The Learning Penalty

vs.

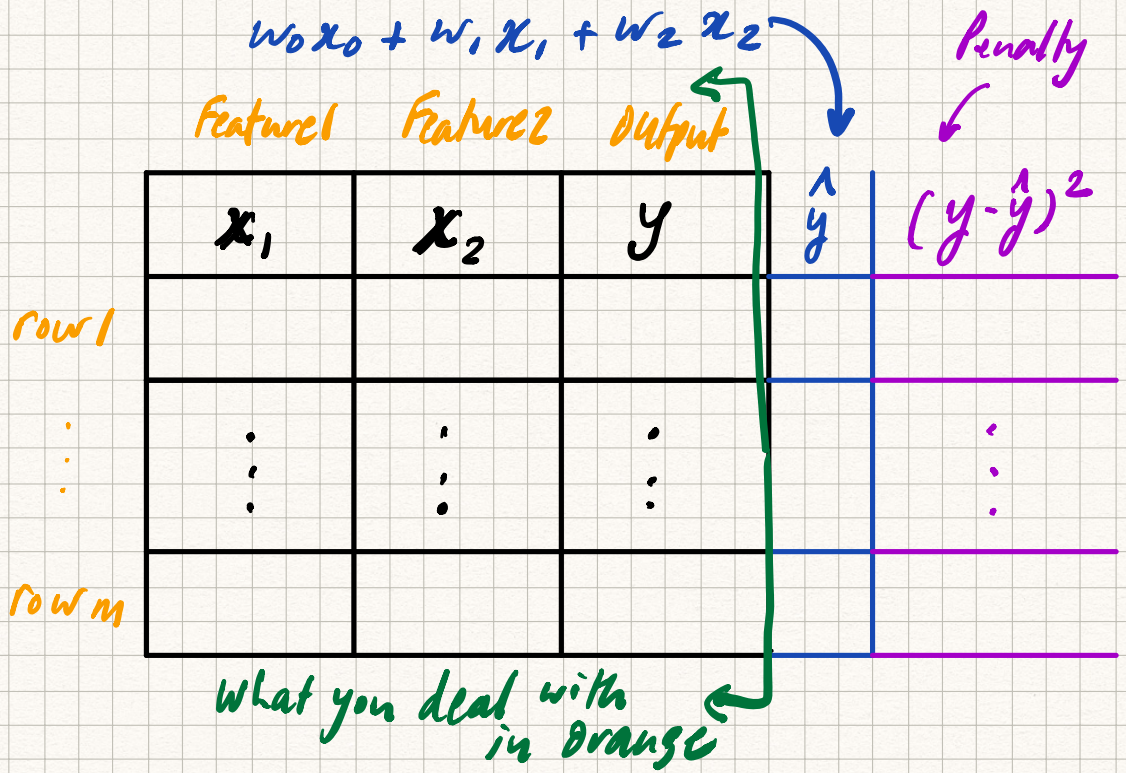
Measuring Prediction Performance

Similar in form but

Different in function.

The Learning Penalty

(aka the Penalty Function)



Cost = Sum of all penalties

(add up the value of each row)

More precisely, Cost = $\frac{1}{m}$ (Sum of all penalties)

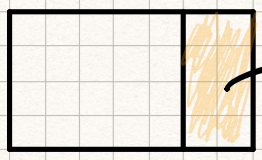
this makes it the average penalty

How Learning Happens

Iteration 1

usually just randomly chosen values

$w_0 = 0.5,\ w_1 = 2.2,\ w_2 = -1.8$



→ Penalty $(y - \hat{y})^2$ for each row

Add up the values

Divide by m (# of rows)

↓

Cost of iteration

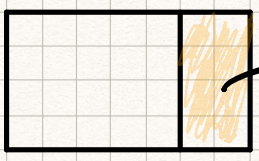
Goal: Find the next set of w_0, w_1, w_2 that lower this cost.

Gradient descent is the method for doing this.

How Learning Happens (Contd.)

Iteration 2

$w_0 = 0.52$, $w_1 = 2.1$, $w_2 = -1.92$



→ Penalty $(y - \hat{y})^2$ for each row

Add up the values

Divide by m (# of rows)

↓

Cost of iteration

Goal: Find the next set of w_0, w_1, w_2 that lower this cost.

↓

Iteration 3, 4, 5, 6, ..., 23

We can choose to stop after a certain # of iterations, or when cost isn't going down by a lot any more.

Summary - For each iteration,

$$\text{Penalty} = (y - \hat{y})^2 \text{ for each row}$$

$$\text{Cost} = \frac{1}{m} \text{Sum} \left[(y - \hat{y})^2 \right]$$

m = # rows in the dataset

Sum [...] = Average or Mean Penalty per iteration

Cost = "Mean Squared" Penalty

Once we get to the last iteration, we have the optimal values for w_0, w_1, and w_2.

Prediction

What

I give you feature values x_1, x_2.

You tell me what the output is.

How

Build a model:

- Take the feature values x_1, x_2.
- Get the optimal values of w_0, w_1, w_2.

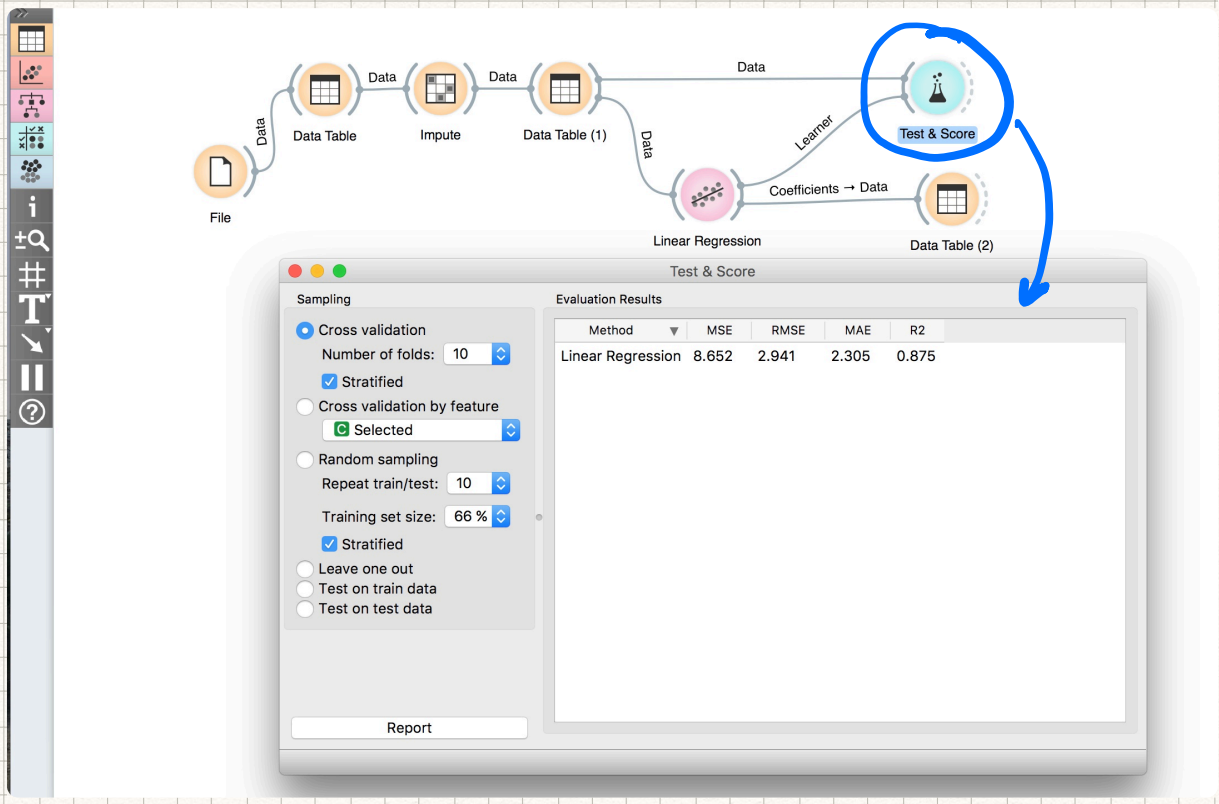
Predicted Output $= (w_0 * 1) + (w_1 * x_1) + (w_2 * x_2)$

(1: always $= 1$)

Use the model

Prediction (contd.)

How good is the prediction?



MSE = Mean Square Error

RMSE = Root Mean Square Error

MAE = Mean Absolute Error

R2 = R^2 = "R-Squared"

Always a good idea to have a single number (if at all possible) to measure how good your predictions are.

MSE, RMSE, MAE and R^2 are such numbers. Any <u>one</u> of them can be used to measure how good the predictions are.

The lower the value, the better for MSE, RMSE, & MAE.

MSE - Mean Square Error

Features

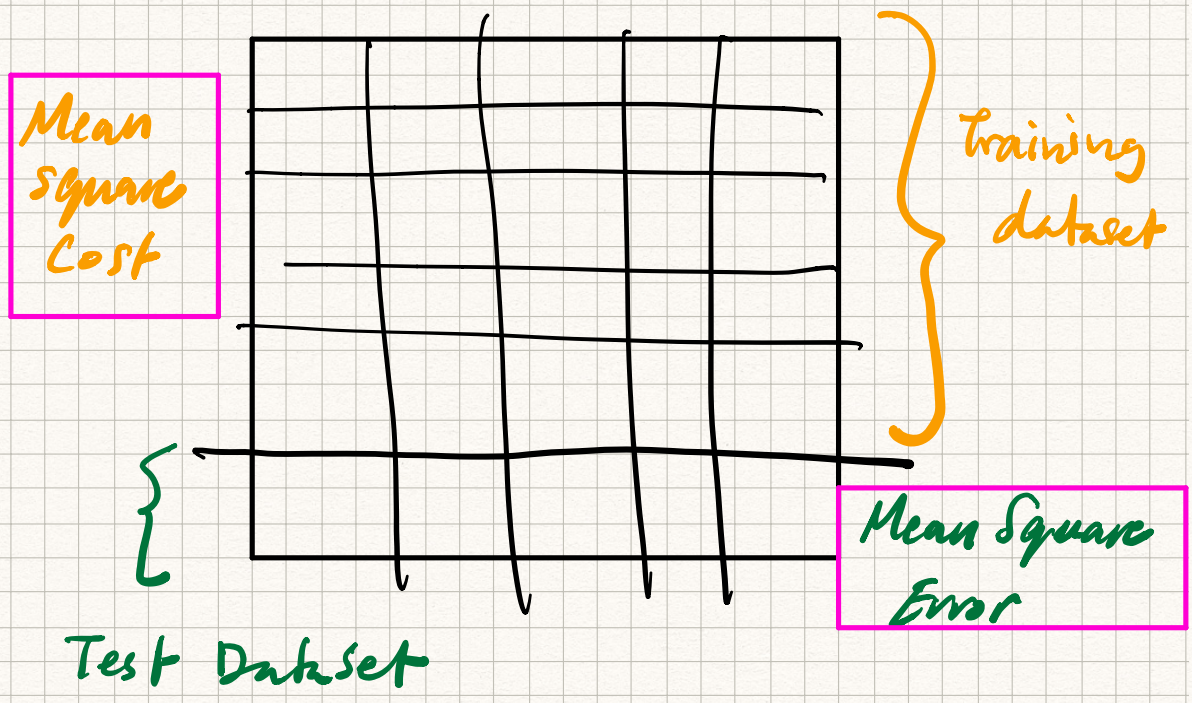
x_1	x_2	Prediction	Actual	(Prediction - Actual)^2
				23.42
				5.8

This is called the "test" dataset

$$MSE = \frac{\left[\frac{(23.42 + 5.8)}{2}\right]}{\#\text{ of rows}}$$

Identical to the Cost used in gradient descent. But serves an entirely different purpose.

Prediction Performance



- Use the training dataset to calculate the optimal values of w_0, w_1 and w_2.
- Use the test dataset to measure the prediction performance of the model.

Some Formulas

(Optional)

SS = Sum of Squares

$\bar{y}$ = average value of actual outputs

$\hat{y}$ = predicted value of output

$y^{(i)}$ = actual output value of the i^{th} row of the dataset.

$\hat{y}^{(i)}$ = predicted output value of the i^{th} row of the dataset.

$|a-b|$ = Absolute value of $a-b$.

$$SS_{total} = \underset{\text{all rows}}{\text{Sum over}} \left[\left(y^{(i)} - \bar{y} \right)^2 \right]$$

$$SS_{regression} = \underset{\text{all rows}}{\text{Sum over}} \left[\left(\hat{y}^{(i)} - \bar{y} \right)^2 \right]$$

$$SS_{residual} = \underset{\text{all rows}}{\text{Sum over}} \left[\left(y^{(i)} - \hat{y}^{(i)} \right)^2 \right]$$

$$R^2 = 1 - \frac{SS_{residual}}{SS_{total}}$$

MSE = Mean Square Error

$$= \frac{SS_{residual}}{\#\text{ of rows in dataset}}$$

RMSE = Root Mean Square Error

$$= \sqrt{MSE}$$

MAE = Mean Absolute Error

$$\frac{\text{Sum over all rows}\left[\left|\hat{y}^{(i)} - y^{(i)}\right|\right]}{\#\text{ of rows in the dataset}}$$

Absolute value