CVEN 626 – Highway Safety

Assignment #5

This assignment is due on Friday, October 31st, 2025. Please be ready to discuss the topic in Friday's lecture.

Use the same team as for Assignment 4.

All questions are 20 points.

Question 1

Using the intersection data I provided on Canvas estimate a Poisson- gamma model for *TotalAllCrashTypes*, *FIAllCrashTypes* (Fatal+Injury), and *PDOAllCrashTypes* with the following functional form:

$$Y = \beta\ 0\ F\ Maj\ \beta\ 1\ F\ Min\ \beta\ 2\ exp\ (\ \beta\ 3\ x\ 3 + \beta\ 4\ x + \ldots + \beta\ j\ x\ j\ )$$

The *Xs* are the variables such as lighting (use 1 for lighting and 0 for no lighting – this is a categorical variable), number of through lanes, etc.

Include variables that are statistically significant at the 5% or 10% level.

Conduct a diagnostic of the results: significance of the coefficients, examination of the residuals, etc. and discuss the results.

Question 2

Repeat Question 1, but use a Poisson-lognormal model. Discuss the differences between the two, if any, and explain why. Use the same criteria for selecting the variables.

Here's the Google search output for "Python Poisson-lognormal model":

See next page.

AI Overview

You can implement Poisson-lognormal models in Python using the `pyPLNmodels` library, which provides tools for analyzing multivariate count data, such as from genomics, ecology, and metagenomics. The library offers a quickstart guide and supports various models like the standard Poisson lognormal model, rank-constrained models for PCA, and zero-inflated versions. To install it, use `pip install pyPLNmodels`.

Key features of `pyPLNmodels`

- **Model types**: Includes standard, rank-constrained (PLNPCA), zero-inflated (ZIPln), and mixture models for clustering.
- **Functionality**: Provides functions for model fitting, transformation, and diagnostics, as well as capabilities for normalization, variable significance analysis, and correlation analysis.
- **Formula interface**: Uses a formula interface similar to R's `lm`, allowing for easy specification of models (e.g., `pln = Pln.from_formula("endog ~ 1 + tree + dist2ground + orientation", data=oaks, take_log_offsets=True)`).

Quickstart

1. **Install**: `pip install pyPLNmodels`
2. **Import**: `import pyPLNmodels`
3. **Load data**: Use the provided `load_oaks` function or your own data.
4. **Fit a model**:
   - `pln = Pln.from_formula("endog ~ 1 + tree + dist2ground + orientation", data = oaks, take_log_offsets = True)`
   - `pln.fit()`
5. **Inspect the results**: `print(pln)`
6. **Transform data**: `transformed_data = pln.transform()`

pyPLNmodels - PyPI

The PLN model described in the PLN-equation is the building block of many different statistical tasks adequate for count da…

PyPI

PLN-team/pyPLNmodels: Blazing fast inference of ... - GitHub

PLNmodels: Poisson lognormal models. The Poisson lognormal model and its variants are used for the analysis of multivariat…

GitHub

pyPLNmodels · PyPI

One main functionality is to normalize the count data to obtain more valuable data. It also analyse the significance of each…

PyPI

Show all