Class-Weighted Classification: Trade-offs and Robust Approaches

Ziyu Xu[1] **Chen Dan**[2] **Justin Khim**[1] **Pradeep Ravikumar**[1]

Abstract

We address imbalanced classification, the problem in which a label may have low marginal probability relative to other labels, by weighting losses according to the correct class. First, we examine the convergence rates of the expected excess weighted risk of plug-in classifiers where the weighting for the plug-in classifier and the risk may be different. This leads to irreducible errors that do not converge to the weighted Bayes risk, which motivates our consideration of robust risks. We define a robust risk that minimizes risk over a set of weightings, show excess risk bounds for this problem, and demonstrate that particular choices of the weighting set leads to a special instance of conditional value at risk (CVaR) from stochastic programming, which we call label conditional value at risk (LCVaR). Additionally, we generalize this weighting to derive a new robust risk problem that we call label heterogeneous conditional value at risk (LHCVaR). Finally, we empirically demonstrate the efficacy of LCVaR and LHCVaR on improving class conditional risks.

1. Introduction

Classification is a fundamental problem in statistics and machine learning, including scientific problems such as cancer diagnosis and satellite image processing as well as engineering applications such as credit card fraud detection, handwritten digit recognition, and text processing (Khan et al., 2001; Lee et al., 2004), but modern applications have brought new challenges. In online retailing, websites such as Amazon have hundreds of thousands or millions of products to taxonomize (Lin et al., 2018). In text data, the distribution of words in documents has been observed to follow a power law in that there are many labels with few instances (Zipf, 1936; Feldman, 2019). Similarly, image data also a long tail of many classes with few examples (Salakhutdinov et al., 2011; Zhu et al., 2014). In such settings, the classes with smaller probabilities are generally classified incorrectly more often, and this is undesirable when the smaller classes are important, such as rare forms of cancer, fraudulent credit card transactions, and expensive online purchases. Thus, we need modern classification methods that work well when there are a large number of classes and when the class-wise probabilities are imbalanced.

When faced with such class imbalance a popular approach in practice is to choose a metric other than zero-one accuracy, such as precision, recall, F_β-measure (Van Rijsbergen, 1974, 1979), which explicitly take class conditional risks into account, and train classifiers to optimize this metric. A difficulty with this approach however is that the right metric for imbalanced classification is often not clear. A related class of approaches keep the zero-one accuracy metric but modifies the samples instead. The popular algorithm SMOTE (Chawla et al., 2002) performs a type of data augmentation for a minority class, i.e., a class with lower probability, and sub-samples the large classes. This has led to variants with different forms of data augmentation (Zhou & Liu, 2006; Mariani et al., 2018), but from a theoretical perspective, these methods remain poorly understood.

A much simpler approach, which is also related to the approaches above, is class-weighting, in which different costs are incurred for mis-classifying samples of different labels. Practically, this is a natural approach because it is often possible to assign different costs to different classes. For example, the average fraudulent credit card transaction may cost hundreds of dollars, or in online retailing, failing to show a customer the correct item causes the company to lose out on the profit of selling that item. Thus, a good classifier should be fairly sensitive to possibly fraudulent transactions, and online retailers should prioritize displaying high-profit products. As a result, class-weighting has been studied in a variety of settings, including modifying black-box classifiers, SVMs, and neural networks (Domingos, 1999; Lin et al., 2002; Scott, 2012; Zhou & Liu, 2006). Additionally, class-weighting has been observed to be useful for estimating class probabilities, since class-weighting amounts to adjusting decision thresholds (Wang et al., 2008; Wu et al., 2010; Wang et al., 2019).

[1] Machine Learning Department, Carnegie Mellon University, Pennsylvania, United States [2] Computer Science Department, Carnegie Mellon University, Pennsylvania, United States. Correspondence to: Ziyu Xu <ziyux@cs.cmu.edu>.

Proceedings of the 37^{th} International Conference on Machine Learning, Online, PMLR 119, 2020. Copyright 2020 by the author(s).

A crucial caveat with cost-weighting however is the right choice of costs is often not clear, and with any one choice of costs, the performance of the corresponding classifier might suffer for some other, perhaps more suitable, choices of costs.

In this paper, we use cost-weighting for imbalanced classification in three ways. We start by examining a weighted sum of class-conditional risks, i.e., the risks conditional on the class Y taking some specific value i. This allows us to upweight a minority class to achieve better performance on the minority examples. We then provide an illuminating analysis of the fundamental tradeoffs that occur with any single choice of costs.

Since we may not understand precisely which weighting q to pick, we examine a robust risk that is a supremum of the weighted risks over an uncertainty set Q of possible weights. This objective can be interpreted as a class-wise distributionally robust optimization problem where we ask for robustness over the marginal distribution of Y. This leads to a minimax problem, for which we provide generalization guarantees. We also note that a standard gradient descent-ascent algorithm may solve the optimization problem when the risk is convex in the classifier parameters.

Finally, we show that for a natural class of uncertainty sets, the robust risk reduces to what call label conditional value at risk (LCVaR). We highlight a connection to conditional value at risk (CVaR), which is a well-studied quantity in portfolio optimization and stochastic programming parametrized by an α in $(0, 1)$ (Rockafellar et al., 2000; Shapiro et al., 2009). Further, we propose a generalization that we call label heterogeneous conditional value at risk (LHCVaR) that allows for different parameters α_i for each class i. To the best of our knowledge, this has not been examined previously, and it could possibly be used more broadly. To give an example in portfolio optimization, we may wish to treat risks arising from different types of assets, e.g., large-cap stocks versus small-cap stocks or domestic debt versus international debt, differently. Next, we show that the dual form for LHCVaR is similar to that for LCVaR as long as the heterogeneity is finite-dimensional, and this leads to an unconstrained optimization problem. Finally, we examine the efficacy of LCVaR, and LHCVaR on real and synthetic data.

The rest of the paper is outlined as follows. In Section 2, we discuss our problem setup. In Section 3, we examine weighting in plug-in classification. In particular, we elucidate the fundamental trade-off in weighted classification and its methodological implications. In Section 4, we examine a robust version of the weighted risk problem, including generalization guarantees and connections to stochastic programming. In Section 5, we provide numerical results, and we conclude with a discussion in Section 6. Additional proofs and results in related settings are deferred to the appendices.

1.1. Further Related Work

We briefly review other research related to imbalanced classification, but for a far more exhaustive treatment, see a survey of the area (He & Garcia, 2009; Fernández et al., 2018). First, two other methods may be employed to solve imbalanced classification problems. The first is class-based margin adjustment (Lin et al., 2002; Scott, 2012; Cao et al., 2019), in which the margin parameter for the margin loss function may vary by class. Broadly, margin adjustment and weighting may both be considered loss modification procedures. The second method is Neyman-Pearson classification, in which one attempts to minimize the error on one class given a constraint on the worst permissible error on the other class (Rigollet & Tong, 2011; Tong, 2013; Tong et al., 2016).

An important topic related to our paper but that has not been well-connected to imbalanced classification is robust optimization. Robust optimization is a well-studied topic (Ben-Tal & Nemirovski, 1999, 2003; Ben-Tal et al., 2004, 2009). A variant that has gained traction more recently is distributionally robust optimization (Ben-Tal et al., 2013; Bertsimas et al., 2014; Namkoong & Duchi, 2017). Unsurprisingly, CVaR, as a coherent risk measure, has been previously connected to distributionally robust optimization (Goh & Sim, 2010). Distributionally robust optimization generally and CVaR specifically have also previously been used in machine learning to deal with imbalance (Duchi et al., 2018; Duchi & Namkoong, 2018), but in these works, the imbalance was considered to exist in the covariates, whether known to the algorithm or not. These are motivated by the recent push toward fairness in machine learning, in particular so that ethnic minorities do not suffer discrimination in high-stakes situations such as loan applications, medical diagnoses, or parole decisions, due to biases in the data.

2. Preliminaries

2.1. Classification with Imbalanced Classes

In this section, we briefly go over the problem setup. First, we draw samples from the space $\mathcal{Z} = \mathcal{X} \times \mathcal{Y}$. For our purposes, we are interested in $\mathcal{Y} = \{0, 1\}$ or $\mathcal{Y} = \{1, \ldots, k\}$. Note there are two slightly different mechanisms for the data-generating process that are considered in imbalanced classification and Neyman-Pearson classification. In the first, we are given n i.i.d. samples $(X_1, Y_1), \ldots, (X_n, Y_n)$ from a distribution $P_{X,Y}$. Here, we let $p_i = \mathbb{P}(Y = i)$ be the probability of class i. Additionally, we sometimes refer to the vector of class probabilities as p. This is our framework of interest, since it corresponds to standard assumptions in nonparametric statistics and learning theory. In the alternative framework, we are given n_i samples $(X_1, i), \dots, (X_{n_i}, i)$ from each marginal distribution $P_{X|Y=i}$. The probability of class i in this case is then known: $p_i = \widehat{p}_i = n_i/n$. For the most part, these two mechanisms yield similar results, but the analyses differ slightly. To streamline the presentation, we only consider the first case in the main paper, although we give a result for the alternative framework in the appendix that illustrates the difference.

2.2. Class Conditioned Risk

We are interested in finding a good classifier $f : \mathcal{X} \to \mathcal{D} \supseteq \mathcal{Y}$ in some function space $\mathcal{F}$, such as linear classifiers or neural networks. In this section, we establish our risk measures of interest. In general, we want to minimize the expectation of some loss function $\ell : \mathcal{F} \times \mathcal{Z} \to [0, 1]$, which we call risk and denote $R(f) = \mathbb{E}[\ell(f, Z)]$. Analogously, we define the class-conditioned risk for class i to be

$$R_{\ell,i}(f) = \mathbb{E}\left[\ell(f, Z) | Y = i\right].$$

At this point, we make some observations for plug-in classification and empirical risk minimization. In the plug-in classification results, we consider the zero-one loss $\ell_{01}(f, z) = \mathbf{1}\{f(x) \neq y\}$, and for our results on empirical risk minimization, we are primarily interested in convex surrogate losses. For simplicity, when ℓ is clear from context, or a statement is made for a generic ℓ, we will denote this as R_i.

Now, we can work toward defining weighted risks. We defined Observe that we can relate the risk to the class-conditioned risk by $R(f) = \mathbb{E}\left[R_Y(f)\right] = \sum_{i \in \mathcal{Y}} p_i R_i(f)$. An important part of our paper is an examination of class-weighted risk.

Definition 1. *Let* $q = (q_1, \dots, q_{|\mathcal{Y}|})$ *be a vector such that* $q_i \geq 0$ *for all* i *and* $\mathbb{E}[q_Y] = \sum_{i \in \mathcal{Y}} q_i p_i = 1$. *Then, the* q*-weighted risk is*

$$R_q(f) = \mathbb{E}\left[q_Y R_Y(f)\right] = \sum_{i \in \mathcal{Y}} q_i p_i R_i(f).$$

Note that the usual risk is recovered by setting $q = (1, \dots, 1)$.

2.3. Plug-in Classification

In this section, we discuss weighted plug-in classification. For plug-in, we restrict our attention to the binary classification case of $\mathcal{Y} = \{0, 1\}$, and the primary quantity of interest is usually the one-zero risk $R_{01}(f)$ i.e the risk under $\ell_{0,1}$. In general, the risk for the best classifier is nonzero because for a given x in $\mathcal{X}$, there is some probability it may take the value 0 or 1.

As a result, we need a way to discuss the convergence of our estimator to the best possible estimator. We define the regression function η by $\eta(x) = \mathbb{P}\left(Y = 1 | X = x\right)$. Now, the Bayes optimal classifier is the classifier that minimizes the risk, and it is defined by $f^*(x) = \mathbf{1}\left\{\eta(x) > 1/2\right\}$. The minimum possible risk is called the Bayes risk and denoted by $R^* = R(f^*)$, and generally we focus on minimizing the excess risk $\mathcal{E}(f) = R(f) - R^*$.

Following the form of the Bayes classifier, a plug-in estimator $\widehat{f}$ attempts to estimate the regression function η by some $\widehat{\eta}$ and then "plugs in" the result to a threshold function. Thus, $\widehat{f}$ has the form $\widehat{f}(x) = \mathbf{1}\left\{\widehat{\eta}(x) > 1/2\right\}$, which is analogous to the form of the Bayes classifier. For additional background on plug-in estimation, see, e.g., (Devroye et al., 1996).

At this point, we wish to define the weighted versions of Bayes classifier, Bayes risk, plug-in classifier, and excess risk. For brevity, define the threshold $t_q = q_0/(q_0 + q_1)$. First, we consider the Bayes classifier.

Lemma 1. *Let* $q = (q_0, q_1)$ *be a weighting. The Bayes optimal classifier for* q*-weighted risk is* $f^*_q(x) = \mathbf{1}\left\{\eta(x) > t_q\right\}$.

The proof, along with proofs of other subsequent results on plug-in classification, appears in the appendix. In this case, we denote the Bayes risk by $R^*_q = R_q(f^*_q)$. Lemma 1 reveals that the Bayes classifier is a plug-in rule, and analogously, we see that a plug-in estimator in the weighted case takes the form $\widehat{f}_q(x) = \mathbf{1}\left\{\widehat{\eta}(x) > t_q\right\}$. Consequently, we define excess q-risk for an empirical classifier $\widehat{f}$. The excess q-risk for an empirical classifier is $\mathcal{E}_q(\widehat{f}) = R_q(\widehat{f}) - R^*_q$, and note that we are interested in bounding the expected excess q-risk for plug-in estimators.

2.4. Empirical Risk Minimization

In this section, we define empirical quantities that we need for empirical risk minimization, particularly the weighted and robust risks. We consider $\mathcal{Y} = \{1, \dots, k\}$. We define the empirical class-conditioned risk by $\widehat{R}_i = (1/N_i) \sum_{j=1}^n \ell(f, z_j) \mathbf{1}\left\{y_i = i\right\}$ where $N_i = \sum_{j=1}^n \mathbf{1}\left\{y_j = i\right\}$. Let $\widehat{p}_i = N_i/n$ denote the empirical proportion of observations of class i, and let q be a weight vector. The empirical q-weighted risk is

$$\widehat{R}_q(f) = \sum_{i=1}^k q_i \widehat{p}_i \widehat{R}_i(f).$$

The empirical Q-weighted risk is defined analogously by $\widehat{R}_Q = \sup_{q \in Q} \widehat{R}_q(f)$. This problem is convex in f when the loss ℓ is convex and concave in q due to linearity; so one may solve the resulting saddle-point problem with standard techniques such as gradient descent-ascent, which we give in the appendix.

Often in empirical risk minimization, generalization bounds are provided, i.e., a bound on the true risk of a classifier f in $\mathcal{F}$ in terms of its empirical risk and a variance term. To bring our results closer to those of plug-in estimation, we also consider a form of excess risk. To distinguish the two, define the excess $(\mathcal{F}, Q)$-weighted risk to be $\mathcal{E}_Q(\mathcal{F}) = R_Q(\widehat{f}) - R_Q(f^*_Q)$ where here $\widehat{f}$ is the Q-weighted empirical risk minimizer in $\mathcal{F}$ and f^*_Q is the population Q-weighted risk minimizer in $\mathcal{F}$. Beyond the robust formulation, the key difference between excess q-weighted risk and excess $(\mathcal{F}, Q)$-weighted risk is that in the former we compete with the true regression function, and in the latter we compete with the best classifier in $\mathcal{F}$.

One additional tool we need for empirical risk minimization is a measure of function class complexity, and a typical measure of the expressiveness of a function class is Rademacher complexity. The empirical Rademacher complexity given a sample $(X_1, Y_1), \ldots, (X_n, Y_n)$ is

$$\widehat{\mathfrak{R}}_n(\mathcal{F}) = \mathbb{E}_\sigma \sup_{f \in \mathcal{F}} \sum_{i=1}^{n} \sigma_i f(X_i),$$

where the expectation is taken with respect to the σ_i, which are Rademacher random variables. The Rademacher complexity is $\mathfrak{R}_n(\mathcal{F}) = \mathbb{E}\widehat{\mathfrak{R}}_n(\mathcal{F})$, where the expectation is with respect to the X_i random variables.

Finally, we make one note about the loss for our empirical risk minimization results. For binary classification, one can obtain bounds for any bounded loss function that is Lipschitz continuous in $f(x)$. Since we present multiclass results, we use the multiclass margin loss, which is a bounded version of the multiclass hinge loss (Mohri et al., 2012). Here, it is assumed that for each i in $\mathcal{Y}$, the function f outputs a score $f_i(x)$, and the chosen class is $\text{argmax}_{i \in \mathcal{Y}} f_i(x)$. The multiclass margin loss is defined as $\ell_{\text{mar}}(f, z) = \Phi\left(f_y(x) - \max_{y' \neq y} f_{y'}(x)\right)$ where $\Phi(a) = \mathbf{1}\{a \leq 0\} + (1-a)\mathbf{1}\{0 < a \leq 1\}$. For simplicity, we ignore the margin parameter, usually denoted by ρ, and treat it as 1 in our results. Finally, we define the projection set $\Pi_1(\mathcal{F}) = \{x \mapsto f_y(x) : y \in \mathcal{Y}, f \in \mathcal{F}\}$.

3. Tradeoffs with Class Weighted Risk

In this section, we examine weighted plug-in classification, and we have two main results. First, we show that weighted plug-in classification enjoys essentially the same rate of convergence as unweighted plug-in classification, although there is dependence on the chosen weights. Second, there is a fundamental trade-off in that optimizing for one set of weights q may lead to suboptimal performance for another set of weights q'.

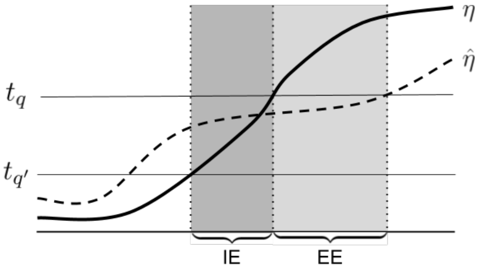


Figure 1: The irreducible error (IE) and estimation error (EE). The irreducible error is the measure of the set of x where $\eta(x)$ is between thresholds of q' and q, which does not depend on $\widehat{\eta}$. The estimation error is the measure of the x for which $\widehat{\eta}(x)$ and $\eta(x)$ lead to different plug-in estimates.

3.1. Excess Risk Bounds

We start with the excess risk bound for plug-in estimators when the weighting is well-specified.

Proposition 1. *Suppose the regression function η is β-Hölder. Then, the q-weighted excess risk of $\widehat{f}_q$ satisfies*

$$\mathbb{E}\mathcal{E}_q(\widehat{f}_q) \leq O\left((q_0 + q_1) n^{-\frac{\beta}{2\beta+d}}\right).$$

Here, we see that the upper bound depends linearly on q_0 and q_1. This implies that when we increase the weight for a class with few examples, then our bound on the excess risk increases. While previous cost weighting setups have normalized the sum of weights (Scott, 2012), our normalization scheme is computed with respect to prior probabilities on each class as well, and consequently we explicitly include q_0, q_1 in our bound. Our choice of domain for weights is defined in Section 4.

Now, we turn to our second task: examining the weighted excess risk of the $\widehat{f}_q$ under a different weighting q'. Observe that we can decompose the excess risk as

$$\mathbb{E}\mathcal{E}_{q'}(\widehat{f}_q) = \underbrace{\mathbb{E}R_{q'}(\widehat{f}_q) - R_{q'}(f^*_q)}_{\text{estimation error}} + \underbrace{R_{q'}(f^*_q) - R_{q'}(f^*_{q'})}_{\text{irreducible error}}$$
$$=: (\text{EE}) + (\text{IE}). \quad (1)$$

Unsurprisingly, we see that an error term that is constant, or ”irreducible” appears in equation (1). Then, we see the irreducible error is given by the measure of the subset of $\mathcal{X}$ where $\eta(x)$ lies between t_q and $t_{q'}$. Given that we know the Bayes optimal classifier for any weighting, we observe that the irreducible error can be upper bounded by a term proportional to the the product of the measure of P_X in the region between t_q and $t_{q'}$, and the difference between the thresholds themselves. We state this formally in the following proposition.

Proposition 2. *Let* $\underline{t}_{q,q'} = \min\{t_q, t_{q'}\}$ *and* $\overline{t}_{q,q'} = \max\{t_q, t_{q'}\}$. *The irreducible error satisfies the bound*

$$\text{(IE)} \leq (q_0' + q_1') \left|t_q - t_{q'}\right| \mathbb{P}\left(\underline{t}_{q,q'} \leq \eta(X) \leq \overline{t}_{q,q'}\right)$$

A visualization is given in Figure 1. Now, we turn to analyze the estimation error. The result is in many ways similar to Proposition 1, but an additional term appears due to the decision threshold t_q for $\widehat{\eta}$ differing from that of the risk measurement $t_{q'}$.

Proposition 3. *For any density estimator* $\widehat{\eta}$, *the estimation error satisfies*

$$\begin{aligned}\text{(EE)} \leq &(q_0' + q_1')\mathbb{E}\left[\|\eta - \widehat{\eta}\|_{L_1(P_X)}\right] + \\ &(q_0' + q_1') \left|t_{q'} - t_q\right| \mathbb{E}\left[\mathbb{P}\left(\widehat{f}_q(x) \neq f_q^*(x)\right)\right]\end{aligned}$$

Corollary 1. *When* η *is* β*-Hölder, using local polynomial estimator (Yang, 1999) for* $\widehat{\eta}$ *gives*

$$\begin{aligned}\text{(EE)} \leq &(q_0' + q_1')O\left(n^{-\frac{\beta}{2\beta+d}}\right) + \\ &(q_0' + q_1') \left|t_{q'} - t_q\right| \mathbb{E}\left[\mathbb{P}\left(\widehat{f}_q(x) \neq f_q^*(x)\right)\right]\end{aligned}$$

Consequently, we can upper bound the expected excess q'-risk. The probability in the bound of the estimation error has been considered in the context of nearest neighbors (Chaudhuri & Dasgupta, 2014), but in general, additional assumptions are required to provide an explicit rate. We consider one such assumption in the appendix.

4. Robust Class Weighted Risk

Based the results in the previous section, we know that the performance degradation need not be graceful when we don't know how to choose the weights. This motivates us to study a more robust version of class weighted risk.

Definition 2. *Let* $Q \subseteq \mathbb{R}^{|\mathcal{Y}|}$ *be a compact convex set such that* $q_i \geq 0$ *for each* i *and* $\mathbb{E}[q_Y] = 1$ *for each* q *in* Q. *Then, the* Q*-weighted risk is*

$$R_Q(f) = \sup_{q \in Q} \mathbb{E}\left[q_Y R_Y(f)\right] = \sup_{q \in Q} \sum_{i \in \mathcal{Y}} q_i p_i R_i(f).$$

Additionally, we refer to the set Q *as the uncertainty set.*

In this section, we have two goals: (1) to provide excess $\mathcal{F}$-risk bounds and generalization bounds for robust weighted risk via uniform convergence and (2) to make connections to stochastic optimization via special choices of uncertainty set. We start with generalization; the proofs are given in the appendix.

Theorem 1. *Let* $\ell = \ell_{mar}$ *be the multiclass margin loss. Recall that* $N_i = \sum_{j=1}^n \mathbf{1}\{y_j = i\}$. *With probability at least* $1 - \delta$, *we have the generalization bound*

$$\begin{aligned}R_Q(f) \leq \sup_{q \in Q} \Bigg\{ &\widehat{R}_q(f) + \sum_{i=1}^k q_i p_i \\ &\times \left(4k\mathbb{E}\left[\frac{N_i}{p_i n}\widehat{\mathfrak{R}}_{N_i}(\Pi_1(\mathcal{F}))\right] + \sqrt{\frac{\log\frac{k}{\delta}}{2p_i^2 n}}\right)\Bigg\}\end{aligned}$$

for every f *in* $\mathcal{F}$ *and the excess risk bound*

$$\begin{aligned}\mathcal{E}_Q(\mathcal{F}) \leq 2 \sup_{q \in Q} &\sum_{i=1}^k q_i p_i \\ &\times \left(8k\mathbb{E}\left[\frac{N_i}{p_i n}\widehat{\mathfrak{R}}_{N_i}(\Pi_1(\mathcal{F}))\right] + \sqrt{\frac{\log\frac{k}{\delta}}{2p_i^2 n}}\right).\end{aligned}$$

A few remarks are in order. First, note that we only use the multiclass margin loss because it leads to simple multiclass bounds. In a binary classification setting, standard results would imply generalization for other Lipschitz losses. Second, in many cases, we can simplify the Rademacher complexity term. The following result applies to commonly-used function classes such as linear functions and neural networks (Bartlett et al., 2017; Golowich et al., 2018; Mohri et al., 2012).

Corollary 2. *Let* $\ell = \ell_{mar}$ *be the multiclass margin loss. Let* $\mathcal{F}$ *be a function class satisfying* $\widehat{\mathfrak{R}}_n(\Pi_1(\mathcal{F})) \leq C(\mathcal{F})n^{-1/2}$ *for some constant* $C(\mathcal{F})$ *that does not depend on* n. *Then with probability at least* $1 - \delta$, *we have the generalization bound*

$$\begin{aligned}R_Q(f) \leq \sup_{q \in Q} \Bigg\{ &\widehat{R}_q(f) + \sum_{i=1}^k q_i p_i \\ &\times \left(\frac{4kC(\mathcal{F})}{\sqrt{p_i n}} + \sqrt{\frac{\log\frac{k}{\delta}}{2p_i^2 n}}\right)\Bigg\}\end{aligned}$$

and the excess $(\mathcal{F}, q)$*-risk bound*

$$\mathcal{E}_Q(\mathcal{F}) \leq 2 \sup_{q \in Q} \sum_{i=1}^k q_i p_i \left(\frac{8kC(\mathcal{F})}{\sqrt{p_i n}} + \sqrt{\frac{\log\frac{k}{\delta}}{2p_i^2 n}}\right).$$

4.1. Connections to Stochastic Programming

In this section, we make concrete connections to stochastic programming (Shapiro et al., 2009). First, we introduce label conditional value at risk, and then we describe the generalization, label heterogeneous conditional value at risk.

4.1.1. Label CVaR

We start with the definition.

Definition 3. *Let α in $(0,1)$ be given. Define the set $Q_\alpha = \left\{q : \mathbb{E}[q_Y] = 1, q_i \in \left[0, \alpha^{-1}\right] \text{ for } i \in 1, \ldots, k\right\}$. The label conditional value at risk (LCVaR) is $\text{LCVaR}_\alpha(f) = R_{Q_\alpha}(f)$.*

Now, we describe the connection to CVaR. Letting Z be a random variable, the CVaR of Z at level α is $\text{CVaR}_\alpha(Z) = \sup_{Q \in Q^*_\alpha} \mathbb{E}_Q[Z] = \sup_{Q \in Q^*_\alpha} \mathbb{E}[(dQ/dP)Z]$, where Q^*_α is the set of all probability measures that are absolutely continuous with respect to the underlying measure P such that $dQ/dP \leq \alpha^{-1}$. If Z takes values on a finite discrete probability space with probability mass function p, then the CVaR may be written as $\text{CVaR}_\alpha(Z) = \sup_{q \in Q_\alpha} \sum_{i=1}^k q_i p_i Z$. Thus, LCVaR is a specialization of CVaR to the variables $R_Y(f)$, which take values on the finite discrete space $\mathcal{Y}$. Notably, this is in contrast to other uses of CVaR in machine learning where, as noted previously, CVaR is used with respect to samples directly, in order to provide robustness or fairness. As with CVaR, LCVaR is a straightforward way to provide robustness. Intuitively, it moves weight to the worst losses, where all weightings are bounded by the same constant α^{-1}. Now, we consider the dual form.

Proposition 4 (LCVaR dual form)**.** *LCVaR permits the dual formulation*

$$\text{LCVaR}_\alpha(f) = \inf_{\lambda \in \mathbb{R}} \left\{ \frac{1}{\alpha} \mathbb{E}[(R_Y(f) - \lambda)_+] + \lambda \right\}.$$

Moreover, if $\mathcal{F}$ is compact in the supremum norm on $\mathcal{X}$ and ℓ is continuous, then the dual form holds for all f in $\mathcal{F}$.

The proof is mostly standard and therefore deferred to the appendix. The only trick compared with CVaR is showing that we may restrict the domain of λ to a compact set; which essentially requires showing that the process $\{R_Y(f) : f \in \mathcal{F}\}$ is sufficiently well-behaved. It would also suffice to assume that ℓ is bounded, as with most theoretical results in learing theory. Note that to minimize LCVaR, we can solve this convex program in λ and f.

4.1.2. Label Heterogeneous CVaR

While the LCVaR approach of the previous section is useful for providing some robustness in a computationally tractable manner, it may not be best suited for imbalanced classification because it treats all classes identically in that each q_i must lie in the interval $[0, \alpha^{-1}]$. Since imbalanced classification is inherently a problem of heterogeneity, we may wish to allow q_i to be in some interval $[0, \alpha_i^{-1}]$ instead. We can formalize this problem as follows.

Definition 4. *Define the uncertainty set $Q_{H,\alpha} = \left\{q : \mathbb{E}[q_Y] = 1, q_i \in [0, \alpha_i^{-1}] \text{ for } i = 1, \ldots, k\right\}$. We call the resulting optimization problem label heterogeneous conditional value at risk (LHCVaR), and we write*

$$\textit{LHCVaR}_\alpha(f) = \sup_{q \in Q_{H,\alpha}} \mathbb{E}\left[q_Y R_Y(f)\right].$$

Similar to LCVaR, this has a dual form.

Proposition 5. *A dual form for LHCVaR is given by*

$$\textit{LHCVaR}_\alpha(f) = \inf_{\lambda \in \mathbb{R}} \mathbb{E}\left[\alpha_Y^{-1} \left(R_Y(f) - \lambda\right)_+\right] + \lambda.$$

Moreover, if $\mathcal{F}$ is compact in the supremum norm on $\mathcal{X}$ and ℓ is continuous, then the dual form holds for all f in $\mathcal{F}$.

Again, we note that an alternative sufficient condition for the dual to hold for all f in $\mathcal{F}$ is that ℓ be bounded. Importantly, the label heterogeneous CVaR dual form is convex in f and λ. As a result, we can still optimize efficiently, in principle.

We also note that the finite dimension k is crucial for label heterogeneous CVaR. This is due to our use of the minimax theorem, which requires compactness in various places; so in general this result cannot be extended to the infinite-dimensional case.

5. Numerical Results

Code for reproducing the results in this section can be found at `https://www.github.com/neilzxu/robust_weighted_classification`.

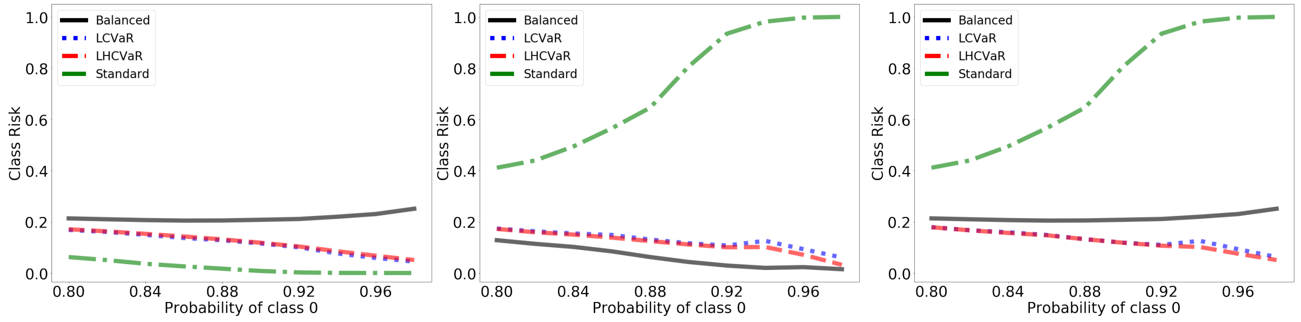
5.1. Methods

We examine the empirical performance of LCVaR and LHCVaR risks, and compare them against the standard risk and a balanced risk as baselines. Let $\widehat{p}_i$ be the empirical proportion of the ith label and $\widehat{R}_i$ be the empirical class conditional risk.

Balanced risk Here, we consider the specific weighting where each class is equally weighted:

$$\widehat{R}_{1/(k\widehat{p})}(f) = \frac{1}{k} \sum_{i=1}^{k} \widehat{R}_i(f)$$

i.e., we fix $q_i = 1/(k\widehat{p}_i)$.



(a) Class 0 risk (b) Class 1 risk (c) Worst class risk

Figure 2: Plots of class 0, class 1, and worst class risk on the test dataset under different choices of $1-p$ in the synthetic experiment. The worst test class risk is the maximum of the risks of the two classes for each choice of the probability of class 0. LCVaR and LHCVaR performs better in worst class risk than both standard and balanced risks as class imbalance increases.

LCVaR The empirical formulation optimizes the dual formulation, in which α is a hyperparameter:

$$\widehat{\mathrm{LCVaR}}_\alpha(f) = \min_{\lambda\in\mathbb{R}} \left\{ \frac{1}{\alpha}\sum_{i=1}^{k} \widehat{p}_i(\widehat{R}_i(f)-\lambda)_+ + \lambda \right\}. \tag{2}$$

LHCVaR We similarly optimize a dual form in the empirical LHCVaR risk. To reduce the number of hyperparameters to only $c \in (0,1]$ and $\kappa \in (0,\infty)$, we calculate α_i as follows:

$$\alpha_i^{(\kappa,c)} = c\left(\frac{\widehat{p}_i^{\,1/\kappa}}{\sum_{j=1}^{k} \widehat{p}_j^{\,1/\kappa}}\right). \tag{3}$$

κ behaves as a temperature parameter (similar to Jang et al. 2016; Wang et al. 2020) and causes α to become a smoother distribution of weights when $\kappa > 1$ and converge to uniform weights as $\kappa \to \infty$. Conversely, when $\kappa < 1$, the alpha distribution becomes sharper and heavily weights the classes with lowest $\widehat{p}_i$ as $\kappa \to 0$. We simply choose a κ of 1 unless otherwise stated. c consequently characterizes the total magnitude of the weights. Ultimately, we formulate the empirical risk as:

$$\widehat{\mathrm{LHCVaR}}_{\kappa,c}(f) = \inf_{\lambda\in\mathbb{R}} \left\{ \sum_{i=1}^{k} \frac{\widehat{p}_i}{\alpha_i^{(\kappa,c)}}(\widehat{R}_i(f)-\lambda)_+ + \lambda \right\}$$

We train a logistic regression model with gradient descent on a cross entropy loss, which acts as a convex surrogate loss for zero-one risk.

5.2. Datasets

We evaluate our methods on both synthetic and real datasets.

Synthetic Datasets The data in our synthetic experiment is constructed for $\mathcal{X} = [0,1]$ and $\mathcal{Y} = \{0,1\}$. For a given $p = P(Y=0)$, we generated a dataset by uniformly randomly sampling an X in $[0,1]$ and sampling a Y with the following distribution:

$$\begin{aligned} P(Y=1 \mid X=x) &= x^{\frac{p}{1-p}} \\ P(Y=0 \mid X=x) &= 1 - x^{\frac{p}{1-p}}. \end{aligned}$$

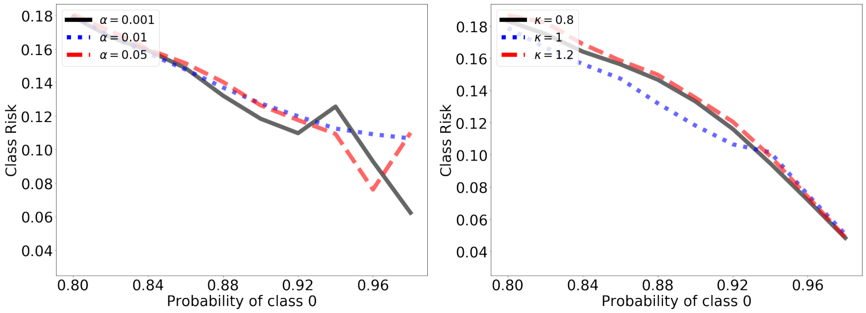
In these synthetic datasets, we note that the Bayes optimal classifier and class risks are:

$$\begin{aligned} f^*(x) &= \mathbf{1}\left\{ x > \left(\frac{1}{2}\right)^{\frac{1-p}{p}} \right\} \\ R_0(f^*) &= 1-(1+p)\left(\frac{1}{2}\right)^{\frac{1}{p}} \\ R_1(f^*) &= \left(\frac{1}{2}\right)^{\frac{1}{p}}. \end{aligned}$$

When p is high, $R_0(f^*) < R_1(f^*)$, which leads to a classifier that has vastly worse performance on class 1 compared to class 0. This discrepancy in class risk is a common issue in classification problems where there is a significant class imbalance.

We randomly generated 10,000 data points for both train and test sets. We generated datasets for each value of p from 0.80 to 0.98, inclusive, in steps of 0.02.

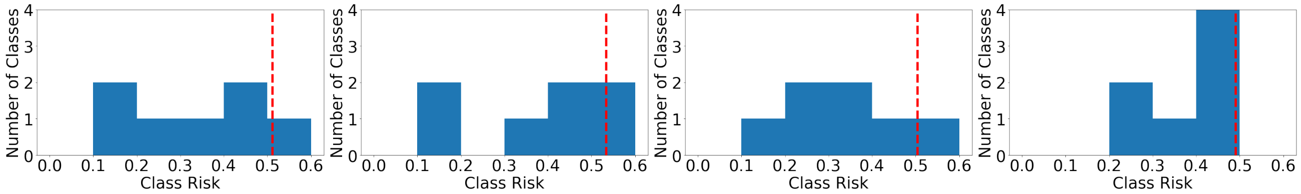
Real World Datasets We also experiment on the Covertype dataset taken from the UCI dataset repository (Dua & Graff, 2017). This dataset is 53-dimensional with 7 classes and has 2%-98% (11340-565892 examples) train-test split.



(a) Varying α for LCVaR

(b) Varying κ for LHCVaR with $\alpha = 0.01$

Figure 3: Worst class risk of different α values for LCVaR and κ values for LHCVaR in the synthetic setting. Across different levels of class imbalance, α and κ do not have a significant impact on worst class risk of LCVaR and LHCVaR.



(a) Standard

(b) Balanced

(c) LCVaR

(d) LHCVaR

Figure 4: Histogram of class risks for each method on the Covertype dataset. The red line marks the largest risk for each method. The distribution of class risks for standard and balanced methods are more spread out, while the class risks for LCVaR and LHCVaR are more concentrated near the max class risk. The max class risks are slightly lower for LCVaR and LHCVaR compared to the other two methods.

5.3. Results

Synthetic In Fig. 2, we can observe that the the worst case class risk of LCVaR and LHCVaR across multiple values of p is better than both the standard and balanced classifier. The classwise risks of LCVaR and LHCVaR are relatively close across different values of p, while there is a large discrepancy between classwise risks of the classifier trained under the standard or balanced risks. Note that the more significant the imbalance, i.e., the smaller the p, the better LCVaR and LHCVaR perform compared to balanced risk on class 0, while paying a progressively smaller price on the class 1 risk. The same is also true between both LCVaR and LHCVaR and the standard risk, although with the classes swapped. We note that while the worst class risk of LCVaR and LHCVaR seem to decrease with greater imbalance, this may not be a general property of these methods. Rather, this is more likely an artifact of the synthetic setup having more probability mass further from the decision boundary as the imbalance increases. The main observation is simply that LCVaR and LHCVaR have lower worst class risk in comparison to the baseline methods. Thus, this empirically demonstrates that both LCVaR and LHCVaR can significantly improve the highest class risks while losing little in performance on classes with lower risks.

In addition to comparing against baselines, we also examine the effect of different choices of α and κ on LCVaR and LHCVaR, respectively. The results of this comparison are in Fig. 3. In both methods, varying the hyperparameters does not have a dramatic impact on the behavior of the worst class risk for both these methods across different values of class imbalance.

Table 1: Standard risk and risk of the worst class for each method on the Covertype dataset. LCVaR and LHCVaR improve on the worst class risk.

Method	*Standard Risk*	*Worst Class Risk*
LHCVaR	0.3979	**0.4907**
LCVaR	0.3384	0.5037
Standard	**0.3275**	0.5111
Balanced	0.3765	0.5333

Real In Table 1, we observe that LCVaR and LHCVaR have better worst class risks than the standard and class weighted baselines. However, improving worst class risk comes at a cost to to the standard risk in the case of both LCVaR and LHCVaR. This tradeoff is reflected in the histograms of class risk shown in Fig. 4, where the class risks under the standard and balanced classifiers are more spread out and have classes with much lower risks. On the other

Table 2: Performance of LCVaR across different α values, and LHCVaR across different κ values. The performance each method is relatively agnostic to choices of α and κ, although the smallest choices of α and κ for each method have the largest changes in worst class risk, respectively.

Method	α	κ	*Standard Risk*	*Worst Class Risk*
LCVaR	0.01	N/A	0.4266	0.5474
	0.05	N/A	0.3993	0.4932
	0.1	N/A	0.4060	0.5037
LHCVaR	0.05	0.8	0.4308	0.5408
	0.05	1	0.3979	0.4907
	0.05	1.2	0.4171	0.5050

hand, LCVaR and LHCVaR have class risk distributions that are more concentrated towards the worst class risk value. Consequently, LCVaR and LHCVaR achieve a lower worst class risk, which is consistent with our theory.

We also compare the effect of choosing different α and κ on LCVaR and LHCVaR, respectively, in Table 2. We see that the worst class risk still performs well under different choices of α and κ, although there is some degradation when the α is smaller than optimal choice, in the case of LCVaR, and when κ is smaller and produces a sharper distribution, in the case of LHCVaR.

6. Discussion

In this work, we have studied the effect of optimizing classifiers with respect to different weightings and developed robust risk measures that minimizes worst case weighted risk across a set of weightings. We subsequently show that optimizing with respect to LCVaR and LHCVaR empirically improves the worst class risk, at a reasonable cost to accuracy. One future direction for research is to understand the Bayes optimal classifier under LCVaR and LHCVaR. Another more applied direction could be to consider domain shift. If we formalize each prior over the classes as a weighting, optimizing LCVaR or LHCVaR may improve performance when the test class priors are different from the training class priors.

7. Acknowledgements

We acknowledge the support of Rakuten Inc., and Microsoft Research. The authors would also like to thank Biswajit Paria for his contributions to the numerical simulations, and to him and Pradipto Das for their helpful comments and discussions.

References

Bartlett, P. L., Foster, D. J., and Telgarsky, M. J. Spectrally-normalized margin bounds for neural networks. In *Advances in Neural Information Processing Systems*, pp. 6240–6249, 2017.

Ben-Tal, A. and Nemirovski, A. Robust solutions of uncertain linear programs. *Operations research letters*, 25(1): 1–13, 1999.

Ben-Tal, A. and Nemirovski, A. Robust solutions of linear programming problems contaminated with uncertain data. *Mathematical programming*, 88(3):411–424, 2003.

Ben-Tal, A., Goryashko, A., Guslitzer, E., and Nemirovski, A. Adjustable robust solutions of uncertain linear programs. *Mathematical Programming*, 99(2):351–376, 2004.

Ben-Tal, A., El Ghaoui, L., and Nemirovski, A. *Robust Optimization*. Princeton University Press, 2009.

Ben-Tal, A., Den Hertog, D., De Waegenaere, A., Melenberg, B., and Rennen, G. Robust solutions of optimization problems affected by uncertain probabilities. *Management Science*, 59(2):341–357, 2013.

Bertsimas, D., Gupta, V., and Kallus, N. Robust sample average approximation. *Mathematical Programming*, pp. 1–66, 2014.

Cao, K., Wei, C., Gaidon, A., Arechiga, N., and Ma, T. Learning imbalanced datasets with label-distribution-aware margin loss. *arXiv preprint arXiv:1906.07413*, 2019.

Chaudhuri, K. and Dasgupta, S. Rates of convergence for nearest neighbor classification. In *Advances in Neural Information Processing Systems*, pp. 3437–3445, 2014.

Chawla, N. V., Bowyer, K. W., Hall, L. O., and Kegelmeyer, W. P. SMOTE: synthetic minority over-sampling technique. *Journal of Artificial Intelligence Research*, 16: 321–357, 2002.

Devroye, L., Györfi, L., and Lugosi, G. *A probabilistic theory of pattern recognition*. Springer Science & Business Media, 1996.

Domingos, P. Metacost: A general method for making classifiers cost-sensitive. In *KDD*, volume 99, pp. 155–164, 1999.

Dua, D. and Graff, C. Uci machine learning repository, 2017.

Duchi, J. and Namkoong, H. Learning models with uniform performance via distributionally robust optimization. *arXiv preprint arXiv:1810.08750*, 2018.

Duchi, J. C., Hashimoto, T., and Namkoong, H. Distributionally robust losses against mixture covariate shifts. *Arxiv*, 2018.

Feldman, V. Does learning require memorization? a short tale about a long tail. *arXiv preprint arXiv:1906.05271*, 2019.

Fernández, A., García, S., Galar, M., Prati, R. C., Krawczyk, B., and Herrera, F. *Learning from imbalanced data sets*. Springer, 2018.

Goh, J. and Sim, M. Distributionally robust optimization and its tractable approximations. *Operations research*, 58 (4-part-1):902–917, 2010.

Golowich, N., Rakhlin, A., and Shamir, O. Size-independent sample complexity of neural networks. In *Conference On Learning Theory*, pp. 297–299, 2018.

He, H. and Garcia, E. A. Learning from imbalanced data. *IEEE Transactions on knowledge and data engineering*, 21(9):1263–1284, 2009.

Jang, E., Gu, S., and Poole, B. Categorical reparameterization with gumbel-softmax. *arXiv preprint arXiv:1611.01144*, 2016.

Khan, J., Wei, J. S., Ringner, M., Saal, L. H., Ladanyi, M., Westermann, F., Berthold, F., Schwab, M., Antonescu, C. R., Peterson, C., et al. Classification and diagnostic prediction of cancers using gene expression profiling and artificial neural networks. *Nature medicine*, 7(6):673, 2001.

Lee, Y., Wahba, G., and Ackerman, S. A. Cloud classification of satellite radiance data by multicategory support vector machines. *Journal of Atmospheric and Oceanic Technology*, 21(2):159–169, 2004.

Lin, Y., Lee, Y., and Wahba, G. Support vector machines for classification in nonstandard situations. *Machine learning*, 46(1-3):191–202, 2002.

Lin, Y.-C., Das, P., and Datta, A. Overview of the SIGIR 2018 eCom Rakuten Data Challenge. In *eCOM@ SIGIR*, 2018.

Mariani, G., Scheidegger, F., Istrate, R., Bekas, C., and Malossi, C. Bagan: Data augmentation with balancing gan. *arXiv preprint arXiv:1803.09655*, 2018.

Mohri, M., Rostamizadeh, A., and Talwalkar, A. *Foundations of Machine Learning*. MIT Press, 2012.

Namkoong, H. and Duchi, J. C. Variance-based regularization with convex objectives. In *Advances in Neural Information Processing Systems*, pp. 2971–2980, 2017.

Rigollet, P. and Tong, X. Neyman-pearson classification, convexity and stochastic constraints. *Journal of Machine Learning Research*, 12(Oct):2831–2855, 2011.

Rockafellar, R. T., Uryasev, S., et al. Optimization of conditional value-at-risk. *Journal of risk*, 2:21–42, 2000.

Salakhutdinov, R., Torralba, A., and Tenenbaum, J. Learning to share visual appearance for multiclass object detection. In *CVPR 2011*, pp. 1481–1488. IEEE, 2011.

Scott, C. Calibrated asymmetric surrogate losses. *Electronic Journal of Statistics*, 6:958–992, 2012.

Shapiro, A., Dentcheva, D., and Ruszczyński, A. *Lectures on stochastic programming: modeling and theory*. SIAM, 2009.

Tong, X. A plug-in approach to neyman-pearson classification. *The Journal of Machine Learning Research*, 14(1): 3011–3040, 2013.

Tong, X., Feng, Y., and Zhao, A. A survey on neyman-pearson classification and suggestions for future research. *Wiley Interdisciplinary Reviews: Computational Statistics*, 8(2):64–81, 2016.

Van Rijsbergen, C. J. Foundation of evaluation. *Journal of Documentation*, 30(4):365–373, 1974.

Van Rijsbergen, C. J. *Information Retrieval*. Butterworth-Heinemann, London, 2nd edition, 1979.

Wang, J., Shen, X., and Liu, Y. Probability estimation for large-margin classifiers. *Biometrika*, 95(1):149–167, 2008.

Wang, X., Helen Zhang, H., and Wu, Y. Multiclass probability estimation with support vector machines. *Journal of Computational and Graphical Statistics*, pp. 1–18, 2019.

Wang, X., Tsvetkov, Y., and Neubig, G. Balancing training for multilingual neural machine translation. *arXiv preprint arXiv:2004.06748*, 2020.

Wu, Y., Zhang, H. H., and Liu, Y. Robust model-free multiclass probability estimation. *Journal of the American Statistical Association*, 105(489):424–436, 2010.

Yang, Y. Minimax nonparametric classification. i. rates of convergence. *IEEE Transactions on Information Theory*, 45(7):2271–2284, 1999.

Zhou, Z.-H. and Liu, X.-Y. Training cost-sensitive neural networks with methods addressing the class imbalance problem. *IEEE Transactions on Knowledge and Data Engineering*, 18(1):63–77, 2006.

Zhu, X., Anguelov, D., and Ramanan, D. Capturing long-tail distributions of object subcategories. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 915–922, 2014.

Zipf, G. K. *The Psycho-Biology of Language: an Introduction to Dynamic Philology*. George Routledge & Sons, Ltd., 1936.