For each of the 8 streets and directions, your 95% confidence interval of the number of cars that drive past per minute at the time of day when you counted cars.

Metadata:

A photo of the intersection where you gathered data.

A video by which the data was generated can be found on: [https://www.youtube.com/watch?v=BE6MNC-YAPE (https://www.youtube.com/watch?v=BE6MNC-YAPE)](https://www.youtube.com/watch?v=BE6MNC-YAPE)

The GPS coordinations of the intersection where you gathered data.

37.492058, 127.032428

The names of the streets at the intersection where you gathered data.

The streets I modeled were *Yeoksam-ro 4-gil* and *Gangnam-daero 66-gil* as can be seen in the image bellow:

Metadata for each of the 8 data sets you produced:

While the traffic supposed to follow the the arrows drawn by Google as in the image, in reallity it can be seen in the video that drivers drove from and towards any direction, since the rules are not strictly followed (apparently, unidirectional streets can be bidirectional for motorcycles, according to the Seoulite's interpretation). The azure arrows depict illegal trafic observed. The Star depicts my position. The data were collected during 5 minutes interval, on Friday October 27 21:05 .

Modeling assumptions:

I used the Poisson Distribution , which " is a discrete probability distribution that expresses the probability of a given number of events occurring in a fixed interval of time and/or space if these events occur with a known constant rate and independently of the time since the last event."(Wikipedia, 2017) This would make sense as Poisson's support is

$$k \in \mathbb{N} \cup 0$$

.

Priors: Since the likelihood is composed of Poisson Distribution, it only makes sense to have a prior composed of the Gamma Distribution. Therefore we can use the conjugate prioir and easily compute the posterioir.

I chose the priors based on the following assumptions:

1. There is a club quite close to the intersection (~20 meters towards the "back" direction). I would expect cars to go towards that direction.
2. I would expect drivers to follow the traffic rules; in Korea I'd even assume almost no violations.
3. For an ordinary day(i.e. a day which is not Halloween) I would expect less traffic.

As it is common in Bayesian statistics, the confidence interval I am using is the credible interval.

Notice the difference:

"they differ on a philosophical basis; Bayesian intervals treat their bounds as fixed and the estimated parameter as a random variable, whereas frequentist confidence intervals treat their bounds as random variables and the parameter as a fixed value" (Wikipedia, 2017).

"A frequentist 95% confidence interval means that with a large number of repeated samples, 95% of such calculated confidence intervals would include the true value of the parameter.In frequentist terms, the parameter is fixed (cannot be considered to have a distribution of possible values) and the confidence interval is random (as it depends on the random sample)."(wikipedia, 2017)

Since there is a bug in the code, the credible intervals are not computed correctly.

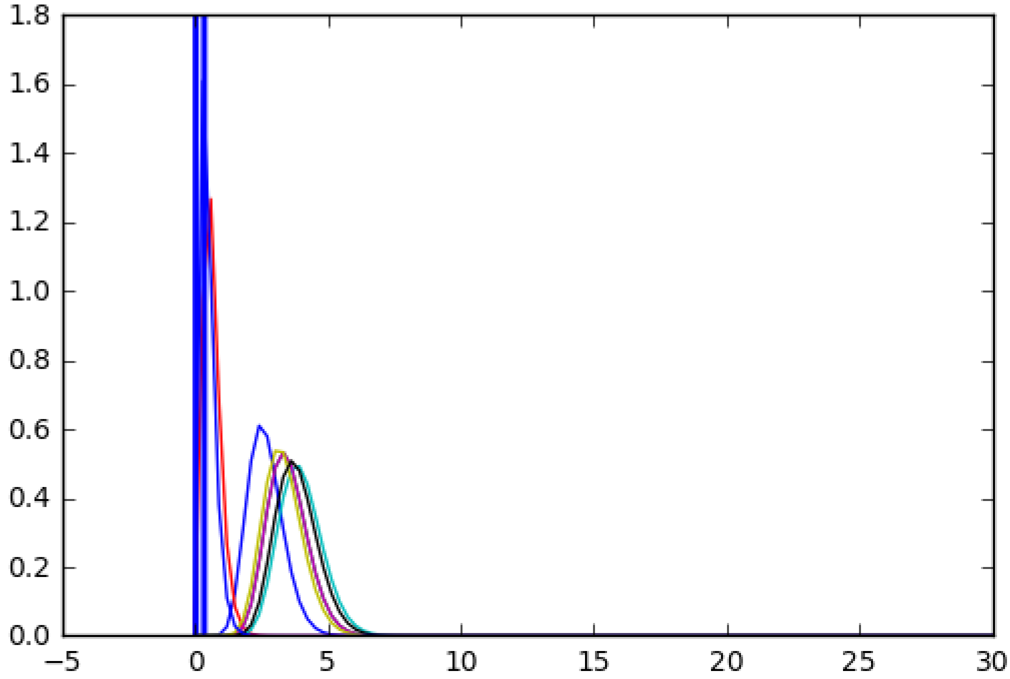
""

Resources: http://nptel.ac.in/courses/105101008/13 (http://nptel.ac.in/courses/105101008/13) https://stats.stackexchange.com/questions/123980/need-help-calculating-poisson-posterior-distribution-given-prior (https://stats.stackexchange.com/questions/123980/need-help-calculating-poisson-posterior-distribution-given-prior)

code

In [40]:

```
import numpy as np
from numpy.random import normal
import matplotlib.pyplot as plt
from scipy.stats import gamma
from scipy.stats.mstats import mquantiles

#For convenience I'll use easier terms (left,right back and front) that can be
#the directions are considered given the camera's position in the Youtube video
import numpy as np
Data = {"Left_to_back":1,"Left_to_right":6,"Left_to_front":2,"Front_to_left":4,
                  "Front_to_right":3, "Front_to_back":7, "Right_to_left":6, "Rig
                 "Back_to_right":1, "Back_to_left":1}

#I built this dataset to allow for a more in-depth analysis later,
#should I want to further explore it. The original task however, is looking for
#less detailed data that I'm constructing here:

from_left = sum([Data["Left_to_back"],Data["Left_to_right"],Data["Left_to_front"
from_right = sum([Data["Right_to_back"],Data["Right_to_left"]])
from_back = sum([Data["Back_to_left"],Data["Back_to_right"],Data["Back_to_front"
from_front = sum([Data["Front_to_back"],Data["Front_to_right"],Data["Front_to_le

to_left = sum([Data["Right_to_left"],Data["Front_to_left"],Data["Back_to_left"]
to_right = sum([Data["Front_to_right"],Data["Left_to_right"],Data["Back_to_right
to_back = sum([Data["Right_to_back"],Data["Left_to_back"],Data["Front_to_back"]
to_front = sum([Data["Left_to_front"],Data["Back_to_front"]])

"""
Deriving the parameters of the posterior:
p(λ|x)=βα+xλα+x−1e−λ(β+1)Γ(α+x)=Gamma(λ;α+x,β+1)
"""
prior_alpha = np.array([7,10,1,10,10,10,10,0]) # alpha will be lambda given tha
#see. For example, I expected (almost) no turns against the law (1, and 0 assig
prior_beta = np.ones(8) #beta is defined 1 here.
obs = np.array([from_left,from_right,from_back,from_front,to_left,to_right,to_ba


posterior_alpha = prior_alpha+obs #α+obs

posterior_beta = prior_beta+np.array([5,5,5,5,5,5,5,5]) #β + n (number of inter
zeros = np.zeros(8)
x = np.linspace(0,30,100)
#Lambda is the model parameter (average / expected number of cars in the interv
predictions = [gamma.pdf(x,a=(posterior_alpha[i]),scale=1/posterior_beta[i]) fo
Credibles = [gamma.pdf(x=[0.025, 0.975], a=(posterior_alpha[i]),scale=1/posteri

data_for_CredI = [gamma(x,posterior_alpha[i],posterior_beta[i]) for  i in range
# print(predictions)

for i,j in enumerate(predictions):
    #constructing Credible Intervals:
#   data=data_for_CredI[i]
    plt.axvline(mquantiles(j,prob=[0.5]))  #credible intervals
    plt.axvline(mquantiles(j,prob=[0.95])) #credible intervals
    plt.plot(x,j)
```


1.8
1.6
1.4
1.2
1.0
0.8
0.6
0.4
0.2
0.0
−5
0
5
10
15
20
25
30