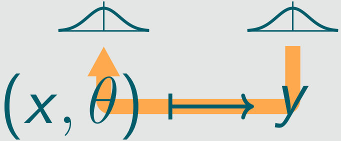
Parameter Estimation

Bayesian Parameter Estimation for Photosynthesis-Irradiance Models

J. Paul Mattern

October 2025



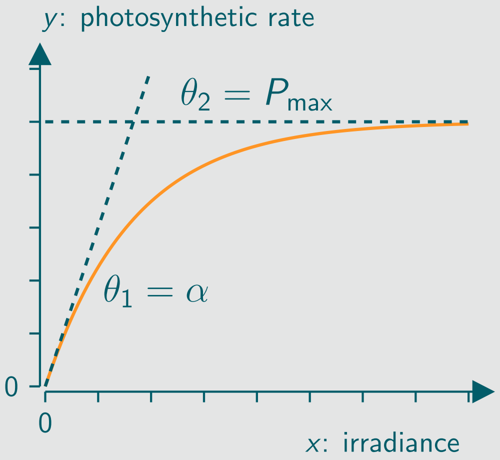
Modelling Primary Production Workshop

Parameter estimation

Parameter estimation describes the process of estimating the values of model parameters with the goal of improving model performance or learning about the processes associated with the parameters.

model:

$$(x, \theta) \longmapsto y$$



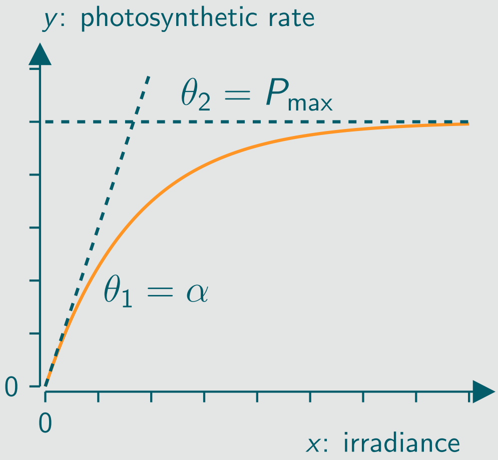
Parameter estimation

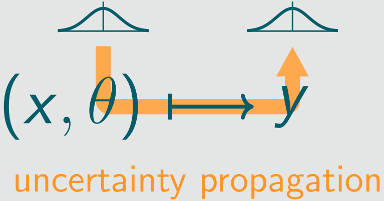
Parameter estimation describes the process of estimating the values of model parameters with the goal of improving model performance or learning about the processes associated with the parameters.

model:

$$(x, \theta) \longmapsto y$$

parameter estimation





- In uncertainty or error propagation, we go from known input uncertainties to output uncertainties.
- Uncertainty propagation provides probability distributions for outputs.
- Findings from uncertainty propagation can help us design better parameter estimation experiments:
  - design better sampling strategies (what needs to be observed?)
  - inform which parameters we can realistically estimate

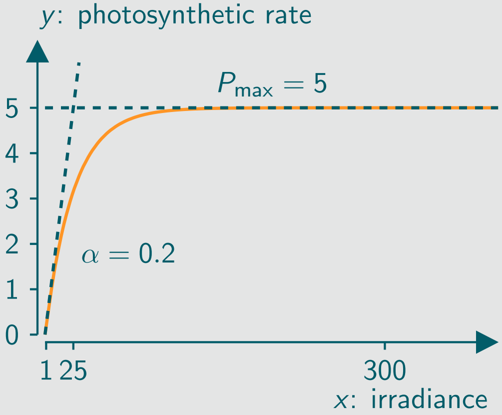
Simple uncertainty propagation in the P-I model

Uncertainty propagation in the P-I model using

- $\alpha = 0.2 \pm 0.02$
- $P_{\text{max}} = 5.0 \pm 0.5$

assuming uncorrelated errors.
The uncertainty is calculated as:

$$\sigma_{p^B} = \sqrt{\left(\frac{\partial p^B}{\partial \alpha}\right)^2 \sigma_\alpha^2 + \left(\frac{\partial p^B}{\partial P_{\text{max}}}\right)^2 \sigma_{P_{\text{max}}}^2}$$



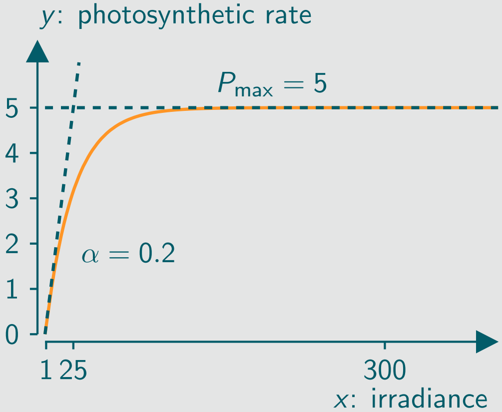
Simple uncertainty propagation in the P-I model

Uncertainty propagation in the P-I model using

- $\alpha = 0.2 \pm 0.02$
- $P_{\text{max}} = 5.0 \pm 0.5$

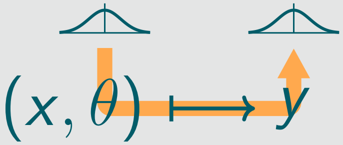
assuming uncorrelated errors.
The uncertainty is calculated as:

$$\sigma_{p^B} = \sqrt{\left(\frac{\partial p^B}{\partial \alpha}\right)^2 \sigma_\alpha^2 + \left(\frac{\partial p^B}{\partial P_{\text{max}}}\right)^2 \sigma_{P_{\text{max}}}^2}$$



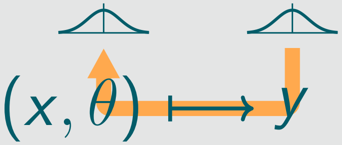
irradiance (W m^{-2})	photosynthetic rate (mg C mg chl^{-1} h^{-1})	uncertainty (mg C mg chl^{-1} h^{-1})	relative uncertainty
1	0.196	0.043	22.1%
25	3.161	0.533	16.9%
300	5.000	0.500	10.0%

Uncertainty propagation summary



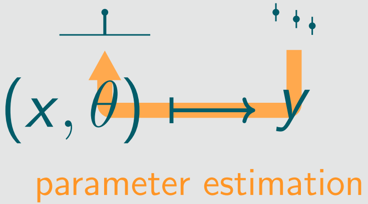
uncertainty propagation

- We go from known input uncertainties to output uncertainties.
- Provides probability distributions for outputs.
- We can use Monte Carlo-based sampling for more complex models and non-Gaussian errors.



parameter estimation

- We typically go backwards, using observed output variations to infer input parameter estimates.
- Provides point estimates or probability distributions for input parameters.



- Simpler parameter estimation techniques often provide point estimates.
- These techniques aim to find the optimal parameter values, that minimize the model misfit to the observations based on some measure.
- The terms “parameter optimization” and “maximum likelihood estimation” describe a wide range of techniques that provide point estimates for parameters.

parameter estimation for the P-I curve

- Most parameter optimization techniques minimize a function that quantifies the model misfit to the observations. This function has many names: “objective function”, “cost function”, “loss function” etc.
- Often the sum of squared errors is used:

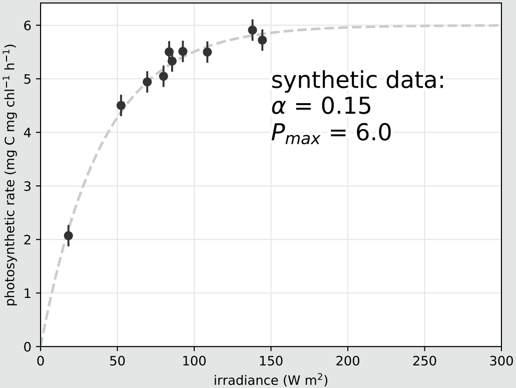
$$L(\theta) = L\left(\begin{bmatrix} \alpha \\ P_{\text{max}} \end{bmatrix}\right) = \sum_{i=1}^{n} \left(P_{\text{obs},i} - P_{\text{max}}(1 - e^{-\alpha I_i / P_{\text{max}}})\right)^2$$

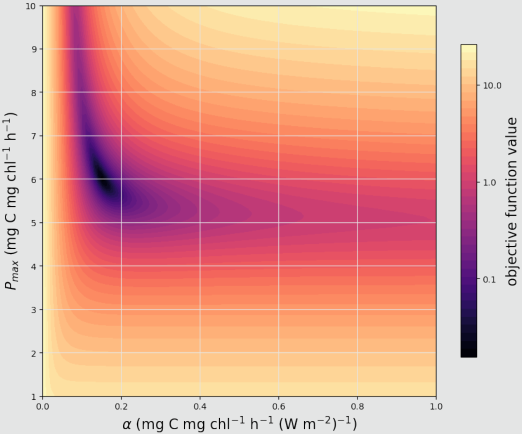
- In maximum likelihood estimation maximizing a Gaussian likelihood is equivalent to minimizing the sum of squared errors: $p(P_{\text{obs}}|\theta) \propto \mathrm{e}^{-L(\theta)}$

parameter estimation for the P-I curve

objective function:

$$L(\theta) = L\left(\begin{bmatrix}\alpha \\ P_{\max}\end{bmatrix}\right) = \sum_{i=1}^{n}\left(P_{\mathrm{obs},i} - P_{\max}\left(1 - e^{-\alpha I_i/P_{\max}}\right)\right)^2$$



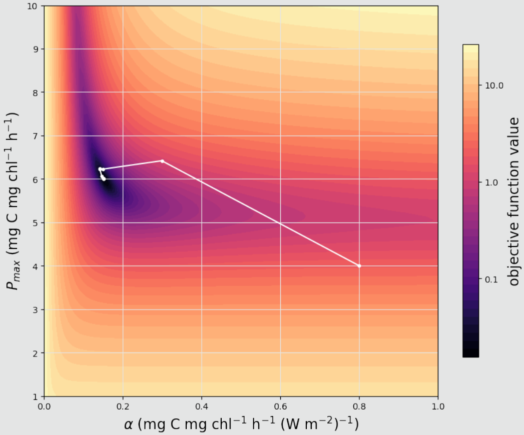


parameter estimation for the P-I curve

objective function:

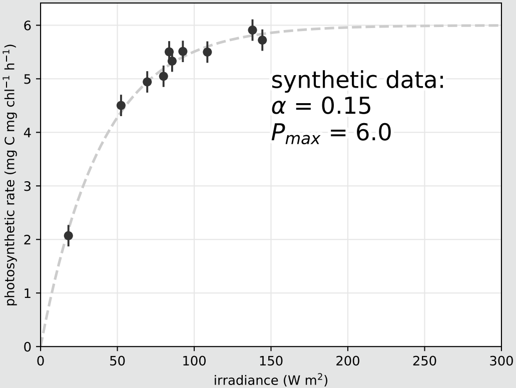
$$L(\theta) = L\left(\begin{bmatrix}\alpha \\ P_{\text{max}}\end{bmatrix}\right) = \sum_{i=1}^{n}\left(P_{\text{obs},i} - P_{\text{max}}(1 - e^{-\alpha I_i/P_{\text{max}}})\right)^2$$

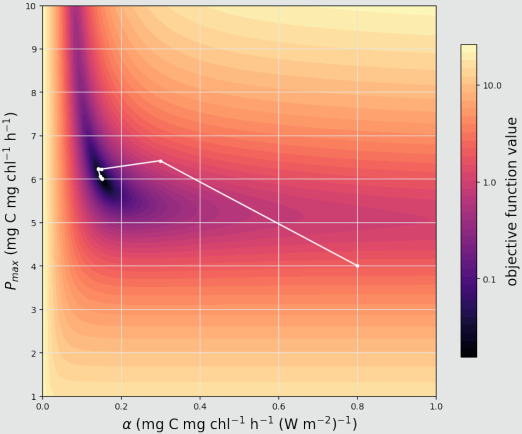
- `scipy.optimize.minimize` using `L-BFGS-B` (quasi-Newton-based Broyden-Fletcher-Goldfarb-Shanno algorithm, limited memory version with bounds on parameters)
- number of function evaluations: 51



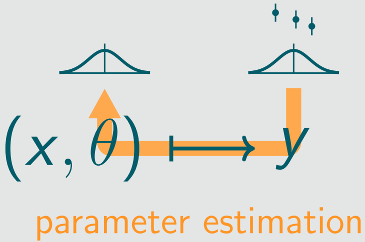
objective function:

$$L(\theta) = L\left(\begin{bmatrix}\alpha \\ P_{\text{max}}\end{bmatrix}\right) = \sum_{i=1}^{n}\left(P_{\text{obs},i} - P_{\text{max}}(1 - e^{-\alpha I_i/P_{\text{max}}})\right)^2$$





Bayesian parameter estimation



- Bayesian techniques provide more than a point estimate of parameter values, they estimate the posterior probability distribution.
- Bayesian parameter estimation requires the specification of prior distributions for each parameter, which can be used to incorporate knowledge into the parameter estimation process and can help alleviate indentifiability issues.
- We first build a Bayesian version of the P-I model using the `pymc` package.

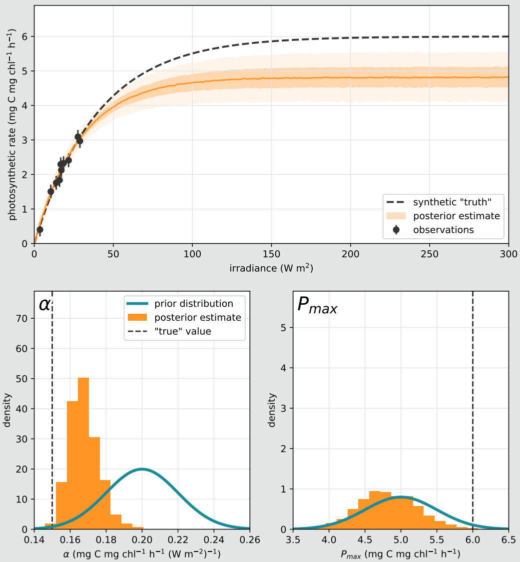
A simple Bayesian model in Python

Using pymc:

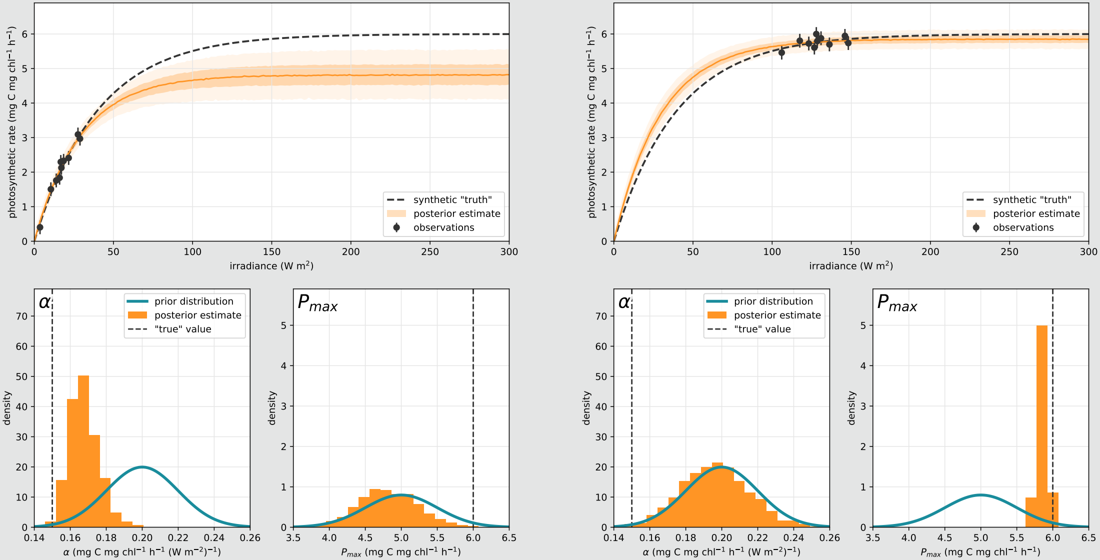
```
import pymc
with pymc.Model(coords={'obs': np.arange(irr_data.size)}) as generative_model:
    irradiance_sample = pymc.Data('irradiance sample', irr_data, dims='obs')

    # priors
    alpha = pymc.Normal('alpha', mu=0.2, sigma=0.02)
    pmax = pymc.Normal('Pmax', mu=5.0, sigma=0.5)

    p_rate_mu = pmax * (1.0-pymc.math.exp(-alpha*irradiance_sample/pmax))
    p_rate_sigma = pymc.HalfNormal('photosynthetic rate sigma', sigma=0.2)
    p_rate = pymc.Normal('photosynthetic rate', mu=p_rate_mu, sigma=p_rate_sigma,
                         dims='obs')
```

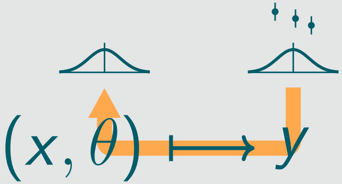
This model can then be used to sample the prior, generate synthetic data from fixed parameters, fit data (synthetic or not) and get predictive samples from the posterior after fitting.

Posterior estimates in the P-I model



Posterior estimates in the P-I model





parameter estimation

- Bayesian models can become complex and used to estimate hundreds of parameter values.
- Here, we go a step in that direction and estimate parameters of the P-I model (α, $P_{\max}$, and respiration rate R) for a large dataset recorded in 1975.

C2

Phytoplankton Productivity Experiments and Nutrient Measurements In Bedford Basin, Nova Scotia, from Sept. 1975 to Dec. 1976

P. Irwin and T. Platt

Marine Ecology Laboratory
Bedford Institute of Oceanography
Fisheries and Marine Service
Department of Fisheries and the Environment
Dartmouth, Nova Scotia B2Y 4A2

Fisheries & Marine Service Technical Report No. 762

Fisheries and Environment Canada — Pêches et Environnement Canada
Fisheries and Marine Service — Service des pêches et de la mer

BEDFORD BASIN

44°41'N 63°39'W

DATE: 09/09/75 SAMPLE DEPTH: 5 m SURFACE TEMP: 15.4°C

Light Intensity W m^{-2}	Specific Production mg C(mg Chl a)$^{-1}$hr^{-1}	Light Intensity W m^{-2}	Specific Production mg C(mg Chl a)$^{-1}$hr^{-1}
179.6	2.39	273.7	2.75
78.1	2.56	113.0	2.89
34.8	1.81	44.5	2.58
19.4	1.31	24.1	1.67
12.1	0.84	13.4	1.12
6.6	0.49	7.9	0.66
4.5	0.27	5.2	0.48
3.1	0.14	3.6	0.27
2.0	0.01	2.5	0.17
1.3	0.01	1.9	0.09

Incubation Temperature: 14.8°C

	mg at m^{-3}		mg m^{-3}
Nitrate:	0.04	Chlorophyll:	14.40
Nitrite:	0.04	Carbon	720
Ammonia:	0.49	Nitrogen:	85
Silicate:	3.70		
Phosphate:	0.29	Salinity:	30.57 ‰

Total number of cells: 0.8 x 10^{6} l^{-1} *
Total volume of cells: 6.45 ppm *
Mean volume of cells: 8050 μ^{3}*

		90% Confidence Interval lower	upper
α mg C(mg Chl a)$^{-1}$hr^{-1}(W m^{-2})$^{-1}$	0.10	0.08	0.11
R^B mg C(mg Chl a)$^{-1}$hr^{-1}	-0.12	-0.19	-0.06
P^B_m mg C(mg Chl a)$^{-1}$hr^{-1}	2.79	2.64	2.96

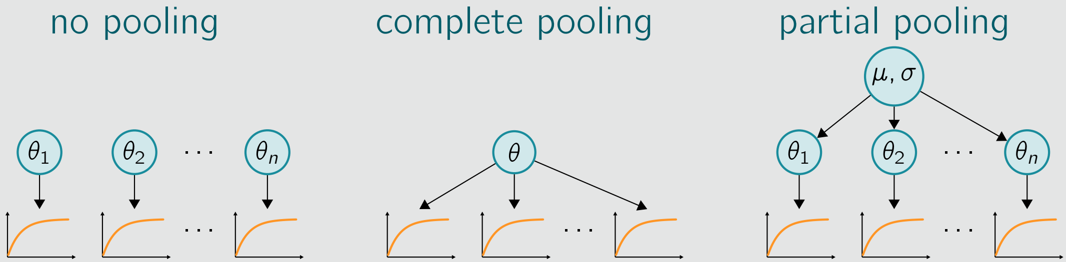
*280 μ tube only

5

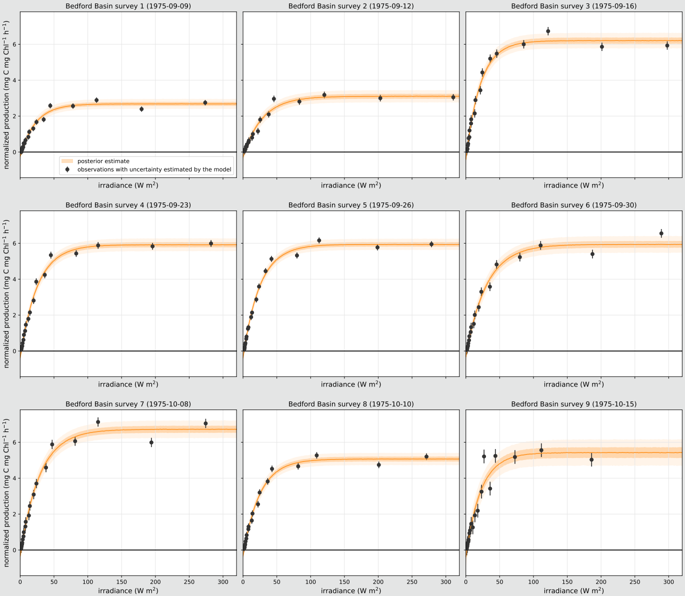
a partial pooling model

Using data from 107 surveys, we want to obtain 107 parameter estimates of $\theta = [\alpha, P_{\text{max}}, R]^{\mathsf{T}}$.

For our Bayesian model, we use a partial pooling approach, an intermediate approach between estimating the parameters of each P-I curve independently and pooling all data together.

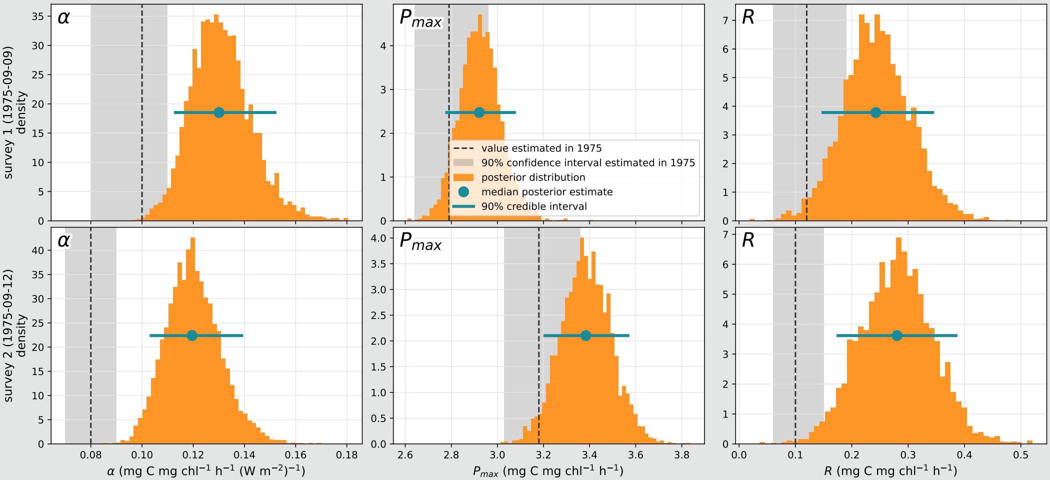


posterior estimates



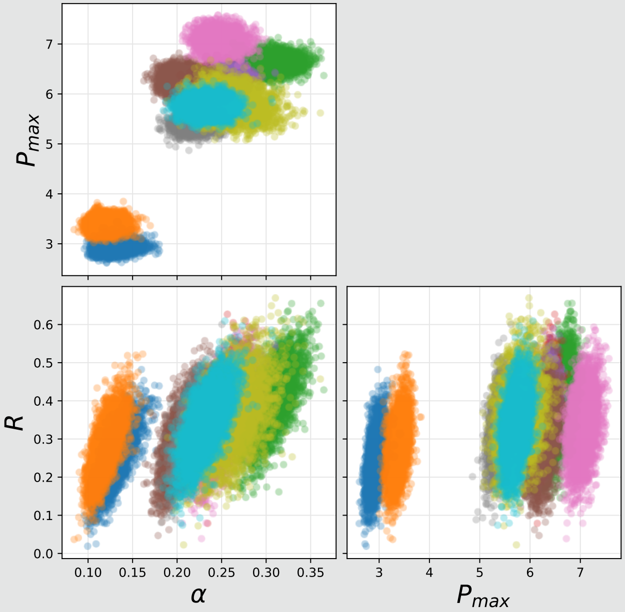
- The model was fit to a data from 107 surveys.
- On a regular laptop (AMD Ryzen 7 4750U), model fitting takes about 65 seconds.

posterior estimates



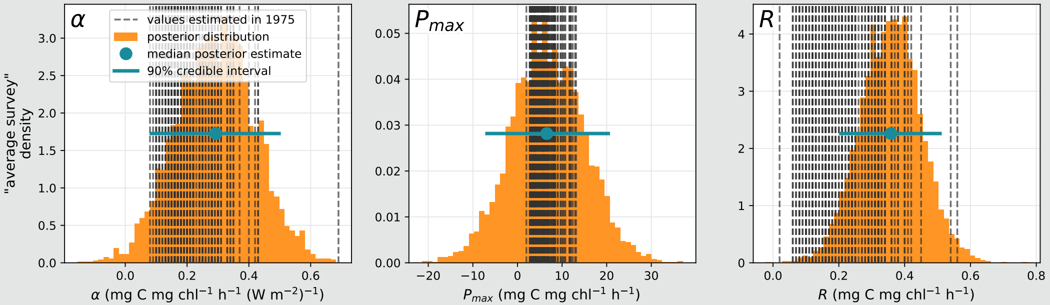
- The dataset contains estimated values for the P-I model parameters. These values were not used to inform the model.

posterior estimates



- The parameter correlations have intuitive meaning.
- Correlations between the estimated parameters can hint at identifiability issues.

“average survey” estimates



- The partial pooling approach provides parameter estimates that correspond to an average survey in the dataset.
- These “average survey” estimates correspond well to the full set of estimates made in 1975.

summary

- Parameter estimation is essential for many models, and with growing model complexity more processes will need to be parameterized.
- Parameter correlation is inevitable in complex models which can lead to parameter identifiability issues.
- Understanding uncertainty in parameter estimates is important for better understanding and communicating parameter estimation results.
- The Bayesian framework extends classical approaches by incorporating:
  - Prior knowledge of parameters.
  - Natural handling of non-Gaussian errors.
  - Full probability distributions rather than point estimates, which can help identify relationships between parameters and quantify uncertainty.

- Parameter estimation is essential for many models, and with growing model complexity more processes will need to be parameterized.
- Parameter correlation is inevitable in complex models which can lead to parameter identifiability issues.
- Understanding uncertainty in parameter estimates is important for better understanding and communicating parameter estimation results.
- The Bayesian framework extends classical approaches by incorporating:
  - Prior knowledge of parameters.
  - Natural handling of non-Gaussian errors.
  - Full probability distributions rather than point estimates, which can help identify relationships between parameters and quantify uncertainty.

Thanks! `jmattern@ucsc.edu`

additional slides

a partial pooling model

The `pymc` package can auto-generate model graphs laying out the random variables and data used in the model: