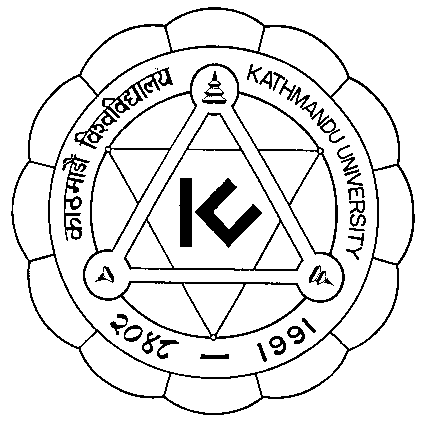
**KATHMANDU UNIVERSITY**

**Department of Computer Science & Engineering**

**Dhulikhel, Kavre.**



**A Mini-project report On**

**Linear Regression Simulation**

**COMP 342**

**Submitted by:**

**Robin Narsingh Ranabhat**

**Roll no. 42**

**CE 3rd year/2nd sem**

**Submitted to**

**Mr. Dhiraj Shrestha**

**Department of Computer Science and Engineering**

**Submission Date: 27/07/2018**

Contents

[Introduction 3](#_Toc520449277)

[Objectives 4](#_Toc520449278)

[Features 4](#_Toc520449279)

[Design and Methods 4](#_Toc520449280)

[TOOLS USED 5](#_Toc520449281)

[LANGUAGE USED 5](#_Toc520449282)

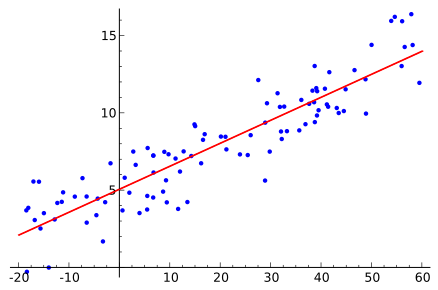
[CONCLUSION 6](#_Toc520449283)

[Screenshot 6](#_Toc520449284)

[References 7](#_Toc520449285)

# Introduction

In statistics, linear regression is a linear approach to modeling the relationship between a scalar response (or dependent variable) and one or more explanatory variables (or independent variables). The case of one explanatory variable is called simple linear regression. For more than one explanatory variable, the process is called multiple linear regression. This term is distinct from multivariate linear regression, where multiple correlated dependent variables are predicted, rather than a single scalar variable.

  
Illustration 1: Linear Regression Visualization

# Objectives

The main objectives of this project are:

* To depict the features of regression mechanism
* To simulate how a regression model learns

# Features

* Add new data points of our own for which the line of best fit is generated

# Design and Methods

In the source code, I used a technique called gradient descent that changes weight and bias of the line , which we want to fit to the points.

Basically, using a mouse event listener, users can draw as many points as they want on the canvas.

And the line of best fit automatically moves accordingly.



In this piece of code, we generate a canvas and setup the canvas such that users can give as many points as they want on which line of best fit will work.

## TOOLS USED

Atom IDE:

Atom is a free and open-source text and source code editor for macOS, Linux, and Microsoft Windows with support for plug-ins written in Node.js, and embedded Git Control, developed by GitHub.

## LANGUAGE USED

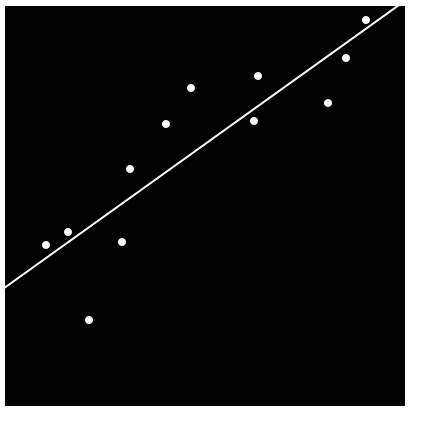
Javascript:  
Informally speaking,it's the most ubiquitous programming language in computing right now that makes it so versatile and useful. Just think about how many people bring JavaScript environments (tablets, phones) into bed with them at night or in the morning. When a developer is into writing JavaScript and want to test or try something out, they almost always have a fast, easy way to run JavaScript right at hand.  
  
p5.js:  
It’s among many, one of the drawing libraries provided for artists to code beautiful concepts and designs.

Moreover, because it’s a JavaScript, we can practically run it in any device with a browser.

# Conclusion

The main aim of this mini-project is to model and simulate various properties of a linear regression model using JavaScript and p5.js graphics library. With this mini-project, I hope to exhibit the practical implementation of the knowledge I’ve gained in the respective field.

## Screenshot

  
Illustration 2: screen of the linear regression simulation

# References

* En.wikipedia.org. (2018). Linear Regression. [online] Available at: https://en.wikipedia.org/wiki/Linear\_regression [Accessed 31 May 2018].