# Machine learning and deep learning approaches for multivariate time series prediction and anomaly detection

Thill, M.

# Summary

Up until today, anomaly detection in general and especially for temporal data such as data streams or time series remains a challenging task. Nevertheless, anomaly detection in time series is becoming increasingly important in many domains. For example, more and more industrial machines are equipped with numerous sensors for condition monitoring and predictive maintenance applications, which produce large amounts of data in the shortest time. In such applications, it is crucial to continuously monitor these large data streams and detect anomalous behavior at an early stage to prevent critical systems' failure and possible further damage. Also, in many other application areas (e.g., server technologies & networking, high energy physics & astronomy, or health monitoring systems), vast amounts of data are collected, which have to be analyzed automatically with suitable algorithms. In this context, the necessity for accurate and robust unsupervised learning methods for anomaly detection tasks arose.

In this thesis, we present and discuss several unsupervised anomaly detection algorithms for time series.

Our first algorithm, called SORAD (Simple Online Regression for Anomaly Detection), is presented in Chapter 4. SORAD learns a linear regression model to predict future values of a time series and simultaneously estimate the prediction errors' mean and variance. Data points resulting in prediction errors that significantly deviate from the mean are considered anomalous. SORAD can operate fully online (or using small batches) on any time series after a short transient phase and adapt to changing concepts (due to the forgetting in the linear model and the estimates of the prediction errors). On the Webscope S5 anomaly benchmark [87] SORAD could significantly outperform other algorithms. However, it had some shortcomings on the Numenta Anomaly Benchmark (NAB) [89], where it did not perform well at first.

In Chapter 5, we introduced the DWT-MLEAD (Discrete Wavelet Transform with Maximum Likelihood Estimation for Anomaly Detection in Time Series) algorithm. This algorithm's central idea is to analyze a time series on different frequency/time scales using the discrete wavelet transform (DWT). DWT-MLEAD (i) computes a decimating DWT using Haar wavelets, (ii) slides a window over the individual frequency scales and estimates the mean and covariance matrix of the collected points, and (iii) identifies unusual points based on a Mahalanobis distance. Then (iv), the unusual points are flagged as events and passed down the DWT-tree. Finally (v), the events of all frequency scales are aggregated at the

original time scale and cause the algorithm to fire an anomaly if a specified threshold is exceeded. Using this multi-scale approach, it is possible to detect short-range and also longer-range anomalies. Initially designed as an offline algorithm, we showed that DWT-MLEAD could also be used in online settings after a few modifications. We found that DWT-MLEAD outperforms other state-of-the-art algorithms, the Webscope S5 and the NAB data using only one hyperparameter setting.

The LSTM-AD algorithm, described in Chapter 6, learns to predict normal time-series behavior using recurrent long short-term memory (LSTM) neural networks. From the general idea, LSTM-AD is similar to SORAD. However, it extends it in several aspects: Instead of using a simple linear regression model LSTM-AD uses a stack of LSTM layers to learn more complicated temporal patterns. Furthermore, now 25 prediction horizons are used instead of only one. The prediction errors for several prediction horizons are used to indicate anomalous behavior (using the Mahalanobis distance of the prediction errors as anomaly score). Several additional extensions, such as a window-based error correction and an unsupervised outlier removal method are introduced. LSTM-AD was evaluated on the well-known MIT-BIH ECG [52, 112, 113] data set and could obtain good initial results. Later, in Chapter 7, even better results are found with an improved hyper-parameter selection.

In Chapter 7, the temporal convolutional network autoencoder (TCN-AE) is introduced. The main idea is to use one TCN [13] as an encoder network and another TCN as a decoder network. In order to create a bottleneck, the encoder downsamples the time series and reduces its dimensionality. After the input time series is encoded into a compressed sequence, a temporal decoder network attempts to reconstruct the original input. TCN employs so-called dilated convolutions, which have their origin in discrete wavelet transforms and create an exponentially increasing receptive field with only a linear increase in the number of trainable weights. We performed our initial experiments with a relatively simple baseline TCN-AE architecture. The baseline TCN-AE demonstrated that it could already learn interesting representations of different Mackey-Glass time series, and it performed well on the Mackey-Glass Anomaly Benchmark (MGAB). However, we noticed a few shortcomings, which we addressed in the following. We showed that several extensions of the baseline model are crucial to improving the overall performance. TCN-AE in its final form produces significantly better results on the MIT-BIH ECG data, outperforming other state-of-the-art algorithms. At the same time, it reduces the computation time and the number of trainable parameters (in comparison to baseline TCN-AE).