Project Guide (HW1)

Project Description link: HW1 Description

Files and Their Functions

1. **deep_networks.py:**
   - Contains the Python code for building deeper neural network models.
2. **batch.sh:**
   - This is a shell script that configures and submits a job array to a SLURM-managed computing cluster. It sets the computational resources, loads the TensorFlow environment, activates a deep neural network (DNN) conda environment, and executes the Python script `hw1_base_skel.py` with specified parameters.
3. **hw1_base_skel.py:**
   - The main Python script that sets up and runs the BMI prediction model. It accepts command-line arguments to set hyperparameters and manage the execution of the experiments.
4. **job_control.py:**
   - A Python script that probably helps in managing the execution of multiple experiments by iterating over a Cartesian product of hyper-parameter combinations.
5. **plots.py:**
   - A Python script intended for generating the visualizations required for the project, such as the prediction accuracy plot and the FVAF versus training folds plot.
6. **"HW1_17789223_358_stdout.txt" and "HW1_17789224_359_stdout.txt":**
   - Output files from the SLURM job array, likely containing logs from the execution of the experiment scripts, indicating the absence of GPUs and the total number of jobs processed.
7. **HW1_report.pdf:**
   - A PDF document that includes the report for the homework, which should contain the figures and a write-up explaining the results.

Graphs and Their Meanings

1. **Fig_1.png:**
    - Represents the prediction accuracy of the model. The blue solid line indicates the actual elbow acceleration, and the orange dashed line shows the predicted elbow acceleration by the neural network. This figure is important to evaluate the time-based performance of the model, showing how closely the predictions follow the true values.
2. **Fig_2.png:**
    - Displays the average Fraction of Variance Accounted For (FVAF) as a function of the number of training folds. The blue line represents the training FVAF, the orange line is the validation FVAF, and the green line shows the testing FVAF. This graph is key to understanding how the model's performance varies with different amounts of training data, indicating potential overfitting or underfitting.

The project's completion involves setting up a series of experiments with different training set sizes and rotations, running them on a computing cluster, and then analyzing the results to understand the prediction performance and the effect of training data volume on model accuracy. The figures are a critical component, visually summarizing the outcomes of these experiments.