AMS 550.400 Homework 3 Additional Instruction Due Date: Nov 19

Last Compiled on November 27, 2012

General Instruction

- failure to follow the instruction will result in severe penalty (graded at 90% or even worse)
  - no second grading is planned for this homework set
  - type up your own homework (i.e., no copy-and-pasting from others & you know one can easily check this)
- from the home directory `~/`, make a directory for this homework set
  - use `mkdir nhleeHW3.git` for the directory name
  - create a Sweave file called `nhleeHW3.Rnw`
  - replace `nhlee` with the "left-hand side" of your school email
- initialize it as a git folder
  - do this from inside of the git folder for your own sake
  - make sure to verify that your folder contains a hidden folder called `.git`
  - set it up from RStudio as a RStudio project with git support
  - look for `[COMMIT]` from the text below for the location where you are supposed to add & commit
- using RStudio for editing and compiling your Rnw file is highly recommended
  - to compile, find and press "Complie PDF" button within the RStudio editor window (typically, the upper left corner window)
  - alternatively, you can use `R CMD <Sweave/pdflatex>` from bash-shell command line provided that you are in the "appropriate" directory

```
R CMD Sweave yourfilename.Rnw
R CMD pdflatex yourfilename.tex
```

- use the `homeworkset3.tex` file as a starting point for typesetting your homework solution
  - find it from the course git folder
  - do not delete the problem statements
  - do not delete/modify the existing codes in the preamble area
- once completed, compress the folder as a `.nhleeHW3.zip` or `nhleeHW3.tar.gz`, where `nhlee` is replaced with one from your school email
  - make sure that your compressed file can actually be decompressed

- on the due date,
  - submit a paper copy to the instructor during the class
  - upload the compressed folder to the designated BB discussion forum before midnight
  - any commit made after the class meeting time will be discarded using `git reset --hard`, and will not be counted as a part of your homework submission

Problems from Chapter 7: Matrix Algebra for MDS

Ex 7.18

(a) [COMMIT] Use Sweave to accomplish this

- Use

```
\includegraphics{nhleeHW3-001}
```

```
\includegraphics{nhleeHW3-002}
```

- Make sure to label the horizontal axis, the vertical axis and give the main title, and give different color for A and B e.g., by filling out the space between the quotation marks, and choose a different symbol for A and B by specifying a number for `cex` and `pch`

```
plot(xydataA,xlab='',ylab='',main='',color='',cex=,pch=)
points(xydataB,cex=,pch=,cex=,color='')
```

- [COMMIT] Include your R codes using `lstlisting` making sure that it has an appropriate caption

(b) [COMMIT] Supplement your calculation using R/Sweave

- computations need to be done before using them in the text using or concurrently

```
The rank of A is 2 or equivalently 2.
The reason why the rank of A is not 1 is that:
The eigenvalues of the matrix is ... and ..., which means the number of the eigenvalues is 2.

As the number of eigenvalues and the rank equals, the rank of A is 2 rather than 1.

The rank of B is 1 or equivalently 1.
```

(c) [COMMIT] directly use the code and output from R/Sweave, but make sure to explain your answers The vector used to explain how we can get the 3rd column from the first two columns is as as follows: For A, the vector is 1

```
\begin{Schunk}
\begin{Sinput}
> #b <- A %*% x
> #solve(A,b)
\end{Sinput}
\end{Schunk}
```

Ex 7.24

- [COMMIT] use `lstlisting` to list your R code
- [COMMIT] use R/Sweave for computation, but do not use the built-in `dist` function

```
\begin{Schunk}
\begin{Sinput}
> yA=diag(diag(c(1,1,1,1)));
> dA=diag(A %*% t(A));
> DA2=dA %*% t(yA) + yA %*% t(dA) - 2* A %*% t(A);
> print(DA2)
\end{Sinput}
\begin{Soutput}
     [,1] [,2] [,3] [,4]
[1,]    0    3   12   27
[2,]    3    0    3   12
[3,]   12    3    0    3
[4,]   27   12    3    0
\end{Soutput}
\end{Schunk}
```

```
\begin{Schunk}
\begin{Sinput}
> yB=diag(diag(c(1,1,1,1)));
> dB=diag(B %*% t(B));
> DB2=dB %*% t(yB) + yB %*% t(dB) - 2* B %*% t(B);
> print(DB2)
\end{Sinput}
\begin{Soutput}
     [,1] [,2] [,3] [,4]
[1,]    0   14   56  126
[2,]   14    0   14   56
[3,]   56   14    0   14
[4,]  126   56   14    0
\end{Soutput}
\end{Schunk}
```

- [COMMIT] use R/Sweave for computation, and this time, do use the built-in `dist` function for comparison

```
\begin{Schunk}
\begin{Sinput}
> #Your R codes go here
\end{Sinput}
\end{Schunk}
```

- [COMMIT] make sure to explain your computation, e.g., compare the two computations

Ex 7.30 Omit (c), (d) and (e). The necessary data is saved in `matlabclown.RData` and can be found from the course git folder. The followings are the equivalent R version:

```
load('matlabclown.RData')
image(X) # omit this in your Sweave code
svdX = svd(X)
U = svdX$u
```

```
S = diag(svdX$d)
V = svdX$v
k = 10
M = U[,1:k,drop=FALSE] %*% S[1:k,1:k,drop=FALSE] %*% t(V[,1:k,drop=FALSE])
image(M) # omit this in your Sweave code
image(M,col=gray.colors(k))
```

(a) [COMMIT] choose a small, a medium and a large value for k

- for each k,
  - do [COMMIT]
  - your performance evaluation is to be included as a caption, and change `tinyK`, `width` and `height` accordingly

(b)
- [COMMIT] code up all your computation using R/Sweave before starting to type your explanation

---

- [COMMIT] write your explanation referring to the numbers computed in the previous step, using

Problems from Chapter 4: Multidimensional Scaling

Ex 4.1

- [COMMIT] Modify the code in Listing 1 for illustrating the first ten objects on a "line"

Listing 1: TikZ Code for Figure 6

```
\begin{tikzpicture}
  \foreach \x in {1,2,...,5,7,8,...,12}
    \foreach \y in {1,...,5}
    {
      \draw (\x,\y) +(-.5,-.5) rectangle ++(.5,.5);
      \draw (\x,\y) node{\footnotesize $(\x,\y)$};
    }
\end{tikzpicture}
```

- [COMMIT] list your R/Sweave codes using `lstlisting`
- [COMMIT] use the R/Sweave codes to compute
- [COMMIT] explain your computed numerical values
- make sure to refer to your R code listing via `\ref` and to the computed values using `\Sexpr`
- refer to [?] if necessary

```
> tinyK = 1
> #smallK =
> #mediumK =
> #largeK =
> load('matlabclown.RData')
> svdX = svd(X)
> U = svdX$u
> S = diag(svdX$d)
> V = svdX$v
> M = U[,1:tinyK,drop=FALSE] %*% S[1:tinyK,1:tinyK,drop=FALSE] %*% t(V[,1:tinyK,drop=FALSE
> image(M,col=gray.colors(tinyK))
```

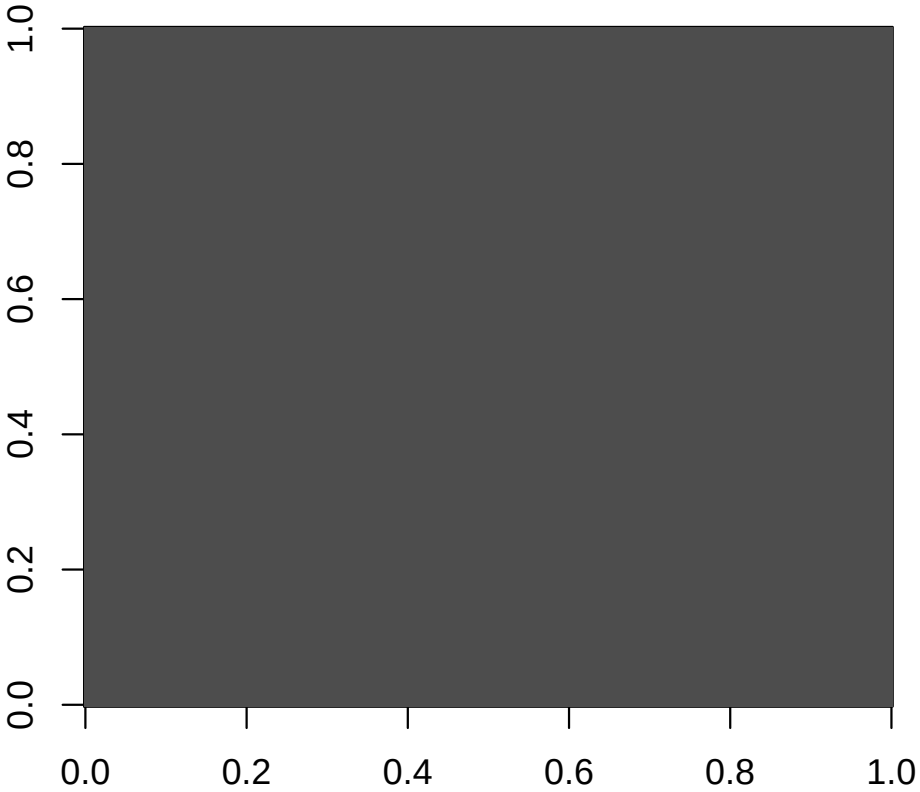

Figure 1: <YOUR PERFORMANCE EVALUATION> on 1

```
> #tinyK = 1
> smallK =10
> #mediumK =
> #largeK =
> M = U[,1:smallK,drop=FALSE] %*% S[1:smallK,1:smallK,drop=FALSE] %*% t(V[,1:smallK,drop=F
> image(M,col=gray.colors(smallK))
```

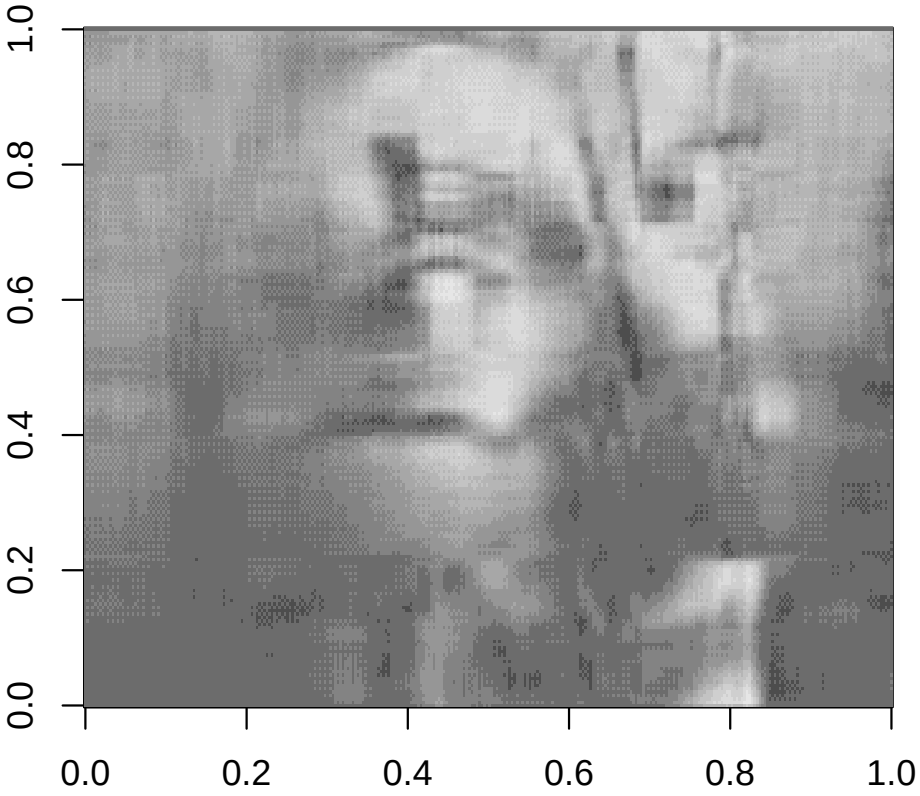

Figure 2: <YOUR PERFORMANCE EVALUATION> on 10

```
> #tinyK = 1
> smallK =10
> #mediumK =
> #largeK =
> M = U[,1:smallK,drop=FALSE] %*% S[1:smallK,1:smallK,drop=FALSE] %*% t(V[,1:smallK,drop=F
> image(M,col=gray.colors(smallK))
```

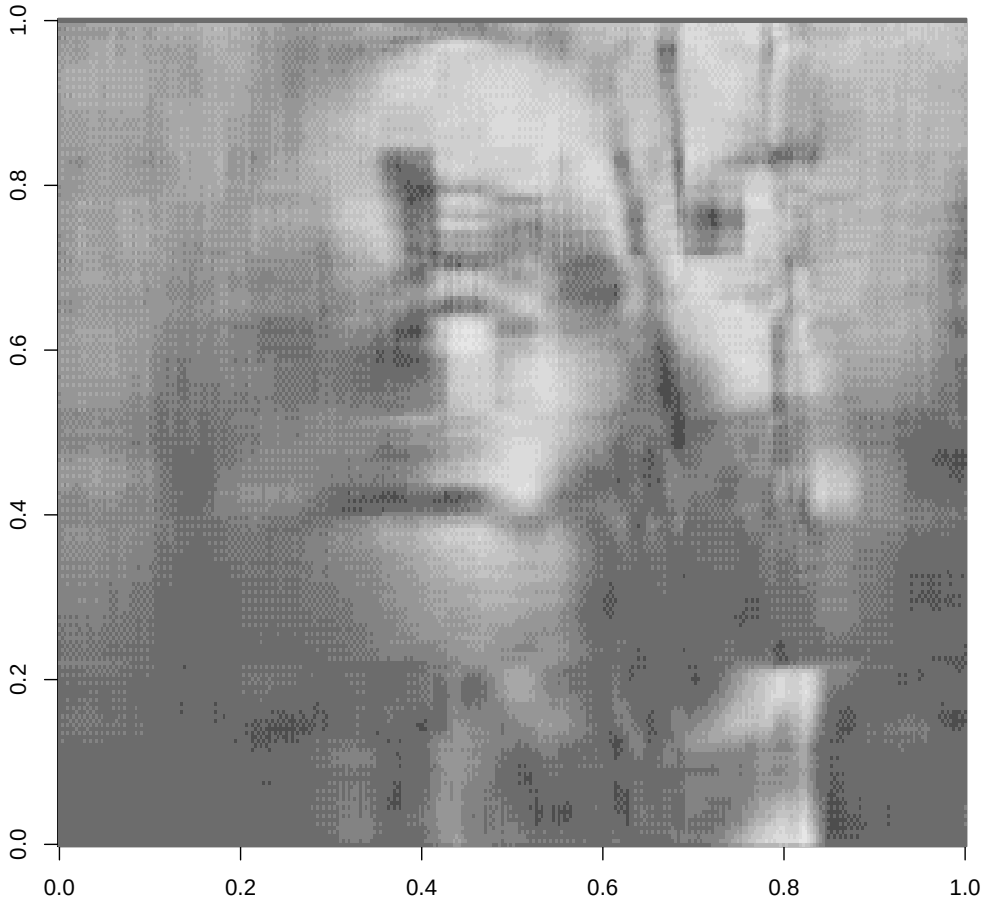

Figure 3: <YOUR PERFORMANCE EVALUATION> on 10

```
> #tinyK = 1
> #smallK = 10
> mediumK = 100
> #largeK =
> M = U[,1:mediumK,drop=FALSE] %*% S[1:mediumK,1:mediumK,drop=FALSE] %*% t(V[,1:mediumK,dr
> image(M,col=gray.colors(mediumK))
```

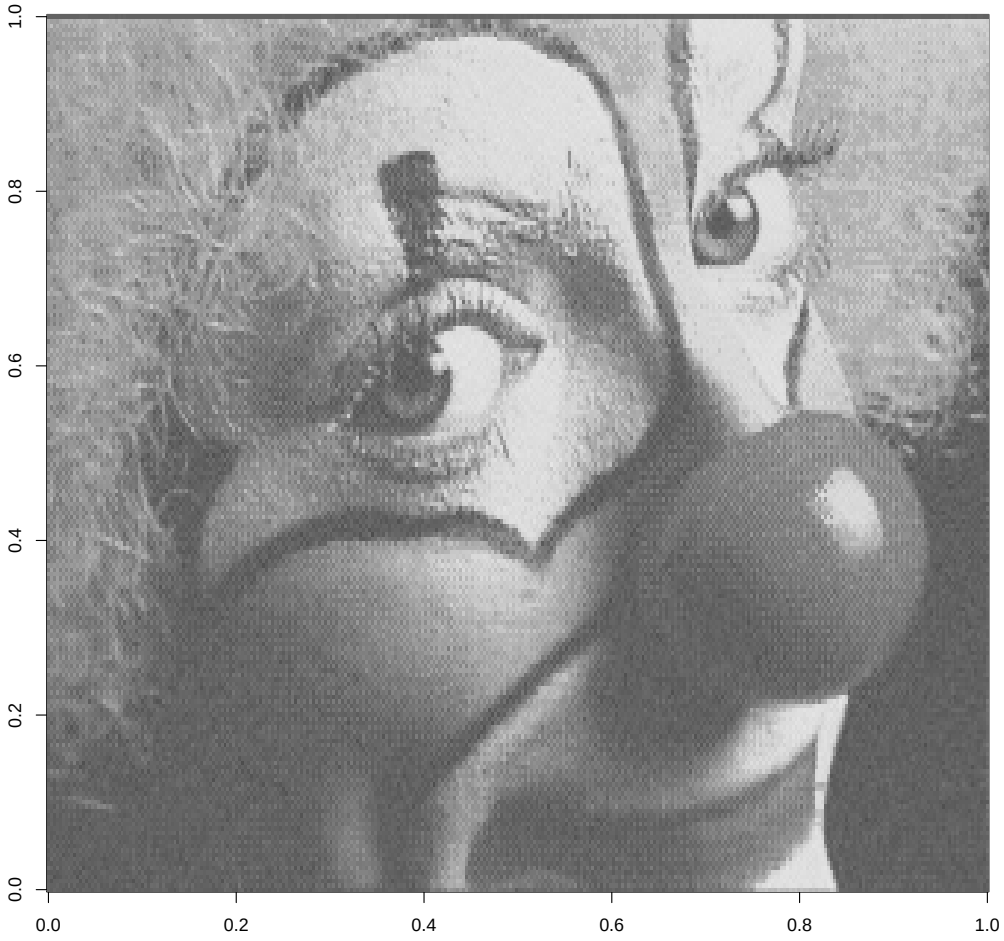

Figure 4: <YOUR PERFORMANCE EVALUATION> on 100

```
> #tinyK = 1
> #smallK =10
> #mediumK = 100
> largeK = 200
> M = U[,1:largeK,drop=FALSE] %*% S[1:largeK,1:largeK,drop=FALSE] %*% t(V[,1:largeK,drop=F
> image(M,col=gray.colors(largeK))
```

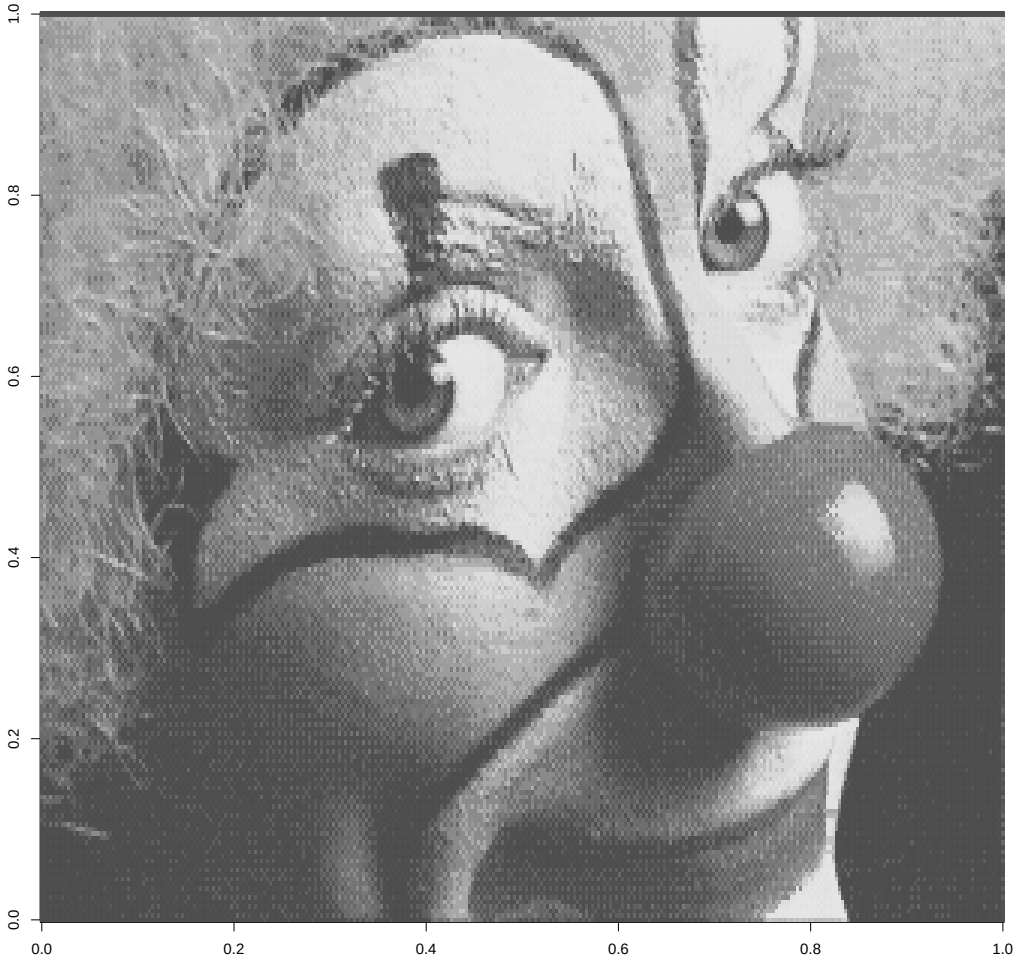

Figure 5: <YOUR PERFORMANCE EVALUATION> on 200

$(1,5)$	$(2,5)$	$(3,5)$	$(4,5)$	$(5,5)$
$(1,4)$	$(2,4)$	$(3,4)$	$(4,4)$	$(5,4)$
$(1,3)$	$(2,3)$	$(3,3)$	$(4,3)$	$(5,3)$
$(1,2)$	$(2,2)$	$(3,2)$	$(4,2)$	$(5,2)$
$(1,1)$	$(2,1)$	$(3,1)$	$(4,1)$	$(5,1)$

$(7,5)$	$(8,5)$	$(9,5)$	$(10,5)$	$(11,5)$	$(12,5)$
$(7,4)$	$(8,4)$	$(9,4)$	$(10,4)$	$(11,4)$	$(12,4)$
$(7,3)$	$(8,3)$	$(9,3)$	$(10,3)$	$(11,3)$	$(12,3)$
$(7,2)$	$(8,2)$	$(9,2)$	$(10,2)$	$(11,2)$	$(12,2)$
$(7,1)$	$(8,1)$	$(9,1)$	$(10,1)$	$(11,1)$	$(12,1)$

Figure 6: An extension of an example from the TikZ & PGF manual [**?**]

Ex 4.3

- `[COMMIT]` list your R code using `lstlisting`
- `[COMMIT]` load the data (`require(MVA);data(gardenflowers)`) and compute using R/Sweave
- `[COMMIT]` include a plot of (relative) positions using R/Sweave
- `[COMMIT]` allocate at least a quarter page of *text* explaining the result

Ex 4.3 solution

- The R code is as follows:

```
\includegraphics{nhleeHW3-014}
```

(c) `[COMMIT]` directly use the code and output from R/Sweave, but make sure to explain your answers The vector used to explain how we can get the 3rd column from the first two columns is as as follows: For A, the vector is 1

References

[1] V. Bos and S. Mauw. *A LaTeX macro package for Message Sequence Charts—Maintenance document—Describing version* , June 2002. Included in MSC macro package distribution.

[2] V. Bos and S. Mauw. *A LaTeX macro package for Message Sequence Charts—Reference Manual—Describing version* , June 2002. Included in MSC macro package distribution.

[3] V. Bos and S. Mauw. *A LaTeX macro package for Message Sequence Charts—User Manual—Describing version* , June 2002. Included in MSC macro package distribution.

[4] Michel Goossens, Sebastian Rahtz, and Frank Mittelbach. *The LaTeX Graphics Companion*. Addison-Wesley, 1997.

[5] ITU-TS. ITU-TS Recommendation Z.120: Message Sequence Chart (MSC). Geneva, 1997.

[6] L. Lamport. *LaTeX—A Document Preparation System—User's Guide and Reference Manual*. Adsison-Wesley, 2nd edition, 1994. Updated for LaTeX 2_ε.

[7] E. Rudolph, P.Graubmann, and J. Grabowski. Tutorial on message sequence charts (MSC'96). In *FORTE*, 1996.