

Drift scale param test

```
# Add the python helper directory to the path

import sys
import os
from pathlib import Path

CWD = Path(os.getcwd())
sys.path.append(str(CWD / "../python_helpers/"))

# Import the python helpers
import proc_output

import arviz as az
import xarray as xr
import matplotlib.pyplot as plt
import warnings
import pandas as pd

# Suppress warning from arviz
# -- it complains that all samples are the same
warnings.filterwarnings(action="ignore", module=r"arviz")
```

Data generation

I generated two datasets for 3 symbionts and 3 hosts, available in `pipeline_outputs/datadir`. The TreePPL implementation was run 3 times per dataset with different random seeds.

The two interaction matrices generated are shown below

```
pipeline_result_dir = Path("pipeline_outputs")
datadir = pipeline_result_dir / "datadir"
```

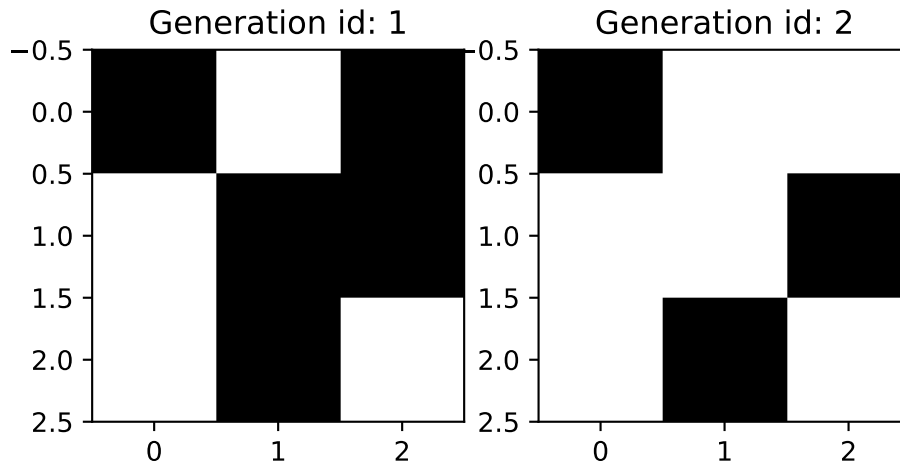
```

genids = [1, 2]
interaction_files = [datadir / f"interactions.{i}.csv" for i in genids]

dfs = [pd.read_csv(fn, index_col=0) for fn in interaction_files]
fig, axs = plt.subplots(1, len(genids))
for i, (genid, df) in enumerate(zip(genids, dfs)):
    axs[i].imshow(df.to_numpy(), cmap="Greys")
    axs[i].set_title(f"Generation id: {genid}")
fig.suptitle("Interaction matrices")
plt.show()

```

Interaction matrices



```

# Set output dirs
output_dir = pipeline_result_dir / "output"
param_comb_path = output_dir / "compile_id_to_param_comb.csv"

# Read files
df, tppl_fns = proc_output.get_files_in_dir(
    output_dir, {"tppl": proc_output.get_tppl_output_pattern()}
)

df = proc_output.create_inference_data_df(df, proc_output.read_tppl_file, 0, 1)

```

Missing simulations

The following simulations failed to finish (too great RAM requirement)

```
missing_df = proc_output.get_missing_params(df, param_comb_path)
print(missing_df)
```

	genid	compile_id	runid	drift_scale	gprob
0	1	8	3	0.10	0.0
1	1	11	3	0.01	0.0
2	1	6	3	10.00	0.0
3	1	7	3	1.00	0.0

```
df_with_compile_params = proc_output.add_compile_params(df, param_comb_path)

reduced_df = proc_output.create_multi_chain_dataset_df(
    df_with_compile_params, ["drift_scale"]
)
```

Trace plot drift=0.01

```
az.plot_trace(reduced_df.loc[0.01, "multi_channel"], compact=False)
plt.show()
```

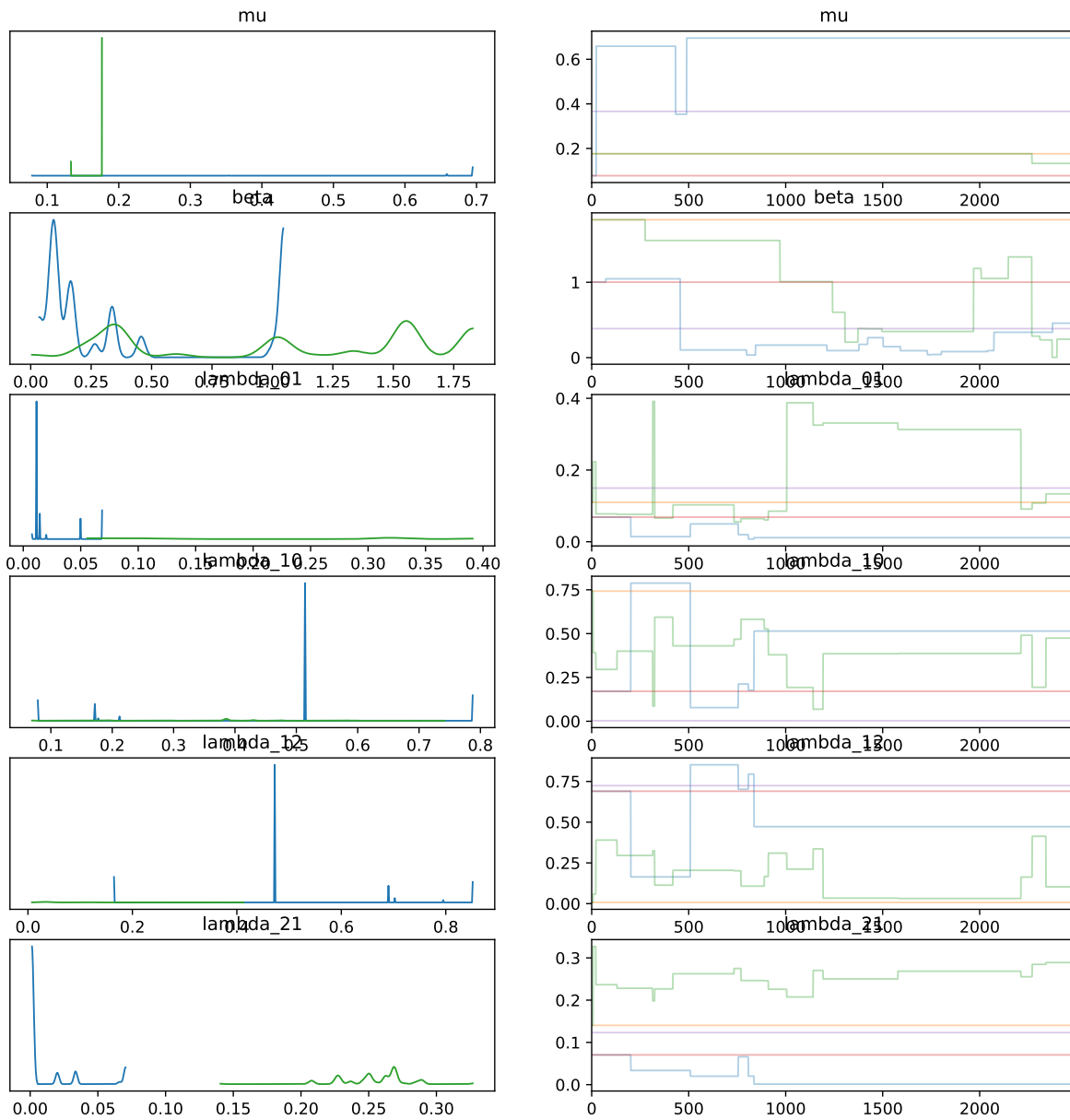


Figure 1: 2500 samples from the TPPL implementation with drift param 0.01

Trace plot drift=0.1

```
az.plot_trace(reduced_df.loc[0.1, "multi_channel"], compact=False)
plt.show()
```

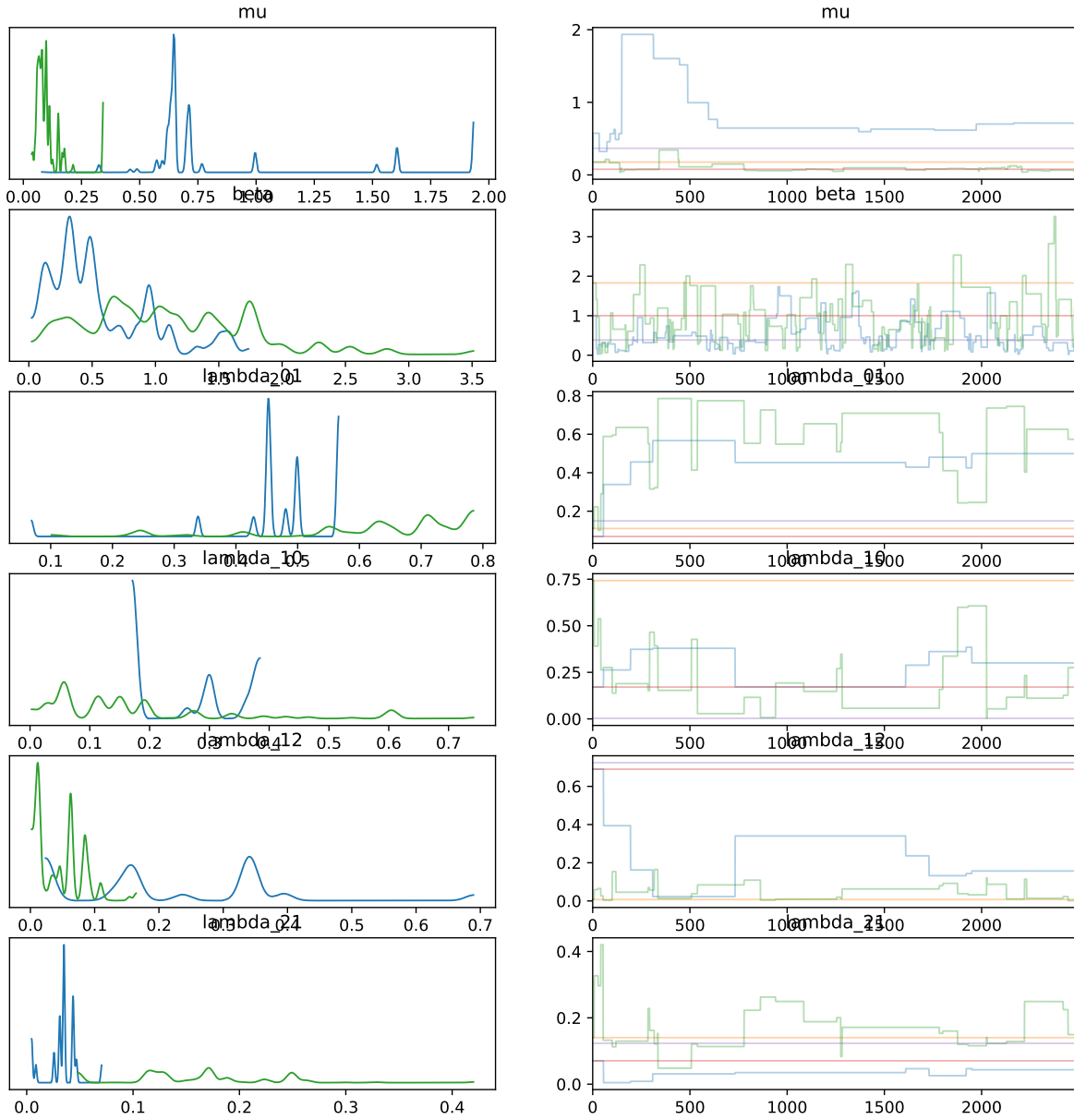


Figure 2: 2500 samples from the TPPL implementation with drift param 0.1

Trace plot drift=1

```
az.plot_trace(reduced_df.loc[1, "multi_channel"], compact=False)
plt.show()
```

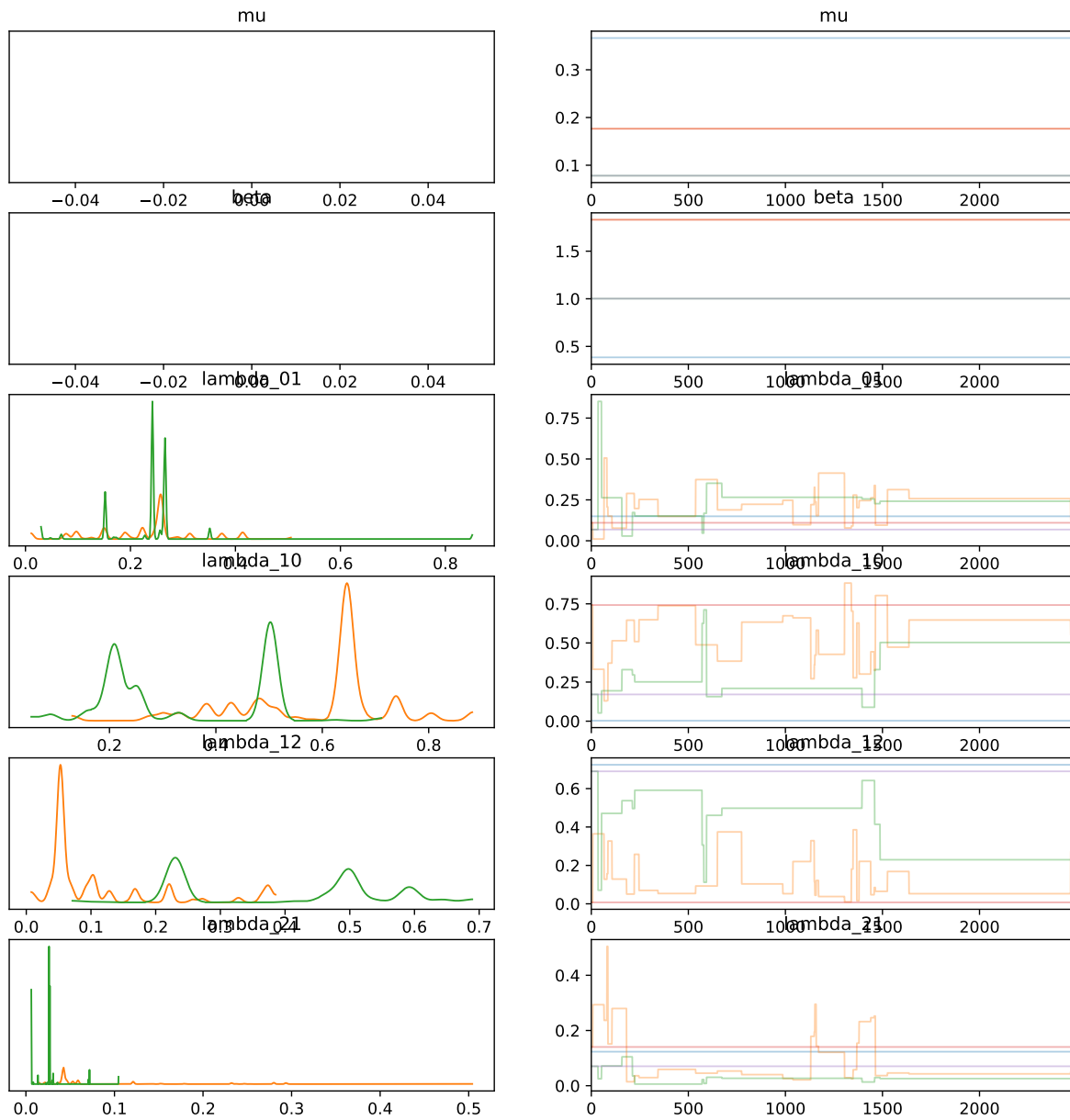


Figure 3: 2500 samples from the TPPL implementation with drift param 1

Trace plot drift=10

```
az.plot_trace(reduced_df.loc[10, "multi_channel"], compact=False)
plt.show()
```

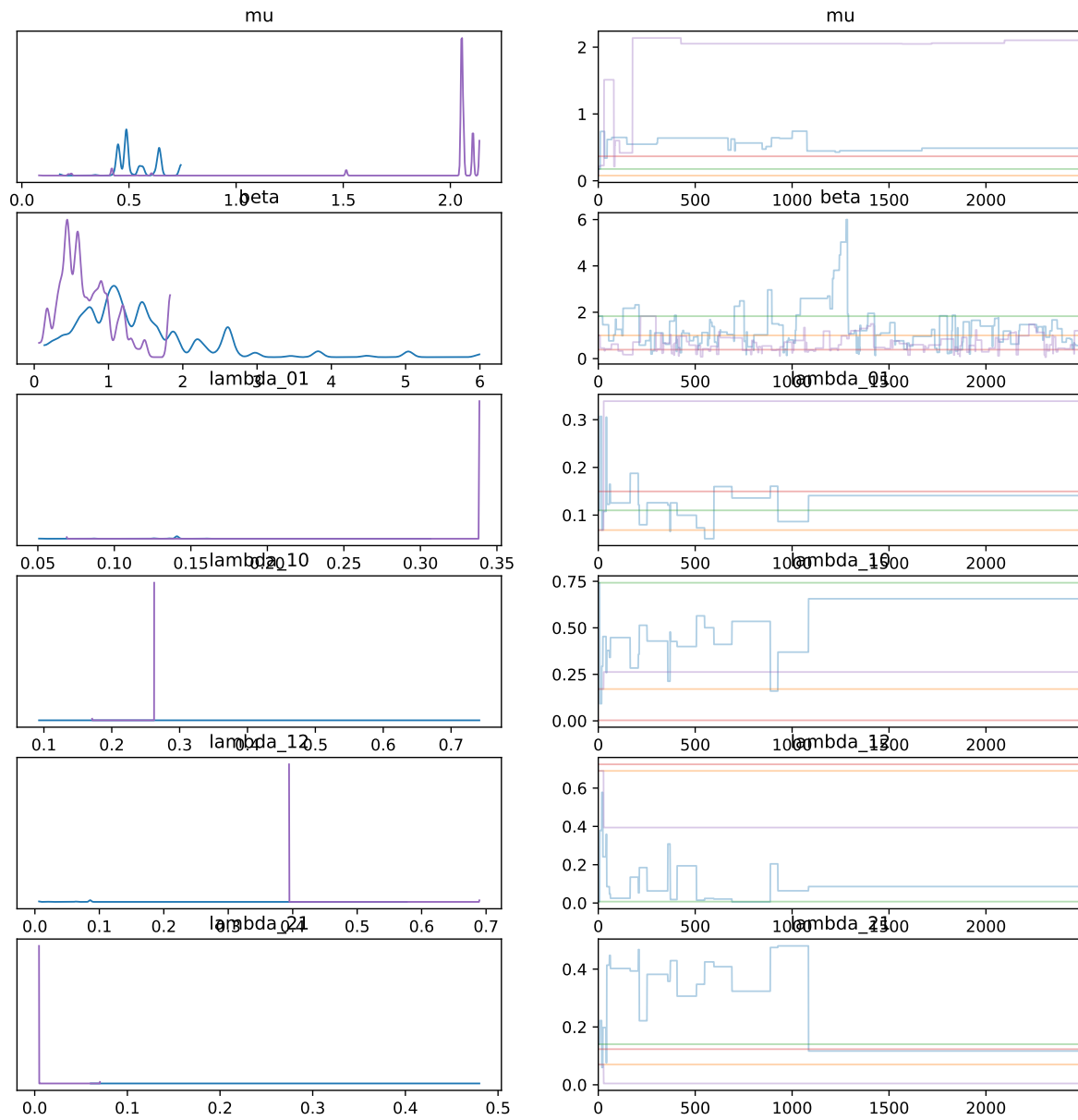


Figure 4: 2500 samples from the TPPL implementation with drift param 10