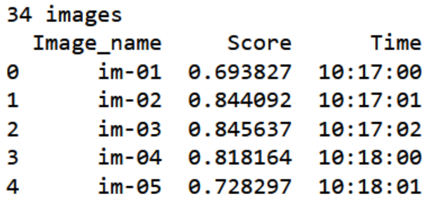
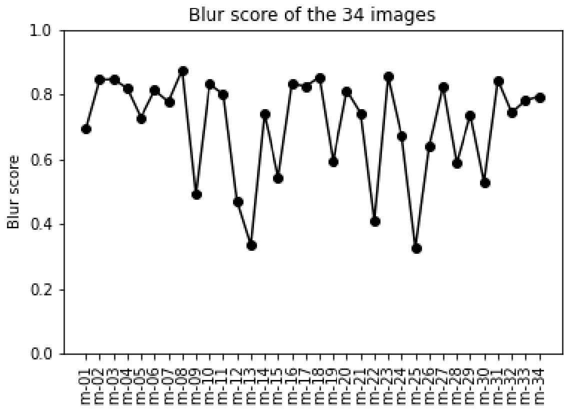
Lab 06: elements of corrections

Q1:

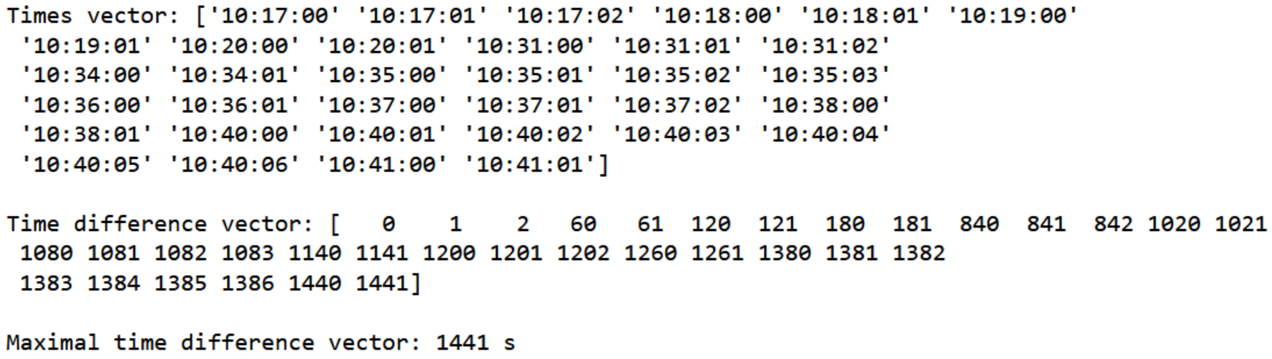


```
34 images
  Image_name     Score      Time
0      im-01  0.693827  10:17:00
1      im-02  0.844092  10:17:01
2      im-03  0.845637  10:17:02
3      im-04  0.818164  10:18:00
4      im-05  0.728297  10:18:01
```

Q2:



Q3:



```
Times vector: ['10:17:00' '10:17:01' '10:17:02' '10:18:00' '10:18:01' '10:19:00'
 '10:19:01' '10:20:00' '10:20:01' '10:31:00' '10:31:01' '10:31:02'
 '10:34:00' '10:34:01' '10:35:00' '10:35:01' '10:35:02' '10:35:03'
 '10:36:00' '10:36:01' '10:37:00' '10:37:01' '10:37:02' '10:38:00'
 '10:38:01' '10:40:00' '10:40:01' '10:40:02' '10:40:03' '10:40:04'
 '10:40:05' '10:40:06' '10:41:00' '10:41:01']

Time difference vector: [   0    1    2   60   61  120  121  180  181  840  841  842 1020 1021
 1080 1081 1082 1083 1140 1141 1200 1201 1202 1260 1261 1380 1381 1382
 1383 1384 1385 1386 1440 1441]

Maximal time difference vector: 1441 s
```

Q4: You should find an initial loss of `0.8969124171515748`

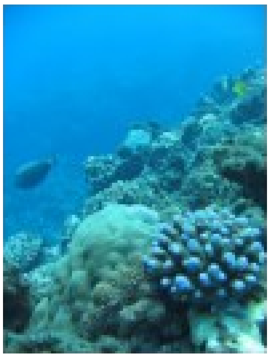
Q5:

Important remark: one of the key elements in simulated annealing is the notion of neighbourhood. At each iteration, you have to compute a neighbour solution. If at each iteration, you just pick up a random solution (that's what you do when you randomly permute all the elements in the x vector), it is not simulated annealing…

Q6:

```
Optimized loss value: 0.504728139829264
```

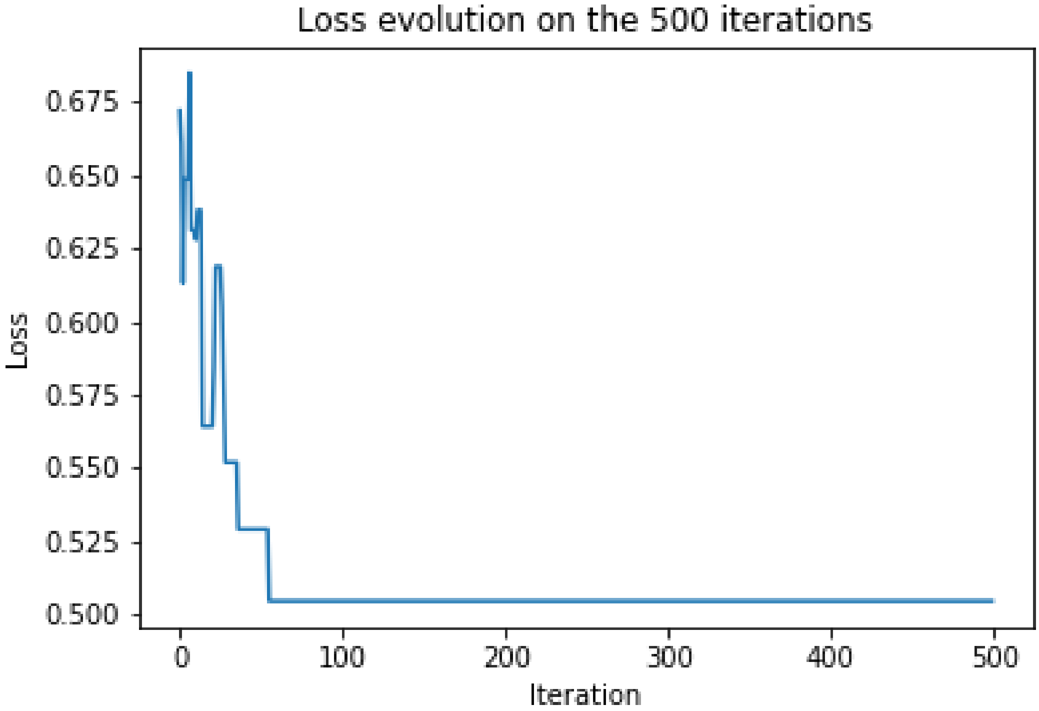
im-09: 0.49



im-13: 0.34

im-25: 0.33

Q7:



Q8:

```
5984.0 possible combinations
Optimized loss value: 0.504728139829264
Selected images: ['im-09', 'im-13', 'im-25']
```

Q9:

```
(1344904.0 possible combinations)
Optimized loss value: 0.6299694895730565
```

im-09: 0.49

im-12: 0.47

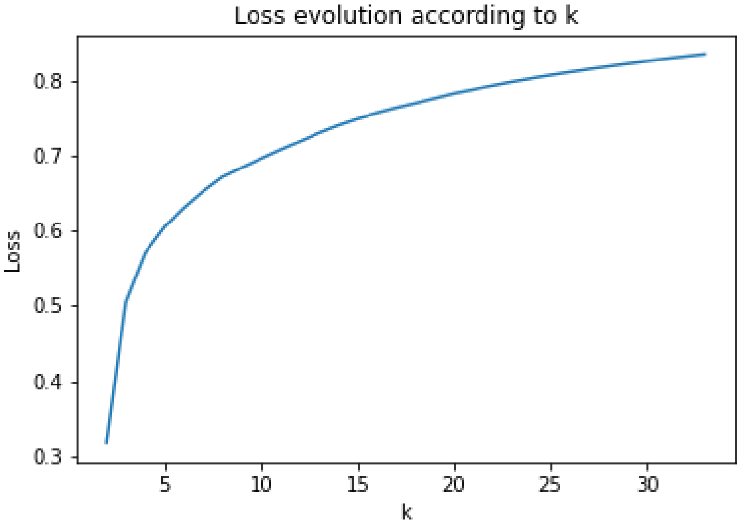
im-13: 0.34

im-22: 0.41

im-25: 0.33

im-30: 0.53

Q10:


Loss evolution according to k
Loss
k