STAT 209
Scraping Data from HTML Tables

March 28, 2018

Colin Reimer Dawson

Data from Web Pages

Sometimes you see data online not in the form of a file:

THANKSGIVING WEEKENDS
(1982-Present)

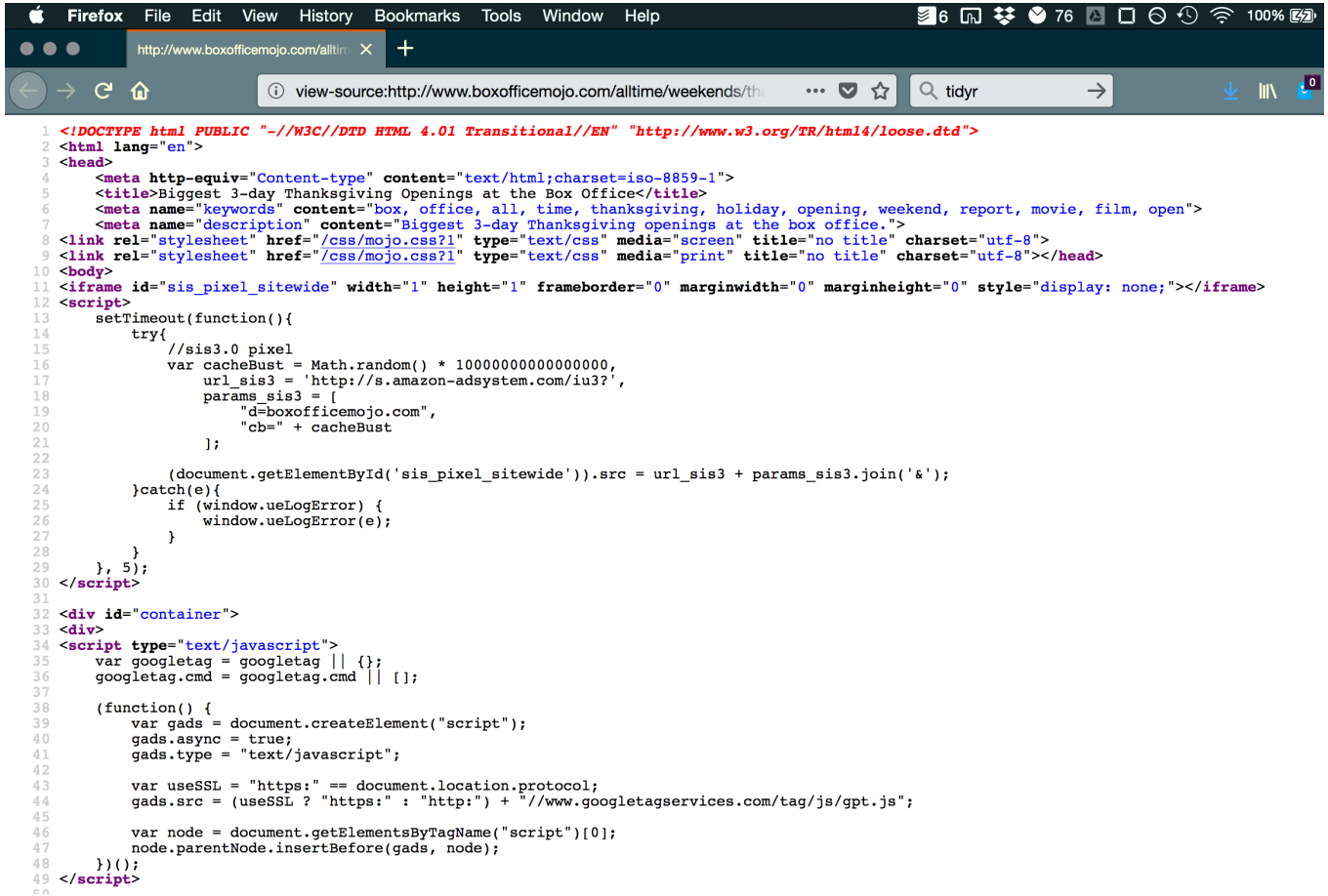
3-day Openings | 3-day All Movies | 5-day Openings | 5-day All Movies

Rank	Title (click to view)	Studio	**Opening***	% of Total	Theaters	Avg.	Total Gross^	Release Date%
1	**Frozen**	BV	**$67,391,326**	16.8%	3,742	$18,009	$400,738,009	11/27/13
2	**Toy Story 2**	BV	**$57,388,839**	23.3%	3,236	$17,734	$245,852,179	11/24/99
3	**Moana**	BV	**$56,631,401**	22.8%	3,875	$14,615	$248,757,044	11/23/16
4	**Coco**	BV	**$50,802,605**	24.3%	3,987	$12,742	$209,372,919	11/22/17
5	**Tangled**	BV	**$48,767,052**	24.3%	3,603	$13,535	$200,821,936	11/24/10
6	**The Good Dinosaur**	BV	**$39,155,217**	31.8%	3,749	$10,444	$123,087,120	11/25/15
7	**Enchanted**	BV	**$34,440,317**	26.9%	3,730	$9,233	$127,807,262	11/21/07
8	**101 Dalmatians (1996)**	BV	**$33,504,025**	24.6%	2,794	$11,991	$136,189,294	11/27/96
9	**A Bug's Life**	BV	**$33,258,052**	20.4%	2,686	$12,382	$162,798,565	11/25/98
10	**Four Christmases**	WB (NL)	**$31,069,826**	25.9%	3,310	$9,387	$120,146,040	11/26/08

Figure: Source: Box Office Mojo (link)

Messy Embedding

The data is embedded (somewhere) in the HTML code of the site




```
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3.org/TR/html4/loose.dtd">
<html lang="en">
<head>
    <meta http-equiv="Content-type" content="text/html;charset=iso-8859-1">
    <title>Biggest 3-day Thanksgiving Openings at the Box Office</title>
    <meta name="keywords" content="box, office, all, time, thanksgiving, holiday, opening, weekend, report, movie, film, open">
    <meta name="description" content="Biggest 3-day Thanksgiving openings at the box office.">
<link rel="stylesheet" href="/css/mojo.css?1" type="text/css" media="screen" title="no title" charset="utf-8">
<link rel="stylesheet" href="/css/mojo.css?1" type="text/css" media="print" title="no title" charset="utf-8"></head>
<body>
<iframe id="sis_pixel_sitewide" width="1" height="1" frameborder="0" marginwidth="0" marginheight="0" style="display: none;"></iframe>
<script>
    setTimeout(function(){
        try{
            //sis3.0 pixel
            var cacheBust = Math.random() * 10000000000000000,
                url_sis3 = 'http://s.amazon-adsystem.com/iu3?',
                params_sis3 = [
                    "d=boxofficemojo.com",
                    "cb=" + cacheBust
                ];

            (document.getElementById('sis_pixel_sitewide')).src = url_sis3 + params_sis3.join('&');
        }catch(e){
            if (window.ueLogError) {
                window.ueLogError(e);
            }
        }
    }, 5);
</script>

<div id="container">
<div>
<script type="text/javascript">
    var googletag = googletag || {};
    googletag.cmd = googletag.cmd || [];

    (function() {
        var gads = document.createElement("script");
        gads.async = true;
        gads.type = "text/javascript";

        var useSSL = "https:" == document.location.protocol;
        gads.src = (useSSL ? "https:" : "http:") + "//www.googletagservices.com/tag/js/gpt.js";

        var node = document.getElementsByTagName("script")[0];
        node.parentNode.insertBefore(gads, node);
    })();
</script>
```

Our Goals

Get the data from the web into R in a *reproducible* way. That means:

1. No manual editing of text
2. No copying and pasting
3. No fiddling with the data in Excel

In other words, we need to go from what's on the web to a data frame (and visualizations) using a *self-contained* script

More `tidyverse` Building Blocks