МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ

РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное бюджетное образовательное учреждение высшего образования

«Московский государственный университет геодезии и картографии»

(МИИГАиК)

Факультет геоинформатики и информационной безопасности

Кафедра прикладной информатики

Отчет по Индивидуальной работе

«Прогнозирование летальности от Covid-19»

Проверил:

Стрельцов А.С.

Выполнил:

Студент группы 2022-ФГиИБ-ПИабпд-1м

Кондратьев П.В.

Москва, 2023

Оглавление

Определение цели

На сегодняшний день существует несколько подходов к прогнозированию эпидемиологических ситуаций, различающихся по используемым моделям:

1) Агентно-ориентированные модели
    a. Модели на основе анализа социальных связей
    b. Модели прогнозирования на основе данных мобильных технологий
2) Модели машинного обучения
    a. Модели классических методов машинного обучения
    b. Модели глубокого обучения

Агентно-ориентированные модели исходят из сведений об отдельных агентах, взаимодействующих между собой. Эти наблюдения сводятся исключительно к поиску наиболее качественной вероятностной модели, дающей наилучший результат при запуске симуляции. В качестве данных могут быть использованы массивы информации, накапливаемые мобильными операторами о перемещении пользователей или агрегированная информация о круге общения потенциального зараженного.

Однако такие модели предполагают использование данных, зачастую доступных или внутри компании, где проводится исследование, или государственным структурам. Другие исследователи, напрямую не аффилированные с указанными выше организациями, пользуются открытой статистикой, обобщающей индивидуальное поведение людей.

Такие большие данные поддаются статистической обработке и позволяют определять глобальные тренды. В частности, при помощи моделей машинного обучения можно прогнозировать летальность вируса.

Одним из самых больших вызовов современности была пандемия Covid-19. Накопленные массивы наблюдений, обновляемые до сих пор, позволяют с высокой точностью производить исследования динамики

распространения вирусных эпидемий в глобальном мире, совершенствуя методы и подходы.

Как было указано ранее, планируется разработать модель, предсказывающую для разных стран уровень смертности от Covid-19, учитывая такие факторы, как число активных случаев заражения и количество вакцинированного населения.

Целью работы является прогнозирование количества летальных исходов от Covid-19.

Сбор и подготовка данных

Существует множество организаций, публикующих отчеты о распространении Covid-19 и сведения о вакцинации населения. Для агрегированная поступающей информации были созданы подразделения при соответствующих научно-исследовательских организациях и в международных структурах. Также были созданы добровольные объединения и сообщества, занятые вопросом верификации данных.

Наиболее яркими представителями, каждой из структур ялвляются:

- Университет Джонса Хопкинса, создавший свое API для получения актуальной информации
- Всемирная организация здравоохранения (ВОЗ), также имеющая свою разветвленную структуру сайта со множеством разделов
- Worldometer – красфаундинговый проект, занимающийся вопросами обобщения социально-экономических показателей по странам мира. С 2020 года имеет раздел отслеживанию данных о COVID-19.

Команда Worldometer состоит из разработчиков, исследователей и добровольцев из разных стран, которые стремятся предоставить мировую статистику в удобном формате для широкой аудитории. Этот проект поддерживается небольшой независимой цифровой медиа-компанией из США.

Поскольку данные о вакцинации получаются централизованного через органы государственного управления, только ВОЗ предоставляет эту информацию публично в виде статистики.

Данные же о летальности, активных случаях заражения публикуются куда более охотно, поэтому являются более доступными. Worldometer обобщает документы Университета Джонса Хопкинса и прочих организаций.

Для парсинга таблицы с Worldometer используется скрипт Листинга 1. Ввиду спада эпидемиологической угрозы показатели обновляются 1 раз в неделю по воскресениям. Исторические данные также предоставляются для указанных периодов.

All | Europe | North America | Asia | South America | Africa | Oceania

#	Country, Other	Total Cases	New Cases	Total Deaths	New Deaths	Total Recovered	New Recovered	Active Cases	Serious, Critical	Tot Cases/ 1M pop	Deaths/ 1M pop	Total Tests	Tests/ 1M pop	Population
	World	701,413,950	+4,081	6,966,045	+35	672,369,810	+9,641	22,078,095	36,482	89,985	893.7			
1	USA	110,342,700		1,191,138		108,065,982		1,085,580	2,228	329,573	3,558	1,186,622,338	3,544,216	334,805,269
2	India	45,019,214		533,402		N/A	N/A	N/A	N/A	32,005	379	935,879,495	665,334	1,406,631,776
3	France	40,138,560		167,642		*39,970,918*		0		612,013	2,556	271,490,188	4,139,547	65,584,518
4	Germany	38,784,494		180,657		38,240,600		363,237	N/A	462,361	2,154	122,332,384	1,458,359	83,883,596
5	Brazil	38,210,864		708,638		36,249,161		1,253,065	N/A	177,433	3,291	63,776,166	296,146	215,353,593
6	S. Korea	34,571,873		35,934		*34,535,939*		0		673,523	700	15,804,065	307,892	51,329,899
7	Japan	33,803,572		74,694		N/A	N/A	N/A	N/A	269,169	595	100,414,883	799,578	125,584,838
8	Italy	26,671,165		195,139		26,257,742		218,284	281	442,581	3,238	279,240,685	4,633,718	60,262,770
9	UK	24,812,582		232,112		*24,580,470*		0	N/A	362,239	3,389	522,526,476	7,628,357	68,497,907

Рис. 1 Пример таблицы

Листинг 1

```
html = requests.get('https://www.worldometers.info/coronavirus/').text
html_soup = BeautifulSoup(html, 'html.parser')
rows = html_soup.find_all('tr')
def extract_text(row, tag):
    element = BeautifulSoup(row, 'html.parser').find_all(tag)
    text = [col.get_text() for col in element]
    return text
heading = rows.pop(0)
heading_row = extract_text(str(heading), 'th')[1:9]

with open('corona_latest.csv', 'w', encoding='utf-8') as store:
    Store = csv.writer(store, delimiter=',')
    Store.writerow(heading_row)
    for row in rows:
        test_data = extract_text(str(row), 'td')[1:9]
        Store.writerow(test_data)
```

API сайта ВОЗ ввиду обилия публикующейся информации, крайне сложен в применении.

URL для работы с интерфейсом формируется следующим образом:

http://HOST[:PORT]/api/v3/CONTROLLER[/CODE][?QUERY_PARAMETER]

Чтобы получить подробную информацию о вхождении, нужно указать его код. Указав нужные параметры запроса, можно выбрать язык ответа, применить к данным фильтр (в списке вхождений или данных внутри вхождения) или выбрать части вхождения, которые необходимо получить.

- HOST: домен веб-сервиса. Для доступа к API нужно указать домен http://dw.euro.who.int.
- PORT: номер порта веб-сервиса. Можно не указывать или указать порт HTTP по умолчанию (80).
- CONTROLLER: код контроллера для получения данных. Возможные контроллеры:
    - countries
    - country_groups
    - data_sets
    - measures
    - version
    - export_metadata

- export_data_set
- classifications
- categories

  Каждый контроллер выдает разные виды данных.

- CODE: код нужного вхождения.
- QUERY_PARAMETERS: опциональные параметры запроса, которые могут быть применимы только к некоторым запросам. Возможные параметры:
  - lang – язык ответа. EN – английский, RU – русский. По умолчанию выдается ответ на английском.
  - filter – может использоваться для применения фильтра к показателям или к фактам внутри показателя. Значение фильтра – список токенов, разделенных точками с запятой, в формате ATTRIBUTE:CODES_LIST. CODES_LIST – это:
    - список разделенных запятыми кодов,
    - " * " — любое установленное значение,
    - " $blank " — неустановленное значение,
    - диапазоны вида 1999-2004 (если значение поля ATTRIBUTE является числовым),
    - " ~[year] " или " [year]~ " — фильтр по ближайшему доступному году, где знак " ~ "означает направление поиска данных, например ~2001 — при отсутствии данных за 2001 г. выводятся данные за ближайший предшествующий год.

    К кодам в рамках одного свойства применяется логическое ИЛИ. К разным свойствам применяется логическое И.

    Атрибут COUNTRY поддерживает коды COUNTRY_GRP, так как значения CODES_LIST означают страны, относящиеся к конкретным группам стран.

  - output – используется при контроллерах measures и categories, чтобы выбрать те части объекта, которые необходимо получить. Возможные значения для measures: data, metadata, attributes, notes и classifications;

для categories: measures и subtree. Можно задать либо одно из этих значений, либо список значений, разделенный запятыми.

- updated_since – позволяет получить информацию о показателях и источниках данных, которые были изменены после той или иной даты.
- dataState – позволяет скачивать коды и метки показателей и классификаций, которые не опубликованы в Data Warehouse.
- format – задает формат экспортируемых файлов. Используется только при экспорте и может иметь значение xlsx либо csv.

Потратив некоторое время на поиск нужного документы, был получен следующий скрипт:

Листинг. 2

```
url = 'https://covid.ourworldindata.org/data/owid-covid-data.csv'

data = pd.read_csv(url)

# Фильтрация и выбор колонок о вакцинации, включая дату
vaccination_data = data.loc[:, ['date', 'location', 'total_vaccinations',
'people_vaccinated', 'people_fully_vaccinated', 'total_boosters']]

print(vaccination_data.head())
```

Полученные данные обновляются, примерно, раз в полгода. К сожалению, ВОЗ не накладывает на себя обязательств по срокам актуализации информации. Также существует ряд проблем с получения прошлых версий документов. Данные крайне разрознены и для разных стран на указанные даты порой отсутствуют целевые значения.

В качестве уже подготовленных исторических данных с предложенных сайтов были взяты верифицированные выборки с Kaggle для предварительного обучения моделей:

1) [2019 Coronavirus dataset (January – February 2020)] (https://www.kaggle.com/datasets/brendaso/2019-coronavirus-dataset-01212020-01262020)
2) [COVID-19 Dataset] (https://www.kaggle.com/datasets/imdevskp/corona-virus-report)
3) [Novel Corona Virus 2019 Dataset] (https://www.kaggle.com/datasets/sudalairajkumar/novel-corona-virus-2019-dataset)
4) [COVID-19 World Vaccination Progress] (https://www.kaggle.com/datasets/gpreda/covid-world-vaccination-progress)

Выбор функций и моделей

Задача прогнозирования данных о летальности основывается на трех показателях: код страны, количество случаев заражения и количество вакцинированного населения.

Прогнозирование количества вакцинированного населения производится при помощи модели глубокого обучения на основе LSTM-модели.

LSTM (Long Short-Term Memory) - это вид рекуррентных нейронных сетей, спроектированных для работы с последовательными данными и учета долгосрочных зависимостей в информации.

Основная идея LSTM заключается в способности сохранять и использовать информацию на протяжении длительного временного интервала, благодаря механизму "ворот", который позволяет модели решать проблему затухания или взрывного увеличения градиентов при обучении на длинных последовательностях.

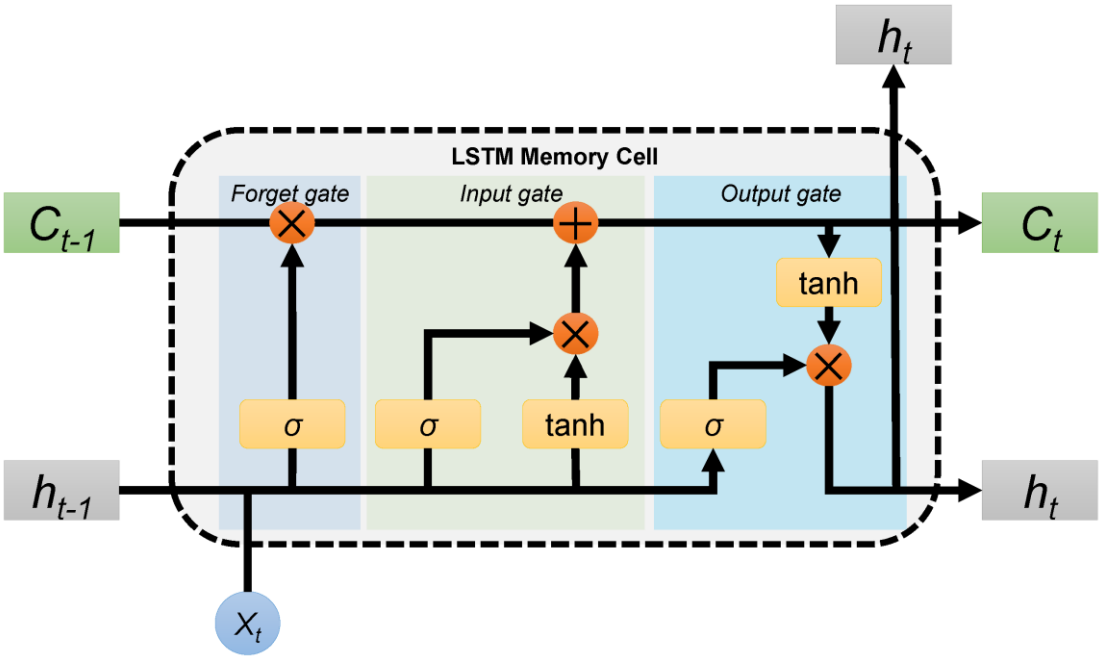

Рис. 2 Схема LSTM-модели

LSTM состоит из нескольких блоков, которые могут обрабатывать, хранить и передавать информацию. Эти блоки включают в себя "ворота" - механизмы, контролирующие поток информации внутри сети. В частности, LSTM имеют три основных типа ворот: входной ворот, забывания и выходной. Они решают проблему затухания градиента, обеспечивают долгосрочную память и контролируют, какая информация будет передаваться или забываться.

Листинг 3

```
# Подготовка данных для LSTM
    X, y = [], []
    for i in range(len(scaled_data) - 10):  # Примерное окно данных
        X.append(scaled_data[i:i+10])
        y.append(scaled_data[i+10])
    X, y = np.array(X), np.array(y)


# Создание модели LSTM
    model = Sequential()
    model.add(LSTM(units=50, return_sequences=True, input_shape=(X.shape[1],
1)))
    model.add(LSTM(units=50))
    model.add(Dense(units=1))
    model.compile(optimizer='adam', loss='mean_squared_error')
```

Для каждой страны, на которую достаточно данных (не менее 10 не нулевых значений), обучается модель LSTM. Таким образом, было создано 192 файла с весами.

Для прогнозирования количество случаев заражения используется модель SARIMAX, предполагая определенную сезонность заболевания, начиная с 2021 года.

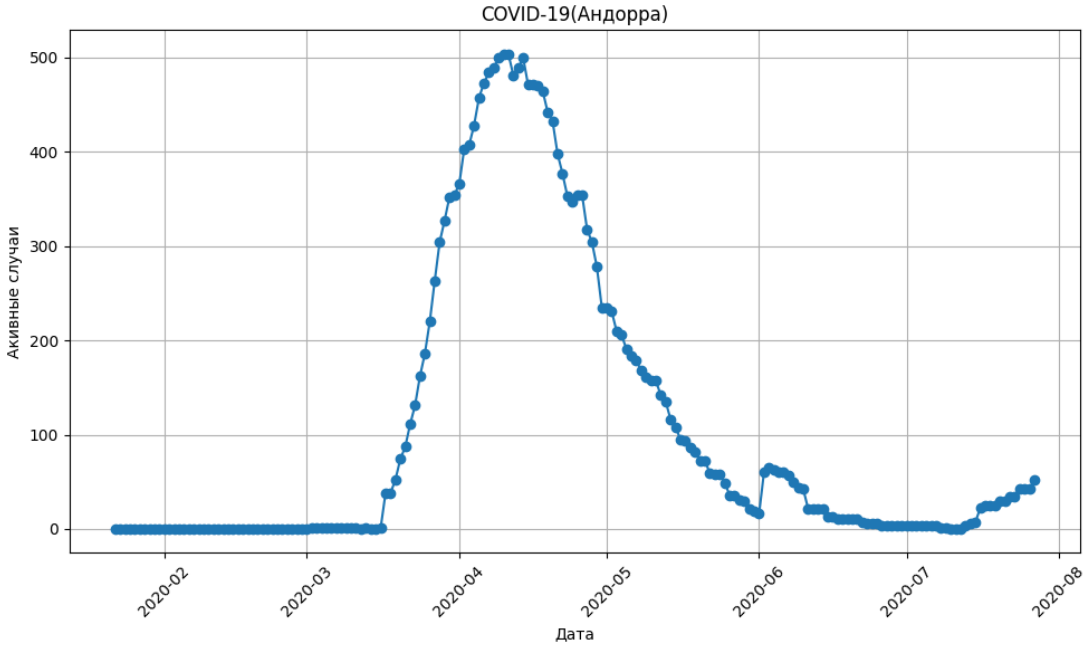

Рис. 3 Данные по активных случаям заражения для выбранной страны

В целом ряды по итогам показали свою стационарность, что свидетельствует о стабилизации эпидемиологической ситуации в большинстве стран мира.

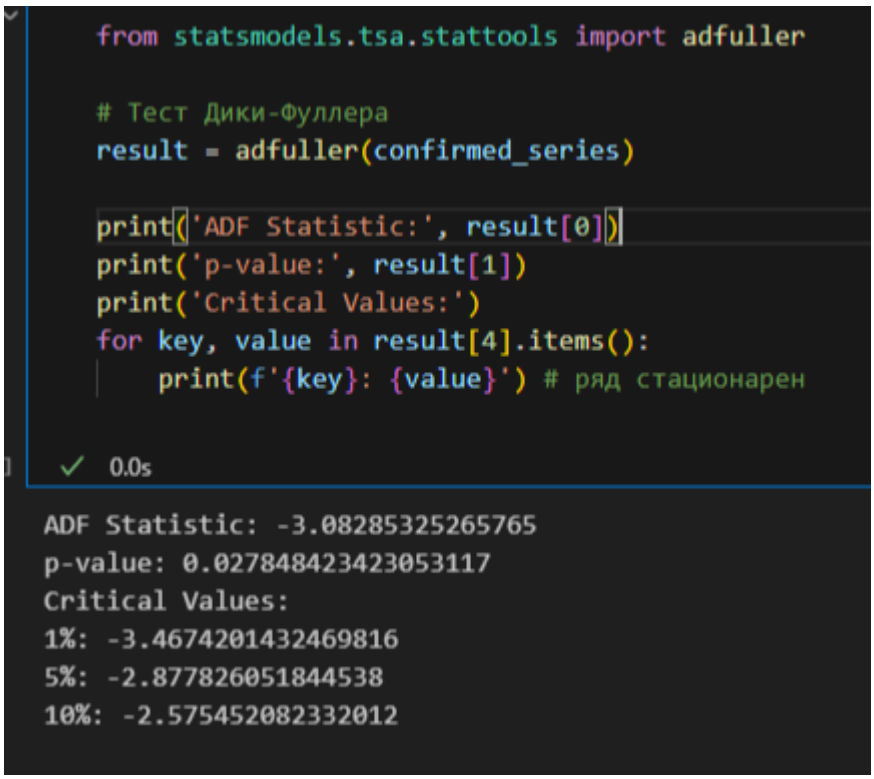

Рис. 4 Результаты теста Дики-Фуллера

Итоговая смертность на основе данных о вакцинации и активных случаях производилась посредствам регрессионной модели.

Обратим внимание, что в качестве входных параметров регрессионной модели используются:

- День
- Месяц
- Год
- Количество вакцинированных
- Шифр страны
- Количество активных случаев

Разработка аналитической базы

Для работы с наборами данных производился разведочный анализ данных. Исследовались механизмы консолидации двух рассмотренных источников. Для создания дашборда использовался plotly, который сохранялся в html файл.

Рис. 5 Информационный элемент общемировой статистики

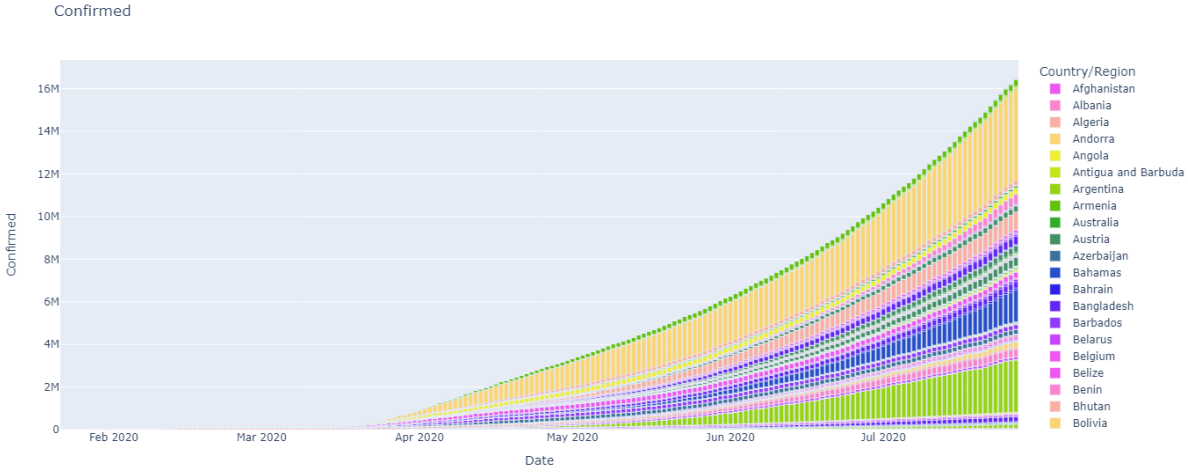

Рис. 6 Информационный элемент количества заражений по странам

Поскольку каждая страна была закодирована при помощи своего названия, было возможно использовать ***px.choropleth***, что существенно облегчило задачу визуализации данных. Таким образом, учитывалась, как пространственная специфика, так и временная динамика.

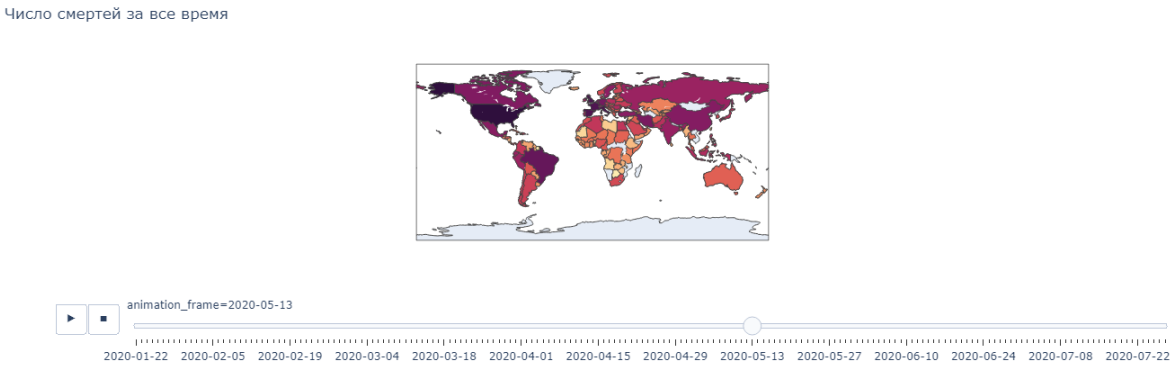

Рис. 7 Число смертей от Covid-19

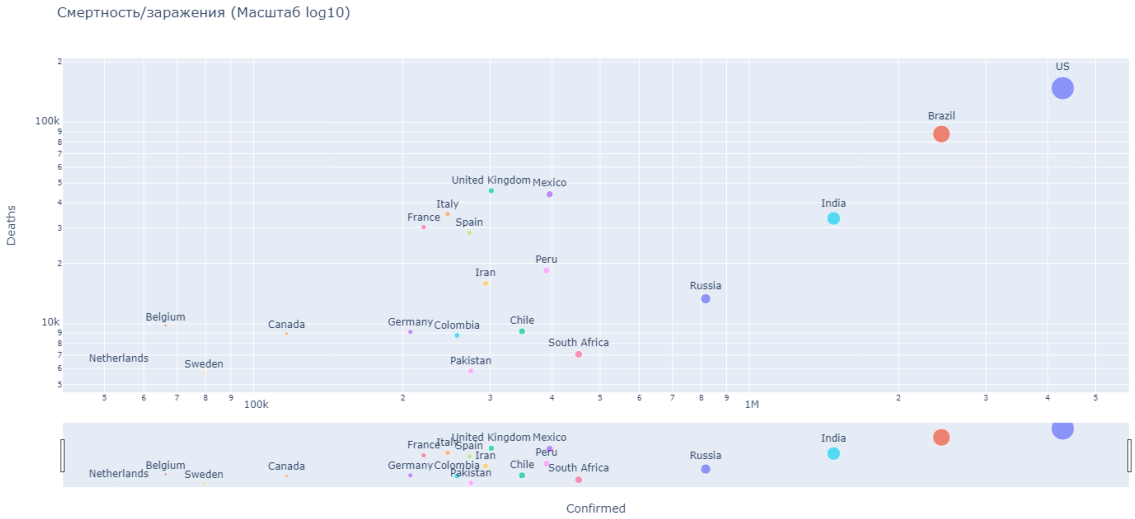

Рис. 8 График для анализа корреляции между заражением и летальностью по страна на 100к населения.

Ряд проблем был связан с использованием данных ВОЗ, поскольку эта организация использует свой способ кодирования стран. Все страны разбиты на 6 регионов, условно по территориальному признаку (см. рис 9)

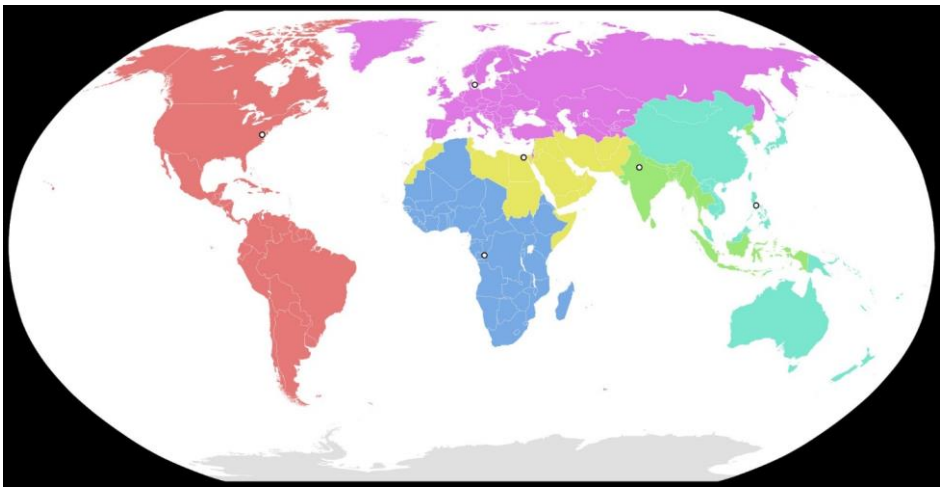

Рис. 9 Регионы и региональные бюро ВОЗ:

- Юго-Восточной Азии (Азиатское)
- Западной части Тихого океана (Тихоокеанское)
- Восточно-Средиземноморское (Средиземноморское)
- Европейское
- Американское
- стран Африки южнее Сахары (Африканское)

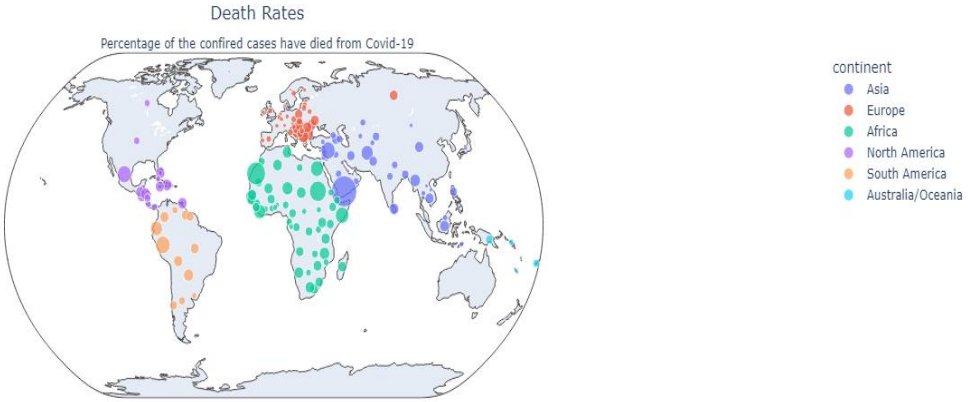

Рис. 10 Визуализация по геотегам ВОЗ

Для объединения датастов необходимо было переименовать ряд стран:

Заменим

- 'Czechia' == "Czech Republic"
- 'Isle of Man' == "Isle Of Man"
- 'United Kingdom' == "UK"
- 'United States' == "USA"
- 'Northern Cyprus' == "Cyprus"

Исключим

- England
- Wales
- Scotland
- Northern Ireland

Затем аналогично Рис. 7 создадим виджет для отображения пространственно-временного ряда Вакцинации по странам

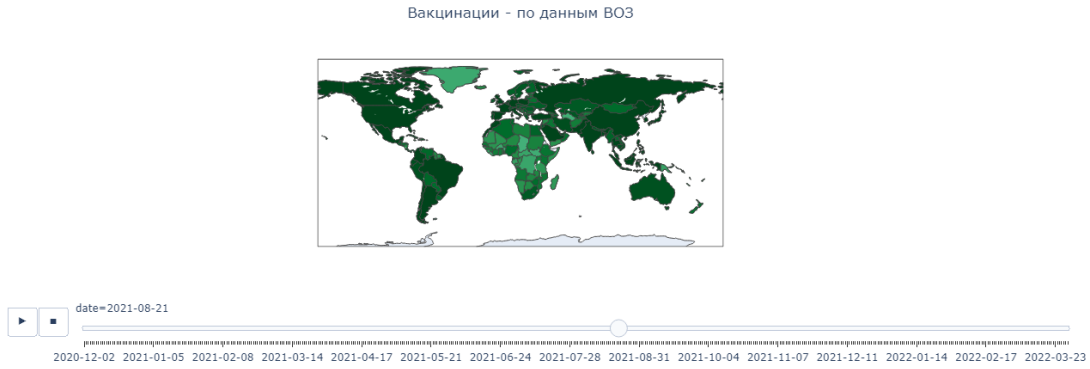

Рис. 11 Вакцинация (2020-12-02 - 2022-03-23)

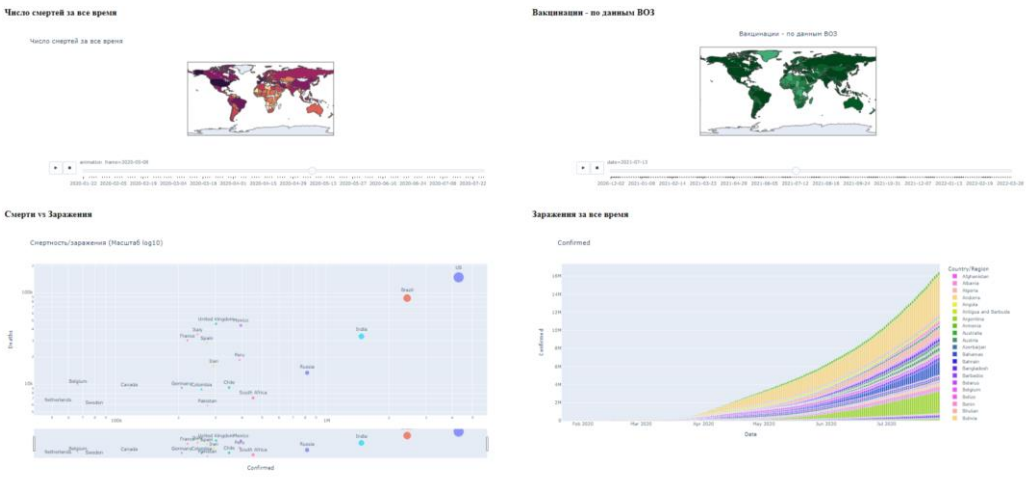

Рис. 12 Итоговый дашборд с объединёнными элементами интерфейса

Проверка и оценка моделей

Проверка моделей осуществлялась путем оценки средней абсолютной квадратической ошибки, как результирующий итог производилась оценка среднего арифметического всех ошибок. Пример обучения и оценки моделей SARIMAX приведен в Листинге 4.

Листинг 4

```
import pandas as pd
from statsmodels.tsa.statespace.sarimax import SARIMAX
from sklearn.metrics import  mean_absolute_error


# Списки для сохранения значений MSE и MAE
mae_values = []

# Получение списка уникальных стран
unique_countries = full_grouped['Country/Region'].unique()

# Цикл для прогнозирования активных случаев SARIMAX для каждой страны
for country in unique_countries:
    country_data = full_grouped[full_grouped['Country/Region'] == country]

    #
    country_data = country_data[['Date', 'Active']]
    country_data['Date'] = pd.to_datetime(country_data['Date'])
    country_data.set_index('Date', inplace=True)

    # Разделение данных на обучающий и тестовый наборы
    train = country_data.iloc[:-30]  # Последние 30 дней оставляем для теста
    test = country_data.iloc[-30:]

    # Обучение модели SARIMAX
    order = (1, 1, 1)
    seasonal_order = (1, 1, 1, 12)

    model = SARIMAX(train['Active'], order=order,
seasonal_order=seasonal_order)
    result = model.fit()


    forecast = result.predict(start=test.index[0], end=test.index[-1])

    # Оценка модели для каждой страны
    mae = mean_absolute_error(test['Active'], forecast)


    mae_values.append(mae)

# Вычисление средних значений MSE и MAE для всех стран
avg_mae = sum(mae_values) / len(mae_values)

print(f"Средняя MAE для всех моделей: {avg_mae}")
```

Таблица. 1 Результаты обучения

Тип модели	Целевой признак	Ср. MAE
SARIMAX	Количества активных случаев	3358
LSTM	Количество вакцинированного населения	27263
Регрессионная модель	Летальность	23.67

Итоговый прогноз по Польше, полученный регрессионной моделью, представлен на рис. 13.

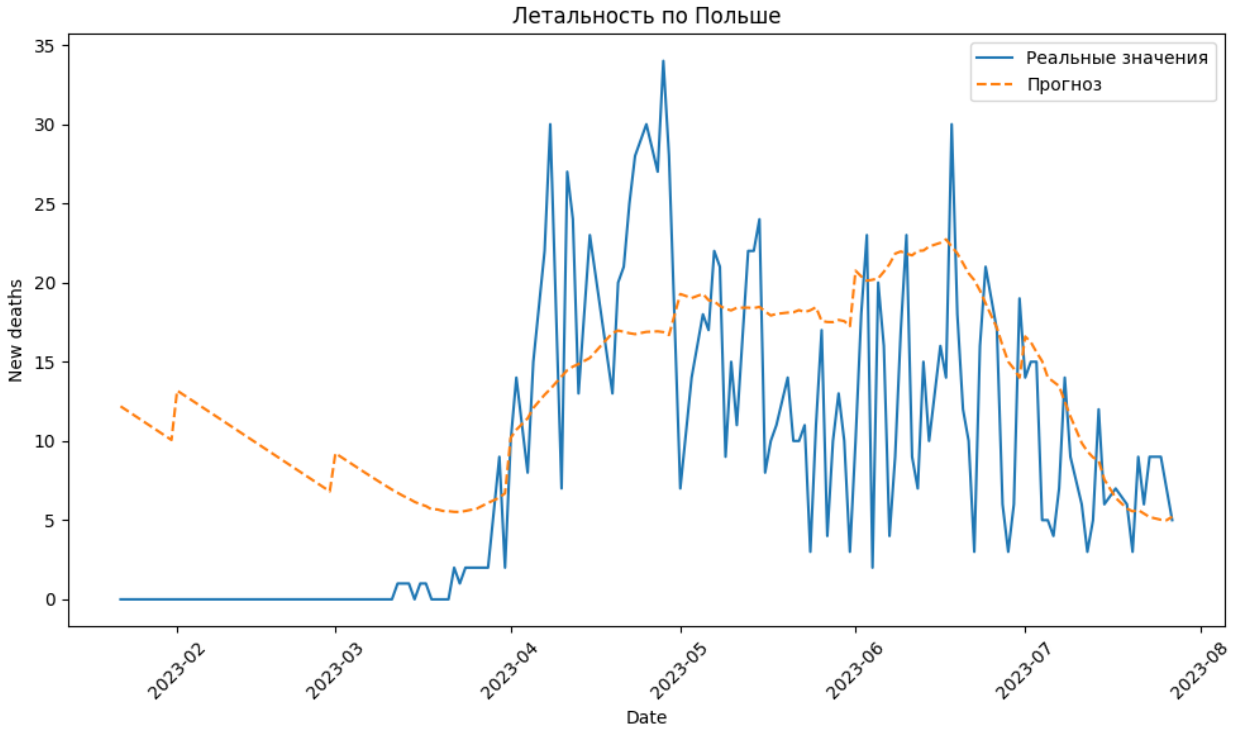

Рис 13. Данные прогноза по Польше

Развертывание и мониторинг

Для развертывания системы написанные функции были разделены по модулям и связаны при помощи Run.py

Data_html_Deaths.py	11.01.2024 2:10	Python File	5 КБ
Data_html_vactination.py	11.01.2024 2:16	Python File	4 КБ
fold_data.py	11.01.2024 2:36	Python File	3 КБ
Loader.py	11.01.2024 1:55	Python File	3 КБ
RUN.py	11.01.2024 2:44	Python File	1 КБ
Train_LSTM.py	11.01.2024 2:36	Python File	3 КБ
Train_Regress.py	11.01.2024 2:39	Python File	3 КБ
Train_SARIMA.py	11.01.2024 2:36	Python File	2 КБ

Рис 14. Структура модуля

После запуска подгружаются новые данные в CSV файл, который дополняется при каждом запуске скрипта. Каждая запись с wordmeter дополняется датой.

	A	B	C	D	E	F	G	H	I
1	Country	Other	TotalCase	NewCase:	TotalDeat	NewDeaths	TotalRecovered	ActiveCases	Date
2	World	701,437,7(	9,49	6,966,096	46	672,377,327	17,158	22,094,279	11.01.2024
3	USA	110,366,4:	5,409	1,191,189	11	108,073,499	7,517	1,101,764	11.01.2024
4	India	45,019,214		533,402		N/A	N/A	N/A	11.01.2024
5	France	40,138,560		167,642		39,970,918		0	11.01.2024
6	Germany	38,784,494		180,657		38,240,600		363,237	11.01.2024
7	Brazil	38,210,864		708,638		36,249,161		1,253,065	11.01.2024
8	S. Korea	34,571,873		35,934		34,535,939		0	11.01.2024
9	Japan	33,803,572		74,694		N/A	N/A	N/A	11.01.2024
10	Italy	26,671,165		195,139		26,257,742		218,284	11.01.2024
11	UK	24,812,582		232,112		24,580,470		0	11.01.2024
12	Russia	23,798,457		401,543		23,198,668		198,246	11.01.2024
13	Turkey	17,232,066		102,174		N/A	N/A	N/A	11.01.2024
14	Spain	13,914,811		121,76		13,762,417		30,634	11.01.2024
15	Australia	11,762,467	1,984	23,881	10	N/A	N/A	N/A	11.01.2024
16	Vietnam	11,624,114		43,206		10,640,971		939,937	11.01.2024
17	Taiwan	10,241,523		19,005		10,222,518		0	11.01.2024
18	Argentina	10,080,046		130,685		9,949,361		0	11.01.2024

Рис 16. Структура файла CSV

В создаваемой папке Result хранятся html файлы, а в models веса моделей (LSTM и SARIMAX)

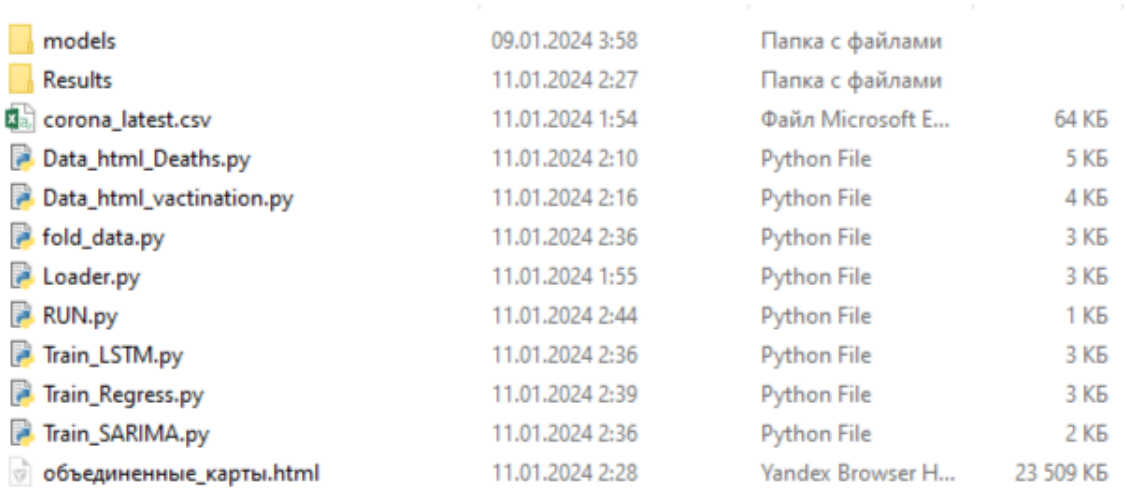

Рис 17. Структура модуля

Для работы используется Docker file

```
# Установка необходимых зависимостей
FROM ubuntu:latest
RUN apt-get update && apt-get install -y xvfb
# Установка Python и других необходимых зависимостей
RUN apt-get install -y python3 python3-pip
RUN pip3 install requests pandas plotly numpy plotly BeautifulSoup scikit-learn
# Создание рабочей директории в контейнере
WORKDIR /app
# Копирование файлов в рабочую директорию
COPY . /app
# Установка Xvfb и настройка переменной окружения DISPLAY
ENV DISPLAY=:99
CMD Xvfb :99 -screen 0 1024x768x16
```

Датасеты скачиваются самостоятельно.