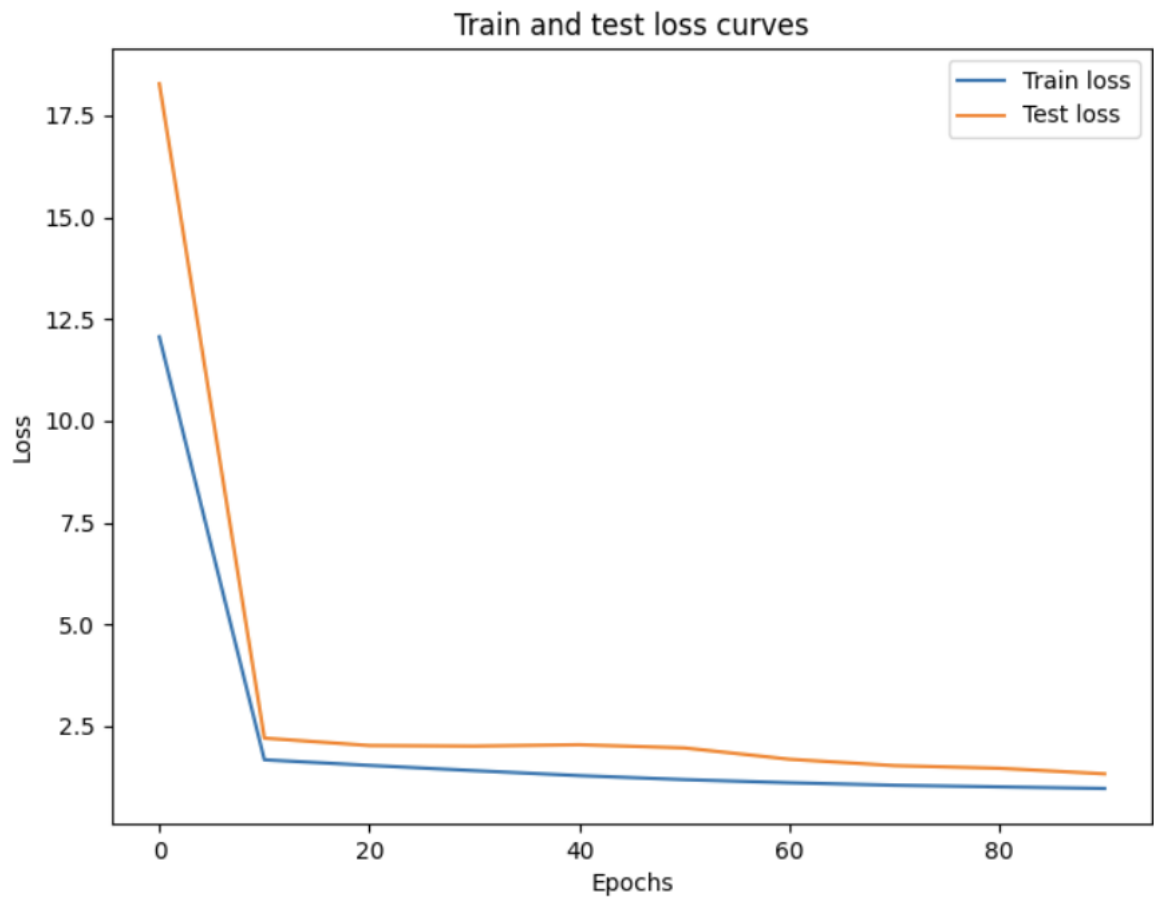


3BIT

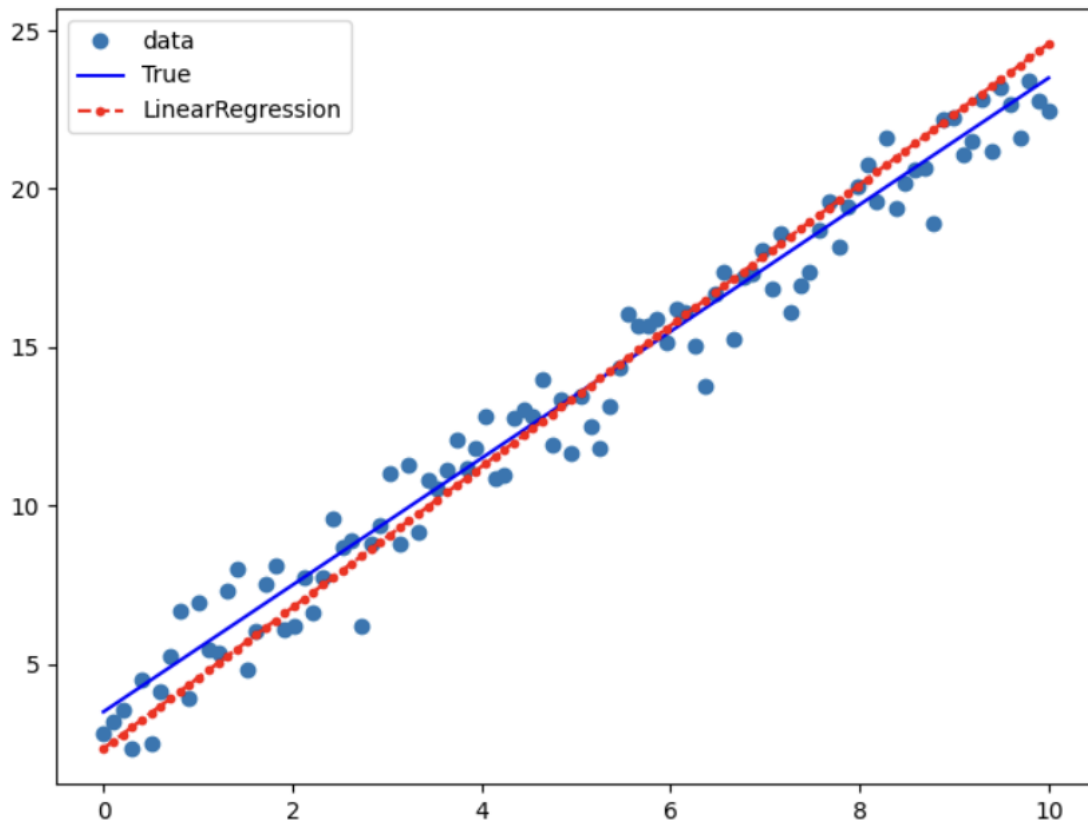
- Plot loss curve (both train and test)



- Print model weights (before and after training) and compare with original ones

Початкові значення weights = 0.0401 bias = -0.9005	На треновані значення weights = 2.2234 bias = 2.3445	Справжні значення weights = 2 bias = 3.5
--	--	--

- Plot results (predicted line along with dataset data)



Висновки: після тренування моделі, лінійна регресія апроксимує дані, але не найкращим чином. Знайдене значення для ваг (weights) (2.2234) досить близьке до справжнього значення (2). Однак, знайдений зсув (bias) (2.3445) далекий від справжнього значення (3.5), що свідчить про те, що модель не змогла визначити зсув для лінійної апроксимації.

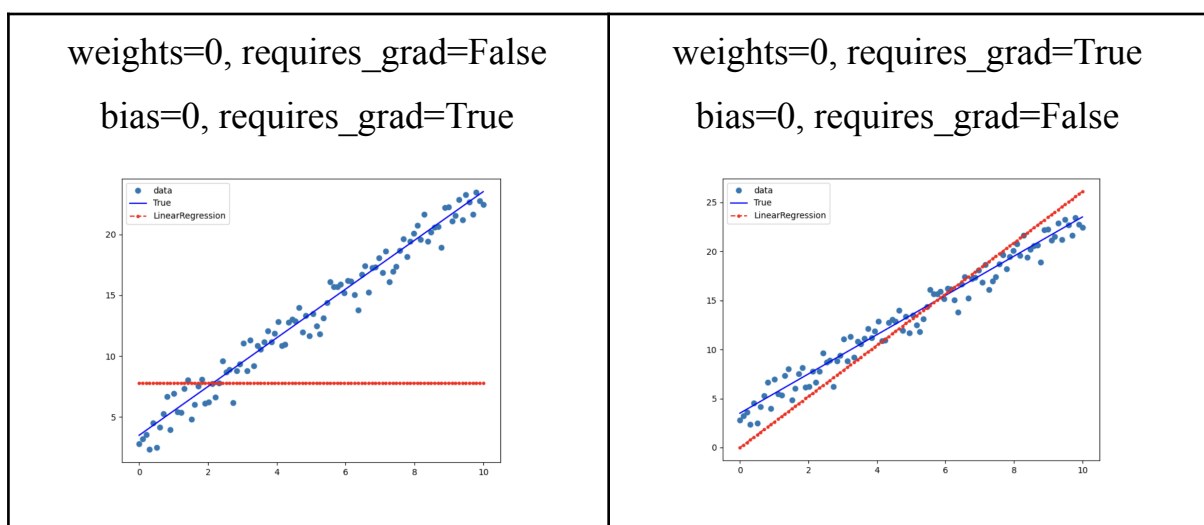
Для отримання більш точних результатів, можна поекспериментувати з параметрами тренування, такими як кількість епох (epochs) та швидкість навчання (learning rate - lr). Експериментування з цими параметрами допоможе досягти кращої апроксимації та зменшити похибку моделі.

Optional:

- How predicted line will change if `self.bias = nn.Parameter(0, requires_grad=False)`?
- What will be the value of `self.bias` if `self.weights = nn.Parameter(0, requires_grad=False)`?

Гіпотеза: при встановленні `requires_grad=False` в `self.weights` (`self.bias`), отримаємо графік лінійної регресії з параметром `weights` (`bias`), який є незмінним в кожній еPOCH. Тобто, графік матиме нахил (`weights`), зсув (`bias`) ініціалізований на початку і не змінюватиме свого значення на протязі всього навчання. Очікується, що лінійна регресія погано апроксимує наші дані.

Перевірка:



Висновки: з графіків видно, що гіпотеза підтвердилась. На другому графіку бачимо дещо кращу картину за рахунок більш-менш вдало підбраного початкового значення для `bias`. Однак, такий результат є скоріше виключенням, аніж загальним правилом.