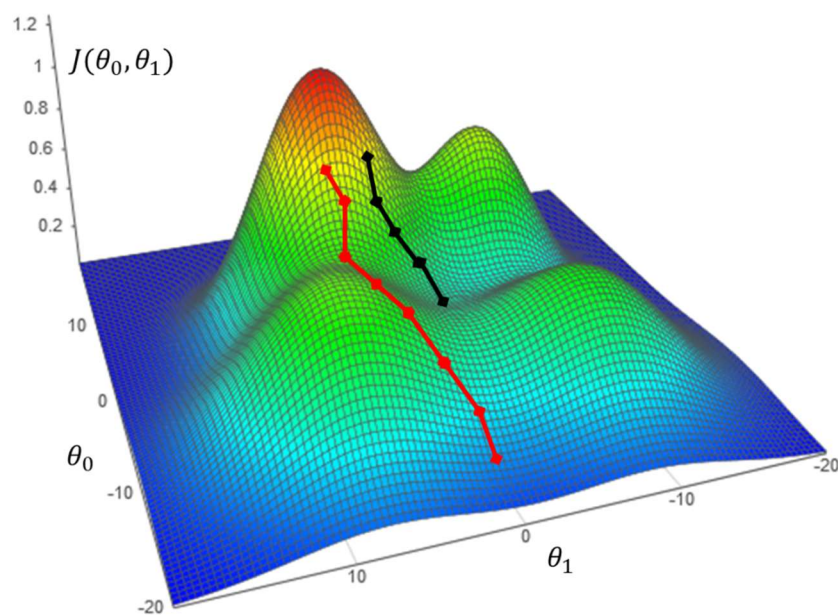


# APPLIED MACHINE LEARNING

---

## *An Introduction*

Kishan Jainandunsing, PhD





## Table of Contents

|       |                                                         |    |
|-------|---------------------------------------------------------|----|
| 1     | Preface .....                                           | 15 |
| 2     | Math Refresher .....                                    | 18 |
| 2.1   | Linear Algebra Refresher .....                          | 18 |
| 2.2   | Calculus Refresher.....                                 | 20 |
| 3     | Linear Regression .....                                 | 22 |
| 3.1   | Single Variable Linear Regression .....                 | 22 |
| 3.1.1 | Model Representation .....                              | 22 |
| 3.1.2 | The Cost Function .....                                 | 24 |
| 3.1.3 | Gradient Descent .....                                  | 27 |
| 3.2   | Multivariate Linear Regression .....                    | 32 |
| 3.2.1 | Multivariate Hypothesis.....                            | 33 |
| 3.2.2 | Multivariate Cost Function.....                         | 35 |
| 3.2.3 | Multivariate Gradient Descent.....                      | 35 |
| 3.2.4 | Direct Solution using the Normal Equation .....         | 36 |
| 3.2.5 | Direct Solution vs. Linear Regression.....              | 38 |
| 4     | Logistic Regression .....                               | 39 |
| 4.1   | Logistic Regression – Sigmoid Hypothesis Function ..... | 40 |
| 4.1.1 | Hypothesis Interpretation.....                          | 41 |
| 4.2   | Logistic Regression – Decision Boundaries .....         | 42 |
| 4.2.1 | Linear Decision Boundaries.....                         | 42 |
| 4.2.2 | Non-Linear Decision Boundaries.....                     | 43 |
| 4.3   | Logistic Regression – Cost Function .....               | 44 |
| 4.4   | Multi-Class Classification .....                        | 47 |
| 5     | Regularization .....                                    | 49 |

|       |                                                                          |    |
|-------|--------------------------------------------------------------------------|----|
| 5.1   | Under and Over-Fitting .....                                             | 49 |
| 5.2   | Regularized Cost Function.....                                           | 50 |
| 5.3   | Regularized Linear Gradient Descent.....                                 | 52 |
| 5.4   | Regularized Normal Equation .....                                        | 53 |
| 5.5   | Regularized Logistic Regression .....                                    | 53 |
| 5.5.1 | Regularized Logistic Regression Cost Function and Gradient Descent.....  | 54 |
| 6     | Neural Networks .....                                                    | 55 |
| 6.1   | Non-Linear Hypotheses.....                                               | 55 |
| 6.2   | Neural Network Model Representation .....                                | 57 |
| 6.3   | Neural Network Self-Learning Capacity .....                              | 59 |
| 6.4   | Neural Network Boolean Algebra .....                                     | 60 |
| 6.4.1 | Non-linear Classification Example: XOR/XNOR.....                         | 60 |
| 6.5   | Multi-Class Classification Neural Networks .....                         | 64 |
| 6.6   | Two Layer Neural Network Binary Logistic Regression.....                 | 65 |
| 6.6.1 | Scalar Representation of the Cost Function, Gradient and Prediction..... | 65 |
| 6.6.2 | Vector Representation of the Cost Function, Gradient and Prediction..... | 66 |
| 6.7   | Multi-Layer, Multi-Class Logistic Regression .....                       | 66 |
| 6.7.1 | Scalar Representation of the Cost Function, Gradient and Prediction..... | 67 |
| 6.7.2 | Vector Representation of the Cost Function, Gradient and Prediction..... | 68 |
| 6.8   | Backpropagation .....                                                    | 69 |
| 6.9   | The Backpropagation Algorithm .....                                      | 73 |
| 6.10  | Gradient Checking.....                                                   | 74 |
| 6.11  | Random Initialization .....                                              | 75 |
| 6.12  | Neural Network Architecture Selection.....                               | 77 |
| 7     | Learning System Tuning Strategies .....                                  | 79 |

|       |                                                           |     |
|-------|-----------------------------------------------------------|-----|
| 7.1   | Hypothesis Evaluation.....                                | 79  |
| 7.1.1 | Hypothesis Evaluation.....                                | 80  |
| 7.1.2 | Model Selection .....                                     | 82  |
| 7.2   | Diagnosing Machine Learning Systems.....                  | 85  |
| 7.2.1 | Bias and Variance .....                                   | 85  |
| 7.2.2 | Regularization Effects on Bias and Variance .....         | 87  |
| 7.2.3 | Learning Curves.....                                      | 89  |
| 7.2.4 | Summarizing.....                                          | 92  |
| 8     | Machine Learning System Design .....                      | 93  |
| 8.1   | Issue No. 1: Prioritization.....                          | 93  |
| 8.2   | Error Analysis .....                                      | 95  |
| 8.3   | Error Metrics for Skewed Classes.....                     | 96  |
| 8.4   | Trading Off Precision and Recall .....                    | 97  |
| 8.5   | Trading-Off Precision and Recall.....                     | 98  |
| 8.6   | Training Data Collection.....                             | 100 |
| 9     | Support Vector Machines .....                             | 102 |
| 9.1   | Large Margin Classifiers .....                            | 104 |
| 9.2   | Mathematical Foundation of Large Margin Classifiers ..... | 106 |
| 9.3   | Kernels .....                                             | 108 |
| 9.4   | Choosing Landmarks .....                                  | 111 |
| 9.5   | Choice of Kernels and Mercer’s Theorem.....               | 113 |
| 9.6   | Multi-Class Classification SVMs.....                      | 114 |
| 9.7   | Logistic Regression vs. SVMs.....                         | 114 |
| 10    | Clustering – Unsupervised Learning .....                  | 116 |
| 10.1  | Clustering versus Classification .....                    | 117 |

|        |                                                                    |     |
|--------|--------------------------------------------------------------------|-----|
| 10.2   | General Principles of Clustering.....                              | 119 |
| 10.2.1 | Distance Measures.....                                             | 120 |
| 10.2.2 | Clustering Algorithms.....                                         | 121 |
| 10.3   | K-Means .....                                                      | 122 |
| 10.3.1 | Seemingly Inseparable Clusters .....                               | 124 |
| 10.3.2 | K-Means Optimization Objective .....                               | 125 |
| 10.3.3 | Random Initialization .....                                        | 126 |
| 10.3.4 | Choosing the Number of Clusters .....                              | 127 |
| 10.4   | Agglomerative Hierarchical Clustering.....                         | 128 |
| 10.4.1 | Graph Distance or Proximity Measures .....                         | 130 |
| 10.4.2 | MIN or Single Link Proximity.....                                  | 130 |
| 10.4.3 | MAX or Complete Link Proximity .....                               | 134 |
| 10.4.4 | Group-Average Proximity.....                                       | 136 |
| 10.4.5 | Centroid Proximity .....                                           | 137 |
| 10.4.6 | Ward’s Method .....                                                | 138 |
| 10.5   | Mean-Shift Clustering .....                                        | 139 |
| 10.5.1 | Formalized Mean-Shift: Density Function, Kernel and Gradient ..... | 140 |
| 10.5.2 | Mean-Shift Properties.....                                         | 141 |
| 10.6   | Expectation Maximization Clustering .....                          | 141 |
| 10.6.1 | A Chicken and Egg Problem .....                                    | 142 |
| 10.6.2 | Iterative Approximation.....                                       | 142 |
| 10.6.3 | Picking the Number of Distributions.....                           | 143 |
| 10.7   | DBSCAN Clustering.....                                             | 143 |
| 10.7.1 | Density Definition .....                                           | 143 |
| 10.7.2 | Core, Border and Noise.....                                        | 144 |

|        |                                                                 |     |
|--------|-----------------------------------------------------------------|-----|
| 10.7.3 | Directly Density-Reachable .....                                | 145 |
| 10.7.4 | Density-Reachable .....                                         | 145 |
| 10.7.5 | Advantages and Disadvantages of DBSCAN.....                     | 146 |
| 10.8   | Self-Organizing Maps .....                                      | 146 |
| 10.8.1 | Dimensionality Reduction .....                                  | 147 |
| 10.8.2 | Intuitive Understanding .....                                   | 149 |
| 10.8.3 | Adjusting Radius and Weight .....                               | 153 |
| 11     | Dimensionality Reduction (Data Compression) .....               | 153 |
| 11.1   | Principal Component Analysis.....                               | 154 |
| 11.2   | The PCA Algorithm .....                                         | 155 |
| 11.3   | PCA Application Advice .....                                    | 158 |
| 12     | Anomaly Detection .....                                         | 160 |
| 12.1   | The Gaussian Distribution Model .....                           | 162 |
| 12.2   | Probability Density Estimation.....                             | 164 |
| 12.3   | Building an Anomaly Detection System .....                      | 165 |
| 12.4   | Anomaly Detection vs. Supervised Learning.....                  | 166 |
| 12.5   | Choosing Features in Anomaly Detection.....                     | 167 |
| 12.6   | Multivariate Gaussian Distribution .....                        | 169 |
| 12.7   | Anomaly Detection using Multivariate Gaussian Distribution..... | 175 |
| 12.8   | Multivariate vs. Univariate Product Gaussian Distribution ..... | 176 |
| 13     | Recommender Systems .....                                       | 178 |
| 13.1   | Problem Formulation .....                                       | 178 |
| 13.2   | Content-Based Recommender Systems.....                          | 179 |
| 13.3   | Collaborative Filtering.....                                    | 180 |
| 13.4   | Finding Similarities .....                                      | 184 |

|        |                                                            |     |
|--------|------------------------------------------------------------|-----|
| 13.5   | New Customer Recommendations – Mean Normalization .....    | 184 |
| 14     | Large Scale Machine Learning.....                          | 186 |
| 14.1   | Stochastic Gradient Descent.....                           | 187 |
| 14.1.1 | Stochastic Gradient Descent Convergence .....              | 191 |
| 14.1.2 | Online Learning .....                                      | 193 |
| 14.2   | Mini-Batch Gradient Descent.....                           | 193 |
| 14.3   | MapReduce and Data Parallelism .....                       | 195 |
| 15     | Application Example – Photo OCR .....                      | 197 |
| 15.1   | Photo OCR Machine Learning Pipeline .....                  | 198 |
| 15.2   | 2D Sliding Windows – Text Detection.....                   | 200 |
| 15.3   | 1D Sliding Windows – Character Segmentation .....          | 203 |
| 15.4   | Training Example Datasets – Artificial Data Synthesis..... | 204 |
| 15.5   | Ceiling Analysis.....                                      | 206 |

## List of Figures

|                                                                                                                 |    |
|-----------------------------------------------------------------------------------------------------------------|----|
| FIGURE 1. THE HOUSE PRICE PREDICTOR.....                                                                        | 23 |
| FIGURE 2. SCATTER PLOT OF EXAMPLE PAIRS $(x(i), y(i))$ .....                                                    | 23 |
| FIGURE 3. ERRORS $\epsilon(i) = h\theta x_i - y_i$ .....                                                        | 24 |
| FIGURE 4. HYPOTHESIS WITH COST OR LSE EQUAL TO 0 FOR THE EXAMPLE SET .....                                      | 25 |
| FIGURE 5. LETTING $\theta_1$ VARY AROUND 1 WHERE THE COST $J\theta = 0$ . .....                                 | 26 |
| FIGURE 6. GRAPH OF $J(\theta_1)$ IN EQUATION (6) AS A FUNCTION OF $\theta_1$ .....                              | 27 |
| FIGURE 7. USING THE PARTIAL DERIVATIVE IN $\theta_1$ TO MINIMIZE $J(\theta_1)$ ITERATIVELY .....                | 28 |
| FIGURE 8. CONVERGENCE TO A LOCAL INSTEAD OF THE GLOBAL MINIMUM .....                                            | 29 |
| FIGURE 9. SHRINKING DERIVATIVE $\partial\theta_j J\theta_j$ AS $\theta_j$ APPROACHES THE MINIMUM .....          | 29 |
| FIGURE 10. VISUALIZATION OF THE GRADIENT DESCENT ALGORITHM IN (17) .....                                        | 31 |
| FIGURE 11. $J(\theta)$ AS A CONCAVE FUNCTION.....                                                               | 31 |
| FIGURE 12. HYPOTHESIS $h\theta(x)$ CHANGES WITH CORRESPONDING GRADIENT DESCENT STEPS FOR COST $J(\theta)$ ..... | 32 |
| FIGURE 13. CLASSIFICATION PROBLEM FOR A TUMOR CLASSIFICATION EXAMPLE.....                                       | 39 |
| FIGURE 14. CONVERSION TO TRUE BINARY CLASSIFICATION .....                                                       | 40 |
| FIGURE 15. GRAPH OF THE SIGMOID FUNCTION.....                                                                   | 41 |
| FIGURE 16. PLOT FOR $y = H(h\theta(x_1, x_2))$ WITH LINEAR BOUNDARY .....                                       | 43 |
| FIGURE 17. PLOT FOR $y = h\theta(x_1, x_2)$ WITH CIRCULAR BOUNDARY .....                                        | 44 |
| FIGURE 18. ARBITRARY, IRREGULAR, NON-LINEAR BOUNDARY .....                                                      | 44 |
| FIGURE 19. COST AS A FUNCTION OF $h\theta(x)$ .....                                                             | 45 |
| FIGURE 20. BINARY VS. MULTI-CLASS CLASSIFICATION.....                                                           | 47 |
| FIGURE 21. ONE-VS-ALL MULTI-CLASS CLASSIFICATION LOGISTIC REGRESSION .....                                      | 48 |
| FIGURE 22. UNDER AND OVER-FITTING IN CASE OF LINEAR REGRESSION.....                                             | 49 |
| FIGURE 23. UNDER AND OVER-FITTING IN CASE OF LOGISTIC REGRESSION .....                                          | 50 |
| FIGURE 24. PICTORIAL REPRESENTATION OF THE EFFECT OF THE REGULARIZATION TERM ON THE COST FUNCTION .....         | 51 |
| FIGURE 25. CHOOSING $\lambda$ TOO LARGE FOR REGULARIZATION LEADS TO UNDER-FITTING OF THE DATA .....             | 52 |
| FIGURE 26. OVER-FITTING AND REGULARIZATION OF LOGISTIC REGRESSION .....                                         | 53 |
| FIGURE 27. COMPUTER VISION REPRESENTATION OF THE IMAGE OF A CAR .....                                           | 55 |
| FIGURE 28. PIXEL-PAIR SIMPLIFICATION OF THE BINARY CLASSIFICATION FOR CARS VS. NON-CARS.....                    | 56 |
| FIGURE 29. ARTIFICIAL NEURON MODEL'S LOGISTIC UNIT.....                                                         | 57 |
| FIGURE 30. MULTI-LAYER ARTIFICIAL NEURAL NETWORK EXAMPLE .....                                                  | 58 |
| FIGURE 31. SELF-LEARNING CAPACITY OF AN ANN VIA EXTRACTION OF NEW FEATURES BY HIDDEN LAYERS.....                | 59 |
| FIGURE 32. THE XNOR FUNCTION REPRESENTED AS A CLASSIFICATION PROBLEM .....                                      | 61 |
| FIGURE 33. COMPLEX DECISION BOUNDARY GENERALIZATION OF THE XNOR EXAMPLE .....                                   | 61 |

|                                                                                                                             |     |
|-----------------------------------------------------------------------------------------------------------------------------|-----|
| FIGURE 34. SIMPLE AND OR OR NEURAL NETWORK FOR BINARY INPUTS.....                                                           | 62  |
| FIGURE 35. THREE-LAYER ANN IMPLEMENTATION OF THE XNOR FUNCTION.....                                                         | 64  |
| FIGURE 36. REPRESENTATION OF A MULTI-CLASS CLASSIFICATION NEURAL NETWORK.....                                               | 65  |
| FIGURE 37. FORWARD PROPAGATION OF ACTIVATIONS.....                                                                          | 70  |
| FIGURE 38. BACKPROPAGATION OF ACTIVATION ERRORS.....                                                                        | 72  |
| FIGURE 39. GRADIENT CHECKING.....                                                                                           | 74  |
| FIGURE 40. NEURAL NETWORK PARAMETER INITIALIZATION.....                                                                     | 75  |
| FIGURE 41. VISUALIZATION OF BACKPROPAGATION.....                                                                            | 76  |
| FIGURE 42. REASONABLE CHOICES OF NEURAL NETWORK ARCHITECTURES WITH ONE OR MORE HIDDEN LAYERS.....                           | 77  |
| FIGURE 43. OVERFITTING HYPOTHESIS PLOT.....                                                                                 | 80  |
| FIGURE 44. EXAMPLES OF HIGH BIAS, JUST RIGHT AND HIGH VARIANCE HYPOTHESES.....                                              | 85  |
| FIGURE 45. TRAINING AND CROSS-VALIDATION DATASET PREDICTION ERRORS VS. MODEL COMPLEXITY.....                                | 86  |
| FIGURE 46. EFFECT OF REGULARIZATION PARAMETER $\lambda$ ON BIAS AND VARIANCE.....                                           | 87  |
| FIGURE 47. EFFECT OF REGULARIZATION TERM $\lambda$ ON BIAS AND VARIANCE.....                                                | 89  |
| FIGURE 48. FITTING A 2 <sup>ND</sup> ORDER POLYNOMIAL HYPOTHESIS TO A TRAINING DATASET OF INCREASING SIZE.....              | 90  |
| FIGURE 49. $J_{train}(\theta)$ AND $J_{cv}(\theta)$ AS A FUNCTION OF TRAINING DATASET $m$ .....                             | 90  |
| FIGURE 50. ASYMPTOTIC $J_{train}(\theta)$ AND $J_{cv}(\theta)$ FOR A GROWING DATASET IN CASE OF A HIGH BIAS HYPOTHESIS..... | 91  |
| FIGURE 51. SLOW INCREASE OF $J_{train}(\theta)$ AND DECREASE OF $J_{cv}(\theta)$ FOR A HIGH VARIANCE HYPOTHESIS.....        | 91  |
| FIGURE 52. TRADING OFF PRECISION AND RECALL.....                                                                            | 99  |
| FIGURE 53. PRECISION/RECALL CURVES.....                                                                                     | 99  |
| FIGURE 54. THE SIGMOID FUNCTION.....                                                                                        | 102 |
| FIGURE 55. COST FUNCTION GRAPHS FOR $y=1$ AND $y=0$ .....                                                                   | 103 |
| FIGURE 56. SVM LARGE MARGIN BOUNDARY.....                                                                                   | 104 |
| FIGURE 57. LARGE MARGIN CLASSIFIER BOUNDARY FOR DATA SETS WITH OUTLIERS.....                                                | 105 |
| FIGURE 58. PROJECTION REPRESENTATION OF THE INNER PRODUCT OF TWO VECTORS.....                                               | 106 |
| FIGURE 59. PROJECTION OF A DATASET ON DIFFERENT CHOICES OF THE PARAMETER VECTOR $\theta$ .....                              | 107 |
| FIGURE 60. DATASET WITH NON-LINEAR DECISION BOUNDARY.....                                                                   | 108 |
| FIGURE 61. LANDMARKS DEFINING THE $similarity(x_i, l_j)$ .....                                                              | 109 |
| FIGURE 62. EFFECT OF DIFFERENT CHOICES FOR THE VARIANCE.....                                                                | 110 |
| FIGURE 63. LANDMARK APPLICATION EXAMPLE.....                                                                                | 110 |
| FIGURE 64. EFFECT OF $\sigma$ ON BIAS AND VARIANCE.....                                                                     | 113 |
| FIGURE 65. LABELED VS. UNLABELED TRAINING DATA.....                                                                         | 116 |
| FIGURE 66: CLASSIFICATION EXAMPLE IN THE MEDICAL FIELD.....                                                                 | 117 |
| FIGURE 67: CLUSTERING EXAMPLE IN THE MEDICAL FIELD.....                                                                     | 118 |

|                                                                                              |     |
|----------------------------------------------------------------------------------------------|-----|
| FIGURE 68: CLASSIFICATION EXAMPLE IN THE ENGINE MANUFACTURING INDUSTRY .....                 | 118 |
| FIGURE 69: CLUSTERING EXAMPLE IN THE ENGINE MANUFACTURING INDUSTRY .....                     | 119 |
| FIGURE 70: EXAMPLE OF SIMILARITY DISTANCES FOR A DATASET .....                               | 119 |
| FIGURE 71. RANDOM CENTROID INITIALIZATION FOR K-MEANS.....                                   | 122 |
| FIGURE 72. ILLUSTRATION OF K-MEANS ITERATIVE STEPS.....                                      | 123 |
| FIGURE 73. CLUSTERING OF SEEMINGLY INSEPARABLE DATASETS.....                                 | 125 |
| FIGURE 74. K-MEANS WITH GLOBAL AND LOCAL MINIMA RESULTS FOR THE SAME EXAMPLE SET.....        | 127 |
| FIGURE 75. THE ELBOW METHOD .....                                                            | 128 |
| FIGURE 76: EXAMPLE OF AN AGGLOMERATIVE HIERARCHICAL CLUSTER TREE FOR FOODS.....              | 129 |
| FIGURE 77: MIN OR SINGLE LINK PROXIMITY DISTANCE BETWEEN CLUSTERS A AND B.....               | 130 |
| FIGURE 78: UNDETECTABLE CLUSTERS USING MIN PROXIMITY .....                                   | 131 |
| FIGURE 79: MIN PROXIMITY EXAMPLE STEP 1.....                                                 | 131 |
| FIGURE 80: MIN PROXIMITY EXAMPLE STEP 2.....                                                 | 132 |
| FIGURE 81: MIN PROXIMITY EXAMPLE STEP 3.....                                                 | 132 |
| FIGURE 82: MIN PROXIMITY EXAMPLE STEP 4.....                                                 | 133 |
| FIGURE 83: MIN PROXIMITY EXAMPLE STEP 5.....                                                 | 133 |
| FIGURE 84: MIN PROXIMITY EXAMPLE FINAL STEP .....                                            | 133 |
| FIGURE 85 MAX OR COMPLETE LINK PROXIMITY DISTANCE BETWEEN CLUSTERS A AND B.....              | 134 |
| FIGURE 86: MAX PROXIMITY EXAMPLE STEP 1 .....                                                | 134 |
| FIGURE 87: MAX PROXIMITY EXAMPLE STEP 2 .....                                                | 135 |
| FIGURE 88: MAX PROXIMITY EXAMPLE STEP 3 .....                                                | 135 |
| FIGURE 89: MAX PROXIMITY EXAMPLE STEP 4 .....                                                | 136 |
| FIGURE 90: MAX PROXIMITY EXAMPLE FINAL STEP .....                                            | 136 |
| FIGURE 91: GROUP-AVERAGE PROXIMITY DISTANCE BETWEEN CLUSTERS A AND B.....                    | 137 |
| FIGURE 92: CENTROID PROXIMITY DISTANCE BETWEEN CLUSTERS A AND B .....                        | 137 |
| FIGURE 93: WARD’S METHOD FOR THE DISTANCE BETWEEN CLUSTERS A AND B.....                      | 138 |
| FIGURE 94: WARD’S METHOD INITIAL STEP.....                                                   | 139 |
| FIGURE 95: MEAN-SHIFT GRAVITATIONAL CLUSTERING .....                                         | 139 |
| FIGURE 96: DISPLACEMENT ALONG THE MEAN-SHIFT VECTOR TOWARDS THE CENTER OF MASS .....         | 141 |
| FIGURE 97: PROBABILITY DENSITY FUNCTION DECOMPOSITION INTO GAUSSIAN DISTRIBUTIONS .....      | 142 |
| FIGURE 98: HIGH AND LOW DENSITY NEIGHBORHOODS.....                                           | 144 |
| FIGURE 99: CORE (GREEN), BORDER (ORANGE) AND NOISE (RED) DATA POINTS .....                   | 144 |
| FIGURE 100: DIRECTLY DENSITY REACHABLE DATA POINT (GREEN) FROM CORE DATA POINT (ORANGE)..... | 145 |
| FIGURE 101: DENSITY REACHABLE DATA POINT $P_3$ FROM $P_1$ .....                              | 145 |

|                                                                                                              |     |
|--------------------------------------------------------------------------------------------------------------|-----|
| FIGURE 102: DBSCAN CLUSTERING NOISE FILTERING EXAMPLE.....                                                   | 146 |
| FIGURE 103: DBSCAN HIGH SENSITIVITY TO CLUSTER DENSITY VARIATIONS .....                                      | 146 |
| FIGURE 104: 2-D POSITIVE CORRELATION VISUALIZATION FOR TWO ATTRIBUTES.....                                   | 147 |
| FIGURE 105: ATTRIBUTE HIDDEN FROM A 2-D VISUALIZATION .....                                                  | 147 |
| FIGURE 106: WORST TO BEST HEALTH SCORE COLOR MAP .....                                                       | 148 |
| FIGURE 107: SOM VISUALIZATION OF TABLE 19 .....                                                              | 148 |
| FIGURE 108: SOM VISUALIZATION EXAMPLE .....                                                                  | 149 |
| FIGURE 109: SOM NEURAL NETWORK EQUIVALENT REPRESENTATION .....                                               | 149 |
| FIGURE 110: SOM NODE BEST MATCHING UNIT WITH SMALLEST EUCLIDEAN DISTANCE .....                               | 150 |
| FIGURE 111: SOM MAP NODE ATTRACTION TO DATA SET POINTS .....                                                 | 151 |
| FIGURE 112: PROGRESSIVE MAPPING OF SOM MAP NODES TO DATA SET POINTS .....                                    | 151 |
| FIGURE 113: COLOR VALUE UPDATES OF SOM MAP NODES WITHIN A BMU'S RADIUS .....                                 | 152 |
| FIGURE 114: COLOR VALUE UPDATES OF SOM MAP NODES WITHIN MORE THAN ONE BMU'S RADIUS.....                      | 152 |
| FIGURE 115. DIMENSIONALITY REDUCTION FROM 2-D TO 1-D .....                                                   | 154 |
| FIGURE 116. PCA VS. LINEAR REGRESSION .....                                                                  | 155 |
| FIGURE 117. PCA RECONSTRUCTION OF FEATURE VECTORS $\mathbf{x}(i)$ FROM $\mathbf{z}(i)$ .....                 | 157 |
| FIGURE 118. AIRCRAFT ENGINE TESTING DATA .....                                                               | 160 |
| FIGURE 119. DECREASING PROBABILITY MOVING FURTHER OUT FROM THE CENTER OF THE DATASET .....                   | 161 |
| FIGURE 120. BELL-SHAPED CURVE OF A GAUSSIAN DISTRIBUTION .....                                               | 162 |
| FIGURE 121. NORMAL DISTRIBUTION PLOTS FOR DIFFERENT VALUES OF THE MEAN AND STANDARD DEVIATION .....          | 163 |
| FIGURE 122. ANOMALY DETECTION EXAMPLE FOR A 2-DIMENSIONAL FEATURE VECTOR .....                               | 165 |
| FIGURE 123. REDUCING $p(\mathbf{x})$ FOR ANOMALOUS EXAMPLES.....                                             | 169 |
| FIGURE 124. FALSE CLASSIFICATION USING SIMPLE PRODUCT OF PROBABILITY DISTRIBUTIONS.....                      | 170 |
| FIGURE 125. MULTIVARIATE GAUSSIAN DISTRIBUTION WITH $\Sigma$ A DIAGONAL MATRIX OF SAME VALUES .....          | 171 |
| FIGURE 126. EFFECT OF DIFFERENT VALUES OF $\sigma_{11}$ ON THE MULTIVARIATE GAUSSIAN DISTRIBUTION .....      | 172 |
| FIGURE 127. EFFECT OF DIFFERENT VALUES OF $\sigma_{22}$ ON THE MULTIVARIATE GAUSSIAN DISTRIBUTION .....      | 172 |
| FIGURE 128. EFFECT OF DIFFERENT POSITIVE OFF-DIAGONAL VALUES ON THE MULTIVARIATE GAUSSIAN DISTRIBUTION ..... | 173 |
| FIGURE 129. EFFECT OF DIFFERENT NEGATIVE OFF-DIAGONAL VALUES ON THE MULTIVARIATE GAUSSIAN DISTRIBUTION ..... | 174 |
| FIGURE 130. EFFECT OF DIFFERENT VALUES FOR $\mu$ ON THE MULTIVARIATE GAUSSIAN DISTRIBUTION .....             | 174 |
| FIGURE 131. ANOMALY DETECTION WITH MULTIVARIATE GAUSSIAN DISTRIBUTION .....                                  | 175 |
| FIGURE 132. HIGH VARIANCE AND HIGH BIAS LEARNING CURVES.....                                                 | 187 |
| FIGURE 133. GRADIENT DESCENT TRAJECTORY FOR STOCHASTIC AND BATCH GRADIENT DESCENT .....                      | 190 |
| FIGURE 134. DIFFERENT POSSIBILITIES WHEN PLOTTING THE COST FOR STOCHASTIC GRADIENT DESCENT LEARNING .....    | 192 |
| FIGURE 135. EXAMPLE IMAGE CONTAINING TEXT FOR OCR .....                                                      | 197 |

|                                                                           |     |
|---------------------------------------------------------------------------|-----|
| FIGURE 136. IDENTIFYING REGIONS CONTAINING TEXT IN AN IMAGE .....         | 198 |
| FIGURE 137. PHOTO OCR MACHINE LEARNING PIPELINE.....                      | 199 |
| FIGURE 138. THE PHOTO OCR MACHINE LEARNING PIPELINE ILLUSTRATED .....     | 199 |
| FIGURE 139. FIXED ASPECT RATIO FOR PEDESTRIAN DETECTION .....             | 200 |
| FIGURE 140. PEDESTRIAN DETECTION TRAINING EXAMPLES.....                   | 201 |
| FIGURE 141. SLIDING WINDOW EXAMPLE.....                                   | 202 |
| FIGURE 142. CHARACTER DETECTION TRAINING EXAMPLES .....                   | 202 |
| FIGURE 143. FINAL PROCESSING STEPS FOR TEXT DETECTION.....                | 203 |
| FIGURE 144. TRAINING EXAMPLES FOR THE CHARACTER SEGMENTATION MODULE ..... | 203 |
| FIGURE 145. 1D SLIDING WINDOW CLASSIFIER FOR CHARACTER SEGMENTATION.....  | 204 |
| FIGURE 146. GENERATING TRAINING EXAMPLES .....                            | 205 |
| FIGURE 147. SYNTHESIZING ARTIFICIAL TRAINING DATA VIA DISTORTIONS .....   | 205 |

## List of Tables

|                                                                                                            |     |
|------------------------------------------------------------------------------------------------------------|-----|
| TABLE 1. HOUSE PRICES ( $y$ ) BY SQUARE FOOTAGE ( $x$ ) .....                                              | 22  |
| TABLE 2. $J\theta$ AS A FUNCTION OF $\theta_1$ , WHERE $\theta_0 = 0$ .....                                | 26  |
| TABLE 3. HOUSE PRICES BY SQ FT, NUMBER OF BEDROOMS, NUMBER OF FLOORS AND AGE.....                          | 33  |
| TABLE 4. TRAINING EXAMPLE SET AUGMENTED WITH $x_0(i) = 1$ .....                                            | 37  |
| TABLE 5. COMPARISON BETWEEN GRADIENT DESCENT AND NORMAL EQUATION.....                                      | 38  |
| TABLE 6. MULTI-CLASS CLASSIFICATION EXAMPLES .....                                                         | 47  |
| TABLE 7. PARAMETER VECTOR AND RESPONSE OF THE $i^{th}$ ACTIVATION UNIT IN THE HIDDEN AND OUTPUT LAYER..... | 58  |
| TABLE 8. EXAMPLE 70-30% SPLIT INTO TRAINING AND TEST DATASETS .....                                        | 81  |
| TABLE 9. EXAMPLE MODEL SELECTION .....                                                                     | 83  |
| TABLE 10. $J_{train}(\theta)$ AND $J_{cv}(\theta)$ VS. HIGH BIAS AND HIGH VARIANCE .....                   | 86  |
| TABLE 11. ACTIONS TO FIX HIGH BIAS AND HIGH VARIANCE OF A LEARNING ALGORITHM .....                         | 92  |
| TABLE 12. SPAM AND NON-SPAM EMAIL EXAMPLE .....                                                            | 93  |
| TABLE 13. EXAMPLES OF SPAM AND NON-SPAM WORDS .....                                                        | 94  |
| TABLE 14. TRUE POSITIVE/NEGATIVE AND FALSE POSITIVE/NEGATIVE CONDITIONS.....                               | 97  |
| TABLE 15. THREE DIFFERENT ALGORITHMS WITH THREE DIFFERENT PAIRS OF PRECISION/RECALL SCORES.....            | 100 |
| TABLE 16. $F1$ SCORES.....                                                                                 | 100 |
| TABLE 17: KEY DIFFERENCES BETWEEN CLASSIFICATION AND CLUSTERING .....                                      | 117 |
| TABLE 18: COMMONLY USED SIMILARITY DISTANCE MEASURES.....                                                  | 120 |
| TABLE 19: COLOR CODING OF A HEALTH ATTRIBUTE FOR A SOM VISUALIZATION.....                                  | 148 |
| TABLE 20. ANOMALY DETECTION VS. SUPERVISED LEARNING ALGORITHMS .....                                       | 167 |

|                                                                                       |     |
|---------------------------------------------------------------------------------------|-----|
| TABLE 21. USEFUL TRANSFORMATIONS FOR MAPPING TO A GAUSSIAN DISTRIBUTION .....         | 168 |
| TABLE 22. UNIVARIATE PRODUCT VS. MULTIVARIATE GAUSSIAN DISTRIBUTION .....             | 177 |
| TABLE 23. MOVIE RATINGS .....                                                         | 178 |
| TABLE 24. MOVIE RATINGS AND FEATURES .....                                            | 179 |
| TABLE 25. PROBLEM FORMULATION WITHOUT A PRIORI FEATURE KNOWLEDGE FOR EACH MOVIE ..... | 181 |
| TABLE 26. BACK-SUBSTITUTION OF $x_1$ AND $x_2$ FROM KNOWN PREDICTION VALUES .....     | 182 |
| TABLE 27. NEW CUSTOMER EVE TO RECOMMEND MOVIES TO .....                               | 184 |
| TABLE 28. CEILING ANALYSIS TABLE FOR THE PHOTO OCR PIPELINING .....                   | 207 |

## 1 Preface

This book is an introduction to Machine Learning (ML) and artificial neural networks (ANNs). It provides the novice to ML and ANNs with a solid understanding of ML and ANNs and veterans in these fields with a useful reference book and refresher. The material in this book not only covers the mathematics which lays at the foundation of ML and ANNs but it also provides numerous examples of real world applications, henceforth, the title. A good understanding of the foundational mathematics is important and therefore the reader is encouraged to not skip through it.

The material in this book covers the broad field that machine learning is, spanning linear regression, logistic regression, stochastic gradient descent, mini-batch gradient descent, regularization, neural networks, clustering, Principal Component Analysis, Support Vector Machines, Gaussian Kernels, stochastic gradient descent, and mini-batch gradient descent. The material also covers important analytical tools to evaluate and debug machine learning algorithms and their performance, such as bias, variance, and learning curve analysis, and provides insightful examples of how these tools are used effectively.

In addition the book covers design methodologies and best practices for the design of large-scale machine learning systems, to focus resources where the biggest improvements in system performance are to be gained. Lastly, the material covers several important machine learning applications, such as anomaly detection, recommender systems, online learning, and photo optical character recognition. The reader will find ample examples that illustrate the concepts and the theory along the way.

The rest of this book is organized as follows:

- Chapter 2 provides a refresher on the linear algebra and calculus used in the book.
- Chapter 3 covers the theory and application of univariate and multivariate linear regression in depth, including gradient descent, hypotheses, cost functions and cost function minimization using iterative and direct methods.
- Chapter 4 covers the theory and application of logistic regression in depth, including complex decision boundaries, hypotheses, cost functions and cost function minimization.
- Chapter 5 covers the theory and application of regularization for linear and logistic regression, including the concepts of under- and over-fitting.

- Chapter 6 covers the theory and application of neural networks as self-learning implementations for single and multi-class classification applications, including backpropagation, scalar and vectorized implementations, and neural network architectures with single and multiple hidden layers.
- Chapter 7 covers tuning strategies for machine learning algorithms, including hypothesis evaluation, model selection, bias and variance, and the usefulness of learning curves.
- Chapter 8 covers best practices in machine learning system design, including error analyses, error metrics, precision and recall, and collection of training examples.
- Chapter 9 covers the theory and application of support vector machines, including the concept of large margin classification and the use of kernels and landmarks.
- Chapter 10 covers the theory and application of clustering for unsupervised learning, including the K-Means algorithm, Agglomerative Hierarchical Clustering, Mean-Shift Clustering, Expectation Maximizing Clustering, DBSCAN and Self-Organizing Maps.
- Chapter 11 covers the theory and application of data compression or dimensionality reduction using principal component analysis.
- Chapter 12 covers the theory and application of anomaly detection using probability densities and kernels.
- Chapter 13 covers the special application of an online recommender system for product or service recommendations, including the use of collaborative filtering.
- Chapter 14 covers the design of large-scale machine learning systems, using stochastic and mini-batch gradient descent, as well as MapReduce.
- Chapter 15 covers an example application for photo OCR to illustrate best practices to optimize resources for improving the performance of a machine learning system under development.

The material in this book is largely based on the Coursera Machine Learning course taught by Prof. Andrew Ng. Novices to ML and ANNs are encouraged to take up the course. Not only is Prof. Ng an excellent teacher and instructor, but the course also offers many hands-on programming exercises.

This page left intentionally blank.

## 2 Math Refresher

### 2.1 Linear Algebra Refresher

The reader is encouraged to read through this chapter if not thoroughly familiar with the linear algebra that underlies the material covered in this book.

#### Dimension of a vector:

Vectors are represented as a row or a column in this book. The notation used to denote a row vector of  $n$  elements is:  $v \in \mathbb{R}^{1 \times n}$  (i.e., 1 by  $n$ ). The notation used to denote a column vector of  $n$  elements is:  $v \in \mathbb{R}^{n \times 1}$  (i.e.,  $n$  by 1).

#### Dimension of a matrix:

A matrix of  $m$  rows and  $n$  columns is denoted as:  $A \in \mathbb{R}^{m \times n}$  (i.e.,  $m$  by  $n$ ). The element in the  $i$ th row and the  $j$ th row is denoted as  $a_{ij}$ .

#### The transpose operator:

The transpose of a vector  $v$  is written as  $v^T$  and turns a row vector into a column vector and vice versa. This means that if  $v \in \mathbb{R}^{1 \times n}$ , then  $v^T \in \mathbb{R}^{n \times 1}$ .

The transpose of a matrix  $A \in \mathbb{R}^{m \times n}$  transposes each column of the matrix to a row. Hence,  $A^T \in \mathbb{R}^{n \times m}$ .

#### Diagonal matrix:

A diagonal matrix  $A \in \mathbb{R}^{n \times n}$  has all its off-diagonal elements equal to zero and has equal number of rows and columns:

$$A = \begin{bmatrix} a_{11} & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & a_{nn} \end{bmatrix}$$

#### Identity matrix:

An Identity matrix  $I \in \mathbb{R}^{n \times n}$  is a special type of diagonal matrix which has all its diagonal elements equal to 1:

$$I = \begin{bmatrix} 1 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & 1 \end{bmatrix}$$

### Vector-vector addition:

The addition of a vector  $v \in \mathbb{R}^{1 \times n}$ , and a vector  $w \in \mathbb{R}^{1 \times n}$  is again a vector  $y \in \mathbb{R}^{1 \times n}$ :

$$v + w = y$$

### Matrix-matrix addition:

The addition of matrix  $A \in \mathbb{R}^{n \times n}$  and matrix  $B \in \mathbb{R}^{n \times n}$  is again a matrix  $C \in \mathbb{R}^{n \times n}$ :

$$A + B = C$$

### Scalar-vector multiplication:

Multiplication of a vector  $v$  by a scalar  $s$  is the element-wise multiplication of the vector by the scalar:

$$s \times v = v \times s = s \times [v_1 \dots v_n] = [s \times v_1 \dots s \times v_n]$$

### Scalar-matrix multiplication:

Multiplication of a matrix  $M$  by a scalar  $s$  is the element-wise multiplication of the matrix by the scalar:

$$A \times s = s \times A = s \times \begin{bmatrix} a_{11} & \dots & a_{1n} \\ \vdots & \ddots & \vdots \\ a_{m1} & \dots & a_{mn} \end{bmatrix} = \begin{bmatrix} s \times a_{11} & \dots & s \times a_{1n} \\ \vdots & \ddots & \vdots \\ s \times a_{m1} & \dots & s \times a_{mn} \end{bmatrix}$$

### Inner-product:

The inner-product of two vectors  $v, w \in \mathbb{R}^{1 \times n}$  is the sum product of the elements of the vectors:

$$v \times w^T = \sum_{i=1}^n v_i \times w_i = s \in \mathbb{R}$$

### Vector-matrix multiplication:

Multiplying a matrix  $A \in \mathbb{R}^{m \times n}$  by a vector  $v \in \mathbb{R}^{n \times 1}$  is performed by taking the inner-product of each row  $a_i \in \mathbb{R}^{1 \times n}$  of the matrix with the vector  $v$ :

$$A \times v = \begin{bmatrix} a_1 \\ \vdots \\ a_m \end{bmatrix} \times v = \begin{bmatrix} a_1 \times v \\ \vdots \\ a_m \times v \end{bmatrix} = \begin{bmatrix} w_1 \\ \vdots \\ w_m \end{bmatrix} = w \in \mathbb{R}^{m \times 1}$$

### Matrix-matrix multiplication:

Multiplying a matrix  $A \in \mathbb{R}^{m \times n}$  by a matrix  $B \in \mathbb{R}^{n \times p}$  is performed by taking the inner-product of each row  $a_i \in \mathbb{R}^{1 \times n}$  of the matrix  $A$  with each column  $b_j \in \mathbb{R}^{n \times 1}$  of the matrix  $B$ :

$$A \times B = \begin{bmatrix} a_1 \\ \vdots \\ a_m \end{bmatrix} \times [b_1 \dots b_p] = \begin{bmatrix} a_1 \times b_1 & \dots & a_1 \times b_p \\ \vdots & \ddots & \vdots \\ a_m \times b_1 & \dots & a_m \times b_p \end{bmatrix} = \begin{bmatrix} c_{11} & \dots & c_{1p} \\ \vdots & \ddots & \vdots \\ c_{m1} & \dots & c_{mp} \end{bmatrix} = C \in \mathbb{R}^{m \times p}$$

A special case of a matrix-matrix multiplication is the multiplication of a matrix  $A \in \mathbb{R}^{m \times n}$  with an identity matrix  $I \in \mathbb{R}^{n \times n}$ :

$$A \times I = A$$

#### Inverse of a matrix:

A matrix  $A \in \mathbb{R}^{n \times n}$  has an inverse  $A^{-1} \in \mathbb{R}^{n \times n}$ , so that  $A \times A^{-1} = I \in \mathbb{R}^{n \times n}$ , if and only if the matrix has full rank.

Note: Several other properties apply to an invertible matrix, such as its determinant being nonzero, and being decomposable in the product of a lower and upper triangular matrix.

#### Full rank:

A matrix  $A \in \mathbb{R}^{n \times n}$  has full rank if all columns and rows of the matrix are linearly independent, i.e., none of the rows of the matrix can be expressed as a linear combination of any set of the rows of the matrix, except of themselves, and none of the columns of the matrix can be expressed as a linear combination of any set of the columns of the matrix, except of themselves.

## 2.2 Calculus Refresher

The reader is encouraged to read through this chapter if not thoroughly familiar with the calculus that underlies the material covered in this book.

#### Derivative of the sum or difference of two functions:

$$\begin{aligned} h &= f \pm g \\ h' &= f' \pm g' \end{aligned} \tag{1}$$

#### Derivative of the product of two functions:

$$\begin{aligned} h &= f \times g \\ h' &= f' \times g + f \times g' \end{aligned} \tag{2}$$

#### Derivative of the quotient of two functions:

$$h = \frac{f}{g}$$

$$h' = (f' \times g - f \times g')/g^2 \quad (3)$$

**The chain rule for derivatives:**

$$h = g(f)$$

$$h' = g'(f) \times f' \quad (4)$$

**Derivative of  $f(x) = e^x$ :**

$$f' = e^x \quad (5)$$

**Derivative of the sigmoid function:**

$$f(x) = \frac{1}{(1 + e^{-x})}$$

Using the quotient rule with  $g(x) = (1 + e^{-x})$  and  $h(x) = 1$ , and the chain rule with  $k(x) = -x$  and  $l(k(x)) = e^{k(x)}$ :

$$f'(x) = -\frac{e^{-x}}{(1+e^{-x})^2} = \frac{1}{(1+e^{-x})} - \frac{1}{(1+e^{-x})^2} = f(x)(1 - f(x)) \quad (6)$$

## 3 Linear Regression

### 3.1 Single Variable Linear Regression

#### 3.1.1 Model Representation

Consider the following problem. We are interested in creating a model that can predict housing prices in Portland, Oregon. We collected a set of data points that link the square footage of a house with its price and tabulated these data points in Table 1.

| $i$ | Size (sqft): $x^{(i)}$ | Price (\$K): $y^{(i)}$ |
|-----|------------------------|------------------------|
| 1   | 2104                   | 460                    |
| 2   | 1416                   | 232                    |
| 3   | 1534                   | 315                    |
| 4   | 852                    | 178                    |
| 5   | 1530                   | 300                    |
| 6   | 1451                   | 235                    |
| 7   | 2051                   | 405                    |
| 8   | 901                    | 185                    |
| 9   | 2181                   | 475                    |
| 10  | 567                    | 131                    |

**Table 1. House prices ( $y$ ) by square footage ( $x$ )**

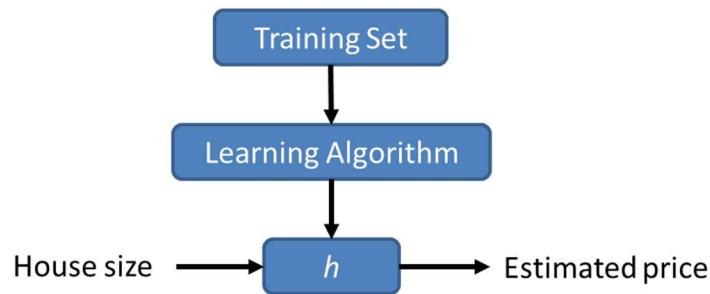
Let's introduce the following terminology:

1.  $m$  = the number of data point pairs  $(x^{(i)}, y^{(i)})$  or *training examples*
2.  $x^{(i)}$  = the  $i^{\text{th}}$  *input variable* or *feature*
3.  $y^{(i)}$  = the  $i^{\text{th}}$  *output variable* or *target variable*

Therefore, for the dataset in Table 1 we get:

1. The 1<sup>st</sup> example pair is  $(x^{(1)}, y^{(1)}) = (2104, 460)$
2. The 2<sup>nd</sup> training example is  $(x^{(2)}, y^{(2)}) = (1416, 232)$
3. The last training example is  $(x^{(10)}, y^{(10)}) = (567, 131)$
4. And  $m = 10$ .

What we are interested in is an algorithm that learns from the training example set and that predicts the price of any size house fed into the algorithm. See Figure 1 .



**Figure 1. The house price predictor**

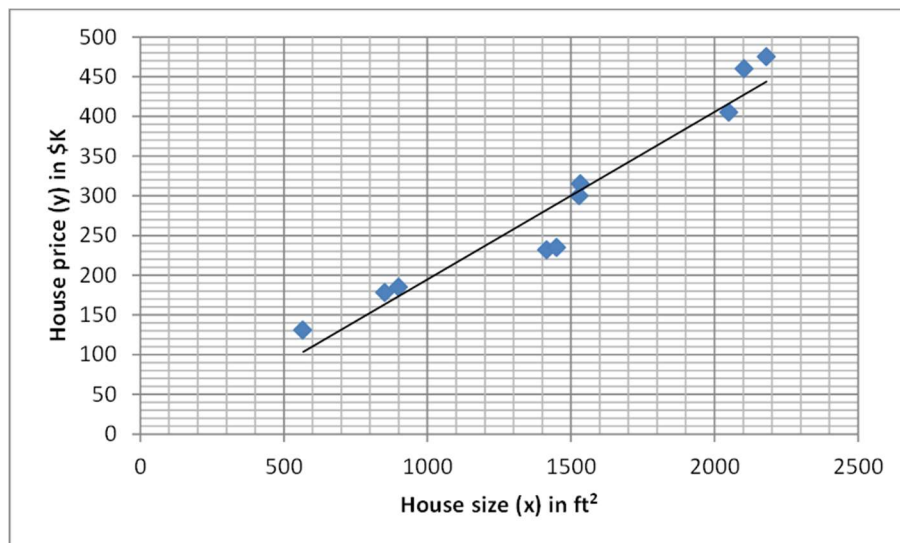
We want to discover the relationship or *hypothesis*  $h$  between the *input variable* or *feature*  $x$ , and the *output variable* or *target variable*  $y$ . Plotting  $y^{(i)}$  against  $x^{(i)}$  can provide us with a clue. In Figure 2 we plotted the example pairs from Table 1 and drew a straight line to represent the *hypothesis*:

$$h(x^{(i)}) = y^{(i)} \quad (7)$$

Given that the model is linear, we can re-write equation (7) as follows:

$$h_{\theta}(x^{(i)}) = \theta_0 + \theta_1 x^{(i)} = y^{(i)} \quad (8)$$

The challenge is to find  $\theta_0$  and  $\theta_1$  such that  $h_{\theta}(x^{(i)})$  predicts  $y^{(i)}$  as best as possible for the complete training example set. This is the subject of the next section.



**Figure 2. Scatter plot of example pairs  $(x^{(i)}, y^{(i)})$**

### 3.1.2 The Cost Function

Given the hypothesis  $h_\theta(x^{(i)})$  in equation (8), the challenge is to choose the parameters  $\theta_0$  and  $\theta_1$  such that we minimize the error between  $h_\theta(x^{(i)})$  and  $y^{(i)}$  for the complete set of 10 training examples ( $m = 10$ ).

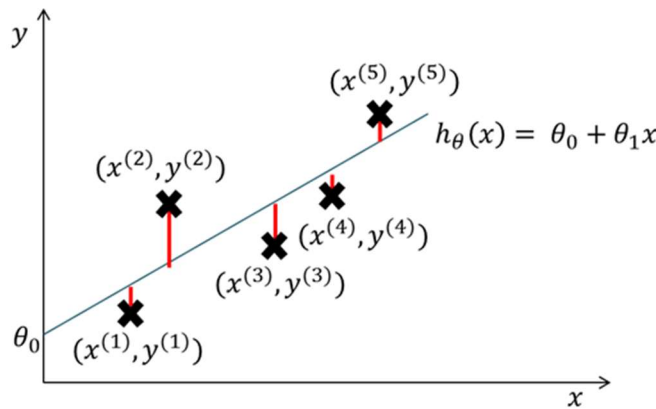
One way of doing this is to minimize the sum of the squares of the differences between the pairs  $(h_\theta(x^{(i)}), y^{(i)})$  for the complete set of example pairs,  $(x^{(i)}, y^{(i)})$ ,  $i \in \{1, \dots, m\}$ . I.e., we are looking to compute the minimum of  $J(\theta)$ :

$$\min_{\theta_0, \theta_1} J(\theta) \quad (9)$$

We can write  $J(\theta)$  as follows:

$$J(\theta) = \frac{1}{2m} \sum_{i=1}^m (h_\theta(x^{(i)}) - y^{(i)})^2, \text{ for } \theta = (\theta_0, \theta_1) \quad (10)$$

This is the classical and well understood Least Squares Minimization (LSM) problem, where  $J(\theta)$  is the Least Square Error (LSE) or the cost associated with the choice of parameter  $\theta$ . A graphical interpretation is shown in Figure 3.



**Figure 3. Errors  $\epsilon^{(i)} = h_\theta(x^{(i)}) - y^{(i)}$**

Solving equation (9)  $\min_{\theta_0, \theta_1} J(\theta)$  requires finding the zeros for the partial derivatives in  $\theta_0$  and  $\theta_1$  of  $J(\theta)$ :

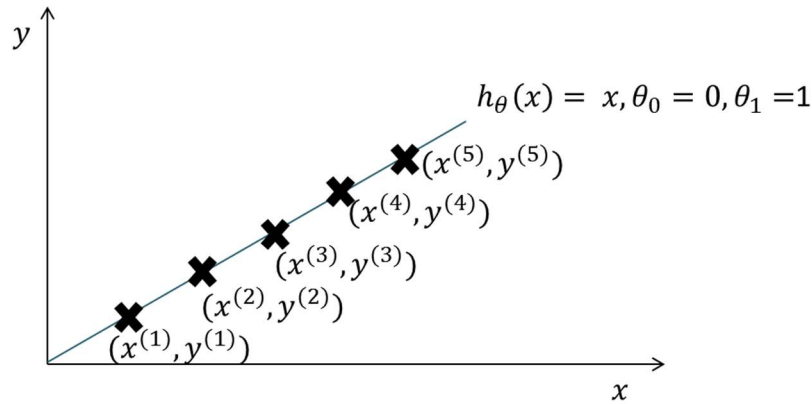
$$\frac{\partial}{\partial \theta_0} J(\theta_0, \theta_1) = 0 \quad (11)$$

$$\frac{\partial}{\partial \theta_1} J(\theta_0, \theta_1) = 0 \quad (12)$$

Before stating the solution, we first develop an intuitive understanding how to approach a solution. Let:

- (1)  $\theta_0 = 0$
- (2)  $h_\theta(x^{(i)}) = \theta_1 x$  for  $\theta_0 = 0$
- (3)  $J(\theta) = \frac{1}{2m} \sum_{i=1}^m (h_\theta(x^{(i)}) - y^{(i)})^2 = \frac{1}{2m} \sum_{i=1}^m (\theta_1 x^{(i)} - y^{(i)})^2$
- (4)  $\min_{\theta_1} J(\theta_1)$

Assume for a moment that we have the situation in Figure 4 for example pairs  $(x^{(i)}, y^{(i)})$ .



**Figure 4. Hypothesis with cost or LSE equal to 0 for the example set**

That is:

- (1)  $\theta_0 = 0$
- (2)  $\theta_1 = 1$
- (3)  $J(\theta) = \frac{1}{2m} \sum_{i=1}^m (h_\theta(x^{(i)}) - y^{(i)})^2 = \frac{1}{2m} \sum_{i=1}^m (y^{(i)} - y^{(i)})^2 = 0$

Hence, in this case  $\min_{\theta} J(\theta)$  is obtained for  $\theta_0 = 0$  and  $\theta_1 = 1$ . I.e.,  $J(\theta) = 0$ .

If we let  $\theta_1$  vary,  $J(\theta)$  becomes a function of  $\theta_1$ . Figure 5 shows how  $h_\theta(x)$  varies with different choices of  $\theta_1$ , where  $\theta_0 = 0$ .

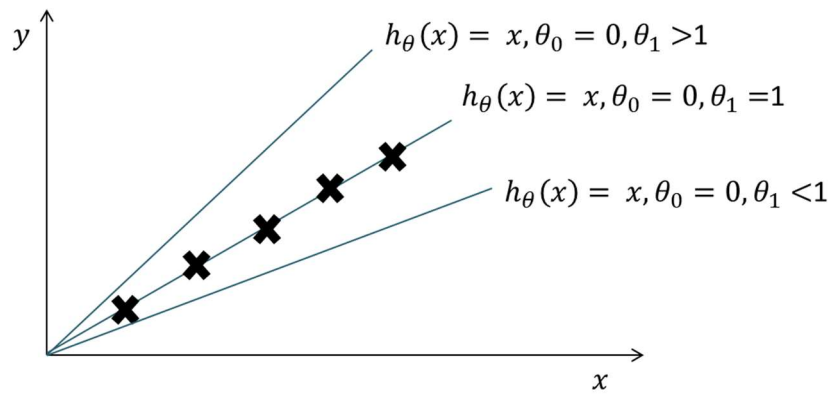


Figure 5. Letting  $\theta_1$  vary around 1 where the cost  $J(\theta) = 0$ .

From equation (10) it follows that the cost function,  $J(\theta)$ , is a quadratic function:

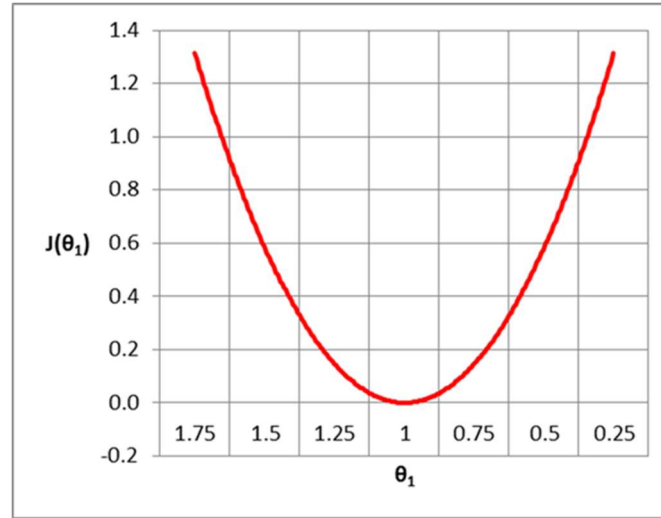
$$J(\theta) = \frac{1}{2m} \sum_{i=1}^m (h_{\theta}(x) - y)^2 = \frac{1}{2m} \sum_{i=1}^m (\theta_1 x - y)^2 \quad (13)$$

The values of  $J(\theta)$  in equation (13) are tabulated against the values of  $\theta = (\theta_0 = 0, \theta_1)$  in Table 2 for a few values of  $\theta_1$  around 1, for an example set of  $m = 3$  and  $x \in \{1, 2, 3\}$ .

| $\theta_1$ | $J(\theta)$ |
|------------|-------------|
| 1.75       | 1.3         |
| 1.5        | 0.6         |
| 1.25       | 0.1         |
| 1          | 0.0         |
| 0.75       | 0.1         |
| 0.5        | 0.6         |
| 0.25       | 1.3         |

Table 2.  $J(\theta)$  as a function of  $\theta_1$ , where  $\theta_0 = 0$

The graph of  $J(\theta_1)$  is shown in Figure 6.



**Figure 6. Graph of  $J(\theta_1)$  in equation (6) as a function of  $\theta_1$**

The key take away from this graph is that the cost function  $J(\theta_1)$  is a hyperbolic function with global minimum at  $\theta_1 = 1$ , for  $\theta_0 = 0$ . Indeed, using calculus:

$$J(\theta_0) = \frac{1}{6}((\theta_0 - 1)^2 + (\theta_0 - 2)^2 + (\theta_0 - 3)^2), \text{ hence, } \frac{\partial}{\partial \theta_0} J(\theta_0) = \theta_0 - 2 = 0, \text{ for } \theta_0 = 2$$

Similarly, from  $J(\theta) = \frac{1}{2m} \sum_{i=1}^m (\theta_0 - y)^2$ , for  $\theta_1 = 0$ , we can find that the cost function  $J(\theta_0)$  is a hyperbolic function with global minimum at  $\theta_0 = 2$ . Indeed, using calculus:

$$J(\theta_1) = \frac{1}{6}((\theta_1 - 1)^2 + (2\theta_1 - 2)^2 + (3\theta_1 - 3)^2), \text{ hence, } \frac{\partial}{\partial \theta_1} J(\theta_1) = \frac{5}{3}(\theta_1 - 1) = 0, \text{ for } \theta_1 = 1$$

### 3.1.3 Gradient Descent

We can solve for the global minimum of the cost function using calculus as shown previously, or we can use an iterative method. Figure 5 and Figure 6 give us a clue how to construct an iterative method. The key in Figure 5 and Figure 6 is that we let  $\theta_1$  change in the direction that reduces  $J(\theta_1)$  with each step. The following algorithm accomplishes this and is called gradient descent:

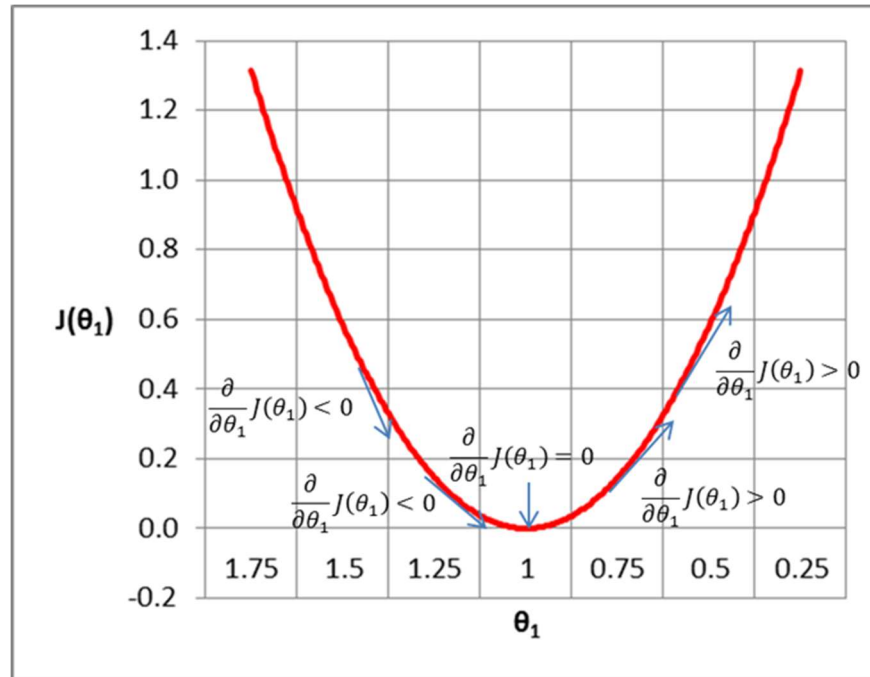
*Repeat until convergence for  $\alpha \geq 0$  {* (14)

$$\theta_0 := \theta_0 - \alpha \frac{\partial}{\partial \theta_0} J(\theta_0, \theta_1),$$

$$\theta_1 := \theta_1 - \alpha \frac{\partial}{\partial \theta_1} J(\theta_0, \theta_1),$$

*}*

The partial derivative term  $\frac{\partial}{\partial \theta_j} J(\theta_0, \theta_1)$  provides the direction in which  $\theta_j$  must be changed for each step, until  $J(\theta_0, \theta_1)$  has reached its minimum. See Figure 7. For the example here above this occurs for  $\theta_0 = 2$  and  $\theta_1 = 1$ .



**Figure 7. Using the partial derivative in  $\theta_1$  to minimize  $J(\theta_1)$  iteratively**

The term  $\alpha$  is called the *learning rate*. Its magnitude determines how fast  $J(\theta_0, \theta_1)$  reaches its minimum. Note from Figure 7 that if  $\alpha$  is too small,  $J(\theta_0, \theta_1)$  will take a long time before reaching its minimum, whereas if  $\alpha$  is too large,  $J(\theta_0, \theta_1)$  keeps overshooting the minimum and never convergence to the minimum.

#### Important notes:

1. When updating  $\theta_j$ , it is important to update  $J(\theta_0, \theta_1)$  only after *all* new  $\theta_j$  variables have been updated.
2. It's not guaranteed that  $J(\theta_0, \theta_1)$  converges to a global minimum. Depending on the starting values of  $\theta_j$ ,  $J(\theta_0, \theta_1)$  can end up in a local minimum. See example in Figure 8.
3. It's not necessary to change the value of  $\alpha$  while  $J(\theta_0, \theta_1)$  approaches a minimum, since the derivative term  $\frac{\partial}{\partial \theta_j} J(\theta_j)$  continues to shrink, provided that  $\alpha$  is not too large. See Figure 9.

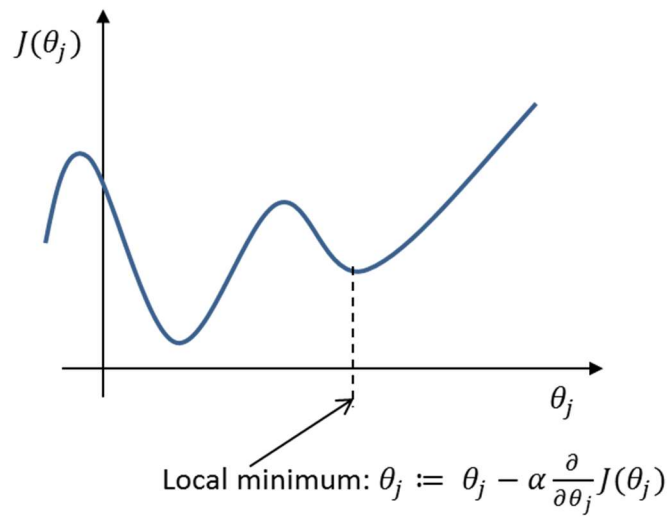


Figure 8. Convergence to a local instead of the global minimum

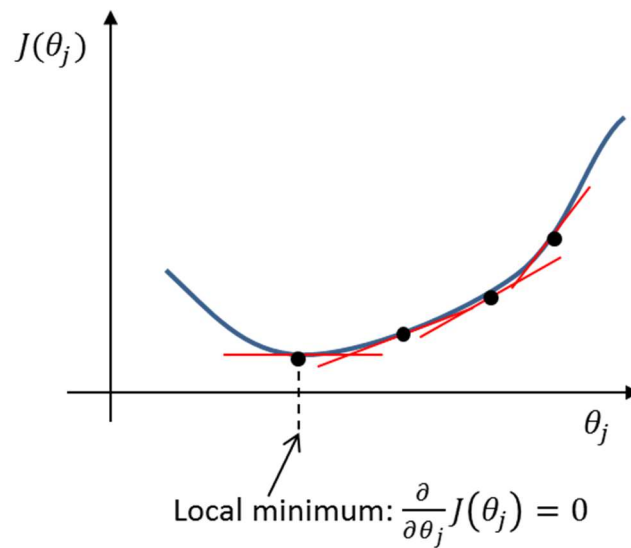


Figure 9. Shrinking derivative  $\frac{\partial}{\partial \theta_j} J(\theta_j)$  as  $\theta_j$  approaches the minimum

Next we compute the derivative term  $\frac{\partial}{\partial \theta_j} J(\theta_j)$  for  $\theta_j$ .

Remember that:

$$J(\theta) = \frac{1}{2m} \sum_{i=1}^m (h_{\theta}(x^{(i)}) - y^{(i)})^2$$

and

$$h_{\theta}(x^{(i)}) = \theta_0 + \theta_1 x^{(i)} \quad (15.b)$$

Using the fact that the derivative of a function  $f(x) = (ax + b)^2$  is  $f'(x) = 2a(ax + b)$ , it is easy to see that:

$$\frac{\partial}{\partial \theta_0} J(\theta) = \frac{1}{m} \sum_{i=1}^m (h_{\theta}(x^{(i)}) - y^{(i)}) \quad (16.a)$$

$$\frac{\partial}{\partial \theta_1} J(\theta) = \frac{1}{m} \sum_{i=1}^m (h_{\theta}(x^{(i)}) - y^{(i)}) x^{(i)} \quad (16.b)$$

Hence, the gradient descent algorithm in (14) can be rewritten as:

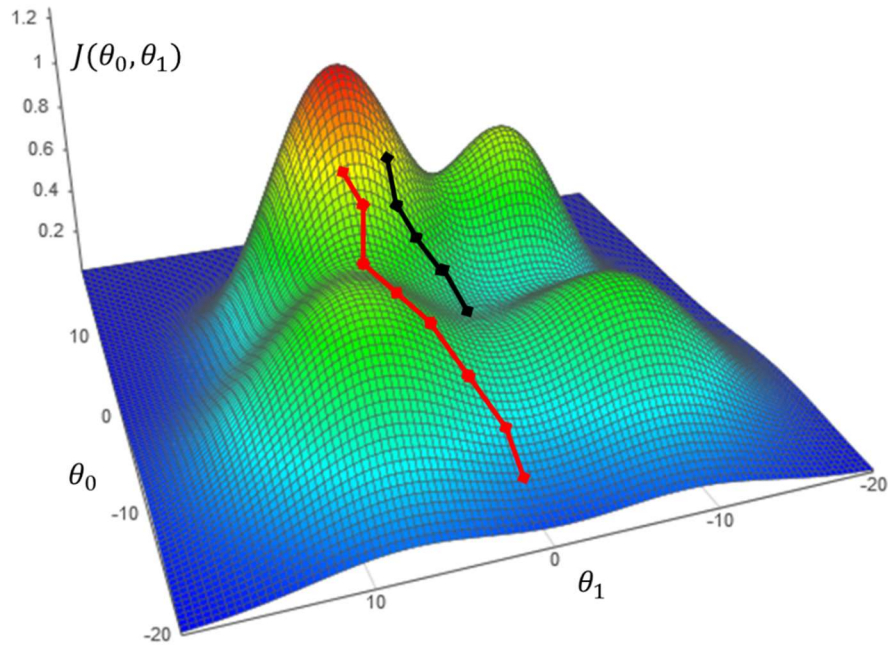
*Repeat until convergence for  $\alpha \geq 0$  {* (17)

$$\theta_0 := \theta_0 - \alpha \left( \frac{1}{m} \sum_{i=1}^m (h_{\theta}(x^{(i)}) - y^{(i)}) \right)$$

$$\theta_1 := \theta_1 - \alpha \left( \frac{1}{m} \sum_{i=1}^m (h_{\theta}(x^{(i)}) - y^{(i)}) x^{(i)} \right),$$

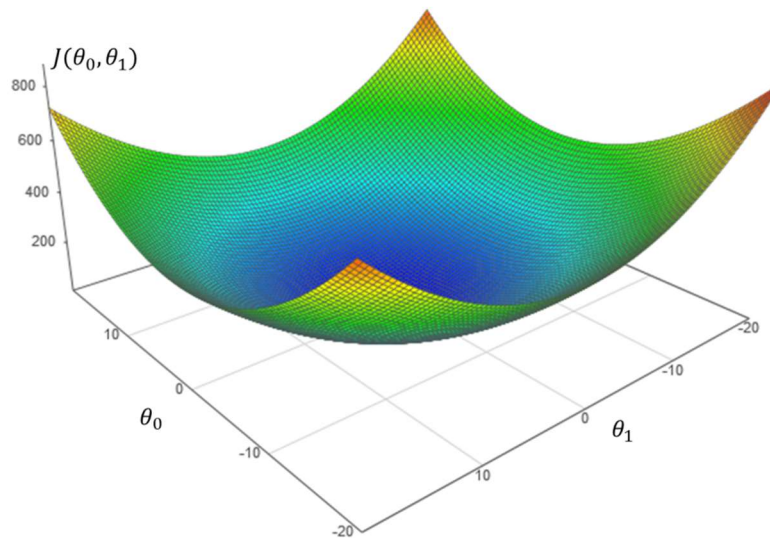
*}*

A visualization of the gradient descent steps is provided in Figure 10 for two different starting points.



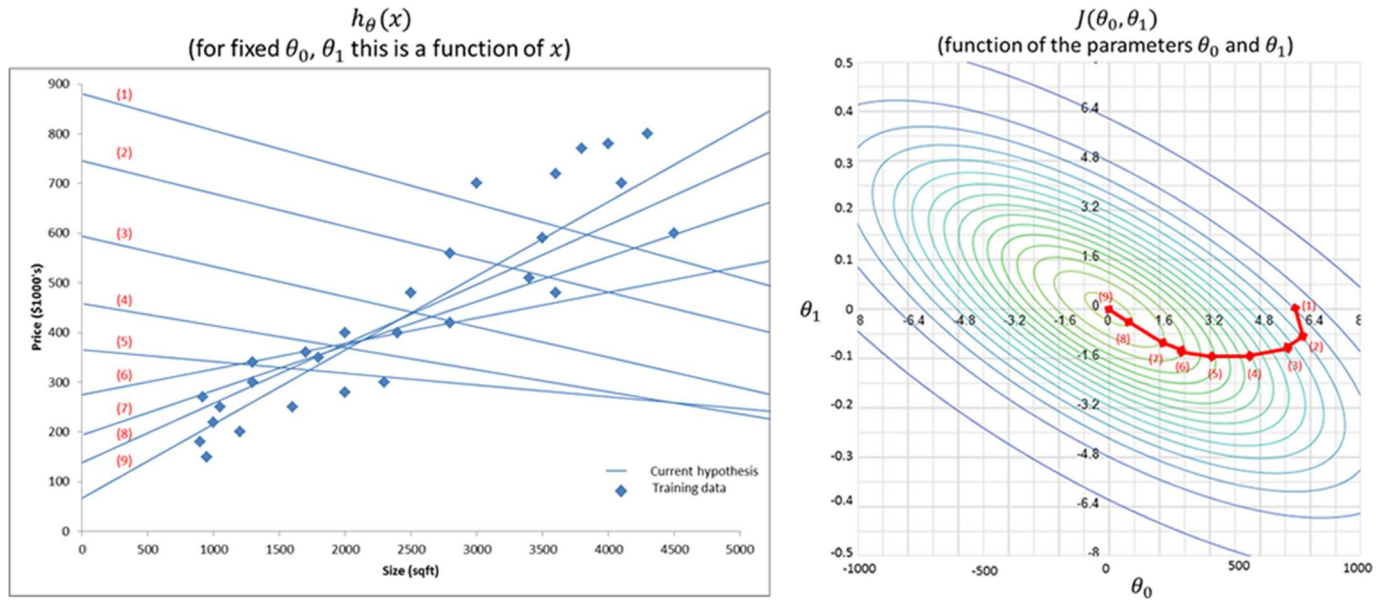
**Figure 10. Visualization of the gradient descent algorithm in (17)**

In case  $J(\theta_0, \theta_1)$  is not a concave function, as shown in Figure 11, there can be multiple local minima, and as noted earlier, depending on the starting values of  $\theta_0$  and  $\theta_1$ , gradient descent can end up in a local minimum (the red path in Figure 10).



**Figure 11.  $J(\theta)$  as a concave function**

An alternative and easier way of visualizing the progression of the gradient descent is through a contour map. The sequence in Figure 12 shows the gradient descent path in the contour map of the cost function  $J(\theta)$  side by side with the change in the hypothesis  $h_\theta(x)$ .



**Figure 12.** Hypothesis  $h_\theta(x)$  changes with corresponding gradient descent steps for cost  $J(\theta)$

As a final note, the gradient descent algorithm in (17) is also called *batch gradient descent* since each step uses all the training examples.

### 3.2 Multivariate Linear Regression

In the previous section we considered single-variable linear regression, or univariate linear regression. I.e., there was only one variable in the training example set as shown in Table 1 for square feet vs. price. Note that the hypothesis for univariate linear regression  $h_\theta(x^{(i)}) = \theta_0 + \theta_1 x^{(i)}$  has two variables though,  $\theta_0$  and  $\theta_1$ . However, univariate refers to single feature and not the number of parameters in the hypothesis.

In this section we expand univariate linear regression to multivariate linear regression. Consider augmenting square feet with number of bedrooms, number of floors and age of the home. See Table 3.