loss_funcs.py

```python
import numpy as np

class quadratic_loss(object):
    @staticmethod
    def calc(x,y):
        return (y-x)**2

    @staticmethod
    def calc_gradient(x,y):
        """calculate gradient of the loss wrt x
            x ([type]): [description]
            y ([type]): [description]

        Returns:
            gradient wrt x ie returns dLoss_dx
        """
        return 2 * (y-x)

    # @staticmethod
    # def calc_gradient_wrt_nu(x,y):
    #     """calculate gradient of the loss
    #         x ([type]): [description]
    #         y ([type]): [description]

    #     Returns:
    #         gradient wrt x
    #     """
    #     return 2 * (y-x)

class absolute_loss(object):
    @staticmethod
    def calc(x,y):
        return np.abs(y-x)

    @staticmethod
    def calc_gradient(x,y):
        """calculate gradient of the loss
            x ([type]): [description]
            y ([type]): [description]

        Returns:
            gradient wrt x
```

```
    """
    return np.sign(y-x)
```