Forecast ticket sales for outdoor swimming pools

Zsombor Hegedus

2021 february 14

Executive summary

The goal of this analysis is to carry out a 12 month forecast for how many tickets will be sold on given day for outdoor swimming pools. This paper uses 6 years of historic transactional data which is available on Gabos Data Analysis site. After trying out several OLS regressions with increasing complexity, I will employ two tree based models; a random forest and an XGBoost. For model evaluation my loss function was RMSE, and I used 5-fold cross validation for control within the time series data (for each fold I used one year of observations as test and train the models on the rest). Random forest was chosen as the final model which was used to forecast sales in 2016, that served the purpose of a holdout set. The model could be confidently used for sales prediction so that the owners of the swimming pool can plan ahead knowing how much they can expect to earn in a given year.

Github: For this assignment, due to size limitations, I only submitted the PDF file, but for every workfiles with all the codes needed to produce this document, please visit my github repo.

Data description, and feature engineering

The dataset used is `swim-transactions` that contained transactional data for tickets sold for swimming pools in the city of Albuquerque between 2010 and 2016. First of all transactional data needed to be aggregated to daily frequency, and then gaps had to be filled (e.g. missing days when there were no tickets sold were inserted with 0 as quantity of tickets sold) so that I have a full gapless daily time-series data. I also filtered for outdoor swimming pools only, (those denoted with location that starts with *AQ* and ends with *01*), and removed tickets sold for a few categories, such as swimming competition etc... so that I can focus on the business-as-usual activities. Figure 1 shows how daily sold ticket volumes looked like in the whole timeseries between 2010 and 2015 (2016 is left out intentionally as that will be the holdout set).

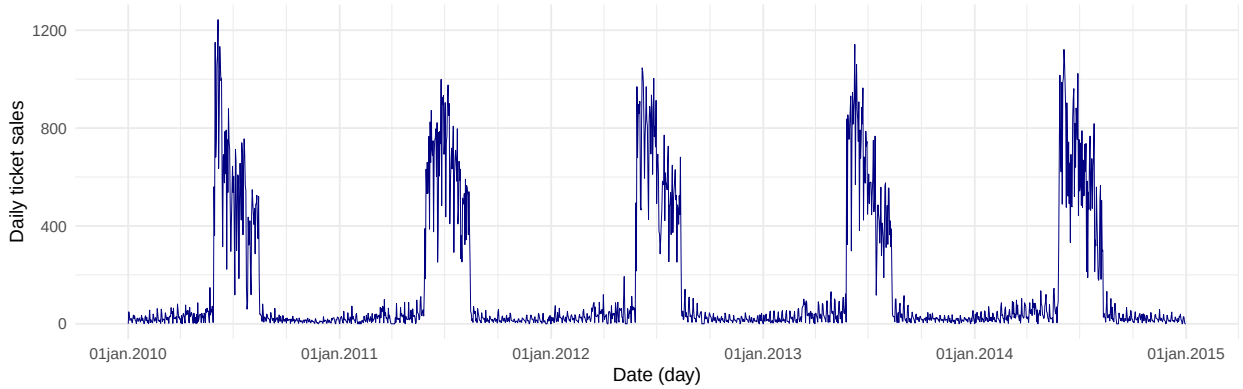

Figure 1: Time series of tickets sold between 2010 and 2016

Figure 1 illustrates very clearly that there is a strong seasonality and that much more tickets are sold for summer months. This is probabily due to the weather, but also, the schools close to these swimming pools are not open at that time, which turned out to be an important factor in the case study of the book. To get to the best possible predictions I introduced the following predictors:

- Variables that leverage the time series properties such as factors for day of the week, factor for months, dummies for weekend and holidays, and also for days when the schools are closed.
- I further enriched my dataset with the 1st and 2nd Δ for the log of tickets sold.
- I also experimented with taking the log of ticket's sold as my outcome variable (given its skewed distribution) but it performed much worse than the other models, so I decided not to pursue that.

To further highlight the purpose of a day of week variable and a weekend dummy, I included a heatmap in Figure 2. It can be seen that on the weekends the ticket sales are always higher, and that on Fridays there is lower frequency of sold tickets. From the heatmap one can also notice that swimming pools are much more visited in the summertime.

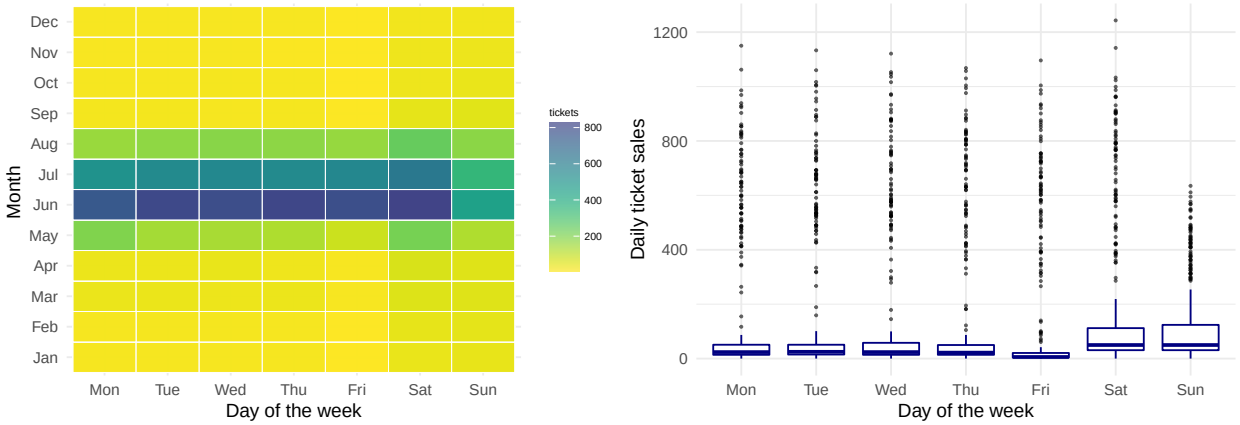

Figure 2: Heatmap of tickets sold for days of the week for different month and boxplots for day of the week ticket sales

Modeling

I started off with 6 simple OLS regression each incrementally more complex compared to the previous. For train control, I used 5 fold cross validation, in which I always stripped out one of the years to be a test set so that the remaining observations can be used to train the models. 2016 data was put aside to be a holdout set which will only be used once the final model for the forecast is chosen.

For the first few cases, I just wanted to see if it makes sense to include the variables that I created and whether RMSE of the prediction on the training set will decrease. The first one uses the trend (an ordered number list that starts with 1 and increases by one for each day), and the month factor only. The second includes the day of the month, while the third also includes holidays. The fourth looks at an interaction between the schools being closed and the day of the week, while the fifth also uses and interaction between weekend and months. The last model looks at the first and second Δ of ticket's sold as well. Table 1 summarises the cross validated RMSEs of the above mentioned cases which clearly indicates that the models increase their predictive performance on the training set with the inclusion of more and more variables. It's interesting to see that the biggest improvement in RMSE terms is when the interaction terms are introduced in the regression - most importantly the *school_off* dummy, which takes up the value of 1, in case schools were closed on given date.

Table 1: Comparing CV RMSE of OLS models with increasing complexity

OLS1	OLS2	OLS3	OLS4	OLS5	OLS6
123.1289	121.8023	121.3916	110.4563	109.6736	108.9308

After seeing what OLS is capable of I experimented with a random forest and an XGBoost model, while also running Facebook's Prophet (Prophet had an RMSE of 101.8 that didn't come close to that of the other two tree based models, so I didn't pursue it further). All three managed to beat the OLS regression models, with the random forest and XGBoost performing on pair with each other. However I needed to choose one model in the end and I went with random forest given it had the lowest CV RMSE. Table 2 summarises the RMSEs both on the training and on the holdout set (and it looks like XGBoost would have done slightly better on the holdout, but model choice should not be impacted by holdout RMSEs). The Random Forest with it's 83,21 RMSE achieves an incredible almost 35% improvement on the fit compared to the simplest OLS, and still had almost 25% improvement when compared to the best OLS.

Table 2: Comparing CV RMSE and holdout RMSE on OLS and tree based models

	CV RMSE	Holdout RMSE
OLS1	123.1289	109.0136
OLS6	108.9308	98.0292
Random Forest	83.2135	81.8565
XGBoost	84.8225	79.5580

Forecast

Forecast using the chosen model can be seen visually on Figure 3 alongside with a barplot with normalised RMSEs for all 12 months.

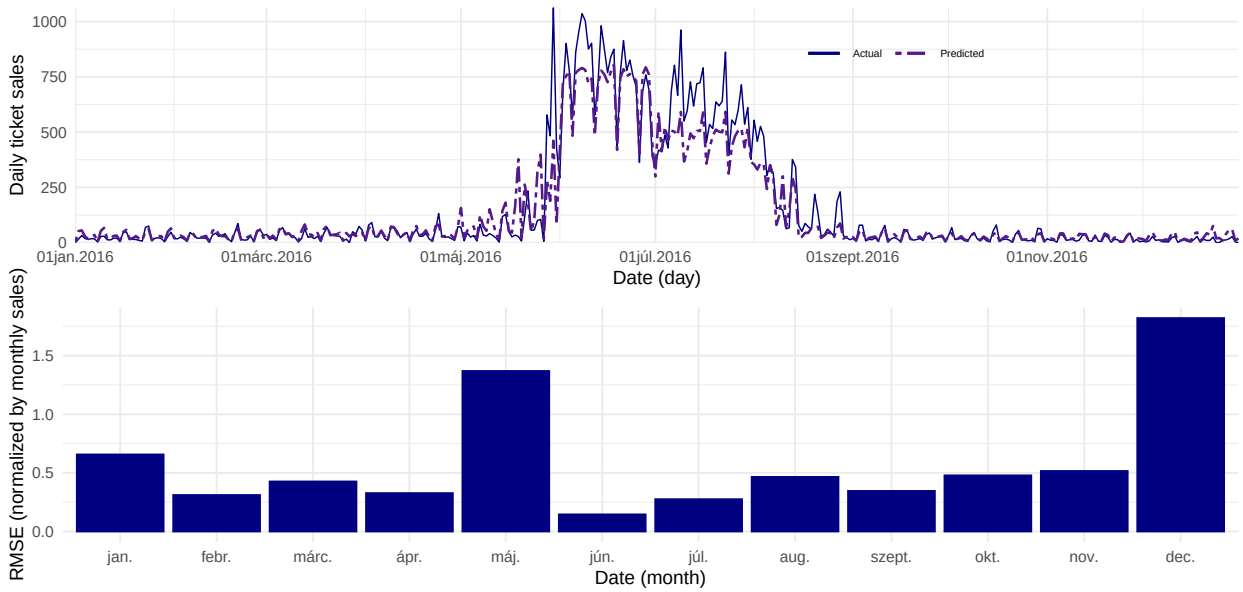

Figure 3: Forecasted and actual data for ticket sales in 2016 and normalised RMSEs for each month

It is visible that the prediction pretty much captured the seasonality and also had a relatively good performance for all months. When looking at each month in detail December, and May looks to be the worst, and Jun and July to be the best when it comes to normalised RMSEs. When taking a closer look at these months in Figure 4 it looks like the model consistently overpredicted in Dec and May whereas it consistently underpredicted for Jun and July. Understanding the dynamics behind these could be valuable for the business and could enhance the forecast by improving model assumptions.

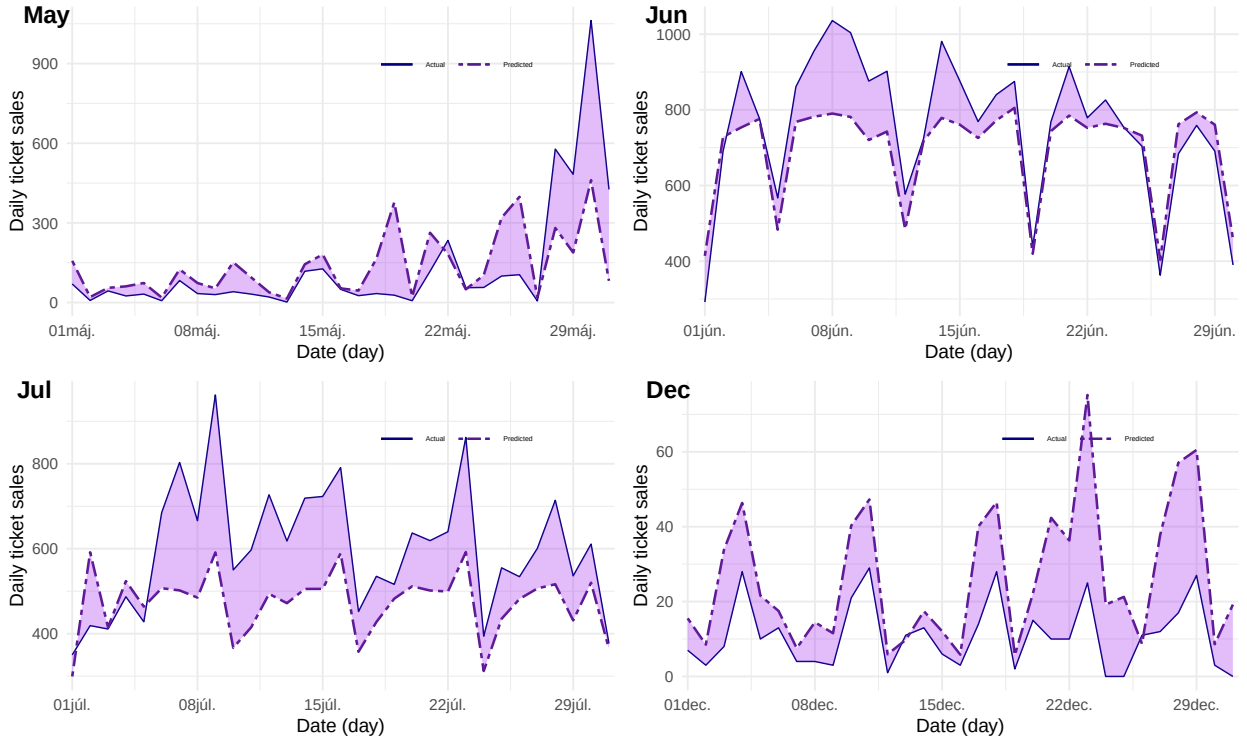

Figure 4: Forecasted and actual data for ticket sales in 4 handpicked months

The variable importance plot in Figure 5 can show even more information as to what variable plays a pivotal role when the random forest forecasts. In this figure I summed up the importance of each factor variable and grouped them together. Months won by a large margin, which is understandable given the seasonality in the summer months, but the day of the week variable might come as a surpirse. I thought that just by seeing a different behaviour between weekdays and weekends the day of the week variable will be an important factor when splitting the nodes of the trees, but it doesn't seem to be this way. Another very interesting finding is that the *school_off* dummy was one of the most important variables to consider and it definitely makes sense to include such a variable in the models.

External validity

I believe that external validity of the forecast of sales within the same region is high. In my point of view, behaviour of people going to the swimming pool is not something that changes abruptly over the years, unless some external factor plays a role, such as a global pandemic that forces these facilities to close - or the renovation of the building and increasing it's capacity which might attract more people to come and spend some time swimming in the pools. However I don't think that the model could be applied to any outdoor swimming pool in general since as we saw school schedule plays a massive role in the sales, and it might be specific to this region only.

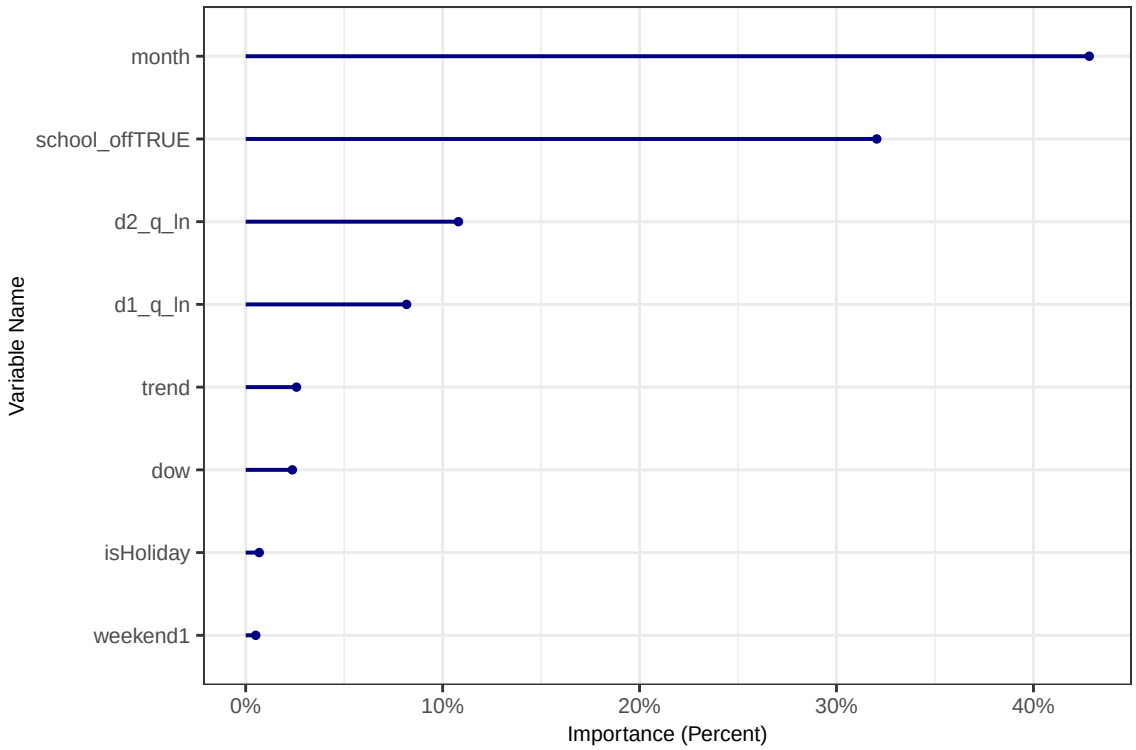

Figure 5: Most important variables in the random forest model (month and dow factor variables are grouped)

Conclusions

My task was to carry out a 12 month forecast on ticket sales of outdoor swimming pools in Albuquerque. I used different machine learning models to carry out a time-series forecast using predictors such as day of the week, month, holidays and so on. It turns out that a random forest model can do a relatively good prediction beating simple OLS, Facebook's prophet and even XGBoost. Models were trained on a 5 year long time series and tested on the year 2016 which was the holdout. Ticket sales had high seasonality in the form of a massive increase in the summer time and higher average sales on the weekends. The random forest model used month, a categorical predictor and *school_off* dummy as the most important predictor. I believe that this model can do pretty well on forecasting sales, has high external validity and could be essential for business planning.