Solar Data

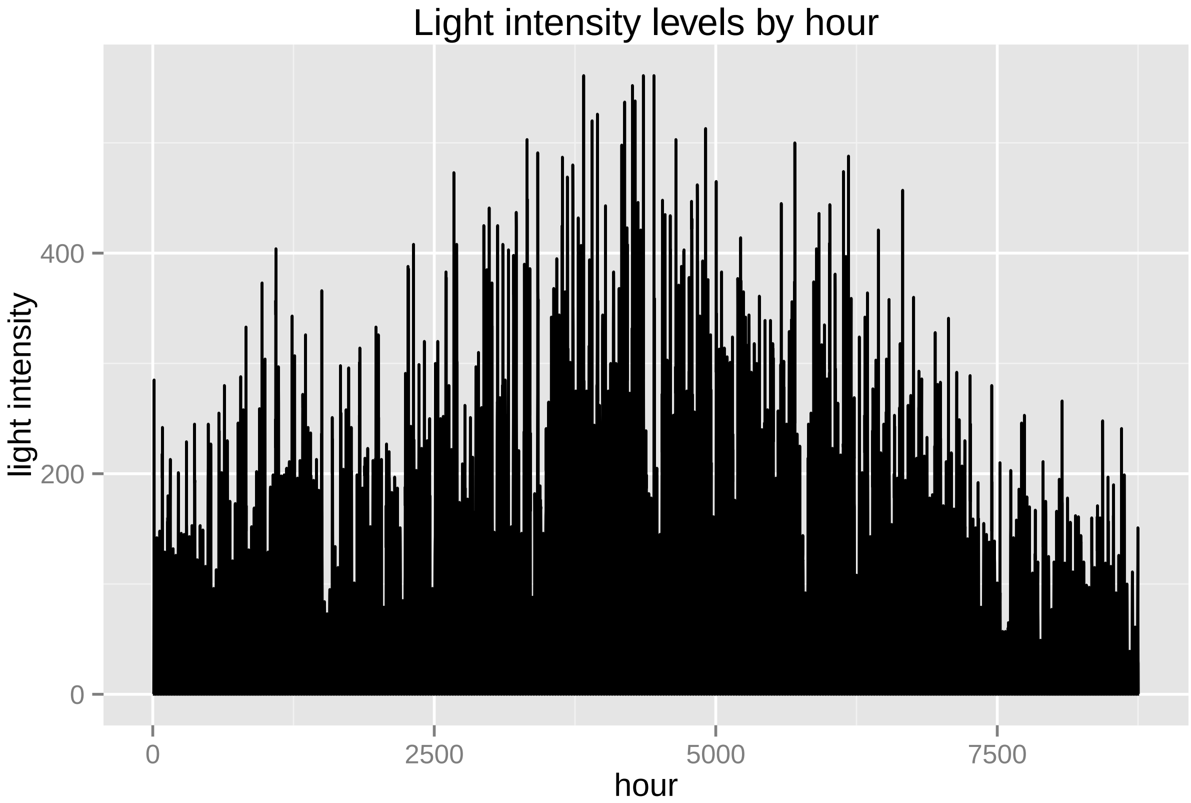
Nick Strayer

March 8, 2015

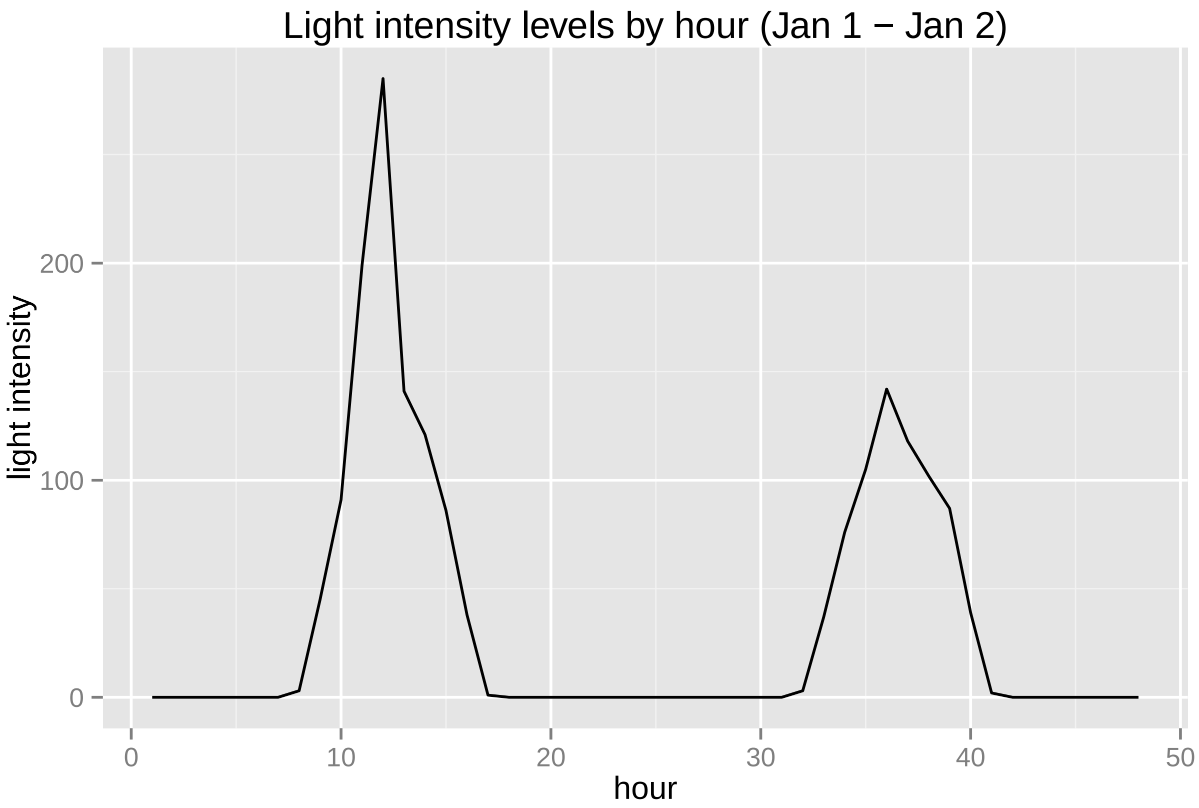
First we bring in raw data obtained from National Renewable Energy Laboratory

```
setwd("/Users/Nick/mysisModeling")
solarRaw = read.csv("outsideData/solarData2010.csv")
realSolar = solarRaw$METSTAT.Dif..Wh.m.2.
```

and run a quick check to make sure the data came in properly…



Taking a closer peak at the data we see that the sensor is not actually accurate enough to pick up moonlight…



Adding the lunar cycle.

Because of this sensor limitation and the importance of the moon cycle on migration patterns we need to find a way to fill in the missing moon data.

By treating the lunar light intensity as a sinusoidal curve with a period of 27 days and a maximum light intensity of 1 lux (as per this PNAS paper) we can model the moon cycle and then fill in the gaps in the real data with the modeled data.

```
hours = 1:(365*24) # Generate an hours vector
cycle = 27*24 #27 days at 24 hours.

moonLux = NULL #initialize list of moon values

for (hour in hours){ #Loop through the hours of the day to generate data.
  cyclePoint = hour %% cycle
  moonLux = c(moonLux, 0.5 * cos((1/cycle)* 2*pi * cyclePoint ) + .501)
}
```

Combined the data:

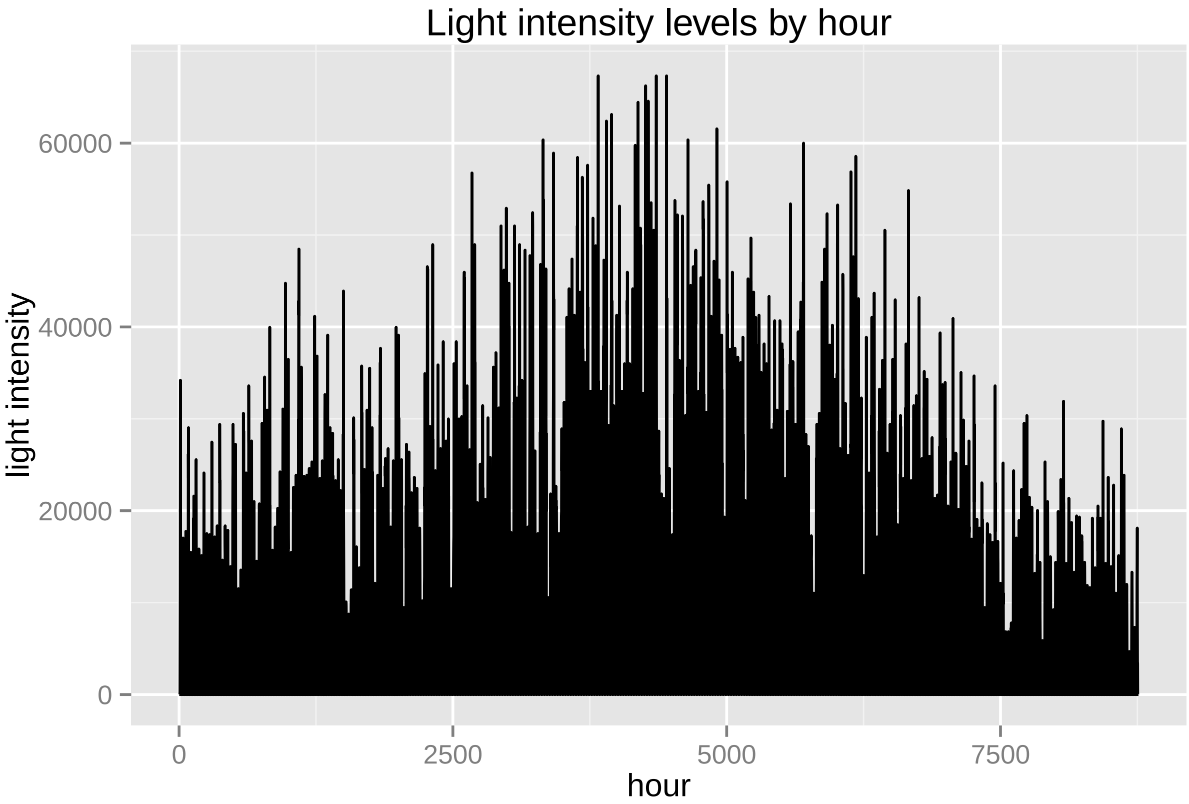
Now that we have the real data in and the modeled lunar cycle we need to combine them to get hour complete solar data.

We are using a conversion ratio of 120 Watt Hours (wh) to a lux as was reported in Jensen et al. 2006 (see papers directory).

```
Wh_to_lux = 120.0; #From Jensen et all.
combinedLight = NULL;
solarLux = NULL;

for (i in hours){
  solarLux = realSolar[i] * Wh_to_lux;
  if (solarLux > moonLux[i]){
    combinedLight = c(combinedLight, solarLux)
  } else {
    combinedLight = c(combinedLight, moonLux[i])
  }
}
```

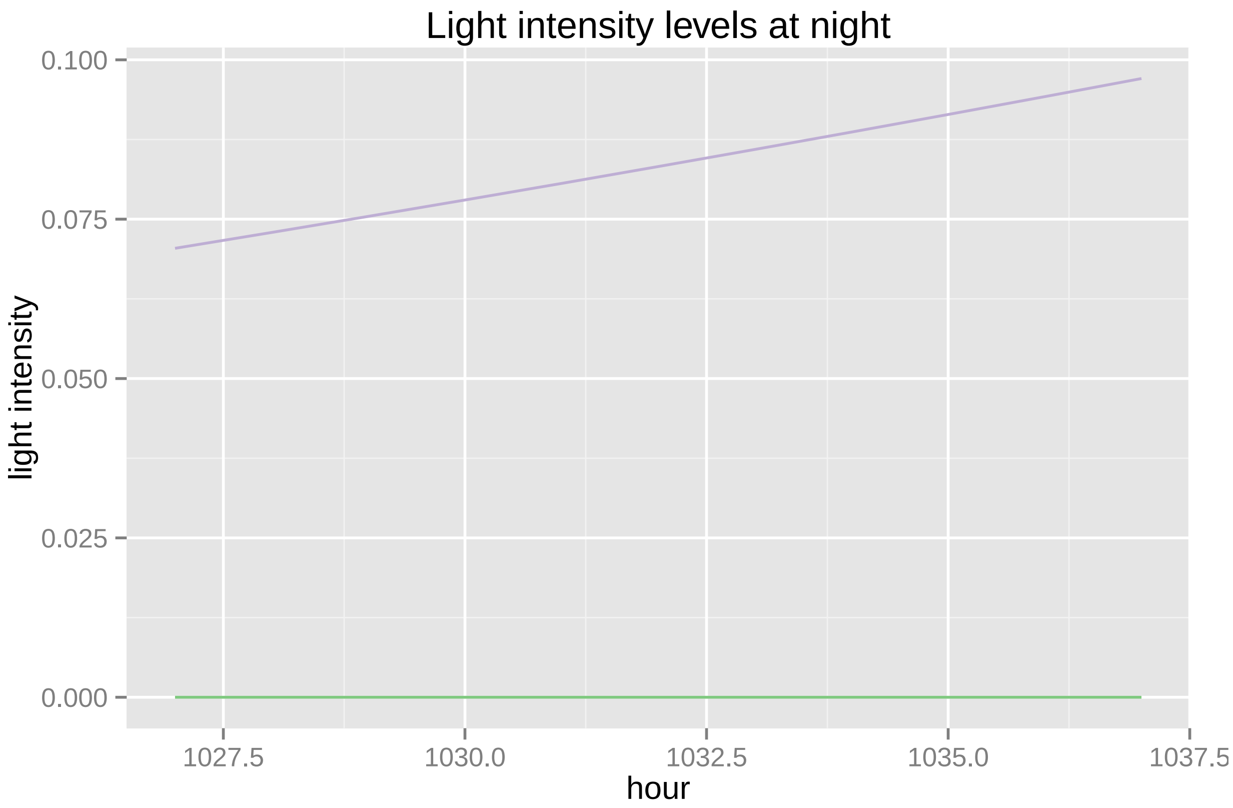
Now we can plot to see if the lunar data successfully made it in:



Well it's pretty hard to tell if anything happened. However, take note of the y-axis. Pretty hard to note a change of < 1 lux when it goes up to 60,000.

To check one more time lets zoom in on a given night. In this case between the hours of 1027 and 1037 (a february night).

We can see that the new (purple) data doesn't touch zero unlike the raw (green) data.



Now we can export this data to the data directory.

```r
setwd("/Users/Nick/mysisModeling")
write.csv(combinedLight, "data/light_day_moon_hour.csv", row.names=FALSE)
```