Homework 2; Psych 186B, Winter 2018

Marshall Briggs

31 January 2017

1 Problem 1 (2 points) - problem1.m

1. I generated two vectors, f and g.
2. Since I generated them using randn(), which generates random numbers from a normal distribution, the means of f and g are already 0.
3. I set the lengths equal to one.
4. I multiplied g and f^T to compute A.
5. You can see that Af produces the same output as g because g' = Cos(Θ) = 1.0 radians. The length of g' is also seen to be one.

   The provided code is pretty self-explanatory for this problem.

2 Problem 2 (2 points) - problem2.m

1. I generated a new normalized random vector f'.
2. I compared its orthogonality to f, which I expect to be zero. Both vectors were computed randomly, meaning on average they should have an orthogonality of zero.
3. I computed Af' and found its length, which I would expect to also be zero. It turned out to be very close to zero, in fact it was equal to the orthogonality computed above.

3 Problem 3 (10 points) - problem3.m

1. I generated two arrays of vectors, f and g. Each f_i would be paired with its appropriate g_i.

2. I computed the outer product matrices ...

3. ... and summed them together to build my linear associator A.

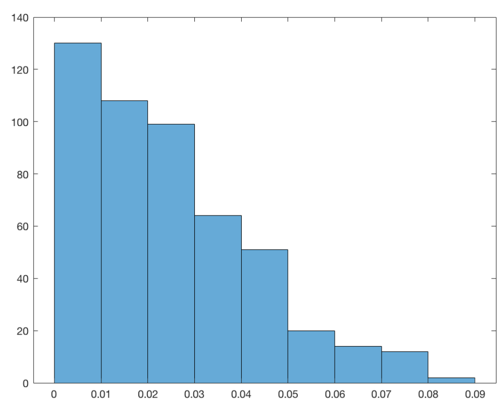
4. I compared the output with what it "should have been" (cosines), then I computed the lengths of the vectors g', then I then tested the selectivity of the system by generating sets of random vectors and seeing the system's response to these vectors as input.

5. Finally, I repeated the previous processes (both output comparison and selectivity testing) on inputs vector sets of different sizes = 1, 20, 40, 60, 80, 100.

   (i) Size: 1

   Output comparison: Avg. output length: 0.019401

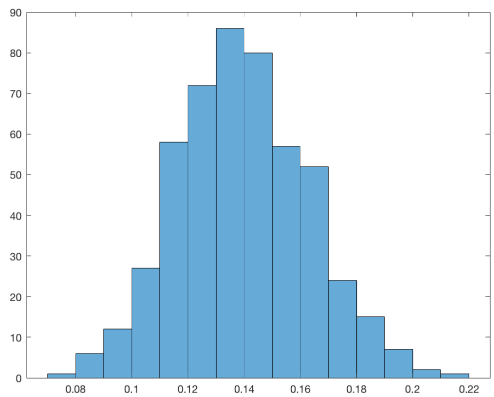
   Selectivity comparison:



(ii) Size: 20

Output comparison: Avg. output length: 0.134580

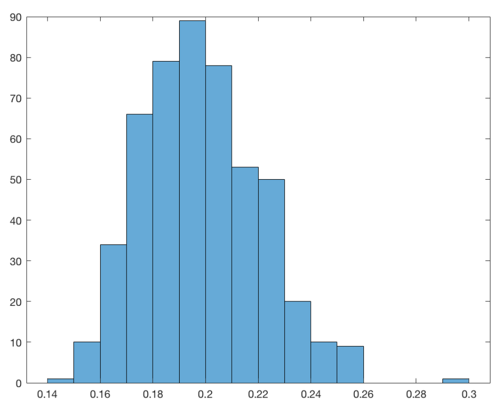
Selectivity comparison:



(iii) Size: 40

Output comparison: Avg. output length: 0.200084

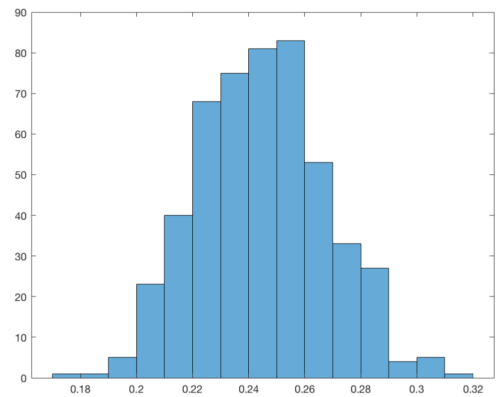
Selectivity comparison:



(iv) Size: 60

Output comparison: Avg. output length: 0.245487

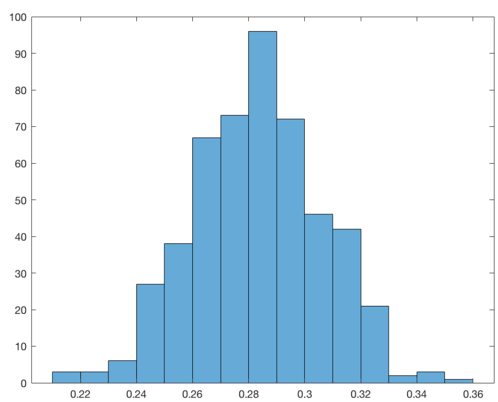
Selectivity comparison:



(v) Size: 80

Output comparison: Avg. output length: 0.280739

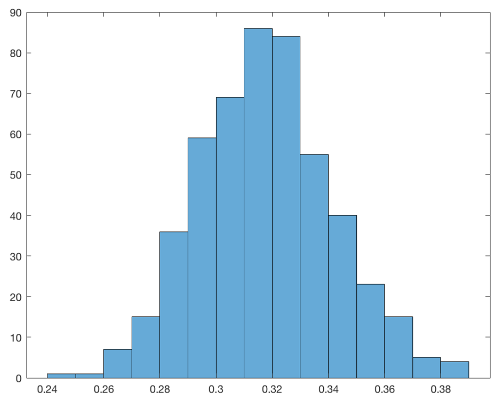
Selectivity comparison:



(vi) Size: 100

Output comparison: Avg. output length: 0.312202

Selectivity comparison:



4 Problem 4 (3 points) - problem4.m

(3) I made chains of associations, using a set f of vectors. Each vector was paired with its following neighbor, such that A was built using vector combinations $f[1]f[2]^T + f[2]f[3]^T + ... + f[n\text{-}1]f[n]^T$.

I then used each original vector as input and tested the average length of the outputs to see the noise of the output set increasing as the number of input vectors increased. Results are shown below.

"Across the chain" noise:
—- Links: 2, Avg. cosine: 1.000000, Avg. length: 1.000000
—- Links: 100, Avg. cosine: 0.998835, Avg. length: 1.047121
—- Links: 500, Avg. cosine: 0.998816, Avg. length: 1.222304
—- Links: 1000, Avg. cosine: 0.998221, Avg. length: 1.410836
—- Links: 5000, Avg. cosine: 1.001326, Avg. length: 2.449740
—- Links: 10000, Avg. cosine: 1.001701, Avg. length: 3.318995

Then, I did as the problem instructed, and used f[i] as input, then used f[i]' as the next input etc., until I had traversed the whole chain.

"Along the chain" noise:
—- Links: 2, Starting input len: 1.000000, Ending output len: 1.000000
—- Links: 100, Starting input len: 1.000000, Ending output len: 1.054125
—- Links: 500, Starting input len: 1.000000, Ending output len: 1.191063
—- Links: 1000, Starting input len: 1.000000, Ending output len: 1.391191
—- Links: 5000, Starting input len: 1.000000, Ending output len: 2.310085
—- Links: 10000, Starting input len: 1.000000, Ending output len: 3.361650