Linear Regression
Theory

Regression Model Assumptions

Assumption 1

Linear Regression Model is linear in the parameters though it may not be linear in the variables i.e. the slope coefficients are always raised to power 1. The variables may be raised to any power. The regression model thus, takes the form $y_i = \beta_1 + \beta_2 X_i + u_i$ (β_1 is intercept, β_2 is coefficient, X_i is variable, u_i is disturbance

The conditional expectation of y, E(y | Xi) is a linear function of the parameters i.e. the βs. y is linearly related to X when the rate of change of y with respect to X (i.e. slope or derivative of y with respect to X, dy / dx) is independent of the value of X. For e.g.

1. if $y = 10x$ then $dy/dx = 10$, which is independent of x i.e. for all values of x, dy/dx is constant.
2. If $y = 10x^2$ then $dy/dx = 20x$ i.e. the rate of change of y depends on the current value of x. It is dependent on X. Hence the function is not linear in X

The deviation ui in each Xi prediction can be positive or negative. Technically ui is known as stochastic disturbance or stochastic error term

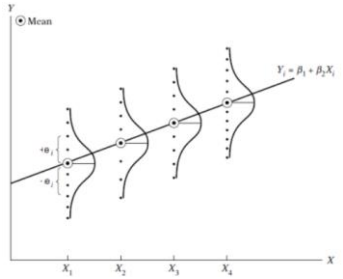
Regression Model Assumptions

Assumption 2

1. X values are independent of the error term. Values taken by the regressor X may be considered fixed in repeated trials / sample. The error in prediction of each trial is independent of the value of X.
2. Error term ui, represents the impact of the variables not considered for the model. Since the assumption that X predictors are independent of one another, it applies to even those variables not considered

Assumption 3

1. The mean value of disturbance ui is zero. Given the value of Xi, the means or expected value of the random disturbance E(ui | Xi) = 0 i.e. E(ui) = 0
2. Population of y corresponding to a given Xi is distributed around its mean value, implies no specification bias / error in the model indicating that the model is correctly specified.



3. Specification error results from leaving out important variables or choosing wrong functional form to express relationship between y and X

Regression Model Assumptions

Assumption 4

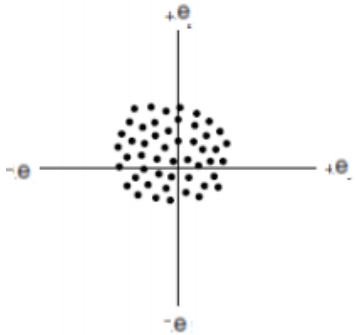
1. Homoscedasticity or Constant Variance of ui, the variance of the error / disturbance is the same regardless of the value of X
2. Var(ui) = E[ui - E(ui | Xi)]^2 = E(ui^2 | Xi) because of assumption 3 (E(ui) = 0).
3. = E(ui^2) for a given Xi = constant variance (representation for homoscedasticity)

Image source: https://www.rhayden.us/regression-models/the-nature-of-heteroscedasticity.html

4. The likelihood that the y observations coming from the population with X = Xi would be closer to the population regression function than those coming from the populations corresponding to X = X2, X = X3 and so on. The reliability of predicted Y will fall
5. By invoking Assumption 4, we stress equal importance to all y values corresponding to different values of X

Regression Model Assumptions

Assumption 5



1. No autocorrelation between disturbances ui. Given any two X values, Xi and Xj (I <> j), the correlation between any to ui and uj is zero i.e. no serial or auto correlation
2. This assumption is justified when time is not an attribute i.e. the trials / records are not generated in any time-series fashion

Assumption 6

1. The number of observations n must be greater than the number of parameters to be estimated. In data science parlance, the depth should be much greater than breadth i.e. number of records much larger than the number of columns to avoid curse of dimensionality situation.

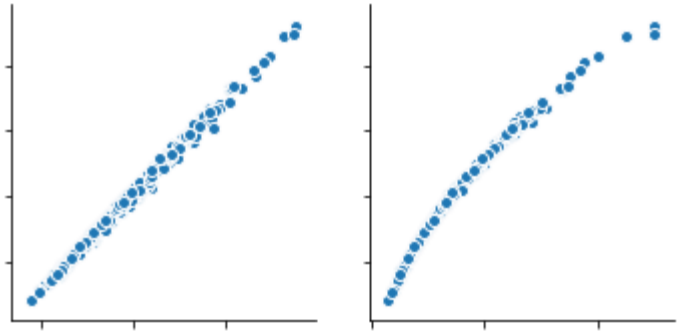
Assumption 7

1. The X should have variance. The values should not be constant. In Data Science parlance, X should have variance. Further the outliers should not exist

Regression Model Assumptions

Assumption 8

1. There no perfect collinearity between the predictor variables X
2. In case of perfect collinearity, the scatter plot will be line
3. Most often we come across less than perfect collinearity



Assumption 9

1. The model is correctly specified i.e. neither overfit or underfit

Assumption 10

The stochastic term ui is normally distributed. The error term ‘e’ follows the normal distribution with zero mean and (constant) variance $u_i \sim N(0, \sigma^2)$

where the symbol ~ means distributed as and N stands for the normal distribution, the terms in the parentheses representing the two parameters of the normal distribution, namely, the mean and the variance. If this assumption is violated, the statistical tests such as t, and F in regression may not be valid.

Note: All the assumptions pertain to population regression function only.

Significance of stochastic disturbance term

1. The disturbance term ui is a surrogate for the variables that are left out of the model but have a collective impact on the output y.
2. The reason why those variables were left out could be many
3. We may not fully understand how those variables impact the output (theoretically weak)
4. Lack of data for those variables. Some variables are not quantitative by nature and we may not have a way to capture such data for e.g personality of an individual that impacts his/her monthly expenses
5. Peripheral variables – Some variables have a weak influence on the target and their joint influence may be very weak. Such variables can be represented by the ui
6. Intrinsic randomness in the process. For e.g. personality of individuals may vary significantly even when the most of the measurable attributes are same
7. Principle of parsimony requires that we keep our models as simple as possible (Occam's razor). If significant part of y's behavior can be captured by a few variables, then why not keep it simple. Let the other variables collective effort be represented by the ui

Disturbance term expected value

Assumptions 3 (The mean value of disturbance ui is zero) ….. Why?

1. Let linear regression model be $y_i = \beta_1 + \beta_2 X_{2i} + \beta_3 X_{3i} + \ldots + \beta_k X_i + u_i$ (a data point in K dimensions)

2. Assume $E(U_i \mid X_{2i}, X_{3i}\ldots..,X_{ki}) = W$ (W is a constant , in standard model W = 0)

3. Conditional expectations of the equation for yi can be expressed as
   a. $E(y_i \mid X_{2i}, X_{3i},\ldots..X_{ki}) = \beta_1 + \beta_2 X_{2i} + \beta_3 X_{3i} + \ldots + \beta_k X_i + W$
   b. $\Rightarrow (\beta_1 + W) + \beta_2 X_{2i} + \beta_3 X_{3i} + \ldots + \beta_k X_i$
   c. $\Rightarrow \alpha + \beta_2 X_{2i} + \beta_3 X_{3i} + \ldots + \beta_k X_i$ where $\alpha = (\beta_1 + W)$

4. Given the training data, the Xs are treated as constant while the βs are the variables

5. **If the assumption 3 is not fulfilled, we cannot solve the equation for β1!**

Heteroscedasticity of disturbance

Homoscedasiticity or Constant Variance of ui, the variance of the error / disturbance is the same regardless of the value of X.

Violation of this assumption leads to Heteroscedasticity. There are several reasons for this –

1. As the processes mature and stabilize over a number of operations, the variability in the output falls. For e.g. a new coder may show more variance in coding productivity and an experienced one
2. As one input variable grows, the process outputs vary more for e.g. as monthly household income grows, there is more choice to spend on and hence the savings may fluctuate depending on the household preferences
3. Data collection techniques improve, the data collected first may show more variations than the data collected last
4. Outliers can lead to heteroscedasticity
5. Skewness in data can lead to heteroscedasticity. For e.g. few individuals with extremely high incomes will contribute most to the variations
6. The model specified may not be the correct one. We chose a simple linear model while the model should be relatively more complex

Heteroscedasticity of disturbance (Contd…)

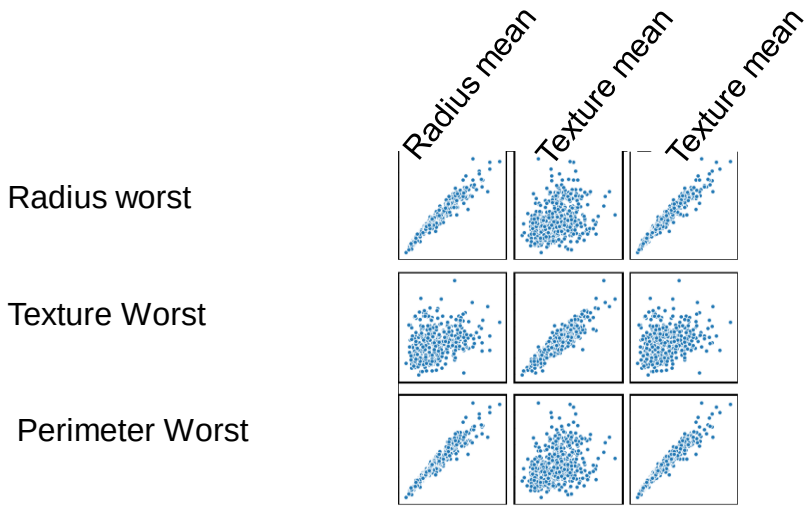
Variance of ui, homoscedastic or heteroscedastic plays no part in the determination of the coefficients. Even with heteroscedasticity the coefficients will converge to the population value of the coefficients. Infact the distribution of the coefficients remains asymptotically normally distributed.

The problems is, the coefficients will have lot of variance and make the overall model less accurate. The OLS method does not take into account the heteroscedastic nature of certain attributes relative to others. It gives same weightage to all attributes. On the other hand GLS (Generalized Least Square) method takes into account such difference and gives lesser weight to attributes with more heteroscedasticity and in the process result in better models

Multi-Collinearity

Multi Collinearity – What is it?

Multicollinearity is that situation where the independent variables in the linear model are not truly independent i.e. they are correlated. For e.g. in the Wisconsin Breast Cancer dataset the first three attributes “radius mean”, “perimeter mean” and “texture mean” is shown below. The first two are strongly correlated



Multi Collinearity – Types of multicollinearity

1. **Structural multicollinearity**: This type occurs when we create features from existing features and build a model using all of the features. For example, using “Radius” and “Area” as two variables. When features are generated, ensure the generated feature and the original features do not strongly correlate, if they do, you may want to drop the original feature as long as the generated feature contains all the information from the original

2. **Data multicollinearity**: This type of multicollinearity is an artifact of the data itself. The nature of the variables is such that they correlate. For e.g. in auto-mpg.csv, the columns “weight” and “horsepower” of a car will correlate positively. In case there are such correlating variables in the data, they may be combined into a composite variable using techniques such as PCA

Multi Collinearity

1. Assumptions 8 **(**There no perfect collinearity between the predictor variables X) – is technically known as the assumption of no collinearity or no multicollinearity when more than one variables is involved

2. Formally, no collinearity means there exists no two numbers λ2 and λ3 such that λ2x2 + λ3X3 = 0. If such a relationship exists, then X2 and X3 are said to be collinear or linearly dependent

3. On the other hand if the equation holds only when λ2 and λ3 = 0 , then the variables are non-collinear

4. In simple terms, multicollinearity is the situation when two or more variables used in a model, are related to each other i.e. change in values of one leads to change in values of other

5. The problem with having multicollinearity is in the inability to understand how one variable influences the target. There is no way to estimate separate influence of each variable on target. Thus no way to estimate the partial regression coefficients

Multi Collinearity (Contd…)

6. If multicollinearity is perfect, the regression coefficients of X variables are indeterminate and their standard errors are infinite

7. If multicollinearity is less than perfect, the regression coefficients, although determinate, posses large standard errors, which means the coefficients cannot be estimated with confidence

8. High degree of multicollinearity will not take away the property of being best unbiased linear estimators. It violates none of the regression assumptions. The only problem is that it will result in hard to determine coefficients with small standard errors

9. But the same problem occurs when we have too few observations or the independent variables have small variances

Multi Collinearity – What is the problem?

1. Independent variables should be *independent* of one another. Instead, if they correlate strongly, it can lead to sub-optimal model and mislead in terms of statistical results such as P values

2. The main objective of regression analysis is to express the relationship between each predictor variable and the dependent variable independently.

3. The regression coefficient is a measure of mean change in the dependent variable for each 1 unit change in an independent variable keeping all other independent variables constant. However, with collinear independent variables, it will not be possible to change one variable keeping others constant!

4. The coefficient estimates for an independent variable Vs the target variable can swing wildly based on inclusion or exclusion of other correlated independent variables are in the model. The coefficients become very sensitive to changes in the model structure

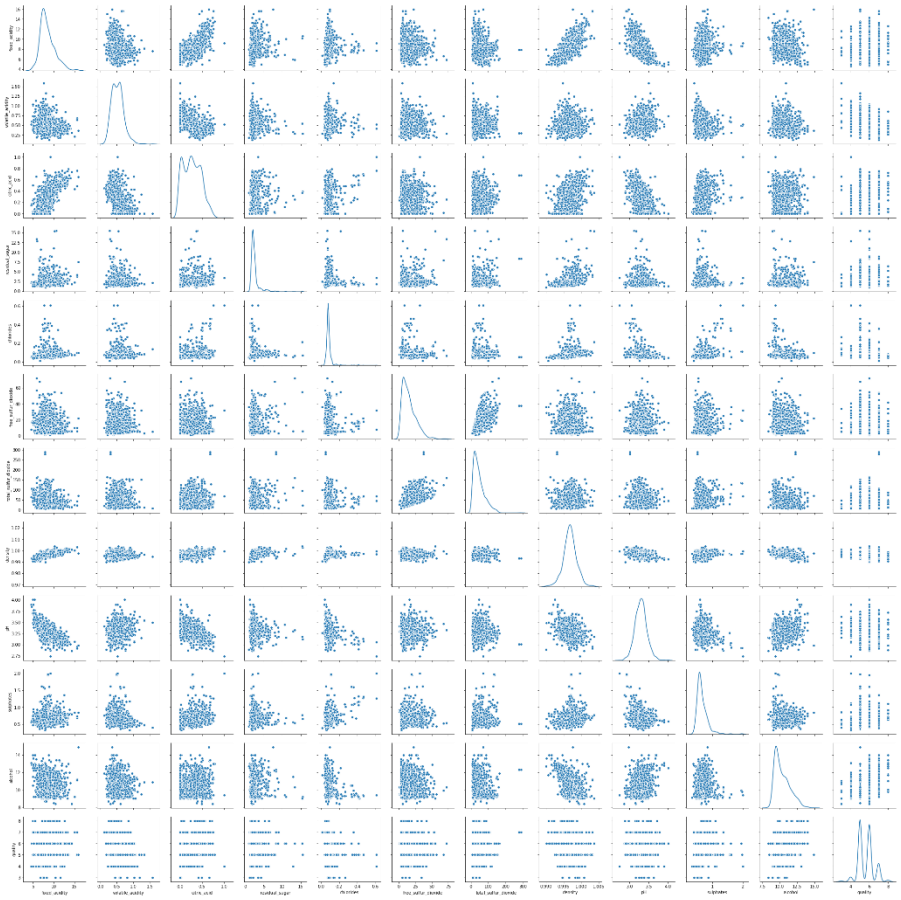
5. Multicollinearity reduces the precision of the estimate coefficients, which weakens the statistical power of your regression model. You might not be able to trust the p-values to identify independent variables that are statistically significant

Multi Collinearity – Testing for multicollinearity with Variation Inflation Factor (VIF)

1. The variance inflation factor (VIF) identifies correlation between independent variables and the strength of that correlation.
2. Statsmodel based linear models provide a VIF for each independent variable
3. VIFs start at 1 and have no upper limit.
    a. A value of 1 indicates that there is no correlation between this independent variable and any others
    b. VIFs between 1 and 5 suggest that there is a moderate correlation, but it is not severe enough to warrant corrective measures
    c. VIFs greater than 5 represent critical levels of multicollinearity where the coefficients are poorly estimated, and the p-values are questionable.

Testing for multicollinearity with Variation Inflation Factor (VIF)

Red wines dataset, correlation between features and VIF values
Most attributes have very high value of VIF



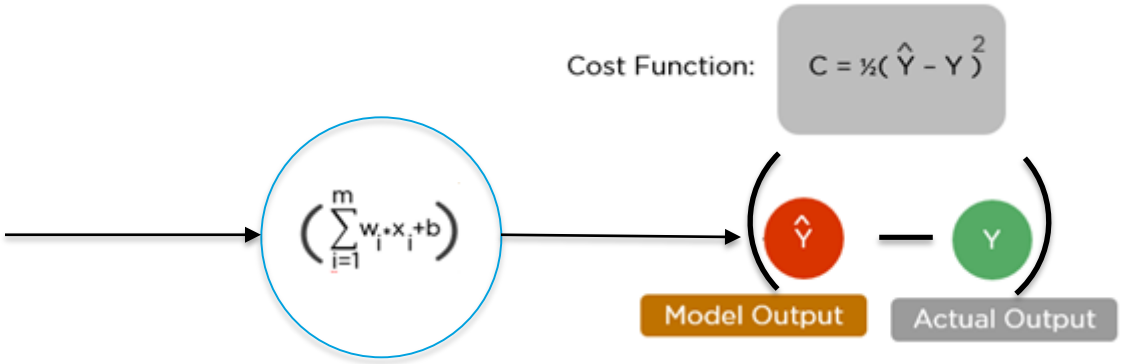
fixed_acidity ---> 80.0219390812402
volatile_acidity ---> 16.57719262878206
citric_acid ---> 9.774864397481704
residual_sugar ---> 4.906541592370461
chlorides ---> 6.529251770198401
free_sulfur_dioxide ---> 6.448644902127925
total_sulfur_dioxide ---> 6.877056111151861
density ---> 1445.240488945372
pH ---> 1037.4662099590764
sulphates ---> 20.65264657492166
alcohol ---> 121.46712238121306

StatsModel+Linea
+Model+Wine+Vl

Loss Function &
Optimization Algorithm

Loss function (Mean Square Loss)

1. What is an optimization algorithm and what is its use? - Optimization algorithms helps us to ***minimize (or maximize)*** an **Objective** function (*another name for* ***Error*** *function*) **E(x)** which is simply a mathematical function dependent on the Model's internal **learnable parameters** which are used in computing the target values(**Y**) from the set of *predictors*(**X**) used in the model

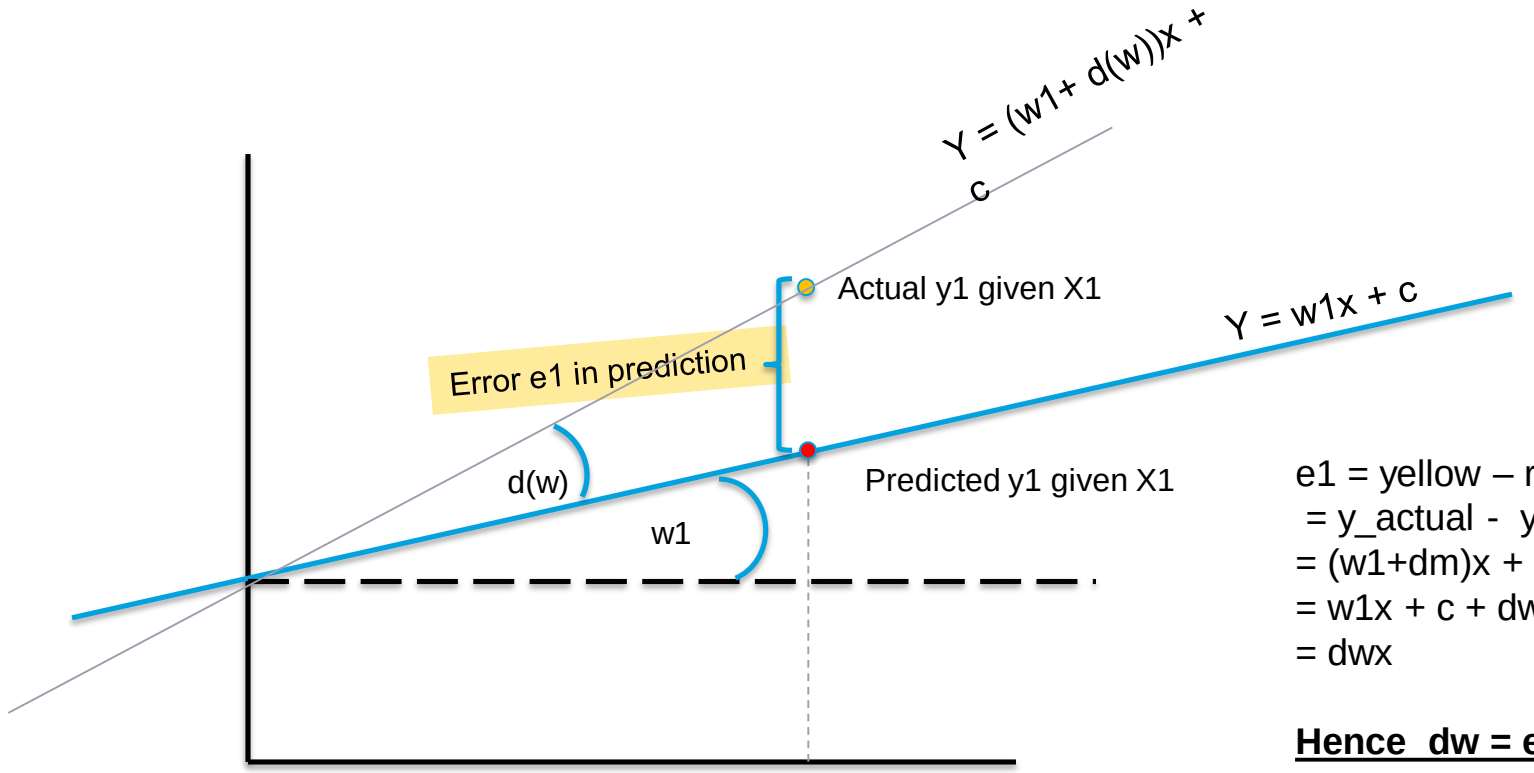


2. C = ½((wi.xi + b) – y). In this expression Xi and y come from the data and are given. What the ML algorithm learns is the weight wi and bias b. Thus C = f(wi, b)

3. The optimizer algorithms try to estimate the values of wi and b which when used, will give minimum or maximum C. In ML we look for minimum

Relation between error and change in weights

Since part of neuron function is linear equation (before applying the non-linear transformation), the error at each neuron can be expressed in terms of the linear equation.



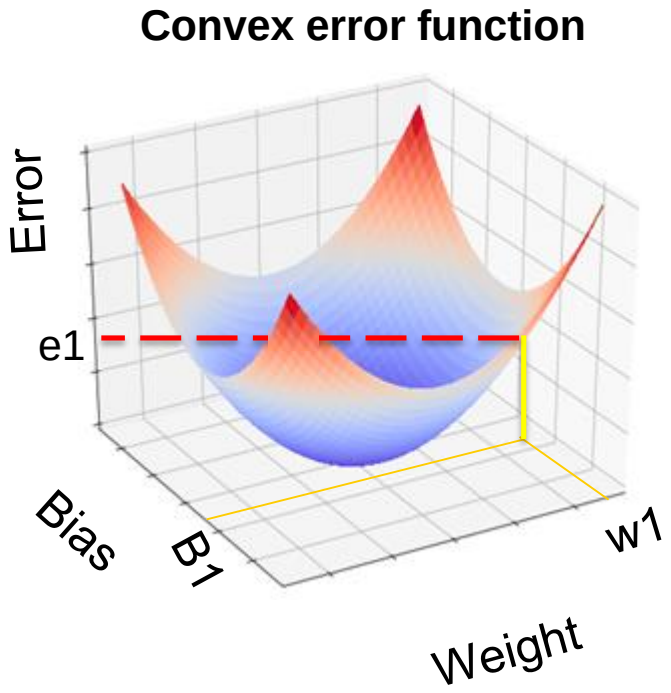
e1 = yellow – red
= y_actual - y_pred
= (w1+dm)x + c - (w1x + c)
= w1x + c + dwx – w1x -c
= dwx

Hence dw = e1 / x

The change required in m (dw) is e1/x. However, change required w.r.t another data point may be different. To prevent jumping around with dw, we moderate the change in W by introducing a **learning rate l** . Hence dw = l(e1/x)

Gradient Descent

The challenge is, all the weights in all the inputs need to be adjusted. It is not manually possible to find the right combination of weights using brute force. Instead, the machine learning algorithm uses a learning function called gradient descent



1. A random combination of bias B1 and input weights W1 (showing only one as more than one is not possible to visualize)

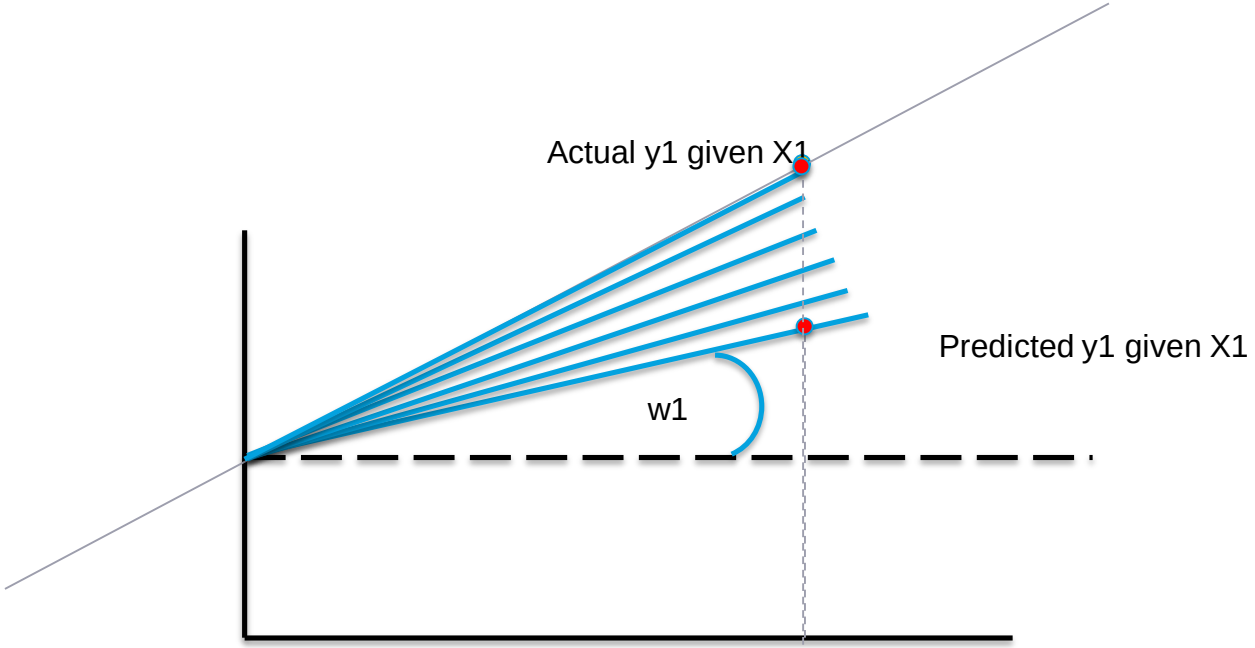
2. Each combination of W1 and B1 is one particular linear model in a neuron. That model is associated with proportionate error e1 (red dashed line).

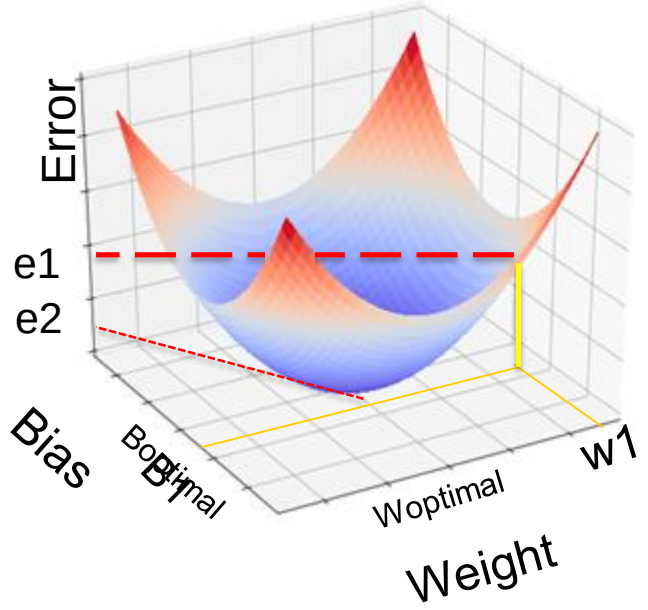
3. Objective is to drive e1 towards 0. For which we need to find the optimal weight (Woptimal) and bias (Boptimal)

4. The algorithm uses gradient descent algorithm to change bias and weight form starting values of B1 and W1 towards the Boptimal, Woptimal.

Note: in 3D error surface can be visualized as shown but not in more than 3 dimensions

Gradient Descent





Least error E2 is at the global minima of the convex function which only one unique combination of weight (woptimal) and bias (boptimal) will fetch us.

Gradient Descent

1. Let target value for a training example X be y i.e. The data frame used for training has value X,y

2. Let the model (represented by random m and c) predict the value for the training example X to be yhat

3. Error in prediction is E = yhat – y. If we sum all the errors across all data points, some will be positive some negative and thus cancel out

4. To prevent the sum of errors becoming 0, we square the error i.e. E = (y – yhat)^2. Note: in squared expression, y – yhat or yhat – y mean the same

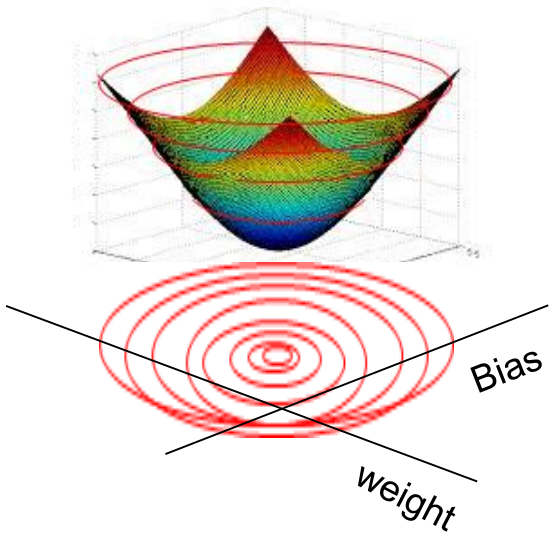
5. Sum of (y – yhat)^2 across all the X values is called SSE (Sum of Squared Errors)

6. Using gradient descent (descend towards the global minima). Gradient descent uses partial derivatives i.e how the SSE changes on slightly modifying the model parameters m and c one at a time

   d(E) / d(m) = d(sum(yhat – y)^2) / d(m)
   d(E) / d(c) = d(sum(yhat – y)^2) / d(c)

Gradient Descent

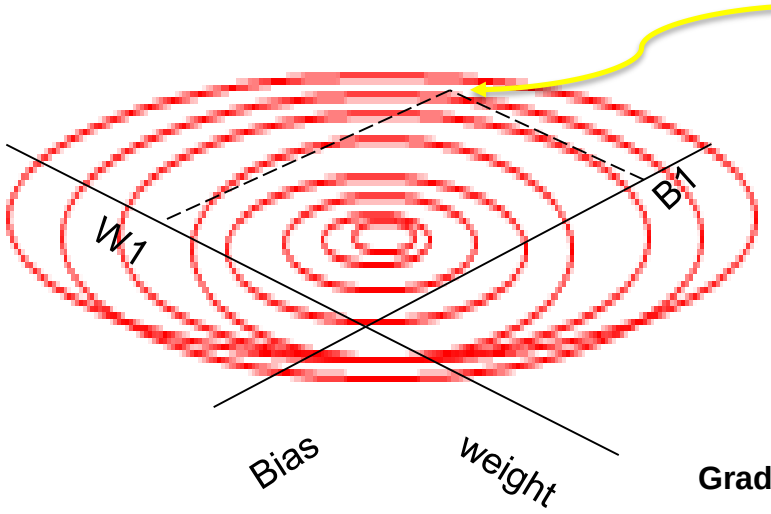
Transform our error function (which is a quadratic / convex function) into a contour graph. Gradient is always found on the input model parameters only



1. Every ring on the error function represents a combination of coefficients (m1 and m2 in the image) which result in same quantum of error i.e. SSE

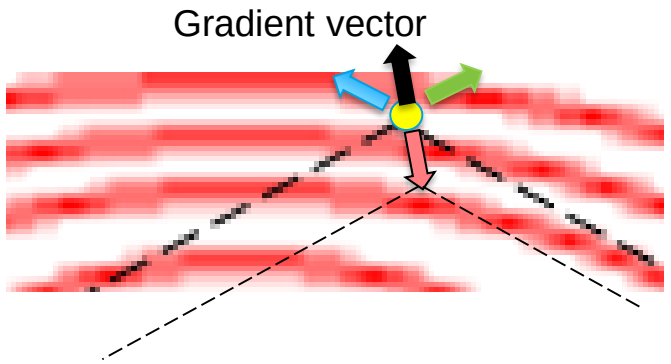
2. Let us convert that to a 2d contour plot. In the contour plot, every ring represents one quantum of error.

3. The innermost ring / bull's eye is the combination of the coefficients that gives the lease SSE



About the contour graph –

1. Outermost circle is highest error while innermost is the least error circle
2. A circle represents combination of paramters which result in same error. Moving on a circle will not reduce error.
3. Objective is to start from anywhere but reach the innermost circle



Gradient Descent Steps –

1. First evaluate dy(error)/d(weight) to find the direction of highest increase in error given a unit change in weight (Blue arrow). Partial derivative w.r.t. to weight
2. Next find dy(error) /d(bias) to find the direction of highest increase in error given a unit change in bias (green arrow). Partial derivative w.r.t. to bias
3. Partial derivatives give the gradient in the given axis and gradient is a vector
4. Add the two vectors to get the direction of gradient (black arrow) i.e. direction of max increase in error
5. We want to decrease error, so find negative of the gradient i.e. opposite to black arrow (Orange arrow). The arrow tip is new value of bias and weight.
6. Recalculate the error at this combination an iterate to step 1 till movement in any direction only increases the error

Gradient Descent

1. Gradient descent is a way to minimize an objective function / cost function such as Sum of Squared Errors (SSE) that is dependent on model parameters of weight / slope and bias

2. The parameters are updated in the direction opposite to the direction of the gradient (direction of maximum increase) of the objective function

3. In other words we change the values of weight and bias following the direction of the slope of the surface of the error function down the hill until we reach minima

4. <u>This movement from starting weight and bias to optimal weight and bias may not happen in one shot</u>. It is likely to happen in multiple iterations. The values change in steps

5. The step size can be influenced using a parameter called Learning Rate. It decides the size of the steps i.e. the amount by which the parameters are updated. Too small learning step will slow down the entire process while too large may lead to an infinite loop

6. The mathematical expression of gradient descent