

Guided Tour of Machine Learning in Finance

Linear Regression

Igor Halperin

NYU Tandon School of Engineering, 2017

Linear Regression as a ML task

Task: predict a scalar value $y \in \mathbb{R}$ from a vector of predictors (“**features**”) $\mathbf{X} = (X_0, X_1, \dots, X_{D-1}) \in \mathbb{R}^D$ ($D \geq 1$ is the dimension of the feature space, or the number of predictors).

Linear Regression as a ML task

Task: predict a scalar value $y \in \mathbb{R}$ from a vector of predictors (“**features**”) $\mathbf{X} = (X_0, X_1, \dots, X_{D-1}) \in \mathbb{R}^D$ ($D \geq 1$ is the dimension of the feature space, or the number of predictors).

Given: a dataset $(\mathbf{X}, y)_{data} = [(\mathbf{X}_{train}, y_{train}), (\mathbf{X}_{test}, y_{test})] \sim P_{data}$

Linear Regression as a ML task

Task: predict a scalar value $y \in \mathbb{R}$ from a vector of predictors (“**features**”) $\mathbf{X} = (X_0, X_1, \dots, X_{D-1}) \in \mathbb{R}^D$ ($D \geq 1$ is the dimension of the feature space, or the number of predictors).

Given: a dataset $(\mathbf{X}, y)_{data} = [(\mathbf{X}_{train}, y_{train}), (\mathbf{X}_{test}, y_{test})] \sim P_{data}$

Architecture: Linear

$$\hat{\mathbf{y}} = \mathbf{X} \mathbf{W}$$

N predicted
values

\mathbf{X}
is a N x D
design matrix

$\mathbf{W} \in \mathbb{R}^D$
is a vector
of parameters

Linear Regression as a ML task

Task: predict a scalar value $y \in \mathbb{R}$ from a vector of predictors (“**features**”) $\mathbf{X} = (X_0, X_1, \dots, X_{D-1}) \in \mathbb{R}^D$ ($D \geq 1$ is the dimension of the feature space, or the number of predictors).

Given: a dataset $(\mathbf{X}, y)_{data} = [(\mathbf{X}_{train}, y_{train}), (\mathbf{X}_{test}, y_{test})] \sim p_{data}$

Architecture: Linear

$$\hat{\mathbf{y}} = \mathbf{X} \mathbf{W}$$



Example: \mathbf{y} is a vector of N daily returns of AMZN, and \mathbf{X} is a $N \times D$ design matrix made of $D - 1$ daily market index returns (S&P 500, NASDAQ, VIX, etc.)

Learning in Linear Regression

Performance measure P : mean square error (**MSE**) on the **test set**:

$$\text{MSE}_{test} = \frac{1}{N_{test}} \sum_{n=1}^{N_{test}} \left(\hat{y}_i^{test} - y_i^{test} \right)^2$$

Learning in Linear Regression

Performance measure P : mean square error (**MSE**) on the **test set**:

$$\text{MSE}_{test} = \frac{1}{N_{test}} \sum_{n=1}^{N_{test}} (\hat{y}_i^{test} - y_i^{test})^2 = \frac{1}{N_{test}} \left\| \hat{\mathbf{Y}}^{test} - \mathbf{Y}^{test} \right\|_2^2$$

Learning in Linear Regression

Performance measure P : mean square error (**MSE**) on the **test set**:

$$\text{MSE}_{test} = \frac{1}{N_{test}} \sum_{n=1}^{N_{test}} (\hat{y}_i^{test} - y_i^{test})^2 = \frac{1}{N_{test}} \left\| \hat{\mathbf{Y}}^{test} - \mathbf{Y}^{test} \right\|_2^2 = \frac{1}{N_{test}} \left\| \mathbf{X}^{test} \mathbf{W} - \mathbf{Y}^{test} \right\|_2^2$$

Learning in Linear Regression

Performance measure P : mean square error (**MSE**) on the **test set**:

$$\text{MSE}_{test} = \frac{1}{N_{test}} \sum_{n=1}^{N_{test}} (\hat{y}_i^{test} - y_i^{test})^2 = \frac{1}{N_{test}} \left\| \hat{\mathbf{Y}}^{test} - \mathbf{Y}^{test} \right\|_2^2 = \frac{1}{N_{test}} \left\| \mathbf{X}^{test} \mathbf{W} - \mathbf{Y}^{test} \right\|_2^2$$

How to optimize parameters $\mathbf{W} \in \mathbb{R}^D$?

Learning in Linear Regression

Performance measure P : mean square error (**MSE**) on the **test set**:

$$\text{MSE}_{test} = \frac{1}{N_{test}} \sum_{n=1}^{N_{test}} (\hat{y}_i^{test} - y_i^{test})^2 = \frac{1}{N_{test}} \left\| \hat{\mathbf{Y}}^{test} - \mathbf{Y}^{test} \right\|_2^2 = \frac{1}{N_{test}} \left\| \mathbf{X}^{test} \mathbf{W} - \mathbf{Y}^{test} \right\|_2^2$$

How to optimize parameters $\mathbf{W} \in \mathbb{R}^D$?

Minimize MSE on the **training set**!

Why? Because both MSE_{train} and MSE_{test} estimate the same generalization (expected) error $\mathbb{E} \left[(\hat{\mathbf{y}} - \mathbf{y})^2 \right]$ from the empirical distribution $\sim p_{data}$.

Note: In ML, we often minimize one function, while actually caring about minimization of another function. This makes ML different from optimization that typically focuses only on one function.

Learning in Linear Regression

Performance measure P : mean square error (**MSE**) on the **test set**:

$$\text{MSE}_{test} = \frac{1}{N_{test}} \sum_{n=1}^{N_{test}} (\hat{y}_i^{test} - y_i^{test})^2 = \frac{1}{N_{test}} \left\| \hat{\mathbf{Y}}^{test} - \mathbf{Y}^{test} \right\|_2^2 = \frac{1}{N_{test}} \left\| \mathbf{X}^{test} \mathbf{W} - \mathbf{Y}^{test} \right\|_2^2$$

How to optimize parameters $\mathbf{W} \in \mathbb{R}^D$? Minimize MSE on the **training set**! Why? Because both MSE_{test} and MSE_{train} estimate the generalization (expected) error.

For optimal weights \mathbf{W} , the gradient of MSE_{train} should be 0:

$$\nabla_{\mathbf{W}} \text{MSE}_{train} = \nabla_{\mathbf{W}} \frac{1}{N_{train}} \left\| \hat{\mathbf{Y}}^{train} - \mathbf{Y}^{train} \right\|_2^2 = \frac{1}{N_{train}} \nabla_{\mathbf{W}} \left\| \mathbf{X}^{train} \mathbf{W} - \mathbf{Y}^{train} \right\|_2^2 = 0$$

Learning in Linear Regression

Performance measure P : mean square error (**MSE**) on the **test set**:

$$\text{MSE}_{test} = \frac{1}{N_{test}} \sum_{n=1}^{N_{test}} (\hat{y}_i^{test} - y_i^{test})^2 = \frac{1}{N_{test}} \left\| \hat{\mathbf{Y}}^{test} - \mathbf{Y}^{test} \right\|_2^2 = \frac{1}{N_{test}} \left\| \mathbf{X}^{test} \mathbf{W} - \mathbf{Y}^{test} \right\|_2^2$$

How to optimize parameters $\mathbf{W} \in \mathbb{R}^D$? Minimize MSE on the **training set**! Why? Because both MSE_{test} and MSE_{train} estimate the generalization (expected) error.

For optimal weights \mathbf{W} , the gradient of MSE_{train} should be 0:

$$\nabla_{\mathbf{W}} \text{MSE}_{train} = \nabla_{\mathbf{W}} \frac{1}{N_{train}} \left\| \hat{\mathbf{Y}}^{train} - \mathbf{Y}^{train} \right\|_2^2 = \frac{1}{N_{train}} \nabla_{\mathbf{W}} \left\| \mathbf{X}^{train} \mathbf{W} - \mathbf{Y}^{train} \right\|_2^2 = 0$$

$$\Rightarrow \nabla_{\mathbf{W}} \left(\mathbf{X}^{train} \mathbf{W} - \mathbf{y}^{train} \right)^T \left(\mathbf{X}^{train} \mathbf{W} - \mathbf{y}^{train} \right) = 0$$

Learning in Linear Regression

Performance measure P : mean square error (**MSE**) on the **test set**:

$$\text{MSE}_{test} = \frac{1}{N_{test}} \sum_{n=1}^{N_{test}} (\hat{y}_i^{test} - y_i^{test})^2 = \frac{1}{N_{test}} \left\| \hat{\mathbf{Y}}^{test} - \mathbf{Y}^{test} \right\|_2^2 = \frac{1}{N_{test}} \left\| \mathbf{X}^{test} \mathbf{W} - \mathbf{Y}^{test} \right\|_2^2$$

How to optimize parameters $\mathbf{W} \in \mathbb{R}^D$? Minimize MSE on the **training set**! Why? Because both MSE_{test} and MSE_{train} estimate the generalization (expected) error.

For optimal weights \mathbf{W} , the gradient of MSE_{train} should be 0:

$$\nabla_{\mathbf{W}} \text{MSE}_{train} = \nabla_{\mathbf{W}} \frac{1}{N_{train}} \left\| \hat{\mathbf{Y}}^{train} - \mathbf{Y}^{train} \right\|_2^2 = \frac{1}{N_{train}} \nabla_{\mathbf{W}} \left\| \mathbf{X}^{train} \mathbf{W} - \mathbf{Y}^{train} \right\|_2^2 = 0$$

$$\Rightarrow \nabla_{\mathbf{W}} \left(\mathbf{X}^{train} \mathbf{W} - \mathbf{y}^{train} \right)^T \left(\mathbf{X}^{train} \mathbf{W} - \mathbf{y}^{train} \right) = 0$$

Simplify notation:
 $\mathbf{X}^{train} \rightarrow \mathbf{X}, \mathbf{y}^{train} \rightarrow \mathbf{y}$

Learning in Linear Regression

Performance measure P : mean square error (**MSE**) on the **test set**:

$$\text{MSE}_{test} = \frac{1}{N_{test}} \sum_{n=1}^{N_{test}} (\hat{y}_i^{test} - y_i^{test})^2 = \frac{1}{N_{test}} \left\| \hat{\mathbf{Y}}^{test} - \mathbf{Y}^{test} \right\|_2^2 = \frac{1}{N_{test}} \left\| \mathbf{X}^{test} \mathbf{W} - \mathbf{Y}^{test} \right\|_2^2$$

How to optimize parameters $\mathbf{W} \in \mathbb{R}^D$? Minimize MSE on the **training set**! Why? Because both MSE_{test} and MSE_{train} estimate the generalization (expected) error.

For optimal weights \mathbf{W} , the gradient of MSE_{train} should be 0:

$$\nabla_{\mathbf{W}} \text{MSE}_{train} = \nabla_{\mathbf{W}} \frac{1}{N_{train}} \left\| \hat{\mathbf{Y}}^{train} - \mathbf{Y}^{train} \right\|_2^2 = \frac{1}{N_{train}} \nabla_{\mathbf{W}} \left\| \mathbf{X}^{train} \mathbf{W} - \mathbf{Y}^{train} \right\|_2^2 = 0$$

$$\Rightarrow \nabla_{\mathbf{W}} \left(\mathbf{X}^{train} \mathbf{W} - \mathbf{y}^{train} \right)^T \left(\mathbf{X}^{train} \mathbf{W} - \mathbf{y}^{train} \right) = 0$$

$$\Rightarrow \nabla_{\mathbf{W}} \left(\mathbf{W}^T \mathbf{X}^T \mathbf{X} \mathbf{W} - 2 \mathbf{W}^T \mathbf{X}^T \mathbf{y} + \mathbf{y}^T \mathbf{y} \right)$$

Simplify notation:
 $\mathbf{X}^{train} \rightarrow \mathbf{X}, \mathbf{y}^{train} \rightarrow \mathbf{y}$

Learning in Linear Regression

Performance measure P : mean square error (**MSE**) on the **test set**:

$$\text{MSE}_{test} = \frac{1}{N_{test}} \sum_{n=1}^{N_{test}} (\hat{y}_i^{test} - y_i^{test})^2 = \frac{1}{N_{test}} \left\| \hat{\mathbf{Y}}^{test} - \mathbf{Y}^{test} \right\|_2^2 = \frac{1}{N_{test}} \left\| \mathbf{X}^{test} \mathbf{W} - \mathbf{Y}^{test} \right\|_2^2$$

How to optimize parameters $\mathbf{W} \in \mathbb{R}^D$? Minimize MSE on the **training set**! Why? Because both MSE_{test} and MSE_{train} estimate the generalization (expected) error.

For optimal weights \mathbf{W} , the gradient of MSE_{train} should be 0:

$$\nabla_{\mathbf{W}} \text{MSE}_{train} = \nabla_{\mathbf{W}} \frac{1}{N_{train}} \left\| \hat{\mathbf{Y}}^{train} - \mathbf{Y}^{train} \right\|_2^2 = \frac{1}{N_{train}} \nabla_{\mathbf{W}} \left\| \mathbf{X}^{train} \mathbf{W} - \mathbf{Y}^{train} \right\|_2^2 = 0$$

$$\Rightarrow \nabla_{\mathbf{W}} \left(\mathbf{X}^{train} \mathbf{W} - \mathbf{y}^{train} \right)^T \left(\mathbf{X}^{train} \mathbf{W} - \mathbf{y}^{train} \right) = 0$$

Simplify notation:
 $\mathbf{X}^{train} \rightarrow \mathbf{X}, \mathbf{y}^{train} \rightarrow \mathbf{y}$

$$\Rightarrow \nabla_{\mathbf{W}} \left(\mathbf{W}^T \mathbf{X}^T \mathbf{X} \mathbf{W} - 2 \mathbf{W}^T \mathbf{X}^T \mathbf{y} + \mathbf{y}^T \mathbf{y} \right) = 2 \left(\mathbf{X}^T \mathbf{X} \right) \mathbf{W} - 2 \left(\mathbf{X}^T \mathbf{y} \right) = 0$$

Learning in Linear Regression

Performance measure P : mean square error (**MSE**) on the **test set**:

$$\text{MSE}_{\text{test}} = \frac{1}{N_{\text{test}}} \sum_{n=1}^{N_{\text{test}}} (\hat{y}_i^{\text{test}} - y_i^{\text{test}})^2 = \frac{1}{N_{\text{test}}} \left\| \hat{\mathbf{Y}}^{\text{test}} - \mathbf{Y}^{\text{test}} \right\|_2^2 = \frac{1}{N_{\text{test}}} \left\| \mathbf{X}^{\text{test}} \mathbf{W} - \mathbf{Y}^{\text{test}} \right\|_2^2$$

How to optimize parameters $\mathbf{W} \in \mathbb{R}^D$? Minimize MSE on the **training set**! Why? Because both MSE_{test} and $\text{MSE}_{\text{train}}$ estimate the generalization (expected) error.

For optimal weights \mathbf{W} , the gradient of $\text{MSE}_{\text{train}}$ should be 0:

$$\nabla_{\mathbf{W}} \text{MSE}_{\text{train}} = \nabla_{\mathbf{W}} \frac{1}{N_{\text{train}}} \left\| \hat{\mathbf{Y}}^{\text{train}} - \mathbf{Y}^{\text{train}} \right\|_2^2 = \frac{1}{N_{\text{train}}} \nabla_{\mathbf{W}} \left\| \mathbf{X}^{\text{train}} \mathbf{W} - \mathbf{Y}^{\text{train}} \right\|_2^2 = 0$$

$$\Rightarrow \nabla_{\mathbf{W}} \left(\mathbf{X}^{\text{train}} \mathbf{W} - \mathbf{y}^{\text{train}} \right)^T \left(\mathbf{X}^{\text{train}} \mathbf{W} - \mathbf{y}^{\text{train}} \right) = 0$$

Simplify notation:
 $\mathbf{X}^{\text{train}} \rightarrow \mathbf{X}, \mathbf{y}^{\text{train}} \rightarrow \mathbf{y}$

$$\Rightarrow \nabla_{\mathbf{W}} \left(\mathbf{W}^T \mathbf{X}^T \mathbf{X} \mathbf{W} - 2 \mathbf{W}^T \mathbf{X}^T \mathbf{y} + \mathbf{y}^T \mathbf{y} \right) = 2 \left(\mathbf{X}^T \mathbf{X} \right) \mathbf{W} - 2 \left(\mathbf{X}^T \mathbf{y} \right) = 0$$

$$\Rightarrow \boxed{\mathbf{W} = \left(\mathbf{X}^T \mathbf{X} \right)^{-1} \mathbf{X}^T \mathbf{y}}$$

Normal equation

Learning in Linear Regression

Performance measure P : mean square error (**MSE**) on the **test set**:

$$\text{MSE}_{\text{test}} = \frac{1}{N_{\text{test}}} \sum_{n=1}^{N_{\text{test}}} (\hat{y}_i^{\text{test}} - y_i^{\text{test}})^2 = \frac{1}{N_{\text{test}}} \left\| \hat{\mathbf{Y}}^{\text{test}} - \mathbf{Y}^{\text{test}} \right\|_2^2 = \frac{1}{N_{\text{test}}} \left\| \mathbf{X}^{\text{test}} \mathbf{W} - \mathbf{Y}^{\text{test}} \right\|_2^2$$

How to optimize parameters $\mathbf{W} \in \mathbb{R}^D$? Minimize MSE on the **training set**! Why? Because both MSE_{test} and $\text{MSE}_{\text{train}}$ estimate the generalization (expected) error.

For optimal weights \mathbf{W} , the gradient of $\text{MSE}_{\text{train}}$ should be 0:

$$\nabla_{\mathbf{W}} \text{MSE}_{\text{train}} = \nabla_{\mathbf{W}} \frac{1}{N_{\text{train}}} \left\| \hat{\mathbf{Y}}^{\text{train}} - \mathbf{Y}^{\text{train}} \right\|_2^2 = \frac{1}{N_{\text{train}}} \nabla_{\mathbf{W}} \left\| \mathbf{X}^{\text{train}} \mathbf{W} - \mathbf{Y}^{\text{train}} \right\|_2^2 = 0$$

$$\Rightarrow \nabla_{\mathbf{W}} \left(\mathbf{X}^{\text{train}} \mathbf{W} - \mathbf{y}^{\text{train}} \right)^T \left(\mathbf{X}^{\text{train}} \mathbf{W} - \mathbf{y}^{\text{train}} \right) = 0$$

Simplify notation:
 $\mathbf{X}^{\text{train}} \rightarrow \mathbf{X}, \mathbf{y}^{\text{train}} \rightarrow \mathbf{y}$

$$\Rightarrow \nabla_{\mathbf{W}} \left(\mathbf{W}^T \mathbf{X}^T \mathbf{X} \mathbf{W} - 2 \mathbf{W}^T \mathbf{X}^T \mathbf{y} + \mathbf{y}^T \mathbf{y} \right) = 2 \left(\mathbf{X}^T \mathbf{X} \right) \mathbf{W} - 2 \left(\mathbf{X}^T \mathbf{y} \right) = 0$$

Here
 $\mathbf{X} = \mathbf{X}^{\text{train}}, \mathbf{y} = \mathbf{y}^{\text{train}}$

\Rightarrow

$$\mathbf{W} = \left(\mathbf{X}^T \mathbf{X} \right)^{-1} \mathbf{X}^T \mathbf{y}$$

Normal
equation

Learning in Linear Regression

Visualize the result for optimal weights \mathbf{W} in the vector form

$$\mathbf{W} = \left(\mathbf{X}^T \mathbf{X} \right)^{-1} \mathbf{X}^T \mathbf{y}$$

Diagram illustrating the dimensions of the matrices in the vector form equation:

\mathbf{W} (Matrix $D \times 1$) = $\left[\begin{array}{c} \mathbf{X}^T \quad \mathbf{X} \end{array} \right]^{-1} \begin{array}{c} \mathbf{X}^T \quad \mathbf{y} \end{array}$

The dimensions are indicated by boxes below the matrices:

- \mathbf{X}^T and \mathbf{X} are grouped as Matrix $D \times D$.
- \mathbf{X}^T and \mathbf{y} are grouped as Matrix $D \times N$.
- \mathbf{y} is also labeled as Matrix $N \times 1$.

Prediction
in-sample:

$$\hat{\mathbf{y}} = \mathbf{XW} = \mathbf{X} \left(\mathbf{X}^T \mathbf{X} \right)^{-1} \mathbf{X}^T \mathbf{y} \equiv \mathbf{H} \mathbf{y}$$

$$= \mathbf{H}$$

The “hat” (projection) matrix

Learning in Linear Regression

Visualize the result for optimal weights \mathbf{W} in the vector form

$$\mathbf{W} = \left(\mathbf{X}^T \mathbf{X} \right)^{-1} \mathbf{X}^T \mathbf{y}$$

Matrix $D \times 1$

$$\mathbf{W} = \left[\begin{array}{c|c} \mathbf{X}^T & \mathbf{X} \end{array} \right]^{-1} \begin{array}{c} \mathbf{X}^T \\ \mathbf{y} \end{array}$$

Matrix $N \times 1$

Matrix $D \times D$

Matrix $D \times N$

Symmetric and
idempotent:

$$\mathbf{H}^2 = \mathbf{H}$$

Prediction
in-sample:

$$\hat{\mathbf{y}} = \mathbf{X}\mathbf{W} = \mathbf{X} \left(\mathbf{X}^T \mathbf{X} \right)^{-1} \mathbf{X}^T \mathbf{y} \equiv \mathbf{H} \mathbf{y}$$

$$= \mathbf{H}$$

The “hat” (projection) matrix

Learning in Linear Regression

Visualize the result for optimal weights \mathbf{W} in the vector form

$$\mathbf{W} = \left(\mathbf{X}^T \mathbf{X} \right)^{-1} \mathbf{X}^T \mathbf{y}$$

The diagram illustrates the vector form of the linear regression equation. At the top, the equation $\mathbf{W} = (\mathbf{X}^T \mathbf{X})^{-1} \mathbf{X}^T \mathbf{y}$ is shown in a box. Below it, the same equation is visualized with colored blocks representing matrices and vectors. The weight vector \mathbf{W} is shown as a blue vertical bar on the left, with a label 'Matrix D x 1' to its left. The matrix $(\mathbf{X}^T \mathbf{X})^{-1}$ is shown as a blue block in the middle, with a label 'Matrix D x D' below it. The matrix \mathbf{X}^T is shown as a blue block to the right of the inverse matrix, with a label 'Matrix D x N' below it. The vector \mathbf{y} is shown as a blue vertical bar on the far right, with a label 'Matrix N x 1' to its right. The entire expression is enclosed in large square brackets with a superscript -1.

Prediction out-of-sample: $\hat{\mathbf{y}} = \mathbf{X}^{test} \mathbf{W}$

How to overfit with Linear Regression: add more features to the design matrix without a proper control!