In this lecture we will analyze the Independent Cascade Model. Consider a network in which every individual vertex has the same preference (i.e. all vertices in the network are assigned the color white). Next, we will assign another color to some of the vertices. Let’s say that a subset $S$ of the vertices becomes black. We will call these vertices early adopters. Next, consider the following propagation routine. For every black vertex, each of its white neighbors becomes black with probability $p$. This process is repeated for every white vertices that turn black. We are interested, for a given $k = |S|$, what would be the best way to choose $S$ such that after the propagation takes place, the number of vertices colored in black is maximal.

Unfortunately, solving the problem of influence maximization is NP-hard. However, in this lecture, we’ll provide an algorithm that gives a good approximation of the optimal. First, we’ll show that the expected number of vertices that change color given a starting set $S$ is a submodular set function.

Let’s take a look at what a submodular set function is. Let $f : 2^V \rightarrow \mathbb{R}$, where $V$ is the set of vertices. We say $f$ is submodular if for any sets $A \subseteq B \subseteq V$ and any vertex $e \in V \setminus B$ the following equation holds:

$$f(A \cup \{v\}) - f(A) \geq f(B \cup \{v\}) - f(B).$$

In other words, adding an element to a smaller set helps more than adding it to a larger set.

The discrete derivative of a function $f$ is $\Delta(v|S) = f(S \cup \{v\}) - f(S)$. This value is called the marginal gain of $v$ in the context of $S$. Thus, a function is submodular if $\Delta(v|A) \geq \Delta(v|B)$, for $A \subseteq B \subseteq V$ and $e \in V \setminus B$.

We say a submodular function is monotone if for any $e \in V$ and $A \subseteq V$, $\Delta(v|A) \geq 0$. The cut function is an example of a non-monotone submodular function.

Next, let’s look at the sum of submodular functions. Let $f_1, f_2, \ldots, f_n$ be a set of submodular functions. Then, $f = \sum_{i=1}^{n} \alpha_i f_i$ is also submodular if $\alpha_i \geq 0$. This can be easily proved by checking the definition of submodular functions. $f(A \cup \{v\}) - f(A) \geq f(B \cup \{v\}) - f(B)$ becomes

$$\sum_{i=1}^{n} \alpha_i f_i(A \cup \{v\}) - \alpha_i f_i(A) \geq \sum_{i=1}^{n} \alpha_i f_i(B \cup \{v\}) - \alpha_i f_i(B).$$

As the inequality holds for every $\alpha_i$ then it’s also true for the sum.

Now, we’ll show that the expectation function for the number of adopters is a submodular set function. Consider the initial network. Each edge will represent a coinflip
landing heads with probability $p$, and tails with probability $1 - p$. Let’s fix this coinflip for every edge. Given such a fixed graph and an initial set $S$, we have a deterministic realization of how the influence will propagate through the graph. Now, let’s look back at our definition of a submodular function. $f(A \cup \{v\}) - f(A) \geq f(B \cup \{v\}) - f(B)$, for $A \subseteq B \subseteq V$ and $e \in V \setminus B$. Note that for a fixed realization graph $f(A \cup \{v\}) - f(A)$ will represent the nodes reachable from $v$ that are not reachable from $A$. Given this observation, the above equation comes out immediately to be true. As $|A| \leq |B|$, the nodes reachable from $v$ that reachable from $A$ will always be more than the nodes reachable from $v$ that are not reachable from $B$. Thus, for a certain deterministic graph $i$, the propagation function is submodular. If we sum over all these graphs we get:

$$f(S) = \sum_{G_i} P(G_i \text{ is realized}) |f_{G_i}(S)|.$$  

As $f(S)$ is the sum of several submodular functions, $f(S)$ will also be submodular. Thus, the expectation function is submodular.

Now, let’s consider an algorithm that offers a good approximation of the optimal set $S^*$. We’ll start off with a set $S = \emptyset$. At each step we’ll add to the set the vertex $v$ such that $\Delta(v|S)$ is maximum amongst all $v$. We’ll repeat the process until our set hits a target size. This algorithm will generate an expected number of adopters $f(S)$ that is at least $(1 - 1/e)f(S^*)$.

Proof for this algorithm will come in a future lecture.