

# MODERN SCIENCE

*International scientific journal № 5, Vol., II. 2022.*

*Founder and publisher: «Strategic Studies Institute» LLC.*



Moscow 2022

UDC 53:51+67.02+54+316+101

LBC 72

Modern science

International scientific journal, № 5, Vol. II., 2022.

**ISSN 2414-9918**

Editor-in-chief – A.N. Zotin

*Founder:*

Scientific-information publishing center  
«Strategic Studies Institute» LLC

№ 5 Vol. II. (May)

**Modern Science** [Text]: scientific publications journal. – № 5 Vol. II. (May) / Scientific-information publishing center «*Strategic Studies Institute*»; Editor-in-chief – A.N. Zotin. – Moscow, 2022. – 348 p.

**ISSN 2414-9918**

M40

In this issue of polylingual scientific journal «Modern Science» traditionally to scientists and specialists is given the opportunity to get acquainted with the achievements of the priority directions of modern science, to demonstrate the results of their researches, to exchange experiences, to publish scientific articles that will undoubtedly contribute to a fruitful scientific work, the realization of the creative potential, the emergence of new ideas and the establishment of friendly relations and co-operation opportunities.

The journal presents research papers of Russian universities scientists, the researchers from the countries of FSU and beyond, the publication purpose of which is enrichment of the researchers and providing the actual problems of modern science to them, to inspire for further scientific research.

The journal is intended for teachers, post-graduate students, as well as for all who are interested in the socio-economic and political life of modern society.

**LBC72.3(2Ros)0**

*Whole or part reproduction or photocopying  
by whatever means the materials, published in this journal  
is permitted only with the written permission of the publisher.*

*For the accuracy of the facts set out in this journal is the responsibility of the authors.*

*Editorial opinion may not coincide with the views of the authors of articles.*

ISSN 2414-9918



9 772414 991007 >

© Authors of scientific articles, 2022

© «Strategic Studies Institute», 2022

**Титов В.А.**

ОБЕСПЕЧЕНИЕ БЕЗОПАСНОГО ИНФОРМАЦИОННОГО ОБМЕНА  
ПО НЕЗАЩИЩЕННОМУ КАНАЛУ СВЯЗИ.....335

**Улендеева Н.И.**

ПРИМЕНЕНИЕ КОДИРОВАНИЯ ДЛЯ ПЕРЕНОСА ИНФОРМАЦИИ  
С СЧИТЫВАТЕЛЯ НА АРМ ИНСПЕКТОРА ИСПРАВИТЕЛЬНОГО  
УЧРЕЖДЕНИЯ.....341

**Шакиров М.А., Галиуллина Э.Р.**

ОБЗОР БИБЛИОТЕК НА ЯЗЫКЕ PYTHON ДЛЯ МАШИННОГО ОБУЧЕНИЯ  
В ЭЛЕКТРОЭНЕРГЕТИЧЕСКОЙ ОТРАСЛИ.....344

2. Основы управления в уголовно-исполнительной системе: курс лекций: учебное пособие / Н. В. Анискина, А. М. Бобров, П. В. Голодов, О. В. Дербина. — Вологда: ВИПЭ ФСИН, 2015. — 365 с.
3. Подколзина Л. А. Аналитический обзор методов распознавания двумерных штрихкодов // Молодой исследователь Дона. 2018. №1 (10). URL: <https://clck.ru/hZUK7> (дата обращения: 13.05.2022).

**Шакиров М.А., Галиуллина Э.Р.**

Федеральное государственное бюджетное образовательное учреждение высшего образования «Казанский государственный энергетический университет»

## ОБЗОР БИБЛИОТЕК НА ЯЗЫКЕ PYTHON ДЛЯ МАШИННОГО ОБУЧЕНИЯ В ЭЛЕКТРОЭНЕРГЕТИЧЕСКОЙ ОТРАСЛИ

### *Аннотация*

*В данной работе рассмотрены библиотеки с открытым исходным кодом Python, которые могут применяться в электроэнергетической отрасли для построения умных сетей и аналитики данных. Описываются возможности библиотек с открытым исходным кодом, таких как pandas, statsmodels и scikit-learn и в обработке данных, моделировании временных рядов и машинном обучении.*

**Ключевые слова:** библиотеки, машинное обучение, электроэнергетика, прогнозирование, нейронные сети, анализ данных.

**Keywords:** Python, libraries, machine learning, SciPy, electric power, forecasting, neural networks, data analysis.

Машинное обучение – это наука о программировании, с помощью которого компьютер способен обучаться на основе различных видов данных. Более общее определение, данное Артуром Самуэлем, звучит так: «Машинное обучение – это область исследований, которая дает компьютерам способность обучаться без явного программирования». Машинное обучение обычно используется для решения различных жизненных проблем.

В прежние времена люди выполняли задачи машинного обучения, вручную кодируя все алгоритмы, математические и статистические формулы. Это делало процесс трудоемким, утомительным и неэффективным. Но в наши дни этот процесс стал очень простым и эффективным по сравнению со старыми временами благодаря различным библиотекам, фреймворкам и модулям Python. Сегодня Python является одним из самых популярных языков программирования для решения этой задачи, он заменил многие языки в отрасли, и одной из причин этого является его обширная коллекция библиотек.

Нейросетевые технологии сегодня являются одним из основных векторов развития в промышленности и экономике. Нейросетевые технологии являются видом машинного обучения, которое активно развивается в программной среде. Одним из ведущих языков программирования в области машинного обучения является язык Python.

Программная среда языка Python является доминирующей платформой для прикладного машинного обучения. Основным аргументом в пользу использования Python является возможность прогнозирования временных рядов является то, что это язык

программирования общего назначения, который можно использовать как для экспериментов, так и для промышленности (в том числе энергетической отрасли). Его легко изучать и использовать, прежде всего потому, что язык ориентирован на удобочитаемость. Python – динамичный язык, хорошо подходящий для интерактивной разработки и быстрого создания прототипов.

Python также широко используется для машинного обучения и анализа данных благодаря библиотечной поддержке. Для временных рядов в нем есть такие библиотеки, как NumPy, pandas, SciPy, scikit-learn, statsmodels, Matplotlib, datetime, Keras и многие другие.

SciPy – это основанная на Python среда программного обеспечения с открытым исходным кодом для математики, естественных наук и инженерии, содержит различные модули для оптимизации, линейной алгебры, интеграции и статистики. [1]. Некоторые из основных пакетов включают NumPy (базовый пакет для работы с n-мерными массивами) [2], Matplotlib (комплексная библиотека для двумерного черчения) [3], IPython (расширенная интерактивная консоль), SymPy (библиотека для символьной математики) и pandas (библиотека для структурирования и анализа данных). Фрагмент программного кода библиотеки SciPy для работы с графическим изображением представлен ниже:

```
from scipy.misc import imread, imsave, imresize
img = imread('D:/Programs / cat.jpg') # path of the image
print(img.dtype, img.shape)
img_tint = img * [1, 0.45, 0.3]
imsave('D:/Programs / cat_tinted.jpg', img_tint)
img_tint_resize = imresize(img_tint, (300, 300))
imsave('D:/Programs / cat_tinted_resized.jpg', img_tint_resize)
```

NumPy – это фундаментальный пакет для научных вычислений на Python [2]. Он содержит, среди прочего, следующие элементы: мощный объект n-мерного массива, сложные (транслирующие) функции, инструменты для интеграции кода на C/C++ и Fortran, преобразования Фурье и случайных чисел [2]. Это особенно полезно для линейной алгебры, преобразования Фурье и возможностей случайных чисел. Высокопроизводительные библиотеки, такие как TensorFlow, используют NumPy внутренне для манипулирования тензорами. Листинг кода библиотеки NumPy для работы с математическими операциями представлен ниже:

```
import numpy as np
x = np.array([[1, 2], [3, 4]])
y = np.array([[5, 6], [7, 8]])
v = np.array([9, 10])
w = np.array([11, 12])
print(np.dot(v, w), "\n")
print(np.dot(x, v), "\n")
print(np.dot(x, y))
```

Matplotlib – это библиотека построения графиков на языке Python [3]. Matplotlib можно использовать в скриптах Python, оболочках Python и IPython, блокноте Jupyter, серверах веб-приложений и инструментах графического интерфейса пользователя [3]. Matplotlib полезен для построения графиков, гистограмм, спектров мощности, гистограмм, графиков ошибок, диаграмм рассеяния и так далее с помощью всего нескольких строк кода [3]. Код программы библиотеки Matplotlib для работы с визуализацией данных показан ниже:

```
import matplotlib.pyplot as plt
import numpy as np
x = np.linspace(0, 10, 100)
plt.plot(x, x, label='linear')
plt.legend()
```

```
plt.show()
```

Кроме того, существуют три библиотеки более высокого уровня SciPy [1], которые предоставляют ключевые возможности для прогнозирования временных рядов на Python, а именно pandas, statsmodels и scikit-learn для обработки данных, моделирования временных рядов и машинного обучения, соответственно.

Pandas – это библиотека с открытым исходным кодом, предоставляющая высокопроизводительные, простые в использовании структуры данных и инструменты анализа данных для языка программирования Python [4]. Python уже давно отлично подходит для обработки и подготовки данных, но в меньшей степени для их анализа и моделирования. Pandas помогает заполнить этот пробел, позволяя выполнять весь рабочий процесс анализа данных на Python без необходимости переходить на более специфический язык, такой как R. Фрагмент программного кода библиотеки Pandas для работы с данными представлен ниже:

```
import pandas as pd
data = {"country": ["Brazil", "Russia", "India", "China", "South Africa"],
        "capital": ["Brasilia", "Moscow", "New Delhi", "Beijing", "Pretoria"],
        "area": [8.516, 17.10, 3.286, 9.597, 1.221],
        "population": [200.4, 143.5, 1252, 1357, 52.98] }
data_table = pd.DataFrame(data)
print(data_table)
```

В Pandas не реализована значительная функциональность моделирования за пределами линейной и панельной регрессии; для этого необходимо использование библиотек statsmodels и scikit-learn, описанных ниже.

Statsmodels – это модуль Python, который предоставляет классы и функции для оценки множества различных статистических моделей, а также для проведения статистических тестов и исследования статистических данных [5]. Для каждого объекта доступен обширный список статистических результатов. Результаты проверяются с помощью существующих статистических пакетов, чтобы убедиться в их верности. Statsmodels поддерживает определение моделей с использованием формул в стиле R и фреймов данных pandas. Ниже представлен простой пример с использованием обычных наименьших квадратов:

```
import numpy as np
import statsmodels.api as sm
import statsmodels.formula.api as smf
dat = sm.datasets.get_rdataset("Guerry", "HistData").data
results = smf.ols('Lottery ~ Literacy + np.log(Pop1831)', data=dat).fit()
```

Scikit-learn – это простой и эффективный инструмент для поиска и анализа данных [6]. Эта библиотека реализует ряд алгоритмов машинного обучения, предварительной обработки, кросс-валидации и визуализации, используя единый интерфейс [6]. Эта библиотека построения 2D-графиков, используемая для создания 2D-графиков и графиков. Модуль с именем pyplot упрощает для программистов построение графиков, поскольку он предоставляет функции для управления стилями линий, свойствами шрифтов, осями форматирования и т.д. Он предоставляет различные виды графиков и графиков для визуализации данных, а именно гистограммы, диаграммы ошибок, столбчатые чаты и т. д. Она построена на NumPy, SciPy и Matplotlib Scikit-learn ориентирован на моделирование данных машинного обучения. Программный код библиотеки Scikit-learn для анализа данных представлен ниже:

```
from sklearn import datasets
from sklearn import metrics
from sklearn.tree import DecisionTreeClassifier
dataset = datasets.load_iris()
model = DecisionTreeClassifier()
```

```
model.fit(dataset.data, dataset.target)
print(model)
expected = dataset.target
predicted = model.predict(dataset.data)
print(metrics.classification_report(expected, predicted))
print(metrics.confusion_matrix(expected, predicted))
```

Он не занимается загрузкой, обработкой, манипулированием и визуализацией данных. По этой причине специалисты по исследованию данных обычно сочетают использование scikit-learn с другими библиотеками, такими как NumPy, pandas и Matplotlib, для обработки данных, предварительной обработки и визуализации.

Таким образом, рассмотренные библиотеки языка Python позволяют разрабатывать решения промышленных задач с помощью машинного обучения и нейросетевых технологий. Применяя эти средства в электроэнергетической отрасли, будет возможна реализация анализа данных для прогнозирования нагрузки потребителей.

### **Литература**

1. Numpy and Scipy Documentation // SciPy.org URL: [docs.scipy.org/doc/](https://docs.scipy.org/doc/) (дата обращения: 07.03.2022).
2. NumPy documentation // NumPy URL: [numpy.org/devdocs/](https://numpy.org/devdocs/) (дата обращения: 07.03.2022).
3. Documentation // Matplotlib URL: <https://clck.ru/hZWzL> (дата обращения: 07.03.2022).
4. Pandas documentation // Pandas URL: [pandas.pydata.org/docs/](https://pandas.pydata.org/docs/) (дата обращения: 07.03.2022).
5. Statistical models, hypothesis tests, and data exploration // Statsmodels URL: [pandas.pydata.org/docs/](https://pandas.pydata.org/docs/) (дата обращения: 07.03.2022).
6. User Guide // Scikit-learn URL: <https://clck.ru/H2noW> (дата обращения: 07.03.2022).

Scientific edition

# **Modern Science**

Format 170x24/8. Typeface Tahoma.  
Conventional printed sheets 7,9. Circulation 200 copies. Order 02.