Практическая работа №3. Анализ временных рядов и прогнозирование данных

Цель работы

Изучить методы анализа временных рядов, визуализации изменений показателей во времени и построения модели прогнозирования.

Датасет для работы

Используется набор данных Air Passengers Dataset.

Источник: Air Passengers Dataset

Файл для работы: `AirPassengers.csv`

Задание 1. Загрузка и анализ данных Пример

выполнения

```
import pandas as pd

# Загрузка данных
data = pd.read_csv("AirPassengers.csv")

# Просмотр первых строк
print(data.head())
```

Объяснение

Метод `read_csv()` загружает данные в DataFrame.

Метод `head()` выводит первые строки таблицы.

```
# Информация о данных
print(data.info())
```

Объяснение

Метод `info()` показывает:

- количество строк;
- типы данных; • наличие пропусков.

Задание 2. Преобразование даты

Пример выполнения

```
# Преобразование столбца Month в дату data['Month']
= pd.to_datetime(data['Month'])

# Установка даты как индекса
data.set_index('Month', inplace=True)

print(data.head())
```

Объяснение

Функция `to_datetime()` преобразует текстовый формат даты в формат datetime.

Метод `set_index()` делает дату индексом таблицы.

Задание 3. Визуализация временного ряда Пример

выполнения

```
import matplotlib.pyplot as plt

# Построение графика
plt.plot(data.index, data['#Passengers'])

plt.title("Количество авиапассажиров")
plt.xlabel("Год") plt.ylabel("Пассажиры")
 plt.show()
```

Объяснение

Линейный график позволяет увидеть:

- общий тренд;
- сезонные изменения;

- рост или снижение показателей.

Задание 4. Скользящее среднее

Пример выполнения

```
# Скользящее среднее за 12 месяцев
rolling_mean = data['#Passengers'].rolling(window=12).mean()

# Построение графика
plt.plot(data['#Passengers'], label='Исходные данные')
plt.plot(rolling_mean, label='Скользящее среднее')

plt.legend()
 plt.show()
```

Объяснение

Скользящее среднее сглаживает временной ряд и помогает выявить тренд.

Параметр `window=12` означает анализ за 12 месяцев.

Задание 5. Построение модели прогнозирования Пример выполнения

```
from sklearn.linear_model import LinearRegression
import numpy as np

# Создание признаков
data['Time'] = np.arange(len(data))

X = data[['Time']] y =
data['#Passengers']

# Создание модели model =
LinearRegression()

# Обучение модели model.fit(X,
y)
 #
Прогноз
predictions = model.predict(X)
```

Объяснение

Создается простой числовой признак времени.

Модель линейной регрессии прогнозирует изменение количества пассажиров.

Задание 6. Визуализация прогноза

Пример выполнения

```
# График прогноза
plt.plot(data['#Passengers'], label='Реальные данные')
plt.plot(predictions, label='Прогноз')

plt.legend()
 plt.show()
```

Объяснение

График позволяет сравнить реальные значения и прогноз модели.

Самостоятельные задания

1. Построить график изменения количества пассажиров по годам
2. Найти минимальное и максимальное значение
3. Рассчитать среднее количество пассажиров
4. Построить прогноз на следующие 12 месяцев
5. Изменить размер окна скользящего среднего
6. Построить более точную модель прогнозирования

Контрольные вопросы

1. Что такое временной ряд?
2. Что такое тренд?
3. Что такое сезонность?
4. Для чего используется скользящее среднее?
5. Что показывает линейный график?
6. Для чего используется метод `rolling()`?

7. Что делает функция `to_datetime()`?
8. Что такое прогнозирование данных?
9. Какие задачи решают временные ряды?
10. Что такое линейная регрессия?
11. Для чего нужна визуализация данных?
12. Что такое временной индекс?
13. Какие библиотеки используются для анализа временных рядов?
14. Что такое модель машинного обучения? Что показывает прогноз модели?