DECODE SURVEILLANCE NYC

METHODOLOGY

AMNESTY INTERNATIONAL

Amnesty International is a movement of 10 million people which mobilizes the humanity in everyone and campaigns for change so we can all enjoy our human rights. Our vision is of a world where those in power keep their promises, respect international law and are held to account. We are independent of any government, political ideology, economic interest or religion and are funded mainly by our membership and individual donations. We believe that acting in solidarity and compassion with people everywhere can change our societies for the better.

First published in 2022
by Amnesty International Ltd
Peter Benenson House, 1 Easton Street
London WC1X 0DW, UK

Index: AMR 51/5205/2022
Original language: English

amnesty.org

Cover photo: *Eliana Rogers. © 2021 Amnesty International*

AMNESTY INTERNATIONAL

CONTENTS

ANALYSIS OF CROWDSOURCED DATA

1.1 FINDING SUMMARIES

We summarize here the counts of the cameras found by the decoders, after the data processing described in this methodology.

The numbers for the boroughs of Manhattan, The Bronx and Brooklyn differ slightly away from those published in the preliminary analysis we described this summer; these minor changes are due to having matched the latitude/longitude coordinates to the exact panorama observed rather than to the theoretical intersection location, and redone the allocation per Borough based on NYC.gov shapefiles.

1.1.1 CAMERA COUNTS

Figure 1: Camera counts

What are we counting?	Total (calculated by summing the median votes)	Percentage (of the sum)
Cameras attached buildings	22,133	86.7%
Cameras attached street lights, traffic signals or poles	3,317	13.0%
Cameras attached something else	86	0.3%
All cameras	25,536 (if we sum the categories above)	100%

When we count only cameras attached to streetlights, traffic signals or poles that are also dome type, and so most likely to be the New York Police Department (NYPD) or the Department of Transportation (DOT):

Cameras attached to street lights, traffic signals or poles AND are dome type	2,266	8.87%

1.1.2 FOCUS ON INTERSECTIONS

Figure 2. Intersections

Total number of intersections analysed	43,406
Total number of intersections with at least one camera	14,100 (32.49%)
Total number of intersections with at least one camera attached to street light, traffic signal or pole	2,442
Total number of intersections with at least one camera attached to street lights, traffic signals, poles that is dome type	1,614

1.1.3 FOCUS ON BOROUGHS

Figure 3: Camera counts by borough

	Manhattan	Brooklyn	The Bronx	Staten Island	Queens	Total
All cameras	3,948	9,232	3,737	1,035	7,580	25,532
Attached streetlights, traffic signals or poles	724 (18.3%)	871	456	178	1,088	3,317

Figure 4: Camera-per-intersection by borough

	Manhattan	Brooklyn	The Bronx	Staten Island	Queens	Total
All cameras	1.00	0.89	0.60	0.16	0.46	3.11
Attached streetlights, traffic signals or poles	0.18	0.08	0.07	0.03	0.07	0.43

1.2 PROJECT DESIGN

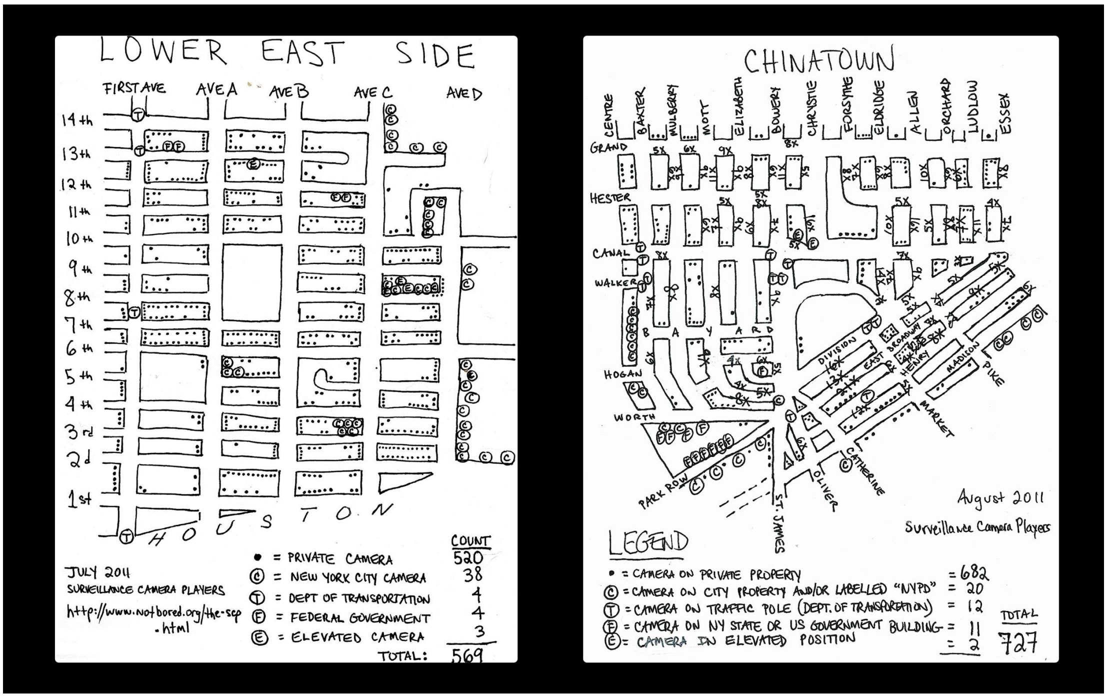
1.2.1 SITUATING AND SCOPING THE WORK

Our first task was to situate the planned research in a history of anti-surveillance activism in the USA, in particular New York City, and, critically, to learn from the work of others. To do this we asked scoping questions, including: Who has attempted camera surveys before and what we can learn from them? Has anyone attempted an online survey of surveillance cameras and if not, why not? What data on surveillance cameras was already in the public domain?

To avoid duplicating existing contemporary data on camera location, the project team assessed commercial surveillance camera data, such as the locations of Link NYC kiosks, and open government data such as the Department of Transport (DOT) camera locations.

We also looked at historical data. The largest citizen-led survey of cameras, by far, was the 1998 NYC Surveillance Camera Project, and its 2006 follow-up, Who's Watching. Both studies were conducted by the New York Civil Liberties Union (NYCLU) and used walking surveys.

Similarly, walking surveys were used by the Surveillance Camera Players, an artists' collective that, in 2011, published a series of maps of New York City with hand-drawn locations of surveillance cameras.



Hand drawn maps of the surveillance cameras in the Lower East Side and Chinatown, created by the Surveillance Camera Players in 2011 © notbored.org

Inspired by the walking surveys, we decided to use Google Street View's API to take the methodology online – widening access to volunteers and allowing us to cover all of NYC.

Bing Streetside and the crowdsourced alternative, Mapillary, were also considered as possible image sources. However, only Google Street View had the geographic coverage and up to date imagery needed for the survey to produce meaningful results. Later analysis, conducted a month after Decode Surveillance NYC launched, found that 90% of Google Street View panoramas used in the project were taken in 2018 or later (82% were taken in 2019), while the earliest was taken in 2007.

The areas in blue have been imaged by Google Street View © Google Earth

1.2.2 MODELLING THREATS TO REDUCE RISK

The novelty of the methodology and scale of the proposed research meant that the project carried significant risk. Two threat modelling workshops helped us tease out discrete risks and identify mitigation strategies. Below are notes from the section on volunteer-related risks, other areas included technical, legal, malicious actors, and privacy.

When completing tasks, volunteers are unsure **which cameras are public**.	• Create a Help section in the task presenter • User test • When the project is live, review the level of "agreement" between decoders
When competing tasks, volunteers at different locations count the same camera causing **duplicates** in the data.	• Limit zoom and user test • Account for the scenario in the onboarding tutorial and user test • When post processing of the data, use the proximity of locations, camera classification, and/or URLs to identify duplicates in the end data
Volunteers to **not agree on the number of cameras**. This pushes up the total tasks that need to be served before the assignment/asset is completed. As a result, progress is slow, and the cost of the API increased as we repeat tasks.	• The completion criterion is the number of submissions. Level of agreement does not affect completion.
Volunteers **do not reliably categorise camera types** meaning it takes more users than expected to reach the matching criteria. This pushes up the total tasks that need to be served before the assignment/asset is completed. As a result, progress is slow, and the cost of the API increased as we repeat tasks.	• Camera categorisation does not determine the completion criteria. • Have a category that is "Don't know or other camera type"

Threat modelling workshop notes, 2021 © Amnesty International

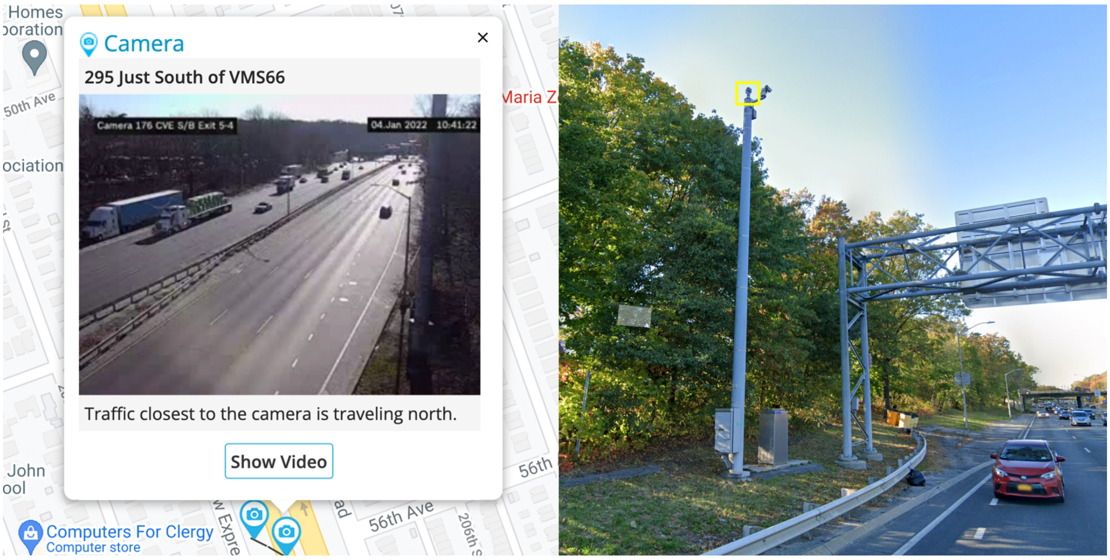
Unlike past Amnesty Decoders projects, which stored the assets used for micro tasking locally, Decode Surveillance NYC was reliant on paid access to Dynamic Street View API. To ensure we covered NYC within budget and to schedule (approximately two months), the team decided to focus on the city's more than 45K traffic intersections.

Traffic intersections afford views of multiple streets, making them strategic locations for the surveillance devices to be installed. Therefore, focusing on intersections rather than random locations makes it more likely to find surveillance cameras.

1.2.3 COLLECTING THE LIST OF INTERSECTIONS

OpenStreetMap (OSM) in conjunction with GIS software was used to generate a geospatial dataset of traffic intersections in New York City. Another dataset was created using the same methodology but using New York City Department of City Planning (DCP) data (instead of OSM) as the source. The OSM dataset was chosen because the midpoints of intersections were more centrally located when used with Google products, and the data contained fewer false positives, that is intersections with multiple coordinates.

During the earlier scoping exercise, Department of Transport (DOT) cameras on expressways were identified as positioned relatively high compared to other DOT or NYPD cameras, and so less likely to produce images at a resolution compatible with facial recognition software.[1] As a result, the team made the decision to exclude expressways from the final dataset, and so the analysis.



Left: A DOT public camera feed of Clearview Expressway (Interstate 295) accessed via 511ny.org © 511 New York. Right: The same DOP camera as seen in a Google Street View image dated October 2019 © Google Street View

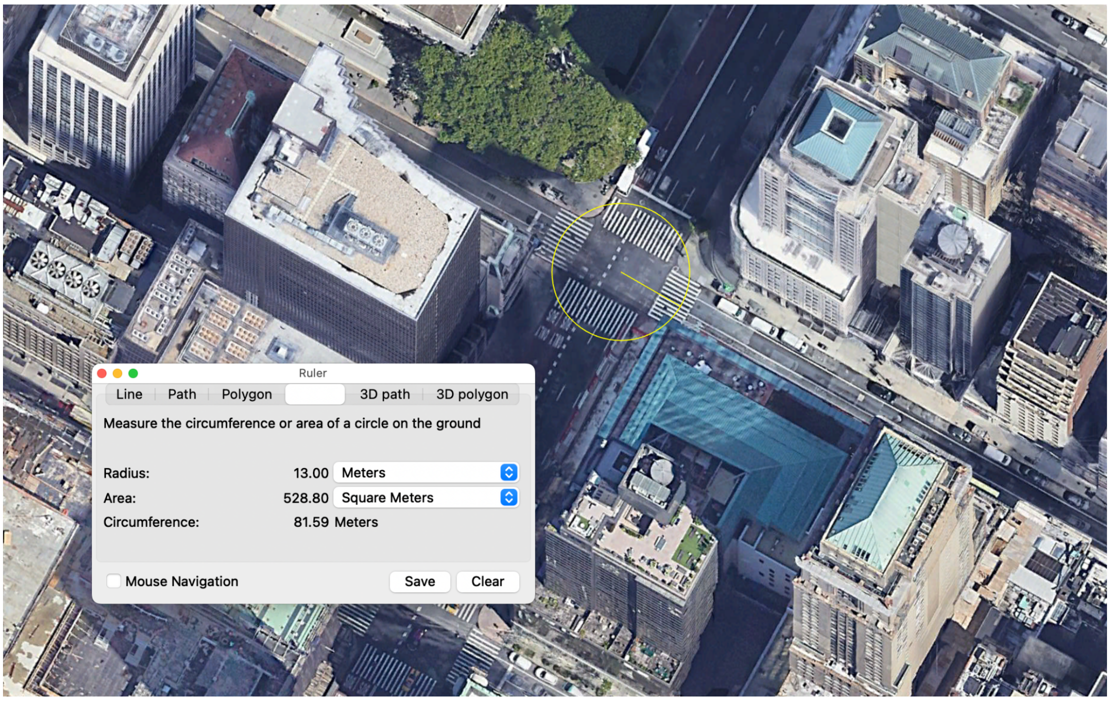
The data was then cleaned manually to remove duplicates as well as stray points such as coordinates in parks and cemeteries.

To avoid user created images (photos uploaded to the Google by people) and indoor images, the following criteria were used to select the panoramas:

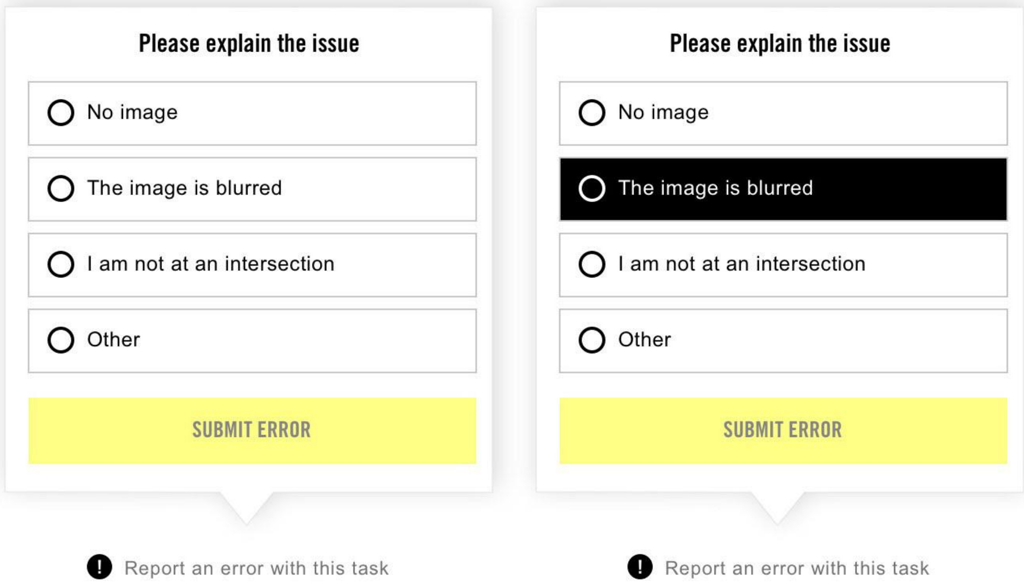
```
preference: window.google.maps.StreetViewPreference.BEST,

source: window.google.maps.StreetViewSource.OUTDOOR
```

[1] Additional research by Amnesty International in 2021 found that the minimum resolution required by commercially available face recognition software can be as little as approximately 24 px /face width. See: Julien Cornebise, Swetha Pillai, Martyna, Marciniak, Sophie Dyer, Citizen Evidence Lab, November 2021. https://citizenevidence.org/2021/11/17/decode-surveillance-early-analysis, "Decode Surveillance NYC early analysis" (accessed 7 January 2021)

A maximum distance of 13 meters from the traffic intersection coordinates was also set. If no panorama was available within 13 meters, the volunteer could report "no image" from the "Report an error" menu.



An intersection in Manhattan with a 13-metre radius overlay © Google Earth Pro



The options given to volunteers when selected "Report an error", 2021 © Amnesty International

1.2.4 MICRO TASKING DESIGN

It was important for the research to be able to classify cameras as publicly or privately owned. Without access to this information and with a view to keeping the micro tasking questions as simple as possible, the team decided to use what cameras were attached to as a proxy for public or private ownership.

Volunteers were asked to find all surveillance cameras and record what they were attached to. Three multiple choice options were given:

1. Streetlight, traffic signal or pole
2. Building
3. Something else

If volunteers selected option 1. "Streetlight, traffic signal or pole", they were asked to identify the camera type. We chose three visually distinct, high-level categories:

1. Dome or PTZ camera*
2. Bullet camera
3. Unknown or Other

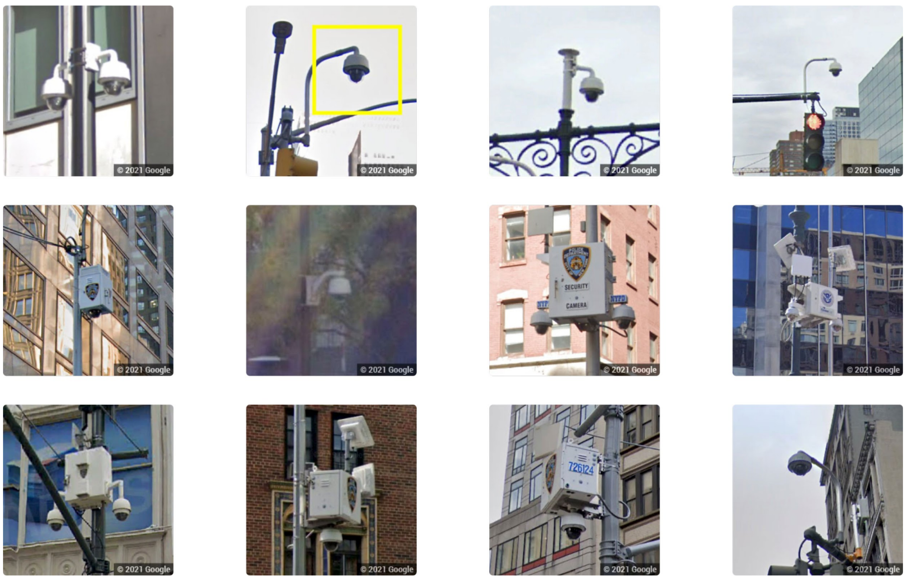
*PTZ is an acronym for Pan Tilt Zoom

The answers were then used as a proxy for public or private ownership. For example, cameras attached to traffic signals or streetlights were assumed to be most likely owned by a government agency. In this sub-category, dome or PTZ cameras were of particular interest as they were likely to be NYPD Argus cameras. Whereas cameras attached to buildings were assumed to be privately owned and so of less relevance to the research, although we recognized that a minority would be attached to federal buildings.

1.2.5 SUPPORT AND ACCESSIBILITY

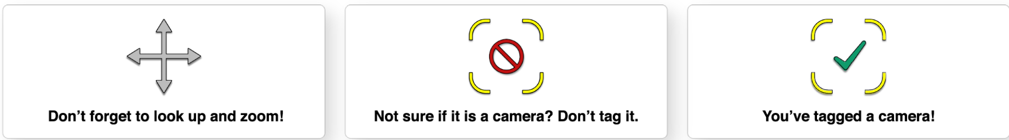
The quality of the data collected rested on successfully onboarding a diverse cohort of volunteers, many lacking pre-existing knowledge of the subject. For the micro tasking to produce reliable data we needed volunteers to give consistent answers or, to put it another way, agree with each other the majority of the time.

The multiple-choice questions were illustrated by pictograms drawn by an illustrator. The illustrations were used throughout the project site to reinforce instructions, and as visual aids for volunteers who were not fluent in English or would otherwise benefit.



The visual help guide featured images of cameras from each category, as well as objects that could be misidentified such as streetlamps, cell towers, and strobe detectors © Google Street View

In addition, the team produced a tutorial video filmed on location in New York city and a visual help guide that was always accessible to logged in volunteers via a sidebar. For the first 20 tasks, occasional pop-up notifications such as, “Don’t forget to look up and zoom in” and “Not sure if it is a camera? Don’t tag it” repeated tips from the tutorial video.



Pop-up notifications repeated tips from the tutorial and encouraged volunteers, 2021 © Amnesty International

Similarly, to previous Amnesty Decoders, volunteers could also access a moderated forum. When logged in, volunteers could access the forum via the navigation bar at the top of the page or by flagging the task for discussion. Flagging an assignment opened a new discussion thread (if the traffic intersection had not been flagged before) or added to an existing thread (if the traffic intersection had already been flagged by another volunteer) in the forum. Volunteers could then add a comment or question as well as upload supporting materials such as screenshots and links.

The forum was designed to a space for:

- peer-to-peer support and learning. Peer-to-peer support via the forum has been critical to the success of Amnesty Decoders since its launch in 2016 by scaling up the support Amnesty can offer, while empowering volunteers to take ownership of the project, share lessons learned, and build community and knowledge around the issues at stake;
- positive feedback and encouragement to volunteers. The forum is a space for volunteers, from new to returning, to share achievements and receive encouragement. Near real time feedback reinforces the feeling of being part of a “live” research project;
- Amnesty to communicate with volunteers. For example, give project updates such as milestones, and share news of events such as an AMA (Ask Me Anything) with a researcher;
- the project team to gain real time feedback about volunteers’ experiences including areas of difficulty, common frustrations, or mistakes, as well as bugs. This allows Amnesty to make improvements such as tweaks to the user interface or data collected while the project is live.

The forum was moderated by the project team, supported by a group of 10–15 volunteer moderators. Our volunteer forum moderators were based in different countries, and ranged from students to workers, with diverse abilities and experiences.

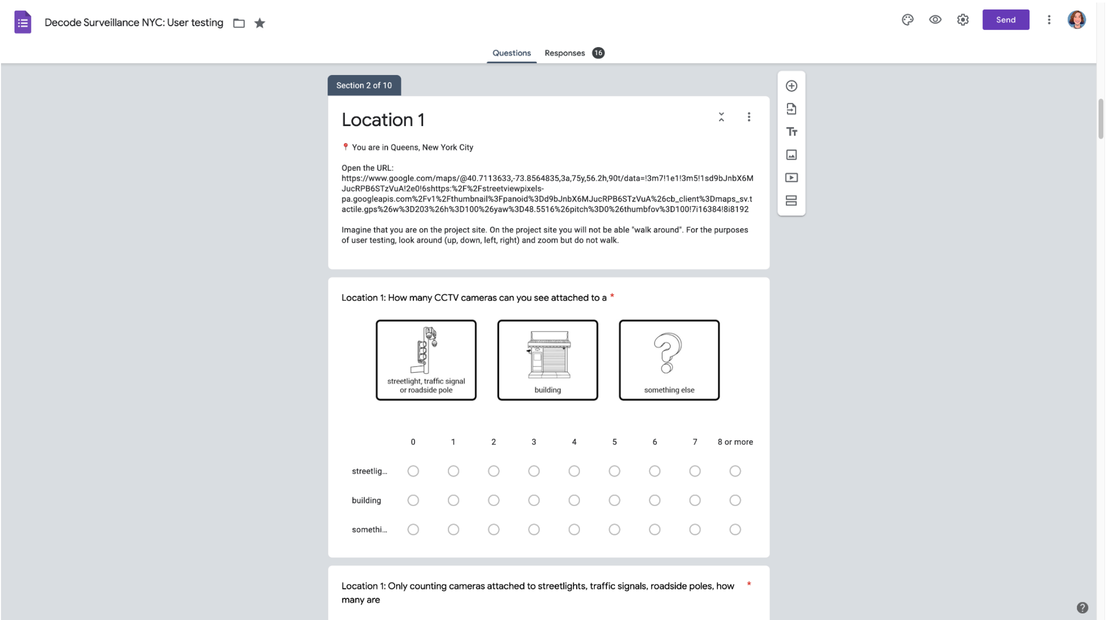
Combined the video tutorial, visual Help Guide and moderated forum ensured that volunteers could access support at every stage of the micro tasking.

Although the materials for the wider Ban the Scan campaign were produced in English, Spanish, and Arabic, because of the challenges of moderating the forum, Amnesty was only able to deliver Decode Surveillance in English. Although the project team worked to simplify and reduce the written instruction as well as use illustrations, we recognize that language was a barrier to access.

Likewise, although the website was designed mobile first, the connectivity requirements limited access to those with an internet connection and the data to participate. To mitigate this in part, the site was designed and developed so that it did not require large bandwidth or the latest operating system to load.

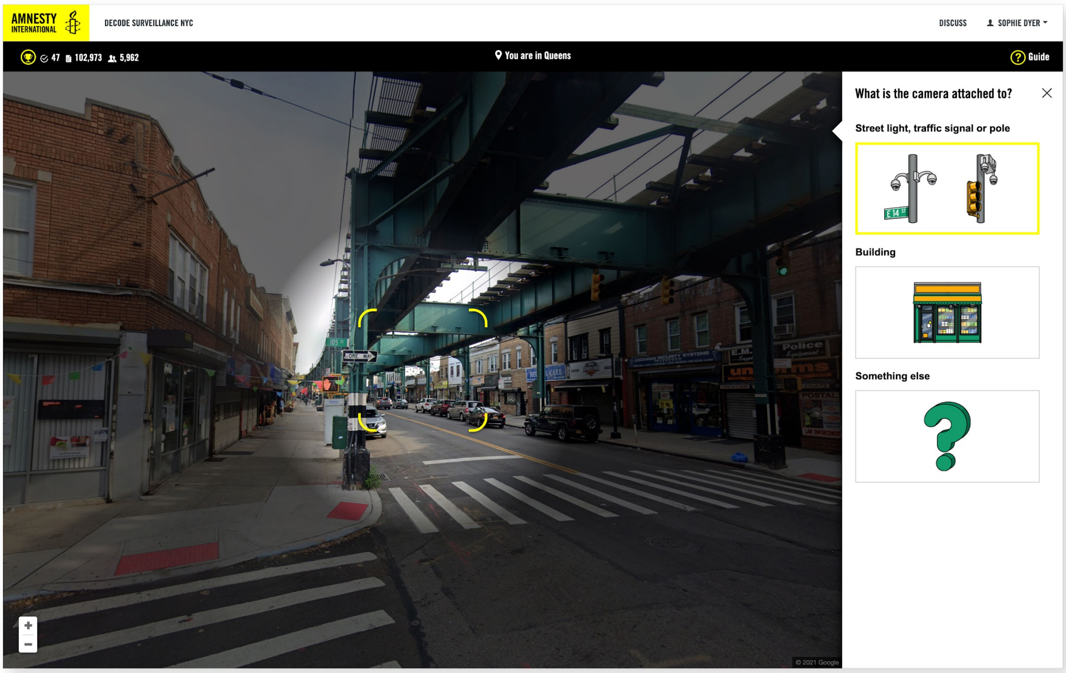
1.2.6 PILOT STUDY

A pilot study was run with 15 volunteers on a prototype of the user interface. The variability of answers indicated the need for multiple decoders on each intersection. Measures of the average labelling time per intersection, scaled by the estimated likely engagement of decoders on the basis of previous campaigns, confirmed that it was reasonably possible to label each intersection by 3 different graders, but that going beyond that amount of replication would risk being unable to cover the total surface of NYC.



The speed of project build meant that it was not possible to conduct user testing on beta versions of the site. Instead, the team used interviews, surveys, and workshops, 2021 © Amnesty International

1.2.7 USER INTERFACE AND TESTING



Final design for user interface on desktop, 2021 © Amnesty International

User testing steered many design decisions. Critically, it helped us identify which instructions volunteers would likely ignore, such as the command to only count cameras attached to streetlights, traffic signals or other roadside poles. User testing revealed that, regardless of how this instruction was delivered, most volunteers counted all cameras including those on buildings.

1.2.8 GEOREFERENCING PANORAMAS

The intersection dataset came with latitude/longitude coordinates, that were fed into Google Street View API upon connection of the first decoder seeing this task, to get the nearest panorama available in Street View. The platform recorded the unique identifier of that panorama and subsequently reused it for every decoder visiting that intersection, ensuring that all decoders saw the same image.

1.2.9 DATA RECORDED

Upon being shown a panorama of an intersection, the decoders were free to rotate around a left-right 360 degree axis, look up and down 90 degrees, zoom forward, and click on any part of the image to "tag" a camera. The platform recorded the spherical coordinates of the decoder's point of view in the panorama, alongside the decoder's observation of the type of attachment of the camera (street furniture/building/unknown), and for those cameras labelled as attached to street furniture, the observation of the type of camera (bullet/dome PTZ/unknown). The platform also stored a pseudonymized unique ID of the decoder and the date/time of the submission.

1.3 DATA CLEANING, EXPLORATORY DATA ANALYSIS, AND QUALITY ASSURANCE

1.3.1 INTERSECTIONS CLEANING

After loading the labelled dataset, we replaced the intersection's original latitude and longitude with that of the matched panorama, as provided by Google Street View metadata. This established the panorama (i.e., the photo labelled by the decoders) as the geographical reference and guaranteed that the geographic coordinates of the labels were matching what the decoders actually saw, removing any potential offset between intersection and panorama.

In the rare occasions where two or more intersections were matched to a common panorama, we merged those into a single intersection at the location of the panorama, therefore these intersections had more decoders than other intersections. We then only kept 3 decoders (sampled uniformly at random) at each such resulting merged intersection, to ensure comparable variance of observations with all other intersections.

For the preliminary results published in the 3 June 2021 press release, we aggregated the intersections by their Neighbourhood Tabulation Areas (NTAs), with arbitrary tie-breaking at boundaries. NTAs are a NYC Planning aggregation of census tracts that approximately represents neighbourhoods — see NYC Population: Geographic Reference for more details on that geographic unit. This allowed mapping and geographic interpretation.

1.3.2 AGGREGATION AND EXPLORATORY DATA ANALYSIS

Aggregating the three decoders' answers over an intersection can be done in multiple ways. Amnesty International tries to be as rigorous and conservative in its numbers as possible. We therefore decided to use the median of each intersection's three decoders' counts, rather than the mean. The median is indeed less sensitive to outliers than the mean – and those outliers would mostly be overcounts, because the number of observed cameras is lower-bounded by 0. Therefore, by using the median at each intersection we favour undercounts rather than overcounts, while still keeping a better representation than if we were to take the minimum count.

As an exploratory analysis, we also compared the amount of disagreement, i.e. three unanimous decoders, or 1 vs 2, or 3 disagreeing decoders; split by median and camera type, i.e. public vs private.

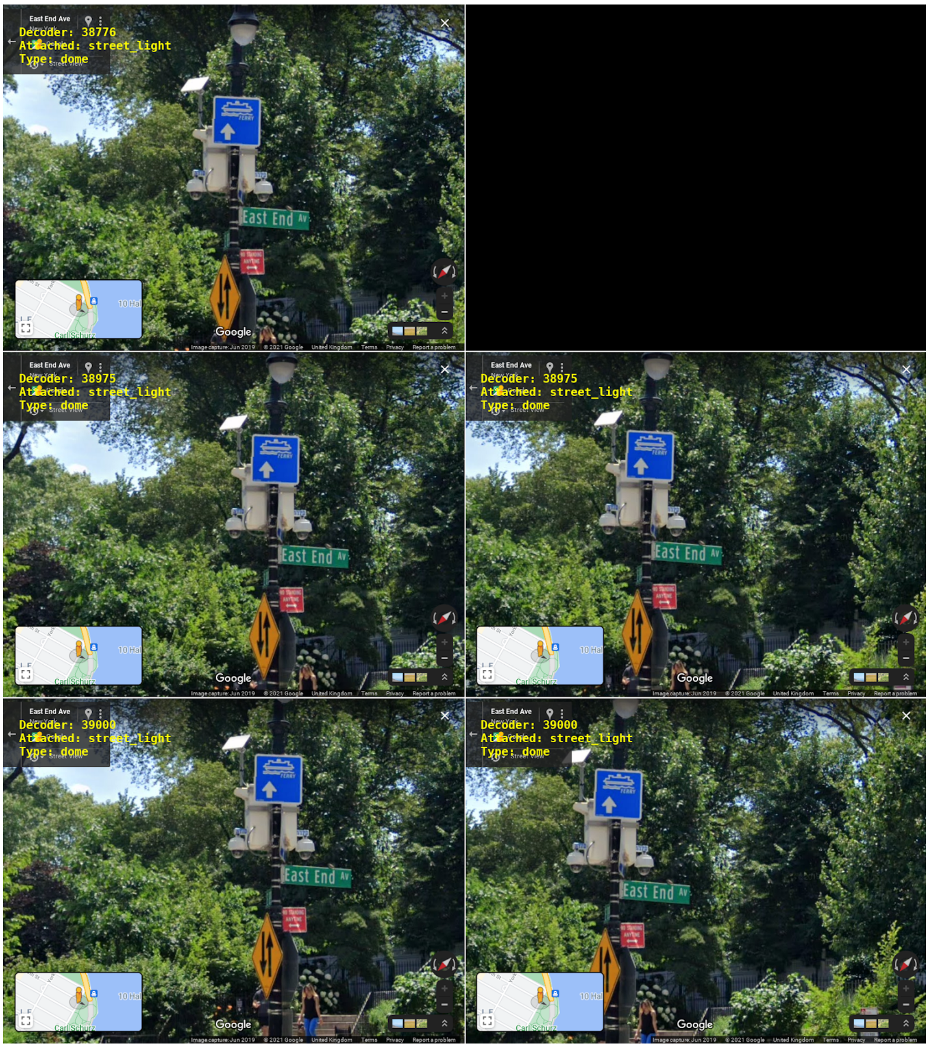
This confirmed the intuitive result that the higher the median (and hence likely the underlying number of cameras and complexity of the intersection), the higher the amount of disagreement (as the potential for missed cameras or false detection increases).

1.4 QUALITY ASSURANCE

In order to assess the quality of the resulting numbers, we proceeded to both a qualitative and a quantitative review.

1.4.1 QUALITATIVE REVIEW: VISUAL INSPECTION

The qualitative review served as a first visual inspection. It consisted of 100 completed intersections selected specially to cover a wide range of disagreement. We extracted the cameras found by the decoders into a mosaic for easy review by a trusted expert, as shown below.



Mosaics provided to the expert for the qualitative review. Each mosaic covers an intersection, and row shows images of all the tags from one decoder at one intersection. Tags are sorted by angle from the North for easier comparison. Black tiles serve as padding when the decoder tagged fewer locations than another decoder in the image.

This expert, chosen as one of the most involved volunteers on the forum, and who had participated in several Decoders campaigns, was then briefed by the head of the project, and they reviewed all found cameras on these intersections as well as checking the intersections directly in Street View.

We asked the expert to make notes of any repeated patterns of mistakes made by the decoders as a whole, to check for any glaring systematic pattern, and understand possible failure modes. This qualitative review confirmed that Decode Surveillance is the most difficult Decoders task undertaken, as there are quite a few different error modes. However, upon inspection, the median seemed to filter out the most egregious mistakes, as it was rare to see two decoders out of three err significantly.

1.4.2 QUANTITATIVE REVIEW: QUANTIFYING TRUSTWORTHINESS

The quantitative review that followed aimed to put actual numbers on the amount of disagreement and verify that we did indeed have an undercount.

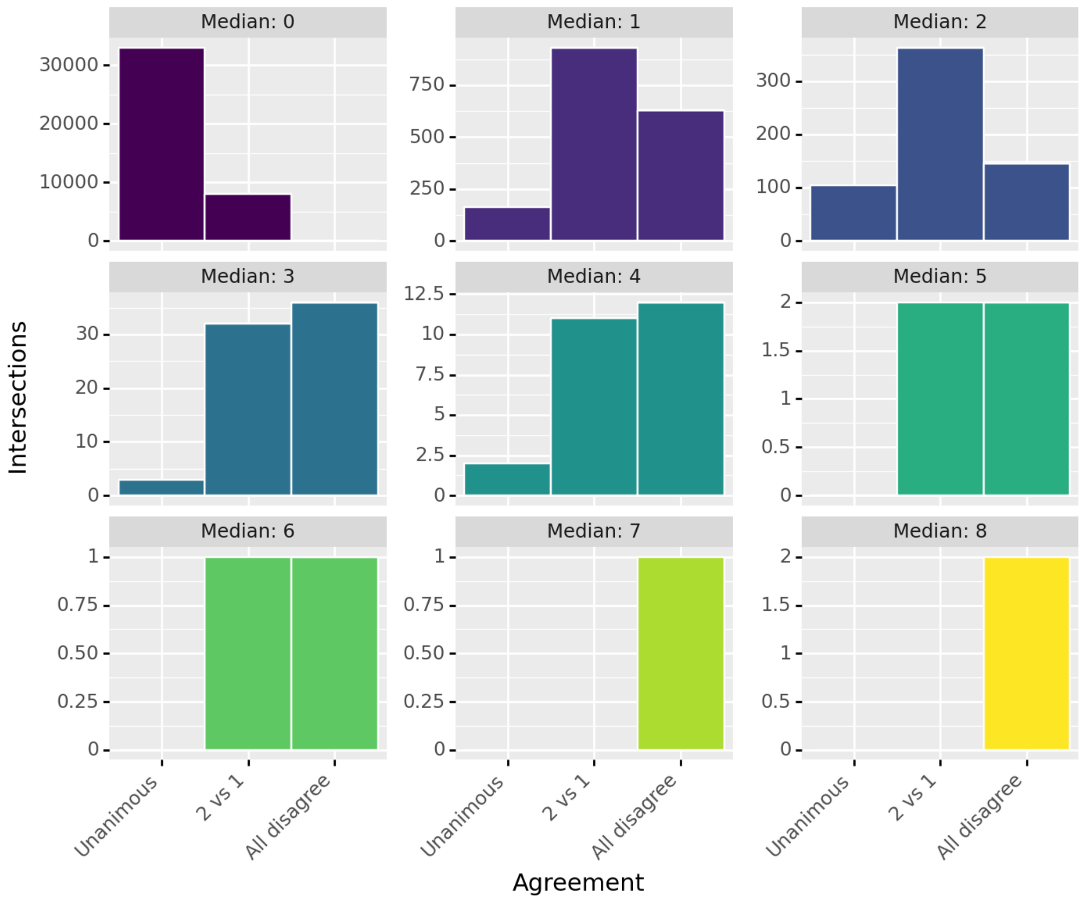
SELECTION OF INTERSECTIONS WITH STRATIFICATION AND IMPORTANCE SAMPLING

We used stratification and importance sampling to select 104 completed intersections with 8 typical combinations of median and number of disagreements, for cameras attached to streetlights, poles, or traffic lights, i.e., our proxy for public cameras:

- median of 0 cameras counted and unanimous agreement of all 3 graders,
- median of 0 cameras counted and 2-vs-1 disagreement,
- median of 1 camera and unanimous agreement,
- median of 1 camera counted and 2-vs-1 disagreement,
- median of 1 camera counted and all 3 decoders disagreeing,
- median of 2 cameras counted and unanimous agreement,
- median of 2 cameras counted and 2-vs-1 disagreement,
- median of 2 cameras counted and all 3 decoders disagreeing.

Please note that it is not possible to have a median of 0 if all 3 decoders disagree, hence only 8 classes. This left aside the few intersections with a median of 3 or more public cameras counted, which are few, as can be seen on the figure below.

Figure 5: Distribution of decoders' medians and agreements, public cameras



Distribution of the strata used in the stratification and importance sampling for the quantitative analysis. Each panel represents all intersections with a given median count of public cameras. Each bar represents the number of intersections with a given type of disagreement within this panel. Note the varying Y-axis, which shows that we capture the vast majority of intersections by reviewing only intersections with a median of 0, 1, and 2 public cameras. Only the first 8 strata were sampled for the quantitative review, i.e., all bars on the first row of panels.

We sampled 13 intersections of each of the above 8 strata. We emphasize that we are using importance sampling: while we over-sample some classes to reduce the variance of the estimator, we do accordingly reweight each sample in the estimator to obtain unbiased estimates according to the population's distribution.

REFERENCE TRUTH WITH 3 EXPERTS AND A META-EXPERT

To obtain a reference truth, we asked 3 experts to review these 104 intersections, and then had the project lead review the intersections on which these decoders were not unanimous. These 3 experts were chosen as some of the most active decoders on the forum, who provided particularly helpful answers to other decoders, denoting a deep understanding of the task and its subtleties, as well as a strong motivation to help. They kindly accepted to be assigned to this extra task of quantitative reviewing. The meta-review of their disagreements by the project lead helped ensure the highest quality of the counts, at the price of a workload that would have been unsustainable over a larger sample. This provides as close to a "ground truth" as possible on these 104 intersections.

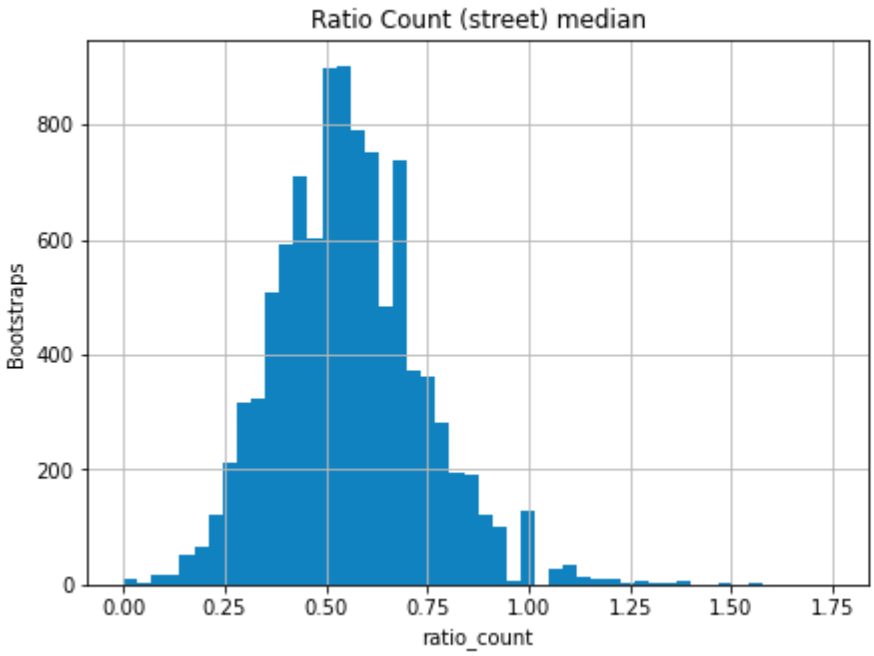
COMPARISON OF REFERENCE TO DECODERS USING WEIGHTED BOOTSTRAP

We then compared the decoders' counts to the experts.

The most important check is the ratio of the overall total count by the decoders, i.e., the sum over all intersections of the medians over three decoders, divided by the total count by the reference. To ensure a conservative analysis, this ratio should be lower than 1.0: decoders seeing fewer cameras than experts. Moreover, given the limited number of intersections with reference (104), we want to obtain credibility intervals on this ratio. We achieve this by bootstrapping the intersections, using a weighted bootstrap taking into account the importance sampling weights.

The figure below depicts the resulting empirical distribution of the bootstrapped estimates of the ratio of the sum of medians by the decoders divided by the sum of the reference counts, for the cameras mounted to streetlights, traffic signals or poles, i.e., our proxy for public cameras. It is quite spread out, meaning there is a high variance to this ratio, as expected from the low number of intersections in the quantitative review. However, most of its mass is way below 1: its 95% credibility interval between the 2.5%-percentile and the 97.5% percentile is [0.23, 1]. This means that, based on the quantitative review, we can say with 95% certainty the decoders (as measured by the sum of the medians) find between 23% and 100% of the reference cameras.

Figure 6: Comparison of median of decoders vs experts on public cameras

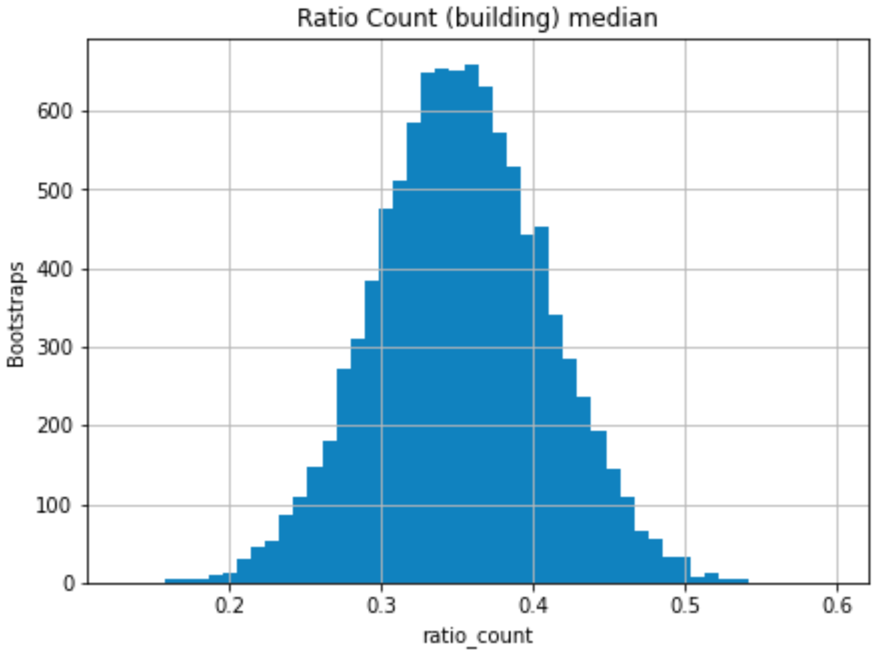


Histogram of the bootstrapped distribution of the ratio of total counts (sum over intersections of medians over decoders) by the decoders divided by total reference count, for cameras attached to streetlights, traffic signals or poles.

While this interval is wide, very importantly, it provides a very high level of confidence that the total from the decoders in our whole study is an undercount of the actual ground truth.

We produced the same analysis for the cameras attached to buildings, our proxy for private cameras, with even clearer-cut results, as visualized below:

Figure 7: Comparison of median of decoders vs experts on private cameras



Histogram of the bootstrapped distribution of the ratio of total counts (sum over intersections of medians over decoders) by the decoders divided by total reference count, for cameras attached to buildings.

The 95% credibility interval for the ratio of count of building-attached cameras is [0.24, 0.46], meaning that the estimate from the decoders over all intersections is between 2x and 4x smaller than what the reference would be.

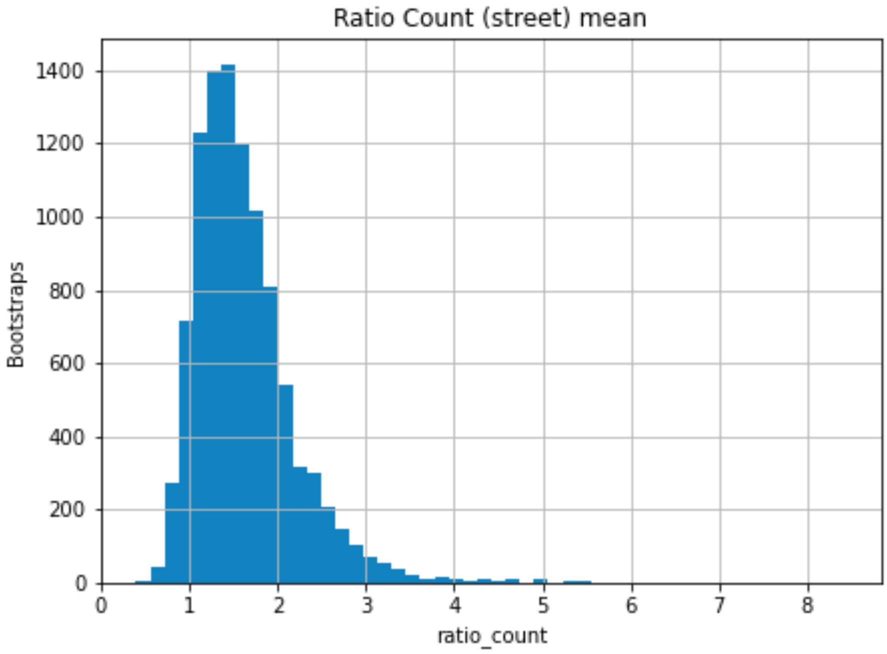
For thoroughness, we applied the same analysis to a binary classification task of presence vs absence of cameras, i.e.: did the decoder tag at least 1 camera, as opposed to the actual count. Results were similar.

CONFIRMING THE USE OF THE MEDIAN INSTEAD OF THE MEAN

As described above, we are using the median of each intersection's three decoders' counts as the value for that intersection, for reasons of robustness to over-estimation compared to the mean. To further confirm this decision, we applied the same quantitative analysis of the ratio of totals to the sum of means rather than the sum of medians. Our hypothesis was that the mean would likely lead to overcounts, and therefore that the sum over the intersections of the means of the decoders' scores would likely be higher than the reference count.

This was confirmed in the following graph, which shows the empirical distribution of the bootstrapped samples of the ratio of the sum of the decoders' means divided by the mean of the experts, for public cameras. The distribution is quite heavily skewed above 1, with a 95% credibility interval of [0.85, 3.06], to be compared to the corresponding graph for the median. This confirms that the choice of the median as aggregation of the decoders is preferable to the mean.

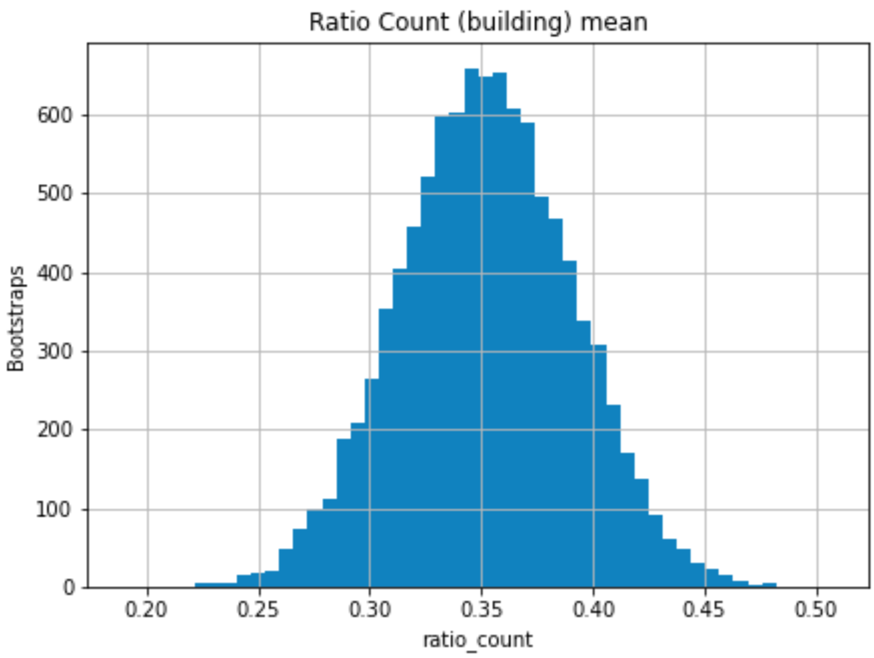
Figure 8: Comparison of mean of decoders vs experts, public cameras



Histogram of the bootstrapped distribution of the ratio of total counts (sum over intersections of means over decoders) by the decoders divided by total reference count, for cameras attached to streetlights, traffic signals or poles.

Interestingly, the same analysis on the cameras attached to buildings does not show such an inversion of the ratio: both mean and median provide comparably under-count by the decoders compared to the experts, as shown in the histogram below.

Figure 9: Comparison of mean of decoders vs experts, private cameras



Histogram of the bootstrapped distribution of the ratio of total counts (sum over intersections of means over decoders) by the decoders divided by total reference count, for cameras attached to buildings.

1.5 FINAL COUNTS

At this stage, we recapitulate the final procedure described above: the final counts for each intersection are the result of the median of the counts from three decoders, for each category of camera. We have demonstrated so far why this procedure provides robust and conservative estimates, which was a key objective of this study.

These numbers can then be relied upon for the aggregations provided at the top of this document, and for further work by Amnesty International and other organizations.

As a final detail, for clarity of exposure and further conservatism of the numbers, all our counts of "total cameras" have been computed as the sum of the median-based subtotals in each three categories (public, private, unknown) — as opposed to the sum of the median of all three decoders' individual total across categories. In other words, we use the sum of the medians, rather than the median of the sum. This brings an even more stringent undercount, and more importantly allows all sums across categories to match the total count, in spite of the non-linearity of the median.

1.6 DISCUSSION AND IDEAS TO EXPLORE

The above analysis has confirmed that this study meets the rigorous standards we set ourselves, being as conservative as possible on the reported totals.

We also investigated but discarded several avenues and extensions. For transparency, we describe them in this section. We hope this will also avoid others going down the same path and encourage future work by any interested reader.

1.6.1 DONE BUT NOT USED FOR FINAL NUMBERS

KRIPPENDORFF ALPHA

We originally investigated the reliability of graders using the celebrated Krippendorff Alpha (Hayes and Krippendorff 2007; Krippendorff 1970; 2004a; 2004b; 2011), and in particular the very recent quadrilogy extension (Krippendorff 2020) which stemmed from a fruitful collaboration on Amnesty International's previous Decode Darfur project in 2017.

We are very grateful to Professor Krippendorff for his care in proposing three new metrics to handle the kind of issues specific to large scale micro tasking data like the Decoders. Sadly, after a deep investigation we eventually had to abandon their inclusion in this report. This is not for lack of interest and relevance of those metrics, but because their development and application to this type of data is so recent that we do not yet have enough experience in interpreting and using them. We made the difficult decision to fall back on metrics we were more experienced with, to ensure our strongest possible analysis of the data within our own abilities.

CONFIDENCE INTERVALS ON TOTALS BY BOOTSTRAPPING THE DECODERS

Although the guarantees above provide sufficient guarantees on the conversative aspect of this study, as statisticians, we would prefer to accompany our total counts with confidence intervals.

We investigated an apparently straightforward way to derive credibility intervals on the total counts of cameras: bootstrapping (Efron and Tibshirani 1994; Efron 2003), which would seem applicable since we have thousands of observations. The set of intersections being fixed, each bootstrapped sample involves resampling three decoders with replacement (that part being particularly important for what follows) within each intersection, computing the resulting median within each intersection, and summing these over the intersections. With enough bootstrap samples we obtained a distribution of the total sum of medians. Its empirical quantiles could provide credibility intervals to go with the point estimate provided by the sum of the medians computed on the original dataset.

However, the confidence intervals thereby computed did not include the point estimate. Indeed, even the 2.5-percentile (lower bound of the bootstrapped credibility interval at 95%) was far superior to the point estimate — actually further from the point estimate than from the 97.5-percentile. A less concerned statistician could have taken this credibility interval at face value and use this to inflate the number of cameras. However, in our constant concern to be as conservative as possible we investigated further.

A manual computation of the actual distribution of the sum of bootstrapped medians leads to a sum of Bernoulli, whose confidence intervals matched those simulated, confirming the validity of our implementation.

Since we have the exhaustive list of all intersections and thus do not resample those, the key issue with our bootstrapping is that we are resampling within each intersection, i.e., bootstrapping from only 3 data points, with replacement. This is a rather extreme use of bootstrapping, but one which would not have been as glaring had we used the average rather than the median, thanks to the linearity of the average. The median, beyond being non-linear, is very sensitive to ties. Resampling 3 from 3 with replacement obviously leads to many such ties, and thus the paradoxical behaviour of the bootstrap we were observing.

We therefore abandoned the use of Bootstrap for confidence intervals of the sum of the medians, preferring a single robust point estimator to an unreliable credibility interval.

CORRECTING THE UNDERCOUNT

The quantitative analysis provided an estimate of the undercount factor by the decoders, albeit with a wide credibility interval. We did consider whether to incorporate this factor into the total counts of cameras, using it as a corrective multiplicative factor. We decided against it for two reasons: simplicity and variance.

The simplicity argument goes as follows: this study must be as easily auditable as possible. Therefore, the benefits of any refinement must be weighed carefully against the extra complexity it adds, and the alternative choices it would provide. Incorporating a multiplicative adjustment ratio could be seen as a crude adjustment, which would open many more questions: Shouldn't the adjustment be done instead by adjusting over each point of the whole joint distribution of experts' counts and decoders' counts on each intersection (which we did compute)? Wouldn't a quantile regression be more adequate? Etc. etc. Although these are all valid questions, we would not have the time to answer all of these during this study. Therefore, simplicity and rigour dictate that we avoid any corrective factor.

The variance rejoins the question of confidence intervals mentioned above. We do have credibility intervals on the ratio of decoders vs experts, and we know these are too wide to be ignored. We do not, as discussed above, have such credibility intervals on the sum of medians. We would therefore either use the confidence intervals on the ratio but only the point estimate of the sum as constant, which is difficult to justify, or would use only point estimates for both, which is dishonest given the amount of uncertainty on the exact value of the ratio beyond the fact that it is smaller than 1. More arguments against such a correction factor.

Finally, for the sake of exhaustivity, and in spite of the aforementioned shortcomings of using the mean instead of the median, we did consider a potential combination of the ratio of the mean with the sum of the means. We do have confidence intervals for both, as the mean does not conflict with the bootstrap across decoders like the median does. We computed credibility intervals by assuming independence of the two variables, then the expectation and the variance of this product with classic formulas and used Chebyshev inequality. The resulting bound does provide theoretically valid confidence intervals but is so loose that it is widely over-covering and results in meaningless intervals. A simpler way would have been to use the cross product of the two bootstrapped samples. We mention this purely for the sake of completeness because the preference for the median makes this a moot point.

1.6.2 POSSIBLE FURTHER ANALYSES, DESIGNED BUT NOT DONE

Amnesty International's priority in this study was to provide robust and conservative results that can be easily audited. Anything beyond that, while interesting from a technical point of view and with many potential upsides, had to be left for future investigations. In order to encourage multiple analysts to go further with this data, we list here the most interesting future directions.

We encourage any interested statistician to contact us.

INTRODUCE A PARAMETRIC ERROR MODEL ON THE COUNTS

Rather than using the median, we could have done any kind of regression by introducing a parametric model of error on the counts. There is a wide literature on error models for count, from Poisson-model errors to zero-inflated models, to ad hoc multiplicative error models.

However, we decided to stick to the simplest possible methodology and limit the number of modelling choices, to make this analysis the easiest to review and the most defensible. This involves foregoing, at least in this analysis, any explicit distributional assumption or modelling choice. An alternative would have been a thorough investigation of multiple possible error models, then proceeding to model criticism and/or model selection, which we simply did not have the time for.

BAYESIAN HIERARCHICAL MODELLING FOR UNCERTAINTY QUANTIFICATION

The most natural modelling approach, given the population nature of the decoders, would be either a mixed-effects model, or, more practical in the authors' professional opinion, a Bayesian hierarchical model, with inference achieved by either MCMC or Variational inference.

This would have had multiple advantages: incorporating custom error models; inferring individual decoders' errors; weighing decoders according to their trustworthiness; incorporating in the same computation the quantitative analysis from the experts and the decoders-led analysis, achieving semi-supervised learning effortlessly; all the while providing distributional results with full uncertainty quantification.

Yet again the priority on robust, conservative, and easily auditable analysis required to leave this for future analysis.

SPATIAL INFORMATION FOR FURTHER INFORMATION RECOVERY

Along with the labels, we recorded the point-of-view data of the decoders' tags, i.e., the 3-dimensional spherical coordinates of the reticle in the panorama where they applied the tags. We recorded this spatial information on the tags knowing it was unlikely we could process it by publication date, because we know it is remarkably important for refinements.

Indeed, using this data allows inference not just of camera counts, but of individual camera identification. It can help confirm whether the three decoders are tagging the exact same cameras or intersecting but distinct sets of different cameras. It can bring precious information to reconcile disagreements amongst decoders more accurately than with the median. For example, it could allow the correction of a tag from one minority label to a majority label if the spherical coordinates match.

Algorithmically, this inference of the latent camera identity could be achieved within the aforementioned Bayesian hierarchical model using an Expectation Maximization algorithm.

COMPUTER VISION FOR CONFIRMATION OF CAMERA PRESENCE

The advances in Computer Vision via Deep Learning make it a realistic suggestion to try and automate detection of the cameras straight from imagery, possibly using the decoders' answers as training data. This was beyond the scope of the present study, but could be interesting on multiple levels, as it would possibly generalize to other cities, depending on how similar their cameras and architectures are to NYC's. To be useful, however, those results would need to be analysed with the same critical eye and conservatism as the current study.

1.7 CONCLUSION

In addition to this methodology note, we are also releasing the resulting data, and the Python code repeating most of the above analysis. With this transparency and release, we intend to further illustrate Amnesty International's commitment to the highest standards of proof and scrutiny, as well as to give back to the community. The authors welcome any questions, contact, ideas, or collaboration on some of the follow-up analyses.

2. ANALYSIS OF STOP-AND-FRISK + CAMERA LOCATIONS

```
library(data.table)
library(ggplot2)
library(sf)
library(units)
library(geojsonsf)
library(lme4)

options(width=150)
theme_set(theme_bw(base_size=16))
```

```
Linking to GEOS 3.8.0, GDAL 3.0.4, PROJ 6.3.1; sf_use_s2() is TRUE

udunits database from /usr/share/xml/udunits/udunits2.xml

Loading required package: Matrix
```

2.1 DATA PREPARATION AND INSPECTION

SOURCES

- Stop-and-frisk (SQF) data from NYPD, 2019 and 2020. https://www1.nyc.gov/site/nypd/stats/reports-analysis/stopfrisk.page
- Census data (demographics etc.) from the American Community Survey (ACS) 2014--2019, downloaded using the `tidycensus` package: https://walker-data.com/tidycensus/articles/spatial-data.html
- Census tract shapefiles 2019 from US Census: https://www.census.gov/cgi-bin/geo/shapefiles/index.php?year=2019&layergroup=Census+Tracts trimmed to the New York state shoreline: http://gis.ny.gov/gisdata/inventories/details.cfm?DSID=927
- Camera locations, as provided by Amnesty's "Decode Surveillance NYC" project

PREPARED DATASETS

- `tracts`: one row per census tract (excluding two entirely-aquatic census tracts, which have been removed). The shape of each tract is included. (All geo coordinates are in EPSG 2908, the State Plane Coordinate System for NY/LongIsland.)
- `census`: one row per census tract, excluding the two aquatic tracts

- `sqf`: one row per stop-and-frisk incident, spanning 2019 and 2020, labelled by census tract (except for four records in 2020 with nonsense locations, which have been removed)
- `camera`: one row per intersection, including geo coordinates
  - `camera_count`: one row per census tract, giving the number of cameras by several different counting methods:
  - `eff_cameras` is the total area within the tract that is visible by public cameras (assuming a 120m radius), divided by the area seen by a camera.
  - `eff_cameras_within_200m` is the total area within 200m of the tract that is visible by public cameras, divided by the area seen by a camera.
  - `cameras_within_200m` is the total number of public cameras within 200m of the tract.

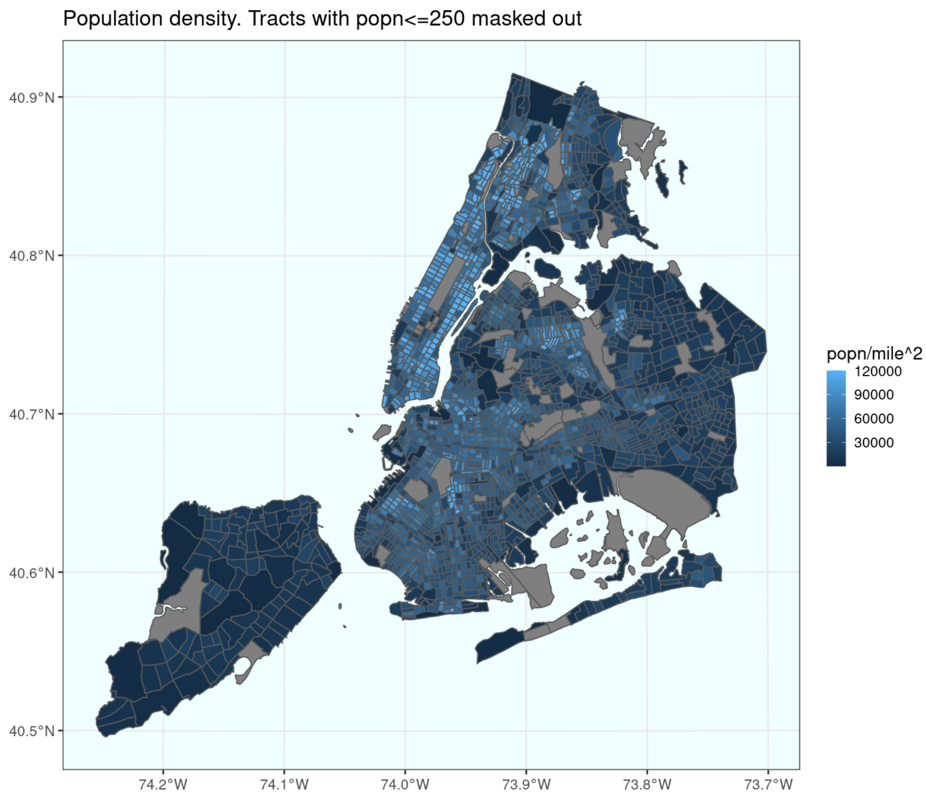
With the latter two metrics, if there are two cameras in nearly the same spot, the total effective number of cameras is just one, since their areas overlap nearly completely. The idea behind this way of measuring surveillance is that if two cameras are in roughly the same spot it's most likely to cope with obstructed sightlines.

```
# Import and pre-process all the datasets
source('prepdata.R')
```

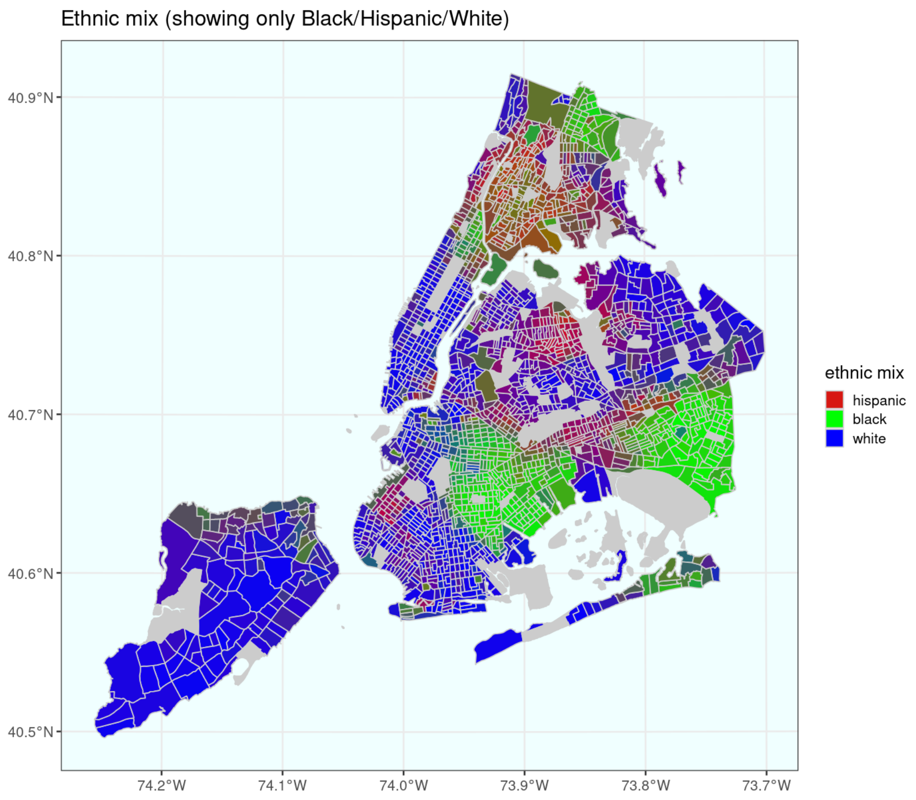
```
To install your API key for use in future sessions, run this function with `install
 = TRUE`.

Getting data from the 2015-2019 5-year ACS

Warning message in eval(ei, envir):
“Assigning 3 stops to nearest tract”
```

2.1.1 DEMOGRAPHICS

```
# Population density, from census data

df <- merge(tracts, census, by='GEOID', all.x=TRUE)
df$area <- set_units(st_area(df), 'mile ^2')
df$density <- df$popn / as.numeric(df$area)

options(repr.plot.width=14, repr.plot.height=10)

ggplot() +
  geom_sf(data=df, aes(fill=ifelse(popn>250,pmin(density,120000),NA))) +
  with(as.list(st_bbox(df)), coord_sf(xlim=c(xmin,xmax), ylim=c(ymin,ymax))) +
  guides(fill=guide_colourbar(title='popn/mile^2')) +
  ggtitle('Population density. Tracts with popn<=250 masked out') +
  theme_bw(base_size=16) +
  theme(panel.background = element_rect(fill='azure'))
```

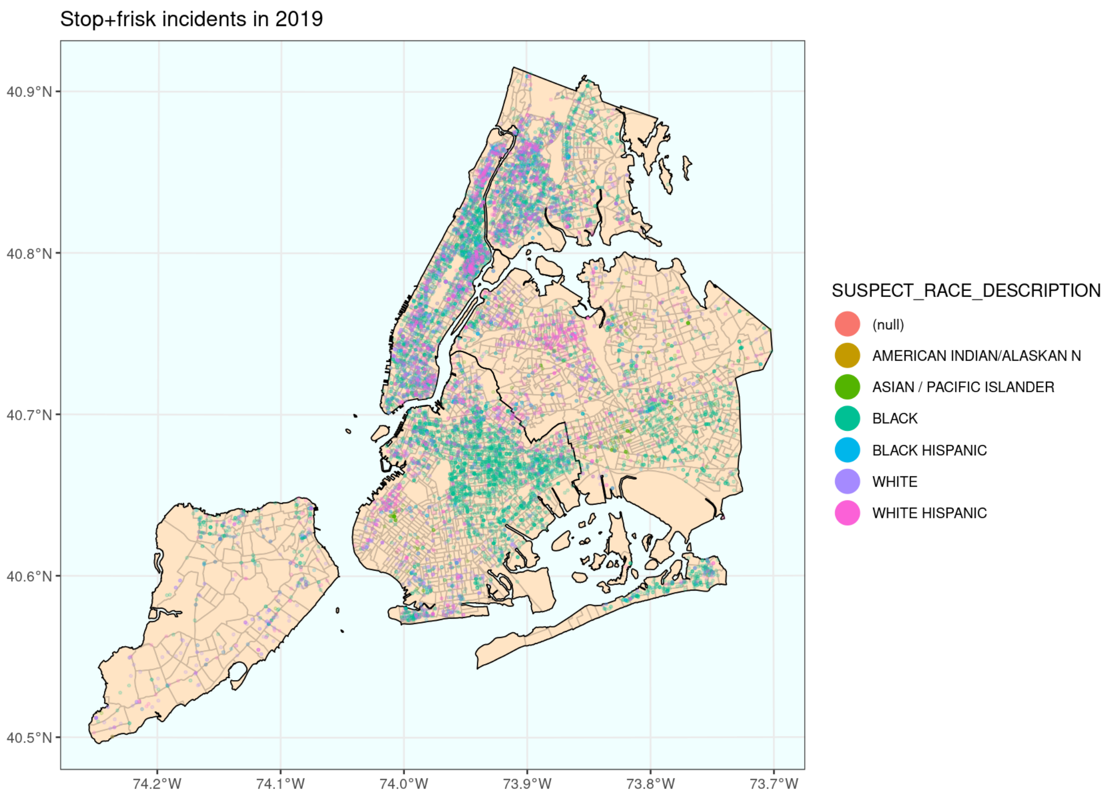



```
# Ethnic mix (black, hispanic, white)

df <- merge(census, camera_count, by='GEOID', all=TRUE)
df <- merge(df, as.data.table(st_drop_geometry(tracts))[, list(GEOID, borough)], by
='GEOID', all=TRUE)

# Colour-code by three-way split, Hispanic / Black / White.
df[, popn2 := ifelse(popn>250, popn.black + popn.hispanic + popn.white, Inf)]
λ.h <- df[, popn.hispanic / popn2]
λ.w <- df[, popn.white / popn2]
λ.b <- df[, popn.black / popn2]
df[, ethcol := rgb(λ.h, λ.b, λ.w)]
eth_guide <- df[c(which.max(λ.h), which.max(λ.b), which.max(λ.w))]
eth_guide[, label := c('hispanic','black','white')]

dft <- merge(tracts[,'GEOID'], df, by='GEOID')

options(repr.plot.width=14, repr.plot.height=10)

ggplot() +
  geom_sf(data=dft, aes(fill=ifelse(popn>250,ethcol,'grey80')), col='grey80') +
  scale_fill_identity(guide='legend', breaks=eth_guide$ethcol, labels=eth_guide$lab
el) +
  guides(fill=guide_legend(title='ethnic mix')) +
  ggtitle('Ethnic mix (showing only Black/Hispanic/White)') +
  theme_bw(base_size=16) +
  theme(panel.background = element_rect(fill='azure'))
```



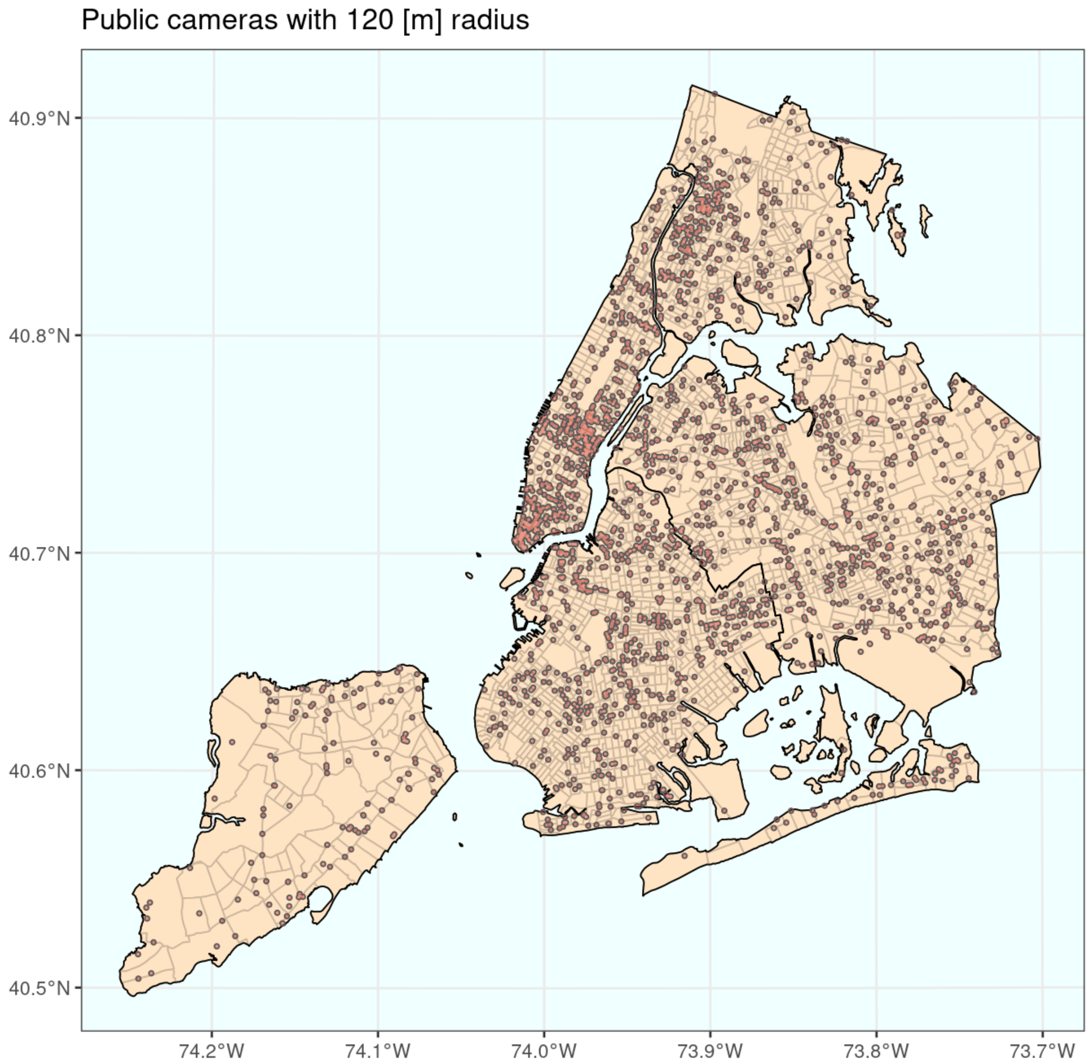
2.1.2 STOP+FRISK

```
# Stop+frisk incidents

options(repr.plot.width=14, repr.plot.height=10)

ggplot() +
  geom_sf(data=tracts, colour='bisque3', fill='bisque') +
  geom_sf(data=BOROUGH, colour='black', fill=NA) +
  geom_sf(data=sqf[sqf$YEAR2==2019,], aes(col=SUSPECT_RACE_DESCRIPTION), alpha=.2,
size=1) +
  with(as.list(st_bbox(sqf)), coord_sf(xlim=c(xmin,xmax), ylim=c(ymin,ymax))) +
  guides(colour = guide_legend(override.aes=list(alpha=1, size=10))) +
  ggtitle('Stop+frisk incidents in 2019') +
  theme_bw(base_size=16) +
  theme(panel.background = element_rect(fill='azure'))
```



2.1.3 SURVEILLANCE

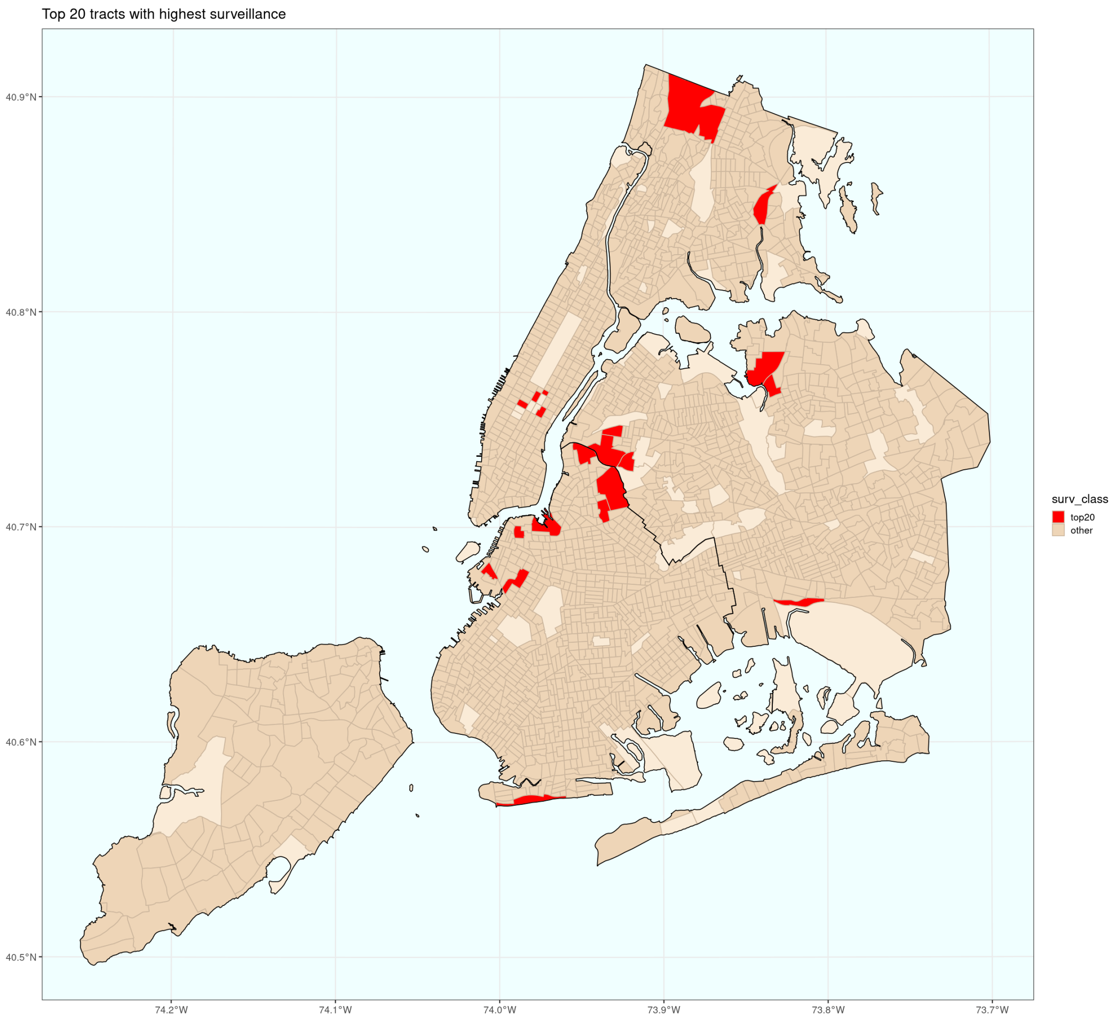
```
# Public camera locations

options(repr.plot.width=10.4, repr.plot.height=10)

camera_coverage <- st_union(st_buffer(camera[camera$public,], dist=CAMERA_RADIUS))

ggplot() +
  geom_sf(data=tracts, colour='bisque3', fill='bisque') +
  geom_sf(data=BOROUGH, colour='black', fill=NA) +
  geom_sf(data=camera_coverage, fill='firebrick3', alpha=.4) +
  with(as.list(st_bbox(sqf)), coord_sf(xlim=c(xmin,xmax), ylim=c(ymin,ymax))) +
  ggtitle(paste('Public cameras with',format(CAMERA_RADIUS),'radius')) +
  theme_bw(base_size=16) +
  theme(panel.background=element_rect(fill='azure'))
```



2.1.4 SURVEILLANCE / POPULATION

```
# Which tracts have the highest surveillance levels?
# (surveillance level = effective number of cameras per 1000 residents)
# Only look at tracts with popn > 250, to exclude non-residential areas.

df <- merge(census, camera_count, by='GEOID', all=TRUE)
df[, surv := eff_cameras / popn * 1000]
df[, surv_rank := rank(-surv), by=popn>250]
df[, surv_class := ifelse(popn>250, ifelse(surv_rank<=20,'top20','other'), NA)]
df <- merge(tracts[,c('GEOID','NAMELSAD','borough')], df, by='GEOID', all=TRUE)
st_agr(df) <- 'constant'
# Get a Google Maps url for the centroid of the tract
df2 <- st_centroid(df[,'GEOID'])
df2 <- st_transform(df2, crs='epsg:4326')
df2 <- cbind(data.table(GEOID=df2$GEOID), st_coordinates(df2))
df2[, url := paste0('http://maps.google.com/maps?z=12&t=m&q=loc:',Y,'+',X)]
df <- merge(df, df2[, list(GEOID,url)], by='GEOID', all=TRUE)

options(repr.plot.width=20.8, repr.plot.height=20)

ggplot() +
  geom_sf(data=df, aes(fill=surv_class), colour='bisque3') +
```

```
  geom_sf(data=BOROUGH, colour='black', fill=NA) +
  scale_fill_manual(values=c('top20'='red', 'other'='bisque2'), na.value='antiquewh
ite') +
  with(as.list(st_bbox(sqf)), coord_sf(xlim=c(xmin,xmax), ylim=c(ymin,ymax))) +
  ggtitle('Top 20 tracts with highest surveillance') +
  theme_bw(base_size=16) +
  theme(panel.background=element_rect(fill='azure'))
```




```
# The top 50 most surveilled census tracts (excluding those with popn <= 25
0)

as.data.table(df)[popn>250 & surv_rank<=50][order(surv_rank), list(GEOID,NAMELSAD,
borough,popn,eff_cameras,surv_rank,url)]
```

surv_rank	GEOID	NAMELSAD	borough	popn	eff_cameras
1	36047054300	Census Tract 543	Brooklyn	283	2.4232144
2	36081019900	Census Tract 199	Queens	697	4.9148482
3	36081017900	Census Tract 179	Queens	1019	4.7119249
4	36061011202	Census Tract 112.02	Manhattan	415	1.6862763
5	36081090700	Census Tract 907	Queens	1434	5.7624049
6	36081084602	Census Tract 846.02	Queens	925	3.5133866

7	36005028400	Census Tract 284	Bronx	554	1.5409576
8	36047044900	Census Tract 449	Brooklyn	3210	8.9022172
9	36061011900	Census Tract 119	Manhattan	1071	2.9550138
10	36061010400	Census Tract 104	Manhattan	811	2.1815545
11	36061009200	Census Tract 92	Manhattan	1474	3.9182786
12	36047057900	Census Tract 579	Brooklyn	1165	3.0745038
13	36005043500	Census Tract 435	Bronx	499	1.2030816
14	36047011900	Census Tract 119	Brooklyn	1322	3.0205297
15	36047035200	Census Tract 352	Brooklyn	1254	2.8244718
16	36081086900	Census Tract 869	Queens	1771	3.8132589
17	36081020500	Census Tract 205	Queens	1176	2.3714092
18	36047005900	Census Tract 59	Brooklyn	1213	2.4192464
19	36047001300	Census Tract 13	Brooklyn	1917	3.7913915
20	36047048500	Census Tract 485	Brooklyn	2289	4.5043668
21	36047109800	Census Tract 1098	Brooklyn	2359	4.3413446
22	36061003700	Census Tract 37	Manhattan	2666	4.7583208
23	36081008500	Census Tract 85	Queens	883	1.5557916
24	36005006300	Census Tract 63	Bronx	4582	7.7866877
25	36061000900	Census Tract 9	Manhattan	1796	2.9950564
26	36005011700	Census Tract 117	Bronx	1443	2.3350008
27	36047001800	Census Tract 18	Brooklyn	1897	3.0655438
28	36061019701	Census Tract 197.01	Manhattan	639	1.012544
29	36081029300	Census Tract 293	Queens	1090	1.6693786
30	36005001900	Census Tract 19	Bronx	3141	4.772945
31	36061004500	Census Tract 45	Manhattan	980	1.4861464
32	36061011203	Census Tract 112.03	Manhattan	1103	1.6466061
33	36061001300	Census Tract 13	Manhattan	4455	6.6253457
34	36061003100	Census Tract 31	Manhattan	2525	3.6979451
35	36061011401	Census Tract 114.01	Manhattan	1173	1.6631851
36	36081020800	Census Tract 208	Queens	3136	4.3191795
37	36081003300	Census Tract 33	Queens	3569	4.8857703
38	36061010100	Census Tract 101	Manhattan	1373	1.8746598
39	36047003500	Census Tract 35	Brooklyn	1907	2.5967407
40	36061012500	Census Tract 125	Manhattan	2311	3.1256031
41	36047036700	Census Tract 367	Brooklyn	1281	1.7225404
42	36061010000	Census Tract 100	Manhattan	1741	2.3397171
43	36047004700	Census Tract 47	Brooklyn	1877	2.4798944
44	36047079400	Census Tract 794	Brooklyn	1716	2.2276285
45	36081042600	Census Tract 426	Queens	477	0.6125966
46	36081066300	Census Tract 663	Queens	2771	3.5518514
47	36081148300	Census Tract 1483	Queens	2900	3.6298691
48	36061009900	Census Tract 99	Manhattan	5981	7.3855809
49	36081140901	Census Tract 1409.01	Queens	990	1.2021309
50	36005028600	Census Tract 286	Bronx	1085	1.3108964

surv_ rank	url
1	http://maps.google.com/maps?z=12&t=m&q=loc:40.7009021108811+-73.9711860731856
2	http://maps.google.com/maps?z=12&t=m&q=loc:40.7351969475049+-73.9340362079201
3	http://maps.google.com/maps?z=12&t=m&q=loc:40.7446892921271+-73.9302772957279
4	http://maps.google.com/maps?z=12&t=m&q=loc:40.7626039615687+-73.9721311338371
5	http://maps.google.com/maps?z=12&t=m&q=loc:40.7742445733847+-73.8380630268819
6	http://maps.google.com/maps?z=12&t=m&q=loc:40.6651262013001+-73.8165808751078
7	http://maps.google.com/maps?z=12&t=m&q=loc:40.8494364375106+-73.8386905127224
8	http://maps.google.com/maps?z=12&t=m&q=loc:40.717832474392+-73.9310189503987
9	http://maps.google.com/maps?z=12&t=m&q=loc:40.7573151200653+-73.9860246553007
10	http://maps.google.com/maps?z=12&t=m&q=loc:40.7607790857575+-73.9776728684315
11	http://maps.google.com/maps?z=12&t=m&q=loc:40.7536475785273+-73.9747422230935
12	http://maps.google.com/maps?z=12&t=m&q=loc:40.7343632301197+-73.9484579099195
13	http://maps.google.com/maps?z=12&t=m&q=loc:40.8945006565832+-73.8819835591425
14	http://maps.google.com/maps?z=12&t=m&q=loc:40.6753263429599+-73.9898082191175
15	http://maps.google.com/maps?z=12&t=m&q=loc:40.5733064393004+-73.9812868769882
16	http://maps.google.com/maps?z=12&t=m&q=loc:40.7653527115675+-73.8336673547186
17	http://maps.google.com/maps?z=12&t=m&q=loc:40.7303893338009+-73.9218036203629
18	http://maps.google.com/maps?z=12&t=m&q=loc:40.6794027860982+-74.0061192594526
19	http://maps.google.com/maps?z=12&t=m&q=loc:40.6976149764291+-73.9883585864722
20	http://maps.google.com/maps?z=12&t=m&q=loc:40.707839855891+-73.9363493490845
21	http://maps.google.com/maps?z=12&t=m&q=loc:40.6526617235089+-73.9016506102304
22	http://maps.google.com/maps?z=12&t=m&q=loc:40.726278095626+-74.0075034204877
23	http://maps.google.com/maps?z=12&t=m&q=loc:40.7600215165174+-73.940722322593
24	http://maps.google.com/maps?z=12&t=m&q=loc:40.8238506185581+-73.9283912087115
25	http://maps.google.com/maps?z=12&t=m&q=loc:40.7023246877833+-74.0098565139476
26	http://maps.google.com/maps?z=12&t=m&q=loc:40.8105018153381+-73.8766835274815
27	http://maps.google.com/maps?z=12&t=m&q=loc:40.6553358361173+-74.0132794525594
28	http://maps.google.com/maps?z=12&t=m&q=loc:40.80531104441+-73.9593993468251
29	http://maps.google.com/maps?z=12&t=m&q=loc:40.7519326725642+-73.8994830070836
30	http://maps.google.com/maps?z=12&t=m&q=loc:40.8030405682337+-73.9146080116714
31	http://maps.google.com/maps?z=12&t=m&q=loc:40.7205701206259+-73.9993912050248

32	http://maps.google.com/maps?z=12&t=m&q=loc:40.7612443989067+-73.9689142454845
33	http://maps.google.com/maps?z=12&t=m&q=loc:40.7091259661897+-74.0129925082366
34	http://maps.google.com/maps?z=12&t=m&q=loc:40.7153109519573+-74.0038150937813
35	http://maps.google.com/maps?z=12&t=m&q=loc:40.7648402252091+-73.9704938939059
36	http://maps.google.com/maps?z=12&t=m&q=loc:40.6983846475047+-73.8068861371267
37	http://maps.google.com/maps?z=12&t=m&q=loc:40.75438447924+-73.9380995999987
38	http://maps.google.com/maps?z=12&t=m&q=loc:40.7497314080164+-73.9915412167347
39	http://maps.google.com/maps?z=12&t=m&q=loc:40.6853251969354+-73.9761802762101
40	http://maps.google.com/maps?z=12&t=m&q=loc:40.7598407190052+-73.9841752446922
41	http://maps.google.com/maps?z=12&t=m&q=loc:40.6775645923497+-73.9046130106438
42	http://maps.google.com/maps?z=12&t=m&q=loc:40.7580652720345+-73.9712318119244
43	http://maps.google.com/maps?z=12&t=m&q=loc:40.6883701898338+-74.0018565551786
44	http://maps.google.com/maps?z=12&t=m&q=loc:40.648605301847+-73.9552860107495
45	http://maps.google.com/maps?z=12&t=m&q=loc:40.6888726143057+-73.7702024656798
46	http://maps.google.com/maps?z=12&t=m&q=loc:40.7211189613178+-73.8773457274675
47	http://maps.google.com/maps?z=12&t=m&q=loc:40.7744932566264+-73.7492503275423
48	http://maps.google.com/maps?z=12&t=m&q=loc:40.7520190332368+-74.0049130132931
49	http://maps.google.com/maps?z=12&t=m&q=loc:40.7465404582977+-73.7749009383578
50	http://maps.google.com/maps?z=12&t=m&q=loc:40.8491849054577+-73.8477751648193

2.2 THE NUMBER OF STOP+FRISK INCIDENTS IS CLOSELY LINKED TO THE LEVEL OF SURVEILLANCE

We first analyze how the number of stop+frisk incidents depends on the number of cameras. The underlying statistical model we'll use is a generalized linear model,

$$\text{average num.stops in tract} = \lambda \times \text{tract.popn}/1000,$$

Here λ is the rate of stop+frisk incidents per 1000 population, and the focus of the analysis is to understand how λ depends on level of surveillance.

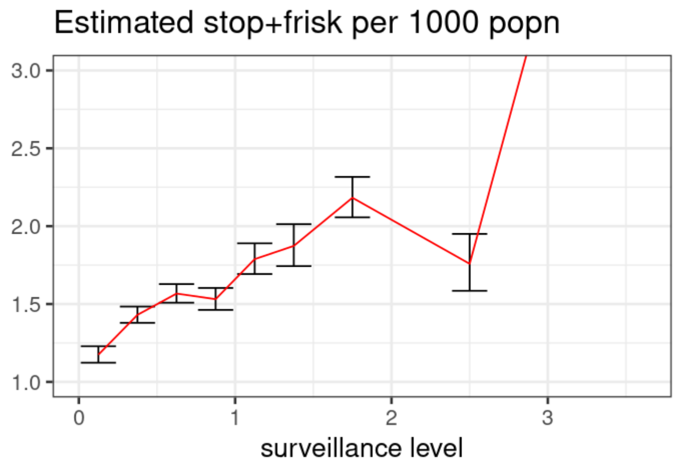
Furthermore, we'll model the actual number of stops as a Poisson random variable, with mean as specified above. This is a standard statistical model for analyzing count data.

We split the tracts into 9 groups, according to level of surveillance, and estimate λ separately for each group. (This allows us to assess the relationship between stop+frisk and surveillance, without assuming any particular form of the equation.) For this analysis, surveillance level is defined as the effective number of cameras within 200m of a given census tract, per 1000 residents. We see that the stop+frisk rate λ increases with the level of surveillance, and the relationship is roughly linear.

Our analysis uses data for 2019. We restrict attention to census tracts with a population of >250, as a simple way to exclude parks etc.

```
df <- as.data.table(expand.grid(GEOID=unique(census$GEOID), YEAR2=unique(sqf$YEAR2)
))
```

```r
df <- merge(df, as.data.table(sqf)[, list(numstops=.N), by=list(GEOID,YEAR2)], all=
TRUE) # numstops per tract,year
df <- merge(df, census, by='GEOID', all=TRUE) # popn, popn.black, popn.hispanic, po
pn.white
df <- merge(df, camera_count, by='GEOID', all=TRUE) # eff_cameras_within_200m and o
ther counts
df <- merge(df, as.data.table(st_drop_geometry(tracts))[, list(GEOID, borough)], by
='GEOID', all=TRUE) # borough
df[is.na(numstops), numstops := 0] # for the tracts with no recorded stops

# Let stoprate = number of stops per 1000 residents in a census tract
# Let surv by the effective number of cameras within 200m of the tract, per 1000 re
sidents

df[, stoprate := numstops/popn*1000]
df[, surv := eff_cameras_within_200m/popn*1000]

# For a non-parametric model, it's useful to split the tracts into separate groups
# according to surveillance level. Let survF be a split version of surv.

breaks <- c(seq(0,1.5,by=.25), 2, 3)
break_midpoint <- c((tail(breaks,-1) + head(breaks,-1)) / 2, 3.5)
df[, survF := cut(surv, breaks=c(breaks,Inf), labels=break_midpoint, include.lowest
=TRUE)]
breaks

[1] 0.00 0.25 0.50 0.75 1.00 1.25 1.50 2.00 3.00

# Estimate λ as a function of level of surveillance (using survF, our discretized v
ersion
# of level-of-surveillance). The plot shows the estimates for λ as well as 95% conf
idence
# intervals, for each surveillance level.

fit <- glm(numstops ~ 0 + survF, offset=log(popn/1000),
           data=df[popn>250 & YEAR2==2019],
           family='poisson')
x <- as.data.table(coef(summary(fit)))
x[, stoprate := exp(Estimate)]
x[, lo := exp(Estimate-1.96*`Std. Error`)]
x[, hi := exp(Estimate+1.96*`Std. Error`)]
x[,'survF' := levels(df$survF)]

options(repr.plot.width=6, repr.plot.height=4)

ggplot(data=x) +
  geom_errorbar(aes(x=as.numeric(survF), ymin=lo, ymax=hi)) +
  geom_line(aes(x=as.numeric(survF), y=stoprate), colour='red') +
  xlab('surveillance level') + ylab('') +
  ggtitle('Estimated stop+frisk per 1000 popn') +
  coord_cartesian(ylim=c(1,3))
```



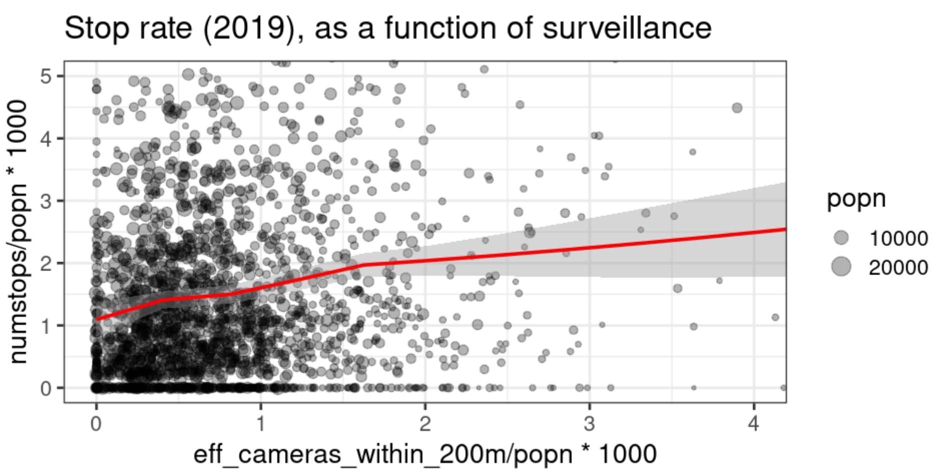
2.2.1 SANITY CHECKS

Here are some plots that support the underlying statistical model described above.

The first plot shows that the stop+frisk rate (number of stops per 1000 residents) grows with the surveillance level. This plot is very noisy.

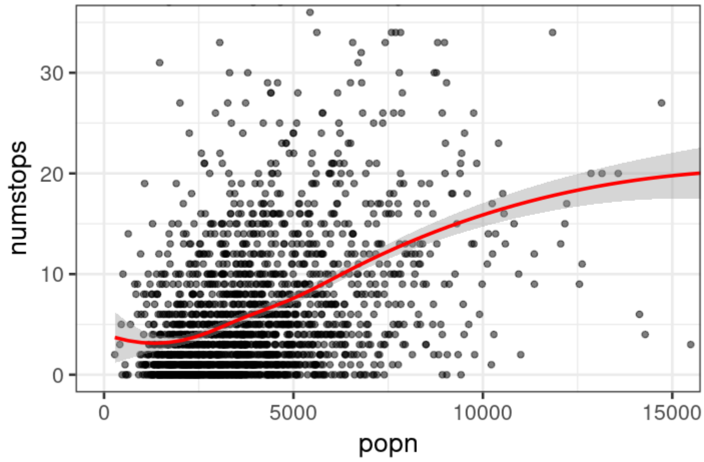
The second and third plots show why there is so much noise. The actual number of stops in a given census tract is a small integer, mostly in the range 0-10, and so there is bound to be lots of noise in the data for a single census tract. The second plot supports the idea that the number of stop+frisk incidents is proportional to population, and the third plot is consistent with a Poisson model.

```
options(repr.plot.width=8, repr.plot.height=4)

ggplot(data=df[popn>250 & YEAR2==2019], aes(y=numstops/popn*1000, x=eff_cameras_wit
hin_200m/popn*1000)) +
  geom_point(aes(size=popn), alpha=.3) +
  geom_smooth(method='loess', colour='red', formula=y~x) +
  scale_size_area() +
  coord_cartesian(xlim=c(0,4), ylim=c(0,5)) +
  ggtitle('Stop rate (2019), as a function of surveillance')
```

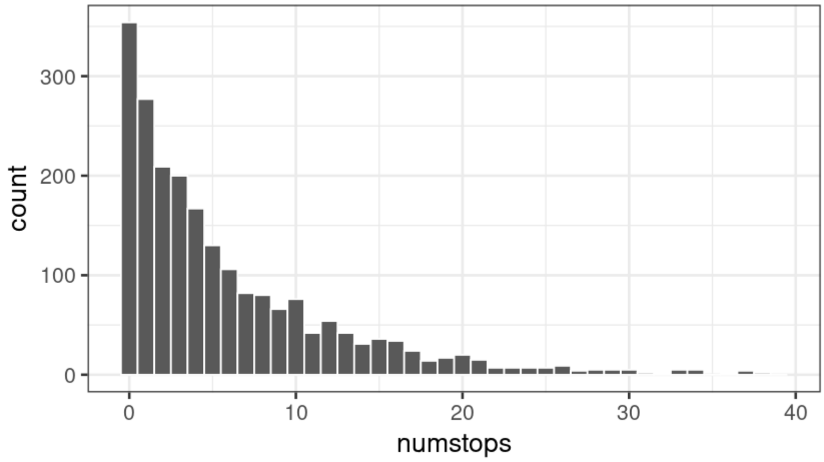



```
options(repr.plot.width=6, repr.plot.height=4)

ggplot(data=df[popn>250 & YEAR2==2019], aes(x=popn, y=numstops)) +
  geom_point(alpha=.5) +
```

```
  geom_smooth(method='loess', colour='red', formula=y~x) +
  coord_cartesian(xlim=c(0,15000), ylim=c(0,35))
```




```
options(repr.plot.width=7, repr.plot.height=4)

ggplot(data=df[YEAR2==2019]) +
  geom_histogram(aes(x=numstops), colour='white', breaks=seq(-.5,40,by=1))
```



2.3 STOP+FRISK RATES ALSO VARY WITH RACIAL MIX, ON TOP OF THE LINK TO SURVEILLANCE

What else does the stop+frisk rate depend on? As before we consider the model

$$\text{average num.stops in tract} = \lambda \times \text{tract.popn}/1000$$

and we investigate what λ depends on. Our baseline model is a simple generalized linear model,

$$\log\lambda = \alpha + \beta \times \text{surveillance.level} + \gamma \times \text{nonwhite.fraction}$$

where α, β, γ, are coefficients to be estimated from the data.

Unsurprisingly, the coefficient for surveillance level (defined as effective number of cameras within 200m of the tract per 1000 residents) is positive, and highly significant (coef=0.02, $p<0.001$, for Queens in 2019).

The coefficient for nonwhite.fraction (defined as the fraction of residents who identify as Black or Hispanic, out of those who identify as Black or Hispanic or White) is also positive, and highly significant (coef=0.83, $p<0.001$, for Queens in 2019).

The β and γ coefficients vary from borough to borough, and they are consistent from 2019 to 2020. (See the chart below for the coefficient values and 95% confidence intervals.) They are consistently positive, and consistently significant.

The fact that both β and γ are highly significant shows that they are not confounding each other. In other words, it is *not* the case that variation in stop+frisk due to surveillance level is entirely explained by the racial mix.

2.3.1 SANITY CHECKS

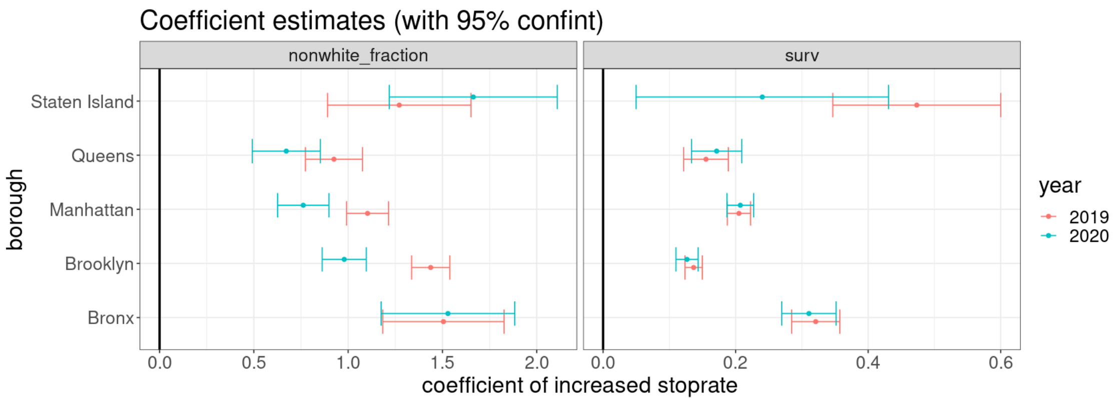
We developed the baseline model using data from a single borough (Queens), to avoid overfitting.

- We tested for a non-linear dependence on surveillance level, but it is not significant.
- We also tested a model with separate coefficients for black.fraction and hispanic.fraction, but the difference between these coefficients is not significant. For all our analyses below, we have therefore pooled Black and Hispanic populations.
- We also tested for an interaction between surveillance.level and nonwhite.fraction. It was not significant. (We might expect we'd need more data to estimate an interaction effect, so we also tried pooling four boroughs excluding Manhattan, and also pooling all five boroughs. It remains not significant in both cases.)

```
df <- as.data.table(expand.grid(GEOID=unique(census$GEOID), YEAR2=unique(sqf$YEAR2)
))
df <- merge(df, as.data.table(sqf)[, list(numstops=.N), by=list(GEOID,YEAR2)], all=
TRUE) # numstops per tract,year
df <- merge(df, census, by='GEOID', all=TRUE) # popn, popn.black, popn.hispanic, po
pn.white
df <- merge(df, camera_count, by='GEOID', all=TRUE) # eff_cameras_within_200m and o
ther counts
df <- merge(df, as.data.table(st_drop_geometry(tracts))[, list(GEOID, borough)], by
='GEOID', all=TRUE) # borough
df[is.na(numstops), numstops := 0] # for the tracts with no recorded stops

# Let stoprate = number of stops per 1000 residents in a census tract
# Let surv by the effective number of cameras within 200m of the tract, per 1000 re
sidents

df[, stoprate := numstops/popn*1000]
df[, surv := eff_cameras_within_200m/popn*1000]
df[, nonwhite_fraction := (popn.black + popn.hispanic) / (popn.black + popn.hispani
c + popn.white)]

# The baseline model
fit0 <- glm(numstops ~ surv + nonwhite_fraction, offset=log(popn),
            family='poisson',
            data=df[YEAR2==2019 & borough=='Queens'],
            subset=popn>250)

# Baseline model sanity check. Is it reasonable to assume a linear dependence on su
rv?
# Is it reasonable to pool Black and Hispanic?

fit1 <- update(fit0, . ~ . + I(surv^2) + I(popn.black/popn))
summary(fit1)


Call:
glm(formula = numstops ~ surv + nonwhite_fraction + I(surv^2) +
    I(popn.black/popn), family = "poisson", data = df[YEAR2 ==
    2019 & borough == "Queens"], subset = popn > 250, offset = log(popn))
```

```
Deviance Residuals:
    Min       1Q   Median       3Q      Max
-3.7321  -1.6934  -0.5816   0.6344   7.4685

Coefficients:
                    Estimate Std. Error  z value Pr(>|z|)
(Intercept)        -7.538773   0.065070 -115.856  < 2e-16 ***
surv                0.204248   0.035519    5.750 8.90e-09 ***
nonwhite_fraction   0.832449   0.116499    7.146 8.96e-13 ***
I(surv^2)          -0.004956   0.003158   -1.569    0.117
I(popn.black/popn)  0.087239   0.099152    0.880    0.379
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for poisson family taken to be 1)

    Null deviance: 2267.1  on 640  degrees of freedom
Residual deviance: 2037.0  on 636  degrees of freedom
AIC: 3614.9

Number of Fisher Scoring iterations: 5

# Further sanity check. Is there any interaction between the surveillance term and
the nonwhite_fraction?
# We'll use a simple binarized version of nonwhite_fraction -- it's more robust to
explore the question
# non-parametrically in the first instance, than to assume a formula.

df[, nwfC := cut(nonwhite_fraction, breaks=2)]

fit <- lm(stoprate ~ nwfC*surv,
          weight=popn,
          data=df,
          subset=popn>250 & YEAR2==2019 & borough!='Manhattan')
summary(fit)


Call:
lm(formula = stoprate ~ nwfC * surv, data = df, subset = popn >
    250 & YEAR2 == 2019 & borough != "Manhattan", weights = popn)

Weighted Residuals:
    Min      1Q  Median      3Q     Max
-174.08  -51.15  -19.24   28.41  831.22

Coefficients:
                 Estimate Std. Error t value Pr(>|t|)
(Intercept)       0.52090    0.07330   7.106 1.71e-12 ***
nwfC(0.5,1]       0.83911    0.09913   8.465  < 2e-16 ***
surv              0.52639    0.07327   7.185 9.78e-13 ***
nwfC(0.5,1]:surv  0.04881    0.09851   0.495     0.62
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 87.59 on 1823 degrees of freedom
Multiple R-squared:  0.1434,	Adjusted R-squared:  0.142
F-statistic: 101.7 on 3 and 1823 DF,  p-value: < 2.2e-16

# How do the coefficients vary from year to year, and borough to borough?

resdf <- as.data.table(expand.grid(YEAR2=unique(df$YEAR2), borough=unique(df$boroug
h)))
```

```r
resdf <- mapply(resdf$YEAR2, resdf$borough, SIMPLIFY=FALSE, FUN=function(y,b) {
  fit <- update(fit0, data=df[YEAR2==y & borough==b])
  x <- coef(summary(fit))
  data.table(YEAR2=y, borough=b, coef=dimnames(x)[[1]], Estimate=x[,'Estimate'], se
=x[,'Std. Error'])
})
resdf <- do.call(rbind, resdf)
resdf[, year := factor(YEAR2)]

options(repr.plot.width=14, repr.plot.height=5)

ggplot(data=resdf[coef!='(Intercept)']) +
  geom_vline(xintercept=0, size=1, color='black') +
  geom_errorbarh(aes(xmin=Estimate-1.96*se, xmax=Estimate+1.96*se, y=borough,col=ye
ar), position=position_dodge(0.3)) +
  geom_point(aes(x=Estimate, y=borough, col=year), position=position_dodge(0.3)) +
  facet_wrap(~coef, scales='free_x') +
  theme_bw() + theme(text=element_text(size=20)) +
  ggtitle('Coefficient estimates (with 95% confint)') +
  xlab('coefficient of increased stoprate')
```



2.4 HOW DOES SURVEILLANCE DEPEND ON DEMOGRAPHICS ETC.?

We have seen that stop+frisk rates depend separately on surveillance level and on the proportion of nonwhite residents. We now investigate what surveillance level depends on.

For these analyses we'll measure surveillance level by effective number of cameras per 1000 residents in a census tract. As before we assign each camera a radius of 120m, and we measure the total area visible, then divide by the area visible by a single camera. In this section we're analysing the attributes of each area of the city, so we'll measure the area surveilled within each census tract (eff_cameras/popn). This in contrast to the earlier analyses of stop+frisk counts, where we analysed the attributes of residents, and we measured the area surveilled within a neighbourhood of the census tract `(eff_cameras_within_200m/popn).`

- In Manhatten, the higher nonwhite_fraction, the lower the level of surveillance (p<0.001).
- In Bronx (p=0.053), Brooklyn (p=0.027), and Queens (p=0.015), the higher the nonwhite_fraction, the higher the level of surveillance
- In Staten Island, no significant relationship (p=0.082).

When we take accout of poverty `(. ~ . + borough:med.income)`, the findings point in the same direction, though they are less significant. This suggests there is some degree of confounding, since there is more poverty linked with greater proportion of nonwhite residents.

Manhattan is most likely a special case: it's a transport hub, so there are many non-resident occupants, and policing e.g. surveillance may well be linked to the number of occupants rather than residents.

2.4.1 MODEL CHOICE

The analyses are based on logistic regression. Surveillance level is most definitely non-Gaussian (it's truncated at zero -- see the histogram below), so it's not sound to fit a linear regression. Instead, we have binarized it into low versus high, with a threshold of 0.18 cameras per 1000 residents (close to the median). This is a simple way to get robust results.

Our baseline model is

$$\text{logitProb(high)} = \alpha_{\text{borough}} + \beta_{\text{borough}} \times \text{nonwhite\_fraction}$$

and we are interested in the β coefficients, one for each borough.

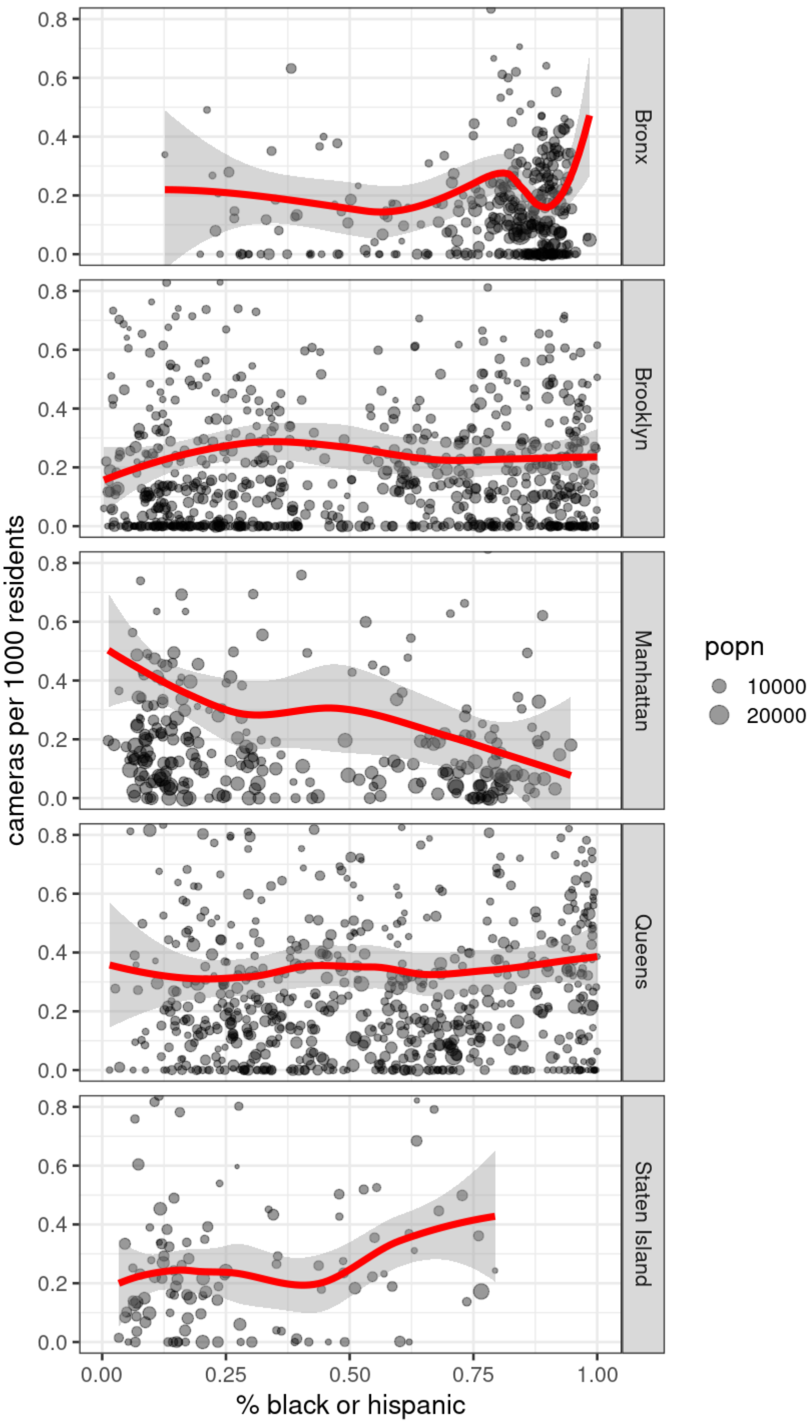
```
df <- merge(census, camera_count, by='GEOID', all=TRUE)
df <- merge(df, as.data.table(st_drop_geometry(tracts))[, list(GEOID, borough)], by
='GEOID', all=TRUE)
df[, nonwhite_fraction := (popn.black + popn.hispanic) / (popn.black + popn.hispani
c + popn.white)]
df[, surv := eff_cameras / popn * 1000]
df[, survF := ifelse(surv < quantile(surv, 2/3, na.rm=TRUE), 'low', 'high')]

SURV_THRESHOLD <- 0.2

# The baseline model

fit <- glm(surv > SURV_THRESHOLD ~ 0 + borough + borough:nonwhite_fraction,
           data=df, subset=popn>250,
          family='binomial')
summary(fit)


Call:
glm(formula = surv > SURV_THRESHOLD ~ 0 + borough + borough:nonwhite_fraction,
    family = "binomial", data = df, subset = popn > 250)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-1.5946  -1.0872  -0.8329   1.1893   1.8189

Coefficients:
                                      Estimate Std. Error z value Pr(>|z|)
boroughBronx                           -1.6062     0.5452  -2.946  0.00321 **
boroughBrooklyn                        -0.5881     0.1384  -4.249 2.15e-05 ***
boroughManhattan                        0.3219     0.2016   1.597  0.11025
boroughQueens                          -0.1307     0.1747  -0.748  0.45419
boroughStaten Island                   -0.3403     0.3307  -1.029  0.30348
boroughBronx:nonwhite_fraction          1.2966     0.6702   1.935  0.05301 .
boroughBrooklyn:nonwhite_fraction       0.5029     0.2268   2.217  0.02662 *
boroughManhattan:nonwhite_fraction     -1.7864     0.4529  -3.944 8.00e-05 ***
boroughQueens:nonwhite_fraction         0.6958     0.2873   2.422  0.01545 *
boroughStaten Island:nonwhite_fraction  1.6749     0.9642   1.737  0.08236 .
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 2914.0  on 2102  degrees of freedom
Residual deviance: 2814.2  on 2092  degrees of freedom
AIC: 2834.2
```

```
Number of Fisher Scoring iterations: 4

# What if surveillance is related to poverty levels instead?
# Doesn't look like it.

fit2 <- update(fit, . ~ . + med.income)
summary(fit2)


Call:
glm(formula = surv > SURV_THRESHOLD ~ borough + med.income +
    borough:nonwhite_fraction - 1, family = "binomial", data = df,
    subset = popn > 250)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-1.5694  -1.0901  -0.8293   1.1998   1.8380

Coefficients:
                                        Estimate Std. Error z value Pr(>|z|)
boroughBronx                          -1.504e+00  5.929e-01  -2.537 0.011177 *
boroughBrooklyn                       -4.437e-01  2.076e-01  -2.137 0.032564 *
boroughManhattan                       5.941e-01  3.496e-01   1.699 0.089305 .
boroughQueens                          1.157e-02  2.299e-01   0.050 0.959853
boroughStaten Island                  -1.356e-01  3.857e-01  -0.352 0.725108
med.income                            -1.746e-06  1.810e-06  -0.965 0.334749
boroughBronx:nonwhite_fraction         1.270e+00  7.014e-01   1.811 0.070181 .
boroughBrooklyn:nonwhite_fraction      4.358e-01  2.363e-01   1.845 0.065108 .
boroughManhattan:nonwhite_fraction    -2.038e+00  5.357e-01  -3.804 0.000142 ***
boroughQueens:nonwhite_fraction        6.659e-01  2.885e-01   2.308 0.020982 *
boroughStaten Island:nonwhite_fraction 1.412e+00  9.828e-01   1.437 0.150747
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 2898.7  on 2091  degrees of freedom
Residual deviance: 2799.2  on 2080  degrees of freedom
  (11 observations deleted due to missingness)
AIC: 2821.2

Number of Fisher Scoring iterations: 4

# We should consider unpacking %non-white into black & hispanic.
# As before, there's no signficant difference.

fit3 <- update(fit, . ~ . + I((popn.black - popn.hispanic)/popn))
summary(fit3)


Call:
glm(formula = surv > SURV_THRESHOLD ~ borough + I((popn.black -
    popn.hispanic)/popn) + borough:nonwhite_fraction - 1, family = "binomial",
    data = df, subset = popn > 250)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-1.6017  -1.0849  -0.8297   1.1824   1.8063

Coefficients:
                                      Estimate Std. Error z value Pr(>|z|)
boroughBronx                          -1.57222    0.54729  -2.873  0.00407 **
```

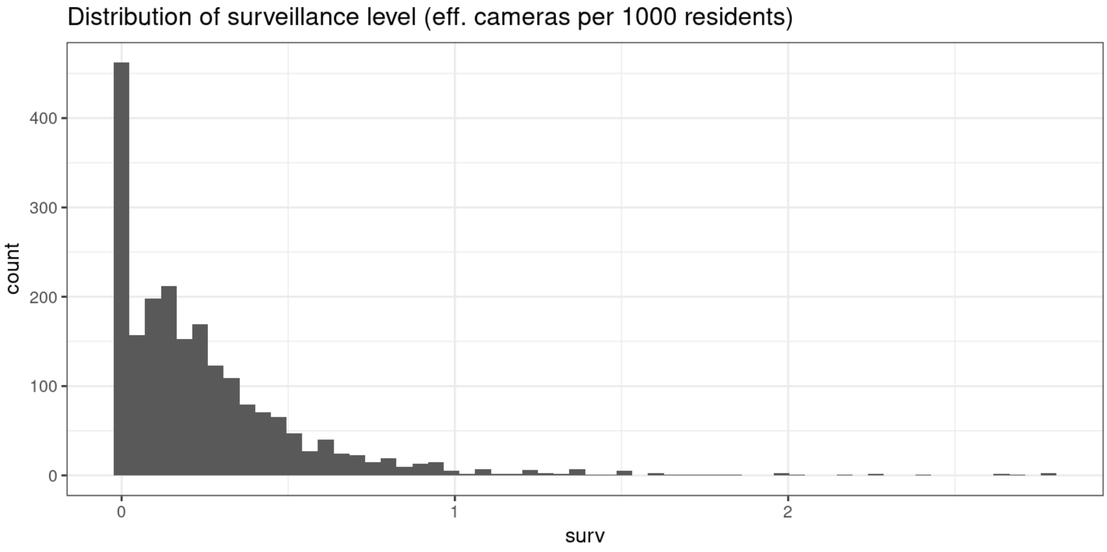
```
boroughBrooklyn                        -0.56567    0.14415  -3.924 8.70e-05 ***
boroughManhattan                        0.32716    0.20178   1.621  0.10493    
boroughQueens                          -0.09321    0.18748  -0.497  0.61905    
boroughStaten Island                   -0.33022    0.33120  -0.997  0.31874    
I((popn.black - popn.hispanic)/popn)    0.08104    0.14687   0.552  0.58110    
boroughBronx:nonwhite_fraction          1.27483    0.66973   1.903  0.05698 .  
boroughBrooklyn:nonwhite_fraction       0.43643    0.25669   1.700  0.08909 .  
boroughManhattan:nonwhite_fraction     -1.78229    0.45263  -3.938 8.23e-05 ***
boroughQueens:nonwhite_fraction         0.63678    0.30684   2.075  0.03796 *  
boroughStaten Island:nonwhite_fraction  1.66227    0.96501   1.723  0.08497 .  
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 2914.0  on 2102  degrees of freedom
Residual deviance: 2813.9  on 2091  degrees of freedom
AIC: 2835.9

Number of Fisher Scoring iterations: 4
```

2.4.2 SANITY CHECKS AND DATA PLOTS

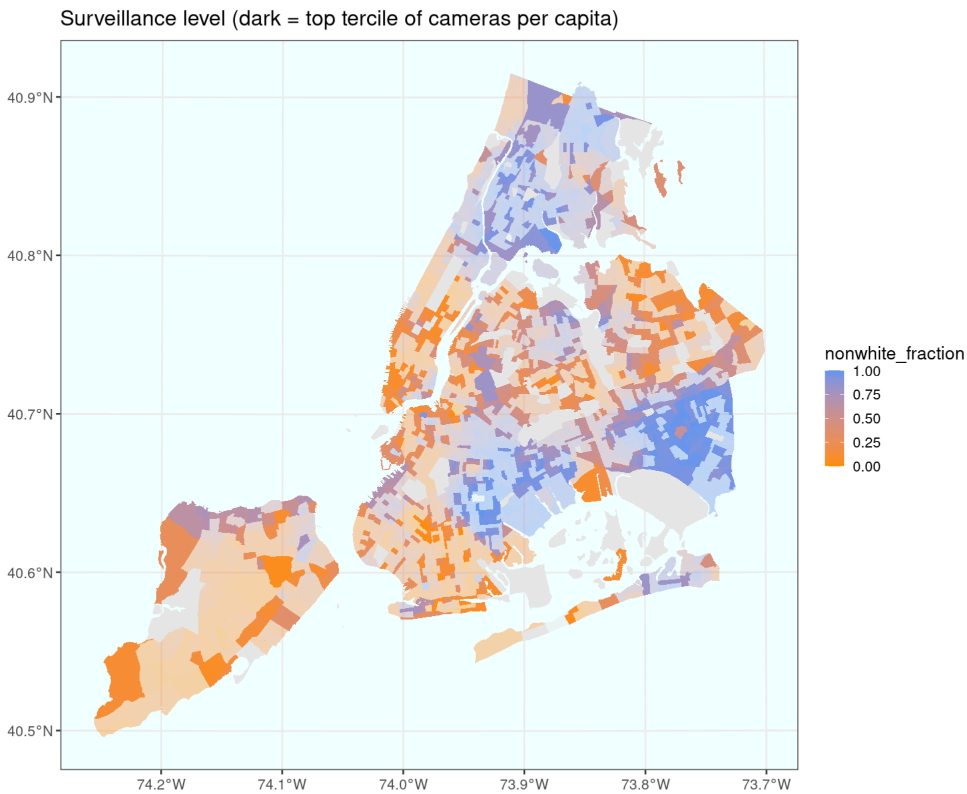
```
# Does the raw data support the link between nonwhite_fraction and higher surveilla
nce?
# In Manhattan, it's abundantly obvious.
# In other boroughs, possibly yes, but it's a small signal and so it's not surprisi
ng
# we need formal statistics to pull it out.

options(repr.plot.width=7, repr.plot.height=12)

ggplot(data=df[popn>250], aes(x=(popn.black + popn.hispanic) / (popn.black + popn.h
ispanic + popn.white), y=surv)) +
  geom_point(aes(size=popn), alpha=.4) +
  geom_smooth(method='loess', col='red', size=2, formula=y~x) +
  facet_grid(borough~.) +
  scale_size_area() +
  coord_cartesian(ylim=c(0,.8)) +
  theme_bw(base_size=16) +
  xlab('% black or hispanic') + ylab('cameras per 1000 residents')
```




```
# The distribution of `surv` is definitely non-Gaussian!
# It has a spike at surv=0, then something like a Gamma distribution, then some out
liers.
# It's not sound to fit a linear regression.
# That's why I've binarized surv, and fitted a logistic regression.

options(repr.plot.width=12, repr.plot.height=6)
ggplot(data=df[popn>250 & surv<3]) + geom_histogram(aes(x=surv), bins=60) +
  ggtitle('Distribution of surveillance level (eff. cameras per 1000 residents)')
```




```
# Map showing relationship between ethnic mix and surveillance.
# Which census tracts have a high level of surveillance?
# This is shown superimposed on nonwhite_fraction.

dft <- merge(tracts[,'GEOID'], df, by='GEOID')

options(repr.plot.width=14, repr.plot.height=10)

ggplot() +
  geom_sf(data=dft, aes(fill=ifelse(popn>250,nonwhite_fraction,NA), alpha=survF), s
ize=0) +
  scale_fill_gradient(limits=c(0,1), low='darkorange', high='cornflowerblue', na.va
lue='grey90',
                     guide=guide_colorbar(title='nonwhite_fraction')) +
  scale_alpha_manual(values=c('high'=1, 'low'=0.4), guide="none") +
  ggtitle('Surveillance level (dark = top tercile of cameras per capita)') +
  theme_bw(base_size=16) +
  theme(panel.background = element_rect(fill='azure'))
```



2.5 HOW ELSE DO STOP-AND-FRISK ACTIONS DEPEND ON SURVEILLANCE?

We have seen that the stop+frisk rate depends on surveillance level: the higher the surveillance level, the higher the rate. We now investigate this link in more granular detail.

SUSPECTED CRIME DESCRIPTION.

Does the correlation between stop+frisk rate and surveillance level depend on the suspected crime description? Yes it does: there are some suspected crimes, especially ASSAULT, CPW, ROBBERY, LARCENY, where the stop+frisk rate is highly correlated with surveillance level (making up 71% of incidents). For other suspected crimes, there is no correlation.

CHANCE OF BEING FOUND INNOCENT.

Might it be that in areas with high surveillance, the police do more uncalled-for stop+frisks, and hence there are more innocent people stopped? No. There is no correlation between the chance of being found innocent and the surveillance level.

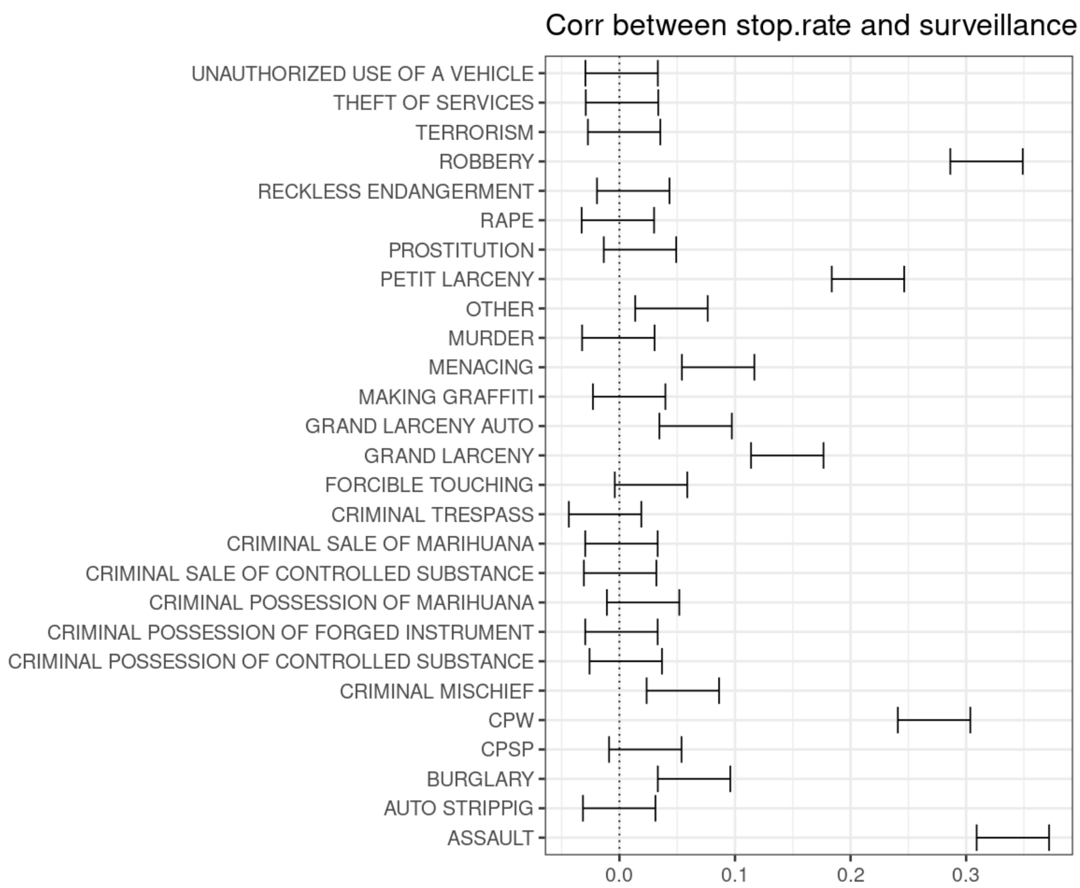
"STOPPED WHILE BLACK."

We'd expect that the stop+frisk rate should depend on the racial mix: in areas with a higher proportion of Black residents, a higher proportion of stop+frisk incidents are likely to be of Black people. Does this ratio vary according to surveillance level? No, not significantly. (The whopping great fact is that there are many more Black people stopped than other races. This is a property of the stop-and-frisk dataset, and it doesn't seem to be linked to camera surveillance, so it's outside the scope of this study of surveillance.)

2.5.1 SUSPECTED CRIME DESCRIPTION

```
# Analyse the correlation between num. stops (per capita) and num. cameras (per cap
ita), across census tracts
# Use 2019 stop-and-frisk data
```

```r
df <- as.data.table(expand.grid(SUSPECTED_CRIME_DESCRIPTION=unique(sqf$SUSPECTED_CR
IME_DESCRIPTION),
                                GEOID=unique(tracts$GEOID)))
df <- merge(df, as.data.table(sqf)[YEAR2==2019, list(numstops=.N), by=list(GEOID,SU
SPECTED_CRIME_DESCRIPTION)],
           by=c('GEOID','SUSPECTED_CRIME_DESCRIPTION'), all=TRUE)
df <- merge(df, census[, list(GEOID, popn)], by='GEOID', all=TRUE)
df <- merge(df, camera_count, by='GEOID', all=TRUE)
df <- merge(df, as.data.table(st_drop_geometry(tracts))[, list(GEOID,borough)], by=
'GEOID', all=TRUE)
df[is.na(numstops), numstops := 0] # for the tracts with no recorded stops

# Surveillance level has some off-the-scale values.
# A quick fix is to truncate. Another is to regress against rank (survC).
# Both give similar results here.
df[, surv := pmin(eff_cameras / popn * 1000, 3)]
#df[, survC := rank(surv) / .N]
fit <- lmList(numstops/popn*1000 ~ 1 + surv | SUSPECTED_CRIME_DESCRIPTION,
              data=df, subset=popn>250)

x <- summary(fit)
x <- as.data.frame(coef(x)[,,2])
x$SUSPECTED_CRIME_DESCRIPTION <- row.names(x)

options(repr.plot.width=10, repr.plot.height=8)

ggplot(data=x) +
  geom_vline(xintercept=0, linetype='dotted') +
  geom_errorbarh(aes(xmin=Estimate-1.96*`Std. Error`, xmax=Estimate+1.96*`Std. Erro
r`, y=SUSPECTED_CRIME_DESCRIPTION)) +
  theme_bw(base_size=16) +
  ggtitle('Corr between stop.rate and surveillance (2019)') + ylab('')
```




```
# What fraction of stops (in 2019) are for the five high-correlation reasons?

x <- as.data.table(sqf)[YEAR2==2019, list(numstops=.N), by=list(SUSPECTED_CRIME_DES
CRIPTION)]
x[, corr := SUSPECTED_CRIME_DESCRIPTION %in% c('CPW','ASSAULT','ROBBERY','PETIT LAR
CENY','GRAND LARCENY')]
x[, list(numstops=sum(numstops)), by=corr][corr==TRUE,numstops] / sum(x$numstops)

[1] 0.7082993
```

4.2 CHANCE OF BEING FOUND INNOCENT

```
# Simple tabulation of #stops, #innocent, per SUSPECT_RACE_DESCRIPTION

df <- as.data.table(st_drop_geometry(sqf))
df <- merge(df, camera_count, by='GEOID', all.x=TRUE)
df <- merge(df, census, by='GEOID', all.x=TRUE)

df[, innocent := SUSPECT_ARRESTED_FLAG=='N' & SUMMONS_ISSUED_FLAG=='N' & WEAPON_FOU
ND_FLAG=='N']
df[, surv := pmin(eff_cameras / popn * 1000, 3)]

df[, bwh := as.character(NA)]
df[SUSPECT_RACE_DESCRIPTION=='BLACK', bwh := 'BLACK']
df[SUSPECT_RACE_DESCRIPTION=='WHITE', bwh := 'WHITE']
df[SUSPECT_RACE_DESCRIPTION %in% c('BLACK HISPANIC', 'WHITE HISPANIC'), bwh := 'HIS
PANIC']

df[YEAR2==2019, list(n=.N, nInnocent=sum(innocent), percentInnocent=sum(innocent)/.
N*100), by=list(SUSPECT_RACE_DESCRIPTION)][order(-n)]
```

```
  SUSPECT_RACE_DESCRIPTION  n    nInnocent percentInnocent
1 BLACK                     7981 5131      64.29019
2 WHITE HISPANIC            2742 1728      63.01969
3 WHITE                     1215  755      62.13992
4 BLACK HISPANIC            1127  693      61.49068
5 ASIAN / PACIFIC ISLANDER   301  214      71.09635
6 (null)                      85   59      69.41176
7 AMERICAN INDIAN/ALASKAN N    8    5      62.50000

# Does your chance of being found innocent relate to surveillance?

fit <- glm(innocent ~ 0 + bwh + bwh:surv,
           data = df,
           subset = popn>250 & !is.na(bwh) & YEAR2==2019)
summary(fit)


Call:
glm(formula = innocent ~ 0 + bwh + bwh:surv, data = df, subset = popn >
    250 & !is.na(bwh) & YEAR2 == 2019)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-0.6788  -0.6293   0.3544   0.3662   0.4304

Coefficients:
                  Estimate Std. Error t value Pr(>|t|)
bwhBLACK          0.646634   0.006973  92.736   <2e-16 ***
bwhHISPANIC       0.631055   0.009780  64.524   <2e-16 ***
bwhWHITE          0.609913   0.017577  34.700   <2e-16 ***
bwhBLACK:surv    -0.009917   0.015486  -0.640    0.522
bwhHISPANIC:surv -0.020469   0.022521  -0.909    0.363
bwhWHITE:surv     0.022974   0.034137   0.673    0.501
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for gaussian family taken to be 0.2314808)

    Null deviance: 8177.0  on 12857  degrees of freedom
Residual deviance: 2974.8  on 12851  degrees of freedom
AIC: 17681

Number of Fisher Scoring iterations: 2

# How about if we restrict attention to suspected_crimes with a known link to surve
illance?

fit2 <- update(fit, subset = popn>250 & !is.na(bwh) & YEAR2==2019 & SUSPECTED_CRIME
_DESCRIPTION %in% c('ASSAULT','ROBBERY'))
summary(fit2)


Call:
glm(formula = innocent ~ 0 + bwh + bwh:surv, data = df, subset = popn >
    250 & !is.na(bwh) & YEAR2 == 2019 & SUSPECTED_CRIME_DESCRIPTION %in%
    c("ASSAULT", "ROBBERY"))

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-0.7014  -0.5940   0.3580   0.3838   0.5902

Coefficients:
                 Estimate Std. Error t value Pr(>|t|)
```

```
bwhBLACK          0.64603    0.01279  50.499   <2e-16 ***
bwhHISPANIC       0.59724    0.01856  32.176   <2e-16 ***
bwhWHITE          0.51712    0.04018  12.870   <2e-16 ***
bwhBLACK:surv    -0.05235    0.02681  -1.952    0.051 .
bwhHISPANIC:surv -0.06248    0.04191  -1.491    0.136
bwhWHITE:surv     0.11626    0.07280   1.597    0.110
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for gaussian family taken to be 0.2367801)

    Null deviance: 2252.00  on 3685  degrees of freedom
Residual deviance:  871.11  on 3679  degrees of freedom
AIC: 5156.9

Number of Fisher Scoring iterations: 2
```

4.3 "STOPPED WHILE BLACK"

```
# We'd expect that #stops.black / #stops ~ popn.black / popn.
# Is this relationship impacted by surveillance level?
# No.
# (Whether or not I exclude Manhattan, which is a special case.)
# (I'm using a binarized version of surveillance level, to make it easier
# to interpret the surveillance:popn.black interaction term.)

df <- as.data.table(st_drop_geometry(sqf))
df <- df[YEAR2==2019, list(numstops=.N, numstops.black = sum(SUSPECT_RACE_DESCRIPTI
ON=='BLACK')), by=GEOID]
df <- merge(df, census, by='GEOID', all=TRUE)
df <- merge(df, camera_count, by='GEOID', all=TRUE)
df <- merge(df, as.data.table(st_drop_geometry(tracts))[, list(GEOID, borough)], by
='GEOID', all=TRUE)
df[, surv := eff_cameras / popn * 1000]

fit <- lm(numstops.black/numstops ~ 1 + I(popn.black/popn)*I(surv>SURV_THRESHOLD),
         data = df,
         subset = popn>250 & borough != 'Manhattan')
summary(fit)


Call:
lm(formula = numstops.black/numstops ~ 1 + I(popn.black/popn) *
    I(surv > SURV_THRESHOLD), data = df, subset = popn > 250 &
    borough != "Manhattan")

Residuals:
     Min       1Q   Median       3Q      Max
-0.97535 -0.29650 -0.00389  0.17134  0.70482

Coefficients:
                                                Estimate Std. Error t value Pr(>|t|
)
(Intercept)                                      0.29518    0.01482  19.913   <2e-1
6 ***
I(popn.black/popn)                               0.76378    0.03564  21.433   <2e-1
6 ***
I(surv > SURV_THRESHOLD)TRUE                     0.01589    0.02159   0.736    0.46
2
I(popn.black/popn):I(surv > SURV_THRESHOLD)TRUE -0.01331    0.05049  -0.264    0.79
2
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
Residual standard error: 0.3027 on 1512 degrees of freedom
  (311 observations deleted due to missingness)
Multiple R-squared:  0.3739, Adjusted R-squared:  0.3727
F-statistic:   301 on 3 and 1512 DF,  p-value: < 2.2e-16
```

ACKNOWLEDGEMENTS

Amnesty International would like to thank the more than 7,000 digital volunteers from around the world who analysed every intersection in New York City to find and categorise surveillance cameras, gave invaluable feedback and peer-to-peer support. Without you this research would not have been possible, thank you.

ANALYSIS OF CROWDSOURCED DATA
Julien Cornebise, Ph.D. Department of Computer Science, University College London

Swetha Pillai

ANALYSIS OF STOP-AND-FRISK + CAMERA LOCATIONS
Damon Wischik, Ph.D.

ADDITIONAL ANALYSIS
Giulia Torino, Ph.D. Department of Geography, University of Cambridge

PARTNER ORGANISATION
BetaNYC

MODELLING OF CAMERA RANGES
Martyna Marciniak

INTERACTIVE WEBSITE
Superposition
Design and development. Concept in collaboration with Amnesty International.

Essex and Cambridge Digital Verification Corps
Early feedback

REFERENCES

Efron, Bradley. 2003. 'Second Thoughts on the Bootstrap'. Statistical Science 18 (2): 135–40. https://www.jstor.org/stable/3182843.

Efron, Bradley, and Robert J. Tibshirani. 1994. An Introduction to the Bootstrap. CRC press.

Cornebise, Julien, and Swetha Pillai, Martyna, Marciniak, Sophie Dyer. 2021. Decode Surveillance NYC early analysis. Citizen Evidence Lab https://citizenevidence.org/2021/11/17/decode-surveillance-early-analysis (accessed 7 January 2021)

Hayes, Andrew F., and Klaus Krippendorff. 2007. 'Answering the Call for a Standard Reliability Measure for Coding Data'. Communication Methods and Measures 1 (1): 77–89. https://doi.org/10.1080/19312450709336664.

Krippendorff, Klaus. 1970. 'Estimating the Reliability, Systematic Error and Random Error of Interval Data'. Educational and Psychological Measurement 30 (1): 61–70. https://doi.org/10.1177/001316447003000105.

———. 2004a. Content Analysis: An Introduction to Its Methodology. 2nd edition. Sage publications.

———. 2004b. 'Reliability in Content Analysis'. Human Communication Research 30 (3): 411–33. https://doi.org/10.1111/j.1468-2958.2004.tb00738.x.

———. 2011. 'Agreement and Information in the Reliability of Coding'. Communication Methods and Measures 5 (2): 93–112. https://doi.org/10.1080/19312458.2011.568376.

———. 2020. 'A Quadrilogy for Big Data Reliabilities', October.

AMNESTY INTERNATIONAL IS A GLOBAL MOVEMENT FOR HUMAN RIGHTS. WHEN INJUSTICE HAPPENS TO ONE PERSON, IT MATTERS TO US ALL.

CONTACT US

info@amnesty.org

+44 (0)20 7413 5500

JOIN THE CONVERSATION

www.facebook.com/AmnestyGlobal

@Amnesty

DECODE SURVEILLANCE NYC

METHODOLOGY

Since 2016, Amnesty Decoders has leveraged microtasking and participatory methods from citizen science to address large-scale research questions in human rights.

Decode Surveillance NYC launched in May 2021. Over ten weeks, more than 7,000 digital volunteers from around the world analysed every intersection in New York City. The effort found and categorised tens of thousands of CCTV cameras, revealing for the first time which areas of the city are most exposed to surveillance via facial recognition technology.

INDEX: AMR 51/5205/2022
FEBRUARY 2022
LANGUAGE: ENGLISH

AMNESTY INTERNATIONAL