ПРАВИТЕЛЬСТВО РОССИЙСКОЙ ФЕДЕРАЦИИ
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ
УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
"Национальный исследовательский университет
"Высшая школа экономики"

НЕГОСУДАРСТВЕННОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ
ВЫСШЕГО ОБРАЗОВАНИЯ
"РОССИЙСКАЯ ЭКОНОМИЧЕСКАЯ ШКОЛА"(ИНСТИТУТ)

ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА

Преодоление предвзятости: изучение эффекта якорения в разметках на краудсорсинговых платформах

Программа Бакалавр экономики

Совместная программа по экономике НИУ ВШЭ и РЭШ

Автор:
С.Н. Ишмуратов

Руководитель ВКР:
И.А. Стельмах

Москва, 2024 г.

Sailing Through Bias: Exploring Anchoring Bias in Human Data Labeling on Crowdsourcing Platforms

Аннотация

Краудсорсинг — популярый способ разметки данных для современных моделей машинного обучения, в том числе для больших языковых моделей (LLMs). Почти все методы агрегации краудсорсинговых данных основаны на методе максимального правдоподобия, который включает в себя предположение о статистической независимости ответов одного человека на разные задания. В этой работе я экспериментально проверяю справедливость этого предположения и тестирую краудсорсинговые данные на наличие эффекта якорения — известного когнитивного искажения, при котором ответы на будущие вопросы зависят от ответов на предыдущие.

Эксперименты на существующих датасетах (Toloka IMDB-WIKI) и новых данных показывают, что популярном дизайне краудсорсинговых заданий эффект якорения может оказывать существенное влияние на качество данных. С другой стороны, использование альтернативного фреймворка попарного сравнения позволяет нивелировать эффект якорения и повысить качество данных.

Abstract

Crowdsourcing is a popular method of machine learning dataset labeling that is also used for large language models (LLMs). Almost all aggregation methods for crowdsourcing data labeling tasks use the maximum likelihood method, thus the assumption of an independent task-by-task distribution of answers. In this paper, I experimentally examine this assumption and check if the anchoring bias exists in the data. The workers might deal with anchoring effects when they subconsciously label the task affected by their previous answers.

The results on existing datasets (Toloka IMDB-WIKI) and new data show that the anchoring bias might significantly affect the quality of the data. On the other hand, using a side-by-side comparison design will help to struggle with this bias and improve the quality of the data.

Contents

1 Introduction **3**

2 Related work **4**

2.1 Empirical work on biases 5

2.2 Theoretical work on modeling 6

3 Data description **7**

3.1 Regression setup 7

3.2 Side-by-side setup (forward bias) 8

3.3 Side-by-side setup (anti-anchor) 8

3.4 Side-by-side setup (real data) 9

4 Modelling anchoring bias **9**

4.1 Regression setup 9

4.2 Side-by-side setup (forward bias) 10

4.3 Side-by-side setup (anti-anchor) 11

4.4 Side-by-side setup (real data) 11

5 Estimating the quality of data **11**

5.1 Regression setup 11

5.2 Side-by-side setups 12

6 Results **12**

7 Discussion **14**

7.1 Internal validity 14

7.2 External validity 14

8 Conclusion **15**

References **16**

Appendix A **18**

Toloka task interface example 18

Appendix B **19**

My dataset 19

1 Introduction

While the machine learning sphere is growing and prospering, crowd-based data labeling is still an important part of collecting data for full-scale applied ML models. Human data labeling is required for both teacher-based learning and reinforcement learning. Crowd-soursing is widely used for the usefulness and ethical estimation of generative models, the improvement of web-search results, speech recognition problems, and a set of others.

Using platforms like Yandex Toloka or Amazon Mechanical Turk, the data *requester* can use the numerous *workers* to label the data for different types of projects. Usually, the setup on these platforms includes a task *overlap*, when the requester gives the same task to multiple workers. In this setup, the workers are assigned for each task randomly, and they don't know the answers of other workers to the task. This design allowed the requester to get multiple (and usually slightly different) independent answers for each task. The next part is to aggregate the multiple answers to a single golden answer. The powerful idea of crowd-based data labeling is that the aggregated answers of multiple workers might be even better than the single answers of the best workers among them. To solve the problem of data aggregation, modern computer science provides plenty of different models. In this work, I will focus on the classification types of tasks and related models. The simplest approach, the *majority vote*, uses only the information about the single task and picks the most popular answer. However, the more advanced algorithms, such as the Dawid and Skene (1979) model or the Whitehill et al. (2009) GLAD model, use the data to predict the skill of each worker and the difficulty of each task. Most of these advanced models are based on probabilistic parametrization using the Expectation-Minimization algorithm. Notice, that this approach includes the assumption that the worker's answer for the current task doesn't depend on the answers for previous tasks. However, in real life, the worker might suffer from the anchoring effects and give the answer while keeping the previous tasks in mind.

The anchoring effect was described by Kahneman and Tversky (1979). This effect occurs when the person is not totally convinced by the answer and uses some *anchor* as a reference point. For instance, Tversky and Kahneman (1973) asked the first group of participants to estimate $1 \cdot 2 \cdot 3 \cdot 4 \cdot 5 \cdot 6 \cdot 7 \cdot 8$ and the second group of participants to estimate $8 \cdot 7 \cdot 6 \cdot 5 \cdot 4 \cdot 3 \cdot 2 \cdot 1$ very quickly. The participants do not have enough time to produce all numbers, so they produce only a few first numbers, and thus they get some anchors. Then, the authors show that the estimates in the first group are significantly lower than in the second.

Research question. In this paper, I create the data labeling project, which would suffer from the anchoring bias of its design. The results of labeling this project will help me to answer the following important questions:

- *What is the size of the effect of anchoring bias on the project with bad design? On*

real data samples? Is it a significantly important effect?

The main reason for the estimation of the badly designed project is to understand whether or not the anchoring effect might be a potential problem for *any* type of project. For the real-life project, this estimation might also be helpful to evaluate the potential damage if *some* parts of tasks have bad design.

My hypothesis is that the anchoring bias is statistically significant only in specific corner cases. The choice of the best model might depend on data and project traits, but I assume that reweighing suspicious answers would help to struggle with this bias.

Another potential outcome of my work is my custom crowd-labeled dataset, which can be used for further research. You can find a link and description in Appendix B.

The remaining part of the work is structured in the following way: in the related work section, I give a brief overview of related papers, both empirical and theoretical studies. Next, in the data description section, I depict how my dataset was constructed. In the modeling section, I explain the regression setups and the definitions of treated indicators. The estimation quality of the data and some summary statistics are provided in the next section. Finally, I comment on the results and discuss the internal and external validity of the models. In the short conclusion, I summarize the achieved result and give some of my thoughts about the development of this field in the future.

2 Related work

My work focuses on the issue of cognitive bias in crowdsourcing. In this section I discuss relevant past studies. There are two different approaches towards this problem in the literature:

- **Empirical** The first group of papers studies empirical aspects of the problem. These works test for biases in real data and aim at identifying the features of the experiment design that may lead to biases.
- **Theoretical** The second group of papers studies theoretical aspects of the problem. These works design algorithms and tools to mitigate the presence of biases.

Below I discuss the aforementioned groups of works in detail. But before that I note that the present work belongs to the massive line of works on crowdsourcing — a distributed approach towards solving complex problems using wisdom of the crowd Surhone et al. (2010); von Ahn et al. (2008); Bernstein et al. (2015); Franklin et al. (2011). The ultimate goal of these works is to find optimal mechanisms that achieve high accuracy when the crowd is diverse and even unskilled. This work continues this line by aiming to remove the impact of behavioural biases on the quality of crowdsourced data. In that, my work complements an ongoing effort towards data excellence in Machine Learning Gebru et al. (2021).

2.1 Empirical work on biases

The fundamental contribution in this field was made by Kahneman and Tversky (1979). Later, they developed these ideas in Tversky and Kahneman (1992). This was the first important result in general thing on biases. In the following review I will focus on the particular application of these ideas to the crowdsourcing sphere.

The problem of bias in human data labeling existed even before the dawn of machine learning. For instance, Day (1969) explored pairwise product tests and tries to find evidence of position bias in this design. This is one of the first articles on the related topic. The author find that for significantly different products the bias is negligible. Also, it's one of the first notions of behavior-based biases in crowd-based tasks. This result illustrates that behavior-based bias effects exist not on the full real data sample but only on some specific parts of it.

The more recent papers in the field of crowdsourcing provide numerous possible reasons for the presence of bias in the data. For example, Draws et al. (2021) presented a checklist for data requesters on how to avoid the typical behavior biases described by Kahneman et al. (2011) in a 12-item checklist for combating cognitive biases in business. In the practical part, the authors provide an example of checking the task design using this checklist for a certain dataset. It's important to keep in mind that this dataset was constructed to show most of these biases, and the real crowd project will demonstrate less exposure to these effects. In particular, they suggest using Krippendorff's alpha as a general measure, which might be helpful to detect cognitive biases *a posteriori*, and a set of Bayesian correlations to detect some types of biases.

I am going to intently concentrate on the anchoring effects in this paper, so I examine the anchoring effects in this paper more deeply. The authors use the correlation between the first annotation made by each worker and their remaining annotations. The task was to prepare viewpoint annotations (estimate the page content on a seven-point scale ranging from "strongly opposing" to "strongly supporting" the topic) for the web pages with controversial topics such as *Should zoos exists?*. They use Spearman correlation coefficients, which let them avoid problems with the different individual scales described in Wang and Shah (2018). Practically, the main problem of this research is that many types of biases (such as anchoring on the previous task or overconfidence bias) are almost inevitable. Even detecting such types of biases requires a specific task design, which may drastically increase the cost of a project. Also, for the anchoring problem, an important setting for the experiment is a random or predetermined first task, but this trait is unclear from the article. An example of a better design for similar experiment in the field of peer-review estimation can be found in Liu et al. (2023)

2.2 Theoretical work on modeling

The crux of crowdsourcing systems is redundancy and aggregation. While a single individual (*worker*) in a typical crowdsouring system may be inaccurate, a clever aggregation of responses from multiple workers may lead to a decent results Dawid and Skene (1979); Shah et al. (2016); Shah and Wainwright (2018)

In contrast to most of such works that try to mitigate noise, in this study I focus on the problem of bias. In what follows, I discuss several works that also try to mitigate biases in crowdsourcing with algorithms and models.

Thus, Saab et al. (2019) concentrates on estimating the Dunning-Kruger effect for the crowd dataset aggregation problem. The authors show that mathematically, the simple plurality-vote model performs better than the popular models with weights, which are based on the confidence level.

Then, they test the model on a dataset that contains the user's answers in an Android game based on the "Who Wants to Be a Millionaire" show. The Android game participants answered the questions from the TV game simultaneously with the TV game participants. The results of testing on this dataset are very impressive; however, it's not a typical design for ML crowd-based datasets. Also, it's important that this design might include cheaters' answers because participants might use the Internet or just occasionally see the answer on TV before submitting the answer.

The problem with this paper is that the model might really perform better than the standard weight-based models, but numerically, this improvement is too low. Moreover, the majority-vote model is also faced with a number of other problems, such as cheaters' labeled data and some general types of biases. The practical part includes only a very specific dataset, which generates the Dunning-Kruger effect by its design, and the real data might not be affected by this effect so much.

The other example of this approach is Bugakova et al. (2019). The authors provide a model that was designed to debias the worker's answers, which were corrupted by some elements of the task's design. They work with a certain type of pairwise comparison task and the dataset of Google and Yandex Search Engine Page Results (SERP). This is a really important ML problem in the field of search engine development. The model is based on the classical Bradley-Terry model (Bradley and Terry (1952)). However, their model contains two additional parameters to model the probability of a biased answer for each worker and the reaction of the worker to task design features.

They test the model on the task of choosing the best SERP among the pair for the given search query. The task was designed in such a way that one SERP was significantly better than the second, so the workers who prefer the second might suffer from bias. They tested the model in this setup and beat some well-known baselines. They also tested the model on the dirty data with additional left-biased malicious workers answers and found

that their model has a better result among the other models for this setup.

This work shows that the pairwise comparisson design might really suffer from task design biases. However, the authors didn't pay much attention to the reasons for these biases (they only mentioned cheaters and some possible preferences of Yandex SERPs for workers from Yandex Toloka). In my opinion, concentrating on the reasons for these biases described in many behavior economics papers might improve the results of this model.

3 Data description

For the empirical part of the work, I use the IMDB-WIKI dataset collected by Rothe et al. (2018). This dataset contains a huge set of human photos and the ages of the people in these photos. The goal of ML models trained on this dataset is to predict the age of the person based on his or her single photo. I also use the Pavlichenko and Ustalov (2021) dataset, which is the balanced by gender and age subset of the IMDB-WIKI dataset already estimated on Yandex Toloka in side-by-side design.

Side-by-side design is a pairwise comparison, where the worker's goal is to predict the older person but not the exact age of both people in the photos. In contrast, the exact prediction design requires the estimation of the age for each photo by a single number.

In my work, I also provide a custom Toloka-labeled dataset, which contains both side-by-side and exact prediction designs. This dataset is a subset of Pavlichenko and Ustalov (2021) dataset. You can find an example of the task from my project in Appendix A.

Notice that I use several different pools in my setups only to clarify the way the datasets were constructed. In the models, I use the whole dataset from every pool for the project with the current setup.

Three of the four setups have side-by-side designs. I focus on this design because I expect that it would be helpful to struggle with anchoring bias and examine this assumption. A straightforward (regression) setup might have a very different design depending on the specific task. So for this setup, I only try to show that the anchoring bias *might* exist.

3.1 Regression setup

I constructed the pools for this setup in such a way that two groups of workers see the middle-aged person after a young and an old person. In this way, I test the anchoring hypothesis for the exact prediction design. I expect that the estimations from the group that sees the young person first will be lower than the estimations from the second group.

Let X be a set of 30 images with persons with a true age 30 - 40.

Table 1: Regression pools settings

pool size	page size	overlap	maximum pages from single worker	price for page
30 pages	3 tasks	2	3	0.01$

- **pool A** contains on page 2 tasks where the true age of the person is under 20, and the last task where the true age of the person is between 30 and 40 from set X.
- **pool B** contains on page 2 tasks where the true age of the person is older than 50, and the last task where the true age of the person is between 30 and 40 from set X.
- **pool C** contains on page 2 randomly chosen tasks and the last task where the true age of the person is between 30 and 40 from set X.

3.2 Side-by-side setup (forward bias)

For side-by-side design, I tested two different types of anchoring. In the first setup, the pair with two people of the same age is followed by the pair with two people of obviously different ages.

Table 2: Side-by-side (forward bias) pools settings

pool size	page size	overlap	maximum pages from single worker	price for page
30 pages	3 tasks	2	3	0.01$

Let X be a set of is 30 pairs, where the difference in true ages is less than 5 years.

- **pool A** contains on page 2 pairs where the left person is under 25 and the right person is older than 50. The last pair from set X.
- **pool B** contains on page 2 randomly chosen pairs. The last pair from set X.

3.3 Side-by-side setup (anti-anchor)

The second setup contains a series of pairs when the left person is obviously older than the second. At the end of the series, I include a pair of approximately the same age.

Table 3: Side-by-side (anti-anchor) pools settings

pool size	page size	overlap	maximum pages from single worker	price for page
30 pages	5 tasks	2	3	0.015$

Again, let X be a set of is 30 pairs, where the difference in true ages is less than 5 years.

- **pool A** contains on page 4 pairs where the left person is under 25 and the right person is older than 50. The last pair from set X.
- **pool B** contains on page 4 randomly chosen pairs. The last pair from set X.

3.4 Side-by-side setup (real data)

In this setup, I check the model on the full Pavlichenko and Ustalov (2021) dataset, which is some sort of real data. Unfortunately, in this project, the authors include 1 control task on each page, and this task was not published in the data, so I model the bias on 3 tasks on pages, keeping in mind that the fourth control task exists but was not presented in the dataset. Notice that the control task is designed to be easy for its nature (the authors use pairs with an age gap greater than 20 years as golden tasks). Thus, I expect that on the full dataset with control tasks, the size of the effect would be bigger.

Table 4: Side-by-side (anti-anchor) pools settings

pool size	page size	overlap	maximum pages from single worker	price for page
83416 pages	3 (+1) tasks	1	-	not published

4 Modelling anchoring bias

To create a bias effect, I use the "obvious" pairs, where one of a pair is significantly older for a side-by-side setup or show several photos with elderly persons before the photo with a young person.

The important thing to notice is that not all of the workers who solve tasks from these pools with specially selected tasks actually demonstrate exposure to the bias-created setup. Some workers pick the older person the youngest person from pairs which were created as "obvious" and cetera. There are many possible reasons for this behavior. The ground truth labels that I used might be incorrect for some photos. The worker might be a cheater who clicked the answers randomly. A good worker might accidentally make a mistake or read the instructions inattentively.

Note that, despite the fact that I use several pools for some setups, the regression models for each setup include all of the labels that are the last on the page from all of the pools in this setup. The reason for using several pools is to rebalance the data in a way that facilitates the bias appearing more frequently compared with real projects.

4.1 Regression setup

In this setup I measure the effect of anchoring bias using following OLS regression:

$$y_i = \alpha + \beta_1 I_i^{A-treated} + \beta_2 I_i^{B-treated} + \beta_3 x_i + \varepsilon_i$$

Here treated indicators are not binary variables, they are defined in the following way:
$\mathrm{I}_i^{A-treated} = (1 - \sigma(g_1 - 25))(1 - \sigma(g_2 - 25))$
$I_i^{B-treated} = \sigma(g_1 - 45)\sigma(g_2 - 45)$

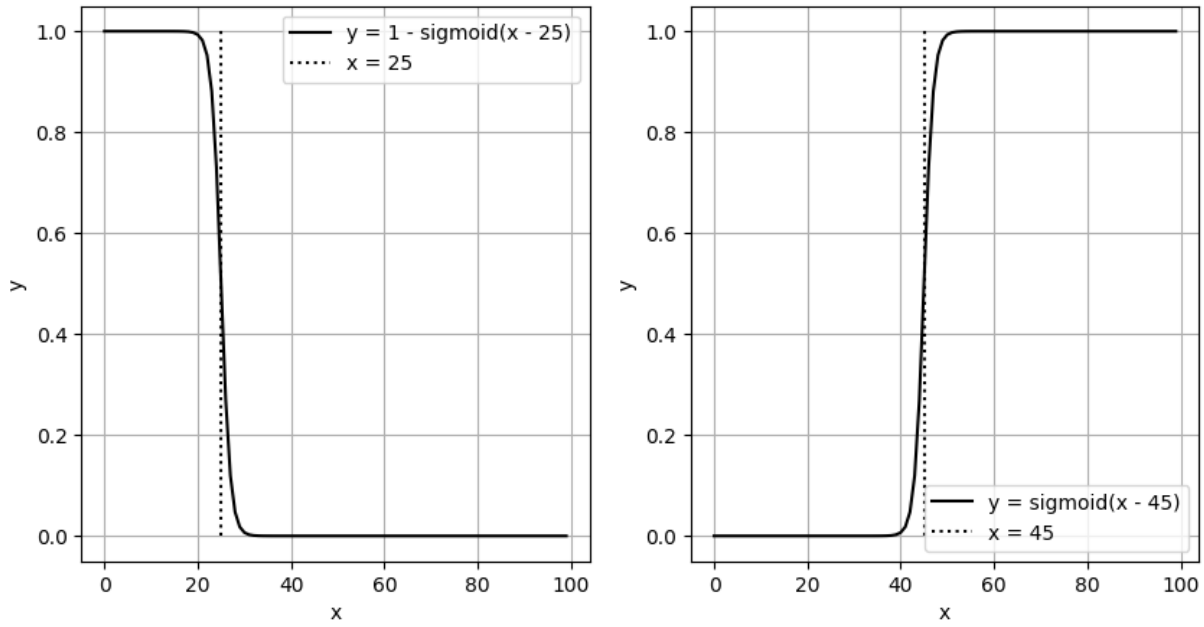

Figure 1: Indicators' multipliers

where g_1, g_2 - worker's guess about the first and second task on the page and $\sigma(x) = \frac{1}{1+e^x}$ - sigmoid function. y_i is a worker's guess about the last task on the page. Here and next ε_i is a residual.

The main reason to use the sigmoid function instead of some binary threshold is to avoid situations where the answer near the threshold impacts the treated indicator too much, as you can see in 1.

Multiplying the sigmoid functions for the first and second guesses, I get the indicator that is close to 1 when the worker can be defined as treated and close to 0 otherwise.

25 and 45 were chosen as appropriate thresholds empirically; there is no deep intuition behind this choice.

4.2 Side-by-side setup (forward bias)

Here, I measure the effect of anchoring bias using the following logistic regression:

$$y_i = \beta_1 I_i^{treated} + \beta_2 x_i^l + \beta_3 x_i^r + \varepsilon_i$$

where $I_i^{treated}$ equal to 1 indicates that the worker selected left for both first and second pairs, $I_i^{treated}$ equal to -1 indicates both right selections, and y_i is a binary variable with 1 if the worker selected left for the last pair on the page i and x_i^l, x_i^r are ground-truth labels for left and right photos from the third pair, we use the data from both pools A and B. Notice that I don't use bias component in all of the side-by-side setups because I assume that, apart from control variables, the average preference should be 0.5.

4.3 Side-by-side setup (anti-anchor)

I estimate the effect of anchoring bias using the following regression for this setup:

$$y_i = \beta_1 I_i^{treated} + \beta_2 x_i^l + \beta_3 x_i^r + \varepsilon_i$$

where $I_i^{treated}$ indicates equal to 1 if the worker labels as left at least 3 from 4 first pairs, -1 if at least 3 are labeled as right, and 0 otherwise. y_i is a binary variable with 1 if the worker selects left for the last pair on the page i and x_i^l, x_i^r are ground-truth labels for left and right photos from the third pair, I use the data from both pools A and B.

4.4 Side-by-side setup (real data)

The setup for real data is similar to the forward bias setup:

$$y_i = \beta_1 I_i^{treated} + \beta_2 x_i^l + \beta_3 x_i^r + \varepsilon_i$$

However, notice that the estimation of this regression is a bit unfair in the sense that the dataset is not full and one honeypot task on each page is missed.

5 Estimating the quality of data

5.1 Regression setup

Table 5: Regression setup statistics

setup	scope	MAE	α_K	avg gs	avg gt	avg time	#workers	#pages	avg treated_A	avg treated_B
regression	all tasks	10.3	0.7482	37	37	39	96	180	-	-
regression	last task on page	7.2	0.2699	36	36	-	96	180	0.1371	0.1953

In table 5, you can see some summary statistics for regression setup data. Here MAE is mean absolute error for the tasks, α_K is Krippendorff's alpha, avg gt is the average ground truth value, avg gs is the average guess value, avg time is the average time for page, workers is the number of workers, pages is the number of pages, and avg treated_X is the average value for treated_X indicator.

The average mean absolute error for both the last page scope and all tasks scope looks adequate for the crowd-labeling project. I also estimate the quality of data using Krippendorff's alpha (Hayes and Krippendorff (2007)) as a measure with $d(x, y) = (x - y)^2$ as a distance function. The value of 0.75 indicates that the quality of workers is good enough, and the value of 0.27 for only the last tasks indicates that the last task is significantly harder for the workers than the previous ones. The average value of treated indicators shows that despite the special settings, a large number of workers were not treated by my definition.

5.2 Side-by-side setups

Table 6: Regression setup statistics

setup	scope	avg acc	α_K	avg pref	avg gt	avg time	#workers	#pages	#treated = 1	#treated = -1
sbs real data	all tasks	0.7544	-	0.5007	0.5076	23	4091	83416	-	-
sbs real data	last task on page	0.7536	-	0.5017	0.5081		4091	83416	20843	20804
sbs forward bias	all tasks	0.6861	0.3714	0.6027	0.7778	42	63	120	-	-
sbs forward bias	last task on page	0.5583	0.3186	0.5583	1		63	120	53	23
sbs anti-anchoring	all tasks	0.72833	0.4527	0.6483	0.7633	63	80	120	-	-
sbs anti-anchoring	last task on page	0.5416	0.4341	0.5416	1		80	120	41	3

Table 6 has almost similar statistics for side-by-side setup. Here, avg accuracy is the average accuracy metric, avg pref is the average preference between right and left options (left is 1, right is 0), and avg gt is the average ground truth value with the same side encoding. treated = X shows the number of rows where the treated indicator is equal to X. The distance function for Krippendorff's alpha is $d(a, b) = I(a = b)$

All of the accuracy values for all tasks' scopes are quite high, which indicates that the workers performance is good and the fraction of cheaters is not so high. Notice that Pavlichenko and Ustalov (2021) use quality control settings with golden tasks in their project to ban the cheaters and that might be a reason why the accuracy for real data setup is higher than for other ones.

Krippendorff's alpha values for side-by-side setups are lower compared with regression setups, which is normal because I use a different distance function. For this function, such low values are expected because the chances of randomly selecting the correct answer for a side-by-side setup are high. Krippendorff's alpha is unavailable for real data setup because of the overlap of one worker per pair.

Also, notice that only a very small fraction of workers were treated in the real data setup, which might be good news for researchers who use side-by-side setups. With my special settings of the pools, the fraction of treated workers significantly grows.

6 Results

The regression results are presented in Table 7. The most interesting result was obtained in the regression setup. As you can see, only one of the treated indicators, treated_indicator_B is statistically significant. Moreover, true age is not significant, and the main impact of estimation is introduced by the constant variable. It means that the worker's guess initially has some constant anchor (probably depending on the worker's own age), and the true age of the person in the photo does not really impact the guess. This is not intuitive but normal result because the true age of workers for the last task on page is in the range between 30 and 40 years (see Section 3.1). In the real world sense, the expected difference of more than seven years for biased workers also looks important.

The reason why only one treated indicator is significant might be that the treated_B indicator is closer by its nature to the constant mean value and might strengthen its effect,

Table 7: Regressions results

	Regression	SbS (forward bias)	SbS (anti-anchoring bias)	SbS (real data)
	(1)	(2)	(3)	(4)
Model	OLS	Logit	Logit	Logit
Dependent variable	guessed age	guessed side	guessed side	guessed side
const	40.671***			
	(8.194)			
left_age		-0.276*	-0.274**	0.061***
		(0.150)	(0.134)	(0.000)
right_age		0.270*	0.265**	-0.061***
		(0.144)	(0.129)	(0.000)
treated_indicator		-0.244	0.264	0.007
		(0.251)	(0.374)	(0.012)
treated_indicator_A	-2.219			
	(2.187)			
treated_indicator_B	5.459***			
	(1.788)			
true_age	-0.141			
	(0.230)			
Observations	180	120	120	83416
R^2	0.071			
Adjusted R^2	0.055			
Pseudo R^2		0.029	0.036	0.252
Residual Std. Error	8.459 (df=176)			
F Statistic	4.194*** (df=3; 176)			

Note: *p<0.1; **p<0.05; ***p<0.01
Standard Errors are heteroscedasticity robust (HC1)

while treated_A indicator in some sense contrasted with constant estimation. In the real world sense, the expected difference of more then seven years for biased workers also looks important.

The indicators of side-by-side regressions are not significant in all of the cases. So, it indicates that the side-by-side setup is less affected by such biases, which agrees with my hypothesis. The reason why the absolute values of left and right age coefficients for a real data setup are lower is that it contains considerably more data points, which leads to a narrow confidence interval.

Note that the R^2 values are small for all of the regression results. It indicates that a large part of dependent variable variation is not explained by control variables. This is quite a normal situation for crowdsoursing datasets, which have a big variation in nature. The reason why R^2 for real data is bigger than other ones is that pairs in this dataset are easier (I constructed my dataset that way when the person in the pair has a close age).

Thus, the results show that for side-by-side setups, the anchoring bias effect is almost negligible. Even with my special pool setups, which were designed to incite bias, the fraction of treated workers is less than half of them, and in the real dataset, this fraction is less than 5% (table 6). So a side-by-side setup might be a good solution to struggle with the anchoring bias; however, it might be expensive for full-scale projects because it requires more tasks for different combinations of pairs than a straightforward estimation. The obtained result for the regression setup should not be interpreted as a proof of the anchoring bias problem for all straightforward setups; it only shows that with poorly designed projects, the significant influence of anchoring bias is possible.

7 Discussion

In this section, I outline the limitations of my work and suggest some possible ways for further research.

7.1 Internal validity

First of all, I do not have enough information about the workers who label my data; their age, gender and skill estimation would be helpful as additional control variables. I have a hypothesis that the worker might also have his own age as an anchor for my type of tasks, which might create an omitted variable bias. Another possible good control variable is the gender of person in the photo in the task. This information is provided in original Toloka-IMDB-wiki dataset, but Pavlichenko and Ustalov (2021) do not provide it for their subsample of data. Also, they notice that the gender label for original datasets are sometimes wrong.

Overall, the quality of Toloka-IMDB-Wiki dataset is not perfect. I encounter examples where the ground truth label looks implausible, so there might be some mistakes in data. However, these cases are relatively rare, so I trust the results with keeping this problem in mind.

Pavlichenko and Ustalov (2021) have all tasks labeled in overlap 1, so it is harder to estimate the quality of data and it make impossible to calculate Krippendorff's alpha and some other standard metrics. My tasks were labeled with overlap 2, which is also might be increased to get a better estimation. For further research it might be an interesting idea to estimate a similar setups with high overlap and a lower number of unique tasks with task-level fixed effects.

Another possible problem is cheating on the tasks. I do not add quality control rules to my projects to increase the probability of workers being treated well. However, it might increase the number of cheaters. I estimate that the damage from cheaters in my projects is not high because one worker can label only three pages. Also, the compensation for tasks might be too low for some workers, and they might do tasks inaccurately for this reason.

7.2 External validity

The main problem with external validity is that my data is limited to only one type of task (age recognition), and for different types of tasks, the results might essentially differ. I suppose that the results for the side-by-side setup can be used as proof that the anchoring bias is not a problem for this design. However, the results for the regression setup are less credible, and I don't recommend blindly expanding it on other projects. Thus, one possible way for further research is to estimate the bias for regression setups

on different datasets and different types of tasks. Also, notice that I estimate only the anchoring effect, but the workers might be exposed to different types of behavior bias. Finally, I want to pay attention to the fact that different aggregation methods might be prone to the anchoring bias less or more, and estimating the quality of aggregates by different methods with biased answers is another good topic for further research.

8 Conclusion

Turning back to the posed research question, I conclude that the anchoring effect is negligible in side-by-side settings. The question of the presence of anchoring bias in regression-like setups depends on task design. However, in this paper, I show that it might exist in a poorly designed project.

Overall, I expect that the problem of human bias in the machine learning datasets will become more and more important in the near future because the machine learning models already have better quality than humans in some fields. The requirements for crowd workers have drastically increased in recent years, and this trend continues.

On the other hand, the research on crowd-labeling data is still mostly in the computer science field. In this work, I try to show that it might also be an interesting research field for economists, and I hope that my work will be one of the first but not the last attempts to get an economic view on this field.

References

Bernstein, M. S., G. Little, R. C. Miller, B. Hartmann, M. S. Ackerman, D. R. Karger, D. Crowell, and K. Panovich (2015, jul). Soylent: A word processor with a crowd inside. *Commun. ACM 58*(8), 85–94.

Bradley, R. A. and M. E. Terry (1952). Rank analysis of incomplete block designs: I. the method of paired comparisons. *Biometrika 39*(3/4), 324–345.

Bugakova, N., V. Fedorova, G. Gusev, and A. Drutsa (2019). Aggregation of pairwise comparisons with reduction of biases. *arXiv preprint arXiv:1906.03711*.

Dawid, A. P. and A. M. Skene (1979). Maximum likelihood estimation of observer error-rates using the em algorithm. *Journal of the Royal Statistical Society. Series C (Applied Statistics) 28*(1), 20–28.

Day, R. L. (1969). Position bias in paired product tests.

Draws, T., A. Rieger, O. Inel, U. Gadiraju, and N. Tintarev (2021). A checklist to combat cognitive biases in crowdsourcing. In *Proceedings of the AAAI conference on human computation and crowdsourcing*, Volume 9, pp. 48–59.

Franklin, M. J., D. Kossmann, T. Kraska, S. Ramesh, and R. Xin (2011). Crowddb: Answering queries with crowdsourcing. In *Proceedings of the 2011 ACM SIGMOD International Conference on Management of Data*, SIGMOD '11, New York, NY, USA, pp. 61–72. Association for Computing Machinery.

Gebru, T., J. Morgenstern, B. Vecchione, J. W. Vaughan, H. Wallach, H. D. I. au2, and K. Crawford (2021). Datasheets for datasets.

Hayes, A. F. and K. Krippendorff (2007). Answering the call for a standard reliability measure for coding data. *Communication methods and measures 1*(1), 77–89.

Kahneman, D., D. Lovallo, and O. Sibony (2011). Before you make that big decision.

Kahneman, D. and A. Tversky (1979). Prospect theory: An analysis of decision under risk. *Econometrica 47*(2), 263–292.

Liu, R., S. Jecmen, V. Conitzer, F. Fang, and N. B. Shah (2023). Testing for reviewer anchoring in peer review: A randomized controlled trial. *arXiv preprint arXiv:2307.05443*.

Pavlichenko, N. and D. Ustalov (2021). Imdb-wiki-sbs: An evaluation dataset for crowdsourced pairwise comparisons. *arXiv preprint arXiv:2110.14990*.

Rothe, R., R. Timofte, and L. Van Gool (2018). Deep expectation of real and apparent age from a single image without facial landmarks. *International Journal of Computer Vision 126*(2-4), 144–157.

Saab, F., I. H. Elhajj, A. Kayssi, and A. Chehab (2019). Modelling cognitive bias in crowdsourcing systems. *Cognitive Systems Research 58*, 1–18.

Shah, N. B., S. Balakrishnan, J. Bradley, A. Parekh, K. Ramch, M. J. Wainwright, et al. (2016). Estimation from pairwise comparisons: Sharp minimax bounds with topology dependence. *Journal of Machine Learning Research 17*(58), 1–47.

Shah, N. B. and M. J. Wainwright (2018). Simple, robust and optimal ranking from pairwise comparisons. *Journal of machine learning research 18*(199), 1–38.

Surhone, L., M. Timpledon, and S. Marseken (2010). *The Wisdom of Crowds*. VDM Publishing.

Tversky, A. and D. Kahneman (1973). Availability: A heuristic for judging frequency and probability. *Cognitive psychology 5*(2), 207–232.

Tversky, A. and D. Kahneman (1992). Advances in prospect theory: Cumulative representation of uncertainty. *Journal of Risk and uncertainty 5*, 297–323.

von Ahn, L., B. Maurer, C. McMillen, D. Abraham, and M. Blum (2008). recaptcha: Human-based character recognition via web security measures. *Science 321*(5895), 1465–1468.

Wang, J. and N. B. Shah (2018). Your 2 is my 1, your 3 is my 9: Handling arbitrary miscalibrations in ratings. *arXiv preprint arXiv:1806.05085*.

Whitehill, J., T.-f. Wu, J. Bergsma, J. Movellan, and P. Ruvolo (2009). Whose vote should count more: Optimal integration of labels from labelers of unknown expertise. *Advances in neural information processing systems 22*.

Appendix A

Toloka task interface example

Appendix B

My dataset

I provide my custom dataset for further research.

It is available through the link https://github.com/SergeyIsh/Anchoring_bias_in_crowd.

The datasets for three of my setups include the input values column, where you can find the link for the photo and ground truth values, the output values column with the worker's answers and user id of the worker.