Homework 3 Solutions

Due Date

Friday, 3/22/24, 9:00pm

Tip

To do this assignment in Julia, you can find a Jupyter notebook with an appropriate environment in the homework's Github repository. Otherwise, you will be responsible for setting up an appropriate package environment in the language of your choosing. Make sure to include your name and NetID on your solution.

Overview

Instructions

The goal of this homework assignment is to practice developing and working with probability models for data.

Learning Outcomes

After completing this assignments, students will be able to:

- develop probability models for data and model residuals under a variety of statistical assumptions;
- evaluate the appropriateness of those assumptions through the use of qualitative and quantitative evaluations of goodness-of-fit;
- fit a basic Bayesian model to data.

Load Environment

The following code loads the environment and makes sure all needed packages are installed. This should be at the start of most Julia scripts.

```
import Pkg
Pkg.activate(@__DIR__)
Pkg.instantiate()
```

The following packages are included in the environment (to help you find other similar packages in other languages). The code below loads these packages for use in the subsequent notebook (the desired functionality for each package is commented next to the package).

```
using Random # random number generation and seed-setting
using DataFrames # tabular data structure
using DataFramesMeta # API which can simplify chains of DataFrames
↪ transformations
using CSVFiles # reads/writes .csv files
using Distributions # interface to work with probability distributions
using Plots # plotting library
using StatsBase # statistical quantities like mean, median, etc
using StatsPlots # some additional statistical plotting tools
using Optim # optimization tools
using Dates # DateTime structures and interface
```

Problems (Total: 30 Points for 4850; 40 for 5850)

Problem 1

Consider the Rahmstorf (2007) sea-level rise model from Homework 2:

$$\frac{dH(t)}{dt} = \alpha(T(t) - T_0),$$

where T_0 is the temperature (in $^\circ C$) where sea-level is in equilibrium ($dH/dt = 0$), and α is the sea-level rise sensitivity to temperature. Discretizing this equation using the Euler method and using an annual timestep ($\delta t = 1$), we get

$$H(t + 1) = H(t) + \alpha(T(t) - T_0).$$

Suppose that we wanted to develop a Bayesian probability model for this problem, assuming independent normal residuals:

$$y(t) = F(t) + \varepsilon_t$$
$$\varepsilon_t \sim \mathcal{N}(0, \sigma^2)$$

We might specify the following priors (assuming independence across parameters):

- $T_0 \sim \mathcal{N}(-0.5, 0.1)$;
- $\alpha \sim \mathcal{TN}(0, 5; 0, \infty)$ (truncated normal between 0 and infinity);
- $H_0 \sim \mathcal{N}(-150, 25)$;
- $\sigma \sim \mathcal{TN}(0, 5; 0, \infty)$

In this problem:

- Historical and RCP 8.5 global mean temperatures from NOAA can be found in `data/NOAA_IPCC_RCPtempsscenarios.csv` (use the fourth column for the temperature series).
- Global mean sea level anomalies (relative to the 1990 mean global sea level) are in `data/CSIRO_Recons_gmsl_yr_2015.csv`, courtesy of CSIRO (https://www.cmar.csiro.au/sealevel/sl_data_cmar.html).
- Simulate from the prior predictive distribution. What do you think about the priors?
- Would you propose new priors? If so, what might they be and why?

Solution:

First, let's load the data.

```julia
# load data files
slr_data = DataFrame(load("data/CSIRO_Recons_gmsl_yr_2015.csv"))
gmt_data = DataFrame(load("data/NOAA_IPCC_RCPtempsscenarios.csv"))
slr_data[:, :Time] = slr_data[:, :Time] .- 0.5; # remove 0.5 from Times
dat = leftjoin(slr_data, gmt_data, on="Time") # join data frames on time
select!(dat, [1, 2, 6])  # drop columns we don't need
first(dat, 6)
```

```
 Warning: In data/NOAA_IPCC_RCPtempsscenarios.csv line 426 has 0 fields but
13 fields are expected. Skipping row.
 @ TextParse ~/.julia/packages/TextParse/gNKVx/src/csv.jl:382
```

	Time	GMSL (mm)	Historical NOAA temp & CNRM RCP 8.5 with respect to 20th century
	Float64	Float64	Float64?
1	1880.0	-158.7	-0.16
2	1881.0	-153.1	-0.1
3	1882.0	-169.9	-0.12
4	1883.0	-164.6	-0.17
5	1884.0	-143.7	-0.23
6	1885.0	-145.2	-0.2

Creating the model:

```
# slr_model: function to simulate sea-level rise from global mean temperature
↪  based on the Rahmstorf (2007) model

function slr_model( , T , H , temp_data)
    temp_effect =   .* (temp_data .- T )
    slr_predict = cumsum(temp_effect) .+ H
    return slr_predict
end
```

```
slr_model (generic function with 1 method)
```

Now, we can sample from the priors and simulate the prior predictive distributions.

- $T_0 \sim \mathcal{N}(-0.5, 0.1)$;
- $\alpha \sim \mathcal{TN}(0, 5; 0, \infty)$ (truncated normal between 0 and infinity);
- $H_0 \sim \mathcal{N}(-150, 25)$;
- $\sigma \sim \mathcal{TN}(0, 5; 0, \infty)$

```
# set up prior distributions
T _prior = Normal(-0.5, 0.1)
 _prior = truncated(Normal(0, 5), lower=0)
H _prior = Normal(-150, 25)
 _prior = truncated(Normal(0, 5), lower=0)

# sample and simulate
n_samples = 10000
T  = rand(T _prior, n_samples)
  = rand( _prior, n_samples)
H  = rand(H _prior, n_samples)
  = rand( _prior, n_samples)
slr_prior = zeros(n_samples, nrow(dat))
for i = 1:n_samples
```