group 4.

Liquid state machines

1. Description of the problem

1.1. Brief description of the problem and its context (industry, environment, etc.)

What liquid state machine is?

- LSM is a machine learning model or system that is part of a series of particular neural network models
- these models build on traditional designs to introduce new and innovative ways of processing information
- neural networks that is based around the neurobiology of the human brain

Liquid state machine is a type of reservoir computer that uses a spiking neural network.
Reservoir computing is a framework for computation derived from recurrent neural network theory that maps input signals into higher dimensional computational spaces through the dynamics of a fixed, nonlinear system called a reservoir.
Recurrent neural network is a class of artificial neural networks where connections between nodes form a directed or undirected graph along a temporal sequence.
Spiking neural networks are artificial neural networks that more closely mimic natural neural networks. In addition to neuronal and synaptic state, SNNs incorporate the concept of time into their operating model.

1.2. Who can benefit from this model, and how?

Our work is mainly theoretical. Even though liquid state machines may be very useful and very important tools, our model will predict simple letters/numbers so it's not gonna be beneficial for anyone but us. Of course undeniable benefit is our progress and knowledge we will get from the process, but basically it's just art for art’s sake.

1.3. Why does the problem seem interesting to you?

This problem creates some interest for our team because of its nontriviality and flexibility. We believe that it can be applied to different varieties of topics, starting from robot control and ending with speech recognition. It general we don’t really have any knowledge about LSM and their use so every new piece of information was interesting.

2. Data

2.1. Sources of data, assessment of its reliability

Data is generated by LSM spiking neural network program created by our professor Karol Chlasta.
https://github.com/KarolChlasta/rnap/tree/main/dat/raw/ex2022.04.24-RetNet40-25PUM

2.2. Brief descriptive analysis of the data

Data consist of placement of the neuron that was spiking (we can say values of x, y, z), time at which it was spiking and a number/letter that was shown.

2.3. Rationale: how can this data help solve the problem?

After the solving the problem we will potentially be able to see the which of the 3 characters was chosen by liquid state machine(LSM)

3. Solution

3.1. Brief description of the chosen model with justification

k-nearest neighbors algorithm
In statistics, the k-nearest neighbors algorithm (k-NN) is a non-parametric supervised learning method used for classification and regression. In both cases, the input consists of the k closest training examples in a data set. The output depends on whether k-NN is used for classification or regression:

- In k-NN classification, the output is a class membership. An object is classified by a plurality vote of its neighbors, with the object being assigned to the class most common among its k nearest neighbors (k is a positive integer, typically small). If k = 1, then the object is simply assigned to the class of that single nearest neighbor.
- In k-NN regression, the output is the property value for the object. This value is the average of the values of k nearest neighbors.

3.2. Stages of project implementation

1. First we combined all data given to us by professor to one file
2. Then using dataiku we separated particular columns and cleaned the data
3. After first modeling the accuracy was 0,463
4. Then there was second modeling, that included some modifications, removing some columns and…
5. It resulted in accuracy increasing to 0,963

3.3. Measures of evaluation (quality assessment) of the model

After working on our project, accuracy increased significantly. It was the result of modeling and other modifications we provided. The growth from 0,463 to 0,963 we believe is a very good result.

4. Discussion of Results and Model Evaluation

4.1. Modeling results

Obviously modeling results were different in each of the modelings we did. First one accuracy was 0,463 and in the second one it increased to 0,963.

4.2. Model evaluation

In general we think our model is good and we did the best we could to improve it. Defining the problem, preparing the dataset and applying machine learning algorithms to solve the problem were all implemented without bigger problems.

5. Reflections

5.1. What was successful?

We achieved all of our goals. Working with data, creating the models, implementing algorithms we think were successful. In general it was all new and exciting for us so we cannot be sure if we did everything properly. Probably there are some mistakes that could have been avoided, but in general we all agree that we did best we could.

5.2. What were the problems? How did we solve them?

Only problems connected to our project were basically with us learning how to use the tools that were new for us.

5.3. How can this be used in the future?

Having provided different or bigger data it would probably be able to recognize other letters or numbers if they would be included in the provided dataset. Using the same methods and algorithms we may teach the model to recognize other patterns.