Inference on the Champagne Model using a Gaussian Process

TODO

- Change outputs

Setting up the Champagne Model

Imports

```
import pandas as pd
import numpy as np
from typing import Any
import matplotlib.pyplot as plt
import multiprocessing as mp
import pickle
import random

from scipy.stats import qmc
from scipy.stats import norm

import tensorflow as tf
import tensorflow_probability as tfp
from tensorflow_probability.python.distributions import normal

tfb = tfp.bijectors
tfd = tfp.distributions
tfk = tfp.math.psd_kernels
tfp_acq = tfp.experimental.bayesopt.acquisition
```

```
gpu_devices = tf.config.experimental.list_physical_devices("GPU")
for device in gpu_devices:
    tf.config.experimental.set_memory_growth(device, True)
```

```
2024-06-18 18:51:23.338532: I tensorflow/core/util/port.cc:113] oneDNN custom operations are
2024-06-18 18:51:23.405539: I tensorflow/core/platform/cpu_feature_guard.cc:210] This Tensorl
To enable the following instructions: AVX2 AVX512F AVX512_VNNI FMA, in other operations, rebu
```

Model itself

```
np.random.seed(590154)

population = 10000
initial_infecteds = 100
epidemic_length = 1000  # not used
number_of_events = 200000

pv_champ_alpha = 0.95 * 0.13  # prop of effective care
pv_champ_beta = 0.429  # prop of radical cure
pv_champ_gamma_L = 1 / 383  # liver stage clearance rate
pv_champ_delta = 0.05  # prop of imported cases
pv_champ_lambda = 0.01  # transmission rate
pv_champ_f = 1 / 69  # relapse frequency
pv_champ_r = 1 / 60  # blood stage clearance rate

gamma_L_max = 1 / 30
lambda_max = 0.05
f_max = 1 / 20
r_max = 1 / 15

upper_bounds = np.array([1, 1, gamma_L_max, lambda_max, f_max, r_max])

num_lhc_samples = 50
initial_repeats = 1
dis_mean_n = 30
```

```
def champagne_stochastic(
    alpha_,
    beta_,
```

```
    gamma_L,
    lambda_,
    f,
    r,
    N=population,
    I_L=initial_infecteds,
    I_0=0,
    S_L=0,
    delta_=0,
    # end_time=epidemic_length,
    num_events=number_of_events,
    seed=12,
):
    np.random.seed(seed)
    if (0 > (alpha_ or beta_)) or (1 < (alpha_ or beta_)):
        return "Alpha or Beta out of bounds"
    if 0 > (gamma_L or lambda_ or f or r):
        return "Gamma, lambda, f or r out of bounds"

    t = 0
    S_0 = N - I_L - I_0 - S_L
    inc_counter = 0

    list_of_outcomes = [
        {"t": 0, "S_0": S_0, "S_L": S_L, "I_0": I_0, "I_L": I_L, "inc_counter": 0}
    ]

    prop_new = alpha_ * beta_ * f / (alpha_ * beta_ * f + gamma_L)
    i = 0

    while (i < num_events) or (t < 30):
        i += 1
        if S_0 == N:
            while t < 31:
                t += 1
                new_stages = {
                    "t": t,
                    "S_0": N,
                    "S_L": 0,
                    "I_0": 0,
                    "I_L": 0,
```

```
                "inc_counter": inc_counter,
            }
            list_of_outcomes.append(new_stages)
        break

    S_0_to_I_L = (1 - alpha_) * lambda_ * (I_L + I_0) / N * S_0
    S_0_to_S_L = alpha_ * (1 - beta_) * lambda_ * (I_0 + I_L) / N * S_0
    I_0_to_S_0 = r * I_0 / N
    I_0_to_I_L = lambda_ * (I_L + I_0) / N * I_0
    I_L_to_I_0 = gamma_L * I_L
    I_L_to_S_L = r * I_L
    S_L_to_S_0 = (gamma_L + (f + lambda_ * (I_0 + I_L) / N) * alpha_ * beta_) * S_L
    S_L_to_I_L = (f + lambda_ * (I_0 + I_L) / N) * (1 - alpha_) * S_L

    total_rate = (
        S_0_to_I_L
        + S_0_to_S_L
        + I_0_to_S_0
        + I_0_to_I_L
        + I_L_to_I_0
        + I_L_to_S_L
        + S_L_to_S_0
        + S_L_to_I_L
    )

    delta_t = np.random.exponential(1 / total_rate)
    new_stages_prob = [
        S_0_to_I_L / total_rate,
        S_0_to_S_L / total_rate,
        I_0_to_S_0 / total_rate,
        I_0_to_I_L / total_rate,
        I_L_to_I_0 / total_rate,
        I_L_to_S_L / total_rate,
        S_L_to_S_0 / total_rate,
        S_L_to_I_L / total_rate,
    ]
    t += delta_t
    silent_incidences = np.random.poisson(
        delta_t * alpha_ * beta_ * lambda_ * (I_L + I_0) * S_0 / N
        + delta_t * alpha_ * (1 - beta_) * (f + lambda_ * (I_L + I_0) / N) * S_L
    )
```

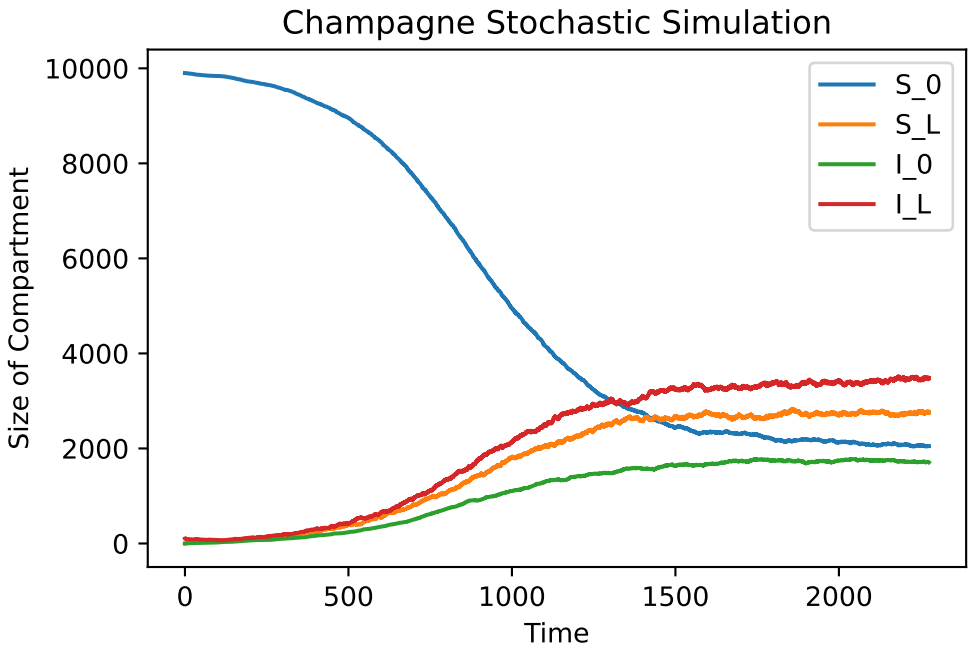
```
        new_stages = np.random.choice(
            [
                {
                    "t": t,
                    "S_0": S_0 - 1,
                    "S_L": S_L,
                    "I_0": I_0,
                    "I_L": I_L + 1,
                    "inc_counter": inc_counter + silent_incidences + 1,
                },
                {
                    "t": t,
                    "S_0": S_0 - 1,
                    "S_L": S_L + 1,
                    "I_0": I_0,
                    "I_L": I_L,
                    "inc_counter": inc_counter + silent_incidences + 1,
                },
                {
                    "t": t,
                    "S_0": S_0 + 1,
                    "S_L": S_L,
                    "I_0": I_0 - 1,
                    "I_L": I_L,
                    "inc_counter": inc_counter + silent_incidences,
                },
                {
                    "t": t,
                    "S_0": S_0,
                    "S_L": S_L,
                    "I_0": I_0 - 1,
                    "I_L": I_L + 1,
                    "inc_counter": inc_counter + silent_incidences,
                },
                {
                    "t": t,
                    "S_0": S_0,
                    "S_L": S_L,
                    "I_0": I_0 + 1,
                    "I_L": I_L - 1,
                    "inc_counter": inc_counter + silent_incidences,
```

```
                },
                {
                    "t": t,
                    "S_0": S_0,
                    "S_L": S_L + 1,
                    "I_0": I_0,
                    "I_L": I_L - 1,
                    "inc_counter": inc_counter + silent_incidences,
                },
                {
                    "t": t,
                    "S_0": S_0 + 1,
                    "S_L": S_L - 1,
                    "I_0": I_0,
                    "I_L": I_L,
                    "inc_counter": inc_counter
                    + silent_incidences
                    + np.random.binomial(1, prop_new),
                },
                {
                    "t": t,
                    "S_0": S_0,
                    "S_L": S_L - 1,
                    "I_0": I_0,
                    "I_L": I_L + 1,
                    "inc_counter": inc_counter + silent_incidences + 1,
                },
            ],
            p=new_stages_prob,
        )

        list_of_outcomes.append(new_stages)

        S_0 = new_stages["S_0"]
        I_0 = new_stages["I_0"]
        I_L = new_stages["I_L"]
        S_L = new_stages["S_L"]
        inc_counter = new_stages["inc_counter"]

    outcome_df = pd.DataFrame(list_of_outcomes)
    return outcome_df
```

```
champ_samp = champagne_stochastic(
    pv_champ_alpha,
    pv_champ_beta,
    pv_champ_gamma_L,
    pv_champ_lambda,
    pv_champ_f,
    pv_champ_r,
)
```

Plotting outcome

```
champ_samp.drop("inc_counter", axis=1).plot(x="t", legend=True)
plt.xlabel("Time")
plt.ylabel("Size of Compartment")
plt.title("Champagne Stochastic Simulation")
plt.savefig("champagne_GP_images/champagne_simulation.pdf", bbox_inches='tight')
plt.show()
```



Function that Outputs Final Prevalence

```
def incidence(df, start, days):
    start_ind = df[df["t"].le(start)].index[-1]
    end_ind = df[df["t"].le(start + days)].index[-1]
    incidence_week = df.iloc[end_ind]["inc_counter"] - df.iloc[start_ind]["inc_counter"]
    return incidence_week


def champ_sum_stats(alpha_, beta_, gamma_L, lambda_, f, r, seed=12301923):
    champ_df_ = champagne_stochastic(alpha_, beta_, gamma_L, lambda_, f, r, seed=seed)
    first_month_inc = incidence(champ_df_, 0, 30)
    fin_t = champ_df_.iloc[-1]["t"]
    fin_week_inc = incidence(champ_df_, fin_t - 7, 7)

    first_month_ind = champ_df_[champ_df_["t"].le(30)].index[-1]
    first_month_prev = (
        champ_df_.iloc[first_month_ind]["I_0"]
        + champ_df_.iloc[first_month_ind]["I_L"]
    )
    fin_prev = champ_df_.iloc[-1]["I_0"] + champ_df_.iloc[-1]["I_L"]

    return np.array([fin_week_inc, fin_prev, first_month_inc, first_month_prev])


observed_sum_stats = champ_sum_stats(
    pv_champ_alpha,
    pv_champ_beta,
    pv_champ_gamma_L,
    pv_champ_lambda,
    pv_champ_f,
    pv_champ_r,
)

print(observed_sum_stats)


def single_discrepency(alpha_, beta_, gamma_L, lambda_, f, r, seed=12301923):
    x = champ_sum_stats(alpha_, beta_, gamma_L, lambda_, f, r, seed=seed)
    return np.log(np.linalg.norm((x - observed_sum_stats) / observed_sum_stats))
```

```
def discrepency_fn(alpha_, beta_, gamma_L, lambda_, f, r, mean_of=dis_mean_n):
    seed = int(np.random.uniform() * 1000000)
    with mp.Pool(processes=mp.cpu_count()) as pool:
        args = [
            (alpha_, beta_, gamma_L, lambda_, f, r, seed * i) for i in range(mean_of)
        ]
        results = pool.starmap(single_discrepency, args)
    mean_obs = np.mean(results)
    return mean_obs
```

```
[ 461. 5205.   42.   87.]
```

Gaussian Process Regression on Final Prevalence Discrepency

```
my_seed = np.random.default_rng(seed=1795)  # For replicability

variables_names = ["alpha", "beta", "gamma_L", "lambda", "f", "r"]

LHC_sampler = qmc.LatinHypercube(d=6, seed=my_seed)
LHC_samples = LHC_sampler.random(n=num_lhc_samples)

# Using Champagne Initialisation table 2
LHC_samples[:, 2] = gamma_L_max * LHC_samples[:, 2]
LHC_samples[:, 3] = lambda_max * LHC_samples[:, 3]
LHC_samples[:, 4] = f_max * LHC_samples[:, 4]
LHC_samples[:, 5] = r_max * LHC_samples[:, 5]

LHC_samples = np.repeat(LHC_samples, initial_repeats, axis = 0)

LHC_indices_df = pd.DataFrame(LHC_samples, columns=variables_names)

print(LHC_indices_df.head())
```

```
      alpha      beta   gamma_L    lambda         f         r
0  0.100008  0.122349  0.009668  0.015376  0.016920  0.015954
1  0.659225  0.590955  0.001070  0.038947  0.007433  0.003318
2  0.503558  0.005003  0.031832  0.027053  0.002028  0.019736
3  0.011840  0.630562  0.023631  0.033488  0.035622  0.035127
4  0.271011  0.942434  0.014052  0.030138  0.031133  0.051736
```

Generate Discrepencies

```
LHC_samples_reps = np.repeat(LHC_samples, dis_mean_n, axis=0)

with mp.Pool(processes=mp.cpu_count()) as pool:
    args = [
        (a, b, c, d, e, f, int(g * np.random.uniform()))
        for (a, b, c, d, e, f), g in zip(
            list(map(tuple, LHC_samples_reps)), range(LHC_samples_reps.shape[0])
        )
    ]
    results = pool.starmap(single_discrepency, args)

random_discrepencies = np.mean(np.array(results).reshape(-1, dis_mean_n), axis=1)

print(random_discrepencies)
```

```
[-0.72112073  0.85173495 -0.10445358  1.04268897  0.56652755  0.46286618
  0.36628974  0.78458654  0.64547033 -0.43466706  0.72250049 -0.10120307
 -0.0321187  -0.16726528  1.26145031  0.92538462 -0.26121501  0.3277712
 -0.70371735  0.2155892   0.81173955  0.59710669  0.51707793  1.33971327
  1.0812211   0.8602283   1.38585448 -0.09602358  0.61782169  0.42402107
  0.31068428 -0.04423537 -0.96457358  0.38741823  0.24964537  0.14481727
  0.22145299  0.33272269 -0.67052626  1.12934722  1.21789051  0.20304941
  0.39821991  0.35048571  0.18218027  1.24579385 -0.49153258  0.34009135
  1.04344262 -0.03480152]
```

Differing Methods to Iterate Function

```
# import timeit

# def function1():
#     np.vectorize(champ_sum_stats)(random_indices_df['alpha'],
#     random_indices_df['beta'], random_indices_df['gamma_L'],
#     random_indices_df['lambda'], random_indices_df['f'], random_indices_df['r'])
#     pass

# def function2():
#     random_indices_df.apply(
#         lambda x: champ_sum_stats(
```

```
#             x['alpha'], x['beta'], x['gamma_L'], x['lambda'], x['f'], x['r']),
#             axis = 1)
#     pass

# # Time function1
# time_taken_function1 = timeit.timeit(
#     "function1()", globals=globals(), number=100)

# # Time function2
# time_taken_function2 = timeit.timeit(
#     "function2()", globals=globals(), number=100)

# print("Time taken for function1:", time_taken_function1)
# print("Time taken for function2:", time_taken_function2)
```

Time taken for function1: 187.48960775700016 Time taken for function2: 204.06618941299985

Constrain Variables to be Positive

```
constrain_positive = tfb.Shift(np.finfo(np.float64).tiny)(tfb.Exp())
```

```
2024-06-18 18:55:44.650270: I tensorflow/core/common_runtime/gpu/gpu_device.cc:1928] Created
2024-06-18 18:55:44.650900: I tensorflow/core/common_runtime/gpu/gpu_device.cc:1928] Created
2024-06-18 18:55:44.651312: I tensorflow/core/common_runtime/gpu/gpu_device.cc:1928] Created
2024-06-18 18:55:44.651684: I tensorflow/core/common_runtime/gpu/gpu_device.cc:1928] Created
```

Custom Quadratic Mean Function

```
class quad_mean_fn(tf.Module):
    def __init__(self):
        super(quad_mean_fn, self).__init__()
        # self.amp_alpha_mean = tfp.util.TransformedVariable(
        #     bijector=constrain_positive,
        #     initial_value=1.0,
        #     dtype=np.float64,
        #     name="amp_alpha_mean",
        # )
        # self.alpha_tp = tf.Variable(pv_champ_alpha, dtype=np.float64, name="alpha_tp")
        # self.amp_beta_mean = tfp.util.TransformedVariable(
```

```
#     bijector=constrain_positive,
#     initial_value=0.5,
#     dtype=np.float64,
#     name="amp_beta_mean",
# )
# self.beta_tp = tf.Variable(pv_champ_beta, dtype=np.float64, name="beta_tp")
self.amp_gamma_L_mean = tfp.util.TransformedVariable(
    bijector=constrain_positive,
    initial_value=1.0,
    dtype=np.float64,
    name="amp_gamma_L_mean",
)
# self.gamma_L_tp = tfp.util.TransformedVariable(
#     bijector=constrain_positive,
#     initial_value=1.0,
#     dtype=np.float64,
#     name="gamma_L_tp",
# )
self.amp_lambda_mean = tfp.util.TransformedVariable(
    bijector=constrain_positive,
    initial_value=1.0,
    dtype=np.float64,
    name="amp_lambda_mean",
)
# self.lambda_tp = tfp.util.TransformedVariable(
#     bijector=constrain_positive,
#     initial_value=1.0,
#     dtype=np.float64,
#     name="lambda_tp",
# )
self.amp_f_mean = tfp.util.TransformedVariable(
    bijector=constrain_positive,
    initial_value=1.0,
    dtype=np.float64,
    name="amp_f_mean",
)
# self.f_tp = tfp.util.TransformedVariable(
#     bijector=constrain_positive,
#     initial_value=1.0,
#     dtype=np.float64,
#     name="f_tp",
```

```
        # )
        self.amp_r_mean = tfp.util.TransformedVariable(
            bijector=constrain_positive,
            initial_value=1.0,
            dtype=np.float64,
            name="amp_r_mean",
        )
        # self.r_tp = tfp.util.TransformedVariable(
        #     bijector=constrain_positive,
        #     initial_value=1.0,
        #     dtype=np.float64,
        #     name="r_tp",
        # )
        # self.bias_mean = tfp.util.TransformedVariable(
        #     bijector=constrain_positive,
        #     initial_value=1.0,
        #     dtype=np.float64,
        #     name="bias_mean",
        # )
        self.bias_mean = tf.Variable(-1.5, dtype=np.float64, name="bias_mean")

    def __call__(self, x):
        return (
            self.bias_mean
            # + self.amp_alpha_mean * (x[..., 0] - self.alpha_tp) ** 2
            # + self.amp_beta_mean * (x[..., 1] - self.beta_tp) ** 2
            # + self.amp_gamma_L_mean * (x[..., 2] - self.gamma_L_tp) ** 2
            # + self.amp_lambda_mean * (x[..., 3] - self.lambda_tp) ** 2
            # + self.amp_f_mean * (x[..., 4] - self.f_tp) ** 2
            # + self.amp_r_mean * (x[..., 5] - self.r_tp) ** 2
            + self.amp_gamma_L_mean * (x[..., 2]) ** 2
            + self.amp_lambda_mean * (x[..., 3]) ** 2
            + self.amp_f_mean * (x[..., 4]) ** 2
            + self.amp_r_mean * (x[..., 5]) ** 2
        )


quad_mean_fn().__call__(x=np.array([[1.0, 1.0, 1.0, 1.0, 1.0, 1.0]]))  # should return 1
```

```
<tf.Tensor: shape=(1,), dtype=float64, numpy=array([2.5])>
```

Custom Linear Mean Function

```
class lin_mean_fn(tf.Module):
    def __init__(self):
        super(lin_mean_fn, self).__init__()
        # self.amp_alpha_lin = tfp.util.TransformedVariable(
        #     bijector=constrain_positive,
        #     initial_value=1.0,
        #     dtype=np.float64,
        #     name="amp_alpha_lin",
        # )
        # self.amp_beta_lin = tfp.util.TransformedVariable(
        #     bijector=constrain_positive,
        #     initial_value=0.5,
        #     dtype=np.float64,
        #     name="amp_beta_lin",
        # )
        self.amp_gamma_L_lin = tfp.util.TransformedVariable(
            bijector=constrain_positive,
            initial_value=1.0,
            dtype=np.float64,
            name="amp_gamma_L_lin",
        )
        self.amp_lambda_lin = tfp.util.TransformedVariable(
            bijector=constrain_positive,
            initial_value=1.0,
            dtype=np.float64,
            name="amp_lambda_lin",
        )
        self.amp_f_lin = tfp.util.TransformedVariable(
            bijector=constrain_positive,
            initial_value=1.0,
            dtype=np.float64,
            name="amp_f_lin",
        )
        self.amp_r_lin = tfp.util.TransformedVariable(
            bijector=constrain_positive,
            initial_value=1.0,
            dtype=np.float64,
            name="amp_r_lin",
        )
```

```
        # self.bias_lin = tfp.util.TransformedVariable(
        #     bijector=constrain_positive,
        #     initial_value=1.0,
        #     dtype=np.float64,
        #     name="bias_lin",
        # )
        self.bias_lin = tf.Variable(0.0, dtype=np.float64, name="bias_mean")

    def __call__(self, x):
        return (
            self.bias_lin
            # + self.amp_alpha_lin * (x[..., 0])
            # + self.amp_beta_lin * (x[..., 1])
            + self.amp_gamma_L_lin * (x[..., 2])
            + self.amp_lambda_lin * (x[..., 3])
            + self.amp_f_lin * (x[..., 4])
            + self.amp_r_lin * (x[..., 5])
        )
```

```
class const_mean_fn(tf.Module):
    def __init__(self):
        super(const_mean_fn, self).__init__()
        self.bias_lin = tf.Variable(0.0, dtype=np.float64, name="bias_mean")

    def __call__(self, x):
        return self.bias_lin
```

Making the ARD Kernel

```
index_vals = LHC_indices_df.values
obs_vals = random_discrepencies

amplitude_champ = tfp.util.TransformedVariable(
    bijector=constrain_positive,
    initial_value=4.0,
    dtype=np.float64,
    name="amplitude_champ",
)
```

```
observation_noise_variance_champ = tfp.util.TransformedVariable(
    bijector=constrain_positive,
    initial_value=1.,
    dtype=np.float64,
    name="observation_noise_variance_champ",
)
```

```
length_scales_champ = tfp.util.TransformedVariable(
    # bijector=tfb.Sigmoid(
    #     np.float64(0.0),
    #     [1.0 / 2, 1.0 / 2, gamma_L_max / 2, lambda_max / 2, f_max / 2, r_max / 2],
    # ),
    bijector=constrain_positive,
    initial_value=[1 / 8, 1 / 8, gamma_L_max / 8, lambda_max / 8, f_max / 8, r_max / 8],
    dtype=np.float64,
    name="length_scales_champ",
)
```

```
kernel_champ = tfk.FeatureScaled(
    tfk.MaternFiveHalves(amplitude=amplitude_champ),
    scale_diag=length_scales_champ,
)
```

Define the Gaussian Process with Quadratic Mean Function and ARD Kernel

```
# Define Gaussian Process with the custom kernel
champ_GP = tfd.GaussianProcess(
    kernel=kernel_champ,
    observation_noise_variance=observation_noise_variance_champ,
    index_points=index_vals,
    mean_fn=const_mean_fn(),
)

print(champ_GP.trainable_variables)

Adam_optim = tf.keras.optimizers.Adam(learning_rate=0.01)
```

```
(<tf.Variable 'amplitude_champ:0' shape=() dtype=float64, numpy=1.3862943611198906>, <tf.Vari
array([-2.07944154, -2.07944154, -5.48063892, -5.07517382, -5.07517382,
       -4.78749174])>, <tf.Variable 'observation_noise_variance_champ:0' shape=() dtype=float
```

Train the Hyperparameters

Leave One Out Predictive Log-likelihood

```
# predictive log stuff
@tf.function(autograph=False, jit_compile=False)
def optimize():
    with tf.GradientTape() as tape:
        K = (
            champ_GP.kernel.matrix(index_vals, index_vals)
            + tf.eye(index_vals.shape[0], dtype=np.float64)
            * observation_noise_variance_champ
        )
        means = champ_GP.mean_fn(index_vals)
        K_inv = tf.linalg.inv(K)
        K_inv_y = K_inv @ tf.reshape(obs_vals - means, shape=[obs_vals.shape[0], 1])
        K_inv_diag = tf.linalg.diag_part(K_inv)
        log_var = tf.math.log(K_inv_diag)
        log_mu = tf.reshape(K_inv_y, shape=[-1]) ** 2
        loss = -tf.math.reduce_sum(log_var - log_mu)
    grads = tape.gradient(loss, champ_GP.trainable_variables)
    Adam_optim.apply_gradients(zip(grads, champ_GP.trainable_variables))
    return loss


num_iters = 10000

lls_ = np.zeros(num_iters, np.float64)
tolerance = 1e-6  # Set your desired tolerance level
previous_loss = float("inf")

for i in range(num_iters):
    loss = optimize()
    lls_[i] = loss

    # Check if change in loss is less than tolerance
    if abs(loss - previous_loss) < tolerance:
        print(f"Hyperparameter convergence reached at iteration {i+1}.")
        lls_ = lls_[range(i + 1)]
        break
```

```
    previous_loss = loss
```

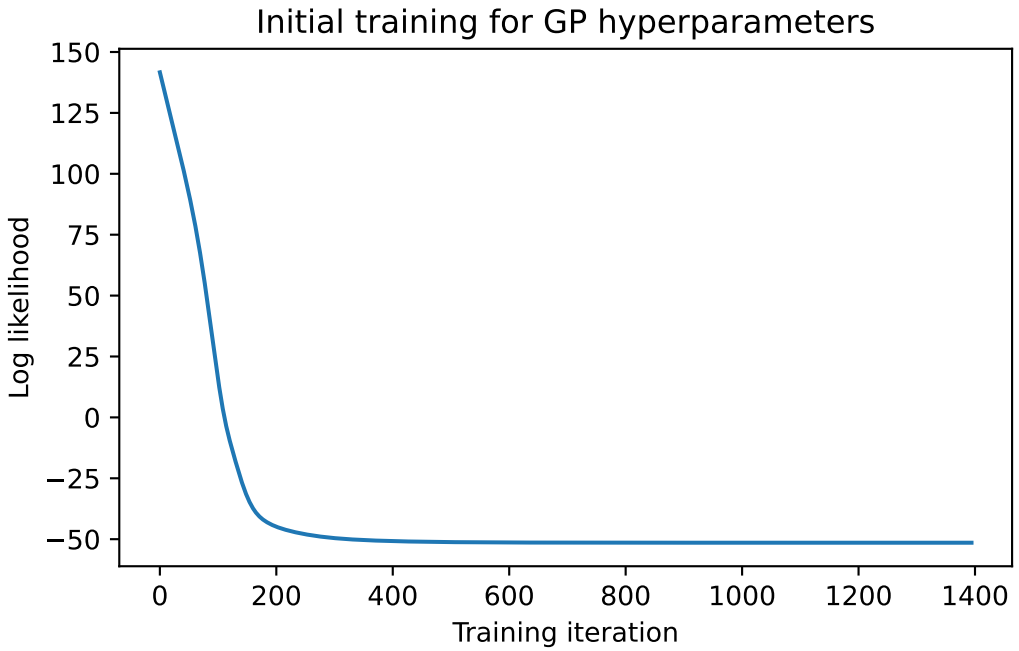
```
2024-06-18 18:55:49.305965: I tensorflow/core/util/cuda_solvers.cc:178] Creating GpuSolver ha
```

```
Hyperparameter convergence reached at iteration 1395.
```

```
print("Trained parameters:")
for var in champ_GP.trainable_variables:
    if "bias" in var.name:
        print("{} is {}\n".format(var.name, var.numpy().round(3)))
    else:
        # if "length" in var.name:
        #     print(
        #         "{} is {}\n".format(
        #             var.name,
        #             tfb.Sigmoid(
        #                 np.float64(0.0),
        #                 [
        #                     1.0 / 2,
        #                     1.0 / 2,
        #                     gamma_L_max / 2,
        #                     lambda_max / 2,
        #                     f_max / 2,
        #                     r_max / 2,
        #                 ],
        #             )
        #             .forward(var)
        #             .numpy()
        #             .round(3),
        #         )
        #     )
        # else:
        print(
            "{} is {}\n".format(
                var.name, constrain_positive.forward(var).numpy().round(3)
            )
        )

initial_losses_LOOCV = lls_
```

```
Trained parameters:
amplitude_champ:0 is 1.526

length_scales_champ:0 is [0.914 3.298 0.076 0.014 0.061 0.075]

observation_noise_variance_champ:0 is 0.018

bias_mean:0 is 0.898
```

```
plt.figure(figsize=(6, 3.5))
plt.plot(lls_)
plt.title("Initial training for GP hyperparameters")
plt.xlabel("Training iteration")
plt.ylabel("Log likelihood")
plt.savefig("champagne_GP_images/hyperparam_loss_log_discrep.pdf", bbox_inches="tight")
plt.show()
```



Creating slices across one variable dimension

```
plot_samp_no = 21
plot_samp_times = 10
plot_gp_no = 100
gp_samp_no = 30
```

```
slice_samples_dict = {
    "alpha_slice_samples": np.repeat(
        np.concatenate(
            (
                np.linspace(0, 1, plot_samp_no, dtype=np.float64).reshape(
                    -1, 1
                ),  # alpha
                np.repeat(pv_champ_beta, plot_samp_no).reshape(-1, 1),  # beta
                np.repeat(pv_champ_gamma_L, plot_samp_no).reshape(-1, 1),  # gamma_L
                np.repeat(pv_champ_lambda, plot_samp_no).reshape(-1, 1),  # lambda
                np.repeat(pv_champ_f, plot_samp_no).reshape(-1, 1),  # f
                np.repeat(pv_champ_r, plot_samp_no).reshape(-1, 1),  # r
            ),
            axis=1,
        ),
        plot_samp_times,
        axis=0,
    ),
    "alpha_gp_samples": np.concatenate(
        (
            np.linspace(0, 1, plot_gp_no, dtype=np.float64).reshape(-1, 1),  # alpha
            np.repeat(pv_champ_beta, plot_gp_no).reshape(-1, 1),  # beta
            np.repeat(pv_champ_gamma_L, plot_gp_no).reshape(-1, 1),  # gamma_L
            np.repeat(pv_champ_lambda, plot_gp_no).reshape(-1, 1),  # lambda
            np.repeat(pv_champ_f, plot_gp_no).reshape(-1, 1),  # f
            np.repeat(pv_champ_r, plot_gp_no).reshape(-1, 1),  # r
        ),
        axis=1,
    ),
    "beta_slice_samples": np.repeat(
        np.concatenate(
            (
                np.repeat(pv_champ_alpha, plot_samp_no).reshape(-1, 1),  # alpha
                np.linspace(0, 1, plot_samp_no, dtype=np.float64).reshape(
```

```
                -1, 1
            ),  # beta
            np.repeat(pv_champ_gamma_L, plot_samp_no).reshape(-1, 1),  # gamma_L
            np.repeat(pv_champ_lambda, plot_samp_no).reshape(-1, 1),  # lambda
            np.repeat(pv_champ_f, plot_samp_no).reshape(-1, 1),  # f
            np.repeat(pv_champ_r, plot_samp_no).reshape(-1, 1),  # r
        ),
        axis=1,
    ),
    plot_samp_times,
    axis=0,
),
"beta_gp_samples": np.concatenate(
    (
        np.repeat(pv_champ_alpha, plot_gp_no).reshape(-1, 1),  # alpha
        np.linspace(0, 1, plot_gp_no, dtype=np.float64).reshape(-1, 1),  # beta
        np.repeat(pv_champ_gamma_L, plot_gp_no).reshape(-1, 1),  # gamma_L
        np.repeat(pv_champ_lambda, plot_gp_no).reshape(-1, 1),  # lambda
        np.repeat(pv_champ_f, plot_gp_no).reshape(-1, 1),  # f
        np.repeat(pv_champ_r, plot_gp_no).reshape(-1, 1),  # r
    ),
    axis=1,
),
"gamma_L_slice_samples": np.repeat(
    np.concatenate(
        (
            np.repeat(pv_champ_alpha, plot_samp_no).reshape(-1, 1),  # alpha
            np.repeat(pv_champ_beta, plot_samp_no).reshape(-1, 1),  # beta
            np.linspace(0, gamma_L_max, plot_samp_no, dtype=np.float64).reshape(
                -1, 1
            ),  # gamma_L
            np.repeat(pv_champ_lambda, plot_samp_no).reshape(-1, 1),  # lambda
            np.repeat(pv_champ_f, plot_samp_no).reshape(-1, 1),  # f
            np.repeat(pv_champ_r, plot_samp_no).reshape(-1, 1),  # r
        ),
        axis=1,
    ),
    plot_samp_times,
    axis=0,
),
"gamma_L_gp_samples": np.concatenate(
```

```
        (
            np.repeat(pv_champ_alpha, plot_gp_no).reshape(-1, 1),  # alpha
            np.repeat(pv_champ_beta, plot_gp_no).reshape(-1, 1),  # beta
            np.linspace(0, gamma_L_max, plot_gp_no, dtype=np.float64).reshape(
                -1, 1
            ),  # gamma_L
            np.repeat(pv_champ_lambda, plot_gp_no).reshape(-1, 1),  # lambda
            np.repeat(pv_champ_f, plot_gp_no).reshape(-1, 1),  # f
            np.repeat(pv_champ_r, plot_gp_no).reshape(-1, 1),  # r
        ),
        axis=1,
    ),
    "lambda_slice_samples": np.repeat(
        np.concatenate(
            (
                np.repeat(pv_champ_alpha, plot_samp_no).reshape(-1, 1),  # alpha
                np.repeat(pv_champ_beta, plot_samp_no).reshape(-1, 1),  # beta
                np.repeat(pv_champ_gamma_L, plot_samp_no).reshape(-1, 1),  # gamma_L
                np.linspace(0, lambda_max, plot_samp_no, dtype=np.float64).reshape(
                    -1, 1
                ),  # lambda
                np.repeat(pv_champ_f, plot_samp_no).reshape(-1, 1),  # f
                np.repeat(pv_champ_r, plot_samp_no).reshape(-1, 1),  # r
            ),
            axis=1,
        ),
        plot_samp_times,
        axis=0,
    ),
    "lambda_gp_samples": np.concatenate(
        (
            np.repeat(pv_champ_alpha, plot_gp_no).reshape(-1, 1),  # alpha
            np.repeat(pv_champ_beta, plot_gp_no).reshape(-1, 1),  # beta
            np.repeat(pv_champ_gamma_L, plot_gp_no).reshape(-1, 1),  # gamma_L
            np.linspace(0, lambda_max, plot_gp_no, dtype=np.float64).reshape(
                -1, 1
            ),  # lambda
            np.repeat(pv_champ_f, plot_gp_no).reshape(-1, 1),  # f
            np.repeat(pv_champ_r, plot_gp_no).reshape(-1, 1),  # r
        ),
        axis=1,
```

```
    ),
    "f_slice_samples": np.repeat(
        np.concatenate(
            (
                np.repeat(pv_champ_alpha, plot_samp_no).reshape(-1, 1),  # alpha
                np.repeat(pv_champ_beta, plot_samp_no).reshape(-1, 1),  # beta
                np.repeat(pv_champ_gamma_L, plot_samp_no).reshape(-1, 1),  # gamma_L
                np.repeat(pv_champ_lambda, plot_samp_no).reshape(-1, 1),  # lambda
                np.linspace(0, f_max, plot_samp_no, dtype=np.float64).reshape(
                    -1, 1
                ),  # f
                np.repeat(pv_champ_r, plot_samp_no).reshape(-1, 1),  # r
            ),
            axis=1,
        ),
        plot_samp_times,
        axis=0,
    ),
    "f_gp_samples": np.concatenate(
        (
            np.repeat(pv_champ_alpha, plot_gp_no).reshape(-1, 1),  # alpha
            np.repeat(pv_champ_beta, plot_gp_no).reshape(-1, 1),  # beta
            np.repeat(pv_champ_gamma_L, plot_gp_no).reshape(-1, 1),  # gamma_L
            np.repeat(pv_champ_lambda, plot_gp_no).reshape(-1, 1),  # lambda
            np.linspace(0, f_max, plot_gp_no, dtype=np.float64).reshape(-1, 1),  # f
            np.repeat(pv_champ_r, plot_gp_no).reshape(-1, 1),  # r
        ),
        axis=1,
    ),
    "r_slice_samples": np.repeat(
        np.concatenate(
            (
                np.repeat(pv_champ_alpha, plot_samp_no).reshape(-1, 1),  # alpha
                np.repeat(pv_champ_beta, plot_samp_no).reshape(-1, 1),  # beta
                np.repeat(pv_champ_gamma_L, plot_samp_no).reshape(-1, 1),  # gamma_L
                np.repeat(pv_champ_lambda, plot_samp_no).reshape(-1, 1),  # lambda
                np.repeat(pv_champ_f, plot_samp_no).reshape(-1, 1),  # f
                np.linspace(0, r_max, plot_samp_no, dtype=np.float64).reshape(
                    -1, 1
                ),  # r
            ),
```

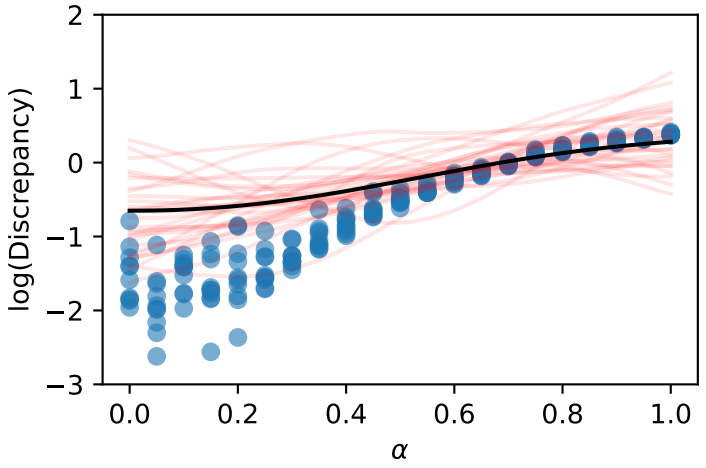
```
            axis=1,
        ),
        plot_samp_times,
        axis=0,
    ),
    "r_gp_samples": np.concatenate(
        (
            np.repeat(pv_champ_alpha, plot_gp_no).reshape(-1, 1),  # alpha
            np.repeat(pv_champ_beta, plot_gp_no).reshape(-1, 1),  # beta
            np.repeat(pv_champ_gamma_L, plot_gp_no).reshape(-1, 1),  # gamma_L
            np.repeat(pv_champ_lambda, plot_gp_no).reshape(-1, 1),  # lambda
            np.repeat(pv_champ_f, plot_gp_no).reshape(-1, 1),  # f
            np.linspace(0, r_max, plot_gp_no, dtype=np.float64).reshape(-1, 1),  # r
        ),
        axis=1,
    ),
}
```

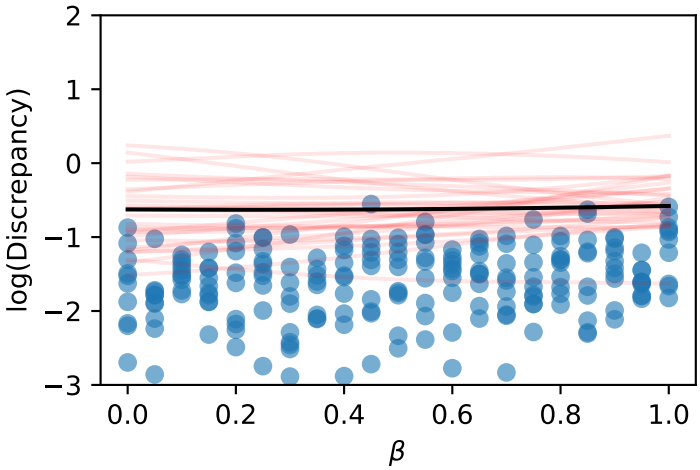
Plotting the GPs across different slices

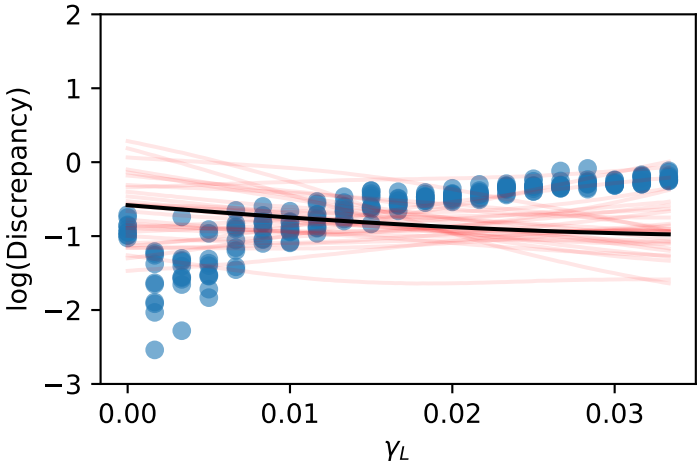
```
GP_seed = tfp.random.sanitize_seed(4362)
vars = ["alpha", "beta", "gamma_L", "lambda", "f", "r"]
slice_indices_dfs_dict = {}
slice_index_vals_dict = {}
slice_discrepencies_dict = {}
gp_samples_dict = {}

for var in vars:
    val_df = pd.DataFrame(
        slice_samples_dict[var + "_slice_samples"], columns=variables_names
    )
    slice_indices_dfs_dict[var + "_slice_indices_df"] = val_df
    slice_index_vals_dict[var + "_slice_index_vals"] = val_df.values

    df_temp = val_df.assign(seed=range(val_df.shape[0]))
    seed = int(np.random.uniform() * 1000000)
    with mp.Pool(processes=mp.cpu_count()) as pool:
        args = list(df_temp.itertuples(index=False, name=None))
        results = pool.starmap(single_discrepency, args)
```

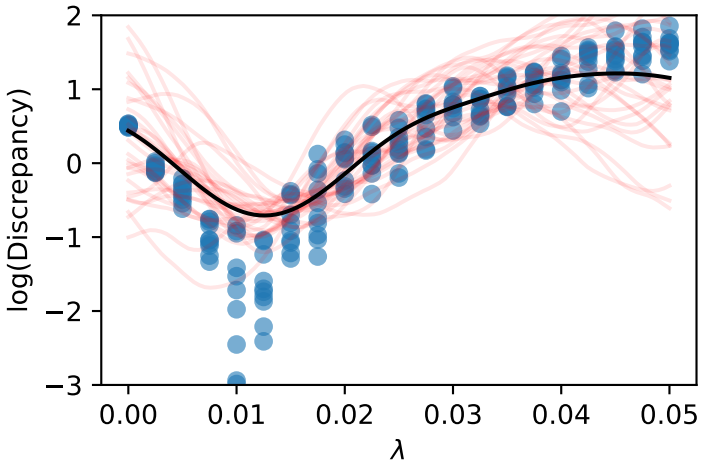
```
discreps = results
slice_discrepencies_dict[var + "_slice_discrepencies"] = discreps

gp_samples_df = pd.DataFrame(
    slice_samples_dict[var + "_gp_samples"], columns=variables_names
)
slice_indices_dfs_dict[var + "_gp_indices_df"] = gp_samples_df
slice_index_vals_dict[var + "_gp_index_vals"] = gp_samples_df.values

champ_GP_reg_plot = tfd.GaussianProcessRegressionModel(
    kernel=kernel_champ,
    index_points=gp_samples_df.values,
    observation_index_points=index_vals,
    observations=obs_vals,
    observation_noise_variance=observation_noise_variance_champ,
    predictive_noise_variance=0.0,
    mean_fn=const_mean_fn(),
)
GP_samples = champ_GP_reg_plot.sample(gp_samp_no, seed=GP_seed)

gp_samples_dict[var + "initial_gp_samps"] = GP_samples

plt.figure(figsize=(4, 2.5))
plt.scatter(
    val_df[var].values,
    discreps,
    label="Untrained Discrepencies",
    alpha=0.6,
)
for i in range(gp_samp_no):
    plt.plot(
        gp_samples_df[var].values,
        GP_samples[i, :],
        c="r",
        alpha=0.1,
        label="Posterior Sample" if i == 0 else None,
    )
plt.plot(
    slice_indices_dfs_dict[var + "_gp_indices_df"][var].values,
    champ_GP_reg_plot.mean_fn(
        slice_indices_dfs_dict[var + "_gp_indices_df"].values
```

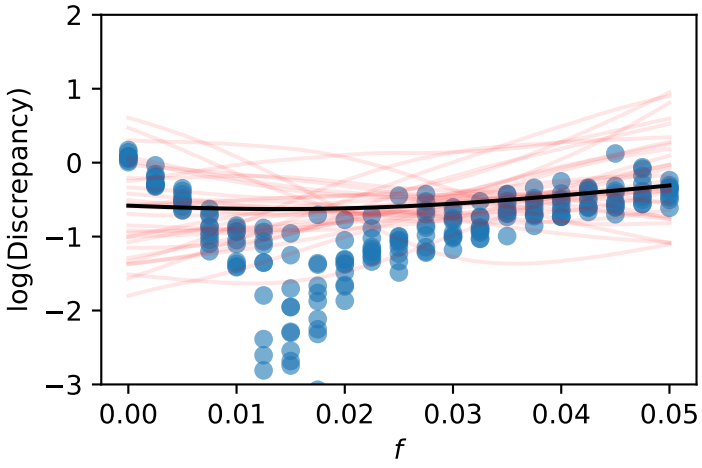
```
    ),
    c="black",
    alpha=1,
    label="Posterior Mean",
)
# leg = plt.legend(loc="upper left")
# for lh in leg.legend_handles:
#     lh.set_alpha(1)
if var in ["f", "r"]:
    plt.xlabel("$" + var + "$")
    # plt.title("$" + var + "$ slice before Bayesian Acquisition")
else:
    plt.xlabel("$\\" + var + "$")
    # plt.title("$\\" + var + "$ slice before Bayesian Acquisition")
# if var not in ["alpha", "beta"]:
#     plt.xscale("log", base=np.e)
plt.ylabel("log(Discrepancy)")
plt.ylim((-3, 2))
plt.savefig(
    "champagne_GP_images/initial_" + var + "_slice_log_discrep.pdf",
    bbox_inches="tight",
)
plt.show()
```

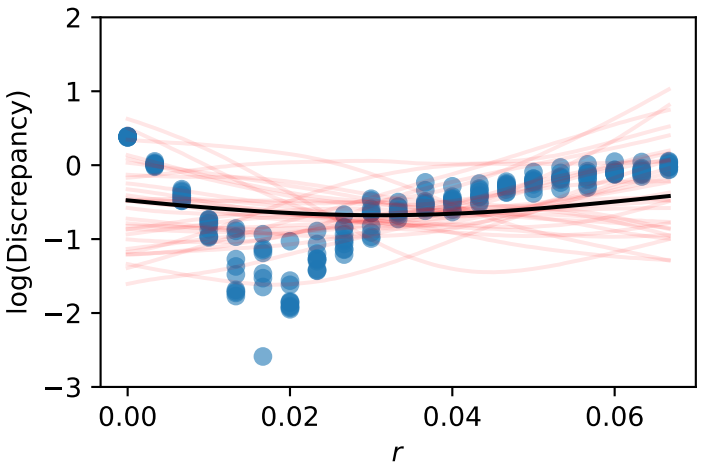



log(Discrepancy)
2
1
0
−1
−2
−3
0.0
0.2
0.4
0.6
0.8
1.0
β


log(Discrepancy)
2
1
0
−1
−2
−3
0.00
0.01
0.02
0.03
γ_L


log(Discrepancy)
2
1
0
−1
−2
−3
0.00
0.01
0.02
0.03
0.04
0.05
λ


log(Discrepancy)
2
1
0
−1
−2
−3
0.00
0.01
0.02
0.03
0.04
0.05
f



Acquiring the next datapoint to test

Proof that .variance returns what we need in acquisition function

```
champ_GP_reg = tfd.GaussianProcessRegressionModel(
    kernel=kernel_champ,
    observation_index_points=index_vals,
    observations=obs_vals,
    observation_noise_variance=observation_noise_variance_champ,
    mean_fn=const_mean_fn(),
)

new_guess = np.array([0.4, 0.4, 0.004, 0.04, 0.01, 0.17])
mean_t = champ_GP_reg.mean_fn(new_guess)
variance_t = champ_GP_reg.variance(index_points=[new_guess])

kernel_self = kernel_champ.apply(new_guess, new_guess)
kernel_others = kernel_champ.apply(new_guess, index_vals)
K = kernel_champ.matrix(
    index_vals, index_vals
) + observation_noise_variance_champ * np.identity(index_vals.shape[0])
inv_K = np.linalg.inv(K)
print("Self Kernel is {}".format(kernel_self.numpy().round(3)))
print("Others Kernel is {}".format(kernel_others.numpy().round(3)))
print(inv_K)
```

```
  my_var_t = kernel_self - kernel_others.numpy() @ inv_K @ kernel_others.numpy()

  print("Variance function is {}".format(variance_t.numpy().round(3)))
  print("Variance function is {}".format(my_var_t.numpy().round(3)))
```

```
Self Kernel is 2.328
Others Kernel is [0.096 0.221 0.217 0.336 0.441 0.556 0.074 0.265 0.167 0.091 0.464 0.044
 0.085 0.245 0.529 0.578 0.149 0.712 0.066 0.313 0.369 0.221 0.614 0.368
 0.224 0.297 0.495 0.145 0.185 0.02  0.202 0.072 0.103 0.036 0.056 0.089
 0.043 0.093 0.086 0.576 0.613 0.032 0.192 0.139 0.061 0.573 0.066 0.247
 0.216 0.189]
[[ 3.74331623e+00 -1.79222250e-02 -4.92029511e-02 ... -2.68667304e-02
   3.64973536e-03 -1.27791184e+00]
 [-1.79222250e-02  2.81344520e+00  1.52743886e-01 ...  3.71902870e-01
  -1.15218941e+00 -1.96118052e-01]
 [-4.92029511e-02  1.52743886e-01  3.32181636e+00 ... -1.19911743e+00
  -1.01439904e-02 -1.78146988e-01]
 ...
 [-2.68667304e-02  3.71902870e-01 -1.19911743e+00 ...  4.68435369e+00
  -6.18945598e-02  1.72952538e-01]
 [ 3.64973536e-03 -1.15218941e+00 -1.01439904e-02 ... -6.18945598e-02
   2.63445217e+00  3.72076630e-02]
 [-1.27791184e+00 -1.96118052e-01 -1.78146988e-01 ...  1.72952538e-01
   3.72076630e-02  4.03959172e+00]]
Variance function is [2.025]
Variance function is 2.007
```

Loss function

```
  next_alpha = tfp.util.TransformedVariable(
      initial_value=0.5,
      bijector=tfb.Sigmoid(),
      dtype=np.float64,
      name="next_alpha",
  )

  next_beta = tfp.util.TransformedVariable(
      initial_value=0.5,
      bijector=tfb.Sigmoid(),
      dtype=np.float64,
```

```
    name="next_beta",
)

next_gamma_L = tfp.util.TransformedVariable(
    initial_value=gamma_L_max/2,
    bijector=tfb.Sigmoid(np.float64(0.), gamma_L_max),
    dtype=np.float64,
    name="next_gamma_L",
)

next_lambda = tfp.util.TransformedVariable(
    initial_value=lambda_max/2,
    bijector=tfb.Sigmoid(np.float64(0.), lambda_max),
    dtype=np.float64,
    name="next_lambda",
)

next_f = tfp.util.TransformedVariable(
    initial_value=f_max/2,
    bijector=tfb.Sigmoid(np.float64(0.), f_max),
    dtype=np.float64,
    name="next_f",
)

next_r = tfp.util.TransformedVariable(
    initial_value=r_max/2,
    bijector=tfb.Sigmoid(np.float64(0.), r_max),
    dtype=np.float64,
    name="next_r",
)

next_vars = (
    (next_alpha.trainable_variables[0],
    next_beta.trainable_variables[0],
    next_gamma_L.trainable_variables[0],
    next_lambda.trainable_variables[0],
    next_f.trainable_variables[0],
    next_r.trainable_variables[0],)
)

next_vars
```

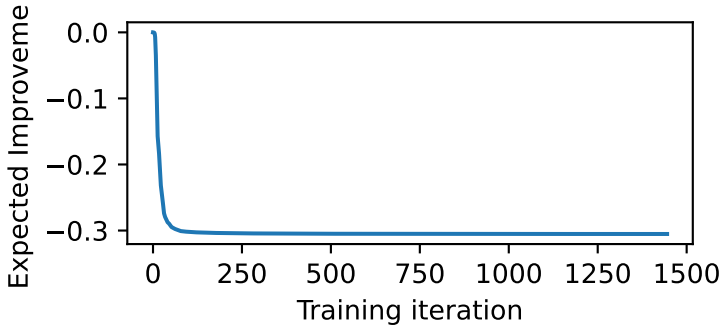
```
(<tf.Variable 'next_alpha:0' shape=() dtype=float64, numpy=0.0>,
 <tf.Variable 'next_beta:0' shape=() dtype=float64, numpy=0.0>,
 <tf.Variable 'next_gamma_L:0' shape=() dtype=float64, numpy=0.0>,
 <tf.Variable 'next_lambda:0' shape=() dtype=float64, numpy=0.0>,
 <tf.Variable 'next_f:0' shape=() dtype=float64, numpy=0.0>,
 <tf.Variable 'next_r:0' shape=() dtype=float64, numpy=0.0>)
```

```
curr_min = min(champ_GP_reg.mean_fn(index_vals))

def EI_loss(champ_GP_reg):
    next_guess = tf.reshape(
        tf.stack([next_alpha, next_beta, next_gamma_L, next_lambda, next_f, next_r]),
        [1, 6],
    )
    mean_t = champ_GP_reg.mean_fn(next_guess)
    std_t = champ_GP_reg.stddev(index_points=next_guess)
    delt = curr_min - mean_t
    return -tf.squeeze(
        delt * tfd.Normal(0, np.float64(1)).cdf(delt / std_t)
        + std_t * tfd.Normal(0, np.float64(1)).prob(delt / std_t)
    )

optimizer_fast = tf.keras.optimizers.Adam(learning_rate=0.1)

@tf.function(autograph=False, jit_compile=False)
def opt_var():
    with tf.GradientTape() as tape:
        loss = EI_loss(champ_GP_reg)
    grads = tape.gradient(loss, next_vars)
    optimizer_fast.apply_gradients(zip(grads, next_vars))
    return loss

num_iters = 10000

lls_ = np.zeros(num_iters, np.float64)
tolerance = 1e-7  # Set your desired tolerance level
previous_loss = np.float64("inf")

for i in range(num_iters):
    loss = opt_var()
    lls_[i] = loss
```

```
    # Check if change in loss is less than tolerance
    if abs(loss - previous_loss) < tolerance:
        print(f"Acquisition function convergence reached at iteration {i+1}.")
        lls_ = lls_[range(i + 1)]
        break

    previous_loss = loss

next_guess = tf.reshape(
    tf.stack([next_alpha, next_beta, next_gamma_L, next_lambda, next_f, next_r]),
    [1, 6],
)
print(
    "The final EI loss was {}".format(loss.numpy().round(3))
    + " with predicted mean of {}".format(
        champ_GP_reg.mean_fn(next_guess).numpy().round(3)
    )
)
```

```
Acquisition function convergence reached at iteration 1446.
The final EI loss was -0.305 with predicted mean of [-1.044]
```

```
plt.figure(figsize=(3.8, 1.5))
plt.plot(lls_)
plt.xlabel("Training iteration")
plt.ylabel("Expected Improvement")
plt.savefig("champagne_GP_images/initial_EI_loss_training.pdf", bbox_inches="tight")
plt.show()
```



```
# eta_t = tf.constant(1.0, dtype=np.float64)

# def UCB_loss(champ_GP_reg):
#     next_guess = tf.reshape(
#         tf.stack([next_alpha, next_beta, next_gamma_L, next_lambda, next_f, next_r]),
#         [1, 6],
#     )
#     mean_t = champ_GP_reg.mean_fn(next_guess)
#     std_t = tf.math.sqrt(
#         champ_GP_reg.variance(index_points=next_guess)
#         - observation_noise_variance_champ
#     )
#     return tf.squeeze(mean_t - std_t)


# optimizer_fast = tf.keras.optimizers.Adam(learning_rate=0.1)

# @tf.function(autograph=False, jit_compile=False)
# def opt_var():
#     with tf.GradientTape() as tape:
#         loss = UCB_loss(champ_GP_reg)
#     grads = tape.gradient(loss, next_vars)
#     optimizer_fast.apply_gradients(zip(grads, next_vars))
#     return loss

# num_iters = 10000

# lls_ = np.zeros(num_iters, np.float64)
# tolerance = 1e-6  # Set your desired tolerance level
# previous_loss = float("inf")

# for i in range(num_iters):
#     loss = opt_var()
#     lls_[i] = loss

#     # Check if change in loss is less than tolerance
#     if abs(loss - previous_loss) < tolerance:
#         print(f"Acquisition function convergence reached at iteration {i+1}.")
#         lls_ = lls_[range(i + 1)]
#         break
```

```
#     previous_loss = loss

# print("Trained parameters:")
# for var in [next_alpha, next_beta, next_gamma_L, next_lambda, next_f, next_r]:
#     print("{} is {}".format(var.name, (var.bijector.forward(var).numpy().round(3))))
```

```
# plt.figure(figsize=(6, 3.5))
# plt.plot(lls_)
# plt.xlabel("Training iteration")
# plt.ylabel("Loss")
# plt.savefig("champagne_GP_images/bolfi_optim_loss_log_discrep.pdf", bbox_inches="tight")
# plt.show()
```

```
def update_GP_LOO(champ_GP, index_vals, obs_vals, observation_noise_variance_champ):

    def LOO_loss(champ_GP, index_vals, obs_vals, observation_noise_variance_champ):
        K = (
            champ_GP.kernel.matrix(index_vals, index_vals)
            + tf.eye(index_vals.shape[0], dtype=np.float64)
            * observation_noise_variance_champ
        )
        means = champ_GP.mean_fn(index_vals)
        K_inv = tf.linalg.inv(K)
        K_inv_y = K_inv @ tf.reshape(obs_vals - means, shape=[obs_vals.shape[0], 1])
        K_inv_diag = tf.linalg.diag_part(K_inv)
        log_var = tf.math.log(K_inv_diag)
        log_mu = tf.reshape(K_inv_y, shape=[-1]) ** 2
        return -tf.math.reduce_sum(log_var - log_mu)

    @tf.function(autograph=False, jit_compile=False)
    def opt_GP():
        with tf.GradientTape() as tape:
            loss = LOO_loss(
                champ_GP, index_vals, obs_vals, observation_noise_variance_champ
            )
        grads = tape.gradient(loss, champ_GP.trainable_variables)
        optimizer_slow.apply_gradients(zip(grads, champ_GP.trainable_variables))
        return loss

    num_iters = 10000
```

```
    lls_ = np.zeros(num_iters, np.float64)
    tolerance = 1e-6  # Set your desired tolerance level
    previous_loss = float("inf")

    for i in range(num_iters):
        loss = opt_GP()

        # Check if change in loss is less than tolerance
        if abs(loss - previous_loss) < tolerance:
            print(f"Hyperparameter convergence reached at iteration {i+1}.")
            break

        previous_loss = loss
    for var in optimizer_slow.variables:
        var.assign(tf.zeros_like(var))


def update_GP_MLE(champ_GP):
    @tf.function(autograph=False, jit_compile=False)
    def train_model():
        with tf.GradientTape() as tape:
            loss = -champ_GP.log_prob(obs_vals)
        grads = tape.gradient(loss, champ_GP.trainable_variables)
        optimizer_slow.apply_gradients(zip(grads, champ_GP.trainable_variables))
        return loss

    num_iters = 10000

    lls_ = np.zeros(num_iters, np.float64)
    tolerance = 1e-6  # Set your desired tolerance level
    previous_loss = float("inf")

    for i in range(num_iters):
        loss = train_model()

        # Check if change in loss is less than tolerance
        if abs(loss - previous_loss) < tolerance:
            print(f"Hyperparameter convergence reached at iteration {i+1}.")
            break

        previous_loss = loss
```

```
    for var in optimizer_slow.variables:
        var.assign(tf.zeros_like(var))


# def UCB_loss(eta_t, champ_GP_reg):
#     next_guess = tf.reshape(
#         tf.stack([next_alpha, next_beta, next_gamma_L, next_lambda, next_f, next_r]),
#         [1, 6],
#     )
#     mean_t = champ_GP_reg.mean_fn(next_guess)
#     std_t = champ_GP_reg.stddev(index_points=next_guess)
#     return tf.squeeze(mean_t - eta_t * std_t)


def update_var_UCB(eta_t, champ_GP_reg, next_vars):
    optimizer_fast = tf.keras.optimizers.Adam(learning_rate=0.1)

    @tf.function(autograph=False, jit_compile=False)
    def opt_var():
        with tf.GradientTape() as tape:
            loss = UCB_loss(eta_t, champ_GP_reg)
        grads = tape.gradient(loss, next_vars)
        optimizer_fast.apply_gradients(zip(grads, next_vars))
        return loss

    num_iters = 10000

    lls_ = np.zeros(num_iters, np.float64)
    tolerance = 1e-3  # Set your desired tolerance level
    previous_loss = float("inf")

    for i in range(num_iters):
        loss = opt_var()
        lls_[i] = loss

        # Check if change in loss is less than tolerance
        if abs(loss - previous_loss) < tolerance:
            print(f"Acquisition function convergence reached at iteration {i+1}.")
            break

        previous_loss = loss
```

```
    next_guess = tf.reshape(
        tf.stack([next_alpha, next_beta, next_gamma_L, next_lambda, next_f, next_r]),
        [1, 6],
    )
    print(
        "The final UCB loss was {}".format(loss.numpy().round(3))
        + " with predicted mean of {}".format(
            champ_GP_reg.mean_fn(next_guess).numpy().round(3)
        )
    )
    for var in optimizer_fast.variables:
        var.assign(tf.zeros_like(var))


def update_var_EI(GP_reg, alpha, beta, gamma_L, lambda_, f, r, min_obs):
    def EI_loss(alpha, beta, gamma_L, lambda_, f, r, min_obs):
        next_guess = tf.reshape(
            tf.stack([alpha, beta, gamma_L, lambda_, f, r]),
            [1, 6],
        )
        mean_t = GP_reg.mean_fn(next_guess)
        std_t = GP_reg.stddev(index_points=next_guess)
        delt = min_obs - mean_t
        return -tf.squeeze(
            delt * tfd.Normal(0, np.float64(1)).cdf(delt / std_t)
            + std_t * tfd.Normal(0, np.float64(1)).prob(delt / std_t)
        )

    optimizer_fast = tf.keras.optimizers.Adam(learning_rate=0.1)

    @tf.function(autograph=False, jit_compile=False)
    def opt_var():
        with tf.GradientTape() as tape:
            loss = EI_loss(alpha, beta, gamma_L, lambda_, f, r, min_obs)
        grads = tape.gradient(loss, next_vars)
        optimizer_fast.apply_gradients(zip(grads, next_vars))
        return loss

    num_iters = 10000

    lls_ = np.zeros(num_iters, np.float64)
```

```
    tolerance = 1e-7  # Set your desired tolerance level
    previous_loss = np.float64("inf")

    for i in range(num_iters):
        loss = opt_var()
        lls_[i] = loss

        # Check if change in loss is less than tolerance
        if abs(loss - previous_loss) < tolerance:
            print(f"Acquisition function convergence reached at iteration {i+1}.")
            lls_ = lls_[range(i + 1)]
            break

        previous_loss = loss

    next_guess = tf.reshape(
        tf.stack([next_alpha, next_beta, next_gamma_L, next_lambda, next_f, next_r]),
        [1, 6],
    )
    print(
        "The final EI loss was {}".format(loss.numpy().round(3))
        + " with predicted mean of {}".format(
            champ_GP_reg.mean_fn(next_guess).numpy().round(3)
        )
    )


def new_eta_t(t, d, exploration_rate):
    # return np.log((t + 1) ** (d * 2 + 2) * np.pi**2 / (3 * exploration_rate))
    return np.sqrt(np.log((t + 1) ** (d * 2 + 2) * np.pi**2 / (3 * exploration_rate)))
```

```
# optimizer_fast = tf.keras.optimizers.Adam(learning_rate=1.)
# update_var_EI()
# plt.figure(figsize=(6, 3.5))
# plt.plot(lls_)
# plt.xlabel("Training iteration")
# plt.ylabel("Loss")
# plt.show()
```

```
num_slice_updates = 15

all_slices = [
    np.linspace(0, 1, num_slice_updates + 2, dtype=np.float64)[1:-1],  # alpha
    np.linspace(0, 1, num_slice_updates + 2, dtype=np.float64)[1:-1],  # beta
    np.linspace(0, gamma_L_max, num_slice_updates + 2, dtype=np.float64)[
        1:-1
    ],  # gamma_L
    np.linspace(0, lambda_max, num_slice_updates + 2, dtype=np.float64)[1:-1],  # lambda
    np.linspace(0, f_max, num_slice_updates + 2, dtype=np.float64)[1:-1],  # f
    np.linspace(0, r_max, num_slice_updates + 2, dtype=np.float64)[1:-1],  # r
]
```

```
exploration_rate = 1
d = 6
update_GP_hp_freq = 20  # how many iterations before updating GP hyperparams
eta_t = tf.Variable(0, dtype=np.float64, name="eta_t")
min_obs = tf.Variable(100, dtype=np.float64, name="min_obs", shape=())
min_index = index_vals[
    champ_GP_reg.mean_fn(index_vals) == min(champ_GP_reg.mean_fn(index_vals))
][0]
simulation_reps = 20

for t in range(501):
    min_index = index_vals[
        champ_GP_reg.mean_fn(index_vals) == min(champ_GP_reg.mean_fn(index_vals))
    ][
        0,
    ]
    optimizer_slow = tf.keras.optimizers.Adam()
    # eta_t.assign(new_eta_t(t, d, exploration_rate))
    min_obs.assign(min(champ_GP_reg.mean_fn(index_vals)))
    print("Iteration " + str(t))
    # print(eta_t)

    ########################################################################

    # for var in [next_alpha, next_beta, next_gamma_L, next_lambda, next_f, next_r]:
    #     var.assign(
    #         var.bijector.forward(np.float64(100000000.0))
    #         * np.float64(np.random.uniform())
```

```
#     )

index_update = 0
for var in [next_alpha, next_beta, next_gamma_L, next_lambda, next_f, next_r]:
    if np.random.uniform() > 0.5:
        var.assign(min_index[index_update])
    else:
        var.assign(
            var.bijector.forward(np.float64(100000000.0))
            * np.float64(np.random.uniform())
        )
    index_update += 1

# update_var_UCB(eta_t, champ_GP_reg)
update_var_EI(
    champ_GP_reg,
    next_alpha,
    next_beta,
    next_gamma_L,
    next_lambda,
    next_f,
    next_r,
    min_obs,
)

bayes_params = np.array(
    [
        next_alpha.numpy(),
        next_beta.numpy(),
        next_gamma_L.numpy(),
        next_lambda.numpy(),
        next_f.numpy(),
        next_r.numpy(),
    ]
).reshape(1, -1)
print("The next parameters to simulate from are {}".format(bayes_params.round(3)))

if t < 6:
    new_params = np.repeat(
        np.array(
            [
```

```
                [
                    next_alpha.numpy(),
                    next_beta.numpy(),
                    next_gamma_L.numpy(),
                    next_lambda.numpy(),
                    next_f.numpy(),
                    next_r.numpy(),
                ]
            ]
        ),
        num_slice_updates,
        axis=0,
    )
    new_params[:, t % 6] = all_slices[t % 6]
else:
    new_params = np.repeat(
        np.array(
            [
                [
                    next_alpha.numpy(),
                    next_beta.numpy(),
                    next_gamma_L.numpy(),
                    next_lambda.numpy(),
                    next_f.numpy(),
                    next_r.numpy(),
                ]
            ]
        ),
        4,
        axis=0,
    )
    new_params[:, t % 6] = np.random.uniform(0, upper_bounds[t % 6], 4)

new_params = np.append(
    new_params,
    np.array(
        [
            [
                next_alpha.numpy(),
                next_beta.numpy(),
                next_gamma_L.numpy(),
```

```
                next_lambda.numpy(),
                next_f.numpy(),
                next_r.numpy(),
            ]
        ]
    ),
    axis=0,
)

new_params_reps = np.repeat(new_params, dis_mean_n, axis=0)
index_vals = np.append(index_vals, new_params, axis=0)

with mp.Pool(processes=mp.cpu_count()) as pool:
    args = [
        (a, b, c, d, e, f, int(g * np.random.uniform()))
        for (a, b, c, d, e, f), g in zip(
            list(map(tuple, new_params_reps)), range(new_params_reps.shape[0])
        )
    ]
    results = pool.starmap(single_discrepency, args)

new_discrepencies = np.mean(np.array(results).reshape(-1, dis_mean_n), axis=1)

print("The mean of the samples was {}".format(new_discrepencies[-1].round(3)))
obs_vals = np.append(obs_vals, new_discrepencies)

#########################################################################

champ_GP_reg = tfd.GaussianProcessRegressionModel(
    kernel=kernel_champ,
    observation_index_points=index_vals,
    observations=obs_vals,
    observation_noise_variance=observation_noise_variance_champ,
    predictive_noise_variance=0.0,
    mean_fn=const_mean_fn(),
)

if t % update_GP_hp_freq == 0:
    champ_GP = tfd.GaussianProcess(
        kernel=kernel_champ,
        observation_noise_variance=observation_noise_variance_champ,
```

```
            index_points=index_vals,
            mean_fn=const_mean_fn(),
        )
        update_GP_LOO(champ_GP, index_vals, obs_vals, observation_noise_variance_champ)
        # update_GP_MLE(champ_GP)
        min_value = min(champ_GP_reg.mean_fn(index_vals))
        min_index = index_vals[champ_GP_reg.mean_fn(index_vals) == min_value][0,]
        print(
            "The minimum predicted mean of the observed indices is {}".format(
                min_value.numpy().round(3)
            )
            + " at the point \n{}".format(min_index.round(3))
        )

    if (t > 0) & (t % 50 == 0):
        print("Trained parameters:")
        for train_var in champ_GP.trainable_variables:
            if "bias" in train_var.name:
                print("{} is {}\n".format(train_var.name, train_var.numpy().round(3)))
            else:
                # if "length" in train_var.name:
                #     print(
                #         "{} is {}\n".format(
                #             train_var.name,
                #             tfb.Sigmoid(
                #                 np.float64(0.0),
                #                 [
                #                     1.0 / 2,
                #                     1.0 / 2,
                #                     gamma_L_max / 2,
                #                     lambda_max / 2,
                #                     f_max / 2,
                #                     r_max / 2,
                #                 ],
                #             )
                #             .forward(train_var)
                #             .numpy()
                #             .round(3),
                #         )
                #     )
                # else:
```

```
            print(
                "{} is {}\n".format(
                    train_var.name,
                    constrain_positive.forward(train_var).numpy().round(3),
                )
            )

    for var in vars:
        champ_GP_reg_plot = tfd.GaussianProcessRegressionModel(
            kernel=kernel_champ,
            index_points=slice_indices_dfs_dict[var + "_gp_indices_df"].values,
            observation_index_points=index_vals,
            observations=obs_vals,
            observation_noise_variance=observation_noise_variance_champ,
            predictive_noise_variance=0.0,
            mean_fn=const_mean_fn(),
        )
        GP_samples = champ_GP_reg_plot.sample(gp_samp_no, seed=GP_seed)
        gp_samples_dict[var + "_gp_samps" + str(t) + "iters"] = GP_samples

        plt.figure(figsize=(4, 2.5))
        plt.scatter(
            slice_indices_dfs_dict[var + "_slice_indices_df"][var].values,
            slice_discrepencies_dict[var + "_slice_discrepencies"],
            label="Untrained Discrepencies",
            alpha=0.6,
        )
        for i in range(gp_samp_no):
            plt.plot(
                slice_indices_dfs_dict[var + "_gp_indices_df"][var].values,
                GP_samples[i, :],
                c="r",
                alpha=0.1,
                label="Posterior Sample" if i == 0 else None,
            )
        plt.plot(
            slice_indices_dfs_dict[var + "_gp_indices_df"][var].values,
            champ_GP_reg_plot.mean_fn(
                slice_indices_dfs_dict[var + "_gp_indices_df"].values
            ),
            c="black",
```

```
            alpha=1,
            label="Posterior Mean",
        )
        # leg = plt.legend(loc="upper left")
        # for lh in leg.legend_handles:
        #     lh.set_alpha(1)
        if var in ["f", "r"]:
            plt.xlabel("$" + var + "$")
            # plt.title(
            #     "$" + var + "$ slice after " + str(t) + " Bayesian acquisitions"
            # )
        else:
            plt.xlabel("$\\" + var + "$")
            # plt.title(
            #     "$\\" + var + "$ slice after " + str(t) + " Bayesian acquisitions"
            # )
        plt.ylabel("log(Discrepancy)")
        plt.ylim((-3, 2))
        plt.savefig(
            "champagne_GP_images/"
            + var
            + "_slice_"
            + str(t)
            + "_bolfi_updates_log_discrep.pdf",
            bbox_inches="tight",
        )
        plt.show()
```

```
Iteration 0
Acquisition function convergence reached at iteration 2.
The final EI loss was -0.0 with predicted mean of [1.104]
The next parameters to simulate from are [[0.169 0.953 0.018 0.035 0.045 0.02 ]]
The mean of the samples was 1.047
Hyperparameter convergence reached at iteration 5178.
The minimum predicted mean of the observed indices is -0.947 at the point
[0.169 0.759 0.018 0.012 0.026 0.04 ]
Iteration 1
Acquisition function convergence reached at iteration 1032.
The final EI loss was -0.391 with predicted mean of [-0.99]
The next parameters to simulate from are [[0.    0.    0.033 0.012 0.    0.063]]
The mean of the samples was -0.097
Iteration 2
```

```
Acquisition function convergence reached at iteration 610.
The final EI loss was -0.29 with predicted mean of [-1.11]
The next parameters to simulate from are [[0.289 0.216 0.033 0.013 0.024 0.017]]
The mean of the samples was -0.417
Iteration 3
Acquisition function convergence reached at iteration 574.
The final EI loss was -0.232 with predicted mean of [-0.87]
The next parameters to simulate from are [[0.    0.001 0.003 0.012 0.043 0.046]]
The mean of the samples was 0.445
Iteration 4
Acquisition function convergence reached at iteration 2.
The final EI loss was -0.0 with predicted mean of [1.176]
The next parameters to simulate from are [[0.289 0.216 0.009 0.039 0.024 0.017]]
The mean of the samples was 1.214
Iteration 5
Acquisition function convergence reached at iteration 1193.
The final EI loss was -0.227 with predicted mean of [-0.863]
The next parameters to simulate from are [[0.086 1.    0.    0.01  0.    0.016]]
The mean of the samples was 0.366
Iteration 6
Acquisition function convergence reached at iteration 1162.
The final EI loss was -0.121 with predicted mean of [-0.662]
The next parameters to simulate from are [[1.    0.    0.033 0.014 0.029 0.   ]]
The mean of the samples was 0.357
Iteration 7
Acquisition function convergence reached at iteration 229.
The final EI loss was -0.149 with predicted mean of [-0.911]
The next parameters to simulate from are [[0.371 0.987 0.033 0.014 0.023 0.046]]
The mean of the samples was -0.673
Iteration 8
Acquisition function convergence reached at iteration 2.
The final EI loss was -0.0 with predicted mean of [0.661]
The next parameters to simulate from are [[0.289 0.761 0.018 0.038 0.019 0.057]]
The mean of the samples was 0.629
Iteration 9
Acquisition function convergence reached at iteration 534.
The final EI loss was -0.022 with predicted mean of [1.047]
The next parameters to simulate from are [[0.    1.    0.033 0.05  0.    0.   ]]
The mean of the samples was 2.097
Iteration 10
Acquisition function convergence reached at iteration 2.
The final EI loss was -0.0 with predicted mean of [1.306]
The next parameters to simulate from are [[0.12  0.749 0.021 0.04  0.024 0.018]]
```

```
The mean of the samples was 1.273
Iteration 11
Acquisition function convergence reached at iteration 95.
The final EI loss was -0.107 with predicted mean of [-0.974]
The next parameters to simulate from are [[0.303 0.112 0.017 0.013 0.02  0.032]]
The mean of the samples was -1.054
Iteration 12
Acquisition function convergence reached at iteration 92.
The final EI loss was -0.036 with predicted mean of [-1.023]
The next parameters to simulate from are [[0.307 0.392 0.017 0.014 0.024 0.031]]
The mean of the samples was -1.201
Iteration 13
Acquisition function convergence reached at iteration 73.
The final EI loss was -0.04 with predicted mean of [-1.108]
The next parameters to simulate from are [[0.217 0.654 0.019 0.014 0.023 0.032]]
The mean of the samples was -1.109
Iteration 14
Acquisition function convergence reached at iteration 91.
The final EI loss was -0.023 with predicted mean of [-1.102]
The next parameters to simulate from are [[0.229 0.306 0.018 0.014 0.023 0.033]]
The mean of the samples was -1.241
Iteration 15
Acquisition function convergence reached at iteration 64.
The final EI loss was -0.028 with predicted mean of [-1.106]
The next parameters to simulate from are [[0.188 0.349 0.024 0.014 0.022 0.035]]
The mean of the samples was -1.133
Iteration 16
Acquisition function convergence reached at iteration 2.
The final EI loss was -0.0 with predicted mean of [0.248]
The next parameters to simulate from are [[0.971 0.413 0.023 0.014 0.023 0.033]]
The mean of the samples was 0.334
Iteration 17
Acquisition function convergence reached at iteration 73.
The final EI loss was -0.026 with predicted mean of [-1.113]
The next parameters to simulate from are [[0.186 0.405 0.024 0.014 0.023 0.035]]
The mean of the samples was -1.049
Iteration 18
Acquisition function convergence reached at iteration 88.
The final EI loss was -0.033 with predicted mean of [-1.099]
The next parameters to simulate from are [[0.26  0.168 0.022 0.013 0.028 0.035]]
The mean of the samples was -1.17
Iteration 19
Acquisition function convergence reached at iteration 2.
```

```
The final EI loss was -0.0 with predicted mean of [0.795]
The next parameters to simulate from are [[0.26  0.94  0.011 0.035 0.028 0.035]]
The mean of the samples was 0.821
Iteration 20
Acquisition function convergence reached at iteration 59.
The final EI loss was -0.028 with predicted mean of [-1.128]
The next parameters to simulate from are [[0.252 0.113 0.023 0.013 0.029 0.036]]
The mean of the samples was -1.002
The minimum predicted mean of the observed indices is -1.226 at the point
[0.303 0.112 0.017 0.013 0.02  0.03 ]
Iteration 21
Acquisition function convergence reached at iteration 506.
The final EI loss was -0.019 with predicted mean of [0.455]
The next parameters to simulate from are [[0.   0.   0.   0.05 0.   0.  ]]
The mean of the samples was 2.103
Iteration 22
Acquisition function convergence reached at iteration 838.
The final EI loss was -0.144 with predicted mean of [-1.283]
The next parameters to simulate from are [[0.239 0.001 0.013 0.013 0.025 0.025]]
The mean of the samples was -0.937
Iteration 23
Acquisition function convergence reached at iteration 891.
The final EI loss was -0.223 with predicted mean of [-1.416]
The next parameters to simulate from are [[0.26  0.283 0.015 0.013 0.    0.026]]
The mean of the samples was -0.375
Iteration 24
Acquisition function convergence reached at iteration 126.
The final EI loss was -0.08 with predicted mean of [-0.972]
The next parameters to simulate from are [[0.464 0.    0.01  0.015 0.012 0.017]]
The mean of the samples was -1.189
Iteration 25
Acquisition function convergence reached at iteration 112.
The final EI loss was -0.062 with predicted mean of [-1.274]
The next parameters to simulate from are [[0.351 0.    0.009 0.014 0.002 0.017]]
The mean of the samples was -0.688
Iteration 26
Acquisition function convergence reached at iteration 548.
The final EI loss was -0.184 with predicted mean of [-1.324]
The next parameters to simulate from are [[0.416 0.001 0.012 0.016 0.023 0.019]]
The mean of the samples was -0.656
Iteration 27
Acquisition function convergence reached at iteration 2.
The final EI loss was -0.0 with predicted mean of [-0.07]
```

```
The next parameters to simulate from are [[0.229 0.134 0.003 0.014 0.032 0.033]]
The mean of the samples was 0.203
Iteration 28
Acquisition function convergence reached at iteration 795.
The final EI loss was -0.064 with predicted mean of [-0.157]
The next parameters to simulate from are [[0.719 0.001 0.033 0.015 0.05  0.067]]
The mean of the samples was 0.146
Iteration 29
Acquisition function convergence reached at iteration 260.
The final EI loss was -0.122 with predicted mean of [-1.231]
The next parameters to simulate from are [[0.228 0.519 0.02  0.013 0.04  0.033]]
The mean of the samples was -0.89
Iteration 30
Acquisition function convergence reached at iteration 564.
The final EI loss was -0.09 with predicted mean of [-0.789]
The next parameters to simulate from are [[0.    0.    0.009 0.    0.05  0.044]]
The mean of the samples was 0.378
Iteration 31
Acquisition function convergence reached at iteration 582.
The final EI loss was -0.076 with predicted mean of [-1.125]
The next parameters to simulate from are [[0.441 0.    0.011 0.012 0.021 0.019]]
The mean of the samples was -1.768
Iteration 32
Acquisition function convergence reached at iteration 112.
The final EI loss was -0.088 with predicted mean of [-1.527]
The next parameters to simulate from are [[0.476 0.    0.011 0.011 0.022 0.019]]
The mean of the samples was -1.621
Iteration 33
Acquisition function convergence reached at iteration 371.
The final EI loss was -0.008 with predicted mean of [0.225]
The next parameters to simulate from are [[1.    0.001 0.033 0.028 0.    0.   ]]
The mean of the samples was 0.53
Iteration 34
Acquisition function convergence reached at iteration 474.
The final EI loss was -0.019 with predicted mean of [-0.713]
The next parameters to simulate from are [[0.388 0.    0.033 0.014 0.    0.033]]
The mean of the samples was -0.504
Iteration 35
Acquisition function convergence reached at iteration 424.
The final EI loss was -0.013 with predicted mean of [0.27]
The next parameters to simulate from are [[1.    1.    0.    0.05  0.05  0.067]]
The mean of the samples was 0.872
Iteration 36
```

```
Acquisition function convergence reached at iteration 72.
The final EI loss was -0.061 with predicted mean of [-1.572]
The next parameters to simulate from are [[0.473 0.    0.012 0.011 0.021 0.019]]
The mean of the samples was -1.744
Iteration 37
Acquisition function convergence reached at iteration 479.
The final EI loss was -0.012 with predicted mean of [0.045]
The next parameters to simulate from are [[0.001 0.    0.033 0.022 0.05  0.067]]
The mean of the samples was 0.41
Iteration 38
Acquisition function convergence reached at iteration 422.
The final EI loss was -0.012 with predicted mean of [0.173]
The next parameters to simulate from are [[0.994 1.    0.033 0.    0.05  0.027]]
The mean of the samples was 0.445
Iteration 39
Acquisition function convergence reached at iteration 2.
The final EI loss was -0.0 with predicted mean of [1.404]
The next parameters to simulate from are [[0.478 0.    0.011 0.045 0.034 0.01 ]]
The mean of the samples was 1.485
Iteration 40
Acquisition function convergence reached at iteration 430.
The final EI loss was -0.015 with predicted mean of [0.114]
The next parameters to simulate from are [[1.    0.001 0.033 0.03  0.    0.067]]
The mean of the samples was 0.436
Hyperparameter convergence reached at iteration 5370.
The minimum predicted mean of the observed indices is -1.64 at the point
[0.473 0.    0.012 0.011 0.021 0.019]
Iteration 41
Acquisition function convergence reached at iteration 296.
The final EI loss was -0.005 with predicted mean of [0.028]
The next parameters to simulate from are [[0.995 0.001 0.    0.01  0.05  0.   ]]
The mean of the samples was -0.247
Iteration 42
Acquisition function convergence reached at iteration 112.
The final EI loss was -0.099 with predicted mean of [-1.671]
The next parameters to simulate from are [[0.436 0.    0.011 0.012 0.017 0.02 ]]
The mean of the samples was -1.647
Iteration 43
Acquisition function convergence reached at iteration 278.
The final EI loss was -0.004 with predicted mean of [0.09]
The next parameters to simulate from are [[1.    0.997 0.    0.05  0.    0.066]]
The mean of the samples was 0.555
Iteration 44
```

```
Acquisition function convergence reached at iteration 88.
The final EI loss was -0.053 with predicted mean of [-1.61]
The next parameters to simulate from are [[0.441 0.    0.012 0.011 0.021 0.021]]
The mean of the samples was -1.632
Iteration 45
Acquisition function convergence reached at iteration 179.
The final EI loss was -0.05 with predicted mean of [-1.609]
The next parameters to simulate from are [[0.445 0.    0.011 0.011 0.019 0.017]]
The mean of the samples was -1.696
Iteration 46
Acquisition function convergence reached at iteration 93.
The final EI loss was -0.049 with predicted mean of [-1.638]
The next parameters to simulate from are [[0.445 0.    0.011 0.011 0.019 0.017]]
The mean of the samples was -1.66
Iteration 47
Acquisition function convergence reached at iteration 162.
The final EI loss was -0.042 with predicted mean of [-1.655]
The next parameters to simulate from are [[0.445 0.    0.011 0.012 0.02  0.018]]
The mean of the samples was -1.732
Iteration 48
Acquisition function convergence reached at iteration 2.
The final EI loss was -0.0 with predicted mean of [1.255]
The next parameters to simulate from are [[0.418 0.    0.011 0.048 0.034 0.018]]
The mean of the samples was 1.589
Iteration 49
Acquisition function convergence reached at iteration 137.
The final EI loss was -0.013 with predicted mean of [-0.806]
The next parameters to simulate from are [[0.411 0.879 0.03  0.012 0.032 0.031]]
The mean of the samples was -0.901
Iteration 50
Acquisition function convergence reached at iteration 75.
The final EI loss was -0.004 with predicted mean of [0.077]
The next parameters to simulate from are [[0.981 0.001 0.    0.03  0.    0.067]]
The mean of the samples was 0.508
Trained parameters:
amplitude_champ:0 is 0.796

length_scales_champ:0 is [0.379 1.216 0.01  0.006 0.017 0.016]

observation_noise_variance_champ:0 is 0.053

bias_mean:0 is 0.663
```

```
Iteration 51
Acquisition function convergence reached at iteration 2.
The final EI loss was -0.0 with predicted mean of [0.867]
The next parameters to simulate from are [[0.56  0.    0.011 0.035 0.02  0.018]]
The mean of the samples was 0.93
Iteration 52
Acquisition function convergence reached at iteration 95.
The final EI loss was -0.01 with predicted mean of [-0.887]
The next parameters to simulate from are [[0.443 0.683 0.028 0.013 0.016 0.031]]
The mean of the samples was -0.727
Iteration 53
Acquisition function convergence reached at iteration 2.
The final EI loss was -0.0 with predicted mean of [1.026]
The next parameters to simulate from are [[0.468 0.    0.011 0.037 0.02  0.018]]
The mean of the samples was 1.002
Iteration 54
Acquisition function convergence reached at iteration 173.
The final EI loss was -0.011 with predicted mean of [-0.686]
The next parameters to simulate from are [[0.056 0.    0.033 0.013 0.018 0.037]]
The mean of the samples was -0.906
Iteration 55
Acquisition function convergence reached at iteration 256.
The final EI loss was -0.01 with predicted mean of [-0.442]
The next parameters to simulate from are [[0.001 0.    0.    0.    0.03  0.05 ]]
The mean of the samples was 0.393
Iteration 56
Acquisition function convergence reached at iteration 2.
The final EI loss was -0.0 with predicted mean of [0.804]
The next parameters to simulate from are [[0.445 0.142 0.013 0.037 0.02  0.052]]
The mean of the samples was 0.731
Iteration 57
Acquisition function convergence reached at iteration 111.
The final EI loss was -0.011 with predicted mean of [-0.602]
The next parameters to simulate from are [[0.055 0.624 0.033 0.012 0.036 0.041]]
The mean of the samples was -0.802
Iteration 58
Acquisition function convergence reached at iteration 116.
The final EI loss was -0.037 with predicted mean of [-1.663]
The next parameters to simulate from are [[0.443 0.    0.012 0.012 0.02  0.019]]
The mean of the samples was -1.707
Iteration 59
Acquisition function convergence reached at iteration 197.
The final EI loss was -0.004 with predicted mean of [0.021]
```

```
The next parameters to simulate from are [[0.999 0.003 0.016 0.016 0.05  0.   ]]
The mean of the samples was 0.381
Iteration 60
Acquisition function convergence reached at iteration 324.
The final EI loss was -0.004 with predicted mean of [0.059]
The next parameters to simulate from are [[1.    0.    0.    0.    0.    0.067]]
The mean of the samples was 0.655
Hyperparameter convergence reached at iteration 1701.
The minimum predicted mean of the observed indices is -1.683 at the point
[0.443 0.    0.012 0.012 0.02  0.019]
Iteration 61
Acquisition function convergence reached at iteration 105.
The final EI loss was -0.009 with predicted mean of [-0.507]
The next parameters to simulate from are [[0.01  0.    0.029 0.011 0.028 0.053]]
The mean of the samples was -0.962
Iteration 62
Acquisition function convergence reached at iteration 110.
The final EI loss was -0.034 with predicted mean of [-1.671]
The next parameters to simulate from are [[0.44  0.    0.012 0.012 0.02  0.019]]
The mean of the samples was -1.646
Iteration 63
Acquisition function convergence reached at iteration 124.
The final EI loss was -0.009 with predicted mean of [-0.639]
The next parameters to simulate from are [[0.236 0.    0.033 0.018 0.014 0.036]]
The mean of the samples was -0.862
Iteration 64
Acquisition function convergence reached at iteration 81.
The final EI loss was -0.009 with predicted mean of [-0.569]
The next parameters to simulate from are [[0.002 0.988 0.027 0.015 0.016 0.048]]
The mean of the samples was -1.047
Iteration 65
Acquisition function convergence reached at iteration 157.
The final EI loss was -0.029 with predicted mean of [-1.687]
The next parameters to simulate from are [[0.445 0.    0.012 0.012 0.021 0.019]]
The mean of the samples was -1.891
Iteration 66
Acquisition function convergence reached at iteration 146.
The final EI loss was -0.01 with predicted mean of [-0.737]
The next parameters to simulate from are [[0.007 0.005 0.033 0.016 0.017 0.052]]
The mean of the samples was -1.034
Iteration 67
Acquisition function convergence reached at iteration 229.
The final EI loss was -0.003 with predicted mean of [0.127]
```

```
The next parameters to simulate from are [[0.998 0.002 0.033 0.05  0.05  0.   ]]
The mean of the samples was 1.541
Iteration 68
Acquisition function convergence reached at iteration 101.
The final EI loss was -0.009 with predicted mean of [-0.613]
The next parameters to simulate from are [[0.011 0.574 0.028 0.015 0.021 0.063]]
The mean of the samples was -0.857
Iteration 69
Acquisition function convergence reached at iteration 2.
The final EI loss was -0.0 with predicted mean of [0.849]
The next parameters to simulate from are [[0.445 0.497 0.012 0.037 0.021 0.034]]
The mean of the samples was 0.712
Iteration 70
Acquisition function convergence reached at iteration 281.
The final EI loss was -0.004 with predicted mean of [0.026]
The next parameters to simulate from are [[0.006 0.994 0.033 0.    0.    0.066]]
The mean of the samples was 0.508
Iteration 71
Acquisition function convergence reached at iteration 19.
The final EI loss was -0.005 with predicted mean of [-0.297]
The next parameters to simulate from are [[0.042 0.    0.007 0.006 0.04  0.02 ]]
The mean of the samples was -0.778
Iteration 72
Acquisition function convergence reached at iteration 2.
The final EI loss was -0.0 with predicted mean of [1.249]
The next parameters to simulate from are [[0.625 0.    0.012 0.042 0.039 0.019]]
The mean of the samples was 1.411
Iteration 73
Acquisition function convergence reached at iteration 54.
The final EI loss was -0.027 with predicted mean of [-1.706]
The next parameters to simulate from are [[0.446 0.    0.012 0.012 0.021 0.019]]
The mean of the samples was -1.727
Iteration 74
Acquisition function convergence reached at iteration 110.
The final EI loss was -0.017 with predicted mean of [-0.92]
The next parameters to simulate from are [[0.06  0.    0.015 0.007 0.038 0.03 ]]
The mean of the samples was -0.882
Iteration 75
Acquisition function convergence reached at iteration 74.
The final EI loss was -0.011 with predicted mean of [-0.758]
The next parameters to simulate from are [[0.007 0.    0.023 0.016 0.011 0.032]]
The mean of the samples was -1.094
Iteration 76
```

```
Acquisition function convergence reached at iteration 74.
The final EI loss was -0.005 with predicted mean of [-0.715]
The next parameters to simulate from are [[0.475 0.009 0.033 0.017 0.034 0.029]]
The mean of the samples was -0.488
Iteration 77
Acquisition function convergence reached at iteration 276.
The final EI loss was -0.012 with predicted mean of [-0.984]
The next parameters to simulate from are [[0.001 0.    0.024 0.014 0.016 0.047]]
The mean of the samples was -0.928
Iteration 78
Acquisition function convergence reached at iteration 26.
The final EI loss was -0.007 with predicted mean of [-0.902]
The next parameters to simulate from are [[0.061 0.    0.007 0.008 0.032 0.026]]
The mean of the samples was -1.323
Iteration 79
Acquisition function convergence reached at iteration 99.
The final EI loss was -0.008 with predicted mean of [-0.609]
The next parameters to simulate from are [[0.012 0.    0.025 0.019 0.011 0.05 ]]
The mean of the samples was -0.918
Iteration 80
Acquisition function convergence reached at iteration 180.
The final EI loss was -0.002 with predicted mean of [0.221]
The next parameters to simulate from are [[0.    0.998 0.    0.05  0.    0.045]]
The mean of the samples was 1.079
Hyperparameter convergence reached at iteration 1911.
The minimum predicted mean of the observed indices is -1.711 at the point
[0.445 0.    0.012 0.012 0.021 0.019]
Iteration 81
Acquisition function convergence reached at iteration 125.
The final EI loss was -0.025 with predicted mean of [-1.691]
The next parameters to simulate from are [[0.445 0.    0.011 0.011 0.022 0.019]]
The mean of the samples was -1.735
Iteration 82
Acquisition function convergence reached at iteration 73.
The final EI loss was -0.011 with predicted mean of [-0.803]
The next parameters to simulate from are [[0.001 0.005 0.014 0.017 0.013 0.034]]
The mean of the samples was -1.016
Iteration 83
Acquisition function convergence reached at iteration 2.
The final EI loss was -0.0 with predicted mean of [0.451]
The next parameters to simulate from are [[0.444 0.916 0.012 0.029 0.017 0.018]]
The mean of the samples was 0.297
Iteration 84
```

```
Acquisition function convergence reached at iteration 67.
The final EI loss was -0.005 with predicted mean of [-0.412]
The next parameters to simulate from are [[0.746 0.    0.033 0.02  0.014 0.024]]
The mean of the samples was -0.374
Iteration 85
Acquisition function convergence reached at iteration 104.
The final EI loss was -0.01 with predicted mean of [-0.808]
The next parameters to simulate from are [[0.138 0.867 0.016 0.017 0.01  0.025]]
The mean of the samples was -1.055
Iteration 86
Acquisition function convergence reached at iteration 131.
The final EI loss was -0.012 with predicted mean of [-0.934]
The next parameters to simulate from are [[0.002 0.999 0.019 0.016 0.012 0.038]]
The mean of the samples was -1.159
Iteration 87
Acquisition function convergence reached at iteration 100.
The final EI loss was -0.006 with predicted mean of [-0.284]
The next parameters to simulate from are [[0.516 0.993 0.011 0.018 0.    0.001]]
The mean of the samples was 0.302
Iteration 88
Acquisition function convergence reached at iteration 95.
The final EI loss was -0.025 with predicted mean of [-1.711]
The next parameters to simulate from are [[0.446 0.    0.011 0.011 0.022 0.019]]
The mean of the samples was -1.82
Iteration 89
Acquisition function convergence reached at iteration 142.
The final EI loss was -0.007 with predicted mean of [-0.337]
The next parameters to simulate from are [[0.372 0.995 0.033 0.02  0.    0.043]]
The mean of the samples was -0.679
Iteration 90
Acquisition function convergence reached at iteration 79.
The final EI loss was -0.002 with predicted mean of [0.019]
The next parameters to simulate from are [[0.004 0.992 0.019 0.    0.042 0.066]]
The mean of the samples was 0.139
Iteration 91
Acquisition function convergence reached at iteration 59.
The final EI loss was -0.002 with predicted mean of [0.079]
The next parameters to simulate from are [[0.996 0.004 0.023 0.022 0.022 0.008]]
The mean of the samples was 0.488
Iteration 92
Acquisition function convergence reached at iteration 2.
The final EI loss was -0.0 with predicted mean of [1.013]
The next parameters to simulate from are [[0.609 0.    0.011 0.038 0.022 0.015]]
```

```
The mean of the samples was 1.037
Iteration 93
Acquisition function convergence reached at iteration 81.
The final EI loss was -0.007 with predicted mean of [-0.384]
The next parameters to simulate from are [[0.471 0.    0.032 0.009 0.049 0.02 ]]
The mean of the samples was -0.673
Iteration 94
Acquisition function convergence reached at iteration 103.
The final EI loss was -0.008 with predicted mean of [-0.429]
The next parameters to simulate from are [[0.003 0.    0.033 0.011 0.034 0.067]]
The mean of the samples was -0.868
Iteration 95
Acquisition function convergence reached at iteration 56.
The final EI loss was -0.012 with predicted mean of [-0.788]
The next parameters to simulate from are [[0.21  0.984 0.023 0.019 0.004 0.034]]
The mean of the samples was -0.862
Iteration 96
Acquisition function convergence reached at iteration 2.
The final EI loss was -0.0 with predicted mean of [1.175]
The next parameters to simulate from are [[0.446 0.6   0.014 0.041 0.022 0.022]]
The mean of the samples was 1.041
Iteration 97
Acquisition function convergence reached at iteration 2.
The final EI loss was -0.0 with predicted mean of [1.368]
The next parameters to simulate from are [[0.341 0.    0.011 0.044 0.022 0.019]]
The mean of the samples was 1.391
Iteration 98
Acquisition function convergence reached at iteration 137.
The final EI loss was -0.009 with predicted mean of [-0.853]
The next parameters to simulate from are [[0.471 1.    0.033 0.018 0.014 0.033]]
The mean of the samples was -0.855
Iteration 99
Acquisition function convergence reached at iteration 91.
The final EI loss was -0.017 with predicted mean of [-0.895]
The next parameters to simulate from are [[0.06  1.    0.007 0.007 0.036 0.027]]
The mean of the samples was -1.446
Iteration 100
Acquisition function convergence reached at iteration 164.
The final EI loss was -0.003 with predicted mean of [0.114]
The next parameters to simulate from are [[0.999 0.    0.    0.017 0.    0.067]]
The mean of the samples was 0.539
Hyperparameter convergence reached at iteration 1917.
The minimum predicted mean of the observed indices is -1.719 at the point
```

```
[0.446 0.    0.011 0.011 0.022 0.019]
Trained parameters:
amplitude_champ:0 is 0.823

length_scales_champ:0 is [0.413 1.532 0.011 0.006 0.014 0.015]

observation_noise_variance_champ:0 is 0.052

bias_mean:0 is 0.754

Iteration 101
Acquisition function convergence reached at iteration 156.
The final EI loss was -0.001 with predicted mean of [0.365]
The next parameters to simulate from are [[0.998 0.    0.033 0.036 0.05  0.04 ]]
The mean of the samples was 1.028
Iteration 102
Acquisition function convergence reached at iteration 2.
The final EI loss was -0.0 with predicted mean of [0.67]
The next parameters to simulate from are [[0.446 0.66  0.011 0.031 0.024 0.019]]
The mean of the samples was 0.643
Iteration 103
Acquisition function convergence reached at iteration 105.
The final EI loss was -0.026 with predicted mean of [-1.708]
The next parameters to simulate from are [[0.446 0.    0.012 0.011 0.022 0.019]]
The mean of the samples was -1.749
Iteration 104
Acquisition function convergence reached at iteration 157.
The final EI loss was -0.005 with predicted mean of [-0.512]
The next parameters to simulate from are [[0.621 0.    0.033 0.021 0.    0.035]]
The mean of the samples was -0.391
Iteration 105
Acquisition function convergence reached at iteration 2.
The final EI loss was -0.0 with predicted mean of [0.562]
The next parameters to simulate from are [[0.438 0.749 0.012 0.034 0.008 0.019]]
The mean of the samples was 0.597
Iteration 106
Acquisition function convergence reached at iteration 109.
The final EI loss was -0.01 with predicted mean of [-0.925]
The next parameters to simulate from are [[0.002 0.    0.    0.006 0.036 0.026]]
The mean of the samples was 0.071
Iteration 107
Acquisition function convergence reached at iteration 44.
The final EI loss was -0.006 with predicted mean of [-0.99]
```

```
The next parameters to simulate from are [[0.226 0.    0.024 0.017 0.012 0.039]]
The mean of the samples was -1.023
Iteration 108
Acquisition function convergence reached at iteration 102.
The final EI loss was -0.024 with predicted mean of [-1.719]
The next parameters to simulate from are [[0.445 0.    0.011 0.012 0.022 0.019]]
The mean of the samples was -1.829
Iteration 109
Acquisition function convergence reached at iteration 11.
The final EI loss was -0.016 with predicted mean of [-1.457]
The next parameters to simulate from are [[0.375 0.    0.011 0.009 0.026 0.021]]
The mean of the samples was -1.531
Iteration 110
Acquisition function convergence reached at iteration 222.
The final EI loss was -0.003 with predicted mean of [0.161]
The next parameters to simulate from are [[0.999 0.    0.    0.028 0.05  0.   ]]
The mean of the samples was 1.559
Iteration 111
Acquisition function convergence reached at iteration 291.
The final EI loss was -0.009 with predicted mean of [-0.947]
The next parameters to simulate from are [[0.001 1.    0.013 0.012 0.016 0.03 ]]
The mean of the samples was -1.401
Iteration 112
Acquisition function convergence reached at iteration 2.
The final EI loss was -0.0 with predicted mean of [-0.574]
The next parameters to simulate from are [[0.446 0.    0.011 0.011 0.022 0.039]]
The mean of the samples was -0.654
Iteration 113
Acquisition function convergence reached at iteration 274.
The final EI loss was -0.004 with predicted mean of [0.084]
The next parameters to simulate from are [[0.999 0.    0.033 0.    0.    0.066]]
The mean of the samples was 0.6
Iteration 114
Acquisition function convergence reached at iteration 115.
The final EI loss was -0.023 with predicted mean of [-1.73]
The next parameters to simulate from are [[0.449 0.    0.011 0.012 0.022 0.019]]
The mean of the samples was -1.715
Iteration 115
Acquisition function convergence reached at iteration 2.
The final EI loss was -0.0 with predicted mean of [0.017]
The next parameters to simulate from are [[0.446 0.    0.012 0.021 0.022 0.019]]
The mean of the samples was 0.035
Iteration 116
```

```
Acquisition function convergence reached at iteration 234.
The final EI loss was -0.003 with predicted mean of [0.128]
The next parameters to simulate from are [[1. 0. 0. 0. 0. 0.]]
The mean of the samples was 0.552
Iteration 117
Acquisition function convergence reached at iteration 110.
The final EI loss was -0.003 with predicted mean of [0.065]
The next parameters to simulate from are [[0.002 0.    0.033 0.006 0.05  0.   ]]
The mean of the samples was 0.407
Iteration 118
Acquisition function convergence reached at iteration 91.
The final EI loss was -0.013 with predicted mean of [-1.084]
The next parameters to simulate from are [[0.013 0.    0.01  0.005 0.03  0.025]]
The mean of the samples was -0.524
Iteration 119
Acquisition function convergence reached at iteration 56.
The final EI loss was -0.021 with predicted mean of [-1.367]
The next parameters to simulate from are [[0.196 0.592 0.01  0.008 0.038 0.026]]
The mean of the samples was -1.431
Iteration 120
Acquisition function convergence reached at iteration 132.
The final EI loss was -0.001 with predicted mean of [0.408]
The next parameters to simulate from are [[0.998 0.    0.033 0.031 0.025 0.   ]]
The mean of the samples was 0.737
Hyperparameter convergence reached at iteration 2019.
The minimum predicted mean of the observed indices is -1.739 at the point
[0.446 0.    0.011 0.011 0.022 0.019]
Iteration 121
Acquisition function convergence reached at iteration 248.
The final EI loss was -0.007 with predicted mean of [-0.502]
The next parameters to simulate from are [[0.001 0.995 0.023 0.01  0.029 0.067]]
The mean of the samples was -0.724
Iteration 122
Acquisition function convergence reached at iteration 341.
The final EI loss was -0.013 with predicted mean of [-0.952]
The next parameters to simulate from are [[0.001 0.    0.011 0.01  0.023 0.037]]
The mean of the samples was -1.361
Iteration 123
Acquisition function convergence reached at iteration 62.
The final EI loss was -0.003 with predicted mean of [0.075]
The next parameters to simulate from are [[0.438 0.991 0.016 0.018 0.049 0.066]]
The mean of the samples was 0.211
Iteration 124
```

```
Acquisition function convergence reached at iteration 111.
The final EI loss was -0.025 with predicted mean of [-1.721]
The next parameters to simulate from are [[0.448 0.001 0.01  0.011 0.022 0.019]]
The mean of the samples was -1.648
Iteration 125
Acquisition function convergence reached at iteration 2.
The final EI loss was -0.0 with predicted mean of [-0.098]
The next parameters to simulate from are [[0.446 0.45  0.    0.011 0.022 0.019]]
The mean of the samples was -0.492
Iteration 126
Acquisition function convergence reached at iteration 2.
The final EI loss was -0.0 with predicted mean of [1.438]
The next parameters to simulate from are [[0.642 0.485 0.011 0.047 0.039 0.019]]
The mean of the samples was 1.353
Iteration 127
Acquisition function convergence reached at iteration 147.
The final EI loss was -0.014 with predicted mean of [-0.95]
The next parameters to simulate from are [[0.001 0.    0.014 0.009 0.029 0.046]]
The mean of the samples was -1.105
Iteration 128
Acquisition function convergence reached at iteration 316.
The final EI loss was -0.008 with predicted mean of [-0.481]
The next parameters to simulate from are [[0.395 0.999 0.033 0.008 0.043 0.042]]
The mean of the samples was -0.481
Iteration 129
Acquisition function convergence reached at iteration 89.
The final EI loss was -0.02 with predicted mean of [-1.726]
The next parameters to simulate from are [[0.453 0.    0.012 0.011 0.022 0.019]]
The mean of the samples was -1.836
Iteration 130
Acquisition function convergence reached at iteration 236.
The final EI loss was -0.003 with predicted mean of [0.158]
The next parameters to simulate from are [[0.999 0.998 0.033 0.    0.035 0.   ]]
The mean of the samples was 0.491
Iteration 131
Acquisition function convergence reached at iteration 184.
The final EI loss was -0.002 with predicted mean of [0.27]
The next parameters to simulate from are [[0.999 0.    0.    0.    0.    0.033]]
The mean of the samples was 0.598
Iteration 132
Acquisition function convergence reached at iteration 122.
The final EI loss was -0.033 with predicted mean of [-1.381]
The next parameters to simulate from are [[0.001 0.723 0.01  0.009 0.031 0.033]]
```

```
The mean of the samples was -1.44
Iteration 133
Acquisition function convergence reached at iteration 76.
The final EI loss was -0.007 with predicted mean of [-1.262]
The next parameters to simulate from are [[0.365 0.    0.007 0.009 0.027 0.017]]
The mean of the samples was -1.548
Iteration 134
Acquisition function convergence reached at iteration 2.
The final EI loss was -0.0 with predicted mean of [0.791]
The next parameters to simulate from are [[0.459 0.255 0.011 0.037 0.022 0.053]]
The mean of the samples was 0.738
Iteration 135
Acquisition function convergence reached at iteration 94.
The final EI loss was -0.025 with predicted mean of [-1.711]
The next parameters to simulate from are [[0.441 0.    0.01  0.011 0.022 0.019]]
The mean of the samples was -1.665
Iteration 136
Acquisition function convergence reached at iteration 2.
The final EI loss was -0.0 with predicted mean of [1.255]
The next parameters to simulate from are [[0.446 0.488 0.011 0.043 0.022 0.01 ]]
The mean of the samples was 1.293
Iteration 137
Acquisition function convergence reached at iteration 121.
The final EI loss was -0.021 with predicted mean of [-1.727]
The next parameters to simulate from are [[0.443 0.    0.011 0.011 0.022 0.019]]
The mean of the samples was -1.731
Iteration 138
Acquisition function convergence reached at iteration 92.
The final EI loss was -0.019 with predicted mean of [-1.728]
The next parameters to simulate from are [[0.449 0.    0.012 0.011 0.022 0.019]]
The mean of the samples was -1.782
Iteration 139
Acquisition function convergence reached at iteration 356.
The final EI loss was -0.019 with predicted mean of [-1.404]
The next parameters to simulate from are [[0.    0.647 0.009 0.008 0.038 0.03 ]]
The mean of the samples was -1.213
Iteration 140
Acquisition function convergence reached at iteration 267.
The final EI loss was -0.004 with predicted mean of [0.085]
The next parameters to simulate from are [[0.999 0.    0.033 0.05  0.    0.067]]
The mean of the samples was 0.584
Hyperparameter convergence reached at iteration 1793.
The minimum predicted mean of the observed indices is -1.739 at the point
```

```
[0.446 0.    0.011 0.011 0.022 0.019]
Iteration 141
Acquisition function convergence reached at iteration 2.
The final EI loss was -0.0 with predicted mean of [-0.531]
The next parameters to simulate from are [[0.446 0.    0.011 0.011 0.022 0.043]]
The mean of the samples was -0.527
Iteration 142
Acquisition function convergence reached at iteration 398.
The final EI loss was -0.008 with predicted mean of [-0.469]
The next parameters to simulate from are [[0.005 0.998 0.033 0.01  0.039 0.067]]
The mean of the samples was -0.817
Iteration 143
Acquisition function convergence reached at iteration 2.
The final EI loss was -0.0 with predicted mean of [0.716]
The next parameters to simulate from are [[0.446 0.    0.011 0.03  0.022 0.019]]
The mean of the samples was 0.674
Iteration 144
Acquisition function convergence reached at iteration 113.
The final EI loss was -0.011 with predicted mean of [-1.284]
The next parameters to simulate from are [[0.05  0.    0.013 0.01  0.029 0.032]]
The mean of the samples was -1.271
Iteration 145
Acquisition function convergence reached at iteration 22.
The final EI loss was -0.006 with predicted mean of [-1.067]
The next parameters to simulate from are [[0.395 0.012 0.022 0.014 0.015 0.027]]
The mean of the samples was -1.159
Iteration 146
Acquisition function convergence reached at iteration 87.
The final EI loss was -0.014 with predicted mean of [-1.181]
The next parameters to simulate from are [[0.    0.    0.014 0.012 0.016 0.035]]
The mean of the samples was -1.347
Iteration 147
Acquisition function convergence reached at iteration 2.
The final EI loss was -0.0 with predicted mean of [0.019]
The next parameters to simulate from are [[0.43  0.    0.011 0.011 0.03  0.002]]
The mean of the samples was 0.211
Iteration 148
Acquisition function convergence reached at iteration 2.
The final EI loss was -0.0 with predicted mean of [-0.389]
The next parameters to simulate from are [[0.824 0.184 0.011 0.011 0.022 0.019]]
The mean of the samples was -0.251
Iteration 149
Acquisition function convergence reached at iteration 112.
```

```
The final EI loss was -0.032 with predicted mean of [-1.384]
The next parameters to simulate from are [[0.001 0.576 0.01  0.01  0.022 0.03 ]]
The mean of the samples was -1.49
Iteration 150
Acquisition function convergence reached at iteration 172.
The final EI loss was -0.01 with predicted mean of [-0.68]
The next parameters to simulate from are [[0.001 0.003 0.028 0.018 0.    0.037]]
The mean of the samples was -0.934
Trained parameters:
amplitude_champ:0 is 0.808

length_scales_champ:0 is [0.402 0.931 0.008 0.005 0.015 0.016]

observation_noise_variance_champ:0 is 0.052

bias_mean:0 is 0.756

Iteration 151
Acquisition function convergence reached at iteration 2.
The final EI loss was -0.0 with predicted mean of [1.192]
The next parameters to simulate from are [[0.446 0.    0.011 0.045 0.049 0.011]]
The mean of the samples was 1.604
Iteration 152
Acquisition function convergence reached at iteration 326.
The final EI loss was -0.013 with predicted mean of [-1.096]
The next parameters to simulate from are [[0.    0.998 0.019 0.013 0.015 0.032]]
The mean of the samples was -1.128
Iteration 153
Acquisition function convergence reached at iteration 2.
The final EI loss was -0.0 with predicted mean of [1.314]
The next parameters to simulate from are [[0.429 0.    0.004 0.046 0.022 0.015]]
The mean of the samples was 1.447
Iteration 154
Acquisition function convergence reached at iteration 86.
The final EI loss was -0.006 with predicted mean of [-0.534]
The next parameters to simulate from are [[0.323 0.984 0.03  0.012 0.033 0.066]]
The mean of the samples was -0.351
Iteration 155
Acquisition function convergence reached at iteration 119.
The final EI loss was -0.032 with predicted mean of [-1.344]
The next parameters to simulate from are [[0.001 0.737 0.007 0.008 0.028 0.023]]
The mean of the samples was -1.302
Iteration 156
```

```
Acquisition function convergence reached at iteration 86.
The final EI loss was -0.007 with predicted mean of [-1.145]
The next parameters to simulate from are [[0.38  0.    0.028 0.015 0.019 0.033]]
The mean of the samples was -1.15
Iteration 157
Acquisition function convergence reached at iteration 132.
The final EI loss was -0.009 with predicted mean of [-0.819]
The next parameters to simulate from are [[0.    0.187 0.019 0.019 0.003 0.038]]
The mean of the samples was -0.912
Iteration 158
Acquisition function convergence reached at iteration 30.
The final EI loss was -0.012 with predicted mean of [-0.983]
The next parameters to simulate from are [[0.447 0.    0.021 0.01  0.033 0.026]]
The mean of the samples was -1.207
Iteration 159
Acquisition function convergence reached at iteration 175.
The final EI loss was -0.003 with predicted mean of [-0.144]
The next parameters to simulate from are [[1.    0.998 0.024 0.013 0.01  0.   ]]
The mean of the samples was 0.272
Iteration 160
Acquisition function convergence reached at iteration 159.
The final EI loss was -0.001 with predicted mean of [0.305]
The next parameters to simulate from are [[1.    0.999 0.    0.041 0.05  0.   ]]
The mean of the samples was 0.538
Hyperparameter convergence reached at iteration 1999.
The minimum predicted mean of the observed indices is -1.736 at the point
[0.449 0.    0.012 0.011 0.022 0.019]
Iteration 161
Acquisition function convergence reached at iteration 2.
The final EI loss was -0.0 with predicted mean of [-0.791]
The next parameters to simulate from are [[0.449 0.88  0.012 0.011 0.022 0.019]]
The mean of the samples was -0.75
Iteration 162
Acquisition function convergence reached at iteration 234.
The final EI loss was -0.002 with predicted mean of [0.107]
The next parameters to simulate from are [[0.999 0.    0.033 0.03  0.    0.034]]
The mean of the samples was 0.439
Iteration 163
Acquisition function convergence reached at iteration 286.
The final EI loss was -0.009 with predicted mean of [-0.575]
The next parameters to simulate from are [[0.005 0.    0.033 0.009 0.045 0.052]]
The mean of the samples was -0.75
Iteration 164
```

```
Acquisition function convergence reached at iteration 77.
The final EI loss was -0.003 with predicted mean of [-0.005]
The next parameters to simulate from are [[0.007 0.    0.    0.016 0.003 0.   ]]
The mean of the samples was 0.524
Iteration 165
Acquisition function convergence reached at iteration 212.
The final EI loss was -0.004 with predicted mean of [-0.449]
The next parameters to simulate from are [[0.001 0.    0.033 0.01  0.05  0.033]]
The mean of the samples was -0.514
Iteration 166
Acquisition function convergence reached at iteration 81.
The final EI loss was -0.005 with predicted mean of [-0.143]
The next parameters to simulate from are [[0.006 0.    0.015 0.019 0.    0.066]]
The mean of the samples was -0.057
Iteration 167
Acquisition function convergence reached at iteration 2.
The final EI loss was -0.0 with predicted mean of [0.968]
The next parameters to simulate from are [[0.714 0.    0.008 0.037 0.022 0.019]]
The mean of the samples was 1.016
Iteration 168
Acquisition function convergence reached at iteration 2.
The final EI loss was -0.0 with predicted mean of [1.119]
The next parameters to simulate from are [[0.456 0.806 0.007 0.046 0.036 0.057]]
The mean of the samples was 1.102
Iteration 169
Acquisition function convergence reached at iteration 89.
The final EI loss was -0.015 with predicted mean of [-1.359]
The next parameters to simulate from are [[0.205 0.206 0.007 0.009 0.023 0.022]]
The mean of the samples was -1.547
Iteration 170
Acquisition function convergence reached at iteration 2.
The final EI loss was -0.0 with predicted mean of [0.444]
The next parameters to simulate from are [[0.446 0.    0.015 0.026 0.022 0.017]]
The mean of the samples was 0.384
Iteration 171
Acquisition function convergence reached at iteration 93.
The final EI loss was -0.009 with predicted mean of [-1.069]
The next parameters to simulate from are [[0.416 0.    0.013 0.007 0.039 0.025]]
The mean of the samples was -1.344
Iteration 172
Acquisition function convergence reached at iteration 70.
The final EI loss was -0.009 with predicted mean of [-0.694]
The next parameters to simulate from are [[0.012 0.    0.021 0.009 0.041 0.047]]
```

```
The mean of the samples was -0.864
Iteration 173
Acquisition function convergence reached at iteration 52.
The final EI loss was -0.003 with predicted mean of [-0.407]
The next parameters to simulate from are [[0.485 0.    0.027 0.014 0.001 0.019]]
The mean of the samples was -0.634
Iteration 174
Acquisition function convergence reached at iteration 75.
The final EI loss was -0.002 with predicted mean of [-0.34]
The next parameters to simulate from are [[0.611 0.    0.024 0.008 0.009 0.018]]
The mean of the samples was -0.293
Iteration 175
Acquisition function convergence reached at iteration 211.
The final EI loss was -0.002 with predicted mean of [0.139]
The next parameters to simulate from are [[0.998 0.    0.    0.017 0.    0.   ]]
The mean of the samples was 0.371
Iteration 176
Acquisition function convergence reached at iteration 194.
The final EI loss was -0.01 with predicted mean of [-0.852]
The next parameters to simulate from are [[0.418 0.002 0.02  0.008 0.045 0.031]]
The mean of the samples was -1.128
Iteration 177
Acquisition function convergence reached at iteration 2.
The final EI loss was -0.0 with predicted mean of [1.338]
The next parameters to simulate from are [[0.655 0.    0.011 0.05  0.022 0.006]]
The mean of the samples was 1.461
Iteration 178
Acquisition function convergence reached at iteration 149.
The final EI loss was -0.021 with predicted mean of [-1.317]
The next parameters to simulate from are [[0.002 0.596 0.01  0.013 0.015 0.028]]
The mean of the samples was -1.604
Iteration 179
Acquisition function convergence reached at iteration 191.
The final EI loss was -0.002 with predicted mean of [0.146]
The next parameters to simulate from are [[0.996 0.997 0.021 0.021 0.02  0.067]]
The mean of the samples was 0.376
Iteration 180
Acquisition function convergence reached at iteration 100.
The final EI loss was -0.01 with predicted mean of [-0.897]
The next parameters to simulate from are [[0.34  0.    0.026 0.01  0.04  0.042]]
The mean of the samples was -0.813
Hyperparameter convergence reached at iteration 1947.
The minimum predicted mean of the observed indices is -1.735 at the point
```

```
[0.446 0.    0.011 0.011 0.022 0.019]
Iteration 181
Acquisition function convergence reached at iteration 73.
The final EI loss was -0.009 with predicted mean of [-0.661]
The next parameters to simulate from are [[0.246 0.956 0.018 0.008 0.046 0.027]]
The mean of the samples was -0.746
Iteration 182
Acquisition function convergence reached at iteration 174.
The final EI loss was -0.002 with predicted mean of [0.221]
The next parameters to simulate from are [[0.998 0.002 0.017 0.007 0.    0.067]]
The mean of the samples was 0.554
Iteration 183
Acquisition function convergence reached at iteration 121.
The final EI loss was -0.014 with predicted mean of [-1.367]
The next parameters to simulate from are [[0.083 0.    0.008 0.01  0.018 0.027]]
The mean of the samples was -1.494
Iteration 184
Acquisition function convergence reached at iteration 116.
The final EI loss was -0.023 with predicted mean of [-1.714]
The next parameters to simulate from are [[0.464 0.    0.012 0.011 0.022 0.02 ]]
The mean of the samples was -1.588
Iteration 185
Acquisition function convergence reached at iteration 115.
The final EI loss was -0.012 with predicted mean of [-1.374]
The next parameters to simulate from are [[0.223 0.    0.008 0.012 0.015 0.023]]
The mean of the samples was -1.751
Iteration 186
Acquisition function convergence reached at iteration 2.
The final EI loss was -0.0 with predicted mean of [-0.552]
The next parameters to simulate from are [[0.446 0.    0.011 0.011 0.045 0.019]]
The mean of the samples was -0.444
Iteration 187
Acquisition function convergence reached at iteration 2.
The final EI loss was -0.0 with predicted mean of [0.99]
The next parameters to simulate from are [[0.961 0.464 0.009 0.047 0.022 0.006]]
The mean of the samples was 1.02
Iteration 188
Acquisition function convergence reached at iteration 2.
The final EI loss was -0.0 with predicted mean of [-0.175]
The next parameters to simulate from are [[0.446 0.299 0.011 0.011 0.    0.019]]
The mean of the samples was -0.165
Iteration 189
WARNING:tensorflow:5 out of the last 122 calls to <function update_var_EI.<locals>.opt_var at
```

```
Acquisition function convergence reached at iteration 210.
The final EI loss was -0.002 with predicted mean of [0.188]
The next parameters to simulate from are [[0.998 0.    0.    0.    0.05  0.067]]
The mean of the samples was 0.585
Iteration 190
Acquisition function convergence reached at iteration 2.
The final EI loss was -0.0 with predicted mean of [1.451]
The next parameters to simulate from are [[0.101 0.413 0.011 0.046 0.014 0.019]]
The mean of the samples was 1.57
Iteration 191
Acquisition function convergence reached at iteration 269.
The final EI loss was -0.007 with predicted mean of [-0.898]
The next parameters to simulate from are [[0.    0.999 0.019 0.013 0.018 0.049]]
The mean of the samples was -1.01
Iteration 192
Acquisition function convergence reached at iteration 74.
The final EI loss was -0.021 with predicted mean of [-1.609]
The next parameters to simulate from are [[0.327 0.    0.008 0.011 0.021 0.021]]
The mean of the samples was -1.911
Iteration 193
Acquisition function convergence reached at iteration 94.
The final EI loss was -0.003 with predicted mean of [-0.808]
The next parameters to simulate from are [[0.549 0.003 0.017 0.016 0.009 0.017]]
The mean of the samples was -1.091
Iteration 194
Acquisition function convergence reached at iteration 279.
The final EI loss was -0.005 with predicted mean of [-0.689]
The next parameters to simulate from are [[0.399 0.787 0.026 0.01  0.05  0.027]]
The mean of the samples was -0.826
Iteration 195
Acquisition function convergence reached at iteration 47.
The final EI loss was -0.005 with predicted mean of [-1.175]
The next parameters to simulate from are [[0.306 0.    0.018 0.009 0.035 0.035]]
The mean of the samples was -1.191
Iteration 196
Acquisition function convergence reached at iteration 85.
The final EI loss was -0.005 with predicted mean of [-0.931]
The next parameters to simulate from are [[0.596 0.    0.019 0.008 0.039 0.02 ]]
The mean of the samples was -1.198
Iteration 197
Acquisition function convergence reached at iteration 2.
The final EI loss was -0.0 with predicted mean of [0.123]
The next parameters to simulate from are [[0.324 0.    0.008 0.022 0.021 0.021]]
```

```
The mean of the samples was 0.125
Iteration 198
Acquisition function convergence reached at iteration 86.
The final EI loss was -0.004 with predicted mean of [-1.541]
The next parameters to simulate from are [[0.495 0.    0.014 0.011 0.025 0.02 ]]
The mean of the samples was -1.642
Iteration 199
Acquisition function convergence reached at iteration 212.
The final EI loss was -0.003 with predicted mean of [0.048]
The next parameters to simulate from are [[0.998 0.    0.    0.    0.05  0.   ]]
The mean of the samples was 0.963
Iteration 200
Acquisition function convergence reached at iteration 35.
The final EI loss was -0.002 with predicted mean of [-0.898]
The next parameters to simulate from are [[0.648 0.005 0.02  0.014 0.019 0.016]]
The mean of the samples was -1.323
Hyperparameter convergence reached at iteration 1975.
The minimum predicted mean of the observed indices is -1.797 at the point
[0.326 0.    0.008 0.011 0.021 0.021]
Trained parameters:
amplitude_champ:0 is 0.789

length_scales_champ:0 is [0.377 1.132 0.007 0.006 0.016 0.017]

observation_noise_variance_champ:0 is 0.061

bias_mean:0 is 0.794

Iteration 201
Acquisition function convergence reached at iteration 112.
The final EI loss was -0.044 with predicted mean of [-1.795]
The next parameters to simulate from are [[0.316 0.    0.008 0.011 0.021 0.021]]
The mean of the samples was -1.714
Iteration 202
Acquisition function convergence reached at iteration 177.
The final EI loss was -0.038 with predicted mean of [-1.776]
The next parameters to simulate from are [[0.322 0.004 0.008 0.01  0.021 0.021]]
The mean of the samples was -1.711
Iteration 203
Acquisition function convergence reached at iteration 74.
The final EI loss was -0.006 with predicted mean of [-0.791]
The next parameters to simulate from are [[0.557 0.    0.016 0.006 0.05  0.021]]
The mean of the samples was -1.068
```

```
Iteration 204
Acquisition function convergence reached at iteration 75.
The final EI loss was -0.0 with predicted mean of [0.566]
The next parameters to simulate from are [[1.    0.    0.019 0.039 0.05  0.   ]]
The mean of the samples was 1.335
Iteration 205
Acquisition function convergence reached at iteration 126.
The final EI loss was -0.001 with predicted mean of [0.247]
The next parameters to simulate from are [[0.997 0.    0.022 0.019 0.05  0.067]]
The mean of the samples was 0.753
Iteration 206
Acquisition function convergence reached at iteration 70.
The final EI loss was -0.035 with predicted mean of [-1.749]
The next parameters to simulate from are [[0.344 0.    0.009 0.011 0.022 0.021]]
The mean of the samples was -1.706
Iteration 207
Acquisition function convergence reached at iteration 62.
The final EI loss was -0.006 with predicted mean of [-1.052]
The next parameters to simulate from are [[0.519 0.601 0.018 0.009 0.04  0.024]]
The mean of the samples was -0.731
Iteration 208
Acquisition function convergence reached at iteration 96.
The final EI loss was -0.032 with predicted mean of [-1.745]
The next parameters to simulate from are [[0.345 0.    0.009 0.011 0.021 0.021]]
The mean of the samples was -1.55
Iteration 209
Acquisition function convergence reached at iteration 2.
The final EI loss was -0.0 with predicted mean of [0.561]
The next parameters to simulate from are [[0.446 0.    0.011 0.027 0.022 0.019]]
The mean of the samples was 0.496
Iteration 210
Acquisition function convergence reached at iteration 180.
The final EI loss was -0.002 with predicted mean of [-0.036]
The next parameters to simulate from are [[0.001 0.    0.    0.001 0.05  0.067]]
The mean of the samples was 0.103
Iteration 211
Acquisition function convergence reached at iteration 256.
The final EI loss was -0.007 with predicted mean of [-0.765]
The next parameters to simulate from are [[0.001 1.    0.022 0.009 0.037 0.042]]
The mean of the samples was -0.973
Iteration 212
Acquisition function convergence reached at iteration 102.
The final EI loss was -0.015 with predicted mean of [-1.405]
```

```
The next parameters to simulate from are [[0.128 0.547 0.007 0.006 0.033 0.027]]
The mean of the samples was -1.161
Iteration 213
Acquisition function convergence reached at iteration 2.
The final EI loss was -0.0 with predicted mean of [-0.25]
The next parameters to simulate from are [[0.446 0.521 0.011 0.011 0.001 0.016]]
The mean of the samples was -0.329
Iteration 214
Acquisition function convergence reached at iteration 79.
The final EI loss was -0.011 with predicted mean of [-1.309]
The next parameters to simulate from are [[0.567 0.    0.018 0.012 0.028 0.019]]
The mean of the samples was -1.471
Iteration 215
Acquisition function convergence reached at iteration 136.
The final EI loss was -0.001 with predicted mean of [0.269]
The next parameters to simulate from are [[0.998 0.002 0.016 0.019 0.    0.067]]
The mean of the samples was 0.478
Iteration 216
Acquisition function convergence reached at iteration 236.
The final EI loss was -0.005 with predicted mean of [-0.498]
The next parameters to simulate from are [[0.002 0.001 0.027 0.009 0.048 0.067]]
The mean of the samples was -0.59
Iteration 217
Acquisition function convergence reached at iteration 2.
The final EI loss was -0.0 with predicted mean of [-0.56]
The next parameters to simulate from are [[0.444 0.    0.009 0.011 0.022 0.038]]
The mean of the samples was -0.653
Iteration 218
Acquisition function convergence reached at iteration 206.
The final EI loss was -0.036 with predicted mean of [-1.611]
The next parameters to simulate from are [[0.169 0.326 0.008 0.012 0.018 0.024]]
The mean of the samples was -1.694
Iteration 219
Acquisition function convergence reached at iteration 170.
The final EI loss was -0.037 with predicted mean of [-1.701]
The next parameters to simulate from are [[0.243 0.143 0.008 0.011 0.02  0.023]]
The mean of the samples was -1.737
Iteration 220
Acquisition function convergence reached at iteration 57.
The final EI loss was -0.005 with predicted mean of [-1.062]
The next parameters to simulate from are [[0.612 0.527 0.019 0.014 0.022 0.015]]
The mean of the samples was -1.071
Hyperparameter convergence reached at iteration 2101.
```

```
The minimum predicted mean of the observed indices is -1.725 at the point
[0.446 0.    0.011 0.011 0.022 0.019]
Iteration 221
Acquisition function convergence reached at iteration 187.
The final EI loss was -0.002 with predicted mean of [0.199]
The next parameters to simulate from are [[0.001 0.999 0.033 0.05  0.    0.067]]
The mean of the samples was 0.996
Iteration 222
Acquisition function convergence reached at iteration 115.
The final EI loss was -0.001 with predicted mean of [0.423]
The next parameters to simulate from are [[1.    0.    0.02  0.039 0.05  0.067]]
The mean of the samples was 1.164
Iteration 223
Acquisition function convergence reached at iteration 59.
The final EI loss was -0.006 with predicted mean of [-0.79]
The next parameters to simulate from are [[0.637 0.003 0.024 0.008 0.05  0.021]]
The mean of the samples was -1.156
Iteration 224
Acquisition function convergence reached at iteration 196.
The final EI loss was -0.002 with predicted mean of [0.113]
The next parameters to simulate from are [[0.002 0.    0.    0.026 0.    0.067]]
The mean of the samples was 0.446
Iteration 225
Acquisition function convergence reached at iteration 2.
The final EI loss was -0.0 with predicted mean of [1.213]
The next parameters to simulate from are [[0.445 0.869 0.011 0.049 0.023 0.008]]
The mean of the samples was 1.456
Iteration 226
Acquisition function convergence reached at iteration 99.
The final EI loss was -0.03 with predicted mean of [-1.594]
The next parameters to simulate from are [[0.076 0.372 0.008 0.012 0.016 0.026]]
The mean of the samples was -1.651
Iteration 227
Acquisition function convergence reached at iteration 257.
The final EI loss was -0.031 with predicted mean of [-1.727]
The next parameters to simulate from are [[0.381 0.005 0.01  0.011 0.021 0.02 ]]
The mean of the samples was -1.792
Iteration 228
Acquisition function convergence reached at iteration 2.
The final EI loss was -0.0 with predicted mean of [1.353]
The next parameters to simulate from are [[0.196 0.993 0.015 0.046 0.031 0.02 ]]
The mean of the samples was 1.52
Iteration 229
```

```
Acquisition function convergence reached at iteration 111.
The final EI loss was -0.029 with predicted mean of [-1.728]
The next parameters to simulate from are [[0.369 0.04  0.009 0.011 0.021 0.02 ]]
The mean of the samples was -1.969
Iteration 230
Acquisition function convergence reached at iteration 16.
The final EI loss was -0.0 with predicted mean of [0.412]
The next parameters to simulate from are [[0.945 0.022 0.004 0.019 0.012 0.049]]
The mean of the samples was 0.389
Iteration 231
Acquisition function convergence reached at iteration 56.
The final EI loss was -0.005 with predicted mean of [-0.934]
The next parameters to simulate from are [[0.683 0.    0.024 0.011 0.032 0.016]]
The mean of the samples was -1.494
Iteration 232
Acquisition function convergence reached at iteration 267.
The final EI loss was -0.015 with predicted mean of [-1.467]
The next parameters to simulate from are [[0.    0.501 0.012 0.013 0.016 0.032]]
The mean of the samples was -1.623
Iteration 233
Acquisition function convergence reached at iteration 186.
The final EI loss was -0.011 with predicted mean of [-1.182]
The next parameters to simulate from are [[0.001 0.996 0.01  0.015 0.014 0.027]]
The mean of the samples was -1.265
Iteration 234
Acquisition function convergence reached at iteration 2.
The final EI loss was -0.0 with predicted mean of [-0.622]
The next parameters to simulate from are [[0.381 0.005 0.01  0.011 0.007 0.02 ]]
The mean of the samples was -0.591
Iteration 235
Acquisition function convergence reached at iteration 2.
The final EI loss was -0.0 with predicted mean of [1.277]
The next parameters to simulate from are [[0.224 0.409 0.015 0.041 0.021 0.015]]
The mean of the samples was 1.311
Iteration 236
Acquisition function convergence reached at iteration 166.
The final EI loss was -0.004 with predicted mean of [-0.594]
The next parameters to simulate from are [[0.002 0.003 0.016 0.007 0.038 0.061]]
The mean of the samples was -0.657
Iteration 237
Acquisition function convergence reached at iteration 92.
The final EI loss was -0.009 with predicted mean of [-0.844]
The next parameters to simulate from are [[0.196 0.999 0.009 0.006 0.048 0.021]]
```

```
The mean of the samples was -0.911
Iteration 238
Acquisition function convergence reached at iteration 319.
The final EI loss was -0.006 with predicted mean of [-0.52]
The next parameters to simulate from are [[0.002 0.    0.033 0.02  0.004 0.056]]
The mean of the samples was -0.896
Iteration 239
Acquisition function convergence reached at iteration 81.
The final EI loss was -0.009 with predicted mean of [-1.253]
The next parameters to simulate from are [[0.701 0.    0.022 0.012 0.027 0.012]]
The mean of the samples was -1.27
Iteration 240
Acquisition function convergence reached at iteration 136.
The final EI loss was -0.028 with predicted mean of [-1.746]
The next parameters to simulate from are [[0.379 0.    0.009 0.011 0.021 0.02 ]]
The mean of the samples was -1.749
Hyperparameter convergence reached at iteration 2075.
The minimum predicted mean of the observed indices is -1.739 at the point
[0.381 0.005 0.01  0.011 0.021 0.02 ]
Iteration 241
Acquisition function convergence reached at iteration 124.
The final EI loss was -0.015 with predicted mean of [-1.584]
The next parameters to simulate from are [[0.109 0.281 0.008 0.011 0.021 0.027]]
The mean of the samples was -1.714
Iteration 242
Acquisition function convergence reached at iteration 192.
The final EI loss was -0.005 with predicted mean of [-0.691]
The next parameters to simulate from are [[0.444 1.    0.012 0.01  0.05  0.013]]
The mean of the samples was -0.859
Iteration 243
Acquisition function convergence reached at iteration 2.
The final EI loss was -0.0 with predicted mean of [1.226]
The next parameters to simulate from are [[0.381 0.816 0.016 0.045 0.008 0.008]]
The mean of the samples was 1.27
Iteration 244
Acquisition function convergence reached at iteration 2.
The final EI loss was -0.0 with predicted mean of [1.281]
The next parameters to simulate from are [[0.381 0.726 0.01  0.049 0.021 0.026]]
The mean of the samples was 1.373
Iteration 245
Acquisition function convergence reached at iteration 168.
The final EI loss was -0.002 with predicted mean of [-0.213]
The next parameters to simulate from are [[0.002 0.003 0.015 0.005 0.02  0.067]]
```

```
The mean of the samples was -0.083
Iteration 246
Acquisition function convergence reached at iteration 58.
The final EI loss was -0.004 with predicted mean of [-0.995]
The next parameters to simulate from are [[0.128 0.997 0.004 0.008 0.037 0.022]]
The mean of the samples was -1.7
Iteration 247
Acquisition function convergence reached at iteration 80.
The final EI loss was -0.008 with predicted mean of [-1.208]
The next parameters to simulate from are [[0.59  0.    0.025 0.011 0.04  0.018]]
The mean of the samples was -1.034
Iteration 248
Acquisition function convergence reached at iteration 2.
The final EI loss was -0.0 with predicted mean of [0.152]
The next parameters to simulate from are [[0.381 0.005 0.01  0.021 0.021 0.033]]
The mean of the samples was 0.135
Iteration 249
Acquisition function convergence reached at iteration 308.
The final EI loss was -0.003 with predicted mean of [-0.827]
The next parameters to simulate from are [[0.003 0.999 0.025 0.012 0.031 0.034]]
The mean of the samples was -0.831
Iteration 250
Acquisition function convergence reached at iteration 34.
The final EI loss was -0.001 with predicted mean of [-0.838]
The next parameters to simulate from are [[0.262 0.001 0.017 0.009 0.037 0.048]]
The mean of the samples was -0.853
Trained parameters:
amplitude_champ:0 is 0.774

length_scales_champ:0 is [0.409 0.912 0.008 0.006 0.016 0.017]

observation_noise_variance_champ:0 is 0.067

bias_mean:0 is 0.845

Iteration 251
Acquisition function convergence reached at iteration 2.
The final EI loss was -0.0 with predicted mean of [0.247]
The next parameters to simulate from are [[0.996 0.005 0.013 0.011 0.021 0.02 ]]
The mean of the samples was 0.362
Iteration 252
Acquisition function convergence reached at iteration 57.
The final EI loss was -0.001 with predicted mean of [0.178]
```

```
The next parameters to simulate from are [[0.999 0.99  0.019 0.036 0.    0.044]]
The mean of the samples was 0.419
Iteration 253
Acquisition function convergence reached at iteration 49.
The final EI loss was -0.001 with predicted mean of [0.033]
The next parameters to simulate from are [[0.991 0.003 0.    0.01  0.024 0.   ]]
The mean of the samples was 0.57
Iteration 254
Acquisition function convergence reached at iteration 88.
The final EI loss was -0.005 with predicted mean of [-0.71]
The next parameters to simulate from are [[0.499 0.998 0.022 0.016 0.039 0.019]]
The mean of the samples was -0.907
Iteration 255
Acquisition function convergence reached at iteration 2.
The final EI loss was -0.0 with predicted mean of [0.842]
The next parameters to simulate from are [[0.686 0.005 0.01  0.037 0.021 0.034]]
The mean of the samples was 0.869
Iteration 256
Acquisition function convergence reached at iteration 112.
The final EI loss was -0.023 with predicted mean of [-1.679]
The next parameters to simulate from are [[0.128 0.293 0.009 0.011 0.02  0.027]]
The mean of the samples was -1.788
Iteration 257
Acquisition function convergence reached at iteration 2.
The final EI loss was -0.0 with predicted mean of [0.563]
The next parameters to simulate from are [[0.381 0.231 0.01  0.028 0.021 0.02 ]]
The mean of the samples was 0.538
Iteration 258
Acquisition function convergence reached at iteration 2.
The final EI loss was -0.0 with predicted mean of [0.186]
The next parameters to simulate from are [[0.869 0.005 0.015 0.011 0.024 0.041]]
The mean of the samples was 0.105
Iteration 259
Acquisition function convergence reached at iteration 209.
The final EI loss was -0.003 with predicted mean of [-0.676]
The next parameters to simulate from are [[0.474 0.001 0.025 0.005 0.05  0.031]]
The mean of the samples was -0.58
Iteration 260
Acquisition function convergence reached at iteration 2.
The final EI loss was -0.0 with predicted mean of [0.08]
The next parameters to simulate from are [[0.381 0.06  0.01  0.022 0.021 0.02 ]]
The mean of the samples was 0.043
Hyperparameter convergence reached at iteration 2093.
```

```
The minimum predicted mean of the observed indices is -1.74 at the point
[0.381 0.005 0.01  0.011 0.021 0.02 ]
Iteration 261
Acquisition function convergence reached at iteration 116.
The final EI loss was -0.001 with predicted mean of [0.288]
The next parameters to simulate from are [[0.998 0.    0.    0.034 0.    0.034]]
The mean of the samples was 0.494
Iteration 262
Acquisition function convergence reached at iteration 108.
The final EI loss was -0.026 with predicted mean of [-1.71]
The next parameters to simulate from are [[0.108 0.338 0.009 0.011 0.02  0.027]]
The mean of the samples was -1.428
Iteration 263
Acquisition function convergence reached at iteration 2.
The final EI loss was -0.0 with predicted mean of [1.013]
The next parameters to simulate from are [[0.86  0.005 0.01  0.042 0.021 0.02 ]]
The mean of the samples was 1.04
Iteration 264
Acquisition function convergence reached at iteration 2.
The final EI loss was -0.0 with predicted mean of [-0.137]
The next parameters to simulate from are [[0.381 0.275 0.01  0.011 0.021 0.061]]
The mean of the samples was -0.092
Iteration 265
Acquisition function convergence reached at iteration 2.
The final EI loss was -0.0 with predicted mean of [-0.579]
The next parameters to simulate from are [[0.381 0.112 0.01  0.011 0.042 0.02 ]]
The mean of the samples was -0.743
Iteration 266
WARNING:tensorflow:5 out of the last 115 calls to <function update_var_EI.<locals>.opt_var at
Acquisition function convergence reached at iteration 2.
The final EI loss was -0.0 with predicted mean of [-0.833]
The next parameters to simulate from are [[0.381 0.005 0.012 0.012 0.033 0.02 ]]
The mean of the samples was -0.766
Iteration 267
Acquisition function convergence reached at iteration 82.
The final EI loss was -0.027 with predicted mean of [-1.465]
The next parameters to simulate from are [[0.222 0.994 0.007 0.009 0.039 0.02 ]]
The mean of the samples was -1.619
Iteration 268
Acquisition function convergence reached at iteration 58.
The final EI loss was -0.003 with predicted mean of [-0.761]
The next parameters to simulate from are [[0.005 0.    0.026 0.015 0.03  0.053]]
The mean of the samples was -0.663
```

```
Iteration 269
Acquisition function convergence reached at iteration 2.
The final EI loss was -0.0 with predicted mean of [0.055]
The next parameters to simulate from are [[0.381 0.005 0.005 0.011 0.034 0.036]]
The mean of the samples was -0.054
Iteration 270
Acquisition function convergence reached at iteration 2.
The final EI loss was -0.0 with predicted mean of [-0.009]
The next parameters to simulate from are [[0.33  0.005 0.001 0.011 0.021 0.03 ]]
The mean of the samples was 0.208
Iteration 271
Acquisition function convergence reached at iteration 2.
The final EI loss was -0.0 with predicted mean of [-0.527]
The next parameters to simulate from are [[0.381 0.005 0.01  0.011 0.021 0.044]]
The mean of the samples was -0.539
Iteration 272
Acquisition function convergence reached at iteration 84.
The final EI loss was -0.004 with predicted mean of [-0.723]
The next parameters to simulate from are [[0.298 0.    0.033 0.01  0.033 0.049]]
The mean of the samples was -0.833
Iteration 273
Acquisition function convergence reached at iteration 81.
The final EI loss was -0.018 with predicted mean of [-1.676]
The next parameters to simulate from are [[0.16  0.28  0.008 0.011 0.02  0.024]]
The mean of the samples was -1.782
Iteration 274
Acquisition function convergence reached at iteration 108.
The final EI loss was -0.0 with predicted mean of [0.303]
The next parameters to simulate from are [[0.998 0.    0.    0.    0.029 0.036]]
The mean of the samples was 0.293
Iteration 275
Acquisition function convergence reached at iteration 108.
The final EI loss was -0.02 with predicted mean of [-1.639]
The next parameters to simulate from are [[0.067 0.395 0.007 0.01  0.02  0.026]]
The mean of the samples was -1.761
Iteration 276
Acquisition function convergence reached at iteration 63.
The final EI loss was -0.003 with predicted mean of [-0.64]
The next parameters to simulate from are [[0.601 0.92  0.033 0.015 0.024 0.023]]
The mean of the samples was -0.858
Iteration 277
Acquisition function convergence reached at iteration 2.
The final EI loss was -0.0 with predicted mean of [-0.716]
```

```
The next parameters to simulate from are [[0.702 0.207 0.01  0.011 0.021 0.02 ]]
The mean of the samples was -0.55
Iteration 278
Acquisition function convergence reached at iteration 2.
The final EI loss was -0.0 with predicted mean of [0.248]
The next parameters to simulate from are [[0.894 0.387 0.01  0.011 0.011 0.02 ]]
The mean of the samples was 0.206
Iteration 279
Acquisition function convergence reached at iteration 70.
The final EI loss was -0.027 with predicted mean of [-1.743]
The next parameters to simulate from are [[0.394 0.    0.01  0.011 0.021 0.02 ]]
The mean of the samples was -1.752
Iteration 280
Acquisition function convergence reached at iteration 150.
The final EI loss was -0.02 with predicted mean of [-1.54]
The next parameters to simulate from are [[0.16  1.    0.006 0.009 0.036 0.022]]
The mean of the samples was -1.827
Hyperparameter convergence reached at iteration 2129.
The minimum predicted mean of the observed indices is -1.744 at the point
[0.394 0.    0.01  0.011 0.021 0.02 ]
Iteration 281
Acquisition function convergence reached at iteration 153.
The final EI loss was -0.001 with predicted mean of [0.117]
The next parameters to simulate from are [[0.003 0.998 0.    0.    0.026 0.   ]]
The mean of the samples was 0.518
Iteration 282
Acquisition function convergence reached at iteration 78.
The final EI loss was -0.002 with predicted mean of [-0.974]
The next parameters to simulate from are [[0.323 0.    0.027 0.011 0.025 0.039]]
The mean of the samples was -0.928
Iteration 283
Acquisition function convergence reached at iteration 2.
The final EI loss was -0.0 with predicted mean of [-0.869]
The next parameters to simulate from are [[0.394 0.    0.01  0.011 0.034 0.02 ]]
The mean of the samples was -0.908
Iteration 284
Acquisition function convergence reached at iteration 129.
The final EI loss was -0.002 with predicted mean of [-0.467]
The next parameters to simulate from are [[0.302 0.352 0.033 0.008 0.034 0.067]]
The mean of the samples was -0.359
Iteration 285
Acquisition function convergence reached at iteration 2.
The final EI loss was -0.0 with predicted mean of [0.01]
```

```
The next parameters to simulate from are [[0.394 0.    0.01  0.011 0.004 0.035]]
The mean of the samples was -0.011
Iteration 286
Acquisition function convergence reached at iteration 170.
The final EI loss was -0.002 with predicted mean of [0.063]
The next parameters to simulate from are [[0.997 0.    0.033 0.    0.05  0.067]]
The mean of the samples was 0.437
Iteration 287
Acquisition function convergence reached at iteration 2.
The final EI loss was -0.0 with predicted mean of [-0.719]
The next parameters to simulate from are [[0.393 0.318 0.01  0.013 0.021 0.034]]
The mean of the samples was -0.672
Iteration 288
Acquisition function convergence reached at iteration 98.
The final EI loss was -0.045 with predicted mean of [-1.696]
The next parameters to simulate from are [[0.182 0.864 0.006 0.009 0.036 0.021]]
The mean of the samples was -1.804
Iteration 289
Acquisition function convergence reached at iteration 123.
The final EI loss was -0.005 with predicted mean of [-1.136]
The next parameters to simulate from are [[0.641 0.386 0.024 0.013 0.031 0.016]]
The mean of the samples was -1.145
Iteration 290
Acquisition function convergence reached at iteration 218.
The final EI loss was -0.003 with predicted mean of [-0.485]
The next parameters to simulate from are [[0.67  0.999 0.027 0.015 0.05  0.009]]
The mean of the samples was -0.693
Iteration 291
Acquisition function convergence reached at iteration 77.
The final EI loss was -0.0 with predicted mean of [0.255]
The next parameters to simulate from are [[0.597 0.    0.    0.009 0.    0.   ]]
The mean of the samples was 0.292
Iteration 292
Acquisition function convergence reached at iteration 95.
The final EI loss was -0.003 with predicted mean of [-0.414]
The next parameters to simulate from are [[0.001 0.    0.033 0.025 0.    0.066]]
The mean of the samples was -0.704
Iteration 293
Acquisition function convergence reached at iteration 138.
The final EI loss was -0.007 with predicted mean of [-0.76]
The next parameters to simulate from are [[0.006 0.584 0.033 0.02  0.    0.04 ]]
The mean of the samples was -0.888
Iteration 294
```

```
Acquisition function convergence reached at iteration 87.
The final EI loss was -0.009 with predicted mean of [-1.28]
The next parameters to simulate from are [[0.    0.513 0.018 0.014 0.014 0.036]]
The mean of the samples was -1.291
Iteration 295
Acquisition function convergence reached at iteration 2.
The final EI loss was -0.0 with predicted mean of [-0.719]
The next parameters to simulate from are [[0.394 0.    0.003 0.011 0.022 0.02 ]]
The mean of the samples was -0.626
Iteration 296
Acquisition function convergence reached at iteration 76.
The final EI loss was -0.022 with predicted mean of [-1.697]
The next parameters to simulate from are [[0.149 0.307 0.008 0.011 0.021 0.024]]
The mean of the samples was -1.655
Iteration 297
Acquisition function convergence reached at iteration 2.
The final EI loss was -0.0 with predicted mean of [1.288]
The next parameters to simulate from are [[0.394 0.244 0.006 0.04  0.034 0.039]]
The mean of the samples was 1.279
Iteration 298
Acquisition function convergence reached at iteration 2.
The final EI loss was -0.0 with predicted mean of [1.134]
The next parameters to simulate from are [[0.772 0.    0.003 0.042 0.029 0.02 ]]
The mean of the samples was 1.462
Iteration 299
Acquisition function convergence reached at iteration 181.
The final EI loss was -0.053 with predicted mean of [-1.732]
The next parameters to simulate from are [[0.111 1.    0.006 0.009 0.036 0.022]]
The mean of the samples was -1.565
Iteration 300
Acquisition function convergence reached at iteration 2.
The final EI loss was -0.0 with predicted mean of [-0.063]
The next parameters to simulate from are [[0.725 0.    0.009 0.011 0.05  0.02 ]]
The mean of the samples was 0.15
Hyperparameter convergence reached at iteration 2092.
The minimum predicted mean of the observed indices is -1.743 at the point
[0.394 0.    0.01  0.011 0.021 0.02 ]
Trained parameters:
amplitude_champ:0 is 0.735

length_scales_champ:0 is [0.351 0.7   0.009 0.006 0.016 0.017]

observation_noise_variance_champ:0 is 0.069
```

```
bias_mean:0 is 0.856

Iteration 301
Acquisition function convergence reached at iteration 2.
The final EI loss was -0.0 with predicted mean of [0.258]
The next parameters to simulate from are [[0.995 0.152 0.01  0.011 0.021 0.018]]
The mean of the samples was 0.355
Iteration 302
Acquisition function convergence reached at iteration 2.
The final EI loss was -0.0 with predicted mean of [0.877]
The next parameters to simulate from are [[0.363 0.    0.01  0.035 0.018 0.027]]
The mean of the samples was 0.942
Iteration 303
Acquisition function convergence reached at iteration 148.
The final EI loss was -0.032 with predicted mean of [-1.698]
The next parameters to simulate from are [[0.141 0.883 0.006 0.009 0.035 0.022]]
The mean of the samples was -1.612
Iteration 304
Acquisition function convergence reached at iteration 2.
The final EI loss was -0.0 with predicted mean of [-0.431]
The next parameters to simulate from are [[0.527 0.    0.01  0.011 0.043 0.02 ]]
The mean of the samples was -0.322
Iteration 305
Acquisition function convergence reached at iteration 142.
The final EI loss was -0.002 with predicted mean of [-0.506]
The next parameters to simulate from are [[0.203 0.    0.02  0.021 0.    0.026]]
The mean of the samples was -0.757
Iteration 306
Acquisition function convergence reached at iteration 89.
The final EI loss was -0.028 with predicted mean of [-1.663]
The next parameters to simulate from are [[0.113 0.979 0.005 0.009 0.034 0.023]]
The mean of the samples was -1.645
Iteration 307
Acquisition function convergence reached at iteration 201.
The final EI loss was -0.002 with predicted mean of [-0.393]
The next parameters to simulate from are [[0.307 0.    0.033 0.021 0.    0.067]]
The mean of the samples was -0.338
Iteration 308
Acquisition function convergence reached at iteration 98.
The final EI loss was -0.002 with predicted mean of [-0.273]
The next parameters to simulate from are [[0.88  0.    0.033 0.008 0.05  0.017]]
The mean of the samples was -0.682
```

```
Iteration 309
Acquisition function convergence reached at iteration 109.
The final EI loss was -0.031 with predicted mean of [-1.738]
The next parameters to simulate from are [[0.374 0.083 0.009 0.011 0.022 0.02 ]]
The mean of the samples was -1.891
Iteration 310
Acquisition function convergence reached at iteration 135.
The final EI loss was -0.03 with predicted mean of [-1.674]
The next parameters to simulate from are [[0.137 0.99  0.007 0.009 0.037 0.021]]
The mean of the samples was -1.412
Iteration 311
Acquisition function convergence reached at iteration 54.
The final EI loss was -0.008 with predicted mean of [-1.127]
The next parameters to simulate from are [[0.02  0.417 0.028 0.016 0.011 0.04 ]]
The mean of the samples was -1.206
Iteration 312
Acquisition function convergence reached at iteration 103.
The final EI loss was -0.029 with predicted mean of [-1.744]
The next parameters to simulate from are [[0.375 0.074 0.009 0.011 0.021 0.02 ]]
The mean of the samples was -1.635
Iteration 313
Acquisition function convergence reached at iteration 222.
The final EI loss was -0.004 with predicted mean of [-0.6]
The next parameters to simulate from are [[0.002 0.999 0.033 0.02  0.013 0.053]]
The mean of the samples was -0.841
Iteration 314
Acquisition function convergence reached at iteration 131.
The final EI loss was -0.022 with predicted mean of [-1.676]
The next parameters to simulate from are [[0.093 0.318 0.007 0.01  0.021 0.025]]
The mean of the samples was -1.638
Iteration 315
Acquisition function convergence reached at iteration 2.
The final EI loss was -0.0 with predicted mean of [0.083]
The next parameters to simulate from are [[0.004 0.    0.005 0.011 0.038 0.042]]
The mean of the samples was 0.099
Iteration 316
Acquisition function convergence reached at iteration 101.
The final EI loss was -0.025 with predicted mean of [-1.736]
The next parameters to simulate from are [[0.39  0.021 0.01  0.011 0.021 0.02 ]]
The mean of the samples was -1.719
Iteration 317
Acquisition function convergence reached at iteration 2.
The final EI loss was -0.0 with predicted mean of [1.222]
```

```
The next parameters to simulate from are [[0.39  0.021 0.01  0.043 0.021 0.039]]
The mean of the samples was 1.185
Iteration 318
Acquisition function convergence reached at iteration 73.
The final EI loss was -0.001 with predicted mean of [-0.093]
The next parameters to simulate from are [[0.003 0.993 0.017 0.012 0.001 0.066]]
The mean of the samples was 0.117
Iteration 319
Acquisition function convergence reached at iteration 2.
The final EI loss was -0.0 with predicted mean of [1.367]
The next parameters to simulate from are [[0.512 0.021 0.01  0.044 0.021 0.005]]
The mean of the samples was 1.387
Iteration 320
Acquisition function convergence reached at iteration 129.
The final EI loss was -0.023 with predicted mean of [-1.596]
The next parameters to simulate from are [[0.157 0.83  0.005 0.009 0.038 0.019]]
The mean of the samples was -1.552
Hyperparameter convergence reached at iteration 2143.
The minimum predicted mean of the observed indices is -1.733 at the point
[0.39  0.021 0.01  0.011 0.021 0.02 ]
Iteration 321
Acquisition function convergence reached at iteration 85.
The final EI loss was -0.003 with predicted mean of [-0.765]
The next parameters to simulate from are [[0.628 0.435 0.033 0.011 0.037 0.024]]
The mean of the samples was -0.85
Iteration 322
Acquisition function convergence reached at iteration 114.
The final EI loss was -0.001 with predicted mean of [-0.257]
The next parameters to simulate from are [[0.002 0.    0.018 0.004 0.05  0.067]]
The mean of the samples was -0.379
Iteration 323
Acquisition function convergence reached at iteration 121.
The final EI loss was -0.021 with predicted mean of [-1.692]
The next parameters to simulate from are [[0.191 0.172 0.008 0.011 0.02  0.024]]
The mean of the samples was -1.574
Iteration 324
Acquisition function convergence reached at iteration 2.
The final EI loss was -0.0 with predicted mean of [1.41]
The next parameters to simulate from are [[0.39  0.021 0.01  0.042 0.044 0.031]]
The mean of the samples was 1.521
Iteration 325
Acquisition function convergence reached at iteration 14.
The final EI loss was -0.0 with predicted mean of [0.285]
```

```
The next parameters to simulate from are [[0.96  0.12  0.033 0.025 0.026 0.041]]
The mean of the samples was 0.403
Iteration 326
Acquisition function convergence reached at iteration 128.
The final EI loss was -0.002 with predicted mean of [-0.325]
The next parameters to simulate from are [[0.796 0.    0.027 0.01  0.05  0.001]]
The mean of the samples was 0.219
Iteration 327
Acquisition function convergence reached at iteration 2.
The final EI loss was -0.0 with predicted mean of [-0.734]
The next parameters to simulate from are [[0.39  0.021 0.029 0.011 0.021 0.02 ]]
The mean of the samples was -0.734
Iteration 328
Acquisition function convergence reached at iteration 105.
The final EI loss was -0.0 with predicted mean of [0.221]
The next parameters to simulate from are [[0.335 0.001 0.033 0.004 0.    0.067]]
The mean of the samples was 0.373
Iteration 329
Acquisition function convergence reached at iteration 2.
The final EI loss was -0.0 with predicted mean of [0.043]
The next parameters to simulate from are [[0.89  0.021 0.006 0.011 0.021 0.02 ]]
The mean of the samples was -0.099
Iteration 330
Acquisition function convergence reached at iteration 220.
The final EI loss was -0.003 with predicted mean of [-0.862]
The next parameters to simulate from are [[0.001 0.999 0.02  0.017 0.023 0.039]]
The mean of the samples was -0.583
Iteration 331
Acquisition function convergence reached at iteration 88.
The final EI loss was -0.003 with predicted mean of [-1.036]
The next parameters to simulate from are [[0.651 0.001 0.024 0.014 0.016 0.017]]
The mean of the samples was -1.217
Iteration 332
Acquisition function convergence reached at iteration 2.
The final EI loss was -0.0 with predicted mean of [-0.032]
The next parameters to simulate from are [[0.39  0.821 0.01  0.011 0.021 0.054]]
The mean of the samples was -0.027
Iteration 333
Acquisition function convergence reached at iteration 2.
The final EI loss was -0.0 with predicted mean of [-0.36]
The next parameters to simulate from are [[0.39  0.021 0.01  0.011 0.023 0.053]]
The mean of the samples was -0.348
Iteration 334
```

```
Acquisition function convergence reached at iteration 2.
The final EI loss was -0.0 with predicted mean of [1.017]
The next parameters to simulate from are [[0.315 0.625 0.01  0.04  0.021 0.039]]
The mean of the samples was 0.968
Iteration 335
Acquisition function convergence reached at iteration 56.
The final EI loss was -0.003 with predicted mean of [-0.674]
The next parameters to simulate from are [[0.278 0.999 0.033 0.015 0.001 0.031]]
The mean of the samples was -0.755
Iteration 336
Acquisition function convergence reached at iteration 2.
The final EI loss was -0.0 with predicted mean of [1.112]
The next parameters to simulate from are [[0.39  0.021 0.002 0.044 0.021 0.037]]
The mean of the samples was 1.305
Iteration 337
Acquisition function convergence reached at iteration 232.
The final EI loss was -0.004 with predicted mean of [-0.589]
The next parameters to simulate from are [[0.533 0.995 0.033 0.014 0.05  0.03 ]]
The mean of the samples was -0.73
Iteration 338
Acquisition function convergence reached at iteration 2.
The final EI loss was -0.0 with predicted mean of [1.316]
The next parameters to simulate from are [[0.39  0.021 0.01  0.043 0.023 0.02 ]]
The mean of the samples was 1.294
Iteration 339
Acquisition function convergence reached at iteration 2.
The final EI loss was -0.0 with predicted mean of [1.29]
The next parameters to simulate from are [[0.149 0.021 0.017 0.039 0.035 0.001]]
The mean of the samples was 1.504
Iteration 340
Acquisition function convergence reached at iteration 139.
The final EI loss was -0.023 with predicted mean of [-1.735]
The next parameters to simulate from are [[0.395 0.    0.01  0.011 0.021 0.02 ]]
The mean of the samples was -1.755
Hyperparameter convergence reached at iteration 2087.
The minimum predicted mean of the observed indices is -1.737 at the point
[0.395 0.    0.01  0.011 0.021 0.02 ]
Iteration 341
Acquisition function convergence reached at iteration 2.
The final EI loss was -0.0 with predicted mean of [-0.157]
The next parameters to simulate from are [[0.395 0.303 0.009 0.011 0.021 0.056]]
The mean of the samples was -0.13
Iteration 342
```

```
Acquisition function convergence reached at iteration 130.
The final EI loss was -0.014 with predicted mean of [-1.659]
The next parameters to simulate from are [[0.155 0.142 0.008 0.011 0.02  0.026]]
The mean of the samples was -1.83
Iteration 343
Acquisition function convergence reached at iteration 2.
The final EI loss was -0.0 with predicted mean of [0.366]
The next parameters to simulate from are [[0.875 0.632 0.01  0.011 0.021 0.039]]
The mean of the samples was 0.332
Iteration 344
Acquisition function convergence reached at iteration 118.
The final EI loss was -0.022 with predicted mean of [-1.737]
The next parameters to simulate from are [[0.398 0.    0.01  0.011 0.021 0.02 ]]
The mean of the samples was -1.938
Iteration 345
Acquisition function convergence reached at iteration 2.
The final EI loss was -0.0 with predicted mean of [0.227]
The next parameters to simulate from are [[0.901 0.431 0.006 0.011 0.015 0.02 ]]
The mean of the samples was 0.248
Iteration 346
Acquisition function convergence reached at iteration 2.
The final EI loss was -0.0 with predicted mean of [-0.541]
The next parameters to simulate from are [[0.398 0.    0.01  0.011 0.021 0.044]]
The mean of the samples was -0.513
Iteration 347
Acquisition function convergence reached at iteration 2.
The final EI loss was -0.0 with predicted mean of [1.044]
The next parameters to simulate from are [[0.398 0.496 0.01  0.039 0.021 0.02 ]]
The mean of the samples was 1.026
Iteration 348
Acquisition function convergence reached at iteration 209.
The final EI loss was -0.003 with predicted mean of [-0.531]
The next parameters to simulate from are [[0.001 0.997 0.017 0.008 0.05  0.047]]
The mean of the samples was -0.813
Iteration 349
Acquisition function convergence reached at iteration 2.
The final EI loss was -0.0 with predicted mean of [0.326]
The next parameters to simulate from are [[0.57  0.    0.025 0.026 0.021 0.02 ]]
The mean of the samples was 0.276
Iteration 350
Acquisition function convergence reached at iteration 85.
The final EI loss was -0.002 with predicted mean of [-0.842]
The next parameters to simulate from are [[0.003 0.    0.033 0.016 0.005 0.042]]
```

```
The mean of the samples was -0.974
Trained parameters:
amplitude_champ:0 is 0.735

length_scales_champ:0 is [0.356 0.762 0.009 0.006 0.016 0.018]

observation_noise_variance_champ:0 is 0.071

bias_mean:0 is 0.846

Iteration 351
Acquisition function convergence reached at iteration 2.
The final EI loss was -0.0 with predicted mean of [-0.07]
The next parameters to simulate from are [[0.02  0.    0.01  0.004 0.021 0.02 ]]
The mean of the samples was -0.124
Iteration 352
Acquisition function convergence reached at iteration 161.
The final EI loss was -0.003 with predicted mean of [-0.447]
The next parameters to simulate from are [[0.    0.998 0.033 0.025 0.    0.052]]
The mean of the samples was -0.665
Iteration 353
Acquisition function convergence reached at iteration 2.
The final EI loss was -0.0 with predicted mean of [-0.264]
The next parameters to simulate from are [[0.398 0.    0.001 0.011 0.021 0.02 ]]
The mean of the samples was -0.102
Iteration 354
Acquisition function convergence reached at iteration 2.
The final EI loss was -0.0 with predicted mean of [-0.334]
The next parameters to simulate from are [[0.554 0.    0.006 0.016 0.021 0.02 ]]
The mean of the samples was -0.272
Iteration 355
Acquisition function convergence reached at iteration 2.
The final EI loss was -0.0 with predicted mean of [0.03]
The next parameters to simulate from are [[0.308 0.205 0.032 0.016 0.021 0.009]]
The mean of the samples was 0.025
Iteration 356
Acquisition function convergence reached at iteration 2.
The final EI loss was -0.0 with predicted mean of [1.241]
The next parameters to simulate from are [[0.398 0.    0.005 0.04  0.021 0.016]]
The mean of the samples was 1.248
Iteration 357
Acquisition function convergence reached at iteration 148.
The final EI loss was -0.003 with predicted mean of [-0.467]
```

```
The next parameters to simulate from are [[0.005 0.999 0.011 0.02  0.    0.03 ]]
The mean of the samples was -0.698
Iteration 358
Acquisition function convergence reached at iteration 2.
The final EI loss was -0.0 with predicted mean of [1.061]
The next parameters to simulate from are [[0.476 0.    0.025 0.04  0.019 0.02 ]]
The mean of the samples was 1.084
Iteration 359
Acquisition function convergence reached at iteration 2.
The final EI loss was -0.0 with predicted mean of [-0.149]
The next parameters to simulate from are [[0.453 0.    0.014 0.011 0.021 0.066]]
The mean of the samples was -0.163
Iteration 360
Acquisition function convergence reached at iteration 2.
The final EI loss was -0.0 with predicted mean of [1.213]
The next parameters to simulate from are [[0.398 0.245 0.001 0.044 0.01  0.02 ]]
The mean of the samples was 1.173
Hyperparameter convergence reached at iteration 1823.
The minimum predicted mean of the observed indices is -1.746 at the point
[0.398 0.    0.01  0.011 0.021 0.02 ]
Iteration 361
Acquisition function convergence reached at iteration 121.
The final EI loss was -0.001 with predicted mean of [0.177]
The next parameters to simulate from are [[0.499 0.998 0.    0.024 0.    0.067]]
The mean of the samples was 0.485
Iteration 362
Acquisition function convergence reached at iteration 2.
The final EI loss was -0.0 with predicted mean of [1.036]
The next parameters to simulate from are [[0.303 0.659 0.014 0.034 0.043 0.045]]
The mean of the samples was 1.008
Iteration 363
Acquisition function convergence reached at iteration 98.
The final EI loss was -0.01 with predicted mean of [-1.495]
The next parameters to simulate from are [[0.238 0.424 0.007 0.009 0.03  0.02 ]]
The mean of the samples was -1.682
Iteration 364
Acquisition function convergence reached at iteration 113.
The final EI loss was -0.0 with predicted mean of [0.338]
The next parameters to simulate from are [[0.    0.    0.033 0.05  0.05  0.067]]
The mean of the samples was 1.71
Iteration 365
Acquisition function convergence reached at iteration 2.
The final EI loss was -0.0 with predicted mean of [-0.843]
```

```
The next parameters to simulate from are [[0.587 0.42  0.01  0.011 0.021 0.02 ]]
The mean of the samples was -0.699
Iteration 366
Acquisition function convergence reached at iteration 56.
The final EI loss was -0.003 with predicted mean of [-1.095]
The next parameters to simulate from are [[0.455 0.445 0.033 0.016 0.019 0.031]]
The mean of the samples was -1.151
Iteration 367
Acquisition function convergence reached at iteration 93.
The final EI loss was -0.0 with predicted mean of [0.282]
The next parameters to simulate from are [[0.727 0.994 0.    0.    0.02  0.   ]]
The mean of the samples was 0.418
Iteration 368
Acquisition function convergence reached at iteration 2.
The final EI loss was -0.0 with predicted mean of [1.098]
The next parameters to simulate from are [[0.811 0.    0.01  0.041 0.021 0.02 ]]
The mean of the samples was 1.081
Iteration 369
Acquisition function convergence reached at iteration 23.
The final EI loss was -0.0 with predicted mean of [0.269]
The next parameters to simulate from are [[0.647 0.046 0.032 0.003 0.024 0.   ]]
The mean of the samples was 0.425
Iteration 370
Acquisition function convergence reached at iteration 217.
The final EI loss was -0.003 with predicted mean of [-0.689]
The next parameters to simulate from are [[0.    0.    0.029 0.023 0.    0.046]]
The mean of the samples was -0.787
Iteration 371
Acquisition function convergence reached at iteration 195.
The final EI loss was -0.001 with predicted mean of [-0.456]
The next parameters to simulate from are [[0.001 0.    0.033 0.013 0.    0.028]]
The mean of the samples was -0.525
Iteration 372
Acquisition function convergence reached at iteration 2.
The final EI loss was -0.0 with predicted mean of [0.618]
The next parameters to simulate from are [[0.398 0.129 0.01  0.023 0.038 0.02 ]]
The mean of the samples was 0.558
Iteration 373
Acquisition function convergence reached at iteration 2.
The final EI loss was -0.0 with predicted mean of [0.584]
The next parameters to simulate from are [[0.017 0.    0.01  0.023 0.033 0.033]]
The mean of the samples was 0.533
Iteration 374
```

```
Acquisition function convergence reached at iteration 2.
The final EI loss was -0.0 with predicted mean of [0.352]
The next parameters to simulate from are [[0.373 0.613 0.01  0.027 0.021 0.02 ]]
The mean of the samples was 0.388
Iteration 375
Acquisition function convergence reached at iteration 2.
The final EI loss was -0.0 with predicted mean of [0.214]
The next parameters to simulate from are [[0.398 0.    0.01  0.011 0.021 0.002]]
The mean of the samples was 0.206
Iteration 376
Acquisition function convergence reached at iteration 106.
The final EI loss was -0.021 with predicted mean of [-1.747]
The next parameters to simulate from are [[0.397 0.    0.01  0.011 0.021 0.02 ]]
The mean of the samples was -1.642
Iteration 377
Acquisition function convergence reached at iteration 100.
The final EI loss was -0.0 with predicted mean of [0.128]
The next parameters to simulate from are [[0.496 0.002 0.023 0.    0.05  0.067]]
The mean of the samples was 0.207
Iteration 378
Acquisition function convergence reached at iteration 2.
The final EI loss was -0.0 with predicted mean of [0.547]
The next parameters to simulate from are [[0.822 0.    0.01  0.021 0.04  0.02 ]]
The mean of the samples was 0.69
Iteration 379
Acquisition function convergence reached at iteration 2.
The final EI loss was -0.0 with predicted mean of [0.173]
The next parameters to simulate from are [[0.147 0.    0.001 0.011 0.021 0.045]]
The mean of the samples was 0.2
Iteration 380
Acquisition function convergence reached at iteration 178.
The final EI loss was -0.002 with predicted mean of [-0.775]
The next parameters to simulate from are [[0.001 0.995 0.033 0.015 0.023 0.056]]
The mean of the samples was -1.033
Hyperparameter convergence reached at iteration 2152.
The minimum predicted mean of the observed indices is -1.738 at the point
[0.398 0.    0.01  0.011 0.021 0.02 ]
Iteration 381
Acquisition function convergence reached at iteration 2.
The final EI loss was -0.0 with predicted mean of [0.071]
The next parameters to simulate from are [[0.406 0.413 0.01  0.003 0.022 0.019]]
The mean of the samples was -0.009
Iteration 382
```

```
Acquisition function convergence reached at iteration 2.
The final EI loss was -0.0 with predicted mean of [0.274]
The next parameters to simulate from are [[0.136 0.    0.014 0.011 0.021 0.006]]
The mean of the samples was 0.005
Iteration 383
Acquisition function convergence reached at iteration 93.
The final EI loss was -0.015 with predicted mean of [-1.576]
The next parameters to simulate from are [[0.071 0.998 0.005 0.009 0.034 0.024]]
The mean of the samples was -1.161
Iteration 384
Acquisition function convergence reached at iteration 71.
The final EI loss was -0.013 with predicted mean of [-1.587]
The next parameters to simulate from are [[0.18  0.686 0.006 0.009 0.033 0.022]]
The mean of the samples was -1.805
Iteration 385
Acquisition function convergence reached at iteration 70.
The final EI loss was -0.015 with predicted mean of [-1.581]
The next parameters to simulate from are [[0.11  0.631 0.009 0.012 0.018 0.027]]
The mean of the samples was -1.597
Iteration 386
Acquisition function convergence reached at iteration 2.
The final EI loss was -0.0 with predicted mean of [-0.464]
The next parameters to simulate from are [[0.396 0.39  0.006 0.011 0.033 0.031]]
The mean of the samples was -0.641
Iteration 387
Acquisition function convergence reached at iteration 93.
The final EI loss was -0.02 with predicted mean of [-1.739]
The next parameters to simulate from are [[0.401 0.    0.01  0.011 0.021 0.02 ]]
The mean of the samples was -1.775
Iteration 388
Acquisition function convergence reached at iteration 2.
The final EI loss was -0.0 with predicted mean of [-0.271]
The next parameters to simulate from are [[0.401 0.    0.01  0.011 0.032 0.054]]
The mean of the samples was -0.257
Iteration 389
Acquisition function convergence reached at iteration 2.
The final EI loss was -0.0 with predicted mean of [1.1]
The next parameters to simulate from are [[0.715 0.    0.01  0.036 0.033 0.02 ]]
The mean of the samples was 1.231
Iteration 390
Acquisition function convergence reached at iteration 2.
The final EI loss was -0.0 with predicted mean of [-0.523]
The next parameters to simulate from are [[0.401 0.    0.01  0.011 0.043 0.02 ]]
```

```
The mean of the samples was -0.625
Iteration 391
Acquisition function convergence reached at iteration 100.
The final EI loss was -0.019 with predicted mean of [-1.739]
The next parameters to simulate from are [[0.401 0.    0.01  0.011 0.022 0.02 ]]
The mean of the samples was -1.624
Iteration 392
Acquisition function convergence reached at iteration 2.
The final EI loss was -0.0 with predicted mean of [1.446]
The next parameters to simulate from are [[0.67  0.051 0.01  0.049 0.039 0.012]]
The mean of the samples was 1.609
Iteration 393
Acquisition function convergence reached at iteration 79.
The final EI loss was -0.015 with predicted mean of [-1.677]
The next parameters to simulate from are [[0.096 0.212 0.009 0.011 0.02  0.027]]
The mean of the samples was -1.593
Iteration 394
Acquisition function convergence reached at iteration 2.
The final EI loss was -0.0 with predicted mean of [-0.835]
The next parameters to simulate from are [[0.401 0.    0.01  0.011 0.036 0.02 ]]
The mean of the samples was -0.711
Iteration 395
Acquisition function convergence reached at iteration 2.
The final EI loss was -0.0 with predicted mean of [1.152]
The next parameters to simulate from are [[0.401 0.    0.029 0.043 0.014 0.02 ]]
The mean of the samples was 1.246
Iteration 396
Acquisition function convergence reached at iteration 2.
The final EI loss was -0.0 with predicted mean of [-0.016]
The next parameters to simulate from are [[0.401 0.999 0.01  0.011 0.021 0.051]]
The mean of the samples was 0.004
Iteration 397
Acquisition function convergence reached at iteration 56.
The final EI loss was -0.002 with predicted mean of [-1.067]
The next parameters to simulate from are [[0.693 0.    0.028 0.012 0.029 0.019]]
The mean of the samples was -1.239
Iteration 398
Acquisition function convergence reached at iteration 220.
The final EI loss was -0.003 with predicted mean of [-0.602]
The next parameters to simulate from are [[0.001 0.997 0.026 0.009 0.05  0.034]]
The mean of the samples was -0.608
Iteration 399
Acquisition function convergence reached at iteration 90.
```

```
The final EI loss was -0.016 with predicted mean of [-1.601]
The next parameters to simulate from are [[0.246 0.509 0.006 0.009 0.033 0.02 ]]
The mean of the samples was -1.747
Iteration 400
Acquisition function convergence reached at iteration 133.
The final EI loss was -0.001 with predicted mean of [0.193]
The next parameters to simulate from are [[0.999 0.998 0.033 0.04  0.    0.067]]
The mean of the samples was 0.435
Hyperparameter convergence reached at iteration 2077.
The minimum predicted mean of the observed indices is -1.736 at the point
[0.401 0.    0.01  0.011 0.021 0.02 ]
Trained parameters:
amplitude_champ:0 is 0.722

length_scales_champ:0 is [0.325 0.747 0.009 0.006 0.016 0.018]

observation_noise_variance_champ:0 is 0.071

bias_mean:0 is 0.856

Iteration 401
Acquisition function convergence reached at iteration 127.
The final EI loss was -0.002 with predicted mean of [-1.026]
The next parameters to simulate from are [[0.696 0.    0.02  0.014 0.017 0.008]]
The mean of the samples was -0.939
Iteration 402
Acquisition function convergence reached at iteration 2.
The final EI loss was -0.0 with predicted mean of [1.244]
The next parameters to simulate from are [[0.401 0.    0.01  0.041 0.021 0.021]]
The mean of the samples was 1.233
Iteration 403
Acquisition function convergence reached at iteration 190.
The final EI loss was -0.02 with predicted mean of [-1.736]
The next parameters to simulate from are [[0.402 0.001 0.01  0.011 0.022 0.02 ]]
The mean of the samples was -1.791
Iteration 404
Acquisition function convergence reached at iteration 97.
The final EI loss was -0.013 with predicted mean of [-1.673]
The next parameters to simulate from are [[0.152 0.132 0.009 0.011 0.02  0.026]]
The mean of the samples was -1.711
Iteration 405
Acquisition function convergence reached at iteration 83.
The final EI loss was -0.019 with predicted mean of [-1.657]
```

```
The next parameters to simulate from are [[0.215 0.661 0.006 0.009 0.033 0.021]]
The mean of the samples was -1.764
Iteration 406
Acquisition function convergence reached at iteration 71.
The final EI loss was -0.021 with predicted mean of [-1.664]
The next parameters to simulate from are [[0.243 0.569 0.006 0.01  0.033 0.02 ]]
The mean of the samples was -1.904
Iteration 407
Acquisition function convergence reached at iteration 89.
The final EI loss was -0.019 with predicted mean of [-1.738]
The next parameters to simulate from are [[0.403 0.    0.01  0.011 0.022 0.02 ]]
The mean of the samples was -1.612
Iteration 408
Acquisition function convergence reached at iteration 2.
The final EI loss was -0.0 with predicted mean of [-1.055]
The next parameters to simulate from are [[0.403 0.658 0.01  0.011 0.022 0.02 ]]
The mean of the samples was -1.064
Iteration 409
Acquisition function convergence reached at iteration 2.
The final EI loss was -0.0 with predicted mean of [0.13]
The next parameters to simulate from are [[0.539 0.828 0.008 0.011 0.022 0.046]]
The mean of the samples was 0.124
Iteration 410
Acquisition function convergence reached at iteration 78.
The final EI loss was -0.019 with predicted mean of [-1.734]
The next parameters to simulate from are [[0.404 0.    0.01  0.011 0.022 0.02 ]]
The mean of the samples was -1.756
Iteration 411
Acquisition function convergence reached at iteration 2.
The final EI loss was -0.0 with predicted mean of [0.272]
The next parameters to simulate from are [[0.404 0.    0.01  0.023 0.022 0.056]]
The mean of the samples was 0.214
Iteration 412
Acquisition function convergence reached at iteration 2.
The final EI loss was -0.0 with predicted mean of [0.896]
The next parameters to simulate from are [[0.694 0.    0.01  0.045 0.026 0.062]]
The mean of the samples was 1.043
Iteration 413
Acquisition function convergence reached at iteration 80.
The final EI loss was -0.018 with predicted mean of [-1.735]
The next parameters to simulate from are [[0.404 0.    0.01  0.011 0.022 0.02 ]]
The mean of the samples was -1.883
Iteration 414
```

```
Acquisition function convergence reached at iteration 93.
The final EI loss was -0.018 with predicted mean of [-1.738]
The next parameters to simulate from are [[0.403 0.    0.01  0.011 0.022 0.02 ]]
The mean of the samples was -1.611
Iteration 415
Acquisition function convergence reached at iteration 2.
The final EI loss was -0.0 with predicted mean of [-0.878]
The next parameters to simulate from are [[0.404 0.    0.026 0.011 0.022 0.02 ]]
The mean of the samples was -0.78
Iteration 416
Acquisition function convergence reached at iteration 137.
The final EI loss was -0.018 with predicted mean of [-1.734]
The next parameters to simulate from are [[0.401 0.    0.01  0.011 0.022 0.02 ]]
The mean of the samples was -1.843
Iteration 417
Acquisition function convergence reached at iteration 2.
The final EI loss was -0.0 with predicted mean of [0.506]
The next parameters to simulate from are [[0.494 0.    0.01  0.026 0.022 0.02 ]]
The mean of the samples was 0.47
Iteration 418
Acquisition function convergence reached at iteration 103.
The final EI loss was -0.0 with predicted mean of [0.262]
The next parameters to simulate from are [[0.001 0.    0.    0.038 0.    0.067]]
The mean of the samples was 0.55
Iteration 419
Acquisition function convergence reached at iteration 2.
The final EI loss was -0.0 with predicted mean of [0.489]
The next parameters to simulate from are [[0.41  0.748 0.01  0.035 0.021 0.065]]
The mean of the samples was 0.462
Iteration 420
Acquisition function convergence reached at iteration 2.
The final EI loss was -0.0 with predicted mean of [0.27]
The next parameters to simulate from are [[0.965 0.    0.01  0.004 0.022 0.02 ]]
The mean of the samples was 0.309
Hyperparameter convergence reached at iteration 2145.
The minimum predicted mean of the observed indices is -1.739 at the point
[0.404 0.    0.01  0.011 0.022 0.02 ]
Iteration 421
Acquisition function convergence reached at iteration 183.
The final EI loss was -0.001 with predicted mean of [-0.078]
The next parameters to simulate from are [[0.3   0.996 0.033 0.029 0.    0.067]]
The mean of the samples was -0.348
Iteration 422
```

```
Acquisition function convergence reached at iteration 2.
The final EI loss was -0.0 with predicted mean of [-0.078]
The next parameters to simulate from are [[0.404 0.    0.005 0.01  0.022 0.053]]
The mean of the samples was -0.245
Iteration 423
Acquisition function convergence reached at iteration 117.
The final EI loss was -0.001 with predicted mean of [0.215]
The next parameters to simulate from are [[0.994 1.    0.    0.038 0.    0.067]]
The mean of the samples was 0.505
Iteration 424
Acquisition function convergence reached at iteration 99.
The final EI loss was -0.018 with predicted mean of [-1.738]
The next parameters to simulate from are [[0.395 0.    0.01  0.011 0.021 0.02 ]]
The mean of the samples was -1.901
Iteration 425
Acquisition function convergence reached at iteration 78.
The final EI loss was -0.002 with predicted mean of [-1.366]
The next parameters to simulate from are [[0.609 0.001 0.023 0.012 0.025 0.017]]
The mean of the samples was -1.306
Iteration 426
Acquisition function convergence reached at iteration 71.
The final EI loss was -0.007 with predicted mean of [-1.453]
The next parameters to simulate from are [[0.11  0.664 0.012 0.013 0.016 0.029]]
The mean of the samples was -1.426
Iteration 427
Acquisition function convergence reached at iteration 2.
The final EI loss was -0.0 with predicted mean of [0.096]
The next parameters to simulate from are [[0.412 0.    0.01  0.002 0.035 0.006]]
The mean of the samples was 0.148
Iteration 428
Acquisition function convergence reached at iteration 240.
The final EI loss was -0.002 with predicted mean of [-0.739]
The next parameters to simulate from are [[0.002 0.999 0.021 0.016 0.    0.029]]
The mean of the samples was -0.783
Iteration 429
Acquisition function convergence reached at iteration 117.
The final EI loss was -0.001 with predicted mean of [-0.756]
The next parameters to simulate from are [[0.262 0.    0.024 0.011 0.05  0.032]]
The mean of the samples was -0.635
Iteration 430
Acquisition function convergence reached at iteration 2.
The final EI loss was -0.0 with predicted mean of [1.354]
The next parameters to simulate from are [[0.404 0.212 0.016 0.045 0.022 0.02 ]]
```

```
The mean of the samples was 1.295
Iteration 431
Acquisition function convergence reached at iteration 93.
The final EI loss was -0.017 with predicted mean of [-1.742]
The next parameters to simulate from are [[0.394 0.    0.01  0.011 0.021 0.02 ]]
The mean of the samples was -1.737
Iteration 432
Acquisition function convergence reached at iteration 177.
The final EI loss was -0.002 with predicted mean of [-0.624]
The next parameters to simulate from are [[0.321 0.362 0.033 0.011 0.05  0.048]]
The mean of the samples was -0.696
Iteration 433
Acquisition function convergence reached at iteration 257.
The final EI loss was -0.011 with predicted mean of [-1.367]
The next parameters to simulate from are [[0.001 1.    0.01  0.01  0.024 0.031]]
The mean of the samples was -1.496
Iteration 434
Acquisition function convergence reached at iteration 2.
The final EI loss was -0.0 with predicted mean of [-0.803]
The next parameters to simulate from are [[0.404 0.415 0.003 0.011 0.022 0.02 ]]
The mean of the samples was -0.957
Iteration 435
Acquisition function convergence reached at iteration 2.
The final EI loss was -0.0 with predicted mean of [0.55]
The next parameters to simulate from are [[0.404 0.058 0.014 0.028 0.022 0.02 ]]
The mean of the samples was 0.522
Iteration 436
Acquisition function convergence reached at iteration 2.
The final EI loss was -0.0 with predicted mean of [-0.752]
The next parameters to simulate from are [[0.404 0.    0.03  0.011 0.022 0.02 ]]
The mean of the samples was -0.752
Iteration 437
Acquisition function convergence reached at iteration 98.
The final EI loss was -0.029 with predicted mean of [-1.689]
The next parameters to simulate from are [[0.259 0.555 0.006 0.009 0.035 0.019]]
The mean of the samples was -1.624
Iteration 438
Acquisition function convergence reached at iteration 2.
The final EI loss was -0.0 with predicted mean of [-0.705]
The next parameters to simulate from are [[0.404 0.199 0.028 0.01  0.022 0.02 ]]
The mean of the samples was -0.697
Iteration 439
Acquisition function convergence reached at iteration 2.
```

```
The final EI loss was -0.0 with predicted mean of [0.808]
The next parameters to simulate from are [[0.613 0.    0.01  0.042 0.022 0.066]]
The mean of the samples was 0.882
Iteration 440
Acquisition function convergence reached at iteration 138.
The final EI loss was -0.017 with predicted mean of [-1.739]
The next parameters to simulate from are [[0.385 0.038 0.01  0.011 0.022 0.02 ]]
The mean of the samples was -1.744
Hyperparameter convergence reached at iteration 2152.
The minimum predicted mean of the observed indices is -1.741 at the point
[0.404 0.    0.01  0.011 0.022 0.02 ]
Iteration 441
Acquisition function convergence reached at iteration 2.
The final EI loss was -0.0 with predicted mean of [-0.684]
The next parameters to simulate from are [[0.638 0.    0.01  0.011 0.022 0.006]]
The mean of the samples was -0.808
Iteration 442
Acquisition function convergence reached at iteration 2.
The final EI loss was -0.0 with predicted mean of [1.464]
The next parameters to simulate from are [[0.404 0.    0.01  0.048 0.022 0.02 ]]
The mean of the samples was 1.473
Iteration 443
Acquisition function convergence reached at iteration 28.
The final EI loss was -0.0 with predicted mean of [0.32]
The next parameters to simulate from are [[0.901 0.    0.019 0.031 0.    0.022]]
The mean of the samples was 0.296
Iteration 444
Acquisition function convergence reached at iteration 2.
The final EI loss was -0.0 with predicted mean of [1.248]
The next parameters to simulate from are [[0.439 0.    0.009 0.044 0.015 0.02 ]]
The mean of the samples was 1.192
Iteration 445
Acquisition function convergence reached at iteration 167.
The final EI loss was -0.002 with predicted mean of [-0.599]
The next parameters to simulate from are [[0.001 1.    0.033 0.018 0.017 0.067]]
The mean of the samples was -0.796
Iteration 446
Acquisition function convergence reached at iteration 2.
The final EI loss was -0.0 with predicted mean of [-0.238]
The next parameters to simulate from are [[0.404 0.528 0.01  0.011 0.022 0.047]]
The mean of the samples was -0.227
Iteration 447
Acquisition function convergence reached at iteration 2.
```

```
The final EI loss was -0.0 with predicted mean of [-0.511]
The next parameters to simulate from are [[0.404 0.    0.01  0.011 0.006 0.02 ]]
The mean of the samples was -0.537
Iteration 448
Acquisition function convergence reached at iteration 2.
The final EI loss was -0.0 with predicted mean of [-0.759]
The next parameters to simulate from are [[0.404 0.075 0.01  0.011 0.009 0.02 ]]
The mean of the samples was -0.749
Iteration 449
Acquisition function convergence reached at iteration 161.
The final EI loss was -0.022 with predicted mean of [-1.668]
The next parameters to simulate from are [[0.244 0.43  0.007 0.009 0.029 0.021]]
The mean of the samples was -1.725
Iteration 450
Acquisition function convergence reached at iteration 219.
The final EI loss was -0.017 with predicted mean of [-1.741]
The next parameters to simulate from are [[0.396 0.004 0.01  0.011 0.021 0.02 ]]
The mean of the samples was -1.838
Trained parameters:
amplitude_champ:0 is 0.719

length_scales_champ:0 is [0.331 0.749 0.01  0.006 0.016 0.016]

observation_noise_variance_champ:0 is 0.07

bias_mean:0 is 0.862

Iteration 451
Acquisition function convergence reached at iteration 2.
The final EI loss was -0.0 with predicted mean of [1.051]
The next parameters to simulate from are [[0.396 0.82  0.01  0.046 0.002 0.02 ]]
The mean of the samples was 1.048
Iteration 452
Acquisition function convergence reached at iteration 2.
The final EI loss was -0.0 with predicted mean of [0.781]
The next parameters to simulate from are [[0.261 0.004 0.01  0.036 0.016 0.051]]
The mean of the samples was 0.751
Iteration 453
Acquisition function convergence reached at iteration 95.
The final EI loss was -0.021 with predicted mean of [-1.701]
The next parameters to simulate from are [[0.199 0.717 0.006 0.009 0.034 0.022]]
The mean of the samples was -1.651
Iteration 454
```

```
Acquisition function convergence reached at iteration 2.
The final EI loss was -0.0 with predicted mean of [0.988]
The next parameters to simulate from are [[0.396 0.004 0.003 0.032 0.031 0.05 ]]
The mean of the samples was 1.202
Iteration 455
Acquisition function convergence reached at iteration 64.
The final EI loss was -0.016 with predicted mean of [-1.69]
The next parameters to simulate from are [[0.201 0.692 0.006 0.009 0.033 0.022]]
The mean of the samples was -1.694
Iteration 456
Acquisition function convergence reached at iteration 2.
The final EI loss was -0.0 with predicted mean of [-0.41]
The next parameters to simulate from are [[0.828 0.004 0.014 0.011 0.021 0.02 ]]
The mean of the samples was -0.419
Iteration 457
Acquisition function convergence reached at iteration 2.
The final EI loss was -0.0 with predicted mean of [-0.422]
The next parameters to simulate from are [[0.386 0.003 0.003 0.011 0.011 0.019]]
The mean of the samples was -1.013
Iteration 458
Acquisition function convergence reached at iteration 2.
The final EI loss was -0.0 with predicted mean of [-0.365]
The next parameters to simulate from are [[0.396 0.004 0.01  0.011 0.021 0.052]]
The mean of the samples was -0.363
Iteration 459
Acquisition function convergence reached at iteration 2.
The final EI loss was -0.0 with predicted mean of [-0.084]
The next parameters to simulate from are [[0.396 0.126 0.01  0.011 0.015 0.059]]
The mean of the samples was -0.057
Iteration 460
Acquisition function convergence reached at iteration 2.
The final EI loss was -0.0 with predicted mean of [-0.059]
The next parameters to simulate from are [[0.447 0.004 0.01  0.011 0.007 0.037]]
The mean of the samples was -0.075
Hyperparameter convergence reached at iteration 2172.
The minimum predicted mean of the observed indices is -1.745 at the point
[0.396 0.004 0.01  0.011 0.021 0.02 ]
Iteration 461
Acquisition function convergence reached at iteration 87.
The final EI loss was -0.012 with predicted mean of [-1.593]
The next parameters to simulate from are [[0.002 0.318 0.009 0.011 0.019 0.029]]
The mean of the samples was -1.676
Iteration 462
```

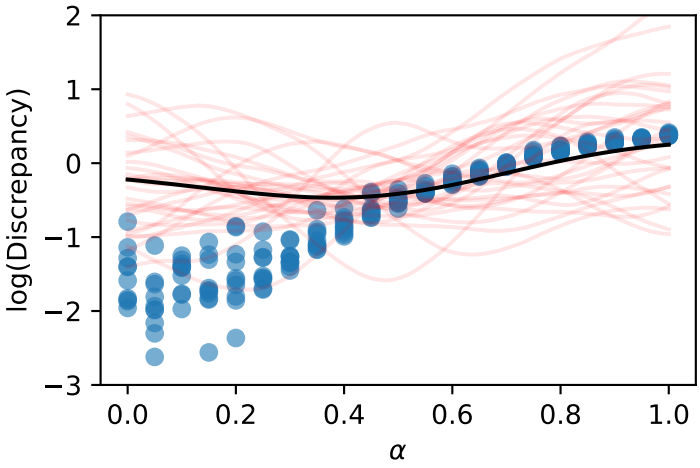
```
Acquisition function convergence reached at iteration 2.
The final EI loss was -0.0 with predicted mean of [-0.701]
The next parameters to simulate from are [[0.396 0.667 0.002 0.011 0.021 0.02 ]]
The mean of the samples was -0.588
Iteration 463
Acquisition function convergence reached at iteration 66.
The final EI loss was -0.002 with predicted mean of [-1.035]
The next parameters to simulate from are [[0.671 0.004 0.026 0.01  0.04  0.027]]
The mean of the samples was -0.975
Iteration 464
Acquisition function convergence reached at iteration 2.
The final EI loss was -0.0 with predicted mean of [0.341]
The next parameters to simulate from are [[0.085 0.004 0.01  0.011 0.021 0.001]]
The mean of the samples was 0.356
Iteration 465
Acquisition function convergence reached at iteration 2.
The final EI loss was -0.0 with predicted mean of [1.102]
The next parameters to simulate from are [[0.036 0.004 0.01  0.035 0.021 0.02 ]]
The mean of the samples was 1.122
Iteration 466
Acquisition function convergence reached at iteration 128.
The final EI loss was -0.016 with predicted mean of [-1.653]
The next parameters to simulate from are [[0.261 0.443 0.007 0.009 0.033 0.021]]
The mean of the samples was -1.622
Iteration 467
Acquisition function convergence reached at iteration 2.
The final EI loss was -0.0 with predicted mean of [-0.851]
The next parameters to simulate from are [[0.622 0.274 0.01  0.011 0.021 0.02 ]]
The mean of the samples was -0.748
Iteration 468
Acquisition function convergence reached at iteration 2.
The final EI loss was -0.0 with predicted mean of [0.166]
The next parameters to simulate from are [[0.94  0.697 0.01  0.011 0.033 0.02 ]]
The mean of the samples was 0.287
Iteration 469
Acquisition function convergence reached at iteration 108.
The final EI loss was -0.011 with predicted mean of [-1.675]
The next parameters to simulate from are [[0.061 0.289 0.009 0.011 0.02  0.028]]
The mean of the samples was -1.678
Iteration 470
Acquisition function convergence reached at iteration 69.
The final EI loss was -0.002 with predicted mean of [-0.778]
The next parameters to simulate from are [[0.406 1.    0.022 0.015 0.05  0.022]]
```

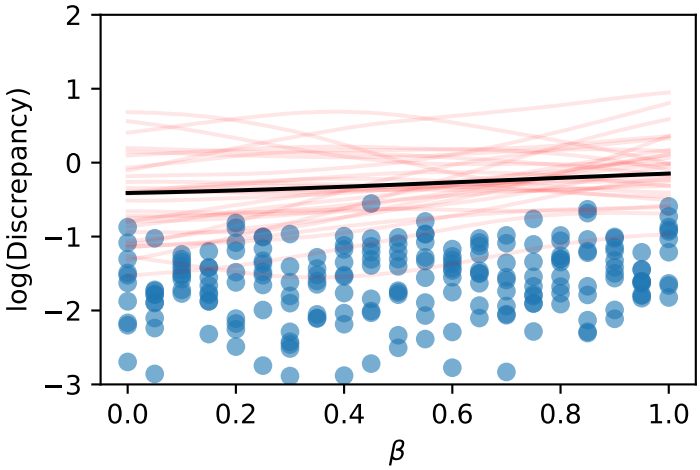
```
The mean of the samples was -0.864
Iteration 471
Acquisition function convergence reached at iteration 113.
The final EI loss was -0.001 with predicted mean of [0.231]
The next parameters to simulate from are [[1.    0.994 0.033 0.05  0.    0.035]]
The mean of the samples was 0.688
Iteration 472
Acquisition function convergence reached at iteration 2.
The final EI loss was -0.0 with predicted mean of [-0.436]
The next parameters to simulate from are [[0.396 0.004 0.01  0.011 0.02  0.048]]
The mean of the samples was -0.354
Iteration 473
Acquisition function convergence reached at iteration 156.
The final EI loss was -0.002 with predicted mean of [-0.706]
The next parameters to simulate from are [[0.219 0.    0.033 0.015 0.028 0.061]]
The mean of the samples was -0.722
Iteration 474
Acquisition function convergence reached at iteration 2.
The final EI loss was -0.0 with predicted mean of [-0.306]
The next parameters to simulate from are [[0.394 0.004 0.011 0.022 0.009 0.028]]
The mean of the samples was -0.306
Iteration 475
Acquisition function convergence reached at iteration 2.
The final EI loss was -0.0 with predicted mean of [0.422]
The next parameters to simulate from are [[0.396 0.657 0.003 0.026 0.024 0.02 ]]
The mean of the samples was 0.373
Iteration 476
Acquisition function convergence reached at iteration 59.
The final EI loss was -0.001 with predicted mean of [-0.75]
The next parameters to simulate from are [[0.002 0.001 0.016 0.012 0.028 0.053]]
The mean of the samples was -0.779
Iteration 477
Acquisition function convergence reached at iteration 91.
The final EI loss was -0.016 with predicted mean of [-1.745]
The next parameters to simulate from are [[0.397 0.    0.01  0.011 0.021 0.02 ]]
The mean of the samples was -1.842
Iteration 478
Acquisition function convergence reached at iteration 86.
The final EI loss was -0.012 with predicted mean of [-1.59]
The next parameters to simulate from are [[0.001 0.679 0.009 0.011 0.019 0.027]]
The mean of the samples was -1.593
Iteration 479
Acquisition function convergence reached at iteration 2.
```

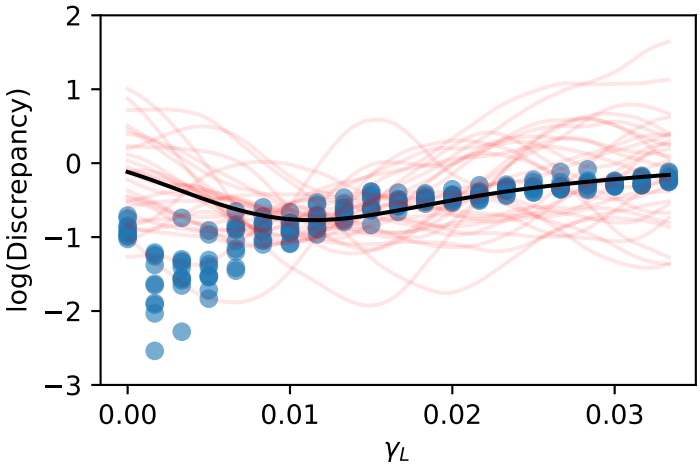
```
The final EI loss was -0.0 with predicted mean of [-1.143]
The next parameters to simulate from are [[0.404 0.    0.01  0.011 0.022 0.029]]
The mean of the samples was -0.972
Iteration 480
Acquisition function convergence reached at iteration 120.
The final EI loss was -0.01 with predicted mean of [-1.666]
The next parameters to simulate from are [[0.11  0.443 0.008 0.011 0.02  0.025]]
The mean of the samples was -1.742
Hyperparameter convergence reached at iteration 2159.
The minimum predicted mean of the observed indices is -1.743 at the point
[0.404 0.    0.01  0.011 0.022 0.02 ]
Iteration 481
Acquisition function convergence reached at iteration 90.
The final EI loss was -0.001 with predicted mean of [-0.626]
The next parameters to simulate from are [[0.224 0.    0.033 0.012 0.044 0.059]]
The mean of the samples was -0.578
Iteration 482
Acquisition function convergence reached at iteration 83.
The final EI loss was -0.03 with predicted mean of [-1.719]
The next parameters to simulate from are [[0.253 0.55  0.007 0.009 0.037 0.021]]
The mean of the samples was -1.704
Iteration 483
Acquisition function convergence reached at iteration 2.
The final EI loss was -0.0 with predicted mean of [0.978]
The next parameters to simulate from are [[0.404 0.    0.015 0.036 0.022 0.023]]
The mean of the samples was 0.97
Iteration 484
Acquisition function convergence reached at iteration 2.
The final EI loss was -0.0 with predicted mean of [-0.348]
The next parameters to simulate from are [[0.404 0.    0.01  0.011 0.05  0.02 ]]
The mean of the samples was -0.353
Iteration 485
Acquisition function convergence reached at iteration 2.
The final EI loss was -0.0 with predicted mean of [-0.476]
The next parameters to simulate from are [[0.404 0.763 0.01  0.008 0.022 0.02 ]]
The mean of the samples was -0.524
Iteration 486
Acquisition function convergence reached at iteration 79.
The final EI loss was -0.0 with predicted mean of [0.365]
The next parameters to simulate from are [[0.999 0.    0.    0.    0.025 0.066]]
The mean of the samples was 0.427
Iteration 487
Acquisition function convergence reached at iteration 2.
```

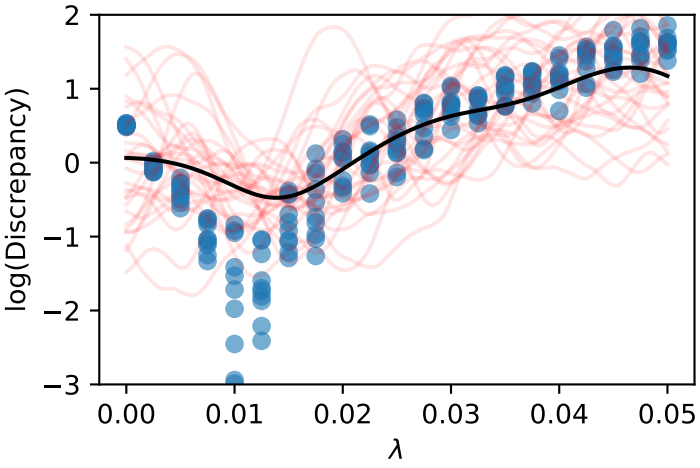
```
The final EI loss was -0.0 with predicted mean of [0.691]
The next parameters to simulate from are [[0.413 0.    0.027 0.026 0.04  0.02 ]]
The mean of the samples was 0.59
Iteration 488
Acquisition function convergence reached at iteration 2.
The final EI loss was -0.0 with predicted mean of [-1.135]
The next parameters to simulate from are [[0.404 0.    0.019 0.011 0.022 0.02 ]]
The mean of the samples was -1.161
Iteration 489
Acquisition function convergence reached at iteration 70.
The final EI loss was -0.0 with predicted mean of [0.244]
The next parameters to simulate from are [[0.387 0.    0.    0.    0.    0.026]]
The mean of the samples was 0.585
Iteration 490
Acquisition function convergence reached at iteration 95.
The final EI loss was -0.011 with predicted mean of [-1.681]
The next parameters to simulate from are [[0.129 0.145 0.008 0.011 0.02  0.026]]
The mean of the samples was -1.805
Iteration 491
Acquisition function convergence reached at iteration 2.
The final EI loss was -0.0 with predicted mean of [0.134]
The next parameters to simulate from are [[0.867 0.048 0.01  0.021 0.011 0.02 ]]
The mean of the samples was -0.014
Iteration 492
Acquisition function convergence reached at iteration 149.
The final EI loss was -0.001 with predicted mean of [-0.645]
The next parameters to simulate from are [[0.681 0.    0.033 0.008 0.05  0.03 ]]
The mean of the samples was -0.777
Iteration 493
Acquisition function convergence reached at iteration 64.
The final EI loss was -0.012 with predicted mean of [-1.687]
The next parameters to simulate from are [[0.146 0.129 0.008 0.011 0.021 0.026]]
The mean of the samples was -1.578
Iteration 494
Acquisition function convergence reached at iteration 2.
The final EI loss was -0.0 with predicted mean of [0.982]
The next parameters to simulate from are [[0.69  0.    0.01  0.039 0.022 0.032]]
The mean of the samples was 0.946
Iteration 495
Acquisition function convergence reached at iteration 102.
The final EI loss was -0.0 with predicted mean of [0.241]
The next parameters to simulate from are [[0.998 0.994 0.018 0.    0.    0.043]]
The mean of the samples was 0.593
```

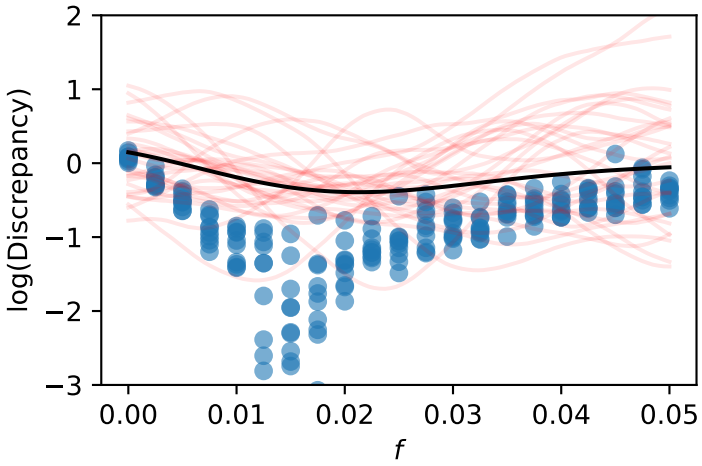
```
Iteration 496
Acquisition function convergence reached at iteration 2.
The final EI loss was -0.0 with predicted mean of [-0.678]
The next parameters to simulate from are [[0.404 0.028 0.032 0.011 0.022 0.02 ]]
The mean of the samples was -0.716
Iteration 497
Acquisition function convergence reached at iteration 2.
The final EI loss was -0.0 with predicted mean of [1.193]
The next parameters to simulate from are [[0.404 0.    0.022 0.036 0.048 0.02 ]]
The mean of the samples was 1.201
Iteration 498
Acquisition function convergence reached at iteration 58.
The final EI loss was -0.001 with predicted mean of [-0.941]
The next parameters to simulate from are [[0.524 0.    0.026 0.017 0.01  0.024]]
The mean of the samples was -1.055
Iteration 499
Acquisition function convergence reached at iteration 2.
The final EI loss was -0.0 with predicted mean of [-0.146]
The next parameters to simulate from are [[0.781 0.419 0.01  0.011 0.022 0.02 ]]
The mean of the samples was -0.137
Iteration 500
Acquisition function convergence reached at iteration 62.
The final EI loss was -0.001 with predicted mean of [-0.63]
The next parameters to simulate from are [[0.316 0.997 0.026 0.012 0.05  0.039]]
The mean of the samples was -0.679
Hyperparameter convergence reached at iteration 2185.
The minimum predicted mean of the observed indices is -1.743 at the point
[0.404 0.    0.01  0.011 0.022 0.02 ]
Trained parameters:
amplitude_champ:0 is 0.707

length_scales_champ:0 is [0.324 0.715 0.01  0.006 0.016 0.016]

observation_noise_variance_champ:0 is 0.07

bias_mean:0 is 0.879
```

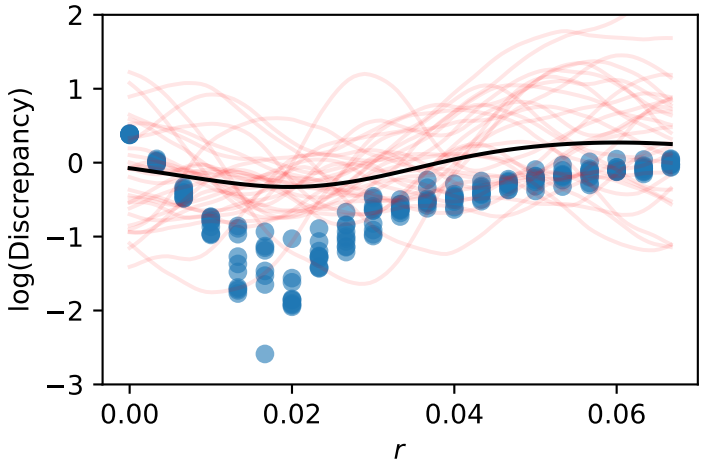

log(Discrepancy)
2
1
0
−1
−2
−3
0.0
0.2
0.4
0.6
0.8
1.0
α

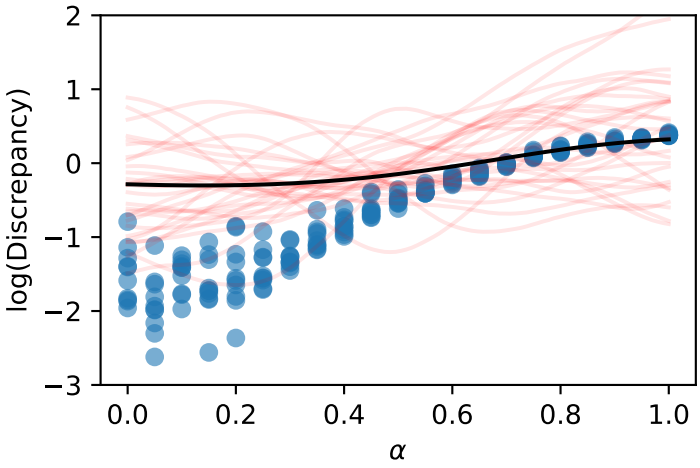

log(Discrepancy)
2
1
0
−1
−2
−3
0.0
0.2
0.4
0.6
0.8
1.0
β

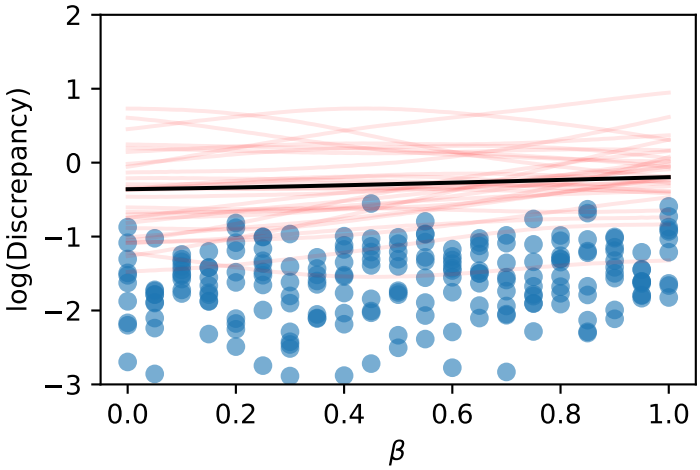

log(Discrepancy)
2
1
0
−1
−2
−3
0.00
0.01
0.02
0.03
γ_L

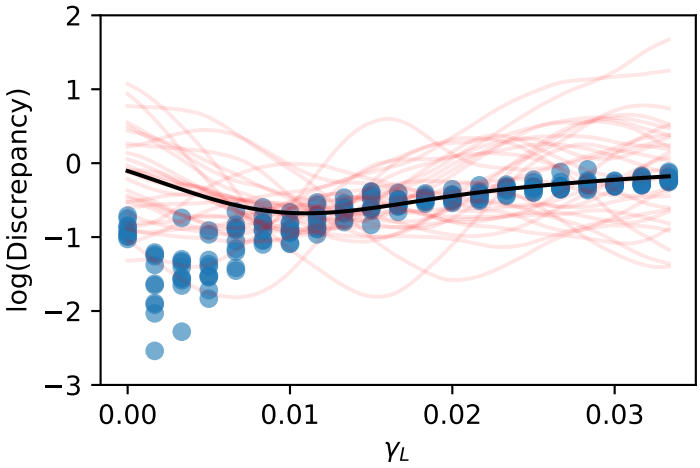

log(Discrepancy)
2
1
0
−1
−2
−3
0.00
0.01
0.02
0.03
0.04
0.05
λ

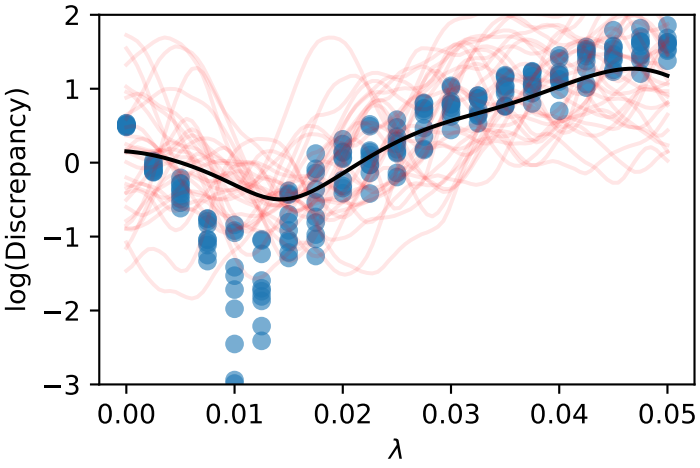

log(Discrepancy)
2
1
0
−1
−2
−3
0.00
0.01
0.02
0.03
0.04
0.05
f

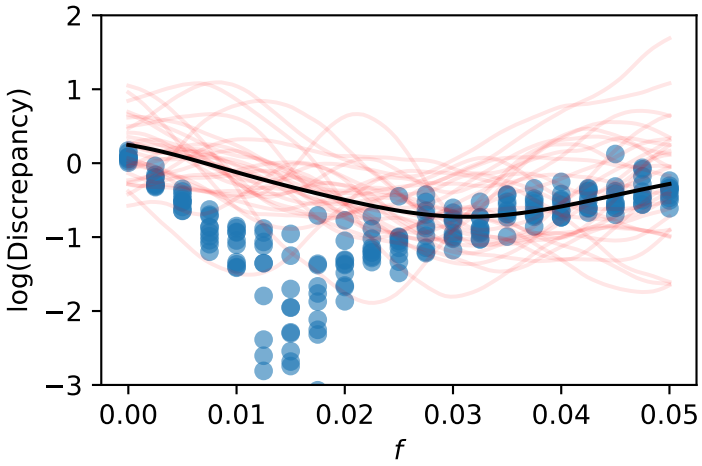

log(Discrepancy)
2
1
0
−1
−2
−3
0.00
0.02
0.04
0.06
r

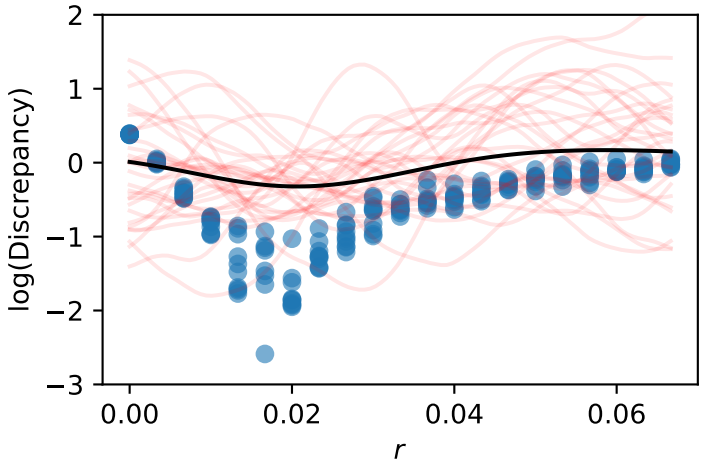

log(Discrepancy)
2
1
0
−1
−2
−3
0.0
0.2
0.4
0.6
0.8
1.0
α

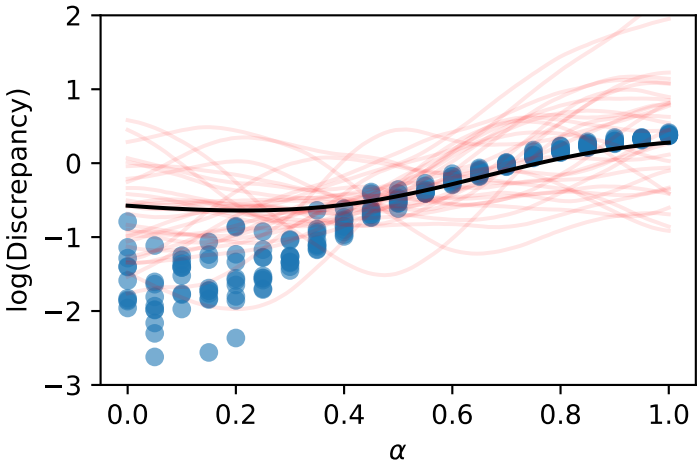

log(Discrepancy)
2
1
0
−1
−2
−3
0.0
0.2
0.4
0.6
0.8
1.0
β

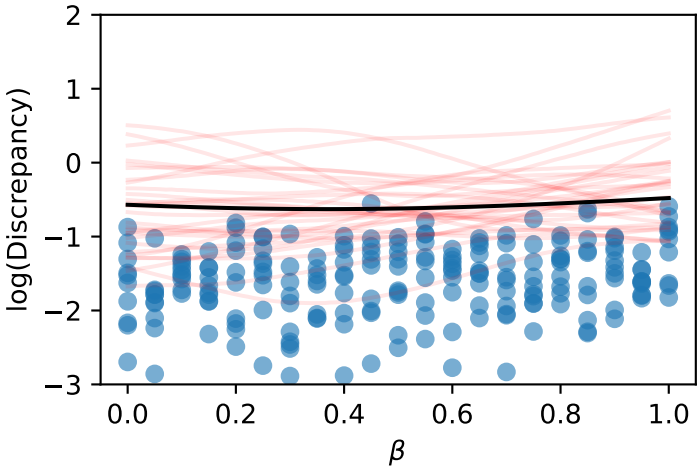

log(Discrepancy)
2
1
0
−1
−2
−3
0.00
0.01
0.02
0.03
γ_L

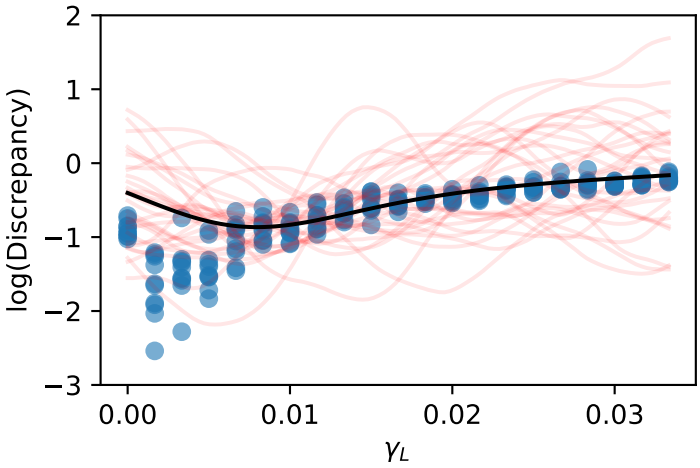

log(Discrepancy)
2
1
0
−1
−2
−3
0.00
0.01
0.02
0.03
0.04
0.05
λ

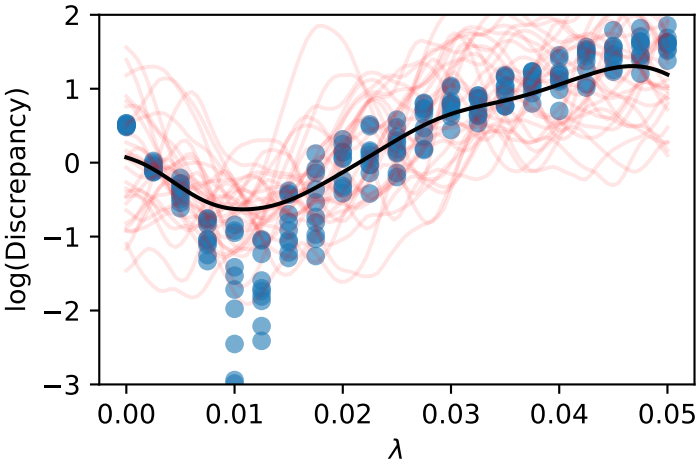

log(Discrepancy)
2
1
0
−1
−2
−3
0.00
0.01
0.02
0.03
0.04
0.05
f

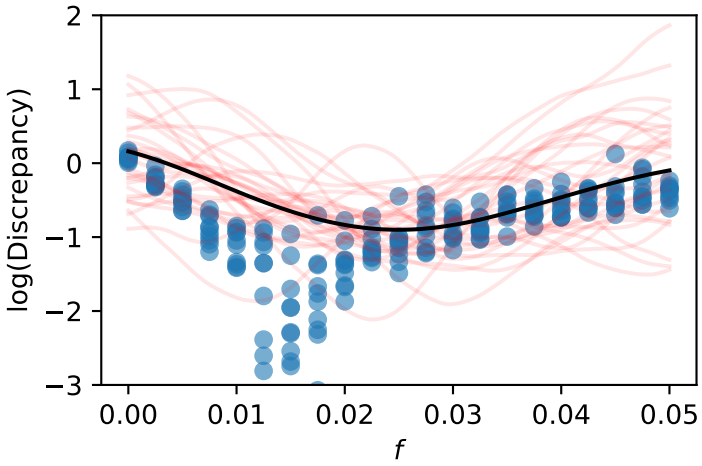

log(Discrepancy)
2
1
0
−1
−2
−3
0.00
0.02
0.04
0.06
r

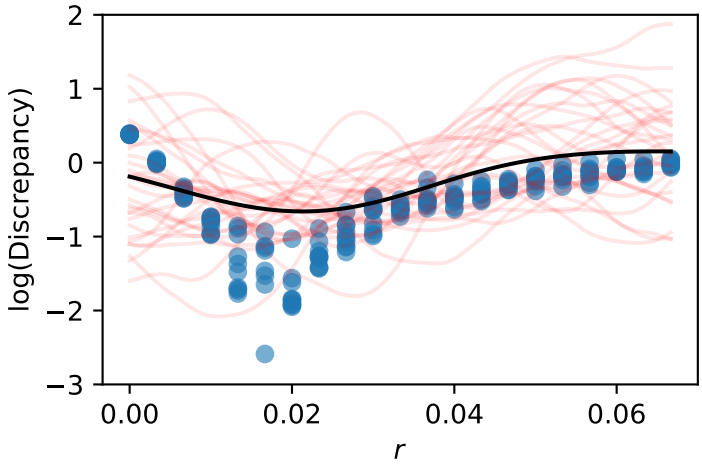

log(Discrepancy)
2
1
0
−1
−2
−3
0.0
0.2
0.4
0.6
0.8
1.0
α

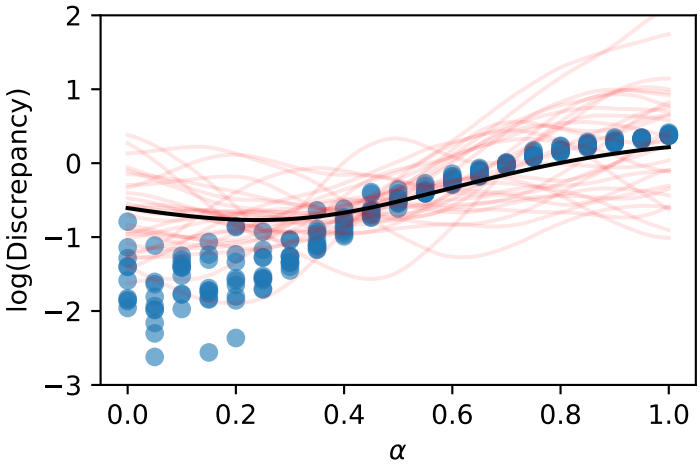

log(Discrepancy)
2
1
0
−1
−2
−3
0.0
0.2
0.4
0.6
0.8
1.0
β

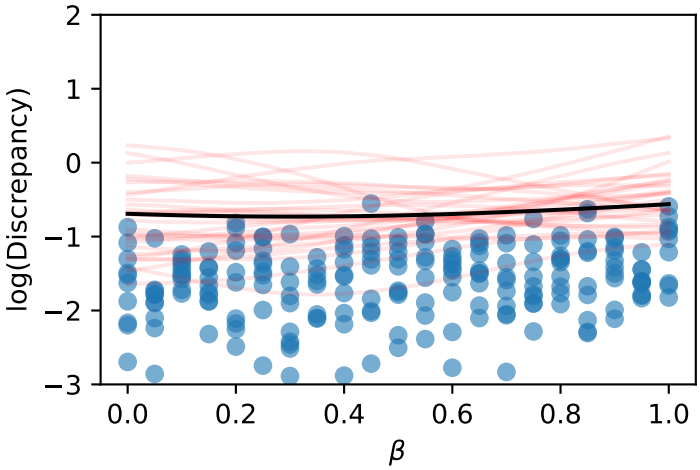

log(Discrepancy)
2
1
0
−1
−2
−3
0.00
0.01
0.02
0.03
γ_L

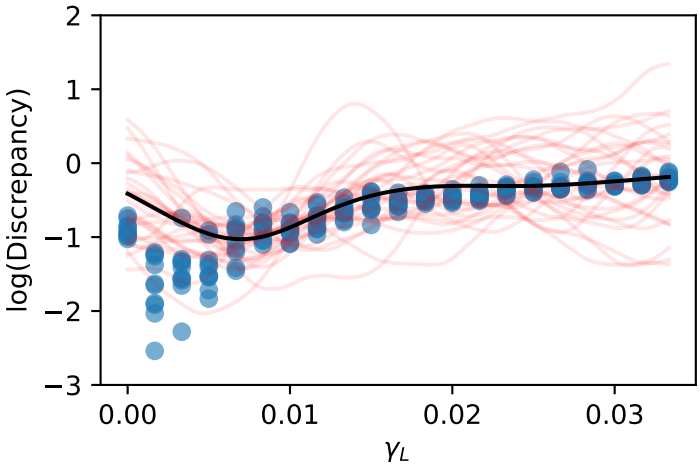

log(Discrepancy)
2
1
0
−1
−2
−3
0.00
0.01
0.02
0.03
0.04
0.05
λ

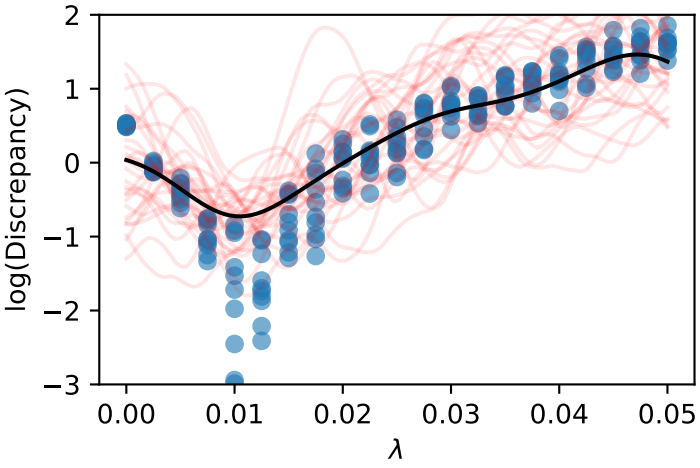

log(Discrepancy)
2
1
0
−1
−2
−3
0.00
0.01
0.02
0.03
0.04
0.05
f

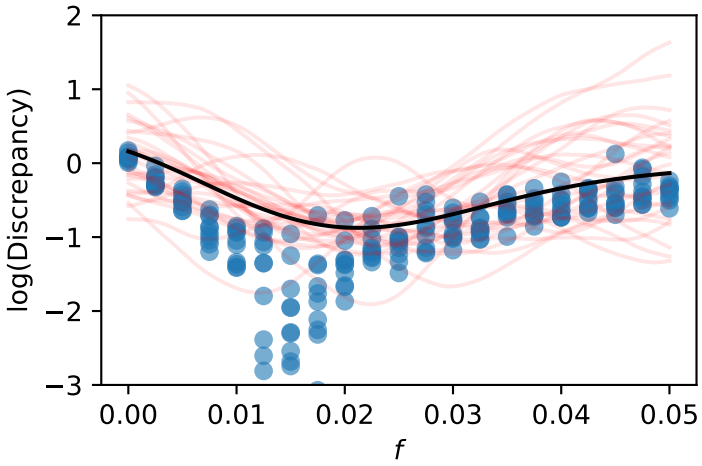

log(Discrepancy)
2
1
0
−1
−2
−3
0.00
0.02
0.04
0.06
r

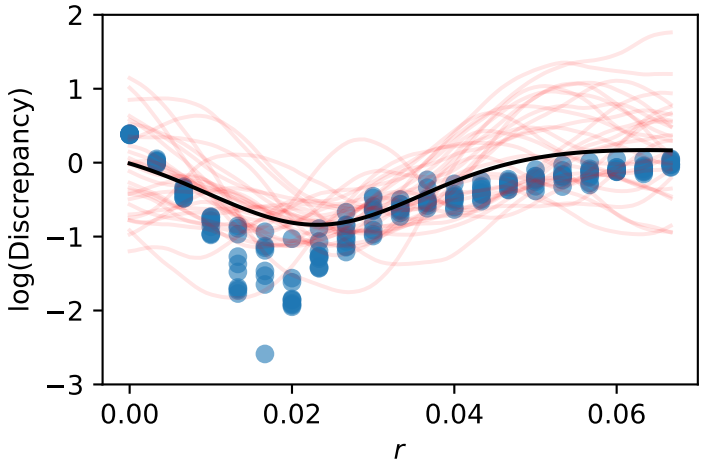

log(Discrepancy)
2
1
0
−1
−2
−3
0.0
0.2
0.4
0.6
0.8
1.0
α

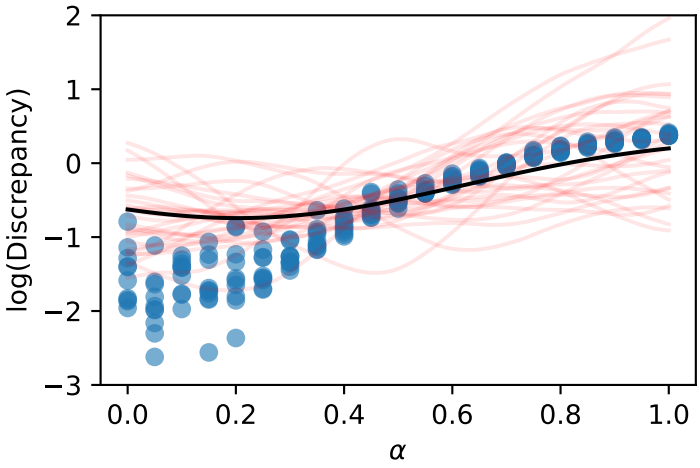

log(Discrepancy)
2
1
0
−1
−2
−3
0.0
0.2
0.4
0.6
0.8
1.0
β

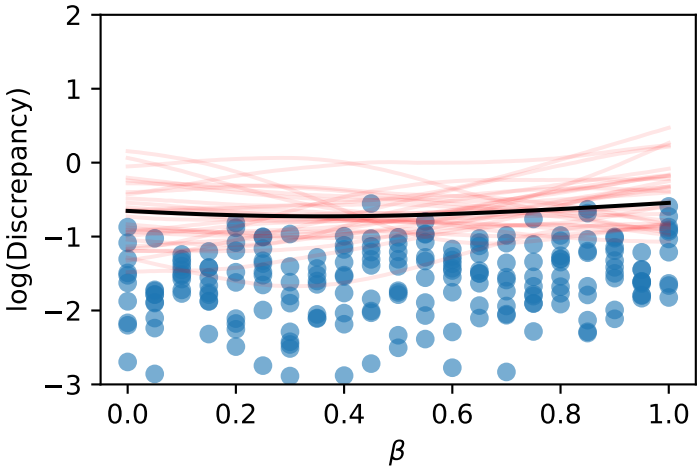

log(Discrepancy)
2
1
0
−1
−2
−3
0.00
0.01
0.02
0.03
γ_L

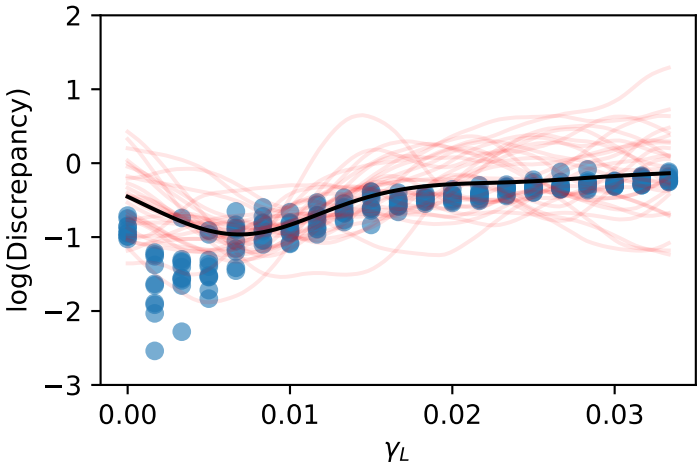

log(Discrepancy)
2
1
0
−1
−2
−3
0.00
0.01
0.02
0.03
0.04
0.05
λ

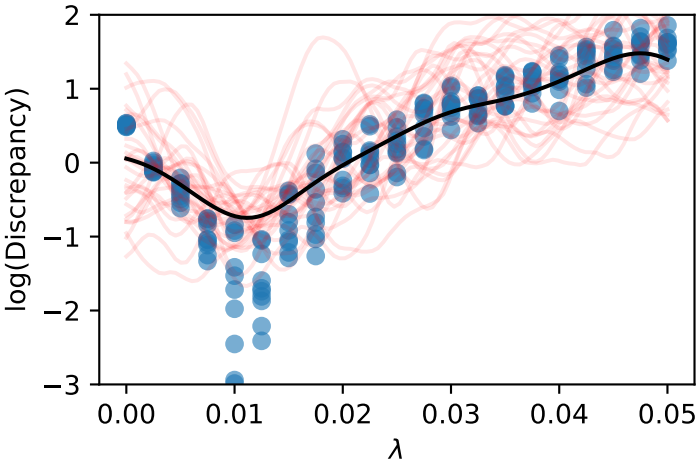

log(Discrepancy)
2
1
0
−1
−2
−3
0.00
0.01
0.02
0.03
0.04
0.05
f

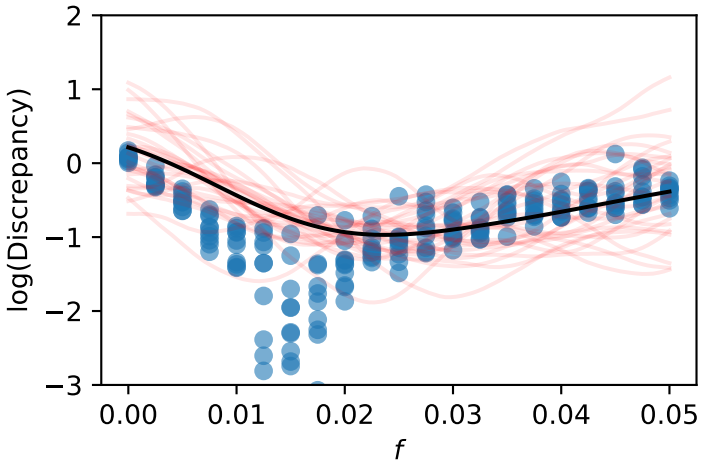

log(Discrepancy)
2
1
0
−1
−2
−3
0.00
0.02
0.04
0.06
r

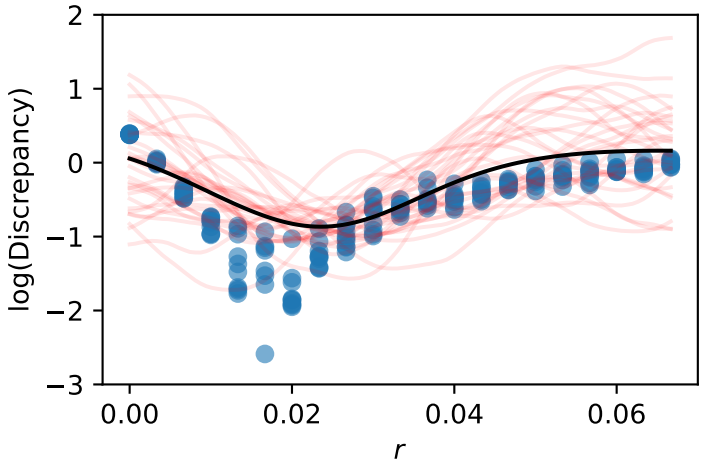

log(Discrepancy)
2
1
0
−1
−2
−3
0.0
0.2
0.4
0.6
0.8
1.0
α

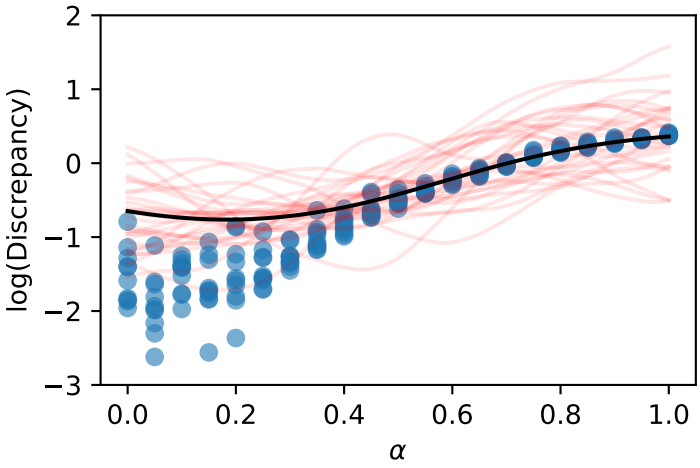

log(Discrepancy)
2
1
0
−1
−2
−3
0.0
0.2
0.4
0.6
0.8
1.0
β

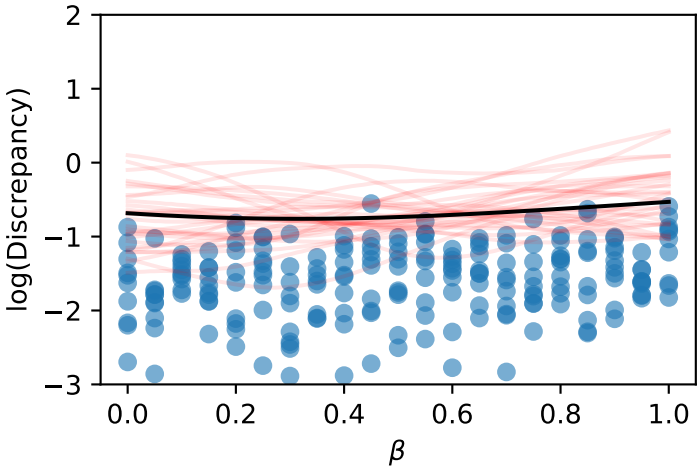

log(Discrepancy)
2
1
0
−1
−2
−3
0.00
0.01
0.02
0.03
γ_L

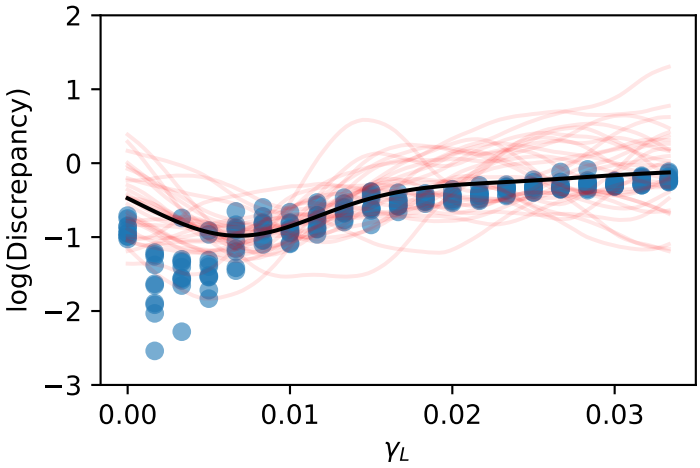

log(Discrepancy)
2
1
0
−1
−2
−3
0.00
0.01
0.02
0.03
0.04
0.05
λ

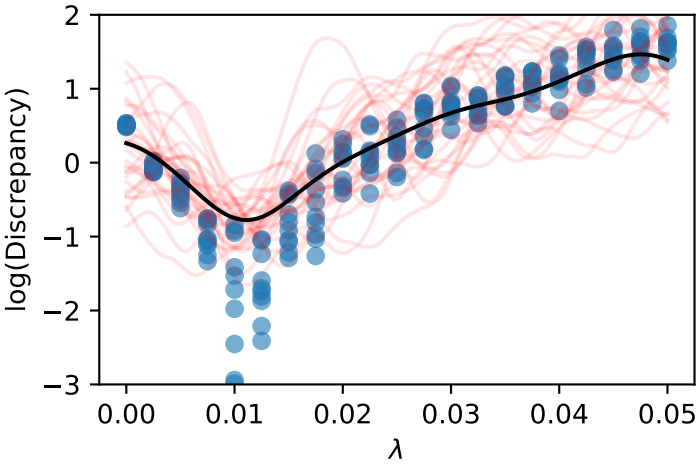

log(Discrepancy)
2
1
0
−1
−2
−3
0.00
0.01
0.02
0.03
0.04
0.05
f

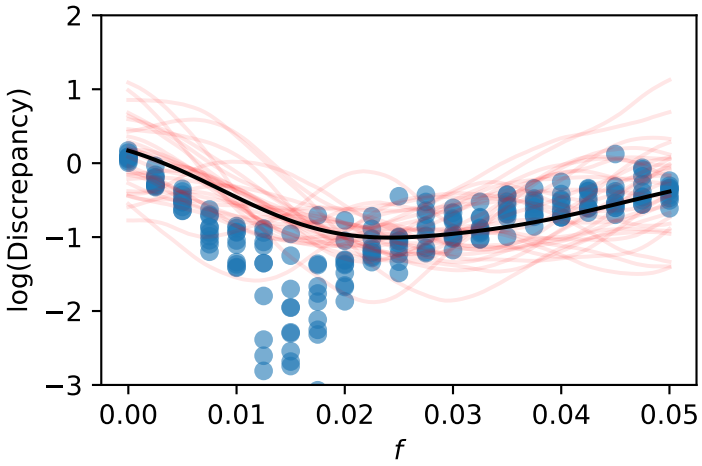

log(Discrepancy)
2
1
0
−1
−2
−3
0.00
0.02
0.04
0.06
r

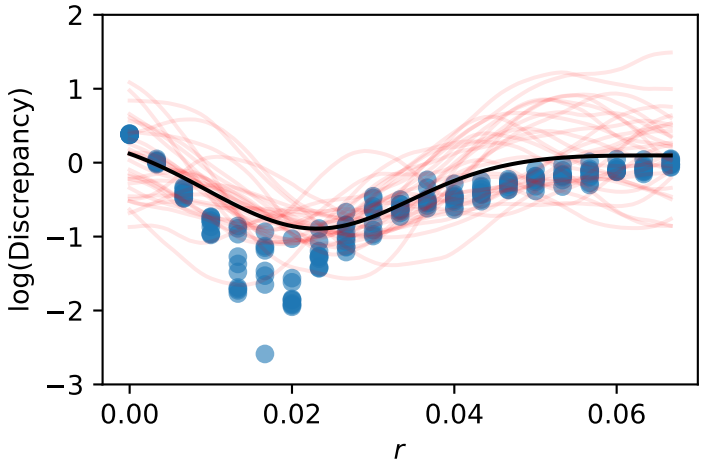

log(Discrepancy)
2
1
0
−1
−2
−3
0.0
0.2
0.4
0.6
0.8
1.0
α

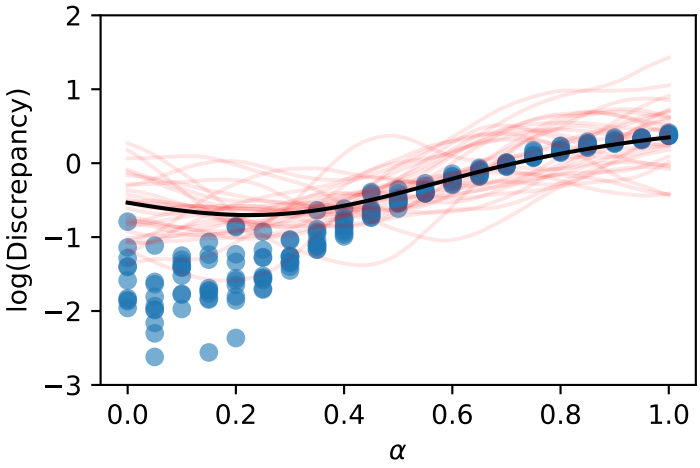

log(Discrepancy)
2
1
0
−1
−2
−3
0.0
0.2
0.4
0.6
0.8
1.0
β

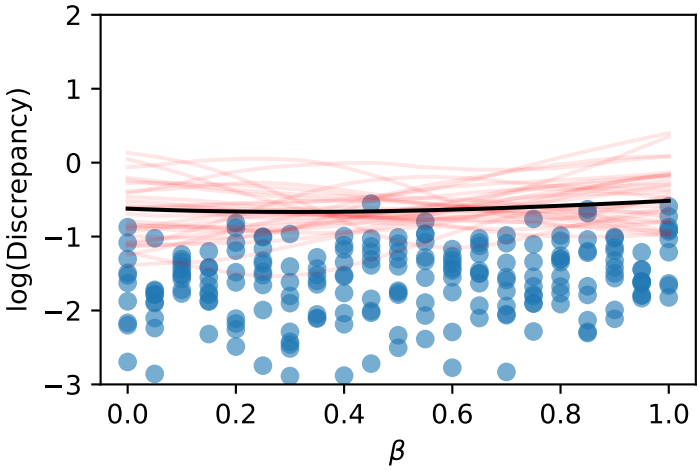

log(Discrepancy)
2
1
0
−1
−2
−3
0.00
0.01
0.02
0.03
γ_L

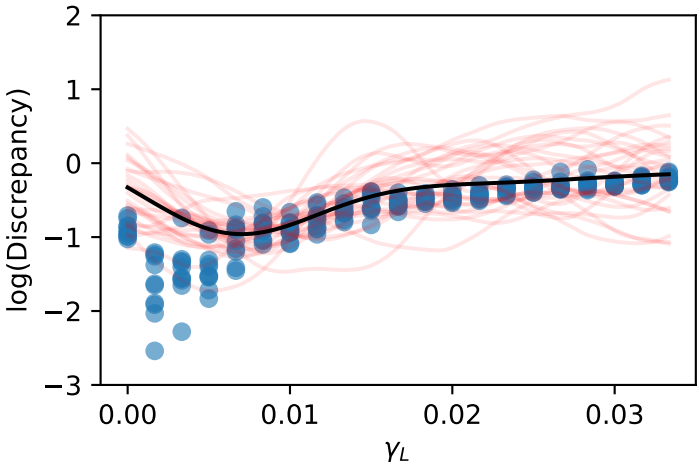

log(Discrepancy)
2
1
0
−1
−2
−3
0.00
0.01
0.02
0.03
0.04
0.05
λ

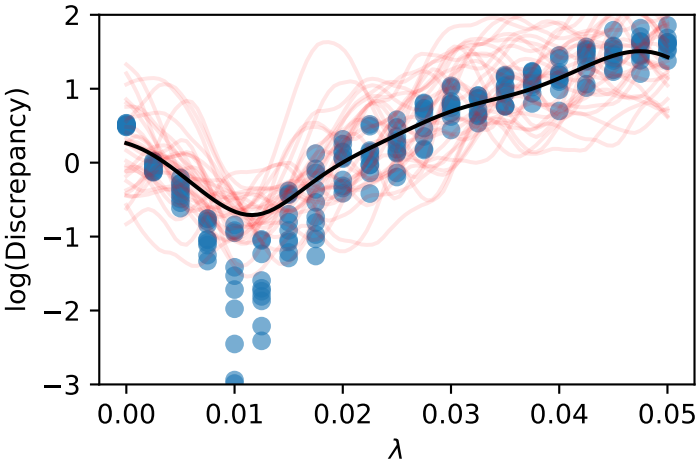

log(Discrepancy)
2
1
0
−1
−2
−3
0.00
0.01
0.02
0.03
0.04
0.05
f

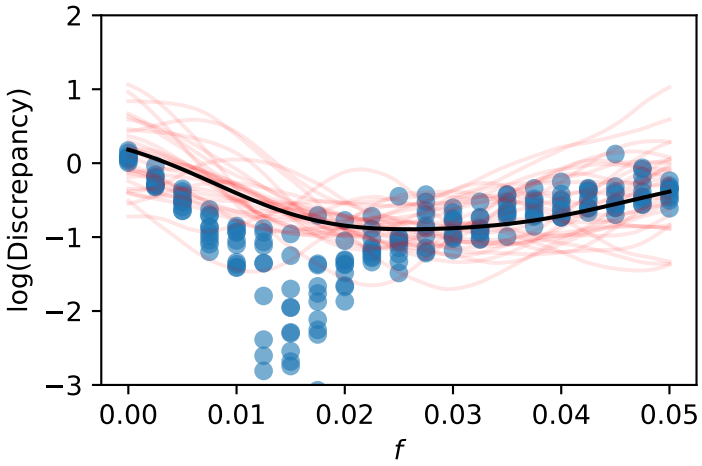

log(Discrepancy)
2
1
0
−1
−2
−3
0.00
0.02
0.04
0.06
r

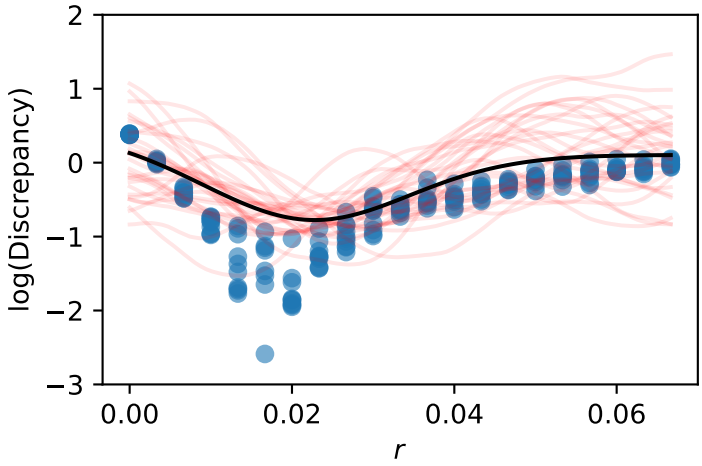

log(Discrepancy)
2
1
0
−1
−2
−3
0.0
0.2
0.4
0.6
0.8
1.0
α

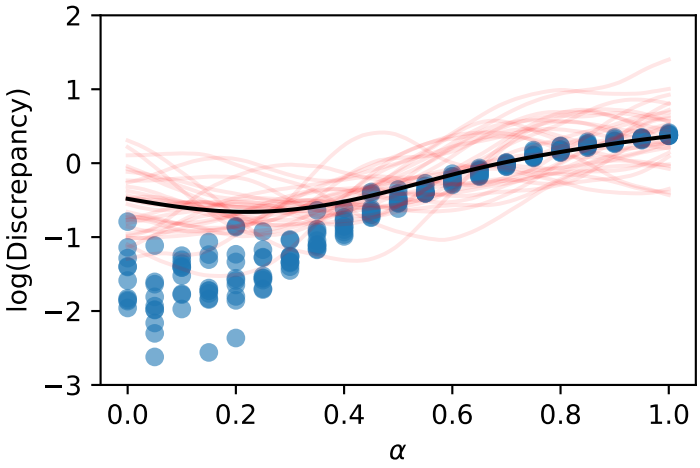

log(Discrepancy)
2
1
0
−1
−2
−3
0.0
0.2
0.4
0.6
0.8
1.0
β

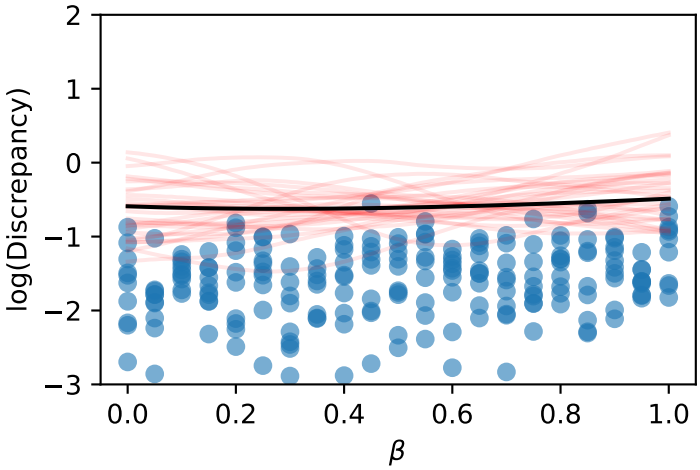

log(Discrepancy)
2
1
0
−1
−2
−3
0.00
0.01
0.02
0.03
γ_L

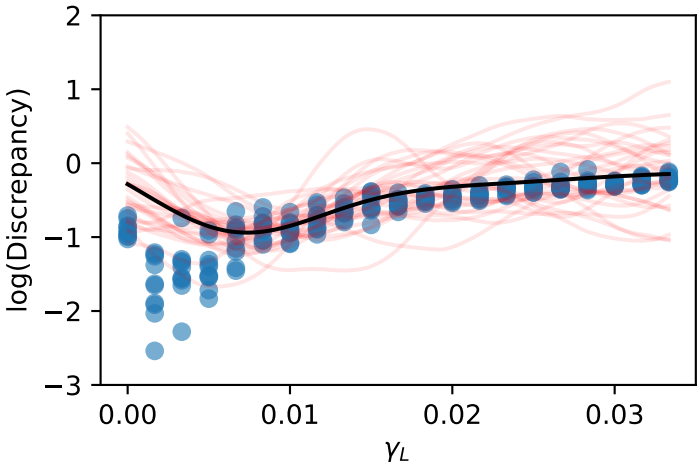

log(Discrepancy)
2
1
0
−1
−2
−3
0.00
0.01
0.02
0.03
0.04
0.05
λ

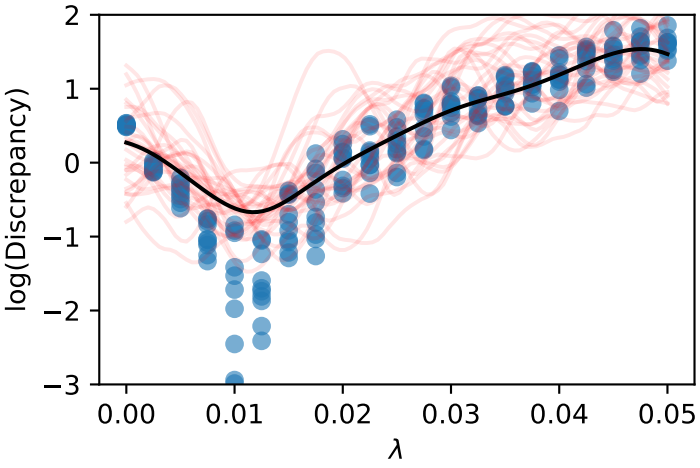

log(Discrepancy)
2
1
0
−1
−2
−3
0.00
0.01
0.02
0.03
0.04
0.05
f

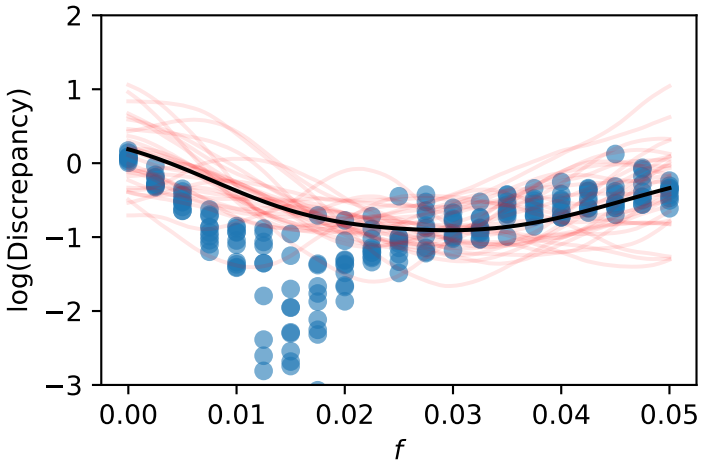

log(Discrepancy)
2
1
0
−1
−2
−3
0.00
0.02
0.04
0.06
r

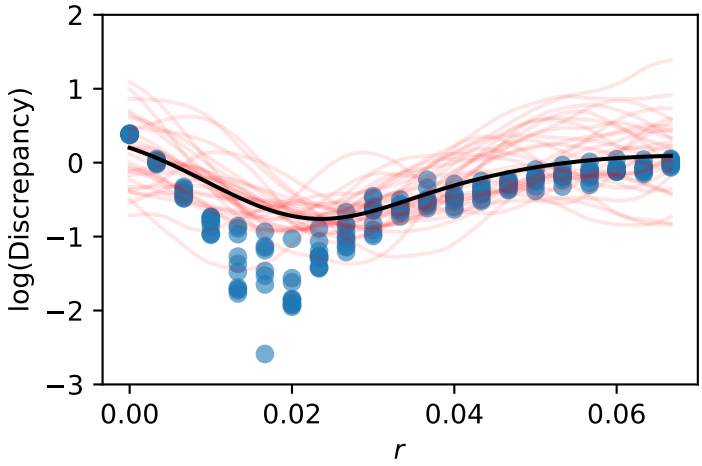

log(Discrepancy)
2
1
0
−1
−2
−3
0.0
0.2
0.4
0.6
0.8
1.0
α

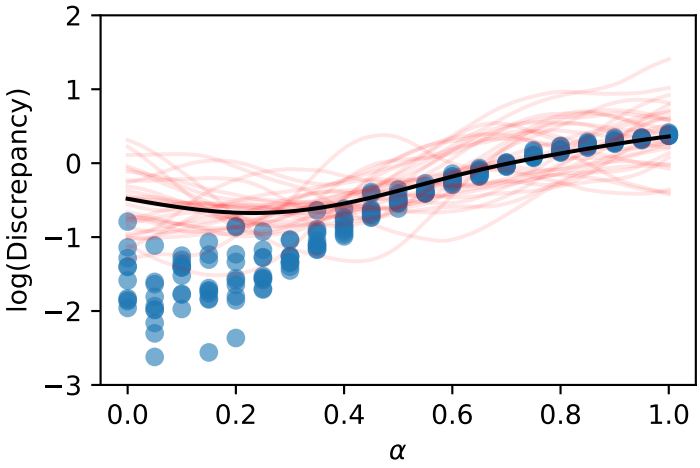

log(Discrepancy)
2
1
0
−1
−2
−3
0.0
0.2
0.4
0.6
0.8
1.0
β

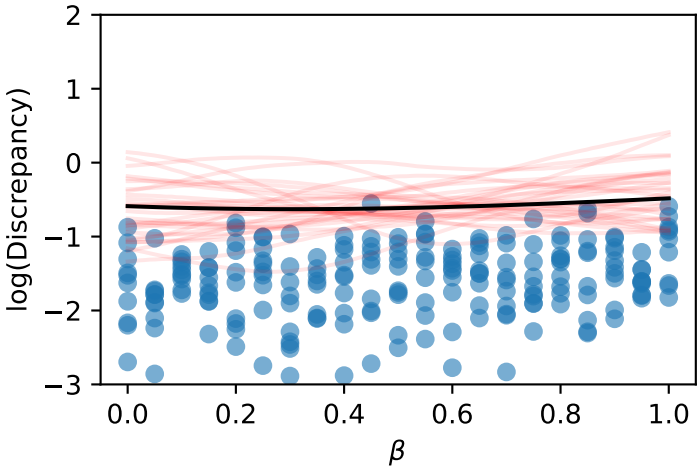

log(Discrepancy)
2
1
0
−1
−2
−3
0.00
0.01
0.02
0.03
γ_L

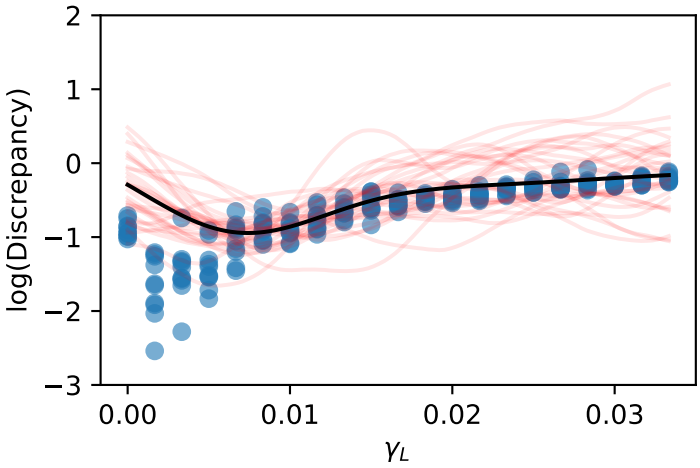

log(Discrepancy)
2
1
0
−1
−2
−3
0.00
0.01
0.02
0.03
0.04
0.05
λ

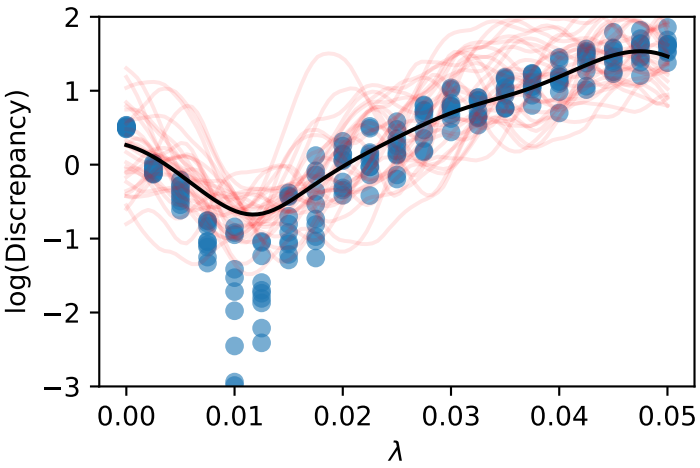

log(Discrepancy)
2
1
0
−1
−2
−3
0.00
0.01
0.02
0.03
0.04
0.05
f

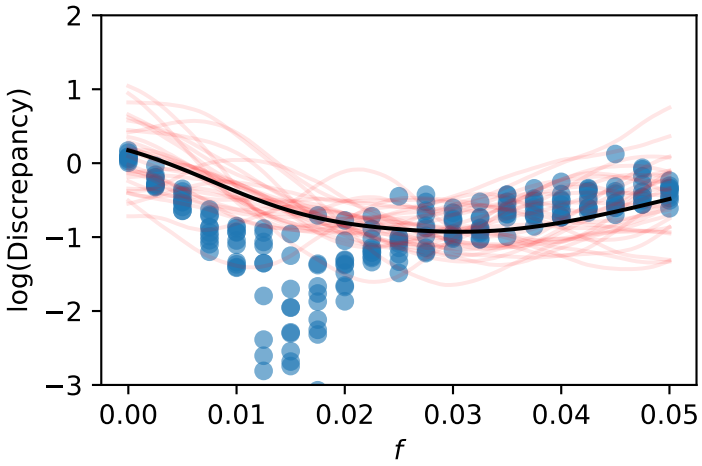

log(Discrepancy)
2
1
0
−1
−2
−3
0.00
0.02
0.04
0.06
r

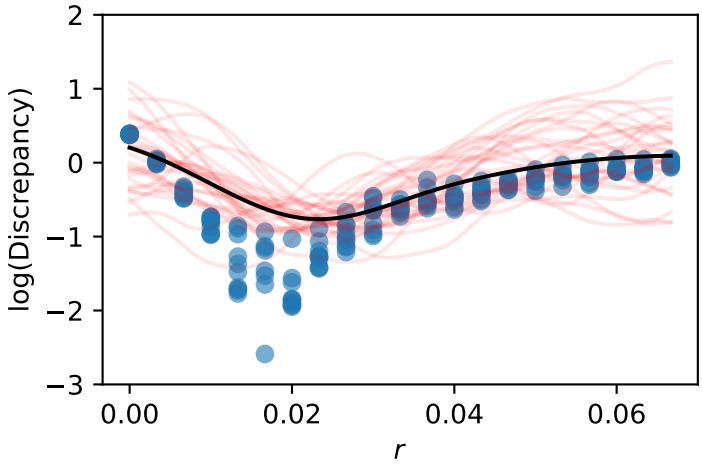

log(Discrepancy)
2
1
0
−1
−2
−3
0.0
0.2
0.4
0.6
0.8
1.0
α

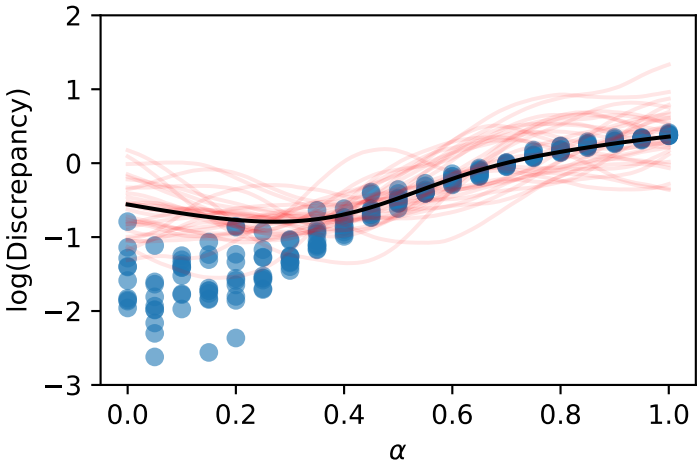

log(Discrepancy)
2
1
0
−1
−2
−3
0.0
0.2
0.4
0.6
0.8
1.0
β

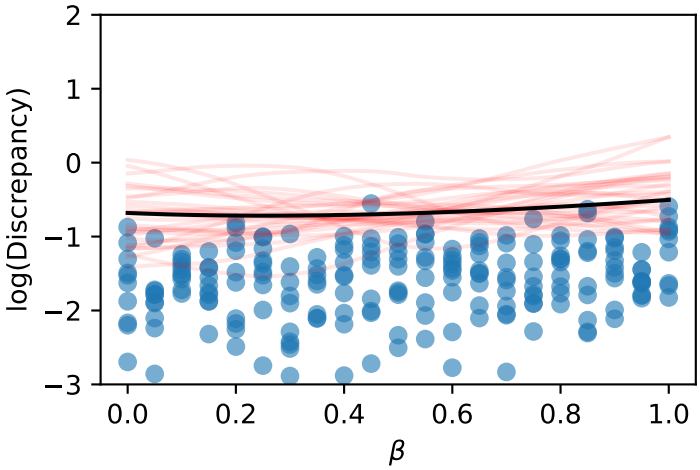

log(Discrepancy)
2
1
0
−1
−2
−3
0.00
0.01
0.02
0.03
γ_L

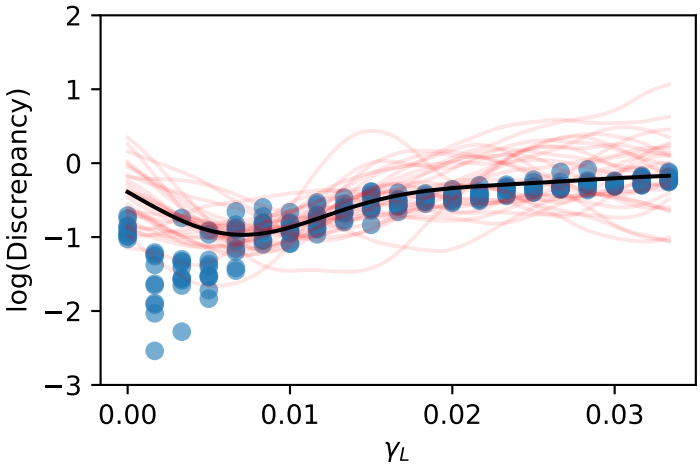

log(Discrepancy)
2
1
0
−1
−2
−3
0.00
0.01
0.02
0.03
0.04
0.05
λ

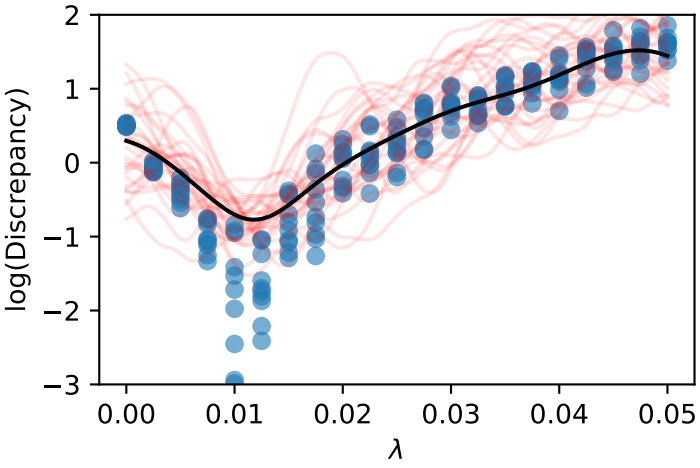

log(Discrepancy)
2
1
0
−1
−2
−3
0.00
0.01
0.02
0.03
0.04
0.05
f

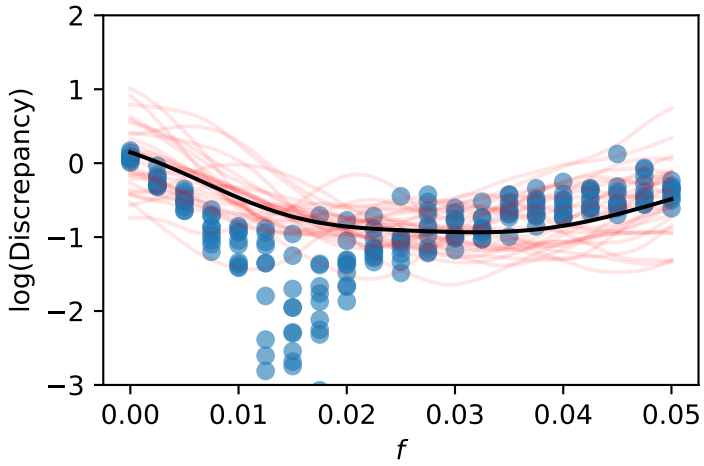

log(Discrepancy)
2
1
0
−1
−2
−3
0.00
0.02
0.04
0.06
r

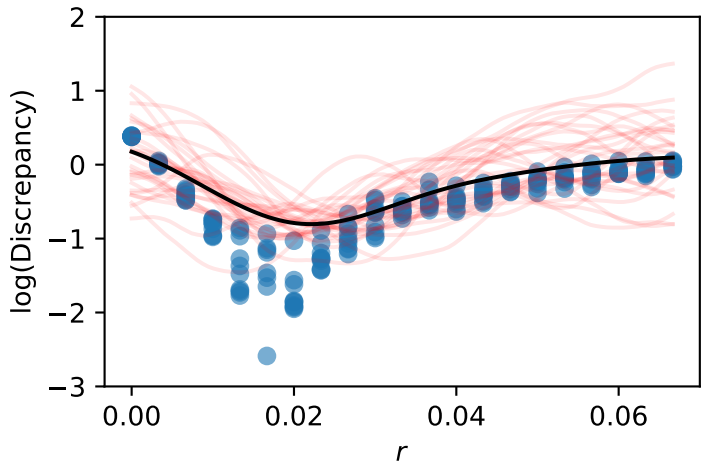

log(Discrepancy)
2
1
0
−1
−2
−3
0.0
0.2
0.4
0.6
0.8
1.0
α


log(Discrepancy)
2
1
0
−1
−2
−3
0.0
0.2
0.4
0.6
0.8
1.0
β


log(Discrepancy)
2
1
0
−1
−2
−3
0.00
0.01
0.02
0.03
γ_L

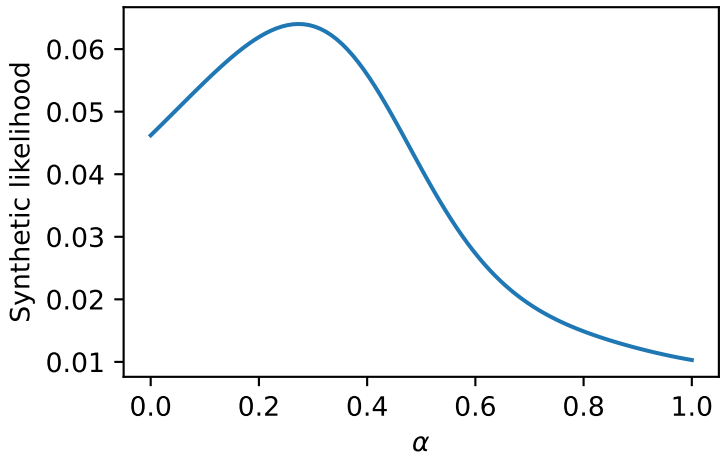

log(Discrepancy)
2
1
0
−1
−2
−3
0.00
0.01
0.02
0.03
0.04
0.05
λ

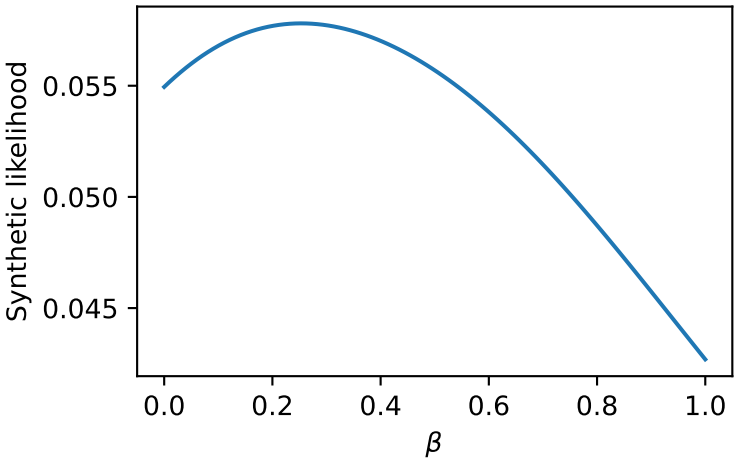


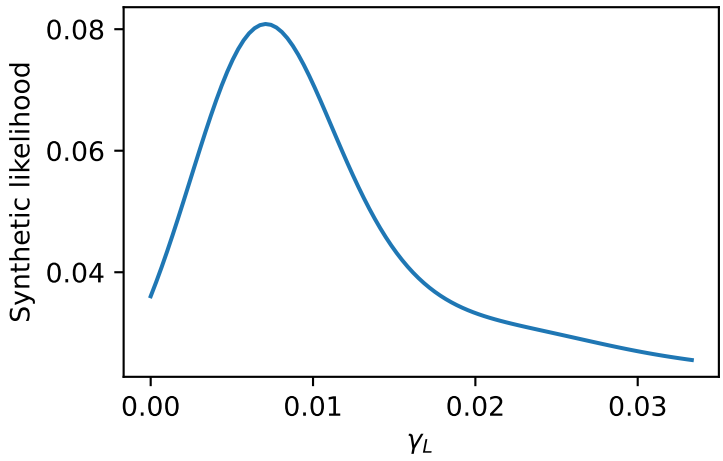


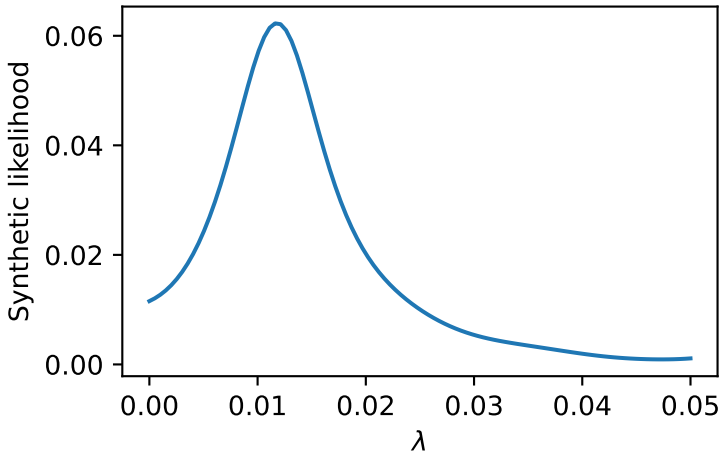

```
epsilon = -3

likelihood_dict = {}
for var in vars:
    champ_GP_reg = tfd.GaussianProcessRegressionModel(
        kernel=kernel_champ,
        index_points=slice_indices_dfs_dict[var + "_gp_indices_df"].values,
        observation_index_points=index_vals,
        observations=obs_vals,
        observation_noise_variance=observation_noise_variance_champ,
        predictive_noise_variance=0.0,
```

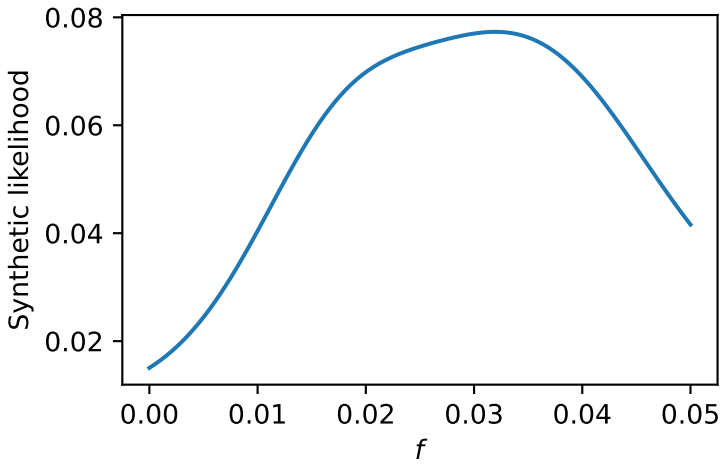
```
    mean_fn=const_mean_fn(),
)

indices_for_lik = slice_indices_dfs_dict[var + "_gp_indices_df"].values

mean = champ_GP_reg.mean_fn(indices_for_lik)
likelihood_dict[var + "_slice_means"] = mean
variance = dis_mean_n * observation_noise_variance_champ.numpy()
post_std = np.sqrt(variance)
log_cdf_vals = tfd.Normal(mean, post_std).log_cdf(epsilon)
likelihood_dict[var + "_synth_log_lik"] = log_cdf_vals

plt.figure(figsize=(4, 2.5))
plt.plot(
    slice_indices_dfs_dict[var + "_gp_indices_df"][var].values,
    np.exp(log_cdf_vals),
)
if var in ["f", "r"]:
    plt.xlabel("$" + var + "$")
    # plt.title("Final Synthetic Likelihood for $" + var + "$ Slice")
else:
    plt.xlabel("$\\" + var + "$")
    # plt.title("Final Synthetic Likelihood for $\\" + var + "$ Slice")
plt.ylabel("Synthetic likelihood")
plt.savefig(
    "champagne_GP_images/" + var + "_slice_" + str(t) + "_synth_likelihood.pdf",
    bbox_inches="tight",
)
plt.show()
```

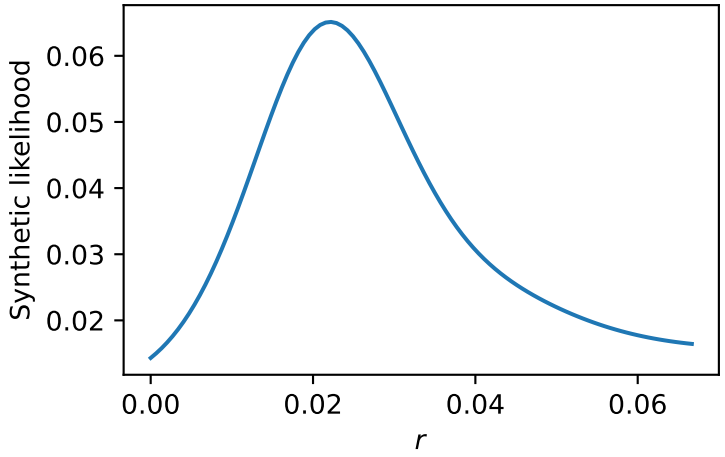

Synthetic likelihood
0.06
0.05
0.04
0.03
0.02
0.01
0.0
0.2
0.4
0.6
0.8
1.0
α


Synthetic likelihood
0.055
0.050
0.045
0.0
0.2
0.4
0.6
0.8
1.0
β


Synthetic likelihood
0.08
0.06
0.04
0.00
0.01
0.02
0.03
γ_L


Synthetic likelihood
0.06
0.04
0.02
0.00
0.00
0.01
0.02
0.03
0.04
0.05
λ






```
# print(index_vals[-600,].round(3))
print(index_vals[-400,].round(3))
print(index_vals[-200,].round(3))
print(index_vals[-80,].round(3))
print(index_vals[-40,].round(3))
print(index_vals[-20,].round(3))
print(index_vals[-8,].round(3))
print(index_vals[-4,].round(3))
print(index_vals[-2,].round(3))
print(index_vals[-1,].round(3))
```

```
[0.3   0.689 0.033 0.029 0.    0.067]
```

```
[0.002 0.318 0.009 0.011 0.019 0.038]
[0.404 0.763 0.01  0.008 0.022 0.028]
[0.146 0.411 0.008 0.011 0.021 0.026]
[0.404 0.    0.022 0.036 0.048 0.028]
[0.781 0.24  0.01  0.011 0.022 0.02 ]
[0.316 0.997 0.012 0.012 0.05  0.039]
[0.316 0.997 0.023 0.012 0.05  0.039]
[0.316 0.997 0.026 0.012 0.05  0.039]
```

```
objects_to_preserve = [
    index_vals,
    obs_vals,
    champ_samp,
    initial_losses_LOOCV,
    slice_samples_dict,
    slice_discrepencies_dict,
    LHC_indices_df,
    gp_samples_dict,
    likelihood_dict,
]

with open("gp_objs.pkl", "wb") as fp:
    pickle.dump(objects_to_preserve, fp)
    print("dictionary saved successfully to file")
```

```
dictionary saved successfully to file
```