SNS COLLEGE OF ENGINEERING

Kurumbapalayam (Po), Coimbatore – 641 107

An Autonomous Institution

Accredited by NBA – AICTE and Accredited by NAAC – UGC with 'A' Grade
Approved by AICTE, New Delhi & Affiliated to Anna University, Chennai

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

COURSE NAME : 19CS501 Introduction to Machine Learning

III YEAR /V SEMESTER

Unit 2- SUPERVISED LEARNING

Topic : Linear Models for Regression


Salary vs Experience
Salary
120000
100000
80000
60000
40000
2
4
6
8
10
Years of Experience

Linear Regression for Machine Learning

- Linear regression is perhaps one of the most well known and well understood algorithms in statistics and machine learning.

  - ✓ ***Why linear regression belongs to both statistics and machine learning.***
  - ✓ ***The many names by which linear regression is known.***
  - ✓ ***The representation and learning algorithms used to create a linear regression model.***
  - ✓ ***How to best prepare your data when modeling using linear regression.***

Isn’t Linear Regression from Statistics?

- ☐ Isn’t it a technique from statistics?

- ☐ Machine learning, more specifically the field of predictive modeling is primarily concerned with minimizing the error of a model or making the most accurate predictions possible, at the expense of explain ability. In applied machine learning we will borrow, reuse and steal algorithms from many different fields, including statistics and use them towards these ends.

- ☐ As such, linear regression was developed in the field of statistics and is studied as a model for understanding the relationship between input and output numerical variables, but has been borrowed by machine learning. It is both a statistical algorithm and a machine learning algorithm.

Many Names of Linear Regression

Linear regression is a linear model, e.g. a model that assumes a linear relationship between the input variables (x) and the single output variable (y). More specifically, that y can be calculated from a linear combination of the input variables (x).

When there is a single input variable (x), the method is referred to as simple linear regression. When there are multiple input variables, literature from statistics often refers to the method as multiple linear regression.

Different techniques can be used to prepare or train the linear regression equation from data, the most common of which is called Ordinary Least Squares. It is common to therefore refer to a model prepared this way as Ordinary Least Squares Linear Regression or just Least Squares Regression.

Linear Regression Model Representation

Linear regression is an attractive model because the representation is so simple.

The representation is a linear equation that combines a specific set of input values (x) the solution to which is the predicted output for that set of input values (y). As such, both the input values (x) and the output value are numeric.

The linear equation assigns one scale factor to each input value or column, called a coefficient and represented by the capital Greek letter Beta (B). One additional coefficient is also added, giving the line an additional degree of freedom (e.g. moving up and down on a two-dimensional plot) and is often called the intercept or the bias coefficient.

For example, in a simple regression problem (a single x and a single y), the form of the model would be:

$$y = B0 + B1*x$$

Linear Regression Model Representation

In higher dimensions when we have more than one input (x), the line is called a plane or a hyper-plane. The representation therefore is the form of the equation and the specific values used for the coefficients (e.g. B0 and B1 in the above example).

It is common to talk about the complexity of a regression model like linear regression. This refers to the number of coefficients used in the model.

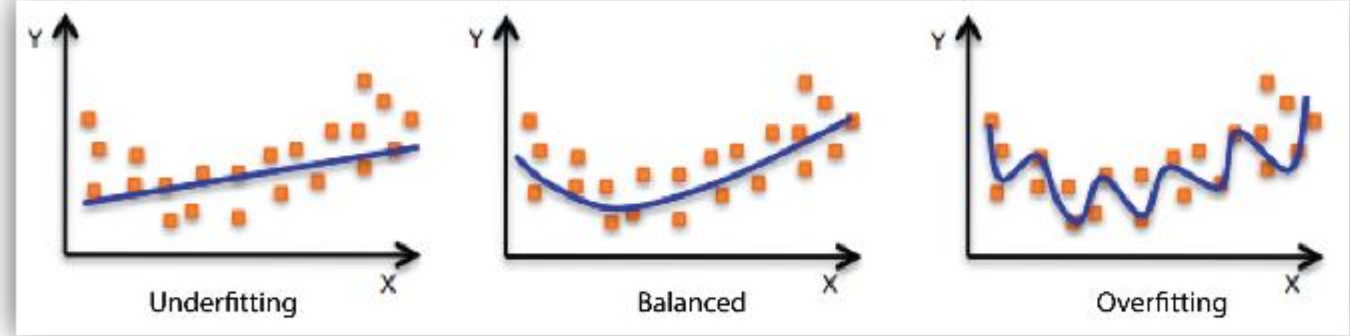
When a coefficient becomes zero, it effectively removes the influence of the input variable on the model and therefore from the prediction made from the model (0 * x = 0).

This becomes relevant if you look at regularization methods that change the learning algorithm to reduce the complexity of regression models by putting pressure on the absolute size of the coefficients, driving some to zero.

Linear Regression Learning the Model

- Learning a linear regression model means estimating the values of the coefficients used in the representation with the data that we have available.

- There are many more techniques because the model is so well studied. Take note of Ordinary Least Squares because it is the most common method used in general. Also take note of Gradient Descent as it is the most common technique taught in machine learning class



Simple Linear Regression

- Simple linear regression is used to estimate the relationship between two quantitative variables. You can use simple linear regression when you want to know:
- How strong the relationship is between two variables (e.g. the relationship between rainfall and soil erosion).
- The value of the dependent variable at a certain value of the independent variable (e.g. the amount of soil erosion at a certain level of rainfall).

Example

You are a social researcher interested in the relationship between income and happiness. You survey 500 people whose incomes range from $15k to $75k and ask them to rank their happiness on a scale from 1 to 10.

Your independent variable (income) and dependent variable (happiness) are both quantitative, so you can do a regression analysis to see if there is a linear relationship between them.

Assumptions of simple linear regression

- Simple linear regression is a parametric test, meaning that it makes certain assumptions about the data. These assumptions are:

1. **Homogeneity of variance** (homoscedasticity): the size of the error in our prediction doesn’t change significantly across the values of the independent variable.
2. **Independence of observations**: the observations in the dataset were collected using statistically valid sampling methods, and there are no hidden relationships among observations.
3. **Normality:** The data follows a normal distribution.

Linear regression makes one additional assumption:

4. The relationship between the independent and dependent variable is linear: the line of best fit through the data points is a straight line (rather than a curve or some sort of grouping factor).

How to perform a simple linear regression

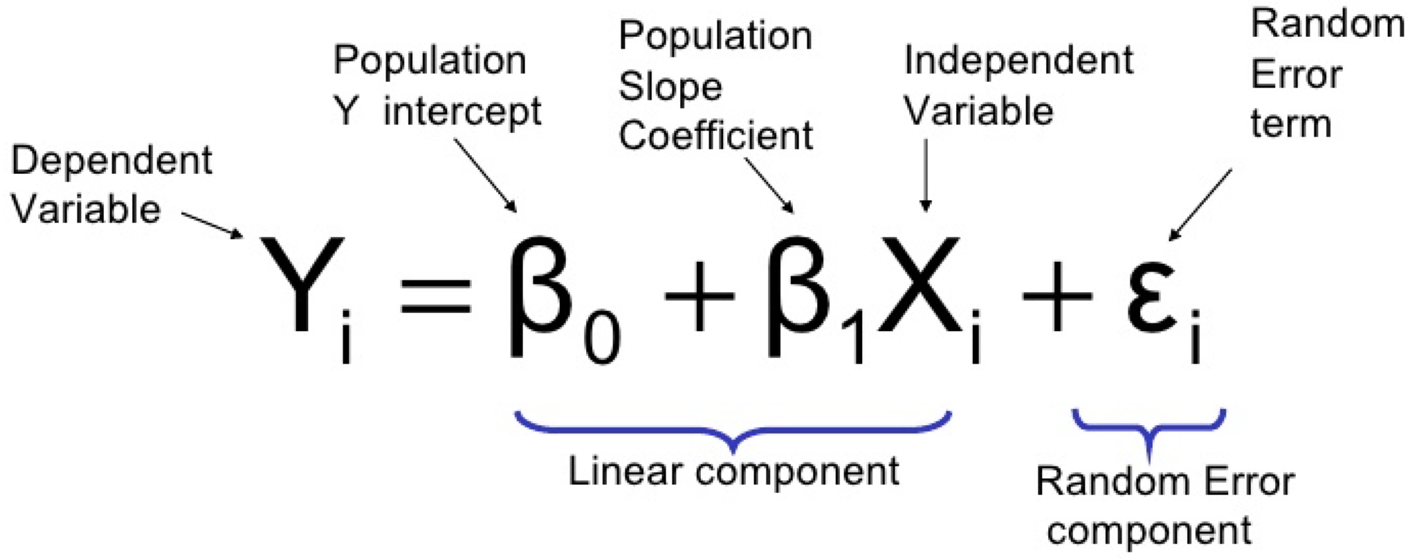
The formula for a simple linear regression is:

$$y = \beta_0 + \beta_1 X + \varepsilon$$

- y is the predicted value of the dependent variable (y) for any given value of the independent variable (x).
- B0 is the intercept, the predicted value of y when the x is 0.
- B1 is the regression coefficient – how much we expect y to change as x increases.
- x is the independent variable (the variable we expect is influencing y).
- e is the error of the estimate, or how much variation there is in our estimate of the regression coefficient.

Ordinary Least Square (OLS) Method

The simple linear regression is a model with a single regressor (independent variable) x that has a relationship with a response (dependent or target) y that is a



Dependent Variable

Population Y intercept

Population Slope Coefficient

Independent Variable

Random Error term

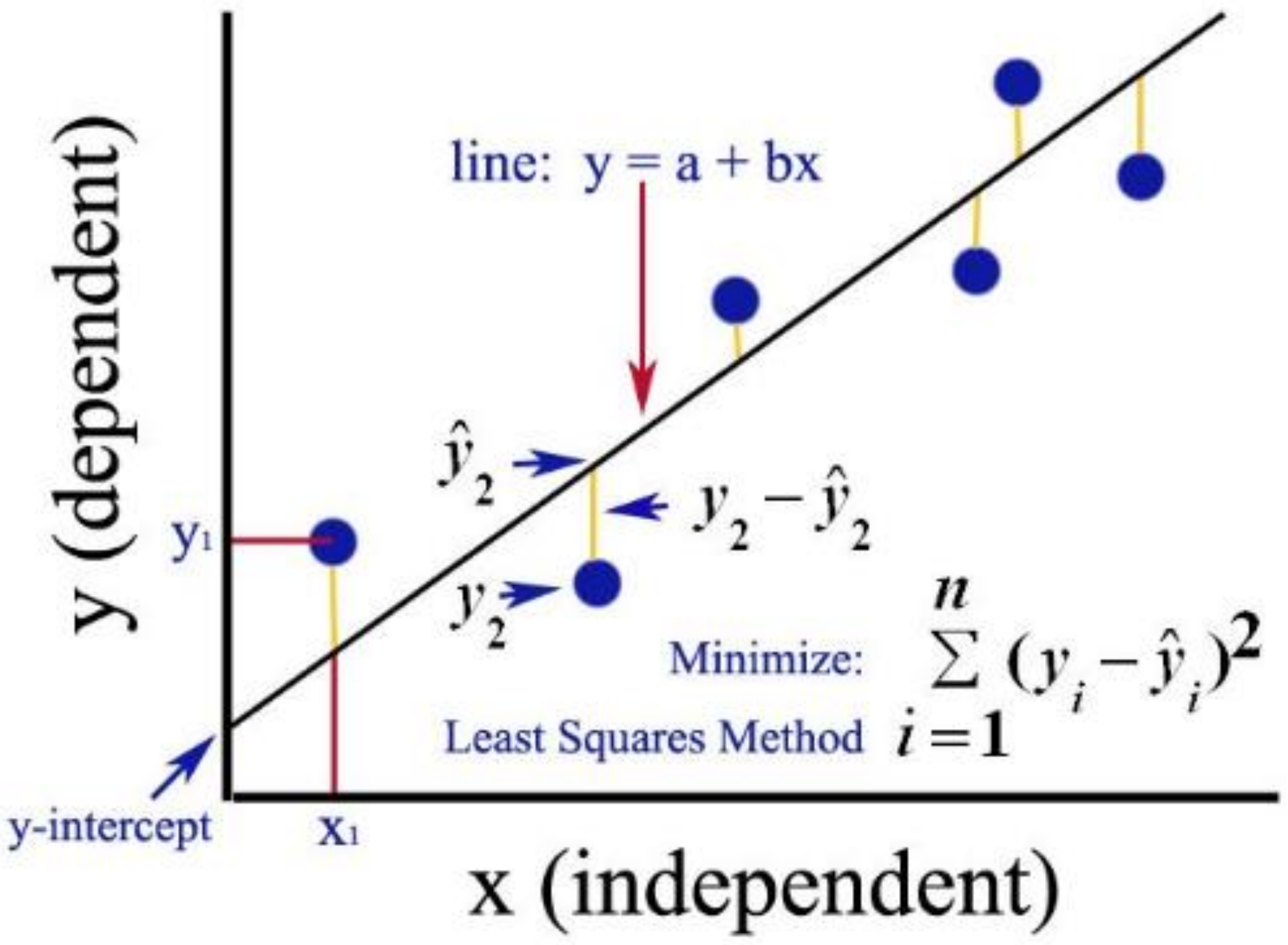
$$Y_i = \underbrace{\beta_0 + \beta_1 X_i}_{\text{Linear component}} + \underbrace{\varepsilon_i}_{\text{Random Error component}}$$

This is a line where y is the dependent variable we want to predict, x is the independent variable, and β0 and β1 are the coefficients that we need to estimate.

Ordinary Least Square (OLS) Method

Estimation of β0 and β1 :

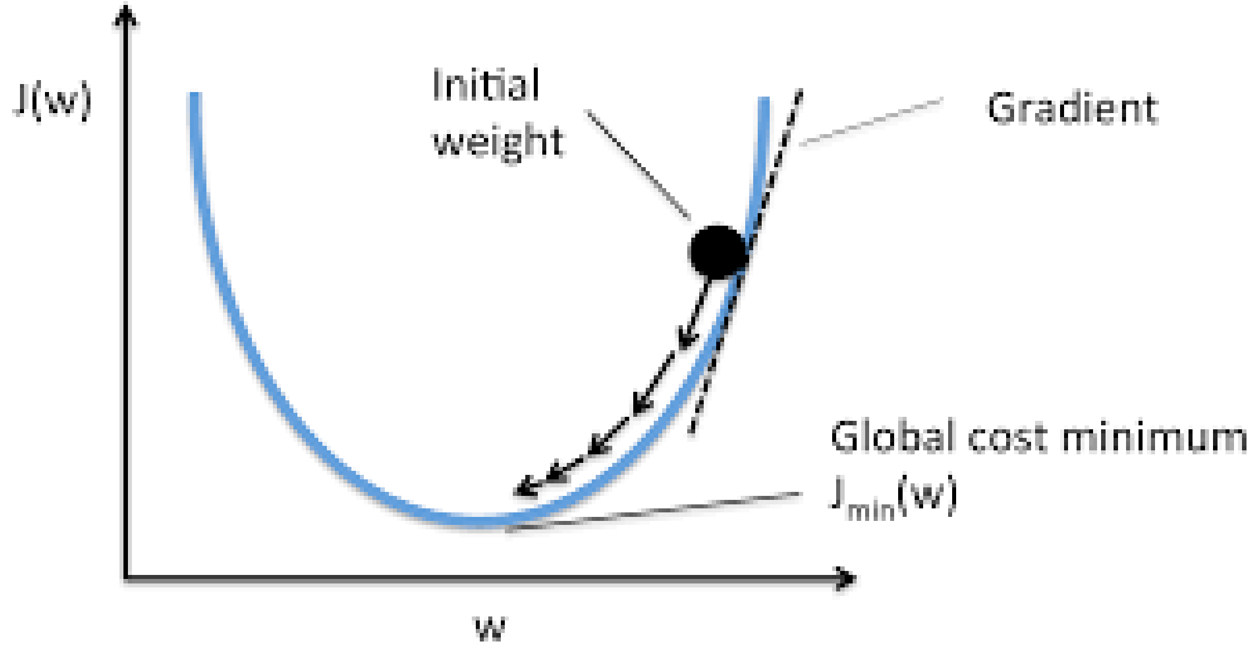
The OLS method is used to estimate β0 and β1. The OLS method seeks to minimize the sum of the squared residuals. This means from the given data we calculate the distance from each data point to the regression line, square it, and the sum of all of the squared errors together.

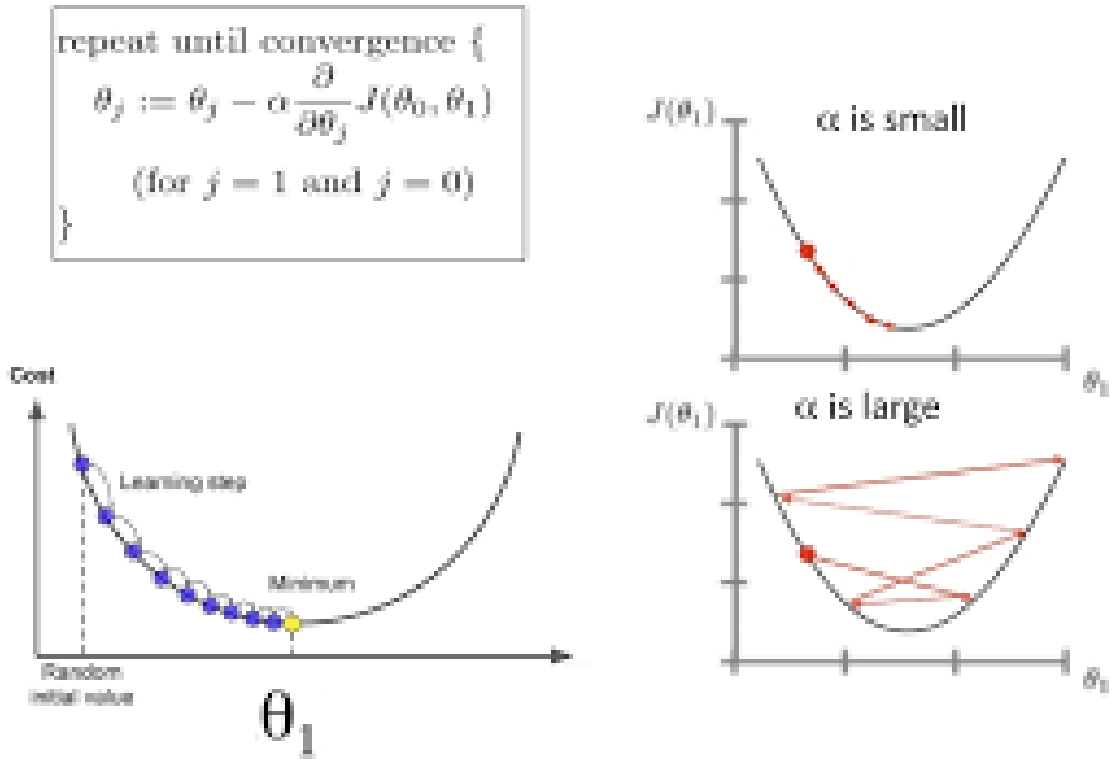


Gradient Descent

- Gradient descent is an optimization algorithm used to find the values of parameters (coefficients) of a function (f) that minimizes a cost function (cost).

- Gradient descent is best used when the parameters cannot be calculated analytically (e.g. using linear algebra) and must be searched for by an optimization algorithm.





```
repeat until convergence {
    $\theta_j := \theta_j - \alpha \frac{\partial}{\partial \theta_j} J(\theta_0, \theta_1)$
        (for $j = 1$ and $j = 0$)
}
```

Intuition for Gradient Descent

- Think of a large bowl like what you would eat cereal out of or store fruit in. This bowl is a plot of the cost function (f).

A random position on the surface of the bowl is the cost of the current values of the coefficients (cost).

The bottom of the bowl is the cost of the best set of coefficients, the minimum of the function.

The goal is to continue to try different values for the coefficients, evaluate their cost and select new coefficients that have a slightly better (lower) cost.

Repeating this process enough times will lead to the bottom of the bowl and you will know the values of the coefficients that result in the minimum cost.

Tips for Gradient Descent

- **Plot Cost versus Time**: Collect and plot the cost values calculated by the algorithm each iteration. The expectation for a well performing gradient descent run is a decrease in cost each iteration. If it does not decrease, try reducing your learning rate.

- **Learning Rate**: The learning rate value is a small real value such as 0.1, 0.001 or 0.0001. Try different values for your problem and see which works best.

- **Rescale Inputs**: The algorithm will reach the minimum cost faster if the shape of the cost function is not skewed and distorted. You can achieved this by rescaling all of the input variables (X) to the same range, such as [0, 1] or [-1, 1].

- **Few Passes**: Stochastic gradient descent often does not need more than 1-to-10 passes through the training dataset to converge on good or good enough coefficients.

- **Plot Mean Cost**: The updates for each training dataset instance can result in a noisy plot of cost over time when using stochastic gradient descent. Taking the average over 10, 100, or 1000 updates can give you a better idea of the learning trend for the algorithm

Regularization

- There are extensions of the training of the linear model called regularization methods. These seek to both minimize the sum of the squared error of the model on the training data (using ordinary least squares) but also to reduce the complexity of the model (like the number or absolute size of the sum of all coefficients in the model).

- Two popular examples of regularization procedures for linear regression are:

- **Lasso Regression**: where Ordinary Least Squares is modified to also minimize the absolute sum of the coefficients (called L1 regularization).
- **Ridge Regression**: where Ordinary Least Squares is modified to also minimize the squared absolute sum of the coefficients (called L2 regularization).

Making Predictions with Linear Regression

Given the representation is a linear equation, making predictions is as simple as solving the equation for a specific set of inputs.

Let's make this concrete with an example. Imagine we are predicting weight (y) from height (x). Our linear regression model representation for this problem would be:

y = B0 + B1 * x1

or

weight =B0 +B1 * height

Where B0 is the bias coefficient and B1 is the coefficient for the height column. We use a learning technique to find a good set of coefficient values. Once found, we can plug in different height values to predict the weight.

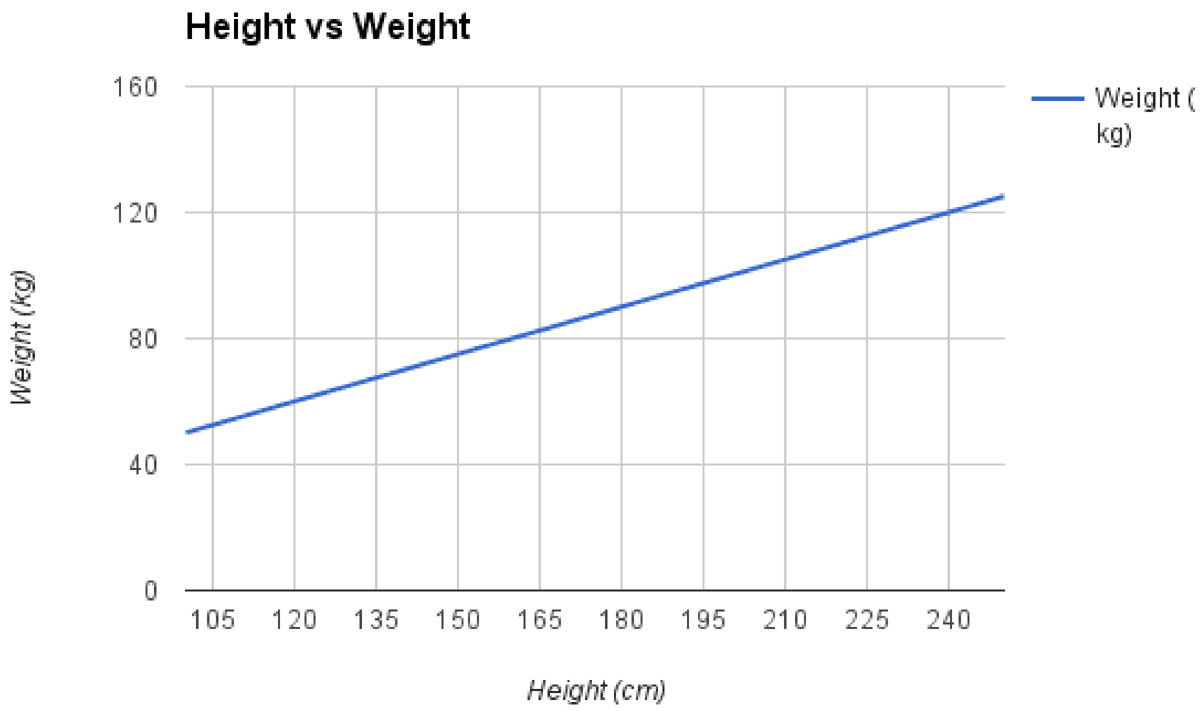
Making Predictions with Linear Regression

For example, lets use B0 = 0.1 and B1 = 0.5. Let's plug them in and calculate the weight (in kilograms) for a person with the height of 182 centimeters.

weight = 0.1 + 0.5 * 182

weight = 91.1

You can see that the above equation could be plotted as a line in two-dimensions. The B0 is our starting point regardless of what height we have. We can run through a bunch of heights from 100 to 250 centimeters and plug them to the equation and get weight values, creating our line.



Preparing Data For Linear Regression

- **Linear Assumption**. Linear regression assumes that the relationship between your input and output is linear. It does not support anything else. This may be obvious, but it is good to remember when you have a lot of attributes. You may need to transform data to make the relationship linear (e.g. log transform for an exponential relationship).
- **Remove Noise**. Linear regression assumes that your input and output variables are not noisy. Consider using data cleaning operations that let you better expose and clarify the signal in your data. This is most important for the output variable and you want to remove outliers in the output variable (y) if possible.
- **Remove Collinearity**. Linear regression will over-fit your data when you have highly correlated input variables. Consider calculating pairwise correlations for your input data and removing the most correlated.
- **Gaussian Distributions.** Linear regression will make more reliable predictions if your input and output variables have a Gaussian distribution. You may get some benefit using transforms (e.g. log or BoxCox) on you variables to make their distribution more Gaussian looking.
- **Rescale Inputs**: Linear regression will often make more reliable predictions if you rescale input variables using standardization

Assessment

- The common names used when describing linear regression models.
- The representation used by the model.
- Learning algorithms used to estimate the coefficients in the model.
- Rules of thumb to consider when preparing data for use with linear regression

REFERENCES

1. Tom M. Mitchell, “Machine Learning”, McGraw-Hill Education (India) Private Limited, 2013.
2. Trevor Hastie, Robert Tibshirani, Jerome Friedman, “The Elements of Statistical Learning: Data Mining, Inference, and Prediction”, Springer; Second Edition, 2009.

THANK YOU