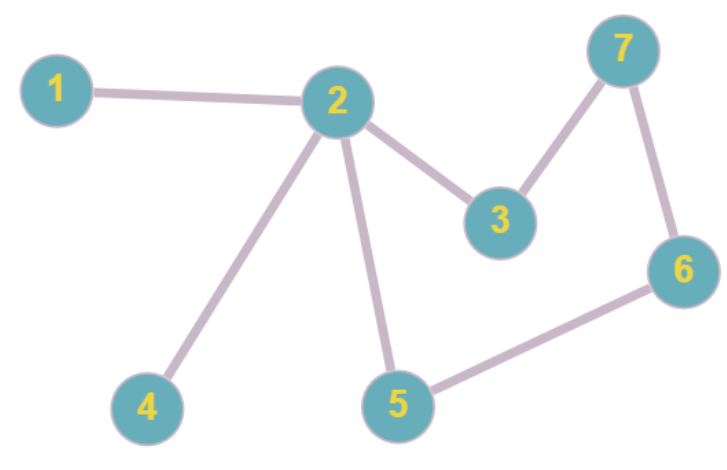
**Übung 10**

**Systembiologie**

**Aufgabe 10.1**



Online software used for the graph: <https://graphonline.ru/en/>

* Degrees of nodes:

1 🡪 1

2 🡪 4

3 🡪 2

4 🡪 1

5 🡪 2

6 🡪 2

7 🡪 2

* Betweenness (the extent to which a particular node lies on the shortest path between other nodes)

Ein Bild, das Text enthält.

Automatisch generierte Beschreibung

Sum of shortest paths having *i* in them divided by the total number of shortest paths of all the nodes to all the other nodes.

Number of potential paths that exist in the network:

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| From | To | Shortest path  between the nodes | | Frequency of occurrence of nodes in between the shortest paths | | | | | | |
|  | | | | **1** | **2** | **3** | **4** | **5** | **6** | **7** |
| 1 | 2 | (1,2) | | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 1 | 3 | (1,2,3) | | 0 | 1 | 0 | 0 | 0 | 0 | 0 |
| 1 | 4 | (1,2,4) | | 0 | 1 | 0 | 0 | 0 | 0 | 0 |
| 1 | 5 | (1,2,5) | | 0 | 1 | 0 | 0 | 0 | 0 | 0 |
| 1 | 6 | (1,2,5,6) | | 0 | 1 | 0 | 0 | 1 | 0 | 0 |
| 1 | 7 | (1,2,3,7) | | 0 | 1 | 1 | 0 | 0 | 0 | 0 |
| 2 | 3 | (2,3) | | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 2 | 4 | (2,4) | | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 2 | 5 | (2,5) | | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 2 | 6 | (2,5,6) | | 0 | 0 | 0 | 0 | 1 | 0 | 0 |
| 2 | 7 | (2,3,7) | | 0 | 0 | 1 | 0 | 0 | 0 | 0 |
| 3 | 4 | (3,2,4) | | 0 | 1 | 0 | 0 | 0 | 0 | 0 |
| 3 | 5 | (3,2,5) | | 0 | 1 | 0 | 0 | 0 | 0 | 0 |
| 3 | 6 | (3,7,6) | | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| 3 | 7 | (3,7) | | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 4 | 5 | (4,2,5) | | 0 | 1 | 0 | 0 | 0 | 0 | 0 |
| 4 | 6 | (4,2,5,6) | | 0 | 1 | 0 | 0 | 1 | 0 | 0 |
| 4 | 7 | (4,2,3,7) | | 0 | 1 | 1 | 0 | 0 | 0 | 0 |
| 5 | 6 | (5,6) | | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 5 | 7 | (5,6,7) | | 0 | 0 | 0 | 0 | 0 | 1 | 0 |
| 6 | 7 | (6,7) | | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
|  | 0 | 10 | 3 | 0 | 3 | 1 | 1 | Sum |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **1** | **2** | **3** | **4** | **5** | **6** | **7** | **Nodes** |
| 0 | 10 | 3 | 0 | 3 | 1 | 1 | Numerator |
| 15 | 15 | 15 | 15 | 15 | 15 | 15 | Denominator |
| 0/15 | 2/3 | 1/5 | 0/15 | 1/5 | 1/15 | 1/15 | Betweenness Score |

* Closeness

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | 1 | 2 | 3 | 4 | 5 | 6 | 7 | Sum |
| 1 | 0 | 1 | 2 | 2 | 2 | 3 | 3 | 13 |
| 2 | 1 | 0 | 1 | 1 | 1 | 2 | 2 | 8 |
| 3 | 2 | 1 | 0 | 2 | 2 | 2 | 1 | 10 |
| 4 | 2 | 1 | 2 | 0 | 2 | 3 | 3 | 13 |
| 5 | 2 | 1 | 2 | 2 | 0 | 1 | 2 | 10 |
| 6 | 3 | 2 | 2 | 3 | 1 | 0 | 1 | 12 |
| 7 | 3 | 2 | 1 | 3 | 2 | 1 | 0 | 12 |

Ein Bild, das Text enthält.

Automatisch generierte Beschreibung

CC(1) = (N-1)/13=6/13 = 0.46

CC(2) = 6/8 = 0.75

CC(3) = 6/10 = 0.6

CC(4) = 6/13 = 0.46

CC(5) = 6/10 = 0.6

CC(6) = 6/12 = 0.5

CC(7) = 6/12 = 0.5

* Clusteringkoeffizient

C(1) = 0

C(2) = 0/6 = 0

C(3) = 0/1 = 0

C(4) = 0

C(5) = 0/1 = 0

C(6) = 0/1 = 0

C(7) = 0/1 = 0

* Komponenten: nur ein
* Dichte:

**Aufgabe 10.2**

> install.packages("igraph")

WARNING: Rtools is required to build R packages but is not currently installed. Please download and install the appropriate version of Rtools before proceeding:

https://cran.rstudio.com/bin/windows/Rtools/

Installing package into ‘C:/Users/raliz/AppData/Local/R/win-library/4.2’

(as ‘lib’ is unspecified)

trying URL 'https://cran.rstudio.com/bin/windows/contrib/4.2/igraph\_1.3.2.zip'

Content type 'application/zip' length 5794714 bytes (5.5 MB)

downloaded 5.5 MB

package ‘igraph’ successfully unpacked and MD5 sums checked

Warning in install.packages :

cannot remove prior installation of package ‘igraph’

Warning in install.packages :

problem copying C:\Users\raliz\AppData\Local\R\win-library\4.2\00LOCK\igraph\libs\x64\igraph.dll to C:\Users\raliz\AppData\Local\R\win-library\4.2\igraph\libs\x64\igraph.dll: Permission denied

Warning in install.packages :

restored ‘igraph’

The downloaded binary packages are in

C:\Users\raliz\AppData\Local\Temp\RtmpGkl8Ld\downloaded\_packages

> library(igraph)

Attaching package: ‘igraph’

The following objects are masked from ‘package:stats’:

decompose, spectrum

The following object is masked from ‘package:base’:

Union

> setwd("~/Universität/12. Semester/Bioinformatik/Übungen/Übung 10")

> data <- read.csv("proteinInteractionNetworkHomoSapiensSmall.csv", header = TRUE)

> data

FROM TO

1 AP4M1 APP

2 APBA1 LRP8

3 APBA1 PSEN1

4 APBA1 APP

5 APBA2 APP

6 APBA2 PSEN1

7 APBB1 APLP1

8 APBB1 LRP1

9 APBB1 APLP2

10 APBB1 NOTCH1

11 APBB1 APP

12 APBB1 LRP8

13 APBB1 KAT5

14 APBB1 APBB2

15 APBB2 APLP1

16 APBB2 APLP2

17 APBB2 APP

18 APH1A APP

19 APH1A NCSTN

20 APH1A PSEN1

21 APH1A PSENEN

22 APLP1 DAB1

23 APLP1 APLP2

24 APLP1 PRNP

25 APLP1 CNTN4

26 APLP1 APP

27 APLP2 MAPK8IP1

28 APLP2 KAT5

29 APLP2 DAB1

30 APLP2 APP

31 APOA1 GPC1

32 APOA1 APOA4

33 APOA1 APP

34 APOA1 CLU

35 APOA1 APOE

36 APOA4 APOE

37 APOA4 GPC1

38 APOA4 APP

39 APOA4 CLU

40 APOE LRP1

41 APOE GPC1

42 APOE TOMM40

43 APOE RELN

44 APOE CLU

45 APOE LRP8

46 APOE SORL1

47 APOE APP

48 APP RTN4R

49 APP NGFR

50 APP NTN1

51 APP MAPK8IP1

52 APP LRP1

53 APP GSAP

54 APP SORL1

55 APP EP300

56 APP GPC1

57 APP IDE

58 APP NOTCH1

59 APP LRP8

60 APP ITGB1

61 APP CASP3

62 APP CNTN4

63 APP TGFB2

64 APP NTRK1

65 APP PSENEN

66 APP GULP1

67 APP F11

68 APP DAB1

69 APP CLU

70 APP CNTN2

71 APP RELN

72 APP KLC1

73 APP ITM2B

74 APP KLK6

75 APP TOMM40

76 APP GRB2

77 APP SPON1

78 APP NAE1

79 APP SHC1

80 APP PITRM1

81 APP PRNP

82 APP KAT5

83 APP BACE1

84 APP PSEN1

85 APP TNFRSF21

86 APP NCSTN

87 BACE1 SORL1

88 BACE1 NCSTN

89 BACE1 ITM2B

90 BACE1 PRNP

91 BACE1 PSEN1

92 CLU LRP8

93 CLU PRNP

94 CNTN2 EP300

95 DAB1 LRP1

96 DAB1 NOTCH1

97 DAB1 LRP8

98 DAB1 ITGB1

99 DAB1 RELN

100 EP300 KAT5

101 EP300 RBPJ

102 EP300 MAML1

103 EP300 NOTCH1

104 EP300 TP53

105 GPC1 PRNP

106 GRB2 NGFR

107 GRB2 PSEN1

108 GRB2 PRNP

109 GRB2 NTRK1

110 GRB2 ITGB1

111 GRB2 SHC1

112 GSAP NCSTN

113 GSAP PSEN1

114 GULP1 LRP1

115 ITGA5 ITGB1

116 ITGB1 LRP1

117 ITGB1 SHC1

118 ITM2B NTRK1

119 KAT5 LRP1

120 KAT5 TP53

121 KAT5 NOTCH1

122 KLC1 MAPK8IP1

123 LRP1 MAPK8IP1

124 LRP1 SORL1

125 LRP1 PRNP

126 LRP1 SHC1

127 LRP8 MAPK8IP1

128 LRP8 NOTCH1

129 LRP8 SPON1

130 LRP8 RELN

131 MAML1 TP53

132 MAML1 NOTCH1

133 MAML1 RBPJ

134 NCSTN NOTCH1

135 NCSTN PSEN1

136 NCSTN PSENEN

137 NGFR RTN4R

138 NGFR TNFRSF21

139 NGFR NTRK1

140 NGFR SHC1

141 NOTCH1 PSEN1

142 NOTCH1 RBPJ

143 NTRK1 PSEN1

144 NTRK1 SHC1

145 PSEN1 PSENEN

146 RBPJ TP53

147 SHC1 TP53

148 TOMM22 TOMM40

> datam <- as.matrix(data)

> graph\_from\_edgelist (datam, directed = FALSE)

IGRAPH 67f4582 UN-- 56 148 --

+ attr: name (v/c)

+ edges from 67f4582 (vertex names):

[1] AP4M1--APP APBA1--LRP8 APBA1--PSEN1 APP --APBA1

[5] APP --APBA2 PSEN1--APBA2 APBB1--APLP1 APBB1--LRP1

[9] APBB1--APLP2 APBB1--NOTCH1 APP --APBB1 LRP8 --APBB1

[13] APBB1--KAT5 APBB1--APBB2 APLP1--APBB2 APLP2--APBB2

[17] APP --APBB2 APP --APH1A APH1A--NCSTN PSEN1--APH1A

[21] APH1A--PSENEN APLP1--DAB1 APLP1--APLP2 APLP1--PRNP

[25] APLP1--CNTN4 APP --APLP1 APLP2--MAPK8IP1 APLP2--KAT5

[29] APLP2--DAB1 APP --APLP2 APOA1--GPC1 APOA1--APOA4

+ ... omitted several edges

> graph <- graph\_from\_edgelist (datam, directed = FALSE)

> plot(graph)

Ein Bild, das Outdoorobjekt, Blume enthält.

Automatisch generierte Beschreibung

> degree(graph,v = V(graph),mode = c("all", "out", "in", "total"),loops = TRUE,normalized = FALSE)

AP4M1 APP APBA1 LRP8 PSEN1 APBA2 APBB1 APLP1

1 50 3 10 11 2 8 7

LRP1 APLP2 NOTCH1 KAT5 APBB2 APH1A NCSTN PSENEN

11 7 10 7 4 4 7 4

DAB1 PRNP CNTN4 MAPK8IP1 APOA1 GPC1 APOA4 CLU

8 7 2 5 5 5 5 6

APOE TOMM40 RELN SORL1 RTN4R NGFR NTN1 GSAP

10 3 4 4 2 6 1 3

EP300 IDE ITGB1 CASP3 TGFB2 NTRK1 GULP1 F11

7 1 6 1 1 6 2 1

CNTN2 KLC1 ITM2B KLK6 GRB2 SPON1 NAE1 SHC1

2 2 3 1 7 2 1 7

PITRM1 BACE1 TNFRSF21 RBPJ MAML1 TP53 ITGA5 TOMM22

1 6 2 4 4 5 1 1

> closeness(

+ graph,

+ vids = V(graph),

+ mode = c("out", "in", "all", "total"),

+ weights = NULL,

+ normalized = FALSE,

+ cutoff = -1

+ )

AP4M1 APP APBA1 LRP8 PSEN1 APBA2

0.008771930 0.016666667 0.008928571 0.009708738 0.009803922 0.008849558

APBB1 APLP1 LRP1 APLP2 NOTCH1 KAT5

0.009615385 0.009259259 0.009803922 0.009345794 0.009803922 0.009523810

APBB2 APH1A NCSTN PSENEN DAB1 PRNP

0.009009009 0.009009009 0.009433962 0.009009009 0.009615385 0.009259259

CNTN4 MAPK8IP1 APOA1 GPC1 APOA4 CLU

0.008849558 0.009090909 0.009090909 0.009090909 0.009090909 0.009174312

APOE TOMM40 RELN SORL1 RTN4R NGFR

0.009615385 0.009009009 0.009009009 0.009009009 0.008849558 0.009259259

NTN1 GSAP EP300 IDE ITGB1 CASP3

0.008771930 0.008928571 0.009523810 0.008771930 0.009345794 0.008771930

TGFB2 NTRK1 GULP1 F11 CNTN2 KLC1

0.008771930 0.009259259 0.008849558 0.008771930 0.009090909 0.008849558

ITM2B KLK6 GRB2 SPON1 NAE1 SHC1

0.008928571 0.008771930 0.009433962 0.008849558 0.008771930 0.009615385

PITRM1 BACE1 TNFRSF21 RBPJ MAML1 TP53

0.008771930 0.009174312 0.008849558 0.006666667 0.006666667 0.006849315

ITGA5 TOMM22

0.006211180 0.006060606

> estimate\_betweenness(

+ graph,

+ vids = V(graph),

+ directed = FALSE,

+ cutoff=0,

+ weights = NULL,

+ nobigint = TRUE

+ )

AP4M1 APP APBA1 LRP8 PSEN1 APBA2

0.0000000 1196.1643939 0.3333333 17.2500000 21.1666667 0.0000000

APBB1 APLP1 LRP1 APLP2 NOTCH1 KAT5

6.7940476 4.6666667 22.6318182 3.9075758 67.7723665 20.2870491

APBB2 APH1A NCSTN PSENEN DAB1 PRNP

0.0000000 0.0000000 5.0000000 0.0000000 11.2690476 6.0333333

CNTN4 MAPK8IP1 APOA1 GPC1 APOA4 CLU

0.0000000 2.1500000 0.2000000 0.9166667 0.2000000 2.0833333

APOE TOMM40 RELN SORL1 RTN4R NGFR

16.4000000 54.0000000 0.2500000 0.8333333 0.0000000 4.0000000

NTN1 GSAP EP300 IDE ITGB1 CASP3

0.0000000 0.0000000 50.8445166 0.0000000 55.1742424 0.0000000

TGFB2 NTRK1 GULP1 F11 CNTN2 KLC1

0.0000000 2.8611111 0.0000000 0.0000000 0.0000000 0.0000000

ITM2B KLK6 GRB2 SPON1 NAE1 SHC1

0.3333333 0.0000000 5.6111111 0.0000000 0.0000000 24.4739177

PITRM1 BACE1 TNFRSF21 RBPJ MAML1 TP53

0.0000000 4.0000000 0.0000000 0.7198773 0.7198773 6.9523810

ITGA5 TOMM22

0.0000000 0.0000000

Warning messages:

1: In estimate\_betweenness(graph, vids = V(graph), directed = FALSE, :

'nobigint' is deprecated since igraph 1.3 and will be removed in igraph 1.4

2: In betweenness(graph, v = vids, directed = directed, cutoff = cutoff, :

`cutoff' == 0 will be treated literally from igraph 1.4. If you want unrestricted betweenness, set it to -1

> max(estimate\_betweenness(

+ graph,

+ vids = V(graph),

+ directed = FALSE,

+ cutoff=0,

+ weights = NULL,

+ nobigint = TRUE

+ ))

[1] 1196.164

Warning messages:

1: In estimate\_betweenness(graph, vids = V(graph), directed = FALSE, :

'nobigint' is deprecated since igraph 1.3 and will be removed in igraph 1.4

2: In betweenness(graph, v = vids, directed = directed, cutoff = cutoff, :

`cutoff' == 0 will be treated literally from igraph 1.4. If you want unrestricted betweenness, set it to -1

D.h. der wichtigste Knoten im Graphen ist APP.

**Aufgabe 10.3**

Freundschaftsparadox: laut dieses Paradoxes haben die Freunde jeder Person mehr Freunde als die Person selbst. Dieses überraschende Phänomen könnte ich wie folgt erklären: Man macht eine Beobachtung hauptsächlich an Orten (in meinem Beispiel ist das ein Ort), an denen sie am leichtesten zu machen ist. Z.B. ich gehe mittags in die Mensa und komme sofort auf den Schluss, dass die Mensa ein Ort ist, der immer voll mit Menschen ist. Dabei ist aber meine Beobachtung *biased* – denn genau meine Entscheidung, mittags die Mensa zu besuchen, wie auch dieselbe Entscheidung der anderen Stundenten, hat zu dieser Beobachtung geführt. D.h. das Gewicht der Subjektivität kann das Ergebnis in eine bestimmte Richtung lenken.

In der Biologie, oder speziell in der Epidemiologie kann man dieses Prinzip sehr gut benutzen.

Eine Ansteckung, die stochastisch einige Personen infiziert und sich dann von Person zu Person im Netzwerk verbreitet, tendiert sich zentral gelegene Personen schneller zu erreichen als periphere Individuen, weil zentrale Individuen eine kleinere Anzahl von Schritten weg von der durchschnittlichen Person im Netzwerk sind.

Jedoch ist es, ein ganzes Netzwerk darzustellen, um die zentralen Einzelpersonen zu identifizieren, teuer, zeitaufwändig und oft unmöglich.

Deswegen gibt es eine alternative Strategie, die dies nicht tut, sondern die auf die Überwachung der Freunde zufällig ausgewählter Personen basiert. Diese Strategie nutzt die obengenannte interessante Eigenschaft menschlicher sozialer Netzwerke aus: Im Durchschnitt besitzen die Freunde zufällig ausgewählter Personen mehr Links (haben einen höheren Grad) und sind auch zentraler (z. B. gemessen durch Betweenness Centrality) zum Netzwerk als die anfänglichen, zufällig ausgewählten Personen. Somit erwarten wir, dass man eine Reihe von Freunden als a infiziert früher findet, als zufällig ausgewählter Personen (die die Bevölkerung als Ganzes repräsentieren). Diese Sensormethode könnte signifikante Ergebnisse und zusätzliche Zeit liefern, um auf Epidemien in kleinen oder großen überwachten Bevölkerungsgruppen zu reagieren.