Regression

Module 4

Regression

Predicting / Forecasting a real-numbered value

A classic statistical method that is being used for Machine Learning - with ML, more complex relationships between variables can be found.

Types:

- Linear
- Polynomial

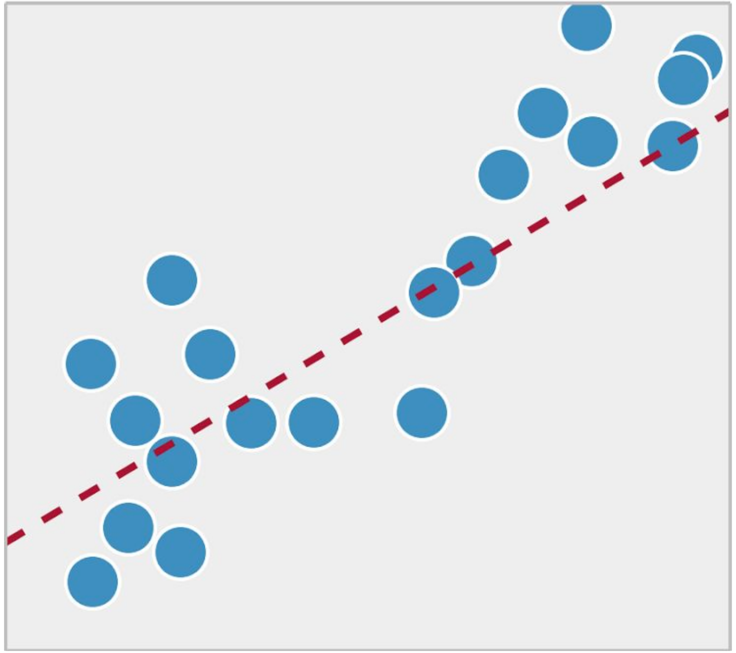
Use Cases:

- Customer Lifetime Value
- Energy Consumption
- Insurance Pricing

Key Metrics:

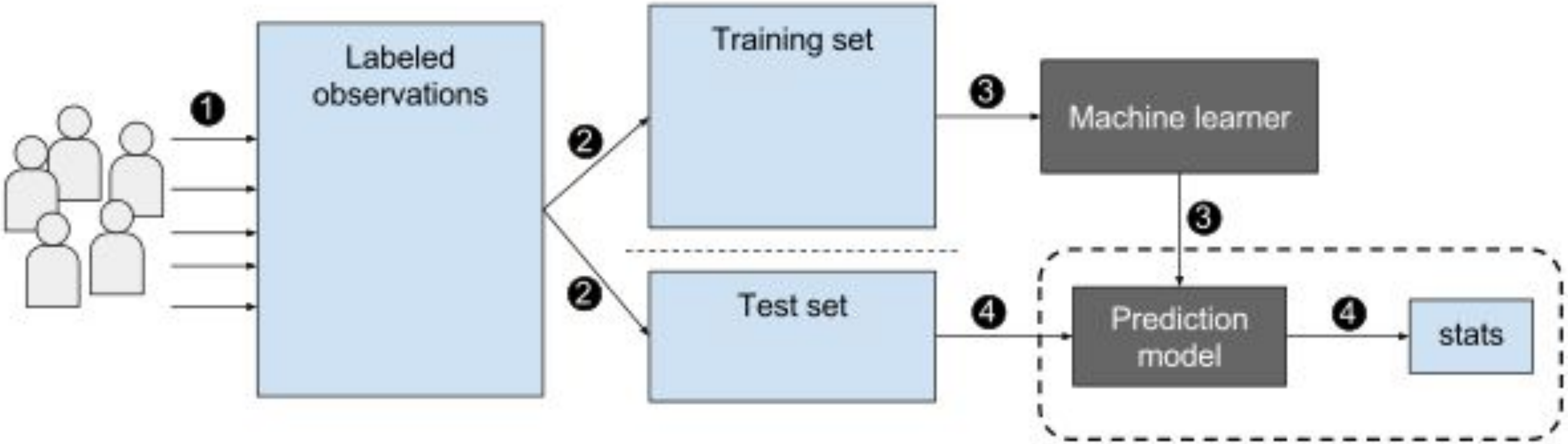
- R-Squared
- Mean Absolute Error
- Root Mean Squared Error

Supervised Learning: Regression



First, A Review

How does a machine learn?



Algorithms

Regression

Algorithms

- Linear Regression
  - Simple Linear Regression
  - Multiple Linear Regression
- Polynomial Regression
- *Logistic Regression*

Linear Regression: Overview

- Uses a linear function to predict the (average) numerical value of Y for a given value of X using a straight line (called the regression line)

$$y = \beta_0 + \beta_1 x$$

- Prediction is performed by plugging in the X values into the linear function

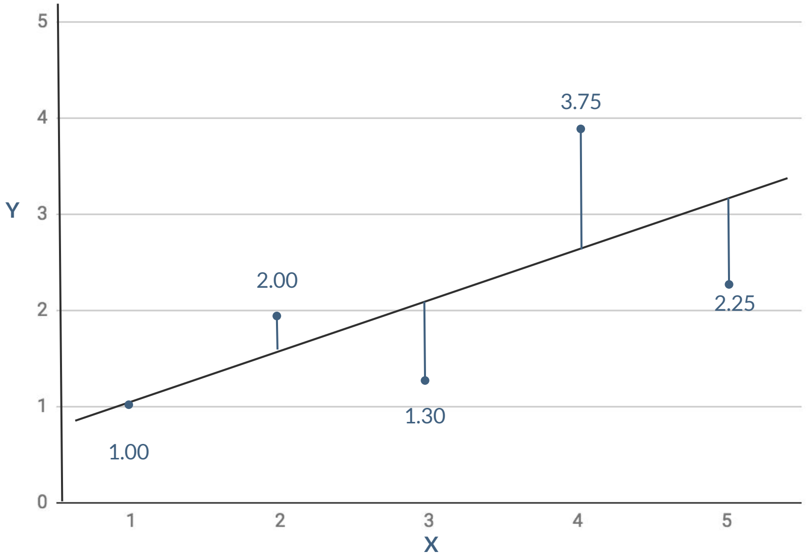
- Simple Linear Regression
  - 1 predictor variable

- Multiple Linear Regression
  - 2 or more predictor variables

Linear Regression: Regression Line

What is meant by the “Best Fitting Line”?

The line that minimizes the sum of squared errors prediction



X	Y	Y'	Y-Y'	$(Y\text{-}Y')^2$
1.00	1.00	1.210	-0.210	0.044
2.00	2.00	1.635	0.365	0.133
3.00	1.30	2.060	-0.760	0.578
4.00	3.75	2.485	1.265	1.600
5.00	2.25	2.910	-0.660	0.436

Linear Regression: Simple Linear Regression

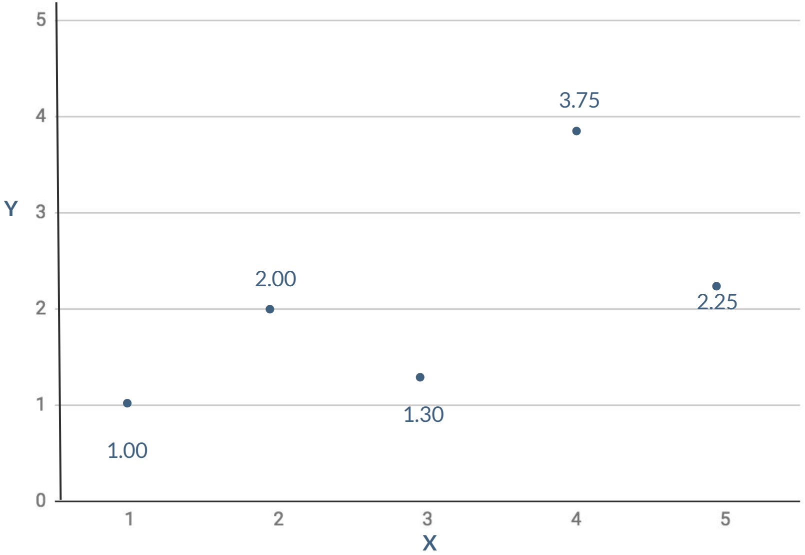
- Continuous response is modeled as a linear combination of the features:

  $$y = \beta_0 + \beta_1 x$$

  - Where:
    - y - target/predicted value
    - β_0 - y-intercept
    - β_1 - slope or coefficient

- Predicts scores on the target variable (Y) from values of the predictor variable (X)
  - Only ONE predictor variable

X	Y
1.00	1.00
2.00	2.00
3.00	1.30
4.00	3.75
5.00	2.25



Linear Regression: Simple Linear Regression

Formula:

$$y = \beta_0 + \beta_1 x$$

Where:

- y - target/predicted value
- β_0 - y-intercept
- β_1 - slope or coefficient

Example:

$$y = 0.785 + 0.425x$$

For x = 1:

$$y = 0.785 + (0.425)(1)$$
$$y = 1.21$$

For x = 2:

$$y = 0.785 + (0.425)(2)$$
$$y = 1.64$$

Linear Regression: Simple Linear Regression

ANALYTIKS

Computing the regression line

Typically computed with Statistical Software

$$m = r\left(\frac{s_y}{s_x}\right)$$

$$b = \bar{y} - m\bar{x}$$

$m = \beta_1$

$b = \beta_0$

M_X	M_Y	S_X	S_Y	r
3.00	2.06	1.581	1.072	0.627

Where:

- M_X = mean of X
- M_Y = mean of Y
- S_X = standard deviation of X
- S_Y = standard deviation of Y
- r = correlation between X and Y

$$s_x = \sqrt{\frac{\sum_{i=1}^{n}(x_i - \bar{x})^2}{n-1}}$$

$n =$ The number of data points

$\bar{x} =$ The mean of the x_i

$x_i =$ Each of the values of the data

$$r = \frac{1}{(n-1)} \sum \frac{(X-\mu_X)(Y-\mu_Y)}{\sigma_X \sigma_Y}$$

Linear Regression: Multiple Linear Regression

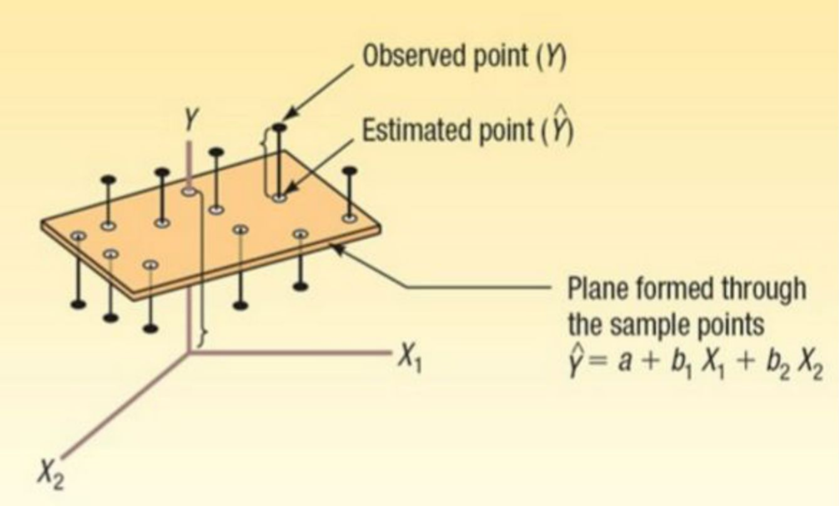
- Concepts in simple linear regression can be extended to multiple predictors

- Formula for n predictor variables

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x^2 + \ldots + \beta_{n-1} x^{n-1} + \beta_n x^n$$

- Formula for 2 predictor variables:

$$y = \beta_0 + \beta_1 x^1 + \beta_2 x^2$$


Observed point (Y)
Estimated point ($\hat{Y}$)
Y
Plane formed through the sample points
$\hat{Y} = a + b_1 X_1 + b_2 X_2$
X_1
X_2

Linear Regression: Coefficients Intuition

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2$$

$$\text{Salary} = \beta_0 + \beta_1(\text{Age}) + \beta_2(\text{Job Complexity})$$

β_1 The marginal effect on salary for every 1 year increase in age

β_2 The marginal effect on salary for every additional unit increase in complexity rating

Linear Regression: Standardization (Standard Scaler)

The regression equation is simpler if variables are standardized so that their means are equal to 0 and standard deviations are equal to 1, for then $\beta_1 = r$ and $\beta_0 = 0$.

- Subtract by the mean
- Then, divide by the standard deviation

This makes the regression line:

$$Z_Y' = (r)(Z_X)$$

Where:

- Z_Y' - predicted standard score for Y
- r - correlation
- Z_X - standardized score for X

Note: the slope of the regression equation for standardized variables is r

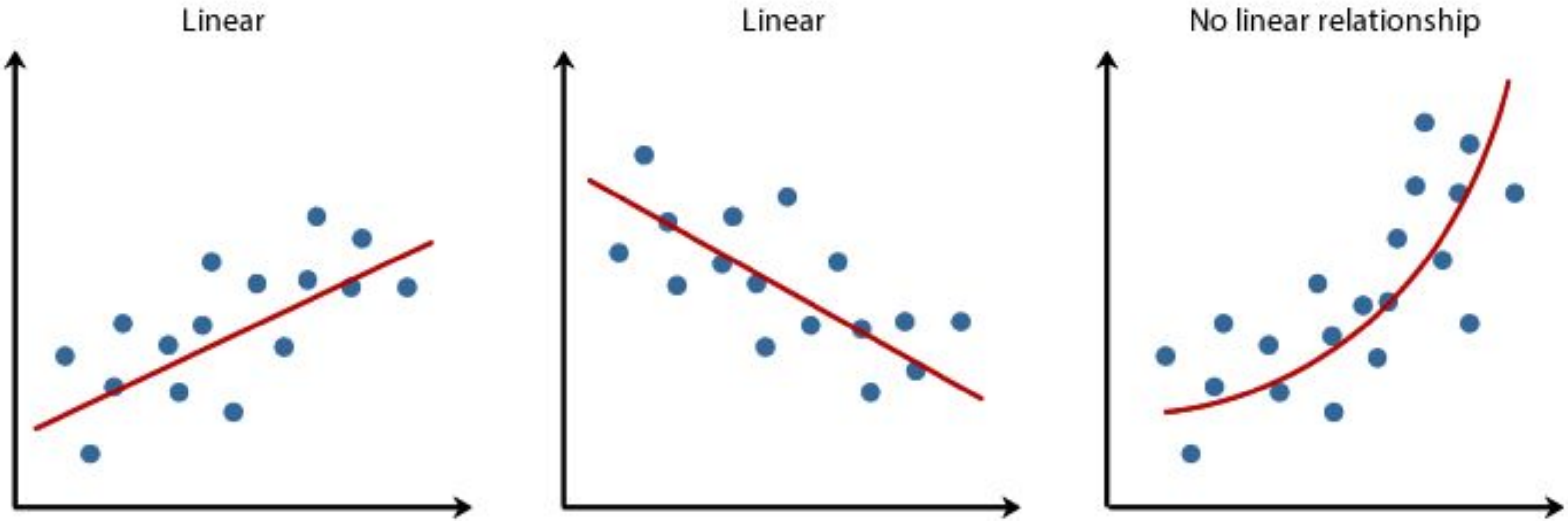
Why standardize?

Simplifies the linear function (units of regression coefficients are the same)

Linear Regression: Key Requirements

- Linear relationship
- Errors have the same variance (homoscedasticity)
- No or little high intercorrelations (multicollinearity) among the predictors
- Variables have normal distribution

Linear Regression Req’t: Linear Relationship

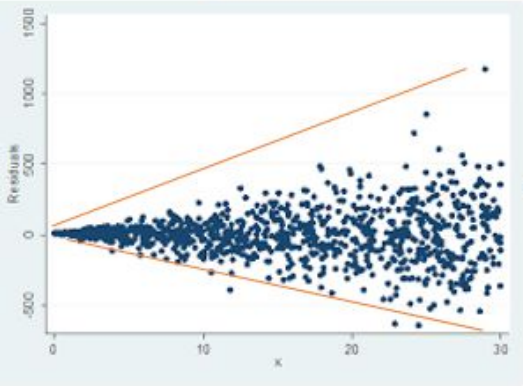


Copyright 2014. Laerd Statistics.

Linear Regression Req’t: Homoscedasticity

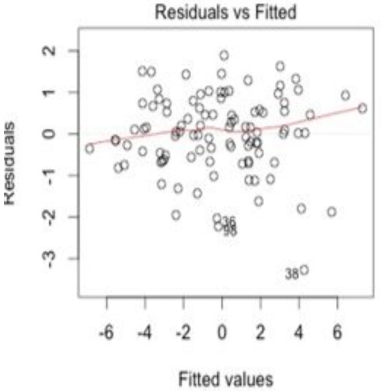
Errors have the same variance as you move along regression line. If residuals plot shows pattern, then there is heteroscedasticity and data is not fit for linear regression

Plot Showing Heteroskedasticity

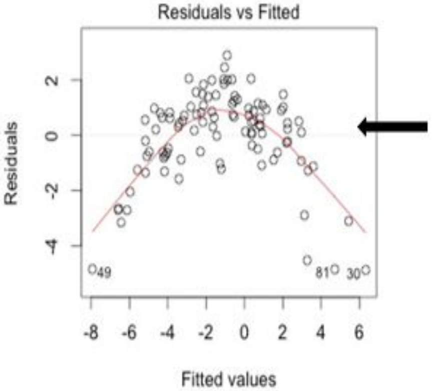


Funnel Shape

Data is Linear

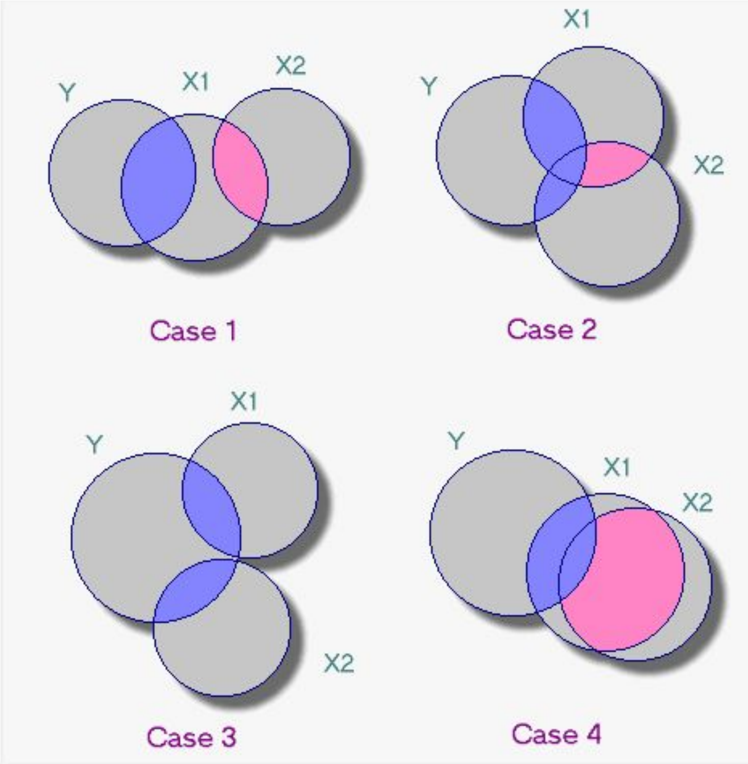


Data is Non-Linear



Nonlinear Curve

Linear Regression Req't: No Multicollinearity



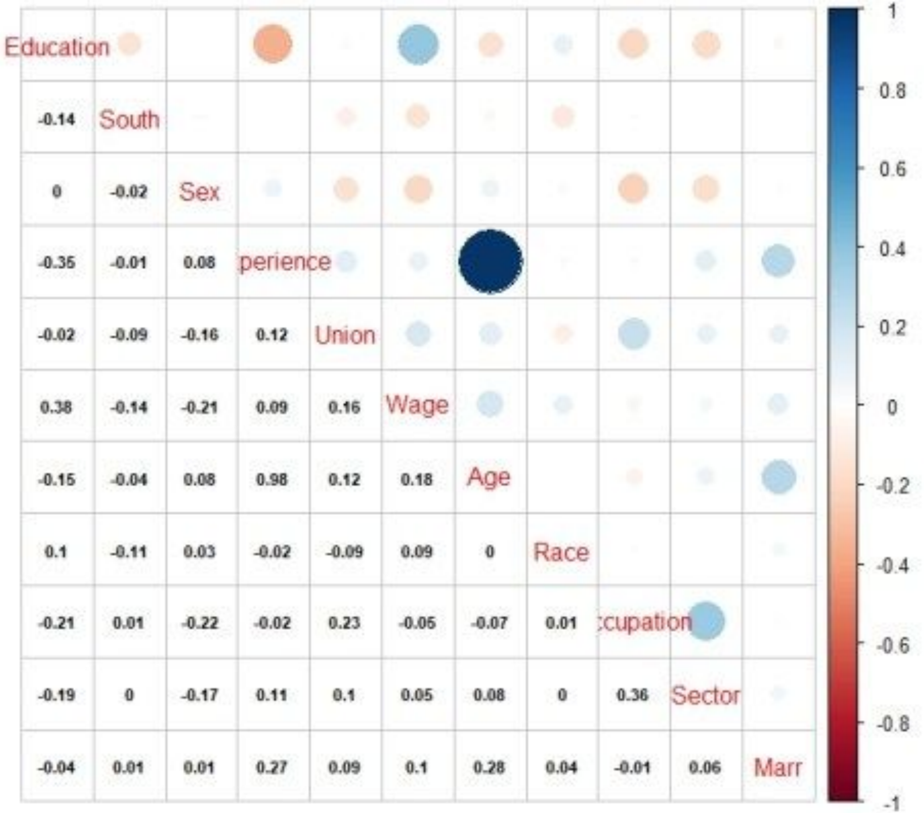
Linear Regression Req’t: No Multicollinearity

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2$$

$$\text{Salary} = \beta_0 + \beta_1(\text{Age}) + \beta_2(\text{Experience})$$

How do you really know how one affects salary?

Linear Regression Req’t: No Multicollinearity
Correlation Plot



Education										
-0.14	South									
0	-0.02	Sex								
-0.35	-0.01	0.08	perience							
-0.02	-0.09	-0.16	0.12	Union						
0.38	-0.14	-0.21	0.09	0.16	Wage					
-0.15	-0.04	0.08	0.98	0.12	0.18	Age				
0.1	-0.11	0.03	-0.02	-0.09	0.09	0	Race			
-0.21	0.01	-0.22	-0.02	0.23	-0.05	-0.07	0.01	:cupation		
-0.19	0	-0.17	0.11	0.1	0.05	0.08	0	0.36	Sector	
-0.04	0.01	0.01	0.27	0.09	0.1	0.28	0.04	-0.01	0.06	Marr

Linear Regression Req’t: Multicollinearity

Impact:

- Higher variance of the beta / coefficients
- Beta estimates/coefficients could be in opposite directions
- Removal of one beta results in a significantly different model

If the model is to be used for prediction only, you can choose to do nothing about collinearity. But if coefficients are important,

- drop one variable or
- combine the two related variables into one variable
- Use Principal Component Analysis (PCA)

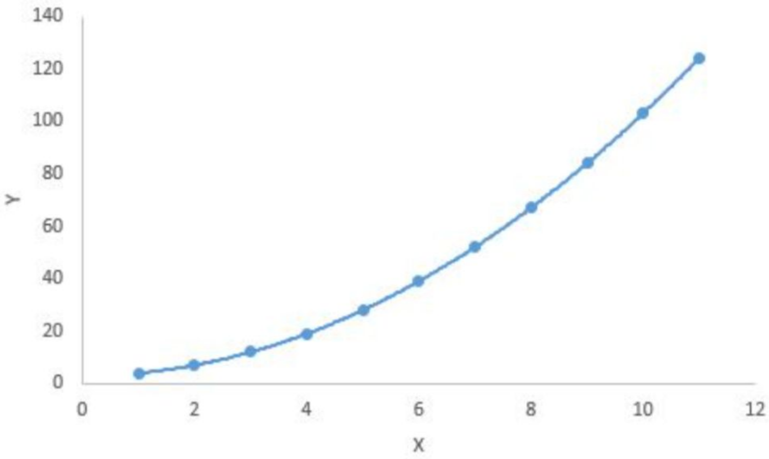
Linear Regression: Pros and Cons

+	-
● Fast prediction	● Often inappropriately used to model non-linear relationships
● Allows understanding of relationship between variables	● Only looks at the mean of the target variable
● Works well if your data has a clear linear trend	● Sensitive to outliers ○ Standardize ○ z-scores - remove those below -3.29 & above 3.29
	● Data must be independent

LAB: Simple Linear &
Multiple Linear Regression

Polynomial Regression: Overview

- Fits the data into a regression line/curve using a polynomial equation
  - Exponent of predictor variable >1
  - $y = \beta_0 + \beta_1 x^2$
  - $y = \beta_0 + \beta_1 x^2 + \beta_1 x^3 + \ldots + \beta_1 x^n$
- Prediction: plug in the X values into the function
- Compared to Linear Regression, the best-fitting line in polynomial regression is a curve rather than a straight line
- In Python, this is done using PolynomialFeatures pre-processor



Polynomial Regression: Implementation

- A simple linear regression can be extended by constructing polynomial features from the coefficients

- In the standard linear regression case:

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2$$

- If we want to fit a paraboloid to the data instead of a plane, we can combine the features in second-order polynomials, so that the model looks like this:

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \beta_3 x_1 x_2 + \beta_4 x_1^2 + \beta_5 x_2^2$$

- This is still a linear model! Create a new variable z:

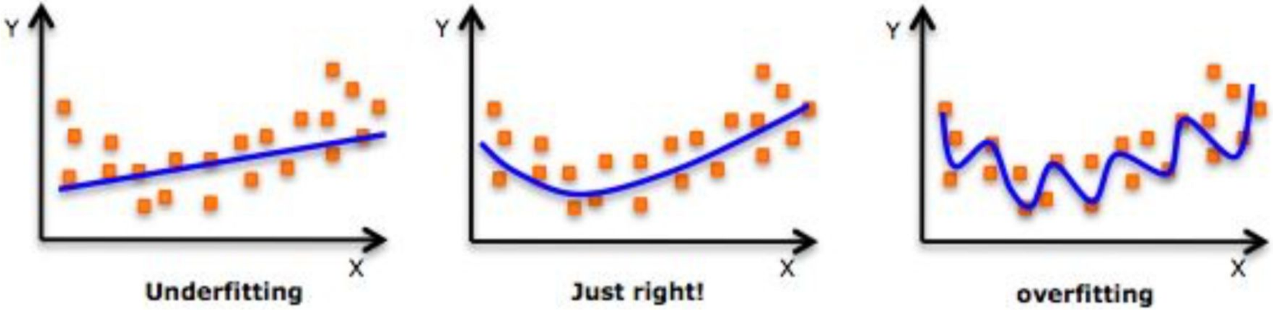
$$z = [x_1, x_2 , x_1 x_2 , x_1^2 , x_2^2]$$

$$y = \beta_0 + \beta_1 z_1 + \beta_2 z_2 + \beta_3 z_3 + \beta_4 z_4 + \beta_5 z_5$$

- In Python, this is done using PolynomialFeatures pre-processor

Polynomial Regression: Overfitting

- There might be a temptation to fit a higher degree polynomial to get lower error
- Can result in overfitting
- Plot the relationships to see the fit and focus on making sure that the curve fits the nature of the problem
- Tip: keep the degree low (< 3)



Polynomial Regression: Pros and Cons

+	-
● Fast prediction ● Allows understanding of relationship between variables ● Have moderate flexibility of shapes ● Useful when curvilinear effect is present in the dataset	● Finds polynomial relationship within the dataset ● Poor interpolatory and extrapolatory properties ● Sensitive to outliers

Prediction Errors

Prediction Errors

- Irreducible Error / Noise
- Bias
- Variance

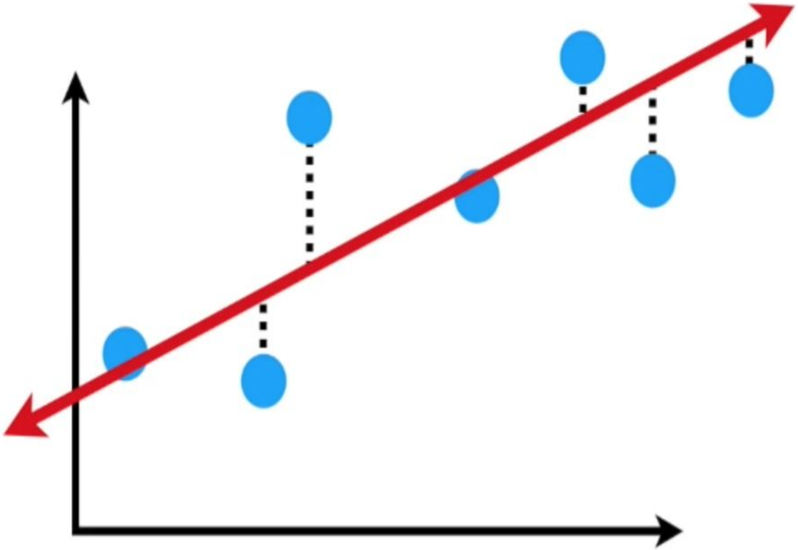
Prediction Errors: Irreducible Error / Noise

- Cannot be reduced regardless of what algorithm is used.

- It is the error likely caused by factors like unknown variables that influence the mapping of the input variables to the output variable.

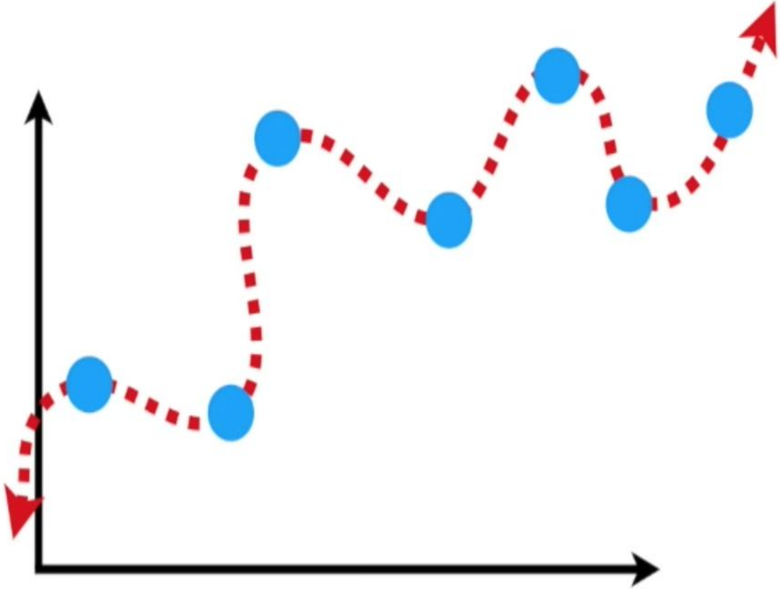
Prediction Errors: Bias

- Simplifying assumptions made by a model to make the target function easier to learn.
- Parametric algorithms generally have a high bias making them fast to learn and easier to understand but generally less flexible with a lower predictive power on complex problems
- Low Bias: Makes less assumptions about the form of the target function.
- High Bias: Makes more assumptions about the form of the target function.



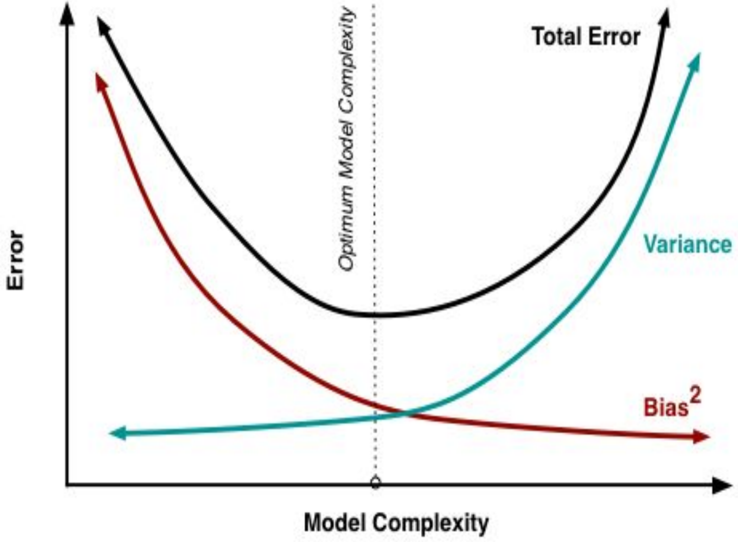
Prediction Errors: Variance

- Amount that the estimate of the target function will change if different training data was used.
- ML algorithms with high variance are strongly influenced by the specifics of the training data. (“Memorizes the Training Data” vs Learning the underlying patterns)
- Low Variance: Makes small changes to the estimate of the target function with changes to the training dataset.
- High Variance: Makes large changes to the estimate of the target function with changes to the training dataset.



Prediction Errors: Bias-Variance Tradeoff

- The goal of any supervised machine learning algorithm is to achieve low bias and low variance
- General Trends:
  - Parametric (Linear) ML algorithms - often have a high bias but a low variance.
  - Non-Parametric (Non-Linear) ML algorithms - often have a low bias but a high variance.
- The trade-off:
  - Increasing the bias will decrease the variance.
  - Increasing the variance will decrease the bias.



Regularization

Regression

Regularization

- Purpose: reduce overfitting (variance)
- Weights are penalized for growing too large
- Balance weights among different features
- Common regularization methods:
    - Ridge (L2) Regularization
    - Lasso (L1) Regularization

Regularization: Ridge Regression (L2)

- Ridge Regression (L2) is a technique for analyzing multiple regression data that suffer from multicollinearity.

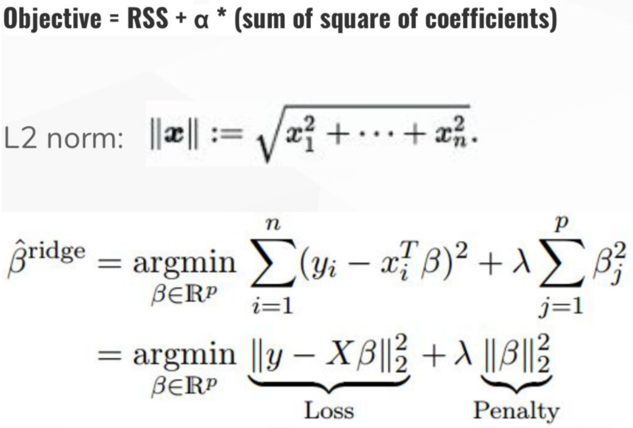
- When multicollinearity occurs, least squares estimates are unbiased, but their variances are large so they may be far from the true value.

- Ridge regularization shrinks the value of coefficients but doesn’t reach zero, which suggests no feature selection

- Regularization also means “desensitization”

Regularization: Ridge Regression (L2)

Regression algorithm combined with L2 regularization (a.k.a., ridge)



Objective = RSS + α * (sum of square of coefficients)

L2 norm: $\|\boldsymbol{x}\| := \sqrt{x_1^2 + \cdots + x_n^2}.$

$$\hat{\beta}^{\text{ridge}} = \underset{\beta\in\mathbb{R}^p}{\operatorname{argmin}} \sum_{i=1}^{n}(y_i - x_i^T\beta)^2 + \lambda\sum_{j=1}^{p}\beta_j^2$$

$$= \underset{\beta\in\mathbb{R}^p}{\operatorname{argmin}} \underbrace{\|y - X\beta\|_2^2}_{\text{Loss}} + \lambda\underbrace{\|\beta\|_2^2}_{\text{Penalty}}$$

1st part: least square term (i.e., prediction error)

2nd part: lambda of the summation of β^2

Least squares:

Sales = y-intercept + slope x Temp

*Minimizes SSE

Ridge Regression:

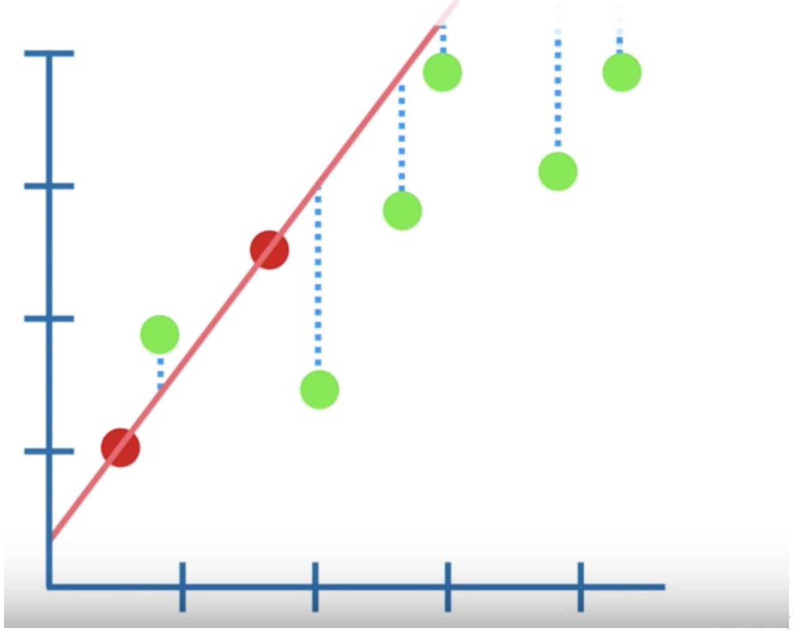
Sales = y-intercept + slope x Temp

+ ƛ x slope^2

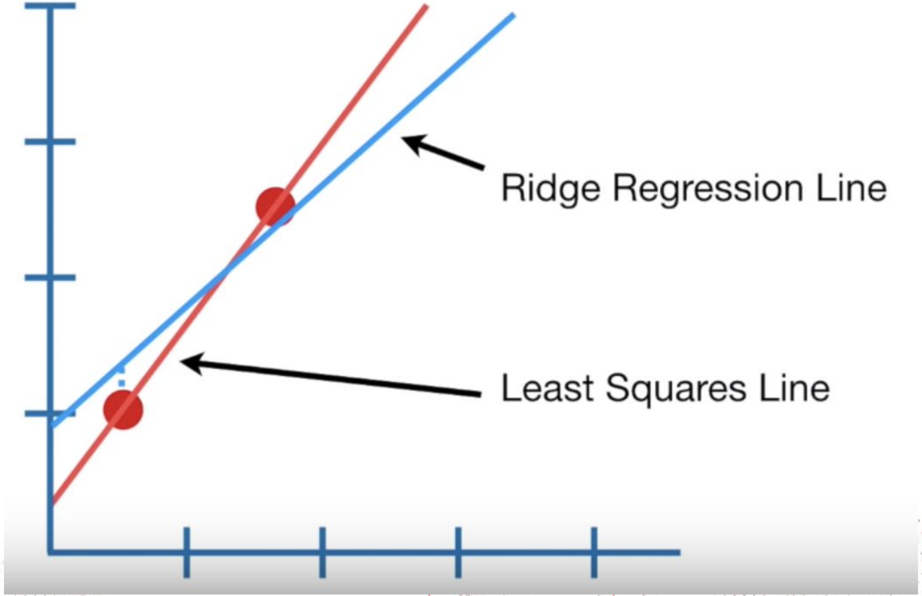
(penalty)

Regularization

This model has high variance, fits training data well but predicts test data badly (high variance)

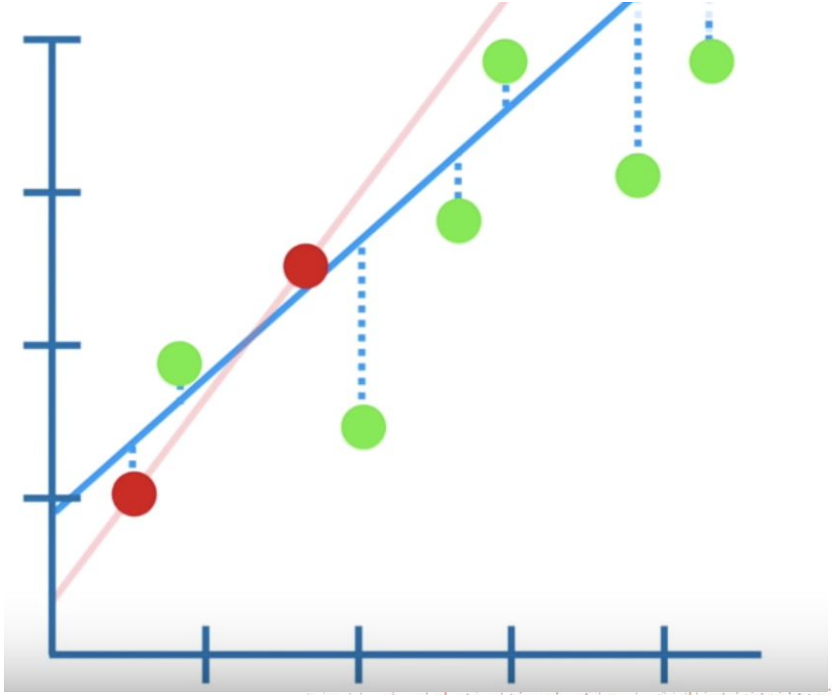


The Ridge Regression line has smaller slope because the slope squared is penalized.



Regularization

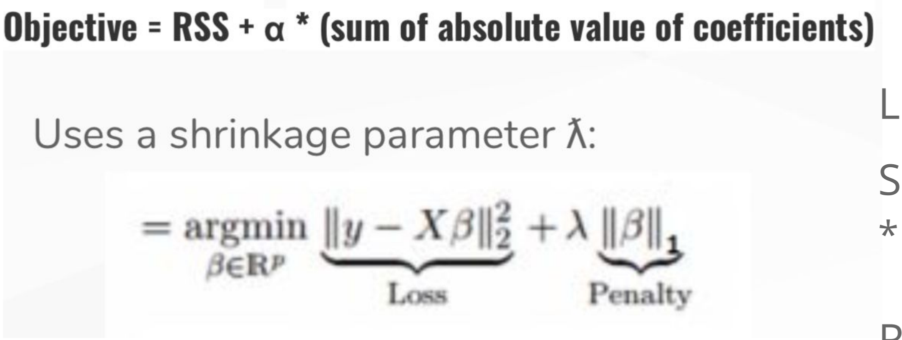
The Ridge Regression line has higher bias, but significantly lower variance



Regularization: Lasso Regression (L1)

- Regression algorithm combined with L1 regularization (a.k.a., lasso)
- Least Absolute Shrinkage and Selection Operator or LASSO (L1)
- Performs both variable selection and regularization in order to enhance the prediction accuracy and interpretability of the statistical model it produces.
- Penalizes the absolute size of the regression coefficients (instead of squares)
- It shrinks coefficients to zero (exactly zero), which certainly helps in feature selection
    - If group of predictors are highly correlated, lasso picks only one of them and shrinks the others to zero

Regularization: Lasso Regression (L1)



Objective = RSS + α * (sum of absolute value of coefficients)

Uses a shrinkage parameter ƛ:

$$= \underset{\beta \in \mathbb{R}^p}{\operatorname{argmin}} \underbrace{\|y - X\beta\|_2^2}_{\text{Loss}} + \lambda \underbrace{\|\beta\|_1}_{\text{Penalty}}$$

1st part: least square term (i.e., prediction error)

2nd part: lambda of the summation of |β|

Least squares:

Sales = y-intercept + slope x Temp

*Minimizes SSE

Ridge Regression:

Sales = y-intercept + slope x Temp

+ ƛ x slope^2 (penalty)

Lasso Regression:

Sales = y-intercept + slope x Temp

+ ƛ x |slope|

LAB: Polynomial Regression and Regularization

Metrics

Regression

Metrics

- R-squared
- Mean Absolute Error
- Weighted Mean Absolute Error
- Root Mean Squared Error

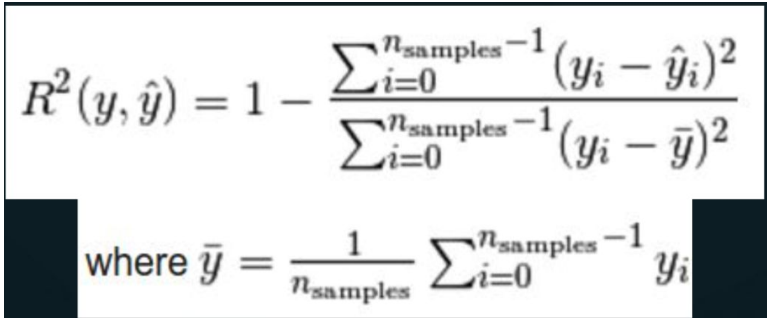
Metrics: R-Squared (R^2)

- Coefficient of Determination

$$R^2 = \frac{\text{Explained variation}}{\text{Total variation}}$$

- % of the Variance explained by the model

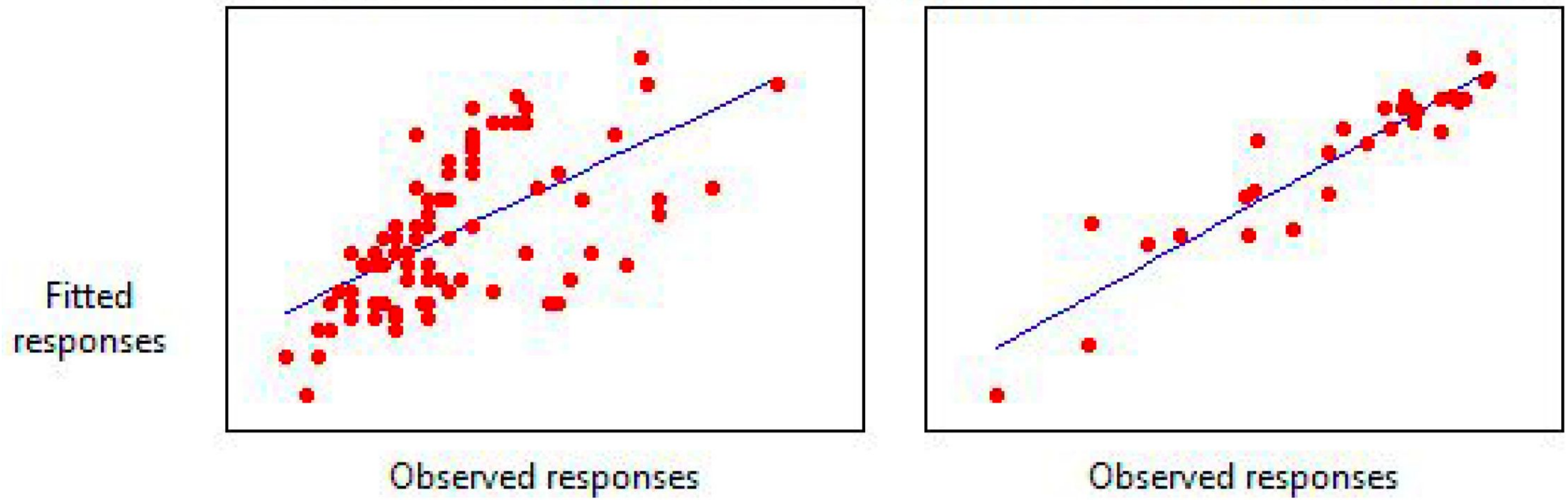
- Between 0 to 100%
  - 0% - none of the variability of the response data around its mean.
  - 100% - all the variability of the response data around its mean.



$$R^2(y, \hat{y}) = 1 - \frac{\sum_{i=0}^{n_{\text{samples}}-1} (y_i - \hat{y}_i)^2}{\sum_{i=0}^{n_{\text{samples}}-1} (y_i - \bar{y})^2}$$

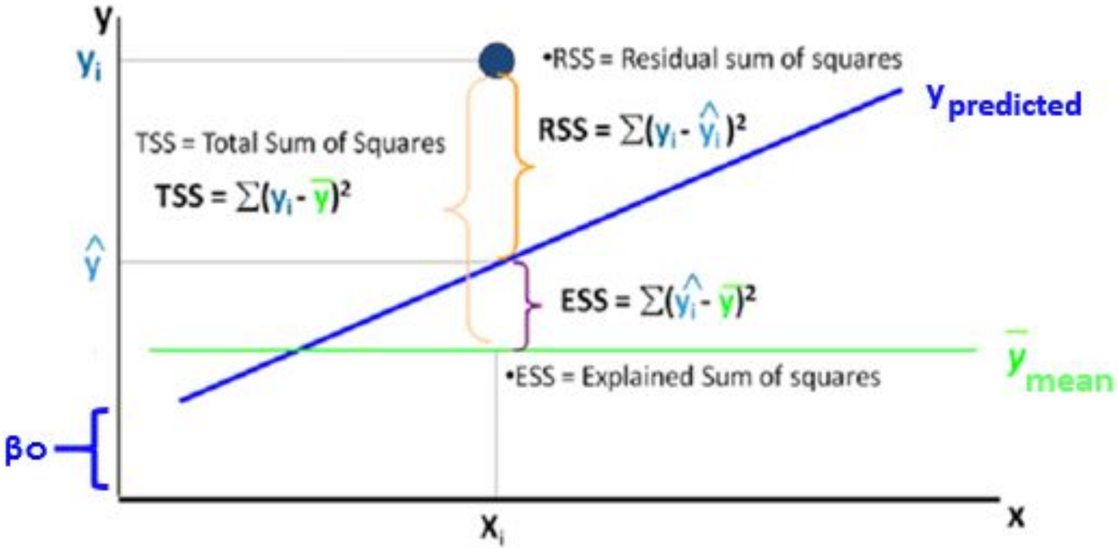
where $\bar{y} = \frac{1}{n_{\text{samples}}} \sum_{i=0}^{n_{\text{samples}}-1} y_i$

Which of the 2 have a higher R^2?



Understanding R^2

Anatomy of Regression Errors


y
y_i
•RSS = Residual sum of squares
$y_{predicted}$
$RSS = \sum(y_i - \hat{y}_i)^2$
TSS = Total Sum of Squares
$TSS = \sum(y_i - \bar{y})^2$
$\hat{y}$
$ESS = \sum(\hat{y}_i - \bar{y})^2$
$\bar{y}_{mean}$
•ESS = Explained Sum of squares
βo
X_i
x

$$R^2 = \frac{ESS}{TSS}$$

$$R^2 = 1 - \frac{RSS}{TSS}$$

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

Metrics: Mean Absolute Error (MAE)

Mean Absolute Error (MAE) is a quantity used to measure how close forecasts or predictions are to the eventual outcomes. The mean absolute error is given by:

$$\mathrm{MAE} = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i| = \frac{1}{n}\sum_{i=1}^{n}|e_i|$$

Where:

- AE = $|e_i| = |y_i - \hat{y}_i|$
- Actual = y_i
- Predicted = $\hat{y}_i$

Metrics: Weighted Mean Absolute Error (WMAE)

A weighting factor would indicate the subjective importance we wish to place on each prediction, relating the error to any feature that might be relevant from both, the user or the seller point of view.

The Weighted Mean Absolute Error can be computed as:

$$\mathrm{WMAE} = \frac{1}{n}\sum_{i=1}^{n} w_i \left| y_i - \hat{y}_i \right|$$

Where:
Actual = y_i
Predicted = $\hat{y}_i$
Weight = w_i

Metrics: Root Mean Squared Error (RMSE)

- Very commonly used
- An excellent general purpose error metric for numerical predictions
- The square root is introduced to make scale of the errors to be the same as the scale of targets.
- Compared to MAE: RMSE amplifies and severely punishes large errors

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

- Where:
  - ○ Actual = y_i
  - ○ Predicted = $\hat{y}_i$

ANALYTIKS

Thank you.

info@analytiksinc.com

www.analytiksinc.com

DATA SCIENCE
SIMPLIFIED

Unit 1206, The Trade and Financial Tower, Bonifacio Global City, Metro Manila, Philippines