Problem Sets Revision History

The table notes major changes between revisions. Minor changes such as small clarifications or formatting changes are not noted.

Version	Date	Changes	Principal Author(s)
0.4		Initial release	NA
0.5	May 15 2017	Added questions for SDR, Encoder, SP and TM chapters	spurdy

Problem Sets

Sparse Distributed Representations

1. What are the disadvantages to SDRs? Consider representing a string of text as ASCII vs. SDRs.
2. What does the "distributed" part of "Sparse Distributed Representation" mean?
3. Why is sparsity necessary? How sparse do representations need to be? Can they be too sparse?
4. Which is better? n=1000 and w=3 or n=300 and w=15?
5. For representations with 33 active bits out of 500 total, how many unique representations are there? If we randomly generate 100 of these representations, how likely is it that two of the hundred will be identical? If we randomly generate two of these representations, how likely is it that they will share more than 5 active bits?
6. Work out the first few terms of equation (4) in the SDR chapter to see how the values quickly diminish. Show the equation can then be approximated w/Work out the first few terms of equation (4) in the SDR chapter to see how the values quickly diminish. Show the equation can then be approximated $\mathrm{w} fp_w^n(\theta) \approx \frac{|\Omega_x(n,w,\theta)|}{\binom{n}{w}}$/.

Encoders

1. When creating an encoder for a new data type, what makes a good SDR? How can you test that the SDRs that are produced are good?
2. How would you encode sound as SDRs?

Spatial Pooling

1. What is Spatial Pooling? How can we tell if the output of Spatial Pooling is good?
2. What is boosting and how does it work? Is it necessary?
3. Is an untrained spatial pooler just a "random hash" (random mapping from input to output vector)? Why or why not? What happens to the output of the spatial pooler if you change one bit in the input?
4. Can untrained spatial poolers with randomly initialized input bit weights be useful or even better than a trained SP?
5. How does online learning happen in the SP?
6. Can you do spatial pooling with small numbers? For example, is it reasonable to have an SP with 20 columns? If not, why are large numbers important in SDR's?
    a. What's the difference between picking "5 columns out of 50" vs "50 out of 500"? Both have 10% sparsity.
    b. What's the difference between picking "50 out of 100" vs "50 out of 1000"? Both will output 50 1's.

7. Is it possible that the SP output for a fixed input can change completely over time? How can this risk be mitigated?
8. Suppose the input vector (input to the SP) is 10,000 bits long, with 5% of the bits active at a time. What percentage of the input space should be in the potential pool for each column? How do you figure this out?
9. How does the SDR representation of input A in isolation, and input B in isolation, compare with the SDR representation of input A unioned with B? Alternatively, how does the representation of a horizontal line and the representation of a vertical line compare with the representation of a cross?
10. Let's look at a Spatial Pooler of 2048 columns with 40 active at a time and learning enabled. A, B, and C represent non-overlapping subsets of 20 bits in the input space (with total of 500 bits). The spatial pooler is trained on the following:
    - All A and B bits active and all other input bits inactive
    - All A and C bits active and all other input bits inactive
    - Many other patterns with 40 active bits, but none that overlap significantly with A, B, or C

    After this training, we example the output when only A bits are active in the input, only B, and only C. How will the overlap compare between these three output representations?
11. Suppose we have an input vector that is 10,000 bits long. Suppose the spatial pooler has 500 columns, of which 50 are active at any time.
    a. Can we distinguish many patterns, or a small number? Which patterns are likely to be confused?
    b. What happens to the SDR representation if we add noise to the patterns?
    c. What happens if we add occlusions?

Temporal Memory

1. What is a first order sequence memory? What is a high order sequence memory? What is a variable order sequence memory?
2. Why do we need high order sequence memory?
3. How are sequences learned in the Temporal Memory?
4. What happens when there are no internal predictions and a new set of columns become active?
5. Once a sequence is learned, how does the Temporal Memory make predictions?
6. What is the difference between one cell per column vs multiple cells per column (in terms of what sequences you can learn)?
7. Can the Temporal Memory recognize a sequence if it starts in the middle? How does it do so?
8. How long of a sequence can the Temporal Memory learn? If you have 10,000 inputs into the spatial pooler, 500 columns, and a "reasonable" set of TM parameters (choose some), can you learn a sequence that is 1000 elements long? What about 100,000 elements long?
9. Suppose a Temporal Memory has learned the following two sequences: ABCDE and ABCDF. You now present the sequence ABCD - what will be predicted next? What is the representation of the Temporal Memory at this point in time?
10. Suppose a Temporal Memory has only learned the following two sequences: ABCDE and FGCDH. You now present the sequence FGCD - what will be predicted next? Suppose you present the sequence CD - what will be predicted next? What is the exact representation of the Temporal Memory at this point in time (predicted and active cells).
11. What would be the difference between enforcing one distal segment per cell vs allowing multiple distal segments per cell?
12. What are the disadvantages of the temporal memory? When will other prediction techniques work better?