Interactive Visualization Analytics:

Applied to Airport On-Time Performance Data
http://flyvis.com

Matthew Delhey
Frank Portman
Mentor: Yeshaya Adler

December 6, 2013

Abstract

We explore the possibility of improving data analysis through the use of interactive visualization. Exploration of data and models is an iterative process. We hypothesize that dynamic, interactive visualizations greatly reduce the cost of new iterations and thus facilitate agile investigation and rapid prototyping. Our web-application framework, flyvis.com, offers evidence for such a hypothesis for a dataset consisting of airline on-time flight performance between 2006-2008. Utilizing our framework we are able to study the feasibility of modeling subsets of flight delays from temporal data - a similar approach fails on the full dataset.

1 Introduction

Exploration of data and models is an iterative process that first allows us to understand the data and then to make predictions with quantifiable uncertainty. In turn, the workflow for most data analysis follows a similar path, moving from broad examinations, visualizations, and mathematical summaries of the data to those more precise, resulting in statistical modeling or inference. Traditionally, this has been an a static procedure conducted a priori by the analyst in isolation from the final audience of the findings. We intuitively experience this disconnect when we quickly iterate on plots and only later fully develop their interpretation and role in the presentation of the analysis. Dynamic and interactive data visualizations bridge this gap between author and audience within the data analysis workflow.

At the general level, interactive data visualizations are representations of information that allow users to easily interact with the data. As such, they facilitate discovery of insights from one's own exploration and interpretation of the data–information that was perhaps beyond the scope of the foresight of the original application author. For large, high-dimensional datasets it is unlikely that a single perspective on the data will uncover all depths of its insights. While static visualization plays a key role in summarizing the main characteristics of a dataset, a dynamic approach allows the user to immerse themselves within the analysis in order to draw the most insightful conclusions.

Similar to their static counterparts, interactive data visualizations generate schematic forms of the data representations and transformations to support analysis and the ability to glean information. Interactive visualizations have the unique asset of behaving as a sort of sandbox in which a user can rapidly prototype a set of hypotheses or ask questions of the data. If well designed and abstracted, such applications break down the rigidness we normally associate with non-programming approaches to data analysis and allow the user the flexibility to discover nuances or subsets in the data that are perhaps more difficult to seek out in a traditional programming or visualization environment. Indeed, the most popular use of interactive visualization is the mapping of a complicated dataset such that it facilitates the discovery of subsets of the data meaningful for the user.

Here we have outlined the general process and workflow of data analysis and explained how interactive visualizations can improve it. By focusing on the end user of the data analysis, dynamic and interactive visualizations allow for new interpretations of the data, which in turn form new iterations of data exploration and model building.

2 Study Design and Data Collection

The fundamental design of our study was the creation and exploration of a interactive visualization framework for a specific example dataset. We seek to demonstrate the effectiveness of this class of interactive visualizations for common tasks in data analysis and modeling. This section pertains to a general account of the design decisions made in the creation of our interactive visualization, flyvis.com, in comparison to its static counterpart, and a discussion of our dataset. The details of an example data analysis actualized using our framework follow in the subsequent section.

2.1 Design of Interactive Visualization

We focused the design study of our interactive visualization into three categories: interactivity, ease of use, and web deployment. While not all interactive visualizations display characteristics of each of these three categories, they capture the essence of our intentions when we speak of interactive visualizations and each have important implications for the practical improvement of data analysis. Additionally, these categories stand in contrast to a static visualization which is often characterized by its lack of interactivity, technical knowledge required for its creation, and its divergence with the modern, interactive web.

The most identifiable feature of any dynamic visualization is its interactivity as it consummates the most significant divergence from traditional plotting methods. User computer interaction is a vast area of research and any schematic of its contents is outside the scope of this report. Nonetheless, important observations relative to our current discussion can be made regarding interactive design. As noted in the introduction, adding interactivity to a visualization decreases the time and cost of new iterations and thus facilitates further data analysis as a whole. The plots, maps, and any other visual representations of data update automatically as the user moves through the data analysis framework thus allowing particularly effective exploration of high dimensional, geo-spatial, and temporal data. Inference about variable interaction in these data, such as changes in time and location, are easier to uncover with a more intuitive means for controlling their visualization.

These considerations informed the implementation of interactivity in our application. We chose a geographical map as the primary user interface because it allows the user to explore the data and easily perform meaningful subsets in a complex feature space. The user is first presented with a high level perspective on the data and asked to explore the data by zooming into a region of interest, thus allowing for meaningful and personal narrative to unfold. The geographical interface is not statically bound to a single variable; the user can easy select what features of the data they want to be mapped to the geographical map allowing for cross-geographical comparisons on the new feature space as well as cross-feature comparisons. Interactivity is also used to speed up variable referencing in the analysis by allowing the user to click on variables, metrics, and airports for their full description as opposed to looking this information up in a data readme. Numerical summaries are similarly dynamic, updating depending on what portion of the map is visible to the user or the current airport the user is currently looking at.

The interactivity of the interface makes the application easy to use. Users can quickly get the application running for data analysis without the use of prior training. The framework simply works as we've come to expect from modern web applications. Despite the inclusion of computationally involved representations of data, there is absolutely no technical knowledge required on behalf of the user. These two categories are additionally complimented by the third, the existence of the application on the web. Hosting our framework in this fashion allows for two key benefits: (i) universal accessibility and complete portability regardless of local operating system or dependency issues and (ii) cloud computation as opposed to local computation allowing for exploration of datasets that are too large to fit on commodity hardware. This second asset is particularly synergistic with more traditional statistics, allowing the user to explore a dataset using the framework that otherwise would not fit in memory, finding a suitable subset of the data for statistical modeling, and finally downloading the data for computation on their local machine.

Despite some of these desired properties of interactive visualizations, they are certainly not suitable for every exploratory data analysis and have their own, different set of costs. From the design of our own framework, we conclude that it is best to use interactive visualization for ill-defined, open ended problems, complex or high dimensional data, and analyses with large, diverse audiences. On the other hand, interactive visualization is not best suited for analyses that begin with a specific result or summary in mind. Each of the characteristic categories of interactive visualizations introduce higher development costs for the author. Interaction, intuitive design, and web deployment each add in an order of magnitude of complexity and abstraction to be considered when developing an interactive framework. As interactive visualizations increase in popularity and technical capacities continue to grow, we expect these costs to continue to decrease.

2.2 Dataset

To facilitate our discussion on interactive data analysis we turned to the On-Time Performance dataset from the Research and Innovative Technology Administration (RITA) in the Bureau of Transportation Statistics (BTS). The BTS tracks the on-time performance of domestic flights operated by large carriers and has released public records dating from 1987. Since the dataset holds every single domestic flight for this time period, we used a smaller subset (2006 - 2008) in order to effectively explore the data without running into computational challenges. The 2006-2008 data has approximately 18 million observations of over 5000 airports. To clean our data, we removed all observations that had NAs, negative flight times, exorbitant delays (negative or positive), and flight times that were suspiciously fast.

There are several reasons why this dataset provides a good base to build an interactive application on. First of all, each airport is associated with a specific latitude and longitude. Geo-spatial data and analysis is greatly supplemented by the introduction of interactive maps. Second, wealth of information provided by the sheer number of observations allows for a very thorough exploration of the data through an interactive tool. For example, over 5000 airports are reported which means our application can provide either a highly stratified or general method of analysis. Regardless of the area you wish to explore, conducting static analysis of so many observations and variables usually results in taking a subset of the data and proceeding from there. In our case, you don't need to limit yourself right away as you can easily explore different aspects before potentially diving in deeper.

2.3 flyvis: In Action

In this section we would like to demonstrate an example of exploratory data analysis conducted using our framework in order to give the reader a better understanding of power of our application. We also strongly urge the reader to visit flyvis.com themselves and play around with their own data analysis in order to better understand both the features of our application and interactive visualizations more generally.

Here we were interested in comparing the two largest airports in Tennessee: Memphis International and Nashville International. We use the interactive map to find the two airports and compare their relative size, noting in the numeric summary that Memphis International is slightly larger than Nashville. Clicking on Memphis automatically generates several plots for the specific airport, the first of which seen in Figure 1 shows the distribution of flights by time of day, colored by type of delay (if any). We generate the very same

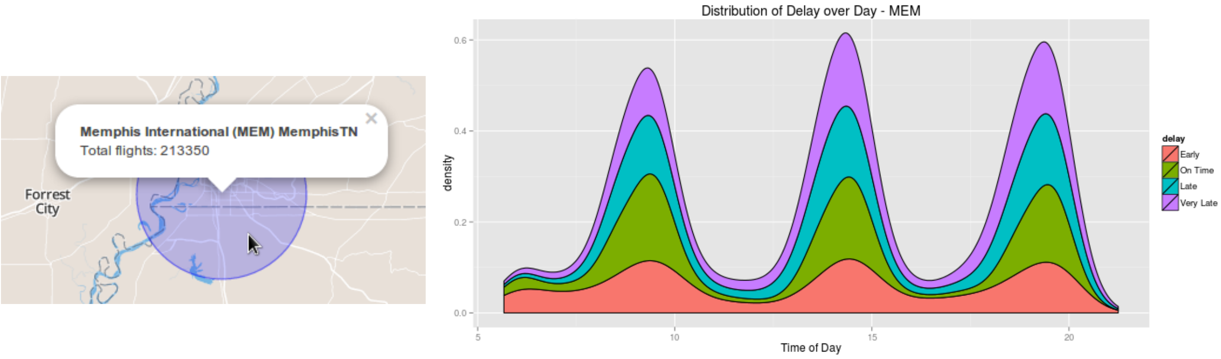


Figure 1: (Left): Selecting Memphis International Airport from its geographical location. (Right): Plot generated within web application showing the density of flights by delay over time of day for all days in the dataset.

plot for Nashville with the same methodology.

Looking at the two plots, we notice that Memphis experiences three major spikes in outgoing flight traffic throughout each day without any significant relative increase in delays. Nashville demonstrates a more traditional distribution of flight traffic, with a single jump in traffic in the evening. For Nashville, delays tend to increase as the day progress. We also compare the heatmap output of our framework, an example of which can be seen in Figure 2. Here we notice that Nashville has significant winter holiday delays whereas Memphis saw its worth month in March. Intrigued by these results, we conducted additional research about these two airports and discovered that Memphis International Airport is home to the FedEx Express global hub and experiences much less commercial traffic compared to Nashville International. A question for future research might be to try and explain the relationship between the spikes in airport traffic at Memphis International in terms of its position as a primarily cargo hub.

3 Methods of Statistical Analysis

We present an example of how interactive and dynamic data analysis can enhance the predictive modeling process. Every year, over 20% of flights are delayed to some extent - causing travelers and airlines much lost time and money. Being able to effectively model and predict the severity of departure delay is a lofty goal which would prove valuable to anyone who travels. The sheer amount of data we have (1̃8 million observations) suggests that we should should be able to leverage all of this information in order to predict a particular flight's delay.

Traditional approaches to this challenge take a pretty similar route. People often build their models using predictors such as “Arrival Delay” and “Local Weather”. We, however, feel that these types of models do not necessarily translate well into useful information. With this in mind, we created several restrictions for which data we would use. For example, none of the individual delay variables were considered when constructing the model - if these features are known, one already has a good idea about whether a flight will be delayed or not. The driving motivation behind our models is to give an individual some idea of how long their flight will be delayed using basic information that can only be known at the time of their purchase (usually weeks to months in advance). Some inputs we used were ’Day of the Week’, ’Month’, ’Time of the Day’, ’Day of the Month’ and other interaction effects between these variables. ’Day of the Week’ and ’Month’ were treated as factors while the rest of the inputs were scaled appropriately to between 0 and 1.

We took two basic approaches in constructing our predictive models - regression and classification. For the regression technique, we were interested in predicting the actual departure delay in minutes. The classification approach was inspired by the results of the regression model. We saw that while RMSE was high for many regression models, the overall magnitude of the delay was preserved which suggests this problem is better posed as an exercise in classification. For that classifier model, we split our departure delays into five different factors as follows in Table 1:

Table 1: Our Classifications

Delay	Classification
Less than -5 Minutes	Early
Between -5 and 5 Minutes	On Time
Between 5 and 30 Minutes	Late
More than 30 Minutes	Very Late

In this way we feel that we fully encapsulate all the possible delay scenarios. The ‘On Time’ grouping takes into consideration human error in recording results because it is very rare for a flight to be EXACTLY (precisely zero seconds of departure delay) on time. There is much evidence to suggest that the data is recorded by hand due to the structured intervals represented in some variables, so we allow a small buffer for that. In addition, knowing that your flight will be ‘delayed’ by three minutes isn’t crucial information, so we feel our ranges are fair.

3.1 Model Selection

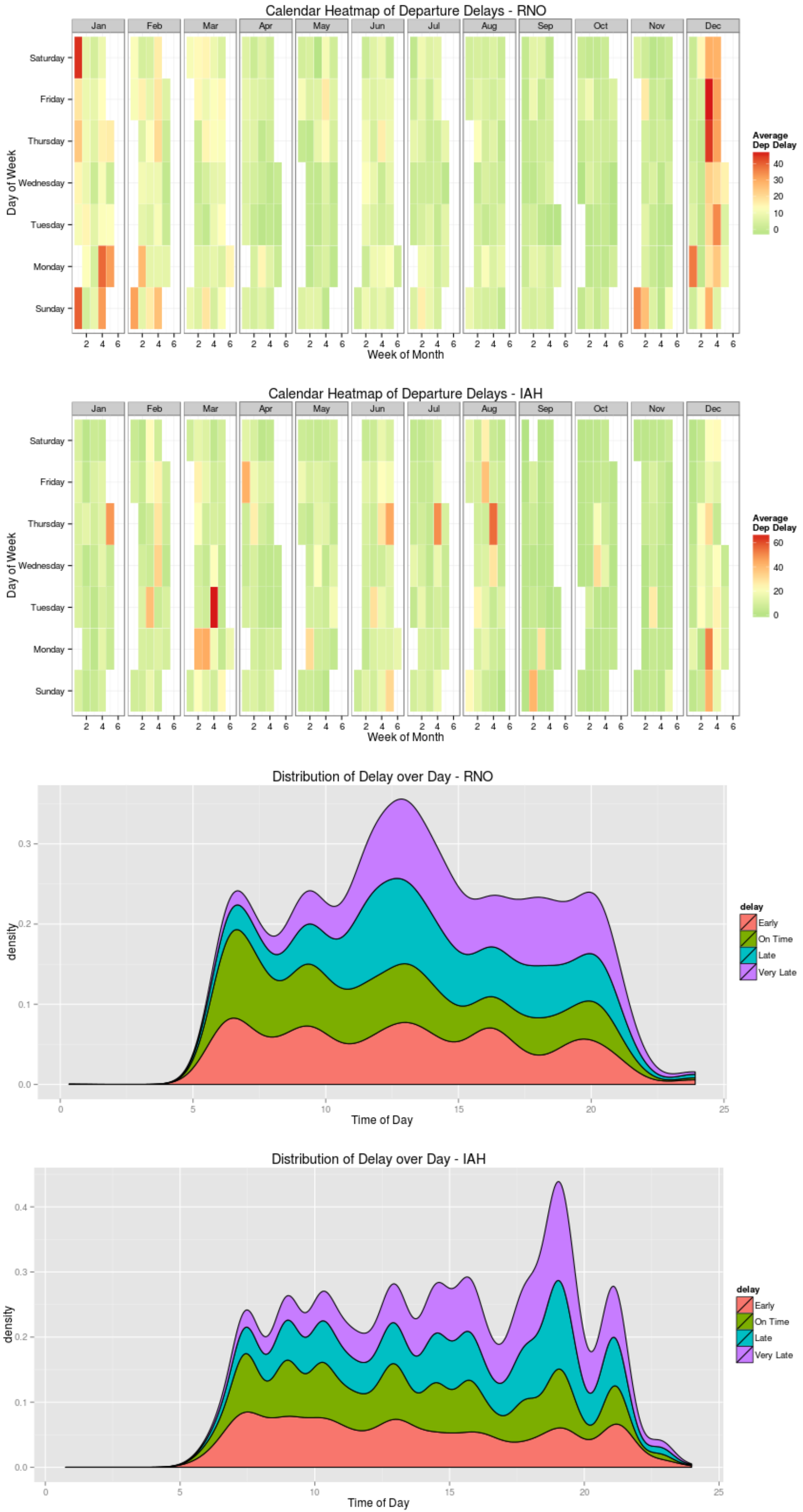
Not unexpectedly, naive approaches on the full dataset failed and were computationally intensive to fit. The reason, however, was only exposed to us after using our interactive framework to explore the temporal features of many airports. Two different trends immediately jumped out at us and paved the way for the effective predictive modeling of the data. First, most airports tended to have the severity of their delays propagate smoothly as the day progressed. These airports usually also had predictable patterns of delays throughout the year as seen in Figure 2. Other airports didn’t seem to encompass any discernible patterns or correlations between delay and temporal features.

Figure 2 shows two airports with completely different structures in their delay vs. time relationships. Using flyvis, we were able to quickly identify that almost every airport fit into one of the two categories. We introduce the term *modelability* to describe the feasibility of the fitting a powerful predictor to a subset of the data. The modelability of an airport is heavily reliant on the distribution and severity of its delays over the course of the day and year. Consequently, we hope to build temporal models on modelable subsets of the data and show that they outperform similar classifiers trained on other airports.

Initially, we had hoped to use a linear classifier (potentially with a Lasso Penalty for sparseness in factors) for interpretability purposes. Traditional multinomial linear classifiers, for the most part, failed alongside other fancier learners such as SVMs and Naive Bayes. In terms of sheer misclassification error, linear models did not perform poorly. Upon closer inspection, it became clear that most of these models, assigned almost observation to the ’On Time’ category. Since that bin makes up the majority of the data, classifying every observation identically could potentially show good, but misleading, predictive power.

Random Forests, which are ensembles of decorrelated decision trees, have long been considered one of the best black-box classification algorithms. In this case, the benefits they offered over some of the more interpretable models did not justify settling for a less than optimal prediction. When applied to our problem, Random Forests were able to not only show low misclassification error rates, but greatly minimized the false positive/negative rate as compared to linear classifiers.

In order to fairly test our hypothesis, we conducted 10-fold cross validation when training each Random Forest in order to optimize for ’mtry’ and ’ntree’ in the model. The motivation behind this was to get each airport down to a baseline classifier and explore the differences between airports with varying levels of modelability. Even after parameter optimization, the more modelable airports still show significantly better error rates on predictions.


Calendar Heatmap of Departure Delays - RNO

Calendar Heatmap of Departure Delays - IAH

Distribution of Delay over Day - RNO

Distribution of Delay over Day - IAH

Figure 2: We see entirely different distributions of delays in RNO and IAH. RNO follows a more traditional pattern of delays propagating over the course of the day, and being somewhat correlated to different times of the year. On the other hand, IAH, does not exhibit a nice structure of this type.

4 Results

Table 2: (Top) RNO Confusion Matrix (Bottom) IAH Confusion Matrix

	Early	On Time	Late	Very Late	Error
Early	1217	2951	57	17	.7131
On Time	300	12803	449	65	.0597
Late	1	1784	1748	233	.5358
Very Late	1	33	209	2043	.1063

	Early	On Time	Late	Very Late	Error
Early	86	15421	188	85	.9946
On Time	10	103370	4379	1292	.0521
Late	0	21721	11225	3260	.6900
Very Late	0	549	2384	16544	.1506

Table 2 shows the confusion matrices of two optimized Random Forest models on the same airports in Figure 2. As we saw before, RNO exhibits characteristics that we deem to be more modelable, while IAH seems to have a less obvious relationship between delays and temporal structures. Table Y and the associated misclassification errors of **.2187753** and **.1411066** for IAH and RNO respectively tell us that our suspicions were indeed correct.

5 Discussion

5.1 Application: Predicting Severity of Flight Delay

We validated our idea of modelability using a Random Forest classifier as seen in Table 2. Interestingly, the temporal model for RNO seems to be particularly good at predicting flights that are 'Very Late'. When it fails, it often places the flight in the 'Late' category (and vice versa) meaning that the false negatives aren't as drastic as if a 'Very Late' flight was predicted early. We see a few outlines that are likely caused by external factors such as weather or security delays. A similar phenomenon is apparent in other pairs of airports that display such juxtaposed structure in their temporal features. While it is nice to know whether your flight will leave Early or On Time, knowing if your flight will be 'Very Late' is perhaps even more useful information. Our models prediction accuracy for that class suggests that we have discovered a very practical result.

The beauty of our notion of modelability is that it is a very simple concept enabled by our interactive framework which can greatly enhance the model building process. As far as actual robustness of the classifier, our models rely on very basic features and principles in giving their predictions. If the model states that a flight should be leaving 'Early', but it is blizzarding on the day of your flight, there's a good chance that it will be wrong. Therefore, considering the fact that many delays are a result of weather, security delays, and other unpredictable events, we feel that our models do a good job at weaving through those special cases to present an individual with an accurate benchmark prediction as shown by our cross-validation errors and confusion matrices.

In this Section 3 we have shown how the interactive data analysis framework enhanced both the exploratory data analysis and predictive modelling of the flights dataset. An initial unguided approach to modeling yields inconsequential results, but by focusing on modelable subsets of the data, one can build robust classifiers on temporal features. We have outlined how flyvis enables users to quickly assess the modelability of certain airports within our scope.

5.2 Summary & Future Work

At a higher level, we can also interpret our work within the general context of interactive visualizations. We have demonstrated the possibility of improving data analysis with interactive visualization using a difficult dataset. By allowing the end user to iterate quickly through the data analysis workflow, interactive visualizations promote agile exploration of data and models.

Future work can be categorized in two ways: improvements which can be made to our application and takeaways for future work regarding interactive visualization. For the application, we would like to add another layer of interaction such that users can interact with the plots themselves–i.e. plots would have certain features that can be held constant or varied to explore other features in our data. For example, for our Calendar Heatmap plot, we would like to add a feature to allow users to click on a certain day and be able to explore that information further to create an even more specific subset of the data. Additionally, adding standard errors to our graphics would aid in the interpretation of their accuracy.

For dynamic and interactive visualizations as a whole, we believe that future work to enhance the speed of computation would be beneficial. One of the best features of interactive visualizations is their ability to speed up the data exploration process. While interactive graphics are almost always faster than comparable static and traditional methods, they usually do not achieve near-instantaneous speeds (1 to 5 seconds). This certainly held true for our dataset and framework despite optimization. Obtaining such speeds, we believe, would be a valuable accomplishment for interactive visualizations.

References

[1] H. Wickham. Practical tools for exploring data and models. Phd thesis, Iowa State, 2008

[2] RStudio and Inc. (2013). shiny: Web Application Framework for R. R package version 0.8.0. http://CRAN.R-project.org/package=shiny

[3] R Core Team (2013). R: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria. URL http://www.R-project.org/.

[4] van Zudilova-Seinstra, E., Adriaansen, T., Van Liere, R. (Eds.). (2009). Trends in interactive visualization: State-of-the-art survey. Springer.