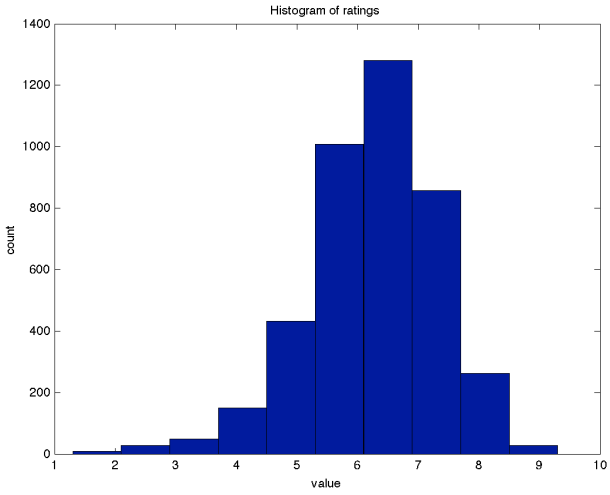
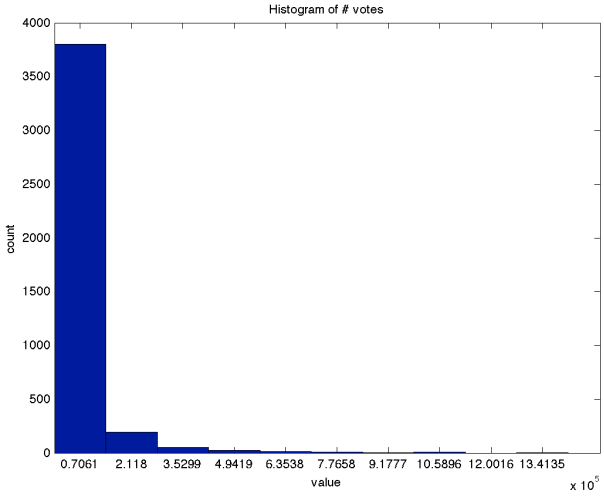
```
%%
% imdb_scrape.m
% written by Jason G. Fleischer
%
%
% Based on an argument we had on The Facebook, I want to test a theory that
% the mid 80s (specifically 84-86) were unusually good at generating
% "classic/iconic" movies or if I just think so because that's my teen years.
% I mean, come on, Ghostbusters? The Breakfast Club? Alien? Stand By Me?
% To address this question I'm going to use MATLAB to scrape data off of
% IMDB and look at the top 100 movies in terms of US Box office.  I'm going
% to try to figure out how to use various combinations of the average user
% rating and/or # of rating votes as a measure of 'classic-ness', and look
% at how that measure changes from year to year.
%
% AT THIS POINT, UNLESS YOU LIKE READING CODE YOU SHOULD
% PROBABLY SKIP AHEAD TO THE RESULTS
%
% Note that my solution is hard-coded to features of the
% current IMDB html... its going to fail if they change anything
% Please pardon the messiness of this code, and know that this is probably
% the best code commenting I've done in years :)
%
% This code is tested and working in MATLAB R2013a and IMDB's website as of
% March 13, 2015
%
% Please feel free to use and adapt this code.  I would like to hear from
% you if you have a different analysis/viewpoint on this data:
% jason.g.fleischer@gmail.com

years=[1964:2004]; % let's not even worry about films that are less than 10
years old, its impossible to decide if they are classic yet

data={}; % data includes title for later exploration
allvotes=[]; % for convenience we'll use these matrices for raw summary
statistics
allratings=[];
xi=0;
for xx=years,
    xi=xi+1;
    disp(['Scraping ' num2str(xx)]);
    s1=urlread(sprintf('http://www.imdb.com/search/title?
at=0&sort=boxoffice_gross_us&title_type=feature&year=%s,
%s',num2str(xx),num2str(xx)));
    s2=urlread(sprintf('http://www.imdb.com/search/title?
at=0&sort=boxoffice_gross_us&start=51&title_type=feature&year=%s,
%s',num2str(xx),num2str(xx)));
    allscrape=[s1 s2]; % imdb only serves 50 movies on a page, combine two
pages to get the reqd data
    indxs=strfind(allscrape,'wlb_wrapper'); % this string marks the beginning
of a film's entry in the html
    % one each page we get 50 of these wlb_wrappers for movies plus one extra
at the end of the page
    yi=0;
```

```
    for yy=[1:50 52:101] % skip the end-of-page wlb_wrapper
        yi=yi+1;
        first=indxs(yy);
        last=indxs(yy+1);
        toParse=allscrape(first:last); % substring we will parse for the film
info
        % the title lays in the beginning, right between a </span> and the
next <span>
        tinds=strfind(toParse,'span');
        temp=toParse(tinds(1)+5:tinds(2)-2); % remove the spans
        titl=strtrim(temp);
        % rating is here
        rind=strfind(toParse,'Users rated this');
        rating=str2num(toParse(rind+16:rind+19));
        % the number of votes lies right after the rating a fixed number of
        % spaces because the format is always:
        % Users rated this X.Y/10 (ZZZ,ZZZ votes)
        vind1=rind+25;
        vind2=strfind(toParse(vind1:(vind1+20)),'votes')+vind1-3;
        votesStr=toParse(vind1:vind2); % this is the number in ZZZ,ZZZ format
        remove=strfind(votesStr,','); % get rid of the commas
        keep=setdiff(1:length(votesStr),remove);
        votes=str2num(votesStr(keep)); % numerical format
        allvotes(yi,xi)=votes;
        allratings(yi,xi)=rating;
        record.title=titl;
        record.rating=rating;
        record.votes=votes;
        record.year=xx;
        data{end+1}=record;
    end
end
```

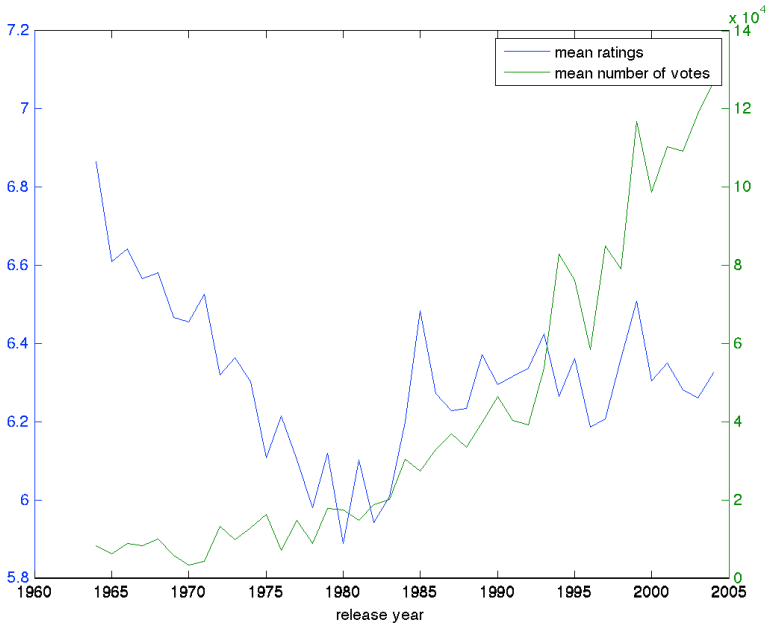
```
%% RESULTS
%
% first questions:  how do the distributions look for rating and votes?
figure; hist(allvotes(:));  title('Histogram of # votes');
xlabel('value'); ylabel('count');
figure; hist(allratings(:)); title('Histogram of ratings');
xlabel('value'); ylabel('count');
```





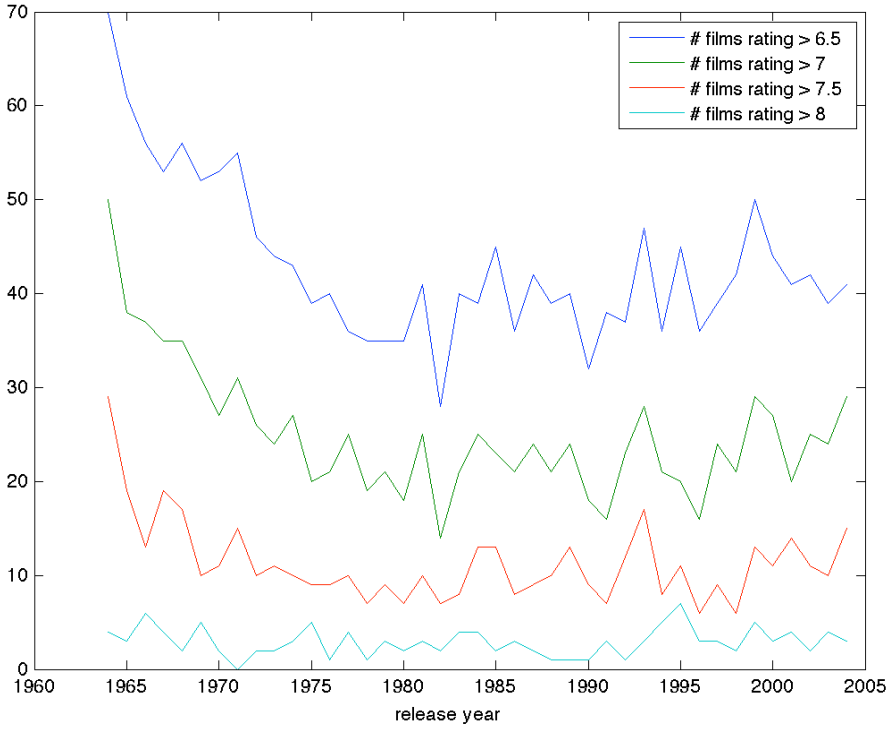

```
% Answer: ratings seem close to normally distributed, # votes is
% nowhere near... very exponential-ish.  I've looked at the distribution of
% votes in individual years as well, and its pretty much always like that,
% every year, as well as across all years.  Even worse, there is a disturbing
% non-sataionarity in the votes data:
```

```
figure; plotyy(1964:2004,mean(allratings),1964:2004,mean(allvotes));
legend('mean ratings','mean number of votes'); xlabel('release year')
```




```
% The mean number of votes increases year on year! I'm guessing this is
% because more people are discovering IMDB every year, and they vote on the
% movies they have seen that year.  This means that there is no easy
threshold
% criteria to define a classic by # of votes.
% This is terrible because I'd hoped votes would be the way to
% quantify this.  It's clear that people's ratings of movies can be very
% multi-modal: true fans love Star Trek movies, everyone else finds them
% mostly blah. I figured that lots of votes would indicate that people
% cared about a movie
%
% Interestingly, the mean ratings go up in the past, even as the number of
% votes drops tremendously.  Only classic film buffs and truefans vote that
% far back?

% let's look at ratings... to give you an idea of how IMDB ratings look
% here's some 1986 films that get the following ratings
% 5-6: 9 1/2 weeks, The Golden Child, Maximum Overdrive
% 6-7: Top Gun, Pretty in Pink, Short Circuit
% 7-8: Ferris Bueller, Blue Velvet, Transformers: The Movie
% 8+:  Aliens, Platoon, Stand by Me
%
% In other words high ratings probably don't correlate much with high brow,
% which suggests that depending on what your tastes are, IMDB
% ratings may not be good predictors of an iconic/classic movie
%
```

```
figure; plot(1964:2004,sum(allratings>6.5),1964:2004,sum(allratings>7),
1964:2004,sum(allratings>7.5),1964:2004,sum(allratings>8));
legend('# films rating > 6.5','# films rating > 7','# films rating > 7.5','#
films rating > 8')
xlabel('release year')
```

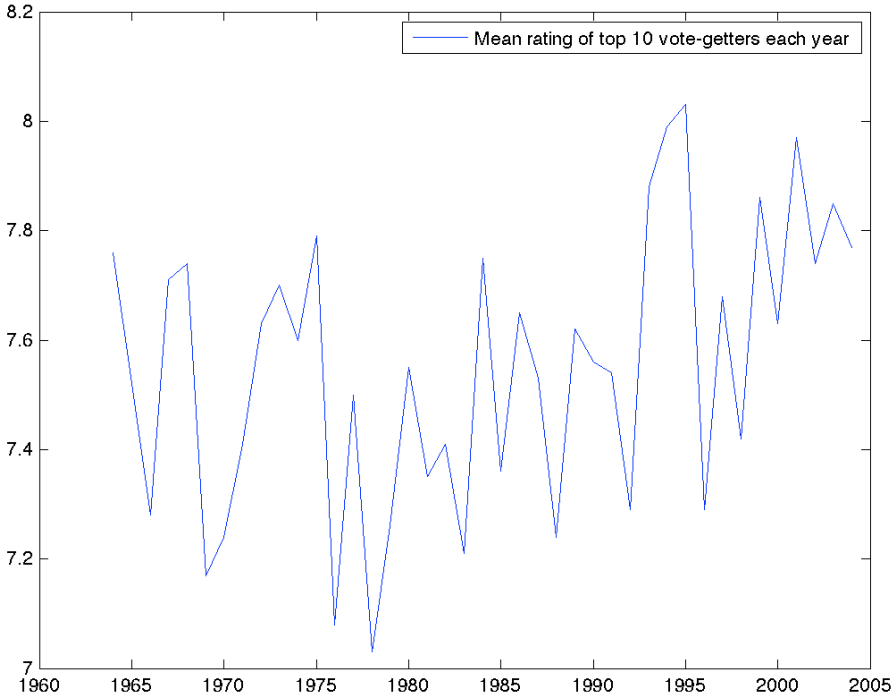



```
% Looking at thresholded ratings, we can see that a "hardline" stance on
% defining a classic (>8) puts it down to quite a constant (and low) level of
% achievement across years.  Using a lesser threshold results in peaks and
% valleys from year to year, and big trends such as observed previously
% where the dim past gets "grade inflation"
%
% this view of user ratings suggests two things:
% 1. It's less that 84-86 were good, and more that the early 80s were
terrible
% 2. There's a very interesting bump in 1993 (mostly 7.5-8 movies) and
another
% bump in 1995 (>8 movies).
%
% Here's all 7.5+ movies in 1993:


minds=find(allratings(:,1993-1964+1)>7.5);
ms=data(((1993-1964)*100+1):((1994-1964)*100));
ms{minds}
```

```
%      title: 'Jurassic Park'
%      votes: 462302
%     rating: 8
%       year: 1993
%
%
%      title: 'The Fugitive'
%      votes: 192146
%     rating: 7.8000
%       year: 1993
%
%
%      title: 'Schindler&#x27;s List'
%      votes: 715139
%     rating: 8.9000
%       year: 1993
%
%
%      title: 'Philadelphia'
%      votes: 160708
%     rating: 7.7000
%       year: 1993
%
%
%      title: 'The Nightmare Before Christmas'
%      votes: 196302
%     rating: 8
%       year: 1993
%
%
%      title: 'Groundhog Day'
%      votes: 362506
%     rating: 8.1000
%       year: 1993
%
%
%      title: 'Tombstone'
%      votes: 84615
%     rating: 7.8000
%       year: 1993
%
%
%      title: 'Falling Down'
%      votes: 122604
%     rating: 7.6000
%       year: 1993
%
%
%      title: 'The Piano'
%      votes: 58157
%     rating: 7.6000
%       year: 1993
%
%
```

```
%      title: 'Carlito&#x27;s Way'
%      votes: 147449
%     rating: 7.9000
%       year: 1993
%
%
%      title: 'The Joy Luck Club'
%      votes: 11908
%     rating: 7.6000
%       year: 1993
%
%
%      title: 'The Sandlot'
%      votes: 49868
%     rating: 7.8000
%       year: 1993
%
%
%      title: 'In the Name of the Father'
%      votes: 89483
%     rating: 8.1000
%       year: 1993
%
%
%      title: 'The Remains of the Day'
%      votes: 41066
%     rating: 7.9000
%       year: 1993
%
%
%      title: 'A Bronx Tale'
%      votes: 87700
%     rating: 7.8000
%       year: 1993
%
%
%      title: 'Iron Monkey'
%      votes: 12177
%     rating: 7.6000
%       year: 1993
%
%
%      title: 'True Romance'
%      votes: 144833
%     rating: 8
%       year: 1993
%

%
% Whew!! Still with me?
%
% It looks like my theory is pretty wrong, but then
% again, what good is a theory if you don't go to bat for it?  I'll make
% one more argument that I hope you'll find attractive.  Again we fall back
```

```
% on the questions: how can we extract "classic-ness" out of this data set?
% How can we seperate Groundhog Day (clearly a classic!) from The Sandlot
% (which as good a movie as it is, just doesn't meet my personal standards.
% Most importantly how can we remove Iron Monkey-like results from the
% above list? It might be a good kung-fu flick, but nobody in this country
% saw it in spite of Tarrantino's backing, and it isn't a classic IMHO.
%
% I will argue that we need a metric that takes into account these things:
% 1) Ratings
% 2) Votes (get rid of low-vote Iron Monkey fanboy noise in the ratings)
% 3) the upward trend of vote #s with year.
%
% Viola... we will use the mean rating of the most-voted-on movies in each
% year.  I did some complicated stuff to account for year-to-year variability
% in how many movies lived out there in the long-tail of the vote
% distribution: taking only the 90th percentile+ voted movies, or using 1.5 *
% inter-quartile range.  But it turned out that just using the top 10 vote-
% getters produced an essentially-identical graph:

for xx=1:length(years), [dummy ginds]=sort(allvotes(:,xx),'descend');
ts(xx)=mean(allratings(ginds(1:10),xx)); end
figure; plot(1964:2004,ts); legend('Mean rating of top 10 vote-getters each
year');

% Well crap.  The mid 80s are better than the early 80s, but still nothing
% on 1993.
```



```
%
% In conclusion, I couldn't figure out how to use this data to show what I
% know to be true: the 1980s are superior movie years. No matter how they
% were massaged, the analyses continue to point to the superiority of the
% mid 90s over the mid 80s in producing "classic/iconic" movies. These
% results are clearly counter to ground truth, and thus I conclude that
% IMDB's DB must have been corrupted by hackers.  Thanks Obama.
```