Files Added:

- **new_nvflare/**
    - fedavg_script_runner.py
        - to train the model with the federated average weighing method
    - fedopt_script_runner.py
        - Federated optimization weighing method
    - fedprox_script_runner.py
        - Federated proximal weighing method
    - scaffold_script_runner.py
        - Stochastic Controlled Averaging for Federated Learning

-> All the above files can be run through the `**script_run.sh**` file which is defined as follows:

```
#!/bin/bash

#SBATCH -c 8
#SBATCH -p gpu --gres=gpu:1
#SBATCH -o OverlapFL_exp5_fedavg-Cifar10-10-%j.out
module load container_env pytorch-gpu/2.2

crun -p /scratch/sbane002/shared/envs/new_nvflare python3 fedavg_script_runner_pt.py
-n 10 -r 1 -d CIFAR10 -a fedavg
```

-> Make sure to change the file name and the arguments passed (particularly the '--a' tag) based on the averaging method you want to use

- **new_nvflare/src**
    - pt_fl.py
        - main() and evaluate() functions defined.
        - main() - this is where the main model gets trained (training loop is defined)
        - Works for fedavg, fedopt, and fedprox
    - scaffold_pt_fl.py
        - Same architecture as pt_fly.py but only works for scaffold averaging method.
        - Make sure to invoke this file name under `training_script` in scaffold_script_runner.py, if not already defined

- **new_nvflare/src/utilities**
    - The file structure is the same as before what we were using.
    - datasets.py
        - Has all the functions to fetch dataset as well as all the dataloader functions
        - This is the file where we can change if we want to run a heterogenous experiment or overlapping experiment.
        - Accordingly, comment out the lines in function load_partitioned_datasets() (line 930 - 987) based on the number of clients and what kind of experiment you wish to run

        - Functions split_dataloader() and validate_mapping() already handle the mapping of the data across the clients, so you don't have to worry about touching those

    - models.py
        - Has all the functions to fetch the model to be used

    -> other files in this directory aren't used as much and haven't been changed

- **new_nvflare/vertical_fl**

    - vertical_fl_script_runner_pt.py
        - This is for running a vertical heterogenous FL experiment where we split the data according to the color channels and not the classes across clients.

        - Max number of clients = 3 for this experiment

- **new_nvflare/vertical_fl/utilities**

    - datasets.py
        - Is the same as previous datasets.py. Only difference is the addition of function load_partitioned_datasets_vertical() - which helps distribute the dataset vertically.
        - This is the function that needs to be invoked whenever experimenting with vertical FL