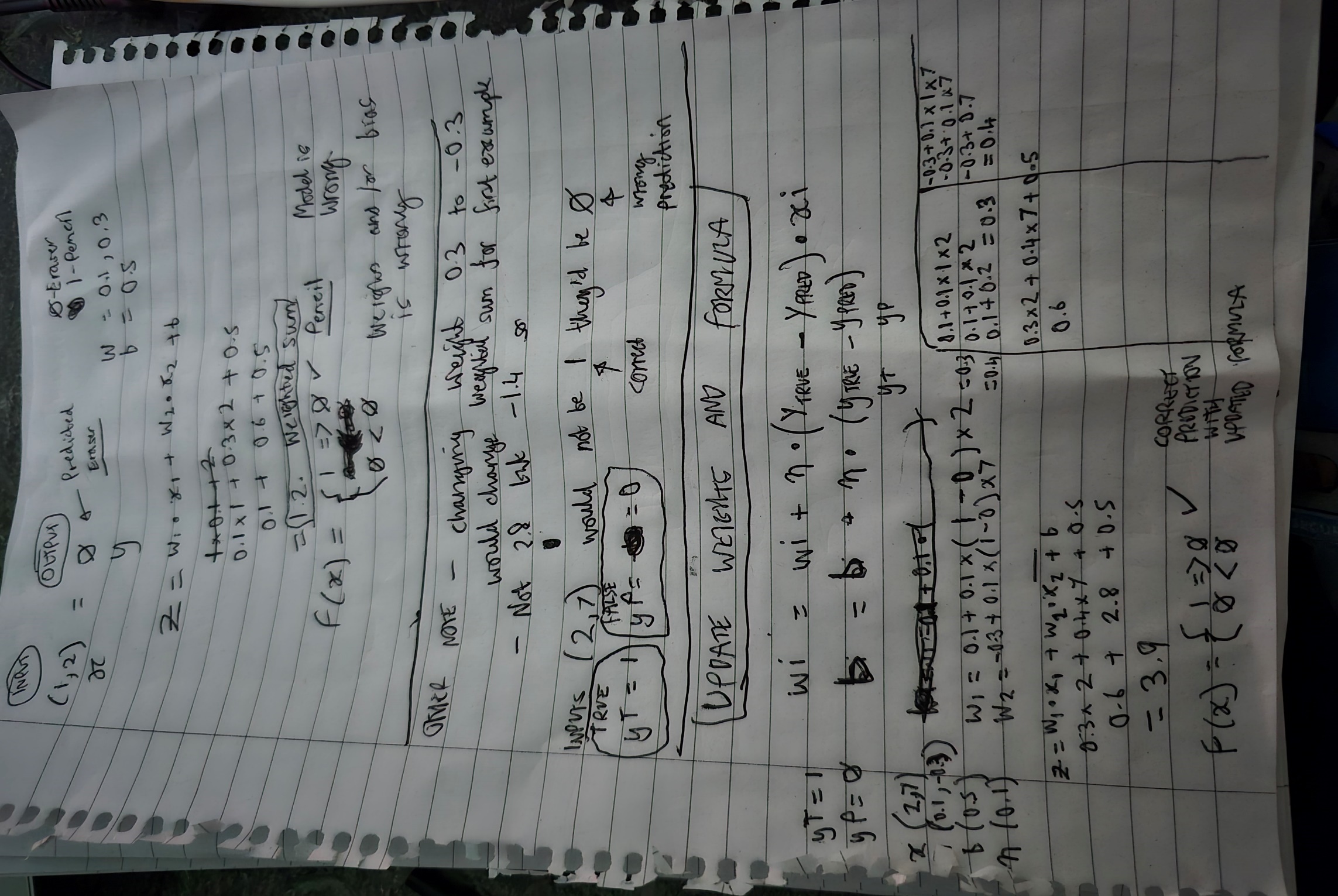
A piece of paper with writing on it

Description automatically generated



***Update - In the last section title UPDATE WEIGHTS AND FORMULA there is an error.***

***The bias value b is still using the original value of 0.5 and hasn’t had the new formula applied. So therefore the calculation as a whole is wrong.***

*Calculate weight and bias*

weight = wi + n \* (yT - yP) \* xi

bias = b + n \* (yT – yP)

*So…*

w1 = 0.1 + 0.1 \* (1 – 0) \* 2 = 0.3

w2 = -0.3 + 0.1 \* (1 – 0) \* 7 = 0.4

b = 0.5 + 0.1 \* (1 – 0) = 0.6

*Then apply new weights and bias to training the model*

Z = w1 \* x1 + w2 \* x2 + b

Z = 0.3 \* 2 + 0.4 \* 7 + 0.6

Z = 0.6 + 2.8 + 0.6

Z = 4

F(x) = 1 => 0 || 0 < 0

**F(4) = 1**

**Put into Code**

**Using weights [0.1, -0.3] and bias [0.5] for all inputs…**

const trainInputs = [

  [2, 7],

  [3, 6],

  [1, 1],

  [1, 2],

  [2, 1],

]

const trainLabels = [1, 1, 0, 0, 0]

class Perceptron {

  constructor(learningRate = 0.1) {

    this.weights = [0.1, -0.3]  // unmodified weights

    this.bias = 0.5             // unmodified bias

    this.learningRate = learningRate

  }

  train(trainData) {

    for (let i = 0; i < trainData.length; i++) {

      let sum = this.bias

      let inputs = trainData[i]

      for (let j = 0; j < inputs.length;  j++) {

        sum += inputs[j] \* this.weights[j]

      }

      console.log(sum)

    }

  }

}

const perceptron = new Perceptron()

perceptron.train(trainInputs)

***Outputs***

-1.4000000000000001

-0.9999999999999998

0.3

0

0.39999999999999997

**Using weights [0.1, 0.3] and bias [0.5] for all inputs…**

const trainInputs = [

  [2, 7],

  [3, 6],

  [1, 1],

  [1, 2],

  [2, 1],

]

const trainLabels = [1, 1, 0, 0, 0]

class Perceptron {

  constructor(learningRate = 0.1) {

    this.weights = [0.1, 0.3]  // unmodified weights

    this.bias = 0.5             // unmodified bias

    this.learningRate = learningRate

  }

  train(trainData) {

    for (let i = 0; i < trainData.length; i++) {

      let sum = this.bias

      let inputs = trainData[i]

      for (let j = 0; j < inputs.length;  j++) {

        sum += inputs[j] \* this.weights[j]

      }

      console.log(sum)

    }

  }

}

const perceptron = new Perceptron()

perceptron.train(trainInputs)

***Outputs***

2.8

2.5999999999999996

0.8999999999999999

1.2

1

**Using modified formula - weights [0.3, 0.4] and bias [0.6] for all inputs…**

const trainLabels = [1, 1, 0, 0, 0]

class Perceptron {

  constructor(learningRate = 0.1) {

    this.weights = [0.3, 0.4]  // modified weights

    this.bias = 0.6             // modified bias

    this.learningRate = learningRate

  }

  train(trainData) {

    for (let i = 0; i < trainData.length; i++) {

      let sum = this.bias

      let inputs = trainData[i]

      for (let j = 0; j < inputs.length;  j++) {

        sum += inputs[j] \* this.weights[j]

      }

      console.log(sum)

    }

  }

}

const perceptron = new Perceptron()

perceptron.train(trainInputs)

***Outputs – modified formula***

4

3.9000000000000004

1.2999999999999998

1.7

1.6

Note – the last three were predicted to be less than 0 – They are not. So the model needs to be modified again.