1. What is Kickstarter?

- Crowdfunding website launched in 2009
- People can create a page to raise money for a project:
  - They have to decide on a funding goal and a campaign duration
  - People pledge money to the project in exchange for various rewards
  - Backers are charged only if the funding goal is reached at the end of the campaign
- As of June 2013:
  - More than 42,000 projects funded
  - $555 million raised
  - 4.1 million of backers
- Only half of the projects reach their goal: can we predict which?

2. Project Example: The Veronica Mars Movie Project

- Project to create a movie sequel of a famous TV show
- Campaign lasted from March 13th to April 13th 2013
- Ambitious funding goal: $2 million
- Huge success: 91,585 backers, $5,702,153 raised

3. Our Dataset

- Web crawler started in September 2012
- Automatically discovers new projects on the Recently launched page
- Regularly checks the status of live projects:
  - Number of backers
  - Money pledged
- Monitor Twitter in parallel to record mentions of Kickstarter
- Preprocessing:
  - Time is normalized to $[0,1]$
  - Pledged money is normalized with respect to the project’s goal

4. Dataset Summary

<table>
<thead>
<tr>
<th>Projects</th>
<th>Backers</th>
<th>Pledges</th>
<th>Tweets</th>
</tr>
</thead>
<tbody>
<tr>
<td>16,042</td>
<td>1,309,295</td>
<td>2,265,156</td>
<td>738,176</td>
</tr>
</tbody>
</table>

- Average project statistics:
  - Successful: 77,395
  - Failed: 830,807
  - Total: 1,604,202
  - Proportion: 48.24%
  - Goal ($\): 9595
  - Duration (days): 30.89
  - Number of backers: 262
  - Final amount: 216.60%
  - Number of tweets: 73

- Pledged money for successful/failed projects relative to their goal:

5. Time-series Predictors

- Predict success of a project based on its pledged money over time
- Use partial information: from time $0$ to time $t$, $t \leq 1$ (trajectory)
- $k$-Nearest Neighbors
  - Find the $k$ projects that have the closest trajectories
  - Predict success if the majority of them are successful, failure otherwise
- Markov Chain
  - Discretize the (time, money) space into a $N_t \times N_M$ grid
  - Consider the pledged money $M(n)$ at each time step $n$ as a random variable
  - Learn transition probabilities $P[M(n + 1) = m_{n+1} | M(n) = m_n, n] = P_{m_n, m_{n+1}}(n)$.

6. Social Predictors

- Instead of using pledged money, use social features at time $t$
- Train a SVM classifier using project goal/duration and social features
- Tweets
  - Number of tweets/replies/retweets
  - Number of unique users that tweeted
  - Estimated number of backers (using tweet’s text, e.g. “I just backed project X”)
- Co-backers graph
  - Build a graph where vertices are projects
  - Edges between projects represent common backers
  - Extract several features from this graph:
    - Number of projects with common backers
    - Number/proportion of successful projects with common backers
  - Number of backers, number/proportion of first-time backers (i.e. with only one backed project)

7. Results

- Can we use social predictors to improve by the time-series ones?
- Train a SVM to combine the four individual predictions into one

- Very useful at the start: first combined prediction 3.6% more accurate