Technical Appendix

Project Name: Streamlining income verification to broaden access to rental assistance

Date Finalized: 4/8/2024

Overview

The purpose of this document is to provide technical details on the analysis. These include decisions not pre-specified in the plan and additional results and analyses not reported in the main abstract. See the analysis plan and abstract for all other details.

Details and decisions not pre-specified in analysis plan

Ex post diagnosis of research design and departure from analysis plan

To constitute a valid estimator of the difference-in-differences (DiD) estimand, the estimator we pre-registered needs to estimate the difference between two differences: first, the difference in application approval rates for FSP and non-FSP ZIP codes *before* the FSP conceivably had any effect on approval rates; second, the difference in application rates for FSP and non-FSP ZIP codes *after* the FSP began to affect approval rates. Estimating this first difference correctly requires identifying applications from FSP ZIP codes that had *not yet* been affected by the FSP. Our original plan involved using the application date to identify such applications: by using applications submitted prior to the implementation of FSP, we reasoned, we could estimate the first difference, between applications in FSP areas and non-FSP areas prior to the impact of FSP. Since the FSP was implemented on 6/1/2021, we assumed applications submitted prior to this date did not benefit from the FSP, even if they were submitted from ZIP codes that benefited from it after this date.

However, upon receiving the data we noticed that nearly 97% of applications received in the pre-FSP period from FSP ZIP codes were coded "yes" for a flag that appeared to indicate whether FSP had been used (`inc_eli_by_fb_proxy`). Our partners in Kentucky subsequently confirmed via email that they had used FSP to clear all existing applications that had not yet been approved, irrespective of when they had been submitted. In principle, this may have left some applications to use to estimate the pre-FSP difference, since at least some would have been fully processed prior to 6/1/2021. In practice, however, the timing did not allow for this. The FSP was implemented 103 days after the first application was received and the average number of days to payment in the pre-FSP period in non-FSP areas was 93 days. This meant that almost all applications from FSP areas submitted in the "pre-FSP" period did in fact benefit from the FSP. Figure 1 illustrates this point.

Figure 1. Percentage of applications flagged as having benefited from FSP, by date

We suspected this discovery invalidated our pre-registered analysis and validated this through a simulation study described below, which showed that our pre-specified approach was biased downwards if in fact the FSP had an impact on applications from the pre-FSP period erroneously treated as though they had not been affected by the FSP. The pre-specified estimator was biased downward, because we are effectively differencing out the treatment effect for the ostensibly untreated, pre-treatment periods.

Fortunately, our evaluation did not require temporal variation to yield causally-valid insights. There are two main alternative approaches we considered while writing the pre-analysis plan: the regression discontinuity design (RD) and adjustment for confounders (AFC) approach. We initially selected the difference-in-differences (DiD) estimator over the alternatives because it performed better under the conditions of a simulation study, in which there was a large control group and a long, valid pre-treatment period. In practice, we have an invalid pre-treatment period and a very small control group, so we revisited the two alternatives.

In addition to the simulation study conducted prior to receiving the data, used to inform the analysis plan, we conducted a second simulation study in which we compared the bias and standard error of different estimation strategies in order to choose the best alternative to the pre-registered analysis, using the actual data. The detailed description of the simulation steps is below. At a high-level, the simulation seeks to replicate a simplified version of the actual random assignment to FSP that occurs due to the random renter median income estimates produced by the ACS, upon which FSP determinations relied. We simulate other possible assignments of ZIP codes to the FSP, and simulate an outcome in which the FSP increases the probability of application approval by roughly six percentage points. Importantly, the simulation stipulates that this causal effect kicks in from the very first application in the data, rather than at the date FSP was implemented (which is what the first approach had assumed). The simulation then reveals the outcomes we would have observed under those alternative assignments by comparing the simulated estimate of renter median income to the county-level AMI limits. The simulation simplifies the actual study somewhat by ignoring data suppression and simulating as though renter median income is observed in all ZIP codes. We have no reason to believe this simplification would invalidate the comparisons between the different estimation approaches. Note that the simulations still keep the June 1, 2021 date as the marker separating pre- from post-FSP periods, but the simulations build in a treatment effect right from the start of the time series. In other words, we are simulating a scenario that resembles one that we are facing in practice, where applications submitted during the pre-FSP period are actually affected, ex-post, by the implementation of FSP.

We compare the following five estimation approaches:

1. **Difference-in-Differences (DiD)**: we estimate this as pre-registered, in order to understand potential issues in this estimation approach. The difference in the pre-post trends in approval for applications from FSP and non-FSP areas is estimated by regressing the approval outcome on an indicator for the post-period, an indicator FSP, and their interaction. We additionally control for confounders of the relationship between FSP and approval, namely the median renter income estimate and the suppression indicator. The size of the renter population in the ZIP code is also included to account for the fact that data suppression happens to ZIP codes with differing probabilities depending on the sample size.

2. **Adjustment for confounders (AFC)**: this regression is the same as the DiD, albeit with the post-period indicator removed. In essence, the estimate of the difference in approval rates for FSP and non-FSP areas is estimated while controlling for the confounders listed above.

3. **Adjustment for confounders with month fixed effects (AFC + month FE)**: this is the same as the AFC regression, but includes a control for temporal trends.

4. **Inverse propensity weighting (IPTW)**: this is the same regression model as the AFC, however: a) the regression is weighted by an inverse propensity weight that corresponds to the probability we observe ZIP codes in the FSP or non-FSP condition we do in fact observe them in; b) we drop any applications from ZIP codes whose probability of being in either condition was zero.

5. **Regression discontinuity design (RD)**: using the distance of the median renter income estimate to the county AMI threshold for FSP as the running variable, the regression discontinuity approach extrapolates polynomial regression functions to estimate the

difference in application approval rates for the FSP and non-FSP applications right at the threshold of passing from non-FSP to FSP.

See below for an in-depth description of how each quantity is estimated. The **Estimand** column tells us the true, underlying effect that we simulated FSP to have on the potential outcomes of the applications. The **Mean estimate** tells us the average estimate of this quantity that the examined strategy provided across all simulations. The **Bias** tells us the difference between the **Mean estimate** and the **Estimand**. The **SE(Bias)** tells us the Monte Carlo standard error of the **Bias** estimate (basically, whether we conducted a sufficient number of simulations to infer that any apparent bias we see doesn't result from random variation in the simulations themselves). The **SD(Estimate)** tells us the standard deviation of the estimates across the simulations, and can be considered as roughly equivalent to the "true" standard error that the standard error estimator is typically trying to estimate. Because the ZIP-clustered standard error estimators included in the packages we used took a long time to compute, and are all ultimately just trying to estimate SD(Estimate), we don't bother estimating standard errors on each simulation and instead focus on the "true" standard error, SD(Estimate), which gives us a relative measure of how imprecise each estimation strategy is. Finally, the **RMSE** is the square root of the average of the squared difference between the estimate and the estimand. It gives us a sense of how different the estimate is from the estimand, on average, due either to bias or the standard error.

The results of the simulations are displayed on the table below.

Table 1. Results of the simulation study

Estimator	Estimand	Mean estimate	Bias	SE(Bias)	SD(Estimate)	RMSE
DiD	0.06	0.01	-0.05	0.0006	0.03	0.06
AFC	0.06	0.14	0.08	0.0010	0.05	0.10
AFC + month FE	0.06	0.14	0.08	0.0011	0.05	0.10
IPTW	0.06	0.09	0.03	0.0005	0.02	0.04
RD	0.06	0.05	0.00	0.0026	0.12	0.12

We point out the following noteworthy results:

- **DiD is downwards-biased.** As expected, erroneously analyzing applications submitted in the pre-FSP period as though they were unaffected by FSP leads us to under-estimate the true effect.
- **AFC is upwards-biased.** Both adjustments for confounders approaches do not fully account for the confounding — in fact, we estimate an impact that, on average, is twice as large as the true underlying effect. This probably arises because, while ZIP codes are randomly assigned to FSP via the Census' estimation procedure, they are not assigned with

equal probabilities: all else equal, applications from ZIP codes that have median renter income estimates that are further away from the county AMI threshold or have smaller standard errors makes them more likely to fall only in FSP or non-FSP conditions.

- **IPTW accounts for much of the bias in the AFC approach.** The IPTW estimator gets very close to the true answer, exhibiting minimal bias.
- **RD is the least biased estimator but is very imprecise**. The RD gets bias very close to zero. However, its standard error is more than twice as large as any other estimator.
- **IPTW fares better than RD in terms of RMSE.** Even though RD is less biased than IPTW, IPTW nevertheless produces answers that are closer to the truth more often (according to the RMSE) because its standard error is six times smaller.

The simulation works as follows:

1. Take the actual data used in the study
2. Create new variables:
    a. Simulated treated potential outcome, Y(T = 1), which is made by duplicating the observed approval outcome
    b. Simulated control potential outcome, Y(T = 0), which is made by randomly switching some of the 1s in Y(T = 1) to 0, such that the average treatment effect is 6 percentage points
    c. A standard error, S, for every estimate of renter median income in every ZIP. Where we have an estimate from the ACS, we turn the margin of error into a standard error by dividing the MOE by 1.645. Where we don't have an estimate of the renter median income from the ACS due to suppression and instead predicted it, we impute the associated standard error using the square root of the average of the observed squared standard errors.
3. We then repeat the following steps 2,000 times:
    a. Create a simulated median renter income estimate, X, by adding mean-zero, normally-distributed and randomly-generated error to the actual median renter income variable used in the study, with standard deviation equal to the standard error created above.
        i. **Note**: This effectively treats the observed median renter income estimate as the true underlying mean that is being estimated, and the standard error estimate as the true underlying standard deviation of the sampling distribution.
    b. Create a simulated FSP variable, Z, which is 1 for any application whose X falls below their county AMI and 0 otherwise.

c. Create the associated running variable.

d. Create an estimated probability of being treated by using the CDF of the normal distribution, with mean X and standard deviation S, and use it to create an inverse propensity weight. Flag any applications from ZIP codes whose probability of being treated was 0 or 1 because their simulated median renter income estimate, X, was too low or too high relative to S such that they did not have any chance of being in either the FSP or non-FSP conditions.

e. Reveal the simulated outcome, Y, as a function of Z: Y = Y(1) * Z + Y(0) * (Z-1)

f. Estimate and store the point estimates from the approaches listed above

4. This exercise gives rise to 2,000 simulations of five different point estimators. In each case, the estimand they are trying to estimate, the average treatment effect, is unchanged (6 percentage points). Using the 10,000 simulated estimates and the estimand, we calculate the following quantities:

   a. Bias: This is the average of the difference between the point estimate of the estimator and the estimand

   b. SE(Bias): This is the Monte Carlo standard error of our estimate of the bias. This can be arbitrarily reduced by increasing the number of simulations, and helps to ensure that any bias we are estimating is not due to random variation in the simulations alone.

   c. SD(estimate): This is the standard deviation of the estimates. It gives us a sense of how imprecise the estimation strategy is.

   d. RMSE: This is the square root of the average of the squared difference between the estimate and the estimand. It gives us a sense of how different the estimate is from the estimand, on average, due either to bias or the standard error.

Standard error estimation: bootstrapping versus parametric standard errors

In this study, applications are affected by the FSP (the "treatment" in our quasi-experimental analogy) at the ZIP code level. As such, all analyses account for this clustering by allowing for error correlation at the ZIP code level in the estimation of standard errors. Our analysis plan specified that we would employ the "parametric" approach to standard error estimation, which we implemented in the RD and IPTW analyses using the closed-form solutions that are included as part of the `rdrobust` and `estimatr` packages, respectively. However, because the clusters exhibit very different sizes (with some ZIP codes only having one or two applications and others many thousands), as a robustness check we also considered the "bootstrapping" approach to estimating clustered standard errors. This approach involved repeating the following steps 200 times:

1. **Cluster-resample of FSP ZIP codes:** Resample, with replacement, all 396 ZIP codes in the analytic sample where FSP was used. When resampling a ZIP code, retain all applications from that ZIP code as-is.
2. **Cluster-resample of non-FSP ZIP codes:** Resample, with replacement, all 293 ZIP codes in the analytic sample where FSP was not used. When resampling a ZIP code, retain all applications from that ZIP code as-is.
3. **Within-cluster resample of applications:** Loop through all ZIP code clusters resampled in steps 1 and 2. In each ZIP code, resample with replacement all applications within that ZIP code.
4. **Re-estimate analyses, and store the results.**

The standard deviation of the 200 estimates obtained from these steps provides the bootstrap standard error. We also implement a method that omits step 3, where there is no within-cluster resampling.

When comparing the parametric and bootstrapped standard errors, the IPTW results look very similar (see Table 2 below). Our results are robust to bootstrapping. If anything, the parametric standard errors reported in the main analysis appear more conservative.

Table 2. Standard error estimation for IPTW model

Note: Because the parametric standard errors that were pre-specified in the analysis plan are appropriately larger after clustering than the non-clustered SEs, the abstract uses the pre-specified approach to standard error estimation. We report the results from bootstrapping to show that the method does not alter the statistical significance of the main results.

Standard error estimation method	Standard error estimate on main IPTW result (application approval)
No clustering (robust SEs from `lm_robust` within `estimatr`)	0.016
Parametric clustering (cluster robust SEs from `lm_robust` within `estimatr`)	0.027
Cluster-bootstrap (with within-cluster resampling)	0.012
Cluster-bootstrap (no within-cluster resampling)	0.013

However, as Table 3 below illustrates, the parametric clustered standard errors on the RD estimate are three times smaller than the bootstrapped clustered standard errors. Moreover, cluster-robust SEs produced using the `rdrobust` package were smaller than the non-clustered SEs estimated using the same package. In the interests of avoiding overconfidence, we thus opted to depart from the analysis plan and use the more conservative of the two bootstrapped standard error approaches. The bolded row indicates the estimate used in the main abstract, the row in italics indicates the pre-specified approach.

Table 3. Standard error estimation for the RD model

Note: To minimize the risk of over-confidence, the abstract employs a more conservative approach to standard error estimation (in bold) for the RD than the approach originally specified (in italics).

Standard error estimation method	Standard error estimate on main RD result (application approval)
No clustering (robust SEs from `rdrobust`)	0.037
Parametric clustering (cluster robust SEs from `rdrobust`)	*0.011*
Cluster-bootstrap (with within-cluster resampling)	**0.033**
Cluster-bootstrap (no within-cluster resampling)	0.028

Robustness checks omitted due to switch from DiD to RD/IPTW

- Because the DiD analysis was invalidated by the discovery, as described above, that there is no pre-FSP period unaffected by the FSP, one pre-registered robustness check — "interaction of DiD with controls" — was omitted from this document. This robustness check relied on the DiD specification in particular and so is omitted here.

Additional details on main results

Predictive modeling of median household income for renters

Our analytic sample consists of applications from ZIP Code Tabulation Areas (ZCTAs) whose median renter income is suppressed by the Census due to small sample sizes and those whose median renter income is observed. In order to include the suppressed ZCTAs in analyses that condition on an estimate of median income, we used machine learning to predict that value.

Our process was as follows:

1. Begin with all ZCTAs nationwide (N = 23,484) for whom median renter income is observed — this includes both ZCTAs inside and outside Kentucky.
2. Split the data into three subsamples: a 20% held-out test set and an 80% training set that is further split into 80% used for training and 20% used for hyperparameter tuning.
3. Predict median renter income at the ZCTA level using the following predictors:
    a. State id
    b. ZCTA-level:

        i. Total population
        ii. White population
        iii. Renter population
        iv. Renter population denominator (different from total population due to not being measured for all respondents)
        v. Median household income
        vi. Total population for ZCTA blocks that overlap with the county
    c. County-level:
        i. Total population
        ii. White population
        iii. Renter population
        iv. Renter population denominator (different from total population due to not being measured for all respondents)
        v. Median household income
        vi. Median household income among renters
4. For each classifier, tune the hyperparameters using the RMSE (root mean squared error) metric.
5. Select the best-performing model by examining the following metric among the held-out test set: percentage of predictions within 10,000 dollars of the correct median renter household income. We chose this metric because it was important for the regression discontinuity estimator to be correct within a relatively narrow bandwidth.

In our initial confirmatory analysis, we compared two models: 1. a random forest with 100 trees, where the main hyperparameters we tuned governed how many input features/predictors to consider within a given tree and the minimum number of data points required to split a node; and 2. LASSO, where the penalty parameter was chosen via tuning. Random forest outperformed LASSO for our chosen metric (see exact numbers below) and was used in the main confirmatory analysis.

In a robustness check, we compared random forest to four additional types of classifiers: a neural network-based approach, a support vector machine, a gradient boosting algorithm (xgboost), and a k-nearest neighbors based algorithm. The table below shows that the neural network *underperformed* our chosen model. This could be due to the fact that the input data contained a limited number of predictors, with neural networks performing better on more high-dimensional data with more predictors. The performance of the other models was similar to the random forest, scoring similarly using either a $10,000 difference bandwidth for our metric or a smaller $5,000 window, with gradient boosting slightly outperforming random forest. In our robustness check below, we use the three similarly-performing models — SVM; KNN; and xgboost — to construct our measures of renter income and the running variable used in RD. We see results are nearly identical to the main findings.

Table 4. Comparison of accuracy of predictive models for median renter household income

Model	Percent predictions within $10,000 of actual median renter income	Percent predictions within $5,000 of actual median renter income
Random forest (used in confirmatory analysis)	87.9	73.6
Gradient-boosting	88.3	74.9
SVM	87.1	71.6
KNN	85.5	70.6
Neural network	82.8	64.2

Additional results

DiD results of impact of FSP on approval from pre-analysis plan specification

Our original analysis plan pre-specified a difference-in-differences design (pg. 11). As explained above and in the Abstract (pg. 2), this approach was invalidated as the necessary assumptions were not met by program implementation. Here, we report the originally pre-specified model.

Our primary coefficient of interest is the interaction between an FSP indicator and the post-period indicator. This result is positive and consistent in sign with our primary results, though statistically insignificant. The results are smaller in magnitude than our primary specification. This is consistent with the bias we expect this estimator to exhibit, given the results of the simulation study described above.

Table 5. DiD Results for Impact of FSP on approval rates

	Estimate	Standard Error	p-Value	Lower 95% CI	Upper 95% CI
Intercept	0.431	0.043	0.0000	0.346	0.516
Post	-0.065	0.022	0.0048	-0.110	-0.020
FSP	0.120	0.033	0.0005	0.055	0.186
Post x FSP	0.036	0.024	0.1308	-0.011	0.083
Median renter income	0.000	0.000	0.0608	0.000	0.000
Renter population size	0.000	0.000	0.0313	0.000	0.000
Suppressed	-0.009	0.031	0.7788	-0.070	0.053

Robustness check: method for predicting median renter income

As shown above, three predictive models performed similarly to the random forest used to predict suppressed median income estimates in the main results: a support vector machine (SVM), k-nearest neighbors (KNN), and a boosted tree-based approach (xgboost). We here show that our results are robust to these alternative prediction algorithms.

First, examining the IPTW results, we see that the magnitude, direction, and statistical significance of our main coefficient of interest (an indicator for FSP) remains nearly identical to the main results.

Table 6. IPTW results for impact of FSP on approval results using SVM-based predictions

	Estimate	Standard Error	p-Value
Intercept	0.381	0.039	0.0000
FSP	0.135	0.027	0.0001
Median renter income	0.000	0.000	0.0639
Renter population size	0.000	0.000	0.0118
Suppressed	0.005	0.032	0.8656

Table 7. IPTW results for impact of FSP on approval results using XGboost-based predictions

	Estimate	Standard Error	p-Value
Intercept	0.383	0.039	0.0000
FSP	0.134	0.027	0.0001
Median renter income	0.000	0.000	0.0700
Renter population size	0.000	0.000	0.0117
Suppressed	0.005	0.031	0.8691

Table 8. IPTW results for impact of FSP on approval results using KNN-based predictions

	Estimate	Standard Error	p-Value
Intercept	0.383	0.039	0.0000
FSP	0.134	0.027	0.0001
Median renter income	0.000	0.000	0.0700
Renter	0.000	0.000	0.0117

population size			
Suppressed	0.005	0.031	0.8691

For the RD with bootstrapped SEs, we also see that the magnitude, direction, and statistical significance of our main coefficient of interest remains nearly identical to the main results. For xgboost, the model that outperformed the model used in the main confirmatory analyses (random forest), the estimate was similar in magnitude (7.0 for the RD using the xgboost predictions as the running variable; 7.5 for the main confirmatory RD using the random forest predictions as the running variable) and the confidence intervals help us reject the null that the coefficient is equal to zero ([1.3, 12.7]). SVM and KNN had confidence intervals that slightly crossed zero, but they performed less well in terms of predictive accuracy than the random forest model used in the confirmatory analysis (see Table 4).

Table 9. RD results for impact of FSP on approval rates using alternative ML-based predictions

	RD estimate	Standard Error	Lower 95% CI	Upper 95% CI
xgboost	0.070	0.029	0.013	0.127
svm	0.051	0.031	-0.010	0.112
knn	0.071	0.037	-0.001	0.144

Robustness check: Accounting for targeted outreach

We pre-specified an analysis that would control for various marketing outreach efforts conducted by KHC (pg. 20 of Analysis Plan). We implement this with a modified version of our IPTW analysis, which adds an extra control for ZIP codes that experienced marketing campaigns from KHC. These results are presented below.

The magnitude, direction, and statistical significance of our main coefficient of interest (an indicator for FSP) remain similar. The coefficient on the marketing indicator is small, near zero, and not statistically significant.

Table 10. IPTW results for impact of FSP on approval rates when accounting for marketing outreach

	Estimate	Standard Error	p-Value	Lower 95% CI	Upper 95% CI
Intercept	0.387	0.039	0.0000	0.308	0.466
FSP	0.134	0.027	0.0001	0.077	0.190
Median renter income	0.000	0.000	0.0768	0.000	0.000
Renter population size	0.000	0.000	0.0086	0.000	0.000
Suppressed	0.009	0.031	0.7764	-0.054	0.072
Marketing	-0.007	0.010	0.5110	-0.026	0.013

We also analyzed the robustness of the RD results to this additional covariate. We see that the RD results are substantively unchanged and statistically significant (estimate: 7.5 pp, 95% CI [0.8, 14.2]).

Robustness check: Total count of applications

Here, we implement a robustness check testing whether FSP increased the number of applications submitted to KHC (described on pg. 21 of the Analysis Plan). We implement this by running a modified version of our zip-level IPTW analysis, where the outcome is the total number of applications submitted from each ZIP code.

Our results are shown below. Our main coefficient of interest, an indicator for FSP, is not statistically significant. The estimate is also substantively small. This is consistent with our expectation that, because the FSP was not publicized, it did not influence applicants' decisions to apply to the Kentucky ERA program.

Table 11. IPTW results for impact of FSP on total applications at the zip code level

	Estimate	Standard Error	p-Value	Lower 95% CI	Upper 95% CI
Intercept	2.770	0.238	0.0000	2.303	3.237
FSP	0.099	0.186	0.5959	-0.267	0.464
Median renter income	0.000	0.000	0.0988	0.000	0.000
Renter population size	0.001	0.000	0.0000	0.001	0.001
Suppressed	-0.790	0.163	0.0000	-1.111	-0.470

Robustness check: Categorical specification of median income

Here, we implement a robustness check testing whether the effect of FSP on application approval is robust to an alternative specification of median income (described on pg. 21 of the Analysis Plan). In particular, this method does not require predicting suppressed median income estimates. We instead split the median income variable into indicators corresponding to deciles of the observed estimates and create a separate category, "suppressed," for ZIPs with suppressed median income. We can only run this analysis for the IPTW model, since the categorical version of the running variable is not compatible with the RD estimator.

Our results are shown below. Our main coefficient of interest, an indicator for FSP, is statistically significant and has similar magnitude to the main results.

Table 12. IPTW results for impact of FSP on approval rates using categorical specification of median renter income

	Estimate	Standard Error	p-Value	Lower 95% CI	Upper 95% CI

Intercept	0.294	0.040	0.000	0.210	0.378
FSP	0.122	0.025	0.000	0.070	0.174
Median renter income decile: 10	0.155	0.036	0.001	0.077	0.232
Median renter income decile: 2	0.048	0.042	0.278	-0.045	0.141
Median renter income decile: 3	0.176	0.044	0.003	0.075	0.277
Median renter income decile: 4	0.152	0.037	0.003	0.068	0.236
Median renter income decile: 5	0.146	0.038	0.004	0.060	0.232
Median renter income decile: 6	0.172	0.037	0.001	0.088	0.257
Median renter income decile: 7	0.135	0.037	0.005	0.052	0.219
Median renter income decile: 8	0.159	0.038	0.002	0.073	0.246
Median renter income decile: 9	0.130	0.035	0.004	0.051	0.208
Median renter income: suppressed	0.133	0.045	0.006	0.041	0.224
Renter population size	0.000	0.000	0.010	0.000	0.000

Robustness check: Alternative ways of designating ZCTAs as FSP eligible

Our main results code ZCTAs as ineligible or eligible for FSP using the same designations as those contained in the original datasets provided by Kentucky. This coding relies on an exercise via which ZCTAs are matched to specific counties, because eligibility thresholds for the FSP (and the ERA program) are defined at the county level. For the most part, it is easy to match ZCTAs to counties as the former are nested within the latter in many cases. There are, however, a handful of ZCTAs that cross counties where a decision must be made about which county to match the ZCTA to. We here show that alternative ways of mapping ZIP codes generally do not change the FSP status of ZCTAs, and furthermore that in the one case where they do, this does not change the results. We compared three ways of matching ZCTAs to counties for the purpose of comparing ZCTA-level median renter income to the county-level AMI:

- Spatial overlap: We intersected block-level shape files with ZCTA shape files. We then found the county that had the highest population overlap with the ZCTA according to this spatial intersection. This approach produces 15 differences in county matches when compared to the dataset used in practice, and no differences in the resulting FSP status.
- Land area: Using an existing crosswalk,[1] we matched each ZCTA to the county with the highest land area overlap. This approach produces 11 differences in county matches when compared to the dataset used in practice, and one difference in the resulting FSP status.
- Population overlap: Using the same existing crosswalk, we matched each ZCTA to the county with the highest population overlap. This approach produces 11 differences in county matches when compared to the dataset used in practice, and one difference in the resulting FSP status.

The ZIP code that has one different FSP status is the same in both the land area and population overlap methods, so we re-estimated the main results with that new FSP indicator. Our results, shown on Table 13, are robust to this alternate coding.

Table 13. IPTW results for impact of FSP on approval rates using alternative FSP designation

	Estimate	Standard Error	p-Value	Lower 95% CI	Upper 95% CI
Intercept	0.392	0.039	0.0000	0.315	0.470
FSP	0.127	0.027	0.0001	0.071	0.182
Median renter income	0.000	0.000	0.0871	0.000	0.000
Renter population size	0.000	0.000	0.0118	0.000	0.000
Suppressed	0.001	0.031	0.9635	-0.061	0.064

Methodology for number of approvals due to FSP

To generate the estimated number of approvals due to FSP, we use the IPTW model from the main confirmatory analysis to predict approval status in the treatment group assuming the treatment indicator is set to non-FSP. We use the approval probability in a binomial draw to generate a binary approval prediction, repeating this process m = 1,000 times and use Rubin's Rules to combine the resulting estimates. We subtracted these predicted approvals under no FSP from the observed approvals in the treatment group (43,480). This gives us a difference of 9,827 approvals predicted to occur due to FSP with a 95% confidence interval of between 9570 and 10,085 additional approvals. We report 9500 in the main text as the rounded lower bound.

[1] These crosswalks are derived from the Missouri Census Data Center's geocorr tool (https://mcdc.missouri.edu/applications/geocorr2018.html). Data was pulled September 2021. The crosswalk provides a mapping between ZCTAs and counties based on a similar approach to our spatial overlap approach that uses block-level data. But instead of performing a spatial overlap, the crosswalk draws upon an existing database the developers have created (MABLE) of geographic correspondences.

Estimating who benefits from FSP

In the main abstract, we report results that FSP helped in the approval of over 9,500 applications that would not have been approved in the absence of FSP. Importantly, we do not think 9,500 *more* applications were approved overall, since the program was oversubscribed. Rather, if the number of approvals is fixed by ERA budget totals, the 9,500 results from a shift in *which applications* got approved. Therefore, a question arises: do the people who FSP helped to get approved look different from the average applicant in an FSP ZIP code? What is different about those people who would not have been approved, *but for* the implementation of FSP? This analysis was not pre-registered.

To investigate this question, we take the following steps 1000 times:

1. **Bootstrap resample the data, using the procedure described above.** By resampling the data before estimating the model, we can ensure that the predictions generated in the next step incorporate the sampling variation represented by the standard error estimates.
2. **In each bootstrap resample, estimate the effect of FSP and use the model to predict the probability of approval for applications from FSP ZIP codes, supposing FSP never existed (setting the FSP indicator to 0).** We estimate a regression model that accounts for a differential effect of the FSP by different demographic attributes. In the simulation used to estimate the 9500 application number in the abstract, we use the IPTW model for approval, which estimates approval probabilities as a function of FSP status and zip code-level controls (renter median income; renter population; and suppressed status). In this simulation,we supplement this model with interactions between the FSP indicator and applicant-level demographic characteristics: race/ethnicity (coded as white non-Hispanic; Black non-Hispanic, Hispanic/Latino, or other), female, disability status, veteran status, extremely low-income, and rural (zip code level). The inclusion of these interactions enables us to make predictions specific to each demographic group.
3. **Generate counterfactual predictions of approval in the absence of FSP.** Using the probabilities of approval predicted in the previous step, we generate counterfactual predictions of approval in the absence of FSP for each applicant using a binomial draw with each applicant's predicted approval probability. We code an applicant as "approved due to FSP" if they were actually approved but are predicted to not have been approved when their FSP status is set to 0. This is the group of applications in our study we think would not be approved *but for* the FSP.
4. **Compare the demographic characteristics of the applications approved due to FSP to the demographic characteristics of all applications from FSP ZIP codes.** We then compare the demographic characteristics of two groups: (1) the applications predicted to be approved due to FSP in step 2 and (2) the full sample of applications from FSP ZIP codes. We estimate the differences in the demographic proportions between these two groups, as well as the associated standard errors.

This procedure provides 1000 sets of demographic differences and their standard errors. We treat each of the *m* = 1,000 predicted approvals as its own dataset, and use Rubin's Rules to combine the point estimates and standard errors. Table 14 shows the results for: (1) the observed proportions of each demographic group in the FSP-eligible population, (2) the proportions of each demographic group in the subsample of those who are predicted to be approved due to FSP, (3) the difference between the two proportions, and (4) 95% confidence intervals constructed using standard errors that take into account the variability across the *m* = 1,000 bootstrapped, random predictions.

Table 14. Do applicants predicted to be approved as a result of FSP look different from the full set of applicants from FSP ZIP codes? Test of differences in proportions

			Difference (approved due to FSP - all FSP eligible)			
Demographic attribute	**% of all FSP-eligible applications**	**% of applications predicted to be approved due to FSP**	**Percentage point (absolute difference)**	**Percent (relative difference)**	**Lower 95% CI**	**Upper 95% CI**
Female	71.35	69.18	-2.17	-3.04	-3.76	-0.59
Male	28.39	30.55	2.17	7.64	0.59	3.74
Extremely low income	68.10	66.58	-1.52	-2.23	-3.22	0.19
Rural	52.50	53.98	1.48	2.83	-3.98	6.95
Has disability	17.48	16.87	-0.61	-3.48	-1.93	0.72
Veteran	4.15	4.14	-0.02	-0.41	-0.80	0.77
White non-Hispanic	66.73	67.49	0.77	1.15	-2.69	4.22
Hispanic/Latino	2.51	3.80	1.29	51.37	0.68	1.89
Black non-Hispanic	27.05	24.72	-2.33	-8.60	-5.73	1.08

Table 14 presents the results and offers some clues as to the types of applicants who were most likely to encounter barriers due to income verification. Broadly, those predicted to have received ERA due to the income verification streamlining are more likely to be male[2] and/or Hispanic/Latino when compared with the broader pool of applicants from FSP areas. It is worth noting, as we report in the abstract, that FSP generally had a positive impact on approval rates for each of the different groups we examined. These results do suggest however that income verification requirements may have disparate impacts, pointing to interesting possible directions for future

[2] Applicants reported their gender as female, male, gender non-confirming, trans female, and trans male. We could not estimate separate interactions for individuals identifying as gender non-conforming and trans due to insufficient sample size.

research. For example, user research could help uncover and understand why certain groups of applicants may find income verification procedures more burdensome than others.