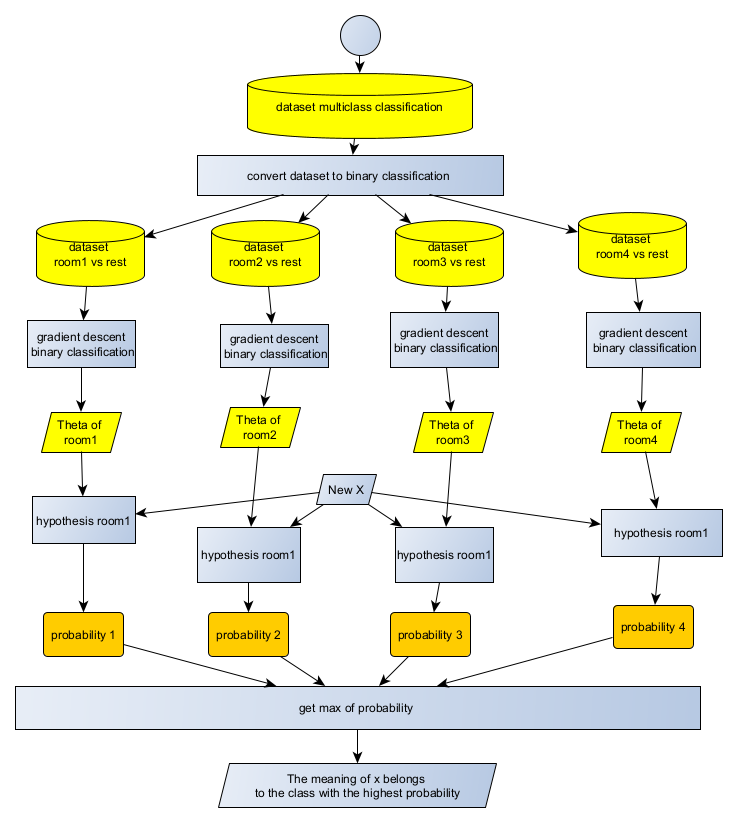
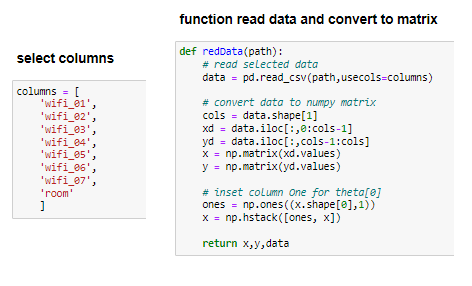
**Logistic Regression:**

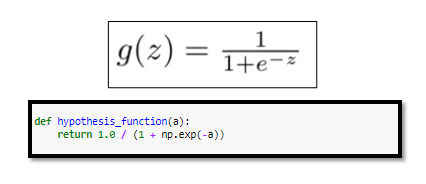
* Logistic regression works only for binary classification and we have multi class classification for this we create 4 dataset everyone is binary classification



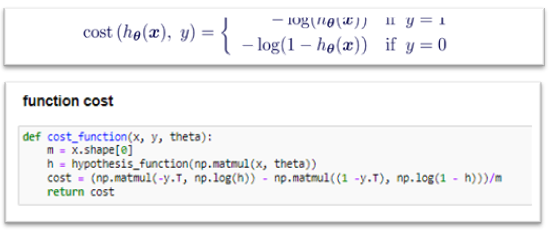
* Read dataset files and select columns
* Converted to matrix X add column One values for Theta0 and matrix Y = {0,1}



* **Hypothesis function:**

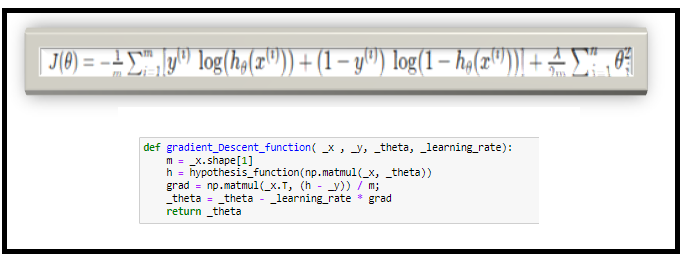
****

* **Cost Function**

****

**Gradient descent function:**

* **Function J**

****

* **Gradient descent function**

**parameter**

1. \_X : array of input data
2. \_Y: array of result = {0,1}
3. **\_**learning\_rate : For Gradient Descent to work, we must choose the learning rate wisely. The learning rate α determines how rapidly we update the parameters. If the learning rate is too large, we may "overshoot" the optimal value. Similarly, if it is too small, we will need too many iterations to converge to the best values.
4. \_n\_iterations : The number of repetitions: the higher the accuracy, the more accurate the results

* Create list of thetas

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **index** | **0** | **1** | **2** | **3** | **4** | **N** |
| **Theta** | **Theta0** | **Theta1** | **Theta2** | **Theta3** | **Theta4** | **Theta N** |

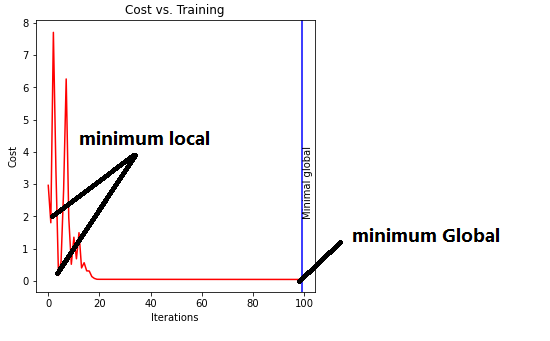
* Create list of cost every Theta in same index

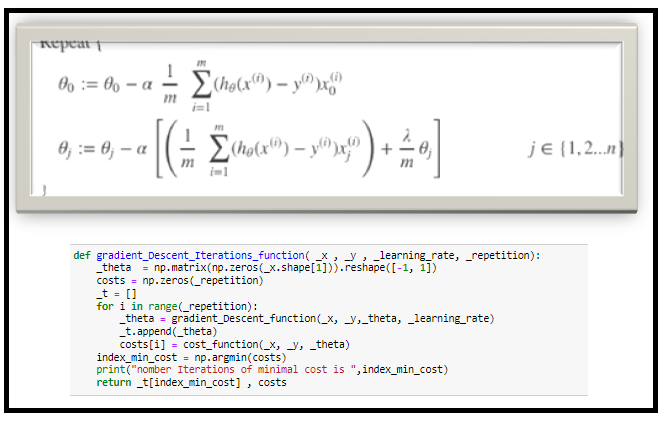
|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **index** | **0** | **1** | **2** | **3** | **4** | **N** |
| **costs** | **Cost theta0** | **Cost theta1** | **Cost theta2** | **Cost theta3** | **Cost theta4** | **Cost theta N** |

* **Skip minimum local**

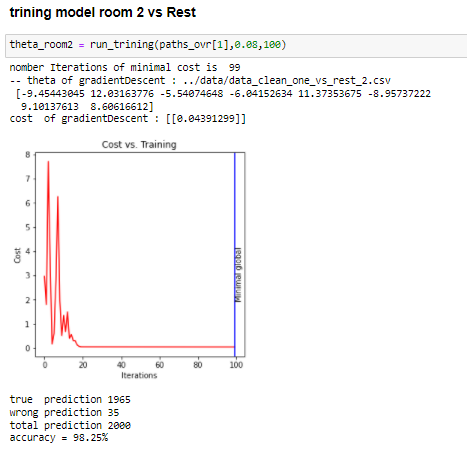
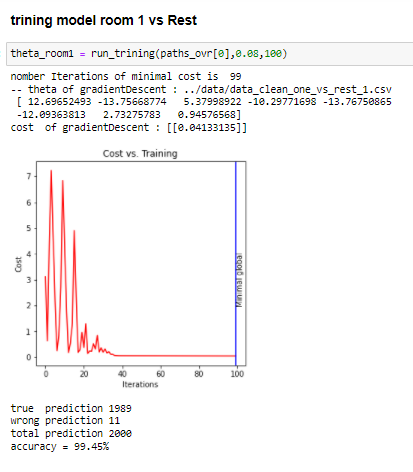
**Best value of** learning\_rate : 0.08

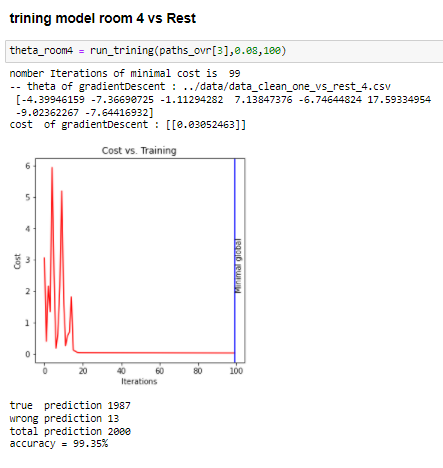
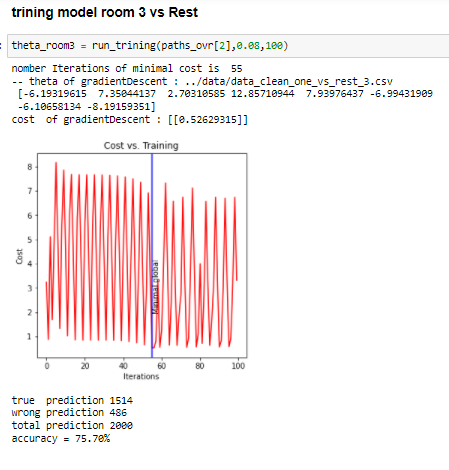
**Optimal value** iterations is : 100

* ****
* We repeat several times, and every time we get a new theta and the cost. Then we calculate the lowest cost value, extract the index number, and restore the value of theta.

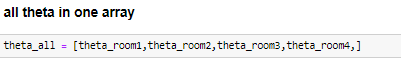
****

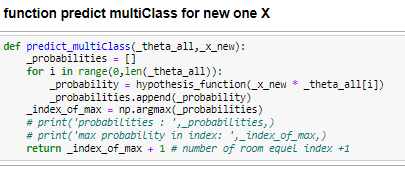
* **Result for every model binary classification theta + cost +accuracy**

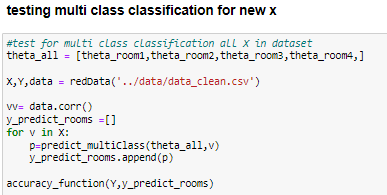




**multi class classification**







* **Result**

|  |
| --- |
| **true prediction 1908** |
| **wrong prediction 92** |
| **total prediction 2000** |
| **Accuracy = 95.40%** |