Abinash Koirala

Lunar cycles and Earthquakes: Are they correlated?

INTRODUCTION

Earthquakes are one of the most devastating natural hazards. Unlike other natural hazards, earthquakes don't give any signal prior to their occurrence. And earthquakes generally occur randomly. Although this area has been studied since long, predicting earthquakes with sufficient confidence is still a long shot. Earthquakes' magnitude are measured in a scale of 0.0 to 10.0 with 10.0 being the strongest and most devastating earthquake. Various research on predicting earthquakes based on data collected during previous earthquakes have been done till now[1][2]. Similarly, some research have been done to predict earthquakes based on other environmental phenomena like 'seismic electric signals'[3].

Lunar cycle are causal agent of an interesting phenomenon in earth's surface. The oceanic tides are bigger during new moon and full moon due to alignment of the moon, the earth and the sun which make the gravitational pull on earth maximum during that time[4].

I decided to explore the correlation between these two physical phenomena in this project. In other words I want to explore whether the effects like high tides exist for the land surface on earth and if that can relate to earthquakes. There have been past studies on this particular idea but most of them have resulted in conclusion that no correlation exist between these two phenomena[5]. And most of those research have been in conversational front. So, there is a lack of concrete data backed conclusion in this topic. My aim is to provide a data supported conclusion on the correlation of lunar cycles with earthquakes.

I pursued data analytics research on this topic based on initial hypothesis that the magnitude of earthquakes are correlated to the phases of moon and the magnitude of previous earthquakes. In addition to this hypothesis, correlation of lunar cycles and frequency of earthquakes is also explored.

So, for evaluating these hypotheses I'll use predictive modeling of earthquake dataset and lunar cycle time series to predict the frequency and magnitude of earthquakes in a particular location.

DATA DESCRIPTION

This analysis project required two datasets for the two physical phenomena that I was trying to correlate i.e, earthquakes and lunar cycles. The decision criteria for these two dataset were mostly related to location, timeframe and resolution. Based on these criteria I had various options as the data source for each of these.

For earthquake dataset, the primary requirement was the dataset should be at least a time series of earthquake events with magnitude and location of the event. Different organizations host time series data for earthquake events. United States Geological Survey (USGS) specializes in earthquake related data products. Similarly, Incorporated Research Institutions for Seismology (IRIS) is another organization that performs researches on earthquakes and they also provide earthquake datasets for research purpose.

While evaluating the data sources my primary decision criteria became ease of access. IRIS had a very comprehensive dataset on earthquake that can be extracted using their SeismiQuery API. They host a dataset web query system[6] that was both convenient to use and the query parameters were easily tunable. In my case the query parameters were timeframe, latitude and longitude range. I decided to limit the location to a region enclosing North California Fault lines. IRIS provided a query interface where I could provide a timeframe and range of latitude and longitude to get the earthquake events that occurred within that range. The dataset returned by the web query was in csv format which made importing into R data-frame lot more easier.

The dataset obtained contained following columns:

X.EventID, Time, Latitude, Longitude, Depth.km, Author, Catalog, Contributor, ContributorID, MagType, Magnitude, MagAuthor, EventLocationName

Among these parameters, the only parameters that I used in my analysis are Latitude, Longitude, Time, and Magnitude. For this analysis I extracted earthquake event data for date range 01/01/1990 to 12/31/2015. And the latitude and longitude was limited to a rectangle that encloses the state of California i.e,

North: 39

South: 37

West: -123

East: -121

For extracting dataset for this long time range, I used the query[] that specifies the above mentioned location and time frame. This returns a csv file that has all the recorded earthquake events and their relevant parameter in the format mentioned above. This date-range was

reduced eventually during further analysis due to the uneven distribution of data between date after and before 01/01/2007. My final analysis was done on the dataset from 01/01/2007 to 12/31/2015.

For the lunar phases dataset, US Navy has a web query service to return a dataset for the percent of illumination of moon for particular day[7][8]. The percent of illumination was representative of the moon phases since 1.0 illumination means Full moon and 0 illumination means New moon. This dataset just contained two columns: datetime and percent of illumination. The total number of observations was 3287, one for each day from 01/01/2007 to 12/31/2015.

Limiting the dataset to the earthquake events from Northern California Fault region was done to make the analysis as specific as possible. One of the criticisms of the previous research on this topic was the non specificity in location of the earthquake events. In addition to that this made sure that the effect of lunar phase is mostly uniform across all the event locations since that effect has potential to vary between places that are located far apart.

ANALYSIS

Transformation

The earthquake dataset required some transformation before I could proceed with exploring its statistical aspect. The key variables that I was going to analyse were time, latitude, longitude and magnitude. The time variable was in format “2014-01-01T15:47:16.460Z” so I had to extract two elements from this. The first one was the date part for which I used substr() method to extract the first 10 characters and casted that substring into date-type object.

As for the moon phase dataset, it was initially in the format of a two dimensional table with each row representing the day and each column representing the month for each year. There were 9 such tables for each year from 2007 to 2015. So, initially I used manual transformation of these tables to obtain a single dataframe containing date and percent illumination for each row. And I also had to transform the date column to date-type object for which I used as.Date() method to cast it into date format.

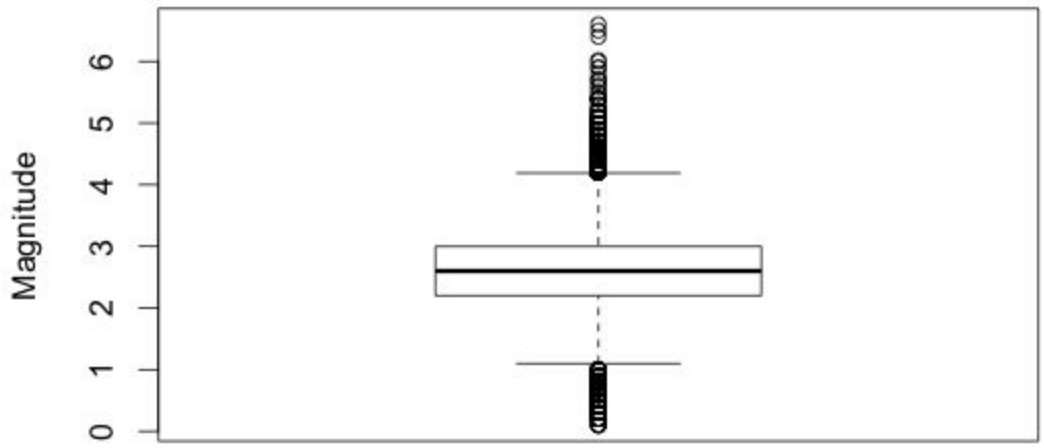
Finally, I had to join the two dataset so that I have for each row of event in earthquake dataset, the percent of illumination of moon for that day too. For this I used join() method from ‘plyr’ package like: join(earthquake.df, moonphase.df, by='day', type='left', match='all'). After this the dataset was ready for further analysis.

Cleaning

For cleaning of the dataset, I excluded the rows in which any of the key variables were absent using NA filtering. But there were no rows with missing magnitude or percent of illumination of moon so no rows were removed.

Analysis

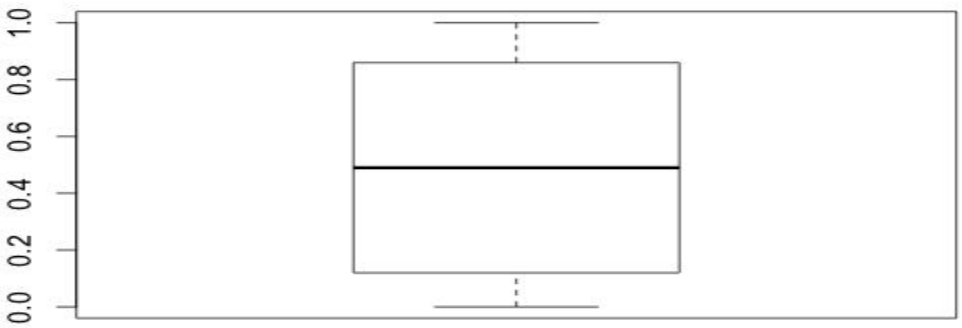
Moving on to the statistical analysis of the dataset, First thing that I did was to boxplot the magnitude variable and percent variable, which were the two continuous variables in the dataset:



Boxplot of Magnitude

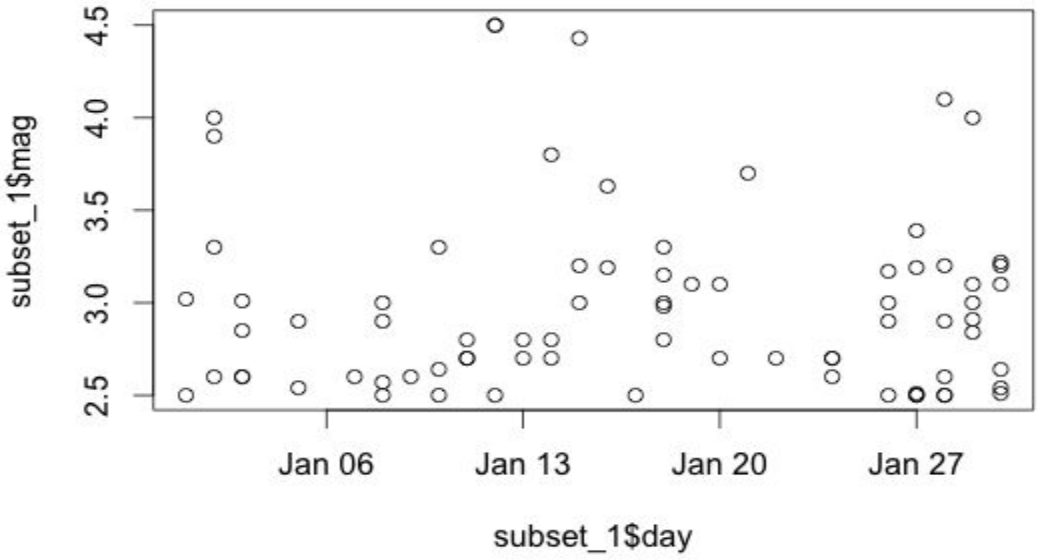
The distribution of magnitude was concentrated towards the range 2 to 3. There were significant number of outliers shown by the boxplot. But those were natural outliers so I didn't filter them out.

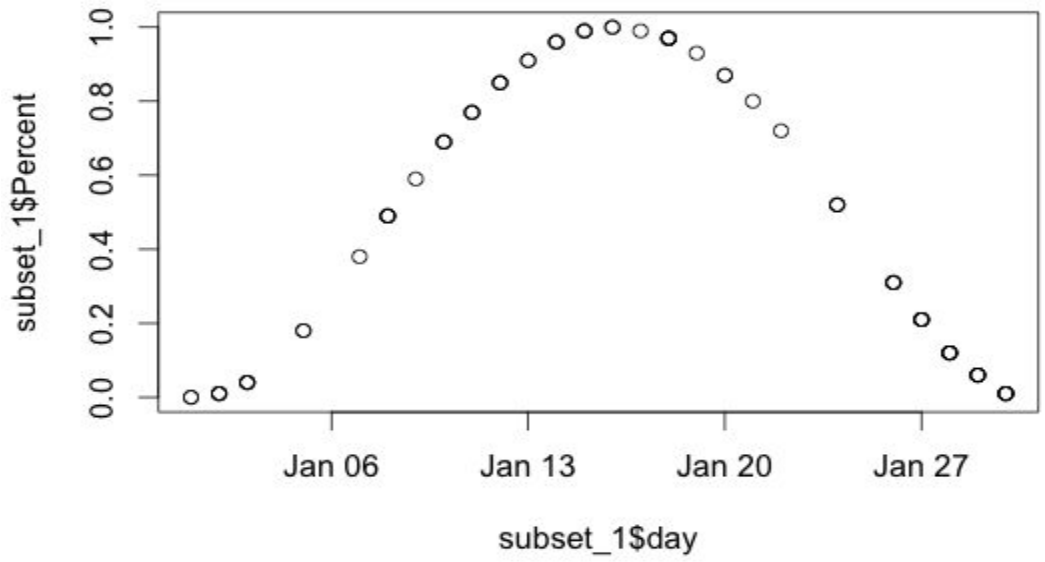
Similarly, the boxplot of 'Percent' variable was as follows:



To explain this perfect distribution of percent, we have to recall that the percentage of illumination of moon at particular day is a symmetric mathematical function so the distribution was perfectly symmetric and well structured.

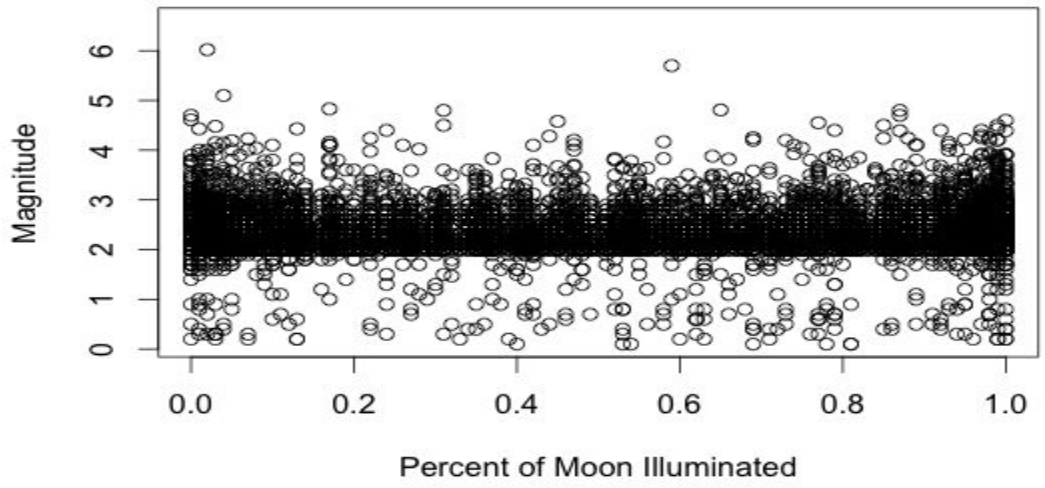
I plotted the same two variables for a date range i.e, for 01/01/2014 to 01/31/2014 :



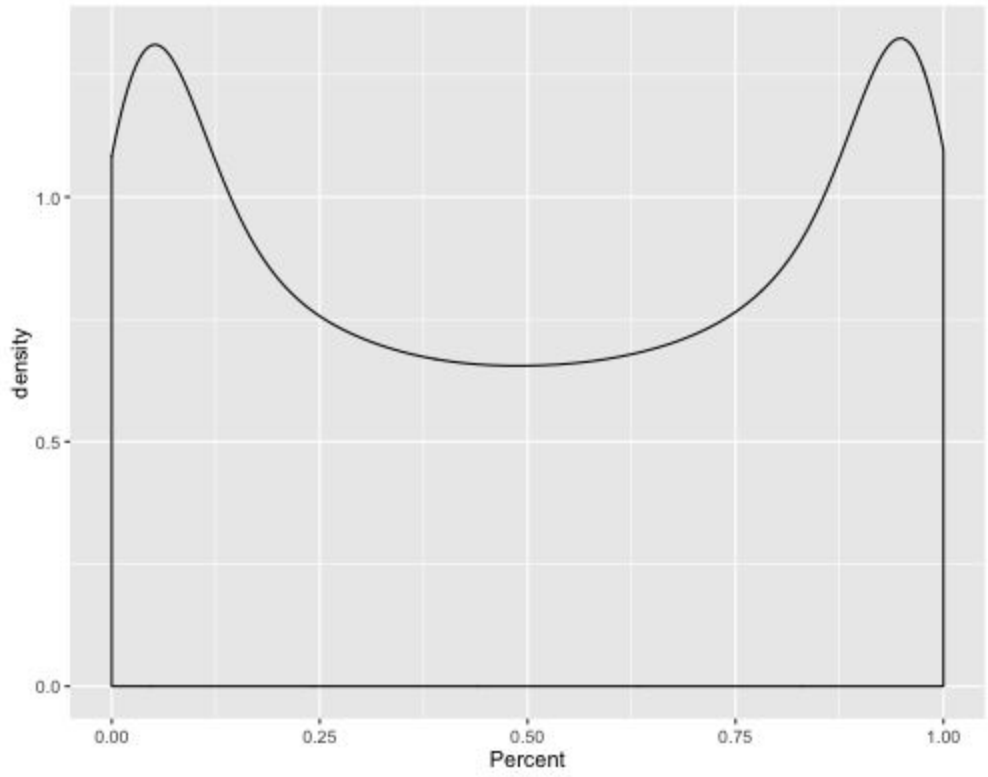


The plot for magnitude didn't show any particular pattern but the plot for percent illumination shows a symmetric seasonality. This is obvious since the moon's illumination is a function of moon cycle which follows an uniform cyclic pattern and that reflects in this plot.

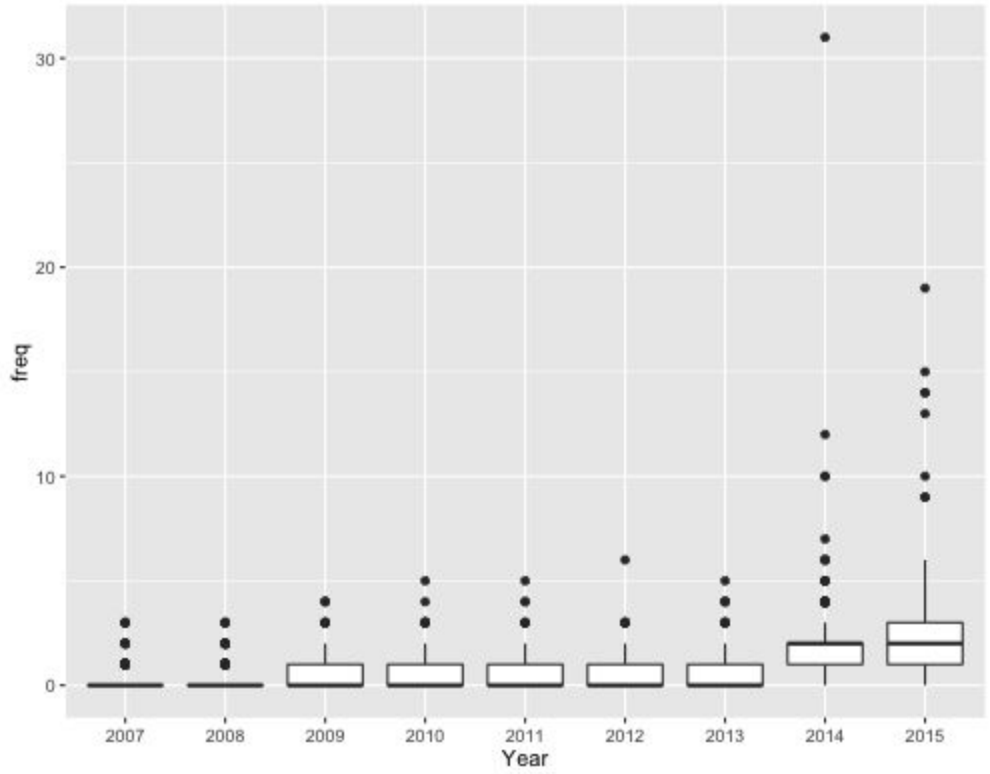
After this I plotted the magnitude against percent which is one of the relationships that I want to explore in this analysis:



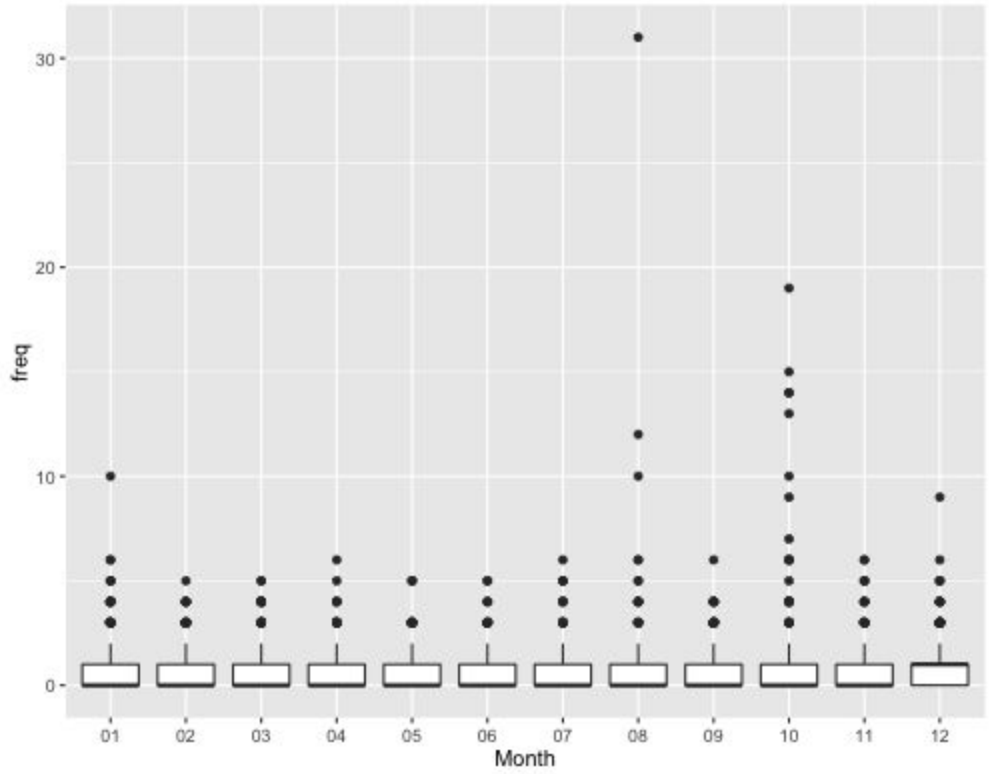
The magnitude doesn't show any distinct correlation with percent illumination. But there seems to be a hint of higher frequency of events around the region of 0.0 and 1.0 percent illumination. These two percent illumination represent the new and full moon, so this observation is infact an encouraging finding. And this led to my alternative approach for my analysis. We can see if the frequency of earthquakes by day is related to the moon illumination percent. So, to explore this I plotted the frequency of earthquake against the independent variables:



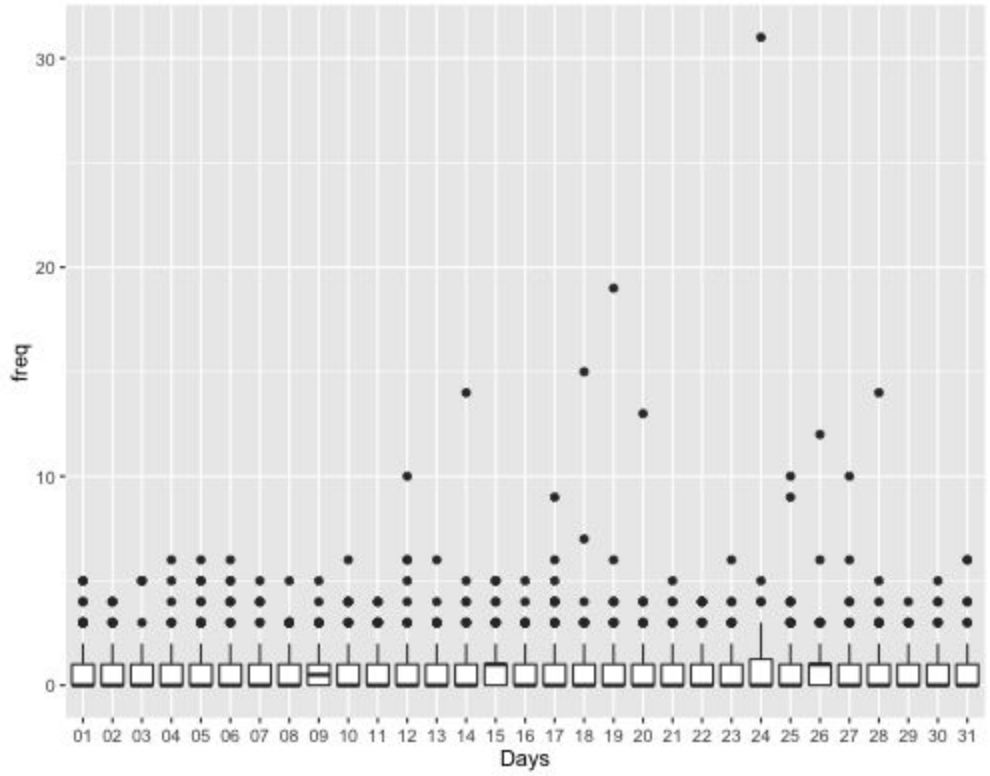
This plot shows the density of earthquake events against the percent of moon illuminated. This clearly shows that the earthquake events are more frequent when the Moon Illumination percent is near 0.0 or 1.0.



This is the year vs frequency plot. It shows that the frequency on earthquake per day is generally higher in 2014 and 2015 than in previous years. This can be useful during model development.



This is month vs frequency plot. This doesn’t show any specific pattern in earthquake frequency per day among the different months. Except for month 12 where the mean shown by the box plot is different than other months.



This day vs frequency plot also doesn't show any interesting pattern. But if we examine each box plots minutely then we can see that some days have mean frequency higher than the mean frequency for other days. This may be useful during model development.

MODEL DEVELOPMENT

Approach 1: Frequency prediction(Regression)

For predicting the frequency of earthquake for a particular day, I started with Poisson's regression model. For this I used the General Linear Model with parameters "family=poisson" and "link=log". The reason behind choosing this model was because my dependent variable was essentially a count data so it made sense to use poisson's regression model to predict count data.

First variation for this model was:

```
northcali.poiss_model <- glm(freq~1+Percent, data=northcali.date_agg,
family=poisson(link=log))
```

I.e, just using Percent Illumination for the predictor

The summary stats of this model revealed that the Percent variable was not at all significant than the Intercept.

```
> summary(northcali.poiss_model)

Call:
glm(formula = freq ~ 1 + Percent, family = poisson(link = log),
    data = northcali.date_agg)

Deviance Residuals:
    Min      1Q  Median    3Q       Max
-1.2787  -1.2575  -1.2307   0.2576  12.8485

Coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept) -0.20150    0.03383  -5.955 2.59e-09 ***
Percent          -0.07959          0.05598  -1.422          0.155
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for poisson family taken to be 1)

    Null deviance: 5470.2  on 3286  degrees of freedom
Residual deviance: 5468.2  on 3285  degrees of freedom
AIC: 8862.5

Number of Fisher Scoring iterations: 6
```

After this I tried adding more variables into the model.

```
northcali.poiss_model2 <- glm(freq~Days+Year+Month, data=northcali.date_agg,
family=poisson(link=log))
```

The summary stats for this model:

```
> summary(northcali.poiss_model2)

Call:
```

```
glm(formula = freq ~ Days + Year + Month, family = poisson(link = log),
    data = northcali.date_agg)

Deviance Residuals:
    Min      1Q  Median    3Q      Max
-2.4355  -0.9780  -0.7121  0.5425  9.4124

Coefficients:
             Estimate Std. Error z value Pr(>|z|)
(Intercept) -1.20243    0.15856  -7.584 3.36e-14 ***
Days02      -0.11248    0.15014  -0.749 0.453775
Days03      -0.29480    0.15787  -1.867 0.061857 .
Days04      -0.29480    0.15787  -1.867 0.061857 .
Days05      -0.04349    0.14748  -0.295 0.768100
Days06      -0.16127    0.15211  -1.060 0.289059
Days07      -0.22581    0.15483  -1.458 0.144720
Days08      -0.23923    0.15541  -1.539 0.123716
Days09      -0.07739    0.14877  -0.520 0.602941
Days10      -0.17385    0.15263  -1.139 0.254706
Days11      -0.18659    0.15316  -1.218 0.223140
Days12       0.01058    0.14548   0.073 0.942014
Days13      -0.32379    0.15920  -2.034 0.041966 *
Days14       0.01058    0.14548   0.073 0.942014
Days15       0.01058    0.14548   0.073 0.942014
Days16      -0.35364    0.16059  -2.202 0.027658 *
Days17      -0.05466    0.14790  -0.370 0.711707
Days18       0.10110    0.14232   0.710 0.477477
Days19      -0.06596    0.14833  -0.445 0.656559
Days20      -0.03244    0.14706  -0.221 0.825439
Days21      -0.17385    0.15263  -1.139 0.254706
Days22      -0.25284    0.15600  -1.621 0.105080
Days23      -0.33860    0.15989  -2.118 0.034195 *
Days24       0.23583    0.13799   1.709 0.087449 .
```

Days25	0.08168	0.14298	0.571	0.567817
Days26	0.07183	0.14332	0.501	0.616250
Days27	-0.08895	0.14922	-0.596	0.551116
Days28	-0.06596	0.14833	-0.445	0.656559
Days29	-0.29732	0.16140	-1.842	0.065460 .
Days30	-0.11819	0.15382	-0.768	0.442253
Days31	-0.07309	0.17147	-0.426	0.669938
Year2008	0.09320	0.14564	0.640	0.522221
Year2009	0.72594	0.12840	5.654	1.57e-08 ***
Year2010	0.75246	0.12785	5.885	3.97e-09 ***
Year2011	0.69869	0.12898	5.417	6.06e-08 ***
Year2012	0.79139	0.12703	6.230	4.67e-10 ***
Year2013	0.78846	0.12713	6.202	5.57e-10 ***
Year2014	2.02815	0.11213	18.088	< 2e-16 ***
Year2015	2.12691	0.11152	19.073	< 2e-16 ***
Month02	-0.16097	0.09769	-1.648	0.099385 .
Month03	-0.07315	0.09281	-0.788	0.430591
Month04	-0.24149	0.09816	-2.460	0.013884 *
Month05	-0.37885	0.10104	-3.749	0.000177 ***
Month06	-0.20923	0.09727	-2.151	0.031473 *
Month07	-0.22730	0.09674	-2.350	0.018787 *
Month08	0.04066	0.09019	0.451	0.652131
Month09	-0.20396	0.09713	-2.100	0.035739 *
Month10	0.20556	0.08676	2.369	0.017824 *
Month11	-0.10387	0.09451	-1.099	0.271748
Month12	0.05647	0.08984	0.629	0.529653

Signif. codes: 0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for poisson family taken to be 1)

Null deviance: 5470.2 on 3286 degrees of freedom

Residual deviance: 3897.0 on 3237 degrees of freedom

AIC: 7387.3

Number of Fisher Scoring iterations: 6

Both of these models suggest that none of the independent variables are significant for prediction. After these two unsuccessful models, I concluded that variant's of linear models might not be useful for this dataset. And may be I should predict the occurrence rather than frequency. So, for this I added a new variable Class which was TRUE if earthquake occurred in that particular day and FALSE if it didn't.

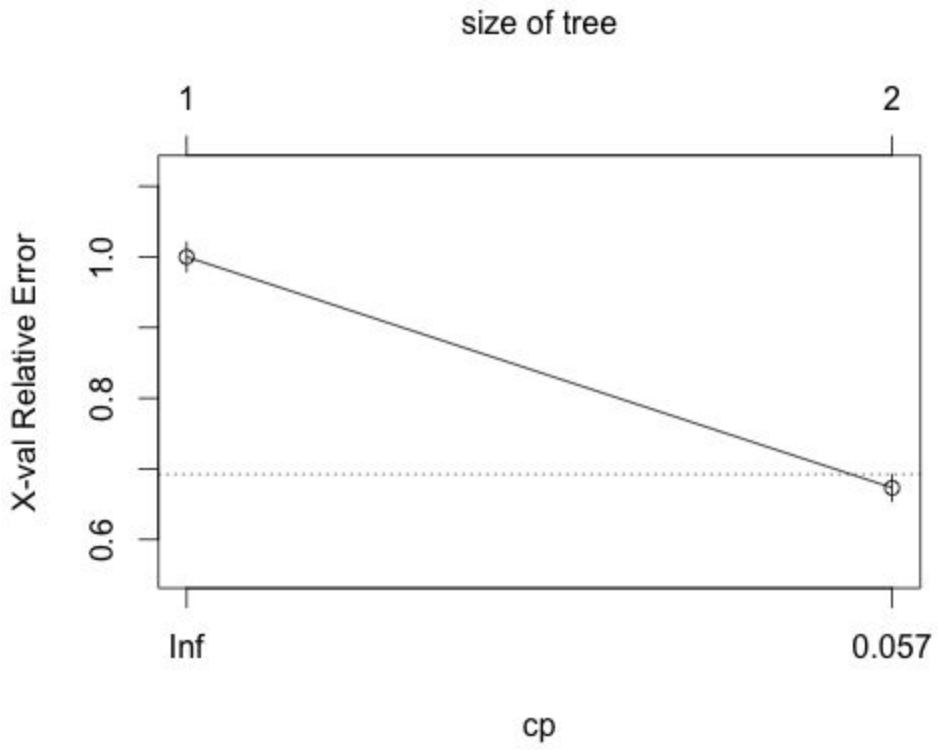
Approach 2: Occurrence Prediction(Classification)

For this approach I partitioned the dataset into training and testing sets with 90:10 ratio randomly.

The first model that I used for this approach was Recursive partitioning tree model. I started with all the independent variables as predictors.

```
northcali.model1 <- rpart(Class~Year+Month+Days+Percent, data=northcali.date_agg.training, method = "class")
plotcp(northcali.model1)
printcp(northcali.model1)
northcali.pred1 <- predict(northcali.model1, northcali.date_agg.testing)
northcali.pred1$Class <- as.factor(northcali.pred1[,1] < northcali.pred1[,2])
mean(northcali.pred1$Class == northcali.date_agg.testing$Class)
```

Model fits were observed as follows:




```
> printcp(northcali.model1)

Classification tree:
rpart(formula = Class ~ Year + Month + Days + Percent, data = northcali.date_agg.training,
        method = "class")

Variables actually used in tree construction:
[1] Year

Root node error: 1288/2959 = 0.43528

n= 2959

        CP nsplit rel error  xerror       xstd
1 0.32686      0   1.00000 1.00000 0.020939
2 0.01000      1   0.67314 0.67314 0.019222
```

And I calculated the accuracy of the prediction and found it to be 0.70. Which was not very good accuracy for two class classification. And the model stats revealed that the variable Year was the only variable used in the tree construction. So I created another model with just Year as the input to verify the accuracy change with predictor variable set.

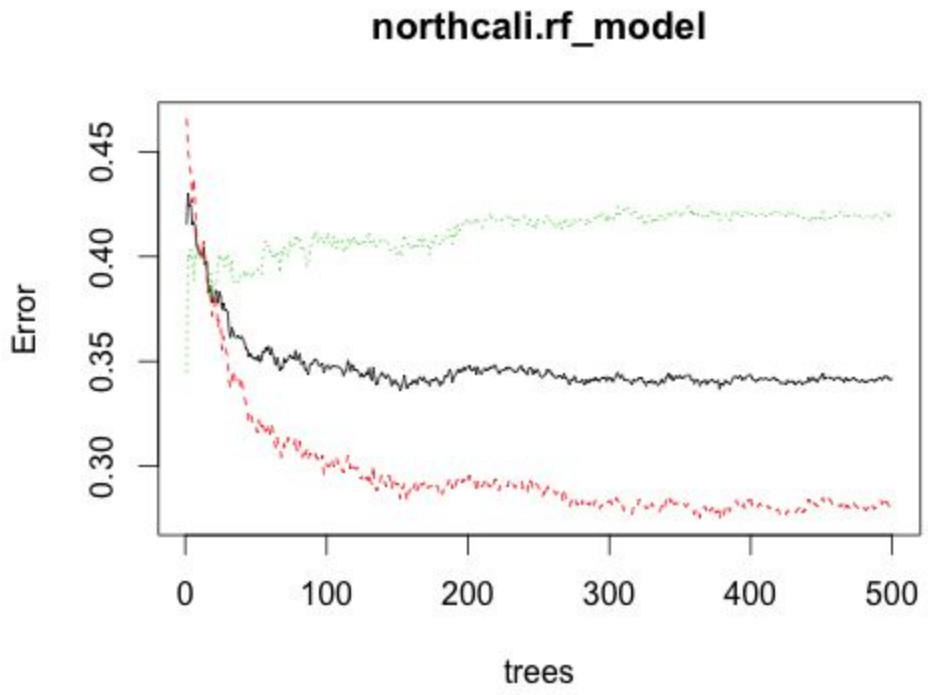
```
northcali.model2 <- rpart(Class~Year, data=northcali.date_agg.training, method = "class")
plotcp(northcali.model2)
printcp(northcali.model2)
northcali.pred2 <- predict(northcali.model2, northcali.date_agg.testing)
northcali.pred2$Class <- as.factor(northcali.pred2[,1] < northcali.pred2[,2])
mean(northcali.pred2$Class == northcali.date_agg.testing$Class)
```

The plots and stats for this single predictor model was also similar to the first model and the accuracy was also same 0.70. And the tree was with just one node so it seems this model is not appropriate and too general.

Now, I moved on to randomForest model. For random forest model I used the same set of training and testing dataset. Firs approach was to feed in all the variables and then start pruning the tree.

```
northcali.rf_model <- randomForest(Class~Year+Month+Days+Percent,
data=northcali.date_agg.training)
plot(northcali.rf_model)
varImpPlot(northcali.rf_model)
northcali.rf_pred <- predict(northcali.rf_model, northcali.date_agg.testing)
northcali.pred1
mean(northcali.rf_pred == northcali.date_agg.testing$Class)
```

The model error plot looked like this:



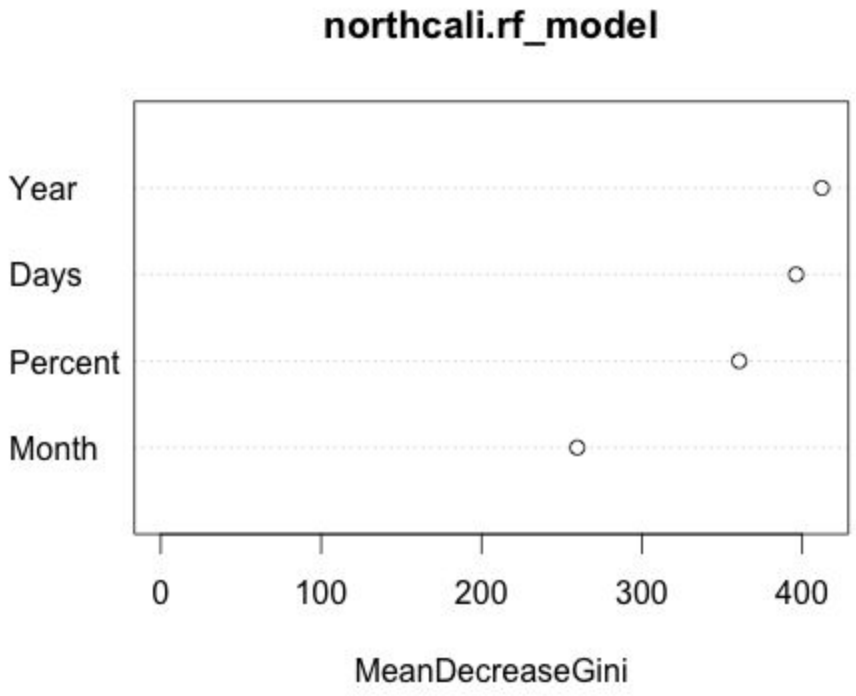

```
> print(northcali.rf_model)

Call:
 randomForest(formula = Class ~ Year + Month + Days + Percent,           data =
northcali.date_agg.training)
          Type of random forest: classification
                Number of trees: 500
No. of variables tried at each split: 2

          OOB estimate of  error rate: 34.2%
Confusion matrix:
          FALSE TRUE class.error
FALSE  1199  472   0.2824656
TRUE    540  748   0.4192547
```

The error rate for TRUE class was higher than the error rate for FALSE class. This is obvious since the dataset contained mostly FALSE class and this plot can't indicate whether this error

rate is due to the distribution of the decision class or is it actually due to classification accuracy. And the variable importance plot looked like this:

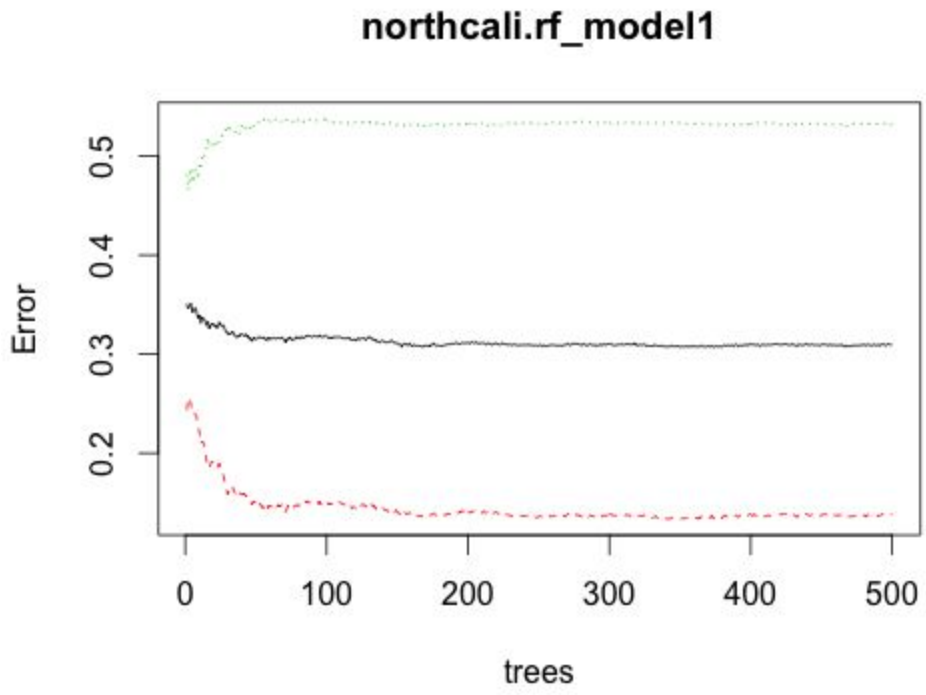


This showed that the most important variable for the prediction was Year, which was also the conclusion that I got from rpart model. But this model also used Days significantly for prediction. And the accuracy of this model came out to be 0.66 for the test set, which was, like the rpart model not very good for two class classification.

I further developed another random forest by excluding the Percent and Month variables:

```
northcali.rf_model1 <- randomForest(Class~Year+Days, data=northcali.date_agg.training)
plot(northcali.rf_model1)
varImpPlot(northcali.rf_model1)
northcali.rf_pred1 <- predict(northcali.rf_model1, northcali.date_agg.testing)
mean(northcali.rf_pred1 == northcali.date_agg.testing$Class)
```

The plots for this model were as follows:



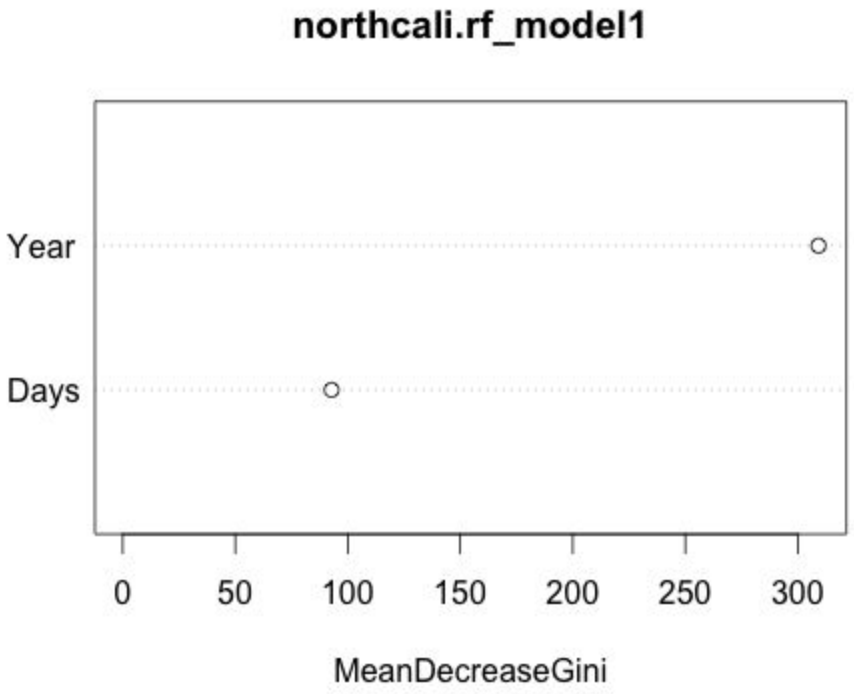

```
> print(northcali.rf_model1)

Call:
 randomForest(formula = Class ~ Year + Days, data = northcali.date_agg.training)
               Type of random forest: classification
                     Number of trees: 500
No. of variables tried at each split: 1

               OOB estimate of  error rate: 30.96%
Confusion matrix:
       FALSE TRUE class.error
FALSE  1440  231   0.1382406
TRUE    685   603    0.5318323
```

This plot shows a very interesting thing about this model. The overall error rate is around 0.3 but the error rate for TRUE class is 0.5 while the error rate for FALSE class is less than 0.2. And this was different from the previous model because from the previous model this model improved the accuracy for FALSE class but it decreased the accuracy for TRUE class. Which

indicates this model is better for predicting the non occurrence of earthquake than for predicting the occurrence of earthquake.



And the variable importance plot shows the same pattern as the previous model showed.
The good part about this model is its accuracy is 0.70 which is the same as the rpart model.

CONCLUSION AND DISCUSSION

The models development indicated that prediction of non-occurrence of earthquake is more accurate than predicting the occurrence of the earthquake.

To recall my analysis, I started with a motivation to predict the magnitude of earthquakes using the moon phase data for the particular day. But after initial exploratory analysis of the dataset magnitude seems to be a variable that didn't follow any significant pattern with any of the independent variable. So, I had to look for another relationship to explore.

Next relationship that I tried to explore was the frequency of earthquake for a particular day from the date (and it parts like Year, Month, Day) and mon illumination. This was also unsuccessful because the model that I developed couldn't predict the frequency with sufficient accuracy.

Finally, I decided to predict the occurrence of earthquake for particular date based on the independent variables Year, Month, Day parts of the date and Moon Illumination Percent for that particular day. For this I used two tree based models rpart and randomForest. This analysis revealed that Moon illumination percent is not significant for predicting the occurrence of earthquake. But the non-occurrence of earthquake can be predicted being based only on The Year and Day parts of the date.

As a concluding remark earthquake magnitude, frequency and occurrence are not related to Lunar Phases. Neither are they correlated nor can Lunar Phase be used to predict the earthquakes or their characteristics.

REFERENCES:

1. https://en.wikipedia.org/wiki/Earthquake_prediction#M8
2. Hough, Susan (2010b), *Predicting the Unpredictable: The Tumultuous Science of Earthquake Prediction*, Princeton University Press, ISBN 978-0-691-13816-9.
3. Varotsos, Alexopoulos & Nomicos 1981, described by Kagan 1997b, §3.3.1, p. 512, and Mulargia & Gasperini 1992, p. 32.
4. http://earthsky.org/earth/tides-and-the-pull-of-the-moon-and-sun
5. http://news.nationalgeographic.com/news/2005/05/0523_050523_moonquake.html
6. IRIS earthquake query url for 2010-01-01 to 2015-12-31 in California Max Lat: 41.886, Min Lat: 32.528, Min Lon: -124.629, Max Lon: -114.170 link: http://service.iris.edu/fdsnws/event/1/query?starttime=2000-01-01T06:30:00&endtime=2015-12-31T06:30:00&minmag=0&maxmag=10&includeallorigins=true&orderby=time&format=text&maxlat=41.886&minlon=-124.629&maxlon=-114.170&minlat=32.528&nodata=404
7. http://aa.usno.navy.mil/cgi-bin/aa_moonill2.pl?form=1&year=2015&task=00&tz=+00
8. http://aa.usno.navy.mil/cgi-bin/aa_moonill2.pl?form=1&year=2014&task=00&tz=+00
9. http://service.iris.edu/fdsnws/event/docs/1/builder/
10. https://en.wikipedia.org/wiki/Hayward_Fault_Zone#/media/File:122-38HaywardFault.jpg