

АЛГОРИТМЫ PSO, FA И GD ДЛЯ ПРОГНОЗИРОВАНИЯ СБОЕВ ПРОМЫШЛЕННОСТИ

PSO, FA AND GD ALGORITHMS FOR PREDICTING INDUSTRY FAILURES

I. Hamameh

Summary. One of the goals of the smart industry is to reduce the number of failures, which reduces costs. To achieve this goal, technological processes are monitored and divided into multiple processes and the probability of success per process are calculated. The main contribution to this work was the proposal of methods for training neural networks.

Keywords: Artificial neural network; Fault prediction; Machine learning; Algorithms inspired by nature.

Хамамех Имад Нехадович

Аспирант, ФГБОУ ВО «МИРЭА — Российский технологический университет», Москва
imad.hamameh@gmail.com

Аннотация. Одна из целей интеллектуальной промышленности является уменьшение количества сбоев, что снижает затраты. Для достижения этой цели ведется мониторинг технологических процессов и определяется вероятность выполнения операций, на которые разделен технологический процесс. Основной вклад в эту работу заключался в предложении методов обучения нейронных сетей.

Ключевые слова: искусственная нейронная сеть, прогнозирование неисправностей, машинное обучение, алгоритмы, вдохновленные природой.

Введение

Повышение производительности технологических процессов является одной из важнейших задач в промышленности, и для достижения этой цели применяются различные технологии. Искусственный интеллект все больше и больше используется в этих задачах.

Искусственный интеллект можно использовать для прогнозирования различных событий с помощью нейронных сетей. Одна из задач при создании нейронных сетей — это обучение нейронной сети.

Существует много методов обучения, и выбор более подходящего метода — это одна из основных проблем, с которой часто сталкиваемся при создании нейронной сети.

Цель данной работы — сравнить алгоритмы обучения нейронных сетей и использовать различные методы минимизации для определения лучшего метода обучения для задач прогнозирования в промышленности.

Несмотря на то, что существует много работ со сравнением и описанием основ использования алгоритмов обучения, в данной работе используются данные производственных процессов, которые отличаются от обычных данных, в основном, количеством исторических

данных, поэтому данная работа поможет упростить обучение нейронных сетей в данной области.

1. Сопутствующие работы

Прогнозирование играет важную роль в улучшении эффективности технологического процесса. Прогнозирование выполнимости технологического процесса позволяет производствам мелкосерийного типа, обслуживающим разные заказы, определять количество и тип заказов, которые они смогут одновременно обслужить, что помогает в принятии решения обслуживания какого-либо заказа.

Также прогнозирование помогает определить вероятность отказа в будущем, что позволяет снизить время, потраченного на ремонт. Нейронные сети являются ключевой технологией в проведении прогнозирования.

1.1. Нейронные сети в прогнозировании

Искусственные нейронные сети (ИНС) — структуры, вдохновленные биологическими нейронами и образованные простыми единицами обработки, называемыми нейронами. Нейроны связаны друг с другом, и для каждого соединения дается синоптический вес.

Фаза обучения ИНС регулирует синоптические веса этих связей, моделируя соотношение входов и выхо-

дов системы. ИНС способны моделировать нелинейные и сложные задачи и их легко реализовать [1]. Одна из задач, которую может выполнять ИНС — это прогнозирование.

В медицине ИНС использовались для прогнозирования риска смертности и заболеваемости [2]. Hao, Usama, Yang, Hossain и Ghoneim представили новую мультимодальную рекуррентную нейронную сеть на основе данных (MD-RCNN) для прогнозирования риска заболеваний.

Li, Ren и Lee разработали инновационную структуру для многоэтапного прогнозирования скорости ветра с использованием рекурсивной нейронной сети на основе скорости ветра и интенсивности турбулентности [3].

Dumont, Rughani и Tranmer использовали нейронную сеть для прогнозирования симптоматического церебрального вазоспазма после аневризматического субарахноидального кровоизлияния и сравнили её точность с моделями логистической регрессии [4].

1.2. ИНС в прогнозировании сбоев

ИНС широко используются в прогнозировании сбоев.

Orrù, Zoccheddu, Sassu, Mattia, Cozza и Arena представили простую и легко реализуемую модель машинного обучения (ML) для раннего прогнозирования неисправностей центробежного насоса в нефтегазовой отрасли [5].

Cheng H., Kong X., Chen G., Wang Q. и Wang R. предложили метод для прогнозирования оставшегося срока полезного использования при множественных отказах системы, используя переносимую сверхточную нейронную сеть (TCNN) для изучения инвариантных функций домена [6]. Atma Ram Sahu, Sanjay Kumar Palei продемонстрировали подход управления данными для прогнозирования неисправностей в системе перетаскивания с использованием многослойного перцептрона (MLP) в искусственной нейронной сети с использованием данных о причинах, симптомах и неисправностях за последние два года, записанных с помощью датчика, журнала и визуального осмотра [7].

1.3. ИНС в расчёте вероятности

ИНС используется для расчета вероятности, для систем планирования ресурсов предприятия (ERP). Достижение надлежащего уровня успеха ERP зависит от множества факторов, связанных с организацией или проектной средой, Rouhani и Ravasan создали систему

для прогнозирования вероятности успеха системы планирования ресурсов, до её реализации [8].

Hsieh M.H., Hsieh M.J., Chen C.M., Hsieh C.C., Chao C.M. и Lai C.C. создали систему для снижения риска длительной искусственной вентиляции легких, которая определяет подходящее время для успешного отлучения пациента от ИВЛ с помощью нейронной сети [9].

2. Методы, примененные при разработке предлагаемой системы

В реальных системах сбор и обработка данных, используемых для обучения модели машинного обучения, должны выполняться исторически относительно интересующей точки, например, сбоя оборудования. Определяя, когда конкретное оборудование вышло из строя, собирается предварительный набор данных, содержащий данные датчиков. Далее обрабатываются данные для получения подходящей обучающей выборки для обучения нейронной сети.

В этой работе представлены данные с 15 параметрами и более 7000 показаний [10].

Использовались 2 вида оптимизации:

1. Метод роя частиц.
2. Стохастический градиентный спуск.
3. Метод FA

Цель данной работы сравнить результаты алгоритмов, и найти более подходящие алгоритмы для обучения нейронной сети при небольшом объеме данных, что помогает расширить области производства, где можно применять методы прогнозирования в технологических процессах.

2.1. Метод роя частиц

Оптимизация роя частиц (PSO) — это метаэвристика, введенная Кеннеди и Эберхартом [15].

Алгоритм PSO был вдохновлен наблюдением за стаей птиц. PSO поддерживает набор решений, которые итеративно обновляются для перемещения в пространстве поиска. Математические формулы применяются к скорости и положению каждой частицы. Скорость частицы определяет, насколько быстро частица перемещается в пространстве поиска, в то время как положение частицы представляет собой решение исследуемой проблемы. Расчеты каждой итерации основаны на двух ранее найденных позициях. Скорость частицы вычисляется на каждой итерации по следующему уравнению:

$$v_i(t+1) = \omega v_i(t) + c_1 r_1 (y(t) - x_i(t)) +$$

$$+ c_2 r_2 (\hat{y}(t) - x_i(t)),$$

где t — текущая итерация, $x_i(t)$ — текущее положение частицы в измерении i , $v_i(t)$ — текущая скорость частицы в размерности i , $y_i(t)$ — личное лучшее положение текущей частицы в измерении i , $\hat{y}(t)$ — лучшая позиция в измерении i , ω инерционный член, который применяет часть предыдущей скорости к следующей скорости, $C1$ — когнитивный компонент, который влияет на эффект личного наилучшего найденного положения, $C2$ — социальный компонент, который влияет на эффект глобального наилучшего найденного положения, а $R1$ и $R2$ — случайные значения в диапазоне $[0, 1]$ [10].

2.2. Стохастический градиентный спуск

Это итерационный метод оптимизации целевой функции с подходящими свойствами гладкости. Его можно рассматривать как стохастическое приближение оптимизации градиентного спуска, поскольку оно заменяет фактический градиент (вычисленный из всего набора данных), его оценкой (вычисленной из случайно выбранного подмножества данных). Это снижает вычислительную нагрузку, особенно в задачах оптимизации большой размерности, обеспечивая более быстрые итерации в торговле для более низкой скорости сходимости.

Пусть имеется функция $C(x_1, x_2, \dots, x_n)$. При движении с малыми шагами Δx_n получим:

$$\Delta C = \frac{\partial C}{\partial x_1} \Delta x_1 + \frac{\partial C}{\partial x_2} \Delta x_2 + \dots + \frac{\partial C}{\partial x_n} \Delta x_n$$

Надо выбрать такие значения $\Delta x_1, \Delta x_2, \dots, \Delta x_n$, при которых ΔC будет отрицательной.

Для этого определим:

$$\Delta x = (\Delta x_1, \Delta x_2, \dots, \Delta x_n)^T,$$

$$\nabla C = \left(\frac{\partial C}{\partial x_1}, \frac{\partial C}{\partial x_2}, \dots, \frac{\partial C}{\partial x_n} \right)$$

где Δx — вектор изменений;

∇C — вектор градиента функции.

Если выбрать $\Delta x = -\eta * \nabla C$, получим

$$\Delta C = \nabla C * \Delta x = \nabla C * (-\eta * \nabla C) = -\eta * \|\nabla C\|^2$$

$$\nabla C = -\eta * \|\nabla C\|^2.$$

С помощью этого можно гарантировано получить $\Delta C \leq 0$ и получим

$$x_{new} = x_{old} - \eta * \|\nabla C\|^2$$

Таким способом функция C будет двигаться в сторону минимума. Одна из проблем при использовании градиентного спуска заключается в том, что при обучении нейронной сети требуется рассчитать Функцию стоимости по формуле (2.1) много раз, и при большой обучающей выборке это займёт много времени. Метод, называемый стохастическим градиентным спуском, может быть использован для ускорения обучения. Идея метода состоит в том, чтобы оценить градиент ∇C путем вычисления небольшой выборки случайно выбранных обучающих входов. При выборе среднего из этой небольшой выборки получается, что мы можем быстро получить хорошую оценку градиента ∇C , и это помогает ускорить обучение.

2.3. Метод FA

Алгоритм FA был впервые разработан Синь-Ше Янгом в конце 2007 и 2008 годов, изначально предназначен для решения задач непрерывной оптимизации [11]. Литература по FA значительно расширилась за последние 5 лет: опубликовано несколько сотен статей об алгоритмы FA. FA использует следующие три идеализированных правил:

- ◆ Светлячки однополюсы, поэтому одного светлячка привлекают другие светлячки независимо от их пола.
- ◆ Привлекательность пропорциональна яркости, и они оба уменьшаются по мере увеличения расстояния. Таким образом, для любых двух мигающих светлячков более яркий будет двигаться к более яркому. Если нет более яркого светлячка, чем конкретный светлячок, он будет перемещаться случайным образом.
- ◆ Яркость светлячка определяется ландшафтом целевой функции.

Поскольку привлекательность светлячка пропорциональна интенсивности света, наблюдаемому соседними светлячками, можно определить изменение привлекательности β с расстоянием r следующим образом:

$$\beta = \beta_0 e^{-r^2},$$

где β_0 — привлекательность при $r = 0$.

Движение светлячка i притягивается к другому, более привлекательному (более яркому) светлячку j определяется следующим образом:

$$x_i^{t+1} = x_i^t + \beta_0 e^{-r_{ij}^2} (x_j^t - x_i^t) + b_t \epsilon_i^t,$$

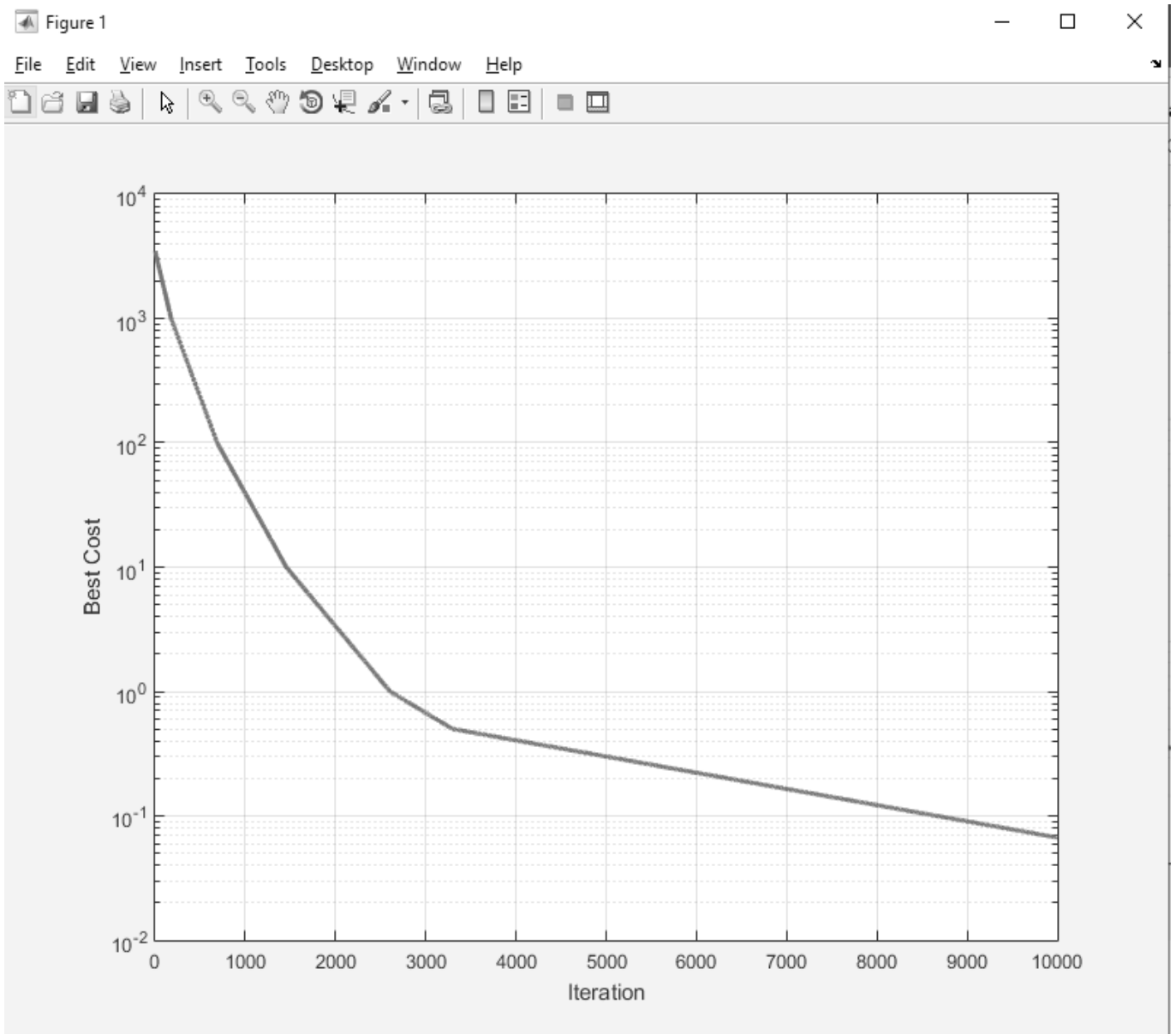


Рис. 1. Метод FA

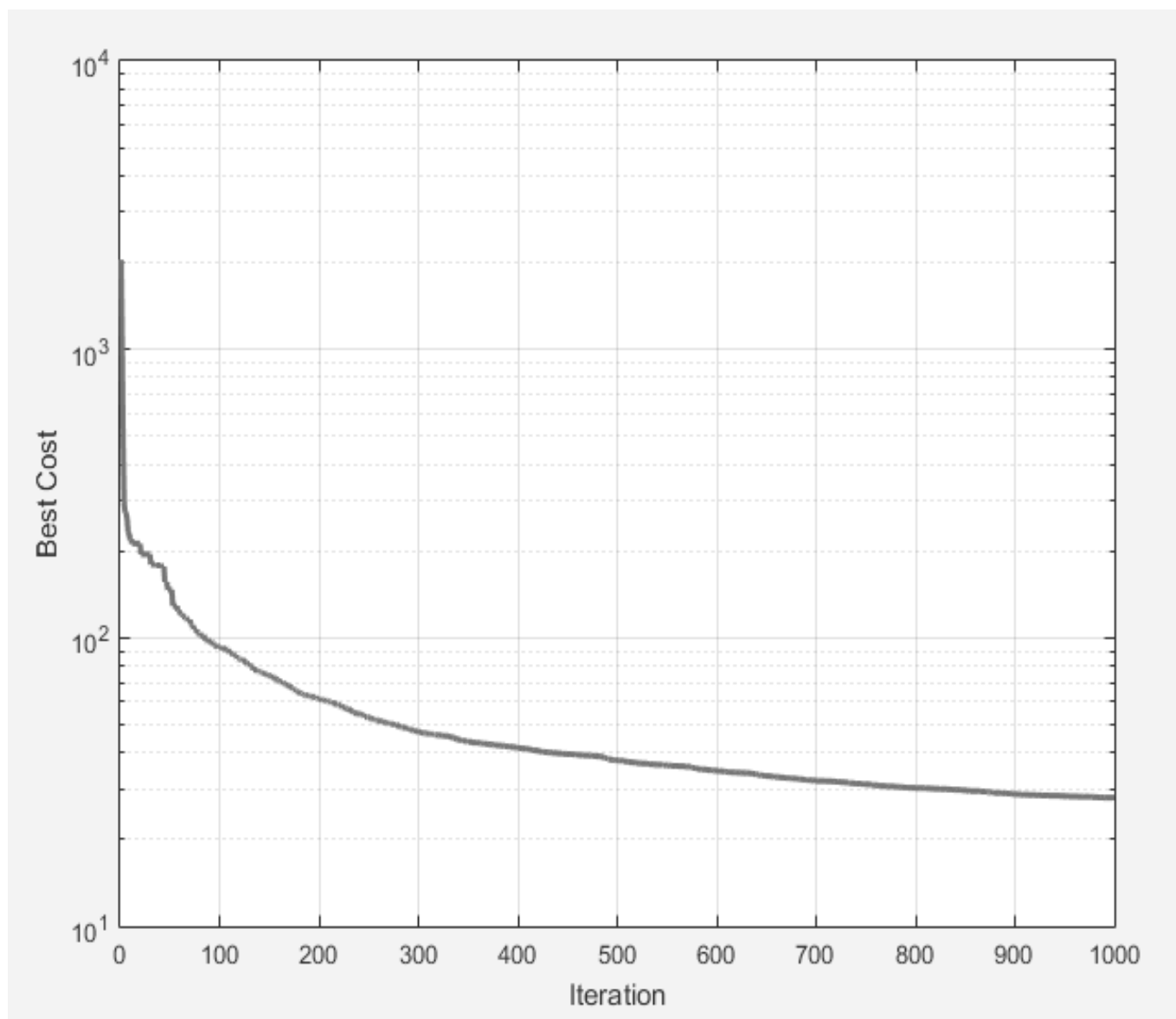


Рис. 2. Метод PSO

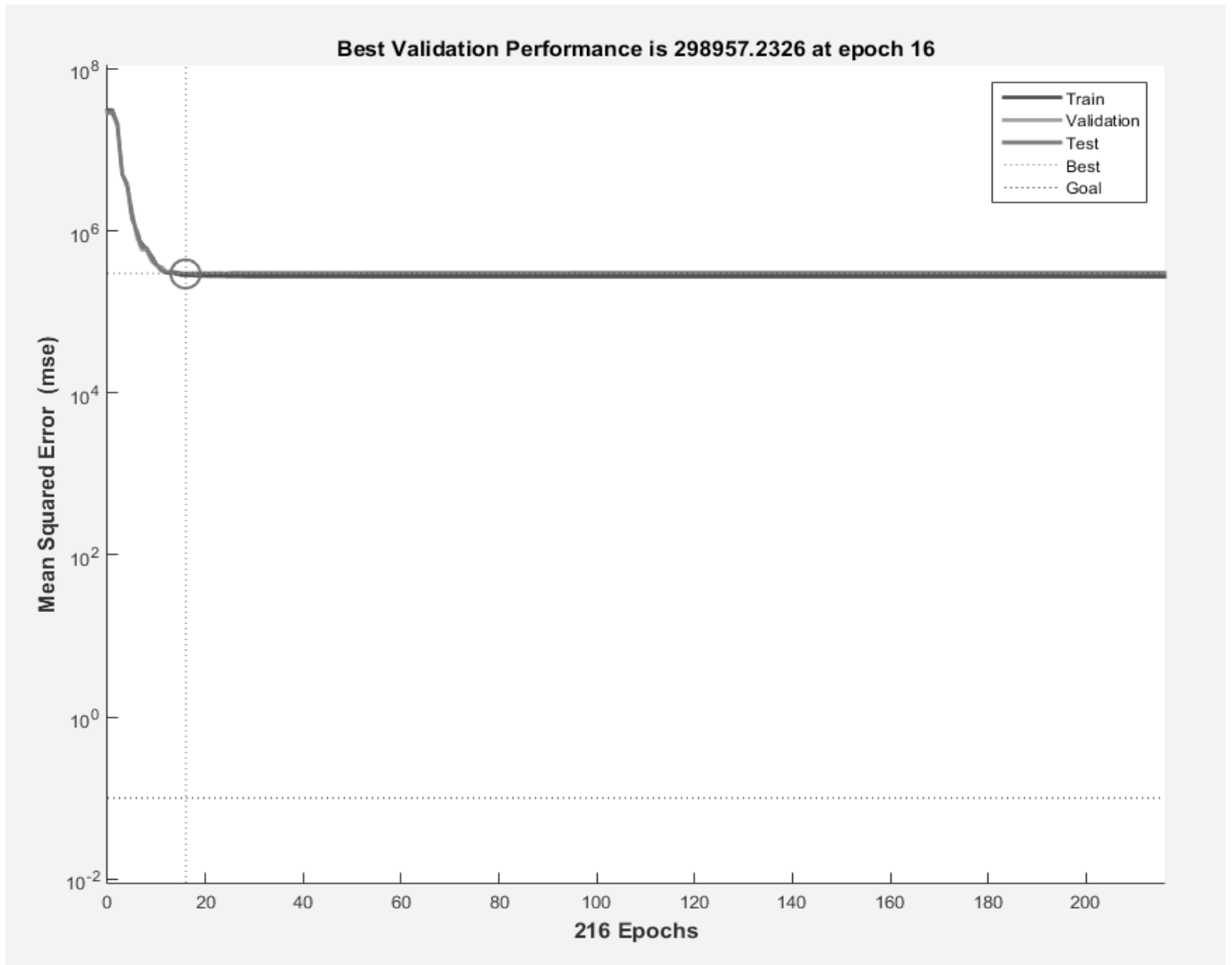


Рис. 3. Метод GD

где вторая часть обусловлена притяжением. Третья часть — это рандомизация, где αt является параметром рандомизации, а ϵ_i^t — вектор случайных чисел, взятых из распределения Гаусса или равномерного распределения в момент времени t . Если $\beta_0 = 0$, это становится простым случайным блужданием. С другой стороны, при $\Gamma = 0$ FA сводится к варианту оптимизации роя частиц. Кроме того, рандомизацию можно легко распространить на другие распределения, такие как рейсы Леви [12].

3. Результаты работы

В ходе исследования были созданы и обучены различные нейронные сети. По результатам исследования было выявлено, что Метод FA дал лучшие результаты за счет времени обучения (рис. 1).

Метод роя частиц (PSO) обучается быстрее, но выдает не лучшие результаты. Для данных прогнозирования с PSO и 250 скрытых нейронов, получаем график обучения рис. 2.

С алгоритмом GD нейронная сеть не обучается рис. 3.

Заключение

По результатам данной работы было выявлено, что алгоритм Firefly, разработанный Янгом (2008), представляет собой очень эффективный популяционный метод. Социальное поведение и мерцание светлячков можно легко связать с целевой функцией данной задачи оптимизации.

В этой статье FA предлагается и тестируется для применения при обучении нейронной сети для прогнозирования ошибок. Сравнение алгоритмов показывает, что модель, полученная с помощью FA, является более точной.

По сравнению с алгоритмами оптимизации и итерационными методами, метаэвристические алгоритмы не гарантируют, что глобально оптимальное решение может быть найдено для некоторого класса задач. Метаэвристические алгоритмы часто находят хорошие решения с минимальными вычислительными затратами, чем обычные алгоритмы оптимизации, что помогает в наших случаях с большим количеством параметров. Следует провести более обширные исследования этих алгоритмов, чтобы получить наилучший результат для обучения сети, способной распознавать ошибки.

ЛИТЕРАТУРА

1. Daoud M.; Mayo M., A survey of neural network-based cancer prediction models from microarray data, 2019, Artificial Intelligence in Medicine 97, Volume 97, Pages 204–214
2. Hao Y., Usama M., Yang J., Hossain M.S., Ghoneim A., Recurrent convolutional neural network based multimodal disease risk prediction, 2018, Future Generation Computer Systems, Volume 92, Pages 76–83
3. Li F., Ren G., Lee J., Multi-step wind speed prediction based on turbulence intensity and hybrid deep neural networks, 2019, Energy Conversion and Management, Volume 186, Pages 306–322
4. Dumont T.M.; Rughani A.I., Tranmer B.L., Prediction of Symptomatic Cerebral Vasospasm after Aneurysmal Subarachnoid Hemorrhage with an Artificial Neural Network: Feasibility and Comparison with Logistic Regression Models, World Neurosurgery Volume 75, Issue 1, 2011, Pages 57–63
5. Orrù P.F.; Zoccheddu A.; Sassu L.; Mattia C.; Cozza R.; Arena S., Machine Learning Approach Using MLP and SVM Algorithms for the Fault Prediction of a Centrifugal Pump in the Oil and Gas Industry, 2020, 9th International Conference "Production Engineering and Management" (PEM 2019)
6. Cheng, H.; Kong, X.; Chen, G.; Wang, Q.; Wang, R., Transferable convolutional neural network based remaining useful life prediction of bearing under multiple failure behaviors, 2021, Measurement, Volume 168
7. Sahu A.R., Palei S.K., Fault prediction of drag system using artificial neural network for prevention of dragline failure, 2020, Engineering Failure Analysis, Volume 113
8. Rouhani S.; Ravasan A.Z., ERP success prediction: An artificial neural network approach, Scientia Iranica, Volume 20, Issue 3, 2013, Pages 992–1001.
9. Hsieh M.H., Hsieh M.J., Chen C.M., Hsieh C.C., Chao C.M. и Lai C.C., An Artificial Neural Network Model for Predicting Successful Extubation in Intensive Care Units, Journal of Clinical Medicine, Volume 7, Issue 3, 2018.
10. Kennedy J. Particle swarm optimization, J. Kennedy, R. Eberhart // Proceedings of IEEE International Conference on Neural Networks. IV.— Perth, WA, Australia, 1995.— С.1942–1948.
11. Xin-She Y., Firefly Algorithm, Swarm Intelligence and Bio-Inspired Computation, 2013.
12. Yang X.S., Nature-Inspired Metaheuristic Algorithms, Nature-Inspired Metaheuristic Algorithms. Luniver Press, UK (2010)

© Хамамех Имад Нехадович (imad.hamameh@gmail.com).

Журнал «Современная наука: актуальные проблемы теории и практики»