Linear Classification

NYU K12 STEM Education: Machine Learning

Department of Electrical and Computer Engineering,
NYU Tandon School of Engineering
Brooklyn, New York

Course Details

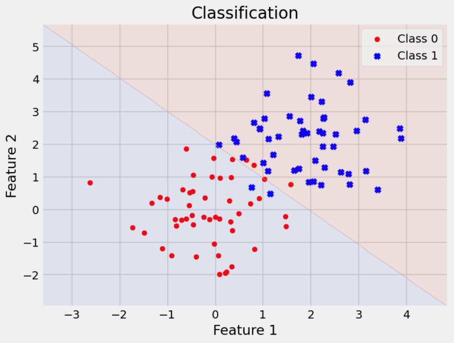
- Course Website
- Instructors:

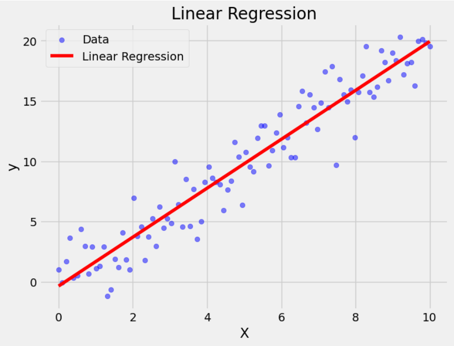
Rugved Mhatre rugved.mhatre@nyu.edu

Akshath Mahajan avm6288@nyu.edu

Outline

Classification vs. Regression

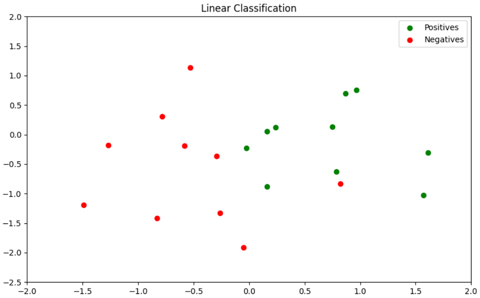




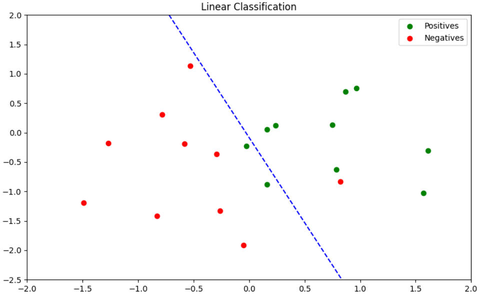
Classification

Given the dataset (x_i, y_i) for $i = 1, 2, \ldots, N$, find a function $f(x)$ (model) so that it can predict the label $\hat{y}$ for some input x, even if it is not in the dataset, i.e. $\hat{y} = f(x)$

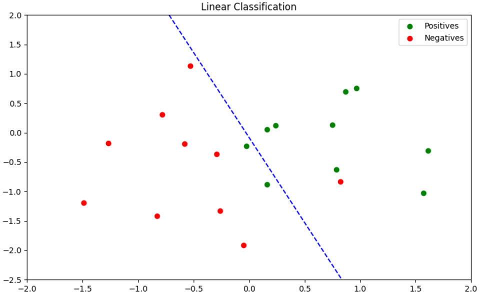
- Positive : $y = 1$
- Negative : $y = 0$



Decision Boundary



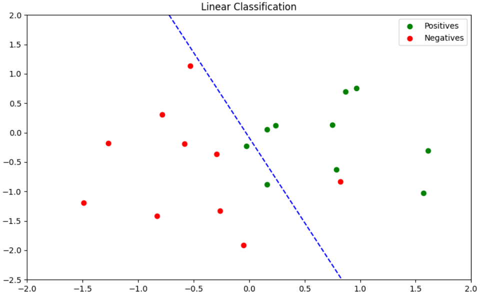
Decision Boundary



- Evaluation Metric:

$$\text{Accuracy} = \frac{\text{Number of correct prediction}}{\text{Total number of prediction}}$$

Decision Boundary

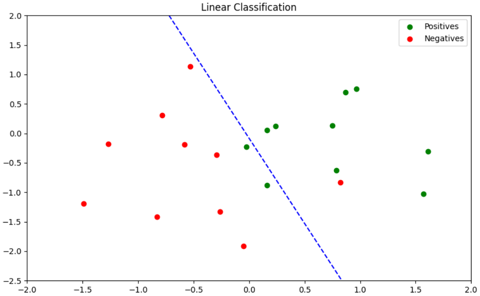


- Evaluation Metric:

$$\text{Accuracy} = \frac{\text{Number of correct prediction}}{\text{Total number of prediction}}$$

- What is the accuracy in this example?

Decision Boundary



$$\text{Accuracy} = \frac{\text{Number of correct prediction}}{\text{Total number of prediction}} = \frac{17}{20} = 0.85 = 85\%$$

Need for a new Model

- What would happen if we used the linear regression model:

$$\hat{y} = w_0 + w_1 x$$

Need for a new Model

- What would happen if we used the linear regression model:

$$\hat{y} = w_0 + w_1 x$$

- y is 0 or 1
- $\hat{y}$ will take any value between $-\infty$ and ∞

Need for a new Model

- What would happen if we used the linear regression model:

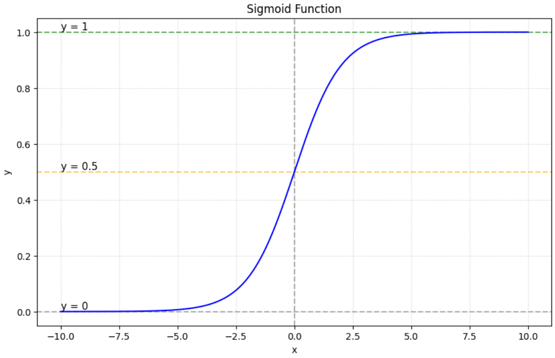
$$\hat{y} = w_0 + w_1 x$$

- y is 0 or 1
- $\hat{y}$ will take any value between $-\infty$ and ∞
- It will be hard to find w_0 and w_1 that make the prediction $\hat{y}$ match the label y

Sigmoid Function

By appling the sigmoid function, we enforce $0 \leq \hat{y} \leq 1$

$$\hat{y} = \text{sigmoid}(w_0 + w_1 x) = \frac{1}{1 + e^{-(w_0 + w_1 x)}}$$



A new loss function

- Binary Cross Entropy Loss:

$$\text{Loss} = \frac{1}{N}\sum_{i=1}^{N}\left[-y_i \log \hat{y}_i - (1-y_i)\log(1-\hat{y}_i)\right]$$

A new loss function

- Binary Cross Entropy Loss:

$$\text{Loss} = \frac{1}{N}\sum_{i=1}^{N}\left[-y_i \log \hat{y}_i - (1-y_i)\log(1-\hat{y}_i)\right]$$

- What happens if $y_i = 0$?

A new loss function

- Binary Cross Entropy Loss:

$$\text{Loss} = \frac{1}{N}\sum_{i=1}^{N}\left[-y_i \log \hat{y}_i - (1 - y_i)\log(1 - \hat{y}_i)\right]$$

- What happens if $y_i = 0$?

$$\left[-y_i \log \hat{y}_i - (1 - y_i)\log(1 - \hat{y}_i)\right] = -\log(1 - \hat{y}_i)$$

A new loss function

- Binary Cross Entropy Loss:

$$\text{Loss} = \frac{1}{N}\sum_{i=1}^{N}\left[-y_i \log \hat{y}_i - (1-y_i)\log(1-\hat{y}_i)\right]$$

- What happens if $y_i = 0$?

$$\left[-y_i \log \hat{y}_i - (1-y_i)\log(1-\hat{y}_i)\right] = -\log(1-\hat{y}_i)$$

- What happens if $y_i = 1$?

A new loss function

- Binary Cross Entropy Loss:

$$\text{Loss} = \frac{1}{N}\sum_{i=1}^{N}\left[-y_i \log \hat{y}_i - (1-y_i)\log(1-\hat{y}_i)\right]$$

- What happens if $y_i = 0$?

$$[-y_i \log \hat{y}_i - (1-y_i)\log(1-\hat{y}_i)] = -\log(1-\hat{y}_i)$$

- What happens if $y_i = 1$?

$$[-y_i \log \hat{y}_i - (1-y_i)\log(1-\hat{y}_i)] = -\log(\hat{y}_i)$$

MSE vs. Binary Cross Entropy Loss

- MSE of a logistic function has many local minima
- Binary Cross Entropy loss has only one minimum

Classifier

$$\hat{y} = \text{sigmoid}(w_0 + w_1 x) = \frac{1}{1 + e^{-(w_0 + w_1 x)}}$$

How to deal with uncertainty?

- Thanks to the sigmoid, $\hat{y} = f(x)$ is between 0 and 1

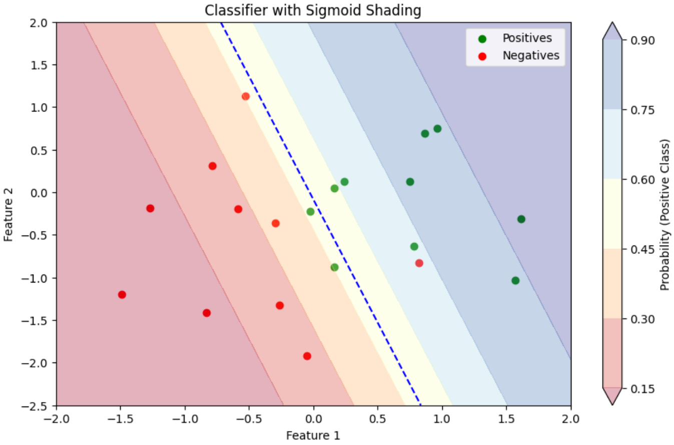
Classifier

$$\hat{y} = \text{sigmoid}(w_0 + w_1 x) = \frac{1}{1 + e^{-(w_0 + w_1 x)}}$$

How to deal with uncertainty?

- Thanks to the sigmoid, $\hat{y} = f(x)$ is between 0 and 1
- If $\hat{y}$ is close to 0, the data is probably negative
- If $\hat{y}$ is close to 1, the data is probably positive
- If $\hat{y}$ is around 0.5, we are not sure.

Classifier


Classifier with Sigmoid Shading
Positives
Negatives
Feature 1
Feature 2
Probability (Positive Class)
2.0
1.5
1.0
0.5
0.0
−0.5
−1.0
−1.5
−2.0
−2.5
0.90
0.75
0.60
0.45
0.30
0.15

Decision Boundary

- Once, we have a classifier outputting a score $0 < \hat{y} < 1$, we need to create a decision rule.

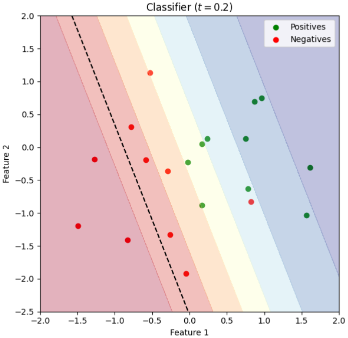
Decision Boundary

- Once, we have a classifier outputting a score $0 < \hat{y} < 1$, we need to create a decision rule.
- Let $0 < t < 1$ be a **threshold**:
  - If $\hat{y} > t$, $\hat{y}$ is classified as positive
  - If $\hat{y} < t$, $\hat{y}$ is classified as negative

Decision Boundary

- Once, we have a classifier outputting a score $0 < \hat{y} < 1$, we need to create a decision rule.
- Let $0 < t < 1$ be a **threshold**:
  - If $\hat{y} > t$, $\hat{y}$ is classified as positive
  - If $\hat{y} < t$, $\hat{y}$ is classified as negative
- How to choose t?

Decision Boundary



Performance metrics for a classifier

- Accuracy of a classifier: percentage of correct classification
- Why accuracy alone is not a good measure for assessing the model?

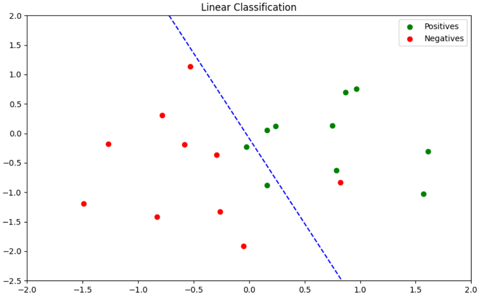
Performance metrics for a classifier

- Accuracy of a classifier: percentage of correct classification
- Why accuracy alone is not a good measure for assessing the model?
  - Example: A rare disease occurs 1 in ten thousand people
  - A test that classifies everyone as free of the disease can achieve 99.999% accuracy when tested with people drawn randomly from the entire population

Types of Errors in Classification

- Correct predictions:
  - True Positive (TP) : Predict $\hat{y} = 1$ when $y = 1$
  - True Negative (TN) : Predict $\hat{y} = 0$ when $y = 0$
- Two types of errors:
  - False Positive/ False Alarm (FP): $\hat{y} = 1$ when $y = 0$
  - False Negative/ Missed Detection (FN): $\hat{y} = 0$ when $y = 1$

Exercise



- How many True Positives (TP) are there?
- How many True Negatives (TN) are there?
- How many False Positives (FP) are there?
- How many False Negatives (FN) are there?

Other Metrics

- Sensitivity/Recall/TPR (How many positives are detected among all positive?)

$$\frac{\text{TP}}{\text{TP} + \text{FN}}$$

- Precision (How many detected positives are actually positive?)

$$\frac{\text{TP}}{\text{TP} + \text{FP}}$$

Outline

Diagnosing Breast Cancer

- We're going to use the breast cancer dataset to predict whether the patients' scans show a malignant tumour or a benign tumour.
- Let's try to find the best linear classifier using logistic regression.
- Open Diagnosing Breast Cancer Demo from Course Website

Outline

Multiclass Classification

- Previous Model:

$$f(x) = \sigma(\phi(x)w)$$

- Representing Multiple Classses:
  - One-hot / 1-of-K vectors, ex : 4 Class
  - Class 1 : $y = [1, 0, 0, 0]$
  - Class 2 : $y = [0, 1, 0, 0]$
  - Class 3 : $y = [0, 0, 1, 0]$
  - Class 4 : $y = [0, 0, 0, 1]$

Multiclass Classfication

- Multiple outputs:

$$f(x) = \text{softmax}(\phi(x)W)$$

- Shape of $\phi(x)W$: $(N, K) = (N, D) \times (D, K)$
- Softmax:

$$\text{softmax}(z_k) = \frac{e^{z_k}}{\sum_j e^{z_j}}$$

Softmax Example

$$z = \begin{bmatrix} -1 \\ 2 \\ 1 \\ -4 \end{bmatrix}$$

$$\text{softmax}(z) = \begin{bmatrix} \frac{e^{-1}}{e^{-1}+e^{2}+e^{1}+e^{-4}} \\ \frac{e^{2}}{e^{-1}+e^{2}+e^{1}+e^{-4}} \\ \frac{e^{1}}{e^{-1}+e^{2}+e^{1}+e^{-4}} \\ \frac{e^{-4}}{e^{-1}+e^{2}+e^{1}+e^{-4}} \end{bmatrix} \approx \begin{bmatrix} 0.035 \\ 0.704 \\ 0.259 \\ 0.002 \end{bmatrix}$$

Cross-Entropy

- Multple Outputs: $\hat{y_i} = \text{softmax}(\phi(x_i)W)$
- Cross-Entropy:

$$J(W) = -\sum_{i=1}^{N} \sum^{K} \_k = 1 y_{ik} \log(\hat{y_{ik}})$$

- Example, $K = 4$, if $y_i = [0, 0, 1, 0]$ then,

$$\sum_{k=1}^{K} y_{ik} \log(\hat{y_{ik}}) = \log(\hat{y_{i3}})$$

Outline

1. Linear Classification
2. Lab I
3. Multiclass Classification
4. Lab II

Iris Dataset

- Open Iris Dataset Demo from Course Website