Identifying Statistical Bias in Dataset Replication

Logan Engstrom*
engstrom@mit.edu
MIT

Andrew Ilyas*
ailyas@mit.edu
MIT

Shibani Santurkar
shibani@mit.edu
MIT

Dimitris Tsipras
tsipras@mit.edu
MIT

Jacob Steinhardt
jsteinhardt@berkeley.edu
UC Berkeley

Aleksander Mądry
madry@mit.edu
MIT

Abstract

Dataset replication is a useful tool for assessing whether improvements in test accuracy on a specific benchmark correspond to improvements in models' ability to generalize reliably. In this work, we present unintuitive yet significant ways in which standard approaches to dataset replication introduce statistical bias, skewing the resulting observations. We study ImageNet-v2, a replication of the ImageNet dataset on which models exhibit a significant (11-14%) drop in accuracy, even after controlling for a standard human-in-the-loop measure of data quality. We show that after correcting for the identified statistical bias, only an estimated $3.6\% \pm 1.5\%$ of the original $11.7\% \pm 1.0\%$ accuracy drop remains unaccounted for. We conclude with concrete recommendations for recognizing and avoiding bias in dataset replication. Code for our study is publicly available[1].

1 Introduction

The primary objective of supervised learning is to develop models that generalize robustly to unseen data. Benchmark test sets provide a proxy for out-of-sample performance, but can outlive their usefulness in some cases. For example, evaluating on benchmarks alone may steer us towards models that adaptively overfit [Reu03; RFR08; Dwo+15] to the finite test set and do not generalize. Alternatively, we might select for models that are sensitive to insignificant aspects of the dataset creation process and thus do not generalize robustly (e.g., models that are sensitive to the exact set of humans who annotated the test set).

To diagnose these issues, recent work has generated new, previously "unseen" testbeds for standard datasets through a process known as dataset replication. Though not yet widespread in machine learning, dataset replication is a natural analogue to experimental replication studies in the natural sciences (cf. [Bel73]). These studies play an important role in verifying empirical findings, and ensure that results are neither affected by adaptive data analysis, nor overly sensitive to experimental artifacts.

Recent dataset replication studies [Rec+19b; Rec+19a; YB19] have generally found little evidence of adaptive overfitting: progress on the original benchmark translates to roughly the same amount (or more) of progress on newly constructed test sets. On the other hand, model performance on the replicated test set tends to drop significantly from the original one.

One of the most striking instances of this accuracy drop is observed by Recht et al. [Rec+19b], who performed a careful replication of the ImageNet dataset and observe an 11-14% gap between model accuracies on ImageNet and their new test set, ImageNet-v2. The magnitude of this gap presents an empirical mystery, and motivates us to understand what factors cause such a large drop in accuracy.

In this paper, we identify a mechanism through which the dataset replication process itself might lead to such a drop: noisy readings during data collection can introduce statistical bias. We show that this

---

[1]https://github.com/MadryLab/dataset-replication-analysis

mechanism may explain all but $3.6\% \pm 1.5\%$ of the original $11.7\% \pm 1.0\%$ accuracy drop between ImageNet and ImageNet-v2.

Our explanation revolves around what we refer to as the "statistic matching" step of dataset replication. Statistic matching ensures that model performance on the original test set and its replication are comparable by controlling for variables that are known to (or hypothesized to) impact model performance.[2] Drawing a parallel to medicine, suppose we wanted to replicate a study about the effect of a certain drug on an age-linked disease. After gathering subjects, we have to reweight or filter them so that the age distribution matches that of the original study—otherwise, the results of the studies are incomparable. This filtering/reweighting step is analogous to statistic matching in our context, with participant age being the relevant statistic.

To construct ImageNet-v2, Recht et al. [Rec+19b] perform statistic matching based on the "selection frequency" statistic, which for a given image-label pair measures the rate at which crowdsourced annotators select the pair as correctly labeled. As we discuss in the next section, selection frequency is a well-motivated choice of matching statistic, since (a) Deng et al. [Den+09] use a similar metric to gather ImageNet images in the first place [Den+09], and (b) Recht et al. [Rec+19b] have found that selection frequency is highly predictive of model accuracy.

Why does a significant drop in accuracy persist even after matching selection frequencies? In this paper, we show that (inevitable) mean-zero noise in selection frequency readings leads to bias in the selection frequencies of the replicated dataset, which translates to a drop in model accuracies. We also discuss how finite-sample reuse makes this bias difficult to detect.

The bias-inducing mechanism that we identify applies whenever statistic matching is performed using noisy estimates. We characterize the mechanism theoretically in Section 2. In Section 3, we remeasure selection frequencies using Mechanical Turk and observe that as our mechanism predicts, ImageNet-v2 images indeed have lower selection frequency on average. After presenting a framework for studying the effect of statistical bias on model accuracy (Section 4), we use de-biasing techniques to estimate a bias-corrected accuracy for ImageNet-v2 (Section 5). In Section 7, we discuss the implications of the identified mechanism for ImageNet-based computer vision models specifically, and for data replication studies more generally.

2 Identifying Sources of Reproduction Bias

The goal of dataset replication is to create a new dataset by reconstructing the pipeline that generated the original test set as closely as possible. We expect (and intend) for this process to introduce a distribution shift, partly by varying parameters that should be irrelevant to model performance (e.g. the exact identity of the annotators used to filter the dataset). To ensure that results are comparable with original test sets, however, dataset replication studies must control for distribution shifts in variables that impact task performance. This is accomplished by subsampling or reweighting the data so that each relevant variable's distributions under the replicated dataset and the original dataset match one another. We refer to this process as *statistic matching*.

Our key observation is that standard approaches to statistic matching can lead to bias in the final replicated dataset: we illustrate this phenomenon in the context of the ImageNet-v2 (v2) dataset replication [Rec+19b][3]. Before we identify the source of this bias in ImageNet-v2 construction, we review the data collection process for both ImageNet and ImageNet-v2.

ImageNet and selection frequency. ImageNet [Den+09; Rus+15] (which we also refer to as ImageNet-v1 or v1) is one of the most widely used datasets in computer vision. To construct ImageNet, Deng et al. [Den+09] first amassed a large candidate pool of image-label pairs using image search engines such as Flickr. The authors then asked annotators on Amazon Mechanical Turk (MTurk) to select the candidate

---

[2]In causal inference terms, statistic matching is an instance of covariate balancing [IR13].

[3]Note that Recht et al. [Rec+19b] actually design three datasets, in order to measure the effect of selection frequency on model performance. We focus our attention on `MatchedFrequency` (referred to as just ImageNet-v2 here, in Recht et al. [Rec+19b] and elsewhere [Tao+20; LDS20; RM19; RSP19]), since it is the only dataset designed to replicate the ImageNet validation set.

| Selection Frequency: 36% | Selection Frequency: 61% | Selection Frequency: 100% |

Figure 1: The smallest, median, and largest selection frequency images from v1 corresponding to the "throne" class (description: *the chair of state for a monarch, bishop, etc.; "the king sat on his throne"*—the "throne" class was randomly chosen). The images become easier to identify as the labeled class as selection frequency increases; for additional context, we give a random sampling of selection frequency/image pairs in Appendix B.

images that were correctly labeled. Each image is shown to multiple annotators, and an image's *selection frequency* [4] is then defined as the fraction of annotators that selected it.

Intuitively, images with low selection frequency are likely confusing or their proposed label is incorrect, while images with high selection frequency should be "easy" for humans to identify as the proposed label (we show examples of selection frequencies in Figure 1; further examples are in Appendix 8). Therefore, Deng et al. [Den+09] include only images with high selection frequency in the final ImageNet dataset[5].

ImageNet-v2. ImageNet-v2 is a replication of ImageNet-v1 that controls for selection frequency via statistic matching. Following the protocol of Deng et al. [Den+09], Recht et al. [Rec+19b] collected a large pool of candidate image-label pairs, and estimated their selection frequencies via MTurk along with a subset of the original ImageNet validation set. Recht et al. [Rec+19b] then estimated the distribution of ImageNet-v1 selection frequencies for each class. Finally, they subsampled ten images of each class from the candidate pool according to the estimated class-specific distributions.

For example, suppose 40% of "goldfish" images in ImageNet-v1 have selection frequency in the histogram bucket $[0.6, 0.8]$—when constructing ImageNet-v2, Recht et al. [Rec+19b] would in turn sample 4 "goldfish" images from the same histogram bucket in the candidate images[6].

Statistic matching should ensure that v1 and v2 are balanced in terms of selection frequency, and partly justifies the expectation that models should perform similarly on both.

Sources of bias. We identify two places where the matching strategy of Recht et al. [Rec+19b] might introduce statistical bias. One potential source of bias could arise from binning the images into histograms—since there are relatively few bins, it is possible that within each bin the ImageNet images have different selection frequencies from the corresponding Flickr images. (For example, the ImageNet-v1 images in the $s(x) \in [0., 0.2]$ bucket might actually have selection frequency concentrated around 0.15, whereas the Flickr images in the same bucket might have been concentrated around $s(x) = 0.1$.) However, this source of error appears to have not had a pronounced effect (at least on average), as Recht et al. [Rec+19b] report that the average selection frequency of the ImageNet-v2 images actually matches that of the ImageNet-v1 test set.

Our analysis revolves around a second and more subtle source of bias, however. This bias stems from the fact that for any given image x, the selection frequency $s(x)$ is never measured exactly. Instead, we are only able to measure $\hat{s}(x)$, a finite-sample estimate of the statistic, attained by averaging over a relatively small number of annotators.

---

[4]Note that the term "selection frequency" was in fact coined by Recht et al. [Rec+19b], but it is also useful for describing the initial setup of Russakovsky et al. [Rus+15], who instead referred to their process as "majority voting."

[5]Specifically, an image is included into the ImageNet test set if a "convincing majority" [Rus+15] of annotators select it.

[6]Note that in Recht et al. [Rec+19b] this process is done on a class-by-class basis and also includes provisions for when there are not enough Flickr images in a particular bin, but the core distribution-matching mechanism is otherwise identical.
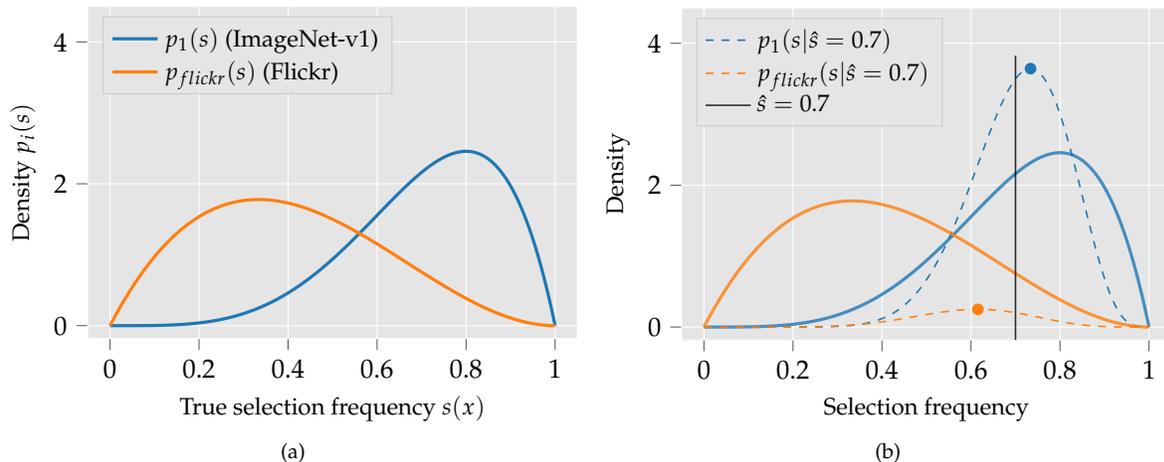
Figure 2: For an image x, the *selection frequency* statistic $s(x)$ described in Section 2 is a single number in $[0, 1]$ that captures how "easy" a given image is to classify for humans. A distribution over images $(p_i(x))$ thus induces a one-dimensional distribution over selection frequencies $(p_i(s(x)))$. In **(a)**, we visualize such hypothetical selection frequency distributions for both the Flickr data distribution $(p_{flickr}(s(x)))$ and the ImageNet-v1 data distribution $(p_1(s(x)))$. In **(b)**, we consider a case where we are given, for a specific image x, a *noisy* version of $s(x)$ ($\hat{s}(x)$). We visualize the corresponding distribution of the true selection frequency $s(x)$ given this noisy $\hat{s}(x) = 0.7$. As discussed in Section 2, note that even though $\hat{s}(x)$ is an *unbiased* estimate of $s(x)$, the most likely value of $s(x)$ for a given noisy reading of $\hat{s}(x)$ actually depends on the distribution from which x is drawn. This is the driving phenomenon behind the observed bias between ImageNet and ImageNet-v2.

To model the impact of this seemingly innocuous detail, suppose that the selection frequencies $s(x)$ of ImageNet and Flickr images are distributed according to $p_1(s(x))$ and $p_{flickr}(s(x))$ respectively (or more briefly, $p_1(s)$ and $p_{flickr}(s)$)—see Figure 2a for a visualization. Now, suppose that for an image x, we get an unbiased noisy measurement $\hat{s}(x) = 0.75$ of the selection frequency via crowdsourcing. Then, even if $\hat{s}(x)$ is an unbiased estimate of $s(x)$, the most likely value of $s(x)$ for the image is not $\hat{s}(x)$, but in fact depends on the distribution from which x was drawn. Indeed, for the (hypothetical) distributions shown in Figure 2a, if x is a Flickr image then it is more likely that $s(x) < 0.75$ and $\hat{s}$ is an overestimate, since a priori an image is likely to have a low selection frequency (i.e., there is more $p_{flickr}(s)$ mass below 0.75) and the noise is unbiased. Conversely, if x is an ImageNet test set image in this same setting, it is more likely that $s(x) > 0.75$. Therefore, if we use a Flickr image with a noisy selection frequency 0.75 to "match" an ImageNet image with the same noisy selection frequency, the true selection frequency of the ImageNet image is actually likely to be higher. We can make this explicit by writing down the likelihood of s given $\hat{s} = 0.75$ (also plotted in Figure 2b):

$$p_i(s|\hat{s} = 0.75) = \frac{p_i(s) \cdot p(\hat{s} = 0.75|s)}{p_i(\hat{s} = 0.75)} \qquad \forall\, i \in \{1, flickr\},$$

which depends on the prior $p_i(\cdot)$ and therefore is not equal for both values of i.

The distribution of candidate Flickr images is likely skewed to have lower selection frequencies than v1—after all, Deng et al. [Den+09] narrowed down Imagenet-v1 from a large set of candidates based on quality. Therefore, one would expect the underlying true selection frequencies of the v1 images to be higher than their matched ImageNet-v2 counterparts (as illustrated in Figure 2a). More generally, bias arises from the fact that even though $\hat{s}(x)$ given x is unbiased (e.g. $E[\hat{s}(x)|x] = E[\hat{s}(x)|s(x)] = s(x)$), functions of x given $\hat{s}(x)$ are not necessarily unbiased for many natural statistics (i.e. $E[q(x)|\hat{s}(x)] \neq E[q(x)|s(x)]$ for statistics q).
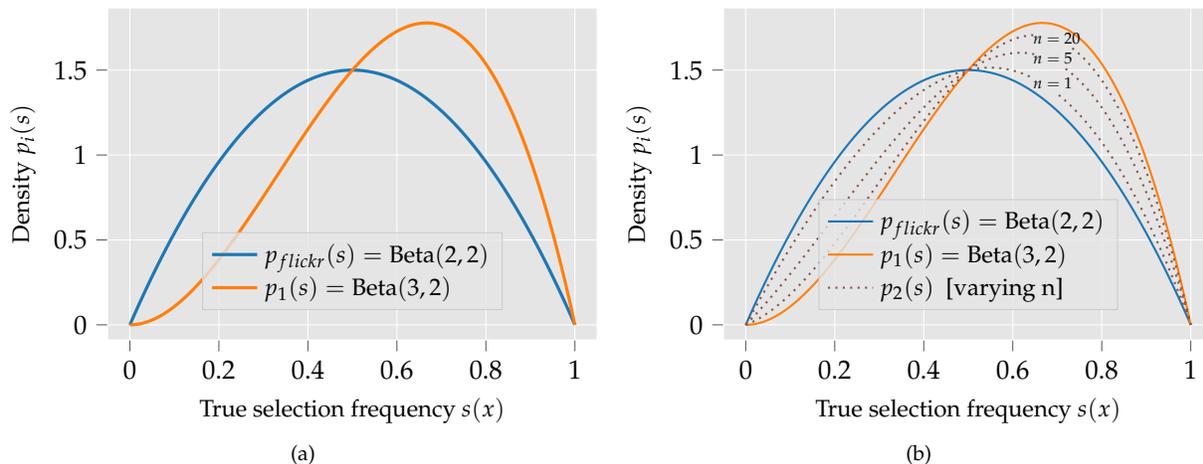
Figure 3: Illustrations accompanying the simple theoretical model. **(a)** In the simple model, we assume $p_1(s(x))$ and $p_{flickr}(s(x))$ are Beta$(\alpha+1,\beta)$ and Beta(α,β), respectively—this is visualized above for the case of $\alpha = \beta = 2$. **(b)** The results of the simple model reveal that as more and more samples are used to estimate $s(x)$ for each image, the resulting ImageNet-v2 distribution tends towards the v1 distribution, but does not actually match the v1 sample for any finite number of samples per image.

A simple model of the bias. To better understand the source of the bias, consider a simple model in which the ImageNet-v2 selection process is cast as a rejection sampling procedure. Here, the densities $p_1(\hat{s}(x))$ and $p_{flickr}(\hat{s}(x))$ are estimated from samples (analogous to the histograms of Recht et al. [Rec+19b])—then, for a given Flickr image x, we "accept" x into the v2 dataset with probability proportional to $p_1(\hat{s}(x))/p_{flickr}(\hat{s}(x))$ (analogous to the bin-wise sampling of Recht et al. [Rec+19b]). If selection frequency readings were not noisy, i.e. if $\hat{s}(x) = s(x)$, then the resulting density of selection frequencies in the v2 dataset would be given by

$$p_{flickr}(s(x)) \cdot \frac{p_1(s(x))}{p_{flickr}(s(x))} = p_1(s(x)),$$

and the selection frequencies of v2 would be distributed in the same way as those of v1, as intended. However, the inevitable noisiness of the selection frequencies means that in reality, the density of selection frequencies for v2 will be given by

$$p_{flickr}(s(x)) \cdot \mathbb{P}(x \text{ is accepted into v2}|s(x)) = p_{flickr}(s(x)) \cdot \int_{\hat{s}} p(\hat{s}|s)\mathbb{P}(x \text{ is accepted}|\hat{s}(x))$$

$$= p_{flickr}(s(x)) \cdot \int_{\hat{s}} p(\hat{s}|s) \frac{p_1(\hat{s}(x))}{p_{flickr}(\hat{s}(x))}.$$

Now, suppose for simplicity that $p_{flickr}(s)$ and $p_1(s)$ are given by beta distributions Beta(α,β) and Beta$(\alpha+1,\beta)$ respectively[7] (c.f. Figure 3a). Furthermore, suppose that $\hat{s}(x)$ is given by an average of n Bernoulli draws with success probability $s(x)$. Then, a series of calculations (shown in Appendix C) reveals that the resulting v2 selection frequency distribution is given by:

$$\frac{n}{n+\beta+\alpha} \cdot \text{Beta}(\alpha+1,\beta) + \frac{\alpha+\beta}{n+\beta+\alpha} \cdot \text{Beta}(\alpha,\beta) = \frac{n}{n+\beta+\alpha} \cdot p_1(s) + \frac{\alpha+\beta}{n+\alpha+\beta} \cdot p_{flickr}(s). \quad (1)$$

Note that as $n \to 0$ (no filtering is done at all), the above expression evaluates to exactly $p_{flickr}(s)$, as expected. Then, as the number of workers n tends to infinity (i.e. $\hat{s}$ becomes less noisy), the distribution of ImageNet-v2 selection frequencies converges to the desired $p_1(s)$. For any finite n, however, the resulting

---

[7]This is a simple model intended for illustrative purpose—we will later use a more sophisticated model to capture the actual distribution of selection frequencies.
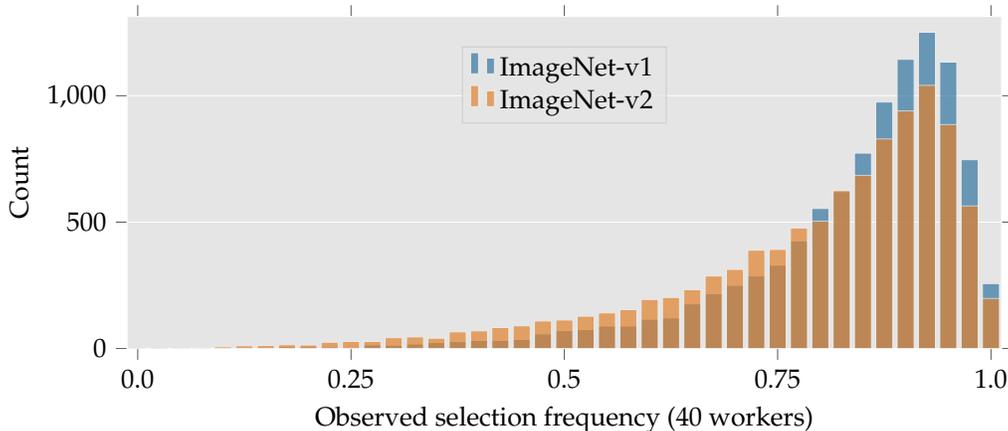
Figure 4: Selection frequency histograms for v1 and v2 based on our selection frequency re-measurement experiment. Results indicate that v2 seems to have lower selection frequency.

v2 distribution will be a non-degenerate mixture between $p_{flickr}(s)$ and $p_1(s)$, and therefore does not match the distribution of selection frequencies $p_1(s)$ exactly. The results of this toy model (depicted in Figure 3b) capture the bias that could be incurred by the data replication pipeline of Recht et al. [Rec+19b]. In the next section, we set out to quantify the bias suffered by the actual pipeline of Recht et al. [Rec+19b].

3 Remeasuring Selection Frequencies

In this section, we measure the effect of the described noise-induced bias on the true and observed selection frequencies of images in ănd v2. Using an annotation task closely resembling those of the ImageNet-v2 and ImageNet MTurk experiments, we collect new selection frequency estimates for all of ImageNet-v2 and for a subset of ImageNet. In these tasks, MTurk annotators were shown grids of 48 images at a time, each corresponding to an ImageNet class. Each grid contained a mixture of ImageNet, Flickr, and in our case, ImageNet-v2 images of the corresponding class (since ImageNet-v2 was not yet realized at the time of the other experiments), as well as control images from other classes. We describe the setup in more detail in Appendix B.1. Annotators were tasked with selecting all the images in the grid containing an object from the class in question. Each image was seen by 40 distinct annotators, and assigned an observed selection frequency equal to the fraction of these workers that selected it.

Histograms of observed selection frequencies for v1 and v2 are shown in Figure 4. We find that the average selection frequencies of the v1 and v2 images were 85.2% ± 0.1% and 80.7% ± 0.1% respectively compared to 71% and 73% reported by Recht et al. [Rec+19b] [8]. This means that the initial 2% gain in selection frequency measured by Recht et al. [Rec+19b] turns into a 5% drop. Our model of dataset replication bias predicts this discrepancy: once observed selection frequencies are used for statistic matching, they no longer provide an unbiased estimate of true selection frequency[9].

Detecting bias using the original data. Our MTurk task measures a significant selection frequency gap between v1 and v2 (~5%), but also measures average selection frequencies for both datasets to be significantly higher than reported by Recht et al. [Rec+19b], suggesting differences in experimental setup. Indeed, while the tasks themselves were identical, we did make a few changes to the deployment setup of Recht

---

[8] 95% bootstrapped CI. The average selection frequency (unlike other statistics we discuss) is always unbiased, and thus average observed selection frequency will converge quickly to the average true selection frequency by law of large numbers.

[9] To draw an analogy here, suppose that instead of matching image datasets, we were matching two piles of coins: a rigged pile A ($P_A(\text{heads}) = 1$) and a fair pile B ($P_B(\text{heads}) = 0.5$). To match pile B to pile A, we flip the coins in both piles 10 times each—inevitably, some of the coins in pile B will land heads all 10 times. These "matched" coins are identical to pile A coins with respect to the observed "number of heads" statistic, but are obviously not identical coins. Yet, even though flipping the matched coins another 10 times would reveal this, it is impossible to conclude anything other than $P(\text{heads}) = 1$ solely from the already-collected data on the matched coins.

et al. [Rec+19b] to improve data quality. These changes are outlined in Appendix B.2: examples include introducing worker screening qualifications[10], and using different proportions of images per grid. Since the task interface remained constant and workers are not able to distinguish between ImageNet-v1 and ImageNet-v2 images while labeling, we believe that the changes made improve data quality across both datasets while negligibly affecting the selection frequency gap between them.

Still, we can fully control for experimental differences by analyzing the raw data of Recht et al. [Rec+19b] directly, taking care to avoid bias from observed selection frequency reuse. We defer the exact data analysis to Appendix D; in summary, we perform three experiments:

- Recall that Recht et al. [Rec+19b] perform statistic matching using observed selection frequencies, measured with $n = 10$ annotators, to get ImageNet-v2. We gradually decreased n to study the effect of finite-sample noise, and found that model accuracy on the resulting replicated dataset degrades. For example, the accuracy gap from v1 to the replication increases from 12% when $n = 10$, to 14% when $n = 5$. This is consistent with our model of statistic matching bias: fewer annotators means noisier observed selection frequencies $\hat{s}_n(x)$, which in turn amplifies the effect of the bias, driving down model accuracies.

- In the second experiment, we repeat the statistic matching process of Recht et al. [Rec+19b] using five "in-sample" annotators per candidate image, reserving the remaining annotations as a held-out set. We find that the average selection frequency measured by the in-sample annotations overestimates the true average selection frequency (i.e., as measured by the held-out set) by 2-3%. In fact, the held-out average selection frequency is consistently lower for the replicated dataset than for ImageNet-v1.

- Finally, we use the held-out set from this second experiment to filter the candidate pool via a held-out selection frequency cutoff. This skews the distribution of true selection frequencies in the candidate pool towards higher values. As predicted by our model of statistic-matching bias, using this skewed subset in place of the full candidate pool for statistic matching results in increased model accuracy (yet identical in-sample selection frequencies).

These results suggest that statistic matching bias affects the v2 dataset, even fully controlling for experimental setup. In the coming sections, we quantify the effects of this bias on the model accuracies observed by Recht et al. [Rec+19b].

4 Understanding the Accuracy Gap

Our findings so far have suggested that statistic matching bias results in a downwards bias in ImageNet-v2 true selection frequencies. In this section, we quantify the impact on this bias on ImageNet-v2 accuracy.

4.1 Notation and terminology

Here we overview the notation and terminology useful in discussing the bias in ImageNet-v2 accuracy.

Selection frequencies. In Section 2 we defined the true selection frequency $s(x)$ for an image x to be the (population) rate at which crowd annotators select the image as "correctly labeled." The true selection frequency of an image is unobservable, and often approximated by the observed selection frequency, $\hat{s}_n(x) \sim \frac{1}{n}\text{Binom}(n, s(x))$, which can be estimated from an n-annotator MTurk experiment. When n is clear from context we will often omit it and write $\hat{s}(x)$.

---

[10]Worker qualification is a service provided by MTurk that only allows "high-reputation" annotators (typically measured by historical annotation quality on the platform) to complete a given task. Qualifications have been shown to significantly impact data quality: in [PVA13], using qualifications lowered the number of inattentive workers from 33% to less than 1%.

Distributions. We will use $\mathcal{D}_1$ and $\mathcal{D}_2$ to denote the distributions of v1 and v2 images respectively, and $\mathcal{S}_1$ and $\mathcal{S}_2$ to denote the corresponding finite test sets. As in Section 2, we denote by $p_i(s)$ the probability density of true selection frequencies for images drawn from $\mathcal{D}_i$ (or Flickr, if $i = flickr$). Similarly, we use $p_i(\hat{s}_n(x))$ to denote the probability mass function of the observed selection frequency for dataset i.

We let $\mathcal{D}_2|s_1$ be the distribution of ImageNet-v2 images reweighted to have the same selection frequency distribution as ImageNet-v1. Formally, $\mathcal{D}_2|s_1$ is the compound distribution $(x_2 \sim \mathcal{D}_2|s(x_2) \sim p_1(s))$. Sampling from $\mathcal{D}_2|s_1$ corresponds to first sampling a v1 image x_1, then sampling an image x_2 from the v2 distribution conditioned on $s(x_2) = s(x_1)$.

Accuracies. For a classifier c, let $f_c(x)$ be an indicator variable of whether c correctly classifies x. Since our analysis applies to any fixed classifier c, we omit it and use $f(x)$. We then define $\mathcal{A}_X$ to represent classifier accuracy on distribution or test set X—for example, classifier accuracy on v1 is given by

$$\mathcal{A}_{\mathcal{D}_1} = \mathbb{P}_{x_1 \sim \mathcal{D}_1} \left(f(x_1) = 1 \right) = \mathbb{E}_{x_1 \sim \mathcal{D}_1} \left[f(x_1) \right].$$

4.2 Breaking down the accuracy gap

The accuracy gap between the v1 and v2 test sets is given by $\mathcal{A}_{\mathcal{S}_1} - \mathcal{A}_{\mathcal{S}_2}$. What fraction of this gap can be attributed to bias in selection frequency? To answer this, we decompose this accuracy gap into three elements whose contribution can be studied separately:

$$\mathcal{A}_{\mathcal{S}_1} - \mathcal{A}_{\mathcal{S}_2} = \underbrace{\left(\mathcal{A}_{\mathcal{S}_1} - \mathcal{A}_{\mathcal{D}_2|s_1} \right)}_{\text{bias-corrected accuracy gap}} + \underbrace{\left(\mathcal{A}_{\mathcal{D}_2|s_1} - \mathcal{A}_{\mathcal{D}_2} \right)}_{\text{selection gap}} + \underbrace{\left(\mathcal{A}_{\mathcal{D}_2} - \mathcal{A}_{\mathcal{S}_2} \right)}_{\text{finite sample gap} \approx 0}. \tag{2}$$

Bias-corrected accuracy gap. The first term of (2), called the bias-corrected accuracy gap, captures the portion of the v1-v2 accuracy drop that *cannot* be explained by a difference in selection frequency, and instead might be explained by benign distribution shift or adaptive overfitting[11].

Selection gap. The second term of (2) is accuracy gap that can *only* be attributed to selection frequency, since it compares accuracy on $\mathcal{D}_2$ to accuracy on a reweighted version of $\mathcal{D}_2$. If there was no bias, and the distribution of selection frequencies for v1 and v2 matched exactly, then this term would equal zero ($\mathcal{D}_2|s_1$ would equal $\mathcal{D}_2$). Thus, the selection gap translates the effect of discrepancy in true selection frequency between v1 and v2 into a discrepancy in accuracy. Since we measured v1 as having higher true selection frequency, we expect the selection gap to be positive and thus explain a portion of the accuracy gap that was previously attributed to distribution shift.

Finite-sample error. The final term refers to the finite-sample error from using $10,000$ images as a proxy for distributional accuracy. We believe that this term is negligible, since (a) 95% bootstrapped confidence intervals for the classifiers we evaluate are all at most 0.1%, and (b) there can be no adaptive overfitting on $\mathcal{S}_2$ with respect to $\mathcal{D}_2$. Thus, we drop this term from consideration and instead use $\mathcal{A}_{\mathcal{D}_2}$ and $\mathcal{A}_{\mathcal{S}_2}$ interchangeably.

Computing selection-adjusted accuracy. We have shown how to decompose the v1-v2 accuracy gap into a component explained by selection frequency (selection gap), and a component unexplained by selection frequency (bias-corrected accuracy gap). The challenge in computing this decomposition is estimating $\mathcal{A}_{\mathcal{D}_2|s_1}$, the selection-adjusted v2 accuracy. While the closed form of $\mathcal{A}_{\mathcal{D}_2|s_1}$ is

$$\int_s \mathbb{E}_{\mathcal{D}_2}[f(x)|s(x) = s] \cdot p_1(s) \, ds,$$

---

[11] In fact, this term can be further decomposed into a sum of an adaptivity gap ($\mathcal{A}_{\mathcal{S}_1} - \mathcal{A}_{\mathcal{D}_1}$), and a distribution shift gap ($\mathcal{A}_{\mathcal{D}_1} - \mathcal{A}_{\mathcal{D}_2|s_1}$). However, since we don't have access to $\mathcal{A}_{\mathcal{D}_1}$ it is difficult to disentangle these.

we have no access to $p_i(s)$ for any value of i (we do not even have direct access to $s(x)$ for any image x). In the next section, we explore methods for estimating $\mathcal{A}_{\mathcal{D}_2|s_1}$ using only the observed selection frequencies that we collected.

5 Quantifying the Bias

In the previous sections, we showed that statistic matching based on noisy observed selection frequencies may lead ImageNet-v2 images to have lower true selection frequencies than expected. In Section 4 we related this discrepancy in selection frequency to a corresponding discrepancy in model accuracy between v1 and v2, which we called the "selection gap." In this section, we explore a series of methods for estimating this gap—we estimate that the selection gap accounts for 8.1% of the 11.7% v1-v2 accuracy drop.

5.1 Naïve approach

In the last section we introduced the selection-adjusted v2 accuracy,

$$\mathcal{A}_{\mathcal{D}_2|s_1} = \int_s \mathbb{E}_{\mathcal{D}_2}[f(x)|s(x) = s] \cdot p_1(s)\ ds, \tag{3}$$

which captures model accuracy on a version of ImageNet-v2 reweighted to have the same true selection frequency distribution of ImageNet-v1. Since we do not observe true selection frequencies, we cannot evaluate $\mathcal{A}_{\mathcal{D}_2|s_1}$, and are instead forced to estimate it. A natural way to do so is to use observed selection frequencies in place of true ones, leading to the following "naïve estimator:"

$$\hat{\mathcal{A}}^n_{\mathcal{D}_2|s_1} = \sum_{k=0}^n \mathbb{E}_{x_2 \sim \mathcal{D}_2}\left[f(x_2)|\hat{s}_n(x_2) = \frac{k}{n}\right] \cdot p_1\left(\hat{s}_n(x_1) = \frac{k}{n}\right). \tag{4}$$

The naïve estimator is a computable[12] but biased estimator of the selection-adjusted accuracy. This follows from our analysis in Section 2, since $\hat{\mathcal{A}}^n_{\mathcal{D}_2|s_1}$ is just a mechanism for statistic matching between ImageNet-v1 and ImageNet-v2 using observed selection frequencies in place of true selection frequencies. Thus, the selection-adjusted v2 accuracy computed by the naïve estimator is likely to still underestimate the true selection-adjusted accuracy $\mathcal{A}_{\mathcal{D}_2|s_1}$.

We can verify this bias empirically by varying the number of annotators n used to calculate $\hat{s}_n(x)$ for each image, and visualizing the resulting trends in $p_i(\hat{s}_n(x))$ (Figure 5a), $p_i(f(x) = 1|s(x))$ (Figure 5b), and $\hat{\mathcal{A}}^n_{\mathcal{D}_2|s_1}$ (Figure 5c). The results corroborate our analysis in Section 2 and our findings from Section 3. Specifically, Figure 5 plots each term in the definition of the naïve estimator,

$$\underbrace{\hat{\mathcal{A}}^n_{\mathcal{D}_2|s_1}}_{\text{Fig. 5c}} = \sum_{k=0}^n \underbrace{\mathbb{E}_{x_2 \sim \mathcal{D}_2}\left[f(x_2)|\hat{s}_n(x_2) = \frac{k}{n}\right]}_{\text{Fig. 5b}} \cdot \underbrace{p_1\left(\hat{s}_n(x_1) = \frac{k}{n}\right)}_{\text{Fig. 5a}}, \tag{5}$$

and allows us to draw the following conclusions:

- Figure 5a shows that the distribution of observed v1 selection frequencies $p_1(\hat{s}_n(x))$ becomes increasingly skewed as more annotators are used to estimate selection frequencies (i.e. as bias decreases).

- Figure 5b plots selection frequency-conditinoed classifier accuracy, $\mathbb{E}_{x_2 \sim \mathcal{D}_2}\left[f(x_2)|\hat{s}_n(x_2) = \frac{k}{n}\right]$ as a function of n. The plot indicates that when we use observed selection frequency in place of true selection frequency, we overestimate model accuracy on images with low selection frequency and underestimate accuracy on images with high selection frequency.

---

[12]We can compute the naïve estimator as long as we have enough images to reliably approximate the expectations. We assume this is the case in our study, since (a) we have 10^4 images and only 41 possible values of $\hat{s}_n(x)$ and (b) halving the number of images negligibly affects the value of the estimator.
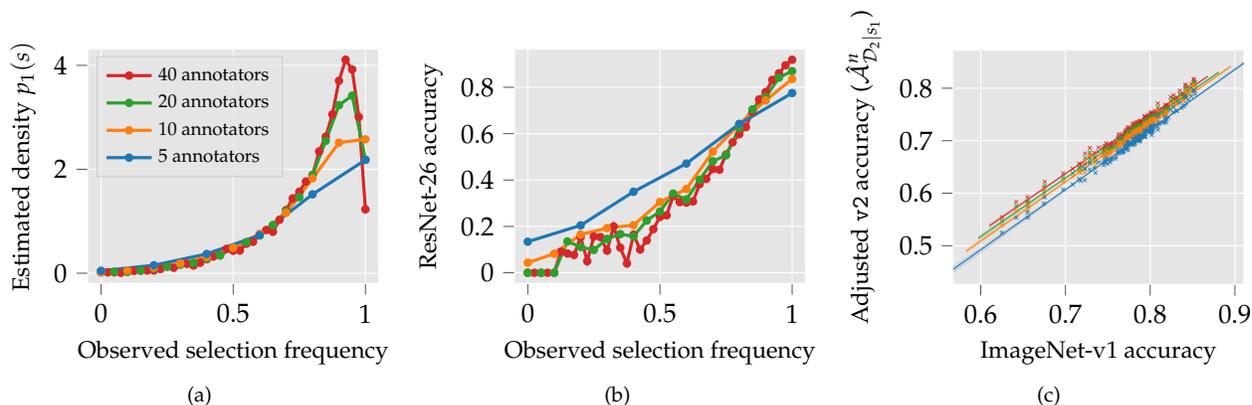
Figure 5: A series of graphs, all demonstrating bias in estimators that condition on selection frequency. **Left**: The estimated population density of selection frequencies, calculated naïvely from samples. For a given number of annotators per image n, the corresponding line in the graph has equally spaced points of the form $(k/n, \sum \mathbf{1}_{\hat{s}=k/n})$. **Middle**: Model accuracy of a ResNet-26 conditioned on selection frequency; once again, we naïvely using empirical selection frequency in place of true selection frequency for conditioning. Just as in the left-most graph, for a given n-annotator line, points at $x = k/n$ in the graph correspond to the accuracy on images with observed selection frequency k/n. **Right**: Adjusted v1 versus v2 accuracy plots, calculated for varying numbers of annotators per image (with adjusted accuracy computed using the naïve estimator of Section 5.1). Each point in the plot corresponds to a trained model.

- Combining these two sources of bias, Figure 5c shows that as we reduce bias by increasing n, the selection-adjusted v2 accuracy increases for every classifier.

It turns out that computing (4) using the 40 annotators per image that we collected in Section 3 already produces selection-adjusted v2 accuracies that are on average 6.0% higher than the initially observed v2 accuracy. Thus, despite still suffering from statistic matching bias, the naïve reduces the v1-v2 accuracy drop to 5.7%. In the following sections, we explore two different techniques for debiasing the naïve estimator and explaining more of the accuracy gap.

5.2 Estimating bias with the statistical jackknife

As a first attempt at correcting for the previously identified bias, we turn to a standard tool from classical statistics. The jackknife [Que49; Que56; Tuk58] is a nonparametric method for reducing the bias of finite-sample estimators. Under mild assumptions, the jackknife can reduce the bias of any estimator from $O(\frac{1}{n})$ to $O(\frac{1}{n^2})$. Concretely, the jackknife bias estimate for an n-sample estimator $\theta_n := \hat{\theta}(X_1, \ldots, X_n)$ is given by:

$$b_{jack}(\hat{\theta}_n) = (n-1) \cdot \left(\frac{1}{n} \sum_{i=1}^{n} \hat{\theta}_{n-1}^{(i)} - \hat{\theta}_n \right),$$

where $\quad \hat{\theta}_{n-1}^{(i)} = \hat{\theta}(X_1, \ldots, X_{i-1}, X_{i+1}, \ldots, X_n)$ is the ith leave-one-out estimate.

The statistical jackknife thus provides us with a technique for estimating and correcting for the bias in finite-sample estimates of the frequency-adjusted accuracy $\mathcal{A}_{\mathcal{D}_2|s_1}$.

Jackknifing the naïve estimator. As a first approach, we can apply the jackknife directly to the naïve estimator (cf. (4)). For the jackknife-corrected estimate to be meaningful, we have to show that the naïve estimator is a statistically consistent estimator of the true selection-adjusted accuracy (i.e., that $\lim_{n\to\infty} \hat{\mathcal{A}}_{\mathcal{D}_2|s_1}^n = \mathcal{A}_{\mathcal{D}_2|s_1}$). We prove this property in Appendix E, under the assumption that we can evaluate quantities of the form $p_i(\hat{s}_n(x) = s)$ exactly (in practice this assumption seems acceptable since the empirical variance of

(a) Adjustment via jackknife   (b) Adjustment via mixture model
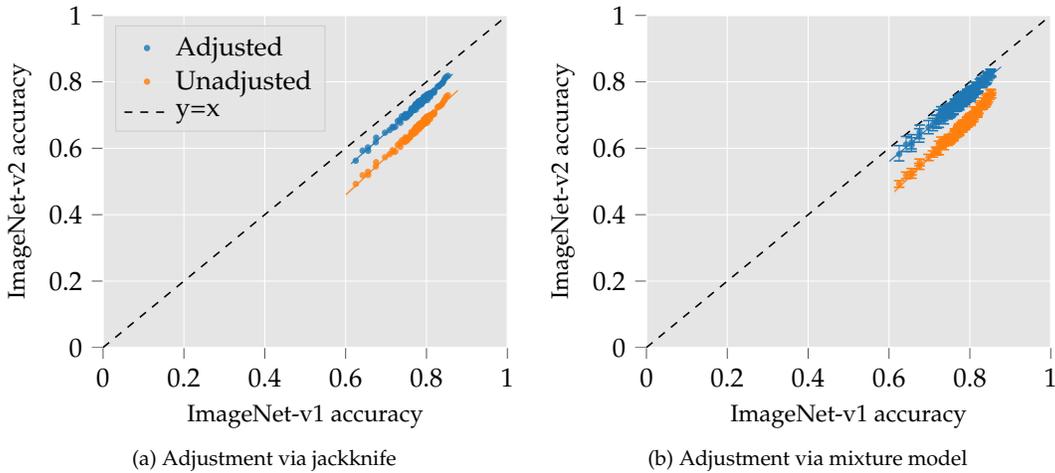
Figure 6: Accuracy on v1 versus v2 adjusted using the two techniques discussed in this section. On the left (respectively, right) we use the jackknife (parametric model) of Section 5.2 (5.3) to estimate adjusted accuracies for v2. The graphs confirm that the "true" gap in accuracy between v1 and v2 is indeed much smaller than the initially observed gap. Confidence intervals on the left are based on the jackknife standard error, and confidence intervals on the right are based on 400-sample 95% bootstrap confidence intervals

the estimator is small)[13]. Applying the jackknife to the naïve estimator reduces the adjusted accuracy gap further, from **5.7%** to **4.6%**.

Considerations and limitations of the jackknife approach. For the jackknife to perform reliably, we must have that (a) the leave-one-out estimators have low enough variance, and (b) the bias is an analytic function in $1/n$ that is dominated by the $\Theta(\frac{1}{n})$ term in its power series expansion. We address the first of these concerns by plotting jackknife confidence intervals (c.f. [ET94]) for our estimates. Consideration (b) carries a bit more weight: as shown in Appendix E.1, the n-sample naïve estimator has a roughly linear relationship in $1/n$, but not a perfect one—in particular, the estimator seems to increase at a rate slightly faster than $1/n$, suggesting that as a result, the jackknife still provides an underestimate of the selection-adjusted accuracy.

Another potential source of error is finite-sample error in measuring the expectations $\mathbb{E}_{x_2 \sim \mathcal{D}_2}[f(x_2)|\hat{s}_n(x_2)]$. If the number of workers per image n is taken to infinity while keeping the number of images constant, the observed selection frequencies will become too sparse to provide reliable estimates of the expectation. Thus we rely on our assumption from Section 4 that finite-sample error is negligible with respect to images (since we have 10^4 images and only 41 possible values of $\hat{s}$). Also, note that in principle, we could also use the jackknife to estimate the distributions of true selection frequencies $(p_i(s(x)))$ by estimating the bias in the statistic $\mathbb{E}[\mathbb{1}_{\hat{s}_n(x)=k}]$ for each value of k. However, this approach is too sample-expensive to yield reliable results.

In the next section, we present another approach to estimating the selection-adjusted accuracy, namely parametric modeling. Through a disjoint set of techniques and assumptions, we obtain estimates of $p_i(s(x))$ and $\mathcal{A}_{\mathcal{D}_2|s_1}$ that further corroborate the results so far.

5.3 Estimating bias with a parametric model

We now explore a more fine-grained approach to estimating the selection-adjusted accuracy of ImageNet-v2, namely explicit parametric modeling. Recall that the adjusted accuracy captures accuracy on ImageNet-

---

[13]It is also possible to prove consistency even without this assumption by assuming a fixed relationship between the number of images and the number of annotators per image, and at the cost of the proof's simplicity.

v2 reweighted to match ImageNet-v1 in terms of true selection frequency distribution, and is given by:

$$\mathcal{A}_{\mathcal{D}_2|s_1} = \int_s \mathbb{E}_{\mathcal{D}_2}[f(x)|s(x) = s] \cdot p_1(s) \, ds$$

$$= \int_{s \in [0,1]} p_2\left(f(x_2)|s(x_2) = s\right) \cdot p_1(s) \, ds \qquad (6)$$

In Sections 5.1 we computed a biased estimate of $\mathcal{A}_{\mathcal{D}_2|s_1}$ using observed selection frequencies $\hat{s}$ in place of true selection frequencies. Then, in 5.2 we corrected for the bias in this naïve estimator post-hoc using the statistical jackknife.

In constrast, the model-based approach tries to circumvent this bias altogether: we parameterize functions of the true selection frequency directly (i.e., $p_1(s)$ and $p_2(f(x) = 1|s(x) = s)$), then fit parameters that maximize the likelihood of the observed data while taking into account the noise model. For example, since the distribution of $\hat{s}_n(x)$ given $s(x])$ is the binomial distribution, we can write (and optimize) a closed-form expression for the likelihood of observing a given set of selection frequencies based on a parameterized true selection frequency distribution $p_1(s; \theta)$. We estimate selection-adjusted accuracy in two steps. First, we fit models for the true selection frequency distributions $p_1(s)$ and $p_2(s)$. Then, we use our estimate of $p_2(s)$ in conjunction with observed data to fit models for $p_2(f(x)|s(x) = s)$. Finally, we recover estimates for $\mathcal{A}_{\mathcal{D}_2|s_1}$ by numerically computing the integral in $\mathcal{A}_{\mathcal{D}_2|s_1}$ (c.f. (6)), plugging in the learned parametric estimates.

Fitting a model to $p_i(s(x))$. We model the $p_i(s(x))$ as members of a parameterized family of distribution $p_i(s(x); \theta)$ with true parameters $\theta_i^\star$. Then, for each dataset i, we model the observed selection frequencies as sampled from a compound distribution, in which one first samples $s \sim p_i(\cdot; \theta_i^\star)$, then observes $\hat{s} \sim$ Binom (n, s) (where n is the number of MTurk annotators).

To infer each $\theta_i^\star$, we use maximum likelihood estimation on the observed samples over the compound distribution. We opt to use mixtures of beta distributions as the family $p_i(\cdot; \theta)$ over which to optimize. Beta-mixture distributions are a fairly popular modeling choice for finite-support data [Ji+05; ML09; Lau+11], since beta distributions (a) have only two parameters; (b) induce smooth, continuous density functions with support $[0, 1]$; and (c) admit a closed-form likelihood when composed with a binomial random variable, by way of the *beta-binomial* distribution[14].

A mixture of k beta distributions is a $3k - 1$-parameter model; we denote the mixture coefficients by γ_j (with $\sum_j \gamma_j = 1$), and the parameters of each individual beta distribution in the mixture as (α_j, β_j). We use Expectation-Maximization (EM) [DLR77] to find, for each dataset $i \in \{1, 2\}$, the maximum likelihood mixture of $k = 3$ beta-binomial distributions for the observed $\hat{s}_n(x)$. The log-likelihood is given by:

$$\hat{\theta}_i := \beta^{(i)}, \alpha^{(i)}, \gamma^{(i)} = \underset{\beta, \alpha \in \mathbb{R}_+^3, \gamma \in \Delta^3}{\arg\max} \sum_{x \sim \mathcal{D}_i} \log\left(\sum_{j=1}^{3} \gamma_j \text{BetaBinom}\left(\alpha_j, \beta_j, N, x\right)\right),$$

where BetaBinom(α, β, N, x) is the density of the beta-binomial distribution parameterized by (α, β, N) and evaluated at x. We provide further detail on the fitting process for $p_i(s(x); \theta)$ in Appendix F, including pseudocode for the EM algorithm. We plot the resulting fitted distributions $p_i(s(x))$ in Figure 7. Our estimated $p_i(\hat{s}; \hat{\theta}_i)$ distributions continue the trend previously seen in Figure 5a, and show the extent to which our naïve 40-sample empirical estimates of $p_i(s(x))$ exhibit bias.

Fitting a model to $p_2(f(x) = 1|s(x) = s)$. Next, we consider accuracy conditioned on selection frequency:

$$g(s) = p_2(f(x_2) = 1|s(x_2) = s).$$

While introducing the naïve estimator (Section 5.1), we found that that estimating $g(s)$ using observed selection frequencies instead of true selection frequencies results in bias (Figure 5b). Under the parametric approach, we instead model $g(s)$ as a member of a parametric class (i.e., $g(s) = g(s; \omega)$), then account for noise in observed selection frequencies via the following identity:

$$p_2(f(x_2) = 1, \hat{s}_n(x_2) = \hat{s}) = \int_{s \in [0,1]} g(s) \cdot p(\hat{s}|s) \cdot p_2(s) \, ds. \qquad (7)$$

---

[14]`https://docs.scipy.org/doc/scipy/reference/generated/scipy.stats.betabinom.html`
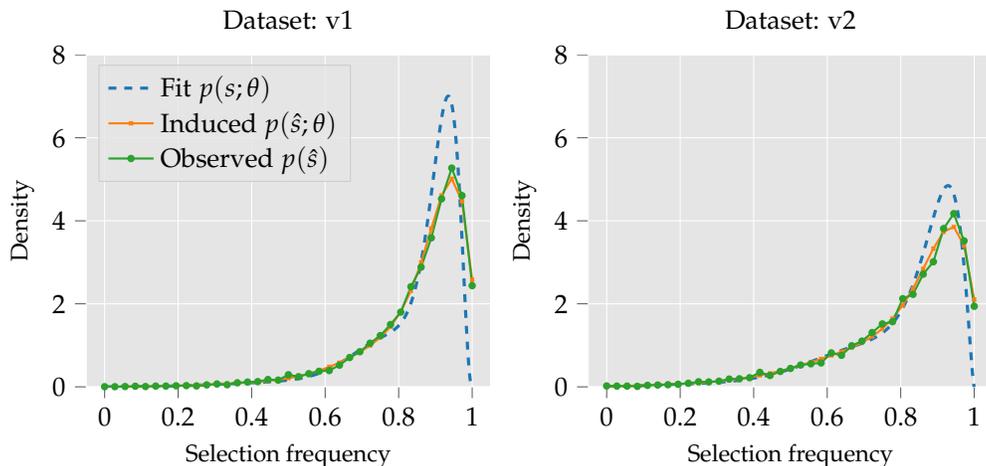
Figure 7: Our fit beta mixture models $p_i(s; \hat{\theta})$ for "true" selection frequency, the noisy selection frequency distribution they induce $p_i(\hat{s}; \hat{\theta})$, and the observed selection frequency $p_i(\hat{s})$. We observe that the fit $p_i(s; \hat{\theta})$ distributions place much more density on higher selection frequencies than naïvely estimating $p_i(s)$ from the observed $p_i(\hat{s})$.

Now, since g is an unknown but most likely smooth function from $[0, 1]$ to $[0, 1]$, a standard parameterized family $g(\cdot; \omega)$ to use is the class of cubic splines [De +78]. In particular, we try to find a $g(s; \omega)$ satisfying the relationship described by (7) by optimizing the following squared error objective:

$$L(\omega) := \sum_{\hat{s}} \left(p_2(f(x_2) = 1, \hat{s}_n(x_2) = \hat{s}) - \int_{s \in [0,1]} g(s; \omega) \cdot p(\hat{s}|s) \cdot p_2(s; \hat{\theta}_2) ds \right)^2. \tag{8}$$

We can compute the left hand term above from observed data, and the right-hand term is a function of the parametrized $g(s; \omega)$, the binomial mass function, and $p_2(s; \hat{\theta}_2)$, which we estimated in the first step. After discretizing the integral, (8) becomes a quadratic optimization problem [15] (since splines are linear in their parameters).

Results. Once we have estimated probability distributions $p_i(s(x); \theta_i)$ and the conditional classification function $g(s(x); \omega)$, we can compute an estimate of $\mathcal{A}_{\mathcal{D}_2|s_1}$ using Equation (6) and numerical integration. Figure 6b depicts various models' v1 and v2 accuracies both with and without the adjustment for selection frequency. Our estimate for the frequency-adjusted gap in accuracy averaged over all models is 3.6% ± 1.5%, around 30% of the original 11.7% ± 1.0% gap in accuracy.

Beyond accuracy gap, Recht et al. [Rec+19b] also studied the linear relationship between v2 accuracy and v1 accuracy while varying the classifier used—this is plotted by the blue dots in Figure 6b. This relationship is linear for our adjusted accuracies as well (cf. Figure 6b), however the slope we find is 1.01 ± 0.09 instead of 1.13 ± 0.05.

Considerations and limitations. Error in parametric modeling generally stems from two sources: finite-sample error and model misspecification. These sources of error affect all parametric models, but we take various precautions to mitigate their impact on our estimates.

To assess our finite-sample error, we give 95% bootstrapped confidence intervals (details are in Appendix F), which are displayed as error bars in Figure 6b. We also ensure that our results are not sensitive to the number of annotators used to fit the parametric models (cf. Appendix F).

---

[15]Note the choice of least-squares loss here is entirely for convenience—in principle one could maximize the likelihood of $p_2(f(x_2) = 1, \hat{s}_n(x_2) = \hat{s})$ instead at the cost of algorithmic simplicity and efficiency

As with any modeling decision, our choices of model classes might not fully capture the ground-truth generative process, and thus may be a source of error. We account for this as much as possible by demonstrating the robustness of our results to varying the number of free parameters (cf. Appendix F).

6 Related Work

Researchers have developed a cornucopia of datasets (e.g. [LeC98; Kri09; Rus+15; Zho+17] in computer vision) for benchmarking the generalization performance of supervised learning algorithms. The recent rapid pace of progress has drawn interest to the question of verifying that the progress made by high-performing models on standard dataset actually corresponds to progress on the underlying task, rather than on the dataset itself. To this end, previous work has characterized cross-dataset generalization on similar tasks [TE11] by measuring "dataset bias." Prior work has also explored the impact of synthetic perturbations on generalization, such as adversarial examples [KGB16; Tsi+19; Ily+19; Su+18] or various other corruption robustness measures [HD19; Kan+19].

In terms of measuring *same*-distribution generalization beyond test set performance, a number of works have emerged around evaluating performance on newly reproduced test sets, including works focusing on ImageNet [Rec+19b], CIFAR [Rec+19b], and MNIST [YB19]. By and large, these works report accuracy drops due to distribution shift, and claim that there is no adaptive overfitting.

Adaptively reusing data is a studied source of both overfitting in machine learning and, more generally, false discovery in the sciences [GL14]. In machine learning, work has gone into theoretically characterizing and counteracting the effects of adaptive overfitting [BH15; Dwo+15; Roe+19]. In computer vision in particular, Recht et al. [Rec+19b] and Mania et al. [Man+19] respectively give evidence that there is no adaptive overfitting, and that model similarity reduces adaptive overfitting.

A common phenomena noted in ecology is overdispersion [Gre83], a form of sampling bias that can emerge when one does not accurately model the underlying distribution generating samples. The bias we discuss can be framed as an instantiation of this problem.

7 Discussion and Conclusions

Dataset replication pipelines can introduce unforeseen, often unintuitive statistical biases. In the case of ImageNet-v2, even using unbiased estimates of image selection frequency in the data generation pipeline results in a significant statistical bias, and ultimately turns out to account for a large portion of the observed accuracy drop. Our findings give rise to the following considerations.

7.1 Remaining accuracy gap and unmodeled bias

Worker heterogeniety. Our study focuses on bias stemming from the fact that for a given image x one never observes $s(x)$ but rather $\hat{s}_n(x) = \text{Binom}(n, s(x))$. There is another source of bias due to noise that we do not model here, namely variance in the MTurk annotator population. Specifically, some annotators are more likely in general to select or reject independently of what image-label pair they are being shown. This unmodeled variance likely translates to unmodeled bias, suggesting that more of the gap might be explained by taking worker heterogeniety into account.

Task shift bias. At the time of the original ImageNet experiment, workers judged image-label pairs by some abstract set of criteria C_1. Suppose that at the time of the ImageNet-v2 experiment several years later, annotators judged image-label pairs based on an overlapping but non-identical set of criteria C_2. Ideally, we should not care about differences between C_1 and C_2—indeed, one of the goals of dataset replication is to test robustness to such benign distribution shifts. The source of the bias lies in the iterated nature of the filtering experiment. In particular, after both the original experiment and the replication, images in ImageNet-v1 now meet both C_1 and C_2. On the other hand, images in ImageNet-v2 only meet criteria C_2, and may be judged to have low selection frequency under C_1—we would thus expect models to perform better on ImageNet-v1 images due to their increased qualifications. Although this may contribute towards

the remaining accuracy gap, this type of bias is difficult to study or correct for without more knowledge of both experiments.

Other sources of error.   The remaining error unexplained by bias in data collection could come from one of the gap sources listed in Section 4, i.e., finite sample error, or distribution shift and adaptive overfitting. Quantifying the potential contribution of the individual terms in the remaining gap will require more experimentation and future work.

7.2   Adaptive overfitting and distribution shift

Identifying sources of distribution shift.   A longstanding goal in computer vision is to develop models that are less prone to failure under small distributional shifts. A step in the journey towards this goal is precisely characterizing the kinds of distribution shifts under which models fail—examples include rotations and translations of natural images Engstrom et al. [Eng+19], or corrupted natural images [HD19]. Our findings imply that the drop may be attributable to differences in selection frequency distribution, corroborating observations by Recht et al. [Rec+19b] that models are sensitive to selection frequency. Differences in selection frequency distribution present another distribution shift to study in depth.

Implications for adaptive overfitting.   A conclusion drawn from the ImageNet-v2 dataset replication is that the slope of best fit line for v1 versus v2 accuracy is significantly larger than one. This means that for every point of progress made on ImageNet, 1.1 points are made on ImageNet-v2, providing another point of evidence towards the absence of adaptive overfitting. However, after correcting for the bias in the v2 sampling process, v1 versus v2 model accuracies still exhibit a linear fit, but with a slope that is not conclusively bounded away from 1 (i.e., slopes are within 95% confidence intervals of 1, c.f. Section 5.3).

Detecting and avoiding bias in dataset replication.   More broadly, our analysis identifies statistical modeling of the data collection pipeline as a useful tool for dataset replication. Indeed, characterizing the ImageNet and ImageNet-v2 generative processes and isolating them in a a simple theoretical model allowed for the discovery and correction of a source of bias in the dataset replication process.

Acknowledgements

References

[Bel73]   David S Bell. "The experimental reproduction of amphetamine psychosis". In: *Archives of General Psychiatry* 29.1 (1973), pp. 35–40.

[BH15]   Avrim Blum and Moritz Hardt. "The ladder: A reliable leaderboard for machine learning competitions". In: *arXiv preprint arXiv:1502.04585* (2015).

[BKG11]   Michael Buhrmester, Tracy Kwang, and Samuel Gosling. "Amazon's Mechanical Turk: A New Source of Inexpensive, Yet High-Quality Data?" In: *Perspectives on Psychological Science*. 2011.

[De +78]   Carl De Boor, Carl De Boor, Etats-Unis Mathématicien, Carl De Boor, and Carl De Boor. *A practical guide to splines*. Vol. 27. springer-verlag New York, 1978.

[Den+09]   Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. "Imagenet: A large-scale hierarchical image database". In: *computer vision and pattern recognition (CVPR)*. 2009.

[DLR77]   Arthur P Dempster, Nan M Laird, and Donald B Rubin. "Maximum likelihood from incomplete data via the EM algorithm". In: *Journal of the Royal Statistical Society: Series B (Methodological)* 39.1 (1977), pp. 1–22.

[Dwo+15]   Cynthia Dwork, Vitaly Feldman, Moritz Hardt, Toni Pitassi, Omer Reingold, and Aaron Roth. "Generalization in adaptive data analysis and holdout reuse". In: *Advances in Neural Information Processing Systems*. 2015, pp. 2350–2358.

[Eng+19]   Logan Engstrom, Brandon Tran, Dimitris Tsipras, Ludwig Schmidt, and Aleksander Madry. "Exploring the Landscape of Spatial Robustness". In: *International Conference on Machine Learning (ICML)*. 2019.

[ET94]   Bradley Efron and Robert Tibshirani. *An Introduction to the Bootstrap*. Chapman and Hall, 1994.

[GL14]   Andrew Gelman and Eric Loken. "The statistical crisis in science: data-dependent analysis–a" garden of forking paths"–explains why many statistically significant comparisons don't hold up". In: *American scientist* 102.6 (2014), pp. 460–466.

[Gre83]   Peter Greig-Smith. *Quantitative plant ecology*. Vol. 9. Univ of California Press, 1983.

[HD19]   Dan Hendrycks and Thomas G. Dietterich. "Benchmarking Neural Network Robustness to Common Corruptions and Surface Variations". In: *International Conference on Learning Representations (ICLR)*. 2019.

[Ily+19]   Andrew Ilyas, Shibani Santurkar, Logan Engstrom, Brandon Tran, and Aleksander Madry. "Adversarial Examples Are Not Bugs, They Are Features". In: *Neural Information Processing Systems (NeurIPS)*. 2019.

[IR13]   Kosuke Imai and Marc Ratkovic. "Covariate balancing propensity score". In: *Journal of the Royal Statistical Society*. 2013.

[Ji+05]   Yuan Ji, Chunlei Wu, Ping Liu, Jing Wang, and Kevin R. Coombes. "Applications of beta-mixture models in bioinformatics". In: *Bioinformatics*. Vol. 21. 9. 2005, pp. 2118–2122.

[Kan+19]   Daniel Kang, Yi Sun, Dan Hendrycks, Tom Brown, and Jacob Steinhardt. "Testing Robustness Against Unforeseen Adversaries". In: *ArXiv preprint arxiv:1908.08016*. 2019.

[KGB16]   Alexey Kurakin, Ian Goodfellow, and Samy Bengio. "Adversarial examples in the physical world". In: *arXiv preprint arXiv:1607.02533* (2016).

[Kri09]   Alex Krizhevsky. "Learning Multiple Layers of Features from Tiny Images". In: *Technical report*. 2009.

[Lau+11]   Kirsti Laurila, Bodil Oster, Claus L Andersen, Philippe Lamy, Torben Orntoft, Olli Yli-Harja, and Carsten Wiuf. "A beta-mixture model for dimensionality reduction, sample classification and analysis". In: *BMC bioinformatics*. 2011.

[LDS20]   Zhijing Li, Christopher De Sa, and Adrian Sampson. "Optimizing JPEG Quantization For Classification Networks". In: *ArXiv preprint arxiv:2003.02874*. 2020.

[LeC98]   Yann LeCun. "The MNIST database of handwritten digits". In: *Technical report*. 1998.

[Man+19]   Horia Mania, John Miller, Ludwig Schmidt, Moritz Hardt, and Benjamin Recht. "Model similarity mitigates test set overuse". In: *Advances in Neural Information Processing Systems*. 2019, pp. 9993–10002.

[ML09]   Zhanyu Ma and Arne Leijon. "Beta mixture models and the application to image classification". In: *International Conference on Image Processing*. Nov. 2009, pp. 2045–2048.

[MW09]   Winter Mason and Duncan Watts. "Financial Incentives and the "Performance of Crowds"". In: *KDD Human Computation Workshop*. 2009.

[PCI10]   Gabriele Paolacci, Jesse Chandler, and Panagiotis Ipeirotis. "Running experiments on Amazon Mechanical Turk". In: *Judgement and Decision Making*. 2010.

[PVA13]   Eyal Peer, Joachim Vosgerau, and Alessandro Acquisti. "Reputation as a sufficient condition for data quality on Amazon Mechanical Turk". In: *Behavior Research Methods*. 2013.

[Que49]   Maurice Quenouille. "Problems in Plane Sampling". In: *Annals of Mathematical Statistics*. 1949.

[Que56]   Maurice Quenouille. "Notes on Bias in Estimation". In: *Biometrika*. 1956.

[Rec+19a] Benjamin Recht, Rebecca Roelofs, Ludwig Schmidt, and Vaishaal Shankar. "Do CIFAR-10 Classifiers Generalize to CIFAR-10?" In: *International Conference on Machine Learning (ICML)*. 2019.

[Rec+19b] Benjamin Recht, Rebecca Roelofs, Ludwig Schmidt, and Vaishaal Shankar. "Do ImageNet Classifiers Generalize to ImageNet?" In: *International Conference on Machine Learning (ICML)*. 2019.

[Reu03]   Juha Reunanen. "Overfitting in Making Comparisons Between Variable Selection Methods". In: *Journal of Machine Learning Research*. Vol. 3. 2003, pp. 1371–1382.

[RFR08]   R. Bharat Rao, Glenn Fung, and Romer Rosales. "On the Dangers of Cross-Validation. An Experimental Evaluation". In: *SIAM International Conference on Data Mining (ICDM)*. 2008, pp. 588–596.

[RM19]   Tiago Ramalho and Miguel Miranda. "Density estimation in representation space to predict model uncertainty". In: *arXiv preprint arXiv:1908.07235* (2019).

[Roe+19]   Rebecca Roelofs, Vaishaal Shankar, Benjamin Recht, Sara Fridovich-Keil, Moritz Hardt, John Miller, and Ludwig Schmidt. "A Meta-Analysis of Overfitting in Machine Learning". In: *Advances in Neural Information Processing Systems*. 2019, pp. 9175–9185.

[RSP19]   Tiago Ramalho, Thierry Sousbie, and Stefano Peluchetti. "An empirical study of pretrained representations for few-shot classification". In: *arXiv preprint arXiv:1910.01319* (2019).

[Rus+15]   Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei. "ImageNet Large Scale Visual Recognition Challenge". In: *International Journal of Computer Vision (IJCV)*. 2015.

[SP81]   Howard Schuman and Stanley Presser. *Questions and Answers in Attitude Surveys*. 1981.

[Su+18]   Dong Su, Huan Zhang, Hongge Chen, Jinfeng Yi, Pin-Yu Chen, and Yupeng Gao. "Is Robustness the Cost of Accuracy? A Comprehensive Study on the Robustness of 18 Deep Image Classification Models". In: *European Conference on Computer Vision (ECCV)*. 2018.

[Tao+20]   Rohan Taori, Achal Dave, Vaishaal Shankar, Nicholas Carlini, Benjamin Recht, and Ludwig Schmidt. *When Robustness Doesn't Promote Robustness: Synthetic vs. Natural Distribution Shifts on ImageNet*. 2020. URL: https://openreview.net/forum?id=HyxPIyrFvH.

[TE11]   Antonio Torralba and Alexei A Efros. "Unbiased look at dataset bias". In: *CVPR 2011*. 2011.

[Tsi+19]   Dimitris Tsipras, Shibani Santurkar, Logan Engstrom, Alexander Turner, and Aleksander Madry. "Robustness May Be at Odds with Accuracy". In: *International Conference on Learning Representations (ICLR)*. 2019.

[Tuk58]   John Tukey. "Bias and confidence in not quite large samples". In: *Annals of Mathematical Statistics*. 1958.

[YB19]   Chhavi Yadav and Leon Bottou. "Cold Case: The Lost MNIST Digits". In: *Neural Information Processing Systems (NeurIPS)*. 2019.

[Zho+17]   Bolei Zhou, Agata Lapedriza, Aditya Khosla, Aude Oliva, and Antonio Torralba. "Places: A 10 million image database for scene recognition". In: *IEEE transactions on pattern analysis and machine intelligence* (2017).

A Understanding Selection Frequency

Understanding Selection Frequencies In Figure 8 we randomly sample images while varying selection frequency. Here, the straightforwardness of identifying images correlates with increasing selection frequency (e.g. all the 36/36 selection frequency images clearly identify with their corresponding class, while some of the 0/36 selection frequency images appear to be mislabeled).
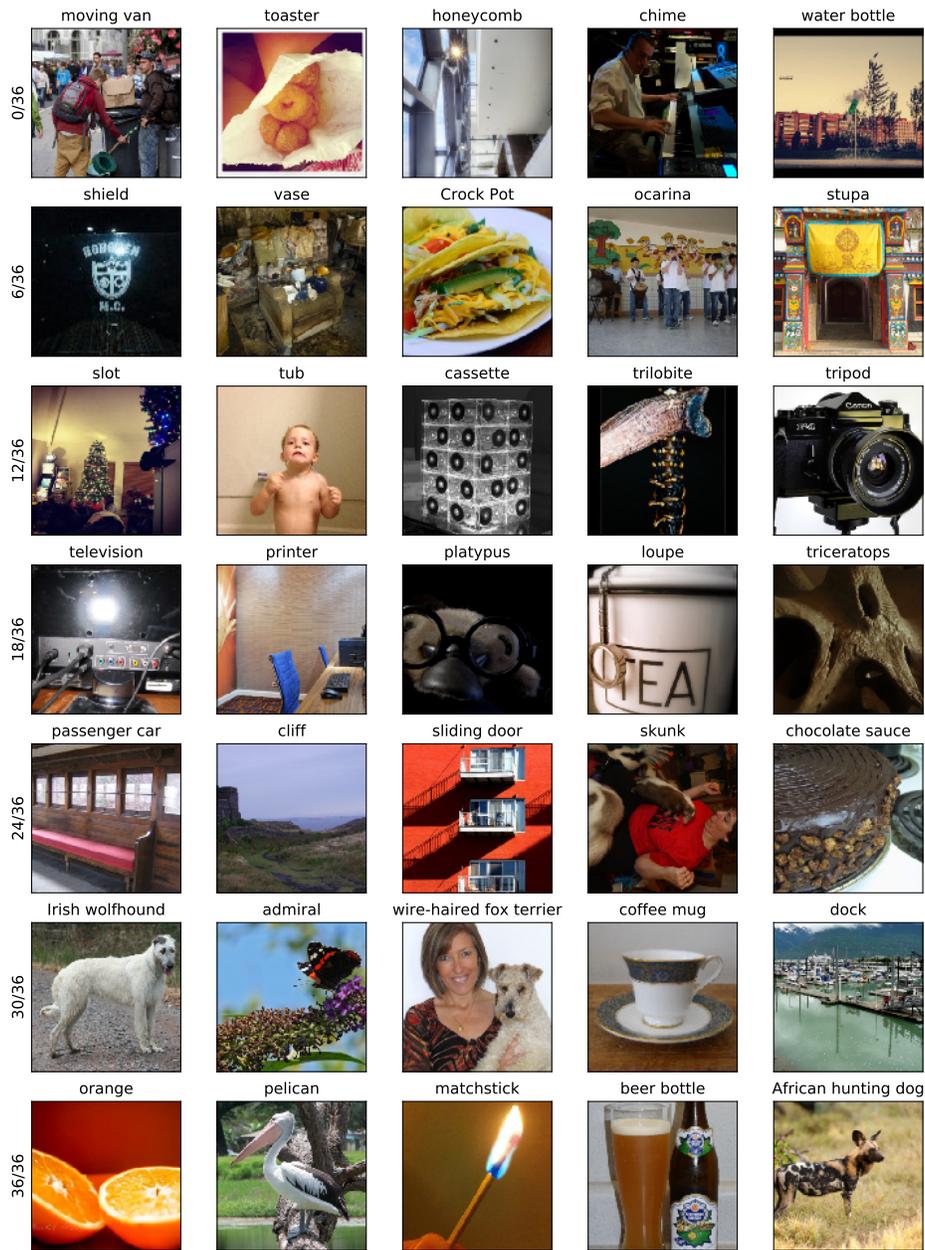
Figure 8: Randomly drawn images from `v1`, varying selection frequency.

B Experimental Setup

Here, we provide more detail of the experimental setup. We first lay out the setup of our Mechanical Turk experiments for remeasuring selection frequency (B.1), and highlight the subtle differences between our setup and that of Recht et al. [Rec+19b] (B.2). In Appendix D we discuss our analysis of the original data and algorithm of Recht et al. [Rec+19b] showing the existence of bias in that setting.

B.1 Selection frequency remeasuring experiment

In Section 3, we replicate the ImageNet experiment to remeasure selection frequencies for the ImageNet v1 and v2 datasets. We present annotators with a grids of 48 images along with an ImageNet class. The annotators are also provided with the WordNet synsets for the ImageNet class being queried, along with a Wikipedia link and asked to selected all images containing instances of that object (ignoring clutter as per the original dataset creation process). The 48 images in each grid consist of: (a) 10 ImageNet v1 validation set images from that class, (b) 10 ImageNet v2 validation set images from that class, (c) 22 related Flickr images scraped from Flickr (using the exact script and queries described in Recht et al. [Rec+19b]) and (c) 6 negative control images (three corresponding to randomly chosen labels, and 3 corresponding to the "nearest" label to the true label in terms of WordNet path similarity).

We implemented our setup by modifying the code made publicly available by Recht et al. [Rec+19b] [16]. A screenshot of our interface appears in Figure 10. Each such grid of images is shown to 40 annotators. For each image-class pair, we can then compute the "selection frequency" based on how often it was selected by the annotators.

Deployment Details.   There are a number of deployment details that could cause variations in results. We compensated MTurk workers with $0.23 per assignment (i.e., each completed grid), which we calibrated to pay a rate of at least $9/hr for most workers. To collect 40 separate MTurk annotations for each submitted grid of images, we obtained 10 annotations on 4 different dates and times, all within the span of a single week. We placed qualification requirements on the workers allowed to complete assignments. Specifically, we filtered for workers that (a) agree to view adult content (as some ImageNet images have content like nudity or gore) and (b) have a larger than 95% assignment approval rate (as to ensure the quality of the results).

Controls.   All of the results presented in this work were run on a "clean" and "raw" version of our data, i.e., without and with data cleaning respectively. We find that the inclusion of data cleaning makes the observed gap between v1 and v2 slightly larger but otherwise does not have a significant effect on results.

Our data cleaning process is as follows: a given batch is "flagged" if: (a) there are less than 6 selected images out of the total 48, or (b) more than one of the negative controls was selected. We only omit data, however, from workers whose batches were rejected at a rate of 30% or higher (e.g., if an annotator completed 30 batches, but more than 10 of them are flagged to be low-quality, then all of the annotator's data is omitted). Finally, to make computing of the statistics easier, we evened out the number of annotators per image to equal the minimum number of remaining assignments per image, which was 36 (compared to 40 originally) by randomly discarding annotations. In total, the entire process corresponds to discarding 10% of the annotations.

B.2 Comparison to the original setup

Recht et al. [Rec+19b] measure the average selection frequency of v1 to be 0.71, whereas our experiment measures the average v1 selection frequency to be 0.85. While our experiments were modeled closely after that of Recht et al. [Rec+19b] (and in fact use the same core codebase to minimize discrepancies in task presentation/inferace), we made a few changes to the setup to ensure high data quality. We hypothesize that these changes, discussed below, are what result in the discrepancy between the measured average selection frequencies. However, since these changes are applied at the task level and annotators are not told

---

[16]https://github.com/modestyachts/ImageNetV2

which dataset each image is sourced from, we find it unlikely that these changes would affect annotations for one dataset more than the other. Furthermore, in Appendix D, we demonstrate that the bias identified in this paper can be found even using the original data collected by Recht et al. [Rec+19b].

Worker pay and qualifications. In our experiment, we paid annotators 20 cents per set of 48 images completed—this was informed by the average time taken to complete a batch, and was calibrated so that the task paid approximately 12 dollars per hour. Conversely, the original experiment of Recht et al. [Rec+19b] pay 10 cents per batch. Although worker pay usually has only a mild impact on worker reliability on MTurk [MW09; BKG11], higher worker pay has been recognized as a tool to boost participation rates [BKG11] and requester reputation for future experiments [PCI10].

Perhaps the most important modification made was our inclusion of worker qualifications, which only allow annotators who have had 95% or more of previous tasks accepted to participate in our task. Prior work has shown that without these worker qualifications, crowdsourced data tends to be of significantly worse quality. For example, Peer, Vosgerau, and Acquisti [PVA13] report that 2.6% of workers with the "95% accepted" qualification failed an "attention-check test," compared to 33.9% of workers without qualifications[17]. We should therefore expect a significant increase in annotation reliability (and so in turn some discrepancy) from using worker qualifications.

Makeup of each batch. Another difference between the two experiments is that in our experiment, each batch of images contains 10 images from ImageNet-v1, and 10 images from ImageNet-v2, in order to ensure that we could obtain 40 annotations for each v1 and v2 image while keeping to a reasonable budget constraint. The experiment of Recht et al. [Rec+19b] uses only three images from ImageNet-v1 per batch (and a variable number of ImageNet-v2 images, since the dataset was not yet realized). Thus, the grids presented in our experiment contain images that are on average more likely (*a priori*) to be selected. This could in part contribute to the higher average selection frequency that we observe (though again, we would expect this effect to apply to both datasets and thus preserve the observed selection frequency gap).

Randomization. Response-order bias is a well-documented phenomenon in literature on crowdsourcing (e.g. [SP81]), although its effects in the domain of image selection are not well-understood yet. In our experiment, we randomize the order of the images per batch per worker (i.e., we used JavaScript to randomize the image order on page load) to mitigate the potential effects of this bias. In the prior experiment, however, the image order is deterministic, and thus the study may have response-order effects.

Worker duplicates. Due to the mechanism by which images were distributed to assignments in the study of Recht et al. [Rec+19b], 5.6% of the annotations are duplicated (i.e., 5.6% of the [worker, image] pairs collected are non-unique), with approximately 3% of the annotations being redundant (unlike the preceding number, this fraction does not count the "original copy" of each non-unique pair). A histogram of the number of times a single worker labeled a single image is shown in Figure 9. Since duplicate workers violate sample indpendence and can skew measured selection frequencies for some images, in our study we ensure that no worker labels the same image more than once.

Data cleaning and controls. Our study also differs in having built-in mechanisms for data cleaning (as discussed in the last section), allowing us to run all of our experiment on the "cleaned" and "raw" versions of our data. These results tend to not be substantially different (for the cleaned data, the selection frequency gap we measure between v1 and v2 slightly increases from 4.5% to 4.6%). Possible reasons for this similarity between cleaned and raw results include any of the quality control protocols outlined in this section.

---

[17]Attention-check tests are a series of three attention-check questions (ACQs). ACQs are questions with right/wrong answers unrelated to the task meant to gauge an annotator's attentiveness, e.g. "Have you ever had a fatal heart attack?". In the Peer, Vosgerau, and Acquisti [PVA13] study, 16.4% of unqualified workers reported that they had suffered a *fatal* heart attack, compared to 0.4% of qualified workers.
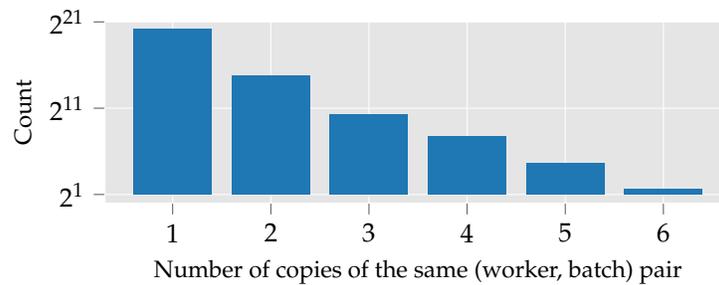
Figure 9: A histogram showing the existence of duplicate (worker, batch) pairs present in the original collected data. Each point in the histogram is a unique (worker, batch) pair, and the x axis corresponds to the number of times that pair is observed in the dataset.

Figure 10: Screenshot of a sample grid to measure selection frequencies (Section 3). Annotators are given a grid of 48 images for a specific label and asked to select all images that contain that label. The interface and instructions are based on Recht et al. [Rec+19b].

C Theoretical Results

In this section we show the series of calculations used in Section 2 to attain the result in Equation 1, i.e. the bias incurred by a matching procedure in the toy model.

Recall that in our setup we have $p_{flickr}(s)$ and $p_1(s)$ given by $\text{Beta}(\alpha, \beta)$ and $\text{Beta}(\alpha + 1, \beta)$ respectively, and that $\hat{s}(x)$ is given by first sampling $s \sim p_i(s)$ then sampling n Bernoulli trials with success probability s. We noted in Section 2 that the distribution of $s(x)$ induced by matching $p_{flickr}(s)$ and $p_1(s)$ based on samples of $\hat{s}(x)$ is given by:

$$p_{flickr}(s(x)) \cdot \mathbb{P}(\text{x is accepted}|s(x)) = p_{flickr}(s(x)) \cdot \int_{\hat{s}} p(\hat{s}|s)\mathbb{P}(\text{x is accepted}|\hat{s}(x))$$

$$= p_{flickr}(s(x)) \cdot \int_{\hat{s}} p(\hat{s}|s)\frac{p_1(\hat{s}(x))}{p_{flickr}(\hat{s}(x))}$$

Now, note that by construction, $\hat{s}(x)$ is distributed according to the beta-binomial distribution[18], and thus (a) has support $\{0, \ldots, n\}$; and (b) induces the following closed-form probability mass function for $p_{flickr}(\hat{s})$ ($p_1(\hat{s})$ can be written analogously, with $\alpha + 1$ replacing α) :

$$p_{flickr}(\hat{s}(x) = k) = \binom{n}{k}\frac{B(k + \alpha, n - k + \beta)}{B(\alpha, \beta)},$$

where

$$B(\alpha, \beta) = \frac{\Gamma(\alpha)\Gamma(\beta)}{\Gamma(\alpha + \beta)},$$

and Γ is the Gamma function—for simplicity we will assume that $\alpha, \beta \in \mathbb{N}$ and so $\Gamma(x) = (x - 1)!$. Thus, returning to the full induced density:

$$p_{flickr}(s(x)) \cdot \int_{\hat{s}} p(\hat{s}|s)\frac{p_1(\hat{s}(x))}{p_{flickr}(\hat{s}(x))}$$

$$= p_{flickr}(s(x)) \cdot \sum_{k=0}^{n} p(\hat{s}|s)\frac{p_1(\hat{s}(x))}{p_{flickr}(\hat{s}(x))}$$

$$= p_{flickr}(s(x)) \cdot \sum_{k=0}^{n}\left[\binom{n}{k}s^k(1-s)^{n-k}\right]\frac{\left[\frac{\binom{n}{k}B(k+\alpha+1, n-k+\beta)}{B(\alpha+1, \beta)}\right]}{\left[\frac{\binom{n}{k}B(k+\alpha, n-k+\beta)}{B(\alpha, \beta)}\right]}$$

$$= \left[\frac{s^{\alpha-1} \cdot (1-s)^{\beta-1}}{B(\alpha, \beta)}\right] \cdot \sum_{k=0}^{n}\left[\binom{n}{k}s^k(1-s)^{n-k}\right]\frac{B(k+\alpha+1, n-k+\beta)}{B(k+\alpha, n-k+\beta)} \cdot \frac{B(\alpha, \beta)}{B(\alpha+1, \beta)}$$

$$= \left[\frac{s^{\alpha-1} \cdot (1-s)^{\beta-1}}{B(\alpha+1, \beta)}\right] \cdot \sum_{k=0}^{n}\left[\binom{n}{k}s^k(1-s)^{n-k}\right]\frac{B(k+\alpha+1, n-k+\beta)}{B(k+\alpha, n-k+\beta)}$$

Now in general, note that

$$\frac{B(x+1, y)}{B(x, y)} = \frac{\frac{\Gamma(x+1)\Gamma(y)}{\Gamma(x+y+1)}}{\frac{\Gamma(x)\Gamma(y)}{\Gamma(x+y)}} = \frac{\Gamma(x+1)}{\Gamma(x)} \cdot \frac{\Gamma(x+y)}{\Gamma(x+y+1)} = \frac{x}{x+y} \qquad \text{for } x, y \in \mathbb{N}.$$

---

[18]https://en.wikipedia.org/wiki/Beta-binomial_distribution

Applying this identity to the above and continuing to simplify:

$$= \left[\frac{s^{\alpha-1} \cdot (1-s)^{\beta-1}}{B(\alpha+1,\beta)} \right] \cdot \sum_{k=0}^{n} \left[\binom{n}{k} s^k (1-s)^{n-k} \right] \frac{k+\alpha}{n+\alpha+\beta}$$

$$= \left[\frac{s^{\alpha-1} \cdot (1-s)^{\beta-1}}{B(\alpha+1,\beta)} \right] \cdot \mathbb{E}_{k \sim \text{Binomial}(n,s)} \left[\frac{k+\alpha}{n+\alpha+\beta} \right]$$

$$= \left[\frac{s^{\alpha-1} \cdot (1-s)^{\beta-1}}{B(\alpha+1,\beta)} \right] \cdot \frac{n \cdot s + \alpha}{n+\alpha+\beta}$$

$$= \frac{n}{n+\alpha+\beta} \cdot \frac{s^{(\alpha+1)-1} \cdot (1-s)^{\beta-1}}{B(\alpha+1,\beta)} + \frac{\alpha}{n+\alpha+\beta} \cdot \frac{s^{\alpha-1} \cdot (1-s)^{\beta-1}}{B(\alpha+1,\beta)}$$

$$= \frac{n}{n+\alpha+\beta} \cdot \frac{s^{(\alpha+1)-1} \cdot (1-s)^{\beta-1}}{B(\alpha+1,\beta)} + \frac{\alpha}{n+\alpha+\beta} \cdot \left(\frac{\alpha+\beta}{\alpha} \cdot \frac{B(\alpha+1,\beta)}{B(\alpha,\beta)} \right) \cdot \frac{s^{\alpha-1} \cdot (1-s)^{\beta-1}}{B(\alpha+1,\beta)}$$

$$= \frac{n}{n+\alpha+\beta} \cdot \text{Beta}(\alpha+1,\beta) + \frac{\alpha+\beta}{n+\alpha+\beta} \cdot \text{Beta}(\alpha,\beta)$$

which matches precisely the result shown in Section 2.

D Analysis of Original Data

In Section 3, we remeasured selection frequencies using a new Mechanical Turk experiment. Here we set out to verify the existence of the hypothesized bias in the original collected data.

We reimplemented the sampling component of the algorithm exactly as described in [Rec+19b] and the corresponding code release, using the `pandas` Python package. The source code is available in our code release, along with a serialized version of the data collected by Recht et al. [Rec+19b][19]. As a sanity check, we verified that all of the results hold using the exact code published by Recht et al. [Rec+19b][20]

D.1 Sampled dataset accuracy increases with more workers

We begin by showing that the accuracy we observe on ImageNet-v2 depends on the number of workers used to sample the dataset. We gradually decrease the number of workers n used in computing observed selection frequencies to study the effect of noise on statistic matching. We find that model accuracy on the resulting replicated dataset degrades as n decreases. For example, the accuracy gap from v1 to the replication increases from 12% when $n = 10$, to 14% when $n = 5$. This is consistent with our model of statistic matching bias: fewer annotators means noisier observed selection frequencies $\hat{s}_n(x)$, which in turn amplifies the effect of the bias, driving down model accuracies.

Methodology. Specifically, we use the frequency-adjusted accuracy introduced in Section 5.1, to estimate model accuracy on a version of the candidate pool reweighted to have the same selection frequency distribution as ImageNet-v1:

$$\hat{\mathcal{A}}^n_{\mathcal{D}_{flickr}|s_1} = \sum_{k=0}^{n} \mathbb{E}_{x_{flickr} \sim \mathcal{D}_{flickr}} \left[f(x_{flickr}) \Big| \hat{s}_n(x_{flickr}) = \frac{k}{n} \right] \cdot p_1 \left(\hat{s}_n(x_1) = \frac{k}{n} \right). \tag{9}$$

This estimator is analogous to the ImageNet-v2 selection process of Recht et al. [Rec+19b], but operates by reweighting the candidate pool rather than filtering it.

We plot this estimator in Figure 11, varying n from 5 to 10. We find that the gap between the adjusted accuracy and ImageNet accuracy shrinks as n grows, until shrinking to (and not plateauing at) 12.3% at $n = 10$. This behavior is predicted by statistic matching bias, and suggests that in the infinite-annotator limit the ImageNet-v2 accuracy is higher. (Ideally, we could estimate the infinite-annotator limit using the data of Recht et al. [Rec+19b], but 10 annotators is too few to get a reliable estimate.)
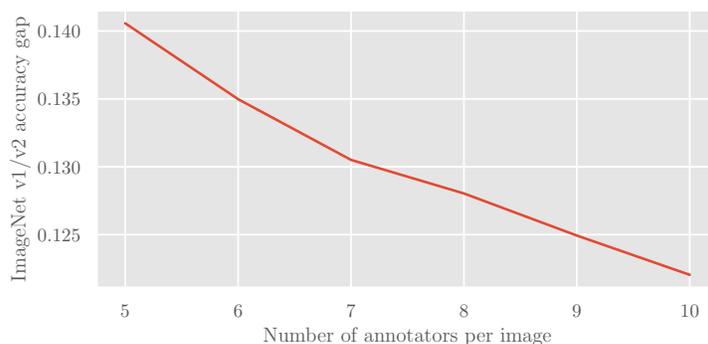
Figure 11: The frequency-adjusted accuracy gap between ImageNet-v1 and ImageNet-v2, using a varying number of annotators to estimate selection frequencies. The gap continually decreases, and does not plateau at 10 annotators. Bootstrapped 95% confidence intervals are shown (shaded).

---

[19]https://github.com/MadryLab/dataset-replication-analysis
[20]Since we only study the sampling component, we opt to rewrite a specialized script that has the benefit of being significantly shorter, simpler, and faster.

D.2 Measuring the selection frequency gap using held-out workers

Recall that most images in Recht et al. [Rec+19b] have (at least) 10 annotations per image, and that to get an unbiased estimate of the selection frequency we must "hold out" some annotations per image (e.g. we should not use the same annotations for both matching and measuring selection frequencies). To that end, we split the annotations for each candidate image into a "train" set and a "test" set. We then mimic the original v2 creation process using only the train set of annotations, and use the remaining test annotations for each image to obtain a held-out measurement of selection frequency.

Since Recht et al. [Rec+19b] collect 10 annotations for most candidate images, and since the original buckets used in the matching process are split by boundaries $\{0, 0.2, 0.4, 0.6, 0.8, 1\}$, we use 5 annotations[21] for each image in the train set and reserve the rest for independently measuring the held-out frequency.

The results of this experiment are given in Table 1. We repeat the experiment using both 5 and 10 annotators to estimate ImageNet-v1 selection frequencies. In both cases, the average selection frequency of the in-sample images overestimates the heldout (true) selection frequency by 2-3%, and the resulting replicated dataset has lower selection frequency than ImageNet-v1.

Table 1: Average in-sample and heldout selection frequencies for the experiment described in D.2—the top (bottom) table presents the result of using 10 (5) annotators per image to estimate ImageNet-v1 selection frequencies. We use five annotators per image to estimate selection frqeuencies, then use the filtering algorithm of Recht et al. [Rec+19b] to get a replicated dataset meant to match the selection frequency distribution of ImageNet-v1. The results show that (a) bias results in the average selection frequency of the new sample being *lower* than that of ImageNet-v1, and that (b) the bias is undetectable without heldout samples.

	ImageNet-v1	Sampled replication
Average selection frequency	0.71	0.71
Heldout selection frequency	0.71	0.69

	ImageNet-v1	Sampled replication
Average selection frequency	0.71	0.73
Heldout selection frequency	0.71	0.70

D.3 Source selection frequencies determine sampled dataset accuracy

We next explore how the choice of source distribution impacts the resulting sampled dataset. We use a setup similar to that of the last experiment, in which we use five workers for the selection process. Then, using four hold-out samples from each image, we create a new candidate data pool called Flickr-E (Flickr-Easy) by including only the images which at least two out of the four heldout workers selected.

We then perform 5-worker statistic matching, both from Flickr and from Flickr-E to ImageNet-v1. In the absence of bias, the source distribution should not affect the accuracy of the resulting classifier. In contrast, we find that the dataset replication obtained from Flickr-v2 has comparable (within 0.2%) average selection frequency, but significantly higher accuracy (by ~3%) than the replication obtained from the unfiltered candidate pool (62%). This discrepancy further corroborates the hypothesis that ImageNet-v2 accuracies are impacted by the statistical bias that we identify in this work.

---

[21]We choose 5 annotations specifically since, as in the 10-annotation case, images fall into the same relative locations in each bin—other choices of annotations per image are severely affected by binning effects.

E   Non-parametric Adjusted Accuracy Estimation

In Section 5 we explore various methods of estimating the adjusted accuracy $\mathcal{A}_{\mathcal{D}_2|s_1}$ from the observable $\hat{s}$ samples. Section 5.1 presents the *naïve estimator*, computed by using $\hat{s}$ directly in place of s in the formula for $\mathcal{A}_{\mathcal{D}_2|s_1}$. In Section 5.2, we show that using the statistical jackknife, we can estimate and account for the bias in the naïve estimator to better estimate the true adjusted accuracy. Here, we first justify the application of the jackknife in Section 5.2, by proving the consistency of the estimator and that bias is roughly linear in (and in fact underestimated by) $1/n$.

E.1   Justifying the use of the statistical jackknife

Recall that in Section 5, the quantity of interest is the following *adjusted accuracy*:

$$\mathcal{A}_{\mathcal{D}_2|s_1} = \int_s \mathbb{E}_{\mathcal{D}_2}[f(x)|s(x) = s] \cdot p_1(s)\, ds. \tag{10}$$

In Section 5.1, we introduced the following naïve estimator

$$\hat{\mathcal{A}}_{\mathcal{D}_2|s_1}^n = \sum_{k=0}^{n} \mathbb{E}_{x_2 \sim \mathcal{D}_2}\left[f(x_2)|\hat{s}_n(x_2) = \frac{k}{n}\right] \cdot p_1\left(\hat{s}_n(x_1) = \frac{k}{n}\right), \tag{11}$$

which evaluates to $\mathcal{A}_{\mathcal{D}_2|s_1}$ if $\hat{s}_n(x) = s(x)$, and assuming the expectations above can be computed exactly.

Consistency of the naïve estimator.   In order for our application of the statistical jackknife to be valid, we must show that the naïve estimator is a *consistent* estimator of $\mathcal{A}_{\mathcal{D}_2|s_1}$—that is, as $n \to \infty$, $\hat{\mathcal{A}}_{\mathcal{D}_2|s_1}^n \to \mathcal{A}_{\mathcal{D}_2|s_1}$. Note that since we operate in the regime where the number of distinct images m greatly exceeds the number of annotators per image n, we will assume that the expectations above can be computed exactly. Note that the estimator remains consistent even if this is not the case, with the additional constraints that $m \to \infty$ and $m/n \to \infty$, but this greatly complicates the proof and we will show empirically that the estimator is robust to changes in m in the relevant regime.

Note that in the "infinite m" regime, the variance of the naïve estimator is 0. Thus, all of the error is due to bias in the estimator. In the following, we assume that $p_1(s)$, $p_2(s)$, and $p_2(s|f = 1)$ are continuous differentiable densities bounded away from zero and with bounded derivatives ($|d^r/dx^r\, p_i(x)| < C$).

$$p_i\left(\hat{s}_n(x) = \frac{k}{n}\right) = \int_0^1 p_i(s) \cdot \text{Binom}\left(n, n, k, s\right)\, ds \tag{12}$$

$$= \int_0^1 p_i(s) \cdot \binom{n}{k} \cdot s^k \cdot (1-s)^{n-k}\, ds \tag{13}$$

$$= \int_0^1 p_i(s) \cdot \frac{s^k(1-s)^{n-k}}{(n+1) \cdot B(k+1, n-k+1)}\, ds \qquad B(\cdot, \cdot) \text{ is the Euler beta function} \tag{14}$$

$$= \frac{1}{n+1} \int_0^1 p_i(s) \cdot \text{Beta}(s; k+1, n-k+1)\, ds \tag{15}$$

$$= \frac{1}{n+1} \mathbb{E}_{s \sim \text{Beta}(\cdot; k+1, n-k+1)}[p_i(s)] \tag{16}$$

Using a Taylor expansion of $p_i(s)$ around $\mathbb{E}_{s \sim \text{Beta}(\cdot; k+1, n-k+1)}[s]$, we can bound the above expression:

$$= \frac{1}{n+1} \mathbb{E}\left[p_i(\mathbb{E}[s]) + (s - \mathbb{E}[s])p_i'(\mathbb{E}[s]) + \frac{(s - \mathbb{E}[s])^2}{2}p_i''(\mathbb{E}[s]) + \sum_{r=3}^{\infty} \frac{(s - \mathbb{E}[s])^r}{r!}p_i^{(r)}(\mathbb{E}[s])\right]$$

$$= \frac{1}{n+1}\left(p_i(\mathbb{E}[s]) + \frac{1}{2}\text{Var}[s] \cdot p_i''(\mathbb{E}[s]) + O\left(\frac{1}{n^{7/2}}\right)\right)$$

$$= \frac{1}{n+1}p_i\left(\frac{k+1}{n+2}\right) + \frac{(k+1)(n-k+1)}{(n+1)(n+2)^2(n+3)} \cdot p_i''(\mathbb{E}[s]) + O\left(\frac{1}{n^{7/2}}\right)$$

27

Now, using the presumed boundedness of derivatives we can write:

$$\left| p_i \left(\hat{s}_n(x) = \frac{k}{n} \right) - \frac{1}{n+1} \cdot p_i \left(\frac{k+1}{n+2} \right) \right| \leq \frac{(k+1)(n-k+1)}{(n+1)(n+2)^2(n+3)} \cdot p_i''(\mathbb{E}[s]) + O \left(\frac{1}{n^{7/2}} \right) \tag{17}$$

$$|\mathcal{A}_{\mathcal{D}_2|s_1} - \hat{\mathcal{A}}_{\mathcal{D}_2|s_1}^n| = p_2(f(x)=1) \left| \int_0^1 \frac{p_2(s|f(x)=1)}{p_2(s)} p_1(s)\,ds - \sum_{k=0}^n \frac{p_2(\hat{s}_n(x)=\frac{k}{n}|f(x)=1)}{p_2(\hat{s}_n(x)=\frac{k}{n})} p_1(\hat{s}_n(x)=\frac{k}{n}) \right|$$

$$\leq \left| \int_0^1 \frac{p_2(s|f(x)=1)}{p_2(s)} p_1(s)\,ds - \frac{1}{n+1} \sum_{k=0}^n \frac{p_2 \left(\frac{k+1}{n+2}|f(x)=1 \right)}{p_2 \left(\frac{k+1}{n+2} \right)} p_1 \left(\frac{k+1}{n+2} \right) \right|$$

$$+ \left| \frac{1}{n+1} \sum_{k=0}^n \frac{p_2 \left(\frac{k+1}{n+2}|f(x)=1 \right)}{p_2 \left(\frac{k+1}{n+2} \right)} p_1 \left(\frac{k+1}{n+2} \right) - \frac{p_2 \left(\hat{s}_n(x)=\frac{k}{n}|f(x)=1 \right)}{p_2 \left(\hat{s}_n(x)=\frac{k}{n} \right)} p_1 \left(\hat{s}_n(x)=\frac{k}{n} \right) \right|$$

Now, note that the first term in the above is simply the error in the Riemmann sum approximation of the integral, which vanishes as $n \to \infty$. The second term is bounded by n times the error in each individual term of the sum, which we bounded as $O(n^{-2})$ in Equation (17).

Near-linearity of bias and likely underestimation. Recall that for the statistical jackknife to yield a reliable estimate of the adjusted accuracy, the bias in the naïve estimator must be analytic in $\frac{1}{n}$, and in particular should be dominated by a $O(\frac{1}{n})$ term (as this corresponds to precisely the term accounted for by the jackknife). In Figure 12 we show that the bias estimated by our jackknife procedure is indeed roughly linear in $1/n$, but grows slightly faster than $\frac{1}{n}$, likely leading the jackknife to provide an underestimate.
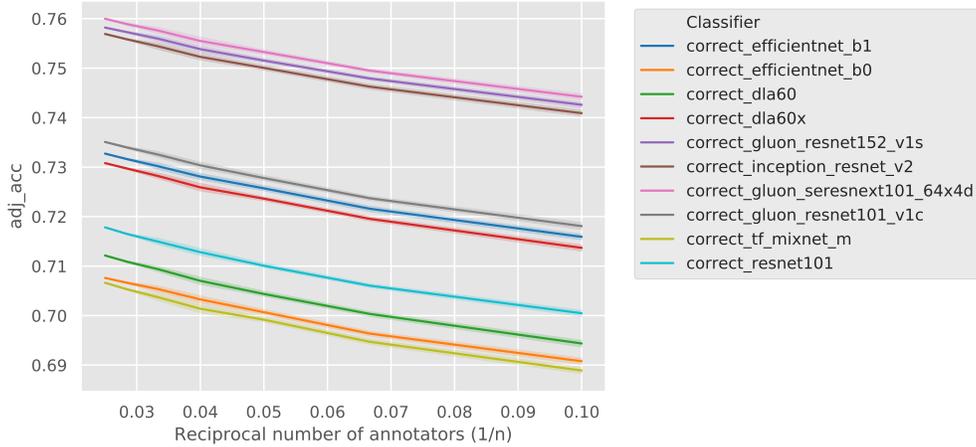
Figure 12: We plot the jackknife adjusted accuracy estimators of 10 models. On the y axis is shown the value of the n-sample jackknife estimate, with $1/n$ on the x axis. The fact that the plot is nearly linear suggests that our bias is indeed dominated by a $O(1/n)$ term, thus further justifying our use of the statistical jackknife in Section 5.2. Furthermore, the slightly accelerating slope as one moves left on the plot indicates that any error in the jackknife estimate is likely to be an underestimation of bias, rather than an overestimation.

F Model Fitting

In this section, we describe our methods for parametric modeling.

F.1 Confidence Intervals

To construct 95% confidence intervals we perform 450 bootstrapped estimates (over the included images) of adjusted accuracy for each classifier. We then plot the 2.5% and 97.5% percentiles from the bootstrap estimates as the confidence intervals for each classifier.

F.2 Varying Annotators

We plot the results of varying the number of annotators in Figure 13.
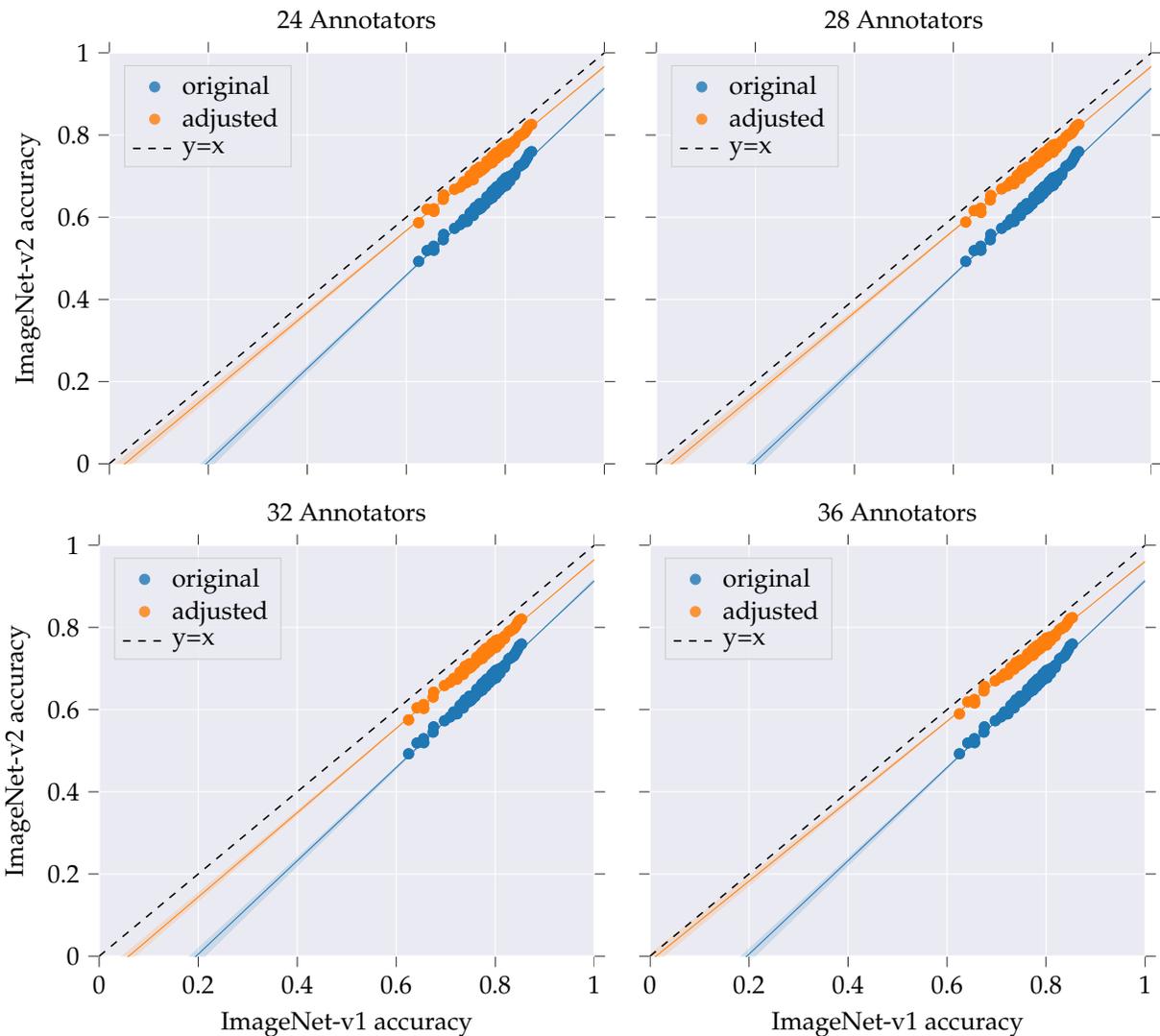
Figure 13: Replicating the v1 vs v2 accuracy plot using different numbers of annotators. We obtain similar results even as we decrease the number of annotators by less than half.

F.3 Varying Model Expressiveness

We plot the results of varying the number of parameters (here, by changing the number of beta distributions in our mixture) in Figure 14.
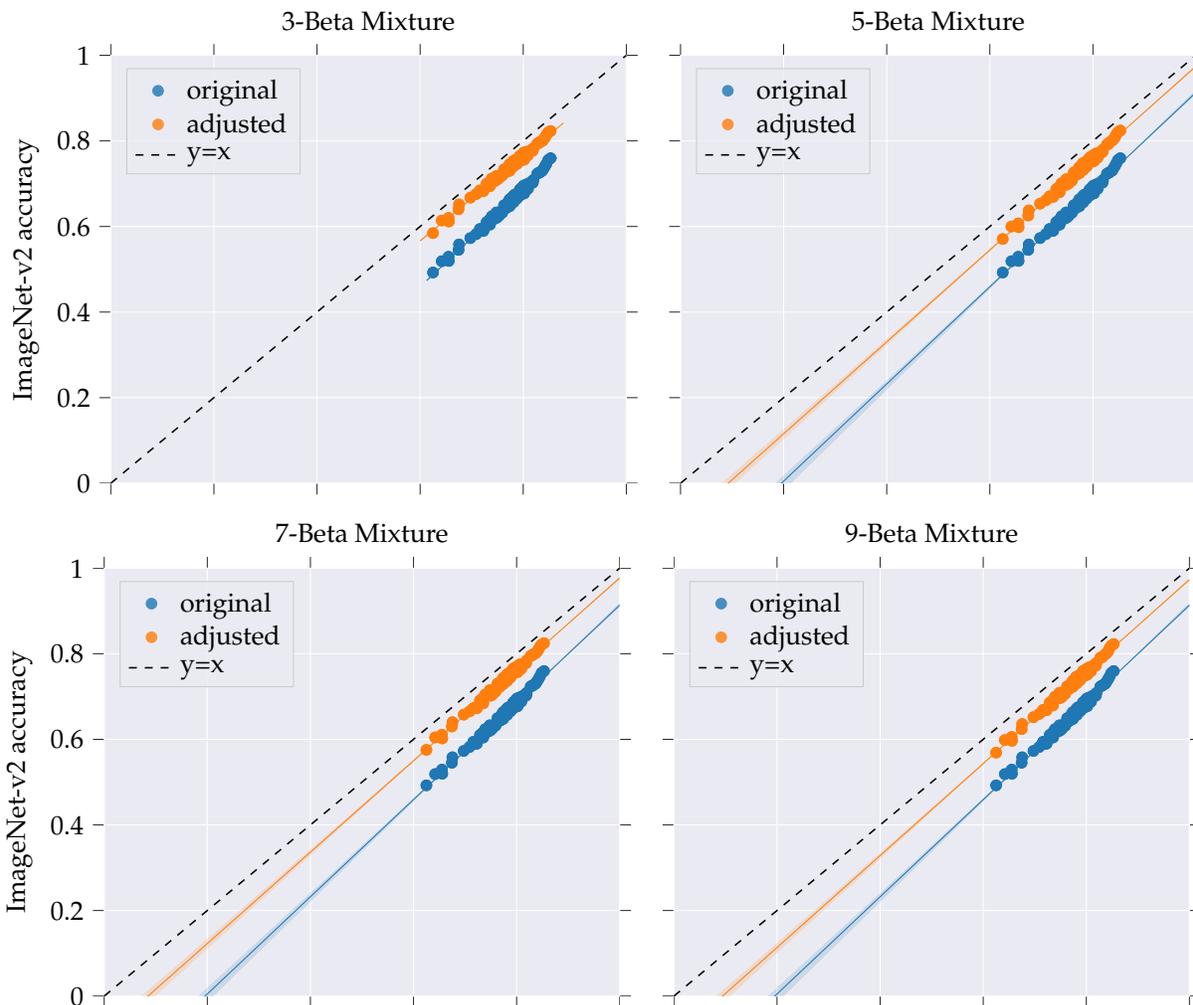
Figure 14: Replicating the v1 vs v2 accuracy plot using different numbers of parameters. We obtain similar results even as we increase the number of parameters by more than four-fold.

F.4 EM Algorithm for Mixture Fitting

To fit the parameters of the beta-binomial mixture model we apply the Expectation-Maximization algorithm, optimizing over mixture coefficients $\{\pi_i\}$, as well as parameters of each mixture element $\{(\alpha_i, \beta_i)\}$. Our application of the EM algorithm is rather canonical—first, we compute membership probabilities p^i_j for each example j with respect to each mixture element i, then minimize the weighted log-likelihood with respect to the mixture probabilities. Pseudocode is given in Algorithm 1.

Algorithm 1 Our instantiation of the EM algorithm

Input: A set of size n of empirical selection probabilities $\{\hat{s}_j\}$, the observed rate at which each image was selected, number of mixture components k.
Start with random guesses for all parameters:

$$\alpha_i, \beta_i, \pi_i \leftarrow \text{random}$$

for each training iteration **do**

    1. Calculate membership probabilities for each observed element:

$$p_j^i = \frac{\pi_i \cdot p(\hat{s}; \alpha_i, \beta_i, 40)}{\sum_{r=1}^{k} \pi_r \cdot p(\hat{s}; \alpha_r, \beta_r, 40)} \qquad \forall\, j \in [n], i \in [k],$$

where $p(\cdot; \alpha, \beta, 40)$ is likelihood under the beta-binomial distribution with 40 samples.
2. As is standard in EM, update the parameters by minimizing the expected log-likelihood, weighted by the membership probabilities—i.e. update

$$\alpha_i, \beta_i = \min_{\alpha, \beta} \sum_{j=1}^{n} p_j^i \cdot \ell(\hat{s}_j; \alpha_i, \beta_i, 40),$$

where $\ell(\cdot) = -\log(p(\cdot))$ is the negative log-likelihood, and

$$\pi_i = \frac{\sum_{j=1}^{n} p_j^i}{\sum_{r=1}^{k} \sum_{j=1}^{n} p_j^r}.$$

end for

G   Full Model Results

In Appendix Table 2, we detail the set of models we use in our evaluation along with their corresponding Top-1 accuracies on ImageNet-v1 and -v2 validation sets. We use open-source pre-trained implementations from `https://github.com/rwightman/pytorch-image-models` for all architectures.

Model	v1	v2
`tf_mobilenetv3_small_minimal_100`	63.070%	48.270%
`dla46_c`	64.950%	51.330%
`tf_mobilenetv3_small_075`	65.490%	50.800%
`dla46x_c`	66.130%	52.200%
`tf_mobilenetv3_small_100`	67.500%	53.960%
`dla60x_c`	68.170%	55.660%
`resnet18`	70.420%	56.850%
`gluon_resnet18_v1b`	71.280%	57.610%
`seresnet18`	71.840%	58.200%
`tf_mobilenetv3_large_minimal_100`	71.910%	57.870%
`hrnet_w18_small`	72.860%	58.120%
`tv_resnet34`	73.080%	60.060%
`spnasnet_100`	73.760%	61.040%
`tf_mobilenetv3_large_075`	73.850%	59.430%
`gluon_resnet34_v1b`	74.470%	61.630%
`mnasnet_100`	74.620%	61.020%
`densenet121`	74.650%	61.810%
`dla34`	74.680%	61.510%
`seresnet34`	74.820%	62.330%
`resnet34`	74.990%	62.240%
`hrnet_w18_small_v2`	75.000%	61.540%
`fbnetc_100`	75.080%	61.240%
`resnet26`	75.270%	62.730%
`semnasnet_100`	75.690%	62.360%
`tf_mobilenetv3_large_100`	75.710%	61.400%
`mobilenetv3_rw`	75.740%	61.870%
`tv_resnet50`	75.820%	62.600%
`dpn68`	76.020%	63.000%
`tf_mixnet_s`	76.210%	62.040%
`tf_efficientnet_b0`	76.240%	63.050%
`densenet169`	76.370%	63.450%
`hrnet_w18`	76.500%	64.560%
`mixnet_s`	76.570%	62.840%
`dla60`	76.800%	64.610%
`efficientnet_b0`	76.820%	64.050%
`seresnext26_32x4d`	76.980%	64.050%
`resnet26d`	77.020%	63.970%
`resnet101`	77.090%	65.020%
`tf_mixnet_m`	77.120%	63.540%
`tf_efficientnet_b0_ap`	77.130%	64.290%
`tf_efficientnet_cc_b0_4e`	77.190%	64.110%
`tf_efficientnet_es`	77.200%	64.360%
`inception_v3`	77.240%	65.090%
`densenet161`	77.240%	64.790%
`densenet201`	77.280%	64.480%
`res2net50_48w_2s`	77.420%	64.260%
`gluon_resnet50_v1b`	77.530%	65.130%

adv_inception_v3	77.680%	65.380%
mixnet_m	77.710%	64.090%
gluon_resnet50_v1c	77.710%	65.180%
dpn68b	77.720%	64.830%
tf_efficientnet_cc_b0_8e	77.740%	64.410%
resnet152	77.760%	66.410%
tv_resnext50_32x4d	77.790%	65.130%
dla60_res2next	77.980%	65.820%
hrnet_w30	78.010%	65.860%
seresnet50	78.020%	65.160%
res2net50_26w_4s	78.050%	64.590%
seresnet101	78.070%	66.150%
dla60x	78.160%	66.090%
res2next50	78.180%	65.370%
tf_inception_v3	78.220%	65.480%
dla102	78.290%	65.710%
hrnet_w44	78.300%	67.130%
dla169	78.380%	66.450%
wide_resnet101_2	78.430%	65.460%
wide_resnet50_2	78.430%	65.750%
tf_efficientnet_b1	78.530%	65.620%
res2net50_14w_8s	78.540%	65.180%
hrnet_w32	78.600%	65.620%
dla60_res2net	78.610%	65.550%
tf_mixnet_l	78.610%	65.750%
hrnet_w40	78.670%	66.600%
efficientnet_b1	78.690%	66.300%
tf_efficientnet_em	78.710%	65.560%
resnext50_32x4d	78.790%	66.530%
dla102x	78.810%	66.140%
seresnet152	78.850%	66.540%
hrnet_w48	78.860%	66.320%
gluon_inception_v3	78.880%	66.110%
res2net50_26w_6s	78.890%	66.200%
mixnet_l	78.890%	66.180%
resnet50	79.000%	65.770%
hrnet_w64	79.090%	67.650%
res2net50_26w_8s	79.100%	66.710%
xception	79.110%	66.320%
gluon_resnet101_v1b	79.110%	66.300%
gluon_resnet50_v1s	79.140%	66.220%
gluon_resnet50_v1d	79.200%	66.740%
tf_efficientnet_b1_ap	79.330%	66.290%
tf_efficientnet_cc_b1_8e	79.360%	65.890%
dla102x2	79.400%	67.830%
seresnext50_32x4d	79.420%	66.790%
resnext101_32x8d	79.490%	66.660%
gluon_resnext50_32x4d	79.630%	67.610%
gluon_resnet101_v1c	79.660%	66.870%
resnext50d_32x4d	79.700%	67.700%
tf_efficientnet_b2	79.730%	67.320%
res2net101_26w_4s	79.740%	66.750%
dpn98	79.830%	67.550%
dpn107	79.950%	67.490%

dpn131	80.020%	67.580%
gluon_resnet152_v1b	80.030%	67.610%
gluon_xception65	80.070%	68.000%
ens_adv_inception_resnet_v2	80.080%	68.630%
efficientnet_b2	80.080%	67.490%
gluon_seresnext50_32x4d	80.100%	67.800%
inception_v4	80.170%	68.490%
gluon_resnet152_v1c	80.230%	67.660%
gluon_resnet101_v1s	80.320%	68.020%
mixnet_xl	80.340%	68.000%
tf_efficientnet_el	80.550%	67.190%
seresnext101_32x4d	80.570%	69.050%
gluon_resnext101_32x4d	80.580%	67.510%
dpn92	80.600%	66.750%
tf_efficientnet_b2_ap	80.680%	67.380%
inception_resnet_v2	80.840%	68.750%
gluon_resnext101_64x4d	80.860%	69.050%
gluon_resnet152_v1d	80.870%	68.730%
gluon_resnet101_v1d	81.020%	67.960%
gluon_seresnext101_32x4d	81.060%	68.890%
gluon_resnet152_v1s	81.470%	68.980%
gluon_seresnext101_64x4d	81.700%	69.040%
tf_efficientnet_b3	81.810%	69.360%
gluon_senet154	81.900%	69.930%
senet154	82.100%	70.020%
tf_efficientnet_b3_ap	82.130%	69.920%
nasnetalarge	82.780%	71.660%
pnasnet5large	83.210%	71.970%
tf_efficientnet_b4	83.350%	71.920%
tf_efficientnet_b5	84.030%	72.200%
tf_efficientnet_b4_ap	84.210%	72.130%
tf_efficientnet_b5_ap	84.260%	73.520%
tf_efficientnet_b6	84.510%	72.940%
tf_efficientnet_b6_ap	85.000%	74.570%
tf_efficientnet_b7_ap	85.460%	75.110%

Table 2: Models used in our analysis with the corresponding Top-1 on the ImageNet v1 and v2 validation sets.