MATH 462 LECTURE NOTES
PART 2: CLASSIFICATION

ADAM M. OBERMAN

CONTENTS

Date: September 28, 2021.

1. Introduction to binary classification

Reference for this section [Mur12, Chapter 8] (mostly the first equation) or [Mur22, Section 5.1.2].

1.1. **Binary classification.** In the general classification problem, the target set $\mathscr{Y}$ is a set of discrete labels. Here we consider the case of binary classification consisting, so there are two labels, which we denote by $-1, +1$, and we write

$$\mathscr{Y} = \mathscr{Y}_{\pm} = \{-1, +1\}$$

We are given a dataset (S_m) consisting of m pairs of (x_i, y_i), $i = 1, \dots, m$, of data, $x_i \in \mathscr{X}$ and labels, $y_i \in \mathscr{Y}$,

$$S_m = \{(x_1, y_1), \dots, (x_m, y_m)\} \tag{S_m}$$

The standard binary classification loss is the zero-one loss,

$$\ell_{0,1}(c, y) = \begin{cases} 0 & c = y \\ 1 & \text{otherwise} \end{cases} \tag{$\ell_{0,1}$}$$

1.2. **Direct classification vs surrogate losses.** The direct classification method is to define a family of classifiers

$$\mathscr{H} = \{c_w : \mathscr{X} \to \mathscr{Y}_{\pm} \mid w \in \mathscr{W}\}$$

where $\mathscr{W}$ can be continuous or discrete.

For a given classification function, $c_w(x)$ and a given dataset, (S_m), the empirical error of c_w on (S_m) is simply given by

$$\widehat{L}_{0-1}(c_w) = \frac{1}{m} \sum_{i=1}^{m} \ell_{0-1}(c(x_i), y_i) \tag{EE}$$

Direct method of classification (abstract)

Inputs:

- Dataset (S_m) with $y_i \in \mathscr{Y}_{\pm} = \{-1, +1\}$ and $x_i \in \mathbb{R}^d$ features.
- Hypothesis class of models $c_w(x) : \mathbb{R}^d \to \mathscr{Y}_{\pm}$, parameterized by w.

Goal:

- Given x, predict $y = c(x)$

Method:

- rule based
- (or) minimize the empirical zero-one loss (EE).

In special cases, or when there is extra structure present, the direct classification method can be effective. We give an example below, when a binning function is available.

(Vote) Bin the classes and use a majority classifier in each bin. The binning can happen in various ways (e.g. cluster), in our context, it is a black box function.

The (Vote) method is presented mainly to develop certain ideas we will use in the sequel. However, direct classification is not our primary interest, due to challenges with the direct methods (which are identified in more detail in advanced textbooks).

The alternative is to consider an indirect classification methods.

Direct classification does not scale

The direct classification methods works in special cases, but it does not scale, because the 0-1 loss is not amenable to optimization.

Exercise 1.1. *Make a simple argument based on continuity why* (EE) *is difficult to minimize for large scale problems. (For example, when $\mathscr{X}$ is discrete, with m elements, how many possible classifiers are there? Is direct search efficient? Is there an obvious way to minimizer the loss (answer: no).*

1.3. **Indirect classsification approaches: scores and probabilities.** The main approaches to (supervised) binary classification we study are:

(Score) Score based. Learn a score function from features. The class is determined by the sign of the score. The main example is Support Vector Machines [DFO20].

(Prob) Probability based. Learn a probability function from features. The class is determined by the probability. The main example is logistic regression.

Hypotheses consisting of a parameterized family of models,

$$\mathscr{H} = \{h_w : \mathscr{X} \to \mathbb{R} \mid w \in \mathscr{W}\}$$

where $\mathscr{W}$ can be continuous or discrete.

The standard hypothesis class in this setting:

Example 1.1. The family of linear functions

$$h_w(x) = w \cdot x = \sum_{i=1}^{d} w_i x_i, \quad w \in \mathbb{R}^d \tag{H-lin}$$

We need to use a *surrogate loss*, define over real-values hypotheses. There is an added complication that our classification losses will be defined on the pair (s,c) for $s \in \mathbb{R}, c \in \mathscr{Y}$.

$$\ell_{class} : \mathbb{R} \times \mathscr{Y} \to \mathbb{R}^+ \tag{ℓ_{class}}$$

Usually we have the additional requirement that the loss is convex, according to the following definition.

Definition 1.2 (convex classification loss)**.** The loss, (ℓ_{class}) is convex if $\ell_{class}(s,y)$ is be convex as a function of s for every $y \in \mathscr{Y}_{\pm}$.

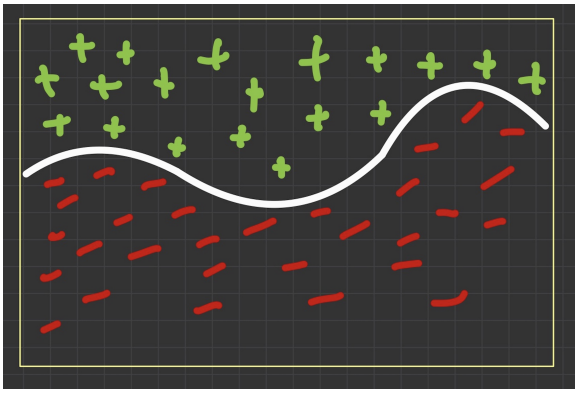

FIGURE 1. Classification problem: nonlinear feature map leads to a nonlinear boundary. However, the model $h_w(x) = w \cdot f(x)$ is still linear as a function of w. White line: the classification boundary $h_w(x) = 0$.

Define the empirical loss of the model, h_w, by

$$\widehat{L}_{class}(h_w) = \frac{1}{m}\sum_{i=1}^{m} \ell_{class}(h_w(x_i), y_i) \tag{EL-C}$$

The following framework applied to both (Score) and (Prob).

Binary Classification via loss minimization (abstract)

Inputs:

- Dataset (S_m) with $y_i \in \mathscr{Y}_{\pm} = \{-1, +1\}$ and $x_i \in \mathbb{R}^d$ features.
- Hypothesis class of models $h_w(x) : \mathbb{R}^d \to \mathbb{R}$, parameterized by w.
- Classifier: $c : \mathbb{R} \to \mathscr{Y}_{\pm}$, a rule which converts the model value to a class.
- Convex classification loss, (ℓ_{class}).

Goal:

- Given x, predict $y = c(x)$.

Method:

- Minimize (EL-C) to find the model which best fits the data, h_{w^*}.
- Set $y = c(h_{w^*}(x))$.

1.4. **Discussion: nonlinear features.** Note that this setting is quite general, since the notation obscures the fact that we include the case where there is a nonlinear feature map $f(x)$, and the model is $h_w(x) = w \cdot f(x) = \sum_{i=1}^{d} w_i f_i(x)$.

Example 1.3. We can have raw data $x \in \mathbb{R}$ and features $f_i(x) = x^i$, for $i = 1, \ldots, d$, which corresponds to fitting one dimensional data with a polynomial. Here we are looking for a nonlinear classification boundary. See Figure 1.

1.5. **Errors from losses.** However, there is something new here that we don't see in the case of regression (where $\mathscr{Y} = \mathbb{R}$). We need to check that our loss minimization (which is defined $h \in \mathbb{R}$) results in an effective *classification*. In other words we care about the average classification error (the 0-1 loss, defined below).

We need to characterize the effect of the loss on the errors. There are two approaches to this question: (i) we can prove *a priori* the properties of the classifier, or (ii) we can simply check *a posteriori* that the results are good. The purpose of understanding classification losses is to achieve the first goal.

Reality check: did it work?

Inputs:

- Dataset (S_m) with $y_i \in \mathscr{Y}_\pm = \{-1, +1\}$ and $x_i \in \mathbb{R}^d$ features.
- Model h_{w^*} which minimizes the empirical loss (EL-C).
- Classifier: $c : \mathbb{R} \to \mathscr{Y}_\pm$, a rule which converts the model value to a class.

Goal:

- Estimate the empirical error, (EE) of the classifier $c(h_{w^*})$.

One possible answer is given by the following idea.

Definition 1.4. Given the triple $(\ell_{class}, c, C_{class})$ consisting of a classification loss, a classification map and a constant $C_{class} > 0$. The triplet is an upper bound for the classification error, if

$$\ell_{class}(h, y) \geq C_{class}\, \ell_{0-1}(c(h), y), \quad \text{for all } h \in \mathbb{R},\ y \in \mathscr{Y}_\pm \tag{LvE}$$

We say the loss is an upper bound for the error when the constant and the classifier are understood.

Theorem 1.5. *Suppose $(\ell_{class}, c, C_{class})$ is an upper bound for the error. Show that for any function $h : \mathscr{X} \to \mathscr{Y}_\pm$, and any dataset S_m*

$$\widehat{L}_{0-1}(c(h)) \leq \frac{1}{C_{class}} \widehat{L}_{class}(h)$$

In particular, the bound above holds for a minimizer h_{w^*} of $\widehat{L}_{class}(h)$.

We summarize the application of the error bounds as follows.

Error bounds on the minimizer

Inputs:

- Model h_{w^*} which minimizes the empirical loss (EL-C).
- Classifier: $c : \mathbb{R} \to \mathscr{Y}_\pm$, a rule which converts the model value to a class.
- Classification loss ℓ_{class}, which (along with the constant C_{class}) is an upper bound for the error

Outputs:

- The empirical error, (EE), is bounded by $\frac{1}{C_{class}} \widehat{L}_{class}(h)$

Exercise 1.2. *Prove Theorem 1.5.*

Exercise 1.3. *In this exercise, use* c_{sgn}. *(i) Is the loss* $\ell(h,y) = (h-y)^2$ *an upper bound for the zero one loss? If so, what is the best constant. (ii) Show that* $\ell(h,y) = |h+y|$ *is not an upper bound for the zero one loss.*

(iii) Given the function $\ell(h,y)$*, suppose there is an* $h < 0$ *with* $\ell(h,1) = 0$*. Show this function cannot be an upper bound for the zero one loss.*

(iv) Can you find a simple converse?

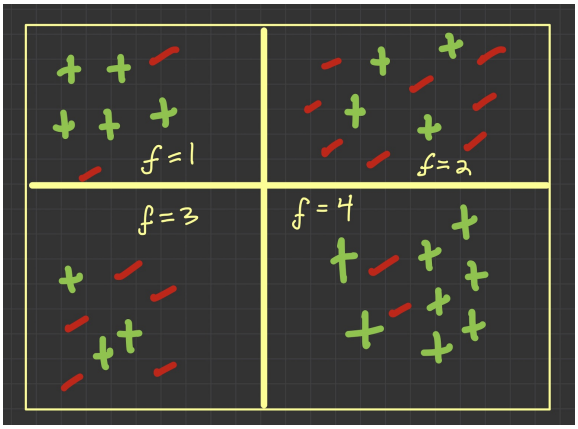

FIGURE 2. Illustration of majority rule classification problem. Four bins (yellow squares), corresponding to $f(x) = 1,2,3,4$. The labelled data (x_i, y_i) is represented by the symbol:+ when $y_i = +1$ and (-) when $y_i = -1$. For example, the $f(x) = 1$ bin has 5 data points with $y = +1$ and two with $y = -1$. The fraction, p_i, of positive examples in bin i, is $p_1 = 5/7, p_2 = 4/12, p_3 = 3/8, p_4 = 8/10$.

2. MAJORITY CLASSIFIERS

We start with a very simple setting. In this setting, we assume that we are given a binning map, which puts similar samples x into the same bins. For now, this is a black box map, in the sense that we do not concern ourselves with how it was obtained. Refer to Figure 2.

2.1. **Class probabilities in bins.**

Definition 2.1. Let $f : \mathscr{X} \to \{1,2,\ldots,N\}$ be a discrete valued function. The set

$$B_j = \{x \in \mathscr{X} \mid f(x) = j\}$$

form a partition of $\mathscr{X}$, which we call bins, we call f a binning map. Given (S_m), define

$$p_j = \frac{|\{y = +1 \mid x \in B_j\}|}{|B_j|} \quad \text{for } (x,y) \in S_m \tag{CP}$$

to be the fraction of positive examples from the dataset in bin j.

Define the bin hypothesis class to be functions $c : \mathscr{X} \to \mathscr{Y}_\pm$ which are constant on bins, parameterized by $w \in \mathscr{Y}^N$,

$$\mathscr{H}_{bin} = \{c_w(x) \mid c_w(x) = w_{f(x)},\ w_j \in \{-1,+1\}\}$$

Definition 2.2. Define the majority classifier to be the map $c : [0,1] \to \mathscr{Y}_\pm$ given by

$$c_{maj}(p) = \operatorname{sgn}(p - .5)$$

Given the dataset (S_m), we can define a simple classifier by majority rule.

Majority rule classifier algorithm

Inputs:

- Dataset (S_m). $y \in \mathscr{Y}_{\pm}$ (i.e. Binary classification)
- Binning map $f : \mathscr{X} \to [1,\dots,N]$ which maps data, x, one of N bins

Goal:

- A simple rule for classification on each bin

Method:

- Find the p_i, the fraction of positive labels in each bin
- Set $c(x) = c_{maj}(p_{f(x)})$ to be the majority classifier for the bin.

Example 2.3. In Figure 2, setting $w = (-1,+1,+1,+1)$ results in $h_w(x) = -1$ if $f(x) = 1$, in other words, if x is in the first bin, and $h_w(x) = +1$ otherwise.

Example 2.4. Consider the example of Figure 2. We consider (EE) with the bin hypothesis class. So the model is constant on each of the four bins. Majority rule for each bin corresponds to

$$c(x) = \begin{cases} +1 \text{ (green +)}, & f(x) = 1,4 \\ -1 \text{ (red -)}, & f(x) = 2,3 \end{cases}$$

The bin error count is: 2, 4, 3, 2, for bins 1, 2, 3, 4, respectively.

2.2. **Zero-one loss minimization for majority rule.** Here we show that we can obtain the majority rule classifier by loss minimization using the 0-1 loss $(\ell_{0,1})$

Theorem 2.5. *Consider* (EL-C) *with the zero-one loss* $(\ell_{0,1})$*. Given a binning map, f, the majority rule classifier is the minimizer of* (EE) *over the binning hypothesis class, $\mathscr{H}_{bin}$.*

$$\min_{c_w \in \mathscr{H}_{bin}} \widehat{L}_{0-1}(c_w)$$

Proof. Write

$$\widehat{L}_{0-1}(c_w) = \frac{1}{m}\sum_{i=1}^{m} \ell_{0-1}(w_{f(x_i)}, y_i) \tag{EL-01}$$

We can separate the problem into each bin

$$\widehat{L}_{0-1}(c_w) = \frac{1}{m}\sum_j \widehat{L}_j(c_w)$$

where

$$\widehat{L}_j(c_w) = \frac{1}{m}\sum_{(x,y)\in B_j} \ell_{0-1}(w_j, y)$$

Using the definition (CP), the last sum is given by

$$\widehat{L}_j(c_w) = p_j \ell_{0-1}(w_j,+1) + (1-p_j)\ell_{0-1}(w_j,-1)$$

where we have summed over the positive and negative labels. Thus we want to

$$\min_{w_j \in \mathscr{Y}_\pm} L_j(c_w)$$

There are two cases to check, and clearly the minimum is when we choose the label with more examples, which corresponds to the majority classifier. □

Binary classifier via loss minimization (binning approach)

Inputs:

- Dataset (S_m). $y \in \mathscr{Y}_\pm$ (binary classification)
- Binning map $f : \mathscr{X} \to [f_1, \ldots, f_N]$ which maps data, x, one of N bins

Model:

- Hypothesis class of models $c(x) = c(f(x))$, which are constant on each bin.

Goal:

- Given x, find $y(x)$.

Loss minimization method:

- Minimize the expected zero-one loss (EL-01) over the hypothesis class. The minimizer $c^*(x)$ corresponds to majority rule.

3. Cost-sensitive classification

In the binary classification case, we may have a preference for different error types. We define the error types as follows.

Definition 3.1. For a given x, let $c \in \mathscr{Y}_\pm$ be the model class prediction, and $y \in \mathscr{Y}_\pm$ be the true label. Suppose the model is wrong, i.e., $c \neq y$. Define the error types,

- False positive when $c = 1$ and $y = -1$,
- False Negative when $c = -1$, $y = +1$.

We also use the terms true positive and true negative in the other cases.

3.1. Important function : the odds ratio.

The odds ratio is a useful quantity when comparing probabilities.

Definition 3.2. Given $p \in (0,1)$ the odds ratio is $r(p) = \frac{p}{1-p}$. Given $r \in \mathbb{R}^+$, define $p(r) = r/(r+1)$.

Exercise 3.1. *Show that the odds ratio function $r(p)$ is invertible on $\mathbb{R}^+$ with inverse $p(r) = r/(r+1)$.*

Important function: odds ratio

- The odds ratio $r(p) = p/(1-p)$ maps probabilities to numbers.
- The inverse of the odds ratio is $p(r) = r/(r+1)$.

Exercise 3.2. *Bets payoffs are usually given by the odds, in the sense that a bet with odds* $3:2$ *(which is* $r = 3/2$ *in our notation), means if you make the bet, and pay* 2 *if you win the bet you receive a payoff of* 3*. The odds imply a corresponding probability of winning the bet. What is the formula for* p *in terms of the odds?*

Conversely, if an event occurs with probability p*, what are the corresponding fair odds for a bet on the event.*

Example 3.3. Suppose we are classifying a disease, and a false negative is considered to be 10 times worse than a false positive. Then, in other words, we want the classification threshold to be when the odds ratio $r(p) = 10$, or when the probability $p(r) = 10/11$.

3.2. **Odds ratio classifier.** Define the odds ratio classifier as

$$c_{odds}(p, r_0) = \begin{cases} +1 & r(p) \geq r_0 \\ -1 & ow \end{cases}, \quad \text{where } r(p) = \frac{p}{1-p} \tag{1}$$

We summarize as follows.

Odds ratio classifier

Inputs:

- Dataset (S_m). $y \in \mathscr{Y}_{\pm}$ (i.e. Binary classification)
- Binning map $f : \mathscr{X} \to [f_1, \ldots, f_N]$ which maps data, x, one of N bins
- A given odds ratio, r, reflecting the desired ratio of false negatives to false positives.

Goal:

- A simple classifier on each bin which results in a ratio r of false negatives to false positives.

Method:

- Find the p_i, the fraction of positive labels in each bin
- Set $c(x) = c_{odds}(p_{f(x)})$ to be the odds ratio classifier for the bin.

Exercise 3.3. *For each of the bins in Figure 2, determine the odds ratio* $p/(1-p)$*. Determine the smallest value of* r *needed to make: (i) all the bins positive (ii) two of the bins positive (iii) only one bin positive.*

3.3. **Loss minimization for odds ratio classifier.** Let r be a given odds ratio. Here we show that the odds ratio classifier can be obtained using minimization of the form (EL-C), for the following loss.

Define the cost-sensitive binary classification loss to be

$$\ell_{0-r}(c, y) = \begin{cases} r & c = 1,\ y = -1 \\ 1 & c = -1,\ y = 1 \\ 0 & \text{otherwise} \end{cases} \tag{ℓ_{0-r}}$$

Then (EL-C) with the cost-sensitive loss (ℓ_{0-r}) can be written

$$\widehat{L}_{0-r}(c) = \frac{1}{m}\sum_{i=1}^{m} \ell_{0-r}(c(x_i), y_i) \tag{EL-0r}$$

Define the bin odds ratio classifier (1) by

$$c^*(x) = c_{odds}(p_{f(x)}, r)$$

Theorem 3.4. *Consider the loss (ℓ_{0-r}) in the context of a binning map and the bin hypothesis class defined above. Let p_j be the fraction of positive examples, in bin j, given by* (CP)*. Then*

$$\arg\min_{c_w \in \mathscr{H}_{bin}} \widehat{L}_{0-r}(c) = c^*$$

Proof. The key steps in the proof:

1. The first step is the same as in Theorem 2.5. We can separate the sum in (EL-0r) into each bin.

2. The second step is to establish, in a given bin with bin probability p_i, that the loss minimizer is given by (1). Following the same steps as in Theorem 2.5, fixing a bin, B_j, and summing those terms in (EL-0r), we arrive at

$$\widehat{L}_j(c) = p_i \ell_{0-r}(c_j, +1) + (1-p_i)\ell_{0-r}(c_j, -1)$$

There are two cases to check, which corresponds to $c_i = 1, -1$. Thus we get

$$\min((1-p_i)r, p_i)$$

Which are equal when $r(p_i) = p_i/(1-p_i) = r$. Thus if $r(p_i) \geq r$ we should set $c_i = 1$ (otherwise set it to be -1. This corresponds to the odds ratio classifier (1), as desired. □

Odds ratio classifier via loss minimization

Inputs:

- Dataset (S_m). $y \in \mathscr{Y}_{\pm}$ (i.e. Binary classification)
- Binning map f which maps data, x, in bins.

Model:

- hypothesis class of models which are constant on each bin.

Goal:

- Find the odds ratio classifier on each bin by loss minimization.

Loss minimization method:

- Minimize the expected loss (EL-0r) over the hypothesis class.

4. Score based losses

4.1. **Introduction.** There is more than one way to define a classification based on scores.

Example 4.1 (Grading). Consider the classification problem of converting a grade, $x \in [0,100]$ in one of $K = 5$ letter grades F, D, B, C, A. We can use an absolute rule, e.g. $x \in [85,100]$ converts to A, or we can grade on the curve: which means having a fixed percentage of the students in each grade.

In each case, the outcomes are difference and there are arguments for and against each method. For example, if a class is particularly strong compared to other classes, the students are penalized by grading on a curve.

Binary Classification via loss minimization (abstract score-based approach)

Inputs:

- Dataset (S_m) with $y_i \in \mathscr{Y}_\pm = \{-1,+1\}$ and $x_i \in \mathbb{R}^d$ features.
- Hypothesis class of score models $s_w(x) : \mathbb{R}^d \to \mathbb{R}$, parameterized by w.

Goal:

- Given x, find $y(x)$

Output:

- The model $s_{w^*}(x)$ maps to $y = c(s_{w^*}(x))$

Method:

- Find w^* by minimizing (EL-S) using a score-based loss.

(EL-S)
$$\widehat{L}_{score}(s_w) = \frac{1}{m}\sum_{i=1}^{m} \ell_{score}(s_w(x_i), y_i)$$

4.2. **Linear models and features.** In this setting, for $x \in \mathbb{R}^d$, we consider the hypothesis class consisting of linear models

$$\mathscr{H}_{score} = \{s_w(x) \mid s_w(x) = w \cdot x + w_0\}$$

(Later can absorb w into features, but it's here for emphasis now).

Here $s_w(x)$ corresponds to the score of x. We want higher scores to corresponds to higher probability of correct classification, as in Figure 3.

4.3. **The standard score-based loss.** Now we define a score-based loss which is piecewise differentiable as a function of s. We need to use a scoring loss.

Given the score s, we define the classifier

(2)
$$c_{\text{sgn}}(s) = \text{sgn}(s)$$

The main loss we study is the standard margin (or hinge) loss.

(3)
$$\ell_{margin}(s,y) = \begin{cases} 0 & sy \geq 1 \\ |s-1| & 0 \leq sy \leq t \\ 1+|s| & sy \leq 0 \end{cases}$$

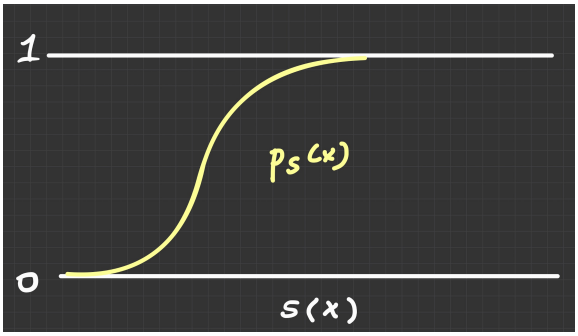

FIGURE 3. Illustration of a score function

Define the following error types

Definition 4.2. Given $y \in \mathscr{Y}_{\pm}$, and $s \in \mathbb{R}$, and $c(s)$ given by (2). Define the pair (y,s) to be

- incorrect: $c(s) \neq y$
- marginal positive $y = 1$, $0 \leq s \leq 1$
- marginal negative $y- = 1$, $-1 \leq s \leq 0$
- marginal if $y = c(s)$ and $|s| \leq 1$
- confident: $c(s) = y$ and $|s| \geq 1$

In this case, (EL-S) becomes

(ELM) $$\widehat{L}_{margin}(s_w) = \frac{1}{m}\sum_{i=1}^{m} \ell_{margin}(h_w(x_i), y_i)$$

We summarize the standard approach as follows

Binary classification via standard margin loss

Inputs:

- Dataset (S_m) with $y_i \in \mathscr{Y}_{\pm} = \{-1,+1\}$ and $x_i \in \mathbb{R}^d$ features.
- Hypothesis class of linear score models $s_w(x) = w \cdot x$
- Standard margin loss (3)
- Standard classifier $c_{\text{sgn}}(s) = \text{sgn}(s)$

Goal:

- Given x, find $y(x)$

Output:

- The model $s_w(x)$ maps to $y = c_{\text{sgn}}(s_w(x))$

Method:

- Find w^* by minimizing (ELM)

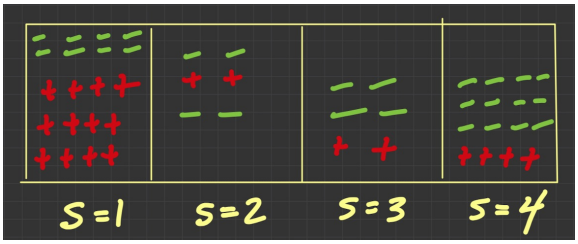

FIGURE 4. Score based classification example

4.4. **Error bounds for margin loss.** We have the following result, which follows from Theorem 1.5. Given any function $h : \mathscr{X} \to \mathscr{Y}_{\pm}$, and any dataset S_m

$$\widehat{L}_{margin}(s) \geq \widehat{L}_{0-1}(c_{\text{sgn}}(s)) \tag{4}$$

The last results leads to the following error bounds

Error bounds for margin classification

Inputs:

- Model h_w which minimizes the empirical loss (ELM).

Outputs:

- The empirical error, (EE), is bounded by the empirical loss $\widehat{L}_{margin}(h_w)$

4.5. **Exercises.**

Exercise 4.1. *Consider the example of score-based classification illustrated in Figure 4. Find the minimizer of* (EL-S) *with the score-based absolute value loss, using the threshold classifier.*

Compare to the majority classifier.

Show that in Figure 4, if we relabel the scores from $1,2,3,4$ *to any other non-decreasing values (e.g. try* $10,15,20,25$*), and use the absolute value loss, we get the same classifier. (Hint: can check this directly or use the condition for a minimizer).*

Exercise 4.2. *Show that with the margin loss* (3)*, the cases in Theorem 4.2 correspond to*

$$\ell_{margin}(s,y) \begin{cases} [1,\infty) & \textit{incorrect} \\ \in [0,1] & \textit{marginal} \\ = 0 & \textit{confident} \end{cases}$$

Exercise 4.3. *Show that* (LvE) *holds for the* $\ell_{margin-t}$ *with* $C_{class} = 1$ *and the* $c = \text{sgn}$ *classifier. Justify* (4).

Definition 4.3. Given a threshold $t \geq 0$. Define the t-margin loss,

$$\ell_{margin,t}(s,y) = \begin{cases} 0 & sy \geq t \\ |s/t - 1| & 0 \leq sy \leq t \\ 1 + |s|/t & sy \leq 0 \end{cases} \tag{5}$$

Exercise 4.4. *(i) Show that setting $t = 1$ in* (5) *recovers that standard margin loss. (ii) Generalize the definitions of the error types Theorem 4.2.*

Exercise 4.5. *Plot the loss* (5) *for $y = 1$ and $t > 1$. Show symmetry of loss $\ell_{margin,t}(-s,-y) = \ell(s,y)$. Use this to plot loss for $y = -1$.*

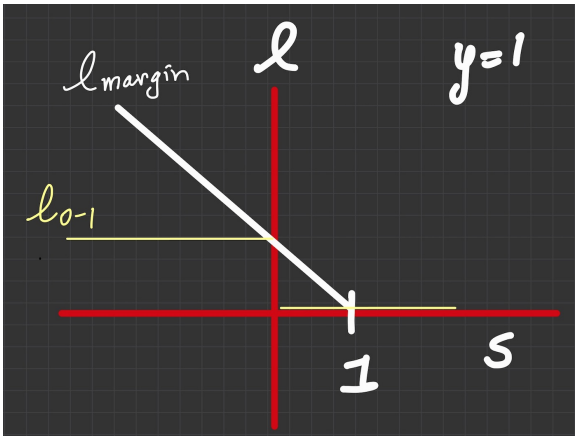

FIGURE 5. Margin loss, this loss is differentiable except at corner, and lies above the 0-1 loss

4.6. **Additional analysis of the loss.** In this example, we consider a dataset of scalar data (scores) and labels.

$$S_m = \{(x_1,y_1),\ldots,(x_m,y_m)\}, \qquad x_i \in \mathbb{R} \tag{6}$$

In order to analyze the loss.

Theorem 4.4. *Consider minimizing* (EL-S) *with the margin loss* (5) *over the dataset* (6) *with the threshold model $s(x) = x - w$ and the classifier $c(s) = \operatorname{sgn}(s)$. Let E_p be the number of false or marginal positives. Let E_n be the number of false or marginal negatives. A sufficient condition for a minimizer w^* is that*

$$E_n = E_p$$

Proof. 1. Differentiate (EL-S) and set $\widehat{L}'(w) = 0$. 2. Each term in the derivative is either (i) zero (for a confident correct) or (ii) equal to ± 1, depending on cases of false/marginal positive or false/marginal negative.

[[details in handwritten class notes, to be filled in]] This leads to

$$\sum_{FP/MP} 1 = \sum_{FN/MP} 1$$

So the w^* is the threshold which $E_n = E_p$. □

Exercise 4.6. *Fill in the details of the proof of Theorem 4.4*

More generally, we can choose the ratio of false positives to false negatives using the the following generalization of the absolute error loss

(LAC-FP)
$$\ell_{abs}(s,y) = \begin{cases} 0 & \text{sgn}(s) = y \\ |y-s| & y = 1, \text{sgn}(s) = -1 \\ C|y-s| & y = -1, \text{sgn}(s) = +1 \end{cases}$$

Exercise 4.7. *Generalize Theorem 4.4 to the case of* (LAC-FP)*. Find the value of C which leads to* $FP = 10FN$

5. Probability based losses

In this section we study how to learn the class probabilities using a linear model. We define the model and the losses first, and then we give an interpretation.

5.1. Important function : the logistic function.

Definition 5.1. The logistic function $\sigma : \mathbb{R} \to (0,1)$, $\sigma(x) = \frac{1}{1+\exp(-x)}$ maps numbers to probabilities. The logit function, $\text{logit}(p) = \log\left(\frac{p}{1-p}\right)$. In this context, the numbers coming from a probability are called *logits*. See Figure 3 for a sketch of σ.

Exercise 5.1. *Show that (i)* $\sigma(x) = p(\exp(x))$*, and* $\text{logit}(p) = \log(r(p))$*. (ii)* σ *and* logit *are inverses. Hint: use the facts that the odds ratio,* $r(p)$ *and* $p(r) = r/(r+1)$ *are inverses,* exp *and* log *are inverses, and use part (i).*

Exercise 5.2. *Verify the following properties for* $\sigma(x)$*.*

- $2\sigma(x) = 1 + tanh(x/2)$
- $1 - \sigma(x) = \sigma(-x)$ *(so* $\sigma(x) - 1/2$ *is an odd function)*
- $\sigma'(x) = \sigma(x)(1-\sigma(x))$

Important function: logistic

- The function $\sigma(x) = \frac{1}{1+\exp(-x)}$ maps $\mathbb{R}$ to $[0,1]$
- The inverse is given by $\text{logit}(p) = \log\left(\frac{p}{1-p}\right)$.

5.2. **Model definition.** For binary classification, the standard choice is the log loss. Thus we take

$$\ell_{\text{Prob}}(p,y) = \ell_{\log}(p,y)$$

The expected loss, (EL-C), becomes

$$\widehat{L}_{\log}(p_w) = \frac{1}{m}\sum_{i=1}^{m} \ell_{\log}(p_w(x_i),y_i) \tag{EL-PC}$$

Since $p \in [0,1]$, the corresponding classifier is given by (7)

$$c_{maj}(p) = \operatorname{sgn}(p-.5) \tag{7}$$

In this setting, for $x \in \mathbb{R}^d$, we consider the linear model (H-lin) composed with the logistic function.

$$\mathscr{H}_{\text{Prob}} = \{p_w(x) \mid p_w(x) = \sigma(w\cdot x)\}$$

Binary Classification via loss minimization (Probability approach)

Inputs:
- Dataset (S_m) with $y_i \in \mathscr{Y}_{\pm} = \{-1,+1\}$ and $x_i \in \mathbb{R}^d$ features.
- Hypothesis class of models $p_w(x) = \sigma(w\cdot x)$ (σ composed with linear).

Goal:
- Given x, find $c(x)$

Output:
- $y = c_{maj}(p_{w^*}(x))$

Method:
- Minimize (EL-PC).

5.3. **Error bounds on the log classification loss.**

Definition 5.2. Define the score-based loss coming from the log loss by composition

$$\ell_{score,\log}(h,y) = -\ell_{\log}(\sigma(h),y) \tag{8}$$

Exercise 5.3. *Explain how* (8) *re-interpreting the probability classifier as a score-based classifier. Show that this loss, along with the corresponding classifier $c(h) = round(\sigma(h)$, is an upper bound for the error. Find the best constant C_{class}. Specialize the result of Theorem 1.5 to this case.*

As a result of the exercise, we have

Error bounds for margin classification

Inputs:

- Model h_w so that $\sigma(h_w)$ minimizes the log loss (EL-PC), and the value of the empirical loss $\widehat{L}_{\log}(h_w)$

Outputs:

- The empirical error, (EE), is bounded by $\frac{1}{C_{class}}\widehat{L}_{\log}(h_w)$

5.4. **Discussion.** Here we avoided all discussion of probabilistic models, as in `https://en.wikipedia.org/wiki/Logistic_regression` and instead re-interpreted the log loss with σ as a score-based model. This allows us to prove classification error bounds based using the empirical loss.

Two natural questions are left unanswered for now

(1) Is there an interpretation of $p_w(x)$ in terms of a probability of the class?
(2) Can we understand the difference between the two classification models (score with margin loss versus log loss with σ?

5.5. **Additional analysis of log loss.**

$$S_m = \{(x_1, y_1), \dots, (x_m, y_m)\}, \qquad x_i \in \mathbb{R} \tag{9}$$

In order to analyze the loss.

Define, using S_m, $J^+ = \{j \in 1, \dots, m \mid y_j = 1\}$ and $J^- = \{j \in 1, \dots, m \mid y_j = -1\}$ Define, for a label y and a probability p (of label $+1$), the error in the probability

$$e(p, y) = \begin{cases} 1 - p & y = 1 \\ p & y = -1 \end{cases}$$

Theorem 5.3. *Consider minimizing* (EL-PC) *over the dataset* (9) *with the threshold model* $p(x) = \sigma(x - w)$ *and the classifier* $c_{mag}(p)$. *A sufficient condition for a minimizer* w^* *is that*

$$\sum_{j \in J^+} e(p_j, 1) = \sum_{j \in J^-} e(p_j, -1)$$

Interpretation:

> The sum of the errors in the positive examples is equal to the sum of the errors over the negative examples

Proof. The main calculation is to minimizer (EL-PC) over this model. We can write

$$\widehat{L}_{\log}(p_w) = \frac{1}{m} \sum_{j \in J^+} -\log(\sigma(x_j - w)) + \frac{1}{m} \sum_{j \in J^-} -\log(1 - \sigma(x_j - w))$$

Differentiate and use $\sigma' = \sigma(1 - \sigma)$ to obtain, at a minimizer

$$0 = \frac{1}{m} \sum_{j \in J^+} (1 - \sigma(x_j - w)) - \frac{1}{m} \sum_{j \in J^-} \sigma(x_j - w)$$

which gives the result. □

Exercise 5.4. *Complete the details of the proof of Theorem 5.3.*

6. Proper Scoring rules: how to learn a probability of an event

Proper scoring rules are a type of loss used to learn the probability of an event. Their use predates machine learning: they were used to score forecasters for sporting events and weather. So the terminology is different from ML terminology.

Reference: [GR07].

6.1. Loss design to learn a probability from samples.

Here we consider the problem

Loss for probabilities: Problem definition

Inputs:

- A data set of the form (10)

Goal:

- Use differentiable loss minimization to find $q(S_m)$.

Although this seems like a simple problem, we will make a definition for the type of losses which work.

Given a dataset consisting only of labels,

$$S_m = \{y_1, \dots, y_m\}, \quad \text{where } y_i \in \mathscr{Y}_\pm = \{-1, +1\} \tag{10}$$

Define

$$q(S_m) = \text{ the fraction of positive labels in } S_m.$$

to be

Given a loss of the form

$$\ell_{\text{Prob}} : [0,1] \times \mathscr{Y}_\pm \to \mathbb{R}^+ \tag{11}$$

define

$$\widehat{L}(p) = \frac{1}{m} \sum_{i=1}^{m} \ell_{\text{Prob}}(p, y_i) \tag{EL-P}$$

Definition 6.1 (proper losses for learning probabilities)**.** Given a loss of the form (11). The loss is *proper* (in the sense of [GR07]) if for every S_m of the form (10),

$$\arg\min_{p \in [0,1]} \widehat{L}(p) = q(S_m)$$

otherwise the loss is improper.

We consider the following losses.

Definition 6.2 (Candidate proper losses)**.** To simplify notation, write $y^+ = \max(y,0)$

$$\ell_2(p,y) = (p-y^+)^2/2$$

$$\ell_1(p,y) = |p-y^+|$$

$$\ell_{\log}(p,y) = \begin{cases} -\log(p) & y=1 \\ -\log(1-p) & y=-1 \end{cases}$$

Theorem 6.3. *The losses ℓ_2 and $\ell_{\log}$ are proper. The loss ℓ_1 is not.*

Proof. Step 1. As in previous proofs, we consider (EL-P) and collect terms, breaking the sum into two parts, depending on the value of y.

$$\widehat{L}(p) = \frac{1}{m}\sum_{i=1}^m \ell(p,y_i) = \frac{1}{m}\sum_{y_i=1} \ell(p,1) + \frac{1}{m}\sum_{y_i=-1} \ell(p,-1)$$

collect terms

$$\widehat{L}(p) = q\ell(p,1) + (1-q)\ell(p,-1)$$

Step 2. For each choice of the loss, we minimize the last equation.

Verify these statements.

$$\min_p q(1-p)^2 + (1-q)(p)^2$$

gives $p=q$.

$$\min_p q\log p + (1-q)\log(1-p)$$

gives $p=q$

But

$$\min_{p\in[0,1]} q(1-p) + (1-q)p$$

has gives $p=0$ or $p=1$ as minimizer □

Exercise 6.1. *Prove that $\ell_2(p,y)$ and $\ell_{\log}$ are (i) convex losses, (ii) proper.*

Exercise 6.2. *Consider the spherical loss*

$$\ell_s(p,y) = \begin{cases} p/(p^2+(1-p)^2)^{1/2} & y=+1 \\ (1-p)/(p^2+(1-p)^2)^{1/2} & y=-1 \end{cases}$$

Determine if it is (i) convex, (ii) proper.

Exercise 6.3. *Show that the loss $\ell_1(p,y)$ is (i) convex, (ii) not proper. (Hint: find the minimizer in the definition).*

Loss for probabilities: solution summary

Inputs:

- A data set of the form (10)

Goal:

- Find $q = q(S_m)$, the fraction of $y_j = 1$ using differentiable loss minimization.

Solution:

- minimize (EL-P) using the loss ℓ_2 or the loss $\ell_{\log}$.

Warning: not all losses work

- Using ℓ_1 in the problem above will not work.

7. Model Calibration

TODO: show that, given score based model, can simply bin the model according to the score. Then do post-hoc calibration. Model in bin i corresponding to score s_i becomes

$$p_i = \sigma(s_i + w_i) \tag{12}$$

Combining ideas from (i) bin classifiers (with the 0-1 loss) to learn probability. (ii) proper losses in bins to learn probability.

Now this allows us to use the log loss in each bin to learn the fraction correct.

Model Calibration

Inputs:

- A scoring function
- Define bins by values of the score

Goal:

- Calibrate the model to (i) classify the same as before (ii) make the new scores the probability of correct classification of the bin

Solution:

- Learn the model (12) by minimizing the log loss.

REFERENCES

[DFO20] Marc Peter Deisenroth, A Aldo Faisal, and Cheng Soon Ong. *Mathematics for machine learning*. Cambridge University Press, 2020.

[GR07] Tilmann Gneiting and Adrian E Raftery. Strictly proper scoring rules, prediction, and estimation. *Journal of the American statistical Association*, 102(477):359–378, 2007.

[Mur12] Kevin P Murphy. *Machine learning: a probabilistic perspective*. MIT press, 2012.

[Mur22] Kevin P Murphy. *Machine learning: a probabilistic perspective*. MIT press, 2022.