We examine for binary response data two universes of loss functions that can serve as estimation devices for classification and for class probability estimation. The two universes differ in the scales on which the loss functions are defined: probability scales on the one hand, and arbitrary function fitting scales on the other hand. The two scales are connected by link functions. Loss functions of the first kind are defined by a consistency property known in subjective probability as being a "proper scoring rule", which is essentially the requirement of consistent class probability estimation. An example of a loss function of the second kind is exponential loss used in the boosting literature.

In the universe of loss functions, we define a one-parameter family that contains all commonly used loss functions. The ordering within this family suggests that Freund and Schapire’s exponential loss is a more drastic booster than Bernoulli loss, which in turn is more drastic than squared error loss on the probability scale, and all boost more strongly than misclassification rate.

We show how suitable choices of loss functions can adapt to classification at desired levels, such as \( P[Y = 1|x] > 0.9 \). In general, the choice of loss functions can be treated similarly to the choice of bandwidth parameters in smoothing problems, and in this sense “optimal” loss functions can be “estimated”.

We show that all loss functions permit Fisher scoring algorithms for fitting models, be they fixed linear models as in logistic regression, or nonparametric models fitted in a stagewise manner as in boosting.

**Keywords:** Proper scoring rules, link functions, binary response data, stumps, logistic regression.
1 Introduction

We consider predictor-response data with a binary response $y$ representing the observation of classes $y = 1$ and $y = 0$. Such data are thought of as realizations of a Bernoulli random variable $Y$ with $p = P[Y = 1]$ and $1 - p = P[Y = 0]$. The class 1 probability $p$ is interpreted as a function of predictors $x$: $p = p(x)$. If the predictors are realizations of a random vector $X$, then $p(x)$ is the conditional probability given $x$: $p(x) = P[Y = 1|X = x]$. Of interest are two types of problems:

- **Classification**: Estimate a region in predictor space in which class 1 is observed with the greatest possible majority. This amounts to estimating a region of the form $p(x) > c$.
- **Class probability estimation**: Approximate $p(x)$ as well as possible by fitting a model $q(x, b)$ ($b =$ parameters to be estimated).

Of the two problems, classification is prevalent in machine learning (where it is sometimes called “concept learning”, betraying its origin in AI), whereas class probability estimation is prevalent in statistics (often in the form of logistic regression).

The classification problem is peculiar in that estimation of a class 1 region requires two kinds of criteria:

- the primary criterion of interest: misclassification rate. This is an intrinsically unstable criterion for estimating models, a fact that necessitates the use of
- auxiliary criteria for estimation, such as the Bernoulli likelihood used in logistic regression, and the exponential loss used in the boosting literature. These are just estimation devices and not of primary interest.

The auxiliary criteria of classification are, however, the primary criteria of class probability estimation. This can be illustrated with the example of CART (Breiman et al. 1984, chapters xxx and xxx): CART grows classification trees with Gini as the auxiliary criterion and class probability trees with Gini as the primary criterion. However, CART prunes trees with the primary criterion, that is, classification trees with misclassification rate and class probability trees with the Gini criterion.

One goal of the present article is to describe two universes of loss functions that can be used as auxiliary criteria in classification and as primary criteria in class probability estimation:

1. One universe consists of loss functions on a probability scale. They are defined by the requirement that the minimizer is the true class 1 probability. Such loss functions have been known in subjective probability as “proper scoring rules”, but for brevity we will usually call them “$p$-losses” because their domain are probability scales. An example is the negative log-likelihood of the Bernoulli model, also called Kullback-Leibler information, log-loss, or cross-entropy (we use “log-loss” throughout).
2. The other universe consists of loss functions on implicitly defined scales. Their only requirement is that the loss-minimizing value and the probability of class 1 are in a monotone relationship. We call such loss functions “\( F \)-losses” because their domains are scales to which functions (models) are fitted. An example is the exponential loss used in boosting.

The monotone relationships between loss-minimizers and class 1 probabilities in the definition of \( F \)-losses have a simple interpretation: they are the link functions that connect fitted functions to expected values of the responses, just as in generalized linear and additive models. The link function of an \( F \)-loss lends itself to mapping the \( F \)-loss to an associated \( p \)-loss. This type of mapping has been demonstrated by Friedman, Hastie and Tibshirani (2000) in two examples: They showed that the inverse of half the logit maps the \( F \)-scale of the exponential loss to a probability scale. They did not write down the associated \( p \)-loss, however. Instead, they mapped log-loss to an \( F \)-loss using the canonical logit link.

Our initial focus is on \( p \)-losses and their structure. In particular, we construct a parametrized family of \( p \)-losses that smoothly interpolates most of the conventional loss functions, including squared error loss, log-loss, and the \( p \)-loss version of exponential loss, and that approximate misclassification loss in the limit. The construction of this family of \( p \)-losses has three purposes:

- Comparison of \( p \)-losses: The line-up of the conventional losses along the one-parameter family of \( p \)-losses allows us to conclude that squared error loss, log-loss and exponential loss are progressively further away from misclassification loss, and that they progressively leverage data with highly reliable predictions.

- Tailoring of \( p \)-losses to particular purposes: The flexibility of the parametrized family of \( p \)-losses lends itself for example to the approximation of misclassification loss with non-identical cost of misclassifying class 1 and class 0.

- Estimation of optimal \( p \)-losses for classification: Cross-validation of misclassification rates is often used for selecting models, but it can also be used for selecting auxiliary loss functions. Note that cross-validation acts on the primary loss, misclassification rate.

The first point can be used to address a puzzling question: Why was it that computer scientists in machine learning invented exponential loss, and why not statisticians? The reason is that exponential loss is not viable from a statistical perspective because of its extreme reliance on data with highly reliable predictions. This reliance is also seen as a liability of logistic regression and its use of log-loss; see Hand and Vinciotti (2003)) for a recent exposition of this argument. This reliance, however, is even more extreme for exponential loss. In so far as exponential loss underlies boosting, one could argue that the reported successes of boosting occur not because but in spite of exponential loss. The only justification for it may be an algorithmic one in that it permits an intuitive reweighting scheme. The reason why exponential loss has not shown its drawbacks is that boosting is usually performed with flexible non-parametric models (sums of stumps, for example); if
exponential loss had been used on traditional linear models, its drawbacks would have shown quickly.

The second and third point above address a peculiarity of classification: if a classification region is of the form \( q(x, b) > c \), it is irrelevant for the performance in terms of misclassification rate whether \( q(x, b) \) fits the true \( p(x) \) well, as long as \( q(x, b) > c \) and \( p(x) > c \) agree well enough. That is, the classifier does not suffer if \( q(x, b) \) is biased vis-à-vis \( p(x) \) as long as \( q(x, b) \) and \( p(x) \) are mostly on the same side of \( c \). It can therefore happen that a fairly inaccurate model yields quite accurate classification. In order to take advantage of this possibility, one should choose a loss function that is closer to misclassification rate than log-loss and exponential loss. Squared error loss is more promising in that regard, but better choices of loss functions can be found, in particular if misclassification cost is not equal for the two classes. We illustrate this situation with a recent artificial example by Hand and Vinciotti (2003); we also show that exactly this situation is found in the well-known Pima Indians Diabetes data from the UCI Machine Learning Repository (2003). The situation was addressed by Hand and Vinciotti (2003) with a proposal to modify the Iteratively Reweighted Least Squares (IRLS) algorithm by upweighting points near the classification boundary. We will show that the same effect can be achieved with suitable choices of \( p \)-losses. This is so because every \( p \)-loss can be minimized with an IRLS algorithm, and the iterative weights derived from suitable \( p \)-losses have exactly the localizing properties that Hand and Vinciotti (2003) Handcrafted into their weights. Conversely, we can show what loss functions Hand and Vinciotti’s reweighting schemes minimize.

As we mentioned earlier, \( F \)-losses can be decomposed into 1) a link function that maps the implicit \( F \)-scale to a probability scale, and 2) a \( p \)-loss on the probability scale. This decomposition provides a conceptual clarity that does not exist for raw \( F \)-losses. The clarity derives from the two very different roles that links and \( p \)-losses play:

- \( p \)-losses have to do with statistical efficiency. They determine the range of class 1 probabilities that are estimated with greatest precision.
- Link functions, together with the fitted functional form, have to do with the quality of fit. The two determine whether the systematic part of the model is correct.

The value of the decomposition can be illustrated with what is called in machine learning the problem of “label noise”. In any data set, label noise can be introduced by flipping each label with a small probability, say, 0.10. If the data originally had a true link function \( q(F) \), then the data with label noise have a true link function \( q_{\text{noise}}(F) = 0.8 \cdot q(F) + 0.1 \), with a baseline probability of 0.1 for both labels. Hence using a standard link function whose range is the full unit interval will generally amount to using an incorrect model. In statistics such situations are not only known from artificial examples but from a real situation: educational testing. Baseline probabilities arise because even random guessing gets the right answer with non-zero probability. For this problem a solution is known that lends itself readily for adaption in classification and class probability estimation: the Rasch model, which permits explicit modeling and estimation of baseline probabilities. Baseline probabilities are probably the most important case of link function violation. Be this as it may, this kind of problem and
its solution might be difficult to detect without decomposing, for example, exponential loss into link and $p$-loss.

A further benefit of the decomposition of $F$-losses into $p$-losses and link functions is of an algorithmic nature: On the probability or $p$-scale, there is a notion of expected value of the observations under the current model. This is just the ingredient that enables us to turn Newton iterations into Fisher scoring by replacing the empirical Hessian with the expected Hessian. Although both Newton’s algorithm and Fisher scoring can be expressed in the form of iteratively reweighted least squares, only the latter exhibits the intuitive form of the weights that lends itself for Hand and Vinciotti’s (2003) reweighting schemes.

Acknowledgments: We thank J.H. Friedman, D. Mease and A.J. Wyner for discussions that resulted in several clarifications. We owe a special debt to the article by Friedman, Hastie and Tibshirani (2000) which got us started. In what follows we refer to this article as FHT (2000).

2 Proper Scoring Rules or $p$-Losses

2.1 Definition of $p$-losses

Given predictor-response data $(x_n, y_n)$ with binary response $y_n \in \{0, 1\}$, we are interested in fitting a model $q(x)$ for the conditional class 1 probabilities $p(x) = P[Y = 1|x]$. (For now it is immaterial whether the model is parametric or nonparametric, which is why we write $q(x)$ without unknown parameters.) This problem would be approached by most statisticians with Maximum Likelihood based on the conditional Bernoulli model. There exist many possibilities other than Maximum Likelihood, however, and the exploration of these possibilities is the goal of this article.

We start with some intuitive reasoning: The model $q(x)$ takes on values in $[0, 1]$ which are interpreted as estimated conditional class 1 probabilities. They are to be fitted to the values 0 and 1 of the response $y$. To do so, we need a goodness-of-fit criterion that measures the closeness of the fitted probabilities $q_n = q(x_n)$ to the response values $y_n$. We assume the goodness-of-fit criterion to be an average of contributions of the individual observations:

$$R(q) = \frac{1}{N} \sum_{n=1}^{N} L(y_n|q_n).$$

We would fit the model $q(x)$ by minimizing $R$ with numerical methods. We must ask, however, whether the estimated values $q(x)$ can be interpreted as consistent estimates of class 1 probabilities. So far we have not imposed any constraints. As usual, one gains clarity by considering the limiting case for $N \to \infty$ when the observations $(x_n, y_n)$ are i.i.d. The limit will be the expectation

$$R(q) = \mathbb{E}_{X,Y} L(Y|q(X)) = \mathbb{E}_X \left( \mathbb{E}_{Y|X} L(Y|q(X)) \right).$$

We assume the model either to be parametric and to contain the truth, $p(x)$, or to be nonparametric and to be able to approximate the truth $p(x)$ arbitrarily well for $N \to \infty$. 

In either case the values \( q(x) \) will qualify as estimates of probabilities if pointwise at each \( x \) the minimizing value \( q \) is unique and equals the true \( p(x) \). The condition boils down to the requirement that for an arbitrary Bernoulli variable \( Y \) with \( P[Y = 1] = p \)

\[
\arg\min_q L(Y|q) = p .
\]  

(1)

We will want \( L() \) to satisfy this condition for all values of \( p \).

Next we re-express the condition by making use of Bernoulli-related simplifications. Because \( Y \) takes on only two values, \( L(y|q) \) consists of only two functions of \( q \): \( L(1|q) \) and \( L(0|q) \). We anticipate \( L(0|q) \) as increasing in \( q \) because values of \( q \) closer to \( y = 0 \) are a better fit, and similarly we anticipate \( L(1|q) \) as a decreasing function of \( q \). Because we prefer to express both with increasing functions only, we define

\[
L_1(1 - q) = L(1|q) , \quad L_0(q) = L(0|q) .
\]  

(2)

where both \( L_1 \) and \( L_0 \) are monotone increasing. We can now write the empirical loss for the response \( y \) and the estimate \( q \) as

\[
L(y|q) = y L_1(1 - q) + (1 - y) L_0(q) ,
\]  

(3)

and the expected loss \( E_Y L(Y|q) \) as

\[
L(p|q) = p L_1(1 - q) + (1 - p) L_0(q) .
\]  

(4)

The requirement (1) for probability estimation becomes

\[
\arg\min_q L(p|q) = p .
\]  

(5)

The condition is so crucial that we fix it in the following

**Definition:** If an expected loss \( L(p|q) \) is uniquely minimized w.r.t. \( q \) by \( q = p \ \forall p \in (0,1) \), we call the loss function a “proper scoring rule” or simply a \( p \)-loss.

The peculiar expression “proper scoring rule” stems from a large literature on subjective probability, forecasting, and meteorology. See the following references and citations therein: Shuford, Albert and Massengill (1966), Savage (1971), Schervish (1989), Murphy and Daan (1985), Winkler (1993). In subjective probability proper scoring rules are interpreted as economic incentive systems that elicit a subject’s true belief with regard to a probability \( p \).

The defining property of \( p \)-losses may or may not include the boundaries 0 and 1, depending on whether the loss is defined on the boundaries. The qualitative behavior of \( p \)-losses is illustrated in Figure 1.
Figure 1: Expected losses $L(p|q) = pL_1(1-q) + (1-p)L_0(q)$ for $p = 0, 0.2, 0.7, \text{ and } 1.0$. For $p$-losses, the weighted losses take on their minima at $q = p$. The example shown is the log-loss.

2.2 A note on $p$-losses and losses in Wald’s decision theory

We note that expected loss $L(p|q)$ is always an affine function of $p$. One is tempted to think of it as a distance between the true probability $p$ and its estimate $q$, but because $L(p|q)$ is affine in $p$, such expressions as $|q - p|$ cannot occur as expected losses. To recapitulate and amplify: We always start with observed losses that relate estimates to observations, and as such $L(y|q) = |y - q|$ will qualify. This, however, is just another way of writing $L_1(1-q) = |1-q| = 1-q$ and $L_0(q) = |0-q| = q$, because $y$ can only take on the values 0 and 1. The corresponding expected loss is $L(p|q) = p(1-q) + (1-p)q$, not $|q - p|$. (We owe this clarification to a conversation with J.H. Friedman.)

It may be helpful to clarify the relation of observed and expected losses to loss functions in the sense of Wald’s decision theory: The purpose of the latter is to relate estimates (more generally: decisions) to true parameter values. Therefore, $|q - p|$ is a valid loss function in the sense of Wald. By comparison, the purpose of observed losses is to relate estimates to observations; and expected losses are just the limiting cases of observed losses for $N \to \infty$. Expected losses can be interpreted as a special type of Wald loss function, but since they must be affine in the true parameter $p$, they form a strict subset. It would be preferable if non-overlapping terminology were used and observed losses were called “cost functions”, but the fact that for example the term “exponential loss” has become ingrained in machine learning...
makes this a futile endeavor. [We will, however, resist the use of the term “hypothesis” for classifiers, despite machine learning conventions.]

2.3 Two commonly used \( p \)-losses

In statistics, the most widely used \( p \)-loss is log-loss, the negative log-likelihood of the Bernoulli model, which is the fitting criterion of linear logistic regression. In our notation,

\[
\begin{align*}
L(y|q) &= -y \log(q) - (1-y) \log(1-q), \\
R(q()) &= \frac{1}{N} \sum_{n=1}^{N} [-y_n \log(q_n) - (1-y_n) \log(1-q_n)], \\
L(p|q) &= -p \log(q) - (1-p) \log(1-q), \\
L_1(1-q) &= -\log(q), \\
L_0(q) &= -\log(1-q),
\end{align*}
\]

where \( q_n = \psi(b^T x_n) \) and \( \psi() \) is the logistic function. Log-loss is sometimes called Kullback-Leibler loss or the cross-entropy term of the Kullback-Leibler divergence.

Another common \( p \)-loss is squared error loss:

\[
\begin{align*}
L(y|q) &= (y-q)^2 = y(1-q)^2 + (1-y)q^2, \\
R(q()) &= \frac{1}{N} \sum_{n=1}^{N} [y_n(1-q_n)^2 + (1-y_n)q_n^2], \\
L(p|q) &= p(1-q)^2 + (1-p)q^2, \\
L_1(1-q) &= (1-q)^2, \\
L_0(q) &= q^2.
\end{align*}
\]

The goodness-of-fit criterion \( R(q()) \) in this case is just the residual sum of squares with binary response values.

Both criteria are used in the construction of classification trees, squared error loss by CART (Breiman et al. 1984), and log-loss by the tree functions of the S-language (Clark and Pregibon 1992). See Section 2.11.

2.4 A new type of \( p \)-loss, derived from boosting’s exponential loss

We derive a new type of \( p \)-loss that underlies so-called “boosting”, a collection of algorithms for classification that grew out of machine learning and that has lately captured the attention of statisticians. As this is not the place to discuss the motivations and detail of boosting, we refer the reader to FHT (2000) for an introduction that is accessible to statisticians.

The one detail that we need for the present purpose is that boosting can be interpreted as minimization of a novel type of loss function, so-called “exponential loss”. Without covariates, observed exponential loss can be written as

\[
e^{-(2y-1)F} = ye^{-F} + (1-y)e^F
\]
and therefore expected exponential loss as
\[ p e^{-F} + (1 - p) e^{F}. \] (6)

The \(F\)-values do not form a probability scale, but FHT (2000) have shown that the following transformation of \(F\) provides estimates of the probability \(p\):
\[ q = \frac{1}{1 + e^{-2F}}. \]

The reason is that, for given \(p\), the minimizer of the expected exponential loss (6) is half the logit of \(p\):
\[ F_{\min}(p) = \frac{1}{2} \log \frac{p}{1-p}, \]

as an elementary calculation proves. FHT did not take the next step: mapping the exponential criterion to the probability scale, which is the natural thing to do if one thinks of \(F(q) = 1/2 \log(q/(1-q))\) as a link function. Substituting \(F(q)\) in the exponential loss gives
\[ L(p|q) = p \left( \frac{1-q}{q} \right)^{1/2} + (1 - p) \left( \frac{q}{1-q} \right)^{1/2}, \]

which by construction is a \(p\)-loss or proper scoring rule: If \(F = F_{\min}(p)\) is the minimizer of exponential loss (6), then \(q = p\) is the minimizer of \(L(p|q)\) with regard to \(q\). We will therefore refer to the \(p\)-loss defined by
\[ L_1(1 - q) = \left( \frac{1-q}{q} \right)^{1/2}, \quad L_0(q) = \left( \frac{q}{1-q} \right)^{1/2} \]
as “boosting loss”. Together with log-loss and squared error loss, it will be a standard example in what follows.

2.5 Counterexamples of \(p\)-losses

It is instructive to examine some seemingly plausible loss functions that are not \(p\)-losses:

Power losses of the form
\[ L(y|q) = |y - q|^r = y (1 - q)^r + (1 - q) q^r, \]
\[ L_1(1 - q) \sim (1 - q)^r, \quad L_0(q) \sim q^r \]
for positive exponent \(r\). The condition for a \(p\)-loss specializes to
\[ r (1 - q)^{r-2} \sim r q^{r-2}, \]
which is violated for all choices of \(r\) except \(r = 2\). That is, in the power family of losses only squared error is a \(p\)-loss.
Absolute deviation, arising from $L(y|q) = |y - q|$ (see above):

$$L_1(1 - q) = 1 - q, \quad L_0(q) = q,$$
$$L(p|q) = p(1 - q) + (1 - p)q.$$ 

Thus this loss turns out to be a special case of a power loss with $r = 1$. It is minimized by $q = 1$ for $p > 1/2$, and $q = 0$ for $p < 1/2$ (arbitrary for $p = 1/2$). It can be interpreted as a classification loss function that elicits a binary decision rather than an estimate of class 1 probabilities.

**Misclassification rate**, which considers only on which side of 1/2 the true $p$ falls:

$$L(y|q) = ,$$
$$L_1(1 - q) = 1_{[q<1/2]}, \quad L_0(q) = 1_{[q\geq1/2]},$$
$$L(p|q) = p1_{[q<1/2]} + (1 - p)1_{[q\geq1/2]}.$$ 

The minimum is taken on by any $q < 1/2$ for $p < 1/2$, and any $q \geq 1/2$ for $p \geq 1/2$. Misclassification rate is therefore distinct from the classification loss of the previous bullet. The latter is not indifferent to the actual value of $q$ on either side of 1/2.

Although misclassification rate is not a $p$-loss, it is a limiting case. See Sections 2.8 and 2.10 for a full account of the relation between misclassification and $p$-loss.

### 2.6 The structure of $p$-losses — weight functions $\omega()$

Most $p$-losses can be characterized by the following structure theorem which will be used throughout this article:

**Theorem:** Assume the losses $L_1(1 - q)$ and $L_0(q)$ are differentiable. They form a $p$-loss if and only if they satisfy

$$L'_1(1 - q) = \omega(q)(1 - q), \quad L'_0(q) = \omega(q)q,$$

for some weight function $\omega(q)$ that is strictly positive for $0 < q < 1$ and locally integrable.

Note that $\omega(q)$ does not need to be globally integrable on $(0, 1)$. We permit the loss functions $L_i$ to be unbounded near zero and one. Integrated, the two equations can be written as

$$L_1(1 - q) = \int_q^1 \omega(t)(1 - t) dt, \quad L_0(q) = \int_q^1 \omega(t)t dt.$$

Leaving the upper and lower integration boundaries indeterminate indicates the presence of arbitrary additive constants.

The theorem, in a slightly different form, goes back to Shuford, Albert and Massengill (1966); Schervish (1989) gave an extension that weakens the differentiability assumption by permitting positive measures instead of positive functions $\omega(q)$. 

9
The “if” part of the theorem is just the stationarity condition: \( \min_q L(p|q) = L(p|p) \) implies \( \frac{\partial}{\partial q}|_{q=p} L(p|q) = 0 \), that is, \(-pL'_1(1-p) + (1-p)L'_0(p) = 0\). Dividing by \( p(1-p) \) gives \( L'_1(1-p)/(1-p) = L'_0(p)/p \), which we denote by \( \omega(q) \). For a more complete proof see Appendix 1.

The theorem shows that there exists a wealth of \( p \)-losses because they are in a 1-1 relation with strictly positive weight functions \( \omega(q) \). For the above examples the weight functions are as follows:

- Boosting loss:
  \[
  \omega(q) = \frac{1}{q(1-q)^{3/2}} \quad (9)
  \]
- Log-loss:
  \[
  \omega(q) = \frac{1}{q(1-q)} \quad (10)
  \]
- Squared error loss:
  \[
  \omega(q) = 1 \quad (11)
  \]

The examples suggest that the meaning of \( \omega(q) \) as weights may go further than the theorem would let on: In the second example \( \omega(q) \) generates the weights used in the Fisher scoring algorithm for logistic regression. In the third example \( \omega(q) \) may be interpreted as generating the constant weights of ordinary LS regression. These interpretations will be born out in Section 4.

2.7 Interpretation of the weight function \( \omega() \)

In the Appendix we show that

\[
\left. \frac{\partial^2}{\partial q^2} \right|_{q=p} L(p|q) = \omega(p) .
\]

Therefore, we have to second order:

\[
L(p|q) \approx \frac{1}{2} \omega(p)(p - q)^2 .
\]

Hence \( \omega(p) \) determines how quickly the loss increases as the estimate \( q \) deviates from the true probability \( p \).

In addition, we see that specifying the second derivatives \( \frac{\partial^2}{\partial q^2}|_{q=p} L(p|q) \) for all values \( p \) completely determines a \( p \)-loss. In geometric terms, \( L(p|q) \) is completely determined by the second derivatives w.r.t. \( q \) along the diagonal \( q = p \).

This interpretation of \( \omega() \) can be used to guide the design of \( p \)-losses. For example, if one is primarily interested in estimating \( p \) in the neighborhood of a value \( p_0 \), one can choose a weight function \( \omega() \) with large values around \( p_0 \) and small values otherwise. This idea is illustrated in Section 2.10.


2.8 $p$-losses are mixtures of misclassification losses

In generalization of misclassification rate, we consider cost-weighted misclassification losses and relate them to $p$-losses. We will show that any $p$-loss is an average or mixture of cost-weighted misclassification losses.

Consider situations in which the two types of misclassification entail differing costs. W.l.o.g. we assume the sum of costs to add to 1:

- $c$ is the observed cost of misclassifying $y = 0$ as class 1; the expected cost of classifying as class 1 is therefore $c \cdot (1 - p)$.
- $1 - c$ is the observed cost of misclassifying $y = 1$ as class 0; the expected cost of classifying as class 0 is therefore $(1 - c) \cdot p$.

The optimal classification is therefore class 1 iff $p \cdot (1 - c) \geq (1 - p) \cdot c$, that is, $p \geq c$. In the absence of knowledge of $p$ but availability of an estimate $q$, we classify as class 1 when $q \geq c$. The observed cost-weighted misclassification loss can be written as

$L_{1,c}(1 - q) = (1 - c) \cdot 1_{[q<c]} , \quad L_{0,c}(q) = c \cdot 1_{[q\geq c]} .

There is some arbitrariness at $q = c$. Any of the following losses can be used also:

$L_{1,c}(1 - q) = (1 - c) \cdot (1_{[q>c]} + (1 - f) \cdot 1_{[q=c]}) , \quad L_{0,c}(q) = c \cdot (1_{[q<c]} + f \cdot 1_{[q=c]}) ,

where the arbitrary value $f \in [0, 1]$ may mean a randomized classification: when $q = c$, pick class 1 with probability $f$. The classification for $q = c$ is immaterial in what follows because in the result below this case has measure zero.

The link between cost-weighted misclassification losses and $p$-losses is established in the following

**Proposition:** Any $p$-loss is a mixture of cost-weighted misclassification losses with $\omega(c) dc$ as the mixing measure:

$L_1(1 - q) = \int_0^1 L_{1,c}(1 - q) \omega(c) \, dc , \quad L_0(q) = \int_0^1 L_{0,c}(q) \omega(c) \, dc . \quad (12)$

The proof is immediate: the right hand sides of Equations (12) are identical to the right hand sides of Equations (8).

The proposition has practical as well as mathematical meaning:

- **Interpretation of $p$-losses:** Looking at the weight function (9) of boosting loss and (10) of log-loss, one may wonder about the infinite mass they place near zero and one, indicating an emphasis on getting classification right for extreme misclassification costs of both classes. We will see below that this eagerness to perform well on both ends of the cost axis can lead to situations in which classification performance is low on both ends. The artificial data example of Hand and Vinciotti (2003) makes just this point. Our analysis of the Pima Indians diabetes data in Section 7 illustrates the effect in a real data example.
• Robustness to misspecified costs: In practice one can often argue that misclassification costs are not uniquely determined. The relative indeterminacy of costs suggests that classification results should be insensitive to small changes in \( c \), or, more precisely, classification should be accurate across a range of indeterminacy of \( c \). It would then seem natural not to seek optimality for a single misclassification loss but for an average of misclassification losses, that is, for a \( p \)-loss. The \( p \)-loss should localize by peaking its weight function \( \omega(q) \) at costs in the range of interest. The hope is that optimizing a \( p \)-loss of this nature would lend stability and near-optimality for the set of misclassification costs of interest.

[The authors have been well aware of this idea for a long time but the first author heard it independently from David Hand in private communication in 2000.]

The argument given here is of a quasi-economic nature, but there exists a statistical argument as well when fitting biased models; see Section 2.10 and Section 7.

• Convexity and closure properties: Mathematically, the proposition shows that the notion of a \( p \)-losses could be expanded to the closed convex cone generated by cost-weighted misclassification losses. The above proposition would then replace the weight function \( \omega(\cdot) \) with arbitrary non-negative measures, as in Schervish (1989). The definition of proper scoring rules or \( p \)-losses should drop the uniqueness requirement for the minimizer of the loss: \( q = p \) will always be required to be a minimizer, but not necessarily a unique one. The resulting notion of generalized \( p \)-losses would comprise our \( p \)-losses as part of the interior of the cone, and cost-weighted misclassification losses as its extremal elements. A Choquet-type theorem would extend the above proposition and state that all generalized \( p \)-losses can be represented by a non-negative measure with support on the extremal points.

2.9 The Beta family of \( p \)-losses

Among \( p \)-losses there exists a 2-parameter family that is sufficiently rich to encompass most commonly used losses, among them boosting loss, log-loss, squared error loss, and misclassification loss in the limit. This family is modeled after the Beta densities:

\[
\omega(q) = q^{\alpha-1} (1 - q)^{\beta-1}.
\]

The losses are hence defined by

\[
L'_1(1 - q) = q^{\alpha-1}(1 - q)^{\beta}, \quad L'_0(q) = q^\alpha(1 - q)^{\beta-1},
\]

up to an irrelevant multiplicative constant. Here is a list of special cases:

• \( \alpha = \beta = -1/2 \): The boosting loss introduced in Section 2.4,

\[
L_1(1 - q) = \left(\frac{1 - q}{q}\right)^{1/2}, \quad L_0(q) = \left(\frac{q}{1 - q}\right)^{1/2}.
\]
\begin{itemize}
  \item \(\alpha = \beta = 0\): Log-loss or negative log-likelihood of the Bernoulli model,
    \[ L_1(1-q) = -\log(q) , \quad L_0(q) = -\log(1-q) . \]
  \item \(\alpha = \beta = 1/2\): A new type of loss, intermediate between log-loss and squared error loss,
    \[ L_1(1-q) = \arcsin((1-q)^{1/2}) - (q(1-q))^{1/2} , \quad L_0(q) = \arcsin(q^{1/2}) - (q(1-q))^{1/2} . \]
  \item \(\alpha = \beta = 1\): Squared error loss,
    \[ L_1(1-q) = (1-q)^2 , \quad L_0(q) = q^2 . \]
  \item \(\alpha = \beta = 2\): A new loss closer to misclassification than squared error loss,
    \[ L_1(1-q) = \frac{1}{3}(1-q)^3 - \frac{1}{4}(1-q)^4 , \quad L_0(q) = \frac{1}{3}q^3 - \frac{1}{4}q^4 . \]
  \item \(\alpha = \beta \to \infty\): The indicator functions that describe misclassification loss,
    \[ L_1(1-q) = 1_{[1-q>1/2]} , \quad L_0(q) = 1_{[q\geq1/2]} . \]
\end{itemize}

Values of \(\alpha\) and \(\beta\) that are integer multiples of 1/2 permit closed formulas for \(L_1\) and \(L_0\). For other values one needs a numeric implementation of the incomplete Beta function.

### 2.10 Approximating cost-weighted misclassification loss for estimation

The purpose of this section is to tailor \(p\)-losses to classification with unequal cost of misclassification or, equivalently, to classification at thresholds other than 1/2.

Consider the limiting case when the weight function \(\omega(q)\) concentrates all its mass around \(q = c\), that is, when \(\omega(q)\) becomes a Dirac measure:
\[ \omega(q) = \delta_{q=c} . \]

We can ask what loss is obtained for such degenerate weight functions. The equations
\[ L'_1(1-q) = \delta_{q=c} (1-q) , \quad L'_0(q) = \delta_{q=c} q , \]
can be formally solved by
\[ L_{1,c}(1-q) = (1-c) \cdot 1_{[q<c]} , \quad L_{0,c}(q) = c \cdot 1_{[q\geq c]} , \]
which is the cost-weighted misclassification loss with cost parameter \(c\). (Again, the definition at \(q = c\) is immaterial.)
Figure 2: Loss functions $L_0(q)$ and weight functions $\omega(q)$ for various values of $\alpha = \beta$: exponential loss ($\alpha = -1/2$), log-loss ($\alpha = 0$), squared error loss ($\alpha = 1$), misclassification error ($\alpha = \infty$). These are scaled to pass through 1 at $q = 0.5$. Also shown are $\alpha = 2$, $20$ and $100$ scaled to show convergence to the step function.

These facts suggest that in order to perform well in classification with general misclassification costs, one could approximate the degenerate weight function with non-degenerate weight functions that spread out the mass but concentrate it around the location of the degenerate weight function. The Beta family of $p$-losses is indeed flexible enough for this purpose: the densities

$$\omega_{\alpha,\beta}(q) = \frac{1}{B(\alpha,\beta)} q^{\alpha - 1} (1 - q)^{\beta - 1}$$

converge to $\omega(q) = \delta_c(q)$, for example, when

$$\alpha, \beta \to \infty, \quad \text{subject to } \frac{\alpha}{\beta} = \frac{c}{1 - c}.$$  

For a proof note the following:

- The expected value of $q$ under a Beta distribution with parameters $\alpha$ and $\beta$ is

$$\mu = \frac{\alpha}{\alpha + \beta} = c,$$

were equality with $c$ follows from the constraint $\alpha/\beta = c/(1 - c)$.

- The peakedness of the Beta distribution can be measured by its variance,

$$\sigma^2 = \frac{\alpha \beta}{(\alpha + \beta)^2(\alpha + \beta + 1)} = \frac{c(1 - c)}{\alpha + \beta + 1},$$
which tends to zero as $\alpha$ and $\beta$ tend to infinity.

In a limiting sense, the ratio of the exponents, $\alpha/\beta$, acts as a cost ratio for the classes.

The above construction has obvious methodological implications for logistic regression and also, as will be shown later, for boosting: log-loss and exponential loss, which they use, respectively, can be replaced with the $p$-losses generated by the above weight functions $\omega_{\alpha,\beta}(q)$. In doing so we can achieve improved classification for particular cost ratios or class-probability thresholds when the fitted model is biased but adequate for describing classification boundaries individually. The appropriate degree of peakedness of the weight function can be estimated from the data. See Section 7 for examples.

This construction enables us to derive in a principled manner a reweighting technique proposed for just this purpose by Hand and Vinciotti (2003). They devised their scheme in an algorithmic way by modifying the weights of the IRLS (Iteratively Reweighted Least Squares) algorithm for Fisher scoring in logistic regression. We will see in Section 4 that their weights amount to a particular choice of weight function $\omega(q)$. Because $\omega$’s are in a 1-1 relation with $p$-losses, we can therefore construct $p$-losses that Hand and Vinciotti implicitly minimize. Their weights are based on Gaussian kernels, that is,

$$\omega(q) = \phi \left( \frac{q - c}{\sigma} \right),$$

where $\phi(t)$ is the standard Gaussian density. The associated $p$-loss is therefore

$$L_1(1-q) = \int_q^1 (1-t) \cdot \phi \left( \frac{t - c}{\sigma} \right) \, dt, \quad L_2(q) = \int_0^q t \cdot \phi \left( \frac{t - c}{\sigma} \right) \, dt.$$

The Gaussian standard deviation $\sigma$ can be used here as a measure of peakedness, but note that it is not the standard deviation of the weight function because of the restriction of the Gaussian distribution to the unit interval $[0, 1]$.

The use of $p$-losses tailored to particular misclassification costs in logistic regression will be demonstrated in Section 7.

### 2.11 Envelopes of $p$-losses as information criteria for tree estimation

This section is about a one-to-one relation between $p$-losses on the one hand and information criteria such as the Gini index and entropy on the other hand. Such information criteria are commonly used in algorithms for classification trees such as: CART (Breiman et al. 1984), which uses the Gini index, and the tree functions in the S language (Clark and Pregibon 1992), which use entropy. These algorithms agree with each other in that they all estimate local conditional class probabilities with simple proportions, but they differ in how they judge the fit of these proportions with information criteria. The reason for discussing trees (and any method that fits local proportions for that matter) is that they are often used...
in boosting algorithms as “base-learners” or “weak learners”. In Section 3 will see that in a broad sense boosting algorithms can be interpreted as minimizers of $p$-losses combined with link functions. We argue that if a boosting algorithm implicitly uses a particular $p$-loss, then its base-learner should use the associated information criterion. The argument is really a request for consistency at all levels so boosting can more properly be interpreted as minimization of a particular $p$-loss.

The simple connection between $p$-losses and information criteria is as follows: Given an expected $p$-loss $L(p|q) = pL_1(1 - q) + (1 - p)L_0(q)$, the associated information criterion is the loss for the proportion $q = p$, that is, $L(p|p) = pL_1(1 - p) + (1 - p)L_0(p)$. We write this information criterion as

$$K(p) = L(p|p).$$

**Proposition:** The information criterion $K(q)$ is the concave lower envelope of its $p$-loss $L(p|q)$. If $K(q)$ is an arbitrary smooth and strictly concave function, it determines a unique $p$-loss $L(p|q)$. The losses $L_i$ and the associated weight function $\omega(q)$ are:

$$L_1(1 - q) = K'(q)(1 - q) + K(q), \quad L_0(q) = K'(q)q + K(q),$$

$$\omega(q) = -K''(q).$$

For a proof, note that $L(p|q)$ is affine in $p$, which makes $K(q)$ its concave lower envelope. The affine function $p \mapsto L(p|q)$ is the tangent of $K$ at $q$, hence $L(p|q) = K'(q)(p - q) + K(q)$, which defines $L_1(1 - q)$ and $L_0(q)$. A simple calculation verifies the relation between $K(q)$ and $\omega(q)$, which links strict concavity of $K$ to positivity of $\omega$.

Examples from the Beta family of $p$-losses:

- $\alpha = \beta = -1/2$: The boosting loss leads to a semi-circle,

$$K(p) = 2 \cdot [p(1 - p)]^{1/2}.$$

- $\alpha = \beta = 0$: Log-loss leads to entropy,

$$K(p) = -p \log(p) - (1 - p) \log(1 - p).$$

- $\alpha = \beta = 1$: Squared error loss leads to the Gini index,

$$K(p) = p(1 - p).$$

- $\alpha = \beta \to \infty$: Misclassification loss with cost $c$ for misclassifying $y = 0$ leads to

$$K(p) = \min((1 - c)p, c(1 - p)).$$

It would be consistent if trees or stumps used by AdaBoost (Freund and Schapire 1996, Schapire and Singer 1998) were built with the first information criterion, and trees or stumps used by LogitBoost (FHT 2000) with the entropy criterion.
2.12 $p$-losses, information criteria, and Bregman distances

Minimizing a $p$-loss $L(p|q)$ can be interpreted as minimizing a so-called “Bregman distance” (Bregman 1967), which we can define in our context as

$$B(p|q) = L(p|q) - K(p).$$

A Bregman distance is therefore just a $p$-loss normalized such that a value zero obtains for $q = p$: $B(p|p) = 0$. Bregman distances are non-negative but not generally symmetric in their arguments. They are therefore not metrics in the usual sense, yet by convention the term “distance” is used.

The usual examples of $p$-losses yield the following Bregman distances:

- **$\alpha = \beta = -1/2$**: The boosting loss leads to

  $$B(p|q) = p \left( \frac{1 - q}{q} \right)^{1/2} + (1 - p) \left( \frac{q}{1 - q} \right)^{1/2} - 2 \cdot [p(1 - p)]^{1/2}.$$ 

- **$\alpha = \beta = 0$**: Log-loss leads to the Kullback-Leibler divergence,

  $$B(p|q) = -p \log \frac{q}{p} - (1 - p) \log \frac{1 - q}{1 - p}.$$
\* \( \alpha = \beta = 1 \): Squared error leads to squared deviation,

\[ B(p|q) = (p - q)^2. \]

In the literature, a general Bregman distance is based on a convex function defined on a convex set in an arbitrary linear space. With information criteria in mind, we prefer concave functions \( K(x) \):

\[ B(y|x) = K(x) + \partial K(x) \cdot (y - x) - K(y). \]

The first two terms form the tangent of \( K \) at \( x \) as a function of \( y \). The three terms together are the tangent minus the function evaluated at \( y \). This is depicted in Figure 3 for probability scales \( p \) and \( q \). (For a convex function one would use function minus tangent instead.)

We discussed the connection between Bregman distances and \( p \)-loss because there has been recent use of Bregman distances in machine learning for proving convergence of boosting algorithms (Lafferty et al. 1997, Collins et al. 2000).

### 3 Boosting and F-Losses

The purpose of this section is to link boosting to \( p \)-losses and develop some structure for loss functions as they are used in the boosting framework. The idea derives from a calculation for a particular case in FHT (2000).

#### 3.1 Boosting and exponential loss

Boosting is a scheme for doing roughly what the name suggests: boost a weak method by repeatedly applying it such that a strong method results. See Freund and Schapire (1996), Schapire and Singer (1998), and Schapire (2002) for a very readable introduction. According to FHT boosting can be understood as “stagewise additive fitting” using so-called “exponential loss”. A barebones formulation of boosting is as follows. Given data \((x_n, y_n^*)\) where \( x_n \in \mathbb{R}^p \) are predictor vectors, \( y_n^* \in \{-1, +1\} \) are recoded class labels \((= 2y_n - 1, \) following the machine learning convention), and given a class of functions \( F \) (called “weak learners” in machine learning), perform the following algorithm:

1. Initialize \( F(x) = 0 \).
2. Repeat for \( t = 1 \) to \( T \):
   
   \[ f_t \leftarrow \text{argmin}_{f \in F} \sum_n e^{-y_n^*(F(x_n) + f(x_n))}, \quad F(x) \leftarrow F(x) + f_t(x) \]

3. Return the classifier: \( c(x) = 1_{[F(x) > 0]} \)

The minimization in each iteration may be heuristic, as is the case when \( F \) is a class of trees of depth greater than one. Although for some authors reweighting is an essential ingredient
of boosting, we purposely wrote the algorithm without reference to weights, because our emphasis is on loss functions. Weights come into play when one introduces current weights \( w_n \sim e^{-y_n^2 p(x_n)} \) and interprets the criterion as a weighted sum \( \sum_n w_n e^{-y_n^2 p(x_n)} \).

In this section we are concerned only with one aspect of boosting, namely, the structure of potential alternatives to exponential loss, and hence alternative reweighting schemes. Boosting loss has the form

\[
L_{\text{exp}}(p|F) = E_{Y^*} e^{-Y^* F} = pe^{-F} + (1-p)e^F ,
\]

where as before the response is recoded as

\[
Y^* = 2Y - 1 \in \{+1, -1\} ,
\]

and hence \( P[Y^* = 1] = P[Y = 1] = p. \) [FHT use just the opposite convention for \( Y \) and \( Y^* \).] Also, we now consider loss functions on scales other than probabilities. The meaning of the scale of \( F \)-values is implicitly defined by the exponential loss. The \( F \)-scale has a symmetry about zero, which follows from the signs inside the losses: \( L_1 = e^{-F} \) and \( L_0 = e^F \). The product \( Y^* F \) estimates the degree to which the fit \( F \) produces correct classification: \( Y^* F > 0 \) indicates a belief in correct classification, but \( Y^* F >> 0 \) more strongly so. The term “boosting the margin” coined by Freund and Schapire (1996) describes the push for large values of \( Y^* F \) when minimizing exponential loss. It is now natural to perform classification by thresholding \( F \) at zero: \( F \geq 0 \leftrightarrow \text{class 1} \).

A goal of this section is to show that FHT’s interpretation of exponential loss (almost) amounts to saying that this loss is equivalent to a \( p \)-loss combined with a suitable link function that links the \( F \)-scale to a probability scale.

### 3.2 Generalizing exponential loss to \( F \)-losses

Our extension of the boosting framework starts with the following thought: If minimizing the exponential loss is a form of “boosting the margin”, so is minimizing any loss of the form

\[
L(p|F) = E L(-Y^* F) = pL(-F) + (1-p)L(F) \]

as long as the loss \( L \) satisfies some minimal conditions. We will shortly derive such conditions, but before doing so we take one more step towards greater generality by removing the restriction of symmetry imposed by the notion of “margin”. We see this notion as a conceptual and practical hindrance because of the symmetry between classes it imposes. Taking a cue from \( p \)-losses, we will allow asymmetric losses as follows:

\[
L(p|F) = E [y L_1(-F) + (1-y) L_0(F)] = p L_1(-F) + (1-p) L_0(F) \quad (13)
\]

Note that we write these loss functions with roman “\( L \)”, whereas we write the loss functions that form \( p \)-losses with italic “\( L \)”. By abuse of notation we use bold \( L \) for combined loss on both \( p \)- and \( F \)-scales.

The conditions on a pair of losses \( L_1 \) and \( L_0 \) that make them meaningful boosters are essentially requirements to link the \( F \)-scale to the \( p \)-scale in a monotone fashion. Informally, if \( p \) increases, \( F \) increases. Technically, the conditions are as follows:
1. The loss \( pL_1(-F) + (1-p)L_0(F) \) has a unique minimum \( F = F(p) \) for every \( p \).

2. The minimizer \( F(p) \) is a strictly increasing function of \( p \).

Together these conditions allow the interpretation of \( F \) as a measure of certainty of classification: the larger \( p \) and hence the more certain class 1 is, the larger the loss-minimizing value \( F = F(p) \) will be. [We can allow \( +\infty \) and \(-\infty \) as possible values for \( F(p) \), which often helps in accommodating the extremes \( p = 1 \) and \( p = 0 \).] We use the following terminology:

**Definition:** A loss function of the form (13) that satisfies conditions 1. and 2. is called an “\( F \)-loss”.

This terminology is (unimaginatively) derived from the fact that the losses live on a scale that is commonly denoted by \( F \). The letter \( F \) is suggestive, however, because it is on this scale that functions (models) are being fitted. In boosting, these functions are sums of “weak learners”, that is, nonlinear model terms such as stumps or shallow trees. [Those readers who enjoy contractions may pronounce “\( F \)-loss” more imaginatively as “\( f \)-loss”.

### 3.3 Decomposition of \( F \)-losses into \( p \)-losses and link functions

In view of the defining conditions of a \( F \)-loss, it is a small step to the idea of a link function: The minimizing \( F \) as a function of \( p \) can be interpreted as a link function that establishes a correspondence between the scale of \( F \)-values and the probability scale of \( p \)-values. We use the notation \( F = F(q) \) for this link function, and \( q = q(F) \) for its inverse. We use for either the term “link”.

**Proposition:** A \( F \)-loss with minimizer \( F(p) \) can be mapped to a \( p \)-loss as follows:

\[
L_1(1-q) = L_1(-F(q)), \quad L_0(q) = L_0(F(q)).
\]

By construction, the loss on the probability scale, \( pL_1(1-q) + (1-p)L_0(q) = pL_1(-F(q)) + (1-p)L_0(F(q)) \), has the unique minimizer \( q = p \), which makes it a \( p \)-loss.

Following FHT (2000), the exponential loss \( pe^{-F} + (1-p)e^F \) with \( L_1(F) = L_0(F) = e^F \) has the unique minimizer

\[
F(p) = \frac{1}{2} \log \left( \frac{p}{1-p} \right),
\]

which is the logit up to the (relevant) factor 1/2. The corresponding \( p \)-loss is therefore:

\[
L_1(1-q) = \left( \frac{1-q}{q} \right)^{1/2}, \quad L_0(q) = \left( \frac{q}{1-q} \right)^{1/2},
\]

which is one of the examples listed in Section 2.9.

In the following proposition we characterize \( F \)-losses under differentiability assumptions:
**Proposition:** Assume that \( L_1(-F) \) and \( L_0(F) \) are defined and continuously differentiable on an open interval of the real line, that \( L_0'(F)/L_1'(-F) \) is strictly increasing, and that

\[
\inf_F \frac{L_0(F)}{L_1'(-F)} = 0, \quad \sup_F \frac{L_0'(F)}{L_1'(-F)} = \infty.
\]

Then \( L_1 \) and \( L_0 \) define a \( F \)-loss whose minimizer \( F = F(q) \) is the unique solution of

\[
\frac{L_0'(F)}{L_1'(-F)} = \frac{q}{1-q}.
\]

Proof: The stationarity condition \( \frac{d}{dF}(pL_1(-F) + (1-p)L_0(F)) = 0 \) produces the above equation as a necessary condition for the minimizer. The assumptions grant the existence of an inverse of the map \( F \to L_0'(F)/L_1'(-F) \). This inverse is defined on the open interval \((0, \infty)\). In addition, this inverse is necessarily strictly monotone, which combines with the strict monotonicity of the odds ratio \( q/(1-q) \) to the strict monotonicity of the minimizer \( F(q) \).

Finally, we note that symmetry of the \( F \)-loss, \( L_1(F) = L_0(F) \), implies symmetry of the \( p \)-loss, \( L_1(q) = L_0(q) \), and symmetry of the natural link about \((0, 1/2): q(F) + q(-F) = 1\). This follows from

\[
F(q) = \arg\min_F \left[ q L_1(-F) + (1-q) L_0(F) \right], \\
F(1-q) = \arg\min_F \left[ (1-q) L_1(-F) + q L_0(F) \right],
\]

and the fact that \( q(F) \) is symmetric about \((0, 1/2) \) iff \( F(q) \) is symmetric about \((1/2, 0): F(q) = -F(1-q) \).

### 3.4 Mapping \( p \)-losses to \( F \)-losses with link functions

So far we have considered mapping \( F \)-losses to \( p \)-losses. FHT (2000) perform the reverse exercise for one particular \( p \)-loss, log-loss. They map it to the logit scale \( F = \log(q/(1-q)) \) with the natural logistic link \( q(F) = 1/(1 + \exp(-F)) \):

\[
L(p|F) = -p \log(q(F)) - (1-p) \log(1-q(F)) = p \log(1+e^{-F}) + (1-p) \log(1+e^F) = \mathbb{E} \log(1+e^{-Y^*F})
\]

The \( F \)-loss is given by \( L_1(F) = L_0(F) = \log(1+e^F) \). FHT (2000) proposed this as a competitor of exponential loss. It also boosts the margin, although less strongly so: for very negative values, \( Y^*F << 0 \), one has \( \log(1+e^{-Y^*F}) \approx -Y^*F \), as opposed to \( e^{-Y^*F} \) for exponential loss.

In general, the reverse transport of a \( p \)-loss to a \( F \)-loss requires only minimal conditions on the link:
Proposition: Assume that $L_0(q)$ and $L_1(1-q)$ form a $p$-loss. Let $q(F)$ be a strictly increasing function on an open interval of $\mathbb{R}$ satisfying

$$\inf_F q(F) = 0, \quad \sup_F q(F) = 1.$$ 

Then $L_1(-F) = L_1(1-q(F))$ and $L_0(F) = L_0(q(F))$ form a $F$-loss.

Finally, we note that symmetry of the $p$-loss, $L_1(q) = L_0(q)$, and symmetry of the link $q(F)$, $q(-F) + q(F) = 1$, imply symmetry of the $F$-loss: $L_1(F) = L_0(F)$.

3.5 A parametrized family of $F$-losses **

We limit ourselves to the symmetric case, $L_1(F) = L_0(F)$ and write $L$ for both. The simplest family of symmetric $F$-losses that may come to mind are exponential criteria:

$$L(F) = e^{F/\gamma}, \quad \gamma > 0.$$ 

These, however, have all a common $p$-loss: $L_i(t) = (t/(1-t))^{1/2}$ ($i = 0, 1$), and the only difference is in the scale of the link: $q_i(F) = 1/(1 + \exp(-F/\gamma))$.

A more successful attempt at a one-parameter family of $F$-losses that interpolate exponential loss and log-loss, is as follows.

$$L(F) = \begin{cases} 
(1 + e^F)^\gamma - 1 \big/ \gamma & (\gamma \neq 0) \\
\log(1 + e^F) & (\gamma = 0)
\end{cases}$$

This family is built after the Box-Cox family of response transformations in regression. The dependence on $\gamma$ is analytic, including at $\gamma = 0$. We will show that $L$ results in a $F$-loss for $\gamma > -1$. For $\gamma = 1$ we obtain the exponential loss and for $\gamma = 0$ log-loss. None of its members maps to squared error as the associated $p$-loss, but the limiting case of misclassification error is approached for $\gamma \downarrow -1$ as we will show below.

We first verify that $L(F)$ results in a $F$-loss for $\gamma > -1$:

$$\frac{L'(F)}{L'(-F)} = \frac{(1 + e^F)^{\gamma-1} e^{2F}}{(1 + e^{-F})} = \frac{(1 + e^F)^{\gamma-1} e^{2F}}{(1 + e^{-F})e^F} = \frac{(1 + e^F)^{\gamma-1} e^{2F}}{e^{F} + 1} = e^{(\gamma+1)F}.$$ 

Therefore $L'(F)/L'(-F)$ is strictly increasing exactly for $\gamma > -1$. Finally note that the infimum and supremum are 0 and $\infty$, respectively, which completes the proof.

The link $q(F)$ for the symmetric $F$-loss $L(F)$ is obtained by equating $q/(1-q)$ with $L'(F)/L'(-F)$, yielding

$$F(q) = \frac{1}{\gamma + 1} \log \frac{q}{1-q}.$$ 

22
It is a peculiarity of this family that the link function is, up to a factor, the same for all members. The factor turns out to be critical, though, as we will now see.

We examine the limiting behavior of the $F$-loss for $\gamma \downarrow -1$. We note that it would be a mistake to look at the limiting behavior of $L(F)$ as a function of $\gamma$ because the $F$-scales differ in meaning for differing values of $\gamma$. Instead we consider the link as it establishes a correspondence between the $F$-scales and the probability scale. From the formula for $F(q)$ in the previous paragraph we infer

$$\lim_{\gamma \downarrow -1} F(q) = \begin{cases} +\infty & q > 1/2 \\ 0 & q = 1/2 \\ -\infty & q < 1/2 \end{cases}$$

In other words, as $\gamma$ descends to $-1$, the $F$-loss minimizer $F(q)$ wanders off to $\pm \infty$, except for equal odds. That is, in the limit we obtain a classifier, as opposed to a probability estimator. If one maps these $F$-losses to the probability scale, the resulting $p$-loss will show the same limiting behavior as the Beta family of Section 2.9.

This limiting behavior illustrates drastically that not all $F$-scales are equal. It reinforces the idea that the probability scale of $p$-losses is where comparisons should be made.

4 Iteratively Reweighted Least Squares Algorithms for Linear Models

We turn to fitting linear models with Newton steps and Fisher scoring, and reformulations of both in terms of Iteratively Reweighted Least Squares or IRLS algorithms. Fisher scoring is possible only when $F$-losses are decomposed into $p$-losses and link functions, because only on the probability scale can one replace empirical with expected Hessians.

Linear models are not the material of the boosting literature; following FHT boosting can be seen as fitting additive models in a stagewise fashion. Just the same, it is useful to apply the loss function methodology accumulated so far to the most conventional of all statistical tools, linear models. In machine learning, iterative algorithms such as Newton steps are called “parallel updating” (Collins et al. 2000), which is what we are doing in this section. Stagewise fitting is the topic of Section 5.

Let $F = \mathbf{x}^T\mathbf{b}$ be a linear model. In boosting the training or estimating criterion is empirical exponential loss, shown in Section 3.1. We render the criterion for symmetric $F$-losses as follows:

$$R(\mathbf{b}) = \frac{1}{N} \sum_{n=1..N} L(-y_n^* \mathbf{x}_n^T\mathbf{b}) .$$

For general, possibly asymmetric $F$-losses the same criterion is

$$R(\mathbf{b}) = \frac{1}{N} \sum_{n=1..N} \left[ y_n L_1(-\mathbf{x}_n^T\mathbf{b}) + (1 - y_n) L_0(\mathbf{x}_n^T\mathbf{b}) \right] .$$
Equivalently, for \( p \)-losses with link function \( q(F) \) the criterion looks as follows:

\[
R(b) = \frac{1}{N} \sum_{n=1}^{N} \left[ y_n L_1(1 - q(x_n^T b)) + (1 - y_n) L_0(q(x_n^T b)) \right].
\]

In all cases Newton updates have the form

\[
b_{new} = b_{old} - \left( \partial^2 R(b) \right)^{-1} \left( \partial_b R(b) \right).
\]

It is well-known that, for criteria such as those above, Newton updates can be expressed as solutions to a weighted LS problem, with iteration-dependent weights and response. That is, the updates an be expressed as solutions to normal equations with a suitable design matrix \( X \), a diagonal weight matrix \( W \), and a response vector \( z \), the latter two changing from update to update:

\[
b_{new} = b_{old} + \left( X^T W X \right)^{-1} \left( X^T W z \right),
\]

Of the three ingredients, only the fixed design matrix can be written down at this point: \( X = (x_1, ..., x_N)^T \). The current weight matrix \( W = \text{diag}(w_1, ..., w_N) \) \((N \times N)\) and the current response \( z = (z_1, ..., z_N)^T \) \((N \times 1)\) are to be determined below. We wrote the IRLS update in an incremental form whereas in the form usually shown in the literature one artifically absorbs \( b_{old} \) in the current response \( z \). We prefer the incremental form because of our interest in stagewise fitting, which is an incremental procedure (Section 5).

The usefulness of the reformulation as IRLS algorithms is two-fold: practical and conceptual. Reweighted LS is the closest analog in statistics to the algorithmic reweighting approach of the boosting literature. The derivation of reweighting schemes from loss functions is a means for interpreting boosting algorithms and for devising new reweighting schemes as well.

The derivation of IRLS algorithms for arbitrary \( p \)-losses amounts to an extension of FHT’s IRLS-based boosting proposal from logistic regression to \( p \)-losses in general.

### 4.1 Newton updates and IRLS for \( F \)-losses

For Newton updates based on symmetric \( F \)-losses with \( L = L_1 = L_0 \), the ingredients are

\[
R(b) = \frac{1}{N} \sum_{n=1}^{N} L(-y_n^* x_n^T b),
\]

\[
\partial_b R(b) = \frac{1}{N} \sum_{n=1}^{N} L'(-y_n^* x_n^T b) (-y_n^*) x_n,
\]

\[
\partial^2_b R(b) = \frac{1}{N} \sum_{n=1}^{N} L''(-y_n^* x_n^T b) x_n x_n^T,
\]

where we made use of \((y_n^*)^2 = 1\). A Newton increment \(- \left( \partial^2_b R(b) \right)^{-1} \left( \partial_b R(b) \right)\) can be written as a weighted LS solution with the following current weights and responses:

\[
w_n = L''(-y_n^* x_n^T b),
\]

\[
z_n = \frac{L'(-y_n^* x_n^T b)}{L''(-y_n^* x_n^T b)} y_n^*.
\]
For exponential loss $L(F) = \exp(F)$, some simplifications occur:

$$w_n = \exp(-y_n^* x_n^T b), \quad z_n = y_n^*.$$  

The weights are exactly those of the boosting algorithm with $F_n = x_n^T b$.

For Newton updates based on general $F$-losses, the ingredients are

$$R(b) = \frac{1}{N} \sum_{n=1..N} \left[ y_n L_1(-x_n^T b) + (1 - y_n) L_0(y_n x_n^T b) \right],$$

$$\partial_b R(b) = \frac{1}{N} \sum_{n=1..N} \left[ -y_n L_1'(-x_n^T b) + (1 - y_n) L_0'(x_n^T b) \right] x_n,$$

$$\partial^2_b R(b) = \frac{1}{N} \sum_{n=1..N} \left[ y_n L_1''(-x_n^T b) + (1 - y_n) L_0''(x_n^T b) \right] x_n x_n^T.$$

The current IRLS weights and responses are therefore

$$w_n = y_n L_0'(x_n^T b), \quad z_n = \frac{L_1'(x_n^T b)}{L_0'(x_n^T b)} - (1 - y_n) \frac{L_0'(x_n^T b)}{L_0''(x_n^T b)}.$$

These algorithms are worthwhile keeping in mind for the common cases where the losses $L_i$ and hence $R(b)$ are convex, as is the case for exponential loss. Convexity and the ensuing stability of minimization may get obscured when an $F$-loss is decomposed into a link function and a $p$-loss. On the other hand, in those cases where the weight function $\omega(q)$ is tailored for classification with unequal cost of misclassification, the $F$-loss is unlikely to be convex, and stability may be gained by using Fisher scoring instead, whose expected Hessian is always positive definite.

### 4.2 Newton updates and IRLS for $p$-losses and link functions

If $q(F)$ is the link function, we abbreviate

$$q_n = q(x_n^T b), \quad q'_n = q'(x_n^T b), \quad q''_n = q''(x_n^T b),$$

and hence

$$\partial_b q_n = q'_n x, \quad \partial^2_b q_n = q''_n xx^T.$$

Making use of the structure theorem for $p$-losses (Section 2.1), the ingredients for Newton updates become

$$R(b) = \frac{1}{N} \sum_{n=1..N} \left[ y_n L_1(1 - q_n) + (1 - y_n) L_0(q_n) \right],$$

$$\partial_b R(b) = -\frac{1}{N} \sum_{n=1..N} (y_n - q_n) \omega(q_n) q'_n x_n,$$

$$\partial^2_b R(b) = \frac{1}{N} \sum_{n=1..N} \left[ \omega(q_n) q''_n^2 - (y_n - q_n) \left( \omega'(q_n) q''^2 + \omega(q_n) q''_n' \right) \right] x_n x_n^T.$$
The current weights and current responses for IRLS are therefore

\[ w_n = \omega(q_n) q_n^2 - (y_n - q_n) \left( \omega'(q_n) q_n^2 + \omega(q_n) q_n'' \right), \quad (14) \]
\[ z_n = (y_n - q_n) \omega(q_n) q_n'/w_n. \quad (15) \]

We note that the loss functions enter only through the weight function \( \omega(q) \). Again the weights \( w_n \) can be negative when the \( F \)-loss is not strictly convex.

### 4.3 Fisher scoring and IRLS for \( p \)-losses and link functions

For Fisher scoring one replaces the Hessian \( \partial^2 R(b) \) with its expectation, assuming the current estimates of the conditional probabilities of the labels to be the true ones: \( P[Y = 1|X = x_n] = q_n \). This means \( E_{y_n}(y_n - q_n) = 0 \), hence the weights and the current response simplify as follows:

\[ w_n = \omega(q_n) q_n^2 , \quad (16) \]
\[ z_n = (y_n - q_n)/q_n'. \quad (17) \]

Some observations:

- The function \( \omega(q) \) is the contribution to the current weights \( w_n \) due to the \( p \)-loss.
- The function \( q'^2 \) is the contribution to the current weights \( w_n \) due to the link function.
- The losses do not enter in the current response \( z_n \).
- The observed responses \( y_n \) do not enter in the current weights \( w_n \).

The last point should be puzzling from a machine learning perspective: a widely held belief about boosting is that it up- or down-weights cases according to whether they have been misclassified in the latest update. Be this as it may, the boosting weights for exponential loss, \( w_n = \exp(-y_n^* x_n^T b) \), are a function of the observed responses \( y_n \), whereas Fisher scoring weights are not; they are functions of \( \omega(q) \) and \( q'(F) \) only.

From a machine learning perspective again, it should be curious to learn that there exist \( F \)-losses for which the dependence of the weights on the responses disappears for structural reasons: this is the case for so-called “canonical links”, which are well-known in statistics from generalized linear models. They have a generalization to certain combinations of \( p \)-losses and link functions as we will show now.

### 4.4 Canonical links: equality of observed and expected Hessians

It is well-known that for the usual Bernoulli model with the logistic link Newton iterations and Fisher scoring iterations are identical. This is not immediately apparent from the equations above. We examine under what condition on the combination of \( p \)-losses and link functions, Newton steps and Fisher scoring steps are identical. The requirement is

\[ R''(b) = E_y [R''(b)] . \]
This condition is equivalent to requiring the weight $w_n$ in Equation (14) to be independent of $y_n$, which in turn is equivalent to
\[
\omega'(q) q'^2 + \omega(q) q'' = 0 .
\]
The left side of this equation equals $(\omega(q) q')'$, hence $\omega(q) q'$ is a constant, which we may choose to be 1:
\[
\omega(q) q' = 1 .
\]
This equation is easily solved for the inverse $F = F(q)$ of $q = q(F)$. It satisfies
\[
F'(q) = \omega(q) = L'_1(1 - q) + L'_0(q) ,
\]
where the second equality follows from $L'_1(1 - q) + L'_0(q) = (1 - q)\omega(q) + q\omega(q)$. The solution, unique up to an irrelevant additive constant, is therefore
\[
F(q) = \int^q \omega(q) dq = L_0(q) - L_1(1 - q) .
\]
We summarize:

**Proposition:** For any $p$-loss there exists a canonical link function $q(F)$. The link function is unique up a shift $q(F - c)$.

Here are the usual examples in the Beta family of $p$-losses for $\alpha = \beta = 1, 0, -1/2$ (Section 2.9):

- **Squared error loss:** $L_1(1 - q) = (1 - q)^2$ and $L_0(q) = q^2$, hence the inverse link is
  \[
  F(q) = 2q - 1 .
  \]
The canonical link is essentially the identity transform.

- **Log-loss:** $L_1(1 - q) = -\log(1 - q)$ and $L_0(q) = -\log(q)$, hence the inverse link is
  \[
  F(q) = \log \frac{q}{1 - q} .
  \]
The canonical link is the usual logistic transform.

- **Boosting loss:** $L_1(1 - q) = ((1 - q)/q)^{1/2}$ and $L_0(q) = (q/(1 - q))^{1/2}$, hence the inverse link is
  \[
  F(q) = \left( \frac{q}{1 - q} \right)^{1/2} - \left( \frac{1 - q}{q} \right)^{1/2} .
  \]
The link function is
  \[
  q(F) = \frac{1}{2} + \left( \frac{F}{2} \right) \left( \left( \frac{F}{2} \right)^2 + 1 \right)^{-1/2} .
  \]
The $F$-loss is
  \[
  L(F) = \frac{F}{2} + \left( \left( \frac{F}{2} \right)^2 + 1 \right)^{1/2} .
  \]
We see that the decomposition of the exponential loss into a $p$-loss and a link function does not result in a canonical pairing in which the observed and the expected Hessian are identical. For exponential loss Newton iterations are not identical with Fisher scoring iteration.

We conclude by noting that the converse of the above proposition holds also: prescribing a link function as canonical essentially determines the $p$-loss.

**Proposition:** For any link function $q(F)$ there exists a $p$-loss for which $q(F)$ is canonical. The $p$-loss is unique up to an additive constant.

This is immediately seen from Equation (18): $F'(q) = \omega(q)$, and the fact that, by Equations (7), $\omega(q)$ determines $L_1(1 - q)$ and $L_0(q)$ up to a constant.

### 4.5 Convexity of $F$-losses

For canonical links, the observed Hessian and its expectation coincide. Since the expected Hessian is always non-negative definite, it follows that for canonical links the $F$-loss is always convex. Convexity, however, occurs more generally for certain combinations of link function and $p$-loss. For example, we saw above that the exponential loss is not obtained through a canonical link function from its $p$-loss, that is, observed and expected Hessians do not coincide, yet exponential loss as a $F$-loss is convex.

When the $F$-loss is convex, Newton steps may be more efficient than Fisher scoring steps. It is therefore useful to examine when convexity holds for $F$-losses.

We give a general condition under which a pairing of a link function and a $p$-loss results in a convex $F$-loss. The condition is derived by requiring the weights in the Hessian of Equation (14) to be non-negative: $w_n \geq 0$, that is,

$$\omega(q) q^2 - (y - q) \left( \omega'(q) q^2 + \omega(q) q'' \right) \geq 0.$$  

This requirement results in two redundant inequalities, one for $y = 1$ and one for $y = 0$. We summarize:

**Proposition:** A combination of $p$-loss and link function results in a convex loss $R(b)$ if $\omega(q)$ and $q(F)$ satisfy

$$-\frac{1}{q} \leq \frac{\omega'}{\omega} + \frac{q''}{q^2} \leq \frac{1}{1 - q}.$$  

As a corollary, the combination of $p$-losses in the Beta family with equal exponents,

$$\omega(q) = q^{\alpha - 1} (1 - q)^{\beta - 1},$$

and scaled logistic links,

$$q(F) = 1/(1 + e^{-F/s}), \quad F(q) = s \log \frac{q}{1 - q},$$
result in convex $F$-losses iff

$$-1 \leq \alpha, \beta \leq 0, \quad s > 0.$$ 

These $p$-losses result in convex $F$-losses for any scaled logistic link. Special cases are exponential loss ($\alpha = \beta = -1/2, s = 1/2$) and log-loss with the logit scale ($\alpha = \beta = 0, s = 1$). But squared error loss ($\alpha = \beta = 1$) does not result in convex $F$-losses with any scaled logistic link.

### 4.6 The role of scale of the link function **

Intuitively, link functions that differ only in scale should result in identical fitted linear models, with coefficient vectors that differ only in scale also. Let $q(F)$ be a fixed link and

$$q_s(F) = q(F/s)$$

the scale family it generates. The corresponding criteria, indexed by the choice of $s$, are:

$$R_s(b) = \frac{1}{N} \sum_{i=1}^{N} L(y_n | q(x_n^T b / s)) .$$

From this it is clear that

$$\argmin_b R_s(b) = s \cdot \argmin_b R(b) .$$

A check of the Newton and Fisher steps confirms that the increments for $R_s(b)$ differ from those for $R(b)$ exactly by a factor $s$.

### 5 Stagewise Fitting of Additive Models

In FHT’s (2000) interpretation of boosting is a form of stagewise forward regression for building a possibly very large additive model. While Freund and Schapire conceived boosting as a weighted majority vote among a collection of weak learners, FHT see it as the thresholding of a linear combination of basis functions, in other words, of an additive model. We adopt FHT’s view of boosting and adapt the above IRLS iterations to stagewise forward additive modeling.

For additive modeling one needs a set $\mathcal{F}$ of possible basis functions $f(x)$. An additive model is a linear combination of elements of $\mathcal{F}$: $F(x) = \sum_k b_k f_k(x)$. One does not assume that $\mathcal{F}$ is a linear space because in a typical application of boosting $\mathcal{F}$ would be a set of trees with a certain structure, such as stumps (trees of depth one), that are not closed under linear combinations.

Stagewise fitting is the successive acquisition of terms $f^{(k)} \in \mathcal{F}$ given that terms $f^{(1)}, \ldots, f^{(k-1)}$ for a model $F^{(k-1)} = \sum_{k=1..K-1} b_k f^{(k)}$ have already been acquired. In what follows
we write $F_{old}$ for $F^{(K-1)}$, $f$ for $f^{(K)}$, $b$ for $b_K$, and finally $F_{new} = F_{old} + b f$. For a loss function
\[ R(F) = \mathbf{E}_{x,y} \mathbf{L}(y|q(F(x))) , \]
the conceptual goal is to find
\[ (f, b) = \operatorname{argmin}_{f \in \mathcal{F}, \ b \in \mathbb{R}} R(F_{old} + b f) . \]
Given a training sample $\{(x_n, y_n)\}_{n=1..N}$, the loss is estimated by
\[ \hat{R}(F_{old} + b f) = \frac{1}{N} \sum_{n=1..N} \mathbf{L}(y_n | q(F_{old}(x_n) + b f(x_n))) . \]

With these modifications stagewise fitting can proceed according to the IRLS scheme in such a way that each IRLS step produces a new term $f$, a coefficient $b$, and an update $F_{new} = F_{old} + b f$ based on current weights $w_n$, current estimated class 1 probabilities $q_n$, and current responses $z_n$, as follows:

\[
\begin{align*}
F_n &= F_{old}(x_n), \\
n_q &= q(F_n), \\
n'_q &= q'(F_n), \\
n''_q &= q''(F_n), \\
w_n &= \omega(q_n)q''_n^2 - (y_n - q_n) \left( \omega'(q_n)q''_n + \omega(q_n)q''_n \right), \\
z_n &= (y_n - q_n) \omega(q_n)q''_n/w_n .
\end{align*}
\]

In practice, $f$ is the result of some heuristic search procedure such as greedy tree construction or, in the simplest case, search for the best fitting stump (a tree with only one split). Whatever this may be, we denote it by
\[
f \leftarrow \text{feature-proposer}(X, W, z) .
\]

We can think of $f$ as a proposal for a derived predictor vector $f = (f(x_1), ..., f(x_N))^T$ which is to be used instead of $X$ in the IRLS step at the current stage. The remaining task is to optimize the coefficient $b$ for $f$. Sometimes, the coefficient $b$ is absorbed in $f$ because the feature-proposer may find $b$ at the same time as it performs the search, so no optimization is necessary. At other times, the coefficient $b$ has to be optimized explicitly, for example through a line search:
\[ b = \operatorname{argmin}_b \frac{1}{N} \sum_{n=1..N} \mathbf{L}(y_n | q(F_n + b f_n)) , \]
or approximately by a simple weighted LS regression of $z$ on $f$:
\[ b = \frac{z^T W f}{f^T W f} . \]

We would like to think of the feature-proposer as estimator of a weighted conditional expectation, but this is usually possible at most in an approximate or heuristic sense. Two standard examples illustrate this point:
• In “real Adaboost”, the class $\mathcal{F}$ is a set of trees and the feature-proposer is a heuristic search consisting of a tree growing phase followed by a tree pruning phase.

• In “discrete Adaboost”, the class $\mathcal{F}$ is the set of indicator functions of decision regions obtained by thresholding trees. The feature-proposer is then the same heuristic tree construction as in real Adaboost, followed by thresholding.

When $\mathcal{F}$ is the set of stumps, the search for an optimal stump can be performed exhaustively. In this case real and discrete Adaboost are almost identical, except for the treatment of the intercept: real Adaboost can adjust the intercept at each step due to the fitting of two constants (one on either side of the split), whereas discrete Adaboost can fit only one constant, on the area where the indicator is 1.

In general the role of the feature-proposer is to propose descent directions from a stock of possible choices, the set $\mathcal{F}$. The behavior of the feature-proposer and the nature of the set $\mathcal{F}$ are of course crucial for the performance of the boosting scheme based on them.

6 Link Functions with Baseline Class Probabilities

According to some reports boosting does not perform well when there is “label noise” in the data. In simulations label noise is generated by flipping the class $0 \leftarrow 1$ with a fixed probability $u (< 1/2)$. As a result, when the original true class 1 probability is 1, the label noise lowers it to $1 - u$, and similarly if the original class 1 probability is 0, the label noise raises it to $u$.

Following the example of common practice in Rasch models (xxx need citation), label noise can be modeled as follows: Define a two-parameter family of links associated with any given link $q$ that satisfies $q(-\infty) = 0$ and $q(+\infty) = 1$:

$$q^{(u,v)}(t) = u + (v - u) q(t).$$

We note: $q^{(u,v)}(-\infty) = u$ and $q^{(u,v)}(+\infty) = v$. That is, $u$ is the baseline probability of class 1, and $(1 - v)$ is the baseline probability of class 0. Label noise is the special case where $v = 1 - u$.

Estimates of $u$ and $v$ can be computed by expanding the Newton algorithm to include these additional parameters:

$$u_{new} \leftarrow u_{old} - \frac{\sum_n \omega(q_n^{(u,v)})(2q_n^{(u,v)} - 1)(1 - q_n)}{\sum_n \omega(q_n^{(u,v)})(2q_n^{(u,v)} - 1)(1 - q_n)}$$

[**** unfinished]
7 Examples

7.1 Examples of biased linear fits with successful classification

We illustrate the fact that linear models can be unsuitable as global models and yet successful for classification at specific thresholds or misclassification costs. To this end we recreate an artificial example of Hand and Vinciotti (2003) and show that our methods for tailoring loss functions to specific classification thresholds allows us to estimate optimal boundaries. We then show that the Hand and Vinciotti (2003) scenario is a fair approximation to some well-known real data, the Pima Indian diabetes data (UCI Machine Learning database 2003).

We start with a linear model, that is, a composition $q(b^T x)$ of a linear function, $b^T x$, and a link function, $q()$, as in logistic regression. Note that the level curves, $\{ x | q(b^T x) = c \}$, are parallel lines. This fact can be used to construct scenarios $p(x)$ for which the model $q(b^T x)$ is globally inadequate and yet can produce excellent classification boundaries. Such a construction was demonstrated by Hand and Vinciotti (2003): they designed $p(x)$ as a surface with the shape of a smooth spiral staircase on the unit square, as depicted in Figure (4).
The critical feature of the surface is that the optimal classification boundaries $p(x) = c$ are all linear but not parallel. The absence of parallelism renders any linear model $q(b^T x)$ unsuitable as a global fit, but the linearity of the classification boundaries allows linear models to describe these boundaries, albeit every level requires a different linear model. The point of Hand and Vinciotti’s (2003) and our schemes is to home in on these level-specific models.

In recreating Hand and Vinciotti’s (2003) example, we simulated 4,000 data points whose two predictors were uniformly distributed in the unit square, and whose class labels had a conditional class 1 probability $p(x_1, x_2) = \cos^{-1}(x_1/(x_1^2 + x_2^2)^{1/2})/(\pi/2)$. We fitted a linear model with the logistic link function $q(t) = 1/(1 + \exp(t))$, using a $p$-loss in the Beta family with $\alpha = 29$ and $\alpha/\beta = 0.3/0.7$ in order to home in on the $c = 0.3$ boundary. Figure 4, which is similar to Hand and Vinciotti’s (2003) Figure xxx, shows the success: the estimated boundary is close to the true 0.3-boundary. The figure also shows that the 0.3-boundary estimated with the log-loss of logistic regression is essentially parallel to the 0.5-boundary, which is sensible because logistic regression is bound to find a compromise model which,
Figure 6: The Pima Indian Diabetes Data, BODY against PLASMA. The highlights represent slices with near-constant estimated class 1 probability $p$: $c - \epsilon \leq p \leq c + \epsilon$. The values of $c$ in the slices increase left to right and top to bottom. The class 1 probabilities were estimated with 20-nearest-neighbor estimates. Glyphs: open squares = no diabetes, class 0; filled circles = diabetes, class 1; small dots = points outside the slice.
for reasons of symmetry between the two labels, should be a linear model with level lines roughly parallel to the true 0.5-boundary.

Our real data example shows that Hand and Vinciotti’s stylized scenario is met for real in the well-known Pima Indians diabetes data when restricting attention to the two major predictors (called “PLASMA” and “BODY”). Figure 5 shows a scatterplot of these two predictors with glyph-coding of the classes. Also shown are the
References


Appendix 1: Proof of the Characterization of $p$-Losses

The original proof is in Shuford, Albert and Massengill (1966).

Necessary conditions for being a $p$-loss are stationarity for the first derivative and a positive second derivative at the stationary solution of $L(p|q)$:

\[
\frac{\partial}{\partial q} L(p|q) = -pL_1'(1-p) + (1-p)L_0'(p) = 0
\]

\[
\frac{\partial^2}{\partial q^2} L(p|q) = pL''_1(1-p) + (1-p)L''_0(p) > 0
\]

In order to satisfy the stationarity condition, we define $\omega(p)$ by

\[
\frac{L_1'(1-p)}{1-p} = \frac{L_0'(p)}{p} = \omega(p)
\]

and solve for $L_1'$ and $L_0'$:

\[
L_1'(1-p) = (1-p) \omega(p), \quad L_0'(p) = p \omega(p).
\]

We check the second derivative using these identities and observing $\partial^2 L_1'(1-q) = -L_1''(1-q)$:

\[
\frac{\partial^2}{\partial q^2} L(p|q) = pL''_1(1-p) + (1-p)L''_0(p)
\]

\[
= p \omega(p) - p(1-p) \omega'(p) + (1-p) \omega(p) + (1-p)p \omega'(p)
\]

\[
= \omega(p).
\]

We see that the second derivatives are positive iff $\omega(p) > 0$.

Furthermore, one can show that the local minimum is indeed global. To this end, note that there exists only one local minimum in the interior of the 0-1 interval. The only other minima would have to be on the boundary 0 or 1, where the derivative of the loss could be non-zero. Therefore, we only have to check that the derivative is negative near 0, and positive near 1. This is immediate from the following identity:

\[
\frac{\partial}{\partial q} L(p|q) = [-p(1-q) + (1-p)q] \omega(q).
\]