EDA

^ S01-intro.py code

# EDA

Rossmann is a German retailer with 1,115 stores in 12 German states. Our task is to forecast sales at Rossmann stores over the course of six weeks. Our dataset contains 942 days (2013-01-01 to 2015-07-31) and 1,115 stores, so we have 942*1115 = 1,050,330 observations (rows of data).

We'll dig in as we go, but for now, here are the most important columns:

- **store** and **date**, our index features.
- **sales**, our target variable.
- **customers**: Number of customers at that store on that date.
- **open**: Whether or not that store is open on that date.
- **promo**: Whether or not that store had a promo on that date.
- **promo2**: Whether or not that store takes part in a 'continuing and consecutive' promotion - maybe like a loyalty program. (We don't get much detail here, but we'll dig in more later.)
- **trend**: again not much detail here, but this appears to be the Google search trend for Rossmann in a particular state for the week. It's an integer between 28 and 100.
- We have other features covering holidays, store types, nearest competitor, and weather. We'll get to those later.

N.B. for notational convenience, we'll consider 'sales' to be denominated in dollars.

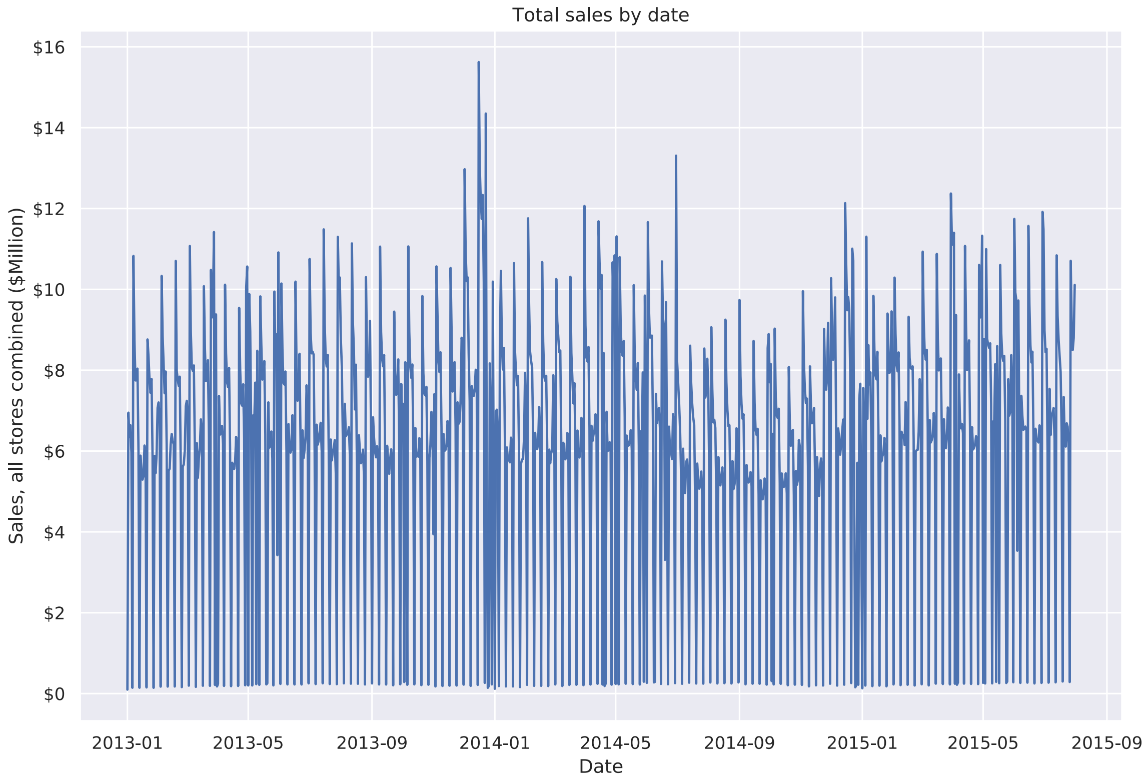
| | store | date | sales | customers | open | promo | promo2 | trend | school_holiday |
|---|---|---|---|---|---|---|---|---|---|
| 1 | 1 | 2013-01-01T00:00:00 | 0 | 0 | 0 | 0 | 0 | 61 | 1 |

| | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| 2 | 56 | 2013-01-01T00:00:00 | 0 | 0 | 0 | 0 | 1 | 61 | 1 |
| 3 | 69 | 2013-01-01T00:00:00 | 0 | 0 | 0 | 0 | 1 | 61 | 1 |
| 4 | 77 | 2013-01-01T00:00:00 | 0 | 0 | 0 | 0 | 1 | 61 | 1 |
| | 111 | 2013-01-01T00:00:00 | 0 | 0 | 0 | 0 | 1 | 61 | 1 |

^ S02-sales_chart.py code

## Overall Sales

Just as a baseline, here's the overall sales picture.

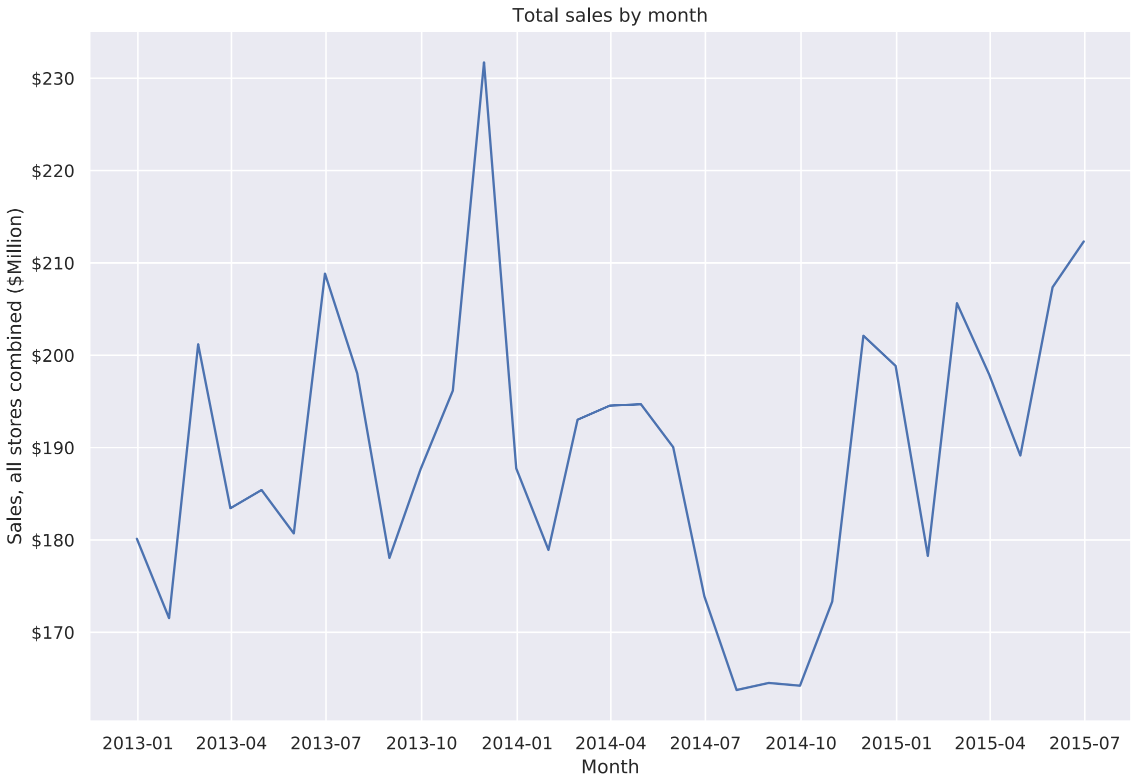


## Aggregated by Month

The previous chart is a little too granular. Are sales going up as we'd expect?

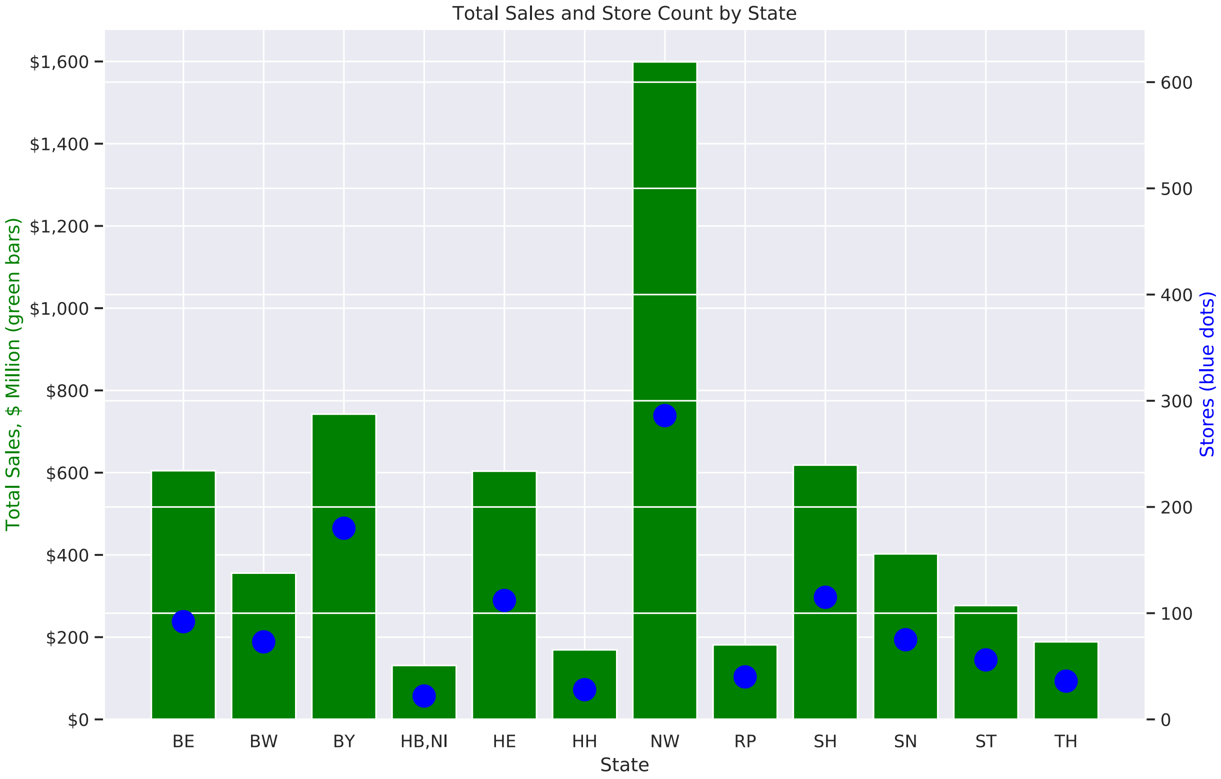
Below is the same chart, aggregated by month. A couple of interesting things jump out:

1. There was a huge year-end spike in December 2013 that we haven't seen replicated yet.
2. There was an enormous dip in 2014. As it turns out, a ton of stores were temporarily closed in 2014 (including all 180 stores in Bavaria, for a period of months). We'll get back to that later.
3. Slightly less obvious, but just as important: 2015 sales are back on track, with each month's sales being above 2013 sales. (The overall 2014 sales aren't a good point of comparison, due to item #2.)



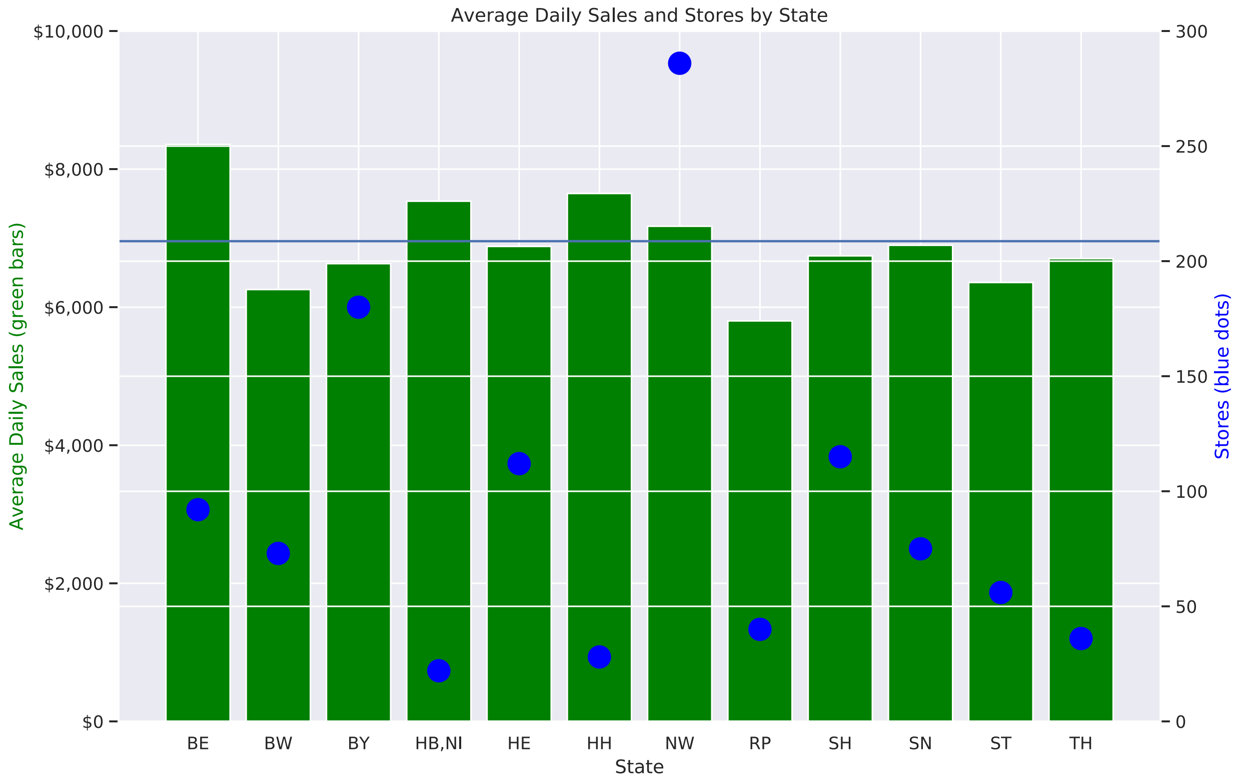
## Sales by State

Here we see total sales by state, over the entire period of the dataset. Clearly North Rhine-Westfalia leads in both revenue (green bars) and store count (blue dots), with Bavaria (BY), Schleswig-Holstein (SH), Berlin (BE), and Hesse (HE) being major players.



## Average Daily Sales by State

In average daily sales per store by state, it's a slightly different story. Berlin (BE) is the clear leader, with Hamburg (HH), Lower Saxony/Bremen (HB,NI), and North Rhine-Westfalia leading in average daily sales per store.



^ S03-correlations.py code

## Correlation - features vs sales

Let's get a quick handle on which variables correlate most strongly with sales. (See the section below the chart for a brief explanation of correlation if you're not familiar with it.)

A few takeaways from the below:

- **customers** has the strongest correlation with sales, at 0.9. **customers** can be viewed as both an input and an output: strong pricing, promotions, merchandising, and assortment will not only bring in more customers, but will also drive higher revenue per customer.
- We can see that **promo** has a correlation with sales of 0.50, and **promo2** and **promo2_interval** have correlations with sales around -0.10.
- **trend** has a correlation with sales of 0.13. Like **customers**, this variable could be viewed (though to a lesser degree) as both an input and an output.
- We don't have great data on pricing, product, or merchandising that help to explain our sales numbers. Specifically, we have **assortment** (0.08 correlation with sales) and **store_type** (-0.02 correlation with sales).
- While **competition_distance** and **competition_open_since_year** seem like they should help us to evaluate the strength of our store locations in some fashion, they have correlations of around 0.01.

The rest of the variables are essentially out of control of the business (weather, state holidays, etc.).

| | Feature | |
|---|---|---|
| 1 | sales | 1 |
| 2 | customers | 0.900 |
| 3 | open | 0.704 |

| | | |
|---|---|---|
| 4 | promo | 0.503 |
| 5 | day_of_week | -0.439 |
| 6 | state_holiday | -0.214 |
| 7 | trend | 0.132 |
| 8 | promo2 | -0.119 |
| 9 | promo2_interval | -0.103 |
| 10 | school_holiday | 0.103 |
| 11 | max_visibility_km | 0.083 |
| 12 | assortment | 0.081 |
| 13 | mean_wind_speed_km_h | 0.061 |
| 14 | max_wind_speed_km_h | 0.051 |
| 15 | mean_visibility_km | 0.051 |
| 16 | min_humidity | -0.049 |
| 17 | max_humidity | -0.047 |
| 18 | mean_humidity | -0.047 |
| 19 | min_visibility_km | 0.042 |

The Pearson correlation measures whether two variables have a linear relationship:

- A correlation of 1 means a perfect positive linear relationship.
- A correlation of -1 means a perfect negative linear relationship.
- A correlation of 0 means no linear relationship at all.
- While two variables may have a non-linear relationship, the Pearson correlation is a common 'eyeball' statistic to help us get our bearings.

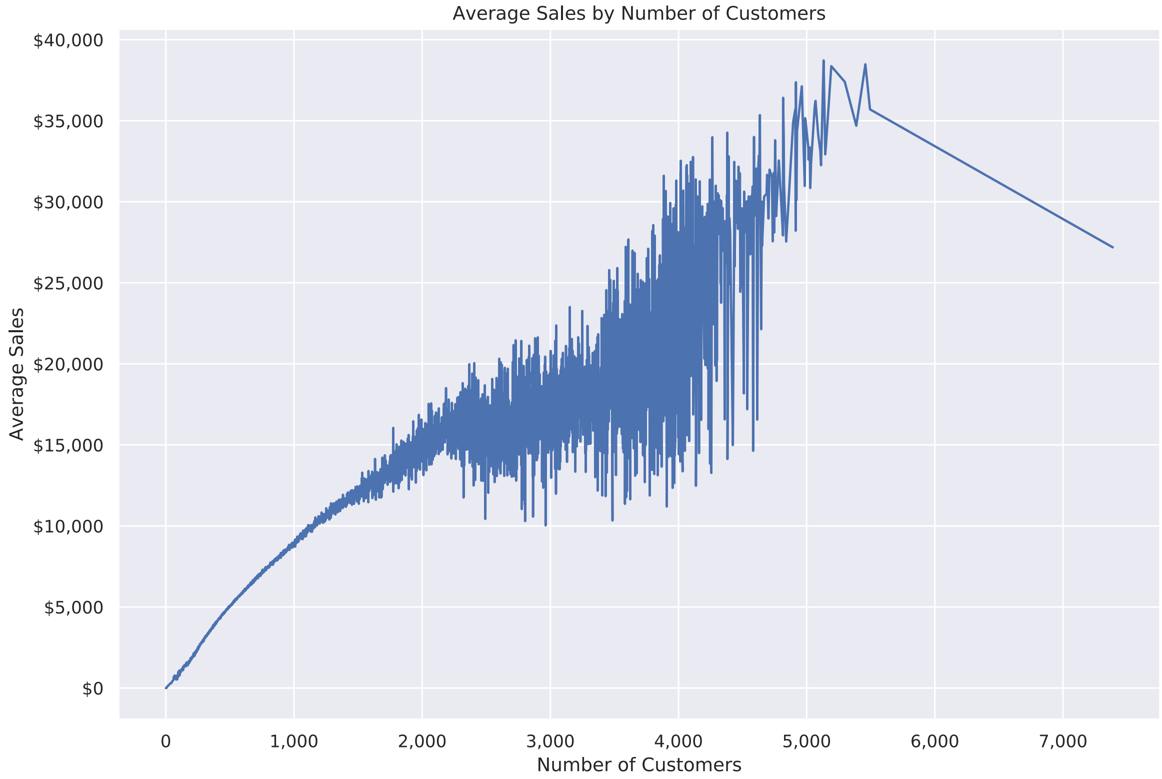
^ S04-customers.py code

## Sales vs Customers

As noted above, **customers** has a correlation of 0.90 with **sales**. It's pretty obvious on the chart below; the more customers, the more sales. Note also that as we bring in more customers, the relationship gets less strong, until it starts to break down around 5,000 customers in a given store (clearly only a few stores could even fit 5,000 customers in a day).

We don't know the specific definition of 'customer' in this case, or how they're counted. Is it someone who bought, or just someone who came into the store? Do internet visitors/buyers count? In any case, we'll want to work with the marketing team to bring more people through the doors (virtual and physical).

For now, since the correlation with sales is so strong, and since our neural network model will manage the relationship between customers and sales implicitly for us, let's continue to focus on **sales** and keep **customers** as a secondary focus.

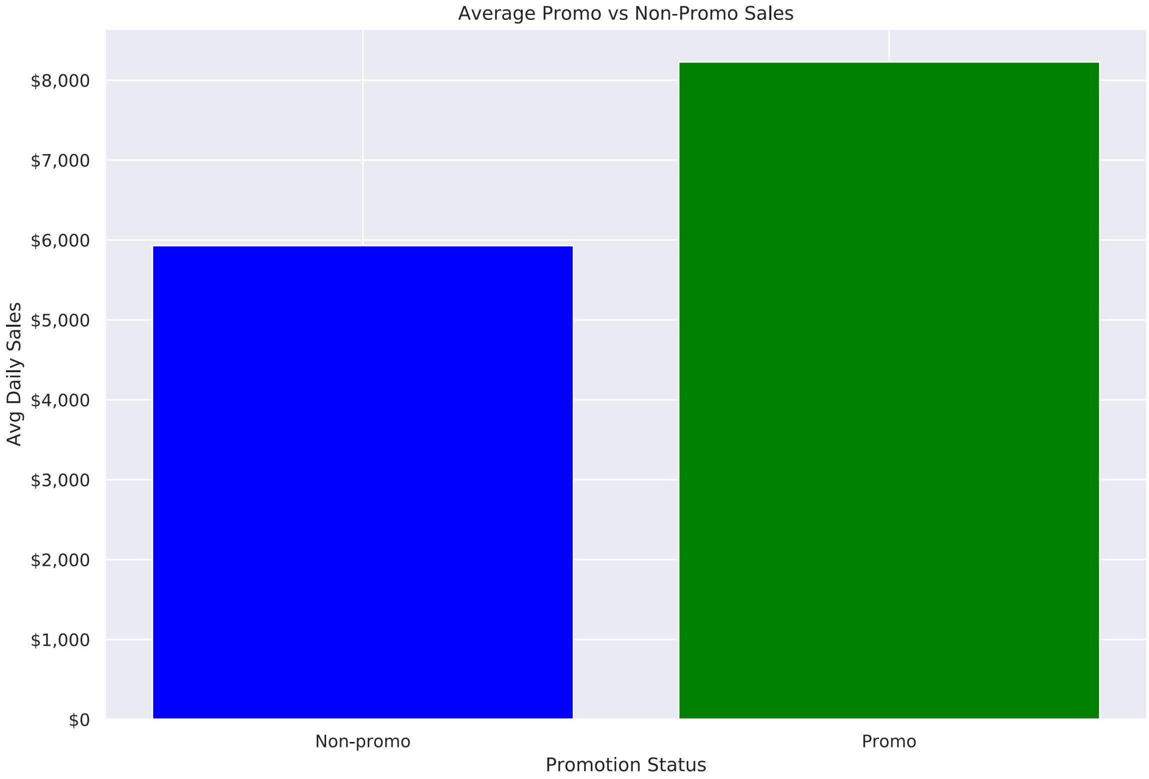


^ S05-promo1.py ⏱ code ⌄

## Sales vs Promo

So let's look at **promo** vs **sales**; **promo** is one of the few things that are squarely under the business's control in this dataset.

Overall, sales at a store with a promotion are about $2,000 (33%) higher than sales at a store without a promotion.



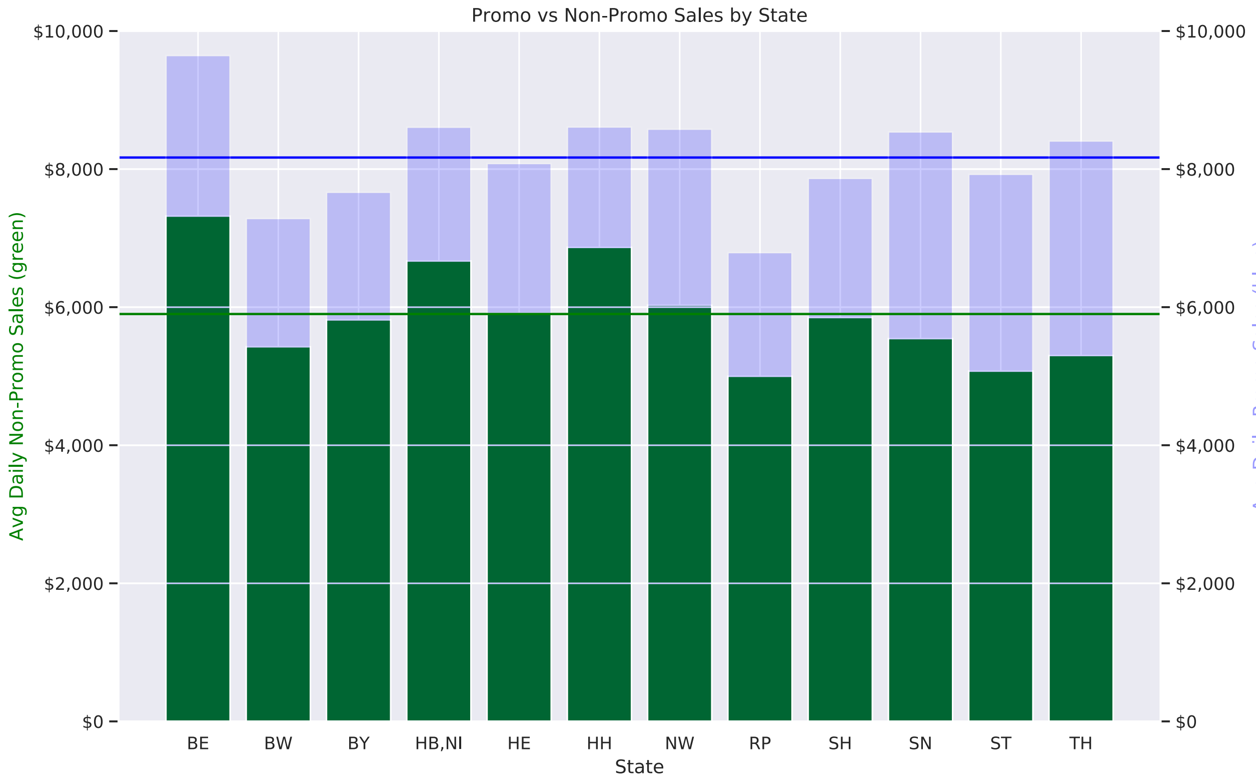
## Sales vs Promo by State

This incremental $2,000 for a promo appears roughly to hold up by state. Below we're comparing non-promoted sales (in green) to promoted sales (in blue); the colored horizontal lines indicate overall averages. As we can see, promotions are very effective in each state, with some slight differences.

In particular, the three states at the right of the chart below (SN, ST, and TH) have below-average daily non-promoted sales (in green); their promoted sales (in blue) are **more than $2,000 above** their average non-promoted sales.

On the other hand, two of our top-performing non-promoted states ('HB,NI' and HH) have average promoted sales that are **not quite** $2,000 more than their non-promoted sales.

Our best-performing state (BE for Berlin, the capital of Germany) does best in both non-promoted and promoted sales. This state does so well that it even **out-performs the $2,000 average uplift** for a promotion.



^ S06-promo2.py

code

## Impact of Multiple Promos

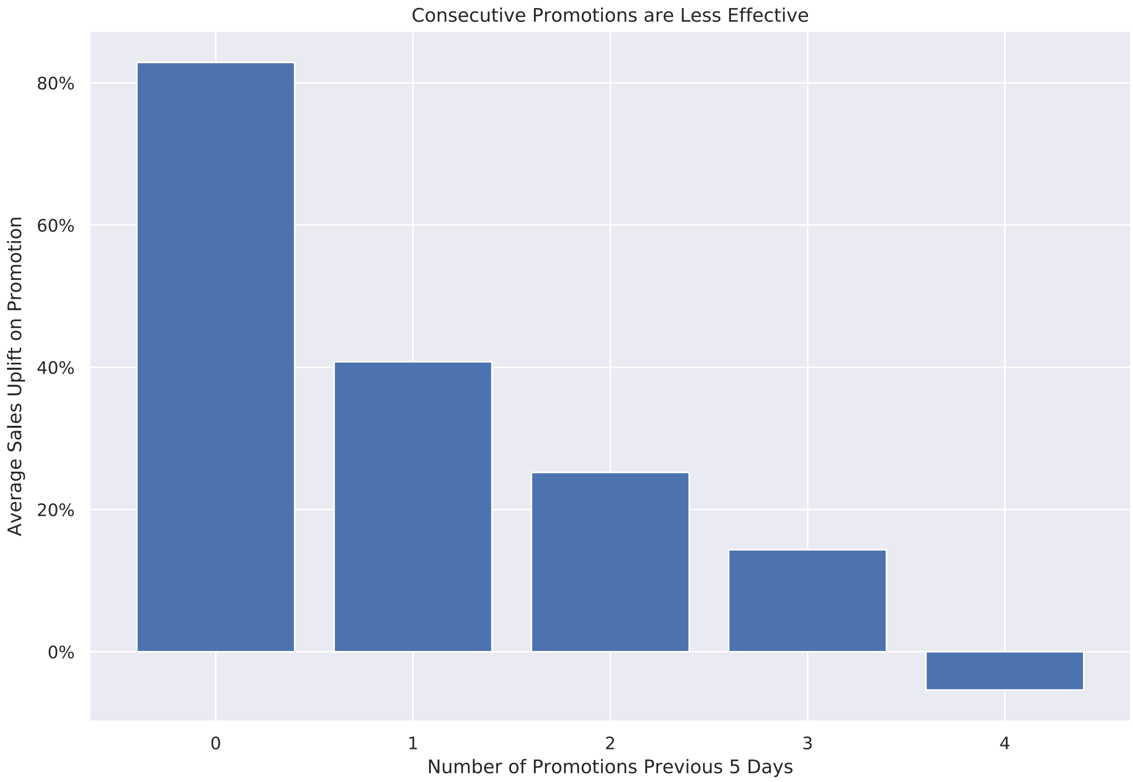
So if we can reliably get an **extra $2,000 in revenue** per store by running a promotion, can we just run more promotions? It depends.

Below we see the uplift of a promo over the last 5 days' sales, broken down by the number of promotions in the previous 5 days.

If there haven't been any promotions in the last 5 days, we can expect on average an 80% uplift; but **each extra promotion cuts our uplift in half**.

At the extreme right, we see that a promo on a 5th consecutive day has a negative uplift (against the previous promoted days' sales).

So, we see diminishing returns as we promote more intensely.

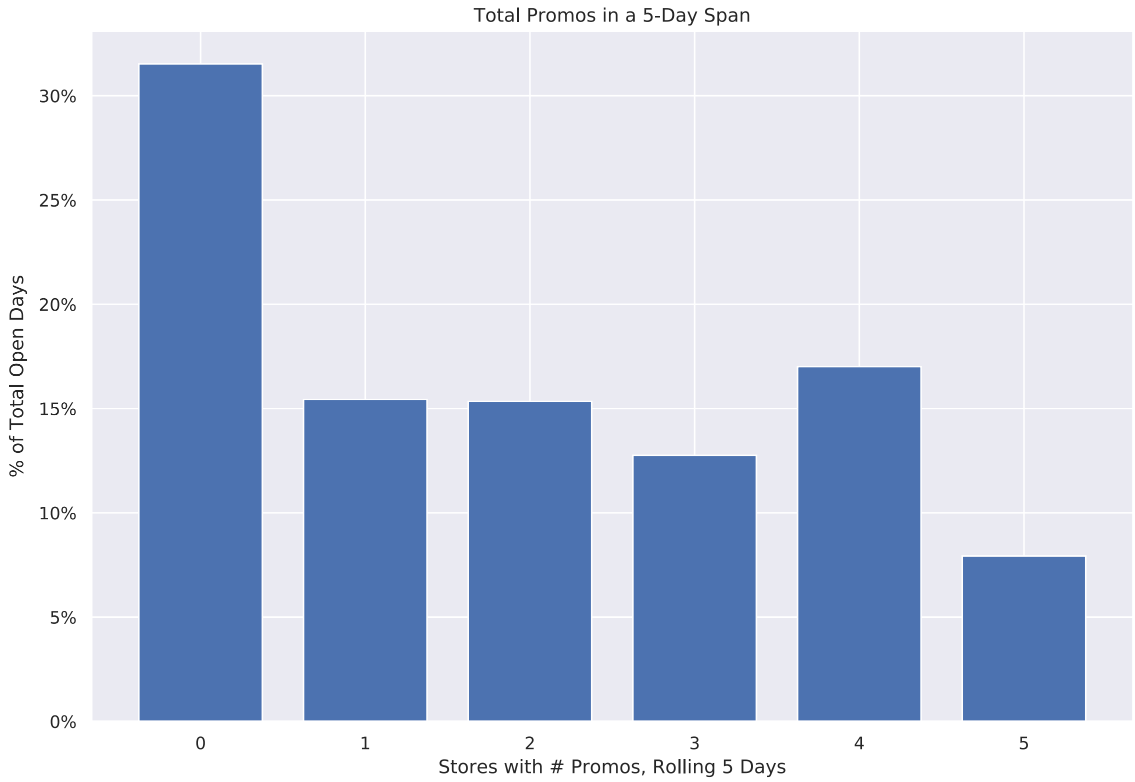


## Opportunities to Promote More

But are there opportunities to fit promotions in appropriately, without killing our current sales and profitability?

Maybe. Below we see that over 30% of store-days have no promotions in the previous 5 days.

So it's possible that we might be able to slot in some more promotions, with some finer analysis of the financials of promotions. This would require additional detail and additional cost/benefit analysis (data that we don't currently have).



S07-promo_ANOVA.py ⏱ code

## Statistical testing of promo uplifts (technical)

Just to be sure, let's check that the difference in effectiveness of promos we see above is real. We'll use a one-way ANOVA test with p=0.05; in other words, if $p < 0.05$, we can conclude with 95% confidence that the difference in uplifts is not a result of random chance.

```
One-way ANOVA test for difference of means:
	0.8287
	0.4078
	0.2521
	0.1434
	-0.0538
F-val: 208193.37374349358, p-val: 0.0
```

Okay, so we can confidently claim that some of the means are different. But we don't know which ones, so let's run a t-test to check whether the two closest are *really* different, to 95% confidence. The two closest are for 2 and 3 days above, with mean uplifts of 25.21% and 14.34% respectively.

Again we'll use a p-value of p=0.05, meaning that if $p < 0.05$, we can conclude with 95% confidence that the difference is real.

```
Two-sided T-test for means:
        0.2521
        0.1434
T-val: 151.63466737789415, p-val: 0.0
```

Pretty convincing.

While we're here, let's just double-check that the difference in overall promoted vs non-promoted sales -- the one that got us going down this path in the first place -- is real as well.

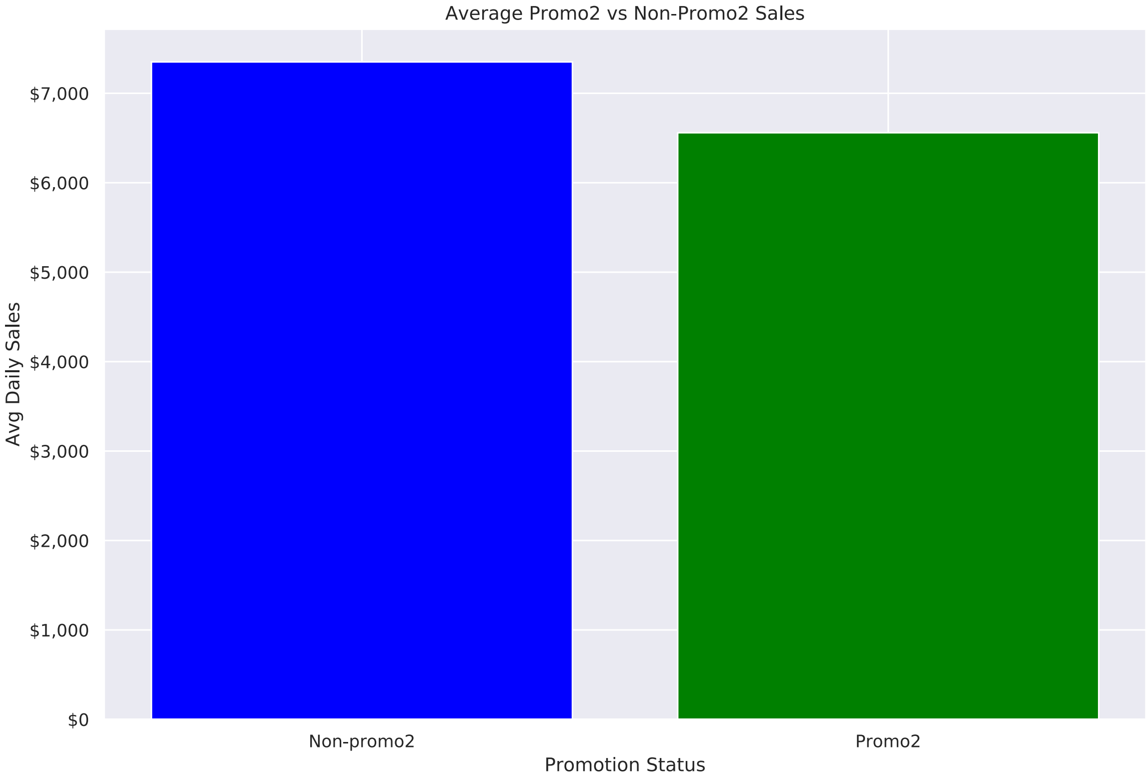
```
Two-sided T-test for means:
        $8,228.70
        $5,929.83
T-val: 363.88536245828976, p-val: 0.0
```

Yup, it's real.

S08-promo2.py code

## Sales vs Promo2

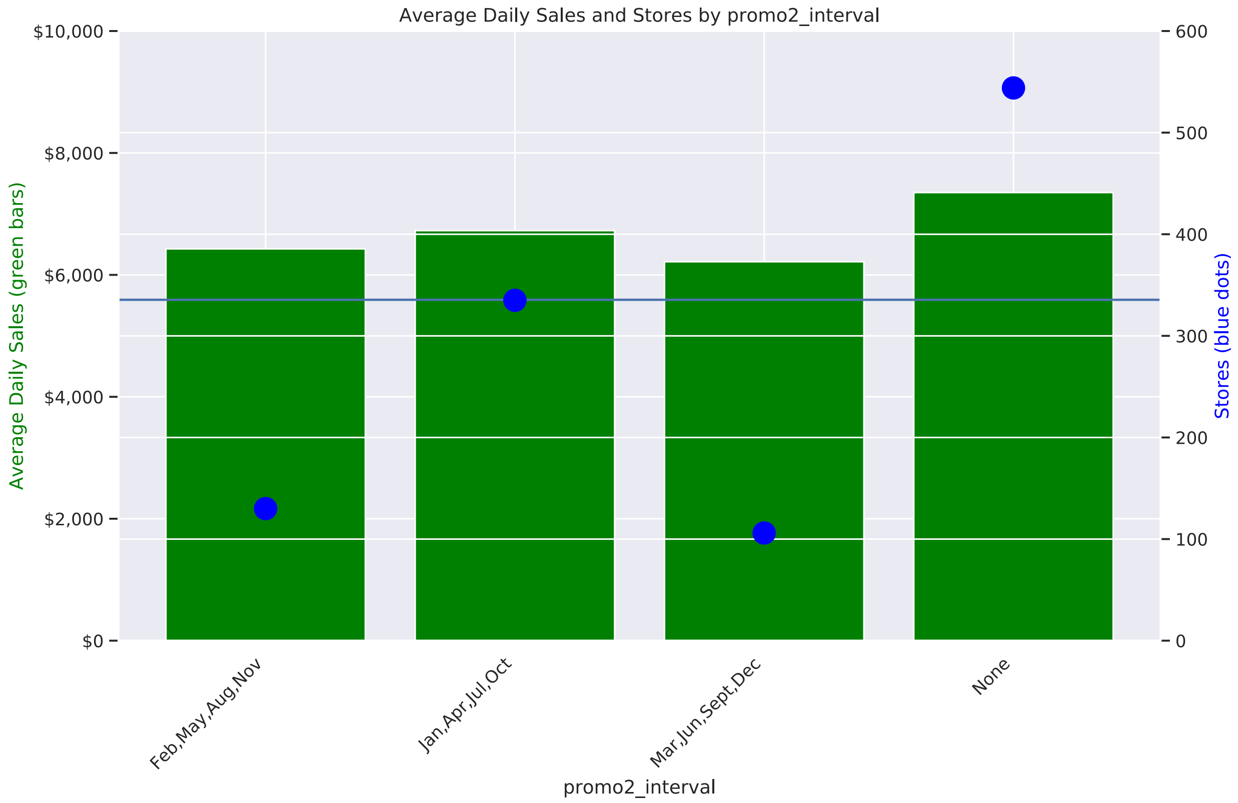
So what about **promo2**? Below we see **promo2** sales vs. non-**promo2** sales. Looks like **promo2** sales are *lower* than non-**promo2** sales.



## Sales vs promo2_interval

Let's look by promo2_interval to get a little more detail. Again, **promo2** is a "continuing and consecutive" promotion for some stores. **promo2_interval** indicates which months the **promo2** is refreshed in each quarter.

Again, below we see that non-**promo2** daily sales are higher than **promo2** stores, regardless of when the **promo2** is refreshed. We also see (the blue dots) that about 550 of the 1,115 stores do not run **promo2**.

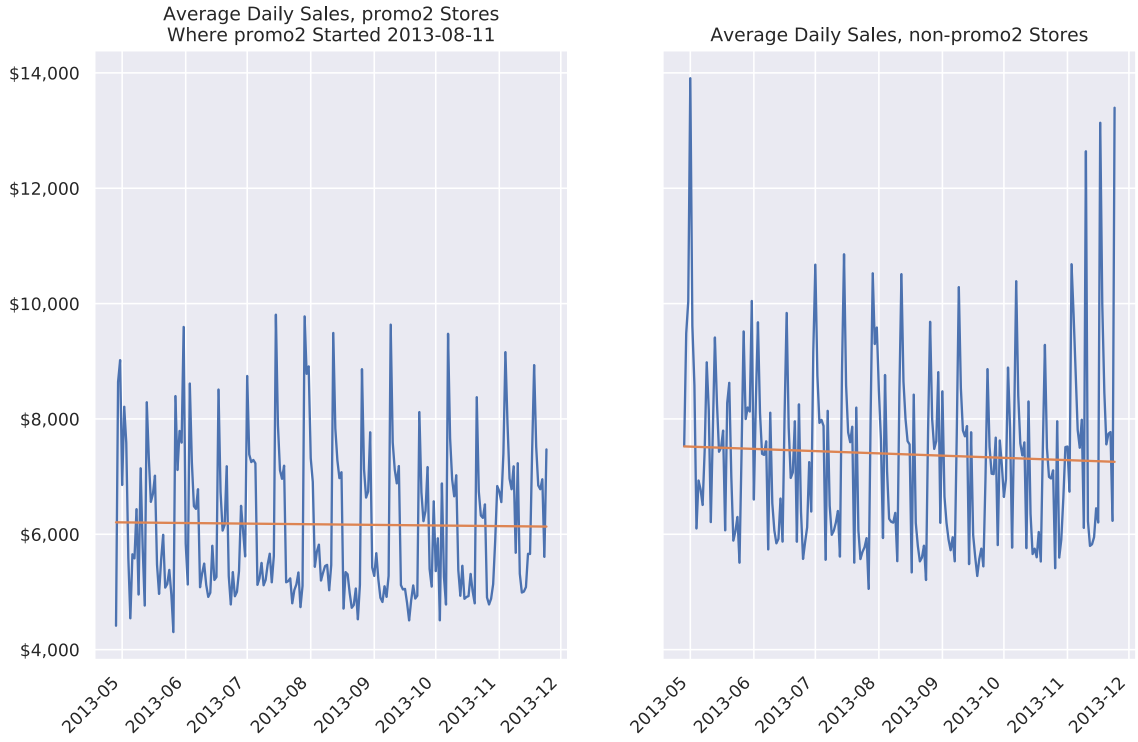


## promo2 before and after

Why would we run **promo2** if daily average sales are *lower* where we run it? We have some examples of stores that started running **promo2** after January 2013 (the beginning of our data), so we can compare those stores' sales before and after they started **promo2**, to the stores that did not run **promo2** at all.

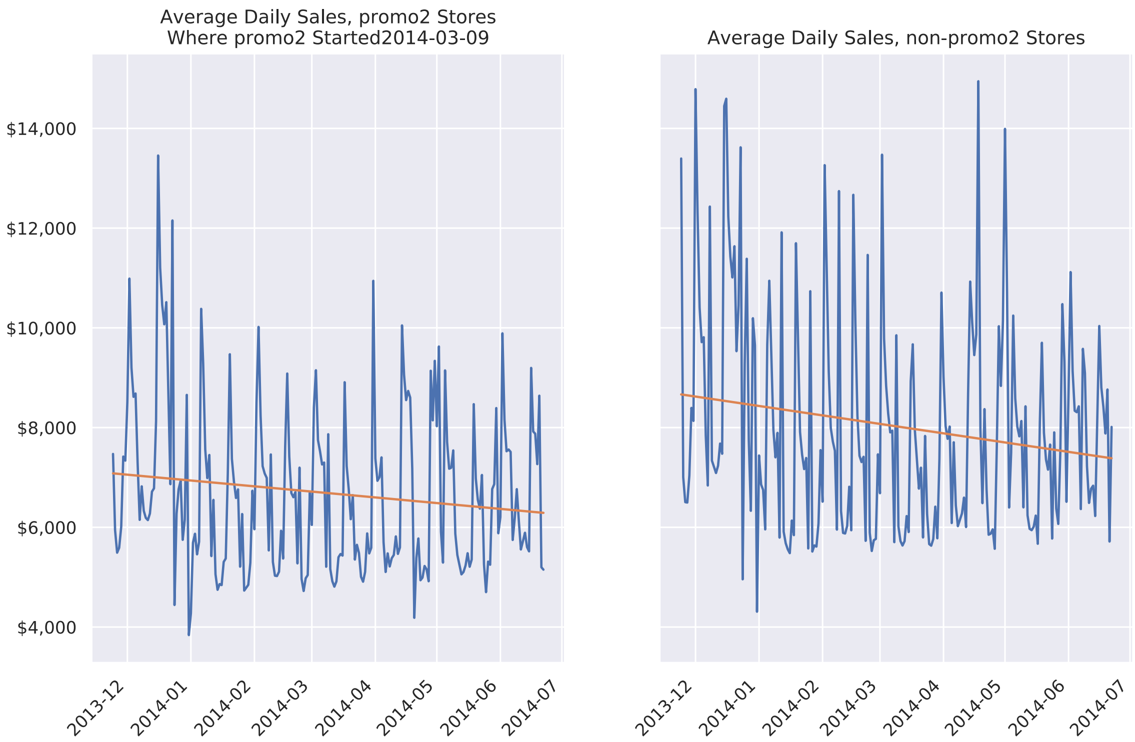
First, there were 32 stores that started **promo2** the week of August 11, 2013. Below we see that both groups of stores have sales declines over the time period, but the **promo2** stores have better performance than the non-**promo2** stores. In case it's hard to see visually, beneath the chart are the slopes of the regression lines over the time period, along with the results of a t-test of the difference in the regression lines.

The t-test results have a lot of detail, but here we're concerned about the section labeled 'P>|t|'. If this is less than 0.05, we can say with 95% confidence that the regression slopes are different - in other words, that the promo2 stores have a sales uplift over this timeframe.



```
Two-sided T-test for means:
        -0.3496
        -1.2659
                             Test for Constraints
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
c0            -0.9208      1.227     -0.750      0.454      -3.340       1.499
==============================================================================
```

Similarly, there were 29 stores that started **promo2** the week of March 9, 2014. Below we see that while all these stores saw seasonal sales declines over the time period, the **promo2** stores saw a smaller dip in sales than the non-**promo2** stores.



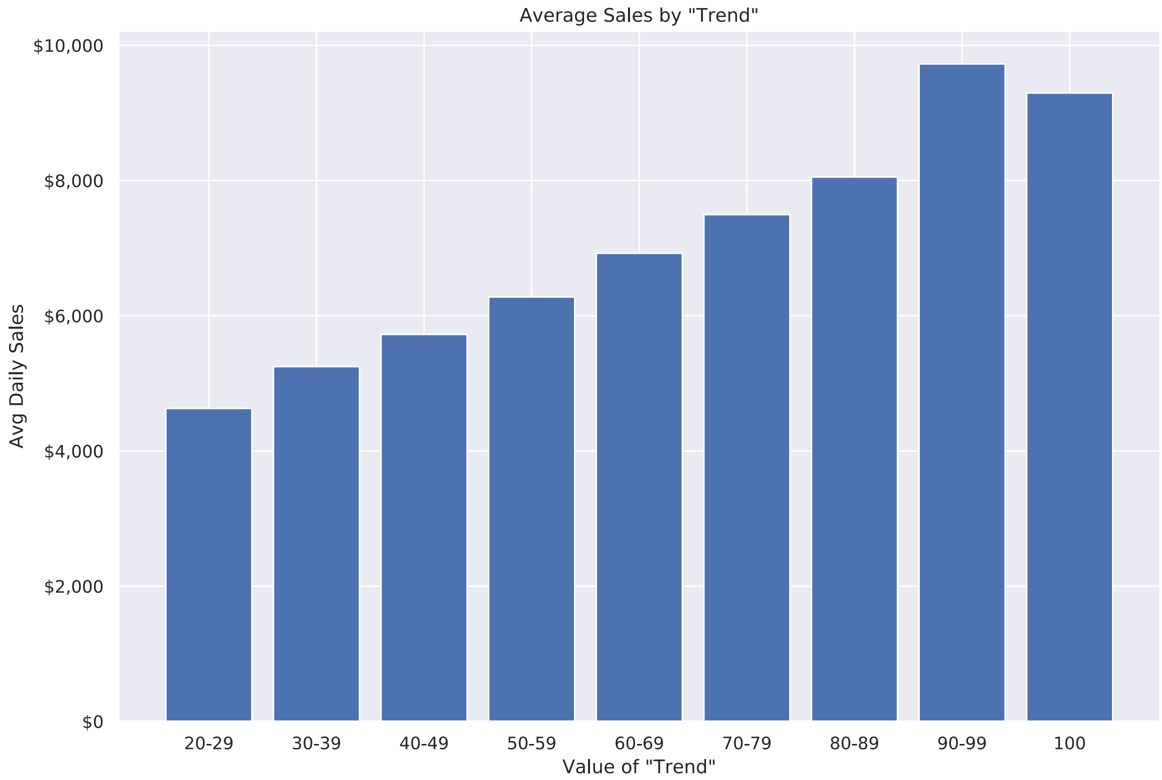
```
Two-sided T-test for means:
        -3.7526
        -6.0667
                             Test for Constraints
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
c0            -2.3251      1.511     -1.539      0.125      -5.303       0.653
==============================================================================
```

Our p-values are greater than 0.05 in both cases. So although we can say that directionally our promo2 stores appear to be doing better, the analysis just run does not provide compelling evidence that promo2 is better than doing nothing. We'll add promo2 to possible "future directions for research" and, since we have many more variables to review, turn our focus back to the EDA for sales.

S09-trend.py code

## Sales vs Trend

Below we see average sales by **trend**. Generally, with increasing **trend** we get increasing sales.

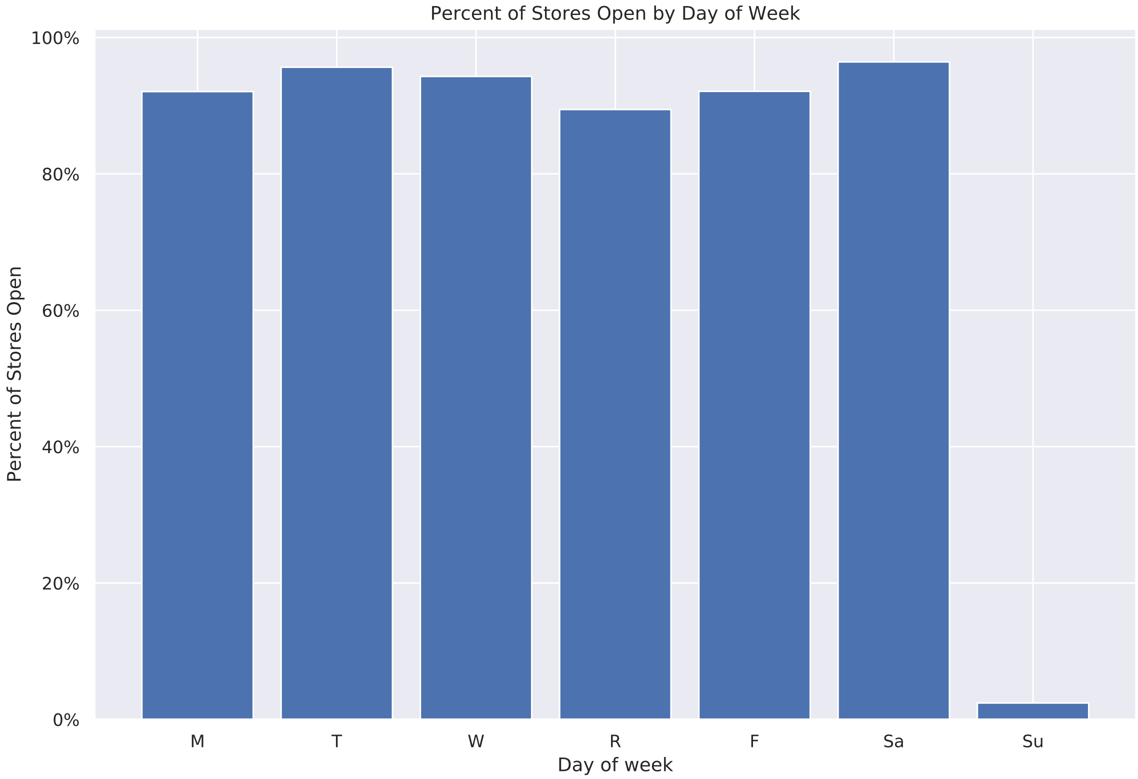


^ S10-day_of_week.py

code

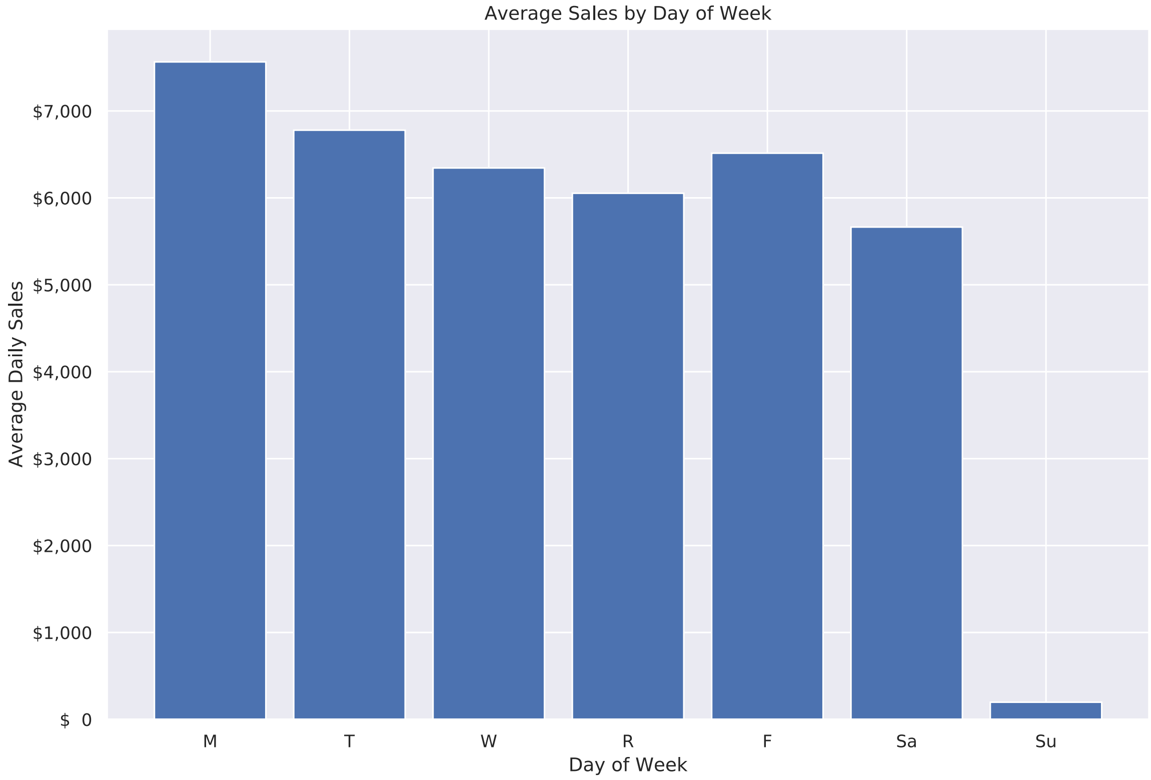
## Day of Week

Day of week has a correlation with sales of -0.44. What's going on? In Germany, many stores are closed on Sunday.



## Day of Week

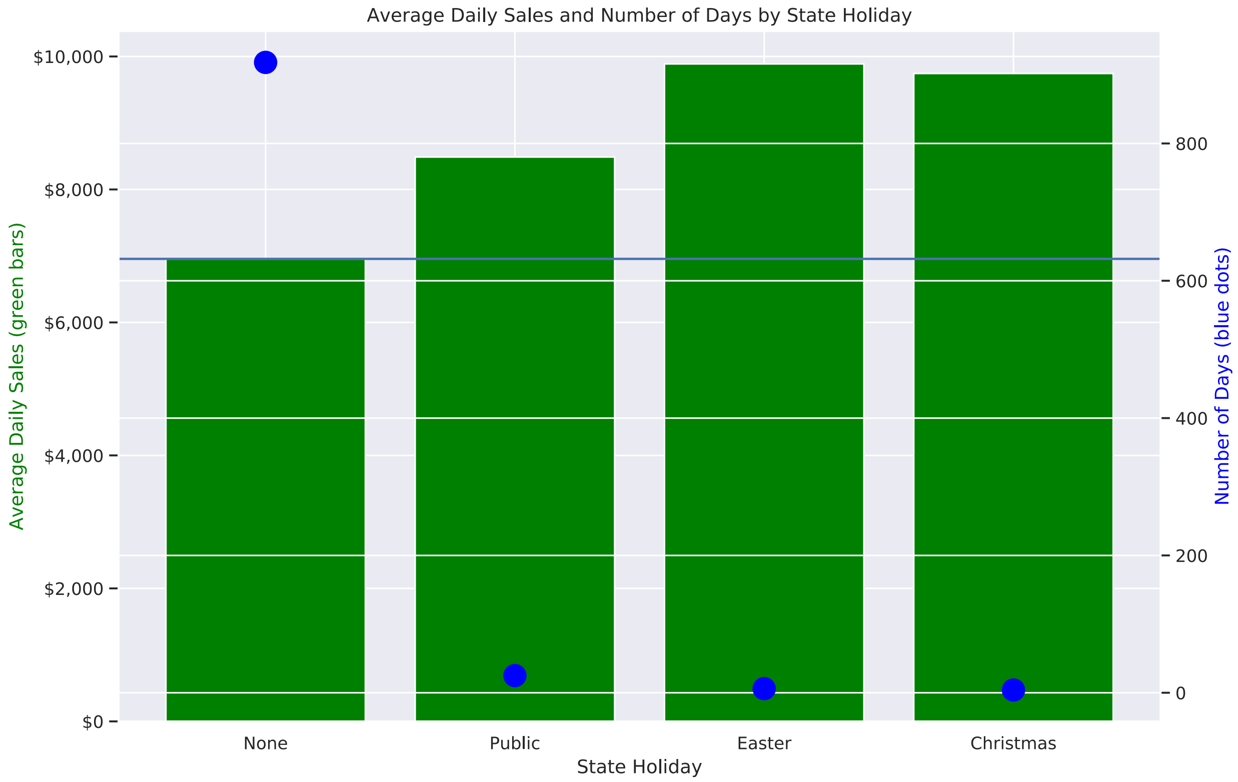
As a result, sales are highest on Mondays, falling steadily throughout the average week before a slight bump on Fridays.



^ S11-state_holiday.py code

## State Holiday

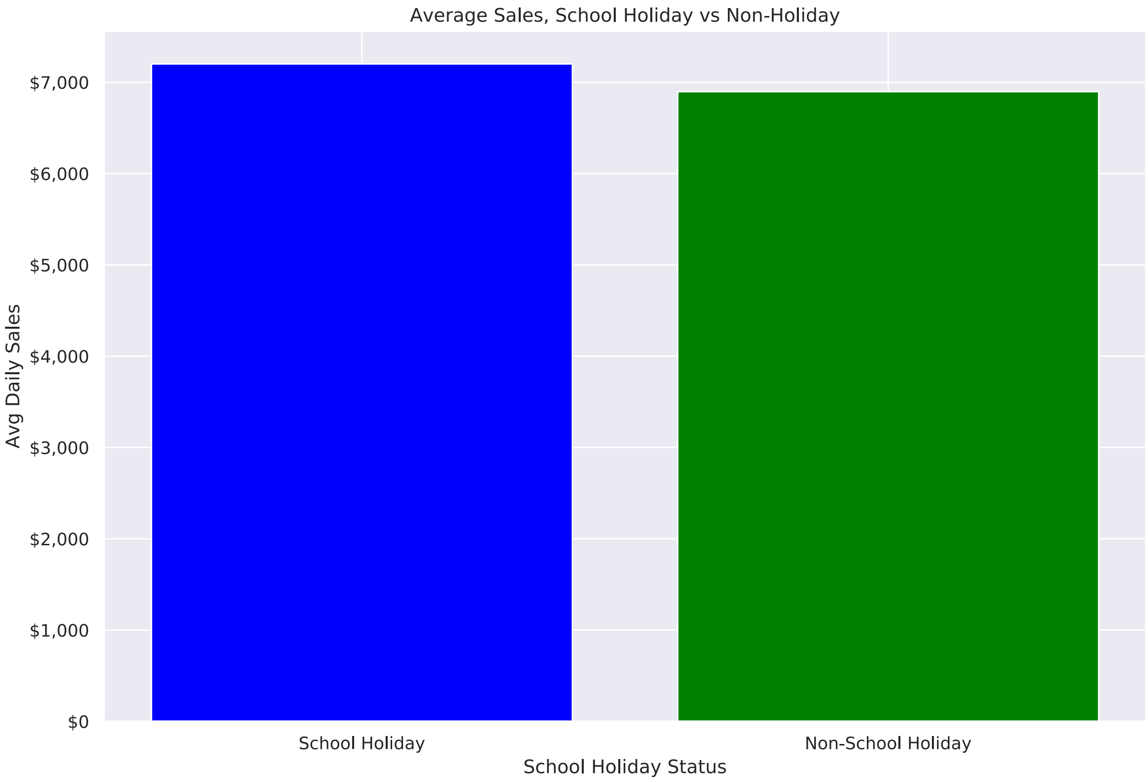
Sales on state holidays are substantially stronger than sales on an average non-holiday. However, only 25 days in our dataset are public holidays.



^ S12-school_holiday.py ⏱ code ⌄

## School Holiday

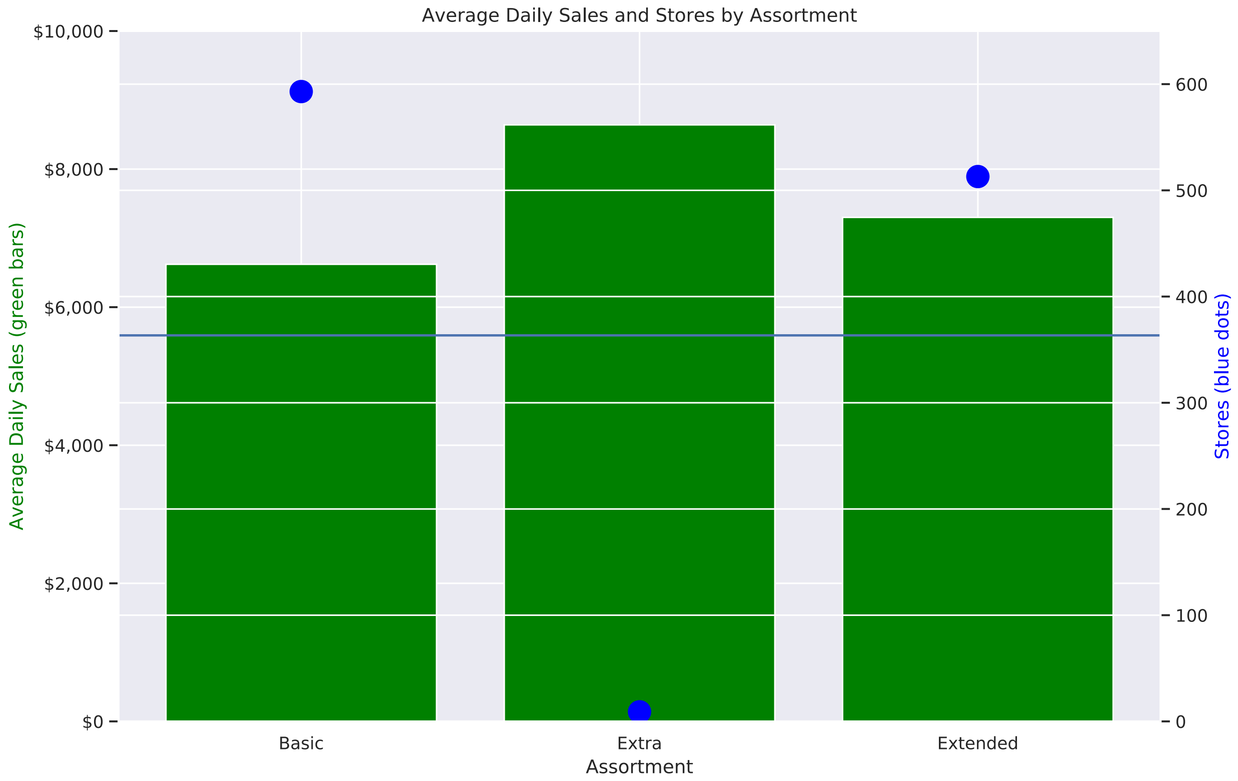
Sales on school holidays are slightly stronger than sales on an average non-holiday. **24%** of days in our dataset are school holidays.



^ S13-assortment_store_type.py code

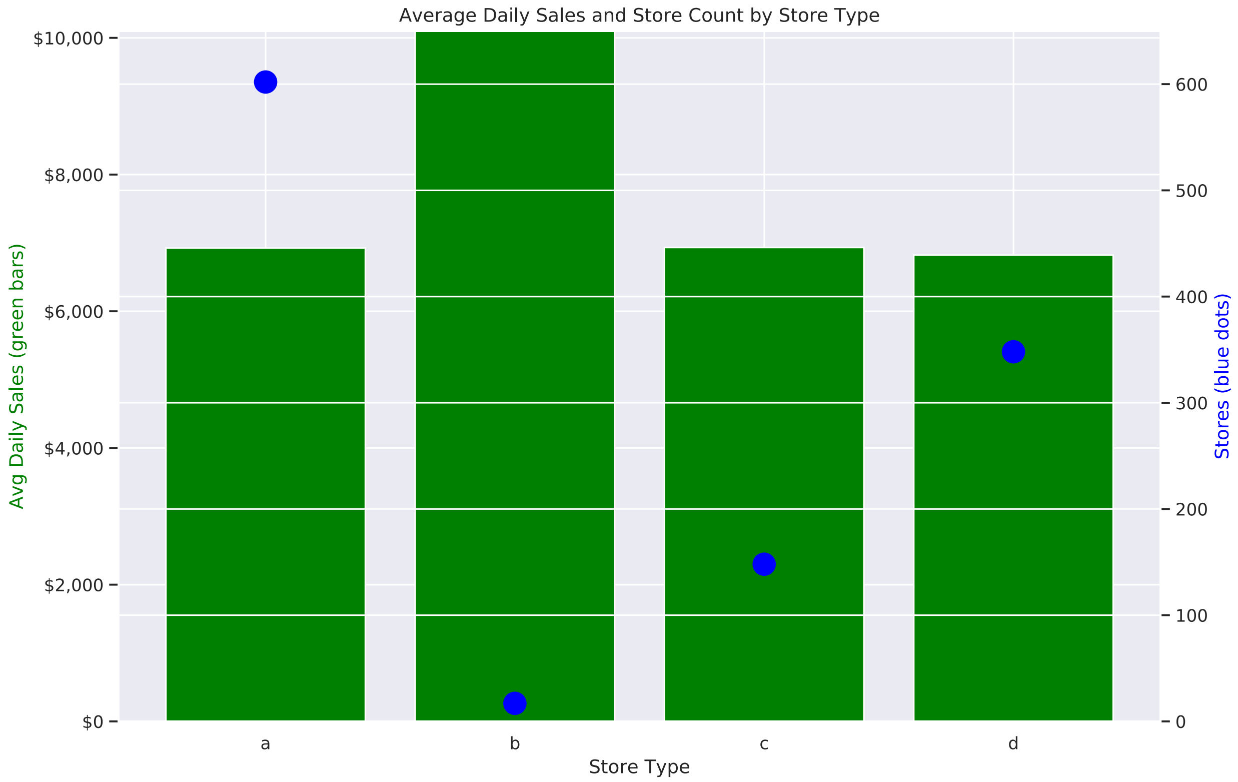
## Assortment

Assortment has a correlation with sales of 0.08. The 'Extra' assortment has outstanding sales figures, but there are very few of them. The 'Extended' assortment has much better sales than the 'Basic' format, and there are over 500 'Extended' stores in the Rossmann Germany business.



## Store Type

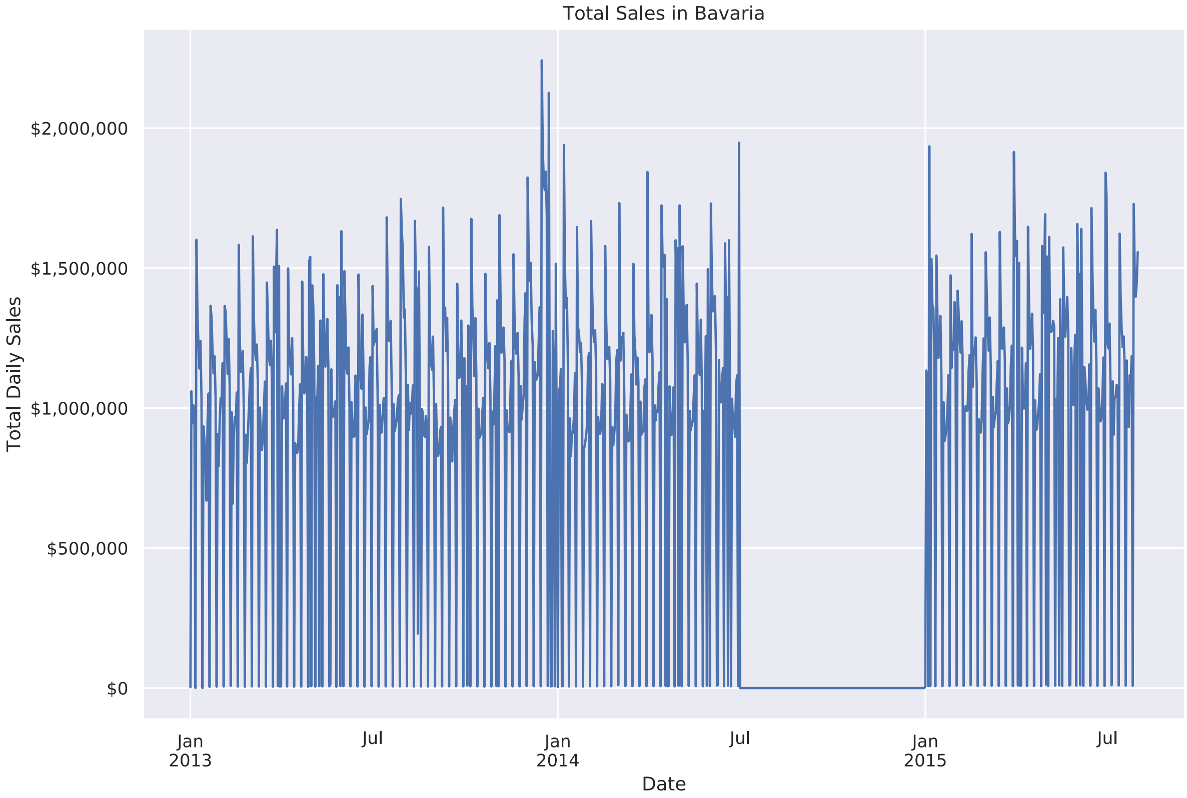
Store type is not strongly correlated with sales. Store type 'b' has daily average sales of about $10,000; but there are a handful of them. The other store types have daily average sales figures that are nearly indistinguishable.

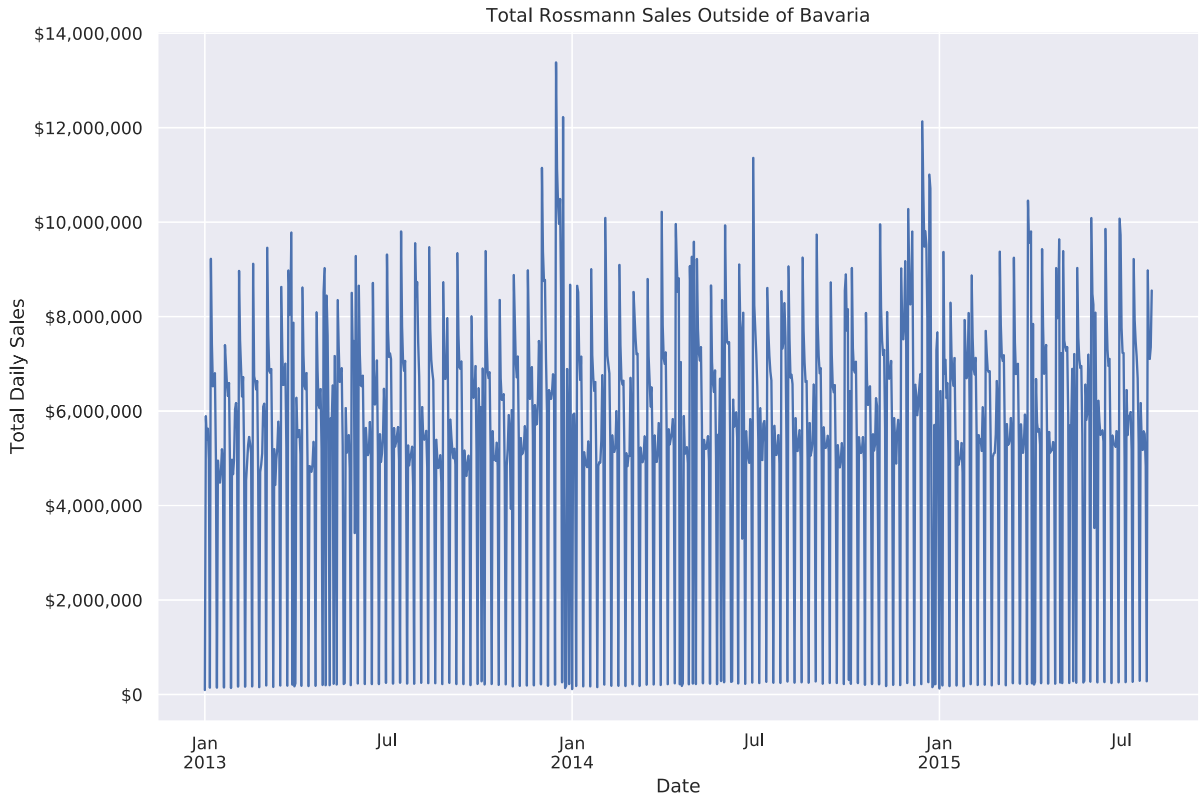


^ S14-store_closings.py ⏱ code ⌄

## Store Closings

From July 1, 2014, to December 31, 2014, all 180 Rossmann stores in Bavaria closed, presumably for remodeling. Below we see the sum of sales for all the Bavaria stores, and then the sales for all other Rossmann stores.


Total Sales in Bavaria
Total Daily Sales
$2,000,000
$1,500,000
$1,000,000
$500,000
$0
Jan
2013
Jul
Jan
2014
Jul
Jan
2015
Jul
Date

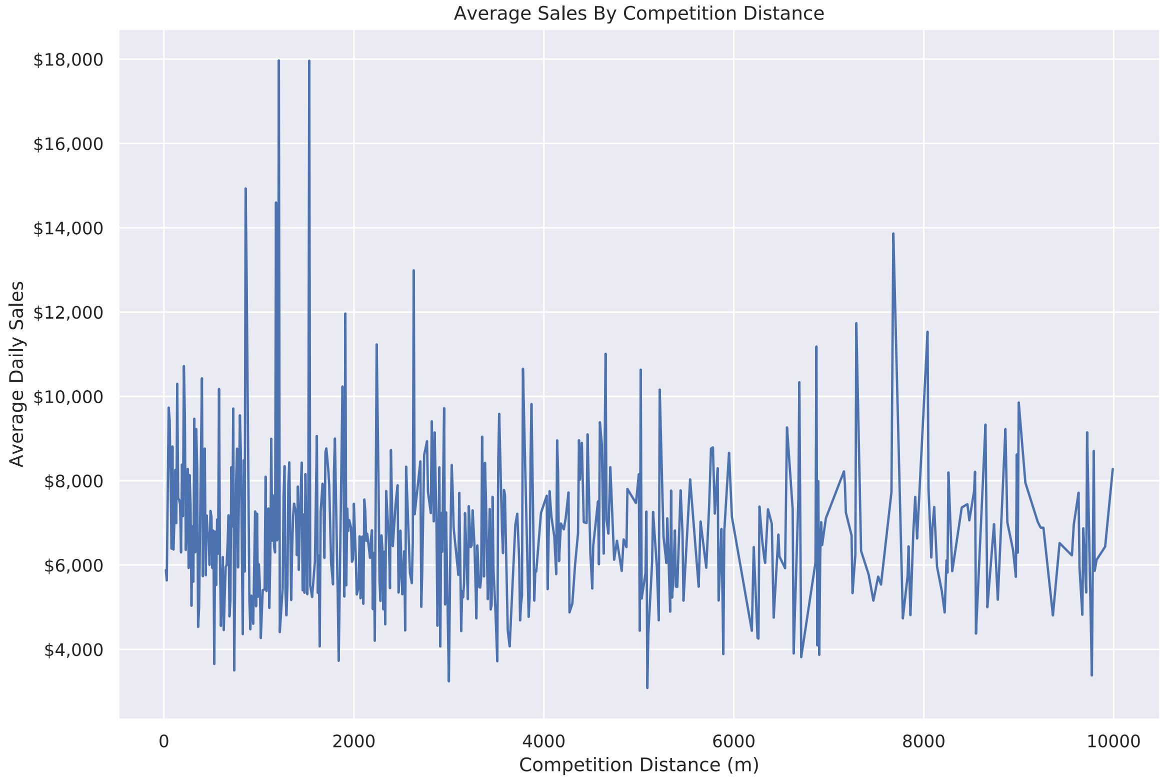


^ S15-competition.py code

## Competition

Although we have data on the competition, the data that we have turns out to be incredibly uninteresting. Below, for example, is average sales by competition distance up to 10km. The competition distance seems to have no relationship with the sales. The same is true for the age of the competition.

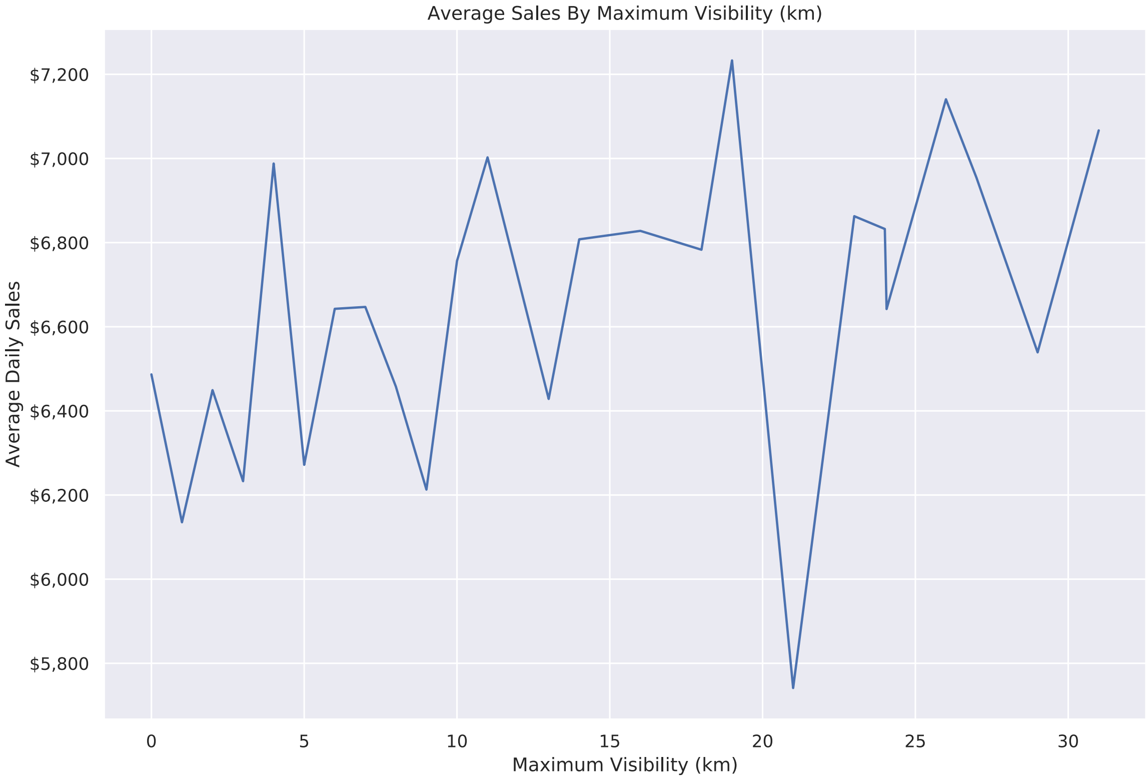
In fact, the competition has so little to do with the sales that among the competition variables, the one that has the strongest correlation with sales is which *month* the competition opened.



S16-weather.py code

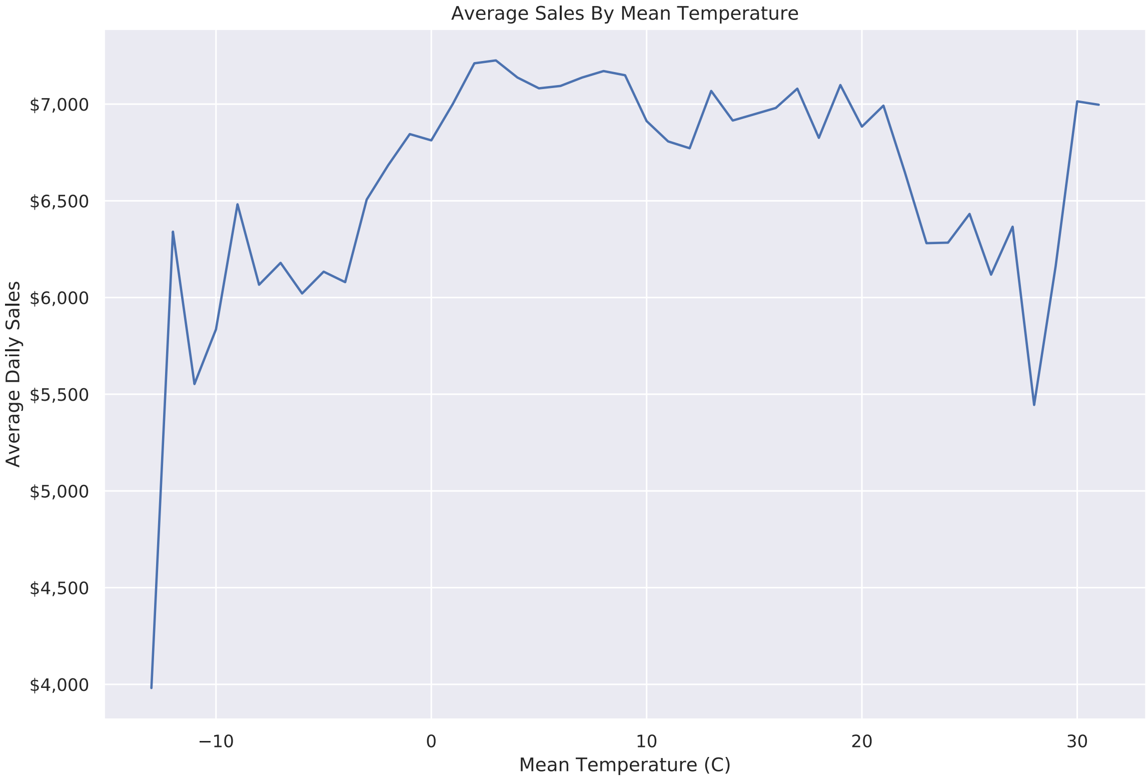
## Weather

Not surprisingly, the weather has a correlation with sales. The most strongly-correlated variable with sales is max_visibility_km:



## Temperature

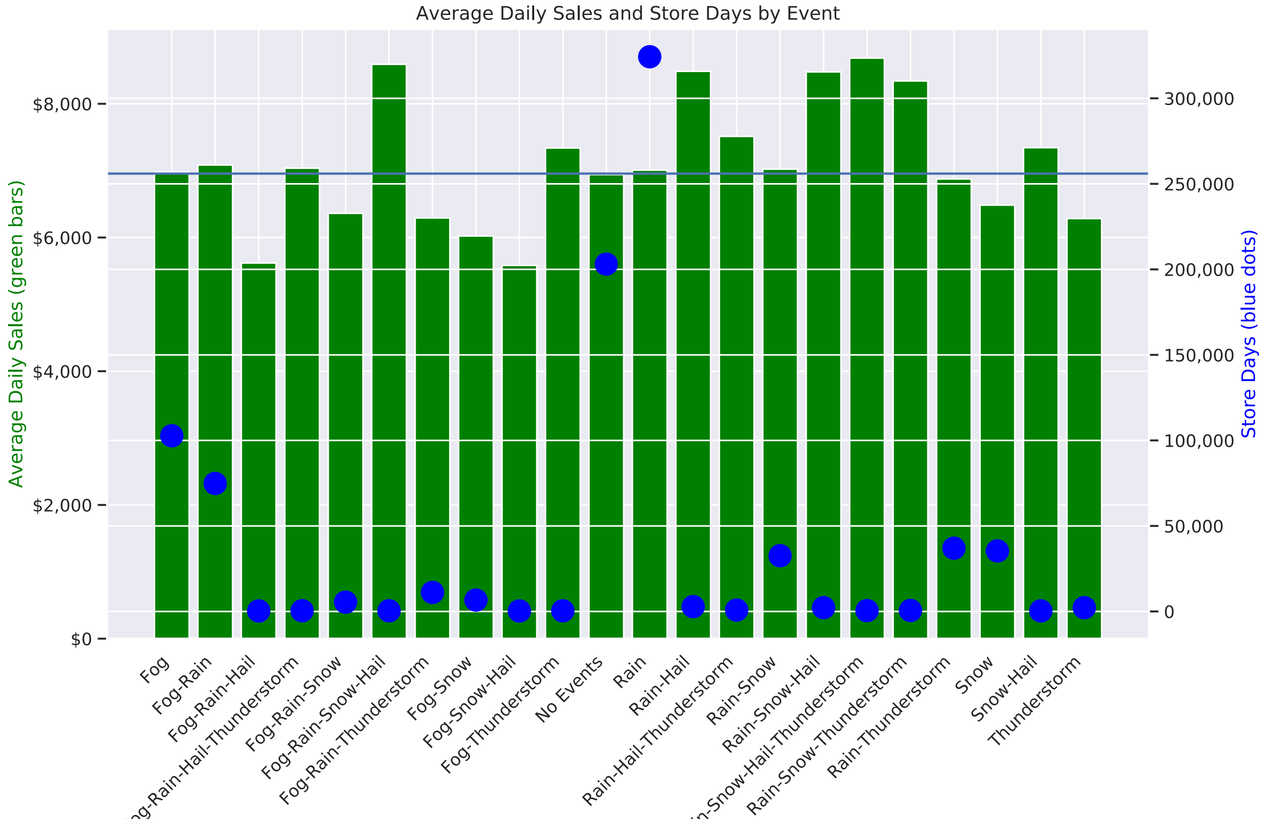
Many other weather-related variables correlate to sales. Here we see the mean temperature. Sales are elevated when the temperature is comfortable, and generally lower when it's too cold or too hot.



## Weather Events

Despite the intricacy of the chart below, it basically says two things: it rains a lot in Germany, and weather events don't particulary correlate with overall sales.

By far the most common weather events ("None" and "Rain") have average daily sales that are right in line with the overall average. The other weather events happen too infrequently to matter in the big picture.


Average Daily Sales and Store Days by Event
Average Daily Sales (green bars)
$0
$2,000
$4,000
$6,000
$8,000
Store Days (blue dots)
0
50,000
100,000
150,000
200,000
250,000
300,000
Fog
Fog-Rain
Fog-Rain-Hail
og-Rain-Hail-Thunderstorm
Fog-Rain-Snow
Fog-Rain-Snow-Hail
Fog-Rain-Thunderstorm
Fog-Snow
Fog-Snow-Hail
Fog-Thunderstorm
No Events
Rain
Rain-Hail
Rain-Hail-Thunderstorm
Rain-Snow
Rain-Snow-Hail
in-Snow-Hail-Thunderstorm
Rain-Snow-Thunderstorm
Rain-Thunderstorm
Snow
Snow-Hail
Thunderstorm