Rossmann Time Series

Allen Huang[1], Jesiska Tandy[2]
[1]Department of Management Science and Engineering, Stanford University, Stanford, California
[2]Department of Chemical Engineering, Stanford University, Stanford, California

Objective

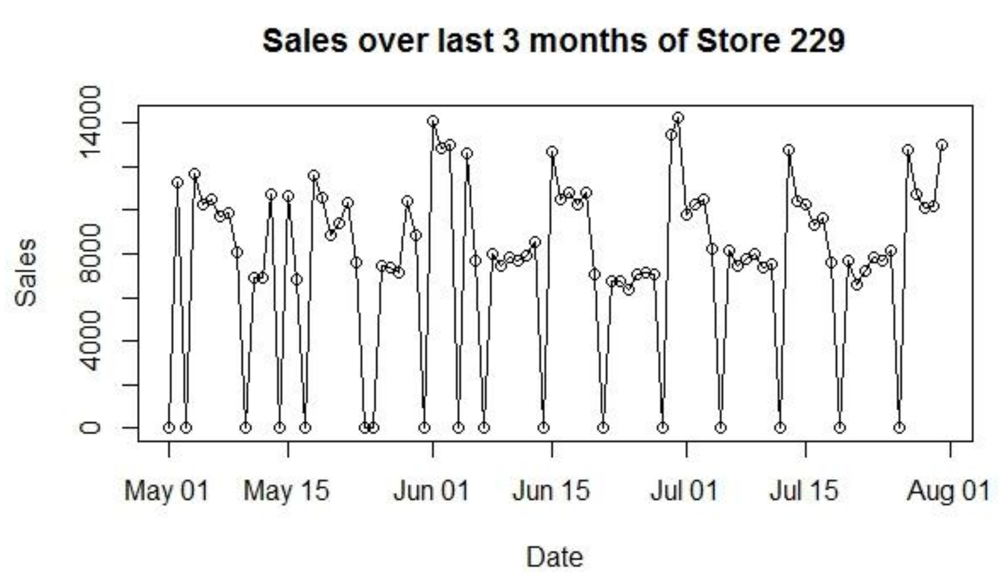
Goal: To explore how incorporating time series will improve learning models.

Methods: Recursive Partitioning and Bagged Trees using Improved Predictors in R.

Dataset

Column / Feature	train.csv	test.csv	store.csv	Comments
Store	x	x	x	Store Identifier (1-1115)
DayOfWeek	x	x		1-7 (1- Monday, 7- Sunday)
Date	x	x		year, month, day
Sales	x			
Customers	x			
Open	x	x		Binary: 1 for open
Promo	x	x		Binary: 1 for active promo
StateHoliday	x	x		Binary: 1
SchoolHoliday	x	x		Factor: 0, a, b, c
StoreType			x	Factor: a, b, c, d
Assortment			x	Factor: a, b, c
CompetitionDistance			x	In meters to nearest

Total number of stores: 1115
Dates: 1/1/2013 – 7/31/2015



Data divided into training, validation, and testing sets according to dates as follows:

Training: 1/1/2014 – 5/31/2015
Validation: 6/1/2015 – 6/30/2015
Test: 7/1/2015 – 7/31/2015

For our bagged tree models, we predicted on the log(sales) to stabilize the variance of the sales.

Note: Data obtained from Kaggle's "Rossmann Store Sales" competition.

Choosing Optimal Complexity Parameter

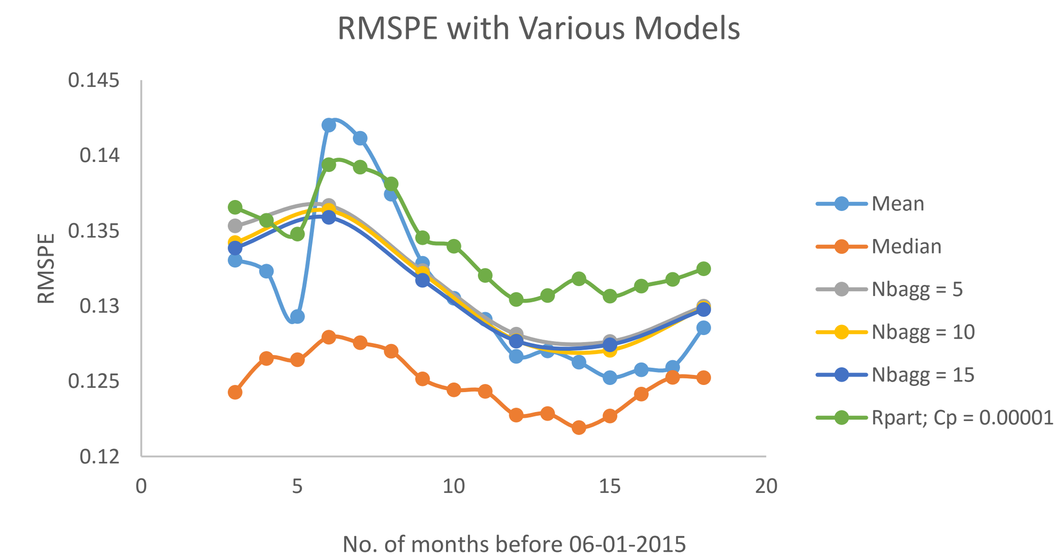
Variables	Cp	RMSPE
Store + Others	0.01	0.273
	0.001	0.217
	0.0001	0.154
	0.00001	0.139
	0.000001	0.137
	0.0000001	0.136

Sales ~ Store + DayOfWeek + Promo + StateHoliday + SchoolHoliday + StoreType + Assortment

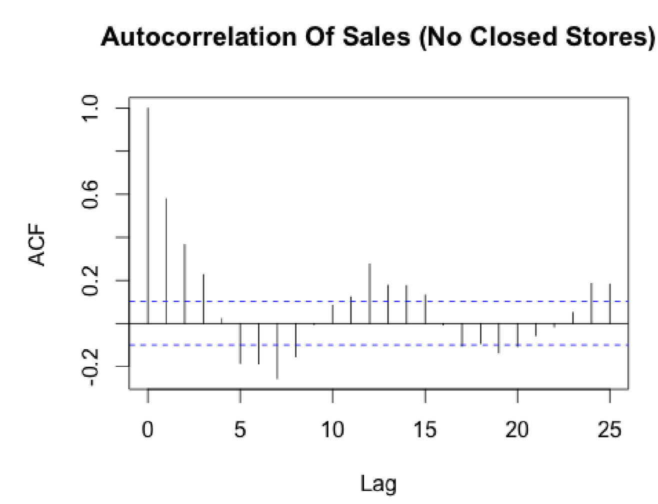
The metric that we chose to quantify our generalization error is the root mean square percent error (RMSPE) across all stores over the validation time period.

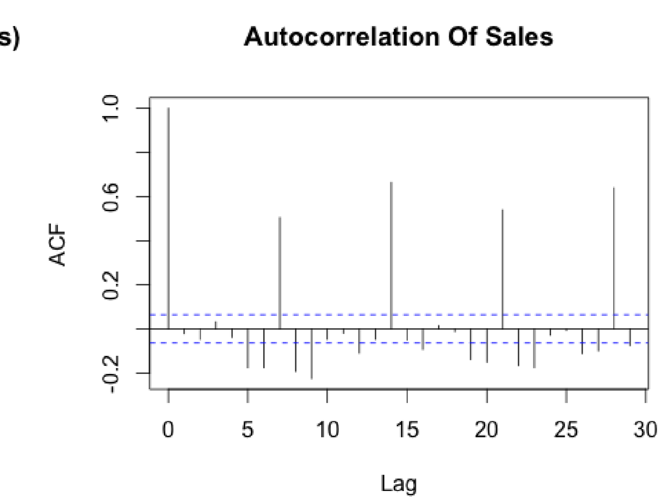
As we increased the complexity of our decision trees, we found that RMSPE decreases, and found that Cp = 0.00001 was the optimal cutoff. Smaller Cp values yielded no significant increase in performance at the cost of computational complexity.

Understanding the Effects of Time Series



We found that the optimal period for training will be 14-15 months prior to 06/01/2015.





Left graph suggests that sales is highly correlated with the last few days that the stores were open.

Right graph shows sales exhibit strong correlation with day of week.

Time Covariates	Description
tm(i)	sales i days before*
MA7	average sales over last week
MA28	average sales over last 4 weeks

*i = {1,2,3,4,5,6,7,14,21,28}

To capture the dependence on time, we decided to add these features in our bagged tree model. The values we decided to incorporate were drawn from the analysis of the autocorrelations shown above.

Results with Time Series

Baseline

Our best baseline model was simply to predict the historical median of all the stores with the same Store ID, DayOfWeek, and Promo.

Bagged Trees Without Time Covariates

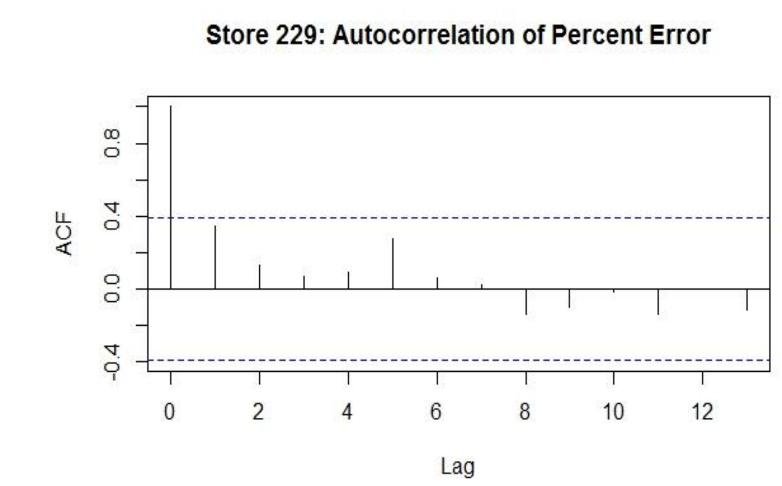
Our best model without incorporating time was the bagged trees, with Cp = 0.00001, and Nbagg = 5. Sales ~ Store + DayOfWeek + Promo + StateHoliday + SchoolHoliday + StoreType + Assortment.

Bagged Trees With Time Covariates

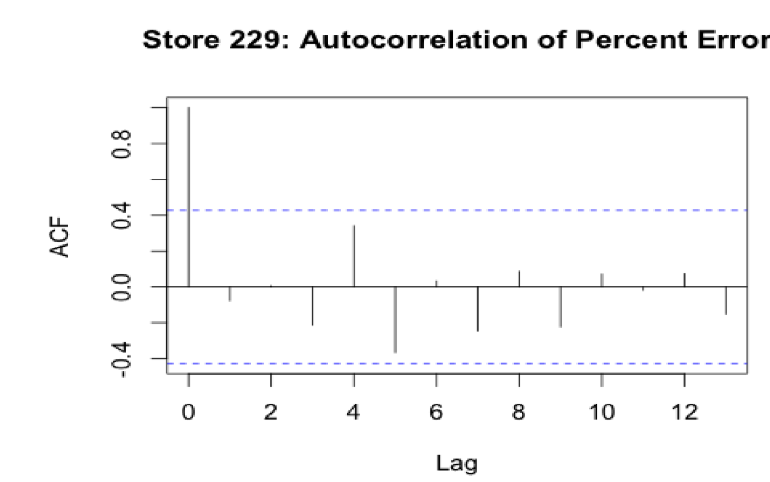
Our best model with time was the bagged trees, with Cp = 0.00001, and Nbagg = 20. Sales ~ Store + DayOfWeek + Promo + StateHoliday + SchoolHoliday + StoreType + Assortment + tm1 + tm2 + tm3 + tm4 + tm5 + tm6 + tm7 + tm14 + tm21 + tm28 + MA7 + MA28 + trend1 + trend2 + tm1:tm2 + tm1:tm7 + tm1:tm14 + tm1:MA7 + tm1:MA28 + MA7:MA28.

Model	RMSPE	MAPE
Baseline (Median)	0.1252	0.093
Bagged Trees without Time Covariates	0.1277	0.095
Bagged Trees with Time Covariates	0.1169	0.085

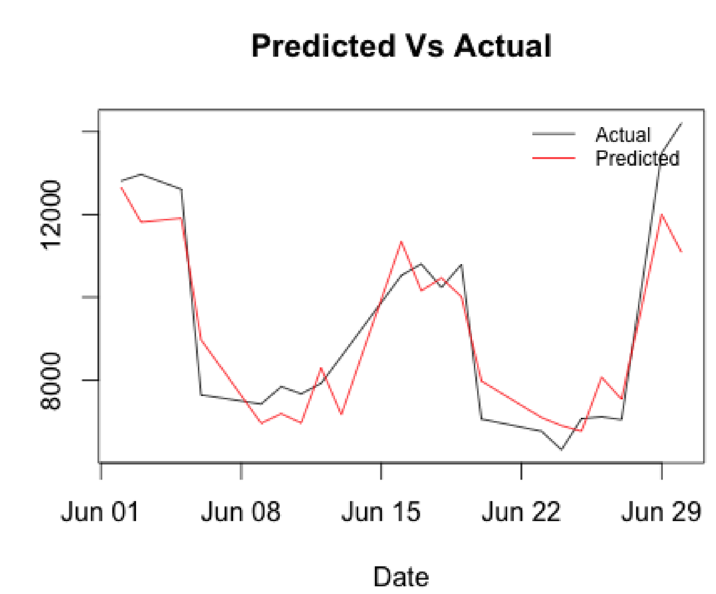
Bagged Trees without Time Covariates



Bagged Trees with Time Covariates



In time series data, one useful diagnostic is to examine the autocorrelation of the residuals. The left graph has a clear pattern, which shows that there is some temporal structure unaccounted for in our "Bagged Trees without Time Covariates" model. **Our best model does!**



The graph on the left shows the predicted vs actual sales for store 229 using our best model with time covariates.