Analysis of my Gmail Inbox

By Andrea Jackson

March 2016

CS 249

As part of a series of projects for my CS 249 class, I have analyzed my gmail inbox, created a few visualizations of my findings, and tested my hypotheses.

Before I started this project, I drafted a few questions I wanted to explore/answer about my gmail inbox.

1. How long does it take me to open an unread email (I have a bad habit of not reading my email).
2. What days of the week do I receive the most emails? Days of the month?
3. When do I receive the most emails? In the morning, afternoon, or evening?

Although I really wanted to answer Q1, I cannot answer it because Gmail does share/track when a user opens an unread email. I chose to explore Q3 because I wanted to see if I receive on average more emails at the beginning and ending days of a month than other days of the month. Similarly, I want to find out what days of the week I receive the most emails.

Question: What day of the week do I receive the most emails? Days of the month?

Hypothesis 1: I receive more emails on Mondays than any other day of the week. I think I receive more emails on Mondays and Tuesdays because it's the beginning of the school week. A lot of assignments and applications are due, it's a time when organizations start sending spam for an event later on in the week etc.

Hypothesis 2: I receive more emails at the beginning and ending days of a month. Deadlines for internships, research opportunities, major projects, and essays tend to fall within the last 5 days and first 5 days of a month. For this reason, I think this set of 10 days will receive on average more email than other days of the month.

In order to anwer my question, I will need to:

1. **Collect the data**
2. **Create a Dataframe of the information**
3. **Explore the data by creating visualizations**
4. **Test my hypothesis**

Data Collection

I downloaded all of my email from gmail by using Google's Takeout service (https://takeout.google.com/settings/takeout/custom/gmail). Most mail clients store emails using the mbox format (https://www.wikiwand.com/en/Mbox). I used python's mailbox.mbox class to create an object of the mbox.

In order to understand what an mbox object is and how to access each email in the mbox, I read python's API documentation and used python's built-in functions dir() and help(). The notebook (http://cs.wellesley.edu/~ajackso2/cs249/Gmail_Analysis_Notebooks/Exploring_mbox.html) I created to explore the mbox also contains functions and a Dataframe.

Dataframe

Creating the Dataframe with the correct variables was probably one of the most time-consuming, yet interesting, aspect of this project. I first created a Dataframe where each row represented one email. The Dataframe contained 5 columns: to, from, subject, time, and email type. Email type represents whether the email was received or sent. Although I only needed to collect the time the email was received and the email type to answer my question, I decided to go ahead and collect the other information to gain more practice extracting pieces of information from an email message.

After making the Dataframe, I created a csv file called gmailData.csv from the Dataframe so I would not have to load the mbox information every time I wanted to access a message.

In another notebook (http://cs.wellesley.edu/%7Eajackso2/cs249/Gmail_Analysis_Notebooks/DataVis_HypothesisTesting.html) I read from the gmailData.csv and created another Dataframe where the index is a DatetimeIndex. I also eliminated all emails I sent to other people because I'm only interested in emails I received. Using a Dataframe with a DatetimeIndex allowed me to resample the Dataframe by daily, weekly, and monthly emails. *Note: I resampled the data just to see how the distribution of the emails looked by day, week and month.*

I made a new Dataframe where each day of the week is a number 0-6. 0 represents Monday and 6 represents Sunday. Similarly I created a Series of the days of the month where 1 stands for the first day of the month etc. After creating the Dataframe for the day of the week and a Series for the day of the month, I used pandas' describe() functions to generate summary statistics of the Dataframe and Series.

dayOfWeekDF

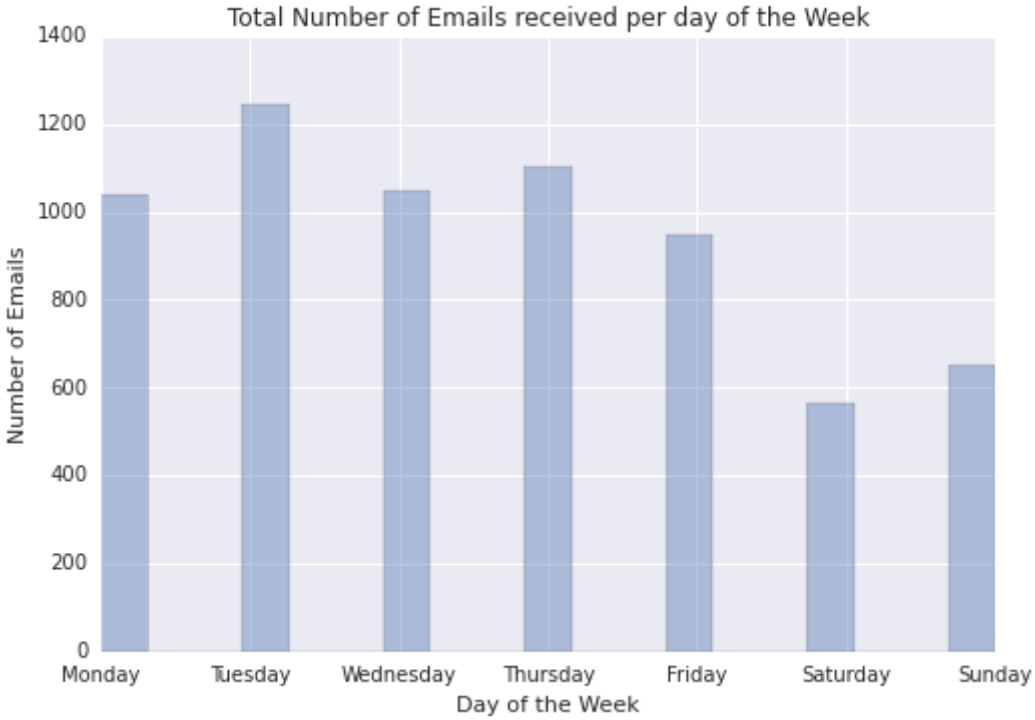
Statistic	Value
count	6598.000000
mean	2.602607
std	1.884011
min	0.000000
25%	1.000000
50%	2.602607
75%	2.000000
max	6.000000

daySeries

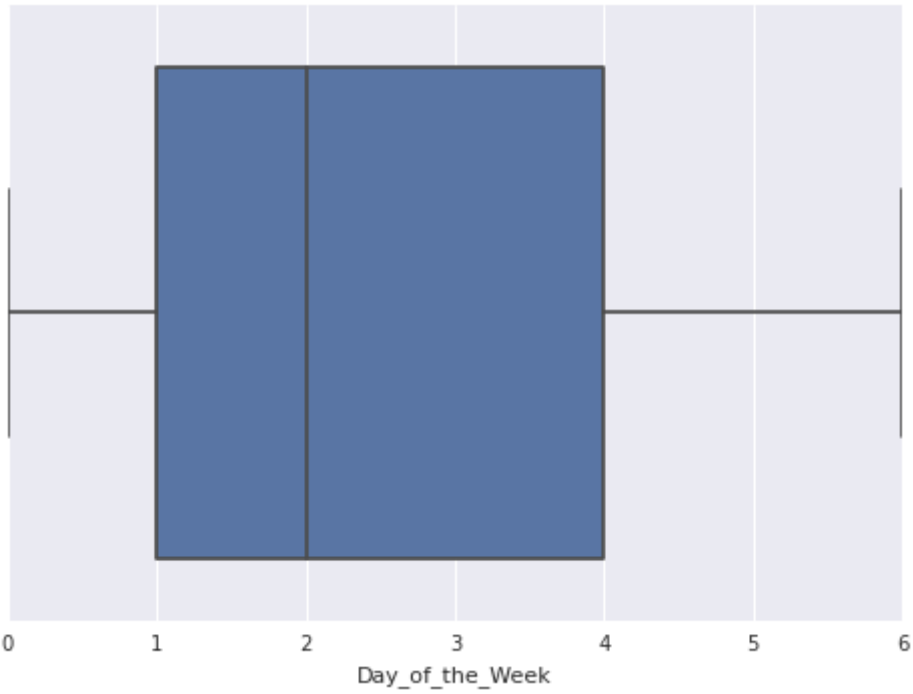
Statistic	Value
count	6598.000000
mean	14.324189
std	9.060201
min	1.000000
25%	7.000000
50%	13.000000
75%	22.000000
max	31.000000

Data Visualizations

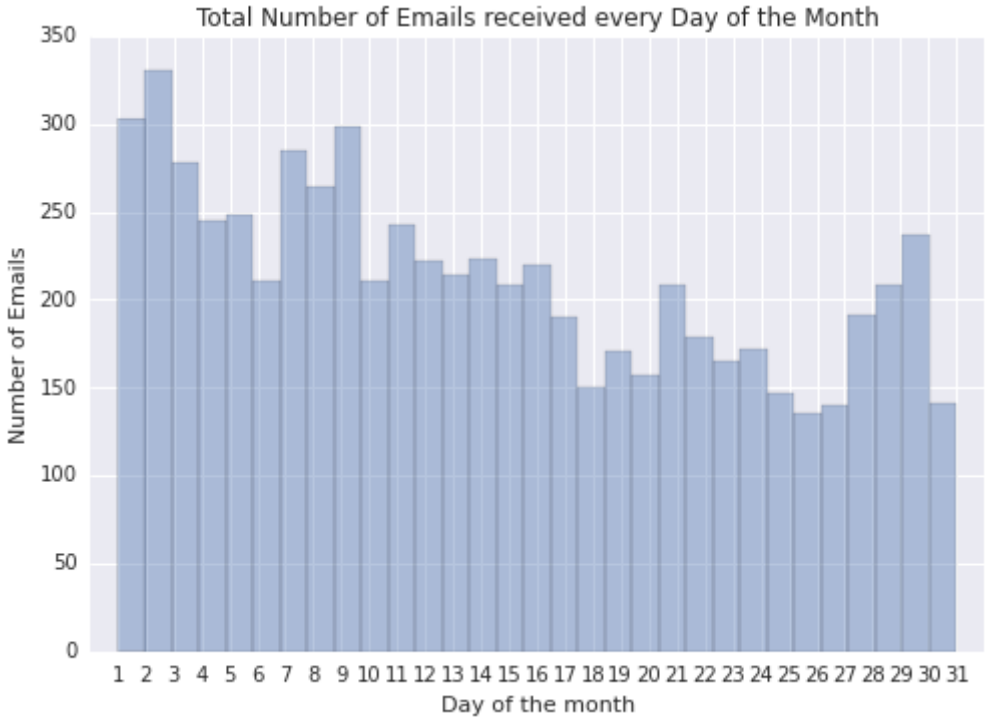
Day of the Week Histogram:



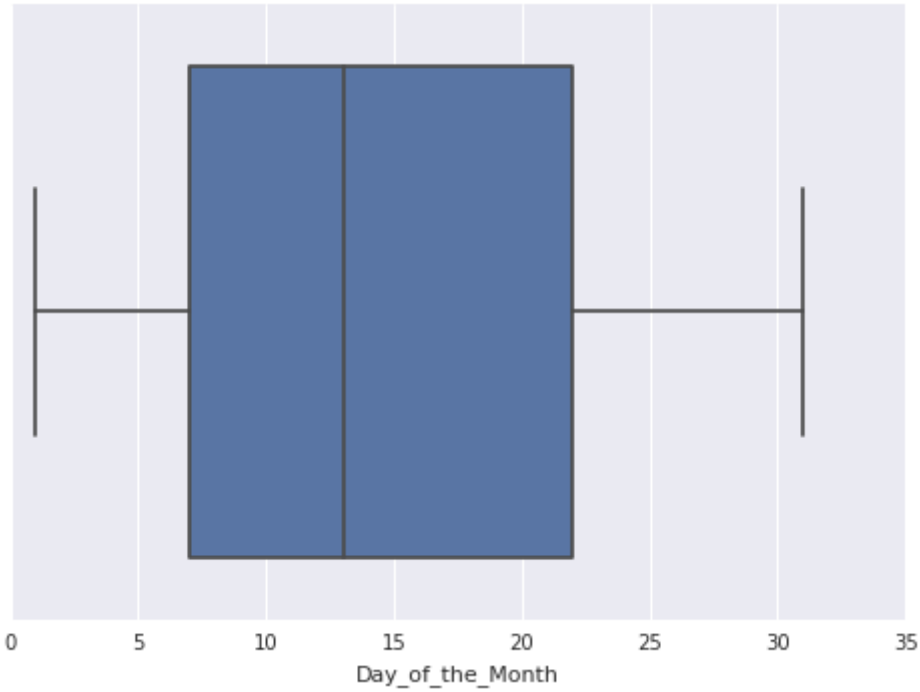
Day of the Week Box Plot:



Day of the Month Histogram:



Day of the Month Box Plot:



Testing my hypotheses

According to Chapter 9 of Think Stats (http://greenteapress.com/thinkstats2/html/thinkstats2010.html), a book that introduces probability and statistics concepts to python programmer, there are four steps to testing a hypothesis through classical hypothesis testing.

1. Choose a test statistic
2. Define a null hypothesis
3. Compute a p-value
4. Interpret the results

Hypothesis 1:

Group 1 is the Monday and Tuesday.

Goup 2 is Wednesday, Thursday and Friday.

Note: Saturday and Sunday are excluded because the weekend is not a part of the school week.

I hypothesized that group 1's average is higher than group 2's average.

Test statistic: The difference in means between group 1 and group 2.

Null Hypothesis: There is no difference in the means of group 1 and group 2.

P-value: 0.314

Interpretation: There is a 31.4% chance that we'll see a difference as big as the difference observed between group 1's and group 2's mean values (~110). The p-value is not less than 0.10, therefore the difference in emails received on each day of the week is not statistically significant.

Hypothesis 2:

Group 1 is defined as the 26th, 27th, 28th, 29th, 30th, 1st, 2nd, 3rd, 4th, and 5th days of a month.

Note: I'm excluding the 31st day of a month to simplify the calculation.

Group 2 is defined as the remaining days of a month.

I hypothesized that group 1's average is higher than group 2's average.

Test statistic: The difference in means between group 1 and group 2.

Null Hypothesis: There is no difference in the means of group 1 and group 2. Another interpretation is that the normal distribution of each group are the same.

P-value: 0.115

Interpretation: There is a 11.5% chance that we'll see a difference as big as the difference observed between group 1's and group 2's mean values (~25). The p-value is not less than 0.10, therefore the difference in emails received on each day of the month is not statistically significant.

Conclusion

Unfortunately, neither of my hypotheses were proven to be true. It appears that there is no difference between the amount of emails I receive on Mondays and Tuesdays and the rest of the school week. Similarly, it appears there's no difference between the amount of emails I receive in the beginning and ending days of a month and the rest of a month.

Last note: I chose to use Think Stat's DiffMeansOneSided class because I had reason to believe group 1's mean was larger than group 2's mean.