

Федеральное государственное автономное образовательное  
учреждение высшего  
образования  
«Национальный исследовательский университет  
«Высшая школа экономики»

Образовательная программа "Прикладная математика"  
бакалавр

ОТЧЕТ  
по проектной работе

Предсказание развития пандемии при помощи SEIR  
модели

Выполнили  
студент группы БПМ-201  
Прокопов Иван Алексеевич

студентка группы БПМ-204  
Пугач Виктория Павловна

Руководитель проекта

д.ф.-м.н., доцент,  
департамент Прикладной математики  
Щур Владимир Львович

---

(оценка)

---

(подпись)

---

(дата)

Москва - 2021

# Введение

Начавшаяся осенью 2019 года и продолжающаяся по сей день пандемия новой коронавирусной инфекции является одним из основных объектов исследования ученых во всем мире. Изучение взаимодействия восприимчивых индивидуумов позволяет предсказывать динамику развития заболевания. В настоящее время важным способом изучения распространения и развития инфекций является математическое моделирование. Математические модели позволяют прогнозировать качественные изменения популяции в период пандемии при заданных эпидемиологических параметрах, таких как: заразность вируса, длительность инкубационного периода, принятие карантинных мер, смертность от заболевания и т.д. Но главным преимуществом моделирования является наглядность результатов. Модели позволяют предсказывать приблизительное общее число инфицированных, продолжительность пандемии и периоды самых сильных скачков заболевания. Достаточно сложные модели позволяют прогнозировать даже количество пациентов, которым будет необходима госпитализация в больничное учреждение, и влияние вакцинации на исход пандемии. Понимание последствий распределения ресурсов здравоохранения играет важную роль в предотвращении новых вспышек заболевания. Результатом прогнозирования развития пандемии в России, например, стал один из самых низких показателей процента смертности от вируса среди стран с тем же количеством зараженных граждан.

Таким образом, основная задача исследования формулируется как "составление модели, достаточно точно описывающей динамику развития пандемии". Решающими критериями при выборе предмета исследования стали актуальность темы и интерес к составлению наиболее точной математической модели.

С задачей анализа динамических систем в условиях большого количества неизвестных параметров позволяет справиться компартментальная модель.

Компартментальная модель (англ. *compartment* — отсек) подразумевает разделение всей популяции восприимчивых к инфекции индивидуумов на определенные классы, такие как: "**S**usceptible" - восприимчивые, "**I**nfectious" - инфицированные, "**R**ecovered" - устойчивые к заболеванию особи. Компартментальная модель отображает переход особей из одного класса в другой, а направление перехода между классами закладывается в название конкретной модели.

**SIR модель** - базовая модель, которая является основой для других, более точных и сложных компартментальных моделей.

## Подготовка к исследованию

Было решено, что для исследования развития пандемии будет взята пандемия новой коронавирусной инфекции, вызванной вирусом SARS-CoV-2. Подготовительным этапом работы стало изучение правила составления базовой SIR модели.

Далее, из широкого спектра компартментальных моделей, рассмотренных в [2] статье, было необходимо выбрать ту, что наиболее точно описывала бы характер распространения инфекции COVID-19. Так как для данной инфекции характерен некоторый инкубационный период и возникновение временного иммунитета у большинства переболевших, в качестве наиболее подходящей на данном этапе выбрана модель SEIR, которая отличается от базовой модели добавлением класса "Exposed" - класса облученных, то есть тех особей, которые уже были заражены, но еще не могут заражать других.

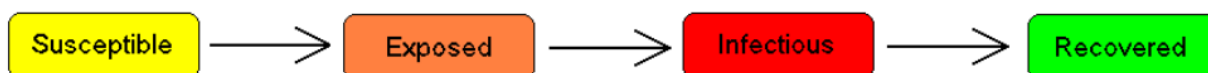


Рис. 1:

Чтобы учесть инкубационный период  $a$  болезни, вводится коэффициент  $p$ .

$$\alpha = \langle a^{-1} \rangle = p$$

Далее составляется система дифференциальных уравнений:

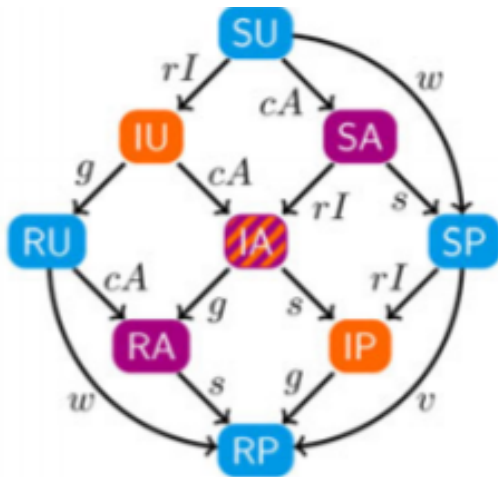
$$\begin{cases} S' = bN - bS - \xi \frac{IS}{N} - vS \\ E' = \xi \frac{IS}{N} - (b + p)E \\ I' = pE - (S + b)I \\ R' = SI - bR + vS \end{cases}$$

Однако классическая модель SEIR не предусматривает такой важный фактор развития пандемии как появление вакцины против коронавирусной инфекции, и вслед за этим появление такого течения как "анти-вакцинщики" - *Anti-vaccine*. В связи с этим возникла необходимость видоизменить модель с учетом влияния вакцинации на имеющуюся модель.

В изученной статье [1] для оценки воздействия вакцинации на протекание пандемии вся популяция была условно разделена на три класса: "Undecided"

- те, кто еще не принял свое решение относительно вакцинации, "**Anti-vaccine**" - резко негативно относящиеся к вакцинации и "**Pro-vaccine**" - положительно относящиеся к вакцинации индивидуумы.

Таким образом, графическое представление модели приобретает следующий вид:



Возникает необходимость описать появившиеся коэффициенты, которые задают переход особей из одного отсека в другой:

$rI$  - коэффициент интенсивности контактов

$v$  - уровень вакцинации

$s$  - переход из U в A

$\omega$  - переход из U в P

$g$  - скорость выздоровления

На данном этапе было решено не разделять популяцию на отдельные классы, отвечающие за отношение к вакцинации, а провести лишь условное разделение, введя необходимые коэффициенты.

## Существующие аналоги

### SIR модель

Были изучены многие аналоги модели, однако наибольший интерес вызвала [эта](#) SIR модель, реализованная на языке **Java Script** [4]. В данной модели реализован интерактивный ползунок, позволяющий изменять эпидемиологические параметры прямо в процессе наблюдения, без правок в самом коде.

Однако модель, рассмотренная в приведенном примере, не учитывает воздействие такого фактора, как возникновение вакцинации и отношение общества к этому.

Было принято решение реализовать данный прием для выбранной SEIR модели на языке Python, так как аналогов данной программы найти в открытом доступе не удалось.

# Описание хода работы

## Составление математической модели

Вторым этапом работы стало составление математической модели, позволяющей учитывать все интересующие нас в исследовании параметры.

Для реализации была составлена новая усовершенствованная система, которая учитывает влияние постановки вакцины.

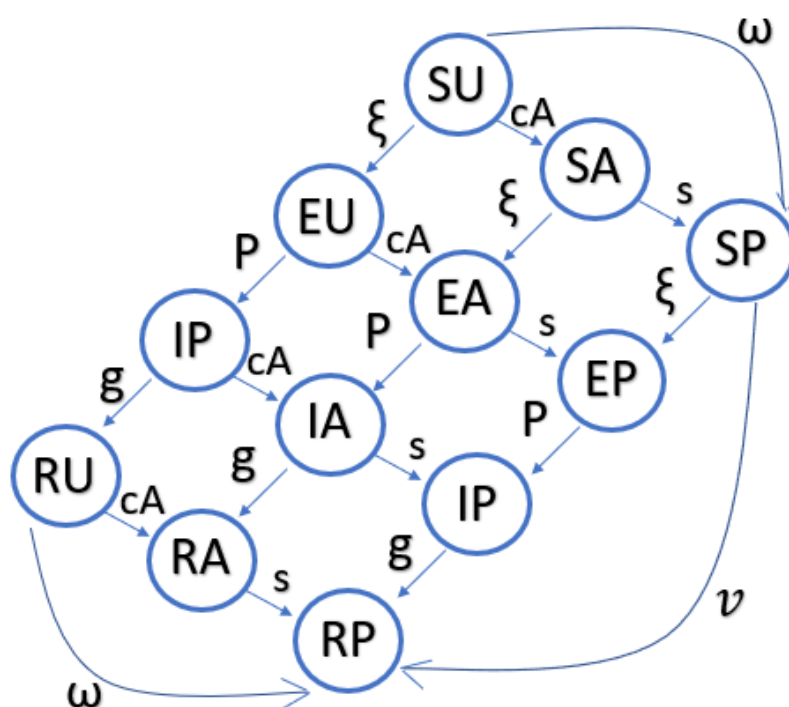


Рис. 2: Графическое представление итоговой модели

Новые введенные коэффициенты и некоторые константы, которые использовались в работе в дальнейшем:

$hard = (906 / 2487) / 10$  - Процент тяжелых случаев перенесения инфекции

$days = 10$  - Дни инкубационного периода

$maskeff = 0.3$  - Коэффициент ношения масок и их эффективности

$da = days^{-1}$  - Инкубационная константа

$b = 11.4 / 1000$  - Рождаемость

$g = 0.04$  - Вероятность получения иммунитета (переболеть с иммунитетом)

$v = 13000 / 12655050$  - Вакцинация (по данным Мэрии Москвы в день)

$e = 1 / (8*30)$  - Потеря иммунитета (по исследованиям иммунитет сохраняется приблизительно 5 месяцев)

$s$  - Переход к пропрививочникам от антипрививочников

$dw$  - Переход к провакцинщикам от неопределившихся

$d$  - Переход к неопределившимся от провакцинщиков

Таким образом, итоговая система уравнений может быть записана так:

$$\left\{ \begin{array}{l} SU = b + e(RU) + d(SP) - (rI + cA + w + b)(SU) \\ SA = cA(SU) - (s + rI + b)(SA) \\ SP = w(SU) + s(SA) + e(RP) - (rI + d + v + b)(SP) \\ \\ EU = rI(SU) - (a + b)(EU) \\ EA = rI(SA) - (a + b)(EA) \\ EP = rI(SP) - (a + b)(EP) \\ \\ IU = a(EU) - (g + b)(IU) \\ IA = a(EA) - (g + b)(IA) \\ IP = a(EP) - (g + b)(IP) \\ \\ RU = g(IU) + d(RP) - (e + cA + w + b)(RU) \\ RA = cA(RU) + g(IA) - (s + b)(RA) \\ RP = w(RU) + s(RA) + g(IP) + v(SP) - (b + d + e)(RP) \end{array} \right.$$

## Создание компьютерной модели

Третьим этапом работы над проектом стало создание программы, позволяющей строить модель, и добавление к полученным в ходе выполнения программы графикам интерактивной легенды.

Для написания программы был выбран язык программирования **Python 3.9**. Также были дополнительно изучены и использованы следующие библиотеки: **SciPy**, **NumPy** и **Plotly**.

## Код программы. [Ссылка на GitHub](#)

```
import matplotlib.pyplot as plt
from scipy.integrate import odeint
import numpy as np
import plotly
import plotly.graph_objs as go
import plotly.express as px
from plotly.subplots import make_subplots
import pandas as pd

def system(SIRUAP, t, a, b, r, v, g, c, s, w, d, e, S, E, I, R, U, A, P):
    SU, SA, SP, EU, EA, EP, IU, IA, IP, RU, RA, RP = SIRUAP
    I = IU + IA + IP
    return [
        #S
        b + e * (RU) + d * (SP) - (r * I + c * A + w + b) * (SU),
        c * A * (SU) - (s + r * I + b) * (SA),
        w * (SU) + s * (SA) + e * (RP) - (r * I + d + v + b) * (SP),
        #E
        r * I * (SU) - (a + b) * (EU),
        r * I * (SA) - (a + b) * (EA),
        r * I * (SP) - (a + b) * (EP),
        #I
        a * (EU) - (g + b) * (IU),
        a * (EA) - (g + b) * (IA),
        a * (EP) - (g + b) * (IP),
        #R
        g * (IU) + d * (RP) - (e + c * A + w + b) * (RU),
        c * A * (RU) + g * (IA) - (s + b) * (RA),
        w * (RU) + s * (RA) + g * (IP) + v * (SP) - (b + d + e) * (RP)]

t = np.linspace(0, 200, 200) # vector of time
S = 997 / 1000
E = 0
I = 3 / 1000
R = 0
A = 0
P = 0
U = 1

hard = (906 / 2487) / 10 #https://mosgorzdrav.ru/ru-RU/news/default/card/5642.html
days = 10 #Дни инкубационного периода
maskeff = 0.3 #Коэффициент ношения масок и их эффективности
a = days**(-1) #Инкубационная константа
b = 11.4 / 1000 #Рождаемость
r = 0.4 * (1 - maskeff) # Заразность
c = 0 #Переход к антивакцинистам из неопределившихся
g = 0.04 #Вероятность получения иммунитета (переболеть с иммунитетом)
v = 13000 / 12655050 #Вакцинация (по данным Мэрии Москвы в день / официальное население)
e = 1 / (8*30) #Потеря иммунитета (по исследованиям иммунитет сохраняется приблизительно 5 месяцев)
s = 0 #Переход к проприивочникам от антипрививочников
w = 0 #Переход к провакцинистам от неопределившихся
d = 0 #Переход к неопределившимся от провакцинистов
SIRUAPO = [S * U, S * A, S * P, E * U, E * A, E * P, I * U, I * A, I * P, R * U, R * A, R * P]
w = odeint(system, SIRUAPO, t, args=(a, b, r, v, g, c, w, s, d, e, S, E, I, R, U, A, P))
```

```

SU = w[:, 0]
SA = w[:, 1]
SP = w[:, 2]
EU = w[:, 3]
EA = w[:, 4]
EP = w[:, 5]
IU = w[:, 6]
IA = w[:, 7]
IP = w[:, 8]
RU = w[:, 9]
RA = w[:, 10]
RP = w[:, 11]
S = SU + SA + SP
E = EU + EA + EP
I = IU + IA + IP
R = RU + RA + RP
U = SU + EU + IU + RU
A = SA + EA + IA + RA
P = SP + EP + IP + RP
Med = []*len(t)
for i in range(len(t)):
    Med.append(22000/12655050) #Доля коек для больных Covid19
fig = go.Figure()
fig.add_trace(go.Scatter(x=t, y=S, name='S'))
fig.add_trace(go.Scatter(x=t, y=E, name='E'))
fig.add_trace(go.Scatter(x=t, y=I, name='I'))
fig.add_trace(go.Scatter(x=t, y=R, name='R'))
fig.add_trace(go.Scatter(x=t, y=hard*I, name='Тяжелые случаи(госпитализация)'))
fig.add_trace(go.Scatter(x=t, y=Med, name='Количество коек'))
fig.update_layout(legend_orientation="h",
                    legend=dict(
                        x=.5,
                        xanchor="center"),
                    xaxis_title="Время в днях",
                    yaxis_title="Доля от популяции",
                    margin=dict(l=0, r=0, t=30, b=0))
fig.update_traces(hoverinfo="all", hovertemplate="Время: %{x}<br>Значение: %{y}")
fig.write_html('first_figure.html', auto_open=True)

```

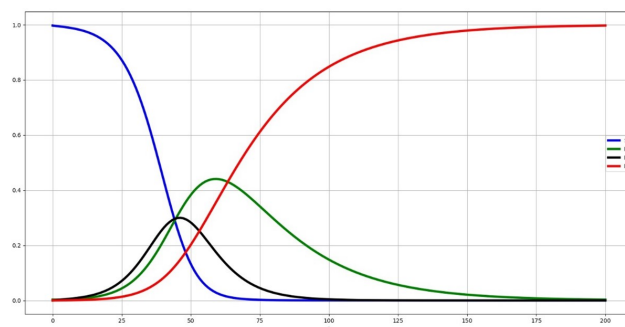


Рис. 3: Первая версия графиков до создания пользовательского интерфейса.



## Анализ полученных графиков

Данная программа позволяет предсказывать динамику распространения коронавирусной инфекции в Москве.

Как уже упоминалось, важно иметь возможность прогнозировать количество пациентов, которым будет необходима госпитализация, то есть количество пациентов с тяжелой формой заболевания.

Написанная программа позволяет наглядно продемонстрировать воздействие соблюдения санитарно-эпидемиологических требований на динамику пандемии.

Для этого, построим два графика, построенные при разных значениях коэффициента ношения масок: **0,3** и **0,7**.

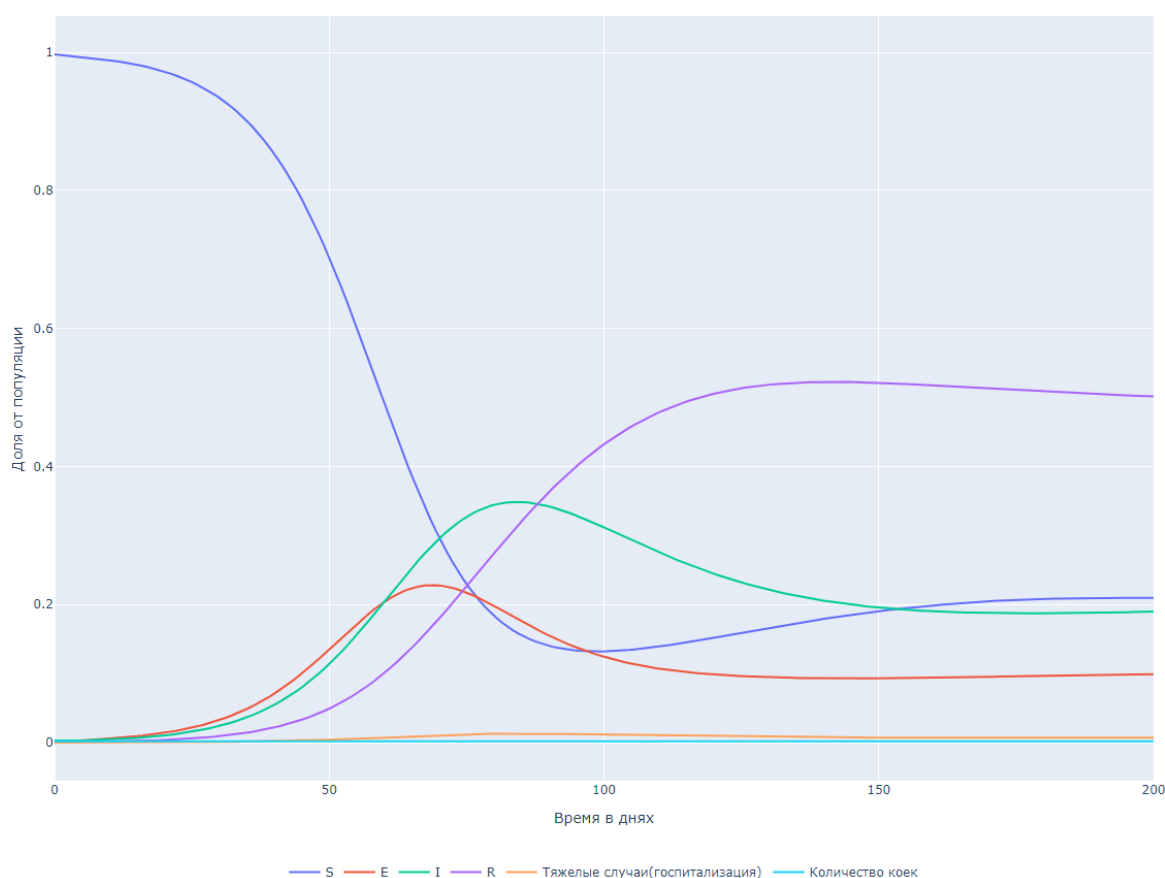


Рис. 4: График при значении коэффициента ношения масок 0.3

Как мы видим, оранжевая линия, отвечающая за количество тяжелых случаев заболевания более чем в два раза выше, чем максимально доступное количество коек в больнице. Это говорит о том, что если коэффициент ношения медицинских масок будет так низок, то система здравоохранения не сможет справиться с наплывом пациентов, что наиболее вероятно приведет к росту летальных исходов.

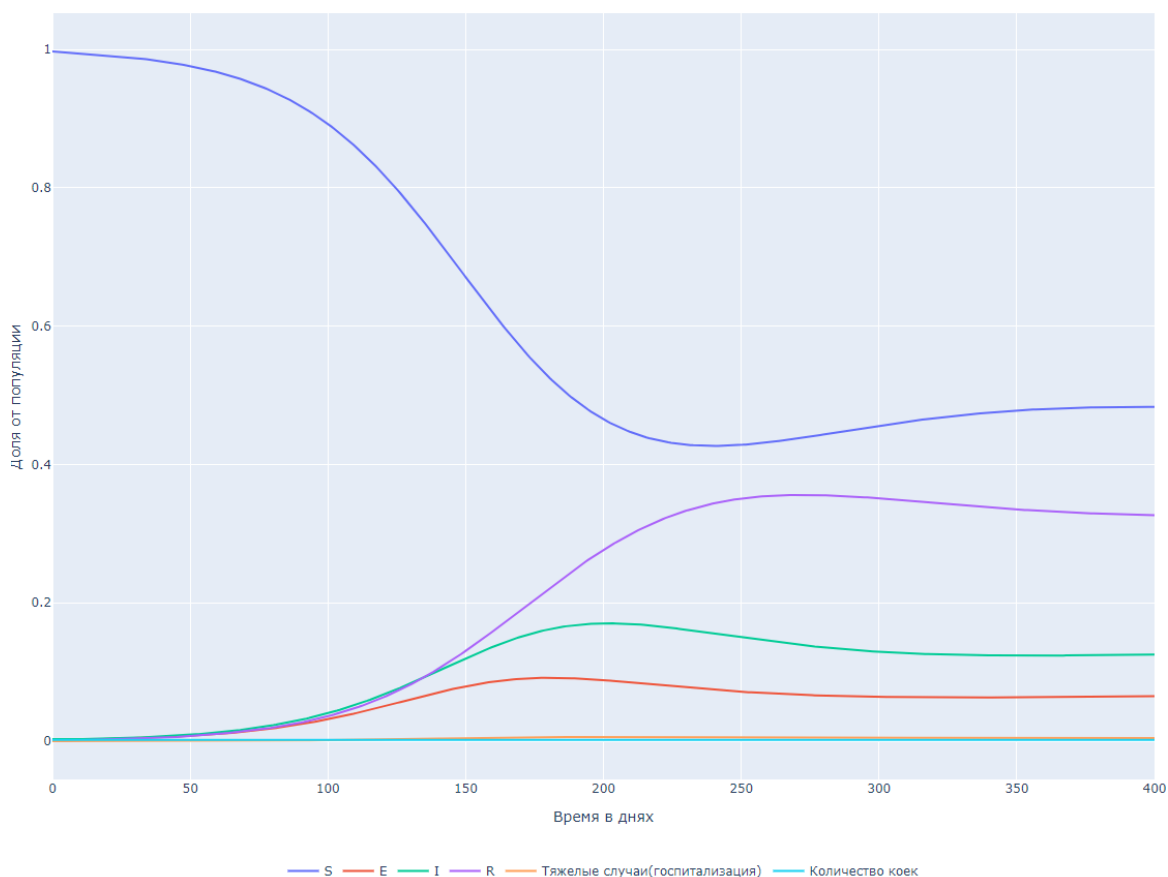


Рис. 5: График при значении коэффициента ношения масок 0.7

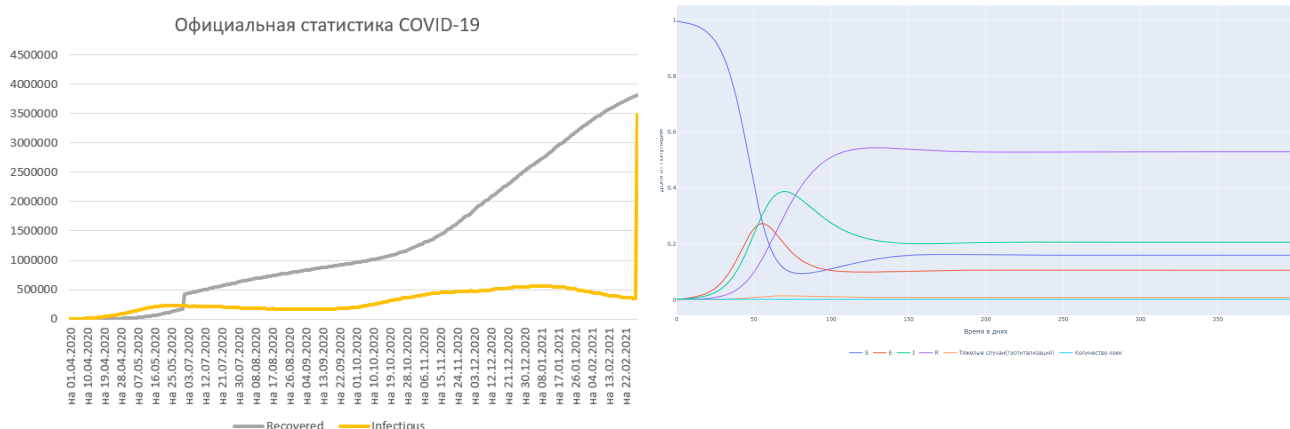
На втором графике уже видно, что прогнозируемое количество больных с тяжелым течением болезни лишь немного больше количества коек. Так же важно обратить внимание, что время наблюдения, указанное нами, увеличено в 2 раза, чтобы в рассматриваемую область попал пик заражаемости. Это очередное подтверждение того факта, что увеличение коэффициента ношения масок замедляет распространение инфекции.

Видим, что визуализация модели позволяет интерпретировать важные эпидемиологические параметры доступным каждому образом.

Именно поэтому было принято решение усовершенствовать полученную модель, добавив пользовательский интерфейс, позволяющий менять коэффициенты в модели, не меняя при этом код.

## Сравнение с реальными данными

Так как мы получили некоторые результаты, есть возможность сравнить их с официальными данными с [сайта Министерства финансов](#). Нетрудными манипуляциями можно построить аналогичный график по найденным данным и наложить на график, полученный с помощью созданной модели.



И даже если не брать во внимание то, что наша модель показывает не количество, а долю особей от популяции, которые принадлежат определенному классу, можно заметить одно из наиболее важных отличий: реализованная модель не позволяет предсказать так называемую **вторую волну**, что является существенным минусом.

## Анализ результатов

Модель с определенной точностью позволяет предсказывать развитие пандемии в зависимости от вводимых эпидемиологических коэффициентов. Однако она не предусматривает рассмотрение каких-либо непредвиденных скачков в динамике протекания пандемии.

Минусы полученной SEIR модели:

- Низкая точность прогнозирования, вызванная большим количеством коэффициентов, численно задать которые не представляется возможным.
- Невозможность прогнозирования воздействия социальных факторов на протекание пандемии. Например нельзя учесть внезапное перекрытие или открытие границ, а также проведение масштабных мероприятий.

- Отсутствие на данный момент достаточных исследований по точному вычислению некоторых параметров распространения и течения болезни:
  1. Заразность (для разных штаммов короновирусной инфекции может быть различной, достаточные комплексные исследования еще не были проведены)
  2. Время сохранения иммунитета после перенесения короновирусной инфекции (точно не выявлено и может сильно варьироваться, в зависимости от тяжести протекания заболевания)
  3. Отсутствие разделения иммунитета на иммунитет, приобретенный после вакцинации и иммунитет, приобретаемый после перенесения заболевания
  4. Отсутствие достаточных исследований отношения общества к вакцинации и перехода между классами в модели в модели UAP(SEIRUAP)
- С технической стороны были встречены также такие трудности:
  1. Модель не предусматривает рассмотрение смертности от вируса. Для исследования этого параметра необходимо составление новой модели
  2. Сложности в изучении библиотеки SciPy, а именно в изучении метода odeint – невозможность использовать его в цикле, т.к. определенные параметры приводят к ошибке при повторном запуске модуля

На основании результатов, полученных в ходе исследования, было решено, какие необходимо ввести дополнительные коэффициенты в модель, чтобы добиться наибольшей точности прогнозирования развития пандемии.

$e$  - коэффициент вероятности заражения

$p$  - коэффициент, отвечающий за время проявления симптомов

$c$  - вероятность стать противником вакцинации

$s$  - вероятность перехода из  $A$  в  $P$

$\omega$  - вероятность перейти в  $P$

$v$  - скорость вакцинации

$g$  - коэффициент выздоровления

$r$  - коэффициент смертности

$f$  - коэффициент потери иммунитета для переболевших

$d$  - коэффициент потери иммунитета для вакцинировавшихся

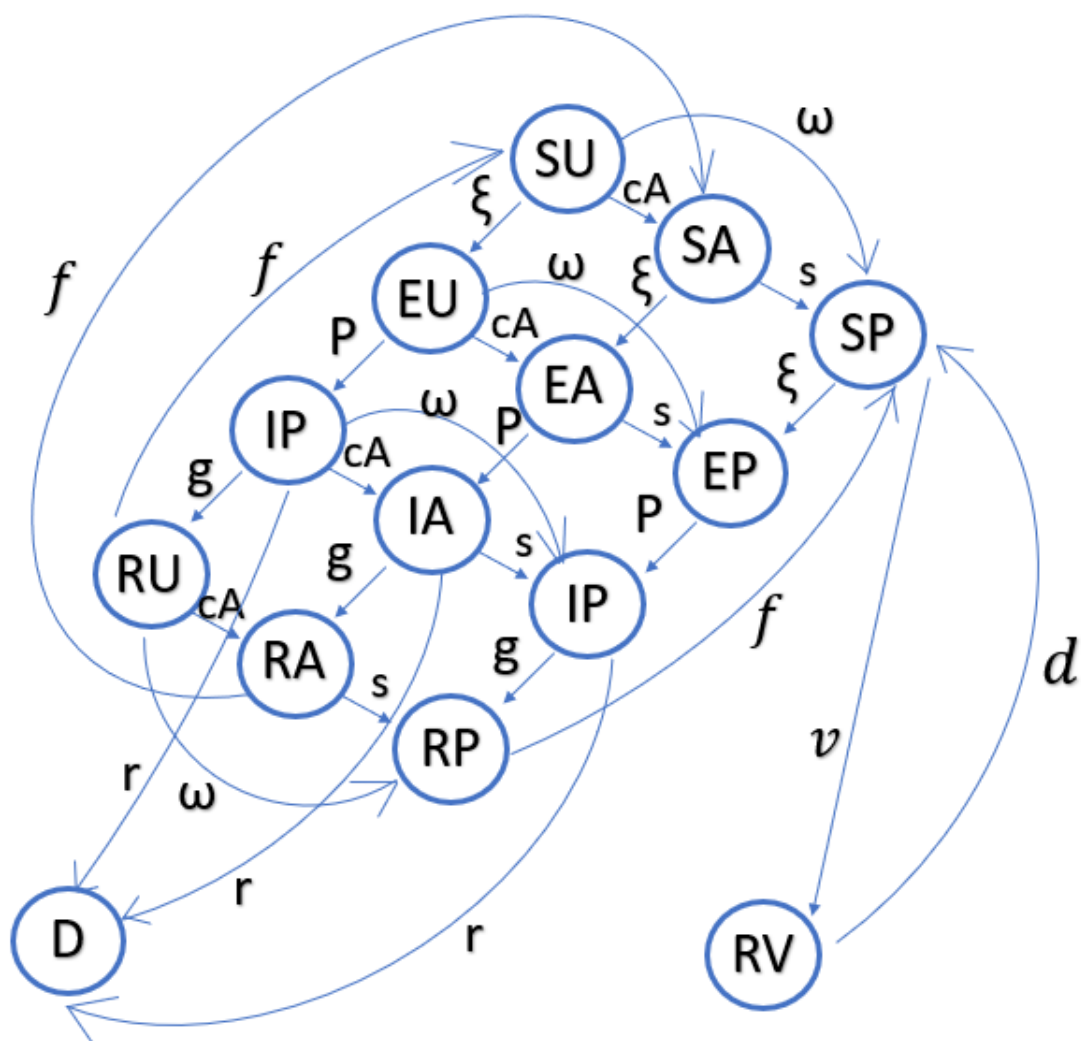


Рис. 6: Модель, которую планируется реализовать в дальнейшем

## Вклад в проект Прокопова Ивана

1. Изучение библиотек SciPy, NumPy, Plotly.
2. Составление системы уравнений, описывающей изменение 12ти классов, на которые была разделена популяция при создании модели, учитывающей отношение к вакцинации.
3. Реализация алгоритма для прогнозирования распространения COVID-19 в IDE PyCharm.
4. Изучение источников в поисках коэффициентов, необходимых для работы алгоритма.

## Вклад в проект Пугач Виктории

1. Изучение библиотек NumPy, Matplotlib.
2. Создание модифицированной SEIR модели. Модификация заключается в добавлении модели UAP – модели отношения к прививкам (U – не решившие будут ли делать прививку, A – антипрививочники, P – решившие сделать прививку).
3. Создание пользовательского интерфейса программы.
4. Проведение сравнительного анализа полученных данных с реальной статистикой распространения коронавирусной инфекции и написание отчета по проекту.

## Перспективы развития проекта

- Расширение количества групп – дифференциация групп по возрасту. Данное разделение позволит более точно рассчитать итоговый результат благодаря различным коэффициентам передачи (более возрастные группы имеют меньшее количество социальных контактов), смертность у более возрастных групп также выше.
- Решение проблемы с повторным вызовом метода odeint и создание графика с наличием ползунка (слайдера) с возможностью изменения графика в реальном времени посредством изменения таких коэффициентов как ношение масок, ограничение социальных контактов со стороны государства, введение жесткого “Локдауна” или пропускной системы.
- Создание новой модели, позволяющей прогнозировать наступление второй волны
- Более глубокий анализ источников для нахождения наиболее точных коэффициентов, используемых в модели.
- Потенциальное использование мощностей вычислительного кластера МИ-ЭМа для ускорения и упрощения работы с тяжелой многофакторной моделью.

## Заключение

В ходе работы над проектом участниками был изучен принцип составления компартментальных моделей и основы решения систем дифференциальных уравнений. Была составлена математическая модель, позволяющая прогнозировать развитие пандемии при определенных эпидемиологических параметрах. Была реализована компьютерная программа, позволяющая прогнозировать динамику пандемии COVID-19, и пользовательский интерфейс для этой программы. А также был проведен сравнительный анализ полученных данных с данными из официальных источников.

Так как модель достаточно гибкая, есть возможность адаптировать ее для изучения динамики и других эпидемий более ранних годов, для которых уже известны многие параметры.

Получено, что построенная модель действительно с определенной точностью может верно предсказывать развитие пандемии. При дальнейшем усовершенствовании модели планируется добавление новых классов индивидуумов и учет других коэффициентов, не вошедших в созданную систему.

## Список литературы

- [1] [Rohan S. Mehta and Noah A. Rosenberg] Modelling anti-vaccine sentiment as a cultural pathogen
- [2] Статья о разновидностях компартментальных моделей  
[//http://surl.li/uwdr](http://surl.li/uwdr)

## Список источников

- [1] Коронавирус: статистика [//https://index.minfin.com.ua/reference/coronavirus](https://index.minfin.com.ua/reference/coronavirus)
- [2] Сайт, откуда были взяты данные о случаях тяжелого протекания заболевания  
<https://mosgorzdrav.ru/ru-RU/news/default/card/5642.html>
- [3] Сайт с данными о рождаемости/ <http://surl.li/uwed>
- [4] Программа, в которой реализованы слайдеры  
<https://jsfiddle.net/pbuzin/tqcx5L8b/19/>
- [5] Ссылка на **GitHub** <https://vk.com/away.php?to=https>