

УДК 004.85

***ИСПОЛЬЗОВАНИЕ НЕЙРОСЕТЕЙ ТИПА АВТОКОДИРОВЩИК ДЛЯ
ОБНАРУЖЕНИЯ АНОМАЛИЙ ВО ВРЕМЕННЫХ РЯДАХ***

Тельшев А.В.,

студент 2 курса магистратуры

Московский государственный технический университет им. Н.Э. Баумана

Россия, г. Москва

Назарков Д.А.,

студент 2 курса магистратуры

Московский государственный технический университет им. Н.Э. Баумана

Россия, г. Москва

Аннотация

В данной статье рассмотрено понятие аномалий во временных рядах, методы, применяемые к обнаружению аномалий, а также алгоритм поиска аномалий на основе нейронных сетей специального типа – автоматических автокодировщиков.

Ключевые слова: нейронная сеть, автокодировщик, временной ряд, аномалия.

***NEURAL NETWORKS TYPE OF AUTOENCODERS FOR ANOMALY
DETECTION IN TIME SERIES***

Telyshev A.V.,

master student

Bauman Moscow State Technical University

Russian Federation, Moscow

Nazarkov D.A.,

master student

Дневник науки | www.dnevnika.ru | СМИ ЭЛ № ФС 77-68405 ISSN 2541-8327

Bauman Moscow State Technical University
Russian Federation, Moscow

Annotation

This article discusses the concept of anomalies in the time series, the methods applied to the detection of anomalies, as well as the algorithm for finding anomalies based on a special type of neural networks – autoencoders.

Keywords: neural network, autoencoder, time series, anomaly.

Аномалии – это образцы данных, не соответствующие определенному понятию нормального (неаномального) поведения. Аномалии приводят к получению полезной информации в различных областях применения – этим обуславливается важность выявления аномалий. Например, аномальная смена трафика в компьютерной сети может означать, что компьютер взломан и отправляет конфиденциальные данные неавторизованному адресату. Аномалии в данных транзакций по кредитной карте могут указывать на банковское мошенничество, а аномалии на МРТ могут быть следствием злокачественных опухолей [3].

Временной ряд – это собранный в разные моменты времени статистический материал о значении каких-либо параметров (в простейшем случае одного) исследуемого процесса. Другими словами, упорядоченная во времени последовательность значений какого-либо датчика, имеющая высокую значимость в определении состоянии системы и часто полученная в реальном времени. Таким образом, остро встает задача анализа данного типа данных [1].

Существует несколько подходов к обнаружению аномалий во временных рядах:

- Методы обнаружения «без учителя», обнаруживающие аномалии в наборе тестовых данных без маркировки при условии, что большинство экземпляров в

наборе данных являются нормальными (путем поиска экземпляров, которые меньше всего соответствуют остальной части набора данных).

- Методы обнаружения «с учителем», требующие наборы данных, помеченные как «нормальный» и «аномальный»: представляют собой обучение классификатора.
- Методы, использующие обучение модели на нормальных данных, а затем анализирующие разницу между реальными данными и данными, предсказанными этой моделью.

В данной статье мы остановимся на первой категории (обучение «без учителя»), а именно, рассмотрим алгоритм определения аномалий во временных рядах на основе автоматического кодировщика. Автокодировщики предоставляют очень мощную альтернативу традиционным методам восстановления сигнала и обнаружения аномалий во временных рядах.

Автоматический кодировщик – это искусственная нейронная сеть, используемая для неконтролируемого изучения эффективных кодировок.

Его цель состоит в том, чтобы получить представление (кодирование) набора данных путем изучения приближения тождественной функции этих данных [4]:

$$Id: X \rightarrow X \quad (1)$$

Архитектурно простейшая форма автокодировщика представляет собой прямую, невозвратную нейронную сеть, которая очень похожа на многослойный перцептрон (multilayer perceptron, MLP), с входным уровнем, уровнем выхода и одним или несколькими скрытыми слоями, соединяющими их.

Однако различия между автокодировщиками и MLP заключаются в том, что в автоматическом кодировщике выходной уровень имеет такое же количество узлов, что и входной, и что вместо того, чтобы обучаться предсказанию целевого значения Y , заданному входом X , автокодировщик обучается реконструировать свои собственные X . Поэтому автокодировщики являются неконтролируемыми обучающимися моделями [4].

Автокодирование всегда состоит из двух частей: зашифровщика и дешифровщика, которые могут быть определены как переходы φ и ψ , такие, что [4]:

$$\begin{cases} \varphi: X \rightarrow F \\ \psi: F \rightarrow X \\ \varphi, \psi = \arg \min_{\varphi, \psi} \|X - (\varphi \circ \psi)X\|^2 \end{cases} \quad (2)$$

В простейшем случае, когда есть один скрытый слой, автокодировщик принимает вход $x \in \mathbb{R}^d$ и отображает его на $z \in \mathbb{R}^p = F$ ($p < d$):

$$z = \sigma_1(Wx + b) \quad (3)$$

Обычно это называют кодовыми или скрытыми переменными (скрытое представление). Здесь σ_1 является элементарной функцией активации, такой как сигмоидальная функция или выпрямленная линейная единица.

После этого z отображается на реконструкцию x' той же формы, что и x :

$$x' = \sigma_2(W'z + b') \quad (4)$$

Автокодировщики также обучены минимизировать ошибки восстановления (например, квадратичные ошибки):

$$\mathcal{L}(x, x') = \|x - x'\|^2 = \|x - \sigma_2(W'(\sigma_1(Wx + b)) + b')\|^2 \quad (5)$$

Основной задачей нейронных сетей типа автокодировщик относительно временных рядов является нахождение некоторого алгоритма, неизвестной ранее зависимости между множеством входных объектов (ситуаций) и множеством возможных ответов (реакций), когда известна только конечная совокупность прецедентов – пар «объект, ответ», называемая обучающей выборкой.

Рассмотрим основной способ выявления аномалий с помощью автокодировщика: нахождение ошибок на основе разницы между исходными данными и данными на выходе натренированной сети при подаче ей на вход тех же исходных данных.

Общая концепция подхода выглядит следующим образом [2]:

- тренируем автокодировщик на данных X_{train} с хорошей регуляризацией (предпочтение отдается повторяющимся данным или данным, представляющим периодический временной процесс);

- отдаем автокодировщику валидационный набор X_{val} , сортируем и визуализируем полученный выход, содержащий ошибки;
- выбираем пороговое значение для единицы данных (обычно равное двум стандартным отклонениям от среднего значения), определяющее является ли значение аномалией или нет; этот порог может быть динамическим и зависеть от предыдущих найденных выбросов (скользящее среднее).

Допустим, что входной вектор X имеет многомерное нормальное распределение $X \in \mathbb{R}^{dt} \sim N(0, \Sigma_t)$, где X_t^i не зависит от X_t^j для данного t и $i \neq j$. Также искусственно создадим некий процент аномалий во входном векторе, такой что $P(a) \ll 1$, где a – множество аномальных точек.

Задача заключается в нахождении временных индексов $t_{a0} \dots t_{an}$, соответствующих аномальным элементам в измененном входном векторе X .

На рисунке 1 изображен пример обнаружения аномалии в синусоидальном сигнале: на верхней части рисунка отображен сам сигнал; на средней части рисунка – сигнал, восстановленный автокодировщиком; а на нижней части рисунка – квадратичная ошибка между двумя верхними временными рядами. Мы видим, что для аномальной области функция ошибки превышает определенную границу, что свидетельствует об аномалии.

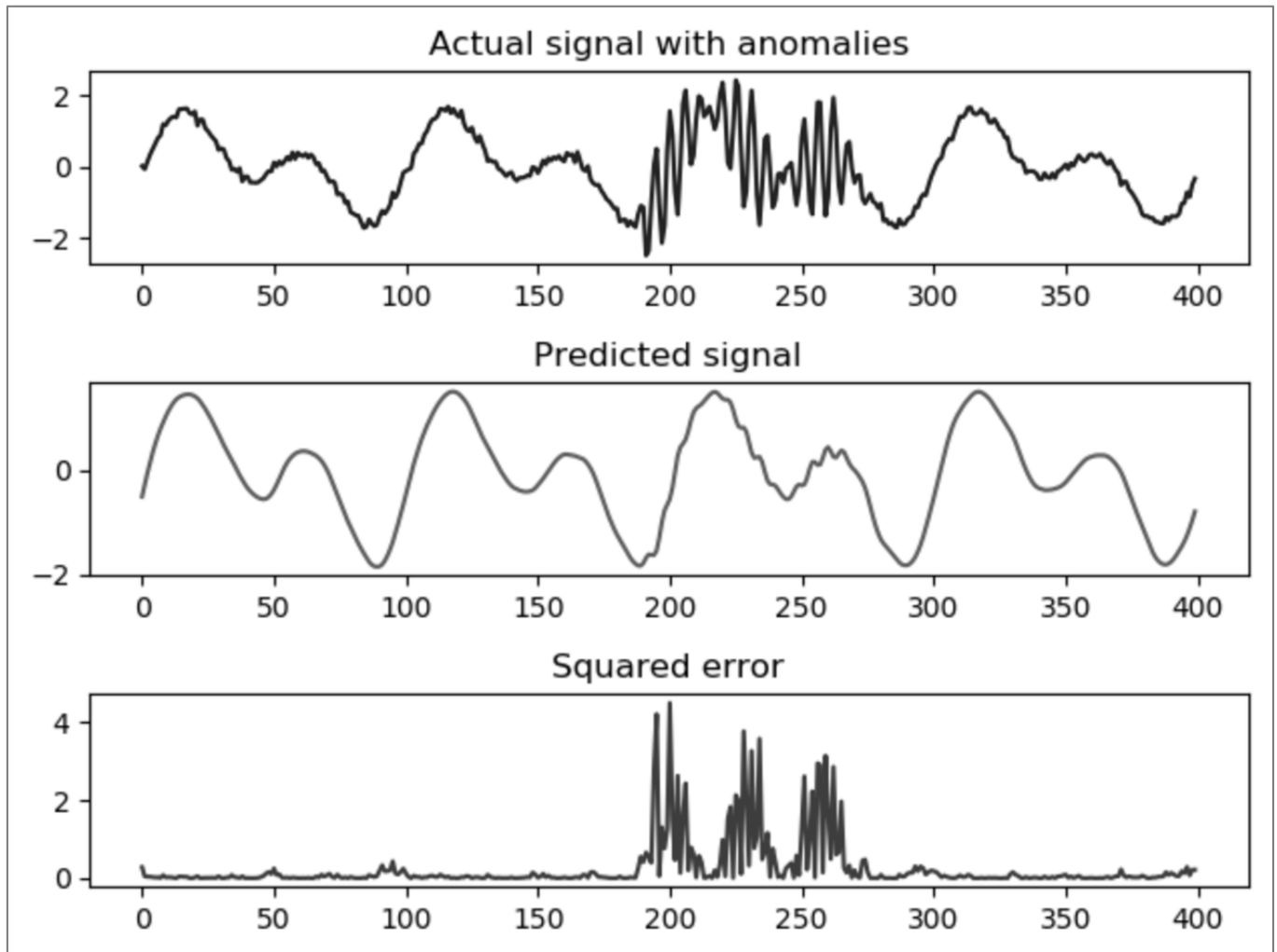


Рис. 1 – Пример обнаружения аномалии в синусоидальном сигнале

Таким образом, мы показали, что нейронные сети типа автокодировщик применимы для решения задачи обнаружения аномалий во временных рядах. Описанный в статье подход наиболее эффективен для периодических временных рядов, но в том или ином случае обучение автокодировщика должно происходить на «нормальных» данных, что можно отнести к недостаткам метода.

Библиографический список:

1. Мишулина О. А. Статистический анализ и обработка временных рядов. — М.: МИФИ, 2004. — С. 180.

2. Anomaly Detection in Time Series using Auto Encoders [Электронный ресурс]. — Режим доступа — URL: <http://philipperemy.github.io/anomaly-detection/> (Дата обращения 5.04.2019).

3. Arindam B., Varun C., Vipin K. Anomaly detection: A survey // ACM Computing Surveys, 2009.

4. Bengio, Y. Learning Deep Architectures for AI // Foundations and Trends in Machine Learning, 2009. P. 1–127.

Оригинальность 96%