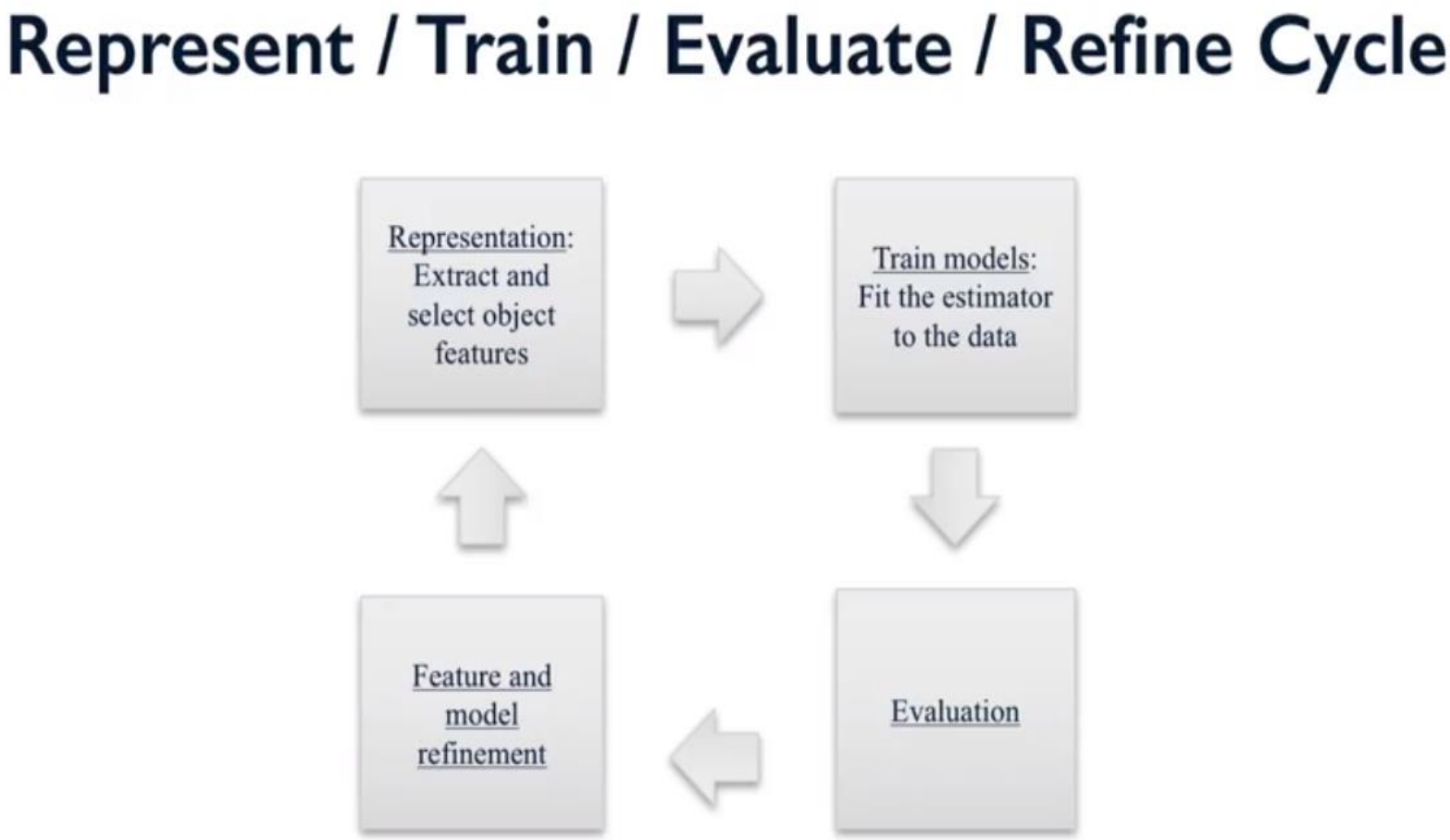
COMP9321:
Data services engineering

Week 8: Linear Regression & Project Briefing

Term 3, 2019

By Mortada Al-Banna, CSE UNSW

Refresher


Represent / Train / Evaluate / Refine Cycle
Representation: Extract and select object features
Train models: Fit the estimator to the data
Evaluation
Feature and model refinement

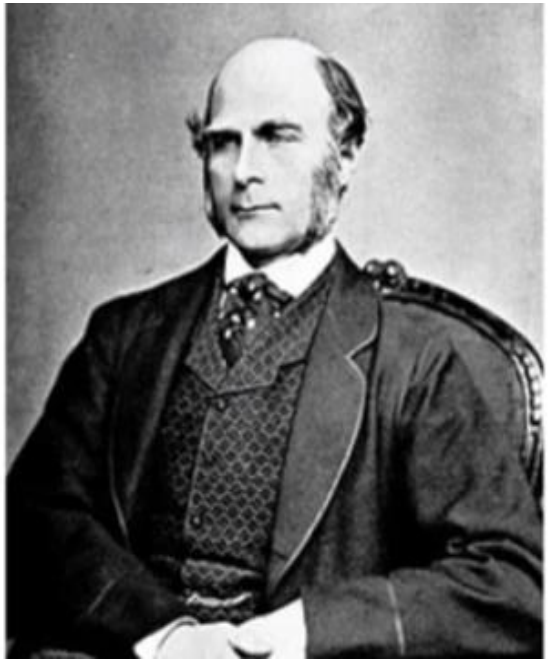
Regression Analysis

https://www.slideshare.net/Simplilearn/linear-regression-analysis-linear-regression-in-python-machine-learning-algorithms-simplilearn?qid=c0cbd932-3ad1-45ec-b123-36805981982d&v=&b=&from_search=4

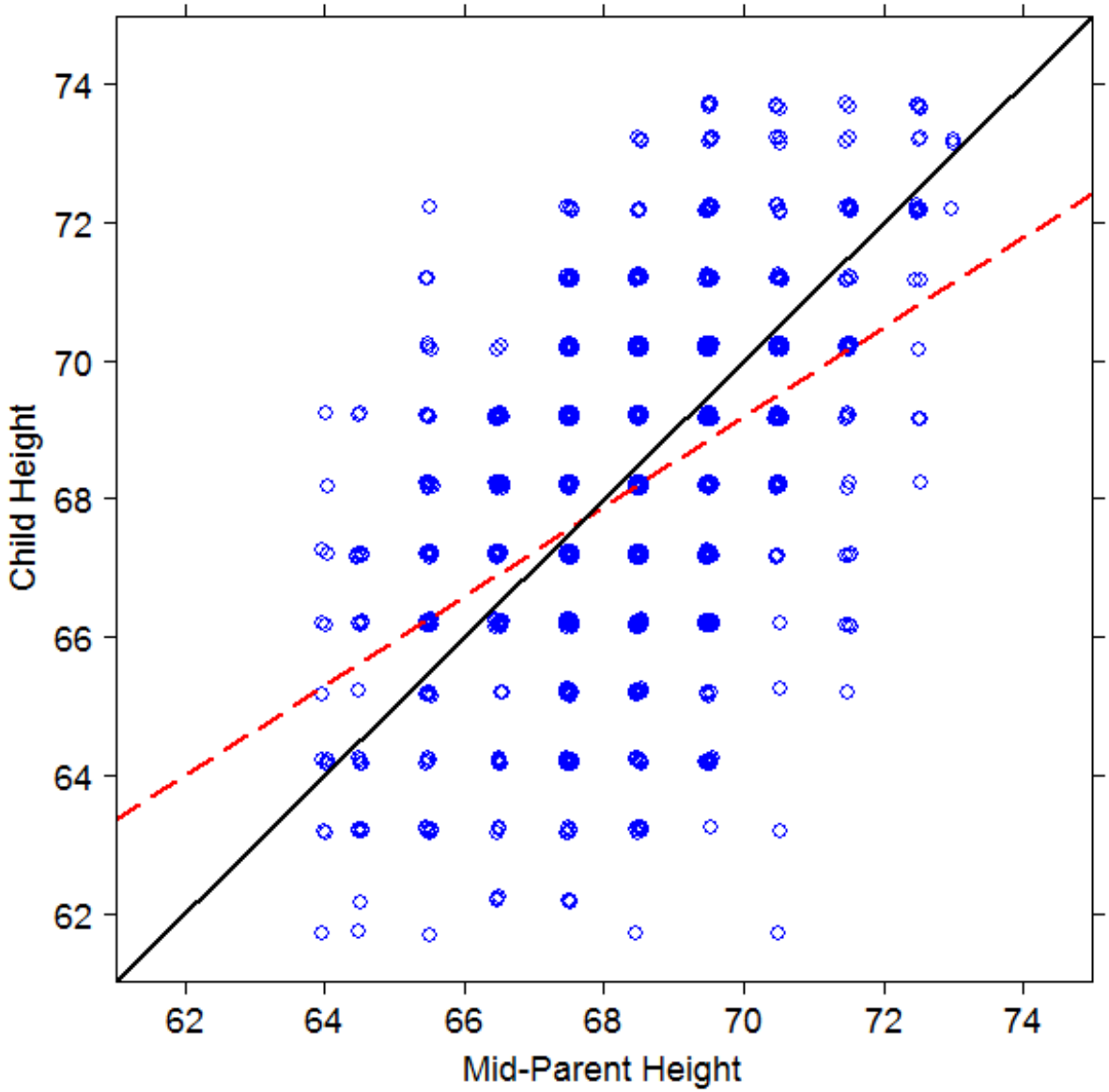
Sir Francis Galton, 1822-1911

Regression Towards Mediocrity in Hereditary Stature

Journal of the Anthropological Institute, 1886; 15:246-63



https://en.wikipedia.org/wiki/Francis_Galton



https://en.wikipedia.org/wiki/Francis_Galton

Regression Analysis

- A linear Model is a sum of weighted variables that predict a target output value given an input data instance

Example: Predicting housing prices

House features: taxes paid per year (X_{tax}), age in years (X_{age})

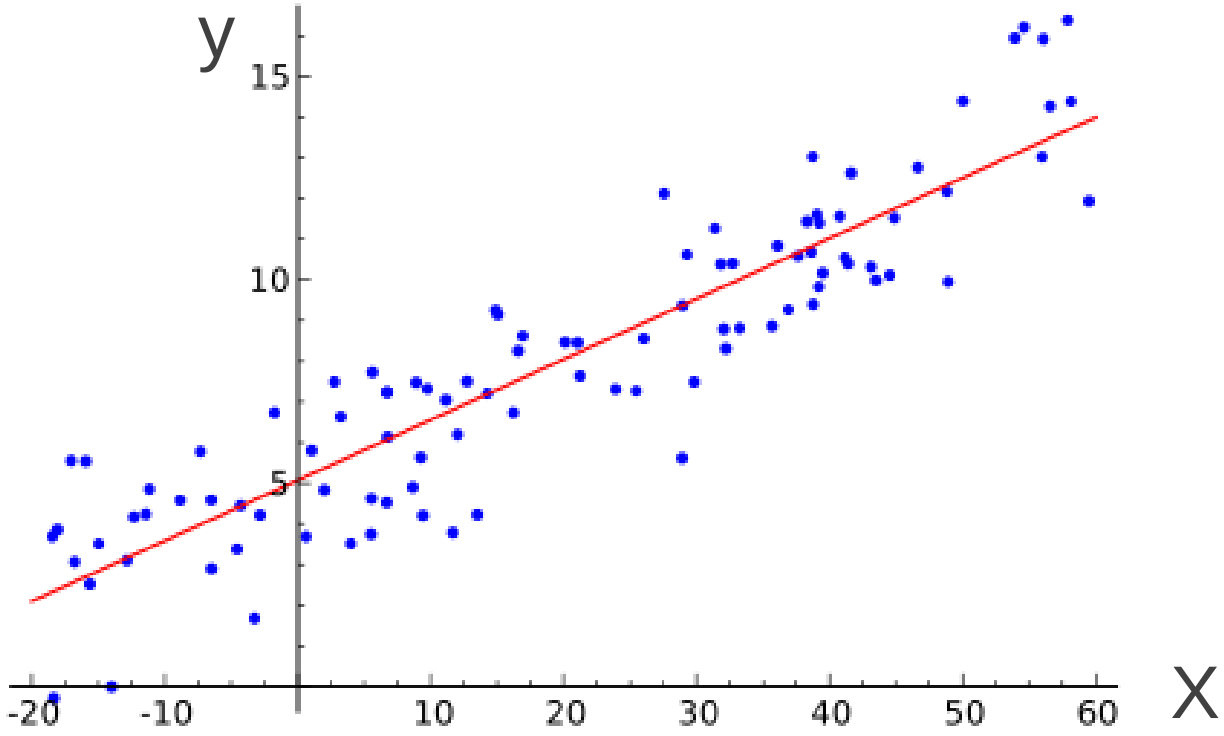
Predicted price= 143000+ 100 X_{tax} – 4000 X_{age}

- So if the house tax per year is 20000, and the age of the house is 60 years then the predicted selling price is:

Predicted price=80000+100 x 20000 – 4000 x 60= 1,840,000

Linear Regression

We want to find the “best” line (linear function y=f(X)) to explain the data.



Linear Regression

The predicted value of y is given by:

$$\hat{y} = \hat{\beta}_0 + \sum_{j=1}^{p} X_j \hat{\beta}_j$$

The vector of coefficients $\hat{\beta}$ is the regression model.

Linear Regression

The regression formula $\hat{y} = \hat{\beta}_0 + \sum_{j=1}^{p} X_j \hat{\beta}_j + \varepsilon$

e.g., j = 1

$$\hat{y} = \hat{\beta}_0 + X_1 \hat{\beta}_1 + \varepsilon$$

Random error

predictor

Slope of the line

Intercept (where the line crosses y-axis)

The slope and intercept of the line are called regression coefficients, model parameters

Our goal is to estimate the model parameters

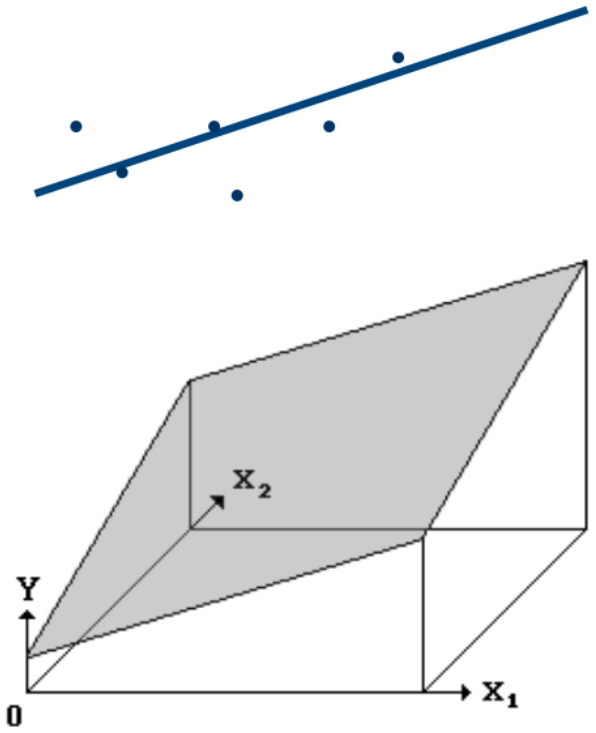
Linear Regression

Simple linear regression

$$Y = \beta 0 + \beta 1X1 + \varepsilon$$

Multiple linear regression

$$Y = \beta 0 + \beta 1X1 + \beta 2X2 + \varepsilon$$



Challenge

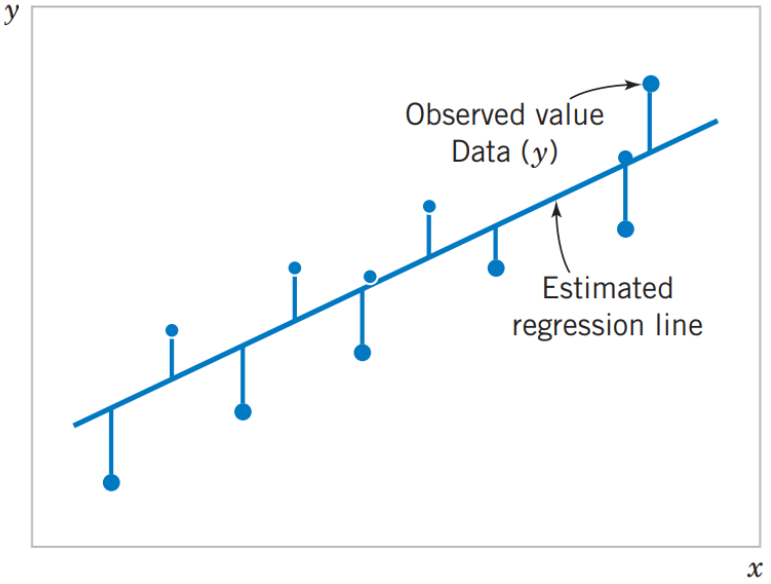
- Find the values of β0 and β1 that the line corresponding to those values is the best fitting line or gives the minimum error (minimum cost)
- Possible solution is to use the Least Square Error solution
- But where do we start and how we determine the proposed line? Gradient descent

Least Square Error Solution

$$y_i = \beta_0 + \beta_1 x_i + \epsilon_i, \qquad i = 1, 2, \ldots, n$$

To estimate (β0,β1) , we find values that minimize squared error

$$L = \sum_{i=1}^{n} \epsilon_i^2 = \sum_{i=1}^{n} (y_i - \beta_0 - \beta_1 x_i)^2$$



Solution:

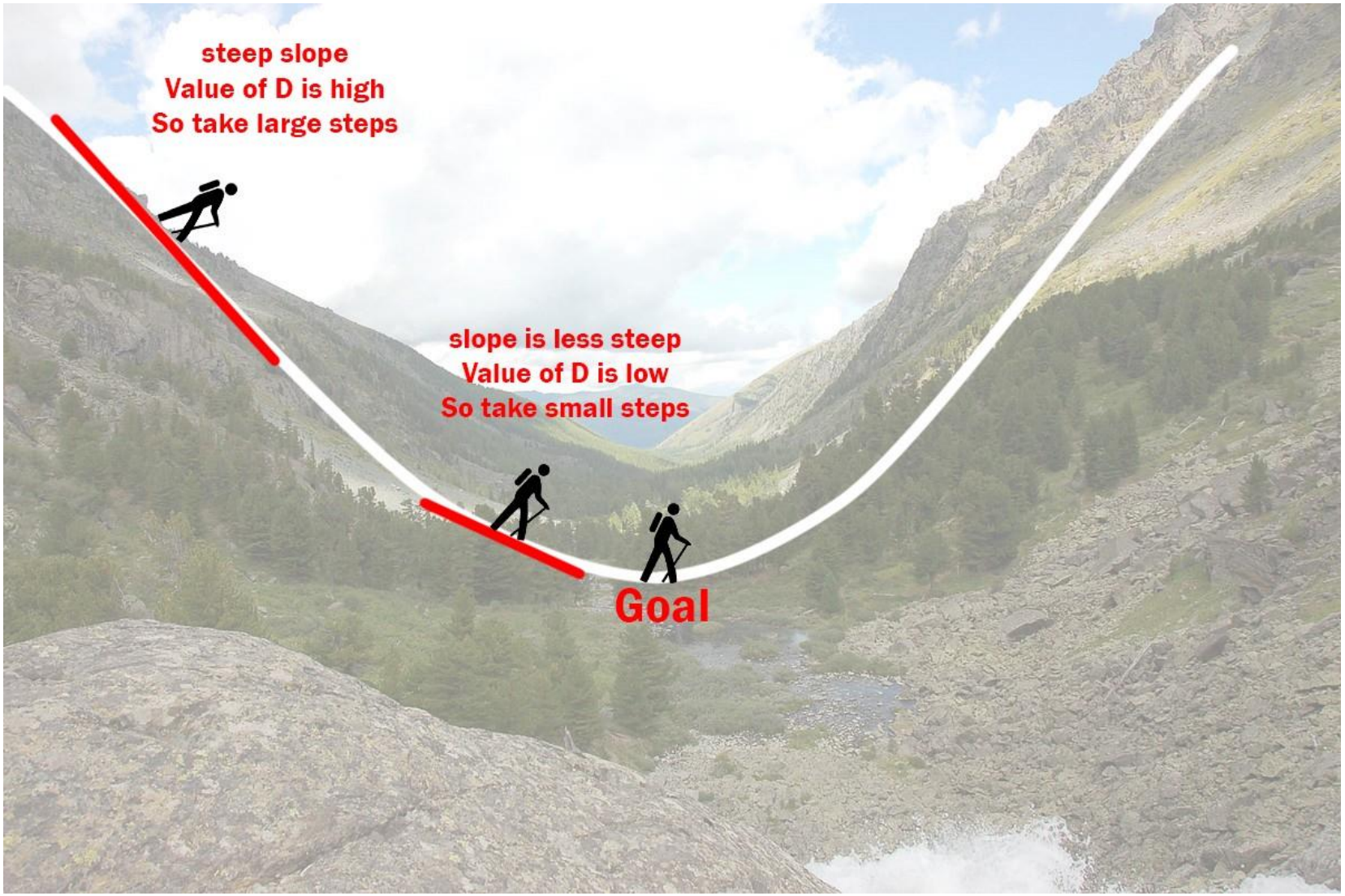
$$\left.\frac{\partial L}{\partial \beta_0}\right|_{\hat{\beta}_0,\hat{\beta}_1} = -2\sum_{i=1}^{n} (y_i - \hat{\beta}_0 - \hat{\beta}_1 x_i) = 0$$

$$\left.\frac{\partial L}{\partial \beta_1}\right|_{\hat{\beta}_0,\hat{\beta}_1} = -2\sum_{i=1}^{n} (y_i - \hat{\beta}_0 - \hat{\beta}_1 x_i) x_i = 0$$

$$n\hat{\beta}_0 + \hat{\beta}_1 \sum_{i=1}^{n} x_i = \sum_{i=1}^{n} y_i$$

$$\hat{\beta}_0 \sum_{i=1}^{n} x_i + \hat{\beta}_1 \sum_{i=1}^{n} x_i^2 = \sum_{i=1}^{n} y_i x_i$$

What is Gradient Descent?



Gradient Descent

- Gradient descent is a method of updating β0 and β1 to reduce the cost function(Least Square Error).
- The idea is that we start with some values for β0 and β1 and then we change these values iteratively to reduce the cost. Gradient descent helps us on how to change the values.
- To update β0 and β1 , we take gradients from the cost function. To find these gradients, we take partial derivatives with respect to β0 and β1.

Multiple Linear Regression

- The multiple linear regression explains the relationship between **one continuous dependent variable** (*y*) and **two or more independent variables** (β1, β2, β3… etc.)
- Challenge: How to determine which features to keep and with to toss?
  - **Forward Selection**
  - **Backward Elimination**
  - **Bidirectional Elimination**

Correlation and Collinearity

- Checking for collinearity helps you get rid of variables that are skewing your data by having a **significant relationship** with another variable
- **Correlation** between variables describe the **relationship** between two variables. If they are **extremely correlated**, then they are **collinear**
- Having high collinearity (correlation of 1.00) between predictors will affect your coefficients and the accuracy, plus its ability to reduce the LSE (Least Squared Errors)
- The simplest method to detect collinearity would be to plot it out in graphs or to view a correlation matrix to check out pairwise correlation.

Questions?

PROJECT

What is the Project About?

- In this assignment, you are asked to **develop** a Data Analytics Service.
- Besides a few requirements that you must meet, the specification of the application you'd build is **deliberately left open**.
- You are expected to **develop a plan** and **execute it** with your **group members** all throughout this assignment period.

Mentoring Meetings

You can arrange meetings with the tutors from Week 9. Before the final demo, you are required to have two meetings. Each meeting will be assessed. During the meetings, your mentor (tutor) will give feedback on your work in progress. So utilise the time as much as you can.

- Meeting One - more or so complete design documentation (10% of the total mark).
- Meeting Two - an early implementation of the Service, demo of work in progress (10% of the total mark).

What you need to do

- Come up with a scenario
- Select the Data sources that would help fulfill your scenario
- Perform Data Integration and pre-processing if needed .
- Building a machine learning model to fulfill the scenario (using the dataset that you have prepared for training and evaluation)
- Designing a RESTful API to allow the consumption of your service (you need to consider an authentication scheme for the consumers of your service)
- Your API must also have some analytic methods representing the API usage, and general information about the use of the API by API users (e.g., number of API calls in the last 24 hours).
- Designing a Simple Client with GUI (you are free to use whatever you like whether it is a simple HTML- java-script , ASP, php, JSP, JSF, or even window-based interfaces).
- Provide documentation for your service (swagger doc and deployment instructions)

How To Start

- Asking the right questions to start the project
- Scope the project, use brainstorming to identify problem statements
  - There are no bad ideas (exploration phase)
  - One conversation at time
  - Consider different points of view
  - Consider 2 stage: (i) harvesting, (ii) decisions
  - Encourage wild and crazy ideas
  - Capture ideas in writing (useful for post-workshop analysis)
  - Avoid tyranny of the pen (no control of the flow of ideas)

Service Examples

- Property Price Prediction
- White Hackers Expertise prediction
- Movies Box Office revenue prediction

How Will We be Marked for this Project

- Mentoring Meeting One - 10%
- Mentoring Meeting Two - 10%
- Week 10 - 70% (demo), 10% (group work)