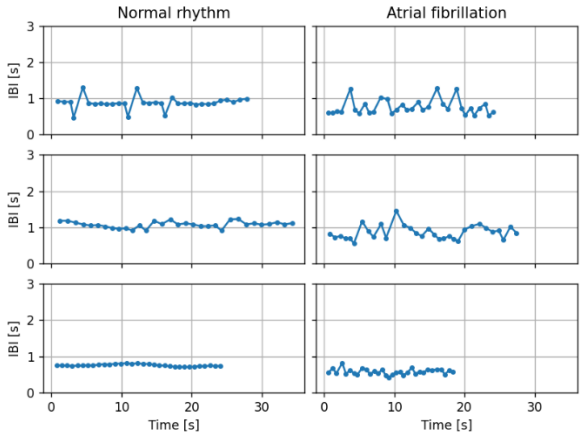
Neural Network Lab 1

Exercise 1: Atrial Fibrillation Classification

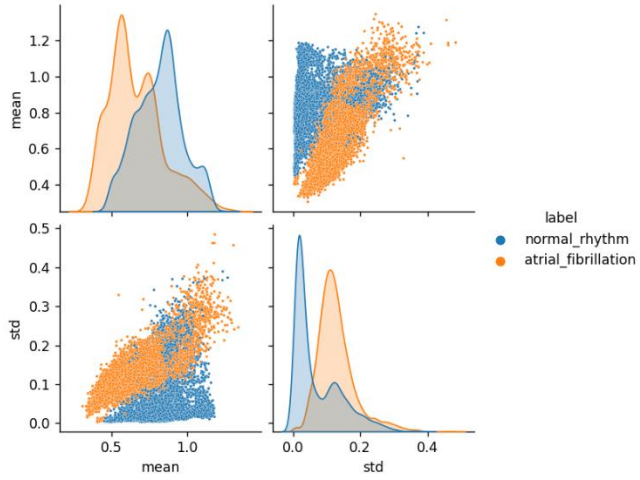
Question 1. Visually, what are the differences between the examples of the two classes?

The sequences of interbeat intervals are **more stable for normal rhythm** compared to atrial fibrillation. In addition, the **mean interbeat interval seems to be shorter** for atrial fibrillation. In the first example of normal rhythm, there are short-long pairs of intervals. They are most likely premature contractions.



Question 2. Would it be possible to discriminate between the two classes with these features and a linear classifier?

Since there is a lot of **overlap** between the two classes for the two features, a linear classifier will probably not achieve a high accuracy.



Question 3. Based on the model summaries printed above and the metrics shown in TensorBoard, answer the following questions:

- Question 3.1. After training is finished, you can see the number of parameters for each model. Why is the number of parameters for the CNN model much lower than for the MLP model?
  CNN has 2K parameters, while MLP has 20.9K. This **10-fold difference** is explained by the structure of both models. In CNN, convolutional layers apply kernels to filter the input data over a small local region of the input data (**local connectivity**). In addition, the same kernel is applied over the entire input to generate the feature map (**weight sharing**), which significantly reduce the overall number of parameters. An MLP is composed of multiple fully connected layers (dense

layers). Each node in a layer (l) is connected to all nodes in the next layer (l+1), and each of these connections has a weight (trainable parameter).

- Question 3.2. What can you say about the loss and accuracy of the different models on the training and validation subsets? Do some model overfit?
  The **MLP** model taking **features** as inputs shows sign of **overfitting**, as its validation loss starts to increase, and its validation accuracy starts to decrease at some point. The **logistic regression** model taking interbeat intervals as inputs has **poor performance** on both the training and validation sets. A linear model has not enough capacity to make use of the relations between successive intervals. It is **underfitting**. The two **best models are the CNN and RNN** models which include layers designed to take into account the relations between successive intervals.
- Question 3.3. Why does the logistic regression that takes features as inputs performs much better than the logistic regression that takes raw interbeat intervals as inputs?
  Logistic regression is a **simple model** and **cannot capture complex features** from raw interbeat intervals. For instance, one of the features computed (the standard deviation) is nonlinear, and as illustrated in the examples, one of the main differences between normal rhythm and atrial fibrillation is the variability of the interbeat intervals. While the logistic regression taking interbeat intervals could mimic features such as the mean, it is not able to learn a feature similar to the standard deviation.
- Question 3.4. Are there models that would benefit from training for more epochs?
  This can be determined by looking at the **learning curves** on the validation set. The models reaching a **plateau** would not benefit from more epochs. The validation loss of the **RNN** model and the **MLP** model taking interbeat intervals as inputs would probably keep decreasing if the training was continued for more epochs.

Question 4.

- Question 4.1. Which are the best models?
  Based on the metrics (but particularly the accuracy and the F1 score), the **CNN and RNN** models are the best models for this classification task.
- Question 4.2. What can you say about the models using features as inputs?
  The models using features as inputs **do not perform as well** as the best models. This is not surprising as only two **very simple features** are proposed. They do not capture all the relevant information about the sequences of interbeat intervals. Feature selection can reduce the dimensionality of the input space and reduce the computation demand. But if they are not chosen properly, it could result in some information loss and performance decrease.
- Question 4.3. (Bonus) Do you think it is honest to compare several models based on metrics computed on the test set in order to select the best one?
  It is not completely honest to compare models based on metrics computed on the test set as this might lead to overfitting. To best model should be **selected** based on metrics computed on the **validation set** only. The **test set** should be used as an unbiased subset, only once the best model is selected, to **evaluate** how it would perform on **unseen data**.

Question 5

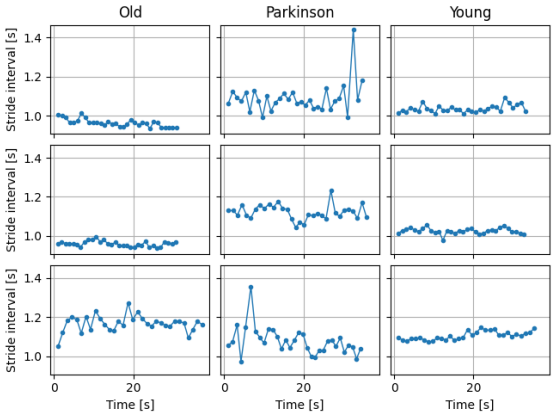
No generic answer since the model configuration must be defined by the students.

Exercise 2: Gait Classification

Question 1. Are there any visible differences between the three classes in these examples?

In the examples from the **Parkinson** class there are much more **extreme stride intervals** compared to the old class. The gait pattern of patient with Parkinson's disease is typically characterized by reduced speed,

short stride lengths, and shuffling steps. These extreme intervals are completely absent from the examples of the young class.



Question 2. Comment on the model used to split data into training, validation, and test sets. Is it appropriate? What would be another approach?

The method for splitting data into subsets for training, validation, and test is not really appropriate as windows of stride intervals from all subjects are present in all subsets. Therefore, the classification performance measured on the validation and test subsets will most likely be **overestimated** compared to the performance measured on new (unseen) data. It has been done to **balance the labels within the different subsets** (stratified sampling). An alternative would be **subject-wise splitting**, where all windows of stride intervals from a specific subject would be exclusively present in either the training, validation or test set. However, due to the small dataset size, splitting data based on subjects will probably lead to poor classification performance since there are only 15 subjects. The real solution in this case would be to **collect more data** from additional subjects.

```
Subjects in training set   : ['o1' 'o2' 'o3' 'o4' 'o5' 'pd1' 'pd2' 'pd3' 'pd4' 'pd5' 'y1' 'y2' 'y3'
 'y4' 'y5']
Subjects in validation set : ['o1' 'o2' 'o3' 'o4' 'o5' 'pd1' 'pd2' 'pd3' 'pd4' 'pd5' 'y1' 'y2' 'y3'
 'y4' 'y5']
Subjects in test set       : ['o1' 'o2' 'o3' 'o4' 'o5' 'pd1' 'pd2' 'pd3' 'pd4' 'pd5' 'y1' 'y2' 'y3'
 'y4' 'y5']
```

Question 3.

- Question 3.1. Based on the metrics shown in TensorBoard, comment on the training procedure. Does the model overfit?

  The model **does not** show sign of **overfitting**. There is a difference between metrics computed on the training and validation sets but it remains limited. The main issue is that the model seems to be stuck as the **training accuracy** remains **constant** after a few epochs.

- Question 3.2. Based on the confusion matrices, what is the main issue of the model? What is a probable explanation of these results?

  The main issue of this model is that all windows from the **Parkinson** class are **misclassified**. A possible explanation is the **imbalances dataset**, as there are much fewer Parkinson's patients than old or young subjects. A solution could be either to give more weight to patient with Parkinson's disease or to get more data of this class.

Question 4. What is the main difference of using momentum for training? Does the model overfit? Does momentum help to significantly improve the model accuracy?

Momentum accelerates gradient vectors in the right direction, by taking small steps in directions where the gradients oscillate and large steps where the past gradients are in a similar direction. Overall, with momentum, the training process **converges faster**, and the **validation accuracy is higher**. There is still

very **limited overfitting**. It starts to correctly **identify a small percentage of patients with Parkinson's** disease.

Question 5. What are the main differences of the MLP model trained with the Adam optimizer compared to the models trained with SGD (with or without momentum)? Does the model overfit? Does the Adam optimizer help to achieve better overall performance (as shown in the confusion matrices)?

Adam (Adaptive Moment Estimation) optimization is an adaptive method that computes individual learning rates for different parameters based on the first and second order momentum of the gradients. With the Adam optimizer, training is even **faster**, and the validation accuracy is even higher. However, at some point the model clearly **overfits** as the validation loss starts to increase. In addition, the gap in terms of accuracy between the training and validation sets is quite large at the end of training.

Question 6.

- Question 6.1. What are the main differences of the CNN model with respect to the MLP models trained with different optimizers? Does the CNN model overfit?
  The CNN model **performs better** than the MLP models and it does **not suffer from overfitting**. The CNN model also has an interesting property: when it **predicts Parkinson** it is almost always correct (very few false positives).
- Question 6.2. Can you think of a few reasons to explain why the MLP model trained with the Adam optimizer and the CNN model behave differently in terms of overfitting?
  The most probable reason the CNN does not overfit compared to the MLP model trained with the Adam optimizer is that it has around **10 times less parameters**. The higher the number of parameters used, the higher the risk of overfitting.
- Question 6.3. Overall, what is the main limitation of this dataset of stride intervals for training neural network models?
  The main limitation of the dataset is that it includes **only 15 subjects** (only 5 per class). This is insufficient to train a model that would perform well on unseen data. Another limitation could be the **length of the windows**. 30-secnd window might not be sufficient to clearly identify irregular peaks.