ISTA 311 Computational Assignment 3
Case Study: Hurricane Rates

Due: Tuesday, November 26th at 11:59 PM

1.1 Problem Description

Is the number of hurricanes increasing? What about the severity? In this assignment, we will create a statistical model of hurricanes to investigate these questions.

1.2 Available Data

To investigate this question, we will use a data set containing the number of significant storms recorded in the Atlantic each year from 1851 (when record-keeping began) and 2017. This data set can be found on D2L as `hurricanes.csv`. The table has 4 columns: year, number of named storms, number of hurricanes, and number of major hurricanes (defined as category 3 or greater).

1.3 Two Inference Models

We will construct a model predicting the number of hurricanes in a year, and fit it

- the number of hurricanes observed each year during a certain time range
- the number of major hurricanes observed each year during a certain time range

1.3.1 The hurricane model: Poisson statistics

We can model the number of hurricanes per year as a Poisson process. The Poisson distribution is a probability distribution that models counts of randomly occuring events. Poisson statistics are suitable when the events occur independently at random intervals, so that the number of events occurring in a given time period is dependent only on the length of that period.

The Poisson distribution is described by a single parameter, called the *rate* λ. This parameter represents the expected number of events in a given time period; in our case, the expected number of (major or total) hurricanes in a given year.

The Poisson distribution is defined by the following formula:

$$P(n \text{ events}|\text{rate } \lambda) = \frac{\lambda^n e^{-\lambda}}{n!} \tag{1}$$

This is the foundation of our likelihood model: the above formula gives the probability that there will be n hurricanes in a given year for a hypothesized rate λ. You do not need to implement this formula yourself, however, because there is a standard implementation in SciPy. After running

```
import scipy as sp
from scipy import stats
```

you can call

```
sp.stats.poisson.pmf(n, lambda)
```

to obtain the probability (1), which you can use in your `likelihood` method. The other operation you will need to do is generate random values from a given Poisson distribution, for which you can use:

```
sp.stats.poisson.rvs(lambda)
```

1.4 Implementing the model (15 points)

In order to implement the model, define a subclass of `InferenceSuite` called `HurricaneModel`. In this class, the dictionary `self.d` contains the probability distribution of the rate parameter λ.

Your class should implement two methods (besides the ones inherited from parent classes):

- a `likelihood` method based on the Poisson distribution
- `predict`, which is used to make predictions.

  `HurricaneModel.predict()` returns a prediction of the number of hurricanes in a year by generating a value from the appropriate Poisson distribution. To obtain `lambda`, use the `sample()` method to sample a value from the posterior distribution stored in the model (recall `sample` is inherited from the `Distribution` class we defined back in HW1).

  This prediction may be interpreted as a prediction of total hurricanes or major hurricanes, depending on which type of observation is used to update the distribution of λ.

1.5 Fitting, trends, and prediction (45 points)

In this section we will create several instances of each of our models, "train" them on subsets of our historical data, and compare the predictions that each model makes.

To load the data as a NumPy/SciPy array, you can use `loadtxt`:

```
data = sp.loadtxt('hurricanes.csv', delimiter = ',', skiprows = 1)
```

1.5.1 Model fitting

For each of the following tasks, initialize an instance of `HurricaneModel`. A uniform prior will suffice (optional reading: see the note at the end of the assignment for more details) since we are using sufficient data to "swamp the prior"; that is, for the influence of the data to outweigh the influence of the prior. What is important is that you choose a reasonable domain for values of λ, use equally spaced values of λ with a spacing no greater than 0.1 (lower limit should be 0; upper limit is up to you, but keep in mind the interpretation of the number λ).

To fit a model, simply call the `update` method once with each number you want to include in the model fitting (e.g., in a for loop).

1. Fit a model for the total number of hurricanes per year using years 1965-1974
2. Fit a model for the total number of hurricanes per year using years 2006-2015
3. Fit a model for the number of major hurricanes per year using years 1965-1974
4. Fit a model for the number of major hurricanes per year using years 2006-2015

1.5.2 Example

As an example, the following code would fit a model for the number of major hurricanes per year, using *all* years in the data set. Here I assume that the data has been loaded into an array called `data`.

```
all_year_major = HurricaneModel(sp.linspace(0, 25, 251))
for i in range(len(data[:,3])):
    all_year_major.update(data[i,3])
```

The argument to `HurricaneModel` initializes a uniform prior with values between 0 and 25 for λ, with a spacing of 0.1. The index 3 refers to the 4th column of the data set, which contains the number of major hurricanes in each year. The 3rd column has the total number of hurricanes.

1.5.3 Prediction

In this section, we predict the number of major hurricanes in a year "today." Using the `predict` method on your `HurricaneModel` instances, generate 1,000 predictions for each of the following:

1. The total number of hurricanes in a hypothetical year, using your model based on 1965-1974
2. The total number of hurricanes in a hypothetical year, using your model based on 2006-2015
3. The number of major hurricanes in a hypothetical year, using your model based on 1965-1974
4. The number of major hurricanes in a hypothetical year, using your model based on 2006-2015

Use your predicted numbers to plot two histograms: one comparing the two predictions for total hurricanes, and one comparing the two predictions for major hurricanes These histograms should display both distributions on the same axis. An example script to generate a similar histogram, and the example histogram, are provided so you can use it as a starting point for your plotting. (The script will only run if you have defined your `HurricaneModel`.)

1.5.4 One final plot

As a final point of investigation, let's see how these 10-year data windows change over time.

For each year starting from 1860 (10 years into the data set), fit a model for the total number of hurricanes based on data from the previous 10 years. That is, the first model will use years 1851-1860, the second 1852-1861, etc. For each model, record the maximum likelihood estimate for λ, which you can obtain by calling the `map` method. Then, plot a line graph of the estimates for λ vs. the year. (Use `plt.plot()` for a line graph.)

Along with your code, submit a brief report (PDF format) with the histograms and the graph. Briefly (one or two paragraphs) comment on the distributions and the line graph. Do your models based on recent data (2006-2015) predict different numbers than the models based on older data? How does the maximum likelihood estimate for λ change over time?

In 2017, there were 10 hurricanes, 6 of which were major. Based on your histograms, does this appear typical for the 1965-1974 era? What about the current era?

2 Notes

Here is some additional reading on some of the details in the choices made above.

2.1 Note: data after 1965

From the previous section, you have probably noticed that we restricted ourselves to data after 1965. This is in part because many hurricanes went undetected before the advent of satellite monitoring in the 1960s. As a result, models fitted on earlier data often underestimate the number of hurricanes.

Therefore, it is instructive to consider the period after 1965 separately, since it is not subject to the undercounting of earlier times.

2.2 A note on priors

Both of these models can perform well with uniform priors; we are using a sufficient amount of data that the choice of prior will not make a large difference.