# ElasticNet Regression

Sunday 14 April 2024 1:33 AM

→ Combination of Ridge & Lasso

→ Helpful when dealing with high dimensionality data and not sure if all features are important or not in predicting the target.

$$L = \sum (y_i - \hat{y}_i)^2 + a\|\beta\|^2 + b\|\beta\|$$

$\lambda = a+b$ → hyperparameters

$l1\_ratio = \frac{a}{a+b}$

default $\lambda = 1$, $l1\_ratio = 0.5$

$l1\_ratio = a/\lambda$

$a = \lambda \, l1\_ratio$

$b = \lambda - a$

→ Also helpful when dealing with multicolinearity

*class* `sklearn.linear_model.`**`ElasticNet`**(*alpha=1.0, *, l1_ratio=0.5, fit_intercept=True, precompute=False, max_iter=1000, copy_X=True, tol=0.0001, warm_start=False, positive=False, random_state=None, selection='cyclic'*) ¶ [source]

Linear regression with combined L1 and L2 priors as regularizer.

Minimizes the objective function:

```
1 / (2 * n_samples) * ||y - Xw||^2_2
+ alpha * l1_ratio * ||w||_1
+ 0.5 * alpha * (1 - l1_ratio) * ||w||^2_2
```

If you are interested in controlling the L1 and L2 penalty separately, keep in mind that this is equivalent to:

```
a * ||w||_1 + 0.5 * b * ||w||_2^2
```

where:

```
alpha = a + b and l1_ratio = a / (a + b)
```

The parameter l1_ratio corresponds to alpha in the glmnet R package while alpha corresponds to the lambda parameter in glmnet. Specifically, l1_ratio = 1 is the lasso penalty. Currently, l1_ratio <= 0.01 is not reliable, unless you supply your own sequence of alpha.