MACHINE LEARNING

Electrical Summer Workshops (ESW) 2022

Electrical Engineering Department - Sharif University of Technology

Instructors: *Alireza Gargoori Motlagh – Amir Mirrashid – Ali Nourian*

LINEAR REGRESSION

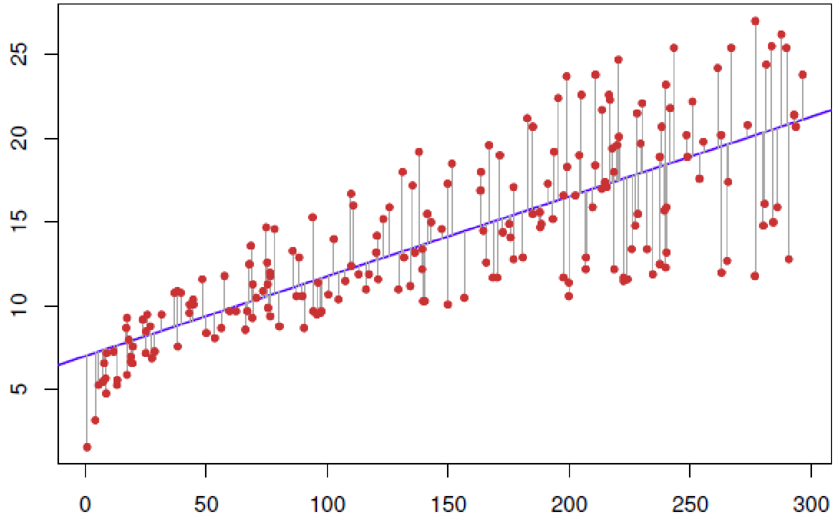
Linear Regression

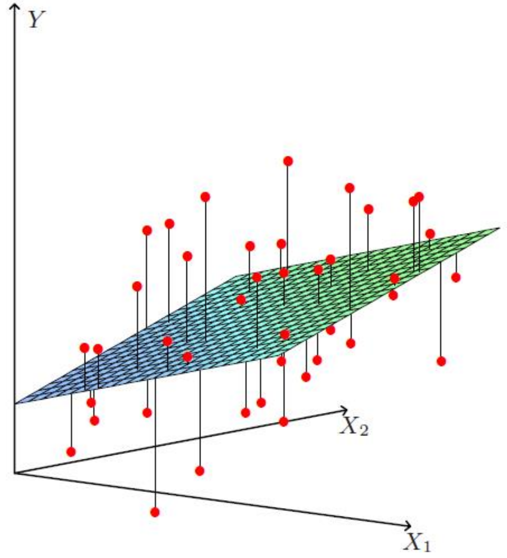
Relationship between the input variables and the output is modeled through a linear function

$$f(X) = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \cdots + \beta_p X_p = \beta_0 + \sum_{j=1}^{p} \beta_j X_j$$

The linear model either assumes that the regression function $E(Y|X)$ is linear, or that the linear model is a reasonable approximation.

$$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \cdots + \beta_p X_p + \epsilon = \beta_0 + \sum_{j=1}^{p} \beta_j X_j + \epsilon$$





Coefficients and Features

Here the β_j's are unknown parameters or coefficients, and the variables X_j can come from different sources:

- quantitative inputs;
- transformations of quantitative inputs, such as log, square-root or square;
- basis expansions, such as $X_2 = X_1^2,\ \ X_3 = X_1^3$, leading to a polynomial representation;
- numeric or “dummy” coding of the levels of qualitative inputs. For example, if G is a five-level factor input, we might create X_j , j = 1, . . . , 5, such that $X_j = I(G = j)$. Together this group of X_j represents the effect of G by a set of level-dependent constants, since in $\sum_{j=1}^{5} \beta_j X_j$,one of the X_j's is one, and the others are zero;
- interactions between variables, for example, $X_3 = X_1 . X_2$;

Estimating the Parameters

Estimation of the output(target value) for training samples would be:

$$\hat{y}^{(i)} = \hat{\beta}_0 + \hat{\beta}_1 x_1^{(i)} + \hat{\beta}_2 x_2^{(i)} + \cdots + \hat{\beta}_p x_p^{(i)}$$

We want to find the optimal values of $\hat{\beta}_j$ in order to minimize the cost function defined as:
(Residual Sum of squares)

$$RSS = \sum_{i=1}^{n} (y^{(i)} - \hat{y}^{(i)})^2$$

Or equivalently:
(Mean Squared Error)

$$MSE = \frac{1}{n} \sum_{i=1}^{n} (y^{(i)} - \hat{y}^{(i)})^2$$

- Convex function of the regression parameters.
- Unbiased estimation of the population regression line. (if some requirements are met)

Cost Function in Matrix Form

$$\boldsymbol{y} = \begin{bmatrix} y^{(1)} \\ \vdots \\ y^{(n)} \end{bmatrix} \qquad \widehat{\boldsymbol{y}} = \begin{bmatrix} \hat{y}^{(1)} \\ \vdots \\ \hat{y}^{(n)} \end{bmatrix}$$

$$RSS = \sum_{i=1}^{n} (y^{(i)} - \hat{y}^{(i)})^2 = (\boldsymbol{y} - \widehat{\boldsymbol{y}})^T (\boldsymbol{y} - \widehat{\boldsymbol{y}}) = \left|\left|\boldsymbol{y} - \widehat{\boldsymbol{y}}\right|\right|_{L2}^2$$

$$\widehat{\boldsymbol{y}} = \begin{bmatrix} \hat{y}^{(1)} \\ \vdots \\ \hat{y}^{(n)} \end{bmatrix} = \begin{bmatrix} \hat{\beta}_0 + \hat{\beta}_1 x_1^{(1)} + \hat{\beta}_2 x_2^{(1)} + \cdots + \hat{\beta}_p x_p^{(1)} \\ \vdots \\ \hat{\beta}_0 + \hat{\beta}_1 x_1^{(n)} + \hat{\beta}_2 x_2^{(n)} + \cdots + \hat{\beta}_p x_p^{(n)} \end{bmatrix} =$$

$$\begin{bmatrix} 1 & x_1^{(1)} & \dots\, x_p^{(1)} \\ \vdots & \vdots & \dots\quad \vdots \\ 1 & x_1^{(n)} & \dots\, x_p^{(n)} \end{bmatrix} \begin{bmatrix} \hat{\beta}_0 \\ \hat{\beta}_1 \\ \vdots \\ \hat{\beta}_p \end{bmatrix} = \boldsymbol{X}\widehat{\boldsymbol{\beta}}$$

Least-Squares Problem

$$RSS = (\boldsymbol{y} - \widehat{\boldsymbol{y}})^T(\boldsymbol{y} - \widehat{\boldsymbol{y}}) = \left|\left|\boldsymbol{y} - \widehat{\boldsymbol{y}}\right|\right|_{L2}^2, \qquad \widehat{\boldsymbol{y}} = \boldsymbol{X}\widehat{\boldsymbol{\beta}}$$

So, the cost would be:

$$RSS = (\boldsymbol{y} - \boldsymbol{X\beta})^T(\boldsymbol{y} - \boldsymbol{X\beta})$$

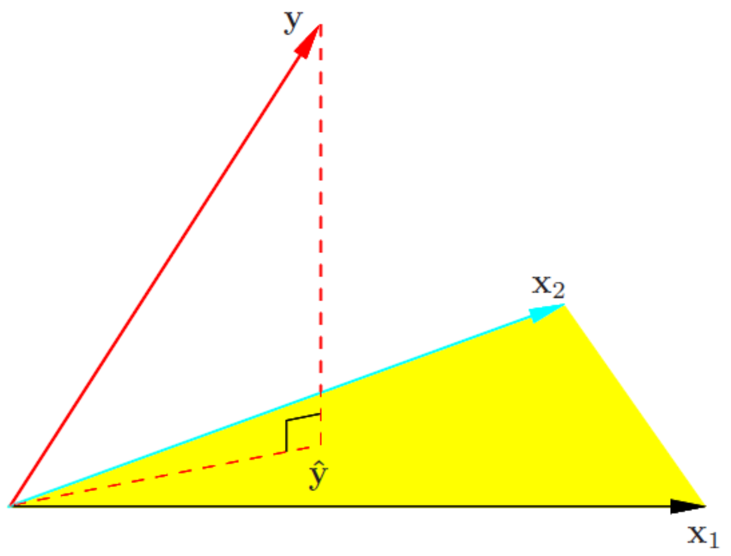
Since this is a convex function in regression parameters, we can set its derivative to zero to find the optimal values of $\widehat{\beta_j}$ for the minimum value of RSS:

$$\frac{\partial\, RSS}{\partial\, \boldsymbol{\beta}} = -2\boldsymbol{X^T}(\boldsymbol{y} - \boldsymbol{X\beta}) = 0 \rightarrow \boldsymbol{X^T X}\,\widehat{\boldsymbol{\beta}} = \boldsymbol{X^T y} \quad (Normal\ eq.)$$

Hence, considering that X is full column-rank (i.e. columns of X, features, are linearly independent), the *unique* solution would be:

$$\widehat{\boldsymbol{\beta}} = \left(\boldsymbol{X^T X}\right)^{\boldsymbol{-1}}\boldsymbol{X^T y}$$

Geometric Interpretation



The prediction of least-squares problem($\hat{\boldsymbol{y}}$), would be the orthogonal projection of outcome vector ($\boldsymbol{y}$) onto the hyperplane spanned by the features of X.

So the residual error, $\boldsymbol{e} = \boldsymbol{y} - \hat{\boldsymbol{y}}$ is always orthogonal to this subspace and it's the part of the target that the model cannot predict.

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

$\boldsymbol{H} = \boldsymbol{X}\left(\boldsymbol{X}^{\boldsymbol{T}}\boldsymbol{X}\right)^{-\mathbf{1}}\boldsymbol{X}^{\boldsymbol{T}}$ is referred as *Hat* matrix or *Projection* matrix.

*Correlated Variables

- What if there is a perfect collinearity between features? (e.g. $X_2 = 4X_1$)
- In this case $\boldsymbol{X}$ is not full column rank and hence $\boldsymbol{X^T X}$ is singular and not invertible.
- The solution still exists, but it is not unique. (The solution are still the projection of y onto the column space of X but there are more than one way to express that projection.)

One simple solution is to drop the redundant (dependent) columns from X so that we can still use the previous results. (Most regression software package detect these redundancies and automatically implement some strategies for removing them.)

- Also, there are cases when 2 or more features are highly correlated.
- P-values of regression coefficients are not significant. (by changing the train set, the coefficients of these features can vary too much.)

There are some methods to detect and solve this problem:

  - Backward Selection
  - Forward Selection
  - Shrinkage Methods

*Unbiasedness of Linear Regression

$$\hat{\beta} = (X^T X)^{-1} X^T y = (X^T X)^{-1} X^T (X\beta + \epsilon) \rightarrow$$
$$\hat{\beta} = \beta + (X^T X)^{-1} X^T \epsilon$$

Assumptions:

- iid samples (indepndent and identically distributed)
- Full column-rank (No perfect collinearity among features)
- $\epsilon|\boldsymbol{X} \sim \mathcal{N}(0, \sigma^2 \boldsymbol{I})$
  - Zero conditional mean ($E(\epsilon|X) = 0$)
  - Constant variance of error term and iid.
  - Normally distributed. (Not really required, but for some other inferences such as distributions of β_j's and their p-values, estimation of variance of noise, etc.)

Under the above assumptions, we can write:

$$E(\widehat{\boldsymbol{\beta}}|\boldsymbol{X}) = E\left(\boldsymbol{\beta} + \left(\boldsymbol{X}^{\boldsymbol{T}}\boldsymbol{X}\right)^{-\mathbf{1}}\boldsymbol{X}^{\boldsymbol{T}}\boldsymbol{\epsilon}\middle|\boldsymbol{X}\right) = E(\boldsymbol{\beta}|\boldsymbol{X}) + E\left(\left(\boldsymbol{X}^{\boldsymbol{T}}\boldsymbol{X}\right)^{-\mathbf{1}}\boldsymbol{X}^{\boldsymbol{T}}\boldsymbol{\epsilon}\middle|\boldsymbol{X}\right) \rightarrow$$
$$E(\widehat{\boldsymbol{\beta}}|\boldsymbol{X}) = \boldsymbol{\beta} + \left(\boldsymbol{X}^{\boldsymbol{T}}\boldsymbol{X}\right)^{-\mathbf{1}}\boldsymbol{X}^{\boldsymbol{T}}\, E(\boldsymbol{\epsilon}|\boldsymbol{X}) = \boldsymbol{\beta}$$

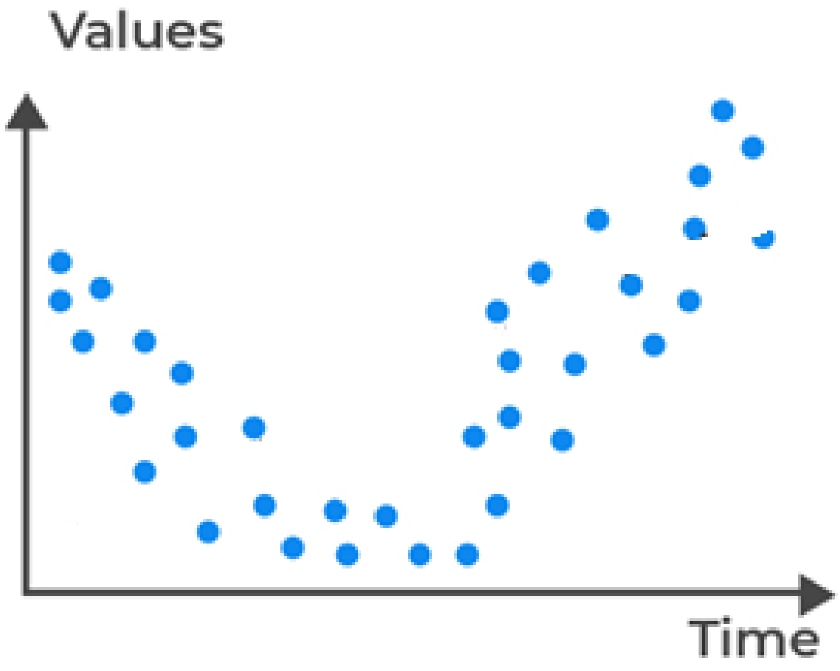
Gauss-Markov theorem: ordinary least squares (OLS) regression produces unbiased estimates that have the smallest variance of all possible linear estimators.

A Case Study: Polynomial Fitting

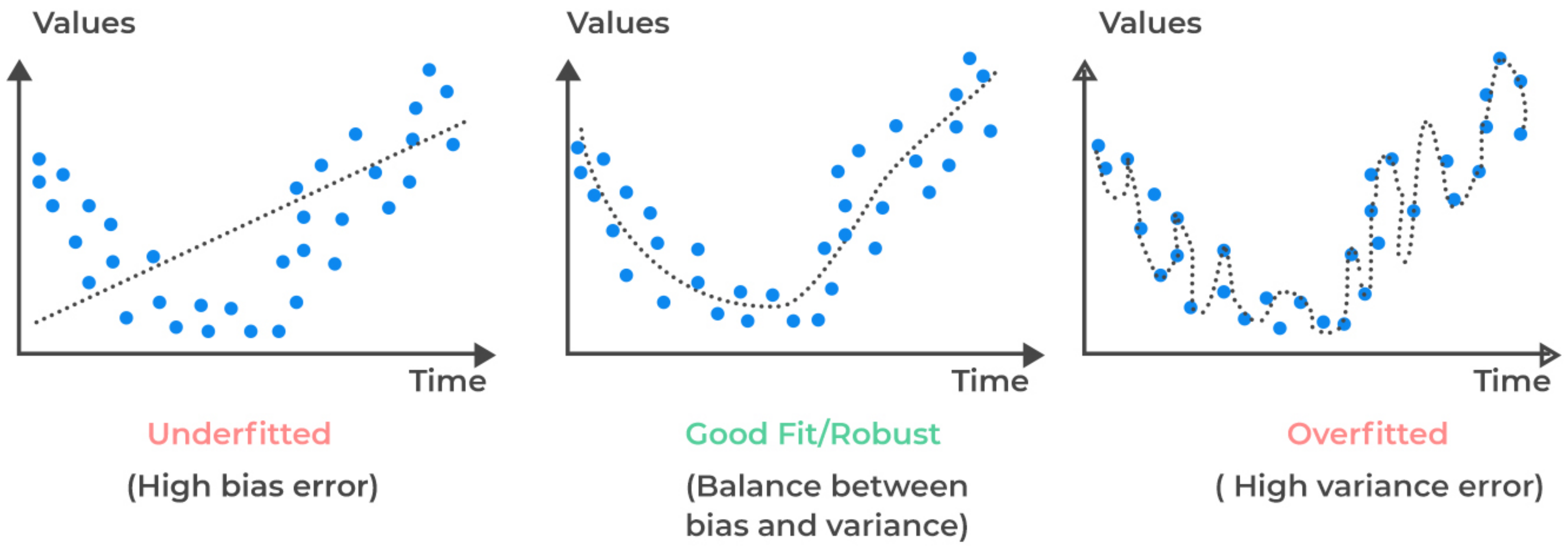
$$Y = \beta_0 + \beta_1 X + \beta_2 X^2 + \beta_3 X^3 + \cdots + \beta_p X^p + \epsilon = \sum_{j=0}^{p} \beta_j X^j + \epsilon$$

$$\boldsymbol{X} = \begin{bmatrix} 1 & x^{(1)} & x^{2\,(1)} & \dots & x^{p\,(1)} \\ 1 & x^{(2)} & x^{2\,(2)} & \dots & x^{p\,(2)} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 1 & x^{(n)} & x^{2\,(n)} & \dots & x^{p\,(n)} \end{bmatrix}, \quad \boldsymbol{y} = \begin{bmatrix} y^{(1)} \\ \vdots \\ y^{(n)} \end{bmatrix}$$

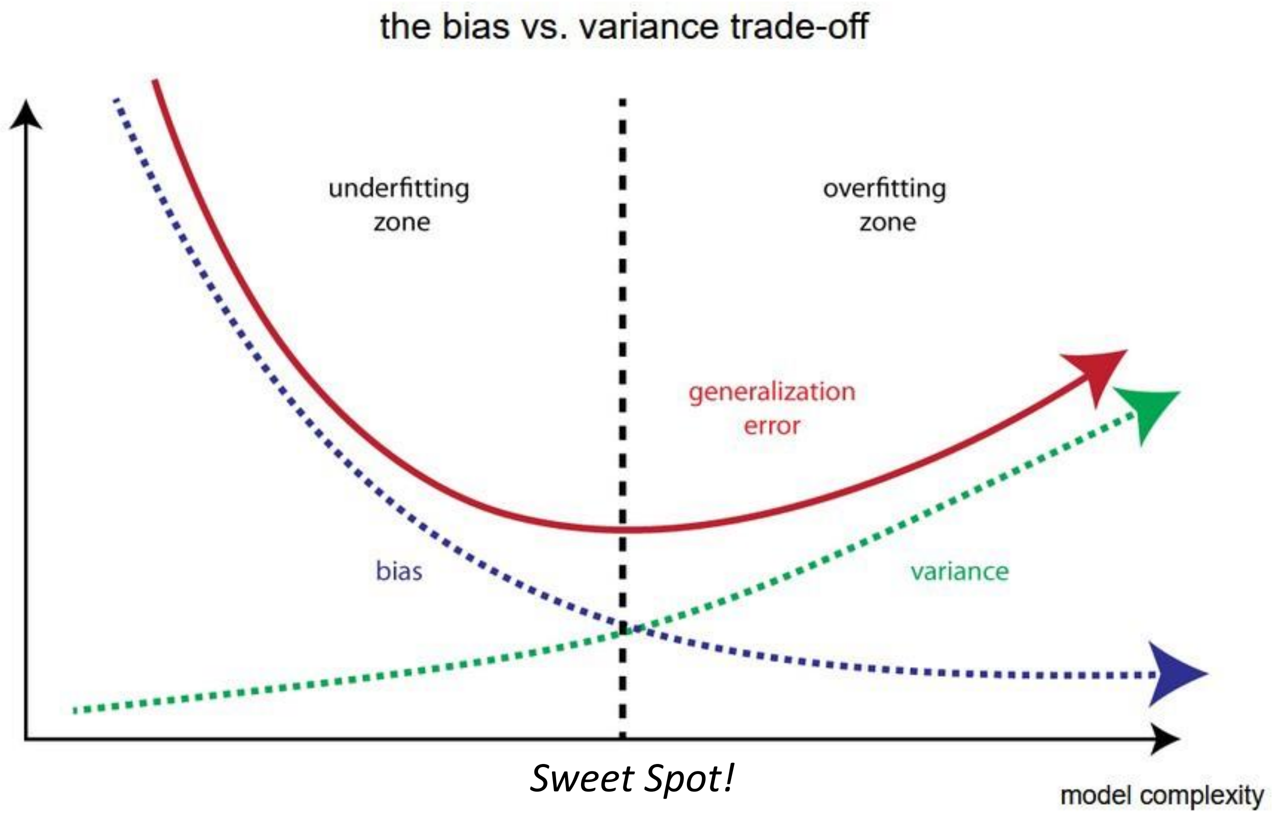
$$\boldsymbol{\beta} = \begin{bmatrix} \beta_0 \\ \beta_1 \\ \beta_2 \\ \vdots \\ \beta_p \end{bmatrix}$$



Overfitting and Underfitting



Bias-Variance Tradeoff



Approach to a ML Problem

1. Collecting your data.
2. Prepare and preprocess the dataset and Splitting the data randomly to a training set, a validation set and a test set.
3. Choose an appropriate algorithm.
4. Training the model on the train set.
5. Evaluating the model on test.
6. Hyperparameter tuning on the validation set.
7. Making predictions.