What Needs Changing

inference_utils.py

- self.one_d_only – need to update the tmnre label in the new dictionary
- self.noise_shuffling – makes reference to tmnre, but not sure what it is?
- Batch sizes, number of parameters, marginals, UNets, flattens + linear compressions – all stay the same
- self.logratios_1d – presumably here we replace with defining a density estimator using the embedding network defined earlier in the code, then SNPE inference
- Same with the 2d parts of it – keep the linear regression, need replacement with logratios
- Self.optimizer_init – potentially same, but if we are not using a swyft module, then change the sl.Adam…
- In the forward – the noise shuffling parts can remain the same, along with the d_t, d_f_w unet bits, but then no need to extract the logratios from the
- init_network – stays the same
- set_up_store – mostly stays the same, might need to change the sl.ZarrStore call if it is no longer a swyft module
- setup_dataloader – need to change some of the "tmnre" keys, but other than that should be good
- setup_trainer – replace the Tensorboard logger with WandB and potentially replace the Swyft trainer with inference and build posterior calls
- save_logratios – probably not needed?
- save_bounds, load_bounds, save_coverage – all stay the same
- UNet and LinearCompression classes – all stay the same (apart from potentially fiddling around with hyperparameters)

simulator_utils.py

- The simulator class is a swyft lightning module – might need to be changed to a regular lightning module, other than that all stays the same
- There is a swyft.to_numpy32 that might not need to be there
- Everything else looks like it should remain the same

tmnre.py

- Renaming some of the stuff at the initial printing
- Throughout there are references to conf["tmnre"] which will need changing
- The observations are generated from sl.Sample – might need changing
- All the stuff with running parallel processes and getting the simulations from zarr stores, setting up dataloaders etc should stay the same
- The trainer.fit call – replaced by inference and posterior building? Not sure if to put this at this point of the code or in the inference_utils bit – seems more likely to fit in here
- The last bit of the code – saving logratios and training on them – not needed; also there is a sl.bounds.get… that probably isn't needed either
- Replace with saving the posterior?

coverage.py

- Replace the "tmnre" keys as above
- The trainer.test and trainer.test_coverage may not work if we change the style of the trainer – need to check how these change when we have a SNPE density estimator

What Needs Changing

config_utils.py

- Need to rename the tmnre_parser
- Everything else should remain the same, unless we need to add anything into the config file

sampling.py, load_simulator.py, run_parallel.py

- Not sure anything needs changing? Some do mention import swyft.lightning as sl but no reference to it throughout