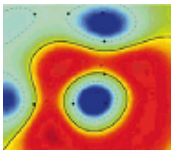
Technische Universität Berlin



Problem Set 1: PCA, LLE, Outlier Detection

Report Machine Learning Lab Course
Fachgebiet Maschinelles Lernen
Prof. Dr. Klaus-Robert Müller
Fakultät IV Elektrotechnik und Informatik
Technische Universität Berlin

submitted by

Budi Yanto

Instructor: Daniel Bartz
Felix Brockherde

Matrikelnummer: 308819
Email: budiyanto@mailbox.tu-berlin.de

Contents

List of Figures

List of Tables

Chapter 1

Implementation

This chapter explains the implementation of some algorithms that are used in this problem set. It includes: *PCA*, γ-*index* and *LLE*.

1.1 Assignment 1: PCA

In this assignment the function *pca* has to be implemented, which receives a *d x n* matrix *X* and the number of components *m* as parameters, and returns the principal components as well as the projected data points in a *m x n* matrix *Z*. The principle components should be returned as a *d x d* matrix *U* and a *1 x d* vector *D*. The vector *D* contains the principal values, sorted in descending order ($D_1 \geq D_2 ...$), whereas the matrix *U* contains the principal directions, which corresponds to the sorted principle values.

The implemented function was tested on the test data and passed the test. Following steps are performed in the implementation of the function:

1. Substract *X* from its mean.
2. Calculate the covariance matrix from the zero-mean *X*.
3. Calculate the eigenvalues and eigenvectors from the covariance matrix.
4. Sort the eigenvalues and eigenvectors in descending order.
5. Form the feature vectors by taking only the first *m* eigenvectors.
6. Project the zero-mean *X* to the feature vectors.
7. Return the projected data points, principal directions and principal values as *Z*, *U* and *D*.

1.2 Assignment 2: γ-Index

The task in this assignment is to implement the γ-index which can be used to detect outliers in data set. In their paper, Harmeling et al. (2006) formulate the formula to calculate the γ-index for each data point as follows:

$$\gamma(x) = \frac{1}{k} \sum_{j=1}^{k} \|x - z_j(x)\| \tag{1.1}$$

where x is a data point, k is the number of nearest neighbours, and $z_1(x), ..., z_k(x)$ are the k nearest neighbours of x.

The implemented function receives a *d x n* matrix *X* containing the data points and a scalar *k* representing the number of neighbours as parameters. It returns the γ-index for each data point in a *1 x n* vector *y*.

The function was tested on the test data and passed the test. Following steps are performed in the implementation of the function:

1. Implement a helper function *distmat* that calculates the distances from the data points to each other and return the distances as a matrix.
2. Get the distance matrix using the function *distmat* mentioned above.
3. Sort the distance matrix in ascending order.
4. Take only the *k*-nearest data points as neighbours for each data point.
5. Calculate the mean from the distances of the *k*-nearest neighbours and set it as the γ-index.
6. Return the calculated γ-index as a *1 x d* vector *y*.

1.3 Assignment 3: LLE

The last task in the implementation part is to implement the *locally linear embedding* method as described by Saul & Roweis (2000) in their paper. The implemented *lle* function returns a *m x n* matrix *Y* representing the resulting embedding and takes following parameters as inputs:

- A *d x n* matrix *X* containing the data points.
- A scalar *m* representing the dimension of the resulting embedding.
- A string *n_rule* determining the method (*'knn'* or *'eps-ball'*) for building the neighbourhood graph.

- A scalar *param* used as parameter for the *n_rule* (k or ϵ, respectively).
- A scalar *tol* determining the size of the regularization parameter.

The implementation is based on the pseudocode described by Saul & Roweis (2000) on their website[1], which contains of three main parts:

1. Find the nearest neighbours of each data point based on *n_rule*.
2. Solve for reconstruction weights W.
3. Compute embedding coordinates Y using weights W.

[1] http://www.cs.nyu.edu/ roweis/lle/algorithm.html

Chapter 2

Application

In this chapter, we are trying to apply the algorithms that are described and implemented in 1 to various datasets: *usps*, *banana*, *fishbowl*, *swissroll* and *flatroll*.

2.1 Assignment 4: PCA

This assignment asks us to apply *PCA* to the *usps* data set and visualizing the results. The *usps* data set consists of 2007 images with the dimension of *16 x 16*. The images are hand-written digits of zero to nine, which can be viewed as classes. Firstly, i separate the data set according to each digit into ten classes and then applied *PCA* to each class. The *PCA* was applied to the original data set and noisy data set.

2.1.1 Original Data Set

PCA was applied to each class and then the results were visualized. As depicted in Figure 2.1 and 2.2, the following were visualized:

- All principle values (as bar plot).
- The largest 25 principal values (as bar plot).
- The first 5 principal directions (as images).

From the visualization, it can be obtained that the information of the data set is only represented by a small subset of principal components, about five to ten components.

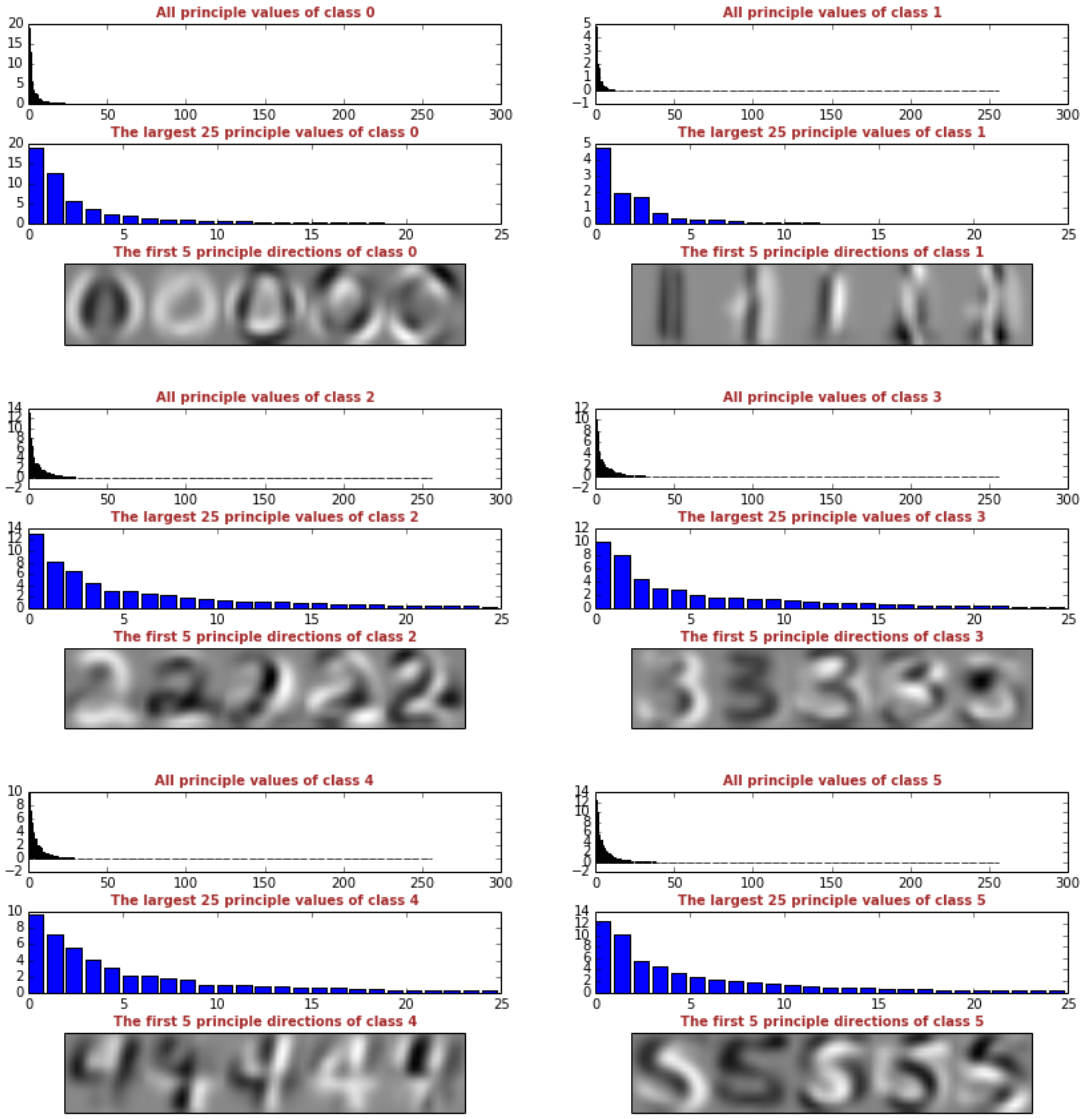


Figure 2.1: Principal components of usps original data set for class 0 - 5

2.1.2 Noisy Data Set

To make it more challenging, some noised are added to the original data set. There are three noise scenarios: *low gaussian noise*, *high gaussian noise* and *outliers*.

For the *low gaussian noise*, standard deviation used to generate the noise is 0.25, whereas for *high gaussian noise* and *outliers*, standard deviations of 0.55 and 1.25 are used, respectively. The same steps as in Subsection 2.1.1 are performed for all noisy data sets. The visualization is depicted in Figure 2.3 to 2.8.

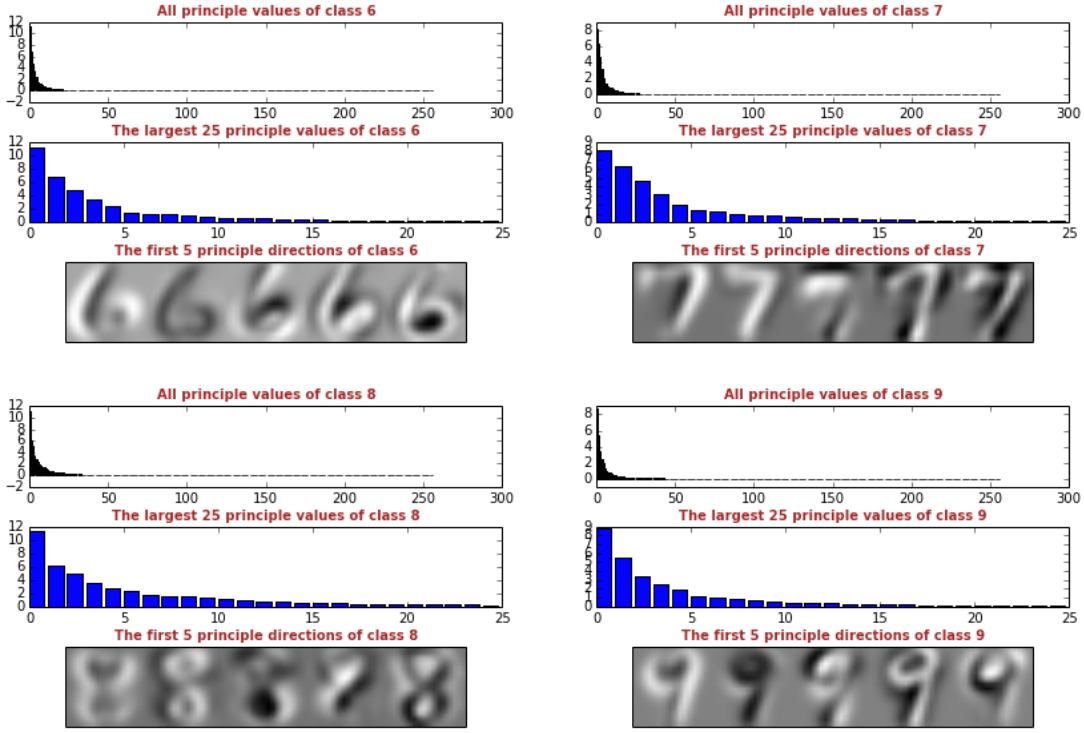


Figure 2.2: Principal components of usps original data set for class 6 - 9

It can be obtained that the information of the data set is now represented by a bigger number of principal components. The noisier a data set is, the bigger the number of principal components is needed to represent the information of the data set.

Figure 2.9 shows ten images of the original data, noisy data, and their reconstruction (denoised) data using *PCA*. The number of components m used for *moderately* noisy images is 11, for *very* noisy images is 75 and for *extremely* noisy images is 150.

2.2 Assignment 5: Outlier Detection Using γ-Index

2.3 Assignment 6: LLE

2.4 Assignment 7: LLE With Noise

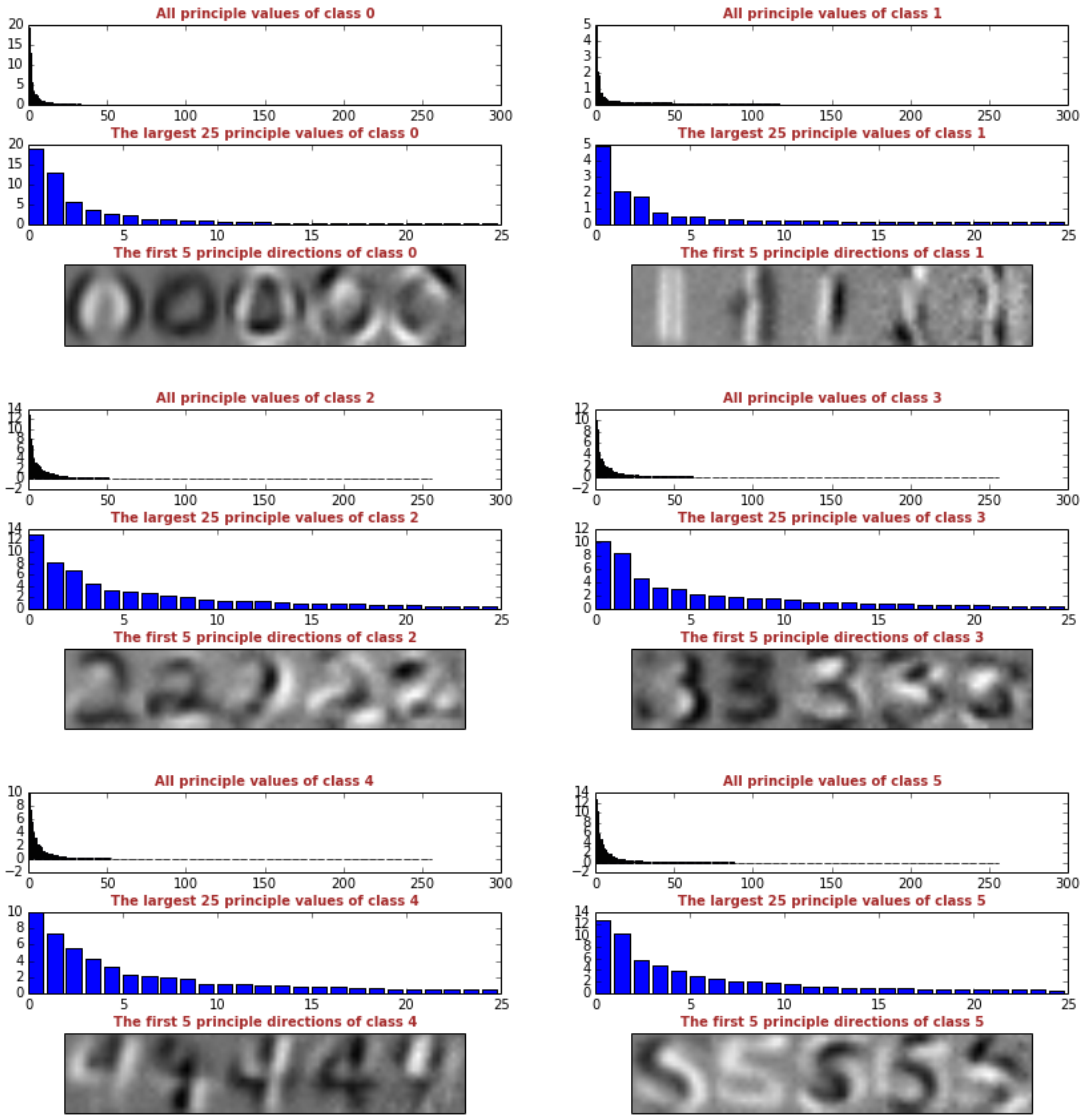


Figure 2.3: Principal components of moderately noisy data set for class 0 - 5

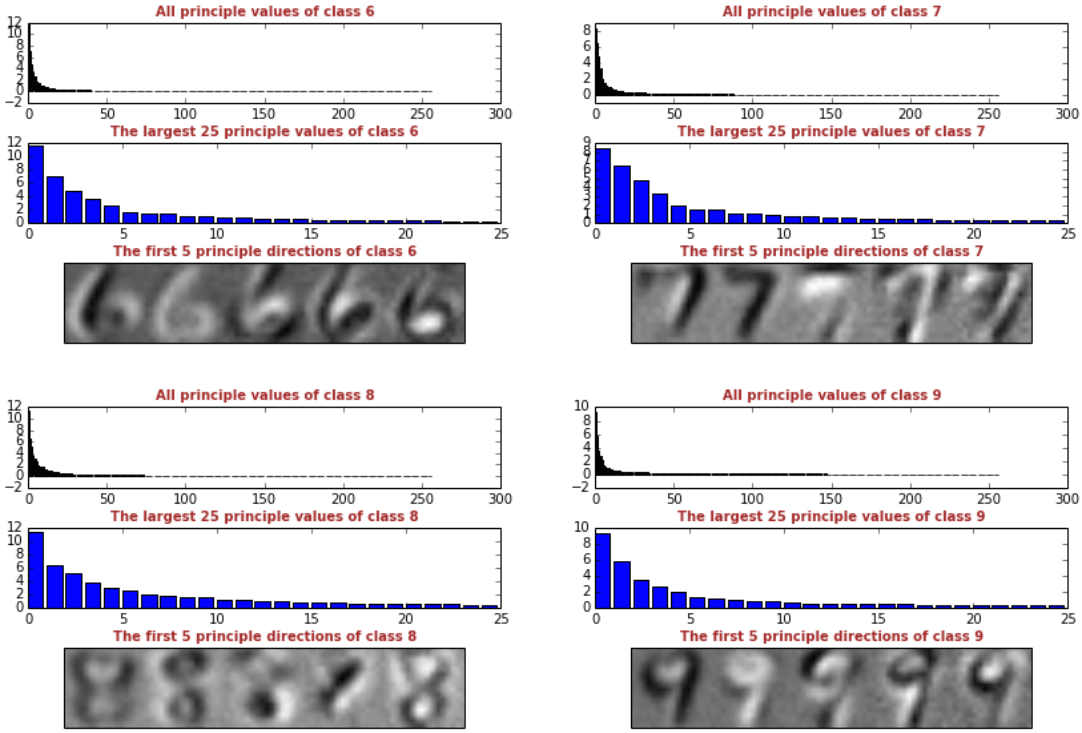


Figure 2.4: Principal components of moderately noisy data set for class 6 - 9

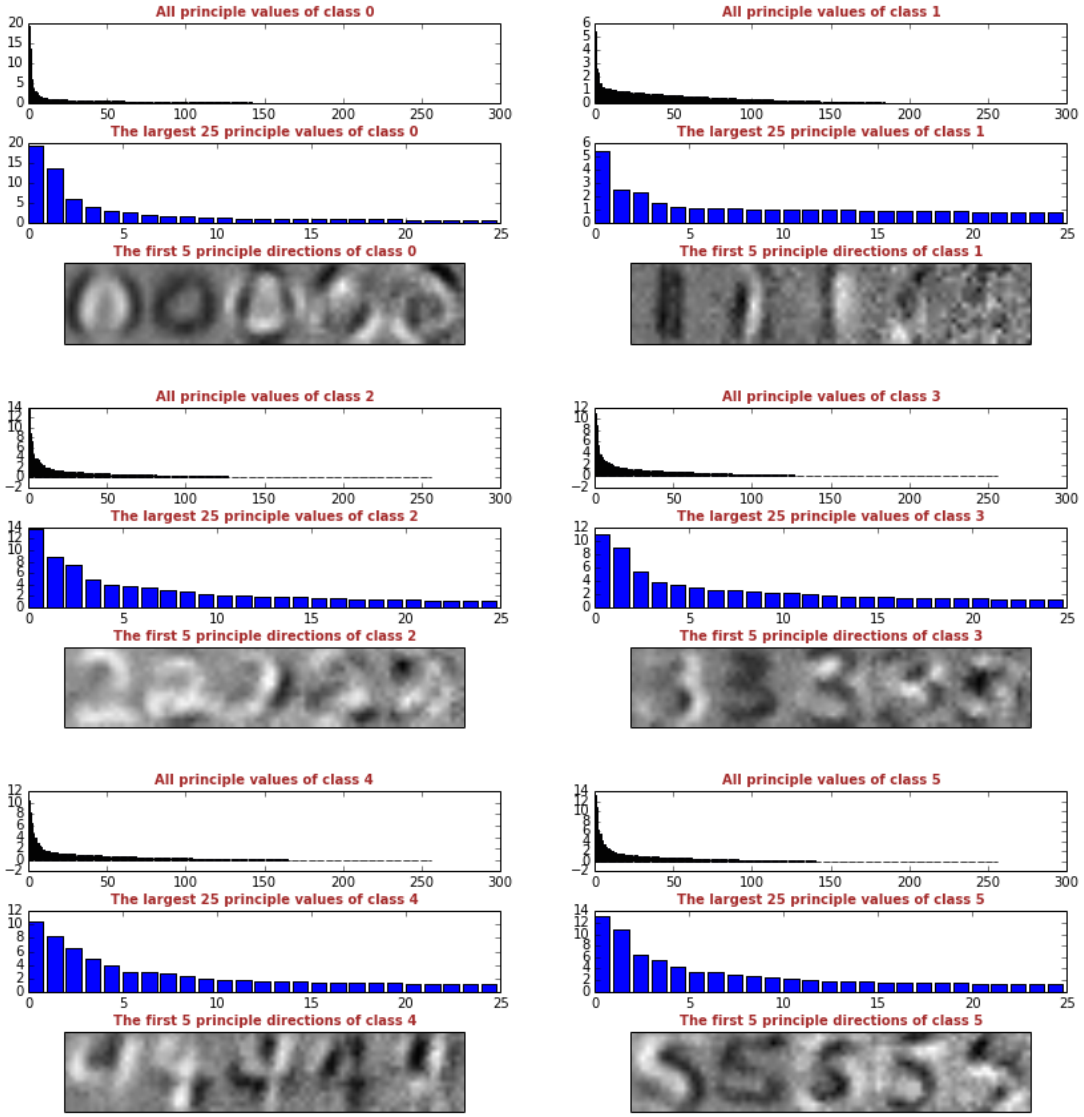


Figure 2.5: Principal components of very noisy data set for class 0 - 5

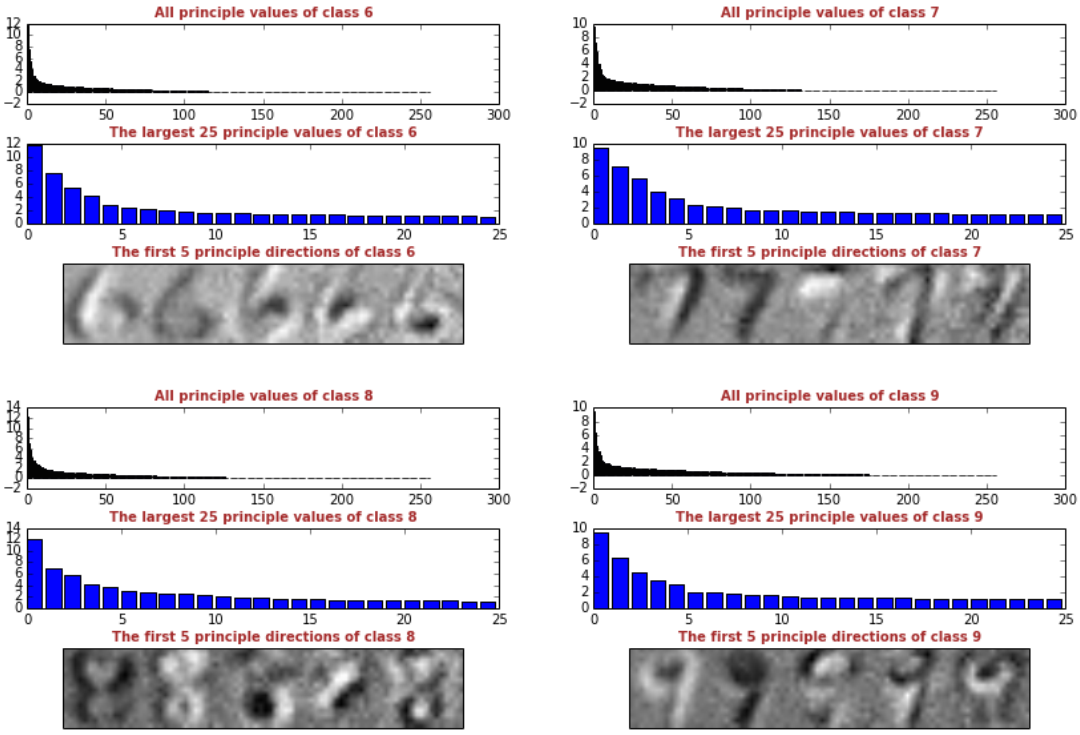


Figure 2.6: Principal components of very noisy data set for class 6 - 9

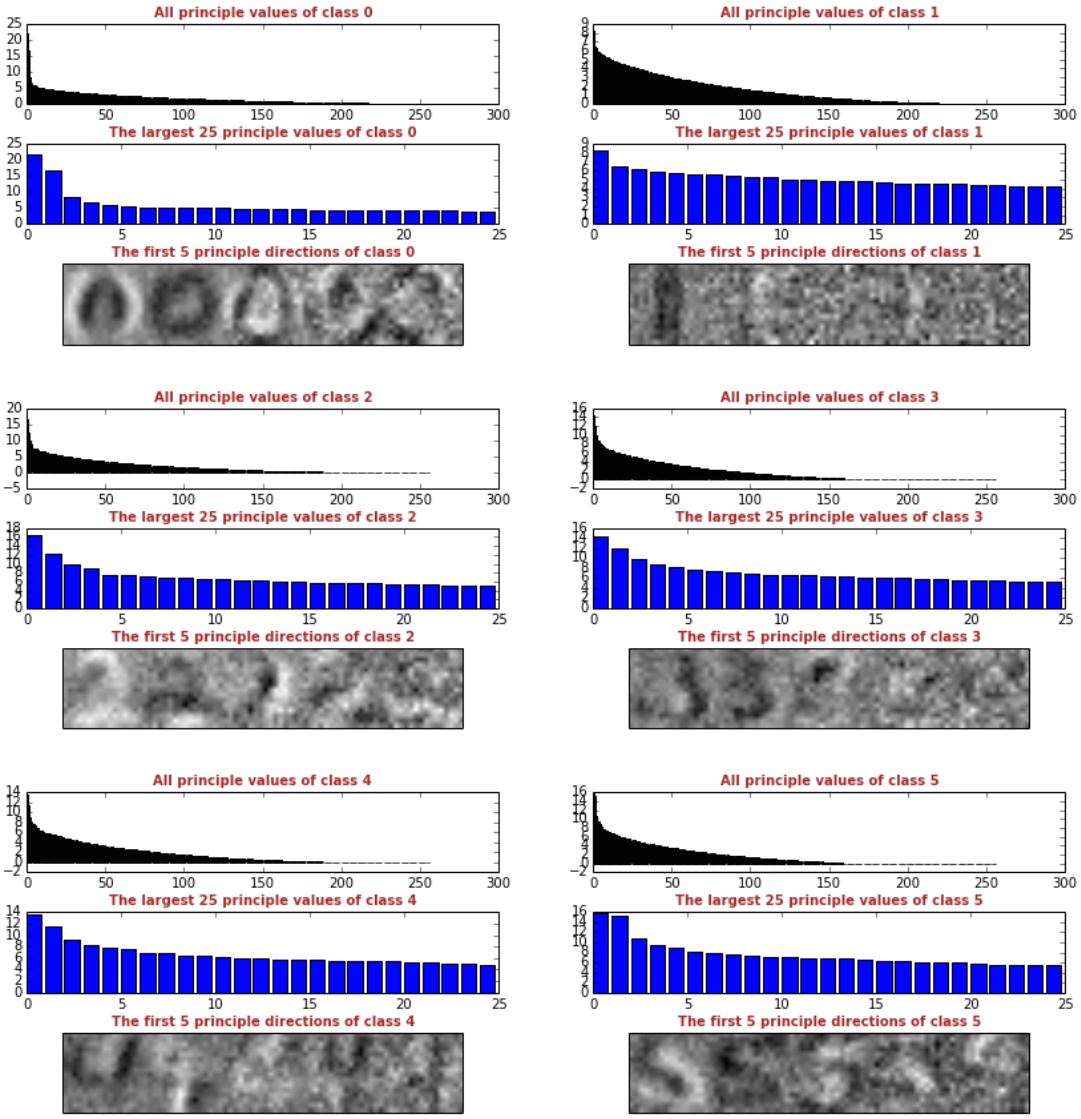


Figure 2.7: Principal components of extremely noisy data set for class 0 - 5

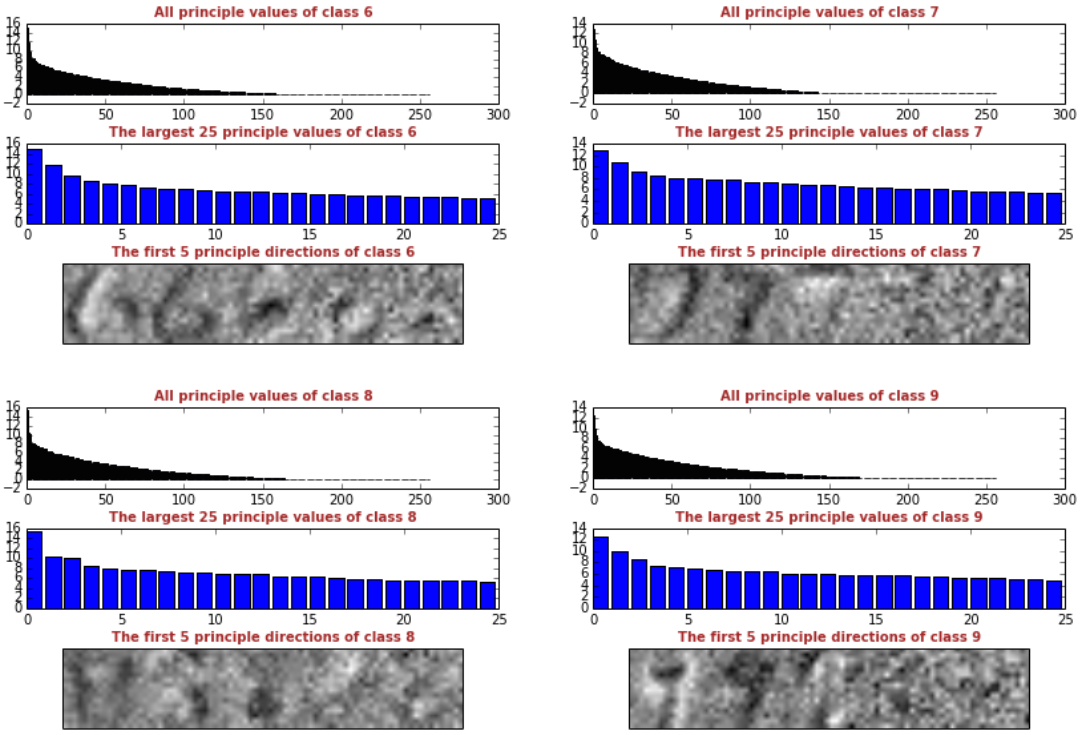


Figure 2.8: Principal components of extremely noisy data set for class 6 - 9

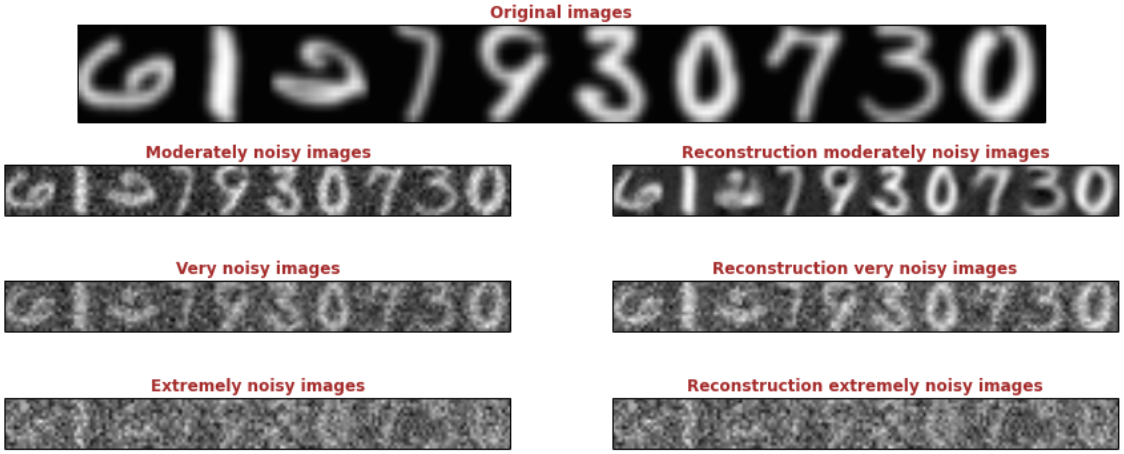


Figure 2.9: Examples of some original, noisy and reconstruction images

Chapter 3

Backend Integration

3.1 Problems

Chapter 4

Frontend Integration

Chapter 5

Agent Framework Integration

5.1 Problems

Yet to be implemented

5.2 Solutions

Yet to be implemented

5.3 Summary

Chapter 6

Evaluation

DELETEME: The evaluation chapter is one of the most important chapters of your work. Here, you will prove usability/efficiency of your approach by presenting and interpreting your results. You should discuss your results and interprete them, if possible. Drawing conclusions on the results will be one important point that your estimators will refer to when grading your work.

Chapter 7

Conclusion and Future Work

7.1 Summary

Summarize from Chapter 2 to Chapter 6

7.2 Conclusion

7.2.1 Good points

7.2.2 Drawbacks

Bibliography

Harmeling, S., Dornhege, G., Tax, D., Meinecke, F. & Müller, K.-R. (2006), 'From outliers to prototypes: Ordering data', *Neurocomputing* **69**(13-15), 1608–1618.

Saul, L. K. & Roweis, S. T. (2000), An introduction to locally linear embedding, Technical report.

Appendices

DELETEME: everything that does not fit into your work, e.g. a 5 page table that breaks the reading flow, should be placed here

Appendix A: Abbreviations

AES	Advanced Encryption Standard (Symmetrisches Verschlüsselungsverfahren)
ASCII	American Standard Code for Information Interchange (Computer-Textstandard)
dpi	dots per intch (Punkte pro Zoll; Maß für Auflösung von Bilddateien)
HTML	Hypertext Markup Language (Textbasierte Webbeschreibungssprache)
JAP	Java Anon Proxy
JPEG	Joint Photographic Experts Group (Grafikformat)
JPG	Joint Photographic Experts Group (Grafikformat; Kurzform)
LED	Light Emitting Diode (lichtemittierende Diode)
LSB	Least Significant Bit
MD5	Message Digest (Kryptographisches Fingerabdruckverfahren)
MPEG	Moving Picture Experts Group (Video- einschließlich Audiokompression)
MP3	MPEG-1 Audio Layer 3 (Audiokompressionformat)
PACS	Picture Archiving and Communication Systems
PNG	Portable Network Graphics (Grafikformat)
RSA	Rivest, Shamir, Adleman (asymmetrisches Verschlüsselungsverfahren)
SHA1	Security Hash Algorithm (Kryptographisches Fingerabdruckverfahren)
WAV	Waveform Audio Format (Audiokompressionsformat von Microsoft)

Appendix B: LaTeX Help

How to Use This Template

- Remove all of my text which is mostly labeled with DELETEME
- Change the information in the 00a_title_page.tex file
- Use the information written in this section
- Ask you supervizor to help you
- If I am not your supervizor and noone else can help you, write me an email (aubrey.schmidt@dai-labor.de)

Citations

Citing is one of the essential points you need to do in you thesis. Statements not basing on results of your own research[1] not being cited represent a breach on the rules of scientific working. Therefore, you every statement needs to be cited basing on information that other people can cross-check. A common way of citing in technical papers is:

- Oberheide et al. (**?**) state that the average time for an anti-virus enginge to be updated with a signature to detect an unknown threat is 48 days.

Note: et al. is used when the paper was written by more than two people. Check the code of this section to learn how to cite from a technical perspective.

Note: you can change the citation style in the `thesis.tex` file, e.g. to harvard style citations. Instructions on this can also be found in this file.

You should not cite anything that can be changed, e.g. it is not that good citing web pages since they might get updated changing the cited content. There are no clear quality measures on citing sources but aubrey believes that the following list is true for several cases, starting with highest quality:

1. Journal article or book
2. Conference paper
3. Workshop paper
4. Technical report
5. Master thesis

[1] in what ever context

6. Bachelor thesis

7. General Web reference

There might be workshop papers that have a higher quality than some journal papers. Therefore this list only gives you a hint on possible quality measures. Another measure can be whether a paper was indexed by ACM/IEEE, although this is not a strong indicator.

Finding and Handling Citation Sources

Following ressources are required for finding and handling articles, books, papers and sources.

- your primary resource will be `http://scholar.google.com`
- `http://www.google.com` might also be used
- `wikipedia.com` can be a good start for finding relevant papers on your topic
- you should download and install JabRef or a similar tool `http://jabref.sourceforge.net/`
- you should point JabRef to the mybib.bib file
- you should immediately enter a relevant paper to JabRef, additionally, you should write a short summary on it; else, you will do this work at least twice.

General Advices

- Do not take care of design, LaTeXwill do this for you. If you still feel that you need to take care of this, do this when you have finished writing, else you will end up in a lot of double and triple work.
- LaTeXwill do exactly that you will tell it to do. If you have problems with this, go for google or ask you supervizor
- use labels in order to be able to reference to chapters, section, subsections, figures, tables, etc. ...

General Commands

- check `http://en.wikibooks.org/wiki/LaTeX`
- check `http://www.uni-giessen.de/hrz/tex/cookbook/cookbook.html` German

Please also check the following source (**?**).

Code

This section shows you how to get your code into a LaTeXdocument. See code for options.

```
1 class Beispiel{
2
3   public static void main(String args[]){
4
5     System.out.println("Hello World");
6
7   }
8
9 }
```

```
1 class Beispiel{
2
3   public static void main(String args[]){
4
5     System.out.println("Hello World");
6
7   }
8
9 }
```

Listing 7.1: Example code is presented here

Figures

This section describes how to include images to your document. Information was taken from `http://en.wikibooks.org/wiki/LaTeX/Floats,_Figures_and_Captions`, visited on 05/08/2011. Please make sure to use original vector graphics as basis since image quality might be used as weak indicator for thesis quality. For this, try to find find files in `.SVG` or `.PDF` format. Exporting a `.PNG` or `.JPG` to `.PDF` will not work since data was already lost while exporting it to these formats. This is the case for most Web graphics. Wikipedia startet entering most in images in `.SVG` which easily can be transformed to `.PDF`, but please do not forget proper citations.

Figure 7.1: Including an image; in this case a PDF. Please note that the caption is placed below the image.

Figure 7.2: See code for caption options: this is a long caption which is printed in the Text. Additionally, image size was increased

(a) Small (b) Large (c) Medium

Figure 7.3: Placing images side by side using the subfig package. Space between the images can be adjusted.

Tables

Here, you will find some example tables.The tables were taken from `http://en.wikibooks.org/wiki/LaTeX/Tables`, visited on 05/08/2011. Table environment was added plus caption and label. For code, check `__help/latex_hinweise.tex`.

Table 7.1: Simple table using vertical lines. Note that the caption is always above the table! Please check code for finding the right place for the table label.

1	2	3
4	5	6
7	8	9

Table 7.2: Table using vertical and horizontal lines

7C0	hexadecimal
3700	octal
11111000000	binary
1984	decimal

Table 7.3: Table with column width specification on last column

Day	Min Temp	Max Temp	Summary
Monday	11C	22C	A clear day with lots of sunshine. However, the strong breeze will bring down the temperatures.
Tuesday	9C	19C	Cloudy with rain, across many northern regions. Clear spells across most of Scotland and Northern Ireland, but rain reaching the far northwest.

Table 7.4: Table using multi-column and multirow

Team sheet		
Goalkeeper	GK	Paul Robinson
Defenders	LB	Lucus Radebe
	DC	Michael Duberry
	DC	Dominic Matteo
	RB	Didier Domi
Midfielders	MC	David Batty
	MC	Eirik Bakke
	MC	Jody Morris
Forward	FW	Jamie McMaster
Strikers	ST	Alan Smith
	ST	Mark Viduka