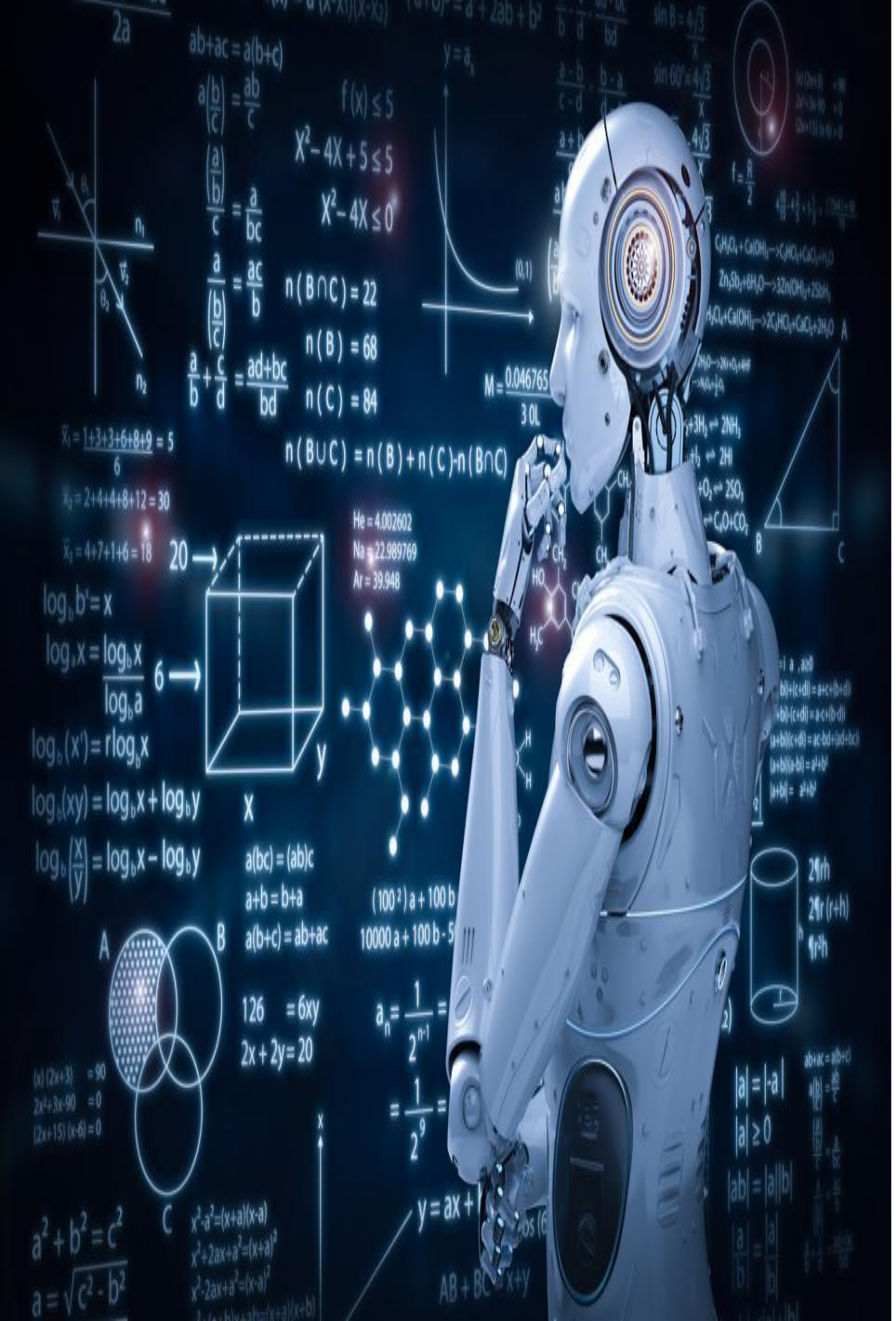


Practical Machine Learning

Day 7: Mar23 DBDA

Kiran Waghmare

Agenda

- Types of Regression
  - Ridge
  - Lasso
  - Elasticnet

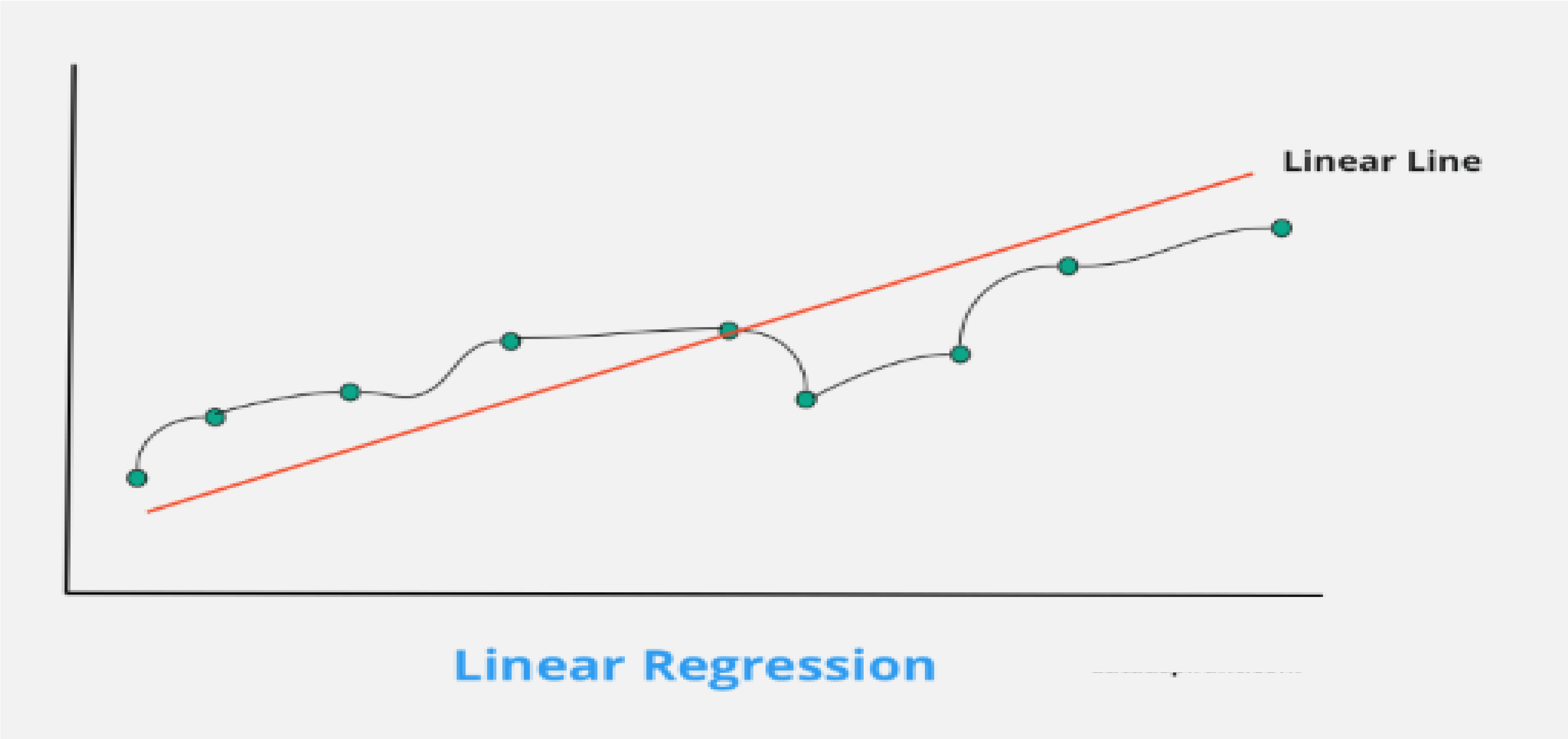
Mean Squared Error (MSE) is one of the regression evaluation metrics. It is calculated as the average squared difference between the predicted values and the real value. The mathematical equation for MSE is as below :

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

- $MeanSquaredError(mse) = \sqrt{(\frac{1}{n})\sum_{i=1}^{n}(y_i - x_i)^2}$
- $MeanAbsoluteError(mae) = (\frac{1}{n})\sum_{i=1}^{n}\left|y_i - x_i\right|$

Ridge Regression

- Ridge regression is the regularized form of linear regression."



Linear Regression

Simple Linear Regression

$$y = b_0 + b_1 x_1$$

Multiple Linear Regression

$$y = b_0 + b_1 x_1 + b_2 x_2 + \dots + b_n x_n$$

Polynomial Linear Regression

$$y = b_0 + b_1 x_1 + b_2 x_1^2 + \dots + b_n x_1^n$$

Brief Review of Linear Regression

The mathematical expression of linear regression is as follows.

$$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_1 + \cdots + \beta_p X_{p-1} + \epsilon$$

Brief Review of Linear Regression

The mathematical expression of linear regression is as follows.

$$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_1 + \cdots + \beta_p X_{p-1} + \epsilon$$

This can then be expressed in matrix version.

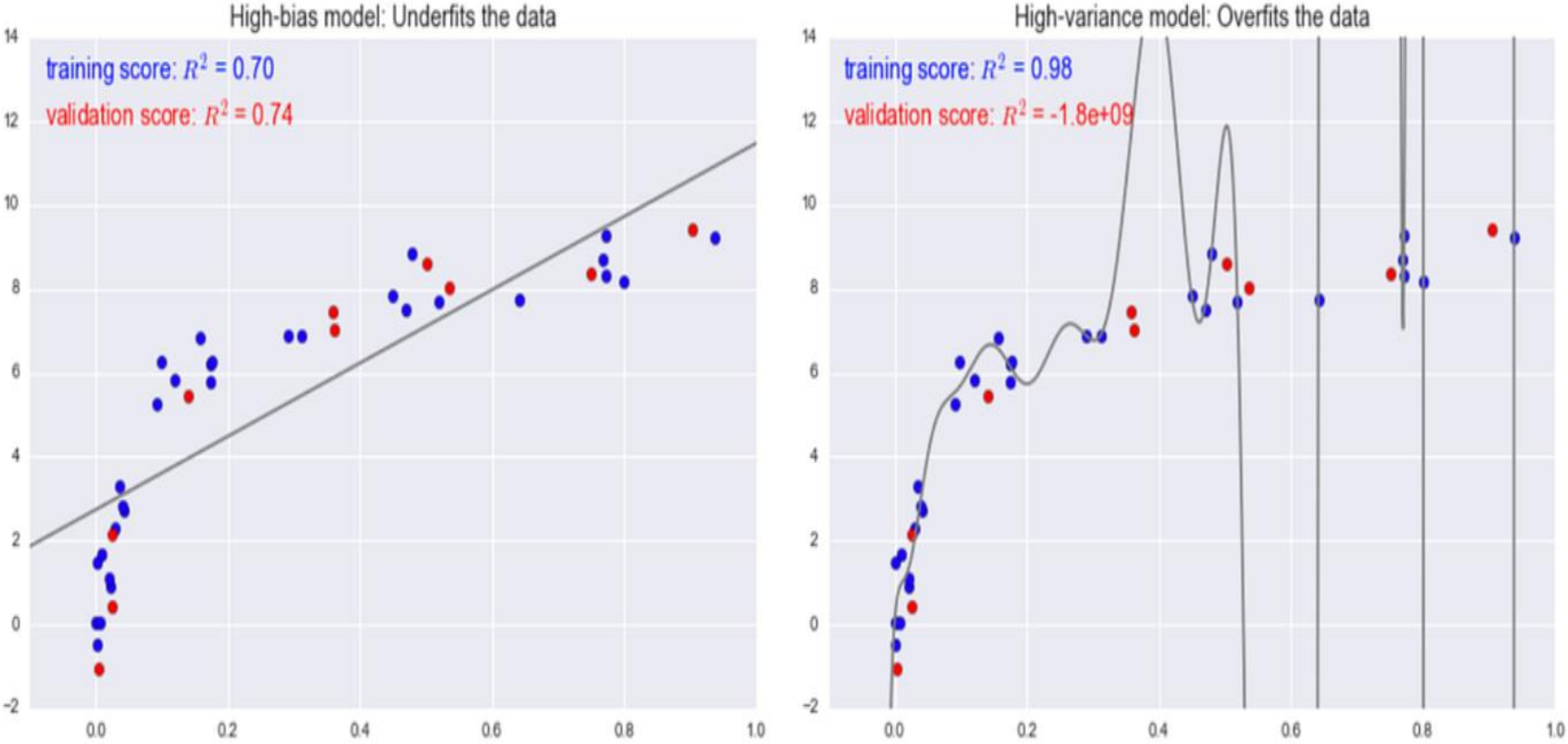
$$Y = XB + e \;\; where\, B\, is\, \beta_0,\ \beta_1,\ \ldots\ \beta_p$$

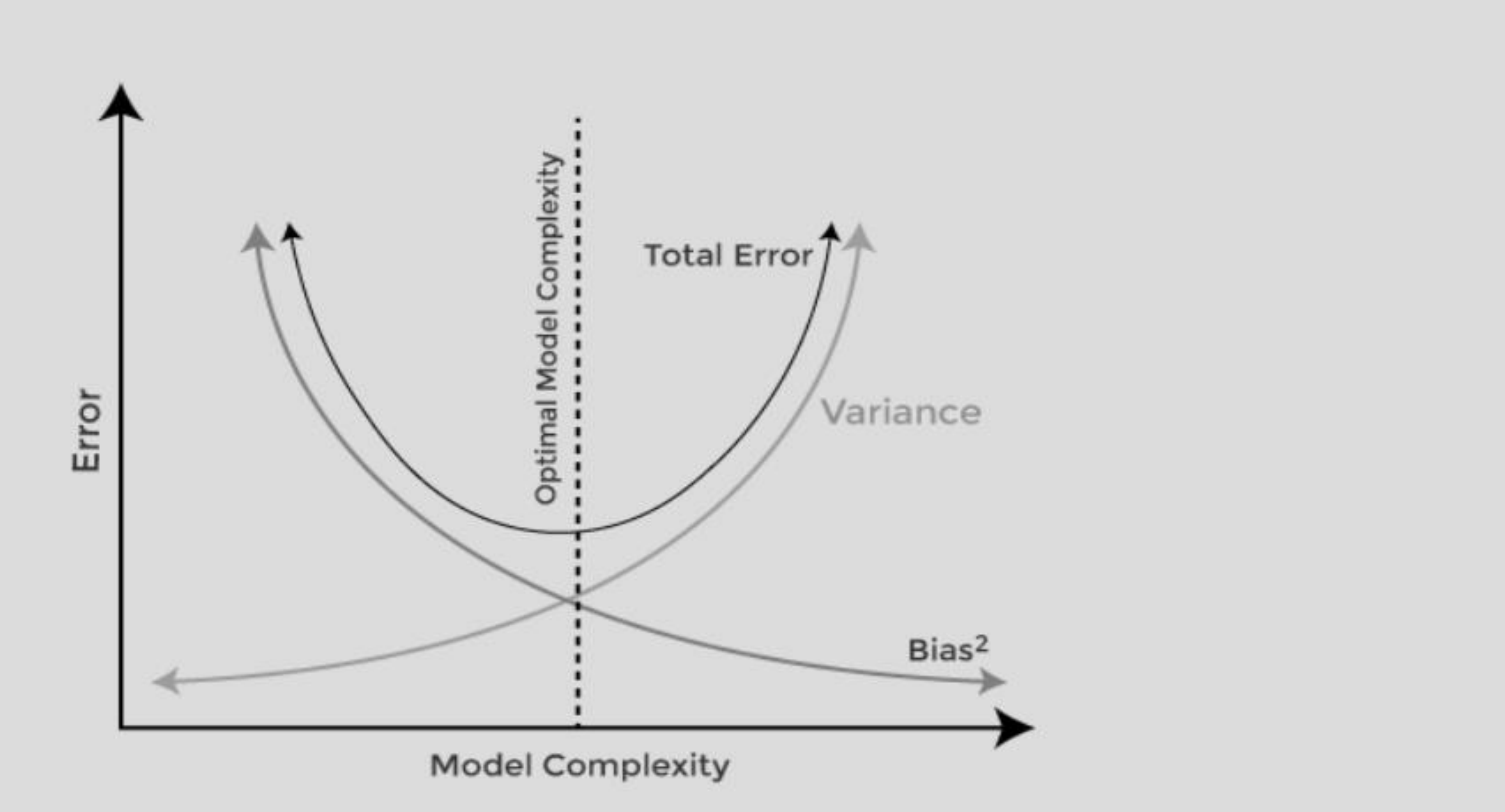
What we want to know is coefficients, the betas, and these can be estimated with Ordinary Least Squares (OLS). Since we just expressed the equation in matrix, we can use the matrix property to find the betas.

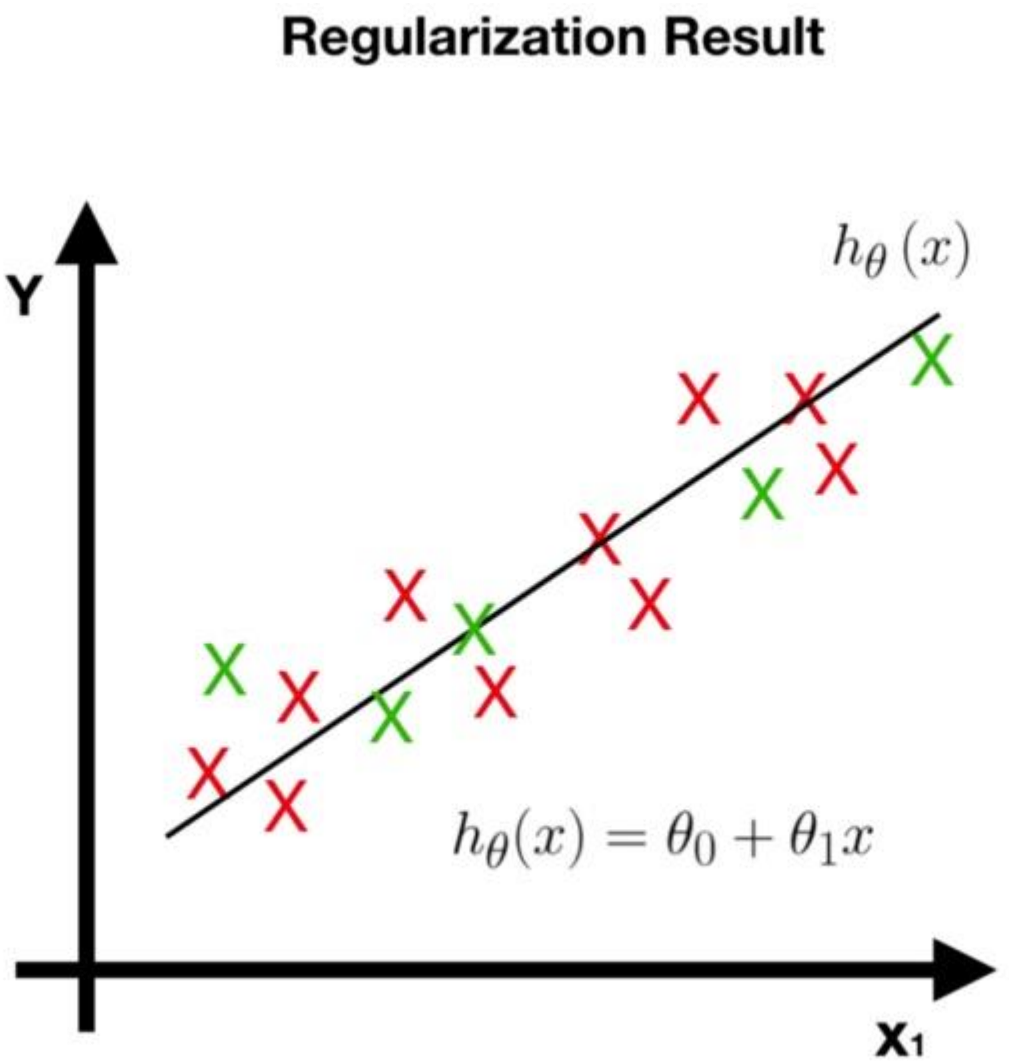
$$\hat{B} = (X^T X)^{-1} X^T Y \;\; where\, \hat{B}\, is\, the\, estimate$$

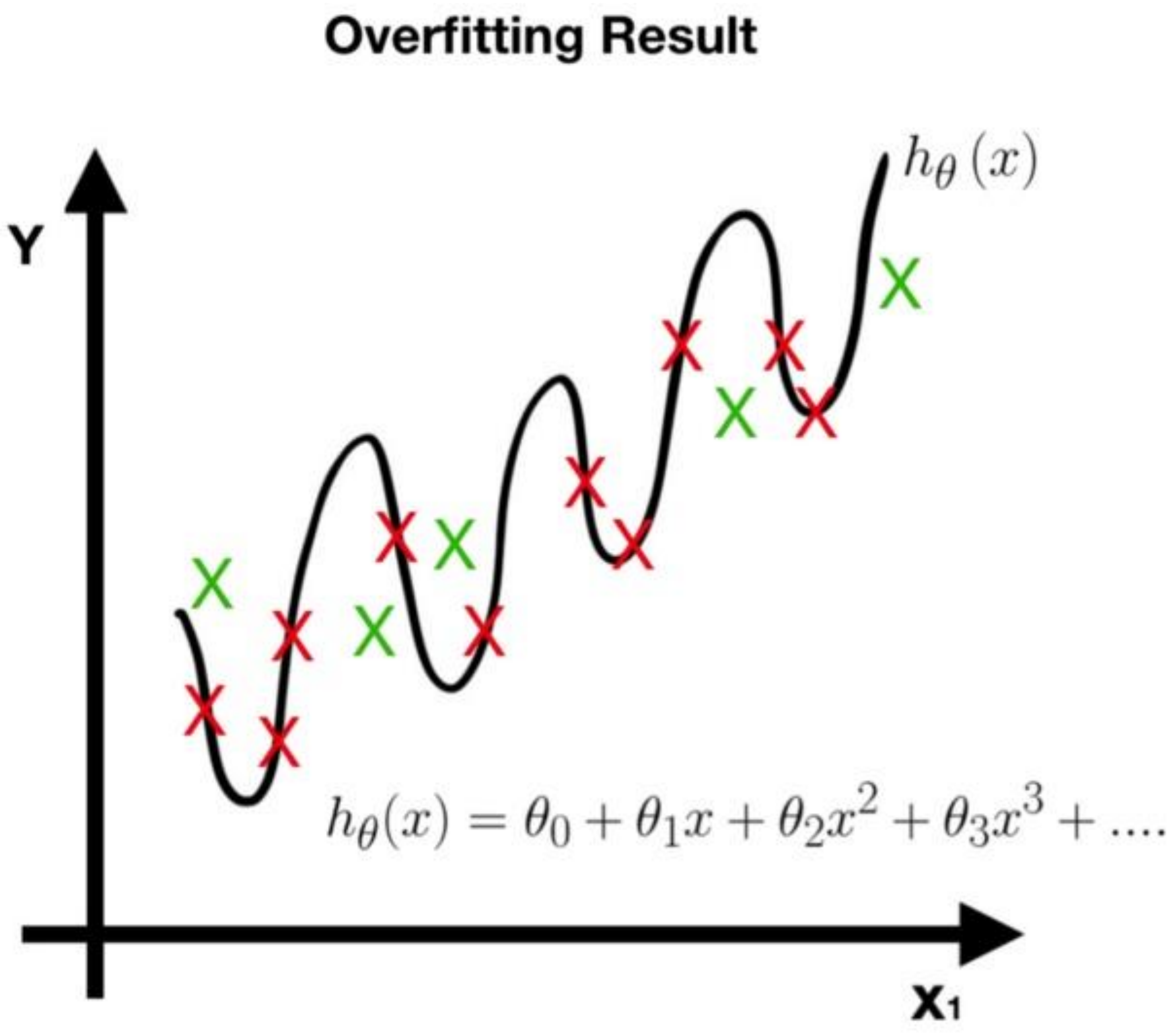
Bias-Variance Trade off

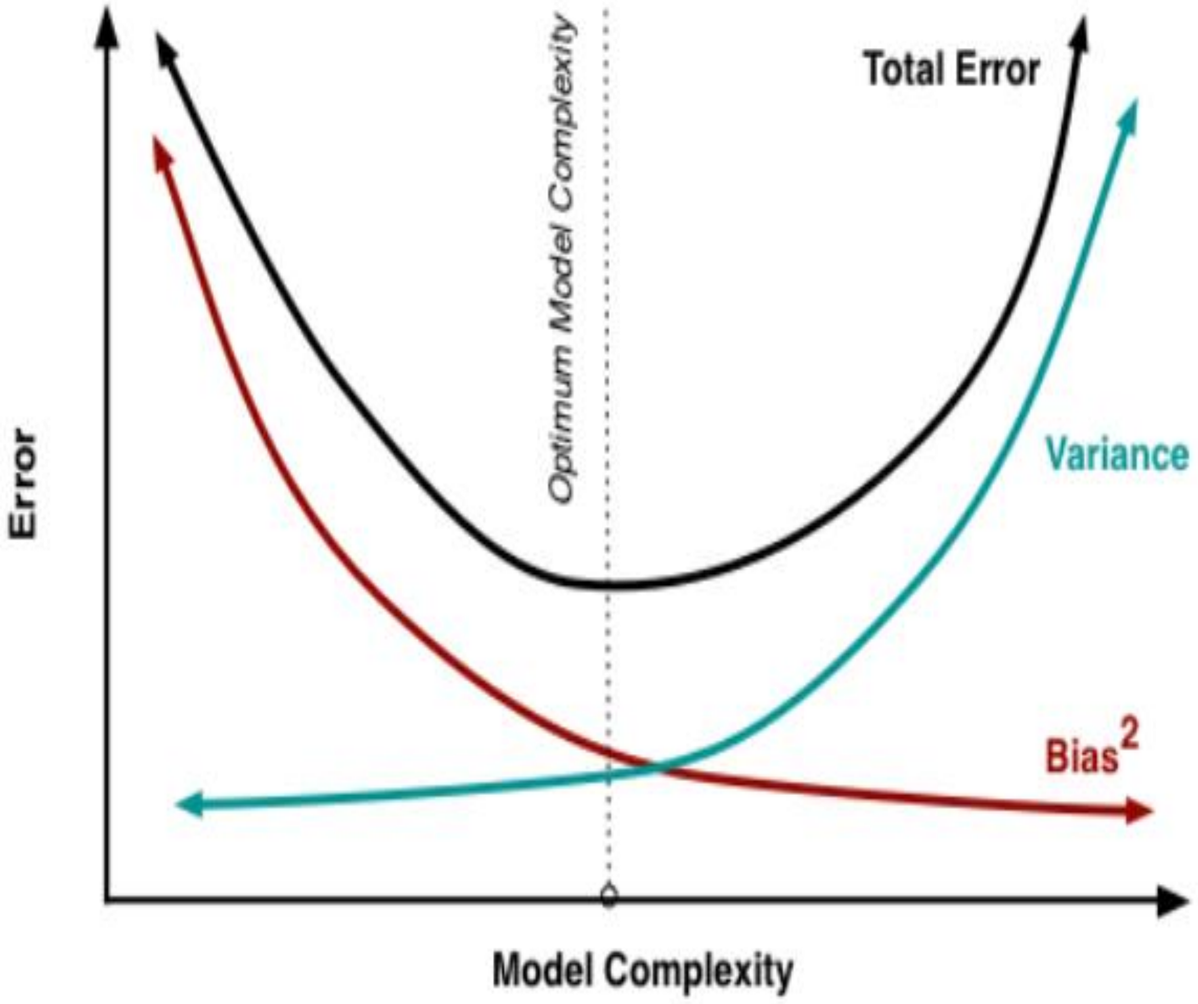
- **Bias** is the simplifying assumptions made by the model to make the target function easier to approximate.
- **Variance** is the amount that the estimate of the target function will change, given different training data.
- **Bias-variance trade-off** is the sweet spot where our machine model performs between the errors introduced by the bias and the variance.


High-bias model: Underfits the data
training score: R^2 = 0.70
validation score: R^2 = 0.74
High-variance model: Overfits the data
training score: R^2 = 0.98
validation score: R^2 = -1.8e+09

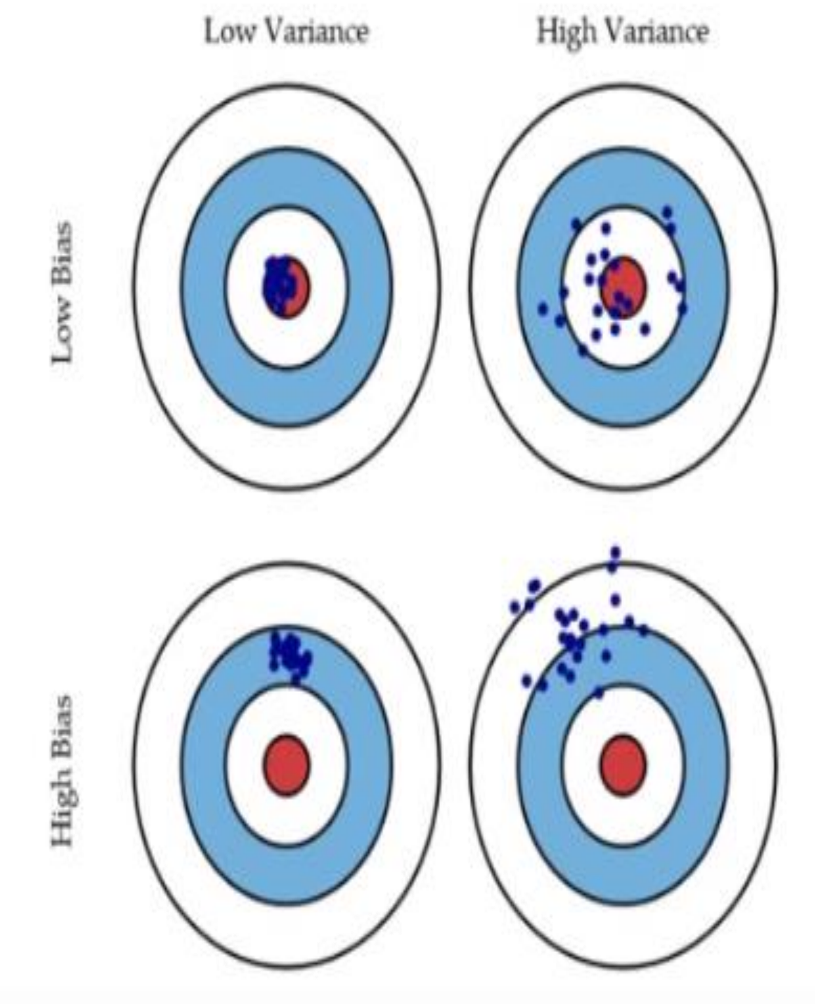

Error
Optimal Model Complexity
Total Error
Variance
Bias2
Model Complexity


Regularization Result
$h_\theta(x)$
Y
$h_\theta(x) = \theta_0 + \theta_1 x$
X1


Overfitting Result
$h_\theta(x)$
Y
$h_\theta(x) = \theta_0 + \theta_1 x + \theta_2 x^2 + \theta_3 x^3 + ...$
X1



Error
Optimum Model Complexity
Total Error
Variance
Bias^2
Model Complexity



Low Variance
High Variance
Low Bias
High Bias

Math for Ridge Regression

OLS method basically finds the β's to minimize Residual Sum of Squares (RSS).

$$RSS = \sum_{i=1}^{n}(y_i - \hat{y})^2 = \sum_{i=1}^{n}(y_i - \sum_{j=1}^{p} x_{ij}\beta_j)^2$$

What Ridge Regression does is penalize RSS by adding another term and for searching the minimization.

$$RSS\ with\ Penalty = \sum_{i=1}^{n}(y_i - \sum_{j=1}^{p} x_{ij}\beta_j)^2 + \lambda \sum_{j=1}^{p} \beta_j^2\ where\ \lambda\ is\ a\ constant$$

Then, our goal becomes to minimize the term. If you are not familiar with the rightmost term, refer to this article about Euclidean Norm

$$\min_{\beta \in R^p} \{ \frac{1}{N} \| Y - X\beta \|_2^2 + \lambda \| \beta \|_2^2 \} \; where \; N \; is \; number \; of \; cases$$

We can find the β's by utilizing properties of matrix.

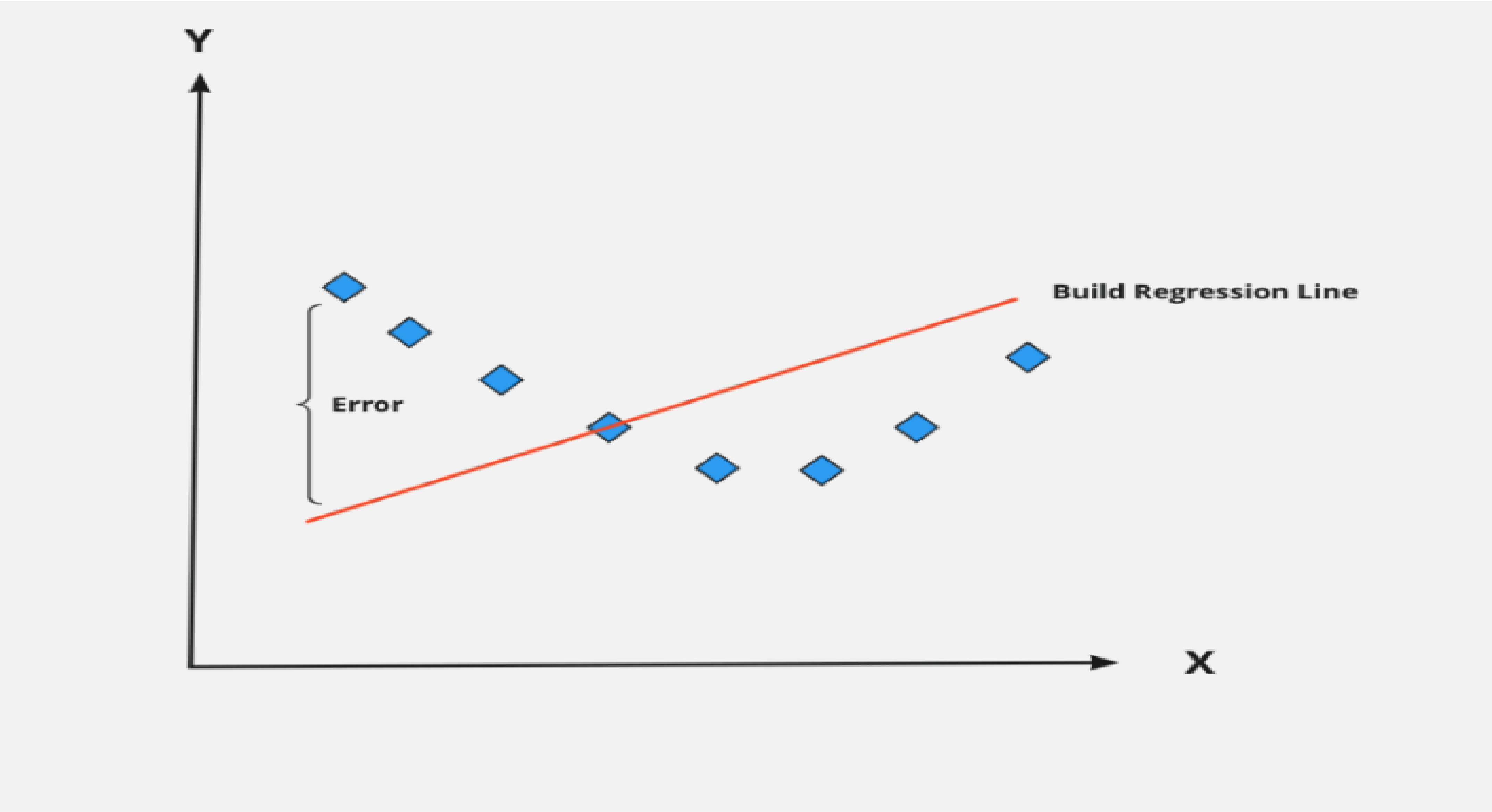
$$\hat{\beta}_{ridge} = (X^T X + \lambda I_p)^{-1} X^T Y$$

We can iterate different λ values to find the best fit for a model.

Math for Lasso Regression

As with Ridge Regression, OLS method is modified for Lasso Regression. In fact, only difference is the penalty term.

$$RSS\ with\ Penalty = \sum_{i=1}^{n}(y_i - \sum_{j=1}^{p} x_{ij}\beta_j)^2 + \lambda \sum_{j=1}^{p} |\beta_j|$$



Regression Line Error

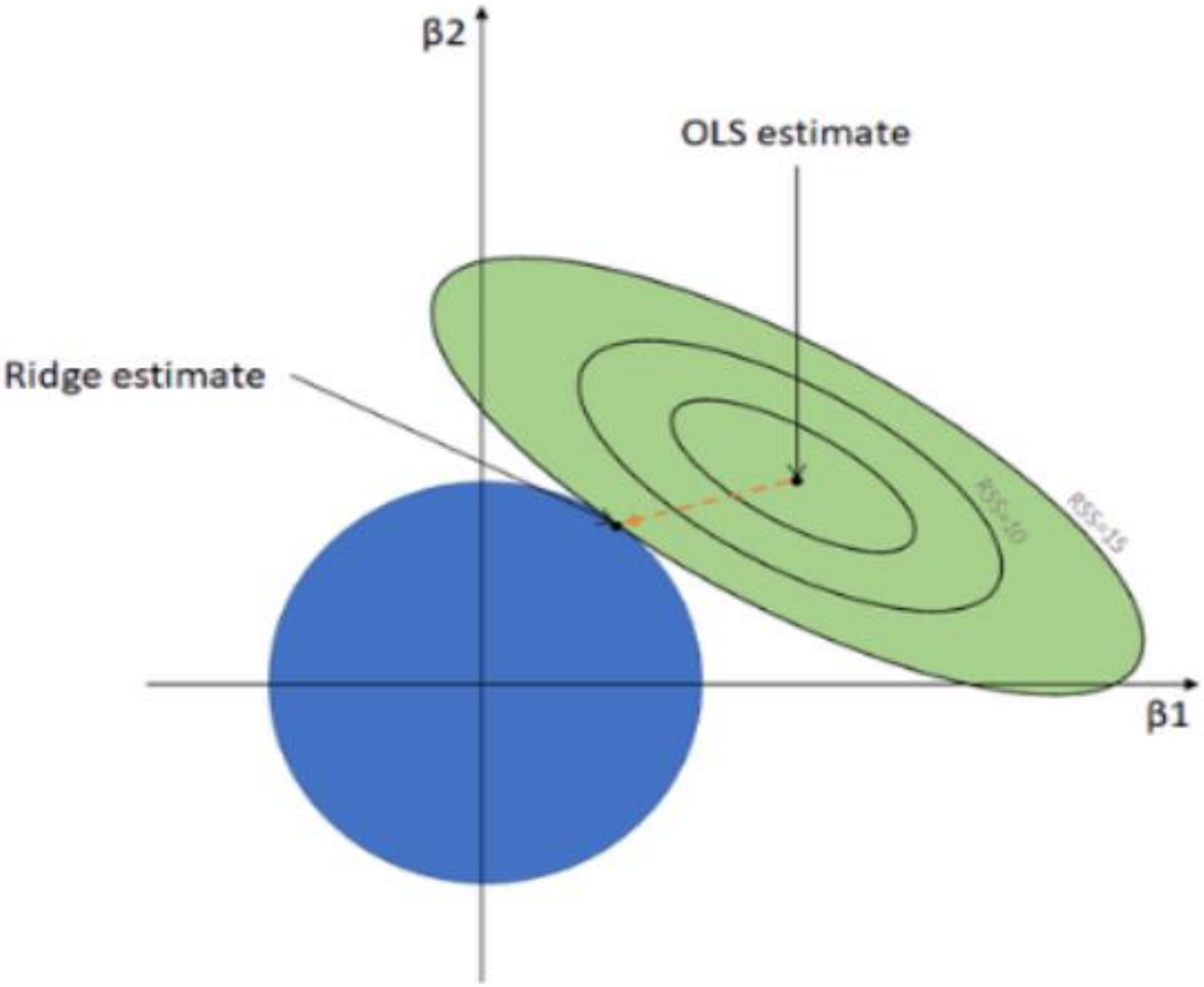
Vectorized Version

The vector norm is nothing but the following definition.

$$\|B\|_2 = \sqrt{\beta_0^2 + \beta_1^2 + \cdots + \beta_p^2}$$

Vectorized Version

The subscript '2' is as in 'L2 norm'. We only care about the **L2 norm** at this moment, so we can construct the equation we've already seen.



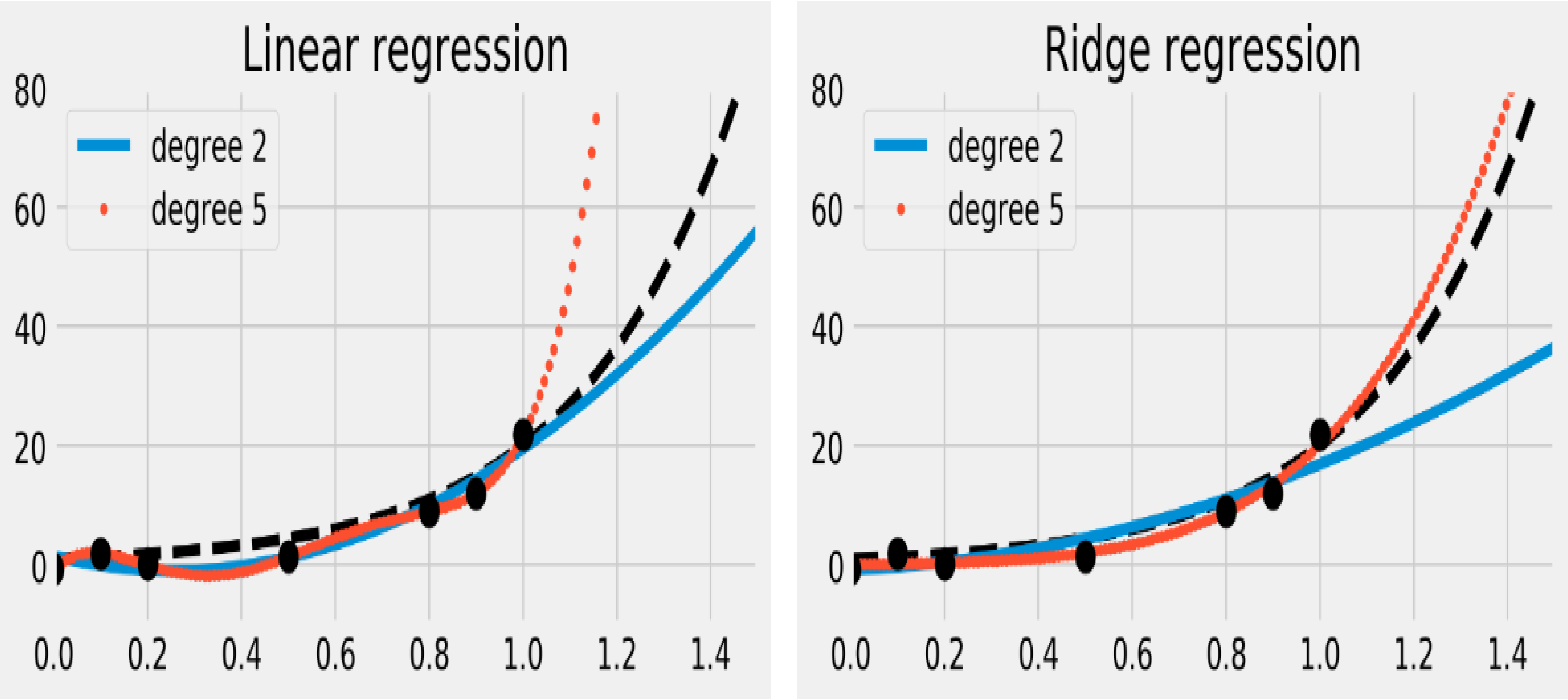
Ordinary Least Squares (OLS)

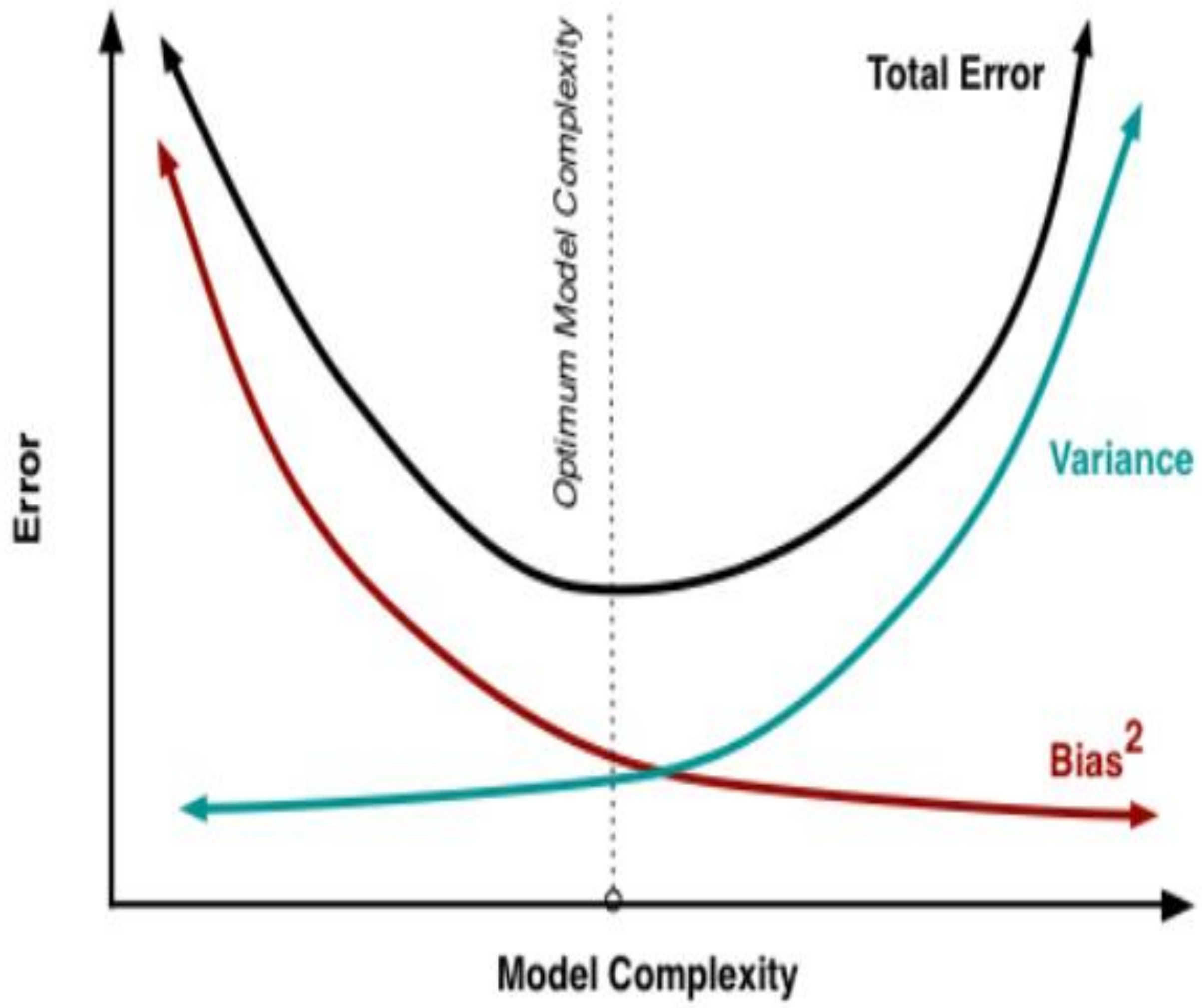
Implementation of Linear Regression

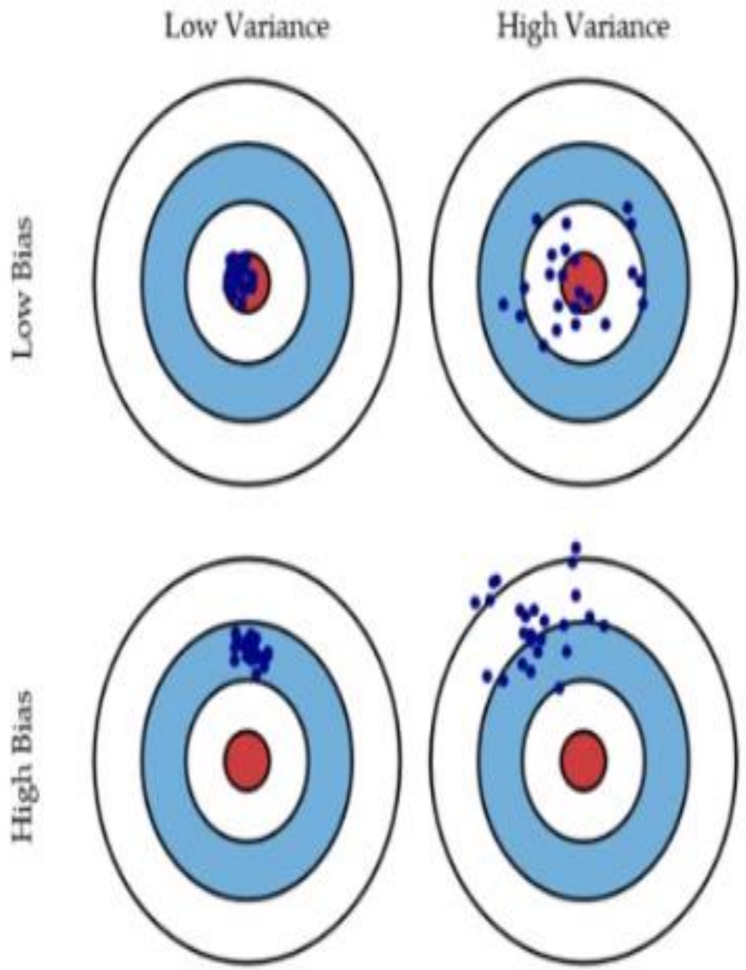
Let's try to implement the linear regression in both scikit-learn and statsmodels.

	Scikit-learn	Statsmodels
Intercept_	Includes intercept_ by default	We need to add the intercept
Model Evaluation	The score method in scikit-learn gives us the accuracy of the model	It shows many statistical results like p-value, F-test, Confidential Interval
Regularization	It uses "L2" by default, We can also set the parameter to "None" if we want	It does not use any regularization parameter.
Advantage	It has a lot of parameters and is easy to use.	Used to infer the population parameters from sample statistics.

Implementation of Linear Regression using sklearn,


Linear regression
degree 2
degree 5
80
60
40
20
0
0.0
0.2
0.4
0.6
0.8
1.0
1.2
1.4
Ridge regression
degree 2
degree 5
80
60
40
20
0
0.0
0.2
0.4
0.6
0.8
1.0
1.2
1.4


Error
Optimum Model Complexity
Total Error
Variance
Bias^2
Model Complexity


Low Variance
High Variance
Low Bias
High Bias

Regularization

Regularization Term

$$J(\theta) = \frac{1}{2m}\sum_{i=1}^{m}\left(h_\theta(x^{(i)}) - y^{(i)}\right)^2 + \boxed{\boxed{\lambda}\sum_{j=1}^{n}\theta_j{}^2}$$

start at θ_1

Regularization Parameter

Ridge Regression

Ridge regression uses the mean squared error loss function and applies L2 Regularization. Its cost function *J(θ)* is given as

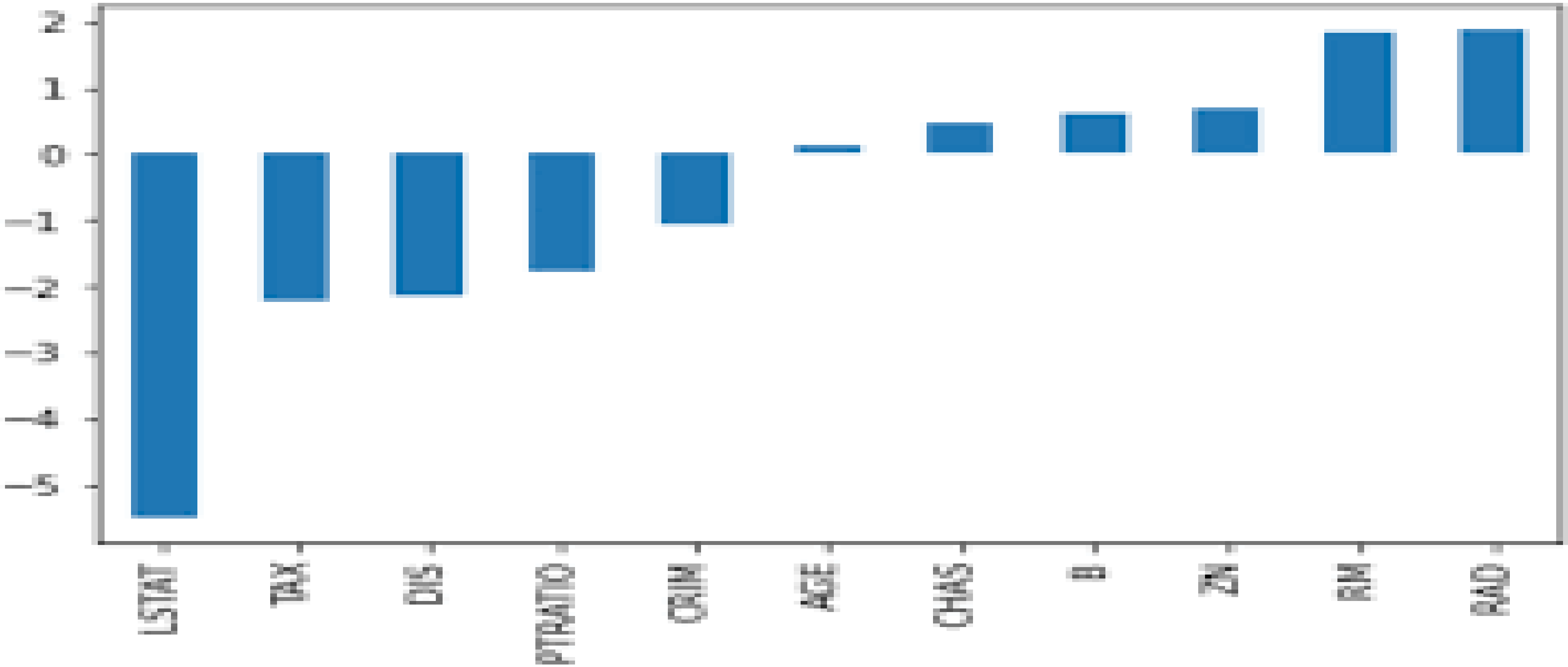
$$J(\theta) = \frac{1}{m}\sum_{i=1}^{m}(y - \hat{y})^2 + \lambda \sum_{j=1}^{n} w_j^2$$

where,

$\frac{1}{m}\sum_{i=1}^{m}(y - \hat{y})^2$ is the Mean Squared error (loss function)

$\lambda \sum_{j=1}^{n} w_j^2$ is the penalty (L2 Regularization)

Now, substitute $\hat{y}$ as $wx_i + b$.


2
1
0
−1
−2
−3
−4
−5
LSTAT
TAX
DIS
PTRATIO
CRIM
AGE
CHAS
B
ZN
RM
RAD

Ridge Regression

$$Regularization\ (L2) = Loss\ Function + \lambda \sum_{i=1}^{m} w_i^2$$

Lasso Regression

Lasso regression uses the same mean squared error loss function and this applies L1 Regularization and will repeat the same steps as Ridge. The cost function of Lasso Regression *J(θ)* is given as

$$J(\theta) = \frac{1}{m}\sum_{i=1}^{m}(y - \hat{y})^2 + \lambda\sum_{j=1}^{n}|w_j|$$

where

$\lambda \sum_{j=1}^{n}|w_j|$ is the penalty (L1 Regularization).

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

$$Regularization\ (L1) = \frac{1}{m}\sum_{i=1}^{n}(y - \hat{y})^2 + \lambda \sum_{j=1}^{m}|w_i|$$

Recall: Covariance

$$\operatorname{cov}(x, y) = \frac{\sum_{i=1}^{n} (x_i - \bar{X})(y_i - \bar{Y})}{n-1}$$

Interpreting Covariance

cov(X,Y) > 0 $\longrightarrow$ X and Y are positively correlated

cov(X,Y) < 0 $\longrightarrow$ X and Y are inversely correlated

cov(X,Y) = 0 $\longrightarrow$ X and Y are independent

Correlation coefficient

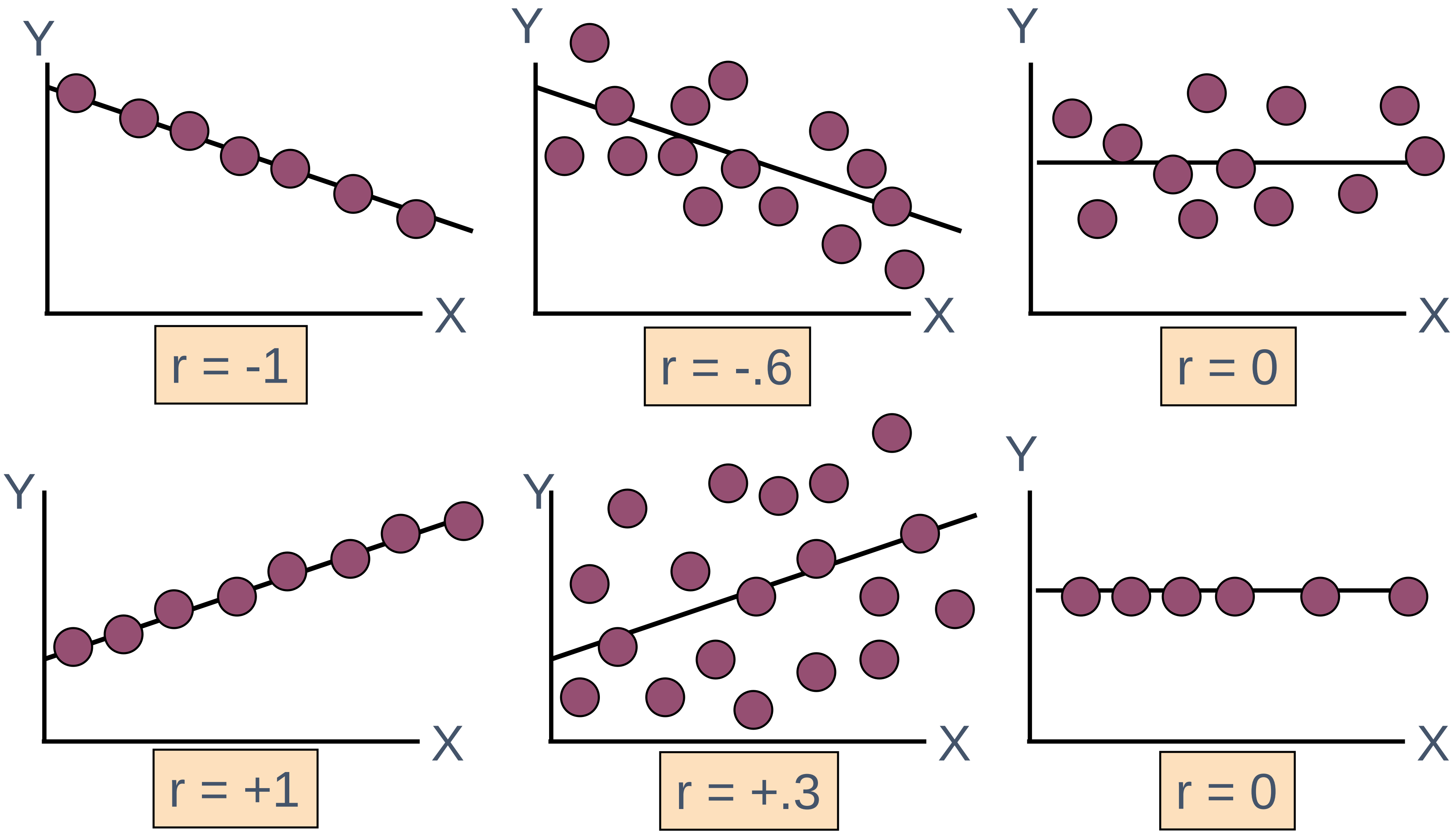
- Pearson's Correlation Coefficient is standardized covariance (unitless):

$$r = \frac{\mathrm{cov}ariance(x, y)}{\sqrt{\mathrm{var}\, x}\sqrt{\mathrm{var}\, y}}$$

Correlation

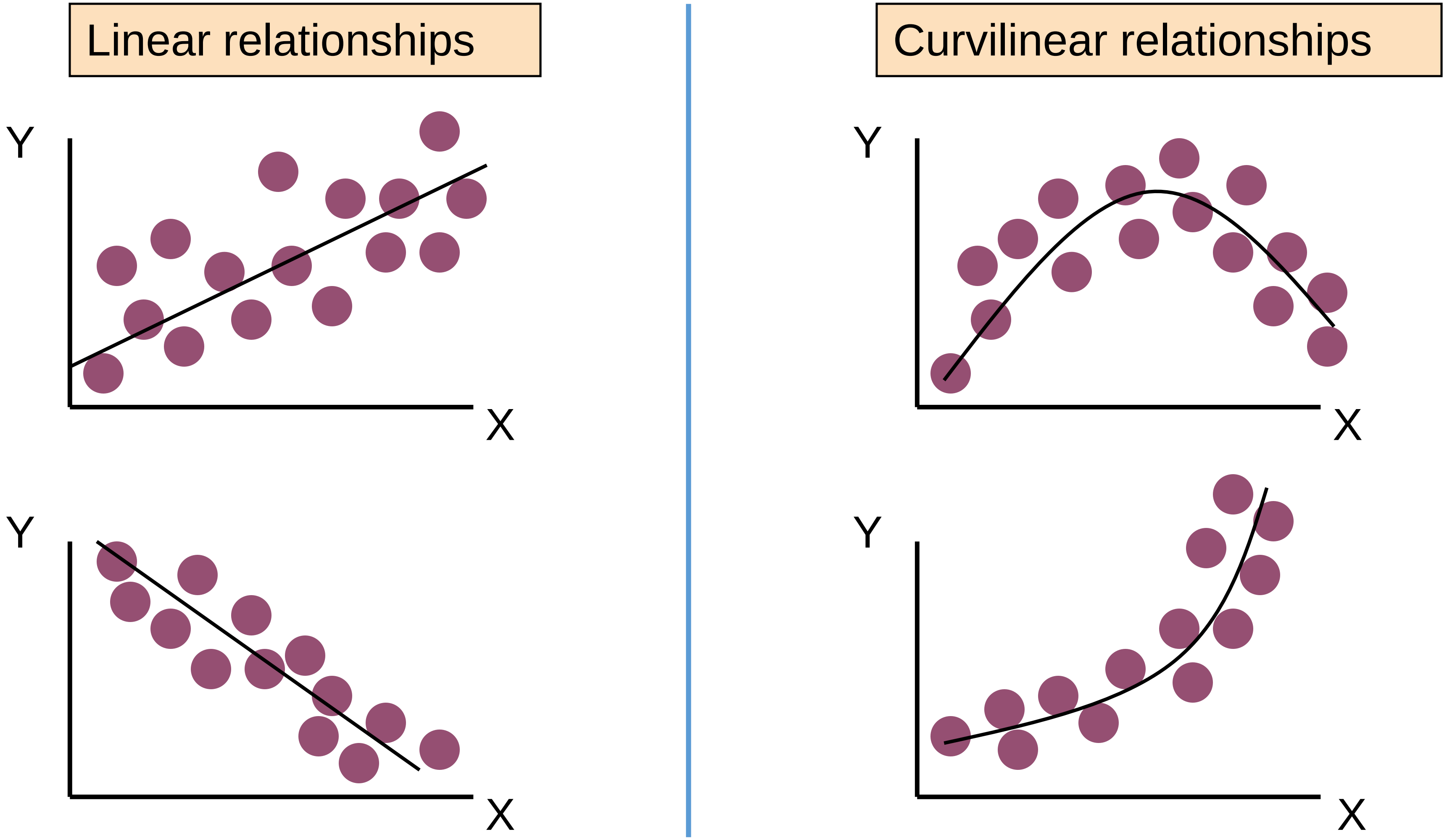
- Measures the relative strength of the *linear* relationship between two variables
- Unit-less
- Ranges between –1 and 1
- The closer to –1, the stronger the negative linear relationship
- The closer to 1, the stronger the positive linear relationship
- The closer to 0, the weaker any positive linear relationship

Scatter Plots of Data with Various Correlation Coefficients



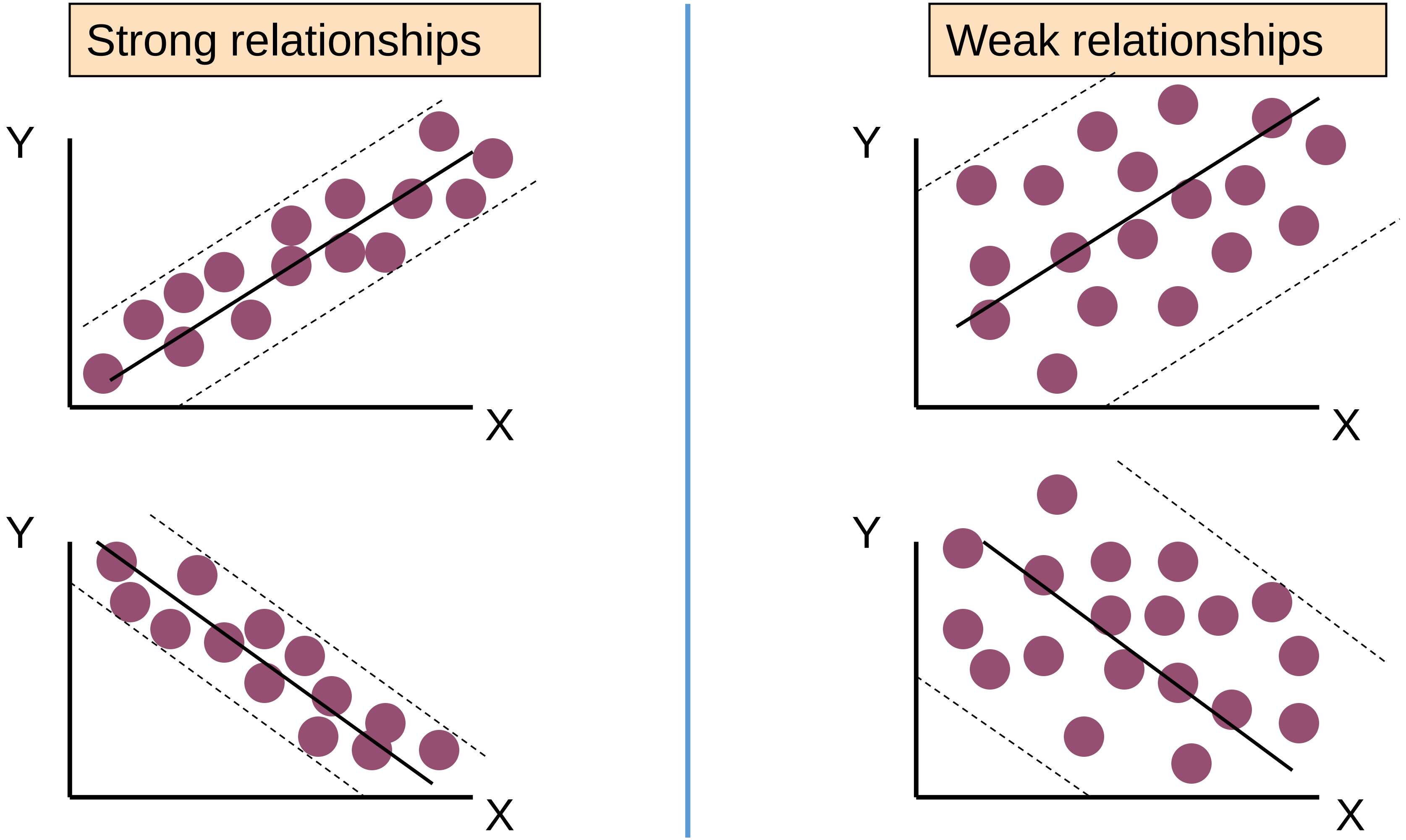
Slide from: Statistics for Managers Using Microsoft® Excel 4th Edition, 2004 Prentice-Hall

Linear Correlation



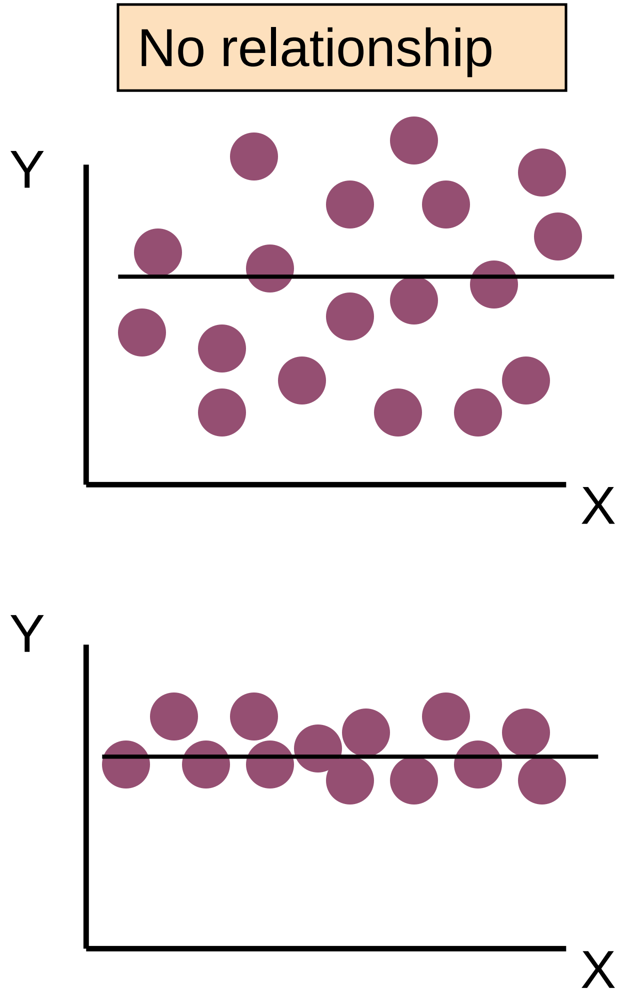
- Slide from: Statistics for Managers Using Microsoft® Excel 4th Edition, 2004 Prentice-Hall

Linear Correlation



▪Slide from: Statistics for Managers Using Microsoft® Excel 4th Edition, 2004 Prentice-Hall

Linear Correlation



▪Slide from: Statistics for Managers Using Microsoft® Excel 4th Edition, 2004 Prentice-Hall

Calculating by hand…

$$\hat{r} = \frac{\operatorname{cov}ariance(x,y)}{\sqrt{\operatorname{var} x}\sqrt{\operatorname{var} y}} = \frac{\dfrac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{n-1}}{\sqrt{\dfrac{\sum_{i=1}^{n}(x_i - \bar{x})^2}{n-1}}\sqrt{\dfrac{\sum_{i=1}^{n}(y_i - \bar{y})^2}{n-1}}}$$

Simpler calculation formula…

$$\hat{r} = \frac{\dfrac{\sum_{i=1}^{n}(x_i-\bar{x})(y_i-\bar{y})}{\cancel{n-1}}}{\sqrt{\dfrac{\sum_{i=1}^{n}(x_i-\bar{x})^2}{\cancel{n-1}}}\sqrt{\dfrac{\sum_{i=1}^{n}(y_i-\bar{y})^2}{\cancel{n-1}}}} =$$

$$\frac{\sum_{i=1}^{n}(x_i-\bar{x})(y_i-\bar{y})}{\sqrt{\sum_{i=1}^{n}(x_i-\bar{x})^2}\sqrt{\sum_{i=1}^{n}(y_i-\bar{y})^2}} = \frac{SS_{xy}}{\sqrt{SS_x SS_y}}$$

Numerator of covariance

$$\hat{r} = \frac{SS_{xy}}{\sqrt{SS_x SS_y}}$$

Numerators of variance

Distribution of the correlation coefficient:

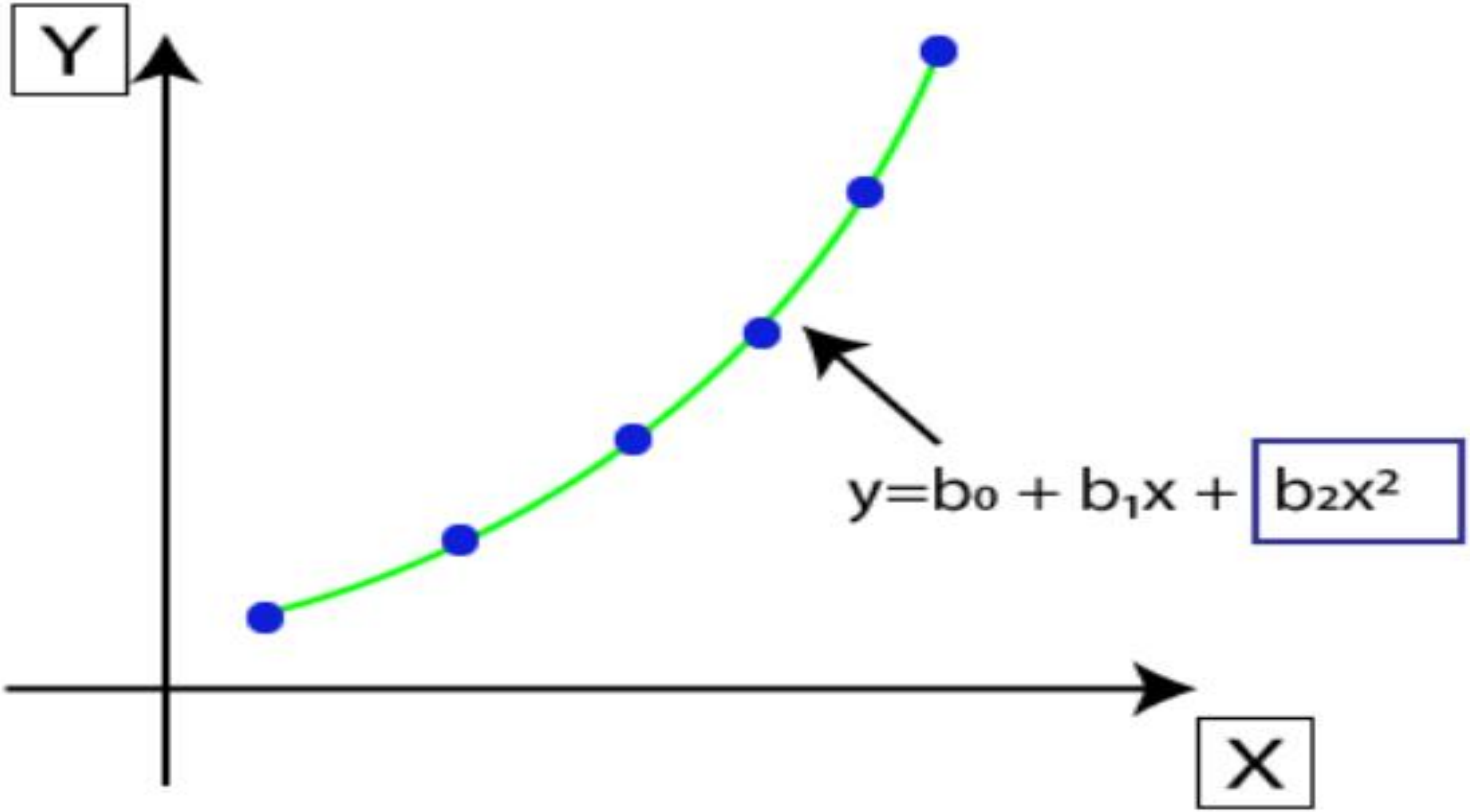
$$SE(\hat{r}) = \sqrt{\frac{1-r^2}{n-2}}$$

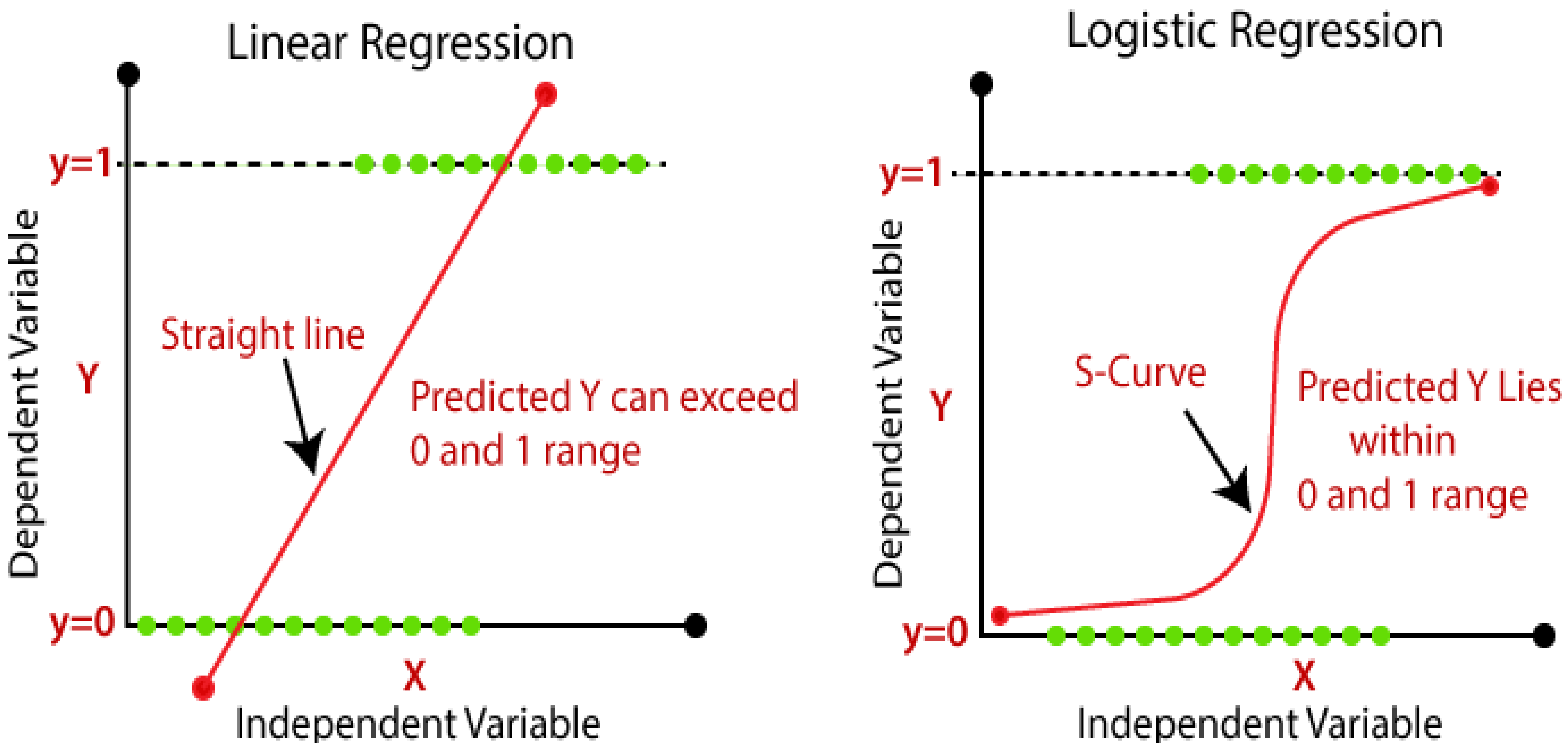
The sample correlation coefficient follows a T-distribution with n-2 degrees of freedom (since you have to estimate the standard error).

*note, like a proportion, the variance of the correlation coefficient depends on the correlation coefficient itself→substitute in estimated r

Polynomial Regression:

- Polynomial Regression is a type of regression which models the **non-linear dataset** using a linear model.
- It is similar to multiple linear regression, but it fits a non-linear curve between the value of x and corresponding conditional values of y.
- Suppose there is a dataset which consists of datapoints which are present in a non-linear fashion, so for such case, linear regression will not best fit to those datapoints. To cover such datapoints, we need Polynomial regression.
- **In Polynomial regression, the original features are transformed into polynomial features of given degree and then modeled using a linear model.** Which means the datapoints are best fitted using a polynomial line.


Y
$y=b_0 + b_1x + \boxed{b_2x^2}$
X


Linear Regression
y=1
y=0
Dependent Variable
Y
Straight line
Predicted Y can exceed
0 and 1 range
X
Independent Variable
Logistic Regression
y=1
y=0
Dependent Variable
Y
S-Curve
Predicted Y Lies
within
0 and 1 range
X
Independent Variable