ABSTRACT

Ramsey Numbers is an important theorem in combinatorics. It gives the minimum number of vertices that is required to guarantee the existence of a monochromatic clique in a graph of any colour, when each edge is coloured with some color c.

Generally, we always consider 2-coloring when referring to Ramsey Theorem.

Ramsey for 3 is 6. It means the minimum number of vertices required to ensure the existence of a monochromatic 3-clique, i.e, a 3-clique formed entirely with red edges, or a 3-clique formed entirely with blue edges, is equal to 6.

However, finding large Ramsey Numbers is computationally difficult, even with quantum computers. It is algorithm-wise exponential in nature.

Hence, in our experiment, instead of trying to compute exponential algorithms, we try to derive some properties of Ramsey Numbers using existing Social Media Analytics. There is a bijection in both because of the famous Party Problem.

The Party Problem states there should be atleast Ramsey of X number of people in a party to ensure atleast X people know each other or atleast X people do not know each other.

This can be easily seen if we represent people knowing each other with Red edge and people not knowing each other with Blue edge.

In this experiment, we form such a graph of blue and red edges from Facebook data, and try to do a survey of the party problem from a practical angle, to find in real world how many people actually know each other versus how many people not know each other practically in a party.

This experiment is a statistical survey and the result is dependant on the data used.

CHAPTER 1

INTRODUCTION

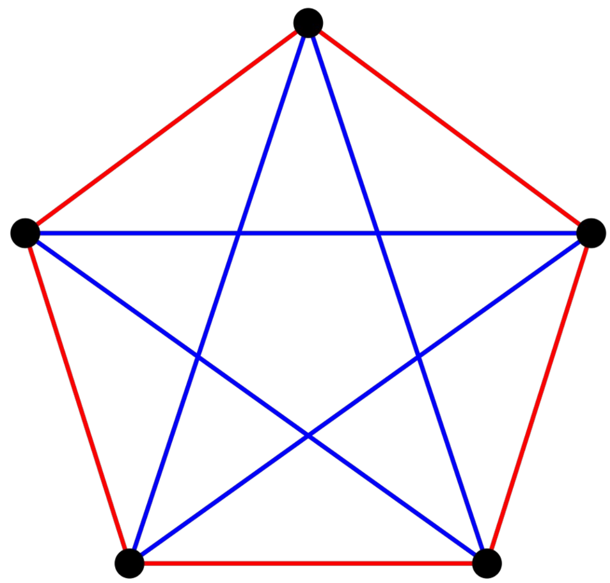
In this Combinatorial Graph Theory and Social Media Analytics research project, we aim to analyze already existing anonymous graph networks taken from social media datas like Facebook and find out how much they tally with Ramsey's theorem.

What is Ramsey's Theorem :

In combinatorics, Ramsey Theorem states that one will always find monochromatic cliques when the edges are coloured in a sufficiently large complete graph. For example, say we have two positive integers r and s. Ramsey-theorem guarantees that there has to be one least positive integer for which every blue-red (in the case of two-coloring) edge coloring of the complete graph of R(r,s) vertices contains a blue r-clique or a red s-clique.

Ramsey-theorem is a fundamental combinatorial result. F. P. Ramsey proved the first version of this result. This sparked the development of the Ramsey theory, a combinatorial theory that seeks regularity in the midst of chaos by establishing conditions which basically prove the existence of regularities amid irregular patterns. It is a question of the presence of monochromatic subsets.

Ramsey Theorem works for more than 2 colours also. The theorem guarantees that for any given number of finite colours we have, when we colour a complete graph with those colours as the edges, we will always find a monochromatic clique of some order if we have sufficient vertices in the graph, that is.



A 2-edge-labeling of K_5 with no monochromatic K_3

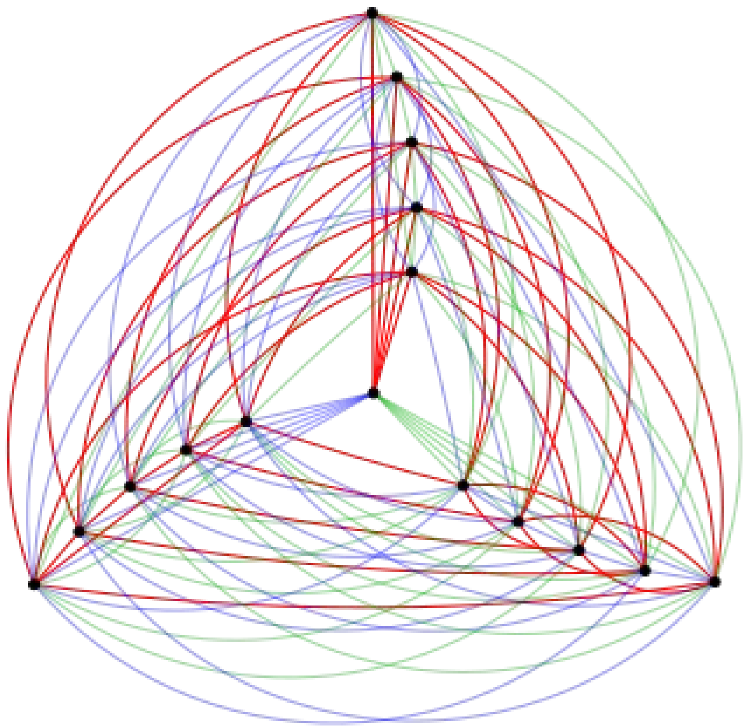
Examples: $R(3, 3) = 6$:

Suppose the edges of a complete graph on 6 vertices are coloured red and blue. Pick a vertex, v. There are 5 edges incident to v and so (by the pigeonhole principle) at least 3 of them must be the same colour. Without loss of generality we can assume at least 3 of these edges, connecting the vertex, v, to vertices, r, s and t, are blue. (If not, exchange red and blue in what follows.) If any of the edges, (r, s), (r, t), (s, t), are also blue then we have an entirely blue triangle. If not, then those three edges are all red and we have an entirely red triangle. Since this argument works for any colouring, *any* K_6 contains a monochromatic K_3, and therefore $R(3, 3) \leq 6$. The popular version of this is called the theorem on friends and strangers.

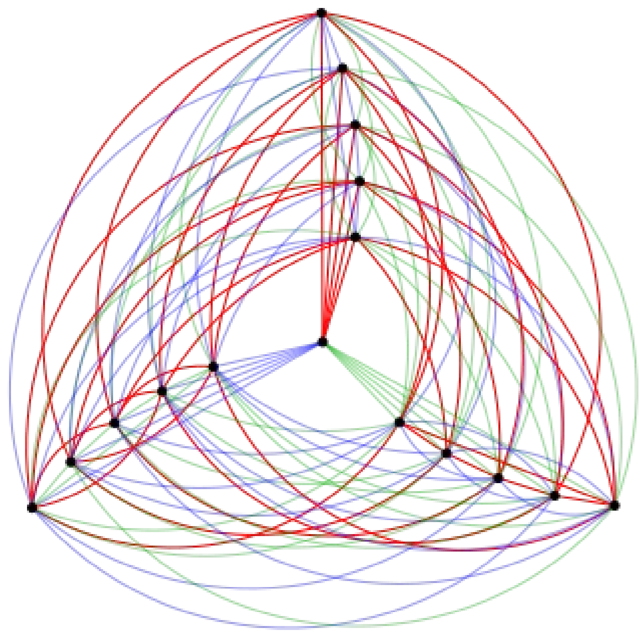
An alternative proof works by double counting. It goes as follows: Count the number of ordered triples of vertices, x, y, z, such that the edge, (xy), is red and the edge, (yz), is blue. Firstly, any given vertex will be the middle of either $0 \times 5 = 0$ (all edges from the vertex are the same colour), $1 \times 4 = 4$ (four are the same colour, one is the other colour), or $2 \times 3 = 6$ (three are the same colour, two are the other colour) such triples. Therefore, there are at most $6 \times 6 = 36$ such triples. Secondly, for any non-monochromatic triangle (**xyz**), there exist precisely two such triples. Therefore, there are at most 18 non-monochromatic triangles. Therefore, at least 2 of the 20 triangles in the K_6 are monochromatic.

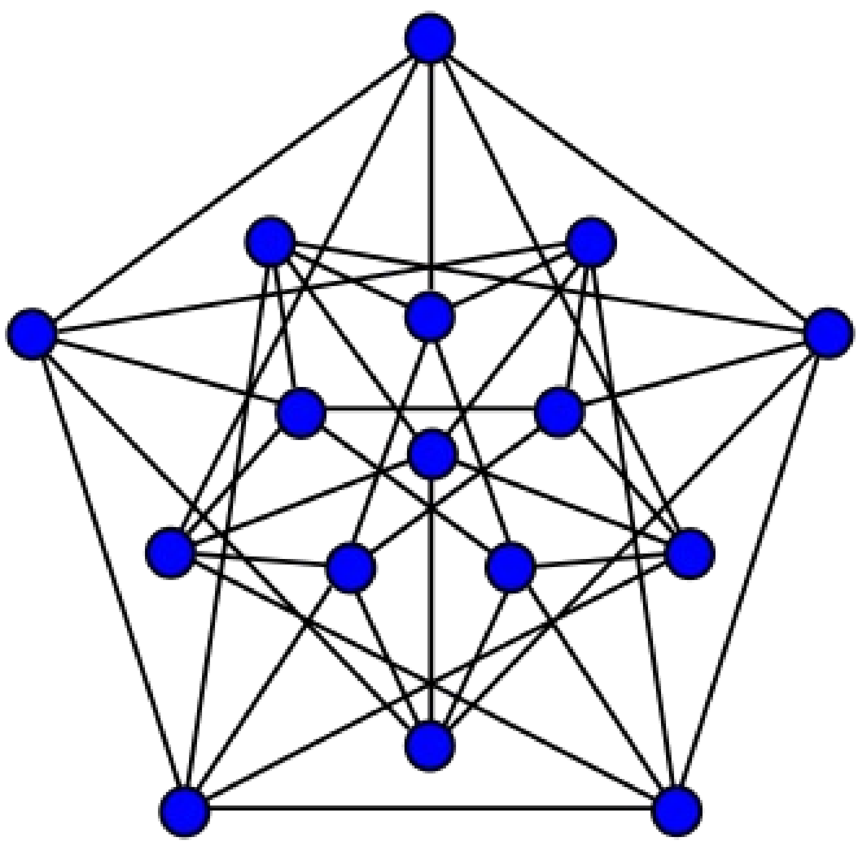
Conversely, it is possible to 2-colour a K_5 without creating any monochromatic K_3, showing that $R(3, 3) > 5$. The unique[2] colouring is shown to the right. Thus $R(3, 3) = 6$.

The task of proving that $R(3, 3) \leq 6$ was one of the problems of William Lowell Putnam Mathematical Competition in 1953, as well as in the Hungarian Math Olympiad in 1947.



The only two 3-colourings of K_{16} with no monochromatic K_3. The untwisted colouring (above) and the twisted colouring (below).





Clebsch graph

Ramsey-numbers :

The numbers $R(r, s)$ in Ramsey's theorem (and their extensions to more than two colours) are known as Ramsey numbers. The Ramsey number, $R(m, n)$, gives the solution to the party problem, which asks the minimum number of guests, $R(m, n)$, that must be invited so that at least m will know each other or at least n will not know each other. In the language of graph theory, the Ramsey number is the minimum number of vertices, $v = R(m, n)$, such that all undirected simple graphs of order v, contain a clique of order m, or an independent set of order n. Ramsey's theorem states that such a number exists for all m and n.

By symmetry, it is true that $R(m, n) = R(n, m)$. An upper bound for $R(r, s)$ can be extracted from the proof of the theorem, and other arguments give lower bounds. (The first exponential lower bound was obtained by Paul Erdős using the probabilistic method.) However, there is a vast gap between the tightest lower bounds and the tightest upper bounds. There are also very few numbers r and s for which we know the exact value of $R(r, s)$.

Computing a lower bound L for $R(r, s)$ usually requires exhibiting a blue/red colouring of the graph K_{L-1} with no blue K_r subgraph and no red K_s subgraph. Such a counterexample is called a *Ramsey graph*. Brendan McKay maintains a list of known Ramsey graphs.[6] Upper bounds are often considerably more difficult to establish: one either has to check all possible colourings to confirm the absence of a counterexample, or to present a mathematical argument for its absence.

Computational complexity

> "Erdős asks us to imagine an alien force, vastly more powerful than us, landing on Earth and demanding the value of $R(5, 5)$ or they will destroy our planet. In that case, he claims, we should marshal all our computers and all our mathematicians and attempt to find the value. But suppose, instead, that they ask for $R(6, 6)$. In that case, he believes, we should attempt to destroy the aliens."
>
> —*Joel Spencer*[7]

A sophisticated computer program does not need to look at all colourings individually in order to eliminate all of them; nevertheless it is a very difficult computational task that existing software can only manage on small sizes. Each complete graph K_n has $1/2n(n - 1)$ edges, so there would be a total of $c^{n(n - 1)/2}$ graphs to search through (for c colours) if brute force is used.[8] Therefore, the complexity for searching all possible graphs (via brute force) is $O(c^{n^2})$ for c colourings and at most n nodes.

The situation is unlikely to improve with the advent of quantum computers. The best known algorithm[*citation needed*] exhibits only a quadratic speedup (c.f. Grover's algorithm) relative to classical computers, so that the computation time is still exponential in the number of nodes.[9]

.

Known values

R(3, 3) = 6 as previously mentioned. Ramsey for 4,2 and Ramsey for any s,2 is always equal to s for any s. It is extremely simple to prove: As a counterexample, let us take a graph with s1 nodes and the edges when all coloured red proves that Ramsey of s,2 is always equal to s. A s-node red subgraph exists in the colouring of all edges coloured red, and a 2-node blue subgraph exists in all other colourings of a graph on s nodes (that is, a pair of nodes connected with a blue edge.)

The values of Ramsey of 4,3 can be calculated using values of Ramsey for 4,2 and 3,3. Similarly, it is easy to deduce the values of Ramsey of 4,4 for 2-coloruing. Various colourings of graph were constructed to find lower bounds of these numbers, and finally Ramsey of 4,4 has been deduced to be 18.

The exact values of Ramsey of 5,5 is unknown, but it known to lie between 43 and 48. The exact value of Ramsey of 4,5 is however known to be exactly 25.

McKay, Radziszowski, and Exoo proposed R(5, 5) = 43 in 1997 using computer-assisted graph generation methods. They could construct exactly 656 graphs by taking various paths to the same set of graphs. There is no way to make a (5, 5, 43) graph out of any of the 656 graphs.

Only weak bounds are available for Ramsey of r,s with r and s greater than 5. Since 1965 and 1972, the lower limits for R(6, 6) and R(8, 8) have remained unchanged. [three]

In the table below, R(r, s) with r, s 10 is shown. The table lists the best known bounds when the exact value is unknown. Ramsey of 1,s being equal to 1 and Ramsey of 2,s is equal to s for all values of s.

The Dynamic Survey 1 of the Electronic Journal of Combinatorics, edited by Stanisaw Radziszowski and updated on a regular basis, is the standard survey on the growth of Ramsey number science. [three] The entries that are there in the table below are from the the edition of 2017 March unless otherwise noted. (Note that R(r, s) = R(s, r) has a trivial symmetry around the diagonal.)

Values / known bounding ranges for Ramsey numbers $R(r, s)$ (sequence A212954 in the OEIS)

r \ s	1	2	3	4	5	6	7	8	9	10
1	**1**	1	1	1	1	1	1	1	1	1
2		**2**	3	4	5	6	7	8	9	10
3			**6**	9	14	18	23	28	36	40–42
4				**18**	25[10]	36–41	49–61	59[14]–84	73–115	92–149
5					**43-48**	58–87	80–143	101–216	133–316	149[14]–442
6						**102-165**	115[14]–298	134[14]–495	183–780	204–1171
7							**205-540**	217–1031	252–1713	292–2826
8								**282-1870**	329–3583	343–6090
9									**565-6588**	581–12677
10										**798-23556**

Social Media Analytics and Ramsey Theorem

The requirement for this research project is to build two-coloured graphs using social media networks . Each node in the graph will represent a person/user of the social media, connected to another node using either of the two colours, e.g. blue and red, one colour representing both of them are connected to each other using the social media, and the other colour representing the two of them not being connected to each other via the social media. Our aim is to find, through statistical experiments, the minimum and maximum number of nodes needed, i.e, the interval with highest probability to find monochromatic cliques each of edges 6,7,8 etc respectively.

Motivation and Problem Statement

In our research analysis project, we will survey on social media graphs taken from Facebook users of MIT and see how much the results are close to the Ramsey Theorem in practice.

That is, Ramsey’s theorem gives a minimum number of edges to guarantee the existence of a monochromatic clique of either red color or blue color.

In our experiment, we have designed red colored edge as knowing each other in Facebook, and blue colored edge as not-knowing each other in Facebook, where the complete graph is formed by vertices where each vertex is an user of Facebook.

We will see by this research survey how many people actually form a blue clique, that is, know each other, versus how many people actually form a red-clique, that is, do not know each other.

In this way, we can have a practical understanding on how many people usually know each other versus how many people usually do-not know each other in a random party thrown at MIT consisting of N number of people (in our experiment, N will be 6 and 18, as we will see). Thus, this is a research analysis of the famous Party Problem as in practice with real-world social media analytics.

Innovation in our project:

1) Use of Social Media Analytics to determine Combinatorial Graph Theoretic results

2) Use of the software Mathematica to build large graphs not explored much before

3) Surveying the Party Problem in practice with real-world social media datas

4) Can open up a broad spectrum of research in the future

CHAPTER 2

LITERATURE SURVEY

2.1 BACKGROUND STUDY:

Radziszowski's Dynamic Survey in the Electronic Journal of Combinatorics was recommended for a survey of the most recent findings on Ramsey graphs. The combinatorial data page of Ramsey Graphs was widely used for details on the data format used.

2.2 SUMMARY OF LITERATURE REVIEW:

[1] "Small Ramsey Numbers
Stanisław Radziszowski
DS1: Jan 15, 2021
2011-08-22 "

In this paper, the author presents data which, to the best of our knowledge, includes all known nontrivial values and bounds for specific graph, hypergraph and multicolor Ramsey numbers, where the avoided graphs are complete or complete without one edge. Many results pertaining to other more studied cases are also presented. They give references to all cited bounds and values, as well as to previous similar compilations. They do not attempt complete coverage of asymptotic behavior of Ramsey numbers, but concentrate on their specific values.

[2] "Ramsey Theory Applications
Vera Rosta
DS13: Dec 7, 2004
2004-12-07 "

There are many interesting applications of Ramsey theory, these include results in number theory, algebra, geometry, topology, set theory, logic, ergodic theory, information theory and theoretical computer science. Relations of Ramsey-type theorems to various fields in mathematics are well documented in published books and monographs. The main objective of this survey is to list applications mostly in theoretical computer science of the last two decades not contained in these.

[3] "Ramsey Numbers of Ordered Graphs

Martin Balko, Josef Cibulka, Karel Král, Jan Kynčl
P1.16
2020-01-10 "

For a few special classes of ordered paths, stars or matchings, they give asymptotically tight bounds on their ordered Ramsey numbers. For so-called monotone cycles they compute their ordered Ramsey numbers exactly. This result implies exact formulas for geometric Ramsey numbers of cycles introduced by Károlyi, Pach, Tóth, and Valtr.

[4] "Constructive Lower Bounds on Classical Multicolor Ramsey Numbers

Xu Xiaodong, Xie Zheng, Geoffrey Exoo, Stanisław P. Radziszowski
R35
2004-06-04 "

This paper looks at lower bounds for classical multicolor Ramsey numbers, first providing a brief summary of previous work and then introducing several general constructions that define new lower bounds for a variety of diagonal and off-diagonal multicolor Ramsey numbers.

The majority of the new lower limits are the product of broad constructions."

[5] "The Ramsey Number of Loose Paths in 3-Uniform Hypergraphs

Leila Maherani, Gholam Reza Omidi, Ghaffar Raeisi, Maryam Shahsiah
P12
2013-01-21 "

Recently, asymptotic values of 2-color Ramsey numbers for loose cycles and also loose paths were determined. Here they determine the 2-color Ramsey number of 3-uniform loose paths when one of the paths is significantly larger than the other.

CHAPTER 3

SYSTEM ANALYSIS

A social friendship network extracted from Facebook consisting of people (nodes) with edges representing friendship ties was used from MIT was used in our research analysis.

Network Data Statistics	
Nodes	6.4K
Edges	251.2K
Density	0.0122613
Maximum degree	708
Minimum degree	1
Average degree	78
Assortativity	0.119896
Number of triangles	7.1M
Average number of triangles	1.1K
Maximum number of triangles	27.8K
Average clustering coefficient	0.27236
Fraction of closed triangles	0.180288
Maximum k-core	73
Lower bound of Maximum Clique	9

Fig 3.1: Network Data Statistics

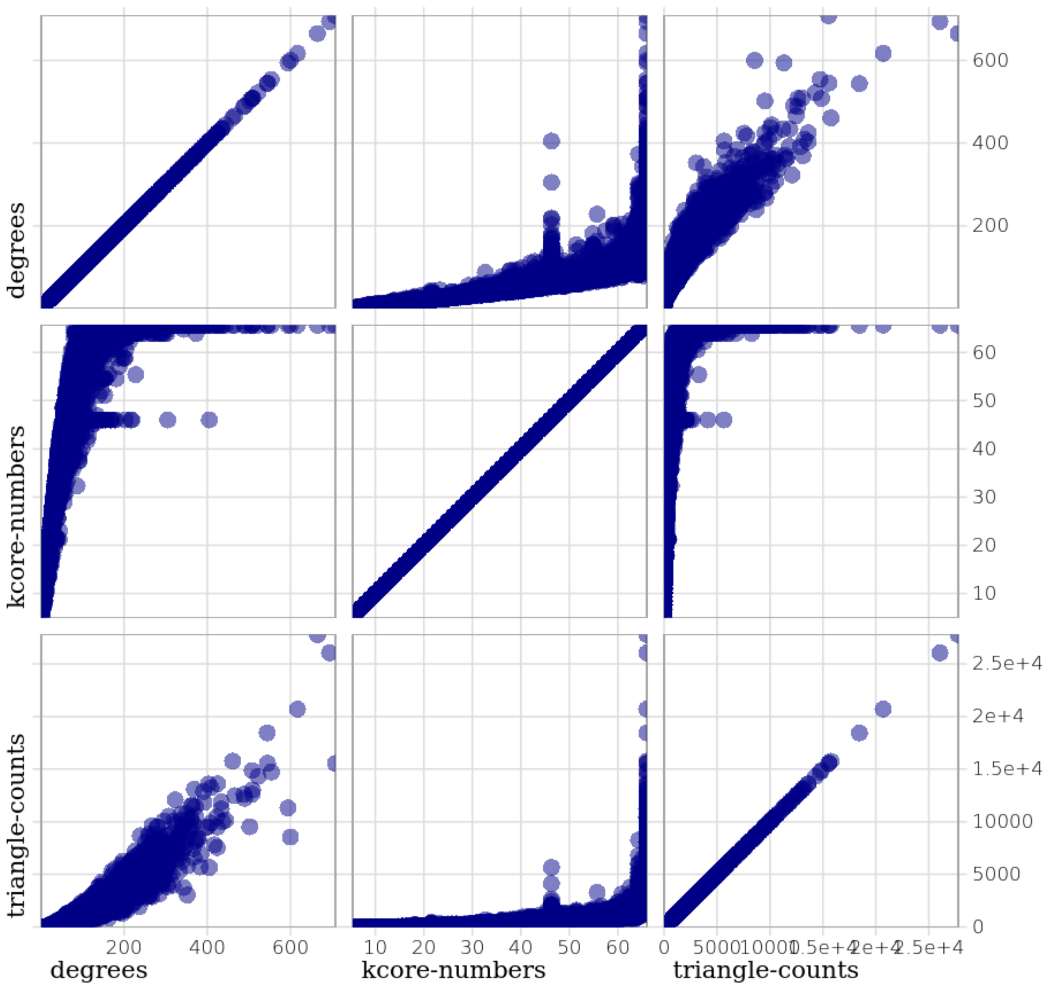
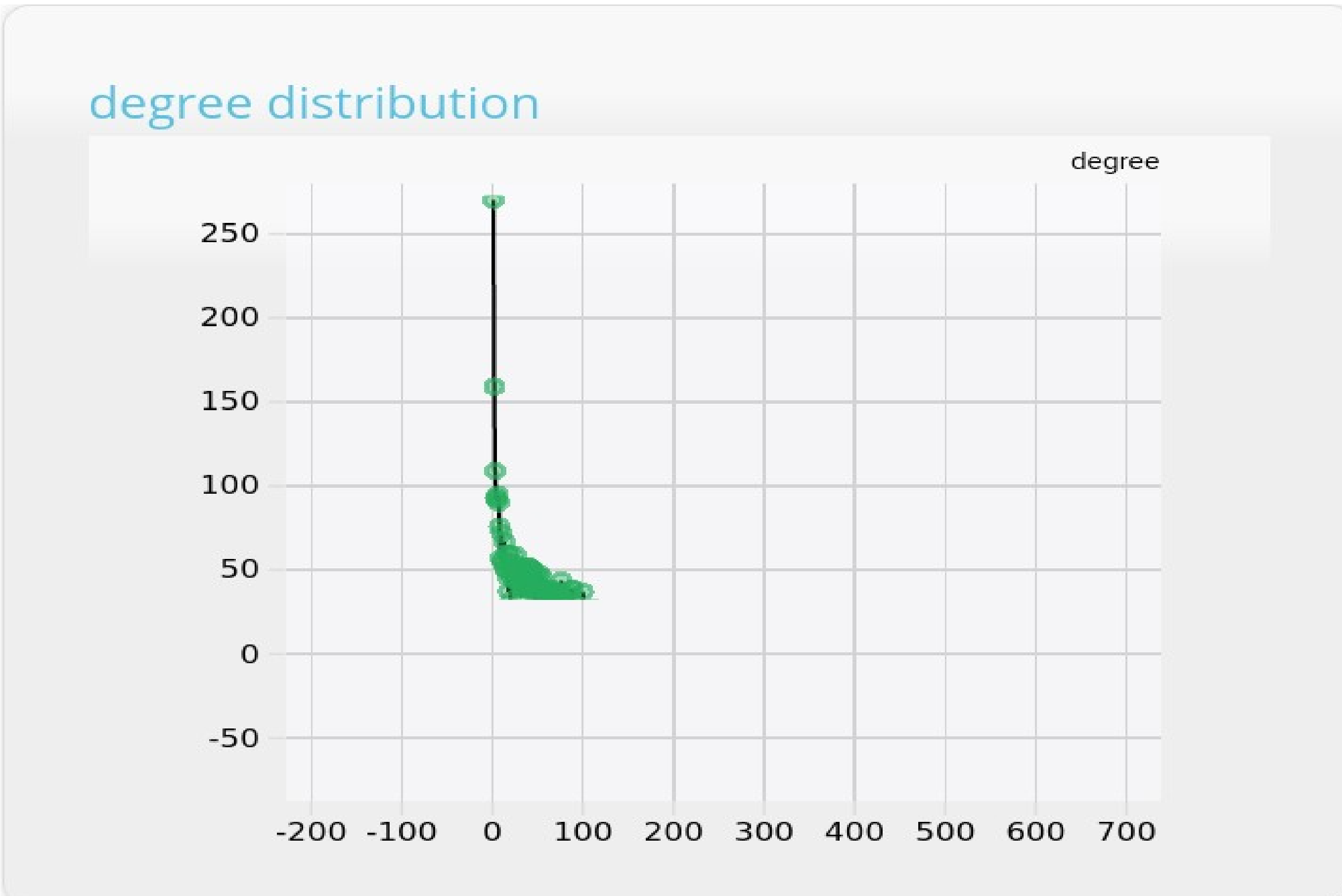
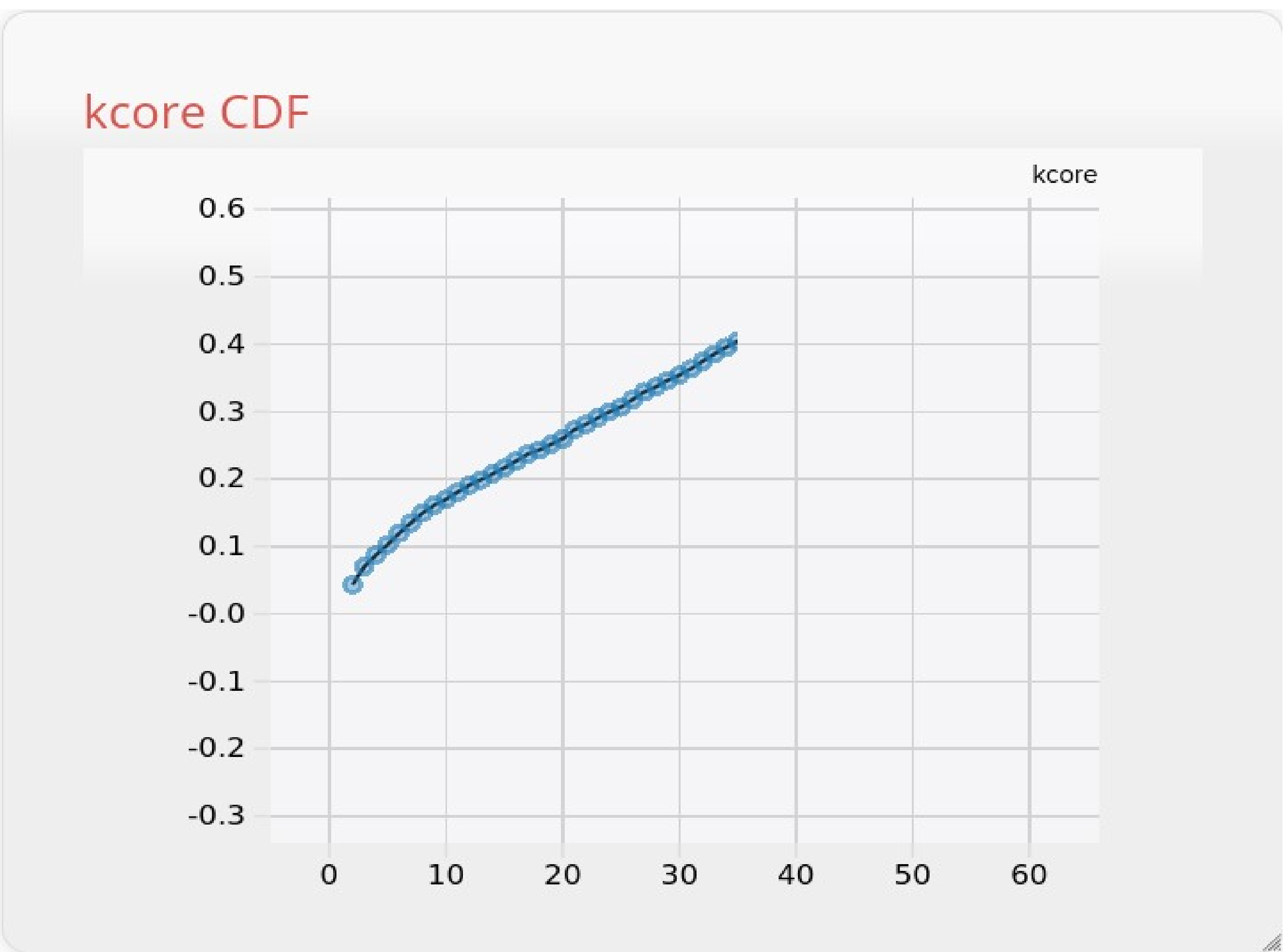


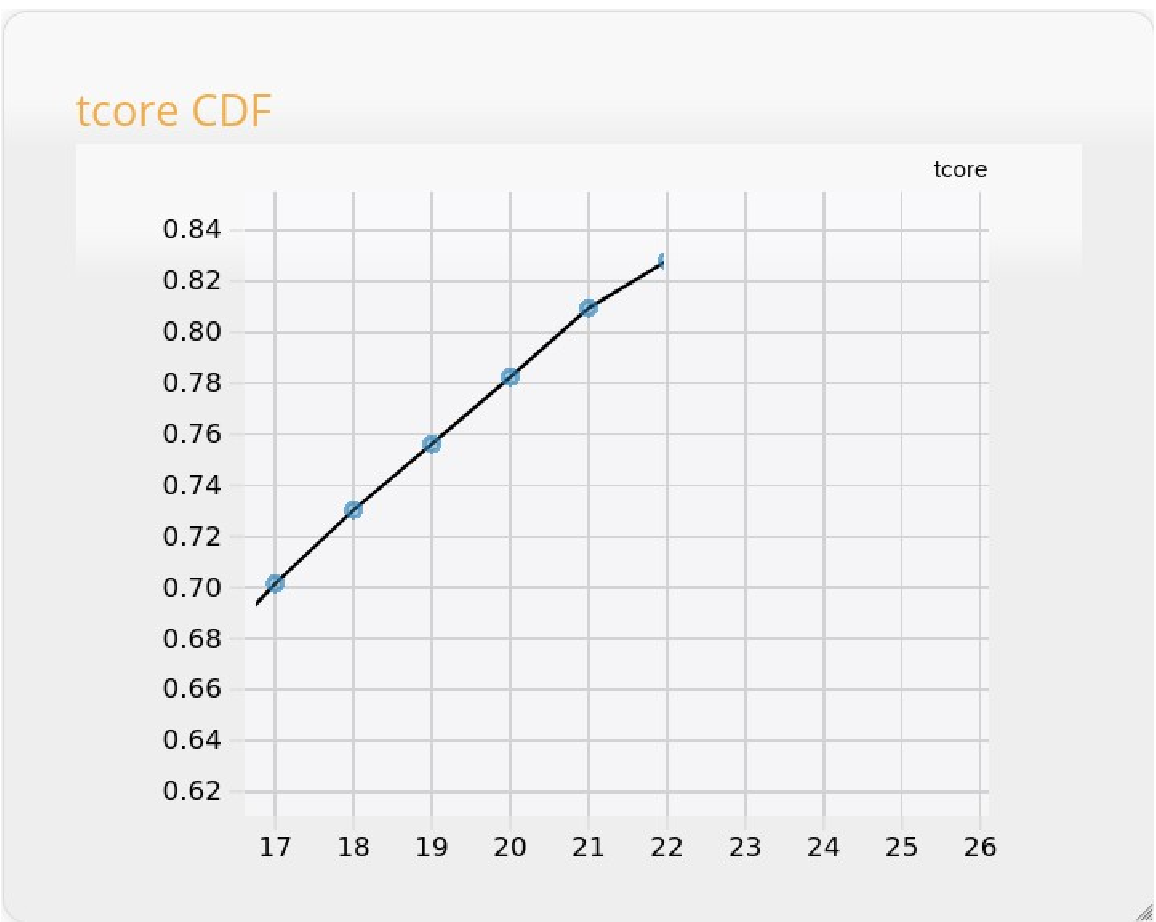
Fig 3.2: Node-level Statistics

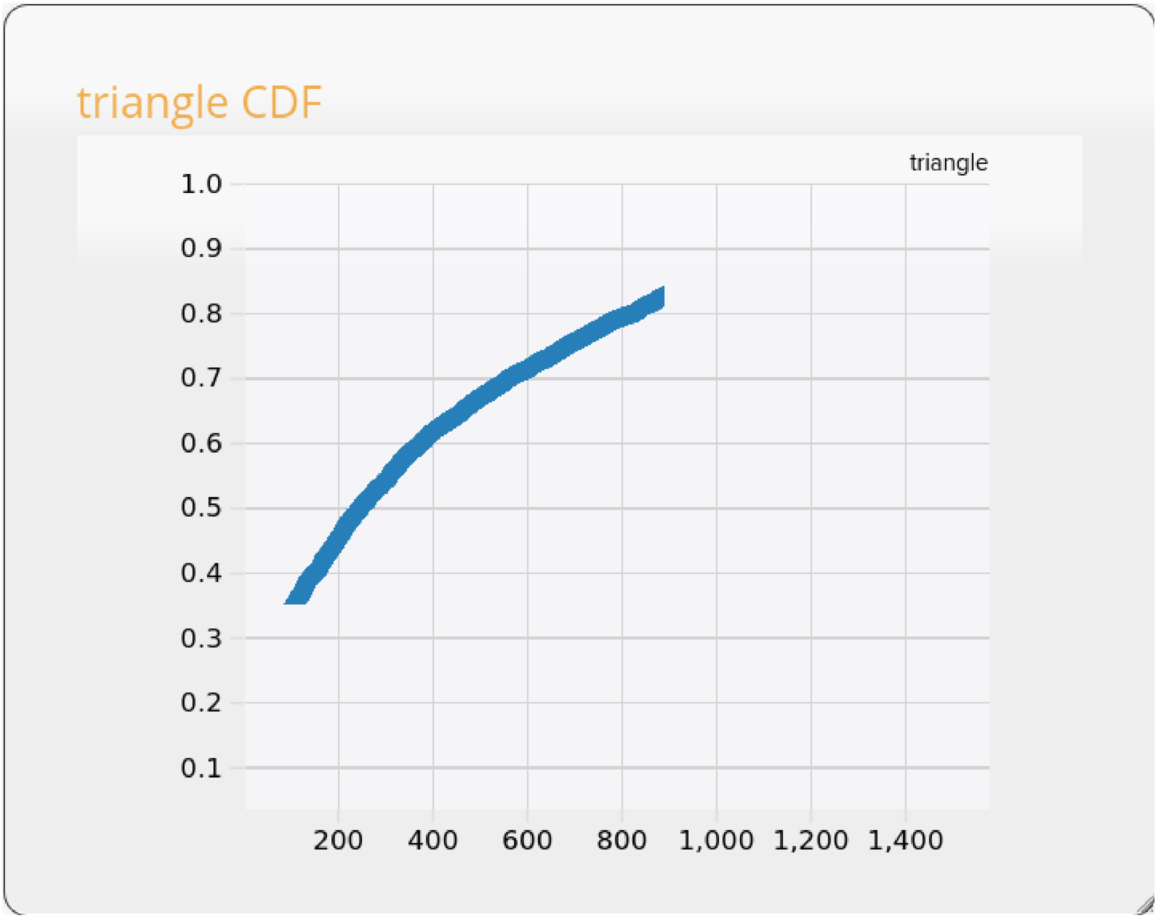
Node-level Feature Distributions

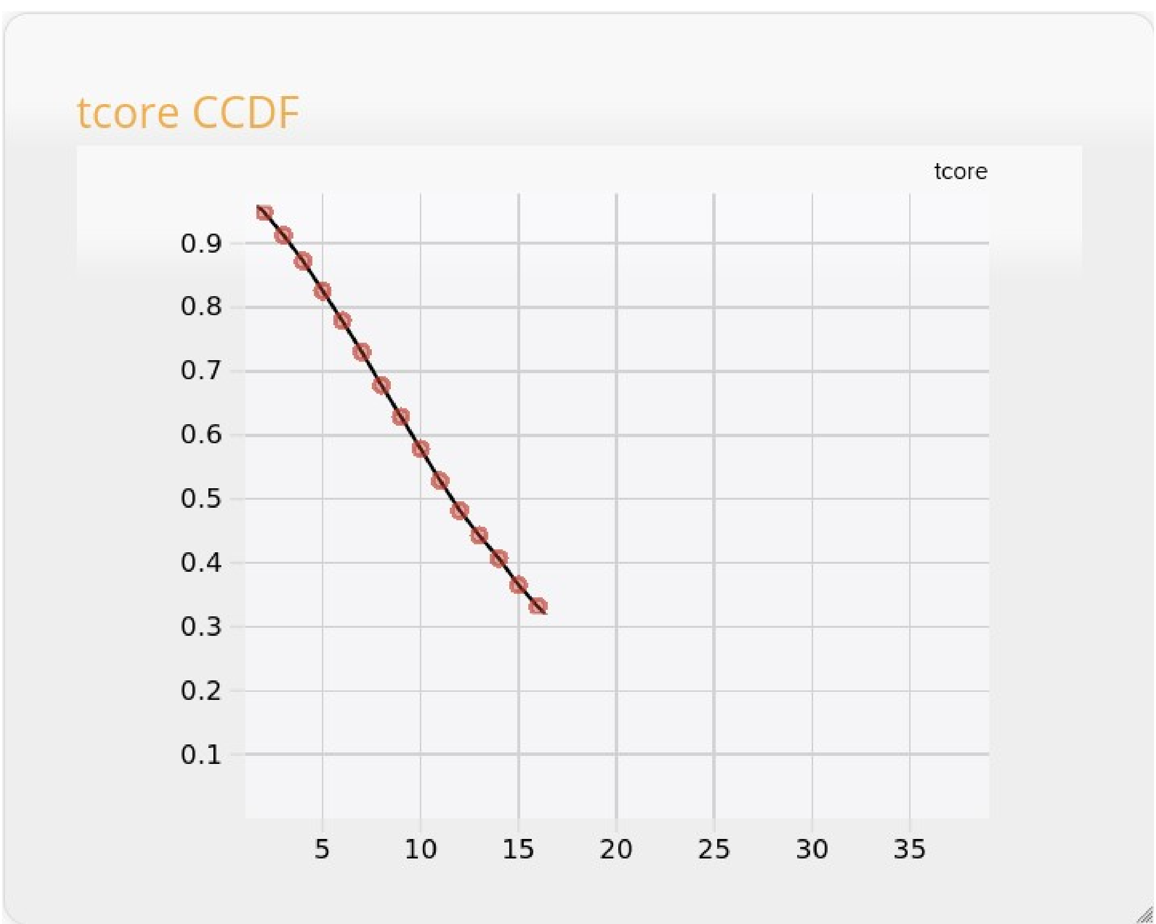


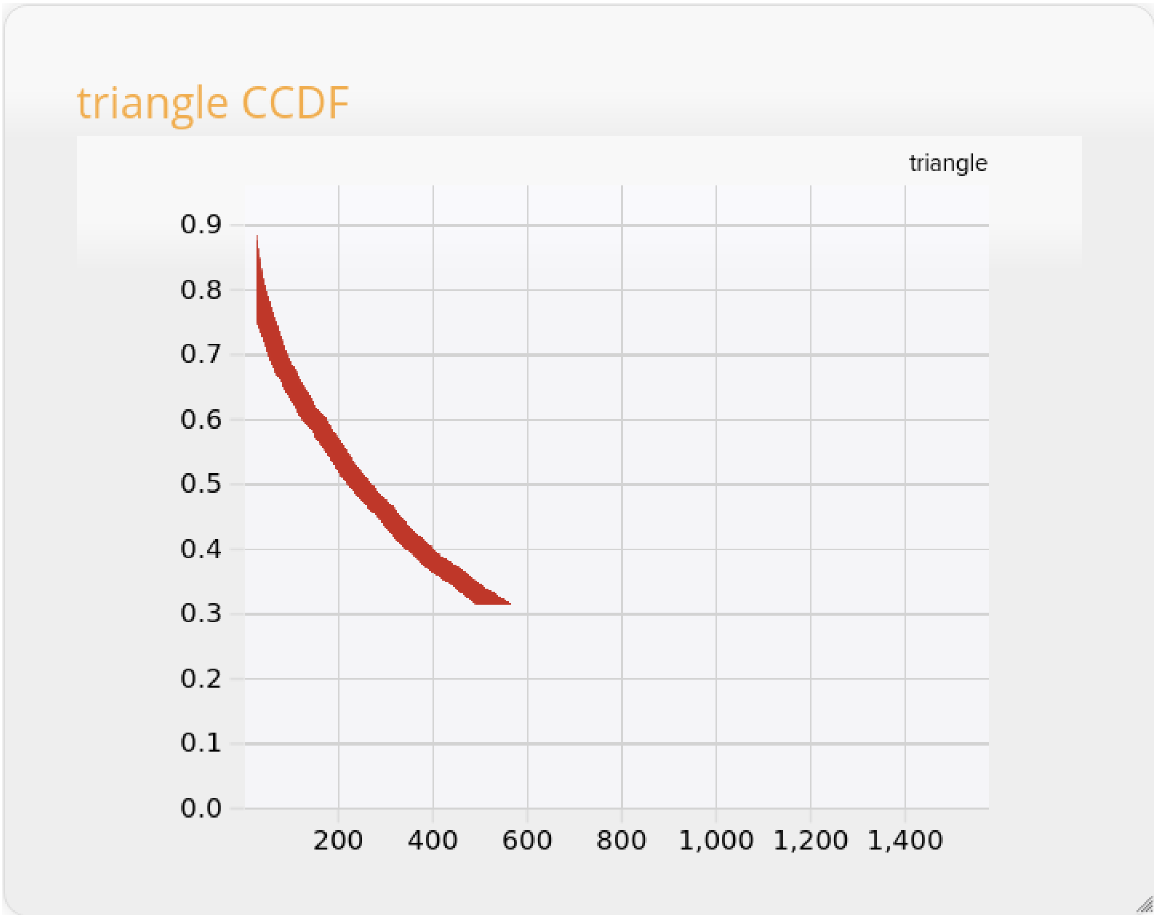

kcore CDF
kcore
0.6
0.5
0.4
0.3
0.2
0.1
-0.0
-0.1
-0.2
-0.3
0
10
20
30
40
50
60


tcore distribution
tcore
12,000
11,000
10,000
9,000
8,000
7,000
6,000
5,000
4,000
3,000
2,000
1,000
5
10
15
20
25
30
35


tcore CDF
tcore
0.84
0.82
0.80
0.78
0.76
0.74
0.72
0.70
0.68
0.66
0.64
0.62
17
18
19
20
21
22
23
24
25
26


triangle CDF
triangle
1.0
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
200
400
600
800
1,000
1,200
1,400


tcore CCDF
tcore
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
5
10
15
20
25
30
35


triangle CCDF
triangle
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0.0
200
400
600
800
1,000
1,200
1,400

CHAPTER 4

SYSTEM DESIGN & ARCHITECTURE

Metadata

Category	Sparse networks
Collection	Facebook networks
Tags	Facebook, social network, friendship graph, friendship relations, friend ties, Facebook networks, Facebook social networks, university friendship network
Short	Facebook social network
Vertex type	Person
Edge type	Friendship, social relationship
Format	Undirected
Edge weights	Unweighted
Description	A social friendship network extracted from Facebook consisting of people (nodes) with edges representing friendship ties.

Fig 4.1 Metadata of the social network graph used from MIT

4.1 WORKING PRINCIPLE

A complete graph consisting of undirected edges were created using MIT Facebook Analytics.

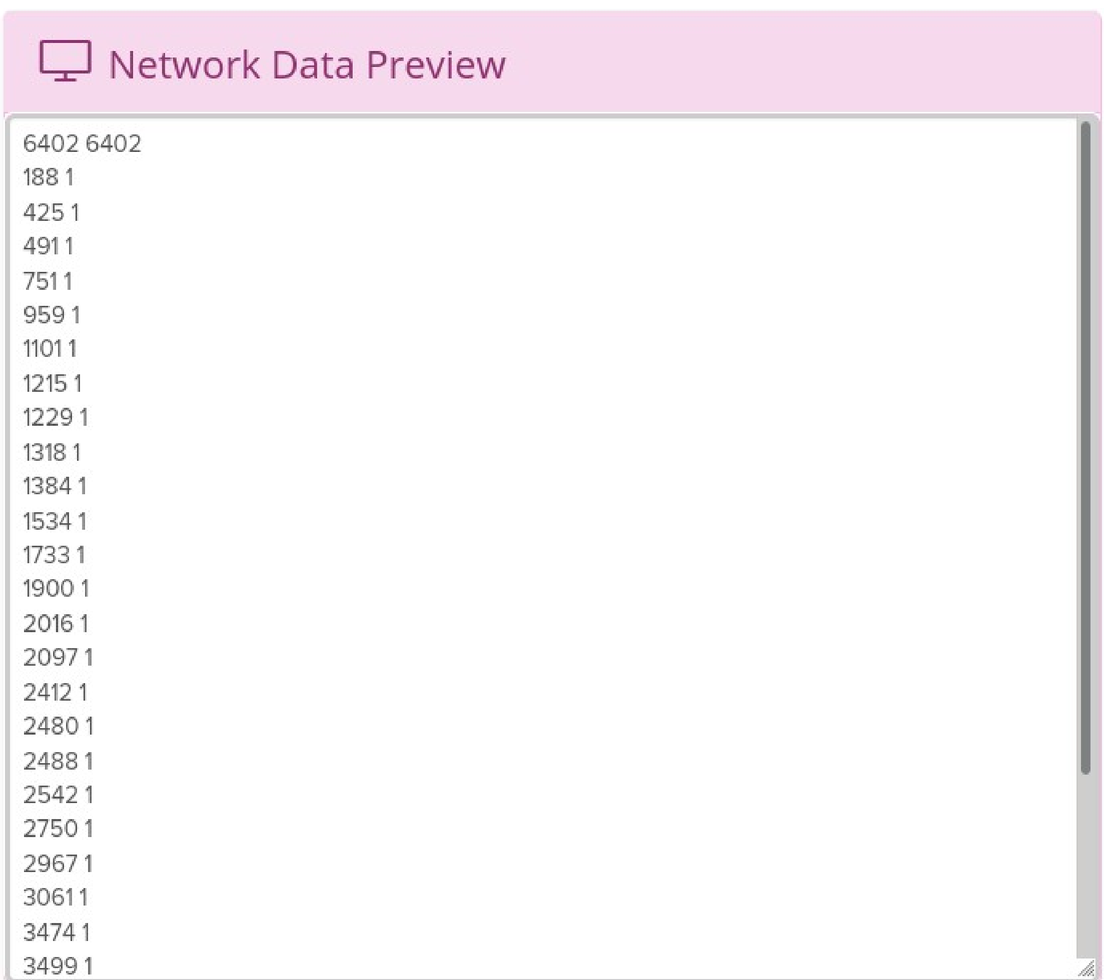
As stated above, the data had 6.4K nodes, where a connected to b meant that node a was connected to node b in Facebook. These edges were coloured red. The nodes that did not know each other were also joined using blue edges to make it a complete graph.

The above manipulation was done using the software Mathematica licensed under Wolfram Research, Inc, a very good tool to do these mathematical manipulations of graphs, etc.

4.2 COMPONENTS OF PROJECT

The Graph Edges

Importing the data in Mathematica and converting it to expression from strings:



```
In[∘]:= g =
  Import[
   "/home/justanotherlad/Documents/BTech Degree
     Project/MIT/socfb-MIT.txt"];

In[∘]:= g1 = ToExpression[StringSplit[g]]

Out[∘]= {188, 1, 425, 1, 491, 1, 751, 1, 959, 1, 1101, 1, 1215,
  1, 1229, 1, 1318, 1, 1384, 1, 1534, 1, 1733, 1, 1900,
  … 502 410 …, 6355, 6375, 6361, 6367, 6363, 6368, 6364,
  6391, 6364, 6377, 6368, 6391, 6368, 6375, 6369, 6377,
  6375, 6380, 6375, 6381, 6375, 6392, 6376, 6395, 6378}

  large output | show less | show more | show all | set size limit...
```

Forming a complete graph with every vertex connected to each other, the number of vertices being same as that in the imported data:

```
In[∘]:= g2 = DeleteCases[
    Flatten[Table[UndirectedEdge[i, j] && i ≠ j, {i, 0, Max[g1]},
      {j, i + 1, Max[g1]}]], False];
```

Edge manipulation:

```
In[]:= g3 = DeleteDuplicates[
         ReplacePart[g1,
          Table[{{i}, {i + 1}} → UndirectedEdge[g1[[i + 1]], g1[[i]]],
           {i, 1, Length@g1, 2}]]]
```

```
Out[]= {1 ⟷ 188, 1 ⟷ 425, 1 ⟷ 491, 1 ⟷ 751,
        1 ⟷ 959, 1 ⟷ 1101, 1 ⟷ 1215, 1 ⟷ 1229, 1 ⟷ 1318,
        1 ⟷ 1384, 1 ⟷ 1534, 1 ⟷ 1733, … 251 206 …,
        6361 ⟷ 6375, 6363 ⟷ 6367, 6364 ⟷ 6368, 6364 ⟷ 6391,
        6368 ⟷ 6377, 6368 ⟷ 6391, 6369 ⟷ 6375, 6375 ⟷ 6377,
        6375 ⟷ 6380, 6375 ⟷ 6381, 6376 ⟷ 6392, 6378 ⟷ 6395}

       large output   show less   show more   show all   set size limit...
```

Manipulating the graph such that the edges representing people connected in Facebook becomes Red, and the edges representing people not connected in Facebook becomes white:

```
In[]:= intersectpos = With[{disp = Dispatch[Thread[g3 → "foo"]]},
          Flatten@Position[Replace[g2, disp, {1}], "foo"]];

In[]:= g2 = ReplacePart[g2,
          Table[intersectpos[[i]] →
            DirectedEdge[g2[[intersectpos[[i]], 1]],
             g2[[intersectpos[[i]], 2]]],
           {i, Length[intersectpos]}]];
```

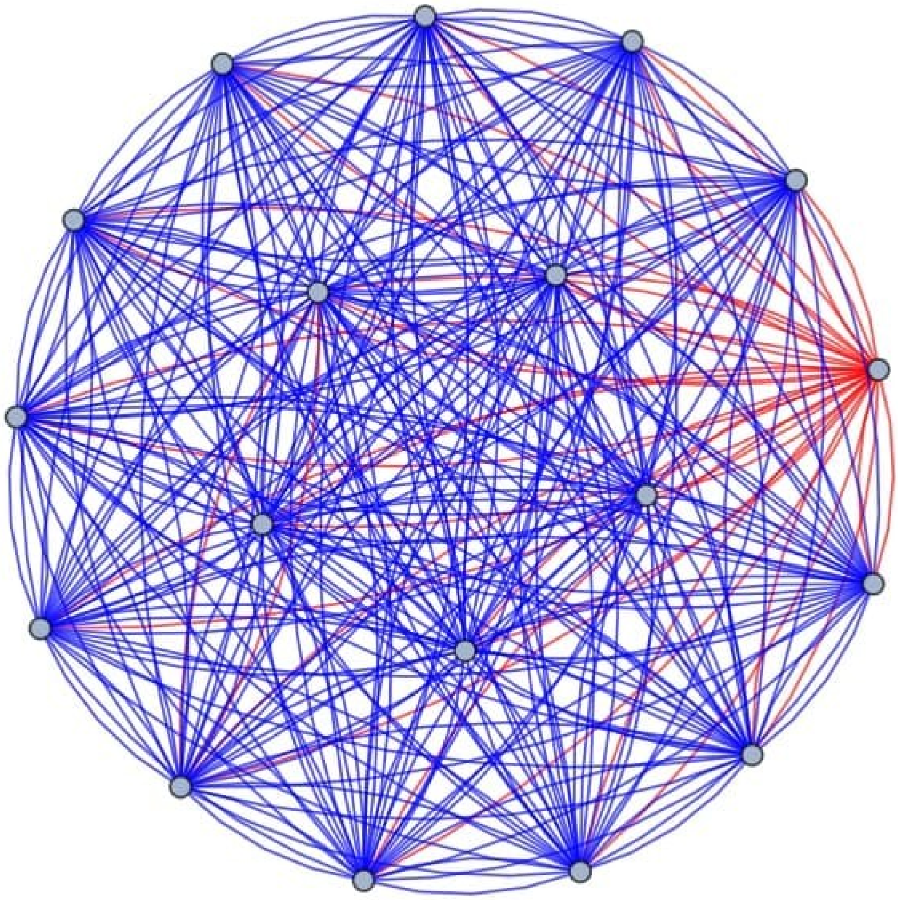


Fig: An example graph where each node represents an user and two nodes connected by blue edge represents they are not connected in Facebook, and two nodes connected via red edge means they are connected in Facebook as per the imported graph data of MIT.

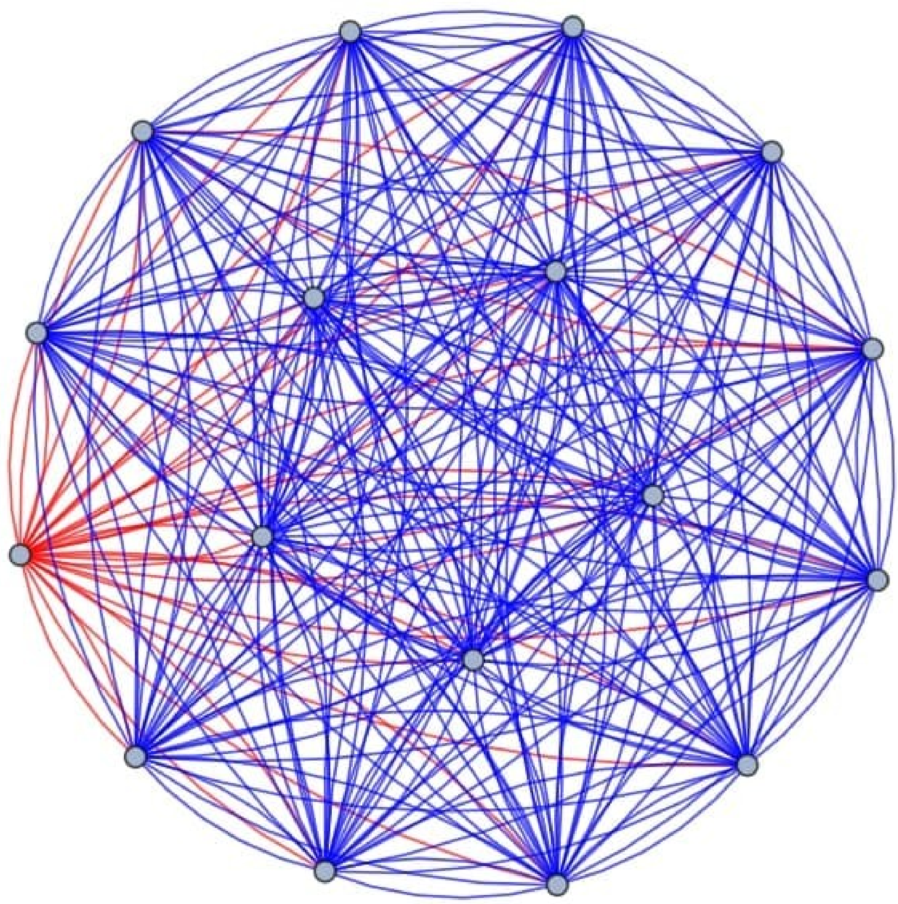


Fig: Another example graph

Final function construction to check monochromatic cliques:

```
MonochromaticCliqueTester[n_, e_, x_] :=
 Module[{subedges, newEdges, graph, i, j, var,
   coloredSubgraphs, cliques, list = {{}, {}}},


  For[i = 0, i < Floor@N@(Max[g1] / 1000), i++,
   subedges =
    Select[g2,
     (1000 * i < #[[1]] < 1000 * (i + 1)) &&
       (1000 * i < #[[2]] < 1000 * (i + 1)) &];
   newEdges = Replace[subedges,
     {DirectedEdge[a_, b_] :>
       Style[UndirectedEdge[a, b], Red, Thick],
      UndirectedEdge[a_, b_] :>
       Style[UndirectedEdge[a, b], Blue]}, {1}];
   graph = Graph[newEdges];

   For[j = 1, j <= x, j++,

    var = Subgraph[graph,
      {RandomInteger[{1000 * i, 1000 * (i + 1)}, e]},
      VertexLabels → Automatic];

    Export[
     "/home/justanotherlad/Documents/BTech Degree
        Project/MIT/Pictures/5-Clique/Ramsey_for_" <>
      ToString[n] <> "_in_" <> ToString[e] <> "_" <>
```

```
graph = Graph[newEdges];

For[j = 1, j <= x, j ++,

 var = Subgraph[graph,
   {RandomInteger[{1000 * i, 1000 * (i + 1)}, e]},
   VertexLabels → Automatic];

 Export[
  "/home/justanotherlad/Documents/BTech Degree
     Project/MIT/Pictures/5-Clique/Ramsey_for_" <>
   ToString[n] <> "_in_" <> ToString[e] <> "_" <>
   ToString[i] <> "_" <> ToString[j] <> ".png", var];



 coloredSubgraphs =
  Quiet[GroupBy[MapAt[Lookup[LineColor],
     AnnotationValue[var, EdgeStyle], {All, 2}],
    Last → First]];
 cliques = Length@First@FindClique[#] ≥ n & /@
   coloredSubgraphs;
 AppendTo[list[[1]], cliques[[1]]];
 If[Length@cliques > 1,
  AppendTo[list[[2]], cliques[[2]]]];
]
];
```

File Edit Insert Format Cell Graphics Evaluation Palettes Window Help

```
Labeled[
 BarChart[
  {{Length@Cases[list[[1]], True],
    Length@Cases[list[[2]], True]}},
  ChartBaseStyle → EdgeForm[Dashed], ChartStyle → {Blue, Red},
  ChartLegends → {"Don't Know Each Other in Facebook",
    "Know Each Other in Facebook"}, BarSpacing → None,
  ChartLabels -> {{"Clique Exists"},
    {"Blue Cliques", "Red Cliques"}},
  LabelingFunction →
   (Placed[Row[{#, "/", N[x * i]}], Above] &),
  PlotLabel →
   Style["Finding " <> ToString[n] <>
     "-Cliques in a Complete " <> ToString[e] <> "-Graph",
    "Title", 16], ImageSize → Medium,
  PerformanceGoal → "Speed"],
 Column[{"Probability that there are " <> ToString[n] <>
    " people knowing each other in a party of " <>
    ToString[e] <> " at MIT is roughly " <>
    ToString[N[Length@Cases[list[[2]], True] / N[x * i]]],
   "Probability that there are " <> ToString[n] <>
    " people not knowing each other in a party of " <>
    ToString[e] <> " at MIT is roughly " <>
    ToString[N[Length@Cases[list[[1]], True] / N[x * i]]]}],
 LabelStyle → Directive[FontSize → 13.5,
   FontFamily → "Times"]]
] (*Gives a bar chart of how many times monochromatic n-
```

100%

File Edit Insert Format Cell Graphics Evaluation Palettes Window Help

```
    LabelingFunction →
     (Placed[Row[{#, "/", N[x * i]}], Above] &),
    PlotLabel →
     Style["Finding " <> ToString[n] <>
       "-Cliques in a Complete " <> ToString[e] <> "-Graph",
      "Title", 16], ImageSize → Medium,
    PerformanceGoal → "Speed"],
   Column[{"Probability that there are " <> ToString[n] <>
      " people knowing each other in a party of " <>
      ToString[e] <> " at MIT is roughly " <>
      ToString[N[Length@Cases[list[[2]], True] / N[x * i]]],
     "Probability that there are " <> ToString[n] <>
      " people not knowing each other in a party of " <>
      ToString[e] <> " at MIT is roughly " <>
      ToString[N[Length@Cases[list[[1]], True] / N[x * i]]]}],
   LabelStyle → Directive[FontSize → 13.5,
     FontFamily → "Times"]]
 ] (*Gives a bar chart of how many times monochromatic n-
  cliques of colors blue and red were found resp. in
   a complete graph of e vertices
   (taken randomly from the entire imported social-
     media graph in batches of successive 1000 vertices
      at a time till all the vertices are taken into
      consideration) when the experiment is run a
   total of x times*)
```

100%

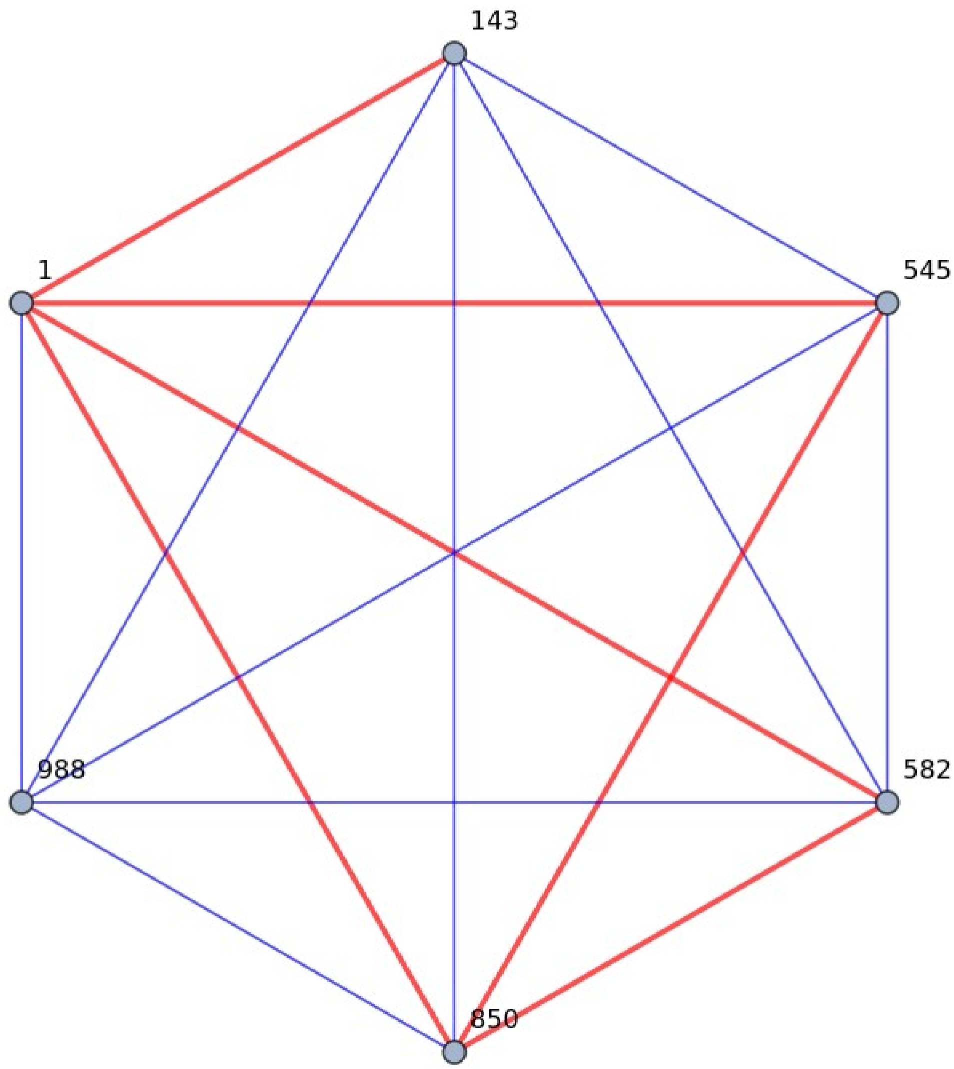


Fig: A final example of the type of clique the MonochromaticCliqueTester[ ] function can find. Here node 1, 545 and 850 form a 3-clique in this complete 6-graph, hence proving Ramsey's theorem that in any random complete graph of 6 vertices, there always exist a 3-clique of either red color or blue colour (in case of 2-coloring).

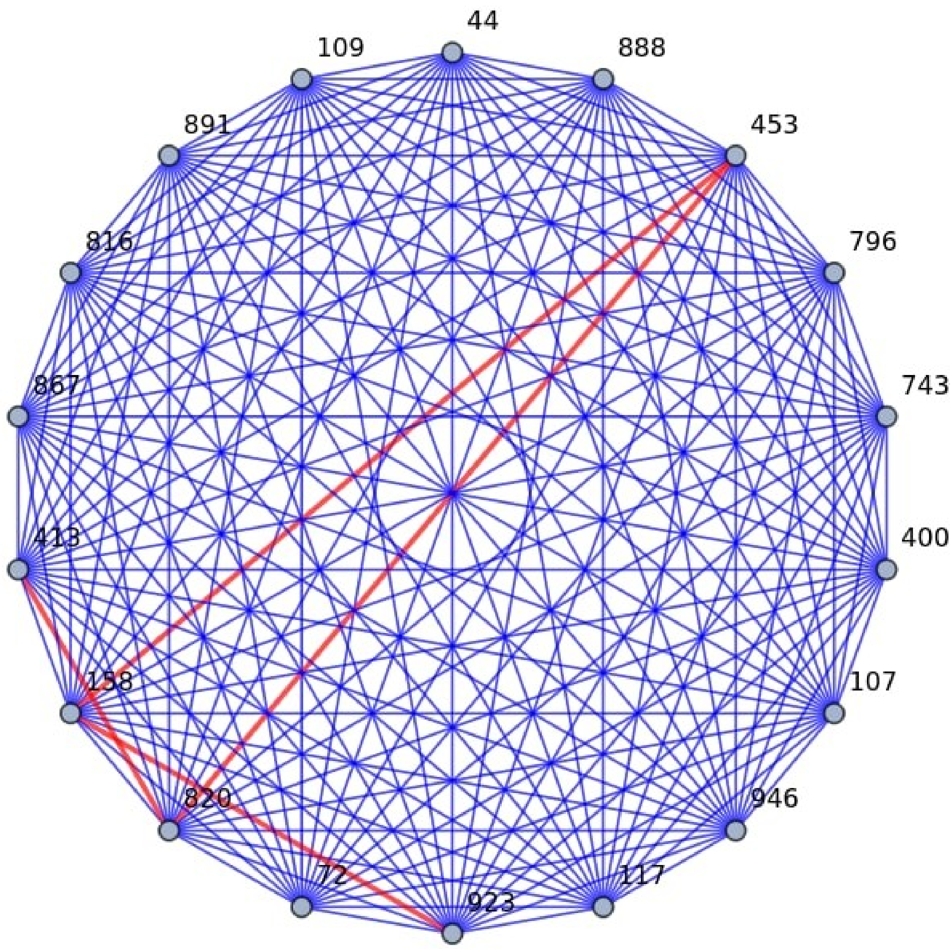


Fig: Another example of attempt to find monochromatic cliques in a larger complete graph of 18 people.

CHAPTER 5

SYSTEM TESTING

1.General

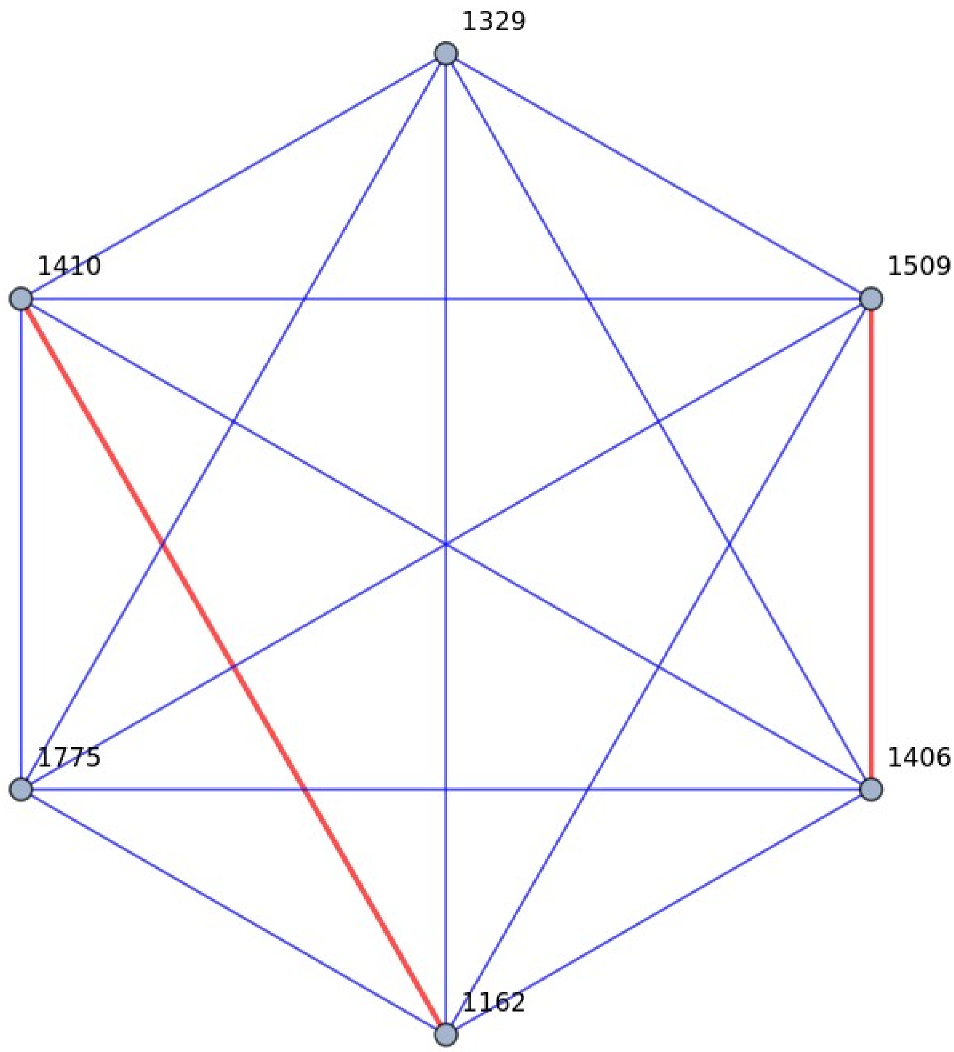
Testing is performed in order to ensure that our project is giving the desired outcome or not, it is also helpful in giving the value of efficiency and accuracy of the project. Testing at multiple stages is performed in order to ensure that our project is giving desired outcomes at each and every level. Even if one module of our project fails then our system as a whole fails and hence, testing at every level is very important because that's the only way of ensuring the working of the project is fine and we can carry on further with our project. Therefore, testing is a non compromisable part of our project. In simple terms, it is performed to do 2 things
i.) to ensure whether we are getting the same result as expected.
ii.) whether or not our system has any bugs.
Hence, we should take testing very seriously.

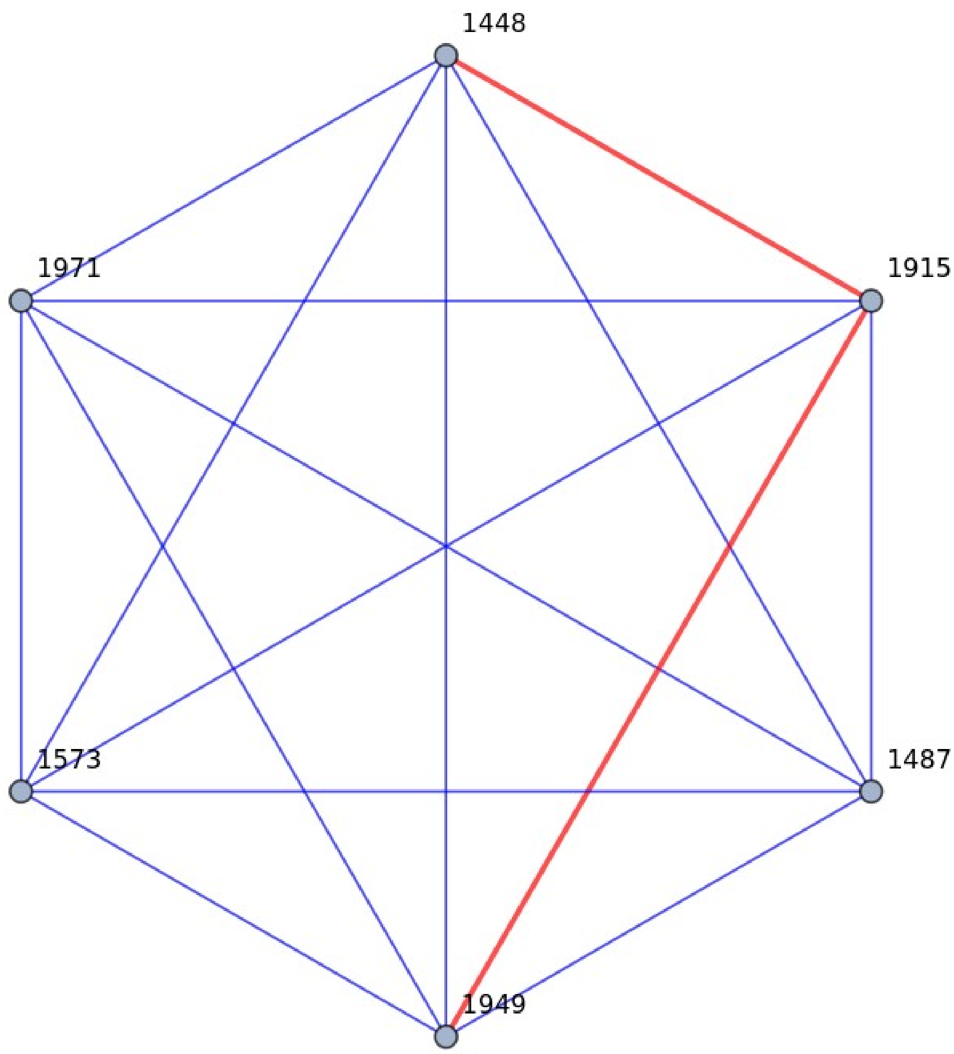
Our experiment was tested on the imported MIT graph network data of 6.4K vertices, where each vertex represented an user in Facebook. An edge connected in Red represented whether they knew each other on Facebook, and an edge in Blue represented that they did not know each other on Facebook.

Our aim was to find, in a random party at MIT consisting of 6 or 18 people, what is the probability that 6 people or 18 people (respectively) will know each other or not know each other, based on this trusted imported graph network.

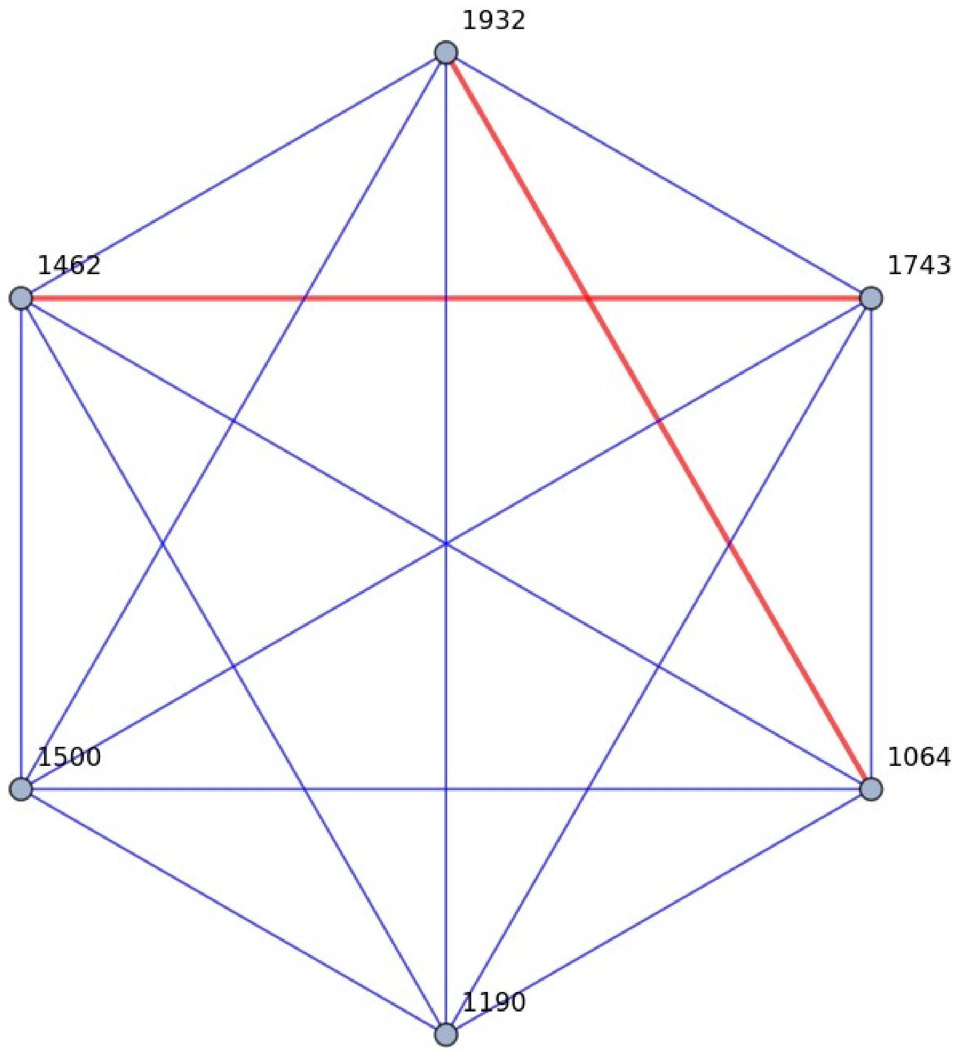
Our experiment was run 100 times each for every successive batch of 1000 vertices, i.e, a total of 600 times each to determine the 3-clique probability for 6-graph and 4-clique probability in 18-graph respectively.
We provide the images of the graphs so formed in our experiment as follows:

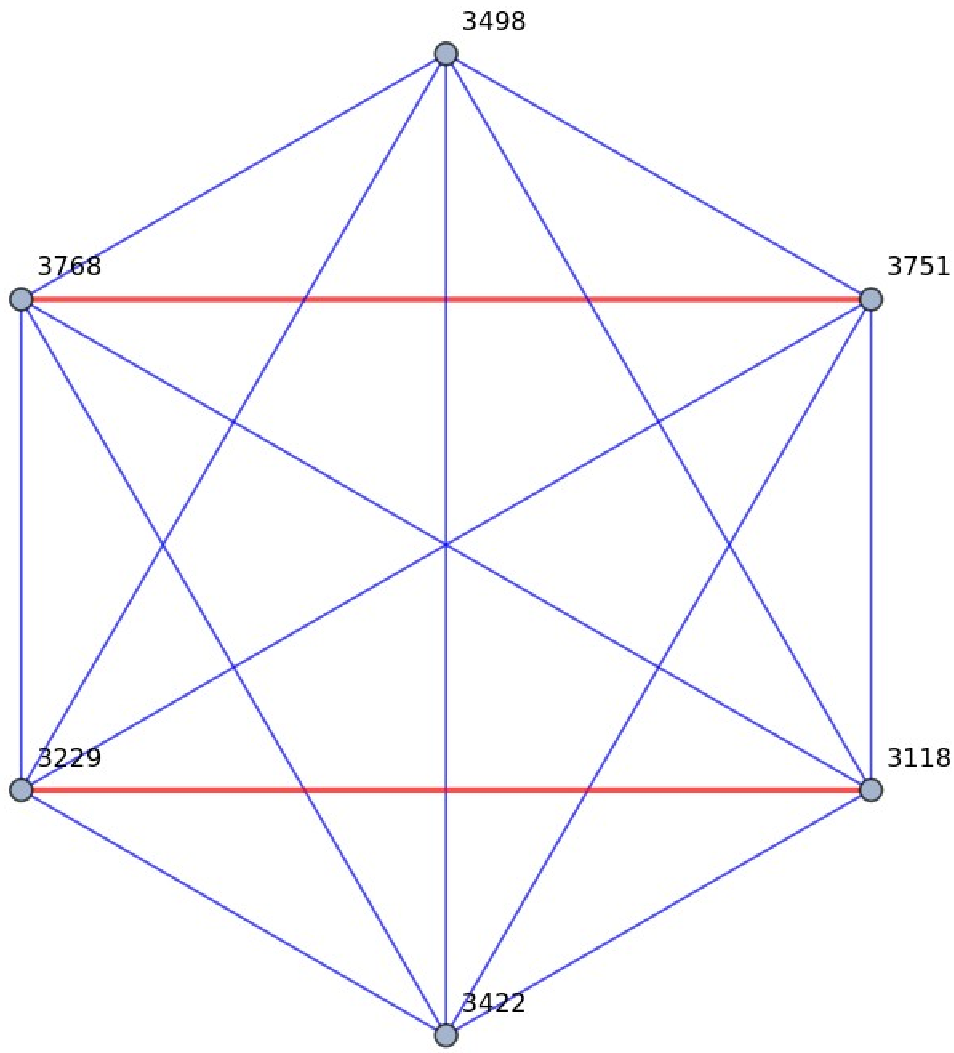
3-Cliques in a complete 6-graph, where Red edges represent the nodes know each other in Facebook, and Blue edges represent they do not know each other.





1448
1971
1915
1573
1487
1949


1932
1462
1743
1500
1064
1190


3498
3768
3751
3229
3118
3422

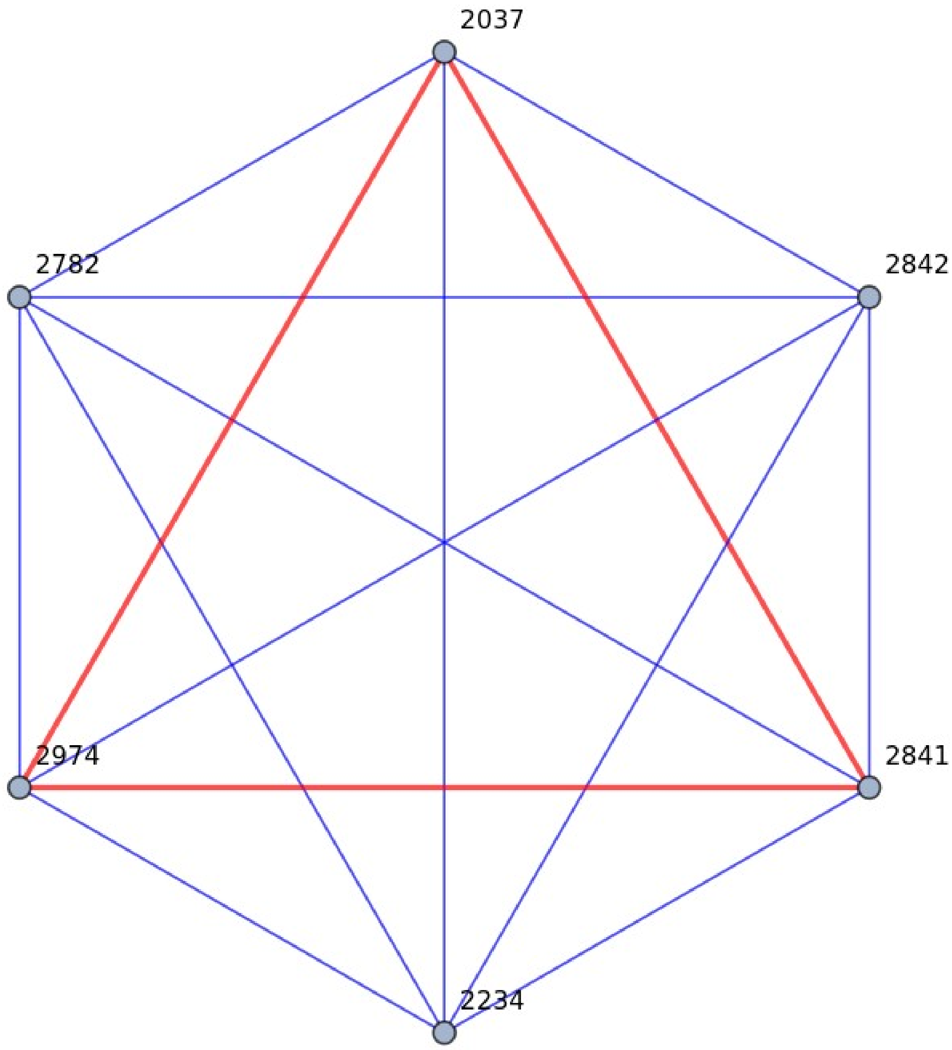


Fig: A Red 3-Clique was found, representing there are 3 people knowing each other, as predicted by Ramsey

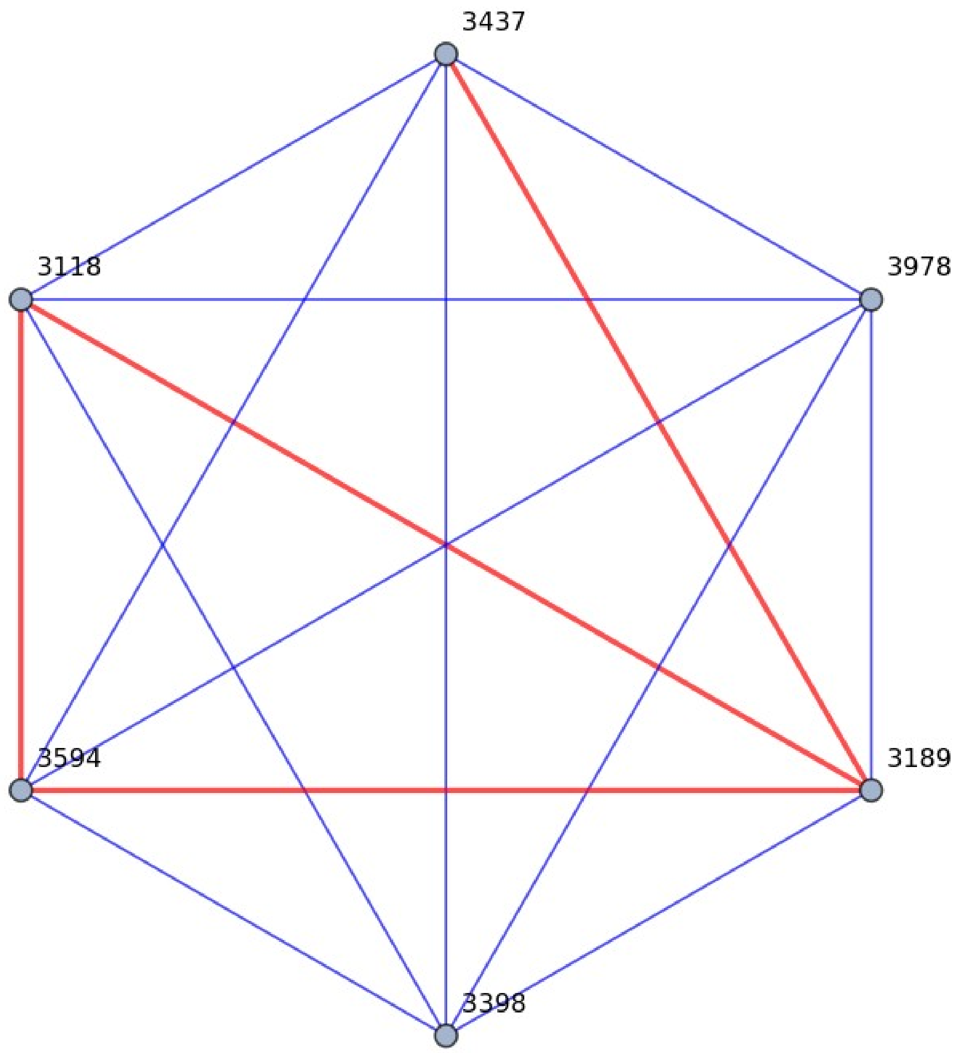


Fig: Another similar graph was found

4-Cliques in a complete 18-graph, where Red edges represent the nodes know each other in Facebook, and Blue edges represent they do not know each other.

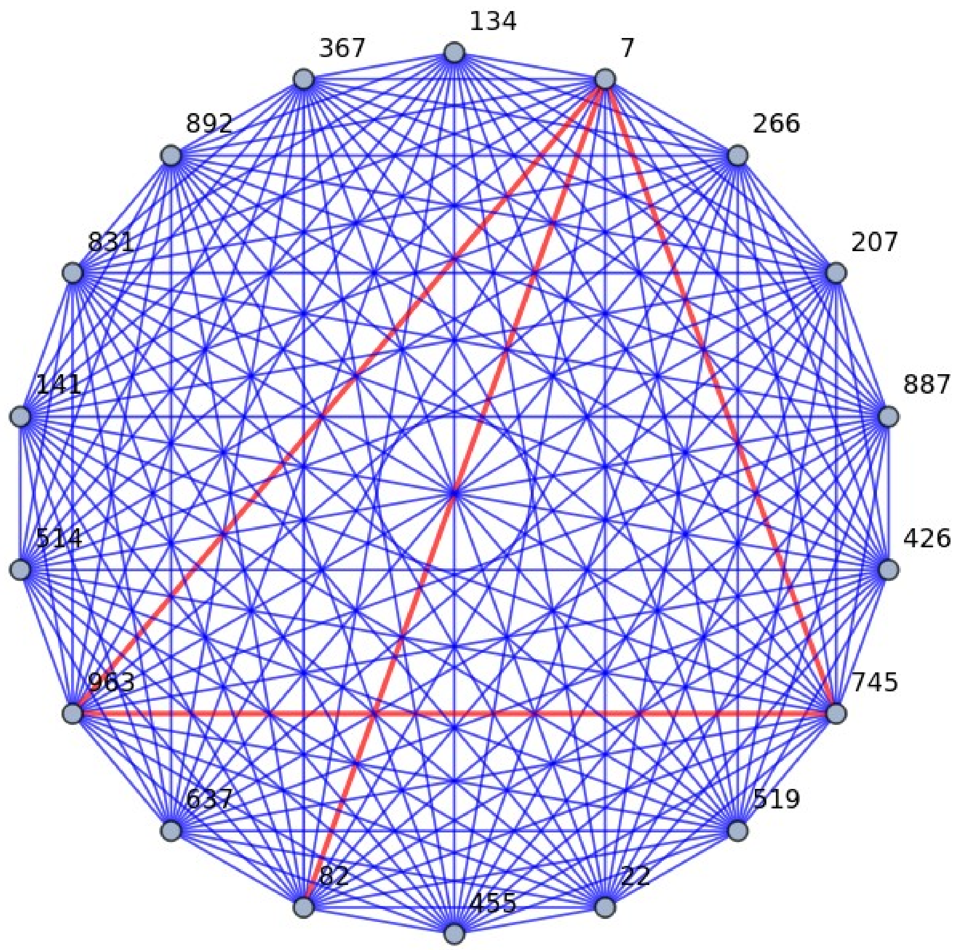


Fig: A 3-Clique found in an 18-graph

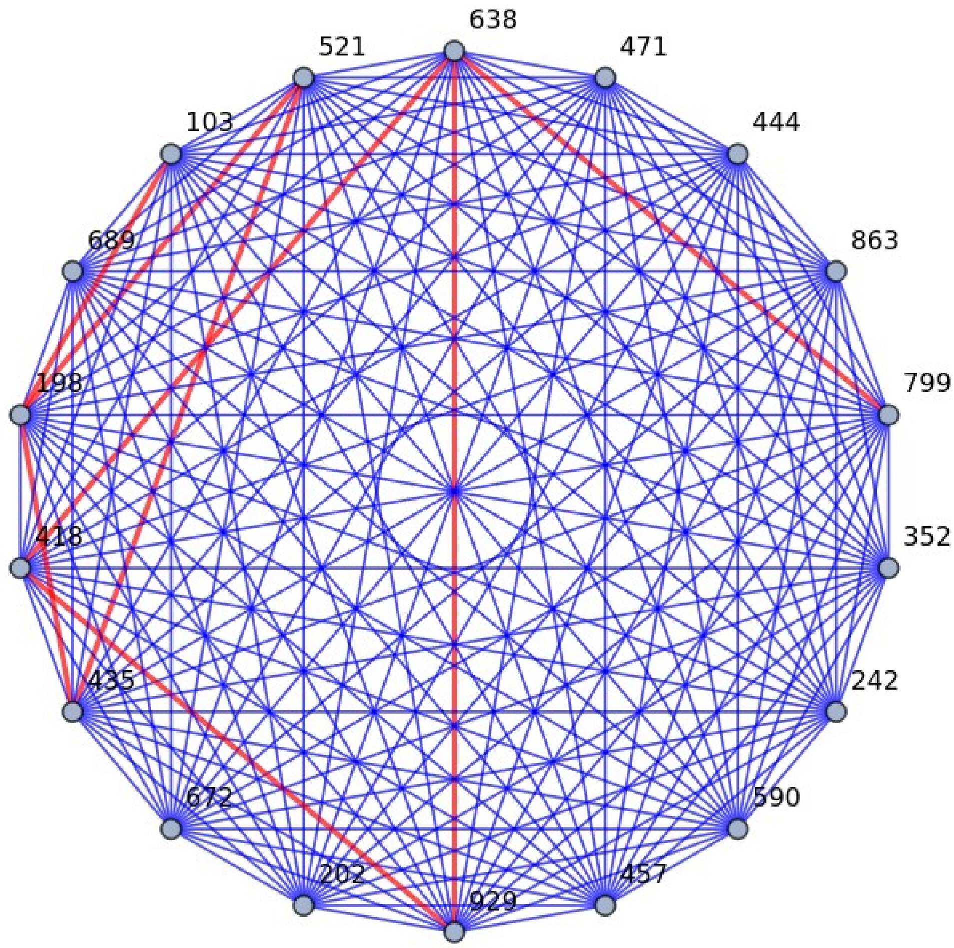


Fig: A much more complex example.

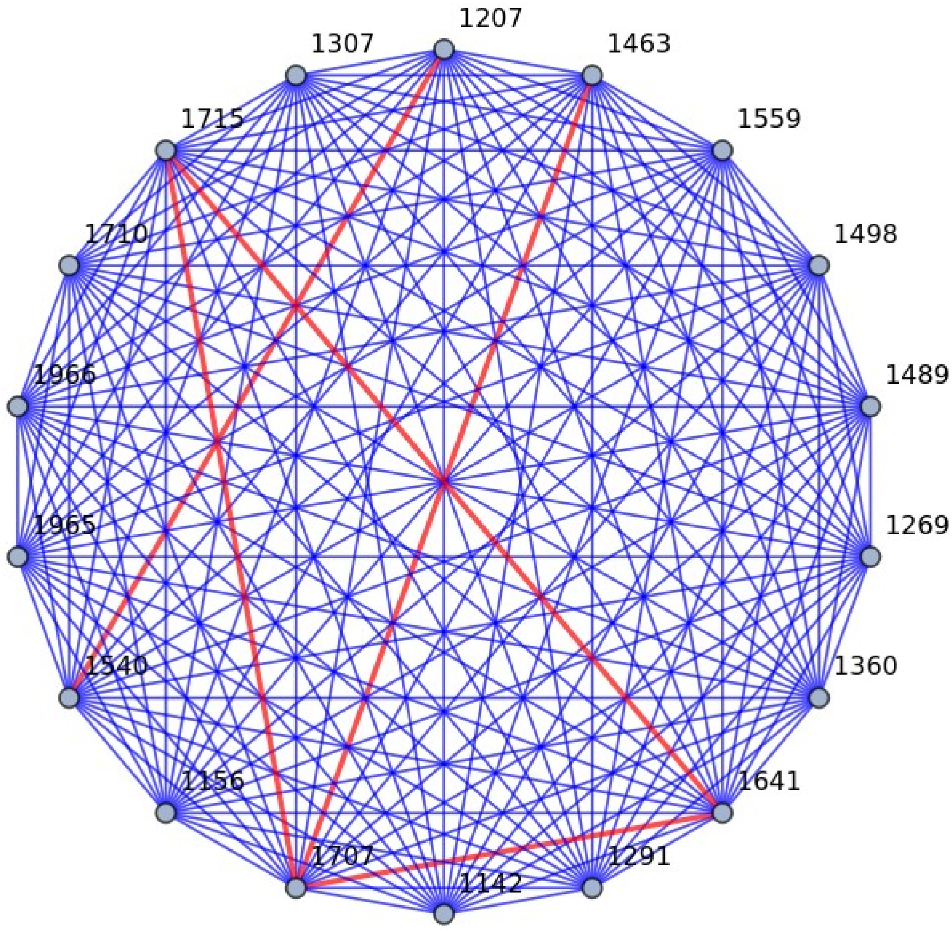
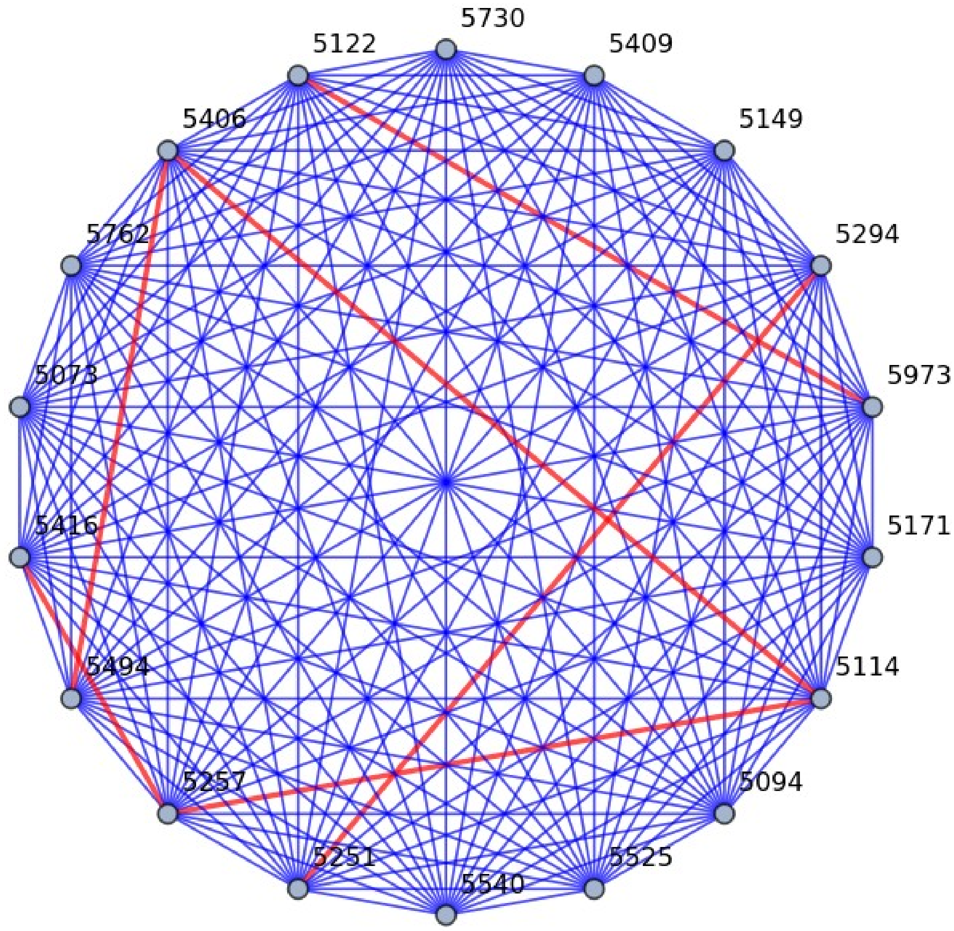


Fig: Another 3-Clique found. However, note, this is not a 4-Clique still that we are searching for in this 18-graph.


5730
5122
5409
5406
5149
5762
5294
5073
5973
5416
5171
5494
5114
5257
5094
5251
5525
5540

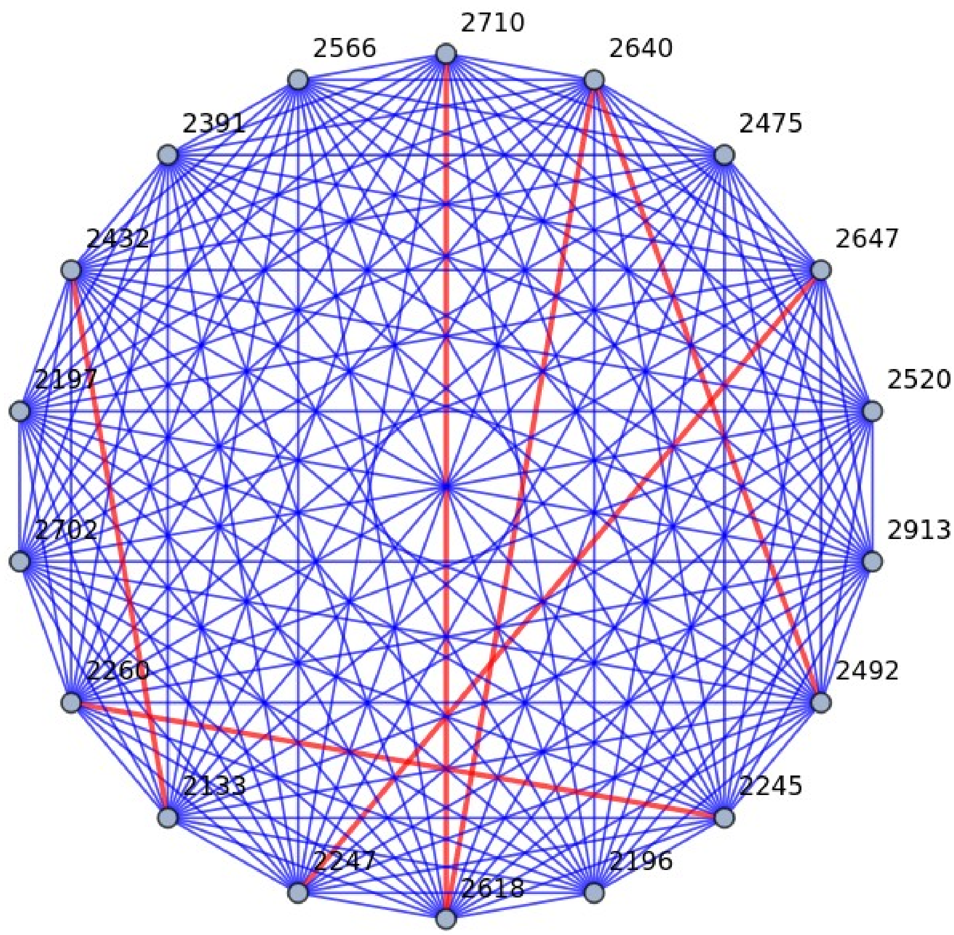


Fig: Another complex graph with vertices 2710 and 2618 knowing each other, etc.

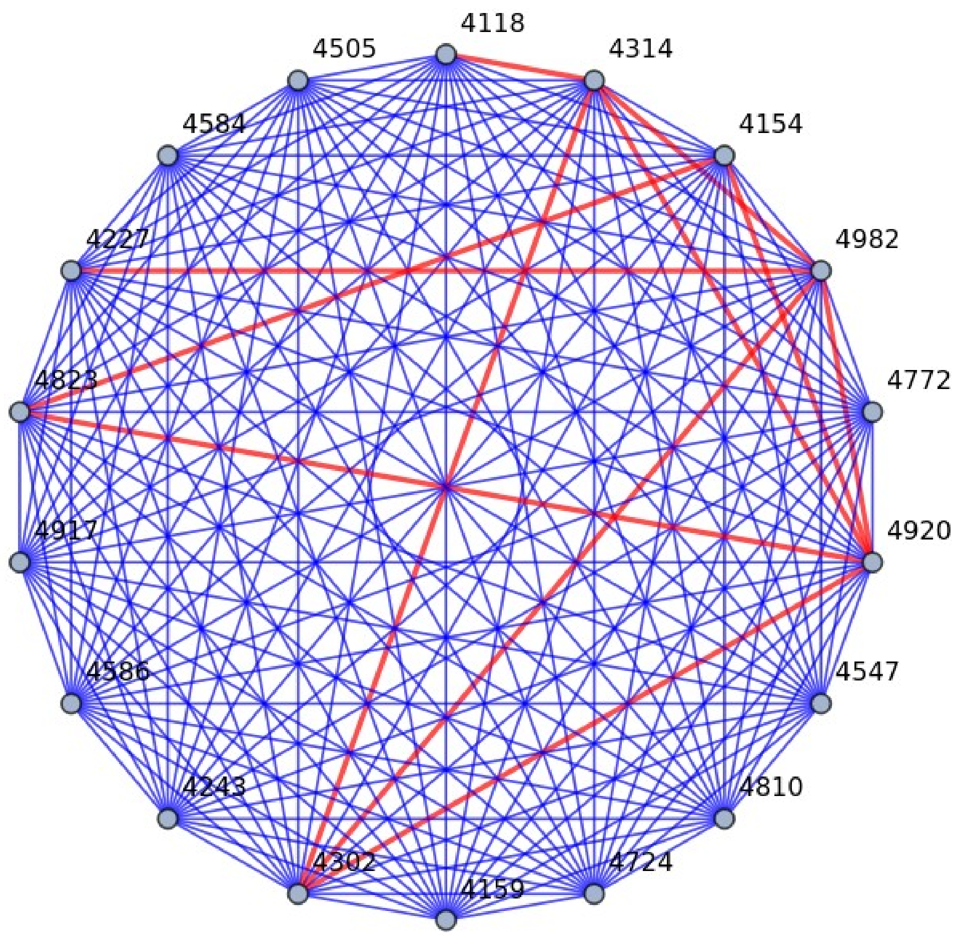


Fig: Yet another example where node 4154, 4828 and 4920 forms a 3-clique, and node s 4982, 4920 and 4302 also forms a 3-clique

CHAPTER 6

RESULT AND ANALYSIS

The expected results for this project are:

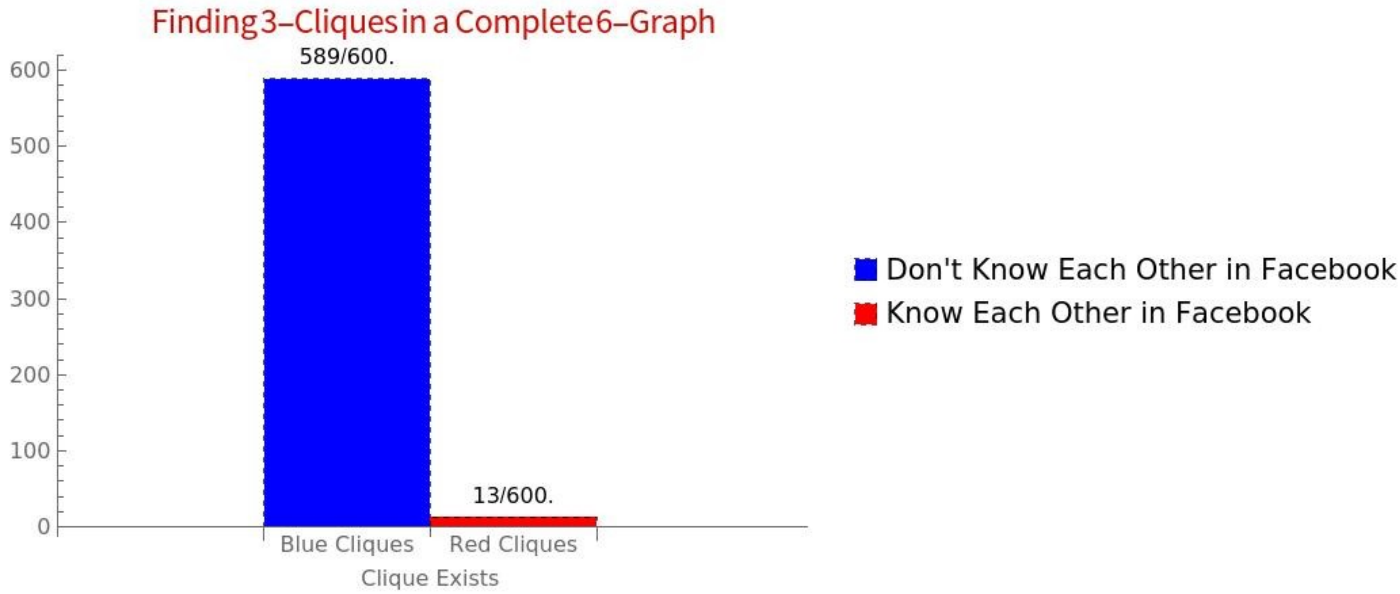
i. Ito determine the probability of 3 people and 4 people knowing each other versus not knowing each other in a random party thrown at MIT consisting of 6 people and 18 people respectively.

ii. Analyze Ramsey Theorem using existing Social Media Analytics

iii. Have a practical understanding of the famous party problem, how likely it is for people to not-know each other compared to know each other based on the data used.

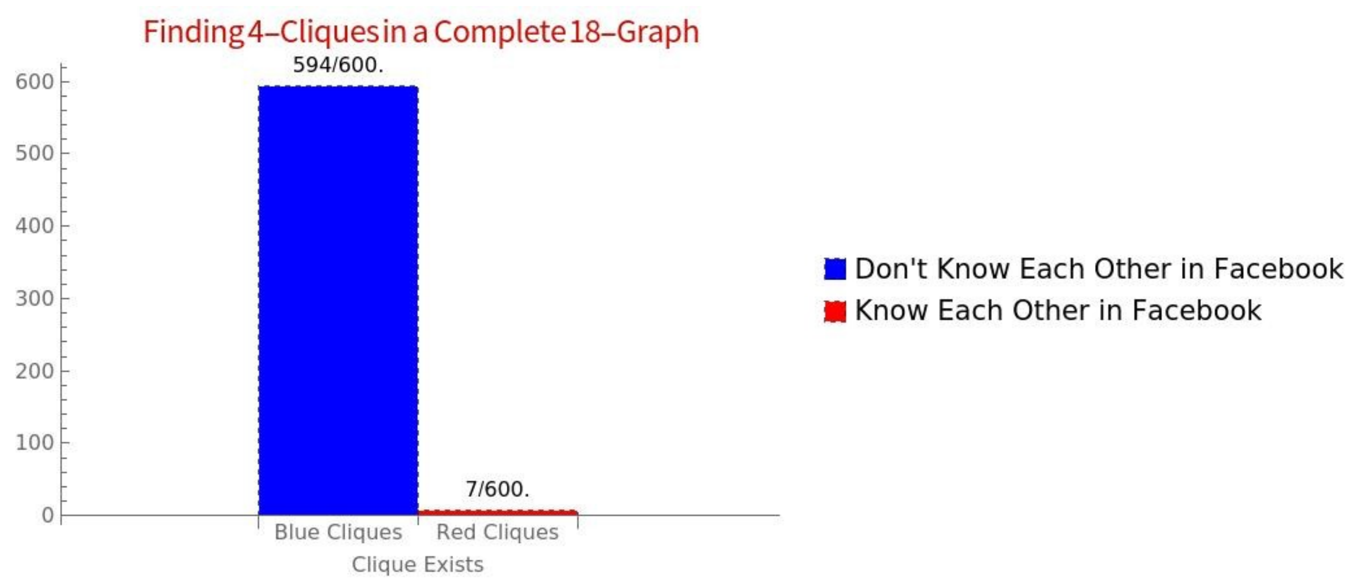
RESULT: In this project I found out that people not-knowing each other inside a college like MIT is far more than people knowing each other, when they are in a random party of 6 people or 18 people. The conclusing was drawn from Social Media Analytics and graph manipulation and concepts of Ramsey Theorem.

.



Probability that there are 3 people knowing each other in a party of 6 at MIT is roughly 0.0216667

Probability that there are 3 people not knowing each other in a party of 6 at MIT is roughly 0.981667



Probability that there are 4 people knowing each other in a party of 18 at MIT is roughly 0.0116667

Probability that there are 4 people not knowing each other in a party of 18 at MIT is roughly 0.99

Difficulties Faced

1. The main problem that was faced during this project was the limited performance of the software and computer hardware used.

In the absence of GPUs, most graph computations can be a cumbersome task. I used my personal computer with a processor of Intel Core i5 for this experiment, which sometimes resulted in lag in obtaining results.

2. Apart from that, another major difficulty was the big size of data. Often dealing with huge datas can be difficult for manipulation unless using a distributed system.

3. The last issue faced in this research was the dependency of the result on data used. The final outcome is entirely dependant on the particular data I am using, and it no way guarantees that the outcome will be any way close to this outcome if any other data is used.

Analysis of project

Table 6.1 : Analysis of the probability values of Knowing Each Other and Not-Knowing Each Other using MIT data.

Sr. No.	Number of People at Party	Category of Determination	while sensor sense
1	6	**Any random 3 people Knowing Each Other:** **3 people Not Knowing Each Other:**	**0.0216667** **0.981667**
2	18	**Any random 4 People Knowing Each Other:** **4 People Not Knowing Each Other:**	**0.0116667** **0.99**

CHAPTER 7

CONCLUSION AND FUTURE SCOPE

7.1 CONCLUSION

Our experiment determines that practically, in the famous party problem, the probability of people not knowing each other is far more than people knowing each other, althought theoretically we assume both are equally likely.

7.2 FUTURE SCOPE

1. This experiment can be used to determine lower bounds for larger Ramsey Numbers
2. Similar Social Media Analysis can be done to obtain results in the fields of Combinatorics or Graph Theory.

REFERENCES

- *Bohman, Tom; Keevash, Peter (2010), "The early evolution of the H-free process", Invent. Math., **181** (2): 291–336, arXiv:0908.0429, Bibcode:2010InMat.181..291B, doi:10.1007/s00222-010-0247-x*

- *Brualdi, Richard A. (2010), Introductory Combinatorics (5th ed.), Prentice-Hall, pp. 77–82, ISBN*

- *Conlon, David (2009), "A new upper bound for diagonal Ramsey numbers", Annals of Mathematics, **170** (2): 941–960, arXiv:math/0607788v1, doi:10.4007/annals.2009.170.941, MR 2552114*.

- *Erdős, Paul (1947), "Some remarks on the theory of graphs", Bull. Amer. Math. Soc., **53** (4): 292–294, doi:10.1090/S0002-9904-1947-08785-1*.

- *Erdős, P.; Moser, L. (1964), "On the representation of directed graphs as unions of orderings" (PDF), A Magyar Tudományos Akadémia, Matematikai Kutató Intézetének Közleményei, **9**: 125–132, MR 0168494*

- *Erdős, Paul; Szekeres, George (1935), "A combinatorial problem in geometry" (PDF), Compositio Mathematica, **2**: 463–470, doi:10.1007/978-0-8176-4842-8_3, ISBN* .

- *Exoo, G. (1989), "A lower bound for R(5,5)", Journal of Graph Theory, **13**: 97–98, doi:10.1002/jgt.3190130113*.

- *Graham, R.; Rothschild, B.; Spencer, J. H. (1990), Ramsey Theory, New York: John Wiley and Sons*.
- *Gross, Jonathan L. (2008), Combinatorial Methods with Computer Applications, CRC Press, p. 458, ISBN*

- *Harary, Frank (1972), Graph Theory, Addison-Wesley, pp. 16–17, ISBN*

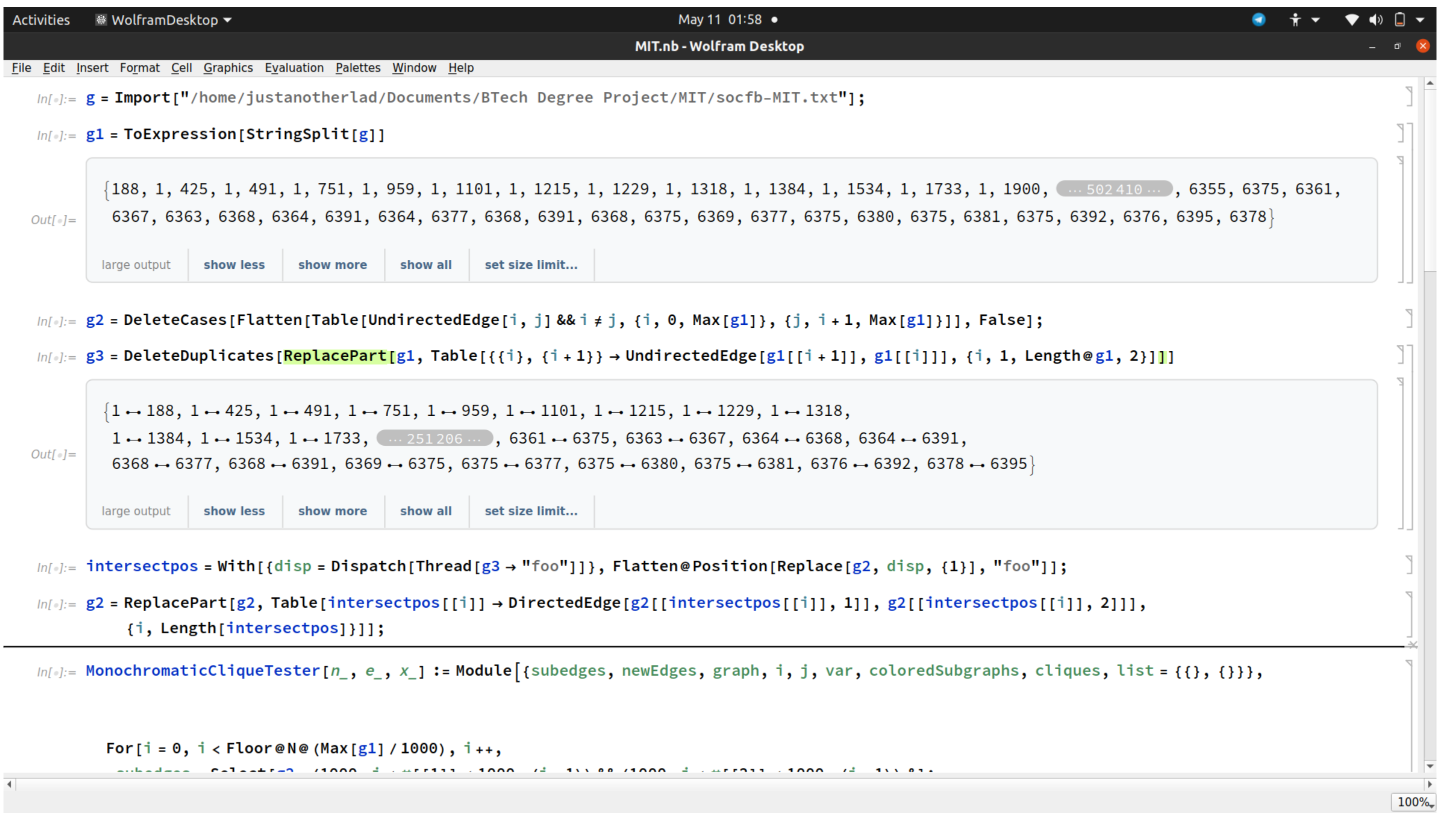
- *Ramsey, F. P. (1930), "On a problem of formal logic", Proceedings of the London Mathematical Society, **30**: 264–286, doi:10.1112/plms/s2-30.1.264*.

- *Spencer, J. (1975), "Ramsey's theorem – a new lower bound", J. Combin. Theory Ser. A, **18**: 108–115, doi:10.1016/0097-3165(75)90071-0*.

- *Bian, Zhengbing; Chudak, Fabian; Macready, William G.; Clark, Lane; Gaitan, Frank (2013), "Experimental determination of Ramsey numbers", Physical Review Letters, **111** (13): 130505, arXiv:1201.1842, Bibcode:2013PhRvL.111m0505B, doi:10.1103/PhysRevLett.111.130505, PMID 24116761*.

APPENDIX

A) PROGRAMMING



Activities WolframDesktop May 11 01:58

MIT.nb - Wolfram Desktop

File Edit Insert Format Cell Graphics Evaluation Palettes Window Help

```
    {i, Length[intersectpos]}]];

In[∘]:= MonochromaticCliqueTester[n_, e_, x_] := Module[{subedges, newEdges, graph, i, j, var, coloredSubgraphs, cliques, list = {{}, {}}},


  For[i = 0, i < Floor@N@(Max[g1] / 1000), i++,
   subedges = Select[g2, (1000 * i < #[[1]] < 1000 * (i + 1)) && (1000 * i < #[[2]] < 1000 * (i + 1)) &];
   newEdges = Replace[subedges, {DirectedEdge[a_, b_] :> Style[UndirectedEdge[a, b], Red, Thick],
      UndirectedEdge[a_, b_] :> Style[UndirectedEdge[a, b], Blue]}, {1}];
   graph = Graph[newEdges];

   For[j = 1, j <= x, j++,

    var = Subgraph[graph, {RandomInteger[{1000 * i, 1000 * (i + 1)}, e]}, VertexLabels → Automatic];

    Export["/home/justanotherlad/Documents/BTech Degree Project/MIT/Pictures/5-Clique/Ramsey_for_" <> ToString[n] <>
      "_in_" <> ToString[e] <> "_" <> ToString[i] <> "_" <> ToString[j] <> ".png", var];


    coloredSubgraphs = Quiet[GroupBy[MapAt[Lookup[LineColor], AnnotationValue[var, EdgeStyle], {All, 2}], Last → First]];
    cliques = Length@First@FindClique[#] ≥ n & /@ coloredSubgraphs;
    AppendTo[list[[1]], cliques[[1]]];
    If[Length@cliques > 1, AppendTo[list[[2]], cliques[[2]]]];
   ]
  ];


  Labeled[BarChart[{{Length@Cases[list[[1]], True], Length@Cases[list[[2]], True]}}, ChartBaseStyle → EdgeForm[Dashed],
```

100%

Activities WolframDesktop May 11 01:58

MIT.nb - Wolfram Desktop

File Edit Insert Format Cell Graphics Evaluation Palettes Window Help

```
     coloredSubgraphs = Quiet[GroupBy[MapAt[Lookup[LineColor], AnnotationValue[var, EdgeStyle], {All, 2}], Last → First]];
     cliques = Length@First@FindClique[#] ≥ n & /@ coloredSubgraphs;
     AppendTo[list[[1]], cliques[[1]]];
     If[Length@cliques > 1, AppendTo[list[[2]], cliques[[2]]]];
    ]
   ];


   Labeled[BarChart[{{Length@Cases[list[[1]], True], Length@Cases[list[[2]], True]}}, ChartBaseStyle → EdgeForm[Dashed],
     ChartStyle → {Blue, Red}, ChartLegends → {"Don't Know Each Other in Facebook", "Know Each Other in Facebook"},
     BarSpacing → None, ChartLabels -> {{"Clique Exists"}, {"Blue Cliques", "Red Cliques"}},
     LabelingFunction → (Placed[Row[{#, "/", N[x * i]}], Above] &),
     PlotLabel → Style["Finding " <> ToString[n] <> "-Cliques in a Complete " <> ToString[e] <> "-Graph", "Title", 16],
     ImageSize → Medium, PerformanceGoal → "Speed"],
    Column[{"Probability that there are " <> ToString[n] <> " people knowing each other in a party of " <> ToString[e] <>
       " at MIT is roughly " <> ToString[N[Length@Cases[list[[2]], True] / N[x * i]]],
      "Probability that there are " <> ToString[n] <> " people not knowing each other in a party of " <> ToString[e] <>
       " at MIT is roughly " <> ToString[N[Length@Cases[list[[1]], True] / N[x * i]]]}],
    LabelStyle → Directive[FontSize → 13.5, FontFamily → "Times"]]
  ] (*Gives a bar chart of how many times monochromatic n-
   cliques of colors blue and red were found resp. in a complete graph of e vertices
    (taken randomly from the entire imported social-media graph in batches of successive 1000 vertices at a time till
      all the vertices are taken into consideration) when the experiment is run a total of x times*)


In[•]:= MonochromaticCliqueTester[5, #, 100] & /@ Range[43, 48]
```

100%

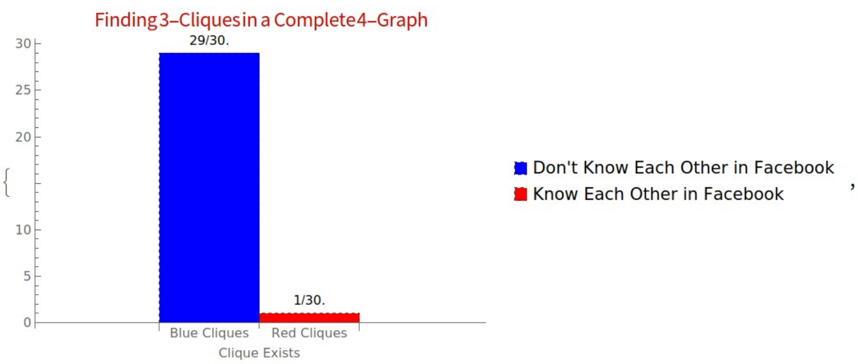
Activities   WolframDesktop   May 11 01:59

Facebook_2.nb - Wolfram Desktop

File Edit Insert Format Cell Graphics Evaluation Palettes Window Help

```
In[•]:= MonochromaticCliqueTester[3, #, 5] & /@ {4, 5, 6, 7, 8}
```

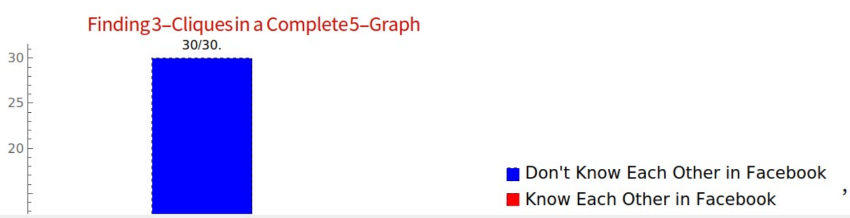
Out[•]= {



,

Probability that there are 3 people knowing each other in a party of 4 at CMU is roughly 0.0333333

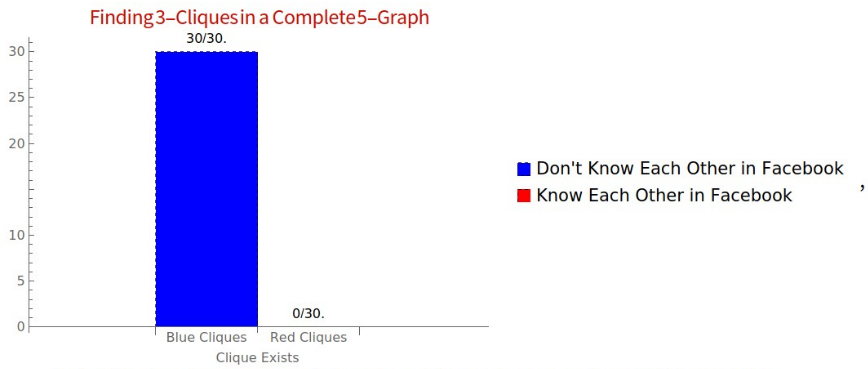
Probability that there are 3 people not knowing each other in a party of 4 at CMU is roughly 0.966667



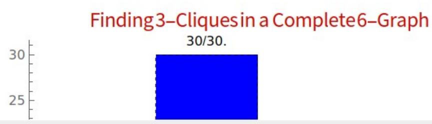
,

100%

Probability that there are 3 people knowing each other in a party of 4 at CMU is roughly 0.0333333
Probability that there are 3 people not knowing each other in a party of 4 at CMU is roughly 0.966667



Probability that there are 3 people knowing each other in a party of 5 at CMU is roughly 0.
Probability that there are 3 people not knowing each other in a party of 5 at CMU is roughly 1.



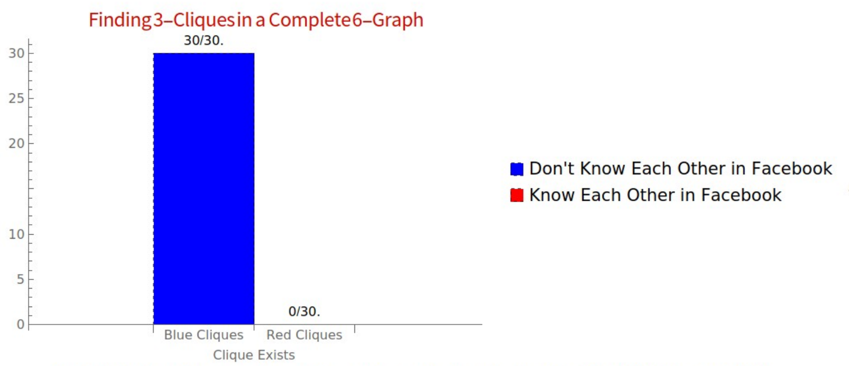
Activities WolframDesktop May 11 01:59

Facebook_2.nb - Wolfram Desktop

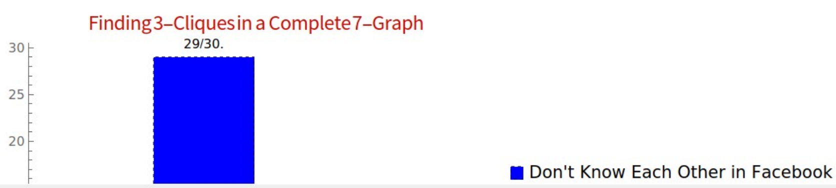
File Edit Insert Format Cell Graphics Evaluation Palettes Window Help

Clique Exists

Probability that there are 3 people knowing each other in a party of 5 at CMU is roughly 0.
Probability that there are 3 people not knowing each other in a party of 5 at CMU is roughly 1.



Probability that there are 3 people knowing each other in a party of 6 at CMU is roughly 0.
Probability that there are 3 people not knowing each other in a party of 6 at CMU is roughly 1.



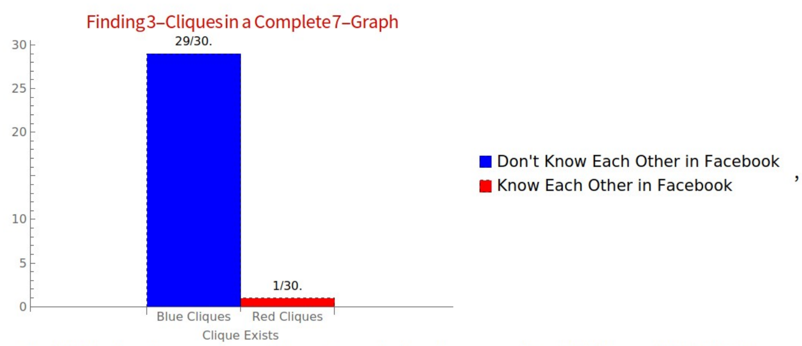
100%

Activities WolframDesktop May 11 01:59

Facebook_2.nb - Wolfram Desktop

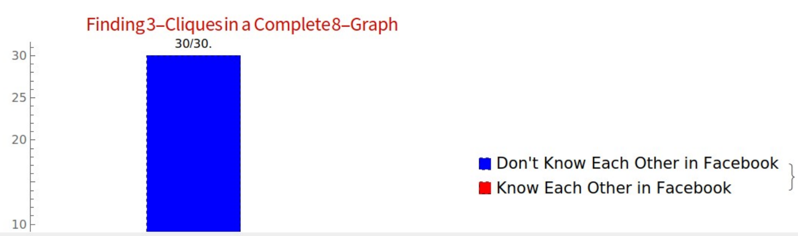
File Edit Insert Format Cell Graphics Evaluation Palettes Window Help

Probability that there are 3 people not knowing each other in a party of 6 at CMU is roughly 1.

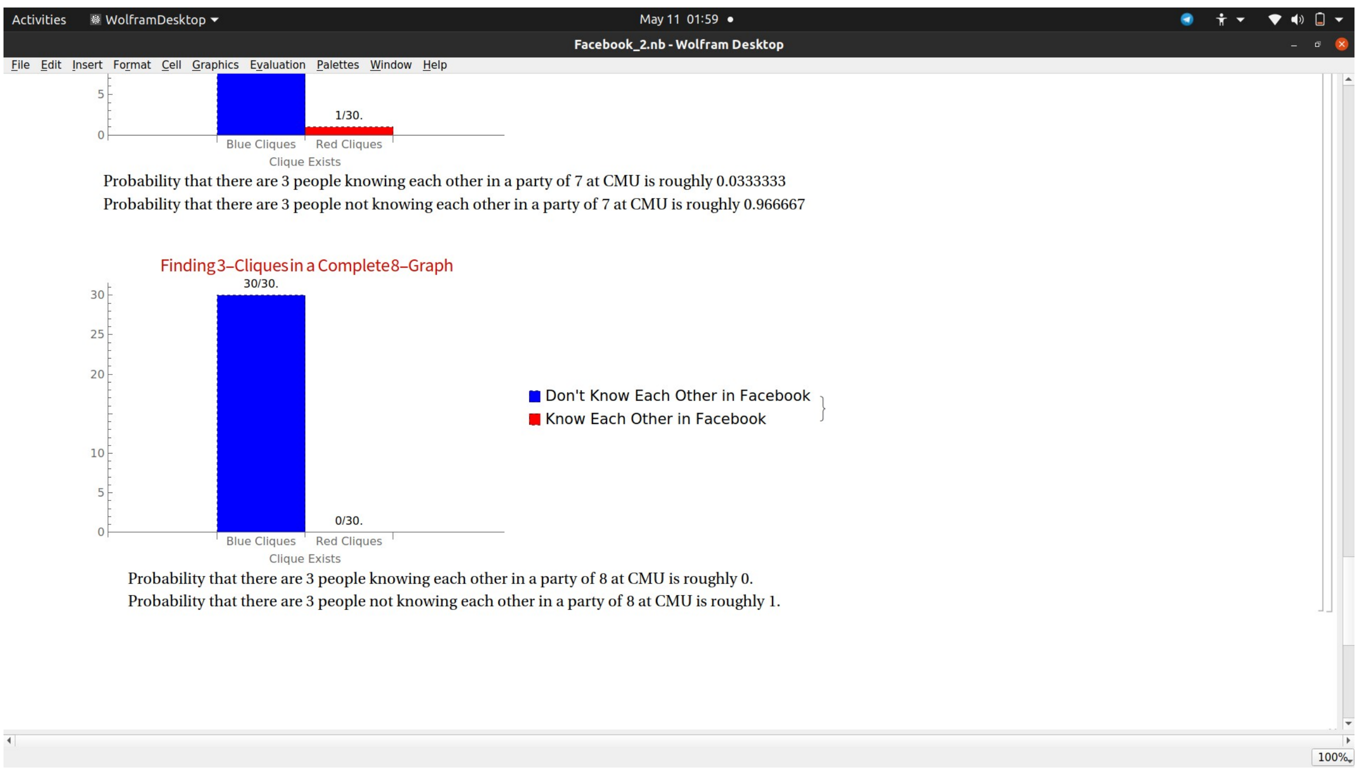


Probability that there are 3 people knowing each other in a party of 7 at CMU is roughly 0.0333333

Probability that there are 3 people not knowing each other in a party of 7 at CMU is roughly 0.966667



100%


Activities
WolframDesktop
May 11 01:59
Facebook_2.nb - Wolfram Desktop
File Edit Insert Format Cell Graphics Evaluation Palettes Window Help
1/30.
Blue Cliques
Red Cliques
Clique Exists
Probability that there are 3 people knowing each other in a party of 7 at CMU is roughly 0.0333333
Probability that there are 3 people not knowing each other in a party of 7 at CMU is roughly 0.966667
Finding 3–Cliques in a Complete 8–Graph
30/30.
0/30.
Blue Cliques
Red Cliques
Clique Exists
Don't Know Each Other in Facebook
Know Each Other in Facebook
Probability that there are 3 people knowing each other in a party of 8 at CMU is roughly 0.
Probability that there are 3 people not knowing each other in a party of 8 at CMU is roughly 1.
100%