

Forecasting Gross Revenues At the Movie Box Office

Andrew Chen
June 4, 2002
ECON 482, Econometric Methods
Final Paper

Professor Richard Startz

Executive Summary

This paper examines the relationship between the film's total gross revenues and properties such as genre, presence of top dramatic talent, MPAA rating, and release date, and others. Using this model, we aim to answer several questions. Specifically, what role does each characteristic play in determining the total box office gross of a movie? For example, because genres and ratings map to specific demographic groups, movies targeted towards certain audiences may consistently do better or worse on average than others. Also we can explore the phenomenon of whether or not the \$20 million contracts actors and actresses get are actually worth it – is “star power” significantly and positively correlated with the revenues of a firm? How about with directors? How about a previously successful distribution company such as 20th Century Fox – does their marketing and business functions significantly boost a film's income? And finally, we aim to be able to predict the total box office grosses of films before they are released.

It should be noted that all the characteristics that are considered in the model are available about movies before their release, from websites such as the Internet Movie Database (www.imdb.com), The Numbers (www.the-numbers.com), Box Office Mojo (www.boxofficemojo.com), and all of these sources will be used in the prediction of unreleased films.

After building the model and examining data, we can note that most of the intuitive conclusions are indeed correct. Having actors and directors that have achieved financial success in previous movies is a critical factor in the success of the current one. Releasing during the weekends or the summer is also a positive indicator. Genres, on the other hand, are not great predictors of success; although dramas generally do quite well, the other categories are not as statistically significant. And finally, although the model is able to predict movie grosses reasonably accurately, the range of predicted values can be quite large, especially for movies with many stars and mass appeal.

Background & Data Collection

With the arrival of blockbusters such as Titanic (\$1.8B worldwide gross), Harry Potter (\$954MM), and Star Wars: Phantom Menace (\$925MM), one may begin to think of movies more as businesses than as works of art. Although it is true that filmmaking is an artistic skill, and there are many subjective factors leading to a film's ultimate success or failure, from the perspective of an economist, the analysis of art is pretty much infeasible. Unlike the numbers for a movie's box office gross, art is completely subjective, and it cannot be compared, measured, and approximated, and of course, without that we could not have an econometric analysis. Thus, instead of examining the art, we will examine the statistics behind moviemaking, and see how various properties of a film contribute to its total gross revenues.

To build this model, first we must find the aforementioned movie data. These statistics are actually quite easy to find on the Internet, as there are many tens of thousands of websites dedicated to

movies and the film industry. In particular, the Internet Movie Database (www.imdb.com) and The Numbers (www.the-numbers.com) contain the information needed for this project. IMDB categorizes films and contains the most information on MPAA ratings, full cast lists, etc., while The Numbers focuses on revenue information, theater counts, etc. Because the necessary information had to be cross-referenced over two websites, it was necessary to write several screen-scrapers to aggregate the data (the code is contained in Appendix A), which was done in Perl (which has great text parsing routines).

The fields that were collected by the program were: Date Released, Gross Revenues, Distribution Company, Actors, Director, Genre (16 categories), MPAA Rating, Country of Origin, and Theater Count. There were over 700 data points collected, but it was necessary to pick a more appropriate subset of the points as well as to convert the fields to something more usable in the model.

Description of Model

The first field collected was the release date of the film, which seems like it should be an important factor, intuitively. We are interested whether or not the film was released at a time when more people go to theaters, such as holiday weekends or during the summer, because generally more people will go to movies during those times. The former is particularly important because the first weekend of a film's release is generally a strong determinant of the total gross, because it is usually the peak attendance. Thus, a strong opening is necessary to make generate large revenues. Additionally, as it is clear in Appendix B, the drop off in revenues in subsequent weeks follows an oscillating exponential decay, so that most of the money is made in the first few weeks. Thus, we create two dummy variables measuring whether or not the movie was released up to two weeks before a three-day weekend or a longer holiday (Christmas season), and whether or not it was released during the summer. Summer is defined as the months of June, July, and August, and the holidays we are counting for the three day weekend are President's Day, Independence Day, Labor Day, Thanksgiving, and Christmas Season (December 18th-31st).

Secondly, we are interested in the talent behind the movie. Since so many actors are getting such high salaries (\$20MM or more, including potential profit sharing), it is interesting to see how important a star really is for a movie to be successful. The same can be said for the director of a movie. To a certain degree, both of these qualities may be able to reflect some underlying artistic qualities of the films, as certain directors and actors may be consistently good in their direction or performance, respectively. In this model, we simply cross reference the cast list of a film with whether or not the actor or director has been financially successful in the past. The variable for the director can either be true or false, while the actors variable counts the number of financially successful actors in each given film.

We are also interested in the target audience of a film. This can be measured in two ways, the genre of the film, as well as the MPAA rating. There were 16 genres collected, from the pages of IMDB. One important point to note is that the categories were not mutually exclusive, and that a film could belong to several categories. However, working with 16 genres is quite cumbersome and so I consolidated several and also threw out a few, leaving 9. Specifically, I dropped musicals and documentaries, and merged some of the genres to create 3 new ones. There were under 50 documentaries and musicals released in the 1998-2001 years we are examining, and they are not available in wide circulation and have different patterns than mainstream films. The 9 remaining genres become dummy variables within the equation for our model. Another demographics issue comes from what country a film originally comes from, and we convert the field of “country of origin” to a measure of whether or not a film is foreign or not. This is because substantial cultural and linguistic barriers keep mainstream audiences from appreciating them, and so foreign films may also target a narrow demographic. All of these operations left 625 data points.

Of course, there is also the practical business aspect to how well a movie does, reflected in the distribution company and the number of theaters it is initially released. One question that might be interesting is whether or not a distribution company that has shown previous financial success is conducive to future box office hits. Specifically, one might assume that the ability to create a movie that has made the top 20 grossing films indicates a skill for marketing or creating buzz that helps films become successful. Thus, the model includes a dummy variable measuring whether or not the distributor of the film has made a top 20 movie in the past. Along the same vein, the model also includes the number of theaters the movie is released to in the first few weeks. One peculiarity about this field is that many films differ in how they are distributed – many are released to a large number, while others (usually smaller budget films) are released to a smaller number that gradually builds up.

Finally, it is clear from a histogram of the grosses of past films that its revenues are exponential. Thus, we will want to take the log of the gross revenues when we are regressing.

Hypotheses

Examining the fields, one may have several intuitive propositions on which factors are most relevant. The presence of name-brand actors being a very important factor seems fairly obvious, as are summer and holiday releases. The gross revenues are probably very strongly related to the number of theaters it was released in. One might also suggest that of the genres, action adventure movies are the highest grossing, whereas science fiction probably appeals to a much narrower audience. One could also suggest that the R rating is the worst, while PG-13 or PG rating may be the best because it can appeal to both children and adults. It seems clear that movies that are sequels have large “built-in” audiences, and thus that should be positive. And finally, because of the cultural barriers described earlier, if a film is foreign, that will probably be a detriment to its total grosses.

Results

There are a few surprises from the regression (listed in Appendix C), but the coefficients returned from the ordinary least squares regression mostly matched our previously stated expectations. Before discussing the coefficients, it is important to note that, unfortunately, many of the terms are not statistically significant. At the 95% level, only the director, actors, drama genre, and the initial theater count are. At the 90% level, we can add whether or not the film was released on a holiday weekend and also whether or not the film was in the “thriller” category. Also, when we look at the different confidence intervals, most contain 0, so it is not clear if the different indicators are even positive or negative! However, the R^2 is 0.711919, which means that a fairly high amount of the variation is explained using this model.

First, it is confirmed at the 90% level that releasing it before a holiday interval is beneficial to the eventual gross of the film. However, whether or not a summer release helps is unclear, as the data points to the conclusion that a summer release is not statistically significant. The explanation for this might reside in the fact that the number of days contained around holiday weekends is much less than the three-month range of the summer season. Thus, there are simply more movies released during the summer interval, and with more movies, there is more variation in the revenues. Additionally, although a summer release may provide a benefit for every weekend of a movie’s release, it probably does not help as much in the first week where most of the money is made. Also, for the dates where significant numbers of audience members can be captured, many studios engage in counter-programming. In a weekend where a large, widely anticipated action movie is released, a rival studio may try to promote a romantic comedy to capture the segment outside of the action movie demographic. This counter-programming often fails (recently “Spiderman” and “Star Wars” soundly trounced “About a Boy”), which creates further smoothing the revenues.

Two of the strongest indicators of how well a movie would perform at the box office turned out to be if a top director or top actors were involved in the film. This was significant at the 95% level, and their confidence intervals contained only positive numbers. In fact, the data proposes that having a previously successful director is actually more important than having a previously successful actor. This may be because usually A-list directors have more influence on the overall artistic merit of their work, so that there is more consistency in quality. Either way, the idea that “star power” drives sales of movies, as common sense would suggest, is confirmed.

The genres were mostly statistically insignificant, as mentioned earlier. Although some of the values turned out to be positive, the confidence interval contained both negative and positive numbers, so that it is not clear whether or not many of them actually contribute or detract from the film’s success. However, at the 90% level, one can note that “thrillers” and “dramas” are statistically significant. Dramas are especially positive, with one of the largest positive coefficients as well as the highest t-statistic among the genres. This definitely makes a lot of sense – for example, if one

examines the top 20 list for movie grosses (Appendix D), there are no movies that fit squarely into comedy other than, perhaps, “Home Alone.” Thrillers, on the other hand, have a negative coefficient. This is possibly due to that genre appealing to much older audiences, limiting the potential market size. Other than those genres, “mainstream” genres such as “action adventure” and “family animation” had positive coefficients, while science fiction didn’t. One surprise was that comedy had a negative coefficient, although it was not statistically significant at the 90% or even 80% level.

As for the different ratings, it was clear that family movies (rated G) did well, as well as PG-13 movies, as I hypothesized. The most obvious explanation for these is that they are the ratings that are most likely to attract both kids and adults. For G, parents bring their small children, and for PG-13, both older kids and adults come to enjoy the movie independently. Both PG and R had similar (and lower) coefficients, as they may hit narrower demographics.

Sequels were fairly hit or miss, as were foreign. They both have very low t-statistics, and their confidence intervals are quite large. It is surprising that the foreign coefficient is even positive, but because in general, the only foreign films that are played in the US are generally the more successful ones produced overseas, they may eventually do well in our markets.

The final issue is that of the distributor and the theater count. Surprisingly, a distributor with previous financial success actually has a negative coefficient. However, it is not statistically significant, with a rather low t-statistic. However, the theater count is very important in this model, with a t-statistic of 22.99209, which makes it quite certain that the number of theaters and gross revenues are positively correlated.

Predictions & Conclusion

Finally, we can use this model to try and predict movie grosses, which may be particularly interesting for movies that have not been released yet. Some of the predictions for movies in past years are shown in Appendix E. Although the model does a fairly good job, there are also “sleeper” movies that continue to have strong revenues for a number of weeks, this phenomenon usually credited to good word-of-mouth. Additionally, this model does a poor job at predicting potential flops, significantly overshooting on star-studded vehicles that are released with poor buzz and poor reviews (for example Godzilla).

Two conclude, this model gives two predictions for movies coming out in the next two weeks, with the theater counts coming from Box Office Mojo:

Divine Secrets of the Ya-Ya Sisterhood (released 6/7/2002):

\$31.9MM (\$8.9MM, \$114.25MM)

Windtalkers (released 6/14/2002):

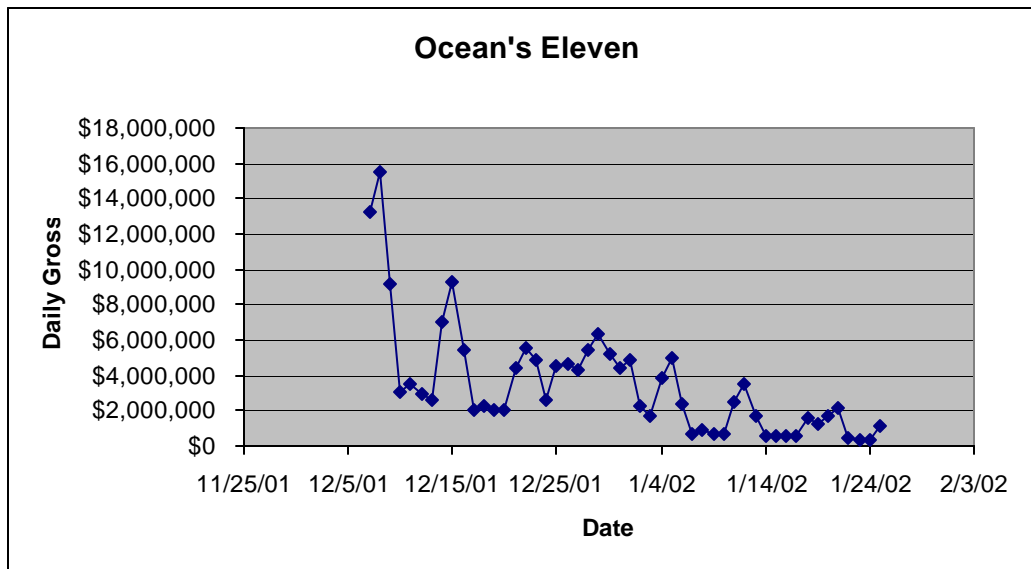
\$63.3MM (\$14.8MM, \$270.6MM)

Appendix A

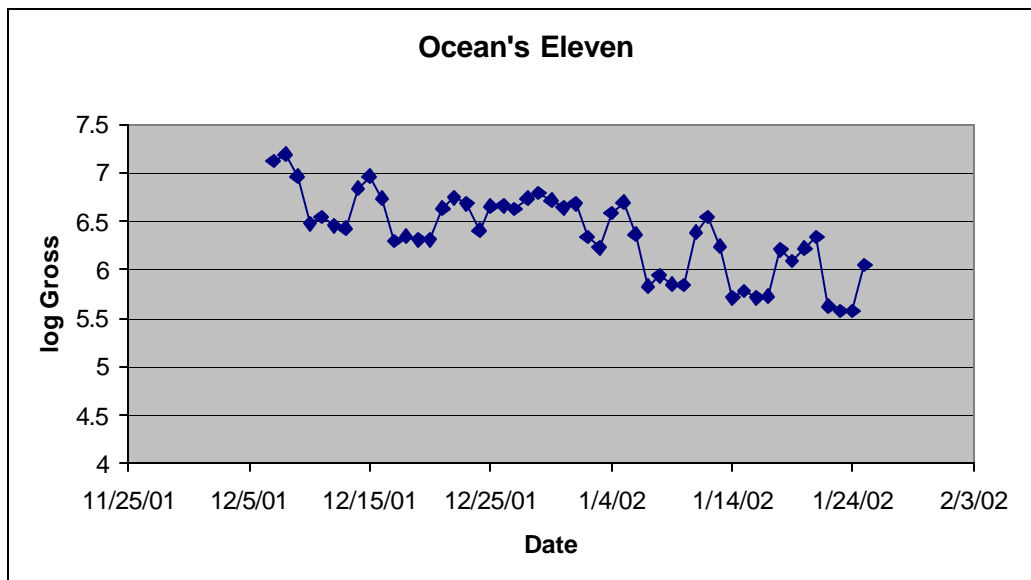
Please see Appendix F

Appendix B

As one can see, most films follow an oscillating exponential decay from the revenues of the opening day. The oscillations are caused the difference in attendance between weekends and weekdays. It's also clear from this particular graph that there's a slight bump when people go on Christmas vacation between 12/25 and 1/1.



If we take the log of the gross revenues, we see that it is a close to a line.



Appendix C

Dependent Variable: LNGROSS

Method: Least Squares

Date: 07/05/02 Time: 19:10

Sample: 1 625

Included observations: 606

Excluded observations: 19

| Variable | Coefficient | Std. Error | t-Statistic | Prob. |
|--------------------|-------------|-----------------------|-------------|--------|
| C | 14.57015 | 0.238728 | 61.03244 | 0.0000 |
| HOLIDAY | 0.131686 | 0.075292 | 1.749008 | 0.0808 |
| SUMMER | 0.106389 | 0.070741 | 1.503913 | 0.1331 |
| DISTRIB | -0.051498 | 0.070417 | -0.731322 | 0.4649 |
| DIRECTOR | 0.392302 | 0.097380 | 4.028560 | 0.0001 |
| ACTORS | 0.161654 | 0.033737 | 4.791598 | 0.0000 |
| ACT_ADVENT | 0.039506 | 0.060568 | 0.652254 | 0.5145 |
| FAMILY_ANIM | 0.038910 | 0.115100 | 0.338056 | 0.7354 |
| DRAMA_NEW | 0.176140 | 0.059317 | 2.969454 | 0.0031 |
| COMEDY | -0.081773 | 0.073186 | -1.117336 | 0.2643 |
| FANTASY | 0.081583 | 0.141857 | 0.575105 | 0.5654 |
| HORROR | 0.026897 | 0.123095 | 0.218508 | 0.8271 |
| ROMANCE | 0.100866 | 0.083683 | 1.205338 | 0.2286 |
| SCIFI | -0.115319 | 0.125977 | -0.915395 | 0.3604 |
| THRILLER | -0.154715 | 0.087117 | -1.775952 | 0.0763 |
| G | 0.345220 | 0.328319 | 1.051477 | 0.2935 |
| PG | 0.197165 | 0.256218 | 0.769519 | 0.4419 |
| PG13 | 0.244253 | 0.235033 | 1.039229 | 0.2991 |
| R | 0.206062 | 0.229799 | 0.896705 | 0.3702 |
| SEQUEL | 0.084155 | 0.154533 | 0.544578 | 0.5863 |
| FOREIGN | 0.019369 | 0.103043 | 0.187971 | 0.8510 |
| THEATER | 0.000984 | 4.28E-05 | 22.99209 | 0.0000 |
| R-squared | 0.711919 | Mean dependent var | 16.78800 | |
| Adjusted R-squared | 0.701560 | S.D. dependent var | 1.355649 | |
| S.E. of regression | 0.740586 | Akaike info criterion | 2.272880 | |
| Sum squared resid | 320.3055 | Schwarz criterion | 2.432865 | |
| Log likelihood | -666.6825 | F-statistic | 68.72426 | |
| Durbin-Watson stat | 1.796799 | Prob(F-statistic) | 0.000000 | |

Appendix D

All Time Top 20 Movies at the US Box Office

| | Released | Film Name | Total Box Office |
|----|----------|--|------------------|
| 1 | 1997 | <u>Titanic</u> | \$600,788,188 |
| 2 | 1977 | <u>Star Wars</u> | \$460,998,007 |
| 3 | 1982 | <u>ET: The Extra-Terrestrial</u> | \$431,197,000 |
| 4 | 1999 | <u>Star Wars: Phantom Menace</u> | \$431,088,297 |
| 5 | 1993 | <u>Jurassic Park</u> | \$357,067,947 |
| 6 | 2002 | <u>Spider-Man</u> | \$355,331,000 |
| 7 | 1994 | <u>Forrest Gump</u> | \$329,693,974 |
| 8 | 2001 | <u>Harry Potter and the Sorcerer's Stone</u> | \$317,557,891 |
| 9 | 1994 | <u>Lion King, The</u> | \$312,855,561 |
| 10 | 2001 | <u>Lord of the Rings: The Fellowship of the Ring</u> | \$311,053,126 |
| 11 | 1983 | <u>Return of the Jedi</u> | \$309,205,079 |
| 12 | 1996 | <u>Independence Day</u> | \$306,169,255 |
| 13 | 1999 | <u>Sixth Sense, The</u> | \$293,501,675 |
| 14 | 1980 | <u>The Empire Strikes Back</u> | \$290,271,960 |
| 15 | 1990 | <u>Home Alone</u> | \$285,761,243 |
| 16 | 2001 | <u>Shrek</u> | \$267,652,016 |
| 17 | 2000 | <u>How the Grinch Stole Christmas</u> | \$260,031,035 |
| 18 | 1975 | <u>Jaws</u> | \$260,000,000 |
| 19 | 2001 | <u>Monsters, Inc.</u> | \$254,252,781 |
| 20 | 1989 | <u>Batman</u> | \$251,188,924 |

Appendix E

Joe Somebody (released December 21, 2001)

| Actual | Predicted | Predicted (Low) | Predicted (High) |
|--------------|----------------|-----------------|------------------|
| \$22,770,864 | \$29,867,963.2 | \$6,469,451.966 | \$137,893,477.2 |

The model gives a fairly tight close answer for this particular movie.

Spy Kids (released March 30, 2001)

| Actual | Predicted | Predicted (Low) | Predicted (High) |
|---------------|-----------------|-----------------|------------------|
| \$112,692,062 | \$54,750,332.52 | \$12,944,774.82 | \$231,568,254.53 |

Bridget Jone's Diary (released April 13, 2001)

| Actual | Predicted | Predicted (Low) | Predicted (High) |
|--------------|-----------------|-----------------|------------------|
| \$71,500,556 | \$28,222,045.90 | \$6,615,078.22 | \$120,404,301.87 |

For both Spy Kids and Bridget Jone's Diary, the model underestimates by quite a bit. For Spy Kids, there was a lot of advertising and merchandizing tie-ins, as well as good reviews. The film did well enough to inspire a sequel. As for Bridget Jone's Diary, there was also a built in audience because of the book, which this model does not consider. Also, it had considerable buzz upon release in the US.

Hannibal (released February 9, 2001)

| Actual | Predicted | Predicted (Low) | Predicted (High) |
|---------------|------------------|-----------------|--------------------|
| \$165,091,986 | \$140,141,754.44 | \$19,095,838.89 | \$1,028,481,202.22 |

The model shows some of its weakness by forecasting a very large upper bound for the movie, at over \$1B domestic. It also gives a higher actual value because of 2 financially successful actors (Hopkins and Moore), as well as a top director (Ridley Scott).

Appendix F

Some of the code is buggy and much of the data in IMDB and The Numbers is not internally consistent. Running these programs still requires a significant amount of maintenance on the data by hand.

Mparse.pm – main parsing module

Cross.pm – some helper routines

Autorun.pl – automatically collect data

Tcount.pl – helper file to count # of theaters in initial release

Genres.pl – helper file to parse genres, to turn into dummy variables

Ratings.pl – helper file to parse ratings, to turn into dummy variables

Mparse.pm – main parsing module

```
use LWP::UserAgent;
use DBI;
use Date::Manip;
use cross;

#for ($j=1998; $j<=2001; $j++) {
#  print $j . "\n";
#  getList($j);
#}

#exit;

sub getList {

my $year = shift;

$ua = new LWP::UserAgent;
$ua->agent("AgentName/0.1 " . $ua->agent);

my $req = new HTTP::Request GET => 'http://www.the-
numbers.com/movies/index' . $year . '.html';

my $res = $ua->request($req);

my $pageData = $res->content;

#print "got my page\n";

$pageData =~ /TABLE/;

while ($pageData =~ /<TD VALIGN="TOP">(\w+,.*)<\/TD>/g)
{
  $date = $1;
  $date = $date . ", " . $year;
  #print $date . "\n";
}
```

```

$pageData =~ /$year\/(\w+)\Whtml/g;
$ticker = $1;

$pageData =~ /<B>(\w.*)<\/B>/g;
$title = $1;

$pageData =~ /<I>(.*?)<\/I>/g;
$distrib = $1;

$pageData =~ /\$(.*?)<\/TD>/g;
$gross = $1;

$date = ParseDate($date);
$date=UnixDate($date,"%m/%d/%Y");

$gross =~ s/\\,\\/g;

if ($gross>1000000) {

    $dir = getDirector($ticker,$year);
    $dirgross = &cross::checkDirector($dir);

my $famDir = 0;
if ($dirgross > 250000000) {
    $famDir = 1;
}

    my @actors = getActors($ticker,$year);

    my $famActor=0;
    for (my $i=0; $i<@actors; $i++) {
        # print "\t$actors[$i]\n";
        $numMovies = cross::checkActor($actors[$i]);
        if ($numMovies > 0) {
            $famActor++;
        }
    }
}

my $pgData = getIMDB($title,$year);

print "$date;$ticker;$distrib;$gross;$famDir;$famActor;";

# print out genres
my @genres = getGenres($title,$year,$pgData);

my $rating = getRating($title,$year,$pgData);

my $country = getCountry($title,$year, $pgData);

for ($i=0;$i<@genres;$i++) {
    print "$genres[$i]";
    print "~" if $i<@genres-2;
}

print ";$rating";

```

```

# foreign??
if ($country eq "USA") {
    print ";0";
} else {
    print ";1";
}

print "\n";

    # print "$date;$ticker;$distrib;$dir;$gross;$dirGross\n";
    #print "$ticker;$dir;$dirgross\n";

}

}
}

sub getDirector {
    my $ticker = shift;
    my $year = shift;

    $u = new LWP::UserAgent;
    $u->agent("AgentName/0.1 " . $ua->agent);

    my $r = new HTTP::Request GET => 'http://www.the-numbers.com/movies/'
    . $year . "/" . $ticker . ".html";
    my $resource = $u->request($r);
    my $pgData = $resource->content;

    if ($pgData =~ /Director/g) {
        $pgData =~ /html>(.*?)<\A>/g;
        $dir = $1;
    }

    return $dir;
}

sub getIMDB {
    my $title = shift;
    my $year = shift;
    my $depth = shift;

    #print "getIMDB called ($title,$year,$depth).\n";

    $u = new LWP::UserAgent;
    $u->agent("AgentName/0.1 " . $ua->agent);

    my $url = "http://us.imdb.com/Title?" . $title;

    # sometimes dates are already in titles
    # however, check that its not a unique id title ref
    unless ($title =~ /\(\d+\)/) {

        if ($year != "") {
            $url .= "+($year)";
        }
    }
}

```

```

}

# print "\n$url\n";
my $r = new HTTP::Request GET => $url;

my $resource = $u->request($r);
my $pgData = $resource->content;

if ($pgData =~ /Invalid title/g) {
    #print "Invalid Title\n";

    # oops dates from the-number and imdb are off!
    $title;
    $year = $year - 1;
    $pgData = getIMDB($title,$year,$depth+1) unless $depth > 1;

    # oops there are still problems! get user to help
    if ($depth>1) {
        print "\nHelp with $url:\n";
        my $newTitle = <STDIN>;
        my $newYear = <STDIN>;

        chomp($newTitle);
        chomp($newYear);

        # print "Trying with $newTitle, $newYear...\n";

        $pgData = getIMDB($newTitle,$newYear);
    }
}

# print $pgData;
return $pgData;
}

sub getRating {
    my $title = shift;
    my $year = shift;
    my $pgData = shift;

    #my $pgData = getIMDB($title,$year);

    $pgData =~ />USA:(.*?)</g;
    my $rating = $1;
    # print "heey $rating";
    #my @tmp = split / /, $rating;
    #$rating = $tmp[0];

    return $rating;
}

sub getCountry {
    my $title = shift;
    my $year = shift;
    my $pgData = shift;

```

```

#my $pgData = getIMDB($title,$year);

$pgData =~ /Country.*">(\w+)</g;
my $country = $1;

return $country;
}

sub getGenres {
my $title = shift;
my $year = shift;
my $pgData = shift;

#my $pgData = getIMDB($title,$year);

$pgData =~ /Genre(.*)/g;
$genreStr = $1;
#print $genreStr;

my @genres;
while ($genreStr =~ /">(\w+.\w+)</g) {
    @genres = (@genres, $1);
}

if ($genres[@genres] == "(more)") {
    $genres[@genres] = "";
}

return @genres;
}

sub getActors {
my $ticker = shift;
my $year = shift;
my @actors;

$u = new LWP::UserAgent;
$u->agent("AgentName/0.1 " . $ua->agent);

my $r = new HTTP::Request GET => 'http://www.the-numbers.com/movies/'
. $year . '/' . $ticker . '.html';
my $resource = $u->request($r);
my $pgData = $resource->content;

$pgData =~ /Cast</g;
while ($pgData =~ /><B>(.*\w+)</g) {
    my $actorName = $1;
    $actorName =~ s/<.*>//g;

    unless ($actorName =~ /\d+//) {
        #print "$ticker $actorName\n";
        @actors = (@actors, $actorName);
    }
}

return @actors;
}

```

```
return 1;
```

Cross.pm – some helper routines

```
package cross;

#&testDir;

sub checkActor {
    $name = shift;

    $ACTOR = "actorlist.txt";
    open ACTOR;
    while (<ACTOR>) {
        $line = $_;
        $line =~ /\d+\s+(\w+.*)\s+(\d+)/;
        $actor{$1}=$2;
        if ($name eq $1) {
            #print "inside: $2\n";
            return $2;
        }
    }
}

close ACTOR;

    return $actor{$name};
}

sub testDir {
    $name = "Robert Altman";
    $gross = checkDirector($name);
    print "$gross\n";
}

sub checkDirector {
    my $name = shift;

    $DIRECTOR = "directorlist.txt";
    open DIRECTOR;
    while (<DIRECTOR>) {
        $line = $_;
        $line =~ /(\w+.*)\s+\d+\s+\$(.*)/;
        $dirname = $1;
        $gross = $2;
        $gross =~ s/,//g;
        # $director{"$dirname"} = $gross;
        if($dirname eq $name) {
            return $gross;
        }
    }
}

close DIRECTOR;

    return $director{$name};
}
```

```
return 1;
```

Autorun.pl – automatically collect data

```
use mparse;  
  
for ($j=1998; $j<=2001; $j++) {  
  # print $j . "\n";  
  getList($j);  
}  
  
exit;
```

Tcount.pl – helper file to count # of theaters in initial release

```
use LWP::UserAgent;  
  
$ua = new LWP::UserAgent;  
$ua->agent("AgentName/0.1 " . $ua->agent);  
  
for ($j=1998;$j<2002;$j++) {  
  getList($j);  
  # print ($j);  
}  
  
exit;  
  
sub getList {  
  my $year = shift;  
  
  my $req = new HTTP::Request GET => 'http://www.the-  
numbers.com/movies/index' . $year . '.html';  
  my $res = $ua->request($req);  
  my $pageData = $res->content;  
  
  #print "got my page\n";  
  
  #print $pageData;  
  
  $pageData =~ /TABLE/;  
  
  while ($pageData =~ /<TD VALIGN="TOP">(\w+,.*)<\/TD>/g)  
  {  
    $date = $1;  
    $date = $date . ", " . $year;  
  
    $pageData =~ /$year\\(\\w+)\\Whtml/g;  
    $ticker = $1;  
  
    #print $ticker;  
  
    $pageData =~ /<B>(\w.*)<\/B>/g;  
    $title = $1;  
  
    $pageData =~ /<I>(.*?)<\/I>/g;
```

```

$distrib = $1;

$pageData =~ /(\$.*)</TD>/g;
$gross = $1;

#print "$date\t$title\t$distrib\t$gross\n";

getTheaterCount($ticker,$year);
}
}

exit;

sub getTheaterCount {
    my $ticker = shift;
    my $year = shift;

    $u = new LWP::UserAgent;
    $u->agent("AgentName/0.1 " . $ua->agent);

    my $r = new HTTP::Request GET => 'http://www.the-numbers.com/movies/'
    . $year . '/' . $ticker . '.html';

    my $resource = $u->request($r);
    my $pgData = $resource->content;

    $pgData =~ /Theaters/g;

    my $maxTCount = 0;
    for ($i=0;$i<4;$i++) {
        $pgData =~ /\$/g;
        $pgData =~ />(\d+.)</g;
        my $theaterCount = $1;
        $theaterCount =~ s/,//g;
        # print "Tcount $theaterCount\n";

        if($maxTCount < $theaterCount) {
            $maxTCount = $theaterCount;
        }
    }

    print "$ticker,$maxTCount\n";

    return $maxTCount;
}

```

Genres.pl – helper file to parse genres, to turn into dummy variables

```

$MOVIEDATA = "movielist-revised-new.csv";
open MOVIEDATA or die "Can't find it!";

while (<MOVIEDATA>) {
    $line = $_;
    @fields = split /,/, $line;

```

```

$genre_field = $fields[6];

@genres = split /\~/, $genre_field;

my @writeBuff;
for ($i=0;$i<@genres;$i++) {
    SWITCH: {
        if ($genres[$i] =~ /Action/) { $writeBuff[0] = 1; last SWITCH; }
        if ($genres[$i] =~ /Adventure/) { $writeBuff[1] = 1; last SWITCH; }
    }
    if ($genres[$i] =~ /Animation/) { $writeBuff[2] = 1; last SWITCH; }
}
    if ($genres[$i] =~ /Comedy/) { $writeBuff[3] = 1; last SWITCH; }
    if ($genres[$i] =~ /Crime/) { $writeBuff[4] = 1; last SWITCH; }
    if ($genres[$i] =~ /Documentary/) { $writeBuff[5] = 1; last
SWITCH; }
    if ($genres[$i] =~ /Drama/) { $writeBuff[6] = 1; last SWITCH; }
    if ($genres[$i] =~ /Family/) { $writeBuff[7] = 1; last SWITCH; }
    if ($genres[$i] =~ /Fantasy/) { $writeBuff[8] = 1; last SWITCH; }
    if ($genres[$i] =~ /Film-Noir/) { $writeBuff[9] = 1; last SWITCH; }
}
    if ($genres[$i] =~ /Horror/) { $writeBuff[10] = 1; last SWITCH; }
    if ($genres[$i] =~ /Musical/) { $writeBuff[11] = 1; last SWITCH; }
}
    if ($genres[$i] =~ /Mystery/) { $writeBuff[12] = 1; last SWITCH; }
}
    if ($genres[$i] =~ /Romance/) { $writeBuff[13] = 1; last SWITCH; }
}
    if ($genres[$i] =~ /Sci-Fi/) { $writeBuff[14] = 1; last SWITCH; }
    if ($genres[$i] =~ /Thriller/) { $writeBuff[15] = 1; last SWITCH; }
}
    if ($genres[$i] =~ /War/) { $writeBuff[16] = 1; last SWITCH; }
    if ($genres[$i] =~ /Western/) { $writeBuff[17] = 1; last SWITCH; }
}
}

}
$sticker = $fields[1] . ",";
print $sticker;

for ($j=0;$j<17;$j++) {
    if ($writeBuff[$j]) { print "1"; } else { print "0"; }

    if($j<17) {
        print ",";
    } else {
        print "\n";
    }
}

}
}

```

Ratings.pl – helper file to parse ratings, to turn into dummy variables

```
$MOVIEDATA = "movielist-revised-new.csv";
open MOVIEDATA or die "Can't find it!";

while (<MOVIEDATA>) {
    $line = $_;
    @fields = split /,/, $line;
    $rating_field = $fields[20];
    $sticker_field = $fields[1];

    # print $rating_field;

    my @writeBuff;
        if ($rating_field eq "G") { $writeBuff[0] = 1; }
        if ($rating_field eq "PG") { $writeBuff[1] = 1; }
        if ($rating_field eq "PG-13") { $writeBuff[2] = 1; }
        if ($rating_field eq "R") { $writeBuff[3] = 1; }
        if ($rating_field eq "NC-17") { $writeBuff[4] = 1; }

    print "$sticker_field,";

    for ($j=0;$j<5;$j++) {
        if ($writeBuff[$j]) {
            print "1";
        } else {
            print "0";
        }

        if ($j<4) {
            print ",";
        } else {
            print "\n";
        }
    }
}
}
```