STAT495 HW#8

Fei and Trevor

December 9, 2015

This homework is due at midnight on Tuesday, December 8th (one report per pair). Please add, commit, and push the Rmd and pdf files to your public github repository as well as create and then close an issue for "HW#8". Your task is to improve upon your model from HW#6, in conjunction with your partner.

The files for this assignment can be found in the `STAT495-Horton` github repository (called `hw08.Rda`). This is a slightly modified dataset: fewer stores, no days with zero sales, and no missing values.

It will be important to control for store in your model.

Use this code as a starting point along with your two draft answers for HW#6 to improve anwers to the following questions.

Predicting sales

You've been promoted to a new position that gives you responsibility for monitoring sales at a chain of stores. Your goal is to predict sales at the store level. Data fields that are available include the `store`, `sales` (in dollars), `customers` (number of customers), `storeopen` (whether the store was open that day), `promotion` (whether there was a sale on), `noschool` (whether there was school that day), `type` (type of store: 4 levels), and `distance` (distance to nearest competitor).

Part one Part one of your challenge is to generate a model for sales that will be used to predict sales for future months. Your R Markdown file should generate a model called `finalmod()` which I will use to generate predicted values and calculate test error.

SOLUTION:

The three things that we originally want to improve in the email:

1. Randomly choose a sample instead of filtering by date in a hope to avoid temporal correlation.

Even though date was removed in the new dataset we included a variable for day of the week in order to account for some level of temporal correlation.

2. Include distance in the model in order to account for spatial correlation.

We included distance as a predictor in our model, but we found that it is only significant as a stand alone predictor and it was not particulary significant when used in interaction with other variables.

3. Control for the store itself.

This was done in our new model.

```
finalmod <- lm(sales ~ type + customers*promotion + as.factor(dow)*promotion +
                 as.factor(store)*as.factor(dow) + distance, data=merged)
```

```
sqrt(mean((predict(finalmod)-merged$sales)^2))
```

```
## [1] 525.288
```

Part two Part two of your challenge is to interpret your model: what do you conclude about predictors of sales?

SOLUTION: So we found that the individual stores themselves are some of the biggest predictors of sales. Another major predictor is the day of the week and we see a general trend where most of the stores see an increase in sales as they reach the weekend. (i.e Friday and Saturday)

We also found that promotions are a major factor that influences sales and especially when looked at with its interaction with customers and days of the week. This intuitively makes sense; when there are sales customers are probably more likely to buy more during their visist to the store and sales are likely to be more effective on weekends when people are not working.

Other good indiviudal predictors include distance and store type, but they did not interact well with other predictors. This makes sense in that different types of stores would expect different sales patterns, but it was a little surprising that the interactions with other predictors, for example customers, did not decrease the error much (if at all). For a better understanding going forward it would be useful to understand what the different "types" are references to.

Part three Part three of your challenge is to assess your model: how well does it work for the training data?

SOLUTION: Our model seems to work very well for our training data. On first glance our Adjust R-squared value seems so show that our model accounts for the vast majority of variance in the data coming in at 0.9589. Digging deeper we see that the mean squared error is 525.288 which is substantially lower than both of our orginial models as well as the models provided above. The majority of the predictors are significant with the exceptions of some of the interactions between the weekdays and the stores which is understandable as we described above. It would be interesting to see how this data works on a test set because one concern might be that we overfitted the data.

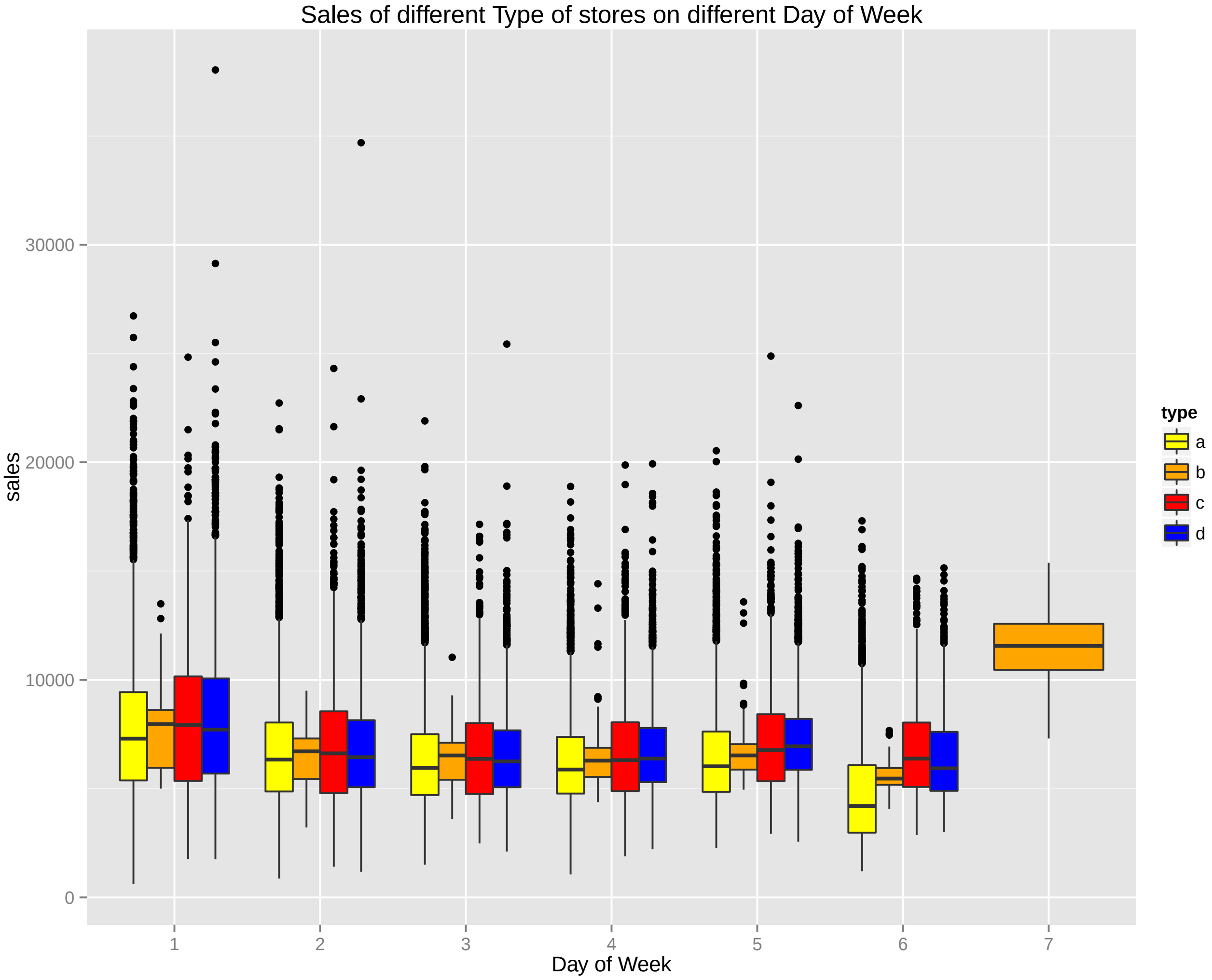
Visualizing patterns of sales In addition to your predictive model, please generate one visualization of the data that tells a story related to sales. Please pay particular attention to the quality of your visualization. You should include a single paragraph interpretation of the figure.

SOLUTION:

```
require(ggplot2)
require(reshape2)
```

```
## Loading required package: reshape2
```

```
ggplot(merged, aes(x = as.factor(dow), y = sales)) +
  geom_boxplot(aes(fill = type)) +
  scale_fill_manual(values = c("yellow", "orange", "red", "blue")) +
  xlab("Day of Week") +
  ylab("sales") +
  ggtitle("Sales of different Type of stores on different Day of Week")
```



The plots are pretty interesting. Firstly, even though we notice that different stores show similar trend of sales during the week (i.e. higher sales during weekends), it is not very evident from the box plots. Besides the substaintially high sales on Day 7, the mean of Day 1 is only slightly higher than that of others. The exception occurs for type a on Day 6, as its mean is much lower than the rest.

Secondly, only type b shows sales on Day 7 and the sales are exceptionally high, around 12,000 compared to only around 7,000 on average of other days. We are interested to know what type b is referring to and why it is the only type having sales on Day 7.

Thirdly, even though type b shows great sales on Day 7, it is not the type that always has the greatest sales on other days. There are quite a number of extreme values occuring for type d on Day 1, 2, 3 and 5, while type b has the least variation among the four types.

```
outlier <- merged %>%
  filter(dow < 4) %>%
  filter(type == "d") %>%
  arrange(desc(sales))

head(outlier)
```

```
## Source: local data frame [6 x 8]
##
```

```
##    store   dow sales customers promotion noschool  type distance
##    (int) (int) (int)     (int)     (int)    (int) (chr)    (int)
## 1     57     1 38037      1970         1        0     d      420
## 2     57     2 34692      1930         1        0     d      420
## 3     57     1 29138      2009         1        0     d      420
## 4     35     1 25506      1461         1        0     d     7660
## 5     57     3 25435      1644         1        0     d      420
## 6     57     1 24615      2153         0        1     d      420
```

Except for one store 35, the rest of the highest sales belong to store 57 (i.e. the star store in type d).