Probabilistic Programming @ DSR

14.03.2020 - Davide Posillipo

About me

- Statistician and Data Scientist | Senior Consultant @ Deloitte, Berlin
- Bachelor and Master degrees in Statistical Science
- DSR Alumnus (Batch 15)
- Data Science mentor (let’s keep in touch!)

Agenda

- Introduction
- Statistics Preliminaries
- Analysis Workflow
- First practical application: switch point analysis
- PyMC3 deep dive
- A bit of theory
- Second practical application: Bayesian t-test alternative
- Diagnostics
- Third practical application: AB testing for advertisement
- Probabilistic Programming for Machine Learning tasks
- Overview of topics we didn’t cover but that are important
- QA

Motivations

- Interpretability
  - This approach let us *model* phenomena reflecting domain-specific knowledge
- Uncertainty
  - We get *for free* a measure of uncertainty of our results

Real life example

- Document Recognition with Probabilistic Programming
- The goal: create a Document Recognition module for an anomaly detection tool
- We need to know if a document is of a known type or not
- How would you do?

Goals for today

- A basic understanding of the Bayesian approach
- Practical knowledge of Probabilistic Programming with PyMC3

Main resources

- Bayesian Methods for Hackers
- PyMC3 website (the documentation is not great, but there are nice tutorials)
- Bayesian Data Analysis, Gelman
- Statistical Rethinking, McElreath

Introduction

A classical example: the coal mining disaster problem

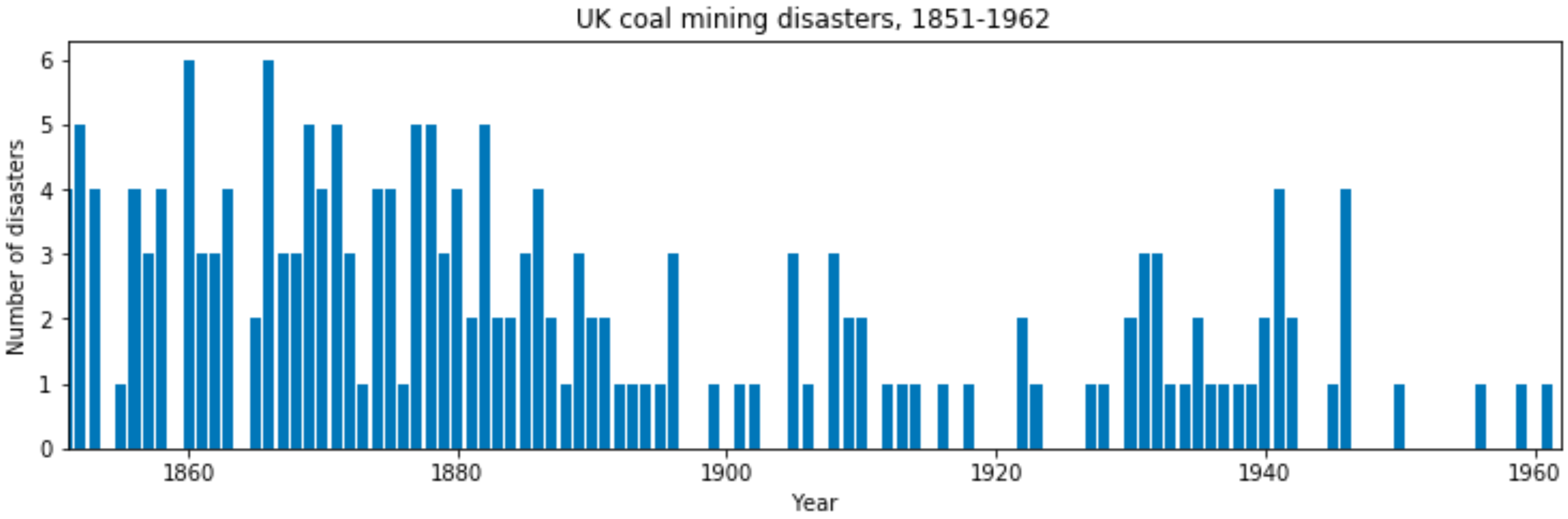
- Each observation is the number of registered coal mining disaster in 1 year, in UK, in the period 1851-1962

- The time series covers 111 years

- The question we would like to answer is: in which year the trend significantly changed?

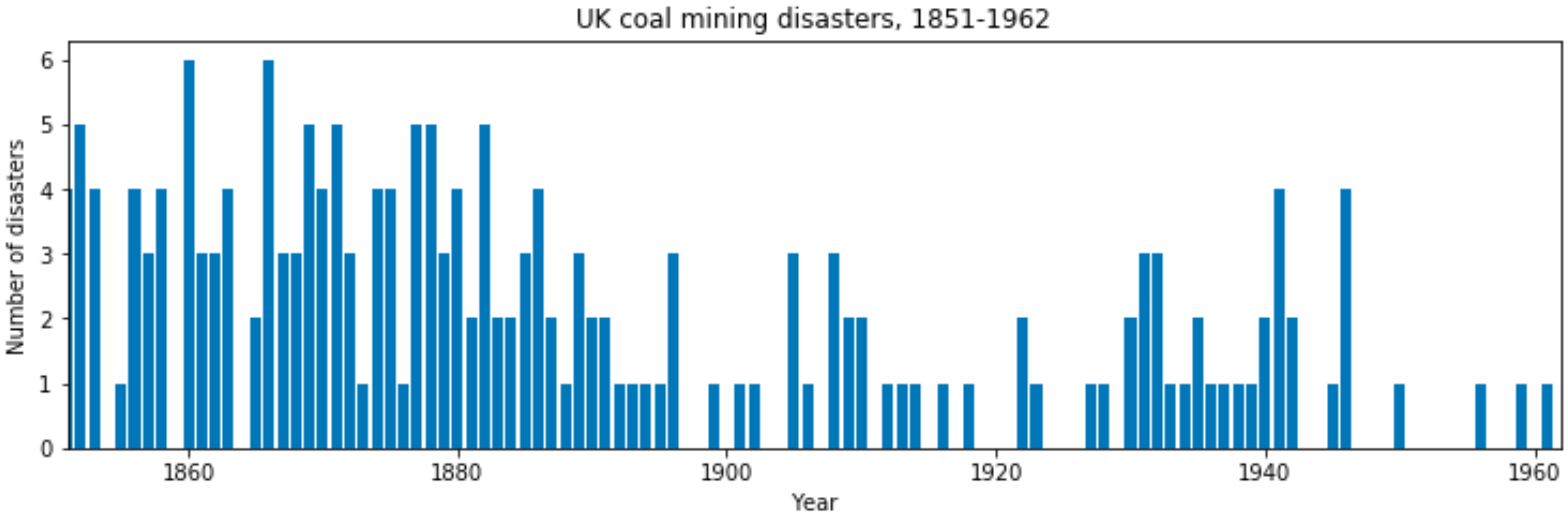
- This application is also called *switch point analysis*

A classical example: the coal mining disaster problem



- How would you solve this task?

A classical example: the coal mining disaster problem



- The Bayesian way: modelling the phenomenon with a random variable, specifying its parameters and estimating them via simulation

What is Probabilistic Programming?

- Probabilistic Programming (PP) is our way to solve problems like the coal mining disaster
- Probabilistic programming is fundamentally about developing languages that allow the denotation of inference problems and evaluators that “solve” those inference problems

What is a Statistical Model?

- A Statistical Model is a representation in terms of random variables of a quantifiable problem or phenomenon

- The key ingredients are usually the parameters that specify the used random variables (at least in Parametrical Statistics)

- Parameters -> estimation

Statistics preliminaries

Conditional Probability

- [Main concept]
- [Properties]
- WHITEBOARD

Bayes Theorem

- [Definition]
- [Diagnosis example]
- [Properties]
- WHITEBOARD

Bayesian Statistics in a nutshell

- When we model a phenomenon in terms of posterior distribution, we are doing Bayesian Statistics

- Why this can be interesting in data science?

Differences Frequentist-Bayesian approaches

- The frequentist approach considers the probability as a frequency and, often, as an inherent property of entities/phenomena
- The bayesian approach considers the probability as a belief
- This is reflected in the meaning and mathematical handling of parameters

Main probabilities distributions

- What is a probability distribution?
- Jupyter Notebook session: excursus over the main distributions

(Bayesian) Analysis Workflow

Analysis workflow overview

- Model specification
  - Priors definition
  - Likelihood definition
  - Posterior computation (analytically, if possible)
- Posterior computation (via simulation)
- Inference
  - Choice of estimate
  - Computation of intervals, test, forecasts, …

Model specification

- Priors distributions can reflect specific prior knowledge or being “agnostic” (for example, a Uniform prior)

- The likelihood function is defined considering the probabilistic mechanism that could have generated the data

Simulation

- We simulate because we are not able to analytically derive the posterior distribution

- There are many different algorithms for the posterior simulation

- With PyMC you can (almost always) ignore the details of the simulation algorithms

Inference

- Having the posterior distribution (or its simulation), it is possible to perform the inference of interest
- This is connected to the meaning of the parameters we are trying to estimate

First application: Switch point analysis

Switch point analysis with PyMC3

- Switch point analysis “Coal Mining Disaster”
- Jupyter Notebook session

Deep dive in PyMC3

PyMC3 main properties

- Model Context
- Variables
- Theano

A bit of theory, or why PP works

Main points so far

- Simulation from posterior distribution
- Priors definition
- Bayesian theorem as we applied it

Simulation algorithms for MCMC method

- Metropolis-Hastings
    - Random Walk Metropolis-Hastings
- Hamiltonian Monte Carlo
- NUTS (No U-Turn Sampling)

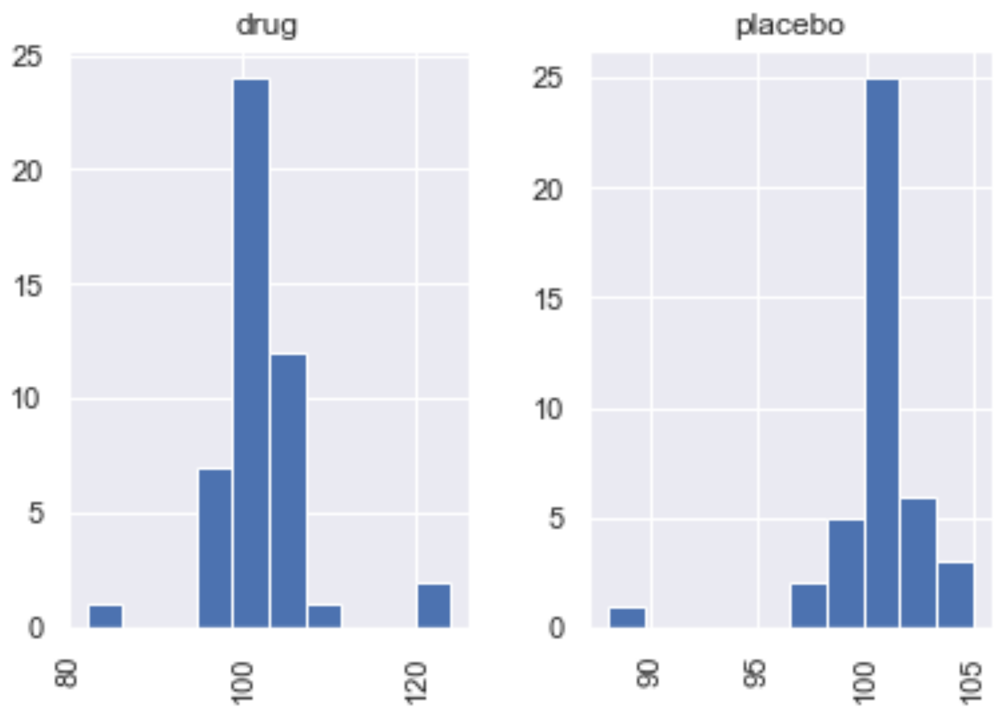
Second application: Drug Trial Evaluation

T-test: a classic of frequentist approach

- The goal is to test if two averages are different or not
- It relies on a parametrical assumption
- It is probably the most used statistical test ever

Drug Trial Evaluation: description

- Clinical trial for drug evaluation
- 47 people in the treatment group
- 42 people in the control group



Diagnostics

Main diagnostics and why they matter

- **Convergence diagnostics**: they refer to the sampling algorithm used for the

- **Goodness of fit diagnostics**: they refer to how good the model is, so they aren’t directly inherent to PP

Third application:
AB testing for
advertisement

AB testing: main concept

- AB testing is a very valuable application for many companies
- It is mainly a treatment vs control analysis, where usually the control on the sample design is very limited or absent
- Compared to the classical clinical settings, the sources of biases are more serious

AB testing for advertisement

- The best practice is the model beta-bernoulli
- We will first give a look to it on the whiteboard and then implement it on PyMC3

Bayesian approach for the Document Recognition problem