Master Thesis (732A64)

Human Age Prediction Based on Real and Simulated RR Intervals using Temporal Convolutional Neural Networks and Gaussian Processes

Mid-Term Summary

Maximilian Pfundstein (maxpf364)

March 26, 2020

1 Context

The aim of this thesis is to predict the human age given the RR intervals of a subject recorded by an electrocardiogram (ECG). This includes handling impurity of the data as well as analysing the statistical nature of this task, meaning e.g. the error distribution. Previous studies used feature based methods for this task, thus extracting 33 features from a given time series and using that for predicting the age of a given subject. The predictions were conducted by classification, thus assuming a specific age range as one class (e.g. 20-29 is one class). The first paper is the same research group that also provides one of the datasets, called the *Gdańsk dataset*. They obtained an accuracy of 98 percent for seven classes and another study obtained an accuracy of 70 percent for three classes. However, there are multiple issues with the previous studies:

- The gap in accuracy results from the issue that the first paper, with an achieved accuracy of 98 percent, did not use any validation set. Therefore, the assumption is, that the proposed model overfitted to the training data.
- Accuracy itself does not really tell anything about the predictive strength of a model, the more classes, the less the accuracy. But we are interested in Confidence Bands around our predictions.
- Ultimately, this is a regression problem, not a classification problem. A missclassification of 10 years should be punished less compared to a missclassification of 50 years. Classes do not account for this.

2 Current Status

Therefore, as a first step, the 33 features were obtained using the Python package `hrvanalysis`[1], which was also used by the authors of the first paper. The package had some small issues, so they were fixed in form of a Pull-Request[2]. The initial aim was to reproduce the previous results. It turned out that the assumption, that only a training set has been used, is true. This yielded in a very high accuracy for the training set. To confirm these findings, another dataset called *CAST RR Interval Sub-Study Database*, provided by PhysioNet, was used in addition[3], showing very similar results. The *PhysioNet dataset* is imbalanced, includes not just nocturnal sleep of patients, but a whole day and holds records of patients with diseases. Throughout the thesis, this dataset will be used as comparison to make statements about the generalisation of our trained models. Therefore, the dataset is left unbalanced on purpose.

To have a strong set of baseline models, the models were trained once on the feature set of the whole time series (*complete*) as well as a feature set for slices of around 5 minutes (*constant*), almost identical to the approach taken by the first paper, slicing 4 hours

[1] https://github.com/Aura-healthcare/hrvanalysis
[2] https://github.com/Aura-healthcare/hrvanalysis/pull/22
[3] https://physionet.org/content/crisdb/1.0.0/

of nocturnal sleep into 48 slices. After classification of the different slices, a majority vote decides for the final predicted label. Additionally, regression was conducted, with MSE as the loss function and the average age of an age decade as the corresponding label. This gives four fits for each model, if the type is supported by the `scikit-learn`[4] library. The models used as a baseline are: Naive Bayes, SVM with cross-validation and hyperparameter search, Random Forest and XGBoost. The results show that, on average, regression is slightly better compared to classification, but generally the models showed poor results. The best accuracies were around 32 percent, whereas the dummy baseline of always guessing one class lies around 14.3 percent.

The architecture which is used for the deep learning approach, which is the main part of this thesis, is inspired by a paper, proposing a model called *DeepSleep*. It consists of a CNN with two heads having different filter sizes, performing 1d-convolutions, concatenating them and then taking these representations as features for an LSTM as well as a simple Feed Forward Network (in the same architecture). The model has been implemented in PyTorch[5] and has been slightly adjusted to embed methods usually used for time series classification. Also, due to computational constraints, the size of the model has been reduced. The results show that the deep learning architecture is performing similar to the feature based models on the tasks of regression, but very poor on classification. In terms of computational time, the deep learning architecture is actually faster compared to the SVM with cross-validation, as the task can be highly parallelised using a GPU.

Looking at the error distributions of all models, it can be seen that the 95 percent prediction bands spans a range of more than 70 years. Concluding from that, the age of a person cannot accurately be predicted only given the RR intervals.

To improve the results and understand the data better, Gaussian Processes are fitted to slices of around 2.5 minutes (around 270 points in the time series) for the smaller *Gdańsk* dataset. The advantages are:

- Missing data points simply increase the variance around the missing points, but can be evaluated.
- As a GP itself is a distribution, it can be used to sample as many training data as desired, thus the hope is that this improves the training process for the deep learning architecture and leading to less overfit as observed with the original data.
- The GP can be evaluated at as many points as required, thus yielding in slices of the exact same length, making it quite applicable for the deep learning approach.

The fit of GPs is conducted for the training, validation and test dataset, as it resembles a transformation (even if stochastic). The kernel function of the GP is rather complex, with about 20 hyperparameters and is implemented using TensorFlow Probability[6], which is basically a GPU accelerated version of (R-)Stan. For obtaining good kernel

[4]https://scikit-learn.org/stable/
[5]https://pytorch.org/
[6]https://www.tensorflow.org/probability

hyperparameters, there exist multiple methods like grid search. This thesis focuses mainly on sampling kernels from the posterior predictive distribution by integrating out the parameters of the model. This is done for each slice, whereas the obtained posterior distribution for each parameter of each slice is then taken as a prior for the next slice, thus consecutively embedding the information within the different slices into the GP. At this time, the code for this is already implemented, where a fit for one slices takes around 45 seconds. Having 96 slices for 108 time series for the training dataset for example, this takes around 130 hours plus the time for the test and validation set. Fortunately, once a posterior predictive distribution is available, multiple samples can be taken as the same time (here 100). This means, even if not all available data can be used for this data generation process, more data than originally will be available. A more traditional approach might be considered as well, where point estimates for the different hyperparameters are obtained by gradient descent, to compare both approaches (how much influence has the uncertainty of the parameters?). As this takes around 8 minutes for each slice, we are currently working towards calculating the derivatives with respect to each parameters analytically, as this could speed up the process. On success, this will be included in the thesis as well.

Having the simulated data available, the best models will be fit again and the predictive strengths are going to be analysed and compared to the original data.

3 Outlook

Should there be more time towards the end and all the above is implemented and documented, then instead of RR intervals the real recorded signals can be analysed and it can be investigated, if there lies more information within this data. This will only be possible for a dataset provided by PhysioNet, as the original signal is not available for the RR data sets. A conversation with a doctor at Norrköpings hospital hinted that there might be more information in the real data (also accounting e.g. for the QRS-complex). Theoretically, a CNN can easily work on multidimensional data (which is just a higher amount of initial filters), thus QQ- and SS intervals could be included in a first step (or simply taking the whole original time series). Apart from that, the available methods can be improved (weighted classes, under- or oversampling of the imbalanced PhysioNet dataset).