

Date.....

Writing my own class

#def class I will made as a Custom class.

class Meta_Ridge-TwoD:

```
def __init__(self, alpha):
```

self.alpha = alpha

~~self.m = None~~

$$m = \sum_{i=1}^n (Y_i - \bar{Y})(X_i - \bar{X}) \rightarrow \text{num}$$

$$\sum_{i=1}^n (X_i - \bar{X})^2 + \lambda \rightarrow \text{denom}$$

```
def fit(self, X_train, Y_train):
```

Initially setting the values of num and deno = 0.

$$n_{\text{eff}}m = 0$$

for i in range(X_train.shape[0]):

$h_{\text{out}} = (\text{num} + (\text{Y_train}[\text{i}] - \text{Y_train}\cdot\text{mean}()) * (\text{X_train}[\text{i}] - \text{X_train}\cdot\text{mean}()))$

$$d_{\text{new}} = \left(d_{\text{new}} + (X_{\text{train}}[i] - X_{\text{train}} \cdot \text{mean}()) * \right. \\ \left. (X_{\text{train}}[i] - X_{\text{train}} \cdot \text{mean}()) \right)$$

$$\text{self.m} = (\text{num} - (\text{zero} + \text{self.alpha}))$$

self.b = Y_train.mean() - (self.m * X_train.mean())

```
def predict(self, X-test)
```

return $(\text{self}.\text{m} * X_{-t}) + \text{self}.b$)