Hierarchical Bayesian models in PyMC

Meenal Jhajharia

github:mjhajharia/pydata21

Meenal Jhajharia. she/her. 19.

CS and Math undergrad student

University of Delhi, India

PyMC contributor | GSoC student - PyMC (under NUMFOCUS)

meenal@mjhajharia.com | mjhajharia.com

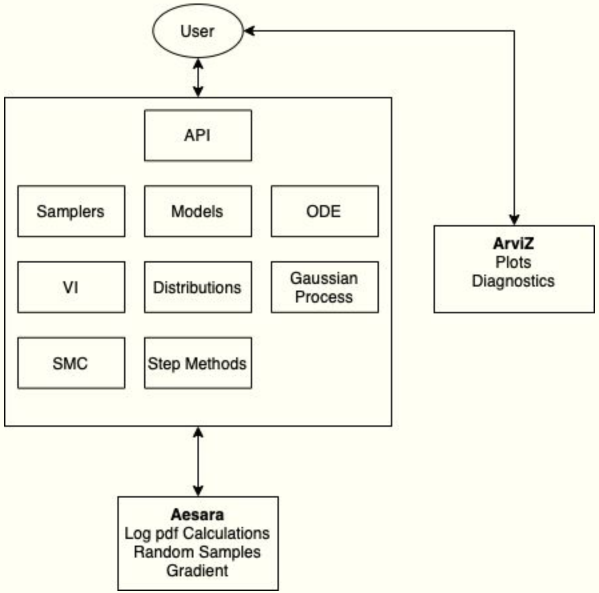
Github: mjhajharia | Twitter: meenaljhajharia

Bayesian Inference

Priors + Likelihood —> Posterior

Model

Probabilistic Programming Languages



pymc-devs/pymc/architecture.md

Bayesian Hierarchical Modeling

Model written in multiple levels (or hierarchies) to estimate the posterior

Hyperparameters: parameters of the prior distribution

Hyperpriors: distributions of Hyperparameters

Two stage Hierarchical Model

$$P(\theta, \phi \mid Y) = \frac{P(Y \mid \theta, \phi)P(\theta, \phi)}{P(Y)} = \frac{P(Y \mid \theta)P(\theta \mid \phi)P(\phi)}{P(Y)}$$

$$P(\theta, \phi \mid Y) \propto P(Y \mid \theta)P(\theta \mid \phi)P(\phi)$$

Let’s build a couple of dummy models to get a taste of the problem

Task: Predict Scores of a football player in Fantasy Premier League

Predictors:

- Previous season’s scores
- Player’s Position

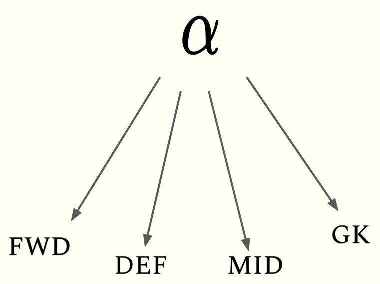
Mixed Effects

- Varying Slope
- Varying Intercept
- Varying Slope and Varying Intercept

PyMC notebook

Complete Pooling

Combines all the information for different positions into a single “pool” of data, which is to say we assume the effects(on the model) of all the positions are identical.



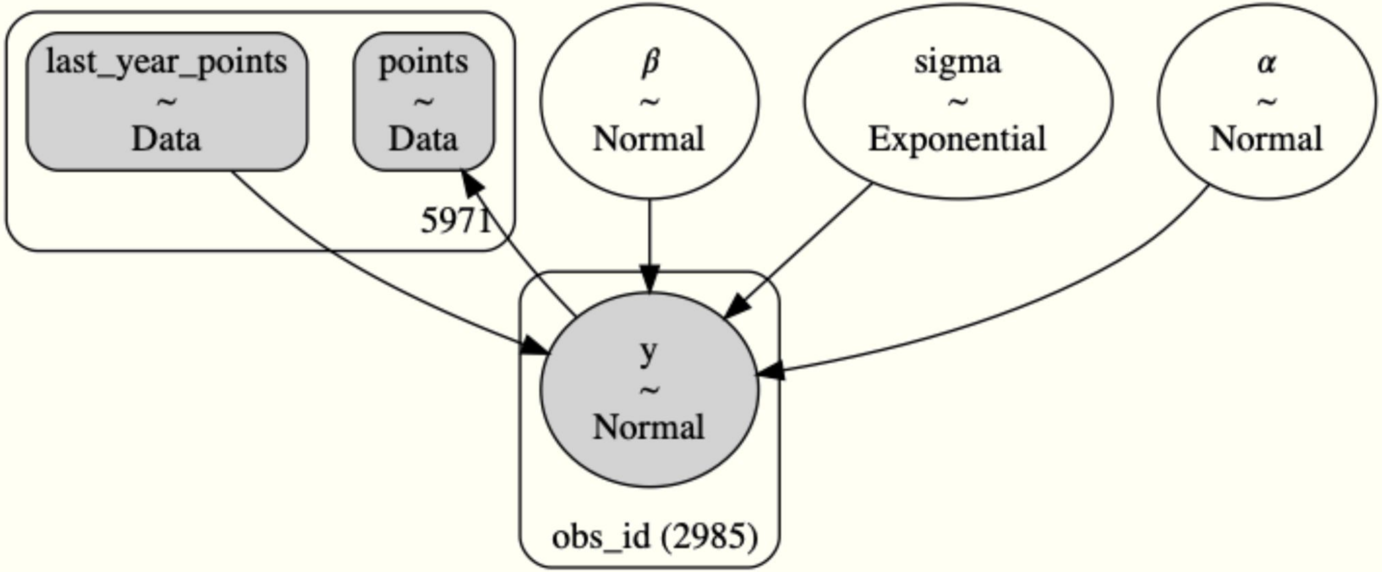
$$y_i = \underline{\alpha} + \beta x_i + \epsilon_i$$

Points to predict

Previous year’s points

Gaussian noise

```
with pm.Model( ) as pooledmodel:
    last_year_points = pm.Data('last_year_points', np.asarray(data['total_points_20'],dtype=float))
    points = pm.Data('points', np.asarray(data['total_points'], dtype=float))
    alpha = pm.Normal("α", mu=0, sigma=10.0)
    beta ------------------------------ {insert suitable priors}
    sigma ----------------------- {insert suitable priors}
    y = pm.Normal("y", alpha + beta*last_year_points, sigma=sigma, observed=points)
```



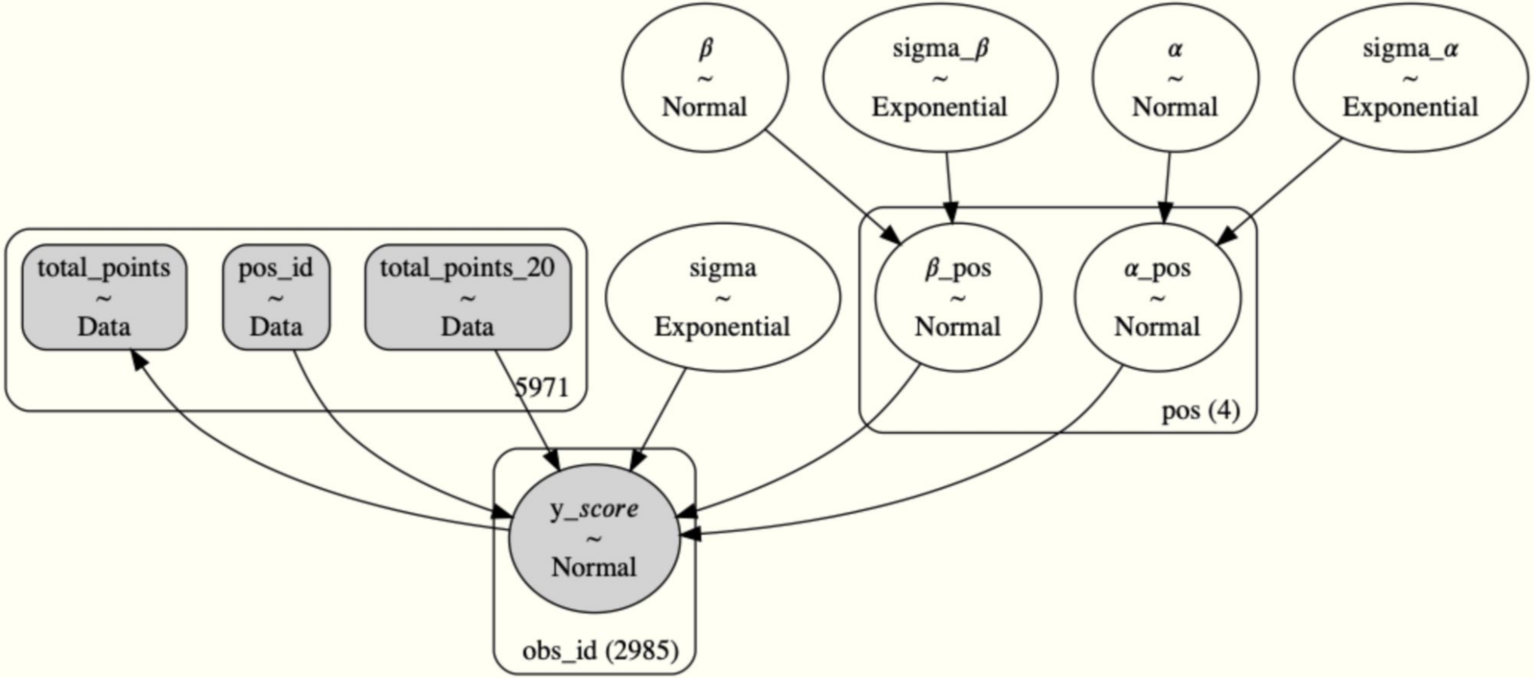
No Pooling

In unpooled models, we assume that variables are sampled independently and have no similarity whatsoever, in our case that is the position variable

$$y_i = \underline{\alpha_{j[i]}} + \beta x_i + \epsilon_i$$

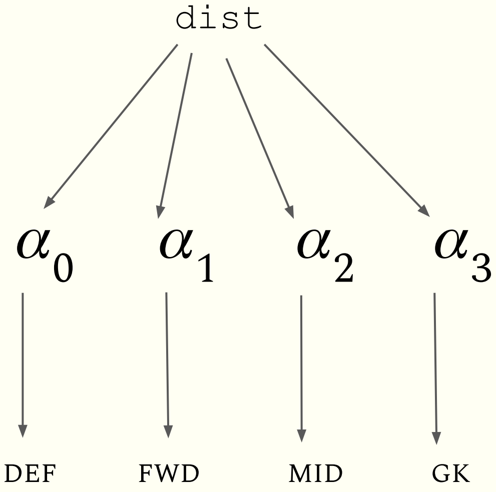
\begin{tikzcd}
\alpha_0 \arrow[d] & \alpha_1 \arrow[d] & \alpha_2 \arrow[d] & \alpha_3 \arrow[d] \\
\text{DEF} & \text{FWD} & \text{MID} & \text{GK}
\end{tikzcd}

```
alpha = pm.Normal("α", 0.0, sigma=10.0, dims=("pos"))
```



Partial Pooling

Model parameters are viewed as different samples from a population distribution of parameters sampled from a single distribution



```
alpha_pos = pm.Normal("α_pos", mu=alpha, sigma=sigma_alpha, dims="pos")
```