Inference on the Champagne Model using a Gaussian Process

TODO

- Set seed for LHC and stuff
- Change to log discrepency with custom observation variance
- Change from MLE to cross validation

Setting up the Champagne Model

Imports

```
import pandas as pd
import numpy as np
from typing import Any
import matplotlib.pyplot as plt

from scipy.stats import qmc

import tensorflow as tf
import tensorflow_probability as tfp

tfb = tfp.bijectors
tfd = tfp.distributions
tfk = tfp.math.psd_kernels
```

```
2024-04-10 22:01:07.636469: I tensorflow/core/platform/cpu_feature_guard.cc:210] This Tensorl
To enable the following instructions: AVX2 FMA, in other operations, rebuild TensorFlow with
2024-04-10 22:01:08.431841: W tensorflow/compiler/tf2tensorrt/utils/py_utils.cc:38] TF-TRT Wa
```

Model itself

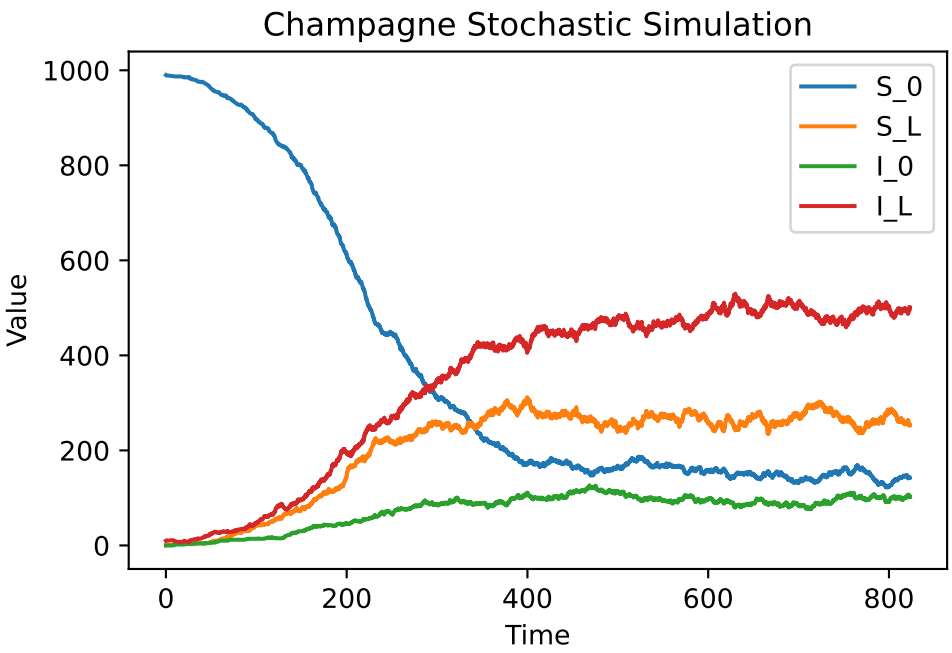
```
np.random.seed(590154)

population = 1000
initial_infecteds = 10
epidemic_length = 1000
number_of_events = 15000

pv_champ_alpha = 0.4  # prop of effective care
pv_champ_beta = 0.4  # prop of radical cure
pv_champ_gamma_L = 1 / 223  # liver stage clearance rate
pv_champ_delta = 0.05  # prop of imported cases
pv_champ_lambda = 0.04  # transmission rate
pv_champ_f = 1 / 72  # relapse frequency
pv_champ_r = 1 / 60  # blood stage clearance rate


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

```
    for i in range(num_events):
        if S_0 == N:
            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
        )

        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,
        ]
        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},
                {"t": t, "S_0": S_0 - 1, "S_L": S_L + 1, "I_0": I_0, "I_L": I_L},
                {"t": t, "S_0": S_0 + 1, "S_L": S_L, "I_0": I_0 - 1, "I_L": I_L},
                {"t": t, "S_0": S_0, "S_L": S_L, "I_0": I_0 - 1, "I_L": I_L + 1},
                {"t": t, "S_0": S_0, "S_L": S_L, "I_0": I_0 + 1, "I_L": I_L - 1},
```

```
                {"t": t, "S_0": S_0, "S_L": S_L + 1, "I_0": I_0, "I_L": I_L - 1},
                {"t": t, "S_0": S_0 + 1, "S_L": S_L - 1, "I_0": I_0, "I_L": I_L},
                {"t": t, "S_0": S_0, "S_L": S_L - 1, "I_0": I_0, "I_L": I_L + 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"]

    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,
)  # .melt(id_vars='t')
```

Plotting outcome

```
champ_samp.plot(x="t", legend=True)
plt.xlabel("Time")
plt.ylabel("Value")
plt.title("Champagne Stochastic Simulation")
plt.savefig("champagne_GP_images/champagne_simulation.pdf")
plt.show()
```



Function that Outputs Final Prevalence

```
def champ_prevalence(alpha_, beta_, gamma_L, lambda_, f, r):
    champ_df_ = champagne_stochastic(alpha_, beta_, gamma_L, lambda_, f, r)

    return champ_df_.iloc[-1]["I_0"] + champ_df_.iloc[-1]["I_L"]


observed_final_prevalence = champ_prevalence(
    pv_champ_alpha,
    pv_champ_beta,
    pv_champ_gamma_L,
    pv_champ_lambda,
    pv_champ_f,
    pv_champ_r,
)


def discrepency_fn(alpha_, beta_, gamma_L, lambda_, f, r):
    x = champ_prevalence(alpha_, beta_, gamma_L, lambda_, f, r)
    return np.abs(x - observed_final_prevalence)
```

Gaussian Process Regression on Final Prevalence Discrepency

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

num_samples = 50

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

pv_champ_alpha = 0.4  # prop of effective care
pv_champ_beta = 0.4  # prop of radical cure
pv_champ_gamma_L = 1 / 223  # liver stage clearance rate
pv_champ_lambda = 0.04  # transmission rate
pv_champ_f = 1 / 72  # relapse frequency
pv_champ_r = 1 / 60  # blood stage clearance rate

samples = np.concatenate(
    (
        my_seed.uniform(low=0, high=1, size=(num_samples, 1)),  # alpha
        my_seed.uniform(low=0, high=1, size=(num_samples, 1)),  # beta
        my_seed.exponential(scale=pv_champ_gamma_L, size=(num_samples, 1)),  # gamma_L
        my_seed.exponential(scale=pv_champ_lambda, size=(num_samples, 1)),  # lambda
        my_seed.exponential(scale=pv_champ_f, size=(num_samples, 1)),  # f
        my_seed.exponential(scale=pv_champ_r, size=(num_samples, 1)),  # r
    ),
    axis=1,
)

LHC_sampler = qmc.LatinHypercube(d=6, seed=my_seed)
LHC_samples = LHC_sampler.random(n=num_samples)
LHC_samples[:, 2] = -pv_champ_gamma_L * np.log(LHC_samples[:, 2])
LHC_samples[:, 3] = -pv_champ_lambda * np.log(LHC_samples[:, 3])
LHC_samples[:, 4] = -pv_champ_f * np.log(LHC_samples[:, 4])
LHC_samples[:, 5] = -pv_champ_r * np.log(LHC_samples[:, 5])

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

print(random_indices_df.head())
print(LHC_indices_df.head())
```

```
      alpha      beta   gamma_L    lambda         f         r
```

```
0  0.201552  0.246202  0.013085  0.051287  0.011657  0.004164
1  0.332324  0.812946  0.000390  0.006251  0.047737  0.018725
2  0.836050  0.343292  0.004725  0.020082  0.004604  0.007983
3  0.566773  0.075311  0.002784  0.007547  0.020959  0.022937
4  0.880603  0.964663  0.004194  0.008378  0.012502  0.009120
      alpha      beta   gamma_L    lambda         f         r
0  0.100008  0.122349  0.005550  0.047169  0.015049  0.023833
1  0.659225  0.590955  0.015422  0.009993  0.026474  0.050003
2  0.503558  0.005003  0.000207  0.024569  0.044514  0.020288
3  0.011840  0.630562  0.001543  0.016033  0.004709  0.010679
4  0.271011  0.942434  0.003873  0.020250  0.006580  0.004226
```

Generate Discrepencies

```
random_discrepencies = LHC_indices_df.apply(
    lambda x: discrepency_fn(
        x["alpha"], x["beta"], x["gamma_L"], x["lambda"], x["f"], x["r"]
    ),
    axis=1,
)

print(random_discrepencies.head())
```

```
0    104.0
1    449.0
2     12.0
3      8.0
4    208.0
dtype: float64
```

Differing Methods to Iterate Function

```
# import timeit

# def function1():
#     np.vectorize(champ_prevalence)(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_prevalence(
#             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())
```

```
2024-04-10 22:01:25.325337: I external/local_xla/xla/stream_executor/cuda/cuda_executor.cc:9
2024-04-10 22:01:25.466987: W tensorflow/core/common_runtime/gpu/gpu_device.cc:2251] Cannot
Skipping registering GPU devices...
```

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=400.0,
```

```
            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=50.0,
            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=500.0,
            dtype=np.float64,
            name="amp_gamma_L_mean",
        )
        self.gamma_L_tp = tf.Variable(
            pv_champ_gamma_L, dtype=np.float64, name="gamma_L_tp"
        )
        self.amp_lambda_mean = tfp.util.TransformedVariable(
            bijector=constrain_positive,
            initial_value=16000.0,
            dtype=np.float64,
            name="amp_lambda_mean",
        )
        self.lambda_tp = tf.Variable(
            pv_champ_lambda, dtype=np.float64, name="lambda_tp"
        )
        self.amp_f_mean = tfp.util.TransformedVariable(
            bijector=constrain_positive,
            initial_value=15000.0,
            dtype=np.float64,
            name="amp_f_mean",
        )
        self.f_tp = tf.Variable(pv_champ_f, dtype=np.float64, name="f_tp")
        self.amp_r_mean = tfp.util.TransformedVariable(
            bijector=constrain_positive,
            initial_value=13000.0,
            dtype=np.float64,
            name="amp_r_mean",
        )
```

```
        self.r_tp = tf.Variable(pv_champ_r, dtype=np.float64, name="r_tp")
        self.bias_mean = tfp.util.TransformedVariable(
            bijector=constrain_positive,
            initial_value=50.0,
            dtype=np.float64,
            name="bias_mean",
        )

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

Making the ARD Kernel

```
index_vals = LHC_indices_df.values
obs_vals = random_discrepencies.values

amplitude_champ = tfp.util.TransformedVariable(
    bijector=constrain_positive,
    initial_value=150.0,
    dtype=np.float64,
    name="amplitude_champ",
)

observation_noise_variance_champ = tfp.util.TransformedVariable(
    bijector=constrain_positive,
    initial_value=1000.0,
    dtype=np.float64,
    name="observation_noise_variance_champ",
)
```

```
length_scales_champ = tfp.util.TransformedVariable(
    bijector=constrain_positive,
```

```
    initial_value=[0.01, 0.01, 0.35, 0.02, 0.27, 0.2],
    dtype=np.float64,
    name="length_scales_champ",
)
```

```
kernel_champ = tfk.FeatureScaled(
    tfk.ExponentiatedQuadratic(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=quad_mean_fn(),
)

print(champ_GP.trainable_variables)

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

```
(<tf.Variable 'amplitude_champ:0' shape=() dtype=float64, numpy=5.0106352940962555>, <tf.Vari
array([-4.60517019, -4.60517019, -1.04982212, -3.91202301, -1.30933332,
       -1.60943791])>, <tf.Variable 'observation_noise_variance_champ:0' shape=() dtype=float
```

Train the Hyperparameters

```
@tf.function(autograph=False, jit_compile=False)
def optimize():
    with tf.GradientTape() as tape:
        loss = -champ_GP.log_prob(obs_vals)
    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
```

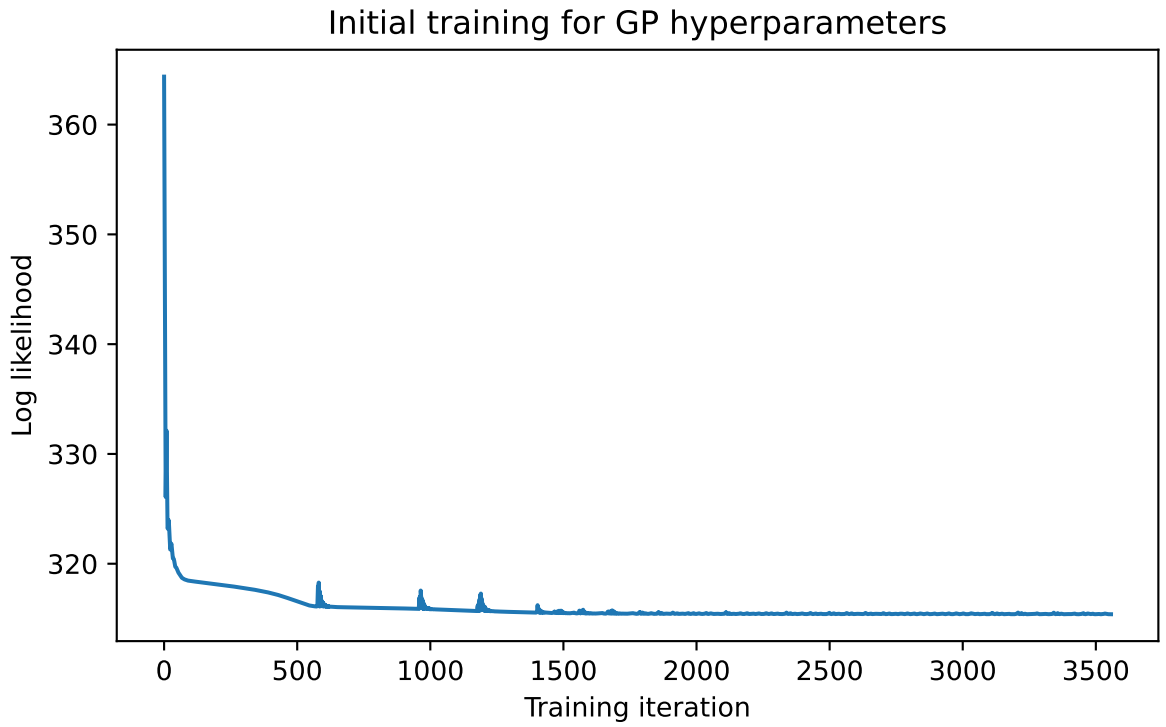
```
Hyperparameter convergence reached at iteration 3558.
```

```
print("Trained parameters:")
for var in champ_GP.trainable_variables:
    if "tp" in var.name:
        print("{} is {}".format(var.name, var.numpy().round(3)))
    else:
        print(
            "{} is {}".format(
                var.name, constrain_positive.forward(var).numpy().round(3)
            )
        )
```

```
Trained parameters:
amplitude_champ:0 is 132.705
length_scales_champ:0 is [7.900e-02 2.000e-02 2.122e+00 1.000e-03 7.661e+00 5.608e+00]
observation_noise_variance_champ:0 is 151.369
alpha_tp:0 is 0.215
amp_alpha_mean:0 is 591.499
amp_beta_mean:0 is 37.292
amp_f_mean:0 is 165497.327
amp_gamma_L_mean:0 is 510282.921
```

```
amp_lambda_mean:0 is 9542.029
amp_r_mean:0 is 47976.955
beta_tp:0 is -0.573
bias_mean:0 is 4.294
f_tp:0 is 0.029
gamma_L_tp:0 is 0.012
lambda_tp:0 is 0.07
r_tp:0 is 0.003
```

```
plt.figure(figsize=(7, 4))
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.pdf")
plt.show()
```



Fitting the GP Regression across alpha

```
plot_samp_no = 21
gp_samp_no = 50
```

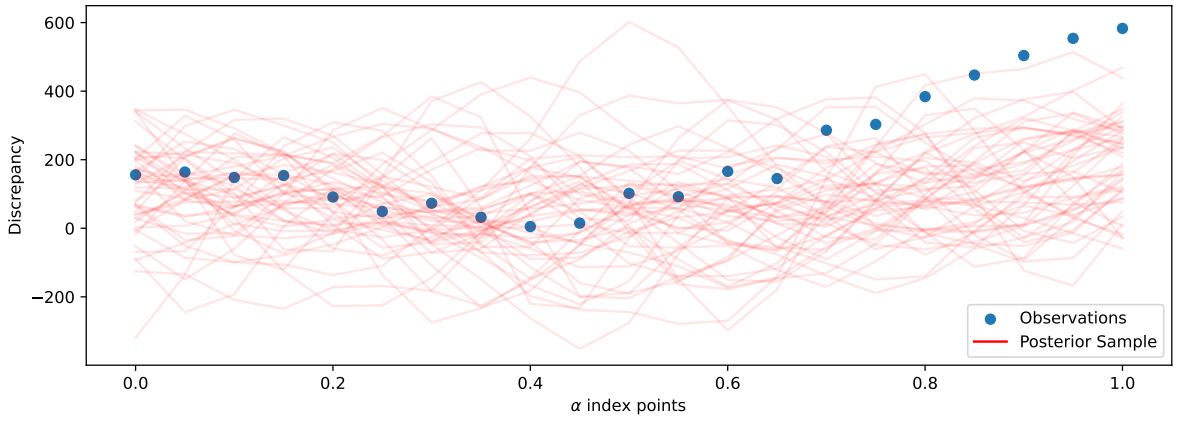
```
alpha_slice_samples = np.concatenate(
    (
        np.linspace(0, 1, plot_samp_no, dtype=np.float64).reshape(-1, 1),  # alpha
        np.repeat(pv_champ_beta, plot_samp_no).reshape(-1, 1),  # beta
        np.repeat(pv_champ_gamma_L, plot_samp_no).reshape(-1, 1),  # gamma_L
        np.repeat(pv_champ_lambda, 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,
)

alpha_slice_indices_df = pd.DataFrame(alpha_slice_samples, columns=variables_names)

print(alpha_slice_indices_df.head())

alpha_slice_discrepencies = alpha_slice_indices_df.apply(
    lambda x: discrepency_fn(
        x["alpha"], x["beta"], x["gamma_L"], x["lambda"], x["f"], x["r"]
    ),
    axis=1,
)

alpha_slice_index_vals = alpha_slice_indices_df.values
```

```
   alpha  beta   gamma_L  lambda         f         r
0   0.00   0.4  0.004484    0.04  0.013889  0.016667
1   0.05   0.4  0.004484    0.04  0.013889  0.016667
2   0.10   0.4  0.004484    0.04  0.013889  0.016667
3   0.15   0.4  0.004484    0.04  0.013889  0.016667
4   0.20   0.4  0.004484    0.04  0.013889  0.016667
```

```
GP_seed = tfp.random.sanitize_seed(4362)

champ_GP_reg = tfd.GaussianProcessRegressionModel(
```

```
    kernel=kernel_champ,
    index_points=alpha_slice_index_vals,
    observation_index_points=index_vals,
    observations=obs_vals,
    observation_noise_variance=observation_noise_variance_champ,
    predictive_noise_variance=0.0,
    mean_fn=quad_mean_fn(),
)

GP_samples = champ_GP_reg.sample(gp_samp_no, seed=GP_seed)
```

```
plt.figure(figsize=(12, 4))
plt.scatter(
    alpha_slice_index_vals[:, 0], alpha_slice_discrepencies, label="Observations"
)
for i in range(gp_samp_no):
    plt.plot(
        alpha_slice_index_vals[:, 0],
        GP_samples[i, :],
        c="r",
        alpha=0.1,
        label="Posterior Sample" if i == 0 else None,
    )
leg = plt.legend(loc="lower right")
for lh in leg.legend_handles:
    lh.set_alpha(1)
plt.xlabel(r"$\alpha$ index points")
plt.ylabel("Discrepancy")
plt.savefig("champagne_GP_images/initial_alpha_slice.pdf")
plt.show()
```



Fitting the GP Regression across beta

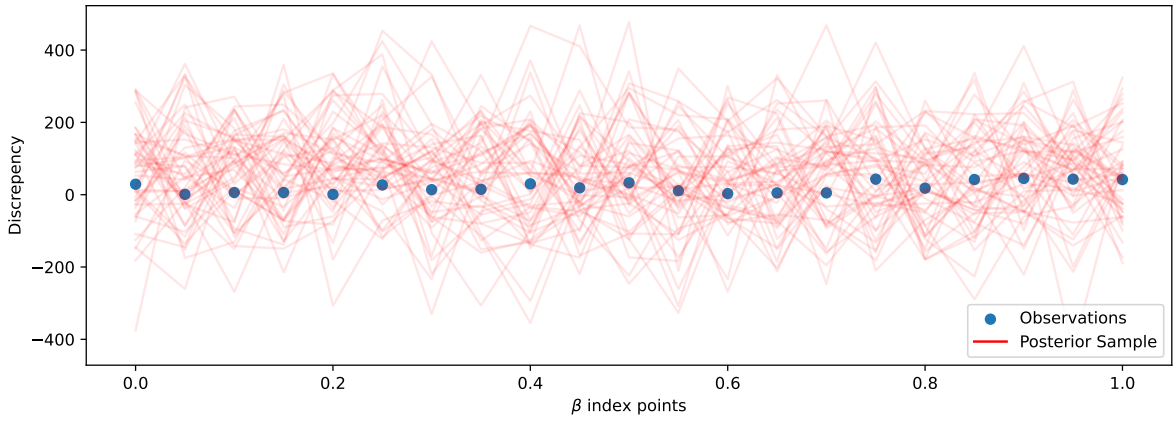
```
beta_slice_samples = np.concatenate(
    (
        np.repeat(pv_champ_alpha, plot_samp_no).reshape(-1, 1),  # alpha
        np.linspace(0, 1, plot_samp_no, dtype=np.float64).reshape(-1, 1),  # beta
        np.repeat(pv_champ_gamma_L, plot_samp_no).reshape(-1, 1),  # gamma_L
        np.repeat(pv_champ_lambda, 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,
)

beta_slice_indices_df = pd.DataFrame(beta_slice_samples, columns=variables_names)

print(beta_slice_indices_df.head())

beta_slice_discrepencies = beta_slice_indices_df.apply(
    lambda x: discrepency_fn(
        x["alpha"], x["beta"], x["gamma_L"], x["lambda"], x["f"], x["r"]
    ),
    axis=1,
)

beta_slice_index_vals = beta_slice_indices_df.values
```

```
   alpha  beta   gamma_L  lambda         f         r
```

```
0    0.4  0.00  0.004484    0.04  0.013889  0.016667
1    0.4  0.05  0.004484    0.04  0.013889  0.016667
2    0.4  0.10  0.004484    0.04  0.013889  0.016667
3    0.4  0.15  0.004484    0.04  0.013889  0.016667
4    0.4  0.20  0.004484    0.04  0.013889  0.016667
```

```
champ_GP_reg = tfd.GaussianProcessRegressionModel(
    kernel=kernel_champ,
    index_points=beta_slice_index_vals,
    observation_index_points=index_vals,
    observations=obs_vals,
    observation_noise_variance=observation_noise_variance_champ,
    predictive_noise_variance=0.0,
    mean_fn=quad_mean_fn(),
)

GP_samples = champ_GP_reg.sample(gp_samp_no, seed=GP_seed)
```

```
plt.figure(figsize=(12, 4))
plt.scatter(beta_slice_index_vals[:, 1], beta_slice_discrepencies, label="Observations")
for i in range(gp_samp_no):
    plt.plot(
        beta_slice_index_vals[:, 1],
        GP_samples[i, :],
        c="r",
        alpha=0.1,
        label="Posterior Sample" if i == 0 else None,
    )
leg = plt.legend(loc="lower right")
for lh in leg.legend_handles:
    lh.set_alpha(1)
plt.xlabel(r"$\beta$ index points")
plt.ylabel("Discrepency")
plt.savefig("champagne_GP_images/initial_beta_slice.pdf")
plt.show()
```



Acquiring the next datapoint to test

Proof that .variance returns what we need in acquisition function

```
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 17610.725
Others Kernel is [0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0.
 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0.
 0. 0.]
```

```
[[ 5.62996682e-005 -3.38945232e-213 -6.00129645e-125 ... -2.07803132e-098
  -2.56893851e-223  1.25300719e-067]
 [-3.38945232e-213  5.62996850e-005 -3.18464650e-140 ... -2.58423919e-131
   2.25082804e-086 -1.62498077e-150]
 [-6.00129645e-125 -3.18464650e-140  5.62996682e-005 ... -4.03392613e-031
   2.40998569e-103 -2.69648092e-062]
 ...
 [-2.07803132e-098 -2.58423919e-131 -4.03392613e-031 ...  5.62996682e-005
  -1.72677825e-129 -9.33693557e-036]
 [-2.56893851e-223  2.25082804e-086  2.40998569e-103 ... -1.72677825e-129
   5.62996682e-005 -1.15426621e-160]
 [ 1.25300719e-067 -1.62498077e-150 -2.69648092e-062 ... -9.33693557e-036
  -1.15426621e-160  5.62996682e-005]]
Variance function is [17610.725]
Variance function is 17610.725
```

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=0.1,
    bijector=constrain_positive,
    dtype=np.float64,
    name="next_gamma_L",
)

next_lambda = tfp.util.TransformedVariable(
    initial_value=0.1,
```

```
    bijector=constrain_positive,
    dtype=np.float64,
    name="next_lambda",
)

next_f = tfp.util.TransformedVariable(
    initial_value=0.1,
    bijector=constrain_positive,
    dtype=np.float64,
    name="next_f",
)

next_r = tfp.util.TransformedVariable(
    initial_value=0.1,
    bijector=constrain_positive,
    dtype=np.float64,
    name="next_r",
)

next_vars = [
    v.trainable_variables[0]
    for v in [next_alpha, next_beta, next_gamma_L, next_lambda, next_f, next_r]
]
```

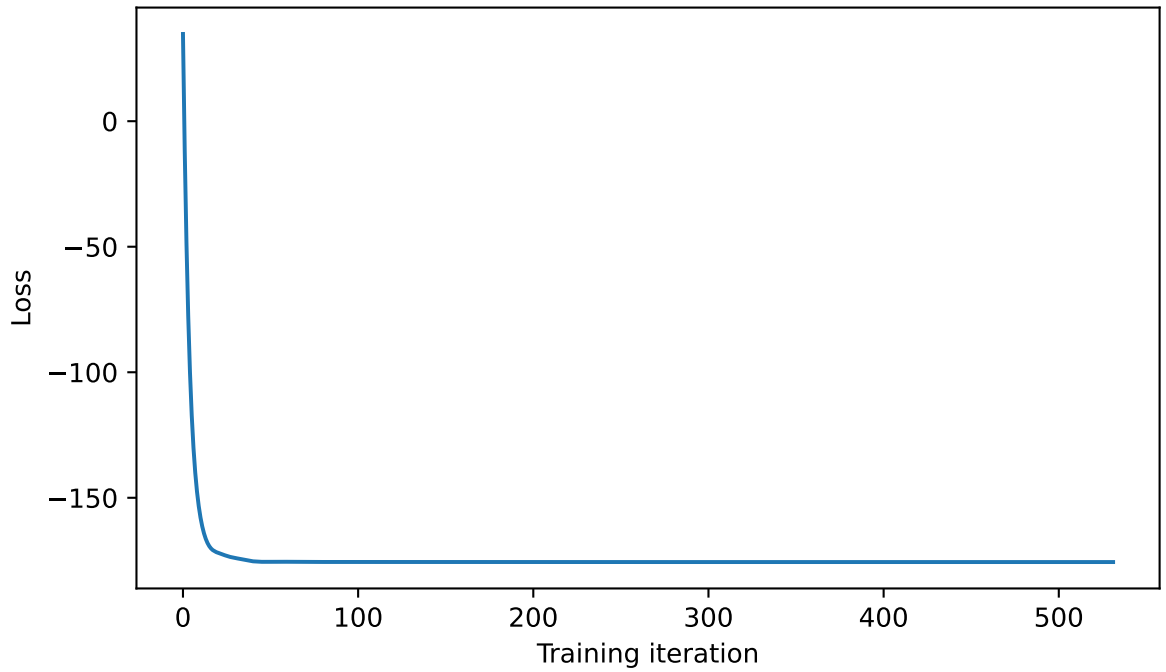
```
Adam_optim = tf.optimizers.Adam(learning_rate=0.1)


@tf.function(autograph=False, jit_compile=False)
def optimize():
    with tf.GradientTape() as tape:
        next_guess = tf.reshape(
            [
                tfb.Sigmoid().forward(next_vars[0]),
                tfb.Sigmoid().forward(next_vars[1]),
                tfb.Sigmoid().forward(next_vars[2]),
                tfb.Sigmoid().forward(next_vars[3]),
                tfb.Sigmoid().forward(next_vars[4]),
                tfb.Sigmoid().forward(next_vars[5]),
            ],
            [1, 6],
        )
        mean_t = champ_GP_reg.mean_fn(next_guess)
```

```
        std_t = champ_GP_reg.stddev(index_points=next_guess)
        loss = tf.squeeze(mean_t - 1.7 * std_t)
    grads = tape.gradient(loss, next_vars)
    Adam_optim.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 = optimize()
    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_vars:
    if ("alpha" in var.name) | ("beta" in var.name):
        print(
            "{} is {}".format(var.name, (tfb.Sigmoid().forward(var).numpy().round(3)))
        )
    else:
        print(
            "{} is {}".format(
                var.name, constrain_positive.forward(var).numpy().round(3)
            )
        )
```

```
Acquisition function convergence reached at iteration 532.
Trained parameters:
next_alpha:0 is 0.4
next_beta:0 is 0.4
```

```
next_gamma_L:0 is 0.005
next_lambda:0 is 0.042
next_f:0 is 0.014
next_r:0 is 0.017
```

```
plt.figure(figsize=(7, 4))
plt.plot(lls_)
plt.xlabel("Training iteration")
plt.ylabel("Loss")
plt.savefig("champagne_GP_images/bolfi_optim_loss.pdf")
plt.show()
```




```
y = 1

def f(x):
    @tf.function
    def g(x):
        x += y
        return x

y = 1
print(f(1))
```

```
y = 2
print(f(1))
```

```
None
None
```

```
exploration_rate = 0.1
d = 6
update_freq = 10  # how many iterations before updating GP hyperparams


def update_GP():
    @tf.function
    def opt_GP():
        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 = opt_GP()
        lls_[i] = loss.numpy()

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

```
def update_var():
    @tf.function
    def opt_var():
        with tf.GradientTape() as tape:
            next_guess = tf.reshape(
                [
                    tfb.Sigmoid().forward(next_vars[0]),
                    tfb.Sigmoid().forward(next_vars[1]),
                    tfb.Sigmoid().forward(next_vars[2]),
                    tfb.Sigmoid().forward(next_vars[3]),
                    tfb.Sigmoid().forward(next_vars[4]),
                    tfb.Sigmoid().forward(next_vars[5]),
                ],
                [1, 6],
            )
            mean_t = champ_GP_reg.mean_fn(next_guess)
            std_t = champ_GP_reg.stddev(index_points=next_guess)
            loss = tf.squeeze(mean_t - eta_t * std_t)
        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(loss)
    for var in optimizer_fast.variables:
        var.assign(tf.zeros_like(var))
```

```
# opt_GP = tf.function(optimize_hypers)
# opt_var = tf.function(optimize_next_var)


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


for t in range(400):
    optimizer_fast = tf.optimizers.Adam(learning_rate=0.01)
    optimizer_slow = tf.optimizers.Adam()
    eta_t = new_eta_t(t, d, exploration_rate)
    print(t)
    new_discrepency = discrepency_fn(
        next_alpha.numpy(),
        next_beta.numpy(),
        next_gamma_L.numpy(),
        next_lambda.numpy(),
        next_f.numpy(),
        next_r.numpy(),
    )

    index_vals = np.append(
        index_vals,
        np.array(
            [
                next_alpha.numpy(),
                next_beta.numpy(),
                next_gamma_L.numpy(),
                next_lambda.numpy(),
                next_f.numpy(),
                next_r.numpy(),
            ]
        ).reshape(1, -1),
        axis=0,
    )
    obs_vals = np.append(obs_vals, new_discrepency)

    if t % update_freq == 0:
        champ_GP = tfd.GaussianProcess(
            kernel=kernel_champ,
```

```
            observation_noise_variance=observation_noise_variance_champ,
            index_points=index_vals,
            mean_fn=quad_mean_fn(),
        )
        update_GP()

    champ_GP_reg = tfd.GaussianProcessRegressionModel(
        kernel=kernel_champ,
        index_points=alpha_slice_index_vals,
        observation_index_points=index_vals,
        observations=obs_vals,
        observation_noise_variance=observation_noise_variance_champ,
        predictive_noise_variance=0.0,
        mean_fn=quad_mean_fn(),
    )
    update_var()


print(index_vals[-200,])
print(index_vals[-20,])
print(index_vals[-2,])
print(index_vals[-1,])
```

```
0
Hyperparameter convergence reached at iteration 5948.
Acquisition function convergence reached at iteration 96.
tf.Tensor(-197.94940914911436, shape=(), dtype=float64)
1
Acquisition function convergence reached at iteration 112.
tf.Tensor(-301.4567938861355, shape=(), dtype=float64)
2
Acquisition function convergence reached at iteration 305.
tf.Tensor(-347.3739472079962, shape=(), dtype=float64)
3
Acquisition function convergence reached at iteration 124.
tf.Tensor(-381.39103842083625, shape=(), dtype=float64)
4
Acquisition function convergence reached at iteration 460.
tf.Tensor(-435.778811458467, shape=(), dtype=float64)
5
Acquisition function convergence reached at iteration 133.
tf.Tensor(-453.2716906814892, shape=(), dtype=float64)
```

```
6
Acquisition function convergence reached at iteration 112.
tf.Tensor(-465.8998052271594, shape=(), dtype=float64)
7
Acquisition function convergence reached at iteration 139.
tf.Tensor(-475.48977245740434, shape=(), dtype=float64)
8
Acquisition function convergence reached at iteration 112.
tf.Tensor(-493.6444814676611, shape=(), dtype=float64)
9
Acquisition function convergence reached at iteration 122.
tf.Tensor(-497.6043173520252, shape=(), dtype=float64)
10
Hyperparameter convergence reached at iteration 7471.
Acquisition function convergence reached at iteration 59.
tf.Tensor(-470.5511351463295, shape=(), dtype=float64)
11
Acquisition function convergence reached at iteration 6586.
tf.Tensor(-490.21931345876493, shape=(), dtype=float64)
12
Acquisition function convergence reached at iteration 361.
tf.Tensor(-493.86672059630484, shape=(), dtype=float64)
13
Acquisition function convergence reached at iteration 3454.
tf.Tensor(-432.4102320052687, shape=(), dtype=float64)
14
Acquisition function convergence reached at iteration 581.
tf.Tensor(-505.119151487905, shape=(), dtype=float64)
15
Acquisition function convergence reached at iteration 379.
tf.Tensor(-511.46683370175475, shape=(), dtype=float64)
16
Acquisition function convergence reached at iteration 4893.
tf.Tensor(-504.30237584095863, shape=(), dtype=float64)
17
Acquisition function convergence reached at iteration 581.
tf.Tensor(-510.3957485832574, shape=(), dtype=float64)
18
Acquisition function convergence reached at iteration 1281.
tf.Tensor(-515.0818514009774, shape=(), dtype=float64)
19
Acquisition function convergence reached at iteration 6694.
tf.Tensor(-517.7485320656162, shape=(), dtype=float64)
```

```
20
Hyperparameter convergence reached at iteration 6834.
Acquisition function convergence reached at iteration 5029.
tf.Tensor(-574.8106951040019, shape=(), dtype=float64)
21
tf.Tensor(-532.9862794901223, shape=(), dtype=float64)
22
tf.Tensor(-424.2386756758308, shape=(), dtype=float64)
23
tf.Tensor(-368.9769688473663, shape=(), dtype=float64)
24
tf.Tensor(-295.14076744453394, shape=(), dtype=float64)
25
tf.Tensor(-311.01971136538907, shape=(), dtype=float64)
26
tf.Tensor(-496.18467237233347, shape=(), dtype=float64)
27
Acquisition function convergence reached at iteration 704.
tf.Tensor(-75.13030153459337, shape=(), dtype=float64)
28
Acquisition function convergence reached at iteration 123.
tf.Tensor(401.77943027396793, shape=(), dtype=float64)
29
tf.Tensor(-22.201867824203646, shape=(), dtype=float64)
30
Hyperparameter convergence reached at iteration 5780.
Acquisition function convergence reached at iteration 506.
tf.Tensor(58.05771317983465, shape=(), dtype=float64)
31
Acquisition function convergence reached at iteration 159.
tf.Tensor(439.09116478938427, shape=(), dtype=float64)
32
Acquisition function convergence reached at iteration 85.
tf.Tensor(517.6828860114805, shape=(), dtype=float64)
33
Acquisition function convergence reached at iteration 183.
tf.Tensor(470.75531839918636, shape=(), dtype=float64)
34
Acquisition function convergence reached at iteration 688.
tf.Tensor(-356.38241296443897, shape=(), dtype=float64)
35
Acquisition function convergence reached at iteration 659.
tf.Tensor(-491.35509819225774, shape=(), dtype=float64)
```

```
36
Acquisition function convergence reached at iteration 172.
tf.Tensor(-132.84362637324438, shape=(), dtype=float64)
37
Acquisition function convergence reached at iteration 180.
tf.Tensor(140.2823630951151, shape=(), dtype=float64)
38
Acquisition function convergence reached at iteration 2444.
tf.Tensor(-496.9730383614632, shape=(), dtype=float64)
39
tf.Tensor(-385.66257626315576, shape=(), dtype=float64)
40
Hyperparameter convergence reached at iteration 5977.
Acquisition function convergence reached at iteration 133.
tf.Tensor(458.38129441093633, shape=(), dtype=float64)
41
Acquisition function convergence reached at iteration 92.
tf.Tensor(519.3471513806087, shape=(), dtype=float64)
42
Acquisition function convergence reached at iteration 4266.
tf.Tensor(-505.42300669027554, shape=(), dtype=float64)
43
Acquisition function convergence reached at iteration 2022.
tf.Tensor(-439.7179621486501, shape=(), dtype=float64)
44
Acquisition function convergence reached at iteration 671.
tf.Tensor(-517.828237176192, shape=(), dtype=float64)
45
Acquisition function convergence reached at iteration 738.
tf.Tensor(-405.78603132161703, shape=(), dtype=float64)
46
Acquisition function convergence reached at iteration 141.
tf.Tensor(-194.4641852965498, shape=(), dtype=float64)
47
tf.Tensor(-122.74823834432107, shape=(), dtype=float64)
48
Acquisition function convergence reached at iteration 530.
tf.Tensor(-6.662470290156449, shape=(), dtype=float64)
49
Acquisition function convergence reached at iteration 87.
tf.Tensor(-34.94033824671974, shape=(), dtype=float64)
50
Hyperparameter convergence reached at iteration 8202.
```

```
Acquisition function convergence reached at iteration 279.
tf.Tensor(-876.6074983562219, shape=(), dtype=float64)
51
Acquisition function convergence reached at iteration 1098.
tf.Tensor(-546.1606209175034, shape=(), dtype=float64)
52
Acquisition function convergence reached at iteration 695.
tf.Tensor(-418.065884369009, shape=(), dtype=float64)
53
tf.Tensor(-539.6040879799774, shape=(), dtype=float64)
54
Acquisition function convergence reached at iteration 2004.
tf.Tensor(-380.5712103073121, shape=(), dtype=float64)
55
Acquisition function convergence reached at iteration 141.
tf.Tensor(38.92690227115054, shape=(), dtype=float64)
56
Acquisition function convergence reached at iteration 616.
tf.Tensor(-236.14866628290105, shape=(), dtype=float64)
57
Acquisition function convergence reached at iteration 308.
tf.Tensor(85.76802217059372, shape=(), dtype=float64)
58
Acquisition function convergence reached at iteration 432.
tf.Tensor(295.62636489787917, shape=(), dtype=float64)
59
Acquisition function convergence reached at iteration 6059.
tf.Tensor(-551.8531204295602, shape=(), dtype=float64)
60
Hyperparameter convergence reached at iteration 7728.
Acquisition function convergence reached at iteration 3666.
tf.Tensor(-664.7979914526373, shape=(), dtype=float64)
61
Acquisition function convergence reached at iteration 369.
tf.Tensor(-554.8404530033705, shape=(), dtype=float64)
62
Acquisition function convergence reached at iteration 195.
tf.Tensor(-412.19848811046194, shape=(), dtype=float64)
63
Acquisition function convergence reached at iteration 2941.
tf.Tensor(-454.93801325799956, shape=(), dtype=float64)
64
Acquisition function convergence reached at iteration 230.
```

```
tf.Tensor(-116.61602536492967, shape=(), dtype=float64)
65
Acquisition function convergence reached at iteration 238.
tf.Tensor(-86.74488564821888, shape=(), dtype=float64)
66
Acquisition function convergence reached at iteration 144.
tf.Tensor(219.64472320744855, shape=(), dtype=float64)
67
Acquisition function convergence reached at iteration 9067.
tf.Tensor(107.0128569436983, shape=(), dtype=float64)
68
Acquisition function convergence reached at iteration 1147.
tf.Tensor(303.01712560764764, shape=(), dtype=float64)
69
Acquisition function convergence reached at iteration 112.
tf.Tensor(362.3507419902261, shape=(), dtype=float64)
70
Hyperparameter convergence reached at iteration 7553.
Acquisition function convergence reached at iteration 147.
tf.Tensor(334.9312545893021, shape=(), dtype=float64)
71
Acquisition function convergence reached at iteration 94.
tf.Tensor(369.62044037818885, shape=(), dtype=float64)
72
Acquisition function convergence reached at iteration 94.
tf.Tensor(378.95240901253453, shape=(), dtype=float64)
73
Acquisition function convergence reached at iteration 69.
tf.Tensor(383.9255089144322, shape=(), dtype=float64)
74
Acquisition function convergence reached at iteration 98.
tf.Tensor(385.69436844103336, shape=(), dtype=float64)
75
Acquisition function convergence reached at iteration 244.
tf.Tensor(-675.2002187903893, shape=(), dtype=float64)
76
Acquisition function convergence reached at iteration 158.
tf.Tensor(-277.76377790245846, shape=(), dtype=float64)
77
Acquisition function convergence reached at iteration 155.
tf.Tensor(168.66816258551182, shape=(), dtype=float64)
78
Acquisition function convergence reached at iteration 81.
```

```
tf.Tensor(328.2408364578107, shape=(), dtype=float64)
79
Acquisition function convergence reached at iteration 139.
tf.Tensor(334.29872243385125, shape=(), dtype=float64)
80
Hyperparameter convergence reached at iteration 9162.
Acquisition function convergence reached at iteration 100.
tf.Tensor(350.0142648771796, shape=(), dtype=float64)
81
Acquisition function convergence reached at iteration 73.
tf.Tensor(347.17567296302946, shape=(), dtype=float64)
82
Acquisition function convergence reached at iteration 79.
tf.Tensor(341.45672763115743, shape=(), dtype=float64)
83
Acquisition function convergence reached at iteration 74.
tf.Tensor(336.90527133464883, shape=(), dtype=float64)
84
Acquisition function convergence reached at iteration 92.
tf.Tensor(327.91351218864787, shape=(), dtype=float64)
85
Acquisition function convergence reached at iteration 74.
tf.Tensor(318.4157138324269, shape=(), dtype=float64)
86
Acquisition function convergence reached at iteration 89.
tf.Tensor(299.6369820941309, shape=(), dtype=float64)
87
Acquisition function convergence reached at iteration 119.
tf.Tensor(257.768555189103, shape=(), dtype=float64)
88
Acquisition function convergence reached at iteration 105.
tf.Tensor(288.1844747966311, shape=(), dtype=float64)
89
Acquisition function convergence reached at iteration 114.
tf.Tensor(313.6344031957819, shape=(), dtype=float64)
90
Hyperparameter convergence reached at iteration 9100.
Acquisition function convergence reached at iteration 158.
tf.Tensor(62.052647823026234, shape=(), dtype=float64)
91
Acquisition function convergence reached at iteration 148.
tf.Tensor(-83.41884184854956, shape=(), dtype=float64)
92
```

```
Acquisition function convergence reached at iteration 155.
tf.Tensor(92.45835859309835, shape=(), dtype=float64)
93
Acquisition function convergence reached at iteration 141.
tf.Tensor(207.10690540614635, shape=(), dtype=float64)
94
Acquisition function convergence reached at iteration 141.
tf.Tensor(196.25005727363734, shape=(), dtype=float64)
95
Acquisition function convergence reached at iteration 76.
tf.Tensor(249.5430935525046, shape=(), dtype=float64)
96
Acquisition function convergence reached at iteration 180.
tf.Tensor(188.43273325777955, shape=(), dtype=float64)
97
Acquisition function convergence reached at iteration 116.
tf.Tensor(252.84921332178013, shape=(), dtype=float64)
98
Acquisition function convergence reached at iteration 1377.
tf.Tensor(-507.6494286488235, shape=(), dtype=float64)
99
Acquisition function convergence reached at iteration 481.
tf.Tensor(-491.0266909207868, shape=(), dtype=float64)
100
Hyperparameter convergence reached at iteration 6564.
Acquisition function convergence reached at iteration 557.
tf.Tensor(-523.3672561824004, shape=(), dtype=float64)
101
Acquisition function convergence reached at iteration 497.
tf.Tensor(-321.28130115959686, shape=(), dtype=float64)
102
Acquisition function convergence reached at iteration 561.
tf.Tensor(-515.0793928778915, shape=(), dtype=float64)
103
Acquisition function convergence reached at iteration 403.
tf.Tensor(-415.2112395762068, shape=(), dtype=float64)
104
Acquisition function convergence reached at iteration 151.
tf.Tensor(-181.5075309947427, shape=(), dtype=float64)
105
Acquisition function convergence reached at iteration 157.
tf.Tensor(-37.06498146919995, shape=(), dtype=float64)
106
```

```
Acquisition function convergence reached at iteration 752.
tf.Tensor(-525.8935764415685, shape=(), dtype=float64)
107
Acquisition function convergence reached at iteration 291.
tf.Tensor(-499.2838051063721, shape=(), dtype=float64)
108
Acquisition function convergence reached at iteration 154.
tf.Tensor(-472.3666496164958, shape=(), dtype=float64)
109
Acquisition function convergence reached at iteration 162.
tf.Tensor(-249.27068502297135, shape=(), dtype=float64)
110
Hyperparameter convergence reached at iteration 6527.
Acquisition function convergence reached at iteration 877.
tf.Tensor(-536.7369600939859, shape=(), dtype=float64)
111
Acquisition function convergence reached at iteration 300.
tf.Tensor(-486.1274889505571, shape=(), dtype=float64)
112
Acquisition function convergence reached at iteration 8163.
tf.Tensor(-377.8892374505922, shape=(), dtype=float64)
113
Acquisition function convergence reached at iteration 1382.
tf.Tensor(-16.459612650070426, shape=(), dtype=float64)
114
Acquisition function convergence reached at iteration 113.
tf.Tensor(154.04400581314, shape=(), dtype=float64)
115
Acquisition function convergence reached at iteration 109.
tf.Tensor(229.88782959272078, shape=(), dtype=float64)
116
Acquisition function convergence reached at iteration 110.
tf.Tensor(273.788907128053, shape=(), dtype=float64)
117
Acquisition function convergence reached at iteration 71.
tf.Tensor(303.12772640604567, shape=(), dtype=float64)
118
Acquisition function convergence reached at iteration 94.
tf.Tensor(324.46962387091855, shape=(), dtype=float64)
119
Acquisition function convergence reached at iteration 89.
tf.Tensor(340.8419363624017, shape=(), dtype=float64)
120
```

```
Hyperparameter convergence reached at iteration 5763.
Acquisition function convergence reached at iteration 120.
tf.Tensor(298.7344590320322, shape=(), dtype=float64)
121
Acquisition function convergence reached at iteration 76.
tf.Tensor(316.77246665382563, shape=(), dtype=float64)
122
Acquisition function convergence reached at iteration 96.
tf.Tensor(331.565270271762, shape=(), dtype=float64)
123
Acquisition function convergence reached at iteration 72.
tf.Tensor(343.87906626155774, shape=(), dtype=float64)
124
Acquisition function convergence reached at iteration 75.
tf.Tensor(354.25329455980295, shape=(), dtype=float64)
125
Acquisition function convergence reached at iteration 80.
tf.Tensor(363.0772516281311, shape=(), dtype=float64)
126
Acquisition function convergence reached at iteration 77.
tf.Tensor(370.6293172644505, shape=(), dtype=float64)
127
Acquisition function convergence reached at iteration 73.
tf.Tensor(377.0994460931049, shape=(), dtype=float64)
128
Acquisition function convergence reached at iteration 5179.
tf.Tensor(320.08886606012527, shape=(), dtype=float64)
129
Acquisition function convergence reached at iteration 1479.
tf.Tensor(303.67038626175326, shape=(), dtype=float64)
130
Hyperparameter convergence reached at iteration 6545.
Acquisition function convergence reached at iteration 42.
tf.Tensor(310.2507819577667, shape=(), dtype=float64)
131
Acquisition function convergence reached at iteration 43.
tf.Tensor(306.44507489464877, shape=(), dtype=float64)
132
Acquisition function convergence reached at iteration 22.
tf.Tensor(302.80729629350805, shape=(), dtype=float64)
133
Acquisition function convergence reached at iteration 54.
tf.Tensor(299.09934806759526, shape=(), dtype=float64)
```

```
134
Acquisition function convergence reached at iteration 42.
tf.Tensor(296.1439260159672, shape=(), dtype=float64)
135
Acquisition function convergence reached at iteration 46.
tf.Tensor(293.61456313354904, shape=(), dtype=float64)
136
Acquisition function convergence reached at iteration 36.
tf.Tensor(290.5552129479187, shape=(), dtype=float64)
137
Acquisition function convergence reached at iteration 55.
tf.Tensor(288.2946067168298, shape=(), dtype=float64)
138
Acquisition function convergence reached at iteration 31.
tf.Tensor(286.5716241165852, shape=(), dtype=float64)
139
Acquisition function convergence reached at iteration 37.
tf.Tensor(283.9741742335914, shape=(), dtype=float64)
140
Hyperparameter convergence reached at iteration 7053.
tf.Tensor(-207.90000249891074, shape=(), dtype=float64)
141
Acquisition function convergence reached at iteration 10.
tf.Tensor(108.65200133602173, shape=(), dtype=float64)
142
Acquisition function convergence reached at iteration 448.
tf.Tensor(195.24283473635973, shape=(), dtype=float64)
143
Acquisition function convergence reached at iteration 70.
tf.Tensor(180.6791384353646, shape=(), dtype=float64)
144
Acquisition function convergence reached at iteration 45.
tf.Tensor(168.64452042560154, shape=(), dtype=float64)
145
Acquisition function convergence reached at iteration 64.
tf.Tensor(156.50237468883768, shape=(), dtype=float64)
146
Acquisition function convergence reached at iteration 66.
tf.Tensor(146.3476952147613, shape=(), dtype=float64)
147
Acquisition function convergence reached at iteration 73.
tf.Tensor(137.29602603422148, shape=(), dtype=float64)
148
```

```
Acquisition function convergence reached at iteration 69.
tf.Tensor(127.08120376508856, shape=(), dtype=float64)
149
Acquisition function convergence reached at iteration 2614.
tf.Tensor(115.59958545629425, shape=(), dtype=float64)
150
Hyperparameter convergence reached at iteration 6129.
Acquisition function convergence reached at iteration 445.
tf.Tensor(-111.78501476848311, shape=(), dtype=float64)
151
Acquisition function convergence reached at iteration 85.
tf.Tensor(-32.521830300083366, shape=(), dtype=float64)
152
Acquisition function convergence reached at iteration 103.
tf.Tensor(18.167642872450784, shape=(), dtype=float64)
153
Acquisition function convergence reached at iteration 91.
tf.Tensor(52.21846835706671, shape=(), dtype=float64)
154
Acquisition function convergence reached at iteration 87.
tf.Tensor(76.6423749065809, shape=(), dtype=float64)
155
Acquisition function convergence reached at iteration 79.
tf.Tensor(92.8343835415765, shape=(), dtype=float64)
156
Acquisition function convergence reached at iteration 706.
tf.Tensor(-701.1610547245867, shape=(), dtype=float64)
157
Acquisition function convergence reached at iteration 404.
tf.Tensor(-612.8618342698799, shape=(), dtype=float64)
158
Acquisition function convergence reached at iteration 2425.
tf.Tensor(-698.914118155856, shape=(), dtype=float64)
159
Acquisition function convergence reached at iteration 469.
tf.Tensor(-657.2003423458806, shape=(), dtype=float64)
160
Hyperparameter convergence reached at iteration 6492.
Acquisition function convergence reached at iteration 858.
tf.Tensor(-622.7560298217677, shape=(), dtype=float64)
161
Acquisition function convergence reached at iteration 195.
tf.Tensor(-558.3786822223165, shape=(), dtype=float64)
```

```
162
Acquisition function convergence reached at iteration 225.
tf.Tensor(-578.8268388758255, shape=(), dtype=float64)
163
tf.Tensor(-372.02236481570947, shape=(), dtype=float64)
164
Acquisition function convergence reached at iteration 142.
tf.Tensor(-24.934727519301077, shape=(), dtype=float64)
165
Acquisition function convergence reached at iteration 521.
tf.Tensor(-17.72941313390413, shape=(), dtype=float64)
166
Acquisition function convergence reached at iteration 101.
tf.Tensor(99.81677933953631, shape=(), dtype=float64)
167
Acquisition function convergence reached at iteration 154.
tf.Tensor(91.66797801904693, shape=(), dtype=float64)
168
Acquisition function convergence reached at iteration 81.
tf.Tensor(147.83540103779254, shape=(), dtype=float64)
169
Acquisition function convergence reached at iteration 176.
tf.Tensor(154.88695498932287, shape=(), dtype=float64)
170
Hyperparameter convergence reached at iteration 5192.
Acquisition function convergence reached at iteration 141.
tf.Tensor(186.72000613781304, shape=(), dtype=float64)
171
Acquisition function convergence reached at iteration 134.
tf.Tensor(202.74852712692936, shape=(), dtype=float64)
172
Acquisition function convergence reached at iteration 200.
tf.Tensor(198.9776614417611, shape=(), dtype=float64)
173
Acquisition function convergence reached at iteration 60.
tf.Tensor(218.61597253368475, shape=(), dtype=float64)
174
Acquisition function convergence reached at iteration 61.
tf.Tensor(233.76709810369752, shape=(), dtype=float64)
175
Acquisition function convergence reached at iteration 76.
tf.Tensor(246.01262512929617, shape=(), dtype=float64)
176
```

```
Acquisition function convergence reached at iteration 60.
tf.Tensor(255.93800974776988, shape=(), dtype=float64)
177
Acquisition function convergence reached at iteration 693.
tf.Tensor(-299.26735245507234, shape=(), dtype=float64)
178
Acquisition function convergence reached at iteration 493.
tf.Tensor(-618.0888257310219, shape=(), dtype=float64)
179
Acquisition function convergence reached at iteration 307.
tf.Tensor(-579.5727628999065, shape=(), dtype=float64)
180
Hyperparameter convergence reached at iteration 4965.
Acquisition function convergence reached at iteration 1983.
tf.Tensor(-492.8563740256126, shape=(), dtype=float64)
181
Acquisition function convergence reached at iteration 161.
tf.Tensor(-417.84068533185626, shape=(), dtype=float64)
182
Acquisition function convergence reached at iteration 150.
tf.Tensor(-345.36004094206123, shape=(), dtype=float64)
183
Acquisition function convergence reached at iteration 206.
tf.Tensor(-294.575255094504, shape=(), dtype=float64)
184
Acquisition function convergence reached at iteration 7508.
tf.Tensor(-360.5571774538904, shape=(), dtype=float64)
185
Acquisition function convergence reached at iteration 1018.
tf.Tensor(-644.5845048496706, shape=(), dtype=float64)
186
Acquisition function convergence reached at iteration 133.
tf.Tensor(-623.3700789141255, shape=(), dtype=float64)
187
Acquisition function convergence reached at iteration 803.
tf.Tensor(-593.3716390335467, shape=(), dtype=float64)
188
Acquisition function convergence reached at iteration 1331.
tf.Tensor(-559.1435494548439, shape=(), dtype=float64)
189
Acquisition function convergence reached at iteration 6455.
tf.Tensor(-382.6120094982882, shape=(), dtype=float64)
190
```

```
Hyperparameter convergence reached at iteration 6384.
Acquisition function convergence reached at iteration 1136.
tf.Tensor(-573.4695873237429, shape=(), dtype=float64)
191
Acquisition function convergence reached at iteration 140.
tf.Tensor(-560.9701890160518, shape=(), dtype=float64)
192
Acquisition function convergence reached at iteration 275.
tf.Tensor(-573.8793856872471, shape=(), dtype=float64)
193
Acquisition function convergence reached at iteration 221.
tf.Tensor(-563.7210368092022, shape=(), dtype=float64)
194
Acquisition function convergence reached at iteration 160.
tf.Tensor(-498.56036912768826, shape=(), dtype=float64)
195
Acquisition function convergence reached at iteration 301.
tf.Tensor(-549.8421582754249, shape=(), dtype=float64)
196
Acquisition function convergence reached at iteration 125.
tf.Tensor(-473.1931680790334, shape=(), dtype=float64)
197
Acquisition function convergence reached at iteration 146.
tf.Tensor(-336.2267510465745, shape=(), dtype=float64)
198
tf.Tensor(-425.43629876796876, shape=(), dtype=float64)
199
Acquisition function convergence reached at iteration 1051.
tf.Tensor(-417.6998530774163, shape=(), dtype=float64)
200
Hyperparameter convergence reached at iteration 8229.
Acquisition function convergence reached at iteration 5302.
tf.Tensor(-24.904895936009268, shape=(), dtype=float64)
201
Acquisition function convergence reached at iteration 29.
tf.Tensor(-20.256938827202447, shape=(), dtype=float64)
202
Acquisition function convergence reached at iteration 56.
tf.Tensor(-14.23735973856708, shape=(), dtype=float64)
203
Acquisition function convergence reached at iteration 71.
tf.Tensor(-9.560591104059995, shape=(), dtype=float64)
204
```

```
Acquisition function convergence reached at iteration 37.
tf.Tensor(-4.696310251295586, shape=(), dtype=float64)
205
Acquisition function convergence reached at iteration 44.
tf.Tensor(0.6750917775864451, shape=(), dtype=float64)
206
Acquisition function convergence reached at iteration 73.
tf.Tensor(4.852680733479119, shape=(), dtype=float64)
207
Acquisition function convergence reached at iteration 74.
tf.Tensor(8.509367328831914, shape=(), dtype=float64)
208
Acquisition function convergence reached at iteration 52.
tf.Tensor(11.53979319797891, shape=(), dtype=float64)
209
Acquisition function convergence reached at iteration 49.
tf.Tensor(14.725551670296127, shape=(), dtype=float64)
210
Hyperparameter convergence reached at iteration 7432.
Acquisition function convergence reached at iteration 105.
tf.Tensor(12.846173347175807, shape=(), dtype=float64)
211
Acquisition function convergence reached at iteration 53.
tf.Tensor(14.73111475015321, shape=(), dtype=float64)
212
Acquisition function convergence reached at iteration 49.
tf.Tensor(16.00990459736302, shape=(), dtype=float64)
213
Acquisition function convergence reached at iteration 69.
tf.Tensor(18.559881312780504, shape=(), dtype=float64)
214
Acquisition function convergence reached at iteration 53.
tf.Tensor(21.13925390090745, shape=(), dtype=float64)
215
Acquisition function convergence reached at iteration 47.
tf.Tensor(22.114882737462622, shape=(), dtype=float64)
216
Acquisition function convergence reached at iteration 60.
tf.Tensor(23.92170431110661, shape=(), dtype=float64)
217
Acquisition function convergence reached at iteration 80.
tf.Tensor(27.206946569187224, shape=(), dtype=float64)
218
```

```
Acquisition function convergence reached at iteration 24.
tf.Tensor(27.717890832478744, shape=(), dtype=float64)
219
Acquisition function convergence reached at iteration 110.
tf.Tensor(28.41186861121055, shape=(), dtype=float64)
220
Hyperparameter convergence reached at iteration 7900.
Acquisition function convergence reached at iteration 73.
tf.Tensor(32.880963528096885, shape=(), dtype=float64)
221
Acquisition function convergence reached at iteration 117.
tf.Tensor(31.280968211696546, shape=(), dtype=float64)
222
Acquisition function convergence reached at iteration 54.
tf.Tensor(31.309112007162085, shape=(), dtype=float64)
223
Acquisition function convergence reached at iteration 65.
tf.Tensor(30.27687953803563, shape=(), dtype=float64)
224
Acquisition function convergence reached at iteration 60.
tf.Tensor(28.34640619370144, shape=(), dtype=float64)
225
Acquisition function convergence reached at iteration 63.
tf.Tensor(27.511384938243538, shape=(), dtype=float64)
226
Acquisition function convergence reached at iteration 62.
tf.Tensor(26.356006848922988, shape=(), dtype=float64)
227
Acquisition function convergence reached at iteration 56.
tf.Tensor(26.169467257651874, shape=(), dtype=float64)
228
Acquisition function convergence reached at iteration 52.
tf.Tensor(25.8190611701811, shape=(), dtype=float64)
229
Acquisition function convergence reached at iteration 74.
tf.Tensor(25.232187718624928, shape=(), dtype=float64)
230
Hyperparameter convergence reached at iteration 7447.
Acquisition function convergence reached at iteration 127.
tf.Tensor(-16.59853961032462, shape=(), dtype=float64)
231
Acquisition function convergence reached at iteration 78.
tf.Tensor(-21.893710318676483, shape=(), dtype=float64)
```

```
232
Acquisition function convergence reached at iteration 75.
tf.Tensor(-24.778825702889208, shape=(), dtype=float64)
233
Acquisition function convergence reached at iteration 65.
tf.Tensor(-25.55090268399789, shape=(), dtype=float64)
234
Acquisition function convergence reached at iteration 74.
tf.Tensor(-25.557989864788993, shape=(), dtype=float64)
235
Acquisition function convergence reached at iteration 53.
tf.Tensor(-26.467246543938984, shape=(), dtype=float64)
236
Acquisition function convergence reached at iteration 73.
tf.Tensor(-25.734260059156412, shape=(), dtype=float64)
237
Acquisition function convergence reached at iteration 74.
tf.Tensor(-25.173573063114205, shape=(), dtype=float64)
238
Acquisition function convergence reached at iteration 84.
tf.Tensor(-25.452236080187603, shape=(), dtype=float64)
239
Acquisition function convergence reached at iteration 40.
tf.Tensor(-24.371461501469355, shape=(), dtype=float64)
240
Hyperparameter convergence reached at iteration 8919.
Acquisition function convergence reached at iteration 113.
tf.Tensor(-82.43009095618751, shape=(), dtype=float64)
241
Acquisition function convergence reached at iteration 72.
tf.Tensor(-71.59163810442153, shape=(), dtype=float64)
242
Acquisition function convergence reached at iteration 82.
tf.Tensor(-65.04443199145258, shape=(), dtype=float64)
243
Acquisition function convergence reached at iteration 86.
tf.Tensor(-58.23617352107155, shape=(), dtype=float64)
244
Acquisition function convergence reached at iteration 58.
tf.Tensor(-55.705980294884185, shape=(), dtype=float64)
245
Acquisition function convergence reached at iteration 67.
tf.Tensor(-52.45693617956992, shape=(), dtype=float64)
```

```
246
Acquisition function convergence reached at iteration 47.
tf.Tensor(-51.632956062359504, shape=(), dtype=float64)
247
Acquisition function convergence reached at iteration 72.
tf.Tensor(-50.69612730121758, shape=(), dtype=float64)
248
Acquisition function convergence reached at iteration 72.
tf.Tensor(-48.78282819631766, shape=(), dtype=float64)
249
Acquisition function convergence reached at iteration 47.
tf.Tensor(-47.967054144103436, shape=(), dtype=float64)
250
Hyperparameter convergence reached at iteration 9103.
Acquisition function convergence reached at iteration 87.
tf.Tensor(-46.13442972088879, shape=(), dtype=float64)
251
Acquisition function convergence reached at iteration 59.
tf.Tensor(-45.27266982107071, shape=(), dtype=float64)
252
Acquisition function convergence reached at iteration 56.
tf.Tensor(-44.23082016510049, shape=(), dtype=float64)
253
Acquisition function convergence reached at iteration 62.
tf.Tensor(-43.732535185382865, shape=(), dtype=float64)
254
Acquisition function convergence reached at iteration 62.
tf.Tensor(-42.29768419031113, shape=(), dtype=float64)
255
Acquisition function convergence reached at iteration 62.
tf.Tensor(-41.72595937495946, shape=(), dtype=float64)
256
Acquisition function convergence reached at iteration 56.
tf.Tensor(-41.26849512212226, shape=(), dtype=float64)
257
Acquisition function convergence reached at iteration 54.
tf.Tensor(-40.20905510432779, shape=(), dtype=float64)
258
Acquisition function convergence reached at iteration 62.
tf.Tensor(-39.645344323400366, shape=(), dtype=float64)
259
Acquisition function convergence reached at iteration 59.
tf.Tensor(-39.129814273287906, shape=(), dtype=float64)
```

```
260
Hyperparameter convergence reached at iteration 9556.
Acquisition function convergence reached at iteration 76.
tf.Tensor(-38.046696068523275, shape=(), dtype=float64)
261
Acquisition function convergence reached at iteration 57.
tf.Tensor(-37.47342668019089, shape=(), dtype=float64)
262
Acquisition function convergence reached at iteration 72.
tf.Tensor(-37.055119960201, shape=(), dtype=float64)
263
Acquisition function convergence reached at iteration 52.
tf.Tensor(-36.70117678800864, shape=(), dtype=float64)
264
Acquisition function convergence reached at iteration 34.
tf.Tensor(-36.27883136365547, shape=(), dtype=float64)
265
Acquisition function convergence reached at iteration 61.
tf.Tensor(-35.719099099349584, shape=(), dtype=float64)
266
Acquisition function convergence reached at iteration 30.
tf.Tensor(-35.10514166574432, shape=(), dtype=float64)
267
Acquisition function convergence reached at iteration 130.
tf.Tensor(-45.919679095367584, shape=(), dtype=float64)
268
Acquisition function convergence reached at iteration 70.
tf.Tensor(-39.768046454254815, shape=(), dtype=float64)
269
Acquisition function convergence reached at iteration 85.
tf.Tensor(-34.31332747946179, shape=(), dtype=float64)
270
Hyperparameter convergence reached at iteration 9837.
Acquisition function convergence reached at iteration 138.
tf.Tensor(-33.66455634959037, shape=(), dtype=float64)
271
Acquisition function convergence reached at iteration 78.
tf.Tensor(-33.157620981730346, shape=(), dtype=float64)
272
Acquisition function convergence reached at iteration 56.
tf.Tensor(-32.66712259181594, shape=(), dtype=float64)
273
Acquisition function convergence reached at iteration 59.
```

```
tf.Tensor(-32.21556168455481, shape=(), dtype=float64)
274
Acquisition function convergence reached at iteration 68.
tf.Tensor(-32.11763902187811, shape=(), dtype=float64)
275
Acquisition function convergence reached at iteration 80.
tf.Tensor(-31.995364987618622, shape=(), dtype=float64)
276
Acquisition function convergence reached at iteration 77.
tf.Tensor(-31.286947688738422, shape=(), dtype=float64)
277
Acquisition function convergence reached at iteration 79.
tf.Tensor(-30.96349393019397, shape=(), dtype=float64)
278
Acquisition function convergence reached at iteration 84.
tf.Tensor(-30.772397793158056, shape=(), dtype=float64)
279
Acquisition function convergence reached at iteration 47.
tf.Tensor(-30.1979504021106, shape=(), dtype=float64)
280
Hyperparameter convergence reached at iteration 8029.
Acquisition function convergence reached at iteration 84.
tf.Tensor(-28.790692027557917, shape=(), dtype=float64)
281
Acquisition function convergence reached at iteration 62.
tf.Tensor(-28.45117684520848, shape=(), dtype=float64)
282
Acquisition function convergence reached at iteration 73.
tf.Tensor(-27.92804239326299, shape=(), dtype=float64)
283
Acquisition function convergence reached at iteration 75.
tf.Tensor(-27.345400017073647, shape=(), dtype=float64)
284
Acquisition function convergence reached at iteration 81.
tf.Tensor(-27.279583974606997, shape=(), dtype=float64)
285
Acquisition function convergence reached at iteration 72.
tf.Tensor(-27.139769404927794, shape=(), dtype=float64)
286
Acquisition function convergence reached at iteration 72.
tf.Tensor(-26.663052690585726, shape=(), dtype=float64)
287
Acquisition function convergence reached at iteration 57.
```

```
tf.Tensor(-26.112601474888873, shape=(), dtype=float64)
288
Acquisition function convergence reached at iteration 72.
tf.Tensor(-25.92092324282345, shape=(), dtype=float64)
289
Acquisition function convergence reached at iteration 67.
tf.Tensor(-25.81675383771566, shape=(), dtype=float64)
290
Hyperparameter convergence reached at iteration 7176.
Acquisition function convergence reached at iteration 57.
tf.Tensor(-24.895820448590797, shape=(), dtype=float64)
291
Acquisition function convergence reached at iteration 71.
tf.Tensor(-24.683693174602567, shape=(), dtype=float64)
292
Acquisition function convergence reached at iteration 57.
tf.Tensor(-24.652764531993782, shape=(), dtype=float64)
293
Acquisition function convergence reached at iteration 65.
tf.Tensor(-24.303214395724225, shape=(), dtype=float64)
294
Acquisition function convergence reached at iteration 68.
tf.Tensor(-24.24254307038037, shape=(), dtype=float64)
295
Acquisition function convergence reached at iteration 82.
tf.Tensor(-23.96668868136986, shape=(), dtype=float64)
296
Acquisition function convergence reached at iteration 57.
tf.Tensor(-23.712014744677333, shape=(), dtype=float64)
297
Acquisition function convergence reached at iteration 58.
tf.Tensor(-23.68525662810506, shape=(), dtype=float64)
298
Acquisition function convergence reached at iteration 70.
tf.Tensor(-23.262613358785686, shape=(), dtype=float64)
299
Acquisition function convergence reached at iteration 50.
tf.Tensor(-22.879469151209033, shape=(), dtype=float64)
300
Hyperparameter convergence reached at iteration 8137.
Acquisition function convergence reached at iteration 63.
tf.Tensor(-22.059770468341622, shape=(), dtype=float64)
301
```

```
Acquisition function convergence reached at iteration 61.
tf.Tensor(-21.791401884477853, shape=(), dtype=float64)
302
Acquisition function convergence reached at iteration 56.
tf.Tensor(-21.63915741332531, shape=(), dtype=float64)
303
Acquisition function convergence reached at iteration 55.
tf.Tensor(-21.584357366586662, shape=(), dtype=float64)
304
Acquisition function convergence reached at iteration 129.
tf.Tensor(-25.71163142529967, shape=(), dtype=float64)
305
Acquisition function convergence reached at iteration 66.
tf.Tensor(-23.498987387284657, shape=(), dtype=float64)
306
Acquisition function convergence reached at iteration 61.
tf.Tensor(-22.568457205608773, shape=(), dtype=float64)
307
Acquisition function convergence reached at iteration 61.
tf.Tensor(-21.120039305725612, shape=(), dtype=float64)
308
Acquisition function convergence reached at iteration 80.
tf.Tensor(-19.764591167358994, shape=(), dtype=float64)
309
Acquisition function convergence reached at iteration 52.
tf.Tensor(-19.070103477911914, shape=(), dtype=float64)
310
Hyperparameter convergence reached at iteration 6753.
Acquisition function convergence reached at iteration 124.
tf.Tensor(-20.107851775718647, shape=(), dtype=float64)
311
Acquisition function convergence reached at iteration 88.
tf.Tensor(-19.561203431737166, shape=(), dtype=float64)
312
Acquisition function convergence reached at iteration 79.
tf.Tensor(-19.462673838753172, shape=(), dtype=float64)
313
Acquisition function convergence reached at iteration 68.
tf.Tensor(-19.32364758321108, shape=(), dtype=float64)
314
Acquisition function convergence reached at iteration 63.
tf.Tensor(-19.268190145548175, shape=(), dtype=float64)
315
```

```
Acquisition function convergence reached at iteration 68.
tf.Tensor(-18.63768198642009, shape=(), dtype=float64)
316
Acquisition function convergence reached at iteration 66.
tf.Tensor(-18.32634105576031, shape=(), dtype=float64)
317
Acquisition function convergence reached at iteration 48.
tf.Tensor(-18.215849750014613, shape=(), dtype=float64)
318
Acquisition function convergence reached at iteration 66.
tf.Tensor(-17.902874403700913, shape=(), dtype=float64)
319
Acquisition function convergence reached at iteration 67.
tf.Tensor(-17.471403605033416, shape=(), dtype=float64)
320
Hyperparameter convergence reached at iteration 6698.
Acquisition function convergence reached at iteration 86.
tf.Tensor(-16.7134541011406, shape=(), dtype=float64)
321
Acquisition function convergence reached at iteration 77.
tf.Tensor(-16.71557101828045, shape=(), dtype=float64)
322
Acquisition function convergence reached at iteration 62.
tf.Tensor(-16.608231569324317, shape=(), dtype=float64)
323
Acquisition function convergence reached at iteration 79.
tf.Tensor(-16.53780552206311, shape=(), dtype=float64)
324
Acquisition function convergence reached at iteration 62.
tf.Tensor(-16.51463377737992, shape=(), dtype=float64)
325
Acquisition function convergence reached at iteration 69.
tf.Tensor(-16.502458830036872, shape=(), dtype=float64)
326
Acquisition function convergence reached at iteration 72.
tf.Tensor(-16.44365469523464, shape=(), dtype=float64)
327
Acquisition function convergence reached at iteration 74.
tf.Tensor(-16.45295972084391, shape=(), dtype=float64)
328
Acquisition function convergence reached at iteration 47.
tf.Tensor(-16.337989427487283, shape=(), dtype=float64)
329
```

```
Acquisition function convergence reached at iteration 64.
tf.Tensor(-16.02611266620009, shape=(), dtype=float64)
330
Hyperparameter convergence reached at iteration 8693.
Acquisition function convergence reached at iteration 78.
tf.Tensor(-15.507203160507498, shape=(), dtype=float64)
331
Acquisition function convergence reached at iteration 55.
tf.Tensor(-15.442968882844927, shape=(), dtype=float64)
332
Acquisition function convergence reached at iteration 57.
tf.Tensor(-15.293992468699233, shape=(), dtype=float64)
333
Acquisition function convergence reached at iteration 24.
tf.Tensor(-15.230166567899818, shape=(), dtype=float64)
334
Acquisition function convergence reached at iteration 55.
tf.Tensor(-15.159365400199171, shape=(), dtype=float64)
335
Acquisition function convergence reached at iteration 77.
tf.Tensor(-15.087578998630221, shape=(), dtype=float64)
336
Acquisition function convergence reached at iteration 75.
tf.Tensor(-15.097792794705668, shape=(), dtype=float64)
337
Acquisition function convergence reached at iteration 72.
tf.Tensor(-15.08694976577678, shape=(), dtype=float64)
338
Acquisition function convergence reached at iteration 81.
tf.Tensor(-14.916067694554641, shape=(), dtype=float64)
339
Acquisition function convergence reached at iteration 81.
tf.Tensor(-14.905486835719941, shape=(), dtype=float64)
340
Acquisition function convergence reached at iteration 72.
tf.Tensor(-14.410075748295647, shape=(), dtype=float64)
341
Acquisition function convergence reached at iteration 51.
tf.Tensor(-14.236310835519866, shape=(), dtype=float64)
342
Acquisition function convergence reached at iteration 57.
tf.Tensor(-14.075023087682613, shape=(), dtype=float64)
343
```

```
Acquisition function convergence reached at iteration 57.
tf.Tensor(-13.963426651881793, shape=(), dtype=float64)
344
Acquisition function convergence reached at iteration 43.
tf.Tensor(-13.834792683910031, shape=(), dtype=float64)
345
Acquisition function convergence reached at iteration 60.
tf.Tensor(-13.643950288092782, shape=(), dtype=float64)
346
Acquisition function convergence reached at iteration 96.
tf.Tensor(-15.143008582372207, shape=(), dtype=float64)
347
Acquisition function convergence reached at iteration 52.
tf.Tensor(-14.818256733556343, shape=(), dtype=float64)
348
Acquisition function convergence reached at iteration 72.
tf.Tensor(-13.985303590180205, shape=(), dtype=float64)
349
Acquisition function convergence reached at iteration 59.
tf.Tensor(-13.104411680019087, shape=(), dtype=float64)
350
Acquisition function convergence reached at iteration 75.
tf.Tensor(-12.250185023001496, shape=(), dtype=float64)
351
Acquisition function convergence reached at iteration 65.
tf.Tensor(-12.110062641884706, shape=(), dtype=float64)
352
Acquisition function convergence reached at iteration 116.
tf.Tensor(-12.705924900089517, shape=(), dtype=float64)
353
Acquisition function convergence reached at iteration 64.
tf.Tensor(-12.369883583277595, shape=(), dtype=float64)
354
Acquisition function convergence reached at iteration 71.
tf.Tensor(-12.15613720838996, shape=(), dtype=float64)
355
Acquisition function convergence reached at iteration 85.
tf.Tensor(-12.041010662236335, shape=(), dtype=float64)
356
Acquisition function convergence reached at iteration 67.
tf.Tensor(-11.964907609016986, shape=(), dtype=float64)
357
Acquisition function convergence reached at iteration 60.
```

```
tf.Tensor(-11.93600994688413, shape=(), dtype=float64)
358
Acquisition function convergence reached at iteration 59.
tf.Tensor(-11.925241004171536, shape=(), dtype=float64)
359
Acquisition function convergence reached at iteration 68.
tf.Tensor(-11.950414535354764, shape=(), dtype=float64)
360
Hyperparameter convergence reached at iteration 9979.
Acquisition function convergence reached at iteration 68.
tf.Tensor(-11.427062709871418, shape=(), dtype=float64)
361
Acquisition function convergence reached at iteration 43.
tf.Tensor(-11.212847222944887, shape=(), dtype=float64)
362
Acquisition function convergence reached at iteration 63.
tf.Tensor(-11.098850317104226, shape=(), dtype=float64)
363
Acquisition function convergence reached at iteration 69.
tf.Tensor(-11.099058051212005, shape=(), dtype=float64)
364
Acquisition function convergence reached at iteration 71.
tf.Tensor(-10.92708946767728, shape=(), dtype=float64)
365
Acquisition function convergence reached at iteration 61.
tf.Tensor(-10.766425893953627, shape=(), dtype=float64)
366
Acquisition function convergence reached at iteration 64.
tf.Tensor(-10.583275267571423, shape=(), dtype=float64)
367
Acquisition function convergence reached at iteration 52.
tf.Tensor(-10.527148860330694, shape=(), dtype=float64)
368
Acquisition function convergence reached at iteration 25.
tf.Tensor(-10.291324551199434, shape=(), dtype=float64)
369
Acquisition function convergence reached at iteration 50.
tf.Tensor(-10.23016367500615, shape=(), dtype=float64)
370
Hyperparameter convergence reached at iteration 9789.
Acquisition function convergence reached at iteration 65.
tf.Tensor(-9.810606856493543, shape=(), dtype=float64)
371
```

```
Acquisition function convergence reached at iteration 42.
tf.Tensor(-9.719670285306488, shape=(), dtype=float64)
372
Acquisition function convergence reached at iteration 64.
tf.Tensor(-9.715550747822288, shape=(), dtype=float64)
373
Acquisition function convergence reached at iteration 59.
tf.Tensor(-9.71864152847234, shape=(), dtype=float64)
374
Acquisition function convergence reached at iteration 61.
tf.Tensor(-9.51484606788403, shape=(), dtype=float64)
375
Acquisition function convergence reached at iteration 52.
tf.Tensor(-9.540985140165606, shape=(), dtype=float64)
376
Acquisition function convergence reached at iteration 52.
tf.Tensor(-9.536399753466139, shape=(), dtype=float64)
377
Acquisition function convergence reached at iteration 56.
tf.Tensor(-9.338699879133138, shape=(), dtype=float64)
378
Acquisition function convergence reached at iteration 121.
tf.Tensor(-10.462762040191915, shape=(), dtype=float64)
379
Acquisition function convergence reached at iteration 67.
tf.Tensor(-10.15981948706531, shape=(), dtype=float64)
380
Hyperparameter convergence reached at iteration 8126.
Acquisition function convergence reached at iteration 67.
tf.Tensor(-9.323609951354417, shape=(), dtype=float64)
381
Acquisition function convergence reached at iteration 69.
tf.Tensor(-9.083042756292816, shape=(), dtype=float64)
382
Acquisition function convergence reached at iteration 69.
tf.Tensor(-8.851178370253017, shape=(), dtype=float64)
383
Acquisition function convergence reached at iteration 52.
tf.Tensor(-8.394647132840603, shape=(), dtype=float64)
384
Acquisition function convergence reached at iteration 57.
tf.Tensor(-8.155209307652179, shape=(), dtype=float64)
385
```

```
Acquisition function convergence reached at iteration 48.
tf.Tensor(-7.888246682833614, shape=(), dtype=float64)
386
Acquisition function convergence reached at iteration 90.
tf.Tensor(-8.63786193752713, shape=(), dtype=float64)
387
Acquisition function convergence reached at iteration 84.
tf.Tensor(-8.658389115625866, shape=(), dtype=float64)
388
Acquisition function convergence reached at iteration 78.
tf.Tensor(-8.406657339104868, shape=(), dtype=float64)
389
Acquisition function convergence reached at iteration 75.
tf.Tensor(-8.238519842885363, shape=(), dtype=float64)
390
Hyperparameter convergence reached at iteration 6209.
Acquisition function convergence reached at iteration 75.
tf.Tensor(-7.93792736323633, shape=(), dtype=float64)
391
Acquisition function convergence reached at iteration 70.
tf.Tensor(-7.705901799031956, shape=(), dtype=float64)
392
Acquisition function convergence reached at iteration 77.
tf.Tensor(-7.667341271891395, shape=(), dtype=float64)
393
Acquisition function convergence reached at iteration 57.
tf.Tensor(-7.666428970277604, shape=(), dtype=float64)
394
Acquisition function convergence reached at iteration 64.
tf.Tensor(-7.665424892951542, shape=(), dtype=float64)
395
Acquisition function convergence reached at iteration 69.
tf.Tensor(-7.493976505458328, shape=(), dtype=float64)
396
Acquisition function convergence reached at iteration 69.
tf.Tensor(-7.500556717088816, shape=(), dtype=float64)
397
Acquisition function convergence reached at iteration 69.
tf.Tensor(-7.521140309873079, shape=(), dtype=float64)
398
Acquisition function convergence reached at iteration 63.
tf.Tensor(-7.404466285136163, shape=(), dtype=float64)
399
```

```
Acquisition function convergence reached at iteration 72.
tf.Tensor(-7.381982686292588, shape=(), dtype=float64)
[9.99998647e-01 4.00000000e-01 6.73712391e-01 4.82790814e-14
 4.63330940e-14 1.43642268e-14]
[9.20767659e-01 3.99946673e-01 4.50452627e-03 5.01265129e-14
 4.67490366e-14 1.32642656e-14]
[9.07115368e-01 3.99954490e-01 4.50397533e-03 5.01447119e-14
 4.67454524e-14 1.32114752e-14]
[9.07138978e-01 4.00083986e-01 4.50427522e-03 5.01453710e-14
 4.67451920e-14 1.32087346e-14]
```

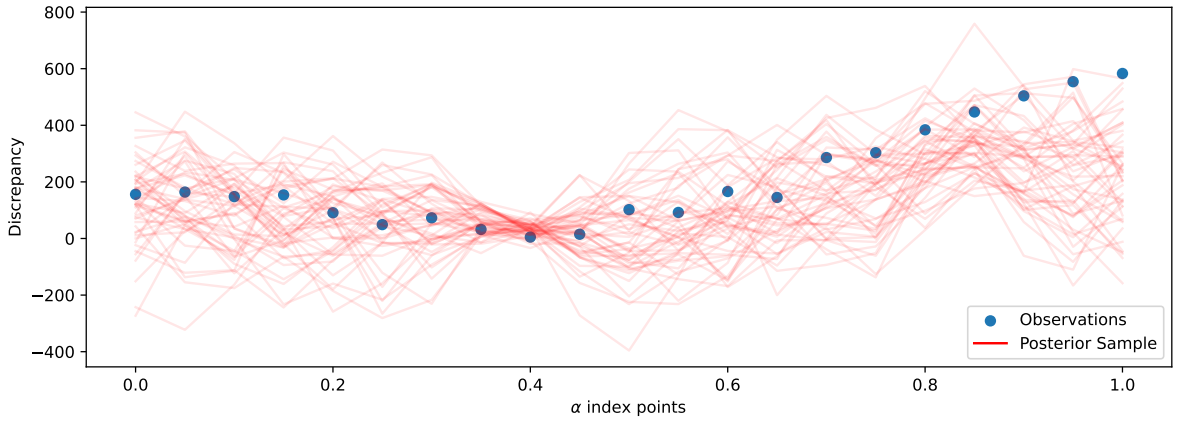
Fitting the GP Regression across alpha

```
plot_samp_no = 21
gp_samp_no = 50
```

```
GP_seed = tfp.random.sanitize_seed(4362)

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

GP_samples = champ_GP_reg.sample(gp_samp_no, seed=GP_seed)
```

```
plt.figure(figsize=(12, 4))
plt.scatter(
    alpha_slice_index_vals[:, 0], alpha_slice_discrepencies, label="Observations"
)
for i in range(gp_samp_no):
    plt.plot(
        alpha_slice_index_vals[:, 0],
        GP_samples[i, :],
        c="r",
        alpha=0.1,
```

```
        label="Posterior Sample" if i == 0 else None,
    )
leg = plt.legend(loc="lower right")
for lh in leg.legend_handles:
    lh.set_alpha(1)
plt.xlabel(r"$\alpha$ index points")
plt.ylabel("Discrepancy")
plt.savefig("champagne_GP_images/new_alpha_slice.pdf")
plt.show()
```



Fitting the GP Regression across beta

```
champ_GP_reg = tfd.GaussianProcessRegressionModel(
    kernel=kernel_champ,
    index_points=beta_slice_index_vals,
    observation_index_points=index_vals,
    observations=obs_vals,
    observation_noise_variance=observation_noise_variance_champ,
    predictive_noise_variance=0.0,
    mean_fn=quad_mean_fn(),
)

GP_samples = champ_GP_reg.sample(gp_samp_no, seed=GP_seed)
```

```
plt.figure(figsize=(12, 4))
plt.scatter(beta_slice_index_vals[:, 1], beta_slice_discrepencies, label="Observations")
for i in range(gp_samp_no):
```

```
    plt.plot(
        beta_slice_index_vals[:, 1],
        GP_samples[i, :],
        c="r",
        alpha=0.1,
        label="Posterior Sample" if i == 0 else None,
    )
leg = plt.legend(loc="lower right")
for lh in leg.legend_handles:
    lh.set_alpha(1)
plt.xlabel(r"$\beta$ index points")
plt.ylabel("Discrepency")
plt.savefig("champagne_GP_images/new_beta_slice.pdf")
plt.show()
```