# Week 1

## Welcome

* Examples
* Ai and neural networks.

## Introduction

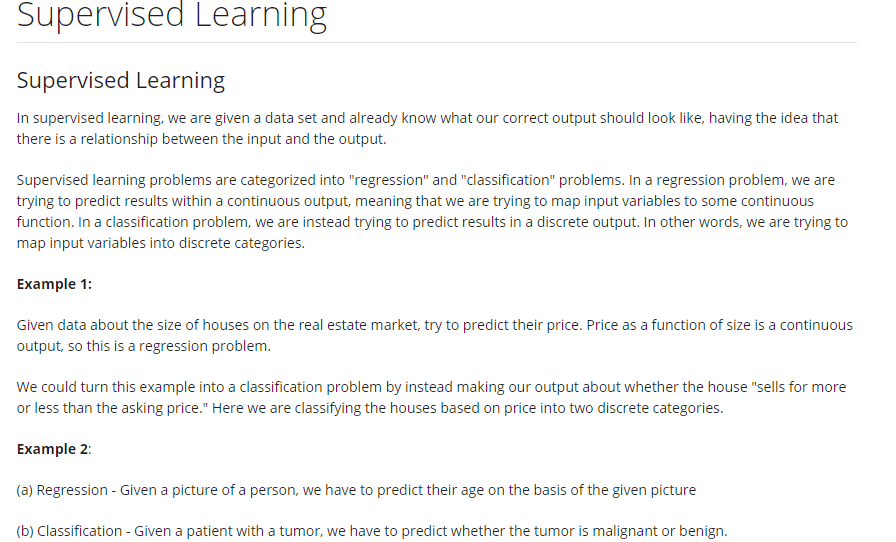
* Examples
* Web search - search engines (Google, Bing): rank web pages
* Photo tagging (FB, Apple’s photo typing app): recognize photos
* Email: filter spam
* In this class
* AI
* Knowledge of algorithms and math + specific implementation
* More examples
* Wide range: autonomous robotics, computational biology
* Database mining: web and automation growth – web click data/clickstream data
* Medical records – turn to knowledge
* Computational biology: automatically collect data about gene sequences, DNA sequences; human genome
* Autonomous helicopters: *only way was get computer to learn by itself how to fly a helicopter*
* Handwriting recognition, natural language processing and computer vision – applied machine learning
* Self-customizing programs: Amazon, Netflix, iTunes Genius
* Understanding human learning
* Opportunities
* Top twelve IT skills
* Unfulfilled demand

What is ML?

* Arthur Samuel: old, informal
* the field of study that gives computers the ability to learn without being explicitly programmed.
* Tom Mitchell: more modern
* A computer program is said to learn from experience E with respect to some class of tasks T and performance measure P, if its performance at tasks T, as measured by P, improves with experience E.
* Example: checkers
* E, T, P
* ML classification
* Supervised
* Unsupervised

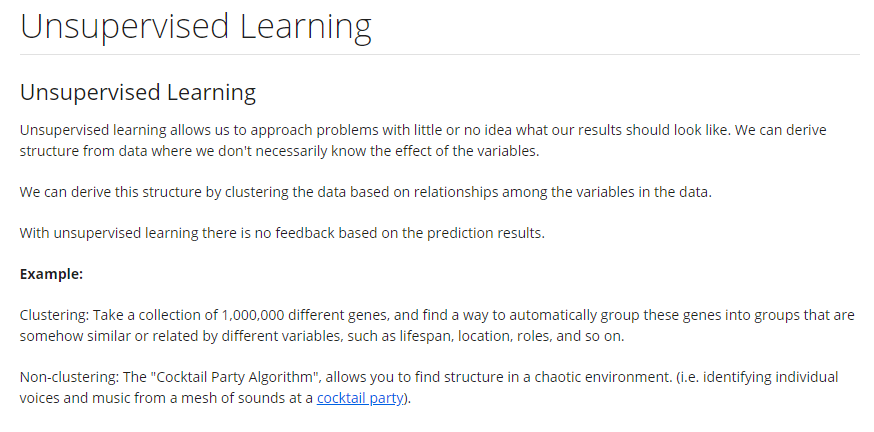
Supervised Learning

* In supervised learning, we are given a data set and already know what our correct output should look like, having the idea that there is a relationship between the input and the output.
* Problem categories
* Regression: use with continuous/real values
* Classification: use with discrete values



Unsupervised Learning

* Rather than being given the “right answer,” we examine data with the same label or no labels at all.
* Find some structure in the data.
* Clustering algorithm
* Google News
* DNA microarray data
* Organizing computer clusters
* Social network analysis
* Market segmentation
* Astronomical data analysis
* Cocktail Party Problem
* Octave programming environment
* SVD (singular value decomposition)



Who Are Mentors?

Get to Know Your Classmates

Frequently Asked Questions

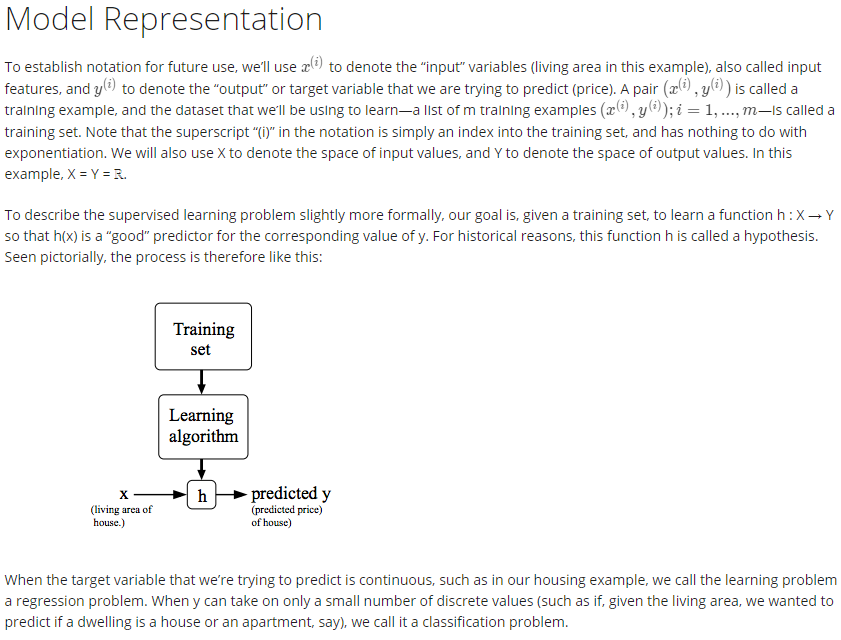
## Review

* Lecture Slides

## Model and Cost Function

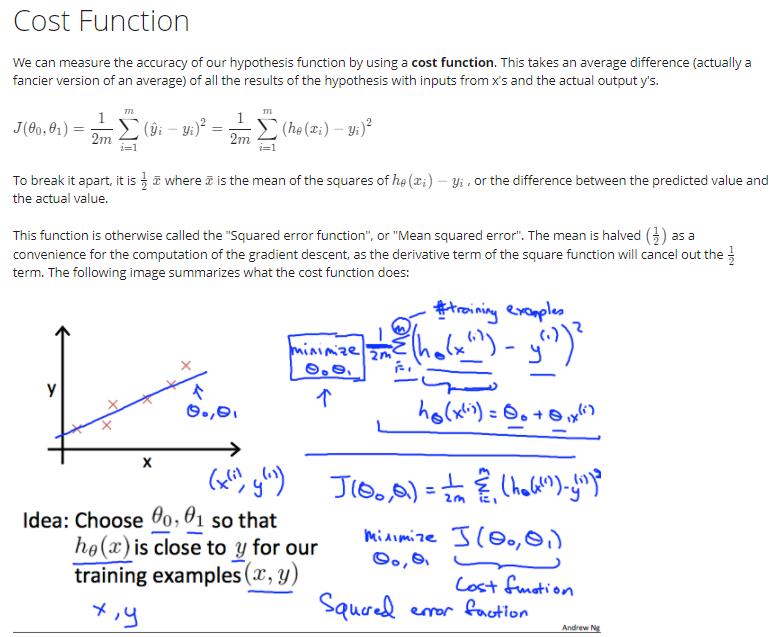
*Model Representation*

* Notation
* m: number of training examples
* x: input variable/feature
* y: output/target variable
* (x, y): one training example
* (x(i), y(i)): ith training example
* h (hypothesis): function – conventional name for mapping from x’s to y’s
* how to represent *h*
* hθ(x) = θ0 + θ1x -> h(x) for shorthand
* (univariate linear regression)

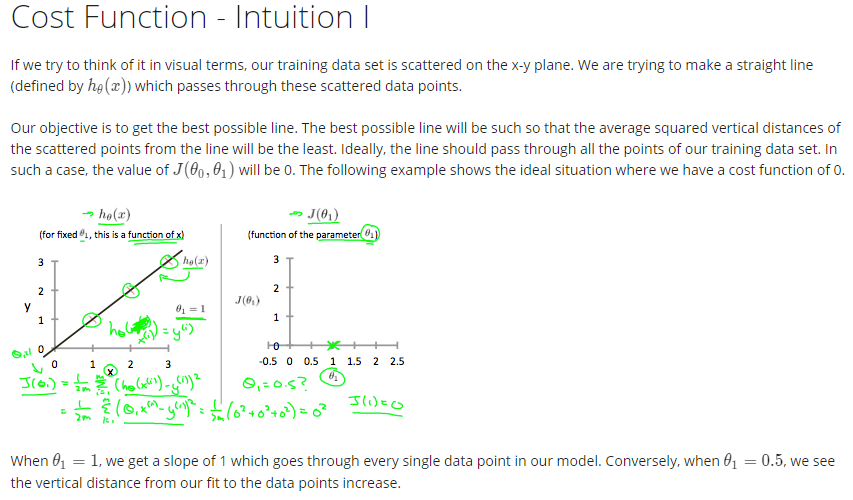


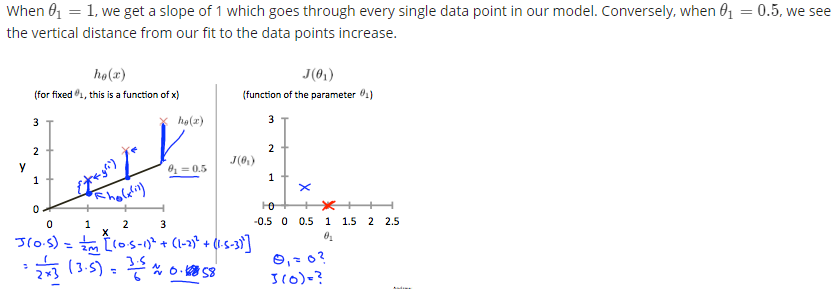
*Cost Function*

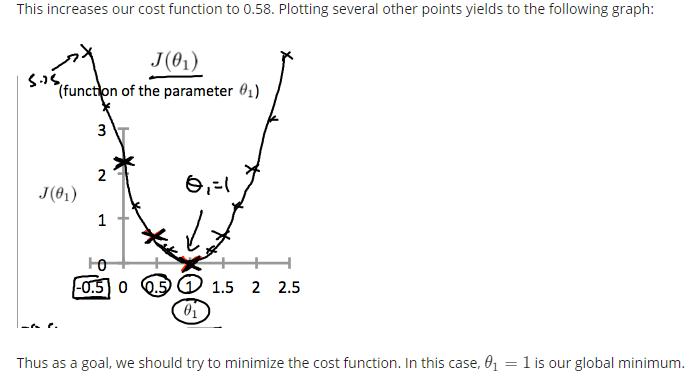
* Choosing parameters θ0, θ1
* Minimization:
* Cost function = squared error function *(most common)*



*Cost Function Intuition I*

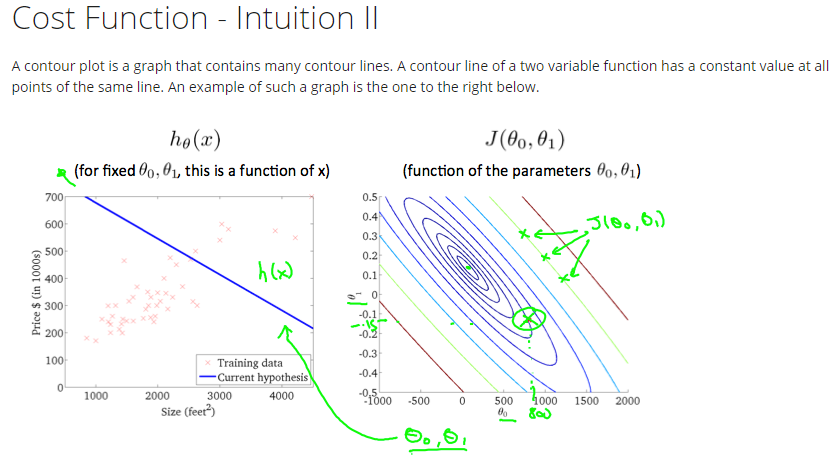


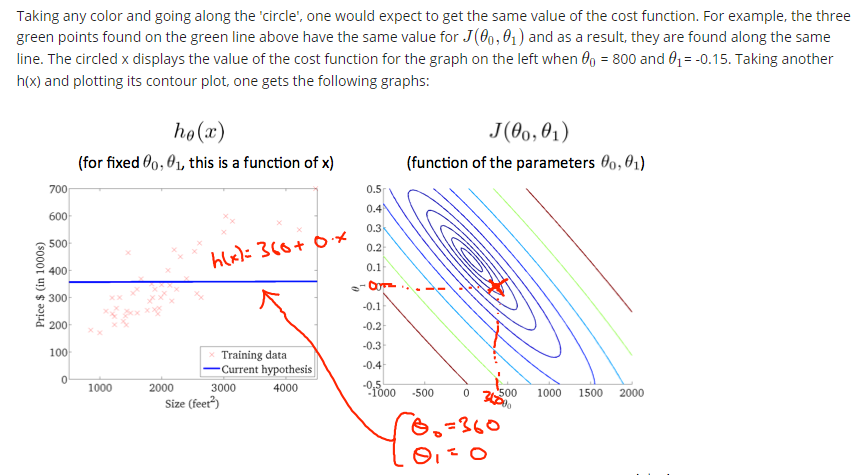


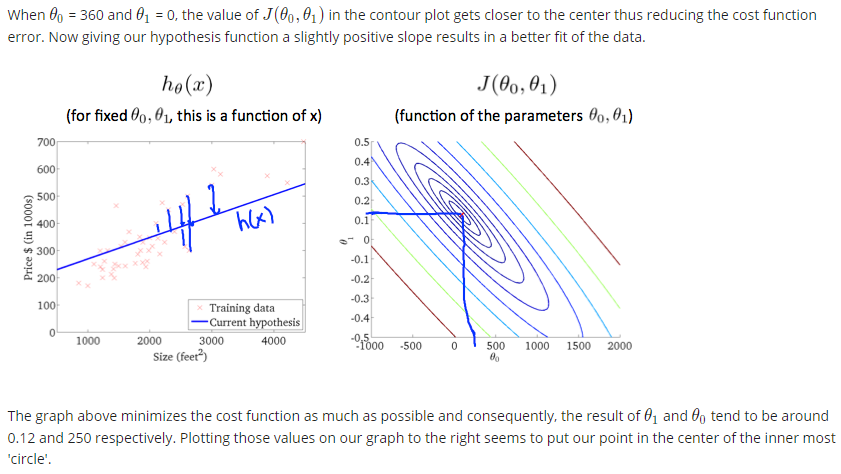


*Cost Function Intuition II*

* Contour plot: slice of 3D surface plot of multi-variable function



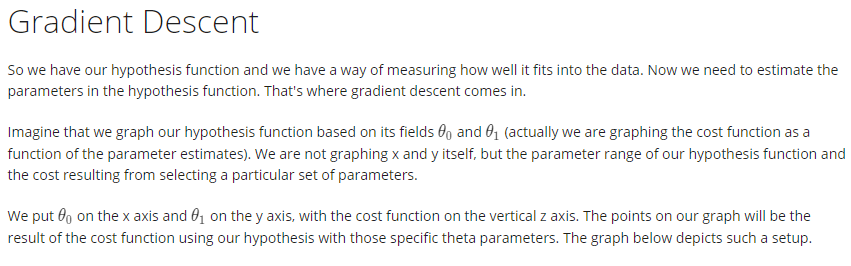


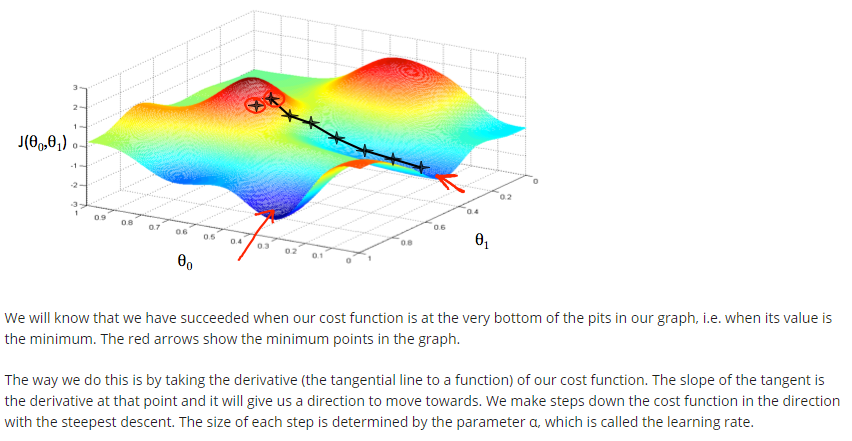


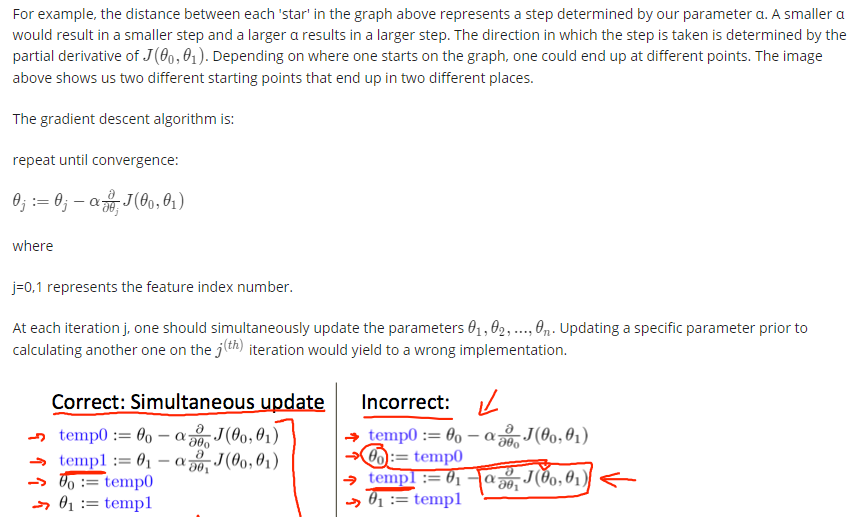
## Parameter Learning

Gradient Descent

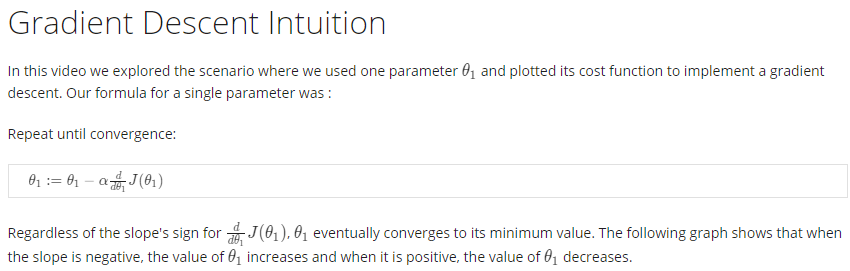
* Algorithm: update parameters until convergence
* Assignment vs. assertion
* Learning rate (α): how big a step to take downhill
* Later: descent property of end point depending on initial decision (choice of first set of parameters); how aggressive a gradient descent procedure is (how to set α); partial derivative term
* Update parameters simultaneously

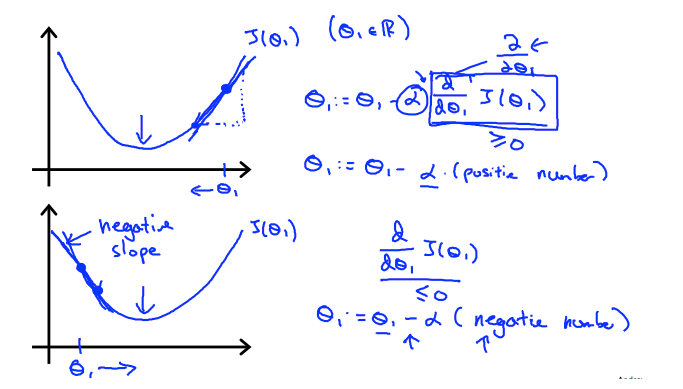


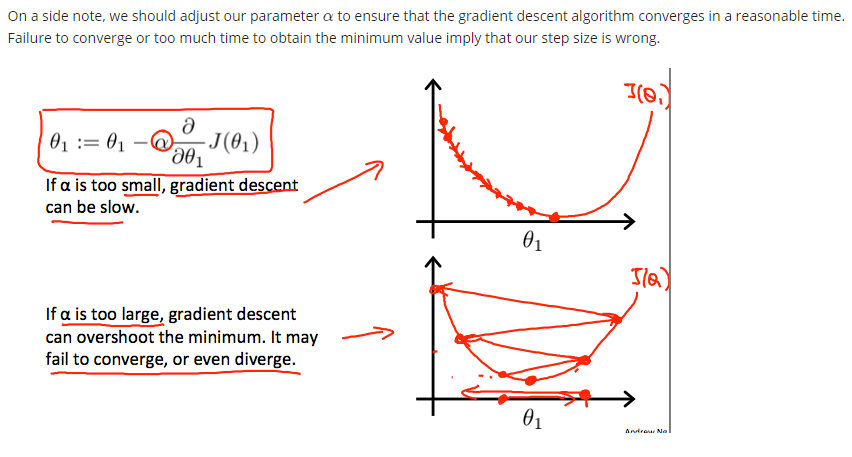


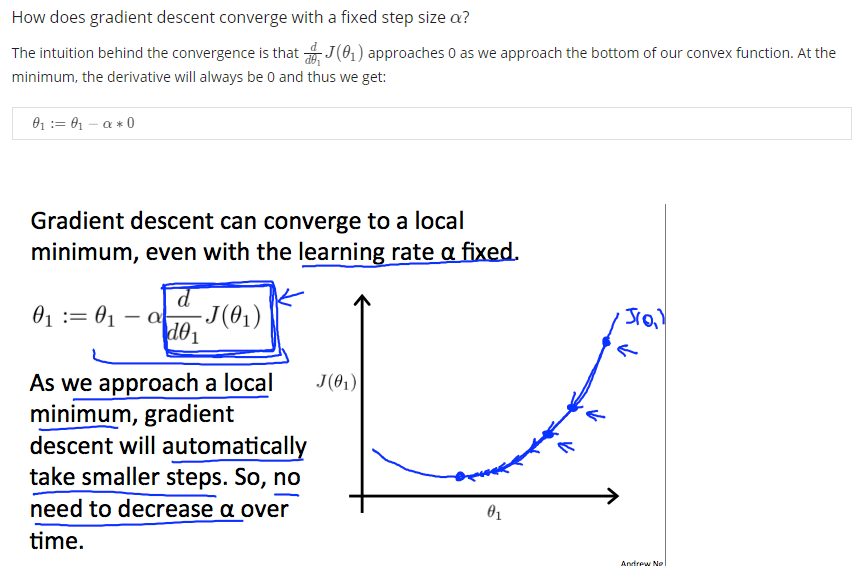


Gradient Descent Intuition





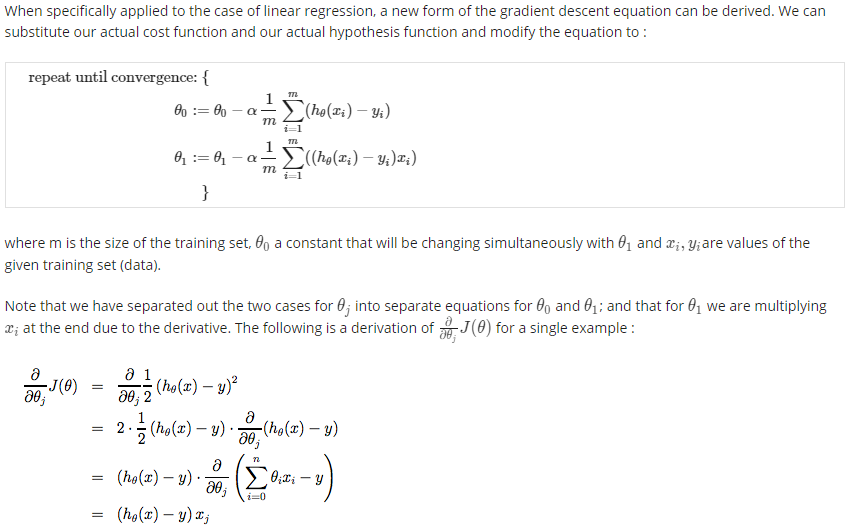


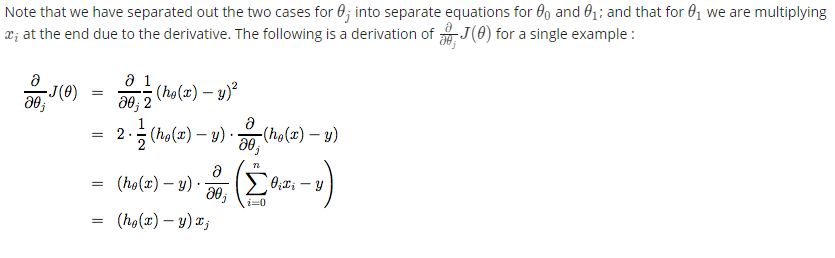
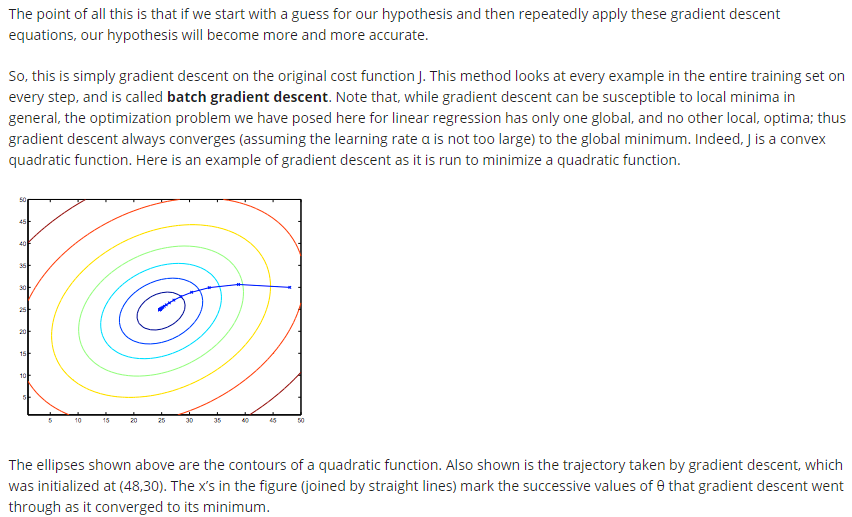


Gradient Descent for Linear Regression

* “Batch” gradient descent
* Normal equation method: analytical (linear algebra) approach

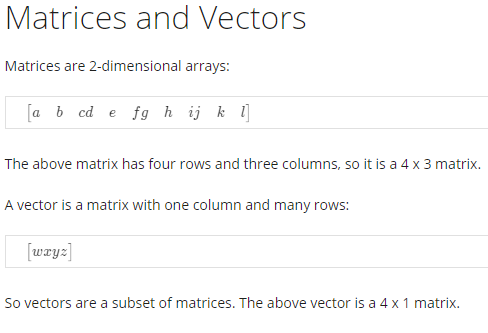


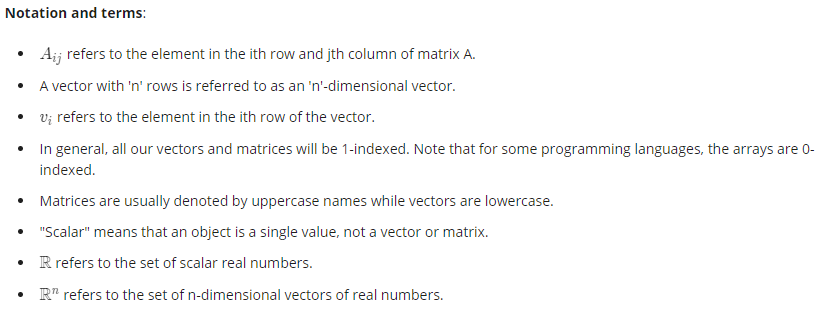


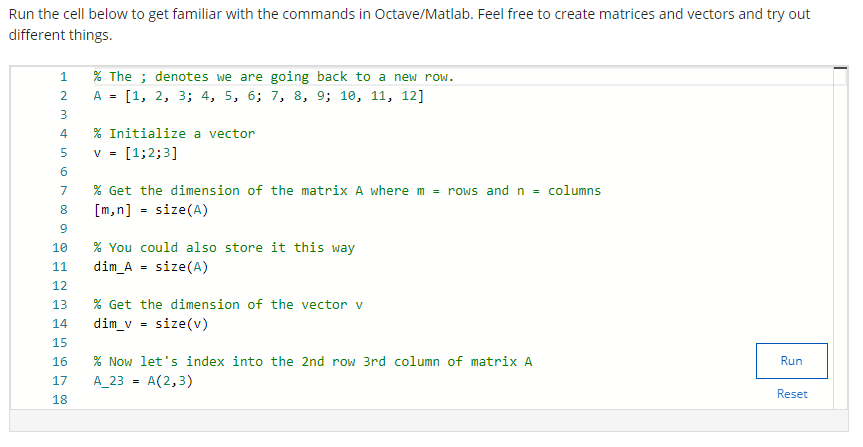


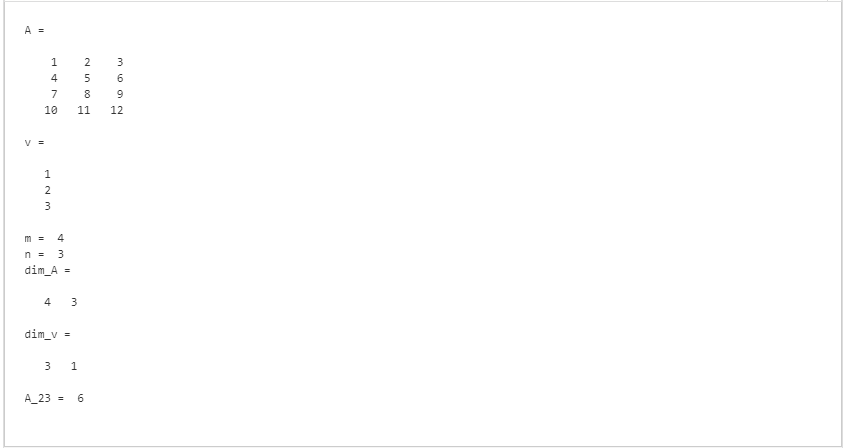
## Linear Algebra Review

Matrices and Vectors

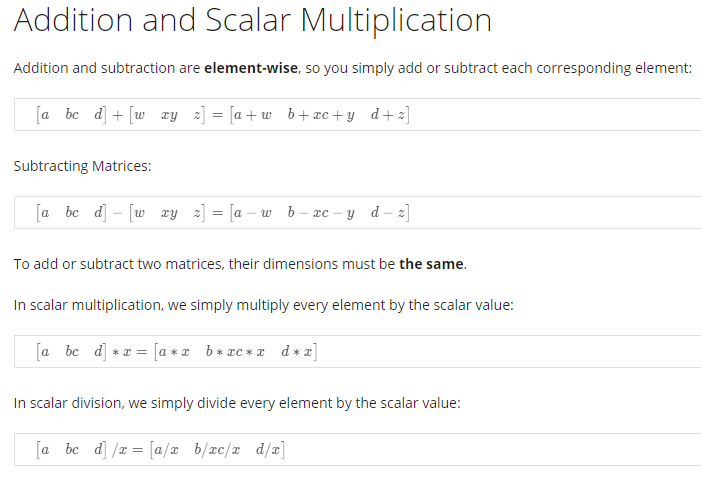


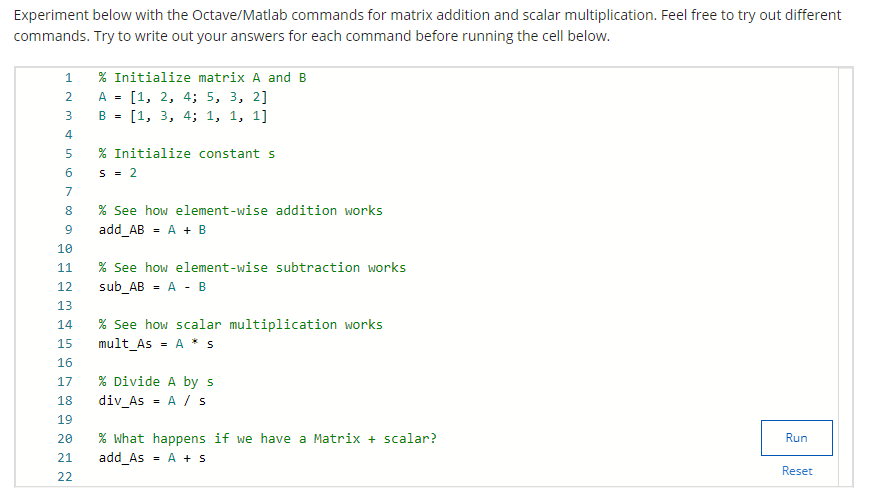






Addition and Scalar Multiplication

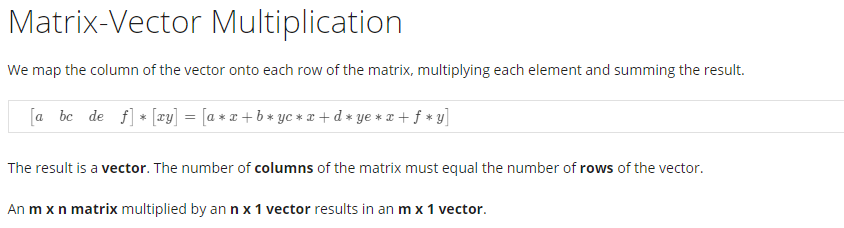


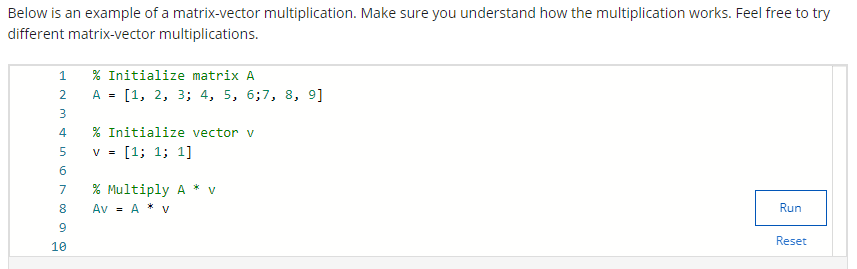


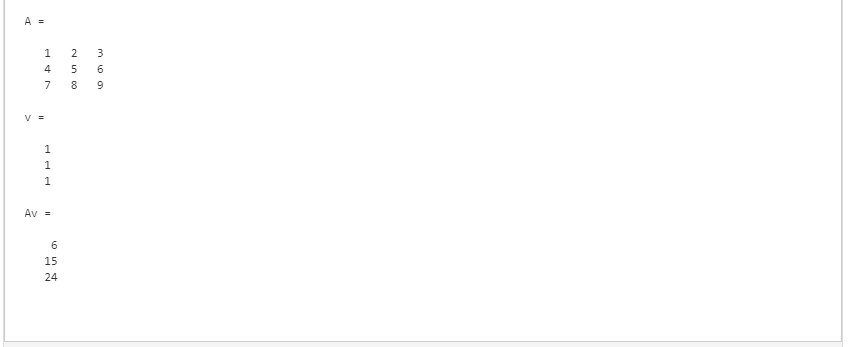


?????

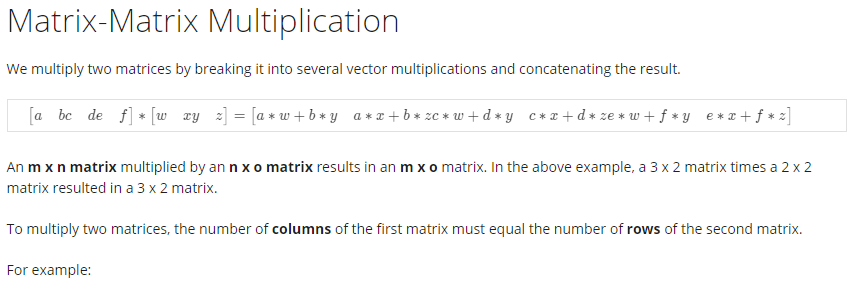
Matrix-Vector Multiplication

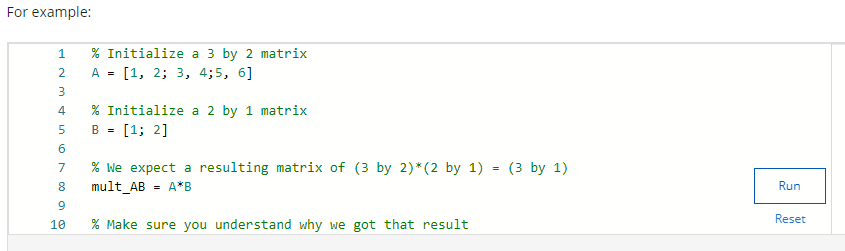






Matrix-Matrix Multiplication

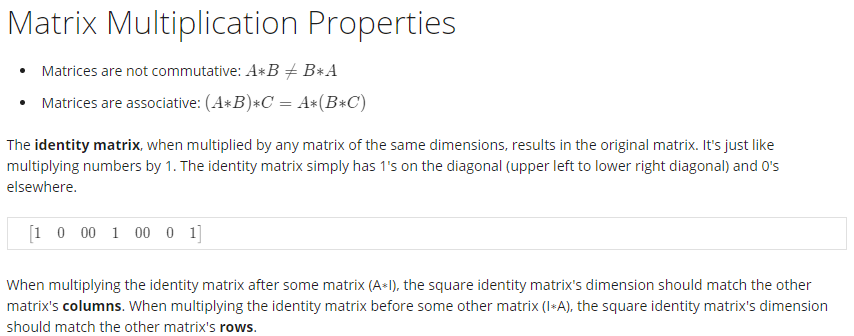


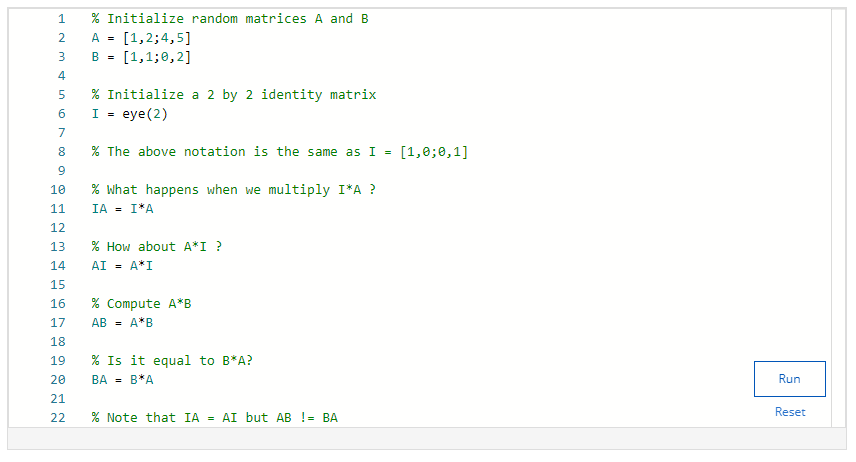




Matrix Multiplication Properties

* Not commutative
* Associative
* Identity matrix (In x n)







Inverse and Transpose

* Singular/degenerate matrix: non-square, zero

