# Обобщенная модель регрессии Пуассона

The Generalized Poisson Regression Model

https://timeseriesreasoning.com/contents/generalized-poisson-regression-model/

[Sachin Date](https://sachin-date.medium.com/?source=post_page-----50cccba15958--------------------------------)

Оглавление

[Тематика статьи 2](#_Toc78236460)

[Что такое счетные данные? 2](#_Toc78236461)

[Равная дисперсия, недостаточная дисперсия, избыточная дисперсия и ограничения модели Пуассона 3](#_Toc78236462)

[Обобщенная модель регрессии Пуассона по Консулу 5](#_Toc78236463)

[Ограниченное обобщенное распределение Пуассона по Фамойе 6](#_Toc78236464)

[Реализация GP-1 и GP-2 в Python и Statsmodels 7](#_Toc78236465)

[Набор данных из реального мира 7](#_Toc78236466)

[Цель регрессии 9](#_Toc78236467)

[Стратегия регрессии 9](#_Toc78236468)

[Стратегия регрессии – шаг за шагом 10](#_Toc78236469)

[Степень соответствия модели GP-1 13](#_Toc78236470)

[Прогноз 13](#_Toc78236471)

[Исходный код 15](#_Toc78236472)

[Резюме 17](#_Toc78236473)

[Ссылки, цитаты и авторские права 17](#_Toc78236474)

[Набор данных 17](#_Toc78236475)

[Статьи 17](#_Toc78236476)

[Книги 17](#_Toc78236477)

[Изображения 18](#_Toc78236478)

Обычная [модель регрессии Пуассона](https://timeseriesreasoning.com/contents/poisson-regression-model/) часто является первоначально выбираемой моделью для счетных наборов данных. Базовое предположение (ограничение) модели регрессии Пуассона состоит в том, что дисперсия данных равна их среднему значению, назовем подобные данные эквидисперсными (equi-dispersed). К сожалению, на практике данные редко бывают таковыми, что побуждает статистиков использовать иные модели, как, например:

* [Модель отрицательной биномиальной регрессии (NB)](https://timeseriesreasoning.com/contents/negative-binomial-regression-model/)
* Модель обобщенной регрессии Пуассона; фактически, это семейство моделей, которые и будут рассмотрены в статье.

Обе модели не требуют предположения *дисперсия = среднее*. Это делает их лучшим практическим выбором для моделирования. В [предыдущей статье](https://timeseriesreasoning.com/contents/negative-binomial-regression-model/) я подробно рассмотрел модель NB. Сейчас рассмотрим модели обобщенной регрессии Пуассона.

# Тематика статьи

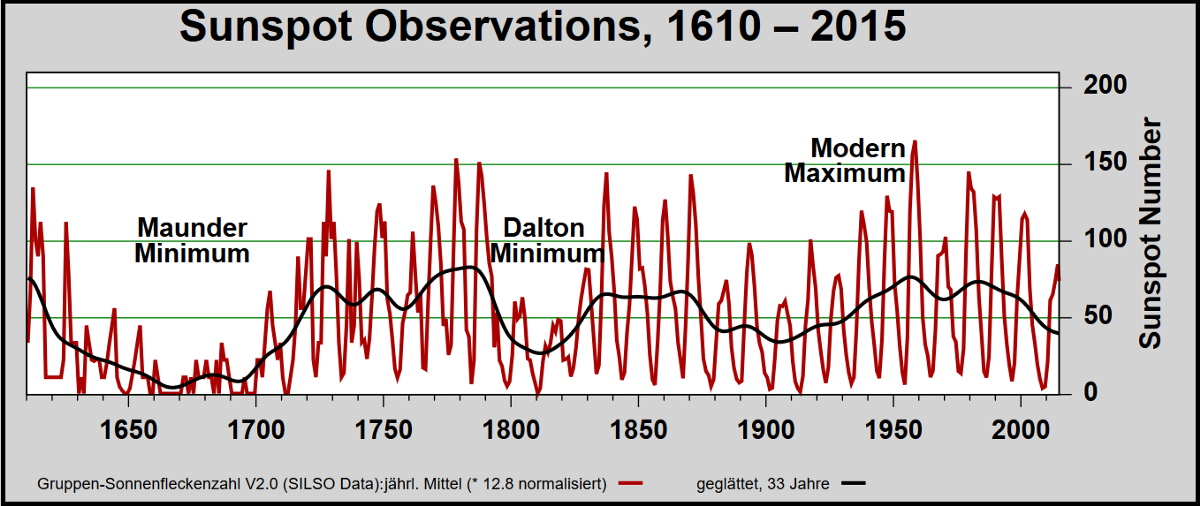
Рассмотрим следующие вопросы:

1. Что такое счетный набор данных (count based dataset)?
2. Что такое равная, недостаточная, избыточная дисперсии. Рассмотрим ограничение модели Пуассона для наборов данных с недостаточной или избыточной дисперсией.
3. Введение в обобщенную модель регрессии Пуассона (GP-1) по Консулу (Consul) и в модель ограниченной обобщенной регрессия Пуассона (GP-2) по Фамойе (Famoye).
4. Python-код для построения и обучения моделей GP-1 и GP-2 и сравнения их эффективности с обычной моделью регрессии Пуассона.

# Что такое счетные данные?

Счетные наборы данных – это наборы, в которых зависимая переменная **y** представляет собой количество появлений некоторого события за период. Вот некоторые примеры:

1. Количество обнаруженных экзопланет за месяц.
2. Количество солнечных пятен по годам.
3. Количество людей, приходящих в отделение неотложной помощи в час.

Солнечные пятна (Источник: [Wikimedia Commons,](https://commons.wikimedia.org/wiki/File:Sunspots-gn-yr-total-smoothed-en.svg) CC [BY-SA 4.0](https://creativecommons.org/licenses/by-sa/4.0/) )

В каждом из перечисленных наборов зависимая переменная **y** представляет количество (подсчеты) наблюдаемых событий, в то время как выбор **X**, матрицы независимых переменных, то есть набора переменных, которые, как считается, объясняют **y**, в основном, к сожалению, оставлен на разработчика статистических моделей.

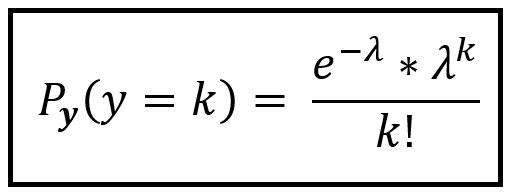
# Равная дисперсия, недостаточная дисперсия, избыточная дисперсия и ограничения модели Пуассона

Можно начать с попытки моделирования поведения зависимой переменной как пуассоновского процесса. К сожалению, на практике данный процесс неудовлетворительно объясняет изменчивость количества наблюдаемых событий. Это в значительной степени связано с ограничением модели регрессии Пуассона *дисперсия = среднее*. Другими словами, модель Пуассона ошибочно предполагает, что данные эквидисперсны.

[Модель Пуассона](https://timeseriesreasoning.com/contents/poisson-regression-model/) не может уйти от этого ограничения поскольку основана на предположении, что зависимая переменная **y** является случайной величиной, подчиняющейся распределению Пуассона, для которого можно показать, что дисперсия случайной величины равна ее среднему значению.

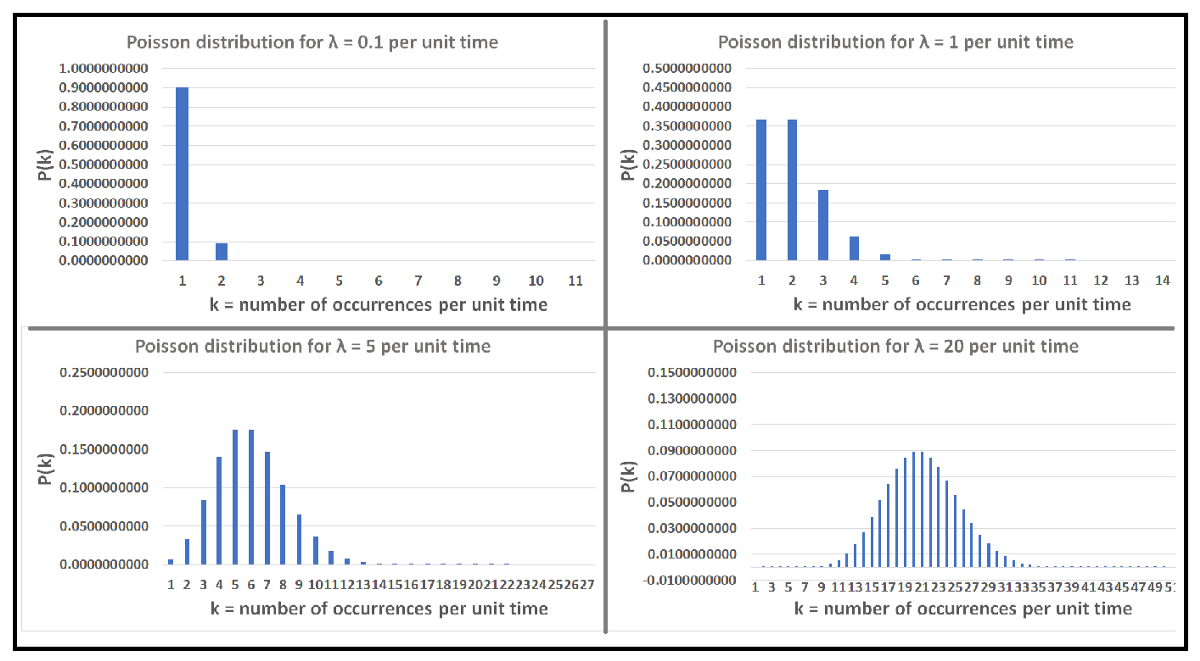
Формула на рисунке представляет пуассоновскую функцию распределения вероятностей (также известную как Probability Mass Function, PMF) случайной величины. Можно показать, что:

*Дисперсия (****X****) = Среднее (****X****) = λ*, где λ– количество событий, происходящих в единицу времени.



Вероятность появления k событий в предположении, что в единицу времени происходит λ событий (Изображение [автора](https://www.linkedin.com/in/sachindate/))

Несколько графиков распределения Пуассона для различных значений λ:

Вероятность появления k событий для различных λ (Изображение [автора](https://www.linkedin.com/in/sachindate/))

К слову сказать, в регрессионном моделировании не принято говорить о безусловной дисперсии и безусловном среднем. Предпочитают говорить о дисперсии как [об условной при независимых переменных **X**,](https://timeseriesreasoning.com/contents/conditionals/) принимающих некоторое значение **x(i)** для i-го наблюдения.

Условную дисперсию можно обозначить как *Дисперсия (****y*** *|* ***X*** *=* ***x(i)****)*. То же самое для условного среднего.

Как упоминалось ранее, на практике многие наборы данных не являются эквидисперсными, т.е. условная дисперсия не равна условному среднему значению. У некоторых наборов дисперсия больше, чем среднее значение, что называют избыточной дисперсией, в то время как у других дисперсия меньше среднего, т.е. имеет место недостаточная дисперсия.

В целом:

когда дисперсия зависимой переменной **y** больше, чем предполагается в обычной модели, набор данных чрезмерно рассредоточен;

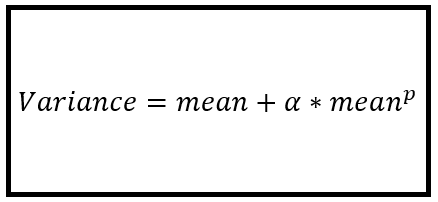
когда дисперсия зависимой переменной **y** меньше того, что предполагается в обычной модели, набор данных недостаточно рассредоточен.

Эффект чрезмерной (или недостаточной) дисперсии заключается в том, что обычная модель не способна адекватно объяснить изменчивость количества наблюдаемых событий. Соответственно, и прогнозы оказываются некачественными.

Распространенное решение этой проблемы – предположить, что дисперсия является некоторой функцией среднего, то есть:

*Дисперсия = f (Среднее)*

Одна из наиболее часто используемых формул для f:



Часто используемая функция дисперсии (Изображение [автора](https://www.linkedin.com/in/sachindate/))

Где p = 0,1,2…

α известен как параметр дисперсии, который представляет дополнительную изменчивость **y**, вызванную некоторым неизвестным набором переменных, из-за которых дисперсия и отличается от ожидаемой для обычной регрессионной модели.

Обратим внимание, что при α = 0 *Дисперсия = Среднее*, и получаем стандартную модель регрессии Пуассона.

При α> 0 выделим два случая для p = 1 и p = 2. В обоих случаях получаем так называемую модель отрицательной биномиальной регрессии (NB).

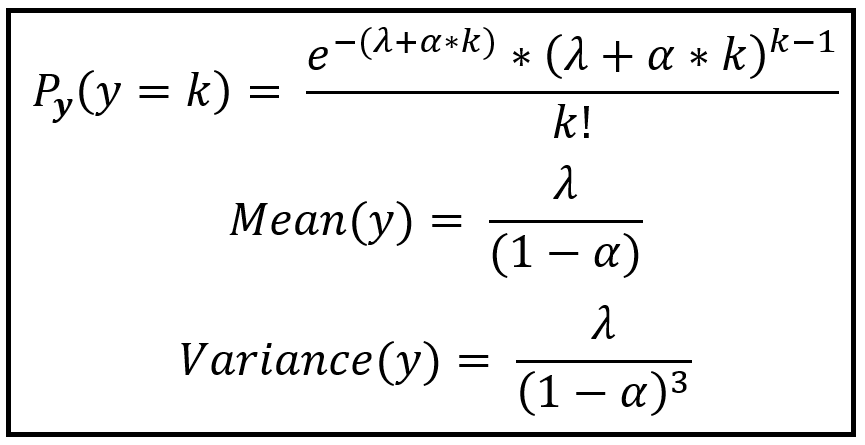
В модели регрессии NB предполагается, что наблюдаемые значения **y** являются случайной величиной с распределением Пуассона с частотой событий λ, а сама λ является случайной величиной с гамма-распределением.

Модель [отрицательной биномиальной регрессии](https://timeseriesreasoning.com/contents/negative-binomial-regression-model/) (часто называемая смесью Пуассон-гамма) оказывается отличной моделью для многих наборов данных.

Есть множество других подходов к обобщению модели регрессии Пуассона с целью расширения ее действия на чрезмерно и недостаточно рассредоточенные данные. Рассмотрим две такие модели.

# Обобщенная модель регрессии Пуассона по Консулу

В 1989 году Прем С. Консул в своей книге «[Обобщенные распределения Пуассона: свойства и применение](https://www.amazon.com/dp/B01K0UCF5C)» предложил модифицировать пуассоновскую функцию распределения вероятностей так, чтобы она могла обрабатывать как избыточно, так и недостаточно рассредоточенные данные. Эта модель стала известна как модель GP-1 (обобщенная модель Пуассона-1). GP-1 предполагает, что зависимая переменная **y** является случайной величиной со следующим распределением вероятностей:

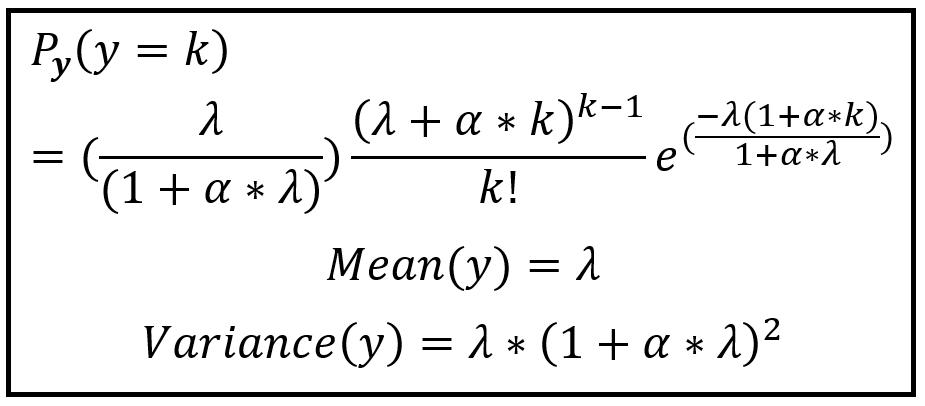
Распределение вероятностей, дисперсия и среднее модели GP-1 (Изображение [автора](https://www.linkedin.com/in/sachindate/))

При установке параметра α равным 0 в приведенных выше выражениях PMF, среднее значение и дисперсия GP-1 преобразуются в выражения для стандартного распределения Пуассона.

# Ограниченное обобщенное распределение Пуассона по Фамойе

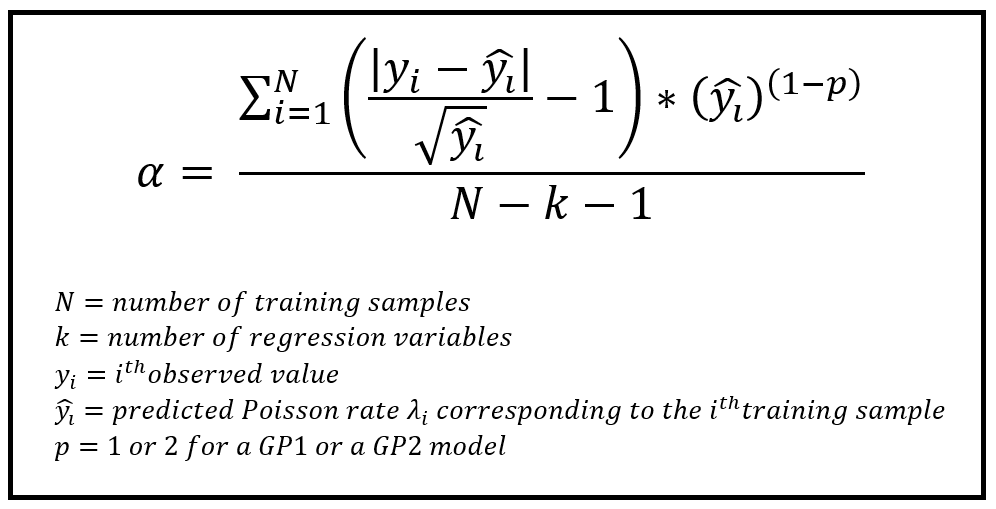
В 1993 году Феликс Фамойе представил то, что он назвал моделью ограниченной обобщенной регрессии Пуассона, в качестве способа расширения стандартной модели Пуассона для обработки чрезмерно и недостаточно рассредоточенных наборов данных. Эта модель стала известна как модель GP-2 (обобщенная модель Пуассона-2).

GP-2 предполагает, что зависимая переменная **y** является случайной величиной со следующим распределением вероятностей:

Распределение вероятностей, дисперсия и среднее модели GP-2 (Изображение [автора](https://www.linkedin.com/in/sachindate/))

Как и ранее, когда параметр дисперсии α равен 0, PMF, среднее и дисперсия GP-2 сводятся к стандартному распределению Пуассона.

Параметр дисперсии α оценивается по следующей формуле:

Формула для расчета параметра дисперсии α в моделях GP-1 и GP-2 (Изображение [автора](https://www.linkedin.com/in/sachindate/))

В библиотеке statsmodels α представляется как элемент настройки моделей GP-1 и GP-2.

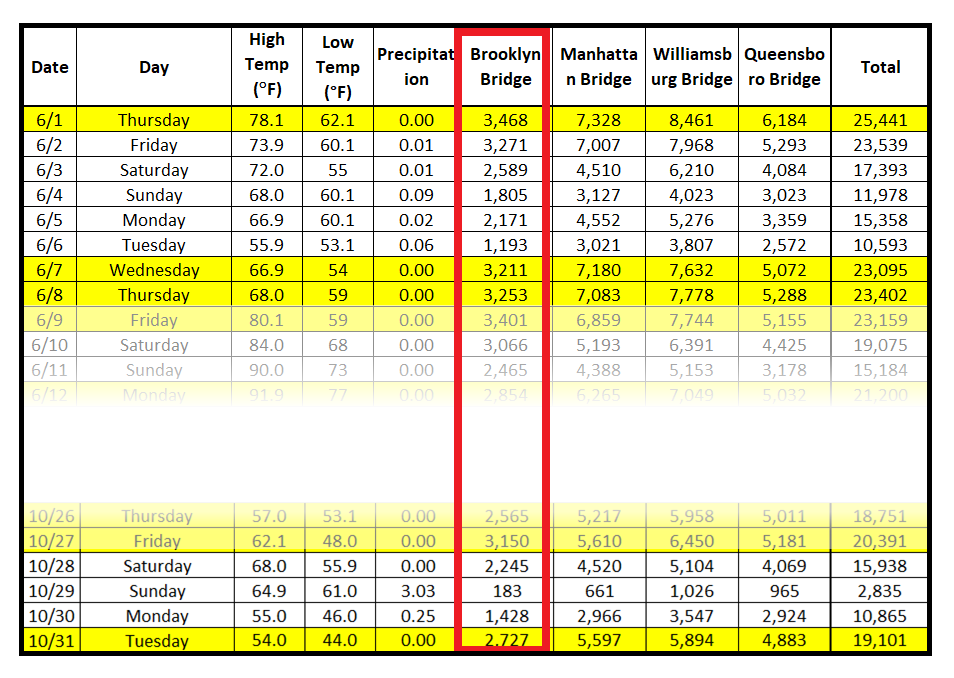
# Реализация GP-1 и GP-2 в Python и Statsmodels

Библиотека statsmodels содержит реализацию моделей GP-1 и GP-2 посредством класса statsmodels.discrete.discrete\_model.GeneralizedPoisson.

В этом разделе покажем, как использовать GP-1 и GP-2 для моделирования реального набора счетных данных.

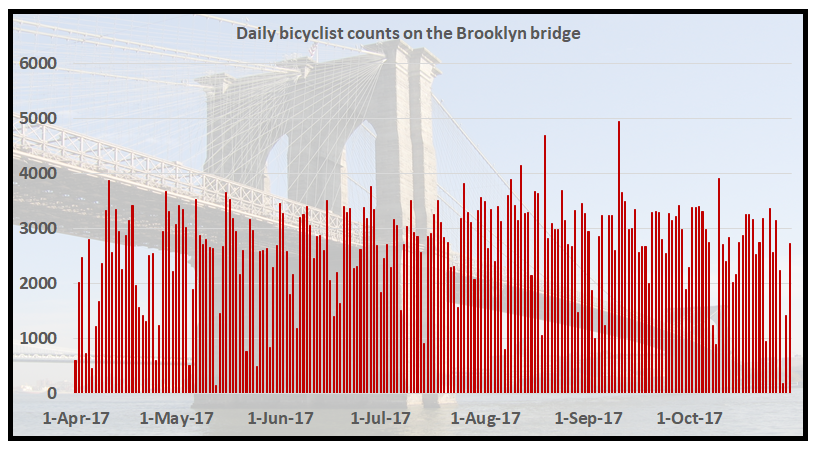
# Набор данных из реального мира

В следующей таблице представлено число велосипедистов, проезжавших по различным мостам Нью-Йорка. Учет проводился ежедневно с 1 апреля по 31 октября 2017 года.

Источник: Количество [велосипедов на мостах Ист-Ривер](https://data.cityofnewyork.us/Transportation/Bicycle-Counts-for-East-River-Bridges/gua4-p9wg) (NYC OpenData) (Изображение [автора](https://www.linkedin.com/in/sachindate/))

[Ссылка на исходный набор данных](https://data.cityofnewyork.us/Transportation/Bicycle-Counts-for-East-River-Bridges/gua4-p9wg).

Проанализируем количестве велосипедистов, ежедневно проезжающих по Бруклинскому мосту. На рисунке график подсчета велосипедистов на Бруклинском мосту. [Ссылка](https://gist.github.com/sachinsdate/c17931a3f000492c1c42cf78bf4ce9fe) на данные подсчета велосипедистов для Бруклинского моста, которые мы и будем анализировать.

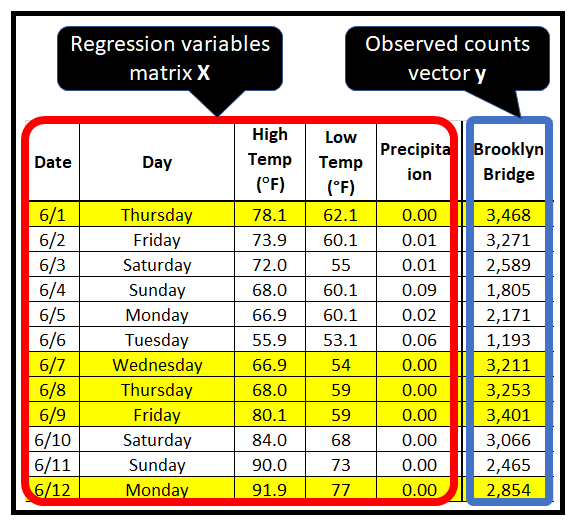
Ежедневное количество велосипедистов на Бруклинском мосту (фоновый источник: вид на [Бруклинский мост с острова Манхэттен](https://en.wikipedia.org/wiki/Brooklyn_Bridge#/media/File:Brooklyn_Bridge_Postdlf.jpg)).

# Цель регрессии

Цель регрессии – предсказать количество велосипедистов, проезжающих по Бруклинскому мосту в произвольный день.

# Стратегия регрессии

Будем использовать набор регрессионных переменных из набора данных, а именно: День, День недели (получаем от переменной Дата), Месяц (получаем от переменной Дата), Высокая температура, Низкая температура и Осадки для «объяснения» дисперсии наблюдаемых значений на Бруклинском мосту.

Матрица регрессии и вектор количества наблюдаемых велосипедистов (Изображение [автора](https://www.linkedin.com/in/sachindate/) )

Алгоритм обучения нашей регрессионной модели будет настраивать наблюдаемую величину у на матрицу регрессии X.

После обучения модели мы протестируем ее эффективность на тестовом наборе, который модель не видела во время обучения.

# Стратегия регрессии – шаг за шагом

1. Сначала на этом наборе данных. обучим стандартную модель регрессии Пуассона Стандартная модель Пуассона будет служить «контрольной» моделью для проверки эффективности обобщенных моделей Пуассона.
2. Если дисперсия в наборе данных примерно такая же, как и среднее значение, то больше ничего не нужно делать, кроме как проверить соответствие модели Пуассона.
3. Если дисперсия окажется больше или меньше среднего Пуассона, мы последовательно обучим модели GP-1 и GP-2 и сравним их эффективность со стандартной моделью.

Начнем с импорта всех необходимых пакетов.

import pandas as pd

from patsy import dmatrices

import numpy as np

import statsmodels.api as sm

import matplotlib.pyplot as plt

Создадим DataFrame для набора данных. [Ссылка на набор данных](https://gist.github.com/sachinsdate/c17931a3f000492c1c42cf78bf4ce9fe).

df = pd.read\_csv('nyc\_bb\_bicyclist\_counts.csv', header=0, infer\_datetime\_format=True, parse\_dates=[0], index\_col=[0])

Добавим несколько производных, полученных из переменной Дата, регрессионных переменных в матрицу **X**.

ds = df.index.to\_series()

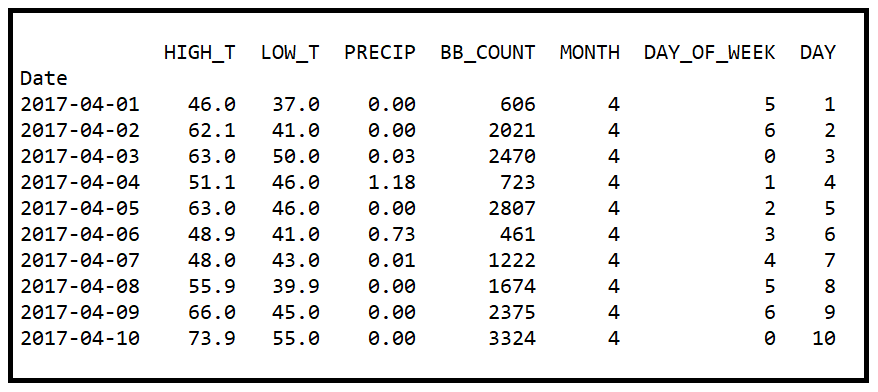
df['MONTH'] = ds.dt.month

df['DAY\_OF\_WEEK'] = ds.dt.dayofweek

df['DAY'] = ds.dt.day

Выведем первые несколько строк нашего набора.

print(df.head(10))

(Изображение [автора](https://www.linkedin.com/in/sachindate/))

Создадим наборы данных для обучения и тестирования.

mask = np.random.rand(len(df)) < 0.8

df\_train = df[mask]

df\_test = df[~mask]

print('Training data set length='+str(len(df\_train)))

print('Testing data set length='+str(len(df\_test)))

Зададим выражение регрессии в нотации [patsy](https://patsy.readthedocs.io/en/latest/quickstart.html) . Скажем patsy, что BB\_COUNT – наша зависимая переменная **y**, и она зависит от переменных регрессии **X**: DAY, DAY\_OF\_WEEK, MONTH, HIGH\_T, LOW\_T и PRECIP.

expr = 'BB\_COUNT ~ DAY + DAY\_OF\_WEEK + MONTH + HIGH\_T + LOW\_T + PRECIP'

Воспользуемся patsy, чтобы выделить матрицы X и у для обучающего и тестового наборов.

y\_train, X\_train = dmatrices(expr, df\_train, return\_type='dataframe')

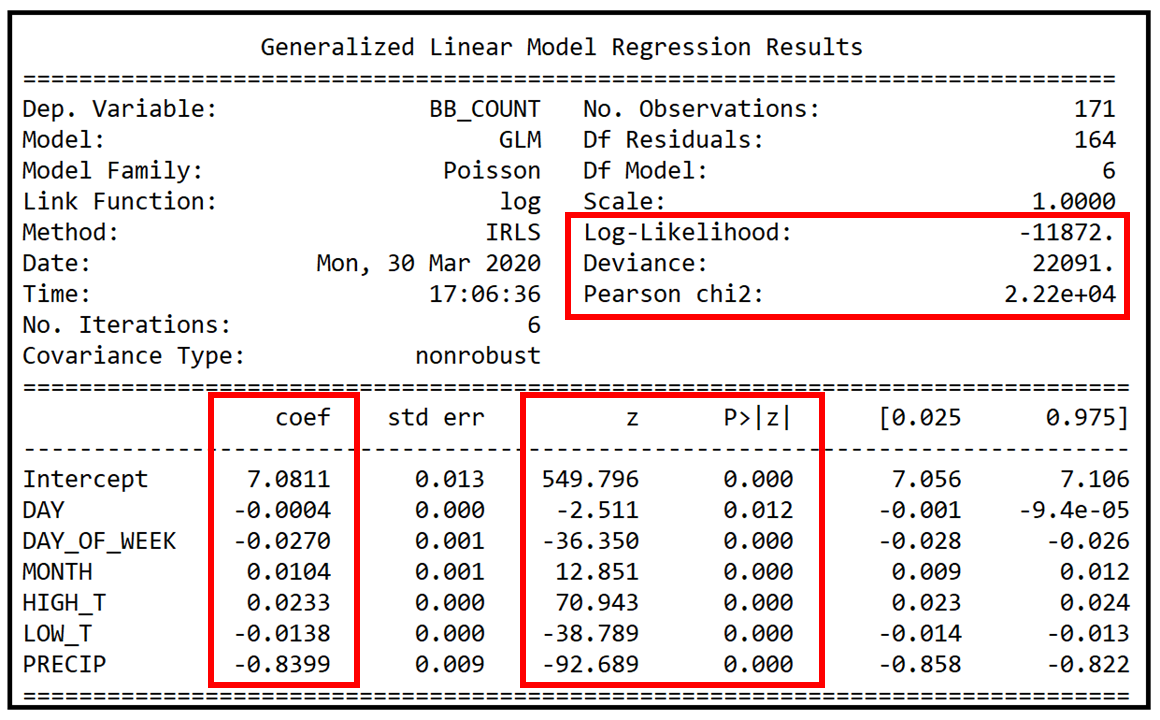
y\_test, X\_test = dmatrices(expr, df\_test, return\_type='dataframe')

Используя класс GLM из statsmodels, обучим модель регрессии Пуассона.

poisson\_training\_results = sm.GLM(y\_train, X\_train, family=sm.families.Poisson()).fit()

Выведем сводный результат обучения.

print(poisson\_training\_results.summary())

Сводка обучения модели регрессии Пуассона (Изображение [автора](https://www.linkedin.com/in/sachindate/))

На сводке выделены некоторые важные для нас элементы.

Видно, что все коэффициенты регрессии (известные как вектор **β** на языке регрессии) статистически значимы на уровне достоверности 95%, поскольку их значение p < 0,05.

Значение максимального логарифмического правдоподобия (-11872) используем позже при сравнении эффективности стандартной модели с моделями GP-1 и GP-2.

Выведем дисперсию и среднее значение.

print('variance='+str(df['BB\_COUNT'].var()))

print('mean='+str(df['BB\_COUNT'].mean()))

Получим.

variance=730530.6601948135

mean=2680.042056074766

Дисперсия существенно больше среднего. Данные сильно рассредоточены, и основное предположение модели Пуассона не выполняется.

Построим и обучим модели GP-1 и GP-2 и посмотрим результат.

statsmodels позволяет сделать это в 3 строчки кода!

Построим модель регрессии Generalized Poison Consul, известную как GP-1.

gen\_poisson\_gp1 = sm.GeneralizedPoisson(y\_train, X\_train, p=1)

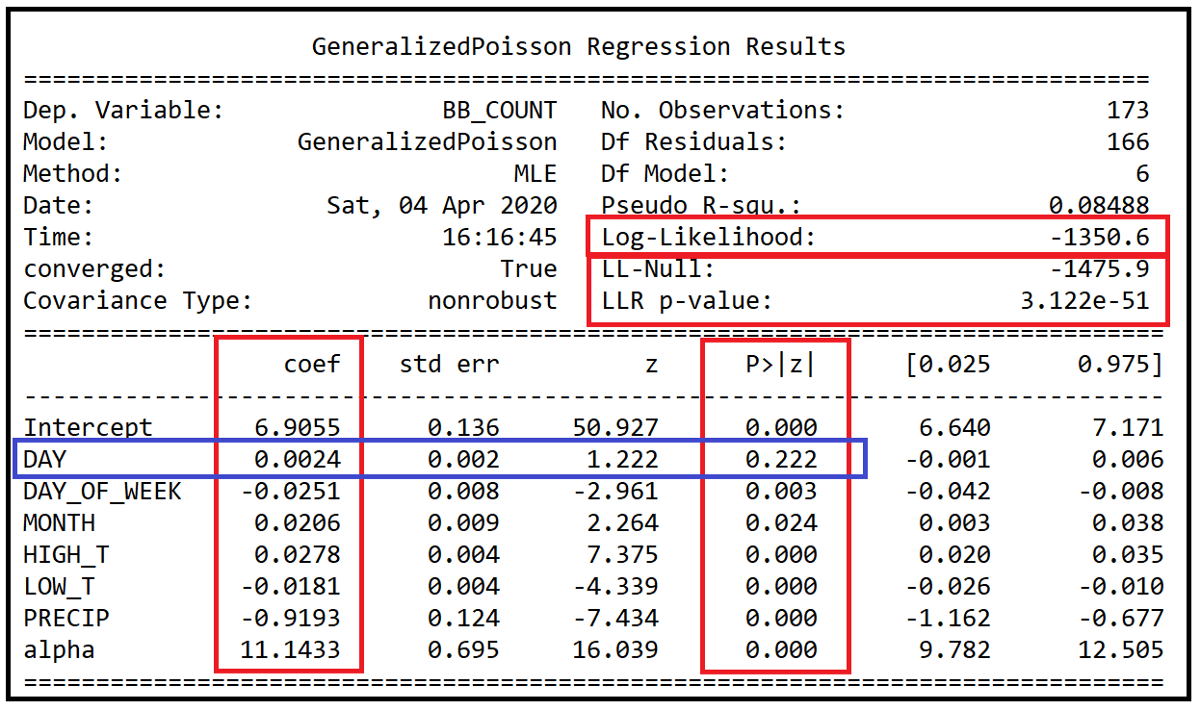
Обучим.

|  |
| --- |
| gen\_poisson\_gp1\_results = gen\_poisson\_gp1.fit() |

Выведем результаты.

print(gen\_poisson\_gp1\_results.summary())

На рисунке выделены значимые для нас элементы.

Результаты обучения модели GP-1 (Изображение [автора](https://www.linkedin.com/in/sachindate/))

Обратим внимание, что, за исключением коэффициента переменной DAY, все остальные коэффициенты регрессии статистически значимы на уровне достоверности 95%, т. к. их p-значение < 0,05. Помним, что в модели Пуассона все коэффициенты регрессии оказались статистически значимыми на уровне достоверности 95%.

# Степень соответствия модели GP-1

Оценка максимального правдоподобия (Maximum Likelihood Estimate, MLE) модели GP-1 составляет -1350,6, что больше, чем у MLE нулевой модели -1475,9. Нулевая модель – это простейшая модель, включающая только свободный член из модели регрессии, т.е. это горизонтальная линия, проходящая через точку пересечения по оси **y** (intercept-only model). Но является ли разница MLE статистически значимой? Показано, что p-значение теста отношения правдоподобия (LR) составляет 3,12e-51. Это очень маленькое число. Так что да, модель GP-1 действительно лучше справляется с моделированием данных, чем нулевая модель.

Но работает ли GP-1лучше обычной модели Пуассона?

Вспомним, что оценка максимального правдоподобия для обычной модели Пуассона составляла -11872. Сравним с MLE GP-1, которая составляет -1350,6. Ясно, что GP-1 дает превосходную степень соответствия. При желании можно провести тест LR, используя две оценки правдоподобия, и оценить значимость, используя критерий хи-квадрат. В нашем случае это необязательно, поскольку MLE GP-1 существенно больше MLE обычной модели Пуассона.

# Прогноз

Спрогнозируем количество велосипедистов с помощью GP-1, используя тестовый набор, который модель не видела во время обучения.

gen\_poisson\_gp1\_predictions = gen\_poisson\_gp1\_results.predict(X\_test)

gen\_poisson\_gp1\_predictions – объект Series из pandas, который содержит прогнозируемое количество велосипедистов для каждой строки в матрице X\_test. Помним, что y\_test – фактические наблюдаемые значения.

Отобразим на одном графике прогноз и факт для визуальной оценки качества прогноза.

predicted\_counts=gen\_poisson\_gp1\_predictions

actual\_counts = y\_test['BB\_COUNT']

fig = plt.figure()

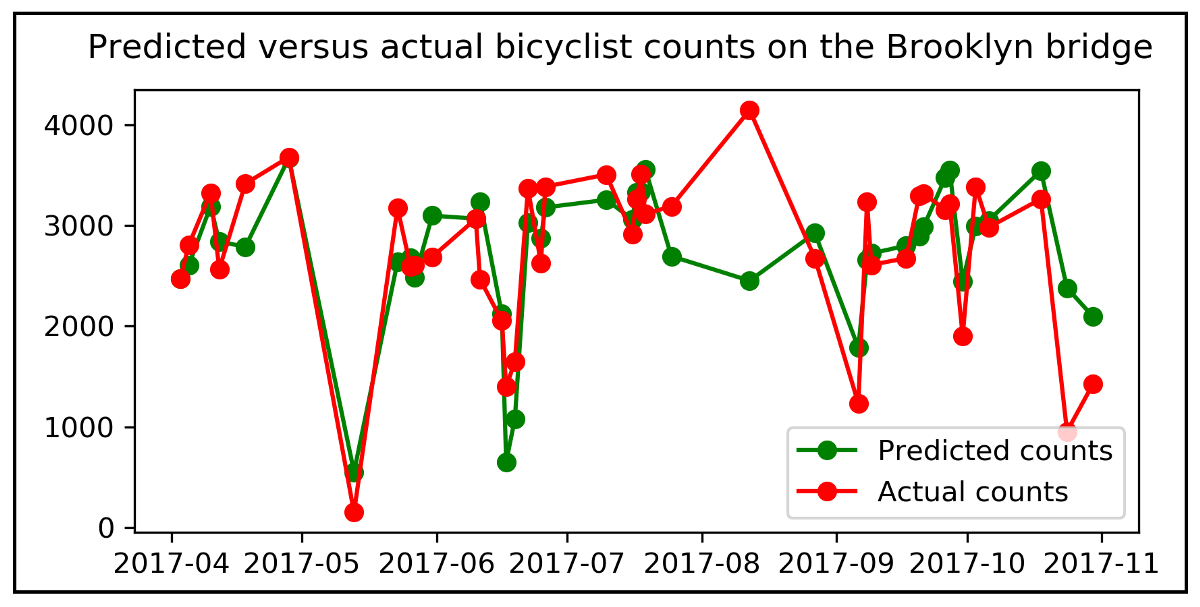
fig.suptitle('Predicted versus actual bicyclist counts on the Brooklyn bridge')

predicted, = plt.plot(X\_test.index, predicted\_counts, 'go-', label='Predicted counts')

actual, = plt.plot(X\_test.index, actual\_counts, 'ro-', label='Actual counts')

plt.legend(handles=[predicted, actual])

plt.show()

Прогноз и факт для модели GP-1 (Изображение [автора](https://www.linkedin.com/in/sachindate/))

Видим, что за редким исключением модель GP-1 сработала достаточно хорошо при прогнозировании ежедневного количества велосипедистов.

При необходимости можем сравнить качество прогноза GP-1 и обычной модели Пуассона, используя среднюю квадратичную ошибку (Root Mean Square Error, RMSE). Оставим это в качестве упражнения для читателя.

Рассмотрим предложенную Фамойе модель ограниченной обобщенной регрессии Пуассона (Famoye’s Restricted Generalized Poisson regression model), известную как GP-2.

Применим аналогичную 3-х шаговую процедуру для построения и обучения модели. В данном случае параметр p=2.

#Build Famoye's Restricted Generalized Poison regression model, know as GP-2

gen\_poisson\_gp2 = sm.GeneralizedPoisson(y\_train, X\_train, p=2)

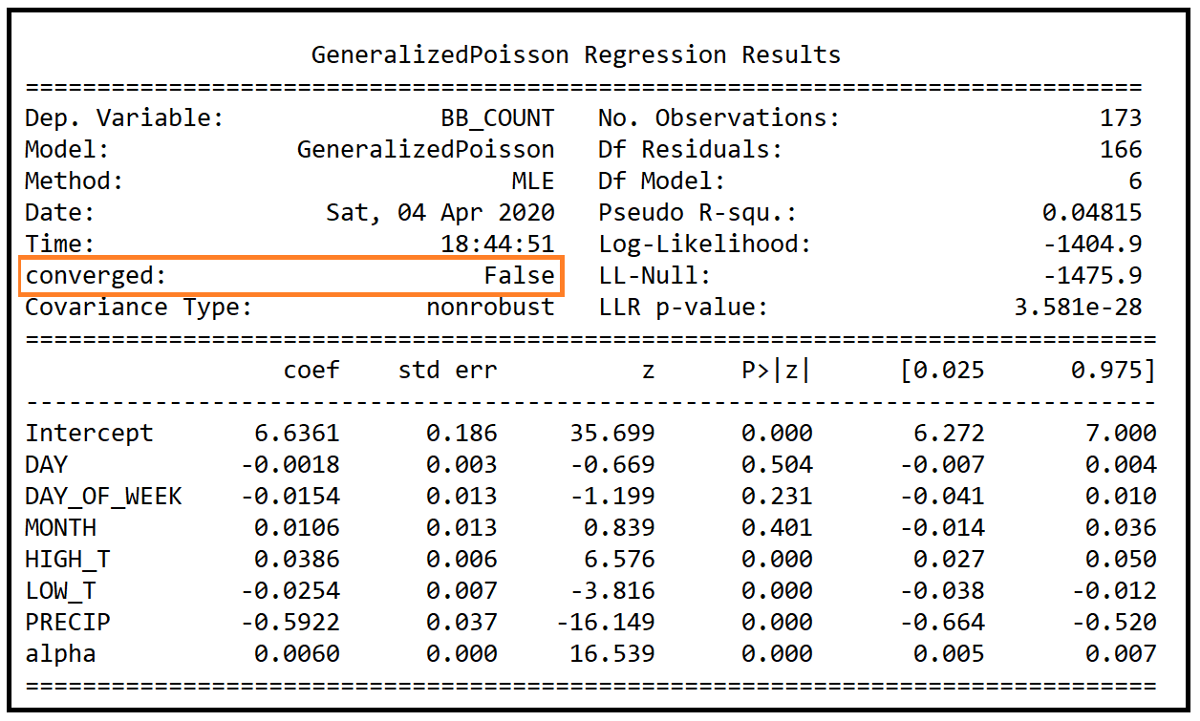
#Fit the model

gen\_poisson\_gp2\_results = gen\_poisson\_gp2.fit()

#print the results

print(gen\_poisson\_gp2\_results.summary())

Результат обучения.



Результат обучения модели GP-2 (Изображение [автора](https://www.linkedin.com/in/sachindate/))

Способ анализа и сравнения критериев соответствия и качества прогноза GP-2 такой же, как и в случае с GP-1. Однако обратим внимание, что в этом случае алгоритм обучения модели не сошелся. Можно попытаться решить эту проблему, установив большее количество итераций в параметре iter метода fit().

# Исходный код

import pandas as pd

from patsy import dmatrices

import numpy as np

import statsmodels.api as sm

import matplotlib.pyplot as plt

#Create a pandas DataFrame for the counts data set.

df = pd.read\_csv('nyc\_bb\_bicyclist\_counts.csv', header=0, infer\_datetime\_format=True, parse\_dates=[0], index\_col=[0])

#We'll add a few derived regression variables to the X matrix.

ds = df.index.to\_series()

df['MONTH'] = ds.dt.month

df['DAY\_OF\_WEEK'] = ds.dt.dayofweek

df['DAY'] = ds.dt.day

#Let's print out the first few rows of our data set to see how it looks like

print(df.head(10))

#Let's create the training and testing data sets.

mask = np.random.rand(len(df)) < 0.8

df\_train = df[mask]

df\_test = df[~mask]

print('Training data set length='+str(len(df\_train)))

print('Testing data set length='+str(len(df\_test)))

#Setup the regression expression in Patsy notation.

#We are telling patsy that BB\_COUNT is our dependent variable y and it depends on the regression

variables X:

#DAY, DAY\_OF\_WEEK, MONTH, HIGH\_T, LOW\_T and PRECIP.

expr = 'BB\_COUNT ~ DAY + DAY\_OF\_WEEK + MONTH + HIGH\_T + LOW\_T + PRECIP'

#Let's use Patsy to carve out the X and y matrices for the training and testing data sets:

y\_train, X\_train = dmatrices(expr, df\_train, return\_type='dataframe')

y\_test, X\_test = dmatrices(expr, df\_test, return\_type='dataframe')

#Using the statsmodels GLM class, train the Poisson regression model on the training data set.

poisson\_training\_results = sm.GLM(y\_train, X\_train, family=sm.families.Poisson()).fit()

#Print the training summary.

print(poisson\_training\_results.summary())

#Let's print out the variance and mean of the data set

print('variance='+str(df['BB\_COUNT'].var()))

print('mean='+str(df['BB\_COUNT'].mean()))

#Build Consul's Generalized Poison regression model, know as GP-1

gen\_poisson\_gp1 = sm.GeneralizedPoisson(y\_train, X\_train, p=1)

#Fit the model

gen\_poisson\_gp1\_results = gen\_poisson\_gp1.fit()

#print the results

print(gen\_poisson\_gp1\_results.summary())

#Get the model's predictions on the test data set

gen\_poisson\_gp1\_predictions = gen\_poisson\_gp1\_results.predict(X\_test)

predicted\_counts=gen\_poisson\_gp1\_predictions

actual\_counts = y\_test['BB\_COUNT']

fig = plt.figure()

fig.suptitle('Predicted versus actual bicyclist counts on the Brooklyn bridge')

predicted, = plt.plot(X\_test.index, predicted\_counts, 'go-', label='Predicted counts')

actual, = plt.plot(X\_test.index, actual\_counts, 'ro-', label='Actual counts')

plt.legend(handles=[predicted, actual])

plt.show()

#Build Famoye's Restricted Generalized Poison regression model, know as GP-2

gen\_poisson\_gp2 = sm.GeneralizedPoisson(y\_train, X\_train, p=2)

#Fit the model

gen\_poisson\_gp2\_results = gen\_poisson\_gp2.fit()

#print the results

print(gen\_poisson\_gp2\_results.summary())

[view raw](https://gist.github.com/sachinsdate/a1645ee37a13292aa58f4df5d105710a/raw/8410f006a75fefbb72e89944bc5e78113d9e8ecb/generalized_poisson_regression.py)[generalized\_poisson\_regression.py](https://gist.github.com/sachinsdate/a1645ee37a13292aa58f4df5d105710a#file-generalized_poisson_regression-py) hosted with by [GitHub](https://github.com/)

[Ссылка](https://gist.github.com/sachinsdate/c17931a3f000492c1c42cf78bf4ce9fe) на набор данных.

# Резюме

* Стандартная модель Пуассона предполагает, что дисперсия счетных данных

такая же, как и среднее значение. Это предположение часто

нарушается наборами данных реального мира, которые либо чрезмерно,

либо недостаточно рассредоточены.

* Следовательно, нам необходимо использовать другие модели для счетных

данных, как, например, модель отрицательной биномиальной регрессии или

модели обобщенной регрессии Пуассона, которые не требуют, чтобы

данные являлись эквидисперсными. Такие модели предполагают,

что дисперсия является некоторой функцией среднего.

* Модель обобщенной регрессии Пуассона Консула (называемая GP-1) и

модель ограниченной обобщенной регрессии Пуассона Фамойе (GP-2) – две

такие модели, которые можно использовать для моделирования реальных наборов

счетных данных.

* Библиотека Python Statsmodels отлично подходит для построения и обучения моделей GP-1 и GP-2.

# Ссылки, цитаты и авторские права

## **Набор данных**

[Подсчеты велосипедистов для мостов Ист-Ривер.](https://data.cityofnewyork.us/Transportation/Bicycle-Counts-for-East-River-Bridges/gua4-p9wg) Ежедневный подсчет велосипедистов, проводимый ежемесячно на Бруклинском мосту, Манхэттенском мосту, Вильямсбургском мосту и мосту Квинсборо. Из открытых данных Нью-Йорка в

соответствии с Условиями использования. [Набор данных для загрузки.](https://gist.github.com/sachinsdate/c17931a3f000492c1c42cf78bf4ce9fe)

## **Статьи**

П.С. Консул и Феликс Фамойе (1992) Обобщенная модель регрессии Пуассона, *Коммуникации в статистике – Теория и методы* , 21: 1, 89-109, DOI: [10.1080 / 03610929208830766](https://doi.org/10.1080/03610929208830766)

P.C. Consul & Felix Famoye (1992) Generalized poisson regression model, Communications in Statistics – Theory and Methods, 21:1, 89-109, DOI: [10.1080/03610929208830766](https://doi.org/10.1080/03610929208830766)

## **Книги**

Кэмерон А.С. и Триведи П.К., [Регрессионный анализ счетных данных](http://faculty.econ.ucdavis.edu/faculty/cameron/racd2/), второе издание, *Монография Эконометрического общества № 53* , Cambridge University Press, Кембридж, май 2013 г.

П.С. Консул, (1989), [Обобщенные распределения Пуассона: свойства и приложения](https://www.worldcat.org/title/generalized-poisson-distributions-properties-and-applications/oclc/18441905) , Нью-Йорк, *Марсель Деккер*.

Cameron A. C. and Trivedi P. K., [Regression Analysis of Count Data](http://faculty.econ.ucdavis.edu/faculty/cameron/racd2/), Second Edition, Econometric Society Monograph No. 53, Cambridge University Press, Cambridge, May 2013.

Consul, P.C. (1989), [Generalized Poisson Distributions: Properties and Applications](https://www.worldcat.org/title/generalized-poisson-distributions-properties-and-applications/oclc/18441905), New York, Marcel Dekker.

## **Изображения**

Все изображения являются собственностью [Sachin Date в](https://www.linkedin.com/in/sachindate/) соответствии с [CC-BY-NC-SA](https://creativecommons.org/licenses/by-nc-sa/4.0/) , если под изображением не указаны другой источник и авторские права.