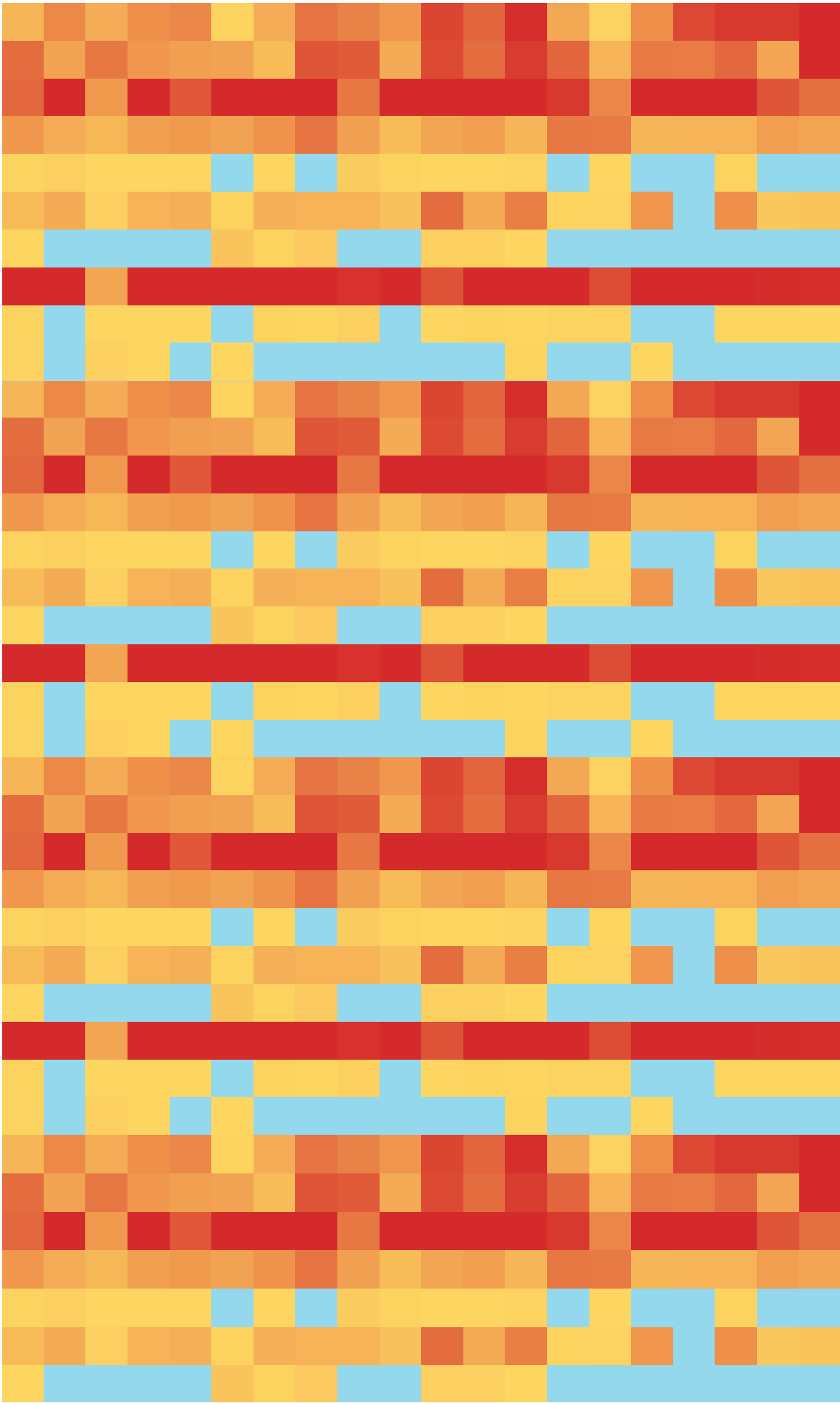


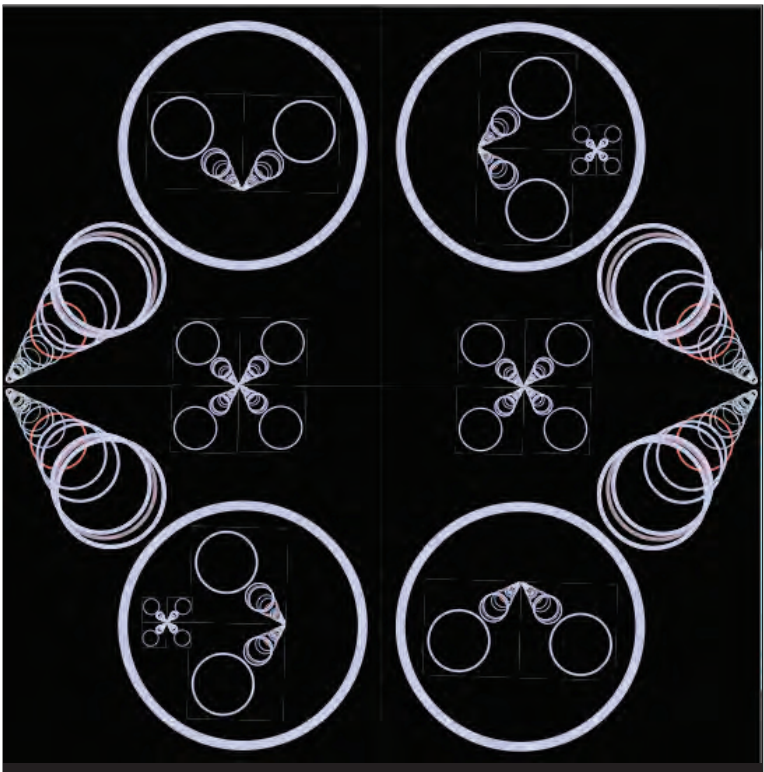
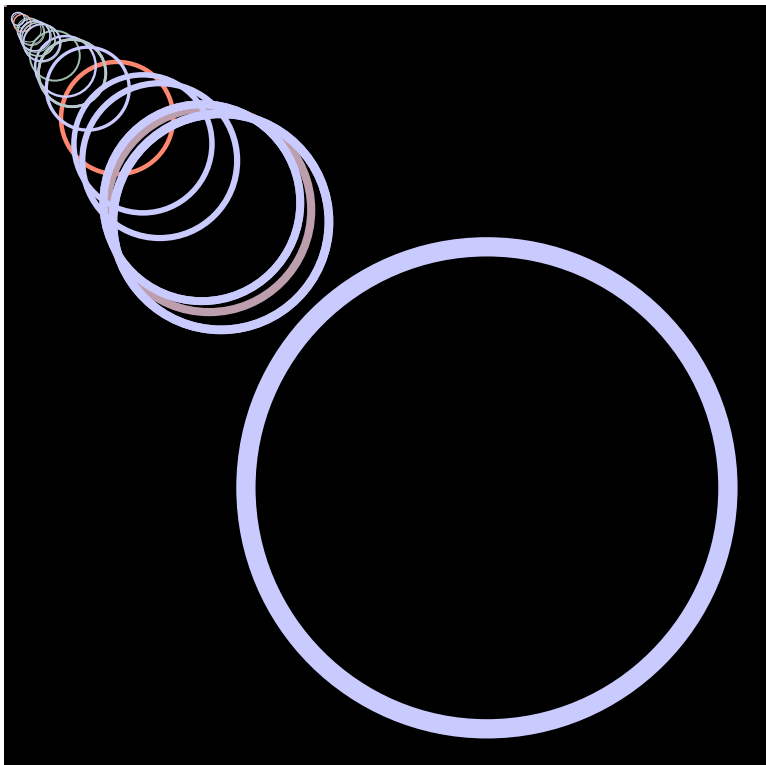






```
project_3
1
2 import processing.pdf.*;
3
4 Table table;
5
6 void setup() {
7   size(500, 1000, PDF, "heatmap.pdf");
8   size(500, 1000);
9   table = loadTable("dataForFNAR.csv", "header");
10
11   int numCol = table.getColumnCount(); // 11
12   int numRows = table.getRowCount(); // 21
13
14   int[][] arr = new int[numRows][numCol];
15
16   // fill array with values from table
17   for (int i = 0; i < numRows - 1; i++) {
18     for (int j = 1; j < numCol; j++) {
19       arr[i][j] = table.getInt(i, j);
20     }
21   }
22
23   // set base colors
24   color low = color(138, 229, 247);
25   color mid = color(255, 218, 99);
26   color high = color(211, 44, 44);
27
28   noStroke();
29
30   // initial x and y positions of rectangle
31   int xPos = 0;
32   int yPos = 0;
33   color c;
34
35   // draw heatmap
36   for (int i = 0; i < numRows - 1; i++) {
37     for (int j = 1; j < numCol; j++) {
38
39       // pick color
40       color low_mid = lerpColor(low, mid, arr[i][j]);
41       color mid_high = lerpColor(mid, high, arr[i][j]);
42       c = lerpColor(low_mid, mid_high, 0.016 * arr[i][j]);
43
44       fill(c);
45       rect(xPos, yPos, 50, 50);
46       xPos += 50;
47     }
48     xPos = 0;
49     yPos += 50;
50   }
51 }
52
53
54 void draw() {
55   exit();
56 }
57
58
59
```

date	FB_rec	groupme_rec	text_rec	email_rec	phone_rec	FB_sent	groupme_se	text_sent	email_sent	phone_sent
10/19/16	119	64	38	19	0	8	0	61	1	0
10/20/16	58	19	48	22	0	7	0	62	1	0
10/21/16	58	41	72	14	2	27	0	82	1	0
10/22/16	53	34	90	14	0	0	0	109	0	0
10/23/16	27	35	114	13	0	24	0	151	0	1
10/24/16	3	14	30	35	1	3	0	51	3	0
10/25/16	18	42	58	36	0	2	0	106	3	0
10/26/16	61	57	99	13	3	33	1	120	1	2
10/27/16	42	39	80	22	1	17	4	121	2	0
10/28/16	54	52	77	19	1	39	4	49	1	0
10/29/16	24	17	72	11	2	9	0	85	0	0
10/30/16	32	46	36	21	5	14	0	60	4	0
10/31/16	37	48	144	37	0	14	6	192	1	0
11/1/16	16	11	74	26	1	15	2	83	2	0
11/2/16	2	20	65	20	0	2	8	111	0	1
11/3/16	30	21	47	23	1	15	0	69	1	0
11/4/16	27	24	69	21	1	14	0	91	1	1
11/5/16	16	36	23	12	1	4	0	19	1	4
11/6/16	29	20	69	16	4	17	0	84	0	0
11/7/16	13	39	41	24	2	11	1	68	2	3



```
sketch_222261103b
Table table;

import processing.pdf.*;

void setup(){
  table = loadTable("data.csv", "header");
  size(1100,1100, PDF, "sketch_222261103b.pdf");
  background(0);
}

void draw(){
  stroke(0);
  noFill();
  for(int i=0; i<table.getRowCount();i++){
    TableRow row = table.getRow(i);

    float restaurants = row.getFloat("restaurants");
    float entertainment = row.getFloat("entertainment");
    float gasoline = row.getFloat("services");
    float merchandise = row.getFloat("merchandise");
    float other = row.getFloat("other");

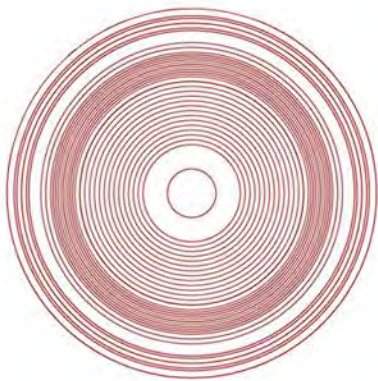
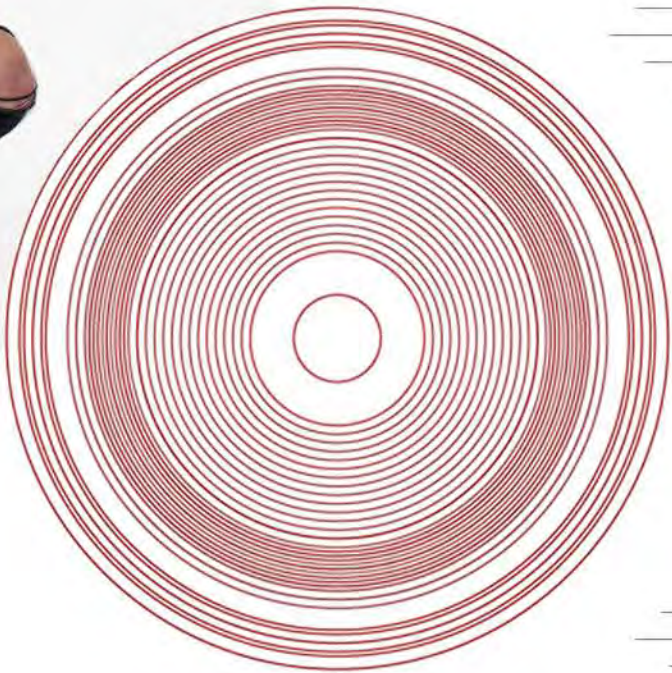
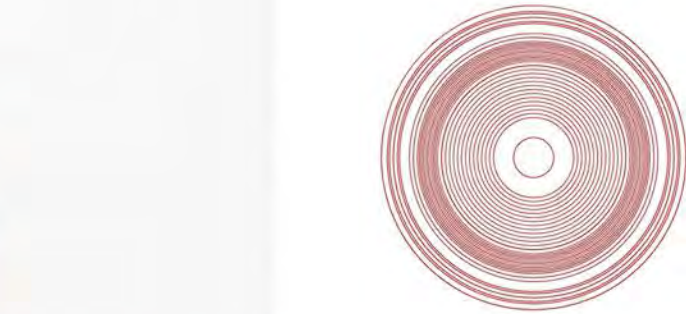
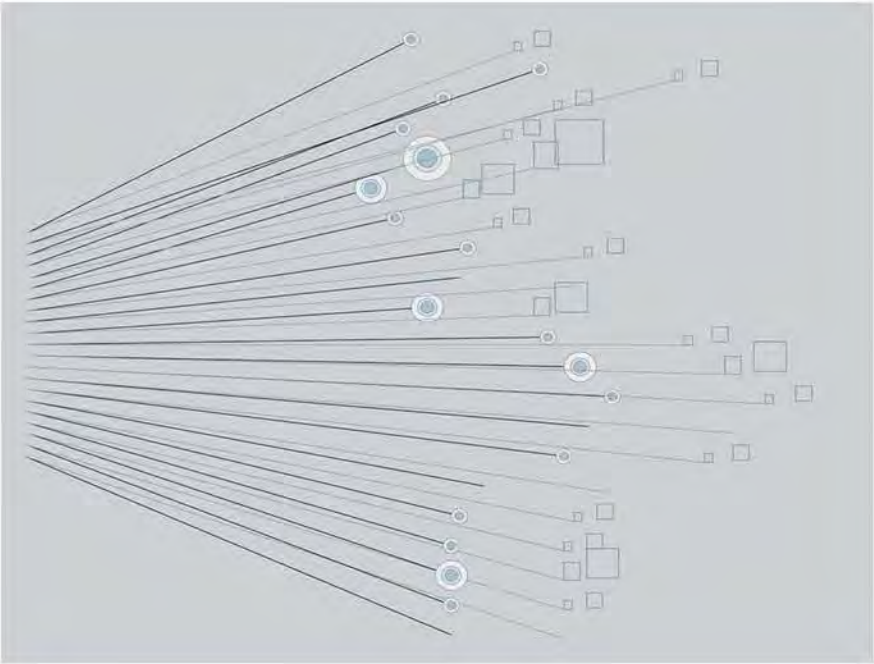
    stroke(#B7C9CB);
    strokeWeight(restaurants/5);
    ellipse(restaurants*5,restaurants*5, restaurants*5, restaurants*5);
    stroke(#9EBCAB);
    strokeWeight(entertainment/5);
    ellipse(entertainment*5,entertainment*5, entertainment*5, entertainment*5);
    stroke(#BC9EAC);
    strokeWeight(gasoline/5);
    ellipse(gasoline*5,gasoline*5, gasoline*5, gasoline*5);
    stroke(#C9CBFF);
    strokeWeight(merchandise/5);
    ellipse(merchandise*5,merchandise*5, merchandise*5, merchandise*5);
    stroke(#FF876F);
    strokeWeight(other/5);
    ellipse(other*5,other*5, other*5, other*5);
  }
  exit();
}
```

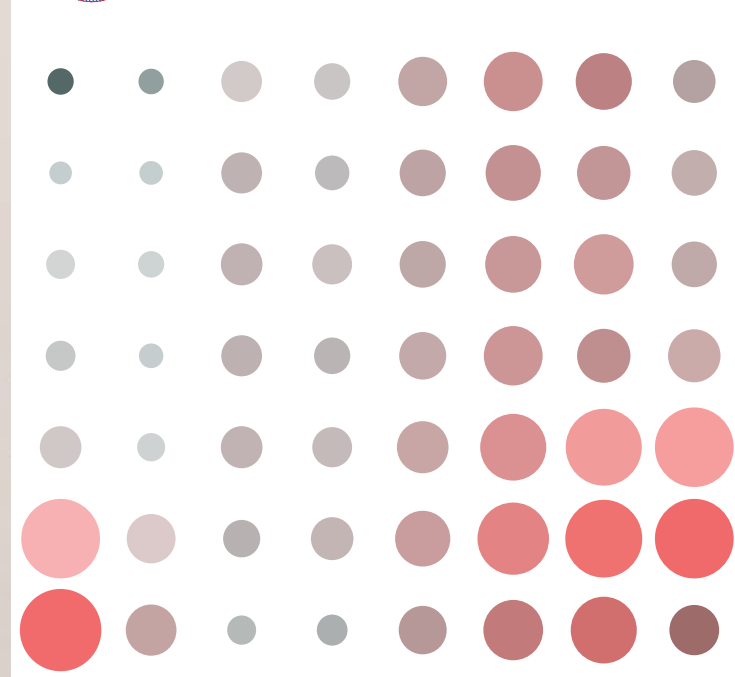
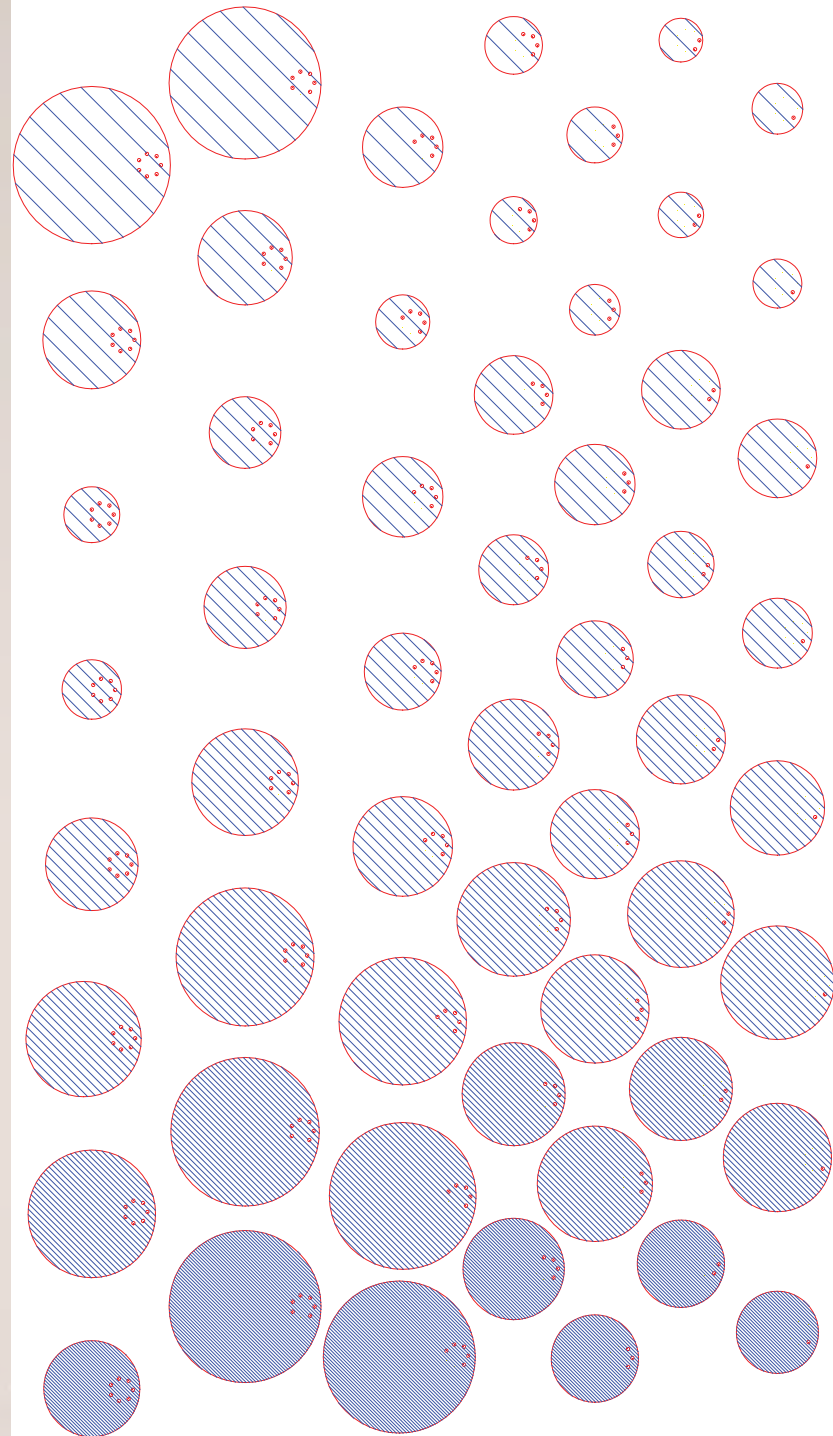
restaurants	entertainment	gasoline	services	merchandise	other
5	7.44	3.24	5	19.44	32.44
10.78	14.42	2.64	5	17.5	5
19.44	9.67	0	59.02	23.95	0
7.63	8.93	0	0	10.59	0
3.74	0	0	0	39.9	0
4.27	0	0	0	139.32	0
6.97	0	0	0	44.75	0
9.67	0	0	0	3.78	0
0	0	0	0	8.66	0
0	0	0	0	62.44	0
0	0	0	0	56.91	0
0	0	0	0	8.66	0
0	0	0	0	62.44	0
0	0	0	0	56.91	0


```
final_data
1 Table table;
2
3 void setup() {
4   table=loadTable('data.csv', 'header');
5   background(215,217,219);
6   size(1800,1800);
7
8   void draw(){
9     for (int i=0; i<table.getRowCount(); i++){
10      TableRow row = table.getRow(i);
11
12      int date = row.getInt("Date");
13      int temperature = row.getInt("Temperature");
14      int coffee = row.getInt("Coffee");
15
16      fill(181,196,208);
17      noFill();
18      stroke(93,108,121);
19      line (date+40, temperature * 10, 680 +15 * i, 30 + coffee * 3);
20      stroke (178,184,189);
21      strokeWeight(1);
22      fill(234,235,238);
23      ellipse (date +40, temperature * 10, coffee * 20, coffee * 20);
24      fill(222, 225, 235);
25      ellipse (date + 40, temperature * 10, coffee +12, coffee * 12);
26      fill(190,206,218);
27      ellipse (date +40, temperature * 10, coffee * 8, coffee * 8);
28      noFill();
29      stroke(158,172,183);
30      rect (date +39.7, temperature * 13, coffee * 20, coffee * 20);
31      rect (date * 39.5, temperature * 12.5, coffee +12, coffee * 10);
32      stroke (178,184,189);
33      line ( 680 +15 * i, 30 + coffee * 3, date * 39.5, temperature * 12.5);
34    }
35  }
```

```
shoulders
1 Table table;
2
3 void setup() {
4   table=loadTable('data.csv', 'header');
5   background(255);
6   size(800,800);
7
8   void draw(){
9     for (int i=0; i<table.getRowCount(); i++){
10      TableRow row = table.getRow(i);
11
12      int date = row.getInt("Date");
13      int temperature = row.getInt("Temperature");
14      int coffee = row.getInt("Coffee");
15
16      noFill();
17      stroke(131,33,33);
18
19      ellipse(300, 300, temperature * 4, temperature * 4);
20      stroke(135,65,65);
21      ellipse(300, 300, date * 8, date * 8);
22      stroke(164,78,78);
23      ellipse (300, 300, coffee +40, coffee * 40);
24    }
25  }
```

Date	Temp	Coffee
11	56	1
12	56	1
13	56	2
14	56	1
15	57	3
16	60	1
17	70	1
18	73	2
19	76	1
20	72	2
21	68	1
22	53	2
23	57	1
24	58	1
25	49	1
26	46	2
27	53	3
28	50	1
29	55	1
30	67	2
31	51	1





```
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40
Table table;

void setup() {
  size(1275, 1275);
  background(255);
  table=loadTable("crashes.csv", "header");
}

void draw() {
  float a;
  a = 3.6;
  for(int i = 0; i < table.getRowCount(); i++) {
    TableRow row = table.getRow(i);
    int sun = row.getInt("Sunday");
    int mon = row.getInt("Monday");
    int tue = row.getInt("Tuesday");
    int wed = row.getInt("Wednesday");
    int thu = row.getInt("Thursday");
    int fri = row.getInt("Friday");
    int sat = row.getInt("Saturday");

    stroke(0, (291-i*36));
    fill(mon/a, 100, 100, sun/a);
    ellipse(100, 50+100*i, mon/10, mon/10);
    fill(tue/a, 100, 100, mon/a);
    ellipse(200, 50+100*i, tue/10, tue/10);
    fill(wed/a, 100, 100, tue/a);
    ellipse(300, 50+100*i, wed/10, wed/10);
    fill(thu/a, 100, 100, wed/a);
    ellipse(400, 50+100*i, thu/10, thu/10);
    fill(fri/a, 100, 100, thu/a);
    ellipse(500, 50+100*i, fri/10, fri/10);
    fill(sat/a, 100, 100, fri/a);
    ellipse(600, 50+100*i, sat/10, sat/10);
    fill(sun/a, 100, 100, sat/a);
    ellipse(700, 50+100*i, sun/10, sun/10);
  }
  noLoop();
}
```


The image displays four identical triangular diagrams stacked vertically. Each triangle is filled with a complex, dense web of thin, colored lines. The lines are primarily red, green, and blue, with some yellow lines interspersed. These lines connect various points distributed across the interior of the triangles, creating a highly interconnected network. The overall effect is a visual representation of a complex system or data structure, possibly a neural network or a social network, where the nodes are the connection points and the edges are the colored lines. The triangles are oriented with their base at the bottom and their apex at the top.

DATA www

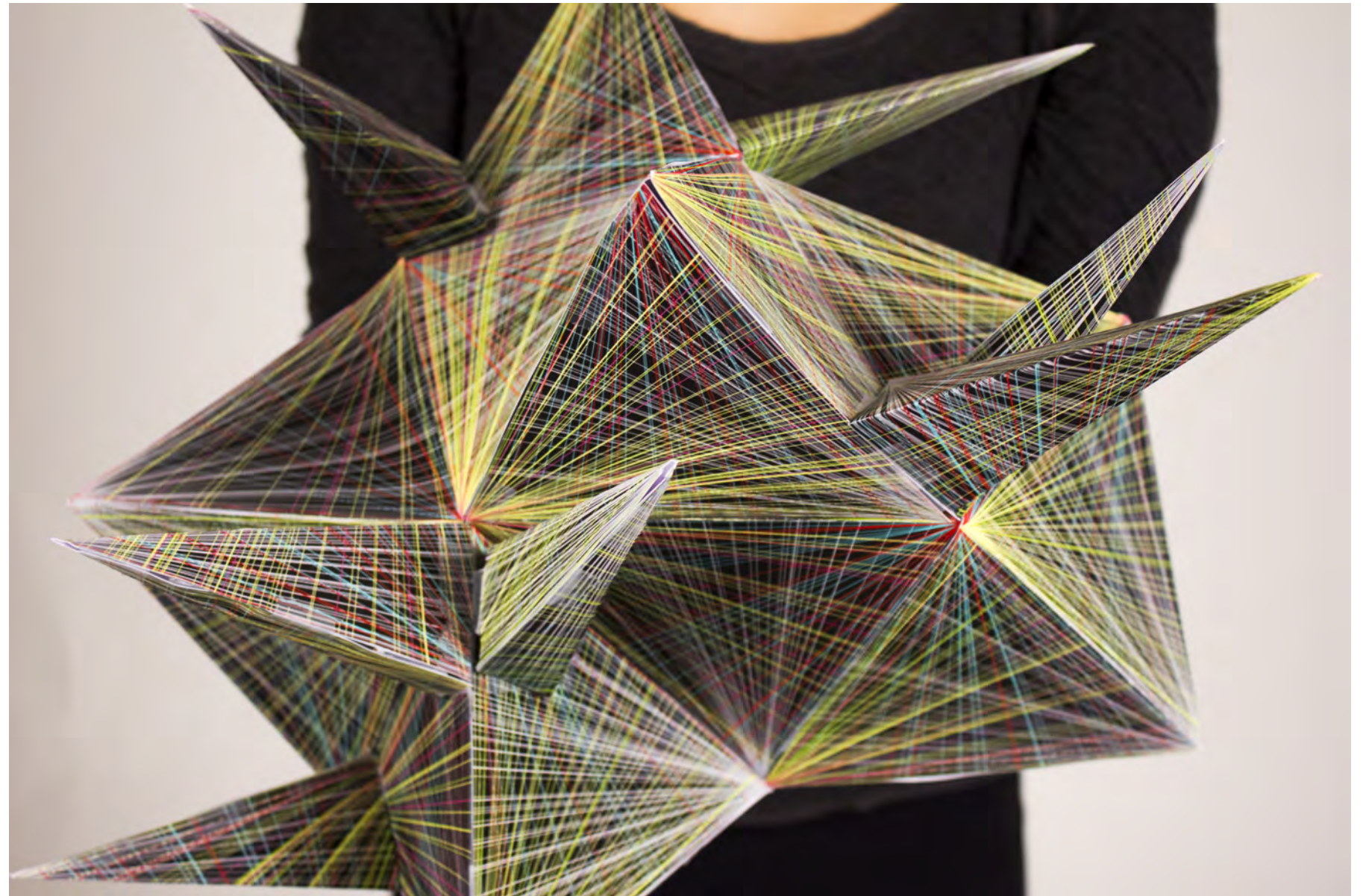
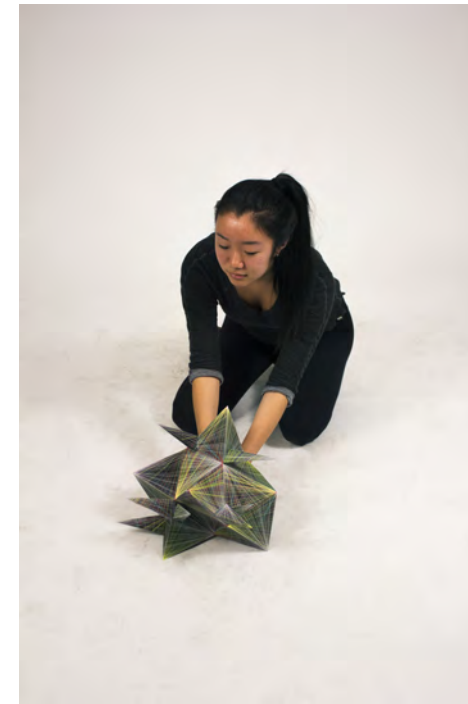
text messages starting 10/22 (sent)

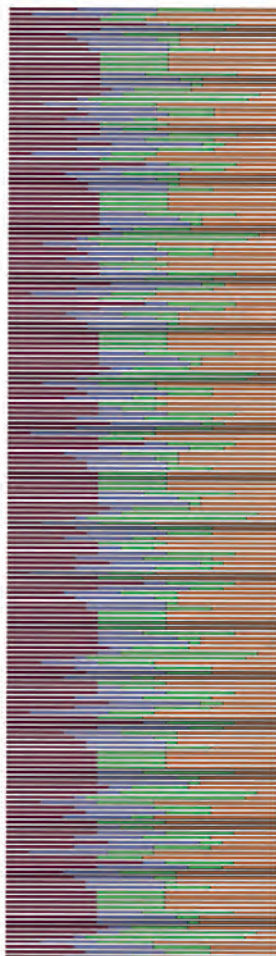
categories:	emotions
- compliments	- mad
- exclamations	- !!!
- questions	- ???
- laughs	- hahaha
- images	- ooo
- noise	- noise

DATE	MAO	!!!	???	hahaha	ooo	noise
10/22 (Saturday)	11	11	11	11	11	11
Total: 61 sent	10	23	11	6	0	11
10/23 (Sunday)	11	11	11	11	11	11
Total: 131 sent	10	24	22	20	0	11
10/24 (Monday)	11	11	11	11	11	11
Total: 141 sent	0	50	25	18	6	11
10/25 (Tuesday)	11	11	11	11	11	11
Total: 182 sent	15	60	0	29	0	11

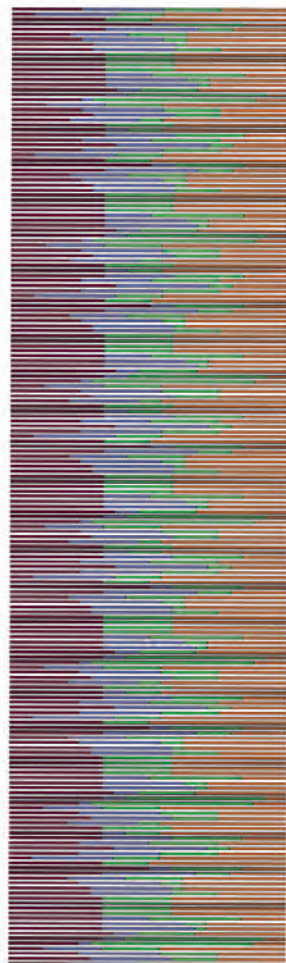
avg 2 min

DATE	MAO	!!!	???	hahaha	ooo	noise
10/26 (Wednesday)	11	11	11	11	11	11
Total: 219 sent	10	24	22	20	0	11





```
data
1
2 void setup() {
3   size(1800, 480);
4   background(255);
5   loadImage("sleep.png");
6   loadImage("classes.png");
7   loadImage("rec.png");
8   loadImage("work.png");
9
10  float x2_start = 0;
11  color c1 = color(200, 0, 80);
12  color c2 = color(200, 100, 200);
13  color c3 = color(0, 200, 0);
14  color c4 = color(200, 100, 0);
15
16  for (int i = 0; i < 10; i++) {
17    for (int j = 0; j < 10; j++) {
18      fill(c1);
19      rect(x2_start, 0.0, 5.0, 17 * float.parseFloat(sleep[j]));
20      fill(c2);
21      float y = 17 * float.parseFloat(sleep[j]);
22      rect(x2_start, y, 5.0, 17 * float.parseFloat(classes[j]));
23      fill(c3);
24      y = y + 17 * float.parseFloat(classes[j]);
25      rect(x2_start, y, 5.0, 17 * float.parseFloat(rec[j]));
26      fill(c4);
27      y = y + 17 * float.parseFloat(rec[j]);
28      rect(x2_start, y, 5.0, 17 * float.parseFloat(work[j]));
29      x2_start = x2_start + 10;
30    }
31  }
32 }
```





```
sketch_161108b
1 Table table;
2
3 void setup() {
4   table = loadTable("Phone Calls - 1.csv", "header");
5   background(255);
6   size(1400,700);
7 }
8 void draw() {
9   for(int i=0; i < table.getRowCount(); i++) {
10    TableRow row = table.getRow(i);
11
12    int date = row.getInt("Date");
13    int person = row.getInt("Person");
14    int minutes = row.getInt("Minutes");
15    int place = row.getInt("Place");
16
17    println(date + " " + person + " " + minutes + " " + place);
18    if (person == 1) {
19      fill(13, 255, 178);
20    } else if (person == 3) {
21      fill(255, 193, 232);
22    } else if (person == 2) {
23      fill(0, 72, 255);
24    } else if (person == 5) {
25      fill(255, 174, 0);
26    } else if (person == 6) {
27      fill(255, 51, 0);
28    } else {
29      // Default to this
30      fill(255, 255, 255);
31    }
32    noStroke();
33    rect(date * 5, 0, minutes * 10, 1500);
34  }
35  noLoop();

```

Save canceled.

```
260 6 1 0
261 5 1 0
262 3 2 0

```

