Efficiently Satisfying Subgroup Fairness in Generalized Classification Settings

Matt King, Fahim Tajwar

1. Introduction

Fairness in machine learning is increasingly topical as machine learning algorithms are leveraged to predict convict recidivism, future ability to pay loans, and many other predictions which correct or not have the ability to influence individuals' lives for decades afterwards. Fairness in classification problems has been defined in several different statistical frameworks, but often, an algorithm is considered fair if no protected group (e.g. a certain race, gender, sex, etc.) endures a significantly higher false positive rate than other groups. However, this method is flawed, as one can form intersection of different groups (called subgroups) that suffer discrimination, but the overall group might not. The following toy example demonstrates this:

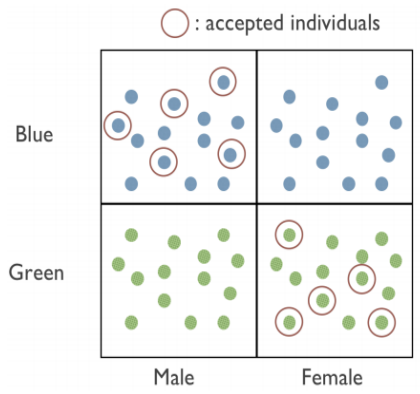

Figure 1.1. Toy example showing fairness garrymandering (Kearns et al., 2018a)

Here, the false positive rate is same over the two races, and two genders separately. However, the 'blue female' and 'green male' persons are surely discriminated against.

We intend to extend the recent work of Kearns et al. (Kearns et al., 2018b), where the authors define fairness criterion for every combinatorial subgroup produced from protected groups, and describes an algorithm that solves the classification problem with a desired bound on fairness without reasonably compromising classification accuracy. Specifically, while the mentioned paper deals with binary classification, we define fairness in the context of multi-class classification, and prove that a similar algorithm exist in this case. While the results of Kearns et al. are foundational to this paper, our extension is novel and highly non-trivial, in part because each definition and theorem must deal with the fact that in k-class classification, no class can be treated as "special" i.e. a false positive rate is not defined, and criteria must be phrased in terms of k-classes whereas in the binary setting they can be phrased in terms of only one.

2. Related Works

With the increasing attention to underlying bias in most widely used datasets, there is also a recent surge of studies on fairness in machine learning. One such paper is authored by Reuben Binns (Binns, 2018), which discusses why one should care about fairness, and not only high accuracy in classification. We find different algorithms in different settings, that assumes a certain definition of fairness first, and then solves the classification problem maintaining fairness as well as accuracy and runtime (though a little worse than algorithms without fairness constraint). Joseph et al. (Joseph et al., 2016) gives such an algorithm for the Rawlsian measure of fairness for online decision making. The theoretical analysis (Kearns et al., 2018b) and empirical validation (Kearns et al., 2018a) papers of Kearns et al. are central to this paper, as much of the definitions of fairness, theorems and algorithms are inspired from them. The works by Holstein et al. (Holstein et al., 2019) gives an interesting perspective on machine learning fairness in industry today. Kamishima et al. (Kamishima et al., 2011) provides additional algorithms and perspectives to fairness, and discusses ways we can modify existing algorithms like logistic regression to be more conscious about fairness. Finally, Corbett-Davies and Goel (Corbett-Davies & Goel, 2018) discusses different measures of fairness and their pros and cons, which inspired us in our own definition of fairness in this paper.

3. Background and preliminary definitions for binary classification

Kearns et al. (Kearns et al., 2018b) formally sets up the problem for binary classification as follows:

1. Each individual is $(X, y) = ((x, x'), y)$ where x is a vector of protected attributes, x' is a vector of unprotected attributes, $y \in 0, 1$ is the associated label.

2. $D(x) :=$ decision made on individual (X, y)
3. For each protected group G, let $g(x)$ be an indicator function for x being in group G
4. Let $S = \{z(i) = ((x(i), x'(i)), y(i))\}_{i=1}^{n}$ be a set of n training examples.
5. Let P be the empirical distribution over S.

With this setup, we use the statistical parity subgroup fairness, as our metric for checking fairness of the learned hypothesis class. This is formally defined as follows:

Definition 3.1. Statistical parity (SP) subgroup fairness: Fix any classifier D, distribution $\mathcal{P}$, group indicator g, and threshold $\gamma \in [0, 1]$. Define

$$\alpha_{\text{SP}}(g, \mathcal{P}) = Pr_{\mathcal{P}}[g(x) = 1]$$
$$\beta_{\text{SP}}(g, D, \mathcal{P}) = |\text{SP}(D) - \text{SP}(D, g)|$$

where $\text{SP}(D) = Pr_{D,\mathcal{P}}[D(X) = 1]$, and $\text{SP}(D, g) = Pr_{D,\mathcal{P}}[D(X) = 1 | g(x) = 1]$ give the probability that $\mathcal{D}$ labels a generic example (and respectively, an example of subgroup g) as positive.

We say D satisfied $\gamma-$statistical parity fairness with respect to $\mathcal{P}$ and g, if

$$\alpha_{SP}(g, \mathcal{P})\beta_{SP}(g, D, \mathcal{P}) \leq \gamma.$$

And D is γ-SP fair with respect to $\mathcal{P}$ and a collection of subgroup indicators $\mathcal{G}$ if D satisfies γ-SP fairness for every $g \in \mathcal{G}$.

Here α_{SP} allows us to ignore small enough groups and β_{SP} lets the positive classification rate for a group be off from the overall rate by a small amount. The smaller γ is, the fairer the classifier D is going to be.

4. Multi-class Classification

Inspired from the definitions in section 3, we proceed towards generalizing the concept to multi-class classification. Assume the number of classes is k, where $k \in \mathbb{N}$ and $k \geq 2$. We use the same setup from section 2, except for any individual datapoint (X, y), we use the convention that $y \in [\![0, 1]\!]$. We define fairness in the next subsection.

4.1. Defining Fairness

Definition 4.1. (Multi-class Statistical Parity Subgroup Fairness) Fix any classifier D, distribution $\mathcal{P}$, subgroup indicator g, and $\gamma \in [0, 1]$, and any $j \in [\![1, k]\!]$. Define

$$\alpha_{SP}(g, \mathcal{P}) := Pr_{\mathcal{P}}[g(x) = 1]$$
$$\beta_{SP}(g, \mathcal{P}, D, j) := |\text{SP}(D, j) - \text{SP}(D, j, g)|$$

with $\text{SP}(D, j) = Pr_{\mathcal{P},D}[D(X) = j]$ and $\text{SP}(D, j, g) = Pr_{\mathcal{P},D}[D(X) = j | g(x) = 1]$ respectively representing the probability of labeling any example and of labeling an example of group g with label j.

D is said to meet γ-statistical parity (SP) fairness with respect to $\mathcal{P}$ and g if for all $j \in [\![1, k]\!]$, we have,

$$\alpha_{SP}(g, \mathcal{P})\beta_{SP}(g, \mathcal{P}, D, j) \leq \gamma.$$

Furthermore, D meets γ-SP fairness with respect to $\mathcal{P}$ and $\mathcal{G}$, a collection of subgroup indicators if it is γ-SP fair for all $g \in \mathcal{G}$.

We state one more result (proven in appendix) that prove that setting $k = 2$ indeed reduces the problem to the original binary classification problem, which in turn proves that our problem is indeed a generalization of Kearns et al.'s binary classification problem. (Kearns et al., 2018b).

Lemma 4.1. *For $k = 2$, multi-class statistical parity fairness (definition 4.1) holds if and only if binary statistical parity fairness (definition 3.1) holds.*

4.2. Establishing preliminary results

It will be convenient to think about the empirical error (the error on S) as essentially equivalent to the true error, the probability that given a new example, a hypothesis classifies it incorrectly. We state a well-known result that in the case of binary classification taking the dataset large enough brings empirical error arbitrarily close to true error and use this to establish an analogous result for the multi-class setting.

The following three preliminary results follow (the first of which non-trivially) from Kearns et al.'s (Kearns et al., 2018b) results in the two-class case. These authors show that for any $\epsilon > 0$ with a large enough size n of a dataset, the true error is within ϵ of the empirical error, enabling us to hereafter concentrate solely on minimizing the empirical error, at least in the two-class regime. For classification into classes $[\![1, k]\!]$, the above definitions of true and empirical error are still well-defined and contextually appropriate. Furthermore, we establish the following results, the proofs of which are deferred to the appendix.

Let $\mathcal{H}$ be any fixed hypothesis class, $\mathcal{P}$ any distribution, and S contain n samples drawn i.i.d. according to $\mathcal{P}$. Let $\delta \in (0, 1)$. For $h \in \mathcal{H}$, let $\text{err}(h, \mathcal{P}) = p_{(X,y)\sim\mathcal{P}}(h(X) \neq y)$ denote the true error of h, and $\text{err}(h, \text{S}) = \frac{1}{n}\sum_{i=1}^{n} \mathbb{1}[h(X_i) \neq y_i]$ denote the empirical error of h on S.

Theorem 4.1. *With probability* $1-\delta$*, for all* $h \in \mathcal{H}$*,*

$$|err(h,\mathcal{P}) - err(h,S)| \leq O\left(k\frac{VCDIM(\mathcal{H})\log(n)+\log(k/\delta)}{n}\right).$$

A similar result bounds the difference between true and empirical SP fairness for large enough datasets.

Theorem 4.2. *For any fixed set of group indicators* $\mathcal{G}$*, with probability* $1-\delta$*, for any* $h \in \mathcal{H}$ *and any* $g \in \mathcal{G}$*, for any* $j \in [\![1,k]\!]$ *, the following holds:*

$$|\alpha_{SP}(g,\mathcal{P})\beta_{SP}(g,\mathcal{P},D,j) - \alpha_{SP}(g,\mathcal{P})\beta_{SP}(g,\mathcal{P},D,j)| \leq$$
$$\tilde{O}\left(\sqrt{\frac{VCDIM(\mathcal{H}) + VCDIM(\mathcal{G})\log(m) + \log(k/\delta)}{m}}\right).$$

Finally, it follows directly from results about the hardness of weak aganostic learning and its equivalence to satisfying all subgroup fairness constraints (Feldman et al., 2012) (Kearns et al., 2018b) that an exactly optimal γ-SP solution to the fair multi-class classification problem cannot be found in polynomial time.

Theorem 4.3. *With reasonable assumptions on the group of indicators* $\mathcal{G}$*, and* $\gamma \in [0,1)$*, there exist distributions* $\mathcal{P}$ *for which no polynomial time algorithm can arrive at the optimal k-class classifier which is* γ*-SP fair.*

4.3. Approximate Polynomial-Time Solution

We begin by stating the central result of Kearns et al., which finds an efficient polynomial time approximate solution to γ-SP fair binary classification for which the amount by which the solution violates the fairness criteria (ν) and the likelihood of further violating fairness constraints (δ) can be taken arbitrarily small.

Theorem 4.4. *Fix any* $\nu, \delta \in (0,1)$ *as accuracy and fairness parameters respectively. Then for binary classification of n datapoints, with access to classification oracles CSC*($\mathcal{H}$) *and CSC*($\mathcal{G}$)*, there exists an algorithm which runs in polynomial time in* ν *and* δ *which with probability at least* $1-\delta$ *finds a randomized classifier* $\hat{D}$ *such that* $err(\hat{D},\mathcal{P}) \leq OPT + \nu$*, and for any* $g \in \mathcal{G}$*, the fairness constraint violation is bounded by*

$$\alpha_{SP}(g,\mathcal{P})\beta_{SP}(g,\hat{D},\mathcal{P}) \leq \gamma + O(\nu).$$

Kearns et al. demonstrate this by reformulating the task as a linear programming problem, proving this is equivalent to a zero-sum game, and provide an algorithm for playing this game. Our culminating result is an analogous gaurantee for multi-class classification, the proof of which similarly uses these two reformulations and cites numerous intermediate results of Kearns et al. to more expediently prove similar multi-class results. We find the following for classification into k classes.

Theorem 4.5. *For* ν*,* δ*, and n as defined, and access to multi-class classification oracles CSC*($\mathcal{H}$) *and CSC*($\mathcal{G}$)*, there exists an algorithm which runs in polynomial time in* ν*,* δ*, and k which with probability at least* $1-\delta$ *finds a randomized classifier* $\hat{D}$ *such that* $err(\hat{D},\mathcal{P}) \leq OPT + \nu$*, and for any* $g \in \mathcal{G}$*, for any* $j \in [\![1,k]\!]$*, the fairness constraint violation is bounded by*

$$\alpha_{SP}(g,\mathcal{P})\beta_{SP}(g,\hat{D},\mathcal{P},j) \leq \gamma + O(\nu).$$

4.3.1. Linear Programming Reformulation

To rewrite the problem as a linear programming objective, we first need to bound the sizes of $|\mathcal{H}(\mathcal{S})|$ and $|\mathcal{G}(\mathcal{S})|$, because these cardinalities will give the number of variables and of constraints respectively in the linear programming problem.

Lemma 4.2. *Let S, and* $n = |S|$*, and k be as defined, and let* $d_1 = VCDIM(\mathcal{H})$ *and* $d_2 = VCDIM(\mathcal{G})$ *be the VC-dimensions of* $\mathcal{H}$ *and* $\mathcal{G}$*. Then*

$$|\mathcal{H}(\mathcal{S})| \leq O(n^{d_1}\log(k)^{d_1}) \text{ and } |\mathcal{G}(\mathcal{S})| \leq O(n^{d_2}).$$

Proof. Sauer's lemma (which is well-known in extremal set theory, see for example the work by Kearns and Vazirani (Kearns & Vazirani, 1994) for a detailed treatment), gives that at most $O(n^{d_1})$ binary tuples $(h(X_1),...,h(X_n))$ with $h(X_i) \in \{0,1\}$ for all i can be generated by all the $h \in \mathcal{H}$, so $|\mathcal{H}(S)| \leq O(n^{d_1})$. Represent k-class classification as at most $\lceil \log_2(k) \rceil$ binary decisions, so any algorithm outputting one of k classes for n datapoints has at most as much freedom as one making binary decisions on $n\lceil \log_2(k) \rceil$. Thus, from Sauer's lemma, $|\mathcal{H}(S)| \leq O((n\lceil \log_2(k) \rceil)^{d_1}) = O(n^{d_1}\log(k)^{d_1})$. Similarly, Sauer's lemma directly gives that $|\mathcal{G}(S)| \leq O(n^{d_2})$, completing the proof.
□

With an eye towards formulating fairness constraints in a Lagrangian, the following notation simplifies calculations.

Definition 4.2. For any $g \in \mathcal{G}$, $h \in \mathcal{H}$, and $j \in [\![1,k]\!]$, let

$$\Phi_+(g,h,j) := \alpha_{SP}(g,\mathcal{P})(\text{SP}(g,h) - \text{SP}(g,h,j)) - \gamma$$
$$\Phi_-(g,h,j) := \alpha_{SP}(g,\mathcal{P})(\text{SP}(g,h,j) - \text{SP}(g,h)) - \gamma.$$

Furthermore, for any distribution $\mathcal{D}$ over $\mathcal{H}$ and sign $\bullet \in \{+,-\}$, define

$$\Phi_\bullet(g,\mathcal{D},j) = \mathop{\mathbb{E}}_{h\sim\mathcal{D}}[\Phi_\bullet(g,h,j)].$$

Note that this ensures that for every $g \in \mathcal{G}$, $h \in \mathcal{H}$, $j \in [\![1,k]\!]$, and $\nu > 0$, $\max(\Phi_+(g,h,j),\Phi_-(g,h,j))$ if and only if $\alpha_{SP}(g,\mathcal{P})\beta_{SP}(g,\mathcal{P},\mathcal{D},j) \leq \gamma + \nu$ (i.e. if $\mathcal{D}$ is $\gamma+\nu$-SP fair to class g.)

Finally, for each $g \in \mathcal{G}$, and $j \in [\![1,k]\!]$, introduce the Lagrange multipliers $\lambda^+_{g,j}$ and $\lambda^-_{g,j}$ to write the partial Lagrangian of the linear program derived:

$$\mathcal{L}(\mathcal{D},\lambda) = \mathop{\mathbb{E}}_{h\sim\mathcal{D}}[err(\mathcal{H},P)] + \tag{4.1}$$

$$\sum_{g\in\mathcal{G}(S)}\sum_{j=1}^{k}(\lambda^+_{g,j}\Phi_+(g,h,j) + \Phi_-(g,h,j)\lambda^-_{g,j}). \tag{4.2}$$

Now, (as Kearns et al. also decide) for guaranteed convergence, and for equating solving the partial Lagrangain with approximately (up to $\nu+\gamma$) satisfying fairness constraints, let $C > 0$ bound the l_1 norm of λ by requiring $\lambda \in \Lambda := \{\lambda \in \mathbb{R}^{2k|\mathcal{G}(s)|} : ||\lambda||_1 \leq C\}$ for λ the tuple of all $\lambda_{g,j}$. Then, the set of all $\mathcal{D}$ is convex and Λ is not only convex but now compact, so by Sion's minmax theorem, (Sion, 1958) Now, the multi-class Lagrangian optimization satisfies the conditions of Kearns et al.'s result below, which follows from previous definitions.

From here, finding an approximate solution to the partial Lagrangian approximately solves the fair multi-class classification problem with bounded error and fairness violations.

4.3.2. Formulation as a Two-Player Zero-Sum Game

The application of Sion's lemma above makes solving the Lagrangian ripe for framing as a two-player zero-sum game, the starting framework of which is restated here, having been demonstrated by Kearns et al. (2018). Let the Learner follow a pure strategy taking actions corresponding to $h \in \mathcal{H}$. For the opposing Auditor, define the set of vertices of Λ,

$$\Lambda_{\text{pure}} := \{\lambda \in \Lambda : \lambda^{\bullet}_{g,j} = C \text{ for some } g,j,\bullet\} \cup \{0\}.$$

; the Learner follows a pure strategy in Λ_{pure}. Naturally, compute the payoff of pairs of interactions $(h,\lambda) \in \mathcal{H} \times \Lambda_{\text{pure}}$ by the partial Lagrangian defined earlier:

$$U(h,\lambda) = \mathop{\mathbb{E}}_{h\sim\mathcal{D}}[err(\mathcal{H},P)] + \sum_{g\in\mathcal{G}(S)}\sum_{j=1}^{k}(\lambda^+_{g,j}\Phi_+(g,h,j) + \Phi_-(g,h,j)\lambda^-_{g,j}).$$

Our formulation culminates in concluding the original result that this intuitive translation of the Lagrangian into the two-player game indeed gives a correct approximate solution to the partial Lagrangian from .

Theorem 4.6. *For a given $\mathcal{D}$ and $h \sim \mathcal{D}$, finding $\arg\max_{g,j,\bullet}\Phi_\bullet(g,h,j)$ (with the notation of Definition 4.2) is equivalent to finding $\arg\min_{g,j}\alpha_{SP}(g,\mathcal{P})\beta_{SP}(g,h,\mathcal{P},j)$ as in Definition 4.2.*

Proof. Compute

$$\arg\max_{g,j,\bullet}\Phi_\bullet(g,D,j)$$

$$= \arg\max_{g,j,\bullet}\mathop{\mathbb{E}}_{h\sim\mathcal{D}}[\Phi_\bullet(g,h,j)]$$

$$= \arg\max_{g,j,\bullet}\mathop{\mathbb{E}}_{h\sim\mathcal{D}}[\alpha_{SP}(g,\mathcal{P})(|\text{SP}(g,h) - \text{SP}(g,h,j))| - \gamma]$$

$$= \arg\max_{g,j,\bullet}\alpha_{SP}(g,\mathcal{P})(|\text{SP}(g,h) - \text{SP}(g,h,j|))$$

$$= \arg\max_{g,j,\bullet}\alpha_{SP}(g,\mathcal{P})(\beta_{\text{SP}}(g,h)).$$

which concludes our proof. □

4.3.3. Solving the game with no regret dynamics

This part of our multi-class problem directly follows from section 4.3 of Kearns et al.'s (Kearns et al., 2018b) section 4.3, since this part of the algorithm in the paper does not depend on the number of classes. This also essentially proves that similar results from this paper holds in the case of multi-class classification.

5. Conclusion

In summary, we proved that most of the steps in section 4 of Kearns et al. (Kearns et al., 2018b) holds. From this, one can define cost functions similar to their paper, and construct a similar algorithm that solves the k-class classification problem with γ-SP multi-class fairness (definition 4.1), and completes the generalization of Kearns et al.

6. Future Work

Our work sheds light into the theoretical generalization of the Kearns et al.'s result. We want to, in future, implement similar generalization of the algorithm in the Kearns et al.'s empirical validation work (Kearns et al., 2018a). We also plan to extend this to continuous outputs. Fairness concerns apply equally continuous predictions, but are much harder to evaluate. We propose the minimization of the function of the following form:

$$C(h) = \sum_{i=1}^{n}\int_{\mathbb{R}} h_i(v)c_i(v)dv$$

where h_i gives predicted probability density over output y_i, and c_i is an integrable function mapping predictions to costs. C denotes the cost of h, the set of h_i, which denotes the single element of a hypothesis class.

References

Binns, R. Fairness in machine learning: Lessons from political philosophy. *Journal of Machine Learning Research*, 81:1–11, 2018.

Corbett-Davies, S. and Goel, S. The measure and mismeasure of fairness: A critical review of fair machine learning. *Arxiv pre-print*, 2018.

Feldman, V., Guruswami, V., Raghavendra, P., and Wu, Y. Agnostic learning of monomials by halfspaces is hard. *SIAM J. Comput.*, 41(6):1558–1590, 2012.

Holstein, K., Vaughan, J. W., III, H. D., Dudík, M., and Wallach, H. Improving fairness in machine learning systems: What do industry practitioners need? *CHI*, 2019.

Joseph, M., Kearns, M., Morgenstern, J., Neel, S., and Roth, A. Rawlsian fairness for machine learning. *Arxiv pre-print*, 2016.

Kamishima, T., Akaho, S., and Sakuma, J. Fairness-aware learning through regularization approach. *IEEE 11th International Conference on Data Mining Workshops*, 2011. URL http://www.kamishima.net/archive/2011-ws-icdm_padm.pdf.

Kearns, M., Neel, S., Roth, A., and Wu, Z. S. An empirical study of rich subgroup fairness for machine learning. *Arxiv pre-print*, 2018a. URL https://arxiv.org/pdf/1808.08166.pdf.

Kearns, M., Neel, S., Roth, A., and Wu, Z. S. Preventing fairness gerrymandering: Auditing and learning for subgroup fairness. *Arxiv pre-print*, 2018b. URL https://arxiv.org/pdf/1711.05144.pdf.

Kearns, M. J. and Vazirani, U. V. *An Introduction to Computational Learning Theory*. MIT Press, 1994.

Sion, M. On general minimax theorems. *Pacific Journal of Mathematics*, 8(1):171–176, 1958. URL https://projecteuclid.org/euclid.pjm/1103040253.

7. Appendix: Deferred Proofs

Lemma 4.1. *For $k = 2$, multi-class statistical parity fairness (definition 4.1) holds if and only if binary statistical parity fairness (definition 3.1) holds.*

Proof. The forward direction is trivial since the classifier D being γ-multi-class-fair with respect to $\mathcal{P}$, g and for all $k \in 0, 1$ implies it is γ-binary-fair with respect to $\mathcal{P}$ and g.

Now assume classifier D maintains fairness as defined in definition 3.1. Then we have,

$$\alpha_{SP}(g, \mathcal{P})\beta_{SP}(g, D, \mathcal{P}, 1) \leq \gamma$$

Now,

$$\beta_{SP}(g, D, P, 0) = |SP(D) - SP(D, 0, P)|$$
$$= |Pr_{\mathcal{P},D}[D(X) = 0] - Pr_{\mathcal{P},D}[D(X) = 0|g(x) = 1]|$$
$$= |(1 - Pr_{\mathcal{P},D}[D(X) = 1]) - (1 - Pr_{\mathcal{P},D}[D(X) = 1|g(x) = 1])|$$
$$= |Pr_{\mathcal{P},D}[D(X) = 1] - Pr_{\mathcal{P},D}[D(X) = 1|g(x) = 1]|$$
$$= \beta_{SP}(g, D, \mathcal{P}, 1)$$

where we have used the fact that probability densities sum up to 1. But we already have

$$\alpha_{SP}(g, \mathcal{P})\beta_{SP}(g, D, \mathcal{P}, 1) \leq \gamma$$

which implies,

$$\alpha_{SP}(g, \mathcal{P})\beta_{SP}(g, D, \mathcal{P}, 0) \leq \gamma$$

or fairness as defined in definition 4.2 holds for classifier D, completing the proof. □

Theorem 4.1. *With probability $1 - \delta$, for all $h \in \mathcal{H}$,*

$$|err(h, \mathcal{P}) - err(h, S)| \leq O\left(k\frac{VCDIM(\mathcal{H})\log(n)+\log(k/\delta)}{n}\right).$$

Proof.

For each $j \in \{0, ..., k-1\}$, conceive of h (which outputs one of these k classes) as making the binary decision to output j or $\neg j$ (i.e. $l \in \{0, ..., k\}\ j\}$) and denote this now-binary classifier h by h_j. By theorem 2.10 of Kearns et al. (Kearns et al., 2018b), with probability $1 - \delta/k$,

$$|\text{err}(h_j, \mathcal{P}) - \text{err}(h_j, \text{S})| \leq \text{O}\left(\frac{\text{VCDIM}(\mathcal{H})\log(n)+\log(k/\delta)}{n}\right). \tag{7.1}$$

Thus, with probability $(1 - \delta/k)^k \geq 1 - \delta$, equation 5.1 holds for all $j \in \{0, ..., k-1\}$.
From here, observe that

$$\text{err}(h, \text{S}) = \frac{1}{n}\sum_{i=1}^{n}\mathbb{1}[h(X_i) \neq y_i] = \frac{1}{2}\sum_{j=0}^{k-1}\text{err}(h_j, \text{S}). \tag{7.2}$$

because any error, misclassifying an example of class p as class q appears as a non-zero term in the summation for exactly two of the h_j, namely h_p and h_q. Similarly,

$$\text{err}(h, \mathcal{P}) = \frac{1}{2}\sum_{j=0}^{k-1}\text{err}(h_j, \mathcal{P}). \tag{7.3}$$

By equation 7.1,

$$\sum_{j=0}^{k-1}|\text{err}(h_j,\mathcal{P})-\text{err}(h_j,\text{S})| \leq \text{O}\left(k\frac{\text{VCDIM}(\mathcal{H})\log(n)+\log(k/\delta)}{n}\right)$$

Equations 7.2, 7.3, and the triangle inequality give

$$|\text{err}(h,\mathcal{P})-\text{err}(h,\text{S})| \leq \frac{1}{2}\sum_{j=0}^{k-1}|\text{err}(h_j,\mathcal{P})-\text{err}(h_j,\text{S})|.$$

From here, absorbing the constant factor of $\frac{1}{2}$ gives the desired result.
□

Theorem 4.2. *For any fixed set of group indicators $\mathcal{G}$, with probability $1-\delta$, for any $h \in \mathcal{H}$ and any $g \in \mathcal{G}$, for any $j \in [\![1,k]\!]$, the following holds:*

$$|\alpha_{SP}(g,\mathcal{P})\beta_{SP}(g,\mathcal{P},D,j)-\alpha_{SP}(g,\mathcal{P})\beta_{SP}(g,\mathcal{P},D,j)| \leq \tilde{O}\left(\sqrt{\frac{VCDIM(\mathcal{H})+VCDIM(\mathcal{G})\log(m)+\log(k/\delta)}{m}}\right).$$

Proof. Fix $j \in [\![0,k]\!]$. Using theorem 2.11 (SP Uniform Convergence) of Kearn et al. (Kearns et al., 2018b), we see that with probability $1-\delta/k$, we have,

$$|\alpha_{SP}(g,\mathcal{P})\beta_{SP}(g,\mathcal{P},D,j)-\alpha_{SP}(g,\mathcal{P})\beta_{SP}(g,\mathcal{P},D,j)| \leq \tilde{O}\left(\sqrt{\frac{\text{VCDIM}(\mathcal{H})+\text{VCDIM}(\mathcal{G})\log(m)+\log(k/\delta)}{m}}\right).$$

Now assuming each of these conditions to hold separately, with probability $(1-\delta/k)^k \geq 1-\delta$, we see that the condition in the theorem holds, completing the proof.

□

Theorem 4.3. *With reasonable assumptions on the group of indicators $\mathcal{G}$, and $\gamma \in [0,1)$, there exist distributions $\mathcal{P}$ for which no polynomial time algorithm can arrive at the optimal k-class classifier which is γ-SP fair.*

Proof. Proceed by contradiction, and assume for some $k \in \mathbb{N}$, $k \geq 2$, we have a polynomial time algorithm that can accomplish the stated task in the theorem. Assume the set up for a binary classification problem from section 3. Arbitrarily assign the datapoints labeled with 1 into $k-1$ classes (from 1 to $k-1$). Now our polynomial time algorithm can arrive at an optimal k-class classifier which is also γ-SP fair. Let this classifier be D. Define classifier D' as: $D'(X) = \begin{cases} 0 & D(X)=0 \\ 1 & 1 \leq D(X) \leq k-1 \end{cases}$

See that D' is a classifier that is γ-SP fair for our original binary classification setup, which we also obtained in polynomial time. However, by theorem 3.6 of Kearns et al. (Kearns et al., 2018b), this is impossible, since there is no polynomial time algorithm that can arrive at the optimal binary classifier which is γ-SP fair. Thus we have reached a contradiction, and prove the desired result. □