**Week 4 Writeup**

# Week 3 Summary

1. **[MON]** Fit 88/101 distributions in Scipy (DISTRIBUTIONS.ipynb) and compare GOF using KS & MAE
2. **[TUE]** Weekly meeting & writeup. Filter out multi-robot data (FILTER.ipynb)
3. **[WED]** Use filtered data to fit 88 distributions in Scipy against 25 edges with most data. Select models with low KS & MAE (section 3 of MAIN3.ipynb)
   1. Lognorm, Powerlognorm, Invgauss, Mielke, Fatiguelife, Invgamma
4. **[THU]** Compare models (see Wk3 Writeup for more details)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Distribution** | **Parameters** | **MLE** | **Conjugate priors** | **Common use** | **Difficulty** |
| Lognorm | Mu, sigma | Simple | Transform lognormal to normal | If X is lognormal, then lnX is normal | Easiest |
| Powerlognorm |  | Hard | None that I could find |  | Hard & unsuitable |
| Invgauss (Wald) | Mu, lambda | Simple | Exists, but complicated | Time for Brownian motion to reach a certain level | Medium |
| Mielke  (Dagum / burr) | a, b, p | Hard | None for optimising both parameters | Modelling income & wealth distributions | Hard & unsuitable |
| Fatiguelife | Mu, gamma, beta | Hard | None for optimising both parameters | Models failure times due to crack growth | Hard & unsuitable |
| Invgamma | Alpha, beta | Algorithm exists | Algorithm exists | Used as a prior | Medium |

1. **[FRI]** Fitting distributions (sections 4-7 of MAIN3.ipynb)
   1. Bayesian optimisation of Lognormal
   2. Manual fit to lognormal, invgauss, invgamma using ipywidgets
   3. Test for Gaussian mixture model – unsuitable
   4. Kernel density estimation

**Unfinished:** Network visualisation.

# Week 4 Plans

**Goal for this week:** fit distribution models to the edge data using Bayesian approach. Select the best out of lognormal/invgamma/invgauss

1. Additional filtering
   1. Did the robot start at the origin node?
      1. Order data for each robot and remove data after is\_final == True
      2. Separate by run\_id & robot\_id and order by time
2. Create a new repo with Charlie’s dataset – congestion\_data
3. Models
   1. Implement invgamma & Invgauss
   2. Hard cutoff minimum / offset
   3. Use probabilistic forecasting to a threshold (i.e. train on a subset of data for one edge and test against unused data for the same edge)
4. Where are the opportunities for generalisation? Use KS between each pair of edges in the environment to check for similarities
   1. Same length, same no. of branches
   2. Use Walmart to start / STRANDS
5. Does the generalisation hold if we increase the number of robots on an edge
   1. Walmart\_targeted/ Blenheim
6. Data from Strands robots
   1. How well can we fit noisy data?
   2. We might overfit if the data is very clean
   3. Tsc is office / aaf is hospital
7. Create Network visualisation map
8. Research update for Tues 27 July (30 mins)

**Goal for next week:** start looking at how to transform the posterior of one edge into the prior of another edge based on spatial similarities

# Questions

1. Reading paper for tomorrow
2. How do you train a model with an offset parameter?
   1. Filter out data that is 3std from mean then use the smallest observed duration as the offset? What happens if you only have 3 data points?
3. Multinomial lognormal?