A Project Report

On

STATISTICS OF GOLDBACH CONJECTURE

BY

RADHESH SARMA

2017B4A70886H

Under the supervision of

Prof. MICHAEL ALPHONSE

SUBMITTED IN PARTIAL FULLFILLMENT OF THE REQUIREMENTS OF

MATH F376 : DESIGN ORIENTED PROJECT

BIRLA INSTITUTE OF TECHNOLOGY AND SCIENCE PILANI (RAJASTHAN)

HYDERABAD CAMPUS

(OCTOBER 2019)

ACKNOWLEDGMENTS

I express my gratitude towards the AUGSD Division of BITS Pilani Hyderabad Campus for giving me an opportunity to see the research work that happens in our academic institution .

I would like to thank Prof. Michael Alphonse ,Department of Mathematics ,BITS Pilani Hyderabad campus for providing us with a project opportunity. He has been extremely supportive and has always encouraged to excel. He has never failed to give the right advice and has been perfect support to us,both in professional and non-professional matters

Birla Institute of Technology and Science-Pilani,

Hyderabad Campus

Certificate

This is to certify that the project report entitled "**STATISTICS ON GOLDBACH CONJECTURE**" submitted by Mr RADHESH SARMA (ID No. 2017B4A70886H) in partial fulfillment of the requirements of the course MATH F376, Design Project Course, embodies the work done by him under my supervision and guidance.

Date: 17th October 2019

(Prof. Michael Alphonse)

BITS- Pilani, Hyderabad Campus

ABSTRACT

The aim of this paper is to exploit statistical nature of prime numbers in the attempt to improve existing integer factorization algorithms. We take a statistical approach towards the Goldbach conjecture. The number of possible representations of even numbers as sum of two primes, i.e. the number of Goldbach partitions, is plotted against the even numbers. From the plot, two important observations are made. First, we see a clear distinction in the set of values attained by multiples of 4 and multiples of 6. Second, we see a positive trend in the number of Goldbach partitions from which we conclude that the Goldbach conjecture is very likely to be true.

CONTENTS

Introduction

We begin our study by exploring one of the oldest unsolved problems in mathematics, the Goldbach conjecture. We stray along this path in an attempt to try and find any patterns in the prime number sequence. This could give us an insight into probable prime factors of a number.

The Goldbach conjecture states that 'Every even number greater than 2 can be written as the sum of two primes'. The two primes are not necessarily distinct. Formally, the conjecture states,

$$\forall n>1,\ \exists\ p1,p2 \text{ where } p1,p2 \text{ are prime, s.t } 2n = p1+p2$$

This is trivially true when n is prime as 2n = n+n fulfils the criteria. So it is enough to check for all n's that are composite.

There are several different forms of the conjecture present in literature. Few of them are,

Every integer greater than 5 can be written as the sum of three primes.

Every even integer greater than or equal to 6 can be written as the sum of two odd primes.

Every odd integer greater than or equal to 9 can be written as the sum of three odd primes.

We shall proceed with the one stated originally.

The Goldbach conjecture has been verified for all even numbers less than or equal to 4×10^{18}. This is very suggestive that the conjecture might be true.

A different approach

Tackling the problem directly is not very easy so we shall see if there are any emergent patterns if the problem is viewed differently. One interesting way is to look at the number of partitions of an even number into two primes. Such a partition is called the Goldbach partition. A partition of natural number n is any combination of natural numbers whose sum is n. The Goldbach partitions for few even numbers are given below.

4 = 2+2

6 = 3+3

8 = 3+5

10 = 3+7, 5+5

12 = 5+7

Doing this manually is clearly impractical so we will write a computer code that takes an even number as input and outputs the number of Goldbach partitions. We will then plot the number of Goldbach partitions in y-axis against the corresponding even number on the x-axis.

The Code

We first create a c++ file prime.cpp in which we store a list of primes till the desired range 1,n. The list of primes is obtained from existing datasets online. We then create a map between even numbers in the range 4,n and the corresponding number of Goldbach partitions. The number of Goldbach partitions is returned by the function fun.

```
int fun(int n)
  {
    //this returns the number of pairs of primes (i,j) such that i<=j and i and j are primes and i+j = n

    int ans=0;
    int j=0;

    // if(n%2==1)
    // {
    //  return bool(binary_search(all(prime),n-2));
    // }
    for(auto i:prime)
    {

      if(n-i==i)
      {
        ans++;
        j++;
      }
      else if (binary_search(prime.begin()+j,prime.end(),n-i))
      {

        ans++;
        j++;
      }
    }

```

```
        return ans;
    }
```

Click here for complete code: www.pastebin.com/SXGBUcEK

We then store the number of Goldbach partitions against the even numbers in a .csv file.

Next we use python code to create a scatter plot of the data.

```
import pandas as pd
import matplotlib.pyplot as plt
from sklearn.linear_model import LinearRegression
from sklearn.preprocessing import PolynomialFeatures

def linearRegression(data, MultiplesofWhat):
    print("Showing Scatter Plot and best Fit Line using Linear Regression ")
    MultiplesofWhat = MultiplesofWhat / 2
    X = data.iloc[:, 0].values.reshape(-1, 1)
    y = data.iloc[:, 1].values.reshape(-1, 1)
    X = [X[i] for i in range(len(X)) if i % MultiplesofWhat == 0]
    y = [y[i] for i in range(len(y)) if i % MultiplesofWhat == 0]
    linear_regressor = LinearRegression()
    linear_regressor.fit(X, y)
    y_pred = linear_regressor.predict(X)
    plt.scatter(X, y, color='blue')
    plt.plot(X, y_pred, color='red')
    plt.title('Data about Goldbach Conjecture')
    plt.xlabel('Even Integers')
    plt.ylabel('Number of pairs of prime numbers (i,j) such that i<j and i+j = x
')
    # plt.savefig('goldbach Statistical Results .pdf',format='pdf')
    # plt.savefig('Goldbach Statistical Results.png',dpi=1200)
    plt.show()
    return


def PolynomialRegression(data, n, Transparency, MultiplesofWhat):
    MultiplesofWhat = MultiplesofWhat / 2
    X = data.iloc[:, 0].values.reshape(-1, 1)
    y = data.iloc[:, 1].values.reshape(-1, 1)
    X = [X[i] for i in range(len(X)) if i % MultiplesofWhat == 0]
    y = [y[i] for i in range(len(y)) if i % MultiplesofWhat == 0]
    polynomial_regressor = PolynomialFeatures(degree=n)
    X_Poly = polynomial_regressor.fit_transform(X)
    polynomial_regressor = LinearRegression()
    polynomial_regressor.fit(X_Poly, y)

    plt.scatter(X, y, color='blue', alpha=Transparency,marker='.')

```

```
        plt.plot(X, polynomial_regressor.predict(X_Poly), color='red')
        plt.title('Data about Goldbach Conjecture')
        plt.xlabel('Multiples of ' + str(MultiplesofWhat * 2))
        plt.ylabel('Number of pairs of prime numbers (i,j) such that i<j and i+j = x')
        # plt.savefig('Goldbach Statistical Results.png', dpi=900)
        #plt.savefig('Goldbach Statistics which are multiples of {0}.png'.format(MultiplesofWhat * 2), dpi=1200)
        plt.show()
        return


    def PolynomialAnalysis(data, n, Transparency):
        X = data.iloc[:, 0].values.reshape(-1, 1)
        y = data.iloc[:, 1].values.reshape(-1, 1)
        polynomial_regressor = PolynomialFeatures(degree=n)
        X_Poly = polynomial_regressor.fit_transform(X)
        polynomial_regressor = LinearRegression()
        polynomial_regressor.fit(X_Poly, y)

        # plt.scatter(X,y,color = 'blue',alpha=Transparency,marker='.')
        X_4and6 = [X[i] for i in range(len(X)) if (i % 2 == 0 and i % 3 == 0)]
        y_4and6 = [y[i] for i in range(len(y)) if (i % 2 == 0 and i % 3 == 0)]
        plt.scatter(X_4and6, y_4and6, color='red', alpha=Transparency,marker='.')

        X_4 = [X[i] for i in range(len(X)) if (i % 2 == 0 and i % 3 != 0)]
        y_4 = [y[i] for i in range(len(y)) if (i % 2 == 0 and i % 3 != 0)]
        plt.scatter(X_4, y_4, color='green', alpha=Transparency,marker='.')

        X_6 = [X[i] for i in range(len(X)) if (i % 3 == 0 and i % 2 != 0)]
        y_6 = [y[i] for i in range(len(y)) if (i % 3 == 0 and i % 2 != 0)]
        plt.scatter(X_6, y_6, color='blue', alpha=Transparency,marker='.')

        # X_8= [X[i] for i in range(len(X)) if (i % 4 == 0 )]
        # y_8 = [y[i] for i in range(len(y)) if (i % 4 ==0)]
        # plt.scatter(X_8, y_8, color='orange', alpha=Transparency)

        plt.plot(X, polynomial_regressor.predict(X_Poly), color='black')
        plt.title('Data about Goldbach Conjecture with Degree of best fit polynomial = ' + str(n))
        plt.xlabel('Multiples of 4 in Green and 6 in Blue,Multiplies of 4 and 6 in Red')
        plt.ylabel('Number of pairs of prime numbers (i,j) such that i<j and i+j = x')
        plt.savefig('Goldbach Statistical Results with Multiples of 4 and 6 .png', dpi=1200)
        plt.show()
        return
```

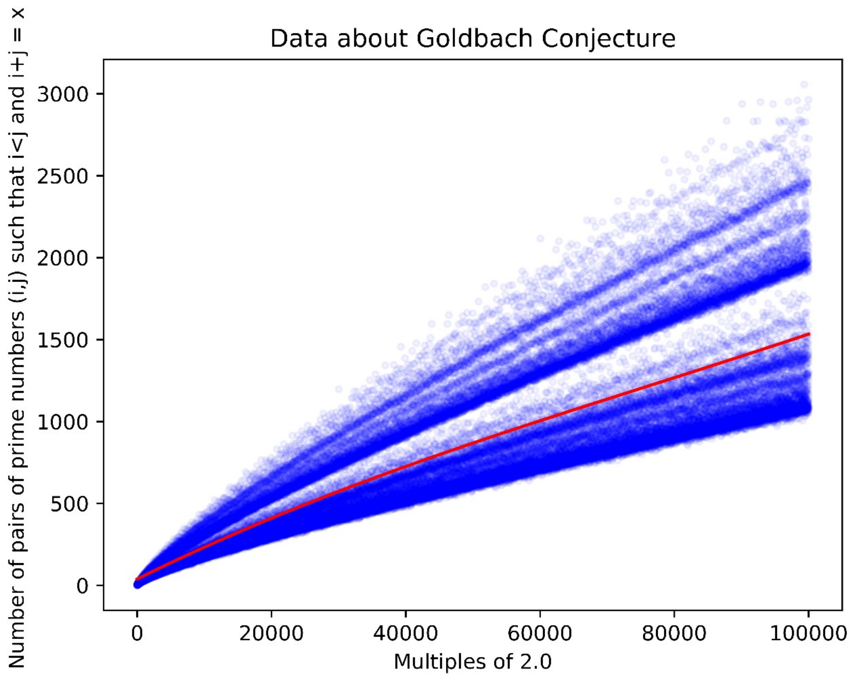
The function LinearRegression plots the best fit straight line through the data points.

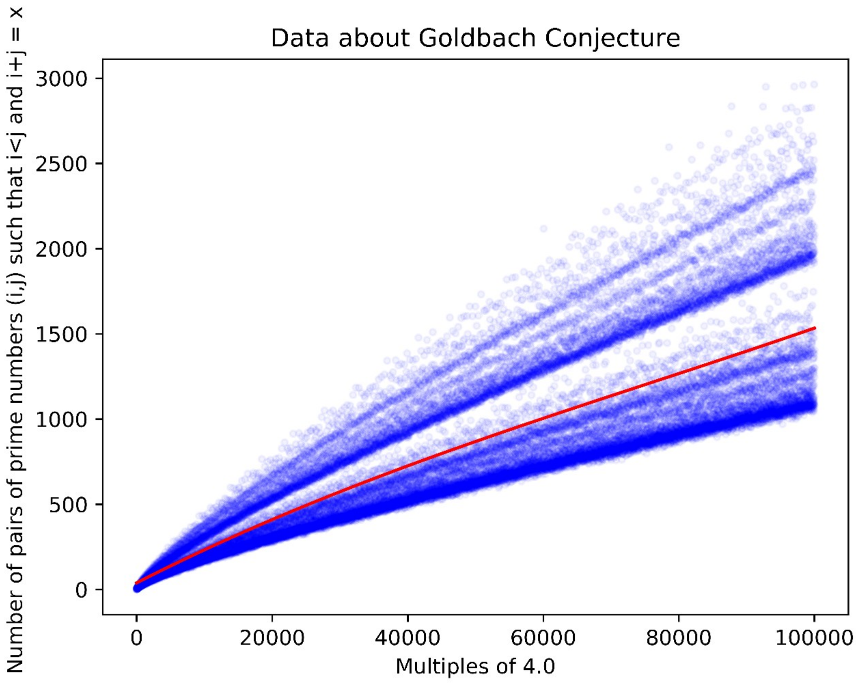
Likewise, the function PolynomialRegression plots the best fit polynomial of degree n through the data points.

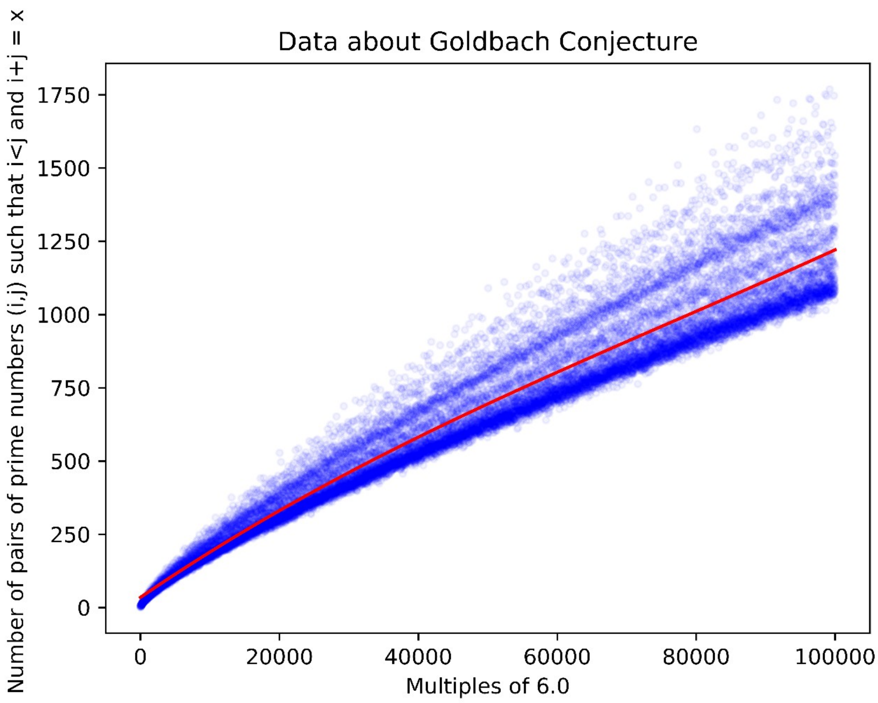
We can plot data sets of different sizes for our visualization purpose.

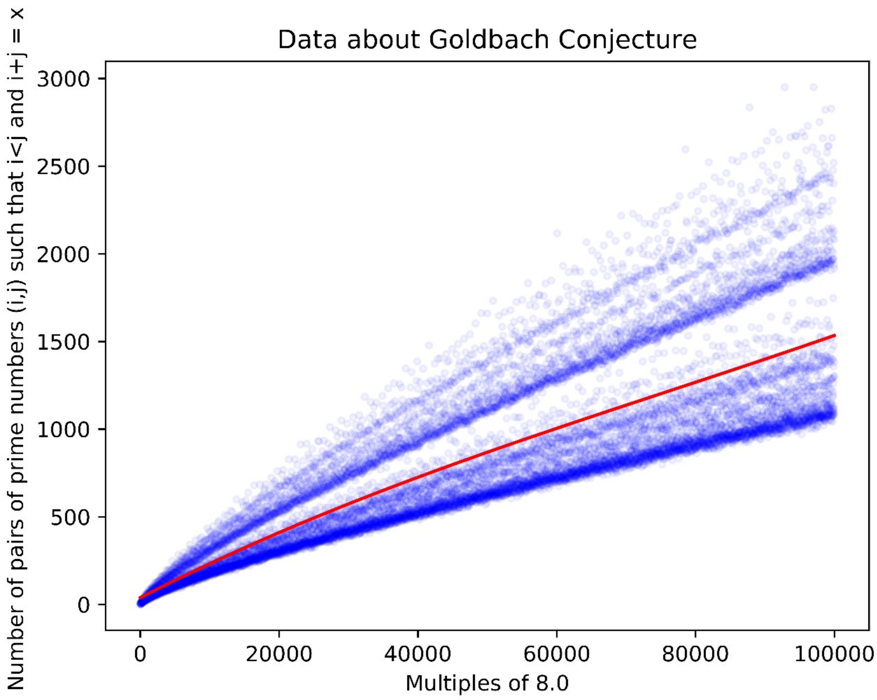
Complete code: https://pastebin.com/Dhduc27H

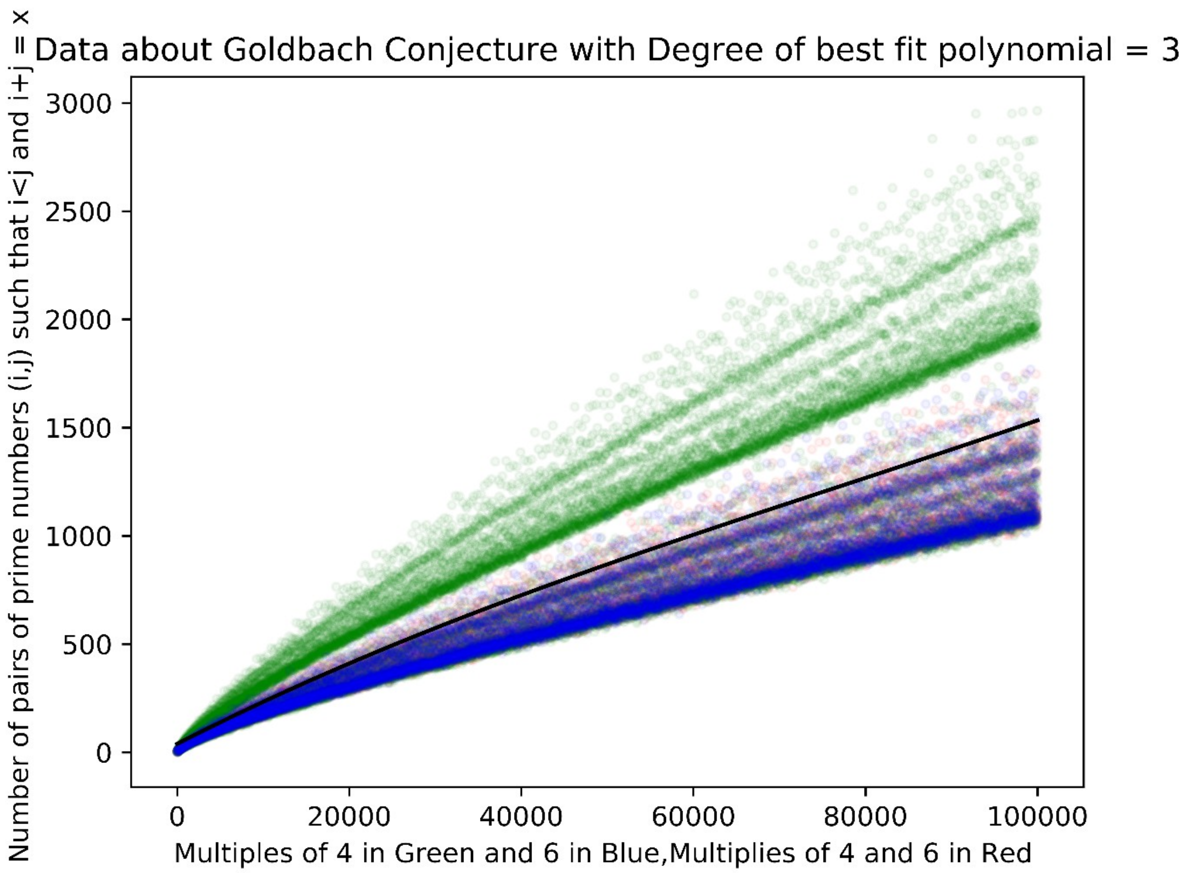
Plots of Goldbach partitions




Data about Goldbach Conjecture
Number of pairs of prime numbers (i,j) such that i<j and i+j = x
0
500
1000
1500
2000
2500
3000
0
20000
40000
60000
80000
100000
Multiples of 4.0


Data about Goldbach Conjecture
Number of pairs of prime numbers (i,j) such that i<j and i+j = x
0
250
500
750
1000
1250
1500
1750
0
20000
40000
60000
80000
100000
Multiples of 6.0


Data about Goldbach Conjecture
Number of pairs of prime numbers (i,j) such that i<j and i+j = x
Multiples of 8.0
0
500
1000
1500
2000
2500
3000
0
20000
40000
60000
80000
100000


Data about Goldbach Conjecture with Degree of best fit polynomial = 3
Number of pairs of prime numbers (i,j) such that i<j and i+j = x
Multiples of 4 in Green and 6 in Blue,Multiplies of 4 and 6 in Red
0
500
1000
1500
2000
2500
3000
0
20000
40000
60000
80000
100000

Observations

From the plots we see that powers of two have similar trend. A very interesting observation is that the trend seems to split into two bands. The band in green is for all multiples of 4. The blue is for multiples of 6. The red is for both multiples of 6 and 4.

Why Goldbach conjecture is probably true

An intuitive argument can be made from a statistical perspective. We will first determine how many of the partitions of an even number are Goldbach partitions. According to the celebrated prime number theorem, any integer x selected at random has approximately 1/ln(x) chance of being a prime. Consider a large enough even integer n and let m be a number between 3 and n/2. Then the probability of both m and n-m to be prime is roughly 1/ln(n-m)*1/ln(m). Summing up the the probabilities by running m from 3 to n/2, we get a rough figure of the number of Goldbach partitions of n as n/2*(ln(n))^2. As n grows faster than ln(n) as n goes to infinity, we expect the number of Goldbach partitions to increase as n increases. This is also evident from the generated plots. Hence its very unlikely that the number of Goldbach partitions for an even integer will suddenly drop to 0. Therefore, we conclude that it is highly probable that for any even integer, there will exist at least one Goldbach partition and hence the Goldbach conjecture directly follows.