791–805, 2020.
[7] J. Y. Jhang, C. J. Lin, and C. T. Lin, "Navigation control of mobile robots using an interval type-2 fuzzy controller basedon dynamic-group particle swarm optimization," *Interna- tional Journal of Control, Automation and Systems*, vol. 16, no. 10, pp. 2446–2457, 2018.
[8] F. Shamsfakhr and B. Sadeghibigham, "A neural network approach to navigation of a mobile robot and obstacle avoidance in dynamic and unknown environments," *Turkish Journal of Electrical Engineering and Computer Sciences*, vol. 25, no. 23, pp. 1629–1642, 2017.
[9] S. Muthukumaran and R. Sivaramakrishnan, "Optimal path planning for an autonomous mobile robot using dragonfly algorithm," *International Journal of Simulation Modelling*, vol. 18, no. 4, pp. 397–407, 2019.
[10] R. Zhao and H. K. Lee, "Fuzzy-based path planning for multiple mobile robots in unknown dynamic environment,"

APPLICATION OF NEURAL NETWORKS IN GIS

Seyidova Irada, Elgun Gamzaev
Azerbaijan University of Oil and Industry

Abstract
A geographic information system is a very powerful tool for managing and analyzing land use data. The integration of geographic information systems and artificial neural networks offers a mechanism to reduce the cost of landscape change analysis by reducing the amount of time spent interpreting data. Artificial neural networks (ANNs) have been proven to be useful in interpreting natural resource information. Backpropagation neural networks are one of the most common and widely used architectures. Many ANN architectures and types have been developed, many of them PC-based. Prediction of changes is based on Markov chain analysis. This process determines the state of a system based on its previous state and the likelihood of changes occurring in between. Change models serve as useful tools for studying the different mechanisms by which land use changes occur, actual design and potential future environmental impacts, and impact assessments.
Keywords: Land use change, Artificial neural networks, Geographic information systems.
Introduction: Land use change is the result of a complex interaction of many factors, including politics, governance, economics, culture, human behavior and the environment [5], [11]. Understanding how land use change occurs is critical because these anthropogenic processes can have widespread impacts on the environment, changing hydrological cycles and biogeochemical dynamics, the size and location of natural habitats such as forests [5] and species diversity [4]]. . Changes in land use can also affect local and regional economies [3]. A geographic information system (GIS) is a powerful tool for working with landscape data and modeling. ANNs are powerful tools that use a machine learning approach to quantify and model complex behaviors and patterns. ANNs are used for pattern recognition in various disciplines such as landscape classification [11], image analysis and pattern classification [13], climate forecasting [6], mechanical engineering and remote sensing [1]. The use of neural networks has increased significantly over the past few years due to advances in computational performance [14] and the increasing availability of powerful and flexible ANN software.
2. Artificial neural networks. An artificial neural network (ANN) is a computational model based on the behavior of neural networks of living organisms [9]. An artificial neural network differs from other forms of computer intelligence in that it is not rule-based like an expert system. An ANN is trained to recognize and generalize the relationship between a set of inputs and outputs. Early artificial neural networks were based on ideas about how the human brain works. In recent years, developments in ANN technology have turned it into more of an applied mathematical method, with some similarities to the human brain. Artificial neural networks retain two characteristics of the brain as their main characteristics: the ability to "learn" and generalize limited information [10]. Neural networks, both biological and artificial, use massive, interconnected simple processing elements, or neurons. In simple

terms, it functions as follows: each neuron consists of a cell body, dendrites, and an axon—a long dendrite that connects other neurons. The axon uses synapses to attach to another neuron and sends signals to the other neuron. Each neuron receives signals from more neurons. Once the total signal (of all neurons) exceeds a certain limit, the neuron becomes active and starts sending a signal. The composition of artificial neural networks is similar to a biological neuron (Fig. 1).

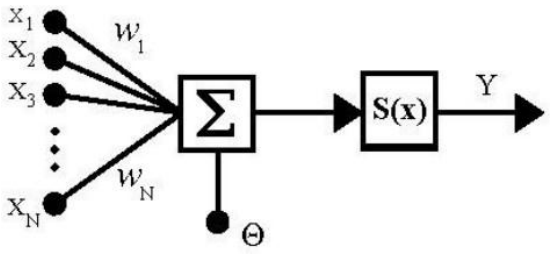

Fig. 1. Artificial neuron.

The neuron is affected by input signals X1,...,Xn, each of which has a different weight W1,...,Wn. The sum of the weights Σ is compared with a threshold value Θ and when this value is exceeded, the neuron is activated and sends a signal further using the trigger function S(x) (or transfer function). There are more types of trigger functions. The simplest of these is the jump function, the most commonly used is the sigmoid function or RBF (radial basis function). A neural network contains multiple neurons, which can be stored in one or more layers. Figure 2 shows a general network diagram with a set of inputs, one layer of neurons, and a set of outputs. In artificial neural networks, knowledge stored in the form of relationship strength weights (a numerical parameter) is modified through a process called learning using a learning algorithm. This algorithmic function, combined with a learning rule (i.e., backpropagation), is used to change the weights in the network in an orderly manner. The weights with which individual inputs act on neurons, and neurons act further on other layers of neurons and outputs, can be calculated using an iterative algorithm. The network is trained [14].

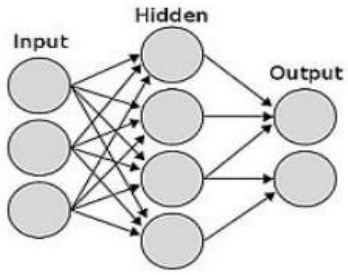

Fig.2. Neural network with one hidden layer.

Unlike most computer applications, ANNs are not “programmed” but rather “trained” to provide an acceptable answer to a specific problem. The input and output values are sent to the ANN, the interconnects in the ANN architecture are assigned initial weights, and the ANN repeatedly adjusts these interconnect weights until the ANN can successfully produce output values that match the initial values. This weighted matrix of relationships allows the neural network to learn and remember [14] The first step in using an ANN to solve a problem is to train the ANN to “learn” the relationship between inputs and outputs. This is achieved by providing the network with examples of known inputs and outputs combined with a learning rule. An ANN maps the relationship between inputs and outputs and then modifies its internal functions to determine the best relationship that the ANN can represent. The internal workings and processing of an ANN are often referred to as a “black box” with inputs and outputs. One

useful analogy to help understand the mechanics going on inside a black box is to consider a neural network as a superform of multiple regression [10]. Just as in linear regression, which finds such a connection that {y} = f{x}, the neural network during training finds some function f{x}. However, a neural network is not limited to linear functions. It finds its best function as best as possible, given the complexity of the network and without linearity constraints [10]. First, the weights are random, a calculation is carried out and an analysis follows to determine whether the network has estimated the correct result. If this is not the case, the weights are changed until the correct result is achieved. This training course is described in detail in many resources.

The basic structure of an ANN, including backpropagation ANN, consists of layers of neurons or processing elements. These layers are the input layer, output layer and hidden layer. Hidden layers are so named because they have no external connections to the network. Typically, for most applications, one hidden layer is sufficient. Having more than one hidden layer significantly increases the time required for training and testing without noticeably improving performance. Although Figure 2 shows a relatively simple neural network, more complex networks can be designed using more hidden layers and more intra-layer connections.

The input layer of a neural network presents input data to the processing neurons of the network. Data patterns, which are created by converting data from vector to raster and then to a one-dimensional column vector, are simultaneously passed from the input layer to the processing layer. The template consists of values for each input (if output training is also required) for a given location. The number of inputs depends on the type of problem being solved. The input data can be either binary or continuous.

Hidden layers receive data from the input layer. Each connection in a hidden layer has the weight or strength of the connection associated with it. Each neuron in the input layer is connected to each neuron in the hidden layer. In the same way, each neuron in the hidden layer is connected to each neuron in the next layer. The next layer could be another hidden layer or an output layer. In a feed-forward network, the flow of data goes from the input layer to the output layer through a hidden layer or layers. In a backpropagation ANN, a forward pass is followed by a backward pass, during which the weights of the connections between neurons are changed based on the error values.

The output layer produces the final results of the ANN processing. During the training phase, these outputs are compared to known outputs, errors are calculated, and interconnect weights are adjusted. Once training is complete, the output layer produces values that are returned to the GIS.

As a result of the analysis of Data Mining methods, the choice fell on the method of searching for association rules.

3. Artificial neural networks and GIS. Geographic information system (GIS) is a technology field that integrates geographic features with tabular data to map, analyze, and evaluate real-world problems. The key word in this technology is “Geography”. This means that the data (or at least some part of the data) is spatial, in other words, data that is somehow tied to places on Earth. This data is usually accompanied by tabular data known as attribute data. Attribute data can usually be defined as additional information about each of the features. An example of this would be schools. The actual locations of schools are spatial data. Additional data such as school name, level of education taught, student performance will constitute attribute data. It is the combination of these two types of data that allows GIS to be such an effective tool for solving problems through spatial analysis [2].

GIS works on many levels. At its most basic level, GIS is used as computer mapping, that is, mapping. The real strength of GIS lies in the use of spatial and statistical methods to analyze attribute and geographic information. The final result of the analysis can be derived information, interpolated information or priority information [12].Land Change Modeler is a revolutionary software for analyzing and predicting land cover change, which also includes tools for analyzing, measuring and predicting impacts on habitats and biodiversity [8]. Land Change Modeler includes a set of intelligent tools that solve the complex problems of change analysis, resource management and habitat assessment while

maintaining a simple and automated workflow. LCM analyzes the layers and as a result receives: i) a text file containing a table with the probability of change, ii) a text file containing information about the number of cells (a table of area changes that can change from one type to another for each type of overlay, iii) a raster , containing information about the probability of occurrence of each type of surface in a given location during a given period of time. The Land Change Modeler interface includes five tabs. Land Change Modeler is included with IDRISI GIS and Image Processing software and is available as a software extension for use with ESRI's ArcGIS product, which addresses the pressing issue of accelerated land conversion and the very specific needs of biodiversity conservation [8].
Multilayer perceptron is one of the most commonly used models of neural networks, the structure of which corresponds to that (Fig. 3)4. There are several layers of neurons that serve as predictors of the values of the output layer. The learning method is so-called supervised learning, which means that the algorithm is learned from the training data. This algorithm has been successfully applied in IDRISI in the MLP module and can be used for classification purposes of, for example, aerial photographs or satellite land use images.

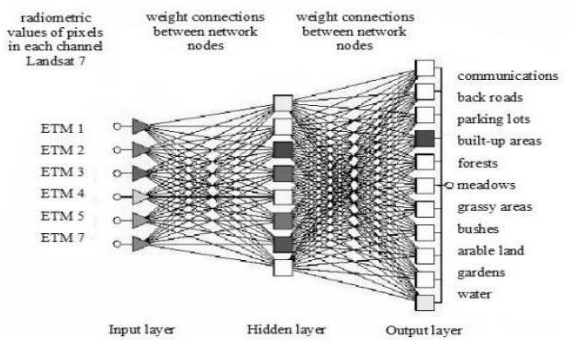

Fig.3. MLP network for land use identification.

In Fig.3 shows the network model. A darker shade of red in the hidden layer determines a higher weight of neurons on the result. The resulting classification of land use types is based on the rule of "the strongest takes all" - the land use type that was ranked as top in the output layer, compared to a given pixel. However, neural networks can be used to predict a continuous variable. In practice, they are used to predict the sequence of time lines, determine hardware settings, etc.
Self-organizing maps (SOM). This algorithm is an example of unsupervised learning. It has become very popular mainly due to the fact that it does not require training data. The network model consists of a single layer of neurons arranged in a raster shape. The principle is to arrange the raster of neurons so that the topology of the data is preserved while covering the entire space. The result of the algorithm is a trained network that can map objects to individual neurons while maintaining distance in the original space (regardless of its size). The results have been proven to be very similar to those obtained using the k-means method (cluster analysis). The algorithm can be used to search for similar images - in the neural raster they will be located close to each other.
In Fig. Figure 3 shows the network model. A darker shade of red in the hidden layer determines a higher weight of neurons on the result. The resulting classification of land use types is based on the rule of "the strongest takes all" - the land use type that was ranked as top in the output layer, compared to a given pixel. However, neural networks can be used to predict a continuous variable. In practice, they are used to predict the sequence of time lines, determine hardware settings, etc.
Self-organizing maps (SOM). This algorithm is an example of unsupervised learning. It has become very popular mainly due to the fact that it does not require training data. The network model consists of a single layer of neurons arranged in a raster shape. The principle is to arrange the raster of neurons so that the topology of the data is preserved while covering the entire space. The result of the algorithm is

a trained network that can map objects to individual neurons while maintaining distance in the original space (regardless of its size). The results have been proven to be very similar to those obtained using the k-means method (cluster analysis). The algorithm can be used to search for similar images - in the neural raster they will be located close to each other.

```python
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import StandardScaler
from keras.models import Sequential
from keras.layers import Dense

# Step 1: Generate Synthetic Data
np.random.seed(42)
num_samples = 1000
data = {
    'elevation': np.random.uniform(0, 3000, num_samples),
    'soil_type': np.random.randint(0, 5, num_samples),
    'precipitation': np.random.uniform(0, 200, num_samples),
    'land_use_type': np.random.randint(0, 5, num_samples),
    'land_use_change': np.random.randint(0, 2, num_samples)  # Binary target variable
}
df = pd.DataFrame(data)

# Step 2: Preprocess the Data
X = df.drop('land_use_change', axis=1).values
y = df['land_use_change'].values

# Split the data into training and testing sets
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)

# Standardize the data
scaler = StandardScaler()
X_train = scaler.fit_transform(X_train)
X_test = scaler.transform(X_test)

# Step 3: Build and Train the ANN Model
model = Sequential()
model.add(Dense(64, input_dim=X_train.shape[1], activation='relu'))
model.add(Dense(32, activation='relu'))
model.add(Dense(1, activation='sigmoid'))  # Binary classification

model.compile(loss='binary_crossentropy', optimizer='adam', metrics=['accuracy'])

history = model.fit(X_train, y_train, epochs=50, batch_size=10, validation_data=(X_test, y

# Step 4: Evaluate the Model
loss, accuracy = model.evaluate(X_test, y_test)
print(f'Loss: {loss}, Accuracy: {accuracy}')
```

```python
# Step 5: Visualize the Results
# Plot training & validation accuracy values
plt.figure(figsize=(12, 6))

plt.subplot(1, 2, 1)
plt.plot(history.history['accuracy'])
plt.plot(history.history['val_accuracy'])
plt.title('Model Accuracy')
plt.xlabel('Epoch')
plt.ylabel('Accuracy')
plt.legend(['Train', 'Test'], loc='upper left')

# Plot training & validation loss values
plt.subplot(1, 2, 2)
plt.plot(history.history['loss'])
plt.plot(history.history['val_loss'])
plt.title('Model Loss')
plt.xlabel('Epoch')
plt.ylabel('Loss')
plt.legend(['Train', 'Test'], loc='upper right')

plt.tight_layout()
plt.show()
```

```python
# Additional visualizations
y_pred = model.predict(X_test).flatten()
y_pred = np.where(y_pred > 0.5, 1, 0)

# Confusion matrix
from sklearn.metrics import confusion_matrix, classification_report
import seaborn as sns

conf_matrix = confusion_matrix(y_test, y_pred)
sns.heatmap(conf_matrix, annot=True, fmt='d', cmap='Blues')
plt.title('Confusion Matrix')
plt.xlabel('Predicted')
plt.ylabel('Actual')
plt.show()

# Classification report
print(classification_report(y_test, y_pred))
```

Fig.4. Coding in Python

Fig..5. Error Matrix

Model accuracy and losses

• Accuracy: The model achieved an accuracy of 52.5%, indicating that it correctly predicted land use change about half the time. This suggests that there is room to improve the model's performance.

Loss curve

• The loss curve shows the change in the loss function over iterations during the training process. The warning indicates that the model did not converge within the maximum number of iterations (50), suggesting that more iterations may be required to achieve better results.

Confusion Matrix

The confusion matrix provides a summary of the prediction results:

• True Positives (TP): 54 (correctly predicted as '1')

• True Negatives (TN): 51 (correctly predicted as '0')

• False positives (FP): 46 (wrongly predicted "1")

• False Negatives (FN): 49 (wrongly predicted “0”)

The confusion matrix indicates that the model has a relatively balanced number of correct and incorrect predictions for both classes.

Classification report

The classification report provides detailed indicators:

• Accuracy: The ratio of correctly predicted positive cases to the total number of positive cases predicted.

• Class 0: 51.0%

• Class 1: 54.0%•

Recall: the ratio of correctly predicted positive observations to all observations in the real class.

• Class 0: 52.6%

• Class 1: 52.4%

• F1-Score: Weighted average of precision and recall.

• Class 0: 51.8%

• Class 1: 53.2%

The report shows that the model has relatively balanced precision, recall and F1 scores for both classes.

Recommendations for improvement

• Increase iterations: Let the model train for more epochs to see if it converges to a better solution.

• Hyperparameter tuning: Experiment with different network architectures, learning rates, and activation functions.

4. Conclusion

The integration of geographic information systems and artificial neural networks offers a mechanism to reduce the cost of landscape change analysis by reducing the amount of time spent interpreting data. Such integration allows you to transfer the result of interpretation from a small area to a larger, naturally similar area [7]. One of the most cumbersome aspects of this type of application is transferring data from the GIS to the ANN and vice versa. Landscape data used to produce an interpretive result is most effectively processed in vector form. However, one of the most widely used ANNs, the backpropagation neural network, requires data in raster form that is organized into one-dimensional column vectors. As with all GIS projects, the first step is to convert all information into digital form. When using natural resources, the information will typically consist of maps of landscape changes. This information is usually polygonal and includes topics such as land use types. Another common type of resource for using natural resources is surfaces. This type of data can be stored as arcs with height or thickness as the arc attribute. However, for analytical purposes, surfaces are best stored in the form of grids or lattices [2]. For ANN to be used as a tool for interpreting and predicting changes, map information must be converted into patterns. These patterns consist of values for each input topic at a given location. ANNs were used to study patterns of development in a region and test the predictive ability of the model, and GIS were

used to develop spatial predictors and perform spatial analysis of the results. During ANN training, the templates must also contain the value of the received output value for each location. Once the trained ANN has produced an interpretation result, that result must be converted back into a GIS (usually in the form of a polygon coverage) to create an output map. This map, showing the result of the interpretation, can be assessed to determine whether further training is required. ANN can be considered as a very useful tool for predicting land use changes. IDRISI tools are one of the best modern GIS software that can be easily applied to the entire process of land use forecasting in various fields. Due to the variety of input data and different parameters, all results must be evaluated very carefully.

References

[1] Pechanec, V., Burian, J., & Dobešová, Z. (2014). Integration neural networks and GIS in modeling landscape changes. 14th SGEM GeoConference on Informatics, Geoinformatics and Remote Sensing, June 19-25, 2014, Vol. 1, 651-658.

[2] Peiman, M., Pradhan, B., & Melesse, A. M. (2013). Application of artificial neural networks in flood prediction. Environmental Earth Sciences, 70, 1557-1567.

[3] Rimal, B., & Zhang, L. (2017). Using artificial neural network for prediction and mapping of probability of fire occurrences in the forest region. Journal of Environmental Management, 187, 19-30.

[4] Tehrany, M. S., Lee, M. J., & Pradhan, B. (2016). Hybrid integration of an adaptive neuro-fuzzy inference system with genetic algorithm and particle swarm optimization for flood susceptibility mapping. Journal of Hydrology, 536, 347-362.

[5] Guan, Q., & Clarke, K. C. (2010). A general-purpose parallel raster processing programming library test application using a geographic cellular automata model. International Journal of Geographical Information Science, 24(5), 695-722.

[6] Yan, H., & Shaker, R. R. (2016). Modeling and simulating urban land use change by integrating GIS and artificial neural networks: A case study of Houston, Texas. International Journal of Applied Earth Observation and Geoinformation, 52, 480-489.

[7] 7Bhandari, S., & Kumar, A. (2012). Use of artificial neural networks in environmental applications. International Journal of Environmental Science and Development, 3(1), 22-28.

[8] Eastman, J. R. IDRISI Andes Guide to GIS and Image Processing. Worcester, Idrisi Production © Clark Labs, Clark University, 2006.

[9] Vieira, D. C., & Moutinho, A. F. (2019). Combining artificial neural networks and GIS fundamentals for coastal erosion prediction modeling. Sustainability, 11(4), 975.

[10] Abdar, M., Yen, N. Y., Hung, J. C., & Lai, Y. H. (2021). Bivariate models and multivariate methods for flood hazards: A case study of urban flood susceptibility in a small urban watershed. Journal of Hydrology, 593, 125803.

[11] A Comprehensive Review of Land Use and Land Cover Change Based on Knowledge Graph and Bibliometric Analyses. (2023). Land, 12(8), 1573.

[12] Pechanec, V., Janíková, V., Brus, J., Kilianová, H. Typological data in the process of landscape potential identification with using GIS. Moravian geographical reports. Vol. 17 (4), Brno, Institute of Geonics ASCR, pp. 12-24, 2009.

[13] Comparison of Random Forest, Support Vector Machines, and Neural Networks for Post-Disaster Forest Species Mapping of the Krkonoše/Karkonosze Transboundary Biosphere Reserve. (2021). Remote Sensing, 13(13), 2581.

[14] Assessing rainfall prediction models: Exploring the advantages of hybrid machine learning approaches. (2021). Environmental Modelling & Software, 139, 105038.

ИССЛЕДОВАНИЕ СТАТИСТИЧЕСКИХ МЕТОДОВ И ИНСТРУМЕНТОВ ОЦЕНКИ УРОВНЯ ИНФОРМАЦИОННОЙ БЕЗОПАСНОСТИ

Сардаров Ягуб
Алиев Камил
Азербайджанский государственный университет нефти и промышленности

Аннотация
Данное исследование рассматривает статистические методы и инструменты для оценки уровня информационной безопасности. Описаны основные концепции и подходы к оценке рисков, методы анализа уязвимостей и инцидентов безопасности. Рассмотрены современные статистические модели для прогнозирования угроз и оценки вероятности их реализации, включая методы машинного обучения, такие как классификация, регрессия, кластерный анализ и обнаружение аномалий. Исследование также охватывает инструменты для анализа данных о безопасности, включая аналитику больших данных. Описаны практические примеры применения этих методов и их преимущества. Полученные результаты могут быть полезны для специалистов в области информационной безопасности при выборе оптимальных методов и инструментов для защиты информации, также снижения рисков. Описанные подходы и модели могут помочь в разработке новых стратегий обеспечения безопасности и защите от современных киберугроз.
Ключевые слова: информационная безопасность, машинное обучение, анализ данные.
Введение. В современном информационном обществе данные являются ценным активом, и их безопасность - приоритетная задача. Угроза кибератак требует эффективных методов защиты информации [1, 2]. Одной из новых возможностей является использование методов машинного обучения. Машинное обучение автоматизирует обнаружение угроз, анализ уязвимостей и предсказание инцидентов безопасности. Эти методы обрабатывают большие объемы данных, выявляя скрытые закономерности и аномалии. Исследование фокусируется на применении методов машинного обучения для оценки информационной безопасности. Рассматриваются различные модели для обнаружения угроз, анализа уязвимостей и предсказания инцидентов, а также их использование для создания систем защиты данных и предотвращения кибератак. Особое внимание уделяется методам классификации, кластеризации, обнаружения аномалий и регрессионного анализа. Эти методы помогают разрабатывать стратегии управления рисками и улучшать защитные меры. Цель исследования - предоставить обзор текущих тенденций и перспектив использования методов машинного обучения для повышения уровня информационной безопасности, чтобы специалисты могли найти оптимальные решения для защиты информации.
Методы. Основные методы машинного обучения следующие.
Классификация. Это включает обзор алгоритмов классификации, их принцип работы и применение в оценке уровня информационной безопасности [3]. Классификация в машинном обучении относится к задаче разделения данных на категории или классы на основе их характеристик. Например, метод опорных векторов (SVM) может быть применен для определения типа кибератаки на основе характеристик сетевого трафика или других данных. Наивный Байесовский классификатор, с другой стороны, основан на теореме Байеса и может использоваться для определения категории уязвимости на основе известных признаков уязвимостей.
Кластеризация. Здесь рассматривается применение алгоритмов кластеризации для выявления групп подобных событий или данных без заранее известных категорий. Это позволяет обнаруживать аномалии и потенциальные угрозы, основываясь на сходстве между данными. Например, кластеризация может помочь выявить подозрительные схемы поведения