Лабораторная работа №7. Модель эффективности рекламы

c/б 1032186063 | НФИбд-01-18

Доборщук Владимир Владимирович

27 марта 2021

Содержание

# Цели и задачи

**Цель:** изучить модель эффективности рекламы, а также реализовать её программно.

**Задачи:**

* изучить теорию о модели эффективности рекламы
* построить модель для 3 различных случаев и

# Теоретическая справка

Модель рекламной кампании описывается следующими величинами:

Считаем, что - скорость изменения со временем числа потребителей, узнавших о товаре и готовых его купить, - время, прошедшее с начала рекламной кампании, - число уже информированных клиентов. Эта величина пропорциональна числу покупателей, еще не знающих о нем, это описывается следующим образом: , где - общее число потенциальных платежеспособных покупателей, - характеризует интенсивность рекламной кампании (зависит от затрат на рекламу в данный момент времени). Помимо этого, узнавшие о товаре потребители также распространяют полученную информацию среди потенциальных покупателей, не знающих о нем (в этом случае работает т.н. сарафанное радио). Этот вклад в рекламу описывается величиной , эта величина увеличивается с увеличением потребителей узнавших о товаре. Математическая модель распространения рекламы описывается уравнением:

При получается модель типа модели Мальтуса, В обратном случае, при получаем уравнение логистической кривой.

# Программная реализация

## Подготовка к моделированию

Все данные соответствуют варианту 14 = .

**Инициализация библиотек**

import numpy as np  
import matplotlib.pyplot as plt  
from scipy.integrate import odeint  
from math import sin  
from scipy.misc import derivative  
  
from jupyterthemes import jtplot  
jtplot.style(context='notebook', fscale=1.2, gridlines='--')

**Начальные данные и необходимые функции**

N = 648  
t0 = 0  
N0 = 12  
  
def k(t):  
 return 0.125  
  
def p(t):  
 return 0.00002  
  
t = np.arange(0, 30, 0.01)

Объявим необходимые функции, исходя из данной нам информации в теоретической справке.

def XD(x,t):  
 xd = (k(t) + p(t)\*x)\*(N-x)  
 return xd

Заложим в переменную решения для наших СДУ с помощью функции odeint модуля scipy.integrate.

x = odeint(XD, N0, t)

## Построение графиков для модели

### Модель 1

plt.plot(t, x)  
plt.ylabel('N')  
plt.xlabel('t')  
plt.show()

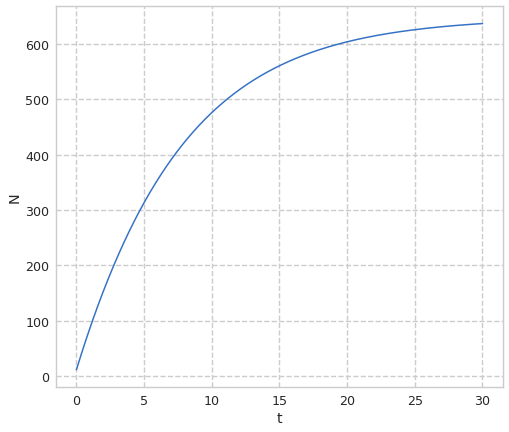


График для модели 1

### Модель 2

def k(t):  
 return 0.000095  
  
def p(t):  
 return 0.92  
  
x = odeint(XD, N0, t)  
  
plt.plot(t, x)  
plt.ylabel('N')  
plt.xlabel('t')  
plt.show()

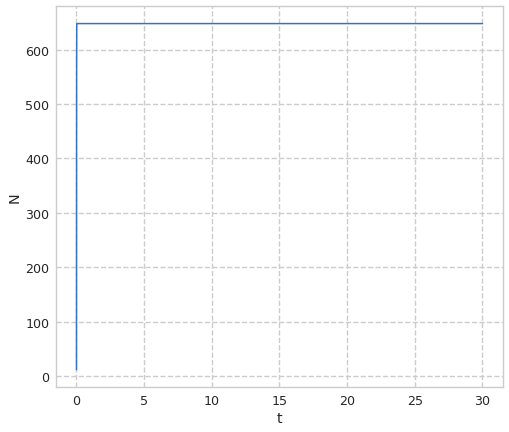


График для модели 2

Найдем момент времени, при котором скорость изменения числа потребителей.

diff = 0  
ind = -1  
for i in range(1, len(x[:,0])):  
 if (x[i][0] - x[i-1][0]) > diff:  
 diff = x[i][0] - x[i-1][0]  
 ind = i

Максимальная скорость изменения числа потребителй будет при t=0.01

## Модель 3

def k(t):  
 return sin(10\*t)  
  
def p(t):  
 return 0.9\*t  
   
x = odeint(XD, N0, t)  
  
plt.plot(t, x)  
plt.ylabel('N')  
plt.xlabel('t')  
plt.show()

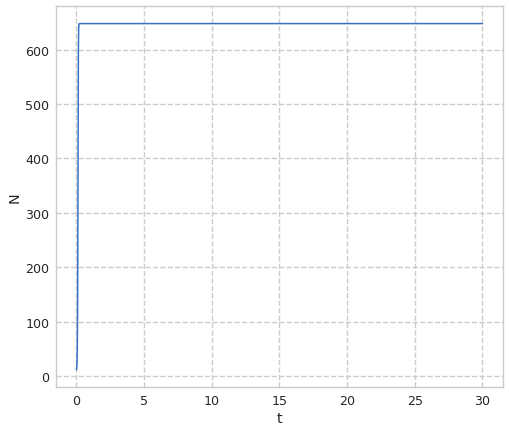


График для модели 3

# Выводы

Мы изучили простейшую модель эффективности рекламы, после чего успешно реализовали её с помощью языка Python и дополняющих его модулей.