ETH
Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

Lecture with Computer Exercises:

Modelling and Simulating Social Systems with MATLAB

Project Report

Innovation Diffusion

Tobias Oberholzer & Janine Süess

Zürich

December 2010

Eigenständigkeitserklärung

Hiermit erkläre ich, dass ich diese Gruppenarbeit selbständig verfasst habe, keine anderen als die angegebenen Quellen-Hilfsmittel verwenden habe, und alle Stellen, die wörtlich oder sinngemäss aus veröffentlichen Schriften entnommen wurden, als solche kenntlich gemacht habe. Darüber hinaus erkläre ich, dass diese Gruppenarbeit nicht, auch nicht auszugsweise, bereits für andere Prüfung ausgefertigt wurde.

Tobias Oberholzer Janine Süess

Agreement for free-download

We hereby agree to make our source code of this project freely available for download from the web pages of the SOMS chair. Furthermore, we assure that all source code is written by ourselves and is not violating any copyright restrictions.

Tobias Oberholzer Janine Süess

Contents

1. Individual Contribution....4

2. Introduction and Motivation....4

3. Description of the Models and Implementation....5

3.1 Main ideas of our models....5

3.1.1 Why is the spreading of a new drug depending on communication?....5

3.1.2 starting position of our model....5

3.2 Main Program....7

3.3 Distribution of the number of contacts....7

3.4 Model 1: Decision-making dependant on communication....8

3.4.1 Model 1: Running the model once....8

3.4.2 Model 1: Running the model certain times....8

3.5 Model 2: Decision-making dependant on communication and feedback....9

3.5.1 Model 2: Running the model once....9

3.5.2 Model 2: Running the model certain times....9

3.6 Dependance of Model 1 & 2 on changeable influences....10

4. Simulation Results and Discussion....11

4.1 Distribution of the number of contacts....11

4.2 Model 1: Decision-making dependant on communication....12

4.2.1 Model 1: Running the model once....12

4.2.2 Model 1: Running the model certain times....13

4.3 Model 2: Decision-making dependant on communication and feedback....14

4.3.1 Model 2: Running the model once....14

4.3.2 Model 2: Running the model certain times....15

4.4 Comparing Model 1 & Model 2....16

4.5 Dependance of Model 1 on changeable influences....17

5. Summary and Outlook....18

5.1 Comparing the results with our goals....18

5.2 Outlook....18

6. Reference....19

7. Appendix, Programs....20

7.1 gessmainprg.m....20

7.2 visconnection02.m....22

7.3 categories02.m....23

7.4 gessmainprg_av1.m & compare01.m....25

7.4.1 gessmainprg_av1.m....25

7.4.2 compare01.m....27

7.5 gessm2negrck.m....28

7.6 gessm2negrck_av2.m & compare02.m....30

7.6.1 gessm2negrck.m....30

7.6.2 compare02.m....34

7.7 threedplot01.m & plot3d01.m....34

7.7.1 threedplot01.m....34

7.7.2 plot3d01.m....38

1. Individual Contribution

The entire project was designed, programmed and realized by both authors simultaneously. Therefore it is not possible to separate the work into individual contributions.

2. Introduction and Motivation

One of the topics people are interested the most in today's life is health. Therefore medical care plays a major role for the modern society. That is the reason why we started to deliberate on the way the information about a new invented drug is spread.

Our goal was to find out on which circumstances the speed of information-spreading is dependant. Therefore we watched the way doctors are introducing a new drug. We would like to find a model which simulates the time passed until a doctor starts using the drug for his patients by observing different influences (for example the communication).

Our first inspiration for the models explained had been taken from the report “The Diffusion of an Innovation among Physicians” written by James Coleman; Elihu Katz; and Herbert Menzel [Ref.1]. In this paper two different models have been explained one of which we took as the main idea of our project. In this model the doctors were divided in different categories depending on the degree of integration in society (communication with other).

We tried to create a model showing the spreading of a drug depending on certain factors influencing the doctors by the process of decision-making. Therefore we basically considered two different expectations. One of them shows the introduction of a new medicament just by observing the communication between doctors while the other one is additionally recording negative feedback of patients.

3. Description of the Models and Implementation

3.1 Main ideas of our models

3.1.1 Why is the spreading of a new drug depending on communication?

The first question we were faced with was which facts are necessary to make a doctor introduce a new medicament. Primarily a doctor needs to get the information that there is a new product available on the market. As a second step somebody has to convince him to use it. Based on this thoughts we decided that doctors should inform each other by keeping in touch their colleague (talking together)

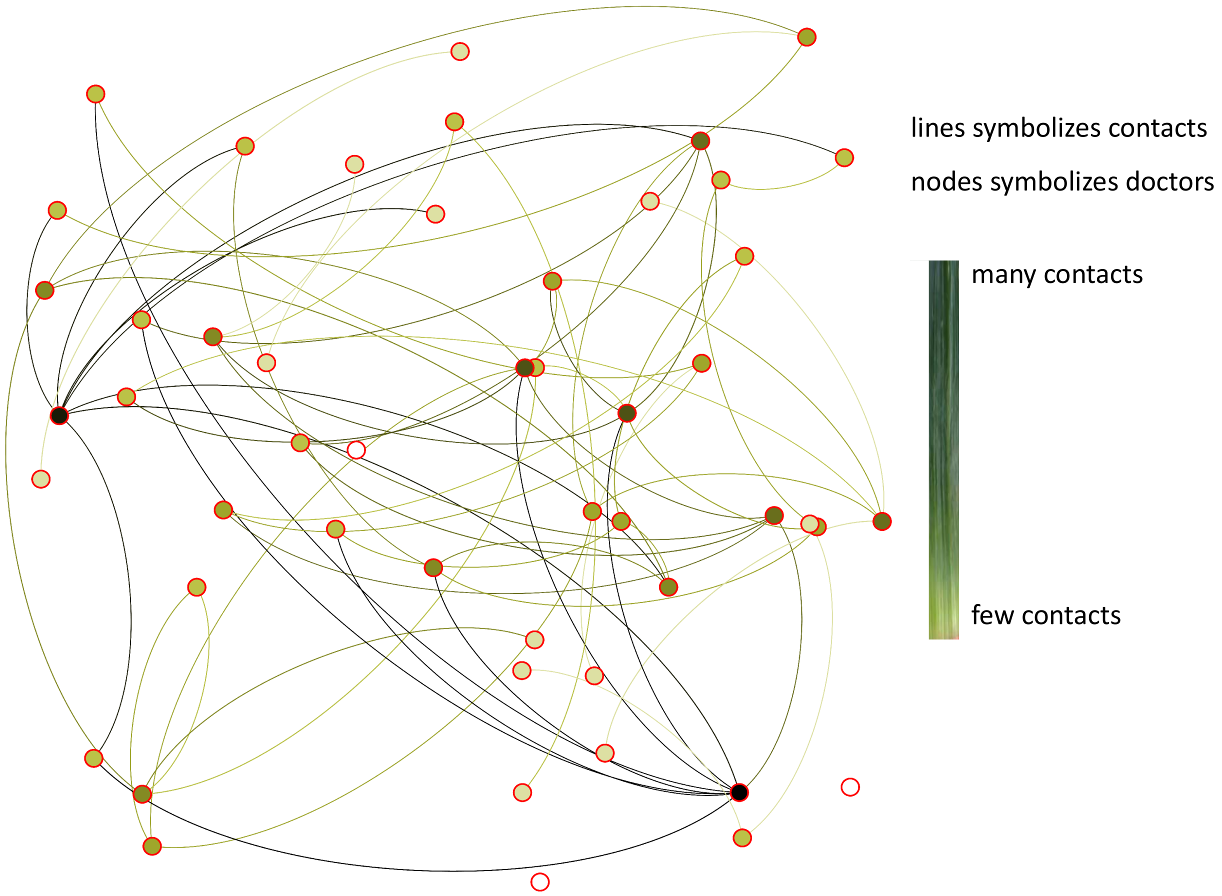
For our models we had to make a few suppositions which will be described shortly. From time to time a doctor speaks to another, the frequency is depending on their character and a little bit different for each of them. Every time two doctors interact, one of them tells “a piece of information” about the new drug which will be remembered by the other. As no personalities would ever be equal every physician is going to need another amount of information before he/she will be persuaded to sell the product to his/her patients.

3.1.2 starting position of our model

We start watching the process at a moment when the first doctors (for example 10%) are already using the medicament. They start to pass the information around which is the reason why it is not absolutely necessary to establish “agents”, information-sources not depending on the communication between the observed doctors.

The second step we have to realize in the beginning is to set the doctors in relation which means to define a social network comprehending contacts of each participant. As a simplification of our model the network is not changing during the simulation. That means that nobody is starting or ending up relations.

Example for a social network (with 50 doctors)



[Ref.2];[Ref.3]

3.2 Main Program

To realize our simulation we made a few programmes containing different ideas how to describe the spreading. All of these functions are elementary structured like the "main program" which is basically a loop allowing doctors to communicate and getting informed about a new product.

Structure of the main program "gessmainprg":

The first step is the initialisation of needed variables, vectors and matrices (including start values). One of the most important of them is the connection-matrix. It makes the decision who is is in contact with whom (knowing each other) which is the foundation to communicate and to exchange news. "connections" is a nxn matrix (n= number of doctors included in the community). After filling the matrix with random numbers and comparing them with a constant factor (depending on the number of connections wished to exist in the network) we receive the connection-matrix containing only zeros and ones by rounding these values. Furthermore there are options defined in the part of initialization to write the data received from the main loop to text files.

The role of the main loop is to let the doctors communicate with each other until most of them started to use the new drug. The loop is a while-loop running as many times as defined with the according variable in the initialization part. Looking at one run, the first step is to decide for each doctor if he/she should have a communication in this round which is depending on coincidence and on how communicative he/she is (number of contacts). If the feedback is positive it must be checked if there are contacts available and then chose one of them for the communication. Afterwards the physician being informed will remember this "piece of information" and probably also start to sell the medicament. All the relevant data calculated in the run will be written to a text file after the described steps have been fulfilled.

3.3 Distribution of the number of contacts

To help us imagine in which relation the physicians being watched in our model stand, the first thing we were interested in is visualizing the distribution of the number of contacts. Therefore we program a function printing an error-bar-plot showing how many doctors do have the same number of contacts. The data giving the information of a doctors contacts is arisen by accessing the main program round by round.

3.4 Model 1: Decision-making dependant on communication

In Model 1 the only factor on which the introduction of the medicament is influenced is the communication between doctors. Therefore we select a value for every doctor (depending on the number of contacts a doctor has) which will be saved in a matrix. The higher the value the higher is the grade of communication (defining the possibility of telling the information to a further doctor). For this case we created two programs. The first one is plotting a graph showing the actual number of physicians using the drug depending on time. Additionally it is able to separate the doctors in three categories demonstrating the degree of integration (are they willing to have a conversation?). The second one compares certain runs of the program observing the moment when a fixed number of doctors started to use the drug.

3.4.1 Model 1: Running the model once

The goal of the program "categories02" was to show a realistic development of a new medicament. Therefore we watch the current number of doctors using the drug as a function of time. "Time" is defined as the number of possibilities a physician already had to talk to someone else. Out of this graph we can read the speed of innovation-diffusion.

Additionally we separated the doctors into three categories depending on their grad of communication. We chose the category "less communicative" for all doctors having less than 45 contacts, the category "medium communicative" for the ones having between 45 and 55 contacts and the category "highly communicative" for all those who do have more than that. This values arose from the mean value of contacts a doctor is supposed to have, which is set at 50.

3.4.2 Model 1: Running the model certain times

By running the main program certain times it can be seen that the result is not exactly the same for every run. In real life it is important to forecast how much time passes on average until a new product is established on the market. This should be realized in the program called "gessmainprg_av1" (and plotted in "compare01"). As far as our experience goes a realistic point to speak about "the product is established " is when at least 90% are using the drug (in our case 450 doctors). Therefore the loop of our main program will be run a hundred times and the amount of time needed to reach the value necessary will be saved to a text file after each run. Within a second program we visualize this values and calculate the mean value.

3.5 Model 2: Decision-making dependant on communication and feedback

In a second step we tried to optimize our first model by adding a second factor influencing the doctors by the process of decision-making. This factor should be understand as negative feedback they got for example from their patients. For Model 2 we again created two programs. The first one is plotting a graph showing the actual number of physicians using the drug depending on time. The second one compares certain runs of the program observing the moment when a fixed number of doctors started to use the drug.

3.5.1 Model 2: Running the model once

As for Model 1, in "gessm2negrck", we watch the current number of doctors using the drug as a function of time and read the speed of diffusion. The difference to Model 1 is that we allocate a number between zero and two for each doctor which represents the amount of negative feedback he/she got from patients. The higher the number representing the feedback of a doctor the lower the chance that he/she is going to recommend the drug. That means if there are many people dissatisfied the number is going to rise.

3.5.2 Model 2: Running the model certain times

In a further program named "gessm2negrck_av2" ("compare02" to make a plot) we write the time-value to a text file, when 90% of our doctors have started using the product. Once more this will be done a hundred times. Subsequently we create another program which draws a graphic showing all these values and visualizing the mean value of them. The result of this programs can be compared easily with the outcome of Model 1.

3.6 Dependance of Model 1 & 2 on changeable influences

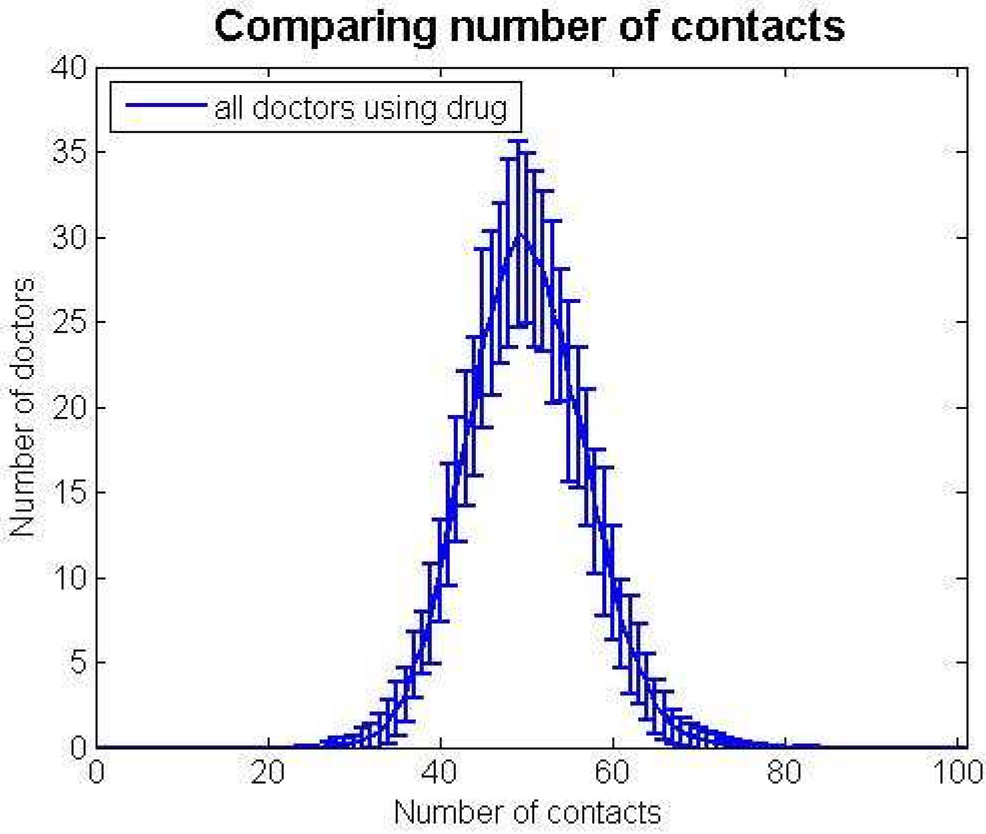
As a last step we take a closer look at the point of time when 90% of the doctors have started using the drug. Our idea was to make this value dependant on the number of doctors using the drug in the beginning and on the average of needed effort to convince somebody (how often do we have to tell him/her about the information). For this reason we generated a 3D-Plot showing this time-values as a function of this two factors. This will be done in the programs "threedplot01" & "threedplot02" ("plot3d01" & "plot3d02" to visualize the values)

4. Simulation Results and Discussion

4.1 Distribution of the number of contacts

The output of the program “visconnection02” is a graph showing us the doctors having the same number of contacts. This graph originate from the average of a hundred runs. This is the reason why we calculated the error-bar-plot too.

A few things which attract attention are that the distribution is symmetric to the value of 50 which is also the number of contacts appearing the most. For a social network of 500 participants this looks like a quit reasonable curve. Furthermore we can see that the variation reaches the maximum at the point at which the graph has its extrema. This makes sense because of the probability distribution.

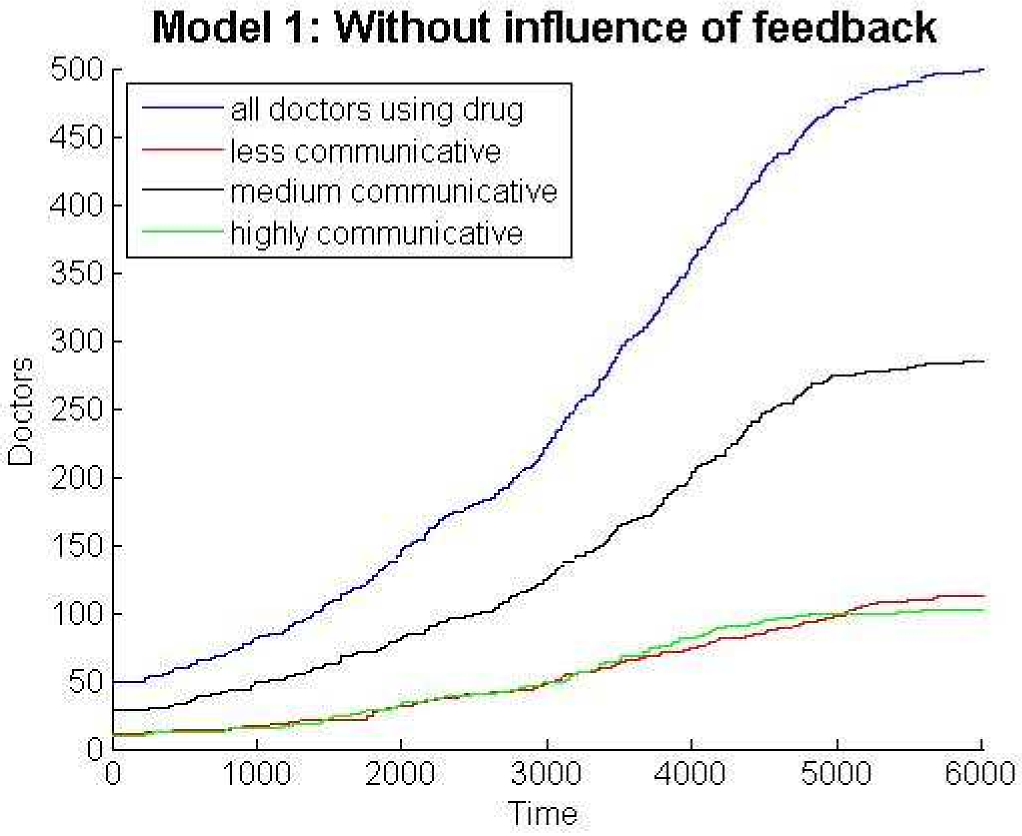


4.2 Model 1: Decision-making dependant on communication

4.2.1 Model 1: Running the model once

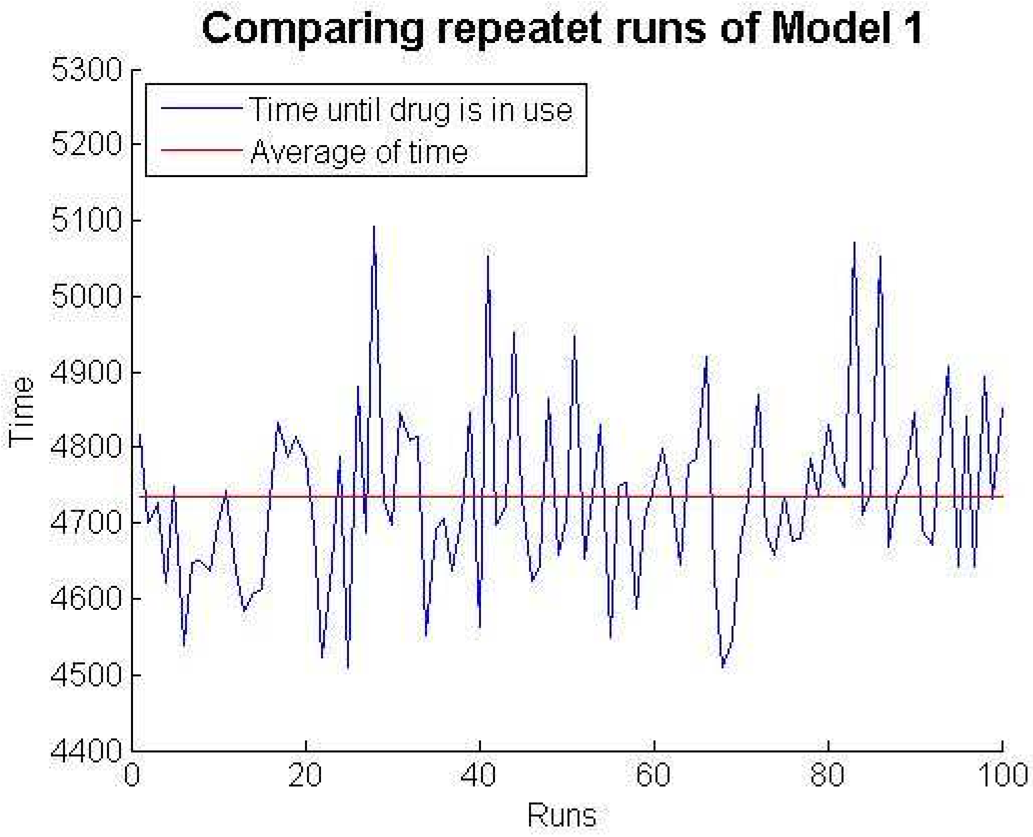
The main curve of this graph is the blue one showing the number of doctors using the drug depending on the time already passed. The remaining three curves are the ones resulting from the blue one by dividing the physicians into three categories. Thereby each category contains just the doctors having a specific number of relations to the other ones.

All in all the doctor being more communicative are getting the information sooner then the other ones. But more interesting is the development of the purple and the green curve. We can see that the green one looses on gradient sooner then the purple one even though the gradient of it was higher then the one of the less communicative in the beginning. The interpretation of this phenomenon is the highly communicative doctors are more likely to communicate in the beginning (coming from their nature) while they get a loss of colleagues to tell in the end because they already told most of them. Because all the doctors being more likely to tell the information do have the same (high) number of contacts which are willing to tell them about an information the scheme looks like the following.



4.2.2 Model 1: Running the model certain times

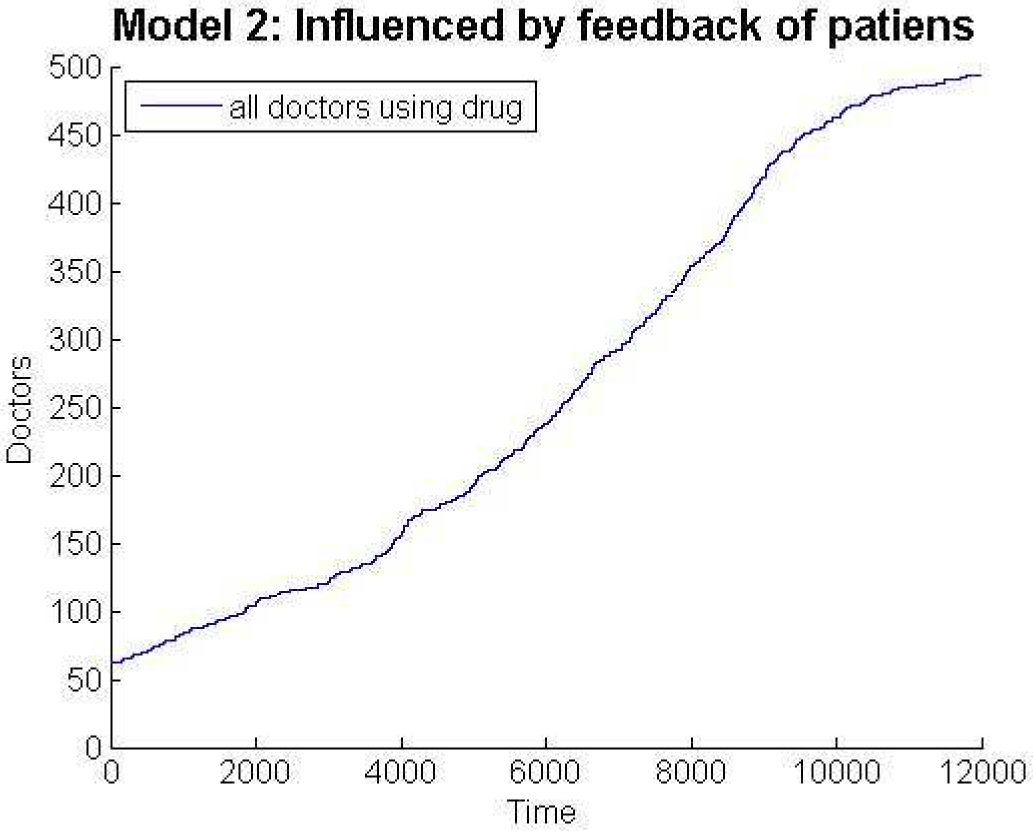
This graph is showing the time until the medicament is in use (90% of the doctors are selling it) for a hundred runs of the loop. To make a forecast how long it will take this drug to be achieved in real life the purple value, which is the mean value of the blue curve, is of a high importance.



4.3 Model 2: Decision-making dependant on communication and feedback

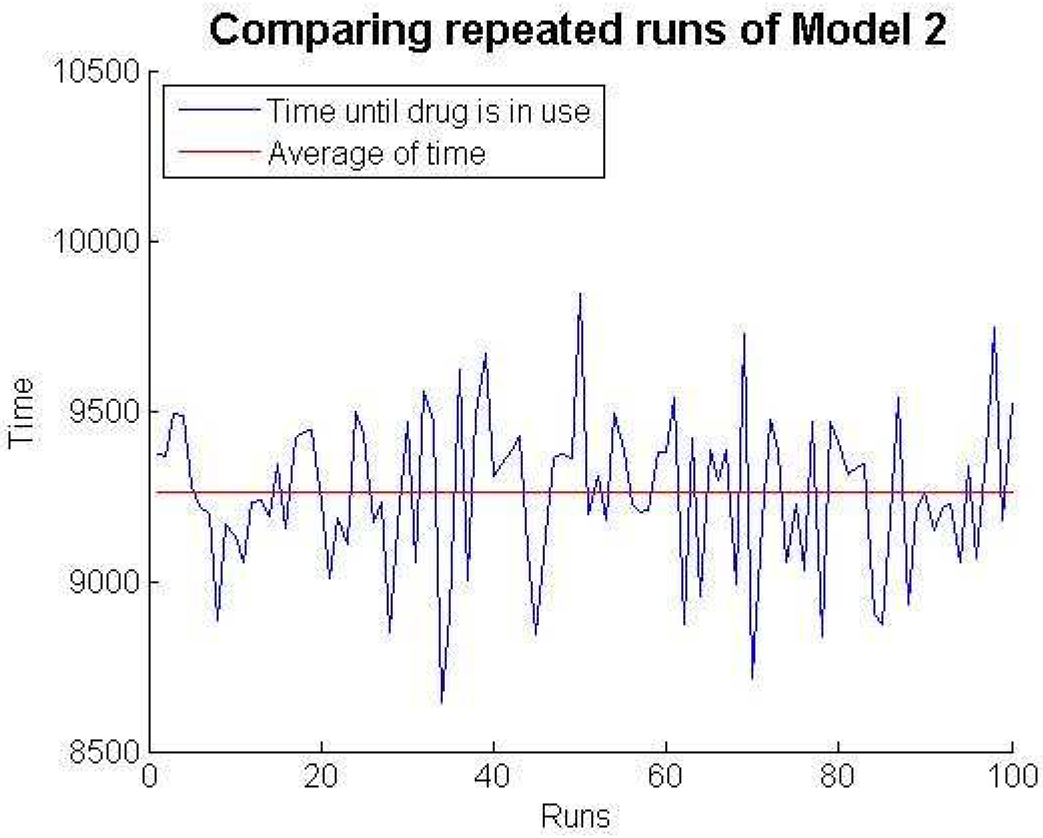
4.3.1 Model 2: Running the model once

For the second model this graph shows the number of doctors using the drug depending on the time already passed as the blue curve in the first model. The only difference is that additionally negative feedback of patients are included. The effect of this is obviously that the speed of information-spreading is lower than before.



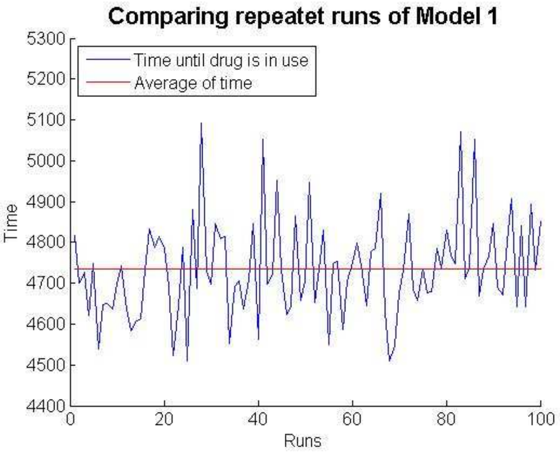
4.3.2 Model 2: Running the model certain times

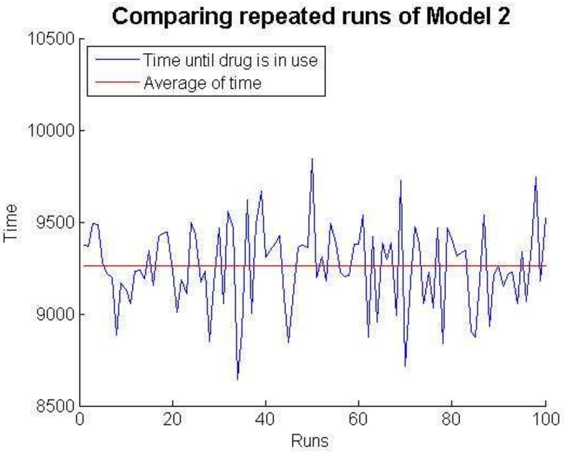
In this graph again it is shown the time needed until the medicament is in use (90% of the doctors are selling it) for a hundred runs of the loop for model 2 (including the mean value).



4.4 Comparing Model 1 & Model 2

Comparing the graphs from model 1 & 2 we can see only one big difference. This is the time-value which is more or less twice as high for model 2 than for model 1. Obviously the variation of the time-values of the blue curve is also bigger for model 2 then for model 1 because the duration is much longer.

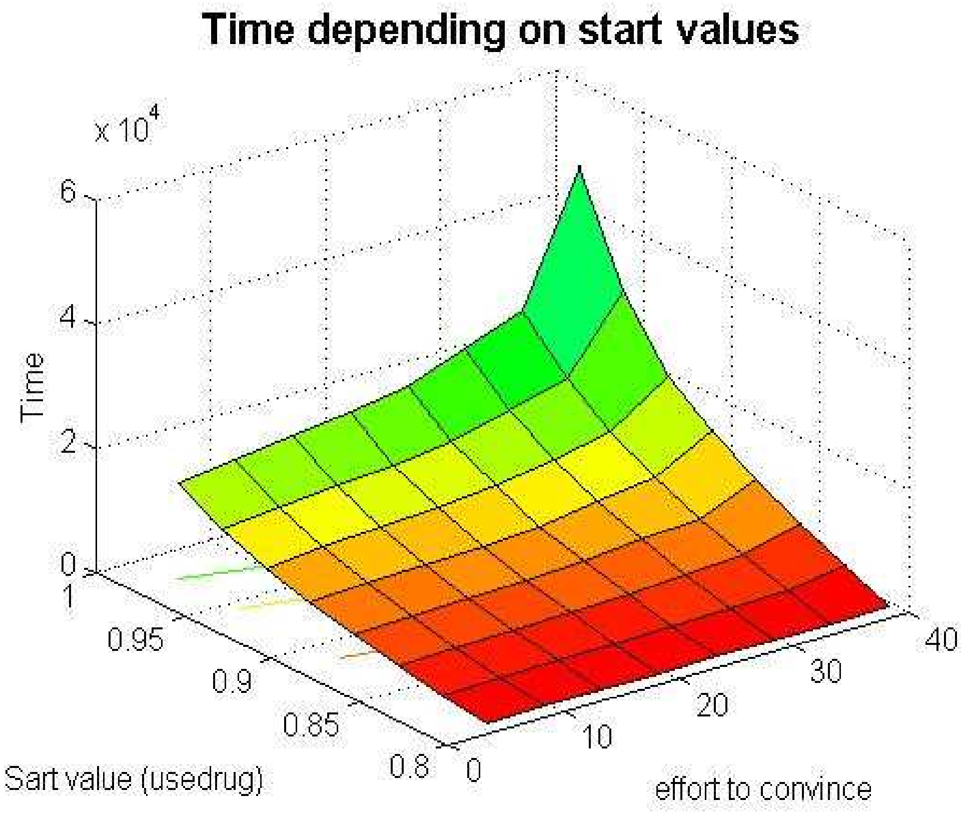




4.5 Dependance of Model 1 on changeable influences

The following graph shows the time passed until the introduction of a drug depending on two factors. The first one was the effort needed to convince a doctor and the second one the number of doctors using the drug in the beginning. The graph is increasing the more the higher the factors are.

The important message we can get from the graphics is that the innovation-diffusion is dependant on the society in a quiet high degree. For real life this means that the introduction of a new drug could be different for example for networks of doctors belonging to a different culture.



5. Summary and Outlook

5.1 Comparing the results with our goals

Our goal was to find a model for innovation diffusion which is as realistic as possible.

We were able to find a simple model which illustrates the number of doctors starting to use a new drug depending on time. This model only works for a closed system which means a network establishing of a constant number of physicians having a constant number of contacts to other people belonging to this social network.

Our model shows the situation of a new medicament getting introduced to the market in a more or less realistic way as far as we can say depending on our experience. But at this point it has to be mentioned that we made many simplifications and arbitrary assumption. There are still a lot of optimizations possible.

5.2 Outlook

For a further research there are almost infinite many open questions and improvements of our models to realize. A few ideas could be:

- involving “agents” , which means to include external information sources (for example pharmacy-agents, professional journal, lectures,...) into the model
- understanding the influence of negative feedback better (specify the importance of a feedback (not every feedback has the same influence on the further development), possibility that a drug would never be able to insist on the market)
- changes in the social network (there will be new friendships and broken up contacts within time)
- better specification of the start values (scientifically foundation)
- competition on the market (different drugs available)
- ...

6. Reference

Ref.1: James Coleman; Elihu Katz; Herbert Menzel,The Diffusion of an Innovation Among Physicians,Sociometry, Vol. 20, No. 4. (Dec., 1957), pp. 253-270, http://links.jstor.org/sici?sici=0038-0431%28195712%2920%3A4%3C253%3ATDOAIA%3E2.0.CO%3B2-N

Ref.2: http://gephi.org/ date:2010-12-07

Ref.3: http://www.google.ch/imgres?imgurl=http://jmvidal.cse.sc.edu/csce242/spring09/farmersmarket/leek.jpg&imgrefurl=http://www.bbc.co.uk/dna/mbfood/NF2670471%3Fthread%3D6948461%26post%3D86142171&usg=__L3LmxeEnsnGAGYUMKtfcfL658B4=&h=1944&w=2592&sz=963&hl=de&start=67&zoom=1&tbnid=5S6DTdnjXuPgYM:&tbnh=133&tbnw=174&prev=/images%3Fq%3Dleek%2Bbbc%26hl%3Dde%26biw%3D1176%26bih%3D577%26gbv%3D2%26tbs%3Disch:10%2C2157&itbs=1&iact=hc&vpx=257&vpy=89&dur=397&hovh=133&hovw=177&tx=177&ty=121&ei=y6MHTcmlD8TPhAe_tPztBw&oei=uqMHTaPnMdKD4QbA8MnzCQ&esq=8&page=5&ndsp=18&ved=1t:429,r:13,s:67&biw=1176&bih=577 date: 2010-12-14

7. Appendix, Programs

7.1 gessmainprg.m

```
function gessmainprg;

%Modelling the diffusion of a new drug by comparing the point of time at which a
%specific number of doctors start selling it to the patients. The moment a
%doctor starts using the drug is depending on his/her personality (how open-
%minded he/she is) and on his/her social network (how many contacts to other
%doctors)

%initialisation
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
n=500; %number of doctors beeing included in the system
m=14; %effort to convince a doctor using the drug
q=0.9; %value to choose the number of doctors using the drug from the beginning
(0<=q<=1)
time=6000; %defining the number of runs of the loop
t=0; %variable to count the number of runs already made in the current loop
z=1/n*50; %variable to define the average of the number of connections for one
doctor

%create file to save data (being used in further programms)
file_1=fopen('usedrug.txt','w'); %saving the matrix containing the information
if a doctor is using the drug at a specific time
file_2=fopen('sumusedrug.txt','w'); %saving the vector containing the
information how many doctors are using the drug at a specific moment
file_3=fopen('numberofconnections.txt','w'); %saving the vector containing the
information how many contacts a docter has

needcon=1+round(m*rand(1,n)); %defining a vector showing the number of needed
information to convince a doctor to use the drug
usedrug=rand(1,n); %defining a matrix to chose the doctors using the drug in the
beginning
connectionsh=rand(n); %definging a matrix to find out the contacts of a doctor
counthear=zeros(1,n); %counting how many times a doctor got the information
about the new drug

%defining connections between doctors of the system (1: connection, 0: no
connedtion)
for i=1:n
    for j=1:n
        if connectionsh(i,j)>=z;
            connections(i,j)=0;
        else connections(i,j)=1;
        end
    end
end

%correct connection-matrix, count the number of contacts a doctor has and
%define the start values of our model (doctors using the drug in the beginning)
for i=1:n
    connections(i,i)=0; %changing the values of the diagonal to zero (because a
doctor can't speak to himself)
```

```
    for j=1:n
        connections(i,j)=connections(j,i); %correct the values of the
connection-matrix which are beneath the diagonal to the values above the
diagonal
        numbcon(1,j)=sum(connections(:,j)); %defining a vector showing the
number of connections of every doctor
        numbconper(1,j)=numbcon(1,j)/(n-1); %defining how often the doctors
communicate with each other depending on the number of contacts they have
           if usedrug(1,j)>q % defining the doctors using the drug in the
beginning (1: using the new drug, 0: not using the new drug)
              usedrug(1,j)=1;
           else usedrug(1,j)=0;
           end
    end
end

count=sum(usedrug);%creating a variable to count how many doctors are using the
drug at a specific time

%writing the start values to the text files
fprintf(file_1,'%i ',usedrug);
fprintf(file_1,'\r\n');
fprintf(file_2,'%i ',count);
fprintf(file_3,'%i ',numbcon);


%loop to run the model for the specific number of runs: time
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
while t<time

   for j=1:n
       if usedrug(1,j)==1
           if rand<numbconper(1,j) %deciding if the doctor is communicating
(possibility is proportional to the number of contacts of a doctor)
               if sum(connections(:,j))>0 %check if there are connections with
other doctors (find someone to tell the information)
                    doc=round(rand*(n-1))+1; %choose a doctor to communicate
with
                    if connections(doc,j)==1
                       counthear(1,doc)=counthear(1,doc)+1; %increase the count
containing the information how often a doctor got to know about the drug
                       connections(doc,j)=0; %make sure that the two doctors do
not communicate to each other again by changing the connection-matrix
                       connections(j,doc)=0;
                    end
                end
           end
       end
       if needcon(1,j)<=counthear(1,j) %check if there is a doctor starting to
use the new drug from now on
           usedrug(1,j)=1;
       end
   end



t=t+1; %increase the count containing the number of runs of the loop

count=sum(usedrug); %count the number of doctor using the drug at the moment
```

```
%writing the start values to the text files
fprintf(file_1,'%i ',usedrug);
fprintf(file_1,'\r\n');
fprintf(file_2,'%i ',count);

end

end
```

7.2 visconnection02.m

```
function visconnection02;
%plotting the mean value of the distribution of number of connection of the
%doctors by running the main program certain times. Additionally the error-bars
%belonging to the graph will be created

close all;

%initialisation
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
nrun=100; %number of runs to approximate a average curve
n=500; %number of doctors (same as in the main program)
ndifgr=100; %number of different groups of doctors having the same number of
contacts
c(1:nrun,1:ndifgr)=0; %define count-matix


%loop to run the main program nrun-times and import the number of
%connections of each doctor
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
for i=1:nrun
    gessmainprg; %run the main program nrun times
    imp=importdata('numberofconnections.txt'); %save the data of one run as a
vector
    for t=1:n
        sumcon(i,t)=imp(t); %save all data in one matrix
    end
end


%loop to count all the number of connections of each doctor
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
for i=1:nrun
    for t=1:n
        for l=1:ndifgr
            if sumcon(i,t)==l
                c(i,l)=c(i,l)+1; %increase the count of number of equal
connections for all loops
            end
        end
    end
c(i,1:ndifgr);
end
```

```
%Plot graph of mean values inclusive error-bars
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
errorbar(1:ndifgr, mean(c), std(c), 'k' ,'linewidth', 2) %plot the averaged
curve inclusive error bars

%options for plot & legend
xlim([0 (ndifgr+1)])
ylim([0 40])
set(gca,'FontSize',12)
xlabel('Nummber of contacts')
ylabel('Doctors')
title('Comparing number of contacts','FontSize',16,'FontWeight','b')
legend('all doctors using drug','Location','northwest')

end
```

7.3 categories02.m

```
function categories02;

%Modelling the diffusion of a new drug by comparing the point of time at which a
%specific number of doctors start selling it to the patients. The moment a
%doctor starts using the drug is depending on his/her personality (how open-
%minded he/she is) and on his/her social network (how many contacts to other
%doctors) the result will be plotted by splitting the doctors in different
%groups depending on how communicative they are

close all
gessmainprg; %running the main program to get data from it

%import the data being saved while running the main program
sumusedrug=importdata('sumusedrug.txt'); %reading the information how many
doctors are using the drug at a specific moment
numberofconnections=importdata('numberofconnections.txt'); %reading the
information how many contacts a doctor has
usedrug=importdata('usedrug.txt'); %reading the information which doctors are
using the drug at a specific moment

siz=size(usedrug) %get to know the number of runs of the main program and the
number of doctors participating

%initialisation
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
n=siz(2); %number of doctors
T=siz(1); %number of runs (time + 1) of the main program
lessc=45; %limit beneath which it can be said that a doctor is isolated
highlyc=55; %limit above which it can be said that a doctor is very
communicative
fc=0; %counts the doctors having few contacts (isolated)
mc=0; %counts the doctors having medium contact
lc=0; %counts the doctors having a lot of contacts (very communicative)
```

```
%options for the plot
xlim([0 T]);
ylim([0 n]);
set(gca,'FontSize',12)
xlabel('Time')
ylabel('Doctors')
title('Model 1: Without influence of feedback','FontSize',16,'FontWeight','b')
hold on

%loop to count the number of doctors using the drug depending on their character
(communication)
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
for t=1:T
    for i=1:n
        if usedrug(t,i)==1;
            if numberofconnections(i)<lessc;
                fc=fc+1; %counting the doctors which have few contacts (isolated
ones)
            end
            if numberofconnections(i)>highlyc;
                lc=lc+1; %counting the doctors which have a lot of contacts
(very communicative ones)
            end
        end
    end
    mc=sumusedrug(t)-(fc+lc); %counting the doctors which have medium contact

%plotting graphs showing the number of doctors using the drug depending on
%time (for all doctors, less communicative ones, medium communicative ones,
%very communicative ones)
  plot(t,sumusedrug(t));
  plot(t,fc,'r');
  plot(t,mc,'k');
  plot(t,lc,'g');

%reset all counts for the new loop
  fc=0;
  lc=0;
  mc=0;

end

%creating a legend fitting to the plot
legend('all doctors using drug','less communicative','medium
communicative','highly communicative','Location','northwest')

end
```

7.4 gessmainprg_av1.m & compare01.m

7.4.1 gessmainprg_av1.m

```
function gessmainprg_av1;

%Modelling the diffusion of a new drug by comparing the point of time at which a
%specific number of doctors start selling it to the patients. The moment a
%doctor starts using the drug is depending on his/her personality (how open-
%minded he/she is) and on his/her social network (how many contacts to other
%doctors) the focus in this program will be on the time needed until a specific
%number of doctors have introduced the new drug. This information will be
%written to a file being used in the further program compare01

%initialisation5
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
n=500; %number of doctors being included in the system
m=14; %effort to convince a doctor using the drug
q=0.9; %value to choose the number of doctors using the drug from the beginning
(0<=q<=1)
time=6000; %defining the number of runs of the loop
t=0; %variable to count the number of runs already made in the current loop
z=1/n*50; %variable to define the average of the number of connections for one
doctor
comdoc=450; %number of doctors using the drug by which we compare the runs of
the program
av=0; %variable remembering if there are already 450 doctors using the drug
r=0; %variable to count the number of runs of the program
runs=100; %number of runs of the program

%create file to save data (being used in further programs)
file_4=fopen('compare1.txt','w');

needcon=1+round(m*rand(1,n)); %defining a vector showing the number of needed
information to convince a doctor to use the drug
usedrug=rand(1,n); %defining a matrix to chose the doctors using the drug in the
beginning
connectionsh=rand(n); %defining a matrix to find out the contacts of a doctor
counthear=zeros(1,n); %counting how many times a doctor got the information
about the new drug
usedrugh=usedrug; %create a variable to remember the circumstances of the
beginning of the model

%defining connections between doctors of the system (1: connection, 0: no
connection)
for i=1:n
    for j=1:n
        if connectionsh(i,j)>=z;
            connections(i,j)=0;
        else connections(i,j)=1;
        end
    end
end



%correct connection-matrix, count the number of contacts a doctor has and
%define the start values of our model (doctors using the drug in the
%beginning)
for i=1:n
    connections(i,i)=0; %changing the values of the diagonal to zero (because a
doctor can't speak to himself)
    for j=1:n
        connections(i,j)=connections(j,i); %correct the values of the
connection-matrix which are beneath the diagonal to the values above the
diagonal
        numbcon(1,j)=sum(connections(:,j)); %defining a vector showing the
number of connections of every doctor
        numbconper(1,j)=numbcon(1,j)/(n-1); %defining how often the doctors
communicate with each other depending on the number of contacts they have
           if usedrug(1,j)>q % defining the doctors using the drug in the
beginning (1: using the new drug, 0: not using the new drug)
              usedrug(1,j)=1;
           else usedrug(1,j)=0;
           end
    end
end

%creating variables to remember the circumstances of the beginning of the model
connectionsh=connections;
numbconh=numbcon;
numbconperh=numbconper;
usedrugh=usedrug;

count=sum(usedrug);%creating a variable to count how many doctors are using the
drug at a specific time

%loop to run the program for the specific number of runs: runs
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
while r<runs

%loop to run the model for the specific number of runs: time
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
while t<time

   for j=1:n
       if usedrug(1,j)==1
           if rand<numbconper(1,j) %deciding if the doctor is communicating
(possibility is proportional to the number of contacts of a doctor)
               if sum(connections(:,j))>0 %check if there are connections with
other doctors (find someone to tell the information)
                    doc=round(rand*(n-1))+1; %choose a doctor to communicate
with
                    if connections(doc,j)==1
                       counthear(1,doc)=counthear(1,doc)+1; %increase the count
containing the information how often a doctor got to know about the drug
                       connections(doc,j)=0; %make sure that the two doctors do
not communicate to each other again by changing the connection-matrix
                       connections(j,doc)=0;
                    end
                end
           end
       end
       if needcon(1,j)<=counthear(1,j) %check if there is a doctor starting to
use the new drug from now on
           usedrug(1,j)=1;
       end
   end

t=t+1; %increase the count containing the number of runs of the loop

count=sum(usedrug); %count the number of doctor using the drug at the moment

    if av==0 %find out if the number of needed doctors to compare the model was
already reached before
        if count>=comdoc %if the number of needed doctors is reached by now,
save the value to a file
            fprintf(file_4,'%i ',t);
            av=1;
        end
    end

end

%change the values to the start values to get the same circumstances
t=0;
av=0;
counthear=zeros(1,n);
connections=connectionsh;
numbcon=numbconh;
numbconper=numbconperh;
usedrug=usedrugh;

r=r+1;%increase the count containing the number of runs of the loop

end

end
```

7.4.2 compare01.m

```
function compare01;
%declare how much time needs to pass until the drug is globally in use:
%comparing the time needed until a specific number of doctors started using
%the drug by running the program for model 1 repeatedly (depending on the
%values found out in gessmainprg_av1)

close all
c=importdata('compare1.txt'); %reading the information how much time passed
until a specific number of doctors started using the drug
s=size(c);

%options for the plot
xlim([0 s(2)]);
ylim([4400 5300]);
set(gca,'FontSize',12)
xlabel('Runs')
ylabel('Time')
title('Comparing repeatet runs of Model 1','FontSize',16,'FontWeight','b')
hold on

m=mean(c); %calculate the mean value of time
M(1,1:s(2))=m;
```

```
%plot graph with time-values & mean value
plot(c);
plot(M,'r');
legend('Time until drug is in use','Average of time','location','Northwest')
%add a legend to the plot

end
```

7.5 gessm2negrck.m

```
function gessm2negrck;

%Modelling the diffusion of a new drug by comparing the point of time at which a
%specific number of doctors start selling it to the patients. The moment a
%doctor starts using the drug is depending on his/her personality (how open-
%minded he/she is) and on his/her social network (how many contacts to other
%doctors) additionally the result is depending on the number of negative
%experience the doctors willing to tell the information have made

close all

%initialisation
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
n=500; %number of doctors being included in the system
m=14; %effort to convince a doctor using the drug
q=0.9; %value to choose the number of doctors using the drug from the beginning
(0<=q<=1)
time=12000; %defining the number of runs of the loop
t=0; %variable to count the number of runs already made in the current loop
z=1/n*50; %variable to define the average of the number of connections for one
doctor

%options for the plot
xlim([0 time]);
ylim([0 n]);
set(gca,'FontSize',12)
xlabel('Time')
ylabel('Doctors')
title('Model 2: Influenced by feedback of
patiens','FontSize',16,'FontWeight','b')
hold on

needcon=1+round(m*rand(1,n)); %defining a vector showing the number of needed
information to convince a doctor to use the drug
usedrug=rand(1,n); %defining a matrix to chose the doctors using the drug in the
beginning
connectionsh=rand(n); %defining a matrix to find out the contacts of a doctor
counthear=zeros(1,n); %counting how many times a doctor got the information
about the new drug
feedbackh=rand(1,n); %defining a variable feedback which contains the
information how often a doctor had a bad experience with the drug (for example
negative feedback of a patient)
for i=1:n %define three categories decided by the number of negative feedback
the doctors get
    if feedbackh(i)<0.4;
        feedback(i)=0;
    else
```

```
        if feedbackh(i)>0.7;
            feedback(i)=2;
        else feedback(i)=1;
        end
    end
end

%defining connections between doctors of the system (1: connection, 0: no
connection)
for i=1:n
    for j=1:n
        if connectionsh(i,j)>=z;
            connections(i,j)=0;
        else connections(i,j)=1;
        end
    end
end

%correct connection-matrix, count the number of contacts a doctor has and
%define the start values of our model (doctors using the drug in the
%beginning)
for i=1:n
    connections(i,i)=0; %changing the values of the diagonal to zero (because a
doctor can't speak to himself)
    for j=1:n
        connections(i,j)=connections(j,i); %correct the values of the
connection-matrix which are beneath the diagonal to the values above the
diagonal
        numbcon(1,j)=sum(connections(:,j)); %defining a vector showing the
number of connections of every doctor
        numbconper(1,j)=numbcon(1,j)/(n-1); %defining how often the doctors
communicate with each other depending on the number of contacts they have
           if usedrug(1,j)>q % defining the doctors using the drug in the
beginning (1: using the new drug, 0: not using the new drug)
              usedrug(1,j)=1;
           else usedrug(1,j)=0;
           end
    end
end


count=sum(usedrug);%creating a variable to count how many doctors are using the
drug at a specific time
plot(t,count) %plot the initial values

%loop to run the model for the specific number of runs: time
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
while t<time

   for j=1:n
       if usedrug(1,j)==1
           if rand<numbconper(1,j) %deciding if the doctor is communicating
(possibility is proportional to the number of contacts of a doctor)
               if sum(connections(:,j))>0 %check if there are connections with
other doctors (find someone to tell the information)
                doc=round(rand*(n-1))+1; %choose a doctor to communicate with
                    if connections(doc,j)==1
                       if feedback(j)==0; %if the doctor got less negative
feedback the other listens to him for sure
```

```
                           counthear(1,doc)=counthear(1,doc)+1; %tell the chosen
doctor about the new drug
                           connections(doc,j)=0; %make sure that the two doctors
do not communicate to each other again by changing the connection-matrix
                           connections(j,doc)=0;
                       end
                       if feedback(j)==1; %if the doctor got a few negative
feedback the chance that the other listens to him is about 50%
                           if rand<0.5;
                               counthear(1,doc)=counthear(1,doc)+1;
                               connections(doc,j)=0;
                               connections(j,doc)=0;
                           end
                       end
                       if feedback(j)==2; %if the doctor got a lot of negative
feedback the chance that the other listens to him is about 30%
                           if rand<0.3;
                               conthear(1,doc)=counthear(1,doc)+1;
                               connections(doc,j)=0;
                               connections(j,doc)=0;
                           end
                       end
                    end
                end
           end
       end
       if needcon(1,j)<=counthear(1,j) %check if there is a doctor starting to
use the new drug from now on
           usedrug(1,j)=1;
       end
   end


t=t+1; %increase the count containing the number of runs of the loop

count=sum(usedrug); %count the number of doctor using the drug at the moment
set(line(t,count),'Color','blue') % plotting the number of doctors using the
drug at the moment into the graphic

end

legend('all doctors using drug','Location','northwest') %add a legend to the
plot

end
```

7.6 gessm2negrck_av2.m & compare02.m

7.6.1 gessm2negrck.m

```
function gessm2negfck_av2;

%Modelling the diffusion of a new drug by comparing the point of time at which a
%specific number of doctors start selling it to the patients. The moment a
%doctor starts using the drug is depending on his/her personality (how open-
%minded he/she is) and on his/her social network (how many contacts to other
%doctors) additionally the result is depending on the number of negative c the
```

```
%doctors willing to tell the information have made
%the focus in this program will be on the time needed until a specific
%number of doctors have introduced the new drug. This information will be
%written to a file being used in the further program compare01


close all

%initialisation
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
n=500; %number of doctors being included in the system
m=14; %effort to convince a doctor using the drug
q=0.9; %value to choose the number of doctors using the drug from the beginning
(0<=q<=1)
time=12000; %defining the number of runs of the loop
t=0; %variable to count the number of runs already made in the current loop
z=1/n*50; %variable to define the average of the number of connections for one
doctor
comdoc=450; %number of doctors using the drug by which we compare the runs of
the program
av=0; %variable remembering if there are already 450 doctors using the drug
r=0; %variable to count the number of runs of the program
runs=100; %number of runs of the program

%create file to save data (being used in further programs)
file_4=fopen('compare2.txt','w');

needcon=1+round(m*rand(1,n)); %defining a vector showing the number of needed
information to convince a doctor to use the drug
usedrug=rand(1,n); %defining a matrix to chose the doctors using the drug in the
beginning
connectionsh=rand(n); %defining a matrix to find out the contacts of a doctor
counthear=zeros(1,n); %counting how many times a doctor got the information
about the new drug
feedbackh=rand(1,n); %defining a variable feedback which contains the
information how often a doctor had a bad experience with the drug (for example
negative feedback of a patient)
usedrugh=usedrug; %create a variable to remember the circumstances of the
beginning of the model

for i=1:n %define three categories decided by the number of negative feedback
the doctors get
    if feedbackh(i)<0.4;
        feedback(i)=0;
    else
        if feedbackh(i)>0.7;
            feedback(i)=2;
        else feedback(i)=1;
        end
    end
end

%defining connections between doctors of the system (1: connection, 0: no
connection)
for i=1:n
    for j=1:n
        if connectionsh(i,j)>=z;
            connections(i,j)=0;
        else connections(i,j)=1;
        end
    end
end

%correct connection-matrix, count the number of contacts a doctor has and
%define the start values of our model (doctors using the drug in the
%beginning)
for i=1:n
    connections(i,i)=0; %changing the values of the diagonal to zero (because a
doctor can't speak to himself)
    for j=1:n
        connections(i,j)=connections(j,i); %correct the values of the
connection-matrix which are beneath the diagonal to the values above the
diagonal
        numbcon(1,j)=sum(connections(:,j)); %defining a vector showing the
number of connections of every doctor
        numbconper(1,j)=numbcon(1,j)/(n-1); %defining how often the doctors
communicate with each other depending on the number of contacts they have
           if usedrug(1,j)>q % defining the doctors using the drug in the
beginning (1: using the new drug, 0: not using the new drug)
              usedrug(1,j)=1;
           else usedrug(1,j)=0;
           end
    end
end

%creating variables to remember the circumstances of the beginning of the model
connectionsh=connections;
numbconh=numbcon;
numbconperh=numbconper;
usedrugh=usedrug;

count=sum(usedrug);%creating a variable to count how many doctors are using the
drug at a specific time


%loop to run the program for the specific number of runs: runs
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
while r<runs

%loop to run the model for the specific number of runs: time
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
while t<time

   for j=1:n
       if usedrug(1,j)==1
           if rand<numbconper(1,j) %deciding if the doctor is communicating
(possibility is proportional to the number of contacts of a doctor)
               if sum(connections(:,j))>0 %check if there are connections with
other doctors (find someone to tell the information)
                doc=round(rand*(n-1))+1; %choose a doctor to communicate with
                    if connections(doc,j)==1
                       if feedback(j)==0; %if the doctor got less negative
feedback the other listens to him for sure
                           counthear(1,doc)=counthear(1,doc)+1; %tell the chosen
doctor about the new drug
                           connections(doc,j)=0; %make sure that the two doctors
do not communicate to each other again by changing the connection-matrix
                           connections(j,doc)=0;
                       end
                       if feedback(j)==1; %if the doctor got a few negative 
feedback the chance that the other listens to him is about 50%
                           if rand<0.5;
                               counthear(1,doc)=counthear(1,doc)+1;
                               connections(doc,j)=0;
                               connections(j,doc)=0;
                           end
                       end
                       if feedback(j)==2; %if the doctor got a lot of negative 
feedback the chance that the other listens to him is about 30%
                           if rand<0.3;
                               conthear(1,doc)=counthear(1,doc)+1;
                               connections(doc,j)=0;
                               connections(j,doc)=0;
                           end
                       end
                    end
                end
           end
       end
       if needcon(1,j)<=counthear(1,j) %check if there is a doctor starting to 
use the new drug from now on
           usedrug(1,j)=1;
       end
   end


t=t+1; %increase the count containing the number of runs of the loop

count=sum(usedrug); %count the number of doctor using the drug at the moment

    if av==0 %find out if the number of needed doctors to compare the model was 
already reached before
        if count>=comdoc %if the number of needed doctors is reached by now, 
save the value to a file
            fprintf(file_4,'%i ',t);
            av=1;
        end
    end

end

%change the values to the start values to get the same circumstances
t=0;
av=0;
counthear=zeros(1,n);
connections=connectionsh;
numbcon=numbconh;
numbconper=numbconperh;
usedrug=usedrugh;

r=r+1;%increase the count containing the number of runs of the loop

end

end
```

7.6.2 compare02.m

```
function compare02;

%declare how much time needs to pass until the drug is globally in use:
%comparing the time needed until a specific number of doctors started using
%the drug by running the program for model 2 repeatedly (depending on the
%values found out in gessm2negrck_av2)

close all
c=importdata('compare2.txt'); %reading the information how much time passed
until a specific number of doctors started using the drug
s=size(c);

%options for the plot
xlim([0 s(2)]);
ylim([8500 10500]);
set(gca,'FontSize',12)
xlabel('Runs')
ylabel('Time')
title('Comparing repeated runs of Model 2','FontSize',16,'FontWeight','b')
hold on

m=mean(c); %calculate the mean value of time
M(1,1:s(2))=m;

%plot graph with time-values & mean value
plot(c);
plot(M,'r');
legend('Time until drug is in use','Average of time','location','Northwest')
%add a legend to the plot

end
```

7.7 threedplot01.m & plot3d01.m

7.7.1 threedplot01.m

```
function threedplot01;

%finding out how the model is changing in dependance on the effort to convince a
doctor (m) and
%the number of doctors using the drug from the beginning (q) by running the
%loop several times and changing values of m and q


%initialisation
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

```
n=500; %number of doctors being included in the system
m=1; %effort to convince a doctor using the drug
time=50000; %defining the number of runs of the loop
t=0; %variable to count the number of runs already made in the current loop
z=1/n*50; %variable to define the average of the number of connections for one
doctor
comdoc=450; %number of doctors using the drug by which we compare the runs of
the program
av=0; %variable remembering if there are already 450 doctors using the drug
runs=10; %number of runs of the program

%create file to save data (being used in further programs)
file_3d=fopen('3dtimes.txt','w');
file_3dplot=fopen('3dplot.txt','w');

usedrug=rand(1,n); %defining a matrix to chose the doctors using the drug in the
beginning
connectionshi=rand(n); %defining a matrix to find out the contacts of a doctor
counthear=zeros(1,n); %counting how many times a doctor got the information
about the new drug
usedrugh1=usedrug; %creating a further variable of usedrug to remember the
starting values

%defining connections between doctors of the system (1: connection, 0: no
connection)
for i=1:n
    for j=1:n
        if connectionshi(i,j)>=z;
            connections(i,j)=0;
        else connections(i,j)=1;
        end
    end
end

%correct connection-matrix, count the number of contacts a doctor has and
%define the start values of our model (doctors using the drug in the
%beginning)
for i=1:n
    connections(i,i)=0; %changing the values of the diagonal to zero (because a
doctor can't speak to himself)
    for j=1:n
        connections(i,j)=connections(j,i); %correct the values of the
connection-matrix which are beneath the diagonal to the values above the
diagonal
        numbcon(1,j)=sum(connections(:,j)); %defining a vector showing the
number of connections of every doctor
        numbconper(1,j)=numbcon(1,j)/(n-1); %defining how often the doctors
communicate with each other depending on the number of contacts they have

    end
end

connectionsh=connections; % creating a further variable of connections to
remember the starting values
ra=rand(1,n); %random number for defining needcon (needs to stay constant for
the whole program)


%loop to run the model for different values of m
```

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
while m<40

     needcon=1+round(m*ra); %defining a vector showing the number of needed
information to convince a doctor to use the drug

q=0.805; %defining starting value for q (smallest one)

%loop to run the model for different values of q
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
while q<1

    for i=1:n
        for j=1:n
            if usedrug(1,j)>q % defining the doctors using the drug in the
beginning (1: using the new drug, 0: not using the new drug)
              usedrug(1,j)=1;
           else usedrug(1,j)=0;
            end
        end
    end
count=sum(usedrug); %defining a value counting how many doctors are using the
drug at the moment
usedrugh2=usedrug; %creating a further variable of usedrug to remember the
actual value

r=0; %defining starting value for r

%loop to run the program for the specific number of runs: runs
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
while r<runs

%loop to run the model for the specific number of runs: time
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
while t<time

   for j=1:n
       if usedrug(1,j)==1
           if rand<numbconper(1,j) %deciding if the doctor is communicating
(possibility is proportional to the number of contacts of a doctor)
               if sum(connections(:,j))>0 %check if there are connections with
other doctors (find someone to tell the information)
                    doc=round(rand*(n-1))+1; %choose a doctor to communicate
with
                    if connections(doc,j)==1
                       counthear(1,doc)=counthear(1,doc)+1; %increase the count
containing the information how often a doctor got to know about the drug
                       connections(doc,j)=0; %make sure that the two doctors do
not communicate to each other again by changing the connection-matrix
                       connections(j,doc)=0;
                    end
                end
           end
       end
       if needcon(1,j)<=counthear(1,j) %check if there is a doctor starting to
use the new drug from now on
           usedrug(1,j)=1;
       end
   end
```

```
t=t+1; %increase the count containing the number of runs of the loop

count=sum(usedrug); %count the number of doctor using the drug at the moment

    if av==0 %find out if the number of needed doctors to compare the model was
already reached before
        if count>=comdoc %if the number of needed doctors is reached by now,
save the value to a file
            fprintf(file_3d,'%i ',t);
            av=1;
        end
    end

end %end time

%change the values to the start values to get the same circumstances
t=0;
av=0;
counthear=zeros(1,n);
connections=connectionsh;
usedrug=usedrugh2;
count=sum(usedrug);

r=r+1;%increase the count containing the number of runs of the loop

end %end runs

%changing the values to the start values to get the same circumstances
usedrug=usedrugh1;

%writing the data to a text file and find the mean value
imp=importdata('3dtimes.txt');
fclose(file_3d);
file_3d=fopen('3dtimes.txt','w');
M=mean(imp);
fprintf(file_3dplot,'%i ',M);

q=q+0.025; %increase the count containing the number of runs of the loop

end %end q

%writing the data to a text file
fprintf(file_3dplot,'\r\n');

m=m+5; %increasing the count containing the number of runs of the loop

end %end m

end
```

7.7.2 plot3d01.m

```
function plot3d01;

%plotting the time-values depending on the effort to convince a doctor and
%the number of doctors using the drug from the beginning (calculated in
%the program threedplot01)

imp=importdata('3dplot werte1.txt'); %importing the information calculated in
the program threedplot
a=[4 9 14 19 24 29 34 39]; %values for the variable which shows the effort to
convince a doctor
b=[0.805 0.83 0.855 0.88 0.905 0.93 0.955 0.98]; %values for the variable which
definies the number of doctors using the drug from the beginning

surfc(a,b,imp) %plotting values

%options for the plot
colormap hsv
axis([0 40 .8 1 0 60000])
set(gca,'FontSize',12)
xlabel('effort to convince')
ylabel('Sart value (usedrug)')
zlabel('Time')
title('Time depending on start values','FontSize',16,'FontWeight','b')
hold on

end
```