Systems Neuroscience
76901
Solution 1st Assignment

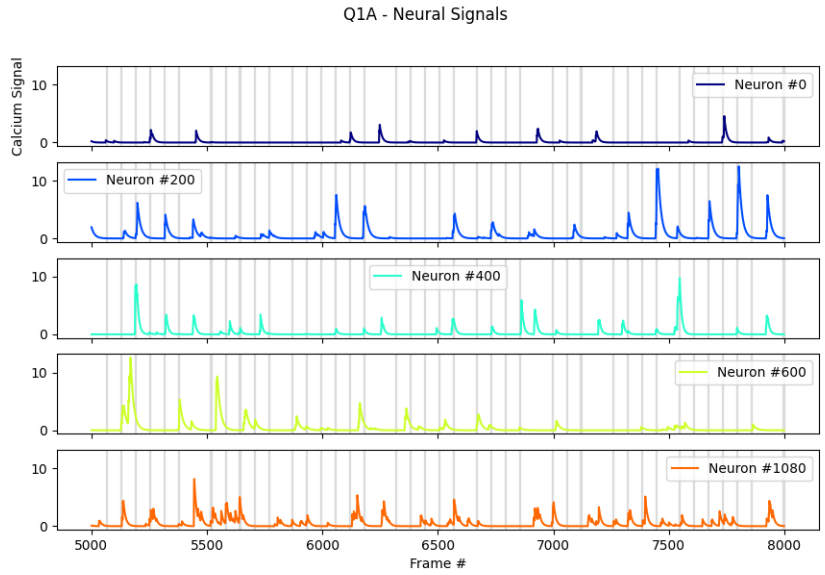
Barak Haim 1

11/02/2024

Contents

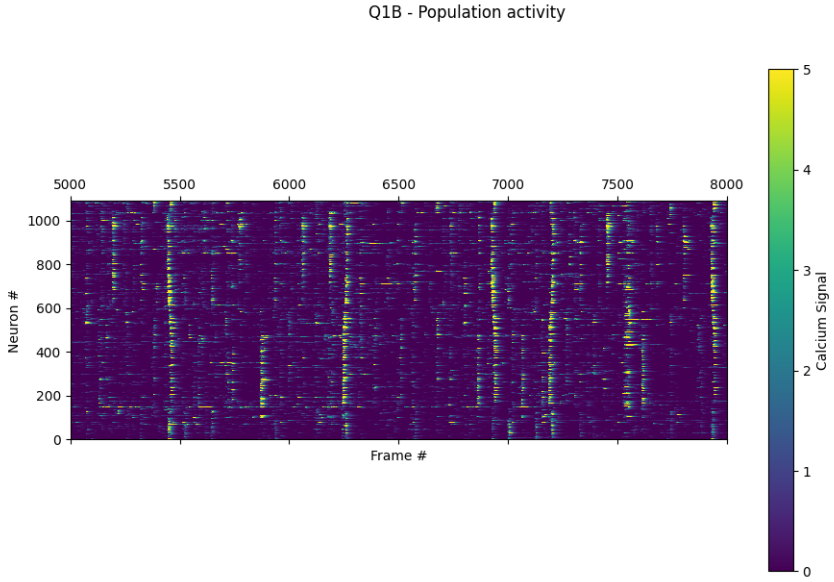
1 Data exploration - visualize the signals and their relation to the stimuli

1.1 1.a



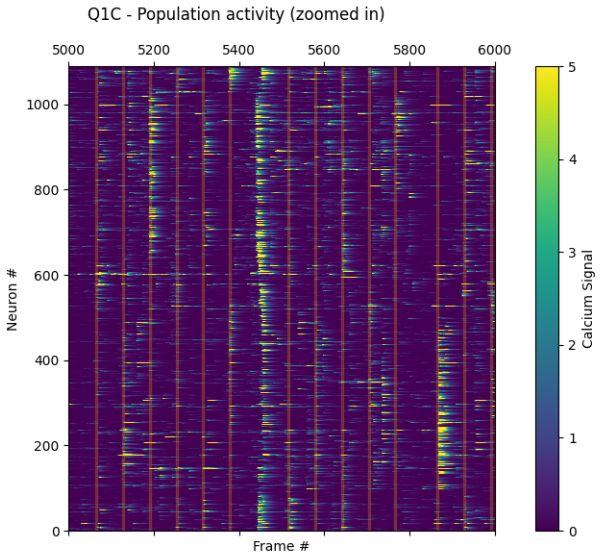
In this figure we see different neurons respond differently to different stimuli.

1.2 1.b



By looking at the whole population we can see how neurons behave in the "big picture". We can se some stimuli cause a large number of neurons to fire together while others are more local.

1.3 1.c



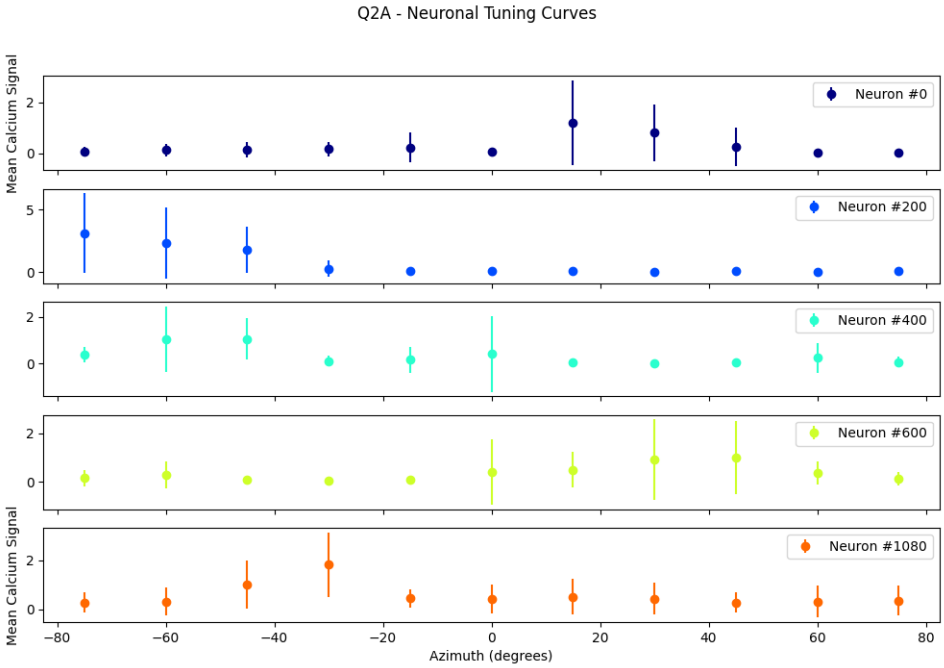
We can see neurons firing well after stimulus ends.

1.4 1.d

See matrix in the code.

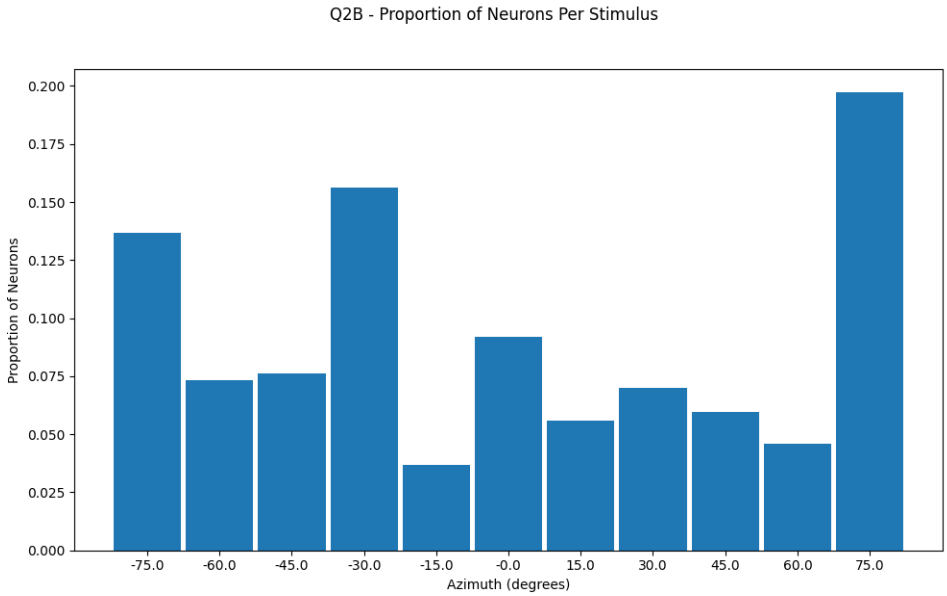
2 Tuning curves

2.1 2.a



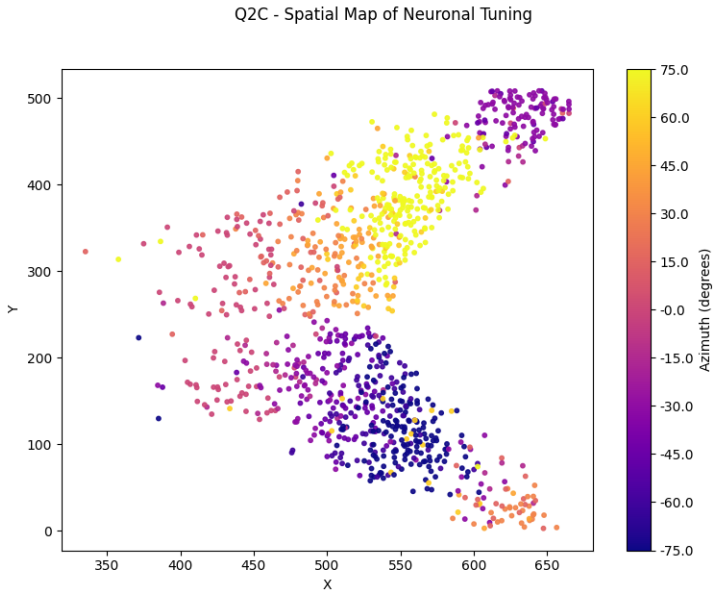
Here we can see 5 exemplar neuronal tuning cures. For some we can see a distinct preferred direction (e.g. - #0 is tuned to around 20 deg, #200 to -75), while others respond to few different directions (e.g.-#400 and #1080). It's important to note some firing rates are extremely variable. In conclusion, we can see here the system is more complex than simply a neuron per direction.

2.2 2.b



It looks generally the fish is more sensitive to stimuli for the far ends of its field of view. Maybe this is enables it more sensitivity to stimuli (predators or prey) which have a small chance of being detected.

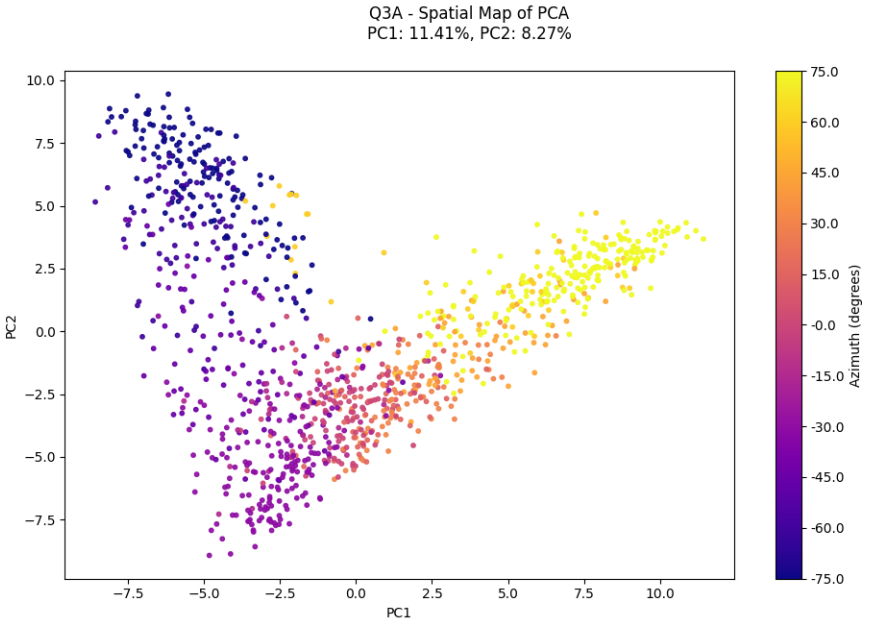
2.3 2.c



We can see clusters of neurons with similar turning cures. Maybe this is due to Hebbian learning - neurons which fire together wire together - keeping them close by is more efficient energetically and keeps latency at it's minimum.

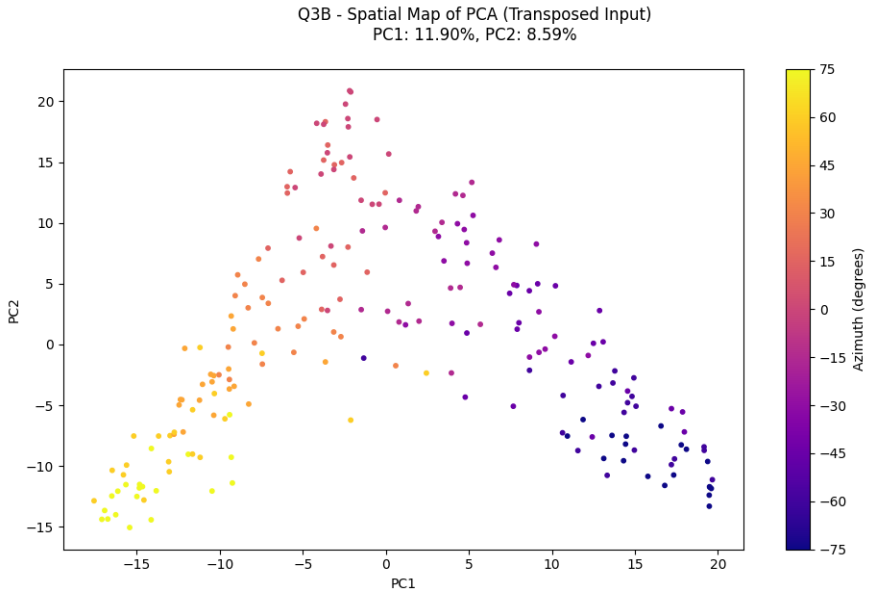
3 Correlation in the data and dimensionality reduction

3.1 3.a



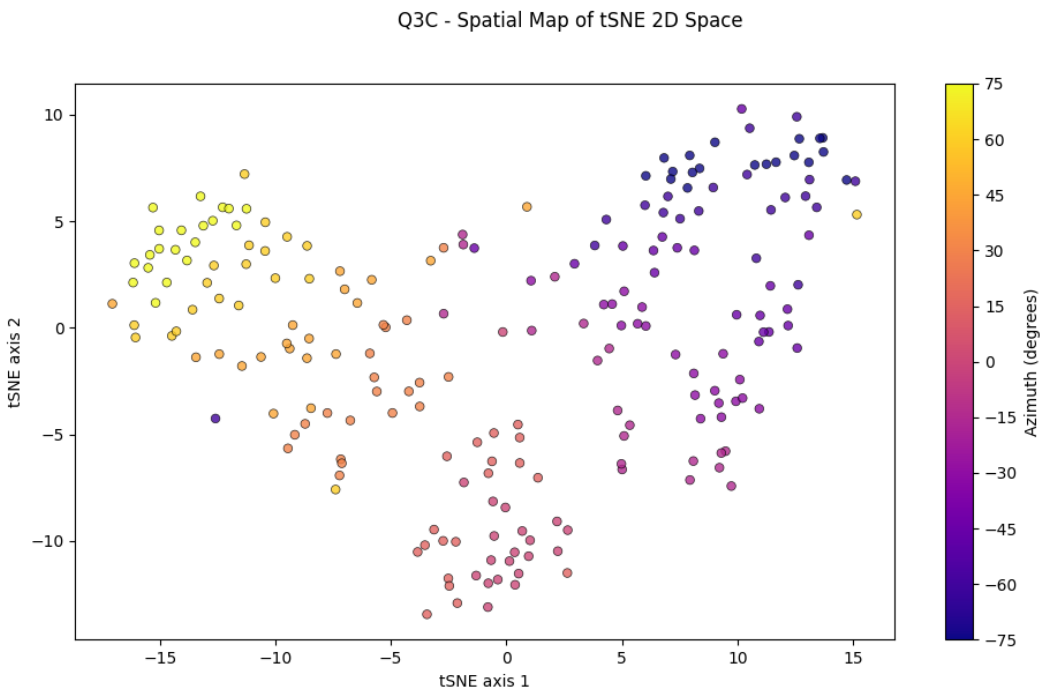
PC1 represents the dimension with greatest variance in the data. PC2 is the second most variable after PC1 and orthogonal to it. In our case, we have matrix A with 1089 rows of neurons and for each 215 stimuli features. PCs 1 and 2 represent the combination of stimuli which are the most crucial to understand how the neurons behave with regards to stimulation. We can see a "V" shape similar to the spatial data. Also, we can see clusters by color (i.e. preferred direction). We can also see how the order of the clusters match the colorbar along the "V" shape.

3.2 3.b



Here we have the transposed matrix A from the previous question, i.e. rows of stimuli and columns of neurons. This is a representation of the different stimuli by variance in the neuronal space. Again, we see the distinct "V" shape and how proximity in the neuronal 2D PC space is correlated with closeness to stimuli type.

3.3 3.c



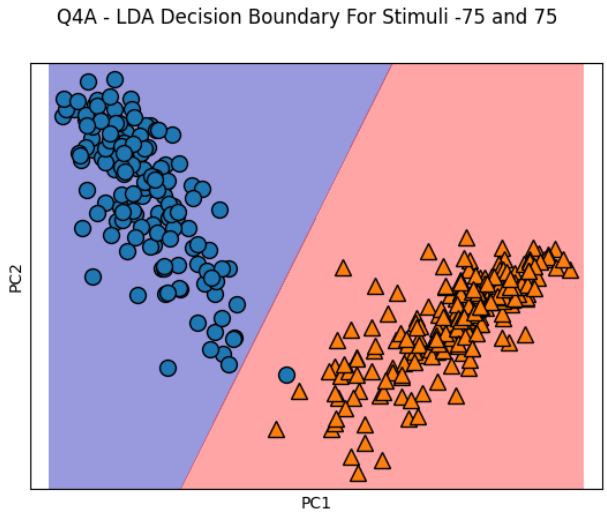
Here we see the data spanned on an arc. Again, proximity in the 2D tNSE space is correlated with closeness to stimuli type. All in all, pretty similar to 3B. In general, tNSE is more robust to outliers, is more suitable for non-linearly separable data and better at preserving distances. In the general case, I hypothesize tNSE would be better for non-linearly separable data.

3.4 3.d

It seems like a single neuron may have "preferences" with regards to stimulus, however, in order to understand how stimuli are represented, we must look at the network level.

4 Decoding the presented stimulus using neural activity only

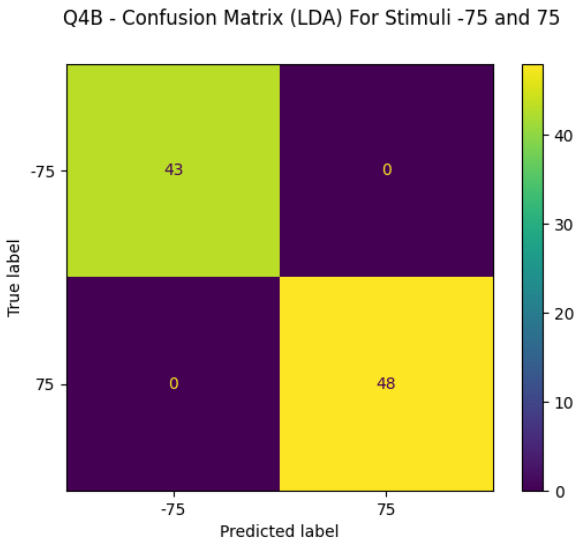
4.1 4.a



LDA assumes there's a hyper-plane which separates 2 distinct clusters (around which are centered around their respective means and have identical covariances). LDA then finds the hyper-plane (as seen in the figure) using the labels and data given to it. We can then measure how well the model did by asking the question how many misclassifications are there regarding the test dataset. The so called "test-set" was not seen by the model during its training and so should give an objective estimate to "real" inputs. We can look at the confusion matrix in the next question.

Moreover, we can ask whats the rate of true labels (True Positives AKA TP) out of those who are real true (TP + False Negatives AKA FP). Also, we can ask how many missed classification were there or what is their rate. This is computed in subsection "c".

4.2 4.b

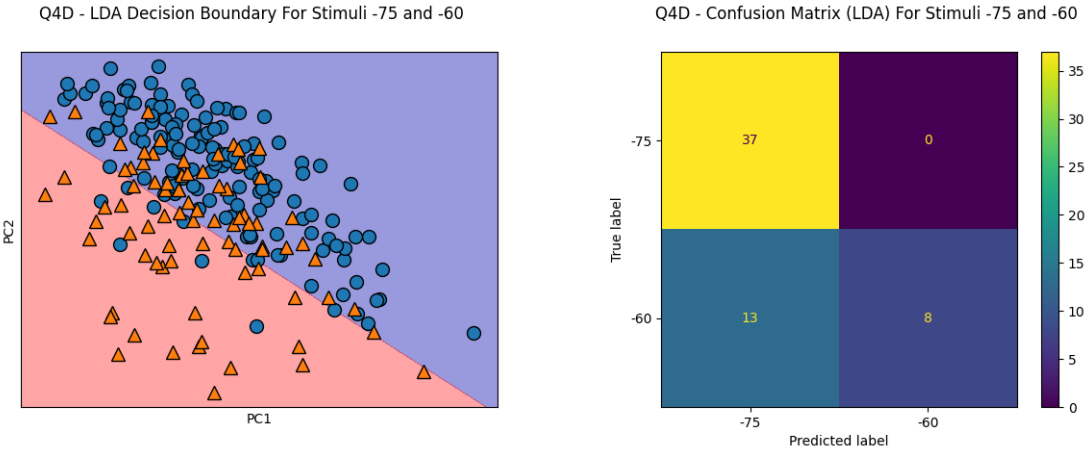


We see there are no misclassifications with regards to this train-set. We can see one misclassification in the train-set above. Overall this is a pretty good classification though still, it doesn't say much about the general training error of all possible data.

4.3 4.c

Both accuracy and sensitivity are 100%. This means the classification is the best we can hope for under these conditions and this test-set.

4.4 4.d



In this case, the accuracy rate is 78% and sensitivity rate is 38%. We can also see by the confusion matrix and the boundary line that a large number of the "-75" neurons

are classified as "-60". This is not very impressive.... We can see the performance of LDA is not necessarily good or bad but depends on the data. Here it seems the data is not linearly separable and the model is not robust enough in order to handle it.

4.5 4.e

LDA assumes the data is:

- Multivariate normality: Independent variables are normally distributed.
- Covariances among group variables are the same across levels of predictors.
- Multi-collinear.
- i.i.d.

In our case, it seems the data isn't necessarily normally distributed or, maybe not i.i.d.

4.6 4.f

We can use a random forest in order to deal with this classification problem. First, i'd use PCA again on matrix A, this time not limiting it to just 2 dimensions. Then, I'd feed the data as a to the random-forest classifier in order to classify according to different stimuli.