

Нарештій дійшов до того що можна змінювати стан моделі з тренувального в тестувальний і назад після обробки кожної епохи

```
model.train()
```

```
model.eval()
```

Тим сами для 20 епох заранив скрипт для різних функцій активації

- Сигміїда
- ELU

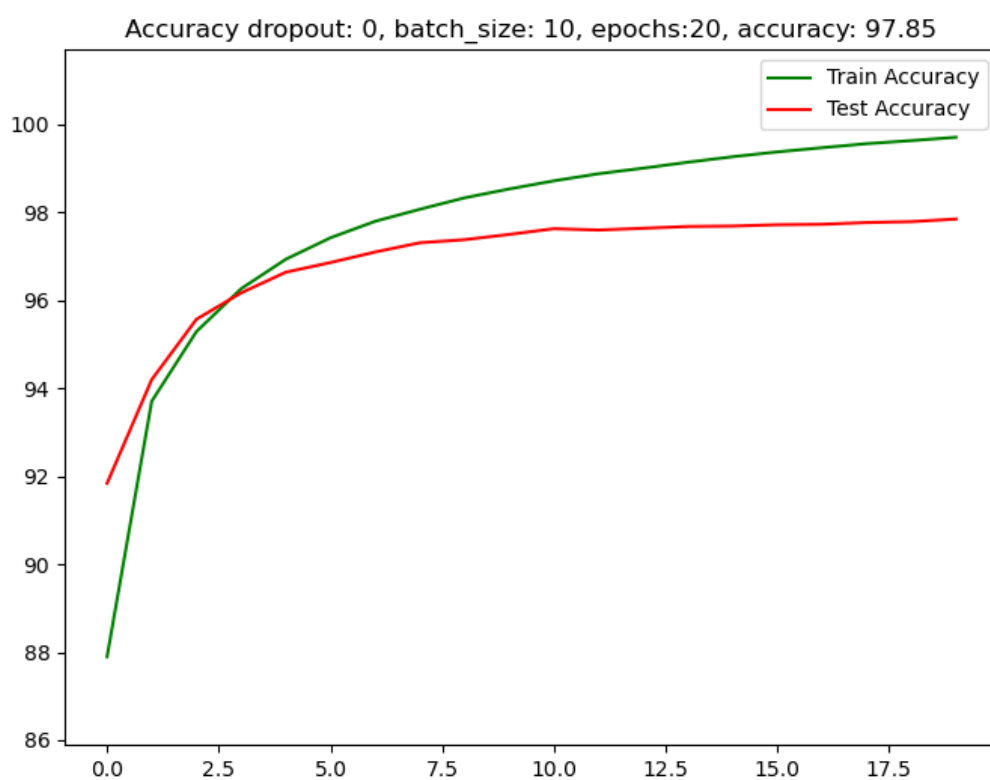
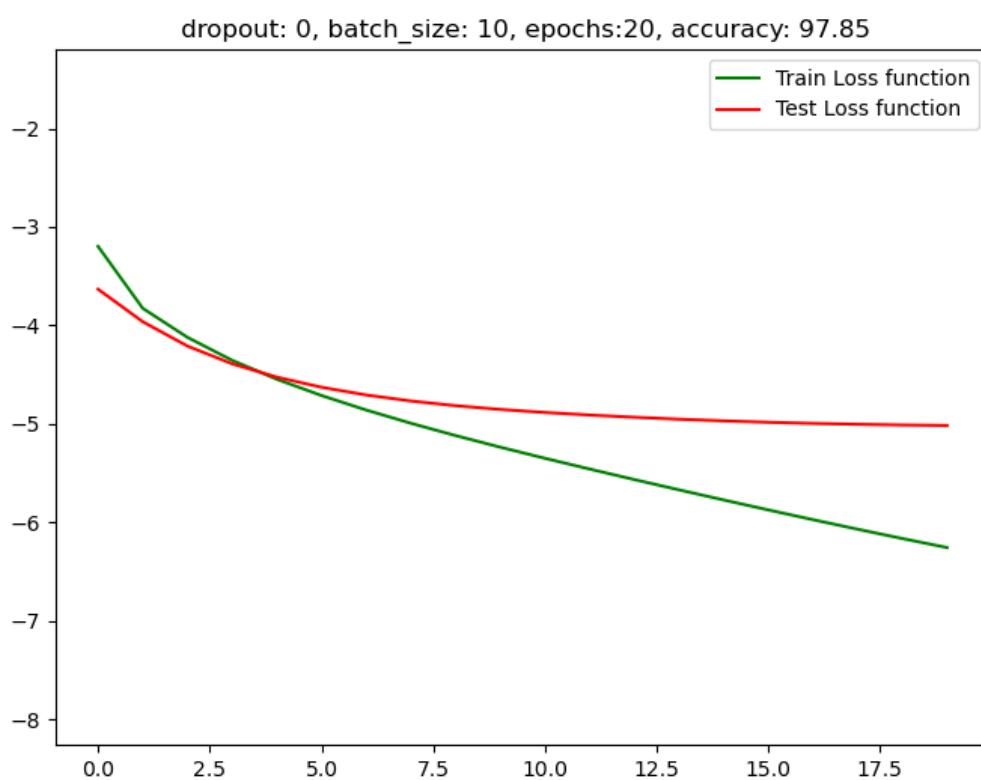
Для різних дропаутів

- 0%
- 10%
- 20%

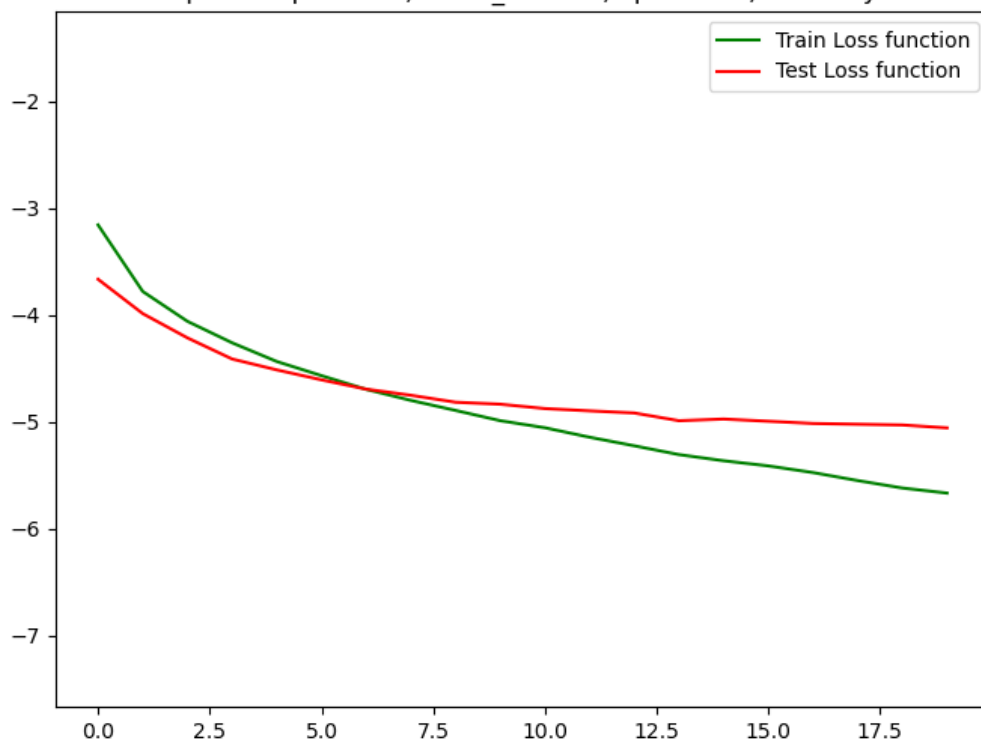
З функцією батч нормалізації

З наведених комбінацій сигміїда виявилась більш плавною оптимальною. Виявилось покращення результатів при використанні дропаута, найбільше стрибок від 0 до 10%, значно менший між 10 та 20%

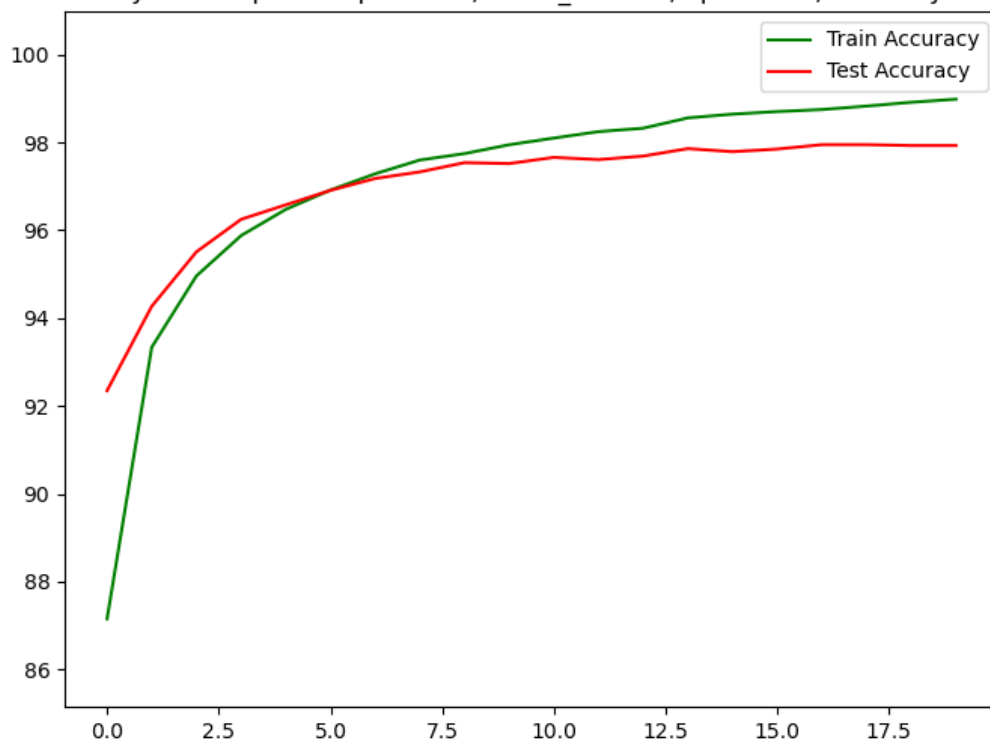
Функція батч нормалізації суттєво не результати не покращила.



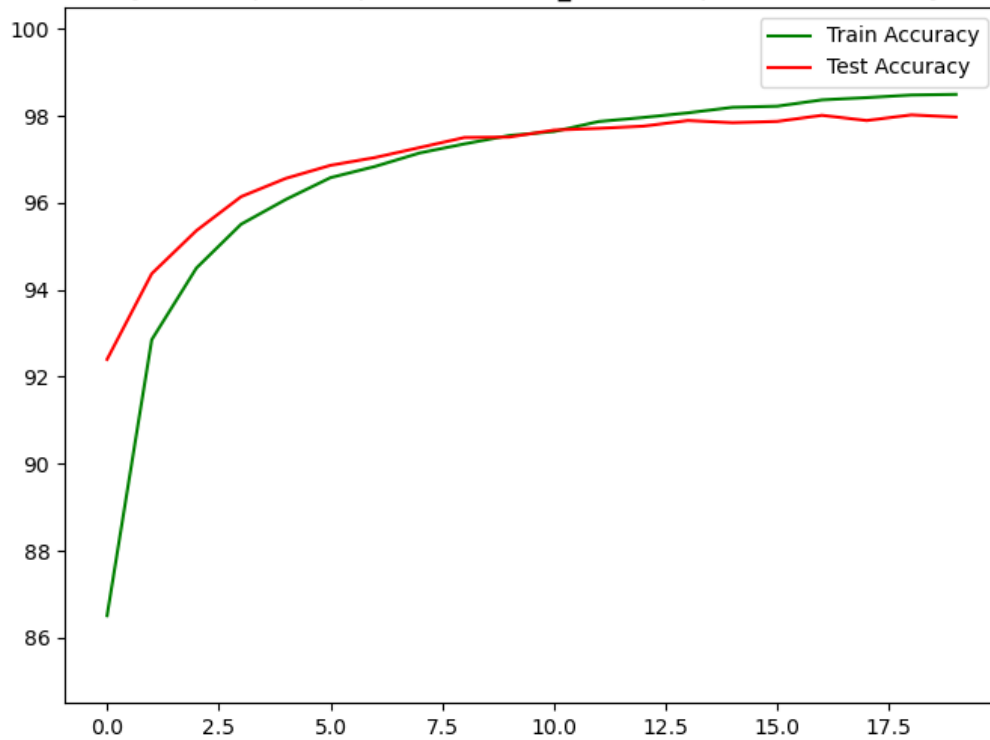
With dropout dropout: 0.1, batch_size: 10, epochs:20, accuracy: 97.93



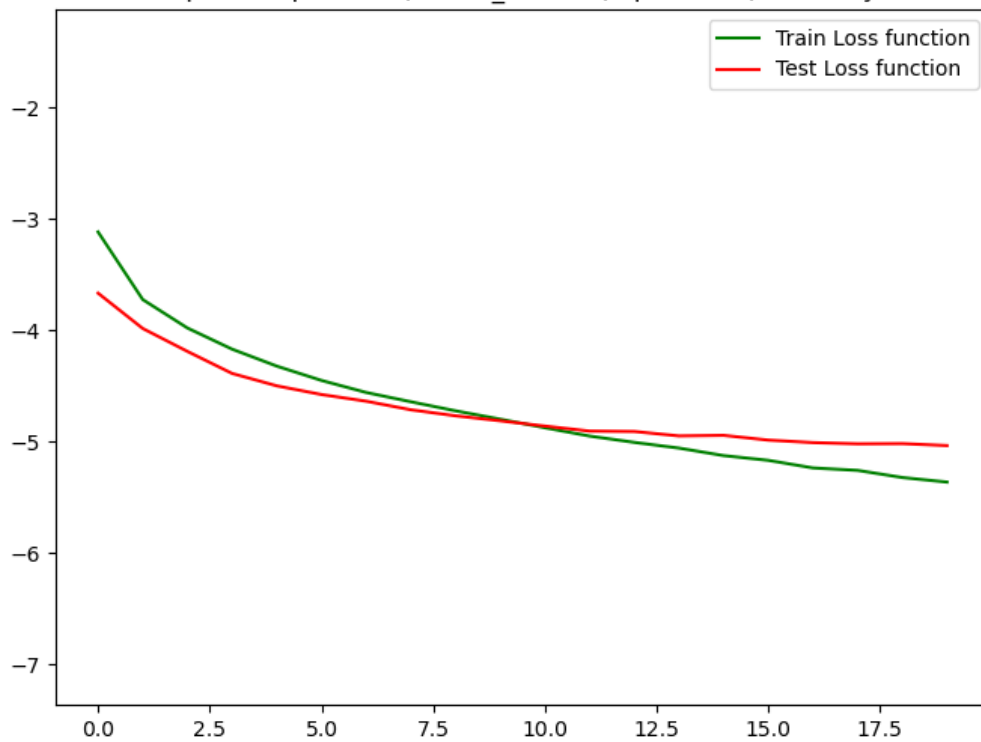
AccuracyWith dropout dropout: 0.1, batch_size: 10, epochs:20, accuracy: 97.93



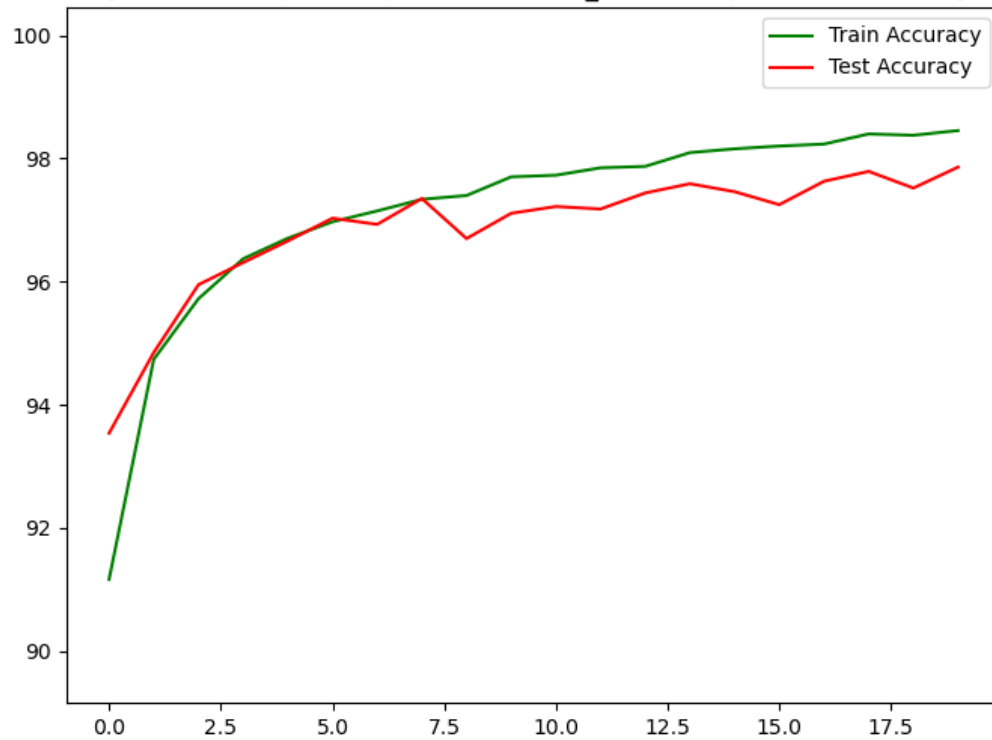
AccuracyWith dropout dropout: 0.2, batch_size: 10, epochs:20, accuracy: 97.97



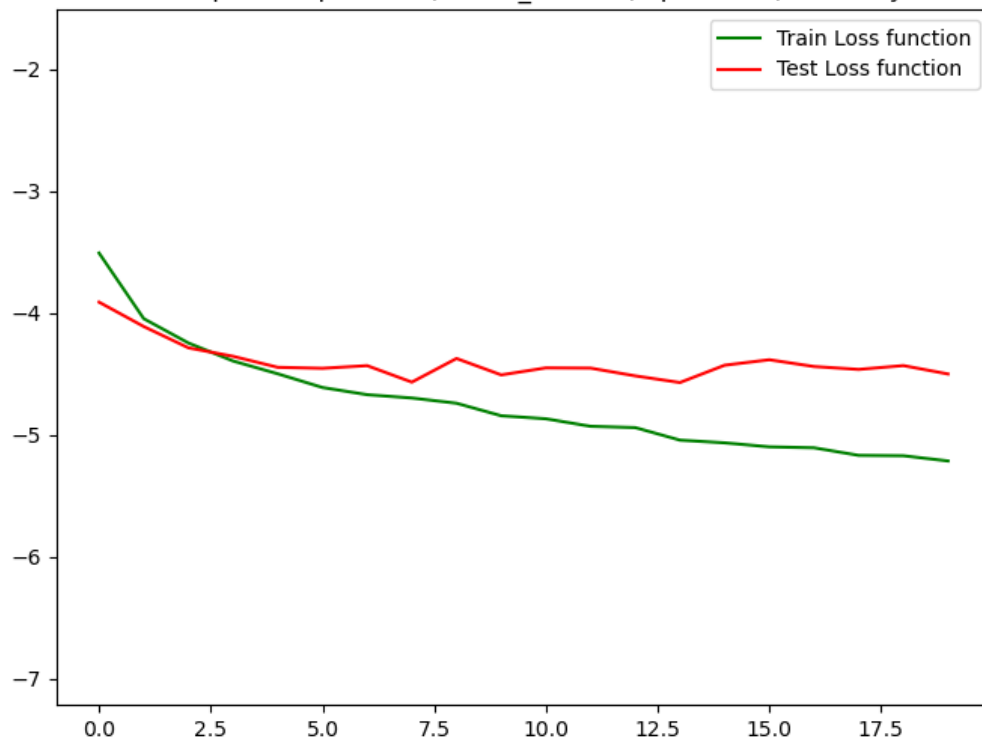
With dropout dropout: 0.2, batch_size: 10, epochs:20, accuracy: 97.97



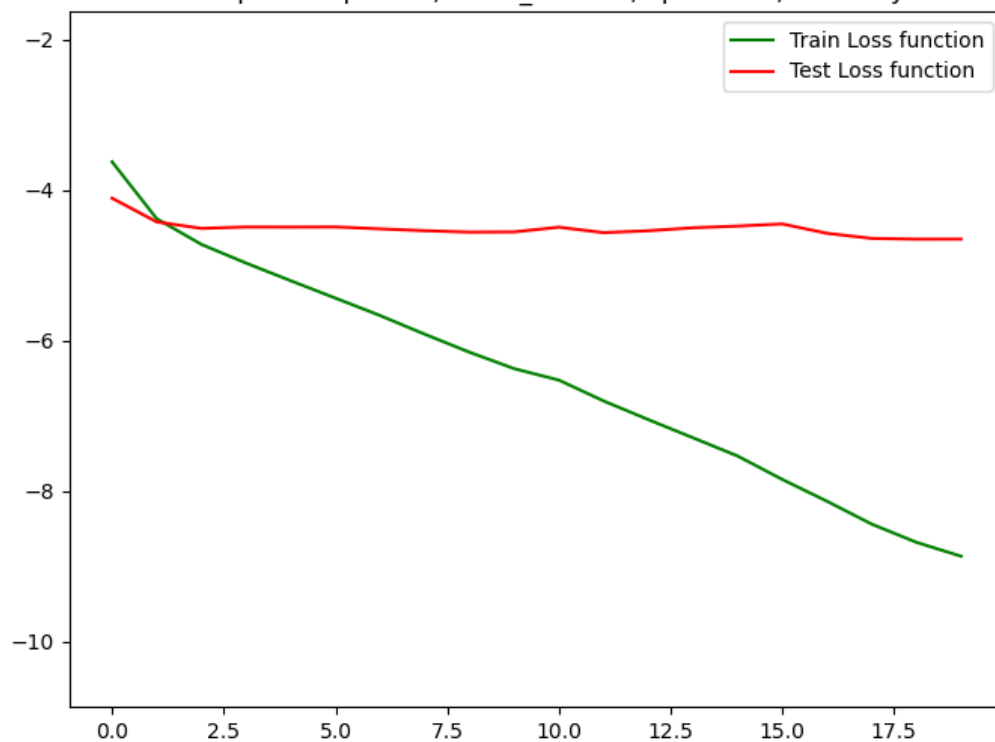
AccuracyELU with dropout dropout: 0.2, batch_size: 10, epochs:20, accuracy: 97.86



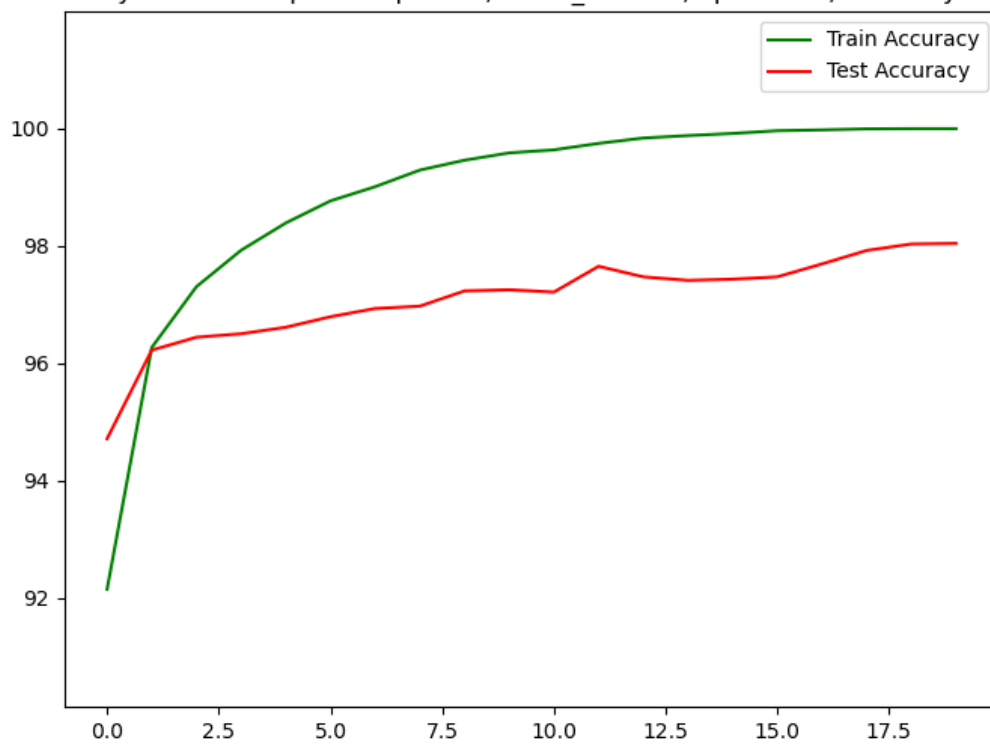
ELU with dropout dropout: 0.2, batch_size: 10, epochs:20, accuracy: 97.86



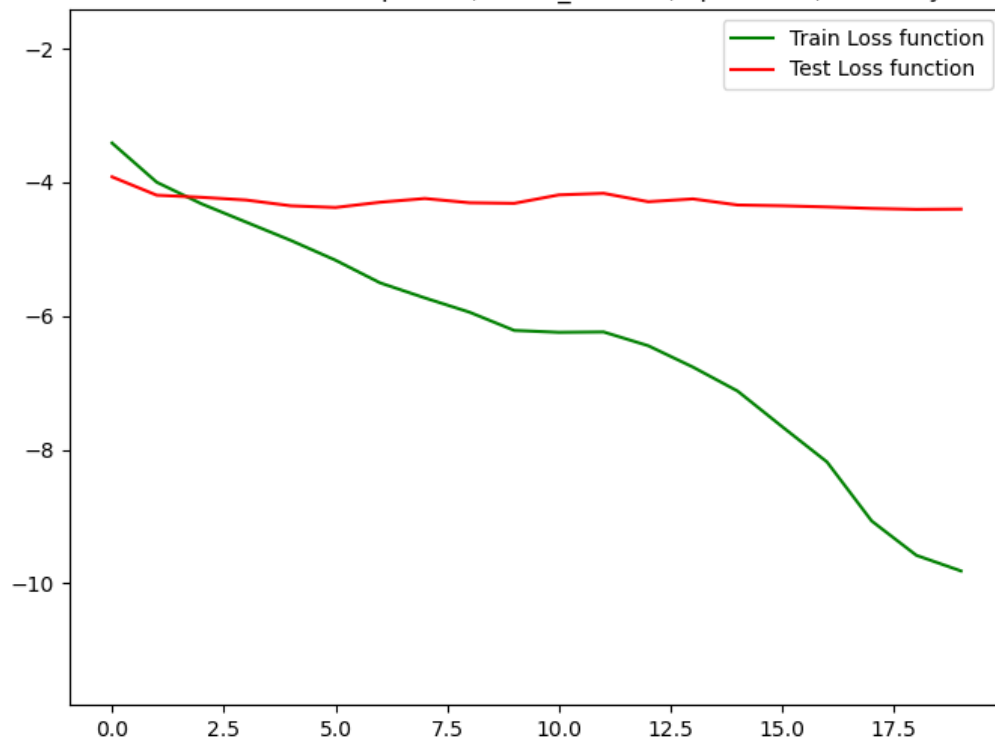
ELU with dropout dropout: 0, batch_size: 10, epochs:20, accuracy: 98.04



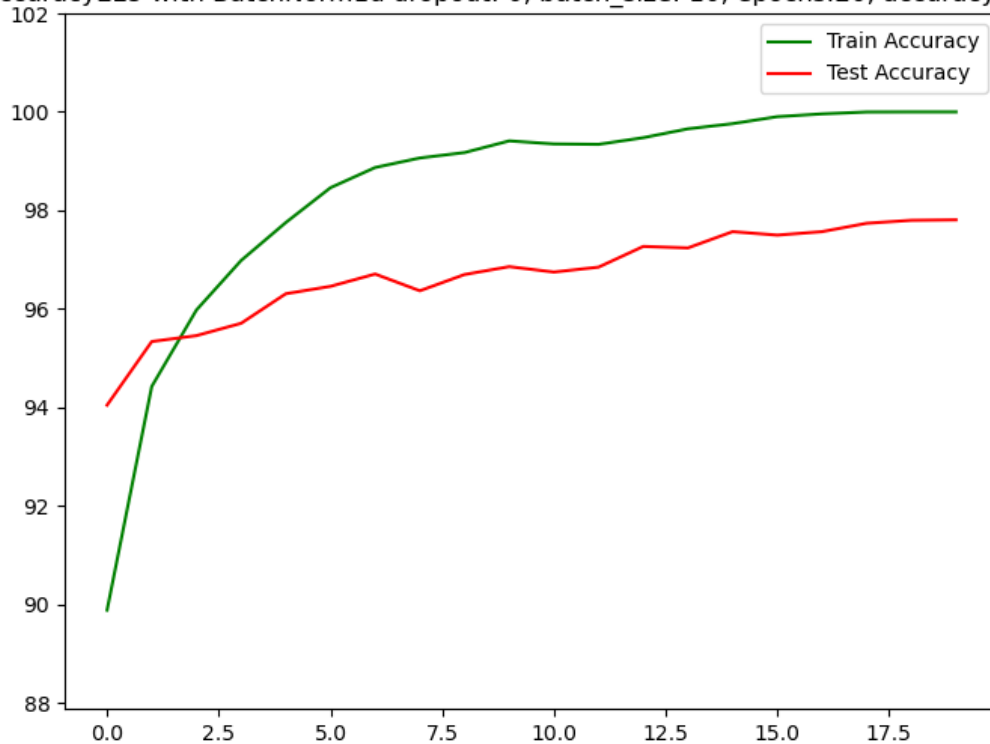
AccuracyELU with dropout dropout: 0, batch_size: 10, epochs:20, accuracy: 98.04



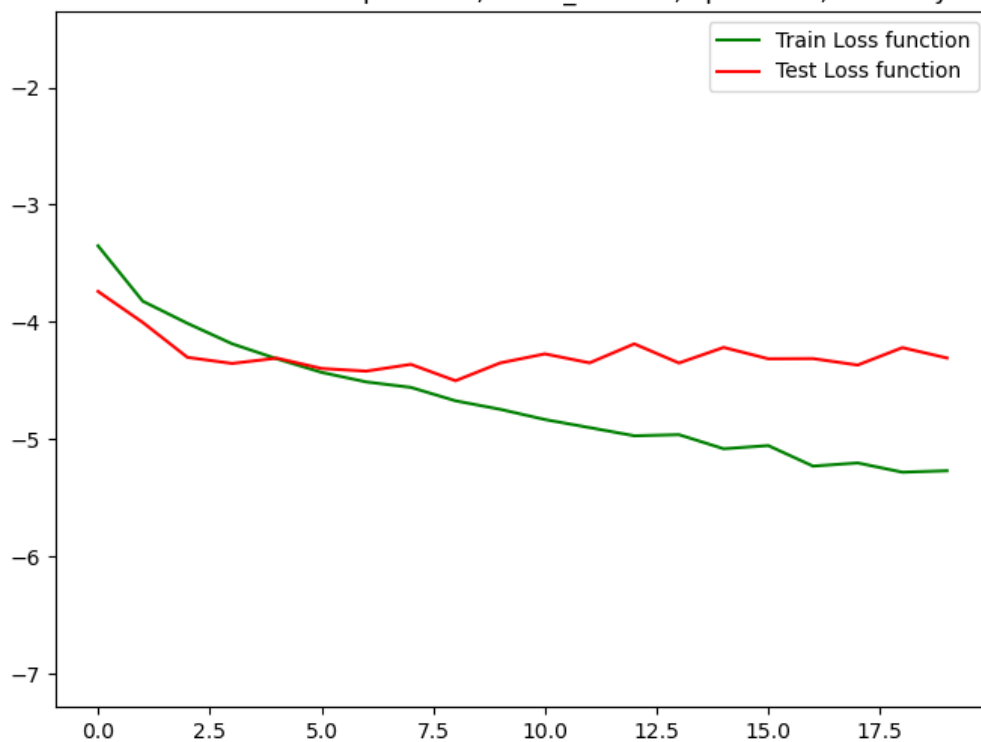
ELU with BatchNorm1d dropout: 0, batch_size: 10, epochs:20, accuracy: 97.81



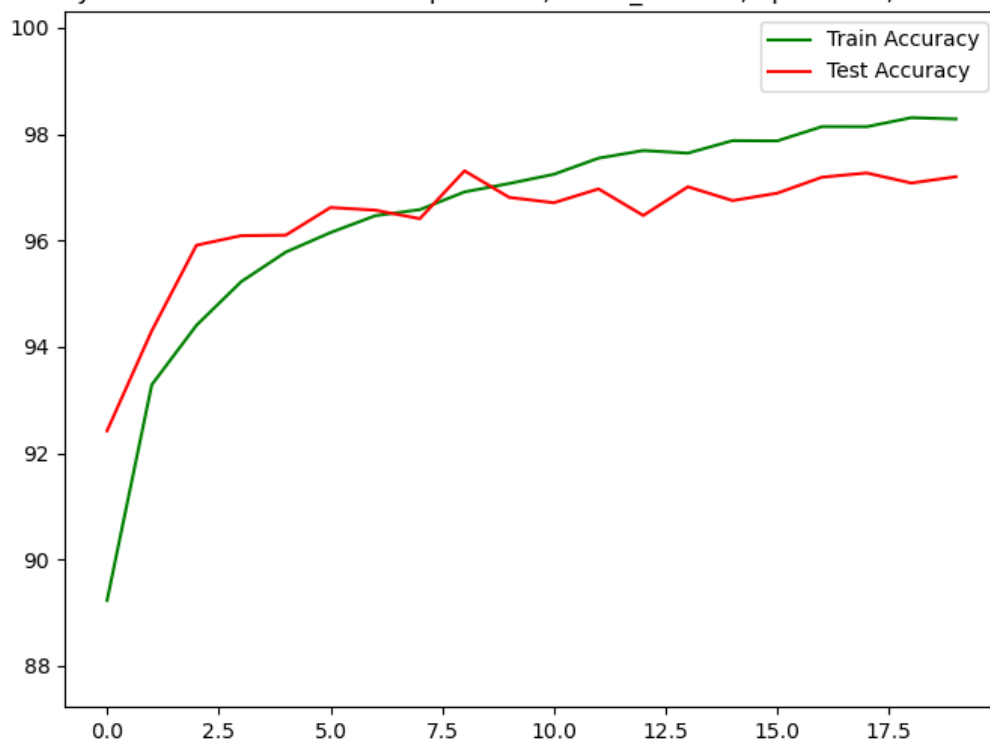
AccuracyELU with BatchNorm1d dropout: 0, batch_size: 10, epochs:20, accuracy: 97.81



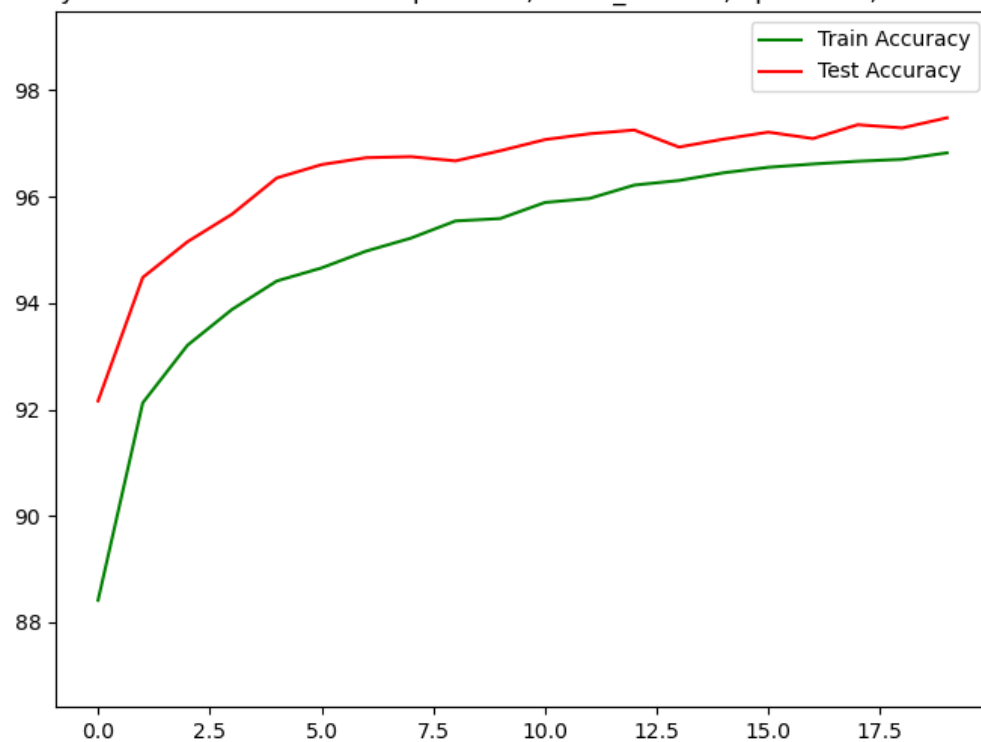
ELU with BatchNorm1d dropout: 0.1, batch_size: 10, epochs:20, accuracy: 97.2



AccuracyELU with BatchNorm1d dropout: 0.1, batch_size: 10, epochs:20, accuracy: 97.2



AccuracyELU with BatchNorm1d dropout: 0.2, batch_size: 10, epochs:20, accuracy: 97.48



ELU with BatchNorm1d dropout: 0.2, batch_size: 10, epochs:20, accuracy: 97.48

