



Machine Learning with Python Linear Regression

What Is Regression

It investigates the relationship between a dependent (target) and independent variable (s) (predictor).

This technique is used for forecasting, time series modelling and finding the [causal effect relationship between the variables](#).

For example, relationship between rash driving and number of road accidents by a driver is best studied through regression.

What Is the use of Regression Analysis

- It indicates the significant relationships between dependent variable and independent variable.
- It indicates the strength of impact of multiple independent variables on a dependent variable.

Linear Regression

- **Linear regression:** Linear regression involves using data to calculate a line that best fits that data, and then using that line to predict scores on one variable from another. **Prediction** is simply the process of estimating scores of the outcome (or dependent) variable based on the scores of the predictor (or independent) variable. To generate the regression line, we look for a **line of best fit**.
- A line which can explain the relationship between independent and dependent variable(s), better is said to be best fit line. The difference between the observed value and actual value gives the **error**.

Linear Regression gives an equation of the following form:

$$Y = m_0 + m_1x_1 + m_2x_2 + m_3x_3 + \dots m_nx_n$$

where Y is the dependent variable and X's are the independent variables.

The right-hand side of this equation is also known as Hypothesis Function - H(x)

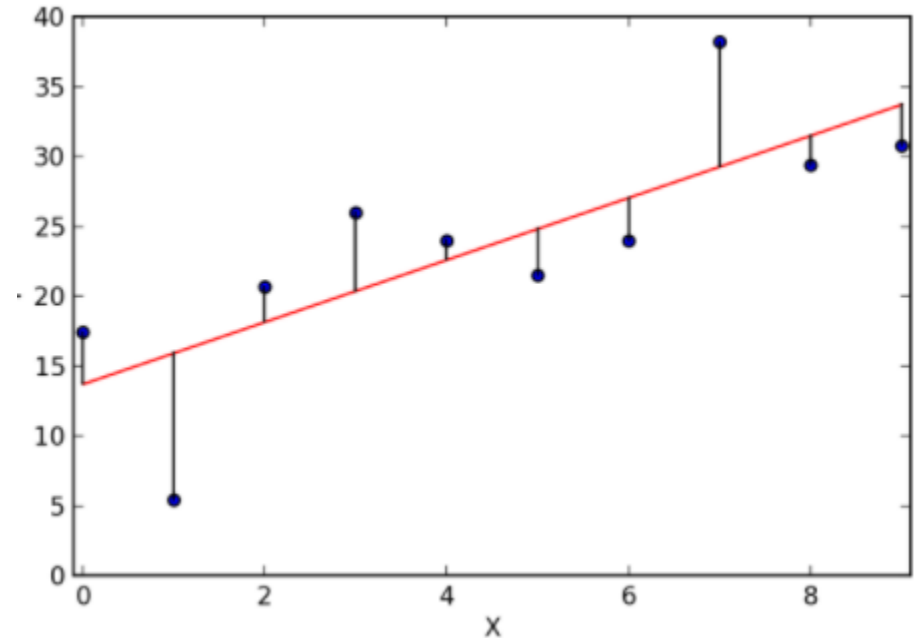
Line of Best Fit

The purpose of line of best fit is that the predicted values should be as close as possible to the actual or observed values. This means the main objective in determining the line of best fit is to “minimize” the difference predicted values and observed values. These differences are called “errors” or “residuals”.

3 ways to calculate the “error”

- Sum of all errors: $(\sum(Y - h(X)))$ (This may result in the cancellation of positive and negative errors. This will not be a correct metric to use)
 - Sum of absolute value of all errors: $(\sum|Y-h(X)|)$
 - Sum of square of all errors $(\sum (Y-h(X))^2)$
- The line of best fit for 1 feature can be represented as :
- $Y = bx + c$** Where Y is the score or outcome variable we are trying to predict
B = regression coefficient or slope
C = Y intercept or the regression constant

This is Linear regression with 1 variable.



Sum of Squared Errors

- Squaring the difference between actual value and predicted value “penalizes” more for each error. Hence minimizing the sum of squared errors improves the quality of regression line.
- This method of fitting the data line so that there is minimal difference between the observations and the line is called the **method of least squares**.
- **Baseline model** refers to the line which predicts each value as the average of the data points.
- **SSE or Sum of Squared Errors** is the total of all squares of the errors. It is a measure of the quality of regression line. SSE is sensitive to the number of input data points. How much the target value varies around the regression line (predicted value).

$$\text{Error} = \sum_{i=1}^n (\text{Actual_output} - \text{predicted_output})^2$$

- **SST is Total Sum of Squares**: It is the SSE for baseline model. This tells how much the data point move around the mean.

$$\text{Error} = \sum_{i=1}^n (\text{Actual_output} - \text{average_of_actual_output})^2$$

Regression Metrics

Mean Absolute Error : One way to measure error is by using absolute error to find the predicted distance from the true value. The mean absolute error takes the total absolute error of each example and averages the error based on the number of data points. By adding up all the absolute values of errors of a model we can avoid canceling out errors from being too high or below the true values and get an overall error metric to evaluate the model on.

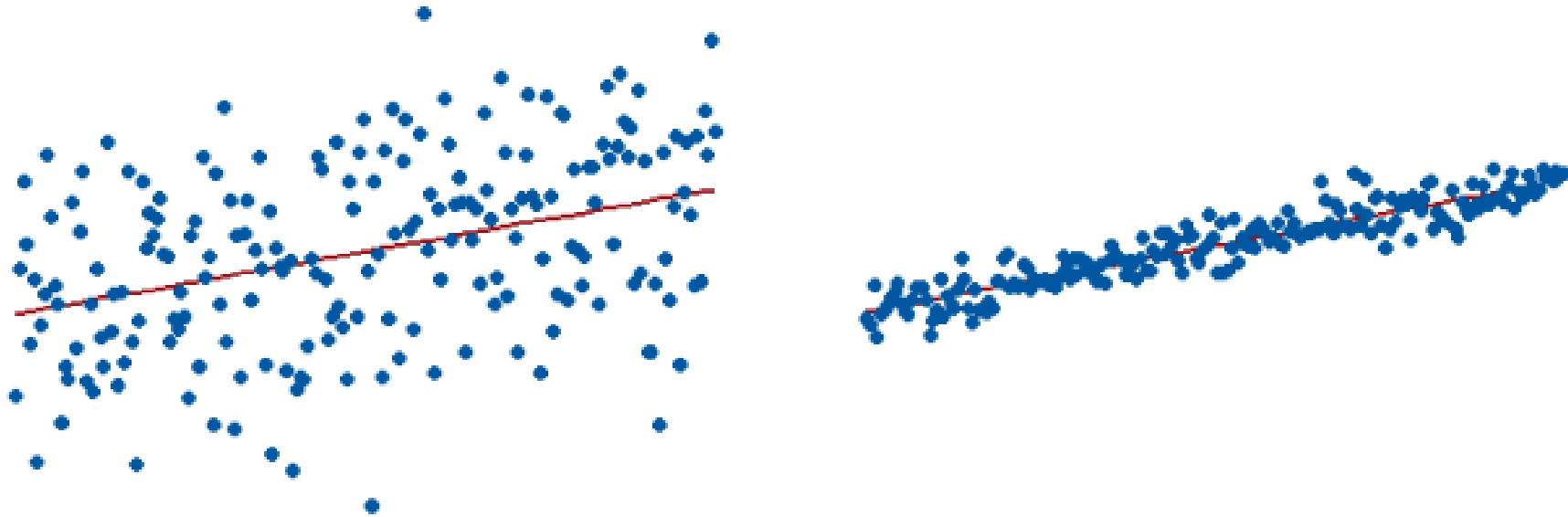
Mean Squared Error : Mean squared is the most common metric to measure model performance. In contrast with absolute error, the residual error (the difference between predicted and the true value) is squared.

Some benefits of squaring the residual error is that error terms are positive, it emphasizes larger errors over smaller errors, and is differentiable. Being differentiable allows us to use calculus to find minimum or maximum values, often resulting in being more computationally efficient.

R-Squared: Its called coefficient of determination. The values for R^2 range from 0 to 1, and it determines how much of the total variation in Y is explained by the variation in X. A model with an R^2 of 0 is no better than a model that always predicts the *mean* of the target variable, whereas a model with an R^2 of 1 perfectly predicts the target variable. Any value between 0 and 1 indicates what percentage of the target variable, using this model, can be explained by the **features**. *A model can be given a negative R^2 as well, which indicates that the model is **arbitrarily worse** than one that always predicts the mean of the target variable.*

$$R - Square = 1 - \frac{\sum(Y_{actual} - Y_{predicted})^2}{\sum(Y_{actual} - Y_{mean})^2}$$

Visual Representation of R-Squared



The R-squared for the regression model on the left is 15%, and for the model on the right it is 85%. When a regression model accounts for more of the variance, the data points are closer to the regression line. In practice, you'll never see a regression model with an R^2 of 100%. In that case, the fitted values equal the data values and, consequently, all of the observations fall exactly on the regression line.

Cost Function

- The error of regression model is expressed as a cost function :

$$J(\theta_0, \theta_1) = \frac{1}{2m} \sum_{i=1}^m (h_{\theta}(x^{(i)}) - y^{(i)})^2$$

Its is similar to sum of squared errors. $1/m$ is means, we are calculating the average. The factor $1/2$ is used to simplify mathematics. This function is minimized to reduce errors in prediction.

Minimizing this function, means we get the values of θ_0 and θ_1 which find on average the minimal deviation of x from y when we use those parameters in our hypothesis function.

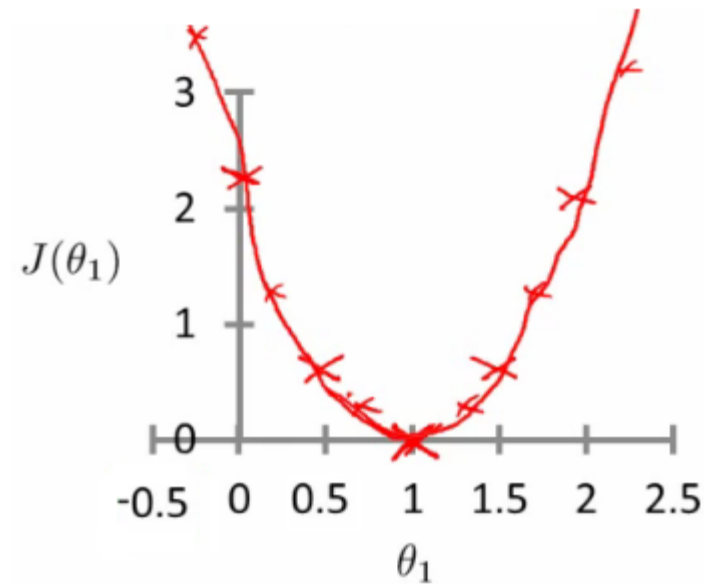
Inside Cost Function

Cost function :

$$J(\theta_0, \theta_1) = \frac{1}{2m} \sum_{i=1}^m (h_{\theta}(x^{(i)}) - y^{(i)})^2$$

Lets assume, θ_0 is 0. (Our hypothesis passes through origin)

So, now we need that value of θ_1 for which Cost function is minimum. To find that out, plot $J(\theta_1)$ vs θ_1



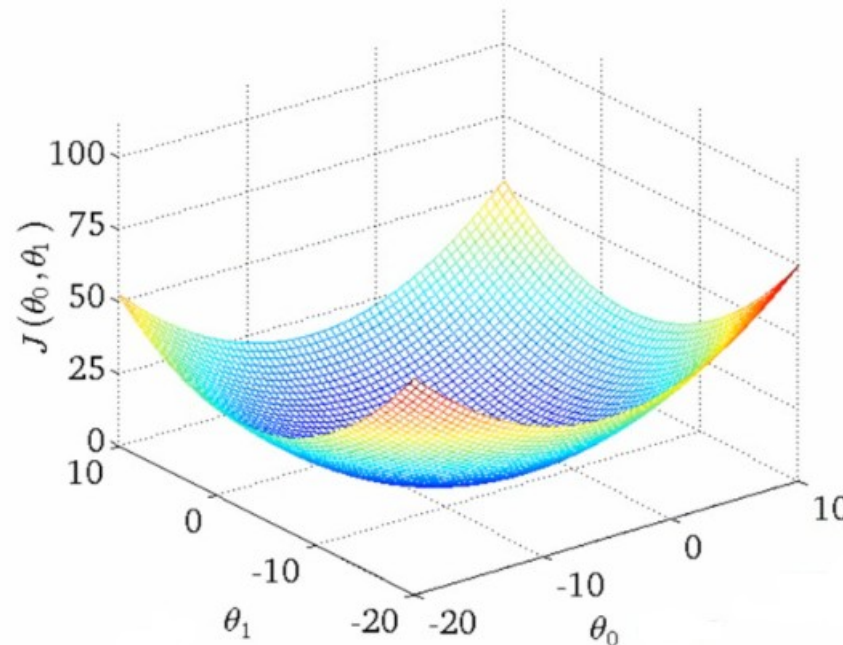
Inside Cost Function

Cost function :

$$J(\theta_0, \theta_1) = \frac{1}{2m} \sum_{i=1}^m (h_{\theta}(x^{(i)}) - y^{(i)})^2$$

With both θ_0 and θ_1 , The plot becomes more complex

So, now we need that value of θ_1 for which Cost function is minimum. To find that out, plot $J(\theta_1, \theta_0)$ vs θ_1 and θ_0

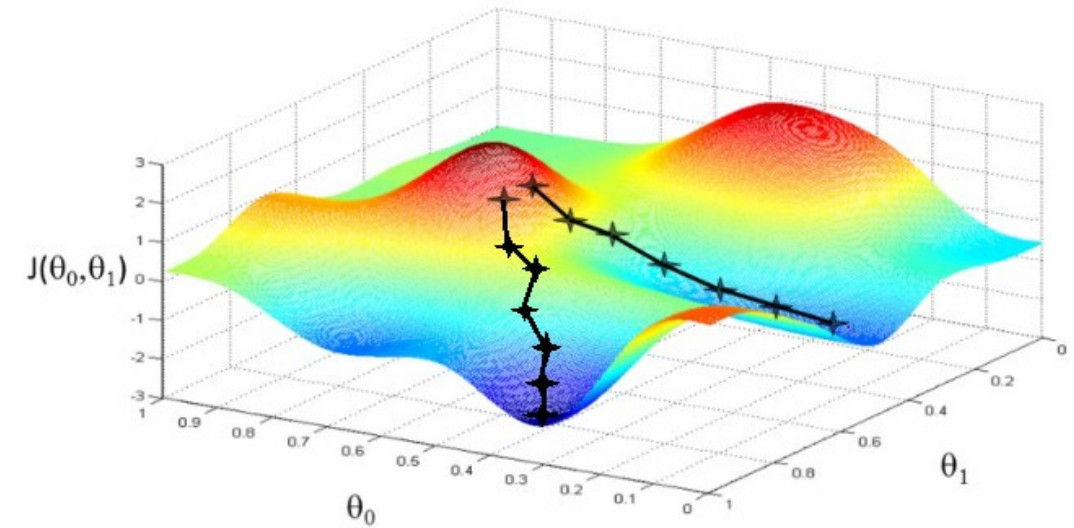


Gradient Descent

The process of minimizing the cost function can be achieved by Gradient Descent algorithm:

The steps are:

1. Start with initial guess of coefficients
2. Keep changing the coefficients a little bit to try and reduce Cost Function $J(\theta_0, \theta_1)$
3. Each time, the parameters are changed, the gradient is chosen which reduces $J(\theta_0, \theta_1)$ the most.
4. Repeat
5. Keep doing till no improvement is made.



Polynomial Regression

Instead of finding a best fit “line” on the given data points, we can also try to find the best fit “curve”.

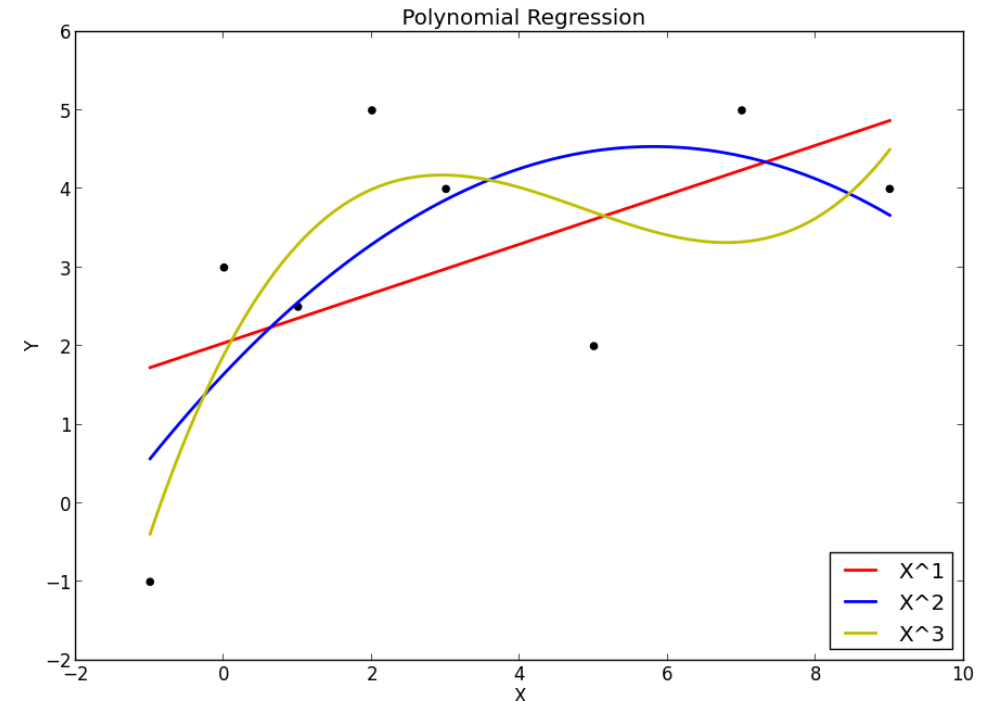
This is the form of Polynomial regression. The equation, in case of second-order polynomial will be:

$$Y = \theta_0 + \theta_1 x + \theta_2 x^2 \text{ (Quadratic Regression)}$$

Third-order polynomial will be:

$$Y = \theta_0 + \theta_1 x + \theta_2 x^2 + \theta_3 x^3 \text{ (Cubic Regression)}$$

When we use higher order powers in our regression model, we say that we are increasing the “**complexity**” of the model. The more the complexity of the model, the better it will “fit” on the given data.

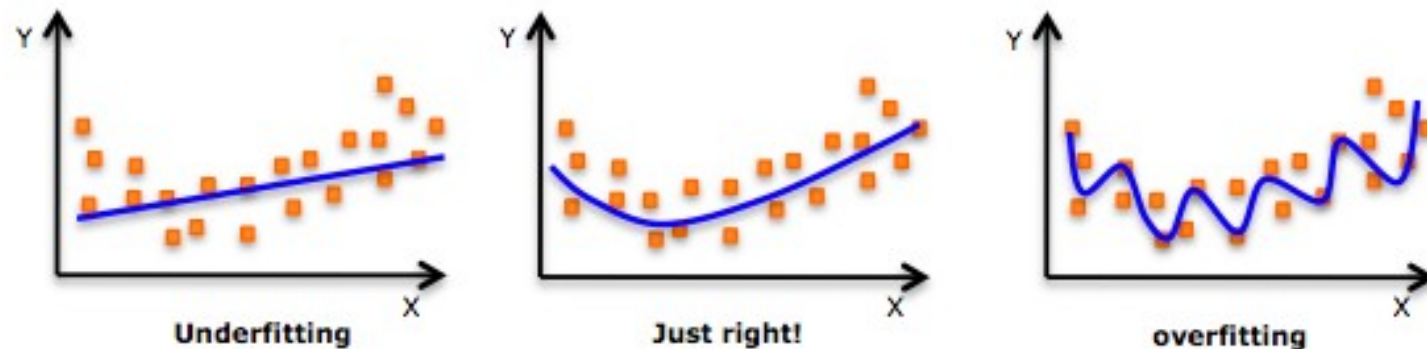


Overfitting and Underfitting

So, should we always choose a “complex” model with higher order polynomials to fit the data set?

NO, it may be possible that such a model gives very wrong predictions on Test data. Though it fits well on training data but fails to estimate the real relationship among variables beyond the training set. This is known as “Over-fitting”

Similarly, we can have **underfitting**, it occurs when our model neither fits the training data nor generalizes on the new data.



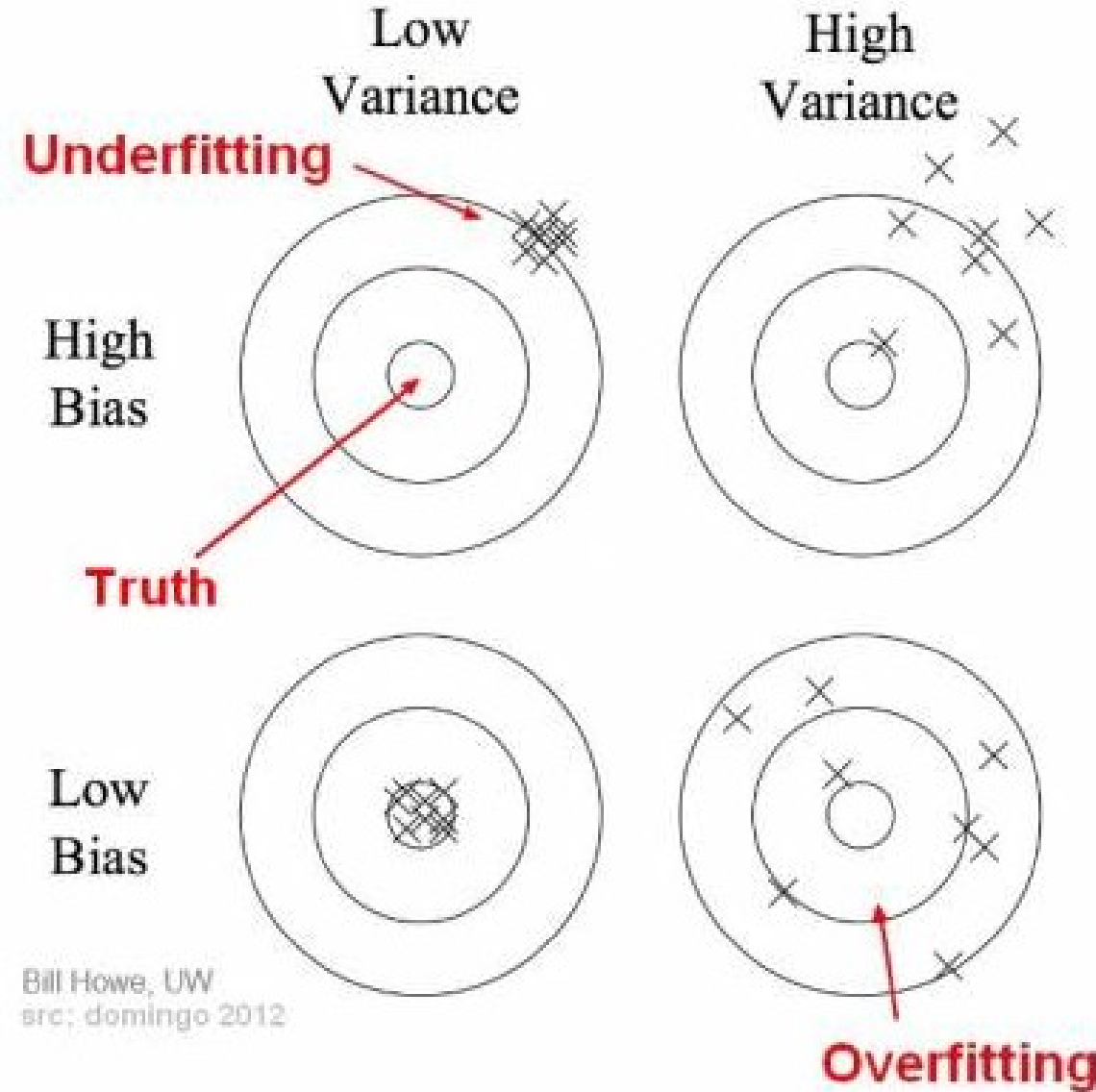
Bias and Variance

Bias: Bias occurs when a model has enough data but is **not complex enough to capture the underlying relationships(or patterns)**. As a result, the model consistently and systematically misrepresents the data, leading to low accuracy in prediction. This is known as *underfitting*. Simply put, bias occurs when we have an inadequate model.

Variance: Variance refers to the amount by which your estimate of $f(X)$ would change if we estimated it using a different training data set. Since the training data is used to fit the statistical learning method, different training data sets will result in a different estimation. But ideally the estimate for $f(X)$ should not vary too much between training sets. However, if a method has high variance then small changes in the training data can result in large changes in $f(X)$. (Pays too much attention to data; high error on test set)

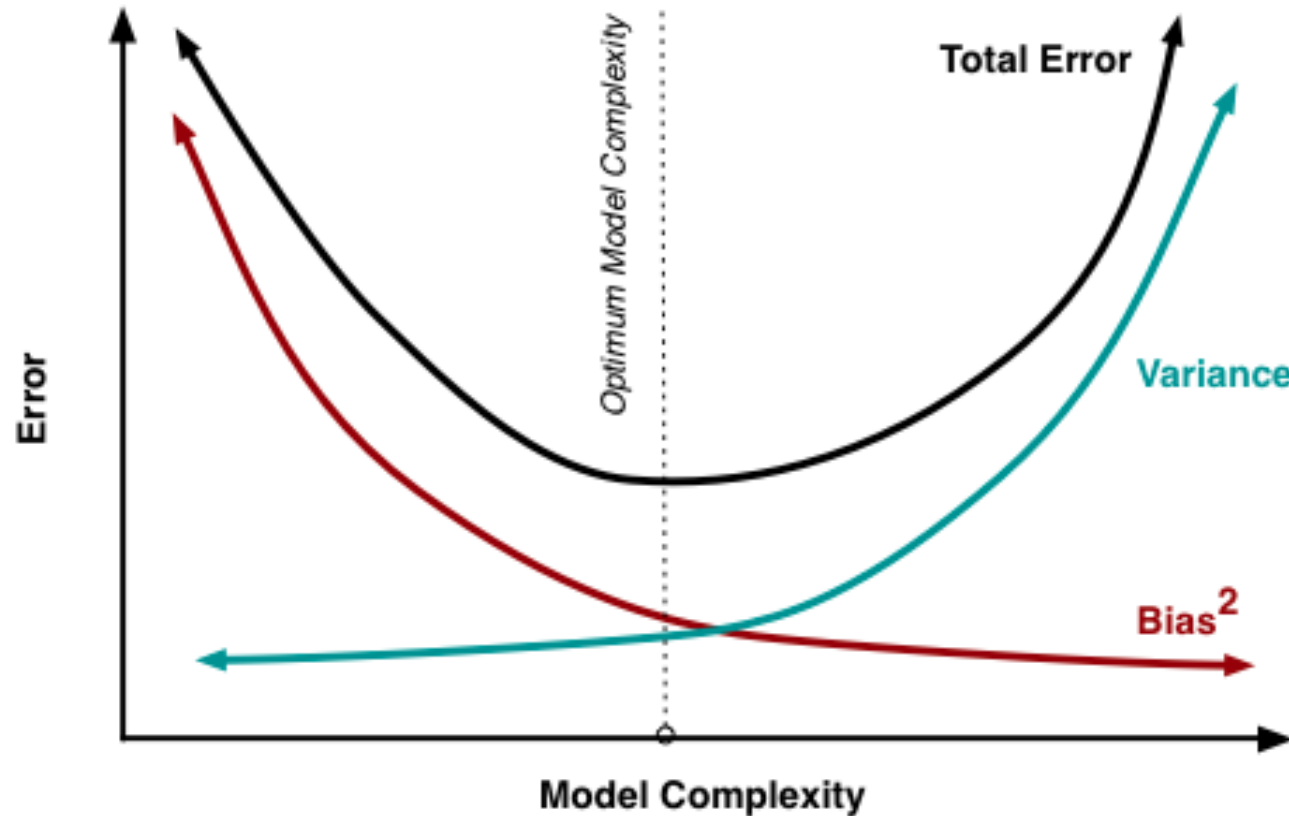
- Some variance is normal, but too much variance indicates that the model is unable to generalize its predictions to the larger population. High sensitivity to the training set is also known as *overfitting*, and generally occurs when either the model is too complex or when we do not have enough data to support it.
- We can typically reduce the variability of a model's predictions and increase precision by training on more data. If more data is unavailable, we can also control variance by limiting our model's complexity.

Bias and variance using bulls-eye diagram



Bill Howe, UW
src: domingo 2012

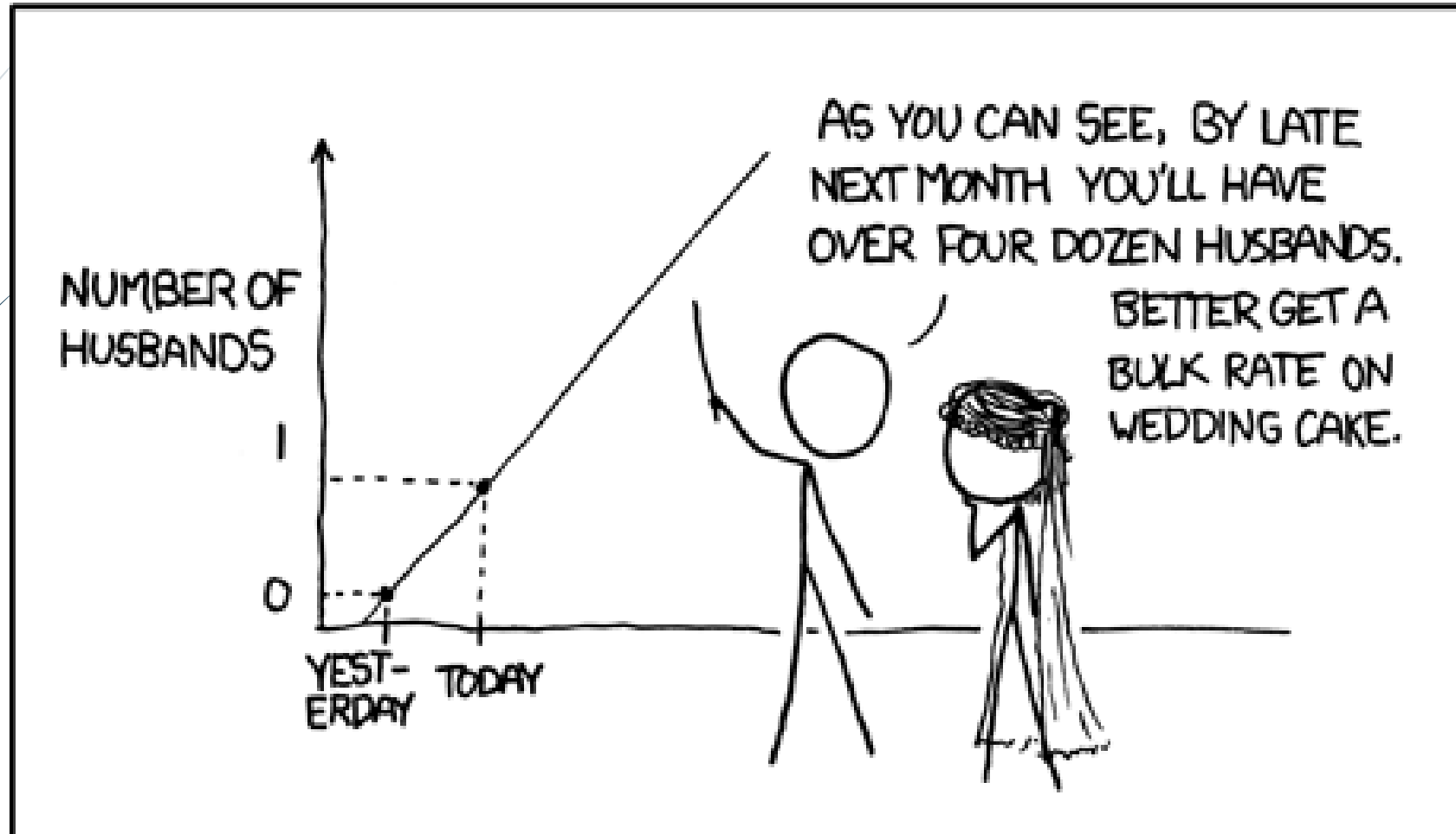
Bias and Variance : Contribution to error



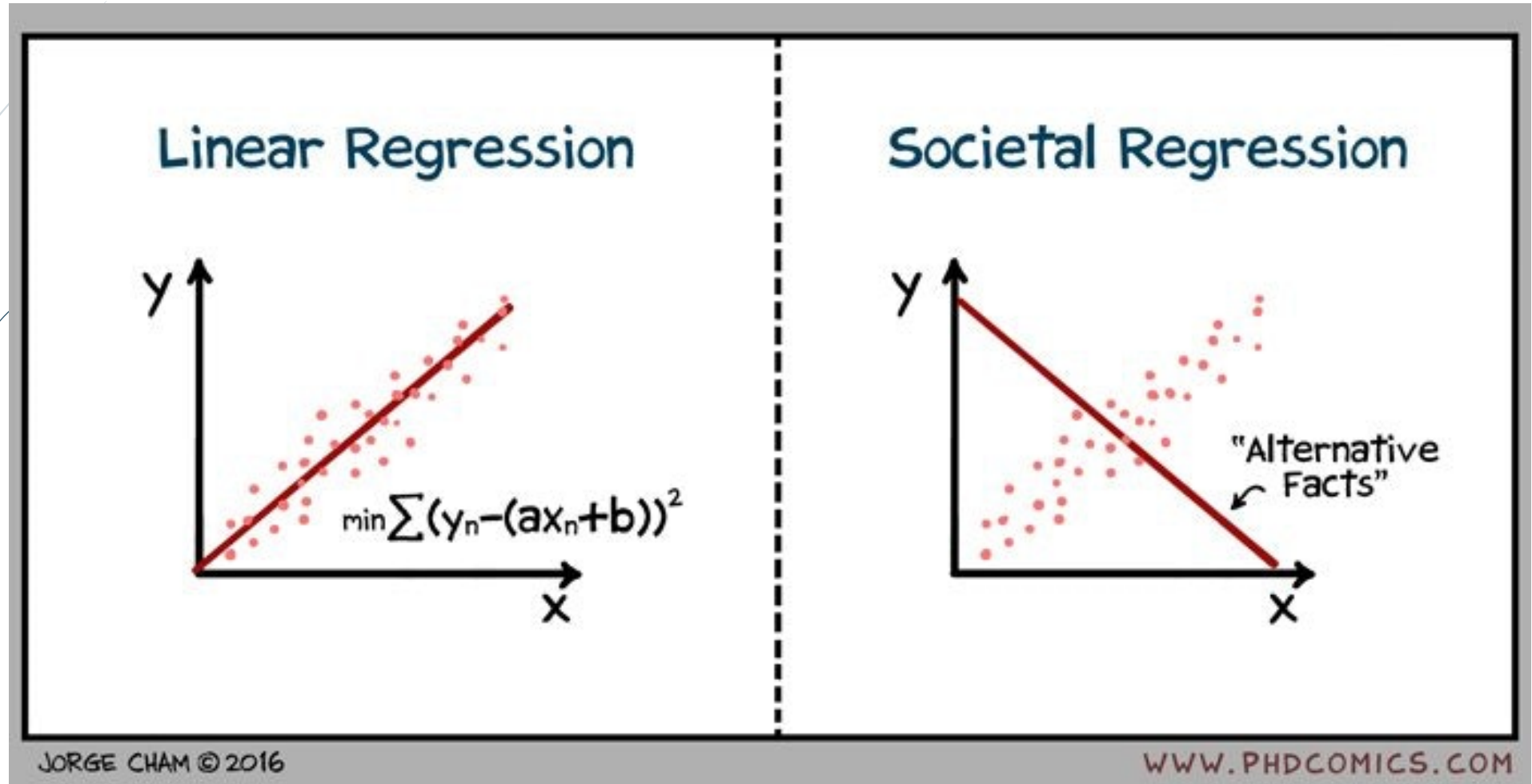
As more and more parameters are added to a model, the complexity of the model rises and variance becomes our primary concern while bias steadily falls. For example, as more polynomial terms are added to a linear regression, the greater the resulting model's complexity will be.

Overfitting!!

MY HOBBY: EXTRAPOLATING



Underfitting!!



Adjusted R-Squared

- R-square will increase or remain constant, if we add new predictors to our model. So there is no way to judge that by increasing complexity of the model, are we making it more accurate?
- We “adjust” R-Square formula to include no of predictors in the model. The adjusted R-Square only increases if the new term improves the model accuracy.

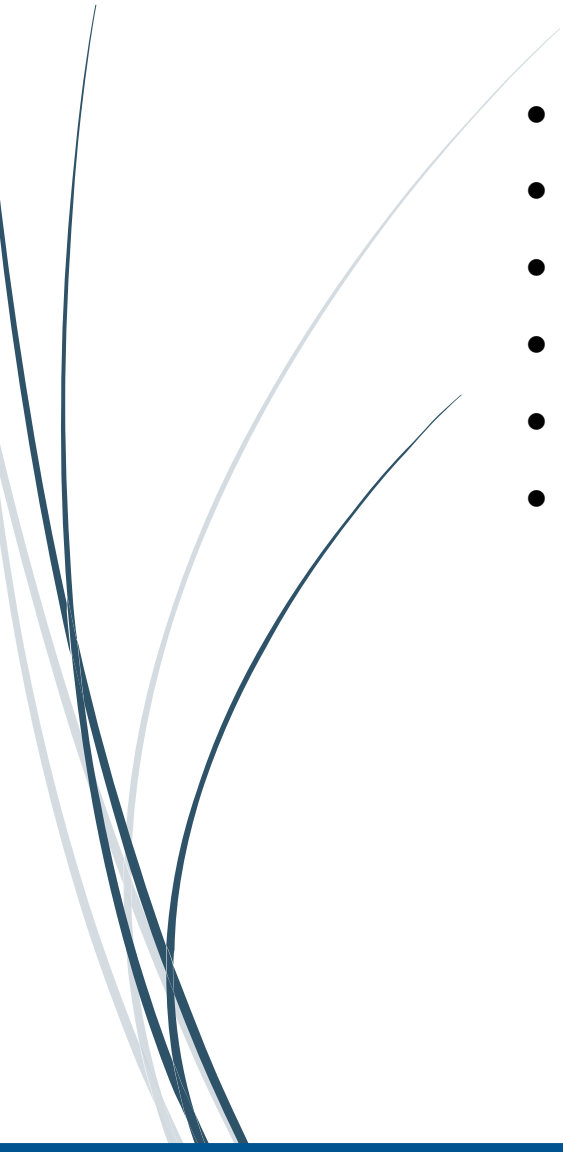
$$R^2 \text{ adjusted} = 1 - \frac{(1 - R^2)(N - 1)}{N - p - 1}$$

R^2 = Sample R square

p = Number of predictors

N = total sample size

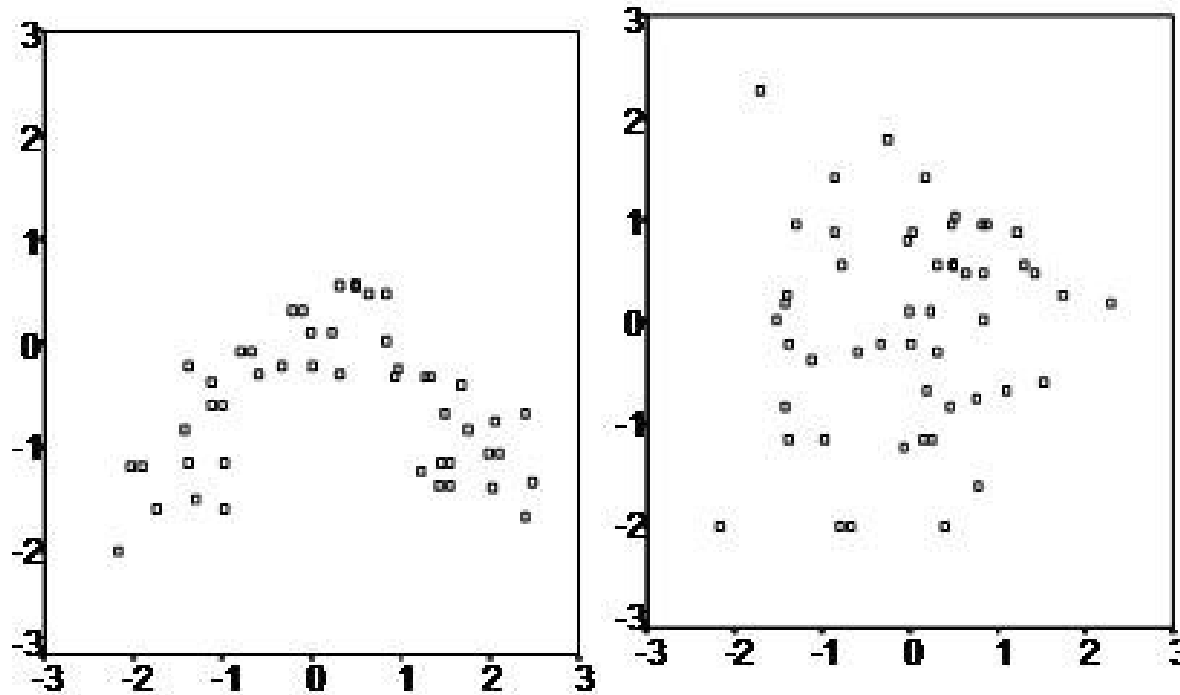
Assumption of Linear Regression about data

- 
- Linearity
 - Multivariate Normal
 - No or Little Multicollinearity
 - No Auto-correlation
 - Homoscedasticity or Equal Variance
 - Outliers

LINEARITY

The expected value of dependent variable is a **straight-line** function of each independent variable, holding the others fixed.

The linearity assumption can best be tested with scatter plots, the following two examples depict two cases. where no and little linearity is present.



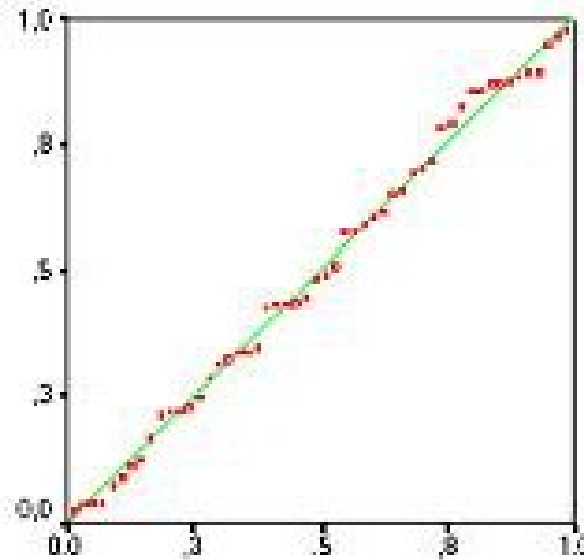
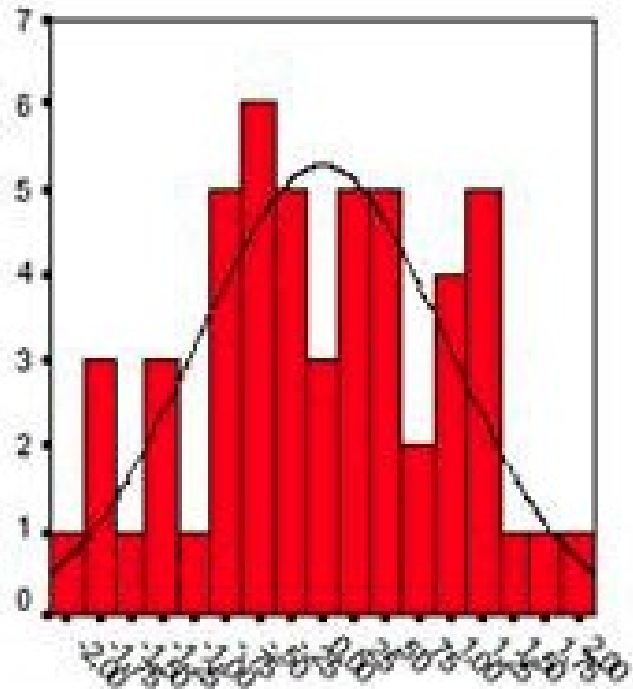
HOW TO IDENTIFY:

- Scatter diagram of target and independent variable.
- Also, you can include polynomial terms (X , X^2 , X^3) in your model to capture the non-linear effect.
- **Harvey-Collier multiplier test**, for *linearity*

MULTIVARIATE NORMAL

Data should be distributed normally.

Violations of normality create problems for determining whether model coefficients are significantly different from zero and for calculating confidence intervals for forecasts. Sometimes the error distribution is "skewed" by the presence of a few large outliers. Since parameter estimation is based on the minimization of *squared* error, a few extreme observations can exert a disproportionate influence on parameter estimates. Calculation of confidence intervals and various significance tests for coefficients are all based on the assumptions of normally distributed errors. If the error distribution is significantly non-normal, confidence intervals may be too wide or too narrow.



HOW TO IDENTIFY:

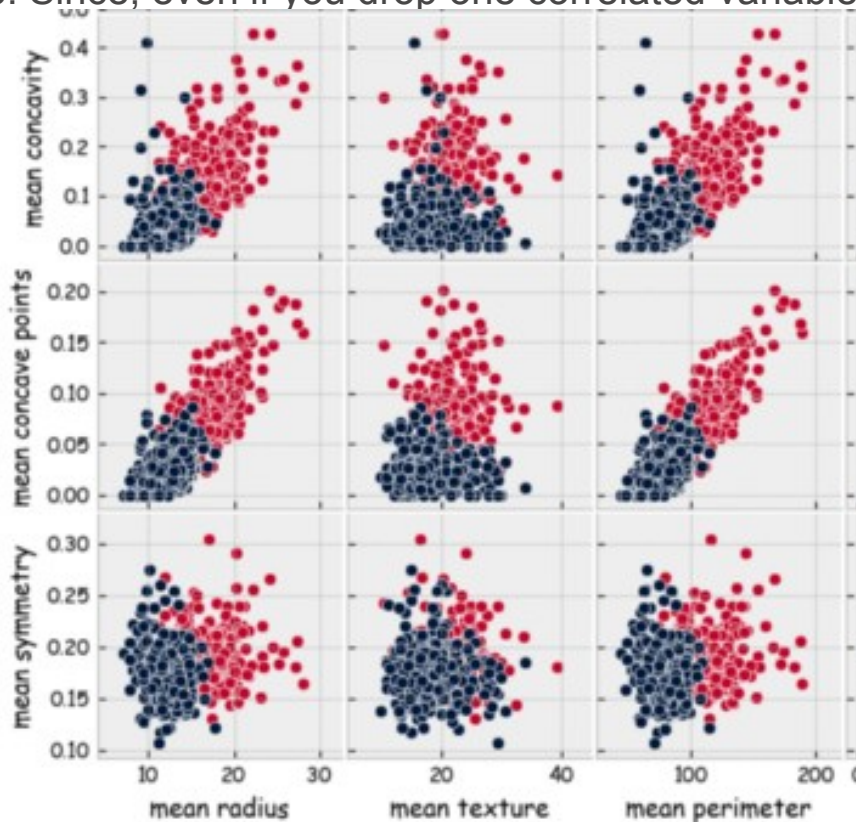
- Histogram of target and independent variable.
- Q-Q plot
- Jarque Bera test

NO OR LITTLE MULTICOLLINEARITY

Multicollinearity occurs when the independent variables are too highly correlated with each other.

In a model with correlated variables, it becomes a tough task to figure out the true relationship of a predictors with response variable. In other words, it becomes difficult to find out which variable is actually contributing to predict the response variable.

when predictors are correlated, the estimated regression coefficient of a correlated variable depends on which other predictors are available in the model. If this happens, you'll end up with an incorrect conclusion that a variable strongly / weakly affects target variable. Since, even if you drop one correlated variable from the model, its estimated regression coefficients would change. That



HOW TO IDENTIFY:

Scatter plot to visualize correlation effect among variables.

Variance Inflation Factor (VIF) – the variance inflation factor of the linear regression is defined as $VIF = 1/T$. With $VIF > 10$ there is an indication that multicollinearity may be present; with $VIF > 100$ there is certainly multicollinearity among the variables.

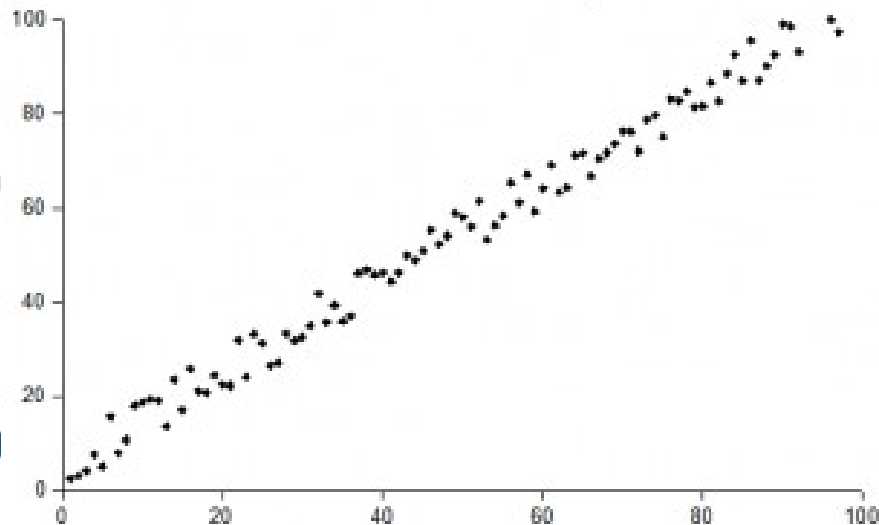
HOMOSCEDASTICITY

Homoscedasticity describes a situation in which the error term (that is, the “noise” or random disturbance in the relationship between the independent variables and the dependent variable) is the same across all values of the independent variables.

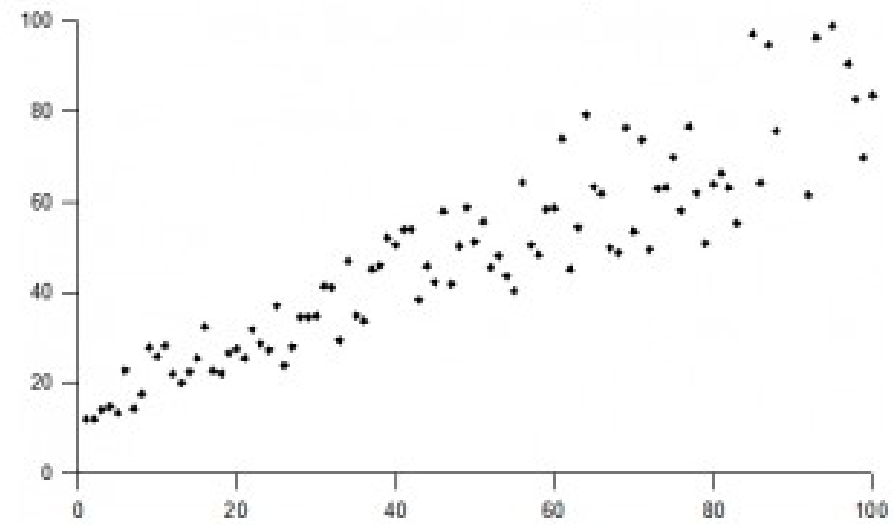
Heteroscedasticity (the violation of homoscedasticity) is present when the size of the error term differs across values of an independent variable.

Heteroscedasticity may also have the effect of giving too much weight to a small subset of the data (namely the subset where the error variance was largest) when estimating coefficients.

Homoscedasticity



Heteroscedasticity



HOW TO IDENTIFY:

Breush-Pagan test & Goldfeld-Quandt test

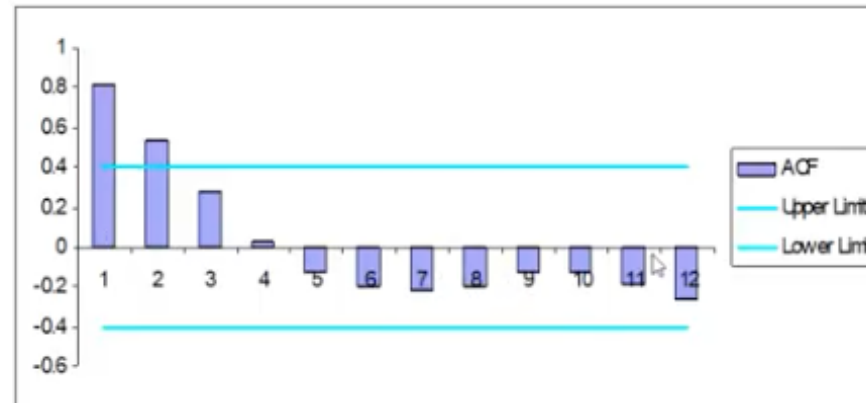
Auto Correlation (Self-Correlation)

When you have a series of numbers, and there is a pattern such that values in the series can be predicted based on preceding values in the series, the series of numbers is said to exhibit autocorrelation. This is also known as serial correlation and serial dependence.

Autocorrelation means that data is correlated with itself, as opposed to being correlated with some other data.

Correlogram

The plot of the autocorrelation function (ACF) versus time lag is called Correlogram. The horizontal scale is the time lag. The vertical axis is the autocorrelation coefficient.



HOW TO IDENTIFY:

- Correlogram also called autocorrelation plot or ACF plot
- Durbin-watson test

Ridge Regression

Ridge Regression is a technique used when the data suffers from multicollinearity (independent variables are highly correlated).

In multicollinearity, even though the least squares estimates (OLS) are unbiased, their variances are large which deviates the observed value far from the true value. By adding a degree of bias to the regression estimates, ridge regression reduces the standard errors.

When Lambda = 0, the penalty term has no effect and the estimates produced by ridge regression will be equal to least squares.

However, as lambda tends to infinity, the impact of the shrinkage penalty grows, and the ridge regression coefficient estimates will approach zero.

As can be seen, selecting a good value of lambda is critical. Cross validation comes in handy for this purpose. The coefficient estimates produced by this method are also known as the L2 norm.

$y = a + b \cdot x + e$ (error term), [error term is the value needed to correct for a prediction error between the observed and predicted value]

Ridge regression solves the multicollinearity problem through shrinkage parameter λ (lambda). Look at the equation below.

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

Lasso Regression

Similar to Ridge Regression, Lasso (Least Absolute Shrinkage and Selection Operator) also penalizes the absolute size of the regression coefficients. In addition, it is capable of reducing the variability and improving the accuracy of linear regression models. Look at the equation below:

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

Lasso regression differs from ridge regression in a way that it uses absolute values in the penalty function, instead of squares. This leads to penalizing (or equivalently constraining the sum of the absolute values of the estimates) values which causes some of the parameter estimates to turn out exactly zero. Larger the penalty applied, further the estimates get shrunk towards absolute zero. This results to variable selection out of given n variables.

ElasticNet Regression

ElasticNet is hybrid of Lasso and Ridge Regression techniques. It is trained with L1 and L2 prior as regularizer. Elastic-net is useful when there are multiple features which are correlated. Lasso is likely to pick one of these at random, while elastic-net is likely to pick both.

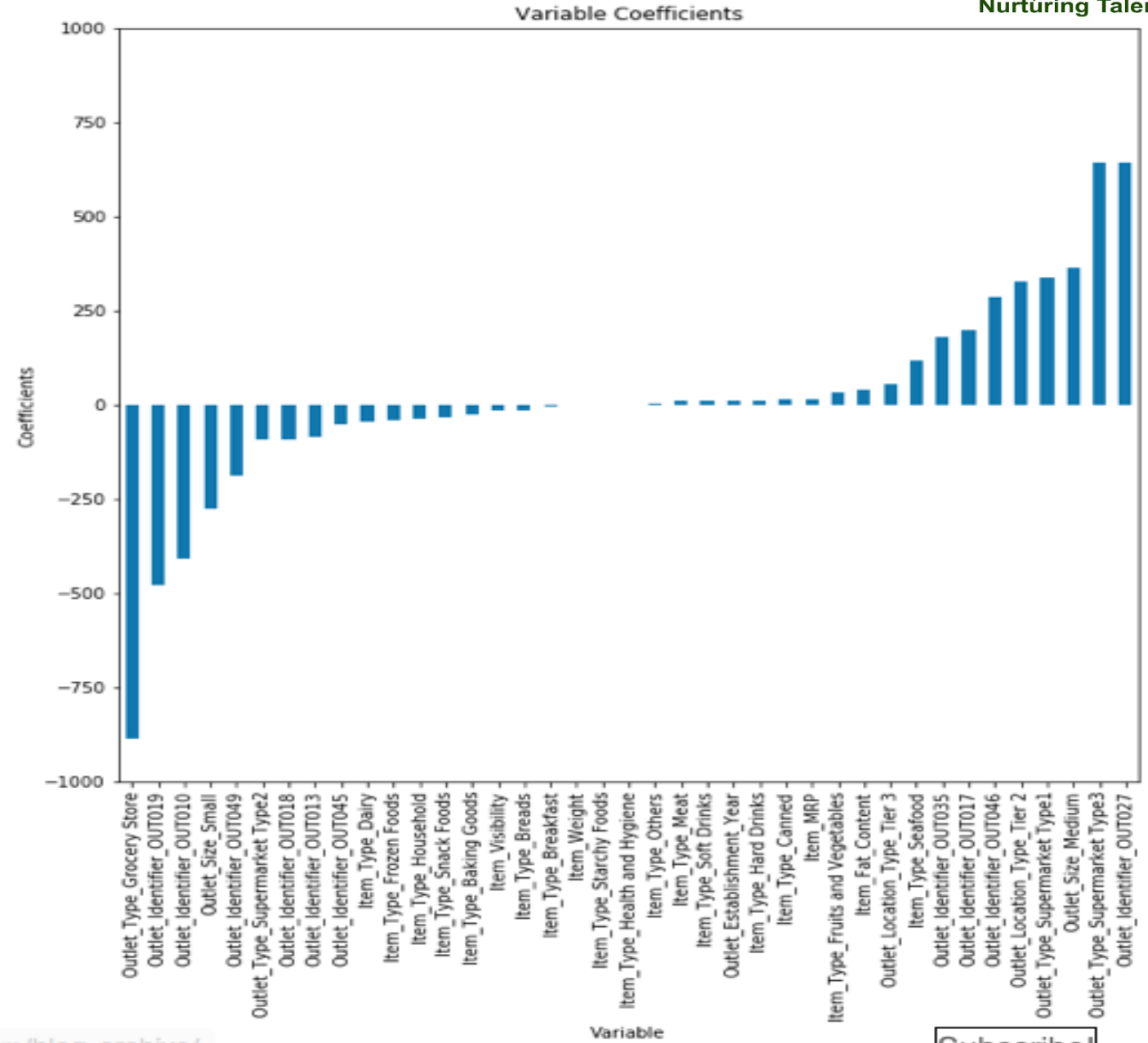
$$\hat{\beta} = \underset{\beta}{\operatorname{argmin}} (\|y - X\beta\|^2 + \lambda_2 \|\beta\|^2 + \lambda_1 \|\beta\|_1).$$

A practical advantage of trading-off between Lasso and Ridge is that, it allows Elastic-Net to inherit some of Ridge's stability under rotation.

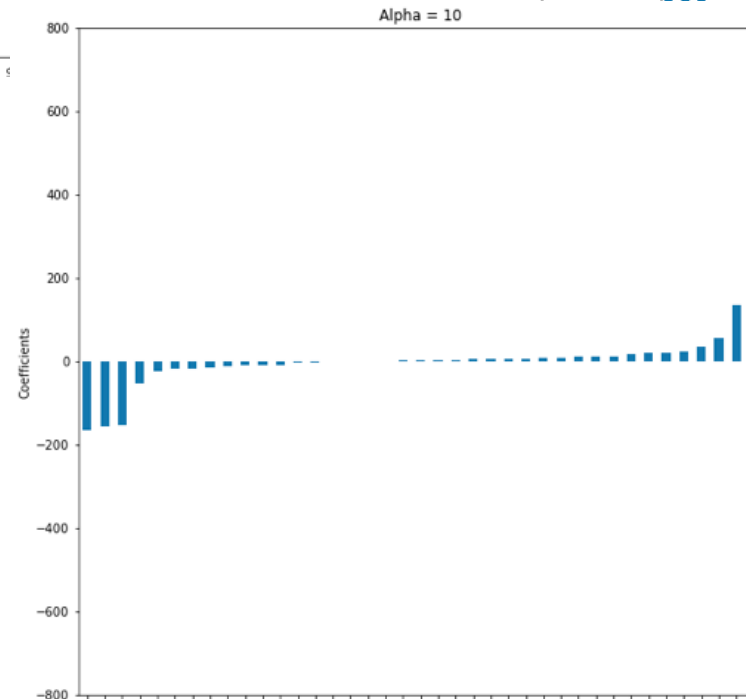
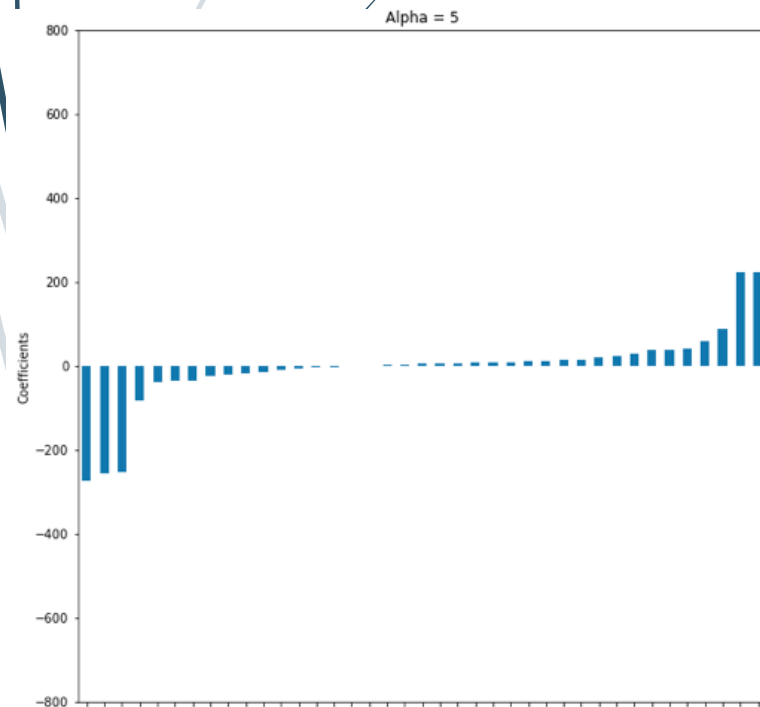
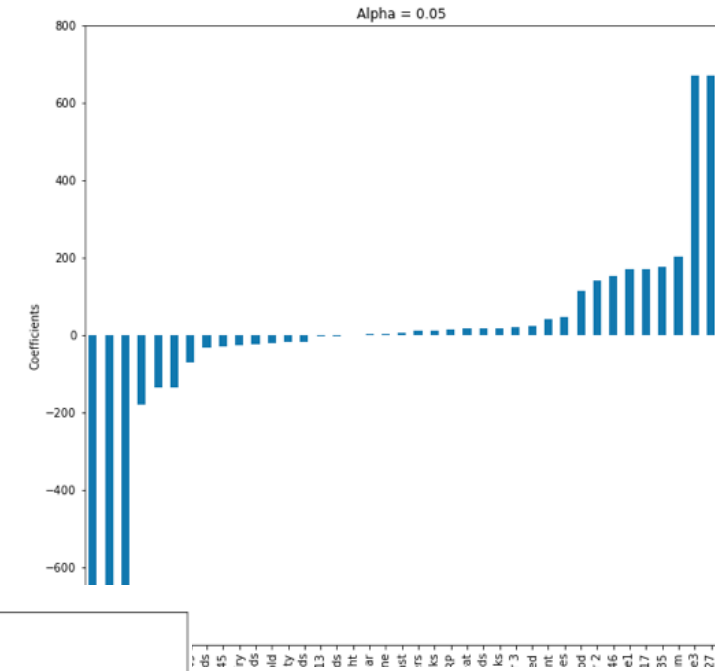
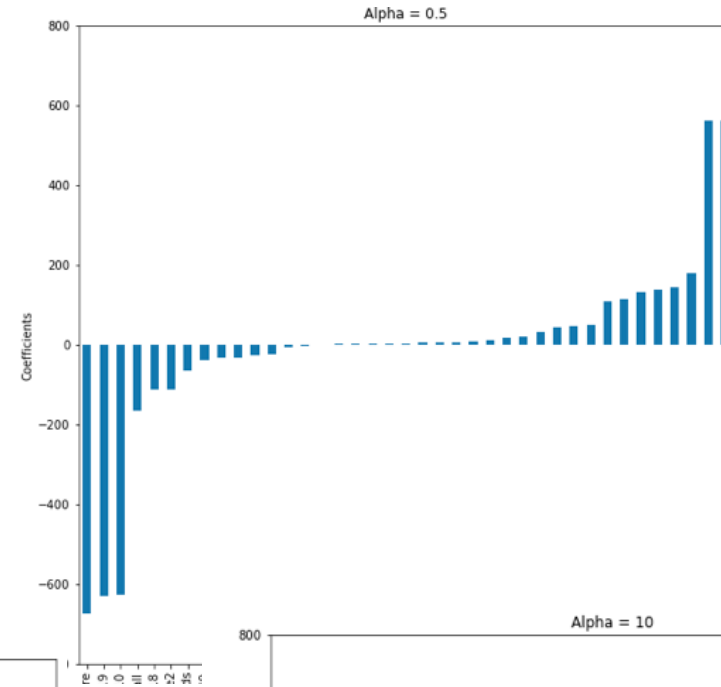
Regression regularization methods(Lasso, Ridge and ElasticNet) works well in case of **high dimensionality and multicollinearity** among the variables in the data set.

Linear Regression

Model is biased by 2 features



Ridge Regression Regularization



But if you calculate R-square for each alpha, we will see that the value of R-square will be maximum at $\alpha=0.05$. So we have to choose it wisely by iterating it through a range of values and using the one which gives us lowest error