

Übungsblatt 8

„Mustererkennung“

J. Cavojska, N. Lehmann, R. Toudic

28.06.2015

Inhaltsverzeichnis

1	Aufgabe 1 - Boolesche Funktionen und Perceptrons	2
1.1	1 a) Funktion zur Klassifizierung von Booleschen Funktionen .	2
1.1.1	Code	2
1.2	1 b) Gleichverteilte Punkte auf einer Einheitskugel	3
1.2.1	Code	3
1.2.2	Bilder	5
1.3	1 c) Histogramm für Boolesche Funktionen mit 0, 1	6
1.3.1	Code	6
1.3.2	Resultate	7
1.4	1 d) Histogramm für Boolesche Funktionen mit -1, 1	8
1.4.1	Code	8
1.4.2	Resultate	8

1 Aufgabe 1 - Boolesche Funktionen und Perceptrons

1.1 1 a) Funktion zur Klassifizierung von Booleschen Funktionen

1.1.1 Code

Hauptdatei: *hw08.m*

```
1 v1 = [0.3;0.5;-0.4];
2 v2 = [-0.8;-0.6;0.5];
3 v3 = [0.7;0.6;-1];
4
5 % signature: int classify_vec([float,float,float], {0|-1})
6 result_of_v1 = classify_vec(v1, 0) % f10
7 result_of_v2 = classify_vec(v2, 0) % f1
8 result_of_v3 = classify_vec(v3, 0) % f8
```

Hilfsdatei: *classify_vec.m*

```
1 function [y] = classify_vec(v, x)
2 if f0(v, x)
3     y = 0;
4 elseif f1(v, x)
5     y = 1;
6 elseif f2(v, x)
7     ...
8 elseif f14(v, x)
9     y = 14;
10 elseif f15(v, x)
11     y = 15;
12 else
13     y = 666;
14 end
```

Hilfsdatei: *f0.m*

```
1 % f00*2^0 + f01*2^1 + f10*2^2 + f11*2^3 = 0
2 function [y] = f0(v, x)
3 f00 = x*v(1) + x*v(2) + v(3) < 0;
4 f01 = x*v(1) + 1*v(2) + v(3) < 0;
5 f10 = 1*v(1) + x*v(2) + v(3) < 0;
6 f11 = 1*v(1) + 1*v(2) + v(3) < 0;
7 if f00 && f01 && f10 && f11
8     y = 1;
9 else
10     y = 0;
```

```
11 end
```

Hilfsdatei: *f1.m*

```
1 % f00*2^0 + f01*2^1 + f10*2^2 + f11*2^3 = 1
2 function [y] = f1(v, x)
3 f00 = x*v(1) + x*v(2) + v(3) >= 0;
4 f01 = x*v(1) + 1*v(2) + v(3) < 0;
5 f10 = 1*v(1) + x*v(2) + v(3) < 0;
6 f11 = 1*v(1) + 1*v(2) + v(3) < 0;
7 if f00 && f01 && f10 && f11
8     y = 1;
9 else
10     y = 0;
11 end
```

Die Hilfsdateien *f2.m* bis *f15.m* sind analog aufgebaut.

1.2 1 b) Gleichverteilte Punkte auf einer Einheitskugel

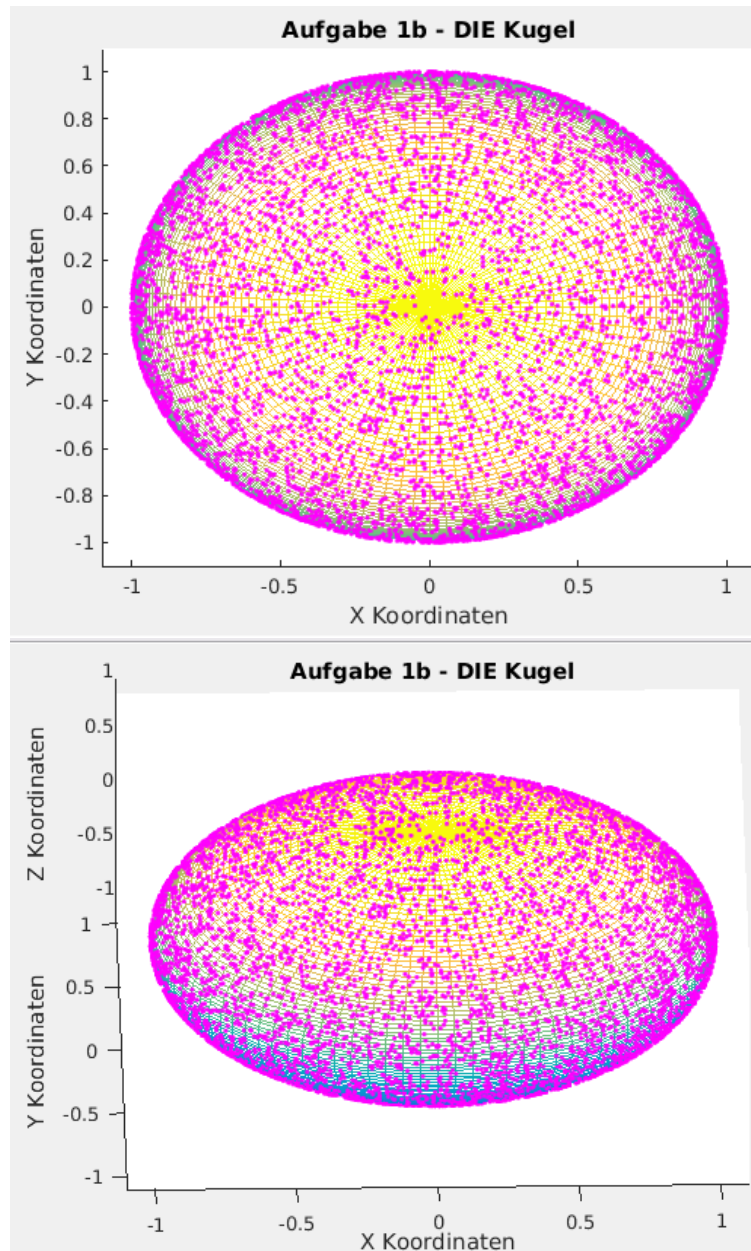
1.2.1 Code

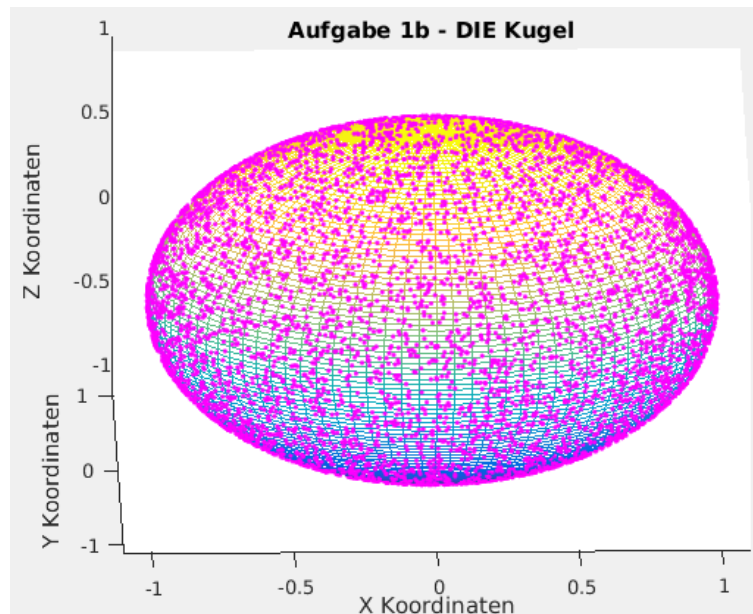
```
1 % create a sphere
2 [X,Y,Z] = sphere(100);
3 x = [X(:); X(:); X(:)];
4 y = [Y(:); Y(:); Y(:)];
5 z = [Z(:); Z(:); Z(:)];
6
7 % plot
8 figure('NumberTitle','off','Name','DIE Kugel');
9 hold on
10 mesh(0.99*x,0.99*y,0.99*z)
11
12 title('Aufgabe 1b - DIE Kugel');
13 xlabel('X Koordinaten');
14 ylabel('Y Koordinaten');
15 zlabel('Z Koordinaten');
16 axis([-1.1 1.1 -1.1 1.1]);
17
18 % 10000 random vectors
19 rv = random_vec(10000);
20
21 % plot
22 scatter3(rv(:,1), rv(:,2), rv(:,3), 'r', 'm');
```

Hilfsdatei: *random_vec.m*

```
1 function [rv] = random_vec(n)
2 rng(0, 'twister');
3 rv = [];
4 for i = 1:n
5     varX = 2*pi*rand();
6     varY = acos(2*rand()-1);
7     x = cos(varX) * sin(varY);
8     y = sin(varX) * sin(varY);
9     z = 1 * cos(varY);
10    vec = [x y z];
11    rv = vertcat(rv,vec);
12 end
```

1.2.2 Bilder





1.3 1 c) Histogramm für Boolesche Funktionen mit 0, 1

1.3.1 Code

```

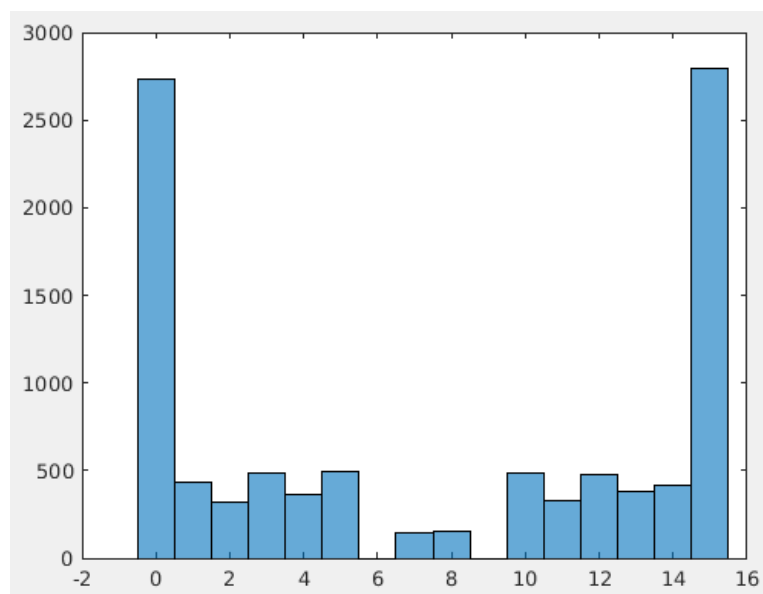
1 classifications1 = [];
2 for i = 1:10000
3     v = rv(i,:);
4     classifications1 = vertcat(classifications1, classify_vec(v, 0)); % ←
5     classify v using 0, 1 as boolean values
6 end
7 % plot
8 figure('NumberTitle','off','Name','Histogram of Boolean Function Frequencies←
9 ');
10 histogram(classifications1)
11 tabulate(classifications1)
12
13 frequencies = tabulate(classifications1);
14 maxfreq     = max(frequencies(:,2))
15 minfreq     = min(frequencies(:,2))
16 relation    = maxfreq / minfreq

```

1.3.2 Resultate

```
1 Value      Count      Percent
2 0         2730      27.30%
3 1         429       4.29%
4 2         318       3.18%
5 3         488       4.88%
6 4         362       3.62%
7 5         491       4.91%
8 7         148       1.48%
9 8         149       1.49%
10 10        485       4.85%
11 11        331       3.31%
12 12        479       4.79%
13 13        384       3.84%
14 14        413       4.13%
15 15       2793      27.93%
16
17 maxfreq = 2793
18
19 minfreq = 148
20
21 relation = 18.8716
```

Histogramm:



1.4 1 d) Histogramm für Boolesche Funktionen mit -1, 1

1.4.1 Code

```
1 classifications2 = [];  
2 for i = 1:10000  
3     v = rv(i,:);  
4     classifications2 = vertcat(classifications2, classify_vec(v, -1)); % ←  
5     classify v using -1, 1 as boolean values  
6 end  
7 % plot  
8 figure('NumberTitle','off','Name','Histogram of -1, 1 Boolean Frequencies');  
9 histogram(classifications2)  
10  
11 tabulate(classifications2)  
12  
13 frequencies = tabulate(classifications2);  
14 maxfreq     = max(frequencies(:,2))  
15 minfreq     = min(frequencies(:,2))  
16 relation    = maxfreq / minfreq
```

1.4.2 Resultate

1	Value	Count	Percent
2	0	1101	11.01%
3	1	442	4.42%
4	2	417	4.17%
5	3	1067	10.67%
6	4	381	3.81%
7	5	1115	11.15%
8	7	443	4.43%
9	8	450	4.50%
10	10	1047	10.47%
11	11	442	4.42%
12	12	1119	11.19%
13	13	427	4.27%
14	14	450	4.50%
15	15	1099	10.99%
16			
17	maxfreq = 1119		
18			
19	minfreq = 381		
20			
21	relation = 2.9370		

Histogramm:

