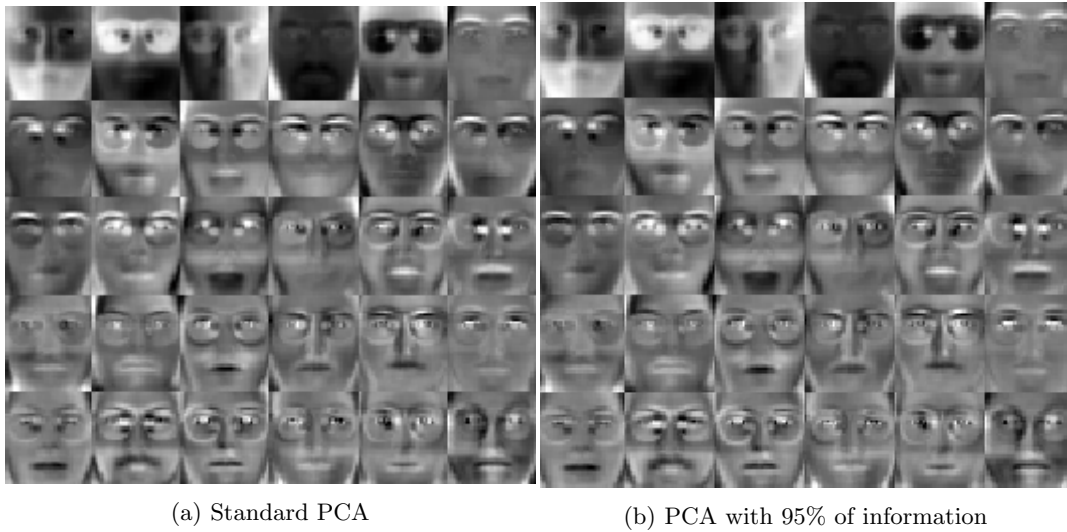


Figure 4: Eigen-faces retrieved for applying PCA.

<i>Algorithm</i>	Mean results obtained when classifying with KNN	
	<b>Accuracy</b>	<b>Best ‘K’ value for knn</b>
	PCA	Acc: 84.03                      7
	PCA 95%	84.8                                7
	LDA	99.82                               3

Table 1: Results obtained when applying knn for subject recognition with the different components reduction algorithms.

As we can observe in the best results table, the best parameter k when using the two variations of the number of PCA’s eigen-vectors, the best k parameter evaluated with knn when doing cross-validation with 10 folds, is 7 in both cases. But when using the data transformed by LDA, we can observe than the best k value is 3.

The LDA gives the best performance results when we apply knn for classifying the features provided by it. This is due than this algorithm is a supervised algorithm, which means, than it’s using class awareness for deciding which dimensions are better for representing the target data, in order to be able to group better the elements from the same classes. In all cases the accuracy for gender classification after using the factor analysis techniques has shown to be very high, this means than the training and testing data is highly representative for our problem.

I noticed an undesirable decision in the cross-validation code. I saw than for doing the k-folds is using the subject labels, which actually will imply than the persons on the training and testing sets will be different people! This is very undesirable, because the training model would not be able to fit well the testing data if the people on the faces dataset were very different among them. Random split of the instances would be the best strategy in order to avoid performing an wrong trained model and to obtain wrong conclusions. Actually, this strategy for splitting will lead to more wrong or at least not good cases.

As the number of subjects is not multiple of 10, we will have always a fold with 5 subjects more than the rest of folds, which seems to be a low number, but actually we are talking than one fold will be 1.625 times the rest. This fact should alarm us, because is a requirement for cross-validation than the folds are from the same size. In order to try to avoid unbalanced iterations during the process.

And also will be unable to handle subject classification, due than there is no representation of the same subjects on the training and testing sets.

## 2 Subject Recognition

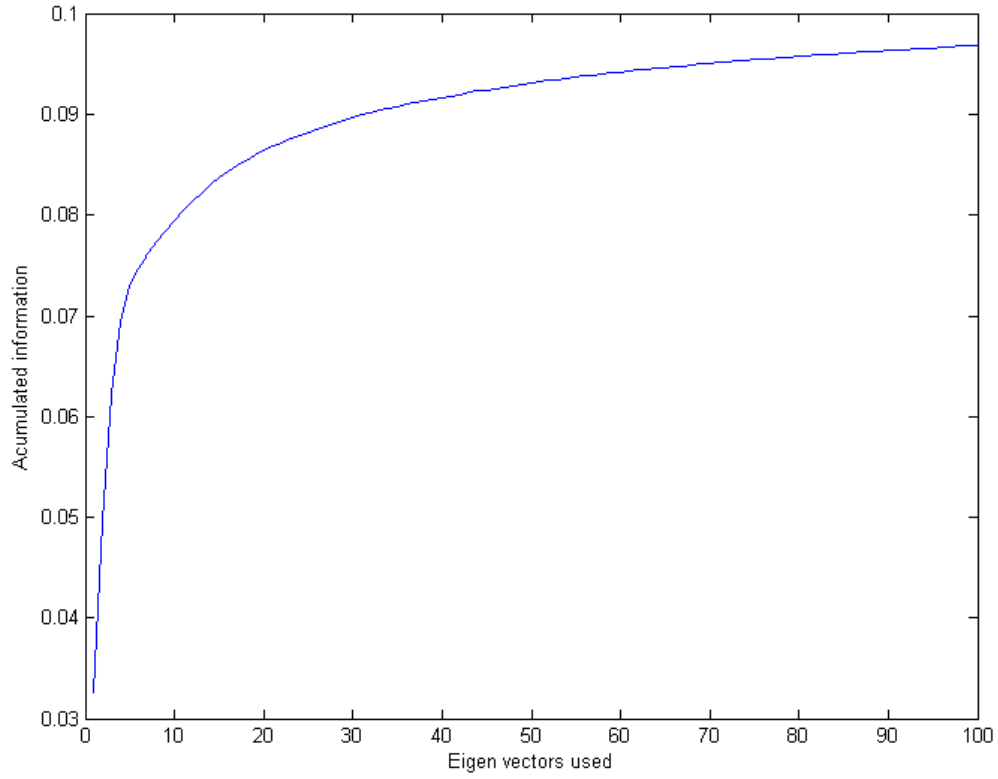


Figure 5: Accumulated information when increasing the number of eigen-vectors for the PCA representation.

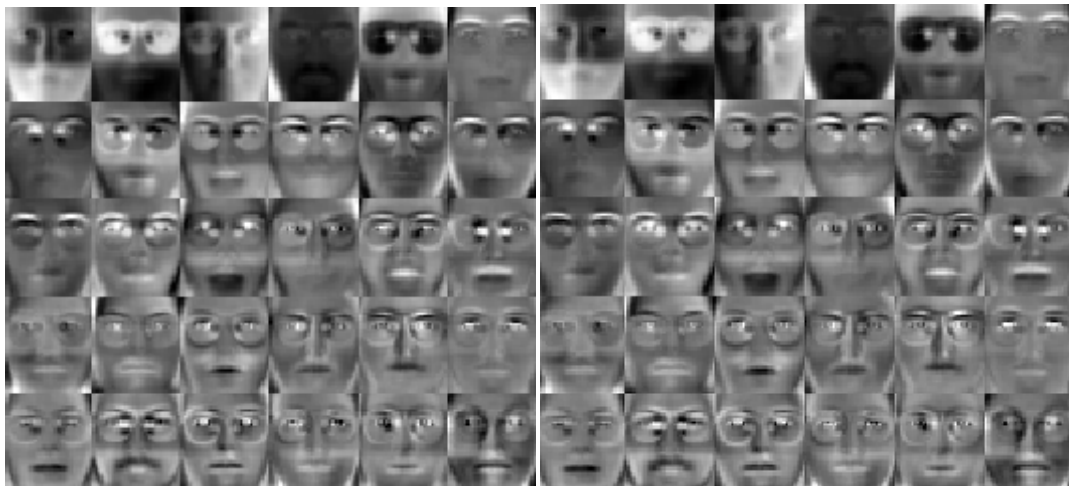


Figure 6: Eigen-faces retrieved for applying PCA.

	Mean results obtained when classifying with KNN	
	Accuracy	Best 'K' value for knn
PCA	26.02	1
PCA 95%	36.38	1
LDA	56.15	7

Table 2: Results obtained when applying knn for subject recognition with the different components reduction algorithms.

As we can observe in the best results table, the best parameter k when using the two variations of the number of PCA's eigen-vectors, the best k parameter evaluated with knn when doing cross-validation with 10 folds, is 1 in both cases. But when using the data transformed by LDA, we can observe than the best k value is 7.

As we already commented on the previous section, the strategy for the cross-validation is the first thing to change in order to be able to apply subject classification using the previous code with knn on the different feature selection strategies. For handling this problem I implemented a new cross-validation function.

I also have changed all references to the labels, to be using the subjects instead.

Which actually has been the tougher part from my point of view, has been to understand and build an 85x85 confusion matrix, and to fit it on the report document. But apart from this, the other changes on the code apart from rewriting the cross-validation function completely have been meaningful.

As we can observe, the features being better classified are the LDA, as when we performed the experiments for gender recognition. In this case we have much lower accuracy rates, this is due the number of classes than we actually have, which is much higher than in the previous experiments, which had just 2 classes, when now we are actually working with 85. Actually the number of classes is the reason for which we are having so bad results, because we are facing a very unbalanced problem. On some of the iterations of the cross-validation process we even don't have representation of some of the evaluated subjects.

From this we conclude than the dataset size is too small in relation with the number of classes being classified. All this can be also observed from the following pages where I have attached the confusion matrixes generated when executing the new cross-validation matlab function.

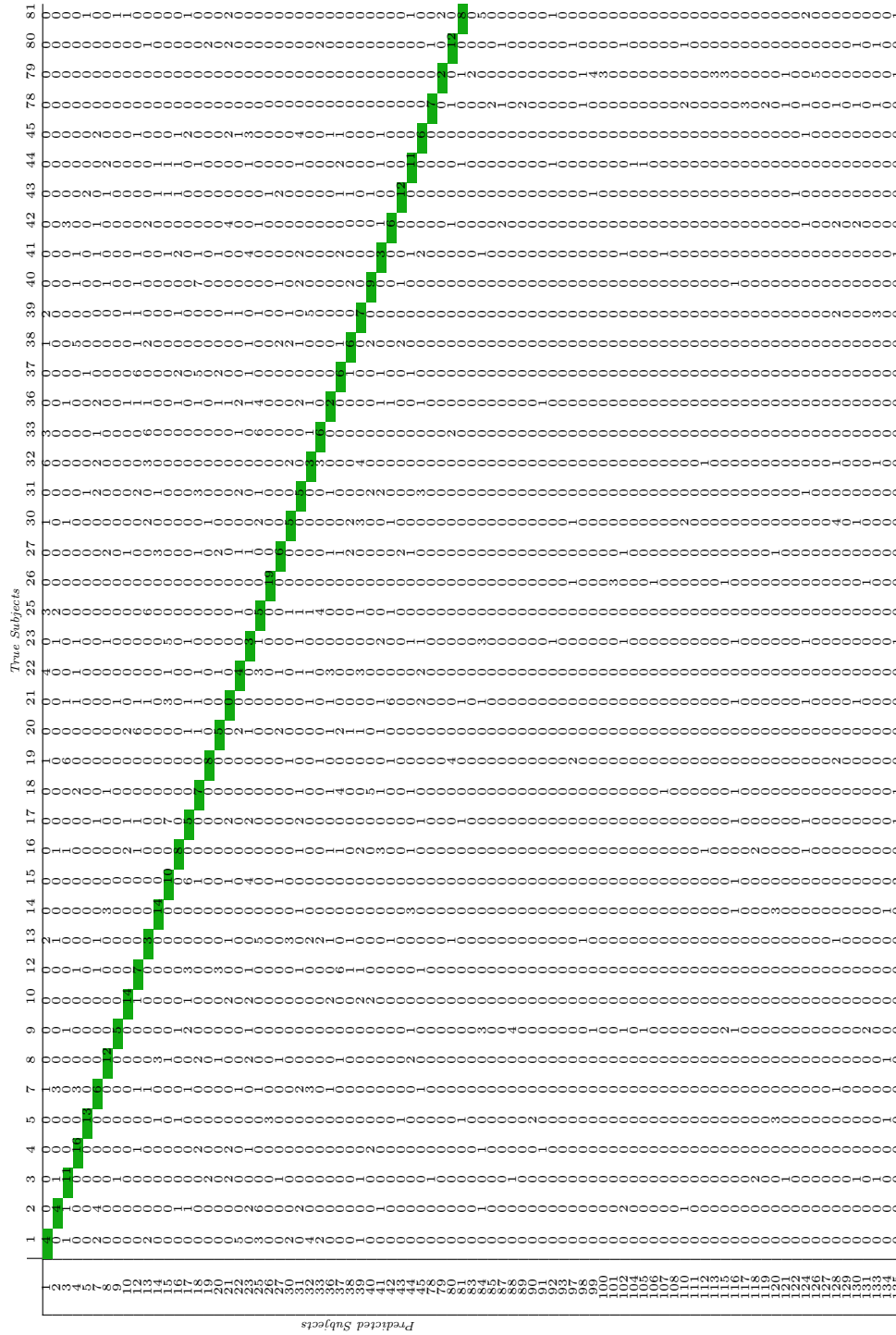


Table 3: Confusion matrix using PCA

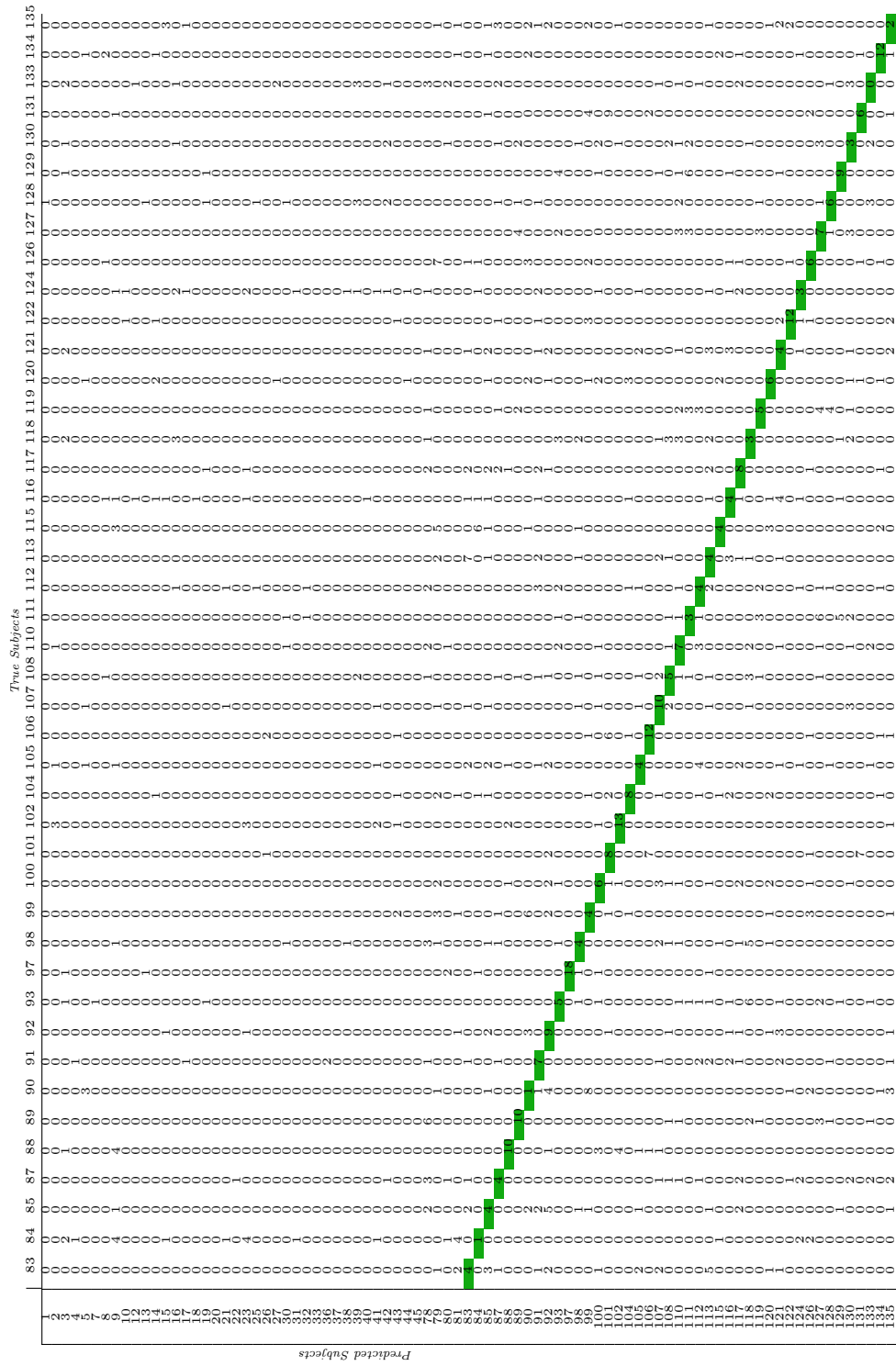


Table 4: Confusion matrix using PCA



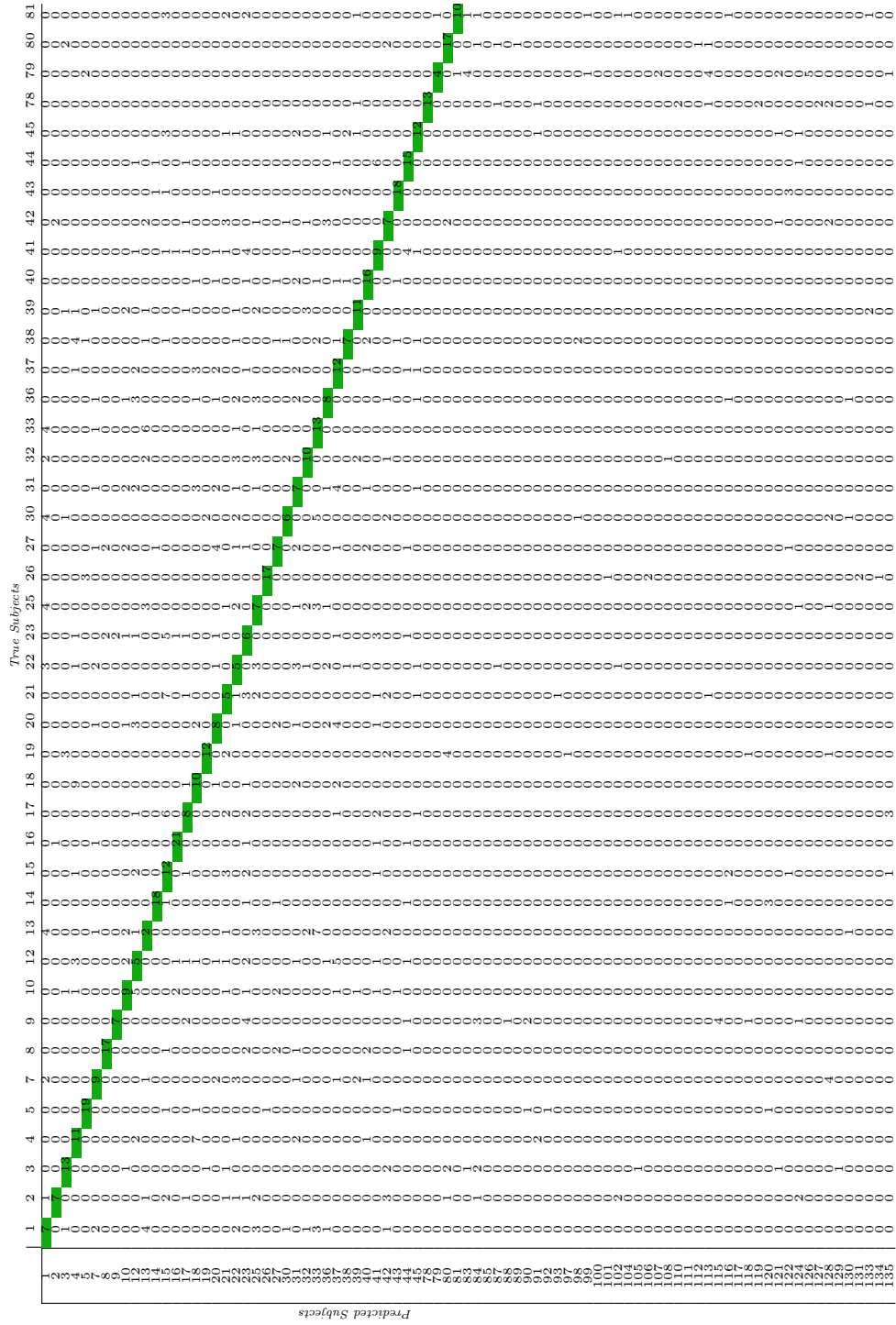


Table 5: Confusion matrix using PCA representing the 95% of initial information

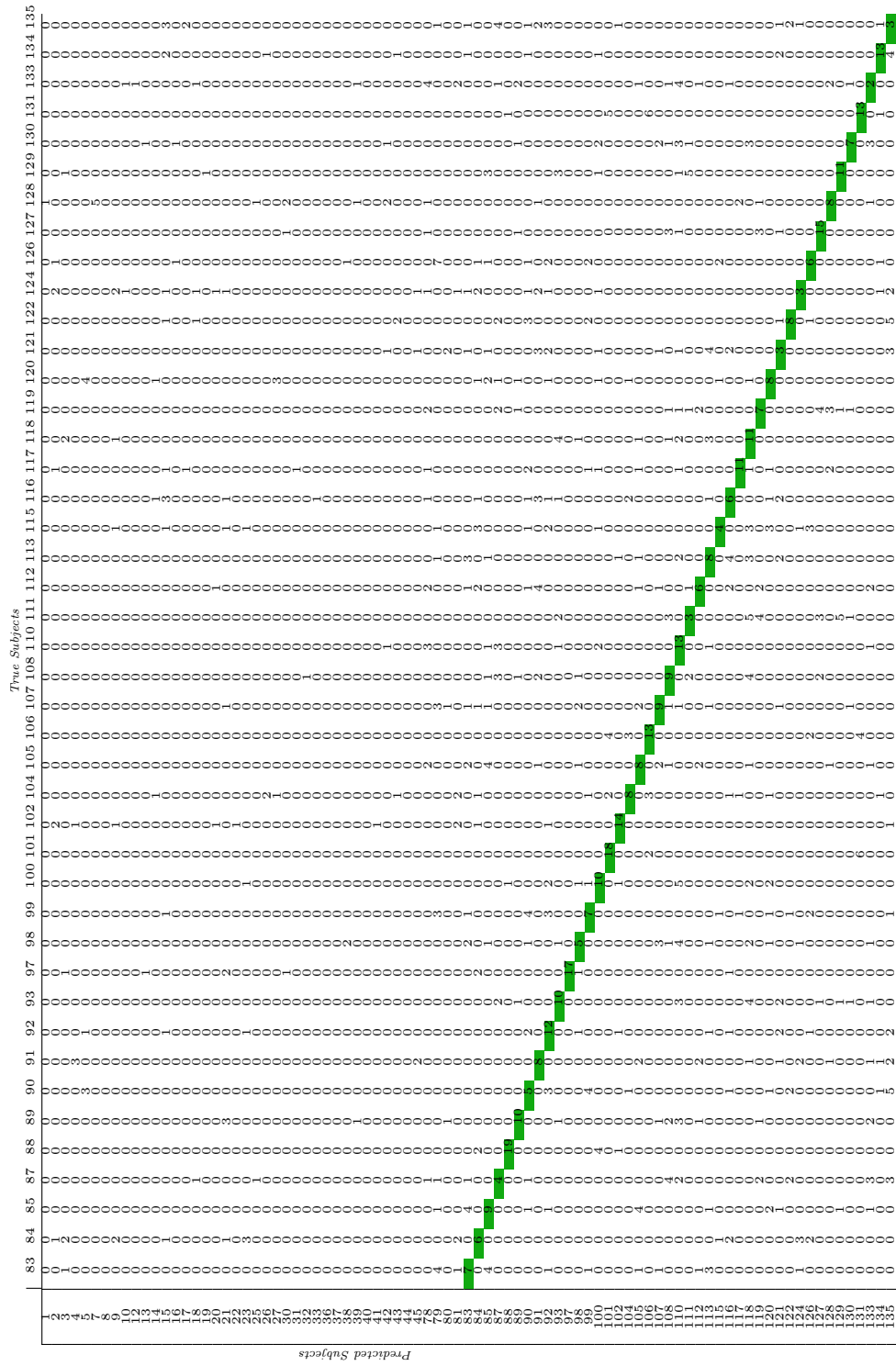


Table 6: Confusion matrix using PCA representing the 95% of initial information

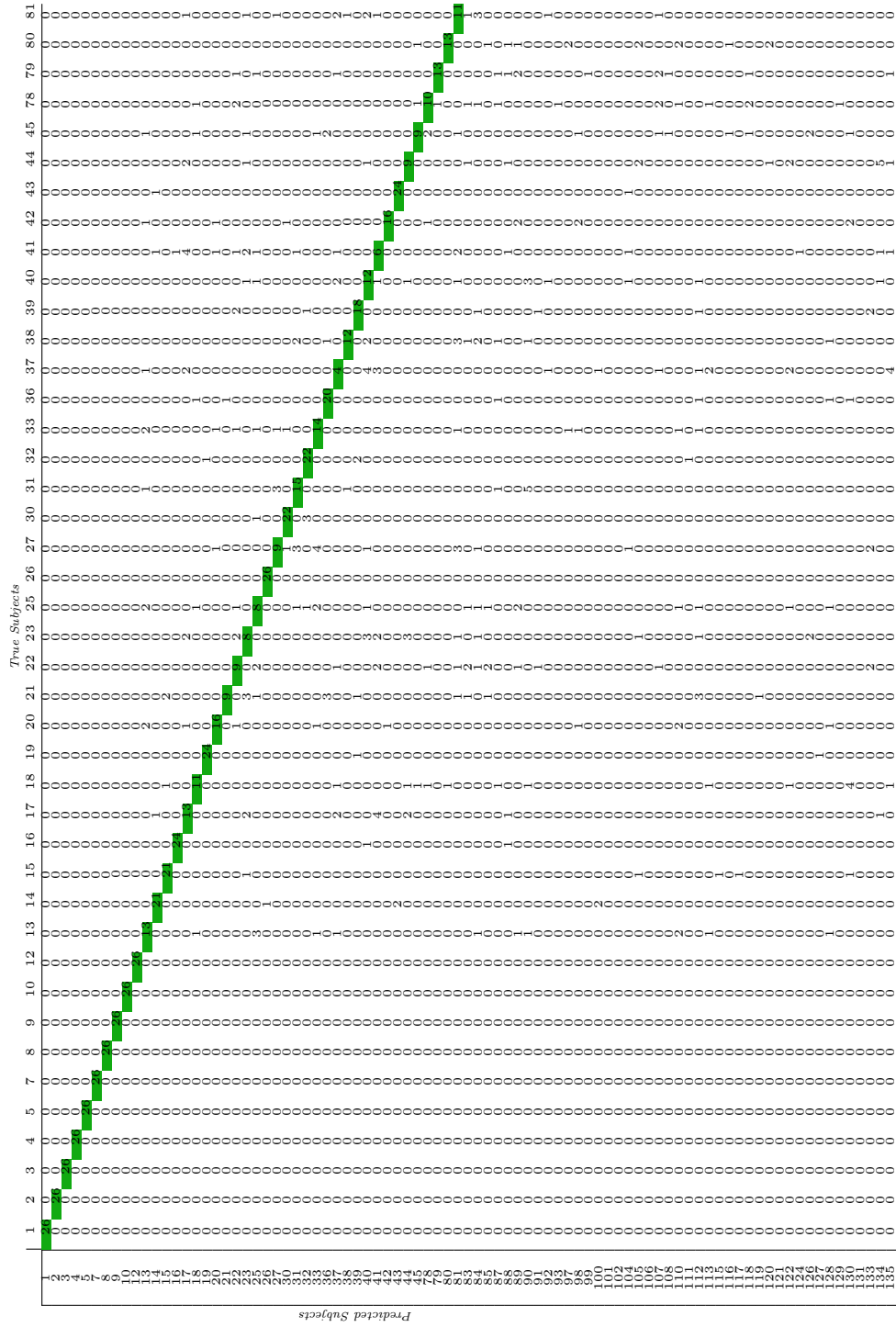


Table 7: Confusion matrix using LDA

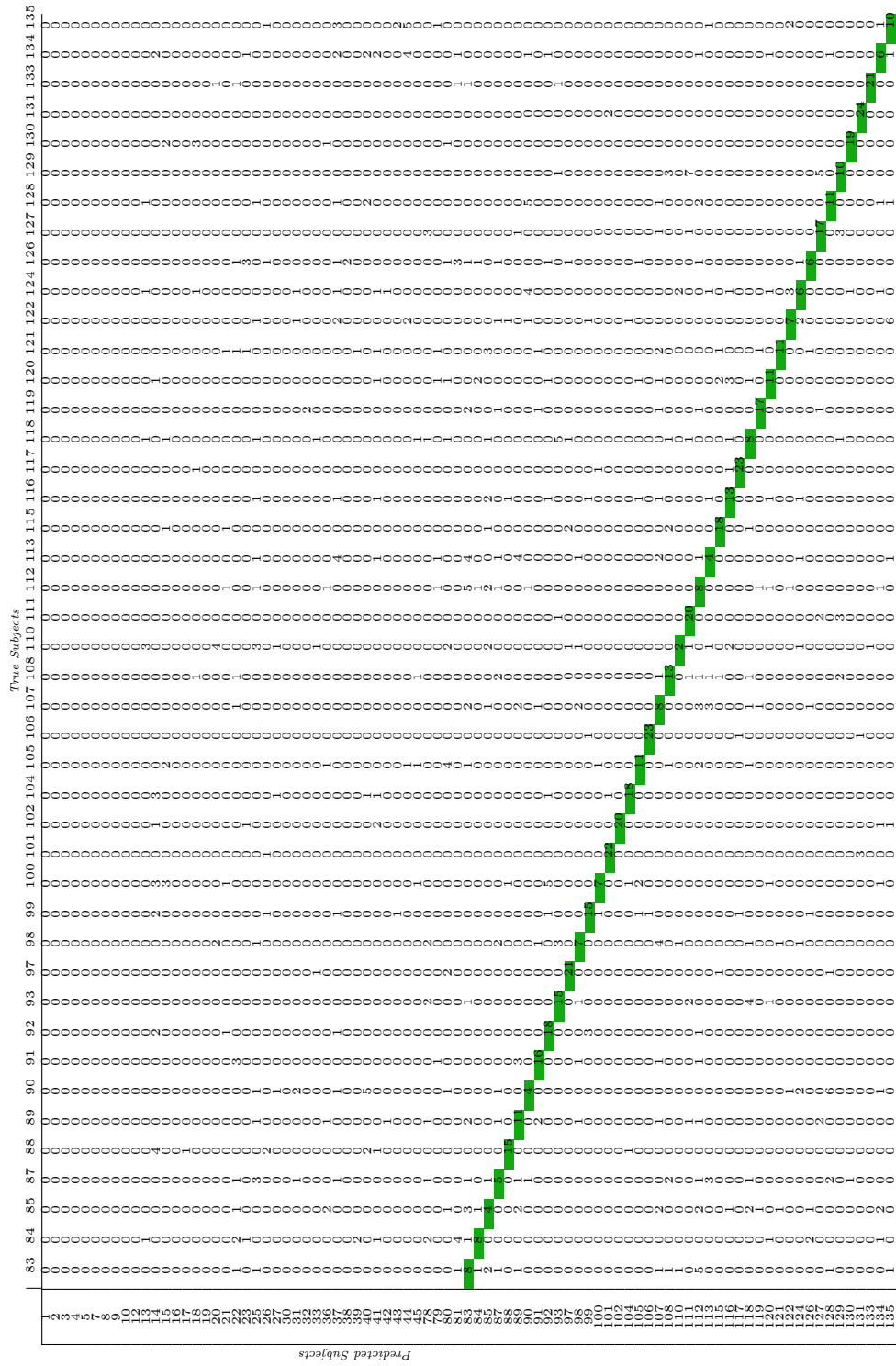


Table 8: Confusion matrix using LDA

