The mean average clustering coefficient for the 100 sampled graph is 4.518×10^{-5}

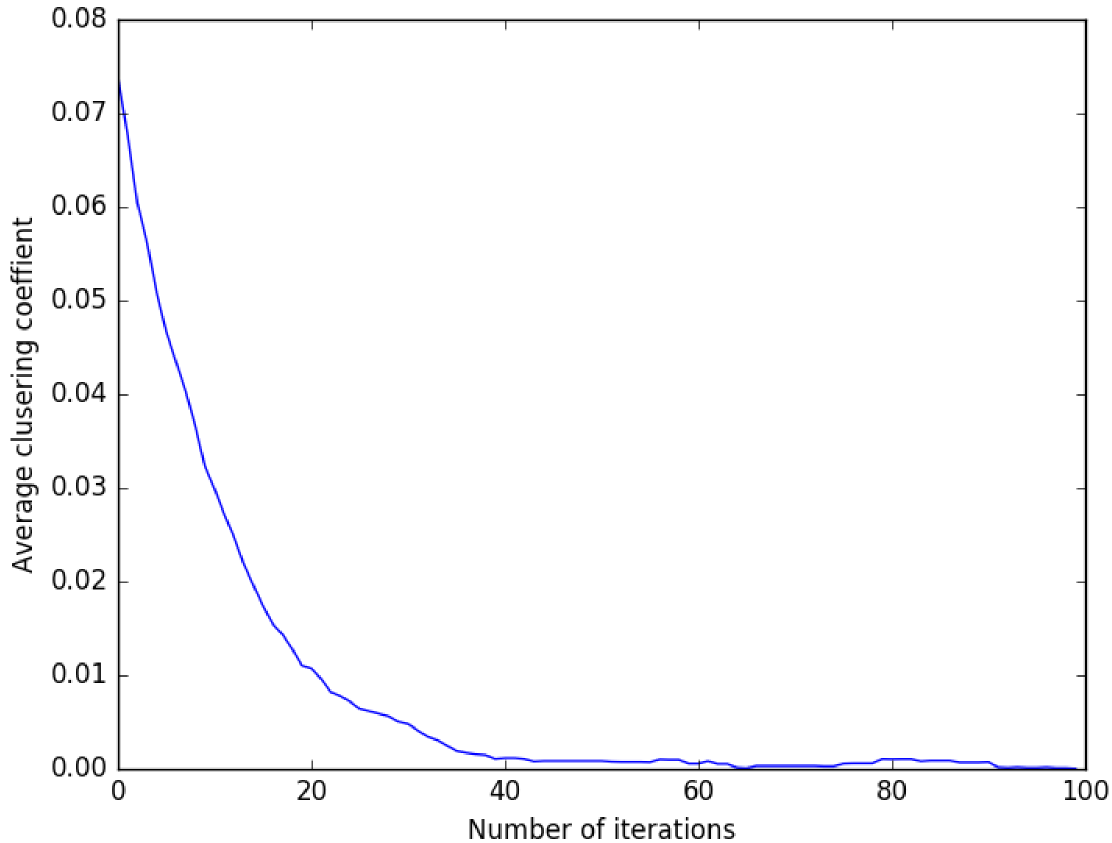

Figure 1: Average clustering coefficient V.S. Number of iterations

The average clustering coefficient decreases as number of iterations increases. This is reasonable because when we randomly rewrite the edges of the original graph, our graph tends more and more towards a completely random graph, which generally have much less clustering coefficient compared to a small-world graph.

(a)

The count of each type is as follows:
type t0: 58732
type t1: 396548
type t2: 451711
type t3: 4003085
The fraction of each type is as follows:
type t0: 0.01196153
type t1: 0.08076209
type t2: 0.09199674
type t3: 0.81527964

(b)

The fraction of positive edge is : 0.8324025
The fraction of negative edge is: 0.1675975
The probability of each type:
type t0: $(1-p)^3 = 0.004707633$
type t1: $3p(1-p)^2 = 0.07014387$
type t2: $3p^2(1-p) = 0.3483819$
type t3: $p^3 = 0.5767666$

(c)

Type t0, t1, t3 seem to be more in the real network as calculated in (a) than in (b).
For t1 and t3, it's because that they are consistent with the balance theory. For t0, it's probably because there are two few of them in both (a) and (b), and the result may be subject to some variance, or the network that we are using in (a) is still under evolution, maybe eventually its t0 fraction will drop below that of (b).

(a)

A simple lower bound for $|T|$ is $\lfloor \frac{n}{3} \rfloor$

(b)

0.5

(c)

The upper bound is $0.5^{\lfloor \frac{n}{3} \rfloor}$. Note that $0.5 < 1$ and $\lfloor \frac{n}{3} \rfloor > \frac{n}{4}$ for big n, let P denote the probability, we have

$$\begin{aligned} 0 &\leq \lim_{n \to \infty} P && (1) \\ &\leq \lim_{n \to \infty} 0.5^{\lfloor \frac{n}{3} \rfloor} && (2) \\ &\leq \lim_{n \to \infty} 0.5^{\frac{n}{4}} && (3) \\ &= 0 && (4) \end{aligned}$$

Therefore, we have $\lim_{n \to \infty} P = 0$.

(d)

Because in order for G_B to happen, the even "all of the triangles in T are balanced" must happen, therefore $P(G_B) \leq P$, therefore $0 \leq \lim_{n \to \infty} P(G_B) \leq \lim_{n \to \infty} P = 0$, hence $\lim_{n \to \infty} P(G_B) = 0$

No, this is not true. We can construct a counterexample as follows:

Suppose we have A, an unbalanced triad, which has 2 positive edges and 1 negative edges. And e, a positive edge of A, forms and only forms 2 extra triads with other 4 positive edges. Let B and C denote these 2 triads(Then B and C are both balanced). Then during the dynamic process, if we happen to choose A and choose e to flip, A will become balanced but B and C will become unbalanced. So the total number of balanced triads decreases by 1.

100%

No, it's impossible.
Let discuss the two signs that the edge AD can take:
If AD is +, then in order for triad ABD and ACD to be balanced, BD and CD both must be +, but this will make triad BCD have two +(BD and CD) and one -(BC), so it's unbalanced. So AD is + is not feasible.
If AD is -, then in order for triad ABD and ACD to be balanced, BD and CD both must be -, but this makes triad BCD have three -, so it's unbalanced. So AD is - is not feasible, either.
Therefore it's impossible to add a node D such that it forms signed edges with all existing nodes (A, B, and C), but isn?t itself part of any unbalanced triangles.

No, it's impossible.
We have proved in 2.5 that it's impossible to add a node D such that it forms signed edges with all existing nodes in a type t2 triad, but isn?t itself part of any unbalanced triangles. We now prove the same thing for type t0 triad:
Similar to 2.5, we denote the 3 nodes of a type t0 triad by A, B and C, respectively. Now A, B and C will all have - edges among them. Let discuss the two signs that the edge AD can take:
If AD is +, then in order for triad ABD and ACD to be balanced, BD and CD both must be -, but this will make triad BCD have three -, so it's unbalanced. So AD is + is not feasible.
If AD is -, then in order for triad ABD and ACD to be balanced, BD and CD both must be +, but this makes triad BCD have two +(BD and CD) and one -(BC), so it's unbalanced. So AD is - is not feasible, either.
Therefore it's impossible to add a node D such that it forms signed edges with all existing nodes (A, B, and C), but isn?t itself part of any unbalanced triangles.
Since for an unbalanced complete graph, there will always be at least one of two kinds of unbalanced triads(type t0 and type t2). So if a new node forms edges to all existing nodes in the unbalanced graph, it has to at least link to one triad that is either type t0 or type t1. But by doing so, the new node must be putting itself in at least on unbalanced triad, according to 2.5 and our previous proof.
Therefore, it is impossible for a new node X able to join the network and form edges to all existing nodes in such a way that it does not become involved in any unbalanced triangles.

For graph1, B will win by 96 votes
For graph2, B will win by 256 votes

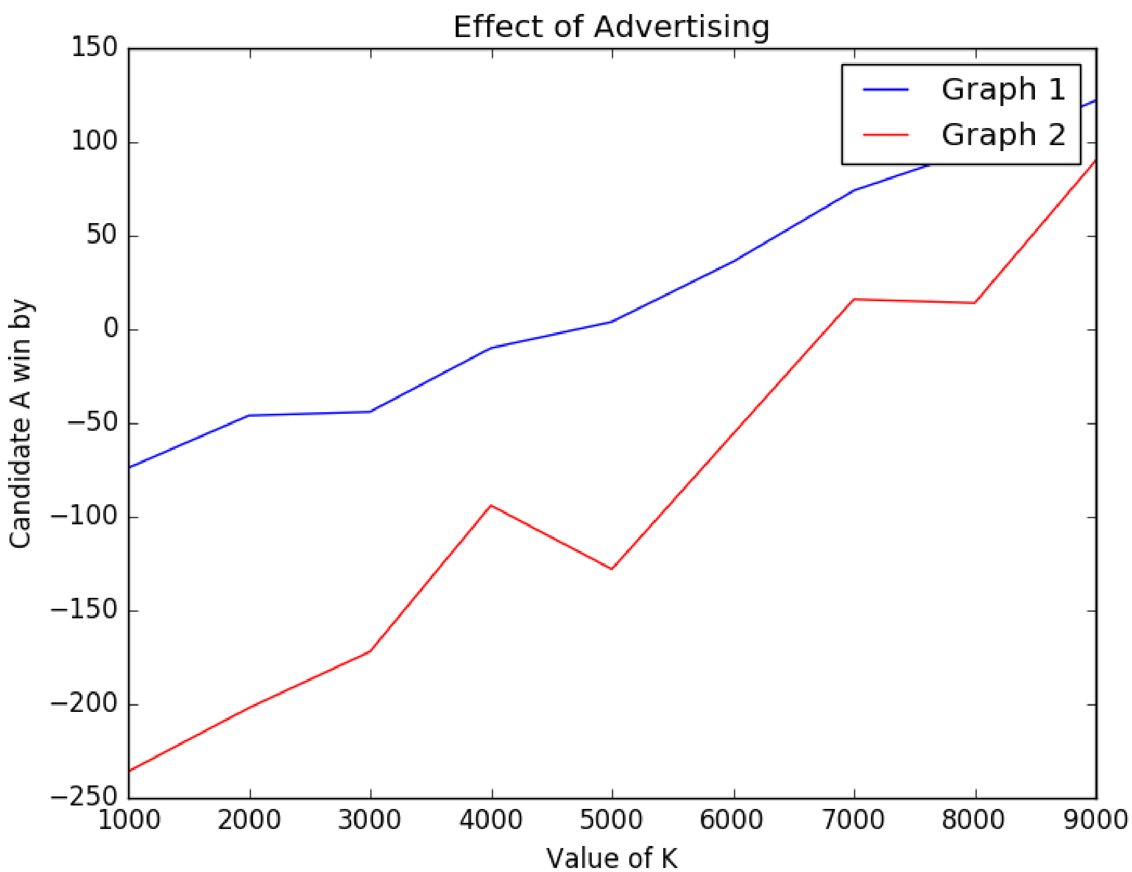

Figure 2: Effect of Advertising

Graph1: \$5000
Graph2: \$7000

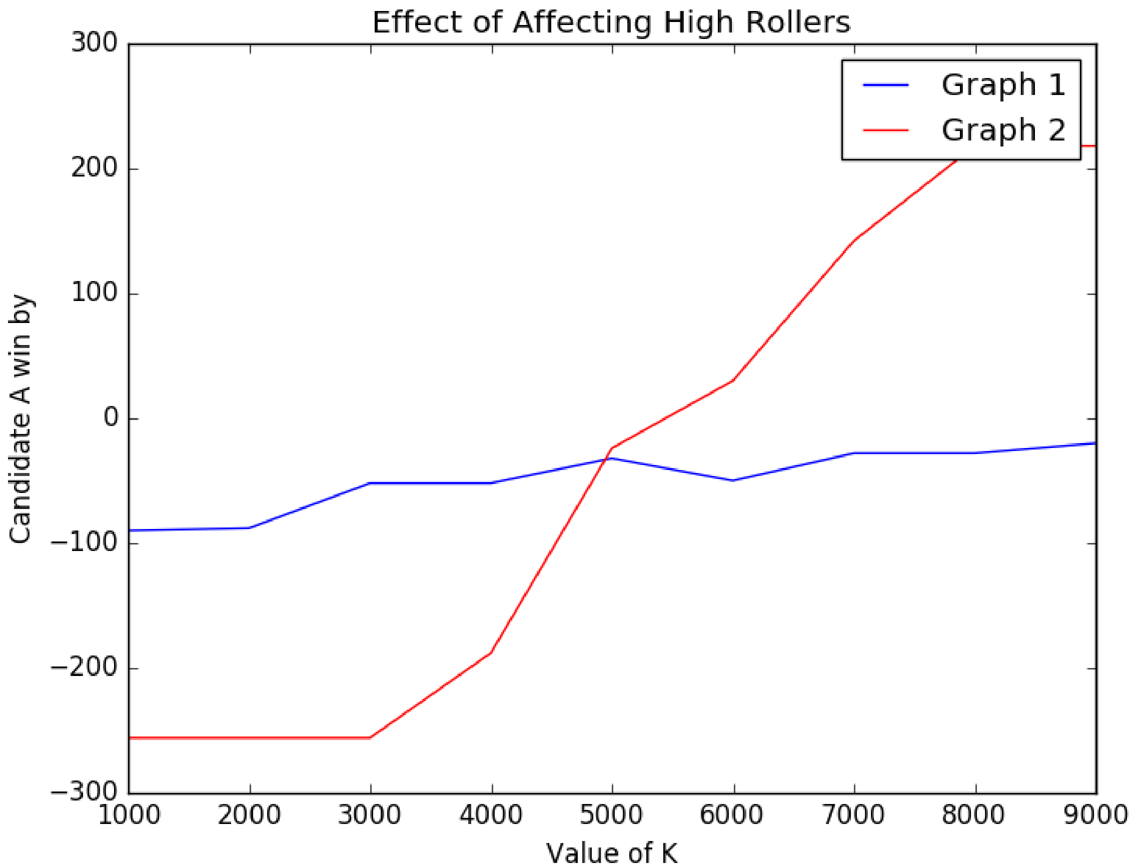

Figure 3: Effect of Affecting High Rollers

Graph1 can never win. Even with $9000, A is still down by 20 votes
Graph2: $6000

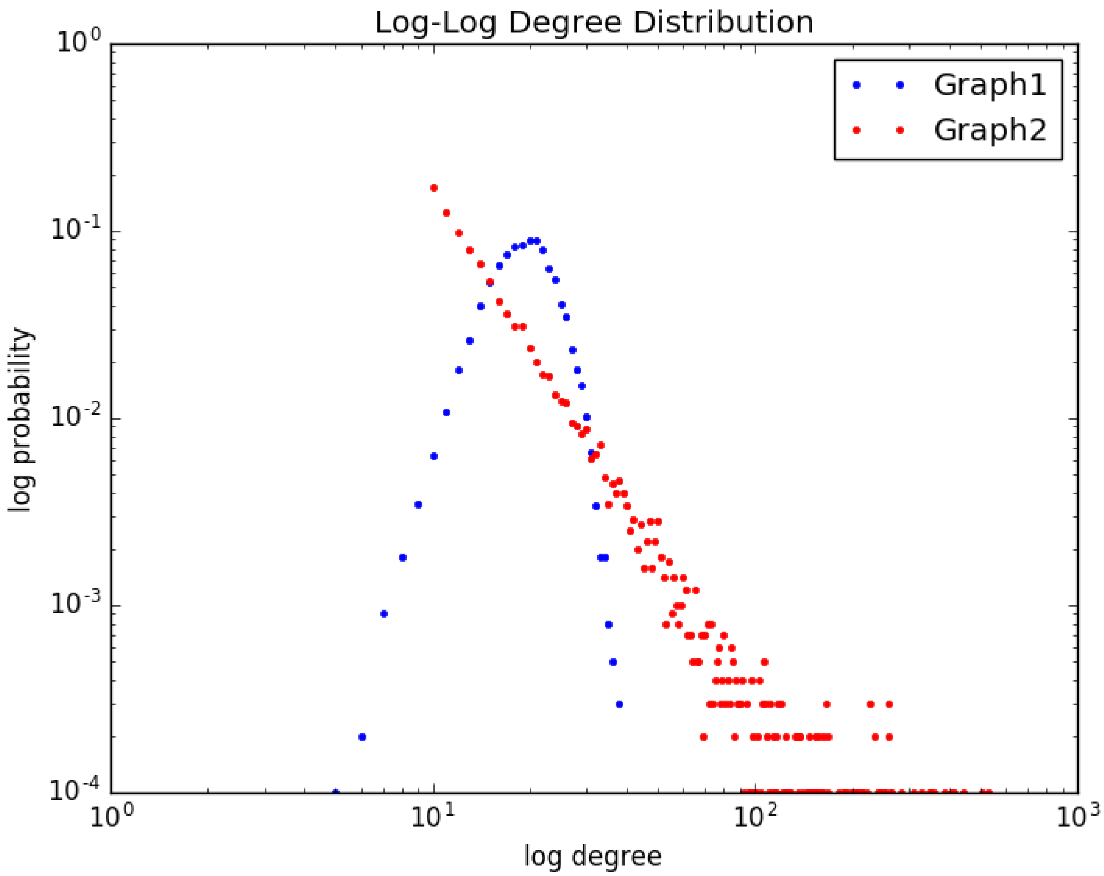

Figure 4: Log-Log Degree Distribution

Graph 1 is Erdos-Renyi random graph and Graph2 was generated from a preferential attachment model.
Because in Erdos-Renyi, no nodes have very high degree, while in a preferential attachment model, some nodes tend to have very high degree. Therefore changing votes for the highest degree people will word well on a preferential attachment model and fail on Erdos-Renyi model.

Information sheet
CS224W: Social and Information Network Analysis

Assignment Submission Fill in and include this information sheet with each of your assignments. This page should be the last page of your submission. Assignments are due at 11:59pm and are always due on a Thursday. All students (SCPD and non-SCPD) must submit their homeworks via GradeScope (http://www.gradescope.com). Students can typeset or scan their homeworks. Make sure that you answer each (sub-)question on a separate page. That is, one answer per page regardless of the answer length. Students also need to upload their code at http://snap.stanford.edu/submit. Put all the code for a single question into a single file and upload it. Please do not put any code in your GradeScope submissions.

Late Homework Policy Each student will have a total of *two* free late periods. *Homeworks are due on Thursdays at 11:59pm PDT and one late period expires on the following Monday at 11:59pm PDT*. Only one late period may be used for an assignment. Any homework received after 11:59pm PDT on the Monday following the homework due date will receive no credit. Once these late periods are exhausted, any assignments turned in late will receive no credit.

Honor Code We strongly encourage students to form study groups. Students may discuss and work on homework problems in groups. However, each student must write down their solutions independently i.e., each student must understand the solution well enough in order to reconstruct it by him/herself. Students should clearly mention the names of all the other students who were part of their discussion group. Using code or solutions obtained from the web (github/google/previous year solutions etc.) is considered an honor code violation. We check all the submissions for plagiarism. We take the honor code very seriously and expect students to do the same.

Your name: Bowen Yao
Email: boweny@stanford.edu **SUID:** 06129749

Discussion Group: None

I acknowledge and accept the Honor Code.

BY