Lesson 07
Notes on Chapter 4
Gradient Descent

CSC357 Advanced Topics—Machine Learning

24 January 2020

- Gradient Descent is very different way to train a Linear Regression model
- better suited for cases where there are...
  - a large number of features
  - or too many training instances to fit in memory
- a generic optimization algorithm
- capable of finding optimal solutions to wide range of problems
- tweak parameters iteratively in order to minimize a cost function
- suppose you are lost in mountains in dense fog
- can only feel the slope of the ground below your feet
- good strategy: get to bottom of valley quickly by going downhill in direction of steepest slope
- measures local gradient of error function with respect to parameter vector θ
- goes in direction of descending gradient
- once the gradient is zero, you have reached a minimum!
- start by filling θ with random values (*random initialization*)
- then, small steps
- at each step, try to decrease cost (MSE)
- learning step size proportional to slope of cost function

- steps gradually get smaller
- size of steps is important parameter (learning rate hyperparameter)
- steps too small → many iterations / slow convergence to solution
- learning rate too high (steps too large) → might "jump across valley"
- "jump across valley" ≡ miss point at which cost is minimum
- steps too large → possibility of divergence (cost increases)
- there may be many local minima—we want global minima
- fortunately, MSE cost function for Linear Regression is convex
  - pick any two points on curve → line segment joining them never crosses curve
  - no local minima, just one global minimum
  - a continuous function with slope that never changes abruptly
  - Gradient Descent guaranteed to approach arbitrarily close to global minimum!
  - (if we wait long enough and learning rate not too high)
- cost function has shape of a bowl,
  - but can be an elongated bowl if features have very different scales
  - rate of convergence can depend upon starting point
  - avoid this by ensuring all features have similar scale
  - (for example, by using Scikit-Learns StandardScaler class)
- training means searching for combination of model parameters that minimizes a cost function (over training set)
- a search in the model's parameter space
- more parameters → more dimensions
- more parameters → harder search